Weak Flow Cover Inequalities for the Capacitated Facility Location Problem

P.Avella[a], M. Boccia[b], S. Mattia[c], F. Rossi[d]

[a]*Dipartimento di Ingegneria (DING), Università del Sannio, avella@unisannio.it.*
[b]*Dipartimento di Energia Elettrica e Tecnologie dell'Informazione (DIETI), Università di Napoli Federico II, maurizio.boccia@unina.it.*
[c]*Istituto di Analisi dei Sistemi ed Informatica (IASI), Consiglio Nazionale delle Ricerche (CNR), sara.mattia@iasi.cnr.it. The author is a member of GNAMPA (INdAM).*
[d]*Dipartimento di Ingegneria e Scienze dell'Informazione e Matematica (DISIM), Università di L'Aquila, fabrizio.rossi@univaq.it.*

Abstract

The Capacitated Facility Location Problem calls for opening a set of facilities with capacity constraints, with the aim of satisfying at the minimum cost the demands of a set of customers. We present a new class of valid inequalities, the Weak Flow Cover inequalities. We show that Weak Flow Cover inequalities can be separated in polynomial time and turned into violated Flow Cover inequalities. In this way, we are able to provide a polynomial separation heuristic for the latter. Embedding the separation procedure into a cut-and-branch approach, we get results significantly better than those reported in the recent literature both for the lower and the upper bounds.

Keywords: Facility Location, Flow Cover, separation, valid inequalities, cut-and-branch

1. Introduction

Consider a set of customers to be served, each one with a corresponding demand, and a set of facilities that can be opened, each one with a given capacity. The *Capacitated Facility Location Problem* (CFLP) consists of selecting a subset of the facilities and of assigning customers to the open facilities. The aim is to minimize the sum of the opening and the assignment costs, without exceeding the capacity of the chosen facilities. A feasible solution to the problem is illustrated in Figure 1. We assume that all the customers have demand 1, that all the facilities have capacity 2.5 and that each customer can be assigned to any facility. The facilities are on the left:

the open ones correspond to triangles, while the others are represented by squares. The customers are on the right of the figure. The assignment of a customer to a facility is represented by a line, whose label is the amount supplied to the customer from the considered facility.

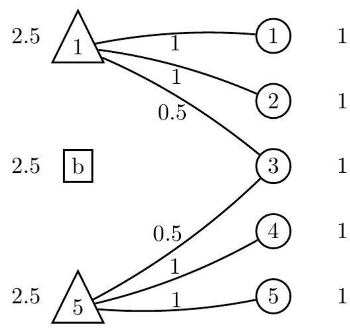

Figure 1: A feasible solution of the Capacitated Facility Location Problem.

The CFLP has received a considerable amount of attention in the literature and exact and heuristic approaches to solve the problem have been proposed. For comprehensive surveys on the CFLP and its applications to real world problems, we address the reader to Drezner and Hamacher (2002); Fernandez and Landete (2015); Klose and Drexl (2005a). The polyhedral structure of the CFLP and several families of valid inequalities have been investigated in Aardal et al. (1995); Leung and Magnanti (1989). Separation algorithms have been addressed in Aardal (1998a). In Avella and Boccia (2009) a reformulation based on *Dicut inequalities* has been proposed as a starting point for a cutting plane algorithm. In Görtz and Klose (2012); Klose and Drexl (2005b); Klose and Görtz (2007), a branch-and-bound method is presented, where the lower bound is computed using Lagrangian duality. The lower bound problem can either be solved to optimality by column generation (Klose and Drexl, 2005b; Klose and Görtz, 2007) or in a less precise way (Görtz and Klose, 2012) and then reconstructing an approximate primal solution to be used for the branching decisions. The current state-of-the-art exact algorithm for solving the CFLP is the one proposed in Fischetti et al. (2016) and is based on a Benders reformulation of the problem.

In this paper we present a new class of valid inequalities for the CFLP polytope, the *Weak Flow Cover* (WFC) inequalities, defining a relaxation of the *Flow Cover* (FC) inequalities introduced in Aardal (1998a). Here we show that, unlike the FC inequalities, for which a polynomial separation algorithm is known only for the special case of constant capacities, WFC inequalities can be separated in polynomial time via a min-cut algorithm, even in case of varying capacities.

Since any violated Weak Flow Cover can be easily transformed into a violated Flow Cover, this class of inequalities naturally leads to a heuristic approach for separating violated Flow Cover inequalities. Computational tests on benchmark instances with 2000 facilities and 2000 customers show that the approach based on the WFC inequalities is helpful in tightening the formulation. To further validate their effectiveness we embed them into an enumeration scheme, which is able to improve the best known results reported in the literature for several large test instances.

The remainder of the paper is organized as follows. In §2 we present the formulation of the CFLP and review known valid inequalities. In §3 we introduce the WFC inequalities, their separation algorithm and show how to use them within a separation heuristic for the Flow Cover inequalities. In §4 we report and discuss the results obtained using cut-and-branch approach based on the use of WFC inequalities.

2. Problem formulation and known valid inequalities

2.1. Mathematical formulation

Let $I = \{1, \dots, m\}$ and $J = \{1, \dots, n\}$ be a set of facilities and a set of customers, respectively. Let C_i be the capacity of facility $i \in I$ and let D_j be the demand of customer $j \in J$. Let F_i be the fixed cost of opening facility $i \in I$ and let Q_{ij} be the cost of serving a unit of demand of the customer $j \in J$ from facility $i \in I$. The CFLP selects a subset of facilities $O \subseteq I$ to open, such that the total capacity of the facilities in O can accommodate the demands of the customers in J, minimizing the sum of the fixed and the service costs. Let y_i be a binary variable associated with facility $i \in I$, that is, $y_i = 1$ if i is selected to be opened ($i \in O$), and 0 otherwise. Let f_{ij} be the amount of demand shipped from facility i to customer j. The CFLP can be formulated as below.

$$\text{M(CFLP)} \quad \min \sum_{i \in I} F_i y_i + \sum_{k \in I} \sum_{j \in J} Q_{ij} f_{ij}$$

$$\sum_{i \in I} f_{ij} = D_j \qquad j \in J \tag{1}$$

$$\sum_{j \in J} f_{ij} \leq C_i y_i \qquad i \in I \tag{2}$$

$$y_i \in \{0, 1\} \qquad i \in I$$

$$f_{ij} \geq 0 \qquad i \in I, j \in j$$

Constraints (1) require that the whole demand of each customer is satisfied. Capacity constraints (2) impose that no customer is supplied from a closed facility and that the total demand supplied from facility $i \in I$ does not exceed its capacity C_i. Moreover, let conv(CFLP) be the convex hull of the points $(\mathbf{y}, \mathbf{f}) \in \text{M(CFLP)}$.

2.2. Known valid inequalities

It is well-known that the formulation above can be strengthened using the inequalities.

$$\sum_{i \in I} C_i y_i \geq \sum_{j \in J} D_j \tag{3}$$

$$f_{ij} \leq D_j y_i \qquad i \in I, j \in J \tag{4}$$

The aggregated capacity constraint (3) is redundant, but induces MIP solvers to derive knapsack cuts, which can effectively tighten the formulation (Aardal, 1998b). Constraints (4) are known as variable upper bounds. ¿From now on, we refer to formulation M(CFLP) plus constraints (3) and (4) as $\hat{\text{M}}$(CFLP).

Several families of valid inequalities have been introduced in the CFLP literature. In Leung and Magnanti (1989), the CFLP polytope in the case of constant capacities is studied and the *Residual Capacity* (RC) inequalities are introduced. RC inequalities are obtained by aggregating subsets of the capacity constraints (2) and then applying a Mixed-Integer Rounding (MIR) procedure. Aardal et al. (1995) introduce the *Submodular Inequalities*, which have the *Flow Cover*, the *Effective Capacity* and the *Single Depot* inequalites as special cases. They also presented the *Combinatorial* inequalities, already introduced by Cho et al. (1983) for the uncapacitated problem, and the *Lot-sizing* inequalities, that are valid in the case of constant capacities. The companion paper Aardal (1998a) presents heuristic separation routines for the Effective Capacity, the Single Depot and the Combinatorial inequalities.

The Flow Cover (FC) inequalities can be defined as follows. Let $S \subseteq I$ and $K \subseteq J$ be a subset of facilities and of customers, respectively, such that $\sum_{i \in S} C_i - \sum_{j \in K} D_j = \lambda > 0$, i.e., S is a *cover* for the demands of the customers in K. Then, the FC inequality associated with (S, K) is the inequality below.

$$\text{FC}(S,K): \quad \sum_{i \in S} (C_i - \lambda)^+ (1 - y_i) + \sum_{i \in S} \sum_{j \in K} f_{ij} \leq \sum_{j \in K} D_j \tag{5}$$

The separation problem for the FC inequalities is polynomial in case of constant capacities, i.e., $C_i = C$ for all $i \in I$, if K is given (Aardal, 1998a). No polynomial separation algorithms are known for the general case with varying capacities and K not preassigned. More generally, modern MIP solvers generate MIR-Flow Cover cutting planes, i.e., derive FC inequalities by detecting single-node flow substructures and then applying a MIR procedure over them, see Wolter (2006); Louveaux and Wolsey (2003) for the details.

3. Weak Flow Cover inequalities

Here we introduce a new family of inequalities, the *Weak Flow Cover* (WFC) inequalities. First, we show that the WFC inequalities are valid for conv(CFLP) but dominated by the FC inequalities. Then, we present a polynomial time separation algorithm, based on the solution of a min-cut problem.

Let $S \subseteq I$ and $K \subseteq J$ be subsets of facilities and of customers, respectively, and let $h \in S$ be a selected facility. Let $\lambda = \sum_{i \in S} C_i - \sum_{j \in K} D_j$.
The WFC inequality associated with (S, K, h) is

$$\text{WFC}(S,K,h): \qquad \sum_{i \in S} \sum_{j \in K} f_{ij} \leq (C_h - \lambda) y_h + \sum_{i \in S \setminus \{h\}} C_i \tag{6}$$

3.1. Validity and relation with the FC inequalities

The following theorem shows that the WFC inequalities are valid for conv(CFLP).

Theorem 1. *WFC inequalities (6) are valid for* conv(CFLP)*.*

Proof. Consider $y_h = 0$. Trivially, $\sum_{i\in S\setminus\{h\}} \sum_{j\in K} f_{ij} \leq \sum_{i\in S\setminus\{h\}} C_i$, is valid for conv(CFLP). Now, consider $y_h = 1$. Inequality (6) becomes

$$\sum_{i\in S} \sum_{j\in K} f_{ij} \leq \sum_{i\in S} C_i - \lambda$$

and recalling that $\lambda = \sum_{i\in S} C_i - \sum_{j\in K} D_j$, we get

$$\sum_{i\in S} \sum_{j\in K} f_{ij} \leq \sum_{j\in K} D_j,$$

which is also valid for the CFLP. □

However, WFC inequalities are dominated by the FC inequalities.

Theorem 2. *WFC inequalities (6) are dominated by FC inequalities (5)*

Proof. Since the FC inequality

$$\sum_{i\in S}(C_i - \lambda)^+(1 - y_i) + \sum_{i\in S} \sum_{j\in K} f_{ij} \leq \sum_{j\in K} D_j$$

is a less-then-or-equal-to inequality and each contribution $(C_i - \lambda)^+(1 - y_i)$ is non-negative, we can weaken it by removing $(C_i - \lambda)^+(1 - y_i)$ for each $i \in S \setminus \{h\}$, obtaining

$$(C_h - \lambda)^+(1 - y_h) + \sum_{i\in S} \sum_{j\in K} f_{ij} \leq \sum_{j\in K} D_j$$

which is trivially still valid. Again, since we have a less-than-or-equal-to inequality, if we replace $(C_h - \lambda)^+$ by $(C_h - \lambda)$ we are further weakening it, without loosing the validity. In fact, we are just possibly transforming a zero coefficient into a negative one, in the left-hand-side of a less-then-or-equal-to inequality. We obtain

$$(C_h - \lambda)(1 - y_h) + \sum_{i\in S} \sum_{j\in K} f_{ij} \leq \sum_{j\in K} D_j$$

which is equivalent to

$$\sum_{i\in S} \sum_{j\in K} f_{ij} \leq (C_h - \lambda)y_h - C_h + \lambda + \sum_{j\in K} D_j$$

Since $\lambda = \sum_{i\in S} C_i - \sum_{j\in K} D_j$, the result is the Weak Flow Cover Inequality (6).

$$\sum_{i\in S}\sum_{j\in K} f_{ij} \leq (C_h - \lambda)y_h + \sum_{i\in S\setminus\{h\}} C_i$$

□

Now, we demonstrate that the WFC inequalities can be separated in polynomial time in the general case, whereas the FC inequalities can be polynomially separated only for constant capacities and K given. Moreover, we show how to use the exact separation of the WFC inequalities to derive a heuristic separation procedure for the Flow Cover inequalities.

3.2. Separation algorithm for WFC

To describe the separation algorithm we first rewrite the WFC inequality in a convenient form. For a given (S, K, h), consider the associated WFC inequality

$$\sum_{i\in S}\sum_{j\in K} f_{ij} \leq (C_h - \lambda)y_h + \sum_{i\in S\setminus\{h\}} C_i$$

Replacing λ by its definition, that is, $\lambda = \sum_{i\in S} C_i - \sum_{j\in K} D_j$, first we get

$$\sum_{i\in S}\sum_{j\in K} f_{ij} \leq (C_h - \sum_{i\in S} C_i + \sum_{j\in K} D_j)y_h + \sum_{i\in S\setminus\{h\}} C_i$$

then

$$\sum_{i\in S}\sum_{j\in K} f_{ij} \leq -(\sum_{i\in S\setminus\{h\}} C_i - \sum_{j\in K} D_j)y_h + \sum_{i\in S\setminus\{h\}} C_i \tag{7}$$

Constraint (1) of $\hat{\mathrm{M}}$(CFLP) implies that $\sum_{i\in I} f_{ij} = D_j$ for any $j \in J$. Since I is partitioned into $\{S : I \setminus S\}$ one can write $D_j = \sum_{i\in S} f_{ij} + \sum_{i\in I\setminus S} f_{ij}$. This holds for any $j \in J$ and also for $j \in K \subseteq J$. Thus, by summing up the equations for all $j \in K$, we get:

$$\sum_{j\in K} D_j = \sum_{j\in K}\sum_{i\in S} f_{ij} + \sum_{j\in K}\sum_{i\in I\setminus S} f_{ij}$$

that is

$$\sum_{i\in S}\sum_{j\in K} f_{ij} = \sum_{j\in K} D_j - \sum_{i\in I\setminus S}\sum_{j\in K} f_{ij}$$

The above equality substituted into inequality (7) returns

$$\sum_{j\in K} D_j - \sum_{i\in I\setminus S}\sum_{j\in K} f_{ij} \leq -(\sum_{i\in S\setminus\{h\}} C_i - \sum_{j\in K} D_j)y_h + \sum_{i\in S\setminus\{h\}} C_i$$

which corresponds to

$$(\sum_{i\in S\setminus\{h\}} C_i - \sum_{j\in K} D_j)(1-y_h) + \sum_{i\in I\setminus S}\sum_{j\in K} f_{ij} \geq 0 \tag{8}$$

Given the optimal solution $(\bar{\mathbf{y}}, \bar{\mathbf{f}})$ to the linear relaxation of $\hat{\mathrm{M}}$(CFLP), to formalize the separation problem we represent the (unknown) sets S and K by, respectively, binary variables u_i (for $i \in I$) and v_j (for $j \in J$). Then, $u_i = 1$ if $i \in S$ (0 otherwise) and $v_j = 1$ if $j \in K$ (0 otherwise). Let w_{ij} be a binary variable associated with the pair (i, j) for $i \in I$, $j \in J$ and such that $w_{ij} = 1$ if $i \in I \setminus S$ and $j \in K$ and 0 otherwise. Now, if we select a facility $h \in I$, the LP solution $(\bar{\mathbf{y}}, \bar{\mathbf{f}})$ violates the WFC inequality (S, K, h) (8) if the following problem has a negative optimal solution.

$$\begin{aligned}
\mathrm{SepWFC}(\bar{\mathbf{y}}, \bar{\mathbf{f}}, h) \quad & \min \sum_{i\in I\setminus\{h\}}\sum_{j\in J} \bar{f}_{ij} w_{ij} + (1-\bar{y}_h)\Big(\sum_{i\in I\setminus\{h\}} C_i u_i - \sum_{j\in J} D_j v_j\Big) \\
& w_{ij} \geq v_j - u_i, \qquad i \in I \setminus \{h\}, j \in J \\
& u_i \in \{0,1\}, \qquad i \in I \setminus \{h\} \\
& v_j \in \{0,1\}, \qquad j \in J \\
& w_{ij} \in \{0,1\}, \qquad i \in I \setminus \{h\}, j \in J
\end{aligned}$$

The problem above, has a simple interpretation as a min-cut problem on a suitable graph.

Let $G(N, A, h)$ be the directed graph with node set $N = \big\{(I\setminus\{h\}) \cup J \cup \{s, t\}\big\}$ and arc set $A = \{(s, i) : i \in I\setminus\{h\}\} \cup \{(j, t) : j \in J\} \cup \{(i, j) : i \in I\setminus\{h\}, j \in J\}$. Given a solution $(\bar{\mathbf{y}}, \bar{\mathbf{f}})$, arc weights are defined as follows: arcs (s, i) have weights $(1-\bar{y}_h)C_i$; arcs (j, t) have weights $(1-\bar{y}_h)D_j$; arcs (i, j) have weights $\bar{f}_{ij}$.

Let $B \subseteq A$, we denote by $\delta^+(B)$ $(\delta^-(B))$ the outgoing (incoming) arcs with respect of B.

Theorem 3. $\mathrm{SepWFC}(\bar{\mathbf{y}}, \bar{\mathbf{f}}, h)$ *can be solved as a minimum capacity* (s, t)*-cut problem on* $G(N, A, h)$.

Proof. If we add auxiliary variables $w_{si} = u_i$ for $i \in I \setminus \{h\}$ and $w_{jt} = 1 - v_j$ for $j \in J$, one can observe that $\mathrm{SepWFC}(\bar{\mathbf{y}}, \bar{\mathbf{f}}, h)$ admits the following relaxation.

$$\min \sum_{i\in I\setminus\{h\}}\sum_{j\in J}\bar{f}_{ij}w_{ij}+(1-\bar{y}_h)(\sum_{i\in I\setminus\{h\}}C_i w_{si}+\sum_{j\in J}D_j w_{jt})-(1-\bar{y}_h)\sum_{j\in J}D_j \tag{9}$$

$$\begin{aligned}
&w_{si}\geq u_i, && i\in I\setminus\{h\}\\
&w_{jt}\geq 1-v_j && j\in J\\
&w_{ij}\geq v_j-u_i && i\in I\setminus\{h\}, j\in J\\
&w_{ij}\geq 0, && i\in\{s,t\}\cup I\setminus\{h\}, j\in J
\end{aligned}$$

The constant term $(1-\bar{y}_h)\sum_{j\in J}D_j$ can be removed from the objective function. Linear program (9) is the formulation of a minimum (s,t)-cut problem on $G=(N,A,h)$, where the weights are assigned according to $(\bar{\mathbf{y}},\bar{\mathbf{f}})$. The optimal solution to the problem above defines an (s,t)-cut $[R,\bar{R}]$ in $G(N,A,h)$. Precisely, variables $u_i=0$ and $v_j=0$ identify the set R while variables $u_i=1$ and $v_j=1$ identify the set $\bar{R}$. Variables w_{si}, w_{jt} and w_{ij} correspond to arcs (s,i), (j,t) and $(i,j)\in A$ taking value one if the corresponding arc is in $\delta^+(R)$ and zero otherwise. Therefore, problem (9) admits an integer optimal solution and, being a relaxation of SepWFC$(\bar{\mathbf{y}},\bar{\mathbf{f}},h)$, this solution is also feasible and optimal for the separation problem. The minimum capacity (s,t)-cut problem provides a violated WFC inequality if its value is less than $\sum_{j\in J}D_j$ with the set S given by $I\cap\bar{R}\cup\{h\}$ and the set K given by $J\cap\bar{R}$. □

In Figure 2 is depicted an (s,t)-cut $[R,\bar{R}]$ (dark gray nodes are in R, white nodes are in $\bar{R}$) and the corresponding sets S and K.

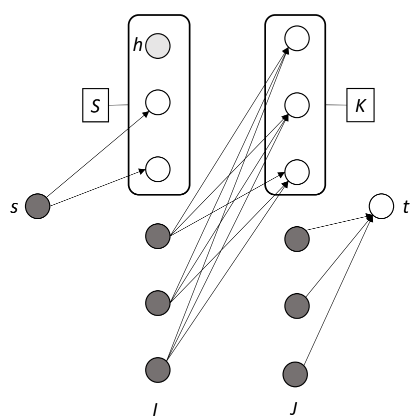

Figure 2: A cut in G

3.3. From Weak Flow Cover to Flow Cover

Any violated Weak Flow Cover for (S, K, h) naturally gives rise to a Flow Cover inequality defined on the sets S and K, which is violated as well. Moreover, the violation is preserved.

Theorem 4. *If a WFC inequality for (S, K, h) is violated by $(\bar{\mathbf{y}}, \bar{\mathbf{f}})$, then the corresponding $FC(S, K)$ inequality*

$$\sum_{i \in S} (C_i - \lambda)^+ (1 - y_i) + \sum_{i \in S} \sum_{j \in K} f_{ij} \leq \sum_{j \in K} D_j$$

is violated by at least the same amount of the WFC.

Proof. Consider the WFC for (S, K, h) and assume that it is violated by $(\bar{\mathbf{y}}, \bar{\mathbf{f}})$, that is

$$\sum_{i \in S} \sum_{j \in K} \bar{f}_{ij} > (C_h - \lambda)\bar{y}_h + \sum_{i \in S \setminus \{h\}} C_i$$

If we subtract a non-negative amount from the right-hand-side, the violation does not decrease. Then, as $(C_i - \lambda)^+(1 - \bar{y}_i)$ is non-negative for any i, we can subtract the value $\sum_{i \in S \setminus \{h\}} (C_i - \lambda)^+ (1 - \bar{y}_i)$ from the right-hand-side. We get

$$\sum_{i \in S} \sum_{j \in K} \bar{f}_{ij} > (C_h - \lambda)\bar{y}_h + \sum_{i \in S \setminus \{h\}} C_i - \sum_{i \in S \setminus \{h\}} (C_i - \lambda)^+ (1 - \bar{y}_i)$$

Recalling that $\lambda = \sum_{i \in S} C_i - \sum_{j \in K} D_j$ and then $\sum_{i \in S \setminus \{h\}} C_i = \lambda + \sum_{j \in K} D_j - C_h$, we get

$$\sum_{i \in S} \sum_{j \in K} \bar{f}_{ij} > \sum_{j \in K} D_j - (C_h - \lambda)(1 - \bar{y}_h) - \sum_{i \in S \setminus \{h\}} (C_i - \lambda)^+ (1 - \bar{y}_i)$$

Again, if we replace amount $(C_h - \lambda)$ by the non-negative amount $(C_h - \lambda)^+$, the violation does not decrease. Since the above transformations do not decrease the violation, the resulting Flow Cover Inequality $\text{FC}(S, K)$

$$\sum_{i \in S} \sum_{j \in K} \bar{f}_{ij} > \sum_{j \in K} D_j - \sum_{i \in S} (C_i - \lambda)^+ (1 - \bar{y}_i)$$

is at least as violated as the original WFC inequality. □

Since the separation problem for the Weak Flow Cover inequalities depends on h, we must iterate over $h \in I$ to see if there is a h corresponding to a violated inequality to be transformed into a Flow Cover. The overall procedure is illustrated in Algorithm 1, where we consider all $h \in I$ such that $\bar{y}$ is fractional.

Algorithm 1: FC separation

```
Data: (ȳ, f̄)
Result: A (possibly empty) collection of Flow Cover inequalities 𝓕
begin
    Y = {i ∈ I : 0 < ȳ_i < 1};
    𝓕 = ∅;
    while Y ≠ ∅ do
        Select h ∈ Y;
        Build G = (N, A, h);
        Find the minimum capacity {s, t}-cut on G and let S and K
          be the corresponding sets of nodes;
        if the inequality FC(S, K) (5) is violated then
            𝓕 := 𝓕 ∪ FC(S, K);
        end
        Y = Y \ {h};
    end
end
return 𝓕
```

4. The computational study

To validate the effectiveness of the procedure based on the Weak Flow Cover inequalities, we embedded the separation Algorithm 1 into the cut-and-branch framework provided by Xpress-Mosel 5.0.2 (Heipcke, 2012). All the experiments were performed through the Xpress-Mosel interface and ran on an Intel Xeon Silver 4110 2.1 GHz 8 Core workstation with 96 GB of RAM (Windows Server 2016 as operating system). Xpress memory size limit has been set to 32GB to mimic performance of less powerful workstations.

4.1. Test-bed description

The most challenging benchmark instances in the recent literature are the $i*$ instances with 1000 facilities and 1000 customers, tested in Fischetti et al. (2016), and the $p*$ instances with 2000 facilities and 2000 customers of the test-bed C, introduced in Guastaroba and Speranza (2012) and also used by Fischetti et al. (2016). Nevertheless, because of the fast progress of MIP solvers, the 1000×1000 $i*$ instances have now become easy, being all solvable to optimality by FICO Xpress 8.7 in less than 1800 seconds on a workstation. Therefore, we restricted our attention to the subgroup of the 2000×2000 $p*$ instances and, in addition, we randomly generated a new set of $i*$ instances with 2000 facilities and 2000 customers, with the same properties of those introduced in Cornuéjols et al. (1991) and tested in several papers (Avella and Boccia, 2009; Fischetti et al., 2016). The 2000×2000 $p*$ instances of the test bed C can be divided into six groups of five instances, each including instances with the same ratio $r = \frac{\sum_{j\in J} D_j}{\sum_{i\in I} C_i}$, namely $r \in \{1.1, 1.5, 2, 3, 5, 10\}$. The $i*$ instances are organized into four groups of five instances with the same ratio r, namely $r \in \{5, 10, 15, 20\}$. The new i^* instances are downloadable at `http://www.ing.unisannio.it/boccia/CFLP.htm`.

4.2. Root node analysis

We first compare three strengthenings of the formulation $\hat{\mathrm{M}}$(CFLP):

XP	$\rightarrow$	$\hat{\mathrm{M}}$(CFLP) strengthened by Xpress cuts
WFC	$\rightarrow$	$\hat{\mathrm{M}}$(CFLP) strengthened only by WFC
XP+WFC	$\rightarrow$	$\hat{\mathrm{M}}$(CFLP) strengthened by Xpress cuts and then WFC

In all configurations that use Xpress cuts the parameter `CUTSTRATEGY` has been set to 3 (aggressive cut generation).
Tables A.6 and A.7 in Appendix A report the best LB obtained by each model, the percentage gap closed and the CPU time (with a time limit of 3600 sec). The percentage gap closed for a model $\mathcal{M} \in \{\mathbf{XP}, \mathbf{WFC}, \mathbf{XP} + \mathbf{WFC}\}$ is calculated as:

$$\frac{\bar{z}_{\mathcal{M}} - \bar{z}}{z^{\mathrm{UB}} - \bar{z}} \times 100$$

where $\bar{z}_{\mathcal{M}}$ is the value of the linear relaxation of the model $\mathcal{M}$, $\bar{z}$ is the value of the linear relaxation of $\hat{\mathrm{M}}$(CFLP) and z^{UB} is the best upper bound known for each instance.

Hence, results are summarized in Tables 1 and 2, reporting the percentage gap closed, the (average) number of separated WFC-s and the CPU time computed as averages over all instances having the same ratio r. Such values are also graphically illustrated in Figures 3 and 4.

	i^* 2000 × 2000							
	XP		**WFC**			**XP+WFC**		
Ratio	% gap closed	Time (sec)	% gap closed	# sep. WFC-s	Time (sec)	% gap closed	# sep. WFC-s	Time (sec)
r=5	21.04	1441	48.69	482.2	2668	49.05	351.0	3600
r=10	37.31	893	59.17	225.0	1084	64.13	177.8	1383
r=15	52.94	473	59.97	78.4	550	73.87	48.2	628
r=20	63.51	322	61.48	77.8	416	72.21	41.6	464

Table 1: % of gap closed at root node by **XP, WFC and XP+WFC**

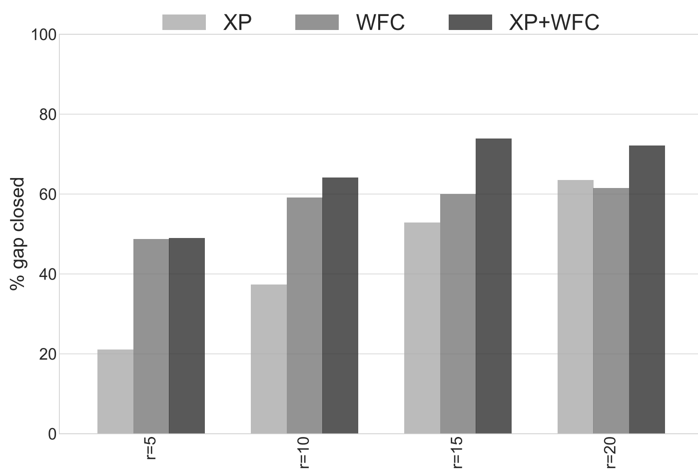

Figure 3: Graphical illustration of Table 1

For the i^* instances with $r \in \{5, 10, 15\}$, the WFC lower bounds are significantly better than those provided by the Xpress built-in cuts, despite Xpress cuts include, among others, general flow cover inequalities. Moreover, WFC inequalities used in combination with Xpress cuts close a remarkable portion of the gap for $r \in \{15, 20\}$ and this improvement has an acceptable computational cost.

As the p^* instances are concerned, one can observe that WFC inequalities outperform Xpress cuts only for $r \in \{1.1, 1.5\}$. However, the combination of Xpress cuts with WFC inequalities once again closes a relevant portion of the gap for instances with $r \in \{1.1, 1.5, 2, 3\}$ and almost all the gap for $r \in \{5, 10\}$ with a moderate additional computational burden.

In summary, the combination of Xpress cuts with WFC inequalities appears to be the most promising model for an enumeration algorithm.

p^* 2000 × 2000								
	XP		**WFC**			**XP+WFC**		
Ratio	% gap closed	Time (sec)	% gap closed	# sep. WFC-s	Time (sec)	% gap closed	# sep. WFC-s	Time (sec)
r=1.1	69.95	390	84.85	389.8	927	89.97	195.6	836
r=1.5	78.44	298	83.67	607.0	1097	89.73	297.2	871
r=2	87.49	241	86.26	501.4	917	94.54	184.0	663
r=3	94.80	175	82.36	213.4	405	97.27	29.8	315
r=5	98.46	113	84.10	104.8	288	99.08	4.4	170
r=10	99.79	88	74.07	32.2	199	99.83	0.2	96

Table 2: % of gap closed at root node by **XP, WFC and XP+WFC**

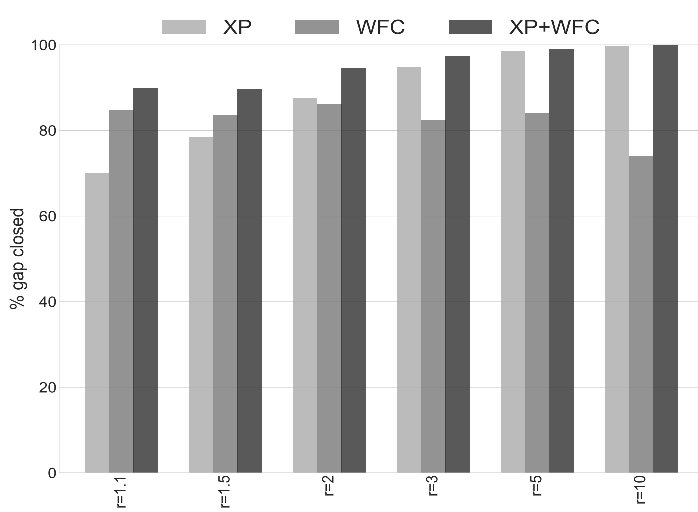

Figure 4: Graphical illustration of Table 2

4.3. Enumeration results

In this section we compare the performance of Xpress fed by the model **XP+WFC** (**XP+WFC**-enum) with the performance of Xpress itself (**XP**-enum). All solver parameters are left to default but `CUTSTRATEGY`=3. As before, we discuss results aggregated by r that are reported in Tables 3 and 4. Detailed results can be found in Appendix A (Table A.8 for the i^* instances and Table A.9 for the $p*$ instances).

i^* 2000 × 2000

Ratio	**XP**-enum % gap left	**XP**-enum # solved instances	**XP**-enum Time (sec)	**XP+WFC**-enum % gap left	**XP+WFC**-enum # solved instances	**XP+WFC**-enum Time (sec)
r=5	0.47	0/5	3600	0.36	0/5	3600
r=10	0.20	0/5	3600	0.09	1/5	3374
r=15	0.00	4/5	2562	0.00	5/5	933
r=20	0.01	4/5	1948	0.00	5/5	809

Table 3: Enumeration aggregated results on i^* instances

p^* 2000 × 2000

Ratio	**XP**-enum % gap left	**XP**-enum # solved instances	**XP**-enum Time (sec)	**XP+WFC**-enum % gap left	**XP+WFC**-enum # solved instances	**XP+WFC**-enum Time (sec)
r=1.1	0.05	0/5	3600	0.01	0/5	3600
r=1.5	0.07	0/5	3600	0.04	0/5	3600
r=2	0.04	0/5	3600	0.02	0/5	3600
r=3	0.00	2/5	2423	0.00	5/5	767
r=5	0.00	5/5	252	0.00	5/5	321
r=10	0.00	5/5	216	0.00	5/5	96

Table 4: Enumeration aggregated results on p^* instances

For each model, the tables report the percentage gap left at the end of computation (time limit is set to 3600 seconds) evaluated as $\frac{\text{(Best UB - Best LB)}}{\text{Best UB}} \times 100$, the number of solved instances within the time limit and the overall computing time (that is, the time to generate the model plus the enumeration).
On i^* instances **XP+WFC**-enum outperforms **XP**-enum both in effectiveness and CPU time. Namely, for $r = 5$ the average gap left by **XP+WFC**-enum at the time limit is 22% better than the gap left by **XP**-enum. For $r = 10$ there is an average improvement of about 50% and one instance out of 5 is also solved to optimality. Finally, for $r = 15$ and $r = 20$ **XP+WFC**-enum is able to solve all instances in less than half the time of **XP**-enum that fails on 2 instances out of 10. An in depth analysis of Table A.8 also shows that **XP+WFC**-enum either improves or ties the best lower bound of

XP-enum on all instances. Concerning the best upper bound, this happens on 16 instances out of 20.

On $p*$ instances the comparison between **XP**-enum and **XP+WFC**-enum leads to similar conclusions: when algorithms reach the time limit the gap left by **XP+WFC**-enum is about 50% of the gap left by **XP**-enum. **XP+WFC**-enum solves three more instances than **XP**-enum and it is typically faster than **XP**-enum with the exception of the instances with $r = 5$. The detailed analysis shows that **XP+WFC**-enum improves or ties both the best upper and lower bounds w.r.t. the **XP**-enum on 19 instances out of 20.

4.4. Comparison with other approaches in the literature

Now we compare **XP+WFC**-enum with the existing approaches in the literature. As discussed in 4.1, the comparison is limited to $p*$ instances. In Figure 5 for each group of the $p*$ instances with the same ratio r, we report the average final gaps returned by **XP+WFC**-enum and the one evaluated by Fischetti et al. (2016) through a Benders decomposition approach (denoted as **Benders** in the figure). This comparison was made possible by M. Fischetti and M. Sinnl that kindly provided us their detailed results (Fischetti and Sinnl (2018)). Their experiments were conducted on a cluster of identical machines each consisting of an Intel Xeon E3-1220V2 @3.1 GHz with 16 GB of RAM and setting a time limit of 50000 seconds.

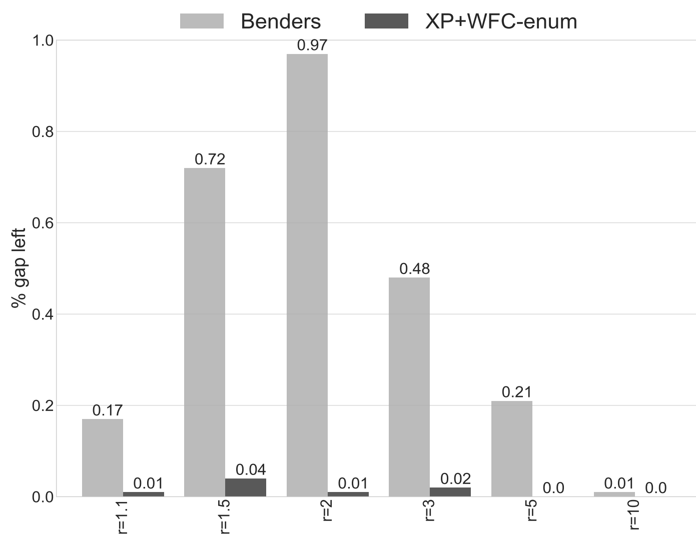

Figure 5: **XP+WFC**-enum vs. Benders reformulation approach of Fischetti et al. (2016): average gaps on $p*$ instances

The chart shows that the final gaps yielded by **XP+WFC**-enum are substiantially better than those obtained by the Benders reformulation, especially for instances with $r \in \{1.5, 2, 3\}$.

In Table 5, for each group of $p*$ instances with the same ratio r, we compare the upper bounds returned by **XP+WFC**-enum with those obtained by the Kernel Search heuristic of Guastaroba and Speranza (2012) (denoted as **Kernel Search**). For a fair comparison, the lower bound used for the percentage gap evaluation is the one reported in Table 15 of Guastaroba and Speranza (2012) that is significantly smaller than the **XP+WFC**-enum lower bound. Results in Guastaroba and Speranza (2012) have been obtained on a PC Intel Xeon @ 2.27 GHz 64-bit processor with 12.0 GB of RAM, Windows 7 64-bit and CPLEX 12.2, setting a time limit of 5000 seconds.

p^* 2000 × 2000				
	XP + WFC-enum		**Kernel Search**	
Ratio	% gap left	Time (sec)	% gap left	Time (sec)
r=1.1	0.34	3600	0.36	2045
r=1.5	2.08	3600	2.24	2031
r=2	4.33	3600	4.57	2021
r=3	4.81	767	4.96	2016
r=5	6.93	532	7.03	2008
r=10	8.72	96	8.72	875

Table 5: **XP+WFC**-enum vs. Kernel Search

Upper bounds provided by **XP+WFC**-enum at the time limit for instances with $r \in \{1, 1.5, 2\}$ are slightly better than those provided by Kernel Search in a comparable time, while for $r \in \{3, 5, 10\}$ **XP+WFC**-enum actually solves to optimality all the instances outperforming Kernel Search.

5. Conclusions

In this paper we considered the Capacitated Facility Location Problem, whose goal is to open a set of facilities, with the aim of satisfying at the minimum cost the demands of a set of customers, respecting the capacities of the facilities. We introduced the Weak Flow Cover inequalities, a new family of valid inequalities for the problem. We showed that they are a weakening of the well-known Flow Cover inequalities and hence valid for the problem. We proposed a polynomial time separation algorithm to find violated Weak Flow Cover inequalities that is based on a minimum cut problem on a suitable graph. Then, we discussed how to use the exact separation of Weak Flow Cover inequalities as a heuristic separation for Flow Cover inequalities. Finally, we embedded the separation procedure into a cut-and-branch scheme that is able to produce results that are significantly better than those reported in the recent literature.

Acknowledgments

We would like to thank the anonymous reviewers for the insightful comments that helped improving the quality of the paper.

Authors are partially supported by MIUR PRIN grant n.20153TXRX9. S. Mattia is also partially supported by MIUR PRIN grant n.2015B5F27W.

References

Aardal, K., 1998a. Capacitated facility location: separation algorithms and computational experience. Mathematical Programming 81, 149–175.

Aardal, K., 1998b. Reformulation of capacitated facility location problems: How redundant information can help. Annals of Operations Research 82, 289–308.

Aardal, K., Pochet, Y., Wolsey, L., 1995. Capacitated facility location: valid inequalities and facets. Mathematics of Operations Research 20, 562–582.

Avella, P., Boccia, M., 2009. A cutting plane algorithm for the capacitated facility location problem. Computational Optimization and Applications 43, 39–65.

Cho, D., Johnson, E., Padberg, M., Rao, M., 1983. On the uncapacitated plant location problem i: Valid inequalities and facets. Mathematics of Operations Research 8, 579–589.

Cornuéjols, G., Sridharan, R., Thizy, J.M., 1991. A comparison of heuristics and relaxations for the capacitated plant location problem. European journal of operational research 50, 280–297.

Drezner, Z., Hamacher, H. (Eds.), 2002. Facility location: applications and theory. Springer, New York.

Fernandez, E., Landete, M., 2015. Fixed-charge facility location problems, in: Laporte, G., Nickel, S., da Gama, F. (Eds.), Location science. Springer, pp. 47–77.

Fischetti, M., Ljubić, I., Sinnl, M., 2016. Benders decomposition without separability: A computational study for capacitated facility location problems. European Journal of Operational Research 253, 557–569.

Fischetti, M., Sinnl, M., 2018. Personal Communication.

Görtz, S., Klose, A., 2012. A simple but usually fast branch-and-bound algorithm for the capacitated facility location problem. INFORMS Journal on Computing 24, 597–610.

Guastaroba, G., Speranza, M.G., 2012. Kernel search for the capacitated facility location problem. Journal of Heuristics 18, 877–917.

Heipcke, S., 2012. Xpress-mosel: Multi-solver, multi-problem, multi-model, multi-node modeling and problem solving. Algebraic Modeling Systems: Modeling and Solving Real World Optimization Problems , 77–110.

Klose, A., Drexl, A., 2005a. Facility location models for distribution system design. European Journal of Operational Research 162, 4–29.

Klose, A., Drexl, A., 2005b. Lower bounds for the capacitated facility location problem based on column generation. Management Science 5, 1689–1705.

Klose, A., Görtz, S., 2007. A branch-and-price algorithm for the capacitated facility location problem. European Journal of Operational Research 179, 1109–1125.

Leung, J., Magnanti, T., 1989. Valid inequalities and facets of the capacitated plant location problem. Mathematical Programming 44, 271–291.

Louveaux, Q., Wolsey, L., 2003. Lifting, superadditivity, mixed integer rounding and single node flow sets revisited. Quarterly Journal of the Belgian, French and Italian Operations Research Societies 1, 173–207.

Wolter, K., 2006. Implementation of cutting plane separators for mixed integer programs. Master's thesis, Technische Universität Berlin .

Appendix A. Detailed computational results

Key to Tables A.6 and A.7	
Ratio	Instance ratio
Name	Instance name
$\hat{\text{M}}$(CFLP) LP value	Value of the linear relaxation of the model (1)-(4)
LB	Value of the linear relaxation of the strenghtened model {**XP**,**WFC**, **XP+WFC**}
%gap closed	Percentage of the gap closed by the strenghtened model {**XP**,**WFC**, **XP+WFC**}
Time	CPU time in seconds

Key to Tables A.8 and A.9	
Ratio	Instance ratio
Name	Instance name
$\hat{\text{M}}$(CFLP) LP value	Value of the linear relaxation of the model (1)-(4)
Best LB	Value of the best lower bound at the end of the computation for **XP**-enum and **XP+WFC**-enum
Best UB	Value of the best upper bound at the end of the computation for **XP**-enum and **XP+WFC**-enum
%gap	Percentage gap left at the end of the computation for **XP**-enum and **XP+WFC**-enum
Nodes	Number of enumerated subproblems for **XP**-enum and **XP+WFC**-enum
WFC-s	Number of added WFC inequalities for **XP+WFC**-enum
Time	CPU time in seconds

		$\hat{\text{M}}$(CFLP)	**XP**			**WFC**			**XP + WFC**		
		LP value	LB	%gap	Time	LB	%gap	Time	LB	%gap	Time
Ratio	Name			closed			closed			closed	
	i2000_1	195219.6	195537.2	21.73	2543	195912.1	47.37	3600	195768.8	37.57	3600
	i2000_2	200295.1	200648.7	25.09	1033	200923.3	44.57	2053	200973.1	48.10	3600
5	i2000_3	194796.5	195018.3	15.13	1074	195425.3	42.90	2840	195454.9	44.92	3600
	i2000_4	188503.7	188731.5	25.50	1491	189074.9	63.93	2924	189097.5	66.47	3600
	i2000_5	191448.2	191683.0	17.76	1066	192039.3	44.70	1921	192085.7	48.21	3600
	i2000_6	161125.3	161244.4	30.53	870	161295.0	43.50	915	161311.7	47.77	1169
	i2000_7	157918.6	158140.1	49.26	969	158232.8	69.87	1189	158277.5	79.81	1368
10	i2000_8	160681.1	160883.8	24.02	700	161134.1	53.68	909	161165.2	57.37	1201
	i2000_9	160138.9	160446.7	40.80	902	160650.5	67.81	1249	160676.4	71.24	1725
	i2000_10	156918.1	157130.3	41.92	1023	157226.8	60.97	1160	157244.4	64.45	1454
	i2000_11	151050.4	151179.5	50.44	561	151200.2	58.51	550	151249.3	77.72	735
	i2000_12	153112.3	153208.8	52.95	439	153214.9	56.27	534	153237.1	68.43	601
15	i2000_13	151543.2	151642.3	41.32	407	151680.2	57.16	472	151709.3	69.27	552
	i2000_14	151906.4	152057.0	57.94	525	152066.8	61.71	610	152096.1	72.98	684
	i2000_15	147603.5	147694.0	62.04	435	147700.0	66.18	585	147721.5	80.95	569
	i2000_16	150481.1	150604.1	47.54	343	150599.2	45.66	418	150617.3	52.65	500
	i2000_17	145530.3	145685.5	64.59	274	145697.1	69.41	380	145705.9	73.10	430
20	i2000_18	146394.0	146502.4	72.16	318	146495.5	67.53	396	146520.1	83.94	442
	i2000_19	147907.8	148082.7	69.02	328	148065.0	62.04	435	148108.6	79.26	467
	i2000_20	147814.8	147948.7	64.22	348	147945.7	62.76	451	147965.1	72.07	479

Table A.6: Lower bounds for the $i*$ 2000×2000 instances

Ratio	Name	$\hat{\text{M}}$(CFLP) LP value	**XP** LB	**XP** %gap closed	**XP** Time	**WFC** LB	**WFC** %gap closed	**WFC** Time	**XP + WFC** LB	**XP + WFC** %gap closed	**XP + WFC** Time
	p2000-2000-61	9406130.6	9415824.2	69.67	434	9417528.9	81.92	983	9418318.4	87.59	852
	p2000-2000-62	9553583.9	9561889.5	63.93	429	9564937.5	87.39	993	9565179.6	89.26	901
1.1	p2000-2000-63	9988755.1	10000942.9	76.15	370	10002145.7	83.66	835	10003422.1	91.64	821
	p2000-2000-64	9753128.3	9763636.8	66.93	327	9765915.6	81.44	905	9767026.9	88.52	832
	p2000-2000-65	9815658.2	9826482.6	73.07	391	9828965.3	89.83	921	9829414.2	92.86	775
	p2000-2000-66	7388283.6	7418690.5	79.44	286	7419886.5	82.57	891	7422984.6	90.66	803
	p2000-2000-67	7744753.6	7777237.3	75.80	299	7779541.0	81.18	1313	7781187.4	85.02	686
1.5	p2000-2000-68	7255942.0	7286942.6	79.83	303	7288821.2	84.67	1148	7291333.4	91.14	916
	p2000-2000-69	7383389.8	7416261.1	79.96	319	7418475.5	85.35	1008	7421045.0	91.60	1025
	p2000-2000-70	7256064.7	7287571.8	77.17	282	7290606.4	84.61	1123	7292911.8	90.25	926
	p2000-2000-71	6437955.1	6478601.8	88.71	243	6476155.7	83.37	880	6481698.7	95.46	627
	p2000-2000-72	6525168.6	6565376.1	88.25	187	6565037.9	87.51	971	6568293.2	94.66	598
2	p2000-2000-73	6507042.7	6551893.9	86.00	260	6552551.8	87.26	1004	6556548.6	94.92	697
	p2000-2000-74	6321174.3	6364473.5	83.36	301	6365287.0	84.93	958	6368678.5	91.46	854
	p2000-2000-75	6422548.6	6462508.4	91.13	214	6461232.9	88.22	771	6464738.2	96.21	540
	p2000-2000-76	5560615.4	5577957.6	94.01	334	5575851.5	82.60	421	5578388.0	96.35	478
	p2000-2000-77	5321478.8	5343290.3	95.88	164	5339957.6	81.23	406	5343789.9	98.08	302
3	p2000-2000-78	5247970.7	5266024.4	96.26	119	5263583.5	83.24	400	5266362.7	98.06	225
	p2000-2000-79	5341576.1	5362864.6	94.45	129	5359852.1	81.08	430	5363595.1	97.69	291
	p2000-2000-80	5250214.2	5271080.2	93.40	129	5268896.1	83.63	370	5271705.2	96.20	278
	p2000-2000-81	5015360.3	5027845.9	96.86	111	5025609.9	79.51	309	5027939.5	97.58	215
	p2000-2000-82	4975802.0	4987222.9	100.0	568	4985491.9	84.84	284	4987222.9	100.0	131
5	p2000-2000-83	4899363.7	4910528.4	99.17	110	4908878.7	84.51	275	4910561.4	99.46	171
	p2000-2000-84	5030533.5	5041073.8	96.91	121	5040098.1	87.94	294	5041385.8	99.78	183
	p2000-2000-85	5114442.9	5124908.4	99.38	110	5123254.2	83.67	278	5124930.2	99.59	149
	p2000-2000-86	4967188.3	4971500.5	100.0	75	4970145.0	68.57	194	4971500.5	100.0	75
	p2000-2000-87	4810796.1	4814108.5	100.0	77	4813378.8	77.97	208	4814108.5	100.0	77
10	p2000-2000-88	4982264.2	4986966.7	100.0	76	4986468.3	89.40	192	4986966.7	100.0	76
	p2000-2000-89	5004397.1	5007821.0	98.95	110	5006632.9	64.61	214	5007827.9	99.15	147
	p2000-2000-90	4944833.7	4947837.8	100.0	104	4946930.5	69.80	189	4947837.8	100.0	104

Table A.7: Lower bounds for the $p*$ 2000×2000 instances

		$\hat{M}$(CFLP)	**XP**-enum					**XP + WFC**-enum				
Ratio	Name	LP value	Best LB	Best UB	%gap	Nodes	Time	Best LB	Best UB	%gap	Nodes	Time
5	i2000_1	195219.6	195592.4	196439.7	0.43	0	3600	195941.8	196681.6	0.38	0	3600
	i2000_2	200295.1	200647.3	201569.7	0.46	0	3600	200944.6	201704.5	0.38	0	3600
	i2000_3	194796.5	195062.9	196241.2	0.60	0	3600	195243.2	196262.3	0.52	0	3600
	i2000_4	188503.7	188764.1	189465.7	0.37	0	3600	189101.4	189397.1	0.16	0	3600
	i2000_5	191448.2	191758.6	192725.9	0.50	0	3600	192044.0	192770.7	0.38	0	3600
10	i2000_6	161125.3	161344.2	161581.6	0.15	114	3600	161400.2	161515.4	0.07	1372	3600
	i2000_7	157918.6	158199.7	158413.3	0.13	6	3600	158368.3	158368.3	0.00	1941	2155
	i2000_8	160681.1	161042.6	161565.2	0.32	8	3600	161189.9	161524.9	0.21	53	3600
	i2000_9	160138.9	160548.1	160929.8	0.24	6	3600	160712.5	160893.4	0.11	989	3600
	i2000_10	156918.0	157188.8	157434.5	0.16	11	3600	157334.8	157424.5	0.06	1501	3600
15	i2000_11	151050.4	151306.3	151306.3	0.00	599	3086	151306.3	151306.3	0.00	350	1085
	i2000_12	153112.3	153257.6	153294.7	0.02	1104	3600	153294.7	153294.7	0.00	495	1096
	i2000_13	151543.2	151782.9	151782.9	0.00	561	2325	151782.9	151782.9	0.00	287	839
	i2000_14	151906.4	152166.4	152166.4	0.00	153	2531	152166.4	152166.4	0.00	269	953
	i2000_15	147603.5	147749.3	147749.3	0.00	157	1267	147749.3	147749.3	0.00	122	691
20	i2000_16	150481.1	150655.0	150716.2	0.04	1021	3600	150716.2	150716.2	0.00	1571	1090
	i2000_17	145530.3	145770.5	145770.5	0.00	115	1770	145770.5	145770.5	0.00	569	999
	i2000_18	146394.0	146544.2	146544.2	0.00	33	872	146544.2	146544.2	0.00	69	511
	i2000_19	147907.8	148161.1	148161.1	0.00	83	1419	148161.1	148161.1	0.00	277	744
	i2000_20	147814.8	148023.3	148023.3	0.00	251	2078	148023.3	148023.3	0.00	224	703

Table A.8: Enumeration results for the $i*$ 2000×2000 instances (time limit: 3600 secs.)

		$\hat{\text{M}}$(CFLP)	**XP**-enum					**XP + WFC**-enum				
Ratio	Name	LP value	Best LB	Best UB	%gap	Nodes	Time	Best LB	Best UB	%gap	Nodes	Time
1.1	p2000-2000-61	9406130.6	9417910.6	9425556.7	0.08	7	3600	9419625.0	9420045.0	0.00	1293	3600
	p2000-2000-62	9553583.9	9564566.5	9568183.1	0.04	18	3600	9565863.4	9566575.4	0.01	851	3600
	p2000-2000-63	9988755.1	10003328.2	10009033.9	0.06	32	3600	10004062.2	10004760.5	0.01	768	3600
	p2000-2000-64	9753128.3	9766153.3	9773373.5	0.07	35	3600	9767432.5	9768829.5	0.01	401	3600
	p2000-2000-65	9815658.2	9829285.3	9830553.8	0.01	10	3600	9830462.6	9830472.4	0.00	2301	3600
1.5	p2000-2000-66	7388283.6	7422785.4	7427559.0	0.06	14	3600	7423698.0	7426559.0	0.04	1195	3600
	p2000-2000-67	7744753.6	7783226.9	7789639.0	0.08	18	3600	7783318.4	7787608.1	0.06	724	3600
	p2000-2000-68	7255942.0	7291240.8	7296448.9	0.07	21	3600	7292097.8	7294775.0	0.04	896	3600
	p2000-2000-69	7383389.8	7421278.0	7424473.0	0.04	15	3600	7421797.7	7424499.6	0.04	628	3600
	p2000-2000-70	7256064.7	7292541.0	7297867.5	0.07	15	3600	7293743.7	7296891.3	0.04	963	3600
2	p2000-2000-71	6437955.1	6482735.2	6483849.5	0.07	354	3600	6483036.7	6483777.1	0.01	3572	3600
	p2000-2000-72	6525168.6	6569503.2	6571087.7	0.02	549	3600	6569844.8	6570727.8	0.01	3360	3600
	p2000-2000-73	6507042.7	6557920.9	6559220.2	0.02	271	3600	6557278.6	6559197.2	0.03	2775	3600
	p2000-2000-74	6321174.3	6369973.1	6373732.8	0.06	470	3600	6370010.2	6373115.5	0.05	2231	3600
	p2000-2000-75	6422548.6	6465892.3	6466402.3	0.01	560	3600	6465934.3	6466400.0	0.01	3776	3600
3	p2000-2000-76	5560615.4	5579056.2	5579061.8	0.00	502	3600	5579061.8	5579061.8	0.00	283	1065
	p2000-2000-77	5321478.8	5344223.8	5344227.6	0.00	503	3600	5344227.6	5344227.6	0.00	51	644
	p2000-2000-78	5247970.7	5266721.6	5266726.8	0.00	433	3600	5266726.8	5266726.8	0.00	147	643
	p2000-2000-79	5341576.1	5364116.2	5364116.2	0.00	9	703	5364116.2	5364116.2	0.00	77	768
	p2000-2000-80	5250214.2	5272553.9	5272553.9	0.00	27	612	5272553.9	5272553.9	0.00	45	714
5	p2000-2000-81	5015360.3	5028251.2	5028251.2	0.00	3	355	5028251.2	5028251.2	0.00	27	398
	p2000-2000-82	4975802.0	4987222.9	4987222.9	0.00	0	207	4987222.9	4987222.9	0.00	3	381
	p2000-2000-83	4899363.7	4910622.4	4910622.4	0.00	0	181	4910622.4	4910622.4	0.00	0	293
	p2000-2000-84	5030533.5	5041409.7	5041409.7	0.00	0	232	5041409.7	5041409.7	0.00	0	362
	p2000-2000-85	5114442.9	5124973.6	5124973.6	0.00	0	284	5124973.6	5124973.6	0.00	5	172
10	p2000-2000-86	4967188.3	4971500.5	4971500.5	0.00	0	178	4971500.5	4971500.5	0.00	0	75
	p2000-2000-87	4810796.1	4814108.5	4814108.5	0.00	0	165	4814108.5	4814108.5	0.00	0	77
	p2000-2000-88	4982264.2	4986966.7	4986966.7	0.00	0	203	4986966.7	4986966.7	0.00	0	76
	p2000-2000-89	5004397.1	5007857.5	5007857.5	0.00	0	340	5007857.5	5007857.5	0.00	0	147
	p2000-2000-90	4944833.7	4947837.8	4947837.8	0.00	0	194	4947837.8	4947837.8	0.00	0	104

Table A.9: Enumeration results for the $p*$ 2000×2000 instances (time limit: 3600 secs.)