Università degli Studi di Milano
Dipartimento di Informatica

Paper Title:

"A local iterative linear approximation framework to integrate discrete choice and mathematical optimization models"

Authors:

Daniel Dissegna (corresponding author)
Università degli Studi di Milano,
Dipartimento di Informatica,
via Celoria 18, 20133 Milano (Italy)
ddissegna@gmail.com

Alberto Ceselli
Università degli Studi di Milano,
Dipartimento di Informatica,
via Celoria 18, 20133 Milano (Italy)
alberto.ceselli@unimi.it

Ethical and conflict of interest statement:

We hereby confirm that the paper contains entirely new contributions, and will be submitted to no other journal or conference during the review process. We also state compliance to the ethical and conflict of interest policies of your journal.

A local iterative linear approximation framework to integrate discrete choice and mathematical optimization models

Daniel Dissegna [*], Alberto Ceselli
Università degli Studi di Milano
Dipartimento di Informatica
Via Celoria 18, 20133 Milano (Italy)

August 2022

Abstract

We consider a class of optimization problems having the following common feature: user demands appear as a term, whose values depend on the discrete choice made by each individual in a certain population. This type of problems arise frequently, for instance in revenue management applications of transportation systems. In the literature, discrete choice models and mathematical programming are known to be effective, respectively in describing users behaviour, and formulating optimization problems. Their integration, to account for user choices which depend on optimization decisions, is however an issue.

We introduce an algorithmic methodology to perform such an integration. Its main idea is to perform local approximations of the choice probabilities in terms of simplified functions, to formulate them as terms of a mixed integer program representing the optimization problem, and to exploit the solutions obtained by the optimization process to refine the local approximation.

We evaluate its applicability and effectiveness through experiments on a revenue maximization problem from the literature, and a few of its variants, exploiting two real world discrete choice models.

Our experiments show our approach to outperform recent ones from the literature by orders of magnitude in terms of computing time, improving solutions accuracy as well.

1 Introduction

Correctly estimating demand values is often an issue in optimization models for decision support. Common practices rely on forecasting, perform parametric

[*]Corresponding Author: ddissegna@gmail.com

analyses or recur to probabilistic settings. Most of the time, these methods makes too simplistic and sometimes unrealistic assumptions. Therefore, we need to move to more appropriate representations that consider the individuals as the ultimate decision makers, whose heterogeneous behavior has a direct impact on the system and on the decisions to be taken. In this paper we therefore consider the following broad class of optimization problems: a system is given, in which a set of services is proposed to users. There is a single decision maker, willing to take decisions on some system parameters, maximizing an objective function. Users are individuals: according to the system parameter settings, the preference of each of them changes, ultimately affecting the demand of each service.

Several practical problems fall into this setting. For instance, the class of Revenue Management problems [Ryz05], the class of Facility Location problems [LNS19], the road tolling [LMS98] and railway timetabling problems [CR11].

The main issue is often the following: demand depends on users, who express their preferences individually, based on the perceived relative utility of various alternatives. This is the fundamental principle of Discrete Choice Models (DCM): based on the random utility principle they provide a disaggregated demand representation and are able to capture in high detail the heterogeneity of individual preferences [Tra09].

The *descriptive* modeling advantage provided by DCMs, however, cannot unfold without a proper way of embedding them in a *prescriptive* model. Such an integration, however, is far from simple. The reasons are clear when considering mathematical programming as a reference framework for optimization. Intuitively, the optimization problem can be modeled as a Mixed Integer Program (MIP), containing some terms representing demand values which affect the optimization process. In turn, the demand values are determined by a DCM, having among its attributes some of the decision variables of the MIP. A straightforward embedding of the DCM in the MIP yields non-linear, in general non-convex, MIPs which therefore become quickly intractable.

In order to come up with tractable and more efficient solutions, different embeddings, making different assumptions on the given problem, have been developed. Nevertheless, as stressed in [Pan+21], these assumptions are usually custom and problem dependent.

An effective solution for the integration of generic DCMs in generic MIPs has been recently proposed in [Pan+21]. The authors in fact consider the set of variables of the decision problem, and split them in hexogenous ones (those actually encoding decisions that are part of both the MIP and the DCM) and endogenous ones (those which are involved in one of the two models, but not both). Then, they generate scenarii from the utility functions of the DCM, by means of simulation runs, to approximate the demand with the Sample Average Approximation Method [KSH02] in terms of the utility functions. Then, they embed this representation in the MIP formulation. This framework proves effective in a Revenue Maximization (RM) case study taken from [Ibe+14]. It is related to the choice of price for two out of three parking alternatives in a city. The operator aims to find the optimal prices of the parking options in order to maximize its revenue. The population of interest is composed of a set of N individuals, whose behavior is describ ed by a DCM. The overall revenue is defined as the sum of products of the price of each parking option by the number of users who choose it.

The approach of [Pan+21] has however one fundamental drawback: the size of the MIP grows in the number of scenarii, which in turn grows in the number of individuals of the DCM. This yields quickly very large MIPs.

In this paper we propose a more effective integration and algorithmic framework for achieving the results of [Pan+21], concerning the integration of MIPs and DCMs.

Our contributions are the following. First, we introduce a new integration methodology for DCMs in MIPs, whose complexity does not depend on the number of individuals composing the DCM, thereby overcoming the weaknesses of [Pan+21]. It therefore improves the state of the art in terms of accuracy and computing effort. Second, we analyze its effectiveness in the Revenue Maximization case, providing an experimental analysis using two DCMs from the literature obtained by fitting real data.

2 Models and algorithms

In the following we introduce notation and background. Then we introduce our framework, and we compare it to the literature from a structural point of view.

2.1 Discrete Choice Model Notation

Let a DCM be given. In details, let us define a set of *individuals* N, indexed by n, and a set of *alternatives* J (choice set), indexed by j. Let us define also a set A of attributes of the DCM, explaining the choice of the individuals. Each attribute may be individual specific, alternative specific, or depend on both. For instance, in the application of [Ibe+14], an individual specific attribute is whether the individual is a resident or not and an alternative specific attribute is the parking fee. We split the set of attributes in two parts: D, including those which do not depend on the choices made by the system decision maker, e.g. individual's residence, and E, including those determined according to different choices of the decision maker, e.g. the parking fee. Let x^d be a vector, containing the specific values for the attributes in D (i.e they are obtained directly from the available dataset), and x^e be a vector of decision variables of the optimization process for the attributes in E. The specification of D and E is fully general. Nevertheless, we remark that, in practical settings, one expects the decision maker to act on either alternative specific or individual and alternative specific attributes, and not on individual specific attributes (i.e it can decide the price of the parking lot but not the residence of an individual).

In a DCM the preference of individuals is represented by a utility function, which for each individual $n \in N$ associates a utility value to each alternative $j \in J$:

$$U_{nj}(x^e_{nj};\epsilon_{nj}) = V_{nj}(x^e_{nj}) + \epsilon_{nj}. \tag{1}$$

In (1), $V_{nj} : \mathbb{R}^{|E|} \to \mathbb{R}$ is the so called representative utility, that includes everything that can be modelled by the researcher, and ϵ_{nj} is a random term, that captures everything that has not been included explicitly in the DCM and is independent of the attributes values x^e_{nj} and x^d_{nj} associated with individual n and alternative j. Since the random term ϵ_{nj} is modelled as a random variable, $U_{nj}(x^e_{nj};\epsilon_{nj})$ is also a random variable.

The representative utility V_{nj} takes the following form:

$$V_{nj}(x^e_{nj}) = h_{nj}(x^e_{nj};\beta_{nj}) + g_{nj}(x^d_{nj}) \tag{2}$$

The term $g_{nj}(x^d_{nj})$ captures the contribution to utility which depends only on the attributes in D and therefore can be preprocessed, becoming simple data. The term $h(x^e_{nj};\beta_{nj})$, instead, is a function which depends on the attributes encoding decision variables in E, and on some parameters β_{nj} of the DCM, which are data. The parameters β_{nj} come from DCM modeling, that is from either domain knowledge, or by fitting historical data; the values x^e_{nj} are instead assumed to be output of an optimization process.

The behavioral assumption is that individual n chooses alternative j if the corresponding utility is the largest within the choice set J [Man77]. We assume that each individual chooses precisely one alternative. The *choice probability* of individual n for alternative j is

$$P_{nj}(x^e) = Pr[U_{nj}(x^e_{nj};\epsilon_{nj}) \geq U_{nk}(x^e_{nk};\epsilon_{nk})\, \forall k \in J] \tag{3}$$

Different DCMs are obtained by assuming different distributions on the random term ϵ_{nj}. For instance, the logit model is obtained by assuming that ϵ_{nj} are independent and identically distributed (i.i.d.) across both n and j, following a standard Gumbel distribution.

It can be shown [Tra09] that the choice probability (3) of the logit model can be written as

$$P_{nj}(x^e) = \frac{e^{V_{nj}(x^e_{nj})}}{\sum_{k\in J} e^{V_{nk}(x^e_{nk})}} \tag{4}$$

The logit model exhibits the *independence from irrelevant alternatives* 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. Since this may be unrealistic in real contexts, several methods have been proposed to relax this property. The mixed logit model is often a convenient choice, as it is highly flexible in the sense that it can approximate any random utility model [MT00].

In the mixed logit model, instead, the function $h_{nj}(x^e_{nj})$ of (2) is assumed to be linear on x^e_{nj}, that is

$$V_{nj}(x^e_{nj};\beta_n) = \beta_n \cdot x^e_{nj} + g_{nj} \tag{5}$$

where each β_n is a vector of $|E|$ elements, each assumed to be randomly distributed across the population. The vector of coefficients β_n is therefore a random vector with probability density function $f(\beta|\theta)$, where the vector θ contains the parameters of the distribution β_n. The choice probability is the integral of the standard logit formula (4) over the density $f(\beta|\theta)$

$$P_{nj}(x^e) = \int \frac{e^{V_{nj}(x^e_{nj};\beta_n)}}{\sum_{k\in J} e^{V_{nk}(x^e;\beta_n)}} f(\beta|\theta)d\beta \tag{6}$$

In most of the cases, the expression associated with the choice probability (6) has no closed form. So in order to estimate the model it is common to approximate the choice probability exploiting a Monte Carlo method. This is done by generating R *scenarios*, that are a set of values $\beta_{n1}\dots\beta_{nR}$ drawn from the distribution of β.

The choice probability (6) can be approximated as

$$P_{nj}(x^e) \approx \frac{1}{R}\sum_{r=1}^{R} \frac{e^{V_{nj}(x^e_{nj};\beta_{nr})}}{\sum_{k\in J} e^{V_{nk}(x^e_{nk};\beta_{nr})}} \tag{7}$$

the higher the value of R, the better the approximation.

An expected demand D_j for alternative j is then given by

$$D_j = \sum_{n\in N} P_{nj}(x^e) \tag{8}$$

Notice that, for both the logit and the mixed logit model, since the expressions of the choice probability (4) and (7) are non-linear in the variables x^e, so is the expression associated with the expected demand (8).

2.2 Optimization Model Notation

Let us now consider the optimization point of view. In particular, we define the following family of optimization models:

$$\text{OM)} \quad \text{maximize } f(x^e; x^o; D_j) \tag{9}$$

$$\text{s.t. } D_j = \sum_{n\in N} P_{nj}(x^e) \qquad \forall j \in J \tag{10}$$

$$(x^e; x^o; D_j) \in X \tag{11}$$

where x^o is a set of additional generic decision variables (not included in the DCM). For instance, considering the DCM in [Ibe+14], we can formulate the following Uncapacitated Revenue Maximization problem (from now URM). $x^e = y$ is a vector of continuous variables of dimension $|J|$, each representing the price of a parking option j, D_j is the number of individuals choosing parking option j and the objective is to maximize the sum of revenues $y_j \cdot D_j$ for each alternative, that is

$$
\begin{aligned}
\text{URM)} \quad & \text{maximize} \sum_{j \in J} y_j \cdot D_j & & (12) \\
& \text{s.t. } D_j = \sum_{n \in N} P_{nj}(y) & \forall j \in J \quad & (13) \\
& l_j \leq y_j \leq m_j & \forall j \in J \quad & (14)
\end{aligned}
$$

where constraints (14) impose upper and lower bounds to prices, although the coefficients in the vectors l and m can be arbitrary.

2.3 A Local Iterative Linear Approximation algorithm

For illustrative purposes, we consider the case of the mixed logit model, which we recall is known to be able to approximate any DCM.

Recall that, for the mixed logit model, the expected demand of an alternative is given by (7) and (8). Since (7) is non-convex in the decision variables x^e, its direct insertion into a MIP quickly yields intractable problems.

We in fact propose a further step, in the form of a Local Iterative Local Approximation framework (LILA). The idea behind LILA is the following. We replace the *expression* of $P_{nj}(x^e)$ using simpler *functions*. In particular, we propose to restrict to first order Taylor approximations.

We remark that this is in contrast to the approach of [Pan+21], in which the expected demand is approximated as a MIP by computing and averaging *values* in expression (3) by means of simulation runs. A full discussion on structural advantages and applicability limitations of our method is reported in Section 2.4. It can however be readily noticed that it allows for the integration of any type of DCM for which there exist a differentiable either closed form or an analytical approximation of the choice probabilities.

Given a point $z \in R^{|E|}$, the choice probability can be approximated as

$$P_{nj}(x^e) \approx P_{nj}(z) + \nabla P_{nj}(z) \cdot ((x^e) - z) \tag{15}$$

where $\nabla P_{nj}(z)$ is the gradient of $P_{nj}(x^e)$ evaluated in z. Now, we can rewrite the expression of the expected demand (8) as

$$
\begin{aligned}
D_j &= \sum_{n}^{N} \left(P_{nj}(z) + \nabla P_{nj}(z) \cdot (x^e - z)\right) \\
&= \sum_{n}^{N} P_{nj}(z) + \sum_{n}^{N} \nabla P_{nj}(z) \cdot x^e - \sum_{n}^{N} \nabla P_{nj}(z) \cdot z
\end{aligned} \tag{16}
$$

That is, such an approximation of the expected demand D_j is linear. We have

$$D_j = \alpha_j(z) \cdot x^e + q_j(z)$$

where, when z is fixed,

$$q_j(z) = \sum_{n}^{N} P_{nj}(z) - \sum_{n}^{N} \nabla P_{nj}(z) \cdot z$$

is a costant and

$$\alpha_j(z) = \sum_{n}^{N} \nabla P_{nj}(z)$$

is a costant vector of dimension $|E|$.

Integration. The integration of the DCM into the optimization formulation is then obtained by simple linear constraints. In our general optimization model, it yields the following MIP:

$$\begin{aligned} \text{TOM(z)} \quad & \text{maximize } f(x^e; x^o; D_j) && (17)\\ & \text{s.t. } D_j = \alpha_j(z) \cdot x^e + q_j(z) \qquad \forall j \in J && (18)\\ & (x^e; x^o; D_j) \in X. && (19) \end{aligned}$$

For instance, in the case of URM, since $x^e = y$, simplifies as follows:

$$\begin{aligned} \text{TURM(z)} \quad & \text{maximize } \sum_{j \in J} y_j \cdot D_j && (20)\\ & \text{s.t. } D_j = \alpha_j(z) \cdot y + q_j(z) \qquad \forall j \in J && (21)\\ & D_j \geq 0 \qquad \forall j \in J && (22)\\ & l_j \leq y_j \leq m_j \qquad \forall j \in J && (23) \end{aligned}$$

and further simplifies by using (21) to project out variables D_j. The resulting TRM(z) formulation is therefore a (continuous) quadratic optimization problem (QP) on the decision variables y_j.

Resolution algorithm. The pseudocode of our LILA algorithm is provided in Alg.1 It works as follows

1. we start by a tentative z^{init} value as a starting point to get an initial approximation (together with possibly lower and upper bounds of x^e and an accuracy threshold thr)

2. we solve the optimization model with terms $\alpha_j(c)$ and $q_j(c)$ evaluated on z^{init}, thus obtaining optimal values x^m x^o and D^m for the decision variables x^e x^o and for the demand D;

3. we calculate the demand D^r by performing a computation of the DCM via (7), using the values of the solution x^m obtained at step 2. That is, while D^m is a local estimate of demands, D^r is a refined evaluation provided by the DCM usage. Then, we calculate the objective function value corresponding to x^m and this refined demand D^r;

Algorithm 1 LILA resolution algorithm

Require: z^{init}, thr
  D^r, D^m, z = 0, $+\infty$, z^{init}
  while dist(D^r, D^m) $\geq$ thr **do**
    x^m, x^o, D^m = solve TOM(z)
    D^r = compute DCM(x^m)
    val = f(x^m, x^o, D^r)
    $z = x^m$
  end while
  return val

4. we calculate a difference (e.g. the Euclidean distance) between the demand values D^r, produced by the DCM computation, and D^m, obtained by the optimization model; if it is less than thr, then convergence is reached and we STOP, otherwise we fix $z = x^m$ and go to step 2

Therefore, the algorithm stops when the demand estimate is *robust*, in the sense that it is coherent with respect to both the DCM model and the optimization one. This means in turn that the optimization model has found a solution that does not differ much from the one found at the previous iteration, as the approximation point c of the current iteration corresponds to the solution found at the previous iteration. This allows the algorithm to converge to a local optimum. We must note that, during computation, the algorithm may find solutions x^m, x^o that are not feasible w.r.t. D^r. This is due to the fact that the model optimize w.r.t an approximation of the demand and, as such, D^m may be feasible but not its refined estimate D^r. Nevertheless, the last solution found will be feasible since D^m will be robust w.r.t to D^r.

Improving the precision of the approximation. Since the framework is a based on a local approximation, the model approximates the demand less precisely as we move away from the point c.

This problem can be mitigated by tightening the lower and upper bound of the demand so to restrict the feasible region without loosing too much precision in the approximation. In this way, it is possible to regulate the search between exploration and intensification: by increasing the exploration we admit solutions with a potentially poor approximation of the demand; by intensifying, we focus the search on solutions with a good approximation of the demand.

This can be easily integrated into the model by inserting parametric constraints on the demand, which allow to adjust how much it can vary w.r.t. the value associated with the point of the approximation z (which is the solution obtained in the previous iteration). For instance, we can add the following parametric constraints on μ

$$(1-\mu)\, D_j^z \leq D_j \leq (1+\mu)\, D_j^z \quad \forall j \tag{24}$$

Where $\mu \in [0, 1]$ and D_j^z is the value of the demand (D^r) associated to the approximation point z.

For instance, the following policy can be adopted to ensure initial exploration and gradual intensification: initialize the value of μ to 1 and halve its value every t iterations (i.e. $t = 10$).

2.4 Structural comparison to the literature

Our framework gives a new method to solve the problem resulting from the integration of a disaggregate demand representation given by a DCM into an optimization problem. It differs in particular from [Pan+21], in which MIP expressions representing the demand are generated, inserted into the OM and solved only once. Our approach resembles more a local search algorithm in which we iteratively generate a local approximation of the demand in the form of a linear function, insert into the OM and solve until convergence.

Both [Pan+21] and our method rely on the formulation of the optimization problem as a MIP. However, the approach of [Pan+21] make the size of the MIP to grow with the number of individuals of the DCM and the number of scenarios generated. In our formulation, instead, its size does not depend on the number of individuals and does not rely on scenario generation. Being linear expressions, our demand approximations can be efficiently integrated into the MIP. This allows us to potentially integrate large scale DCMs with no change in the computational effort.

On the other hand, our approach can be applied under the condition that the choice probabilities are differentiable functions in the decision variables x^e. Some model rewriting is of course possible. For instance, the decision variables x^e must be continuous, but x^o integer decision variables may appear in the optimization model, affecting x^e values indirectly (as long as they do not appear directly as terms of the DCM). Furthermore, while [Pan+21] requires a single MIP to be solved, our approach requires a sequence of MIP to be iteratively solved.

In Section 3 we indeed show such a drawback to be marginal.

3 Experimental evaluation

The objective of this section is twofold. First, we deal with the integration of two different real world DCMs within the Uncapacited Revenue Maximization Problem (Section 2.2) using both our LILA approach and the approach of [Pan+21] (PBGA in the reminder) and conduct an analysis of the performance of each framework in terms of solutions quality and computing times. Second, we deal with the integration of a DCM in different kinds of RMs (Capacitated Case, Population Segmentation, Capacity Allocation) and proving the flexibility of our approach.

3.1 Discrete Choice Models

For evaluating the integration in the context of URM, we rely on two DCMs, discussed hereafter.

Parking model The first DCM (from now on Parking model) is developed and tested in [Ibe+14]. We had direct access to the original data, which were kindly provided us by the authors of [Ibe+14].

The model is motivated by the economic viability of an underground parking in the city of Santoña (Spain), which aims to represent the user parking choice (demand) in a disaggregated way. A park and ride situation is assumed where users leave their vehicles in parking lots for the whole day and then travel to their final destination by public transport.

The choice set is defined by three services: free on-street parking (FSP), paid on-street parking (PSP) and paid parking in an underground car park (PUP).

The attributes of individuals are:

- $Origin_{FSP} \in \{0, 1\}$, 1 if the origin of the trip is within the city, 0 otherwise
- $LowInc \in \{0, 1\}$, 1 if the user's salary is less than 1200 euros/month, 0 otherwise
- $Resident \in \{0, 1\}$, 1 if the individual is a resident of the city, 0 otherwise
- $AgeVeh \leq 3 \in \{0, 1\}$, 1 if the vehicle's age is less than or equal to 3, 0 otherwise

The attributes of the alternatives are:

- AT, access time to the parking space after arriving in the parking lot
- TD, access time from the parking lot to the final destination
- FEE, parking fee

The DCM is defined as a mixed logit model. The coefficients of the attributes AT and FEE are modelled as correlated normal variables. The FEE attribute of the alternatives PSP and PUP is treated as decision variable and therefore corresponds to the price decision variable y_j when the DCM is integrated into URM. The alternative FSP has no FEE and therefore its price decision variable is set to 0 (it is the opt-out alternative). For a complete specification of the parameters, see table 6 of Appendix A. The original dataset yielding the model is composed by $N = 197$ individuals.

ModeCanada model The second DCM has been obtained by a dataset taken from the R package mlogit [Cro]. It provides a sample of 3880 travellers for the Montreal-Toronto corridor. The corresponding DCM was developed by us and is defined as follows.

The choice set is defined by three transportation modes: train (TRAIN), airplane (AIR) and the individual's car (CAR).

The only attribute of the individual is:

- INC, income of the individual

The attributes that vary over both individuals and alternatives are:

- IVT, in-vehicle time
- OVT, out-vehicle time
- $COST$, monetary cost
- $FREQ$, frequency

The DCM is defined as a mixed logit model. The coefficient of the attribute IVT is modelled as a normal variable and thus varies over individuals. For a complete specification of the parameters, see table 7 of Appendix A.

The integration within the RM is performed by treating the price decision variables y_j as a surplus on the $COST$ attribute of the alternative j. This corresponds to an equal increase or decrease in the cost of the transportation mode j for all individuals. The value of the objective function then becomes the surplus revenue that we can make by performing an increase or decrease in the cost of the services for all individuals. The opt-out alternative is CAR and has surplus $y_{CAR} = 0$. The dataset of the model is composed by $N = 2779$ individuals.

3.2 Experimental Evaluation

We performed the integration of the two DCMs in URM with both LILA and PBGA.

The integration of a general DCM in URM with LILA has already been illustrated in Section 2.3. For the integration with PBGA, it is needed to specify two parameters: the number R of scenarios which are generated and incorporated into the optimization model and the cardinality S of the sample of individuals from the population N used in this scenario generation. For the resulting MIP formulation, we refer to the original paper [Pan+21].

All experiments have been performed by code implementations in Python 3.8 using the DOcplex Python Modeling API library to embed the IBM CPLEX solver version 20.1. Tests were run on a machine with an AMD Ryzen 1950x 3.4GHz 16 core CPU and 32GB RAM.

Key Performance Indicators. Summarizing, a solution of the URM with a DCM representation of the demand is composed by the following attributes:

- *price* y_j of each alternative $j \in J$, found by the optimization model
- *approximated demand* D_j^m estimated by the optimization model, corresponding to prices y_j
- *approximated revenue* g^m estimated by the optimization model, which is obtained as $g^m = \sum_{j \in J} y_j \cdot D_j^m$

To evaluate the robustness of a solution obtained from a given framework, we evaluated the corresponding DCM via simulation with R $= 10^4$ scenarios by using the entire population N and the prices of the alternatives set to p. In this way we calculate a refined demand estimate D^r using the DCM, and the corresponding DCM revenue $g^r = \sum_{j \in J} y_j \cdot D_j^m$ of the solution.

Given these, we defined 4 *Key Performance Indicator*:

- *revenue*: DCM revenue g^r; measures the quality of the solution and therefore the effectiveness of the framework according to DCM simulation
- Δ *revenue*: relative difference between the approximated revenue g^m found by the optimization model and the refined DCM revenue g^r; measures the robustness of the solution in terms of quality
- Δ *demand*: sum of the squared differences between the approximated demand D^m and the refined DCM demand D^r, normalized w.r.t. the norm of the DCM demand D^r

  $$||D^r - D^m||_2/||D^r||_2$$

  it measures the robustness of the solution in terms of convergence
- *CPU*: CPU time needed to obtain to complete the iterative process and obtain the final solution prices

Performing the integration of a DCM in URM gives rise to an instance. We refer as *Parking instance* the instance generated from the integration of the Parking model into RM and as *ModeCanada instance* the one generated from the integration of ModeCanada model. For both instances, we set a lower and an upper bound on the prices in an experimental way so that the optimal solution do not turn out to be at the lower or upper limits, thus being able to falsify the analysis of the performance of the frameworks. The bounds on the prices are the following: $y_{PSP} \in [0.5, 0.9]$ and $y_{PUP} \in [0.7, 1]$ for the Parking instance; $y_{TRAIN} \in [0, 100]$ and $y_{AIR} \in [0, 100]$ for the ModeCanada instance.

DCM and URM integration results. For the integration with LILA, since the resulting formulation does not depend on the number of individuals, during the integration of both DCMs we considered the entire population (197 individuals for the Parking instance, 2779 individuals for the ModeCanada instance). The initial approximation of the choice probabilities was set to the lower and upper bound of the prices for both instances, and also to a solution in the middle for the ModeCanada instance. The threshold of the termination criteria has been set to 0.0005. For the Parking instance, the average CPU (per iteration) was 0.050s, while for the ModeCanada instance was 0.291s. As we expected, the CPU does not depend on the number of considered individuals.

For the integration with PBGA we have experimented on many parameter settings. In the following we present the most representative ones. Full results are reported in Appendix B).

In charts 1, we report the values of the prices found by LILA iteration by iteration. For both instances, LILA converges to the same local optimum, no matter the initial approximation. This let us conjecture that it may also be the global one. A more refined analysis on the shape of the objective function is provided in Appendix D.1.

In charts 2 we report respectively the value of the *revenue* and Δ *revenue* KPIs iteration by iteration when solving the ModeCanada instance with LILA, with initialization at lower, middle and upper bound of the prices (the behavior on the Parking instance was the same).

Tables 1 and 2 collect results on the Parking instance and ModeCanada instance, respectively. In the upper table we report the KPIs (as indicated in the leading row) of the best solutions found by LILA with different initializations (as indicated in the leading column). In the lower table we report the same KPIs for the execution of PBGA, for different settings of the scenario generation parameters S and R (as indicated in the leading column). The results for PBGA have been obtained as average ones over 15 runs for each setting of S and R (except for the $S = 2000$ $R = 25$ case in the ModeCanada instance, for which 3 runs where performed).

Our analysis of the results yields the following observations.

- In LILA, both solutions quality and robustness improve monotonically iteration by iteration.
- From Figure 2, we can observe how LILA finds better solutions in terms of *revenue* as the number of iterations increases, up to convergence.
- From Figures 1 and 2, we can see that the higher the difference between solutions in two subsequent iterations, the lower the robustness of the solution (high Δ *revenue*); this is consistent with our expectations, being the previous solution the center of the approximation. Δ *revenue* becomes 0 as the algorithm converges.
- LILA clearly outperforms PBGA in both DCMs, being always able to find better solutions in less CPU time. In details, from the tables 1 and 2, we can see that, for both instances, LILA was able to find a better solution in term of *revenue* w.r.t. all solutions obtained with PBGA; furthermore, the time to find such a solution is very low (1.187s on average) compared to PBGA (17658.984 and 352137.393 for the best solutions).

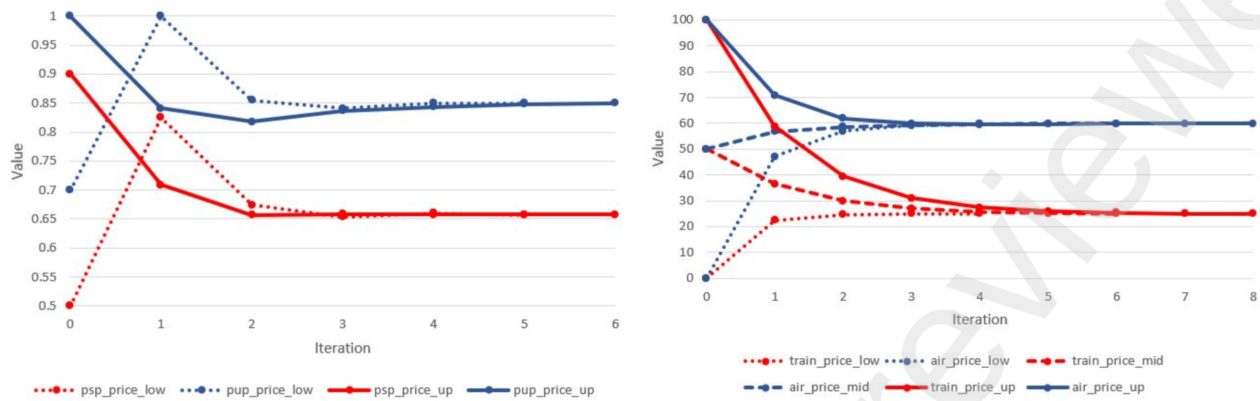

Figure 1: Behavior of the value of the prices found by LILA for the Parking instance (left side) and ModeCanada instance (right side) with the different initializations

LILA				
init. approx.	revenue	Δ revenue	Δ demand	CPU (sec)
(0, 0)	33496.958	0.002%	0.000%	0.453 (4 iter.)
(50, 50)	33496.986	0.001%	0.000%	1.500 (6 iter.)
(100, 100)	33497.144	0.000%	0.000%	3.281 (8 iter.)

PBGA					
S	R	avg. revenue	avg. Δ revenue	avg. Δ demand	CPU (sec)
250	2	33018.658	10.440%	0.288%	7.962
500	2	33079.797	7.706%	0.129%	17.706
1000	2	33269.098	4.000%	0.043%	78.231
50	5	32707.356	18.796%	0.781%	4.337
250	5	33100.090	6.856%	0.100%	33.173
50	10	33163.248	17.212%	0.599%	7.138
250	10	33375.047	3.835%	0.057%	115.550
50	25	33148.652	12.227%	0.378%	38.072
2000	25	33488.722	0.637%	0.002%	352137.393

Table 2: KPI of the best solution found for the ModeCanada instance with LILA and some solutions found with PBGA

4 Case study

Finally, we assess the flexibility of our framework in embedding further modeling features. Inspired by the experiments of [Pan+21], we perform the integration of the ParkingModel DCM in more realistic kinds of RM optimization problems. We consider three different cases: Capacitated RM (CRM), in which a fixed capacity is assumed for each parking option; Population Segmentation

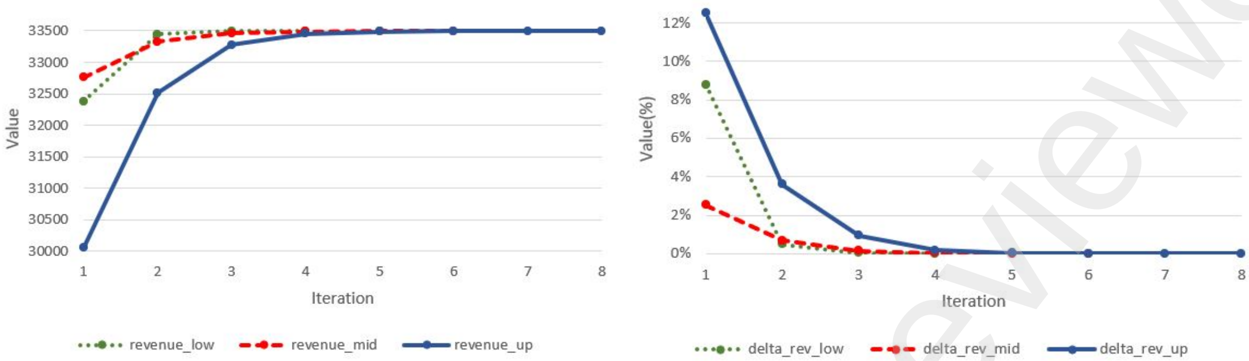

Figure 2: Behavior of the *revenue* (left side) and Δ *revenue* (right side) when solving the ModeCanada instance with initialization at lower, middle and upper bound of the prices

RM (PSRM), in which a different price is offered to different market segments (i.e. Resident and Non-Resident people); Capacity Allocation RM (CARM), where capacities and availability of services are considered as decision variables (with both variable and fixed costs in the objective function of the optimization problem).

In these experiments no change is made to the framework: only the MIP concerning the optimization problem changes.

CRM In Capacitated RM models, services have already been allocated. That is, PSP and PUP have fixed capacity. The objective is still to maximize the revenue. We model the capacity restrictions by adding one constraint in the OM for each alternative, limiting its demand. The optimization model is therefore the following.

$$\text{TCRM(z)} \quad \text{maximize} \sum_{j \in J} y_j \cdot D_j \tag{25}$$

$$\text{s.t. } D_j = \alpha_j(z) \cdot p + q_j(z) \qquad \forall j \in J \tag{26}$$

$$0 \leq D_j \leq c_j \qquad \forall j \in J \tag{27}$$

$$l_j \leq y_j \leq m_j \qquad \forall j \in J \tag{28}$$

We have tested two levels of capacity: $c = (197, 60, 90)$ and $c = (197, 100, 60)$. The bounds on the prices were set to $y_{PSP} \in [0.5, 0.9]$ and $y_{PUP} \in [0.7, 1]$. Two initializations of the choice probability were performed: $z \in \{(0.5, 0.7), (0.9, 1)\}$. The threshold of the termination criteria were set to $thr = 0.00005$. Since the same solution was found with both inizializations, we report only the solution and average the CPU time of the two runs. The solutions found are reported is Table 3. The results of the uncapacitated case are also reported for comparison.

In the case $c = (60, 90)$, the capacity of PSP makes it infeasible the demand of an optimal uncapacitated solution. The PSP price can be increased so that a

LILA				
init. approx.	revenue	Δ revenue	Δ demand	CPU (sec)
(0.5, 0.7)	114.233	0.000%	0.000%	0.328 (5 iter.)
(0.9, 1)	114.233	0.000%	0.000%	0.375 (9 iter.)

PBGA					
S	R	avg. revenue	avg. Δ revenue	avg. Δ demand	CPU (sec)
50	2	111.361	7.182%	1.856%	4.883
100	2	113.143	4.673%	0.868%	15.193
197	2	113.515	3.786%	0.768%	53.594
50	5	113.050	5.440%	1.232%	21.567
100	5	113.383	3.437%	0.748%	62.798
197	25	114.173	0.706%	0.035%	17658.984

Table 1: KPI of the best solution found for the Parking instance with LILA and some solutions found with PBGA

higher revenue from the individuals accessing the service is obtained. The price of PUP is lower, and its demand is in fact higher than in the uncapacitated case. However, the total demand of PSP and PUP is lower. That means that not all individuals leaving PSP are willing to opt for PUP, even if its price is decreased from 0.85 to 0.83. Simmetrically, in the case $c = (100, 60)$, the capacity of PUP makes it infeasible the demand of an optimal uncapacitated solution. However, in this case, almost all individuals leaving PUP opt for PSP. Concerning prices, that of PUP can be increased still reaching the demand limit; that of PSP needs to be kept to 0.66, otherwise the demand decrease would not balance the additional revenue.

The efficiency of our framework allowed also for a more refined plot of the objective function, which is reported in Appendix D.2. That shows that LILA is indeed robust to price initialization also in this case.

Capacities		Prices		Demand			Revenue	CPU (sec)
PSP	PUP	PSP	PUP	FSP	PSP	PUP		
197	197	0.66	0.85	44.9	78.3	73.8	114.233	0.375
60	90	0.67	0.83	48.3	60.0	88.7	113.860	0.078
100	60	0.66	0.89	45.0	92.0	60.0	114.016	0.109

Table 3: Results of the best solution found by solving CRM with LILA. The first row reports the solution of the uncapacitated case.

PSRM The Population Segmentation RM models the real situation in which the population is divided in more segments according to some attributes, still keeping they heterogeneous behavior. In our case, the population was segmented in Resident (R) and Non-Resident (NR) people. Then, a discounted price was offered to Residents based on a rate parameter $d \in \{0.9, 0.8, 0.75, 0.7, 0.6, 0.5\}$ indicating the fraction of the original price that is offered: the higher d, the lower the discount. Two situations were considered: the difference between the actual price is paid by the municipality and contributes to the revenue of the operator (PSRM1), or the difference is paid by the operator (PSRM2). Situation (PSRM2) is clearly more realistic, since in (PSRM1) the operator is allowed to change prices *after* a discount agreement with municipality. The segmentation was modeled by evaluating the DCM two times (one on each segment) giving rise to two demand representation (D^R and D^{NR}). Capacities on the alternatives have also been considered as an optional modeling feature. The corresponding optimization model is the following:

$$\text{PSRM1(z)} \quad \text{maximize} \sum_{j \in J} y_j \cdot D_j^R + y_j \cdot D_j^{NR} \tag{29}$$

$$\text{PSRM2(z)} \quad \text{maximize} \sum_{j \in J} dy_j \cdot D_j^R + y_j \cdot D_j^{NR} \tag{30}$$

$$\text{s.t.} \quad D_j^R = \alpha_j^R(dz) \cdot dy + q_j^R(dz) \qquad \forall j \in J \tag{31}$$

$$D_j^{NR} = \alpha_j^{NR}(z) \cdot y + q_j^{NR}(z) \qquad \forall j \in J \tag{32}$$

$$D_j^R + D_j^{NR} \leq c_j \qquad \forall j \in J \tag{33}$$

$$D_j^R, D_j^{NR} \geq 0 \qquad \forall j \in J \tag{34}$$

$$l_j \leq y_j \leq m_j \qquad \forall j \in J \tag{35}$$

In our experiment the capacities were set to $c = (60, 90)$. The bounds on the prices were set to $y_{PSP} \in [0.5, 0.2]$ and $y_{PUP} \in [0.5, 2]$. The threshold of the termination criteria were set to $thr = 0.00005$. Two initializations of the choice probability were performed: $z \in \{(0.5, 0.5), (2, 2)\}$. As before, since the same solution was always found with both initializations, we report only the solution and average the CPU time of the two runs.

Full results for (PSRM1) are reported in Appendix C. In short, our models reflect intuition: the higher the discount, the higher the offered prices, and thus the lower the Non-Residents demand of PSP and PUP, and the higher the number of Non-Residents deciding to opt-out. Certainly, the Resident demand experiences the opposite: as the discount increases, the offered price is lower than the original one, and therefore more individuals choose PSP and PUP.

The results for (PSRM2) are reported in Table 4. The no discount case $d = 1.0$ is also reported for reference. These results are even more insightful. As for (PSRM1) the larger the discount, the higher the demand of PSP and PUP from residents, the lower that of non residents. Additionally, the discount rate provides the operator a further option to increase the overall revenue, which in our experiment is maximum for $d = 0.8$. That is, our models are able to capture another phenomenon which is expected in practice: by fixing different prices for different segments of the population, more demand can be captured, thus allowing higher revenues to the operator.

d	Prices		Demand R			Demand NR			Revenue	CPU(s)
	PSP	PUP	FSP	PSP	PUP	FSP	PSP	PUP		
1.0	0.67	0.83							113.860	0.078
0.9	0.72	0.88	27.3	31.8	45.9	21.3	28.2	42.5	114.935	1.672
0.8	0.77	0.95	23.1	33.8	48.1	26.0	26.2	39.9	115.297	1.156
0.75	0.8	0.98	20.8	35.0	49.2	28.5	25.0	38.4	115.100	0.844
0.7	0.83	1.02	18.6	36.1	50.3	31.4	23.9	36.8	114.593	1.391
0.6	0.9	1.11	13.7	38.6	52.7	37.4	21.4	33.2	112.422	1.016
0.5	0.99	1.24	9.0	41.2	54.8	44.6	18.8	28.5	108.275	1.078

Table 4: Solving (PSRM2) with LILA.

CARM Finally, we consider a challenging Capacity Allocation RM model, in which (1) the capacities c_j are integer decision variables, yielding a variable operating cost in the objective (in this case, a cost per parking spot) and (2) offering each parking option or not is also a decision variable, yielding a fixed cost in the objective (in this case, a cost for opening the whole parking place). A fixed cost f_j represents the cost for enabling parking option j, while the variable cost v_j is the cost per unit of capacity of option j. Availability of a parking spot (i.e allocating the parking place) is modeled by inserting a binary variable $a_j \in \{0, 1\}\ \forall j \in J$ and the set of constraints $D_j \leq a_j|N|\ \forall j \in J$ imposing the demand of parking j to be 0 when j is not selected ($a_j = 0$). The optimization model is therefore the following:

$$\text{CARM(z)} \quad \text{maximize} \sum_{j \in J} y_j \cdot D_j - v_j c_j - f_j a_j \tag{36}$$

$$\text{s.t.} \quad D_j = \alpha_j(z) \cdot y + q_j(z) \qquad \forall j \in J \tag{37}$$

$$0 \leq D_j \leq c_j \qquad \forall j \in J \tag{38}$$

$$D_j \leq a_j|N| \qquad \forall j \in J \tag{39}$$

$$l_j \leq y_j \leq m_j \qquad \forall j \in J \tag{40}$$

$$a_j \in \{0, 1\} \qquad \forall j \in J \tag{41}$$

$$c_j \in \{0, \ldots, |N|\} \qquad \forall j \in J \tag{42}$$

In our experiment the bounds on the prices were set to $y_{PSP} \in [0.5, 6]$ and $y_{PUP} \in [0.5, 6]$. The fixed and variable cost were set to $f_{PSP} = 6$, $f_{PUP} = 12$, $v_{PSP} = 0.35$ and $v_{PUP} = 0.5$. The initializations of the choice probability were performed for each combination of $y_{PSP} \in \{0.5, 1.5, 2.5\}$ and $y_{PUP} \in \{0.5, 1.5, 2.5\}$. The threshold of the termination criteria were set to $thr = 0.00005$. Indeed, when $a_j = 0$ the optimization model finds it profitable to set also a very high price to alternative j. This yields the demand of alternative j to approach 0 also during the DCM evaluation. We report that these high prices led to a positive coefficient for the FEE attribute in some scenarios when calculating 7, being able to falsify the choice probability for high values of the

Init. Approx.		Capacities		Prices		Demand			Revenue	CPU(s)
PSP	PUP	PSP	PUP	PSP	PUP	FSP	PSP	PUP		
0.5	0.5	63	39	0.88	1.31	95.0	63.0	39.0	47.179	2.844
0.5	1.5	65	37	0.88	1.33	95.0	65.0	37.0	47.178	3.094
0.5	2.5	101	-	0.89	6.0	96.0	101.0	0.0	48.241	0.141
1.5	0.5	-	74	6.0	1.26	123.0	0.0	74.0	44.305	0.078
1.5	1.5	-	73	6.0	1.27	124.0	0.0	73.0	44.300	0.219
1.5	2.5	65	37	0.88	1.33	95.0	65.0	37.0	47.178	2.156
2.5	0.5	-	74	6.0	1.26	123.0	0.0	74.0	44.305	0.172
2.5	1.5	-	73	6.0	1.27	124.0	0.0	73.0	44.300	0.156
2.5	2.5	-	74	6.0	1.26	123.0	0.0	74.0	44.305	0.422

Table 5: Solving CARM with LILA.

prices. Therefore, to avoid numerical issues, the std. dev. of β_{FEE} was changed from 14.2 to 5 in this experiment.

Our results are shown in Table 5. We also report that, differently from previous experiments, the initialization of prices matters: different initialization points lead to different local optima. In Appendix D.3 we report more details on the objective function for the specific CARM case, with the behaviour of each run.

LILA converged to 3 different solutions in terms of capacity: (63,39) with a revenue of 47.179 in which both alternatives have been allocated, (101,0) with revenue 48.241 in which only PSP has been allocated and (0,74) with revenue 44.305 in which only PUP has been allocated. With such fixed and variable costs, a decision maker can state that it is not convenient to allocate PUP.

5 Conclusions

The LILA framework we have introduced tackles the problem of integrating DCMs and optimization models with a different perspective than previous attempts from the literature such as [Pan+21]. In fact, our technique allows to keep the size of optimization models independent from the number of individuals in the DCM. It also removes the need of scenario generation and embedding. This comes at the price of making the method iterative rather than producing a single MIP to be optimized.

To assess the computational effectiveness of our LILA framework we have therefore run experiments on a Revenue Maximization (RM) problem, using data from two real world DCMs. In terms of computing time, in all our tests, our framework outperforms the state-of-the-art method of [Pan+21] by orders of magnitude. Since LILA does not rely on random sampling steps, also the numerical accuracy of the solutions produced is improved.

We have also verified that LILA keeps offering full modeling flexibility: on the RM, it is enough to change the optimization MIP to embed different modeling features, including those discussed in [Pan+21]. In all these variants neither the computational effectiveness nor the solutions quality was affected. The method proved to be also robust in terms of parameter settings.

The additional computational effectiveness of our method, and its independence from the size of the DCM data, make it therefore an appealing tool in decision making. For instance, in our analysis on the RM, higher revenues can be obtained by choosing different prices for different population segments.

Our experiments indicate that the presence of binary decision variables in the optimization models, affecting users' choices either directly or indirectly (as in the CARM), represent a further challenge. Multistart methods seem to be beneficial in this case. As future research steps, we plan to better understand this phenomenon.

Acknowledgements. The authors are grateful to Angel Ibeas and Luigi dell'Olio for sharing their data concerning the city of Santoña. The research has been partially funded by the program Piano Sostegno alla Ricerca (PSR), Università degli Studi di Milano.

References

[Man77] Charles F. Manski. "The Structure of Random Utility Models". In: *Theory and Decision* 8.3 (1977), pp. 229–254.

[LMS98] Martine Labbé, Patrice Marcotte, and Gilles Savard. "A Bilevel Model of Taxation and Its Application to Optimal Highway Pricing". In: *Management Science* 44 (Dec. 1998), pp. 1608–1622.

[MT00] Daniel McFadden and Kenneth Train. "Mixed MNL models for discrete response". In: *Journal of Applied Econometrics* 15.5 (2000), pp. 447–470.

[KSH02] Anton J. Kleywegt, Alexander Shapiro, and Tito Homem-de-Mello. "The Sample Average Approximation Method for Stochastic Discrete Optimization". In: *SIAM Journal on Optimization* 12.2 (2002), pp. 479–502.

[Ryz05] Garrett van Ryzin. "Future of Revenue Management: Models of demand". In: *Journal of Revenue and Pricing Management* 4 (Apr. 2005), pp. 204–210.

[Tra09] Kenneth E. Train. *Discrete Choice Methods with Simulation*. 2nd ed. Cambridge University Press, 2009, pp. 74–75.

[CR11] Roberto Cordone and Francesco Redaelli. "Optimizing the demand captured by a railway system with a regular timetable". In: *Transportation Research Part B: Methodological* 45.2 (2011), pp. 430–446.

[Ibe+14] A. Ibeas, L. dell'Olio, M. Bordagaray, and J. de D. Ortúzar. "Modelling parking choices considering user heterogeneity". In: *Transportation Research Part A: Policy and Practice* 70 (2014), pp. 41–49.

[LNS19] Gilbert Laporte, Stefan Nickel, and Francisco Saldanha-da-Gama. "Introduction to Location Science". In: *Location Science*. Ed. by Gilbert Laporte, Stefan Nickel, and Francisco Saldanha da Gama. Cham: Springer International Publishing, 2019, pp. 1–21.

[Pan+21] Meritxell Pacheco Paneque, Michel Bierlaire, Bernard Gendron, and Shadi Sharif Azadeh. "Integrating advanced discrete choice models in mixed integer linear optimization". In: *Transportation Research Part B: Methodological* 146 (2021), pp. 26–49.

[Cro] Yves Croissant. *mlogit: Multinomial Logit Models*. URL: `https://CRAN.R-project.org/package=mlogit`.

A Details of the DCMs used in the experiments

Tables 6 and 7 report the details of the DCMs used in our experimental evaluation.

	Value	FSP	PSP	PUP
C_{PSP}	32	0	1	0
C_{PUP}	34	0	0	1
β_{AT}	$\sim N(-0.788, 1.06)$	AT_{FSP}	AT_{PSP}	AT_{PUP}
β_{TD}	-0.612	TD_{FSP}	TD_{PSP}	TD_{PUP}
$\beta_{Origin_{FSP}}$	-5.76	$Origin_{FSP}$	0	0
β_{FEE}	$\sim N(-32.3, 14.2)$	0	FEE_{PSP}	FEE_{PUP}
$\beta_{FEE_{PSP(LowInc)}}$	-11	0	$FEE_{PSP}LowInc$	0
$\beta_{FEE_{PSP(Resident)}}$	-11.4	0	$FEE_{PSP}Resident$	0
$\beta_{FEE_{PUP(LowInc)}}$	-13.7	0	0	$FEE_{PUP}LowInc$
$\beta_{FEE_{PUP(Resident)}}$	-10.7	0	0	$FEE_{PUP}Resident$
$\beta_{AgeVeh\leq 3}$	4.04	0	0	$AgeVeh \leq 3$
$cov(\beta_{AT}, \beta_{FEE})$	-12.8	0	0	0

Table 6: Specification of parameters for the Parking model in [Ibe+14]

	Value	TRAIN	AIR	CAR
C_{TRAIN}	2.679	1	0	0
C_{AIR}	1.928	0	1	0
β_{COST}	-0.069	1	1	1
β_{IVT}	$\sim N(-0.014, 0.0128)$	1	1	1
β_{OVT}	-0.054	1	1	1
β_{FREQ}	0.160	1	1	1
β_{INC}	-0.011	1	1	0

Table 7: Specification of parameters for the ModeCanada model

B Integration with PBGA

The analysis of the performance has been made with respect to the number of generated scenarios and the size of the sample of individuals. More specifically: (Parking model) $R \in \{2, 5, 10, 25, 50\}$ and $S \in \{50, 100, 197\}$; (ModeCanada model) $R \in \{2, 5, 10, 25\}$ and $S \in \{50, 150, 500, 1000, 2000\}$. For each subcase, $T = 15$ experiments have been performed, which correspond to independent extractions of the R scenarios and independent extractions of the sample S (for the subcase $R = 25, S = 2000$ we performed only 3 experiments; it took about 98hrs cpu time to solve one experiment).

The aggregated results are shown in tables 8 and 9.

Analysis of the results:

- the robustness of the solution (Δ *revenue* and Δ *demand*) improves both as the number of generated scenarios increases and the size of the sample increases (excepting for the outlier $S = 500, R = 25; S = 1000, R = 25$, this may be due to the randomness of the sample and the small number of repetitions $T = 15$)
- the quality of the solution (*revenue*) strictly improves both as the number of scenarios increases and the number of individuals increases (excepting for the outliers $S = 1000, R = 5; S = 1500, R = 5$ and $S = 250, R = 10; S = 500, R = 10$)
- the *CPU* increases exponentially both as the number of scenarios increases and the number of individuals increases (from about 389s with R = 2 and S = 2000 to more than 352137s for R = 25 and S = 2000 and from 38s with R = 25 and S = 50 to more than 352137s for R = 25 and S = 2000)

S	R	avg. revenue	avg. Δ revenue	avg. Δ demand	CPU (sec)
50	2	111.361	7.182%	1.856%	4.883
100	2	113.143	4.673%	0.868%	15.193
197	2	113.515	3.786%	0.768%	53.594
50	5	113.050	5.440%	1.232%	21.567
100	5	113.383	3.437%	0.748%	62.798
197	5	113.815	1.541%	0.183%	502.459
50	10	113.706	3.086%	0.595%	93.259
100	10	113.829	2.168%	0.371%	519.468
197	10	114.013	1.389%	0.098%	2050.942
50	25	113.844	2.513%	0.415%	867.300
100	25	114.056	1.679%	0.176%	3318.354
197	25	114.173	0.706%	0.035%	17658.984
50	50	113.937	1.513%	0.213%	2887.983
100	50	114.044	1.287%	0.161%	21840.202
197	50	114.143	0.574%	0.031%	135476.953

Table 8: KPI for the integration of the Parking Model with PBGA

S	R	avg. revenue	avg. Δ revenue	avg. Δ demand	CPU (sec)
50	2	31947.526	20.778%	0.656%	2.243
250	2	33018.658	10.440%	0.288%	7.962
500	2	33079.797	7.706%	0.129%	17.706
1000	2	33269.098	4.000%	0.043%	78.231
1500	2	33334.369	3.572%	0.035%	214.625
2000	2	33342.937	2.832%	0.022%	389.125
50	5	32707.356	18.796%	0.781%	4.337
250	5	33100.090	6.856%	0.100%	33.173
500	5	33111.166	6.606%	0.088%	126.337
1000	5	33393.258	2.676%	0.020%	493.978
1500	5	33339.252	2.019%	0.012%	2547.449
2000	5	33433.638	1.643%	0.008%	10781.924
50	10	33163.248	17.212%	0.599%	7.138
250	10	33375.047	3.835%	0.057%	115.550
500	10	33334.233	3.752%	0.041%	519.328
1000	10	33405.735	2.726%	0.018%	13879.282
1500	10	33434.568	1.434%	0.008%	16836.819
2000	10	33453.838	1.204%	0.005%	29002.600
50	25	33148.652	12.227%	0.378%	38.072
250	25	33392.999	6.478%	0.100%	1076.866
500	25	33412.667	3.081%	0.020%	9243.808
1000	25	33457.851	3.326%	0.025%	53418.374
1500	25	33469.849	1.591%	0.007%	147084.186
2000	25	33488.722	0.637%	0.002%	352137.393

Table 9: KPI for the integration of the ModeCanada model with PBGA

C Case study detailed results

In Table 10 we report the detailed results of experiment (PSRM1).

d	Prices		Demand R			Demand NR			Revenue	CPU (sec)
	PSP	PUP	FSP	PSP	PUP	FSP	PSP	PUP		
0.9	0.72	0.88	27.1	31.8	46.2	21.1	28.2	42.7	121.28	1.75
0.8	0.77	0.95	23.1	33.8	48.1	26.0	26.2	39.9	129.625	2.234
0.75	0.8	0.99	21.2	35.0	48.8	28.9	25.0	38.1	134.195	1.406
0.7	0.84	1.04	19.4	36.2	49.3	32.2	23.8	36.0	139.086	2.156
0.6	0.94	1.17	16.2	39.1	49.7	40.4	20.9	30.7	150.112	1.375
0.5	1.08	1.37	13.6	42.7	48.7	50.9	17.3	23.8	163.840	1.719

Table 10: Solving (PSRM1) with LILA.

D Analysis on the objective functions

D.1 URM

In Figure 3 we plot the objective function of the Parking URM (blue grid). The steps performed by two runs of LILA, using very different price initialization, are depicted in blue and red: they indeed converge to the same global optimal solution.

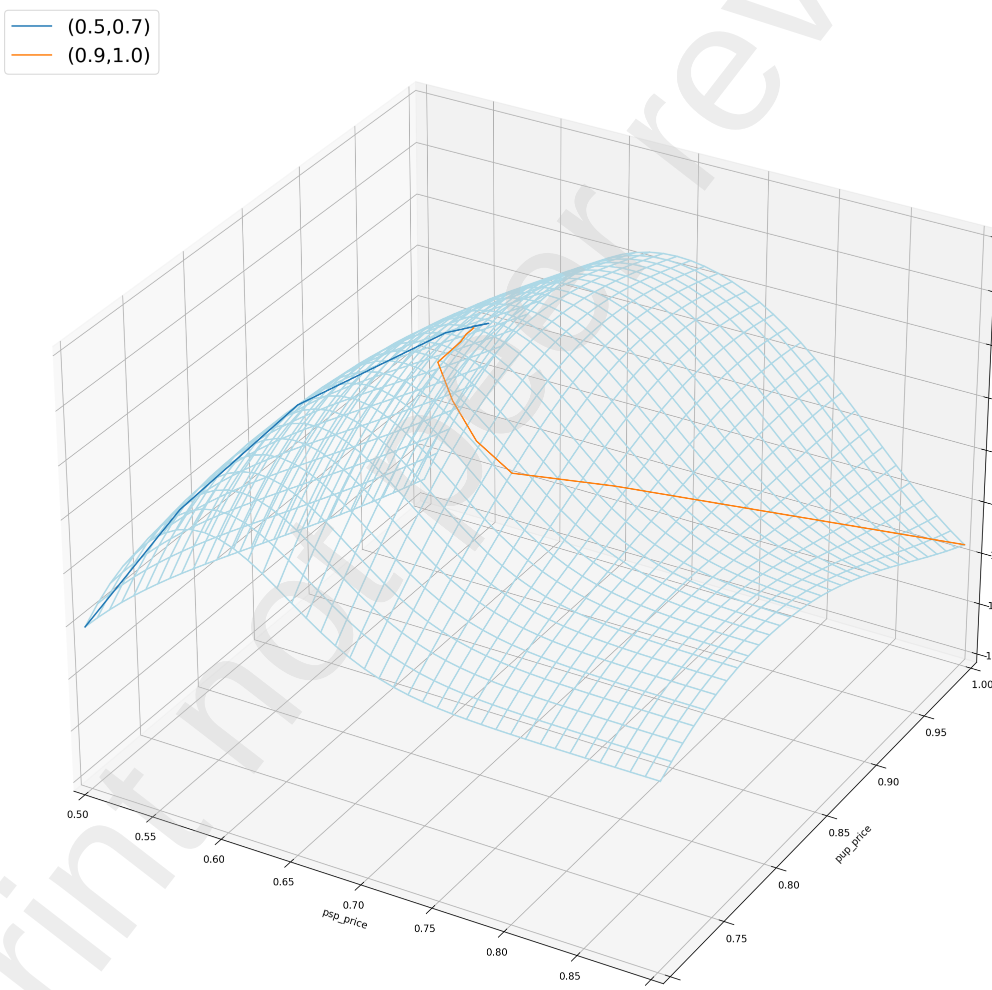

Figure 3: Objective function of the Parking URM with behaviour of LILA

D.2 CRM

Figure 4 and 5 show an analysis of the objective function for the two levels of capacity (60,90) and (100,60). Green points represent feasible points, that is, prices for which the demand is feasible w.r.t the capacity. We can observe that LILA, during the execution, finds unfeasible solutions but ultimately converges to the feasible and optimal one in both cases.

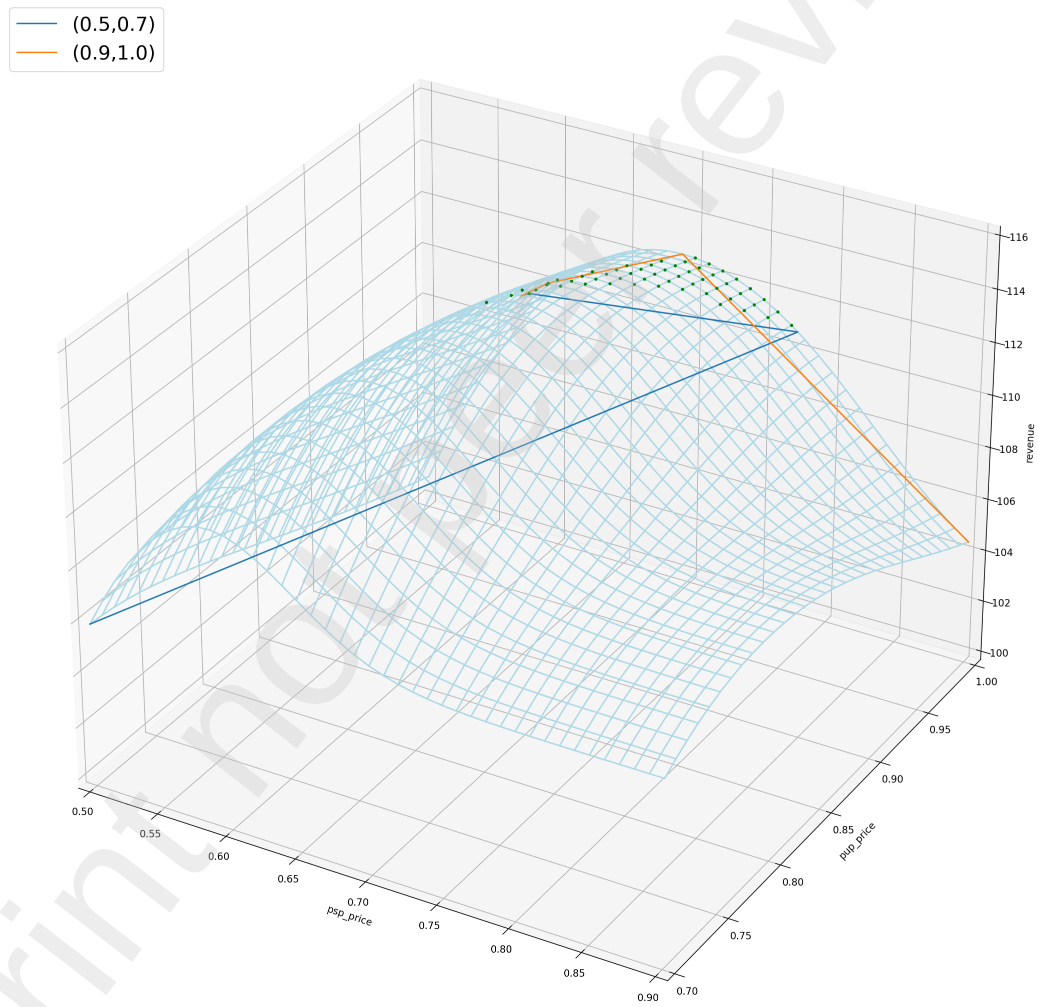

Figure 4: Objective function of the Parking CRM with capacities (60,90) and behaviour of LILA

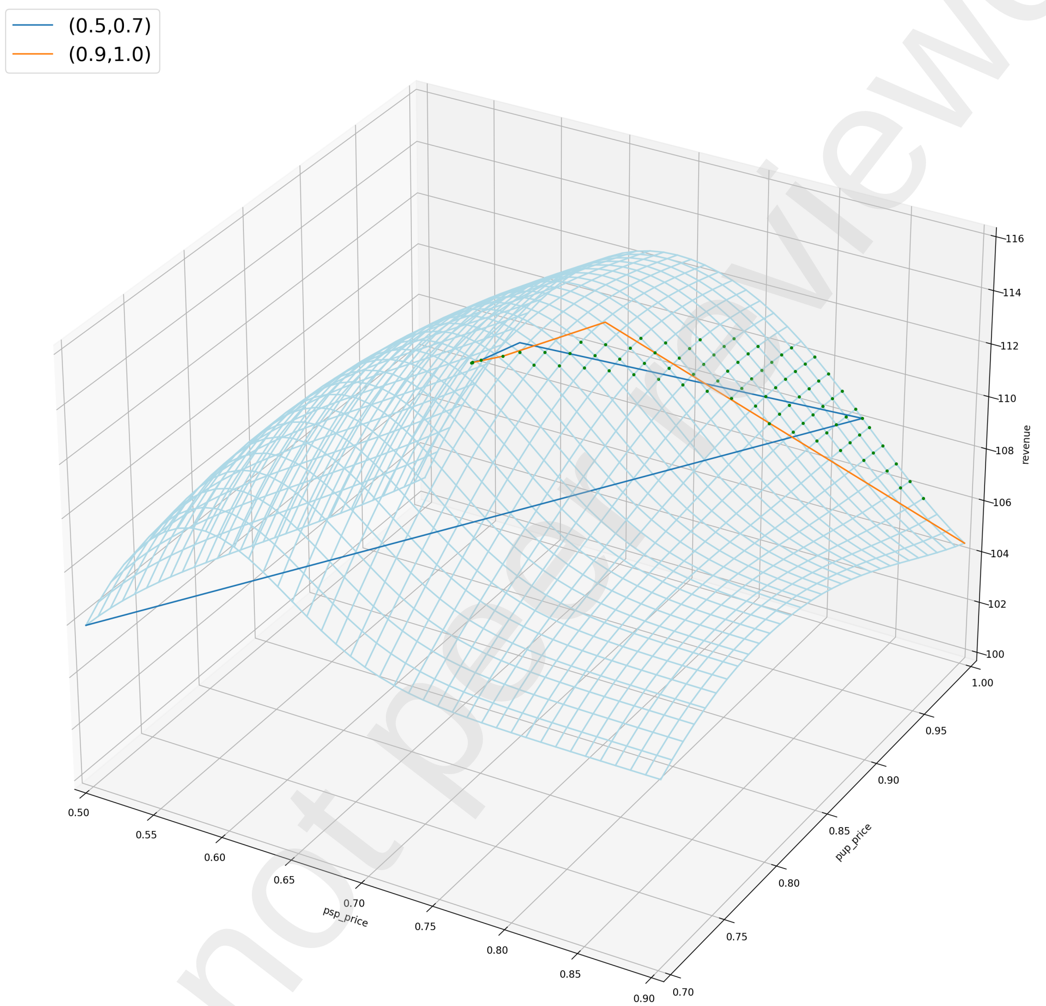

Figure 5: Objective function of the Parking CRM with capacities (100,60) and behaviour of LILA

D.3 CARM

Figure 6 shows the objective function of CARM with different LILA runs. We can observe that the shape of the objective is more complicated than before and has more than one local optimum. Nevertheless, LILA is still able to converge and, by using a multi-start approach, it was able to find the global one, that is, $c = (101, 0)$ and $p = (0.89, 6)$.

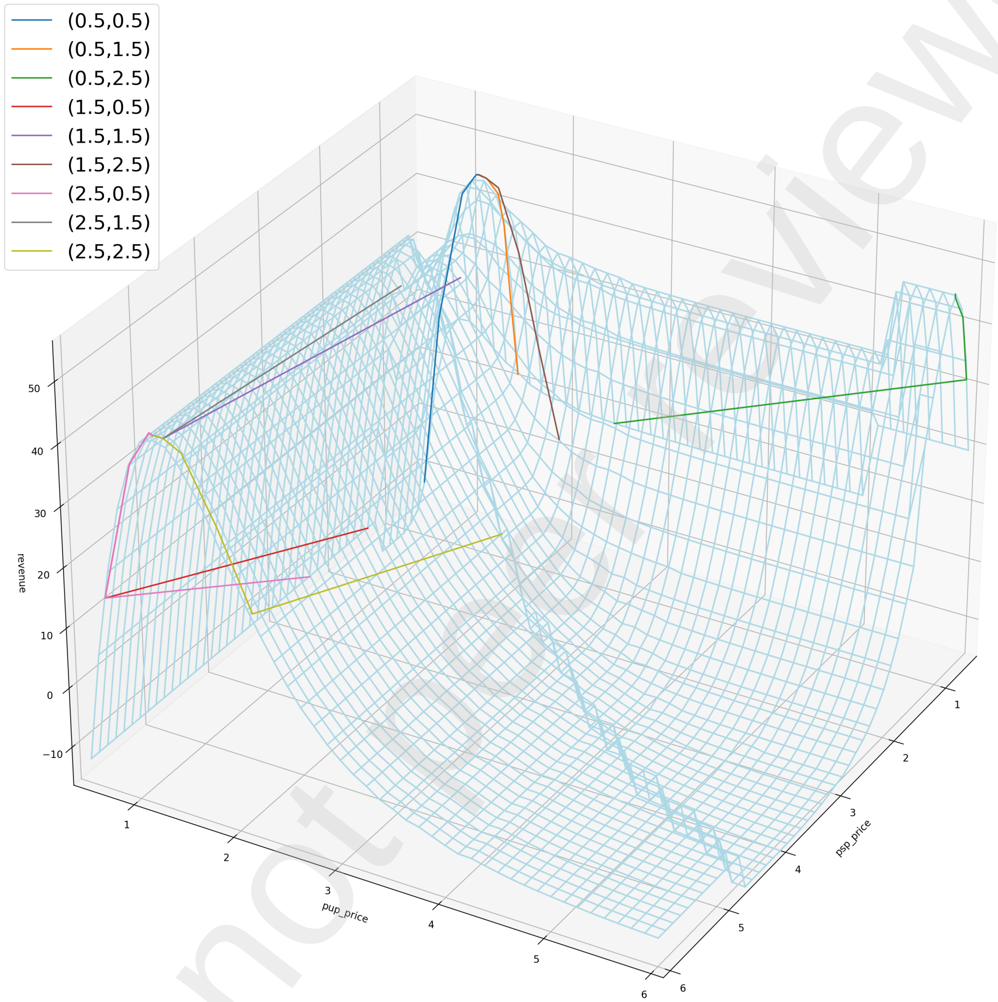

Figure 6: Objective function of the Parking URM with behaviour of LILA