CANADA EXCELLENCE RESEARCH CHAIR

DATA SCIENCE FOR REAL-TIME DECISION-MAKING

SOLVING LARGE-SCALE COMPETITIVE FACILITY LOCATION UNDER RANDOM UTILITY MAXIMIZATION MODELS

Tien Mai
Andrea Lodi

July 2017

DS4DM-2017-007

POLYTECHNIQUE MONTRÉAL
DÉPARTEMENT DE MATHÉMATIQUES ET GÉNIE INDUSTRIEL
Pavillon André-Aisenstadt
Succursale Centre-Ville C.P. 6079
Montréal - Québec
H3C 3A7 - Canada
Téléphone: 514-340-5121 # 3314

Solving large-scale competitive facility location under random utility maximization models

Tien Mai * Andrea Lodi *

July 31, 2017

Abstract

This work concerns the maximum capture problem with random utilities, i.e., the problem of seeking to locate new facilities in a competitive market such that the captured demand of users is maximized, assuming that each individual chooses among all available facilities according to a random utility maximization model. The challenge when solving this problem is the nonlinearity of the objective function. Existing approaches often only use the multinomial (MNL) logit model due to its simple structure, and address this challenge by formulating the problem into Mixed-Integer Linear Programing (MILP) models, so it can be solved by a MILP solver. There are two issues associated with this approach, namely, (i) the use of the MNL model retains the well-known independence of irrelevant alternatives property that may be undesirable in the context, and (ii) the MILP model could be difficult to solve for large instances.

In this paper, we address the aforementioned issues by dealing with the facility location problem under the mixed MNL (MMNL) model, which is general and fully flexible, compared to the MNL model. In addition, we do not solve equivalent MILP models, instead we propose an exact algorithm based on the outer approximation framework to quickly solve the associated 0-1 nonlinear problems. We test our algorithm using generated instances of different sizes, and show that our algorithm is much faster, compared to other existing approaches in the context. Especially, for several large-scale instances (thousands of locations and zones) under the MMNL model, our algorithm manages to find an optimal solution in few seconds, while other approaches cannot converge within a time budget of several hours.

Keywords: competitive facility location, maximum capture, multinomial logit, mixed multinomial logit, outer approximation.

1 Introduction

This paper concerns the facility location problem in competitive market, which has been receiving a growing attention in the last decade due to its appealing properties. In this problem, two

*CERC - Data Science for Real-time Decision-making, Polytechnique Montréal

aspects are taken into account, namely, the demand of customers and the competitors in the market. For the latter, the companies that would like to locate new facilities have to compete for their market share. Several competitive facility location models have been proposed. In general, theses models are based on the assumption that customers choose among different facilities based on a given utility that they assign for each location, and these utilities are typically functions of facility attributes/features, e.g., distances, prices, transportation costs. There are basically two main modeling approaches for the problem. The first approach, which we refer to as the *deterministic approach*, is based on the assumption that customers choose a facility in a deterministic way. For example, ReVelle (1986) proposes a model in which customers choose the closest facility among different competitors. This model therefore implies that all the demand of a zone is assigned entirely to a facility, which is not realistic. An alternative approach is the model proposed in Huff (1964), in which the demand captured by a facility is proportional to the attractiveness of the facility and inversely proportional to the distance. The reader can consult Berman et al. (2009) for a review.

The second modeling approach is referred to as the *probabilistic approach*, in which the demand of customers is captured by a probabilistic model, i.e., a model that can assign a probability distribution over the facilities. The random utility maximization framework (Ben-Akiva and Lerman, 1985, McFadden, 1978) is convenient to use in the context. Under this framework, a random utility is associated with each facility, and a customer is assumed to choose a facility by maximizing his/her utilities. More precisely, we assume that there is an utility u_j associated with a facility j, and it includes the attributes/features of the facility, i.e., $u_j = \beta^{\mathrm{T}} a_j + \epsilon_j$, where a_j is the vector of attributes of facility j and β is the vector of the model parameters, which can be obtained by estimating/training the choice model, and ϵ_j is the random component that cannot be observed by the analyst. Under the "utility maximization" assumption, this way of modeling allows us to compute the probability that a customer chooses a facility i versus other facilities as $P(u_i \geq u_j,\ \forall j)$. In this context, the facility location problem can be described as follows: How to locate facilities in a competitive market such that the expected market share captured by the new facilities is maximized (so the problem is also called as the "maximum capture" problem). This modeling approach was first introduced by Benati and Hansen (2002) and had several applications afterward (Aros-Vera et al., 2013, Haase and Müller, 2013, 2015). The advantage of this approach, compared to the deterministic one, is that the probabilistic models can be trained/estimated using real data, so the demand of customers can be predicted more accurately. The challenge, however, lies in the fact that the corresponding data-driven discrete optimization problems are nonlinear, thus they are typically difficult to solve. Existing approaches address this challenge by linearizing the problems into Mixed-Integer Linear Programing (MILP) models, so they become more convenient to solve (Benati and Hansen, 2002, Hasse, 2009, Zhang et al., 2012). Recently, Haase and Müller (2014) have made a comparison between three MILP formulations in the literature, and concluded that the one proposed in Hasse (2009) is the most efficient. Even more, Freire et al. (2016) have proposed a Brand-and-Bound algorithm to strengthen the MILP approach of Hasse (2009).

There are two different aspects that need to be taken into account when using the random utility maximization framework, namely, the prediction performance of the choice model and the complexity of the resulting optimization problems. For the first aspect, existing studies that we mentioned only focus on the multinomial logit (MNL) model due to its simple structure. The MNL model is based on the assumption that the random components ϵ_j, $\forall j$, in the utilities are independent and identically distributed (i.i.d.) and follow the standard Gumbel distribution, so the model exhibits the so-called independence of irrelevant alternatives (IIA) property, which could be an issue in many contexts (see Ben-Akiva and Lerman, 1985, for instance). In Section 2.1 we present a counter example showing why the use of the MNL model can lead to inaccurate prediction results in the context of facility location. There are other advanced choice models that can be used to relax the IIA property and provide better prediction, namely, the mixed MNL (MMNL) (McFadden and Train, 2000) and multivariate extreme value (MEV) (McFadden, 1978). In particular, the MMNL model is fully flexible, as it can approximate any random utility model (McFadden and Train, 2000). The use of these models, however, results in complicated optimization problems, which could be intractable to solve.

Regarding the second aspect, there is a trade-off between using advanced choice models and the complexity of the resulting optimization problem. To be more precise, the use of models that relaxes the IIA could lead to intractable optimization models. An mentioned, existing studies are only based on the MNL model. Moreover, even with the simple MNL, the MILP formulations only enable commercial solvers (e.g., CPLEX) to solve quite small instances (less than 100 locations) (see Haase and Müller, 2014, for instance).

In this paper, we advance the state of the art for the aforementioned aspects. More precisely, we propose a new algorithm solution that allows to quickly solve large-scale instances under the MMNL model.

Our contributions:

(i) We formulate and solve the maximum capture problem under the MNL and MMNL models. To the best of our knowledge, this is the first study in the context that deals with the MMNL model. We propose an algorithm based on the outer approximation scheme (Duran and Grossmann, 1986) that allows to solve large-scale instances much more quickly, compared to the existing approaches in the literature. The idea is that we formulate the equivalent minimization of the maximum capture problem, and we iteratively generate linear cuts in order to create an outer-approximation of the objective function, which is piecewise-linear and convex, and we minimize this piecewise linear function instead of the nonlinear objective to obtain optimal solutions.

(ii) We test our algorithm on several instances of different sizes, and show that even for instances of thousands of locations, our approach manages to find an optimal solution in fews seconds, while other existing approaches cannot converge after several hours, even with small instances of less than 100 locations.

(iii) The use of the MMNL model allows to flexibly and accurately predict the customers' demand. Our algorithm opens a very efficient way to deal with the problem under the MMNL model, and we believe that it is a significant step forward in the facility location literature.

(iv) Our approach also opens the possibility to deal with more complicated variants of the problem considered in this paper, e.g., the maximal covering salesman problem (Church and Velle, 1974), where the demands of users are captured by a random utility maximization model.

The paper is structured as follows. Section 2 presents the random utility maximization framework and the maximum capture problem under random utility choice models. In Section 3 we present how to formulate the problem into MILP formulations. Our algorithm is presented in Section 4. Section 5 reports the computational results comparing the performance of our approach with the MILP formulations and some existing mixed integer nonlinear programming solvers. Finally, Section 6 concludes.

2 Facility location under the random utility maximization

In this section we first present the discrete choice framework used for analyzing and predicting the behavior of decision makers. We then describe the maximum capture problem in competitive market under these models.

2.1 Random utility maximization models

The random utility maximization (RUM) framework (McFadden, 1978) is the most widely used approach to model discrete choice behavior. Under this framework, each customer (decision maker) n associates an utility u_{ni} with each alternative/location i in a choice set S_n. This utility consists of two parts: a deterministic part v_{ni} that contains observed attributes/features, and a random term ϵ_{ni} that is unknown to the analyst. In general, a linear-in-parameters formula is used, i.e, $v_{ni} = \beta^{\mathrm{T}} a_{ni}$, where $^{\mathrm{T}}$ is the transpose operator, β is a vector of parameters to be estimated and a_{ni} is the vector of attributes of alternative i as observed by customer n.

The framework assumes that the customer aims at maximizing the utility, and the choice probability that an alternative/option i in choice set S_n is chosen by individual n is given as

$$P(i|S_n) = P(u_{ni} \geq u_{nj},\ \forall j \in S_n) = P(v_{ni} + \epsilon_{ni} \geq u_{nj} + \epsilon_{nj}, \forall j \in S_n). \tag{1}$$

Different assumptions for the random terms lead to different types of discrete choice models. The MNL model is widely used in this context. This model results from the assumption that the random terms $\epsilon_{ni}, i \in S_n$, are i.i.d. and follow the standard Gumbel distribution. The corresponding choice probability has the simple form

$$P(i|S_n) = \frac{e^{v_{ni}}}{\sum_{j\in S_n} e^{v_{nj}}}.$$

If the model is linear-in-parameters, the choice probabilities can be written as a function of parameters β as

$$P(i|S_n) = \frac{e^{\beta^{\mathrm{T}} a_{ni}}}{\sum_{j\in S_n} e^{\beta^{\mathrm{T}} a_{nj}}}.$$

It is well known that the MNL model exhibits the IIA property, which implies proportional substitution across alternatives. This property implies that for two alternatives, the ratio of the choice probabilities is the same no matter what other alternatives are available or what the attributes of the other alternatives are. However, if alternatives share unobserved attributes (i.e., random terms are correlated), then the IIA property does not hold. In the following we give a counter example showing that the use of the MNL and the IIA property would be undesirable, and may lead to inaccurate predictions in the context of facility location.

We consider an example of one zone and three locations. We assume that customers are located at *Zone i* (i.e., a geographical area), and there are three facilities located at Location 1, 2 and 3. Moreover, the distances between Zone i and the three locations are equal, but Location 2 and Location 3 are close to each other (see Figure 1). Now we want to have a model that can predict the probabilities that customers located at Zone i select locations, under the assumption that the decisions of customers are only driven by the distances, i.e., the utility associated with a location j depends only on the distance between j and Zone i, i.e., $v_{ij} = \alpha d(i,j)$, where $d(i,j)$ is the distance (or travel time) from Zone i to location j, and α is a negative constant.

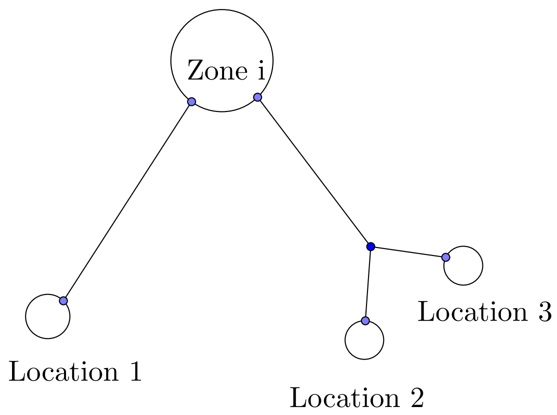

Figure 1: Example of one zone and three locations

If the MNL model is used, because $d(i,1) = d(i,2) = d(i,3) = d^*$, then the probabilities that a customer located at Zone i selects the three locations are equal, i.e.,

$$P_{i1} = P_{i2} = P_{i3} = \frac{\exp(\alpha d^*)}{3 \times \exp(\alpha d^*)} = \frac{1}{3}.$$

However, we expect that the probability that the customer goes to the left side and select Location 1 is 1/2, and the probability that he/she goes to the right side and select Location 2 or 3 is 1/2, i.e., $P_{i1} = 1/2$, and $P_{i2} = P_{i3} = 1/4$. Another illustration for the drawback of the IIA property is as follows. Assuming that a MNL model can assign a probability of 1/2 for Location 1, and probabilities of 1/4 for Locations 2 and 3. The ratio between probabilities P_{i1} and P_{i2} is $P_{i1}/P_{i2} = 2$. The IIA property of the MNL model leads to the fact that this ratio does not change even when the facility at Location 3 is closed. However, we expect that when there are only two facilities at Locations 1 and 2 in the market, P_{i1} and P_{i2} should be the same. This inaccurate prediction is due to the fact that the MNL cannot account for the correlation between the utilities of Locations 2 and 3. In this context, other more advanced choice models like the mixed logit or nested logit models (Train, 2003) could be used to correct the random utilities for the correlation and provide better prediction. This counter example is similar to the well-known "*three paths*" example in transportation demand modeling (Ben-Akiva and Bierlaire, 1999), which is also used to illustrate why it is important to relax the IIA property of the MNL model.

As mentioned, several choice models have been proposed to relax the IIA property from the MNL, and the mixed MNL (MMNL) is one of the most preferable, as it is fully flexible in the sense that it can approximate any random utility model (McFadden and Train, 2000). The model has been widely used in practice due to this convenience.

In the MMNL model, the parameters β are assumed to be random, and the choice probabilities are the integrals of standard MNL probabilities over a density $f(\beta)$ of parameters β

$$P(i|S_n) = \int \frac{e^{\beta^{\mathrm{T}} a_{ni}}}{\sum_{j \in S_n} e^{\beta^{\mathrm{T}} a_{nj}}} f(\beta) d\beta.$$

In most of the cases, $P(i|S_n)$ has no closed form, so in order to estimate the model, the probabilities need to be approximated by a Monte Carlo method. To be more precise, if we assume $\beta_1, \ldots, \beta_T$ are T realizations sampled over the distribution of β, then the choice probabilities can be approximated as

$$P(i|S_n) \approx \bar{P}_T(i|S_n) = \frac{1}{T} \sum_{t=1}^{T} \frac{e^{\beta_t^{\mathrm{T}} a_{ni}}}{\sum_{j \in S_n} e^{\beta_t^{\mathrm{T}} a_{nj}}}.$$

There are two ways of modeling the MMNL model. The first way is based on random parameters where the parameters β of the utilities v_{ni} are random and vary over the decision makers in the population with density $f(\beta)$, and the second way adds error components with zero means to the utilities in order to create correlation between alternatives. Although these approaches are formally equivalent, they are associated different interpretations (see Train, 2003, for more details).

In general, the parameters of a MNL or MMNL model can be obtained by maximum likelihood estimation, i.e., we maximize the log-likelihood function defined over observations $i_1, \ldots, i_N$ (the observations that customers choose facilities among a set of availability ones)

$$\max \frac{1}{N} \sum_{n=1}^{N} \log P(i_n|S_n), \tag{2}$$

where (2) can be solved using an unconstrained nonlinear optimization algorithm (Nocedal and Wright, 2006). Note that the estimation of the MMNL model is more complicated, compared to the MNL, as it requires an integration over the distribution of the random parameters. The integration can be approximated numerically by sampling over the random parameters, and the sample can be generated by standard Monte Carlo or quasi-Monte Carlo techniques (Munger et al., 2012).

There are also several ways to relax the IIA property from the MNL model by making different assumptions on the random terms of the utilities, i.e., ϵ_{ni}. For example, the nested logit (Ben-Akiva, 1973), the cross-nested logit Vovsha and Bekhor (1998), or generalized network MEV model (Daly and Bierlaire, 2006). These models all belong to the multivariate extreme value (MEV) family of models. Such models are convenient to use in some contexts, due to the fact that they allow to represent the correlation between alternatives by a network (Daly and Bierlaire, 2006). Even MEV models can be trained/estimated quickly using dynamic programming techniques (Mai et al., 2017), but the use of these models in the context of competitive facility location leads to complicated optimization models in which the objective functions are highly nonlinear and non-convex, so difficult to solve.

It is interesting to mention that there is a number of non-parametric choice models having been proposed as alternatives to the classical parametric models presented above. In particular, Farias et al. (2013) have proposed a general model of choice where one represents choice behavior by a probability distribution over all of the possible rankings of the alternatives. This model has received a growing attention in the literature of revenue management (Bertsimas and Mišic, 2016, Farias et al., 2013). This model has an advantage of being able to represent a wide variety of choice models, but the disadvantage is that the estimation and application are still difficult, as it is based on all possible rankings of alternatives, and the number of rankings are typically very large (the factorial of the number of alternatives).

2.2 Facility location in competitive market

In the market we assume that there are $V = \{1, \ldots, m\}$ available locations, and we denote by $Y \subset V$ the set of locations that have facilities of the competitor company. Let I the set of zones where customers are located, and q_i be the number of customers located in zone $i \in I$.

The objective is to maximize the market share (i.e. number of customers) by locating facilities in a subset of locations $X \subset V$. We denote by $R(X)$ the expected market share given by X facilities. So, $R(X)$ can be computed as

$$R(X) = \sum_{i \in I} q_i \sum_{j \in X} P(i, j|X, Y),$$

where $P(i, j|X, Y)$ is the probability that a customer located in i selects facility $j \in X$. If the MNL model is used to predict the choice probabilities of customers, then $R(X)$ can be computed as

$$R(X) = \sum_{i \in I} q_i \frac{\sum_{j \in X} e^{v_{ij}}}{\sum_{j \in X} e^{v_{ij}} + \sum_{j \in Y} e^{v_{ij}}},$$

where $v_{ij} = (\beta^*)^{\mathrm{T}} a_{ij}$ is the utility associated with location j and a customer located in zone i, and β^* is the vector of the model parameters of the choice model, and a_{ij} is the vector of features/attributes associated with location j and customers at zone i.

For the MMNL model, it is assumed that the parameters β are random numbers. The choice probabilities as well as the expected market share can be obtained by taking the expectation over distribution of β, i.e.,

$$R(X) = \sum_{i \in I} q_i \int \left(\frac{\sum_{j \in X} e^{v_{ij}}}{\sum_{j \in X} e^{v_{ij}} + \sum_{j \in Y} e^{v_{ij}}} \right) f(\beta) d\beta,$$

where $f(\cdot)$ is the density function of β. As mentioned, this integration can be approximated by a Monte Carlo method

$$\bar{R}_T(X) = \sum_{i \in I} \frac{q_i}{T} \sum_{t=1}^{T} \left(\frac{\sum_{j \in X} e^{v_{ijt}}}{\sum_{j \in X} e^{v_{ijt}} + \sum_{j \in Y} e^{v_{ijt}}} \right),$$

where $v_{ij1}, \ldots, v_{ijT}$ are T realizations of the random utility v_{ij}, $i \in I, j \in V$, taking over the randomness of parameters β. The maximum capture problem under the MNL can be written as

$$\max_{X \subset V} \sum_{i \in I} q_i \frac{\sum_{j \in X} e^{v_{ij}}}{\sum_{j \in X} e^{v_{ij}} + U_Y^i}, \tag{3}$$

where $U_Y^i = \sum_{j \in Y} e^{v_{ij}}$, which is a constant in the optimization problem. We also can formulate (3) into a mixed integer nonlinear programming model as

$$\max_{\substack{x_j \in \{0,1\} \\ j \in \{1,2,\ldots,m\}}} \quad \sum_{i \in I}^{n} q_i \left(\frac{\sum_{j=1}^{m} x_j e^{v_{ij}}}{\sum_{j=1}^{m} x_j e^{v_{ij}} + U_Y^i} \right), \tag{FL-MNL}$$

where x_j, $j \in V$, is equal to 1 if location j is selected, and 0 otherwise. If the MMNL is used to predict the choice probabilities, the maximum capture problem is written in a similar way as

$$\max_{X \subset V} \sum_{i \in I} \frac{q_i}{T} \sum_{t=1}^{T} \left(\frac{\sum_{j \in X} e^{v_{ijt}}}{\sum_{j \in X} e^{v_{ijt}} + U_Y^i} \right),$$

and this is equivalent to the mixed-integer nonlinear programming problem

$$\max_{\substack{x_j \in \{0,1\} \\ j \in \{1,2,\ldots,m\}}} \quad \sum_{i \in I}^{n} \frac{q_i}{T} \sum_{t=1}^{T} \left(\frac{\sum_{j=1}^{m} x_j e^{v_{ijt}}}{\sum_{j=1}^{m} x_j e^{v_{ijt}} + U_Y^{it}} \right). \tag{FL-MMNL}$$

Due to the nonlinearity of the objective function, (FL-MNL) and (FL-MMNL) are typically difficult to solve. Note that we write (FL-MNL) and (FL-MMNL) in their simplest forms, and different business constraints can be included, e.g., a constraint on the number of facilities that the firm would like to locate, or constraints on the budget the firm has to open facilities.

Basically, (FL-MNL) and (FL-MMNL) are 0-1 fractional linear programming models, for which it is possible to reformulate the nonlinear models into mixed-integer linear programming ones (Wu, 1997). In the context of competitive facility location, this has been done in some previous studies, e.g., Benati and Hansen (2002), Hasse (2009) and Zhang et al. (2012). We present this approach in detail in the following section.

3 Mixed integer linear programing formulations

In this section we present how (FL-MNL) and (FL-MMNL) can be formulated as MILPs, so these problems can be solved using a general-purpose MILP solvers (e.g., CPLEX). First, for the ease of notation, we define $V_{ij} = e^{v_{ij}}$ and rewrite (FL-MNL) as

$$\begin{aligned} \underset{x}{\text{maximize}} \quad & \sum_{i \in I} q_i \left(\frac{\sum_{j=1}^{m} x_j V_{ij}}{\sum_{j=1}^{m} x_j V_{ij} + U_Y^i} \right) \\ \text{subject to} \quad & Ax \leq b \\ & x \in \{0,1\}^m, \end{aligned} \tag{4}$$

where $Ax \leq b$ are some linear business constraints. There are different ways of formulating the above problem as a MILP (Benati and Hansen, 2002, Hasse, 2009, Zhang et al., 2012). Haase and Müller (2014) show that among different MILP formulations in the literature, the one proposed by Hasse (2009) performs the best. In the following, we present Hasse (2009)'s formulation and use it as a benchmark for comparison with our approach.

Following Hasse (2009), if we define variables and constants as

$$z_{ij} = \frac{V_{ij} x_j}{\sum_{j=1}^{m} x_j V_{ij} + U_Y^i}, \; y_i = \frac{U_Y^i}{\sum_{j=1}^{m} x_j V_{ij} + U_Y^i}, \; \phi_{ij} = \frac{V_{ij}}{U_Y^i},$$

then (4) can be written equivalently as

$$\begin{aligned}
\underset{x}{\text{maximize}} \quad & \sum_{i\in I} q_i \sum_{j=1}^{m} z_{ij} \\
\text{subject to} \quad & Ax \leq b \\
& y_i + \sum_{j=1}^{m} z_{ij} \leq 1, \ \forall i \in I \\
& z_{ij} - \frac{\phi_{ij}}{\phi_{ij}+1} x_j \leq 0, \ \forall i \in I, j \in V \\
& z_{ij} - \phi_{ij} y_j \leq 0, \ \forall i \in I, j \in V \\
& z_{ij}, y_j \geq 0, \ \forall i \in I, j \in V \\
& x_i \in \{0,1\}, \ \forall i \in I.
\end{aligned} \tag{MILP-MNL}$$

Under the MMNL model, if we define (FL-MMNL) $V^t_{ij} = e^{v_{ijt}}$, then (FL-MMNL) can be written as

$$\begin{aligned}
\underset{x}{\text{maximize}} \quad & \frac{1}{T}\sum_{t=1}^{T}\sum_{i\in I}\left(\frac{q_i \sum_{j=1}^{m} x_j V^t_{ij}}{\sum_{j=1}^{m} x_j V^t_{ij} + U^{it}_Y}\right) \\
\text{subject to} \quad & Ax \leq b \\
& x \in \{0,1\}^m.
\end{aligned} \tag{5}$$

Even if Hasse (2009) only formulate the problem under the MNL, it is straightforward to formulate the maximum capture problem under the MMNL model as a MILP. More precisely, we can define

$$z^t_{ij} = \frac{V^t_{ij} x_j}{\sum_{j=1}^{m} x_j V^t_{ij} + U^{it}_Y}, \ y^t_i = \frac{U^{it}_X}{\sum_{j=1}^{m} x_j V^t_{ij} + U^{it}_Y}, \ \phi^t_{ij} = \frac{V^t_{ij}}{U^{it}_Y},$$

then (4) can be written equivalently as

$$\begin{aligned}
\underset{x}{\text{maximize}} \quad & \frac{1}{T}\sum_{t=1}^{T}\sum_{i\in I} q_i \sum_{j=1}^{m} z^t_{ij} \\
\text{subject to} \quad & Ax \leq b \\
& y^t_i + \sum_{j=1}^{m} z^t_{ij} \leq 1, \ \forall i \in I, \ t = 1,\ldots,T \\
& z^t_{ij} - \frac{\phi^t_{ij}}{\phi^t_{ij}+1} x_j \leq 0, \ \forall i \in I, j \in V, \ t = 1,\ldots,T \\
& z^t_{ij} - \phi^t_{ij} y^t_j \leq 0, \ \forall i \in I, j \in V, \ t = 1,\ldots,T \\
& z^t_{ij}, y^t_j \geq 0, \ \forall i \in I, j \in V, \ t = 1,\ldots,T \\
& x \in \{0,1\}^m, \ \forall i \in I.
\end{aligned} \tag{MILP-MMNL}$$

For the problem under the MNL model, the MILP formulation has $|V|$ binary variables, and $(|I| + |V| + |I| \times |V|)$ continuous variables, and the number of constraints is $|I| + 2 \times |I| \times |V|$.

Under the MMNL model, these numbers are multiplied by the sample size T. This means that these MILP formulations would become heavy and difficult to solve with large instances, especially for the case of the MMNL model. The approach in the following section open a possibility to solve large problems very quickly.

4 Outer approximation algorithm

We first write (4) as follows

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & Q(x) = -\sum_{i\in I} q_i \left(\frac{\sum_{j=1}^m x_j V_{ij}}{\sum_{j=1}^m x_j V_{ij} + U_Y^i} \right) \\ \text{subject to} \quad & Ax \leq b \\ & x \in \{0,1\}^m, \end{aligned} \tag{P1}$$

and in the case of the MMNL model, (FL-MMNL) can be written as

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & Q_T(x) = -\sum_{i\in I} \frac{q_i}{T} \sum_{t=1}^{T} \left(\frac{\sum_{j=1}^m x_j V_{ijt}}{\sum_{j=1}^m x_j V_{ijt} + U_Y^{it}} \right) \\ \text{subject to} \quad & Ax \leq b \\ & x \in \{0,1\}^m. \end{aligned} \tag{P2}$$

We introduce the following proposition that supports the use of the outer approximation scheme.

Proposition 1 *The continuous relaxation of $Q(x)$ and $Q_T(x)$ is convex.*

The convexity of the objective function in the case of the MNL was also established in Benati and Hansen (2002), and their proof can be adapted for the case of the MMNL model. For the sake of self-containment we provide a proof in Appendix A for the both MNL and MMNL models.

The idea of the outer approximation method (Duran and Grossmann, 1986) is to create a piecewise linear and convex function $G(x)$ that underestimate $Q(x)$, i.e., $G(x) \leq Q(x)$, $\forall x$. If this function is tight at every integer point in the feasible set of the problem, i.e., $G(x) = Q(x)$, $\forall x \in \{0,1\}^n$, $Ax \leq b$, then we can find an optimal solution to (FL-MNL) by solving $\min\{G(x) |\ x \in \{0,1\}^n, Ax \leq b\}$.

In order to create such function, we first consider the following mixed integer equivalence of

(P1).

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & \theta & \text{(P3)} \\ \text{subject to} \quad & Ax \leq b \\ & \theta \geq Q(x) & (6) \\ & x_i \in \{0,1\}. \end{aligned}$$

The idea of the algorithm is to relax (6) and consider θ as an underestimator of $Q(x)$, and successively add cuts in the (x,θ)-space to better approximate the shape of $Q(x)$. This is done until an optimal solution (x^*,θ^*) satisfying $\theta^* = Q(x^*)$ is found. Now, we describe the algorithm. We first define the master problem by projecting the feasible region of (P3) onto the x-space

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & \theta & \text{(MP)} \\ \text{subject to} \quad & Ax \leq b \\ & \Pi x - \mathbf{1}\theta \leq \pi_0 & (7) \\ & \theta \geq L \\ & x_i \in \{0,1\} \end{aligned}$$

where $\mathbf{1}$ denotes a vector of ones of appropriate size, and L is a lower bound of $Q(x)$ in the feasible set. Since $Q(x)$ is convex in $[0,1]^n$, L can be computed by solving the following nonlinear continuous optimization problem, which is the continuous relaxation of (P1)

$$L = \min_{\substack{x\in[0,1]^n \\ Ax\leq b}} Q(x). \tag{8}$$

In some cases solving (8) could be costly, so we can simply choose L according to the following proposition

Proposition 2 *The following inequalities are valid* $\forall x \in \{0,1\}^m$,

$$Q(x) \geq -\sum_{i\in I} q_i \left(\frac{\sum_{j=1}^m V_{ij}}{\sum_{j=1}^m V_{ij} + U_Y^i} \right),$$

$$Q_T(x) \geq -\sum_{i\in I} \frac{q_i}{T} \sum_{t=1}^{T} \left(\frac{\sum_{j=1}^m V_{ijt}}{\sum_{j=1}^m V_{ijt} + U_Y^{it}} \right).$$

Proposition 2 also indicates that in the context of the MNL or MMNL model, if there is no restriction on the number of facilities, then the optimal solution is $x_j = 1,\ j = 1,\dots,m$, i.e., all the locations should be selected to locate facilities.

It can be shown that (P3) is equivalent to (MP) if for each $x^* \in M = \{\{0,1\}^m | Ax \leq b\}$ the constraint set (7) contains cuts of the form $\pi^k x - \theta \leq \pi_0^k$ such that $\pi^k x - \pi_0^k \leq Q(x)$ for all $x \in M$, and $\pi^k x^* - \pi_0^k \leq Q(x^*)$. The idea of the approach is that we initialize the master problem (MP) with empty constraint (7). Given that $Q(x)$ is convex, thus for any $x^* \in X$ the following equality is valid

$$Q(x) \geq \nabla Q(x^*)(x - x^*) + Q(x^*),$$

where $\nabla Q(x^*)$ is the gradient of $Q(x)$ at x^*. Because we want $\theta \geq Q(x)$, we can add a sub-gradient cut of the form

$$\theta \geq \nabla Q(x^*)(x - x^*) + Q(x^*). \tag{9}$$

Note that $\nabla Q(x^*)$ can be easily computed analytically using (P1). Before presenting the algorithm we highlight some properties of the approach that were already verified by Duran and Grossmann (1986). Namely,

(i) Let (x^*, θ^*) be a solution to (MP), if $\theta^* \geq Q(x^*)$, then x^* is an optimal solution and θ^* is the optimal value of the problem.

(ii) If the master problem contains cuts based on a set of binary solutions Z, then if at an iteration we find a solution (x^*, θ^*) such that $x^* \in Z$, then x^* and θ^* are an optimal solution and the optimal value, respectively.

Property (ii) can be easily verified by the fact that if we find a solution (x^*, θ^*) such that $x^* \in Z$, then (x^*, θ^*) is a solution to (MP) containing constraint $\theta \geq \nabla Q(x^*)(x - x^*) + Q(x^*)$, meaning that $\theta^* \geq Q(x^*)$. Hence, according to Property (i), x^* is an optimal solution. Property (ii) also suggests a way to avoid recomputing the objective function ($Q(x)$ or $Q_T(x)$), which could be costly with large instances. More precisely, each time a solution x^* is found, we can add it to Z and also save the objective value $Q(x^*)$ or $Q_T(x^*)$. At each iteration, after solving the master problem to obtain (x^*, θ^*), we can first check if x^* is in Z, then we can return x^* as an optimal solution. Otherwise we compute the gradient of $Q(x^*)$ to create and add a sub-gradient cut to the master problem.

This description of the outer approximation method is summarized in Algorithm 1. The difference between Algorithm 1 and the outer-approximation algorithm presented in Bonami et al. (2008) is that: (i) we do not solve the continuous relaxation of the problem to get the first cut, instead, we compute the lower bound using Proposition 2, and (ii) we save the set of binary solutions found at each iteration to avoid recomputing the objective function and its gradient. Actually, in our numerical studies, for the maximum capture problem the OA algorithm only needs a few iterations to converge, so these modifications would help to remarkably reduce the computational cost, in particular for large instances.

We have the following remarks for Algorithm 1. First, the theoretical results from Duran and Grossmann (1986) guarantee that the outer approximation algorithm finishes after a finite number of iterations, and returns an optimal solution. Second, the algorithm is described based on function $Q(x)$ but it can apply for $Q_T(x)$ in a similar way. Third, in order to compute

Algorithm 1: Outer approximation algorithm

begin

 # Initialization

Step 1. Chose a lower bound L according to Proposition 2, a convergence tolerance $\epsilon > 0$, and $Z = \emptyset$.

Step 2. Initialize the master problem (MP) with empty Π.

Step 3. Compute (x^*, θ^*) as the first solution by solving (MP).

 # Iteratively adding cuts until getting an optimal solution

Step 4. If $x^* \in Z$ then go to Step 6, otherwise w set $Z = Z \cup \{x^*\}$ and compute $Q(x^*)$

Step 5. If $\theta^* \geq Q(x^*) - \epsilon$, then go to Step 6, otherwise

- Compute $\nabla Q(x^*)$ and add subgradient cut to the master problem (MP)

$$\theta \geq \nabla Q(x^*)(x - x^*) + Q(x^*)$$

- Solve (MP) to obtain new solution (x^*, θ^*), and move back to Step 4

 # Finalization

Step 6. Return x^* as an optimal solution and θ^* as the optimal value.

sub-gradient cuts, we need to have the first derivative of $Q(x)$, which can be easily derived analytically. Fourth, the use of the algorithm requires the convexity of the objective function (expected maximum capture), which does not hold under the MEV model. So, in the context of the MEV model, the algorithm is no-longer exact but heuristic. Finally, the size of the master problem (in terms of number of variables and constraints) in (MP) is remarkably smaller than the MILP models in (MILP-MNL) and (MILP-MMNL), and the size of this master problem is independent of the choice model (the MNL or MMNL model) that we use, so with this nice property, we expect that (MP) can be quickly solved for large-scale instances under the MMNL model.

5 Computational studies

In this section we test our algorithm using generated datasets of different sizes. We first show how the cutting algorithm performs with the MNL model, in comparison with the MILP approach. We then extend the experiments with the MMNL model.

5.1 Experimental setting

In this section we present computational results comparing our integer cutting plane to the MILP approach in Hasse (2009), which has been showed to have the best performance among the existing MILP formulations in the literature (Haase and Müller, 2014). We generate instances of different sizes for the experiments. More precisely, we generate sets of locations by randomly and uniformly generating their coordinates in a two dimensional rectangular. For each set of locations, we generate 4 sets of zones, and the coordinates of these zones are also generated uniformly in the same rectangular containing the locations. For each set of location, we randomly take a subset to locate facilities of the competitor. This way of generating instances is similar to those in previous studies (Benati and Hansen, 2002, Freire et al., 2016). The constraints that we consider in this experiment have the form

$$L \leq \sum_{j \in V} x_j \leq U,$$

where L and U are integer constants such that $1 \leq L \leq U \leq |V|$. These lower and upper bounds are also generated randomly such that L is greater than 25% of the total number of locations, i.e., $|V|$. We also set the time limit for the MILP solver to 1,200 CPU seconds. If the solver stops because it exceeds the time limit, we report the percentage gaps (%) between the objective values given by the solver and the corresponding optimal values. The experiments were conducted on a PC with processor Intel(R)Core(TM) CPUs of 2.8 Ghz, RAM of 8 GB and operating system Window 10. The algorithms were coded in MATLAB and linked to IBM-ILOG CPLEX 12.6 optimization routines under default settings. We use the MILP solver of CPLEX to solve (MILP-MNL) and (MILP-MMNL), and the master problem of the cutting plane algorithm, i.e., (MP). In these numerical studies we assume that the estimation procedure is done, so all the model parameters are known for the optimization problem. For the sake of simplicity, we use the rectangular distance as the only attribute of the facilities, and we choose the model parameters for the utilities $\beta = -1$.

5.2 Case study 1: MNL model

In the first case study we test our algorithm on the problem under the MNL model. Table 1 reports the computational results for small- and medium-size instances (the number of locations varies from 50 to 500), where OA stands for Algorithm 1, and MILP denotes the approach that in which MILP formulation in (MILP-MNL) is solved directly by CPLEX. The symbol "-" in the table indicates the cases when the MILP fails to return an optimal solution within 1,200 seconds. In this case, we report the gaps (%) between the best objective values given by CPLEX and the corresponding optimal values computed by the OA. The results clearly show the superiority of our OA approach. The OA algorithm manages to find an optimal solution within 0.1 seconds for all the instances, while the MILP exceeds the time limit for 22/24 instances. When the number of locations becomes large, the gaps are also increased and significant. We note that when the number of locations is larger then 200, the MILP approach fails to return optimal solutions within several hours. These results are also consistent with those reported in previous studies (Benati and Hansen, 2002, Haase and Müller, 2014), which point out that when the number of locations is larger than 100, the MILP approach cannot find an optimal solution within 30 minutes.

It is important to mention that Freire et al. (2016) has strengthened the MILP approach from

$\lvert V \rvert$	$\lvert I \rvert$	OA	MILP	
		Time(s)	Time(s)	Gap (%)
50	20	0.05	23.8	0.0
	30	0.10	121.4	0.0
	40	0.07	-	0.0
	60	0.06	-	0.0
100	60	0.10	-	0.0
	80	0.05	-	0.0
	120	0.06	-	0.1
	140	0.06	-	0.0
200	100	0.06	-	0.1
	200	0.08	-	0.4
	300	0.08	-	1.8
	400	0.08	-	11.3
300	200	0.11	-	4.8
	300	0.08	-	5.9
	400	0.09	-	15.9
	500	0.12	-	7.7
400	300	0.09	-	17.4
	400	0.09	-	6.9
	500	0.10	-	5.2
	600	0.14	-	17.2
500	400	0.13	-	9.8
	500	0.13	-	16.9
	600	0.12	-	21.8
	700	0.14	-	4.4

Table 1: Comparison results with small and medium instances, the "-" indicates that the MILP solver fails to give an optimal solution within the time budget (1200 seconds)

Hasse (2009), but their computational results show that their Brand-and-Bound (B&B) algorithm does not make a "huge" improvement, e.g., for the largest instance that they considered (100 locations), the B&B needs about 300 seconds to find an optimal solution, and there are several instances that cannot be solved to optimality. Even if their computational results may be based on a different experimental setting, it is enough to say that the B&B is much slower and less efficient, compared to the OA, as our OA algorithm can easily find optimal solutions for larger instances (up to 500 locations) in less than 0.2 seconds.

Given that with the small and large instances reported in Table 1, the OA needs only less than 0.2 seconds to find an optimal solution, we challenge our algorithm by testing on much larger instances. More precisely, we test with the numbers of locations varying from $1,000$ to $10,000$. Table 2 reports the computational time and the number of iterations for 20 large instances. The results clearly show the power of the OA algorithm, since it needs only less than 0.5 second to solve instances of 1,000 locations, and less than 4 seconds to solve those of 10,000 locations. It is important to note that the computing time for the OA algorithm can be approximated as $(\nu_{Q(x)} + \nu_{MP}) \times$ *n-iters*, where $\nu_{Q(x)}$ stands for the computational time to evaluate $Q(x)$, ν_{MP} is the computational time to solve the master problem by CPLEX, and *n-iters* is the number of iterations. In the cases we test, the number of iterations is small (less than 6) and seems to be not affected by the size of the instances. This explains why the OA runs quickly even for the cases of very large instances. Note that with these large instances and under our experimental setting, it is impossible for the MILP approach to find an optimal solution within several hours, even days.

$\|V\|$	$\|I\|$	OA	
		Time(s)	N^o iterations
1000	200	0.36	4
	300	0.45	5
	400	0.29	4
	600	0.29	5
2000	1000	0.52	5
	1200	0.53	5
	1400	0.53	5
	1600	0.53	4
5000	1000	1.77	4
	2000	1.94	6
	3000	1.41	6
	4000	1.14	2
7000	3000	2.05	5
	4000	1.55	2
	5000	2.21	5
	6000	2.01	4
10000	3000	3.66	6
	4000	3.99	5
	6000	2.63	4
	8000	3.66	6

Table 2: Computational results for the OA with large instances

5.3 Case study 2: Mixed MNL model

In this section we report the computational results with the MMNL model. In order to generate datasets, we assume that the utilities v_{ij} associated with a customer located in zone i and location j is no-longer deterministic, but contains an error component that follows a normal distribution of zero mean. We also assume that the variance of the error component is proportional to the distance $d(i,j)$ between zone i and location j. More precisely, v_{ij} can be written as

$$v_{ij} = \beta^T a_{ij} + \alpha \varsigma_{ij},$$

where α and β are the parameter estimates of the MMNL model, which are constants in the optimization problem, and ς_{ij} is a $N(0, d(i,j))$ random number. Note that the parameters α, β can be obtained by estimating/training the corresponding MMNL model specification, and different assumptions can be made on the error component. In this paper we do not explicitly explain how to do it, but the reader is refereed to several studies in discrete choice modeling (e.g., Train, 2003) for more details. In this experiment, we choose β as the one used for the MNL model, and $\alpha = 0.5$. For each sample size T, we generate T realizations of $\{v_{ij},\ i \in I,\ j \in V\}$ by a Monte Carlo method.

We test the OA and MILP approaches on sets of 50, 100, 200, 300, 500, 1000, 2000 and 3000 locations. For each set of locations we also randomly generate 4 sets of zones. For each dataset, we solve the maximum capture problem with the sample size T being chosen in the set $\{100, 200, 400, 600, 800, 1000\}$. In total we have 192 instances. Table 3 reports the computational results for the OA and MILP approaches. On each OA column we report the computational time and number of iterations (in the parentheses). The "-" on MILP columns indicates that the CPLEX solver fails to return an optimal solution with a time budget of 1,200 seconds. In this case we compute the percentage gaps (%) between the best objective values and the corresponding optimal values given by the OA, and if the gaps are less than 20% we report them in the parentheses.

T		100		200		400		600		800		1000	
$\|V\|$	$\|I\|$	OA	MILP	OA	MILP	OA	MILP	OA	MILP	OA	MILP	OA	MILP
50	20	0.07(5)	-(0.22)	0.08(6)	53.79	0.06(3)	-(0.09)	0.06(3)	27.84	0.32(14)	-	0.14(4)	-
	30	0.09(6)	-(0.08)	0.06(4)	-(0.02)	0.10(6)	-(7.93)	0.08(3)	-(0.06)	0.09(2)	-(5.15)	0.20(8)	73.09
	40	0.06(4)	52.46	0.13(9)	-(0.03)	0.08(4)	-(3.62)	0.12(5)	-(13.51)	0.12(4)	72.48	0.14(5)	114.5
	60	0.05(3)	-(0.28)	0.06(4)	-(3.77)	0.06(2)	-	0.13(5)	-	0.16(5)	-	0.47(14)	-(6.55)
100	60	0.11(5)	-(5.40)	0.10(4)	-	0.15(5)	-	0.19(5)	-	0.54(10)	-	0.24(4)	-
	80	0.08(3)	-	0.13(4)	-	0.35(11)	-	0.22(4)	-	0.31(6)	-	0.26(3)	-
	120	0.07(3)	-(14.87)	0.10(3)	-	0.20(6)	-	0.36(8)	-	0.36(7)	-	0.60(10)	-
	140	0.08(3)	-	0.17(5)	-	0.22(4)	-	0.26(2)	-	0.28(3)	-	0.68(8)	-
200	100	0.10(3)	-	0.18(4)	-	0.27(5)	-	0.70(10)	-	0.62(7)	-	0.44(3)	-
	200	0.11(2)	-	0.26(6)	-	0.29(4)	-	0.33(4)	-	0.48(4)	-	0.65(6)	-
	300	0.12(2)	-	0.35(6)	-	0.48(7)	-	0.57(7)	-	0.78(4)	-	1.21(8)	-
	400	0.17(4)	-	0.21(3)	-	0.31(3)	-	0.49(4)	-	0.69(5)	-	1.17(6)	-
300	200	0.33(9)	-	0.25(4)	-	0.45(6)	-	0.55(5)	-	0.63(4)	-	0.91(6)	-
	300	0.17(2)	-	0.40(8)	-	0.33(2)	-	0.64(6)	-	0.64(4)	-	1.00(7)	-
	400	0.20(4)	-	0.60(11)	-	0.55(6)	-	1.64(9)	-	1.20(5)	-	1.46(5)	-
	500	0.29(5)	-	0.33(5)	-	0.44(4)	-	0.57(4)	-	0.66(3)	-	2.05(4)	-
400	300	0.31(7)	-	0.30(4)	-	0.69(4)	-	0.48(2)	-	2.37(9)	-	1.37(5)	-
	400	0.38(9)	-	0.73(12)	-	0.82(6)	-	1.39(6)	-	1.74(10)	-	1.03(4)	-
	500	0.28(5)	-	0.44(7)	-	0.84(4)	-	1.57(2)	-	1.39(8)	-	2.18(10)	-
	600	1.04(19)	-	1.01(11)	-	1.18(5)	-	1.34(3)	-	1.88(4)	-	3.93(3)	-
500	400	0.22(2)	-	0.55(8)	-	0.63(5)	-	1.12(4)	-	1.31(5)	-	3.80(2)	-
	500	0.38(5)	-	0.51(3)	-	1.69(8)	-	2.61(7)	-	1.96(4)	-	1.67(6)	-
	600	0.53(8)	-	0.77(10)	-	0.90(4)	-	1.31(6)	-	2.95(3)	-	3.23(3)	-
	700	0.42(7)	-	0.61(6)	-	1.35(6)	-	1.62(2)	-	1.76(4)	-	2.49(2)	-
1000	700	0.40(4)	-	0.50(3)	-	0.70(2)	-	1.57(6)	-	2.46(7)	-	2.55(5)	-
	800	0.60(5)	-	0.87(5)	-	1.08(4)	-	2.43(8)	-	4.77(6)	-	4.14(5)	-
	900	0.46(5)	-	0.57(4)	-	1.16(4)	-	2.03(8)	-	2.30(5)	-	1.70(2)	-
	1100	1.11(13)	-	0.99(5)	-	1.41(3)	-	5.00(15)	-	4.75(7)	-	5.38(9)	-
2000	1000	1.11(2)	-	2.02(4)	-	5.47(4)	-	7.97(4)	-	10.91(5)	-	9.85(2)	-
	1200	2.05(6)	-	2.82(4)	-	5.45(5)	-	9.34(8)	-	12.01(7)	-	15.39(9)	-
	1400	1.85(8)	-	2.32(5)	-	6.28(7)	-	10.00(7)	-	9.85(3)	-	10.54(4)	-
	1600	1.48(4)	-	3.57(6)	-	8.0(10)	-	9.21(6)	-	12.08(7)	-	13.34(5)	-
3000	1000	3.02(4)	-	2.33(2)	-	6.12(4)	-	9.73(6)	-	10.76(4)	-	14.80(5)	-
	1500	2.73(5)	-	4.64(4)	-	8.55(4)	-	10.58(2)	-	21.13(7)	-	24.06(6)	-
	2500	3.51(4)	-	6.95(5)	-	15.19(7)	-	19.28(5)	-	18.22(5)	-	20.21(4)	-
	3000	4.11(5)	-	6.23(4)	-	14.21(6)	-	17.24(3)	-	16.11(3)	-	24.45(4)	-

Table 3: Comparison results under the MMNL model.

We have some remarks on the performance of the two approaches as follows. First, similar to the case of the MNL model, the OA shows its superiority, compared to the MILP approach. For all the instances, the OA manages to find an optimal solution very quickly; when the number of locations is less than 1000, the OA needs less than 4 seconds, and even for the most complicated cases (3000 locations and 1000 samples) the computational times are less than 25 seconds. The number of iterations are also small (less than 12) and also seems to be not affected by the size of the instances. Therefore, the computational times mostly depend on the time to compute the objective function $Q_T(x)$ and solve the master problem, which is typically fast even with large $|V|$, $|I|$ and T. Second, the MILP only enables CPLEX to find optimal solutions for a few instances of $|V| = 50$. In most of the cases, CPLEX only returns objective values with gaps are greater than 20%. Finally, in Table 3 we report the computational results for instances of up to 3000 locations, but note that even with larger numbers of locations (e.g., up to 10,000), the OA also allows us to find optimal solutions within 60 seconds.

In order to give an idea of the size of the MILP model in (MILP-MMNL), in Table 4 we report the number of variables (binary and continuous) and number of constraints (not including the business constraints) for some large instances. Note that the number of binary variables is $|V|$, which is similar to the instances under the MNL. However, under the MMNL, the number of continuous variables and constraints is increased dramatically, which makes the MILP models too heavy to be solved. This explains why the MILP approach is no-longer efficient under the MMML model even for small size instances.

T		400		600		800		1000	
$\|V\|$	$\|I\|$	n^o var.	n^o cons.	n^o var.	n^o cons.	n^o var.	n^o cons.	n^o var.	n^o cons.
1000	700	2.81E+08	5.60E+08	4.21E+08	8.40E+08	5.61E+08	1.12E+09	7.02E+08	1.40E+09
	1100	4.41E+08	8.80E+08	6.61E+08	1.32E+09	8.82E+08	1.76E+09	1.10E+09	2.20E+09
2000	1000	8.01E+08	1.60E+09	1.20E+09	2.40E+09	1.60E+09	3.20E+09	2.00E+09	4.00E+09
	1600	1.28E+09	2.56E+09	1.92E+09	3.84E+09	2.56E+09	5.12E+09	3.20E+09	6.40E+09
3000	1000	1.20E+09	2.40E+09	1.80E+09	3.60E+09	2.40E+09	4.80E+09	3.00E+09	6.00E+09
	3000	3.60E+09	7.20E+09	5.40E+09	1.08E+10	7.20E+09	1.44E+10	9.01E+09	1.80E+10

Table 4: Size of the MILP (numbers of variables and constraints) under the MMNL model

5.4 Comparison of the OA algorithm and existing mixed integer nonlinear programming (MINLP) solvers

In this section we compare the performance of Algorithm 1 to other existing convex MINLP solvers that are also based on the outer approximation scheme. More precisely, we compare our OA algorithm with algorithms implemented in the BONMIN package (Bonami et al., 2008). Note that the BONMIN contains 4 different algorithms for solving convex MINLP problems, and two of them are based on the outer approximation scheme, i.e., one is an outer-approximation decomposition algorithm (denoted as BONMIN-OA) and the other one is a hybrid outer-approximation based branch-and-cut algorithm (denoted as BONMIN-HYB). We refer the reader to Bonami et al. (2008) for a detailed description of these algorithms. We use an MATLAB interface of BONMIN, i.e., the *OPTI Toolbox* (`http://www.i2c2.aut.ac.nz/Wiki/OPTI/`) for the experiment.

We first group instances into sets that have the same number of locations, we then report the average computing time for each set. Figures 2 and 3 show the average computing time for the MNL and MMNL models, where the vertical axis is the computing time (seconds), and the horizontal one is the number of locations. We first remark that all the approaches are able to find optimal solutions in less than 80 seconds, which clearly shows the superiority of the outer-approximation based approaches, compared to the MILP one. Even the number of iterations required by the OA and BONMIN-OA is similar, the OA is generally faster than the BONMIN solvers, and the difference between the OA and BONMIN-OA/BONMIN-HYB becomes more remarkable on MMNL instances, especially when the sample size T is increased. This is due to the difference between our OA and the outer-approximation algorithm described in Bonami et al. (2008), namely, we do not solve the continuous relaxation of (P1) and we save solutions after each iteration to avoid recomputing the objective function.

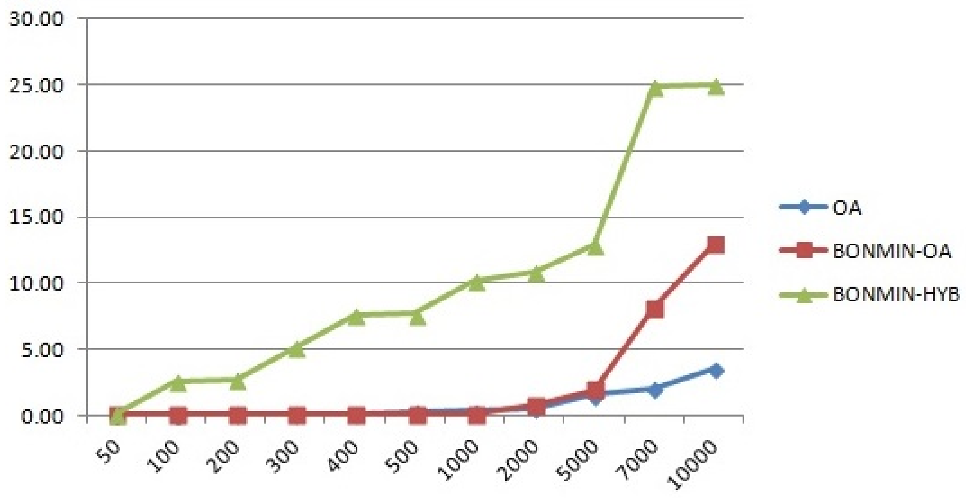

Figure 2: Comparison results under the MNL model

6 Conclusion

In this paper we presented an innovative (and exact) algorithm to deal with the competitive facility location problem under random utility maximization framework. Two aspects have been considered, namely, the IIA property of the MNL model used in the existing studies, and the intractability of large-scale instances. We deal with these aspects by formulating the problem with the MMNL model, which is known to be fully flexible to capture customers' demand. Under this model, existing approaches in the context (based on MILP formulations) are no-longer efficient, even with small instances. We have proposed a new algorithm based on the convexity of the objective functions, and the outer-approximation scheme. We have tested our algorithm using generated large-scale instances and showed that the OA algorithm impressively dominated the MILP approach, especially in the case of the MMNL model. For example, the OA manages to easily find optimal solutions for very large instances (up to 3,000 locations, and

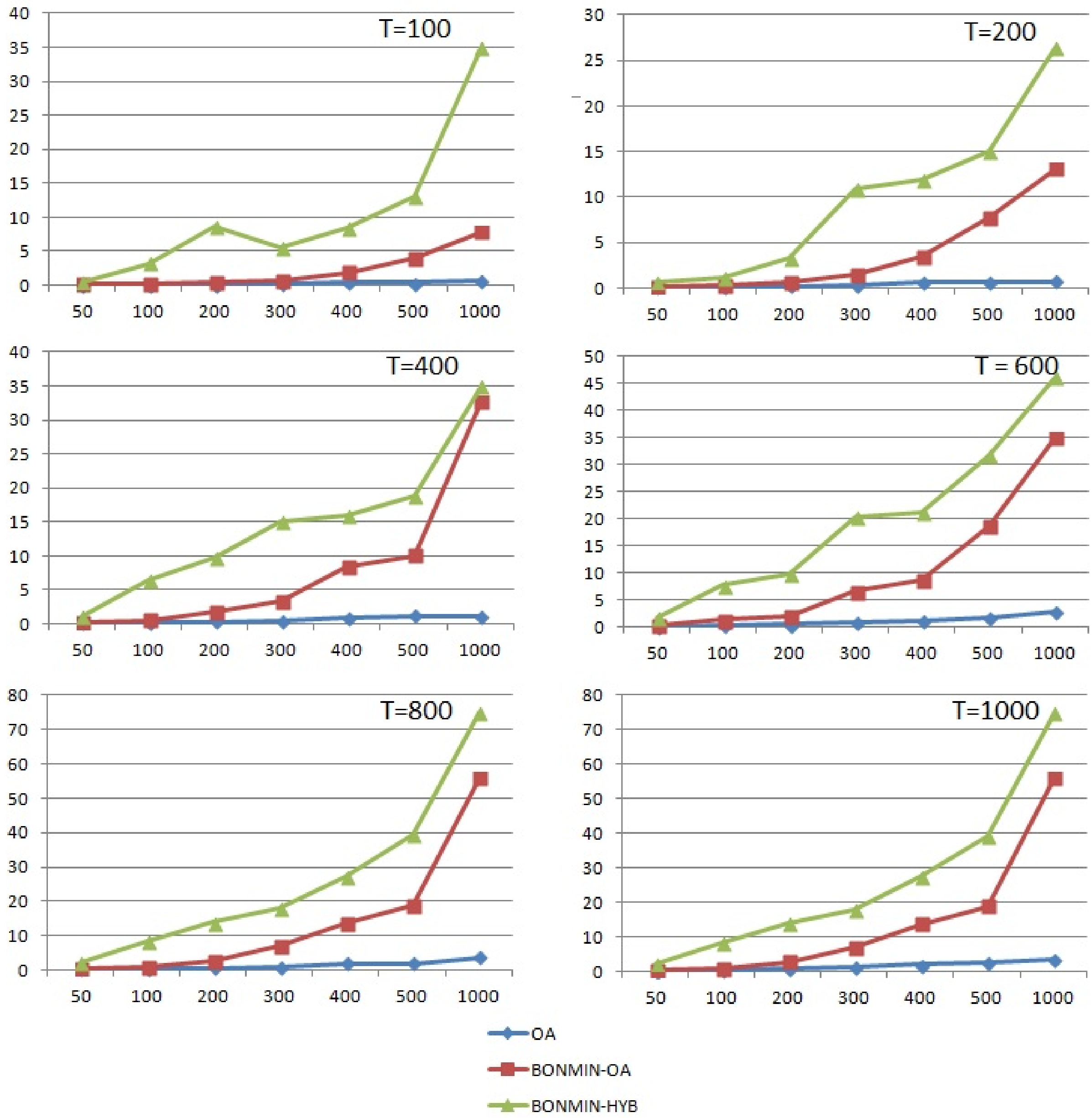

Figure 3: Comparison results under the MMNL model, with $T \in \{100, 200, 400, 600, 800, 1000\}$

sample size is 1,000) in seconds, while the MILP approach fails to converge to optimal solution within 1,200 seconds for several small instances (50 locations) and seems to never converge to optimality for larger instances (more than 200 locations).

We also compared the performance of our algorithm with two outer-approximation based algorithms implemented in the BONMIN package (Bonami et al., 2008). The results showed that these algorithms also manage to quickly find optimal solutions for all the instances considered, which demonstrates the efficiency of the outer approximation scheme in the context. We also showed that the BONMIN's solvers perform quickly for small and medium size instances, but become less efficient for large size instances, compared to our OA algorithm.

The impressive performance of the OA opens many interesting directions for future research, e.g., the facility location under more complicated constraints, the maximal covering salesman problem under random utility maximization models. This is also interesting to apply the models and methods presented in this paper to real large-scale applications, so the advantages of the MMNL model as well as the power of the OA algorithm can be better demonstrated.

Finally, we note that the MEV model is also flexible to model the customers' demand. However, the use of such model would result in highly nonlinear non-convex objective functions, which is difficult to solve. The algorithm proposed in this paper is designed based on the convexity of the objective function, so it cannot apply for the case of MEV model. Nevertheless, the problem under the MEV remains to be interesting to investigate. We are also interested in non-parametric choice models, e.g., the generic ranking-based choice model (Farias et al., 2013), which is able to represent any choice models based on random utility maximization.

Acknowledgment

This work has been supported by the Canada Excellence Research Chair in Data Science for Real-Time Decision-Making held by the second author Andrea Lodi.

References

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.

Ben-Akiva, M. and Bierlaire, M. Discrete choice methods and their applications to short-term travel decisions. In Hall, R., editor, *Handbook of Transportation Science*, pages 5–34. Kluwer, 1999.

Ben-Akiva, M. and Lerman, S. R. *Discrete Choice Analysis: Theory and Application to Travel Demand*. MIT Press, Cambridge, Massachusetts, 1985.

Ben-Akiva, M. *The structure of travel demand models*. PhD thesis, MIT, 1973.

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.

Berman, O., Drezner, T., Drezner, Z., and Krass, D. Modeling competitive facility location problems: New approaches and results. In *Decision Technologies and Applications*, pages 156–181. INFORMS, 2009.

Bertsimas, D. and Mišic, V. V. Data-driven assortment optimization. Working paper, MIT, 2016.

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

Church, R. and Velle, C. R. The maximal covering location problem. *Papers in regional science*, 32(1):101–118, 1974.

Daly, A. and Bierlaire, M. A general and operational representation of generalised extreme value models. *Transportation Research Part B*, 40(4):285 – 305, 2006.

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.

Farias, V. F., Jagabathula, S., and Shah, D. A nonparametric approach to modeling choice with limited data. *Management Science*, 59(2):305–322, 2013.

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.

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. and Müller, S. Insights into clients choice in preventive health care facility location planning. *OR spectrum*, 37(1):273–291, 2015.

Hasse, K. *Discrete location planning.* Institute of Transport and Logistics Studies, 2009.

Huff, D. L. Defining and estimating a trading area. *The Journal of Marketing*, pages 34–38, 1964.

Mai, T., Frejinger, E., Fosgerau, M., and Bastin, F. A dynamic programming approach for quickly estimating large network-based mev models. *Transportation Research Part B*, 98: 179–197, 2017.

McFadden, D. Modelling the choice of residential location. In Karlqvist, A., Lundqvist, L., Snickars, F., and Weibull, J., editors, *Spatial Interaction Theory and Residential Location*, pages 75–96. North-Holland, Amsterdam, 1978.

McFadden, D. and Train, K. Mixed mnl models for discrete response. *Journal of applied Econometrics*, pages 447–470, 2000.

Munger, D., LEcuyer, P., Bastin, F., Cirillo, C., and Tuffin, B. Estimation of the mixed logit likelihood function by randomized quasi-monte carlo. *Transportation Research Part B*, 46(2): 305–320, 2012.

Nocedal, J. and Wright, S. J. *Numerical Optimization.* Springer, New York, NY, USA, 2nd edition, 2006.

ReVelle, C. The maximum capture or sphere of influence location problem: Hotelling revisited on a network. *Journal of Regional Science*, 26(2):343–358, 1986.

Train, K. *Discrete Choice Methods with Simulation.* Cambridge University Press, 2003.

Vovsha, P. and Bekhor, S. Link-nested logit model of route choice Overcoming route overlapping problem. *Transportation Research Record*, 1645:133–142, 1998.

Wu, T.-H. A note on a global approach for general 0–1 fractional programming. *European Journal of Operational Research*, 101(1):220–223, 1997.

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.

A Proof of Proposition 1

We first introduce the following lemma

Lemma 3 *Given a vector α of size $(n \times 1)$, and a scalar α_0, function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ defined as*

$$f(x) = \frac{1}{\alpha^T x + \alpha^0},$$

is a convex function.

Proof. Proof of Lemma 3. We just need to prove that the second derivative of $f(x)$ is positive semidefinite. We have the first and second derivatives of $f(x)$ are

$$\nabla f(x) = \frac{-\alpha}{(\alpha^T x + \alpha^0)^2}$$

$$\nabla^2 f(x) = \frac{2\alpha\alpha^T}{(\alpha^T x + \alpha^0)^3}.$$

So it is obvious that $\nabla^2 f(x)$ is a rank-1 positive semidefinite matrix (positive definite if $\alpha \neq 0$), thus $f(x)$ is a convex function. ■ ■Now we prove that $Q(x)$ in (P1) is a convex function. Indeed, we have for each $i \in I$,

$$-\frac{\sum_{j=1}^m x_j V_{ij}}{\sum_{j=1}^m x_j V_{ij} + U_Y^i} = -1 + \frac{U_Y^i}{\sum_{j=1}^m x_j V_{ij} + U_Y^i}.$$

According to Lemma 3 we have that $\frac{U_Y^i}{\sum_{j=1}^m x_j V_{ij} + U_Y^i}$ is convex, meaning that $-\frac{\sum_{j=1}^m x_j V_{ij}}{\sum_{j=1}^m x_j V_{ij} + U_Y^i}$ is also convex, so the $Q(x)$. In a similar way we can also prove that $Q^T(x)$ defined in (P2) is also a convex function.

B Proof of Proposition 2

We first prove the inequality for $Q(x)$ by showing that

$$\frac{\sum_{j=1}^m x_j V_{ij}}{\sum_{j=1}^m x_j V_{ij} + U_Y^i} \leq \frac{\sum_{j=1}^m V_{ij}}{\sum_{j=1}^m V_{ij} + U_Y^i}, \; i \in I, \forall x \in \{0,1\}^m \tag{10}$$

Indeed, (10) is equivalent to

$$\left(\sum_{j=1}^m x_j V_{ij}\right)\left(\sum_{j=1}^m V_{ij} + U_Y^i\right) \leq \left(\sum_{j=1}^m x_j V_{ij} + U_Y^i\right)\left(\sum_{j=1}^m V_{ij}\right)$$

$$\Leftrightarrow U_Y^i \left(\sum_{j=1}^m x_j V_{ij}\right) \leq U_Y^i \left(\sum_{j=1}^m V_{ij}\right)$$

$$\Leftrightarrow \left(\sum_{j=1}^{m} x_j V_{ij}\right) \leq \left(\sum_{j=1}^{m} V_{ij}\right),$$

which is indeed valid because $x \in \{0,1\}^m$. So the have

$$Q(x) = -\sum_{i\in I} q_i \left(\frac{\sum_{j=1}^{m} x_j V_{ij}}{\sum_{j=1}^{m} x_j V_{ij} + U_Y^i}\right) \geq -\sum_{i\in I} q_i \left(\frac{\sum_{j=1}^{m} V_{ij}}{\sum_{j=1}^{m} V_{ij} + U_Y^i}\right),\ \forall x \in \{0,1\}^m.$$

The inequality for $Q_T(x)$ can be proved in a similar way.