대한산업공학회 석사논문 경진대회

Strategic bundling approach for gig economy operation of electric scooter recharging

발표자: Minjeong Kim
지도 교수: Ilkyeong Moon

Supply Chain Management Lab,
Department of Industrial Engineering,
Seoul National University

November 13, 2020

ENGINEERING
COLLEGE OF ENGINEERING
SEOUL NATIONAL UNIVERSITY

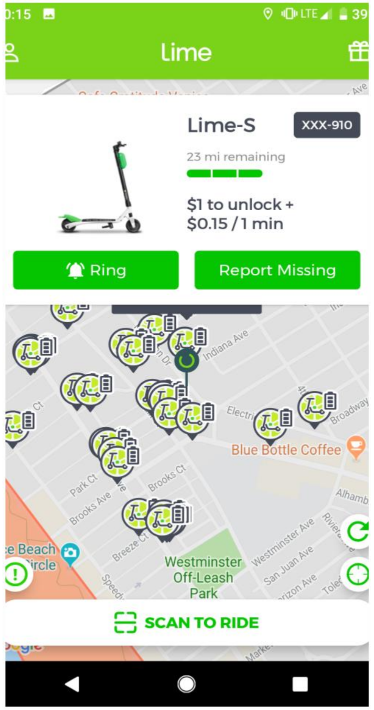
Context

1. Introduction

Electric scooter sharing service

Introduction(1/5)

First mile-last mile transportation



→ **Dockless**: No designated station for pick-up and return

→ **Convenient for short-distance travel** connecting transportation hubs

Increase in usage due to COVID-19

→ The number of usage has increased by more than **4.3 times** compared to the second half of last year

Mobility as a service(MaaS) for daily, non-face-to-face, micro transportation

Motivation

Introduction(2/5)

Gig economy

→ Free market system where organizations and independent workers engage in temporary or short-term work arrangements via platform

Gig Workers

- ✓ **Self-regulated**
- ✓ **Contingent**
- ✓ **Individual stakeholders**

Lime juicer

→ Network of independent contractors who collect and charge electric kickboards every night and redeploy them in the morning.

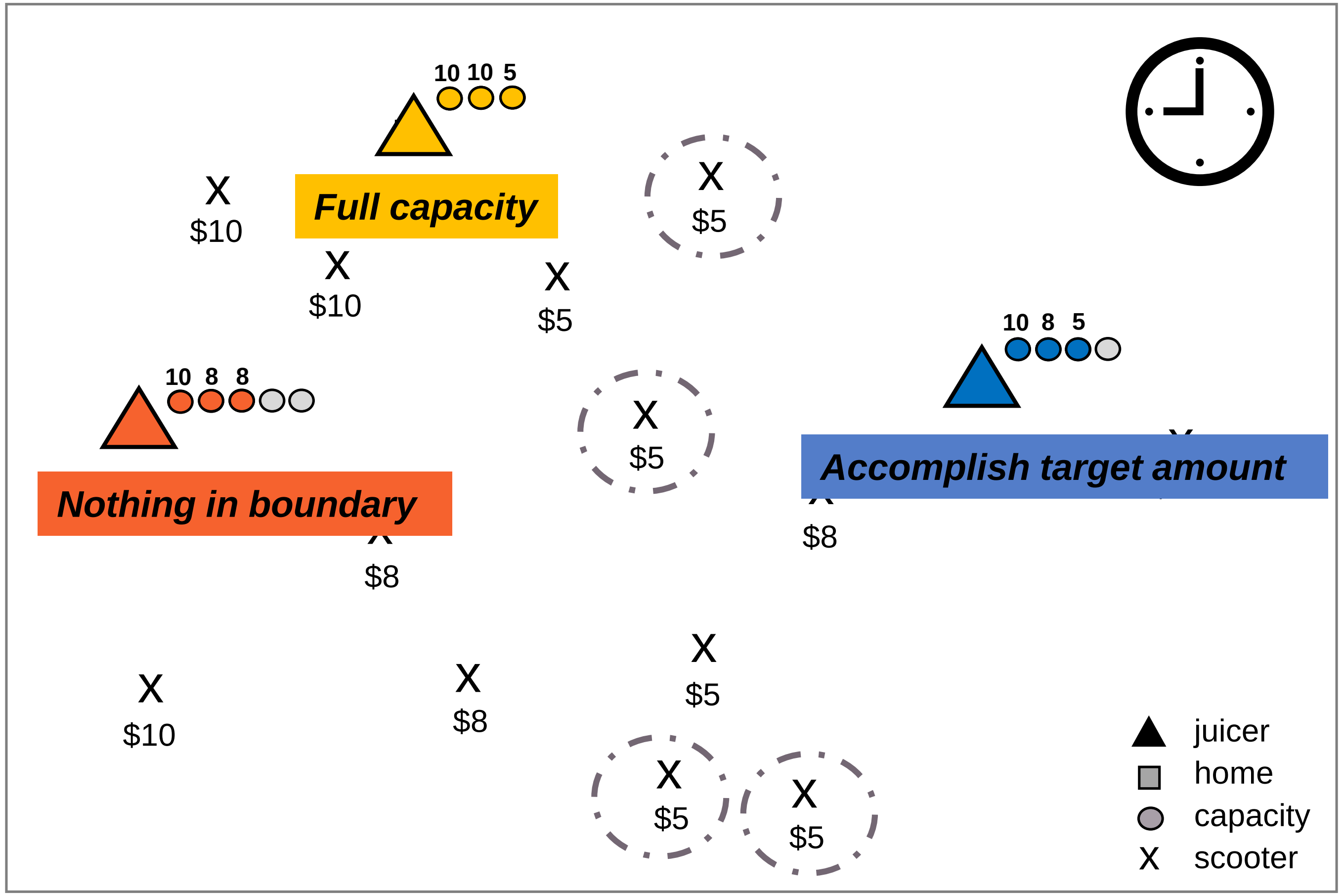
→ Collection Price

	USA	KOREA
Price	\$5~\$12 (\$8 avg)	₩3,000~₩4,000
Max. Price	\$20	₩6,000

Juicer System

Introduction(3/5)

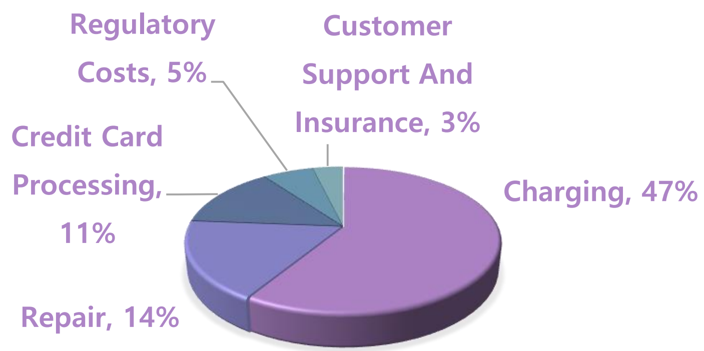
Lime's charging strategy with juicer



Collection rate: 9/13

1. Pricing of each scooter
2. Participation intention of juicer
3. Reservation (max 1 scooter)
4. Collection
5. **Recharging at home**

Electric Scooter Costs of Revenue



(Source: Crunchbase news, 2018.08)

Literature Review

Introduction(4/5)

E-scooter recharging operation

- Hub station design for e-scooters
  - Goshtasb (2018). *San Jose State University Graduate Research*

  → Violates the dockless system
- E-scooter assignment problem
  - Masoud et al. (2019) : *IEEE Access* 7

  → Allow company's intervention to chargers
- Stochastic model for e-scooter systems
  - Pender et al (2020). *Cornell University*

  → Different charging method

Combinatorial auction bundling

- Bidding strategy of combinatorial auction
  - An et al. (2005). *Journal of Revenue and Pricing Management*
- Scale and density discount of package
  - Olivares et al. (2012) : *Management Science*
- Transportation service procurement
  - Song et al (2020). *Transportation Research Record*

No research on efficient management of gig economy-based operation of charging e-scooters

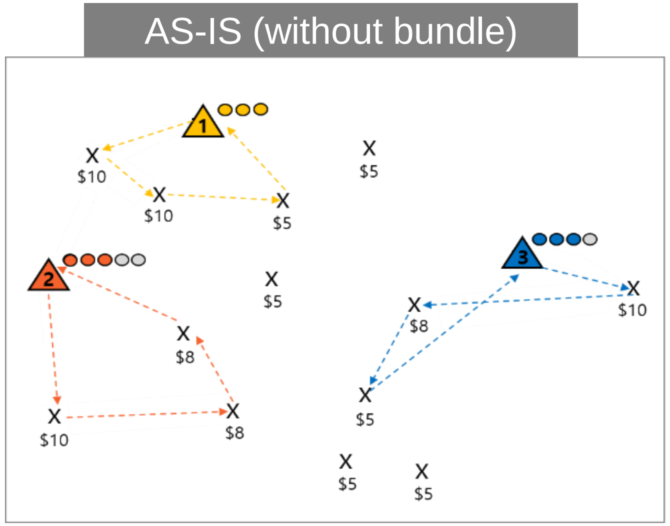
\+

Valuation of bidders toward package of items from cost synergies

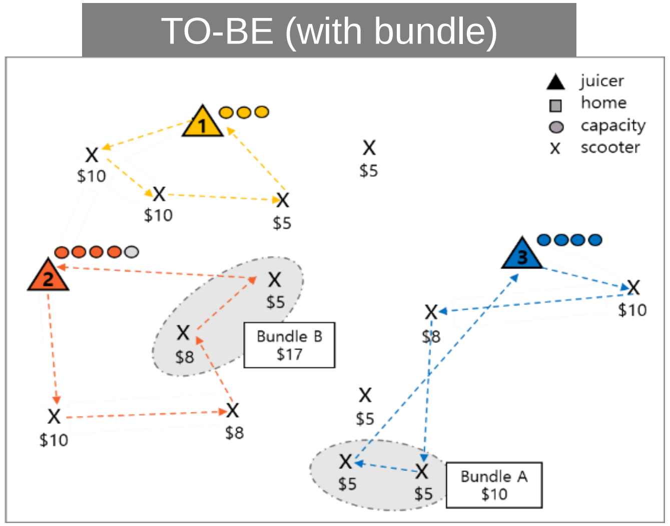
→ **Bundling strategy to increase collection rate of scooters**

In this thesis…

Introduction(3/5)



Collection rate: 9/13



Collection rate: 11/13

"Which bundle configuration is the best? "

→ Provide bundle of scooters as an alternative to collect

→ Promote juicers' collection activity when a company cannot directly manage workers

Objective

Create optimization-based bundling strategy to increase collection rate of scooters

2. Problem Statement

Preliminary concepts

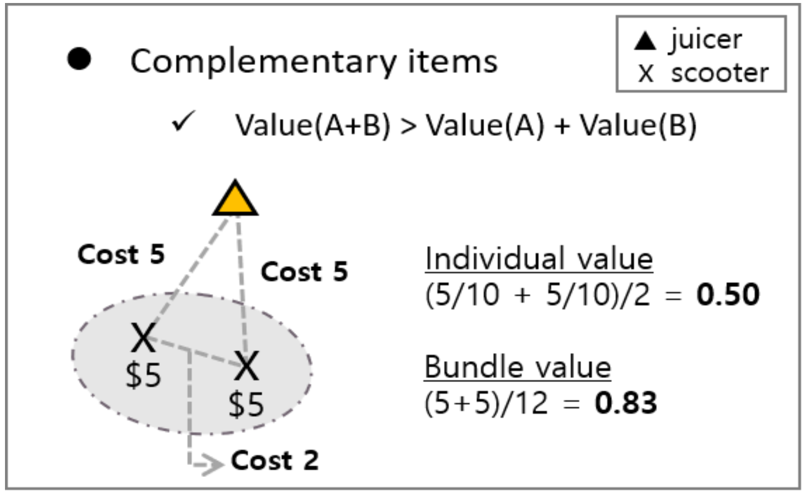
Problem Statement (1/3)

Combinatorial auction

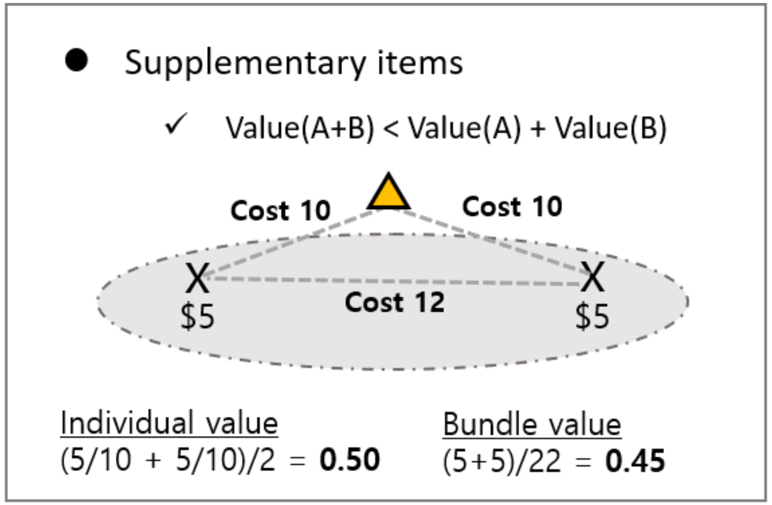
→ participants can place bids on combinations of items rather than individual items

2 properties of a bundle

1. A person's value of getting items together as a package is **greater than** the sum of values for each item individually



2. A person's value of getting items together as a package is **less than** the sum of values for each item individually



Assumptions

Problem Statement (1/3)

Juicer system

→ Juicers start collection all together at predetermined time.
→ Juicers do not know other juicer's location or status.
→ Maximum number of reservations is limited to one scooter or one bundle.
→ **Partial collection of a bundle is allowed**; scooters in a bundle can be collected individually not as whole.

Juicer capacity

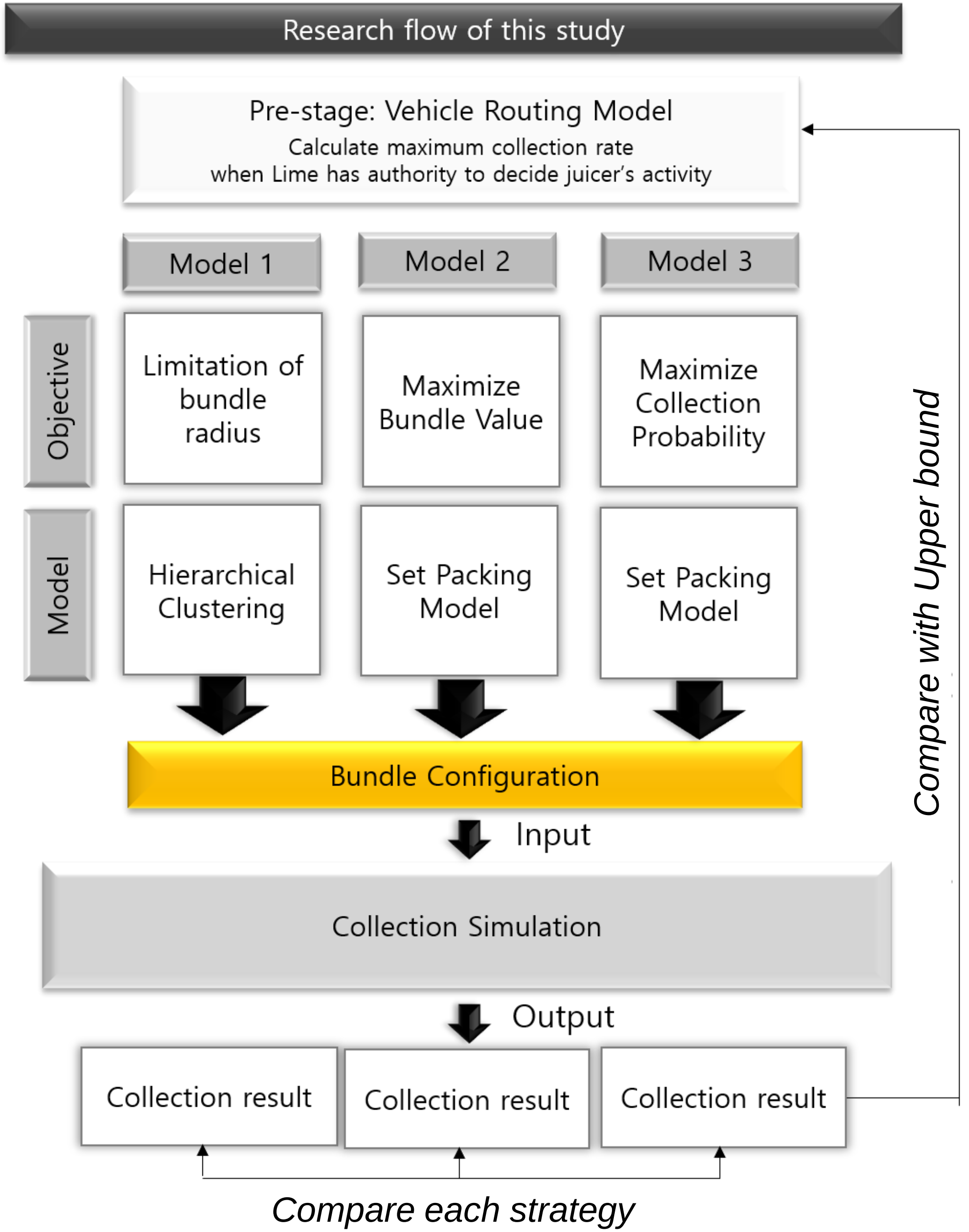
→ Juicers cannot charge more than the total charge amount possible with the chargers at home.
→ Juicer's capacity is equal.

Juicer behavior

→ Once the juicer completes the collection and return home, the collection activity that day is over.
→ Juicer collects **the most valuable scooter or bundles**, which offer the highest profit per distance to collect and return home.
→ If juicer achieves the target amount (e.g. collected half of chargers), they **only collect scooters of which value is more than certain criterion.**

Overview

Problem Statement (1/3)



3. Mathematical Models

Pre-stage Model

Mathematical Models (1/8)

Multi depot capacitated VRP

- Assumes that the company can control the juicers to pick up allocated scooters
- Calculate maximum collection rate of given scooter/juicer data

Sets

N : Set of scooter nodes

K : Set of juicer nodes (= depot nodes)

V : Set of nodes (= $N \cup K$)

Π : Set of arcs which violates relationship of juicer and depot

Decision Variables

x_{ij}^{k} : 1, if juicer k made collection from scooter i to j
0, otherwise

u_{i}^{k} : Juicer k's filled capacity in after collecting scooter i

Parameters

M : Large number

r : Small number

Q : Juicer's capacity of possible amount to charge battery

β : Threshold value of capacity filled by juicer where juicer satisfied expected amount of reward

α : Baseline value of scooter (=profit/distance to reach and go back home) when juicer's filled capacity is over β

p_j : Collection price of scooter j

q_j : Battery that needs to be charged of scooter j

$A[i,k,\alpha]$: Set of scooters j of which value is under α when juicer k's capacity is over β% after collecting scooter i

Pre-stage Model

Mathematical Models (2/8)

Multi depot capacitated VRP

$$Max \sum_{i\in V}\sum_{j\in V, i\neq j}\sum_{k\in K}(1 - r * p_j) * x_{ij}^k$$

Maximize the number of collected scooters while lowering the cost of the sum of collection price

(1) $x_{ij}^k = 0, \quad (\mathrm{i,j,k}) \in \Pi$ — *Remove juicer-depot violation*

(2) $\sum_{j\in N} x_{kj}^k = 1, \quad \forall k \in K$

(3) $\sum_{i\in N} x_{ik}^k = 1, \quad \forall k \in K$

Only travels once

(4) $\sum_{p\in V, p\neq i} x_{pi}^k = \sum_{t\in V, t\neq i} x_{it}^k, \quad \forall i \in N, \forall k \in K$

Balance equation

(5) $\sum_{i\in V, i\neq j}\sum_{k\in K} x_{ij}^k \leq 1, \quad \forall j \in N$

(6) $\sum_{j\in V, i\neq j}\sum_{k\in K} x_{ij}^k \leq 1, \quad \forall i \in N$

Scooter can be collected by one juicer maximum or not visited at all

(7) $u_i^k = 0, \quad \forall i \in K, \forall k \in K, i = k$ — *Filled capacity in depot is zero*

(8) $u_i^k + q_j \leq u_j^k + M(1 - x_{ij}^k), \quad \forall i \in V, \forall j \in N, \forall k \in K, i \neq j$ — *Capacity update*

(9) $u_j^k \leq \left(\sum_{p\in V, p\neq j} x_{pj}^k\right) * M, \quad \forall j \in N, \forall k \in K$ — *Capacity is positive only when juicer visited*

(10) $u_i^k - Q * \beta \leq M * (1 - x_{ij}^k), \quad \forall i \in V, \forall k \in K, \forall j \in A[i,k,\alpha]$ — *Juicer behavior constraint*

(11) $x_{ij}^k \in \{0,1\}, \quad \forall i,j \in V, \forall k \in K$

(12) $u_i^k \geq 0, \quad \forall i \in V, \forall k \in K$

Pre-stage Model

Mathematical Models (3/8)

Juicer behavior constraint

precomputed

$$(10) \quad u_i^k - Q * \beta \le M * \left(1 - x_{ij}^k\right), \qquad \forall i \in V, \forall k \in K, \forall j \in A[i,k,\alpha]$$

Contains scooters j which are impossible to collect when juicer collected scooter i and k's filled capacity is over $\beta\%$ of the maximum capacity

Juicer behavior assumption

If a juicer collects more than $\beta\%$ of the maximum capacity, only the scooters that are worth more than α will be collected

$$\text{Value of scooter (node) j from scooter (node) i} = \frac{\text{Price of scooter j}}{\text{Distance from node i to j} + \text{Distance from j to depot}}$$

Pre-stage Model

Mathematical Models (4/8)

Cut constraint

(13) $$\sum_{i\in V}\sum_{j\in V, i\neq j}\sum_{k\in K} x_{ij}^{k} \leq findcut + |K|$$

Upper bound of the number of collected scooters

(14) $$\sum_{i\in V}\sum_{j\in V, i\neq j}\sum_{k\in K} \left(1 - r * p_j\right) * \ x_{ij}^{k} \leq findcut + |K| - r * pricesumcut$$

Upper bound of objective function considering price sum of collected scooters

Each scooter j sorted by q_j

20%	26%	40%	50%	...	78%	80%	90%

Cumulation

20%	46%	86%	136%	...	1190%	1250%	1300%

$findcut$

$\boldsymbol{|K| * Q = 1200}$

Each scooter j sorted by p_j

$5	$5	$5	$8	...	$10	$10	$10

Cumulation

$5	$10	$15	$23	...	$95	$105	$115

$findcut$

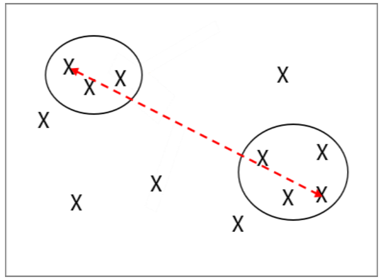
$\boldsymbol{pricesumcut} = \$95$

Three types of bundling model (1)

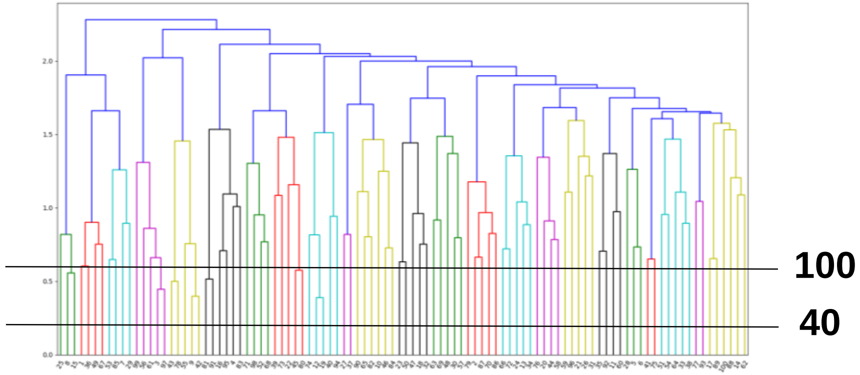
Mathematical Models (5/8)

Model Ⅰ – Limiting cluster size

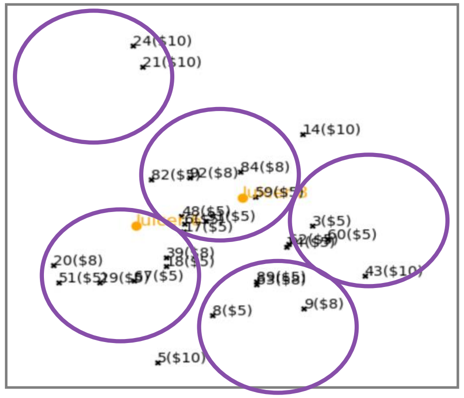
- Generate bundle of scooters with high proximity
- Create bundles with complementary properties
- Limitation of bundle size with a predetermined radius



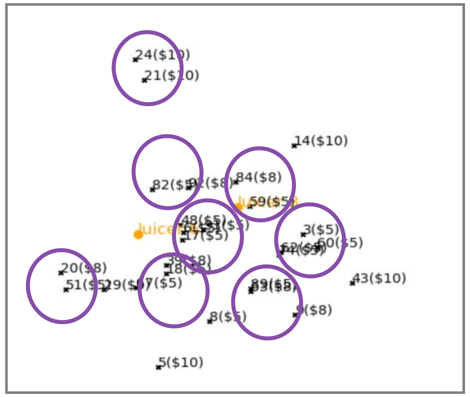
Complete linkage distance measure



Agglomerative clustering dendrogram based on scooter proximity



Bundle size = 100



Bundle size = 40

Three types of bundling model (2)

Mathematical Models (6/8)

*Standard distance deviation of bundle j with scooter $1, \dots, P$

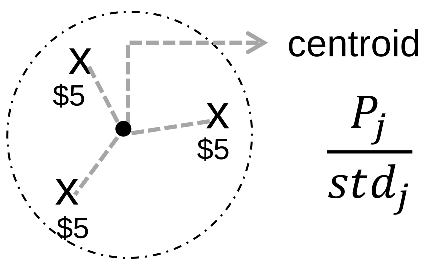
$$\begin{cases} = \sqrt[2]{\sum_{p=1}^{P} dist(centroid,p)^2 / _{P-2}}, & \text{If } P > 2 \\ = \sum_{p=1}^{P} dist(centroid,p), & \text{Otherwise} \end{cases}$$

Model Ⅱ – Maximize the sum of bundle value

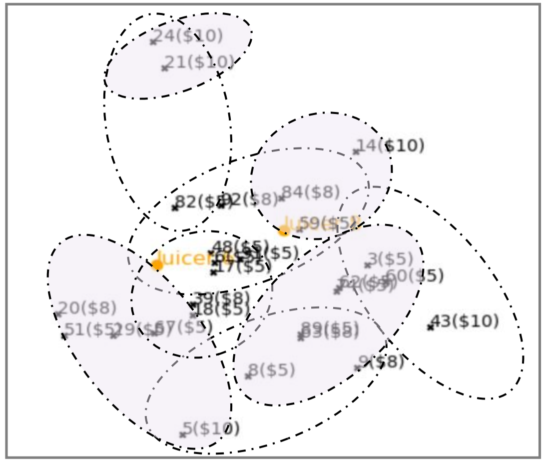
- Idea
  - ✓ Generate bundle candidates with nearby scooters
  - ✓ Select winning bundles by Set packing model

Bundle generator ➡ Generate bundle with scooters less than $MaxBundleSize$

Value of bundle j



$$\frac{P_j}{std_j}$$



(MD 2) Set Packing Model

$$Max \sum_{j \in B} \left(\frac{P_j}{std_j}\right) x_j$$

Maximize bundle value

$$s.t \quad \sum_{j \in B} a_{ij} x_j \leq 1, \quad i \in N$$

$$x_j \in \{0,1\}, \quad j \in B$$

Each scooter should be included in only one bundle

x_j : Binary decision variable

$$\begin{cases} = 1, & \text{If bundle } j \in B \text{ is selected as winning bundle} \\ = 0, & \text{Otherwise} \end{cases}$$

P_j : Price of bundle j (= sum of scooter price in the bundle j)

std_j : Standard distance deviation of scooters in bundle j *

a_{ij}

$$\begin{cases} = 1, & \text{If bundle } j \in B \text{ contains scooter } i \in N \\ = 0, & \text{Otherwise} \end{cases}$$

*Levine, Ned. "CrimeStat III: a spatial statistics program for the analysis of crime incident locations (version 3.0)." *Houston (TX): Ned Levine & Associates/Washington, DC: National Institute of Justice* (2004).

Three types of bundling model (2)

Mathematical Models (7/8)

Bundle generator

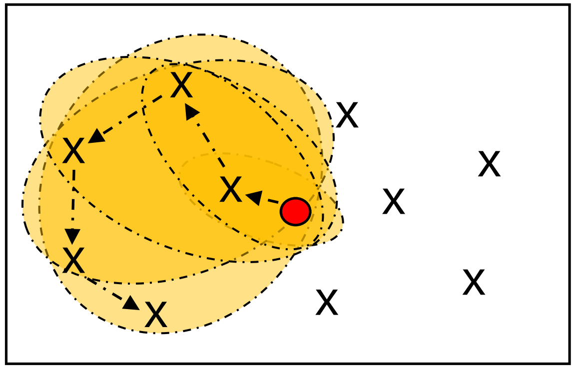
$$\text{Standard distance deviation of bundle } j \text{ with scooter } 1,\dots,P \begin{cases} = \sqrt[2]{\sum_{p=1}^{P} dist(centroid,p)^2 / _{P-2}}, & \text{If } P > 2 \\ = \sum_{p=1}^{P} dist(centroid,p) \;, & \text{Otherwise} \end{cases}$$

INPUT scooter node $\forall i \in \mathrm{N}$

PROCEDURE

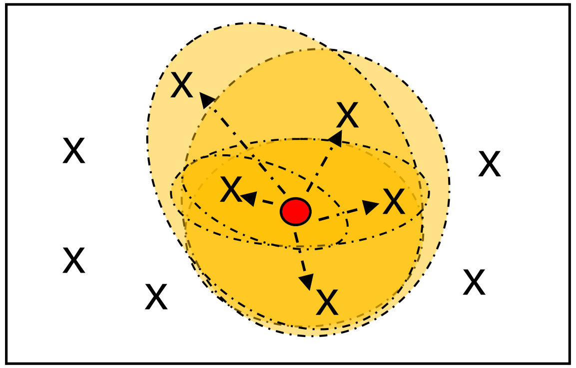
1. Near pack generator

Generate bundles by adding nearest scooter to the last added scooter



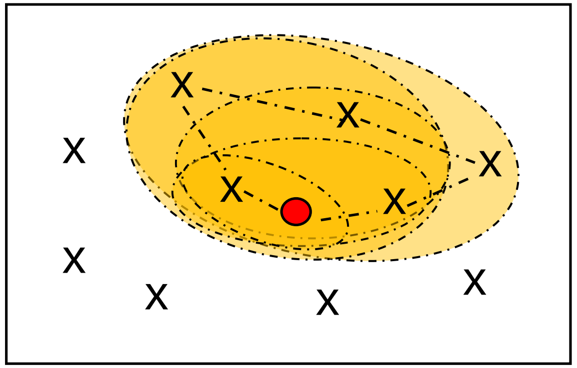
2. Node pack generator

Generate bundles by adding the nearest scooters one by one depending on the distance from the base node



3. STD pack generator

Generate bundles by adding the scooter which increases the standard distance deviation to the smallest



OUTPUT Result bundle = 1. + 2. + 3.

● Base node
X Candidate node

Three types of bundling model (3)

Mathematical Models (8/8)

Model Ⅲ – Maximize the sum of probability of collection

- Idea
  - ✓ Select bundle configuration of which collection probability sum is maximized
  - ✓ Generate bundle candidates with bundle generator
  - ✓ Select winning bundles by Set packing model

$$\Pr(k,j) \begin{cases} = \exp(-\lambda * dist(k,j)/V_j), & \text{If } dist(k,j) \le \delta \\ = 0, & \text{Otherwise} \end{cases}$$

λ	: decay coefficient
δ	: distance limitation parameter
$dist(k,j)$	: distance between juicer k's depot and central point of bundle j

$$\Pr(j) = 1 - \prod_{k \in K} (1 - \Pr(k,j))$$

(MD 3) Set Packing Model

$$Max \quad \sum_{j \in B} (\Pr(j) - \gamma * Incentive_j) x_j$$

$$s.t \quad \sum_{j \in B} a_{ij} x_j \le 1, \quad i \in N$$

$$\sum_{j \in B} Incentive_j x_j \le Budget$$

$$x_j \in \{0,1\}, \quad j \in B$$

Maximize collection probability

Each scooter should be included in only one bundle

Total incentive cannot exceed Budget

x_j	: Binary decision variable $\begin{cases} = 1, & \text{If bundle } j \in B \text{ is selected as winning bundle} \\ = 0, & \text{Otherwise} \end{cases}$
$\Pr(j)$	: Probability of bundle j being collected
$Incentive_j$	: incentive provided for supplement bundle according to bundle incentive rate
Budget	: Budget for incentive
γ	: coefficient
a_{ij}	$\begin{cases} = 1, & \text{If bundle } j \in B \text{ contains scooter } i \in N \\ = 0, & \text{Otherwise} \end{cases}$

4. Computational Experiments

Simulation – Performance measurement

Computational Experiments (1/5)

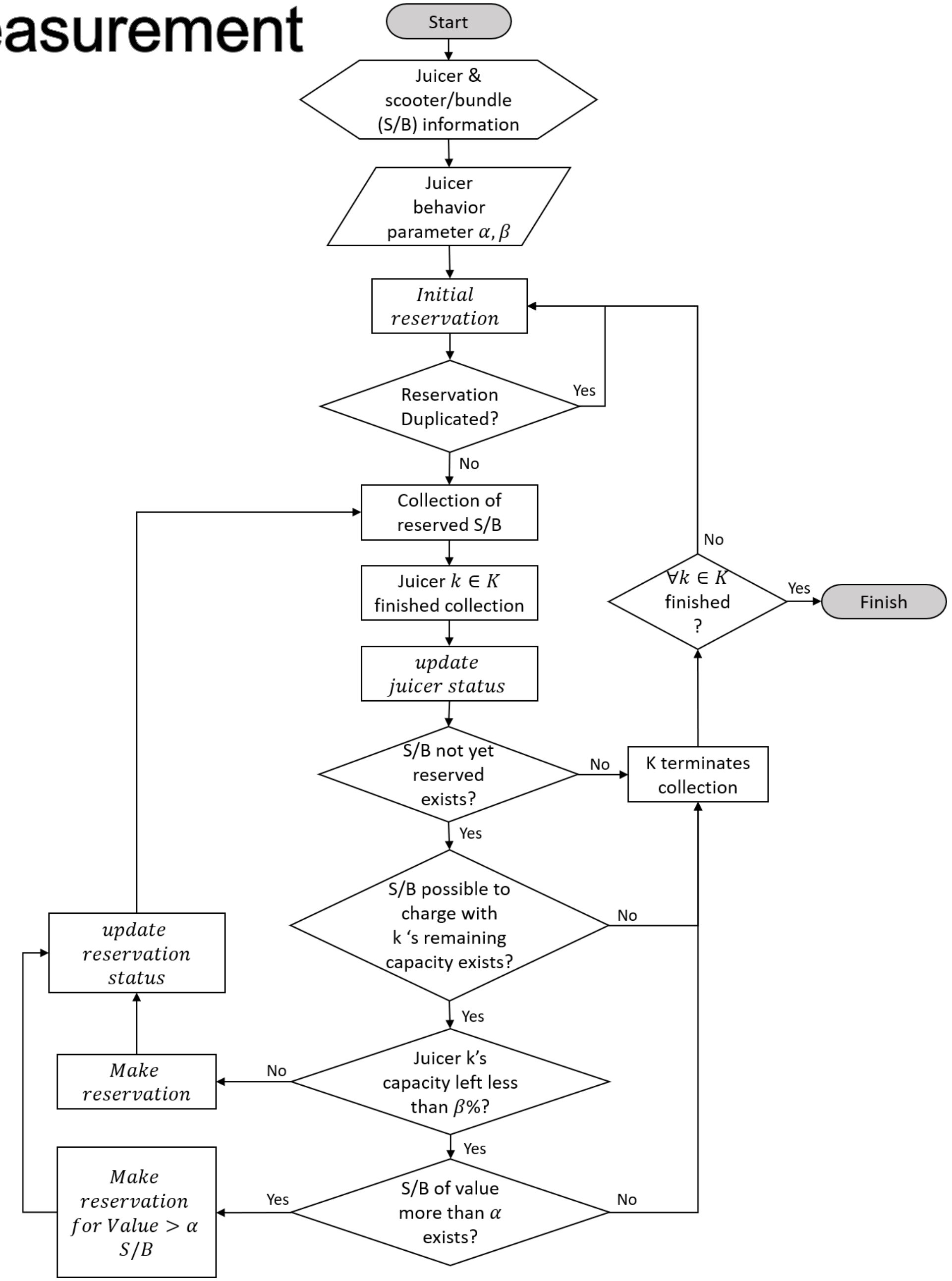
Collection simulation of juicers

Juicer collection rule

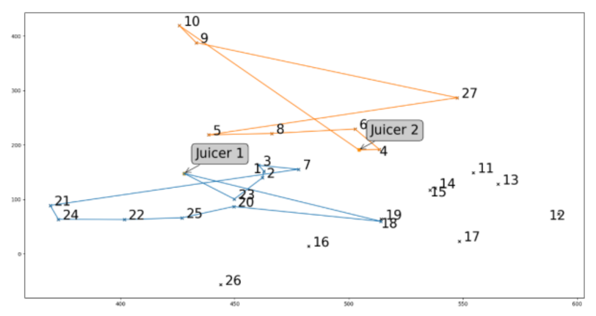
- Collect the most valuable scooter regardless of distance
- Partial collection allowed for bundles (no incentive)
- Collect scooters with value more than α when juicer capacity is over $\beta\%$

Collection Termination

- Code 01: No scooter left to collect
- Code 02: Out of capacity
- Code 03: No scooter value more than α (if capacity is over $\beta\%$)



Without bundle (AS-IS)

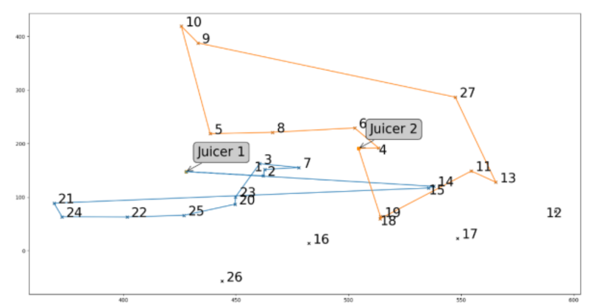


Total collected scooter: **18 out of 27**

J1:[depot→23→2→1→3→7→21→24→22→25→20→18→depot]

J2: [depot→4→6→8→5→27→9→10→depot]

With bundle (Model Ⅱ)



Total collected scooter: **23 out of 27**

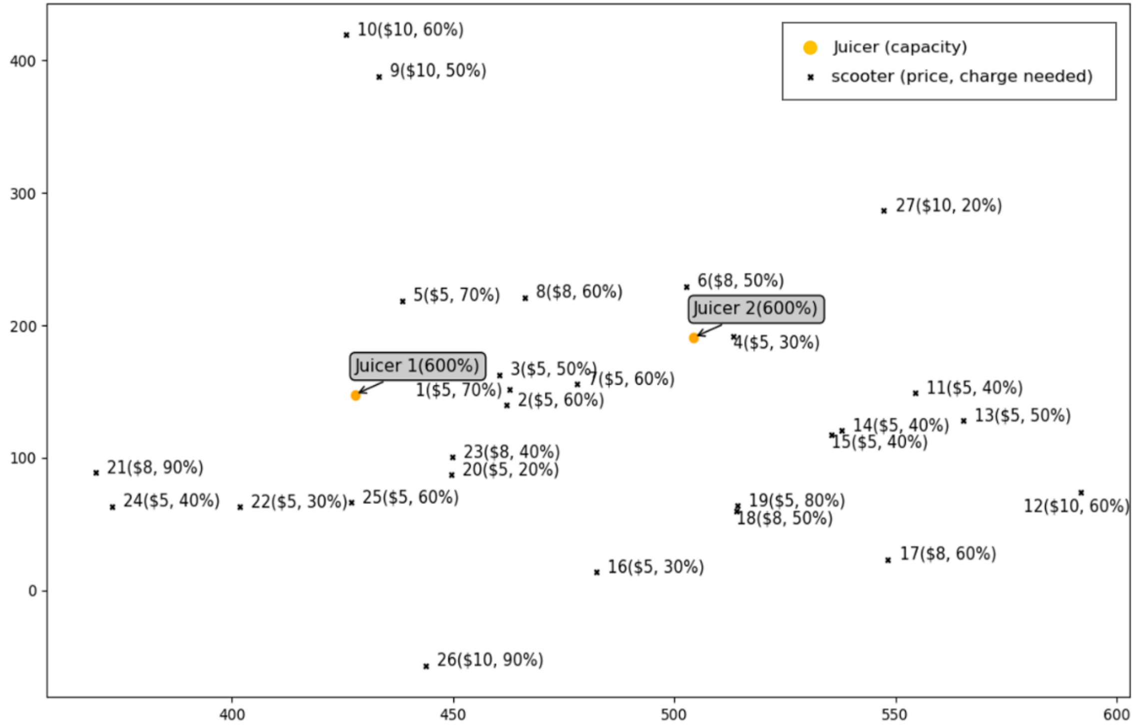
J1:[depot→28(2→1→7→3)→36(23→20)→38(25→22)→37(24→21)→33(15→14) →depot]

J2:[depot→4→6→30(8→5)→31(10→9)→27→32(13→11)→35(18→19)→depot]

Description of parameters

Computational Experiments (2/5)

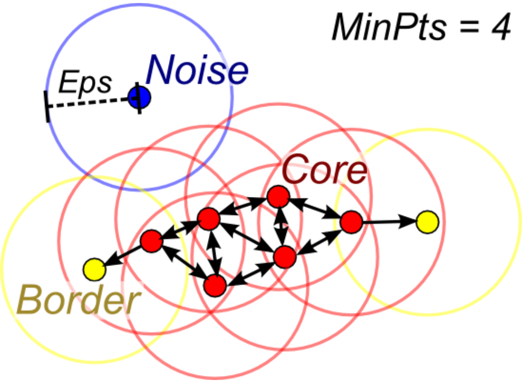
- test instance of 2 juicers and 27 scooters



- Description of data generation

		Value	Method
Scooter	Price	\$5, \$8, \$10	DBSCAN clustering
	Battery level	10%~80%	Selected randomly
Juicer	Capacity	600%	Fixed
	α	0.03	Fixed
	β	0.5	Fixed

- DBSCAN clustering (Density based spatial clustering of applications with noise)



- ✓ Core: \$5
- ✓ Border: \$8
- ✓ Noise: \$10

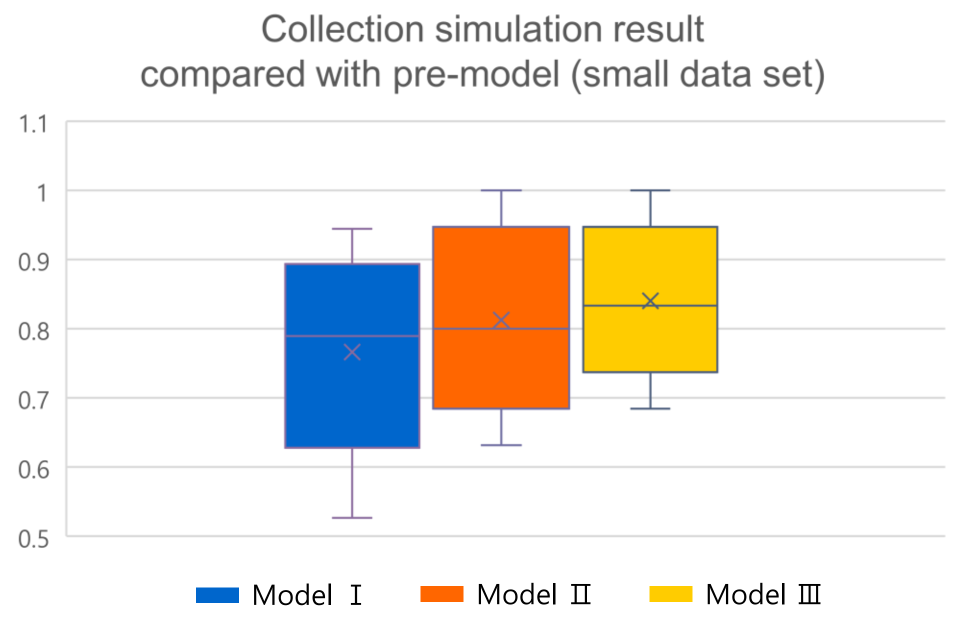
MinPts = 3, Eps = 60

Results

Computational Experiments (3/5)

Pre-model vs bundling model

Clustered scooter(#) : Randomly located scooter(#)	Total scooter(#)	Total juicers
15 : 5	20	2



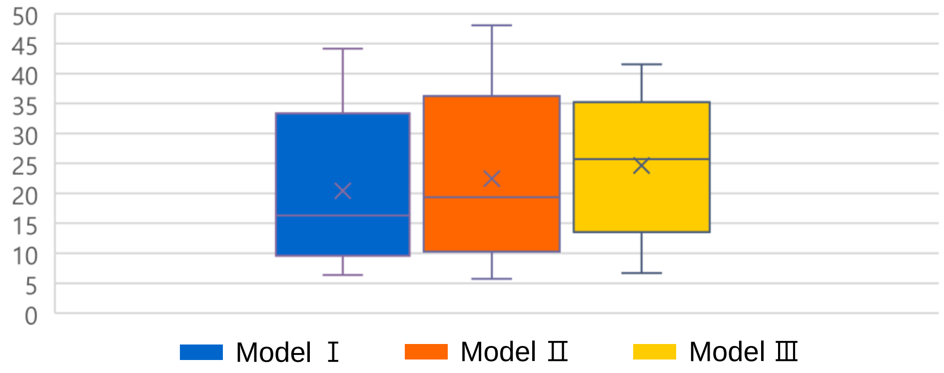
Results

Computational Experiments (4/5)

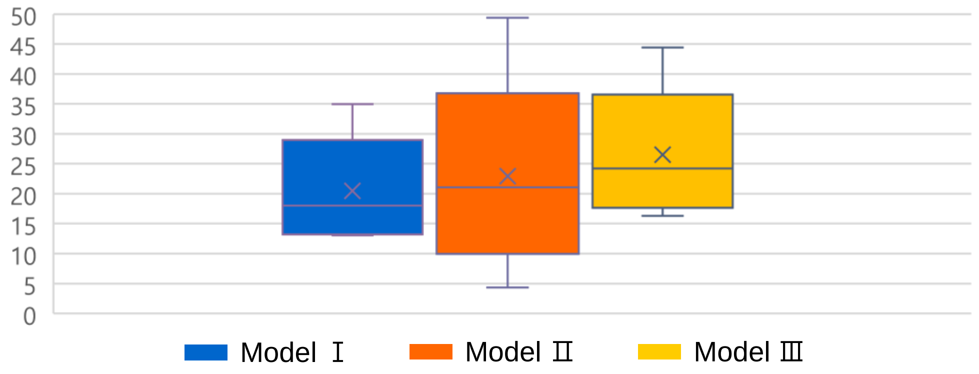
AS-IS vs Model

Scenario	Clustered scooter(#) : Randomly located scooter(#)	Total scooter(#)	Total Juicers
I	120 : 30	150	10
II	80 : 70		

Collection simulation result
compared with AS-IS (big data set) - Scenario Ⅰ



Collection simulation result
compared with AS-IS (big data) – Scenario Ⅱ



Results

Computational Experiments (5/5)

Juicer status

Scenario	Clustered scooter(#) : Randomly located scooter(#)	Total scooter(#)	Total Juicers
I	120 : 30	150	10
II	80 : 70		

Juicer information - Scenario Ⅰ

Scenario Ⅰ (120:30)	AS-IS	Model Ⅰ	Model Ⅱ	Model Ⅲ
Achievement rate(%)	62.5333	73.4083	74.9292	77.1417
Total distance(10m)	1038.6482	991.4776	1080.9820	1098.6147
Profit($)	64.6067	69.7500	72.8667	73.8549
Margin	54.2202	59.8352	62.0568	**62.8688**

Juicer information - Scenario Ⅱ

Scenario Ⅱ (80:70)	AS-IS	Model Ⅰ	Model Ⅱ	Model Ⅲ
Achievement rate(%)	60.9458	72.3750	73.9875	76.3000
Total distance(10m)	1165.1820	1074.5587	1210.7677	1191.5665
Profit($)	72.0667	76.2900	81.3600	82.4004
Margin	60.4148	65.5444	69.2523	**70.4847**

5. Conclusion

Conclusion

Contribution

- Research on gig economy-based scooter charging operation
- Proposed three bundling models to improve collection rate of scooters
- Provided efficient strategy for both the company and juicers

→ For Lime (Company)

✓ Lowering cost of recharging, increasing competitive advantage

✓ Help to control proportion of juicer and Lime’s full-time worker and induce juicers to join the community

→ For Juicer (Gig worker)

✓ Lower uncertainty of juicers from competition by providing bundles and helping juicers to plan their route

✓ Assist juicers to obtain higher margin

감사합니다