Hub and Spoke Network Design for a Fast-Growth Company

by

Canh Phan Xuan
Bachelor’s in Economics, Foreign Trade University, Vietnam
and
Benjamin Sanford
MBA, University of Southern Mississippi, USA
and
Daniel Golenbock
MBA, Quinnipiac University, USA

Submitted to the Zaragoza Logistics Center in Partial Fulfillment of the Requirements for the Degree of

MASTER OF ENGINEERING IN LOGISTICS AND SUPPLY CHAIN MANAGEMENT

in the

MIT-ZARAGOZA INTERNATIONAL LOGISTICS PROGRAM

at the

ZARAGOZA LOGISTICS CENTER,
A RESEARCH INSTITUTE ASSOCIATED WITH THE UNIVERSITY OF ZARAGOZA

May 2018

© 2018. All rights reserved.

The author hereby grants to MIT and to the Zaragoza Logistics Center permission to reproduce and to distribute publicly paper and electronic copies of this thesis document in whole or in part.

Signature of Authors

Mit-Zaragoza International Logistics Program
Zaragoza Logistics Center
May 18th 2018

Certified by

Rafael Díaz, PhD
Professor of Supply Chain Management
Thesis Supervisor
Accepted by

Susana Val, PhD
Director, Research Institute associated with the University of Zaragoza – Zaragoza Logistics Center
Hub and Spoke Network Design for a Fast-Growth Company

by

Canh Phan Xuan

and

Benjamin Sanford

and

Daniel Golenbock

Submitted to the Zaragoza Logistics Center in Partial Fulfillment of the Requirements for the Degree of

MASTER OF ENGINEERING IN LOGISTICS AND SUPPLY CHAIN MANAGEMENT

in the

MIT-ZARAGOZA INTERNATIONAL LOGISTICS PROGRAM

at the

ZARAGOZA LOGISTICS CENTER, A RESEARCH INSTITUTE ASSOCIATED WITH THE UNIVERSITY OF ZARAGOZA

ABSTRACT

As an ever-increasing number of e-commerce companies come into existence, express delivery companies must rely heavily on efficient expansion of their network to keep up with growing demand and lower margins. The objective of this paper is to provide a framework for expanding the role of network design to fulfill growth demands within the express delivery industry.

This paper presents a multi-period optimization model applying a mixed linear programming methodology to build a network roadmap within a specific express delivery company’s primary market. The research explores multiple techniques for aggregating customer demand by implementing and validating center of gravity, k-means clustering, and driving distance matrices. In order to validate the robustness of this optimal network, a simulation was implemented describing a thousand of scenarios and cost models.

As the company continues to grow, the framework described in this research can be developed and customized to handle complex cost structures and can be replicated in all of its markets.
TABLE OF CONTENTS

LIST OF FIGURES ................................................................................................................................................. 6
LIST OF TABLES ......................................................................................................................................................... 7

1. Introduction .......................................................................................................................................................... 8
   1.1. Project overview ............................................................................................................................................. 8
   1.2. Company overview ....................................................................................................................................... 8
   1.3. Brief of operations network and process ................................................................................................. 9

2. Motivation .......................................................................................................................................................... 10

3. Literature Review ............................................................................................................................................... 11
   3.1. Network design ............................................................................................................................................ 11
   3.2. Customer demand aggregation ................................................................................................................ 13

4. Methodology ...................................................................................................................................................... 14
   4.1. Collecting and initial analyzing of data ...................................................................................................... 15
   4.2. Model input .................................................................................................................................................. 15
       4.2.1. Customer demand points .................................................................................................................. 15
       4.2.2. Customer demand forecasting and allocation by week .................................................................. 22
       4.2.3. Outbound transportation cost ......................................................................................................... 23
       4.2.4. Fixed hub cost, potential hub locations, and hub capacity ............................................................ 23
   4.3. Modeling and Network Model design ....................................................................................................... 23
   4.4. Model Formulation ..................................................................................................................................... 24
       4.4.1. Hub network optimization ................................................................................................................. 24
       4.4.2. Hub sizing optimization model ....................................................................................................... 25

5. Interpretation & Results .................................................................................................................................. 27
   5.1. Baseline model ............................................................................................................................................ 27
   5.2. Model Result and analysis ....................................................................................................................... 28
   5.3. Simulation ................................................................................................................................................... 32

6. Conclusion ......................................................................................................................................................... 34
   6.1. Reflection .................................................................................................................................................... 34
   6.2. Recommendations ...................................................................................................................................... 35
   6.3. Future improvements .................................................................................................................................. 35
   6.4. Discussion ................................................................................................................................................... 36
6.4.1. Limitations ................................................................. 36
6.4.2. Delimitation ............................................................. 36
6.5. Final thoughts .............................................................. 37

BIBLIOGRAPHY .................................................................. 38
APPENDIX ........................................................................ 39
Clustering driving distance ..................................................... 39
LIST OF FIGURES

Figure 1: Actual number of packages and forecasted in next 5 quarters................................................. 9
Figure 2: The company operations network.................................................................................................. 10
Figure 3: Framework for Network Design Decisions – Adapted from (Chopra & Meindl, 2016). 11
Figure 4: Taxonomy of location models......................................................................................................... 12
Figure 5: Methodology summary.................................................................................................................. 15
Figure 6: 7,600 shipments (400 from each 19 districts) plotted with the size of the point commensurate with the package weight .............................................................................................................. 17
Figure 7: COGs plotted on a map of Ho Chi Minh city, with each district labeled ............................... 18
Figure 8: Greenfield and Brownfield Center of Gravity Clustering Points ................................................. 18
Figure 9: 5,000 randomly selected shipments plotted with the size of the point commensurate with the package weight ................................................................................................................................. 20
Figure 10: Greenfield and Brownfield Driving Distance Clustering Points ............................................... 22
Figure 11: Orders by week .............................................................................................................................. 27
LIST OF TABLES

Table 1: Population vs Sample Variance ................................................................. 19
Table 2: Set of Indices - Hub network optimization .................................................. 24
Table 3: Set of Variables - Hub network optimization ............................................... 24
Table 4: Set of Parameters - Hub network optimization ............................................ 25
Table 5: Set of Indices - Hub size optimization ....................................................... 26
Table 6: Set of Variables - Hub size optimization ..................................................... 26
Table 7: Set of Parameters - Hub size optimization .................................................. 26
Table 8: Comparison between baseline and optimal solution .................................... 28
Table 9: Order flow: optimal and baseline model ..................................................... 28
Table 10: Sample distance between hub and customer demand .................................. 29
Table 11: Hub Costs – COG Brownfield ................................................................. 29
Table 12: Matrix to compare the optimal solutions .................................................. 30
Table 13: Hub selection road map & utilization rate over the next 3 years .................... 31
Table 14: Hub size optimized .................................................................................. 32
Table 15: Simulation Results ................................................................................... 33
Table 16: Average Hub Capacity Utilization (%) ...................................................... 33
1. Introduction

1.1. Project overview

In striving to meet the demands of a fast-growing industry while maintaining superior service levels, it is not enough to rely on experience and instinct when selecting facility locations. Network design is of paramount importance. Placing distribution centers in suboptimal locations will invariably lead to under- or inefficient utilization of the resources at these locations. By accurately forecasting demand and tying projections to network design decisions, companies can optimize the utilization of their resources.

In many cases companies make network design decisions based on accessibility or intuition. By first evaluating these important decisions with a relatively simple framework, companies can position themselves to make better decisions based on sound reasoning, allowing them to defend their capital expenses to stakeholders. This framework consists of six steps.

1. Data collection and cleaning
2. Customer demand aggregation
3. Building a baseline model
4. Building an optimization model – comparison to baseline
5. Multiperiod optimization to account for growth
6. Analysis and interpretation

A common drawback when making network design decisions derives from the fact that traditional clustering algorithms do not reflect real world conditions. When designing a network within a relatively small radius, straight line and curved distances do not approximate the true physical geographical distance and efficiency of the network. When physical boundaries and urban features such as traffic are taken into account what once was an optimal distribution network becomes counterproductive. For these reasons it is critical that driving distance between distribution centers and demand clusters (and by extension the circuity of the region) be considered.

After the hub locations have been selected a simulation can be performed highlighting the effects that demand variability has on the network. Although the ideal hub locations have been determined, the optimal time to open a new hub to meet increased demand must still be confirmed. By running a simulation that varies both cost and demand parameters, the company will be able to reflect on not only which hubs are being opened most frequently but on how efficiently they are being utilized as well.

1.2. Company overview

The sponsoring company, a startup established six years ago, operates primarily in express delivery in Vietnam. In the past three years, it has enjoyed rapid growth of 2-300% CAGR from inception to present day by taking advantage of the booming e-commerce market in Vietnam (Figure 1). The company has a network that includes the two largest sorting centers in
the northern and southern parts of the country, and approximately 200 hubs and post offices in Vietnam (Company Data, 2018). Employing a customer-centric strategy, the company has transformed from an unknown brand to one of the largest companies in the highly competitive express delivery industry formerly dominated by two State Owned Enterprises (SOE). By delivering an average of 120,000 parcels per day in the past six months, it officially surpassed the second largest SOE company to become the second largest express company in Vietnam.

More than 80% of the company’s revenue is contributed by top e-commerce players. This means that it faces an enormous challenge to simultaneously keep pace with very fast-growing demand and very high demand variability from its customers. In 2017, for instance, the peak daily order volume surpassed 250,000 packages, four times the average daily volume.

The objective of this research project is to design the most efficient delivery hub network in the company’s largest market in preparation for the next three years of growth.

1.3. Brief of operations network and process

The company’s network consists of three main facility types: sorting centers, hubs, and post offices (Figure 2). In this research, we will not consider the post office facility as its primary role is as a drop-off point for individual senders and small merchants. A parcel’s life cycle typically flows through the following five steps: (1) Pick-up, (2) Storage & Transiting, (3) Sorting, (4) Transiting to hub, and (5) Last-leg delivery. In this research, the term “hub” will refer to the final consolidation point from which parcels are delivered to customers. Though perhaps confusing from an academic viewpoint, this term corresponds to the sponsoring company’s internal usage. Upon transiting from a pick-up hub or post office, an order is placed into the sorting center where it is allocated to a specific and planned hub defined by the software system. This order is then
sorted into a “big bag” pre-assigned to a specific route and driver before being transited to an assigned hub. From there, the order is assigned to a delivery employee or a freelancer to deliver to a receiver.

Figure 2: The company operations network

2. Motivation

Despite the fact that the company has strong operational management capabilities (it has been able to keep up with not only the fast-growing order volume but also the higher service levels required by big e-commerce players), it has struggled to develop the processes necessary to attain a sustainable competitive advantage. Because of the effort required to meet its high SLAs as well as the enormous scalability in order volumes, the company has not been able to dedicate the proper effort to develop a long-term facility network. This design model is critical to establish competitive advantage and sustain business growth. The facility location problem has been recognized widely by academics and practitioners. As Daskin (2013) stated: “The success or failure of both private and public sector facilities depends in part on the locations chosen for those facilities”.

Taniguchi (2008) explained that this century is going to be an era of urbanization in which the number of people living in cities will surpass the rural population (United Nations Population Fund, 2007). Ho Chi Minh City, the largest city of Vietnam, contributes up to 40% of the total delivery volume of the company. Successfully designing a last-mile logistics network in this city not only contributes to the company’s market share in southern Vietnam (contributing 60%
revenue) but also becomes a replicable model to apply in the second largest market in the north of the country.

Consequently, it was agreed that the primary focus of research should be to design an optimal facility network of hub locations as well as planned capacity for the entire network over a three-year period. Our research question was broken into four smaller questions:

1. **How many** hub facilities should be established?
2. **When** should the new hubs be opened?
3. **Where** should the new hubs be situated?
4. **How much** capacity should each hub support?

3. Literature Review

3.1. Network design

There is a large body of research dedicated to network design problems. Supply chain network design has four main phases (Chopra & Meindl, 2016) (Figure 3). Despite the fact that this framework was developed for a global supply chain network, there is a generic approach in this framework that permits its application without considering factors related to global scale. In applying it to this case, tariffs and tax incentives; exchange rate; and political risk will not be considered.

![Figure 3: Framework for Network Design Decisions — Adapted from (Chopra & Meindl, 2016)](image)

Before building an optimization model (phase IV above), the problem must be elaborately classified into a specific type as laid out in Figure 4 below (Daskin M. S., 2013).
There are multiple ways to model the facility network. These include linear programming, stochastic programming, and robust optimization. Mixed integer programming (MIP), a part of the linear programming category, provides a thorough approach to location problems and is widely applied in network design. A benefit of this model is the very important assumption of certainty in data inputs including costs, demand, and capacity (Shapiro, 2001).

Within the scope of this project, uncertainty is the overriding concern. Mitigating this risk, the following two modeling techniques are considered.

The first approach is to conduct sensitivity analysis and construct multiple scenarios of an uncertain future, optimizing in each scenario (Shapiro, 2001). Unfortunately, this approach merely provides an optimal point in one specific scenario without considering scalable demand and seasonality. As a result, it could lead to a scenario, for example, in which the model’s recommendation is to close a facility that might be needed during the next peak season. Under another scenario, a model that does not consider growth could suggest a local or short-term optimal location. These challenges posed by the sponsoring company cannot be solved by a classic MIP. Rather than using a static input, the model must incorporate multiple time periods that are categorized into dynamic location problems (Daskin M. S., 2013). According to multiple bodies of research, this approach is appropriate not only in dealing with periodic demand Osleeb & Ratick (1990) but also in finding an optimal evolution of network configuration over time (Daskin, Hopp, & Medina, 1992). According to these researchers, however, in some cases the best option is to be adaptive and to find an optimal solution in the first period rather than all future periods. This adaptiveness is useful in practice as demand is stochastic.

The second approach is to incorporate a randomness into the optimization model in which stochastic and robust optimization are developed to mitigate the uncertainty. Though both
techniques involve randomness, they are different not only from a mathematical approach but also in terms of the goal of each technique. On the one hand, stochastic optimization problems deal with risk where their uncertainty parameters’ values could be represented by probability distributions known by the decision maker. On the other hand, robust optimization problems’ objective is to optimize for a potential worst-case scenario where the system parameters are not only uncertain but also unable to be described by a specific probability distribution (Snyder, 2006). Moreover, these techniques, according to this research, seem to have limited application in industry as they require a cumbersome amount of data and are challenging to be solved for realistic instances. As a result, the number of successful applications of deterministic facility location models is still considerably higher in the literature.

3.2. Customer demand aggregation

In evaluating the optimality of the current network, we applied center of gravity. According to Krajewski, Ritzman, & Malhotra (2007) the center of gravity method is a good means with which to start evaluating those locations that are best in a given target area even though the coordinates given are often not optimal. Nevertheless, Murphy & Wood (2008) mention that center of gravity is not an ideal method to use as it does not take into account many real-life variables that can greatly affect cost. It is typically a very useful heuristic for approximations derived from limited data.

In order to remove the constraints engendered by the arbitrary and artificial divisions of administrative districts, we applied clustering techniques to the aggregation starting with k-means. As a centroid-based clustering technique, k-means can be used to quantify inter-location distances. Simply put, these distances are a visual and spatial representation of attribute similarity between locations. Centers of clusters are selected and used to identify locations that are closest. These locations then become associated with those clusters, decreasing the number of interactions between entities. In this case, k-means appeared unsuitable as it only looks at distance between points and no other attributes. Our coordinates are composed of latitude and longitude. There are numerous physical geographical boundaries and obstacles that exist between them. The precision of output data will decrease as the deviation in distance compared to Cartesian coordinates increases. In this case, Ho Chi Minh City is lined with numerous rivers that greatly increase the discrepancy between Cartesian coordinate distances and driving distances. As a result, we abandoned k-means as a viable option.

The next clustering method we attempted was DBSCAN. Density-Based Spatial Clustering of Applications with Noise is the most well-known density-based clustering algorithm, first introduced by (Ester, Kriegel, Sander, & Xu, 1996). Unlike in k-means, in DBSCAN the number of clusters is not input as a parameter. Instead the algorithm extrapolates the optimal number of clusters based on the data, determining clusters of random shape (whereas k-means detects spherical clusters). It clusters datasets based on two considerations: physical distance from each coordinate and minimum cluster size (Ester, Kriegel, Sander, & Xu, 1996). In other words, in the realm of spatial latitude-longitude dataset clustering, DBSCAN is a far superior algorithm to k-
means. When adding weights (volume density) the algorithm makes a spherical structure based on neighborhood. Using generalized DBSCAN allows modification of the neighborhood function. As a result, the algorithm is not limited to simplistic Euclidean distances but can make the neighborhood definition as complex as needed (Schubert, Sander, Ester, Kriegel, & Xu, 2017). Unfortunately, in this case output returned an error because the pairwise distance matrix was too large to compute using our minimum sample size of 5200 data points.

The next method attempted was agglomerative hierarchical clustering. Here each coordinate is connected by distance to its nearest neighbor to create a cluster. A series of iterations use various methods for linking remaining coordinates which are assembled into a decreasing number of clusters until finally all the coordinates reform into a single group (Hastie, Tibshirani, & Friedman, 2009). In hierarchical clustering connectivity constraints are used so that only connected samples are allowed in the same cluster. In this case latitude and longitude are used by function sklearn.neighbors.kneighbors_graph() to create the list of neighbors, and the weight variable is used in the clustering. This method proved more effective than using the connectivity constraints on the weight and clustering the x-y coordinates; however, the weight variable proved to be less predictive of cost than anticipated. Moreover, we still preferred to utilize a method that accounted for the difference between driving distance and Euclidean distance due to the danger of separating hub location from vehicle routing (Rand, 1976). As a result, we continued to research more recently implemented methodologies. Consequently, we discovered an article published September 29, 2015 by Han Shih entitled “Facility Location Decisions Based on Driving Distances on Spherical Surface” (Han, 2015). Han’s work deviates from past literature on facility location problems in that it does not base its approach on Euclidean distances, but rather optimal driving distances generated from Google Maps (with all assumptions considered including traffic density). As a result, the (often vast) difference in distance between Cartesian coordinates and driving distances is eliminated. Using Google API, driving distance (from Google Maps) between each input location and every other location is queried and input in a matrix which is then used to generate the clusters. The methodology described allowed us to account for both shipment density as well as precise driving distance. Please see the Appendix section for the full source code of our driving clustering methodology.

4. Methodology

We designed our methodology flow chart based on the literature survey discussed above (Figure 5). In step 3, a baseline model structured on the company’s historical data ensured that our model could describe the company’s current network configuration. After completing this step and receiving confirmation from the company, an optimization model was built. We identified the optimal operations solution for the existing network and compared the result with the current network. In step 5, a multi-period model was developed to recommend a roadmap for the company as it keeps pace with rapid demand growth over the next three years. Finally, in step 6, a simulation was built to evaluate the robustness of the model from an individual hub
perspective. This provides the company with a strategic plan to secure the most critical locations in the city.

![Figure 5: Methodology summary]

4.1. Collecting and initial analyzing of data
The data required to build the model is as follows:

- Facility locations: Sorting centers, hubs
- Potential facility locations
- Demand point: customers’ address and behavior, forecasted demand over next three years divided into quarters
- Transportation cost: In-transit cost and last-leg delivery cost
- Facility fixed cost
- Facilities capacity: hub

Due to the size of the data - up to 700,000 transactional data points in a month - querying and validating is a very time-consuming process. Moreover, until now changing and testing the operations model created a lot of noise in the data that also required manual cleaning. For example, one of the cases that was provided created several logic hubs, or virtual hubs sharing the same location. Working with the company’s operations manager, we were able to identify, account for and clean this data.

4.2. Model input

4.2.1. Customer demand points
As mentioned above, due to the large amount of data generated by delivery companies it made sense to narrow the geographical scope to Ho Chi Minh City. Even with this reduction, however, there were almost 700,000 shipments over the specified time period after completing the cleaning process. Data from the sponsoring company consisted of:
• The district and location from which each package was shipped
• The Hub ID
• The Shipment Date
• Package Weight
• Delivery Hubs

Power query was used to create calculated fields to determine in which week of the year each package was shipped. Due to the large amount of data it was decided that taking a representative random sample from the population would save on processing time while still giving accurate results. With the given time and computing constraints, it would have been disadvantageous to analyze more data than necessary.

Due to the fact that the study is descriptive, there is no predictor or outcome variable and differing groups are not being compared. Consequently, all that is required is to specify the desired confidence interval to arrive at our sample size (Hulley, 2001).

We used the below formula for calculating sample size \( n \):

\[
n = \frac{N 
 \times \ X}{(X + N - 1)}
\]

where,

\[
X = z_{\alpha/2} \times \frac{p \times (1 - p)}{MOE^2}
\]

and \( z_{\alpha/2} \) is the critical value of the Normal distribution at \( \alpha/2 \) (e.g. for a confidence level of 95%, \( \alpha \) is 0.05 and the critical value is 1.96), \( MOE \) is the margin of error, \( p \) is the sample proportion, and \( N \) is the population size. Note that a Finite Population Correction has been applied to the sample size formula (Select Statistical Services Limited, 2018).

A two-pronged approach was essayed. The first method was to take equal size samples of package deliveries from each district. We set our sample size at \( n=400 \). This gives a 95% confidence interval while maintaining a 5% margin of error. To obtain these random samples Excel assigned every shipment a randomly generated number. This was achieved using the “=Rand()” function. The data was then sorted from smallest to largest and data samples were taken for further analysis. Each of the 400 samples from the 19 districts were then geocoded using Google API and a VBA code in Microsoft Excel. This provided the latitude and longitude of each delivery address in order to determine the center of gravity (CoG).

Below are the formulas used to calculate CoG.

\[
\begin{align*}
C_x &= \frac{\sum x_i \times w_i}{\sum w_i} \\
C_y &= \frac{\sum y_i \times w_i}{\sum w_i}
\end{align*}
\]
These calculations were performed for each district in order to determine that district’s optimal COG based on historical shipment data. Below in Figure 6 are the 7,600 shipments (400 from each 19 districts) plotted with the size of the point commensurate with the package weight. In Figure 7 are CoGs (calculated from the below data points) plotted on a map of Ho Chi Minh City, with labels showing each district.

\[ C_X \] — Center of Gravity X coordinate
\[ C_Y \] — Center of Gravity Y coordinate
\[ X_i \] — Shipment location X coordinate
\[ Y_i \] — Shipment location Y coordinate
\[ W_i \] — Package i weight

Figure 6: 7,600 shipments (400 from each 19 districts) plotted with the size of the point commensurate with the package weight

Figure 7: CoGs plotted on a map of Ho Chi Minh City, with labels showing each district.
Figure 7: COGs plotted on a map of Ho Chi Minh city, with each district labeled

Figure 8: Greenfield and Brownfield Center of Gravity Clustering Points
With these 19 CoGs the optimization model was able to determine if any cost savings could be realized by relocating the hubs.

The next method used to obtain a data sample was a random sample from the total number of shipments. Some areas have as many as five times the shipment volume of others. These higher shipment densities should affect where the CoGs are generated when using clustering methods as discussed below.

To obtain this random sample the “=Rand()” function was once again used and the results were sorted from smallest to largest as before. A sample of n=5,000 was taken. All of these addresses were geocoded and stacked for further analysis. The proportion of shipments in the population for which each district was responsible was analyzed and compared to the proportion of the sample shipments in each district. Below is a table with these results.

<table>
<thead>
<tr>
<th>District</th>
<th>% of Population</th>
<th>% of Sample</th>
<th>% Variance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quận Thủ Đức - Hồ Chí Minh</td>
<td>8.22%</td>
<td>7.48%</td>
<td>-0.74%</td>
</tr>
<tr>
<td>Quận 7 - Hồ Chí Minh</td>
<td>7.78%</td>
<td>7.53%</td>
<td>-0.24%</td>
</tr>
<tr>
<td>Quận Tân Bình - Hồ Chí Minh</td>
<td>7.73%</td>
<td>8.40%</td>
<td>0.67%</td>
</tr>
<tr>
<td>Quận Bình Thạnh - Hồ Chí Minh</td>
<td>7.68%</td>
<td>9.22%</td>
<td>1.54%</td>
</tr>
<tr>
<td>Quận Bình Tân - Hồ Chí Minh</td>
<td>7.57%</td>
<td>7.19%</td>
<td>-0.38%</td>
</tr>
<tr>
<td>Quận Gò Vấp - Hồ Chí Minh</td>
<td>6.71%</td>
<td>6.64%</td>
<td>-0.08%</td>
</tr>
<tr>
<td>Quận 1 - Hồ Chí Minh</td>
<td>6.61%</td>
<td>7.11%</td>
<td>0.51%</td>
</tr>
<tr>
<td>Quận 12 - Hồ Chí Minh</td>
<td>6.19%</td>
<td>5.34%</td>
<td>-0.85%</td>
</tr>
<tr>
<td>Quận Tân Phú - Hồ Chí Minh</td>
<td>5.69%</td>
<td>6.39%</td>
<td>0.70%</td>
</tr>
<tr>
<td>Quận 9 - Hồ Chí Minh</td>
<td>4.95%</td>
<td>2.89%</td>
<td>-2.06%</td>
</tr>
<tr>
<td>Quận 8 - Hồ Chí Minh</td>
<td>4.82%</td>
<td>4.63%</td>
<td>-0.19%</td>
</tr>
<tr>
<td>Quận 3 - Hồ Chí Minh</td>
<td>4.16%</td>
<td>5.07%</td>
<td>0.91%</td>
</tr>
<tr>
<td>Quận 10 - Hồ Chí Minh</td>
<td>4.04%</td>
<td>4.26%</td>
<td>0.22%</td>
</tr>
<tr>
<td>Quận 2 - Hồ Chí Minh</td>
<td>3.45%</td>
<td>2.77%</td>
<td>-0.68%</td>
</tr>
<tr>
<td>Quận Phú Nhuận - Hồ Chí Minh</td>
<td>3.23%</td>
<td>3.38%</td>
<td>0.16%</td>
</tr>
<tr>
<td>Quận 5 - Hồ Chí Minh</td>
<td>3.04%</td>
<td>3.19%</td>
<td>0.15%</td>
</tr>
<tr>
<td>Quận 11 - Hồ Chí Minh</td>
<td>2.94%</td>
<td>3.35%</td>
<td>0.41%</td>
</tr>
<tr>
<td>Quận 6 - Hồ Chí Minh</td>
<td>2.87%</td>
<td>2.89%</td>
<td>0.01%</td>
</tr>
<tr>
<td>Quận 4 - Hồ Chí Minh</td>
<td>2.34%</td>
<td>2.28%</td>
<td>-0.07%</td>
</tr>
</tbody>
</table>

As evidenced, the proportions remained very similar as would be expected with an adequately large sized random sample. Below are the random 5,000 shipments plotted on a map of Ho Chi Minh City, with the circle’s radius increasing with shipment weight.
As described in the literature review, Han Shih’s “Facility Location Decisions Based on Driving Distances on Spherical Surface” provided a methodology to generate optimal driving distances from Google Maps. Using Google API, driving distance between each input location and every other location was queried and input in a matrix which was then used to generate the clusters.

The (88 X 88) 7744 points generated in the paper represent distances – a single number each – rather than locations. The algorithm works as follows: It takes each row of distances (for example, from location 1 to all the other locations' distances; in this case there are 88 per row) and clusters each row. The resulting output is clusters of locations with x, y coordinates, and not the distance values. In Appendix A (Han, 2015) we can see that the output of the algorithm is clusters of locations so if there are 88 locations as input, then the cluster results will consist of 88 locations, not 7744 distance values. In our case, we have 19 COG weighted locations with 400 random variable locations, so the output clustering result will consist of 19 locations. The clustering is being computed based on the distance, but the actual clustering is done on the location coordinates. Essentially, Han Shih’s algorithm first generates the distance matrix, and then uses sets of these values as a single entity to cluster location, so each location has 110 inputs; in our case each location has 400 inputs. The clustering is then computed with weighted volume distances on the 88 (in our case 19 X 400)-dimensional location data.
The paper differs from our project in that it is clustering 88 addresses into different clusters before finding the centroid of each cluster (these are Midwest major market cities). We have 19 COGs representing the 19 districts and 400 random addresses within the city selected from the original data set. Thus, we are creating a 19 X 400 matrix. The paper necessarily uses a symmetrical 88 X 88 matrix (Appendix B, visible up to 18 columns, Han, 2015). The paper, moreover, accounts for facility cost and transportation rate in its algorithm. We are only accounting for volume density representing the number of shipments. Our transportation rate is variable and uniform for all routes. Aside from that our calculations followed the paper very closely.

Using Python, we were able to follow the author’s algorithm as follows:

(1) Obtained valid API key from Google’s cloud platform at distance-matrix-backend.googleapis.com and input into front_url = https://maps.googleapis.com/maps/api/distancematrix/json?units=metric&origins=

(2) Input GPS coordinates data representing the centers of gravity for the volume of shipments of each of the 19 districts in Ho Chi Minh City

(3) Input GPS coordinates data representing 400 randomly selected shipments in Ho Chi Minh City

(4) Constructed empty arrays to hold the distance matrix

(5) Made API calls from Google and parsed returned data to get distance values, creating a 19*400 matrix of distances in meters.

(6) Input weight data representing the volume of shipments of the 19 COGs.

(7) Set parameter of the minimum distance that the algorithm needs to find between a centroid and its corresponding location. The smaller the distance, the more clusters are needed. min_distance = 20000; num_clusters = 19

(8) Initialized distance_data which keeps track of all the distances between centroids and their corresponding locations

(9) Prepared distance data to make sure the while loop runs at least once. The while loop dictates that if any element in the array distance_data is greater than the user input min_distance if the first iteration fails to find the right solution, then it increases the cluster number by 1 and continues to perform k-means algorithm

(10) Performed k-means clustering on the matrix_data

(11) Initialized data for calculating coordinates, distances, and centroids

(12) For loop goes through each cluster and calculates relevant data

(13) For loop identifies the appropriate GPS coordinates and weight data for a particular cluster

(14) Cluster_data and weight_date used to calculate centroids

(15) Initialized Cartesian coordinates

(16) Estimated the radius of Earth in meter @ r = 6371000
(17) Calculated x, y, z using GPS coordinates
(18) Calculated centroid coordinates
(19) Translated centroid coordinates from Cartesian back to degrees
(20) Went through each location in the cluster and calculated its distance from the centroid using Google API
(21) Data padded for distance_data array, coordinate_data, and center
(22) Exported (to Excel) the GPS coordinates based on the cluster #
(23) Tracked distance_data and center variables
(24) Increased the number of clusters for the next iteration (+= 1)
(25) Exported (to Excel) distance_data and center (distances from centroid to each location in the cluster and center locations for each cluster)

![Map of Greenfield and Brownfield Driving Distance Clustering Points](image)

**Figure 10: Greenfield and Brownfield Driving Distance Clustering Points**

### 4.2.2. Customer demand forecasting and allocation by week

According to the operations manager of the company, forecasting output should be the total number of orders going through the system during a peak week of each quarter. The first reason for this is to design in capacity for a hub; it should be able to handle the volume generated during peak demand, typically prolonged within a week. The second reason is that a quarter may be the minimum timeframe within which the company can search, open and set up its initial operations for a hub. We were inclined to agree as aggregating demand would return more accurate forecasting results.
It is worth mentioning that the forecasting model was not a priority for this project; therefore, it was significantly simplified with a growth rate-based projection. We simply needed to validate the idea of finding optimal allocation of demand for the company. Clearly, this method can be improved upon with more data; however, due to time and resource constraints, it was not made a priority in this research.

4.2.3. Outbound transportation cost
The company’s pay strategy is based on the productivity of the delivery driver. According to the company, the outbound transportation cost contains two parts:

- Fixed cost per parcel
- Variable cost per parcel per distance (in km) which is converted to miles in our model.

4.2.4. Fixed hub cost, potential hub locations, and hub capacity
We collected the following data related to current hubs in the network:

- Total hub cost: fixed cost to open a hub; maintenance cost – percentage based on the total fixed cost; and cost per square meter which varies by location.
- Size of hub in square meters: translated to capacity at each hub.

Rather than search for live hubs to open at a future point, the cluster centroids identifying customer demand were selected as a proxy for potential hubs. And because, as mentioned above, the cost of each hub was tied to its location, it was possible to keep costs realistic.

4.3. Modeling and Network Model design
As discussed in the previous section, stochastic programming is an attractive option that allows for uncertainty and the control of risk. However, it requires not only large amounts of data but also more complicated processes to form a model (Snyder, 2006). Considering these drawbacks and the time constraint for this research, the MIP option with multi-period model extension was selected to accomplish the network design model.

(1) 1-stage network model: Hub and customers. This model was built based on a forecast of 12 quarter-periods.

(2) As a continuation of step (1), an optimization model determines the size of each hub in the network.

At first, a compact model was attempted to simultaneously solve both above steps. Due to the exponentially increasing complexity of the model, it proved nearly impossible. In more detail, the model (1) includes 12 periods, 19 demand points, 19 potential hubs, and 12 decision variables to decide the location and period within which a hub should be opened to handle demand growth. These combine to total 4,560 unique decision variables without incorporating the size of each hub. It is worth mentioning that model (1) was constructed as both greenfield and brownfield as discussed in section 4.2.1 to allow the company to consider the trade-off of completely revamping the current network configuration.
4.4. Model Formulation

4.4.1. Hub network optimization

As previously mentioned, the project objective was to find an optimal network configuration that includes (1) optimal flow for orders throughout the hub network; (2) a roadmap of open hubs corresponding with demand growth. To accomplish this task, we formed a multi-period optimization model for a three-year planning horizon (12 quarters).

The mixed integer linear program aims to answer the following questions:

- How much cost savings can be realized by designing a new network?
- How much more efficient, in terms of cost, is clustering vs. CoG?
- Can demand be met with the facilities selected?
- At what time should each hub be opened to minimize cost?

The model includes an input page in which the sponsoring company may input its hub locations. These locations can be determined with center of gravity, clustering, or any potential locations it has chosen. For our brownfield analysis we used the existing hub locations mixed with the center of demand gravity and clustering of demand locations that were discussed earlier. We used actual cost data for the existing hubs and 1,500 m² with actual rental costs varying with region for the proposed hubs. One additional input is required to capture ancillary costs at each location. This input is the percentage of monthly rent that covers costs such as maintenance, electricity, depreciation, financing, etc. One final key input is the predicted yearly growth of the company’s shipment volume. A demand forecast is generated for the following four years based on these numbers. This allows the multi-period model to open hubs as needed over time.

Once all of the inputs are entered, a distance matrix is generated that gives the distance between each hub and the 19 demand centroids. These are used in determining the transit and outbound cost estimates of each option. Each hub for each quarter has a binary variable to determine if it is open, as well as a decision variable to determine the optimal shipment volume to send through it.

Indices

<table>
<thead>
<tr>
<th>I</th>
<th>set of hub facilities ( i \in I )</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>set of forecasting period ( t \in T )</td>
</tr>
<tr>
<td>J</td>
<td>set of demand points ( j \in J )</td>
</tr>
</tbody>
</table>

Table 2: Set of Indices - Hub network optimization

Variables

<table>
<thead>
<tr>
<th>( x_{ijt} )</th>
<th>Quantity of parcel ( X ) flow through Hub ( i ) to customer ( j ) in period ( t ), unit: parcel</th>
</tr>
</thead>
<tbody>
<tr>
<td>( y_{it} )</td>
<td>1 if the hub ( i ) is recommended by the model to open in period ( t ), 0 otherwise</td>
</tr>
</tbody>
</table>

Table 3: Set of Variables - Hub network optimization
Parameters:

- $\varphi_{ijt}$: Fixed outbound last-leg transportation cost from hub $i$ to customer demand point $j$ in period $t$, unit: USD/parcel
- $\theta_i$: Fixed hub cost is defined by the default size of each hub multiplied times the renting cost/ sqm in location $i$, unit: USD/sqm
- $v_0$: Variable outbound last-leg transportation cost, unit: USD/mile
- $d_{ij}$: Distance from hub $i$ to customer demand point $j$, unit: mile
- $K_i$: Delivery capacity of Hub $i$, unit: parcel
- $D_{jt}$: Demand of customer $j$ in period $t$, unit: parcel

Table 4: Set of Parameters - Hub network optimization

Minimize (z)

$$
Min \sum_{i \in I} \sum_{t \in T} \theta_i \cdot y_{it} + \sum_{i \in I} \sum_{j \in J} \sum_{t \in T} (\varphi_{ijt} + v_0 \cdot d_{ij}) \cdot x_{ijt}
$$

Subject to:

1. $\sum_{i \in I} x_{ijt} = D_{jt}$ for $\forall j \in J$ (2) Demand constraint
2. $\sum_{j \in J} x_{ij} - M y_{it} \leq 0$ (3) Linking constraint
3. $\sum_{j \in J} x_{ij} \leq K_i$ for $\forall i \in I$ (4) Delivery capacity constraint for each hub $i$
4. $y_{it} = binary$ (5) Binary constraint
5. $x_{ij} \geq 0$ (6) Non-negativity constraint
6. $y_{it} \geq y_{it-1}$ (7) Constraint signaling once a hub is opened it will not be closed

The objective function (1) minimizes the total cost of transportation including outbound fixed and variable cost, and fixed hub cost. It is important to note that constraint (7) opens and closes a hub as suggested by the sponsoring company. This can be attributed to the fierce competition for real estate for logistics facilities in Ho Chi Minh City.

4.4.2. Hub sizing optimization model

One drawback of the above approach is the universally assumed hub size of 1,500 m$^2$ suggested by the company. This approach can lead to underutilization of hubs as well as difficulty locating a hub within the specified range, particularly within the city center. To tackle this problem a hub size optimization model was developed.

Indices
I : set of hub facilities $i \in I$
T : set of forecasting period $t \in T$
J : set of demand points $j \in J$

Table 5: Set of Indices - Hub size optimization

| $x_{ij}$ : Quantity of parcel $X$ flow through Hub $i$ to customer $j$, unit: parcel |
| $s_{i}$ : Size of hub $i$, unit: squared meter |

Table 6: Set of Variables - Hub size optimization

| $\phi_{ijt}$ : Fixed outbound last-leg transportation cost from hub $i$ to customer demand point $j$ in period $t$, unit: USD/parcel |
| $y_{it}$ : 1 if hub $i$ is recommended to open in period $t$, 0 otherwise |
| $v_{o}$ : Variable outbound last-leg transportation cost, unit: USD/mile |
| $d_{ij}$ : Distance from hub $i$ to customer demand point $j$, unit: mile |
| $K_{i}$ : Delivery capacity of Hub $i$, unit: parcel |
| $r_{i}$ : Rental cost per square meter of hub $i$ |
| $D_{jt}$ : Demand of customer $j$ in period $t$, unit: parcel |
| $m_{i}$ : Minimum space of hub $i$ |
| $M_{i}$ : Maximum space of hub $i$ |

Table 7: Set of Parameters - Hub size optimization

Minimize (z)

$$\text{Min} \sum_{i \in I} \sum_{t \in T} s_{i} * r_{i} * y_{it} + \sum_{i \in I} \sum_{j \in J} \sum_{t \in T} (\phi_{ijt} + v_{o} * d_{ij}) * x_{ijt}$$  \hfill (8)

Subject to:

$$\sum_{i \in I} x_{ij} = D_{jt} \text{ for } \forall j \in J$$  \hfill (9) Demand constraint

$$\sum_{j \in J} x_{ij} - M y_{it} \leq 0$$  \hfill (10) Linking constraint

$$\sum_{j \in J} x_{ij} \leq K_{i} \text{ for } \forall i \in I$$  \hfill (11) Delivery capacity constraint for each hub $i$

$$s_{i} - m_{i} * y_{it} \geq 0$$  \hfill (12) Minimum hub size

$$s_{i} - M_{i} * y_{it} \leq 0$$  \hfill (13) Maximum hub size

$$x_{ij} \geq 0$$  \hfill (14) Non-negativity constraint

Just as objective function (1) minimizes total fixed hub cost and transportation costs, objective function (8) is designed to find an optimal solution for each hub’s size. However, instead
of representing binary decision variables as in model (1), \( y_{it} \) becomes a parameter of the hub size to ensure the solution defined by model (1) is preserved. Constraints (12) and (13) reflect real-world market condition in that each district will necessarily provide a different range of hub sizes. For instance, in Ho Chi Minh City’s central business district, it is nearly impossible to lease a 2000 m\(^2\) logistics facility. The parameters \( m_i \) and \( M_i \) are set to 300 m\(^2\) and 2,000 m\(^2\) respectively.

5. Interpretation & Results

5.1. Baseline model

A baseline model, a replication of the current network with historical data, was built to ensure our model captures the main cost drivers of the company’s operations. Our replicated model confirmed that excluding fixed cost, each order would cost the company $0.339, compared to the actual average cost per order of $0.340. In other words, the simulated cost per order was lower than the actual cost by 0.27%. This gap can be accounted for by the cost factors that could not be captured in our model. Nevertheless, it is important to note that the outbound transportation cost function was a result of multiple revisions by the operations manager based on his experience. Due to time constraints and limited data availability, this cost function was agreed to by both the research team and the company in place of a more sophisticated process.

Because of the high variance in the quantity of orders in a given month (Figure 11), monthly data was not applied to the model. Rather, data from a peak week (week 51) was selected to be simulated and optimized. As a result, the total number of 203,248 orders was observed instead of nearly 700,000 orders mentioned in Section 4.1.

The model solution shows that the potential cost savings is $6,291 for week 51, equal to 8.42% of total cost (Table 8). The optimal network solution suggested that the company should have closed one of its current operations at Hub ID 1416 (Table 9). As we discovered later from interviews with the company’s operations staff there are two reasons for this. First, Hub 1416 is the company’s experimental hub for a new operations model offering various delivery services; hence it receives a considerable number of orders to validate the feasibility of implementing the new model. Furthermore, we were informed that at the company’s current size it has experienced many challenges with system hardware and software for sorting as well as its S&OP processes. Moreover, we observed that the highest contribution to cost savings came from the last-leg delivery cost.
5.2. Model Result and analysis

The model was used to optimize the network from one week of historical demand data in order to set a baseline to test the model. It was then decided to run four separate multi-period scenarios all with the same demand growth projection. The four scenarios were laid out as follows:

1) Center of Gravity Clustering – Greenfield
2) Center of Gravity Clustering – Brownfield
3) K-Means Clustering – Greenfield
4) K-Means Clustering – Brownfield

The greenfield analysis assumes that current as well as potential locations are ignored in favor of optimal locations. Brownfield analysis uses current hub locations mixed with optimal hub locations. In practice it is preferable to compare the two scenarios to determine what level of cost savings can be achieved by redesigning the entire network.

The brownfield locations were determined through communication with the company representatives. A constraint was placed in the model’s brownfield scenarios signifying that the existing hubs must be opened before a new hub location can subsequently be opened. A further constraint was applied to all scenarios that once a hub is opened it must remain open. It was determined that this more accurately reflects the investment requirements of getting a hub operational. It simply does not make sense to open and close a hub repeatedly and without this constraint, during a simulation the model would generate the lowest cost by closing a hub during a slow quarter.

To complete this model, we used Open Solver, an open source Excel add-in. When facing budget and time constraints as well as licensing issues associated with commercial grade solvers, it is often advisable to implement an open source alternative (Jared L. Gearhart, October 2013). One advantage of open source add-ins is that companies can easily install and integrate them into Excel. Any employee that needs access to this tool can do so without paying exorbitant licensing fees or undergoing extensive training.
Table 10: Sample distance between hub and customer demand

<table>
<thead>
<tr>
<th>Hub ID</th>
<th>Hub Address</th>
<th>Quan 1 - Ho Chi Minh</th>
<th>Quan 2 - Ho Chi Minh</th>
<th>Quan 3 - Ho Chi Minh</th>
<th>Quan 4 - Ho Chi Minh</th>
</tr>
</thead>
<tbody>
<tr>
<td>New</td>
<td>Hub 1</td>
<td>1.0</td>
<td>4.7</td>
<td>1.9</td>
<td>1.9</td>
</tr>
<tr>
<td>Old</td>
<td>Hub 2</td>
<td>6.6</td>
<td>2.4</td>
<td>8.9</td>
<td>6.7</td>
</tr>
<tr>
<td>New</td>
<td>Hub 3</td>
<td>1.3</td>
<td>6.4</td>
<td>1.0</td>
<td>2.2</td>
</tr>
<tr>
<td>New</td>
<td>Hub 4</td>
<td>1.4</td>
<td>5.1</td>
<td>2.4</td>
<td>1.0</td>
</tr>
<tr>
<td>New</td>
<td>Hub 5</td>
<td>2.6</td>
<td>7.3</td>
<td>2.4</td>
<td>3.5</td>
</tr>
<tr>
<td>Old</td>
<td>Hub 6</td>
<td>6.4</td>
<td>12.1</td>
<td>5.6</td>
<td>8.4</td>
</tr>
<tr>
<td>New</td>
<td>Hub 7</td>
<td>4.0</td>
<td>7.9</td>
<td>4.9</td>
<td>2.9</td>
</tr>
<tr>
<td>New</td>
<td>Hub 8</td>
<td>5.1</td>
<td>8.7</td>
<td>4.5</td>
<td>5.1</td>
</tr>
<tr>
<td>Old</td>
<td>Hub 9</td>
<td>11.7</td>
<td>6.9</td>
<td>10.6</td>
<td>11.9</td>
</tr>
<tr>
<td>Old</td>
<td>Hub 10</td>
<td>2.1</td>
<td>7.7</td>
<td>1.2</td>
<td>3.8</td>
</tr>
<tr>
<td>New</td>
<td>Hub 11</td>
<td>4.0</td>
<td>9.2</td>
<td>3.1</td>
<td>5.2</td>
</tr>
<tr>
<td>Old</td>
<td>Hub 12</td>
<td>11.5</td>
<td>13.9</td>
<td>10.6</td>
<td>12.4</td>
</tr>
<tr>
<td>New</td>
<td>Hub 13</td>
<td>7.4</td>
<td>12.0</td>
<td>6.5</td>
<td>8.4</td>
</tr>
<tr>
<td>Old</td>
<td>Hub 14</td>
<td>5.5</td>
<td>6.4</td>
<td>4.9</td>
<td>6.4</td>
</tr>
<tr>
<td>New</td>
<td>Hub 15</td>
<td>6.6</td>
<td>8.7</td>
<td>5.5</td>
<td>7.6</td>
</tr>
<tr>
<td>Old</td>
<td>Hub 16</td>
<td>4.6</td>
<td>7.6</td>
<td>3.6</td>
<td>5.7</td>
</tr>
<tr>
<td>New</td>
<td>Hub 17</td>
<td>3.8</td>
<td>9.5</td>
<td>2.9</td>
<td>4.7</td>
</tr>
<tr>
<td>New</td>
<td>Hub 18</td>
<td>5.3</td>
<td>12.1</td>
<td>4.4</td>
<td>7.1</td>
</tr>
<tr>
<td>Old</td>
<td>Hub 19</td>
<td>8.6</td>
<td>6.7</td>
<td>8.0</td>
<td>11.7</td>
</tr>
</tbody>
</table>

Table 11: Hub Costs – COG Brownfield

<table>
<thead>
<tr>
<th>Hub ID</th>
<th>Hub Address</th>
<th>Space</th>
<th>Renting cost/sqm</th>
<th>Costs (Maint,Elec, Depr,Emp)*</th>
</tr>
</thead>
<tbody>
<tr>
<td>New</td>
<td>Hub 1</td>
<td>1,500</td>
<td>150,000</td>
<td>15%</td>
</tr>
<tr>
<td>Old</td>
<td>Hub 2</td>
<td>114</td>
<td>120,000</td>
<td>15%</td>
</tr>
<tr>
<td>New</td>
<td>Hub 3</td>
<td>1,500</td>
<td>125,000</td>
<td>15%</td>
</tr>
<tr>
<td>New</td>
<td>Hub 4</td>
<td>1,500</td>
<td>120,000</td>
<td>15%</td>
</tr>
<tr>
<td>New</td>
<td>Hub 5</td>
<td>1,500</td>
<td>122,000</td>
<td>15%</td>
</tr>
<tr>
<td>Old</td>
<td>Hub 6</td>
<td>390</td>
<td>60,000</td>
<td>15%</td>
</tr>
<tr>
<td>New</td>
<td>Hub 7</td>
<td>1,500</td>
<td>110,000</td>
<td>15%</td>
</tr>
<tr>
<td>New</td>
<td>Hub 8</td>
<td>1,500</td>
<td>102,000</td>
<td>15%</td>
</tr>
<tr>
<td>Old</td>
<td>Hub 9</td>
<td>240</td>
<td>310,000</td>
<td>20%</td>
</tr>
<tr>
<td>Old</td>
<td>Hub 10</td>
<td>1,750</td>
<td>200,000</td>
<td>18%</td>
</tr>
<tr>
<td>New</td>
<td>Hub 11</td>
<td>1,500</td>
<td>160,000</td>
<td>15%</td>
</tr>
<tr>
<td>Old</td>
<td>Hub 12</td>
<td>97</td>
<td>160,000</td>
<td>15%</td>
</tr>
<tr>
<td>New</td>
<td>Hub 13</td>
<td>1,500</td>
<td>100,000</td>
<td>15%</td>
</tr>
<tr>
<td>Old</td>
<td>Hub 14</td>
<td>350</td>
<td>120,000</td>
<td>15%</td>
</tr>
<tr>
<td>New</td>
<td>Hub 15</td>
<td>1,500</td>
<td>100,000</td>
<td>15%</td>
</tr>
<tr>
<td>Old</td>
<td>Hub 16</td>
<td>1,100</td>
<td>150,000</td>
<td>15%</td>
</tr>
<tr>
<td>New</td>
<td>Hub 17</td>
<td>1,500</td>
<td>90,000</td>
<td>15%</td>
</tr>
<tr>
<td>New</td>
<td>Hub 18</td>
<td>1,500</td>
<td>200,000</td>
<td>15%</td>
</tr>
<tr>
<td>Old</td>
<td>Hub 19</td>
<td>300</td>
<td>100,000</td>
<td>15%</td>
</tr>
</tbody>
</table>

*% of total renting cost/month
Results:

As observed in Table 12, the greenfield CoG proved to be the lowest cost network. We expected the greenfield clustering analysis to provide the lowest cost network and although it was not the lowest it was within 1.3% of the greenfield CoG network cost. The initial explanation for this discrepancy is the sample size of shipments for the driving distance clustering. While we were able to run 400 data points, a sample of 1,000 or ideally 5,000 would produce a realistic picture of demand without the inherent risks of outliers in small samples. The algorithm that was developed for the clustering relies on Google API calls. These are free of cost up to a limit of 2,000 queries per day. With 1,000 shipment points and 19 hubs (=19,000 distances) this analysis would have been either time or cost prohibitive. With little time remaining we decided to take a sample of 400 and analyze the results.

The next explanation for this disparity is that both greenfield methods are setting the hub locations at the center of demand. CoG is using each geographical region as the constraint for its center of demand while driving distance clustering is only using the sample demand locations as the constraint in determining where to place each demand cluster. It is thus very prone to outliers. Because the model is measuring distance from each demand point to each hub, both greenfield networks are going to be very efficient. It is more important to analyze where future demand will likely occur rather than rely solely on the cost outputs of the model. For this one must evaluate larger samples of historical shipments and only then build optimal cluster locations. It is essential to avoid using arbitrary regions assigned by the government to constrain hub and demand locations. For these reasons, the driving distance clustering is the preferred method. One other topic of note is that if the company expects future demand to increase or decrease in certain areas dummy data can be generated and added to the driving distance clustering to help pull the demand and hub clusters in these directions.

Table 12: Matrix to compare the optimal solutions

<table>
<thead>
<tr>
<th></th>
<th>Driving Distance Clustering</th>
<th>K-means Clustering</th>
<th>COG</th>
</tr>
</thead>
<tbody>
<tr>
<td>Greenfield</td>
<td>$3,091,952</td>
<td>$3,180,577</td>
<td>$3,051,146</td>
</tr>
<tr>
<td>Brownfield</td>
<td>$3,338,654</td>
<td>$3,346,460</td>
<td>$3,398,860</td>
</tr>
</tbody>
</table>

Below is the roadmap for the greenfield clustering network option (Table 13). This table shows when to open each hub along with the utilization of each hub at any given time. Within the model identical charts are developed for every combination of brownfield/greenfield and clustering/CoG.
Table 13: Hub selection road map & utilization rate over the next 3 years

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>New Hub 3</td>
<td>CLOSED CLOSED CLOSED CLOSED OPEN OPEN OPEN OPEN OPEN OPEN OPEN OPEN OPEN OPEN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>New Hub 5</td>
<td>CLOSED CLOSED CLOSED CLOSED CLOSED OPEN OPEN OPEN OPEN OPEN OPEN OPEN OPEN OPEN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>New Hub 6</td>
<td>CLOSED CLOSED CLOSED CLOSED CLOSED CLOSED OPEN OPEN OPEN OPEN OPEN OPEN OPEN OPEN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>New Hub 8</td>
<td>CLOSED CLOSED CLOSED CLOSED CLOSED CLOSED CLOSED OPEN OPEN OPEN OPEN OPEN OPEN OPEN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>New Hub 9</td>
<td>CLOSED CLOSED CLOSED CLOSED CLOSED CLOSED CLOSED OPEN OPEN OPEN OPEN OPEN OPEN OPEN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>New Hub 11</td>
<td>CLOSED CLOSED CLOSED CLOSED CLOSED CLOSED CLOSED OPEN OPEN OPEN OPEN OPEN OPEN OPEN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>New Hub 12</td>
<td>CLOSED CLOSED CLOSED CLOSED CLOSED CLOSED CLOSED CLOSED OPEN OPEN OPEN OPEN OPEN OPEN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>New Hub 15</td>
<td>CLOSED CLOSED CLOSED CLOSED CLOSED CLOSED CLOSED CLOSED CLOSED OPEN OPEN OPEN OPEN OPEN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>New Hub 17</td>
<td>CLOSED CLOSED CLOSED CLOSED CLOSED CLOSED CLOSED CLOSED CLOSED CLOSED OPEN OPEN OPEN OPEN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>New Hub 19</td>
<td>CLOSED CLOSED CLOSED CLOSED CLOSED CLOSED CLOSED CLOSED CLOSED CLOSED CLOSED CLOSED CLOSED</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

In Table 13 we see three hubs (Hub 3, 11 and 17) would be underutilized (less than 50%) from serving areas with relatively low demand. As mentioned in section 4.4.2, at this point solver would be run to optimize the hub size keeping the optimal network configuration from the previous step unchanged. The optimized hub size suggests that at the end of the designated period, all of the opened hubs should be fully utilized (Table 14). Moreover, we also tested a “naïve” solution to find hub size by multiplying the default hub size (1,500 m²) by the maximum utilization rate of the entire period (2019-Q1 to 2021-Q4). The hub size optimization model and “naïve” solution returned $287,525 and $302,609 over the whole period, respectively. In other words, this model saves the company approximately 5.25% of the fixed cost of its hub network. This approach performs best in situations in which the decision maker is confronted with capacity constraints as well as variance in set-up cost across regions. Finally, in making recommendations to key decision makers, this simulation allows the planners to express their solution confidently, securely planning their hub network.
5.3. Simulation

To simulate variability in demand forecasting we used NORM.INV and RAND() to generate a random probability of demand. A Max() function was used to guard against extreme scenarios where we selected higher standard deviations of demand that could hypothetically reduce demand below 0; in this instance, the program will choose the maximum of 0 or the calculated demand.

A VBA code was written that inputs the values of the randomly generated demand into the model. The code then takes the results indicating which hub is to be open during each period and inputs it into a table. This data is then compiled after every simulation to show what percentage of the time that hub was selected.

In the model one can input how many simulations are to be run; the model then displays how many times the simulation has been executed. The figures below demonstrate an instance in which 1,000 simulations are performed with a 15% standard deviation of demand. The standard deviation and growth rates can be altered and the simulation re-run to determine their effect on hub selection.
This simulation is an important step in validating the network design decisions. If the network does not have sufficient capacity the simulation will yield an error message due to the fact that the hub capacity constraint (4) has been violated and no feasible solution can be found. If this error occurs frequently this is a sign that given increased variability of demand, network capacity should be increased as well. If the network has been designed with enough capacity to handle the variability, the simulation will run the requested number of times and the output can then be analyzed.

Table 15 displays the percentage of simulations that each hub was open given variable demand. This is a very useful chart to analyze because regardless of volume and location of
demand, the optimal hubs are going to have the highest selection percentages. We see in Table 15 that hubs 14 and 15 are open in 100% of the simulations. This is evidence of the fact that these are close to optimal locations. As the opening percentage falls, operations management can further scrutinize the potential location as a proper fit.

When the open hub percentage simulation matrix is used in conjunction with the average hub capacity utilization this process provides valuable insight into an uncertain future. The average capacity matrix informs the planner how utilized on average each hub is if it is opened during a simulation. The lower utilized hubs are the costlier to open and operate. They are not selected and utilized until the variable demand in the simulation absolutely requires them to be. By looking at the proportion of time they are selected as well as their capacity utilization, management can be sure that their network design decisions can withstand volatile demand fluctuations.

6. Conclusion
6.1. Reflection

The second largest express delivery company in Vietnam requested support in determining the optimal number and location of hubs to service delivery routes for its primary market operations. The company had a network of eight (mostly small 100-400 m²) hubs already in place in Ho Chi Minh City. However, this was the first time that an optimization model had been administered utilizing demand and cost data to identify optimal facility location. Working with the company, we created a model that could be easily reiterated for continuous improvement as well as for the company’s other markets. Greenfield analysis was conducted with 19 districts’ center of gravities as well as with clustering. In clustering we used driving distance to determine the ideal location of hubs while in both greenfield models we disregarded the current facilities in the network. The network optimization was performed on both the brownfield and greenfield sites, using both fixed and variable costs. The model determined the lowest cost to service the 19 clusters managing the tradeoff between facility and transportation as well as fixed and variable costs. Simulating variable demand with an estimated 15% standard deviation confirmed that the models were robust against stochastic fluctuations over a 12-quarter period, confirming that most hubs are to remain open under all scenarios. Both greenfield scenarios proved cost-efficient compared to the brownfields, as expected. Moreover, both the COG and clustering greenfield models were comparable, within approximately 1% cost of one another. Evidently, the network cost proved to be an indecisive factor in selecting a COG versus a clustering model. This is due to the relatively low variable transportation costs. Because the true cost will be determined by the locations of future customers and not merely by the facility costs and the distances between the validation data points, the model is being selected based on which is the most accurate and not merely the lowest cost in the validation model. Clustering is unconstrained by arbitrary administrative divisions. Moreover, by applying real time driving distances, the clustering model clearly surpasses COG for accuracy and distance efficiency.
6.2. Recommendations

Both greenfield and brownfield clustering models have advantages and disadvantages. The greenfield model is designed to meet current needs and future needs over the next twelve quarters. It allows the company to lease property for the next three years based upon an incremental growth agreement. In other words, additional space is leased as needed. As a result, the overall utilization will be maximized, reducing cost. It is an opportunity to start fresh disregarding suboptimal location placement while eliminating required maintenance on company-owned facilities. Seven of the eight current hubs are between 100 and 400 m². Thus, the sunk costs are minimal and the disruption to current operations will also be minimal. The greenfield model provides optimal cost and service levels, as well as flexibility to meet customer requirements.

The greenfield sites will, however, require a team to scout and ultimately develop as hubs. The increasing demand for sites in the city center may make negotiating for prime real estate with ideal conditions difficult. The brownfield sites have already been licensed and approved and the infrastructure is already in place. The total startup cost for brownfield are lower; however, this is dependent on other factors such as the condition of the sites. They may already require significant upgrades and modifications. The brownfields, moreover, will make extension beyond their current very limited capacity difficult due to space restrictions and possible structure and service issues. As a result, some design and operation efficiencies have already been conceded to meet these preexisting constraints. For instance, they have insufficient truck loading and unloading space. There is thus a higher risk of cost overruns for unforeseen vicissitudes. Site locations will be suboptimal not only based on distance to future customers but based on the operating difficulties inherent in poorly situated city center locations (for instance, traffic congestion, noise pollution, complaints from residential neighbors, etc.)

The most effective and economical solution is the greenfield model in this case. It is the recommendation of the authors that the company pursue this strategy.

6.3. Future improvements

The next step for the model is to train the company’s operations team to run a daily k-nearest neighbors algorithm to assign daily orders to the hubs. The travelling salesman heuristic can then be applied to find the shortest closed tour through each clusters’ set of nodes, delivering to each customer address only once before returning to the initial node at the centroid. The purpose of grouping the nodes to the clusters is that all nodes of an individual cluster will be touched contiguously. Adding time windows of deliveries and finite delivery capacity to the problem will extend the heuristic to a vehicle routing problem.

We also recommend that the company determine the most accurate variable costs by running a regression in which employee remuneration is input as the dependent variable and distance and weight as the independent variables. Currently, an approximation is being used based on the operations manager’s experience.
Finally, critical ratio can be used to determine the priority of on-demand orders. By subtracting the current date from the due date for an expedited order and subtracting the total number of delivery manhours remaining from that and dividing it all by the number of on-demand shipments remaining, the delivery job with the lowest score can be determined and prioritized for scheduling next. Moreover, the Newsvendor critical ratio can also be used to determine how many part time on-demand employees are needed based on the ratio of labor cost to penalty of late on-demand shipments.

6.4. Discussion

6.4.1. Limitations

Data collection – Because the company was in the process of a database changeover during the course of the project, it was unable to provide us with a merged sheet containing both shipment metrics and employee remuneration. We were thus unable to use regression to determine the most accurate variable costs where employee remuneration is a dependent variable and distance and weight are independent variables. Currently, an approximation is being used based on the operations manager’s experience. As a result, the simulated cost per order was lower than the actual cost by 0.27%. The company was also unable to provide us with GPS coordinates. We necessarily needed to convert the addresses to x,y coordinates using Google API. However, as a result the converted data in our samples contained outlier coordinates that marginally impacted our clustering analysis.

Processes – The scope of research of this paper was to design an optimal facility network of hub locations as well as planned capacity for the entire network. Consequently, the demand forecasting model we employed was not made a priority for this project primarily due to time and resource constraints. In fact, it was significantly simplified with a growth rate-based projection. This method can be improved upon on the condition that at least two years of historical demand data are provided by the company.

Software – At first, we attempted to simultaneously solve for both optimal location and size of hubs in our model. Because this model is composed of 12 periods, 19 demand points, 19 potential hubs, and 12 decision variables, the complexity increased exponentially with a compact model. Without incorporating the size of each hub, these combined variables give rise to 4,560 unique decision variables. While Open-Solver allows far more decision variables than the very limited 200 permitted with standard Excel solver functionality, the processing time is still a prohibiting factor. As a result, without the training and accessibility of resources such as Supply Chain Guru or other specialized optimization software, we were impelled to combine two separate optimization models, somewhat decreasing the accuracy of our results.

6.4.2. Delimitations

The research described in this paper can be categorized as a hub location-routing problem. We considered the fact that Ho Chi Minh City, while the largest city in Vietnam, is still relatively small when compared to Vietnam as a whole. As a result, we decided that optimizing
delivery drivers’ routes should not be our priority. Moreover, Ho Chi Minh City is a very dynamic and rapidly changing place with dense traffic in which an optimal route one day could be a dead-end the next. In fact, the drivers have their own algorithm to optimize their individual routes. And they still manage to deliver 40 – 50 orders/day which corresponds to $3 \times 10^{64}$ possible routes from which to choose. Moreover, delivery staff remember their customers’ behavior. After multiple discussions with company managers and justifying which elements would contribute the most value, we eschewed this approach to addressing the fundamental problem of hub location.

As mentioned in section 4.1, we selected the centroids produced by our clustering algorithm to serve as the hub locations. But in reality, the precise optimal location is always slightly elusive primary because there is always the strong possibility that it is presently occupied and unavailable to purchase or lease. However, given the limited time for this project, the sponsoring company and the research team decided not to invest the time hunting down potential real estate for hub locations one at a time in Ho Chi Minh City based on the centroids we generated.

The 1-stage model was selected in lieu of a 2-stage model for several reasons. The first rationale is that last-leg delivery accounts for almost 60% of total operational cost. Consequently, optimizing hub location must be a priority. The second justification to un-incorporate sorting centers is the requirement of a great deal of data that would have required more time to clean than we had available. Finally, hub facilities can be considered more fungible/exchangeable as compared to sorting centers. A focus on optimizing hub location proved more practical and feasible to implement.

6.5. Final thoughts

Our model optimized which of the 19 hubs were to be operational over a 12-quarter period. Moreover, it selected which route centroids they serve, comparing changes in cost and distance between a baseline network and the new hub network. The company now has a repeatable process for implementing similar analyses in its other markets. The project proved that using this process produces a more optimal solution much faster, reducing administrative and software costs. The new process uses open source software that is easily accessible and user friendly. The company can now change constraints to allow more involved services and costs, reduce its margin of error and rapidly implement many scenarios and cost models with one repeatable process.
BIBLIOGRAPHY


APPENDIX

Clustering driving distance

Python version: 2.x

```
1. # -*- coding: utf-8 -*-
2. ***
3. Spyder Editor
4. ***
5. This is a temporary script file.
6. ***
7. import numpy as np
8. import pandas as pd
9. import math
10. import requests
11. import os.path
12. from openpyxl import load_workbook
13. from sklearn.cluster import KMeans
14. 15. # Input valid API_KEY
16. api_key = "&key=AIzaSyAfbHIDwemZnPqhuzYRtAY7Ijn7frNyk"
17. front_url = "https://maps.googleapis.com/maps/api/distancematrix/json?units=metric&origins=
18. "
19. wb = load_workbook(filename='Clustering Data.xlsx')
20. ws = wb['COGs NW']
21. data_rows = []
22. for row in ws['H3':'I102']:
23.     data_cols = []
24.     for cell in row:
25.         data_cols.append(cell.value)
26.     data_rows.append(data_cols)
27. df = pd.DataFrame(data_rows)
28. gps_coords = np.asarray(df)
29. 30. data_rows = []
31. for row in ws['C3':'D21']:
32.     data_cols = []
33.     for cell in row:
34.         data_cols.append(cell.value)
35.     data_rows.append(data_cols)
36. df = pd.DataFrame(data_rows)
37. des_coords = np.asarray(df)
38. 39. 40. matrix_data = np.zeros((gps_coords.shape[0], des_coords.shape[0]))
41. 42. if not(os.path.isfile('temp_data.xlsx')):
43.     for i in range(0,gps_coords.shape[0]):
44.         for j in range(0,des_coords.shape[0]):
45.             url = front_url + str(gps_coords[i,0]) + "," + str(gps_coords[i,1]) + "," + str(des_coords[j,0]) + "," + str(des_coords[j,1]) + api_key
46.             request = requests.get(url)
47.             json_data = request.json()
48.```
matrix_data[i,j] = json_data['rows'][0]['elements'][0]['distance']['value']

print(i,j)

df = pd.DataFrame(matrix_data)
writer = pd.ExcelWriter('temp_data.xlsx', engine='xlsxwriter')
df.to_excel(writer, sheet_name='distance_matrix')
writer.save()

else:
    wb = load_workbook(filename='temp_data.xlsx')
    ws = wb['distance_matrix']
data_rows = []
for row in ws['B2': 'T101']:
    data_cols = []
    for cell in row:
        data_cols.append(cell.value)
data_rows.append(data_cols)
df = pd.DataFrame(data_rows)
matrix_data = np.asarray(df)

weights = np.ones((gps_coords.shape[0], 1))
min_distance = 20000
num_clusters = 19
distance_data = []
distance_data.append([min_distance+1])
distance_data = np.asarray(distance_data)
while (any(np.greater(distance_data, min_distance))):
    print('

' + "New K-Means Cluster with Cluster Number: " + str(num_clusters))
kmeans = KMeans(n_clusters=num_clusters, random_state=0).fit_predict(matrix_data)
coordinate_data = []
distance_data = []
center = []
for cluster in range(0, num_clusters):
    cluster_data = []
    weight_data = []
    for i in range(0, kmeans.shape[0]):
        if kmeans[i,] == cluster:
            cluster_data.append([gps_coords[i,0], gps_coords[i,1]])
            weight_data.append([weights[i,]])
    cluster_data = np.asarray(cluster_data)
    weight_data = np.asarray(weight_data)
x=[]
y=[]
z=[]

r = 6371000
\[ x = r \times \cos(cluster\_data[:,0]\times\text{\textpi/180}) \times \cos(cluster\_data[:,1]\times\text{\textpi/180}) \] 
\[ y = r \times \cos(cluster\_data[:,0]\times\text{\textpi/180}) \times \sin(cluster\_data[:,1]\times\text{\textpi/180}) \] 
\[ z = r \times \sin(cluster\_data[:,0]\times\text{\textpi/180}) \] 
\[ xc = \text{\textnp.dot(x.T, weight\_data)}/\text{\textnp.sum(weight\_data)} \] 
\[ yc = \text{\textnp.dot(y.T, weight\_data)}/\text{\textnp.sum(weight\_data)} \] 
\[ zc = \text{\textnp.dot(z.T, weight\_data)}/\text{\textnp.sum(weight\_data)} \] 
\[ \text{\textlon} = \text{\textnp.arctan2(yc, xc)} \times 180 / \text{\text\textpi} \] 
\[ \text{\texthyp} = \text{\textnp.sqrt(xc*xc + yc*yc)} \] 
\[ \text{\textlat} = \text{\textnp.arctan2(zc, hyp)} \times 180 / \text{\text\textpi} \] 

for \( j \) in range(0, x.shape[0]): 
url = front_url + str(lat) + "," + str(lon) + "&destinations=" + str(cluster_data[j,0]) + "," + str(cluster_data[j,1]) + api_key 
request = requests.get(url) 
json_data = request.json() 
distance_data.append([json_data['rows'][0]['elements'][0]['distance']['value']]) 

distance_data.append([-1]) 
center.append([lat, lon]) 
coordinate_data.append(cluster_data) 
print('\n' + "Coordinate data from cluster #" + str(cluster) + ":") 
print(np.asarray(cluster_data))

distance_data = np.asarray(distance_data) 
center = np.asarray(center) 
coordinate_data = np.asarray(coordinate_data) 
coordinate_data = np.concatenate(coordinate_data, axis=0) 
num_clusters += 1 

df = pd.DataFrame(coordinate_data) 
writer = pd.ExcelWriter('result_data.xlsx', engine='xlsxwriter') 
df.to_excel(writer, sheet_name='results') 
writer.save() 
print('\n' + "Distances from centroid to each location in the cluster: ") 
print(distance_data) 

print('\n' + "Center locations for each cluster: ") 
print(center)