A Lagrangian decomposition scheme for the choice-based optimization framework

Meritxell Pacheco Paneque

OR seminar - Erasmus University Rotterdam

June 26th, 2020
Outline

1. Introduction
2. Choice-based optimization framework
3. Decomposition techniques: preliminaries
4. Lagrangian decomposition scheme
5. Numerical experiments
6. Conclusions and future work
Outline

1. Introduction
2. Choice-based optimization framework
3. Decomposition techniques: preliminaries
4. Lagrangian decomposition scheme
5. Numerical experiments
6. Conclusions and future work
Mismatch between supply and demand

- Reduced profitability
- Decrease in consumer goodwill
- Spillover effects
Mismatch between supply and demand

- reduced profitability
Mismatch between supply and demand

- reduced profitability
- decrease in consumer goodwill
Mismatch between supply and demand

- reduced profitability
- decrease in consumer goodwill
- spillover effects
Supply-demand interplay

Take into account the interactions between supply and demand
Demand model

- Behavioral realism
- Disaggregate representation
Demand model

- Behavioral realism
- Disaggregate representation

Discrete choice models (DCM)
Demand model

- Behavioral realism
- Disaggregate representation

Discrete choice models (DCM)

- Causality between explanatory variables and choice (random utility)
Demand model

- Behavioral realism
- Disaggregate representation

Discrete choice models (DCM)
- Causality between explanatory variables and choice (random utility)
- Probabilistic
Demand model

- Behavioral realism
- Disaggregate representation

Discrete choice models (DCM)

- Causality between explanatory variables and choice (random utility)
- Probabilistic
- Heterogeneity of tastes and preferences in high detail
Supply-related decisions

- Optimization models
- Tractability
Supply-related decisions

- Optimization models
- Tractability

Mixed Integer Linear Problems (MILP)
Supply-related decisions

- Optimization models
- Tractability

Mixed Integer Linear Problems (MILP)
- Modeling flexibility (integer and continuous variables)
Supply-related decisions

- Optimization models
- Tractability

Mixed Integer Linear Problems (MILP)
- Modeling flexibility (integer and continuous variables)
- Commercial MILP solvers to find the global optima
Supply-related decisions

- Optimization models
- Tractability

Mixed Integer Linear Problems (MILP)
- Modeling flexibility (integer and continuous variables)
- Commercial MILP solvers to find the global optima
- Variety of strategies and solution techniques
DCM and MILP: an illustrative example

Simple DCM and 2 groups in the population

A service offered by an operator and an opt-out option

Supply-related decision: price (also in the DCM)

Revenue maximization (revenue = expected demand \cdot price)

<table>
<thead>
<tr>
<th></th>
<th>Group 1</th>
<th>Group 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total</td>
<td>0.2</td>
<td>0.4</td>
</tr>
<tr>
<td></td>
<td>0.6</td>
<td>0.8</td>
</tr>
<tr>
<td></td>
<td>1.2</td>
<td>1.4</td>
</tr>
<tr>
<td></td>
<td>1.6</td>
<td>1.8</td>
</tr>
<tr>
<td></td>
<td>2.0</td>
<td>2.2</td>
</tr>
</tbody>
</table>

Price

Expected revenue
DCM and MILP: an illustrative example

- Simple DCM and 2 groups in the population

<table>
<thead>
<tr>
<th></th>
<th>Group 1</th>
<th>Group 2</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0.2</td>
<td>0</td>
<td>0.2</td>
</tr>
<tr>
<td></td>
<td>0.4</td>
<td>0</td>
<td>0.4</td>
</tr>
<tr>
<td></td>
<td>0.6</td>
<td>0</td>
<td>0.6</td>
</tr>
<tr>
<td></td>
<td>0.8</td>
<td>0</td>
<td>0.8</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>

Price

Expected revenue
DCM and MILP: an illustrative example

- Simple DCM and 2 groups in the population
- A service offered by an operator and an opt-out option
DCM and MILP: an illustrative example

- Simple DCM and 2 groups in the population
- A service offered by an operator and an opt-out option
- Supply-related decision: price (also in the DCM)
DCM and MILP: an illustrative example

- Simple DCM and 2 groups in the population
- A service offered by an operator and an opt-out option
- Supply-related decision: price (also in the DCM)
- Revenue maximization (revenue = expected demand \cdot price)
DCM and MILP: an illustrative example

- Simple DCM and 2 groups in the population
- A service offered by an operator and an opt-out option
- Supply-related decision: price (also in the DCM)
- Revenue maximization (revenue = expected demand · price)
- Expected demand obtained from the individual choice probabilities

<table>
<thead>
<tr>
<th>Price</th>
<th>Expected revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
DCM and MILP: an illustrative example

- Simple DCM and 2 groups in the population
- A service offered by an operator and an opt-out option
- Supply-related decision: price (also in the DCM)
- Revenue maximization \( (\text{revenue} = \text{expected demand} \cdot \text{price}) \)
- Expected demand obtained from the individual choice probabilities

<table>
<thead>
<tr>
<th>Price</th>
<th>Group 1</th>
<th>Group 2</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0.2</td>
<td>1.6</td>
<td>0.1</td>
<td>1.7</td>
</tr>
<tr>
<td>0.4</td>
<td>1.0</td>
<td>0.6</td>
<td>1.6</td>
</tr>
<tr>
<td>0.6</td>
<td>0.8</td>
<td>0.4</td>
<td>1.2</td>
</tr>
<tr>
<td>0.8</td>
<td>0.5</td>
<td>0.3</td>
<td>0.8</td>
</tr>
<tr>
<td>1</td>
<td>0.2</td>
<td>0.1</td>
<td>0.3</td>
</tr>
<tr>
<td>1.2</td>
<td>0.1</td>
<td>0.05</td>
<td>0.15</td>
</tr>
<tr>
<td>1.4</td>
<td>0.05</td>
<td>0.025</td>
<td>0.075</td>
</tr>
<tr>
<td>1.6</td>
<td>0.025</td>
<td>0.0125</td>
<td>0.0375</td>
</tr>
<tr>
<td>1.8</td>
<td>0.0125</td>
<td>0.00625</td>
<td>0.01875</td>
</tr>
<tr>
<td>2</td>
<td>0.00625</td>
<td>0.003125</td>
<td>0.009375</td>
</tr>
</tbody>
</table>

Expected revenue vs. Price

Lagrangian decomposition scheme

M. Pacheco, TRANSP-OR, EPFL
June 26th, 2020
Choice-based optimization framework

- General framework that accommodates DCM in MILP
Choice-based optimization framework

- General framework that accommodates DCM in MILP
- Decision variables of MILP as explanatory variables of DCM
Choice-based optimization framework

- General framework that accommodates DCM in MILP
- Decision variables of MILP as explanatory variables of DCM
- Simulation-based linearization of the preference structure of DCM
Outline

1. Introduction
2. Choice-based optimization framework
3. Decomposition techniques: preliminaries
4. Lagrangian decomposition scheme
5. Numerical experiments
6. Conclusions and future work
Choice-based optimization framework

DCM

- Population $N(n)$ and set of alternatives $C(i)$

Utility associated with alternative $i$ and individual $n$: $U_{in} = V_{in}\text{systematic} + \epsilon_{in}\text{random}$

$V_{in}$: modeled by the analyst (attributes, socioeconomic information)

$\epsilon_{in}$: follows a probability distribution (e.g., Gumbel, normal)

Behavioral assumption: alternative with the highest utility is chosen

Choice probability: $P_{n}(i) = P(U_{in} \geq U_{jn}, \forall j \in C)$

Expected demand: $D_i = \sum_{n} P_{n}(i)$
Choice-based optimization framework

DCM

- Population $N$ ($n$) and set of alternatives $C$ ($i$)
- Utility associated with alternative $i$ and individual $n$ ($U_{in}$):

$$U_{in} = V_{in} + \varepsilon_{in}$$

- $V_{in}$: modeled by the analyst (attributes, socioeconomic information), e.g.,
  $$V_{in} = ASC_i + \beta_{c} \text{cost} + \beta_{t} \text{time} + \beta_{i} \text{income}$$
- $\varepsilon_{in}$: follows a probability distribution (e.g., Gumbel, normal)
- Behavioral assumption: alternative with the highest utility is chosen
- Choice probability:
  $$P_n(i) = P(U_{in} \geq U_{jn}, \forall j \in C)$$
- Expected demand:
  $$D_i = \sum_n P_n(i)$$
DCM

- Population $N(n)$ and set of alternatives $C(i)$
- Utility associated with alternative $i$ and individual $n$ ($U_{in}$):
  \[ U_{in} = V_{in} + \varepsilon_{in} \]
  - Systematic
  - Random
- $V_{in}$: modeled by the analyst (attributes, socioeconomic information)
  \[ V_{in} = ASC_i + \beta_{\text{cost}} \text{cost}_{in} + \beta_{\text{time}} \text{time}_{in} + \beta_{\text{income}} \text{income}_n \]
Choice-based optimization framework

DCM

- Population $N(n)$ and set of alternatives $C(i)$
- Utility associated with alternative $i$ and individual $n$ ($U_{in}$):
  
  $$U_{in} = V_{in}^{\text{systematic}} + \varepsilon_{in}^{\text{random}}$$

- $V_{in}$: modeled by the analyst (attributes, socioeconomic information)
  
  e.g., $V_{in} = ASC_i + \beta_{\text{cost}} \text{cost}_{in} + \beta_{\text{time}} \text{time}_{in} + \beta_{\text{income}} \text{income}_n$

- $\varepsilon_{in}$: follows a probability distribution (e.g., Gumbel, normal)
DCM

- Population \( N (n) \) and set of alternatives \( C (i) \)
- Utility associated with alternative \( i \) and individual \( n \) (\( U_{in} \)):
  \[
  U_{in} = V_{in} + \varepsilon_{in}
  \]
    \[
    \text{systematic} + \text{random}
  \]
- \( V_{in} \): modeled by the analyst (attributes, socioeconomic information)
  e.g., \( V_{in} = \text{ASC}_i + \beta_{\text{cost}} \text{cost}_{in} + \beta_{\text{time}} \text{time}_{in} + \beta_{\text{income}} \text{income}_n \)
- \( \varepsilon_{in} \): follows a probability distribution (e.g., Gumbel, normal)
- Behavioral assumption: alternative with the highest utility is chosen
DCM

- Population $N_n$ and set of alternatives $C_i$
- Utility associated with alternative $i$ and individual $n$ ($U_{in}$):
  \[ U_{in} = V_{in} + \varepsilon_{in} \]
    \[
    \begin{aligned}
    &\text{systematic} \quad \text{random} \\
    \end{aligned}
  \]
- $V_{in}$: modeled by the analyst (attributes, socioeconomic information)
  \[ e.g., V_{in} = ASC_i + \beta_{cost} \text{cost}_{in} + \beta_{time} \text{time}_{in} + \beta_{income} \text{income}_n \]
- $\varepsilon_{in}$: follows a probability distribution (e.g., Gumbel, normal)
- Behavioral assumption: alternative with the highest utility is chosen
- Choice probability: $P_n(i) = P(U_{in} \geq U_{jn}, \forall j \in C)$
DCM

- Population $N(n)$ and set of alternatives $\mathcal{C}(i)$
- Utility associated with alternative $i$ and individual $n$ ($U_{in}$):

$$U_{in} = V_{in} + \varepsilon_{in}$$

  - Systematic: modeled by the analyst (attributes, socioeconomic information)
    - e.g., $V_{in} = ASC_i + \beta_{cost}\text{cost}_{in} + \beta_{time}\text{time}_{in} + \beta_{income}\text{income}_{n}$

  - Random: follows a probability distribution (e.g., Gumbel, normal)

- Behavioral assumption: alternative with the highest utility is chosen
- Choice probability: $P_n(i) = P(U_{in} \geq U_{jn}, \forall j \in \mathcal{C})$
- Expected demand: $D_i = \sum_n P_n(i)$
Simulation-based linearization

- Simulation to overcome the probabilistic nature of the utility ($\varepsilon_{in}$)
Simulation-based linearization

- Simulation to overcome the probabilistic nature of the utility ($\varepsilon_{in}$)
- $R$ draws ($r$) from the distribution of $\varepsilon_{in}$ ($\xi_{inr}$)
Simulation-based linearization

- Simulation to overcome the probabilistic nature of the utility \( (\varepsilon_{in}) \)
- \( R \) draws \( (r) \) from the distribution of \( \varepsilon_{in} (\xi_{inr}) \)

\[
U_{inr} = V_{in} + \xi_{inr}
\]
Simulation-based linearization

- Simulation to overcome the probabilistic nature of the utility ($\varepsilon_{in}$)
- $R$ draws ($r$) from the distribution of $\varepsilon_{in}$ ($\xi_{inr}$)

$$U_{inr} = V_{in} + \xi_{inr}$$

- $U_{inr}$ are deterministic expressions (can be included in a MILP)
Simulation-based linearization

- Simulation to overcome the probabilistic nature of the utility ($\varepsilon_{in}$)
- $R$ draws ($r$) from the distribution of $\varepsilon_{in}$ ($\xi_{inr}$)

$$U_{inr} = V_{in} + \xi_{inr}$$

- $U_{inr}$ are deterministic expressions (can be included in a MILP)
- Explanatory variables of $V_{in}$:
Simulation-based linearization

- Simulation to overcome the probabilistic nature of the utility ($\varepsilon_{in}$)
- $R$ draws ($r$) from the distribution of $\varepsilon_{in}$ ($\xi_{inr}$)

$$U_{inr} = V_{in} + \xi_{inr}$$

- $U_{inr}$ are deterministic expressions (can be included in a MILP)
- Explanatory variables of $V_{in}$:
  - Exogenous to the optimization problem: $x_{in}^d$ (e.g., income)
  - Endogenous to the optimization problem: $x_{in}^e$ (e.g., cost)
Simulation-based linearization

- Simulation to overcome the probabilistic nature of the utility ($\varepsilon_{in}$)
- $R$ draws ($r$) from the distribution of $\varepsilon_{in}$ ($\xi_{inr}$)

$$U_{inr} = V_{in} + \xi_{inr}$$

- $U_{inr}$ are deterministic expressions (can be included in a MILP)
- Explanatory variables of $V_{in}$:
  - Exogenous to the optimization problem: $x_{in}^d$ (e.g., income)
  - Endogenous to the optimization problem: $x_{in}^e$ (e.g., cost)
- Integration in MILP: $V_{in}(x_{in}^d, x_{in}^e)$ linear in $x^e$
Mixed-integer linear formulation

- Capacity associated with each alternative: $c_i$
Mixed-integer linear formulation

- Capacity associated with each alternative: \( c_i \)
- **Availability:** to propose an alternative \((y_{in})\) and to keep track of the occupancy of the alternatives \((y_{inr})\)
Mixed-integer linear formulation

- Capacity associated with each alternative: $c_i$
- **Availability:** to propose an alternative ($y_{in}$) and to keep track of the occupancy of the alternatives ($y_{inr}$)
- **Capacity allocation:** controlled with the variables $y_{inr}$ with an exogenous priority list (Binder et al., 2017)
Mixed-integer linear formulation

- Capacity associated with each alternative: $c_i$
- **Availability**: to propose an alternative ($y_{in}$) and to keep track of the occupancy of the alternatives ($y_{inr}$)
- **Capacity allocation**: controlled with the variables $y_{inr}$ with an exogenous priority list (Binder et al., 2017)
- **Discounted utility**: unavailable alternative cannot be associated with the largest $U_{inr}$ ($z_{inr}$)
Choice-based optimization framework

Mixed-integer linear formulation

- Capacity associated with each alternative: $c_i$
- **Availability**: to propose an alternative ($y_{in}$) and to keep track of the occupancy of the alternatives ($y_{inr}$)
- **Capacity allocation**: controlled with the variables $y_{inr}$ with an exogenous priority list (Binder et al., 2017)
- **Discounted utility**: unavailable alternative cannot be associated with the largest $U_{inr}$ ($z_{inr}$)
- **Choice**: only one alternative can be chosen for each $n$ and $r$

$$w_{inr} = \begin{cases} 1 & \text{if } z_{inr} = U_{nr} := \max_{j \in C} z_{jnr} \\ 0 & \text{otherwise} \end{cases}$$
Expected demand

- \( \{ w_{inr} \}_r \) count number of times the behavioral assumption is met
Expected demand

- $\{w_{inr}\}_r$ count number of times the behavioral assumption is met
- Law of large numbers: $\frac{1}{R} \sum_r w_{inr} \xrightarrow{R \to \infty} P_n(i|x_{in}^d, x_{in}^e)$
Choice-based optimization framework

Expected demand

- \{w_{inr}\}_r count number of times the behavioral assumption is met
- Law of large numbers: \( \frac{1}{R} \sum_r w_{inr} \xrightarrow{R \to \infty} P_n(i|x_{in}^d, x_{in}^e) \)

\[
D_i \approx \frac{1}{R} \sum_{r=1}^R \sum_{n=1}^N w_{inr}
\]
Expected demand

- \( \{ w_{inr} \}_r \) count number of times the behavioral assumption is met
- Law of large numbers: \( \frac{1}{R} \sum_r w_{inr} \xrightarrow{R \to \infty} P_n(i|x_{in}^d, x_{in}^e) \)

\[
D_i \approx \frac{1}{R} \sum_{r=1}^{R} \sum_{n=1}^{N} w_{inr}
\]

- Original problem (\( \mathcal{P} \)): demand via choice probabilities
Choice-based optimization framework

Expected demand

- \{w_{inr}\}_r \text{ count number of times the behavioral assumption is met}
- Law of large numbers: \( \frac{1}{R} \sum_r w_{inr} \xrightarrow[R\to\infty]{} P_n(i|x_{in}^d, x_{in}^e) \)

\[
D_i \approx \frac{1}{R} \sum_{r=1}^{R} \sum_{n=1}^{N} w_{inr}
\]

- Original problem (\(P\)): demand via choice probabilities
- Approximated problems (\(P_R\)): demand via linear approximation
Expected demand

- \( \{w_{inr}\}_r \) count number of times the behavioral assumption is met
- Law of large numbers: \( \frac{1}{R} \sum_r w_{inr} \xrightarrow{R \to \infty} P_n(i|x_{in}^d, x_{in}^e) \)

\[
D_i \approx \frac{1}{R} \sum_{r=1}^{R} \sum_{n=1}^{N} w_{inr}
\]

- Original problem \((P)\): demand via choice probabilities
- Approximated problems \((P_R)\): demand via linear approximation
- Sequence of optimal sols. of \(P_R\) converges to an optimal sol. of \(P\)
Profit maximization

- Mixed-integer linear formulation can be embedded in any MILP
Profit maximization

- Mixed-integer linear formulation can be embedded in any MILP
- **Illustration:** profit maximization problem
Profit maximization

- Mixed-integer linear formulation can be embedded in any MILP
- **Illustration:** profit maximization problem
  - Operator proposing services + opt-out option
Profit maximization

- Mixed-integer linear formulation can be embedded in any MILP
- **Illustration:** profit maximization problem
  - Operator proposing services + opt-out option
  - Price (and capacity) to be decided
Profit maximization

- Mixed-integer linear formulation can be embedded in any MILP
- **Illustration:** profit maximization problem
  - Operator proposing services + opt-out option
  - Price (and capacity) to be decided

<table>
<thead>
<tr>
<th>obj. fun.</th>
<th>max profit from all services but the opt-out</th>
</tr>
</thead>
<tbody>
<tr>
<td>DCM</td>
<td>availability, discounted utility, choice</td>
</tr>
<tr>
<td>capacity</td>
<td>fixed or variable (discretized)</td>
</tr>
<tr>
<td>price</td>
<td>$p_{in}$ endogenous (continuous or discrete)</td>
</tr>
</tbody>
</table>
Parking case study

- DCM estimated in Ibeas et al. [2014]: non-closed form choice probs.
Choice-based optimization framework

Parking case study

- DCM estimated in Ibeas et al. [2014]: non-closed form choice probs.
- $C$: PSP (paid on-street), PUP (underground), FSP (opt-out)
Parking case study

- DCM estimated in Ibeas et al. [2014]: non-closed form choice probs.
- $C$: PSP (paid on-street), PUP (underground), FSP (opt-out)
- $N = 50$ (random priority list)
Parking case study

- DCM estimated in Ibeas et al. [2014]: non-closed form choice probs.
- $C$: PSP (paid on-street), PUP (underground), FSP (opt-out)
- $N = 50$ (random priority list)
- Common price (same price proposed to everyone): $p_i$
Computational complexity

- Flexibility of the framework: price segmentation, aggregation of individuals with similar characteristics, capacity allocation strategies
Choice-based optimization framework

Computational complexity

- Flexibility of the framework: price segmentation, aggregation of individuals with similar characteristics, capacity allocation strategies
- Revenue maximization problem (without and with fixed capacity)
Computational complexity

- Flexibility of the framework: price segmentation, aggregation of individuals with similar characteristics, capacity allocation strategies
- Revenue maximization problem (without and with fixed capacity)
  - Without capacity constraints: 1.75h for $R = 250$ draws
  - With capacity constraints: 21h for $R = 250$ draws
- Profit maximization problem
  - Parking facilities might not be open: 9.8h for $R = 25$ draws
  - All parking facilities must be open: 11.5h for $R = 25$ draws

Exploit the decomposable structure of the framework!
Computational complexity

- Flexibility of the framework: price segmentation, aggregation of individuals with similar characteristics, capacity allocation strategies
- Revenue maximization problem (without and with fixed capacity)
  - Without capacity constraints: 1.75h for $R = 250$ draws
  - With capacity constraints: 21h for $R = 250$ draws
- Profit maximization problem
Computational complexity

- Flexibility of the framework: price segmentation, aggregation of individuals with similar characteristics, capacity allocation strategies
- Revenue maximization problem (without and with fixed capacity)
  - Without capacity constraints: 1.75h for $R = 250$ draws
  - With capacity constraints: 21h for $R = 250$ draws
- Profit maximization problem
  - Parking facilities might not be open: 9.8h for $R = 25$ draws
  - All parking facilities must be open: 11.5h for $R = 25$ draws
Computational complexity

- Flexibility of the framework: price segmentation, aggregation of individuals with similar characteristics, capacity allocation strategies
- Revenue maximization problem (without and with fixed capacity)
  - Without capacity constraints: 1.75h for $R = 250$ draws
  - With capacity constraints: 21h for $R = 250$ draws
- Profit maximization problem
  - Parking facilities might not be open: 9.8h for $R = 25$ draws
  - All parking facilities must be open: 11.5h for $R = 25$ draws
- Exploit the decomposable structure of the framework!
Outline

1. Introduction
2. Choice-based optimization framework
3. Decomposition techniques: preliminaries
4. Lagrangian decomposition scheme
5. Numerical experiments
6. Conclusions and future work
Motivation

- Disaggregate demand
- Simulation-based linearization

\{ high computational complexity \}
Motivation

- Disaggregate demand
- Simulation-based linearization

\{ \text{high computational complexity} \}

- In practice, large populations and/or considerable number of draws
Decomposition techniques: preliminaries

Motivation

- Disaggregate demand
- Simulation-based linearization

\{ high computational complexity \\

- In practice, large populations and/or considerable number of draws
- Framework built on two dimensions that can be addressed separately:
Motivation

- Disaggregate demand
- Simulation-based linearization

}$\begin{equation*}
\text{In practice, large populations and/or considerable number of draws}
\end{equation*}$

- Framework built on two dimensions that can be addressed separately:
  - **Individuals:** most fundamental unit of demand

}$\begin{equation*}$
\text{high computational complexity}$
\end{equation*}$
Motivation

- Disaggregate demand
- Simulation-based linearization

\{ high computational complexity \}

- In practice, large populations and/or considerable number of draws
- Framework built on two dimensions that can be addressed separately:
  - **Individuals**: most fundamental unit of demand
  - **Draws**: independent behavioral scenario
### Revenue maximization problem

<table>
<thead>
<tr>
<th><strong>obj. fun. ( (Z) )</strong></th>
<th>max revenue from all services but the opt-out</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>DCM</strong></td>
<td>availability, discounted utility, choice</td>
</tr>
<tr>
<td><strong>price</strong></td>
<td>( p_{in} ) continuous, ( \eta_{inr} = p_{in}w_{inr} ) + linearizing constraints (revenue calculation)</td>
</tr>
<tr>
<td><strong>capacity</strong></td>
<td>uncapacitated or fixed capacity</td>
</tr>
</tbody>
</table>
# Revenue maximization problem

<table>
<thead>
<tr>
<th><strong>obj. fun. ((Z))</strong></th>
<th>max revenue from all services but the opt-out</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>DCM</strong></td>
<td>availability, discounted utility, choice</td>
</tr>
<tr>
<td><strong>price</strong></td>
<td>(p_{in} ) continuous, (\eta_{inr} = p_{in}w_{inr}) + linearizing constraints (revenue calculation)</td>
</tr>
<tr>
<td><strong>capacity</strong></td>
<td>uncapacitated or fixed capacity</td>
</tr>
</tbody>
</table>

- \(p_{in}\): different price per individual, groups or same price for everyone
## Revenue maximization problem

<table>
<thead>
<tr>
<th>obj. fun. ((Z))</th>
<th>max revenue from all services but the opt-out</th>
</tr>
</thead>
<tbody>
<tr>
<td>DCM</td>
<td>availability, discounted utility, choice</td>
</tr>
<tr>
<td>price</td>
<td>(p_{in} ) continuous, (\eta_{inr} = p_{in}w_{inr}) + linearizing constraints (revenue calculation)</td>
</tr>
<tr>
<td>capacity</td>
<td>uncapaticated or fixed capacity</td>
</tr>
</tbody>
</table>

- \(p_{in}\): different price per individual, groups or same price for everyone
- Individual price: disaggregate formulation (iterative procedure)
### Revenue maximization problem

<table>
<thead>
<tr>
<th>obj. fun. ($Z$)</th>
<th>max revenue from all services but the opt-out</th>
</tr>
</thead>
<tbody>
<tr>
<td>DCM</td>
<td>availability, discounted utility, choice</td>
</tr>
<tr>
<td>price</td>
<td>$p_{in}$ continuous, $\eta_{inr} = p_{in}w_{inr}$ + linearizing constraints (revenue calculation)</td>
</tr>
<tr>
<td>capacity</td>
<td>uncapacitated or fixed capacity</td>
</tr>
</tbody>
</table>

- $p_{in}$: different price per individual, groups or same price for everyone
- Individual price: disaggregate formulation (iterative procedure)
- Common price: aggregate formulation
Lagrangian relaxation

- Uncapacitated revenue maximization problem
Decomposition techniques: preliminaries

Lagrangian relaxation

- Uncapacitated revenue maximization problem
- **Idea:** relax utility functions (link between operator and customers)

\[
Z_{LR}(\rho, \psi) = \max \sum_{i \in C \{0\}} \sum_{n} \sum_{r} R_{\eta inr} \kappa + \sum_{i \in C} \sum_{n} \sum_{r} \rho_{inr} (U_{inr} - d_{inr} - \beta_{inp}) + \sum_{i \in C} \sum_{n} \sum_{r} \gamma_{inr} (v_{inr} - w_{inr})
\]

Relaxation utility function
Relaxation duplicate choice
Lagrangian relaxation

- Uncapacitated revenue maximization problem
- **Idea:** relax utility functions (link between operator and customers)
- Introduce duplicates of the choice to come up with independent sets of variables for each subproblem

\[
\text{LR}(\rho, \psi) = \max \sum_{i \in C \cup \{0\}} \sum_{n} \sum_{r} R_{inr} \rho_{inr} (U_{inr} - d_{inr} - \beta_{in} p_{in}) + \sum_{i \in C} \sum_{n} \sum_{r} \gamma_{inr} (v_{inr} - w_{inr})
\]
Lagrangian relaxation

- Uncapacitated revenue maximization problem
- **Idea:** relax utility functions (link between operator and customers)
- Introduce duplicates of the choice to come up with independent sets of variables for each subproblem

\[
Z^{LR}(\rho, \psi) = \max_{i \in C \setminus \{0\}, n, r} \sum_{i \in C \setminus \{0\}} \sum_{n} \sum_{r} \frac{1}{R} \eta_{inr} \\
+ \sum_{i \in C} \sum_{n} \sum_{r} \rho_{inr} (U_{inr} - d_{inr} - \beta_{in} p_{in}) \\
+ \sum_{i \in C} \sum_{n} \sum_{r} \gamma_{inr} (v_{inr} - w_{inr})
\]

- Relaxation utility function
- Relaxation duplicate choice

M. Pacheco, TRANSP-OR, EPFL
Limitations of the Lagrangian relaxation

Customer subproblem  Operator subproblem
Limitations of the Lagrangian relaxation

**Customer subproblem**
- DCM-related variables

**Operator subproblem**
Limitations of the Lagrangian relaxation

**Customer subproblem**
- DCM-related variables
- Decomposition by $n$ and $r$

**Operator subproblem**
Limitations of the Lagrangian relaxation

**Customer subproblem**
- DCM-related variables
- Decomposition by $n$ and $r$
- $U_{inr}$ set to the bounds

**Operator subproblem**
- Iterative methods to approximate the Lagrangian dual (e.g., subgradient method)
- Derivation of feasible solutions to the original problem
- Preserve supply-demand interplay: Lagrangian decomposition
Limitations of the Lagrangian relaxation

**Customer subproblem**
- DCM-related variables
- Decomposition by $n$ and $r$
- $U_{inr}$ set to the bounds

**Operator subproblem**
- Supply-related variables
Limitations of the Lagrangian relaxation

Customer subproblem
- DCM-related variables
- Decomposition by $n$ and $r$
- $U_{inr}$ set to the bounds

Operator subproblem
- Supply-related variables
- Decomposition by $n$
Limitations of the Lagrangian relaxation

**Customer subproblem**
- DCM-related variables
- Decomposition by $n$ and $r$
- $U_{inr}$ set to the bounds

**Operator subproblem**
- Supply-related variables
- Decomposition by $n$
- $p_{in}$ set to the bounds
Limitations of the Lagrangian relaxation

Customer subproblem
- DCM-related variables
- Decomposition by $n$ and $r$
- $U_{inr}$ set to the bounds

Operator subproblem
- Supply-related variables
- Decomposition by $n$
- $p_{in}$ set to the bounds

Trivial solutions:
Limitations of the Lagrangian relaxation

Customer subproblem
- DCM-related variables
- Decomposition by \( n \) and \( r \)
- \( U_{inr} \) set to the bounds

Operator subproblem
- Supply-related variables
- Decomposition by \( n \)
- \( p_{in} \) set to the bounds

Trivial solutions:
- Iterative methods to approximate the Lagrangian dual (e.g., subgradient method)
Limitations of the Lagrangian relaxation

Customer subproblem
- DCM-related variables
- Decomposition by $n$ and $r$
- $U_{inr}$ set to the bounds

Operator subproblem
- Supply-related variables
- Decomposition by $n$
- $p_{in}$ set to the bounds

Trivial solutions:
- Iterative methods to approximate the Lagrangian dual (e.g., subgradient method)
- Derivation of feasible solutions to the original problem $Z$
Limitations of the Lagrangian relaxation

**Customer subproblem**
- DCM-related variables
- Decomposition by $n$ and $r$
- $U_{inr}$ set to the bounds

**Operator subproblem**
- Supply-related variables
- Decomposition by $n$
- $p_{in}$ set to the bounds

- Trivial solutions:
  - Iterative methods to approximate the Lagrangian dual (e.g., subgradient method)
  - Derivation of feasible solutions to the original problem $Z$

- Preserve supply-demand interplay: Lagrangian decomposition
Lagrangian decomposition

- Capacitated revenue maximization problem
Lagrangian decomposition

- Capacitated revenue maximization problem
- **Idea:** duplicate the variables that are not draw-dependent ($p_{in}$)
Lagrangian decomposition

- Capacitated revenue maximization problem
- **Idea:** duplicate the variables that are not draw-dependent \((p_{in})\)
  - Lagrangian decomposition (variable splitting) in combinatorial optimization
Lagrangian decomposition

- Capacitated revenue maximization problem
- **Idea:** duplicate the variables that are not draw-dependent ($p_{in}$)
  - Lagrangian decomposition (variable splitting) in combinatorial optimization
  - Scenario decomposition in stochastic programming
Lagrangian decomposition

- Capacitated revenue maximization problem
- **Idea:** duplicate the variables that are not draw-dependent ($p_{in}$)
  - Lagrangian decomposition (variable splitting) in combinatorial optimization
  - Scenario decomposition in stochastic programming

$$p_{in1} = p_{in2} = \cdots = p_{inR}$$
Decomposition techniques: preliminaries

### Lagrangian decomposition

- Capacitated revenue maximization problem
- **Idea:** duplicate the variables that are not draw-dependent \( (p_{in}) \)
  - Lagrangian decomposition (variable splitting) in combinatorial optimization
  - Scenario decomposition in stochastic programming

\[
\begin{align*}
p_{in1} & = p_{in2} = \cdots = p_{inR} \\
Z^{LD}(\alpha) & = \max \sum_{i \in C \setminus \{0\}} \sum_n \sum_r \frac{1}{R} \eta_{inr} + \sum_{i \in C \setminus \{0\}} \sum_n \sum_r \alpha_{inr} (p_{inr} - p_{in(r+1)})
\end{align*}
\]

- Revenue
- Relaxation copy constraints
Limitations of the Lagrangian decomposition

- Decomposition by scenario (original problem for each draw)
Limitations of the Lagrangian decomposition

- Decomposition by scenario (original problem for each draw)
- Individual prices: might be set to the bounds if the service is not chosen (trivial solutions)
Limitations of the Lagrangian decomposition

- Decomposition by scenario (original problem for each draw)
- Individual prices: might be set to the bounds if the service is not chosen (trivial solutions)

\[ p_{inr} = \begin{cases} \max\{a_{in}, p^*_{inr}\}, & \text{if } \alpha_{inr} - \alpha_{in(r-1)} \leq 0, \ \forall i \in C_n \setminus \{0\} | w_{inr} = 0, \\ b_{in}, & \text{otherwise}, \end{cases} \]

\[ U_{inr}(p_{inr}) \]

\[ U_{inr}(p_{inr}) \]

\[ U_{jnr} \]

\[ p^*_{inr} \]
Outline

1. Introduction
2. Choice-based optimization framework
3. Decomposition techniques: preliminaries
4. Lagrangian decomposition scheme
5. Numerical experiments
6. Conclusions and future work
Generalization of the Lagrangian decomposition

- Capacitated revenue maximization problem
Generalization of the Lagrangian decomposition

- Capacitated revenue maximization problem
- Motivated by scenario grouping in stochastic programming
Generalization of the Lagrangian decomposition

- Capacitated revenue maximization problem
- Motivated by scenario grouping in stochastic programming
- Grouping scenarios within each subproblem
  - Improve the bound
  - Solving larger subproblems
Generalization of the Lagrangian decomposition

- Capacitated revenue maximization problem
- Motivated by scenario grouping in stochastic programming
- Grouping scenarios within each subproblem
  - Improve the bound
  - Solving larger subproblems

**Key idea**

1. $S$ groups of $R/S$ draws each
2. $p_{in}^1 = p_{in}^2 = \cdots = p_{in}^S$
Generalization of the Lagrangian decomposition

- Capacitated revenue maximization problem
- Motivated by scenario grouping in stochastic programming
- Grouping scenarios within each subproblem
  - Improve the bound
  - Solving larger subproblems

Key idea

1. $S$ groups of $R/S$ draws each
2. $p_{in}^1 = p_{in}^2 = \cdots = p_{in}^S$

- Relax copy constraints (Lagrangian decomposition)
Generalization of the Lagrangian decomposition

- Capacitated revenue maximization problem
- Motivated by scenario grouping in stochastic programming
- Grouping scenarios within each subproblem
  - Improve the bound
  - Solving larger subproblems

**Key idea**

1. $S$ groups of $R/S$ draws each
2. $p_{in}^1 = p_{in}^2 = \cdots = p_{in}^S$

- Relax copy constraints (Lagrangian decomposition)
- Relaxed problem splits into $S$ subproblems: $Z_{s}^{UB}(\alpha)$
Generalization of the Lagrangian decomposition

- Capacitated revenue maximization problem
- Motivated by scenario grouping in stochastic programming
- Grouping scenarios within each subproblem
  - Improve the bound
  - Solving larger subproblems

### Key idea

1. \( S \) groups of \( R/S \) draws each
2. \( p_{in}^1 = p_{in}^2 = \cdots = p_{in}^S \)

- Relax copy constraints (Lagrangian decomposition)
- Relaxed problem splits into \( S \) subproblems: \( Z_{UB}^S(\alpha) \)
- \( Z_{UB}(\alpha) = \sum_s Z_{UB}^s(\alpha) \) upper bound on \( Z \)
Generalization of the Lagrangian decomposition

- Capacitated revenue maximization problem
- Motivated by scenario grouping in stochastic programming
- Grouping scenarios within each subproblem
  - Improve the bound
  - Solving larger subproblems

**Key idea**

1. $S$ groups of $R/S$ draws each
2. $p_{in}^1 = p_{in}^2 = \cdots = p_{in}^S$

- Relax copy constraints (Lagrangian decomposition)
- Relaxed problem splits into $S$ subproblems: $Z_{UB}^S(\alpha)$
- $Z_{UB}(\alpha) = \sum_s Z_{UB}^s(\alpha)$ upper bound on $Z$
- Best upper bound (Lagrangian dual): subgradient method
Subgradient method

- Initialize Lagrangian multipliers: $\alpha^0$ (e.g., $\alpha^0 = 0$)

Diagram:

1. **initial values**
   - Lag. mult.
2. **solve subproblems**
3. **step size and direction**
4. **update Lag. multipliers**

Until stopping criterion
Subgradient method

- **Initial values**
- **Lag. mult.**
- **Solve subproblems**
- **Step size and direction**
- **Update Lag. multipliers**

Until stopping criterion

- **Upper bound**: Solve $Z_s^{UB}(\alpha^0), \forall s$ (CPLEX solver)
Subgradient method

- **Upper bound**: Solve $Z_{s}^{UB}(\alpha^0), \forall s$ (CPLEX solver)
- **Lower bound**: Obtain $Z^{LB}$ by generating feasible solutions for $Z$

Diagram:
- Initial values
  - Lag. mult.
- Solve subproblems
- Step size and direction
- Update Lag. multipliers
- Until stopping criterion
**Subgradient method**

- **Initial values**: Lag. mult.
- **Solve subproblems**
- **Step size and direction**
- **Update Lag. multipliers**

**Until stopping criterion**

- **Upper bound**: Solve $Z_s^{UB}(\alpha^0), \forall s$ (CPLEX solver)
- **Lower bound**: Obtain $Z^{LB}$ by generating feasible solutions for $Z$
- **Keep track of the best bounds found so far**: $Z_{UB,best}^{UB}$ and $Z_{LB,best}^{LB}$
Feasible solutions

- Sequence of prices: \( \{ \bar{p}_{in}^s \}_s \)
Feasible solutions

- Sequence of prices: \( \{ \bar{p}_i^s \}_s \)
- Solve \( Z \) for all configurations \( \bar{p}_i = \bar{p}_i^s \ \forall s \) and pick the highest
Feasible solutions

- Sequence of prices: \( \{\bar{p}^s_{in}\}_s \)
- Solve \( Z \) for all configurations \( \bar{p}_{in} = \bar{p}^s_{in} \) \( \forall s \) and pick the highest

\[
\text{Input:} \quad \text{Fixed prices} \ \bar{p}_{in}; \\
\text{Output:} \quad \text{Values for} \ y_{inr}, w_{inr}, U_{inr}, U_{nr} \text{ and } Z; \\
\text{Initialize} \ Z = 0; \\
\text{for} \ r = 1 \ldots R \ \text{do} \\
\text{Initialize occupancy level} \ o_{ir} = 0 \ \text{and} \ y_{inr} = 1; \\
\text{for} \ n = 1 \ldots N \ \text{do} \\
\text{for} \ i \in C_n \ \{0\} \ \text{do} \\
\quad \text{if} \ o_{ir} < c_i \ \text{then} \\
\quad \quad \text{Calculate} \ U_{inr} = \beta_{in} \bar{p}_{in} + d_{inr}; \\
\quad \text{else} \\
\quad \quad \text{Set} \ y_{inr} = 0 \ \text{and} \ U_{inr} = \ell_{nr}; \\
\text{Determine} \ w_{inr}, U_{inr}, U_{nr}; \\
\text{Update} \ Z = Z + \sum_{i \in C_n \ \{0\}} \frac{1}{R} w_{inr} \bar{p}_{in} \ \text{and} \ o_{jr} = o_{jr} + 1;
\]
Subgradient method

- **Step size:** $\gamma^k = \lambda^k \frac{Z^{\text{UB}}(\alpha^k) - Z^{\text{LB},\text{best}}}{\|v^k\|^2}$ ($\lambda^k$ step decreasing parameter)
Subgradient method

- **Step size:** \( \gamma^k = \lambda^k \frac{Z_{UB}(\alpha^k) - Z_{LB,best}}{\|v^k\|^2} \) (\( \lambda^k \) step decreasing parameter)

- **Step direction:** \( v^k = -(g^k + \zeta^k v^{k-1}) \)
  - subgradient: \( g_{ins}^k = p_{ins}^k - p_{in(s+1)}^k \)
  - deflection parameter: \( \zeta^k \)
Deflected subgradient method: zigzagging of kind I

- Angle between current subgradient and previous one might be obtuse
Deflected subgradient method: zigzagging of kind I

- Angle between current subgradient and previous one might be obtuse
- ⇒ next iterate near to the previous one (slows down convergence)
Deflected subgradient method: zigzagging of kind I

- Angle between current subgradient and previous one might be obtuse
- \[ \Rightarrow \] next iterate near to the previous one (slows down convergence)
Deflected subgradient method: zigzagging of kind I

- Angle between current subgradient and previous one might be obtuse
- $\Rightarrow$ next iterate near to the previous one (slows down convergence)

- Deflect the step direction to decrease the angle
Deflected subgradient method: zigzagging of kind I

- Angle between current subgradient and previous one might be obtuse
- \( \Rightarrow \) next iterate near to the previous one (slows down convergence)

- Deflect the step direction to decrease the angle
- Only when \( g^k \) forms an obtuse angle with the previous direction
Subgradient method

- Update Lagrangian multipliers: $\alpha^{k+1} = \alpha^k + \gamma^k v^k$

initial values → solve subproblems → step size and direction → update Lag. multipliers

until stopping criterion
Subgradient method

- Update Lagrangian multipliers: $\alpha^{k+1} = \alpha^k + \gamma^k v^k$
- Stopping criterion: computational time
Outline

1 Introduction
2 Choice-based optimization framework
3 Decomposition techniques: preliminaries
4 Lagrangian decomposition scheme
5 Numerical experiments
6 Conclusions and future work
Case study

- Parking choices
Case study

- Parking choices
- Common price: $p_i$ (aggregate formulation)
Comparison with optimal solutions

- $N = 50$, 5 draws per group and $R \in \{100, 250, 500\}$

<table>
<thead>
<tr>
<th>$R$ #Iter.</th>
<th>$Z_{UB,best}$ (it.)</th>
<th>$Z_{LB,best}$ (it.)</th>
<th>Avg. time it. (min)</th>
<th>gap dual (%)</th>
<th>gap opt (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>26.70</td>
<td>26.18</td>
<td>5.16</td>
<td>1.98</td>
<td>0.11</td>
</tr>
<tr>
<td>250</td>
<td>26.46</td>
<td>26.02</td>
<td>16.81</td>
<td>1.70</td>
<td>0.09</td>
</tr>
<tr>
<td>500</td>
<td>26.40</td>
<td>25.99</td>
<td>35.82</td>
<td>1.58</td>
<td>0.02</td>
</tr>
</tbody>
</table>
Comparison with optimal solutions

- \( N = 50, 5 \) draws per group and \( R \in \{100, 250, 500\} \)
- Run the LD scheme for 10\% of the exact computational time
Comparison with optimal solutions

- $N = 50$, 5 draws per group and $R \in \{100, 250, 500\}$
- Run the LD scheme for 10% of the exact computational time
- $\text{gap}_{\text{opt}} = \frac{Z - Z_{\text{LB,best}}}{Z}$, $\text{gap}_{\text{dual}} = \frac{Z_{\text{UB,best}} - Z_{\text{LB,best}}}{Z_{\text{LB,best}}}$
Comparison with optimal solutions

- $N = 50$, 5 draws per group and $R \in \{100, 250, 500\}$
- Run the LD scheme for 10% of the exact computational time
- $\text{gap}_{\text{opt}} = \frac{Z_{\text{LB,best}} - Z}{Z}$, $\text{gap}_{\text{dual}} = \frac{Z_{\text{UB,best}} - Z_{\text{LB,best}}}{Z_{\text{LB,best}}}$

<table>
<thead>
<tr>
<th>$R$</th>
<th>#Iter.</th>
<th>$Z_{\text{UB,best}}$ (it.)</th>
<th>$Z_{\text{LB,best}}$ (it.)</th>
<th>Avg. time it. (min)</th>
<th>$\text{gap}_{\text{dual}}$ (%)</th>
<th>$\text{gap}_{\text{opt}}$ (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>5</td>
<td>26.70 (5)</td>
<td>26.18 (2)</td>
<td>5.16</td>
<td>1.98</td>
<td>0.11</td>
</tr>
<tr>
<td>250</td>
<td>14</td>
<td>26.46 (14)</td>
<td>26.02 (1)</td>
<td>16.81</td>
<td>1.70</td>
<td>0.09</td>
</tr>
<tr>
<td>500</td>
<td>21</td>
<td>26.40 (21)</td>
<td>25.99 (7)</td>
<td>35.82</td>
<td>1.58</td>
<td>0.02</td>
</tr>
</tbody>
</table>
Numerical experiments

Evolution of bounds

\[ R = 100 \]

\[ R = 250 \]

\[ R = 500 \]

\( Z^{UB} \quad Z^{LB} \quad Z \)
Large number of draws

- $N = 50$, 2 draws per group, $R \in \{100, 250, 500, 1000, 2500, 5000\}$
Large number of draws

- $N = 50$, 2 draws per group, $R \in \{100, 250, 500, 1000, 2500, 5000\}$
- Time [min] as stopping criterion: $T \in \{30, 75, 150, 300, 750, 1500\}$
Numerical experiments

**Large number of draws**

- $N = 50$, 2 draws per group, $R \in \{100, 250, 500, 1000, 2500, 5000\}$
- Time [min] as stopping criterion: $T \in \{30, 75, 150, 300, 750, 1500\}$
- Average iteration time: 57 min ($R = 2500$) and 145 min ($R = 5000$)
Numerical experiments

Large number of draws

- $N = 50$, 2 draws per group, $R \in \{100, 250, 500, 1000, 2500, 5000\}$
- Time [min] as stopping criterion: $T \in \{30, 75, 150, 300, 750, 1500\}$
- Average iteration time: 57 min ($R = 2500$) and 145 min ($R = 5000$)
Large populations

- $R = 500$, 2 draws per group, $N \in \{50, 100, 150, 197\}$
Numerical experiments

Large populations

- \( R = 500 \), 2 draws per group, \( N \in \{50, 100, 150, 197\} \)
- Time [min] as stopping criterion: \( T \in \{150, 300, 450, 600\} \)
Large populations

- $R = 500$, 2 draws per group, $N \in \{50, 100, 150, 197\}$
- Time [min] as stopping criterion: $T \in \{150, 300, 450, 600\}$
- Average iteration time: 8 min ($N = 50$) and 95 min ($N = 100$)
Large populations

- $R = 500$, 2 draws per group, $N \in \{50, 100, 150, 197\}$
- Time [min] as stopping criterion: $T \in \{150, 300, 450, 600\}$
- Average iteration time: 8 min ($N = 50$) and 95 min ($N = 100$)
Trade-off with respect to the size of the draw groups

- $N = 50$, $R = 500$, number of draws per group $\in \{1, 2, 3, 4, 5, 10\}$

Same computational time limit $T = 150$ min

Less iterations as the number of draws per group increases
Trade-off with respect to the size of the draw groups

- $N = 50$, $R = 500$, \#draws per group $\in \{1, 2, 3, 4, 5, 10\}$
- Same computational time limit $T = 150$ min
Trade-off with respect to the size of the draw groups

- $N = 50$, $R = 500$, $\#\text{draws per group} \in \{1, 2, 3, 4, 5, 10\}$
- Same computational time limit $T = 150$ min
- Less iterations as the number of draws per group increases
Trade-off with respect to the size of the draw groups

- $N = 50$, $R = 500$, $\#$draws per group $\in \{1, 2, 3, 4, 5, 10\}$
- Same computational time limit $T = 150$ min
- Less iterations as the number of draws per group increases

![Graph showing the trade-off](image)

Lagrangian decomposition scheme

M. Pacheco, TRANSP-OR, EPFL
Outline

1. Introduction
2. Choice-based optimization framework
3. Decomposition techniques: preliminaries
4. Lagrangian decomposition scheme
5. Numerical experiments
6. Conclusions and future work
Conclusions

- Supply-demand interplay should not be dualized
Conclusions

- Supply-demand interplay should not be dualized
- Heuristic approach based on Lagrangian decomposition for the revenue maximization problem
Conclusions

- Supply-demand interplay should not be dualized
- Heuristic approach based on Lagrangian decomposition for the revenue maximization problem
- Speed up the solution approach with the generation of good feasible solutions (duality gaps < 4% in all instances)
Conclusions

- Supply-demand interplay should not be dualized
- Heuristic approach based on Lagrangian decomposition for the revenue maximization problem
- Speed up the solution approach with the generation of good feasible solutions (duality gaps < 4% in all instances)
- As long as the subproblems are computationally manageable, large number of draws per group is recommended
Future research directions

- Parallelization routines (to solve the subproblems, to generate feasible solutions)
Future research directions

- Parallelization routines (to solve the subproblems, to generate feasible solutions)
- Generalization of the approach with additional endogenous variables
Future research directions

- Parallelization routines (to solve the subproblems, to generate feasible solutions)
- Generalization of the approach with additional endogenous variables
- Combination with other techniques (e.g., Benders decomposition in the presence of discrete design variables) and variance reduction methods (to decrease the number of draws)
Questions?

THANK YOU