Université de Montréal

Efficient Reformulations for Deterministic and Choice-Based Network Design Problems

par

Robin Legault

Département d'informatique et de recherche opérationnelle
Faculté des arts et des sciences

Mémoire présenté en vue de l'obtention du grade de
Maître ès sciences (M.Sc.)
en Informatique

Orientation recherche opérationnelle

28 août 2023

Université de Montréal

Faculté des arts et des sciences

Ce mémoire intitulé

Efficient Reformulations for Deterministic and Choice-Based Network Design Problems

présenté par

Robin Legault

a été évalué par un jury composé des personnes suivantes :

Nadia El-Mabrouk
(président-rapporteur)

Emma Frejinger
(directeur de recherche)

Jean-François Côté
(codirecteur)

Margarida Carvalho
(membre du jury)

Résumé

La conception de réseaux est un riche sous-domaine de l'optimisation combinatoire ayant de nombreuses applications pratiques. Du point de vue méthodologique, la plupart des problèmes de cette classe sont notoirement difficiles en raison de leur nature combinatoire et de l'interdépendance des décisions qu'ils impliquent. Ce mémoire aborde deux problèmes de conception de réseaux dont les structures respectives posent des défis bien distincts. Tout d'abord, nous examinons un problème déterministe dans lequel un client doit acquérir au prix minimum un certain nombre d'unités d'un produit auprès d'un ensemble de fournisseurs proposant différents coûts fixes et unitaires, et dont les stocks sont limités. Ensuite, nous étudions un problème probabiliste dans lequel une entreprise entrant sur un marché existant cherche, en ouvrant un certain nombre d'installations parmi un ensemble de sites disponibles, à maximiser sa part espérée d'un marché composé de clients maximisant une fonction d'utilité aléatoire. Ces deux problèmes, soit le problème de transport à coût fixe à un puits et le problème d'emplacement d'installations compétitif basé sur les choix, sont étroitement liés au problème du sac à dos et au problème de couverture maximale, respectivement. Nous introduisons de nouvelles reformulations prenant avantage de ces connexions avec des problèmes classiques d'optimisation combinatoire. Dans les deux cas, nous exploitons ces reformulations pour démontrer de nouvelles propriétés théoriques et développer des méthodes de résolution efficaces. Notre nouvel algorithme pour le problème de transport à coûts fixes à un puits domine les meilleurs algorithmes de la littérature, réduisant le temps de résolution des instances de grande taille jusqu'à quatre ordres de grandeur. Une autre contribution notable de ce mémoire est la démonstration que la fonction objectif du problème d'emplacement d'installations compétitif basé sur les choix est sous-modulaire sous n'importe quel modèle de maximisation d'utilité aléatoire. Notre méthode de résolution basée sur la simulation exploite cette propriété et améliore l'état de l'art pour plusieurs groupes d'instances.

Mots clés : Conception de réseaux, Optimisation combinatoire, Optimisation basée sur les choix, Emplacement d'installations, Problème de transport à coûts fixes, Problème du sac à dos, Relaxation lagrangienne, Simulation, Sous-modularité, Entropie.

Abstract

Network design is a rich subfield of combinatorial optimization with wide-ranging real-life applications. From a methodological standpoint, most problems in this class are notoriously difficult due to their combinatorial nature and the interdependence of the decisions they involve. This thesis addresses two network design problems whose respective structures pose very distinct challenges. First, we consider a deterministic problem in which a customer must acquire at the minimum price a number of units of a product from a set of vendors offering different fixed and unit costs and whose supply is limited. Second, we study a probabilistic problem in which a firm entering an existing market seeks, by opening a number of facilities from a set of available locations, to maximize its expected share in a market composed of random utility-maximizing customers. These two problems, namely the single-sink fixed-charge-transportation problem and the choice-based competitive facility location problem, are closely related to the knapsack problem and the maximum covering problem, respectively. We introduce novel model reformulations that leverage these connections to classical combinatorial optimization problems. In both cases, we exploit these reformulations to prove new theoretical properties and to develop efficient solution methods. Our novel algorithm for the single-sink fixed-charge-transportation problem dominates the state-of-the-art methods from the literature, reducing the solving time of large instances by up to four orders of magnitude. Another notable contribution of this thesis is the demonstration that the objective function of the choice-based competitive facility location problem is submodular under any random utility maximization model. Our simulation-based method exploits this property and achieves state-of-the-art results for several groups of instances.

Keywords : Network design, Combinatorial optimization, Choice-based optimization, Facility location, Fixed-charge transportation problem, Knapsack problem, Lagrangian relaxation, Simulation, Submodularity, Entropy.

Contents

List of Tables

List of Figures

List of Acronyms & Abbreviations

B&C	Branch-and-cut
CBCFLP	Choice-based competitive facility location problem
DPA	Dynamic programming algorithm
EA	Enumerative algorithm
FCTP	Fixed-charge transportation problem
IID	Independent and identically distributed
GEV	Generalized extreme value
GGX	Greedy heuristic, gradient-based local search, and exchanging
KP	Knapsack problem (binary)
KTA	Knapsack transformation algorithm

MCP	Maximum covering problem
MILP	Mixed-integer linear programming
MINLP	Mixed-integer nonlinear programming
MMNL	Mixed multinomial logit
MNL	Multinomial logit
MOA	Multicut outer-approximation
PMF	Probability mass function
RUM	Random utility maximization
SAA	Sample average approximation
SAAA	Sample average approximation with aggregation
SHS	Simulation-based hybrid submodular
SSFCTP	Single-sink fixed-charge transportation problem

Remerciements

Avant toute chose, je tiens à remercier Bernard Gendron, qui m'a transmis avec tant de bienveillance sa curiosité et son enthousiasme pour la recherche opérationnelle. Bernard aura été pour moi un modèle d'exemplarité académique et humaine dont j'aspire à honorer la mémoire.

Je souhaite adresser les remerciements les plus sincères à ma superviseure Emma Frejinger qui, avec la même bonté et la même rigueur, m'a offert un encadrement académiquement stimulant et profondément enrichissant au niveau personnel. Je suis tout aussi redevable à mon co-superviseur Jean-François Côté, qui m'a accompagné depuis mes débuts en recherche et avec qui je prends toujours autant de plaisir à collaborer. Vous avez contribué substantiellement à l'épanouissement de la personne que je suis aujourd'hui et à l'établissement des bases de la carrière de chercheur que je travaille à construire.

Je remercie également mes collaborateurs Zsuzsa Bakk, Michel Bierlaire, Guillaume Dubé, Michel Gendreau, Charles-Édouard Giguère, Tom Hearing, Eric Lacourse, Félix Laliberté, Sacha Morin, Lucas Parada, Roxane de la Sablonnière et Fabian Torres, avec qui j'ai eu la chance de travailler lors des deux dernières années au CIRRELT et au Laboratoire sur les changements sociaux et l'identité de l'Université de Montréal ainsi qu'au Laboratoire transport et mobilité (TRANSP-OR) de l'EPFL. Ces collaborations multidisciplinaires diversifiées ont enrichi mon expérience de maîtrise, ont étendu l'horizon de mes intérêts et m'ont permis de côtoyer de magnifiques équipes de recherche.

Je souhaite adresser des remerciements à quelques professeurs dont les enseignements ont particulièrement marqué mon passage à l'Université de Montréal : Fabian Bastin, Mylène Bédard, Margarida Carvalho, Pierre L'Ecuyer, Simon Lacoste-Julien et Florian Maire. J'adresse également des remerciements particuliers à Julia Fulvi, Marie-Ève Provost-Larose, Danny St-Pierre et Dominique Tessier qui ont été les premiers à réellement éveiller mon intérêt pour les mathématiques lors de mes études préuniversitaires au Collège de Valleyfield.

Merci à mes amis et colocataires Guillaume, Justin, Mathieu et Philippe, avec qui j'ai partagé de magnifiques années universitaires à Montréal.

Finalement, merci à mon amoureuse Shophika, à mes parents Claire et Martin, et à toute ma famille qui m'a toujours témoigné une totale confiance et un support inconditionnel à travers les projets que j'entreprends.

Introduction

The core of this thesis is composed of two articles that respectively address a deterministic transportation problem and a probabilistic facility location problem, both belonging to the broader class of network design problems. This introduction provides a brief overview of the history of network design, the motivation behind the choice of problems under study, and a summary of our main contributions.

Network design is defined as the class of optimization problems in which the decisions can be represented as the selection of a subset of edges in a graph (Crainic et al., 2021, Magnanti and Wong, 1984, Minoux, 1989). This characterization encompasses a wide variety of important applications in several areas, including transportation (e.g., Crainic, 2000, Farahani et al., 2013, Wieberneit, 2008), logistics (e.g., Chanintrakul et al., 2009, Cordeau et al., 2021, Nourbakhsh et al., 2016), and telecommunication (e.g., Kershenbaum, 1993, Mateus et al., 2000, Resende and Pardalos, 2008). Depending on the topology of the graph, the nature of the constraints, and the complexity of the objective function, network design problems can take on completely different structures. The variety and inherent difficulty of these combinatorial problems have made the area of network design fertile ground for the development of novel model reformulations (e.g., Crainic and Gendron, 2020, Frangioni and Gendron, 2009, Gendron, 2019) and mathematical decomposition methods (Gendron, 2011).

Starting from the early 1960s, within the foundational literature on network design, considerable focus has been directed toward fixed-charge problems. Notable contributions in this realm include the works of Balinski (1961) and Hirsch and Dantzig (1968). Fixed-charge problems involve binary decision variables, each corresponding to selecting an edge in the graph and associated with a fixed cost in the objective function. A prime example of a fixed-charge network design problem is the fixed-charge transportation problem (FCTP). Introduced by Balinski (1961), this problem is defined on a bipartite graph where the nodes are divided into sources and sinks. In addition to its fixed cost, each activated arc incurs a variable cost proportional to the number of units of a product sent from its source to its sink. The problem is to minimize the total cost while respecting the arcs' capacities and the integer supplies and demands of the nodes. The special case of the FCTP in which

the bipartite graph includes either a single source or a single sink is called the single-sink fixed-charge transportation problem (SSFCTP).

One reason for the prominent place of the FCTP in the network design literature is that the more general single-commodity capacitated fixed-charge network design problem, which does not require the graph to be bipartite, can be reformulated as a FCTP through network flow transformations (Malek-Zavarei and Frisch, 1972). Recent works on the FCTP include branch-and-price methods by Roberti et al. (2015) and Mingozzi and Roberti (2018). These exact algorithms exploit integer programming formulations with exponentially many decision variables corresponding to flow patterns from sources to sinks, and from sinks to sources. Zhao et al. (2018) proposed a convexified formulation of the FCTP based on Lagrangian decomposition and column generation. The column generation problems that have to be solved in this approach are separable by source node and can be formulated as instances of the SSFCTP. A similar structure is found in Lagrangian relaxations of the FCTP (Görtz and Klose, 2007). An efficient exact method for solving the SSFCTP is thus an essential component of different solution methods for the FCTP.

The first specialized method proposed in the literature for solving the SSFCTP was a branch-and-bound algorithm by Haberl (1991). This work was motivated by an application in sawn timber production (Haberl et al., 1991). At that time, the denomination SSFCTP had not yet appeared. Haberl (1991) instead referred to the problem as a *continuous knapsack problem with fixed charges*. The name SSFCTP was introduced by Herer et al. (1996), who developed an enumeration algorithm based on dominance rules and lower bounds. A dynamic programming algorithm for the SSFCTP was then developed by Alidaee and Kochenberger (2005). Klose (2008) proposed exploiting dual information to reduce the problem size and revisited both the enumeration algorithm of Herer et al. (1996) and the dynamic programming algorithm of Alidaee and Kochenberger (2005). To obtain their improved enumerative algorithm, Klose (2008) directly adapted an algorithm of Martello and Toth (1977) for the binary knapsack problem (KP) to the SSFCTP. This approach was the first to substantially leverage the close connection to the KP that had initially been underscored in the original name of the SSFCTP.

The first paper of this thesis follows the same line and seeks to fully exploit the common structure of the SSFCTP and the KP to fill the gap between the state of the art on these problems. Whereas the best algorithms for the KP are the result of an extremely rich and dense body of literature that has been developed for decades (e.g., Martello and Toth, 1987, 1990, Martello et al., 1999), only a few methods for solving the SSFCTP have been proposed in the literature, leaving room for significant improvement. We introduce a novel binary nonlinear formulation of the SSFCTP and present an array of upper and lower bounds that can be computed based on the solutions of KPs derived from this reformulation. We also show that the SSFCTP can be solved exactly through a number of KPs equal to the number

of source nodes in the graph. We develop a sequence of tests to eliminate the vast majority of these subproblems. The resulting exact method, which we call the *knapsack transformation algorithm* (KTA), can solve most instances of the SSFCTP to proven optimality by solving only one or two KPs. Our algorithm dominates existing methods from the literature across all our benchmark instances and requires up to four orders of magnitude less computing time than the previous state-of-the-art methods on large instances.

The process of reformulating a challenging network design problem into a simpler combinatorial problem is also at the core of the second part of this thesis, where we turn our attention to facility location. This class of problems concerns the selection of an optimal subset of locations from a set of possible sites for warehouses, plants, sensors, or other types of equipment. With roots going back to a geometric problem generally attributed to the French mathematician Pierre de Fermat (1601-1665), facility location is one of the oldest and most extensively studied topics in the operations research literature (see, for example, the reviews by Laporte et al., 2019, Owen and Daskin, 1998, Snyder, 2006). Although facility location problems are usually not presented as network design problems, they involve the selection of a subset of nodes in a graph. This graph can be slightly modified, for instance by adding a loop to each node, to cast these decisions as the selection of edges. As pointed out by Crainic et al. (2021), facility location thus falls within the scope of network design.

Most of the location models in the literature do not account for competition or differences among facilities and make simple assumptions about the structure of the demand. For example, it is common to allocate customers to facilities deterministically based on proximity (Drezner, 2014). However, several real-world applications, such as retail location (Brown, 1989), occur in competitive environments where the demand is captured by different noncooperative organizations. Furthermore, the attractiveness of each available location generally varies across individuals and depends on a combination of factors, as discussed by Aros-Vera et al. (2013) and Holguin-Veras et al. (2012) in the context of park-and-ride facility location.

The choice-based competitive facility location problem (CBCFLP) integrates these considerations on competition and customer preferences heterogeneity. In this problem, a firm entering an existing market occupied by competitors seeks to select a subset of facilities from a set of available locations to maximize its expected market share while respecting linear business constraints, such as a fixed number of facilities to open or a limited budget. It is assumed that the customers behave according to the random utility maximization (RUM) principle, meaning that they evaluate each available alternative and patronize the one that maximizes a random utility function (Manski, 1977, McFadden and Train, 2000b). Significant attention has been devoted to this problem over the last two decades (e.g., Aros-Vera et al., 2013, Benati and Hansen, 2002, Freire et al., 2016, Ljubić and Moreno, 2018, Mai and Lodi, 2020, Zhang et al., 2012). However, most of the literature focuses on specific families of RUM models that make restrictive assumptions about the structure of the demand.

The second paper of this thesis addresses the CBCFLP from a model-free perspective. This work shows that, without relying on any restrictive assumption about the utility functions of the customers, we can reformulate the problem as a maximum covering problem (MCP) (Church and ReVelle, 1974) with a number of demand points that can be exponential in the number of available locations. Through this novel deterministic equivalent reformulation, we show that the objective function of the CBCFLP is submodular under any RUM model. This result extends earlier findings by Benati (1997) and Dam et al. (2022), who established the submodularity of the objective function under specific RUM models. We propose a simulation approach to approximate the deterministic equivalent reformulation via a MCP of manageable size, which we solve by a branch-and-cut algorithm based on submodular cuts. We develop an information-theoretic analysis of our solution method and draw formal connections between its performance and the entropy level of the customers' preferences in the population. Our computational results indicate that our method achieves state-of-the-art results on large instances defined under a fully flexible RUM model.

To summarize, this thesis explores novel reformulations and proposes new solution methods for two important network design problems. In the first paper, we reformulate the SSFCTP into a number of KPs that is linear in the size of the original problem and develop a sequence of procedures to eliminate most of them based on a small set of initial solutions. In the second paper, we reformulate the CBCFLP into a large-scale maximum covering problem and develop a solution method based on submodularity to solve a sample average approximation of this new model. These contributions lead to significant improvements to the state of the art for both problems. In the remainder of the thesis, the two papers are presented sequentially and are followed by a conclusion.

First Article.

A Novel Reformulation for the Single-Sink Fixed-Charge Transportation Problem

by

Robin Legault[1], Jean-François Côté[2], and Bernard Gendron[1]

([1]) Department of Computer Science and Operations Research and CIRRELT, Université de Montréal, Montreal, Quebec H3T 1J4, Canada

([2]) Department of Operations and Decision Systems and CIRRELT, Université Laval, Quebec City, Quebec G1V 0A6, Canada

Journal submission

This article was submitted to Mathematical Programming in February 2022, was accepted for publication in January 2023, and appeared online in February 2023 (Legault et al., 2023). The version presented in this thesis incorporates minor modifications and additions to the published article.

Authors contributions

- Bernard Gendron proposed the original research topic (Lagrangian relaxation methods for the single-sink fixed-charge transportation problem).
- The core of the research (development of models, bounds, dominance relations, filtering procedures, proofs, algorithms, code, etc.) was carried out by Robin Legault.
- Throughout the project, the three authors participated in regular meetings to discuss new research ideas and the latest results.
- The article was written by Robin Legault, and it was revised by Jean-François Côté.

RÉSUMÉ. Le problème de transport à coûts fixes à un puits a des applications naturelles dans les domaines manufacturiers et des transports et constitue un sous-problème important du problème de transport à coûts fixes. Cependant, les algorithmes spécialisés de la littérature n'exploitent pas pleinement la structure de ce problème, au point d'être surpassés par les solveurs commerciaux pour les instances de grande taille. Nous introduisons une nouvelle reformulation du problème et étudions ses propriétés théoriques. Cette reformulation conduit à une gamme de nouvelles bornes supérieures et inférieures, des relations de dominance, des relaxations linéaires et des procédures de filtrage. L'algorithme qui en résulte comprend une phase heuristique et une phase exacte, dont l'étape principale consiste à résoudre un petit nombre de sous-problèmes de sac-à-dos. Des expériences computationnelles basées sur des instances existantes et nouvelles sont présentées. Ces tests indiquent que notre algorithme divise systématiquement le temps de résolution par plusieurs ordres de grandeur comparativement à l'état de l'art.
Mots clés : Optimisation combinatoire, Problème de transport à coûts fixes, Relaxation lagrangienne, Problème du sac à dos.

ABSTRACT. The single-sink fixed-charge transportation problem has natural applications in the area of manufacturing and transportation and is an important subproblem of the fixed-charge transportation problem. However, even the best algorithms from the literature do not fully leverage the structure of this problem, to the point of being outperformed by modern general-purpose mixed-integer programming solvers for large instances. We introduce a novel reformulation of the problem and study its theoretical properties. This reformulation leads to a range of new upper and lower bounds, dominance relations, linear relaxations, and filtering procedures. The resulting algorithm includes a heuristic phase and an exact phase, the main step of which is to solve a small number of knapsack subproblems. Computational experiments are presented for existing and new types of instances. These tests indicate that the new algorithm systematically reduces the solving time of the state-of-the-art exact methods by several orders of magnitude.
Keywords : Combinatorial optimization, Fixed-charge transportation problem, Lagrangian relaxation, Knapsack problem.

1. Introduction

The single-sink fixed-charge transportation problem (SSFCTP) can be stated as a distribution problem in which a single customer acting as a sink node and having access to a set of n supplier nodes $j = 1, \ldots, n$ must acquire D units of the same product. Each node j can ship up to b_j units to the sink at a fixed-charge f_j and a cost c_j per unit. The problem is to minimize the total cost while respecting the demand of the sink. The following model, which we call P1, is the classical formulation of the SSFCTP. It was first studied by Herer et al. (1996).

$$[\text{P1}] \quad Z^1 = \min \sum_{j=1}^{n} (c_j x_j + f_j y_j) \tag{1.1}$$

$$\text{s.t.} \sum_{j=1}^{n} x_j = D, \tag{1.2}$$

$$0 \leq x_j \leq b_j y_j \quad \text{for } j = 1, \ldots, n, \tag{1.3}$$

$$y_j \in \{0,1\} \quad \text{for } j = 1, \ldots, n. \tag{1.4}$$

The binary decision variable y_j equals 1 if the fixed cost of node j is paid. In this case, as specified in constraint (1.3), variable x_j, which specifies the number of units ordered from this supplier, can take any positive value less than or equal to b_j. The demand constraint is given by equation (1.2).

It is assumed that each supplier offers at least one unit and that the total offer suffices to satisfy the demand, i.e., $1 \leq b_j \leq D$ and $\sum_{j=1}^{n} b_j \geq D$. Furthermore, the demand D, the number of units x_j ordered from each supplier, and the capacities b_j are assumed to be integer-valued. Without loss of generality, it is also assumed that all the unit and fixed costs are nonnegative, i.e., $c_j \geq 0$ and $f_j \geq 0$ for each supplier $j \in \{1, \ldots, n\}$. An instance that contains negative unit costs can be transformed to respect this assumption by redefining the unit costs as $\hat{c}_j = c_j - \min_{i \in \{1,\ldots,n\}} \{c_i\}$ without affecting the optimal solution. Similarly, the restricted problem that is obtained after fixing $y_j = 1$ for each node j such that $f_j < 0$ shares the same optimal solution as the original problem.

As shown by Klose (2008), the SSFCTP is $\mathcal{NP}$-hard. This problem first gained attention in the literature due to its natural applications in manufacturing and transportation (Haberl, 1991, Herer et al., 1996). Furthermore, staircase transportation cost functions can be introduced to the SSFCTP to obtain the so-called *single-sink, fixed-charge, multiple-choice transportation problem*, which arises as a relaxation of more general minimum-cost network flow problems (Christensen et al., 2013). Most importantly, the SSFCTP arises as a subproblem in Lagrangian relaxation (Görtz and Klose, 2007) and column generation (Zhao et al., 2018) methods for solving the fixed-charge transportation problem (FCTP). The SSFCTP

is also a special case of a relaxation of the multicommodity capacitated fixed-charge network design problem (Akhavan Kazemzadeh et al., 2022).

Two main approaches have been used for solving the SSFCTP in the literature. The first is an implicit enumeration algorithm proposed by Herer et al. (1996) that refines an algorithm from Haberl (1991). The second is a dynamic programming method introduced by Alidaee and Kochenberger (2005). Both methods were revisited and significantly improved by Klose (2008). In particular, some of these improvements over the original algorithms exploit the similarities between the SSFCTP and the binary knapsack problem (KP) to take advantage of ideas first developed by Martello and Toth (1977).

This paper takes a step further in this direction by introducing a new mathematical formulation of the problem that allows us to express both a relaxation and subproblems of the SSFCTP as KPs. The algorithm we introduce is thus referred to as the *knapsack transformation algorithm* (KTA). It is composed of a heuristic phase, a filtering phase and an exact phase that are executed sequentially.

The different models developed in this article aim to reduce the search space as much as possible while ensuring that at least one optimal solution to the original problem remains accessible. Doing so, the fast algorithms that have been proposed over time for solving the KP (e.g., Martello and Toth, 1997, Pisinger, 1997, 2000) can be fully mobilized to solve the SSFCTP efficiently.

Our paper offers five main contributions that are detailed in Sections 2, 3, 4, 5, and 7, respectively.

Section 2: We introduce model P2, a binary nonlinear reformulation of the problem. This new mathematical formulation takes advantage of important properties from the literature to exploit the strong similarity that links the SSFCTP to the KP.

Section 3: A new heuristic method is proposed. Its main step is to solve knapsack subproblems, noted P3, that are obtained through a relaxation of P2.

Section 4: Two filtering techniques based on a new strict total order on the suppliers are introduced. Together, they drastically reduce the size of the subproblems that need to be solved in the exact phase of KTA.

Section 5: We present a transformation of the subproblem that arises after allowing a single node to supply a positive number of units that is less than its capacity. This reformulation, which is at the core of the exact phase of the algorithm, is also expressed as a KP.

Section 7: We introduce new instances, with the aim of providing an in-depth analysis of the performance of KTA against existing techniques from the literature.

The remainder of the paper is structured as follows. Section 6 contains a detailed summary of KTA. In addition to new generation methods, Section 7 presents the results of

our numerical experiments, including an analysis of the performance of each step of our algorithm. Section 8 concludes the article.

2. Novel reformulation

First, let us introduce some definitions that are used throughout the paper. In a solution $(\boldsymbol{x}, \boldsymbol{y})$ to P1, a node $j \in \{1, \ldots, n\}$ is said to be:

- **unused** if $x_j = 0$,
- **partial** if $1 \leq x_j \leq b_j - 1$,
- **complete** if $x_j = b_j$.

The following propositions recall well-known properties of the SSFCTP, which are proven in Rosenblatt et al. (1998).

Proposition 1. *There exists an optimal solution to P1 that contains at most one partial node.*

Proposition 2. *If an optimal solution contains a partial node $p \in \{1, \ldots, n\}$, then $y_j = 0$ for each node $j \in \{1, \ldots, n\}$ such that $c_j > c_p$.*

We now introduce a new binary nonlinear formulation of the SSFCTP that is based on Proposition 1. In this model, the integer-valued decision variables of P1 are replaced by binary variables representing the selection of each node in the solution, among which only one can be partial.

$$\text{[P2]} \quad Z^2 = \min \sum_{j=1}^{n} (c_j b_j + f_j) y_j - \Big(\sum_{j=1}^{n} b_j y_j - D \Big) \sum_{j=1}^{n} c_j z_j \tag{2.1}$$

$$\text{s.t.} \quad \sum_{j=1}^{n} b_j y_j \geq D, \tag{2.2}$$

$$\sum_{j=1}^{n} b_j y_j - D \leq \sum_{j=1}^{n} b_j z_j - 1, \tag{2.3}$$

$$\sum_{j=1}^{n} z_j = 1, \tag{2.4}$$

$$z_j \leq y_j \qquad \text{for } j = 1, \ldots, n, \tag{2.5}$$

$$y_j, z_j \in \{0,1\} \qquad \text{for } j = 1, \ldots, n. \tag{2.6}$$

In this formulation, a node $j \in \{1, \ldots, n\}$ is by default considered to be complete if y_j=1, leading to a cost of $c_j b_j + f_j$. The node $p \in \{1, \ldots, n\}$ such that z_p=1 then becomes partial if constraint (2.2) does not hold with equality. This leads to a reimbursement of c_p for each of the $\sum_{j=1}^{n} b_j y_j - D$ excess units that were initially paid. The node p that can be partial is unique and is selected among the nodes that are used in the solution, by constraints (2.4) and (2.5). Constraint (2.3) prevents it from supplying a negative number of units.

Model P2 significantly reduces the search space compared to P1 by excluding all the solutions containing more than one partial node from the feasible set. Proposition 2 can also be exploited to strengthen this reformulation of the problem. The following strict total order on the nodes serves this purpose.

Definition 3. *The strict total order $\prec$ on the set of nodes $\{1,\ldots,n\}$ is defined as follows: $i \prec j \iff \Big((c_i{<}c_j)$ or $(c_i{=}c_j$ and $b_i{<}b_j)$ or $(c_i{=}c_j$ and $b_i{=}b_j$ and $i{<}j)\Big)$. We denote by $\succ$ its inverse relation.*

Definition 4. *Given a vector $\bar{\boldsymbol{y}} \in \{0,1\}^n$, we define the critical node $p(\bar{\boldsymbol{y}})$ as the maximal element of the strictly totally ordered set $(N(\bar{\boldsymbol{y}}), \prec)$, where $N(\bar{\boldsymbol{y}}){=}\Big\{j{\in}\{1,\ldots,n\} : \bar{y}_j{=}1\Big\}$ is the set of nodes that are used in this solution.*

Proposition 5. *There exists an optimal solution to P2 that respects the following nonlinear constraint:*

$$z_j = \mathbb{1}(j{=}p(\boldsymbol{y})) \qquad \text{for } j = 1,\ldots,n. \tag{2.7}$$

Proof. Let $(\boldsymbol{y}^2, \boldsymbol{z}^2)$ be an optimal solution to P2 and let i be the node such that $z_i^2{=}1$. The proof proceeds by construction of an optimal solution $(\boldsymbol{y}^2, \boldsymbol{z}^*)$, where $z_j^* = \mathbb{1}(j{=}p(\boldsymbol{y}^2))$ $\forall j$. First, if $\sum_{j=1}^n b_j^2 y_j^2 - D = 0$, then the objective value does not depend on the variables z_j and constraint (2.3) is trivially respected by solution $(\boldsymbol{y}^2, \boldsymbol{z}^*)$, since $b_{p(\boldsymbol{y}^2)}{\geq}1$. Otherwise, node i is partial in the optimal solution $(\boldsymbol{y}^2, \boldsymbol{z}^2)$. In this case, it follows from Proposition 2 and the definition of the critical node $p(\boldsymbol{y}^2)$ that $c_i{=}c_{p(\boldsymbol{y}^2)}$ and $b_i{\leq}b_{p(\boldsymbol{y}^2)}$. Since $c_i{=}c_{p(\boldsymbol{y}^2)}$, the objective value of both solutions $(\boldsymbol{y}^2, \boldsymbol{z}^2)$ and $(\boldsymbol{y}^2, \boldsymbol{z}^*)$ is the same. Furthermore, as $b_i{\leq}b_{p(\boldsymbol{y}^2)}$, the right-hand side of constraint (2.3) is not smaller for solution $(\boldsymbol{y}^2, \boldsymbol{z}^*)$ than for solution $(\boldsymbol{y}^2, \boldsymbol{z}^2)$. In both cases, $(\boldsymbol{y}^2, \boldsymbol{z}^*)$, which respects constraint (2.7) by construction, is thus feasible and optimal. □

Neither P2 nor P2 with (2.7) are intended to be solved directly. However, Proposition 5 offers a direct rule that can be applied for the selection of a partial node given a vector $\boldsymbol{y}$ that specifies the set of suppliers to be used in a solution. Using it, the SSFCTP reduces to identifying the optimal subset of nodes that must provide a positive number of units. This idea motivates the introduction, in the next section, of a transformation of P2 that can be efficiently solved to identify such subsets.

3. Lower and upper bounds

This section presents the four main steps of the heuristic phase in their order of execution. First, the classical linear relaxation of the SSFCTP and a simple greedy upper bound are briefly presented in Section 3.1. The rest of the section consists of new contributions.

Section 3.2 develops model P3, a KP derived from a relaxation of P2, and studies its properties. Section 3.3 presents several bounds obtained through P3. Section 3.4 presents a new dominance relation between partial nodes. Finally, Section 3.5 introduces a strong linear relaxation method that can be applied after fixing a supplier p as the partial node. Both the dominance relation and the strong linear relaxation use the results of Section 3.2 to eliminate as many potential partial nodes as possible.

Different types of bounds are extracted from each step. These include bounds on the objective value, on the conditional objective value and on the number of units x_p ordered from node p given that it is partial, and on the partial node's unit cost.

3.1. Elementary bounds

We start by considering a pair of simple lower and upper bounds. Although generally weak, they can in some cases reduce the computing time required by the more expensive steps of the algorithm.

3.1.1. Classical linear relaxation

A classical lower bound on the objective value of P1 is obtained by solving its linear relaxation.

$$[\text{P1}^{\text{LP}}] \quad Z_{\text{LB}}^{\text{LP}} = \min \sum_{j=1}^{n} e_j x_j \tag{3.1}$$

$$\text{s.t. } (1.2),$$

$$0 \leq x_j \leq b_j \text{ for } j = 1,\ldots,n, \tag{3.2}$$

where $e_j{=}c_j{+}f_j/b_j$ denotes the linearized cost of node $j \in \{1,\ldots,n\}$.

Assuming that the nodes are sorted in non-decreasing order of linearized cost, the optimal solution to the LP relaxation can be calculated in $\mathcal{O}(n)$ and is given by

$$x_j = \begin{cases} b_j & \text{for } j = 1,\ldots,s-1, \\ D - \sum_{j=1}^{s-1} b_j & \text{for } j = s, \\ 0 & \text{for } j = s+1,\ldots,n, \end{cases}$$

where the so-called *split node* $s \in \{1,\ldots,n\}$ is such that

$$\sum_{j=1}^{s-1} b_j < D \leq \sum_{j=1}^{s} b_j.$$

The optimality of this solution can be demonstrated by considering the dual of the linear program P1^{LP}, as it is done by Klose (2008). Throughout the article, it is assumed that $e_1 \leq e_2 \leq \cdots \leq e_n$. This ordering is required by various steps of our algorithm. Even if

the nodes were not initially sorted, computing $Z_{\mathrm{LB}}^{\mathrm{LP}}$ could still be done in $\mathcal{O}(n)$ using the algorithm of Balas and Zemel (1980) to determine the split node in linear time.

3.1.2. Greedy upper bound

Görtz and Klose (2009) analyzed a range of popular greedy algorithms for the SSFCTP and showed that the upper bounds they offer can be arbitrarily bad. Nevertheless, it can be useful to derive a simple upper bound from the optimal solution $\boldsymbol{x}^*$ to $\mathrm{P1}^{\mathrm{LP}}$ in $\mathcal{O}(n)$. Our slightly modified version of the greedy algorithm for the SSFCTP uses Definition 4 to improve this classical method from the literature. Let p be the maximal element of the set of nodes $\{1,\ldots,s\}$ with respect to the strict total order $\prec$. Also, let us denote by $k{=}\min\{\sum_{i=1}^{s} b_i{-}D, b_p\}$ the number of additional units that can be supplied by node s instead of node p without violating constraint (1.3). A feasible solution to P1 is given by $(\boldsymbol{x}, \boldsymbol{y})$, where

$$y_j = \begin{cases} 1 & \text{for } j = 1,\ldots,s, \\ 0 & \text{for } j = s+1,\ldots,n, \end{cases}$$

$$x_j = \begin{cases} x_j^* - k \cdot \mathbb{1}(j = p) & \text{for } j = 1,\ldots,s-1, \\ x_j^* + k \cdot \mathbb{1}(j \neq p) & \text{for } j = s, \\ 0 & \text{for } j = s+1,\ldots,n. \end{cases}$$

The objective value of this solution is $Z_{\mathrm{UB}}^{\mathrm{G}}{=}Z_{\mathrm{LB}}^{\mathrm{LP}}{+}f_s{-}(e_s{-}c_s)x_s^*{-}k(c_p{-}c_s)$. Increasing the number of units used on the split node by k to reduce the number of units ordered from node p lowers the objective value by $k(c_p{-}c_s)$ compared to the classical greedy upper bound, in which nodes $j = 1,\ldots,s-1$ are completely used and node s provides the remaining demand.

3.2. Knapsack relaxation

We now introduce a transformation of P2, noted P3, in which constraint (2.3) is relaxed, and the decision variables z_j are removed. Furthermore, the reimbursement rate $\sum_{j=1}^{n} c_j z_j$ per excess unit is replaced by a multiplier $\lambda{\geq}0$, which can be interpreted as an approximation of the unit cost of the partial node, if it exists. This relaxation of P2 corresponds to a min-KP, with weight b_j and cost $(c_j{-}\lambda)b_j{+}f_j$ on item $j \in \{1,\ldots,n\}$, and demand D. The following problem can be transformed into a standard KP using the procedure provided in

Martello and Toth (1990).

$$\text{[P3]} \quad Z^3(\lambda) = \min \sum_{j=1}^{n}(c_j b_j + f_j)y_j - \Big(\sum_{j=1}^{n} b_j y_j - D\Big)\lambda \tag{3.3}$$

$$= \lambda D + \min \sum_{j=1}^{n}\Big((c_j - \lambda)b_j + f_j\Big)y_j \tag{3.4}$$

$$\text{s.t. } (1.4),(2.2).$$

Let $S^1 \subseteq \mathbb{N}^n \times \{0,1\}^n$, $S^2 \subseteq \{0,1\}^n \times \{0,1\}^n$ and $S^3 \subseteq \{0,1\}^n$ denote the feasible sets of problems P1, P2 and P3, respectively. Important subsets of solutions for each of these models are now introduced. We note:

- $\tilde{S}^1 \subseteq S^1$ the set of feasible solutions $(\boldsymbol{x}^1,\boldsymbol{y}^1)$ to P1 in which only the critical node $p(\boldsymbol{y}^1)$ can be partial and no fixed cost is unnecessarily paid (i.e. $y_j^1 \leq x_j^1 \ \forall j$).
- $\tilde{S}^2 \subseteq S^2$ the set of feasible solutions $(\boldsymbol{y}^2, \boldsymbol{z}^2)$ to P2 such that $z^2_{p(\boldsymbol{y}^2)} = 1$.
- $\tilde{S}^3 \subseteq S^3$ the set of feasible solutions $\boldsymbol{y}^3$ to P3 that respect the following inequality:

$$\sum_{j=1}^{n} b_j y_j^3 - D \leq b_{p(\boldsymbol{y}^3)} - 1. \tag{3.5}$$

In a solution $\boldsymbol{y}^3 \in \tilde{S}^3$, the nodes j such that $y_j^3{=}1$ are used and, among them, the critical node $p(\boldsymbol{y}^3)$ is the only one that can be partial. The number of units supplied by the critical node is given by the difference between the demand D and the total offer of the nodes of set $N(\boldsymbol{y}^3) \setminus \{p(\boldsymbol{y}^3)\}$. This difference ranges between 1 and $b_{p(\boldsymbol{y}^3)}$, by inequalities (2.2) and (3.5). Consequently, $\boldsymbol{y}^3$ is associated with the solution of $\tilde{S}^2$ in which the same nodes are used. Formally, $\tilde{S}^1$, $\tilde{S}^2$ and $\tilde{S}^3$ are linked through the following bijections.

- $\tilde{f}_{3,2} : \tilde{S}^3 \to \tilde{S}^2$, $\tilde{f}_{3,2}(\boldsymbol{y}^3) = (\boldsymbol{y}^3,\boldsymbol{z}^2)$, where $z_j^2 = \mathbb{1}(j{=}p(\boldsymbol{y}^3)), \ \forall \ j{\in}\{1,\ldots,n\}$.
- $\tilde{f}_{2,1} : \tilde{S}^2 \to \tilde{S}^1$, $\tilde{f}_{2,1}(\boldsymbol{y}^2, \boldsymbol{z}^2) = (\boldsymbol{x}^1,\boldsymbol{y}^2)$, where $x_j^1 = b_j y_j^2 - (\sum_{i=1}^{n} b_i y_i^2 - D)z_j^2, \ \forall \ j \in \{1,\ldots,n\}$.
- $\tilde{f}_{3,1} : \tilde{S}^3 \to \tilde{S}^1$, $\tilde{f}_{3,1}(\boldsymbol{y}^3) = \tilde{f}_{2,1}\Big(\tilde{f}_{3,2}(\boldsymbol{y}^3)\Big)$.

Let $(\boldsymbol{x}^1,\boldsymbol{y}^1)$, $(\boldsymbol{y}^2,\boldsymbol{z}^2)$ and $\boldsymbol{y}^3$ be feasible solutions to P1, P2 and P3, respectively. Their objective values are denoted by $Z^1(\boldsymbol{x}^1,\boldsymbol{y}^1)$, $Z^2(\boldsymbol{y}^2,\boldsymbol{z}^2)$ and $Z^3(\lambda, \boldsymbol{y}^3)$, where $\lambda{\geq}0$ is a given multiplier. The bijections defined above can be used to identify the feasible solutions to P2 and P1 associated with a solution $\boldsymbol{y}^3 \in \tilde{S}^3$. Their objective value is obtained from $Z^3(\lambda, \boldsymbol{y}^3)$ by replacing the anticipated reimbursement rate λ by the unit cost of the critical node $p(\boldsymbol{y}^3)$. This leads to the following equalities.

$$Z^1\Big(\tilde{f}_{3,1}(\boldsymbol{y}^3)\Big) = Z^2\Big(\tilde{f}_{3,2}(\boldsymbol{y}^3)\Big) = Z^3(\lambda, \boldsymbol{y}^3) + \Big(\sum_{j=1}^{n} b_j y_j^3 - D\Big)(\lambda - c_{p(\boldsymbol{y}^3)}) \tag{3.6}$$

By construction, $\tilde{S}^2$ is composed of the feasible solutions to P2 that respect constraint (2.7). By Proposition 5, at least one of them is thus optimal. Therefore, there exists a feasible solution to P3 to which the bijections $\tilde{f}_{3,2}$ and $\tilde{f}_{3,1}$ can be applied to produce an

optimal solution to both formulations P2 and P1 of the SSFCTP. The next propositions state other essential properties of P3 that are exploited throughout the algorithm.

Proposition 6. *$Z^3(\lambda)$ is a lower bound on the objective value of any solution to the SSFCTP that does not contain a partial node or contains a partial node whose unit cost is less than or equal to λ.*

Proof. Such a solution corresponds to an element $(\boldsymbol{y}^2, \boldsymbol{z}^2)$ of S^2 respecting either $\sum_{j=1}^{n} b_j z_j^2 - D = 0$ (if the solution does not contain a partial node) or $\sum_{j=1}^{n} c_j z_j^2 \leq \lambda$ (if the solution contains a partial node whose unit cost is inferior or equal to λ). Furthermore, since $(\boldsymbol{y}^2, \boldsymbol{z}^2) \in S^2$, the vector $\boldsymbol{y}^2$ is also a feasible solution to P3 and provides an upper bound on the optimal value $Z^3(\lambda)$ of P3 for multiplier λ. The expected inequality follows:

$$\begin{aligned} Z^3(\lambda) &\leq Z^3(\lambda, \boldsymbol{y}^2) \\ &= \sum_{j=1}^{n} (c_j b_j + f_j) y_j^2 - \Big(\sum_{j=1}^{n} b_j y_j^2 - D \Big) \lambda \\ &\leq \sum_{j=1}^{n} (c_j b_j + f_j) y_j^2 - \Big(\sum_{j=1}^{n} b_j y_j^2 - D \Big) \sum_{j=1}^{n} c_j z_j^2 \\ &= Z^2(\boldsymbol{y}^2, \boldsymbol{z}^2) \end{aligned}$$

If the solution does not contain a partial node, the second inequality holds with equality. Otherwise, it follows from the demand constraint (2.2) and the fact that $\sum_{j=1}^{n} c_j z_j^2 \leq \lambda$. $\square$

Proposition 7. *The function $Z^3(\lambda)$ is non-increasing and concave for $\lambda \geq 0$.*

Proof. For any solution $\boldsymbol{y}^3 \in S^3$, $\frac{\partial Z^3(\lambda, \boldsymbol{y}^3)}{\partial \lambda} = -(\sum_{j=1}^{n} b_j y_j^3 - D) \leq 0$. Since the multiplier λ does not affect the feasible domain of P3, $Z^3(\lambda)$ is non-increasing.

Let $g(\lambda)$ be the equation of the line joining two points $\Big(\lambda_0, Z^3(\lambda_0)\Big)$ and $\Big(\lambda_1, Z^3(\lambda_1)\Big)$ in the plane, where $0 \leq \lambda_0 < \lambda_1$. This equation is given by

$$g(\lambda) = Z^3(\lambda_0) - (\lambda - \lambda_0) \frac{Z^3(\lambda_0) - Z^3(\lambda_1)}{\lambda_1 - \lambda_0}.$$

To prove the concavity, we show that inequality $g(\lambda) \leq Z^3(\lambda)$ holds for any $\lambda \in [\lambda_0, \lambda_1]$. As $g(\lambda_0) = Z^3(\lambda_0)$ and $g(\lambda_1) = Z^3(\lambda_1)$, the result is trivial at the limits of the interval $[\lambda_0, \lambda_1]$. Now, let us consider the case $\lambda_0 < \lambda < \lambda_1$. Let $\boldsymbol{y}^\lambda$ be an optimal solution to P3 for multiplier λ. For multiplier λ^0, the objective value $Z^3(\lambda^0, \boldsymbol{y}^\lambda)$ of this solution, which can be obtained from $Z^3(\lambda)$ by replacing the reimbursement rate λ by λ_0, is an upper bound on the optimal value $Z^3(\lambda_0)$. Furthermore, the demand constraint (2.2) implies that $(\sum_{j=1}^{n} b_j y_j^\lambda - D) \geq 0$.

Hence, we have:

$$
\begin{aligned}
& Z^3(\lambda_0) \leq Z^3(\lambda_0, \boldsymbol{y}^\lambda) \\
\iff \quad & Z^3(\lambda_0) \leq Z^3(\lambda) - \Big(\sum_{j=1}^{n} b_j y_j^\lambda - D\Big)(\lambda_0 - \lambda) \\
\iff \quad & \frac{Z^3(\lambda_0) - Z^3(\lambda)}{(\lambda - \lambda_0)} \leq \Big(\sum_{j=1}^{n} b_j y_j^\lambda - D\Big)
\end{aligned}
$$

Similarly, since $Z^3(\lambda_1)$ is the optimal value of P3 for multiplier λ_1, the objective value $Z^3(\lambda_1, \boldsymbol{y}^\lambda)$ is greater than or equal to $Z^3(\lambda_1)$. By applying the same steps as above, we obtain the following inequality.

$$
\frac{Z^3(\lambda) - Z^3(\lambda_1)}{(\lambda_1 - \lambda)} \geq \Big(\sum_{j=1}^{n} b_j y_j^\lambda - D\Big)
$$

The two previous inequalities lead to the expected lower bound on $Z^3(\lambda)$.

$$
\begin{aligned}
& \frac{Z^3(\lambda_0) - Z^3(\lambda)}{(\lambda - \lambda_0)} \leq \frac{Z^3(\lambda) - Z^3(\lambda_1)}{(\lambda_1 - \lambda)} \\
\iff \quad & Z^3(\lambda) \geq \Big(\frac{\lambda - \lambda_1}{\lambda_0 - \lambda_1}\Big) Z^3(\lambda_0) - \Big(\frac{\lambda - \lambda_0}{\lambda_0 - \lambda_1}\Big) Z^3(\lambda_1) \\
\iff \quad & Z^3(\lambda) \geq Z^3(\lambda_0) - (\lambda - \lambda_0) \frac{Z^3(\lambda_0) - Z^3(\lambda_1)}{\lambda_1 - \lambda_0} \\
\iff \quad & Z^3(\lambda) \geq g(\lambda)
\end{aligned}
$$

□

3.3. Heuristic search

This section presents `P3Search`, the algorithm we use to select the set L of multipliers λ for which model P3 is solved in the heuristic phase of KTA. Given an optimal solution $\boldsymbol{y}^\lambda$ to P3 for each multiplier $\lambda \in L$, we propose new lower and upper bounds, which are detailed in Sections 3.3.1 to 3.3.4.

Adding multipliers to L generally improves these bounds, but solving a high number of KPs is computationally cumbersome. In order to optimize the overall performance of KTA, the goal of `P3Search` is thus to select a small set of multipliers L that nevertheless makes the bounds relatively tight.

The algorithm starts by solving P3 for $\lambda = \min\Big\{e_s,\ \max_{j\in\{1,\dots,n\}}\{c_j\}\Big\}$. In the subsequent steps, λ is set to the highest unit cost among the nodes that are selected in the previous solution. This process is repeated until a multiplier is visited twice, λ exceeds the linearized

Algorithm 1 `P3Search`

$L \leftarrow \emptyset$
$\lambda \leftarrow \min\left\{e_s,\ \max_{j\in\{1,\dots,n\}}\{c_j\}\right\}$
do
  $\boldsymbol{y}^\lambda \leftarrow \arg\min_{\boldsymbol{y}\in S^3}\left\{Z^3(\lambda,\boldsymbol{y})\right\}$
  $L \leftarrow L \cup \{\lambda\}$
  $\lambda \leftarrow \max_{j\in\{1,\dots,n\}}\{y_j^\lambda c_j\}$
while ($\lambda \notin L$ and $\lambda \leq e_s$ and $\sum_{j=1}^n b_j y_j^\lambda > D$)

cost of the split node, or a solution without a partial node is found. Additional comments are given in Section 3.3.5 on the choice of the initial multiplier value.

3.3.1. Upper bound $Z_{\mathrm{UB}}^{\mathrm{P3}}$ on the optimal value

An upper bound on the optimal value $Z^1{=}Z^2$ of the SSFCTP is obtained from the solutions $\boldsymbol{y}^\lambda$ computed in the heuristic search. It is given by $Z_{\mathrm{UB}}^{\mathrm{P3}} = \min_{\lambda\in L}\{Z_{\mathrm{UB}}^{\mathrm{P3}}(\lambda)\}$, where

$$Z_{\mathrm{UB}}^{\mathrm{P3}}(\lambda) = \begin{cases} Z^1(\tilde{f}_{3,1}(\boldsymbol{y}^\lambda)) & \text{if } \boldsymbol{y}^\lambda \in \tilde{S}^3, \\ \sum_{j=1}^n (c_j b_j + f_j) y_j^\lambda - (c_{p(\boldsymbol{y}^\lambda)} b_{p(\boldsymbol{y}^\lambda)} + f_{p(\boldsymbol{y}^\lambda)}) & \text{otherwise.} \end{cases}$$

If $\boldsymbol{y}^\lambda{\in}\tilde{S}^3$, then $\tilde{f}_{3,1}(\boldsymbol{y}^\lambda)$ is a feasible solution to P1 and its objective value $Z^1(\tilde{f}_{3,1}(\boldsymbol{y}^\lambda))$ is therefore an upper bound on Z^1. Otherwise, inequality (3.5) is not respected, which implies that the total offer of the nodes of set $N(\boldsymbol{y}^\lambda) \setminus \{p(\boldsymbol{y}^\lambda)\}$ is greater than or equal to D. Their total cost is thus a valid upper bound on Z^1, since all the costs c_j and f_j are assumed to be nonnegative.

3.3.2. Lower bounds $C_{\min}^{\mathrm{P3}}$ on the partial node's unit cost

Two lower bounds $C_{\min}^{\mathrm{P3}_1}$ and $C_{\min}^{\mathrm{P3}_2}$ on the variable cost of the partial node of any solution of $\tilde{S}^1$ that may improve the incumbent value $Z_{\mathrm{UB}}^{\mathrm{P3}}$ are now introduced. By excluding potential partial nodes, they contribute to reducing the number of knapsack subproblems that need to be solved during the exact phase of KTA.

Let $\boldsymbol{y}^\lambda$ be the optimal solution to P3 for a given multiplier $\lambda \in L$. If $\boldsymbol{y}^\lambda \in \tilde{S}^3$ and $\lambda{=}c_{p(\boldsymbol{y}^\lambda)}$, then $Z_{\mathrm{UB}}^{\mathrm{P3}}(\lambda){=}Z^3(\lambda)$. In this case, by Proposition 6, the objective value of a solution of $\tilde{S}^1$ that includes a partial node p with a unit cost $c_p{\leq}\lambda$ cannot be less than $Z_{\mathrm{UB}}^{\mathrm{P3}}(\lambda)$. A first lower bound is thus given by $C_{\min}^{\mathrm{P3}_1}{=}\max M$, where $M{=}\{\lambda{\in}L : \boldsymbol{y}^\lambda{\in}\tilde{S}^3$ and $\lambda{=}c_{p(\boldsymbol{y}^\lambda)}\}$. In the case where M is empty, this bound cannot be computed and $C_{\min}^{\mathrm{P3}_1}{=}0$.

If $\max L{>}C_{\min}^{\mathrm{P3}_1}$, then another bound may improve $C_{\min}^{\mathrm{P3}_1}$. Consider the multipliers $\lambda_0{=}\max\left\{\lambda{\in}L\ :\ Z^3(\lambda){>}Z_{\mathrm{UB}}\right\}$ and $\lambda_1{=}\min\left\{\lambda{\in}L\ :\ Z^3(\lambda){<}Z_{\mathrm{UB}}\right\}$, where Z_{UB} is the best known upper bound on Z^1. Let $g(\lambda)$ be the equation of the line joining points $(\lambda_0, Z^3(\lambda_0))$

and $(\lambda_1, Z^3(\lambda_1))$, and let $\lambda^* \in (\lambda_0, \lambda_1)$ be the multiplier such that $g(\lambda^*)=Z_{\text{UB}}$. By the concavity of $Z^3(\lambda)$, this multiplier respects $Z^3(\lambda^*)\geq g(\lambda^*)$. Therefore, it follows from Proposition 6 that the objective value of a solution in which the unit cost of the partial node is less than or equal to λ^* cannot be less than Z_{UB}. The resulting bound is given by

$$C_{\min}^{\text{P3}_2} = \begin{cases} \lambda^* = \lambda_0 + (\lambda_1 - \lambda_0)\frac{Z^3(\lambda_0)-Z_{\text{UB}}}{Z^3(\lambda_0)-Z^3(\lambda_1)} & \text{if } \max L > C_{\min}^{\text{P3}_1}, \\ 0 & \text{otherwise.} \end{cases}$$

We denote the best lower bound on the partial node's unit cost that is extracted from the solutions collected in `P3Search` by $C_{\min}^{\text{P3}}=\max\{C_{\min}^{\text{P3}_1}, C_{\min}^{\text{P3}_2}\}$.

3.3.3. Lower bound $Z_{\text{LB}(p)}^{\text{P3}}$ given that node p is partial

A lower bound on the objective value of any solution of $\tilde{S}^1$ in which a specific node $p \in \{1,\ldots,n\}$ is partial can also be obtained. This bound is defined as:

$$Z_{\text{LB}(p)}^{\text{P3}} = \begin{cases} Z^3(c_p) & \text{if } c_p \in L, \\ Z^3(\lambda_0) - (c_p - \lambda_0)\frac{Z^3(\lambda_0)-Z^3(\lambda_1)}{\lambda_1-\lambda_0} & \text{otherwise,} \end{cases}$$

where $\lambda_0=\max\{\lambda\in L : \lambda<c_p\}$ and $\lambda_1=\min\{\lambda\in L : \lambda>c_p\}$.

The validity of the lower bound $Z^3(c_p)$ is directly given by Proposition 6. If $c_p \notin L$, using the concavity of function $Z^3(\lambda)$, then $Z^3(c_p)$ is approximated from below by interpolating points $(\lambda_0, Z^3(\lambda_0))$ and $(\lambda_1, Z^3(\lambda_1))$.

3.3.4. Lower and upper bounds $\boldsymbol{LB}_{x_p}^{\text{P3}}$ and $\boldsymbol{UB}_{x_p}^{\text{P3}}$ on x_p given that p is partial

The last bounds that are computed in this step of KTA are on the number of units that can be ordered from node p given that it is partial.

Let $(\boldsymbol{x}^1, \boldsymbol{y}^1) \in \tilde{S}^1$ be a solution whose critical node is given by $p(\boldsymbol{y}^1)=p$. It follows from the definition of $\tilde{S}^1$ that $\boldsymbol{y}^1$ is also an element of $\tilde{S}^3$. Hence, by equation (3.6), the inequality $Z^1(\boldsymbol{x}^1, \boldsymbol{y}^1)<Z_{\text{UB}}$ can be written as

$$Z^3(\lambda, \boldsymbol{y}^1) + \Big(\sum_{j=1}^{n} b_j y_j^1 - D\Big)(\lambda - c_p) < Z_{\text{UB}}$$

for any multiplier $\lambda\geq 0$. Since $Z^3(\lambda, \boldsymbol{y}^1)$ is an upper bound on $Z^3(\lambda)$ by construction, the inequality

$$Z^3(\lambda) + \Big(\sum_{j=1}^{n} b_j y_j^1 - D\Big)(\lambda - c_p) < Z_{\text{UB}}$$

must hold for each $\lambda \in L$ for $(\boldsymbol{x}^1, \boldsymbol{y}^1)$ to improve the incumbent solution to P1. This is equivalent to the following conditions.

$$\sum_{j=1}^{n} b_j y_j^1 - D > \frac{Z^3(\lambda) - Z_{\text{UB}}}{c_p - \lambda} \qquad \forall \lambda \in L : \lambda < c_p,$$
$$\sum_{j=1}^{n} b_j y_j^1 - D < \frac{Z^3(\lambda) - Z_{\text{UB}}}{c_p - \lambda} \qquad \forall \lambda \in L : \lambda > c_p.$$

As any solution of $\tilde{S}^1$ includes at most one partial node, the number of units taken from p is directly given by $x_p = b_p - (\sum_{j=1}^{n} b_j y_j^1 - D)$. Also, the excess supply is always an integer. This leads to the following bounds.

$$LB_{x_p}^{\text{P3}} = b_p - \min_{\{\lambda \in L : \lambda > c_p\}} \left\{ \left\lceil \frac{Z^3(\lambda) - Z_{\text{UB}}}{c_p - \lambda} - 1 \right\rceil \right\},$$
$$UB_{x_p}^{\text{P3}} = b_p - \max_{\{\lambda \in L : \lambda < c_p\}} \left\{ \left\lfloor \frac{Z^3(\lambda) - Z_{\text{UB}}}{c_p - \lambda} + 1 \right\rfloor \right\}.$$

When considering solutions where node p is partial, we can therefore narrow down our search to solutions in which $LB_{x_p}^{\text{P3}} \leq x_p \leq UB_{x_p}^{\text{P3}}$.

3.3.5. Choice of initial multiplier λ in `P3Search`

Initial experiments have shown that taking $\lambda = \min\left\{e_s,\ \max_{j \in \{1,\ldots,n\}}\{c_j\}\right\}$ as the initial multiplier generally minimizes the overall execution time of KTA. This is due to the fact that it is frequent for the equality $\sum_{j=1}^{n} b_j y_j^{\lambda} = D$ to hold although this value of λ tends to overestimate the unit cost $c_{p(\boldsymbol{y}^{\lambda})}$ of the critical node and therefore favors solutions with excess offer. If the total offer of the nodes of set $N(\boldsymbol{y}^{\lambda})$ is equal to D for this multiplier, and if $\max_{j \in \{1,\ldots,n\}}\{c_j\} \leq e_s$, then $C_{\min}^{\text{P3}} = \max_{j \in \{1,\ldots,n\}}\{c_j\}$. This means that any solution that includes a partial node can be ignored and $\tilde{f}_{3,1}(\boldsymbol{y}^{\lambda})$ is therefore optimal. This case is frequent enough and the execution time reduction associated with this early proof of optimality sufficiently large to justify this choice of initial multiplier.

Another reasonable choice would be $\lambda = \max_{j \in \{1,\ldots,s\}}\{c_j\}$. This initial multiplier generally leads to identifying the best reachable upper bound $Z_{\text{UB}}^{\text{P3}}$ in fewer iterations, since $\max_{j \in \{1,\ldots,s\}}\{c_j\}$ is usually a better approximation of the maximum unit cost among the nodes that are used in an optimal solution than $\min\left\{e_s,\ \max_{j \in \{1,\ldots,n\}}\{c_j\}\right\}$. However, this advantage generally does not compensate for the weaker bound $C_{\min}^{\text{P3}}$ in the overall performance of the algorithm.

3.4. Dominance relation

Some of the most expensive calculations of KTA are performed after selecting a specific partial node and must be repeated for each node that may still be partial in an optimal

solution. Consequently, the efficiency of the algorithm relies in good part on our ability to exclude a significant proportion of candidates before performing these calculations. For the remaining nodes $p \in \{1,\ldots,n\}$, restricting the number of units x_p they can supply given that they are partial to a narrow interval $[LB_{x_p}, UB_{x_p}]$ can still reduce the computing time. The following dominance relation, by seeking to improve the bounds of Section 3.3.4, serves both of these purposes.

Definition 8. *If q is a node such that $q \succ p$ and $f_q \leq f_p$, then p is said to be dominated by q as a partial node when supplying any number of units $x_p \leq LB^{\mathrm{Dom}}_{x_p}(q)$, where*

$$LB^{\mathrm{Dom}}_{x_p}(q) = \begin{cases} b_q & \text{if } c_q = c_p, \\ \min\left\{\frac{f_p - f_q}{c_q - c_p}, b_q\right\} & \text{otherwise.} \end{cases}$$

Indeed, for any solution $(\boldsymbol{x}^1,\boldsymbol{y}^1) \in \tilde{S}^1$ in which the partial node p supplies no more than $LB^{\mathrm{Dom}}_{x_p}(q)$ units, p can be replaced by the unused node q to obtain a feasible solution with an objective value of at most $Z^1(\boldsymbol{x}^1,\boldsymbol{y}^1)$. This leads to the following lower bound on x_p given that p is partial.

Proposition 9. *Let $LB^{\mathrm{Dom}}_{x_p}$ be defined as follows.*

$$LB^{\mathrm{Dom}}_{x_p} = \max_{\left\{q \in \{1,\ldots,n\} : q \succ p \text{ and } f_q \leq f_p\right\}} \left\{\left\lfloor LB^{\mathrm{Dom}}_{x_p}(q) + 1 \right\rfloor\right\} \tag{3.7}$$

There exists an optimal solution in $\tilde{S}^1$ in which p is not partial or $x_p \geq LB^{\mathrm{Dom}}_{x_p}$.

Proof. Let $(\boldsymbol{x}^1, \boldsymbol{y}^1) \in \tilde{S}^1$ be a solution that contains a partial node p supplying $x^1_p \leq LB^{\mathrm{Dom}}_{x_p}(q)$ units, where $q \succ p$ respects $f_q \leq f_p$. Since node p is partial and $q \succ p$, the definition of $\tilde{S}^1$ implies that $x^1_q = y^1_q = 0$. From there, a modified solution $(\boldsymbol{x}', \boldsymbol{y}')$ can be built from $(\boldsymbol{x}^1, \boldsymbol{y}^1)$ by setting $x'_p = y'_p = 0$, $y'_q = 1$, and $x'_q = x^1_p$. This modification corresponds to using x^1_p units from node q instead of node p, which becomes unused. Since $b_q \geq LB^{\mathrm{Dom}}_{x_p}(q)$ by definition of the dominance relation, and since $LB^{\mathrm{Dom}}_{x_p}(q) \geq x^1_p$ by hypothesis, $(\boldsymbol{x}', \boldsymbol{y}')$ remains a feasible solution to P1. It also respects the definition of $\tilde{S}^1$, as q is by construction the critical node associated with $\boldsymbol{y}'$.

From there, it suffices to show that $Z^1(\boldsymbol{x}', \boldsymbol{y}') \leq Z^1(\boldsymbol{x}^1, \boldsymbol{y}^1)$ to conclude that $(\boldsymbol{x}^1, \boldsymbol{y}^1)$ does not need to be considered to identify an optimal solution. Since the changes made to $(\boldsymbol{x}^1, \boldsymbol{y}^1)$ modify the objective value by $f_q - f_p + x^1_p(c_q - c_p)$, this is done by proving that this value is nonpositive.

First, if $c_q = c_p$, then $f_q - f_p + x^1_p(c_q - c_p) = f_q - f_p$. Since $f_q \leq f_p$ by hypothesis, this value is indeed nonpositive.

Since $q \succ p$, the other possibility is that $c_q > c_p$. In this case, it follows from the definition of the dominance relation that $LB^{\mathrm{Dom}}_{x_p}(q) \leq \frac{f_p - f_q}{c_q - c_p}$. Furthermore, $x^1_p \leq LB^{\mathrm{Dom}}_{x_p}(q)$ by

hypothesis. Therefore, replacing x_p by its upper bound $\frac{f_p-f_q}{c_q-c_p}$ allows us to conclude that $f_q-f_p+x_p^1(c_q-c_p) \leq f_q-f_p+\frac{f_p-f_q}{c_q-c_p}(c_q-c_p) = 0$.

The lower bound $LB_{x_p}^{\text{Dom}}$ corresponds to the smallest number of units x_p for which the previous construction does not prove the existence of a solution $(\boldsymbol{x}', \boldsymbol{y}') \in \tilde{S}^1$ with critical node $q \succ p$ such that $Z^1(\boldsymbol{x}', \boldsymbol{y}') \leq Z^1(\boldsymbol{x}^1, \boldsymbol{y}^1)$. □

3.5. Strong linear relaxation

By taking advantage of the theory developed in Section 3.2, the classical linear relaxation of the SSFCTP can be adapted to pursue the same objective as the previous section. We propose a stronger linear relaxation that is computationally more demanding than the dominance relation but leads to a better lower bound on x_p given that node p is partial. It also offers an upper bound on this same variable, as well as a lower bound $Z_{\text{LB}(p)}^{\text{LP}}$ that aims to improve the one described in Section 3.3.3.

3.5.1. Lower bound $Z_{\text{LB}(p)}^{\text{LP}}$ given that node p is partial

Given a partial node p and previously computed bounds $LB_{x_p} \leq x_p \leq UB_{x_p}$, the fixed cost f_p and the cost $c_p LB_{x_p}$ of the minimal number of units it has to supply must first be paid. Then, each node $i \succ p$ must be fixed as unused for the solution to be an element of $\tilde{S}^1$. Finally, the remaining demand must be satisfied using the set of nodes $\{i_1, \ldots, i_{\tilde{n}}\} = \{i \in \{1, \ldots, n\} : i \prec p\}$ and the $UB_{x_p} - LB_{x_p}$ units that can still be supplied by p. The resulting problem corresponds to a SSFCTP with demand $\tilde{D} = D - LB_{x_p}$ defined on the set of suppliers $\{i_1, \ldots, i_{\tilde{n}}, i_{\tilde{n}+1}\}$, where $b_{i_{\tilde{n}+1}} = UB_{x_p} - LB_{x_p}$, $f_{i_{\tilde{n}+1}} = 0$ and $c_{i_{\tilde{n}+1}} = c_p$. If $\sum_{j=1}^{\tilde{n}+1} b_{i_j} < \tilde{D}$, then the demand constraint cannot be respected. This means that fixing node p as partial and imposing constraint $LB_{x_p} \leq x_p \leq UB_{x_p}$ cannot lead to a solution in $\tilde{S}^1$. Otherwise, a lower bound $\tilde{Z}_{\text{LB}}^{\text{LP}}$ on the optimal value $\tilde{Z}^1$ of the SSFCTP subproblem can be obtained by solving its LP relaxation.

The following value is therefore a valid lower bound on the objective value of any solution $(\boldsymbol{x}^1, \boldsymbol{y}^1) \in \tilde{S}^1$ respecting $LB_{x_p} \leq x_p \leq UB_{x_p}$.

$$Z_{\text{LB}(p)}^{\text{LP}} = \begin{cases} f_p + c_p LB_{x_p} + \tilde{Z}_{\text{LB}}^{\text{LP}} & \text{if } \sum_{j=1}^{\tilde{n}+1} b_{i_j} \geq \tilde{D}, \\ +\infty & \text{otherwise.} \end{cases}$$

It follows that node p no longer needs to be considered as a potential partial node if $Z_{\text{LB}(p)}^{\text{LP}} \geq Z_{\text{UB}}$, where Z_{UB} is the incumbent objective value.

3.5.2. Lower and upper bounds $\boldsymbol{LB}_{x_p}^{\text{LP}}$ and $\boldsymbol{UB}_{x_p}^{\text{LP}}$ on x_p given that p is partial

When $Z_{\text{LB}(p)}^{\text{LP}} < Z_{\text{UB}}$, the number of units node p can supply given that it is partial can be bounded based on the optimal solution $\tilde{\boldsymbol{x}}$ of the strong linear relaxation. The lower

and upper bounds $LB_{x_p}^{\mathrm{LP}}$ and $UB_{x_p}^{\mathrm{LP}}$ are the lowest and highest number of units that can be ordered from node p without the optimal value of the subproblem to reach Z_{UB}.

Without lost of generality, it is assumed that nodes $i_1,\ldots,i_{\tilde{n}}$ are sorted in non-decreasing order of linearized cost. The split node of the strong linear relaxation is then denoted by $\tilde{s}=\max\left\{j\in\{1,\ldots,\tilde{n}\} : \tilde{x}_{i_j}\geq 1\right\}$.

The lower bound $LB_{x_p}^{\mathrm{LP}}$ is obtained by taking the cheapest remaining units from nodes $i_{\tilde{s}}, i_{\tilde{s}+1},\ldots,i_{\tilde{n}}$ to reduce the number of units $LB_{x_p}+\tilde{x}_{i_{\tilde{n}+1}}$ that are ordered from the partial node. Each time a unit is supplied by a node $i\in\{i_{\tilde{s}}, i_{\tilde{s}+1},\ldots,i_{\tilde{n}}\}$ instead of $i_{\tilde{n}+1}$, the objective value increases by (e_i-c_p). This process is repeated until the objective value of the solution reaches Z_{UB}, all the nodes except $i_{\tilde{n}+1}$ are used to capacity, or $x_{i_{\tilde{n}+1}}=0$. The resulting lower bound is $LB_{x_p}^{\mathrm{LP}}=LB_{x_p}+x_{i_{\tilde{n}+1}}$, where $x_{i_{\tilde{n}+1}}$ is the smallest number of units for which the value of the solution remains less than Z_{UB}.

Similarly, $UB_{x_p}^{\mathrm{LP}}$ is computed by removing from the solution the most expensive units supplied by nodes $i_{\tilde{s}}, i_{\tilde{s}-1},\ldots,i_1$ while increasing $x_{i_{\tilde{n}+1}}$ to $\tilde{x}_{i_{\tilde{n}+1}}+1$, $\tilde{x}_{i_{\tilde{n}+1}}+2$, and so on. In this case, each time a unit is supplied by the partial instead of node $i\in\{i_{\tilde{s}}, i_{\tilde{s}-1},\ldots,i_1\}$, the objective value increases by (c_p-e_i). This process is repeated until the objective value reaches Z_{UB}, all the nodes but $i_{\tilde{n}+1}$ are unused, or $x_{i_{\tilde{n}+1}}=b_{i_{\tilde{n}+1}}$. The resulting upper bound is $UB_{x_p}^{\mathrm{LP}}=LB_{x_p}+x_{i_{\tilde{n}+1}}$, where $x_{i_{\tilde{n}+1}}$ is the largest number of units for which the objective value is less than Z_{UB}.

It is worth noticing that $LB_{x_p}^{\mathrm{LP}}$ and $UB_{x_p}^{\mathrm{LP}}$ can only improve the previous lower and upper bounds LB_{x_p} and UB_{x_p} if $\tilde{x}_{i_{\tilde{n}+1}}>0$ and $\tilde{x}_{i_{\tilde{n}+1}}<b_{i_{\tilde{n}+1}}$, respectively.

4. Filtering

The various bounds described in the previous section generally restrict the set of nodes that can be partially used to a very small subset of suppliers $P\subseteq\{1,\ldots,n\}$. From there, a knapsack subproblem will need to be solved for each of the remaining candidates $p\in P$. To reduce as much as possible the number of items in each of these subproblems, a filtering of the non-partial nodes is carried out beforehand.

4.1. Linear filtering

As mentioned by Klose (2008), not using a node $j<s$ increases the LP bound by at least $b_j(e_s-e_j)$, since b_j units must then be taken from suppliers $i\geq s$ instead of supplier j. Hence, if $Z_{\mathrm{LB}}^{\mathrm{LP}}+b_j(e_s-e_j)>Z_{\mathrm{UB}}$, then node j cannot be unused in an optimal solution. Similarly, $Z_{\mathrm{LB}}^{\mathrm{LP}}$ increases by at least $b_j(e_j-e_s)$ when node $j>s$ is complete. Therefore, if $Z_{\mathrm{LB}}^{\mathrm{LP}}+b_j(e_j-e_s)>Z_{\mathrm{UB}}$, then j cannot be complete in an optimal solution.

After designating a specific partial node, these rules allow us to fix non-partial nodes $j<s$ as complete and non-partial nodes $j>s$ as unused.

4.2. Restricted filtering

The solutions that can be obtained during the exact phase of KTA always contain a partial node $p \in P$. Denoting by q the maximal element of the strictly totally ordered set $(P, \prec)$, it follows from the definition of $\tilde{S}^1$ that any node $j \succ q$ must be unused in the solution of each remaining subproblem. This leads to the following reformulation of the LP relaxation $\mathrm{P1}^{\mathrm{LP}}$.

$$[\mathrm{P1}^{\mathrm{LP}}(q)] \quad Z^{\mathrm{LP}}_{\mathrm{LB}}(q) = \min \sum_{j=1}^{n} e_j x_j \tag{4.1}$$

$$\text{s.t. } (1.2),$$

$$0 \leq x_j \leq \Big(1 - \mathbb{1}(j \succ q)\Big) b_j \quad \text{for } j = 1, \ldots, n. \tag{4.2}$$

This problem corresponds to the linear program $\mathrm{P1}^{\mathrm{LP}}$ defined on the subset of suppliers $Q = \Big\{j \in \{1, \ldots, n\} : j \prec q \text{ or } j = q\Big\}$. Consequently, its optimal solution is obtained as explained in Section 3.1.1. Applying the linear filtering of Section 4.1 to this restricted LP relaxation leads to the so-called *restricted filtering*.

4.3. Strong restricted filtering

Let $\boldsymbol{x}^*$ be the optimal solution to $\mathrm{P1}^{\mathrm{LP}}(q)$ and $s^q \in Q$ its split node, defined as follows:

$$\sum_{\substack{i=1 \\ i \in Q}}^{s^q - 1} b_i < D \leq \sum_{\substack{i=1 \\ i \in Q}}^{s^q} b_i.$$

The linear filtering of Section 4.1 is already strengthened in Section 4.2 by forcing all the nodes that are not part of set Q to be unused. However, an even stronger bound can be calculated for each node $j \in Q$ when the remaining offer $b_{s^q} - x^*_{s^q}$ on the split node or the number $x^*_{s^q}$ of units it supplies is insufficient for b_j more units to be taken (if $j < s^q$) or unused (if $j > s^q$) from s^q. In the case where $j < s^q$, fixing x_j to 0 increases the bound $Z^{\mathrm{LP}}(q)$ by

$$k^0_j = -b_j e_j + (b_{s^q} - x^*_{s^q}) e_{s^q} + \sum_{\substack{i=s^q+1 \\ i \in Q}}^{t^j - 1} b_i e_i + \Big(b_j - (b_{s^q} - x^*_{s^q}) - \sum_{\substack{i=s^q+1 \\ i \in Q}}^{t^j - 1} b_i\Big) e_{t^j}.$$

After removing the b_j units that were supplied by node j from the solution, the $b_{s^q} - x^*_{s^q}$ remaining units of the split node s^q are used, after what the residual demand is fulfilled using nodes $s^q + 1, \ldots, t^j \in Q$. The last node that is added to the solution, denoted by t^j, corresponds to the split node of model $\mathrm{P1}^{\mathrm{LP}}(q)$ given that $x_j = 0$. If the total offer of the nodes of set $Q \setminus \{j\}$ is less than D, then this problem is infeasible, and k^0_j is fixed to $+\infty$.

Analogously, for a node $j > s^q$ such that $b_j > x^*_{s^q}$, reducing the number of units that are supplied by nodes $s^q, s^q - 1, \ldots r^j \in Q$ by b_j in order to use node j completely increases the

lower bound $Z^{\mathrm{LP}}(q)$ by

$$k_j^1 = b_j e_j - x^*_{s^q} e_{s^q} - \sum_{\substack{i=r^j+1 \\ i\in Q}}^{s^q-1} b_i e_i - \Big(b_j - x^*_{s^q} - \sum_{\substack{i=r^j+1 \\ i\in Q}}^{s^q-1} b_i\Big) e_{r^j},$$

where r^j, which is the last node to supply a reduced number of units, respects

$$\sum_{\substack{i=r^j+1 \\ i\in Q}}^{s^q-1} b_j < b_j - x^*_{s^q} \leq \sum_{\substack{i=r^j \\ i\in Q}}^{s^q-1} b_j.$$

In summary, $Z^{\mathrm{LP}}(q)+k_j^0$ corresponds to the optimal value of model $\mathrm{P1}^{\mathrm{LP}}(q)$ when supplier $j<s^q$ is unused whereas $Z^{\mathrm{LP}}(q)+k_j^1$ is the optimal value of this same problem when supplier $j>s^q$ is complete. Let us define the two following sets of nodes:

$$R = \Big\{j \in \{1,\ldots,s^q-1\} \cap Q : Z^{\mathrm{LP}}(q) + k_j^0 > Z_{\mathrm{UB}}\Big\},$$
$$T = \Big\{j \in \{s^q+1,\ldots,n\} \cap Q : Z^{\mathrm{LP}}(q) + k_j^1 > Z_{\mathrm{UB}}\Big\}.$$

After fixing a partial node $p \in P$, the problem can thus be reduced by fixing each node $j \in R \setminus \{p\}$ as complete and each node $j \in T \setminus \{p\}$ as unused. As will be seen in Section 7, this method filters the vast majority of suppliers for all the classes of problems we consider.

5. Exact method

After solving P3 for each multiplier $\lambda \in L$ as described in Section 3.2, if the incumbent value Z_{UB} is such that $Z_{\mathrm{UB}} \leq Z^3(\lambda)$ for at least one value $\lambda \geq 0$, then it follows from Proposition 6 that any solution that can improve the current upper bound must contain a partial node. The exact phase of KTA thus consists in solving P1 given that node p is partial for each remaining candidate $p \in P$ concludes the algorithm. This approach owes its efficiency to the very small cardinality of P at this point and to the fact that the filtering methods of Section 4 fix the vast majority of the decision variables for each subproblem.

5.1. Knapsack transformation

Model P1 can be reduced to a min-KP when it is given that a specific node $p \in P$ is partial. After paying for the fixed cost of the partial node and the minimal number of units LB_{x_p} it must supply, the problem reduces to selecting which nodes $j \in \{1,\ldots,p-1,p+1,n\}$ to use completely and how many additional units to take from p in order to fulfill the residual demand $D-LB_{x_p}$ at the lowest possible cost. In the following min-KP, each item $j \in \{1,\ldots,p-1,p+1,\ldots,n\}$ costs $u_j=c_jb_j+f_j$ and weighs b_j. The residual offer $\hat{b}_p=UB_{x_p}-LB_{x_p}$ of the partial node is decomposed into $d^p=\lfloor \log_2(\hat{b}_p) \rfloor+1$ items $j \in \{n+1,\ldots,n+d^p\}$ whose weights b_j are respectively $2^0,2^1,\ldots,2^{d^p-2}$ and $\hat{b}_p-(2^{d^p-1}-1)$, and whose costs are given

by $u_j{=}c_p b_j$. This decomposition method, which has been introduced by Martello and Toth (1990) to transform the bounded knapsack problem into a standard KP, makes it possible to use any number of additional units $\{0,1,\ldots,\hat{b}_p\}$ from the partial node by selecting one of the 2^{d^p} possible subsets of the items $\{n{+}1,\ldots,n{+}d^p\}$.

$$[\mathrm{P1}(p)] \quad Z^1(p) = (c_p LB_{x_p} + f_p) + \min \sum_{\substack{j=1 \\ j\neq p}}^{n+d^p} u_j y_j \tag{5.1}$$

$$\text{s.t.} \sum_{\substack{j=1 \\ j\neq p}}^{n+d^p} b_j y_j = D - LB_{x_p}, \tag{5.2}$$

$$y_j \in \{0,1\} \qquad \text{for } j = 1,\ldots,p{-}1,p{+}1,\ldots,n+d^p. \tag{5.3}$$

Solving P1(p) for each node $p \in P$ constitutes the exact phase of KTA and marks the end of the algorithm. Note that restricting our search to solutions of set $\tilde{S}^1$ allows us to fix $y_j{=}0$ for each node $j \in \{1,\ldots,p{-}1,p{+}1,\ldots,n\}$ such that $j \succ p$.

6. Outline of KTA

This section summarizes the overall steps performed by our algorithm. Knapsack subproblems are solved in both the heuristic and exact phases. To do so, COMBO, the state-of-the-art algorithm presented in Martello et al. (1999), is used with its default parameters.

6.1. Heuristic phase

The execution of the algorithm begins with the heuristic phase, which essentially consists of calculating the bounds given in Section 3.

Step H1).

(1) Calculate the lower and upper bounds $Z^{\mathrm{LP}}_{\mathrm{LB}}$ and $Z^{\mathrm{G}}_{\mathrm{UB}}$. Set the global lower and upper bounds Z_{LB} and Z_{UB} to $Z^{\mathrm{LP}}_{\mathrm{LB}}$ and $Z^{\mathrm{G}}_{\mathrm{UB}}$, respectively.

(2) If $Z_{\mathrm{LB}}{=}Z_{\mathrm{UB}}$, then the greedy solution is optimal. Terminate the algorithm.

The time complexity of this step is $\mathcal{O}(n)$.

Step H2).

(1) Apply `P3Search`. Terminate the execution after five iterations if the halting conditions have not been met earlier.

(2) Compute the upper bound $Z^{\mathrm{P3}}_{\mathrm{UB}}$. Update Z_{UB} and the incumbent solution. If $Z_{\mathrm{UB}} \leq Z^3(\lambda_0)$, where $\lambda_0 = \min L$, then proceed to point 3. Otherwise, solve P3 for $\lambda{=}0$ and update the incumbent solution accordingly. Since $Z^{\mathrm{P3}}_{\mathrm{UB}}(0) \leq Z^3(0)$ by construction, it follows from Proposition 6 that solutions without a partial node can now be ignored.

(3) Calculate the lower bound $C^{\mathrm{P3}}_{\min}$ on the partial node's unit cost.

(4) Let $P=\left\{j\in\{1,\ldots,n\} : c_j>C^{\mathrm{P3}}_{\min}\right\}$ be the set of nodes that could strictly improve the incumbent solution when fixed as partial. For each node $p \in P$, calculate the lower bound $Z^{\mathrm{P3}}_{\mathrm{LB}(p)}$ and initialize $Z_{\mathrm{LB}(p)}$ to this value. Remove p from P if $Z_{\mathrm{LB}(p)}\geq Z_{\mathrm{UB}}$. Otherwise, set the bounds LB_{x_p} and UB_{x_p} to $LB^{\mathrm{P3}}_{x_p}$ and $UB^{\mathrm{P3}}_{x_p}$, respectively.

(5) If $P=\emptyset$, then an optimal solution has been found. Terminate the algorithm.

The dominant operation of this step is the execution of the `P3Search` algorithm. Since P3 is solved for at most a constant number of multipliers λ, the time complexity of this step is the same as solving a KP with n items and capacity $\sum_{j=1}^{n} b_j - D$. It is given by $\mathcal{O}(2^{t-s+1})$ in the worst case, where s and t correspond to the bounds of the decision core built in COMBO.

Step H3).

(1) For each node $p \in P$, update the current bound LB_{x_p} by computing $LB^{\mathrm{Dom}}_{x_p}$. For computational efficiency concerns, only consider suppliers $q \in P$ as potential dominant nodes in (3.7). Remove p from P if $LB_{x_p} > UB_{x_p}$.

Denoting the remaining number of candidate partial nodes at the beginning of a given step S by $|P(\mathrm{S})|$, the time complexity of H3 is bounded by $\mathcal{O}(|P(\mathrm{H3})|^2)$.

Step H4).

(1) For each node $p \in P$, calculate $Z^{\mathrm{LP}}_{\mathrm{LB}(p)}$ and update $Z_{\mathrm{LB}(p)}$ accordingly. If $Z_{\mathrm{LB}(p)}\geq Z_{\mathrm{UB}}$, remove p from P. Otherwise, calculate $LB^{\mathrm{LP}}_{x_p}$ and $UB^{\mathrm{LP}}_{x_p}$ to improve the bounds LB_{x_p} and UB_{x_p}.

(2) If $P=\emptyset$, then an optimal solution has been found. Terminate algorithm.

(3) If the optimal solution does not have a partial node, it has necessarily already been found during step H2. Setting the global lower bound Z_{LB} to $\min_{p\in P}\{Z_{\mathrm{LB}(p)}\}$ is therefore valid and terminates the heuristic phase.

Using a naive implementation, the time complexity of H4 is $\mathcal{O}(|P(\mathrm{H4})| \cdot n)$. However, respecting the strict total order $\prec$ on the nodes of set P when computing the values $Z^{\mathrm{LP}}_{\mathrm{LB}(p)}$ makes it possible to efficiently adjust the solution to the strong linear relaxation when updating the partial node p. Doing so, the complexity of this step is reduced to $\mathcal{O}\Big(n\log(n) + |P(\mathrm{H4})| \cdot \min\{n, \frac{\max\{b_j\}}{\min\{b_j\}}\}\Big)$.

6.2. Filtering phase

The filtering phase is applied prior to the exact phase in order the reduce the size of the subproblems P1(p) associated with the remaining nodes $p \in P$.

Step F).

(1) Execute the restricted filtering of Section 4.2 to identify nodes $j \in \{i \in Q : i < s^q\}$ that cannot be unused and nodes $j \in \{i \in Q : i > s^q\}$ that cannot be complete in an optimal solution $(\boldsymbol{x}^1, \boldsymbol{y}^1) \in \tilde{S}^1$.

(2) Apply the strong restricted filtering of Section 4.3 to the nodes that have not been fixed by the restricted filtering. The resulting sets R and T contain lists of nodes $j \in Q$ that must be complete and unused in a solution $(\boldsymbol{x}^1, \boldsymbol{y}^1) \in \tilde{S}^1$ to P1(p) for any node $p \in P$ such that $p \neq j$.

The time complexity of this step is $\mathcal{O}\Big(n \cdot \min\{n, \frac{\max\{b_j\}}{\min\{b_j\}}\}\Big)$ in the worst case.

6.3. Exact phase

The algorithm terminates by solving P1(p) for each remaining candidate $p \in P$.

Step E).

(1) For each element $p \in P$, set y_j=1 for each node $j \neq p$ such that $j \in R$ and y_j=0 for each node $j \neq p$ such that $j \in T$ or $j \succ p$, then solve the reduced problem P1(p). If $Z^1(p) < Z_{UB}$, update the incumbent solution. At the end of this step, the incumbent solution $(\boldsymbol{x}^*, \boldsymbol{y}^*) \in \tilde{S}^1$ is optimal.

The exact phase has a complexity of $\mathcal{O}(|P(\text{E})| \cdot 2^{t-s+1})$, where s and t correspond to the bounds of the decision core built in COMBO.

In summary, a total of $\mathcal{O}(1+|P(\text{E})|)$ KPs need to be solved during steps H2 and E. The time complexity of the other steps of KTA is bounded by $\mathcal{O}\Big(n \log(n) + |P(\text{H3})|^2 + n \cdot \min\{n, \frac{\max\{b_j\}}{\min\{b_j\}}\}\Big) \subseteq \mathcal{O}(n^2)$.

7. Computational experiments

There are two main objectives guiding our computational experiments. The first is to compare the performance of our algorithm with that of the state-of-the-art algorithms from the SSFCTP literature and a recent MIP solver on a range of original and existing classes of instances. The second is to provide an in-depth analysis of the performance of KTA and its constituent steps.

To do so, our algorithm, which has been coded in C, is compared with the original C implementation of the enumerative algorithm (EA) and the dynamic programming algorithm (DPA) proposed by Klose (2008), and to Gurobi's MIP solver (version 9.1.1).

All experiments were conducted on a machine with an Intel(R) i7-10875H CPU @ 2.30GHz along with 32 GB of RAM operated with Windows 10 Pro using the Gnu C compiler (version 10.2.0). The instances and detailed results are available at `https://github.com/robinlegault/SSFCTP`.

7.1. Classes of problems

Four classes of problems are considered. The first two groups of instances are inspired by the *uncorrelated* and *almost strongly correlated* instances proposed by Pisinger (2005) for the KP, while the other two are obtained using the generation methods described by Klose (2008).

Different numbers of suppliers n are considered for each group. In Group 2, a correlation parameter β also links the linearized cost of the nodes to their capacity. In Group 3, the b-ratio B_r=$100 \cdot D / \sum_{j=1}^{n} b_j$ controls the percentage of the total offer that is used in a feasible solution, while the f-ratio F_r=$\bar{f}/(\bar{c}\bar{b})$ is the ratio between the average fixed cost $\bar{f}$=$(1/n)\sum_{j=1}^{n} f_j$ and the product of the average unit cost $\bar{c}$=$(1/n)\sum_{j=1}^{n} c_j$ with the average capacity $\bar{b}$=$(1/n)\sum_{j=1}^{n} b_j$ of the suppliers. When generating instances of Group 3, the capacities b_j and the fixed costs f_j are rescaled to meet the desired b-ratio and f-ratio and are then rounded to the nearest integer. The experiments are based on the following configurations of each instance generation method.

- Group 1: $n \in \{500, 1000, 5000, 10000, 25000\}$
- Group 2: (n,β) $\in \{500, 1000, 5000, 10000, 25000\} \times \{5,10,100,1000\}$
- Group 3: (n,B_r,F_r) $\in \{500, 1000, 5000, 10000\} \times \{5, 10, 25, 50\} \times \{0.3, 0.6, 1\}$
- Group 4: $n \in \{500, 1000, 5000, 10000\}$

For each configuration of each class of problems, a group of N=10 instances $i \in \{1, \ldots, N\}$ with n suppliers $j \in \{1, \ldots, n\}$ has been generated as described in Table 1, for a total of 770 instances. The uniform random variables used in the generation process are independently distributed.

Table 1. Instance generation methods

	Group 1: Uncorrelated	**Group 2**: Correlated	**Group 3**: Uncorrelated, fixed demand	**Group 4**: Correlated, small capacities
b_j	unif$\{0.5n, n\}$	unif$\{5000, 10000\}$	unif$(3, 5)$	unif{10,100}
u_j	$c_j b_j + f_j$	$b_j + \alpha_j \cdot 10000/\beta$, where $\alpha_j \sim$ unif$(-1,1)$	$c_j b_j + f_j$	unif$\{b_j, 2b_j\}$
f_j	unif$\{5n, 10n\}$	unif$\{0.3u_j, 0.5u_j\}$	unif$(3, 5)$	$\phi_j u_j$, where $\phi_j \sim$ unif$(0.75,1)$
c_j	unif$(8, 12)$	$(u_j - f_j)/b_j$	unif$(8, 12)$	$(u_j - f_j)/b_j$
D_i	$\frac{i}{N+1}\sum_{j=1}^{n} b_j$	$\frac{i}{N+1}\sum_{j=1}^{n} b_j$	100,000	$\frac{1}{2}\sum_{j=1}^{n} b_j$

7.2. Computational results

This section presents the average computation time required by KTA, the DPA and EA of Klose (2008), and Gurobi to solve the instances of each group. We applied a time limit

of 5 minutes (300,000 ms) per instance for each method. Since the relative optimality gap given by elementary bounds is generally small for instances with a large number of suppliers, the optimality tolerance has been set to 0 in order to properly compare the performance of the algorithms. The other default parameters of the Gurobi MIP solver were left unchanged.

Table 2. Average CPU times (ms), Groups 1, 2, and 4

Group	n	β	KTA	DPA	EA	Gurobi
1	500	-	**0.1**	28.7	6.6	137.8
	1,000	-	**0.3**	163.4	48.8	192.6
	5,000	-	**1.1**	6,892.7	838.1	1,361.2
	10,000	-	**2.2**	39,541.4	1,943.6	2,665.6
	25,000	-	**6.0**	66,384.5$^{(7)}$	20,879.1	10,612.9
2	500	5	**0.3**	1,226.9	2.5	104.7
		10	**0.3**	892.2	4.3	143.0
		100	**0.5**	5,484.6	117.6	1,045.7
		1,000	**1.6**	17,942.9	1,658.4	3,538.3
2	1,000	5	**0.4**	1,103.9	7.1	124.7
		10	**0.4**	1,446.7	16.9	194.3
		100	**0.9**	10,770.8	766.0	1,623.8
		1,000	**1.4**	56,891.8	7,708.3	5,814.7
2	5,000	5	**1.7**	7,530.5	367.5	829.6
		10	**1.7**	9,136.5	1,709.7	693.7
		100	**1.9**	79,393.8	31,709.9	13,121.3
		1,000	**1.7**	24,092.2$^{(9)}$	139,554.8$^{(5)}$	17,871.7
2	10,000	5	**3.2**	10,215.4	977.6	2,274.1
		10	**3.4**	8,389.9	3,393.3	3,238.0
		100	**2.4**	84,854.4$^{(6)}$	107,727.6	26,031.2
		1,000	**3.4**	44,854.4$^{(8)}$	215,663.3$^{(8)}$	54,575.1
2	25,000	5	**9.0**	22,975.6$^{(1)}$	13,533.5	13,558.3
		10	**6.2**	39,935.5	30,926.3	22,579.6
		100	**6.7**	66,884.5$^{(7)}$	84,229.2$^{(9)}$	67,885.7
		1,000	**6.0**	*	*	96,583.9
4	500	-	**0.1**	1.6	5.5	52.2
	1,000	-	**0.1**	3.2	149.6	118.3
	5,000	-	**0.5**	15.0	20,429.4$^{(5)}$	371.1
	10,000	-	**0.9**	24.5	115.5$^{(8)}$	760.2

$^{(\cdot)}$ Number of instances, out of 10, that were not solved to optimality within the time limit of 5 minutes (300,000 ms) or could not be solved due to a memory error. The reported average is only based on the instances that were solved within the time limit.
$*$ None of the 10 instances has been solved to optimality within the time limit.

The results of Tables 2 and 3 show that our algorithm significantly outperforms the other methods for all classes of problems. For Group 1, the running time of KTA grows linearly

Table 3. Average CPU times (ms), Group 3

n	B_r	KTA, F_r			DPA, F_r			EA, F_r			Gurobi, F_r		
		0.3	0.6	1	0.3	0.6	1	0.3	0.6	1	0.3	0.6	1
500	5	**0.3**	**0.3**	**0.3**	570.4	1,271.4	1,568.0	0.6	8.5	7.5	77.9	92.3	66.5
	10	**0.3**	**0.2**	**0.3**	268.2	335.7	553.5	1.0	3.9	26.4	95.7	120.9	88.1
	25	**0.3**	**0.2**	**0.2**	73.8	111.9	112.3	5.1	4.2	14.0	96.5	83.8	108.9
	50	**0.2**	**0.1**	**0.1**	24.8	41.6	48.5	4.6	8.3	16.1	90.8	108.8	127.1
1,000	5	**0.5**	**0.5**	**0.5**	348.8	390.1	855.5	3.9	20.0	42.4	85.9	106.8	214.5
	10	**0.5**	**0.4**	**0.4**	84.5	240.1	361.9	10.1	44.4	42.1	86.1	149.9	165.7
	25	**0.4**	**0.3**	**0.2**	50.2	60.9	73.4	9.5	20.2	31.0	135.2	145.2	171.5
	50	**0.3**	**0.1**	**0.2**	19.5	17.3	45.5	6.0	35.8	37.0	178.6	123.6	226.3
5,000	5	**2.0**	**1.1**	**1.2**	84.8	166.4	312.6	157.6	256.6	19,856.7	240.2	325.5	397.0
	10	**1.7**	**0.5**	**0.4**	73.5	116.2	116.7	101.8	660.5	337.0	257.1	355.2	324.1
	25	**1.3**	**0.4**	**0.4**	34.8	44.0	64.4	62.8	89.3	106.5	355.8	387.1	380.7
	50	**0.4**	**0.4**	**0.4**	20.1	30.8	62.9	40.5	83.4	167.1	301.7	398.9	612.8
10,000	5	**3.7**	**1.3**	**0.9**	80.9	155.6	324.3	524.2	3,432.2	13,799.1	433.1	515.4	675.7
	10	**3.1**	**0.8**	**0.8**	75.3	167.9	114.8	218.9	414.3	3,123.2	491.2	421.4	736.0
	25	**2.1**	**0.8**	**0.9**	21.5	75.9	53.7	72.3	239.0	297.6	424.9	716.0	723.6
	50	**0.8**	**0.8**	**0.7**	18.2	35.7	40.1	85.0	229.9	198.0	568.2	826.0	663.7

with n, while the empirical time complexity of DPA and EA shows exponential growth. In particular, 7 of the 10 instances of size n=25,000 could not be solved by DPA within the time limit. Interestingly, as the size of the problems increases, the general-purpose Gurobi MIP solver starts to outperform the state-of-the-art algorithms of Klose (2008).

The same trends emerge even more clearly on the instances of Group 2. Furthermore, while the computing time required by DPA, EA and Gurobi drastically increases with the correlation parameter β, the performance of KTA remains essentially unchanged by β for a given number of suppliers. Moreover, for n=25,000 and β=1,000, where both DPA and EA were unable to solve a single instance in less than 300,000 ms, KTA only required 6.0 ms on average, which is less than the 9.0 ms required for the weakly correlated instances obtained for n=25,000 and β=5. In comparison, the time required by Gurobi was multiplied by more than 7, from 13,558.3 ms to 96,583.9 ms, for the same pair of configurations. The structure of the instances of Group 2 can be analyzed to better interpret these results. As β increases, the linearized cost e_j of each node converges to 1, making it more difficult to reduce the size of the problem using dominance relations and filtering procedures. Consequently, one should expect the problem to be challenging when both β and n are set to a high value. On the other hand, as the value of the correlation parameter increases, the problem eventually reduces to finding a feasible solution in which the excess offer that remains unused on the partial node is minimized. In particular, for β=∞, the objective value of any feasible solution that does not include a partial node would reach the linear relaxation lower bound $Z_{\text{LB}}^{\text{LP}}$=$D$ and its optimality could thus be trivially proven. In this case, the SSFCTP would thus be very similar to a subset sum problem. Since the capacities b_j take their value in a fixed interval

in the instances of Group 2, finding a subset of nodes whose total capacity is exactly D becomes easier as the number of nodes increases. Problems with very large values of both β and n may thus become easy to solve using appropriate approaches. The results confirm that KTA effectively exploits this structure, which is not the case for DPA, EA, and Gurobi.

The experiments carried out on Groups 3 and 4 also confirm the superiority of our algorithm over the existing methods. Regarding the instances of Group 3, as noticed by Klose (2008), the execution time of both DPA and EA increases as the f-ratio approaches 1. While this same behavior is also observable for Gurobi, KTA shows the opposite tendency. These results can be interpreted in a similar way to those obtained for the large instances of Group 2. In the case of Group 3, as F_r increases, the structure of the instances becomes closer to that of a min-KP, since the contribution to the objective of the fixed costs becomes proportionally more important. Our approximation of the SSFCTP by a KP thus becomes more precise, which strengthens the bounds of step H2. This is especially the case for the largest instances of Group 3. Since the demand D does not depend on n in this generation method, the largest instances have small capacities b_j, which leads to optimal solutions that generally do not have a partial node or have a partial node that is practically used at capacity. This also improves the quality of the bounds we obtain by solving P3.

7.3. Frequency of optimal solutions with a partial node

This section presents the proportion of instances of each group for which the optimal solution contains a partial node. Each entry in Table 4, except for Group 3, has been calculated on a set of N=100 new instances. We used common random numbers to minimize the variance in results due to the generation process in order to better isolate the influence of the parameters on the problem structure. For Group 3, the same set of 120 instances that were used in the previous section for each value of n has been preserved to cover the range of f-ratios and b-ratios that were considered by Herer et al. (1996) and Klose (2008).

Table 4. Percentage of instances whose optimal solution includes a partial node

n	Group 1	Group 2, $\beta = 5$	Group 2, $\beta = 10$	Group 2, $\beta = 100$	Group 2, $\beta = 1{,}000$	Group 3	Group 4
500	60	100	97	80	19	90	0
1,000	68	96	92	67	7	58	0
5,000	69	84	76	19	0	9	0
10,000	77	80	66	1	0	1	0
25,000	71	63	47	0	0	-	-

This measure is an important indicator of the structure of the SSFCTP. Indeed, for some groups of instances, the optimal solution never includes a partial node. For such problems, the distinction between the fixed cost f_j and the transportation cost $c_j b_j$ of a

supplier completely disappears, since the total cost $u_j{=}c_jb_j{+}f_j$ is systematically paid on each selected node. More importantly, this means that solving a single min-KP, which is given by model P3 with $\lambda{=}0$, always suffices to find the optimal solution for these instances. Notably, the optimal solution of the instances of Group 4 never contained a partial node. Furthermore, when using KTA, `P3Search` required solving a single KP per instance, and the bounds of step H2 sufficed in each case to prove the optimality of the resulting solution. This means that no computations were performed during steps H3, H4, F, and E for the instances of Group 4. As a consequence, the correlated instances of Klose (2008) can now essentially be regarded as relatively simple instances of the min-KP.

Also, as opposed to the uncorrelated instances of Group 3, those of Group 1 preserve the same structure for every problem size, with approximately 70% of the optimal solutions containing a partial node. This property makes it possible to test the influence of problem size on the performance of algorithms without introducing a bias in the analysis that would be due to an underlying transformation of the optimal solution properties.

In this sense, the generation method of Group 2 is complementary to that of Group 1. Its parameters jointly transform the problem structure from a pure SSFCTP for which the optimal solution always requires a partial node and is therefore quite distinct from a KP, to a problem that closely resembles the KP or the subset sum problem as n and β increase. Together, these two classes of problems permit a robust analysis of the performance of algorithms on a range of SSFCTPs that aims to cover most of the important problem structures that are likely to emerge from future application contexts.

7.4. Performance analysis of KTA

This section presents detailed results on the computational efficiency of each step of KTA and their ability to reduce the problem size by filtering candidate partial nodes and reducing the number of items in the knapsack subproblems solved in the exact phase of the algorithm.

Table 5 reports the average computation time required by each step of the algorithm to solve the instances of Groups 1 and 2 from Section 7.2. The majority of the execution time, respectively 65% for Group 1 and 72% for Group 2, has been spent in H2, the knapsack relaxation step. Overall, H4, the strong linear relaxation step, was the second most time-consuming, followed by H3, the dominance relation step, and H1, the elementary bounds phase. The filtering and exact phases together required less than 1% of the total computing time for both groups.

These computation times reflect the relative contribution of each step to the entire solving process. A notable fact is that, for each of the 3,670 instances we considered, the optimal solution was identified during step H2. This means that the equality $Z_{\mathrm{UB}}^{\mathrm{P3}}{=}Z^1$ held for each problem. Although it is easy to build small instances in which the exact phase improves the

Table 5. Percentage of the total CPU time spent in each step of KTA, Groups 1 and 2

Group	n	β	H1	H2	H3	H4	F	E
1	500	-	5.9	82.5	1.0	10.7	0.0	0.0
	1,000	-	6.3	73.8	2.8	17.1	0.0	0.0
	5,000	-	8.2	63.0	2.7	26.0	0.0	0.0
	10,000	-	7.9	61.4	3.2	26.2	0.8	0.5
	25,000	-	8.1	66.2	2.8	22.9	0.0	0.0
2	500	5	3.3	56.5	12.4	16.3	1.9	9.6
		10	3.0	57.8	7.6	14.7	2.7	14.1
		100	1.6	83.3	1.0	5.8	0.8	7.5
		1,000	0.5	98.3	0.1	0.5	0.1	0.5
2	1,000	5	4.9	50.5	15.1	23.5	2.4	3.5
		10	4.6	61.7	10.9	22.8	0.0	0.0
		100	2.0	92.2	0.5	4.8	0.3	0.3
		1,000	1.5	98.5	0.0	0.0	0.0	0.0
2	5,000	5	5.7	50.1	20.0	22.0	1.2	1.0
		10	5.5	57.7	14.2	21.6	0.6	0.3
		100	4.6	91.6	0.1	2.3	0.6	0.7
		1,000	5.0	95.0	0.0	0.0	0.0	0.0
2	10,000	5	6.5	45.4	24.3	21.0	1.9	0.9
		10	6.0	54.8	14.5	22.3	1.2	1.2
		100	7.3	92.7	0.0	0.0	0.0	0.0
		1,000	5.3	94.7	0.0	0.0	0.0	0.0
2	25,000	5	5.5	43.4	26.4	22.3	1.7	0.7
		10	9.3	56.3	12.9	21.5	0.0	0.0
		100	8.0	92.0	0.0	0.0	0.0	0.0
		1,000	8.6	91.4	0.0	0.0	0.0	0.0
	Average		5.4	72.4	3.1	13.0	0.7	0.5

incumbent solution of step H2, this never happened in our test instances. In practice, the only purpose of the subsequent steps, including the exact phase, is therefore to prove the optimality of the best feasible solution identified in `P3Search`.

Table 6 shows the percentage of nodes that could still be partial after steps H2, H3, and H4. For Group 1, the observed percentages after each step are relatively stable from one size of problem to another. The knapsack relaxation step filters approximately 97.5% of the nodes on average. From there, the dominance step divides the cardinality of P by approximately 10 and the strong linear relaxation eliminates virtually all the remaining candidates. Therefore, the exact phase is almost never necessary for the instances of Group 1. In Group 2, as the correlation parameter increases, the average percentage of candidates remaining after H2 goes from more than 25% for β=5 to less than 0.1% for β=1,000. Fortunately, for small values of β, the bounds from H3 and H4 compensate for the weakness of those of H2. The

Table 6. Percentage of nodes that can be partial after each filtering step, Groups 1 and 2

Group	n	β	H2	H3	H4
1	500	-	0.64	0.32	0.00
	1,000	-	3.86	0.66	0.00
	5,000	-	2.70	0.16	0.00
	10,000	-	3.09	0.10	0.00
	25,000	-	2.48	0.06	0.00
2	500	5	29.08	10.84	0.16
		10	17.40	7.94	0.24
		100	2.62	2.38	0.08
		1,000	0.34	0.32	0.04
2	1,000	5	27.86	8.55	0.05
		10	15.94	6.49	0.00
		100	0.59	0.57	0.01
		1,000	0.00	0.00	0.00
2	5,000	5	25.60	4.79	0.00
		10	13.27	3.55	0.00
		100	0.05	0.05	0.00
		1,000	0.00	0.00	0.00
2	10,000	5	24.95	3.67	0.00
		10	14.42	3.04	0.00
		100	0.00	0.00	0.00
		1,000	0.00	0.00	0.00
2	25,000	5	26.00	2.93	0.00
		10	7.44	1.26	0.00
		100	0.00	0.00	0.00
		1,000	0.00	0.00	0.00
	Average		8.73	2.31	0.02

average number of subproblems P1(p) that need to be solved during the exact phase is consequently extremely small in all configurations.

Table 7 gives further details about the knapsack subproblems that are solved in both the heuristic and exact phases. The KPs that are considered in the heuristic phase are solved within `P3Search` during step H2 and correspond to model P3 for different reimbursement rates λ. They are quite different in structure from the ones of the exact phase, which are solved during step E to obtain the optimal solutions to the subproblems P1(p) for the remaining candidates $p \in P$. In the latter case, the filtering procedure described in step F significantly reduces the size of the KPs. Indeed, for both Groups 1 and 2 and for every problem size, the average number of items to be considered per knapsack subproblem was approximately 40. This corresponds to 8% of the suppliers when n=500 and less than 0.002% of them when n=25,000. Furthermore, the exact phase was required for only 1 of the 40 instances of the first group of problems and for 35 of the 200 instances of the second group,

Table 7. KPs solved, Groups 1 and 2

			Avg. number of KPs/instance			Properties of exact phase's KPs	
Group	n	β	Heuristic phase	Exact phase	Total	Instances requiring the exact phase	Avg. number of items per KP in the exact phase
1	500	-	1.4	0.0	1.4	0	-
	1,000	-	1.7	0.0	1.7	0	-
	5,000	-	1.6	0.0	1.6	0	-
	10,000	-	1.7	0.1	1.8	1	40.0
	25,000	-	1.7	0.0	1.7	0	-
2	500	5	2.4	0.8	3.2	5	27.9
		10	2.1	1.3	3.4	7	27.4
		100	1.8	0.4	2.2	4	44.0
		1,000	1.2	0.2	1.4	1	70.0
2	1,000	5	2.0	0.5	2.5	5	21.6
		10	2.0	0.0	2.0	0	-
		100	1.5	0.1	1.6	1	56.0
		1,000	1.0	0.0	1.0	0	-
2	5,000	5	2.0	0.2	2.2	2	37.5
		10	1.9	0.1	2.0	1	29.0
		100	1.1	0.1	1.2	1	56.0
		1,000	1.0	0.0	1.0	0	-
2	10,000	5	1.8	0.3	2.1	3	37.7
		10	1.9	0.3	2.2	2	36.3
		100	1.0	0.0	1.0	0	-
		1,000	1.0	0.0	1.0	0	-
2	25,000	5	1.9	0.3	2.2	3	39.3
		10	1.6	0.0	1.6	0	-
		100	1.0	0.0	1.0	0	-
		1,000	1.0	0.0	1.0	0	-
Average			1.6	0.2	1.8	1.4	34.6*

* Weighted by the number of KPs solved in the exact phase for each type of instance

for an average of less than 0.2 KP solved per instance. During the heuristic phase, 1.6 KP was required per instance for both groups on average.

In total, KTA required solving less than two KPs per instance on average for both groups. Since solving these subproblems represents most of the computation time of KTA, we can conclude that our new method, at least for the types of instances that were considered in this article, fills the gap between the state of the art on the KP and the SSFCTP.

8. Conclusion

We introduced a binary nonlinear programming reformulation of the single-sink fixed-charge transportation problem. Using the knapsack problem that is obtained by a relaxation of this new model, we developed several bounds, some of which are subsequently improved by a dominance relation between potential partial nodes and a strong linear relaxation to produce an efficient and robust heuristic method. A new filtering procedure allows us to complete the exact algorithm by iteratively fixing a partial node and solving a reduced-size knapsack problem. As shown on a large set of instances, the knapsack transformation algorithm completely outperforms the existing algorithms from the literature. In particular, a reduction of several orders of magnitude in the solving time of the state-of-the-art methods occurs for large and highly structured instances of the SSFCTP.

Acknowledgments

This work is dedicated to our colleague Bernard Gendron who was taken far too soon and will be deeply missed. Financial support for this work was provided by the Canadian Natural Sciences and Engineering Research Council (NSERC) under grants 2017-06054 and 2021-04037. This support is gratefully acknowledged. We thank Andreas Klose for making available the source code of his algorithms.

Second Article.

A Model-Free Approach for Solving Choice-Based Competitive Facility Location Problems Using Simulation and Submodularity

by

Robin Legault[1], and Emma Frejinger[1]

([1]) Department of Computer Science and Operations Research and CIRRELT, Université de Montréal, Montreal, Quebec H3T 1J4, Canada

Journal submission
This article was submitted to INFORMS Journal on Computing and appeared as a preprint (Legault and Frejinger, 2023) in August 2023.

Authors contributions

- Emma Frejinger proposed the original research topic (improvement of simulation-based methods for choice-based optimization).
- The core of the research (development of models, theoretical results, proofs, algorithms, code, etc.) was carried out by Robin Legault.
- Throughout the project, the two authors participated in regular meetings to discuss new research ideas and the latest results.
- The article was written by Robin Legault, and it was revised by Emma Frejinger.

RÉSUMÉ. Cet article considère un problème de localisation d'installations dans lequel une firme s'implantant sur un marché cherche à ouvrir un ensemble d'emplacements dans le but de maximiser sa part de marché espérée, en supposant que les clients choisissent l'alternative qui maximise une fonction d'utilité aléatoire. Nous introduisons une nouvel équivalent déterministe de ce modèle probabiliste et généralisons les résultats d'études antérieures en montrant que sa fonction objectif est sous-modulaire sous n'importe quel modèle de maximisation d'utilité aléatoire. Cette reformulation caractérise la demande sur la base d'un ensemble fini de profils de préférence. L'estimation de leur prévalence par simulation généralise une méthode d'approximation de la moyenne de l'échantillon (*sample average approximation*) de la littérature et résulte en un problème de couverture maximale pour lequel nous développons un nouvel algorithme de séparation et d'élagage (*branch-and-cut*). La méthode proposée exploite la sous-modularité de la fonction objectif pour remplacer les profils de préférence les moins influents par une variable auxiliaire bornée par des coupes sous-modulaires. Cet ensemble de profils est sélectionné par une méthode du genou (*knee method*). Nous fournissons une analyse théorique de notre approche et montrons que sa performance computationnelle, la qualité des solutions qu'elle fournit et l'efficacité de la méthode du genou qu'elle exploite sont directement liées à l'entropie des profils de préférence dans la population. Des expériences réalisées sur des instances de la littérature et de nouvelles instances indiquent que notre approche domine la méthode classique d'approximation de la moyenne de l'échantillon sur les instances de grande taille, qu'elle peut surpasser la meilleure méthode heuristique de la littérature sous le modèle logit multinomial et qu'elle améliore l'état de l'art sous le modèle logit multinomial mixte.
Mots clés : Emplacement d'installations, Optimisation basée sur les choix, Simulation, Sous-modularité, Entropie.

ABSTRACT. This paper considers facility location problems in which a firm entering a market seeks to open a set of available locations so as to maximize its expected market share, assuming that customers choose the alternative that maximizes a random utility function. We introduce a novel deterministic equivalent reformulation of this probabilistic model and, extending the results of previous studies, show that its objective function is submodular under any random utility maximization model. This reformulation characterizes the demand based on a finite set of preference profiles. Estimating their prevalence through simulation generalizes a sample average approximation method from the literature and results in a maximum covering problem for which we develop a new branch-and-cut algorithm. The proposed method takes advantage of the submodularity of the objective value to replace the least influential preference profiles by an auxiliary variable that is bounded by submodular cuts. This set of profiles is selected by a knee detection method. We provide a theoretical analysis of our approach and show that its computational performance, the solution quality it provides, and the efficiency of the knee detection method it exploits are directly connected to the entropy of the preference profiles in the population. Computational experiments on existing and new benchmark sets indicate that our approach dominates the classical sample average approximation method on large instances, can outperform the best heuristic method from the literature under the multinomial logit model, and achieves state-of-the-art results under the mixed multinomial logit model.
Keywords : Facility location, Choice-based optimization, Simulation, Submodularity, Entropy.

1. Introduction

In a wide array of real-world management applications, the impact of a firm's decisions on the level of demand for its products or services depends on the preferences of a population of heterogeneous customers. The core assumption of utility maximization theory is that agents evaluate each available alternative and select the one that maximizes their utility function. Embedding random utility maximization (RUM) models into optimization problems leads to so-called choice-based optimization problems, prime examples of which include pricing (e.g., Davis et al., 2017, Gallego and Wang, 2014, Li et al., 2019, Paneque et al., 2022) and assortment optimization problems (e.g., Liu et al., 2020, Rusmevichientong et al., 2010).

The competitive facility location problem is another important problem that requires modeling customer demand at a disaggregate level. In the last two decades, a growing body of literature has proposed exact methods for solving choice-based competitive facility location problems under the multinomial logit (MNL) model (Aros-Vera et al., 2013, Benati and Hansen, 2002, Freire et al., 2016, Ljubić and Moreno, 2018, Mai and Lodi, 2020, Zhang et al., 2012). A limitation of the MNL model is that it implies proportional substitution patterns, meaning that introducing a new alternative divides by the same factor the probability for each existing alternative to be selected by any given customer. This property leads to unrealistic demand representation in many cases. Although empirical studies underscore the need to capture unobserved taste variations and spatial correlation to faithfully model location-related behavior (Bhat and Guo, 2004, Miyamoto et al., 2004, Müller et al., 2012), relatively little attention has been devoted to the study of the competitive facility location problem under less restrictive modeling assumptions. Notable exceptions include an exact branch-and-cut (B&C) method for the nested logit choice rule (Méndez-Vogel et al., 2023) and a heuristic local search approach that can be applied under any generalized extreme value (GEV) models (Dam et al., 2022). In addition to the MNL and nested logit models (Williams, 1977), prominent members of the GEV family include the paired combinatorial logit model (Chu, 1989) and the cross-nested logit model (Vovsha, 1997).

The most widely studied RUM model outside of the GEV family is the mixed multinomial logit (MMNL) model, which can approximate any RUM model with arbitrary precision (McFadden and Train, 2000a). Some facility location studies (e.g., Dam et al., 2022, Haase et al., 2016, Mai and Lodi, 2020) have highlighted that this fully flexible model can be approximated by a MNL model through simulation. Unfortunately, obtaining near-optimal solutions for MMNL instances with this approach requires solving very large MNL instances in some cases, which can be computationally intractable even for state-of-the-art methods.

Another approach that can accommodate flexible RUM models is the sample average approximation framework introduced by Haase (2009). Initially developed for the MNL model and later extended to the MMNL model (Haase et al., 2016, Haase and Müller, 2013), this

method approximates the utility function of each customer by sampling realizations of its random terms. The resulting model can be formulated as a deterministic maximum covering problem (Church and ReVelle, 1974) by describing each simulated customer by the subset of available locations that would capture their demand. This method has been applied in school location planning (Haase et al., 2019) and to optimize electric vehicle charging station placement (Lamontagne et al., 2023). More generally, sample average approximation is receiving increasing attention in choice-based optimization as it makes it relatively straightforward to integrate advanced RUM models into mixed-integer linear programming (MILP) models (Paneque et al., 2021).

In this paper, we propose a B&C algorithm that exploits the submodularity property of the choice-based competitive facility location problem, which we show holds under any RUM model. Our approach approximates a novel deterministic equivalent reformulation of the problem by aggregating simulated customers by preference profile, leading to a smaller yet equivalent reformulation of the sample average approximation method of Haase et al. (2016). These preference profiles are partitioned in what we refer to as a hybrid submodular reformulation using a knee detection method. As opposed to model-specific methods that exploit the structure of a given family of RUM models, our approach does not make any restrictive assumption on the utility functions of the customers and is thus said to be model-free. We develop an information-theoretic analysis of the hybrid submodular method and provide insights regarding the impact of the concentration of the demand, measured by the entropy of the preference profiles, on its computational performance and the solution quality it provides. We compare our approach with the sample average approximation of Haase et al. (2016) and state-of-the-art model-specific exact (Mai and Lodi, 2020) and heuristic (Dam et al., 2022) methods on existing MNL instances and new MMNL instances. Our computational study shows that our algorithm performs better than the classical sample average approximation method and the state-of-the-art heuristic method in terms of CPU time and solution quality for MNL instances with low to moderate entropy. Furthermore, it achieves state-of-the-art results under the MMNL model.

The paper is structured as follows. Section 2 presents the choice-based competitive facility location problem and introduces our notation. Section 3 discusses existing methods from the literature. Section 4 presents the simulation-based hybrid submodular method, which is studied from an information-theoretic perspective in Section 5. The computational experiments are reported in Section 6. Section 7 concludes the paper.

2. Choice-based competitive facility location

We consider a probabilistic facility location problem in a competitive market with utility-maximizing customers. Given a set D of available locations on which a firm can install new

facilities and a set E of locations that competing facilities already occupy, the goal is to identify a feasible configuration $\boldsymbol{x} \in X$ that maximizes the expected market share captured by the new facilities. A binary variable x_d indicates whether the company decides to open a facility at location $d \in D$. The feasible domain $X \subseteq \{0,1\}^{|D|}$ is specified by linear business constraints, such as the number of facilities that can be opened or the firm's budget. We denote by $C = D \cup E$ the set of both candidate and existing locations.

Customers select the open facility that maximizes their utility function. The attributes of the customers among the population, such as their location and personal preferences, are modeled by a random vector $\boldsymbol{\theta}$ with support Θ. The impact of these attributes on the utility of each location is modulated by coefficients $\boldsymbol{\beta} \in B$. The specification of these coefficients depends on the model, and they are possibly random. In real applications, the corresponding values are estimated on data. Finally, a random term ε_c affects the utility of each alternative $c \in C$. For a solution $\bar{\boldsymbol{x}} \in X$, the open alternatives are given by the set $C_{\bar{\boldsymbol{x}}} = D_{\bar{\boldsymbol{x}}} \cup E$, where $D_{\bar{\boldsymbol{x}}} = \{d \in D : \bar{x}_d = 1\}$. The utility of each available alternative $c \in C$ is denoted by $u_c(\boldsymbol{\theta}, \boldsymbol{\beta}, \boldsymbol{\varepsilon})$. In the case of RUM models with additive error terms, this value decomposes as $u_c(\boldsymbol{\theta},\boldsymbol{\beta},\boldsymbol{\varepsilon}) = v_c(\boldsymbol{\theta},\boldsymbol{\beta}) + \varepsilon_c$. In general, the random vectors $\boldsymbol{\theta}$,$\boldsymbol{\beta}$ and $\boldsymbol{\varepsilon}$ can be dependent, and $v_c : \Theta \times B \to \mathbb{R}$ can be any real-valued function for each location $c \in C$.

In most of the literature, it is assumed that the population is composed of a finite number of customers $n \in N$, each described by observed attributes $\boldsymbol{\theta}^n$. In this context, the distribution of $\boldsymbol{\theta}$ is given by the empirical distribution of the observations $\boldsymbol{\theta}^n$. In machine learning terminology, this interpretation underlies a *conditional*, or *discriminative* perspective on the attributes of the customers. In other words, the observed set of customers is considered fixed instead of being seen as a random sample drawn from an underlying probability distribution. For the sake of generality, the notation adopted in this paper instead proposes a *generative* perspective, in which all of the customers' attributes are modeled as random variables. The problem can then be stated as maximizing the expected market share $F : 2^D \to [0,1]$ based on the joint distribution of the random vectors $\boldsymbol{\theta}$, $\boldsymbol{\beta}$, and $\boldsymbol{\varepsilon}$. Assuming continuous probability distributions, it can be expressed as follows:

$$\max_{\boldsymbol{x}\in X} F(D_{\boldsymbol{x}}) = \max_{\boldsymbol{x}\in X} \mathbb{P}_{\boldsymbol{\theta},\boldsymbol{\beta},\boldsymbol{\varepsilon}}\Big[\arg\max_{c\in C_{\boldsymbol{x}}}\{u_c(\boldsymbol{\theta},\boldsymbol{\beta},\boldsymbol{\varepsilon})\} \in D_{\boldsymbol{x}}\Big], \tag{2.1}$$

$$= \max_{\boldsymbol{x}\in X} \int_{\boldsymbol{\theta}\in\Theta} \mathbb{P}_{\boldsymbol{\beta},\boldsymbol{\varepsilon}|\boldsymbol{\theta}}\Big[\arg\max_{c\in C_{\boldsymbol{x}}}\{u_c(\boldsymbol{\theta},\boldsymbol{\beta},\boldsymbol{\varepsilon})\} \in D_{\boldsymbol{x}}\Big] p(\boldsymbol{\theta}) d\boldsymbol{\theta}, \tag{2.2}$$

$$= \max_{\boldsymbol{x}\in X} \int_{\boldsymbol{\theta}\in\Theta} \sum_{d\in D_{\boldsymbol{x}}} \mathbb{P}_{\boldsymbol{\beta},\boldsymbol{\varepsilon}|\boldsymbol{\theta}}\Big[u_d(\boldsymbol{\theta},\boldsymbol{\beta},\boldsymbol{\varepsilon}) \geq u_c(\boldsymbol{\theta},\boldsymbol{\beta},\boldsymbol{\varepsilon}),\ \forall c \in C_{\boldsymbol{x}}\Big] p(\boldsymbol{\theta}) d\boldsymbol{\theta}, \tag{2.3}$$

which, for RUM models with additive error terms, can be rewritten as:

$$\max_{\boldsymbol{x}\in X} \int_{\boldsymbol{\theta}\in\Theta} \sum_{d\in D_{\boldsymbol{x}}} \mathbb{P}_{\boldsymbol{\beta},\boldsymbol{\varepsilon}|\boldsymbol{\theta}}\Big[\varepsilon_c - \varepsilon_d \leq v_d(\boldsymbol{\theta},\boldsymbol{\beta}) - v_c(\boldsymbol{\theta},\boldsymbol{\beta}),\ \forall c \in C_{\boldsymbol{x}}\Big] p(\boldsymbol{\theta}) d\boldsymbol{\theta}. \tag{2.4}$$

For the choice probability in (2.4) to be directly embedded in an optimization problem, the cumulative distribution function of the difference $\varepsilon_c - \varepsilon_d$ must admit an analytical form. This has motivated the development of increasingly flexible models belonging to the GEV family, which all result in closed-form expressions (Fosgerau et al., 2013). In particular, a large part of the RUM literature assumes the coefficients $\boldsymbol{\beta}$ to be independent of the customers' attributes $\boldsymbol{\theta}$ and is based on additive error terms ε_c defined as independent and identically distributed (IID) type I extreme value (or Gumbel) random variables. These assumptions result in the MMNL model, in which the probability of capturing the demand of a customer with attributes $\boldsymbol{\theta}$ is given by the following expression:

$$\mathbb{P}_{\boldsymbol{\beta},\boldsymbol{\varepsilon}|\boldsymbol{\theta}}\Big[\varepsilon_c - \varepsilon_d \leq v_d(\boldsymbol{\theta},\boldsymbol{\beta}) - v_c(\boldsymbol{\theta},\boldsymbol{\beta}),\ \forall c \in C_{\boldsymbol{x}}\Big] = \int_{\beta\in B} \frac{\sum_{d\in D_{\boldsymbol{x}}} e^{v_d(\boldsymbol{\theta},\boldsymbol{\beta})}}{\sum_{c\in C_{\boldsymbol{x}}} e^{v_c(\boldsymbol{\theta},\boldsymbol{\beta})}} p(\boldsymbol{\beta}) d\boldsymbol{\beta}. \tag{2.5}$$

Under the MMNL model and a generative perspective, model (2.1) can be formulated as:

$$\max_{\boldsymbol{x}\in X} \int_{\boldsymbol{\theta}\in\Theta} \int_{\beta\in B} \frac{\sum_{d\in D_{\boldsymbol{x}}} e^{v_d(\boldsymbol{\theta},\boldsymbol{\beta})}}{\sum_{c\in C_{\boldsymbol{x}}} e^{v_c(\boldsymbol{\theta},\boldsymbol{\beta})}} p(\boldsymbol{\beta}) p(\boldsymbol{\theta}) d\boldsymbol{\beta} d\boldsymbol{\theta}. \tag{2.6}$$

In turn, under the conditional perspective, it becomes:

$$\max_{\boldsymbol{x}\in X} \sum_{n\in N} q_n \int_{\beta\in B} \frac{\sum_{d\in D_{\boldsymbol{x}}} e^{v_d(\boldsymbol{\theta},\boldsymbol{\beta})}}{\sum_{c\in C_{\boldsymbol{x}}} e^{v_c(\boldsymbol{\theta},\boldsymbol{\beta})}} p(\boldsymbol{\beta}) d\boldsymbol{\beta}, \tag{2.7}$$

where $q_n = p(\boldsymbol{\theta}^n)$ denotes the weight of each customer $n \in N$.

The MNL model is a specific case where the coefficients $\boldsymbol{\beta}$ are deterministic, which simplifies the generative MMNL model (2.6) to:

$$\max_{\boldsymbol{x}\in X} \int_{\boldsymbol{\theta}\in\Theta} \frac{\sum_{d\in D_{\boldsymbol{x}}} e^{v_d(\boldsymbol{\theta},\boldsymbol{\beta})}}{\sum_{c\in C_{\boldsymbol{x}}} e^{v_c(\boldsymbol{\theta},\boldsymbol{\beta})}} p(\boldsymbol{\theta}) d\boldsymbol{\theta}. \tag{2.8}$$

Similarily, under the conditional perspective, (2.7) simplifies to:

$$\max_{\boldsymbol{x}\in X} \sum_{n\in N} q_n \frac{\sum_{d\in D_{\boldsymbol{x}}} e^{v_d^n}}{\sum_{c\in C_{\boldsymbol{x}}} e^{v_c^n}}, \tag{2.9}$$

where the perceived utility of an available alternative $c \in C$ for customer $n \in N$ is given by $v_c^n := v_c(\boldsymbol{\theta}^n, \boldsymbol{\beta})$. In this case, and more generally if the support Θ is discrete and finite, the problem be solved exactly by mixed-integer nonlinear programming (MINLP) solvers or by model-specific algorithms (Ljubić and Moreno, 2018, Mai and Lodi, 2020). In order to solve MMNL or generative MNL instances approximately using these methods, the distribution of the random variables $\boldsymbol{\theta}$ and/or $\boldsymbol{\beta}$ can be approximated by the empirical distribution provided by a set of their realizations, resulting in a conditional MNL problem.

3. Methods from the literature

This section discusses the most recent state-of-the-art exact and heuristic methods for the competitive facility location problem under the MNL and MMNL models, as well as the sample average approximation method our approach builds on.

3.1. State-of-the-art exact method

The multicut outer-approximation (MOA) algorithm (Mai and Lodi, 2020) uses an outer-approximation scheme (Duran and Grossmann, 1986) to solve the MNL model exactly under a conditional perspective. To do so, model (2.9) is first reformulated as a minimization problem, with objective function $G(\boldsymbol{x}) = -\sum_{n\in N} q_n \frac{\sum_{d\in D_{\boldsymbol{x}}} e^{v_d^n}}{\sum_{c\in C_{\boldsymbol{x}}} e^{v_c^n}}$. The main idea of the MOA algorithm is to partition the set N of customers into T subsets so that $G(\boldsymbol{x})$ can be expressed as a sum of T convex and continuously differentiable functions $g_t(\boldsymbol{x})$. At each iteration of the algorithm, the master problem $\min_{\boldsymbol{x}\in X}\{\sum_{t=1}^{T}\phi_t | \phi_t \geq L_t,\ \Pi_t\boldsymbol{x} - \mathbf{1}\phi_t \leq \pi_{0t}\ \forall t\}$ is solved, where the value of each component $g_t(\boldsymbol{x})$ is replaced by the decision variable ϕ_t, $\Pi_t\boldsymbol{x} - \mathbf{1}\phi_t \leq \pi_{0t}$ is the set of subgradient cuts corresponding to $g_t(\boldsymbol{x})$, and L_t is a lower bound on $g_t(\boldsymbol{x})$. Its optimal solution $\boldsymbol{x}^*$ is then used to add up to T new subgradient cuts $\phi_t \geq \nabla g_t(\boldsymbol{x}^*)(\boldsymbol{x} - \boldsymbol{x}^*) + g_t(\boldsymbol{x}^*)$, $t = 1,\ldots,T$ to the master problem, where $\nabla g_t(\boldsymbol{x}^*)$ is the gradient of g_t evaluated at $\boldsymbol{x}^*$.

This multicut approach generalizes an earlier single-cut algorithm by Bonami et al. (2008). The single-cut version is obtained by using only $T = 1$ objective function $g_1(\boldsymbol{x}) = G(\boldsymbol{x})$, while taking $T \in \{2,\ldots,|N|-1\}$ leads to what is sometimes referred to as a *hybrid* approach (Birge and Louveaux, 1988) in the stochastic programming literature. Selecting a large value of T tends to limit the number of iterations of MOA, but adding too many cuts at each iteration makes the master problem more challenging to solve. Mai and Lodi (2020) report that neither the single-cut nor the pure multicut version with $T = |N|$ usually results in the most efficient version of the MOA. Furthermore, the best value of T varies significantly across instances. Unfortunately, no efficient rule for automatically selecting T is known, making the performance of MOA heavily dependent on a user-defined parameter.

3.2. State-of-the-art heuristic method

An efficient heuristic algorithm that can be applied under any GEV model has been proposed by Dam et al. (2022). This algorithm, called GGX (for Greedy heuristic, Gradient-based local search, and eXchanging), has been shown to identify an optimal solution to most MNL and MMNL instances from classical benchmark sets in a fraction of the time required by MOA to solve the problem to proven optimality. Encouraging computational results

have also been observed on nested logit instances, although the performance of GGX on these instances is difficult to evaluate properly due to the lack of comparison with an exact method in this study.

GGX consists of three phases. First, starting from the trivial set $D_{\boldsymbol{x}} = \emptyset$, a greedy solution is constructed by repeatedly opening the location $d \in D \setminus D_{\boldsymbol{x}}$ that leads to the largest increases in the objective value. In the second phase, a local search within a region of increasing size based on a linear approximation of the objective function at the current solution is iteratively performed. The last phase is a greedy local search in which sets of closed and open locations are iteratively exchanged until a local maximum is found. Dam et al. (2022) allow for at most two pairs of locations to be exchanged at each iteration.

Due to the monotonicity and submodularity of the competitive facility location problem under GEV models (Dam et al., 2022), it follows from Nemhauser et al. (1978) that the greedy heuristic performed in the first phase of GGX is a $(1-1/e)$ approximation algorithm. In other words, the objective value of the solution returned by GGX is guaranteed to be at least ≈ 0.632 times the optimal value. However, like most local search algorithms, GGX provides no stronger theoretical guarantee and can lead to severely suboptimal solutions, even for small instances. We provide such an example next.

Fig. 1. Optimal solution (left, expected market share of 65.61%) and solution obtained with GGX (right, expected market share of 58.66%). The shade of a customer indicates the probability that they select a facility of the firm.

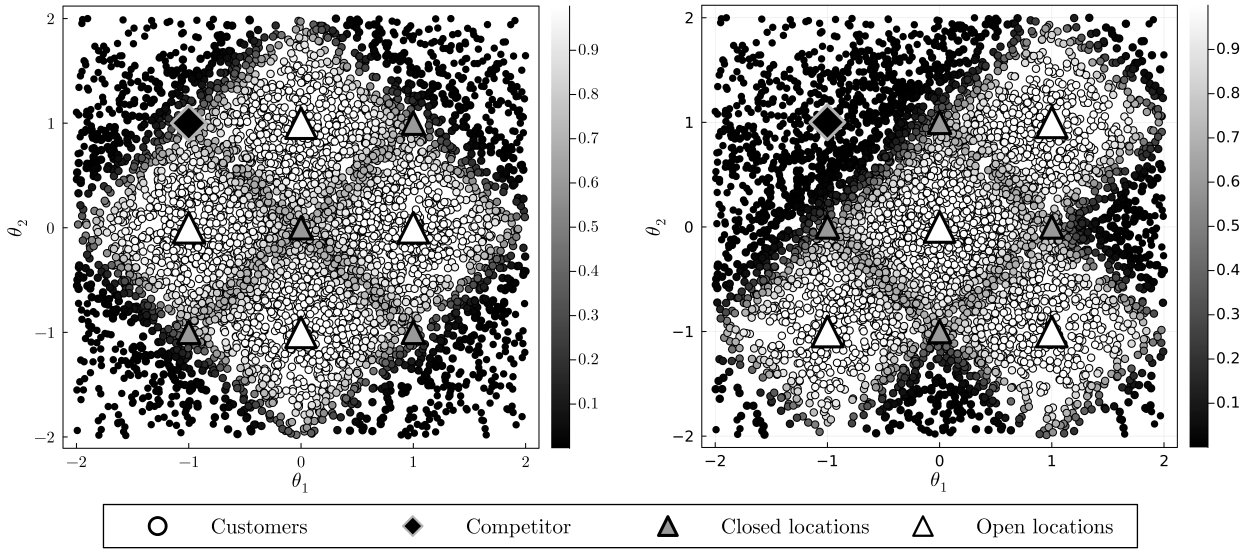


Figure 1 illustrates the behavior of GGX on a conditional MNL instance with $|D| = 8$ candidate locations, a budget of 4 facilities, and $|E| = 1$ existing location that provides the same deterministic utility to each customer. The instance comprises $|N| = 5{,}000$ customers generated according to a censored spherical normal centered at the origin. In this example,

the central location is the first to be included in the greedy solution, followed by the three corner locations. This local maximum cannot be improved by the gradient-based local search of the second phase of GGX. Similarly, it cannot be improved by the greedy exchanging phase unless up to four pairs of locations can be exchanged at each iteration, which corresponds to performing an exhaustive search on the feasible domain X. The solution returned by GGX thus corresponds to opening all the locations that are closed in the optimal solution, and vice versa.

3.3. Sample average approximation method

In the sample average approximation approach introduced by Haase (2009) and studied by Haase and Müller (2013) and Haase et al. (2016) in the context of the MMNL model under a conditional perspective, a set S of scenarios $\{\boldsymbol{\beta}^s\}_{s\in S}$ is sampled from the distribution of the random coefficients $\boldsymbol{\beta}$. For each scenario $s \in S$ and each customer $n \in N$, a realization $\boldsymbol{\varepsilon}^{ns}$ of a vector of $|C|$ IID standard Gumbel variables is drawn. The problem associated with the empirical distribution $\{\boldsymbol{\theta}^n\}_{n\in N}$, $\{\boldsymbol{\beta}^s\}_{s\in S}$, $\{\boldsymbol{\varepsilon}^{ns}\}_{n\in N, s\in S}$ is given by:

$$\max_{\boldsymbol{x}\in X} \frac{1}{|S|} \sum_{n\in N} q_n \sum_{s\in S} \mathbb{1}\Big[\arg\max_{c\in C_{\boldsymbol{x}}}\{u_c(\boldsymbol{\theta}^n,\boldsymbol{\beta}^s,\boldsymbol{\varepsilon}^{ns})\} \in D_{\boldsymbol{x}}\Big]. \tag{3.1}$$

Model (3.1) can be rewritten as the following 0-1 linear program:

$$[\text{SAA}(N,S)] \qquad \max_{\substack{\boldsymbol{x}\in X \\ \boldsymbol{y}\in\{0,1\}^{|N|\times|S|}}} \quad \frac{1}{|S|} \sum_{n\in N} q_n \sum_{s\in S} y_{ns}, \tag{3.2}$$

$$\text{s.t.} \qquad y_{ns} \le \sum_{d\in D} a_d^{ns} x_d, \quad \forall n \in N, \forall s \in S, \tag{3.3}$$

where the binary decision variable y_{ns} indicates whether the simulated customer $(n,s) \in N\times S$ selects one of the new facilities of the firm. Binary coefficients $a_d^{ns} = a_d(\boldsymbol{\theta}^n, \boldsymbol{\beta}^s, \boldsymbol{\varepsilon}^{ns}) := \mathbb{1}[u_d(\boldsymbol{\theta}^n, \boldsymbol{\beta}^s, \boldsymbol{\varepsilon}^{ns}) \ge u_e(\boldsymbol{\theta}^n, \boldsymbol{\beta}^s, \boldsymbol{\varepsilon}^{ns}) \ \forall e \in E]$ specify the candidate locations $d \in D$ that would be preferred by customer $n \in N$ to all the competing facilities $e \in E$ under scenario $s \in S$. Constraints (3.3) require the firm to open at least one such facility $d \in D$ to capture the demand of simulated customer (n,s).

Since the coefficients a_c^{ns} are computed outside of the optimization problem, SAA(N,S) (3.2)-(3.3) is a 0-1 linear programming problem on a set of $|D| + |N| \cdot |S|$ decision variables. This formulation corresponds to a deterministic maximum covering problem and can be solved directly using a MILP solver.

As $|S| \to \infty$, model (3.1) approximates the conditional MMNL model (2.7) with an arbitrarily high precision. The asymptotic convergence property of sample average approximation provides probabilistic guarantees (Kim et al., 2015) that can make this approach preferable to greedy heuristics to avoid severely suboptimal solutions such as the one illustrated in Figure 1. However, since the number of decision variables and constraints of

model SAA(N,S) grow linearly with $|S|$, the quality of the optimal solution's sample average estimate that can be obtained with a reasonable computational budget is often limited.

4. Simulation-based hybrid submodular method

This section presents our model-free approach for solving the choice-based competitive facility location problem. Section 4.1 introduces a deterministic equivalent reformulation of the problem and describes how it can be approximated by simulation. In Section 4.2, we present our simulation-based hybrid submodular method and demonstrate that the submodularity property of the competitive facility location problem is preserved under any RUM model.

4.1. Deterministic equivalent reformulation

Our approach relies on a deterministic equivalent reformulation of model (2.1):

$$[\text{DEQ}] \qquad \max_{\substack{\boldsymbol{x}\in X \\ \boldsymbol{y}\in\{0,1\}^{|P|}}} \sum_{p\in P} \underline{\omega}_p y_p, \tag{4.1}$$

$$\text{s.t.} \quad y_p \leq \sum_{d\in D} a_d^p x_d, \quad \forall p \in P. \tag{4.2}$$

Preference profiles, defined by subsets of candidate locations customers would patronize over the competing facilities, are the basic unit of demand in this maximum covering problem. A binary decision variable y_p represents the capture of each possible preference profile $p \in P$ by the configuration $\boldsymbol{x} \in X$. The binary vectors $a^1, a^2, \ldots, a^{2^{|D|}-1} = [0,\ldots,0,1], [0,\ldots,1,0], \ldots, [1,\ldots,1,1]$ cover the $2^{|D|}-1$ non-trivial preference profiles customers may have. We define P as the set of profiles that can occur with a positive probability. The weight $\underline{\omega}_p$ is the probability $\mathbb{P}_{\boldsymbol{\theta},\boldsymbol{\beta},\boldsymbol{\varepsilon}}\left[a_d(\boldsymbol{\theta},\boldsymbol{\beta},\boldsymbol{\varepsilon}) = a_d^p \ \forall d \in D\right]$ for a randomly selected customer to exhibit preference profile a^p. Normalizing these weights lead to the unit vector $\boldsymbol{\omega}$, whose element $p \in P$ is given by $\omega_p = \mathbb{P}_{\boldsymbol{\theta},\boldsymbol{\beta},\boldsymbol{\varepsilon}}\left[a_d(\boldsymbol{\theta},\boldsymbol{\beta},\boldsymbol{\varepsilon}) = a_d^p \ \forall d \in D | a_d(\boldsymbol{\theta},\boldsymbol{\beta},\boldsymbol{\varepsilon}) \neq \mathbf{0}\right] = \underline{\omega}_p / \sum_{p'\in P} \underline{\omega}_{p'}$. The following proposition shows the validity of this reformulation.

Proposition 1. *DEQ is a valid reformulation of model (2.1).*

Proof. The set X of feasible configurations is the same for both models. Therefore, it suffices to show that the optimal value of DEQ given that the decision vector $\boldsymbol{x}$ is fixed to a feasible configuration $\bar{\boldsymbol{x}} \in X$ is equal to the objective value $F(D_{\bar{\boldsymbol{x}}})$ of the original model:

$$\max_{\boldsymbol{y}\in\{0,1\}^{|P|}} \left\{ \sum_{p\in P} \underline{\omega}_p y_p \middle| y_p \leq \sum_{d\in D} a_d^p \bar{x}_d, \ \forall p \in P \right\}, \tag{4.3}$$

$$= \sum_{p\in P} \underline{\omega}_p \mathbb{1}\left[\sum_{d\in D} a_d^p \bar{x}_d \geq 1\right], \tag{4.4}$$

$$= \sum_{p \in P} \mathbb{P}_{\boldsymbol{\theta},\boldsymbol{\beta},\boldsymbol{\varepsilon}} \left[a_d(\boldsymbol{\theta},\boldsymbol{\beta},\boldsymbol{\varepsilon}) = a_d^p \ \forall d \in D \right] \mathbb{1} \left[\sum_{d \in D} a_d^p \bar{x}_d \geq 1 \right], \tag{4.5}$$

$$= \mathbb{P}_{\boldsymbol{\theta},\boldsymbol{\beta},\boldsymbol{\varepsilon}} \Big[\sum_{d \in D} a_d(\boldsymbol{\theta},\boldsymbol{\beta},\boldsymbol{\epsilon}) \bar{x}_d \geq 1 \Big], \tag{4.6}$$

$$= \mathbb{P}_{\boldsymbol{\theta},\boldsymbol{\beta},\boldsymbol{\varepsilon}} \Big[\sum_{d \in D_{\bar{\boldsymbol{x}}}} a_d(\boldsymbol{\theta},\boldsymbol{\beta},\boldsymbol{\epsilon}) \geq 1 \Big], \tag{4.7}$$

$$= \mathbb{P}_{\boldsymbol{\theta},\boldsymbol{\beta},\boldsymbol{\varepsilon}} \Big[\arg\max_{c \in C_{\bar{\boldsymbol{x}}}} \{ u_c(\boldsymbol{\theta},\boldsymbol{\beta},\boldsymbol{\varepsilon}) \} \in D_{\bar{\boldsymbol{x}}} \Big], \tag{4.8}$$

$$= F(D_{\bar{\boldsymbol{x}}}). \tag{4.9}$$

The restricted model DEQ with $\boldsymbol{x} = \bar{\boldsymbol{x}}$ is given by (4.3) and is maximized by setting each decision variable y_p to its maximum feasible value, hence equation (4.4). We obtain (4.5) by replacing the weights $\underline{\omega}_p$ with their definition. Expression (4.5) can be derived from (4.6) by applying the law of total probability. The last three equations are obtained by subsequently applying the definition of set $D_{\bar{\boldsymbol{x}}}$, coefficients $a_d(\boldsymbol{\theta},\boldsymbol{\beta},\boldsymbol{\epsilon})$ and function $F(D_{\bar{\boldsymbol{x}}})$. □

In general, computing exactly the vector of weights $\underline{\boldsymbol{\omega}}$ is not possible, as it requires to evaluate the following expression for each profile $p \in P$:

$$\underline{\omega}_p = \int_{\boldsymbol{\theta} \in \Theta} \int_{\boldsymbol{\beta} \in B} \int_{\boldsymbol{\varepsilon} \in \mathbb{R}^{|C|}} \mathbb{1} \left[a_d(\boldsymbol{\theta},\boldsymbol{\beta},\boldsymbol{\varepsilon}) = a_d^p \ \forall d \in D \right] p(\boldsymbol{\theta},\boldsymbol{\beta},\boldsymbol{\varepsilon}) d\boldsymbol{\varepsilon} d\boldsymbol{\beta} d\boldsymbol{\theta}. \tag{4.10}$$

However, as long as it is possible to draw samples from the joint distribution of the random variables $(\boldsymbol{\theta},\boldsymbol{\beta},\boldsymbol{\varepsilon})$, $\underline{\boldsymbol{\omega}}$ can be estimated by simulation. Let us consider a set R of realizations $\{(\boldsymbol{\theta}^r,\boldsymbol{\beta}^r,\boldsymbol{\varepsilon}^r)\}_{r \in R}$ of these random variables. The estimated coefficients $\underline{\omega}_p$, $p \in P$, are then given by:

$$\hat{\underline{\omega}}_p = \frac{1}{|R|} \sum_{r \in R} \mathbb{1} \left[a_d(\boldsymbol{\theta}^r,\boldsymbol{\beta}^r,\boldsymbol{\varepsilon}^r) = a_d^p \ \forall d \in D \right], \tag{4.11}$$

which leads to the following approximation of DEQ:

$$\widehat{[\mathrm{DEQ}]} \qquad \max_{\substack{\boldsymbol{x} \in X \\ \boldsymbol{y} \in \{0,1\}^{|\hat{P}|}}} \sum_{p \in \hat{P}} \hat{\underline{\omega}}_p y_p, \tag{4.12}$$

$$\text{s.t.} \quad y_p \leq \sum_{d \in D} a_d^p x_d, \quad \forall p \in \hat{P}. \tag{4.13}$$

Here, $\hat{P} \subseteq P$ is the set of profiles that have been observed at least once over the sample of simulated customers R, i.e., such that $\hat{\underline{\omega}}_p > 0$.

This simulation framework generalizes the classical sample average approximation approach. Indeed, the simulation method of Haase et al. (2016) corresponds to building the set of realizations $R = N \times S$, with $\{(\boldsymbol{\theta}^r,\boldsymbol{\beta}^r,\boldsymbol{\varepsilon}^r)\}_{r \in R} = \{(\boldsymbol{\theta}^n,\boldsymbol{\beta}^s,\boldsymbol{\varepsilon}^{ns})\}_{(n,s) \in N \times S}$. Model SAA($N$,$S$) can then be reformulated as $\widehat{\mathrm{DEQ}}$ by computing the weight of each profile

$p \in \hat{P} = \{p \in P : \exists (n,s) \in N \times S \text{ such that } a^p = a^{ns}\}$ as:

$$\hat{\underline{\omega}}_p = \sum_{\substack{(n,s)\in N\times S \\ a^{ns}=a^p}} \frac{q_n}{|S|}. \tag{4.14}$$

The following proposition shows that model $\widehat{\text{DEQ}}$ generalizes SAA(N,S).

Proposition 2. *The optimal value and the optimal configuration $\boldsymbol{x}^* \in X$ of model $\widehat{DEQ}$, where the weights $\hat{\underline{\omega}}_p$ are given by (4.14), are identical to those of SAA(N,S).*

Proof. The set X of feasible configurations is the same for both models. Therefore, it suffices to show that the restricted models $\widehat{\text{DEQ}}$ and SAA(N,S) obtained by fixing the decision vector $\boldsymbol{x}$ to a feasible configuration $\bar{\boldsymbol{x}} \in X$ share the same optimal value:

$$\max_{\boldsymbol{y}\in\{0,1\}^{|\hat{P}|}} \left\{ \sum_{p\in\hat{P}} \hat{\underline{\omega}}_p y_p \middle| y_p \leq \sum_{d\in D} a_d^p \bar{x}_d,\ \forall p \in \hat{P} \right\}, \tag{4.15}$$

$$= \sum_{p\in\hat{P}} \hat{\underline{\omega}}_p \mathbb{1}\left[\sum_{d\in D} a_d^p \bar{x}_d \geq 1 \right], \tag{4.16}$$

$$= \sum_{p\in\hat{P}} \left(\sum_{\substack{(n,s)\in N\times S \\ a^{ns}=a^p}} \frac{q_n}{|S|} \right) \mathbb{1}\left[\sum_{d\in D} a_d^p \bar{x}_d \geq 1 \right], \tag{4.17}$$

$$= \sum_{(n,s)\in N\times S} \frac{q_n}{|S|} \mathbb{1}\left[\sum_{d\in D} a_d^{ns} \bar{x}_d \geq 1 \right] \tag{4.18}$$

$$= \max_{\boldsymbol{y}\in\{0,1\}^{|N|\times|S|}} \left\{ \sum_{(n,s)\in N\times S} \frac{q_n}{|S|} y_{ns} \middle| y_{ns} \leq \sum_{d\in D} a_d^{ns} \bar{x}_d,\ \forall (n,s) \in N \times S \right\}. \tag{4.19}$$

The restricted models $\widehat{\text{DEQ}}$ and SAA(N,S) with $\boldsymbol{x} = \bar{\boldsymbol{x}}$ are respectively given by (4.15) and (4.19). We obtain (4.17) by replacing the weights $\hat{\underline{\omega}}_p$ with their definition. □

The resulting $\widehat{\text{DEQ}}$ model contains fewer decision variables and constraints than SAA(N,S) as it aggregates under a single profile $p \in P$ all the simulated customers $(n,s) \in N \times S$ sharing the same preferences $a^{ns} = a^p$. We thus call SAAA (for Sample Average Approximation with Aggregation) the method that consists in solving $\widehat{\text{DEQ}}$ directly using a solver. In contrast, SAA denotes the sample average approximation method of Haase et al. (2016).

4.2. Hybrid submodular reformulation

Since the number of possible preference profiles grows exponentially with the number of candidate locations, solving $\widehat{\text{DEQ}}$ directly can be expected to become inefficient when $|D|$ is large and a high number of realizations $|R|$ is observed. In particular, for RUM models

that include independent unbounded additive error terms, such as the MNL model, each of the $2^{|D|}-1$ non-trivial preference profiles can occur with positive probability, even in a conditional setting with only one customer. Most profiles $p \in \hat{P}$, however, typically have a negligible weight $\hat{\underline{\omega}}_p$ and thus only a small impact on the objective function.

The motivation behind the hybrid submodular reformulation is to reduce the computational sensitivity of our method to the number of observed preference profiles by aggregating the least important ones into a single composite customer. To do so, we exploit the submodularity of the objective function and its separability by preference profile:

$$[\widehat{\text{DEQ}}(\hat{P}_1)] \quad \max_{\substack{\boldsymbol{x}\in X \\ \boldsymbol{y}\in\{0,1\}^{|\hat{P}_1|}}} \sum_{p\in\hat{P}_1} \hat{\underline{\omega}}_p y_p + \nu \tag{4.20}$$

$$\text{s.t. } y_p \leq \sum_{d\in D} a_d^p x_d, \qquad \forall p \in \hat{P}_1, \tag{4.21}$$

$$\nu \leq f\left(D_{\bar{\boldsymbol{x}}}\right) + \sum_{d\notin D_{\bar{\boldsymbol{x}}}} \rho_d\left(D_{\bar{\boldsymbol{x}}}\right) x_d - \sum_{d\in D_{\bar{\boldsymbol{x}}}} \rho_d(D\setminus\{d\})(1-x_d), \qquad \forall \bar{\boldsymbol{x}} \in X. \tag{4.22}$$

In this model, only a subset $\hat{P}_1$ of the preference profiles $p \in \hat{P}$ are explicitly represented in the objective function. The captured market share on the remaining profiles $p \in \hat{P}_2 = \hat{P} \setminus \hat{P}_1$ is represented by an auxiliary decision variable ν, which is bounded by an exponential number of submodular cuts.

For a set of open locations $D' \subseteq D$, the contribution to the objective value of the set of profiles $\hat{P}_2$ is given by $f\left(D'\right) = \sum_{p\in\hat{P}_2} \hat{\underline{\omega}}_p \mathbb{1}\left[\sum_{d\in D'} a_d^p \geq 1\right]$. We denote by $\rho_d(D') = f\left(D'\cup\{d\}\right) - f\left(D'\right)$ the marginal gain of opening a new location $d \in D \setminus D'$.

Assuming that the set function f is submodular, the equivalence of models $\widehat{\text{DEQ}}$ and $\widehat{\text{DEQ}}(\hat{P}_1)$ is a direct extension of the works of Nemhauser and Wolsey (1981) and Ljubić and Moreno (2018). The submodularity property of f is shown in the next proposition.

Proposition 3. *The set function $f : 2^D \to \mathbb{R}$, where 2^D denotes the power set of D, is submodular.*

Proof. We show that, for a pair of subsets $D_1, D_2 \subseteq D$ such that $D_1 \subseteq D_2$ and a location $d \in D \setminus D_2$, the marginal gain $\rho_d(D_1)$ is larger than, or equal to, $\rho_d(D_2)$.

$$\rho_d(D_1) = f(D_1\cup\{d\}) - f(D_1) \tag{4.23}$$

$$= \sum_{p\in\hat{P}_2} \hat{\underline{\omega}}_p \mathbb{1}\left[\sum_{d'\in D_1\cup\{d\}} a_{d'}^p \geq 1\right] - \sum_{p\in\hat{P}_2} \hat{\underline{\omega}}_p \mathbb{1}\left[\sum_{d'\in D_1} a_{d'}^p \geq 1\right] \tag{4.24}$$

$$= \sum_{p\in\hat{P}_2} \hat{\underline{\omega}}_p \left(\mathbb{1}\left[\sum_{d'\in D_1\cup\{d\}} a_{d'}^p \geq 1\right] - \mathbb{1}\left[\sum_{d'\in D_1} a_{d'}^p \geq 1\right]\right) \tag{4.25}$$

$$= \sum_{p\in\hat{P}_2} \hat{\underline{\omega}}_p \mathbb{1}\left[a_d^p = 1 \text{ and } a_{d'}^p = 0 \ \forall d' \in D_1\right] \tag{4.26}$$

$$\geq \sum_{p \in \hat{P}_2} \hat{\underline{\omega}}_p \mathbb{1}\left[a_d^p = 1 \text{ and } a_{d'}^p = 0 \ \forall d' \in D_2\right] \tag{4.27}$$

$$= \sum_{p \in \hat{P}_2} \hat{\underline{\omega}}_p \left(\mathbb{1}\left[\sum_{d' \in D_2 \cup \{d\}} a_{d'}^p \geq 1\right] - \mathbb{1}\left[\sum_{d' \in D_2} a_{d'}^p \geq 1\right] \right) \tag{4.28}$$

$$= \sum_{p \in \hat{P}_2} \hat{\underline{\omega}}_p \mathbb{1}\left[\sum_{d' \in D_2 \cup \{d\}} a_{d'}^p \geq 1\right] - \sum_{p \in \hat{P}_2} \hat{\underline{\omega}}_p \mathbb{1}\left[\sum_{d' \in D_2} a_{d'}^p \geq 1\right] \tag{4.29}$$

$$= f(D_2 \cup \{d\}) - f(D_2) \tag{4.30}$$

$$= \rho_d(D_2) \tag{4.31}$$

Since $D_1 \subseteq D_2$, condition $a_{d'}^p = 0 \ \forall d' \in D_2$ implies condition $a_{d'}^p = 0 \ \forall d' \in D_1$, which leads to inequality (4.27). □

A direct corollary of this result is that the objective function of the choice-based competitive facility location is submodular under any RUM model. This finding generalizes the results of Benati (1997) and Dam et al. (2022), who proved the submodularity of the competitive facility location problem's objective function under the MNL model and GEV models, respectively.

Corollary 4. *The set function $F : 2^D \to \mathbb{R}$ is submodular.*

Proof. Let $D' \subseteq D$ be an arbitrary set of locations. By setting $\hat{P} = P$, $\hat{P}_1 = \emptyset$, and $\hat{\underline{\boldsymbol{\omega}}} = \underline{\boldsymbol{\omega}}$, we obtain:

$$F(D') = \sum_{p \in P} \underline{\omega}_p \mathbb{1}\left[\sum_{d \in D'} a_d^p \geq 1\right], \tag{4.32}$$

$$= \sum_{p \in \hat{P}_2} \hat{\underline{\omega}}_p \mathbb{1}\left[\sum_{d \in D'} a_d^p \geq 1\right], \tag{4.33}$$

$$= f(D'). \tag{4.34}$$

The first equation can be derived as in the proof of Proposition 1. We then apply the equalities $\hat{P}_2 = \hat{P} \setminus \hat{P}_1 = P \setminus \emptyset = P$ and $\underline{\boldsymbol{\omega}} = \hat{\underline{\boldsymbol{\omega}}}$. This result indicates that the objective function $F : 2^D \to \mathbb{R}$ of model (2.1) can be expressed as a special case of function $f : 2^D \to \mathbb{R}$, and it thus submodular by Proposition 3. □

As shown by Nemhauser and Wolsey (1981), the submodular cuts (4.22) can equivalently be replaced by the following ones:

$$\nu \leq f\left(D_{\bar{\boldsymbol{x}}}\right) + \sum_{d \notin D_{\bar{\boldsymbol{x}}}} \rho_d\left(\emptyset\right) x_d - \sum_{d \in D_{\bar{\boldsymbol{x}}}} \rho_d(D_{\bar{\boldsymbol{x}}} \setminus \{d\})(1 - x_d), \quad \forall \bar{\boldsymbol{x}} \in X. \tag{4.35}$$

Both cuts (4.22) and (4.35) are implemented using the `lazy-cut callback` routine of CPLEX and are applied globally in the branch-and-bound tree each time an integer solution violating them is found. Constraint $\nu \leq \sum_{p \in \hat{P}_2} \hat{\underline{\omega}}_p$ is added to the model to provide an initial valid upper bound on ν. The pure submodular method obtained by setting $\hat{P}_1 = \emptyset$ can be regarded as a single-cut variant of the submodular method proposed by Ljubić and Moreno (2018) for the conditional MNL model, in which an auxiliary variable ν_n bounds the contribution of each customer $n \in N$ to the objective function.

4.2.1. Set partitioning parameter

The hybrid submodular approach is agnostic to the partition of $\hat{P}$ regarding solution quality, as solving $\widehat{\text{DEQ}}(\hat{P}_1)$ yields an optimal solution to $\widehat{\text{DEQ}}$ for any $\hat{P}_1 \subseteq P$. However, its computational efficiency is highly dependent on this choice. Whereas a large part of the observed preference profiles must be included in $\hat{P}_2$ to significantly reduce the number of decision variables, a high number of submodular cuts may have to be generated in the B&C tree if the auxiliary variable ν aggregates an excessively large fraction of the demand. This motivates the inclusion in $\hat{P}_1$ of the profiles that contribute the most to the objective function. Indexing the observed preference profiles by $p_1, \ldots, p_{|\hat{P}|}$, where $\hat{\underline{\omega}}_{p_1} \geq \hat{\underline{\omega}}_{p_2} \geq \cdots \geq \hat{\underline{\omega}}_{p_{|\hat{P}|}}$, we set $\hat{P}_1 = \{p_1, \ldots, p_{i^*}\}$ for a given number of profiles $i^* \in \{0, \ldots, |\hat{P}|\}$.

As i^* increases, including additional profiles in $\hat{P}_1$ provides a decreasing marginal gain in the ratio $\Omega = \sum_{p \in \hat{P}_1} \hat{\underline{\omega}}_p / \sum_{p \in \hat{P}} \hat{\underline{\omega}}_p$ of the demand that is explicitly represented in the objective function (4.20). Selecting an adequate cardinality for $\hat{P}_1$ thus corresponds to fixing an appropriate cutoff point in an increasing function with diminishing returns. A prime example of this type of problem arises in clustering tasks, where the marginal increase in the explained variation of the data decreases with the number of clusters. In this area, the number of clusters is usually determined based on a variant of the *knee detection method*, where the knee of a function is defined as the maximizer of a curvature measure (Salvador and Chan, 2004).

Curvature is only well-defined for continuous functions. However, a simple approach for approximating the point of maximum curvature for discrete data sets has been proposed by Satopaa et al. (2011). After normalizing the points in the unit square, the so-called *Kneedle* method defines the knee as the point whose distance between the y-axis and x-axis coordinates is maximal. Formally, for a set of points $\{(i, j_i), i \in \{0, \ldots, n\}\}$ respecting $j_i < j_{i+1}\ \forall i \in \{0, \ldots, n-1\}$ and $(j_{i+2} - j_{i+1}) \leq (j_{i+1} - j_i)\ \forall i \in \{0, \ldots, n-2\}$, Kneedle selects a point (i^*, j_{i^*}) such that:

$$i^* \in I^* = \underset{i \in \{0, \ldots, n\}}{\arg\max} \frac{j_i - j_0}{j_n - j_0} - \frac{i}{n}. \tag{4.36}$$

This approach can be applied directly to our parameter selection problem by considering the points $\{(i, \Omega_i), i \in \{0, \ldots, |\hat{P}|\}\}$, where $\Omega_i = \sum_{k=1}^{i} \hat{\omega}_{p_k}$. It leads to selecting a point that

maximizes the difference $\delta_i = \Omega_i - i/|\hat{P}|$. When $|I^*| > 1$, we select $i^* = \max I^*$. Henceforth, we denote the optimal value of the knee detection problem and the resulting relative weight of $\hat{P}_1$ by $\delta^* := \delta_{i^*}$ and $\Omega^* := \Omega_{i^*}$, respectively.

Our preliminary computational experiments have shown that selecting the number i^* of profiles to be included in $\hat{P}_1$ by this knee method consistently provides a good approximation of the optimal cardinality of $\hat{P}_1$ for instances that can benefit from the hybrid submodular reformulation. The characterization of such instances is discussed in Section 5. We call the version of the hybrid submodular approach that integrates the knee detection method the SHS (for Simulation-based Hybrid Submodular) method. It has the practical advantage of being independent of any user-defined parameter, which also makes its performance comparison with other methods more objective.

4.2.2. Illustrative example

In this section, we illustrate the knee detection method and provide computational insights on the impact of the set partitioning parameter on the performance of the hybrid submodular reformulation. We consider randomly generated instances based on the conditional MNL NYC dataset (Aros-Vera et al., 2013) with parameters $\alpha = 1$, $\beta = 0.1$, and $|S| = 1$ scenario (see Section 6.1 for a detailed presentation of this family of instances).

Figure 2 compares the points (i, Ω_i) and the difference $\delta_i = \Omega_i - i/|\hat{P}|$ obtained via the knee detection method and using predetermined weights $\Omega_i \in \{0,0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9,1\}$. The knee is reached at point $(i^*, \Omega^*) = (3611, 0.54)$ and achieves a difference $\delta^* = 0.44$. In this instance, approximately 10% of the observed preference profiles thus account for 54% of the demand.

Fig. 2. Relative weight of $\hat{P}_1$ and value of δ_i by fraction of profiles in $\hat{P}_1$

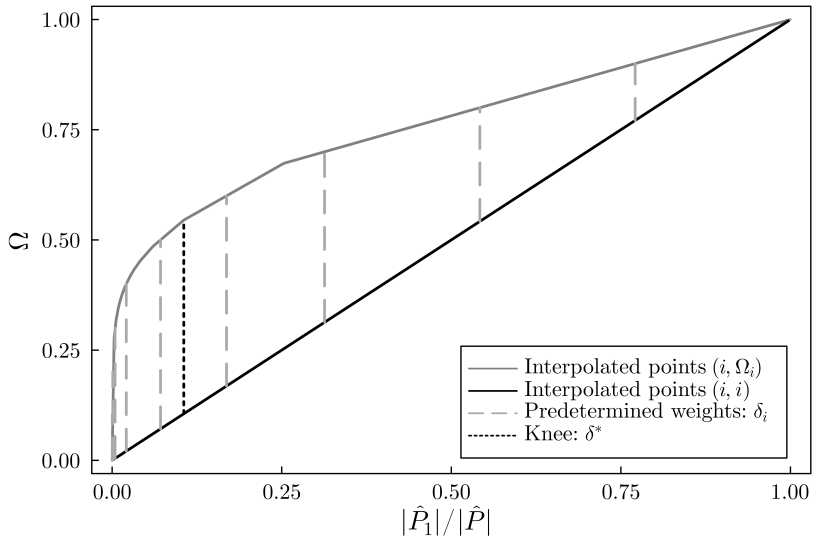


Figure 3 depicts the average cardinality of $\hat{P}_1$ and the average number of submodular cuts generated in the B&C tree when solving model $\widehat{\text{DEQ}}(\hat{P}_1)$ based on the knee detection method and for predetermined weights Ω_i. The average CPU times over 100 instances with a budget of 10 facilities are reported in Figure 4.

Fig. 3. Number of submodular cuts and $|\hat{P}_1|$ by relative weight of $\hat{P}_1$

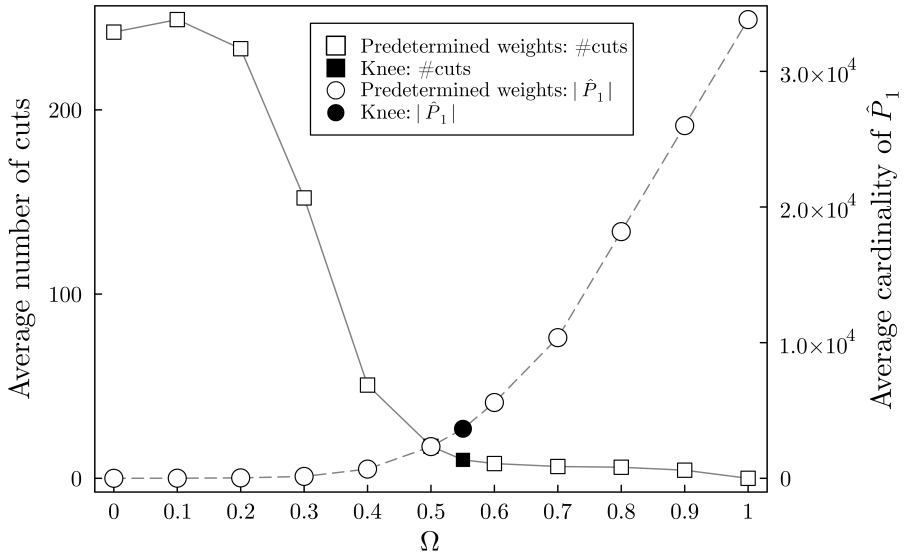
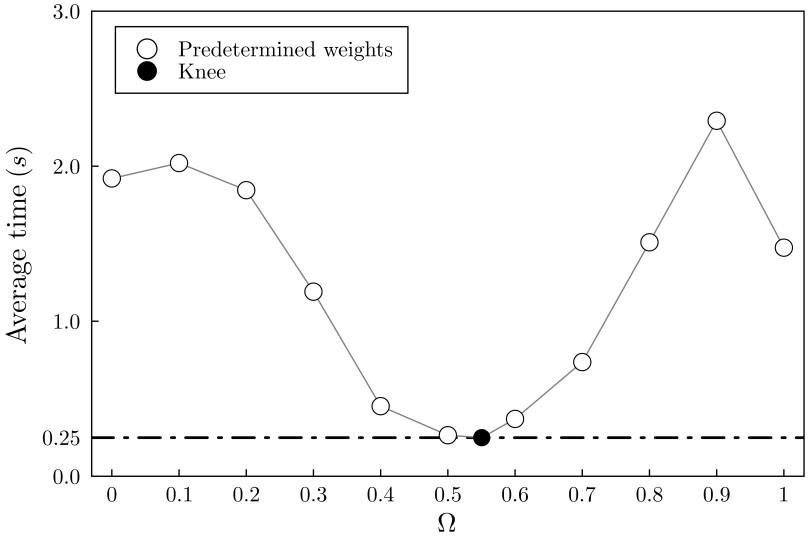


Fig. 4. CPU time by relative weight of $\hat{P}_1$



It appears that the knee method achieves a good trade-off between the size of the model and the number of generated cuts, with an average of $|\hat{P}_1| = 3{,}651$ explicitly represented preference profiles and 10 submodular cuts. The pure submodular approach ($\Omega_i = 0, \hat{P}_1 = \emptyset$) requires generating 242 cuts on average for these instances. SAAA, which is the other extreme

case ($\Omega_i = 1, \hat{P}_1 = \hat{P}$), leads to a fairly large 0-1 linear program with $|\hat{P}_1| = 33{,}817$ demand decision variables on average. These differences in the problem size and the number of generated submodular cuts directly translate into performance differences. Solving $\widehat{\text{DEQ}}$ with SHS required 0.25 second on average, compared to 1.72 and 2.74 seconds SAAA and the pure submodular method, respectively.

5. Information-theoretic analysis

In this section, we adopt an information-theoretic perspective to characterize the level of concentration of the demand across preference profiles, a key attribute of the problem that impacts the expected solution quality and computational performance of SHS. This analysis is motivated by two main considerations.

First, if the customers' behavior is mostly random and all the possible preference profiles are nearly equiprobable, then the coefficients of model $\widehat{\text{DEQ}}$ should converge relatively slowly to those of DEQ. In this case, we can expect simulation-based methods to produce suboptimal solutions for the original problem unless a large sample of preferences profiles is observed.

Second, solving $\widehat{\text{DEQ}}$ to proven optimality should be easier if there exists a small set of profiles that cover a large proportion of the observed demand, as this allows pruning early in the solving process the solutions that do not capture the most important profiles. SHS explicitly exploits this structure through the knee detection method by retaining the most influential profiles in $\hat{P}_1$ and correcting the objective function for the remaining profiles $\hat{P} \setminus \hat{P}_1$ using submodular cuts. SHS is thus expected to offer the most significant computational gain over SAAA when the demand is sufficiently concentrated.

The distribution of the preference profiles defines a categorical random variable W with support P and probability mass function (PMF) $\boldsymbol{\omega}$. We consider the entropy of W to measure the stochasticity of the customers' behavior and characterize the resulting level of concentration of the preference profiles:

$$H(W) = -\sum_{p \in P} \omega_p \log(\omega_p). \tag{5.1}$$

The lowest concentration is achieved when all the preference profiles $p \in P$ share the same normalized weight $\omega_p = \frac{1}{|P|}$. In this case, W follows the discrete uniform distribution over P, and $H(W) = \log(|P|)$ reaches the theoretical upper bound on the entropy of a discrete random variable defined on a finite support of cardinality $|P|$. On the contrary, if a unique profile $p' \in P$ concentrates practically all the demand, say $\omega_{p'} = 1 - \epsilon$, then $H(W) \xrightarrow{\epsilon \to 0} 0$.

5.1. Impact of entropy on solution quality

The implicit assumption underlying any simulation approach in choice-based optimization is that the stochastic model describing the population's choices can be approximated

efficiently from a sample of observed or reported preferences. In the case of model (2.1), it means that it should be possible to faithfully approximate the coefficients $\underline{\omega}_p$, $p \in P$ of the deterministic equivalent formulation DEQ based on a finite set of realizations $\{(\boldsymbol{\theta}^r, \boldsymbol{\beta}^r, \boldsymbol{\varepsilon}^r)\}_{r \in R}$ of the random variables $(\boldsymbol{\theta}, \boldsymbol{\beta}, \boldsymbol{\varepsilon})$. Indeed, solving model $\widehat{\text{DEQ}}$ based on estimated weights $\hat{\underline{\omega}}_p$ that largely deviate from the ground truth $\underline{\omega}_p$ could lead to a severely suboptimal solution to the original choice-based competitive facility location problem and a poor approximation of its optimal value.

A natural measure for evaluating the discrepancy between the objective functions of models $\widehat{\text{DEQ}}$ and DEQ is the expected L_1 distance $\Phi(\boldsymbol{\omega}) := \mathbb{E}\left[||\hat{\boldsymbol{\omega}} - \boldsymbol{\omega}||_1\right]$. This measure corresponds to twice the expected total variation distance $\delta(\hat{W}, W)$ between the random variables $\hat{W}$ and W with PMFs $\hat{\boldsymbol{\omega}}$ and $\boldsymbol{\omega}$. We consider the normalized vector $\hat{\boldsymbol{\omega}}$ obtained through Equation (4.11), with distribution $\hat{\boldsymbol{\omega}} \sim \text{Mult}(|R'|, \boldsymbol{\omega})$, where $|R'| = |R| - \sum_{r \in R} \mathbb{1}[a_d(\boldsymbol{\theta}^r, \boldsymbol{\beta}^r, \boldsymbol{\varepsilon}^r) = 0,\ \forall d \in D]$ is the number of non-trivial preferences profiles in the sample. The L_1 distance $\Phi(\boldsymbol{\omega})$ can be approximated as follows:

$$\Phi(\boldsymbol{\omega}) = \mathbb{E}\left[||\hat{\boldsymbol{\omega}} - \boldsymbol{\omega}||_1\right], \tag{5.2}$$

$$= \mathbb{E}\left[\sum_{p \in P} |\hat{\omega}_p - \omega_p|\right], \tag{5.3}$$

$$= \sum_{p \in P} \mathbb{E}\left[|\hat{\omega}_p - \omega_p|\right], \tag{5.4}$$

$$\xrightarrow{|R'| \to \infty} \sum_{p \in P} \sqrt{\frac{2}{\pi}} \sqrt{\frac{\omega_p(1-\omega_p)}{|R'|}}, \tag{5.5}$$

$$= \sqrt{\frac{2}{\pi |R'|}} \sum_{p \in P} \sqrt{\omega_p(1-\omega_p)}, \tag{5.6}$$

$$=: \tilde{\Phi}(\boldsymbol{\omega}). \tag{5.7}$$

Equation (5.5) is obtained by applying the approximation of the expected absolute error of the estimator of the binomial parameter presented in Blyth (1980) to each term in the summation.

The following result allows us to draw a formal connection between the entropy of W and the expected discrepancy between models $\widehat{\text{DEQ}}$ and DEQ.

Proposition 5. *For any parameter $\eta \in [0,1]$, the optimal solution of the following maximization problem:*

$$\max_{\boldsymbol{\omega} \in [0,\eta]^{|P|}} \sum_{p \in P} \sqrt{\omega_p(1-\omega_p)} \tag{5.8}$$

$$s.t. \sum_{p \in P} \omega_p = \eta, \tag{5.9}$$

is $\boldsymbol{\omega}^* = (\eta/|P|, \ldots, \eta/|P|)$*, with objective value* $\sqrt{\eta(|P| - \eta)}$.

Proof. The proof is by induction on $|P|$. The base case $|P| = 1$ trivially holds, as $\boldsymbol{\omega}^* = (\eta)$ is the only feasible solution. Its objective value is $\sum_{p \in P} \sqrt{\omega_p^*(1 - \omega_p^*)} = \sqrt{\eta(1-\eta)} = \sqrt{\eta(|P| - \eta)}$.

Assuming that the result holds for a set of cardinality $|P| = k$, we demonstrate the case $|P| = k + 1$. To do so, we consider an element $\bar{p} \in P$ and the set $\tilde{P} = P \setminus \{\bar{p}\}$, with $|\tilde{P}| = k$.

$$\max_{\boldsymbol{\omega} \in [0,\eta]^{|P|}} \left\{ \sum_{p \in P} \sqrt{\omega_p(1 - \omega_p)} \middle| \sum_{p \in P} \omega_p = \eta \right\} \tag{5.10}$$

$$= \max_{\boldsymbol{\omega} \in [0,\eta]^{|P|}} \left\{ \sqrt{\omega_{\bar{p}}(1 - \omega_{\bar{p}})} + \sum_{p \in \tilde{P}} \sqrt{\omega_p(1 - \omega_p)} \middle| \omega_{\bar{p}} + \sum_{p \in \tilde{P}} \omega_p = \eta \right\} \tag{5.11}$$

$$= \max_{\omega_{\bar{p}} \in [0,\eta]} \left\{ \sqrt{\omega_{\bar{p}}(1 - \omega_{\bar{p}})} + \max_{\boldsymbol{\omega} \in [0,\eta - \omega_{\bar{p}}]^k} \left\{ \sum_{p \in \tilde{P}} \sqrt{\omega_p(1 - \omega_p)} \middle| \sum_{p \in \tilde{P}} \omega_p = \eta - \omega_{\bar{p}} \right\} \right\} \tag{5.12}$$

$$= \max_{\omega_{\bar{p}} \in [0,\eta]} \left\{ \sqrt{\omega_{\bar{p}}(1 - \omega_{\bar{p}})} + \sqrt{(\eta - \omega_{\bar{p}})(k - (\eta - \omega_{\bar{p}}))} \right\} \tag{5.13}$$

$$=: \max_{\omega_{\bar{p}} \in [0,\eta]} g(\omega_{\bar{p}}) \tag{5.14}$$

Equation (5.13) is obtained by applying the inductive hypothesis to set $\tilde{P}$ with parameter $\eta - \omega_{\bar{p}}$. The inductive hypothesis also stipulates that the optimal solution to this inner problem is given by $\omega_p^* = (\eta - \omega_{\bar{p}})/|\tilde{P}| = (\eta - \omega_{\bar{p}})/k, \ \forall p \in \tilde{P}$. We now seek the maximizer $\omega_{\bar{p}}^* \in [0, \eta]$ of function $g(\omega_{\bar{p}})$.

$$\frac{\partial g(\omega_{\bar{p}})}{\partial \omega_{\bar{p}}} = 0 \iff \frac{1 - 2\omega_{\bar{p}}}{2\sqrt{\omega_{\bar{p}}(1 - \omega_{\bar{p}})}} + \frac{-k + 2\eta - 2\omega_{\bar{p}}}{2\sqrt{(\eta - \omega_{\bar{p}})(k - (\eta - \omega_{\bar{p}}))}} = 0 \tag{5.15}$$

$$\iff \omega_{\bar{p}} = \frac{\eta}{k+1} \tag{5.16}$$

Equation (5.16) is obtained through simple operations by developing expression (5.15) and isolating $\omega_{\bar{p}}$. The only critical point $\omega_{\bar{p}}^* = \eta/(k+1)$ lies in the feasible interval [0,η]. The second derivative of function $g(\omega_{\bar{p}})$ is given by:

$$\frac{\partial^2 g(\omega_{\bar{p}})}{\partial \omega_{\bar{p}}^2} = -\frac{k^2\sqrt{(\eta - \omega_{\bar{p}})(k - \eta + \omega_{\bar{p}})}}{4(\eta - \omega_{\bar{p}})^2(k - \eta + \omega_{\bar{p}})^2} - \frac{\sqrt{\omega_{\bar{p}}(1 - \omega_{\bar{p}})}}{4\omega_{\bar{p}}^2(1 - \omega_{\bar{p}})^2} < 0. \tag{5.17}$$

The maximizer of $g(\omega_{\bar{p}})$ is thus $\omega_{\bar{p}} = \eta/(k+1)$. It follows that $\omega_p^* = (\eta - \omega_{\bar{p}})/k = (\eta - \eta/(k+1))/k = \eta/(k+1) = \eta/|P|, \ \forall p \in \tilde{P}$. The optimal solution to problem (5.8)-(5.9) in the inductive case is thus, as expected, $\boldsymbol{\omega}^* = (\eta/|P|, \ldots, \eta/|P|)$.

The optimal value, in accordance with the expected result, is given by:

$$\sum_{p\in P}\sqrt{\omega_p^*(1-\omega_p^*)} = \sum_{p\in P}\sqrt{\frac{\eta}{|P|}(1-\frac{\eta}{|P|})}, \tag{5.18}$$

$$= \sqrt{\eta(|P|-\eta)}. \tag{5.19}$$

□

A direct consequence of Proposition 5 is that the maximum entropy distribution is also the maximizer of $\tilde{\Phi}(\boldsymbol{\omega})$. This is shown in the following corollary.

Corollary 6. *The estimated expected L_1 distance $\tilde{\Phi}(\boldsymbol{\omega})$ is maximized by the maximum entropy distribution $\boldsymbol{\omega}^* = (1/|P|, 1/|P|, \ldots, 1/|P|)$.*

Proof. We consider the following maximization problem.

$$\arg\max_{\boldsymbol{\omega}\in[0,1]^{|P|}}\left\{\tilde{\Phi}(\boldsymbol{\omega})\middle|\sum_{p\in P}\omega_p = 1\right\} \tag{5.20}$$

$$=\arg\max_{\boldsymbol{\omega}\in[0,1]^{|P|}}\left\{\sqrt{\frac{2}{\pi|R'|}}\sum_{p\in P}\sqrt{\omega_p(1-\omega_p)}\middle|\sum_{p\in P}\omega_p = 1\right\} \tag{5.21}$$

$$=\arg\max_{\boldsymbol{\omega}\in[0,1]^{|P|}}\left\{\sum_{p\in P}\sqrt{\omega_p(1-\omega_p)}\middle|\sum_{p\in P}\omega_p = 1\right\} \tag{5.22}$$

$$=\left(\frac{1}{|P|},\frac{1}{|P|},\ldots,\frac{1}{|P|}\right) \tag{5.23}$$

The optimal solution $\boldsymbol{\omega}^* = (1/|P|, 1/|P|, \ldots, 1/|P|)$ to problem (5.22) is given by Proposition 5, with $\eta = 1$. The maximum estimated expected L_1 distance between $\hat{\boldsymbol{\omega}}$ and $\boldsymbol{\omega}$ is also given by Proposition 5:

$$\tilde{\Phi}(\boldsymbol{\omega}^*) = \sqrt{\frac{2}{\pi|R'|}}\sqrt{1(|P|-1)} = \sqrt{\frac{2(|P|-1)}{\pi|R'|}}. \tag{5.24}$$

□

Conversely, in the least entropy setting where $\omega_{p'} = 1-\epsilon$ for a profile $p' \in P$, each term $\omega_p(1-\omega)$ converges to zero as $\epsilon \to 0$. The measure of discrepancy $\tilde{\Phi}(\boldsymbol{\omega})$ between the original problem and its simulation-based approximation $\widehat{\text{DEQ}}$ thus supports the idea that simulation-based methods should generally yield solutions of lower quality when the entropy of the preference profiles is high.

5.2. Impact of entropy on computational performance

We expect SHS to offer a significant advantage over SAAA when the optimal value δ^* to the distance maximization problem (4.36) is large. Indeed, this value corresponds to the difference between the proportion of the demand explicitly represented in the objective function used in SHS and the proportion of the decision variables y_p, $p \in \hat{P}$, included in the model. A large value of δ^* thus means that most of the demand can be taken into account from the root node in SHS. This tends to limit the number of submodular cuts that have to be generated in the B&C while significantly decreasing the number of decision variables compared to SAAA.

Once again, a formal connection between the entropy of the preference profiles and SHS's computational properties can be established. The following proposition serves this purpose. It shows that δ^* corresponds to the total variation distance between the empirical distribution of the observed preference profiles and the uniform discrete distribution U defined on the same support $\hat{P}_1$.

Proposition 7. *The maximum value* $\delta^* = \max_{i \in \{1,\dots,|\hat{P}|\}} \Omega_i - i/|\hat{P}|$ *is given by* $\delta^* = \delta(\hat{W}, U) = 1/2||\hat{\boldsymbol{\omega}} - \bar{\boldsymbol{\omega}}||_1$, *where* $\bar{\boldsymbol{\omega}} = \{1/|\hat{P}|, \dots, 1/|\hat{P}|\}$ *is the PMF of the uniform discrete distribution* U *over* $\hat{P}$.

Proof. The difference between two subsequent values of δ_i is given by:

$$\delta_i - \delta_{i-1} = \Omega_i - \frac{i}{|\hat{P}|} - \left(\Omega_{i-1} - \frac{i-1}{|\hat{P}|}\right), \tag{5.25}$$

$$= \Omega_i - \Omega_{i-1} - \frac{1}{|\hat{P}|}, \tag{5.26}$$

$$= \sum_{k=1}^{i} \hat{\omega}_{p_k} - \sum_{k=1}^{i-1} \hat{\omega}_{p_k} - \frac{1}{|\hat{P}|}, \tag{5.27}$$

$$= \hat{\omega}_{p_i} - \frac{1}{|\hat{P}|}. \tag{5.28}$$

Since $\{\hat{\omega}_{p_i}\}_{i=1}^{|\hat{P}|}$ is an increasing sequence, it follows from the definition of the knee index i^* that $\hat{\omega}_{p_i} \geq \frac{1}{|\hat{P}|}$ $\forall i \leq i^*$ and that $\hat{\omega}_{p_i} \leq \frac{1}{|\hat{P}|}$ $\forall i > i^*$. The total variation distance $\delta(\hat{W}, U)$ can thus be developed as:

$$\delta(\hat{W}, U) = \frac{1}{2}||\hat{\boldsymbol{\omega}} - \bar{\boldsymbol{\omega}}||_1, \tag{5.29}$$

$$= \frac{1}{2} \sum_{k=1}^{|\hat{P}|} \left| \hat{\omega}_{p_k} - \frac{1}{|\hat{P}|} \right|, \tag{5.30}$$

$$= \frac{1}{2} \left(\sum_{k=1}^{i^*} \left(\hat{\omega}_{p_k} - \frac{1}{|\hat{P}|} \right) + \sum_{k=i^*+1}^{|\hat{P}|} \left(\frac{1}{|\hat{P}|} - \hat{\omega}_{p_k} \right) \right), \tag{5.31}$$

$$= \frac{1}{2}\left(\sum_{k=1}^{i^*} \hat{\omega}_{p_k} - \frac{i^*}{|\hat{P}|} + \frac{|\hat{P}| - i^*}{|\hat{P}|} - \sum_{k=i^*+1}^{|\hat{P}|} \hat{\omega}_{p_k}\right), \tag{5.32}$$

$$= \frac{1}{2}\left(\Omega^* - \frac{i^*}{|\hat{P}|} + \frac{|\hat{P}| - i^*}{|\hat{P}|} - (1 - \Omega^*)\right), \tag{5.33}$$

$$= \Omega^* - \frac{i^*}{|\hat{P}|}, \tag{5.34}$$

$$= \delta^*. \tag{5.35}$$

□

The total variation distance $\delta^* = \delta(\hat{W}, U)$ is linked to the entropy of $\hat{W}$ through the Kullback–Leibler divergence from $\hat{W}$ to U. From Pinsker's inequality (Pinsker, 1964) and by Proposition 7:

$$\delta^* \leq \sqrt{\frac{1}{2} D_{KL}(\hat{W}||U)} = \sqrt{\frac{1}{2}\left(\log|\hat{P}| - H(\hat{W})\right)}. \tag{5.36}$$

Other theoretical results, such as Bretagnolle–Huber inequality (Bretagnolle and Huber, 1978),

$$\delta^* \leq \sqrt{1 - e^{-D_{KL}(\hat{W}||U)}} = \sqrt{1 - e^{H(\hat{W}) - \log|\hat{P}|}}, \tag{5.37}$$

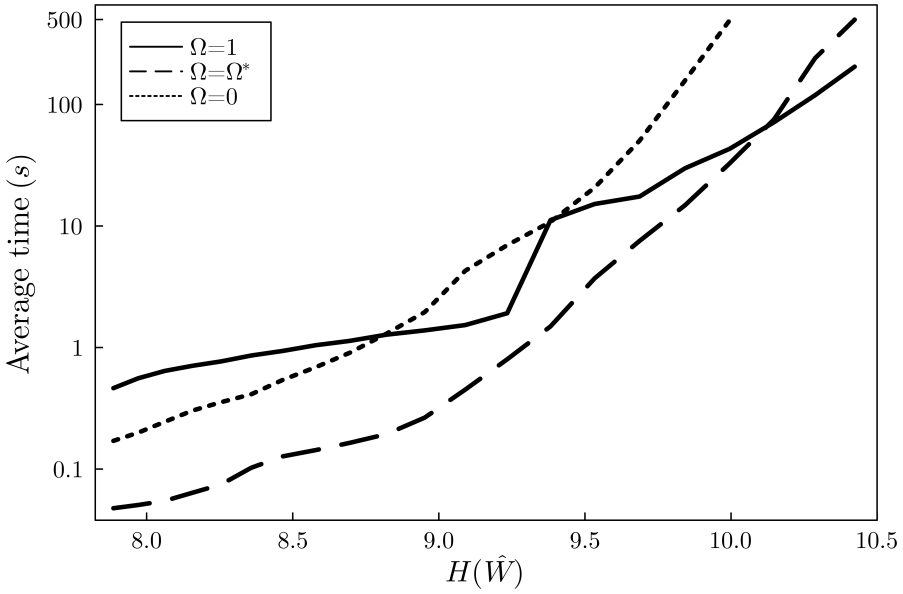
also provide a bound on the total variation distance based on the entropy of $\hat{W}$. The optimal value δ^* of the knee detection method used in SHS is thus bounded by monotonically decreasing functions of the entropy of $\hat{W}$. This result suggests that exploiting the submodularity property of model $\widehat{\text{DEQ}}$ could be of limited interest in the context of observed preference profiles with high entropy.

5.3. Illustrative example

Figure 5 compares the computational performance of SHS ($\Omega = \Omega^*$) with that of SAAA ($\Omega = 1$) and the pure submodular method ($\Omega = 0$) on instances of different entropy levels. The experimental setup is the same as in Section 4.2.2, except that we consider different values of the parameter β, which controls the problem's stochasticity level. We solve 10 randomly generated instances with each method for each value of β between 0.05 and 0.15, in steps of 0.005. As β gets smaller, so does the sensitivity of the customers to distances, which makes their behavior more uncertain (i.e., the relative importance of $\boldsymbol{\varepsilon}$ increases).

In the lowest entropy setting, SHS is the most efficient method, with an average CPU time of 0.05 seconds. The pure submodular method and SAAA take 0.17 and 0.46 seconds per instance on average, respectively. The problem becomes more challenging for all the methods as the entropy increases because so does the number of observed preference profiles, which goes from 19,866 to 60,143 on average between the lowest and highest entropy instances.

Fig. 5. Average CPU time for NYC instances of different levels of entropy



As expected, the submodularity-based methods are more sensitive to entropy than SAAA. SHS ceases to be the most efficient method when β exceeds 0.06, and SAAA then provides the best performance. This switch only occurs when the demand is extremely scattered across preference profiles and the empirical distribution of observed preferences approaches the uniform distribution. For $\beta = 0.06$, $|\hat{P}| = 54{,}269$ different nontrivial preference profiles were observed on average, and the resulting empirical distribution $\hat{\boldsymbol{\omega}}$ was associated with an entropy of $H(\hat{W}) = 10.15$. In comparison, the entropy of the discrete uniform distribution on the same support is 10.90. SHS dominates the pure submodular method for every entropy level.

The impact of the entropy on the solution quality is only partly visible in this set of instances, as the average relative optimality gap of the solution obtained through model $\widehat{\text{DEQ}}$ for the original MNL model is less than 0.05% for each entropy level. The conclusions of the theoretical analysis developed in Section 5.1 are instead illustrated and discussed throughout the computational experiments of the next section.

6. Computational experiments

The purpose of this computational study is twofold. The first objective is to assess the potential of simulation-based methods for efficiently solving competitive facility location problems under RUM models for which model-specific algorithms have already been extensively studied. To do so, SHS, SAAA, and SAA are compared to GGX and MOA on MNL

benchmark instances from the literature. The second objective is to compare the computational performance and solution quality provided by these methods under less restrictive modeling assumptions, which is done based on a new set of generative MMNL instances.

The experiments are conducted on a PC with processor Intel(R) i7-10875H CPU @ 2.30GHz along with 32 GB of RAM operated with Windows 10 Pro. The simulation-based methods SHS, SAAA and SAA are implemented in Julia and are linked to IBM-ILOG CPLEX 20.1.0 optimization routines under default settings. The solver is warm-started with a simple greedy solution. The code and detailed computational results are available at https://github.com/robinlegault/CBCFLP. We use the original MATLAB implementation of MOA and GGX.

6.1. Conditional MNL instances

The experiments of this section are carried out on two datasets on which the most recent methods for the MNL competitive facility location problem have been benchmarked (Dam et al., 2022, Freire et al., 2016, Ljubić and Moreno, 2018, Mai and Lodi, 2020):

- HM14: This dataset has been proposed by Haase and Müller (2014). To produce these problems, the location of the facilities and the customers have been uniformly generated on the square [0,30] × [0,30]. This dataset includes instances with $|D| \in \{25,50,100\}$ available locations and $|N| \in \{50,100,200,400,800\}$ customers. In our experiments, we only consider the largest set of customers $|N| = 800$.
- NYC: This dataset is based on a real-life park-and-ride location problem in New York City. It includes $|D| = 59$ available locations and $|N| = 82{,}341$ customers with weights ranging from 1 to 19. These are generally regarded as the most challenging MNL instances in the literature (Dam et al., 2022, Mai and Lodi, 2020). Holguin-Veras et al. (2012) provide a more detailed presentation of the NYC instances.

For both datasets, the deterministic utility of an open facility for customer $n \in N$ is given by $v_c^n = -\beta\theta_c^n$ for the new locations $c \in D$ and by $v_c^n = -\alpha\beta\theta_c^n$ for the competing locations $c \in E$. The attribute θ_c^n of customer $n \in N$ denotes its distance from facility $c \in C$ in HM14, and aggregates several factors (including travel time, tolls, auto costs, and waiting time) in NYC. The coefficient β controls the importance of the deterministic term in the total utility and, thus, the entropy level of the preference profiles. The competitiveness of the existing facilities is controlled by α. Smaller values of this coefficient lead to more attractive competitors.

Like most previous studies, we consider a unique business constraint, namely that the firm can open at most b facilities. Hence, the set of feasible configurations is given by $X = \{\boldsymbol{x} \in \{0,1\}^{|D|} : \sum_{d\in D} x_d \leq b\}$. As pointed out by Mai and Lodi (2020) for the MNL model, opening additional locations cannot reduce the market share captured by the firm.

This result trivially holds under any RUM model, as the effect of opening new facilities is to relax constraints (4.2) in the deterministic equivalent model DEQ, or analogously constraints (4.13) in model $\widehat{\text{DEQ}}$. An equality constraint can thus equivalently replace the inequality constraint in the definition of X.

For each number of available locations $|D| \in \{25, 50, 100\}$ in the HM14 dataset, we solve an instance for each configuration $(\beta, \alpha, b) \in \{1,2,5,10\} \times \{0.05, 0.1, 0.2\} \times \{2,3,4,5,6,7,8,9,10\}$. These parameters cover those used by Ljubić and Moreno (2018) and Mai and Lodi (2020), except for α. This parameter was taken in a wider interval in both studies, leading to degenerate instances in which more than 99.99% of the market can be captured by a single location. The parameters we consider lead to more reasonable instances for which the optimal value ranges from 2% to 90% of the total market share. Similarly, each configuration $(\beta, \alpha, b) \in \{0.1, 0.15, 0.2, 0.25, 0.5, 0.75, 1, 1.25, 1.5, 1.75, 2\} \times \{0.75,1,1.25\} \times \{2,3,4,5,6,7,8,9,10\}$ is considered for the NYC dataset.

The simulation-based methods are applied with three different numbers of scenarios $|S_1| = 10$, $|S_2| = 100$ and $|S_3| = 1{,}000$ for the HM14 instances and $|S_1| = 1$, $|S_2| = 5$ and $|S_3| = 10$ for NYC. We use the code provided by the authors of the MOA algorithm, which sets $T = \min\{1000, |N|\}$ groups for the HM14 instances and $T = 20$ for the NYC dataset.

The instances are grouped by number of candidate locations and value of β to illustrate the impact of entropy on the performance of each method. The average CPU times are reported in Table 8. Table 9 presents the relative size of the models solved by SAAA and SAA (see columns $|\hat{P}|/|R|$) and by SHS and SAAA (see columns $|\hat{P}_1|/|\hat{P}|$), as well as the optimal value δ^* of the knee detection problem. The last columns of Table 9 report the optimality gap for the simulation-based methods and GGX. For a feasible solution $x \in X$, it is defined as:

$$\text{Gap} = \frac{Z_N(\boldsymbol{x}_N^*) - Z_N(\boldsymbol{x})}{Z_N(\boldsymbol{x}_N^*)}, \tag{6.1}$$

where $Z_N(\cdot)$ and $\boldsymbol{x}_N^*$ respectively denote the objective function and the optimal solution of the conditional MNL model (2.9). The number of B&C nodes explored by the simulation-based methods and the number of submodular cuts generated by SHS are reported in Table 10.

The results in Table 8 indicate that SHS is almost always the fastest among the simulation-based methods. An exception is the highest entropy instances of the HM14 dataset, for which all the simulation-based methods, especially SHS, have relatively large computing times. SAAA consistently dominates SAA, most noticeably for instances with low entropy and a high number of simulated customers, for which the aggregation of simulated customers with identical preference profiles produces the most drastic reduction in the problem size, as reported in Table 9.

Table 8. Average CPU times (seconds) for conditional MNL instances, by entropy level (27 instances per row)

Set	Entropy		SHS			SAAA			SAA			GGX	MOA
	β	$H(\hat{W})$	$\lvert S_1\rvert$	$\lvert S_2\rvert$	$\lvert S_3\rvert$	$\lvert S_1\rvert$	$\lvert S_2\rvert$	$\lvert S_3\rvert$	$\lvert S_1\rvert$	$\lvert S_2\rvert$	$\lvert S_3\rvert$		
HM14 $\lvert D\rvert = 25$	10	3.40	0.00	0.01	0.05	0.01	0.01	0.06	0.02	0.16	2.05	0.15	20.69
	5	3.50	0.00	0.01	0.05	0.01	0.01	0.06	0.02	0.16	2.06	0.15	22.98
	2	3.87	0.01	0.01	0.05	0.01	0.02	0.07	0.02	0.18	2.26	0.19	30.58
	1	4.54	0.01	0.02	0.06	0.02	0.05	0.19	0.04	0.23	2.59	0.17	40.92
HM14 $\lvert D\rvert = 50$	10	4.40	0.01	0.01	0.07	0.01	0.02	0.08	0.03	0.23	2.34	0.22	22.71
	5	4.52	0.01	0.01	0.08	0.01	0.02	0.09	0.03	0.23	2.39	0.21	33.87
	2	4.97	0.02	0.02	0.09	0.02	0.06	0.17	0.03	0.26	3.28	0.22	51.74
	1	5.86	0.13	0.10	0.27	0.09	0.25	1.29	0.11	0.50	4.29	0.20	82.31
HM14 $\lvert D\rvert = 100$	10	5.44	0.06	0.03	0.14	0.03	0.06	0.17	0.06	0.38	3.90	0.46	31.35
	5	5.76	0.20	0.05	0.15	0.06	0.14	0.79	0.09	0.45	4.16	0.46	52.42
	2	6.72	6.56	1.93	2.44	0.17	0.77	8.23	0.23	1.14	13.21	0.45	122.30
	1	8.12	18.96$^{(8)}$	44.16$^{(8)}$	47.36$^{(9)}$	1.21	34.53	78.53$^{(8)}$	1.20	38.95	81.43$^{(8)}$	0.48	942.47
NYC	2.00	3.88	0.02	0.06	0.10	0.07	0.27	0.44	0.30	1.63	3.33	5.95	22.44
	1.75	3.91	0.02	0.06	0.10	0.07	0.29	0.46	0.30	1.57	3.33	6.01	17.12
	1.50	3.96	0.02	0.06	0.10	0.08	0.31	0.49	0.30	1.59	3.43	5.93	11.95
	1.25	4.04	0.02	0.06	0.11	0.09	0.37	0.61	0.31	1.65	3.53	5.95	7.25
	1.00	4.17	0.02	0.07	0.12	0.10	0.43	0.77	0.32	1.80	3.79	5.94	4.49
	0.75	4.42	0.02	0.09	0.15	0.19	0.66	1.14	0.42	2.10	4.48	5.93	2.70
	0.50	4.99	0.03	0.12	0.21	0.32	1.32	2.59	0.50	2.94	6.49	5.95	1.54
	0.25	6.62	0.06	0.32	0.68	1.11	6.09	12.00	1.42	8.85	20.42	5.95	0.75
	0.20	7.25	0.09	0.64	0.82	1.73	9.75	24.89	2.07	11.70	32.23	6.01	0.69
	0.15	8.10	0.14	1.02	2.79	3.44	19.95	41.21	3.32	21.12	57.15	5.92	0.60
	0.10	9.37	0.88	10.55	33.67	5.64	42.25	104.13	5.62	47.17	143.90	5.98	0.53

$^{(\cdot)}$ Number of instances that were not solved to optimality within the time limit of 10 minutes applied to simulation-based methods

As expected, the computing time of SHS is less sensitive to the number of scenarios than SAAA and SAA. In some cases, the CPU time of SHS even decreases with the number of scenarios. This happens for the largest instances of the HM14 dataset with $\beta = 2$, where SHS requires 6.55 seconds on average for 10 scenarios and only 2.44 seconds for 1,000 scenarios. For the same group of instances, multiplying the number of scenarios by 100 (comparing $|S_1|$ and $|S_3|$) increases the CPU time of SAAA and SAA by a factor of 48 and 58, respectively. Supporting the theoretical analysis of Section 5.2, the progressive improvement in the relative performance of SHS compared to SAAA and SAA that comes with more scenarios is accompanied by an increase in the optimal value δ^* of the knee detection problem (see Table 9). In the previously discussed group with 10 scenarios, $\hat{P}_1$ comprises 18.2% of the observed preference profiles accounting for 65.1% of the demand, for a difference of $\delta^* = 46.6\%$. In comparison, with 1,000 scenarios, $\delta^* = 78.5\%$ and the 10.3% most important profiles account for 88.8% of the demand. Furthermore, across both datasets, SHS requires exploring fewer nodes and generating fewer submodular cuts in the B&C tree as the number of scenarios increases. SAAA and SAA solve almost all the instances to optimality at the root node (see Table 10).

Since GGX only relies on local search procedures, its execution time is mostly unaffected by the entropy level, whereas the instances with high entropy are generally more challenging

Table 9. Attributes of model $\widehat{\text{DEQ}}(\hat{P}_1)$ and solution quality compared to GGX for conditional MNL instances, by entropy level (27 instances per row)

Set	Entropy		$\lvert\hat{P}\rvert/\lvert R\rvert$, (%)			$\lvert\hat{P}_1\rvert/\lvert\hat{P}\rvert$, (%)			δ^*, (%)			Gap, (%)			
	β	$H(\hat{W})$	$\lvert S_1\rvert$	$\lvert S_2\rvert$	$\lvert S_3\rvert$	$\lvert S_1\rvert$	$\lvert S_2\rvert$	$\lvert S_3\rvert$	$\lvert S_1\rvert$	$\lvert S_2\rvert$	$\lvert S_3\rvert$	$\lvert S_1\rvert$	$\lvert S_2\rvert$	$\lvert S_3\rvert$	GGX
HM14 $\lvert D\rvert = 25$	10	3.40	0.9	0.1	0.0	31.6	27.0	24.0	46.3	52.9	56.0	.04	.01	**.00**	.10
	5	3.50	1.1	0.2	0.0	27.2	21.1	18.5	49.4	56.4	60.4	.12	.02	**.00**	**.00**
	2	3.87	2.0	0.4	0.1	22.0	16.9	12.2	52.1	64.9	74.5	.18	.03	**.00**	**.00**
	1	4.54	4.7	1.3	0.3	19.0	11.5	7.8	61.9	76.5	85.2	.31	.11	**.00**	**.00**
HM14 $\lvert D\rvert = 50$	10	4.40	2.1	0.3	0.0	32.9	31.2	28.5	40.9	47.0	52.5	.09	**.00**	**.00**	**.00**
	5	4.52	2.7	0.4	0.1	30.7	25.4	20.8	45.0	55.1	62.7	.44	**.00**	.01	.01
	2	4.97	5.5	1.2	0.2	25.6	17.2	11.8	51.9	66.9	77.3	1.01	.03	**.00**	.02
	1	5.86	12.9	4.3	1.3	20.6	13.2	8.6	55.6	72.5	83.1	1.11	.08	.01	**.00**
HM14 $\lvert D\rvert = 100$	10	5.44	6.8	1.2	0.2	29.2	23.9	18.7	45.6	57.0	64.9	.28	.02	**.00**	.40
	5	5.76	10.4	2.3	0.5	24.3	18.7	13.7	48.5	62.5	72.0	.20	.11	**.00**	.16
	2	6.72	22.0	8.6	2.9	18.2	15.4	10.3	46.6	65.6	78.5	1.11	.15	**.01**	.08
	1	8.12	41.7	25.6	14.3	13.6	16.2	9.0	37.1	55.1	69.3	.62	.11	**.02***	.06
NYC	2.00	3.88	3.2	1.3	0.9	12.9	10.2	9.4	70.5	76.3	78.0	**.00**	**.00**	**.00**	**.00**
	1.75	3.91	3.3	1.4	1.0	12.3	10.1	9.0	70.6	76.7	78.4	**.00**	**.00**	**.00**	**.00**
	1.50	3.96	3.6	1.6	1.1	12.5	9.7	8.9	70.7	76.9	78.3	.01	**.00**	**.00**	**.00**
	1.25	4.04	3.9	1.8	1.3	11.3	9.2	8.8	71.0	77.3	79.1	**.00**	**.00**	**.00**	**.00**
	1.00	4.17	4.5	2.2	1.6	10.9	9.0	8.3	71.5	77.0	79.9	**.00**	**.00**	**.00**	**.00**
	0.75	4.42	5.4	3.0	2.3	10.0	9.1	8.3	70.2	77.9	80.6	**.00**	**.00**	**.00**	**.00**
	0.50	4.99	7.8	4.9	3.9	10.2	9.0	7.7	70.9	77.7	80.2	.02	.01	.01	**.00**
	0.25	6.62	15.4	11.6	10.2	13.0	8.9	9.2	63.4	71.4	74.3	.03	**.00**	**.00**	**.00**
	0.20	7.25	19.6	15.3	13.8	11.3	11.7	7.8	60.2	67.8	70.6	**.00**	**.00**	**.00**	**.00**
	0.15	8.10	26.7	21.9	20.1	9.8	9.8	10.1	54.7	61.8	64.6	.01	**.00**	**.00**	**.00**
	0.10	9.37	40.9	35.4	33.3	16.4	17.3	17.5	45.5	51.5	53.8	.01	.03	**.00**	**.00**

* Based on the instances that were solved to optimality within the time limit

for the other methods. The only exception occurs in the NYC dataset, where the CPU time of MOA decreases significantly with the entropy. A possible explanation is the large number of customers (4117 on average for $T = 20$) aggregated in each component g_t of the objective function. When the entropy is low, the customer's choice probabilities are mainly determined by their attributes instead of the random error term. As β increases, so does the heterogeneity of the customers' preferences in each group, which we conjecture makes the aggregated subgradient cuts weaker and negatively affects the performance of this version of MOA. This behavior is not observed on the HM14 dataset, as this set of instances is solved using the multicut version of MOA.

The results of Table 9 support the analysis of Section 5.1, as they illustrate the negative impact of entropy on the solution quality obtained by sample average approximation. This is especially visible for the smallest number of scenarios on the HM14 dataset, where the optimality gap is, on average, five times larger with $\beta = 1$ than with $\beta = 10$. Fortunately, closing the optimality gap does not require an unmanageable number of scenarios, even for high entropy instances. For the largest number of scenarios, SHS remains significantly faster

Table 10. Number of submodular cuts generated and B&C nodes explored by the simulation-based methods for conditional MNL instances, by entropy level (27 instances per row)

Set	Entropy		# submodular cuts			# B&C nodes								
			SHS			SHS			SAAA			SAA		
	β	$H(\hat{W})$	$\lvert S_1\rvert$	$\lvert S_2\rvert$	$\lvert S_3\rvert$	$\lvert S_1\rvert$	$\lvert S_2\rvert$	$\lvert S_3\rvert$	$\lvert S_1\rvert$	$\lvert S_2\rvert$	$\lvert S_3\rvert$	$\lvert S_1\rvert$	$\lvert S_2\rvert$	$\lvert S_3\rvert$
HM14 $\lvert D\rvert = 25$	10	3.40	8.7	6.1	7.0	4.4	3.8	4.1	0.0	0.0	0.0	0.0	0.0	0.0
	5	3.50	6.9	12.7	8.7	2.8	5.0	1.8	0.0	0.0	0.0	0.0	0.0	0.0
	2	3.87	7.2	5.5	4.1	5.2	1.5	0.4	0.0	0.0	0.0	0.0	0.0	0.0
	1	4.54	20.7	11.9	7.5	37.9	11.3	5.2	0.0	0.0	0.0	0.0	0.0	0.1
HM14 $\lvert D\rvert = 50$	10	4.40	8.3	7.0	11.7	10.1	5.5	3.7	0.0	0.0	0.0	0.0	0.0	0.0
	5	4.52	11.1	6.7	7.0	9.5	2.6	1.2	0.0	0.0	0.0	0.0	0.0	0.0
	2	4.97	14.1	11.1	8.1	29.1	12.3	6.4	0.0	0.0	0.0	0.0	0.0	0.0
	1	5.86	66.1	35.0	32.0	360.0	99.7	47.6	0.0	0.0	1.8	0.0	0.0	2.6
HM14 $\lvert D\rvert = 100$	10	5.44	29.3	7.2	15.0	96.6	7.8	7.9	0.0	0.0	0.0	0.0	0.0	0.0
	5	5.76	40.1	14.4	6.9	309.9	20.6	3.6	0.0	0.0	0.0	0.0	0.0	0.0
	2	6.72	249.3	93.1	64.3	7229.4	747.1	223.7	0.0	0.0	0.0	0.0	0.0	0.0
	1	8.12	2990.4	1253.6	391.0	18317.2	11281.8	3429.6	11.8	43.5	20.1	11.3	42.8	21.1
NYC	2.00	3.88	2.1	2.0	2.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	1.75	3.91	2.1	2.2	2.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	1.50	3.96	1.9	2.1	2.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	1.25	4.04	2.1	2.1	2.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	1.00	4.17	2.1	2.1	2.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	0.75	4.42	2.2	2.1	2.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	0.50	4.99	2.1	2.1	2.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	0.25	6.62	2.5	2.5	2.4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	0.20	7.25	3.6	2.7	2.7	0.1	0.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	0.15	8.10	5.6	4.2	3.9	2.7	1.5	1.2	0.0	0.0	0.0	0.0	0.0	0.0
	0.10	9.37	12.3	10.1	7.0	13.9	8.0	5.8	0.0	0.0	0.1	0.0	0.0	0.0

than MOA and even GGX in most cases and provides near-optimal solutions for all the instances, with an average optimality gap that does not exceed 0.02%. In comparison, the optimality gap of GGX reaches 0.40% for the HM14 dataset with $|D| = 100$ and $\beta = 10$. For the same group of instances, SHS with 1,000 scenarios requires 229 times less CPU time than MOA and three times less than GGX and leads to a negligible optimality gap.

These results demonstrate that SHS, despite being a model-free approach, can provide better solution quality and computational performance than the state-of-the-art heuristic method for conditional MNL instances. In addition, its asymptotic optimality property makes it a sound alternative to exact methods for challenging large-scale instances.

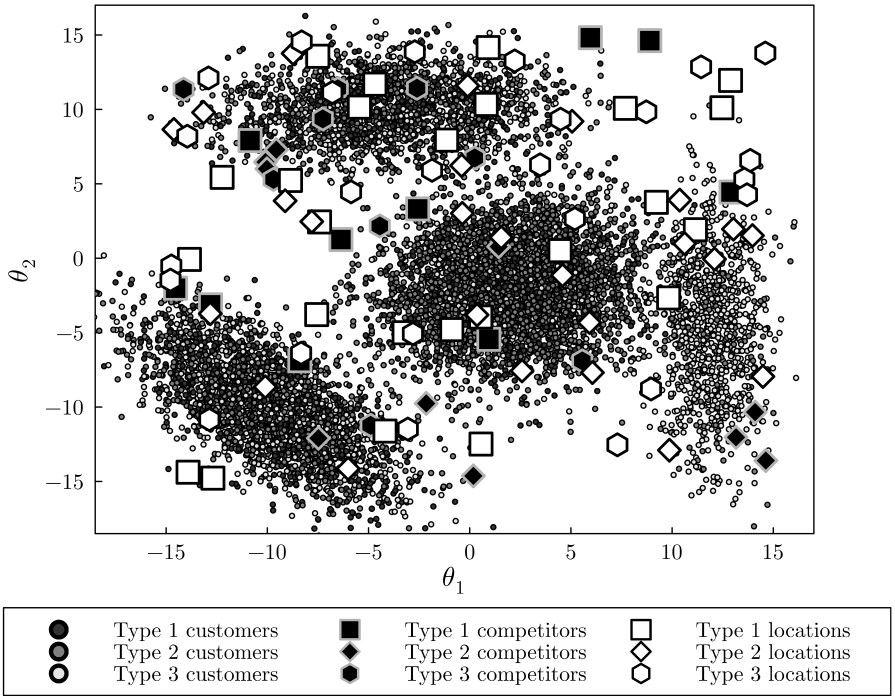
6.2. Generative MMNL instances

This section presents experiments based on a new set of generative MMNL instances, which we call the MIX dataset. We assume that there are three types of locations $l \in L$ and three types of customers $k \in \mathcal{K}$. We consider a fixed set of 10 competing facilities (resulting in $|E| = 3 \cdot 10 = 30$) and 25 available locations of each type (resulting in

$|D| = 3 \cdot 25 = 75$), which were generated uniformly on the square $[-15{,}15] \times [-15{,}15]$. The customers' attributes is given by the random vector $\boldsymbol{\theta} = (\theta_1, \theta_2, K)$, where (θ_1, θ_2) denotes a position in the plane, and the categorical random variable K indicates the type of customer. Fixed parameters $\{\delta_k\}_{k\in\mathcal{K}}$ and $\{\gamma_{kl}\}_{k\in\mathcal{K},l\in L}$ respectively control the level of aversion of each customer type to travel distances, and to each type of location. Those are set to $\boldsymbol{\delta} = [3,1,2], \boldsymbol{\gamma}_1 = [20,60,30], \boldsymbol{\gamma}_2 = [40,20,60]$ and $\boldsymbol{\gamma}_3 = [60,40,20]$. A random multiplier $\beta \sim U[\beta^-, \beta^+]$ controls the weight of the perceived utility in the function $u_c(\boldsymbol{\theta}, \boldsymbol{\varepsilon}) = v_c(\boldsymbol{\theta}) + \varepsilon_c$. For a facility c of type l, the perceived utility is defined as $v_c(\boldsymbol{\theta}) = -\beta(\delta_K M_c(\theta_1, \theta_2) + \gamma_{Kl})$, where $M_c(\theta_1, \theta_2)$ denotes the Manhattan distance separating position (θ_1, θ_2) from location c. As illustrated in Figure 6, the population is distributed across four neighborhoods $j \in \{1{,}2{,}3{,}4\}$. The part of the customers that reside in neighborhood j and the proportion of customers of type k in neighborhood j are specified by two parameters π_j and ρ_{jk}. The spatial distribution of the population in neighborhood j follows a bivariate normal variable with mean $\boldsymbol{\mu}_j$ and covariance matrix Σ_j. The following parameters are used in our experiments:

$$\begin{cases} \boldsymbol{\pi} & = [0.4, 0.3, 0.2, 0.1], \\ [\boldsymbol{\rho}_1, \boldsymbol{\rho}_2, \boldsymbol{\rho}_3, \boldsymbol{\rho}_4] & = \Big[[0.2, 0.7, 0.1], [0.3, 0.4, 0.3], [0.3, 0.4, 0.3], [0.0, 0.2, 0.8]\Big], \\ [\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \boldsymbol{\mu}_3, \boldsymbol{\mu}_4] & = \Big[[2, -2], [-10, -10], [-4, 10], [12, -5]\Big], \\ [\Sigma_1, \Sigma_2, \Sigma_3, \Sigma_4] & = \left[\begin{bmatrix} 9 & 1 \\ 1 & 9 \end{bmatrix}, \begin{bmatrix} 9 & -6 \\ -6 & 9 \end{bmatrix}, \begin{bmatrix} 16 & 1 \\ 1 & 4 \end{bmatrix}, \begin{bmatrix} 2 & 0 \\ 0 & 21 \end{bmatrix}\right]. \end{cases}$$

Fig. 6. Visualization of the MIX dataset based on a sample of 10,000 customers.



To solve the generative MMNL model with MOA and GGX, we sample a set N of realizations $\{(\boldsymbol{\theta}^n, \beta^n)\}_{n\in N}$ of the customers' random attributes $\boldsymbol{\theta}$ and of the random coefficient β. For SHS, SAAA and SAA, the resulting conditional MNL model is then approximated as in the previous section through a set S of realizations $\{\boldsymbol{\varepsilon}\}_{s\in S}$ of the error component for each customer $n \in N$.

For the same number of realizations, solving the MNL model exactly with MOA provides the best solution on average but is more expensive than solving it heuristically with GGX, which in turn is more expensive than solving the simulation-based model with SHS. To account for the different computational performance and solution quality of each approach, different numbers of customers are considered for each of them. We generate $|N| \in \{125, 250, 500, 1000, 2000\}$ customers for MOA, $|N| \in \{250, 1000, 4000, 16000, 64000\}$ for GGX, and $|N| \in \{16000, 32000, 64000, 128000, 256000\}$ with $|S| = 5$ scenarios for SHS, SAAA and SAA. We study the model with four different pairs of bounds $(\beta^-, \beta^+) \in \{(0.125,0.25), (0.25,0.5), (0.5,1), (1,2)\}$. The average entropy of the observed preference profiles for the simulation-based instances generated with these parameters ranges from 5.11 to 8.14. For each entropy level and each sample size, five instances are generated and solved with five different budgets $b \in \{5, 10, 15, 20, 25\}$.

Table 11 reports the CPU times for each method and the relative size of the different simulation-based models. For a sample of customers N and a set of scenarios S, let us respectively denote by $\boldsymbol{x}^*_{NS}$, $\boldsymbol{x}^G_N$ and $\boldsymbol{x}^*_N$ the optimal solution of the simulation-based model $\widehat{\text{DEQ}}$, the heuristic solution returned by GGX, and the optimal solution of the conditional MNL model. Also, let $Z_{NS}(\cdot)$, $Z_N(\cdot)$ and $Z(\cdot)$ be the objective functions of the simulation-based, conditional MNL, and generative MMNL models. Table 12 presents the average values of solutions $\boldsymbol{x}^*_{NS}$, $\boldsymbol{x}^G_N$ and $\boldsymbol{x}^*_N$ for each objective function. $Z(\cdot)$ is evaluated based on an independent sample of $|N|$ = 1,000,000 customers.

The results presented in Table 11 indicate that SHS is consistently the most efficient simulation-based method for the MIX dataset. In the lowest entropy settings, the advantage of SHS over SAAA is only marginal, as the number of observed preference profiles remains limited even for a very large number of simulated customers. For example, for $(\beta^-, \beta^+) = (1,2)$ with $|N|$ = 256,000 customers and $|S| = 5$ scenarios, aggregating the simulated customers by preference profile reduces the model's size from $|R|$ = 1,280,000 to $|\hat{P}|$ = 2,840 (i.e., $|\hat{P}|/|R| \approx 0.2\%$). However, for parameters $(\beta^-, \beta^+) = (0.125, 0.25)$ and the same number of simulated customers, the average number of observed preference profiles reaches $|\hat{P}|$ = 145,175 (i.e., $|\hat{P}|/|R| \approx 11.3\%$). This makes solving model $\widehat{\text{DEQ}}$ with CPLEX computationally demanding. In this setting, $\hat{P}$ contains a very high number of preference profiles with negligible weight, and aggregating their contribution to the objective value becomes increasingly profitable computationally. The subset $\hat{P}_1$ of preference profiles that are explicitly represented in model $\widehat{\text{DEQ}}(\hat{P}_1)$ has an average cardinality of 8,998

Table 11. Average CPU times (seconds) and attributes of model $\widehat{\text{DEQ}}(\hat{P}_1)$ for the MIX dataset, by entropy level and sample size (25 instances per row)

Entropy			$\|N\|$, (thousands)			Model $\widehat{\text{DEQ}}(\hat{P}_1)$, (%)			Time				
β^-	β^+	$H(\hat{W})$	SHS	GGX	MOA	$\|\hat{P}\|/\|R\|$	$\|\hat{P}_1\|/\|\hat{P}\|$	δ^*	SHS	SAAA	SAA	GGX	MOA
			16	0.25	0.125	1.6	13.6	74.2	0.02	0.03	0.33	0.73	0.43
			32	1	0.25	1.0	12.6	77.1	0.03	0.05	0.65	1.20	1.27
1	2	5.11	64	4	0.5	0.6	11.5	79.4	0.05	0.07	1.76	7.40	7.50
			128	16	1	0.4	10.3	81.7	0.10	0.12	3.81	24.26	56.59
			256	64	2	0.2	9.6	83.4	0.18	0.19	7.27	47.17	126.83
			16	0.25	0.125	3.4	11.2	75.7	0.03	0.06	0.36	0.73	0.47
			32	1	0.25	2.4	9.7	79.0	0.04	0.08	0.73	1.20	1.78
0.5	1	5.64	64	4	0.5	1.6	8.6	81.7	0.06	0.12	1.75	7.69	9.82
			128	16	1	1.1	7.6	84.1	0.10	0.19	3.71	24.44	69.35
			256	64	2	0.7	6.9	86.0	0.20	0.30	8.06	47.29	135.05
			16	0.25	0.125	9.5	10.0	71.2	0.07	0.28	0.63	0.71	1.02
			32	1	0.25	7.3	8.1	74.8	0.12	0.61	1.39	1.25	3.06
0.25	0.5	6.73	64	4	0.5	5.6	7.4	77.9	0.16	1.02	3.07	7.45	14.76
			128	16	1	4.2	6.8	80.8	0.21	1.23	7.42	24.16	95.18
			256	64	2	3.1	6.1	83.2	0.34	7.15	18.59	47.25	182.12
			16	0.25	0.125	21.9	8.7	59.5	0.49	1.64	1.87	0.73	1.74
			32	1	0.25	18.5	9.2	63.7	0.66	5.68	7.28	1.26	6.06
0.125	0.25	8.14	64	4	0.5	15.9	7.7	67.4	0.78	6.87	11.66	7.17	20.70
			128	16	1	13.4	6.7	70.7	1.31	52.98	59.88	23.75	115.48
			256	64	2	11.3	6.2	73.7	2.44	134.20	218.34	48.03	206.64

(6.2% of the observed preference profiles) and accounts for approximately 80% of the demand ($\Omega^* = |\hat{P}_1|/|\hat{P}| + \delta^* \approx 79.9\%$). Although $\hat{P}_2$ regroups 136,177 preference profiles on average, SHS terminates after generating only 23.6 submodular cuts on average (see Table 13). As a result, the average CPU time of SHS for this group of instances is 2.44 seconds compared to 134.20 seconds and 218.34 seconds for SAAA and SAA.

The objective value of solution $\boldsymbol{x}^*_{NS}$ for the generative MMNL model stabilizes at a near-optimal level between $|N| = 64{,}000$ and $|N| = 256{,}000$ (see figures highlighted in bold in Table 12). SHS generally solves these instances in less than one second. Similar quality solutions can be obtained in approximately 47 seconds using GGX with $|N| = 64{,}000$. MOA is the worst-performing method for the MIX dataset. Indeed, solving the conditional MNL model to optimality for $|N| = 2{,}000$ customers is three to four orders of magnitude longer than solving the simulation-based model with SHS for $|N| = 16{,}000$ and $|S| = 5$ and consistently provides solutions of lower quality for the generative MMNL model.

The results of this section demonstrate the clear advantage of our simulation-based method over model-specific algorithms and classical sample average approximation for problems based on flexible RUM models and a generative perspective.

Table 12. Average objective value evaluated on an independent sample of 1,000,000 customers (expected market share, in %) under the simulation-based, conditional MNL and generative MMNL models for the MIX dataset, by entropy level and sample size (25 instances per row)

Entropy			$\lvert N\rvert$, (thousands)			SHS			GGX		MOA	
β^-	β^+	$H(\hat{W})$	SHS	GGX	MOA	$Z_{NS}(\boldsymbol{x}^*_{NS})$	$Z_N(\boldsymbol{x}^*_{NS})$	$Z(\boldsymbol{x}^*_{NS})$	$Z_N(\boldsymbol{x}^G_N)$	$Z(\boldsymbol{x}^G_N)$	$Z_N(\boldsymbol{x}^*_N)$	$Z(\boldsymbol{x}^*_N)$
			16	0.25	0.125	62.59	62.54	62.62	63.38	61.02	65.59	60.29
			32	1	0.25	62.71	62.70	62.62	62.84	62.35	63.38	61.02
1	2	5.11	64	4	0.5	62.49	62.50	**62.63**	62.78	62.51	63.55	61.32
			128	16	1	62.52	62.53	62.62	62.55	62.62	62.84	62.35
			256	64	2	62.59	62.59	**62.63**	62.50	**62.63**	63.02	62.44
			16	0.25	0.125	60.63	60.58	60.70	60.98	59.28	63.68	58.71
			32	1	0.25	60.64	60.67	60.70	61.09	60.44	60.98	59.28
0.5	1	5.64	64	4	0.5	60.65	60.68	60.70	60.81	60.68	61.25	60.16
			128	16	1	60.66	60.67	**60.71**	60.58	60.70	61.09	60.44
			256	64	2	60.72	60.74	**60.71**	60.68	**60.71**	61.35	60.61
			16	0.25	0.125	56.96	56.88	56.87	58.37	55.90	58.69	55.04
			32	1	0.25	56.90	56.95	56.87	56.41	56.70	58.37	55.90
0.25	0.5	6.73	64	4	0.5	56.86	56.84	**56.88**	56.87	56.86	57.17	56.33
			128	16	1	56.89	56.92	**56.88**	56.88	56.87	56.41	56.70
			256	64	2	56.92	56.92	**56.88**	56.84	**56.88**	57.52	56.81
			16	0.25	0.125	51.43	51.29	51.23	53.05	50.57	52.24	49.89
			32	1	0.25	51.04	51.07	51.24	50.85	51.09	53.05	50.57
0.125	0.25	8.14	64	4	0.5	51.24	51.27	**51.25**	51.28	51.21	52.23	50.94
			128	16	1	51.18	51.20	**51.25**	51.31	**51.25**	50.85	51.09
			256	64	2	51.25	51.24	**51.25**	51.28	**51.25**	51.16	51.18

Table 13. Number of submodular cuts and B&C nodes generated by the simulation-based methods for the MIX dataset, by entropy level and sample size (25 instances per row)

Entropy			$\lvert N\rvert$ (thousands)	# submodular cuts	# B&C nodes		
β^-	β^+	$H(\hat{W})$		SHS	SHS	SAAA	SAA
			16	15.1	3.2	0.0	0.0
			32	8.8	1.6	0.0	0.0
1	2	5.11	64	3.7	0.0	0.0	0.0
			128	10.3	3.0	0.0	0.0
			256	15.8	3.2	0.0	0.0
			16	10.2	7.2	0.0	0.0
			32	8.0	3.4	0.0	0.0
0.5	1	5.64	64	4.9	1.2	0.0	0.0
			128	3.8	0.4	0.0	0.0
			256	10.1	1.5	0.0	0.0
			16	13.3	12.5	0.0	0.0
			32	11.2	11.6	0.0	0.0
0.25	0.5	6.73	64	8.7	7.6	0.0	0.0
			128	6.5	3.7	0.0	0.0
			256	5.7	2.6	0.0	0.0
			16	47.5	166.0	0.0	0.0
			32	40.2	94.9	0.3	0.0
0.125	0.25	8.14	64	22.9	48.9	0.2	1.8
			128	23.8	33.7	1.0	1.4
			256	23.6	26.9	0.0	0.0

7. Conclusion

This paper presents a model-free approach for solving probabilistic competitive facility location problems with utility-maximizing customers. The proposed methodology is based on a new deterministic equivalent reformulation of the problem. Approximating this model by simulation leads to a generalization of the classical sample average approximation model in which preference profiles replace simulated customers as the basic unit of demand.

We show that the objective function of the competitive facility location problem is submodular under any RUM model. We exploit this property, which is preserved by our simulation-based model, in the so-called hybrid submodular reformulation. The idea of this reformulation is to partition the preference profiles into two sets that are respectively explicitly represented in the model and aggregated into a unique composite customer whose contribution to the objective is bounded by submodular cuts.

We propose constructing this partition using a knee detection method, resulting in an algorithm that does not rely on user-defined parameters. We develop an information-theoretic analysis of the problem and draw connections between the entropy of the preference profiles and the computational performance of our approach. Computational experiments on conditional MNL and generative MMNL instances show that our method can significantly outperform state-of-the-art model-specific algorithms in terms of computing time and solution quality. A key takeaway from the experiments is that our branch-and-cut methodology scales significantly better than the classical sample average approximation method with respect to the number of simulated customers. Combined with the absence of restrictive modeling assumptions of our approach, this opens the way to integrating larger and more complex populations in choice-based competitive facility location problems.

Regarding future research directions, the proposed methodology could be generalized to a multicut version that would include multiple auxiliary variables bounded by independent submodular cuts. This approach would require a different set partitioning approach than the knee detection method we use in our single-cut version. Relevant modeling extensions of this work for real-life applications include adding capacity constraints on the facilities and accounting for the anticipated reaction of the competitors to the firm's decisions.

Acknowledgments

This research was supported by FRQNT and IVADO through scholarships to the first author. The second author benefits from funding from a Canada Research Chair. We wish to thank Tien Mai and Andrea Lodi for sharing the code of the MOA and GGX algorithms as well as their benchmark instances.

Conclusion

In this thesis, we tackled two distinct network design problems: the single-sink fixed-charge transportation problem (SSFCTP) and the choice-based competitive facility location problem (CBCFLP). As previously noticed in the literature, these problems are directly related to the binary knapsack problem (KP) and the maximum covering problem (MCP), respectively. However, these structural similarities to classical combinatorial optimization problems were only superficially leveraged by existing algorithms. By further exploring and formalizing these connections, we proposed novel model reformulations, proved theoretical properties, and devised efficient solution methods for both problems.

In the case of the SSFCTP, we developed the *knapsack transformation algorithm* (KTA), an exact method constituted of three phases. First, in the heuristic phase, we consider a Lagrangian-inspired relaxation of a binary nonlinear reformulation of the SSFCTP. This relaxation, which can be expressed as a KP, is iteratively solved for different multiplier values. This yields a set of optimal solutions to KPs from which we extract several bounds, which are then improved based on a new dominance relation and a strengthened linear relaxation of the SSFCTP. Second, we execute a filtering phase in which the number of units sent to the sink can be fixed to either zero or the source node's capacity on several arcs. Finally, the exact phase consists in solving a residual set of KPs from an original pool that was reduced throughout the heuristic phase of KTA. This residual set is usually empty or contains a very small number of problems whose size is also very small due to the preceding filtering phase. For each group of instances in our experiments, KTA required solving less than two KPs with a size equal to or inferior to that of the original SSFCTP. As solving these KPs represents most of the computation time of KTA, we conclude that the SSFCTP should now be regarded, like the KP (Pisinger, 2005), as one of the easiest $\mathcal{NP}$-hard problems.

The solution approach we developed for the CBCFLP is called the *simulation-based hybrid submodular* (SHS) method. This algorithm applies to a sample average estimate of a deterministic equivalent reformulation of the CBCFLP as a large-scale MCP. Starting from a set of simulated customers, the first step of SHS is to aggregate the customers sharing the same preference profile, defined as the set of available locations they would patronize over the competition. The weight of each preference profile in the resulting MCP approximates

a coefficient of our deterministic equivalent reformulation. The second step of SHS is to partition these preference profiles into two groups. The first one, constituted of the preference profiles with the largest weight, is kept as is in the model. The least prevalent profiles are aggregated and their contribution to the objective, represented by a single auxiliary variable, is bounded by submodular cuts. The resulting model, which we describe as an *hybrid submodular reformulation* of the MCP, is then solved using a branch-and-cut methodology. Our empirical results indicate that SHS generally performs better than the state of the art on large instances, except in cases where customer behavior is nearly entirely random. Furthermore, SHS scales significantly better than classical sample average approximation with respect to the number of simulated customers. These conclusions are supported by a theoretical analysis relating the level of entropy of the preference profiles in the population to the structure of the problem and the performance of simulation-based methods.

The two articles in this thesis open very different, but equally promising perspectives for future research. First, the huge leap in computational efficiency achieved by KTA compared with the previous state-of-the-art methods for the SSFCTP (instances previously requiring minutes of computation can now be solved in a few milliseconds) significantly increases the potential of methodologies necessitating to solve SSFCTPs repeatedly. Known examples of such solution frameworks for more challenging problems include Lagrangian relaxation (Görtz and Klose, 2007) and column generation (Zhao et al., 2018) approaches to the FCTP. Our results could also motivate the investigation of further connections between the SSFCTP and other network design problems of interest.

Regarding the second article, we believe that the most promising lines of research stemming from our work concern modeling and methodological extensions of the SHS method. Developing a multi-cut variant of SHS and studying the properties of the CBCFLP (especially submodularity) when adding capacity constraints to the model constitute two potentially fruitful research directions. Furthermore, we only considered static competition in this thesis. Although this modeling framework is the most widespread in the competitive facility location literature (Plastria, 2001), it is also quite unrealistic for several applications. It would thus be interesting to investigate the potential of simulation-based methods in the more general context of sequential competitive facility location (Qi et al., 2022).

References

Akhavan Kazemzadeh, M. R., Bektaş, T., Crainic, T. G., Frangioni, A., Gendron, B., and Gorgone, E. Node-based lagrangian relaxations for multicommodity capacitated fixed-charge network design. *Discrete Applied Mathematics*, 308(C):255–275, 2022.

Alidaee, B. and Kochenberger, G. A. A note on a simple dynamic programming approach to the single-sink, fixed-charge transportation problem. *Transportation Science*, 39(1): 140–143, 2005.

Aros-Vera, F., Marianov, V., and Mitchell, J. E. p-Hub approach for the optimal park-and-ride facility location problem. *European Journal of Operational Research*, 226(2):277–285, 2013.

Balas, E. and Zemel, E. An algorithm for large zero-one knapsack problems. *Operations Research*, 28(5):1130–1154, 1980.

Balinski, M. L. Fixed-cost transportation problems. *Naval Research Logistics Quarterly*, 8 (1):41–54, 1961.

Benati, S. Submodularity in competitive location problems. *Ricerca Operativa*, 26:3–34, 1997.

Benati, S. and Hansen, P. The maximum capture problem with random utilities: Problem formulation and algorithms. *European Journal of Operational Research*, 143(3):518–530, 2002.

Bhat, C. R. and Guo, J. A mixed spatially correlated logit model: formulation and application to residential choice modeling. *Transportation Research Part B: Methodological*, 38 (2):147–168, 2004.

Birge, J. R. and Louveaux, F. V. A multicut algorithm for two-stage stochastic linear programs. *European Journal of Operational Research*, 34(3):384–392, 1988.

Blyth, C. R. Expected absolute error of the usual estimator of the binomial parameter. *The American Statistician*, 34(3):155–157, 1980.

Bonami, P., Biegler, L. T., Conn, A. R., Cornuéjols, G., Grossmann, I. E., Laird, C. D., Lee, J., Lodi, A., Margot, F., Sawaya, N., and Wächter, A. An algorithmic framework for convex mixed integer nonlinear programs. *Discrete Optimization*, 5(2):186–204, 2008.

Bretagnolle, J. and Huber, C. Estimation des densités: risque minimax. *Séminaire de probabilités de Strasbourg*, 12:342–363, 1978.

Brown, S. Retail location theory: the legacy of harold hotelling. *Journal of Retailing*, 65(4): 450–468, 1989.

Chanintrakul, P., Coronado Mondragon, A. E., Lalwani, C., and Wong, C. Y. Reverse logistics network design: a state-of-the-art literature review. *International Journal of Business Performance and Supply Chain Modelling*, 1(1):61–81, 2009.

Christensen, T. R. L., Andersen, K. A., and Klose, A. Solving the single-sink, fixed-charge, multiple-choice transportation problem by dynamic programming. *Transportation Science*, 47(3):428–438, 2013.

Chu, C. A paired combinatorial logit model for travel demand analysis. In *Proceedings of the Fifth World Conference on Transportation Research, 1989*, volume 4, pages 295–309, 1989.

Church, R. and ReVelle, C. The maximal covering location problem. In *Papers of the regional science association*, volume 32, pages 101–118. Springer-Verlag Berlin/Heidelberg, 1974.

Cordeau, J.-F., Klibi, W., and Nickel, S. Logistics network design. *Network Design with Applications to Transportation and Logistics*, pages 599–625, 2021.

Crainic, T. G. Service network design in freight transportation. *European Journal of Operational Research*, 122(2):272–288, 2000.

Crainic, T. G. and Gendron, B. Exact methods for fixed-charge network design. In *Network Design with Applications to Transportation and Logistics*, pages 29–89. Springer, 2020.

Crainic, T. G., Gendreau, M., and Gendron, B. *Network design with applications to transportation and logistics*. Springer, 2021.

Dam, T. T., Ta, T. A., and Mai, T. Submodularity and local search approaches for maximum capture problems under generalized extreme value models. *European Journal of Operational Research*, 300(3):953–965, 2022.

Davis, J. M., Topaloglu, H., and Williamson, D. P. Pricing problems under the nested logit model with a quality consistency constraint. *INFORMS Journal on Computing*, 29(1): 54–76, 2017.

Drezner, T. A review of competitive facility location in the plane. *Logistics Research*, 7: 1–12, 2014.

Duran, M. A. and Grossmann, I. E. An outer-approximation algorithm for a class of mixed-integer nonlinear programs. *Mathematical Programming*, 36(3):307–339, 1986.

Farahani, R. Z., Miandoabchi, E., Szeto, W. Y., and Rashidi, H. A review of urban transportation network design problems. *European Journal of Operational Research*, 229(2): 281–302, 2013.

Fosgerau, M., McFadden, D., and Bierlaire, M. Choice probability generating functions. *Journal of Choice Modelling*, 8:1–18, 2013.

Frangioni, A. and Gendron, B. 0–1 reformulations of the multicommodity capacitated network design problem. *Discrete Applied Mathematics*, 157(6):1229–1241, 2009.

Freire, A. S., Moreno, E., and Yushimito, W. F. A branch-and-bound algorithm for the maximum capture problem with random utilities. *European Journal of Operational Research*, 252(1):204–212, 2016.

Gallego, G. and Wang, R. Multiproduct price optimization and competition under the nested logit model with product-differentiated price sensitivities. *Operations Research*, 62 (2):450–461, 2014.

Gendron, B. Decomposition methods for network design. *Procedia-Social and Behavioral Sciences*, 20:31–37, 2011.

Gendron, B. Revisiting lagrangian relaxation for network design. *Discrete Applied Mathematics*, 261:203–218, 2019.

Görtz, S. and Klose, A. The single-sink fixed-charge transportation problem: Applications and solution methods. In Günther, H.-O., Mattfeld, D. C., and Suhl, L., editors, *Management logistischer Netzwerke*, pages 383–406, Heidelberg, 2007. Physica-Verlag HD.

Görtz, S. and Klose, A. Analysis of some greedy algorithms for the single-sink fixed-charge transportation problem. *Journal of Heuristics*, 15(4):331–349, 2009.

Haase, K. Discrete location planning. Technical Report ITLS-WP-09-07, Institute of Transport and Logistics Studies, University of Sydney, 2009.

Haase, K., Müller, S., Krohn, R., and Hensher, D. The maximum capture problem with flexible substitution patterns. Working paper, Hamburg University, 2016.

Haase, K. and Müller, S. Management of school locations allowing for free school choice. *Omega*, 41(5):847–855, 2013.

Haase, K. and Müller, S. A comparison of linear reformulations for multinomial logit choice probabilities in facility location models. *European Journal of Operational Research*, 232 (3):689–691, 2014.

Haase, K., Knörr, L., Krohn, R., Müller, S., and Wagner, M. Facility location in the public sector. *Location Science*, pages 745–764, 2019.

Haberl, J. Exact algorithm for solving a special fixed-charge linear programming problem. *Journal of Optimization Theory and Applications*, 69:489–529, 1991.

Haberl, J., Nowak, C., Stettner, H., Stoiser, G., and Woschitz, H. A branch-and-bound algorithm for solving a fixed charge problem in the profit optimization of sawn timber production. *Zeitschrift für Operations Research*, 35:151–166, 1991.

Herer, Y. T., Rosenblatt, M. J., and Hefter, I. Fast algorithms for single-sink fixed charge transportation problems with applications to manufacturing and transportation. *Transportation Science*, 30:276–290, 1996.

Hirsch, W. M. and Dantzig, G. B. The fixed charge problem. *Naval Research Logistics Quarterly*, 15(3):413–424, 1968.

Holguin-Veras, J., Reilly, J., Aros-Vera, F., et al. New york city park and ride study. Technical report, University Transportation Research Center, 2012.

Kershenbaum, A. *Telecommunications network design algorithms*. McGraw-Hill, Inc., 1993.

Kim, S., Pasupathy, R., and Henderson, S. G. A guide to sample average approximation. *Handbook of Simulation Optimization*, pages 207–243, 2015.

Klose, A. Algorithms for solving the single-sink fixed-charge transportation problem. *Computers & Operations Research*, 35(6):2079–2092, 2008.

Lamontagne, S., Carvalho, M., Frejinger, E., Gendron, B., Anjos, M. F., and Atallah, R. Optimising electric vehicle charging station placement using advanced discrete choice models. *INFORMS Journal on Computing*, 2023.

Laporte, G., Nickel, S., and Saldanha-da Gama, F. *Introduction to location science*. Springer, 2019.

Legault, R. and Frejinger, E. A model-free approach for solving choice-based competitive facility location problems using simulation and submodularity. *arXiv preprint arXiv:2203.11329*, 2023.

Legault, R., Côté, J.-F., and Gendron, B. A novel reformulation for the single-sink fixed-charge transportation problem. *Mathematical Programming*, pages 1–30, 2023.

Li, H., Webster, S., Mason, N., and Kempf, K. Product-line pricing under discrete mixed multinomial logit demand: winner—2017 m&som practice-based research competition. *Manufacturing & Service Operations Management*, 21(1):14–28, 2019.

Liu, N., Ma, Y., and Topaloglu, H. Assortment optimization under the multinomial logit model with sequential offerings. *INFORMS Journal on Computing*, 32(3):835–853, 2020.

Ljubić, I. and Moreno, E. Outer approximation and submodular cuts for maximum capture facility location problems with random utilities. *European Journal of Operational Research*, 266(1):46–56, 2018.

Magnanti, T. L. and Wong, R. T. Network design and transportation planning: Models and algorithms. *Transportation Science*, 18(1):1–55, 1984.

Mai, T. and Lodi, A. A multicut outer-approximation approach for competitive facility location under random utilities. *European Journal of Operational Research*, 284:874–881, 2020.

Malek-Zavarei, M. and Frisch, I. T. On the fixed cost flow problem. *International Journal of Control*, 16(5):897–902, 1972.

Manski, C. F. The structure of random utility models. *Theory and Decision*, 8(3):229, 1977.

Martello, S. and Toth, P. An upper bound for the zero-one knapsack problem and a branch and bound algorithm. *European Journal of Operational Research*, 1(3):169–175, 1977.

Martello, S. and Toth, P. Algorithms for knapsack problems. *North-Holland Mathematics Studies*, 132:213–257, 1987.

Martello, S. and Toth, P. *Knapsack Problems: Algorithms and Computer Implementations*. John Wiley & Sons, Inc., USA, 1990.

Martello, S. and Toth, P. Upper bounds and algorithms for hard 0-1 knapsack problems. *Operations Research*, 45(5):768–778, 1997.

Martello, S., Pisinger, D., and Toth, P. Dynamic programming and strong bounds for the 0-1 knapsack problem. *Management Science*, 45(3):414–424, 1999.

Mateus, G. R., Luna, H. P., and Sirihal, A. B. Heuristics for distribution network design in telecommunication. *Heuristic Approaches for Telecommunications Network Management, Planning and Expansion: A Special Issue of the Journal of Heuristics*, pages 131–148, 2000.

McFadden, D. and Train, K. Mixed MNL models for discrete response. *Journal of Applied Econometrics*, 15(5):447–470, 2000a.

McFadden, D. and Train, K. Mixed mnl models for discrete response. *Journal of Applied Econometrics*, 15(5):447–470, 2000b.

Méndez-Vogel, G., Marianov, V., and Lüer-Villagra, A. The follower competitive facility location problem under the nested logit choice rule. *European Journal of Operational Research*, 2023.

Mingozzi, A. and Roberti, R. An exact algorithm for the fixed charge transportation problem based on matching source and sink patterns. *Transportation Science*, 52(2):229–238, 2018.

Minoux, M. Networks synthesis and optimum network design problems: Models, solution methods and applications. *Networks*, 19(3):313–360, 1989.

Miyamoto, K., Vichiensan, V., Shimomura, N., and Páez, A. Discrete choice model with structuralized spatial effects for location analysis. *Transportation Research Record*, 1898 (1):183–190, 2004.

Müller, S., Haase, K., and Seidel, F. Exposing unobserved spatial similarity: Evidence from german school choice data. *Geographical Analysis*, 44(1):65–86, 2012.

Nemhauser, G. L. and Wolsey, L. A. Maximizing submodular set functions: formulations and analysis of algorithms. In *North-Holland Mathematics Studies*, volume 59, pages 279–301. Elsevier, 1981.

Nemhauser, G. L., Wolsey, L. A., and Fisher, M. L. An analysis of approximations for maximizing submodular set functions—i. *Mathematical Programming*, 14:265–294, 1978.

Nourbakhsh, S. M., Bai, Y., Maia, G. D., Ouyang, Y., and Rodriguez, L. Grain supply chain network design and logistics planning for reducing post-harvest loss. *Biosystems Engineering*, 151:105–115, 2016.

Owen, S. H. and Daskin, M. S. Strategic facility location: A review. *European Journal of Operational Research*, 111(3):423–447, 1998.

Paneque, M. P., Bierlaire, M., Gendron, B., and Azadeh, S. S. Integrating advanced discrete choice models in mixed integer linear optimization. *Transportation Research Part B:*

Methodological, 146:26–49, 2021.

Paneque, M. P., Gendron, B., Azadeh, S. S., and Bierlaire, M. A lagrangian decomposition scheme for choice-based optimization. *Computers & Operations Research*, 148:105985, 2022.

Pinsker, M. S. *Information and information stability of random variables and processes*. Holden-Day, 1964.

Pisinger, D. A minimal algorithm for the 0-1 knapsack problem. *Operations Research*, 45 (5):758–767, 1997.

Pisinger, D. A minimal algorithm for the bounded knapsack problem. *INFORMS Journal on Computing*, 12(1):75–82, 2000.

Pisinger, D. Where are the hard knapsack problems? *Computers & Operations Research*, 32(9):2271–2284, 2005.

Plastria, F. Static competitive facility location: an overview of optimisation approaches. *European Journal of Operational Research*, 129(3):461–470, 2001.

Qi, M., Jiang, R., and Shen, S. Sequential competitive facility location: exact and approximate algorithms. *Operations Research*, 2022.

Resende, M. G. and Pardalos, P. M. *Handbook of optimization in telecommunications*. Springer Science & Business Media, 2008.

Roberti, R., Bartolini, E., and Mingozzi, A. The fixed charge transportation problem: An exact algorithm based on a new integer programming formulation. *Management Science*, 61(6):1275–1291, 2015.

Rosenblatt, M. J., Herer, Y. T., and Hefter, I. Note. an acquisition policy for a single item multi-supplier system. *Management Science*, 44(11-part-2):S96–S100, 1998.

Rusmevichientong, P., Shen, Z.-J. M., and Shmoys, D. B. Dynamic assortment optimization with a multinomial logit choice model and capacity constraint. *Operations Research*, 58 (6):1666–1680, 2010.

Salvador, S. and Chan, P. Determining the number of clusters/segments in hierarchical clustering/segmentation algorithms. In *16th IEEE international conference on tools with artificial intelligence*, pages 576–584. IEEE, 2004.

Satopaa, V., Albrecht, J., Irwin, D., and Raghavan, B. Finding a "kneedle" in a haystack: Detecting knee points in system behavior. In *2011 31st international conference on distributed computing systems workshops*, pages 166–171. IEEE, 2011.

Snyder, L. V. Facility location under uncertainty: a review. *IIE Transactions*, 38(7):547–564, 2006.

Vovsha, P. *The cross-nested logit model: application to mode choice in the Tel-Aviv metropolitan area*. Transportation Research Board, 1997.

Wieberneit, N. Service network design for freight transportation: a review. *OR Spectrum*, 30(1):77–112, 2008.

Williams, H. C. On the formation of travel demand models and economic evaluation measures of user benefit. *Environment and Planning A*, 9(3):285–344, 1977.

Zhang, Y., Berman, O., and Verter, V. The impact of client choice on preventive healthcare facility network design. *OR Spectrum*, 34(2):349–370, 2012.

Zhao, Y., Larsson, T., Rönnberg, E., and Pardalos, P. M. The fixed charge transportation problem: a strong formulation based on lagrangian decomposition and column generation. *Journal of Global Optimization*, 72:517–538, 2018.