

國 立 交 通 大 學

運 輸 與 物 流 管 理 學 系

碩 士 論 文

考量重疊服務區域與容量限制之車輛路徑問題

Capacitated Vehicle Routing Problem with
Overlapping Service Regions

研 究 生 ː 謝宗桓

指 導 教 授 ː 姚銘忠 教授

林春成 教授

中 華 民 國 一 〇 七 年 七 月

考量重疊服務區域與容量限制之車輛路徑問題

Capacitated Vehicle Routing Problem with Overlapping Service Regions

研究生：謝宗桓

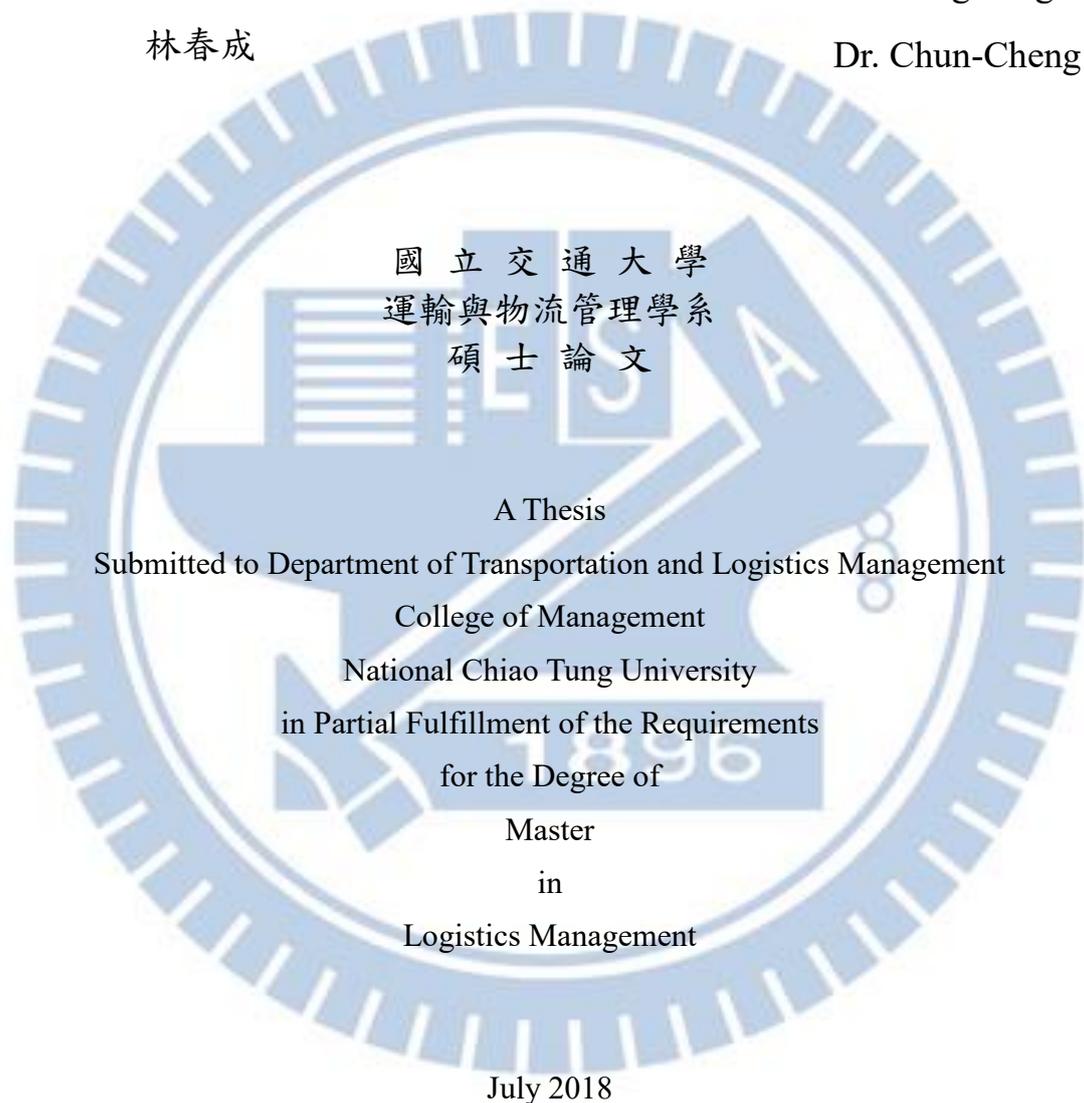
Student: Chung-Huan Hsieh

指導教授：姚銘忠

Advisor: Dr. Ming-Jong Yao

林春成

Dr. Chun-Cheng Lin



Hsinchu, Taiwan, Republic of China

中華民國一〇七年七月

考量重疊服務區域與容量限制之車輛路徑問題

研究生：謝宗桓

指導教授：姚銘忠 教授

林春成 教授

國立交通大學運輸與物流管理學系碩士班

摘要

本研究所探討的「考量重疊服務區域與容量限制之車輛路徑問題」為傳統車輛路徑問題之延伸，在已知分區且考量重疊服務區域之架構下，求解物流公司配送顧客貨物需求之最佳車輛路徑決策，其中包含在分區中使用該分區之原有車輛、跨分區車輛或外包車輛與車輛路徑之安排。本研究提出一個創新的「廣義型重疊服務區域」，其乃是將過去研究所運用僅有兩分區彼此重疊之「鏈狀重疊服務區域」，允許多個分區相互重疊，增加車輛路徑安排與物流車隊運用之彈性，提供物流公司配送作業創新之模式。

本研究依照「考量重疊服務區域與容量限制之車輛路徑問題」之情境，建構數學模型。而由於車輛路徑問題為 NP-hard 問題，本問題有更高的複雜度，故本研究另提出基因演算法求解本問題。透過本研究提出基因演算法染色體編碼之資料結構，可以直觀且容易地掌握廣義型重疊服務區域之特性與描述對應的情境，且本研究設計一區域搜尋機制加強基因演算法之搜索能力。本研究以傳統車輛路徑問題之標竿題庫，加上廣義型重疊服務區域的特性，運用隨機產生之例題進行數據實驗分析。數據實驗的結果顯示，本研究所提出的基因演算法及區域搜尋機制確實具有優異的求解品質與效率，且驗證本研究採用之廣義型重疊服務區域確實能有效地降低物流公司之配送成本最多至 10%。

關鍵字：廣義型重疊服務區域、車輛路徑問題、基因演算法、區域搜尋

Capacitated Vehicle Routing Problem with Overlapping Service Regions

Student: Chung-Huan Hsieh

Advisor: Dr. Ming-Jong Yao
Dr. Chun-Cheng Lin

Department of Transportation and Logistics Management
National Chiao Tung University

Abstract

This study investigates the capacitated vehicle routing problem with overlapping service regions, which is an extension of the conventional vehicle routing problem (VRP). We are interested in solving the optimal decisions including the fleet deployment using the vehicles in the original region, the trans-regional vehicles and the vehicles from outsourcing and the corresponding vehicle routes for logistics companies to satisfy customers' demand with pre-determined and overlapping service regions. We present a new districting concept of "general overlapping service regions" (GOSR) in this study, that is a more generic version of the "chain-type overlapping service regions" than that presented in a previous study. GOSR allows more than two service regions overlapping with each other, increases the flexibility in vehicle routing and fleet deployment, and serves as a novel model of the distribution operations for logistics companies.

We formulate a mathematical model following the scenario of the capacitated VRP with overlapping service regions. It is well known that the conventional VRP is NP-hard. Therefore, the concerned problem in this study is more complicated and hence, more difficult, and we propose a genetic algorithm (GA) as our solution approach. The Data structure of chromosome encoding in our GA is not only comprehensive, but also easy to deal with the situation of GOSR. In addition, we also develop a local search mechanism to enhance the search ability of our GA. We randomly generate our instances in our numerical experiments by referring to the benchmark problems for the conventional VRP and taking into account of the characteristics of GOSR. Our experimental results show that our proposed GA with the proposed local search mechanism is able to obtain solutions with excellently quality effectively, and making use of GOSR may save distribution cost up to 10% for logistics companies.

Keywords: General overlapping service regions; vehicle routing problem; genetic algorithm; local search

誌謝

在交大運管所的兩年研究生涯中，首先要感謝指導老師姚銘忠教授，謝謝老師在兩年的學習過程中給予的指導和教誨，並且教導與叮囑未來在社會上人生的道路中為人處事應有的態度。同時也感謝共同指導老師林春成教授，在論文的研究與撰寫中，老師都給予了許多寶貴的建議以及觀念，使我受益良多。另外感謝口試委員陳銘芷教授與洪一薰教授，承蒙兩位老師在口試期間針對論文架構與內文的指正與建議，使我的論文更臻完善。也感謝兩年來交大運管所老師們課程上給予的指導，您們的教誨是我未來人生旅途的基石，在此致上由衷的感謝。

感謝研究室的大家，對比兩年前進入交大運管新環境的懵懂，侑進學長、孟嫻學姊和庭好學姊在生活上與研究中的帶領，你們的經驗分享與解惑使我能快速地融入碩班生活。仲柏、美好和慶慧兩年的共患難與互相扶持是我研究所前進的動力，許多課程與論文上的迷惘和壓力，以及研究室的一些行政程序，總是能在討論與鬥嘴中獲得解決的方案與壓力的釋放。國隆、柏諺和妘如也謝謝你們在我們忙碌的碩二生活中給予的協助與支持，祝你們在未來順利地完成研究所學業。另外也感謝所有碩二的同學們，一起出國與環島的旅行點綴了我的碩班生活，也豐富了人生的色彩，這兩年的點點滴滴將成為畢生難忘的回憶。

最後感謝我的家人，在求學階段總是給予我無條件的支持與鼓勵，讓我擁有前進的動力，在論文完成即將邁向人生的下個階段，感謝您們在課業與生活上的支持，使我能順利完成學業。

謝宗桓 謹誌

2018年7月於新竹交大

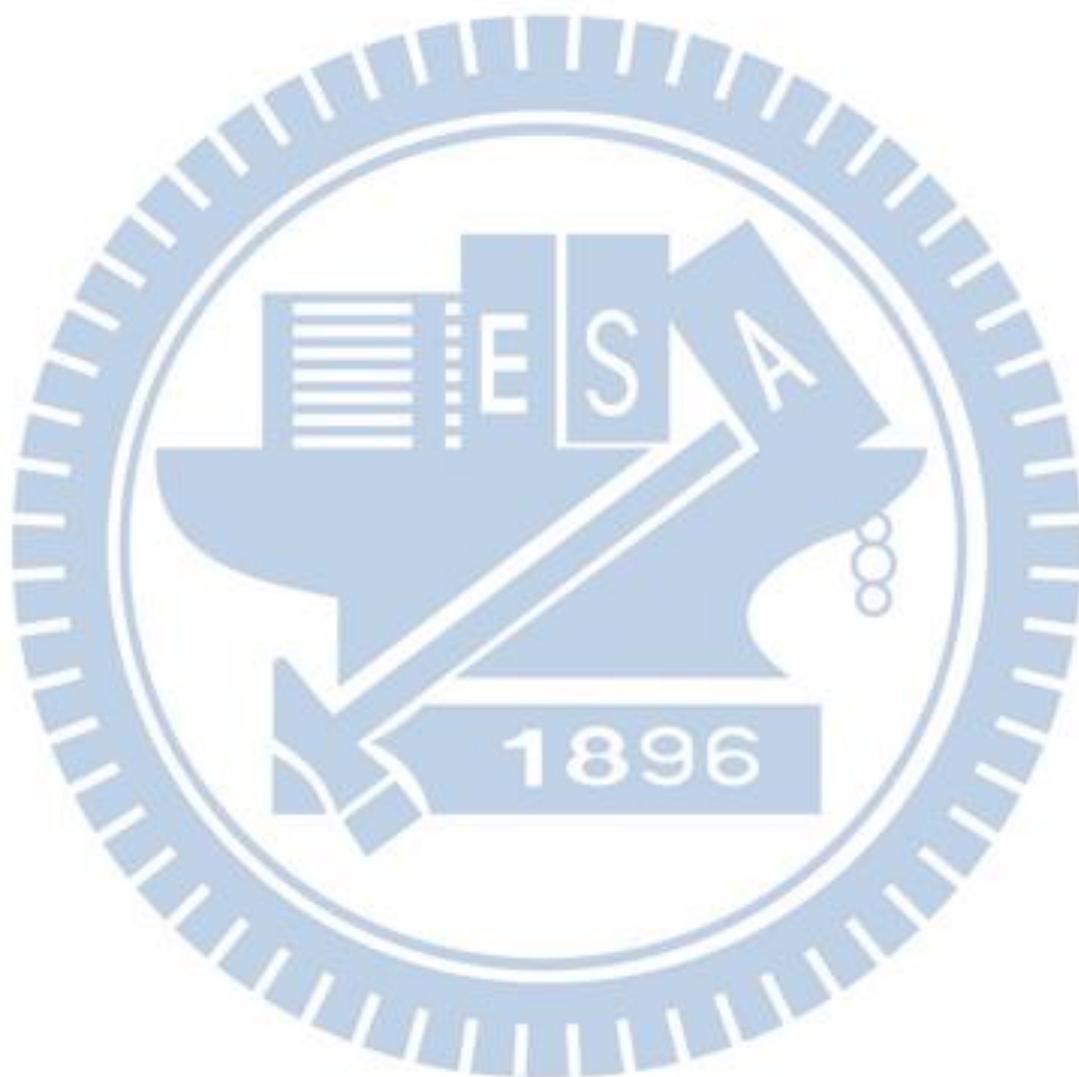
目錄

摘要	i
Abstract.....	ii
誌謝	iii
目錄	iv
圖目錄	vii
表目錄	ix
一、 緒論	1
1.1 研究背景與動機	1
1.2 研究目的	8
1.3 論文架構	11
二、 文獻回顧	13
2.1 車輛路徑問題	13
2.1.1 車輛路徑問題之種類	14
2.1.2 求解之目標與外包車輛之考量	15
2.1.3 車輛路徑問題之求解方法	15
2.2 分區問題	17
2.3 小結	19
三、 問題描述	21
3.1 研究情境與假設	21
3.2 數學模式符號定義	24
3.3 數學模型	27
四、 本研究所提之基因演算法	32
4.1 編碼	34
4.2 解碼	35
4.2.1 重疊服務區域之解碼	35
4.2.2 顧客服務順序之解碼	38

4.2.3	車輛路徑與使用車輛之決策	39
4.2.4	染色體成本值計算	41
4.3	運算子	41
4.3.1	選擇	42
4.3.2	交配	45
4.3.3	突變	47
4.4	區域搜尋	49
4.5	終止條件	52
五、	實驗測試與結果	53
5.1	數學模式求解	53
5.2	實驗例題設計	54
5.3	基因演算法參數測定	56
5.3.1	選擇法	56
5.3.2	迭代數	57
5.3.3	染色體數量與交配率	58
5.3.4	突變率	62
5.3.5	菁英群組比例	63
5.3.6	收斂分析	65
5.4	實驗數據結果	66
5.4.1	數學模式與基因演算法之比較	67
5.4.2	有無考量重疊服務區域之比較	68
5.4.3	有無使用區域搜尋機制之比較	70
5.5	小結	74
六、	結論與建議	76
6.1	結論	76
6.2	未來研究方向與建議	76
	參考文獻	78
	附錄一、交配率與染色體數量參數測定數據	84

附錄二、有無考量重疊服務區域之數據88

附錄三、有無考量區域搜尋機制之數據90



圖目錄

圖 1.1 車輛路徑安排示意圖	3
圖 1.2 派車模式示意圖	3
圖 1.3 鏈狀重疊服務區域示意圖	5
圖 1.4 廣義型重疊服務區域示意圖	6
圖 1.5 重疊服務區域車輛路徑示意圖 (行政區為例).....	7
圖 1.6 廣義型重疊服務區域示意圖 (行政區為例).....	7
圖 1.7 傳統車輛路徑問題	9
圖 1.8 本研究情境之車輛路徑問題	9
圖 1.9 本研究之研究架構	11
圖 2.1 車輛路徑問題示意圖	13
圖 2.2 多個配送員之車輛路徑問題示意圖	18
圖 2.3 保安人員之巡邏問題示意圖	19
圖 3.1 車輛使用範例之車輛初始配置圖	22
圖 3.2 車輛使用範例之車輛實際使用示意圖	23
圖 3.3 顧客點分布範例	26
圖 4.1 基因演算法架構圖	33
圖 4.2 染色體重疊服務區域段解碼示意圖	37
圖 4.3 染色體顧客服務順序段解碼示意圖	39
圖 4.4 車輛路徑示意圖	40
圖 4.5 輪盤選擇法操作之範例	43
圖 4.6 輪盤選擇法操作之染色體新排序	43
圖 4.7 染色體重疊服務區域段交配示意圖	46
圖 4.8 染色體顧客服務順序段交配示意圖	47
圖 4.9 染色體重疊服務區域段突變示意圖	48
圖 4.10 染色體顧客服務順序段突變示意圖	49
圖 4.11 區域搜尋插入法示意圖	50

圖 4.12 區域搜尋交換法示意圖	51
圖 4.13 區域搜尋倒轉法示意圖	52
圖 5.1 無重疊服務區域之分區方法示意圖	55
圖 5.2 有重疊服務區域之分區方法示意圖	55
圖 5.3 各選擇法所測得之 20 組數據分布盒鬚圖	56
圖 5.4 各選擇法所測得之平均成本值與最佳解平均出現代數折線圖	57
圖 5.5 各迭代數所測得之 20 組數據分布盒鬚圖	57
圖 5.6 各迭代數所測得之 20 組最佳解平均出現代數盒鬚圖	58
圖 5.7 交配率 0.5 下各染色體數量所測得之 20 組數據盒鬚圖	59
圖 5.8 交配率 0.5 下各染色體數量所測得之平均成本值折線圖	59
圖 5.9 各參數組合中所測得之 20 組數據分布盒鬚圖	61
圖 5.10 各參數組合中所測得之平均成本值折線圖	61
圖 5.11 各突變率所測得之 20 組數據分布盒鬚圖	62
圖 5.12 各突變率所測得之平均成本值折線圖	62
圖 5.13 各菁英群組比例所測得之 20 組數據分布盒鬚圖	64
圖 5.14 各菁英群組比例所測得之平均成本值折線圖	64
圖 5.15 收斂分析中各迭代數所測得之 20 組數據分布盒鬚圖	65
圖 5.16 收斂分析中各迭代數所測得之 20 組最佳解平均出現代數盒鬚圖	65
圖 5.17 基因演算法求解 $n = 18$ 、 $k = 3$ 之範例成本與時間分析圖	67
圖 A1.1 交配率 0.6 下各染色體數量所測得之 20 組數據盒鬚圖	84
圖 A1.2 交配率 0.6 下各染色體數量所測得之平均成本值折線圖	84
圖 A1.3 交配率 0.7 下各染色體數量所測得之 20 組數據盒鬚圖	85
圖 A1.4 交配率 0.7 下各染色體數量所測得之平均成本值折線圖	85
圖 A1.5 交配率 0.8 下各染色體數量所測得之 20 組數據盒鬚圖	86
圖 A1.6 交配率 0.8 下各染色體數量所測得之平均成本值折線圖	86
圖 A1.7 交配率 0.9 下各染色體數量所測得之 20 組數據盒鬚圖	87
圖 A1.8 交配率 0.9 下各染色體數量所測得之平均成本值折線圖	87

表目錄

表 3.1 顧客點可行指派區域範例參數表	26
表 3.2 顧客所在區域之二元參數表示範例	26
表 4.1 顧客點可行指派區域範例參數表 (含選項標號).....	34
表 4.2 染色體示意表	35
表 4.3 分區參數與染色體基因值編碼結合示意表	36
表 4.4 染色體重疊服務區域段解碼完成表	37
表 4.5 顧客所屬區域與染色體顧客服務順序段基因值	38
表 4.6 依顧客所屬區域排序後之染色體顧客服務順序段	38
表 4.7 染色體顧客服務順序段解碼完成表	39
表 4.8 車輛路徑安排與所需車輛數示意表	40
表 5.1 範例中各顧客點需求之重量與體積	53
表 5.2 範例中各顧客點間距離參數表	54
表 5.3 交配率 0.5 下各染色體數量對組之 t 檢定結果	60
表 5.4 各突變率對組之 t 檢定結果	63
表 5.5 各迭代數對組之 t 檢定結果	66
表 5.6 基因演算法求解 $n = 18$ 、 $k = 3$ 之範例成本與時間數據表	68
表 5.7 有無考量重疊服務區域之數據 ($n = 31$).....	69
表 5.8 各數量顧客點有無考量重疊服務區域對組之 t 檢定結果	70
表 5.9 各顧客點數量考量重疊服務區域之成本值平均改善幅度比較表	70
表 5.10 有無考量區域搜尋機制之數據 ($n = 31$).....	72
表 5.11 各數量顧客水準不使用與單步驟區域搜尋對組之 t 檢定	73
表 5.12 各數量顧客水準不使用與多步驟區域搜尋對組之 t 檢定	73
表 5.13 各數量顧客水準單步驟與多步驟區域搜尋對組之 t 檢定	74
表 A1.1 交配率 0.6 下各染色體數量對組之 t 檢定結果	84
表 A1.2 交配率 0.7 下各染色體數量對組之 t 檢定結果	85
表 A1.3 交配率 0.8 下各染色體數量對組之 t 檢定結果	86

表 A1.4 交配率 0.9 下各染色體數量對組之 t 檢定結果	87
表 A2.1 有無考量重疊服務區域之數據 ($n = 37$).....	88
表 A2.2 有無考量重疊服務區域之數據 ($n = 43$).....	88
表 A2.3 有無考量重疊服務區域之數據 ($n = 49$).....	89
表 A3.1 有無考量區域搜尋機制之數據 ($n = 37$).....	90
表 A3.2 有無考量區域搜尋機制之數據 ($n = 43$).....	91
表 A3.3 有無考量區域搜尋機制之數據 ($n = 49$).....	92



一、緒論

1.1 研究背景與動機

現今物流業發展迅速，貨物運輸成了近年來最熱門的問題之一，而貨物在越洋、跨洲方面主要依靠貨輪運送，至於陸地上的輸送則為卡車所負責，本研究主要針對陸上卡車配送貨物的路徑問題做深度的探討。在此領域中，物流公司主要之業務為將各顧客所需之貨物由配送場站發送至各顧客需求點，並針對其產生配送模式，目標最小化公司整體配送之成本。而為解決與探討此議題，Dantzig & Ramser (1959)提出車輛路徑問題 (Vehicle Routing Problem; VRP)之概念，車輛路徑問題為模式中每位顧客擁有不同數量之貨物需求，物流公司須利用其車隊安排適當之車輛路徑，目標為在滿足所有顧客需求皆完成配送下，最小化物流公司配送總成本。

車輛路徑問題已被證明為一 NP-hard 問題(Lenstra & Rinnooy Kan, 1981)，因此多數的學者以萬用啟發式演算法(Metaheuristic)為基礎進行各類 VRP 問題之求解，其使用之方法如基因演算法(Thangiah, 1993; Jung & Moon, 2002; Baker & Ayechev, 2003; Ombuki et al., 2006; Frutos & Tohmé, 2012; Elhassania et al., 2014; Wang & Wu, 2017)、禁忌搜尋法(Osman, 1993; Gendreau et al., 1994; Taillard et al., 1997; Barbarosoglu & Ozgur, 1999; Bräysy & Gendreau, 2002; Brandão, J., 2009; Cordeau & Maischberger, 2012)、模擬退火法(Osman, 1993; Van Breedam, 1995; Vincent & Lin, 2015; Vincent et al., 2017)、蟻群優化法(Bell & McMullen, 2004; Chen & Ting, 2006; Yu et al., 2009; Lucká & Piecka, 2011; Yao et al., 2017)、粒子群最佳化(Kachitvichyanukul, 2009; Gong et al., 2012)、變動鄰域搜尋法(Polacek et al., 2004; Kytöjoki, 2007; Schneider et al., 2015)等。除了萬用啟發式演算法外，仍有學者使用可求得各類車輛路徑問題中精確最佳解之算法，如 Laporte & Nobert (1983)使用分枝界限法、Lysgaard et al. (2004)使用分枝切割法與 Gutiérrez-Jarpa et al. (2010)使用分枝價格法等等。由此可見 VRP 雖為一 NP-hard 問題，但因其能貢獻之價值與概念而被廣為使用，故本研究將以此問題作為尋找卡車車輛配送路徑的基礎，並使用萬用啟發式演算法

進行求解。

在物流公司配送貨物之車輛路徑問題中，又可細分為「車輛路徑安排」與「車輛派車模式」兩步驟，首先依據車輛路徑問題之概念針對顧客需求之配送執行路徑之安排，如圖 1.1，後再藉由所安排之路徑確認所需之車輛，是為派車之模式，如圖 1.2。在派車模式中，本研究依據探討之分區情境將使用車輛之派車模式分為三種方式：(1) 使用該區域之公司自有車輛 (2) 使用其他區域之公司自有車輛 (3) 外包租用車輛，其中相關之詳細車輛派車操作模式與作法將於 3.1 節中介紹。

實務上多數物流公司為配送顧客需求貨物進行規劃時，通常在得知顧客需求分布後，直接進行車輛路徑的安排，進而確認車輛派車之模式與物流士之派遣，並未如圖 1.1 和圖 1.2 將顧客點進行區域的劃分，然而未先針對顧客所在地進行分區而直接安排車輛路徑，將導致物流士沒有固定服務之區域，意即物流士每次上班執勤前並不知道其配送目的地為何處，因此此方式較為物流士所排斥，且當物流士對於地理位置與路況環境不熟悉時，找路與繞路皆會造成運輸成本之增加。在 Sahin et al. (2009)的分析中也顯示，陸上的卡車在運輸旅行時，其油料成本約佔總成本之 60%，因此若物流士對於目的地地域不熟悉，找路與繞路所造成之成本可不容小覷。甚者 Groër et al. (2009)也提出物流士與顧客建立良好關係對於物流公司有正面之效益，倘若物流士每天服務同一區域之顧客，雖說顧客並不會每天皆有貨物需求，但久而久之對於該名物流士也會較為熟悉，然而此舉也將為公司留住該名顧客形成正面之效果，由此可見分區對於物流公司之重要性，因此本研究將以分區之形式探討車輛路徑之安排。

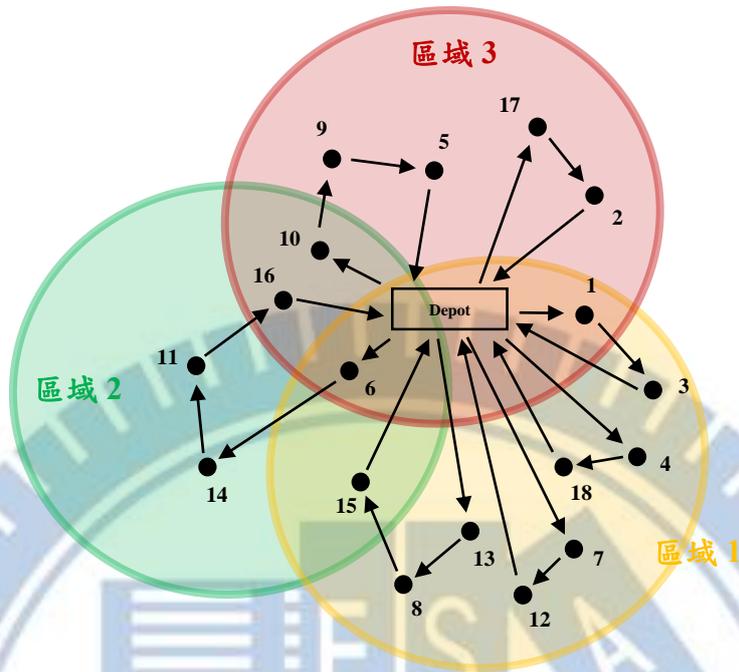


圖 1.1 車輛路徑安排示意圖

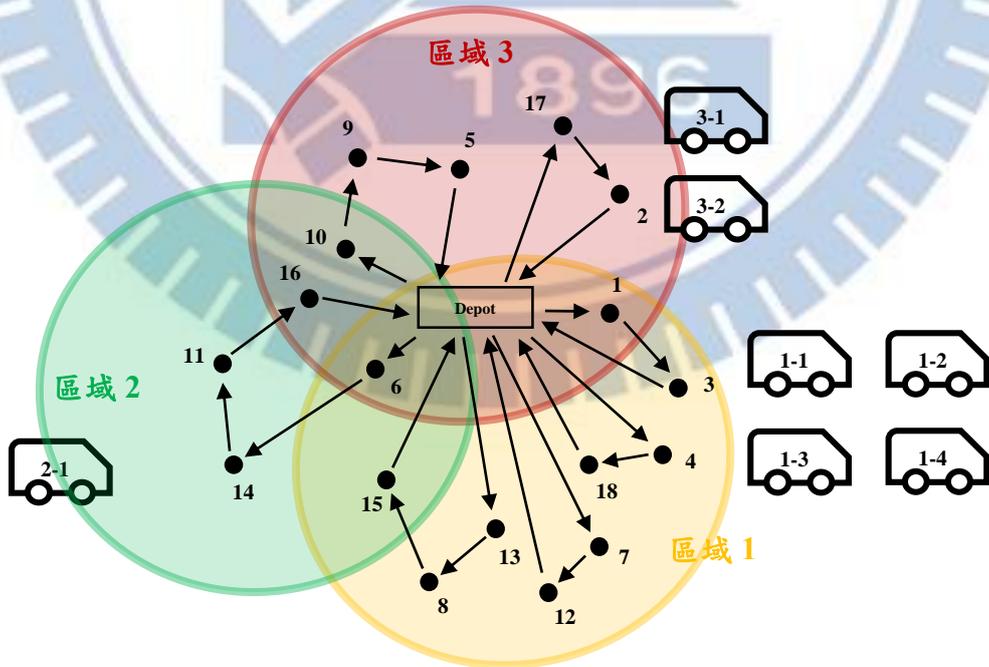


圖 1.2 派車模式示意圖

分區問題在多個領域中皆有其應用，如 Bozkaya et al. (2003)在政治領域進行候選人選區的規劃，Camacho-Collados et al. (2015)則是針對警察局的管轄區域進行劃分，而 Caro et al. (2004)主要分析學校學區的設計，至於在物流領域方面有 Haugland et al. (2007)、Jarrah & Bard (2012)、Lei et al. (2012)與 Lei et al. (2016)考量在顧客需求隨機產生的情況下，針對車輛路徑問題進行分區的設計，另 Prischink (2016)與 Lin et al. (2017)則是先依顧客需求點進行分區與分組之規劃，並在各個區域中分別安排保安人員的巡邏路徑與家庭醫護領域的送餐規劃。而至於本研究則主要架構在已知區域劃分的情況下，依據各個顧客所屬之區域進行分類，並針對每個區域分別安排車輛之路徑。

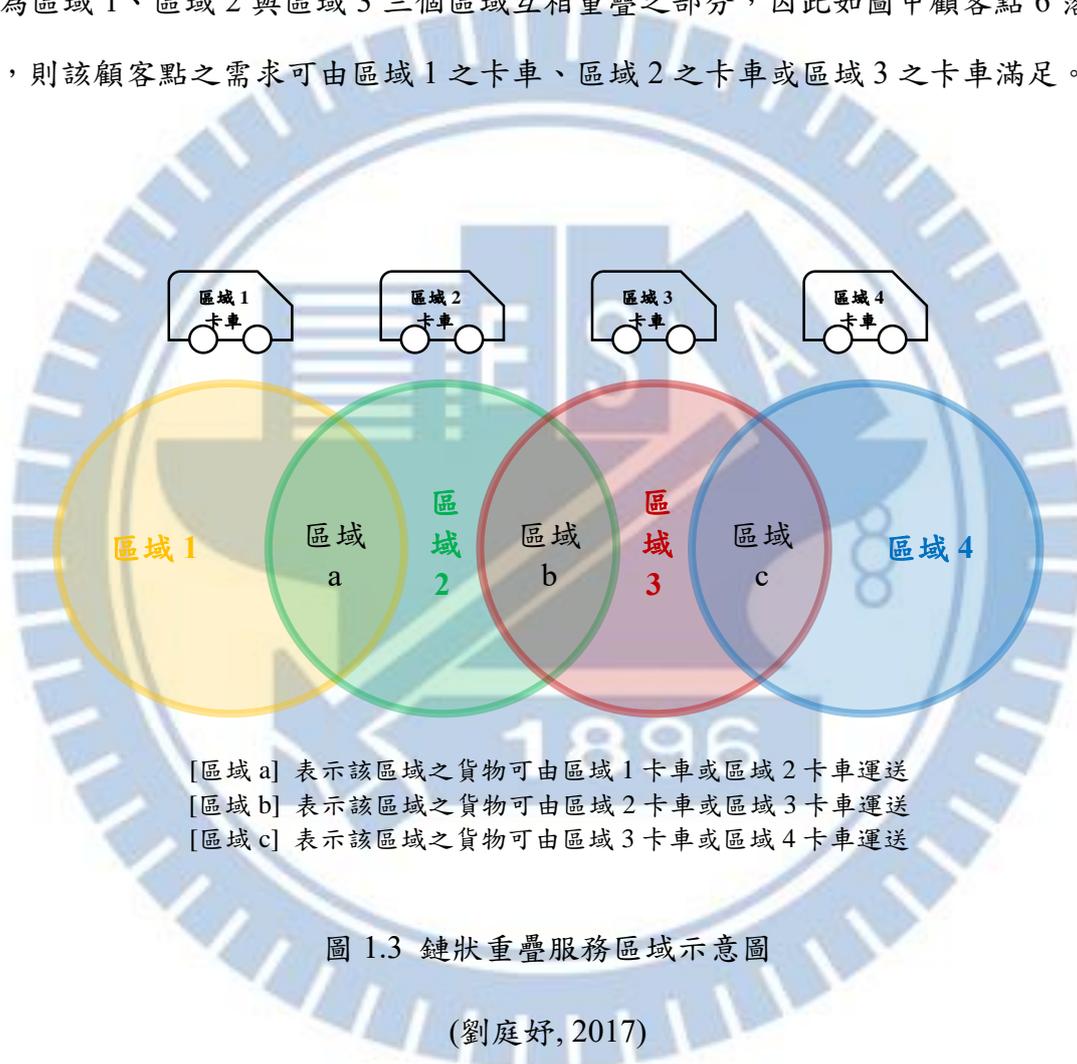
物流公司採用分區架構便於公司管理之業務，主因為考量駕駛人對於各區域之熟悉度不同，因此藉由分區可降低意外發生之風險並提升貨物配送之效率，然而在區域劃分上，物流公司通常依據當地行政區劃或交通道路作為上述分區模式之依據，但此類分區方式卻過於僵化，舉例來說，行政區劃常以河流、溪流或道路主要幹道作為分界，因此常形成橋樑兩端或道路兩側分屬不同之行政區劃，造成兩鄰近顧客點之路徑安排上須以不同區域之卡車車輛進行服務，導致車輛路徑安排受限，亦使車輛無法靈活運用而增加配送成本。因此本研究思考著藉由劉庭好(2017)所提出之「重疊服務區域」概念解決分區過於僵化之問題，探討在為顧客需求點進行分區的架構下，又能彈性安排車輛路徑之方式，是否能有效降低物流公司成本上之問題？

故本研究除了將「車輛路徑問題」與「分區問題」做結合外，亦考量「重疊服務區域」，劉庭好(2017)所提出之「重疊服務區域」概念為一鏈狀分區重疊之形式，各區域為一區接著一區之長條形規劃，其中最大之重疊服務區域數量為 2，意即相鄰之兩區域的卡車車輛所負責配送之區域具有重疊的部分，若顧客點所在之位置恰巧落在這重疊區域之中，則物流公司可從兩區域之卡車車輛擇一為其服務，端看兩區域間之卡車裝載可行性與車輛的旅行成本決定，如圖 1.3 所示。

此一概念的提出增加了卡車路徑安排上的彈性，也增加了卡車空間的使用率，是個具有經濟效益的概念(劉庭好, 2017)，因此本研究將此納入考量，並以較廣之區域面相作

探討，進而以多個區域相互重疊進行研究，且不限制服務區域重疊之數量，並將此類重疊服務區域定義為「廣義型重疊服務區域(General Overlapping Service Regions)」。

此外，與劉庭好(2017)另一不同之處為在本研究中每個區域內亦不限制車輛路徑之數量，即每個區域被允許擁有多條路徑，如圖 1.4，每個區域擁有多輛卡車為其服務，以卡車集合表示之，而圖中區域 a~區域 d 為重疊服務區域，區域 a、b、c 為兩區域重疊，至於區域 d 則為區域 1、區域 2 與區域 3 三個區域互相重疊之部分，因此如圖中顧客點 6 落於區域 d，則該顧客點之需求可由區域 1 之卡車、區域 2 之卡車或區域 3 之卡車滿足。



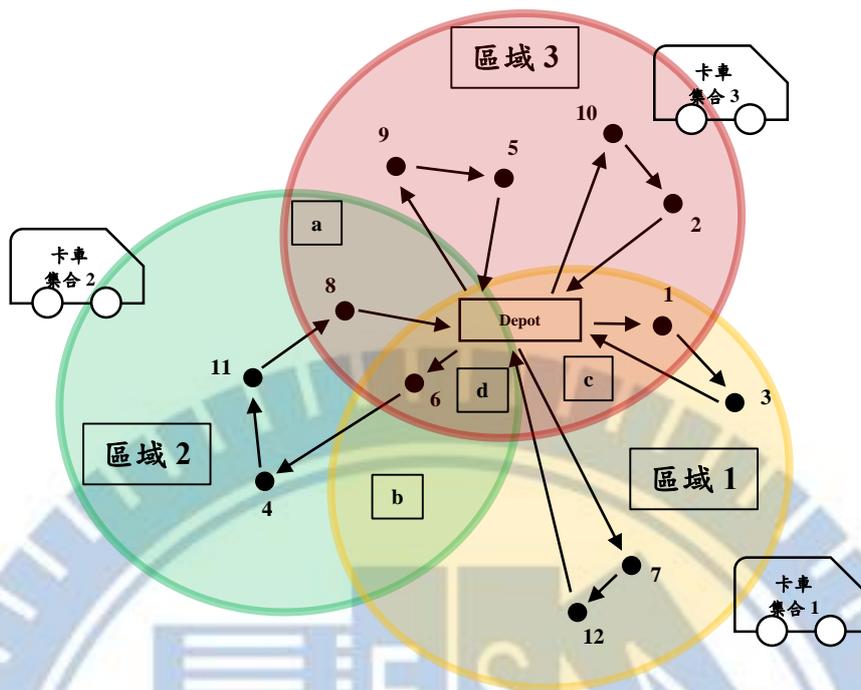
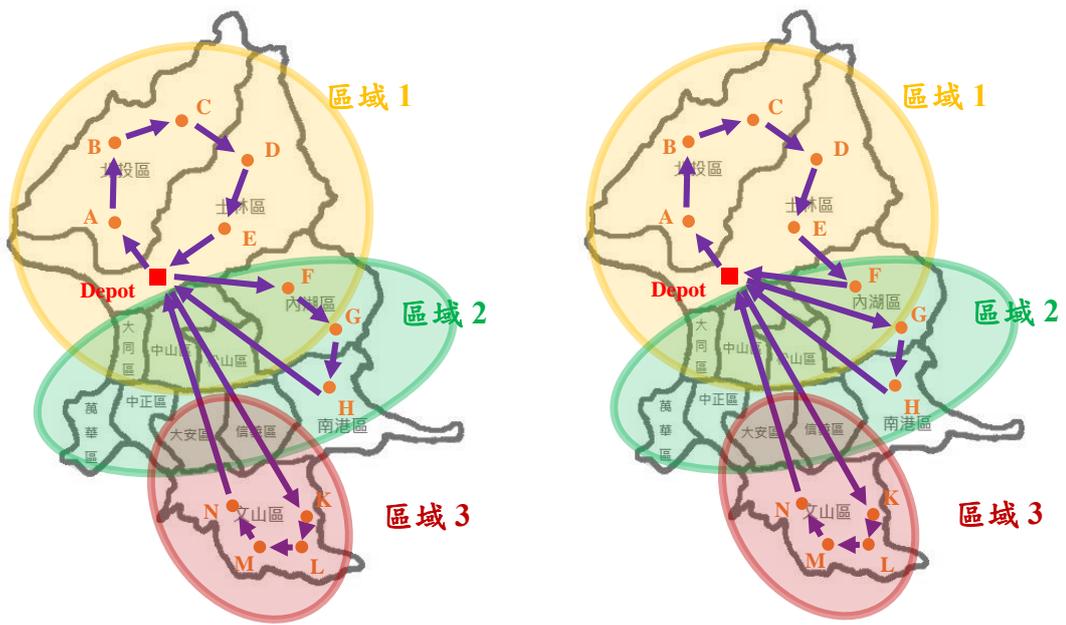


圖 1.4 廣義型重疊服務區域示意圖

重疊服務區域之概念延伸自傳統之分區模式，以現實層面之台北市行政區劃分區為例，圖 1.5 中大同、中山、松山與內湖四區屬於區域 1 與區域 2 重疊區域，因此其中之顧客點可於區域 1 之車輛與區域 2 之車輛中擇一為其服務，另大安區與信義區亦然，可於區域 2 之車輛與區域 3 之車輛中擇一服務。又以圖中之 F 點為例，其位於屬重疊區域之內湖區，因此可由區域 1 之車輛或區域 2 之車輛為其服務，故有圖中(A)與(B)兩種路徑安排模式。在劉庭好(2017)所測試的例題中，考量重疊服務區域之模式確實增加了 4%~6%的卡車貨櫃裝載率，因此得證此舉確實能有效利用卡車剩餘之空間，並藉由減少配送車輛之數目達到降低成本之目的。另本研究為更貼近現實層面，將「鏈狀重疊服務區域」延伸為「廣義型重疊服務區域」如圖 1.6 所示。



(A)

(B)

圖 1.5 重疊服務區域車輛路徑示意圖 (行政區為例)

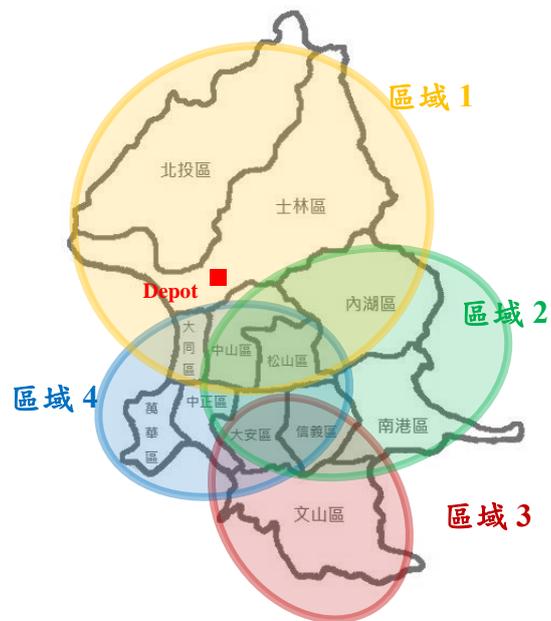


圖 1.6 廣義型重疊服務區域示意圖 (行政區為例)

為同時考量「車輛路徑問題」、「分區問題」與「廣義型重疊服務區域」三構面，本研究將問題定義為「考量重疊服務區域與容量限制之車輛路徑問題」，在已知分區架構與顧客貨物之需求量的情況下，尋求物流公司各區域中每輛卡車的最佳路徑模式。

1.2 研究目的

本研究將針對「考量重疊服務區域與容量限制之車輛路徑問題」提出一數學模式與一演算法，同時考量廣義型重疊服務區域之分區問題與卡車配送路徑安排之車輛路徑問題，目的為最小化顧客需求之貨物配送總成本，為物流公司之管理者提供一有效且經濟的配送模式。首先在車輛配送路徑之安排上必須考量顧客需求貨物裝載之可行性，本研究除了採用有容量限制的車輛路徑問題(Capacitated Vehicle Routing Problem; CVRP)，考量需求重量與卡車載重限制之概念，另也針對貨物體積做限制，意即考量需求貨物之總體積與卡車之容積限制，藉以在重量與體積兩方面確保貨物裝載之可行性。然而另一方面在考量顧客分區時，透過前述廣義型重疊服務區域的概念，增加卡車車輛路徑安排上的彈性。本研究與傳統車輛路徑問題之主要差異如圖 1.7 和圖 1.8 所示，圖 1.7 為傳統車輛路徑問題，圖 1.8 為本研究之情境，由此可見本研究除探討廣義型重疊服務區域之架構，亦考量各分區中車輛使用之模式。

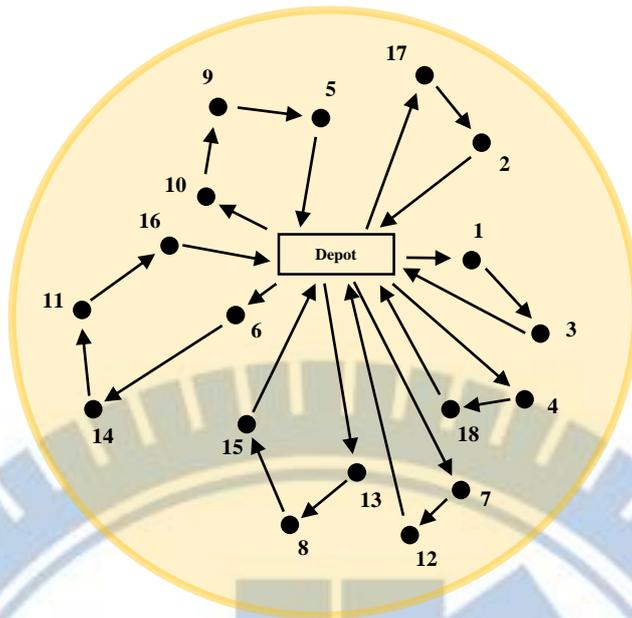


圖 1.7 傳統車輛路徑問題

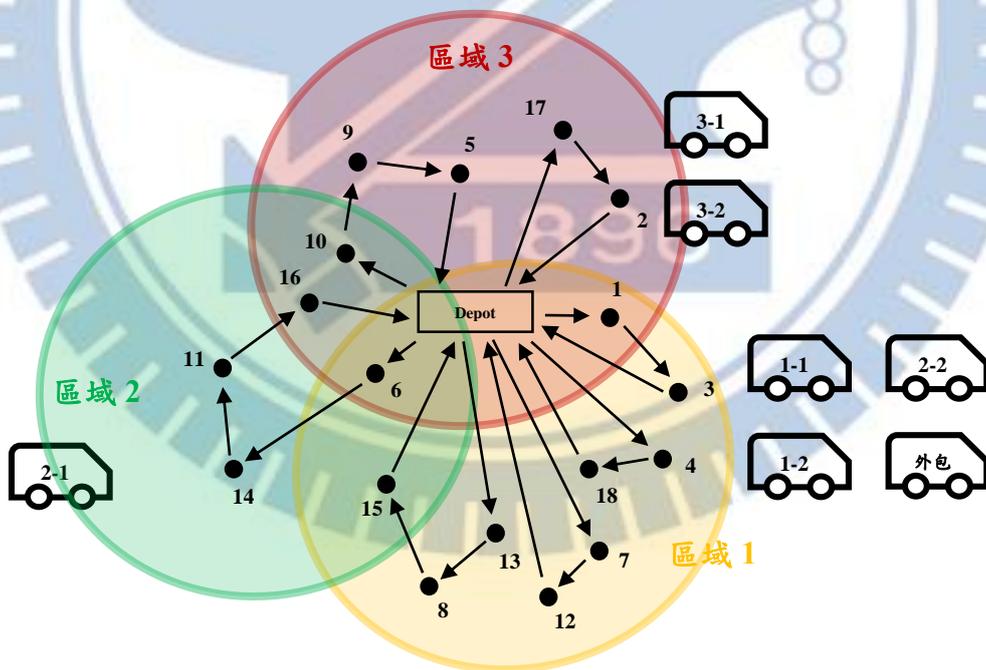


圖 1.8 本研究情境之車輛路徑問題

在求解方法方面，本研究採用之數學模式主要參考 Prischink (2016) 使用之分區模型與延伸自劉庭妤(2017)之廣義型重疊服務區域概念結合，再針對本研究之內容與假設做調整，最後與車輛路徑問題之基本模型整合，並針對卡車車輛之裝載重量與體積做限制，另依據後續 3.1 節介紹之車輛使用模式將使用卡車車輛之成本代入目標式中考量，藉以更貼近現實層面，而詳細之數學模型將於 3.2 節與 3.3 節介紹。

另由於車輛路徑問題為 NP-hard 問題，在大範圍之題目與情況下，使用數學模式求解較為耗時且效率不佳，況且本研究為車輛路徑問題之延伸類型，因此將使用萬用啟發式演算法進行求解，而考量本研究所涉及之廣義型重疊服務區域之資料結構，認為基因演算法之特殊的編碼方式可直觀的描述顧客所屬之分區，且其程序上之操作也應易與廣義型重疊服務區域整合，因此本研究在求解方法上採用基因演算法，並且於演算法中增設一區域搜尋機制加強鄰域解之搜尋能力，期望尋找物流公司卡車配送問題之最佳車輛路徑安排模式，解決考量重疊服務區域與容量限制之車輛路徑問題。最後，參照車輛路徑問題之標竿題庫中例題產生之方式，設計適合本研究使用之範例進行情境分析，驗證本研究所提出之概念與演算法架構之使用效能。

本研究所探討之問題與過去文獻主要之差異為：

- (1) 以廣義型重疊服務區域之架構探討分區之車輛路徑問題，並分析各區域中之車輛使用模式與成本。
- (2) 在有容量限制之車輛路徑問題中，通常僅以貨物重量作為裝載之限制，而本研究以重量與體積作雙方面之考量，藉以更加貼近現實層面。
- (3) 提出一數學模型與一基因演算法用以描述與求解本研究所提之問題，並於基因演算法中增加一區域搜尋機制，加強演算法之搜尋能力。

1.3 論文架構

本研究之各章節架構如圖 1.9 所示：



圖 1.9 本研究之研究架構

(1) 緒論

本章節先介紹物流公司配送顧客需求貨物之方式，並對本研究會探討的領域與目的做初步的介紹，最後概述本篇論文各章節之架構。

(2) 文獻回顧

此章節將蒐集近年來探討路徑與分區問題之文獻，藉以了解此領域之研究類型，並針對各文獻之探討點與方法做比較。

(3) 問題描述

明確定義研究主題「考量重疊服務區域與容量限制之車輛路徑問題」，並深入描述問題之情境與假設，最後提出適合本研究之數學模型。

(4) 求解方法

詳述本研究所提出基因演算法之操作流程與步驟，並針對各程序之意涵做深入的探討。

(5) 實驗測試與結果

將本研究所提出之數學模式與基因演算法分別以 C++ 撰寫程式，再套用例子做比較，並分析此研究之結果。

(6) 結論與建議

彙整本研究之結果，並分析所提出之數學模式與演算法的益處，總結此論文之貢獻與不足，並建議未來之研究方向。



二、 文獻回顧

2.1 車輛路徑問題

車輛路徑問題(Vehicle Routing Problem; VRP)最早由 Dantzig & Ramser (1959)提出，為模式中需求量不一的每位顧客進行為數不同之貨物需求的配送，物流中心須利用其車隊安排適當之車輛路徑，目標為在滿足所有顧客需求皆完成配送下，最小化物流公司配送總成本，然而車輛路徑問題有其基本條件之限制：

- (1) 每輛車必須從場站(Depot)出發並返回場站
- (2) 每個顧客點僅允許被服務一次且每條路徑僅能指派一輛車
- (3) 每個顧客都必須完成服務

除上述限制外，在派遣車輛的過程中，亦須注意車輛之容量限制，而相關車輛路徑問題示意圖如圖 2.1 所示，A~L 每個顧客點都有被卡車車輛服務到，且都恰好服務一次，每輛卡車也都由場站出發並回到場站。

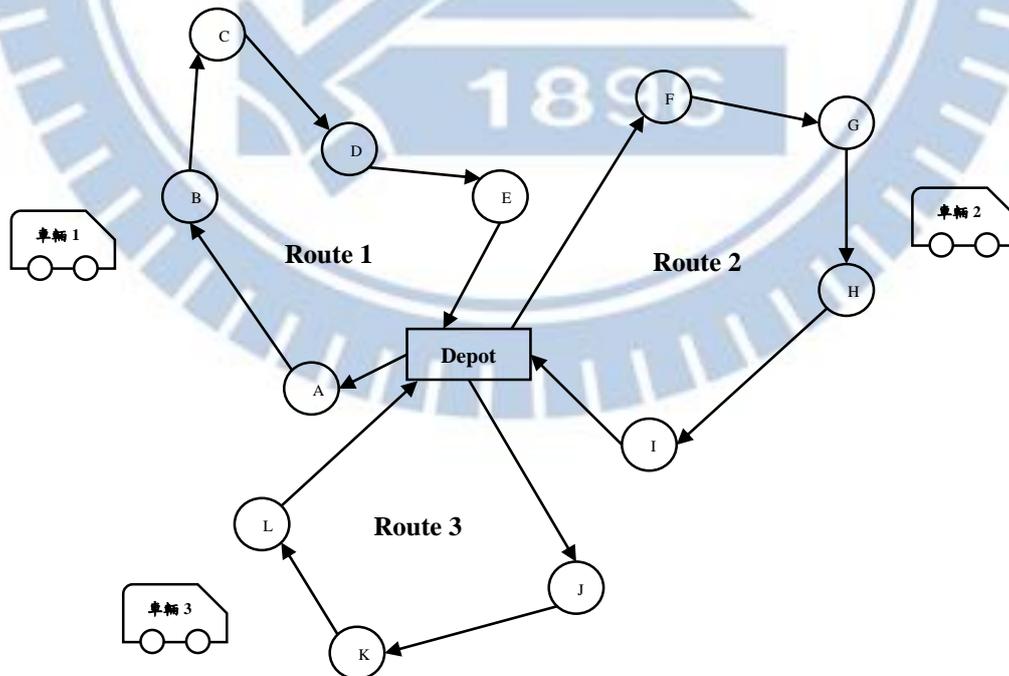


圖 2.1 車輛路徑問題示意圖

2.1.1 車輛路徑問題之種類

傳統車輛路徑問題僅考量車輛容量(Capacity)之限制，且為單場站(Single-depot)、同質車(Homogeneous fleet)之模型，不考慮時間窗(Time windows)、週期性(Period)、隨機需求(Stochastic demand)與分割交貨(Split deliveries)等問題，並僅於配送(Delivery)與收貨(Pickup)間擇一考量，然而隨著時間演進，各領域學者為求更貼近現實面，越來越多種類的車輛路徑問題之延伸類型被提出(Laporte, 2009)，而 Eksioglu et al. (2009)、Lahyani et al. (2015)與 Braekers et al. (2016)也將各類型之車輛路徑問題與其求解方法做詳細地整理與歸納，其中大致上可分為 12 大類：有容量限制之車輛路徑問題(Capacitated VRP; CVRP)、時窗限制之車輛路徑問題(VRP with time windows; VRPTW)、週期性之車輛路徑問題(Period VRP; PVRP)、同時收送貨之車輛路徑問題(VRP with pickup and delivery; VRPPD)、回程取貨之車輛路徑問題(VRP with backhauls; VRPB)、分割配送之車輛路徑問題(Split deliveries VRP; SDVRP)、異質車隊之車輛路徑問題(Heterogeneous fleet VRP; HFVRP)、多場站之車輛路徑問題(Multi-depot VRP; MDVRP)、隨機顧客需求之車輛路徑問題(VRP with stochastic demand; VRPSD)、動態車輛路徑問題(Dynamic VRP; DVRP)、綠能車輛路徑問題(Green VRP; GVRP)與開放式車輛路徑問題(Open VRP; OVRP)。

其中最常見的類型為有容量限制之車輛路徑問題，CVRP 與傳統 VRP 之差異為其增加了卡車裝載重量之限制，又因此限制容易與 VRP 結合之特性且貼近於現實層面，根據 Braekers et al. (2016)在 2009 年至 2015 年間的統計，約有 90%的文獻將 CVRP 列入其研究考量之限制當中，也因此現今在多數文獻中並未特別強調與區分 CVRP 與傳統 VRP 之差異，反而將之視為一體，故本研究也將 CVRP 之限制做為卡車車輛路徑安排上裝載之考量，並且另外加入體積做為卡車裝載之雙重限制，意即顧客需求之貨物必須同時滿足重量裝載可行與體積裝載可行才可將之放入卡車車輛之中，藉以作為更嚴謹之裝載限制。

然而另外可以發現本研究所考量之分區問題與目標式中所含之外包車輛模式均不在文獻常見之分類當中，因此本研究將於 2.1.2 節介紹外包車輛之模式，另分區問題之

回顧將於 2.2 節中詳述。

2.1.2 求解之目標與外包車輛之考量

本研究所考量之成本函數目標式包含自有車輛使用成本、外包租用車輛成本與車輛路徑成本三個部分，其中車輛路徑成本於車輛路徑問題中最常被列入目標式做考量，其比例約占 92%，另探討車輛相關成本之文獻則約佔 38% (Braekers et al., 2016)，然而本研究又將車輛相關成本細分為自有車輛使用成本和外包租用車輛成本，其中外包租用車輛主要參照 Chu (2005)所提出之 Truckload and Less-Than-Truckload Problem 概念以及 Bolduc et al. (2008)提出之兼具自有與委外之車輛路徑問題(Vehicle Routing Problem with Private Fleet and Common Carrier; VRPPC)。

而本研究在外包架構之考量與文獻仍有些許不同，在 VRPPC 之模式下，物流公司不需規劃委外之車輛路線，其類似為將顧客點外包給其他公司之模式，意即該外包顧客點全權交由其他公司所負責；但本研究之外包考量方式為向其他公司租用卡車，用以服務模式中因車輛不足所無法服務之顧客，因此在本研究中仍需考量外包租用車輛之行駛路徑，故亦可將本研究之 VRP 模式視為車輛使用數量不限之情況，僅在計算總成本時需考量使用公司自有車輛與外包租用車輛之成本間的差異，因此在本研究所提出之演算法求解過程中，不會有不可行解之產生，外包車輛僅會因成本較高造成可行解之優劣性差異。

2.1.3 車輛路徑問題之求解方法

車輛路徑問題已被證明為一 NP-hard 問題(Lenstra & Rinnooy Kan, 1981)，故當模式中所涉及之範圍與領域較為龐大時，使用精確求解之方式顯得耗時且效率較為不佳，因此發展出以啟發式演算法(Heuristic)進行車輛路徑問題之求解，目標於短時間內求得可

行解，然而傳統之啟發式演算法容易陷於區域最佳解，因而在 90 年代開始，越來越多學者提出萬用啟發式演算法(Metaheuristic)求解各種車輛路徑問題及其衍生類型(Cordeau et al., 2002; Ombuki et al., 2006)，其中傳統啟發式演算法與萬用啟發式演算法之差異為萬用啟發式演算法多了一可跳脫區域最佳解之程序，藉以增加搜尋求解之廣度，據 Braekers et al. (2016)之統計，2009~2015 年間約有 70%之學者使用萬用啟發式演算法進行車輛路徑問題之求解，而使用精確解法與傳統啟發式演算法之文獻僅占約 17%與 10%，由此可見萬用啟發式演算法已是近年求解 VRP 問題之主流，而在萬用啟發式演算法中最常見用於 VRP 問題之求解方法有禁忌搜尋法(Tabu Search)、模擬退火法(Simulated Annealing)與基因演算法(Genetic Algorithm) (Tan et al., 2001)。

然而現今車輛路徑問題之標竿題庫範例之最佳解多數由禁忌搜尋法所保持(Cordeau et al., 2002; Baker & Ayechev, 2003)，但 Thangiah (1999)也表示基因演算法在時窗限制(VRPTW)、多場站(MDVRP)等車輛路徑延伸問題有成功的表現。另 Tan et al. (2001)談及在禁忌搜尋法、模擬退火法與基因演算法三者之中，模擬退火法擁有最快速之求解時間，而至於禁忌搜尋法與基因演算法則擁有較好之解，但禁忌搜尋法之求解時間約為模擬退火法之 20 倍、基因演算法之 2 倍，因此基因演算法為質量與時間上良好之折衷選擇。Baker & Ayechev (2003)則針對基因演算法求解車輛路徑問題做深入的研究，根據其統計，基因演算法可以在最短的時間內找到與禁忌搜尋法相近的解，且表示若在基因演算法中另加入簡單的鄰域搜索程序，可大幅優化基因演算法之求解效能，並且不會增加過多之求解時間，在其實驗中所花費之時間也比禁忌搜尋法來的節省，故基因演算法在 VRP 領域是有效且具有競爭力之方法。

由於本研究為車輛路徑問題之延伸類型，且綜合以上所述，並考量本研究所涉及之廣義型重疊服務區域之資料結構，因此本研究在求解方法上將採用基因演算法，並且於演算法中增設一區域搜尋機制期望能更加優化解之質量，而詳細之演算法流程將於第四章詳述。

2.2 分區問題

分區是將模式中之全域，意即物流公司之服務範圍劃分為多個區域，針對每個區域分別求取卡車車輛最佳配送路徑，根據 2.1 節中文獻之回顧可發現在車輛路徑問題之體系中，分區問題之文獻較為少見，因此本節將以分區問題為主體，探討與車輛路徑問題相關之文獻。

文獻中分區問題與車輛路徑問題結合之研究主要建構在分區未知的情況下，探討如何進行區域之設計，且設計區域時以一個區域內的顧客皆由同一車輛提供服務為標準，意即每個分區僅允許擁有一輛車，目標在顧客需求隨機產生之情況下，最小化整體模式之成本，分區問題之重點主要在於區域設計、隨機規劃與預期成本之方式。

Haugland et al. (2007)考慮具有隨機需求之車輛路徑的區域設計問題，顧客之需求為一變量，必須在確定分區之決策後才會顯現出來，故需以預測之方式預期各區域之總路徑成本，藉以用作定義與設計分區之依據。

Bard et al. (2010)主要考量在同時收送貨之車輛路徑問題(VRPPD)之架構下，將顧客進行分區，目標最小化車輛路徑成本與區域數量，由於每個區域僅以一輛車服務所有顧客，因此最小化區域數量亦可視為最小化車輛數，而 Jarrah & Bard (2012)亦考量同時收送貨之車輛路徑問題，然其與 Bard et al. (2010)之主要差異在於顧客需求為隨機發生。

Lei et al. (2012)和 Lei et al. (2016)使用兩階段方式解決隨機顧客的車輛路徑和分區問題，其第一階段先進行分區之決策，第二階段再針對每個區域進行車輛路徑安排之成本估算。

Lin et al. (2017)之研究則主要建構於多個配送員之車輛路徑問題(Vehicle routing with multiple deliverymen)，此問題之概念為將鄰近顧客點進行分組，而各組將會指定一卡車卸貨區，車輛送貨時以組為單位進行需求之配送，各組之需求貨物藉由卡車車輛送達該組卸貨區後，再由車輛之配送員將貨物運送至顧客需求之指定位置，如圖 2.2 所示。Pureza

et al. (2012)指出此問題適用於人口稠密地區，而貨物需求通常為小型零售商品，主因為城市地區由於停車位較為稀少且交通擁擠常使卡車車輛停放困難，故鄰近的顧客通常被視為一集群，卡車車輛以組為單位進行配送，而組內再由配送員將需求送至指定位置。Lin et al. (2017)將此概念應用於家庭護理之送餐問題，送餐車輛將顧客需求之膳食送達社區停車場後，再由護理人員將餐點送至需求之各戶人家，然而此類問題針對分組設計之主要限制為車輛之裝載容量限制與配送員之工時限制。

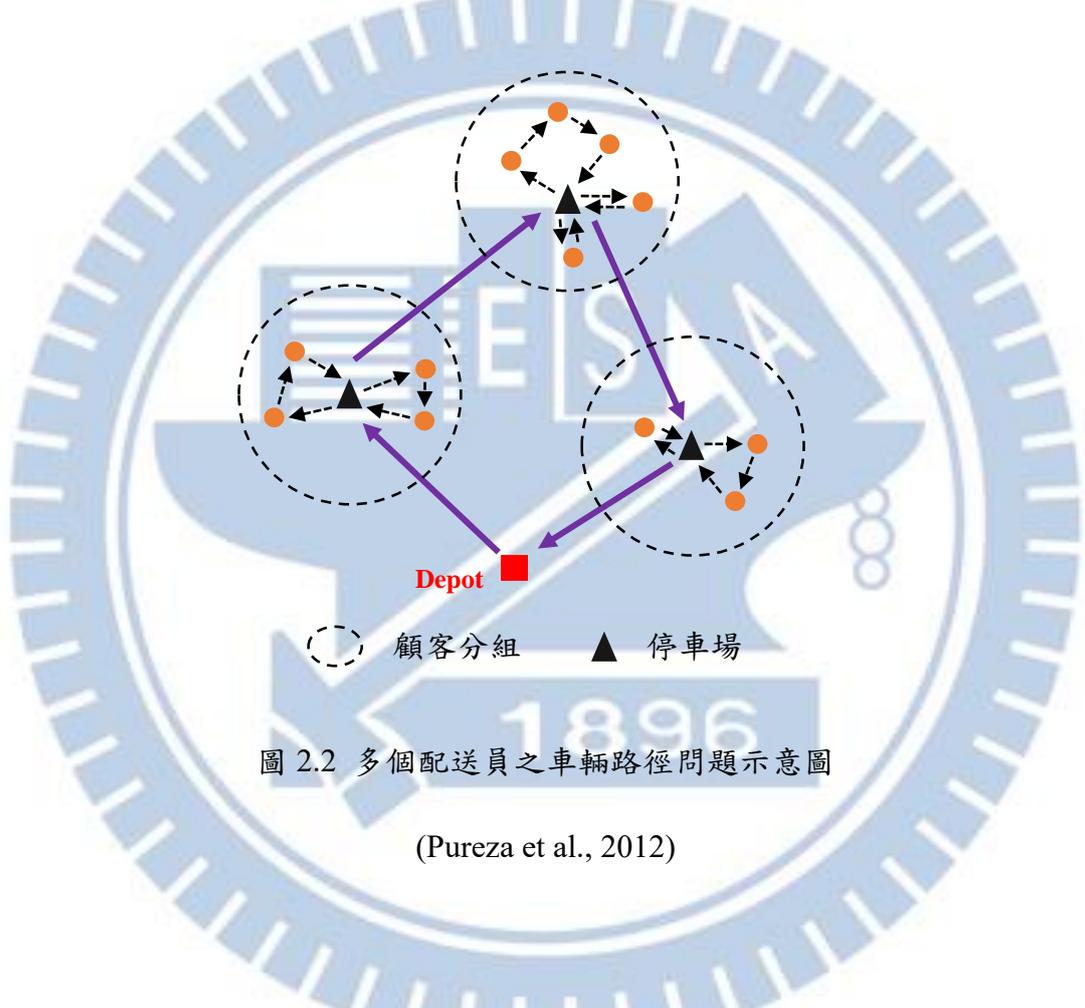


圖 2.2 多個配送員之車輛路徑問題示意圖
(Pureza et al., 2012)

Prischink (2016)考慮保安人員之巡邏問題，此問題可視為分區與路徑兩個部分，分區部分表示需在模式中之計畫週期內將顧客點進行分區，然而針對分區設計之限制為在計畫週期內的每一天，各分區都必須要有一條可行之保安人員巡邏路徑，此路徑之安排上則主要以保安人員之工時與顧客服務時窗作為限制，目標最小化分區數量，如圖 2.3。

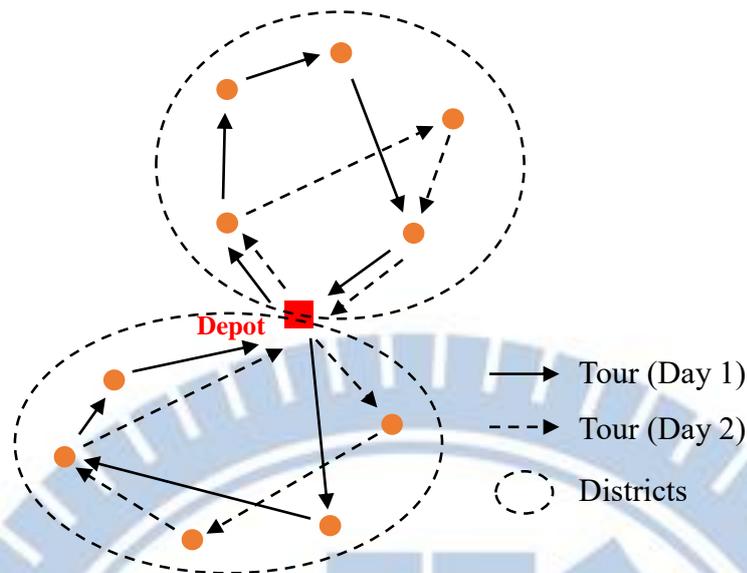


圖 2.3 保安人員之巡邏問題示意圖

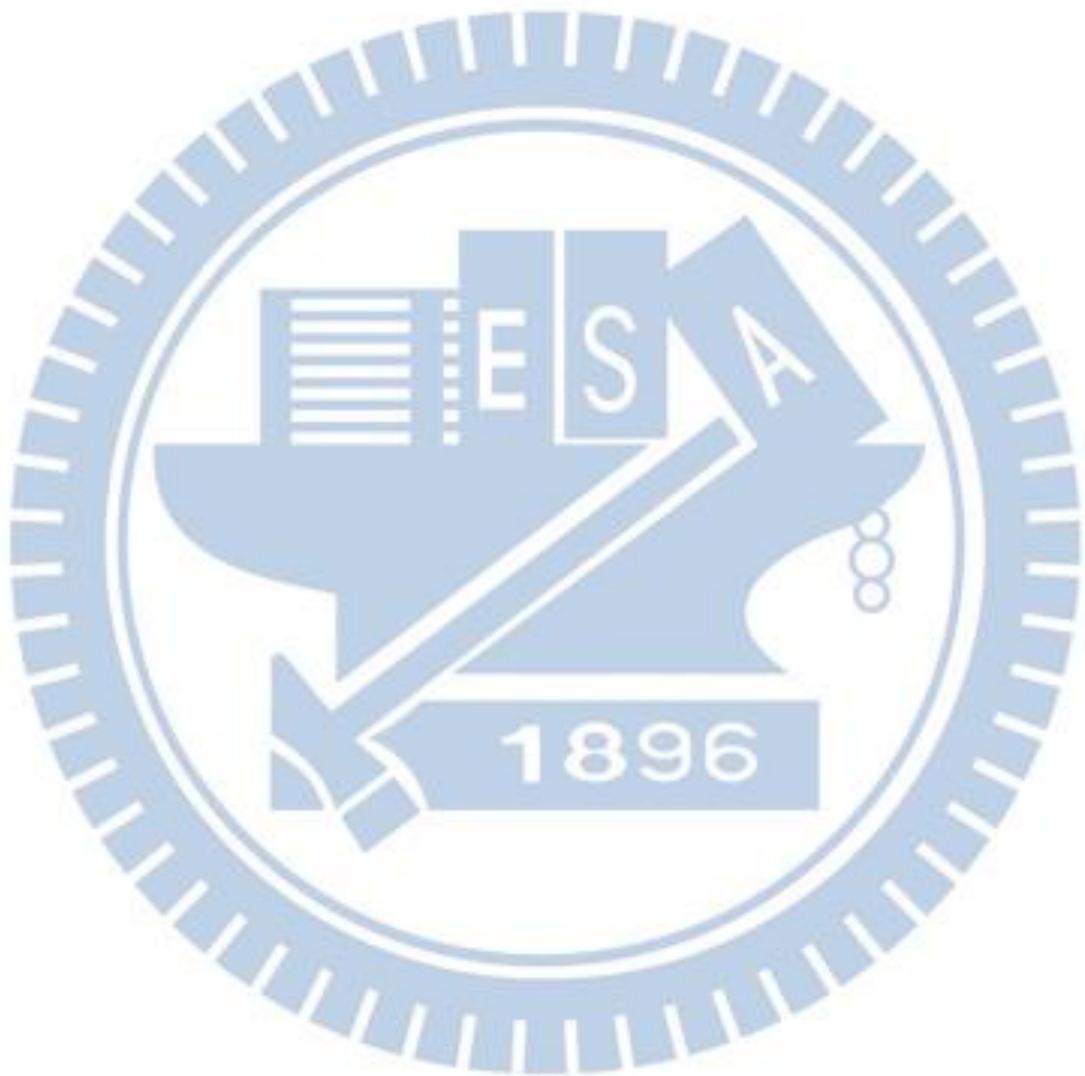
(Prischink, 2016)

由上述文獻中考量之內容可發現此類研究皆著重於如何設計分區，然而本研究主要之考量點為在已知分區架構與顧客點之分組下，如何安排車輛路徑，因此可發現本研究與一般分區文獻所探討之主軸有些許之不同，另本研究更於模式中加入前述之廣義型重疊服務區域之概念。

2.3 小結

觀察以上兩小節所整理車輛路徑問題與分區問題之文獻後，可以發現車輛路徑問題之研究被延伸為許多種類，其所考量之目標也各有差異，並且主要以發展萬用啟發式演算法作為解決問題之工具。而本研究主要以車輛之路徑與使用之成本做為衡量解之優劣之依據，且由於「考量重疊服務區域與容量限制之車輛路徑問題」為車輛路徑問題之延伸，故本研究亦使用萬用啟發式演算法做為求解之方法。除此之外，本研究與以往文獻

之另一差異為在分區問題中所考量與切入之角度不同，並且將導入廣義型重疊服務區域之概念，以此分區模式執行車輛路徑問題之安排，藉以更貼近物流業現實層面，而本研究後續之求解將提出一數學模式與一基因演算法，期望能藉此得到更優之最佳解，提供物流公司之管理者更優異的車輛路徑安排與車輛使用模式之決策。



三、 問題描述

本研究欲解決「考量重疊服務區域與容量限制之車輛路徑問題」，此問題為在分區服務的架構下，探討區域間互相重疊時，物流公司該如何安排卡車車輛路徑才可完成所有顧客需求之配送，其目的為尋找可滿足所有顧客需求之車輛路徑，並使物流公司之總配送成本最小。本章將分為三個小節，3.1 節中描述本研究所建立之研究情境，而 3.2 節將介紹本研究所使用之數學符號，最後 3.3 節詳述所提出之數學模式與考量之相關成本結構。

3.1 研究情境與假設

本研究為考量「廣義型重疊服務區域」與容量限制之車輛路徑問題，為一卡車配送貨物之議題，探討在已知分區之架構與顧客點分布和需求下，如何安排車輛路徑以及自有卡車和外包車輛之派車模式，以達到成本最小化之目標，而在本研究的情境中，決策者為物流公司，其擁有單一倉儲配送場站(Depot)，所有的卡車車輛均自該場站出發配送貨物，並且每一車輛配送完其所負責之顧客需求後皆須返回場站，而物流公司之決策目標為如何安排車輛路徑以滿足所有顧客需求之配送與自有卡車和外包車輛之使用方式。

關於車輛派車之模式，一般而言駕駛與車輛僅會服務其所限定之區域，不會越區提供服務，不過本研究認為儘管各分區有其專屬之車輛，但其車輛仍然屬物流公司所有，因此雖駕駛與車輛跨區服務會增加些許成本，但就常理而言向其他公司外包租用車輛之成本將高於使用物流公司自有車輛跨區服務之成本，故若有區域之車輛不敷使用而其他區域有剩餘車輛之情況下，貿然外包租用車輛而未使用其他區域之公司自有車輛有違常理，因而本研究提出車輛使用模式之三種方式：(1) 使用該區域之公司自有車輛 (2) 使用其他區域之公司自有車輛 (3) 外包租用車輛，此三種方式之車輛使用成本呈遞增模式，因此物流公司必須在上述三種方式間依序選擇。

以圖 3.1 為例，該範例中設有 3 個分區及 18 個顧客點分布，且每個分區擁有公司自有車輛 2 輛，倘若安排之車輛路徑如圖中所示，則區域 1、區域 2 與區域 3 所需之車輛數分別為 4 輛、1 輛和 2 輛，故可以明顯看出區域 1 不足兩輛車，而區域 2 有一輛車閒置，至於區域 3 則剛好兩輛車服務兩條路徑，因此在此範例架構下，區域 1 需要以方式 (2) 之模式向區域 2 挪用一輛公司自有卡車，不過在挪用後會發現區域 1 仍然不足一輛卡車，則此時需要再以方式 (3) 外包租用一輛卡車藉以滿足區域 1 之需求，故在此範例架構下車輛實際使用情形如圖 3.2 所示。

至於方式 (2) 相較於方式 (1) 所增加之跨區服務成本將包含配送效率降低與駕駛員心理情緒之成本，因此造成此類成本難以估算與量化，故在本研究中將忽略上述跨區服務所增加之成本，意即本研究將方式 (1) 與方式 (2) 兩成本訂為相同，但在車輛使用之模式上，則依然在上述三種方式間依序選擇，另相關之成本計算方式將在 3.3 節中詳述。

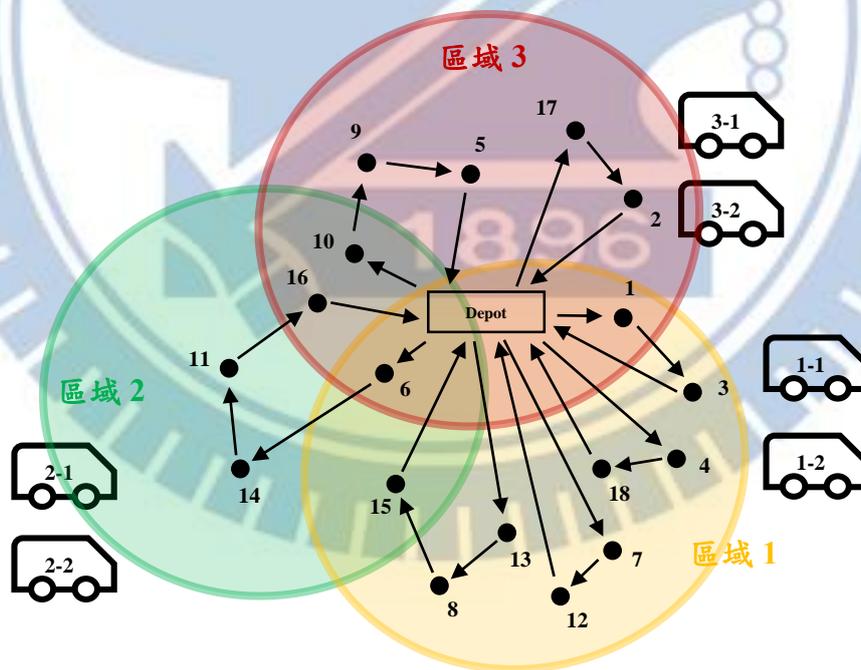


圖 3.1 車輛使用範例之車輛初始配置圖

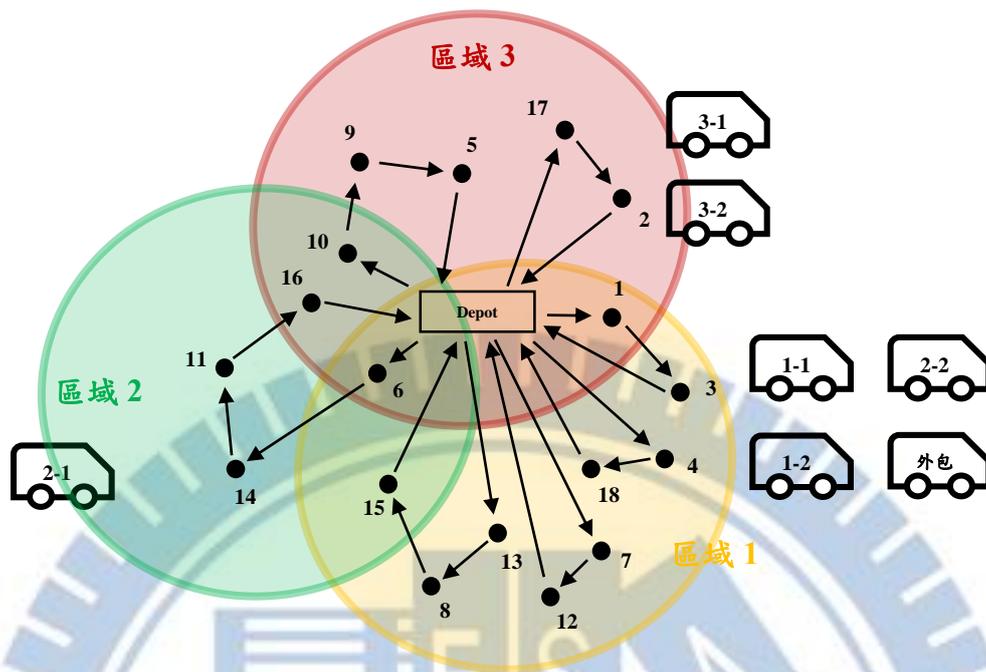


圖 3.2 車輛使用範例之車輛實際使用示意圖

物流公司之配送問題主要發生在物流倉儲配送場站，當公司將每位顧客之需求於倉儲場站集貨完成後，即需安排車輛將各個需求配送到各指定之顧客點，且如上所述，各個顧客點所位在之服務區域不盡相同，又有重疊服務區域之考量，每個顧客需求貨物之體積與重量也有所差異，而如何在區域之限制下有效率地使用卡車，並將貨物需求分送至指定顧客點，為物流公司配送貨物之一大難題。本研究所探討的即是在廣義型重疊服務區域之概念下，如何有效地安排卡車車輛路徑，為物流公司發展出一套能使配送成本最低且符合容積限制之車輛路徑演算法，提供物流公司在決策上之依據，另本研究所考量之基本限制與相關決策情境之假設如下所述：

- (1) 每輛卡車必須由倉儲配送場站出發並且於所有顧客需求配送完成後返回場站。
- (2) 每個顧客都必須完成配送，意即每個顧客之需求皆須被滿足。
- (3) 每個顧客僅能被服務一次，意即同一顧客之需求必須裝載於同一輛卡車上。
- (4) 安排車輛路徑配送貨物時所需之車輛數量不限，意即當物流公司自有車隊不敷使用

時，剩餘之車輛路徑皆以外包之方式租用卡車，直至滿足所有車輛路徑。

- (5) 無論是物流公司自有車輛，抑或是外包租用之車輛，本研究所考量之卡車車輛皆相同，意即每輛卡車擁有相同之載重限制與容積限制。
- (6) 一車輛路徑上所有顧客點之貨物總重量不可超過車輛的載重限制。
- (7) 一車輛路徑上所有顧客點之貨物總體積不可超過車輛的容積限制。
- (8) 單一顧客之需求皆可裝入一空車輛，意即不會發生顧客點需求無法被裝載之情況，另在本研究之模式下，最大之卡車車輛需求數會等於顧客點之數目。
- (9) 本研究之重疊服務區域，考量允許多個區域相互重疊之「廣義型重疊服務區域」。
- (10) 顧客需求配送之車輛路徑安排必須遵守其所屬區域之限制。

3.2 數學模式符號定義

本章節將介紹本研究提出之數學模型所使用之參數與變數，首先模型中使用之參數符號如下所示：

n : 顧客點數量

k : 區域數量

N : 顧客點集合， $N \in \{1, \dots, n\}$

K : 區域集合， $K \in \{1, \dots, k\}$

T : 車輛集合，本研究假設單一顧客之需求皆可裝入一空車輛，因此最大需求車輛數為顧客點之數量， $T \in \{1, \dots, n\}$

D : 物流公司場站(Depot)

Q : 車輛之裝載重量上限

V : 車輛之裝載體積上限

m : 物流公司自有車輛之總數

FC : 自有車輛使用之固定成本

OC : 外包租用車輛之成本

d_i : i 顧客需求之重量, $\forall i \in N$

h_i : i 顧客需求之體積, $\forall i \in N$

C_{ij} : 車輛自顧客點 i 行駛至顧客點 j 之成本, 其中 $C_{ij} = C_{ji} \quad \forall i \in D \cup N; j \in D \cup N$

A_i^r : 分區二元參數, $A_i^r = \begin{cases} 1, & \text{表示}i\text{顧客點可以指派至}r\text{區域} \\ 0, & \text{表示}i\text{顧客點不可指派至}r\text{區域} \end{cases}, \forall i \in N; r \in K$

其中 A_i^r 為一分區二元參數陣列, 1 或 0 分別代表 i 顧客點可否指派至 r 區域, 如表 3.2 所示, 而該表由題目所給定之分區參數表(如表 3.1)轉換而來, 又以圖 3.3 之 3 個分區及 18 個顧客點分布為例, 該範例會給定如表 3.1 之分區參數, 像是顧客點 1 位於區域 1 與區域 3 之重疊服務區域, 在表 3.1 中會列出該顧客點之可行指派區域, 如表中顧客點 1 所對應之數值 1 與 3, 其餘顧客點則依此類推, 另外將表 3.1 轉換至表 3.2 以 A_i^r 二元參數表示時, 同樣以顧客點 1 為例, 其位於區域 1 與區域 3, 故轉換為表 3.2 之參數值為 $A_1^1 = 1$ 、 $A_1^2 = 0$ 和 $A_1^3 = 1$, 同理顧客點 2 位於區域 3, 因此參數值轉換為 $A_2^1 = 0$ 、 $A_2^2 = 0$ 和 $A_2^3 = 1$, 其餘顧客點之參數值則依此類推, 並於表 3.2 所示。

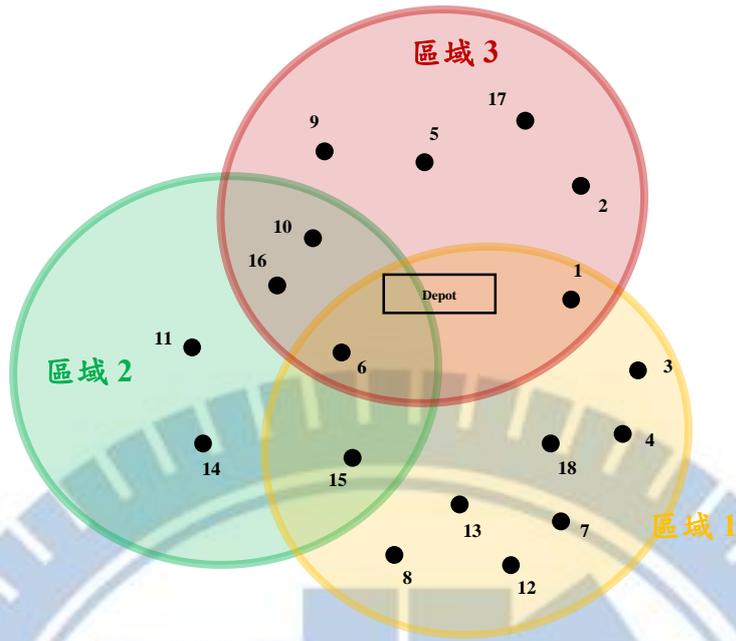


圖 3.3 顧客點分布範例

表 3.1 顧客點可行指派區域範例參數表

顧客 i	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
可行指派區域	區域 1	區域 3	區域 1	區域 1	區域 3	區域 1	區域 1	區域 1	區域 3	區域 2	區域 2	區域 1	區域 1	區域 2	區域 1	區域 2	區域 3	區域 1
	區域 3					區域 2				區域 3					區域 2	區域 3		
						區域 3												

表 3.2 顧客所在區域之二元參數表示範例

區域 \ 顧客 i	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
區域 1	1	0	1	1	0	1	1	1	0	0	0	1	1	0	1	0	0	1
區域 2	0	0	0	0	0	1	0	0	0	1	1	0	0	1	1	1	0	0
區域 3	1	1	0	0	1	1	0	0	1	1	0	0	0	0	0	1	1	0

除此之外，本研究於數學模型中亦使用兩個變數與兩個決策變數，在變數方面為 TC 與 z_i ， TC 主要用於計算物流公司之配送總成本，而 z_i 為一對各顧客點 i 給定之實數變數，其主要為避免車輛路徑問題中出現未經過場站之子迴圈所設之實數變數。另外兩個決策變數方面則如下所示。

首先 x_{ij}^{rt} 為一二元決策變數：

$$x_{ij}^{rt} = \begin{cases} 1, & \text{若 } i \text{ 顧客點與 } j \text{ 顧客點連結，且由區域 } r \text{ 之 } t \text{ 車輛配送} \\ 0, & \text{則否} \end{cases}$$

其中區域 r 之 t 車輛之表示法，如前述所提及本研究假設單一顧客之需求皆可裝入一空車輛，因此模式中最大需求車輛數為顧客點之數量 n ，且又因本研究允許外包車輛之使用方式，因此可視為安排車輛路徑配送貨物時所需之車輛數量不限，意即如發生自有車輛不足，即採用外包車輛方式，直至滿足所有車輛路徑，因此不會發生車輛不足之問題，僅會產生各路徑使用車輛之成本不同之現象，綜合以上所述，本研究為確保車輛數量足夠，在數學模式中將給定各分區擁有車輛數 n 輛，藉以確保各區域內之車輛數量足夠，並且在最小化總成本之情境下，模式會盡可能減少使用之車輛數以達成目標，因此在 x_{ij}^{rt} 上採用區域 r 之 t 車輛之表示法，然而關於各車輛在使用上之成本計算則於3.3節中描述。

本研究另提出一分區二元決策變數 y_i^r ，其中

$$y_i^r = \begin{cases} 1, & \text{若 } i \text{ 顧客點指派至 } r \text{ 區域} \\ 0, & \text{則否} \end{cases}$$

無論顧客點落於重疊服務區域抑或非重疊服務區域，其最終皆只被允許指派至唯一區域，因此 y_i^r 主要用於確認各顧客點被指派之分區。

3.3 數學模型

本研究旨在最小化決策者物流公司之整體配送成本，相關之目標式與限制式如下所

示：

Min TC

$$TC = FC \cdot \sum_{r \in K} \sum_{j \in N} \sum_{t \in T} x_{Dj}^{rt} + (OC - FC) \cdot \max \left(\sum_{r \in K} \sum_{j \in N} \sum_{t \in T} x_{Dj}^{rt} - m, 0 \right) \quad (3.1)$$

$$+ \sum_{r \in K} \sum_{t \in T} \sum_{i \in DUN} \sum_{j \in DUN} C_{ij} x_{ij}^{rt}$$

Subject to

$$\sum_{r \in K} A_i^r y_i^r = 1 \quad \forall i \in N \quad (3.2)$$

$$\sum_{r \in K} y_i^r = 1 \quad \forall i \in N \quad (3.3)$$

$$y_i^r \geq x_{ij}^{rt} \quad \forall i \in N; j \in DUN; t \in T; r \in K \quad (3.4)$$

$$y_j^r \geq x_{ij}^{rt} \quad \forall i \in DUN; j \in N; t \in T; r \in K \quad (3.5)$$

$$\sum_{r \in K} \sum_{t \in T} \sum_{i \in DUN} x_{ij}^{rt} = 1 \quad \forall j \in N \quad (3.6)$$

$$\sum_{r \in K} \sum_{t \in T} \sum_{j \in DUN} x_{ij}^{rt} = 1 \quad \forall i \in N \quad (3.7)$$

$$\sum_{i \in DUN} x_{ip}^{rt} - \sum_{j \in DUN} x_{pj}^{rt} = 0 \quad \forall p \in N; t \in T; r \in K \quad (3.8)$$

$$\sum_{j \in N} x_{Dj}^{rt} \leq 1 \quad \forall t \in T; r \in K \quad (3.9)$$

$$\sum_{i \in N} x_{iD}^{rt} \leq 1 \quad \forall t \in T; r \in K \quad (3.10)$$

$$\sum_{i \in N} d_i \sum_{j \in D \cup N} x_{ij}^{rt} \leq Q \quad \forall t \in T; r \in K \quad (3.11)$$

$$\sum_{i \in N} h_i \sum_{j \in D \cup N} x_{ij}^{rt} \leq V \quad \forall t \in T; r \in K \quad (3.12)$$

$$z_i - z_j + n \cdot x_{ij}^{rt} \leq n - 1 \quad \forall i \in N; j \in N; t \in T; r \in K \quad (3.13)$$

$$x_{ij}^{rt} \in \{0, 1\} \quad \forall i \in D \cup N; j \in D \cup N; t \in T; r \in K \quad (3.14)$$

$$y_i^r \in \{0, 1\} \quad \forall i \in N; r \in K \quad (3.15)$$

$$z_i \in \mathbb{R} \quad \forall i \in N \quad (3.16)$$

公式(3.1)為本研究定義之目標式，其中物流配送成本(TC)包含車輛固定成本(FC)、外包車輛成本(OC)與車輛路徑成本(C_{ij})三個部分，至於總成本計算之方式本研究在第一部分先將所有使用之車輛以車輛固定成本計算，而若有使用外包租用車輛則在第二部分加上外包車輛成本與車輛固定成本之差額，另外第三部分再計算整體路徑之總成本。

公式(3.2)與公式(3.3)為顧客分區指派之限制式，也是本研究與傳統車輛路徑問題之主要不同處，其限制每個顧客點必須指派至剛好一個區域；公式(3.4)與公式(3.5)則是本研究使用之分區與路徑兩個二元決策變數間的關係式，限制必須在顧客點 i 與顧客點 j 被指派至同一區域時，才可連結該條路徑；而公式(3.6)至公式(3.10)為車輛路徑問題的基本限制式，在此一部分本研究與文獻之差異僅在於公式中以增設 $r \in K$ 之方式額外考量了分區之限制，如前述 3.2 節中所提及，以區域 r 之 t 車輛方式限制車輛不可同時服務多個區域，藉以達成分區之效果，其中公式(3.6)與公式(3.7)表示每個顧客點必須被服務剛好一次，公式(3.8)為流量守恆公式，公式(3.9)與公式(3.10)則限制一台車至多僅允許值勤一條路徑；而公式(3.11)為車輛載重限制，公式(3.12)則為車輛裝載體積之限制；至於公式(3.13)為避免模式中出現未經過場站之子迴圈路徑限制；另外 x_{ij}^{rt} 與 y_i^r 為 0-1 二元變數標示於公式(3.14)與公式(3.15)，而 z_i 則為一實數變數於公式(3.16)所示。

本研究將所提出之數學模式應用於 Gurobi 軟體求解，但由於此模式之目標式為非線性之方程，因此本研究將透過一轉換之程序，將目標式以線性方程之方式表達，以便於使用最佳化軟體求解，故本研究將新增一整數變數 w ， $w \in \{0, \dots, n\}$ ，將公式(3.1)轉換如公式(3.17)，並新增一限制式如公式(3.18)所示：

$$TC = FC \cdot \sum_{r \in K} \sum_{j \in N} \sum_{t \in T} x_{Dj}^{rt} + (OC - FC) \cdot w + \sum_{r \in K} \sum_{t \in T} \sum_{i \in DUN} \sum_{j \in DUN} C_{ij} x_{ij}^{rt} \quad (3.17)$$

$$\sum_{r \in K} \sum_{j \in N} \sum_{t \in T} x_{Dj}^{rt} - m \leq w \quad (3.18)$$

公式(3.17)主要將公式(3.1)中外包之車輛數 $\max(\sum_{r \in K} \sum_{j \in N} \sum_{t \in T} x_{Dj}^{rt} - m, 0)$ 改以變數 w 表示，若有使用外包車輛，意即 $\sum_{r \in K} \sum_{j \in N} \sum_{t \in T} x_{Dj}^{rt} - m$ 為正數之情況下，在最小化總成本的架構中，公式(3.18)之等式會成立，則 w 等於外包之車輛數；另倘若未使用外包租用車輛，則 $\sum_{r \in K} \sum_{j \in N} \sum_{t \in T} x_{Dj}^{rt} - m$ 為負值或0，此時 w 取其定義之最小值為0，則目標式不會計算外包租用車輛之成本。本研究以此新增變數與限制式之方式將目標式轉換為線性方程，藉以使用最佳化軟體 Gurobi 求解。

綜合以上所述，本研究將此數學模型中使用之限制式數量與變數數量彙整為函數式所示。首先在限制式方面，公式(3.2)依據其中 $\forall i \in N$ 計算表示其中共有 n 條限制式；而公式(3.3)亦依據 $\forall i \in N$ 為 n 條限制式；至於公式(3.4)之 $\forall i \in N; j \in DUN; t \in T; r \in K$ 則為 $n \cdot (n + 1) \cdot n \cdot k$ 條限制式，然其餘公式(3.5)至公式(3.13)及公式(3.18)之限制式數量則依此計算方式類推，後加總與彙整如公式(3.19)所示：

$$3n^3k + 3n^2k + 4nk + 4n + 1 \quad (3.19)$$

至於在變數數量之計算方面，公式(3.14)之 x_{ij}^{rt} 中為 $\forall i \in D \cup N; j \in D \cup N; t \in T; r \in K$ ，因此其中變數數量是為 $(n + 1) \cdot (n + 1) \cdot n \cdot k$ 個，而其餘 y_i^r 與 z_i 之變數數量計算方式亦同，後經由彙整並加上 1 個 w 變數之數量函式如公式(3.20)所示：

$$n^3k + 2n^2k + 2nk + n + 1 \quad (3.20)$$

由上述二公式可見在本研究之情境中，數學模型最佳化之整體求解複雜度，至於相關之數據實驗結果將於第五章詳述。



四、 本研究所提之基因演算法

本文將提出一解決「考量重疊服務區域與容量限制之車輛路徑問題」之基因演算法 (Genetic Algorithm)，目的為最小化物流公司之配送總成本(TC)。基因演算法之概念主要由 Holland(1975)所提出，其架構多年來也被廣泛用於各個領域，如物流業裝箱問題、旅行推銷員問題、車輛運輸路徑問題、工作排程問題、機器排程問題、設施規劃問題等方面之最佳化都有突出的表現。而基因演算法搜索之方式為多點搜尋，因此較不容易陷入區域最佳解，且透過其染色體編碼之特性可以直觀且容易地描述問題之架構與情境，如本研究應用之顧客點分區與服務順序之現況，可透過基因演算法之調整與轉換，直接並完整地呈現染色體編碼，且其應用達爾文演化論「適者生存」的精神淘汰表現不佳之染色體，意即成本值較高之染色體將遭到淘汰，藉此保留較優良的染色體解，並於解碼後直觀地呈現物流公司所需之決策，因此本研究使用基因演算法作為求解之工具。

本研究所提出之基因演算法相關之架構圖如圖 4.1 所示，第一步先產生初始母體，接著使用 3.3 節中提出之目標式公式(3.1)計算初始母體中每條染色體之成本值，後判斷演算法是否符合終止條件，若否，則選擇所需數量之染色體群進入交配與突變程序，此部分本研究將於六種選擇法中擇一進行染色體之選擇，詳細作法於 4.3.1 節中介紹。而在交配與突變的程序中，由於本研究所設計之染色體擁有兩段代表不同意義之編碼，分別為重疊服務區域段和顧客服務順序段，因兩者基因值之結構不同，故本研究將其分開進行交配與突變，藉以確保不會產生不可行解。最後，本研究另提出一區域搜尋機制，期望能藉此增加演算法搜尋之廣度，提升優質解出現之機率，接著再透過公式(3.1)分別計算區域搜尋前後所產生之染色體子代的成本值，並擇優進入下一世代，且若染色體有優於母代之表現，則取代母代成為現有最佳解，如此反覆迭代執行基因演算法程序直至符合終止條件，最後產出演算法最終解。

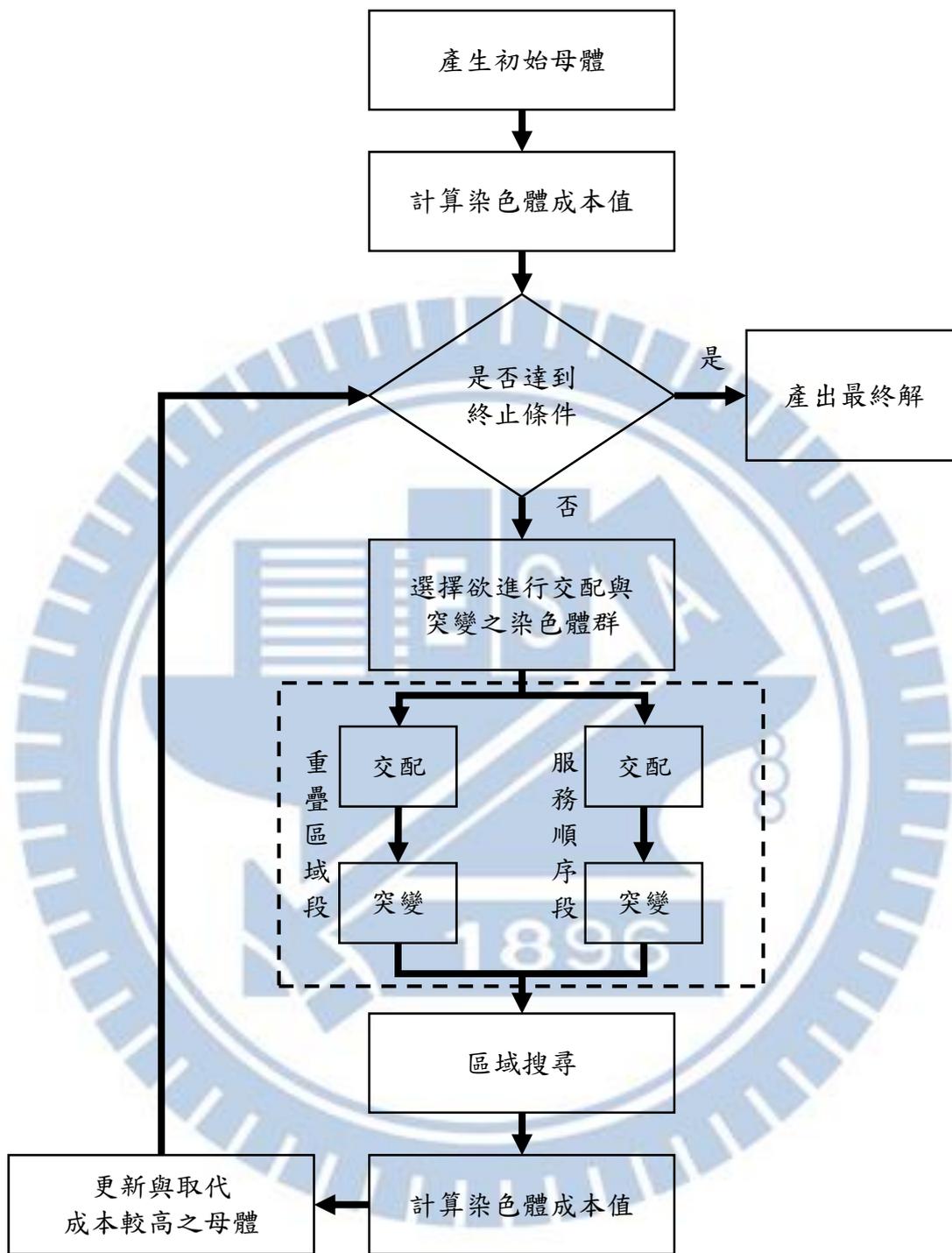


圖 4.1 基因演算法架構圖

本研究基因演算法相關之編碼方式將於 4.1 節中描述，另 4.2 節主要介紹解碼之程序與步驟，而關於基因演算法運算子的詳細內容於 4.3 節中詳述，錯誤! 找不到參照來源。節則是介紹本研究所提出之區域搜尋機制，最後 4.5 節說明本研究針對演算法所使用之終止條件。而為了詳細介紹此基因演算法，本章將沿用圖 3.3 之範例，以 18 個顧客點($n = 18$)與 3 個分區($k = 3$)進行說明，並將範例給定表 3.1 之參數以含有選項標號(#1、#2、#3)之表 4.1 呈現，以利後續演算法之描述。而表中之顧客點所對應之可行指派區域，若僅有選項 1(#1)有數值者，則為非重疊服務區域之顧客點，如顧客點 2；反之則為重疊服務區域之顧客點，如顧客點 1。

表 4.1 顧客點可行指派區域範例參數表 (含選項標號)

顧客 <i>i</i>	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	
可行指派區域	#1	區域 1	區域 3	區域 1	區域 1	區域 3	區域 1	區域 1	區域 1	區域 3	區域 2	區域 2	區域 1	區域 1	區域 2	區域 1	區域 2	區域 3	區域 1
	#2	區域 3				區域 2				區域 3					區域 2	區域 3			
	#3					區域 3													

4.1 編碼

本研究使用之基因演算法將每條染色體分為兩個部分，第一部分為重疊服務區域段，而第二部分為顧客服務順序段，如表 4.2 之染色體示意表所示，其中第一段染色體僅針對落於重疊服務區域之顧客點進行編碼，目的為落於重疊服務區域之顧客點決定分區，而落於非重疊服務區域之顧客則已有所屬之唯一區域限制，故不須參與此部分之編碼。然而於圖 3.3 和表 4.1 可得知此範例有顧客點 1、6、10、15、16 落於重疊服務區域，因此此段染色體僅針對這 5 個顧客點進行編碼，且此段基因值編碼之範圍與該顧客點可供

指派之區域數量相關，以顧客點 1 為例，其有兩個區域選項(#1、#2)可供選擇，因此編碼範圍為 1~2，又以顧客點 6 為例，其有三個區域選項(#1、#2、#3)可供選擇，因此編碼範圍為 1~3，而剩餘之重疊服務區域顧客點則依此類推；至於第二部分之染色體則為顧客服務順序段，為呈現顧客接受服務的先後順序，此段基因值編碼設定為不重複之數值，且其範圍與模式中之顧客點數量相關，以上述範例為例，此段編碼範圍為 1~18。

表 4.2 染色體示意表

顧客 <i>i</i>	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
重疊服務區域	1					2				2					1	1		
顧客服務順序	2	12	3	4	8	1	13	17	6	5	10	14	16	7	18	15	11	9

4.2 解碼

本節將介紹染色體之解碼，意即一條染色體所代表之含意，並藉由解碼程序將染色體轉換回現實案例之決策內容，再以成本值量化染色體之優劣程度，而 4.1 節中所介紹之兩段編碼，其解碼之方式將於 4.2.1 和 4.2.2 兩小節中介紹，而 4.2.3 節則詳述如何依解碼後之染色體決定車輛路徑與使用車輛之方式，最後 4.2.4 節則是計算該條染色體之成本值。

4.2.1 重疊服務區域之解碼

染色體第一部分重疊服務區域之基因值編碼目的為落於重疊服務區域之顧客點決定分區，為了完整呈現其含意，本研究將表 4.1 給定之分區參數列於染色體中兩段基因值編碼之前，如表 4.3 所示，非重疊服務區域之顧客直接列出其所屬之唯一區域，重疊服務區域之顧客則暫時以「重疊」標示，而本節之目的則是透過染色體第一段之基因值

與表 4.1 之分區參數為重疊服務區域之顧客點指派分區。

表 4.3 分區參數與染色體基因值編碼結合示意表

顧客 <i>i</i>	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
給定分區參數	重疊	區域 3	區域 1	區域 1	區域 3	重疊	區域 1	區域 1	區域 3	重疊	區域 2	區域 1	區域 1	區域 2	重疊	重疊	區域 3	區域 1
重疊服務區域	1					2				2					1	1		
顧客服務順序	2	12	3	4	8	1	13	17	6	5	10	14	16	7	18	15	11	9

染色體第一部分重疊服務區域段之基因值代表演算法欲選擇之選項，以圖 4.2 說明之，圖中第一個表為表 4.2 一條染色體之基因值編碼，而本節主要使用其中第一列之重疊服務區域段，另第二個表為表 4.1 含選項標號之分區參數表，至於第三個表則為上述表 4.3 之內容。以圖中顧客點 1 為例，其染色體基因值為 1，故應對應至參數表#1，代表顧客點 1 選取區域 1，最後將其填入給定分區參數之表格中，如圖中所示；又以顧客點 6 為例，其染色體編碼為 2，對應至參數表#2，代表顧客點 6 選取區域 2；至於顧客點 10，其染色體編碼為 2，對應至參數表#2，代表顧客點 10 選取區域 3，其餘顧客點則依此方式類推，而此段染色體基因值解碼完成後得到之結果如表 4.4 所示，所有顧客點將確定其所屬之區域。

顧客 <i>i</i>	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
重疊服務區域	1					2				2					1	1		
顧客服務順序	2	12	3	4	8		13	17	6		10	14	16	7	13	15	11	9

顧客 <i>i</i>	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
可行指派區域	#1	區域 1	區域 3	區域 1	區域 1	區域 3	區域 1	區域 1	區域 3	區域 2	區域 1	區域 1	區域 2	區域 1	區域 2	區域 1	區域 3	區域 1
	#2	區域 2				區域 2				區域 3					區域 2	區域 3		
	#3					區域 3												

顧客 <i>i</i>	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
給定分區參數	重疊	區域 3	區域 1	區域 1	區域 3	重疊	區域 1	區域 1	區域 3	重疊	區域 2	區域 1	區域 1	區域 2	重疊	重疊	區域 3	區域 1
重疊服務區域	1					2				2					1	1		
顧客服務順序	2	12	3	4	8	1	13	17	6	5	10	14	16	7	18	15	11	9

圖 4.2 染色體重疊服務區域段解碼示意圖

表 4.4 染色體重疊服務區域段解碼完成表

顧客 <i>i</i>	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
給定分區參數	區域 1	區域 3	區域 1	區域 1	區域 3	區域 2	區域 1	區域 1	區域 3	區域 3	區域 2	區域 1	區域 1	區域 2	區域 1	區域 2	區域 3	區域 1
重疊服務區域	1					2				2					1	1		
顧客服務順序	2	12	3	4	8	1	13	17	6	5	10	14	16	7	18	15	11	9

4.2.2 顧客服務順序之解碼

本節介紹染色體第二段顧客服務順序之解碼，為直觀地詳述解碼之過程，本研究將在染色體第一部分直接呈現 4.2.1 節解碼完成後所獲得之顧客區域，取代原本重疊服務區域段之基因值，並將其標示為顧客所屬區域，如表 4.5 所示。

表 4.5 顧客所屬區域與染色體顧客服務順序段基因值

顧客 <i>i</i>	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
顧客所屬區域	區域 1	區域 3	區域 1	區域 1	區域 3	區域 2	區域 1	區域 1	區域 3	區域 3	區域 2	區域 1	區域 1	區域 2	區域 1	區域 2	區域 3	區域 1
顧客服務順序	2	12	3	4	8	1	13	17	6	5	10	14	16	7	18	15	11	9

染色體顧客服務順序段之解碼過程分為兩個步驟，首先第一步將顧客依所屬區域進行分群，如表 4.6 所示，將所有顧客點依「顧客所屬區域」進行排列。

表 4.6 依顧客所屬區域排序後之染色體顧客服務順序段

顧客 <i>i</i>	1	3	4	7	8	12	13	15	18	6	11	14	16	2	5	9	10	17
顧客所屬區域	區域 1	區域 2	區域 2	區域 2	區域 2	區域 3												
顧客服務順序	2	3	4	13	17	14	16	18	9	1	10	7	15	12	8	6	5	11

接著第二步驟則是分別針對各個區域之顧客點依「顧客服務順序」段之基因值由小至大排列，意即基因值小之顧客將先行服務，如圖 4.3 針對區域 1 之顧客點排序所示，各個區域將分別執行排序之動作，而此部分排序完成則如表 4.7 所示，意即染色體第二部分顧客服務順序段之解碼完成。

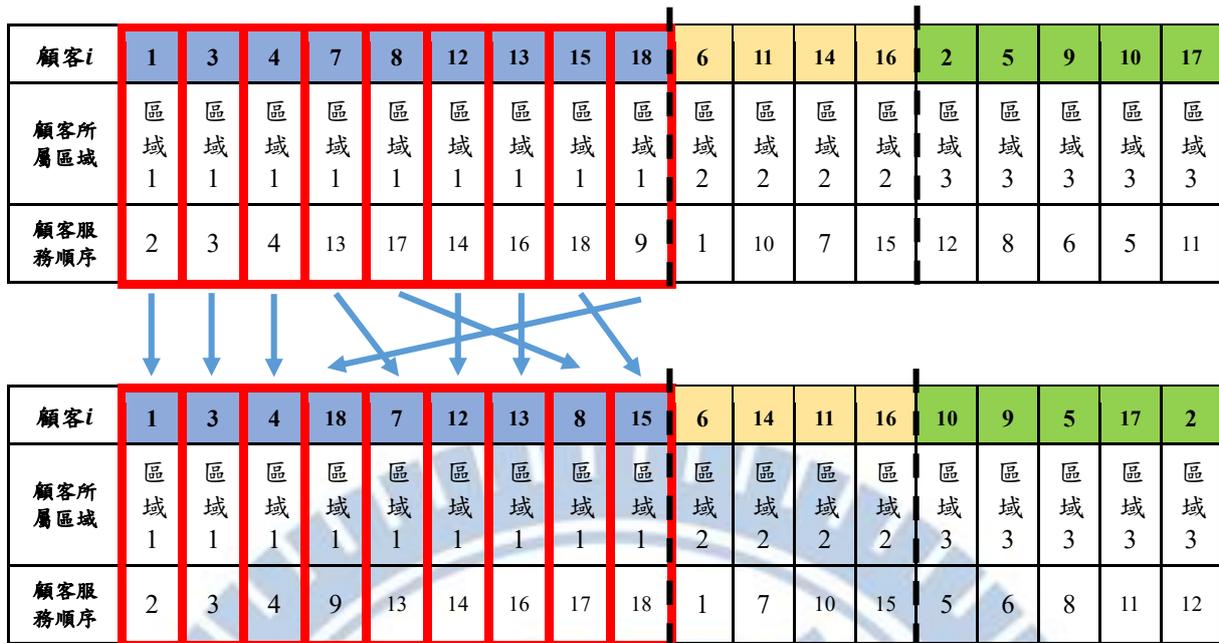


圖 4.3 染色體顧客服務順序段解碼示意圖

表 4.7 染色體顧客服務順序段解碼完成表

顧客 <i>i</i>	1	3	4	18	7	12	13	8	15	6	14	11	16	10	9	5	17	2
顧客所屬區域	區域 1	區域 2	區域 2	區域 2	區域 2	區域 3												
顧客服務順序	2	3	4	9	13	14	16	17	18	1	7	10	15	5	6	8	11	12

4.2.3 車輛路徑與使用車輛之決策

在染色體兩部分之基因值皆已完成解碼程序後，本節將介紹物流公司如何透過染色體之含意進行其決策目標。首先進行車輛路徑之安排，在染色體解碼完成後獲得顧客之分區資訊與服務之順序，由於顧客需求配送之車輛路徑安排必須遵守其所屬區域之限制，因此在安排車輛路徑時亦分別針對各個區域進行探討，另由於本研究考量卡車裝載之重量與體積限制，因此必須針對每個顧客點逐一測試其裝載可行性，以區域 1 為例，安排路徑時依顧客服務順序編碼由小至大服務，因此第一個服務之顧客點為顧客 1，接著再服務顧客 3，然而在將顧客 3 插入路徑的同時，必須先測試其所需貨物之重量與體積是

否允許被放入該輛卡車中，若允許，則將顧客 3 插入該路徑，接著往下一順位之顧客 4 邁進，倘若顧客 4 於裝載時已違反卡車裝載限制，則將新增第二輛卡車為區域 1 進行服務，然而在新增卡車的同時，亦可確認前一輛卡車所配送服務之路徑，至於剩餘之顧客點與其他區域則依此方式類推，路徑分類完成如表 4.8 所示，可明顯看出在此範例中區域 1 需要 4 輛卡車、區域 2 需要 1 輛、區域 3 需要 2 輛，而此圖中所代表之路徑含意則如圖 4.4 所示。

表 4.8 車輛路徑安排與所需車輛數示意表

顧客 i	1	3	4	18	7	12	13	8	15	6	14	11	16	10	9	5	17	2
顧客所屬區域	區域 1	區域 2	區域 2	區域 2	區域 2	區域 3												
顧客服務順序	2	3	4	9	13	14	16	17	18	1	7	10	15	5	6	8	11	12

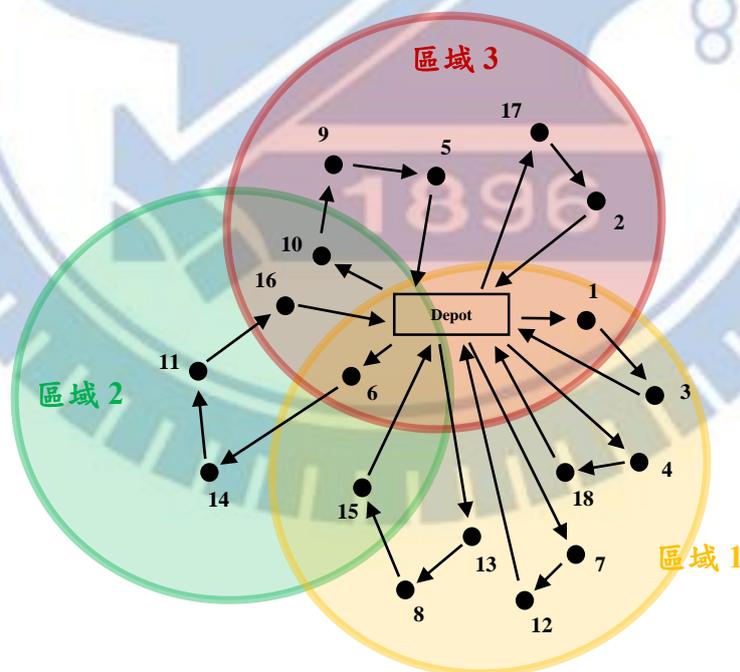


圖 4.4 車輛路徑示意圖

而在使用車輛之決策方面，依據 3.1 節中所介紹之車輛使用模式的三種方式：(1) 使用該區域之公司自有車輛 (2) 使用其他區域之公司自有車輛 (3) 外包租用車輛，在此三種方式中依序使用。同圖 3.1 之範例，倘若每個區域給定自有卡車 2 輛，則在圖 4.4 中可以明顯看出區域 1 不足 2 輛車與區域 2 剩餘 1 輛車，因此區域 1 在自有車輛不敷使用之情況下，將以方式(2)向區域 2 挪用剩餘之車輛，然而其還是不足一輛車，則區域 1 再以方式(3)外包租用一輛車，詳細之操作如 3.1 節與圖 3.2 所述。

4.2.4 染色體成本值計算

在完成決策模式後，最後一個步驟即是要量化一條染色體，藉以比較其優劣性，意即計算在此車輛路徑與車輛使用模式下染色體之成本值，本研究使用 3.3 節中所定義之目標式公式(3.1)作為染色體成本值之計算公式。在最小化總成本的架構下，成本值越低則代表染色體有越佳的表現，其中物流配送成本(TC)包含車輛固定成本、外包車輛成本與車輛路徑成本三個部分，以上述範例而言，總共使用 7 輛卡車，故車輛固定成本(FC)以 7 輛車計算，而另外外包租用 1 輛卡車，則在第二項外包車輛成本與車輛固定成本之差額($OC - FC$)以 1 輛車計算，最後再計算整體行經路徑之總成本，此三項加總所得之物流配送成本即為本研究所定義之染色體成本值。

4.3 運算子

本節將介紹基因演算法之運算子，包含選擇、交配與突變三個程序，而基因演算法藉由此三種程序產生新的個體子代，藉以尋求較佳之解，直至符合終止條件方可停止演算法之搜尋。

4.3.1 選擇

運算子選擇之主要目的為在母體中選取所需之染色體個數進入交配與突變之程序，而本研究採用之選擇法有六種方式，前三種為基本類型之選擇法，是為輪盤選擇法(Roulette Wheel Selection)、等級輪盤選擇法(Rank-based Roulette Wheel Selection)與競爭選擇法(Tournament Selection)，其詳細步驟與做法將分別於 4.3.1.1 節、4.3.1.2 節與 4.3.1.3 節中詳述；而後三種選擇法則是將菁英群組(Elite Group)之概念套入前三種選擇法做整合性之使用，是為菁英輪盤選擇法(Roulette Wheel Selection with Elite Group)、菁英等級輪盤選擇法(Rank-based Roulette Wheel Selection with Elite Group)與菁英競爭選擇法(Tournament Selection with Elite Group)，而菁英群組之詳細概念將於 4.3.1.4 節介紹。最後本研究將於第五章進行數據實驗時，藉由參數測試之效果，於此六種選擇法中擇一最優之方法作為本研究基因演算法之選擇運算子程序。

4.3.1.1 輪盤選擇法

輪盤選擇法(Roulette Wheel Selection)之核心概念為以較高之機率選取表現較好之染色體，意即成本值較低之染色體有較高之機率被選擇，藉以保留基因優良之母代，反之則淘汰表現較差之染色體。

輪盤選擇法之操作模式如圖 4.5 所示，設有母代染色體 p 條，而圖中第一欄與第二欄為染色體之母代與其對應之成本值，另因本研究之目標為最小化成本值，與多數文獻中輪盤選擇法最大化適應度值之精神有考量上之差異，故本研究以倒數之方式轉換成本值為染色體適應度值，如圖中第三欄所示，並於第四欄以適應度值除以所有染色體適應度值之總和(F)計算各染色體所對應之概率，而第五欄則是以累加之方式呈現第四欄之概率，最後選取與染色體數量相同之 p 個隨機變數於第六欄，並令其值介於 0~1 之間，再逐一判斷與選擇隨機變數值所位在之累加區間，用以決定染色體之排序，如第一個隨機變數值為 0.0812648，其介於染色體 1 之 0.0557179 與染色體 2 之 0.1045220 之間，故選

擇染色體 2，如圖中第七欄所示，而其餘隨機變數之選取規則則依此類推，最後產生 p 條染色體之新排序如圖 4.6 所示。由此可見成本值越低之染色體，在圖 4.5 第五欄中擁有較大之累加區間，意即有較大之被選機率，且在圖 4.6 中也較容易重複出現，反之成本值高之染色體出現之次數則較低，最後在圖 4.6 中選取所需數量之染色體進入下一子代，藉此達成輪盤選擇法之優選目的。

p 條染色體	成本值	適應度值 (1 / 成本值)	適應度值 / F	累加	p 個隨機變數 (0~1 之間)	
Chromosome 1	2965	0.0003373	0.0557179	0.0557179	0.0812648	2
Chromosome 2	3385	0.0002954	0.0488041	0.1045220	0.5646487	12
Chromosome 3	3682	0.0002716	0.0448680	0.1493900	0.0135464	1
Chromosome 4	3780	0.0002646	0.0437050	0.1930950	0.1026513	2
Chromosome 5	3407	0.0002935	0.0484890	0.2415840	0.6545634	13
⋮	⋮	⋮	⋮	⋮	⋮	⋮
Chromosome p	3357	0.0002979	0.0492120	1	0.0479854	1
		$F=0.006053$	1			

圖 4.5 輪盤選擇法操作之範例



圖 4.6 輪盤選擇法操作之染色體新排序

4.3.1.2 等級輪盤選擇法

等級輪盤選擇法(Rank-based Roulette Wheel Selection)之核心精神與上述之輪盤選擇法類似，同樣以較高之概率選取表現較好之染色體，然而其中之差異僅在染色體適應度值之制定上，等級輪盤選擇法不採用染色體實際之成本數值或適應度數值進行概率之計算，其以染色體之間的優劣排名值進行適應度之安排，同樣以 p 條染色體之圖 4.5 為例，在圖中第三欄以適應度排名值取代適應度值，意即在此欄以 $1\sim p$ 之數值針對染色體之優劣性進行排名，染色體成本值最大，意即表現最差之染色體給予適應度排名值 1，染色體成本值次大者給予適應度排名值 2，依此類推，而表現最佳之染色體適應度排名值則為 p ，等級輪盤選擇法之特色為依此染色體適應度排名值進行染色體選取概率之後續計算，此為其與輪盤選擇法之主要差異。

4.3.1.3 競爭選擇法

競爭選擇法(Tournament Selection)之操作程序相對於前述兩者則較為簡單，其單純以兩兩比賽之方式擇優晉級，同樣以 p 條染色體為例，每次隨機選取其中 2 條染色體進行比較，擇其成本值優者，如此反覆執行 p 次將得到一 p 條染色體之新排序，用以進入下一階段運算子之交配程序。

4.3.1.4 菁英群組

菁英群組(Elite Group)為演算法中額外紀錄優良解之方式，在此機制下演算法將針對歷代搜尋所獲得之優良解進行紀錄，並與其他選擇法做結合，在每次執行運算子選擇程序時，確保菁英群組內之染色體擁有 100% 之被選機率。以 20 條染色體為例，倘若菁英群組比例設定為 0.1，菁英群組將紀錄歷代以來所獲得之優良解前 2 名，意即每次執行選擇法時都必須先將此 2 條菁英染色體放入其中，後再以前述之選擇法選取剩餘之 18

條染色體，以此機制作為確保優良解得以進入運算子交配與突變之程序，然本研究亦將此概念套入前述之三種選擇法，整合為菁英輪盤選擇法(Roulette Wheel Selection with Elite Group)、菁英等級輪盤選擇法(Rank-based Roulette Wheel Selection with Elite Group)與菁英競爭選擇法(Tournament Selection with Elite Group)，是為三種新的選擇模式，以此共 6 種之選擇法進行效能之比較擇一最佳者使用。

4.3.2 交配

本研究在染色體交配之方式採用單點交配(Single-point Crossover)，但由於染色體中兩段基因值編碼之數值結構不同，因此在交配之過程中本研究將兩段染色體分別操作，藉以避免不可行解之產生。

首先介紹第一部分重疊服務區域段之交配程序，在單點交配之架構下於染色體上隨機選取一切斷點，並交換兩染色體於此切斷點之後之子染色體片段，如圖 4.7 所示。

母代 1

顧客 <i>i</i>	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
重疊服務區域	1					2				2					1	1		

母代 2

顧客 <i>i</i>	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
重疊服務區域	1					3				1					1	2		



子代 1

顧客 <i>i</i>	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
重疊服務區域	1					2				1					1	2		

子代 2

顧客 <i>i</i>	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
重疊服務區域	1					3				2					1	1		

圖 4.7 染色體重疊服務區域段交配示意圖

接著為染色體第二部分顧客服務順序段之交配程序，首先同樣於染色體上隨機選取一切斷點，並交換兩染色體於此切斷點之後之子染色體片段，但由於此段染色體之基因值有不可重複之特性，因此若直接進行子染色體片段交換會有不可行解之產生，為避免上述之情況，本研究採用 Gen & Lin (2005)所提出的權重對映交配法(Weight Mapping Crossover; WMX)，在交換切斷點之後之子染色體片段後，將其基因值分別由小至大排列，可得兩組互向對映之數值，如圖 4.8 所示，上排之 4 與下排之 7 對映，上排之 7 與下排之 9 對映，依此類推，接著依對映之數值位置進行交配，以上排之 4 為例，其原為母代 2 顧客 17 之基因值，對映至下排之 7，而下排之 7 原為母代 1 顧客 14 之基因值，

將其對映替換後上排之 4 成為子代 2 顧客 14 之基因值，而下排之 7 成為子代 1 顧客 17 之基因值，其餘對映數值之替換方式則依此類推，而經由權重對映交配法交配後所得之結果如圖 4.8 中之兩條子代所示。

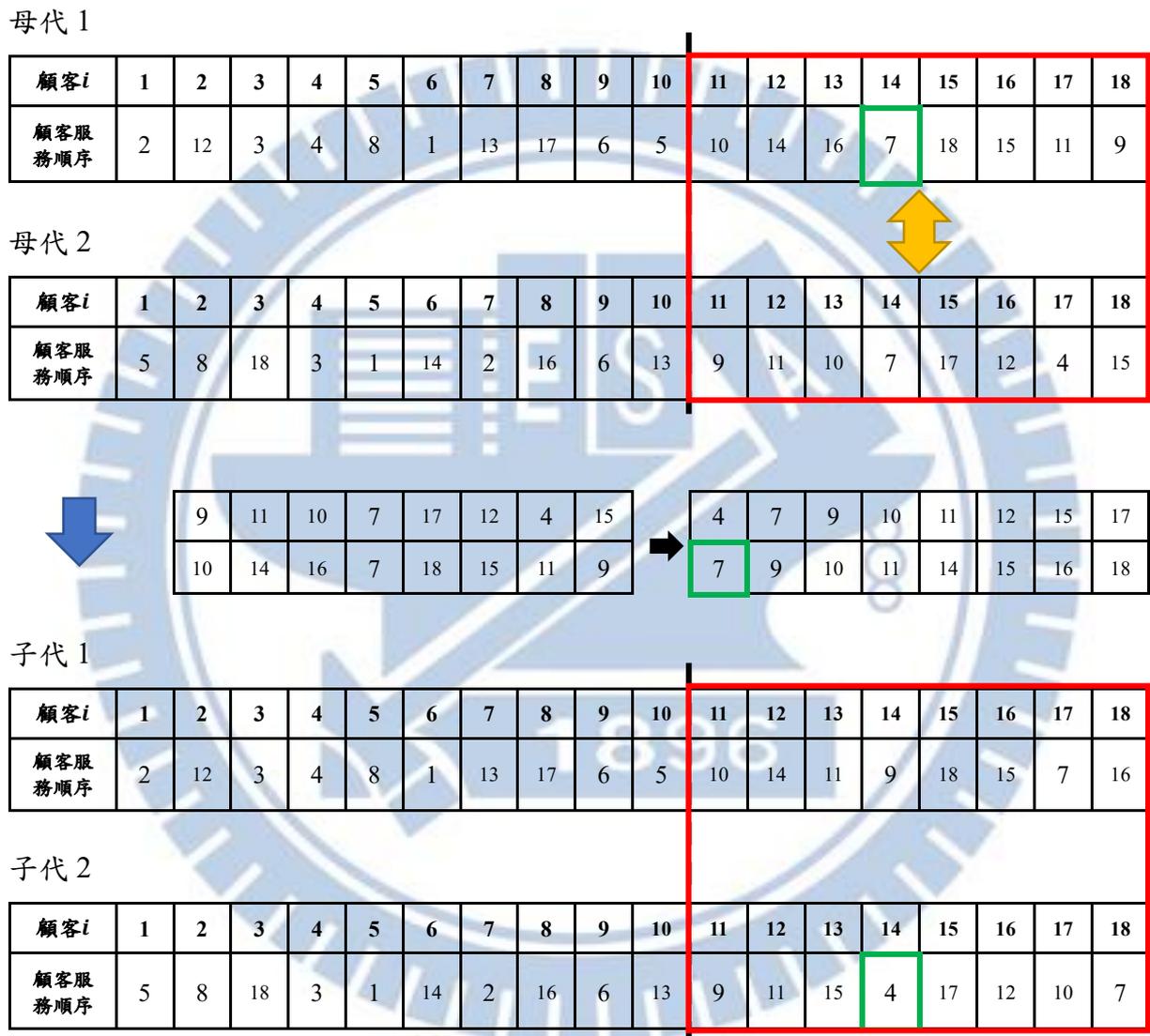


圖 4.8 染色體顧客服務順序段交配示意圖

4.3.3 突變

本研究所採用之突變模式亦將兩段染色體分別執行，第一段重疊服務區域之部分，

首先隨機選取一欲執行突變之基因，並將其對應至表 4.1 之顧客點可行指派區域參數表，於該顧客點之可行區域選項中，重新隨機選取一數值作為突變後之子代基因值，如圖 4.9 所示，隨機選取母代顧客點 6 之基因值執行突變程序，並將其對應至參數表，顧客點 6 之可行區域選項為 #1~#3，亦即突變數值可在 1~3 間隨機產生，然而此範例所產生之突變後基因值為 3，如圖中子代所示。而第二段顧客服務順序之突變程序採用交換突變法，意即隨機選取染色體上兩基因，並將其基因值交換即完成突變，如圖 4.10 所示。

母代

顧客 <i>i</i>	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
重疊服務區域	1					2				1					1	2		

顧客 <i>i</i>	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
可行指派區域	#1	區域 1	區域 3	區域 1	區域 1	區域 3	區域 1	區域 1	區域 3	區域 2	區域 2	區域 1	區域 1	區域 2	區域 1	區域 2	區域 3	區域 1
	#2	區域 3				區域 2				區域 3					區域 2	區域 3		
	#3					區域 3												

子代

顧客 <i>i</i>	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
重疊服務區域	1					3				1					1	2		

圖 4.9 染色體重疊服務區域段突變示意圖

母代

顧客 <i>i</i>	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
顧客服務順序	2	12	3	4	8	1	13	17	6	5	10	14	11	9	18	15	7	16

子代

顧客 <i>i</i>	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
顧客服務順序	2	12	3	4	8	1	7	17	6	5	10	14	11	9	18	15	13	16

圖 4.10 染色體顧客服務順序段突變示意圖

4.4 區域搜尋

前述文獻回顧提及 Baker & Ayechev (2003)指出若在基因演算法中另加入簡單的鄰域搜索程序，可大幅優化基因演算法之求解效能，並且不會增加過多之求解時間，因此本研究試著針對求解「考量重疊服務區域與容量限制之車輛路徑問題」之基因演算法提出一區域搜尋機制，期望能藉以優化此基因演算法之求解效能。

由於交配與突變程序均針對一整條染色體為單位執行，是為全域搜索，因此在區域搜尋機制本研究將試著以區域內之鄰域搜索作為設計之核心概念，意即在表 4.7 解碼完成之染色體中，隨機選擇其中一區域執行區域搜尋，以區域為單位之鄰域搜尋方式取代整條染色體之全域搜尋，類似組內變異之概念，調整一區域內之顧客服務順序，然而本研究所設計之區域搜尋操作模式參考 Leung et al. (2013)所應用之鄰域變化之三種方式，並將之修改應用於本研究之區域搜尋機制欲探討之組內變異，三種區域搜尋模式分別如下所示：

(1) 插入法

插入法首先選定一欲執行區域搜尋之區域，並在其中選定一顧客點與一插入點，如

圖 4.11 區域 1 之顧客 8，插入點為顧客 3 與顧客 4 之間，將原定顧客服務順序 1-3-4-18-7-12-13-8-15 變更為 1-3-8-4-18-7-12-13-15，然此變化必須以染色體基因值描述方能於基因演算法中應用，而為避免基因值重複造成染色體出現不可行解之現象，顧客服務順序之基因值轉換如圖 4.11 所示。

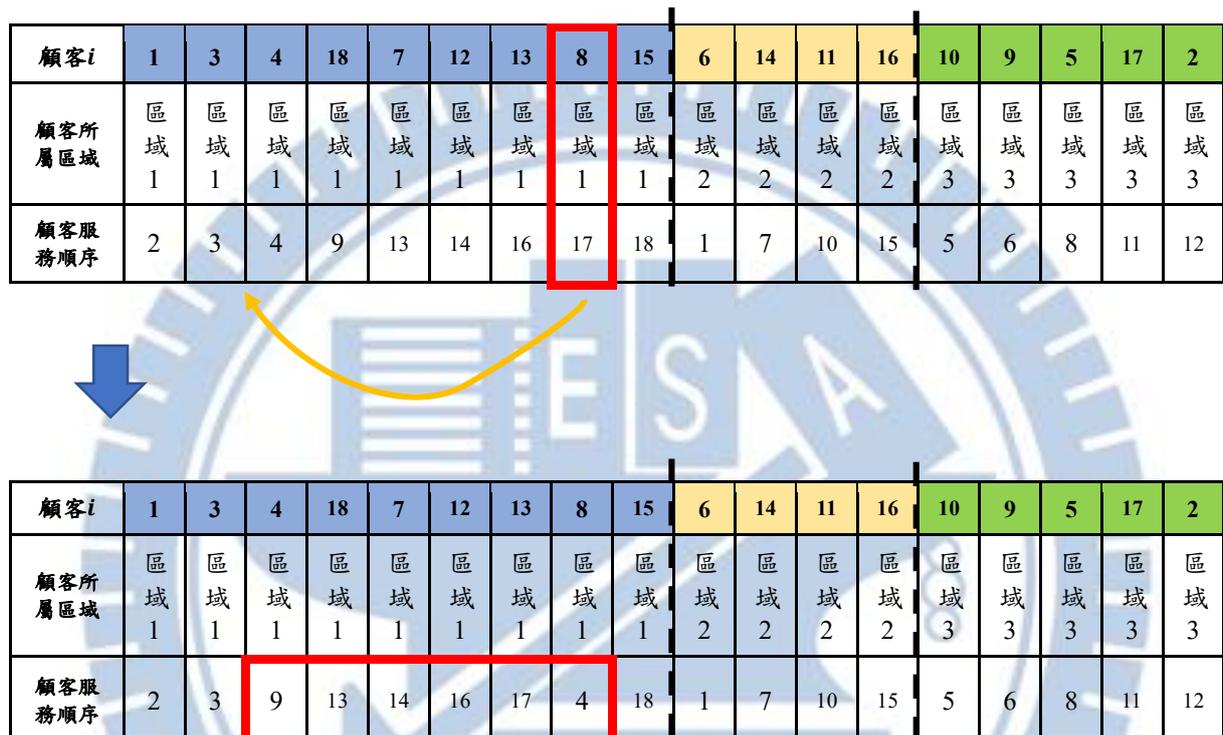


圖 4.11 區域搜尋插入法示意圖

(2) 交換法

交換法之操作顧名思義為將兩顧客點之服務順序進行對調，首先選定一欲執行區域搜尋之區域，並在其中選定兩顧客點做服務順序之交換，如圖 4.12 區域 1 之顧客 4 與顧客 8，意即將區域 1 原定顧客服務順序 1-3-4-18-7-12-13-8-15 變更為 1-3-8-18-7-12-13-4-15，然於基因演算法中此變化亦須以染色體基因值描述，作法為將選定之兩顧客點之基因值對換如圖 4.12。

顧客 <i>i</i>	1	3	4	18	7	12	13	8	15	6	14	11	16	10	9	5	17	2
顧客所屬區域	區域 1	區域 2	區域 2	區域 2	區域 2	區域 3												
顧客服務順序	2	3	4	9	13	14	16	17	18	1	7	10	15	5	6	8	11	12

顧客 <i>i</i>	1	3	4	18	7	12	13	8	15	6	14	11	16	10	9	5	17	2
顧客所屬區域	區域 1	區域 2	區域 2	區域 2	區域 2	區域 3												
顧客服務順序	2	3	17	9	13	14	16	4	18	1	7	10	15	5	6	8	11	12

圖 4.12 區域搜尋交換法示意圖

(3) 倒轉法

倒轉法之操作模式為將服務順序中兩顧客點間所有顧客之服務順序進行倒轉，詳細作法如圖 4.13 所示，首先選定一欲執行區域搜尋之區域，並於其中選定兩顧客點，如區域 1 之顧客 4 與顧客 8，後將此二顧客點間之所有顧客服務順序倒轉，意即將原 4-18-7-12-13-8 之順序倒轉為 8-13-12-7-18-4，故原區域 1 之顧客服務順序 1-3-4-18-7-12-13-8-15 將變更為 1-3-8-13-12-7-18-4-15，而其中染色體基因值之變換亦將於圖 4.13 呈現，此為本研究區域搜尋程序之第三種模式。

顧客 <i>i</i>	1	3	4	18	7	12	13	8	15	6	14	11	16	10	9	5	17	2
顧客所屬區域	區域 1	區域 2	區域 2	區域 2	區域 2	區域 3												
顧客服務順序	2	3	4	9	13	14	16	17	18	1	7	10	15	5	6	8	11	12



顧客 <i>i</i>	1	3	4	18	7	12	13	8	15	6	14	11	16	10	9	5	17	2
顧客所屬區域	區域 1	區域 2	區域 2	區域 2	區域 2	區域 3												
顧客服務順序	2	3	17	16	14	13	9	4	18	1	7	10	15	5	6	8	11	12

圖 4.13 區域搜尋倒轉法示意圖

4.5 終止條件

在經過基因演算法運算子之程序後，會產生新的染色體子代，而經過反覆進行與運算程序，子代又會再產生新的子代，基因演算法藉由如此反覆的搜索期望找出較低成本值之染色體解，然而在反覆進行搜索後需要設定一終止條件停止演算法的搜尋程序。一般基因演算法所使用的終止條件有設定世代數，意即演算法反覆迭代進行，當達到所設定之迭代次數限制後，即終止演算法，另有使用目標值已趨於平穩，意即演算法反覆執行但目標值改善幅度極小時終止演算法。

本研究採用之終止條件為設定世代數，即本研究之基因演算法反覆進行搜尋直至所設定之迭代次數後，停止搜索程序，產出最終解。

五、 實驗測試與結果

本章將針對前述第三章與第四章所提出之數學模式與基因演算法進行實驗數據之分析與比較，藉以了解本研究所提出之方法對於「考量重疊服務區域與容量限制之車輛路徑問題」是否適用，而以下小節所提之程式實驗數據皆以 Visual Studio 2017 C++ 進行撰寫與編輯。

5.1 數學模式求解

本研究將第三章所提之數學模式應用於 Gurobi 7.5.2 最佳化軟體進行求解，並使用本研究設計之圖 3.3 與表 3.1 之 18 個顧客點($n = 18$)、3 個分區($k = 3$)的小規模範例進行測試，並依圖 3.1 指派各區域擁有 2 輛自有車輛，每輛車輛裝載上限為重量 $Q = 10$ 、體積 $V = 25$ ，自有車輛每輛使用費用為 $FC = 200$ ，外包租用車輛之價格為 $OC = 300$ ，而相關之顧客點需求量與距離數據如表 5.1 與表 5.2 所示，其中 D 代表物流中心場站。

表 5.1 範例中各顧客點需求之重量與體積

顧客點	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
需求重量	7	2	2	6	2	3	3	2	2	5	1	3	5	2	2	4	8	3
需求體積	16	4	5	14	6	6	12	5	6	12	2	9	14	5	4	10	20	8

根據 Gurobi 程式求解數學模式之計算結果，此範例之最佳解為 2382，而其車輛路徑安排與車輛使用模式如圖 3.2 所呈現之結果，但此數學模式於 Gurobi 程式求解共花費長達 1,180,310 秒，然而若將此範例代入公式(3.19)與公式(3.20)中探究其原因，則可發現其限制式數量多達 55,693 條，變數數量則為 19,567 個，因此 Gurobi 程式花費約 13 日 15 時 51 分 50 秒求得上述之最佳解。故本研究認為 Gurobi 程式雖求得範例之最佳解，但所花費之時間也相當可觀，因此以數學模式求解較難為物流公司之決策者使用。

表 5.2 範例中各顧客點間距離參數表

	D	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
D	0	33	42	51	55	32	28	62	63	44	33	62	68	51	67	44	39	47	45
1	33	0	27	23	35	48	57	54	75	69	64	92	66	56	96	65	71	44	35
2	42	27	0	47	61	38	70	82	100	62	65	102	93	83	110	86	77	20	63
3	51	23	47	0	16	72	72	41	73	92	85	107	56	53	106	71	89	66	27
4	55	35	61	16	0	81	70	26	62	99	88	106	42	43	101	65	91	79	17
5	32	48	38	72	81	0	50	93	95	23	32	72	100	83	87	73	46	26	74
6	28	57	70	72	70	50	0	66	56	48	28	36	66	46	40	26	22	72	55
7	62	54	82	41	26	93	66	0	41	106	91	98	16	24	88	53	89	97	19
8	63	75	100	73	62	95	56	41	0	99	79	69	29	20	53	25	71	110	46
9	44	69	62	92	99	23	48	106	99	0	20	57	110	91	76	74	34	49	89
10	33	64	65	85	88	32	28	91	79	20	0	39	92	73	106	53	14	59	76
11	62	92	102	107	106	72	36	98	69	57	39	0	93	74	23	47	25	97	89
12	68	66	93	56	42	100	66	16	29	110	92	93	0	18	79	46	88	107	31
13	51	56	83	53	43	83	46	24	20	91	73	74	18	0	63	28	69	95	27
14	67	96	110	106	101	87	40	88	53	76	106	23	79	63	0	36	43	110	84
15	44	65	86	71	65	73	26	53	25	74	53	47	46	28	36	0	45	91	47
16	39	71	77	89	91	46	22	89	71	34	14	25	88	69	43	45	0	71	76
17	47	44	20	66	79	26	72	97	110	49	59	97	107	95	110	91	71	0	78
18	45	35	63	27	17	74	55	19	46	89	76	89	31	27	84	47	76	78	0

5.2 實驗例題設計

由於過去文獻中並無適合本研究所提「考量重疊服務區域與容量限制之車輛路徑問題」使用之例題，故本研究參考車輛路徑問題中標竿題庫例題之生成方式，以 Augerat et al. (1995)所整理之標竿題庫為基礎，場站與各顧客點皆座落於 100*100 之座標圖當中，另建立一支用於生成實驗例題之程式，計算各顧客點間之距離，然而為凸顯 3.1 節中所提之車輛使用模式，本研究將原標竿題庫中需求重量產生方式放大，意即本研究探討之需求重量與體積產生方式由原 1~30 隨機亂數放大至 1~60 間隨機產生，而卡車車輛之容積限制仍設為重量 $Q = 100$ 、體積 $V = 100$ ，藉以完整呈現車輛使用之模式。至於在分區之架構中，本研究以象限之概念針對座標圖進行分區，如圖 5.1 所示，每 50*50 之區

塊為一區域，共將座標圖分為 4 區($k = 4$)，而至於重疊服務區域之分區架構則如圖 5.2 所示，將區域 1、區域 2 與區域 3 之服務範圍放大為 60×60 ，而區域 4 維持 50×50 ，藉以擁有兩區域互相重疊、三區域互相重疊與四區域重疊之部分，用以呈現完整之廣義型重疊服務區域之概念。

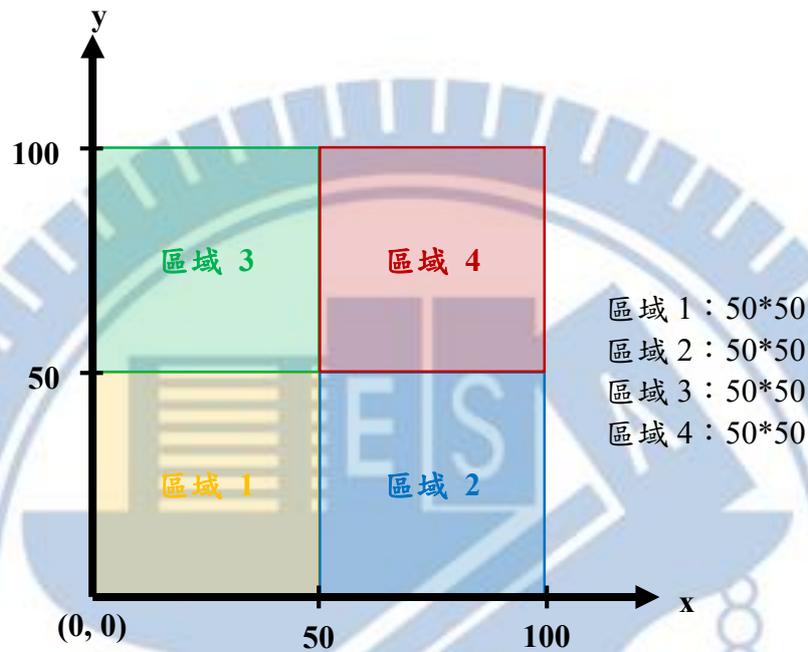


圖 5.1 無重疊服務區域之分區方法示意圖

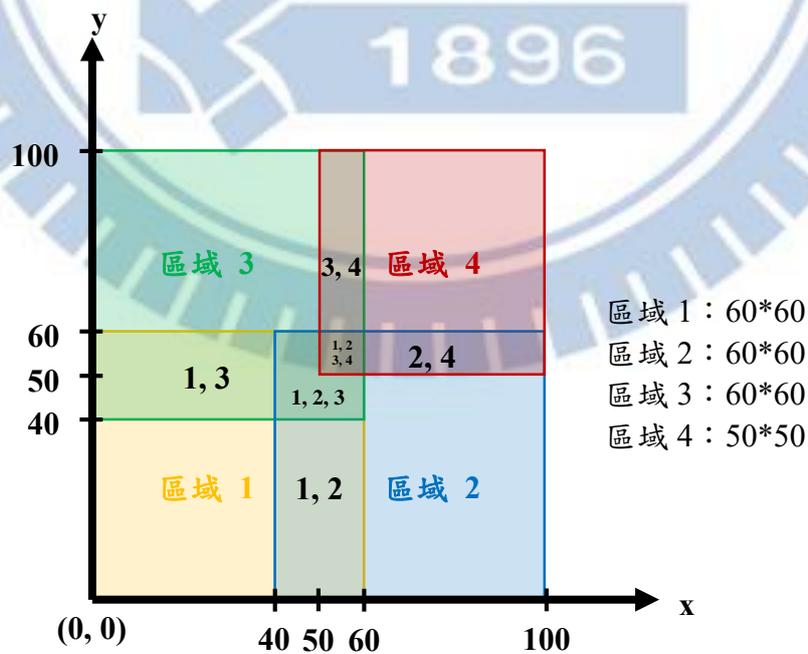


圖 5.2 有重疊服務區域之分區方法示意圖

5.3 基因演算法參數測定

本節將針對本研究第四章所提出之基因演算法進行參數測定，用以執行後續實驗中演算法參數之設定，其中測定內容包含選擇法之選擇、迭代數、染色體數量、交配率、突變率與菁英群組之比例。

5.3.1 選擇法

本節針對 4.3.1 節中提出之六種基因演算法運算子選擇法進行參數實驗，目的選出效果最佳之選擇法用於演算法中，而參數實驗中以 5.2 節提出之例題生成方式，隨機生成 20 題例題，並針對每個例題分別以程式執行 30 次，藉以凸顯隨機之代表性，然六種選擇法所得之各 20 組數據分布盒鬚圖如圖 5.3 所示。

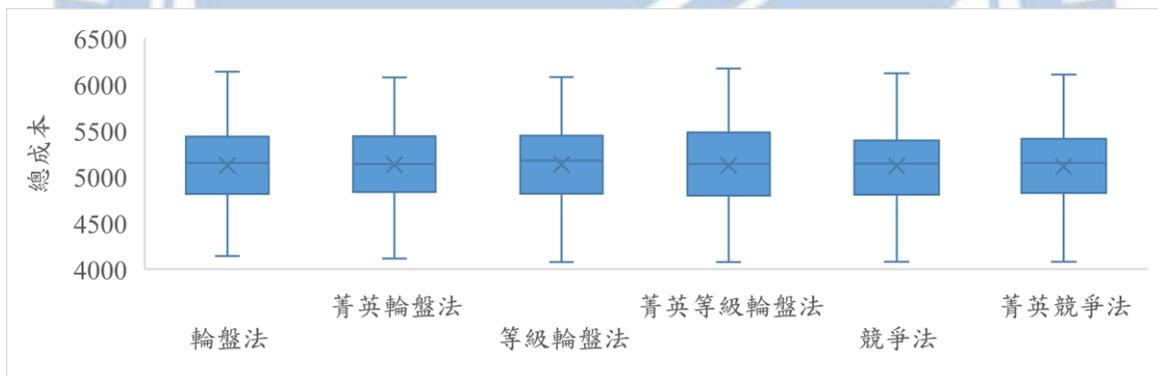


圖 5.3 各選擇法所測得之 20 組數據分布盒鬚圖

然而盒鬚圖中並無法明顯分辨六種選擇法之優劣，故本研究另針對 20 組數據之成本平均值與演算法最佳解平均出現的代數進行折線圖之分析，詳如圖 5.4 所示，其中可以發現使用菁英群組之競爭選擇法擁有最低之平均成本，並且演算法最佳解出現之時機也最早，故而本研究認為使用含菁英群組之競爭選擇法將有最佳之演算法求解效能，因此本研究後續實驗皆使用此選擇法作為基因演算法之運算子。

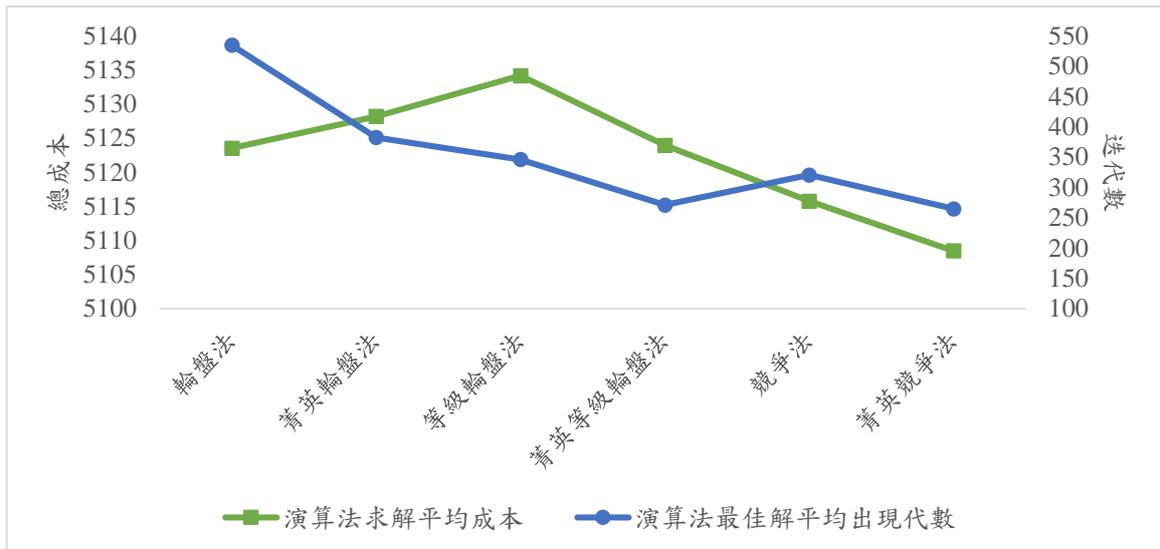


圖 5.4 各選擇法所測得之平均成本值與最佳解平均出現代數折線圖

5.3.2 迭代數

本研究將迭代數設為基因演算法之終止條件，因此迭代數之設定至關重要，其影響著演算法求解之效能，迭代數過高將使演算法耗時且效率不佳，然而過低則有可能使解的品質較差，意即演算法未及收斂即停止搜索，故本研究同上述選擇法使用隨機之 20 個例題並分別執行 30 次，以其平均值分別針對 100~1000 之迭代數目每 100 一個級距進行參數之測定，而詳細之測定結果如圖 5.5 與圖 5.6 所示。

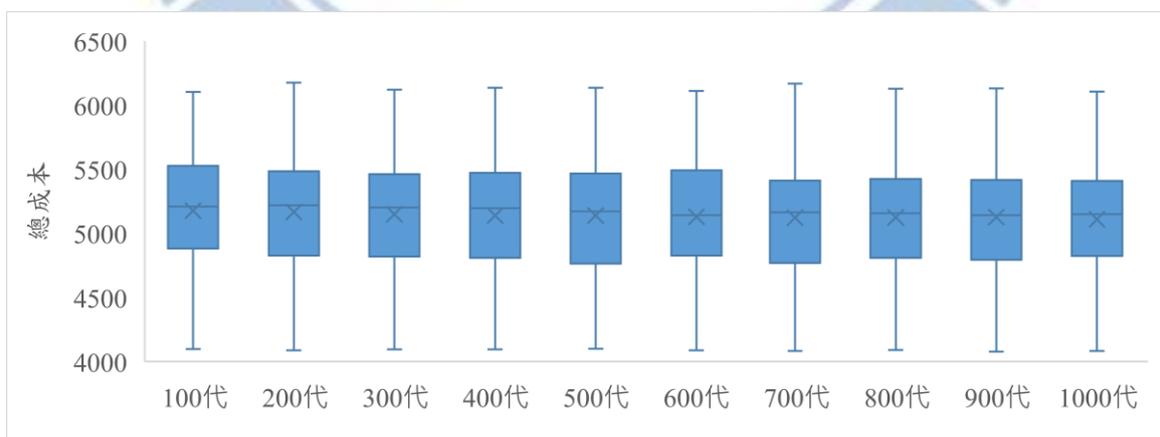


圖 5.5 各迭代數所測得之 20 組數據分布盒鬚圖

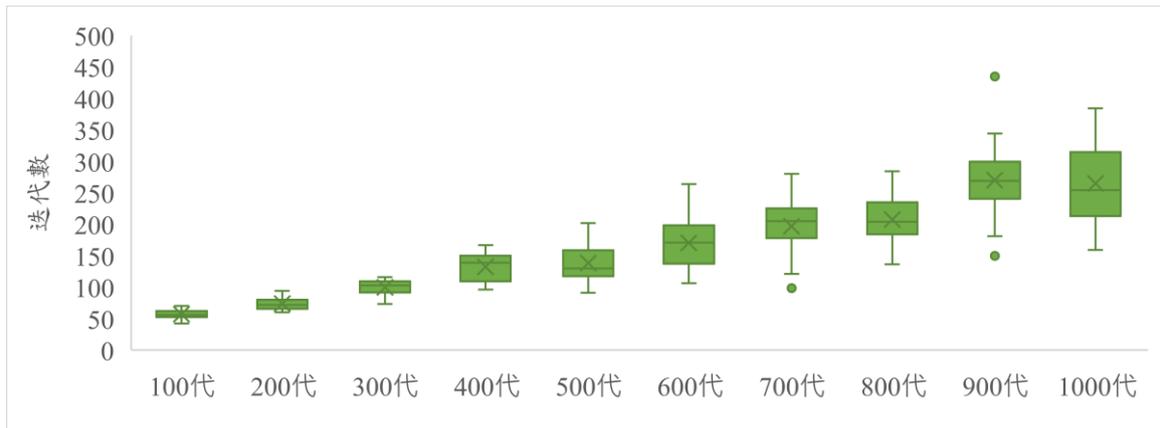


圖 5.6 各迭代數所測得之 20 組最佳解平均出現代數盒鬚圖

在圖 5.5 中各組所得之平均成本分布差異性不大，然而以圖 5.6 之最佳解出現時機做比較，則可明顯發現無論迭代數設定為多少，每次實驗演算法均於 500 代前即收斂，獲得演算法之最佳解，因此本研究將基因演算法之迭代數目設定為 500 代，用以確保演算法之品質與效能。

5.3.3 染色體數量與交配率

染色體數量之多寡與交配率設定之大小亦會影響演算法之效能，染色體數量如同上述之迭代數目，過多之染色體雖然搜尋之範圍較廣，也有較高之機率獲得全域最佳解，但其將使演算法耗時且效率不佳；然而染色體數量過少則有較高之機率陷於區域最佳解，降低演算法的求解品質。另交配率之高低亦影響著演算法之求解，過高將使搜索太頻繁造成效能降低之現象；而過低則可能使演算法之搜索停滯不前，因此染色體數量與交配率之設定極為重要。

本研究針對染色體數量與交配率兩參數作交互配對測定，將染色體數量以 20、30、40、50、60、70、80、90、100 條與交配率 0.5、0.6、0.7、0.8、0.9 進行參數組合，藉以分析在各個交配率下，其分別與多少數量之染色體配對會有最佳之組合，然測定方式同樣以隨機之 20 個例題分別執行 30 次為準，以交配率 0.5 為例，其測定之結果如圖 5.7

與圖 5.8 所示。

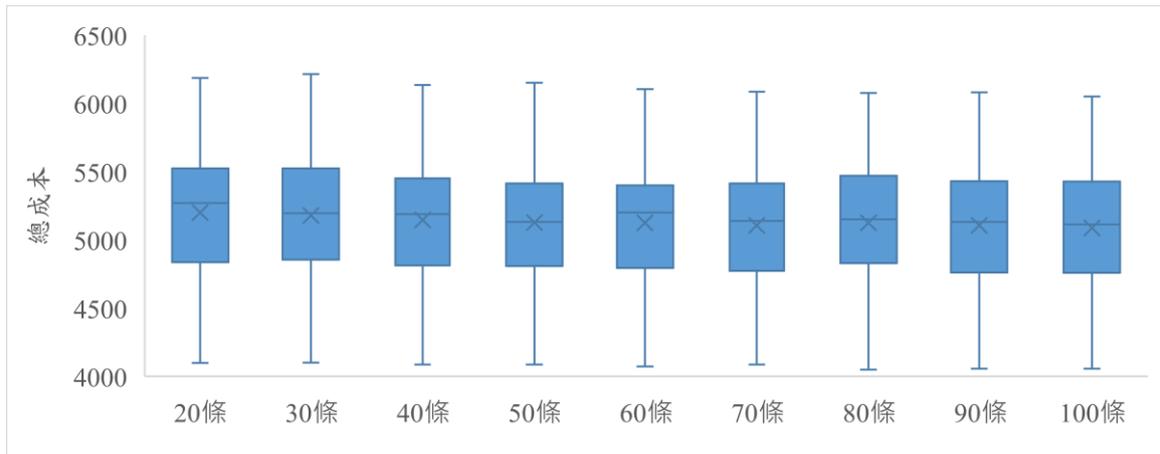


圖 5.7 交配率 0.5 下各染色體數量所測得之 20 組數據盒鬚圖

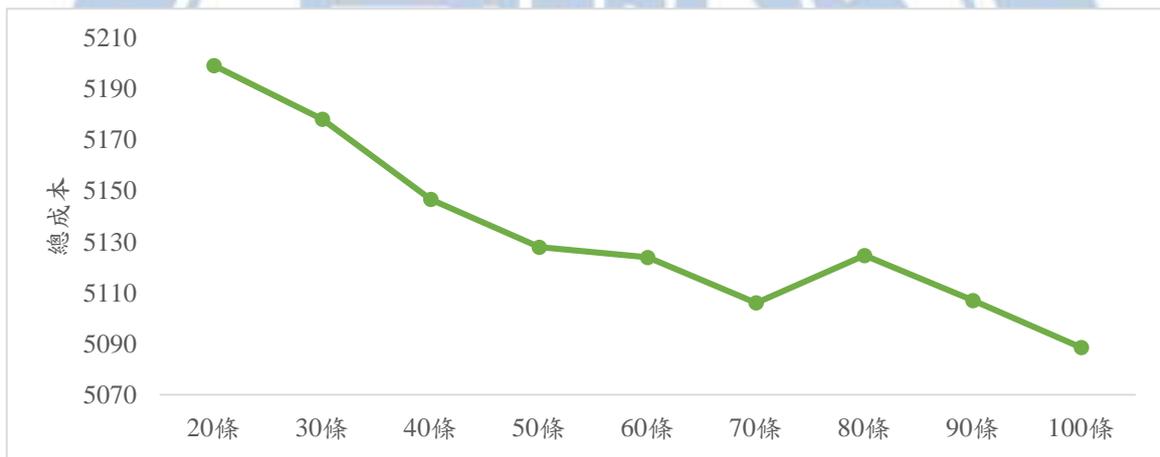


圖 5.8 交配率 0.5 下各染色體數量所測得之平均成本值折線圖

本研究首先針對上述圖中 9 組數據執行 ANOVA 分析，測定其中之數據是否具有顯著差異，然測定結果之 p-value 為 0.9992，顯示 9 組數據中並無顯著之差異，因此本研究另針對其中之數據執行 t 檢定，以染色體數量之對組方式分析各水準之間是否具有顯著差異，其測定結果如表 5.3 所示，(20 vs. 30)、(40 vs. 50)、(50 vs. 60)、(60 vs. 70)、(70 vs. 80)、(80 vs. 90)、(90 vs. 100)、(60 vs. 80)八組之 p-value 大於 0.05，意即在 95%之信心水準下，接受虛無假設，表示兩者間之數據並無顯著之差異；另(30 vs. 40)、(40 vs. 60)

與(80 vs. 100)三組之 p-value 則小於 0.05，意即在 95%之信心水準下，拒絕虛無假設，表示兩者間之數據有顯著之差異，因此由此表之分析結果可見，在交配率 0.5 之設定下，其與染色體數量 100 條擁有最佳之配對效果。

表 5.3 交配率 0.5 下各染色體數量對組之 t 檢定結果

染色體數量對組	雙尾檢定 p-value	95%信心水準結果	染色體數量對組	雙尾檢定 p-value	95%信心水準結果
20 vs. 30	0.0613	無差異	80 vs. 90	0.0664	無差異
30 vs. 40	0.0043	有差異	90 vs. 100	0.0978	無差異
40 vs. 50	0.2220	無差異	40 vs. 60	0.0125	有差異
50 vs. 60	0.8141	無差異	60 vs. 80	0.9525	無差異
60 vs. 70	0.1191	無差異	80 vs. 100	0.0011	有差異
70 vs. 80	0.0680	無差異			

另在交配率 0.6、0.7、0.8 與 0.9 之設定下，亦使用與上述相同之測定方式，首先針對各個情境下之 9 組數據分別執行 ANOVA 分析，檢定其中數據之差異，然而在交配率 0.6、0.7、0.8 與 0.9 之設定下，ANOVA 分析之 p-value 分別為 0.9995、0.9996、0.9986 與 0.9993，顯示在各情境下，其中之數據皆無顯著差異，因此本研究另針對交配率 0.6、0.7、0.8 與 0.9 四個情境分別如上述執行 t 檢定，探討其中交配率與染色體數量之最佳配對組合，詳細之圖表數據內容請參見於附錄一。而最終測定之結果為(交配率 0.6—染色體數量 90)、(交配率 0.7—染色體數量 100)、(交配率 0.8—染色體數量 90)與(交配率 0.9—染色體數量 90)擁有其各自之最佳配對效果。

最後，本研究將針對上述(交配率 0.5—染色體數量 100)、(交配率 0.6—染色體數量 90)、(交配率 0.7—染色體數量 100)、(交配率 0.8—染色體數量 90)與(交配率 0.9—染色體數量 90)五個組合，執行最終之交配率與染色體數量之參數測定程序，選出五個組合中具有最佳求解效益者作為此基因演算法之參數，詳細之數據比較如圖 5.9 與圖 5.10 所示。

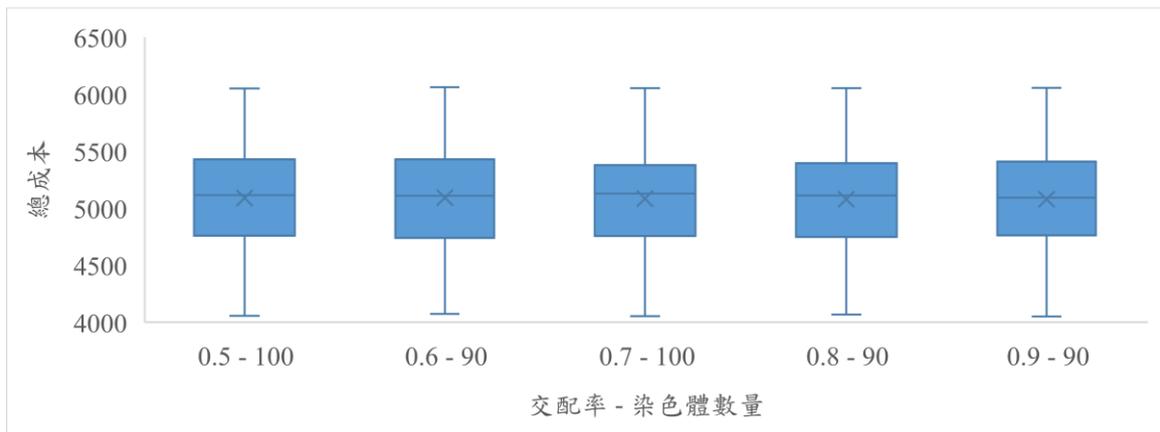


圖 5.9 各參數組合中所測得之 20 組數據分布盒鬚圖

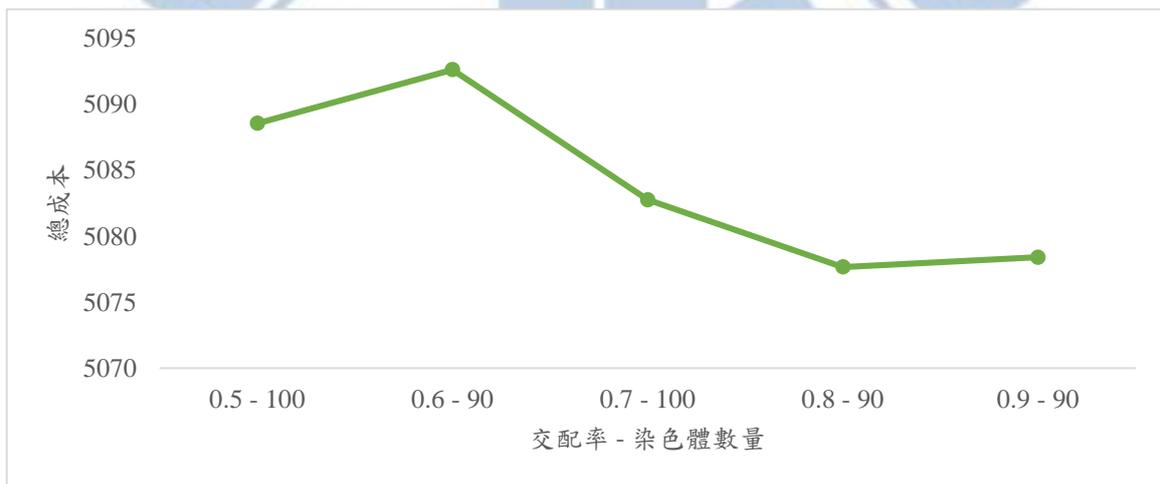


圖 5.10 各參數組合中所測得之平均成本值折線圖

本研究針對五個參數組合先以 ANOVA 測定，其 p-value 為 0.99998，顯示 5 組數據中並無顯著差異，後再將其相互配對設立對組執行 t 檢定之測試，期望確認其中之數據是否確無顯著之差異。而本研究發現所設立之對組中 p-value 均大於 0.05，意即在 95% 之信心水準下，確認各參數組合間之數據並無顯著之差異，表示各參數組合皆有效能相當之演算法求解能力，故本研究以圖 5.10 中平均成本值最低之組合作為基因演算法之參數，意即本研究將染色體數量設定為 90，而交配率則定為 0.8。

5.3.4 突變率

突變為一跳脫區域最佳解之機制，用於避免基因演算法經過前述選擇與交配運算後，所得之解仍然沒有改變之情況，且因本研究所制定之染色體中，第二段編碼採用交換突變法之模式，如 4.3.3 節中之介紹，因此突變率須針對一條染色體為單位制定，而並非針對染色體中之基因值做設定，詳細之參數測定方式仍採用隨機之 20 個例題分別執行 30 次為測定之標準，並針對突變率 0.1、0.15、0.2、0.25、0.3、0.35、0.4、0.45 與 0.5 分別執行程式，其中所得之數據分布與平均成本值如圖 5.11 與圖 5.12 所示。

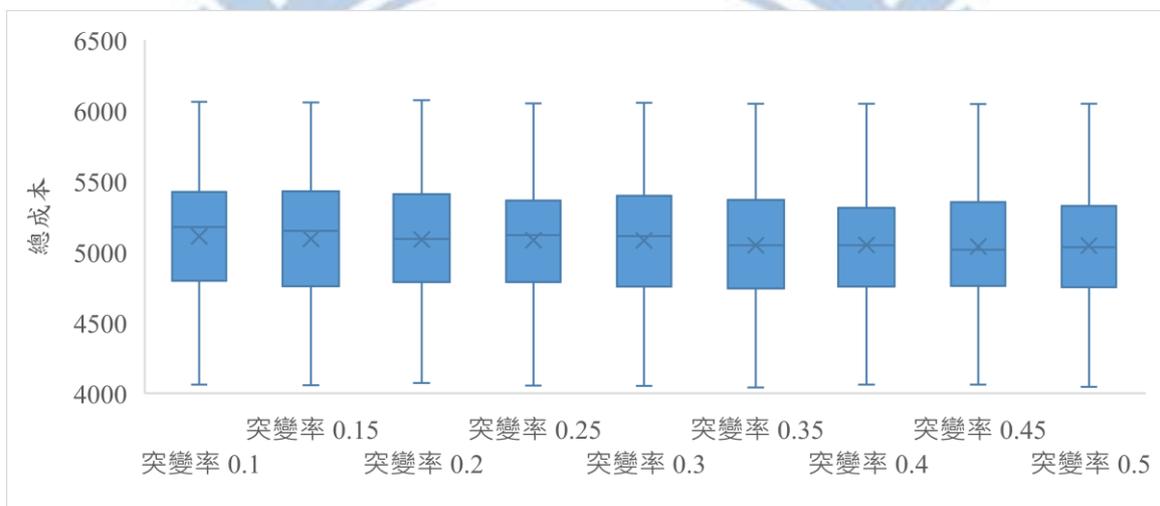


圖 5.11 各突變率所測得之 20 組數據分布盒鬚圖

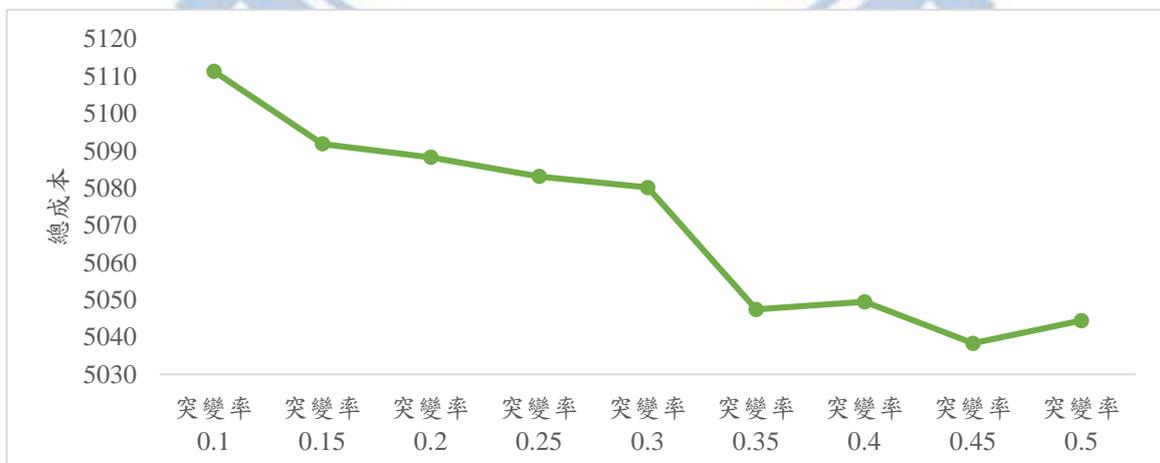


圖 5.12 各突變率所測得之平均成本值折線圖

本研究首先針對上述圖中 9 組數據執行 ANOVA 分析，測定其中之數據是否具有顯著差異，然測定結果之 p-value 為 0.9999，顯示 9 組數據中並無顯著之差異，因此本研究另針對其中之數據執行 t 檢定，以各突變率對組之方式分析各組數據之間是否具有顯著差異，其測定結果如表 5.4 所示，僅(0.1 vs. 0.15)、(0.3 vs. 0.35)兩對組之 p-value 小於 0.05，意即在 95%之信心水準下，拒絕虛無假設，表示兩者間之數據有顯著之差異；而其餘 9 個對組之 p-value 皆大於 0.05，表示在 95%之信心水準下其中之數據並無顯著之差異，由此可知演算法自突變率 0.35 後呈收斂之趨勢，故本研究將演算法之突變率參數定為 0.35。

表 5.4 各突變率對組之 t 檢定結果

突變率對組	雙尾檢定 p-value	95%信心水準結果	突變率對組	雙尾檢定 p-value	95%信心水準結果
0.1 vs. 0.15	0.0056	有差異	0.4 vs. 0.45	0.1012	無差異
0.15 vs. 0.2	0.6949	無差異	0.45 vs. 0.5	0.4598	無差異
0.2 vs. 0.25	0.4877	無差異	0.35 vs. 0.45	0.3380	無差異
0.25 vs. 0.3	0.7375	無差異	0.35 vs. 0.5	0.7711	無差異
0.3 vs. 0.35	0.0131	有差異	0.4 vs. 0.5	0.5717	無差異
0.35 vs. 0.4	0.8394	無差異			

5.3.5 菁英群組比例

菁英群組比例參數主要用於設定 4.3.1.4 節中選擇法紀錄優良解之多寡，其設定之大小影響著演算法之效能，比例過大選擇法中將記錄較多之優良染色體，雖然保留了較多優質解，但卻也限制了演算法搜尋新解之範圍與廣度，且較大的比例值相對也使演算法較為耗時；然而比例過小雖然增加了搜尋新解之範圍，但也容易失去藉由優良解產生更優質解的機會，降低了演算法的求解品質，因此須藉由參數之測定，設定菁英群組比例值之大小。詳細之參數測定方式仍採用隨機之 20 個例題分別執行 30 次為測定之標準，並針對菁英群組比例 0.1、0.2、0.3、0.4 與 0.5 分別執行程式，其中所得之數據分布

與平均成本值如圖 5.13 與圖 5.14 所示。

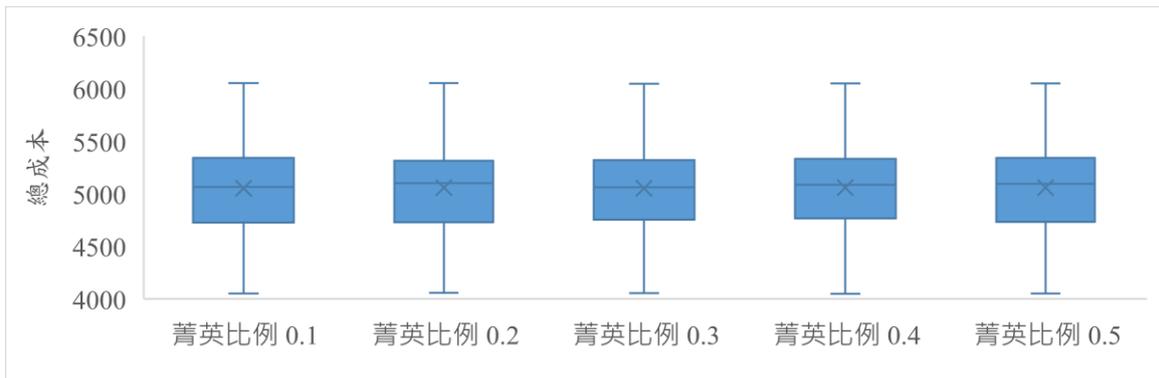


圖 5.13 各菁英群組比例所測得之 20 組數據分布盒鬚圖



圖 5.14 各菁英群組比例所測得之平均成本值折線圖

本研究首先針對上述圖中 5 個水準執行 ANOVA 分析，測定其中之數據是否具有顯著差異，然測定結果之 p-value 為 0.9990，顯示 5 組菁英比例數據並無顯著之差異。因此本研究另針對上述 5 個數據水準以兩兩對組之方式執行 t 檢定，探討各組數據間是否確實無顯著差異，然測定結果各對組之 p-value 皆大於 0.05，意即在 95% 之信心水準下，對組間之兩組數據皆無顯著之差異。

如上述分析與檢定，在 5 個菁英群組比例之水準中，其之間皆不存在顯著之差異，因此本研究選擇圖 5.14 中平均值最低者作為菁英群組比例之設定，即本研究將菁英群組之比例設定為 0.1，且在比例值較小之情況演算法將有較快之求解速度。

5.3.6 收斂分析

經由 5.3.2 節、5.3.3 節、5.3.4 節與 5.3.5 節之參數測定，本研究將基因演算法之迭代數設定為 500 代，染色體數量定為 90 條，交配率為 0.8，突變率與菁英群組比例則分別定為 0.35 與 0.1，然為測定此最佳參數組合在各個迭代數下是否呈收斂之情形，本節將以隨機之 20 個例題分別執行 30 次為 100~1000 代之 10 個迭代水準分別測定此參數組合之收斂情形，而詳細之數據分布與演算法最佳解出現之時機分別如圖 5.15 與圖 5.16 所示。

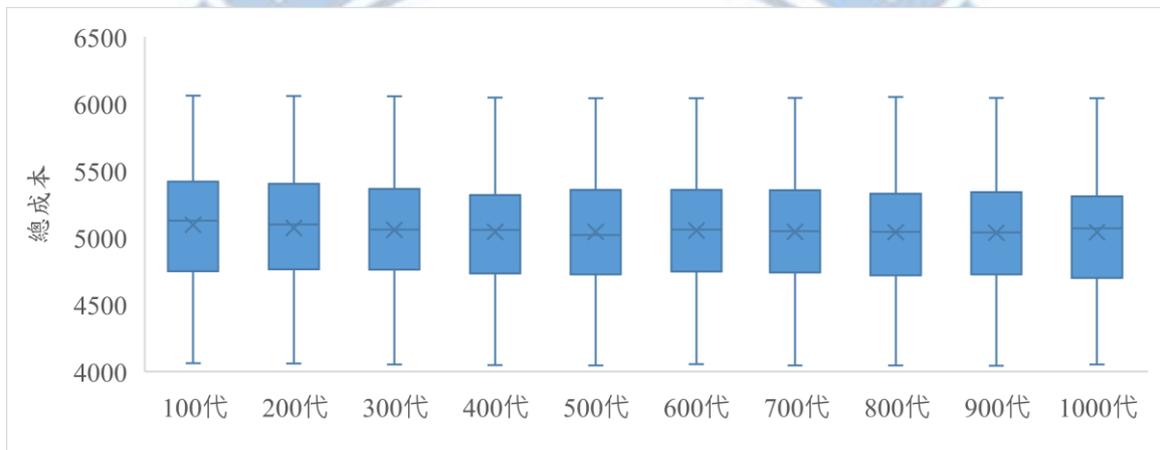


圖 5.15 收斂分析中各迭代數所測得之 20 組數據分布盒鬚圖

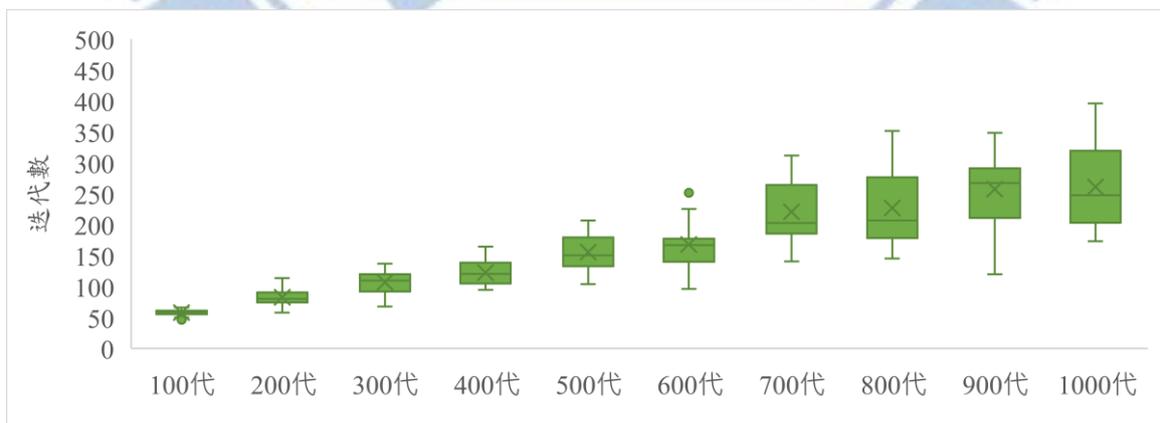


圖 5.16 收斂分析中各迭代數所測得之 20 組最佳解平均出現代數盒鬚圖

在圖 5.15 所呈現之盒鬚圖中，各組所得之平均成本分布差異性並不大，因此本研究另以圖 5.16 呈現之最佳解出現時機做比較，則可明顯發現在各迭代數之程式實驗中，演算法最佳解最晚約於 350 代左右出現，意即無論迭代數設定為多少，每次實驗演算法均於 500 代前即收斂。然而為確保其中數據比較之準確性，本研究另對此執行 t 檢定如表 5.5，其中可發現僅(100 vs. 200)、(200 vs. 400)兩對組之 p-value 小於 0.05，意即在 95% 之信心水準下，拒絕虛無假設，表示兩者間之數據有顯著之差異；而其餘 16 個對組之 p-value 皆大於 0.05，在 95% 之信心水準下其中之數據並無顯著之差異，由此確認演算法約於 400 代開始呈收斂之趨勢，因此驗證本研究之基因演算法於 5.3.2 節所設定之迭代數 500 時，已呈現收斂趨勢，故本研究之實驗將以迭代數 500 執行。

表 5.5 各迭代數對組之 t 檢定結果

迭代數對組	雙尾檢定 p-value	95%信心水準結果	迭代數對組	雙尾檢定 p-value	95%信心水準結果
100 vs. 200	0.0149	有差異	200 vs. 400	0.0078	有差異
200 vs. 300	0.2499	無差異	400 vs. 600	0.0914	無差異
300 vs. 400	0.1878	無差異	400 vs. 800	0.6932	無差異
400 vs. 500	0.9599	無差異	400 vs. 1000	0.7036	無差異
500 vs. 600	0.2906	無差異	500 vs. 800	0.7706	無差異
600 vs. 700	0.1890	無差異	500 vs. 1000	0.7087	無差異
700 vs. 800	0.8678	無差異	600 vs. 800	0.2563	無差異
800 vs. 900	0.6399	無差異	600 vs. 1000	0.1595	無差異
900 vs. 1000	0.6721	無差異	800 vs. 1000	0.9567	無差異

5.4 實驗數據結果

本節將針對三種實驗情境進行數據之比較與分析，分別為「數學模式與基因演算法之比較」、「有無考量重疊服務區域之比較」與「有無使用區域搜尋機制之比較」，然而執行基因演算法時將使用 5.3 節所測定之參數，迭代數 500、染色體數 90、交配率 0.8、突變率 0.35 與菁英群組比例 0.1。

5.4.1 數學模式與基因演算法之比較

5.1 節中已提及使用數學模式求解「考量重疊服務區域與容量限制之車輛路徑問題」較為耗時，數學模式求解本研究所設計之 $n = 18$ 、 $k = 3$ 之例題，所得之最佳解為 2382，其中求解過程共需花費 13 日 15 時 51 分 50 秒，因此本節將探討使用基因演算法與數學模式間求解效能與品質之差異。

本研究以基因演算法反覆執行上述範例 30 次，所得之成本與時間分析圖如圖 5.17 所示，其中可發現在此小型例子中，演算法共有 24 次求得全域最佳解，並且其花費之時間約為 0.3 秒，與數學模型所花費之時間有偌大之差距。並且於表 5.6 之統計顯示，演算法未尋得最佳解之 6 次實驗中，其所求得之解與全域最佳解之差距也僅在 2% 之內，其中平均值更顯示在基因演算法求解之 30 次的過程當中，與最佳解平均之差異僅 0.24%，另平均最佳解出現時機為第 55.67 代，平均花費時間更是僅有 0.29 秒，由此可見此基因演算法之求解品質與全域最佳解相近，並且效率極高，故本研究認為所提出之基因演算法應可於短時間內提供物流公司管理者良好之決策。

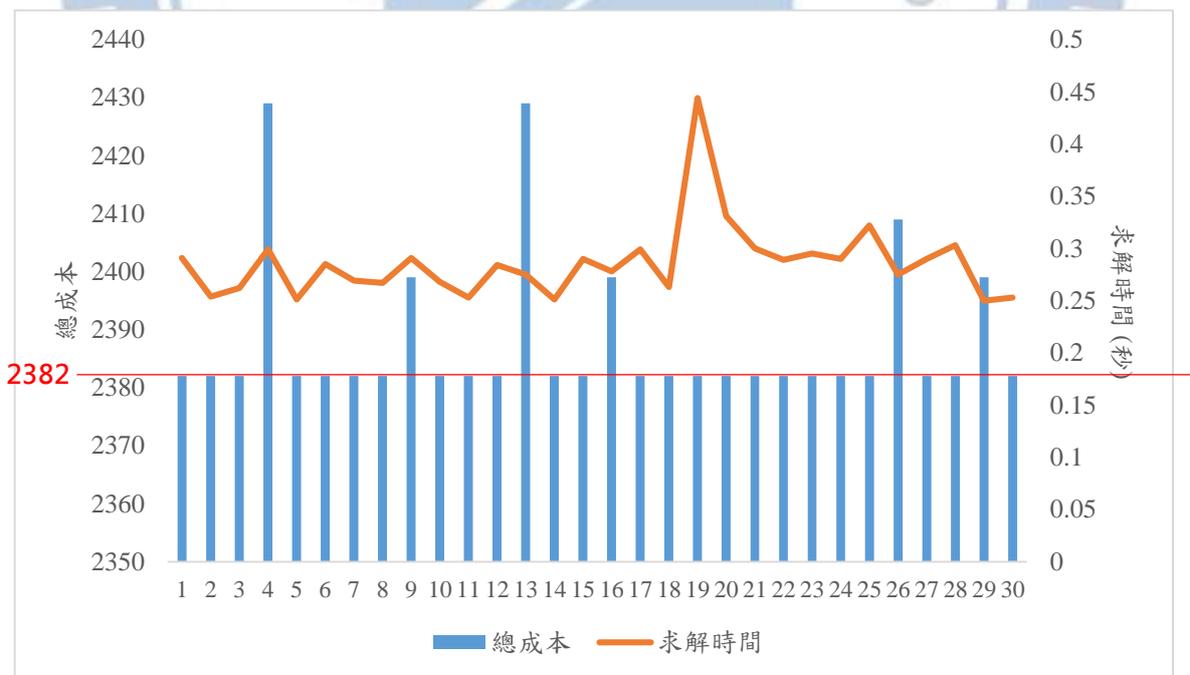


圖 5.17 基因演算法求解 $n = 18$ 、 $k = 3$ 之範例成本與時間分析圖

表 5.6 基因演算法求解 $n = 18$ 、 $k = 3$ 之範例成本與時間數據表

	演算法 最佳解	與數學模 式最佳解 之差異	演算法 最佳解 出現時機 (代)	演算法 求解時間 (秒)		演算法 最佳解	與數學模 式最佳解 之差異	演算法 最佳解 出現時機 (代)	演算法 求解時間 (秒)
1	2382	0	182	0.291	17	2382	0	56	0.299
2	2382	0	92	0.254	18	2382	0	48	0.263
3	2382	0	86	0.262	19	2382	0	23	0.444
4	2429	1.97%	17	0.299	20	2382	0	27	0.331
5	2382	0	20	0.251	21	2382	0	23	0.3
6	2382	0	11	0.285	22	2382	0	26	0.289
7	2382	0	17	0.269	23	2382	0	236	0.295
8	2382	0	10	0.267	24	2382	0	35	0.29
9	2399	0.71%	9	0.291	25	2382	0	11	0.322
10	2382	0	229	0.268	26	2409	1.13%	12	0.275
11	2382	0	17	0.253	27	2382	0	14	0.29
12	2382	0	8	0.284	28	2382	0	19	0.303
13	2429	1.97%	12	0.275	29	2399	0.71%	12	0.25
14	2382	0	9	0.251	30	2382	0	11	0.253
15	2382	0	383	0.29	平 均	2387.73	0.24%	55.67	0.29
16	2399	0.71%	14	0.278					

5.4.2 有無考量重疊服務區域之比較

本研究之主題定義為「考量重疊服務區域與容量限制之車輛路徑問題」，因此本節將針對重疊服務區域進行深入的探討，以實驗測試本研究延伸自劉庭妤(2017)所定之「廣義型重疊服務區域」於例題中之表現結果。其中將針對 31 個顧客($n = 31$)、37 個顧客($n = 37$)、43 個顧客($n = 43$)與 49 個顧客($n = 49$)4 個水準分別產生 20 組例題，並針對每組例題分別執行 30 次程式以平均值作為測定之標準，意即在每個水準下皆有 20 組數據，後再以圖 5.1 與圖 5.2 之分區模式進行分區情境架構，並針對其中是否考量重疊服務區域之概念進行實驗數據分析。如表 5.7 所示，在 $n = 31$ 之水準下實驗之數據，其中包含無考量重疊服務區域之成本值與有考量重疊服務區域之成本值，並針對其計算使用

重疊服務區域架構之成本值改善幅度，而在其他顧客數水準下之實驗測試數據則請參見附錄二。

表 5.7 有無考量重疊服務區域之數據 ($n = 31$)

	無重疊 成本值	有重疊 成本值	改善幅度		無重疊 成本值	有重疊 成本值	改善幅度
1	4717.70	4635.50	1.74%	12	5835.73	4867.23	16.60%
2	6600.20	6050.93	8.32%	13	5218.30	4903.97	6.02%
3	5850.33	5272.63	9.87%	14	4722.33	4051.93	14.20%
4	5134.73	4775.73	6.99%	15	5952.00	5340.90	10.27%
5	5501.57	4884.23	11.22%	16	4702.93	4471.97	4.91%
6	5552.00	5478.47	1.32%	17	6131.97	5340.90	12.90%
7	6168.00	5226.20	15.27%	18	6220.70	5329.77	14.32%
8	5034.80	4750.97	5.64%	19	5586.97	5675.33	-1.58%
9	4962.03	4324.40	12.85%	20	6246.07	5643.60	9.65%
10	5086.90	4715.87	7.29%	平均	5565.26	5052.02	9.03%
11	6079.83	5299.80	12.83%				

另本研究也針對各顧客數水準中有無考量重疊服務區域兩實驗設立對組，以 t 檢定之方式測定在 95% 之信心水準下其中之數據是否有顯著之差異，意即使用廣義型重疊服務區域之概念是否能有效降低物流公司之配送成本，相關之檢定結果如表 5.8 所示。其中各水準之對組中，雙尾檢定 p-value 皆小於 0.05，意即(無重疊 vs. 有重疊)之對組數據於 95% 之信心水準下有顯著之差異，意即使用重疊服務區域之架構可以明顯降低配送總成本，故本研究所使用之廣義型重疊服務區域有助於物流公司之配送。

而表 5.9 則整理各顧客數水準下之平均實驗數據， $n = 31$ 、 $n = 37$ 、 $n = 43$ 與 $n = 49$ 分別有著 9.03%、7.72%、6.31%、3.95% 之改善幅度，由此可見模式中有 30 多個顧客點時，重疊服務區域有將近 10% 之改善效能，而於 40 多個顧客點時，仍有約 5% 之改善效能，因此重疊服務區域能有效應用於物流公司之配送問題。然而本研究認為在相同之分

區模式下，顧客點數量越少，使用重疊服務區域之效益會越明顯，原因為在顧客點數量大時，每個分區內之顧客點數量與密度皆較高，因此演算法有更多之組合求解車輛路徑，提高各分區內每輛卡車之裝載率以降低剩餘空間；反之顧客點數量較少時，各區域內之卡車剩餘空間較大，此時利用重疊服務區域之概念，靈活運用各分區之卡車剩餘空間，效果將較為明顯，此也應證了劉庭好(2017)所指出重疊服務區域共享之效益，並且無論模式中有多少顧客點，本研究所提之「廣義型重疊服務區域」皆能有效降低物流配送成本。

表 5.8 各數量顧客點有無考量重疊服務區域對組之 t 檢定結果

各數量顧客點有無重疊對組	雙尾檢定 p-value	95%信心水準結果
$n = 31$ ：無重疊 vs. 有重疊	0.0000	有差異
$n = 37$ ：無重疊 vs. 有重疊	0.0000	有差異
$n = 43$ ：無重疊 vs. 有重疊	0.0000	有差異
$n = 49$ ：無重疊 vs. 有重疊	0.0000	有差異

表 5.9 各顧客點數量考量重疊服務區域之成本值平均改善幅度比較表

顧客點數量	無考量重疊區域 成本值	有考量重疊區域 成本值	成本值 平均改善幅度
$n = 31$	5565.26	5052.02	9.03%
$n = 37$	6906.48	6338.87	7.72%
$n = 43$	8281.64	7756.64	6.31%
$n = 49$	9546.51	9164.80	3.95%

5.4.3 有無使用區域搜尋機制之比較

本節將針對錯誤! 找不到參照來源。節演算法中所設計之區域搜尋機制，進行效能之測定，以實驗測試有無使用區域搜尋機制對於例題中求解之表現差異，而在區域搜尋使用方式本研究將之分為兩種：(1) 單步驟區域搜尋 (2) 多步驟區域搜尋，單步驟區域

搜尋為針對突變完成之染色體分別執行一次區域搜尋機制，並選擇執行區域搜尋前與後兩者中較佳者進入下一世代；而多步驟區域搜尋同樣針對突變完成之染色體分別執行區域搜尋機制，但其差異在於若該染色體執行區域搜尋後所得之解較原本之解佳，則針對其再執行一次區域搜尋，如此反覆進行直至該染色體解未再改善，並使之進入下一世代。

本節實驗同 5.4.2 節將針對 31 個顧客($n = 31$)、37 個顧客($n = 37$)、43 個顧客($n = 43$)與 49 個顧客($n = 49$)4 個水準分別產生 20 組例題，並針對每組例題分別執行 30 次程式以平均值作為測定之標準，意即在每個水準下皆有 20 組數據，後再針對每次實驗分別以不使用區域搜尋、單步驟區域搜尋與多步驟區域搜尋三種方式執行，進行實驗數據分析之比較。而表 5.10 為在 $n = 31$ 之水準下實驗之數據，其中包含不使用區域搜尋、單步驟區域搜尋與多步驟區域搜尋之成本值、最佳解出現時機與求解時間，並針對兩種區域搜尋使用方式分別計算其與不使用區域搜尋比較之成本值改善幅度，而在其他顧客數水準下之實驗測試數據則請參見附錄三。

表 5.10 有無考量區域搜尋機制之數據 (n = 31)

	不使用區域搜尋			單步驟區域搜尋				多步驟區域搜尋			
	成本值	最佳解出現時機 (代)	求解時間	成本值	最佳解出現時機 (代)	求解時間	成本值改善幅度	成本值	最佳解出現時機 (代)	求解時間	成本值改善幅度
1	4639.43	222.23	0.30	4603.60	166.93	0.58	0.77%	4635.50	180.87	0.60	0.08%
2	6054.87	139.73	0.29	6051.67	127.13	0.57	0.05%	6050.93	98.83	0.60	0.06%
3	5279.10	181.00	0.31	5273.67	127.10	0.57	0.10%	5272.63	107.57	0.63	0.12%
4	4816.80	160.57	0.31	4807.57	154.93	0.57	0.19%	4775.73	113.03	0.62	0.85%
5	4922.93	161.60	0.31	4888.97	203.10	0.56	0.69%	4884.23	218.27	0.62	0.79%
6	5526.87	144.90	0.31	5475.63	130.00	0.57	0.93%	5478.47	111.90	0.62	0.88%
7	5321.50	128.33	0.31	5300.07	101.30	0.57	0.40%	5226.20	181.70	0.61	1.79%
8	4815.60	197.23	0.31	4739.70	133.23	0.57	1.58%	4750.97	133.43	0.60	1.34%
9	4523.60	190.20	0.31	4420.83	212.43	0.60	2.27%	4324.40	183.90	0.62	4.40%
10	4775.03	156.40	0.31	4733.73	160.47	0.59	0.86%	4715.87	161.37	0.62	1.24%
11	5341.87	144.57	0.31	5298.03	93.20	0.61	0.82%	5299.80	136.80	0.61	0.79%
12	5066.63	170.80	0.30	4870.33	141.77	0.59	3.87%	4867.23	187.17	0.60	3.94%
13	5002.00	182.30	0.30	4889.57	153.20	0.60	2.25%	4903.97	193.10	0.63	1.96%
14	4100.90	214.77	0.31	4059.97	216.20	0.58	1.00%	4051.93	190.97	0.61	1.19%
15	5347.50	120.83	0.32	5218.97	124.00	0.60	2.40%	5340.90	165.37	0.61	0.12%
16	4479.77	184.37	0.31	4476.70	153.63	0.59	0.07%	4471.97	171.87	0.60	0.17%
17	5241.97	184.93	0.31	5362.47	137.67	0.59	-2.30%	5340.90	116.60	0.61	-1.89%
18	5482.07	207.70	0.31	5420.70	185.37	0.60	1.12%	5329.77	161.43	0.60	2.78%
19	5689.93	210.10	0.32	5675.63	174.63	0.59	0.25%	5675.33	142.53	0.61	0.26%
20	5639.60	153.87	0.30	5606.20	170.93	0.60	0.59%	5643.60	139.83	0.62	-0.07%
平均	5103.40	172.82	0.31	5058.70	153.36	0.58	0.90%	5052.02	154.83	0.61	1.04%

[註]：成本值改善幅度為與不使用區域搜尋之成本值比較

另本研究也針對各顧客數水準中有無使用區域搜尋機制之實驗設立對組，以 t 檢定之方式測定在 95% 之信心水準下其中之數據是否有顯著之差異，意即以上述兩種方式使用本研究錯誤！找不到參照來源。節中設計之區域搜尋機制對於演算法是否有求解品質與效率上之助益，而相關之檢定結果如表 5.11、錯誤！找不到參照來源。與表 5.13 所示。

表 5.11 各數量顧客水準不使用與單步驟區域搜尋對組之 t 檢定

各數量顧客點 區域搜尋對組	成本值		最佳解出現時機	
	雙尾檢定 p-value	95%信心 水準結果	雙尾檢定 p-value	95%信心 水準結果
$n = 31$ ：不使用 vs. 單步驟	0.0048	有差異	0.0065	有差異
$n = 37$ ：不使用 vs. 單步驟	0.0000	有差異	0.0000	有差異
$n = 43$ ：不使用 vs. 單步驟	0.0000	有差異	0.0000	有差異
$n = 49$ ：不使用 vs. 單步驟	0.0000	有差異	0.0000	有差異

表 5.11 為不使用區域搜尋與單步驟區域搜尋對組之比較，其中於成本值檢定的各水準之對組中，雙尾檢定 p-value 皆小於 0.05，意即(不使用區域搜尋 vs. 單步驟區域搜尋)之對組數據於 95%之信心水準下有顯著之差異，故以單步驟之方式使用區域搜尋機制可以明顯改善演算法求解之品質。另最佳解出現時機檢定的各水準之對組中，雙尾檢定 p-value 亦皆小於 0.05，即(不使用區域搜尋 vs. 單步驟區域搜尋)之對組數據於 95%之信心水準下有顯著之差異，故以單步驟之方式使用區域搜尋機制亦可以明顯改善演算法求解之效率，使演算法最佳解較早出現並提早收斂。

表 5.12 各數量顧客水準不使用與多步驟區域搜尋對組之 t 檢定

各數量顧客點 區域搜尋對組	成本值		最佳解出現時機	
	雙尾檢定 p-value	95%信心 水準結果	雙尾檢定 p-value	95%信心 水準結果
$n = 31$ ：不使用 vs. 多步驟	0.0044	有差異	0.0593	無差異
$n = 37$ ：不使用 vs. 多步驟	0.0000	有差異	0.0000	有差異
$n = 43$ ：不使用 vs. 多步驟	0.0000	有差異	0.0000	有差異
$n = 49$ ：不使用 vs. 多步驟	0.0000	有差異	0.0000	有差異

而錯誤! 找不到參照來源。為不使用區域搜尋與多步驟區域搜尋對組之比較，其中除了 $n = 31$ 之最佳解出現時機未有顯著差異外，其餘各水準下，成本值與最佳解出現時

機皆有顯著差異，意即以多步驟之方式使用區域搜尋機制亦可以明顯改善演算法求解之品質與效率。

表 5.13 各數量顧客水準單步驟與多步驟區域搜尋對組之 t 檢定

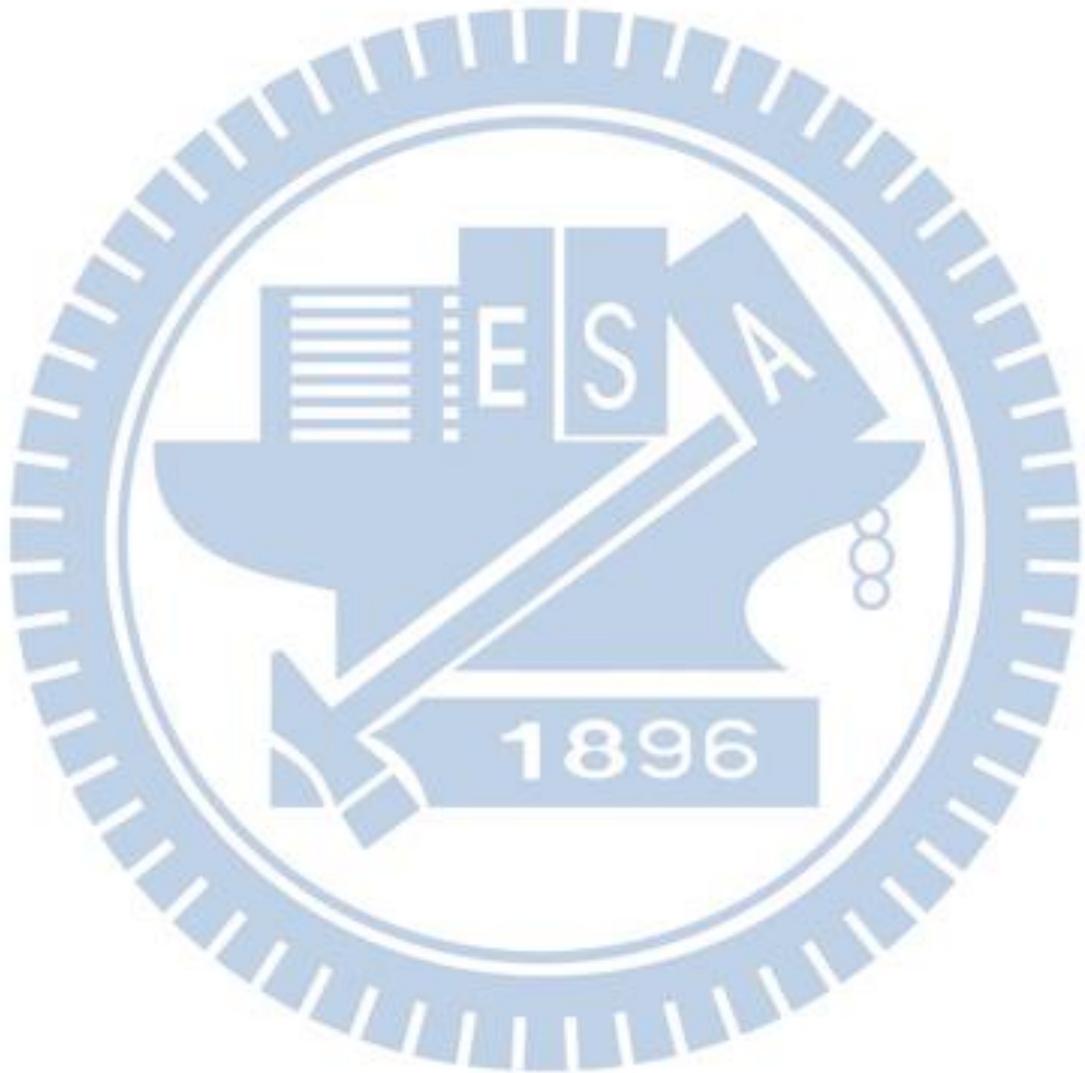
各數量顧客點 區域搜尋對組	成本值		最佳解出現時機	
	雙尾檢定 p-value	95%信心 水準結果	雙尾檢定 p-value	95%信心 水準結果
$n = 31$: 單步驟 vs. 多步驟	0.5310	無差異	0.8500	無差異
$n = 37$: 單步驟 vs. 多步驟	0.4913	無差異	0.4641	無差異
$n = 43$: 單步驟 vs. 多步驟	0.6787	無差異	0.1020	無差異
$n = 49$: 單步驟 vs. 多步驟	0.2645	無差異	0.5239	無差異

最後由於表 5.10 與附錄三之表皆顯示以多步驟執行區域搜尋之方式成本值改善幅度略優於單步驟，但其求解時間也略高，因此本研究針對兩種區域搜尋使用方式執行對組之比較如表 5.13，意即(單步驟區域搜尋 vs. 多步驟區域搜尋)，其中可發現無論於成本值之檢定抑或最佳解出現時機之檢定，各水準之對組中雙尾檢定 p-value 皆大於 0.05，意即各對組數據於 95%之信心水準下皆無顯著之差異，故以單步驟之方式使用區域搜尋機制與多步驟之方式使用區域搜尋機制，演算法求解之品質與效率相當，但無論以何種方式使用此區域搜尋機制，其皆可有效增強演算法之求解效能。

5.5 小結

本章首先呈現數學模式對於小例子求解之結果，並介紹本研究使用之建立在標竿題庫上設計大例題之方式。而於使用基因演算法進行數據實驗之前，本研究亦針對迭代數、染色體數量、交配率、突變率與菁英群組比例進行參數之測定，藉以最佳參數組合確保演算法擁有最好之求解效能，並且於實驗數據測試時探討三個面相，第一項為比較數學模式與基因演算法之求解效能，而第二項為針對有無使用重疊服務區域之架構進行實驗

分析，最後則為驗證本研究所設計之區域搜尋機制之效能。然而經由實驗數據測試與驗證亦獲得如下之結論，本研究所設計之基因演算法擁有比數學模式更好之求解效率，且其求得之解亦與全域最佳解相去不遠，另使用「廣義型重疊服務區域」之架構能有效降低物流公司配送之成本，而應用區域搜尋亦能增進演算法之求解效能。



六、 結論與建議

本章將於 6.1 節中針對前述之研究進行總結，6.2 節中則針對未來研究方向作建議，用以更貼近物流業之現實問題與情境。

6.1 結論

本研究主要延伸劉庭好(2017)所提出之重疊服務區域概念，並定義「廣義型重疊服務區域(General Overlapping Service Regions)」，且將其應用於考量容量限制之車輛路徑問題，提出「考量重疊服務區域與容量限制之車輛路徑問題」之研究，透過重疊服務區域之架構中，物流車隊靈活運用之模式，增加車輛路徑安排之彈性，用以降低物流公司配送之成本，提供決策者一顧客需求配送之模式。

本研究提出一數學模型與一基因演算法，數學模型主要用於問題情境之描述與小例子實驗測試，而基因演算法為本研究主要求解「考量重疊服務區域與容量限制之車輛路徑問題」之方法，因而另提出一區域搜尋機制優化演算法，然本研究提供決策者之顧客需求配送模式包含「車輛路徑安排」與「車輛使用模式」兩部分，車輛路徑安排主要以上述之求解方法藉由基因演算法之染色體所獲得之資訊，而車輛使用模式則主要依據路徑安排之結果分析自有車輛、跨區挪用與外包租用之執行模式。另本研究於最後進行實驗數據分析，首先確認基因演算法之求解品質與效率，然而驗證本研究之主軸「廣義型重疊服務區域」確實能有效降低物流公司顧客需求配送之總成本最多至 10%，並測定執行區域搜尋機制可提升基因演算法之效能，且降低最多至 2%之總成本。

6.2 未來研究方向與建議

本研究於定義問題時設定些許假設，因此將未來研究方向建議如下，期望能更貼近物流業之現實層面：

- (1) 本研究為於已知分區架構與方式下進行探討，因此建議未來可將分區設計議題與本研究做結合，進行廣義型重疊服務區域之設計。
- (2) 本研究之情境為在已知場站位置進行車輛路徑安排，建議未來研究可納入物流場站選址問題，並與廣義型重疊服務區域概念結合，提供物流公司場站之設站規劃。
- (3) 本研究於卡車裝載方面僅以重量與體積兩方面進行限制，因此建議未來可將貨櫃裝載模式之議題納入考量，藉以更貼近物流業之操作。



參考文獻

1. 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 949-M, Université Joseph Fourier*.
2. Baker, B. M., & Ayechev, M. A. (2003). A genetic algorithm for the vehicle routing problem. *Computers & Operations Research*, 30(5), 787-800.
3. Barbarosoglu, G., & Ozgur, D. (1999). A tabu search algorithm for the vehicle routing problem. *Computers & Operations Research*, 26(3), 255-270.
4. Bard, J. F., Jarrah, A. I., & Zan, J. (2010). Validating vehicle routing zone construction using Monte Carlo simulation. *European Journal of Operational Research*, 206(1), 73-85.
5. Bell, J. E., & McMullen, P. R. (2004). Ant colony optimization techniques for the vehicle routing problem. *Advanced engineering informatics*, 18(1), 41-48.
6. Bolduc, M. C., Renaud, J., Boctor, F., & Laporte, G. (2008). A perturbation metaheuristic for the vehicle routing problem with private fleet and common carriers. *Journal of the Operational Research Society*, 59(6), 776-787.
7. Bozkaya, B., Erkut, E., & Laporte, G. (2003). A tabu search heuristic and adaptive memory procedure for political districting. *European Journal of Operational Research*, 144(1), 12-26.
8. Braekers, K., Ramaekers, K., & Van Nieuwenhuysse, I. (2016). The vehicle routing problem: State of the art classification and review. *Computers & Industrial Engineering*, 99, 300-313.
9. Brandão, J. (2009). A deterministic tabu search algorithm for the fleet size and mix vehicle routing problem. *European journal of operational research*, 195(3), 716-728.
10. Bräysy, O., & Gendreau, M. (2002). Tabu search heuristics for the vehicle routing problem

- with time windows. *Top*, 10(2), 211-237.
11. Camacho-Collados, M., Liberatore, F., & Angulo, J. M. (2015). A multi-criteria police districting problem for the efficient and effective design of patrol sector. *European Journal of Operational Research*, 246(2), 674-684.
 12. Caro, F., Shirabe, T., Guignard, M., & Weintraub, A. (2004). School redistricting: Embedding GIS tools with integer programming. *Journal of the Operational Research Society*, 55(8), 836-849.
 13. Chen, C. H., & Ting, C. J. (2006). An improved ant colony system algorithm for the vehicle routing problem. *Journal of the Chinese institute of industrial engineers*, 23(2), 115-126.
 14. Chu, C. W. (2005). A heuristic algorithm for the truckload and less-than-truckload problem. *European Journal of Operational Research*, 165(3), 657-667.
 15. Cordeau, J. F., Gendreau, M., Laporte, G., Potvin, J. Y., & Semet, F. (2002). A guide to vehicle routing heuristics. *Journal of the Operational Research society*, 53(5), 512-522.
 16. Cordeau, J. F., & Maischberger, M. (2012). A parallel iterated tabu search heuristic for vehicle routing problems. *Computers & Operations Research*, 39(9), 2033-2050.
 17. Dantzig, G. B., & Ramser, J. H. (1959). The truck dispatching problem. *Management science*, 6(1), 80-91.
 18. Eksioglu, B., Vural, A. V., & Reisman, A. (2009). The vehicle routing problem: A taxonomic review. *Computers & Industrial Engineering*, 57(4), 1472-1483.
 19. Elhassania, M., Jaouad, B., & Ahmed, E. A. (2014). Solving the dynamic vehicle routing problem using genetic algorithms. *International Conference on Logistics and Operations Management (GOL)*, 62-69.
 20. Frutos, M., & Tohmé, F. (2012). A New Approach to the Optimization of the CVRP through Genetic Algorithms. *American Journal of Operations Research*, 2(04), 495.
 21. Gen, M., & Lin, L. (2005). Multiobjective hybrid genetic algorithm for bicriteria network design problem. *Complex Int*, 11(11), 73-83.

22. Gendreau, M., Hertz, A., & Laporte, G. (1994). A tabu search heuristic for the vehicle routing problem. *Management science*, 40(10), 1276-1290.
23. Gong, Y. J., Zhang, J., Liu, O., Huang, R. Z., Chung, H. S. H., & Shi, Y. H. (2012). Optimizing the vehicle routing problem with time windows: a discrete particle swarm optimization approach. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(2), 254-267.
24. Groër, C., Golden, B., & Wasil, E. (2009). The consistent vehicle routing problem. *Manufacturing & service operations management*, 11(4), 630-643.
25. Gutiérrez-Jarpa, G., Desaulniers, G., Laporte, G., & Marianov, V. (2010). A branch-and-price algorithm for the vehicle routing problem with deliveries, selective pickups and time windows. *European Journal of Operational Research*, 206(2), 341-349.
26. Haugland, D., Ho, S. C., & Laporte, G. (2007). Designing delivery districts for the vehicle routing problem with stochastic demands. *European Journal of Operational Research*, 180(3), 997-1010.
27. Holland, J. H. (1975). Adaptation in natural and artificial systems: An introductory analysis with applications to biology. *Control and artificial intelligence*.
28. Jarrah, A. I., & Bard, J. F. (2012). Large-scale pickup and delivery work area design. *Computers & Operations Research*, 39(12), 3102-3118.
29. Jung, S., & Moon, B. R. (2002). A hybrid genetic algorithm for the vehicle routing problem with time windows. *Proceedings of Genetic and Evolutionary Computation Conference*, Morgan Kaufmann Publishers Inc., 1309-1316.
30. Kachitvichyanukul, V. (2009). A particle swarm optimization for the vehicle routing problem with simultaneous pickup and delivery. *Computers & Operations Research*, 36(5), 1693-1702.
31. Kytöjoki, J., Nuortio, T., Bräysy, O., & Gendreau, M. (2007). An efficient variable neighborhood search heuristic for very large scale vehicle routing problems. *Computers &*

- operations research*, 34(9), 2743-2757.
32. Lahyani, R., Khemakhem, M., & Semet, F. (2015). Rich vehicle routing problems: From a taxonomy to a definition. *European Journal of Operational Research*, 241(1), 1-14.
 33. Laporte, G. (2009). Fifty years of vehicle routing. *Transportation Science*, 43(4), 408-416.
 34. Laporte, G., & Nobert, Y. (1983). A branch and bound algorithm for the capacitated vehicle routing problem. *Operations-Research-Spektrum*, 5(2), 77-85.
 35. Lei, H., Laporte, G., & Guo, B. (2012). Districting for routing with stochastic customers. *EURO Journal on Transportation and Logistics*, 1(1-2), 67-85.
 36. Lei, H., Wang, R., & Laporte, G. (2016). Solving a multi-objective dynamic stochastic districting and routing problem with a co-evolutionary algorithm. *Computers & Operations Research*, 67, 12-24.
 37. Lenstra, J. K., & Kan, A. R. (1981). Complexity of vehicle routing and scheduling problems. *Networks*, 11(2), 221-227.
 38. Leung, S. C., Zhang, Z., Zhang, D., Hua, X., & Lim, M. K. (2013). A meta-heuristic algorithm for heterogeneous fleet vehicle routing problems with two-dimensional loading constraints. *European Journal of Operational Research*, 225(2), 199-210.
 39. Lin, M., Chin, K. S., Fu, C., & Tsui, K. L. (2017). An effective greedy method for the Meals-On-Wheels service districting problem. *Computers & Industrial Engineering*, 106, 1-19.
 40. Lucka, M., & Piecka, S. (2011). Ant colony optimizer with application to the vehicle routing problem. *J. Appl. Math*, 4.
 41. Lysgaard, J., Letchford, A. N., & Eglese, R. W. (2004). A new branch-and-cut algorithm for the capacitated vehicle routing problem. *Mathematical Programming*, 100(2), 423-445.
 42. Ombuki, B., Ross, B. J., & Hanshar, F. (2006). Multi-objective genetic algorithms for vehicle routing problem with time windows. *Applied Intelligence*, 24(1), 17-30.
 43. Osman, I. H. (1993). Metastrategy simulated annealing and tabu search algorithms for the

- vehicle routing problem. *Annals of operations research*, 41(4), 421-451.
44. Polacek, M., Hartl, R. F., Doerner, K., & Reimann, M. (2004). A variable neighborhood search for the multi depot vehicle routing problem with time windows. *Journal of heuristics*, 10(6), 613-627.
 45. Prischink, M. (2016). Metaheuristics for the districting and routing problem for security control. *Doctoral dissertation, Master's thesis, TU Wien, Institute of Computer Graphics and Algorithms*.
 46. Pureza, V., Morabito, R., & Reimann, M. (2012). Vehicle routing with multiple deliverymen: Modeling and heuristic approaches for the VRPTW. *European Journal of Operational Research*, 218(3), 636-647.
 47. Sahin, B., Yilmaz, H., Ust, Y., Guneri, A. F., & Gulsun, B. (2009). An approach for analysing transportation costs and a case study. *European Journal of Operational Research*, 193(1), 1-11.
 48. Schneider, M., Stenger, A., & Hof, J. (2015). An adaptive VNS algorithm for vehicle routing problems with intermediate stops. *OR Spectrum*, 37(2), 353-387.
 49. Taillard, É., Badeau, P., Gendreau, M., Guertin, F., & Potvin, J. Y. (1997). A tabu search heuristic for the vehicle routing problem with soft time windows. *Transportation science*, 31(2), 170-186.
 50. Tan, K. C., Lee, L. H., Zhu, Q. L., & Ou, K. (2001). Heuristic methods for vehicle routing problem with time windows. *Artificial intelligence in Engineering*, 15(3), 281-295.
 51. Thangiah, S. R. (1993). Vehicle routing with time windows using genetic algorithms. *Technical Report SRU-CpSc-TR-93-23, Slippery Rock University*.
 52. Thangiah, S. R. (1999). A hybrid genetic algorithms, simulated annealing and tabu search heuristic for vehicle routing problems with time windows. *Practical handbook of genetic algorithms*, 3, 347-381.
 53. Van Breedam, A. (1995). Improvement heuristics for the vehicle routing problem based

- on simulated annealing. *European Journal of Operational Research*, 86(3), 480-490.
54. Wang, S., & Wu, Y. (2017). A genetic algorithm for energy minimization Vehicle Routing Problem. *International Conference on Service Systems and Service Management (ICSSSM)*, 1-5.
55. Yao, B., Chen, C., Song, X., & Yang, X. (2017). Fresh seafood delivery routing problem using an improved ant colony optimization. *Annals of Operations Research*, 1-24.
56. Yu, B., Yang, Z. Z., & Yao, B. (2009). An improved ant colony optimization for vehicle routing problem. *European journal of operational research*, 196(1), 171-176.
57. Yu, V. F., & Lin, S. Y. (2015). A simulated annealing heuristic for the open location-routing problem. *Computers & Operations Research*, 62, 184-196.
58. Yu, V. F., Redi, A. P., Hidayat, Y. A., & Wibowo, O. J. (2017). A simulated annealing heuristic for the hybrid vehicle routing problem. *Applied Soft Computing*, 53, 119-132.
59. 劉庭好，(2017)，有重疊服務區域之多貨櫃裝載問題，國立交通大學運輸與物流管理學系碩士學位論文。

附錄一、交配率與染色體數量參數測定數據

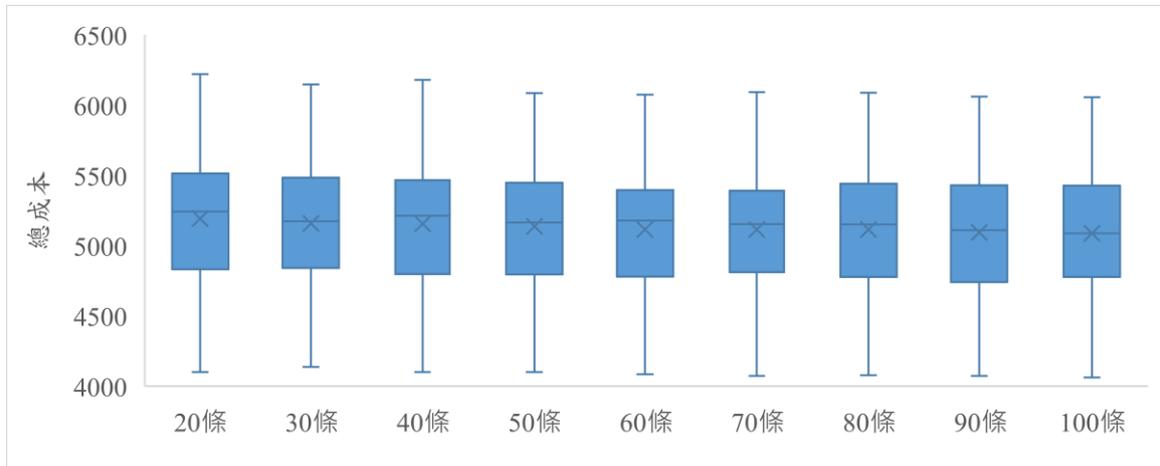


圖 A1.1 交配率 0.6 下各染色體數量所測得之 20 組數據盒鬚圖

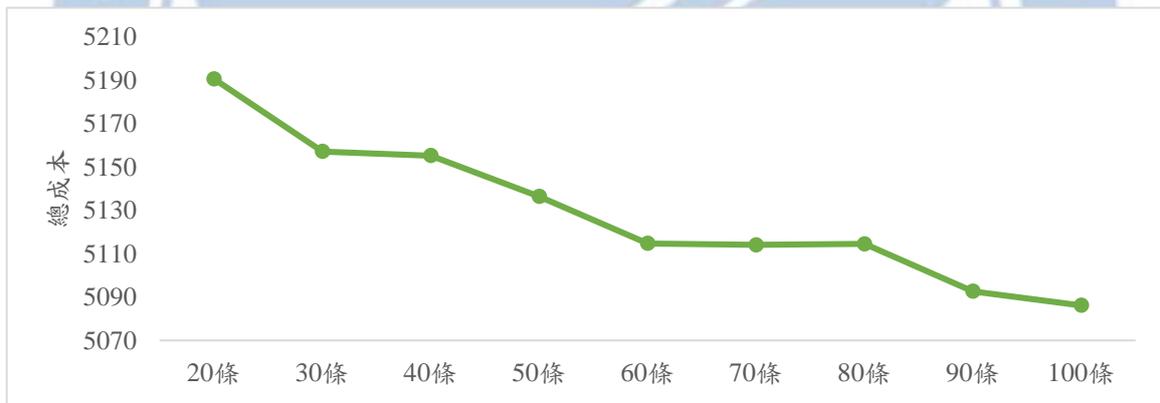


圖 A1.2 交配率 0.6 下各染色體數量所測得之平均成本值折線圖

表 A1.1 交配率 0.6 下各染色體數量對組之 t 檢定結果

染色體數量對組	雙尾檢定 p-value	95%信心水準結果	染色體數量對組	雙尾檢定 p-value	95%信心水準結果
20 vs. 30	0.0371	有差異	60 vs. 70	0.9471	無差異
30 vs. 40	0.8163	無差異	70 vs. 80	0.9601	無差異
40 vs. 50	0.1013	無差異	80 vs. 90	0.0019	有差異
50 vs. 60	0.0797	無差異	90 vs. 100	0.5902	無差異

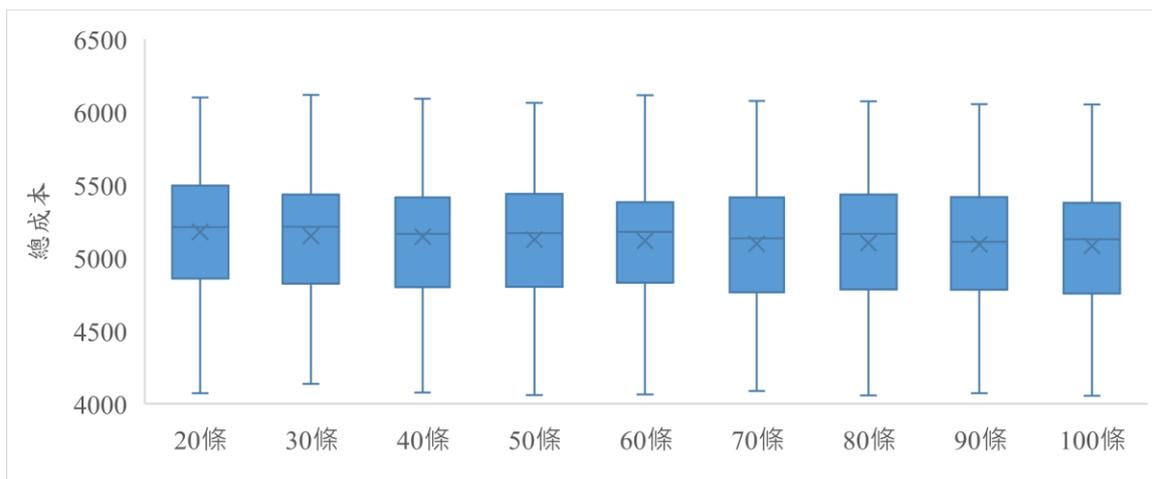


圖 A1.3 交配率 0.7 下各染色體數量所測得之 20 組數據盒鬚圖

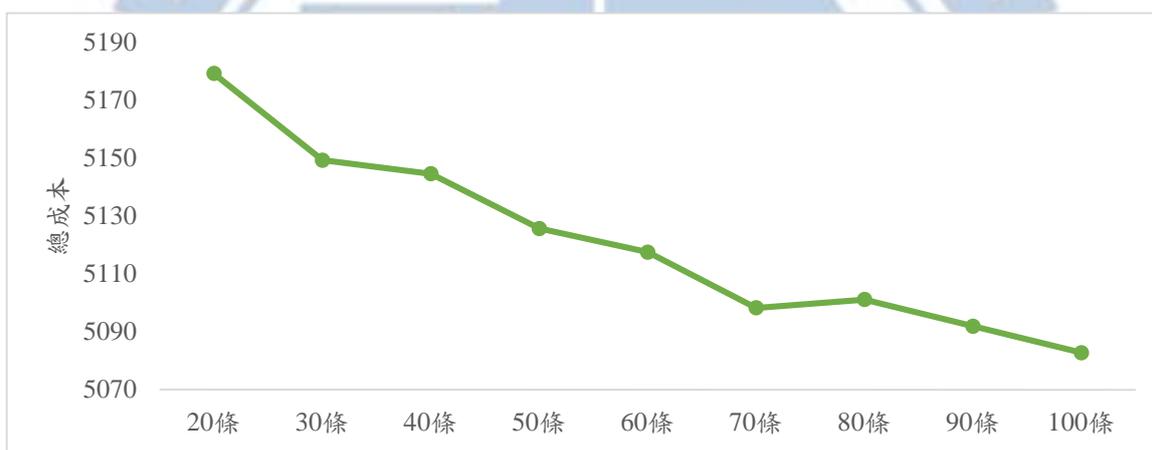


圖 A1.4 交配率 0.7 下各染色體數量所測得之平均成本值折線圖

表 A1.2 交配率 0.7 下各染色體數量對組之 t 檢定結果

染色體數量對組	雙尾檢定 p-value	95%信心水準結果	染色體數量對組	雙尾檢定 p-value	95%信心水準結果
20 vs. 30	0.0377	有差異	80 vs. 90	0.3878	無差異
30 vs. 40	0.6796	無差異	90 vs. 100	0.2805	無差異
40 vs. 50	0.1080	無差異	70 vs. 90	0.4883	無差異
50 vs. 60	0.3051	無差異	70 vs. 100	0.0065	有差異
60 vs. 70	0.0416	有差異	80 vs. 100	0.0298	有差異
70 vs. 80	0.7289	無差異			

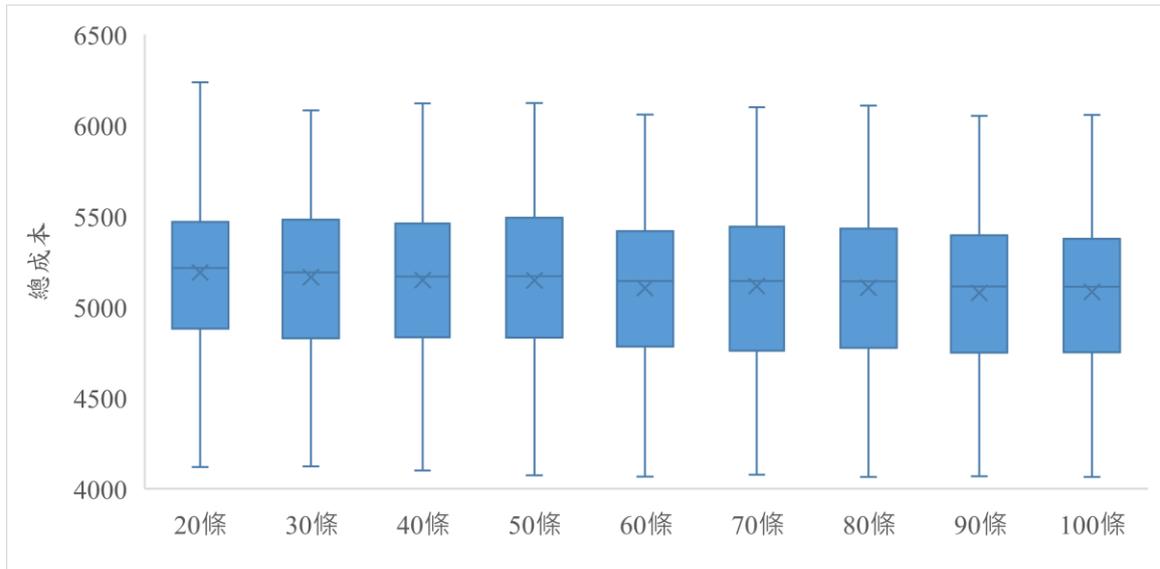


圖 A1.5 交配率 0.8 下各染色體數量所測得之 20 組數據盒鬚圖

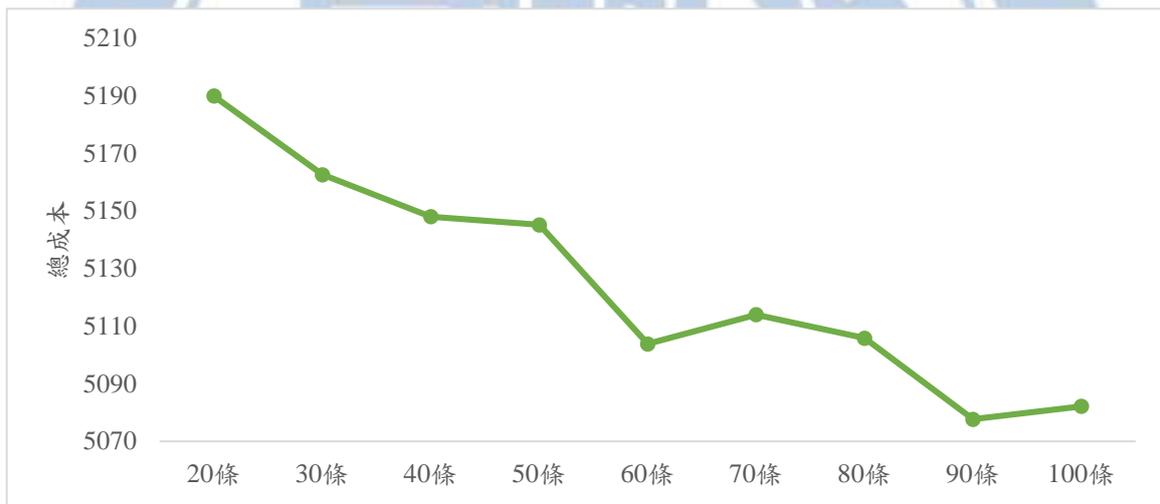


圖 A1.6 交配率 0.8 下各染色體數量所測得之平均成本值折線圖

表 A1.3 交配率 0.8 下各染色體數量對組之 t 檢定結果

染色體數量對組	雙尾檢定 p-value	95%信心水準結果	染色體數量對組	雙尾檢定 p-value	95%信心水準結果
20 vs. 30	0.0355	有差異	60 vs. 70	0.1265	無差異
30 vs. 40	0.1695	無差異	70 vs. 80	0.2642	無差異
40 vs. 50	0.8220	無差異	80 vs. 90	0.0240	有差異
50 vs. 60	0.0003	有差異	90 vs. 100	0.6798	無差異

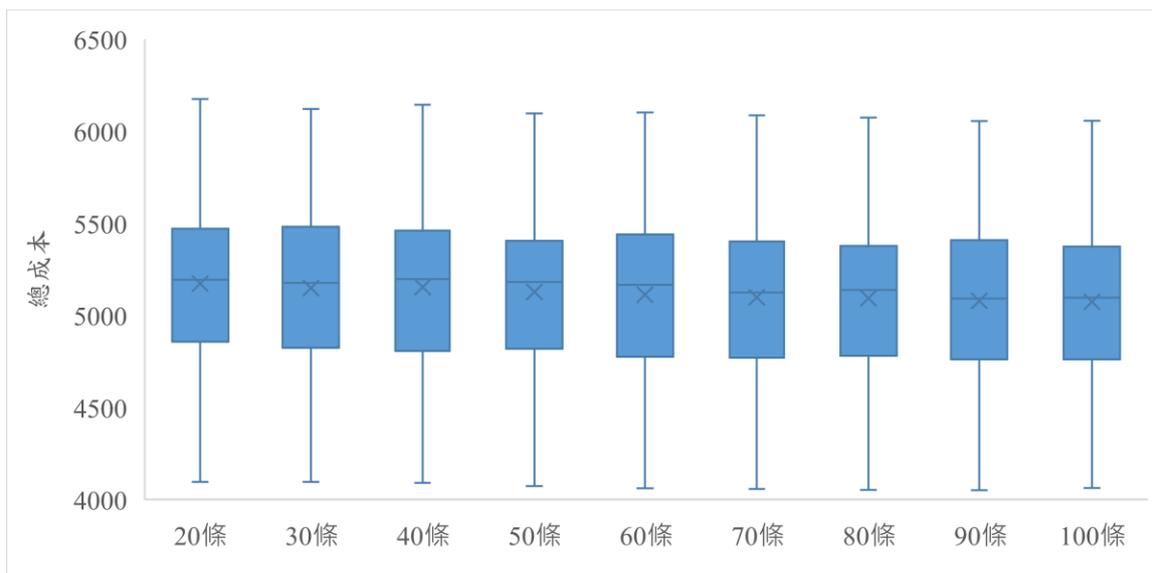


圖 A1.7 交配率 0.9 下各染色體數量所測得之 20 組數據盒鬚圖



圖 A1.8 交配率 0.9 下各染色體數量所測得之平均成本值折線圖

表 A1.4 交配率 0.9 下各染色體數量對組之 t 檢定結果

染色體數量對組	雙尾檢定 p-value	95%信心水準結果	染色體數量對組	雙尾檢定 p-value	95%信心水準結果
20 vs. 30	0.0024	有差異	70 vs. 80	0.6225	無差異
30 vs. 40	0.5694	無差異	80 vs. 90	0.1935	無差異
40 vs. 50	0.0045	有差異	90 vs. 100	0.6342	無差異
50 vs. 60	0.2085	無差異	50 vs. 70	0.0010	有差異
60 vs. 70	0.0750	無差異	70 vs. 90	0.0294	有差異

附錄二、有無考量重疊服務區域之數據

表 A2.1 有無考量重疊服務區域之數據 (n = 37)

	無重疊 成本值	有重疊 成本值	改善幅度		無重疊 成本值	有重疊 成本值	改善幅度
1	6874.13	6860.17	0.20%	12	9027.20	7799.50	13.60%
2	8637.40	7490.40	13.28%	13	6086.03	6066.20	0.33%
3	6412.90	6049.27	5.67%	14	6183.37	5746.83	7.06%
4	5495.47	5631.47	-2.47%	15	6921.57	6369.47	7.98%
5	6813.37	6096.03	10.53%	16	7110.83	6661.33	6.32%
6	7557.93	6537.93	13.50%	17	6269.60	6075.23	3.10%
7	5444.63	5368.30	1.40%	18	5924.57	5640.37	4.80%
8	6712.77	5697.83	15.12%	19	8555.40	7545.67	11.80%
9	6741.33	6191.00	8.16%	20	8167.43	7139.20	12.59%
10	7081.20	6521.20	7.91%	平均	6906.48	6338.87	7.72%
11	6112.50	5289.90	13.46%				

表 A2.2 有無考量重疊服務區域之數據 (n = 43)

	無重疊 成本值	有重疊 成本值	改善幅度		無重疊 成本值	有重疊 成本值	改善幅度
1	7120.20	7172.97	-0.74%	12	8923.17	7980.60	10.56%
2	9462.23	9105.67	3.77%	13	8629.00	8449.63	2.08%
3	7847.80	7005.80	10.73%	14	8508.10	7794.97	8.38%
4	8052.83	7748.57	3.78%	15	8572.30	7797.30	9.04%
5	8283.27	7531.07	9.08%	16	7894.57	6814.17	13.69%
6	7297.23	7128.47	2.31%	17	7290.40	6900.20	5.35%
7	9421.30	8730.93	7.33%	18	7791.30	7304.30	6.25%
8	9403.93	8929.03	5.05%	19	6874.77	6251.30	9.07%
9	8229.40	7913.70	3.84%	20	8277.57	7886.67	4.72%
10	9175.17	8592.73	6.35%	平均	8281.64	7756.64	6.31%
11	8578.30	8094.77	5.64%				

表 A2.3 有無考量重疊服務區域之數據 (n = 49)

	無重疊 成本值	有重疊 成本值	改善幅度		無重疊 成本值	有重疊 成本值	改善幅度
1	9234.90	8385.10	9.20%	12	8865.73	8947.20	-0.92%
2	9650.30	9344.13	3.17%	13	9436.50	9253.33	1.94%
3	9989.17	9588.83	4.01%	14	8808.60	8257.90	6.25%
4	9987.87	9413.63	5.75%	15	8494.37	8121.30	4.39%
5	9648.70	9353.23	3.06%	16	9536.83	9106.00	4.52%
6	9496.60	8811.83	7.21%	17	10047.57	9604.63	4.41%
7	10292.27	9936.63	3.46%	18	9749.27	9652.57	0.99%
8	10935.43	10158.53	7.10%	19	9434.13	9106.70	3.47%
9	10233.20	9915.03	3.11%	20	7930.00	7865.80	0.81%
10	9595.53	8969.07	6.53%	平均	9546.51	9164.80	3.95%
11	9563.30	9504.50	0.61%				



附錄三、有無考量區域搜尋機制之數據

表 A3.1 有無考量區域搜尋機制之數據 (n = 37)

	不使用區域搜尋			單步驟區域搜尋				多步驟區域搜尋			
	成本值	最佳解出現時機 (代)	求解時間	成本值	最佳解出現時機 (代)	求解時間	成本值改善幅度	成本值	最佳解出現時機 (代)	求解時間	成本值改善幅度
1	6979.13	221.73	0.36	6881.00	146.33	0.70	1.41%	6860.17	182.53	0.76	1.70%
2	7503.50	182.53	0.35	7463.90	143.50	0.71	0.53%	7490.40	165.83	0.76	0.17%
3	6070.07	233.70	0.36	5995.93	168.50	0.71	1.22%	6049.27	221.40	0.75	0.34%
4	5682.00	242.17	0.37	5636.00	225.73	0.72	0.81%	5631.47	220.03	0.74	0.89%
5	6110.97	229.13	0.37	6150.50	176.47	0.69	-0.65%	6096.03	181.07	0.76	0.24%
6	6801.50	245.63	0.36	6524.47	223.87	0.69	4.07%	6537.93	221.03	0.77	3.88%
7	5506.80	197.57	0.36	5451.27	190.57	0.70	1.01%	5368.30	182.80	0.80	2.52%
8	5765.83	236.20	0.36	5681.30	170.17	0.71	1.47%	5697.83	183.80	0.76	1.18%
9	6276.23	205.83	0.38	6195.23	239.27	0.70	1.29%	6191.00	184.30	0.78	1.36%
10	6652.37	206.83	0.37	6487.87	181.03	0.70	2.47%	6521.20	150.13	0.80	1.97%
11	5549.20	248.10	0.36	5445.93	205.50	0.72	1.86%	5289.90	244.13	0.78	4.67%
12	7883.33	247.70	0.36	7828.43	130.60	0.69	0.70%	7799.50	166.77	0.77	1.06%
13	6083.57	215.50	0.36	6029.83	168.60	0.70	0.88%	6066.20	193.23	0.79	0.29%
14	5876.40	249.67	0.36	5675.63	176.47	0.71	3.42%	5746.83	195.83	0.77	2.20%
15	6486.97	222.17	0.36	6394.60	168.23	0.71	1.42%	6369.47	140.20	0.78	1.81%
16	6770.87	207.10	0.36	6671.43	158.87	0.69	1.47%	6661.33	138.13	0.77	1.62%
17	6179.57	238.83	0.36	6144.20	212.77	0.70	0.57%	6075.23	207.40	0.77	1.69%
18	5659.17	205.57	0.35	5669.20	178.47	0.70	-0.18%	5640.37	193.37	0.77	0.33%
19	7671.70	210.07	0.37	7485.83	138.00	0.70	2.42%	7545.67	171.23	0.78	1.64%
20	7278.07	208.87	0.37	7134.37	158.80	0.70	1.97%	7139.20	117.00	0.77	1.91%
平均	6439.36	222.75	0.36	6347.35	178.09	0.70	1.41%	6338.87	183.01	0.77	1.57%

[註]：成本值改善幅度為與不使用區域搜尋之成本值比較

表 A3.2 有無考量區域搜尋機制之數據 (n = 43)

	不使用區域搜尋			單步驟區域搜尋				多步驟區域搜尋			
	成本值	最佳解出現時機 (代)	求解時間	成本值	最佳解出現時機 (代)	求解時間	成本值改善幅度	成本值	最佳解出現時機 (代)	求解時間	成本值改善幅度
1	7255.03	300.87	0.44	7189.17	219.10	0.80	0.91%	7172.97	228.60	0.91	1.13%
2	9209.37	314.77	0.43	9042.90	169.27	0.81	1.81%	9105.67	244.40	0.94	1.13%
3	7004.97	333.67	0.44	6959.97	227.53	0.82	0.64%	7005.80	235.40	0.92	-0.01%
4	7779.17	326.63	0.44	7734.33	214.33	0.81	0.58%	7748.57	258.47	0.94	0.39%
5	7693.67	313.03	0.44	7601.93	272.20	0.80	1.19%	7531.07	283.87	0.89	2.11%
6	7198.87	300.50	0.43	7027.83	203.40	0.79	2.38%	7128.47	223.00	0.90	0.98%
7	8971.20	350.07	0.44	8772.73	228.53	0.79	2.21%	8730.93	211.90	0.87	2.68%
8	8939.93	246.23	0.43	8851.33	194.33	0.80	0.99%	8929.03	196.97	0.94	0.12%
9	8003.87	331.70	0.44	7899.73	226.33	0.79	1.30%	7913.70	269.17	0.90	1.13%
10	8752.87	266.67	0.43	8635.73	162.00	0.79	1.34%	8592.73	222.53	0.91	1.83%
11	8259.10	281.33	0.43	8074.87	209.37	0.79	2.23%	8094.77	187.43	0.92	1.99%
12	8325.23	290.33	0.43	8067.20	179.50	0.80	3.10%	7980.60	197.77	0.91	4.14%
13	8581.13	303.17	0.42	8499.17	242.83	0.82	0.96%	8449.63	237.90	0.90	1.53%
14	7899.17	299.47	0.44	7804.47	236.97	0.81	1.20%	7794.97	173.60	0.91	1.32%
15	7918.50	253.10	0.43	7866.90	208.43	0.80	0.65%	7797.30	241.97	0.91	1.53%
16	6901.10	317.17	0.42	6800.30	191.37	0.81	1.46%	6814.17	209.77	0.94	1.26%
17	6983.00	323.93	0.44	6921.57	254.43	0.82	0.88%	6900.20	252.13	0.96	1.19%
18	7354.27	296.47	0.44	7259.90	193.97	0.80	1.28%	7304.30	221.30	0.93	0.68%
19	6311.43	303.90	0.43	6301.70	211.33	0.82	0.15%	6251.30	237.60	0.95	0.95%
20	8010.27	263.70	0.43	7920.03	229.97	0.79	1.13%	7886.67	191.67	0.91	1.54%
平均	7867.61	300.84	0.43	7761.59	213.76	0.80	1.32%	7756.64	226.27	0.92	1.38%

[註]：成本值改善幅度為與不使用區域搜尋之成本值比較

表 A3.3 有無考量區域搜尋機制之數據 (n = 49)

	不使用區域搜尋			單步驟區域搜尋				多步驟區域搜尋			
	成本值	最佳解出現時機 (代)	求解時間	成本值	最佳解出現時機 (代)	求解時間	成本值改善幅度	成本值	最佳解出現時機 (代)	求解時間	成本值改善幅度
1	8556.93	327.17	0.48	8450.17	282.43	0.90	1.25%	8385.10	264.90	0.98	2.01%
2	9556.57	378.87	0.47	9430.03	267.10	0.90	1.32%	9344.13	273.57	0.95	2.22%
3	9784.87	336.43	0.47	9580.63	243.63	0.87	2.09%	9588.83	225.93	0.96	2.00%
4	9664.63	347.43	0.48	9451.23	248.80	0.87	2.21%	9413.63	237.60	0.99	2.60%
5	9449.47	362.73	0.47	9342.87	258.13	0.87	1.13%	9353.23	262.63	0.98	1.02%
6	8973.10	330.83	0.48	8825.37	246.90	0.88	1.65%	8811.83	248.03	1.02	1.80%
7	10071.67	325.20	0.50	9917.53	240.60	0.87	1.53%	9936.63	247.70	1.01	1.34%
8	10445.30	285.53	0.48	10206.80	265.57	0.87	2.28%	10158.53	247.67	1.04	2.75%
9	10208.13	324.93	0.47	10037.63	233.30	0.87	1.67%	9915.03	219.13	1.01	2.87%
10	9076.13	383.63	0.47	8918.70	265.50	0.87	1.73%	8969.07	233.67	1.00	1.18%
11	9550.13	353.63	0.48	9525.13	225.73	0.87	0.26%	9504.50	187.30	0.98	0.48%
12	9134.07	305.37	0.48	9002.70	236.70	0.88	1.44%	8947.20	254.63	0.99	2.05%
13	9334.20	328.10	0.48	9246.03	241.77	0.87	0.94%	9253.33	240.67	0.99	0.87%
14	8396.07	340.37	0.47	8258.80	239.10	0.86	1.63%	8257.90	236.73	0.98	1.65%
15	8286.93	352.53	0.48	8093.20	250.40	0.87	2.34%	8121.30	265.37	1.00	2.00%
16	9358.40	332.23	0.46	9197.50	198.47	0.86	1.72%	9106.00	256.63	1.00	2.70%
17	9842.83	324.70	0.46	9641.07	272.03	0.87	2.05%	9604.63	213.67	1.02	2.42%
18	9754.33	351.57	0.46	9655.77	236.47	0.91	1.01%	9652.57	267.13	1.01	1.04%
19	9223.77	345.67	0.47	9132.83	250.10	0.89	0.99%	9106.70	260.50	0.99	1.27%
20	7967.57	360.40	0.47	7697.73	282.70	0.90	3.39%	7865.80	268.40	1.01	1.28%
平均	9331.76	339.87	0.47	9180.59	249.27	0.88	1.63%	9164.80	245.59	1.00	1.78%

[註]：成本值改善幅度為與不使用區域搜尋之成本值比較