Stable formulations for the Capacitated Facility Location Problem with Customer Preferences

Concepción Domínguez*[a], Juan de Dios Jaime-Alcántara[a]

[a]*Department of Statistics and Operations Research, University of Murcia, Murcia 30100, Spain*

Abstract

In the Simple Plant Location Problem with Order (SPLPO), the aim is to open a subset of plants to assign every customer taking into account their preferences. Customers rank the plants in strict order and are assigned to their favorite open plant, and the objective is to minimize the location plus allocation costs. Here, we study a generalization of SPLPO named the Capacitated Facility Location Problem with Customer Preferences (CFLCP) where a limited number of customers can be allocated to each facility. We consider the global preference maximization setting, where the customers preferences are globally maximized. We define three new types of stable allocations, namely customer stable, pairwise stable and cyclic-coalition stable allocations, and we provide two mixed-integer linear formulations for each setting. In particular, our cyclic-coalition stable formulations are Pareto optimal in a global-preference maximization setting, in the sense that no customer can improve their allocation without making another one worse off. We assess the performance of the proposed formulations and the quality of the resulting stable allocations through extensive computational experiments. As an additional contribution, we present a novel formulation that provides maximum-cardinality Pareto-optimal matchings for the Capacitated House Allocation problem.
Keywords: Location With Preferences; Capacitated Facility Location; Stable Matchings; Combinatorial Optimization; Matchings Under Preferences

1. Introduction

The Simple Plant Location Problem with Order (SPLPO) is a generalization of the well-known Facility Location Problem (FLP) that takes into account customers' preferences over the plants in the allocation process. The aim of SPLPO is to open a number of facilities from a set of candidate ones and allocate every customer, minimizing the overall cost of location plus allocation. Customers have certain preferences over the candidate locations that may be related to their size, distance, and other features, and they express their preferences ranking the candidate locations from the best to the worst (ties are not allowed in this

*Corresponding author.

Email addresses: concepcion.dominguez@um.es (Concepción Domínguez*), jd.jaimealcantara@um.es (Juan de Dios Jaime-Alcántara)

setting). Once the plants are installed, each customer is assigned to their favorite open plant. It is assumed that the locator plays no role in the allocation of customers but has full knowledge of their rankings when making the location decision.

SPLPO was introduced in Hanjoul and Peeters (1987), where the state-of-the-art formulation available to this day is provided and a heuristic is developed to tackle small instances. Hansen et al. (2004) introduce a bilevel formulation that is then reformulated into a linear model in several ways, the best being the same formulation given by Hanjoul and Peeters. Cánovas et al. (2007) build upon the work from Hanjoul and Peeters (1987), developing some valid inequalities as well as a basic preprocessing analysis, and Vasil'ev et al. (2009) also derive new valid inequalities based on the subyacent set packing polytope. The efficiency of said family of valid inequalities is checked in Vasilyev and Klimentova (2010) by implementing a cutting plane algorithm to find an optimal solution. A simulated annealing method is used to obtain the upper bounds in these exact methods. Vasilyev et al. (2013) carry out a polyhedral analysis of the problem, stating which inequalities are facet defining and developing a new set of facet-defining valid inequalities that are tested through extensive computational experiments. Very recently, Cabezas and García (2023) study some advantages of using a Lagrangian and semi-Lagrangian approach for the SPLPO, and they use the solution of a Lagrangian relaxation algorithm as the starting point of a semi-Lagrangian relaxation method. Then they apply a variable fixing heuristic to find good feasible solutions (often the optimal solution) in a so-called semi-Lagrangian relaxation heuristic algorithm. Note that, if the ranking is given by the distance to the facilities, then the preference constraints included in Hanjoul and Peeters (1987) and subsequent works match the well-known Closest Assignment Constraints (CAC). See Espejo et al. (2012) for a detailed comparison of CAC in integer programming.

In this work, we study an extension of SPLPO where each facility has a capacity constraint, that is, a limited number of customers that can be allocated to it. The problem is known as Capacitated Facility Location with Customer's Preferences (CFLCP). This is a much more general setting with many more realistic applications, such as assigning children to schools, students to universities, and patients to hospitals, among others. Its counterpart without customer preferences, the Capacitated Facility Location (CFL) problem, has been thoroughly studied (without being exhaustive, see e.g. Avella and Boccia, 2009; Görtz and Klose, 2012; Fernández and Landete, 2015; Fischetti et al., 2016). However, the combination of a cardinality constraint and customers with preferences makes the allocation rule unclear. In fact, knowing the preferences of customers and taking them into account when selecting the facilities to be opened is not enough to properly define the assignments in the presence of capacity constraints. Thus, additional assumptions regarding the customers' preferences and willingness to exchange locations are necessary and result in various allocation rules that give rise to different problems and formulations. This might be the reason why there are only a handful of references in the location literature that combine these two ingredients.

There are two main approaches to customer allocation. In the first, customers are assigned to their favorite open plant, as in SPLPO. This approach corresponds to an *individual preference maximization* setting. Naturally, CAC are to be used to model the problem and facilities that are overloaded cannot ever be opened (note that the overloading of a facility depends on the rest of the open facilities). To the best of our knowledge, there are only two works focusing on this approach. Büsing et al. (2022) analyze the CFLCP if it is defined on a graph, where the assignment costs are equal to the distances between the customer nodes and the facility nodes. They derive some settings that are NP-complete and some that are polynomially solvable, closing the research gap regarding the computational complexity of FLPs with CAC mixed with capacity and revenue constraints. Büsing et al. (2025) build upon the previous work and contribute two novel preprocessing methods, which reduce the size of the considered integer programming formulation. Furthermore, they introduce sets of valid inequalities that decrease the integrality gaps.

The second approach is the one followed in this article and consists in a *global preference maximization* setting. In this case, any facility can be opened, and if a facility is overloaded customers can be served at their next favorite one. The few works that follow this approach all introduce bilevel formulations with a second-level global-preference maximization problem. Calvete et al. (2020) introduce a bilevel formulation that is then reformulated into a single-level mixed-integer problem using duality theory. They also develop a matheuristic based on an evolutionary algorithm and compare the performance of both approaches. We build upon the work developed in this article, as it serves as motivation for the formulations hereby introduced. In Casas-Ramírez et al. (2018), a very similar bilevel model is proposed for a slightly different setting where each customer has a demand and the capacity of a facility refers to the amount of demand it can deal with. This increases the difficulty of the second-level formulation, which is a generalized assignment problem and hence NP-hard. To overcome this issue, the authors define attainable bilevel solutions based on an efficient approximation of the inducible region and develop heuristic algorithms to tackle the problem. Finally, Polino et al. (2024) develop a bilevel problem for extracurricular workshop planning that is a generalization of the previous where different types of facilities (with different capacities) can be installed in the same location. They propose a single-level reformulation and two metaheuristics (a path-relinking metaheuristic and an iterated local search procedure).

1.1. Our contribution

We analyze how the inclusion of capacity constraints poses a challenge in the characterization of solutions. As stated, the maximization of individual customers' preferences may clash with the capacity restrictions and lead to unrealistic infeasible solutions depending on the application. Moreover, we raise awareness that a global preference maximization that relies on numeric preference values as introduced in the previous bilevel models is also disadvantageous, since it can lead to biased solutions where one customer is favored or disfavored over the rest. So, the question arises: If numeric values must not characterize preferences in a global preference maximization objective function, how do we find a fair allocation that maximizes their preferences?

We give an answer to this question by replacing the second-level objective (maximizing customers' preferences) with allocations that meet different stability criteria. To this end, we introduce customer stable, pairwise stable, and cyclic-coalition stable allocations. These stable allocations meet fairness criteria in the sense that they provide unbiased solutions towards the customers. They are not based on numeric preference values, but rather on the ranking (strict ordering) of the plants given by the customers. As an additional advantage, characterizing these stable allocations can be made through constraints, so the models obtained are all mixed-integer linear or integer linear single-level models.

In a customer stable allocation, no customer can improve their position by switching to an undersubscribed plant. Therefore, customer stable solutions arise naturally in contexts where, once allocated, customers are unable to switch allocations unless their desired plant is undersubscribed. Pairwise stable settings are an extension of the previous that fit applications where customers are also allowed to swap locations, and they are willing to do so if both of them improve. Hence, pairwise stable allocations forbid assignments with pairs of clients in this situation. Lastly, cyclic-coalition stable solutions are an extension of pairwise stable solutions to any subset of customers. They are undominated in the sense that no customer can improve their allocation without making another one worse off. They constitute a non-biased global-preference-maximization setting.

These allocations are inspired by the literature on matching under preferences (see Manlove, 2013) and extended here to account for the location of the facilities. The allocation problem in our setting (capacities and one-sided preferences) corresponds to the Capacitated House Allocation problem (CHA), where a series of *applicants* are to be allocated to a series of *houses* of a certain capacity, and the applicants rank a subset of the houses in strict order. Although the customer stable and pairwise stable are not defined for matchings, our cyclic-coalition stable allocation corresponds to a Pareto optimal matching in the CHA (Manlove, 2013, Chapter 6). Note that there are no optimization models given in the matching literature or in the location literature to find any of these matchings. The fastest algorithms to find Pareto optimal matchings of maximum cardinality are given in Sng (2008) along with complexity results.

For each of the three settings, we provide an initial integer formulation and a reduced mixed-integer formulation with fewer binary variables that has some relevant properties. As an additional result, we derive a model to solve the maximum Pareto optimal CHA. We compare each initial model with its reduced counterpart by conducting extensive computational experiments with randomly created instances. Finally, we compare our Pareto optimal allocations with the (also Pareto optimal) allocations provided in

Calvete et al. (2020) and show reductions of up to 17% in the objective value.

The paper is organized as follows. The notation and motivation for the problem are given in Section 2. Three types of stable allocations are defined in Section 3. In Sections 4-6 we introduce initial and reduced customer, pairwise and cyclic-coalition stable formulations, respectively, and a maximum Pareto optimal formulation for CHA. Computational experiments are carried out in Section 7, and some conclusions and future research are provided in Section 8.

2. Notation and previous bilevel models for CFLCP

In this section, we review two formulations proposed in the literature to tackle CFLCP and provide examples to illustrate the type of solutions they provide. Recall that the aim is to open a subset of facilities and to allocate all the customers satisfying the capacity constraints. Along with these requirements, we have two competing objectives which are the maximization of customers' preferences and the minimization of the location and allocation costs.

First, let us introduce the notation that will be used throughout the work. Let $\mathcal{J}$ be the set of potential plants or facilities to be opened, and $\mathcal{I}$ be the set of customers to be allocated. Each plant $j \in \mathcal{J}$ has an associated cost f_j which refers to its opening, and for each pair i, j of customer-facility there is an associated cost g_{ij} of allocating customer i to facility j. The capacity of each facility $j \in \mathcal{J}$ is c_j. W.l.o.g., it is assumed that $c_j < |I|$ for at least one facility j (or else the problem becomes the (uncapacitated) SPLPO). Each customer $i \in \mathcal{I}$ has a ranking of the facilities from the best to the worst (ties are not allowed). The notation $j \prec_i j'$ states that i *prefers* j to j'.

As is customary in the literature, we define binary variables y_j, $j \in \mathcal{J}$, equal to 1 if and only if the corresponding facility is open. Likewise, we define binary variables x_{ij} for each customer $i \in \mathcal{I}$ and plant $j \in \mathcal{J}$, such that $x_{ij} = 1$ if and only if customer i is assigned to plant j.

The model given in Calvete et al. (2020) tackles the resolution of the CFLCP. To do so, the authors assume that each customer $i \in \mathcal{I}$ has a ranking of the facilities characterized by means of a set of predefined nonnegative preferences $p_{ij} \in \{1, \ldots, |J|\}$, $j \in \mathcal{J}$, with $p_{ij} \neq p_{ij'}$ for $j \neq j'$. The lower the value, the greater the preference: $p_{ij} < p_{ij'} \Leftrightarrow j \prec_i j'$. Their bilevel model reads:

$$\min_{\boldsymbol{y}} \quad \sum_{j \in \mathcal{J}} f_j y_j + \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} g_{ij} x_{ij} \tag{1a}$$

$$\text{s.t.} \quad \sum_{j \in \mathcal{J}} c_j y_j \geq |I|, \tag{1b}$$

$$y_j \in \{0, 1\}, \quad \forall j \in \mathcal{J}, \tag{1c}$$

$$x_{ij} \in \quad \arg\min_{\boldsymbol{x}} \quad \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} p_{ij} x_{ij} \tag{1d}$$

$$\text{s.t.} \quad \sum_{j \in \mathcal{J}} x_{ij} = 1, \quad \forall i \in \mathcal{I}, \tag{1e}$$

$$\sum_{i \in \mathcal{I}} x_{ij} \leq c_j y_j, \quad \forall j \in \mathcal{J}, \tag{1f}$$

$$x_{ij} \in \{0, 1\}, \quad \forall i \in \mathcal{I}, j \in \mathcal{J}. \tag{1g}$$

Model (1) has a bilevel structure where the first level (1a)-(1c) is the *location* (i.e., the company's decision of the facilities to open) and the second level (1d)-(1g) is the *allocation* (that is, the assignment of the customers to the facilities). The objective function of the first level (1a) is to minimize the overall costs of opening plants and allocating customers. Constraints (1b) state that there must be enough open plants to allocate every customer. This constraint is redundant, but it is added because it acts as a global feasibility cut guaranteeing that any feasible location $\boldsymbol{y}$ of the first-level problem admits a feasible allocation $\boldsymbol{x}$. The objective function of the second level (1d) is to maximize the preferences of customers globally. Constraints (1e) ensure that every customer is allocated to one plant, whereas the *capacity constraints* (1f) guarantee that the capacity of each plant is not exceeded. Finally, constraints (1c) and (1g) express the binary character of the variables in the model. When constraints (1f) are included, the set of constraints ensuring that customers are allocated only to open plants,

$$x_{ij} \leq y_j, \quad \forall i \in \mathcal{I}, j \in \mathcal{J}, \tag{2}$$

is redundant. However, they are known to act as valid inequalities, strengthening the linear relaxation bounds of the second-level problem. Model (1) is reformulated in Calvete et al. (2020), obtaining a single-level mixed-integer linear model that can be solved using off-the-shelf solvers.

Note that the second-level problem (1d)-(1g) is separable by customer in the uncapacitated SPLPO, since there are no linking constraints (1f) (but (2) should be included). On the other hand, if the capacity constraints (1f) are included in the first level, then again the second level problem is separable by customer, and thus each customer must be allocated to their most preferred open facility. As stated in Calvete et al. (2020), the main drawback of this approach is that, since customers choose *individually*, the allocations given by the solutions of the second-level problem are very frequently infeasible due to the capacity restrictions. Hence, the proposal of the bilevel model (1), in which the preferences of the customers are *globally* maximized and for any location satisfying (1b) there exists a feasible allocation.

In the following example, we use several toy preference matrices to illustrate how the values of the parameters p_{ij} can affect the allocations given by the second-level problem in model (1) and provide biased solutions.

Example 1. *Consider a toy instance with* $|\mathcal{I}| = |\mathcal{J}| = 4$ *and capacities* $c_j = 2 \ \forall j \in \mathcal{J}$*. Assuming that the costs of opening a plant greatly surpass those of allocating customers, i.e.,* $f_j \gg g_{ij}$*, exactly two plants are*

Table 1: Preference matrices for Example 1. All the plants have capacity equal to 2.

	Cust. 1	Cust. 2	Cust. 3	Cust. 4
Plant 1	1	1	1	1
Plant 2	2	3	4	10
Plant 3	3	9	16	100
Plant 4	4	27	64	1000

(a) $p_{ij} \notin \{1, \ldots, 4\}$, $\forall i \in \mathcal{I}$, $j \in \mathcal{J}$

	Cust. 1	Cust. 2	Cust. 3	Cust. 4
Plant 1	1	1	2	3
Plant 2	2	2	4	2
Plant 3	3	4	3	1
Plant 4	4	3	1	4

(b) $p_{ij} \in \{1, \ldots, 4\}$, $\forall i \in \mathcal{I}$, $j \in \mathcal{J}$

open in an optimal solution of the bilevel problem. We are interested in the optimal allocations provided by the second-level problem (1d)-(1g) *for different solutions of the first-level problem.*

First, consider the preference matrix (p_{ij}) *given in Table 1a. In this case, the ranking of the customers is clearly the same for all of them:* $1 \prec_i 2 \prec_i 3 \prec_i 4$ $\forall i \in \mathcal{I}$. *In theory, then, any feasible solution consists in allocating two customers to their favorite open plant, and the other two customers to their second favorite plant, so any feasible allocation should be optimal for the second-level problem. However, due to the values* p_{ij} *assigned to the preferences of each customer, only one allocation is optimal if two plants are open: customers 3 and 4 are allocated to their most preferred plant (between the two open ones), whereas customers 1 and 2 are allocated to their second favorite. This biased allocation where the preferences of customers 3 and 4 are always fulfilled before the preferences of the other two customers regardless of the location of the plants occurs because the difference between the values* p_{ij} *is greater for customers 3 and 4 than for customers 1 and 2.*

In the previous setting it can be argued that the feasible allocations are influenced by the fact that the values p_{ij} *are very different among customers and they are not in the set* $\{1, \ldots, |\mathcal{J}|\}$. *To illustrate how the allocations can be biased even if* $p_{ij} \in \{1, 2, \ldots, |\mathcal{J}|\}$, *consider the same setting with the preference matrix given in Table 1b. Suppose plants 1 and 2 are open. Then customers 1, 2 and 3 would rather be allocated to plant 1 (and customer 4 to plant 2) but plant 1 has capacity equal to 2, so the preferences of one customer will not be maximized. Thus, in principle any of the three feasible allocations with customer 4 allocated to plant 2 is maximizing the preferences of three customers and should be optimal. In fact, two of these allocations are optimal: allocating customer 3 to plant 1 and customer 4 to plant 2 (and customers 1 and 2 each to a plant indistinctly) provides solutions with objective value 7. However, if customers 1 and 2 are allocated to plant 1 and customers 3 and 4 to plant 2, the solution is not optimal for the second-level problem (and thus it is not feasible), since is has objective value equal to 8. In this case, customer 3 is favored over customers 1 and 2.*

Consider now that plants 1 and 3 are open. Then a similar situation arises where customers 1, 2 and 3 *prefer plant 1 over plant 3, but the* only *optimal solution of the second-level problem consists in allocating customers 1 and 2 to plant 1 and customers 3 and 4 to plant 3. In this case, the preferences of customer 3 are dismissed in favor of those of customers 1 and 2.*

As can be seen, although the second-level problem provides an optimal solution where customers maximize their preferences, it does not provide *all* the solutions where this happens. The numbers p_{ij} have no particular meaning: they are only included to express the ordering in the ranking of each customer. Yet, they influence the number and type of optimal solutions of the second level, limiting the number of feasible allocations for given first-level solutions in an undesirable manner. Finally, note that there is no bias when maximizing preferences expressed using numeric values in the uncapacitated SPLPO, because the preferences of the customers are satisfied individually, so the preferences of a customer are not compared to those of others.

3. Three new types of stable allocations for the CFLCP

As illustrated by means of Example 1, models that maximize numeric preferences are biased with respect to the preferences of certain customers, regardless of the actual numbers that we choose for parameters p_{ij}. Therefore, to formulate the problem we avoid the modeling of preferences using numeric values. Instead, we consider only the individual ranking of each customer, and we characterize solutions by the type of allocations they provide. To do so, we define three types of allocations: customer stable, pairwise stable and cyclic-coalition stable allocations. The concept of stability has been previously used in the context of matchings under preferences (see Manlove, 2013), and we redefine it here for our setting. We first introduce blocking customers, blocking pairs and blocking sets in an allocation. Recall that throughout the paper we assume that customers have strict preferences over plants.

Definition 1. *Given a feasible allocation A in an instance of CFLCP, a customer i, allocated to plant j, is a* blocking customer *if the following conditions are satisfied:*

1. *i prefers plant j' to his assigned plant j in A,*
2. *j' is open, and*
3. *there are less than $c_{j'}$ customers allocated to j' in A.*

Furthermore, A is customer stable *if it has no blocking customers.*

Definition 2. *Let A be an allocation in an instance of CFLCP. A pair of customers (i, i') allocated respectively to plants j, j' $(j \neq j')$ is a* blocking pair *for A if i prefers j' over his current allocation j and i' prefers j over j'.*

Furthermore, A is said to be pairwise stable *if it does not admit blocking customers or blocking pairs.*

Definition 3. *A set of customers* $\bar{I} = \{i_1, \ldots, i_n\}$*,* $|\bar{I}| \geq 3$*, allocated to plants* $\bar{J} = \{j_1, \ldots, j_n\}$ *respectively in an allocation* A*, is a* blocking set *if there exists a bijection* $\gamma : \bar{I} \to \bar{J}$ *such that every customer* i_t *prefers* $\gamma(i_t)$ *to its current allocation* j_t*.*

Moreover, A *is* cyclic-coalition stable *if it has no blocking customers, blocking pairs, or blocking sets.*

In an instance of CHA, our customer stable allocation corresponds to a maximal trade-in-free matching, our cyclic-coalition stable allocation corresponds to a maximal, trade-in-free and cyclic coalition-free matching and our pairwise stable allocation is not defined. According to (Manlove, 2013, Prop. 6.2 in page 306), any matching is Pareto optimal if and only if it is maximal, trade-in-free and cyclic coalition-free. Therefore, the notion of cyclic-coalition stability introduced here is equivalent to their Pareto optimality criterion. However, we adopt this terminology in order to maintain consistency with the other stability notions introduced in this paper. By extension, a formulation for the CFLCP is customer stable (resp. pairwise stable, cyclic-coalition stable) if any feasible allocation is customer stable (resp. pairwise stable, cyclic-coalition stable).

These three types of allocations fit different applications, depending on the circumstances under which we assume that customers can change their allocation. Intuitively, when we seek for a customer stable allocation we are assuming that, once allocated, customers can only change their plant for another one that they prefer over their current one if it is undersubscribed. As for pairwise stable and cyclic-coalition stable allocations, they fit applications where customers are also able to swap allocations (in the case of pairwise stable) or *rotate* (for three or more customers in the cyclic-coalition stable allocations) if, in doing so, all of them improve in terms of their preferences (hence the name Pareto optimal in the related matching literature).

Remark 1. *Model (1) provides cyclic-coalition stable allocations. In fact, consider an allocation* A *that is not cyclic-coalition stable. If there is a blocking customer* i*, then* i *is allocated to* j *but prefers* j'*, and* j' *is undersubscribed. Hence,* $p_{ij'} < p_{ij}$ *and reassigning* i *to* j *produces an allocation with strictly lower objective value (1d). Alternatively, there exists a blocking pair or a blocking set of customers* $\bar{I} = \{i_1, \ldots, i_n\}$*,* $|\bar{I}| \geq 2$*, allocated to different plants* $\bar{J} = \{j_1, \ldots, j_n\}$*, and a bijection* γ *such that* i_t *prefers* $\gamma(i_t)$ *to its current allocation* j_t*. But this implies* $p_{i_t\gamma(i_t)} < p_{i_tj_t}$ *for all* $t \in \{1, \ldots, n\}$*, so there exists a feasible allocation with strictly lower objective value (1d). Hence, in any case the initial allocation* A *is infeasible for (1).*

Remark 2. *Not every cyclic-coalition stable allocation is feasible for model (1). To illustrate this, consider Example 1 for the preference matrix given in Table 1b. Consider the allocation given by customers 1 and 2 allocated to plant 1, and customers 3 and 4 allocated to plant 2 (only plants 1 and 2 are open). This allocation is infeasible for model (1), but is cyclic-coalition stable.*

In the following sections, we introduce customer stable, pairwise stable and cyclic-coalition stable formulations for the CFLCP.

4. Customer stable models for the CFLCP

In this section, we begin by introducing a formulation that minimizes the overall cost of locating plants and allocating customers, and that provides customer stable allocations. To do so, consider the notation and variables introduced in Section 2. To replace the numeric preferences, for each customer $i \in \mathcal{I}$ we use the previously defined total ordering $\prec_i$ such that $j \prec_i j'$ (resp. $j \succ_i j'$) if customer i prefers plant j over plant j' (resp. plant j' over plant j). Furthermore, $j \preceq_i j'$ if customer i prefers plant j over plant j' or $j = j'$. With this notation, we introduce an integer linear formulation:

$$
\begin{aligned}
(\texttt{CFLCP})_{\texttt{CS}} \quad \min_{\boldsymbol{x},\boldsymbol{y}} \quad & \sum_{j\in\mathcal{J}} f_j y_j + \sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}} g_{ij}x_{ij} && \text{(3a)}\\
\text{s.t.} \quad & \sum_{j\in\mathcal{J}} x_{ij} = 1, \quad \forall i \in \mathcal{I}, && \text{(3b)}\\
& \sum_{i\in\mathcal{I}} x_{ij} \le c_j y_j, \quad \forall j \in \mathcal{J}, && \text{(3c)}\\
& x_{ij} \le y_j, \quad \forall i \in \mathcal{I}, j \in \mathcal{J}, && \text{(3d)}\\
& c_j y_j \le c_j \sum_{\substack{j'\in\mathcal{J}:\\ j'\preceq_i j}} x_{ij'} + \sum_{i'\in\mathcal{I}\setminus\{i\}} x_{i'j}, \quad \forall i \in \mathcal{I}, j \in \mathcal{J}, && \text{(3e)}\\
& x_{ij} \in \{0,1\}, \quad \forall i \in \mathcal{I}, j \in \mathcal{J}, && \text{(3f)}\\
& y_j \in \{0,1\}, \quad \forall j \in \mathcal{J}. && \text{(3g)}
\end{aligned}
$$

The objective (3a) is to minimize the location and allocation costs. Constraints (3b) and (3c) ensure that all customers are allocated and the capacity of the plants is fulfilled. Constraints (3d) are redundant but are known to strengthen the linear relaxation. Constraints (3e) guarantee that the allocation is customer stable. Indeed, when plant j is open but customer i is allocated to a plant $j' \succ_i j$, then $y_j = 1$ and $\sum_{j'\in\mathcal{J}:j'\preceq_i j} x_{ij'} = 0$. In this case, (3e) force plant j to be full. Constraints (3f)-(3g) ensure the binary nature of the variables.

Proposition 1. *Every feasible allocation provided by model* $(\texttt{CFLCP})_{\texttt{CS}}$ *is customer stable.*

Proof. Assume that allocation A is not customer stable, and let i be a blocking customer, and $j, j' \in \mathcal{J}$ under the conditions of Definition 1: $x_{ij} = 1$, $j' \prec_i j$ and $\sum_{i'\in\mathcal{I}} x_{i'j'} < c_{j'}$. Then for the pair $i \in \mathcal{I}$, $j' \in \mathcal{J}$, constraint (3e) is not satisfied: the left-hand side of (3e) is equal to $c_{j'}$ due to 2., whereas $\sum_{j''\in\mathcal{J}:j''\preceq_i j'} c_{j'}x_{ij''} = 0$ due to 1. and $\sum_{i'\in\mathcal{I}\setminus\{i\}} x_{i'j'} < c_{j'}$ due to 3. Hence, if an allocation is feasible for (3), it is customer stable. □

Table 2: Small instance of the CFLCP. $c_j = 3$ and $f_j = 2\ \forall j \in \mathcal{J}$.

	Cust. 1	Cust. 2	Cust. 3	Cust. 4	Cust. 5	Cust. 6	Cust. 7	Cust. 8
Plant 1	3	3	3	1	3	1	3	1
Plant 2	3	3	3	2	2	3	1	2
Plant 3	3	3	1	1	1	1	2	3
Plant 4	3	3	1	3	3	2	1	1

(a) Cost matrix (g_{ij}), $\forall i \in \mathcal{I}$, $j \in \mathcal{J}$.

	Cust. 1	Cust. 2	Cust. 3	Cust. 4	Cust. 5	Cust. 6	Cust. 7	Cust. 8
Plant 1	3	3	2	1	4	3	2	4
Plant 2	4	2	3	2	1	1	3	1
Plant 3	2	4	4	3	3	2	1	3
Plant 4	1	1	1	4	2	4	4	2

(b) Preference matrix (p_{ij}), $\forall i \in \mathcal{I}$, $j \in \mathcal{J}$.

In the uncapacitated version of this problem, it is well-known that the integrality constraints on the x-variables can be relaxed. However, as we see in the following, this is not the case for the previous model.

Proposition 2. *In general, the integrality constraints on the x-variables (3f) cannot be relaxed in model* $\texttt{(CFLCP)}_{\texttt{CS}}$.

Proof. Consider an instance such that $|\mathcal{I}| = 8$, $|\mathcal{J}| = 4$, the capacity of all plants is equal to 3, and the cost f_j of opening plant j is equal to 2 for any $j \in \mathcal{J}$. The allocation cost matrix and the preference matrix can be observed in Table 2. The previous instance constitutes a counterexample. The optimal solution given by formulation $\texttt{(CFLCP)}_{\texttt{CS}}$ with constraints (3f) relaxed has objective value equal to 18.2. Since all coefficients in the objective function are integer, there are no feasible solutions with integer x variables that produce this optimal value. Actually, the true optimal value of this instance is 19. □

In the following, we present an alternative customer stable model for which the integrality constraints on the x-variables can be relaxed. To state it, we use an additional set of binary variables u_j, $j \in \mathcal{J}$, equal to 1 if and only if plant j is full in the solution.

$$\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CS}} \quad \min_{\boldsymbol{x},\boldsymbol{y}} \quad \sum_{j\in\mathcal{J}} f_j y_j + \sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{J}} g_{ij} x_{ij} \tag{4a}$$

$$\text{s.t.} \quad \sum_{j\in\mathcal{J}} x_{ij} = 1, \quad \forall i \in \mathcal{I}, \tag{4b}$$

$$\sum_{i\in\mathcal{I}} x_{ij} \le (c_j - 1)y_j + u_j, \quad \forall j \in \mathcal{J}, \tag{4c}$$

$$c_j u_j \le \sum_{i\in\mathcal{I}} x_{ij}, \quad \forall j \in \mathcal{J}, \tag{4d}$$

$$x_{ij} \le y_j, \quad \forall i \in \mathcal{I}, j \in \mathcal{J}, \tag{4e}$$

$$y_j \le u_j + \sum_{\substack{j'\in\mathcal{J}:\\ j'\preceq_i j}} x_{ij'}, \quad \forall i \in \mathcal{I}, j \in \mathcal{J}, \tag{4f}$$

$$x_{ij} \in \{0,1\}, \quad \forall i \in \mathcal{I}, j \in \mathcal{J}, \tag{4g}$$

$$u_j, y_j \in \{0,1\}, \quad \forall j \in \mathcal{J}. \tag{4h}$$

Constraints (4c) and (4d) are now the *capacity constraints*: (4c) ensure that $u_j = 1$ when plant j is full, whereas (4d) guarantee that $u_j = 0$ when j is undersubscribed. In addition, constraints (4f) prohibit blocking customers. When $y_j = 1$, either the plant is full ($u_j = 1$) or customer i is allocated to a plant j' he likes the same or more than j ($\sum_{j'\in\mathcal{J}:j'\preceq_i j} x_{ij'} = 1$).

Proposition 3. *The integrality constraints on the x-variables (4g) can be relaxed in the model* $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CS}}$.

Proof. For given integer values of variables y and u, let us consider a partition of $\mathcal{J}$ into three sets:

- $\mathcal{J}_1 := \{j \in \mathcal{J} : y_j = u_j = 0\}$ is the set of unopened plants,
- $\mathcal{J}_2 := \{j \in \mathcal{J} : y_j = 1, u_j = 0\}$ is the set of undersubscribed plants,
- $\mathcal{J}_3 := \{j \in \mathcal{J} : y_j = u_j = 1\}$ is the set of full plants.

Using this partition, we can rewrite the constraints in formulation $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CS}}$ eliminating y-variables and redundant constraints as follows:

$$\min_{\boldsymbol{x}} \quad \sum_{j\in\mathcal{J}_2\cap\mathcal{J}_3} f_j + \sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}} g_{ij}x_{ij} \tag{5a}$$

$$\text{s.t.} \quad \sum_{j\in\mathcal{J}} x_{ij} = 1, \quad \forall i \in \mathcal{I}, \tag{5b}$$

$$\sum_{i\in\mathcal{I}} x_{ij} \le 0, \quad \forall j \in \mathcal{J}_1, \tag{5c}$$

$$\sum_{i\in\mathcal{I}} x_{ij} \le (c_j - 1), \quad \forall j \in \mathcal{J}_2, \tag{5d}$$

$$\sum_{i\in\mathcal{I}} x_{ij} = c_j, \quad \forall j \in \mathcal{J}_3, \tag{5e}$$

$$1 \le \sum_{\substack{j'\in\mathcal{J}:\\ j'\preceq_i j}} x_{ij'}, \quad \forall i \in \mathcal{I}, j \in \mathcal{J}_2, \tag{5f}$$

$$x_{ij} \in \{0,1\}, \quad \forall i \in \mathcal{I}, j \in \mathcal{J}. \tag{5g}$$

Assuming w.l.o.g that $J_2 \neq \varnothing$, let j_i be the favorite plant of customer $i \in \mathcal{I}$. Then constraint (i, j_i) dominates the rest of the constraints (i, j) from set (5f). Hence, combining constraints (5b) and (5f), we can further reduce the formulation to:

$$\min_{\boldsymbol{x}} \quad \sum_{j \in \mathcal{J}_2 \cap \mathcal{J}_3} f_j + \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} g_{ij} x_{ij} \tag{6a}$$

$$\text{s.t.} \quad \sum_{\substack{j \in \mathcal{J}: \\ j \preceq_i j_i}} x_{ij} = 1, \quad \forall i \in \mathcal{I}, \tag{6b}$$

$$(5c) - (5e) \tag{6c}$$

$$x_{ij} \in \{0, 1\}, \quad \forall i \in \mathcal{I}, j \in \mathcal{J}. \tag{6d}$$

The constraints matrix in (6) is totally unimodular (Wolsey and Nemhauser, 1999, Corollary 2.8. for the partition of rows given by constraints $\{$(6b) and $\{$(5c) $-$ (5e)$\}\}$ and the right-hand side vector is integer, so the integrality constraints (6d) can be relaxed (Wolsey and Nemhauser, 1999, Proposition 2.3.). □

Note that since usually $|\mathcal{J}| \ll |\mathcal{I}|$, the number of variables in model (4) is larger compared to model (3), but the number of binary variables has been drastically reduced from $|\mathcal{J}| \cdot (|\mathcal{I}| + 1)$ to only $2|\mathcal{J}|$.

5. Pairwise stable models for the CFLCP

We begin the section with a pairwise stable formulation that uses the initial sets of variables y_j, $\forall j \in \mathcal{J}$, for the location of the plants and x_{ij}, $\forall i \in \mathcal{I}$, $j \in \mathcal{J}$, for the allocation of the customers:

$$\texttt{(CFLCP)}_{\texttt{PW}} \quad \min_{\boldsymbol{x}, \boldsymbol{y}} \quad \sum_{j \in \mathcal{J}} f_j y_j + \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} g_{ij} x_{ij} \tag{7a}$$

$$\text{s.t.} \quad (3b) - (3d) \tag{7b}$$

$$c_j (y_j + x_{ij'} - 1) \leq \sum_{\substack{i' \in \mathcal{I} \setminus \{i\}: \\ j \prec_{i'} j'}} x_{i'j}, \quad \forall i \in \mathcal{I}, j, j' \in \mathcal{J} : j \prec_i j', \tag{7c}$$

$$x_{ij} \in \{0, 1\}, \quad \forall i \in \mathcal{I}, j \in \mathcal{J}, \tag{7d}$$

$$y_j \in \{0, 1\}, \quad \forall j \in \mathcal{J}. \tag{7e}$$

Constraints (3b)-(3d) are the same as in formulation $\texttt{(CFLCP)}_{\texttt{CS}}$. In addition, constraints (7c) guarantee that there are no blocking customers or blocking pairs. If $y_j + x_{ij'} - 1 = 1$, then j is open, i is allocated to j' and i prefers j over j'. Then there must exist c_j customers allocated to j that also prefer j over j'.

Constraints (7c) in $\texttt{(CFLCP)}_{\texttt{PW}}$ share some similarities with those in the integer programming model presented in Kwanashie and Manlove (2014) and strengthened in Delorme et al. (2019) for the Hospitals/Residents problem. However, their definition of blocking pair is different from the one stated here:

we consider a pair of customers (which corresponds with a pair of residents), while they consider a pair hospital-resident to define the blocking pair because in their setting both hospitals and residents have preferences.

Proposition 4. *In general, the integrality constraints on the x-variables (7d) cannot be relaxed in model* $\texttt{(CFLCP)}_{\texttt{PW}}$.

Proof. As a counterexample, consider the instance given in Table 2 with $|\mathcal{I}| = 8$, $|\mathcal{J}| = 4$, the capacity of all plants is equal to 3, and the cost f_j of opening plant j is equal to 2 for any $j \in \mathcal{J}$. The optimal solution given by formulation $\texttt{(CFLCP)}_{\texttt{PW}}$ with (7d) relaxed has objective value equal to 18.8, whereas the optimal value of this instance is 19. $\square$

To provide a reduced pairwise stable formulation, we use variables u_j, $\forall j \in \mathcal{J}$, defined in the previous section for model (4). We also define an additional set of binary variables that help avoid blocking pairs. Thus, $v_{jj'} \in \{0,1\}$ $\forall j, j' \in \mathcal{J}$, $j \neq j'$, such that $v_{jj'} = 1$ if there exists a customer $i \in \mathcal{I}$ who prefers j over j' but is assigned to j'.

$$\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{PW}} \quad \min_{\boldsymbol{x},\boldsymbol{y},\boldsymbol{v}} \quad \sum_{j \in \mathcal{J}} f_j y_j + \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} g_{ij} x_{ij} \tag{8a}$$

$$\text{s.t.} \quad (4b) - (4f), \tag{8b}$$

$$v_{jj'} + v_{j'j} \leq 1, \quad \forall j \neq j' \in \mathcal{J}, \tag{8c}$$

$$x_{ij'} \leq v_{jj'}, \quad \forall i \in \mathcal{I}, j, j' \in \mathcal{J} : j \prec_i j', \tag{8d}$$

$$x_{ij} \in \{0,1\}, \quad \forall i \in \mathcal{I}, j \in \mathcal{J}, \tag{8e}$$

$$u_j, y_j \in \{0,1\}, \quad \forall j \in \mathcal{J}, \tag{8f}$$

$$v_{jj'} \in \{0,1\}, \quad \forall j \neq j' \in \mathcal{J}. \tag{8g}$$

Constraints (4b)-(4f) are the same as in model $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CS}}$, and provide a formulation that yields customer stable allocations. Moreover, constraints (8d) ensure that if there is a customer i assigned to j' but prefers j over j', then $v_{jj'} = 1$. Together with (8c), these constraints forbid blocking pairs.

Proposition 5. *The integrality constraints on the x-variables (8e) can be relaxed in model* $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{PW}}$.

Proof. For given integer values of variables y, u and v, observe that the constraints in model $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{PW}}$ involving the variables x match those in $\texttt{(CFLCP)}_{\texttt{PW}}$, except for constraints (8d).

From the proof of Proposition 3, it follows that the constraint matrix, excluding (8d), is totally unimodular. The addition of constraints (8d) does not alter this property, as it merely appends rows of zeros with a single one into the matrix, preserving total unimodularity.

The right-hand side vector is integer, so the integrality constraint (8e) can be relaxed (Wolsey and Nemhauser, 1999, Proposition 2.3). $\square$

We compare formulations $\texttt{(CFLCP)}_{\texttt{CS}}$ and $\texttt{(CFLCP)}_{\texttt{PW}}$ with their reduced counterparts in Section 7.

6. Cyclic-coalition stable models for the CFLCP

In this section, we minimize the overall cost of location and allocation considering only cyclic-coalition stable allocations for customers.

6.1. Basic Model

Our first integer model is based on the following characterizations of cyclic-coalition stable allocations:

Lemma 6. *Consider an instance of* CFLCP *with a set of customers $\mathcal{I} = \{1, 2, \ldots, |\mathcal{I}|\}$ and a set of plants $\mathcal{J}$. Let $\mathcal{J}' \subseteq \mathcal{J}$ with $|\mathcal{J}'| \geq |\mathcal{I}|$ and let $\sigma : I \to I$ be a permutation of the customers in I. Next, consider the allocation that results from allocating customers in the order given by σ (that is, first $\sigma(1)$, then $\sigma(2)$ and so on) to their favorite plant (not full) in $\mathcal{J}'$. The resulting allocation is cyclic-coalition stable.*

Proof. Let $\bar{I} \subseteq \mathcal{I}$ be a set of customers allocated to different plants $\bar{J}$, respectively, following the order given by permutation σ. Let $\bar{i} := \min\{i \in \bar{I}\}$. Then customer $\sigma(\bar{i})$ was allocated to his favorite plant among the ones in $\bar{J}$ (none of them was at full capacity because the rest of the customers in $\bar{I}$ were allocated afterwards). Hence, $\bar{i}$ prefers their choice to that of the customers in $\bar{I}$, so $\bar{I}$ is not a blocking set (or a blocking pair if $|\mathcal{I}| = 2$). Since each customer is allocated to their favorite undersubscribed plant, the fact that there are no blocking customers is straightforward. $\square$

Lemma 7. *Every blocking set contains a blocking subset in which the assigned plants are all distinct.*

Proof. Let $\bar{\mathcal{I}} = \{i_1, i_2, \ldots, i_s\} \subseteq \mathcal{I}$, with corresponding assigned plants $j_1, j_2, \ldots, j_s \in \mathcal{J}$, and let $\gamma : \{1, 2, \ldots, s\} \to \{1, 2, \ldots, s\}$ be a bijection such that each customer i_t prefers plant $j_{\gamma(t)}$ over its current allocation j_t, as in the definition of blocking set. Suppose, by contradiction, that there exist $a, b \in \{1, \ldots, s\}$, with $a \neq b$, such that $j_a = j_b$.

If $\gamma^r(a) \neq b$ for all $r = 1, 2, \ldots, s-1$, where γ^r denotes the composition of γ with itself r times, consider the subset of customers $\{i_{\gamma^r(a)} : r = 1, 2, \ldots, s-1\}$ along with their assigned plants and the induced permutation. This subset excludes the specific repetition between j_a and j_b.

If instead $\gamma^r(a) = b$ for some $r = 1, 2, \ldots, s-1$, let $r^\star$ be the smallest such index. Consider the subset $\{i_{\gamma^r(a)} : r = 1, 2, \ldots, r^\star\} \setminus \{i_a\}$ with their assigned plants and the induced permutation $\gamma^\star$, modified so that $\gamma^\star(b) = \gamma(a)$. This again eliminates the repetition between j_a and j_b.

In both cases, the resulting subset remains a blocking set or a blocking customer, since the induced permutation preserves the preference relations among the remaining customers.

Since the customer set is finite, we can repeat this procedure to eliminate all repeated plant assignments. Thus, we eventually obtain a blocking subset in which the assigned plants are all distinct or a blocking pair. □

The proposed model creates a permutation in the customers' set to allocate the customers, preventing any blocking customer/pair/set. With this aim, we use a set of binary variables y_j, $j \in \mathcal{J}$, equal to 1 if j is open. Next, let x_{ij}^r, for $i, r \in \mathcal{I}$, $j \in \mathcal{J}$, be a binary variable equal to 1 if and only if the customer i is assigned to the plant j and $\sigma(i) = r$ in the permutation σ that gives rise to the feasible allocation. Furthermore, let u_j^r, for $j \in \mathcal{J}$, $r \in \mathcal{I}$, be a binary variable that takes value 1 if and only if the plant j fills up when the customer $i = \sigma^{-1}(r)$ is assigned. With these variables, the model is:

$$\texttt{(CFLCP)}_{\texttt{CC}} \quad \min_{\boldsymbol{x},\boldsymbol{y},\boldsymbol{u}} \quad \sum_{j\in\mathcal{J}} f_j y_j + \sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}}\sum_{r\in\mathcal{I}} g_{ij} x_{ij}^r \tag{9a}$$

$$\text{s.t.} \quad \sum_{j\in\mathcal{J}}\sum_{r\in\mathcal{I}} x_{ij}^r = 1, \quad \forall i \in \mathcal{I}, \tag{9b}$$

$$\sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}} x_{ij}^r = 1, \quad \forall r \in \mathcal{I}, \tag{9c}$$

$$\sum_{r\in\mathcal{I}} x_{ij}^r \le y_j, \quad \forall i \in \mathcal{I}, j \in \mathcal{J}, \tag{9d}$$

$$\sum_{r\in\mathcal{I}} u_j^r \le y_j, \quad \forall j \in \mathcal{J}, \tag{9e}$$

$$\sum_{j\in\mathcal{J}} u_j^r \le 1, \quad \forall r \in \mathcal{I}, \tag{9f}$$

$$\sum_{i\in\mathcal{I}}\sum_{r'=1}^{r} x_{ij}^{r'} \le (c_j - 1) y_j + \sum_{r'=1}^{r} u_j^{r'}, \quad \forall j \in \mathcal{J}, r \in \mathcal{I} \tag{9g}$$

$$c_j \sum_{r'=1}^{r} u_j^{r'} \le \sum_{i\in\mathcal{I}}\sum_{r'=1}^{r} x_{ij}^{r'}, \quad \forall j \in \mathcal{J}, r \in \mathcal{I} \tag{9h}$$

$$y_j - \sum_{r'=1}^{r-1} u_j^{r'} \le \sum_{j'\preceq_i j}\sum_{r'=1}^{r} x_{ij'}^{r'} + \sum_{j'\in\mathcal{J}}\sum_{r'=r+1}^{|\mathcal{I}|} x_{ij'}^{r'}, \quad \forall i, r \in \mathcal{I}, j \in \mathcal{J}, \tag{9i}$$

$$y_j, x_{ij}^r, u_j^r \in \{0,1\}, \quad \forall i, r \in \mathcal{I}, j \in \mathcal{J}. \tag{9j}$$

Constraints (9b) ensure that i is assigned to a single $j \in \mathcal{J}$, and together with (9c) they guarantee that σ is a permutation. Constraints (9d) guarantee that i is allocated to an open plant. They are redundant when (9g) are included, but are known to strengthen the linear relaxation in a similar manner to (3d). Constraints (9e) impose that a plant can only fill up if it is open in the feasible solution. Constraints (9f) ensure that up to one plant j fills up with customer $\sigma(r)$. These constraints again are redundant

Table 3: Preference matrix for Example 2. All the plants have capacity equal to 3.

	Cust. 1	Cust. 2	Cust. 3	Cust. 4	Cust. 5	Cust. 6	Cust. 7	Cust. 8
Plant 1	1	2	3	3	2	1	1	1
Plant 2	2	1	2	1	1	3	2	3
Plant 3	3	3	1	2	3	2	3	2

when (9g) and (9h) are included, but significantly strengthen the formulation. Due to the capacity of the plants, it must hold $\sum_{i\in\mathcal{I}}\sum_{r'=1}^{r} x_{ij}^{r'} \leq c_j$ for any j, r. When $\sum_{i\in\mathcal{I}}\sum_{r'=1}^{r} x_{ij}^{r'} = c_j$, (9g) force $u_j^r = 1$; and when $\sum_{i\in\mathcal{I}}\sum_{r'=1}^{r} x_{ij}^{r'} < c_j$, (9h) guarantee that $u_j^r = 0$. Constraints (9i) ensure that the solution is cyclic-coalition stable, i.e. that every customer is allocated to their most preferred plant (among the ones that are open and not full) following the order given by permutation σ. Consider a plant j that is open ($y_j = 1$) and not full when the first $r-1$ customers are allocated ($\sum_{r'=1}^{r-1} u_j^{r'} = 0$). Then for any customer i and for r, the left-hand side of the corresponding constraint in (9i) is equal to 1, so one of the two sums of the right-hand side must be equal to one. In the first case, $i = \sigma(\bar{r})$ with $\bar{r} \leq r$ and i is allocated either to plant j or to a plant better than j to i. In the second case, $\bar{r} > r$ and thus the constraint holds.

6.2. Reduced Model

In this section, we propose a reduced model that provides the same set of cyclic-coalition stable allocations as model $\texttt{(CFLCP)}_{\texttt{CC}}$. Before introducing it, we illustrate the disadvantages of the previous one and the reasoning leading to the reduced model by means of an example.

Example 2. *Consider the toy instance with* $\mathcal{I} = 8$*,* $\mathcal{J} = 3$*,* $c_j = 3\ \forall j \in \mathcal{J}$ *and the preference matrix given in Table 3. Since all the customers need to be allocated, all the facilities must be open in any feasible allocation and two of them fill up.*

We can characterize every cyclic-coalition stable allocation following Lemma 6. For instance, permutation $P = (1, 2, 3, 4, 6, 7, 5, 8)$ *results in customers* $\{1, 6, 7\}$ *being allocated to plant* 1*,* $\{2, 4, 5\}$ *to plant* 2*, and* $\{3, 8\}$ *to plant* 3*. The number of permutations of the customers in this toy instance amounts to* $8! = 40320$*. However, many different permutations result in the same allocation. For instance, permutation* $P' = (5, 2, 6, 7, 1, 3, 4, 8)$ *results in the same allocation than* P*, as well as any permutation where customer* 8 *gets to choose after customers* 1*,* 6 *and* 7*. This is due to the fact that there are four customers whose first choice is plant* 1*, so the plant is full when* 8 *gets to choose. In fact, there are only 10 cyclic-coalition stable allocations for this instance, and all of them can be found in Table 4 (the allocation given by* P *is the first one). Given that studying every single permutation does not seem very efficient, let us see how the amount of permutations to consider can be reduced.*

The first six customers of P are allocated to their favorite plant because all of the plants are available. However, plant 1 fills up with customer 7, so 5 and 8 can only choose between plants 2 and 3. Customer 5 prefers plant 2 over 3 (casually, plant 2 is his favorite) and after his allocation plant 2 is also full. Finally, 8 is allocated to plant 3. Clearly, any rearrangement of the first seven customers results in the same allocation, because their favorite plant is not full when they get to choose. So, as a representative of the permutation P, we can consider $\bar{P} = (1, 6, 7, 2, 4, 5, 3, 8)$. In $\bar{P}$, the first three customers fill up plant 1, the next three fill up plant 2, and the last two are allocated to 3. Moreover, the order in which customers 1, 6 and 7 are allocated to plant 1 is also irrelevant, so they could be allocated at a time (in one single round *or* en bloc*). So, to obtain a cyclic-coalition stable allocation, it suffices to define a permutation giving the order in which the* plants *are filled up (in the sense that the customers in any block are all allocated to the same plant), followed by the actual allocation of customers. An example of this "permutation plus allocation" is $\hat{P} = (1, 2, 3)$, where the order is established only among plants (plant 1 fills up first, then plant 2, and finally plant 3), together with the allocations $(\{1, 6, 7\}, \{2, 4, 5\}, \{3, 8\})$. Note that $\hat{P}$ only results in a cyclic-coalition stable allocation provided that the first block of customers is allocated to their favorite plant out of the three available (which is 1), and the next block to their favorite plant in the set $\{2, 3\}$ (2 in this case).*

To further develop this idea, let us consider three different permutations of plants and the cyclic-coalition stable allocations that can be obtained. For $P_1 = (1, 2, 3)$, there are 4 customers $\{1, 6, 7, 8\}$ whose favorite plant is 1. If we allocate $\{1, 6, 7\}$ to plant 1, then the remaining customers are $\{2, 4, 5\}$ (who prefer plant 2 over plant 3) and $\{3, 8\}$ (who prefer plant 3 over plant 2), so allocating $\{1, 6, 7\}$ to plant 1 results in one cyclic-coalition stable allocation (the first one in Table 4, $\hat{P}$). If we still follow P_1 but allocate $\{1, 6, 8\}$ to plant 1, then there are 4 customers $\{2, 4, 5, 7\}$ whose favorite plant is 2, so we obtain the cyclic-coalition stable allocations $2 - 5$ in Table 4. For a different plant permutation $P_2 = (2, 1, 3)$, there are exactly three customers $\{2, 4, 5\}$ whose favorite plant is 2. Out of the remaining five, $\{1, 6, 7, 8\}$ prefer plant 1, so in this case we obtain allocations 1, 2, 6 and 10 in Table 4 (note that permutations P_1 and P_2 might lead to the same allocation). Lastly, there is no cyclic-coalition stable allocation following the order $P_3 = (3, 1, 2)$. This occurs because there is no set of 3 customers whose favorite plant is 3.

To finish, let us study the allocation where customers $\{1, 5, 6\}$ fill up plant 1, $\{2, 4, 7\}$ are allocated to plant 2 and $\{3, 8\}$ to plant 3. This allocation is customer stable because none of the customers of plants 1 or 2 would rather be allocated to plant 3 (which is the only plant with availability). However, it is not cyclic-coalition stable, since customers 5 and 7 would rather swap allocations. There is no permutation of customers giving rise to this allocation: if 5 chooses before 7, then when 5 is allocated, neither plant 1 nor 2 are full, so 5 is allocated to plant 1 (and vice versa). Likewise, there is no permutation of plants giving rise to this allocation.

Table 4: Cyclic-coalition stable allocations for the instance depicted in Table 3.

	j=1	j=2	j=3
i	167	245	38
	168	245	37
	168	247	35
	168	257	34
	168	457	23
	178	245	36
	678	124	35
	678	125	34
	678	145	23
	678	245	13

For the reduced formulation, we keep the customary sets of binary variables y_j, $j \in \mathcal{J}$ and x_{ij}, $i \in \mathcal{I}$, $j \in \mathcal{J}$ that determine the location and allocation. We also define a set of binary variables u_j^r, $j, r, \in \mathcal{J}$, equal to one if and only if plant j is the r-th plant to fill up. Using these variables, the formulation is:

$$
\begin{aligned}
\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CC}} \quad \min_{\boldsymbol{y},\boldsymbol{x},\boldsymbol{u}} \quad & \sum_{j\in\mathcal{J}} f_j y_j + \sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}} g_{ij} x_{ij} && \text{(10a)}\\
\text{s.t.} \quad & \sum_{j\in\mathcal{J}} x_{ij} = 1, \quad \forall i \in \mathcal{I}, && \text{(10b)}\\
& x_{ij} \le y_j, \quad \forall i \in \mathcal{I}, j \in \mathcal{J}, && \text{(10c)}\\
& \sum_{r\in\mathcal{J}} u_j^r \le y_j, \quad \forall j \in \mathcal{J}, && \text{(10d)}\\
& \sum_{j\in\mathcal{J}} u_j^1 \le 1, && \text{(10e)}\\
& \sum_{j\in\mathcal{J}} u_j^r \le \sum_{j\in\mathcal{J}} u_j^{r-1}, \quad \forall r \in \mathcal{J}, r > 1 && \text{(10f)}\\
& \sum_{i\in\mathcal{I}} x_{ij} \le (c_j - 1) y_j + \sum_{r\in\mathcal{J}} u_j^r, \quad \forall j \in \mathcal{J} && \text{(10g)}\\
& c_j \sum_{r\in\mathcal{J}} u_j^r \le \sum_{i\in\mathcal{I}} x_{ij}, \quad \forall j \in \mathcal{J} && \text{(10h)}\\
& x_{ij} + \sum_{r'=1}^{r-1} u_j^{r'} + \sum_{\substack{j'\in\mathcal{J}:\\ j' \prec_i j}} u_{j'}^r \le 1 + y_j, \quad \forall i \in \mathcal{I}, j, r \in \mathcal{J}, && \text{(10i)}\\
& y_j - \sum_{r\in\mathcal{J}} u_j^r \le \sum_{j' \preceq_i j} x_{ij'}, \quad \forall i \in \mathcal{I}, j \in \mathcal{J}, && \text{(10j)}\\
& x_{ij} \in \{0,1\}, \quad \forall i \in \mathcal{I}, j \in \mathcal{J}, && \text{(10k)}\\
& y_j, u_j^r \in \{0,1\}, \quad \forall j, r \in \mathcal{J}. && \text{(10l)}
\end{aligned}
$$

Constraints (10b) and (10c) guarantee that each customer is assigned to an open facility. Constraints (10d) impose that a facility can only be filled if it is open, while (10e) and (10f) enforce that facilities are filled sequentially, with only one being filled per round. Constraints (10g) and (10h) establish the capacity limitations. Moreover, (10i) prevent a facility assigned to a customer from being filled before another that the customer prefers, guaranteeing that there are no blocking pairs or blocking sets of customers. And (10j) require that if a facility is open and not full, then each customer must be served by a facility that is at least as preferred as that one, preventing blocking customers. Finally, constraints (10k) and (10l) specify that all decision variables are binary.

In this case, the reduction in the number of variables and constraints with respect to $\texttt{(CFLCP)}_{\texttt{CC}}$ is quite significant. Constraints (9c) are no longer required; the families of constraints defined $\forall r$ now have fewer constraints because $r \in \mathcal{J}$ instead of $\mathcal{I}$; and the capacity constraints (9g) and (9h) correspond now to the smaller families (10g) and (10h). The number of u-variables is reduced to only $|\mathcal{J}|^2$, and the number of x-variables is reduced from $|\mathcal{I}|^2 \cdot |\mathcal{J}|$ to $|\mathcal{I}| \cdot |\mathcal{J}|$. Furthermore, in the following proposition it is stated that the integrality constraints on the latter set of variables can be relaxed.

Proposition 8. *The integrality constraints on the x-variables (10k) can be relaxed in model $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CC}}$.*

Proof. For given integer values of the variables y and u, we consider the same partition of the set $\mathcal{J}$ into $\mathcal{J}_1$, $\mathcal{J}_2$ and $\mathcal{J}_3$ as in Proposition 3, where a plant is full if $\sum_{r \in \mathcal{J}} u_j^r = 1$.

With this partition, the structure of the model $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CC}}$ becomes equivalent to that of Proposition 3, except for the presence of the additional constraints (10i). These constraints consist of inequalities with a single nonzero coefficient equal to 1, and the right-hand side vector remains integral.

Adding such constraints to a totally unimodular matrix preserves total unimodularity. Hence, the full constraint matrix remains totally unimodular, and by (Wolsey and Nemhauser, 1999, Proposition 2.3), the integrality constraints (10k) can be relaxed. □

6.3. Formulation for a maximum Pareto optimal matching for the CHA

As an additional result, we note that formulation $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CC}}$ can be slightly modified to provide a Pareto optimal matching of maximum cardinality in an instance of CHA. In this setting, all plants are assumed to be open (we are only interested in the allocation problem) and the costs f_j can be omitted. In addition, we maximize the number of clients allocated, so the objective weights $g_{ij} := 1$ and constraints (10b) are relaxed. All in all, the model reads:

$$\texttt{(CHA)}^{\texttt{R}}_{\texttt{PO}} \quad \max_{\boldsymbol{x},\boldsymbol{u}} \quad \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} x_{ij} \tag{11a}$$

$$\text{s.t.} \quad (10b), (10e) - (10f), (10h) \tag{11b}$$

$$\sum_{r\in\mathcal{J}} u_j^r \leq 1, \quad \forall j \in \mathcal{J}, \tag{11c}$$

$$\sum_{i\in\mathcal{I}} x_{ij} \leq (c_j - 1) + \sum_{r\in\mathcal{J}} u_j^r, \quad \forall j \in \mathcal{J} \tag{11d}$$

$$x_{ij} + \sum_{r'=1}^{r-1} u_j^{r'} + \sum_{\substack{j'\in\mathcal{J}:\\ j'\prec_i j}} u_{j'}^r \leq 2, \quad \forall i \in \mathcal{I}, j, r \in \mathcal{J}, \tag{11e}$$

$$1 - \sum_{r\in\mathcal{J}} u_j^r \leq \sum_{j'\preceq_i j} x_{ij'}, \quad \forall i \in \mathcal{I}, j \in \mathcal{J}, \tag{11f}$$

$$x_{ij} \geq 0, \quad \forall i \in \mathcal{I}, j \in \mathcal{J}, \tag{11g}$$

$$u_j^r \in \{0,1\}, \quad \forall j, r \in \mathcal{J}. \tag{11h}$$

7. Computational experiments

In this section, we present the computational experiments carried out to evaluate and compare the proposed models. The objective is threefold: first, to compare the original formulations associated with each type of stability with their reduced counterparts; second, to analyze the impact of relaxing the integrality constraints in the reduced models; and third, to assess the performance quality of solutions of our models against those proposed in Calvete et al. (2020). All tests were performed using the commercial IP solver Xpress Mosel on a server equipped with an AMD Ryzen 9 7950X processor (3.4 GHz, 16 cores and 32 threads) and 4 × 48 GB of DDR5 RAM (196 GB in total). The system allows up to 16 single-threaded tasks to run in parallel, each with 16 GB of allocated memory. For storage, it uses 2 × 2 TB Western Digital Black SN770 SSDs configured in RAID 1 to ensure data reliability during execution.

7.1. *Comparison of each stable model with its reduced counterpart*

We begin with a comparison between the original models and their reduced counterparts for the different notions of stability. To this end, several instances were randomly generated, with the number of clients, number of products, and capacity values fixed in advance. For each combination of these parameters, five different instances were considered, and the values shown in the tables correspond to the average of the results obtained across these five instances. The instances can be found in: https://doi.org/10.5281/zenodo.19127602.

The instances were generated as follows: for the case of 50 clients, the number of facilities ranged between 5 and 10, and the capacity assigned to each facility was set between 12 and 20. For the case of 100 clients, the number of facilities varied among 5, 10, 15, and 20, with capacities ranging from 24 to 40. The cost of opening each facility was chosen as a random integer between 100 and 130, while the cost of assigning each client to each facility was selected as a random integer between 10 and 30. As for client preferences, a complete ranking over all facilities was generated randomly for each client.

Table 5: Results for the customer stable model $\texttt{(CFLCP)}_{\texttt{CS}}$ and its reduced counterpart $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CS}}$. The column Customers indicates the number of clients, Plants the number of facilities, and Capacity the identical capacity assigned to each facility. The columns Nodes, Time, LR Gap, and Gap refer respectively to the number of nodes explored in the branch-and-bound tree, the total running time (in seconds), the linear relaxation gap, and the optimality gap.

Customers	Plants	Capacity	$\texttt{(CFLCP)}_{\texttt{CS}}$				$\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CS}}$			
			Nodes	Time	LR Gap	Gap	Nodes	Time	LR Gap	Gap
50	5	12	14641.0	7.5	32.72	0.00	1.0	0.2	32.72	0.00
50	5	20	3980.6	5.2	23.04	0.00	88.6	0.4	23.04	0.00
50	10	12	74266.0	151	25.16	0.00	2337.8	2.6	25.16	0.00
50	10	20	13959.8	43.6	19.24	0.00	620.2	1.9	19.24	0.00
100	5	24	536979.0	484.8	39.45	0.00	5.8	0.5	39.45	0.00
100	5	40	86035.8	91.3	28.93	0.00	108.2	0.6	28.93	0.00
100	10	24	2474721.0	(2) 2850.4	34.74	3.68	4495.8	5.4	34.24	0.00
100	10	40	232802.8	307.5	27.38	0.00	1075.0	3.8	27.38	0.00
100	15	24	1475028.4	3600.0	32.49	11.28	35526.2	39.3	31.90	0.00
100	15	40	393874.6	1361.4	24.89	0.00	3197.0	11.0	24.90	0.00
100	20	24	629432.0	3600.0	32.99	15.71	168120.6	195.5	31.20	0.00
100	20	40	495892.0	(1) 3176.5	26.65	5.46	7481.0	26.9	26.46	0.00

For each type of stability, we present a table comparing the results obtained with the original model and its reduced version with continuous x-variables. Thus, the results for customer, pairwise, and cyclic-coalition stable models are depicted, respectively, in Tables 5–7. Each table includes the number of clients, the number of facilities, and the identical capacity assigned to each facility. These are followed by performance indicators obtained during the execution of the solver: the number of nodes explored in the branch-and-bound process; the total running time in seconds; the linear relaxation gap; and the final optimality gap. The time limit was set to one hour for all runs. In cases where no feasible integer solution was found within the time limit, the corresponding entries in the Gap column are marked with "–". The linear relaxation gap is computed as $100 \cdot \frac{UB-LRB}{LRB}\%$, and the optimality gap is computed as $100 \cdot \frac{UB-LB}{LB}\%$, where UB is the best integer solution value found within the time limit, LRB is the linear relaxation bound, and LB is the final lower bound reported by the solver. When only a subset of the five instances is solved to optimality, the number of solved instances is indicated in parentheses next to the running time. This detail is omitted when all five instances are solved (which is reflected by a zero optimality gap) or when none are solved (which is evident from the maximum time of 3600 seconds).

Regarding the comparison of models $\texttt{(CFLCP)}_{\texttt{CS}}$ and $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CS}}$, in Table 5 we observe that the reduced model successfully solves all instances, while the original model fails in some cases. Whenever the total running time reaches 3600 seconds, none of the five instances were solved to optimality. If the time limit is not reached but a nonzero gap remains, it indicates that at least one instance was not solved within the time limit. On average, the reduced model explores significantly fewer nodes and consistently solves the instances in under 4 minutes, unlike the original model, which struggles to do so in several configurations. It is also worth noting that the reduced model does not appear to be adversely affected by the seemingly large optimality gaps of the linear relaxation. Overall, these empirical results provide solid support for discarding the original formulation in favor of the reduced model, whose superior performance is evident across all tested instances.

Interestingly, both models yield the same linear relaxation bound at the root node. We notice this because the linear relaxation gaps are practically the same for both settings. This can be explained by the fact that any solution to the relaxation of the reduced model can be transformed into a feasible solution for the relaxation of the original model by suitably increasing the values of the auxiliary variables u_j up to their upper bounds—that is, so that constraint (4d) is satisfied with equality. Since these variables do not appear in the objective function, this adjustment has no effect on the objective value. As a result, all constraints in the original model are satisfied, and the same bound is preserved.

In general, when two of the three parameters (number of clients, number of facilities, and facility capacity) are fixed, the model tends to become more complex as the number of clients or facilities increases, due to the larger number of variables and constraints involved. Conversely, increasing the facility capacity tends to simplify the problem, as higher capacities reduce the combinatorial effort required to avoid exceeding them.

The performance of models $\texttt{(CFLCP)}_{\texttt{PW}}$ and $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{PW}}$ is depicted in Table 6. We observe that in this case neither model is able to solve all instances. In particular, neither of the two formulations manages to solve any of the instances with 100 clients, 15 or 20 facilities, and facility capacity equal to 24 within the time limit. However, the reduced model succeeds in solving the remaining instances, whereas the original formulation fails to solve several additional configurations.

A behavior similar to the one observed in the customer stable case arises here as well: the reduced model consistently explores fewer nodes and achieves shorter running times. Moreover, unlike in the previous case, the linear relaxation gaps differ between the two formulations, with the reduced model providing tighter gaps in several configurations. This additional advantage further supports the empirical preference for the reduced model, which outperforms the original formulation in terms of both time and solvability. These results justify the dismissal of the original model in favor of its reduced counterpart.

Regarding parameter influence, the same trend holds: the instances that combine a higher number

Table 6: Results for the pairwise stable model $\texttt{(CFLCP)}_{\texttt{PW}}$ and its reduced counterpart $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{PW}}$. The column Customers indicates the number of clients, Plants the number of facilities, and Capacity the identical capacity assigned to each facility. The columns Nodes, Time, LR Gap, and Gap refer respectively to the number of nodes explored in the branch-and-bound tree, the total running time (in seconds), the linear relaxation gap, and the optimality gap.

Customers	Plants	Capacity	$\texttt{(CFLCP)}_{\texttt{PW}}$				$\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{PW}}$			
			Nodes	Time	LR Gap	Gap	Nodes	Time	LR Gap	Gap
50	5	12	5158.6	7.8	28.50	0.00	51.8	0.3	26.95	0.00
50	5	20	158.2	3.0	22.68	0.00	104.6	0.3	22.37	0.00
50	10	12	68717.8	346.5	28.12	0.00	31979.8	16.3	27.44	0.00
50	10	20	1508.6	14.2	22.08	0.00	1344.2	3.1	22.06	0.00
100	5	24	67063.2	262.8	34.89	0.00	169.4	0.6	32.95	0.00
100	5	40	3309.9	21.9	28.98	0.00	196.6	0.6	28.93	0.00
100	10	24	127253.6	3600.0	41.62	21.1	246916.8	419.7	36.95	0.00
100	10	40	18346.6	184.4	28.1	0.00	2735.0	12.0	27.46	0.00
100	15	24	23150.2	3600.0	48.68	37.84	409771.6	3600.0	36.27	16.3
100	15	40	41507.6	(1) 3454.0	29.76	16.42	9162.6	83.3	28.38	0.00
100	20	24	6583.4	3600.0	(3) 52.38	(3) 45.12	139989.4	3600.0	36.27	23.3
100	20	40	6325.4	3600.0	35.43	29.34	14292.2	555.8	28.29	0.00

Table 7: Results for the cyclic-coalition stable model $\texttt{(CFLCP)}_{\texttt{CC}}$ and its reduced counterpart $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CC}}$. The column Customers indicates the number of clients, Plants the number of facilities, and Capacity the identical capacity assigned to each facility. The columns Nodes, Time, LR Gap, and Gap refer respectively to the number of nodes explored in the branch-and-bound tree, the total running time (in seconds), the linear relaxation gap, and the optimality gap.

Customers	Plants	Capacity	$\texttt{(CFLCP)}_{\texttt{CC}}$				$\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CC}}$			
			Nodes	Time	LR Gap	Gap	Nodes	Time	LR Gap	Gap
50	5	12	10926.4	3600.0	32.36	16.22	35.8	0.9	32.57	0.00
50	5	20	22458	3600.0	22.89	12.54	98.6	1.1	23.04	0.00
50	10	12	23.2	3600.0	30.56	27.76	19539.8	58.1	28.74	0.00
50	10	20	304.8	3600.0	22.01	17.18	985.4	9.3	22.07	0.00
100	5	24	3.0	3600.0	–	–	145	2.1	39.07	0.00
100	5	40	364.8	3600.0	30.72	24.51	141.4	1.4	28.93	0.00
100	10	24	0.0	3600.0	–	–	24029.4	166.9	37.80	0.00
100	10	40	0.0	3600.0	(1) 31.66	(1) 31.53	1199.8	24.3	27.46	0.00
100	15	24	0.0	3600.0	–	–	157516.4	(4) 2290.5	35.81	1.18
100	15	40	0.0	3600.0	–	–	3325.6	167.7	28.38	0.00
100	20	24	0.0	3600.0	–	–	230371.6	(1) 3484.5	35.44	10.86
100	20	40	0.0	3600.0	–	–	5187.0	617.1	28.29	0.00

of clients and facilities with lower facility capacity are the most computationally demanding. In fact, the hardest instances (those unsolved by both models) correspond to the most extreme parameter combinations within our set of instances, or to configurations that are nearly as extreme.

A detailed comparison between models $\texttt{(CFLCP)}_{\texttt{CC}}$ and $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CC}}$ is presented in Table 7. In this case, the initial model is unable to solve any of the generated instances. Moreover, it only finds feasible solutions for a subset of them, and with very large gaps: the symbol "–" in columns LR Gap and Gap indicates that no feasible solution was found. In contrast, the reduced model manages to solve almost all the provided instances, except for two specific cases in which it fails to solve some of them. Moreover, the reduced model consistently exhibits a much lower number of explored nodes and significantly shorter running times compared to the original formulation. This reinforces the pattern already observed in the other stability variants, further highlighting the computational advantages of the reduced approach in terms of both efficiency and reliability. In this case, the improvement is more significant than in previous cases because there is a noticeable reduction in the number of variables: model $\texttt{(CFLCP)}_{\texttt{PW}}$ makes use of binary variables x^r_{ij} (a total of $|\mathcal{I}|^2 \cdot |\mathcal{J}|$) to account for customers' decision choice, while model $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{PW}}$ uses the much smaller set of continuous variables x_{ij} (only $|\mathcal{I}| \cdot |\mathcal{J}|$).

A similar effect is observed regarding the influence of each parameter. When two of the three parameters (number of clients, number of facilities, and facility capacity) are fixed, increasing the number of clients or facilities tends to make the problem more complex, as it enlarges the solution space and the number of constraints to manage. In contrast, increasing the facility capacity generally eases the computational burden, since higher capacities reduce the number of combinations that must be evaluated to ensure feasibility.

Finally, note that more realistic configurations where the list of preferences is incomplete would be easier to tackle for the models. In fact, these instances are naturally sparse and, since every customer needs to be assigned to a plant, they force some sets of plants to open and reduce the combinatorics. These advantages allow to solve instances with a larger number of customers and plants within the same time limit. However, we decided to test only instances with a complete list of preferences because our interest focuses on the comparison between the two formulations presented for each setting.

From an algorithmic perspective, several promising research directions emerge from this work. In particular, the development of decomposition-based approaches, such as Benders decomposition, appears especially appealing given the structural separation between location and allocation decisions induced by the stability constraints and the added binary variables. Likewise, the design of dedicated heuristic or metaheuristic procedures could provide scalable solution strategies for larger instances and deliver high-quality approximate solutions for the largest instances considered in the computational study. Moreover, each of the three proposed notions of stability leads to a distinct mathematical structure; therefore, any exact or heuristic algorithm would need to be specifically designed for the corresponding stability concept rather than directly transferred from one variant to another. Investigating and comparing such tailored algorithmic frameworks constitutes a natural and relevant avenue for future research. A more detailed discussion of these directions is provided in Section 8.

7.2. Comparison of the impact of variable relaxation on the reduced models.

Propositions 3, 5 and 8 state that the integrality constraints on the allocation variables x can be relaxed in models $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CS}}$, $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{PW}}$ and $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CC}}$, respectively. However, for certain integer linear models such a relaxation does not lead to an improvement in computational performance. From a practical standpoint, it may therefore be preferable to retain these variables as integer-valued. For this reason, we conduct a computational comparison of the reduced formulations under the three stability settings, both with and without relaxing the integrality of the x-variables.

Table 8 reports a comparison between the three reduced non-relaxed models and their counterparts in which the x-variables are relaxed, considering the same instances analyzed in the previous subsection. For

Table 8: Computational comparison of formulations $\texttt{(CFLCP)}_{\texttt{CS}}$, $\texttt{(CFLCP)}_{\texttt{PW}}$, and $\texttt{(CFLCP)}_{\texttt{CC}}$ under relaxed $\boldsymbol{x}$ and binary $\boldsymbol{x}$ configurations. Customers and Plants denote the number of clients and candidate facilities, respectively, and Capacity is the identical capacity assigned to each facility. For each formulation, Time (seconds) and Gap represent the total running time and the final optimality gap.

Customers	Plants	Capacity	$\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CS}}$				$\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{PW}}$				$\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CC}}$			
			Relaxed $\boldsymbol{x}$		Binary $\boldsymbol{x}$		Relaxed $\boldsymbol{x}$		Binary $\boldsymbol{x}$		Relaxed $\boldsymbol{x}$		Binary $\boldsymbol{x}$	
			Time	Gap	Time	Gap	Time	Gap	Time	Gap	Time	Gap	Time	Gap
50	5	12	0.2	0.00	0.1	0.00	0.3	0.00	0.4	0.00	0.9	0.00	2.3	0.00
50	5	20	0.4	0.00	0.4	0.00	0.3	0.00	0.7	0.00	1.1	0.00	1.4	0.00
50	10	12	2.6	0.00	2.4	0.00	16.3	0.00	35.1	0.00	58.1	0.00	960.0	0.00
50	10	20	1.9	0.00	1.5	0.00	3.1	0.00	3.2	0.00	9.3	0.00	35.8	0.00
100	5	12	0.5	0.00	0.5	0.00	0.6	0.00	1.3	0.00	2.1	0.00	46.9	0.00
100	5	20	0.6	0.00	0.7	0.00	0.6	0.00	1.6	0.00	1.4	0.00	9.0	0.00
100	10	24	5.4	0.00	10.3	0.00	419.7	0.00	92.8	0.00	166.9	0.00	3600.0	18.33
100	10	40	3.8	0.00	5.3	0.00	12.0	0.00	18.3	0.00	24.3	0.00	180.9	0.00
100	15	24	39.3	0.00	105.0	0.00	3600.0	16.30	(2) 2825.6	8.38	(4) 2290.5	1.18	3600.0	32.51
100	15	40	11.0	0.00	41.0	0.00	83.3	0.00	146.4	0.00	167.7	0.00	(3) 2368.1	10.52
100	20	24	195.5	0.00	656.6	0.00	3600.0	23.35	3600.0	26.43	(1) 3484.5	10.86	3600.0	78.62
100	20	40	26.9	0.00	137.6	0.00	555.8	0.00	(4) 2234.3	3.73	617.1	0.00	3600.0	30.09

each setting, the table provides the total computational time (Time, in seconds, with a time limit of 3600 seconds) and the final optimality gap (Gap). When only a subset of the five instances is solved within the time limit, the number of solved instances is indicated in parentheses next to the reported running time.

For model $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CS}}$, relaxing the x-variables leads to improved solution times for the most challenging instances (those with 100 customers and 15 or 20 plants). The remaining instances are relatively easy for both formulations, exhibiting comparable running times, and all are solved to optimality within the time limit.

A similar comparison for model $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{PW}}$, shown in the lower part of the table, indicates that the relaxed formulation achieves better solution times and smaller optimality gaps across almost all instances, with the exception of those involving 100 customers, 15 plants, and capacity equal to 24.

For model $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CC}}$, the computational benefit of relaxing the variables is even more pronounced, as reflected in both the final optimality gaps and the running times. In particular, among the last 20 instances, only three are solved to optimality in the non-relaxed formulation, whereas fifteen reach optimality within one hour when the variables are relaxed. Moreover, for the instances that remain unsolved, the relaxed formulation consistently yields significantly smaller optimality gaps.

All in all, relaxing the variables results in an improvement of the performance in the three settings.

7.3. Comparison of the solution quality provided by each stable setting

In this section, we conduct a computational experiment using the instances provided in Calvete et al. (2020), where model (1) was used. Our first goal is to analyze the impact of the three stability assumptions on the optimal values obtained by the models. Secondly, we aim to further assess the computational performance of our stable models in terms of running times, gaps, number of nodes, etc., using larger instances. Additionally, we report the optimal solutions given by model (1) found in Calvete et al. (2020) solely to compare the quality of their cyclic-coalition stable solutions with those provided by our reduced cyclic-coalition stable model $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CC}}$.

We consider the original dataset of 71 instances (P_1-P_{71}) in our study. The procedure for generating the instances is detailed in that work. The capacity of each plant, as well as the cost and preference values, are randomly generated, and the structural parameters of the instances (namely, the number of clients and the number of plants) are fixed and known. The number of customers ranges from 50 to 200, while the number of candidate plants ranges from 10 to 30. In our experiments, we set a time limit of 2 hours per instance.

Table 9: Results for the reduced models $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CS}}$, $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{PW}}$, and $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CC}}$, compared with the original model (CGICC) proposed by Calvete et al. (2020). The columns Set and Instance provide the dataset group and identifier; C and P denote the number of customers and candidate facilities, respectively. For each reduced model, the columns Obj, Time, and Gap report the best objective value found, total execution time in seconds, and the final optimality gap. For the (CGICC) model, Obj represents the objective value reported in the original study. The final column, Relative Diff. (%), represents the percentage change between the objective values of $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CC}}$ and (CGICC).

Set	Instance	C	P	$\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CS}}$			$\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{PW}}$			$\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CC}}$			(CGICC)	
				Obj	Time	Gap	Obj	Time	Gap	Obj	Time	Gap	Obj	Relative Diff. (%)
S_1	P_1	50	10	10016	0.6	0.00	15017	2.1	0.00	15335	21.8	0.00	18592	-17.52
	P_2	50	10	9327	0.6	0.00	14404	1.8	0.00	14598	26.3	0.00	17658	-17.33
	P_3	50	10	10527	0.8	0.00	15604	3.2	0.00	15798	29.6	0.00	19058	-17.11
	P_4	50	10	11727	0.7	0.00	16736	3.2	0.00	16998	26.5	0.00	20442	-16.85
	P_5	50	10	13832	0.7	0.00	15347	21.0	0.00	16050	60.5	0.00	18552	-13.49
	P_6	50	10	12924	0.6	0.00	14601	22.2	0.00	15304	100.7	0.00	17806	-14.05
	P_7	50	10	14324	0.6	0.00	16001	20.0	0.00	16704	109.5	0.00	19206	-13.03
	P_8	50	10	15724	0.5	0.00	17401	23.1	0.00	18104	112.0	0.00	20606	-12.14
	P_9	50	10	14072	1.2	0.00	15141	4.7	0.00	15717	29.1	0.00	17651	-10.96
	P_{10}	50	10	13616	1.3	0.00	14685	6.7	0.00	15261	23.6	0.00	17146	-10.99
	P_{11}	50	10	14616	1.2	0.00	15685	6.3	0.00	16261	23.9	0.00	18146	-10.39
	P_{12}	50	10	15616	1.3	0.00	16685	3.1	0.00	17261	29.3	0.00	19146	-9.85
S_2	P_{13}	50	20	8696	1.2	0.00	13990	3706.9	0.00	15202	7200.0	30.18	17745	-14.33
	P_{14}	50	20	7922	1.5	0.00	13210	2189.7	0.00	14508	7200.0	33.67	16720	-13.23
	P_{15}	50	20	9322	1.3	0.00	14410	3186.3	0.00	15849	7200.0	27.72	18120	-12.53
	P_{16}	50	20	10717	1.3	0.00	15610	2296.1	0.00	16719	7200.0	19.09	19427	-13.94

(continued on next page)

Table 9: Results for the reduced models $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CS}}$, $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{PW}}$, and $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CC}}$, compared with the original model proposed by Calvete et al. (2020), referred to as `(CGICC)` (continued).

Set	Instance	C	P	$\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CS}}$			$\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{PW}}$			$\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CC}}$			`(CGICC)`	
				Obj	Time	Gap	Obj	Time	Gap	Obj	Time	Gap	Obj	Relative Diff. (%)
	P_{17}	50	20	13511	52.0	0.00	15302	7200.0	10.50	15384	7200.0	6.61	17613	-12.66
	P_{18}	50	20	12554	56.1	0.00	14173	7200.0	7.48	14292	6582.0	0.00	16718	-14.51
	P_{19}	50	20	13954	56.1	0.00	15692	7200.0	10.33	15692	6541.6	0.00	18118	-13.39
	P_{20}	50	20	15354	59.8	0.00	17057	7200.0	10.88	17139	7200.0	8.90	19518	-12.19
	P_{21}	50	20	13442	31.1	0.00	15027	5319.6	0.00	15263	2096.9	0.00	17253	-11.53
	P_{22}	50	20	12433	25.1	0.00	14024	2095.9	0.00	14099	1228.5	0.00	16407	-14.07
	P_{23}	50	20	13633	29.2	0.00	15224	5625.5	0.00	15299	1495.7	0.00	17607	-13.11
	P_{24}	50	20	14833	30.9	0.00	16424	5181.4	0.00	16499	1799.4	0.00	18807	-12.27
S_3	P_{25}	150	30	12167	4.8	0.00	29399	7200.0	66.45	37739	7200.0	218.46	40164	-6.04
	P_{26}	150	30	11255	3.7	0.00	31106	7200.0	84.73	43700	7200.0	296.31	39266	11.29
	P_{27}	150	30	12655	3.8	0.00	31511	7200.0	75.14	35753	7200.0	184.94	40866	-12.51
	P_{28}	150	30	14055	2.6	0.00	33668	7200.0	73.03	54776	7200.0	297.42	42466	28.99
	P_{29}	150	30	28662	7200.0	75.60	41783	7200.0	144.29	54408	7200.0	345.25	44715	21.68
	P_{30}	150	30	26929	7200.0	86.34	39616	7200.0	152.91	42898	7200.0	288.32	43480	-1.34
	P_{31}	150	30	28488	7200.0	74.83	42275	7200.0	142.98	47116	7200.0	263.83	45480	3.60
	P_{32}	150	30	31310	7200.0	71.95	42601	7200.0	117.15	55697	7200.0	273.00	47480	17.31
	P_{33}	150	30	32822	7200.0	74.49	48189	7200.0	190.54	51138	7200.0	330.34	40428	26.49
	P_{34}	150	30	31756	7200.0	79.40	38510	7200.0	146.83	51711	7200.0	362.59	39517	30.86
	P_{35}	150	30	33507	7200.0	73.83	43153	7200.0	153.84	40401	7200.0	175.87	41117	-1.74
	P_{36}	150	30	34956	7200.0	69.73	45194	7200.0	140.51	53752	7200.0	283.21	42717	25.83
	P_{37}	150	30	24238	135.0	0.00	30234	7200.0	70.94	30252	7200.0	144.26	33134	-8.70
	P_{38}	150	30	23655	161.5	0.00	32377	7200.0	87.70	29604	7200.0	156.94	32486	-8.87
	P_{39}	150	30	24655	171.8	0.00	30651	7200.0	69.42	32611	7200.0	153.28	33486	-2.61
	P_{40}	150	30	25655	139.7	0.00	31641	7200.0	66.19	33754	7200.0	148.38	34486	-2.12
S_4	P_{41}	90	10	7605	1.8	0.00	9805	6.1	0.00	9805	19.0	0.00	11574	-15.28
	P_{42}	80	20	6113	4.6	0.00	8102	53.2	0.00	8102	7200.0	14.20	9708	-16.54
	P_{43}	70	30	5675	5.2	0.00	7110	1251.1	0.00	7456	7200.0	33.07	8637	-13.67
	P_{44}	90	10	9193	1.6	0.00	12658	8.3	0.00	12716	92.5	0.00	16426	-22.59
	P_{45}	80	20	7056	4.1	0.00	10773	915.3	0.00	11425	7200.0	29.07	12514	-8.70
	P_{46}	70	30	6394	5.4	0.00	8979	6835.1	0.00	9813	7200.0	62.74	10741	-8.64
	P_{47}	90	10	7673	1.5	0.00	11527	4.5	0.00	11620	58.3	0.00	13534	-14.14
	P_{48}	80	20	6493	2.3	0.00	9618	327.6	0.00	9785	2102.0	0.00	11070	-11.61
	P_{49}	70	30	5690	3.7	0.00	8009	3080.8	0.00	8975	7200.0	59.85	9175	-2.18
	P_{50}	100	10	9438	1.9	0.00	14127	14.3	0.00	14468	96.0	0.00	16749	-13.62
	P_{51}	100	20	7912	2.9	0.00	12685	2454.9	0.00	13362	7200.0	68.09	15510	-13.85
	P_{52}	100	10	10714	1.5	0.00	17807	13.1	0.00	17860	121.4	0.00	21872	-18.34
	P_{53}	100	20	9833	2.4	0.00	16244	6747.4	0.00	17295	7200.0	79.61	20358	-15.05
	P_{54}	100	10	13446	1.7	0.00	16220	23.8	0.00	16220	43.3	0.00	19114	-15.14
	P_{55}	100	20	8152	3.4	0.00	13006	522.7	0.00	14031	7200.0	78.43	17405	-19.39
S_5	P_{56}	200	30	35546	7200.0	58.86	60225	7200.0	150.31	83298	7200.0	290.29	68082	22.35
	P_{57}	200	30	40141	7200.0	50.88	64484	7200.0	130.82	77800	7200.0	203.83	72582	7.19

(continued on next page)

Table 9: Results for the reduced models $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CS}}$, $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{PW}}$, and $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CC}}$, compared with the original model proposed by Calvete et al. (2020), referred to as (CGICC) (continued).

Set	Instance	C	P	$\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CS}}$			$\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{PW}}$			$\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CC}}$			(CGICC)	
				Obj	Time	Gap	Obj	Time	Gap	Obj	Time	Gap	Obj	Relative Diff. (%)
	P_{58}	200	30	50598	7200.0	38.22	72880	7200.0	90.40	–	7200.0	–	83082	–
	P_{59}	200	30	42046	7200.0	48.62	65003	7200.0	113.88	–	7200.0	–	74524	–
	P_{60}	200	30	22630	12.4	0.00	54441	7200.0	121.48	63172	7200.0	181.56	62434	1.18
	P_{61}	200	30	25630	9.1	0.00	56715	7200.0	106.79	79486	7200.0	210.65	65434	21.48
	P_{62}	200	30	32630	9.4	0.00	62006	7200.0	78.69	82643	7200.0	153.47	72434	14.09
	P_{63}	200	30	25629	1.8	0.00	59363	7200.0	103.23	77391	7200.0	202.76	66192	16.92
	P_{64}	200	30	42123	7200.0	57.19	60016	7200.0	127.16	57923	7200.0	140.62	61953	-6.50
	P_{65}	200	30	44431	7200.0	52.69	66577	7200.0	137.63	65711	7200.0	150.10	64353	2.11
	P_{66}	200	30	50858	7200.0	48.56	66274	7200.0	94.72	67581	7200.0	114.34	69953	-3.39
	P_{67}	200	30	46240	7200.0	50.61	68958	7200.0	133.23	67252	7200.0	150.20	64962	3.53
	P_{68}	200	30	22710	11.3	0.00	53885	7200.0	113.64	60871	7200.0	170.96	63798	-4.59
	P_{69}	200	30	25857	14.0	0.00	58731	7200.0	108.23	58891	7200.0	127.72	66870	-11.93
	P_{70}	200	30	32726	8.9	0.00	67360	7200.0	93.71	69468	7200.0	114.12	73870	-5.96
	P_{71}	200	30	26448	10.5	0.00	59238	7200.0	101.77	65626	7200.0	156.64	67801	-3.21

The performance of the reduced models $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CS}}$, and $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{PW}}$ is summarized in Table 9. For each reduced model, we include the objective value, the computational time (in seconds), and the final optimality gap in the columns Obj, Time and Gap, respectively.

Among the three models, we observe that model $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CS}}$ provides the best results in terms of time, gap, and overall performance. In particular, it is able to solve all the instances P_1-P_{24} (with 50 customers and 10 or 20 candidate facilities) within the imposed time limit of two hours. In contrast, model $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{PW}}$ solves 15 of the first 16 instances, while $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CC}}$ successfully solves only the first 12. This observation is also evident for instances P_{60}-P_{63}, P_{68}-P_{71}, which are solved by the model in a few seconds but remain unsolved for the other two stable models.

The comparison between models $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{PW}}$ and $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CC}}$ is not straightforward, but we can see that for instances of intermediate size, such as those in set S_4, model $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{PW}}$ solves all the proposed instances, while model $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CC}}$ fails to solve several of them.

It is worth highlighting a clear pattern in the results: as the stability concept becomes more restrictive (from customer stability to pairwise stability, and finally to cyclic-coalition stability), the objective function value systematically increases. This behavior confirms that blocking structures involving more than two agents do appear in practice, and they are effectively eliminated only under the strongest stability notion. Hence, each increase in the restrictiveness of the model translates into higher total cost, as the solution space becomes progressively constrained. It is also noticeable that forbidding blocking pairs results in a greater relative increase in the objective value (with respect to forbidding only blocking customers) than forbidding blocking sets (with respect to the pairwise stable setting). This is probably due to the fact that there are far more blocking pairs than blocking sets in the instances, since the greater the number of customers in the set, the rarer such sets become.

Among these, model $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CC}}$ deserves particular attention due to its conceptual similarity to model (1). Both aim to deliver allocations that are globally satisfactory for all clients (Pareto optimal, attending to the definition given in Manlove (2013)); however, while $\texttt{(CGICC)}$ identifies a subset of Pareto-optimal solutions, $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CC}}$ explores the full space of cyclic-coalition stable allocations. To see whether it has an impact on the objective value, we compare the cyclic-coalition stable optimal values or incumbent solutions provided by model $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CC}}$ with the optimal values given in Calvete et al. (2020). To better illustrate this comparison, the column (CGICC) in Table 9 reports the optimal solution found by model (1) in Calvete et al. (2020), and the last column of the table reports the relative percentage difference of the two objective values. Negative values in this column indicate that our model achieved a lower total cost.

When we obtain optimal solutions using model $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CC}}$, all of them improve upon those of the original model. Even in the instances where optimality is not reached, in most cases $\texttt{(CFLCP)}^{\texttt{R}}_{\texttt{CC}}$ obtains strictly better objective values, failing only for the largest and most difficult instances. These results reinforce the idea that the cyclic-coalition stable formulation is less restrictive than model (1), allowing for a broader set of feasible allocations. As a consequence, it can potentially capture more efficient configurations that still satisfy a meaningful notion of stability, although this may come at the cost of greater computational effort. Importantly, in the few instances where our model yields a worse objective value, this is necessarily due to the fact that optimality was not reached within the time limit. Since any feasible solution to the original model is also feasible for ours, no degradation in solution quality can occur unless the solver is interrupted before convergence.

It is also worth noting that the original model $\texttt{(CGICC)}$ demonstrates robustness, as it provides optimal solutions for all tested instances under its own formulation. The solution reported in (CGICC) is also a feasible solution in our setting, so when the model provides better feasible solutions they should be preferred over those given by our cyclic-coalition reduced model. However, this model imposes a very restrictive notion of stability, which considerably narrows the solution space and may exclude other desirable allocations that remain stable under broader definitions. All in all, the reduced models introduced in this work not only yield improved objective values, but also offer the flexibility to accommodate varying degrees of stability, allowing the decision-maker to balance efficiency and fairness according to the specific context.

8. Conclusions and future research

In this work, we have provided new definitions of stability and derived associated mixed-integer linear optimization models to solve the CFLCP in the global preference maximization framework.

A key contribution of this work is the introduction of three alternative stability notions—customer, pairwise and cyclic-coalition stability—together with corresponding mixed-integer linear formulations that allow the CFLCP to be solved without relying on numerical preference values. These formulations provide a hierarchy of increasingly restrictive stability concepts, each capturing different behavioral assumptions on how customers may improve their allocations. From a modeling perspective, this framework offers a flexible way to balance efficiency and fairness, as the choice of stability notion directly determines the structure of the feasible set.

From a computational standpoint, the reduced formulations clearly outperform their original counterparts across all stability settings. In particular, they significantly reduce the number of binary variables and constraints, leading to substantial improvements in running times and solvability. This effect is especially pronounced in the cyclic-coalition stable setting, where the original formulation becomes computationally intractable, while the reduced model is able to solve most instances to optimality within the time limit. Moreover, relaxing the integrality of the allocation variables further enhances performance, especially for the most challenging instances.

Regarding solution quality, an important takeaway is that stronger stability requirements lead to higher objective values. Customer stable solutions provide the lowest costs but allow for simple improvements by individual customers, while pairwise and cyclic-coalition stable solutions progressively eliminate more complex blocking structures at the expense of increased cost. In particular, cyclic-coalition stability ensures Pareto optimality, offering the strongest fairness guarantees, but at a significantly higher computational cost. This highlights a clear trade-off between efficiency and stability that practitioners must consider when selecting the appropriate formulation.

From an algorithmic perspective, several promising research directions emerge from this work. In particular, the development of decomposition-based approaches, such as Benders decomposition, appears especially appealing given the structural separation between location and allocation decisions induced by the stability constraints and the added binary variables. Likewise, the design of dedicated heuristic or metaheuristic procedures could provide scalable solution strategies for larger instances and deliver high-quality approximate solutions for the largest instances considered in the computational study. Moreover, each of the three proposed notions of stability leads to a distinct mathematical structure; therefore, any exact or heuristic algorithm would need to be specifically designed for the corresponding stability concept rather than directly transferred from one variant to another. Investigating and comparing such tailored algorithmic frameworks constitutes a natural and relevant avenue for future research.

Furthermore, the cyclic-coalition stable setting poses several challenges that we plan to address in future research. In particular, when ties arise in the customers' preference lists, the models proposed in this article are no longer directly applicable. In such cases, new formulations are required that incorporate additional structural properties and draw on results from the matching theory literature. Exploring these extensions would not only allow for a better understanding of stability under customer preferences with indifference, but could also lead to novel algorithms capable of handling more realistic preference structures.

In the deterministic version of the problem solved in this article, it is assumed that all customers can be allocated *simultaneously* in the most favorable way for the decision maker. This corresponds to an *optimistic* solution. However, in many practical applications of this and related problems (see, e.g., Hassin and Puerto, 2024), customers arrive sequentially. This motivates considering extensions of the model that yield more robust solutions. A *pessimistic approach*, for instance, determines the optimal plant locations under the assumption that customers are allocated in the most costly way, leading to a bilevel min–max formulation. Other intermediate approaches could also be explored, such as computing mean-cost or median-cost allocations when customer arrivals are assumed to be random.

Acknowledgements

All the authors acknowledge financial support through the grant RED2022-134149-T funded by MICIU/AEI/10.13039/501100011033 (Thematic Network on Location Science and Related Problems).

References

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

Büsing, C., Gersing, T., Wrede, S., 2022. Analysing the complexity of facility location problems with capacities, revenues, and closest assignments, in: INOC, pp. 1–6.

Büsing, C., Leitner, M., Wrede, S., 2025. Cover-based inequalities for the single-source capacitated facility location problem with customer preferences. Computers and Operations Research 182.

Cabezas, X., García, S., 2023. A semi-lagrangian relaxation heuristic algorithm for the simple plant location problem with order. Journal of the Operational Research Society 74, 2391–2402.

Calvete, H.I., Galé, C., Iranzo, J.A., Camacho-Vallejo, J.F., Casas-Ramírez, M.S., 2020. A matheuristic for solving the bilevel approach of the facility location problem with cardinality constraints and preferences. Computers & Operations Research 124, 105066.

Cánovas, L., García, S., Labbé, M., Marín, A., 2007. A strengthened formulation for the simple plant location problem with order. Operations Research Letters 35, 141–150.

Casas-Ramírez, M.S., Camacho-Vallejo, J.F., Martínez-Salazar, I.A., 2018. Approximating solutions to a bilevel capacitated facility location problem with customer's patronization toward a list of preferences. Applied Mathematics and Computation 319, 369–386.

Delorme, M., García, S., Gondzio, J., Kalcsics, J., Manlove, D., Pettersson, W., 2019. Mathematical models for stable matching problems with ties and incomplete lists. European Journal of Operational Research 277, 426–441.

Espejo, I., Marín, A., Rodríguez-Chía, A.M., 2012. Closest assignment constraints in discrete location problems. European Journal of Operational Research 219, 49–58.

Fernández, E., Landete, M., 2015. Fixed-charge facility location problems, in: 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.

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.

Hanjoul, P., Peeters, D., 1987. A facility location problem with clients' preference orderings. Regional Science and Urban Economics 17, 451–473.

Hansen, P., Kochetov, Y., Mladenovic, N., 2004. Lower bounds for the uncapacitated facility location problem with user preferences. Groupe d'études et de recherche en analyse des décisions, HEC Montréal. , 6.

Hassin, R., Puerto, J., 2024. Pricing heterogeneous products to heterogeneous customers who buy sequentially. Annals of Operations Research 340, 863–890.

Kwanashie, A., Manlove, D.F., 2014. An integer programming approach to the hospitals/residents problem with ties, in: Operations Research Proceedings 2013: Selected Papers of the International Conference on Operations Research, OR2013, organized by the German Operations Research Society (GOR), the Dutch Society of Operations Research (NGB) and Erasmus University Rotterdam, September 3-6, 2013, Springer. pp. 263–269.

Manlove, D., 2013. Algorithmics of matching under preferences. volume 2. World Scientific.

Polino, S., Camacho-Vallejo, J.F., Villegas, J.G., 2024. A facility location problem for extracurricular workshop planning: bi-level model and metaheuristics. International Transactions in Operational Research 31, 4025–4067.

Sng, C.T.S., 2008. Efficient algorithms for bipartite matching problems with preferences. Ph.D. thesis. University of Glasgow.

Vasilyev, I., Klimentova, X., Boccia, M., 2013. Polyhedral study of simple plant location problem with order. Operations Research Letters 41, 153–158.

Vasilyev, I.L., Klimentova, K., 2010. The branch and cut method for the facility location problem with client's preferences. Journal of Applied and Industrial Mathematics 4, 441–454.

Vasil'ev, I.L., Klimentova, K.B., Kochetov, Y.A., 2009. New lower bounds for the facility location problem with clients' preferences. Computational Mathematics and Mathematical Physics 49, 1010–1020.

Wolsey, L.A., Nemhauser, G.L., 1999. Integer and combinatorial optimization. John Wiley & Sons.