

國立成功大學
工業與資訊管理學系
碩士論文

無樁式電動機車共享系統之
最佳動態運補與眾包運送任務規劃研究

Optimal Task Planning for Dynamic Repositioning and
Crowdsourced Shipping in a Free-floating Electric Motorcycle
Sharing System

研究生：孫雪湄

指導教授：王逸琳 教授

中華民國一百零八年五月

國立成功大學

碩士論文

無樁式電動機車共享系統之最佳動態運補與眾包運送
任務規劃研究

Optimal Task Planning for Dynamic Repositioning and
Crowdsourced Shipping in a Free-floating Electric
Motorcycle Sharing System

研究生：孫雪湄

本論文業經審查及口試合格特此證明

論文考試委員：

陳凱 王逸琳

莊坤達

吳政弼

指導教授：王逸琳

系(所)主管：

王惠嘉

中華民國 108 年 5 月 26 日

摘要

將閒置資源分享共用以達成雙贏的共享經濟近年來蓬勃發展，本研究首先針對都會區內新興的無樁式電動機車共享系統，在日間的車輛配置時時受到使用者行為影響之際，探討如何指派其員工移動閒置車輛，以滿足更多租車需求之「動態運補」問題，避免車輛配置不利於使用者的租借；再進一步分析該共享系統所存取之會員騎乘路線資料，以協助「眾包運送」平台找尋合適對象，進行推播媒合度較高的運送任務。

在動態運補部分，本研究先劃分營運區域為數個虛擬站點，每 30 分鐘為一期，考慮各虛擬站點在各期的歷史租借記錄、當下閒置車數與員工人數後，再建構滾動式的數學規劃模型，以指派各員工移車或移位，將電動機車與租借需求的時地配置最佳化。由於無樁式的營運模式不受到固定站點設施的限制，讓其共享載具的租還地點與使用者的真實需求十分貼近，且該類系統每數秒會將車輛即時的狀態與位置資訊上傳，這些使用者移動路徑巨量資料加以分析後，恰好可被諸如 UberEATS 或 GOGOVAN 等國內外新創的眾包運送平台用來精準行銷，針對各運送任務找出資料庫中較順路的會員，對其推播媒合度較高的任務。本研究將以路徑分群手法統整會員的騎乘習慣，建構具有推播數量限制的任務推送數學規劃模型，避免過多的推播成為無效的垃圾訊息，達到精準媒合運送任務的效果，徹底發揮電動載具、個人時間等閒置資源的共享綜效。

關鍵字：無樁式、電動機車共享系統、動態運補、路徑分群、眾包運送

Optimal Task Planning for Dynamic Repositioning and Crowdsourced Shipping in a Free-floating Electric Motorcycle Sharing System

Shiue-Mei Sun

I-Lin Wang

Department of Industrial and Information Management

SUMMARY

The sharing economy has created opportunities for idle asset sharing and increased asset utilization, with the vehicle sharing system as a successful application. This thesis focused on issues related to the emerging free-floating electric motorcycle sharing system. We first investigate the dynamic repositioning strategies, and then discuss how to analyze the historical user trajectory data which can then be used for a crowdsourced shipping platform to seek optimally matched riders to plan shipping tasks.

In the first part, we first divide the entire operating area as a set of regions, where each region is treated as a virtual station. An ideal inventory mathematical model is proposed to determine the optimal ideal initial inventory for each period at each virtual station. Two dynamic repositioning models which assign staff to move idle motorcycles are proposed: the vehicle ideal inventory repositioning model, and the maximum demand satisfaction repositioning model.

In the second part, we explore a good application using the historical user trajectory data in this vehicle sharing system. For each user at each period, we group his/her historical trajectory data by trajectory clustering. Then, for each delivery task appeared in a crowdsourced shipping platform, we calculate the best matched users whose clustered trajectories have smaller expected detour costs, then only push these matched tasks to those users. Our proposed task pushing model achieves more accurate push effects and avoids ineffective delivery task pushes.

Key words: Free-floating, Electric motorcycle sharing system, Dynamic repositioning, Trajectory clustering, Crowdsourced shipping

INTRODUCTION

The dense urbanization phenomenon has created the popularity of vehicle sharing systems, which increase the need to use vehicles, but weaken the ownership of vehicles. In Taiwan, bike sharing systems have become popular in many cities, but the only free-floating bike sharing system (o-bike) has failed. On the other hand, a free-floating electric motorcycle sharing system launched in Taipei in 2016 is still running well and has become the largest free-floating e-motorcycle sharing system in Asia. Recently, another new free-floating e-motorcycle sharing system has launched in Taoyuan, too. We expect similar trend may appear elsewhere in the near future, because a motorcycle is more mobile and faster than a bike, easier to park, less space than a car, and an e-motorcycle is also environmental friendly. The major challenge for running such a system relies on effective repositioning strategy, which is especially difficult because the rental demands in a free-floating system are scattered in the operating area, and thus are more difficult to estimate. In addition, we need to hire staff or crowd to reposition an e-motorcycle, similar to the case of car sharing systems, but different from the case of bike sharing system which usually repositions bikes by trucks.

MATERIALS AND METHODS

In the first part, this study proposes a complete repositioning mechanism based on mathematical programming for the free-floating electric motorcycle sharing system, including the definition of virtual stations for geographical division, the vehicle ideal inventory model for the initial system configuration at each virtual station and period, and two integer programming (IP) models for vehicle repositioning decision: (1) the vehicle ideal inventory relocation model (VIIR), and (2) the maximum demand satisfaction relocation model (MDSR).

To deal with the geographically scattered demands, we divide the operating area into equidistant grids, which forms a set of rectangles that we call virtual stations. Similarly, we can divide a day into discrete time periods (e.g., every 30 minutes). The historical rental records of the same period at the same virtual station will be consolidated. Then, we propose an IP formulation to calculate the ideal vehicle inventory for each period at each virtual station, so that the unmet rental demands are minimized. With the ideal inventory as a target value to achieve, VIIR model calculates optimal staff routings for vehicle repositioning and personnel movement, which in turn helps to satisfy more rental demands indirectly. On the other hand, MDSR model tries to directly maximize the rental demands. To improve the efficiency of these IP models, we use an iterative rolling-horizon technique, where we only consider a smaller IP that covers a few contiguous periods, use part of the solution, and then repeat the same planning every few periods. The rolling-horizon mechanism can calculate an optimal solution for the nearest future periods, and is flexible to adapt new change. On

the other hand, the original IP models that consider all periods at one time is too time consuming and cannot adapt the demands that may dynamically change to be very different from the historical data.

In the second part, this study investigates a good application on the use of the recorded historical user trajectory data stored in the database of an e-motorcycle sharing system. In order to monitor the status of each e-motorcycle, the sharing system usually stores the real-time on-line data collected by the vehicle sensors and upload these data every few seconds to the cloud. Therefore, the system can not only monitor the current status of vehicles, but also record how a user rides. In other words, the sharing system could analyze the historical trajectory data to understand the riding behavior for each user. Such a historical trajectory data can be used to find suitable users to help reposition an e-motorcycle, or to suggest a user accept suitable crowdsourced shipping task. In particular, we analyze the riding habits of users in the free-floating electric motorcycle sharing system through trajectory clustering, and use this as a reference for the delivery task pushing model.

RESULTS AND DISCUSSION

We generate vehicle borrowing demand between stations by Poisson distribution. The ideal inventory model shows that when we have sufficient e-motorcycles, the dynamic repositioning does reduce the unmet demands to 3% or less, and the solution can be completed within a reasonable time. Analyzing the effects of different length of periods in the rolling-horizon planning, we find that VIIR has better performance (less unmet demands) than MDSR for cases of shorter planning horizon. On the other hand, for longer periods in the rolling-horizon planning, MDSR performs better than VIIR, possibly because the former directly reduce the unmet rental demands but the latter does so in an indirect way.

The numerical test of our second part indicates the solution time of trajectory clustering increases very quickly by the number of trajectories. We demonstrate how our proposed task push model work, and observe that the solution time is less affected by the test scale, but the number of matched tasks increases for more tasks and crowds.

CONCLUSIONS

This thesis deals with repositioning strategies and a good application for managing a free-floating e-motorcycle sharing system. The concept of the virtual station, as well as two proposed IP models are proposed, and the rolling-horizon framework can calculate routing and scheduling decisions for moving staff in an effective and efficient way. In addition, we also explain how to analyze and cluster the riding trajectories for users, and how to exploit

these data to match suitable delivery tasks. We suggest the investigation of more efficient ways to cluster trajectory as a potentially challenging topic for future research.



誌謝

首先感謝我的家人，讓我可以在台南求學生活無後顧之憂，資助我生活與出國的需求，希望我的表現不會讓你們失望。

感謝王逸琳老師，當初不嫌麻煩收一個工業與資訊管理的門外漢，給予我許多挑戰自己的機會，在過程中不辭辛勞的教導與訂正，我獲得了充實的研究所生活，還有許多寶貴的經驗，休士頓與北京的出征，中華郵政大數據比賽與富邦論文獎，都要感謝老師。

感謝昀軒在課業與生活上給予我最可靠的幫助與最大的支持，陪我走過求學的路途，並且帶我和原本陌生的同學打成一片。

感謝嘉豪，多虧有親切的你，實驗室充滿了歡笑，你對程式的熱情也著實讓我佩服，謝謝你多次的技術支援。

感謝 Meidy 與 Jib，你們的加入讓實驗室變得更國際化了，很開心有緣能與你們在台灣成為同學。

感謝筱昀、柏寬、思涵、冠瑋，親切的你們讓我得以順利適應實驗室的生活。

感謝晏慈、宗瀚、彥瑋、Ari，你們的加入讓實驗室增加了不少的戰力，有你們這樣有 sense 的學弟妹是實驗室的福氣。

謝謝裕荃、韋呈、怡潔、雅宣、陞瑋，平日的聚餐慶生、小琉球與澎湖的出遊都是很棒的回憶，研究所有你們當同學很幸運。

謝謝台南，6 年來滋潤我的生活，我想我一定會很懷念這裡的生活。

目錄

第一章 緒論	1
1.1 研究背景	1
1.2 研究動機	2
1.3 研究目的	3
1.4 論文架構	5
第二章 文獻回顧	6
2.1 載具共享系統之相關文獻	6
2.1.1 靜態運補	6
2.1.2 動態運補	7
2.2 眾包運送之相關文獻	10
2.3 路徑分群相關文獻	12
2.4 小結	13
第三章 無樁式載具共享系統動態運補研究	14
3.1 虛擬站點轉化	14
3.2 最適車數數學模式	15
3.2.1 問題描述	15
3.2.2 問題假設	16
3.2.3 數學模式	16
3.3 最適車數運補數學模式	18
3.3.1 問題描述	18
3.3.2 問題假設	18
3.3.3 數學模式	19
3.4 最大需求滿足運補數學模式	22
3.4.1 問題描述	22
3.4.2 問題假設	22
3.4.3 數學模式	23

3.4 數值測試	25
3.4.1 最適車數數學模式之數值測試.....	25
3.4.2 車輛運補數學模式之數值測試.....	27
第四章 眾包運送平台任務推送研究	31
4.1 路徑分群演算法	31
4.1.1 問題描述.....	31
4.1.2 路徑間的相異度.....	31
4.1.3 路徑分群模型.....	32
4.1.4 路徑分群演算法設計.....	33
4.2 任務推送數學模型	34
4.2.1 問題描述.....	34
4.2.2 問題假設.....	34
4.2.3 數學模式.....	34
4.3 數值測試	38
4.3.1 路徑分群演算法之數值測試.....	38
4.3.2 任務推送數學模式之數值測試.....	40
第五章 結論與未來研究方向建議	42
5.1 結論與貢獻.....	42
5.2 未來研究方向建議.....	45
參考文獻	47

表目錄

表 2.1 載具共享系統動態運補相關文獻比較表	10
表 2.2 眾包運送相關文獻比較表	12
表 2.3 路徑分群相關文獻比較表	13
表 3.1 最適車數數學模式數值測試 (站點數 $N=10$)	26
表 3.2 最適車數數學模式數值測試 (站點數 $N=50$)	26
表 3.3 最適車數數學模式數值測試 (站點數 $N=100$)	26
表 3.4 運補模式數值測試 (站點數 $N=10$)	28
表 3.5 運補模式數值測試 (站點數 $N=50$)	29
表 3.6 運補模式數值測試 (站點數 $N=100$)	30
表 4.1 路徑分群演算法流程	33
表 4.2 路徑分群演算法數值測試	39
表 4.3 任務推送數學模式數值測試 (任務數量為 20)	40
表 4.4 任務推送數學模式數值測試 (任務數量為 50、100)	41

圖目錄

圖 3.1 原始虛擬站點	14
圖 3.2 需求大於下限之虛擬站點	15
圖 3.3 時間表達定義示意圖	16
圖 3.4 第 1 次員工運補決策示意圖	20
圖 3.5 第 2 次員工運補決策示意圖	20
圖 4.1 路徑間相異度($c=10$)示意圖	31
圖 4.2 繞行路徑估計方式	35
圖 4.3 任務群眾配適前處理示意圖	36
圖 4.4 任務推送網路示意圖	36



第一章 緒論

1.1 研究背景

拜科技快速發展所賜，近年來共享經濟逐漸崛起，共享經濟是一種運用網路媒介提供共用資源服務的商業模式，包括實體的資產或虛擬的人力資源的共享，讓社會過剩的產能得以充分利用，達到對環境友善的效果。共享平台的發展大致可以分成兩種類型，第一種是單純提供資產共享的模式，例如提供房間共享的 Airbnb、單車共享的 Youbike 與電動機車共享的 WeMo，為單價昂貴的資產提供「分時租賃」的服務，讓資產在閒置時間時有機會可以滿足有需求的人，具有弱化資產擁有權，強化資產使用權的效果。另一種共享平台為提供服務的模式，例如提供私家車搭乘服務的知名平台 Uber，或是提供眾包運送服務的 UberEATS 與美團外賣，為有需求方與供給方提供一個媒合的平台，以「群眾外包」(Crowdsourcing) 的方式，讓非特定的志願者或兼職者有機會可以承接需求方發布的工作並完成。

共享經濟得以蓬勃發展的原因除了網路的普及率，還有其它的重要因素刺激所致。例如人口成長與都市化的影響，都會區的人們生活在擁擠的環境，使得為交通工具安排停放空間必須付出金錢與時間成本，這讓有通勤需求的人們轉向大眾交通運輸工具，然而大眾交通運輸工具無法滿足的地區，體積輕巧、使用便利的 WeMo 電動機車共享平台就得以應用於「城市最後一哩路」的交通手段。而第三方支付的交易方式，使得群眾外包系統中非特定的需求者與志願者願意互相信任，確保交易可以順利完成。其中眾包運送平台提供的運送服務性質單純，具有標準化的指標評量服務的好壞，例如是否送達、是否在時限內。餐點外送屬於城市運送需求的一種，具有強烈的需求時間不平均特性，都會區在用餐時段的需求爆炸讓眾包運送平台的交易量大增，台灣營運中眾包運送平台包括 Ubereats、honestbee、foodpanda，大陸知名的眾包運送平台包含美團外賣與滴滴外賣，競爭非常激烈。

企業在共享經濟的角色，將提供使資產使用率更高、使服務更有效率的管理，而共享平台最大的問題就是使用者的不確定性，可能破壞系統的平衡或無法掌握，因此本研究將首先探討無樁式載具共享系統的動態運補以提升系統自身營運效益，再將無樁式載具共享系統的路徑資料庫應用至眾包運送平台，達到互補互利效果。

1.2 研究動機

電動機車共享系統，以 WEMO 為例，屬於無樁式載具共享系統。有樁式載具共享系統，例如 Youbike，透過在城市各處設立租借站，提供使用者異地租還的服務。而無樁式載具共享系統則取消了租借站的限制，使用者不需要到特定站點進行租借，藉由 GPS 定位系統整合即時掌握車輛的狀態，選擇租借距離最近的機車，使用完畢後將機車停放在城市中合法的停車格內即完成還車，達到「隨借隨停」的目標。無樁式載具共享系統最大的好處在於，使用者不受到站點的限制，可以自由決定起訖點的位置，解決城市最初與最後一哩路的交通問題。也因為隨借隨停的特性，因此無樁式的載具共享系統使用紀錄非常貼近使用者真實的需求。

無樁式載具共享系統給了使用者最大的自由，但使用者行動的不確定性同時也帶來了載具共享系統挑戰，如何在不限制使用者借還行為的前提下，滿足不確定的使用者需求成為棘手的問題。對使用者而言，載具共享系統最大的兩個問題為「無車可借」與「無位可還」。在無樁式的無樁式載具共享系統中，無位可還的不便程度較有樁式載具共享系統來得低，原因在於歸還車輛不受租借站容量的限制，若使用者在預定停靠處找不到合法停車位，可自行轉向鄰近區域的停車位。而無車可借卻可能使無樁式載具共享系統流失顧客，並且降低了顧客對系統的滿意度。對載具共享系統進行有系統的车辆調度工作，有助於減少無車可借與無位可還的問題。考量電動機車的體積大小，比起使用有容量上限的卡車進行點到點之間的車輛運補，更理想的方式是使用人力進行運補，使電動機車均勻地散布在城市中，並且避免搬動電動機車集中至卡車的作業時間。

除了進行提升系統服務品質的車輛運補外，透過無樁式載具共享系統的使用紀錄掌握了用戶習慣，本研究將加入群眾外包的運送服務，讓無樁式載具共享系統的數據輔助群眾外包系統，配合無樁式載具共享系統的歷史紀錄，可以篩選出具有和運送需求相近使用紀錄的用戶進行推送，減少完成任務的成本，增加用戶接受任務的意願，同時也避免了垃圾訊息，幫用戶簡化繁雜的搜尋任務作業時間，讓用戶視完成運送任務為舉手之勞。本研究將以無樁式載具共享系統的使用紀錄為基礎，進行無樁式載具共享系統中車輛運補任務規劃與運送任務推送決策。

1.3 研究目的

本研究提出結合無樁式載具共享系統與群眾外包系統之整合模式。針對在城市各處分散不均的車輛，本研究提出轉化虛擬站點概念之無樁式載具共享系統車輛運補決策模型，標準化運補作業指示，增加使用者對系統的滿意度，有效提升車輛使用率。本研究更結合無樁式載具共享系統與群眾外包概念之運送服務，讓無樁式載具共享系統的數據輔助群眾外包系統的運送任務精準推送，避免推送成為垃圾訊息並增加接受任務的意願，而群眾使用無樁式載具共享系統進行運送，可同時增加載具使用率與用戶對系統的黏著性。研究目的可分成以下主題：

1. 無樁式載具共享系統轉化

比起有樁式載具共享系統，無樁式載具共享系統的挑戰在於閒置車輛並非聚集停放在固定站點，而是散落在城市各處。本研究將提出虛擬站點之設置方法，將無樁式模式進行轉化，有利於運補任務決策模型的建立。

2. 滾動式動態運補決策模型

由於日間載具共享系統的營運時時影響系統的車輛平衡，因此本研究將提出滾動式的動態運補決策模型，將營運期間分為多期，每期皆進行運補決策，直至營運結束。每次決策皆更新實際的車輛與人員參數，讓模型的輸入貼近真實分布情況，且模型放大規劃時間的長度，使決策的眼光放遠，考慮運補工作的連續性，有利於系統流暢的運作。

3. 路徑資料分群

從無樁式載具共享系統蒐集的使用紀錄非常貼近真實的使用者需求，因此有助於篩選適合的人選進行任務推送。然而，使用紀錄的資料量非常龐大，本研究設計一路徑分群演算法，定義路徑間的相異度，並結合數學規劃模型，以分群結果的代表路徑作為使用者習慣路徑。

4. 運送任務的配對

本研究設定群眾外包提供運送服務，必須將貨物從需求起點運送至需求終點。考量系統與群眾兩方的需求：系統方希望多多推送任務，讓任務的可見度增加，而群眾不希望收到過多的任務，造成資訊疲勞，本研究定義一配適分數作為運送任務與群眾配對的基準，提出一數學規劃模型滿足推送的限制，最大化系統配適分數值，求解推送任務的決策。

1.4 論文架構

第二章為文獻回顧，將分別針對載具共享系統、眾包運送與路徑分群的相關文獻進行探討。第三章將探討無樁式載具共享系統動態運補研究，解決載具共享系統車輛配置引發使用者租借不便的問題。包含無樁式共享系統的轉化，建構無樁式共享系統的地理分區，使運補工作指示具有清楚地理資訊；接著提出最適車數數學模式與兩種車輛運補數學模式。最適車數數學模式將決策動態運補的車輛初始配置，而最適車數運補數學模式與最大需求滿足運補數學模式將應用滾動模式決策員工的運補工作，最後進行數值測試。第四章則探討眾包運送平台任務推送研究，應用無樁式載具共享系統後台紀錄之使用者路徑資料庫，篩選對運送任務媒合度高之使用者進行任務推送。首先針對使用者的路徑使用紀錄提出路徑分群演算法的前處理，接著建構數學模型決策運送任務應推送給哪些群眾，使推送的效益極大化。最後進行路徑分群與任務推送模型的數值測試。第五章將總結本研究的結論與未來研究建議。

第二章 文獻回顧

2.1 載具共享系統之相關文獻

載具共享系統的發展日漸蓬勃，因此，現今載具共享系統的發展非常多元，載具的種類包括汽車、摩托車與單車，而營運的型態可分為「有樁式」與「無樁式」。有樁式載具共享系統中交通工具必須停放在有限的規劃站點區域，例如 Youbike。無樁式載具共享系統則沒有規劃的站點區域，而是允許使用者停放在任何一個合格的停車位即可，例如 Obike、WeMo。管理工作依時間則可分為「靜態運補」與「動態運補」。以下將分別介紹靜態運補與動態運補的相關文獻。

2.1.1 靜態運補

靜態運補是指在載具共享系統的使用率為零時（或幾乎為零），例如深夜時分，在空間中進行車輛重新分配工作，目的是使系統在隔日開始營運時可以達到最佳的車輛初始配置。

Raviv, Tzur, and Forma (2013)對有樁式自行車共享系統提出兩種建模方式之數學規劃模型，分別從節線網路與時空網路角度，決策自行車在夜間的靜態運補工作，包含每個站點的存貨數量如何重新配置，與每輛運補卡車的巡迴路徑。給定站點與自行車的容量上限，模型的目標為最小化最終配置所造成的使用短缺次數與運補車隊的營運成本。

廖敏婷(2012)在有樁式自行車共享系統，考量運補卡車的繞行長度、工作量與運補卡車間工作量的差異，決策運補卡車靜態運補的作業內容。廖敏婷(2012)提出粒子群演算法加速此問題求解速度，使此模型可解決規模較大的問題。

2.1.2 動態運補

載具共享系統的營運期間，車輛的移動變化可能讓使用者遭遇的「無車可借」與「無位可還」，為了解決這兩大問題而在營運期間進行的車輛調度工作則為動態運補，由於營運期間車輛的位置時時刻刻受到使用者借還行為影響，使用者行為的不確定性增加了動態運補的困難性，比起靜態運補更具挑戰性。

Kek, Cheu, Meng, and Fung (2009)針對有樁式汽車共享系統，建立三階段的車輛運補決策模擬系統。第一階段以時空網路建立數學規劃模型，目標為最小化營運成本，決策多位員工的配置與行動指示，員工行動包括在站點等待、在站點進行保養、在站點間獨自移動及在站點間進行車輛的運補。第二階段將對第一階段所決策之員工行動指示進行過濾，使系統的運作更有效率。第三階段將使用 Kek, Cheu, and Chor (2006)提出之模型進行模擬。在案例分析中，此車輛運補決策模擬系統展現了優秀的性能。

張立蓁(2010)在有樁式自行車共享系統中，提出兩階段模式決策卡車的動態運補作業。第一階段放寬卡車數量的限制，決策下一期各站點需要的運補卡車數，第二階段再指派有限卡車進行運補，目標為最小化無法滿足的運補需求。洪菁蓬(2011)更提出三種租借歷史資訊情境，分別為已知使用者起訖點的一般車輛運補配置問題、已知使用者起訖租借站點之固定租借者行程之車輛運補問題、已知使用者起訖租借站點且考量流量具比例分配之車輛運補問題，提出最小成本多元商品網路流量模式進行求解。由於租借站數量的增加將急速增加問題的規模，因此即使加上粒子群演算法等啟發式演算法，此兩研究的整數規劃模式僅能處理小規模的運補問題(11站、8期、2卡車，洪菁蓬(2011))。

廖敏婷(2012)針對有樁式自行車共享系統的動態運補作業加入暫時人力配置的資源，也就是以群眾外包進行載具共享系統維護的模式。假設招募的群眾人數具有上限，建立一數學規劃模型決策何時招募群眾為系統進行兩種作業，分別為看管超過站點容量上限之超量自行車，與特定起訖站點之自行車騎乘以維持站點間的車輛均勻分

佈。

梁瑜庭(2013)結合有樁式電動機車共享系統，探討充電站與電池交換站兩種不同的電池續航策略，並考量電池的充耗電速率，依據起訖借還需求的比例關係，建立混整數規劃模式，進行期初車輛配置的決策，並且設計兩個粒子演算法以加速求解效率。

Weikl and Bogenberger (2013)在無樁式汽車共享系統中，對使用者歷史數據進行分析後，提出一目標為最小化營運成本的數學規劃模型，提供員工運補的決策。數據分析過程首先將使用數據依不同的時段與行政區分開處理，再對特定時段與行政區設定下的每日使用數據進行主成份分析，以主成份的結果進行 K-means 分群，觀察使用者如何隨時間流動，提供決策運補工作時的參考值。

Jorge, Correia, and Barnhart (2014)對提供單程租還的有樁式汽車共享系統提出車輛運補的數學規劃模型，使用時空網路建構，假設已知一日顧客的需求，目標式為最大化汽車共享系統一日的營運利潤，決策站點容量及車輛運補需求。並使用兩種運補策略模擬後進行比較，差別在於判別站點屬於需求站點或供給站點的定義不同，案例分析的模擬結果顯示對汽車共享系統進行運補，可增加營運利潤，而數學規劃模型求解可視為汽車共享系統加上運補作業的營運利潤上限。

Nourinejad, Zhu, Bahrami, and Roorda (2015)在提供單程租還的有樁式汽車共享系統中，假設站點無容量限制及已知一日的顧客時地需求，目標為最小化整體營運成本，提供員工如何進行運補與滿足該運補作業的車隊最少車數與最少員工數。Nourinejad et al. (2015) 將其視為兩個多旅行銷售員問題(Multi Traveling Salesman Problem, m-TSP)組成並進行建模，設計一分解演算法求解之。m-TSP 問題是旅行銷售員問題(Traveling Salesman Problem, TSP)的延伸，給定數個城市與多個銷售員，限制每個城市必須恰好被一位銷售員經過一次，目標為繞行成本最小化。

周佰賢(2015)對有樁式單車共享系統提供兩種站點的分群方法，分別為 K-means 分群法與預測趨勢之分群數學模式，劃分運補卡車的責任區域，降低運補模型求解維度。基於分群結果，提出三種情境之運補模式，包含直接由運補卡車進行運補的原始

運補模式，所有站點皆增設臨停區並派遣人力的運補模式，與進行臨停區選址後派遣人力的運補模式，三種運補模式皆以最大化使用者的租還需求車次為目標。

侯貞泰(2016)對於有樁式單車共享系統的車輛運補工作，首先提出最適車數的整數規劃模式作為目標，再進行群眾運補數學模式決策群眾外包需求，最後建立一模擬系統進行測試。本研究將以侯貞泰(2016)的模型為基礎，提出單純考量車輛起訖租借紀錄之最適車數模型，應用至無樁式載具共享系統，雇用員工進行車輛運補工作的模式。

Caggiani, Camporeale, Ottomanelli, and Szeto (2018)對無樁式單車共享系統進行統整性的單車重新分配工作設計。對使用者歷史紀錄使用時空分群方法進行分群，再以類神經網路進行使用者行動的預測，以此做為運補決策數學規劃模型之參數輸入。Caggiani et al. (2018)的運補決策數學規劃模型在營運期間會決策多次，相隔固定時間。模型決策目標為最小化該期間服務水準不足量與流失顧客數，決策單車在該期間，自時空分群之供給群流至需求群的移動量。其中每個時空分群具有一給定的單車量，作為服務水準基準。

表 2.1 載具共享系統動態運補相關文獻比較表

動態運補相關文獻	載具類型			營運模式		求解設計
	單車	電動 摩托車	汽車	有樁式	無樁式	
Kek, Cheu, Meng, and Fung (2009)			V	V		-
張立蓁(2010)	V			V		粒子群演算法
洪菁蓬(2011)	V			V		粒子群演算法
廖敏婷(2012)	V			V		-
梁瑜庭(2013)		V		V		粒子群演算法
Weikl and Bogenberger (2013)			V		V	-
Jorge, Correia, and Barnhart (2014)			V	V		-
Nourinejad, Zhu, Bahrami, and Roorda (2015)			V	V		分解演算法
周佰賢(2015)	V			V		-
侯貞泰(2016)	V			V		-
Caggiani, Camporeale, Ottomanelli, and Szeto (2018)	V				V	基因演算法
本研究		V			V	滾動式求解

2.2 眾包運送之相關文獻

群眾外包是透過網路平台的媒介，召集群眾的力量完成目標的手段，例如維基百科建立群眾的協作編撰平台建構龐大的網路百科工具書，Uber 建立代僱駕駛平台連結載客的供需兩端。其中，眾包運送（Crowdsourced shipping）是常見的群眾外包應用之一，應用群眾物流（Crowd Logistics）提供運送的服務，優點在於較專業性的快遞公司低廉的價格與靈活的服務範疇與應用，可做為組織勞動力的來源，然而缺點在於群眾的不確定性將降低服務的可靠度，成為近年來研究的重點。

Sampaio Oliveira, Savelsbergh, Veelenturf, and vanWoensel (2017)對運用眾包運送相關研究主題進行分析，眾包運送是一種以群眾外包的方式，讓有意願的群眾提供運送服務的模式。隨著都市化的影響，都會區的人口愈來愈多，再加上電子商務的興起，即時快遞的需求激增，需要創新的解決方案以符合成本效益、友善社會環境原則，而人口密集的城正是群眾物流得以蓬勃發展的絕佳環境。但運用群眾物流帶來機會的同

時，也帶來了挑戰。Sampaio Oliveira et al. (2017)指出，如何整合利用現有流量並保持一定服務水準，如何使群眾願意加入眾包運送的行列，群眾物流的規模以及動態的應用將會是運用群眾物流的重要議題。例如眾包運送服務，同時雇用全職快遞員可以增加系統的可靠性與服務質量，而已恰好排定與運送需求符合的行程之群眾具有較低的邊際成本，因此系統能以較低廉的價格滿足快遞需求。

Cheng and Chen (2018)對眾包運送的移動服務系統建立一數學規劃模型。系統中有多位可推送任務的群眾，已知群眾各自擁有多條習慣經過之路徑，路徑具有被選擇實現的機率，且群眾可選擇拒絕完成被推送的任務。網路中有多個在節點上的眾包運送任務，給定在節點的作業時間與任務的報酬，群眾必須繞路至任務節點進行服務。模型決策每個任務應推送給那些群眾，使得任務無法被完成的機率最小，也就是最大化所有任務被完成的機率。數值結果顯示考量群眾選擇路線的機率性讓結果比確定性模型表現更佳。

Yildiz and Savelsbergh (2017)對餐點外送平台的途程規劃問題建立一數學規劃模型，並設計同步列行生成法(simultaneous column and row generation method)進行求解。假設已知餐點外送的需求時間窗，群眾的排班時段與起始位置，模型考量的服務指標包含給最小化給群眾的報酬、最小化顧客等待時間與最小化餐點等待間，藉由不同的目標式考量不同的系統服務表現，決策群眾進行的外送行程。其中，限制每個餐點外送需求皆被滿足，並允許群眾在一次途程運送多個餐點，且群眾的移動時間會隨時間改變。群眾具有一最低工資保證，若運送量超過標準，則論件計價。數值分析結果顯示，讓群眾在一次途程進行運送多個餐點無法顯著降低系統的營運成本，而妥善設定的最低工資保證並不會使系統營運成本顯著上升。

Yildiz and Savelsbergh (2018)針對餐點外送平台之中，面對顧客需求與群眾外包的隨機性，考量服務收費、支付給群眾的成本與服務品質，建立最大化餐點外送平台收益數學模型，分析參數的影響力。已知餐廳以其為圓心，服務範圍在半徑以內，給定顧客需求與群眾的到達率，定義群眾出發的原點皆在餐廳，每次完成餐點外送後可

回到餐廳繼續服務。假設群眾接受指派的機率與距離有關，討論是否額外雇用費用更高的全職員工協助外送，以及外送區域的需求強度與距離影響。該研究指出，如果外送區域與餐廳相距不遠，且需求強度高，在提供眾包運送服務的餐點外送平台系統中額外雇用全職員工可以增加系統收益，且需求強度對收益具有顯著影響。

表 2.2 眾包運送相關文獻比較表

眾包運送相關文獻	數學模型	群眾機率設計
Sampaio Oliveira, Savelsbergh, Veelenturf, and vanWoensel (2017)	-	-
Yildiz and Savelsbergh (2017)	V	給定參數
Yildiz and Savelsbergh (2018)	V	給定參數
Cheng and Chen (2018)	V	給定參數
本研究	V	無樁式載具共享系統的路徑數據分群結果

2.3 路徑分群相關文獻

為了讓任務配對的決策過程可以掌握機車共享系統使用者們的習慣，必須對路徑的歷史數據做分群，以簡化後的路徑分群資訊作為基準，提供精準的任務推送。將路徑分群的目標是最小化群內差異同時也最大化群間差異，此舉可將彼此差異不大的路徑歸為同一群，視為相同行為，在龐大路徑數據中找出規律。

在過去的文獻中，Nanni and Pedreschi (2006)應用 OPTICS 演算法於路徑分群，給定一路徑相距界線參數與路徑數據，即可得到具有時間與空間維度資訊的路徑分群結果。OPTICS 是 DBSCAN 演算法的分支，DBSCAN 演算法以密度作為分群指標，優點在於不易受雜訊點與離群值的影響，具有穩健性。而 OPTICS 演算法提出了可達圖 (reachability plot) 概念，視覺化數據的分群結構。

而 Lee and Han (2007)參考 DBSCAN 演算法，提出以密度作為分群指標的 TRACCLUS 演算法。TRACCLUS 演算法可分為三個階段，首先將原始的路徑數據進行平滑化，簡化路徑的分段。接著將路徑段進行分群，最後以平均座標的概念為每個路徑段群產生代表性路徑。和 Nanni and Pedreschi (2006)不同之處在於路徑分群結果並

不具有時間資訊，且分群的對象為路徑的分段，而非完整的路徑。

表 2.3 路徑分群相關文獻比較表

路徑分群相關文獻	分群指標	針對路徑原始數據分群	代表性路徑
Nanni and Pedreschi (2006)	密度	V	-
Lee and Han (2007)	密度	-	V
本研究	路徑段連線長度平均值	V	V

2.4 小結

從本章的文獻回顧可以發現，在載具共享系統中，針對無樁式載具共享系統之車輛運補決策研究稀少，有樁式載具共享系統的研究相對豐沛，因此本研究將參考有樁式載具共享系統的建模手法，為無樁式載具共享系統提供創新的車輛動態運補決策模式。而眾包運送的相關研究中，如何善用群眾物流為研究的議題之一，本研究將利用無樁式載具共享系統的使用紀錄，配合設計路徑分群演算法的前處理，輔助決策群眾物流與運送任務的配對。

第三章 無樁式載具共享系統動態運補研究

本章將探討無樁式載具共享系統中車輛的動態運補進行探討。3.1 節將提出虛擬站點的概念，將無樁式載具共享系統轉化。3.2 節基於使用者的歷史借還紀錄提出最適車數數學模型，決策車輛初始配置。3.3 節針對 3.2 節的最適車數為目標建構車輛運補數學模型。3.4 節針對歷史借還紀錄為目標建構車輛運補數學模型。3.5 節為數值測試結果。

3.1 虛擬站點轉化

無樁式載具共享系統沒有站點的概念，無法確實指出運補工作與需求變動的目標範圍，因此不利於管理。本研究提出以等距的方式，首先將目標區域分割為數個等大的矩形，視為虛擬站點，如圖 3.1 所示。再對虛擬站點進行篩選，將需求微小的站點予以刪除，如圖 3.2 所示。提供固定等大的虛擬站點系統可整合相鄰區域的車子位置資訊，且統一固定的站點位置有利於指派運送任務時提供明確的指示。



圖 3.1 原始虛擬站點

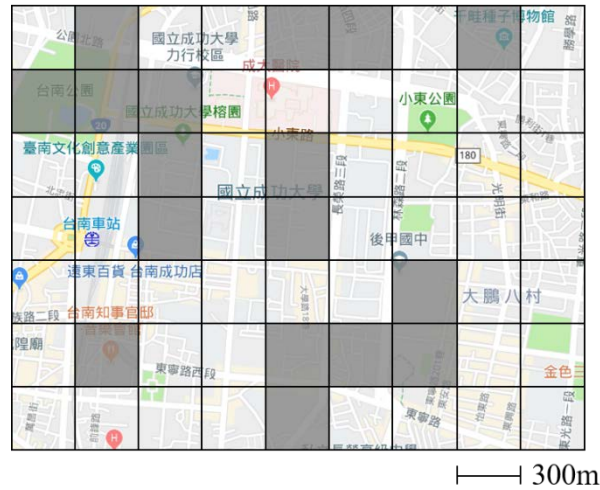


圖 3.2 需求大於下限之虛擬站點

3.2 最適車數數學模式

3.2.1 問題描述

在載具共享系統中，為了避免使用者在借還時遇到困難，經常參考歷史使用數據，在事前進行運補工作，排除使用者不便的情形。給定載具共享系統起訖使用紀錄，本研究將提出以最小化借還需求缺量變數為目標，即最大化滿足借還需求之數學模式，決策每日虛擬站點在不同時刻的最適配置車數，並以初始時刻的車數決策量作為 3.3 節之最適車數運補數學模式與 3.4 節之最大需求滿足運補數學模式的初始車輛配置。

本研究與侯貞泰 (2016)提出的最適車數數學模式不同之處在於，侯貞泰 (2016)提出的數學模式應用於有樁式載具共享系統，其站點有容量上限的限制，且模型的參考為各站點借還數量。本研究提出的最適車數數學模式針對無樁式載具共享系統轉化的虛擬站點系統，其站點無硬性的容量上限，且模型參考使用者的起訖使用紀錄，將更好的掌握使用者行為。

3.2.2 問題假設

1. 本研究定義兩時間表示方式：「時刻」表示時間軸上的一點，「時期」表示時間軸上介於兩時刻中的期間。如圖 3.3 所示，時期 t 介於時刻 $t-1$ 與時刻 t 之間。而時期的長度為事先給定之參數，例如 30 分鐘。

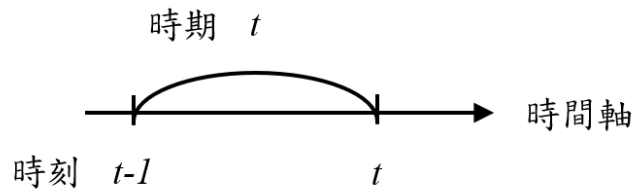


圖 3.3 時間表達定義示意圖

2. 假設使用者的借還作業發生在時刻 $[t-1, t)$ 的區間內，則歸屬於時期 t 。
3. 各時期各虛擬站點的起訖使用紀錄已知。

3.2.3 數學模式

以下定義符號：

集合

N 所有虛擬站點的集合

T 進行動態運補之營運時間集合

參數

V 共享系統的總車數

B'_{ij} 在時期 t 車輛從站點 i 移動至站點 j 的歷史借還量， $i, j \in N$ ， $t \in T$

ε 一個極小的數字

決策變數

\bar{I}_i^t 站點 i 在時刻 t 的最適配置車輛數， $i \in N$ ， $t \in T$

b_{ij}^t 在時期 t 車輛從站點 i 移動至站點 j 的需求滿足量， $i, j \in N$ ， $t \in T$

x_{ij}^t 在時期 t 車輛從站點 i 移動至站點 j 的運補工作量， $i, j \in N$ ， $t \in T$

U_{ij}^t 虛擬站點 i 在模型規劃時刻 t 的歷史借還超量， $i, j \in N$ ， $t \in T \setminus \{r-1\}$

L_{ij}^t 虛擬站點 i 在模型規劃時刻 t 的歷史借還缺量， $i, j \in N$ ， $t \in T \setminus \{r-1\}$

數學模式

$$\min \sum_{t \in T} \sum_{i \in N} \sum_{j \in N} (U_{ij}^t + L_{ij}^t) + \varepsilon \sum_{t \in T} \sum_{i \in N} \sum_{j \in N} x_{ij}^t \quad (3.1)$$

$$\bar{I}_i^t = \bar{I}_i^{t-1} - \sum_{j \in N} b_{ij}^t + \sum_{j \in N} b_{ji}^t - \sum_{j \in N} x_{ij}^t + \sum_{j \in N} x_{ji}^t \quad \forall i \in N, t \in T \quad (3.2)$$

$$b_{ij}^t = B_{ij}^t + U_{ij}^t - L_{ij}^t \quad \forall i, j \in N, t \in T \quad (3.3)$$

$$\bar{I}_i^{t-1} - \sum_{j \in N} b_{ij}^t \geq \sum_{j \in N} x_{ij}^t \quad \forall i \in N, t \in T \quad (3.4)$$

$$\sum_{i \in N} \bar{I}_i^0 = V \quad (3.5)$$

$$\bar{I}_i^t, b_{ij}^t, x_{ij}^t \geq 0 \quad \forall i \in N, t \in T \quad (3.6)$$

式 3.1 為目標式，最小化所有超量變數與缺量變數，並加上次要的運補工作懲罰項，避免產生多餘的運補工作。式 3.2 為站點車輛守恆限制式，給定在時刻 $t-1$ 站點 i 的車量，加上時期 t 的站點 i 的借還需求滿足量與運補工作決策量，即為時刻 t 站點 i 的車量。式 3.3 定義起訖借還需求滿足量為歷史借還量加上超量變數減去缺量變數。式 3.4 限制站點的車量必須大於該站點下個時期之借還需求滿足量與運車工作決策量總和。式 3.5 限制所有站點在時刻 0 的總車數等於給定參數。式 3.6 限制決策變數必須大於零。

3.3 最適車數運補數學模式

3.2 節提出最適車數數學模型決策之最適車數包含營運期間的各個時刻，可作為運補工作基準。本節將建構目標為最小化最適車數差距之動態車輛運補數學模式，決策如何指派員工進行車輛運補工作。

3.3.1 問題描述

以 3.2 節最適車數數學模型決策之各時刻最適車數為目標，本節提出之最適車數運補數學模式採用滾動式決策，在載具共享系統的營運期間，每期依現況調整車輛配置與員工配置的參數，再執行此數學模型，此數學模型的輸出為各位員工下一期的執行動作，包括員工是否駕車移動、員工是否獨自移動或是員工是否留在原地，提供員工明確的指示。

3.3.2 問題假設

1. 給定員工數量與車隊數量，且員工與車隊的位置可供追蹤。
2. 載具共享系統在各時期各虛擬站點的使用者借還量數據已知。
3. 虛擬站點不具有容量上限。
4. 虛擬站點之間的旅行時間皆小於一倍時期的設定時間長度。

3.3.3 數學模式

集合

- N 所有虛擬站點的集合
- T 進行動態運補之營運時間集合

參數

- B_{ij}^t 在時期 t 車輛從虛擬站點 i 至虛擬站點 j 的歷史借還量， $i, j \in N$ ， $t \in T$
- \bar{I}_i^t 虛擬站點 i 在時刻 t 的最佳配置車輛數， $i \in N$ ， $t \in T$
- I_i^{r-1} 虛擬站點 i 在模型規劃的起始時刻的車輛數， $i \in N$
- Q_i^{r-1} 虛擬站點 i 在模型規劃的起始時刻的員工數， $i \in N$
- r 模型執行的次數
- h 模型規劃的時期長度
- ε 為一極小的數字
- V 車隊數量， $\sum_{i \in N} I_i^{r-1} = V$
- P 員工數量， $\sum_{i \in N} Q_i^{r-1} = P$

決策變數

- I_i^t 虛擬站點 i 在模型規劃的時刻 t 之車輛數， $i \in N$ ， $t \in T \setminus \{r-1\}$
- Q_i^t 虛擬站點 i 在模型規劃的時刻 t 之員工數， $i \in N$ ， $t \in T \setminus \{r-1\}$
- b_{ij}^t 時期 t 車輛從站點 i 至站點 j 的借還需求滿足量， $i, j \in N, t \in T \setminus \{r-1\}$
- x_{ij}^t 在模型規劃的時期 t 指派員工從虛擬站點 i 駕車至虛擬站點 j 的車輛數，
 $i, j \in N$ ， $t \in T \setminus \{r-1\}$
- y_{ij}^t 在模型規劃的時期 t 指派員工從虛擬站點 i 獨自移動至虛擬站點 j 的人數，

$$i, j \in N, t \in T \setminus \{r-1\}$$

U_{ij}^t 虛擬站點 i 在模型規劃時刻 t 的歷史借還超量, $i, j \in N, t \in T \setminus \{r-1\}$

L_{ij}^t 虛擬站點 i 在模型規劃時刻 t 的歷史借還缺量, $i, j \in N, t \in T \setminus \{r-1\}$

\tilde{U}_i^t 虛擬站點 i 在模型規劃時刻 t 的最適配置車輛數超量, $i \in N, t \in T \setminus \{r-1\}$

\tilde{L}_i^t 虛擬站點 i 在模型規劃時刻 t 的最適配置車輛數缺量, $i \in N, t \in T \setminus \{r-1\}$

滾動式決策

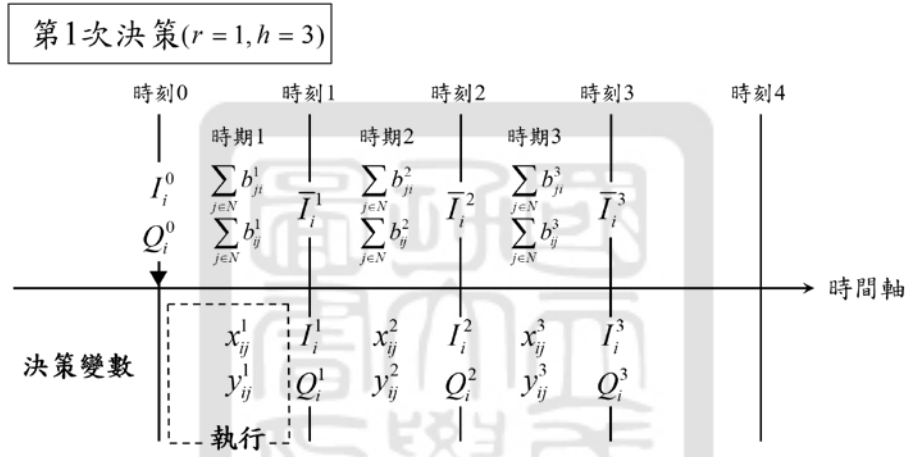


圖 3.4 第1次員工運補決策示意圖

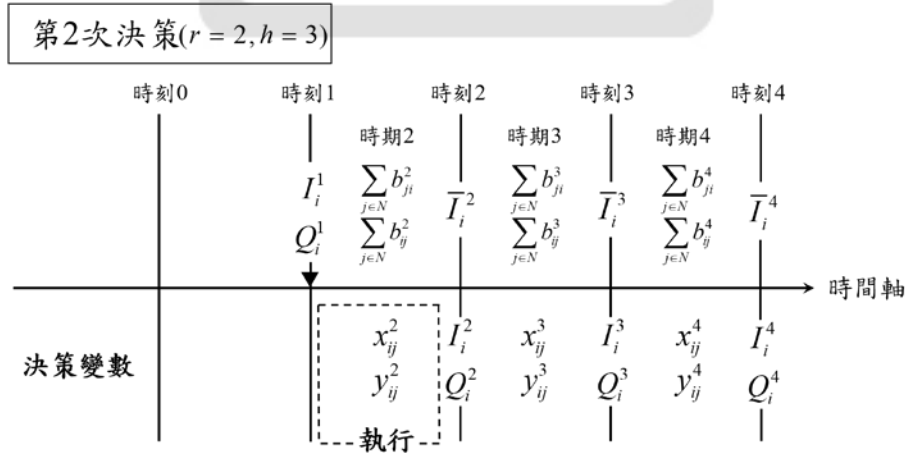


圖 3.5 第2次員工運補決策示意圖

圖 3.4 與 3.5 為系統初始連續兩期員工運補任務決策示意圖。在時刻 0 進行第 1 次決策，模型往後考量連續 3 期（時期 1 至時期 3），包括起始時刻的車數與員工

數、使用者借還量與 3.2 節決策之最佳配置車數。第 1 次決策輸出往後 3 期（時期 1 至時期 3）的員工駕車與移動決策，與時刻 1 至時刻 3 的虛擬站點車數、員工數，實際執行第 1 次決策中第 1 期的員工指派後，時間軸位於時刻 1，更新虛擬站點車數與員工數，進行第 2 次決策。模型同樣往後考量連續 3 期（時期 2 至時期 4），並決策往後 3 期（時期 2 至時期 4）的員工駕車與移動決策、時刻 2 至時刻 4 的虛擬站點車數、員工數，並執行第 2 次決策中第 2 期的員工指派。

數學模式

$$\min \sum_{i \in N} \sum_{j \in N} \sum_{t=r}^{r+h-1} (U_{ij}^t + L_{ij}^t + \tilde{U}_i^t + \tilde{L}_i^t) + \varepsilon \sum_{i \in N} \sum_{j \in N} \sum_{t=r}^{r+h-1} (x_{ij}^t + y_{ij}^t) \quad (3.7)$$

$$I_i^t = I_i^{t-1} + \sum_{j \in N} b_{ji}^t - \sum_{j \in N} b_{ij}^t + \sum_{j \in N} x_{ji}^t - \sum_{j \in N} x_{ij}^t \quad \forall i \in N; t = r, \dots, r+h-1 \quad (3.8)$$

$$Q_i^t = Q_i^{t-1} + \sum_{j \in N} x_{ji}^t - \sum_{j \in N} x_{ij}^t + \sum_{j \in N} y_{ji}^t - \sum_{j \in N} y_{ij}^t \quad \forall i \in N; t = r, \dots, r+h-1 \quad (3.9)$$

$$b_{ij}^t = B_{ij}^t + U_{ij}^t - L_{ij}^t \quad \forall i, j \in N; t = r, \dots, r+h-1 \quad (3.10)$$

$$I_i^t - \sum_{j \in N} b_{ij}^t \geq \sum_{j \in N} x_{ji}^t \quad \forall i \in N; t = r, \dots, r+h-1 \quad (3.11)$$

$$Q_i^t - \sum_{j \in N} x_{ij}^t \geq \sum_{j \in N} y_{ji}^t \quad \forall i \in N; t = r, \dots, r+h-1 \quad (3.12)$$

$$I_i^t - \bar{I}_i^t = \tilde{U}_i^t - \tilde{L}_i^t \quad \forall i \in N; t = r, \dots, r+h-1 \quad (3.13)$$

$$\sum_i I_i^t = V \quad \forall t = r, \dots, r+h-1 \quad (3.14)$$

$$I_i^t, Q_i^t, b_{ij}^t, x_{ij}^t, y_{ij}^t, U_{ij}^t, L_{ij}^t, \tilde{U}_i^t, \tilde{L}_i^t \geq 0 \quad (3.15)$$

式 3.7 為目標式，目標式前項最小化模型規劃期間所有的缺量變數與超量變數，即最小化規劃期間無法準確滿足需求的差異量；而目標式後項最小化規劃期間員工的移動成本，避免多餘的員工移動。式 3.8 為車輛守恆限制式，給定時刻 $t-1$ 的站點 i 的車量，加上時期 t 的站點 i 借還需求滿足量與運補工作決策量，即為時刻 t 站點 i 的車量。式 3.9 為員工數守恆限制式，給定在時刻 $t-1$ 之站點 i 的員工數，加上時期 t 的

站點 i 員工駕車移動決策量與員工獨自移動決策量，即為時刻 t 站點 i 的員工數。式 3.10 定義借還需求滿足量為歷史借還量加上超量變數減去缺量變數。式 3.11 限制站點車輛數扣除借出需求滿足量後必須大於下期的員工駕車移動決策量。式 3.12 限制站點員工數扣除員工駕車決策量後必須大於下期的員工獨自移動決策量。式 3.13 定義各時刻虛擬站點的車輛決策數與最佳配置車數的差距，等於超量變數減去缺量變數。式 3.14 限制各時刻的車輛決策量總和等於系統車隊數量。式 3.15 定義決策變數必須大於或等於零。

3.4 最大需求滿足運補數學模式

本節將提出最大需求滿足量為目標之車輛運補模式，參考使用者的起訖使用紀錄，以最小化歷史借還紀錄差量為目標，也就是最大化起訖使用需求滿足量，同樣採用滾動式手法建構車輛運補模式，決策包含每期員工與車輛的移動。

3.4.1 問題描述

給定載具共享系統的起訖使用需求量，本節提出之車輛運補數學模式以最大化起訖使用需求滿足量為目標，同樣使用滾動式決策，在載具共享系統的營運期間，每期可依現況調整車輛配置與員工配置的參數，再執行此數學模型，輸出提供員工明確的指示，包括員工是否駕車移動、員工是否獨自移動或是員工是否留在原地。

3.4.2 問題假設

1. 給定員工數量與車隊數量，且員工與車隊的位置可供追蹤。
2. 在各時期的載具起訖使用紀錄已知。
3. 虛擬站點不具有容量上限。
4. 虛擬站點之間的旅行時間皆小於一倍時期的設定時間長度。

3.4.3 數學模式

集合

N 所有虛擬站點的集合

T 進行動態運補之營運時間集合

參數

B_{ij}^t 在時期 t 車輛從站點 i 移動至站點 j 的歷史借還量, $i, j \in N$, $t \in T$

I_i^{r-1} 站點 i 在模型規劃的起始時刻的車輛數, $i \in N$

Q_i^{r-1} 站點 i 在模型規劃的起始時刻的員工數, $i \in N$

r 模型執行的次數

h 模型規劃的時期長度

ε 為一極小的數字

V 車隊數量, $\sum_{i \in N} I_i^{r-1} = V$

P 員工數量, $\sum_{i \in N} Q_i^{r-1} = P$

決策變數

I_i^t 站點 i 在時刻 t 之車輛數, $i \in N$, $t \in T \setminus \{r-1\}$

Q_i^t 站點 i 在時刻 t 之員工數, $i \in N$, $t \in T \setminus \{r-1\}$

b_{ij}^t 在時期 t 車輛從站點 i 移動至站點 j 的需求滿足量, $i, j \in N, t = r, \dots, r+h-1$

x_{ij}^t 在時期 t 指派員工從站點 i 駕車至站點 j 的車輛數, $i, j \in N, t = r, \dots, r+h-1$

y_{ij}^t 在時期 t 指派員工從站點 i 獨自移動至站點 j 的人數, $i, j \in N, t = r, \dots, r+h-1$

U_{ij}^t 虛擬站點 i 在模型規劃時刻 t 的歷史借還超量， $i, j \in N$ ， $t \in T \setminus \{r-1\}$

L_{ij}^t 虛擬站點 i 在模型規劃時刻 t 的歷史借還缺量， $i, j \in N$ ， $t \in T \setminus \{r-1\}$

數學模式

$$\min \sum_{i \in N} \sum_{j \in N} \sum_{t=r}^{r+h-1} (U_{ij}^t + L_{ij}^t) + \varepsilon \sum_{i \in N} \sum_{j \in N} \sum_{t=r}^{r+h-1} (x_{ij}^t + y_{ij}^t) \quad (3.16)$$

$$I_i^t = I_i^{t-1} + \sum_{j \in N} b_{ji}^t - \sum_{j \in N} b_{ij}^t + \sum_{j \in N} x_{ji}^t - \sum_{j \in N} x_{ij}^t \quad \forall i \in N, t = r, \dots, r+h-1 \quad (3.17)$$

$$Q_i^t = Q_i^{t-1} + \sum_{j \in N} x_{ji}^t - \sum_{j \in N} x_{ij}^t + \sum_{j \in N} y_{ji}^t - \sum_{j \in N} y_{ij}^t \quad \forall i \in N, t = r, \dots, r+h-1 \quad (3.18)$$

$$b_{ij}^t = B_{ij}^t + U_{ij}^t - L_{ij}^t \quad \forall i, j \in N, t = r, \dots, r+h-1 \quad (3.19)$$

$$I_i^{t-1} - \sum_j b_{ij}^t \geq \sum_j x_{ij}^t \quad \forall i, j \in N, t = r, \dots, r+h-1 \quad (3.20)$$

$$Q_i^{t-1} - \sum_j x_{ij}^t \geq \sum_j y_{ij}^t \quad \forall i, j \in N, t = r, \dots, r+h-1 \quad (3.21)$$

$$I_i^t, Q_i^t, b_{ij}^t, x_{ij}^t, y_{ij}^t, U_{ij}^t, L_{ij}^t \geq 0 \quad \forall i \in N, t = r, \dots, r+h-1 \quad (3.22)$$

式 3.16 為目標式，目標式前項表示最小化超量變數與缺量變數，而目標式後項最小化規劃期間員工的移動成本，避免多餘的員工移動。式 3.17 為車輛守恆限制式，給定時刻 $t-1$ 的站點 i 的車量，加上時期 t 的站點 i 借還需求滿足量與運補工作決策量，即為時刻 t 站點 i 的車量。式 3.18 為員工數守恆限制式，給定時刻 $t-1$ 的站點 i 的員工數，加上時期 t 的站點 i 員工駕車移動決策量與員工獨自移動決策量，即為時刻 t 站點 i 的員工數。式 3.19 定義起訖需求滿足量為歷史借還量加上超量變數減去缺量變數。式 3.20 限制站點車量必須大於下期的起訖需求滿足量與運車出站決策量總和。式 3.21 限制站點員工數必須大於下期的員工移動決策數總和。式 3.22 定義決策變數須大於等於零。

3.4 數值測試

測試環境為 Window10 作業系統，搭配 IntelCore i7-6770，3.40GHz*8 處理器與 16GB 記憶體，以 Python3.6.3 為程式語言界面，搭配 Gurobi 8.0.1 版求解數學模式。

3.4.1 最適車數數學模式之數值測試

表 3.1 至 3.3 為最適車數數學模式的數值測試結果。隨機產生起訖需求 (B'_{ij}) 來自期望值為 3 的普瓦松分配，車數 (V) 設定為起訖需求的時刻總和最大值，即 $V = \max \left\{ \sum_{i,j} B'_{ij}, \forall t \right\}$ ，使共享系統的總車數足夠應付系統的起訖需求最大流量值。分別探討站點數 (N) 為 10、50、100 與期數 (T) 為 10、20、25 的設定。

由表 3.1 至 3.3 可以看出，模型可在合理的時間範圍內完成求解，運算時間隨站點數 (N) 與期數 (T) 的增大而有增加的趨勢，而需求缺量比例皆在 3% 以下，且具有運補決策 (x'_{ij}) 數值，這表示當系統設定車數為起訖需求的時刻總和最大值時，配合適量的運補工作，足夠應付所有的起訖借還需求。

3.4.1 節之最適車數數學模式在時刻 0 的車輛配置決策與運補工作決策量的時刻總和最大值 ($\max \left\{ \sum_{i,j} x'_{ij}, \forall t \right\}$) 將作為 3.4.2 節車輛運補模式之數值測試的車輛初始配置與員工數量參數設定。

表 3.1 最適車數數學模式數值測試（站點數 $N=10$ ）

T	V	CPU Time(sec)	ObjVal	需求缺量比例	$\sum_{i,j,t} x_{ij}^t$	$\max \left\{ \sum_{i,j} x_{ij}^t, \forall t \right\}$
				$\sum_{i,j,t} (U_{ij}^t + L_{ij}^t) / \sum_{i,j,t} B_{ij}^t$		
10	286	0.03	71.000082	71/2686(2.6%)	82	24
20	306	0.05	26.00021	26/5352(0.5%)	210	29
25	309	0.06	30.000294	30/6723(0.4%)	294	26

表 3.2 最適車數數學模式數值測試（站點數 $N=50$ ）

T	V	CPU Time(sec)	ObjVal	需求缺量比例	$\sum_{i,j,t} x_{ij}^t$	$\max \left\{ \sum_{i,j} x_{ij}^t, \forall t \right\}$
				$\sum_{i,j,t} (U_{ij}^t + L_{ij}^t) / \sum_{i,j,t} B_{ij}^t$		
10	7559	1.53	760.00124	760/73396(1.0%)	1240	222
20	7493	8.09	2021.002239	2021/147136(1.4%)	2239	251
25	7560	8.89	1668.003603	1668/183644(0.9%)	3603	250

表 3.3 最適車數數學模式數值測試（站點數 $N=100$ ）

T	V	CPU Time(sec)	ObjVal	需求缺量比例	$\sum_{i,j,t} x_{ij}^t$	$\max \left\{ \sum_{i,j} x_{ij}^t, \forall t \right\}$
				$\sum_{i,j,t} (U_{ij}^t + L_{ij}^t) / \sum_{i,j,t} B_{ij}^t$		
10	29986	19.35	3728.001891	3728/297653(1.3%)	1891	433
20	30426	73.18	3048.009499	3048/595229(0.5%)	9499	791
25	29977	130.45	8601.008494	8601/741801(1.2%)	8494	837

3.4.2 車輛運補數學模式之數值測試

表 3.4 至 3.6 為最適車數運補模式與最大需求滿足運補模式數值測試結果比較，起訖需求 (B_{ij}^t) 參數與 3.4.1 節相同，並使用 3.4.1 節最適車數數學模式決策之車輛初始配置與員工數量。需求缺量比例為超量變數與缺量變數總合除以歷史借還需求總合，即最適車數運補模式需求缺量比例為 $\sum_{i,j,t} (U_{ij}^t + L_{ij}^t + \tilde{U}_i^t + \tilde{L}_i^t) / \sum_{i,j,t} B_{ij}^t$ ，而最大需求滿足運補模式需求缺量比例為 $\sum_{i,j,t} (U_{ij}^t + L_{ij}^t) / \sum_{i,j,t} B_{ij}^t$ 。當規劃時期長度等於期數 ($h = T$) 時，即直接對總時間長度的運補工作進行一次求解；當規劃時期長度小於期數 ($h < T$) 時，則為滾動式的分次求解。

觀察表 3.4 至 3.6，除了站點數 (N) 為 100 的最大需求滿足運補模式部分情境下，其它皆可在單次求解時間 3600 秒的限制下求解完畢，而單次求解時間受到站點數 (N) 與期數 (T) 的規模大小影響，最大化需求滿足運補模式的單次求解普遍需要較最適車數運補模式多的時間。在規劃時期長度 (h) 較短時，最適車數運補模式的需求缺量比例較低，表現較佳，推測是因為最適車數運補模式具有總期間最適車數的資訊，而最大化需求滿足運補模式則是進行短期規劃的區域最佳解，時間連接性較差。在規劃時期長度 (h) 較長時，則是最大化需求滿足運補模式的需求缺量比例較低，表現較佳，推測是最大化需求滿足運補模式直接對歷史借還需求進行最佳化，而最適車數模式則同時考量歷史借還需求與最適車數，限制較多。

表 3.4 運補模式數值測試 (站點數 $N=10$)

(T, V, P)	h	最適車數運補模式			最大需求滿足運補模式		
		CPU Time Avg (sec)	需求缺量比例	運補 工作	CPU Time Avg (sec)	需求缺量比例	運補 工作
(10, 286, 24)	1	0	150/2686(5.6%)	3	0	269/2686(10.0%)	0
	2	0.01	100/2686(3.7%)	137	0.01	217/2686(8.1%)	7
	3	0.02	98/2686(3.6%)	133	0.01	115/2686(4.3%)	123
	5	0.03	99/2686(3.7%)	140	0.03	100/2686(3.7%)	121
	7	0.05	102/2686(3.8%)	139	0.08	99/2686(3.7%)	124
	9	0.05	102/2686(3.8%)	129	0.06	94/2686(3.5%)	119
	10	0.07	102/2686(3.8%)	125	0.06	98/2686(3.6%)	109
(20, 306, 29)	1	0	208/5352(3.9%)	28	0	400/5352(7.5%)	0
	2	0.01	73/5352(1.4%)	324	0	261/5352(4.9%)	59
	3	0.02	74/5352(1.4%)	347	0.01	79/5352(1.5%)	397
	5	0.02	74/5352(1.4%)	339	0.03	50/5352(0.9%)	367
	7	0.04	70/5352(1.3%)	318	0.05	48/5352(0.9%)	354
	9	0.05	70/5352(1.3%)	335	0.08	48/5352(0.9%)	341
	20	0.11	78/5352(1.5%)	320	0.46	48/5352(0.9%)	340
(25, 309, 26)	1	0	246/6723(3.7%)	78	0	506/6723(7.5%)	0
	2	0.01	111/6723(1.7%)	436	0	381/6723(5.7%)	77
	3	0.01	107/6723(1.6%)	429	0.01	124/6723(1.8%)	447
	5	0.02	112/6723(1.7%)	419	0.02	97/6723(1.4%)	403
	7	0.04	116/6723(1.7%)	409	0.04	98/6723(1.5%)	375
	9	0.05	101/6723(1.5%)	411	0.06	81/6723(1.2%)	405
	25	0.17	106/6723(1.6%)	413	0.43	90/6723(1.3%)	402

表 3.5 運補模式數值測試 (站點數 $N = 50$)

(T, V, P)	h	最適車數運補模式			最大需求滿足運補模式		
		CPU Time Avg (sec)	需求缺量比例	運補 工作	CPU Time Avg (sec)	需求缺量比例	運補 工作
(10,7559,222)	1	0.06	1985/73396(2.7%)	17	0.02	4765/73396(6.5%)	0
	2	0.29	1404/73396(1.9%)	1631	0.09	3461/73396(4.7%)	15
	3	1.01	1375/73396(1.9%)	1482	0.41	1387/73396(1.9%)	1736
	5	3.22	1341/73396(1.8%)	1506	1.97	1222/73396(1.7%)	854
	7	5.54	1351/73396(1.8%)	1474	10.18	1181/73396(1.6%)	1365
	9	8.8	1364/73396(1.9%)	1503	16.28	1083/73396(1.5%)	1481
	10	10.75	1407/73396(1.9%)	1509	20.47	1202/73396(1.6%)	1199
(20,7493,251)	1	0.06	4055/147136(2.8%)	212	0.02	12507/147136(8.5%)	0
	2	0.25	2592/147136(1.8%)	3721	0.1	6449/147136(4.4%)	101
	3	0.75	2626/147136(1.8%)	3431	0.35	2679/147136(1.8%)	3166
	5	2.31	2530/147136(1.7%)	3461	1.77	2565/147136(1.7%)	2456
	7	4.98	2486/147136(1.7%)	3477	6.67	2425/147136(1.6%)	2464
	9	6.18	2539/147136(1.7%)	3470	16.57	2418/147136(1.6%)	2688
	20	18.71	2509/147136(1.7%)	3478	133.69	2370/147136(1.6%)	2864
(25,7560,250)	1	0.05	4983/183644(2.7%)	310	0.02	13662/183644(7.4%)	0
	2	0.24	2852/183644(1.6%)	5051	0.09	7139/183644(3.9%)	233
	3	0.71	2920/183644(1.6%)	4591	0.34	2982/183644(1.6%)	4494
	5	2.57	2808/183644(1.5%)	4753	2.02	2639/183644(1.4%)	3481
	7	5.15	2729/183644(1.5%)	4939	10.05	2410/183644(1.3%)	4021
	9	9.09	2752/183644(1.5%)	4950	15.87	2429/183644(1.3%)	3973
	25	47.93	2774/183644(1.5%)	4934	135.87	2370/183644(1.3%)	4068

表 3.6 運補模式數值測試（站點數 $N=100$ ）

(T, V, P)	h	最適車數運補模式			最大需求滿足運補模式		
		CPU Time Avg (sec)	需求缺量比例	運補工作	CPU Time Avg (sec)	需求缺量比例	運補工作
(10,29986,433)	1	0.29	5565/297653(1.9%)	56	0.08	16797/297653(5.6%)	0
	2	1.76	4976/297653(1.7%)	3211	0.46	10085/297653(3.4%)	30
	3	5.93	4874/297653(1.6%)	2934	3.26	4940/297653(1.7%)	3266
	5	25.66	4763/297653(1.6%)	2955	23.27	4588/297653(1.5%)	2178
	7	61.86	4864/297653(1.6%)	2973	91.43	4568/297653(1.5%)	2410
	9	127.18	4788/297653(1.6%)	3001	276.76	4432/297653(1.5%)	2398
	10	94.37	4862/297653(1.6%)	3046	446.24	4535/297653(1.5%)	2470
(20,30426,791)	1	0.41	12344/595229(2.1%)	231	0.11	40852/595229(6.9%)	0
	2	2.58	6584/595229(1.1%)	1252	0.58	17550/595229(2.9%)	89
	3	6.44	6524/595229(1.1%)	1104	2.49	6040/595229(1.0%)	1251
	5	28.05	6312/595229(1.1%)	1158	27.4	5225/595229(0.9%)	8521
	7	66.71	6183/595229(1.0%)	1176	380.14	RunTimeOut, r=12	-
	9	138.97	6126/595229(1.0%)	1196	540.2	RunTimeOut, r=11	-
	20	566.94	6134/595229(1.0%)	1213	1472.2	4525/595229(0.8%)	1159
(25,29977,837)	1	0.29	16484/741801(2.2%)	631	0.09	59972/741801(8.1%)	0
	2	1.6	9925/741801(1.3%)	1480	0.46	24190/741801(3.3%)	405
	3	5.11	9653/741801(1.3%)	1429	2.36	10649/741801(1.4%)	1154
	5	22.44	9649/741801(1.3%)	1410	26.75	9829/741801(1.3%)	1060
	7	55.33	9578/741801(1.3%)	1406	948.19	RunTimeOut, r=4	-
	9	97.69	9549/741801(1.3%)	1409	540.49	RunTimeOut, r=10	-
	25	634.79	9580/741801(1.3%)	1400	3600.09	RunTimeOut, r=1	-

第四章 眾包運送平台任務推送研究

本章將探討眾包運送平台的任務推送模型與路徑分群前處理進行探討。4.1 節基於使用者的路徑使用紀錄提出路徑分群演算法，統整路徑使用紀錄資訊。4.2 節建構有推送與接收限制的任務推送模型，目的為精準推送。4.3 節為數值測試結果。

4.1 路徑分群演算法

4.1.1 問題描述

無樁式載具共享系統中的使用紀錄資料量非常龐大，本研究提出路徑分群演算法，作為使用者路徑資料的前處理。分期輸入路徑紀錄，定義路徑間的相異度，結合數學規劃模型設計演算法，輸出使用者的路徑分群結果，包含分群機率與代表性路徑，有效統整各期的路徑資訊。

4.1.2 路徑間的相異度

給定兩路徑與分割數(c)，兩個路徑分別均勻分為 c 段， $c+1$ 個節點依序相連。本研究定義路徑間的相異度為 $c+1$ 條連線的平均長度，如圖 4.1 所示，給定分割數為 10，路徑間的相異度為 11 條路徑間連線的平均長度。

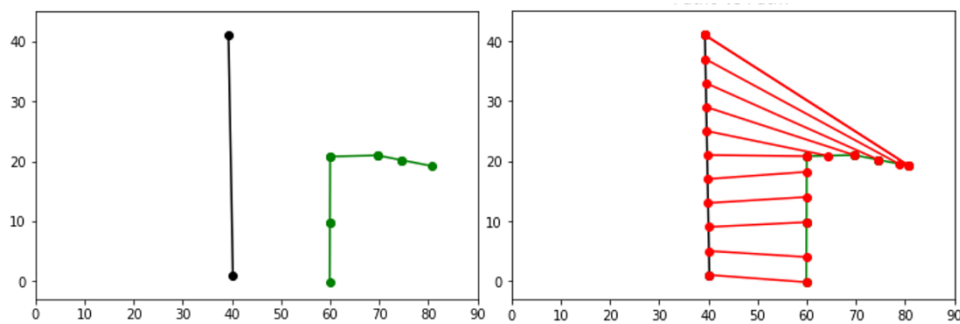


圖 4.1 路徑間相異度($c=10$)示意圖

4.1.3 路徑分群模型

基於 4.1.2 節定義路徑之間的相異度，本節提出給定群集數的路徑分群數學規劃模型，將在 4.1.4 節配合參數無異長度(d)設計路徑分群演算法。

參數

n 路徑記錄數量， $\forall i, j \in \{1, \dots, n\}$

d_{ij} 路徑 i 與路徑 j 之間的相異度

K 群集數， $\forall k \in \{1, \dots, K\}$

決策變數

x_{ik} 若路徑 i 分至第 k 群則 $x_{ik} = 1$ ，反之 $x_{ik} = 0$

y_{ij} 若路徑 i 與路徑 j 被分至同一群則 $y_{ij} = 1$ ，反之 $y_{ij} = 0$

數學模式

$$\min \sum_i \sum_j d_{ij} y_{ij} - \sum_i \sum_j d_{ij} (1 - y_{ij}) \quad (4.1)$$

$$\sum_k x_{ik} = 1, \quad \forall i \quad (4.2)$$

$$\sum_i x_{ik} \geq 1, \quad \forall k \quad (4.3)$$

$$y_{ij} + (2 - x_{ik} - x_{jk}) \geq 1, \quad \forall i, j, k \quad (4.4)$$

$$x_{ik} \in \{0, 1\}, \quad \forall i, k \quad (4.5)$$

式 4.1 為目標式，目標為最小化組內路徑差異度總量與組間差異度總量，前項 $\sum_i \sum_j d_{ij} y_{ij}$ 為組內差異度總量，後項 $\sum_i \sum_j d_{ij} (1 - y_{ij})$ 為組間差異度總量。式 4.2 限制每條路徑恰好屬於一群集。式 4.3 限制每個群集至少具有一條路徑。式 4.4 定義決策變數 y_{ij} 。式 4.5 定義決策變數 x_{ik} 為二元變數。

4.1.4 路徑分群演算法設計

【步驟一】在給定時段之路徑紀錄集中依序取出使用者與該使用者的路徑紀錄，計算

該使用者的路徑紀錄相異度矩陣，並令群集數 $K=1$ 。

【步驟二】給定群集數 K 與使用者的路徑紀錄，運行 4.1.3 節的路徑分群模型。

【步驟三】每個群集中，選出與群集中其它路徑相異度加總最低之路徑作為該群集的代表性路徑。

【步驟四】檢驗路徑分群結果，若存在兩個群集的代表性路徑相異度小於參數無異長度(d)時，確定該使用者的群集數為 $K-1$ ，回到步驟二；若不存在，則回到步驟三，群集數 $K=K+1$ 。

表 4.1 路徑分群演算法流程

Trajectory Clustering Algorithm

Input: Trajectory data in given period

Output: Clustering result for each user

Parameter: Indistinct distance

```
1: For each user do
2:   calculate difference matrix of each trajectory
3:    $K=1$ 
4:   While true do
5:     Run clustering model with  $K$ 
6:     For each cluster do
7:       Elect representative trajectory( $Traj_i$ )
8:     If there is difference of  $Traj_i, Traj_j$  is lower than Indistinct distance do:
9:       set user's clustering result =  $K-1$ 
10:    break
11:     $K=K+1$ 
```

4.2 任務推送數學模型

4.2.1 問題描述

眾包運送系統提供將貨物或餐點從需求起點運送至需求終點的運送服務。本研究定義群眾與任務的配適分數作為基準，並考量系統與群眾雙方需求，系統方期望任務多多被推送，至少超過推送次數下界；群眾方期望接收適量的任務，具有接收任務量上限，提出一數學規劃模型以最大化系統配適分數值為目標，求解推送任務決策。

4.2.2 問題假設

1. 已知每位使用者紀錄的路徑分群結果，包含群集的代表性路徑與機率。
2. 已知每個外部委託之運送任務對系統之收益。
3. 假設每個運送任務完成時，系統給群眾的報酬皆相同。

4.2.3 數學模式

集合

N 節點集合， $i, j \in N$

A 節線集合， $(i, j) \in A$ ， $i, j \in N$

M 任務節點集合， $m_i \in M$ ， $M \subseteq N$

C 群眾節點集合， $c_i \in C$ ， $C \subseteq N$

參數

s 虛擬供給節點

t 虛擬需求節點

c_d 虛擬群眾節點， $c_d \in C$

r_i 任務報酬

p_{ij} 任務 i 被群眾 j 承接執行的機率， $i \in M, j \in C$

d_{ij} 任務 i 被群眾 j 承接執行的繞行成本， $i \in M, j \in C$

lb 任務推送次數下限

ub 接收任務量上限

w_r 任務報酬權重

w_d 繞行成本權重

決策變數

x_{ij} 流經節線 (i,j) 的流量

繞行成本 d_{ij} 數值

本研究將以最短距離作為群眾承接任務的繞行路徑估計方式。如圖 4.2 所示，給定一群眾行駛路徑，起始於節點 O_u 且終止於節點 D_u ，現有一運送任務，具有取貨節點 O_m 與交貨節點 D_m 的需求，該群眾承接此任務的路徑估計將為 O_u 與 O_m 、 O_m 與 D_m 、 D_m 與 D_u 連線，而繞行成本 d_{ij} 則為原路徑與承接任務估計路徑的長度差距。

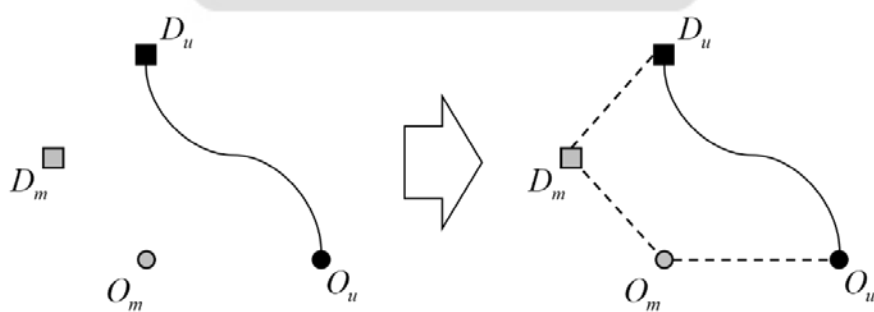


圖 4.2 繞行路徑估計方式

模型前處理

執行模型前，必須完成模型的前處理，即找出每個任務與每位群眾之路徑分群的最高配適分數值之配對方式。如圖 4.3 所示，群眾 c_j 具有 k 個路徑分群，任務 m_i 與路

徑分群的配適分數為 $f(m_i, cluser k) = p_{ik}(w_r r_i - w_d d_{ik})$ ，模型前處理將選出最大的路徑分群配適分數 $f(m_i, cluser k')$ ，作為任務 m_i 與群眾 c_j 的配適分數。

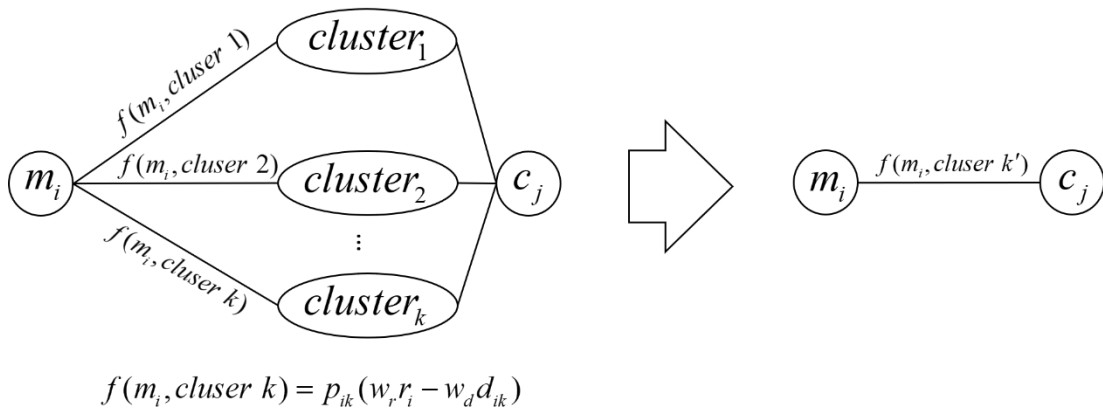


圖 4.3 任務群眾配適前處理示意圖

數學模式

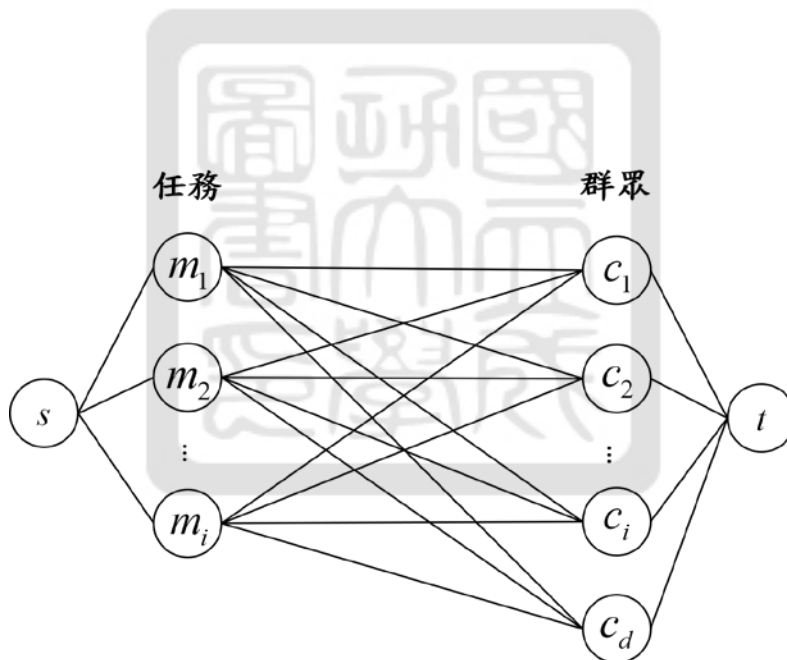


圖 4.4 任務推送網路示意圖

$$\max \sum_{i \in M, j \in C} x_{ij} \cdot p_{ij} (w_r r_i - w_d d_{ij}) \quad (4.6)$$

$$\sum_j x_{ij} = \sum_j x_{ji} \quad \forall i \in M \cup C \quad (4.7)$$

$$x_{sj} \geq lb \quad \forall j \in M \quad (4.8)$$

$$x_{it} \leq ub \quad \forall i \in C \setminus \{c_d\} \quad (4.9)$$

$$x_{ij} \leq 1 \quad \forall i \in M, j \in C \setminus \{c_d\} \quad (4.10)$$

式 4.6 為目標式，最大化模型整體配適分數值，其中 $p_{ij}(w_r r_i - w_d d_{ij})$ 為任務 m_i 與群眾 m_j 的配對分數。如圖 4.4 所示，式 4.7 定義任務節點與群眾節點的流量守恆，流進等於流出。式 4.8 限制連接虛擬供給節點與任務節點之節線流量至少為推送任務次數下限。式 4.9 限制連接群眾節點與虛擬需求節點之節線流量至多為接收任務量上限，而虛擬群眾節點 c_d 則不受此限。式 4.10 限制除了虛擬群眾節點 c_d ，其它連結群眾節點與任務節點之節線流量至多為 1 單位流量。



4.3 數值測試

測試環境為 Window10 作業系統，搭配 IntelCore i7-6770，3.40GHz*8 處理器與 16GB 記憶體，以 Python3.6.3 為程式語言界面，搭配 Gurobi 8.0.1 版求解數學模式。

4.3.1 路徑分群演算法之數值測試

表 4.2 為路徑分群演算法的數值測試結果。隨機產生路徑資料集，給定無異長度 (d) 為 10，進行不同路徑數量的分群。

從數值測試的結果看來，差異矩陣的運算時間皆不久，而求解時間隨著路徑使用紀錄遞增時增加，且在路徑數量到達 50 時，大部分求解時間將超過運算時間 3600 秒的限制，大多運算至 $K=2$ 或 $K=3$ ，可見路徑數量對路徑分群模型求解速度的影響甚大。而在路徑數量 40 的測試中，求解時間最小為 11.5 秒，最大為 295.36，變異程度非常大，演算法的求解效率可能受到路徑資料集很大的影響。

表 4.2 路徑分群演算法數值測試

路徑數量	差異矩陣運算時間(sec)	分群演算法求解時間(sec)	分群結果(K)
20	0.21	0.11	2
	0.2	0.39	3
	0.2	0.34	3
	0.2	0.62	4
	0.19	0.17	2
30	0.45	6.66	4
	0.44	2.14	3
	0.46	1.48	3
	0.46	7.45	4
	0.44	1.95	3
40	0.83	31.96	4
	0.83	215.28	5
	0.81	295.36	5
	0.8	11.5	3
	0.83	36.32	4
50	1.3	*3599.27 (3)	-
	1.29	*3599.31 (4)	-
	1.28	*3598.98 (2)	-
	1.39	*3599.19 (3)	-
	1.26	658.38	4
60	1.91	*3598.46 (2)	-
	1.95	*3598.45 (2)	-
	2.1	*3598.32 (2)	-
	1.83	*3598.53 (2)	-
	1.86	*3598.5 (2)	-

「*」表示求解時間超過 3600 秒，括弧內的數字表示演算法終止時的 K 值

4.3.2 任務推送數學模式之數值測試

表 4.3 與 4.4 為任務推送數學模式的數值測試結果。給定任務數量、群眾數量、推送下限與推送上限的參數值，隨機產生任務與群眾的最佳配適分數值進行求解。

測試結果顯示求解時間皆小於 1 秒，速度受測試規模的影響較小。從系統總配適分數的結果來看，系統的總配適分數受推送下限與接受上限的影響較小，受任務數量與群眾數量的影響較大，顯示系統中的任務與群眾的數量將有助於增加系統的總效益。

表 4.3 任務推送數學模式數值測試（任務數量為 20）

任務數量	群眾數量	推送下限	接收上限	系統總配適分數	求解時間(sec)
20	300	30	10	$1.14 \cdot 10^{35}$	0.0648
			20	$1.14 \cdot 10^{35}$	0.0426
		50	10	$1.14 \cdot 10^{35}$	0.0429
			20	$1.14 \cdot 10^{35}$	0.0768
	500	30	10	$1.91 \cdot 10^{35}$	0.0788
			20	$1.91 \cdot 10^{35}$	0.0738
		50	10	$1.91 \cdot 10^{35}$	0.0718
			20	$1.91 \cdot 10^{35}$	0.1057
	1000	30	10	$3.80 \cdot 10^{35}$	0.1587
			20	$3.80 \cdot 10^{35}$	0.1866
		50	10	$3.80 \cdot 10^{35}$	0.161
			20	$3.80 \cdot 10^{35}$	0.198

表 4.4 任務推送數學模式數值測試（任務數量為 50、100）

任務數量	群眾數量	推送下限	接收上限	系統總配適分數	求解時間(sec)
50	300	30	10	$3.43 \cdot 10^{35}$	0.1187
			20	$3.43 \cdot 10^{35}$	0.1307
		50	10	$3.43 \cdot 10^{35}$	0.0928
			20	$3.43 \cdot 10^{35}$	0.1237
	500	30	10	$5.67 \cdot 10^{35}$	0.1572
			20	$5.67 \cdot 10^{35}$	0.2294
		50	10	$5.67 \cdot 10^{35}$	0.2168
			20	$5.67 \cdot 10^{35}$	0.1955
	1000	30	10	$1.14 \cdot 10^{36}$	0.4328
			20	$1.14 \cdot 10^{36}$	0.3562
		50	10	$1.14 \cdot 10^{36}$	0.3911
			20	$1.14 \cdot 10^{36}$	0.3634
100	300	30	10	$7.27 \cdot 10^{35}$	0.1791
			20	$7.27 \cdot 10^{35}$	0.2124
		50	10	$7.27 \cdot 10^{35}$	0.2074
			20	$7.27 \cdot 10^{35}$	0.2399
	500	30	10	$1.21 \cdot 10^{36}$	0.3347
			20	$1.21 \cdot 10^{36}$	0.3328
		50	10	$1.21 \cdot 10^{36}$	0.3674
			20	$1.21 \cdot 10^{36}$	0.386
	1000	30	10	$2.42 \cdot 10^{36}$	0.708
			20	$2.42 \cdot 10^{36}$	0.708
		50	10	$2.42 \cdot 10^{36}$	0.7291
			20	$2.42 \cdot 10^{36}$	0.6938

第五章 結論與未來研究方向建議

5.1 結論與貢獻

在共享經濟蓬勃發展的時代，如何維持穩定的成長是共享平台經營者重要的課題。在載具共享系統中，維持載具長時間的使用率是最大的目標，因此分時租賃的手法將吸引大量具有短程需求的顧客上門，成為「城市的最初與最後一哩路」的交通手段。然而，依時間變動的大量使用需求將對載具共享系統產生劇烈的影響，尤其是無樁式載具共享系統，鬆弛了固定的站點設施限制，載具將散落在城市的各處，而車輛的配置方式有可能不利於使用者的需求，因此對載具共享系統進行動態運補是必要的。而眾包運送平台的目標則是提供城市內的短途運送需求以群眾外包方式進行快遞，倚靠群眾接單執行而非雇用全職駕駛，因為城市內的運送需求分布並不穩定，以最熱門的餐點運送平台來說，有限的員工無法短時間消化用餐時間的爆量需求，成為了群眾物流絕佳的應用情境。然而現今的眾包運送平台必須讓群眾主動搜尋運送任務，繁雜的資訊不但會降低群眾接單的意願，而具有時間限制的運送任務將處於被選擇的被動狀態，群眾與運送任務的配對沒有效率。

在共享平台的後台數據紀錄，若善加應用將會是具有高價值的材料。本研究基於無樁式載具共享系統的使用紀錄，提出兩方面的應用，首先為無樁式載具共享系統的動態運補，目的為最大化使用者的借還需求滿足量，提升無樁式載具共享系統自身的營運效益。過去的文獻對於無樁式共享系統多基於營運區域的借還需求進行分群處理，然而因時間變動的借還需求將劇烈影響分群大小與範圍，因此在不同時段的地理分群大小範圍不一，可能忽略分群間的交界地區，且無法控制的分群大小將不易聚焦運補工作的效用，難以落實於實務上的應用，本研究提出以等距的方式進行虛擬站點的劃分，此舉將有助於降低應用於實務的複雜度，並提供較細緻的運補指示。而運補車輛的方式，本研究考量電動機車體積與重量，選擇以員工駕駛車輛的移動方式，將營運

期間切分成多期，建構滾動式數學模型決策運補任務，使載具時地配置最佳化。過往文獻多對營運時的運補工作一次求解全期，由數值分析的結果可知，藉由設定介於單期與全期之間的決策期間長度，採用滾動式的運補工作決策模型的求解時間將小於一次求解全期運補工作模型，與求解單期運補工作相比，滾動式的運補工作決策也將顯著提升決策之運補工作的效益。

而無樁式載具共享系統之使用紀錄的另一個應用為眾包運送平台的任務推送，由於不受到固定站點設施的限制，無樁式載具共享系統的使用紀錄將非常接近使用者的真實需求，適合應用於眾包運送平台找出低成本的順風車進行運送任務的推播，過往的文獻多假設已知眾包運送平台的志願者習慣使用路徑，未交代志願者習慣的數據來源，本研究提出結合無樁式載具共享系統的數據與眾包運送平台的經營模式，提供無樁式載具共享系統數據的新應用，並降低眾包運送平台的志願者接受任務的成本，彼此達到互利的效果。本研究以路徑分群手法統整無樁式載具共享系統使用者的使用習慣，包含路徑分群的代表性路徑與機率，讓眾包運送平台掌握駕駛的習慣，接著考量駕駛與眾包運送平台雙方的需求，建構具有推送限制的任務推送模型，讓任務的推送次數超過推送下限，達到多多推播任務的效果，同時限制群眾接收的任務不超過接收上限，避免過多的任務推播成為垃圾訊息，達到精準推送的效果。

本研究提出之無樁式載具共享系統動態運補研究與眾包運送平台任務推送研究皆是致力於推廣新興服務的平台，有別於傳統的租賃模式與快遞模式，提供實體載具的分時租借與運送服務的外包機會，讓使用者透過平台享受自由、便利的服務。

綜上所述，本研究的貢獻可以歸納為以下幾點：

1. 應用無樁式載具共享系統的使用紀錄於眾包運送平台：

本研究提出將無樁式載具共享系統的使用紀錄應用於眾包運送平台，考量了載具共享系統的路徑資料與眾包運送平台需求的相符性加以應用，有別於文獻多假設眾包運送平台的候選駕駛習慣已知，而無交代如何得知。

2. 建構動態運補的滾動式求解模式：

本研究建構的滾動式動態運補數學模型放大了規劃期間的長度，將在有限的時間內有效提高運補任務的效益，應用於實務時，運行多次的動態運補模型可依據運行當下車數與員工配置進行參數調整，具有可逐步調整模型參數的優點。

3. 提出具有代表性路徑與機率數值的路徑分群方式：

本研究提出之路徑分群方式考量眾包運送平台的需求，設計路徑分群輸出代表性路徑與其機率數值，其中的代表性路徑為駕駛的實際使用紀錄之一，相較於使用重組路線片段方式的文獻，本研究的路徑較合理且符合現實。

4. 建構具有精準推送效果之任務推送模型：

本研究建構之任務推送模型考量系統與群眾雙方的需求，具有推送數量的下限與接收任務的上限，多多推播運送任務的同時避免群眾接收過多的垃圾訊息，配合群眾的使用紀錄路徑分群結果，預期可達到精準推送的目的。

5.2 未來研究方向建議

本研究針對無樁式載具共享系統的動態運補與群眾運送系統的任務推送進行決策上的探討，尚有不少未臻完美的部分，以下列舉幾點可繼續延伸之議題：

1. 虛擬站點的轉化方式：

本研究針對無樁式載具共享系統動態運補決策所提出的虛擬站點轉化方式必須決定虛擬站點的長寬大小與地理劃分的初始位置，然而，這兩個參數將大大的影響使用者的借還需求如何被統整。當虛擬站點的面積較大時，較容易稀釋實際的借還需求點，虛擬站點的面積較小時則容易造成過多的虛擬站點數量；而虛擬站點地理劃分的初始位置座標則是影響邊界的借還需求，若是初始位置進行小幅度的偏移，則邊界的借還需求有機會被劃分至不同的站點。不同的虛擬站點大小與劃分初始位置的參數組合將會有不同的借還需求統整結果，因此未來可考慮整合多個參數組合的虛擬站點轉化方式。

2. 滾動式動態運補模型的求解效率：

在無樁式載具共享系統的虛擬站點轉化中，雖然可過濾需求微小的虛擬站點，然而考量無樁式載具共享系統的營運範圍，仍有機會具有數量可觀的站點數，在站點數眾多時，滾動式動態運補仍存在求解效率不彰的問題，因此未來研究可朝基於滾動式動態運補模型的基礎，進一步探討如何應付大量的虛擬站點數量的運補工作求解。

3. 分群演算法的求解效率：

本研究結合數學規劃模型提出之分群演算法，雖然在路徑數量小的時候表現優秀，可在非常短的時間內求解，然而當路徑數量稍具規模時，分群演算法在有限的時間內將無法提供好的求解效果。本研究基於眾包運送平台的需求定義路徑的差異度、代表性路徑與其機率，未來研究可考慮建立其它演算法流程以加強分群的穩健性與分群效果。

4. 任務推送模型的設定：

本研究在任務推送模型主要考量系統與群眾雙方的需求，設定推送下限與接收上限，限制任務的推送次數必須超過推送下限，而群眾接收任務的次數必須小於接收上限。然而，群眾與任務的關係則是經過過濾，在群眾具有的多個路徑分群中，選擇以配適度最高的路徑群集與任務連結，作為群眾與任務媒合度的標準，此方式忽略了一位群眾在同一時間最多只能實現一種路徑分群的行駛，因此未來研究可考慮將此程序省去，以不同的手法建構任務推送模型。

參考文獻

- 周佰賢. (2015). 考慮需求變化狀況及增設臨停區之公共自行車共享系統租借站分群與車輛調度策略研究. 成功大學工業與資訊管理學系學位論文, 1-72.
- 洪菁蓬. (2011). 公共自行車租借系統之最佳租借站位址設置及車輛運補策略之研究. 成功大學工業與資訊管理學系學位論文, 1-80.
- 侯貞泰. (2016). 公共自行車共享系統之群眾運補策略數學模型與資料視覺化工具設計研究. 成功大學工業與資訊管理學系學位論文, 1-71.
- 張立蓁. (2010). 都會區公共自行車租借系統之設計與營運方式研究. 成功大學工業與資訊管理學系學位論文, 1-83.
- 梁瑜庭. (2013). 公共電動機車共享系統之最佳車輛佈署策略研究. 成功大學工業與資訊管理學系學位論文, 1-83.
- 廖敏婷. (2012). 考慮需求比例及暫時人力配置之公共自行車租借系統管理策略研究. 成功大學工業與資訊管理學系學位論文, 1-65.
- Caggiani, L., Camporeale, R., Ottomanelli, M., & Szeto, W. Y. (2018). A modeling framework for the dynamic management of free-floating bike-sharing systems. *Transportation Research Part C: Emerging Technologies*, 87, 159-182.
- Cheng, S. F., Chen, C., Kandappu, T., Lau, H. C., Misra, A., Jaiman, N., ... & Koh, D. (2018). Scalable urban mobile crowdsourcing: Handling uncertainty in worker movement. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 9(3), 26.
- Jorge, D., Correia, G. H., & Barnhart, C. (2014). Comparing optimal relocation operations with simulated relocation policies in one-way carsharing systems. *IEEE Transactions on Intelligent Transportation Systems*, 15(4), 1667-1675.
- Kek, A. G., Cheu, R. L., & Chor, M. L. (2006). Relocation simulation model for multiple-

- station shared-use vehicle systems. *Transportation research record*, 1986(1), 81-88.
- Kek, A. G., Cheu, R. L., Meng, Q., & Fung, C. H. (2009). A decision support system for vehicle relocation operations in carsharing systems. *Transportation Research Part E: Logistics and Transportation Review*, 45(1), 149-158.
- Lee, J. G., Han, J., & Whang, K. Y. (2007, June). Trajectory clustering: a partition-and-group framework. In *Proceedings of the 2007 ACM SIGMOD international conference on Management of data* (pp. 593-604). ACM.
- Nanni, M., & Pedreschi, D. (2006). Time-focused clustering of trajectories of moving objects. *Journal of Intelligent Information Systems*, 27(3), 267-289.
- Nourinejad, M., Zhu, S., Bahrami, S., & Roorda, M. J. (2015). Vehicle relocation and staff rebalancing in one-way carsharing systems. *Transportation Research Part E: Logistics and Transportation Review*, 81, 98-113.
- Raviv, T., Tzur, M., & Forma, I. A. (2013). Static repositioning in a bike-sharing system: models and solution approaches. *EURO Journal on Transportation and Logistics*, 2(3), 187-229.
- Sampaio Oliveira, A. H., Savelsbergh, M. W. P., Veelenturf, L. P., & van Woensel, T. (2017). Crowd-based city logistics. *SCL Report Series*, (17-02).
- Weikl, S., & Bogenberger, K. (2013). Relocation strategies and algorithms for free-floating car sharing systems. *IEEE Intelligent Transportation Systems Magazine*, 5(4), 100-111.
- Yildiz, B., & Savelsbergh, M. (2017). Provably high-quality solutions for the meal delivery routing problem. *Georgia Institute of Technology*.
- Yildiz, B., & Savelsbergh, M. (2018). Service and Capacity Planning in Crowd-Sourced Delivery. <https://doi.org/10.2139/ssrn.3135550>