Operations Research II: Algorithms

Case Study: Service Facility Location Selection

Ling-Chieh Kung

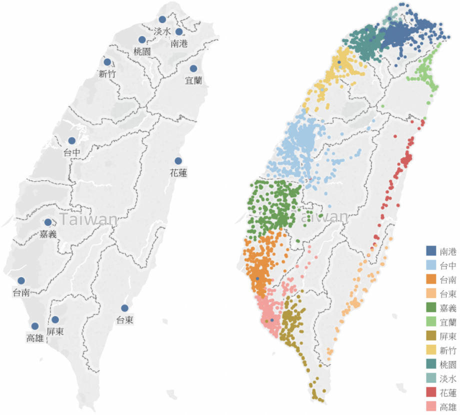
Department of Information Management
National Taiwan University

Road map

- **Background, motivation, and research objective**.
- Model formulation.
- Results.
- A heuristic algorithm.

Background

- NEC Taiwan, established in 1983, provides integrated IT/IS services to many customers.
  - Taiwan Railways, 7-11, post offices, etc.
- After-sales technical services are needed.
- As of 2017, it operates twelve service facilities in Taiwan to host more than 140 engineers to serve more than 14,000 customer sites.



After-sales service costs a lot

Facility	Service cost	Office rent	Utility fee	Total
Nangang	$9,671,712	$30,086,232	$7,607,526	$47,365,470
Tamsui	$1,611,300	$1,553,897	$207,843	$3,373,040
Taoyuan	$2,715,840	$1,314,286	$149,107	$4,179,233
Hsinchu	$2,926,584	$3,255,720	$886,615	$7,068,919
Taichung	$7,668,960	$4,861,392	$2,075,182	$14,605,534
Chiayi	$2,831,340	$1,351,896	$317,184	$4,500,420
Tainan	$3,676,620	$2,454,120	$614,917	$6,745,657
Kaohsiung	$3,159,336	$4,671,624	$1,930,277	$9,761,237
Pingtung	$1,619,364	$378,840	$194,952	$2,193,156
Yilan	$1,188,036	$864,720	$207,648	$2,260,404
Huanlien	$1,722,012	$681,792	$328,176	$2,731,980
Taitung	$1,007,124	$432,000	$2,832	$1,441,956
Total	$39,798,228	$51,906,519	$14,522,259	$106,227,006

Motivation

- Is operating twelve facilities the most **efficient** solution?
  - Fewer facilities: lower operating cost but higher service cost.
  - More facilities: higher operating cost but lower service cost.
- Whether to shut down a facilities must be determined by considering **all** facilities, customers, and engineers.
- The environment is always changing:
  - There may be more (or fewer) customers.
  - The required frequency for visiting a customer may be higher or lower.
- We need a suggestion about **facility location** selection:
  - It should find a balance between the operating and service costs.
  - It should be able to generate new suggestions after the environment changes.
- A **model** can be helpful.

Research objective

- In this study, we plan to build a **mathematical model** to formulate the facility location problem.
- We will also design a **heuristic algorithm** to solve this problem.
- Application 1: **Optimization for the current environment**.
- For the current 14,766 customers and 142 engineers, should we keep all the twelve facilities? If not, which facilities should we shut down? How to reallocate engineers and reassign customers?
- Application 2: **Cost estimation facing new business chance**.
- When there is a potential customer, is obtaining this customer profitable? This depends on the (additional) cost required to serve this customer. Using the model, we may fix the facility locations and adjust engineer allocation and customer assignment to estimate the cost.

Road map

- Background, motivation, and research objective.
- **Modeling**.
  - Conceptual model.
  - Mathematical model.
  - Computer model.
- Results.
- A heuristic algorithm.

Three levels of modeling

- As always, we solve a problem by creating three models one by one.
- **Conceptual model**:
  - Problem description in words that a manager may understand.
  - Typically is not precise enough for one to build a computer model.
- **Mathematical model**:
  - All those things that we have taught you in Operations Research I.
  - Hard to be understood by a traditional manager.
  - Precise enough for one to build a computer model.
- **Computer model**:
  - A concrete program that generates a solution.
  - Hard to be built without a mathematical model.
  - Can be an Excel Solver spreadsheet, a Python program invoking Gurobi Optimizer, an implementation of a heuristic algorithm, etc.

Conceptual model: decisions and the objective

- What **decisions** should we make?
  - For each given location, determine whether there should be a **facility**.
  - If there should be one, determine its **scale**.
  - Allocate **engineers** to built facilities.
  - Assign all **customers** to built facilities considering capacity (maximum possible number of services per year).
- Objective: **cost minimization**.
  - **Office rents**.
  - **Traveling cost** paid to subsidize engineers to move from facilities to customer sites.
  - **Utility costs** for hosting engineers in facilities.

Conceptual model: constraints

- At most one facility may be built at one candidate location.
- For a built facility, exactly one scale level must be chosen.
- A facility must be assigned to serve a customer site and get engineers allocated only if it is built.
- For each facility, the allocated engineers should be enough to serve the assigned customers.
- For each facility, the number of engineers that it may host is limited according to its scale level.

Mathematical model: sets, indices, and variables

- Let I be the set of customers, J be the set of candidate locations (or facilities), and K be the set of scale levels.
  - A customer in our formulation means a customer site in practice.
- Let $i \in I$ represent a customer, $j \in J$ represent a facility location (or a facility), and $k \in K$ represent a facility scale level.
- Decision variables:

Variable	Description
x_{jk}	$x_{jk} = 1$ if a facility is built at location j with scale level k or 0 otherwise.
y_{ij}	$y_{ij} = 1$ if customer i is served by site j or 0 otherwise.
w_j	Number of engineers allocated in location j.

Mathematical model: parameters

Parameter	Description
f_{jk}	Annual office rent of location j with scale level k.
h_i	Annual number of services needed for customer i.
d_{ij}	Cost per service for an engineer to travel between facility j to customer i.
c_j	Cost for hosting one engineer in location j.
m_{jk}	Maximum number of engineers that may be hosted in facility j with scale level k.
s	Number of services that an engineer may complete in a year.
a_{ij}	$a_{ij} = 1$ if location j and customer i are "close enough."

Mathematical model: objective function

- Our goal is to **minimize the total cost**, i.e.,

$$\min \quad \underbrace{\sum_{j\in J}\sum_{k\in K} f_{jk}x_{jk}}_{\text{office rent}} + \underbrace{\sum_{i\in I}\sum_{j\in J} h_i d_{ij} y_{ij}}_{\text{traveling cost}} + \underbrace{\sum_{j\in J} c_j w_j}_{\substack{\text{engineer}\\ \text{hosting cost}}} .$$

Mathematical model: constraints

- For each location, we may choose at most one scale level (choosing nothing means not building a facility):

$$\sum_{k \in K} x_{jk} \leq 1 \quad \forall j \in J.$$

- For each customer and each location, only a built facility may serve customers:

$$y_{ij} \leq \sum_{k \in K} x_{jk} \quad \forall i \in I, j \in J.$$

- For each customer, it must be served by a close enough facility:

$$\sum_{j \in J} a_{ij} y_{ij} = 1 \quad i \in I.$$

Mathematical model: constraints

- For each facility, the number of engineers allocated to it cannot exceed its capacity:

$$w_j \leq \sum_{k \in K} m_{jk} x_{jk} \quad \forall j \in J.$$

- For each facility, the number of engineers allocated to it should be enough to complete the services assigned to it:

$$s w_j \geq \sum_{i \in I} h_i y_{ij} \quad \forall j \in J.$$

- All decision variables must be binary:

$$x_{jk} \in \{0, 1\} \quad \forall j \in J, k \in K,$$
$$y_{ij} \in \{0, 1\} \quad \forall i \in I, j \in J.$$

Computer models

- We write Python to invoke Gurobi Optimizer for finding an optimal solution.
- MS Excel Solver is typically not enough for large-scale real problems.

Road map

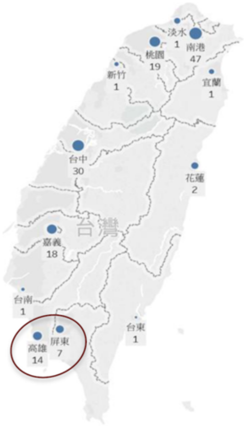
- Background, motivation, and research objective.
- Modeling.
- **Results**.
- A heuristic algorithm.

Research result

Location	Scale	Engineers allocated	Customer assigned	Services assigned
Nangang	Large	47	4,997	37,784
Danshui	Small	1	236	3,015
Taoyuan	Large	19	2,052	16,620
Hsinchu	Small	1	284	3,015
Taichung	Large	30	2,600	24,935
Chiayi	Large	18	1,495	15,708
Tainan	Small	1	323	3,015
Kaohsiung	Large	14	1,297	12,841
Pingtung	Large	7	694	7,020
Yilan	Small	1	305	3,015
Hualien	Large	2	278	3,164
Taitung	Small	1	205	2,258
Total	Large 7 small 5	142	14,766	132,389

Interpretations

- **Downsizing** to small facilities is helpful.
- Still having large facilities in both Kaohsiung and Pingtung.
  - The office rent for the Pingtung facility is so low.



Location	Office rent
Nangang	\$30,086,232
Danshui	\$1,553,897
Taoyuan	\$1,314,286
Hsinchu	\$3,255,720
Taichung	\$4,861,392
Chiayi	\$1,351,896
Tainan	\$2,454,120
Kaohsiung	\$4,671,624
Pingtung	\$378,840
Yilan	\$864,720
Hualien	\$681,792
Taitung	\$432,000

Discussions

- Benefits brought by the suggestion:
  - Saving \$7,889,650 (7.43% of the current total cost) per year.
- Potential detriments:
  - Subsidies are needed for engineer reallocation.
  - Maintaining customer relationship may be a challenge.
  - It is uneasy for one single engineer to have all maintenance skills.
- The model provides the company a way to quantify the potential benefit.
  - In the current environment.
  - In future environments.

Road map

- Background, motivation, and research objective.
- Modeling.
- Results.
- **A heuristic algorithm**.

A greedy downsizing strategy

- An MIP is good, but may not be enough.
  - A heuristic algorithm that finds a near-optimal solution is valuable.
- The heuristic algorithm is a naïve **greedy** algorithm.
  - We start from a naïve solution: Opening all facilities at the largest scale level.
  - In each iteration, we then **greedily** try to improve the current solution by **downsizing** exactly **one** facility by one scale level.
  - If there are multiple facilities whose downsizing reduces the total cost, we choose the one that brings in the **largest reduction**.
  - We keep iterating until no further feasible downsizing may reduce the total cost.

A greedy evaluation process

- Given a construction plan that determines the scale level of each facility, we need to **evaluate its feasibility and cost**.
  - We need to have a way to assign customers and allocate engineers.
- To assign customers to open facilities:
  - We first **list** all customers (in which way?).
  - If for any customer there is no open facility that is **close enough**, conclude that the current construction plan is infeasible.
  - Otherwise, for customer i, we assign it to the **closest open** facility j^* that has **enough capacity** (i.e., adding h_i to the facility does not make the total number of assigned services exceed $sm_{j^*,k}$, where k is its current scale level.)
  - Whenever one more **engineer is needed** for a facility, add one there.

Pseudocode for facility downsizing

$x \leftarrow 0$. $x_{j,|K|} \leftarrow 1$ for all $j \in J$. $\pi^* \leftarrow f(x)$.
while true do
  $\bar{\pi} \leftarrow \pi^*$.
  for each $j \in J$ **do**
    Let k' be the current scale level of facility j.
    $x' \leftarrow x$. $x'_{j,k'-1} \leftarrow 1$. $x'_{j,k'} \leftarrow 0$.
    $\pi' \leftarrow f(x')$. // $f(x')$ returns the cost or ∞ if x' is infeasible
    if $\pi' < \bar{\pi}$ **then**
      $\bar{\pi} \leftarrow \pi'$. $j^* \leftarrow j$, $k^* \leftarrow k'$.
    end if
  end for
  if $\bar{\pi} < \pi^*$ **then**
    $x_{j^*,k^*-1} \leftarrow 1$. $x_{j^*,k^*} \leftarrow 0$. $\pi^* \leftarrow \bar{\pi}$.
  end if
end while

Performance evaluation

Location	Optimal	Heuristic	Current
Nangang	Large	Large	Large
Danshui	Small	Small	Large
Taoyuan	Large	Large	Large
Hsinchu	Small	Large	Large
Taichung	Large	Large	Large
Chiayi	Large	Large	Large
Tainan	Small	Large	Large
Kaohsiung	Large	Large	Large
Pingtung	Large	Small	Large
Yilan	Small	Small	Large
Hualien	Large	Small	Large
Taitung	Small	Small	Large
Total Cost	\$98,337,356	\$104,116,624	\$106,227,006