Simulated Annealing Algorithm for Customer–Centric Location Routing Problem

May 22, 2018

Eugene Sohn
Advisor: Mohammad Moshref-Javadi, PhD

Agenda

- **Why this research?**
- **What is this research?**
- **Methodology**
- **Mathematical Model & Computational Results**
- **Heuristic Algorithm & Computational Results**
- **Conclusion**

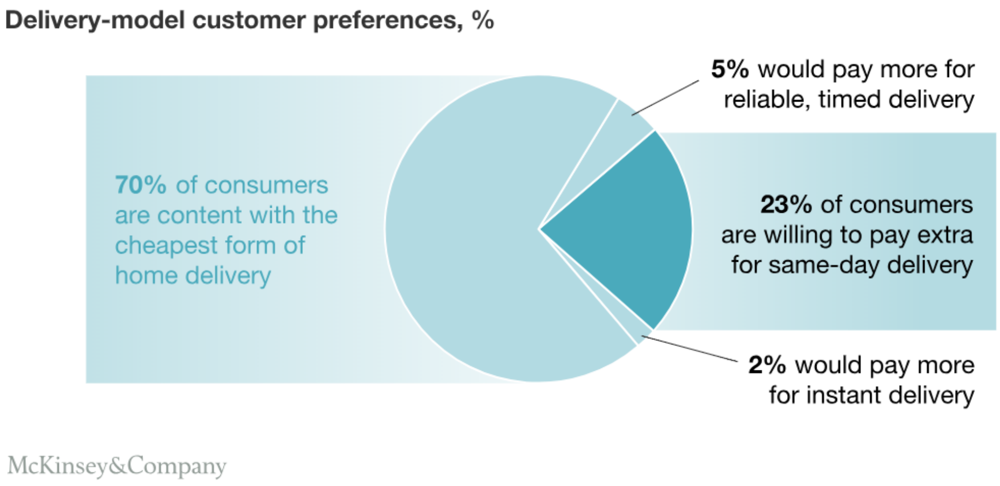
Why This Research?

E-Commerce Market Growing Rapidly

- Sales grew from $360b to $409b from 2016 to 2017 (14% growth)
- Expected sales in 2021 to be $603b (YoY 8.5% growth)

Change in Consumer's Expectation on Deliveries and Shipping

- A quarter of consumers would pay a premium for same-day delivery



Why This Research? (cont.)

Growing Market + Change in Consumer Expectation

=

Opportunities for Disruption and Market Share Gain

By providing faster delivery for better customer shopping experience

What is This Research?

Capacitated Latency Location Routing Problem:

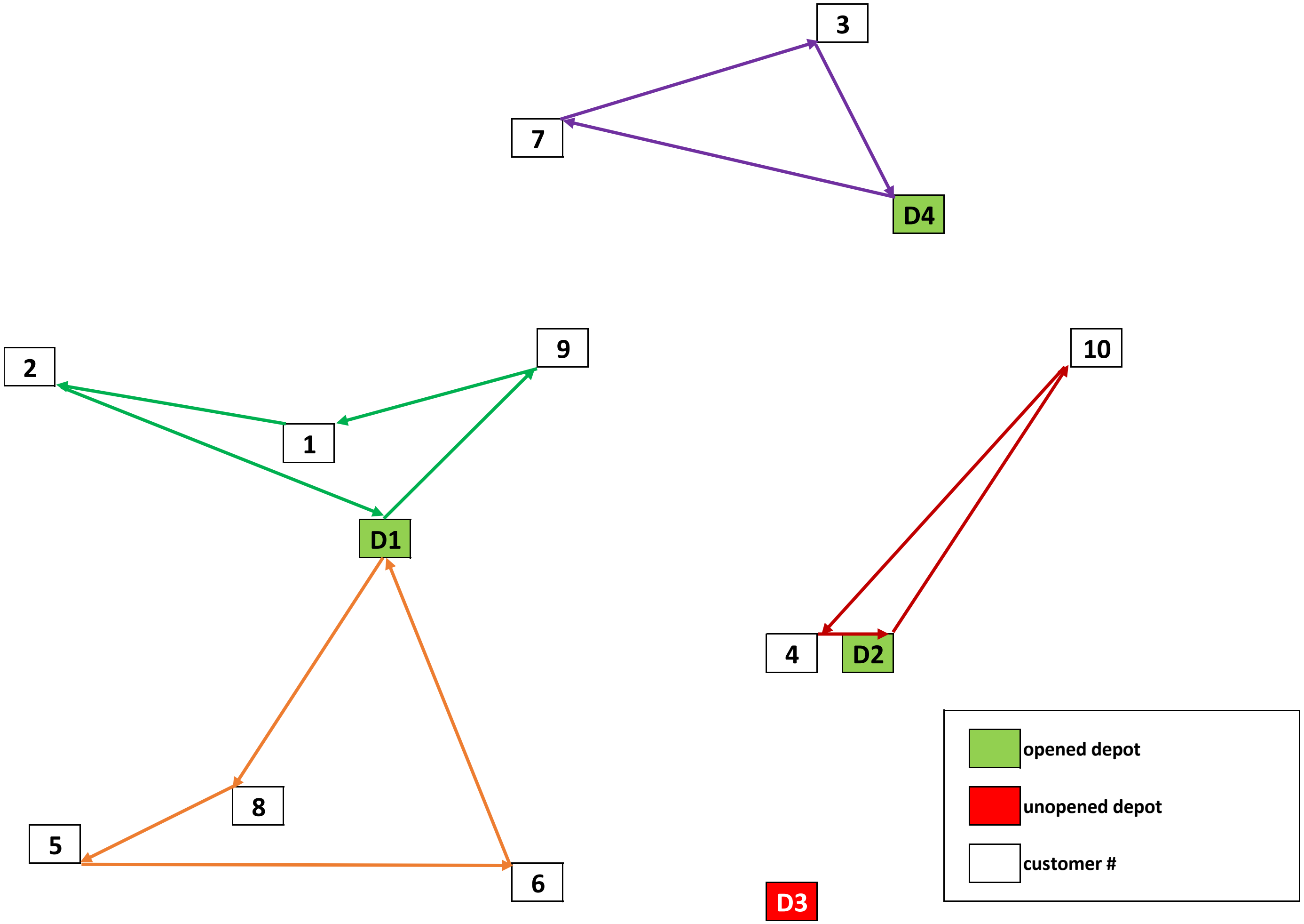
Location Routing Problem

- Determines location of depots, allocation of vehicles, and routes of the vehicles concurrently
- Facility Location Problem (NP-Hard)
- Vehicle Routing Problem (NP-Hard)

Capacitated Latency

- Minimize Customer Waiting Time vs Cost
- Capacity Constraints on Vehicles and Warehouses

Capacitated Latency Location Routing Problem



Example of a location routing problem

Methodology

Two Ways to Solve CLLRP

- **Mathematical Model**
- **Heuristic Algorithm**

Mathematical Model	Heuristic Algorithm
Always provides the best solution	Provides good enough solution
Computationally inefficient	Computes very quickly

Mathematical Model

Assumptions

- Number and locations of candidate depots are known
- Number of depots to open and vehicles to use are pre-determined
- Capacities of depots and vehicles are pre-determined
- All the demands are satisfied
- Travel time between customer i and j are symmetric

GUROBI OPTIMIZATION

python™

Mathematical Model (cont.)

Indices	
i,j,u	Represent customers, totally N_c customers
k	Represents vehicle
g	Represents candidate depots, totally N_f

Sets	
K	Set of vehicles, $\|K\|$
G	Set of candidate depots, $\|G\| = N_f$
V’	Set of customers, $\|V'\| = N_c$
V	Set of all customers and candidate depots $\|V\| = N = N_c + N_f$

Mathematical Model (cont.)

Parameters	
N_v	Number of vehicles
Wg	Capacity of depot g
q_j	Demand quantity at customer j
Q_k	Capacity of vehicle k
c_{ij}	Travel time between nodes i and j
N_g	Number of facilities to open
M	Large positive constant

Variables	
t_i^k	Arrival time of vehicle k at customer i
x_{ij}^k	1 if vehicle k traverses arc (i,j) from customer i to customer j ; otherwise, 0
f_{gi}	1 if customer i is supplied from depot g; otherwise, 0
z_g	1, if facility g is open; otherwise 0

Mathematical Model (cont.)

Minimize:

$$\sum_{k \in K, i \in V'} t_i^k \quad (1)$$

s.t.

$$\sum_{i \in V'} f_{gi} q_i \leq W_g \ \forall g \in G \quad (2)$$

$$\sum_{j \in V} x_{ij}^k = \sum_{j \in V} x_{ji}^k \ \forall i \in V, \forall k \in K \quad (3)$$

Mathematical Model (cont.)

$$\sum_{k\in K, j\in V', i\neq j} x_{ij}^{k} = 1 \ \forall i \in V' \qquad (4)$$

$$\sum_{g\in G} f_{gj} = 1 \ \forall j \in V' \qquad (5)$$

$$\sum_{i\in V, j\in V'} x_{ij}^{k} q_j \leq Q_k \ \forall k \in K \qquad (6)$$

$$\sum_{g\in G, i\in V'} x_{gi}^{k} = 1 \ \forall k \in K \qquad (7)$$

$$\sum_{u\in V'} x_{gu}^{k} + \sum_{u\in V\setminus\{i\}} x_{ui}^{k} \leq 1 + f_{gi} \ \forall i \in V', \forall k \in K, \forall g \in G \qquad (8)$$

Mathematical Model (cont.)

$$t_i^k + c_{ij} - (1 - x_{ij}^k)M \leq t_j^k, \forall i \in V, \forall j \in V', \forall i \neq j, \forall k \in K, \forall g \in G \quad (9)$$

$$\sum_{i \in V'} f_{gi} \leq M z_g \; \forall g \in G \quad (10)$$

$$\sum_{i \in V'} f_{gi} \geq z_g \; \forall g \in G \quad (11)$$

$$\sum_{g \in G} z_g = N_g \quad (12)$$

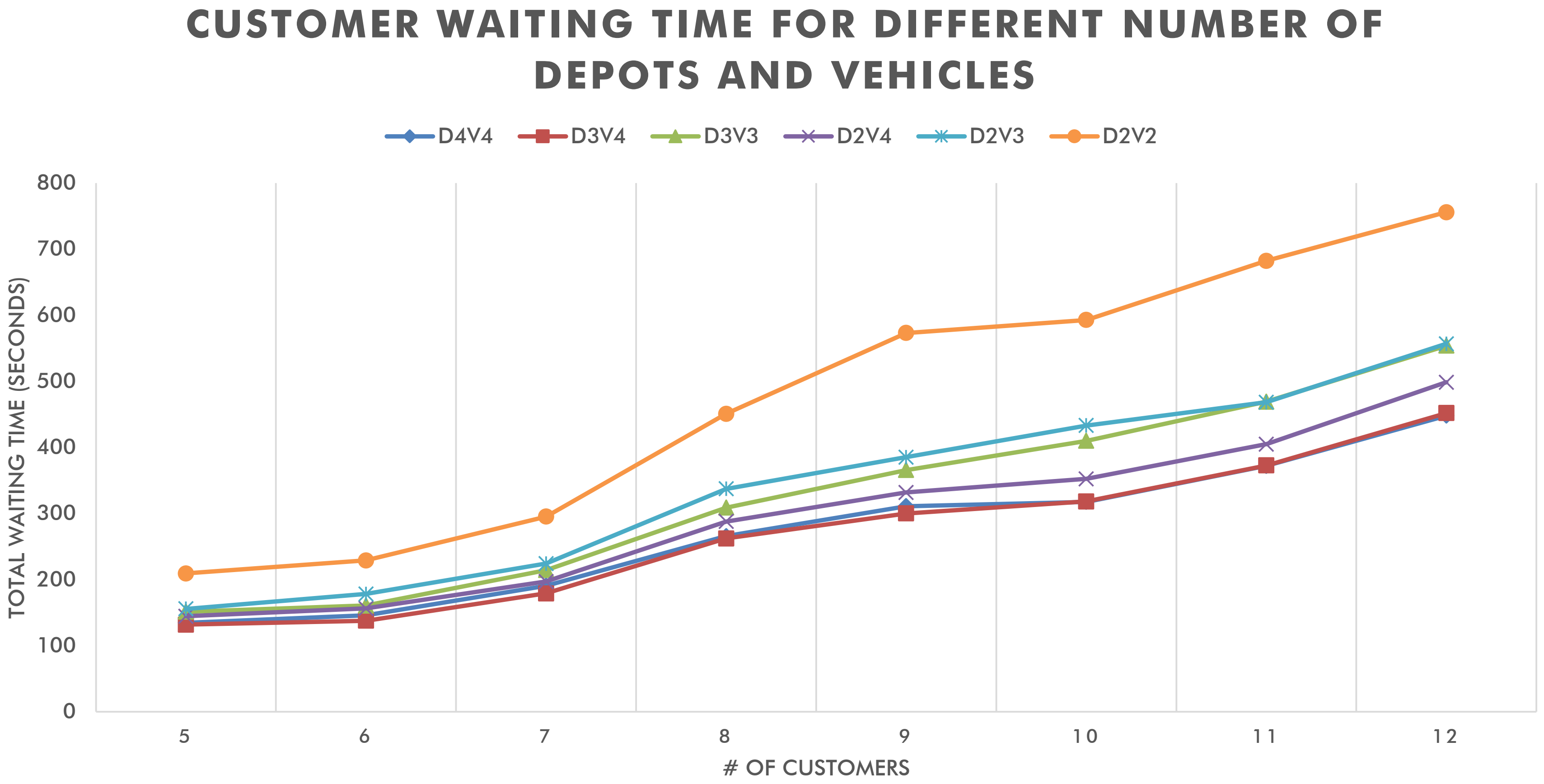
$$t_i^k \geq 0, \forall i \in V, \forall k \in K$$

$$z_g \in \{0,1\} \; \forall g \in G \quad (13)$$

$$x_{ij}^k \in \{0,1\} \; \forall i \in j$$

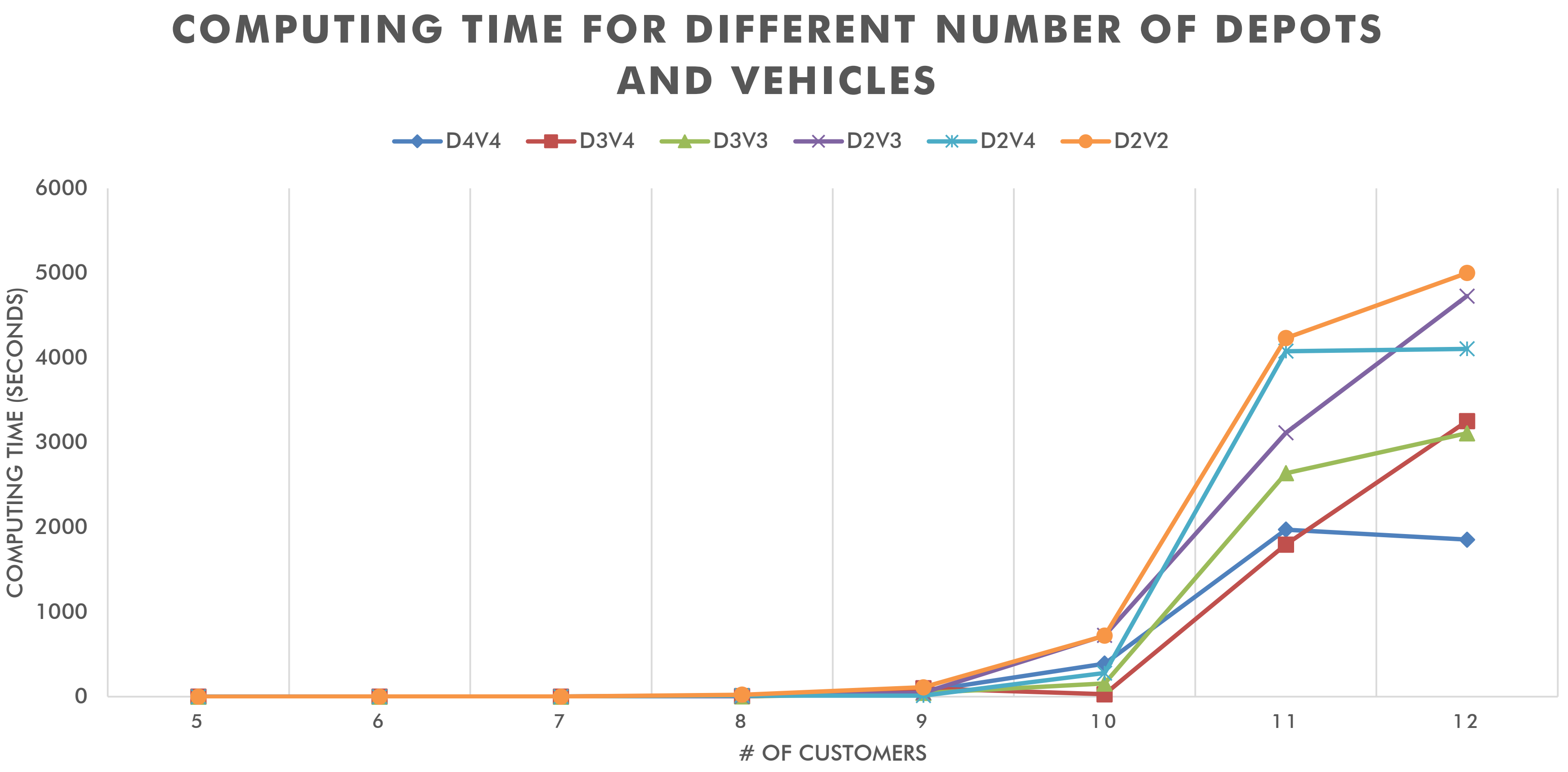
Computational Results

Mathematical Model



Computational Results

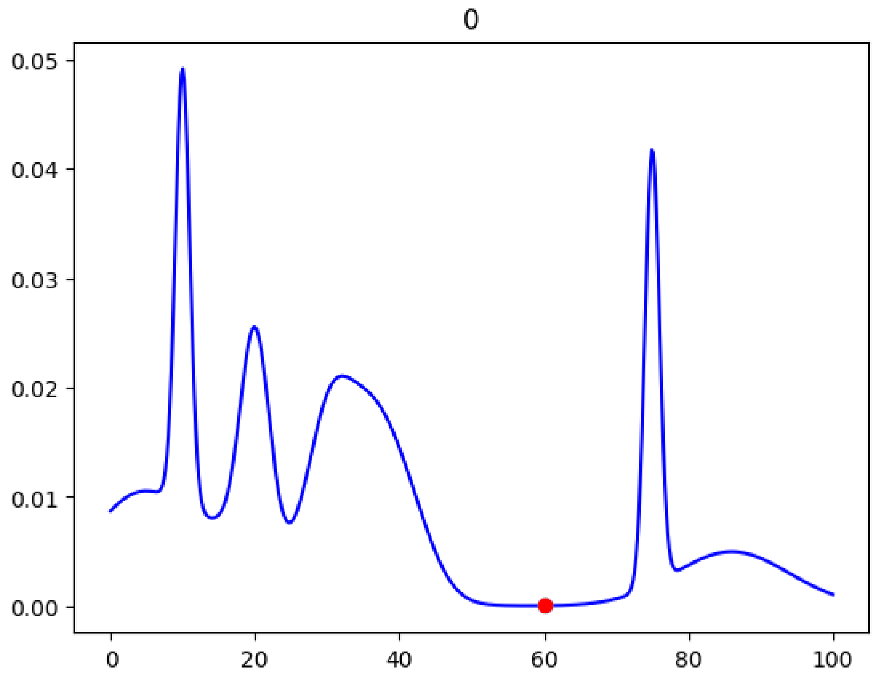
Mathematical Model



Heuristic Algorithm

Simulated Annealing

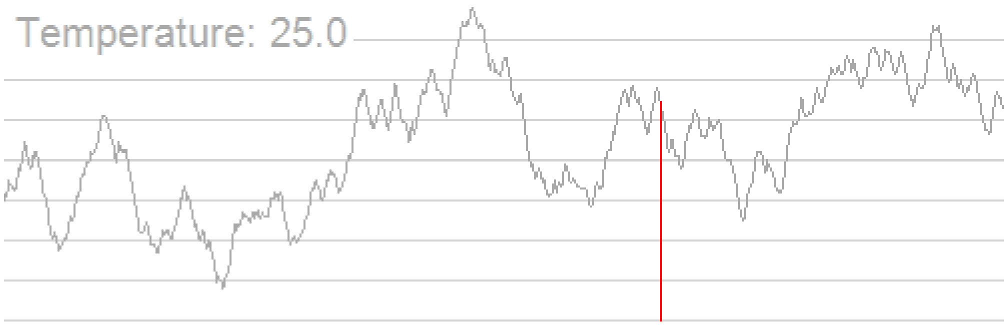
- Inspired by annealing process
- Accepts worse solutions initially in order to leave the local optimum



Simulated Annealing

Parameters

- Initial Temperature, T_0
- Final Temperature, T_f
- Cooling Rate, α
- Boltzmann Constant, K



$e^{-\Delta/(KT)}$ **= number between 0 and 1.**

Simulated Annealing (cont.)

Initial Solution

- Nearest Neighborhood algorithm with probabilistic centrality

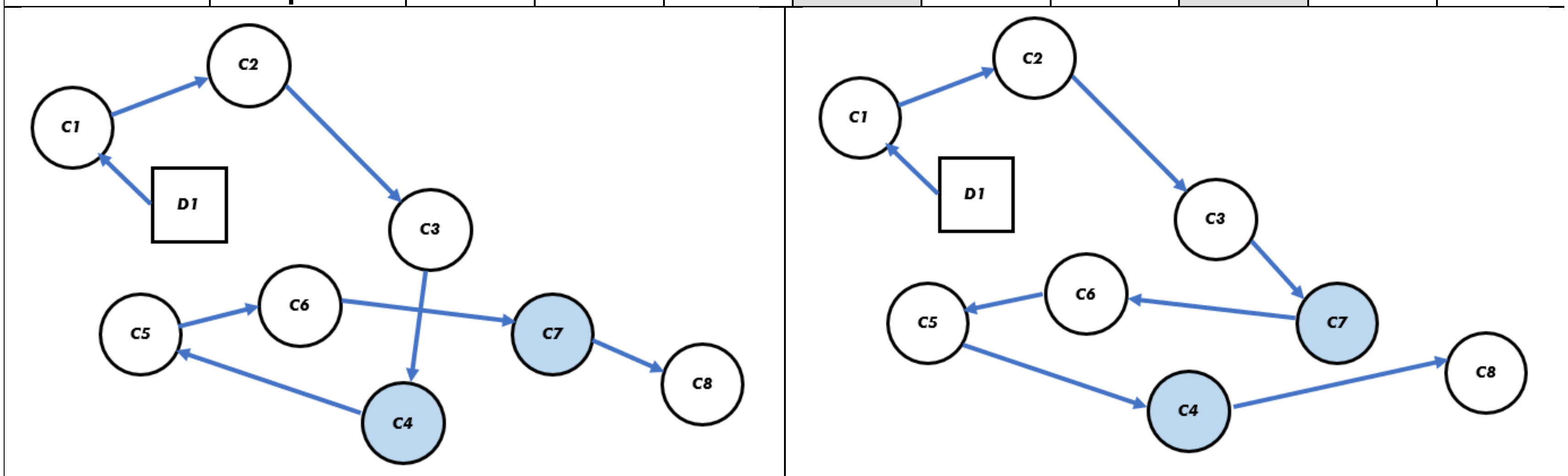
Operators

- Local operators: local insertion, local swap, flip
- Non-local operators: non-local insertion, non-local swap

Simulated Annealing (cont.)

Local Operator Example—Flip

Vehicle 1	Depot 1	C1	C2	C3	***C4***	C5	C6	***C7***	C8	-

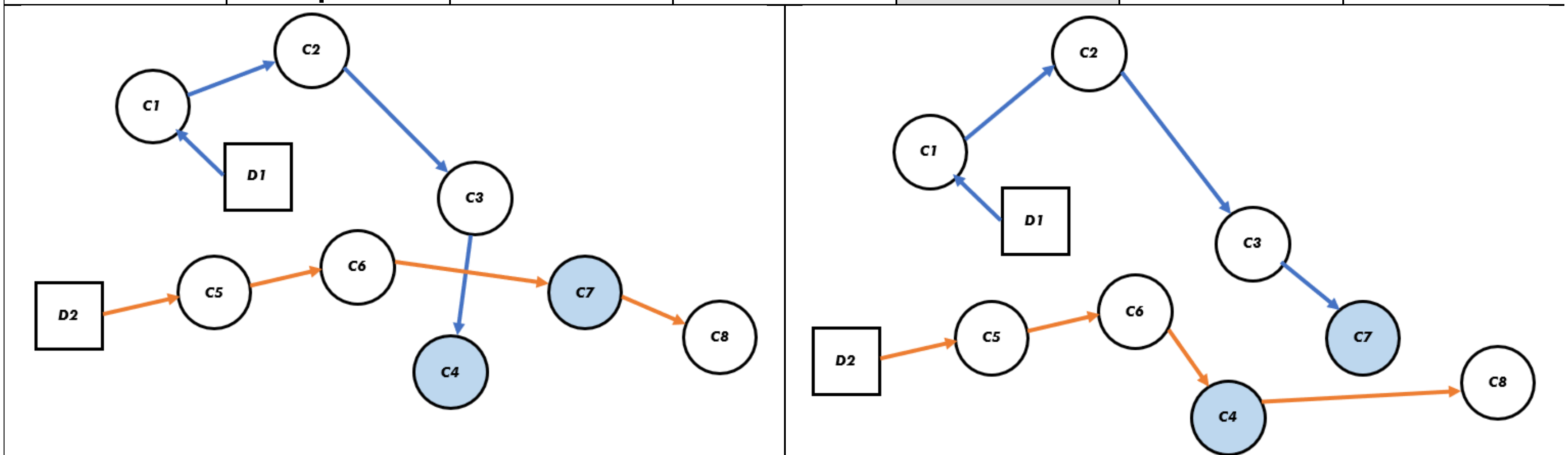


Vehicle 1	Depot 1	C1	C2	C3	***C7***	C6	C5	***C4***	C8	-

Simulated Annealing (cont.)

Non-Local Operator Example—Non-local Swap

Vehicle 1	Depot 1	C1	C2	C3	***C4***	-
Vehicle 2	Depot 2	C5	C6	***C7***	C8	-



Vehicle 1	Depot 1	C1	C2	C3	***C7***	-
Vehicle 2	Depot 2	C5	C6	***C4***	C8	-

Simulated Annealing (cont.)

Three Different Types of Simulated Annealing

- Adaptive Simulated Annealing (SA1)
  - Uses all local operators within each prime operator and give preference to better performing prime operators
- Sequential Simulated Annealing (SA2)
  - Uses all five operators in a row
- Iterative Simulated Annealing (SA3)
  - Uses all local operators within each non-local operator

Computational Results

Simulated Annealing– Prins et al. Benchmark Test

Problem instance	Customers	Vehicle Capacity	Depot Capacity	Depots	Vehicles	Open Depots	LLRP Best	SA1 Best	Average	Average Time
20-5-1	20	70	140	5	5	3	331.9	337.6	383.8	11.1
20-5-1b	20	150	300	5	3	2	608.1	608.1	647.4	6.6
20-5-2	20	70	70/140	5	5	3	304.8	302.0	333.9	8.7
20-5-2b	20	150	150/300	5	3	2	486.5	486.5	511.1	6.0
50-5-1	50	70	420/350	5	12	3	859.9	958.1	1,164.7	26.7
50-5-1b	50	150	420/350	5	6	2	1,330.2	1,326.3	1,479.8	32.2
50-5-2	50	70	350	5	12	3	723.4	822.4	826.1	23.8
50-5-2b	50	150	350	5	6	3	965.7	967.2	1,105.6	30.4
50-5-2BIS	50	70	350	5	12	3	955.2	987.1	1,127.4	28.1
50-5-2bBIS	50	150	300	5	6	3	811.8	868.8	1,050.0	46.6
50-5-3	50	70	350/420	5	12	2	848.1	948.7	957.4	26.9
100-5-1	100	70	700/770	5	24	3	2,030.9	-	-	118.6
100-5-1b	100	150	700/770	5	12	3	2,374.9	2,442.2	2,634.9	148.9

Computational Results

Iterative Simulated Annealing vs Mathematical Model

Problem Instance	Customers	Vehicle Capacity	Depot Capacity	Depots	Vehicles	Open Depots	Math Best	Math Time (seconds)	SA3 Best	Average	SA3 Time (seconds)
10-4-4-1	10	70	140	5	4	4	335.85	201.73	**335.85**	350.70	3.16
10-4-4-2	10	70	140	5	4	4	311.36	18.93	**311.36**	323.29	3.69
10-4-4-3	10	70	140	5	4	4	229.18	16.43	**229.17**	241.53	3.18
10-4-4-4	10	70	140	5	4	4	300.48	105.01	**300.48**	305.68	3.17
10-4-4-5	10	70	140	5	4	4	410.80	1615.89	**410.80**	410.80	3.56
11-4-4-1	11	70	140	5	4	4	431.02	296.37	**431.02**	435.33	3.86
11-4-4-2	11	70	140	5	4	4	409.13	2732.94	**409.13**	455.73	3.77
11-4-4-3	11	70	140	5	4	4	347.75	1582.21	**347.75**	351.83	4.31
11-4-4-4	11	70	140	5	4	4	**352.06**	5000.00	354.27	361.43	4.21
11-4-4-5	11	70	140	5	4	4	320.14	252.46	**320.14**	326.05	4.26
12-4-4-1	12	70	140	5	4	4	**502.76**	2506.55	510.50	511.26	5.07
12-4-4-2	12	70	140	5	4	4	419.54	1378.80	**419.54**	421.57	4.69

Conclusion

- Minimize total customer waiting time instead of cost
- Heuristic algorithm is necessary to solve large-sized problems
- All three simulated annealing algorithms perform competitively with the algorithms in the literature and the mathematical model