國 立 交 通 大 學

運 輸 與 物 流 管 理 學 系

博 士 論 文

No.023

考量不確定性的群眾外包配送作業

Last-mile Delivery with Crowdsources Integration Considering the Crowdsources Uncertainty

研 究 生 ：戴汐雅

指導教授：黃寬丞 博士

吳昆峯 博士

中 華 民 國 一 ○ 八 年 九 月

考量不確定性的群眾外包配送作業

Last-mile Delivery with Crowdsources Integration Considering the Crowdsources Uncertainty

研 究 生 ：戴汐雅　　Student：Muhammad Nashir A

指導教授：黃寬丞 博士　　Advisor：Dr. Kuancheng Huang

吳昆峯 博士　　Dr. Kun-Feng Wu

國 立 交 通 大 學

運輸與物流管理學系

博 士 論 文

A Dissertation

Submitted to Department of Transportation and Logistics Management

College of Management

National Chiao Tung University

in Partial Fulfillment of the Requirements

for the Degree of

Doctor of Philosophy

in

Transportation and Logistics Management

September 2019

Hsinchu, Taiwan, Republic of China

中華民國一〇八年九月

Last-mile Delivery with Crowdsources Integration Considering the Crowdsources Uncertainty

Student: Muhammad Nashir Ardiansyah

Advisor: Dr. Kuancheng Huang

Dr. Kun-Feng Wu

Department of Transportation and Logistics Management National Chiao Tung University

ABSTRACT

Crowdsource delivery is reported to contribute a significant role for last-mile delivery (LMD). Lower operational cost and capital investment, as well as delivery flexibility, are the main advantages of crowdsource delivery when compared to the conventional LMD. Positive results of integrating crowdsource delivery into the LMD have been reported in terms of delivery cost, service level, and environmental impact. This study investigates the delivery plan of LMD in a collaboration with the crowdsources as one of the delivery options. The crowdsources provide delivery assistance from transfer points to the customer locations. This collaboration requires parcel relay between main delivery trucks and crowdsources at transfer points. In the real situation, this parcel relay activity might be subjected to several kinds of uncertainties (e.g. congestion, weather condition, etc.) that can create disturbance to the process.

In this study, the decision problem is tackled from two aspects, the deterministic and stochastic points of view. In the deterministic point of view, the benefits of crowdsources delivery collaboration are investigated given the perfect situation (with no uncertainty) by formulating a problem as a mixed integer linear program (MILP). Upon the uncertainty considered in the stochastic point of view, this study models the parcel transfer or relay event as an uncertain event, which involves the success or failure of the crowdsources' show-up. A two-stage stochastic MILP model is formulated to as the optimization model considering the associated uncertainty. The heuristics algorithms based on Tabu Search (TS) are designed to handle the large-scale problems for both the deterministic and stochastic versions of the mathematical programming models.

In summary, the crowdsource delivery collaboration improves the LMD plan by properly outsourcing some delivery orders to reduce the overall delivery costs. The balance between the delivery fleet utilization and the usage of crowdsourcing service must be carefully achieved to provide the maximum benefit of crowdsources delivery collaboration. These benefits can still be preserved even after the consideration of uncertainty. Based on the numerical experiment, the heuristics algorithm is able to provide the high quality solution with fast computation time.

Keyword: Crowdsource delivery, Last-mile delivery, two-echelon routing problem, stochastic routing problem.

考量不確定性的群眾外包配送作業

研 究 生 ：戴汐雅　　　　　　　　　　指導教授：黃寬丞 博士
吳昆峯 博士

國 立 交 通 大 學 運輸與物流管理學系

摘要

根據研究，群眾外包在終端客戶配送或者稱最後一哩配送(Last-Mile Delivery, LMD)中扮演著重要的角色。與傳統 LMD 相比，群眾外包運送的主要優點在於其擁有較低的營運成本、資本投資以及運送作業的彈性。就運送成本、服務水準與環境影響而言，將群眾外包運送整合至 LMD 中已有實質的效益。本研究將群眾外包整合作為 LMD 運送計劃其中一項的可行的選項。群眾外包提供從轉運點至客戶所在地之運送協助，此整合模式需在轉運點進行主要運送車隊與群眾外包配送者間的包裹轉運。在實際運作情形中，包裹轉運的過程可能會受到各種不確定性因素的干擾(例如：壅塞、天氣變化等)。

本研究將此決策問題區分為兩個角度：確定性與隨機性觀點。就確定性的角度而言，假設在最理想(即沒有不確定性)的情況下，透過將問題規劃成混和整數線性規劃模式(MILP)，藉以研究群眾外包運送的優點。就隨機性觀點來考量不確定性，本研究將群眾外包包裹轉運的成功與否視為導致不確定性的事件，並使用兩階段隨機規劃(SP)來建構最佳化模型以考量此不確定性。除此之外，為處理確定性與隨機性之大規模問題，本研究以禁忌搜尋法(TS)為基礎設計了啟發式演算法。

總體而言，群眾外包運送的整合方式可透過適當分配運送訂單以降低總運送成本，進而改善 LMD 計劃。然而必須小心達成運送車隊與群眾外包服務之間的平衡，才能達到群眾外包運送合作模式的最大效益。即使將不確定性納入考量，本研究所發展模式依然能達到群眾外包的效益。而透過數值實驗發現，啟發式解法能在快速的運算時間提供高品質的解決方案。

關鍵字：眾包交付、最後一英里交付、兩級路由問題、隨機路由問題

ACKNOWLEDGEMENT

Alhamdulillah! Thanks to Allah Almighty Who enabled me to finish my study and research in Department of Transportation and Logistics Management, National Chiao Tung University, Taiwan.

Firstly, I would like to express my sincere gratitude to my advisors Dr. Kuancheng Huang and Dr. Kun-Feng Wu for the continuous support of my Ph.D. study and related research, for his patience, motivation, and immense knowledge. His guidance helped me in all the time of research and writing of this thesis. I could not have imagined having a better advisor and mentor for my Ph.D. study.

Besides my advisor, I would like to thank the rest of my dissertation committee members: Dr. Ming-Jong Yao, Dr. Wen-Chih Chen, Dr. Ching-Jung Ting, and Dr. Jenn-Rong Lin, for their valuable comments, insightful critics, feedbacks, suggestions, and also for the hard question which incented me to widen my research from various perspectives.

I would like to express my special thanks to my family, especially my wife, Anissa, and my daughter, Naraya for accompanying me during my difficult time pursuing my doctoral degree in Taiwan. Thanks to all my family members, my father, my mother, my sister, and all of my family members for their supports and motivations.

And finally, last but by no means least, I want to thank all members of Department Transportation and Logistics Management, NCTU, from the Professors, Officers, Students, and others whose names are not listed here, for their support, knowledge, experience, and friendship. It was great experiences here in this campus and Taiwan.

Thank you TLM, NCTU, Taiwan for the great experience. I hope to see you again in the future!

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1 INTRODUCTION

1.1. Background and motivation

As one of the emerging trends, e-commerce changes the consumer behavior of purchasing consumer goods. It changes the majority of last-mile delivery (LMD) from conventional LMD to the final home-delivery which is characterized as rapid, relatively small size (in volume or weight), and scattered. As a result, LMD can consume up to 75% of the total supply chain costs (Gevaers et al., 2009). This number is predicted to increase due to the growth of e-retailing giants (e.g. Amazon, eBay, Taobao. etc.) and online e-commerce stores (Mckinnon, 2016). In addition to the cost increase, new challenges have been discovered, such as failed delivery issues (ping-pong effect), reverse logistics problem, local policy implementation, etc. In urban city environment, an increase of congestion and pollution have been reported due to the increasing number of deliveries (Allen, et al., 2000).

Several concepts have been proposed to overcome and reduce the LMD issues, such as collaboration between several logistics companies to maximize their resources utilization (Park et al., 2016; Liakos & Delis, 2015; de Souze et al., 2014; Petrovic et al., 2013), collaboration with the convenience store as a drop off point, and the concept of shared reception box as a drop off point (Wang et al., 2016; DellAmico & Hadjidimitriou, 2012; Punakivi et al., 2001). The latest concept to improve LMD problem is to introduce the crowdsourcing concept to LMD in which the crowdsources perform certain logistical tasks in return of a reward (Arslan et al., 2019; Pitchka et al., 2018; Kafle et al., 2017; Devari et al., 2017; Wang et al., 2016; Rouges & Montreuil, 2016).

The evolution of sharing economy and the advancement of communication technology had led the crowdsourcing concept to be evolved drastically. Wikipedia, Kickstarter, Uber, etc. are some real examples of the crowdsourcing business today. Crowdsources participation enables the new opportunities for doing any tasks with cost-efficient, flexible, and relatively high speed manner. In the logistics business, crowdsourcing the logistics tasks has been around for years, especially the crowd-delivery (Rouges & Montreuil, 2014). More than fifty start-up companies which can be classified as crowdsource delivery service providers were established to perform LMD tasks (Carbone et al., 2017). Crowdsourcing the logistics task can be considered as an innovative idea due to characteristics of e-commerce goods which are relatively small in size and can be transferable to the crowdsources (Schenk and Guittard, 2011). In addition, the crowdsources are also faster in the congested area, use more environmental friendly mode, and flexible to match the delivery schedule. In terms of cost, crowdsource delivery offers a lower delivery costs due to the usage of unused resources in terms of time, assets and capacity to perform the delivery task.

Positive results of integrating crowdsource delivery into the LMD have been reported in terms of delivery cost, service level, and environmental impact. The collaboration with crowdsources can increase the logistics company's fleet utilization in terms of distance with approximately 57% mileage reduction can be made based on the simulation (Devari et al., 2017). Delivery cost reduction for about 10-20% can also be made by integrating crowdsources in to the LMD (Kafle et al., 2017; Huang & Ardiansyah, 2019). In terms of service level, crowdsources delivery also can reduce LMD failure and increase the service level (Akeb et al., 2018). As for the society gains, crowdsource delivery can reduce the environmental negative impact, such as polution and traffic congestion. Based on the case study, a potential reduction on carbon footprint is reported to be equivalent to 1.6 km in average (Paloheimo et al., 2016).

In practice, several crowdsource delivery implementations are found and have been implemented by several logistics and retailer companies. A concept of parcel pick-up and delivery on the way to the crowd commuter's destination has been proposed by DHL and Wal-Mart (Barr and Wohl, 2013). AmazonFlex implements a different concept of crowdsourced delivery as it requires the crowdsources to pick up the packages from the headquarter, retailer, or store and deliver it to the consumer location (Reilly, 2015). Grocery delivery or typical meal delivery services by GrubHub, UberEats, Panda, GO-FOOD, etc. are several popular implementations of crowdsouce delivery today (Sampaio et al., 2019).

In general, crowdsource delivery can be categorized into two categories based on the crowdsources participation over the delivery task, namely full-coverage crowdsource delivery and partial-coverage crowdsource delivery (Kafle et al., 2017 & Huang & Ardiansyah, 2019). In full-coverage crowdsource delivery, the crowdsources cover the whole distance of the delivery order. The crowdsources pick up the customer order from the location of the shipper (e.g. warehouse, DC, store, retailer etc.) and deliver it directly to the customer locations (Archetti et al., 2016; Palheimo et al., 2016; Arslan et al., 2019). In partial-coverage or relayed crowdsource delivery, the customer order is relayed by the main delivery truck to the crowdsources at the transfer location, then the crowdsources will continue deliver customer order to the customer location (Kafle et al., 2017; Pichka et al., 2018; Huang & Ardiansyah., 2019). The partial-coverage crowdsource delivery offers several potential benefits over the full-crowdsource delivery service, such as ease of finding crowdsources, more environmental friendly, and more flexible to match the recipient available time (Chen et al., 2017). The illustration of partial-coverage crowdsource delivery is provided in Figure 1.

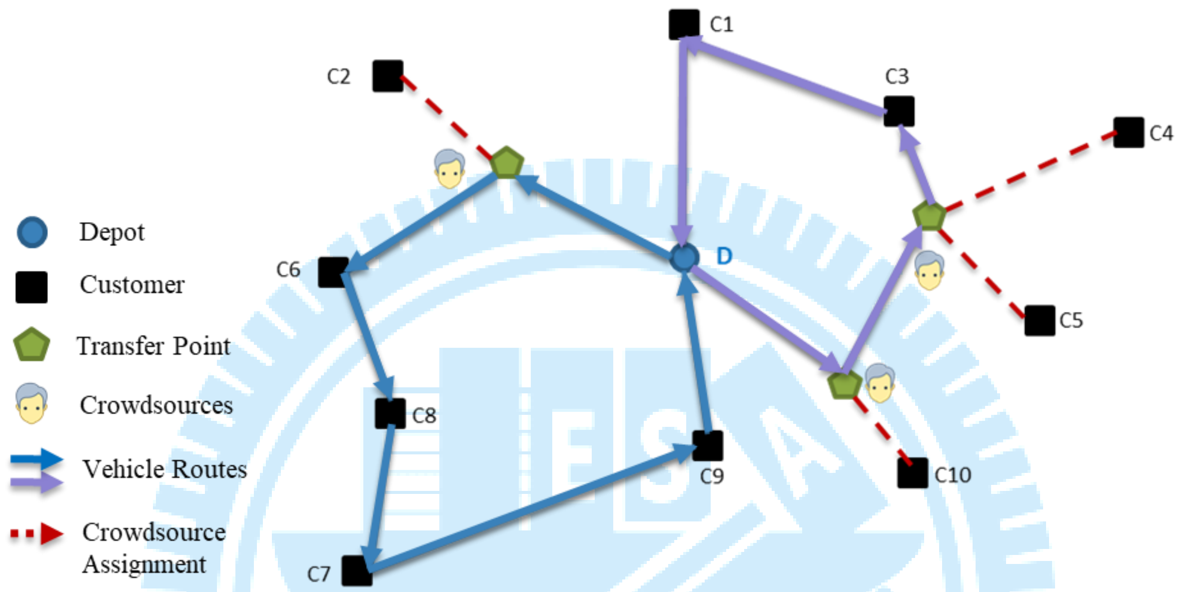


Figure 1. Illustration of crowd-delivery in an urban area (Source: Kafle et al., 2017)

Although partial-coverage crowdsource delivery offers many benefits, it also has limitations, such as unstandard services, possibility of crimes and violations, and possibility of crowdsources late arrival and task cancelation. This study will focus to consider the last limitation which is the possibility of late arrival and task cancelation. Any delay or crowdsources task cancelation will disrupt the transfer or relay process and make the whole crowdsource delivery plan fail (referred as the crowdsource transfer failure). The customer order which is relayed by the delivery truck to the crowdsources at the transfer location will become unsend customer order when the crowdsources and delivery truck fail to initiate the relay process. The delivery truck may have no time to send the unsend customer order due to the next delivery plan. An illustration for crowdsource transfer failure is provided in Figure 2.

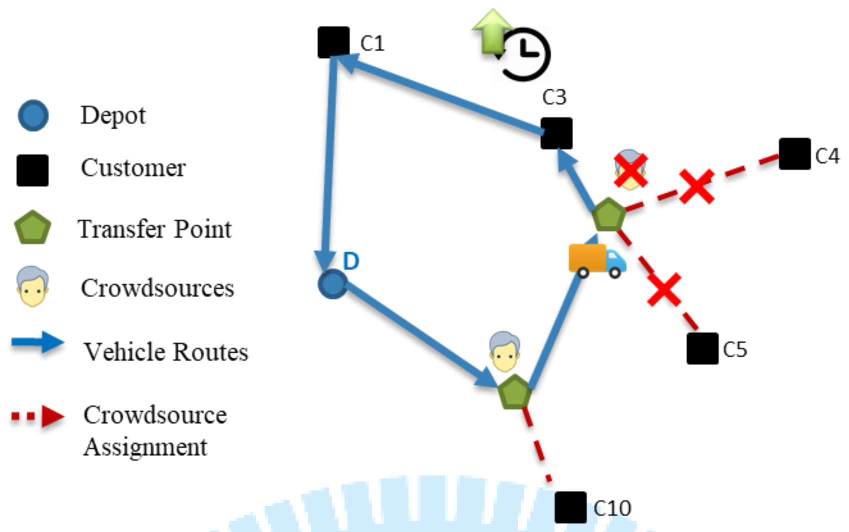


Figure 2. Illustration of crowdsource transfer failure and unsend customer order

The crowdsources who usually use small and user-friendly mode (e.g. pedestrians, cyclist, or scooter-rider) are prone to the weather changes, as sudden rain might delay or cancel their transfer arrangement. In addition, the probability of late arrival may depend on the traffic congestion, especially in the urban area. Since the crowdsources are strangers to the logistics company, they can easily cancel the transfer process when the situation become difficult. In fact, the big hitchhiker companies such as Uber and Grab allow their drivers or crowdsourcing partners to cancel their service up until 5 – 20% before they get evaluated (ABS-CBN News, 2018; Siddiqui, 2016).

This study investigates the concept of partial-coverage crowdsource delivery collaboration to improve the LMD. The uncertainty in terms of the possibility to have successful crowdsource transfer is considered to represent the real crowdsourcing situation. The problem will be formulated as the optimization model in order to get efficient, effective, and robust delivery plan integrating the conventional LMD and the crowdsource delivery. The results of this study can help the decision maker (e.g. logistics operator, retailer) to decide several important delivery and crowdsource decisions, such as the selection of customers that need to be crowdsourced, the number of crowdsource partners, the location of the parcel transfer, and the schedule of crowdsource transfer.

1.2. Research objective

This main objective of this research is to generate the delivery plan of LMD with the crowdsource delivery integration. The problem is approached as two different models based on the uncertainty consideration, namely deterministic model for no uncertainty consideration and stochastic model with the uncertainty consideration. For deterministic model, the main objective is to generate an efficient and effective delivery plan by including the crowdsources as one of the delivery options. In the stochastic model, the crowdsource transfer event is considered as the uncertain event to represent the actual situations and generate robust delivery plan. In addition to the optimization models, this study also designs the heuristic algorithms for both deterministic and stochastic models to handle large-scale problems in a fast computation time.

1.3. Research framework

In order to develop a comprehensive study, the problem is approached with two different models. The first model is to study the problem in the deterministic environment assuming every aspect is deterministic to give a baseline of how much this collaboration benefits the LMD. An optimization model is designed to represent the problem and generate the final delivery plan. In the second model, this study considers a crucial aspect in the partial-coverage crowdsource delivery which is the crowdsource transfer process as an uncertain event. This consideration shifts the deterministic problem into a stochastic problem. Stochastic optimization model is proposed based on the uncertainty realization of the problem. By considering the uncertain environment, the results of this study become more robust and ready to be implemented in the real LMD.

Based on the deterministic and stochastic models, the heuristic solution algorithms are proposed to generate a good solution quality with relatively fast computation time. The heuristic solution algorithm consists of two sub-algorithms, namely construction algorithm to generate the good initial solution and improvement algorithm to improve the initial solution. Numerical experiments will be performed based on the artificial instances and classical routing problem benchmark instances to show how much improvement can be made. Senstivity analysis to identify the important parameters are performed as part of the numerical experiment. In the end, the discussion related to the important findings will be presented, as well as the conclusion of this study. The research framework of this study is presented in the Figure 3.

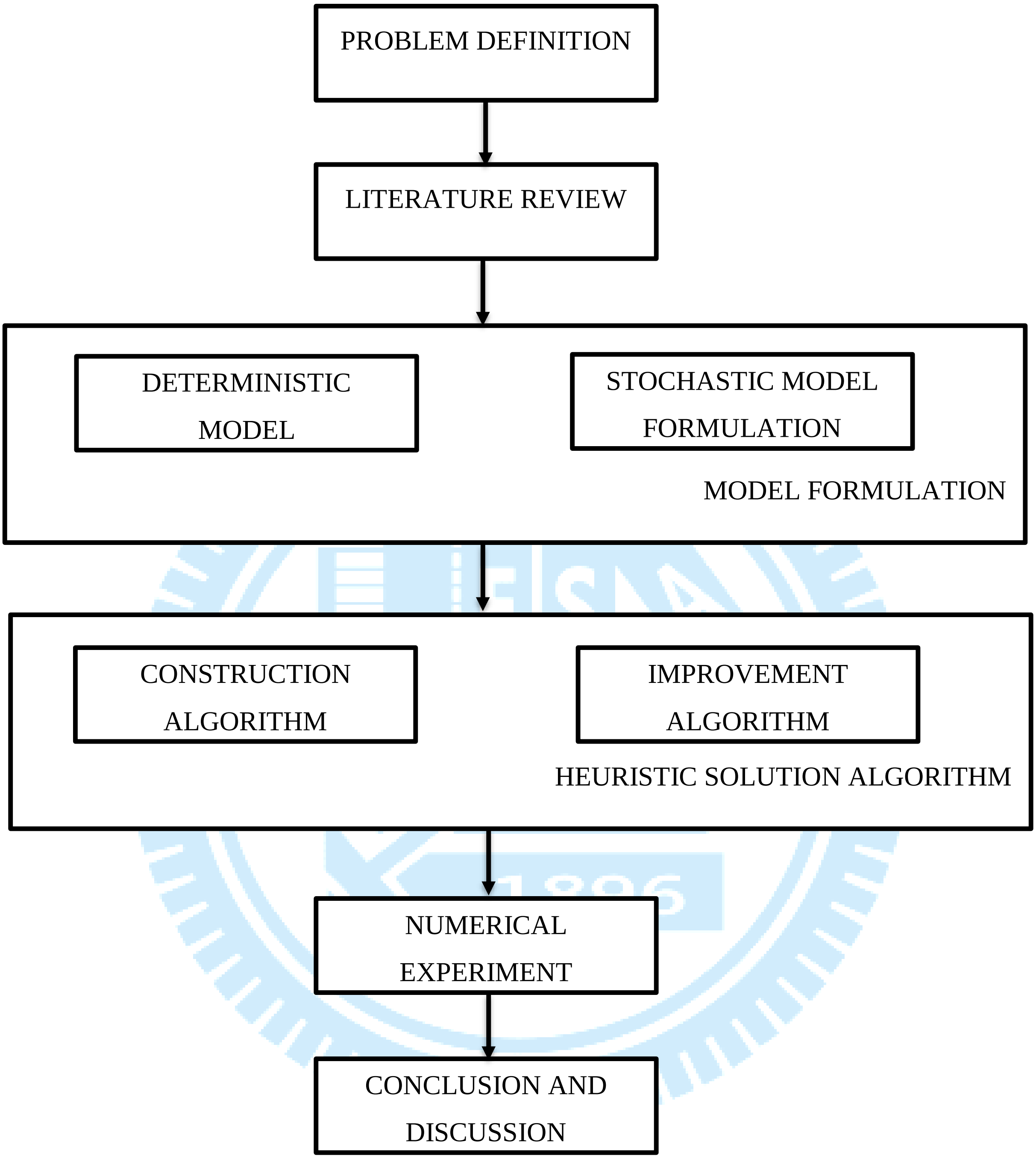


Figure 3. Research framework

CHAPTER 2 LITERATURE REVIEW

In this chapter, a comprehensive literature review is presented by reviewing three different research areas related to the study. The crowdsource delivery literature will be the first research area to be reviewed due to the topic of this study. The second review is based on the two-echelon routing problem as most of partial-crowdsource delivery problem researches are categorized as the two-echelon routing problem. In the last literature review, the stochastic routing problem will be presented to get the latest and related routing problem literatures involving the uncertainty.

2.1. Crowdsource delivery

The concept of crowdsourcing in the delivery is not new. Rouges & Montreuil (2014) published a specific article about crowdsourcing delivery which might be the frontier study in this area. Some great reviews of crowds-logistics can be found in Carbone et al. (2017) and Rouges & Montreuil (2014). In general, the crowdsource delivery researches are classified in two categories based on the crowdsources participation in performing delivery tasks, namely full-coverage crowdsource delivery and partial-coverage crowdsource delivery.

In the full-coverage crowdsource delivery, crowdsources perform the delivery task from the point of origin (e.g. warehouse, DC, store, retailer, etc.) to the customer locations. In this category, the crowdsources usually are the travelers which have fully or partially similar trip as the delivery order trip, or occasional couriers. The delivery and return task of library service in Finland was reported by Paloheimo et al. (2016). The delivery task was posted on the smartphone apps to deliver or pick up a book which was borrowed from the library. This approach was able to reduce the average car driven and some CO2 emissions. Arslan et al. (2019) investigated dynamic pickup and delivery problem for which ad-hoc driver is utilized as the crowdsourcing partner. This study proposed the viability condition to incorporate crowdsources to perform on-demand delivery. A significant improvement was reported when comparing non-crowdsourced delivery and crowdsourced delivery. Punel and Stathopoulos (2017) reported a choice model to identify the most significant factor of the crowdsource delivery preferences and acceptance in the perspective of the customer. The issues about trusts on delivering a parcel by crowdsources were discussed by Devari et al. (2017). This study proposed a friendship modeling to determine who can be considered as trustworthy crowdsourcing partner. Almost 72% of respondents as a shipper agree to entrust their parcel to their friends. Based on a simulation, crowdsourced delivery is beneficial compared to the traditional delivery.

In the partial-coverage crowdsource delivery, the delivery truck initiates the process by delivering customer order to the transfer location and relaying it to the crowdsources. This concept shares some similarities with the shared-reception box (SRB) concept. However, there is time limitation to meet at the transfer point which significantly differs this concept from the SRB concept. Wang et al. (2016) proposed a crowdsourcing concept which assigns the parcels to the SRB so the crowdsources can collect a parcel from SRB and deliver it to the customer location. By this mechanism, the unattended delivery problem will be reduced because they can synchronize the available time between parcel receiver and the crowdsources. Chen and Pan (2016) reported a concept of a crowd-taxi LMD. The integration between transporting passenger and delivering goods was proposed to perform LMD. Kafle et al. (2017) designed the relay LMD system which incorporates crowdsources as last-leg delivery. Determination of which relay point and which crowdsources to be selected is one of many interesting aspects in their formulation. Akeb et al. (2018) designed a solution to solve the unattended parcel delivery problem in the urban delivery system by using crowdsource delivery. The nearest available crowdsources can be utilized to temporarily store the customer parcel if the receiver is not home. The problem was formulized as packing problem to cover the distribution area of the neighborhood relays. Pichka et al. (2018) proposed a model to allows independent contractors, professional and occasional drivers, or crowd-worker to help delivery the customer order in urban area delivery. In this study, the problem was formulized as a two-echelon open routing model.

The research by Kafle et al. (2017) is the closest study to our deterministic model. In Kafle et al. (2017), the crowd-outsourced customer order is determined by the availability of crowdsourcing bids with no consideration of the crowdsources routing simultaneously. The crowdsources availability in terms of bidding may only be focused on the easy tasks preferred by crowdsources. It can limit the cost reduction by the crowdsource delivery integration. Our model determines the crowdsourcing decision based on the balance between the main delivery trucks and the crowdsources assuming they are all always available. Thus, making our model is suitable for the early decision to answer which customer should be outsourced and use the solution to design the bid invitations, accordingly.

The comparison between our study and the latest related research in the area of crowd-delivery is provided in the Table 1.

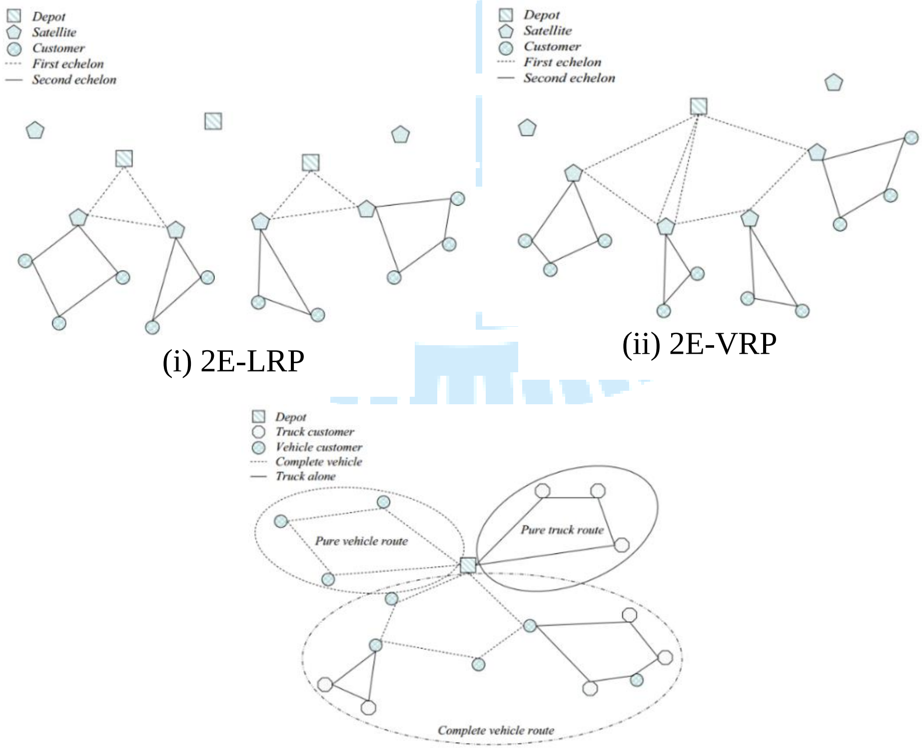
Table 1. Crowdsource delivery research

	Wang et al. (2016)	**Kafle et al. (2017)**	**Akeb et al. (2018)**	**This study (Deterministic)**	**This study (Stochastic)**
Problem Phenomena	Deterministic	Deterministic	Deterministic	Deterministic	Stochastic
Problem Characteristics	Crowdsource delivery by shared reception box (pop station)	Two-echelon delivery and pickup with crowdsource integration	Relay-based delivery for unattended parcel delivery	Two-echelon delivery problem with crowdsource delivery option	Two-echelon delivery problem with crowdsource delivery option and crowdsource transfer uncertainty
Modeling Technique	Minimum cost flow problem	Mixed integer non-linear (MINLP) model Winner determination problem (WDP)	-	MILP model	Two-stage stochastic mixed integer linear model
Decision Variables	• Parcel assignment	• Main fleet route • Crowdsource selection • Transfer point selection	• Neighborhood relay	• Main fleet route • Crowdsource decision • Transfer point selection	• Main fleet route • Crowdsource decision • Transfer point selection • Recourse action route
Heuristics Approach	Several pruning techniques	Tabu search	-	Tabu search	Tabu search.

2.2. Two echelon routing problem

Partial-coverage crowdsource delivery forms a two echelon delivery system in which the main delivery truck covers the first echelon route and the crowdsources perform the second echelon delivery orders. Additional transfer facilities called satellites are located between customer location and depot. A comprehensive review about two-echelon routing problem can be found in Cuda et al. (2015).

Basically, the two-echelon routing problem is categorized into three categories. The two-echelon location routing problem (2E-LRP) which presents the basic form of the two-echelon routing problem with fixed customer location is the first category. The depot and satellites are determined based on the model. The second category is two-echelon vehicle routing problem (2E-VRP). This problem is a special case of 2E-LRP where the locations of satellites and depot are given in advance. The decision problem is to find a good visiting sequence or route. The last category is truck and trailer routing problem (TTRP). This class is also special case of 2E-VRP in which the trucks (from second-echelon operation) are attached to trailers in the first-echelon operations. Figure 4 shows the illustration of three problem types in two-echelon routing problem.



(i) 2E-LRP

(ii) 2E-VRP

(iii) TTRP

Figure 4. Illustration of variants problem in two-echelon routing problem (Cuda et al., 2015)

In this study, parcel relay or transfer point can be considered as satellite which will be selected to perform the crowdsource transfer if at least one crowdsources is assigned to transfer in that particular transfer point. The selection of the satellite in two echelon routing problem belong to the 2E-LRP. In 2E-LRP, there is a clear separation between the first echelon and second echelon inferring the first echelon fleet can only deliver the cargo to the satellite and the end customer delivery will be performed by the second echelon fleet. Our study allows the first echelon fleet to deliver the cargo to the transfer point and/or deliver the customer order directly to the customer location. TTRP has no clear separation between first echelon and second echelon delivery as the first stage fleet can deliver the cargo directly to customer location. In TTRP, the customer node can be used as a satellite which is different from our study transfer point definition. In this study, the transfer points are public places that can be used as temporary transfer facility. Based on the definition between 2E-LRP and TTRP, our study combines two different two-echelon categories to match our problem definition.

Rothenbacher et al. (2018) designed a novel brach-and-price-and-cut algorithm to solve TTRP with time windows in a multi period planning horizon. A combination of column generation and dynamic programming labeling algorithm were utilized to generate linear relaxation to the formulation. Pichka et al. (2018) proposed two-echelon open location routing problem (2E-OLRP) allowing the main vehicle fleet to not come back to the depot as well as the second echelon fleet to represent the individual contractor, logistics providers, and crowds. They proposed several MILP models based on the index of their decision variables and hybrid heuristics algorithm. Belgin et al. (2018) designed an optimization model to solve 2E-VRP with simultaneous pickup and delivery with three valid inequalities based on the literatures. A hybrid heuristics combining local search and variable neighborhood descend (VND) was proposed to generate a fast and good solution. Zhou et al. (2018) proposed multi-depot 2E-VRP with multi delivery options for the end customer, such as direct delivery by second echelon vehicle fleet and self pick up at intermedieate pick up facilities. A multi-population genetic algorithm was proposed as an efficient approach to generate good solution in a fast computation time. The comparison between our study and the latest related research in the area of two-echelon routing problem is provided in the Table 2.

Table 2. Two-echelon routing problem

	Pichka et al. (2018)	**Belgin et al. (2018)**	**Zhou et al. (2018)**	**This study (Deterministic)**	**This study (Stochastic)**
Problem Phenomena	Deterministic	Deterministic	Deterministic	Deterministic	Stochastic
Problem Characteristics	Two-echelon open location routing problem	Two-echelon problem with simultaneous pickup and delivery	Two-echelon multi-depot routing problem	Two-echelon delivery problem with crowdsource delivery option	Two-echelon delivery problem with crowdsource delivery option and crowdsource transfer uncertainty
Modeling Technique	MILP Model	MILP Model	-	MILP model	Two-stage stochastic MILP model
Decision Variables	• Main fleet route • Second-echelon route Decision to open facility	• Main fleet route • Decision to assign delivery or pickup to the second echelon • Second-echelon route	• Main fleet route • Second-echelon route	• Main fleet route • Crowdsource decision • Transfer point selection	• Main fleet route • Crowdsource decision • Transfer point selection • Recourse action route
Heuristics Approach	Hybrid Simulated Annealing	Variable Neighborhood Descend and Local Search	Hybrid Multi-Population Genetic Algorithm	Tabu search	Tabu search.

2.3. Stochastic routing problem

As mentioned in the introduction, this research considers the uncertainty of crowdsource transfer in the transfer location. Therefore, any uncertainty consideration will transform the delivery problem into the stochastic vehicle routing problem (SVRP). Several related studies in the area of SVRP will be reviewed in this section. Two comprehensive SVRP's literature reviews can be found in Gendreau et al. (1996) and Gendreau et al. (2016). A survey that differentiate SVRP with Dynamic VRP (DVRP) can also be found in Ritzinger et al. (2016).

Basically, there are three categories in SVRP based on the source of the uncertainty. The first category is SVRP with uncertain demand (VRPSD). The VRPSD is the most studied of all SVRPs. In this problem, the focus of the uncertainty is the customer demands treated as the random variables. The distinction between various studies in VRPSD lies in the choice of the recourse policies and solution methodologies. The classical recourse policy in VRPSD is the vehicle returns to the depot to replenish its capacity, then continue to deliver its planned route from the point of failure if delivery failure occurs. The second category is vehicle routing problem with stochastic travel time (VRPSTT). This problem is motivated to capture the nature of congestion, weather condition, different modes travel time, link-dependent travel time, etc. One of the important aspect of the VRPSTT is the customer time windows. In a soft time window, the deviations from time windows are usually penalized. In a such problem, the probability distributions to compute the expected penalty of route need to be formulated. This method can be implemented in the hard time windows.

The third category is SVRP with uncertain customer (VRPSC). The presentation of customers is uncertain or follow certain probability distribution, however, their demand is deterministic. Using the a priori paradigm, the routes are decided in the first stage, and are executed in the second stage while skipping the absent customer. The VRPSC can assume that the presence of customers is revealed prior to the arrival of the vehicle. There are few researches have been done on the VRPSC (Gendreau et al., 2016). The VRPSC also can be extended to have stochastic demand (VRPSCD). In this problem, routes are designed in the first stage, the presence of customers is revealed prior to the execution of the routes. Upon the arrival of the vehicle, the information of customer demands will be discovered. In the second stage, the routes are executed with the skipping absent customer and classical recourse policty will be implemented when vehicle capacity is violated.

The closest category of SVRP to this study problem is the SVRP with uncertain customer. Although the concept of stochastic customer is different from the stochastic crowdsources, the crowdsources can be treated as one type of customers. Still, the differences will create the different recourse action and recourse costs, as the absent customer in SVRP can be treated as the saving opportunity due to the travel time elimination. In our study, the absent crowdsources will generate more travel times since the delivery truck will need to deliver the shipment by themselves to the customer location or impose a penalty. In addition, the additional travel time also may cause the delivery failure or late delivery.

Sungur et al. (2010) considered the courier delivery problem (CDP) with uncertain customer presence and service time. They proposed a model that maximize the coverage of customers and the similarity of the routes by generating the master plan and daily schedules. An insertion-based solution heuristics based on the tabu search heuristics was developed to solve big-size of problem. Heilporn et al. (2011) studied the dial-a-ride problem with stochastic customer and customer delay. In this problem, the customer pickup can have a possibility to be delayed due to the uncertainty. If a customer request is unable to be performed due to the delay, the request will be fulfilled by an alternative service. Ulmer et al. (2015) proposed a MIP model to solve the VRP with stochastic customer. This problem was motivated by the courier service in their operations to deal with additional uncertain requests from more customers during the service. A rollout algorithm (RA) to solve the same problem was proposed one year after (Ulmer et al., 2016). The problems are modeled as markov decision process (MDP) to anticipate the future events in the current decision making process. An integer L-shaped algorithm was proposed to solve the problem. Saint-Guillain et al. (2017) introduced a static-and-stochastic vehicle routing problem (SS-VRP) in which the customer reveal times are stochastic in addition to the stochastic customer data. The problem was motivated by the application of the elderly and disabled people on-demand health care service. A local search algorithm combined by simulated annealing was designed to solve the problem with fast computation time.

To our best knowledge, the most related literature to our study is the study that was proposed by Gdowska et al. (2018). In their study, they extended the deterministic full-coverage crowdsource delivery proposed by Archetti et al. (2016) as a base problem. They proposed a heuristics algorithm which considers the crowdsources probability for accepting the delivery task in a single echelon stochastic routing problem. In our study, we deal with the two-echelon stochastic routing problem with the uncertainty focusing on the realization of crowdsource transfer at the transfer point, after the acceptance of the crowdsources for the task. This problem is formulated as a novel two-stage stochastic MILP model.

Finally, the comparison among our study and the latest related research in the area of stochastic routing problem is presented in the Table 3.

Table 3. Vehicle routing problem with stochastic customer review

	Heilporn et al. (2011)	Ulmer et al. (2015 & 2016)	Saint-Guillain et al. (2017)	Gdowska et al. (2018)	This study (Stochastic)
Problem Characteristics	General dial-a-ride problem to pick up customer	General delivery problem	On demand health care service for elderly and disabled people	Delivery problem with crowdsource delivery options	Two-echelon delivery problem with crowdsource delivery option
Problem Modeling	Two-stage stochastic binary model	Mixed integer model	Two-stage stochastic model	-	Two-stage stochastic MILP model
Decision Variables	• Main fleet route • Main fleet arrival time at each node	• Vehicle fleet route	• Visiting location • Visiting time • Waiting time	• Main fleet route • Crowdsource decision	• Main fleet route • Crowdsource decision • Transfer point selection • Recourse action route
Probabilistic Parameters	Random arrival time of customer Random service time of customer	Customer request	Location of customer and reveal time of the data	Crowdsources accepting the task	Crowdsource and main delivery fleet parcel transfer
Heuristics Approach	Integer L-shaped method	Rollout algorithm and Markov decision process	Local search based on Simulated Annealing	Bi-level stochastic algorithm	Tabu Search

CHAPTER 3 PROBLEM FORMULATION

This chapter presents how this study formulates the problem into optimization or mathematical model. The model will be solved by the blackbox solver to get the optimal solution. In the first section, the problem is formulated into a deterministic model which assumes every aspects of the problem are deterministic and given in the beginning. The stochastic problem formulation is presented as the second section in which one of the aspects or paramters is assumed to be uncertain to account for the realistic condition.

3.1. Deterministic problem formulation

3.1.1. Problem definition

In the deterministic model, the delivery operations are performed by the logistics operator which manages the customer delivery orders (e.g., e-commerce parcels) and operates the delivery fleets. The delivery operation starts from the single depot (e.g. DC, warehouse, retailer, etc.) to all of the customer location. To deliver the customer order, the decision maker has two options. First, the delivery by in-house delivery truck and/or the delivery by utilizing crowd service as the second option.

Crowdsources are assumed as a people who lives or have a daily commuter trip nearby the customer locations. Crowdsources cannot obtain the customer parcel directly from the depot. Instead, the main delivery fleet need to relay the customer parcel to the crowdsources at a transfer location. The transfers location is referred as transfer point, a public available places, such as parking areas, convenience stores, etc. After relaying the parcel, the crowdsources deliver the rest of the distances to the customer location while delivery trucks continue to deliver the customer orders according to the original plan.

Customers are classified into two groups based on the availability of crowd-delivery service. The first group is the customers who are located nearby (e.g. less than two km) the possible transfer point(s). Therefore, the first group has two delivery options. In the second group, all customers are located far from the possible transfer point making the crowds-delivery service unavailable for them. The characteristics of the customer order (e.g. weight, volume, commodity type, etc.) can make the first group customers as the second group customers due to the limitation of crowdsource carrying capacity.

The assignment of crowdsources are one-to-many as one crowd-worker may handle more than one delivery tasks. However, the number of delivery tasks for one crowdsources is limited due to the capacity and/or time limitation. This crowdsources are not required to come back to the transfer point after they finish the delivery tasks making the crowdsource assignment as an open vehicle routing problem (VRP). As for the compensation of their service, the crowdsources will be paid based on the two components, such as a fixed cost per each crowdsources and time-based cost. The illustration of the problem is presented in the Figure 5.

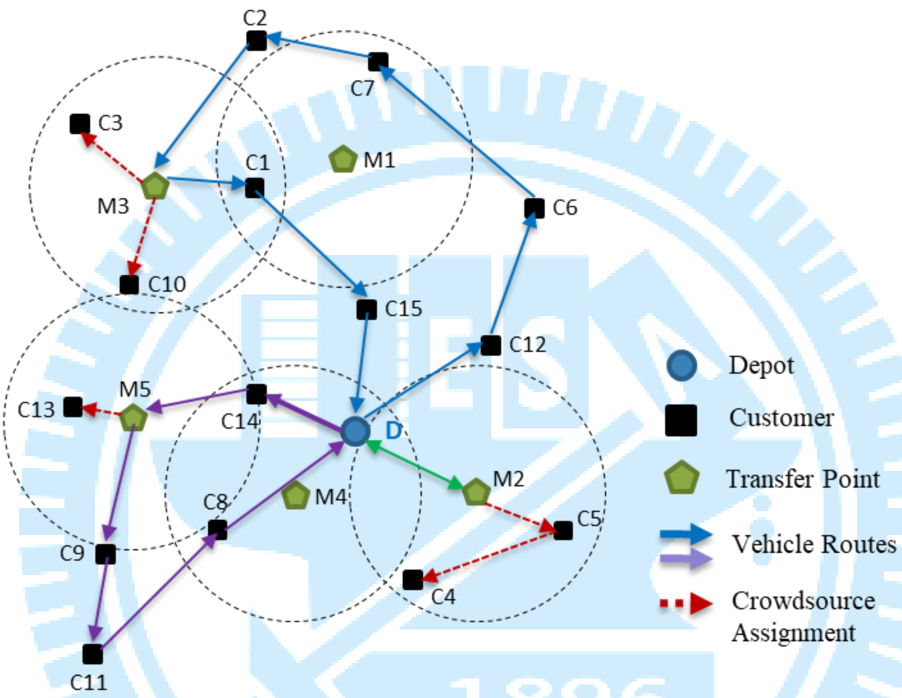


Figure 5. Illustration of crowdsource delivery

3.1.2. Problem formulation

The problem is formulated as MILP with the main decision variables are the selection of outsourced customer, the selection of the transfer locations, the schedule of crowdsource transfer, and the number of required crowd-workers. In addition to the outsourcing decisions, the delivery truck routes for the second group customer and the transfer points are determined.

The objective function is to minimize the total delivery cost containing the delivery fleet cost and the crowdsources cost in (1). For the delivery truck costs, the gasoline cost as a variable of distance or time is considered. The crowdsource cost consisting of the fixed cost and the operational variable cost are also considered. IThe MILP formulation is described in the next following paragrahps.

Sets

N Set of customer nodes.

M Set of transfer points.

N_l Set of customer nodes reachable from transfer point $l, l \in M$.

N^A Set of customer nodes, transfer points, and depot, $N^A = N \cup M \cup \{0\}$.

N^B Set of customer nodes and transfer points, $N^B = N \cup M$.

N^C Set of customer nodes and depot, $N^C = N \cup \{0\}$.

M^A Set of transfer points and depot, $M^A = M \cup \{0\}$.

V Set of vehicles.

B Set of crowd-workers or crowdsources.

Parameter

T_{ij}^f Travel time from node i to node j by vehicles , $i, j \in N^A$.

T_{ij}^c Travel time from node i to node j by crowdsources, $i, j \in N^{\mathrm{B}}$.

T_{iR}^c Travel time from node i to the artificial node R ($T_{iR}^c = 0$), by crowdsources.

G_i Demand of node $i, i \in N$.

Q^v Vehicle maximum capacity.

Q^c Crowdsources Maximum capacity.

C^r Variable cost of vehicle usage.

C^a Crowdsources fixed cost.

C^b Crowdsources variable cost.

L Maximum hours of service.

U Sufficient large number.

Variables

x_{ij}^v Binary variable taking value 1 if vehicle v travels from node i to j, where $v \in V$ and $i, j \in N^A$, and 0 otherwise

a_i^v Accumulated travel time of vehicle v at node i, where $v \in V, i \in N^A$

g_i^v Accumulated load of vehicle v at node i, where $v \in V, i \in N^A$

y_{li}^b Binary variable taking value 1 if crowd-worker b relays at transfer point l and serves customer i which is in the coverage of meeting point l, where $b \in B,\ l \in M, i \in N_l$, and 0 otherwise

$\tilde{x}\,_{ij}^{b}$ Binary variable taking value 1 if crowd-worker b travels from node i to j, where $b \in B$ and $i, j \in N^B$, and 0 otherwise

w_l^b Binary variable taking value 1 if crowd-worker b relays at transfer point l, where $b \in B, l \in M$, 0 otherwise

h_i^b Accumulated travel time of crowd-worker b at node i, where $b \in B, i \in N^B$

$$\min\left(\sum_{i\in N^A}\sum_{j\in N^A}\sum_{v\in V} C^r T_{ij}^f x_{ij}^v + \sum_{l\in M}\sum_{b\in B} C^a w_l^b + \sum_{i\in N^B}\sum_{j\in N}\sum_{b\in B} C^b T_{ij}^c \tilde{x}_{ij}^b\right) \tag{1}$$

subject to:

$$\sum_{j\in N^A}\sum_{v\in V} x_{ij}^v + \sum_{l\in M}\sum_{b\in B} y_{li}^b = 1 \qquad \forall i \in N \tag{2}$$

$$\sum_{j\in N^A}\sum_{v\in V} x_{lj}^v U \geq \sum_{b\in B}\sum_{i\in N_l} y_{li}^b \qquad \forall l \in M \tag{3}$$

$$\sum_{i\in N^A} x_{ik}^v - \sum_{j\in N^A} x_{kj}^v = 0 \qquad \forall k \in N^B, v \in V \tag{4}$$

$$\sum_{i\in N^A} x_{0i}^v \leq 1 \qquad \forall v \in V \tag{5}$$

$$a_j^v \geq a_i^v + T_{ij}^f + (x_{ij}^v - 1)U \qquad \forall i \in N^A, j \in N^B, v \in V \tag{6}$$

$$a_i^v + T_{i0}^f \leq L \qquad \forall i \in N^B, v \in V \tag{7}$$

$$g_j^v \leq g_i^v - G_j - (x_{ij}^v - 1)U \qquad \forall\ i \in N^A, j \in N, v \in V \tag{8}$$

$$g_l^v \le g_i^v - \left(\sum_{j\in N}\sum_{b\in B} y_{lj}^b G_j\right) - (x_{il}^v - 1)U \qquad \forall\, i \in N^A, l \in M, v \in V \tag{9}$$

$$g_j^v \le Q^v \qquad \forall j \in N^B, v \in V \tag{10}$$

$$\sum_{l\in M} w_l^b \le 1 \qquad \forall b \in B \tag{11}$$

$$\sum_{i\in N_l} y_{li}^b \le w_l^b U \qquad \forall l \in M, b \in B \tag{12}$$

$$\sum_{j\in N_l\cup\{l\}} \tilde{x}_{ji}^b = y_{li}^b \qquad \forall l \in M, i \in N_l, b \in B, \tag{13}$$

$$\sum_{i\in N_l} \tilde{x}_{li}^b = w_l^b \qquad \forall l \in M, b \in B \tag{14}$$

$$\sum_{i\in N_l\cup\{l\}} \tilde{x}_{ik}^b - \sum_{j\in N_l\cup\{R\}} \tilde{x}_{kj}^b = 0 \qquad \forall l \in M, k \in N_l, b \in B \tag{15}$$

$$\sum_{b\in B}\sum_{i\in N_l} \tilde{x}_{li}^b \le \left(\sum_{j\in N^A}\sum_{v\in V} x_{lj}^v\right)U \qquad \forall l \in M \tag{16}$$

$$h_l^b \ge a_l^v + \left(w_l^b - 1\right)U \qquad \forall l \in M, b \in B, v \in V \tag{17}$$

$$h_j^b \ge h_i^b + \left(\tilde{x}_{ij}^b - 1\right)U + T_{ij}^c \qquad \forall b \in B, i, j \in N^B \tag{18}$$

$$h_i^b \le L \qquad \forall b \in B, i \in N \tag{19}$$

$$\sum_{i\in N_l} G_i y_{li}^b \le Q^c \qquad \forall b \in B, i \in N, l \in M \tag{20}$$

$$x_{ij}^v \in \{0,1\} \qquad \forall i, j \in N^A, v \in V \tag{21}$$

$$y_{li}^b \in \{0,1\} \qquad \forall l \in M, i \in N_l, b \in B \tag{22}$$

$$\tilde{x}_{ij}^b \in \{0,1\} \qquad \forall i, j \in N^A \cup \{R\}, b \in B \tag{23}$$

$$w_l^b \in \{0,1\} \qquad \forall l \in M, b \in B \tag{24}$$

$$a_i^v, g_i^v \ge 0 \qquad \forall i \in N^A, v \in V \tag{25}$$

$$h_i^b \ge 0 \qquad \forall i \in N, b \in B \tag{26}$$

Constraint (2) enforces every customer node to be visited by a crowd-worker or a delivery truck. Constraint (3) ensures the customer order is relayed by the delivery truck at the transfer point if there is at least one customer order transfer related to the transfer point. The delivery truck flow conservation is defined in constraint (4). Constraint (5) prevents the delivery truck to only be used once. Constraint (6) defines the accumulation of travel time at each node. The regulation of hours of service is defined in constraint (7). The constraints related to the truck capacity are managed in constraints (8) – (10). Constraint (8) defines the load accumulation at each customer node. Constraint (9) determines the load information at each transfer point. The truck load capacity limitation is enforced in the constraint (10).

Constraint (11) ensures the limitation of crowdsources to only be assigned in only one transfer location. Constraint (12) determines that the crowdource service to deliver customer order must begin with crowdsources transfer at the related transfer point. The links between customer and crowd-worker visits are defined in the constraint (13). Constraint (14) enforces the crowd-worker to leave the transfer point after transfer process by taking one of the outbound links at the transfer point. Constraint (15) defines the flow conservation of crowdsources route. Crowdsources are not required to come back to the transfer point, therefore, artificial node which have zero travel distance and travel time is defined at the end of the crowdsources route in constraint (15). Subtour elimination is defined in (14) - (16). Constraint (16) ensures that the crowdsource service must be started only if the delivery truck visit the transfer location.

Constraint (17) - (18) manage the accumulation time at customer nodes and transfer location. Constraint (19) imposes the regulation of hours of service for the crowdsource delivery. Constraint (20) defines the crowdsources maximum load capacity. Binary and non-negativity constraints of the variables are defined in constraints (21) - (26).

3.2. Stochastic problem formulation

The crowdsource parcel transfer between delivery truck and crowdsources is a crucial point in the partial-coverage crowdsource delivery. So far many studies regard this event as a 100% certain event. In reality, this event is subjected to several uncertainties, such as weather condition, traffic condition, customer reliability, etc. Any weather changes or upredicted traffic congestion might delay and/or cancel the transfer event due to limited waiting time of the crowdsource transfer event. As the crowdsource transfer fail, the delivery plan will be distrupted and the benefit of incorporating crowdsources into the delivery plan will be destroyed. Therefore, the uncertainty of crowdsource parcel transfer need to be considered in order to preserve the benefits of crowdsource delivery collaboration.

3.2.1. Problem definition

The stochastic model is an extension of the previous deterministic model. Both of the models share some basic characteristics, such as the definition of customers and crowdsources, the transfer operations at transfer points, and the delivery truck operations. The following paragraphs describe additional problem descriptions which are different from the problem descriptions in deterministic model due to the uncertainty consideration.

As this study considers the uncertainty of crowdsource transfer, the mechanism of crowdsource transfer is modified. Instead of assuming all crowdsource transfers always success, this study only considers two possibilities in the crowdsource transfer event (e.g. success and failure outcome). By this definition, this study disregards any reason of why the event fail or success. In the first outcome when crowdsources transfer success, the crowdsources are able to transfer the parcel in time. Transfer failure will be defined as the second outcome when the crowdsources are not able to complete the parcel transfer in time. In this study, the probability of the crowdsources and main carrier delivery parcel transfer failure is called the crowdsources transfer failure rate.

The crowdsources transfer uncertainty is modeled as success or failure outcome of the transfer process. Let's define P_l as the crowdsource transfer failure rate for any crowdsources delivery task at the transfer point l and $1 - P_l$ as the transfer success rate at transfer point l. The probability of P_l is assumed to be independent among all of transfer point $l, l \in M$. Then, the possible event, ω is denoted as the possible combination of outcomes in all available transfer points with the total events follow $2^{|M|}$, where M is the set of available transfer points. The terms of event and realization (denoted by ω) are used interchangeably to represent the success and failure outcome combinations of available transfer points. As an example, the total events or realizations with the probability of each realizations is illustrated in Table 4.

Table 4. Example of realization and its probability with two available transfer points

Realization $\boldsymbol{\omega}$	**Crowdsrouces transfer outcome**		**Probability of realization** $\boldsymbol{P(\omega)}$
	1st transfer point	2nd transfer point	
1	Success	Success	$P(\omega = \{S, S\}) = (1 - P_{l_1})(1 - P_{l_2})$
2	Success	Failure	$P(\omega = \{S, F\}) = (1 - P_{l_1})P_{l_2}$
3	Failure	Success	$P(\omega = \{F, S\}) = P_{l_1}(1 - P_{l_2})$
$\mathbf{2^{\lvert 2 \rvert} = 4}$	Failure	Failure	$P(\omega = \{F, F\}) = P_{l_1}P_{l_2}$

Where F is crowdsources transfer failure outcome and S is crowdsources transfer success outcome.

In terms of the uncertainty revelation time, the information of crowdsource transfer failure or success are revealed on the spot when the delivery truck arrives at the transfer point or at the time between the delivery truck depart from depot to the arrival of delivery truck at the transfer point. The dynamic re-planning or online re-planning features are assumed to be unavailable due to limited time and resources. As a consequence, the backup plan needs to be defined to mitigate both all possible realizations in the beginning of the planning. The term of recourse strategy is used to describe back up plan strategy in this study.

This study proposes two recourse strategies, namely penalty-only recourse and detour-combined recourse strategies as the backup plan to respond for the failure crowdsource transfer outcome. In the first recourse strategy, the logistics operator is assumed to not taking any further action when crowdsource failure transfer occurs. The un-send customer parcels will be re-delivered on the next day. It is a common practice for logistics operator to re-deliver the parcel on the next day when there is no recipient available. Penalty cost need to be incurred as a cost to re-deliver the parcel on the next day. The penalty cost will be imposed everytime a failure transfer outcome occur.

The second recourse action strategy requires the delivery truck to perform additional trip (detour) to deliver the customer order as the replacement delivery due to the transfer failure between delivery truck and crowdsources. The delivery truck which carry the customer's parcel makes additional trips, starting from the transfer point to every customer location. After making a detour the truck needs to come back to the original delivery after the detour trip finish. One or multiple customer orders in the detour trip can be skipped and penalty will be imposed if there is no time left to avoid the violation driver hour of service. The illustration about detour-combined trvoutdr strategy is provided in Figure 6. As the delivery truck initiates detour plan, some distances will be added and some distances will be removed from the original delivery plan. The distance from the transfer point to the next node can be skipped because the detour trip takes different path. The term skipped distance is used to represent the reduced distance from the original plan as the impact of truck detour. As an example in Figure 6, the distance between transfer point to customer 3 or C3 is omitted when failure transfer occurs and truck detour is initiated.

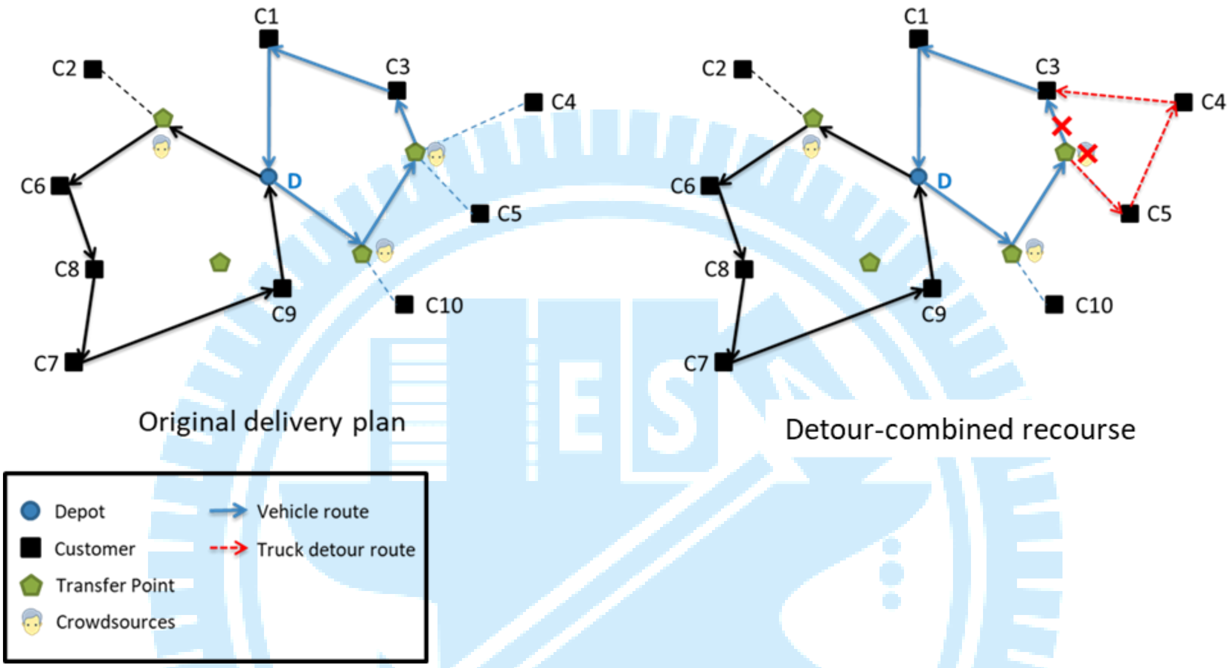


Figure 6. Illustration of detour-combined recourse strategy

The stochastic model assumes the crowdsources transfer failure rate depend on the transfer point making any crowdsources transfer at the same transfer point will have the same probability. The crowdsource transfer probabilitiy is assumed to be independent across all transfer points. The stochastic model also treats the crowdsources delivery task as the assignment problem instead of routing problem to avoid model complexity.

3.2.2. Problem formulation

The problem is formulated as the two-stage stochastic model. In the first stage, the decision of delivery route, number of delivery trucks, crowdsource assignments, and the crowdsource transfer place and schedule are made. In the second stage, the recourse action will be formulated as the consequence of crowdsource transfer failure. The illustration of decision timeline is presented in Figure 7.

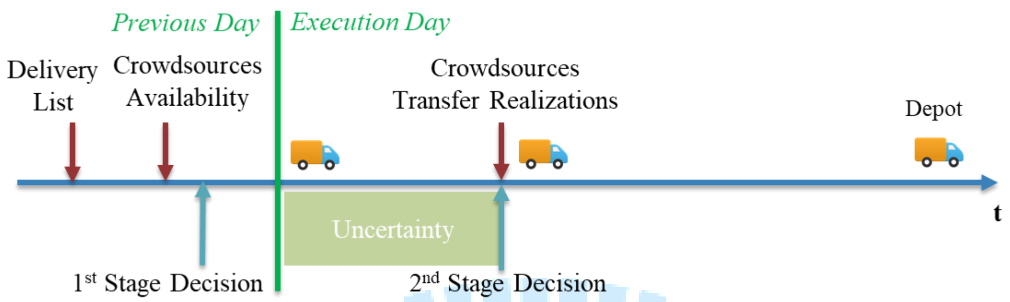


Figure 7. Illustration of decision timeline

As this study proposes two recourse strategies, the model formulation also divided into two parts based on the recourse strategy. In general, the objective function is to minimize the delivery costs consisting of operational vehicle costs considered in the first stage objective function and the expected recourse costs in the second stage model. In the second stage model, the objective function contains the expected cost of crowdsources payment, detour cost, and penalty costs for omitting customer. The crowdsources payment consists of distance-based cost and usage-based cost.

Since the stochastic models are the extension of deterministic model, some of the set and parameter definitions from the deterministic model are re-used. New variable sets are defined to accommodate new problem features. The new sets, parameters, and variables of the stochastic model are defined as follows.

<u>Sets</u>

Ω	Set of realizations.

<u>Parameters</u>

α	Penalty cost for omitting a customer.
P_ω	Probability of realization $\omega, \omega \in \Omega$.

<u>Uncertain Parameters</u>

$R_{l\omega}$ Binary parameter to indicate crowdsource transfer failure at transfer point l for any given realization ω, where $l \in M, \omega \in \Omega$.

Variables

First Stage

x_{ij}^v Binary variable taking value 1 if vehicle v travels from node i to j, where $v \in V$ and $i, j \in N^A$, 0 othersiwse

y_{li} Binary variable taking value 1 if crowd-delivery transfer at transfer point l and serves customer i which is in coverage of meeting point $l, l \in M, i \in N_l$. And 0 otherwise

a_i^v Travel time accumulation of vehicle v at node i, where $v \in V, i \in N^A$

g_i^v Customer demand accumulation of vehicle v at node i, where $v \in V, i \in N^A$

Second Stage

$z_{ij\omega}^v$ Binary variable taking value 1 if vehicle v travels from node i to j as a replacement of failed transfer for any given realization ω, where $i \in N^B, j \in N^A, v \in V, \omega \in \Omega$, and 0 otherwise

$c_{l\omega}$ Binary variable taking value 1 if there is a failure transfer at transfer point l for any given realization of ω, where $l \in M, \omega \in \Omega$, and 0 otherwise

$r_{l\omega}^v$ Distance reduction of vehicle v due to replacement tour at transfer point l for any given realization of ω, where $v \in V, l \in M, \omega \in \Omega$.

$h_{i\omega}^v$ Accumulated travel time of vehicle v at node i in the replacement tour for any given realization ω, where $i \in N, v \in V, \omega \in \Omega$.

$o_{i\omega}$ Binary variable taking value 1 if customer i is omitted in the second stage for any given realization ω, where $i \in N, \omega \in \Omega$, and 0 otherwise

3.2.2.1. **Penalty-only recourse model**

In the penalty-only recourse model, the penalty cost will be imposed everytime failure transfer occurs. The problem is formulated as a MILP in which the penalty cost is a part of the decision of outsourcing customer order. The extensive form of the penalty-only recourse model is presented as:

<u>Objective Function</u>

$$\min \sum_{i\in N^A}\sum_{j\in N^A}\sum_{v\in V} T_{ij}^{f} C^{r} x_{ij}^{v}$$

$$+\sum_{\omega\in\Omega} P(\omega)\left\{\sum_{l\in M}\sum_{i\in N_l} \alpha R_{l\omega} y_{li} + \sum_{l\in M}\sum_{i\in N_l} (1-R_{l\omega})\,(C^{a}+C^{b}T_{li}^{c}) y_{li}\right\} \tag{27}$$

Subject to:

$$\sum_{j\in N^A}\sum_{v\in V} x_{ij}^{v} + \sum_{l\in M} y_{li} = 1 \qquad \forall i\in N \tag{28}$$

$$\left(\sum_{j\in N^A}\sum_{v\in V} x_{lj}^{v}\right) U \geq \sum_{i\in N_l} y_{li} \qquad \forall\, l\in M \tag{29}$$

$$\sum_{j\in N^A} x_{lj}^{v} \leq \sum_{i\in N_l} y_{li} \qquad \forall\, l\in M, v\in V \tag{30}$$

$$\sum_{i\in N^A} x_{ik}^{v} - \sum_{j\in N^A} x_{kj}^{v} = 0 \qquad \forall k\in N^B, v\in V \tag{31}$$

$$\sum_{i\in N^A} x_{0i}^{v} \leq 1 \qquad \forall v\in V, i\in N \tag{32}$$

$$a_j^{v} \geq a_i^{v} + (x_{ij}^{v}-1)U + T_{ij}^{f} \qquad \forall v\in V, j\in N^B, i\in N^A \tag{33}$$

$$L \geq a_i^{v} + T_{i0} \qquad \forall v\in V, i\in N^B \tag{34}$$

$$a_l^{v} + y_{li} T_{li}^{c} \leq L \qquad \forall l\in M, i\in N_l, v\in V \tag{35}$$

$$g_j^{v} \leq g_i^{v} - G_j - (x_{ij}^{v}-1)U \qquad \forall\, i\in N^A, j\in N^B, v\in V \tag{36}$$

$$g_l^{v} \leq g_i^{v} - \left(\sum_{j\in N} y_{lj} G_j\right) - (x_{il}^{v}-1)U \qquad \forall\, i\in N^A, l\in M, v\in V \tag{37}$$

$$g_j^v \leq Q \qquad \forall j \in N^A, v \in V \tag{38}$$

$$x_{ij}^v \in \{0,1\} \qquad \forall i,j \in N^A, v \in V \tag{39}$$

$$y_{li} \in \{0,1\} \qquad \forall i \in N, l \in M \tag{40}$$

$$a_i^v, g_i^v \qquad \forall i \in N^B, v \in M \tag{41}$$

The objective function in Constraint (27) consists of the distance-based cost and the expectation of recourse costs, which are crowdsources reward and penalty cost. Constraint (28) enforces every customer node to be visited by a crowd-worker or the delivery truck. Constraints (29) - (30) ensure the customer order is relayed by the delivery truck at the transfer point if there is at least one customer order transfer related to the transfer point. The delivery truck flow conservation is defined in constraint (31). Constraint (32) prevents the delivery truck to only be used once. Constraint (33) defines the accumulation of travel time at each node. The regulation of hours of service is defined in constraint (34). Constraint (35) defines the arrival time of the crowdsources must be before the end-of-day limitation. Constraint (36) defines the delivery truck load information at each customer node, whereas the transfer point load accumulation is defined in constraint (37). Constraint (38) enforces delivery truck load capacity limitation. Binary and Non-negativity constraints of the variables are defined in constraints (39) - (41).

3.2.2.2.**Detour-combined recourse model**

In the next recourse strategy, a detour-combined recourse by initiating additional truck detour trip to replace failure transfer is defined. This model is an extension of the penalty-only recourse strategy model adding cost components related to the truck detour in the objective function, such as distance-based cost and skipped distance cost in (42), as well as several additional second-stage constraints in (43) - (60). A penalty for deliberately skip the customer during the detour trip is defined to account for unfeasibility because of hours of service limitation. The extensive form of the detour-combined recourse model is presented as follow.

<u>Objective Function</u>

$$\min \sum_{i\in N^A}\sum_{j\in N^A}\sum_{v\in V} T_{ij}^f C^r x_{ij}^v + \sum_{\omega\in\Omega} P(\omega)\left\{\sum_{l\in M}\sum_{i\in N_l} \alpha R_{l\omega} o_{i\omega} + \sum_{l\in M}\sum_{i\in N_l}(1-R_{l\omega})\,(C^a + C^b T_{li}^c) y_{li} + \sum_{v\in V}\sum_{i\in N^A}\sum_{j\in N^A} T_{ij}^f C^r z_{ij\omega}^v - \sum_{l\in M}\sum_{v\in V} C^r s_{l\omega}^v\right\} \tag{42}$$

Subject to:

Constraints (28) - (41)

$$\sum_{j\in N^A}\sum_{v\in V} z_{ij\omega}^v + o_{i\omega} = \sum_{l\in M} y_{li} R_{l\omega} \qquad \forall i \in N, \omega \in \Omega \tag{43}$$

$$(y_{li} - o_{i\omega}) R_{l\omega} \le c_{l\omega} \qquad \forall l \in M, i \in N_l, \omega \in \Omega \tag{44}$$

$$\left(\sum_{j\in N^A}\sum_{v\in V} z_{lj\omega}^v\right) \ge c_{l\omega} \qquad \forall l \in M, \omega \in \Omega \tag{45}$$

$$\sum_{i\in N_l\cup\{l\}} z_{ij\omega}^v \ge x_{lj}^v - (1 - c_{l\omega})U \qquad \forall l \in M, j \in N^A, v \in V, \omega \in \Omega \tag{46}$$

$$\sum_{i\in N_l\cup\{l\}} z_{ik\omega}^v - \sum_{j\in N^A} z_{kj\omega}^v \ge c_{l\omega} - 1 \qquad \forall l \in M, k \in N_l, v \in V, \omega \in \Omega \tag{47}$$

$$\sum_{i\in N_l} z_{li\omega}^v \le \left(\sum_{j\in N^A} x_{lj}^v\right) U \qquad \forall l \in M, v \in V, \omega \in \Omega \tag{48}$$

$$\sum_{v\in V}\sum_{i\in N_l} z_{li\omega}^v \le 1 \qquad \forall l \in M, \omega \in \Omega \tag{49}$$

$$r_{l\omega}^v \le \sum_{j\in N^A} x_{lj}^v T_{lj}^f \qquad \forall l \in M, v \in V, \omega \in \Omega \tag{50}$$

$$r_{l\omega}^v \le c_{l\omega} U \qquad \forall l \in M, v \in V, \omega \in \Omega \tag{51}$$

$$\left(\sum_{i\in N_l} y_{li} - \sum_{j\in N_l} o_{j\omega}\right) U \ge r_{l\omega}^v \qquad \forall l \in M, v \in V, \omega \in \Omega \tag{52}$$

$$h^v_{j\omega} \geq h^v_{i\omega} + (z^v_{ij\omega} - 1)U + T^f_{ij} \qquad \forall i \in N^B, j \in N, v \in V, \omega \in \Omega \qquad (53)$$

$$h^v_{l\omega} \geq a^v_l + \left(\sum_{j \in N^A} z^v_{lj\omega} - 1\right)U \qquad \forall l \in M, v \in V, \omega \in \Omega \qquad (54)$$

$$h^v_{i\omega} \leq L \qquad \forall i \in N, v \in V, \omega \in \Omega \qquad (55)$$

$$z^v_{ij\omega} \in \{0,1\} \qquad \forall i, j \in N^B, v \in V, \omega \in \Omega \qquad (56)$$

$$c_{l\omega} \in \{0,1\} \qquad \forall l \in M, \omega \in \Omega \qquad (57)$$

$$r^v_{l\omega} \geq 0 \qquad \forall l \in M, v \in V, \omega \in \Omega \qquad (58)$$

$$h^v_{i\omega} \geq 0 \qquad \forall i \in N, v \in V, \omega \in \Omega \qquad (59)$$

$$o_{i\omega} \in \{0,1\} \qquad \forall i \in N, \omega \in \Omega \qquad (60)$$

Constraint (43) ensures the delivery truck visits all customers which are involve in the crowdsources failed transfer, otherwise the penalty will be incurred. Constraint (44) defines the variable $c_{l\omega}$ to represent the failed crowdsource transfer at transfer point l event ω. Constraint (45) makes sure the truck detour trip starts from the related transfer point. Constraint (46) forces the truck detour trip to end at the next customer based on the original plan. Constraint (47) manages the route flow conservation of truck detour. Constraint (48) ensures the delivery truck that carries the parcel and the truck perform the detour are the same truck. Constraint (49) limits all truck detours to only have one trip. The constraints about distance skipping are defined by the constraints (50) - (52). Constraint (50) defines the amount of distance to skip. Constraint (51) limits the distance skipped to only have positive value when there is a crowdsource transfer failure. Constraint (52) nullifies the value of distance skipped if the truck detour is canceled due to the decision to omit the customer. Constraints (53) - (55) deal with the time-related regulations in the truck detour. Constraint (53) determines the accumulated travel time in the truck detour trip. The accumulated travel time must start from the time delivery truck arrives at the transfer point in constraint (54). Constraint (55) ensures the arrival time of truck detour in customer node must be less than the hours of service limitation. Biniary and non-negativity constraints of the variables are defined in constraints (56) - (60).

CHAPTER 4 SOLUTION ALGORITHM

In this chapter, the heuristic algorithms are presented to generate good solution quality with fast computation time. The proposed algorithms are classified into solution algorithm for deterministic problem and algorithm for stochastic problem. For each heuristic algorithm in the deterministic and stochastic problem, it consists of construction algorithm to generate initial solution and improvement algorithm to improve initial solution are proposed. Some of the definition and mechanism in both algorithms can be used interchangeably.

The improvement heuristics development was inspired by the TS algorithm. It has been showed as a good and efficient heuristics approach to solve several classifications of routing problem. TS was introduced by Glover (1986) to solve the classical routing problem. It has been reported to achieve the best results for various benchmarking instances in routing problem (Barbarosoglu and Ozgur, 1999). A combination between serveral meta-heursitics algorithms (e.g. genetic algorithm, simulated annealing) (Kafle et al., 2017) were reported to improve the efficiency and effectiveness of TS, including the integration of several local search operators (e.g. 2-opt and 3-opt) (Wang et al., 2017). In two-echelon routing problem, several variants of 2E-VRP, 2E-LRP, and TTRP have been reported to successfully implement TS-based algorithm (Chao, 2002; Scheuerer, 2006; Nguyen et al., 2012; Kafle et al., 2018). One of the advantages of TS algorithm is the ease to modify based on the problem characteristics. In this study, the solution candidates or neighborhood solution search are associated to the randomly selected nodes. The following sub-sections describe the detail of solution procedure.

4.1. Heuristic algorithm for deterministic problem

4.1.1. Solution representation and evaluation

In the heuristics algorithm development, our study defines the solution representation consisting of a solution for vehicle routes, $\bar{\mathcal{X}}$ and a solution for crowdsources route, $\bar{\mathcal{Y}}$. For each vehicle v, a visiting sequence of nodes is determined, starting from depot, s_0^v as the first node, ending at the last visiting node $s_{F_v}^v$ (e.g. customer or a transfer point) with the final node as a depot, $(s_{F_v+1}^v)$. F_v is defined as the number of nodes in route v excluding the depot. $V_{\bar{\mathcal{X}}}$ is also defined as selected vehicles in vehicle route set, $\bar{\mathcal{X}}$, $V_{\bar{\mathcal{X}}} \subseteq V$. As for representing crowdsource routes, vector $\bar{\mathcal{Y}}$ is defined containing crowdsourcing assignment $\mathcal{Y}_b$ $(b \in B_{\bar{\mathcal{Y}}})$. Each $\mathcal{Y}_b$ starts from the transfer point related to crowdsource b, (s_0^b), goes to all customers defined in the current sequence and ends at the last assignment $s^b_{F_b}$, as F_b is the number of crowdsources assignment for crowdsource b and $B_{\bar{\mathcal{Y}}}$ is the set of selected crowdsources used in the solution $\bar{\mathcal{Y}}$, $B_{\bar{\mathcal{Y}}} \subseteq B$. The solution representations are presented as follow.

$$\bar{\mathcal{X}} = \left(\mathcal{X}_1, \mathcal{X}_2, \dots, \mathcal{X}_v, \dots, \mathcal{X}_{|V_{\bar{\mathcal{X}}}|}\right);\ \mathcal{X}_v = \left(s^v_0, s^v_1, \dots, s^v_{F_v}, s^v_{F_v+1} = s^v_0\right)$$

$$\bar{\mathcal{Y}} = \left(\mathcal{Y}_1, \mathcal{Y}_{2,} \dots, \mathcal{Y}_b, \dots, \mathcal{Y}_{|V_{\bar{\mathcal{X}}}|}\right);\ \mathcal{Y}_b = \left(s^b_0, s^b_1, \dots, s^b_{F_b}\right)$$

The objective function $F(\bar{\mathcal{X}}, \bar{\mathcal{Y}})$ to evaluate the solution is defined in (61). The component of the objective function is similar to the objective function in (1) adding the penalty terms as a representation of the constraints violation, such as vehicle load capacity and hour of services constraint. For the feasible solution, the penalty terms are set to zero. Penalty parameter β is defined as a non-negative parameter to balance the penalty value, starting from initial value of 1 and adjusted dynamically based on the progress of the solution throughout the iteration.

$$\begin{aligned}\mathcal{F}(\bar{\mathcal{X}}, \bar{\mathcal{Y}}) = {} & \sum_{v \in V_{\bar{\mathcal{X}}}} \sum_{i=0}^{F_v} C^r T^f_{s^v_i s^v_{i+1}} + \sum_{b \in V_{\bar{\mathcal{X}}}} \sum_{i=0}^{F_b - 1} C^b T^c_{s^b_i s^b_{i+1}} + \sum_{b \in B_{\bar{\mathcal{Y}}}} C^a \\ & + \beta \sum_{v \in V_{\bar{\mathcal{X}}}} \left(\left[\left(\sum_{i=1}^{F_v} G_{s^v_i} \right) - Q^v, 0 \right]^+ + \left[\left(\sum_{i=0}^{F_v} T^f_{s^v_i s^v_{i+1}} \right) - L, 0 \right]^+ \right) \\ & + \beta \sum_{b \in B_{\bar{\mathcal{Y}}}} \left(\left[\left(\sum_{i=1}^{F_b} G_{s^b_i} \right) - Q^c, 0 \right]^+ + \left[\left(\sum_{i=0}^{F_b - 1} T^C_{s^b_i s^b_{i+1}} \right) - L, 0 \right]^+ \right)\end{aligned}$$

(61)

4.1.2. Construction algorithm

The objective of construction algorithm is to generate an initial feasible solution with fast computation time. Later, it will be used as an input for the improvement heuristics to get the best solution. The focus is to balance the usage of crowdsourcing service and the usage of delivery fleets by carefully select the promising customers for crowdsourcing. This study defines two classification of customer orders, the customer orders which are delivered by crowdsources service and the customer orders which are delivered by delivery fleets. The combination of the nearest neighbor procedure and the sweep algorithm are designed to construct with the vehicle routes and the modified saving algorithm is used to manage the route for crowdsourced customer.

The main idea of construction heuristics is to outsource the customers which located far away from its closest neighbor and depot, assuming at least one available transfer point near them. Thus, the remote and low-density deliveries by the delivery trucks can be reduced to save the costs, assuming the crowdsource costs are lower than the operational costs of delivery trucks. Based on this idea, the customer location ratings in terms of how distant the customers from its neighbor and depot are calculated in (62), consisting of the n nearest neighbors's average distance and the distance from the depot. Where $f_i(j)$ is defined as the j^{th} element of the ascending-ordered set based on distance for node i's neighbors, $i \in N^B$. Based on the initial tuning, parameter n is set to be 3. The construction heuristics procedures are presented as follows.

$$W_i = \left(\frac{\sum_{j=1}^{n} T^f_{i,f_i(j)}}{n} \right) + T^v_{0i} \qquad (62)$$

Step 1: **Node Evaluation for Classification.** Based on the customer location rating in (62), select the first λ customers to be outsourced as the set of R^c. Set the value of λ to 1. The set of R^m is defined as the set of transfer points related to the selected outsourced customers in R^c. The nearest transfer point is selected if there are multiple transfer points available for one customer.

Step 2: **Main Truck Route Generation.** Based on the set of R^c, the customer order which are delivered by delivery fleets as vehicle routes customer are defined, $(N \backslash R^c) \cup R^m$. Execute the combination of the sweep procedure (Huang et al., 2018) and the nearest neighbor procedure (Hurkens et al., 2004) in Appendix 1 to construct the main vehicle routes with the respect of truckload capacity and hours of services constraints. The transfer point load is accumulated by the total customer demand which are assigned to relay in the transfer point,

Step 3: **Crowdsource Routing.** Define the set $R^c_l, R^c_l \subseteq R^c$ as the set of crowdsourced customers in transfer point l. Generate the crowdsource assignment based on the combination of modified saving method from Ghiani et al. (2004) in Appendix 2.

Step 4: **Intensification.** Obtain and save the objective function in (61). Re-do the Steps 1-3 and add the value of λ, $\lambda \coloneqq \lambda + 1$ if $\lambda < \Lambda$; Otherwise go to Step 5.

Step 5: **Termination.** The initial solution $(\bar{\mathcal{X}}, \bar{\mathcal{Y}})'$ is the initial solution as the best solution among Λ different solutions based on Step 1-4.

In the construction heuristic procedure, the parameter value λ is defined as the number of outsourced customers. The different solutions of different outsourced customer orders can be examined by increasing the value of λ from 1 to Λ. The maximum value of Λ is determined as $\min(|N|, 2|M|)$ by the initial experiment to balance the solution quality and computation time.

4.1.3. Improvement algorithm

4.1.3.1. **Neighborhood definition**

In the neighborhood structure, the solution candidate as a neighborhood solution are searched. Based on the current solution, $\bar{\mathcal{X}}, \bar{\mathcal{Y}}$, three nodes are selected randomly. The selected nodes is categorized in the the node category to defines the possible movements in Table 5. In general, three node categories are defined, such as category V containing the customer node in the vehicle route, category C containing the customer nodes in the crowdsource route, and category T containing the transfer point node in the vehicle route. A random selection of movement is implemented if multiple movements are possible for one node category. The execution of movements is determined by the sequence of random selection based on Table 5.

This study classifies the movements (or later referred to as search operator) into four categories based on the node movements, namely the inter-route movements, intra-route movements, the crowd-only search movements, the vehicle-crowdsource movements. The inter-route and intra-route movements deal with the optimization of vehicle routes. The optimization of crowdsource route is managed by the crowd-only search movements. The node assignment transfers between vehicle route and crowdsource route are defined to advance the outsourcing decision in the vehicle-crowdsource movements. The movements or search operators are described in the following paragraphs.

Table 5. Possible search operator by various combination of node categories

No	Node Category			Possible Search Operators
	1st	2nd	3rd	
1.	V/T	V/T	C	Intra-route operator: Exchange, Insertion VV, 2-opt Inter-route operator: Shift(1,0), Shift(2,0), SwapVV(1,1)
2.	V/T	V/T	V/T	Intra-route operator: Exchange, Insertion VV, 2-opt Inter-route operator: Shift(1,0), Shift(2,0), SwapVV(1,1), SwapVV(2,1) ,
3.	V	C	V/C/T	Vehicle-crowdsource operator: Insertion VC and Swap VC
4.	C	C/T	V/C/T	Crowd-only operator: Break, Change Transfer Point, Re-insertion Vehicle-crowdsource operator: Insertion CV
5.	C	V	V/C/T	Crowd-only operator: Break, Change Transfer Point, Insertion CV Vehicle-crowdsource operator: Insertion CV and Swap CV
6.	T	C	V/C/T	Vehicle-crowdsource operator: Destroy

Intra-route search operators use the first two randomly selected nodes to optimize the inner route of the vehicle route as illustrated in Figure 8.

Insertion VV: This search operator reinserts the first selected node to the best but different position from the current position in the same vehicle route. (Figure 8-A).

Exchange: This search operator exchanges the position of the first two nodes in the same vehicle route (Figure 8-B).

2-Opt: This search operator utilizes the first two nodes which are non-adjacent nodes in the same vehicle route. Two non-adjacent arcs (from the selected nodes) are removed and replaced with the new and reversed (path) arcs (Figure 8-C).

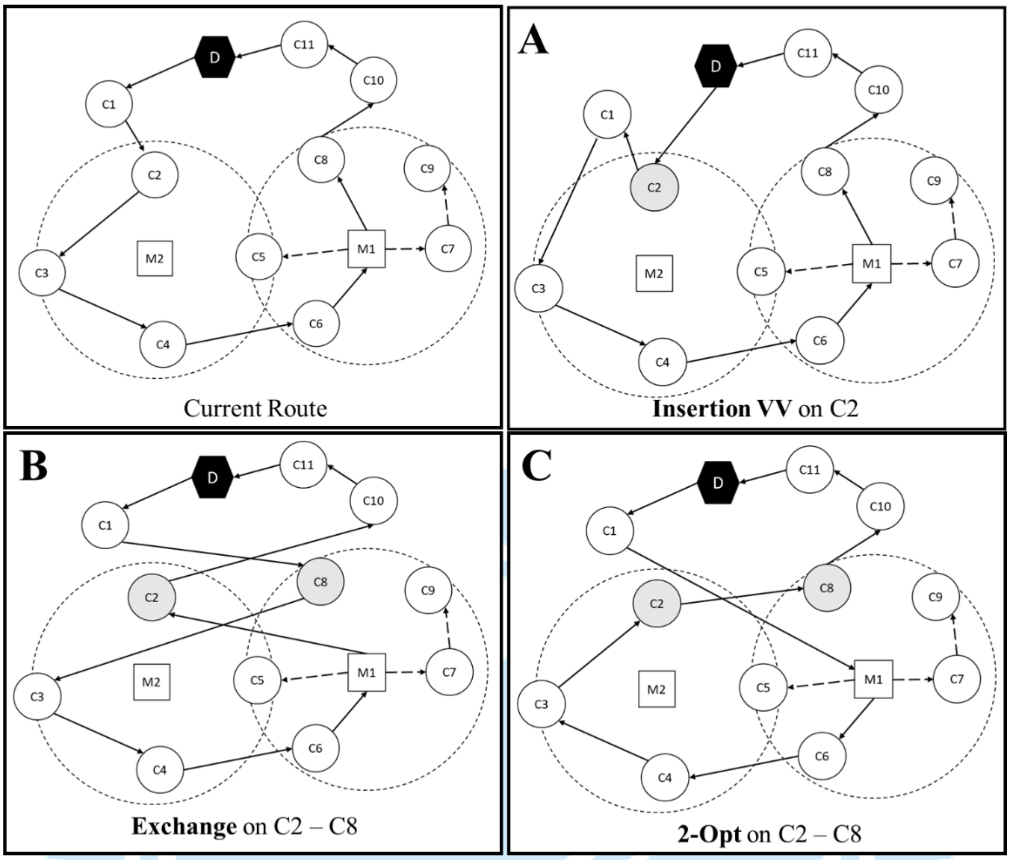


Figure 8. Illustration of the intra-route search operators

Inter-route search operators use three selected nodes. When a transfer point (Category T) is relocated from one vehicle route to another vehicle route, the associated crowdsources assignment will also be relocated together with the transfer point and the crowdsource route remains unchanged. The illustrations of these search operators are provided in Figure 9 .

Swap VV(1,1): This search operator utilizes the first two nodes covered in two different vehicle routes. Each of these nodes is deleted from their current routes and re-inserted at the best position of the other route (Figure 9-A).

Swap VV(2,1): This search operator utilizes three nodes in two different vehicle routes (one route may contain two selected nodes). Each of these nodes is deleted from their current routes. and re-inserted at the best position of the other route (Figure 9-B).

Shift(1,0): This search operator utilizes one node in a vehicle route. The node is deleted and it is re-inserted to the best location in the nearest vehicle route. (Figure 9-C).

Shift(2,0): This search operator utilizes two nodes in a vehicle route. Two nodes are deleted from the same vehicle route and it is re-inserted to its best location in its nearest vehicle route (See Figure 9-D).

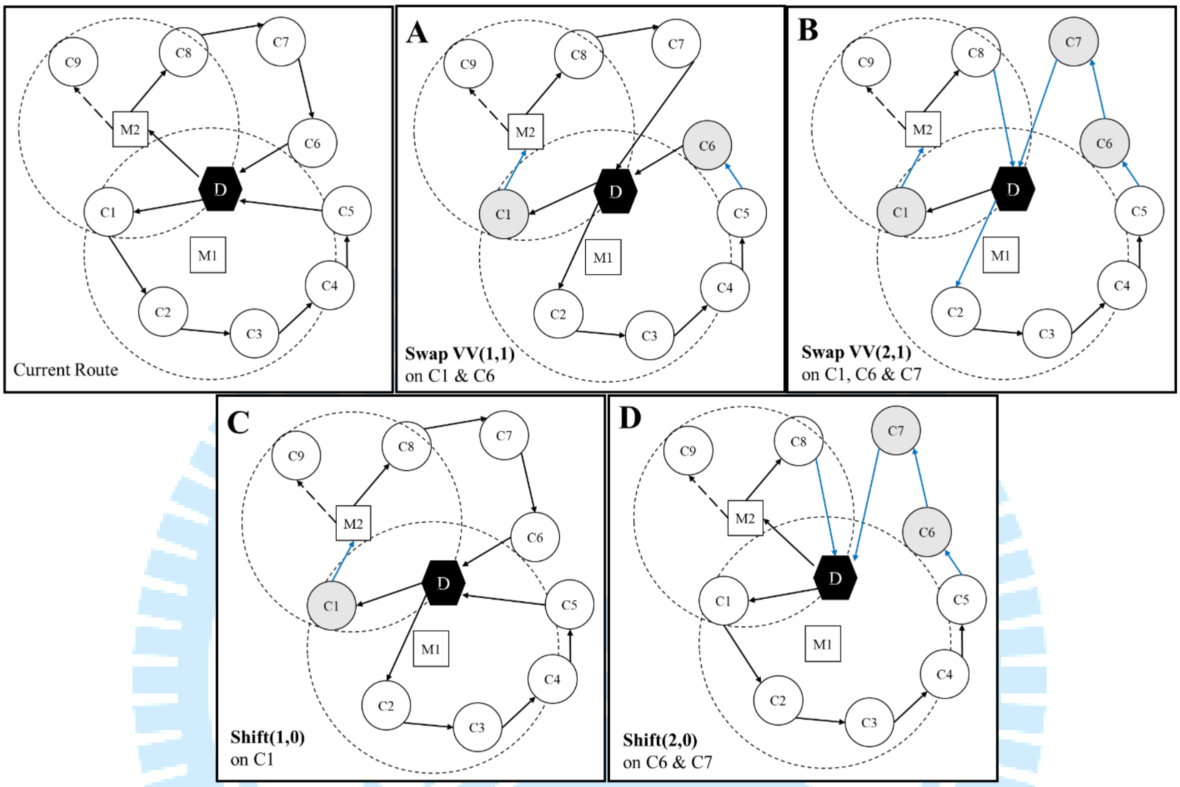


Figure 9. Illustration of the inter-route search operators

The crowd-only search operators utilize the first selected node to improve the crowdsource assignment. The illustrations of these search operators are presented in Figure 10.

Re-insertion: This search operator utilizes the first node in its current crowdsource route. The selected node is removed and re-inserted at its best position in the same crowdsources route or different crowdsource route containing the same transfer point associated with the node (Figure 10-A).

Break: This search operator utilizes the first node in its current crowdsource route. The selected node is removed from its current crowdsource route. The new created crowdsourced is generated with the same transfer point containing only the selected node (Figure 10-B).

Transfer Point Change: This search operator utilizes the first node in its current crowdsource route. The selected node is moved from one transfer point to another transfer point related to the selected node (Figure 10-C).

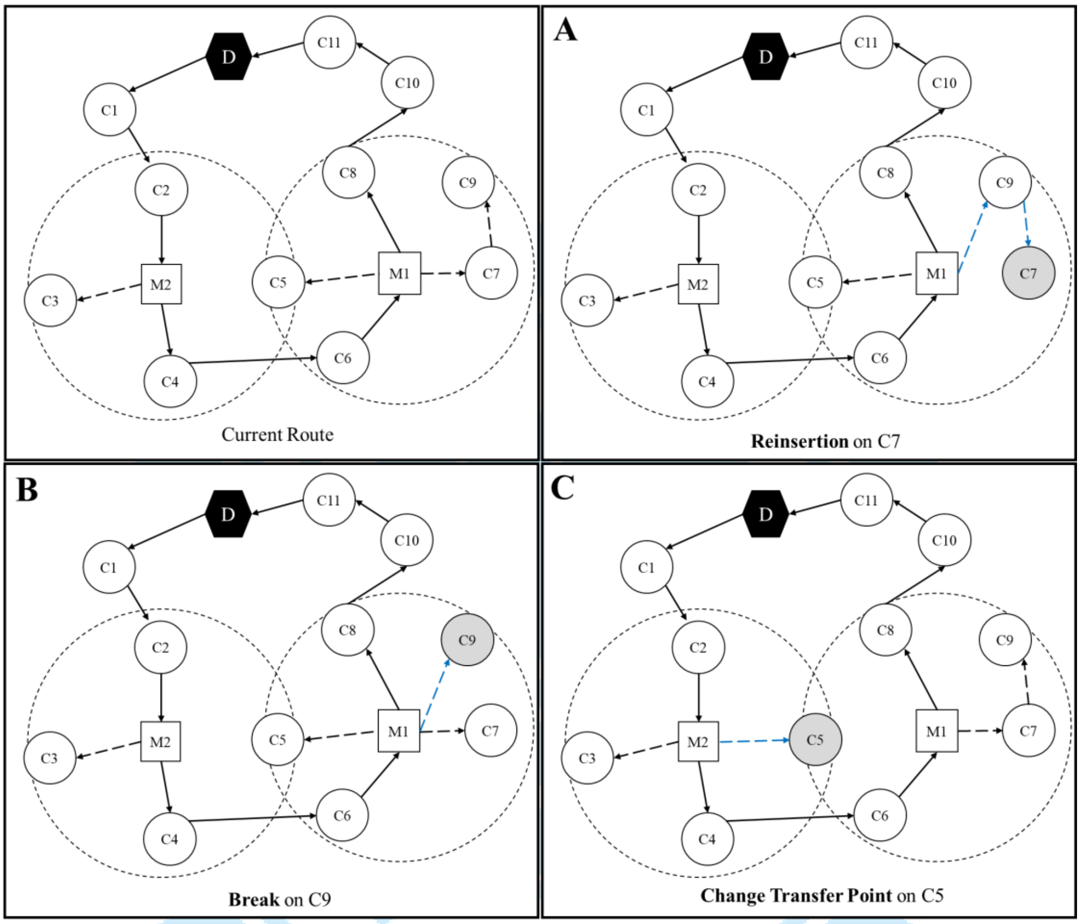


Figure 10. Illustration of crowd-only search operators

The vehicle-crowdsource transfer search operators are defined to manage the search involving both the nodes in the crowdsource route and vehicle route. The direction of node transfers between vehicle route from/to crowdsource route are abbreviated as "VC" and "CV". The illustrations of these search operators are provided in Figure 11.

Insertion VC: This search operator utilizes the first two selected nodes in which the first node is located in the vehicle route and the second node is located in the crowdsource route. The first node is removed from the vehicle route and inserted to the nearest related crowdsource route if possible. The transfer point related to the selected node have to be available in any vehicle route; otherwise an insertion of transfer point to the closest feasible vehicle route at the best location must be initiated (Figure 11-A).

Insertion CV: This search operator utilizes the first two selected nodes in which the first node is located in the crowdsource route and the second node is located the vehicle route. The first node is removed from the crowdsource route and inserted to the closest vehicle route at its best position. The transfer point related will be removed if there is no assignment in it (Figure 11-B).

Swap VC: This search operator utilizes the first two selected nodes in which the first node is located in the vehicle route and the second node is located in the crowdsource route. Two sequential operations are executed, first is the Insertion VC to the first selected node and second is Insertion CV to the second selected node.

Swap CV: This search operator utilizes the first two selected nodes in which the first node is located in the crowdsource route and the second node is located the vehicle route. Two sequential operations are executed, first is the Insertion CV to the first selected node and second is Insertion VC to the second selected node.

Destroy: This search operator utilizes one selected node which is a transfer point. The transfer point is removed from its current vehicle route and the crowdsource assignments related it are released. All customer nodes related to the released crowdsource assignments are re-inserted to its closest feasible vehicle route at its best position (Figure 11-C).

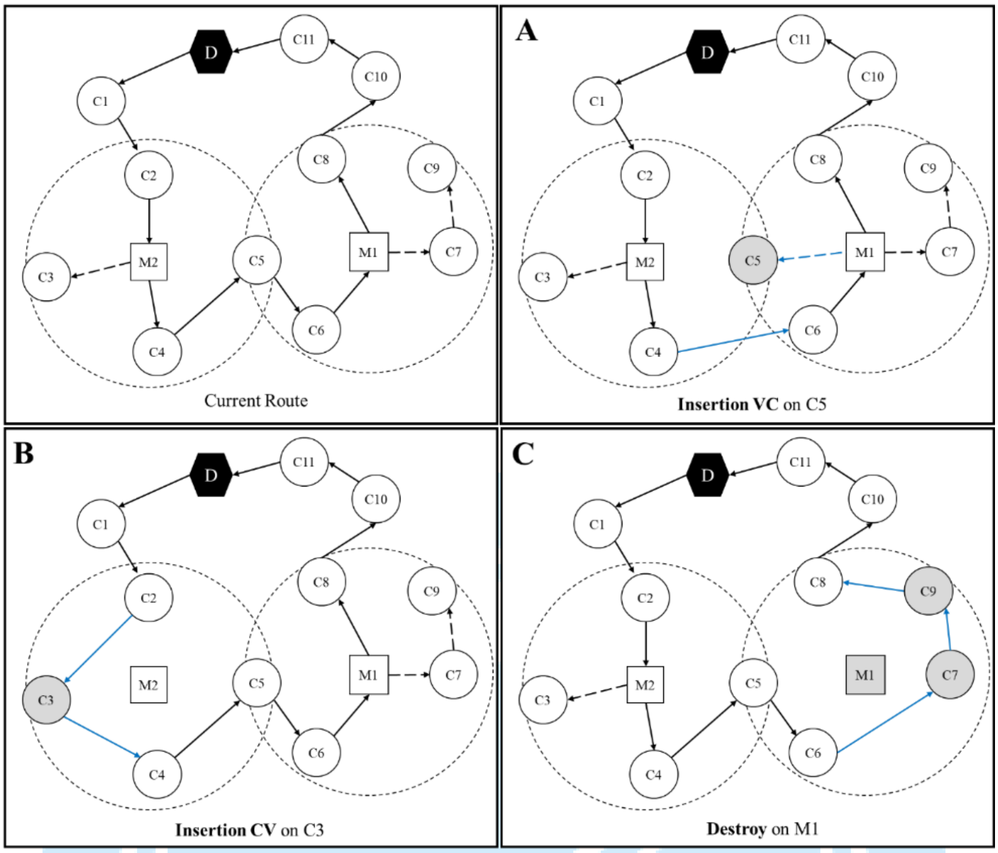


Figure 11. Illustration of the vehicle-crowdsource transfer search operators

4.1.3.2.**Tabu status and aspiration criterion**

In general, TS algorithm prevents the cycle of the similar local search by initiating the tabu list. Tabu list records the movements in the previous iterations to be banned for the next several determined iterations. In tabu list, each movement recorded is defined by the node $i, (i \in N^B)$ and the routes before and after the movement, for which both can be a vehicle route $\mathcal{X}_v \in \bar{\mathcal{X}}$ or a crowdsource route $\mathcal{Y}_b \in \bar{\mathcal{Y}}$. As a prevention mechanism, any similar move for θ iterations ahead will be excluded. An exception called aspiration criterion is defined to allow the cycle movement if it improves the current solution. Based on the initial experiments, parameter θ is set to 8.

4.1.3.3.**The tabu-search procedure**

In general, the improvement heuristics is controlled by the parameter μ as the level of intensification. The overall procedures are presented as follows.

Step 0: **Initialization**.

Set the iteration counters σ_1 (main iteration counter), σ_2 (consecutive non-improvement iterations counter), and σ_3 (infeasible iterations counter) to 1. Set the output of the construction heuristic algorithm as the initial solution and the best known solution $(\bar{\mathcal{X}},\bar{\mathcal{Y}})^* \coloneqq (\bar{\mathcal{X}},\bar{\mathcal{Y}})'$ and the current solution at iteration σ_1, $(\bar{\mathcal{X}},\bar{\mathcal{Y}})^{\sigma_1} := (\bar{\mathcal{X}},\bar{\mathcal{Y}})^*$. Set the intensification level, $\mu := 2$ and penalty value in (61), $\beta := 1$.

Step 1: **Neighborhood Search**.

Randomly generate three ordered nodes as the node sets for μ times based on $(\bar{\mathcal{X}},\bar{\mathcal{Y}})^{\sigma_1}$. Perform the search operators described in the neighborhood definition, started from the first node sets to search a new neighborhood solution. Use the new solution as the starting point to search another new neighbor solution. This process are repeated for μ times until all node sets have been selected. Evaluate the neighbor solution based on (61) and select the best and non-tabu solution as the best solution in this current iteration. Go to Step 2.

Step 2: **Solution, Tabu, and Parameter Update.**

Set $(\bar{\mathcal{X}},\bar{\mathcal{Y}})^* := (\bar{\mathcal{X}},\bar{\mathcal{Y}})^{\sigma_1+1}$, re-set the second counter $\sigma_2 \coloneqq 1$, and re-set $\mu \coloneqq 2$ if $(\bar{\mathcal{X}},\bar{\mathcal{Y}})^{\sigma_1+1}$ is feasible and $\mathcal{F}((\bar{\mathcal{X}},\bar{\mathcal{Y}})^{\sigma_1+1}) < \mathcal{F}((\bar{\mathcal{X}},\bar{\mathcal{Y}})^*)$; otherwise, $\sigma_2 \coloneqq \sigma_2 + 1$. Set $\sigma_3 \coloneqq \sigma_3$ if the solution is infeasible; otherwise set $\sigma_3 := 1$. Set β based on σ_3 to adjust the penalty level. Set $\sigma_1 \coloneqq \sigma_1 + 1$ for the main interation count. As for updating tabu list, increment the length of stay in tabu list for each stored record by 1 and remove any record with the length of stay more than θ. Set $\mu \coloneqq \mu + 1$ if $\sigma_2 > \eta\, min(|N| + |M|,15)$. Go to Step 3.

Step 3: **Intensification or termination**.

Go to Step 1 if $\mu \leq 5$; otherwise, stop the procedure and best known solution $(\bar{\mathcal{X}},\bar{\mathcal{Y}})^*$ is the final solution of the terminated procedure.

In the construction algorithm, the solution candidates in each iteration are evaluated after μ sequential movements. Local seach of neighbor solution can be represented by small value of μ, while larger value of μ (due to non-improvement solution) tries to jump to a less-constrained solution to avoid local optima trap. A new improved solution will reset the value of μ to its initial value of 2 to start new local search for the new improved solution. The value of μ starts from 2 and raised to 5 in Step 3 to balance between computation time and solution quality.

The overall improvement heuristics procedure is also controlled by the maximum number of non-improvement solution in each iteration before μ gets evaluated, η. A larger number of η might improve the solution quality in return of the computation time increase. Based on the initial experiments, the value of η is tuned to 20 balancing the solution quality and computation time trade off.

The value of β to represent the penalty is updated dynamically referred to the infeasible iterations counter, σ_3. The frequency of β update is controlled by parameter ξ. The penalty value is set to double, $\beta \coloneqq 2\beta$ after ξ consecutive no feasible iterations. In contrast, the penalty value is halved, $\beta \coloneqq \left(\frac{1}{2}\right)\beta$ after ξ consecutive feasible iterations. The value of ξ is set to be 6 by initial experiments.

4.2. Heuristic algorithm for stochastic problem

The heuristics algorithm for the stochastic problem is an extension of the heuristics algorithm in the deterministic problem. Additional features to address the uncertainty, such as new customer location rating to generate construction heuristics, new approximate cost evaluation of neighbor solution, etc. are provided in this sub-subsection.

4.2.1. Solution representation and evaluation

As a solution representation, this study defines three different vectors to represent vehicle routes, crowdsource assignments, and skipped customer visit in the detour route. The vehicle routes vector defines the sequence of each vehicle fleet visits, whereas the crowdsource assignments vector denote the assignment of each crowd-worker along with its transfer point location. The skipped customer visit represents the intentionally skipped customer detour in each of realization.

In the heuristics algorithm development, our study defines the solution representation consisting of a solution for vehicle routes, $\bar{\mathcal{X}}$ and a solution for crowdsources route, $\bar{\mathcal{Y}}$. For each vehicle v, a visiting sequence of nodes is determined, starting from depot, s_0^v as the first node, ending at the last visiting node $s_{F_v}^v$ (e.g. customer or a transfer point) and the final node as a depot, $(s_{F_v+1}^v)$. F_v is defined as the number of nodes in route v excluding the depot. $V_{\bar{\mathcal{X}}}$ is also defined as selected vehicles in vehicle route set, $\bar{\mathcal{X}}$, $V_{\bar{\mathcal{X}}} \subseteq V$. The crowdsource assignments are represented by a vector $\bar{\mathcal{Y}}$. The list of crowdsource assignments at transfer point l is denoted by $\mathcal{Y}_l$, $l \in M_{\bar{\mathcal{Y}}}$ contains all customers which are selected to be outsourced and it must be covered in transfer point l, with the last one being denoted by $s^l_{F_l}$, where F_l is the number of crowdsource assignments at transfer point l and $M_{\bar{\mathcal{Y}}}$ is the set of selected transfer points used in the solution $\bar{\mathcal{Y}}$, $M_{\bar{\mathcal{Y}}} \subseteq M$. A node γ_l is defined to denote the next node destination after visiting transfer point $l, \gamma_l \in V_{\bar{\mathcal{X}}} \subseteq V$.

A vector $\bar{\mathcal{Z}}$ is determined to represent the customers who are intentionally omitted during the detour trip in each of the realization $\omega, \omega \in \Omega$. These customers are skipped because of higher detour cost compared to the penalty cost or time unfeasibility in a certain realization. In each realization, ω, one or more omitted customers are denoted in $\mathcal{Z}_\omega$ consisting of the first node s^ω_0, until the last node $s^\omega_{F_\omega}$, where F_ω is the number of omitted customers which are skipped in the realization ω, $s^\omega_i \in \mathcal{Y}_l \backslash M$, for each $l \in M$, and $\omega \in \Omega$. A new vector of $\hat{\mathcal{Y}}$ is defined as the original vector $\bar{\mathcal{Y}}$ in which its elements $\mathcal{Y}_l$ are not in each $\mathcal{Z}_\omega$, for all $\omega \in \Omega$.

$$\bar{\mathcal{X}} = \left(\mathcal{X}_1, \mathcal{X}_2, \dots, \mathcal{X}_v, \dots, \mathcal{X}_{|V_{\bar{\mathcal{X}}}|}\right);\ \mathcal{X}_v = \left(s^v_0, s^v_1, \dots, s^v_{F_v}, s^v_{F_v+1} = s^v_0\right)$$

$$\bar{\mathcal{Y}} = \left(\mathcal{Y}_1, \mathcal{Y}_{2,} \dots, \mathcal{Y}_l, \dots, \mathcal{Y}_{|M_{\bar{\mathcal{Y}}}|}\right);\ \mathcal{Y}_l = \left(s^l_0, s^l_1, \dots, s^l_{F_l}\right)$$

$$\bar{\mathcal{Z}} = \left(\mathcal{Z}_1, \mathcal{Z}_2, \dots, \mathcal{Z}_\omega, \dots, \mathcal{Z}_{|\Omega|}\right); \mathcal{Z}_\omega = \left(s^\omega_0, s^\omega_1, \dots, s^\omega_{F_\omega}\right)$$

$$\hat{\mathcal{Y}} = \left(\hat{\mathcal{Y}}_1, \hat{\mathcal{Y}}_{2,} \dots, \hat{\mathcal{Y}}_l, \dots, \hat{\mathcal{Y}}_{|M_{\bar{\mathcal{Y}}}|}\right);\ \hat{\mathcal{Y}}_l = \mathcal{Y}_l \backslash \mathcal{Z}_\omega \tag{63}$$

The objective function $\mathcal{F}(\bar{\mathcal{X}}, \hat{\mathcal{Y}}, \bar{\mathcal{Z}})$ to evaluate the solution in (63) is defined consisting of delivery costs, crowdsource costs, and penalty terms in (64). The expected detour cost is calculated based on the set of realizations, Ω. A parameter $R_{l\omega}$ is used to define the outcome of crowdsources transfers at transfer point l in realization ω. The crowdsources cost or payment will be paid if there is any crowdsources transfer success. The truck detour cost will be imposed to replace the failed transfer, unless some customers are deliberately omitted in vector $\bar{\mathcal{Z}}$. The truck detour cost consists of the travelling cost from the transfer point to each of the crowdsourced customer in $\hat{\mathcal{Y}}$ plus the trip from the end of detour trip to the next destination according to the original plan. A skipped distance is deducted from the entire cost due to skipped distance. Finally, there is a penalty cost for intentionally skipping customers.

$$\mathcal{F}(\bar{\mathcal{X}}, \hat{\mathcal{Y}}, \bar{\mathcal{Z}}) = \sum_{v \in V_{\bar{\mathcal{X}}}} \sum_{i=0}^{F_v} C^r T^f_{s_i^v s_{i+1}^v} + \beta\rho + \sum_{\omega \in \Omega} P(\omega) \left\{ \sum_{l \in M_{\bar{\mathcal{Y}}}} \left[(1 - R_{l\omega}) \left(C^a + \sum_{i=0}^{F_l} C^b T^f_{l, s_i^l} \right) + R_{l\omega} \left(\sum_{i=0}^{F_l - 1} C^r T^f_{s_i^l, s_{i+1}^l} + C^r T^f_{s_{F_l}^l, \gamma_l} - C^r T^f_{l, \gamma_l} \right) \right] + \sum_{i=0}^{F_\omega} \alpha \right\} \tag{64}$$

Where γ_l is the next node destination after visiting transfer point l according to the original route. $P(\omega)$ is the probability of transfer failure in the event ω. The paramter ρ consists of the positive deviation of vehicle maximum capacity minus total load of each vehicle, and positive deviation between the ending driver service time and total travel time of each vehicle. For the feasible solution, the penalty terms are set to zero. Penalty parameter β is defined as a non-negative parameter to balance the penalty value, starting from initial value of 1 and adjusted dynamically based on the progress of the solution throughout the iteration.

One of the disadvantages when considering uncertainty in the optimization model is the difficulty to handle enormous realizations in the objective function. In our case, the number of realizations to be evaluated spikes up when more transfer points are considered. As mentioned earlier, the relationship between the number of transfer point and the number of realizations is $2^{|m|}$. Therefore, our study proposed an efficient cost evaluation approximation to evaluate the objective function without any need to evaluate every single realization ω.

4.2.1.1. **Movement cost as an approximation of objective function**

Instead of evaluating every single solution in each realization $\omega, \omega \in \Omega$, this study proposes an approximate evaluation for every neighborhood search in each iterations. Each neighborhood search consists of moving one or more nodes from its original position to another position which is not prevented by tabu mechanism. The cost approximation calculates the impact of a node movement and sum up the impact as a cost to the true cost evaluation. The true cost evaluation is the objective evaluation based on the equation (64). The term movement cost is used to represent the calculated cost as the impact of a node movement. Several movement costs will be calculated as more than one nodes are moved in one iteration. Overall, the approximate cost evaluation of multiple node movement in a neighborhood search can be illustrated in Figure 12.

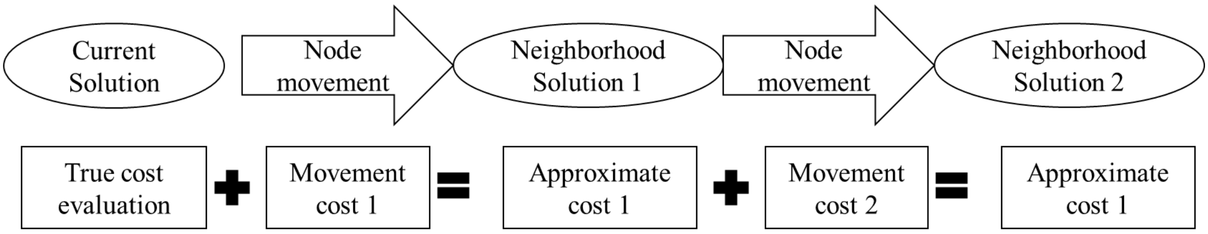


Figure 12. Approximate cost evaluation scheme

This study also performs a complexity analysis to compare the computations efficiency between true cost evaluation and approximate cost evaluation. The complexity of true cost evaluation in (64) is exponentialy increase depend on the number of available transfer points $2^{|M|}$ or can be represented as $O(2^N)$. In the other hand, the approximate cost evaluation is not dependent on the number of available transfer points, instead, it depends on the number of node movements or can be represented as $O(N)$.

The calculation of movement cost consists of two type calculations. The first calculation type is related to the first-stage objective function which does not involve any probability. In the second calculation type, any movement related to second-stage objective function will be calculated as well as the probability of the movement. As an example, the current solution after some iterations, the movement of the node, and the neighborhood solution are defined as follows.

Current solution:	$\mathcal{X}_1 = \{D, C_1, C_2, M_1, C_3, C_4, D\}, \mathcal{Y}_{l_1} = \{C_5, C_6\}$
Movement:	Move C_3 from vehicle route $\mathcal{X}_1$ to crowdsource assignment $\mathcal{Y}_{M_1}$. C_3 is in a radius of transfer point M_1.
Neighborhood solution:	$\mathcal{X}_1 = \{D, C_1, C_2, M_1, C_4, D\}, \mathcal{Y}_{M_1} = \{C_5, C_6, C_3\}$

The neighborhood solution cost is calculated by summing up the current solution cost (true cost) based on (64) with the neighborhood solution's movement cost. Evaluation of the movement cost is provided in Table 6.

Table 6. The calculation of the movement cost

Movement	1st stage objective function evaluation	2nd stage objective function evaluation
Removing $\boldsymbol{C_3}$ from $\boldsymbol{\mathcal{X}_1}$	$D_{C_3,C_4} + D_{M_1,C_3} - D_{M_1,C_4}$ Reason: Vehicle route changes from: $D - C_1 - C_2 - M_1 - \boldsymbol{C_3} - C_4 - D$ to: $D - C_1 - C_2 - M_1 - C_4 - D$	$\left(D_{C_6,\boldsymbol{C_3}} - D_{C_6,\boldsymbol{C_4}}\right)P_{M_1}\, C^r$ Reason: Changes in Vehicle detour sequence from: $\boldsymbol{M_1} - C_5 - C_6 - \boldsymbol{C_3}$ to: $\boldsymbol{M_1} - C_5 - C_6 - \boldsymbol{C_4}$
Inserting $\boldsymbol{C_3}$ in $\boldsymbol{\mathcal{Y}_{l_1}}$	None	$-\left(D_{\boldsymbol{M_1},\boldsymbol{C_3}} C^b + C^a\right)\left(1 - P_{M_1}\right)$ Reason: Crowdsource cost for success crowdsource transfer $\left(D_{\boldsymbol{C_3},C_4} + D_{C_6,\boldsymbol{C_3}} - D_{C_6,C_4}\right)P_{M_1}\, C^r$ Reason: Changes in vehicle detour sequence From: $\boldsymbol{M_1} - C_5 - C_6 - C_4$ to: $\boldsymbol{M_1} - C_5 - C_6 - \boldsymbol{C_3} - C_4$

The movement described in Table 6 is categorized as the movement of customer node from vehicle route to crowdsource assignment route which also described in (68). Several movements cost calculations are also possible due to the combination of the search operators. All movement cost calculations are described as follows.

a. Movement cost of customer node from vehicle route to another vehicle route

The simplest movement is the movement from customer node in the vehicle route to another vehicle route. It is defined as the removing the selected node from vehicle route of the current solution and inserting the selected node into the new vehicle route of the neighborhood solution. This type movement cost does not affect the second stage objective function because no crowdsource assignment movement involved. The cost movement calculation is defined as $\mathcal{F}_1'(i, \mathcal{X})$ in (65).

$$\mathcal{F}_1'(i, \mathcal{X}) = \left[\left(D_{i,i^+} + D_{i^-,i} - D_{i^-,i^+}\right)^{\mathcal{X}} - \left(D_{i,i^+} + D_{i^-,i} - D_{i^-,i^+}\right)^{\tilde{\mathcal{X}}}\right] C^r \quad (65)$$

Where

i Selected customer node

i^-, i^+ Preceding node and following node of i, respectively

$(\,)^{\mathcal{X}}$ Current solution

$(\,)^{\tilde{\mathcal{X}}}$ Neighborhood solution

Additional cost related to the second stage objective function need to be incurred if the preceding node of i in the current solution and/or the preceding node of i in the neighborhood solution are transfer point. The following node after transfer point is the detour trip comeback node. Therefore, the changes of the following node after transfer point will change the detour trip cost. In addition, the cost of distance skipped will be changed due to this movement. The additional cost is defined as $\mathcal{F}_2'(i|i^- \in M)$, in (66).

$$\mathcal{F}_2'(i|i^- \in M) = \left[\left(D_{\tau(i^-),i} - D_{\tau(i^-).i^+}\right) - \left(D_{i^-,i} - D_{i^-,i^+}\right)\right] P_{i^-} C^r \quad (66)$$

Where $\tau(i^-)$ is the last visiting sequence of detour trip defined in the crowdsource assignment related to the transfer point $i^-, i^- \in M$.

b. Movement cost of transfer point from vehicle route to another vehicle route

The transfer point in the vehicle route may also be moved during the construction of neighborhood solution. The movement of transfer point will affect all the crowdsource assignments related. In the first stage objective function, the removal of transfer point from the current solution and the insertion of transfer point to the neighbor solution are calculated. In the second stage objective function, the comeback detour trip and skipped distance will be re-calculated because the changes of preceding and following node prior to the movement. The movement cost is defined as $\mathcal{F}_3'(m)$ in (67).

$$\mathcal{F}'_3(m) = \left[\left(D_{m,m^+} + D_{m^-,m} - D_{m^-,m^+}\right)^{\mathcal{X}} - \left(D_{m,m^+} + D_{m^-,m} - D_{m^-,m^+}\right)^{\tilde{\mathcal{X}}}\right] C^r + \left[\left(D_{\mu(m),m^+} - D_{m,m^+}\right)^{\mathcal{X}} - \left(D_{\mu(m),m^+} - D_{m,m^+}\right)^{\tilde{\mathcal{X}}}\right] C^r P_m \quad (67)$$

Additional cost defined in (66) related to the second stage objective function need to be incurred if the preceding node of m in the current solution and/or the preceding node of m in the neighborhood solution are transfer point.

c. Movement cost of customer node from vehicle route to crowdsource assignment

The customer orders have the option of crowdsource delivery if the location is close to the transfer point. This type of movements involves the movement of customer node from the vehicle route in the current solution to the crowdsource assignment in the neighborhood solution. This movement will affect the first and second stage objective function. In the first stage objective function, the removal of the customer node from the vehicle route in the current solution will be evaluated. Then, the insertion cost in terms of crowdsource cost (for success outcome) and detour trip cost (for failure outcome) will be incurred as the second stage objective function. The cost function is defined as $\mathcal{F}'_{4a}(i, m_i)$, in (68) assuming that transfer point m_i related to selected customer node i is available in crowdsource assignment.

$$\mathcal{F}'_{4a}(i, m_i) = \left(D_{i,i^+} + D_{i^-,i} - D_{i^-,i^+}\right)^{\mathcal{X}} C^r - \left(D_{m_i,i} C^b + C^a\right)\left(1 - P_{m_i}\right) - \left(D_{i,i^+} + D_{i^-,i} - D_{i^-,i^+}\right)^{\mathcal{Y}_{m_i}} P_{m_i} C^r \quad (68)$$

Additional cost is needed when the position of new customer node in the crowdsource assignment (in neighborhood solution) is the last sequence. The comeback detour trip will be altered due to the detour trip sequence change. The new cost function is defined as $\mathcal{F}'_{4b}(i, m_i)$, in (69).

$$\mathcal{F}'_{4b}(i, m^i) = \left(D_{i,i^+} + D_{i^-,i} - D_{i^-,i^+}\right)^{\mathcal{X}} C^r - \left(D_{m_i,i} C^b + C^a\right)\left(1 - P_{m_i}\right) - \left[\left(D_{i,i^+} + D_{i^-,i} - D_{i^-,i^+}\right)^{\mathcal{Y}_{m_i}} - \left(D_{i,m_i^+} + D_{i^-,m_i^+}\right)^{\tilde{\mathcal{X}}}\right] P_{m_i} C^r \quad (69)$$

Where

m_i Transfer point related to customer node i

${m_i}^+$ Following node after transfer point m_i in the vehicle route of neighrbohood solution

$(\,)^{\tilde{y}_{m_i}}$ Crowdsource assignment related to transfer point m_i in neighborhood solution.

Different cost function will be defined if the related transfer point m_i is not available through all of vehicle route in the current solution. Transfer point will need to be inserted to the vehicle route in the neighborhood solution. As the consequence, the transfer point insertion cost need to be added in the first stage objective function. In the second stage objective function, the crowdsource cost, detour trip cost, and skipped distance will also be calculated. The movement cost function is defined as $\mathcal{F}'_{4c}(i, m_i)$, in (70).

$$\begin{aligned}\mathcal{F}'_{4c}(i, m_i) = {} & \left(D_{i,i^+} + D_{i^-,i} - D_{i^-,i^+}\right)^{x} C^r - \left(D_{m_i,{m_i}^+} + D_{{m_i}^-,m_i} - D_{{m_i}^-,{m_i}^+}\right)^{\tilde{x}} C^r \\ & - \left(D_{m_i,i} C^b + C^a\right)\left(1 - P_{m_i}\right) \\ & - \left[\left(D_{m_i,i} + D_{i,m_i^+}\right)^{\tilde{y}_{m_i}} + \left(D_{m_i,m_i^+}\right)^{\tilde{x}}\right] P_{m_i}\, C^r\end{aligned} \tag{70}$$

The movement cost defined in (66) need to be incurred if the preceding node of i in the current solution and/or the preceding node of m_i in the neighborhood solution are transfer point.

d. Movement cost of customer node from crowdsource assignment to vehicle route

In contrast with part c, the customer order can also be transferred from crowdsource assignment in the current solution to the vehicle route in the neighborhood solution. The cost evaluation structure is similar to $\mathcal{F}'_{4a}(i, m_i)$, however it is reversed in the opposite direction to form function $\mathcal{F}'_{5a}(i, m_i)$, in (71).

$$\begin{aligned}\mathcal{F}'_{5a}(i, m_i) = {} & \left(D_{i,i^+} + D_{i^-,i} - D_{i^-,i^+}\right)^{y_{m_i}} C^r P_{m_i} + \left(D_{m_i,i} C^b + C^a\right)\left(1 - P_{m_i}\right) \\ & - \left(D_{\hat{\imath},\hat{\imath}^+} + D_{\hat{\imath}^-,\hat{\imath}} - D_{\hat{\imath}^-,\hat{\imath}^+}\right)^{\tilde{x}} C^r\end{aligned} \tag{71}$$

Following the same formulation in $\mathcal{F}'_{4b}(i, m_i)$ with different order, the cost function $\mathcal{F}'_{5b}(i, m_i)$ in (72) is formulated if the position of customer node i in the crowdsource assignment of current solution is the last position.

$$\mathcal{F}'_{5b}(i, m_i) = \left[\left(D_{i,i^+} + D_{i^-,i} - D_{i^-,i^+}\right)^{y_{m_i}} + \left(D_{i,m_i^+} + D_{i^-,m_i^+}\right)^{x}\right] C^r P_{m_i} + \left(D_{m_i,i} C^b + C^a\right)\left(1 - P_{m_i}\right) - \left(D_{\hat{\imath},\hat{\imath}^+} + D_{\hat{\imath}^-,\hat{\imath}} - D_{\hat{\imath}^-,\hat{\imath}^+}\right)^{\tilde{x}} C^r \tag{72}$$

A more comprehensive cost evaluation is developed when the selected node i, is the only node in the crowdsource assignment related to transfer point m^i of the current solution. The transfer point m^i will be removed from the vehicle route and the crowdsource assignment related to m^i will be destroyed due to no more assignment. New cost function is defined as $\mathcal{F}'_{5c}(i, m_i)$ in (74).

$$\begin{aligned} \mathcal{F}'_{5c}(i, m^i) = {} & \left[\left(D_{m_i,i} + D_{i,m_i^+}\right)^{y_{m_i}} - \left(D_{m_i,m_i^+}\right)^{x}\right] P_{m_i} C^r \\ & + \left(D_{m_i,i} C^b + C^a\right)\left(1 - P_{m_i}\right) \\ & + \left(D_{m_i,m_i^+} + D_{m_i^-,m_i} - D_{m_i^-,m_i^+}\right)^{x} C^r \\ & - \left(D_{i,i^+} + D_{i^-,i} - D_{i^-,i^+}\right)^{\tilde{x}} C^r \end{aligned} \tag{73}$$

The movement cost defined in (66) need to be incurred if the preceding node of i in the neigborhod solution and/or the preceding node of m_i in the current solution are transfer point.

4.2.2. Construction algorithm

In the construction algorithm, the customers which are possible to be outsourced to crowdsources will be selected based on the location rating. When a customer order is outsourced, the delivery fleet reduces its travel cost. However, there will be the expectation of crowdsources cost and the truck detour cost (or penalty cost). Therefore, the rank of customer orders to be outsourced is defined by averaging the distance of n nearest neighbors and distance to depot, as well as the expectation of crowdsource transfer outcome in (74).

$$W_i = \left\{\left(\frac{\sum_{j=1}^{n} T_{i,f_i(j)}^{f}}{n}\right) + T_{0i}^{v}\right\} C^r - P_{\pi(i)}\left(T_{i,\pi(i)} + \frac{\sum_{j=1}^{n} T_{i,f_i(j)}^{f}}{n}\right) C^r - \left(1 - P_{\pi(i)}\right)\left(C^a + C^b T_{i,\pi(i)}^{f}\right) \tag{74}$$

where $f_i(j)$ is the j^{th} element of the ascending-ordered set based on distance for node i's neighbors, $i \in N^B$. $\pi(i)$ is the closest and relevant transfer location to customer $i, i \in N$. Based on the initial tuning, parameter n is set to be 3 The construction heuristics procedures are presented as follows.

Step 1: **Node Evaluation for Classification.** Based on the customer location rating in (74), select the first λ customers to be outsourced as the set of R^c. Set the value of λ to 1. The set of R^m is defined as the set of transfer points related to the selected outsourced customers in R^c. The nearest transfer point is selected if there are multiple transfer points available for one customer.

Step 2: **Main Truck Route Generation.** Based on the set of R^c, the customer order which are delivered by delivery fleets as vehicle routes customer are defined, $(N \backslash R^c) \cup R^m$. Execute the combination of the sweep procedure (Huang et al., 2018) and the nearest neighbor procedure (Hurkens et al., 2004) in Appendix 1 to construct the main vehicle routes with the respect of truckload capacity and hours of services constraints. The transfer point load is accumulated by the total customer demand which are assigned to relay in the transfer point,

Step 3: **Crowdsource Assignment Generation.** Define the set $R_l^c, R_l^c \subseteq R^c$ as the set of crowdsourced customers in transfer point l. Generate the crowdsource assignment based on the combination of modified saving method from Ghiani et al. (2004) in Appendix 2.

Step 4: **Skipped detour.** Perform *omitted customer procedure* to determine the omitted customers (See Appendix 3).

Step 5: **Intensification.** Obtain and save the objective function in (64). Re-do the Steps 1-3 and add the value of λ, $\lambda \coloneqq \lambda + 1$ if $\lambda < \Lambda$; Otherwise go to Step 5.

Step 6: **Termination.** The initial solution $(\bar{\mathcal{X}}, \bar{\mathcal{Y}})'$ is the initial solution as the best solution among Λ different solutions based on Step 1-4.

The algorithm emphasizes the appropriate balance between delivery truck utilization and the crowdsources task. The parameter λ denoting the number of outsourced customers is increased from 1 to Λ to obtain the balance. The maximum value of Λ is determined as $\min(|N|, 2|M|)$ by the initial experiment to balance the solution quality and computation time. The initial solution will be improved by the tabu search algorithm as the improvement algorithm in the next section.

4.2.3. Improvement algorithm

4.2.3.1. **Neighborhood definition, tabu status, and aspiration criterion**

The neighborhood definition, tabu status, and aspiration criterion in of the stochastic problem are defined as the same as the definition in the deterministic problem due to the similar problem characteristics of two-echelon routing system.

4.2.3.2. **The tabu-search procedure**

In general, the improvement heuristics is controlled by the parameter μ as the level of intensification. The overall procedures are presented as follows.

Step 0: **Initialization**.

Set the iteration counters σ_1 (main iteration counter), σ_2 (consecutive non-improvement iterations counter), and σ_3 (infeasible iterations counter) to 1. Set the output of the construction heuristic algorithm as the initial solution and the best known solution $(\bar{\mathcal{X}}, \bar{\mathcal{Y}})^* \coloneqq (\bar{\mathcal{X}}, \bar{\mathcal{Y}})'$ and the current solution at iteration σ_1, $(\bar{\mathcal{X}}, \bar{\mathcal{Y}})^{\sigma_1} := (\bar{\mathcal{X}}, \bar{\mathcal{Y}})^*$. Set the intensification level, $\mu := 2$ and penalty value in (64), $\beta := 1$.

Step 1: **Current Solution Evaluation.**

Let $(\bar{\mathcal{X}}, \bar{\mathcal{Y}}, \bar{\mathcal{Z}})^* \coloneqq (\bar{\mathcal{X}}, \bar{\mathcal{Y}}, \bar{\mathcal{Z}})'$ be the best known solution, and set $(\bar{\mathcal{X}}, \bar{\mathcal{Y}}, \bar{\mathcal{Z}})^{\sigma_1} := (\bar{\mathcal{X}}, \bar{\mathcal{Y}}, \bar{\mathcal{Z}})^*$ as the current solution at iteration σ_1. Evaluate the current solution using

$\mathcal{F}(\bar{\mathcal{X}}, \hat{\mathcal{Y}}, \bar{\mathcal{Z}})$ to obtain the true value of objective function, $\mathcal{F}^*(\bar{\mathcal{X}}, \bar{\mathcal{Y}}, \bar{\mathcal{Z}})$. Go to step 2.

Step 2: **Neighborhood Search**.

Randomly generate three ordered nodes as the node sets for μ times based on $(\bar{\mathcal{X}}, \bar{\mathcal{Y}})^{\sigma_1}$. Perform the search described in the neighborhood definition, started from the first node sets. Use the new neighborhood solution as the starting point to search another new neighbor solution. This process are repeated for μ times until all node sets have been selected. Evaluate the neighbor solution using the movement cost defined in (65) - (73) and obtain the approximated objective function by subtracting the previous approximated objective function with the movement cost. Exclude the neighborhood solutions which are listed in tabu list. Select the best neighborhood solution by the best approximate objective function. Go to Step 3.

Step 3: **Skipped detour.**

Perform *omitted customer procedure* to determine the omitted customers, $\bar{\mathcal{Z}}$ (See Appendix 3). Go to Step 4.

Step 4: **Solution, Tabu, and Parameter Update.**

Set $(\bar{\mathcal{X}}, \bar{\mathcal{Y}}, \bar{\mathcal{Z}})^* := (\bar{\mathcal{X}}, \bar{\mathcal{Y}}, \bar{\mathcal{Z}})^{\sigma_1}$, re-set the second counter $\sigma_2 \coloneqq 1$, and re-set $\mu \coloneqq 2$ if $(\bar{\mathcal{X}}, \bar{\mathcal{Y}}, \bar{\mathcal{Z}})^{\sigma_1}$ is feasible and $\bar{\mathcal{F}}^*(\bar{\mathcal{X}}, \bar{\mathcal{Y}}, \bar{\mathcal{Z}}) < \mathcal{F}^*(\bar{\mathcal{X}}, \bar{\mathcal{Y}}, \bar{\mathcal{Z}})$; otherwise, $\sigma_2 \coloneqq \sigma_2 + 1$. Set $\sigma_3 \coloneqq \sigma_3$ if the solution is infeasible; otherwise set $\sigma_3 := 1$. Set β based on σ_3 to adjust the penalty level. Set $\sigma_1 \coloneqq \sigma_1 + 1$ for the main interation count. As for updating tabu list, increment the length of stay in tabu list for each stored record by 1 and remove any record with the length of stay more than θ. Set $\mu \coloneqq \mu + 1$ if $\sigma_2 > \eta\, min(|N| + |M|, 15)$. Go to Step 5.

Step 5: **Intensification or termination**. Go to Step 1 if $\mu \leq 5$; otherwise, stop the procedure and best known solution $(\bar{\mathcal{X}}, \bar{\mathcal{Y}})^*$ is the final solution of the terminated procedure.

In the construction algorithm, the solution candidates in each iteration are evaluated after μ sequential movements. Local search of neighbor solution can be represented by small value of μ, while larger value of μ (due to non-improvement solution) tries to jump to a less-constrained solution to avoid local optima trap. A new improved neighborhood solution will reset the value of μ to its initial value of 2 to start new local search for the new improved neighborhood solution. The value of μ starts from 2 and incremented by 1 to 5 in Step 3 to balance between computation time and solution quality.

The overall improvement heuristics procedure is also controlled by the maximum number of non-improvement solution, η in each iteration. A larger number of η might improve the solution quality in return of the computation time increase. Based on the initial experiments, the value of η is tuned to 20 to balance the solution quality and computation time trade off.

The value of β to represent the penalty is updated dynamically based on the infeasible iterations counter, σ_3. The frequency of β update is controlled by parameter ξ. The penalty value is set to double, $\beta \coloneqq 2\beta$ after ξ consecutive no feasible iterations. In contrast, the penalty value is halved, $\beta \coloneqq (1/2)\beta$ after ξ consecutive feasible iterations. The value of ξ is set to be 6 by initial experiments.

CHAPTER 5 NUMERICAL EXPERIMENT

5.1. Test problem design

In this study, the problem tests are categorized into two groups. The first group is labeled "H" for hypothetical instances consisting of 15 customer nodes and 5 transfer points. These problem tests are generated by a two-phase process. Five problem instances labeled by A-E are generated in the first phase with random depot and customer locations. Uniform distribution is utilized to generate the random coordinates of customer and depot nodes in *–x* and *–y*. A cluster analysis of customer locations was performed to assign the location of transfer point in the center of each generated cluster. In the second phase, for each problem sets in phase one, the new sub-problem sets are generated by perturbing the customer locations and fixing the location of depot and transfer points. The perturbation of customer location follow a normal random distribution, $N(\mu, \sigma^2)$ with mean, μ is the coordinate in *-x* or *-y* of the node and the standard deviation, $\sigma = 2$. Based on the two-phase probem set generation process, five configurations or distribution systems with five sub test instances for each of the distribution system can be generated to represent the day to day customer order patterns.

In the second category, two classical VRP problem benchmarks from Augerat et al. (1995) and Christofides et al. (1979) are used and modified to match this study problem definition. The problems from Augerat et al. (1995) are labeled by "P," and Christofides et al. (1979) are labeled by "CMT1", "CMT2", and "CMT3". This study adds the transfer points to the original problem instances randomly as the modification to match the problem definition. In general, the problem test labels consist of three indexes to represent the number of customers (n), the number of transfer points (m), as well as the series of the instances (e.g. A1, A2, B1, etc.). The generated benchmark problem can be downloaded at Mendeley Data (https://data.mendeley.com/datasets/rxm98px352/2).

The parameter sets in this study are presented in Table 7. Mainly, the cost parameters are adopted from Kafle et al. (2017) and Huang and Ardiansyah (2019). The fixed crowdsources cost is based on the Uber base fare in the US (Dough, 2018).

Table 7. Parameter sets

Parameter	Value
Vehicle capacity, $\boldsymbol{Q^v}$	25 unit capacities
Crowdsouces carrying capacity, $\boldsymbol{Q^c}$	3 unit capacities
Vehicle variable cost, $\boldsymbol{C^r}$	US$ 68.9/hour
Crowdsources fixed cost, $\boldsymbol{C^a}$	US$ 5/crowdsource
Crowdsource variable cost, $\boldsymbol{C^b}$	US$ 10/hour
Maximum hours of service, $\boldsymbol{L}$	8 hours
Vehicle speed	20 unit distance /hour
Crowdsources speed	10 unit distance /hour

5.2. Experiment of deterministic problem

5.2.1. Crowdsource delivery contribution

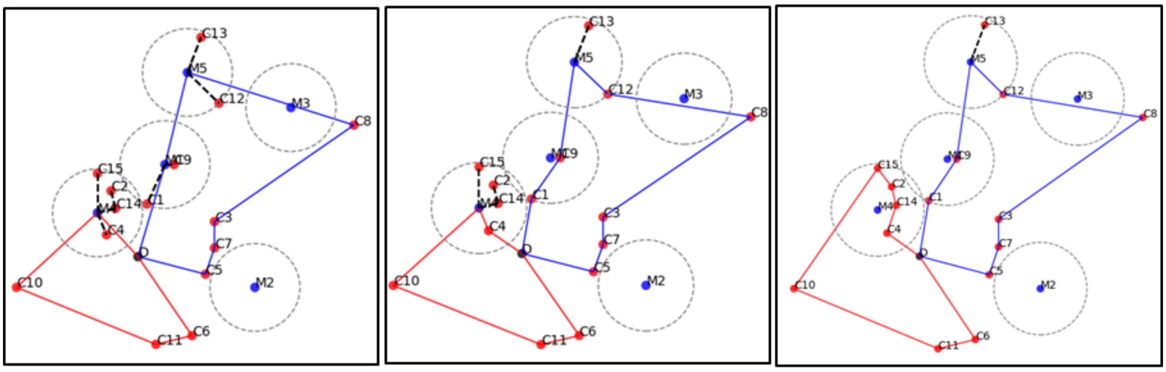
In this part, the test problem "H" is utilized and the optimal solution of the test problems are generated by the GUROBI solver for the small-size problem instances. The results will be used as an elaboration upon the decision problem nature and characteristics. The crowdsource delivery integration represented by the model in (1) - (26) will be compared with the all outsourcing strategy and no outsourcing strategy to illustrate the cost saving from the crowdsource delivery integration. The all outsourcing and no outsourcing solutions are generated by fixing the associated crowdsourcing decisions variables in (1) - (26). The comparison of each problem set (labeled by A - E) solutions (i.e. objective function and number of crowdsourced customers) are presented in Table 7.

Based on the results in Table 7, the *crowdsource delivery* as the representation of the original model provides 4.98% cost reductions when compared with the *no crowdsourcing* strategy and generate 5.19% cost improvements when compared with the case of *all crowdsourcing* strategy, in which the decision outsource all possible customers (reachable from transfer points). As presented in the number of customers outsourced, crowdsourcing does not always lead to cost reduction. A carefull decision to balance the truck operational cost and the crowdsourcing cost need to be examined. In addition, the non-crowdsouring strategy can provide better results in several problem tests compared to full crowdsource strategy, emphasizing the crowdsource delivery collaboration need to be planned carefully in order to bring the best cost reductions.

Table 8. Cost levels for different crowdsourcing strategies

No	Problem	Crowdsource Delivery		All Crowdsourcing		No Crowdsourcing	
		Objective Value	The number of customers crowdsourced	Objective Value	%GAP	Objective Value	%GAP
1	H-n15m5-A1	821.8	1	873.0	6.23%	827.2	0.66%
2	H-n15m5-A2	938.4	3	995.4	6.08%	966.8	3.03%
3	H-n15m5-A3	1062.7	2	1119.9	5.38%	1065.6	0.27%
4	H-n15m5-A4	893.5	4	910.8	1.94%	997.5	11.64%
5	H-n15m5-A5	1344.7	0	1398.7	4.02%	1344.7	0.00%
6	H-n15m5-B1	788.7	4	852.8	8.12%	905.6	14.81%
7	H-n15m5-B2	676.2	4	729.6	7.90%	694.5	2.70%
8	H-n15m5-B3	816.4	1	884.0	8.28%	820.4	0.49%
9	H-n15m5-B4	928.1	4	959.1	3.34%	1011.1	8.94%
10	H-n15m5-B5	926.1	0	962.4	3.93%	926.1	0.00%
11	H-n15m5-C1	1205.1	4	1220.5	1.28%	1273.2	5.65%
12	H-n15m5-C2	925.7	3	962.9	4.01%	970.2	4.81%
13	H-n15m5-C3	902.9	4	952.1	5.44%	1017.9	12.73%
14	H-n15m5-C4	837.8	1	948.1	13.16%	864.7	3.20%
15	H-n15m5-C5	1191.9	2	1226.9	2.94%	1212.1	1.68%
16	H-n15m5-D1	1081.1	4	1131.7	4.68%	1201.7	11.16%
17	H-n15m5-D2	815.5	12	815.5	0.00%	915.8	12.30%
18	H-n15m5-D3	902.5	6	942.1	4.39%	1031.5	14.30%
19	H-n15m5-D4	940.1	5	963.1	2.44%	1011.1	7.55%
20	H-n15m5-D5	1072.8	0	1129.8	5.31%	1072.8	0.00%
21	H-n15m5-E1	939.6	0	996.8	6.09%	939.6	0.00%
22	H-n15m5-E2	1164.3	0	1228.9	5.55%	1164.3	0.00%
23	H-n15m5-E3	885.5	1	901.3	1.78%	926.1	4.57%
24	H-n15m5-E4	1087.4	2	1167.2	7.34%	1177.9	8.33%
25	H-n15m5-E5	889.9	1	933.9	4.95%	898.7	0.99%
Average Gap					4.98%		5.19%

The illustration of the delivery plan to highlight the model behavior and the decision problem nature is provided for one test instance (H-n15m5-B4) with different crowdsouring cost level in Figure 13. all of the feasible crowdsource service will be maximized for a low crowdsourcing cost (crowdsourcing cost is zero). In contrast, lower crowdsource delivery collaboration is generated for the double crowdsource service.



Zero crowdsource cost | Default crowdsource cost | Doubled crowdsource cost

Figure 13. Illustration of route decisions for different crowdsourcing cost

5.2.2. Performance of heuristic algorithm

In this part, the test problems in the second category are utilized (problem test P and CMT). The solutions are evaluated from two aspects (i.e. solution quality and computation time). The evaluation of solution quality is available only for the small problem tests (with 15 customers and 5 transfer points) because of the limitation on the mathematical solver computing resources. The results are presented in **Table 9**. The heuristics algorithm is able to generate a nearly optimal solution with fast computation time (less than 3 seconds) for small instances with the differences between optimal solutions from the mathematical solver and heuristics solution are less than 0.1% (available as gap in the last column of **Table 9**). The lower bounds are provided for the larger problem size of 20 to 30 customers by the GUROBI solver with four hours limited computation times. The negative percentage gap shows the heuristics algorithm can provide better feasible solution with faster computation time compared to the lower bound in some instances.

The evaluation of larger instances is performed based on the significant improvements made by the improvement algorithm (available in the second last column of **Table 9**) due to the unavailable optimal solution or lower bound by the mathematical solver. As observed, the substantial and stable improvements can be generated by the heuristics algorithm indicating independent and insensitive relationship between the proposed improvement heuristics algorithm and the initial solution. The computation time increases depend on the problem size. However, acceptable computation time can still be achieved even for the biggest problem instances (with 99 customers and 12 transfer points) with total computation is less than four minutes. The growth of the heuristics algorithm runtime depends on the problem size (e.g. the number of customer and the number of available transfer point). In average, it can be best approximated by $runtime = 0.013n^2 + 0.011m^2 + 0.352$. where n is the number of customers and m is the number of available transfer points.

Table 9. Heuristics algorithm results

Problem	MIP Solver	Construction Algorithm		Improvement Algorithm		Percentage Improvement	Percentage Gap
	Objective Bound	Obj. Function	Elapsed Time	Obj. Function	Elapsed Time		
H-n15m5-A1*	821.9	1183.7	0.1	821.9	1.8	17.68%	0.00%
H-n15m5-B1*	788.8	941.5	0.1	788.8	1.4	11.64%	0.00%
H-n15m5-C1*	925.8	1211.7	0.1	925.8	1.5	17.48%	0.00%
H-n15m5-D1*	1081.2	1146.2	0.2	1082.0	1.3	18.31%	0.07%
H-n15m5-E1*	939.6	1330.4	0.1	939.6	1.8	30.00%	0.00%
P-n15m5*	496.6	1053.2	0.1	496.6	3.0	20.93%	0.00%
P-n30m10	1166.4	1654.7	0.8	1197.5	18.9	37.81%	2.67%
P-n50m12		2510.7	1.8	1782.3	27.9	40.17%	
P-n75-m12		3453.1	4.1	2355.7	57.9	41.21%	
P-n100-m12		3651.9	8.7	2816.3	108.5	27.07%	
CMT1-n20m7	950.8	1151.9	0.3	950.8	6.6	21.15%	0.00%
CMT1-n30m10	1077.3	1593.2	0.7	1095.7	19.0	33.68%	1.71%
CMT1-n49m12		2045.7	2.2	1673.0	38.2	22.43%	
CMT2-n20m7	972.1	1541.4	0.3	972.0	6.2	58.41%	0.10%
CMT2-n30m10	1346.4	1905.7	0.7	1173.6	12.3	52.69%	-11.56%
CMT2-n50m12		2313.0	2.0	1570.2	32.5	38.44%	
CMT2-n74m12		4794.9	3.9	2436.5	70.4	83.60%	
CMT3-n20m7	1044.4	1250.2	0.3	1136.5	8.7	10.01%	-0.71%
CMT3-n30m10	1323.6	1801.4	0.7	1299.0	16.9	38.67%	-1.86%
CMT3-n50m12		3057.4	1.7	1831.9	47.4	52.74%	
CMT3-n75m12		3808.4	4.1	2444.7	88.4	40.45%	
CMT3-n99m12-A		4537.0	7.7	3026.0	180.5	46.31%	
CMT3-n99m12-B		3938.6	7.7	2821.2	134.7	39.61%	
CMT3-n99m12-C		4459.6	7.8	2930.2	184.7	52.19%	
CMT3-n99m12-D		4719.5	7.4	2968.0	134.4	59.01%	
CMT3-n99m12-E		4166.2	7.7	2831.2	133.6	47.15%	
CMT3-n99m12-F		3520.6	8.0	2733.9	124.9	28.78%	
CMT3-n99m12-G		4696.0	7.6	2899.6	118.1	61.95%	
CMT3-n99m12-H		4085.2	7.7	2642.9	148.3	54.57%	
CMT3-n99m12-I		3863.0	7.2	2842.5	118.9	35.90%	
CMT3-n99m12-J		3573.3	8.1	2841.4	150.6	46.31%	

5.2.3. Sensitivity analysis

In this section, the model parameters are examined to derive the managerial insights. Thus, several sensitivity analyses are presented in terms of the transfer points availability, the crowdsources cost levels, and the hours of service. Two categories of results (i.e. optimal results and heuristics results) are presented in two different representations, namely objective function and the number of outsourced customers order. The first category of results consists of small-size problems, H-n15m5 (B1-B5 and C1-C5) with 15 customers and 5 transfer points. In the second category, large-size instances (problem sets P and CMT) are used. In addition, ten test instances are generated by the two-phase process for the test problem CMT3-n99m12, with 99 customers and 12 transfer points to get CMT3-n99m12A – J. The small-size problems are solved by GUROBI solver to optimality and large-size problems are solved by TS algorithm for an approximate solution. All parameters are preserved in Table 7, except the one focused in each sub-subsection. As a baseline, all results in both problem scales are compared with the pure truck delivery (no outsourcing) strategy, in which the related decision variables in the model are fixed, or the TS algorithm is modified accordingly.

5.2.3.1. **Availability of transfer points**

In this part, the number of available transfer points are exercised in a form of sensitivity analysis. As the base instance, initial value of transfer point is set to five for the small problems and twelve for the large problems while keeping other parameters values in Table 7.

Figure 14 and **Figure 15** present the overall cost and the number of crowdsourced customer orders to emphasize the relationship of transfer points availability to the cost saving by the crowdsource delivery.

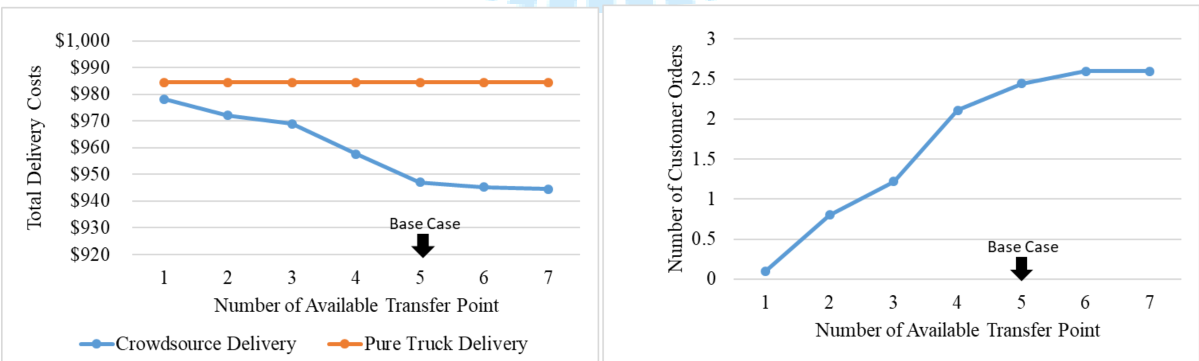


Figure 14. Total delivery cost and number of outsourced customer orders for various available transfer points (small instances with optimal results)

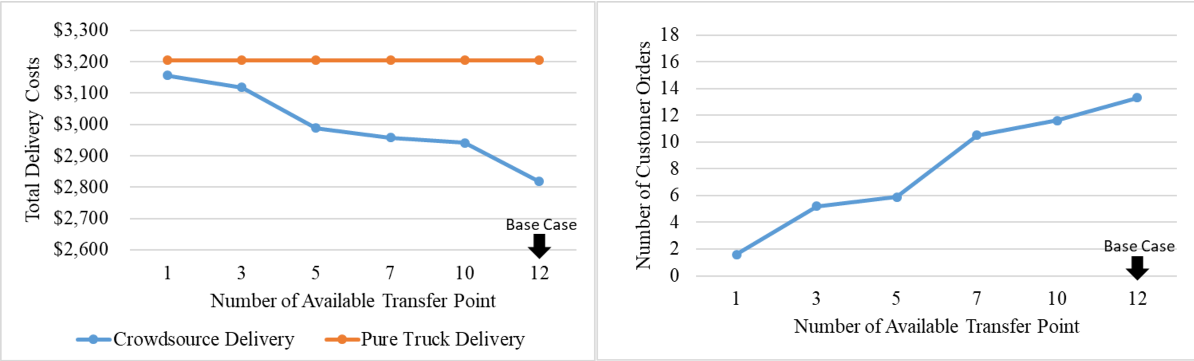


Figure 15. Total delivery cost and number of outsourced customer orders for various available transfer points (large instances with heuristic results)

The results show the importance of the transfer points as it can provide a significant cost savings from the crowdsource delivery integration. The number of available transfer points are reduced from the base case to as few as one to examine the impact. Similar trends are observed for the small and large problem sizes. Based on the results, less available transfer points may increase the overall costs substantially, together with the number of crowdsourced customer reduction. However, the crowdsource delivery are still better than non-outsourcing strategy, even at the lowest number of available transfer points. As an average, reducing one transfer point may increase the overall cost for about 2.3% (or $65) based on the large test problem.

As an insight, the number of available transfer points and its locations can be considered as operational or tactical decision in the crowdsource delivery problem. The potential of transfer points impact can be maximized and adjusted to cope with the dynamic day-to-day situation since many public spaces are available for free. However, this decision may become even more crucial for achieving the benefit of integrating the crowdsources if some amounts of costs (e.g., a renting or parking cost) are required to use the transfer points.

5.2.3.2.**Crowdsourcing costs**

In this sub-subsection, the sensitivity analysis is performed with respect of different crowdsourcing costs given the base cases with the crowdsource fixed cost of $5/crowdsource and the crowdsource variable cost of $10/ hour. **Figure 16** and **Figure 17** present the results in terms of the overall cost and the number of crowdsourced customer orders to emphasize the relationship between the level of crowdsourcing cost to the overall cost and the outsourcing decision.

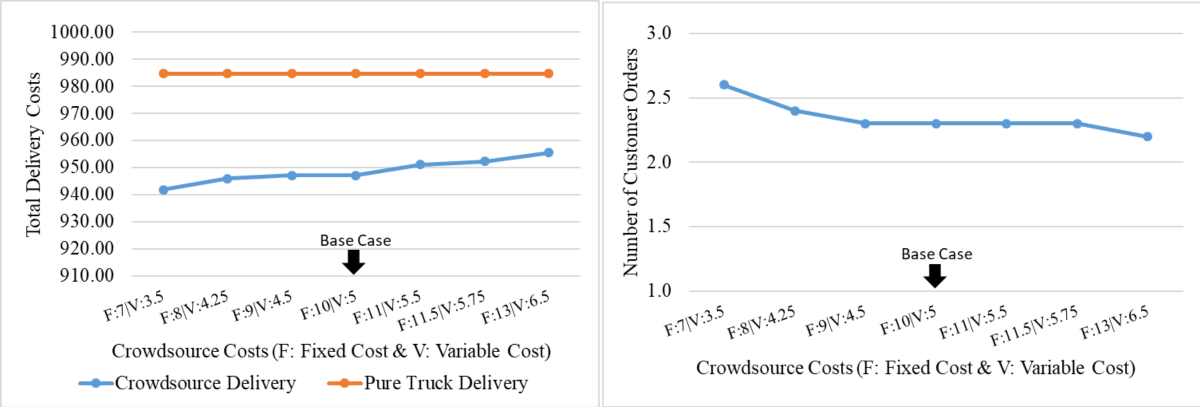


Figure 16. Total delivery cost and number of outsourced customer orders for various crowdsourcing cost level (small instances with optimal results)

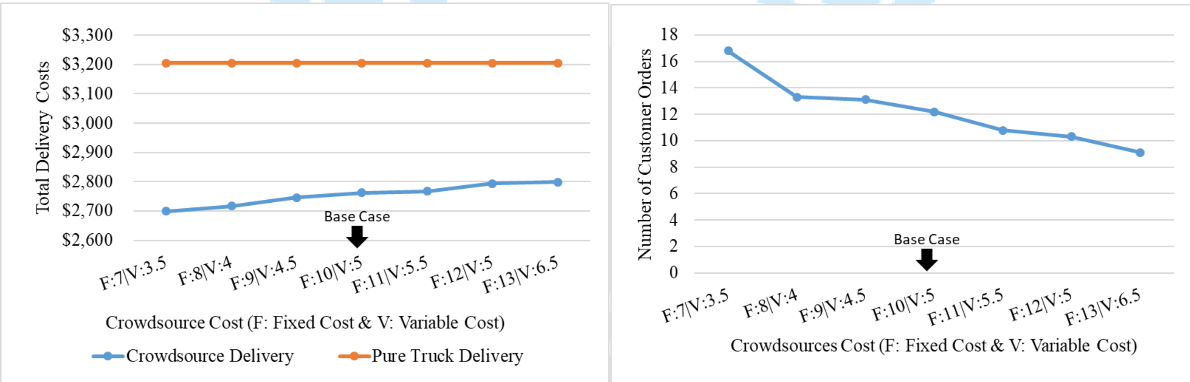


Figure 17. Total delivery cost and number of outsourced customer orders for various crowdsourcing cost level (large instances with heuristic results)

As indicated in the introduction and the design of heuristic algorithm, the balance between the crowdsources service and delivery truck utilization is the focus of this study. The crowdsouring costs in the bases cases are increased and decreased by 30% in order to illustrate the model relationship to the different level of crowdsource costs. Similar trends have been observed in both problem sizes. The percentage difference in crowdsource cost is substantially bigger than the overall costs (about 30% vs. 4% for the large test problems). Based on the large problem instances, cost savings for about 14.5% (or $405) can still be preserved even with the 30% crowdsource cost increase. As expected from the higher crowdsourcing cost, the interest of integrating crowdsources to the delivery plan is reduced.

The results of this analysis can be usefull as a reference to adjust the crowdsource cost. The crowdsource cost can be adjusted to attract more crowd-worker in the low available crowdsources areas or low crowdsource availability time, as one of the crowdsource delivery problem is demand-supply matching (Rouges & Montreuil, 2014).

5.2.3.3. **Impact of hours of service**

In this sub-subsection, the hours of service representing the driver operation time is exercised. For the base cases, the hours of service are set to be eight and it will be extended or shortened by 10% to 20% to form the sensitivity analysis. The results are provided in **Figure 18** and **Figure 19**.

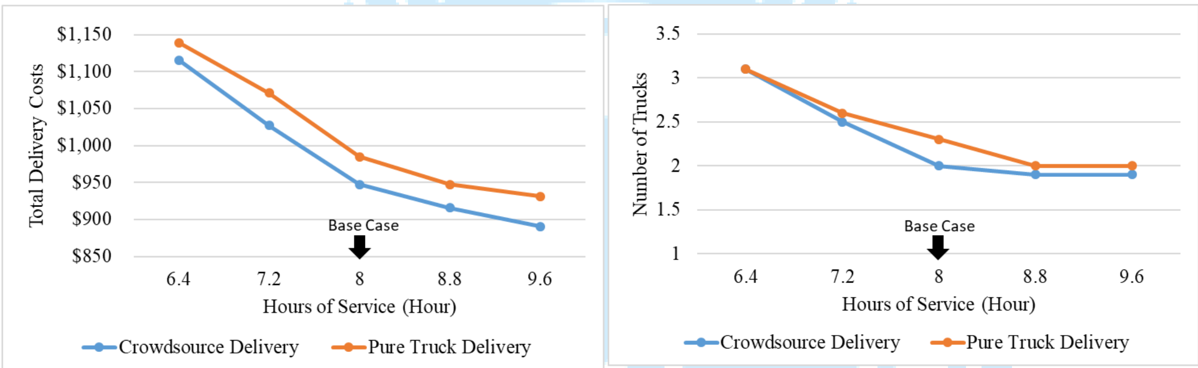


Figure 18. Total delivery cost and number of required trucks for various hours of service (small instances with optimal results)

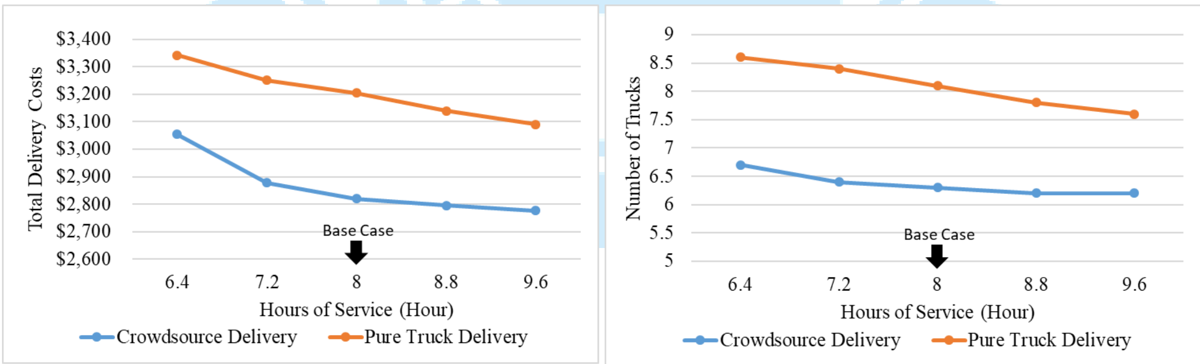


Figure 19. Total delivery cost and number of required trucks for various hours of service (large instances with heuristic results)

Based on the results, hours of service can be a crucial aspect in the crowdsource delivery integration, as extending the hours of service can lead to a cost reduction due to fewer fleets are required. A consistent cost difference between the crowdsource delivery and the pure truck delivery is observed, indicating the crowdsource delivery is able to make cost savings by utilizing the crowdsources to perform the difficult tasks under different level of hours of service. The crowdsource delivery reduces the cost to 12% (or $336) with 25% fewer trucks in average based on the large size problems.

This analysis can provide a good simulation of the relationship between the additional hours of service and the total delivery costs in terms of crowdsource delivery integration strategy. The options of overtime can be a good solution (in respect of safety regulations) to maximize the benefit of crowdsource delivery as long as the associated cost does not surpass the estimated cost saving.

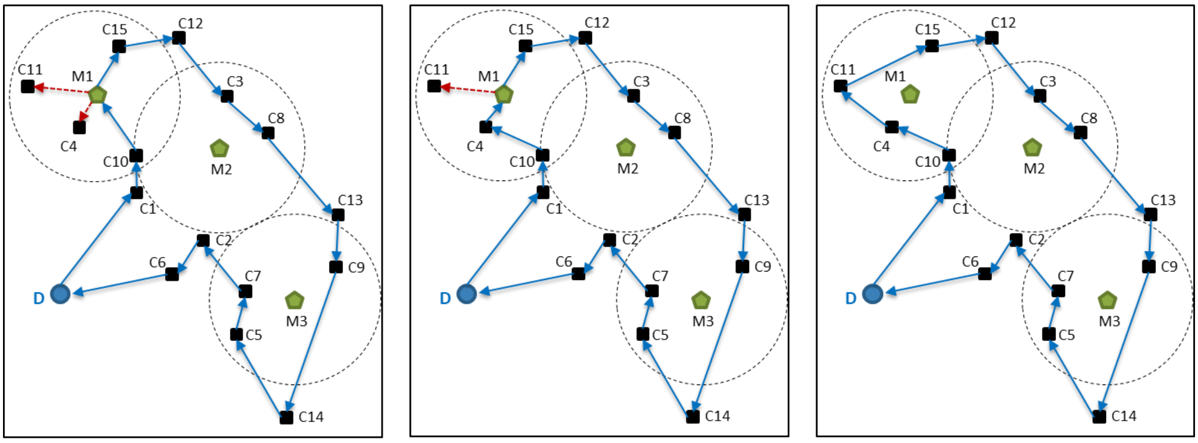
5.3. Experiment of stochastic problem

5.3.1. Results based on optimal solution

In this sub-section, all results in terms of graphs and tables are based on the optimal solutions by solving the extensive form of the model using GUROBI solver for the small problem size. The main objectives are to show the nature and behavior of the problem decision.

5.3.1.1.Illustration of crowdsources transfer uncertainty behavior

In this part, the results of considering crowdsourcing uncertainty are inspected and compared with the non-stochastic results to show the impact of the crowdsource transfer uncertainty. Problem instance "P" with 15 customers and 3 transfer points are used as an example to illustrate the final delivery plan with the parameters are preserved in Table 7. The consideration of uncertainty affects the decision to include crowdsources as highlighted in Figure 20. When considering uncertainty, the failure outcome of crowdsources transfer can be considered as an additional cost for the decision maker. When the risk is higher than the benefit to outsource customer order, then there is no advantage to use crowdsource service. As illustrated in the Figure 20, one crowdsource assignment is eliminated due to the uncertainty consideration. This result is further elaborated when the crowdsources transfer failure rate is high. It can even remove the benefit of crowdsource delivery entirely as illustrated in the last figure in Figure 20.



Deterministic Solution

Stochastic Solution
Failure rate 0.1

Stochastic Solution
Failure rate 0.3

Figure 20. Illustration of deterministic and stochastic solution

Although considering uncertainty may appear reducing the benefit of the crowdsources collaboration, it actually prevents additional loses when the crowdsource transfer failure occurs. Based on the comparison between expected deterministic solution in the uncertain environment and the stochastic solution, the advantages of considering uncertainty are beneficial. Based on Table 10, the possible loses are reduced up to 11.1% when the uncertainties are considered. This result indicates the importance of considering the uncertainties to reduce the impact of the uncertainty.

Table 10. Comparison of stochastic solution and deterministic solution in the uncertain environment

Instances	Failure Rate	Stochastic Solution	Crowdsource Transfer Realization 1 = Failure, 0 = Success		Expected Objective Function	GAP
			Transfer Success	Transfer Failure		
P	0.1	516.7	513.2	690.2	530.9	2.8%
P	0.2	520.2	513.2	690.2	548.6	5.5%
P	0.3	523.7	513.2	690.2	566.3	8.1%
H	0.1	678.2	669.2	669.2	703.9	3.8%
H	0.2	687.2	669.2	669.2	738.6	7.5%
H	0.3	696.2	669.2	669.2	773.3	11.1%
CMT1	0.1	837.5	837.5	837.5	837.5	0.0%
CMT1	0.2	837.5	837.5	837.5	837.5	0.0%
CMT1	0.3	837.5	837.5	837.5	837.5	0.0%
CMT2	0.1	847.3	843.3	843.3	855.0	0.9%
CMT2	0.2	851.1	843.3	843.3	866.7	1.8%
CMT2	0.3	855.0	843.3	843.3	878.4	2.7%

In the detour-combined strategy, the delivery truck which carry the customer's order will make additional detour from the transfer point to the customer location, otherwise a penalty will be imposed to represent the next-day delivery. The penalty as a valuation of customer order can affect the decision of initiating truck detour. Fast delivery or one-day delivery service creates high valuation of customer order or penalty. In this condition, imposing penalty can have higher cost than initiaing a truck detour. In the other hand, low penalty value can make the detour trip useless, as skipping a customer order and pay penalty can be cheaper than initiating a detour trip. The illustration of the difference between low and high penalty are highlighted in Figure 21.

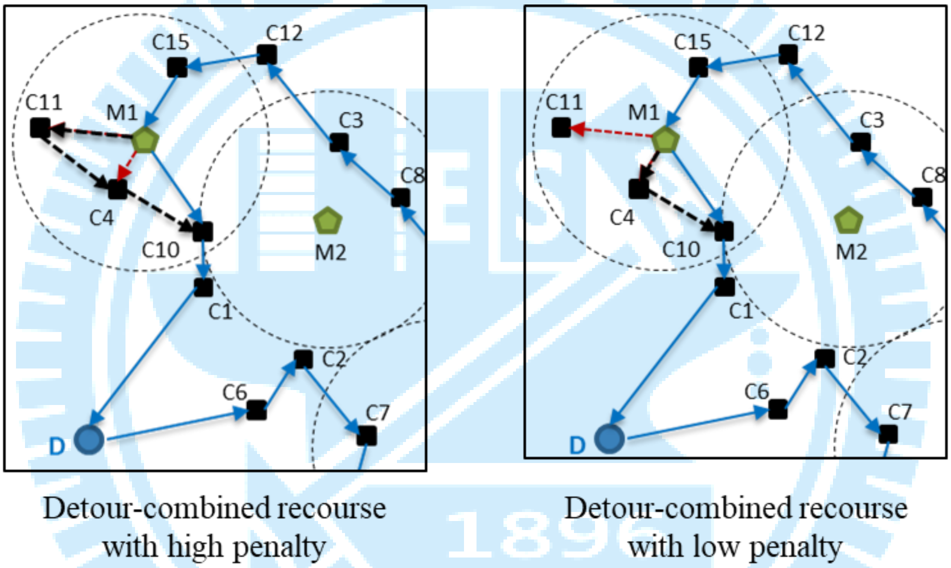


Figure 21. Illustration of detour-combined recourse under different customer order valuation

The detour-combined model in (42) - (60) generates the delivery truck final route and the truck detour route for all realizations. Each realization 's detour truck can be regarded as a backup plan when the transfer failure occurs, as it contains every single combination of transfer failures. Therefore, in practical situation, this detour route can be useful for the delivery truck driver as their basis when crowdsources transfers fail.

5.3.1.2.Comparison of deterministic model results and stochastic model results

This part of the experiment emphasizes the comparison of optimal results in small problem instances between two different recourse strategies over different transfer failure probability.

As a baseline, the deterministic results are included. All of the problem instances used have the same problem size which is 15 customers and 3 transfer points.

Table 11. Comparison of different recourse strategies

Problem Instance	**Penalty**	**Deterministic**		**Penalty-only Recourse**		**Detour-combined Recourse**			**GAP**
		Obj. Func.	**#Crowds**	**Obj. Func.**	**#Crowds**	**Obj. Func.**	**#Crowds**	**Fulfilment %**	
H	$100	669.21	4	703.91	4	678.21	4	100	5.19
P	$100	513.25	2	523.06	1	516.74	2	100	1.91
CMT1	$100	837.51	0	837.51	0	837.51	0	100	0.00
CMT2	$100	843.27	3	847.31	2	847.31	2	100	0.48
CMT3	$100	917.19	1	925.69	1	921.48	1	100	0.93
H	$50	669.21	4	683.91	4	678.11	4	75	2.20
P	$50	513.25	2	518.06	1	516.74	2	100	0.94
CMT1	$50	837.51	0	837.51	0	837.51	0	100	0.00
CMT2	$50	843.27	3	847.31	2	847.31	2	100	0.48
CMT3	$50	917.19	1	920.69	1	920.69	1	0	0.38
H	$20	669.21	4	671.91	4	671.62	4	25	0.40
P	$20	513.25	2	514.95	2	513.97	2	50	0.33
CMT1	$20	837.51	0	837.51	0	837.51	0	100	0.00
CMT2	$20	843.27	3	845.33	3	845.33	3	33.33	0.24
CMT3	$20	917.19	1	917.69	1	917.69	1	0	0.05

Deterministic model results generally have lower objective function and more crowdsouring service compared to the stochastic model results. Additional risks cost for the uncertainty will increase the cost and reduce the crowdsourcing decision. However, as it is observed in the Table 10, the deterministic model results provide worse solution win the stochastic environment. The stochastic model which considers the uncertainty will reduce the impact of uncertain crowdsource transfer and provide better results in the uncertain environment.

Detour-combined recourse strategy generates better results compared to the penalty-only recourse strategy, in Table 11. The detour-combined recourse strategy has the flexibility to choose between initiating detour route or give up the detour and pay penalty when the detour cost surpasses the penalty. Thus, detour-combined strategy will have an advantage on the penalty-only recourse. When the penalty valuation is low, both of recourse strategies generated the same solution.

5.3.2. Results based on heuristics algorithm solution

5.3.2.1.Heuristic algorithm solution quality

The experiments of heuristics algorithm to generate a near-optimal solution are presented in this part. Based on Table 12, small-instances with 10 customers and 3 transfer points are exercised and compared with the baseline of optimal solution generated by the model with detour-combined recourse.

Table 12. Comparison of heuristics algorithm results and the optimal results

Instance	Penalty	OPTIMAL SOLUTIONS			HEURISTICS				
		Obj. Function	#Crowds	Fulfilment %	Best Solution	time	#Crowds	Fulfilment %	GAP
H	20.00	671.62	4	25.00	671.62	2.82	4	25.00	0.00%
H	50.00	678.11	4	75.00	678.11	2.05	4	75.00	0.00%
H	100.00	678.21	4	100.00	678.21	2.34	4	100.00	0.00%
P	20.00	513.97	2	50.00	514.95	3.90	2	0.00	0.19%
P	50.00	516.74	2	100.00	516.74	3.34	2	100.00	0.00%
P	100.00	516.74	2	100.00	516.74	2.86	2	100.00	0.00%
CMT1	20.00	837.51	0	100.00	837.51	1.32	0	100.00	0.00%
CMT1	50.00	837.51	0	100.00	837.51	1.54	0	100.00	0.00%
CMT1	100.00	837.51	0	100.00	837.51	1.63	0	100.00	0.00%
CMT2	20.00	845.33	3	33.33	845.53	2.85	3	33.33	0.02%
CMT2	50.00	847.31	2	100.00	847.31	3.04	2	100.00	0.00%
CMT2	100.00	847.31	2	100.00	847.31	2.93	2	100.00	0.00%
CMT3	20.00	917.69	1	0.00	917.69	3.22	1	0.00	0.00%
CMT3	50.00	920.69	1	0.00	922.50	3.07	1	100.00	0.20%
CMT3	100.00	921.48	1	100.00	922.50	2.76	1	100.00	0.11%

In terms of solution quality, the heuristics algorithm generates a near-optimal solution with average gap between optimal solution and heuristics algorithm solution is less than 0.1% for the small problem size. Generaly, the differences are caused by the detour route fulfillment percentage as some of the customer order are deliberately omitted due to low penalty cost.

5.3.2.2.Performance of heuristic algorithm of medium to large size problem

This part of the experiment has been performed mainly for investigating the performance of heuristics algorithm in the medium to large size problem. The various problem instances (P and CMT) with different problem scale denoted by number of customer, N and number of transfer point available, M are utilized in this experiment. All parameters used are provided in

Table **7**.

Table 13. Performance of heuristics algorithm in medium-large size problem

Instance	N	M	Initial Solution	Comp. Time	Best solution	Comp. Time	#Crowds	Fulfilment %	%Improvement
P	20	7	1144.64	0.22	939.95	7.78	5	100.00	21.78%
P	30	10	1665.59	1.03	1166.48	18.51	9	100.00	42.79%
P	50	12	2513.83	8.71	1828.47	58.53	10	80.00	37.48%
P	74	12	3406.57	7.15	2359.02	106.51	11	81.82	44.41%
P	100	12	3616.37	8.48	2802.48	166.56	15	80.00	29.04%
CMT1	20	7	1156.09	0.19	956.74	6.52	3	66.67	20.84%
CMT1	30	10	1600.63	1.09	1109.47	16.09	8	87.50	44.27%
CMT1	49	12	2103.18	8.63	1675.35	51.69	10	90.00	25.54%
CMT2	20	7	1545.58	0.18	980.47	6.41	4	75.00	57.64%
CMT2	30	10	1910.20	1.10	1237.04	15.49	10	60.00	54.42%
CMT2	50	12	2328.62	7.35	1630.20	51.07	5	80.00	42.84%
CMT2	74	12	3997.74	4.61	2349.26	90.74	14	78.57	70.17%
CMT3	20	7	1249.14	0.13	1040.66	5.40	2	100.00	20.03%
CMT3	30	10	1802.71	1.08	1318.62	19.56	4	100.00	36.71%
CMT3	50	12	3137.62	6.44	1849.17	57.63	13	76.92	69.68%
CMT3	75	12	3809.56	5.51	2434.28	92.28	14	100.00	56.50%
CMT3	99	12	3854.18	6.17	2852.67	155.25	15	66.67	35.11%
CMT3-A	99	12	4250.33	6.37	2819.37	188.93	15	53.33	50.75%
CMT3-B	99	12	3991.21	9.75	2821.56	185.95	18	77.78	41.45%
CMT3-C	99	12	4334.5	5.42	2993.45	187.06	10	90	44.80%
CMT3-D	99	12	4273.29	8.68	2754	190.99	6	66.67	55.17%
CMT3-E	99	12	3991.02	10.56	2855.61	224.65	5	100	39.76%
CMT3-F	99	12	4110.98	11.26	2808.82	233.83	10	60	46.36%
CMT3-G	99	12	4121.68	6.13	2878.89	170.78	7	100	43.17%
CMT3-H	99	12	3982.15	9.43	2692.48	187.21	8	75	47.90%
CMT3-I	99	12	4447.01	8.69	2862.08	177.89	8	87.5	55.38%
CMT3-J	99	12	3990.96	8.89	2914.69	186.43	8	100	36.93%

As there is no optimal solution available for the baseline to evaluate the optimality, the quality of solutions is evaluated by the improvement of the final solution compared to the initial solution generated by initial solution algorithm (indicated in the last column of the table). Based on Table 13, substantial of improvements are reported indicating a good heuristics solution are produced with the average improvement is 43.37% compared to the initial solution. In particular, the large improvement shows that the proposed heuristics algorithm is not dependent on the initial solution.

In terms of computation time, the biggest problem instance with 99 customers and 12 available transfer points can be solved in 189 seconds or less than 4 minutes in average. The growth of the heuristics algorithm runtime depends on the problem size (e.g. the number of customer and the number of available transfer point). In average, it can be best approximated by $runtime = 0.017n^2 + 1.82m$, where n is the number of customers and m is the number of available transfer points.

5.3.3. Analysis of stochastic crowdsource delivery

In this part, this study investigates the impact of parameters to the decision output and managerial implications. There are two experiments in the form of sensitivity analysis which are performed, namely the crowdsource transfer failure rate and failure transfer penalty. The results are presented in the graphs which contain of objective function and the number of outsourced customer order with all parameters preserved in Table 7, except one parameter which is being inspected in each section. All results are generated based on the average of 10 problem instances (CMT3A-J) with 99 customers and 12 transfer points.

5.3.3.1.Impact of failure rates

In this analysis, the crowdsources transfer failure rates are exercised from 10% to 50%. The results are derived under the detour-combined strategy and presented in the form of the overall cost and the number of outsourced customer in Figure 22. For the base case, the failure rates are set to be 10%.

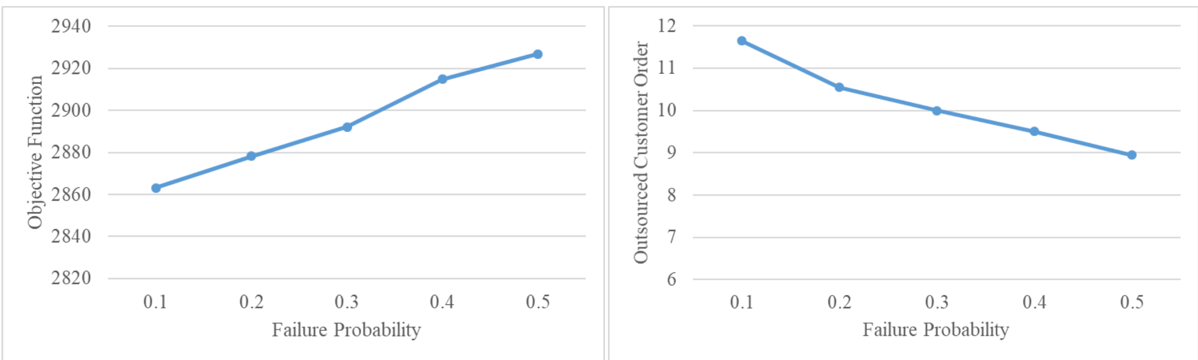


Figure 22. Impact of crowdsource transfer failure rates

As expected, the higher failure rates increase the overall cost due to higher expected penalty costs. In average, each 10% increase of crowdsource transfer failure rate will increase the cost around 0.6% ($16). The cost increases are also subjected based on the solution change in terms of the outsourced customer order. A negative trend is found in the outsourced customer order when the failure probability increases. Based on the large test problems, an increase of 10% failure probability will reduce one outsourced customer orders in average. The crowdsources assignment becomes unattractive when the valuation of risk in terms of detour cost or penalty is higher than the benefit of assigning customer order to crowdsources although the crowdsources delivery is beneficial in the deterministic environment.

Based on these experiments, determining the failure probability is important since it directly affect the final decision of this problem. The environment factor such as weather prediction, traffic condition report, etc. can be a good factor to consider in determining the failure probability, as well as internal factor such as the crowdsources track record or even driver record.

5.3.3.2.Comparison of different penalty values on the failure recourse strategies

In this sub-subsection, the impact of penalty value is investigated to analyze the impact of penalty value to the selection of recourse strategies. This experiment is performed with respect to the penalty value across different probability. The results are presented in terms of overall cost in Figure 23.

Based on the Figure 23, small difference in terms of overall cost can be found between penalty-only strategy and detour-combined strategy given the low valuation of penalty. However, when high penalty cost is imposed, higher gaps are observed between those two recourse strategies indicating that a substantial amount of money can be saved by using the recourse strategy. The cost gaps are also different across different crowdsource failure rate.

Detour-combined recourse strategy can be very complicated for logistics operator compared to the penalty-only recourse strategy as a representation of skipping the crowdsource transfer failure and resend the customer order next day. However, detour-combined recourse strategy can be very beneficial to the logistics operator as it can keep a good customer service level or to maintain the special delivery service (e.g. one-day rush delivery service) while still having efficient delivery cost in the uncertain environment.

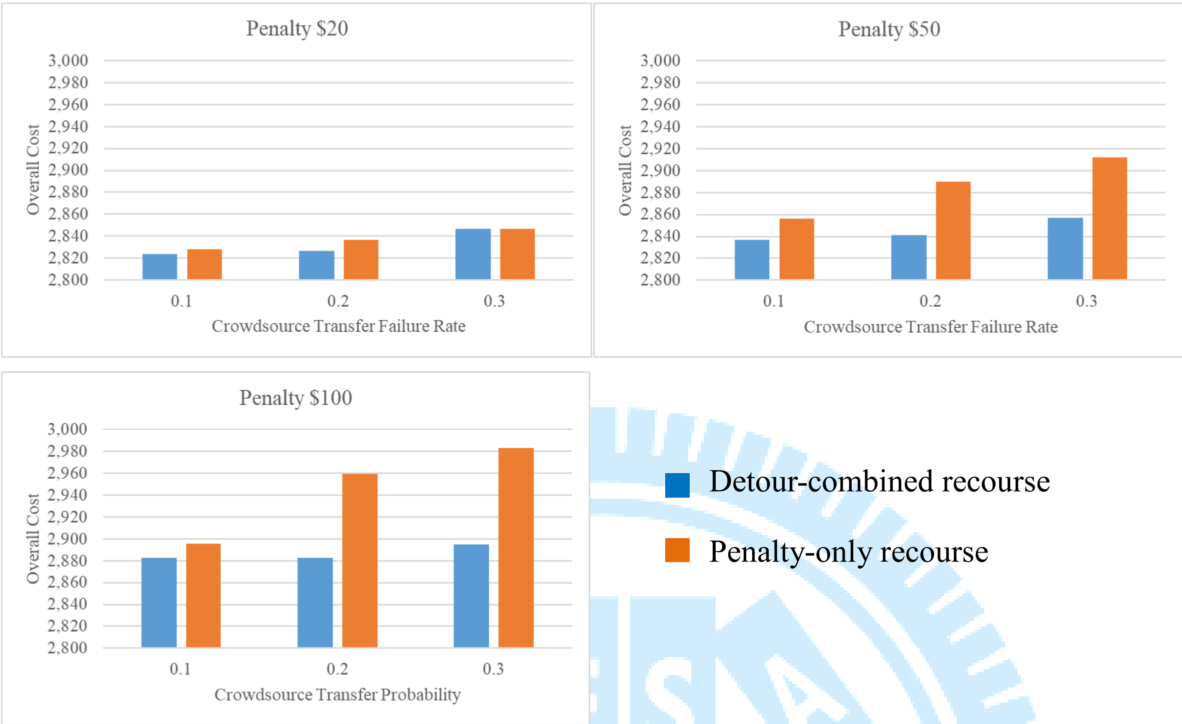


Figure 23. Impact of crowdsources transfer failure penalty rates

CHAPTER 6 CONCLUSION AND DISCUSSION

6.1. Conclusion

This study focuses on generating delivery plan for LMD with the crowdsource delivery as one of the delivery options. In this study, the logistics operator as the decision maker has two options to deliver the customer order, one being to rely on an in-house delivery truck and the other use the crowd-delivery service. The crowd-delivery must be performed through the relays at the transfer point in which crowdsources and delivery truck transfer the customer order. Overall, the results may answer several crowdsourcing decisions, such as the selection of customers to be outsourced, the selection of outsourcing partner, and the time and location to relay the customer orders. The problem is formulated as two different approaches based on the uncertainty consideration, namely deterministic model and stochastic model.

6.1.1. Deterministic model

In the deterministic model, every aspect of the problem is assumed to be deterministic. The objectives are to observe the maximum benefits of crowdsource delivery collaborations and the factors that significantly affect the decision. The deterministic problem is formulated into MILP model. As mathematical model possesses a limitation to solve large problem instances with fast computation time, the heuristics algorithm is proposed. The heuristics algorithm consists of the construction algorithm to generate initial solution and the improvement algorithm to improve the initial solution. The heuristic algorithm is designed based on the well-known TS algorithm with different types of search operators based on the unique problem features.

In general, the crowdsources delivery collaboration is able to provide cost reduction compared to the traditional last-mile delivery (pure delivery truck). Maximum benefit of crowdsource delivery can be achieved by carefully balancing the usage of crowdsourcing service and delivery fleet. These results support the past literatures findings confirming the benefits of crowdsource delivery collaboration (Kafle et al., 2017; Devari et al., 2017; Huang & Ardiansyah, 2019). Several aspects that significantly affect the decision in the deterministic model are the number of available transfer point and its location, the cost of crowdsources, and the driver hours of service.

6.1.2. Stochastic model

In the stochastic model, the main objective is to maximize the benefit of crowdsource delivery while considering the risk of uncertain crowdsource transfer. The stochastic model is an extension of the deterministic model as one of the crucial aspect, crowdsource transfer, is treated as the stochastic event. The crowdsource transfer event has two possible outcomes, namely crowdsource transfer success and crowdsource transfer failure. Recourse action strategies are defined as the back up plan to respond for the failure crowdsource transfer. Two recourse action strategies are proposed, namely penalty-only recourse strategy to represent the next-day delivery for the unsend customer order, and detour-combined recourse strategy. In this strategy the delivery truck will make additional trip to deliver the customer order which is involved in the crowdsource transfer failure. The problem and recourse strategies are formulated into two-stage stochastic MILP model. An extension of deterministic heuristic algorithm is designed to handle the stochastic model with medium to large stochastic problems with fast computation time.

The consideration of crowdsource transfer uncertainty is important to reduce the impact of crowdsource transfer failure. Prior to the uncertainty consideration, the crowdsource delivery collaboration can still provide cost reduction compared to the traditional last-mile delivery (pure truck delivery). The crowdsource failure rate, the penalty rate of omitting customer order during detour trip, and the recourse strategy are the variables which can significantly affect the results. Detour-combined recourse strategy can provide better cost reduction compared to the penalty-only recourse strategy, especially when the penalty or customer order valuation is high (e.g. rush delivery, one-day delivery).

6.2. Discussion

This study assumes several realistic issues that limit the implementation of the model in the real LMD. It is assumed that crowdsources are always available in the selection process. However, the crowdsources availability may affect the results significantly. The crowdsourcing availability can be affected by many factors including the time of the day, the location of delivery, the crowdsources payment or fee, etc. The result of this study is suitable for preparing the crowdsource bidding process, as the output of the model suggests which customers order need to be outsourced, by which crowdsouces at which transfer location. Therefore, to materialize the final decision, the same model with a fixed number of crowdsources in a transfer point after the bidding process or another model from Kafle et al. (2017) can be implemented.

In addition, this model is also flexible to be implemented to the case without any crowd-bidding process.

The consideration of vehicle fix cost in the model is important for the strategic decision. It can further increase the benefit of crowdsources delivery because of the possibility to reduce the vehicle fleet. In this study, the problem is assumed to be the operational or tactical level decision. Generaly in operational or tactical level, only vehicle operational cost based on the travel time is considered. The cost related to the vehicle and driver fix costs are shared based on the hourly basis. The vehicle fleet fix cost can be accomodated as one of the objective function component if necessary.

In this study, the time window constraint associated with each customer order is not imposed due to the nature of home delivery, which is delivery time window is generaly not very restrictive for many places. Instead, the total service hour constraint is included to address the possible considerations from the operator for cost and regulation compliance. However, the model can be easily modified to adapt with the time window constraints whenever needed.

The transfer point or relay location is assumed to be the public space which is free and available (e.g. parking lot, park, etc). No cost or fee are needed to use this place. In many place, free and available public space might not be available. A substantial amount of costs might be needed to make the public place available (e.g. park fee, renting fee, etc). Additional transfer point usage cost may change the final decision. In this study, the additional transfer point usage cost can be easily added by utilizing the current available variable which indicate the transfer point usage.

The stochastic model can be considered as an extension of the deterministic model without the consideration of crowdsources assignment routing. The crowdsources delivery assignment in the stochastic model is simplified as an assignment problem due to the complexity of the model after the consideration of crowdsources transfer uncertainty. This assumption can be relaxed whanever needed with the tradeoff of the computational resources.

The stochastic model assumes the crowdsource transfer failure rate are independent among all transfer points. In reality, several transfer point can be related to each other due to the weather condition, traffic condition, or disaster event. The extension of this research can be directed to accomodate the independent assumtion of the crowdsource failure rate at transfer point.

Customer service level is important to keep the customer satisfaction for the logistic company. Due to the uncertainty, the customer service level can be affected, as some of the deliveries can be failed due to failed crowdsource transfer and skipped delivery in the detour trip. In this study, the main focus is to maximizing the benefit of crowdsources delivery by minimizing the total delivery cost. Customer service level can be accommodated in the model whenever needed by applying the service level constraints or additional objective function component to maintain the service level.

Additional trip to fix the crowdsource transfer failure (detour trip) adds travel times to the initial delivery plan. The additional travel time accumulates everytime the detour trip is initiated. The accumulation of travel time creates a chain effect which affect the crowdsources willingness to wait for the delivery truck and perform the parcel transfer or relay process. The extension of this study can be directed to consider the accumulated detour trip waiting time for crowdsources, as it can reshape the outcome of the crowdsources relay and transfer process.

As for the heuristic solution algorithm, the design of the heuristics algorithms can further be improved by implementing another metaheuristics or hybrid-based heuristics. Another good heuristics approach to solve stochastic model is the L-shape algorithm which based on the decomposition technique. It can generate optimal solution in acceptable amount of time for medium to large size problem instance. The optimal results for medium size problem can be used as a baseline solution to verify the solution improvement of any heuristic solution algorithm.

REFERENCES

ABS-CBN News. (2018, April 23). Grab sanctions 500 drivers over unjustified cancellations. Retrieved May 30, 2018, from http://news.abs-cbn.com: http://news.abs-cbn.com/business/04/23/18/grab-sanctions-500-drivers-over-unjustified-cancellations

Akeb, H., Moncef, B., & Durand, B. (2018). Building a collaborative solution in dense urban city settings to enhance parcel delivery: An effective crowd model in Paris. Transportation Research Part E, 119, 223-233.

Allen, A. J., Browne, M., & Jones, P. (2000). A framework for considering policies to encourage sustainable urban freight traffic and goods/service flows: summary report. London: Transport Study Group, University of Westminster.

Archetti, C., Savelsbergh, M., Speranza, M.G. (2016). The vehicle routing problem with occasional drivers. European Journal of Operational Research, 254 (2), 472-480.

Arslan, A., Agatz, N., Kroon, L., & Zuidwijk, R. (2019). Crowdsourced Delivery: A Dynamic Pickup and Delivery Problem with Ad-Hoc Drivers. Transportation Science, 53 (1), 1-318.

Augerat, P., Belenguer, J.M., Benavent, E., Corberán, A., Naddef, D., Rinaldi, G. (1995). Computational results with a branch and cut code for the capacitated vehicle routing problem. Technical Report RR 949-M, Université Joseph Fourier, Grenoble.

Barbarosoglu, G., & Ozgur, D. (1999) A tabu search algorithm for the vehicle routing problem. Computers & Operations Research, 26, 255-270.

Barr, A., & Wohl., J. (2013) Exclusive: Wal-mart may get customers to deliver packages to online buyers. Reuters, March 28. Available at: https://www.reuters.com/article/us-retail-walmart-delivery/exclusive-wal-mart-may-get-customers-to-deliver-packages-to-online-buyers-idUSBRE92R03820130328. (Accessed: 10 July 2019)

Belgin, O., Karaoglan., I., & Altiparmak, F. (2018). Two-echelon vehicle routing problem with simultaneous pickup and delivery: Mathematical model and heuristic approach. Computer & Industrial Engineering, 115, 1-16.

Carbone, V., Rouquet, A., & Roussat, C. (2017). The rise of crowd logistics: A new way to co-create logistics value. Journal of Business Logistics, 38 (4), 238-252.

Chao, I. M. (2002). A tahu search method for the truck and trailer routing problem. Computer & Operations Research, 29 (1), 33-51.

Chen C., Pan S. (2016) Using the Crowd of Taxis to Last Mile Delivery in E-Commerce: a methodological research. In: Borangiu T., Trentesaux D., Thomas A., McFarlane D. (eds) Service Orientation in Holonic and Multi-Agent Manufacturing. Studies in Computational Intelligence, 640. Springer, Cham

Chen, C., Pan, S., Wang, Z., & Zhong, R. Y. (2017). Using taxis to collect citywide e-commerce reverse flows: a crowdsourcing solution. International Journal of Production Research, 55 (7), 1833-1844.

Christofides, N., Mingozzi, A., and Toth, P. (1979). The vehicle routing problem, in: N. Christofides, A. Mingozzi, P. Toth and C. Sandi (eds.), Combinatorial Optimization, Wiley, Chichester, 315-338.

Cuda, R., Guastaroba, G., & Speranza, M. (2015). A survey on two-echelon routing problems. Computers & Operations Research, 55, 185-199.

De Souza, R., Goh, R., Lau, H. C., Ng, W. S., & Tan, P. S. (2014). Collaborative Urban Logistics – Synchronizing the Last Mile a Singapore Research Perspective. Procedia - Social and Behavioral Sciences, 125, 422-431.

DellAmico, M., & Hadjidimitriou, S. (2012). Innovative logistics model and containers solution for efficient last mile delivery. Procedia - Social Behavior Science, 48, 1505-1514.

Devari, A., Nikolaev, A. G., & He, Q. (2017). Crowdsourcing the last mile delivery of online orders by exploiting the social networks of retail store customers. Transportation Research Part E, 105, 105-122.

Dough (2018). Ridesharing Driver. Retrieved on February 16, 2018 from www.ridesharingdriver.com: https://www.ridesharingdriver.com/whats-the-difference-between-uberx-xl-uberplus-and-black-car/

Gendreau, M., Jabali, O., & Rei, W. (2016). Future research directions in stochastic vehicle routing. Transportation Science, 4, 1163-1173.

Gendreau, M., Laporte, G., & Seguin, R. (1996). Stochastic vehicle routing. European Journal of Operational Research, 88 (1), 3-12.

Gevaers, R., Van de Voorde, E., & Thierry, V. (2009). Characteristics of innovations in last-mile logistics-using best practices, case studies and making the. European Transport Conference. Leiden Leeuwenhorst Conference Centre, Netherlands.

Gdowska, K., Viana, A., Pedroso., J. (2018). Stochastic last-mile delivery with crowdshipping. Transportation Research Procedia, 30, 90- 100.

Ghiani, G., Laporte, G., & Musmanno, R. (2004). Introduction to Logistics System Planning and Control. Chichester: John Wiley & Sons.

Glover, F. (1986). Future paths for integer programming and links to artificial intelligence. Computer and Operations Research, 13, 533-549.

Gurobi. (2019). Gurobi. Retrieved from www.gurobi.com: http://www.gurobi.com

Heilporn, G., Cordeau, J.F., & Laporte, G. (2011). An Integer L-shaped algorithm for the dial-a-ride problem with stochastic customer delay. Discrete Applied Mathematics, 159 (9), 883-895

Huang, K., Wu, K.-F., & Ardiansyah, M. N. (2018). A stochastic dairy transportation problem considering collection and delivery phases. Transportation Research Part E. https://doi.org/10.1016/j.tre.2018.01.018

Huang, K., & Ardiansyah, M. (2019). A decision model for last-mile delivery planning with crowdsourcing integration. Computers & Industrial Engineering, 135, 898-912.

Hurkens, C. O. A., & Woeginger, G. J. (2004). On the nearest neighbor rule for the traveling salesman problem. Operations Research Letters, 32(1), 1-4.

Kafle, N., Zou, B., & Lin, J. (2017). Design and modeling of a crowdsource-enabled system for urban parcel relay and delivery. Transportation Research Part B, 99, 62-82.

Liakos, P., & Delis, A. (2015). An interactive freight-pooling service for efficient last-mile delivery. 2015 16th IEEE Conference on Mobile Data Management (MDM) (pp. 23-25). Pittsburgh, PA, USA: IEEE.

McKinnon, A. (2016) Crowdshipping: a communal approach to reducing urban traffic levels. https://www.alanmckinnon.co.uk/story_layout.html?IDX=714&b=56. Accessed 10 July 2019.

Nguyen, V-P., Prins, C., & Prodhon, C. (2012). Solving the two-echelon location routing problem by a GRASP reinforced by a learning process and path relinking. European Journal of Operational Research, 216, 113-126.

Paloheimo, H., Lettenmeier, M., & Waris, H. (2016). Transport reduction by crowdsourced deliveries - a library case in Finland. Journal of Cleaner Production, 132, 240-251.

Park, H., Park, D., & Jeong, I.-J. (2016). An effects analysis of logistics collaboration in last-mile networks for CEP delivery services. Transport Policy, 50, 115-125.

Pichka, K., Bajgiran, A. H., Petering, M. E. H., Jang, J., & Yue, X. (2018). The two-echelon open location routing problem: Mathematical model and hybrid heuristic. Computer & Industrial Engineering, 121, 97-112.

Petrovic, O., Harnisch, M. J., & Puchleitner, T. (2013). Opportunities of mobile communication systems for applications in last-mile logistics. International Conference on Advanced Logistics and Transport (ICALT), (pp. 354–359). Sousse.

Punakivi, M., Yrjola, H., & Holmstrom, J. (2001). Solving the last mile issue: reception box or delivery box? International Journal of Physical Distribution & Logistics Management, 31 (6), 427-439.

Punel, A., & Stathopoulos, A. (2017). Modeling the acceptability of crowdsourced goods deliveries: Role of context and experience effects. Transportation Research Part E, 105, 18-38.

Reilly, D., 2015. Amazon ponders crowdsourcing deliveries with 'On My Way' program. Fortune: http://fortune.com/2015/06/16/amazon-crowdsource. Accessed 10 July 2019.

Ritzinger, U., Puchinger, J., Hartl, R. (2016). A survey on dynamic and stochastic vehicle routing problems. International Journal of Production Research, 54(1), 215-231.

Rob, O. (2016). The uberisation of logistics. MHD Supply Chain Solutions, 46(5), 26-28. Retrieved from https://search.informit.com.au/documentSummary;dn=465114672772785;res=IELBUS

Rothenbacher, A., Drexl, M., Irnich, S. (2018) Branch-and-price-and-cut for the truck-and-trailer routing problem with time windows. Transportation Science, 52(5), 1035-1296.

Rouges, J.-F., & Montreuil, B. (2014). Crowdsourcing delivery: New interconnected business models to reinvent delivery. 1st International Physical Internet Conference. Quebec City.

Siddiqui, F. (2016, November 1). Your Uber driver is twice as likely to cancel if you are black. Retrieved from https://www.washingtonpost.com:

https://www.washingtonpost.com/news/tripping/wp/2016/11/01/your-uber-driver-is-twice-as-likely-to-cancel-if-youre-black/?noredirect=on&utm_term=.951b887d6cfe.

Saint-Guillain M., Solnon C., Deville Y. (2017) The Static and Stochastic VRP with Time Windows and both Random Customers and Reveal Times. In: Squillero G., Sim K. (eds) Applications of Evolutionary Computation. EvoApplications 2017. Lecture Notes in Computer Science, vol 10200. Springer, Cham

Sampaio, A., Savelsbergh, M., Veelenturf, L., van Woensel, T. (2019). Crowd-Based City Logistics, In Sustainable Transportation and Smart Logistics. Elsevier, 381-400.

Scheuerer, S. (2006). A tabu search heuristics for the truck and trailer routing problem. Computer & Operation Research, 33(4), 894-909.

Su, H., Zhen, K., Huang, J., Jeung, H., Chen, L., & Zhou, X. (2014). CrowdPlanner: A Crowd-Based Route Recommendation System. The 30th International Conference on Data Engineering. Chicago, IL, USA: IEEE.

Sungur, I., Ren, Y., Ordonez, F., & Dessouky, M. (2010). A Model and Algorithm for the Courier Delivery Problem with Uncertainty. Transportation Science, 44, 193-205.

Ulmer M.W., Brinkmann J., Mattfeld D.C. (2015) Anticipatory Planning for Courier, Express and Parcel Services. In: Dethloff J., Haasis HD., Kopfer H., Kotzab H., Schönberger J. (eds) Logistics Management. Lecture Notes in Logistics. Springer, Cham

Ulmer M.W., Mattfeld D.C., Hennig M., Goodson J.C. (2016) A Rollout Algorithm for Vehicle Routing with Stochastic Customer Requests. In: Mattfeld D., Spengler T., Brinkmann J., Grunewald M. (eds) Logistics Management. Lecture Notes in Logistics. Springer, Cham

Wang, Y., Zhang, D., Liu, Q., Shen, F., & Lee, L. H. (2016). Towards enhancing the last-mile delivery: An effective crowd-tasking model with scalable solutions. Transportation Research Part E, 93, 279-293.

Wang, J., Jagannathan, A.K.R., Zuo, X., & Murray, C.C. (2017). Two-layer simulated annealing and tabu search heuristics for a vehicle routing problem with cross docks and split deliveries. Computers & Industrial Engineering, 112, 84-98.

Willard, J. (1989). Vehicle routing using r-optimal tabu search. M.Sc. Dissertation. The Management School, Imperial College. London.

Zhou, L., Baldacci, R., Vigo, D., Wang, X. (2018). A multi-depot two-echelon vehicle routing problem with delivery options arising in the last mile distribution. European Journal of Operational Research, 265, 765-778.

APPENDIX 1. Vehicle route generation procedure

Step 0: Let $\mathcal{R}^v$ be the set of initial vehicle routes consisting of $r_i = \{0, i, 0\}, i \in (N \backslash R^c) \cup R^m \cup \{0\}$.

The definition of $(N \backslash R^c) \cup R^m \cup \{0\}$ is provided in Step 2 of construction algorithm.

Let $\mathcal{A}$ be the set of angle in two dimensional with depot as the coordinate zero (0,0). For each node $i \in (N \backslash R^c) \cup R^m \cup \{0\}$, $a_i = \tan^{-1}\left(\frac{y_i - y_0}{x_i - x_0}\right)$.

Step 1: Set the base angle, $\mathcal{B}$ equals to 0 and sort the angle list $\mathcal{A}$ in an increasing order starts from $\mathcal{B}$.

Step 2: Extract an angle a_i from $\mathcal{A}$.

Merge node i with any possible route containing the nearest node to node i in $\mathcal{R}^v$ and remove r_i from $\mathcal{R}^v$ if merging node i with another route in $\mathcal{R}^v$ is feasible.

Step 3: Continue to Step 4, if all elements in $\mathcal{A} = \emptyset$. Otherwise go back to step 2.

Step 4: Continue to increase the base angle, $\mathcal{B} = \mathcal{B} + 5^0$. STOP if $\mathcal{B} > 360$.

APPENDIX 2. Crowdsource route generation procedure

Step 0: Let $\mathcal{R}^c$ be the set of initial crowdsource routes, containing the route $\{l, i\}$, from the transfer point l to each customer node $i, i \in R_l^c$. Thus, $|\mathcal{R}^c| = |R_l^c|$.

The definition of R_l^c is provided in Step 3 of construction algorithm (Page 12).

Step 1: For each pair of elements in R_l^c, calculate the saving cost, which is defined by $\mathcal{S}_{ij} = T_{li}^c + T_{lj}^c - T_{ij}^c, i, j \in R_l^c$. Sort the saving costs, $\mathcal{S}_{ij}$ in an decreasing order from the biggest to the smallest.

Step 2: Select a pair of node i, j based on the sorted saving cost, $\mathcal{S}_{ij}$.

Merge the two crowdsource routes containing i and j to create a new route $\{l, i, j\}$ which always starts from transfer point l for replacing the original two routes in $\mathcal{R}^c$ if the combined route is feasible.

Skip the merging if a pair of node i and j belong to the same route in $\mathcal{R}^c$.

Step 3: Re-do the Step 2 until all pair of nodes i, j from saving cost, $\mathcal{S}_{ij}$ are selected. Otherwise, STOP.

APPENDIX 3. Omitted customer procedure

Step 0: Let $\bar{\mathcal{X}}$ and $\bar{\mathcal{Y}}$ be the current solution consisting of vehicle routes and crowdsources assignment, respectively, as defined in the subchapter 4.1.

Let $\mathcal{L}$ be the omitted customer detour list with empty set as the initial value, $\mathcal{L} = \emptyset$

Let $\mathcal{O}_\omega$ be the omitted customer detour list in realization or event ω with empty set as the initial value $\mathcal{O}_\omega = \emptyset, \omega \in \Omega$.

Let $\mathcal{D}_l$ be detour cost which delivery truck will take when failed outcome occurs in transfer point l following the sequence of $\mathcal{Y}_l$, $l \in M$.

Step 1: Select one assignment route $\mathcal{Y}_l$ from $\bar{\mathcal{Y}}$.

Step 2: Select one customer node i_l from $\mathcal{Y}_l$ and calculate the detour cost $\bar{\mathcal{D}}_l^i$ after i is removed from $\mathcal{Y}_l$. Include i_l into the omitted customer detour list, $\mathcal{L}$, if $\mathcal{D}_l - \bar{\mathcal{D}}_l^i > \alpha$.

Repeat Step 2, until all customer nodes in $\mathcal{Y}_l$ have been selected. Otherwise, go to Step 3.

Step 3: Repeat Step 1, until all crowdsource assignment in $\bar{\mathcal{Y}}$ have been selected. Otherwise go to Step 4.

Step 4: Assign the customer node i_l in omitted customer detour, $\mathcal{L}$ to the $\mathcal{O}_\omega$ if $R_{l\omega} = 1, \omega \in \Omega$. Repeat step 4 until all elements in $\mathcal{L}$ have been assigned.