國立交通大學

運輸科技與管理學系

碩 士 論 文

以拉式鬆弛演算法求解車輛定線問題

A Lagrangian Relaxation Based Heuristic for Vehicle Routing Problem

研 究 生：曾筠予

指導教授：黃寬丞 博士

中 華 民 國 九 十 四 年 七 月

以拉式鬆弛演算法求解車輛定線問題

A Lagrangian Relaxation Based Heuristic for Vehicle Routing Problem

研 究 生：曾筠予　　Student：Yun-Yu Tseng

指導教授：黃寬丞　　Advisor：Kuan-Cheng Huang

國 立 交 通 大 學

運輸科技與管理學系

碩 士 論 文

A Thesis
Submitted to Department of Transportation Technology and Management
College of Management
National Chiao Tung University
in partial Fulfillment of the Requirements
for the Degree of
Master
in
Transportation Technology and Management

July 2005
Hsinchu, Taiwan, Republic of China

中華民國九十四年七月

以拉式鬆弛演算法求解車輛定線問題

學生：曾筠予　　　　　　　　　　　指導教授：黃寬丞 博士

國立交通大學運輸科技與管理學系

摘　要

車輛定線問題之混合整數規劃模式，在處理較大規模的問題時，其求解效能並不理想。根據文獻，大多數路由以及排班問題，可藉由集合涵蓋問題來求解，並且集合涵蓋問題已有相當成熟且效能不錯之求解方法，如拉式鬆弛法即是其中之一。

因此本研究將車輛定線問題模式，以集合涵蓋問題描述，而後以常用之拉式鬆弛法為基礎，並結合列運算技巧之觀念來設計一啟發式演算法，求解過程中只產生「部分候選車輛巡迴路線」，並利用拉式鬆弛演算法遞迴運算的資料為指標，對集合空間進行調整，使得演算法解能逐步逼近最佳解。本研究最主要之貢獻在於新建一簡單之啟發式解法，在其中顧客分群後並未求解旅行推銷員問題，而是透過逐步的修正來提升車輛巡迴路線之適當性。

本研究之演算法經數值範例測試求解後，演算法在 50 個顧客數以下具穩定之求解品質，且與最佳解皆相當接近，後續研究可進一步加強更多車輛定線問題之測試以及演算法之修正與調整。

關鍵字：車輛定線問題、整數規劃、集合涵蓋模式、拉式鬆弛演算法

A Lagrangian Relaxation Based Heuristic for Vehicle Routing Problem

Student：Yun-Yu Tseng　　　　　　　　Advisor：Dr. Kuan-Cheng Huang

Institute of Transportation Technology and Management
National Chiao Tung University

Abstract

Vehicle routing problem can be modeled by integer programming. However, as the problem size grows, the performance of the solution algorithms tend to unsatisfactory. According to the literature, set covering problem (SCP) is a good way to model many routing or scheduling problems. Particularly, SCP model has been well studied for years, and Lagrangian relaxation is one of the effective methods.

Therefore, we use SCP to model the vehicle routing problem and develop a Lagrangian relaxation based heuristic. We also adopt the concept of column generation, 'generating a partial set of feasible routes to solve the problem.' We employ the information from the iterative procedure of Lagrangian relaxation as the indices to improve the solution space. In the iteration of the solution procedure, we modify the solution without solving traveler salesman problems.

The algorithm is tested on the test instances of the literature. Our heuristic has good and stable performance for the vehicle routing problems with 50-or-less customers. To sum up, the major contribution of this study is to design a simple but effective Lagrangian relaxation based heuristic. The further research can be focused on performing more numerical experiments of various vehicle routing problems and refining the heuristic algorithm.

Key Words: Vehicle Routing Problem, Integer Programming, Set Covering Problem, Lagrangian Relaxation.

誌　謝

首先要感謝恩師 黃寬丞教授的悉心指導與教誨，使得本論文得以順利完成。每當學生遇到研究上的瓶頸時，與老師討論後，總是能找到解決的方法或是有另一番新的看法及發現，使得研究進度得以持續。指導老師提供許多寶貴的意見，使得本論文能井然有序且完整地表達本研究的過程內容及結果。在此，再次感謝老師這兩年來的指導。

論文口試，承蒙系上 王晉元及 徐巧鶯兩位教授細心地審閱，對有所缺失的地方不吝指教與斧正，並提供許多寶貴的意見及看法，使得本論文的內容更臻完備充實。

在兩年的碩士生活中，也要感謝系上所有老師的照顧，不管在學業或是做人處世上皆有相當的收穫。研究進行過程中，也要謝謝學長、同學及學弟，小六、建名、雯瑋、健綸、明昌等不厭其煩的幫助，同樣也是造就這篇論文的幕後功臣。

最後，將此篇論文獻給我的家人和崚，謝謝你們不斷給我支持及關愛，讓我能有動力去克服各種的困難，順利地完成這篇論文，取得碩士學位。未來，希望在各個人生階段中都能有你們陪伴成長，也期許自己能帶給你們更多。

筠予　2005.7

于 新竹交大

目　錄

表目錄

圖目錄

第一章 緒論

1.1 研究動機

當前經濟環境隨著全球化世紀與資訊數位時代的發展，企業在產品的研發、上市進而運送，已到了以小時甚至以分鐘計算。多數資訊產品在市場上的生命週期已由兩年縮短至三個月以內，產品價格也多呈現等比級數下降，過去跨國企業內部需一至二星期才得已傳遞之訊息，已可在數秒內傳達完畢，消費者亦可藉網路訂製產品，且要求以最快速度送達指定地點。在產品週期短與消費者期盼快速收到貨品的需求之下，運輸配送作業的重要性可見一斑，同時運輸配送作業良好，也代表著企業更能在產品銷售上掌控得宜，也代表服務品質的優良。

而運輸配送作業，學術上意指「物流」(logistics)，就狹意而言即指商業流程中的**倉儲及運輸**，從工廠生產製造出成品，透過一個對集貨、理貨、庫存、配送等具專業運作之單位，配送至零售單位，以求降低後勤作業的成本，並提高後勤支援的效益。此一過程稱為販賣物流，從事販賣物流的單位被稱為Distribution Center，即所謂之「物流中心」,「配送中心」,「發貨倉庫」等等。

近年來隨著區域經濟的快速發展，原來分散的、低效率和高成本的物流活動轉化成物流資源互補整合、相互聯系、分工協作的產業鏈條，形成以供應鏈管理為核心的社會化物流系統。現代物流活動逐漸從生產、交易和消費過程中分化出來，成為專業化的新型經濟活動。美國物流協會對物流有更寬的解釋：配合顧客之要求，以計劃、執行、管制等管理流程，俾對貨品、服務及相關資訊等，由生產地點與消費地點間，有效率、效益的流通及儲存。

國內目前所指的物流業，即是指狹義面中滿足「銷售需求」的物流中心。是故物流業的定義，是指物品流通過程中提供支援服務的行業。更進一步申述，其任務是將貨品或產品，由製造業送至零售業或使用者的流通過程中，提供了產品集散、產品開發、產品計劃、管理、採購、保管、流通加工、暫存及配送等功能。

在企業運作中，物流被看成是企業與其供應商和客戶相聯系的能力。物流作業可分成三個領域：「**配送**」,「**製造**」和「**採購**」。這三個領域的結合使在特定位置和地點、供應源和客戶之間進行材料、半成品和成品等運輸的綜合管理成為可能。企業通過存貨的移動(存貨流)使物流過程增值。

本研究動機在於欲使企業於物流作業中的「**物流配送**」一環效率良好，幫助按**總成本**創造客戶價值，於短時間內規劃出適當車輛路線，即在派車前將分析訂單資料及路線經計算後再指派路徑，大幅減少巡迴、送貨時間，以及降低因巡迴、送貨逾時所造成的費用支出，此問題為「車輛定線問題」(vehicle routing problem, VRP)所追求之目標，而車輛定線問題之設計與演算就成為最重要的關鍵。

近半世紀以來，在車輛定線問題的課題上，國內外有許多相關文獻在這個領域裡有豐富的研究成果。此類問題本身具有高度的求解複雜性，是屬於NP-hard的問題

類型，在一般的VRP問題中，會由場站(depot)派出一組車輛來服務一組顧客，而且考慮在每個顧客點只能服務一次的限制下，必須要完成所有顧客點的服務，在以上的條件下求解出車輛路線路徑之最佳解或是良解。但是隨著顧客點增加，路線的組合數也成指數上升，造成解題的難度提高，所需要之求解演算時間隨問題變數數量呈指數關係成長，使得問題無法在具效率的時間下求得精確解（exact solution）。

所以當問題規模擴大時，為了確保在可接受的時間內可以得到精確度高的近似解(approximate solution)，發展解題上具效率性的啟發式解法(heuristic method)成為研究車輛定線問題相關課題的重點之一，需要發展有效率的啟發式解法，以確保在可接受時間內求出接近最佳解的答案，Laporte等人於1998年提出，未來在車輛定線問題方面的研究，將朝向發展具有演算策略更簡單、更精簡之計算量、可解更大規模的問題及適用於任何問題結構等特性之演算法發展，即使需要犧牲某種程度的解答精度，所以本研究的重點所在即是發展一更簡略計算量之演算法求解問題，但犧牲的解答精度能於可接受範圍。

1.2 研究目的與範圍

車輛定線問題相關問題眾多，包括基本車輛定線問題（VRP）、具時間窗限制車輛定線問題（vehicle routing problem with time windows, VRPTW）、多場站車輛定線問題（multiple depot vehicle routing problem, MDVRP）、多車種車輛定線問題（fleet size and mix vehicle routing problem, FSMVRP）、週期性車輛定線問題（periodic vehicle routing problem）等，都是在實務應用上重要的車輛定線問題類型。

本研究之研究對象為基本車輛定線問題，即僅考慮單一場站、單一車種、以最小路線成本為目標、固定節線成本與固定客戶點需求、有車輛容量限制但無最大時間限制或時間窗限制，亦無車輛數限制。此種問題屬於高複雜度之組合最佳化問題，特色為問題描述容易，但求解相當困難，在求解空間中，其決策變數為整數變數，其解具有排列及組合的特性。車輛定線問題已被證明屬於 NP-hard 問題，亦即目前尚無法在多項式時間（polynomial time）內求得最佳解的演算法。

近來，而學者們常使用啟發式解法搭配搜尋法如基因演算法(genetic algorithm)、禁忌搜尋法(tabu algorithm)來求解，成效亦頗佳。而在本研究中目的也為發展一啟發式解法求解車輛定線問題，以已有良好發展之集合涵蓋模式與拉式放鬆法為基礎之演算法，針對基礎的車輛定線問題，提出數學規劃模式，以啟發式解法等求解並分析問題。

1.3 研究方法與流程

首先將過去有關車輛定線問題相關的文獻，進行探討，並作為本研究之參考。

本研究方法為發展一啟發式解法求解車輛定線問題，並針對不同顧客需求的車輛定線問題，提出數學規劃模式，發展為集合涵蓋問題(set covering problem)，並以拉式鬆弛演算法、貪心法則、次梯度法之啟發式解法等求解並分析問題。

研究流程如圖1-1所示，主要內容與進行流程簡述如下：

1. 文獻回顧：蒐集國內外文獻中已發表之 VRP 相關研究，以及新近求解集合涵蓋問題之發展情況。
2. 啟發式解法之解題架構設計：提出數學規劃模式，發展為集合涵蓋問題(set covering problem)，並以拉格藍式鬆弛演算法、貪心法則、次梯度法之啟發式解法等求解並分析問題。
3. 解題程式撰寫：以電腦語言將整個演算法寫成電腦程式，借助電腦計算能力，以利測試試驗之進行。
4. 測試實驗之設計：為了有系統了解本研究發展之車輛定線問題啟發式解法之解題特性與績效，找出最佳組合方式。
5. 綜合分析：利用數值測試對本研究所發展之演算法進行測試，比較本演算法與其他以發表之演算法之執行解題績效，檢討其發展可行性與應用潛力。
6. 結論與分析：根據綜合分析所得到之結果，提出具體結論與建議，並研擬未來相關領域之研究方向。

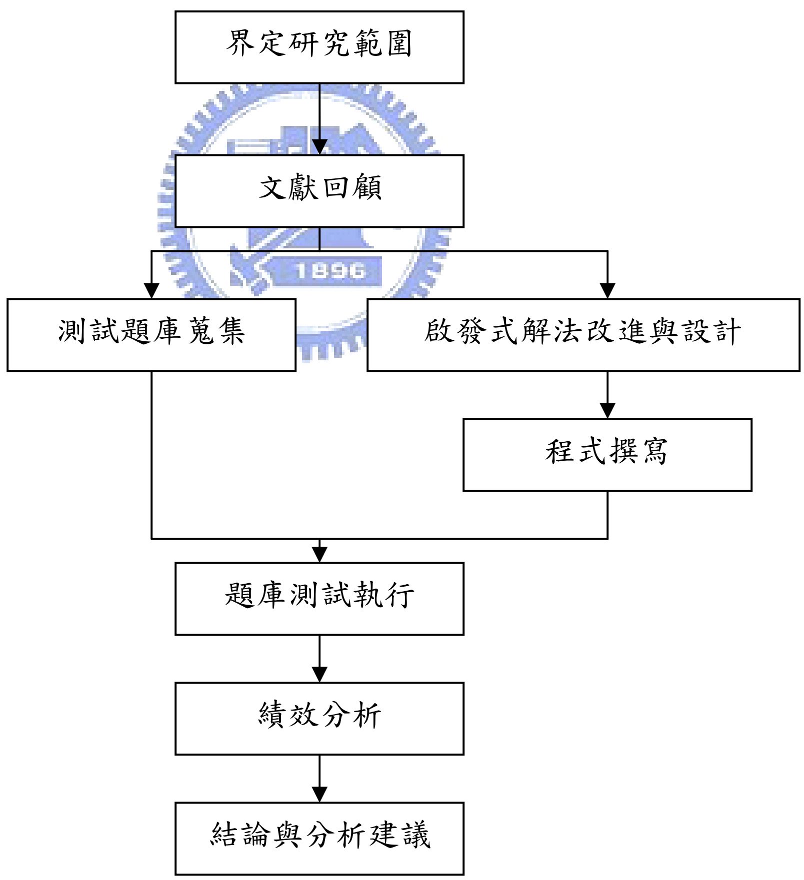


圖 1-1 研究流程圖

第二章 文獻回顧

車輛定線問題的研究領域非常廣泛，無論是實務上的物流配送業、快遞服務業，或是學術上的工業工程、交通運輸領域中，其研究與應用都相當廣泛，因此本研究將先對以往研究過之車輛定線問題作回顧及探討，本研究將相關文獻探討分為二部分：(1) 車輛定線問題文獻回顧；(2) 車輛定線問題相關演算法文獻回顧，以下針對這二部分分別說明如以下。

2.1 車輛定線問題

傳統車輛定線問題是延伸組合旅行推銷員問題的最佳化問題，即是在一個場站下，對數量眾多的顧客需求點進行純粹送貨或收貨的作業，並尋找最短總途程距離。

在這個問題中，各途程需求點之需求量不可超過車容量限制，每輛車均由場站出發，服務完所指定需求點後再回到場站，且每個需求點只能由一輛車來服務。根據 Lenstra and Rinnooy(1981)文獻，VRP 是屬於 NP-hard 問題。詳見**圖 2-1 VRP 問題示意圖**，中央方形為場站（depot），圓形為客戶點。

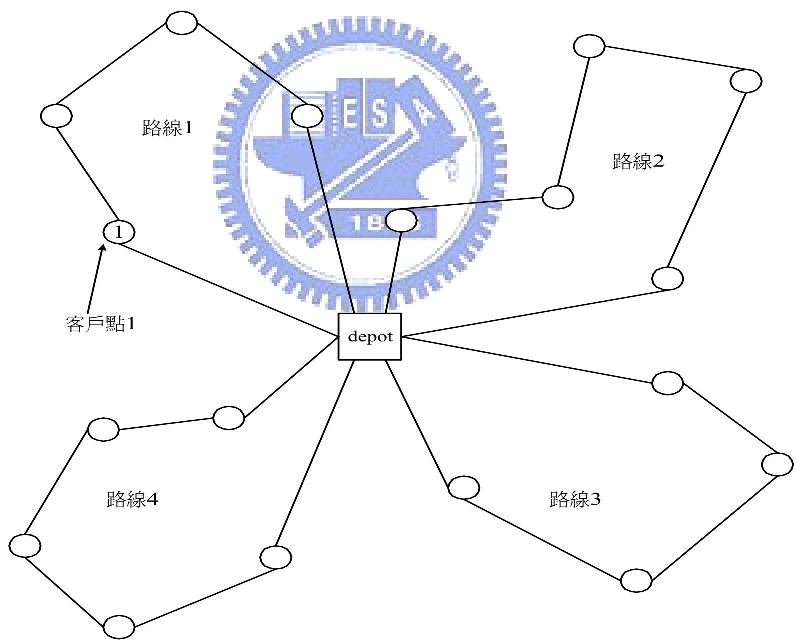


圖 2-1 VRP 示意圖

車輛定線問題定義一網路，G（V,E；D,C），V 代表網路上所有節點（vertex）之集合，E 代表網路上所有節線（edge）之集合，D 代表發生在節點上之顧客需求（demand）之集合，C 代表使用節線之一般化成本（generalized cost）集合。當 G 為一個完全性網路（complete graph），即任意兩節點之間均存在著直接連接該兩點的節線，車輛定線問題（VRP）即在上述網路 G 中，給定具有相同容量的配送車隊，然後決定出一組具有最小總成本的配送路線（自場站出發，最後回到場站）。其中每

條路線由一輛車行駛，各車所服務的顧客需求總和不得超過車輛容量；每位顧客皆須被服務，而且僅能被一輛車服務。

表2-1將以上所簡介紹之VRP研究以指標、需求點、及資源限制型態作彙整：

表 2-1 VRP 研究彙整

類型	指標	需求點型態	資源限制
VRP	總路線成本最小	固定需求 只收貨或送貨	單一場站 單一種類車輛 有車輛容量限制

2.2 車輛定線問題相關啟發式解法文獻回顧

這一小節將會舉列文獻中所研究不同問題背景的相關演算法及啟發式解法，相關車輛定線問題解法的文獻眾多，Bodin(1981)將車輛定線問題分類如下：

1. 先分群再排程（cluster first-route second）：先將需求點分成幾個群組，然後再對個群組進行途程安排，如Gillett 及 Miler (1974) 提出的掃描法(sweep procedure)。

2. 先排程再分群（route first-cluster second）：首先將所有需求點建構成一路線，接著再依限制條件如車容量，將建構的起始路線分割成數個可行的路線，如Golden *et al.*(1975)為求解多車種車輛定線問題所使用的方法。節省、插入（savings、insertion）：首先建立一起始途程，接著記算每個需求點插入途程後的節省值，考量節省值最大的需求點為插入依據，直到無法再改善為止，如Clark及Wright(1964)所提出的節省法（saving algorithm）、Mole及Jameson(1976)所提出的插入法。

3. 交換改善（improvement exchange）：必須先建立一可行途程為起始解，接著利用路線交換來改善，直到無法再改善為止，如Lin(1965)的2-opt及3-opt、Lin及Kernighan的K-optimal(1973)、Christofides及Eilon(1969)的交換法（exchange algorithm）。

4. 數學規劃基礎（mathematical-programming based）：利用數學模式進行最佳化作業，如拉格蘭式鬆弛法(lagrangian relaxation)。

5. 人機互動（interactive optimization）：利用使用者的直覺、經驗、以及知識來納入求解過程的一個方法。

6. 最佳解方法（exact procedure）：在可行域中搜尋出最佳解，如分支界限法(branch and bound)。

另外根據 Fisher(1995)的分析指出求解車輛途程與規劃問題之演算法可以分成三個時代。第一代是從 1960 年代到 1970 年代，屬於簡單啟發式方式，包括有各種局部改善之啟發式法和貪婪法(Greedy)；第二代是從 1970 年代到 1980 年代，屬

於一種以數學規劃為主的啟發式解法，包括指派法則（generalized assignment）、集合分割法（set partitioning）和集合涵蓋法(Set Covering)；第三代是從 1990 開始至今，屬於較新的發展方法，包括利用嚴謹啟發式方法、人工智慧法及最佳化演算法。

第一類：簡單啟發式解法（simple heuristic）：

發展時間約為60年代至70年代之間，是利用貪心(greedy)法則、局部改善(local improvement)等概念，設計出簡單、易執行的啟發式解法。具代表性的解法有：Clarke及Wright(1964)的節省法、Christofides及Eilon(1969)的交換法、Gillett及Miller(1974)的掃描法和Mole及Jameson(1976)所提出的插入法。另外Savelsbergh (1985)、Solomon (1987)、Kontoravdis及Bard (1992)、Potvin及Rousseau (1995)、Russel(1995)和Baker及Sheasby(1999)所發表的文章都應用到此類的解法。

第二類：數學規劃基礎解法（MP-based heuristic）：

發展時間約自 70 年代中期至今，第二類的演算法是以數學規劃為主，是將 VRP 的問題鬆弛（relax）成較簡單的 MP 模式，如一般化指派問題，或集合分割問題、集合涵蓋問題，以便將需求點分群，再針對各子群求解 TSP 子問題，代表性的研究有：Fisher 及 Jaikumar(1981)、Christofides *et al.*(1985)、Kolen (1987)、Desrochers (1992)、Konskosidis (1992)、Bramel (1996, 1997)、Fisher (1997) 和 Kohl (1997) 所發表的文章都應用到此類的解法。

第三類：人工智慧啟發式解法（artificial intelligence techniques）：

發展時間約自80年代開始。本階段的研究主要是在於人工智慧型啟發式解法的發展，一方面是借重專家系統（expert system）的建立，輔助不同問題個案建議最適合之求解方法，另一方面則是改變傳統局部搜尋方法，屬於萬用啟發式解法(metaheuristic)，主要是利用模擬退火法(simulated annealing)、基因演算法(genetic algorithms)和禁忌搜尋法(tabu search)來求解，如Pureza及Franca (1991)提出一個move-generation的改善程序，這個改善程序是對途程間作單一個節點的交換，或是嘗試將一途程裡的一個節點插入到別的途程裡，並利用禁忌搜尋法的反覆搜尋來找尋全域的最好解，其他著名的解法還有：門檻接受法、成本擾動法及搜尋空間平滑法等。

Gendreau *et al.*(1992)提出一個禁忌搜尋法來求解VRP，在他們的搜尋改善中，他們嘗試每次將一個路線的一個節點插入至另一條路線中，以對總路線做改善。Osman (1991, 1993a)利用路線間單一個節點的交換改善法來對總路線做改善，在改善過程中，他加入了禁忌搜尋法與模擬退火法的混合搜尋法，來使改善的解能夠跳脫局部最佳解。Van Breedam (2001)利用Savelsbergh (1988)所提出的三種路線間的改善法，再加入模擬退火法的機制來做改善。這三個改善法分別是線段交換法(string exchange)、線段移轉法(string relocation)、混合線段交換及移轉法(string mix)。線段交換法是考量將某一條途程的顧客點或兩個以上相鄰的顧客

點，與另一條途程的顧客點或兩個以上相鄰的顧客點作交換。線段移轉法是將起始解裡某一條途程的一個顧客點或是一群相鄰的顧客點，嘗試將它們從原途程中釋放，並插入其它某一條途程裡的相鄰兩個顧客點之間。而混合線段交換及移轉法則是混合考量線段交換法與線段移轉法的方法。根據Van Breedam(2001)的測試，當加入模擬退火法的機制後之線段交換法、線段移轉法、混合線段交換及移轉法，所得到的解均較只以單純做這三種改善要好。

以下本研究依據Fisher學者之分類加以介紹相關文獻，如2.2.1至2.2.3。

2.2.1 簡單式啟發式解法

所謂「簡單式啟發式解法」，在分類上屬於第一類啟發式解法，搜尋策略為先分群後建構路線、插入節省等原則，設計出簡單易執行的啟發式解法，優點為運算架構簡單，缺點為求解精確度不高且易陷入局部最佳的狀況。下面各節詳細說明幾種著名的傳統啟發式解法，掃描法、節省法及節線交換法。

A.掃描法（sweep algorithm）

1974 年 Billy E. Gillett 和 Leland R. Miller 首次發表了掃描法，討論了掃描法之啟發式解題邏輯，並測試掃描法的解題績效。

一個具效率的啟發式演算法名為掃描法。掃描法的特色為：1.計算量隨著節點數呈線性增加 2.每個節點位置以極座標定義。掃描法是最簡單也是最被廣為流傳的兩階段演算法(two phase method)之一，整個演算法的第一個階段為集群化，掃描法集群化的做法為：客戶點以極座標表示，隨機選取某一客戶點為起點，以該客戶點與場站之間之射線為基準，進行順時針方向或逆時針方向掃描，掃描到的客戶點即加入該群中，直到該群客戶點總需求量達到車輛容量限制，則該群集群完成，以上述原則，再開始下一集群之集群化動作，集群化的動作直到所有客戶點皆加入集群中為止。第二階段達建構路線，在第一階段客戶點集群化動作完成之後，每一群之客戶點以旅行推銷員問題之演算法進行路線構建。掃描法演算流程為：

1. 將每個節點依角度大小編號由 2 號開始（1 號節點永遠是 depot）。
2. 在符合車輛容量限制條件下，由二號節點開始以順時針（或逆時針）方向掃描角度連接各節點，形成路徑。
3. 為使總路徑長度縮短的情況下，進行路線間節點交換，來改善路線成本。
4. 分群組完成後，再進行每個群組之 TSP 問題最佳解求解。

掃描法演算流程如圖 2-2。

其他兩階段演算法的研究尚有 Christofides *et al.*(1979)和 Tyagi(1968)等。

B.節省法（saving algorithm）

節省法是由Clarke 及 Wright在1964年首先提出，屬於直接建構路線的貪心（greedy）方法，所謂節省值即指合併兩節點後所能減少的路線成本。首先計算出所有點對（i,j）的節省值：$S_{ij}=C_{i0}+C_{oj}-C_{ij}$，按照節省值由大至小排列，然後選擇序列中節省值最大之點對構建新路線，並依序檢查下一組點對合併到路線端點之可行性（主要是檢查是否符合車輛容量限制與是否為連接場站之節點），若可則合併，否則掠過該點對。若序列中尚有點對未撿視完畢，則繼續檢查下一對點對。若所有點對皆有車輛服務，則停止。若尚有節點未有車輛路線，則由未有車輛路線的點對中選擇節省值最大者產生新的路線，再檢視節省值序列中之點對合併到新路線之可行性，重複上述步驟直到所有客戶點皆受到車輛路線服務為止。

節省法演算流程如圖 2-3。

其後有許多對於Clarke 及Wright的基本節省法模式加以調整的研究，Gaskell(1967)以及Yello(1970)獨立地介紹一個節省法的修正觀念：$S_{ij}-\theta C_{ij}$，θ是一個數量參數(scalar parameter)，藉由改變θ值，使用者可以任命兩點之間的旅行成本之重視程度大小，這個參數可以改變，同時會得到不同解，再由這些解裡面選出最佳解。Golden *et al.*(1977)已經使用電腦科學計算技術來減少Clarke 及Wright的基本節省法模式的運算時間。

Altinkemer 及 Gavish(1991)在 Clarke 及 Wright 的基本節省法模式中，達到於單次重複中多對點合併的明顯程度改善計算實行結果。

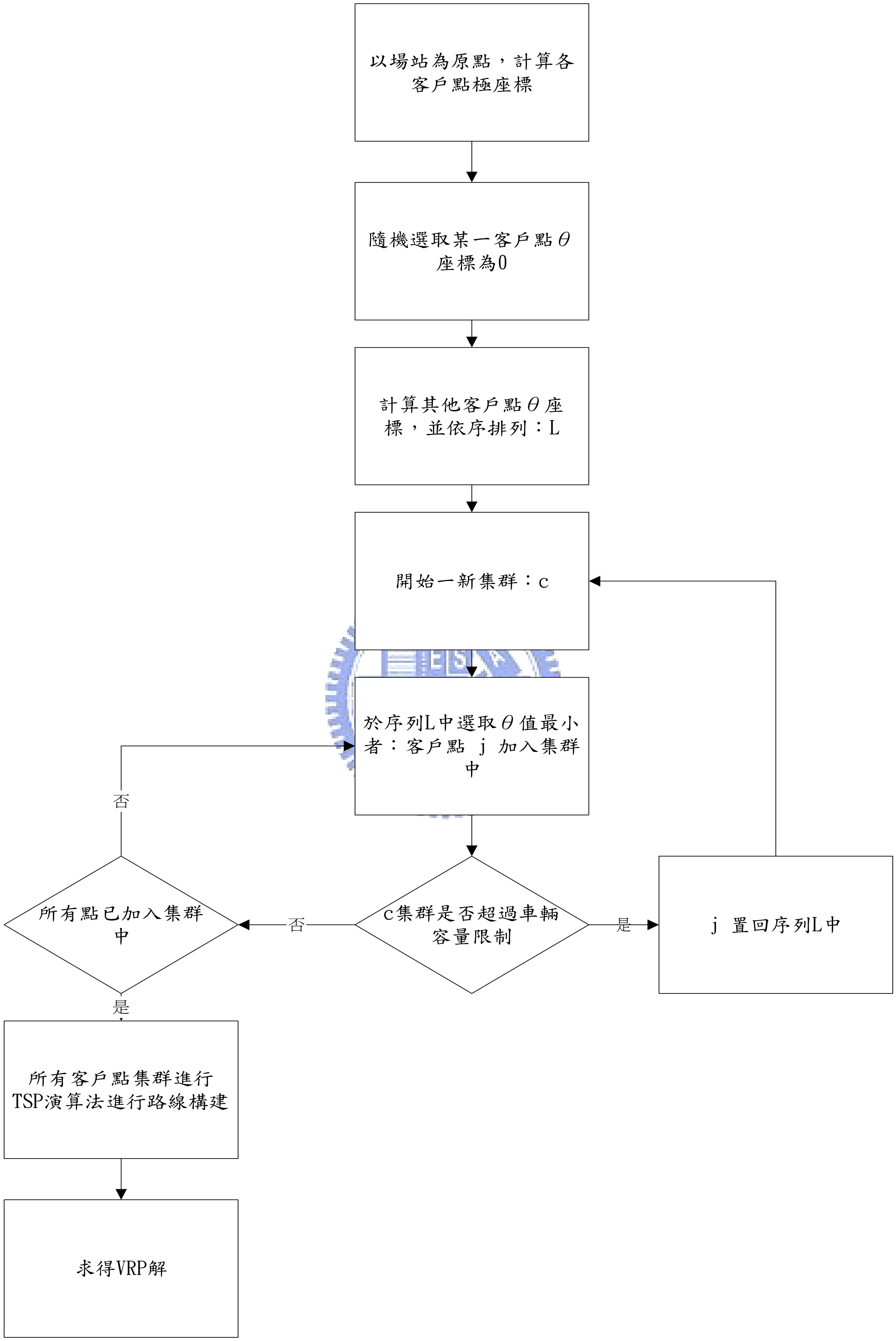


圖 2-2 掃描法演算流程圖

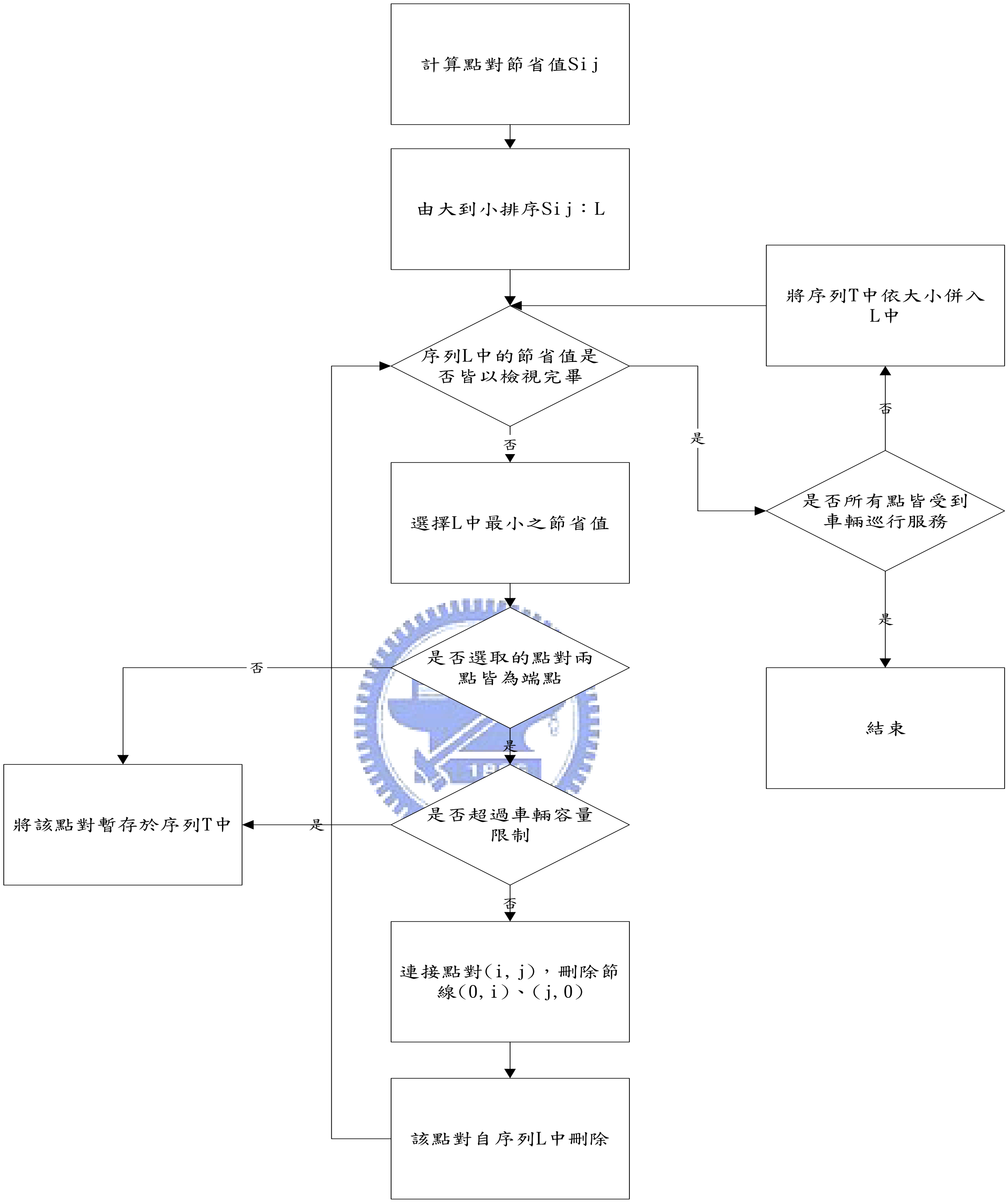


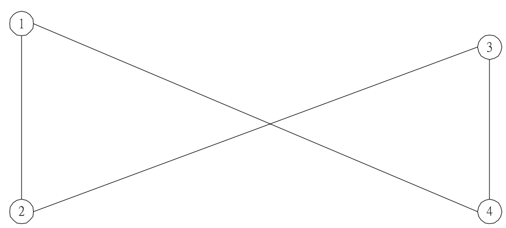
圖 2-3　節省法演算流程圖

C.K-Opt 節線交換法

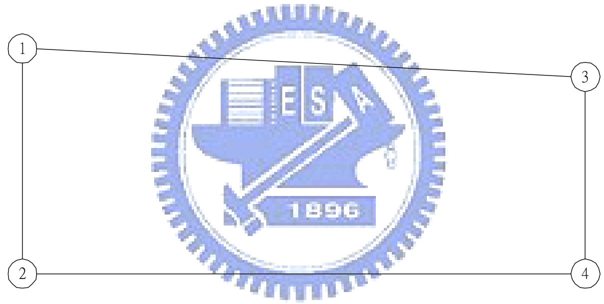
K-Opt 節線交換法是由 Lin 在 1965 年所提出，其中 K 代表的每次交換的節線

數目，一般常用者為 K 為 2 或 3，此交換法原先設計用於旅行推銷員問題上，而對於車輛定線問題而言，可用在路線內的節線交換。

以圖 2-4 說明 2-Opt 的解題觀念由圖 2-4(a)中可看出，若刪除(1,4)及(2,3)兩條節線，然後連接（1,2）及（4,3）兩條節線，將可維持一個路線的完整性且可能節省路線成本。根據這樣的觀念，對任一條路線而言，可依上述的觀念交換路線上任兩條不相鄰的節線，然後檢查交換後得到之新解是否優於交換前之解，若是，則更新解；若否則維持原解，繼續交換其他條節線直到所有可能交換之節線皆檢視完成。



(a)



(b)

圖 2-4　2-Opt 的解題觀念

Christofides 及 Eilon(1969)使用 Lin 及 Kernighan(1973)的 3-Opt 交換方法來求解 VRP 問題解，其中加入一個要求為任何交換的節線不可以超越車輛容量限制。Russell(1977)改善 Christofides 及 Eilon(1969)的程序，他選擇性的考量一些包含三節線以上的交換。更多的改善方法研究可見 Savelsberg(1985)和 Thompson(1988)的報告。

表 2-2 中的第一個啟發式方法結果為 Christofides *et al.*(1979)的節省法，第二個為 Gillett 及 Miller(1974)的掃描法，第三個啟發式方法為 Christofides 及 Eilon(1969)，第四個啟發式方法為 Russell(1977)，掃描法的解是使用不同起始顧客 n 次執行後最佳的那個解，求解時間是做這些運算所需時間和，應該注意的是 Altinkemer 及 Gavish(1991)使用大型機器運算 Clarke-Wright 基礎的平行運算已經達到問題五解為 1351，但是他們並無揭露出求解計算時間為多少。

表 **2-2** 簡單式啟發試解法計算比較 **(Fisher,1995)**

			Clarke & Wright		Gillett & Miller		Christofides & Eilon		Russell	
Problem	n	k	Cost	CPU[a]	Cost	CPU[a]	Cost	CPU[b]	Cost	CPU[c]
1	50	5	585	0.8	532	12.2	556	120	524	15
2	75	10	900	1.7	874	24.3	876	240	854	244.8
3	100	8	886	2.4	851	65.1	863	600	833	100
4	150	12	1204	6.6	1079	142.0	d	d	d	d
5	199	17	1540	11.0	1389	252.2	d	d	d	d
6	100	10	831	2.4	937	50.8	d	d	d	d

[a] CDC 6600 秒 [b] IBM 7090 秒 [c] IBM 370/168 秒 [d] 尚未有解

2.2.2 數學規劃基礎啟發式解法

到 1980 年代，Fisher 及 Jaikumar(1981)提出以數學規劃為主之最佳化方法來處理包含大約 50 個顧客點之問題。將 VRP 問題放鬆(Relax)成較簡單的 MP 模式，如一般化指派問題、集合分割問題或集合涵蓋法，以便將需求點分群，然後再針對各群求解 TSP 子問題。

數學規劃基礎啟發式解法與簡單式啟發式解法特點差異大，此類方法是解出車輛定線問題之概似最佳解，以下介紹兩種普遍且通常優於簡單式啟發式解法之模式，第一為一般性指派問題(generalized assignment problem)，第二為集合分割問題(set partitioning problem, 簡稱 SPP)。

A. 一般性指派問題

Fisher 及 Jaikumar(1981)以一般性指派問題求解車輛定線問題,顧客被分配到對應的車輛，然後再針對顧客作 TSP 問題求解，為了激發一般性指派問題概似最佳，第一個要注意的是車輛定線問題可用以下「非線性」一般性指派問題表示，定義：

$$y_{ik}=\begin{cases}1,\text{如果點}i(\text{顧客或是}depot)\text{被車輛k服務到}\\0,\text{否則爲0}\end{cases}$$

及 $y_k=(y_{0k},...,y_{nk})$，車輛定線問題可定義為

$$\min\sum_k f(y_k) \tag{2-1}$$

such that

$$\sum_i a_i y_{ik}\le b,\qquad k=1,...,K \tag{2-2}$$

$$\sum_k y_{ik}=\begin{cases}K,i=0\\1,i=1,...,n\end{cases} \tag{2-3}$$

$$y_{ik} = binary, \qquad i = 0,\ldots,n, k = 1,\ldots,K \qquad (2\text{-}4)$$

$f(y_k)$ 是最佳化推銷員問題之旅行成本，定義：

$$X_{ij} = \begin{cases} 1, \text{如果車輛k直接由點i到達點j} \\ 0, \text{否則爲零} \end{cases}$$

所以方程式 $f(y_k)$ 就可以表示為

其中 $f(y_k) = \min \sum_{ij} c_{ij} x_{ijk}$ （2-5）

such that

$$\sum_i x_{ijk} = y_{jk}, \qquad j = 0,\ldots,n \qquad (2\text{-}6)$$

$$\sum_j x_{ijk} = y_{ik}, \qquad i = 0,..,n \qquad (2\text{-}7)$$

$$\sum_{ij \in S} x_{ijk} \le |S| - 1, \qquad S \subseteq N(y_k), 2 \le |S| \le n \qquad (2\text{-}8)$$

$$x_{ijk} = binary, \qquad i = 0,..,n, j = 0,\ldots,n \qquad (2\text{-}9)$$

當然，這個定義在我們計算上並無幫助，既然我們缺乏式(2-5)之 $f(y_k)$ 的封閉表(closed form)，一般性指派問題的啟發式解法通常將 $f(y_k)$ 用一個線性的趨近式表示：$\sum_i d_{ik} y_{ik}$，加以求解此一般性指派問題而把顧客指派到相對的車輛上，為了得到線性趨近式的解，首先會指派 K 個種子(seed)顧客 $i_1,\ldots,i_k$ 到每台車輛上，假設顧客 i_k 是被指派到每台車輛 k，$k = 1,\ldots,K$，接著設定係數 d_{ik} 為插入顧客 i 到到車輛 k 由 depot 到顧客 i_k 再回去的路程，特別注意的是，$d_{ik} = c_{oi} + c_{ii_k} - c_{0i_k}$，明顯可見種子顧客決定了一開始車輛會走的方向，一般性指派問題一直持續指派顧客到此一開始的路網上。

最佳化一般性指派問題已經存在很多演算法，都是有線性目標式，例如 Fisher *et al.*(1986)使用拉格蘭式放鬆法求解一般性指派問題。我們也可以最小的 d_{ik} 為指標，持續指派顧客到車輛上來求解一般性指派問題。

選擇種子顧客的方法有許多啟發式解法，Fisher 及 Jaikumar(1981)建議用平面的方法，意指他們在平面上指派 K 個種子點 $w_1,\ldots,w_k$，而不是指派 K 個種子顧客，這些種子點的功用類似於種子顧客。

一般性指派問題可以考慮到許多限制式，例如時間窗，Nygard *et al.* (1988)還有 Koskosides *et al.*(1989)都做過此類研究。

B.　集合分割問題

集合分割啟發式解法由產生一些候選車輛路線解開始，一個候選車輛路線解可以被定義成一個集合$S \subseteq \{1,\ldots,n\}$，其中顧客都被相同的車輛運送，把這些候選車輛路線解指標為 j ，並定義以下參數：

c_j＝候選車輛路線 j 的成本

$$a_{ij} = \begin{cases} 1, \text{如果顧客i有被涵蓋在車輛路線j裡} \\ 0, \text{否則為零} \end{cases}$$

J＝所有產生的車輛數目

並讓

$$y_j = \begin{cases} 1, \text{如果候選車輛路線j被選用} \\ 0, \text{否則為零} \end{cases}$$

車輛路線問題就可以概似用以下集合分割問題表示：

$$\min \sum_{j=1}^{J} c_j y_j \qquad (2\text{-}10)$$

such that

$$\sum_{j=1}^{J} y_j = K \qquad (2\text{-}11)$$

$$\sum_{j=1}^{J} \sum_{k=1}^{K} a_{ij} y_j = 1, \qquad i = 1,\ldots,n \qquad (2\text{-}12)$$

$$y_j = binary, \qquad j = 1,\ldots,J \qquad (2\text{-}13)$$

目標式(2-10)設定極小化總運輸成本之目標，限制式(2-11)為所有選擇車輛路線巡迴數目不可超過車輛數限制，限制式(2-12)為所有顧客皆需要被涵蓋的限制式。限制式(2-12)中所使用的是等號，因此限制顧客只可被涵蓋一次，此即為集合分割問題。限制式(2-13)表示所有 y_j 為 0-1 整數變數。

典型的集合分割問題一開始由 Balinski 及 Ouandt(1964)所提出，雖然當時並未能求解大規模問題，大規模問題一直到許久之後才有人解出。Foster 及 Ryan(1976)成功應用此模式在拖車排程(truck scheduling)問題，Fleuren(1988)則應用在有時間窗的的排程問題，Fisher 及 Rosenwein(1989)也有相同應用在排程問題上。

表 2-3 整理此節介紹 Fisher 及 Jaikumar(1981)的一般性指派模式與 Foster 及 Ryan(1976)的集合分割模式之數學規劃啟發式解法比較。

關於亦屬於此節解法之集合涵蓋模式，由於此模式與本研究最為相關，故提放在 2.3 節加以說明。

表 2-3　數學規劃式啟發式解法計算比較(Fisher,1995)

Problem	n	Fisher & Jaikumar		Foster & Ryan	
		Cost	CPU[a]	Cost	CPU[b]
1	50	524	9.33	523	2.6
2	75	857	11.95	864	6.8
3	100	833	17.7	825	16.9
4	150	1014	33.6	[c]	[c]
5	199	1420	40.1	[c]	[c]
6	100	824	6.1	[c]	[c]

[a]DEC 10 秒　[b] IBM 370/168 秒　[c]尚未有解

2.2.3 人工智慧啟發式解法

自 80 年代開始，由於電腦運算能力的精進，組合最佳化問題之求解方法乃發展朝向更有效率的解題工具，其發展方向大致有二：一方面借重專家系統(expert system）的建立，對不同問題個案建議最合適求解方法；另一方面則是改變傳統的局部搜尋法，建立通用性搜尋（genenic search）的演算法，其中這些演算法與傳統演算法之間的差異在於，具有可跳脫局部最佳解的機制，但在運算過程中仍需要借助傳統的搜尋方法。以下是一些人工智慧啟發式演算法的介紹與回顧。

A.基因演算法（Genetic Algorithms,GA）

基因演算法（genetic algorithm,GA）是 Holland 教授於 1975 年和他的同事與學生共同研究出來，其基本理論以達爾文之「進化論」為根基，模擬「物競天擇，適者生存」自然遺傳之搜尋法則。GA 是以一個族群為模擬對象，並選擇族群中環境適應能力較強之個體，當作繁殖下一代之種子(親代)，經過複製、交配、突變等演化過程，產生新的下一代，如此反覆進行，最後得到適應環境最強的下一代。以演算法的角度來看，GA 中相關名詞意義為：

1. 基因（Gene)：相當於決策變數，其實際的值視編碼方法而定。
2. 染色體（Chromosome,String)：一組決策變數，相當於搜尋過程中的一個暫行解，也就是在決策空間中的一個搜尋點。
3. 族群：所有的染色體集合的集合，這也是基因最佳化方法的特色，傳統方法大多以一個暫行解單線搜尋，而 GA 是多線平行搜尋。
4. 世代（Generation)：相當於所有暫行解同時進行的一個搜尋步驟（Iteration)。
5. 適應度（Fittness)：相當於某個暫行解的目標值，目標值愈好則適應度越好。

GA 包括幾個重要部分，一個典型的 GA 流程為：

1. 將每個染色體的基因設定一個初始值;
2. 一個個世代的反覆演進;

3. 將所有過程中最好的結果輸出。

其中第二個部分「世代反覆演進」包括計算適應度、與三個基因運算：複製(Reproduction)、交配(Crossover)、突變(Mutation)，這個部分是 GA 中能跳出局部最佳解的機制，也是 GA 中最主要的一個部分。

複製運算是平行搜尋中重新分配搜尋點的重要機制，其功能主要是在模擬適者生存法則，表現出來的結果是：適應環境度愈高的染色體可以產生更多的子代，運算所需要的輸入是每個染色體的適應度，輸出則是每個染色體所產生的子代數目。最常見的方法是按比例分配，也就是該染色體所產生子代的個數比例為該染色體的適應度佔所有染色體適應度總和之比例。若第 i 個染色體的適應度為 f_i，所有染色體適應度總和為 $\sum f_j$，則該染色體所產生之子代佔全部子代的 $f_i / \sum f_j$。分配方式不只上述一種，但基本上都還是循著機率的觀念作分配，這樣的分配方式基本觀念是在於，表現好的染色體可以繁殖出更好的子代，所以表現好的染色體有較多機會繁殖子代，另外，也可以使得表現再差的染色體也有複製子代的機會，可以增加演化過程的變異性，也可以說是一項接受暫劣解的機制。

交配運算的目的在於交換部分資訊(變數數值)，需要的輸入為兩個父代染色體(parent)，輸出為一個或兩個混合後的子代染色體(child)。兩個父代之解中，交換若干個變數，產生出兩個新的暫行解即子代，以下面幾個例子說明交配運算的執行。其中兩個父代分別為(x1,x2,x3,x4,x5,x6)、(y1,y2,y3,y4,y5,y6)

父代－－－－－－－－－－－－－－－－－＞子代

1. x1x2x3|x4x5x6 + y1y2y3|y4y5y6 －＞x1x2x3|y4y5y6 + y1y2y3|x4x5x6
2. x1x2x3|x4x5|x6 + y1y2y3|y4y5|y6 －＞x1x2x3|y4y5|x6 + y1y2y3|x4x5|y6
3. x1|x2x3|x4x5|x6 + y1|y2y3|y4y5|y6 －＞y1|x2x3|y4y5|x6 + x1|y2y3|x4x5|y6

上面例子中「|」符號表示切點(cut point)，依切點數目的多寡，上面三種方式依次稱為單點交配、雙點交配及多點交配，三者差異在於切點愈多，所能產生出的子代變異性愈大，可以產生所有的可能組合。另外，交配運算的結果有可能跳出可行區域外，故在交配後必須要有檢查子代是否為可行的機制。

突變運算是模擬生物染色體內基因的變異。生物染色體可能因為外在因素(X 射線或化學污染)或內在自發性產生基因變異的現象，使得部分遺傳基因順序進行重組。突變運算與一般鄰近區域搜尋(neighberhood search)頗為類似，都是在已知之暫行解中，進行某部分之重組，傳統 GA 提供了幾種二元突變運算：

1. 變號(Invertion)：001110000 ⟶ 001100000
2. 反轉(Reversion)：001110000 ⟶ 001100100

3.搬移（Shifting）：001110000 ⟶ 001001100

使用者可以針對特殊問題發展特殊的突變運算以增加效率。

經過複製、交配及突變運算後得到的子代可能會跳脫出可行解區間，GA 的處理方式有兩種：1.設計一個限制篩選機制，檢查子代是否滿足所有限制式，滿足者才加入族群中。2.直接令在不可行解區間的子代其適應度為零，也就是沒有繁殖子代的權利，進行自然淘汰。

Baker 及 Ayechew(2003)以基因演算法求解車輛路線問題，結果顯示基因演算法能在短時間內求解出近似解。

B.禁制搜尋法（Tabu Search,TS）

禁制搜尋法是由 Glover（1989）提出，而首位將此演算法應用來求解 VRP 的是 Willard(1989)，TS 基本上可視為是鄰近搜尋法的一種變形，屬於一種人工智慧的局部搜尋方法，其觀念在於構建一智慧型的記憶機制，在搜尋鄰近解的過程中，將已經搜尋過之解記錄在禁制名單（tabu list）中，以避免重複性或毫無目的的搜尋。禁制搜尋法主要的運算在於禁制名單的運作，禁制名單是存放之前搜尋鄰近解時，利用交換法或插入法所已經搜尋過的路徑，當再尋找下一個鄰近解時，只利用不在禁制名單中的路徑進行交換或插入，以期能找到比上一解更佳的解。禁制名單的設計最主要的重點在於禁制名單中記憶的量，若禁制路徑多，侷限了鄰近解的搜尋，反而難找到下一個解甚至更佳的解；若禁制的路徑少，又無法發揮禁制名單原有的設計功能。依問題類型的不同，禁制名單記憶量可以採用固定狀態或變動狀態，採用不同的記憶體型式會影響到解題效率，故該使用何種型式，視所欲解決之問題型態而定。

Renaud *et al.*(1996)考慮在車容量及路線長度限制的條件下，應用禁制搜尋法求解多場站的車輛定線問題，並發展一個演算法 FIND，用此方法測試了 Chrisofides、Eilon、Gillettc 與 Johnson 的 11 個典型問題，發現該演算法能得到較佳的目標函數值。

Barbarosoglu及Ozgur（1999）應用禁制搜尋法來求解車輛定線問題，並探討一種新的鄰近搜尋法TANE與TANEC；TANE主要是給定一個搜尋範圍，並決定最佳的非禁制移步，以目前的解來進行搜尋的動作；而TANEC最主要的目的是尋找最佳的非禁制移步，並找出最佳的目標函數值。

亦有許多位學者，如Pureza 及 Franca(1991)Osman(1993)、、Gendreau *et al.*(1994)、Taillard *et al.*(1997)等也發表了求解VRP的TS演算法。

C.模擬退火法（Simulated Anneal,SA）

模擬退火法(SA)是一種區域性搜尋法，由 Metropolis *et al.*(1953)所提出。它可視為是一種機率性攀爬搜尋演算法(probabilistic hill-climbing search algorithm)，結合了最陡坡降法(steepest descent)與隨機過程的方式來搜尋能量函數之總體最小值。一般而言，SA 為了達到總體最小值，除了需在迭代過程做交換改善外，尚需透過波茲曼函數，使其解的狀態在某些條件下往目標函數值較劣處移動，藉以跳脫出局部最小值，再利用降溫的過程，漸漸的收斂至整體最佳解。

模擬退火法的解體架構為（以最小化問題為例）：

1. 找到一起始解 S_k ，其目標函數為 Z（ S_k ）。
2. 設定起始溫度 Ti。
3. 利用局部搜尋找到下一個可行解 S_{k+1} ，其目標函數為 Z（ S_{k+1} ）。
4. 計算目標函數差值△Z= Z（ S_{k+1} ）-Z（ S_k ），若△Z≦0，則接受新解 S_{k+1} ，若△Z＞0，則隨機產生一個變數 u~U（0,1），若 exp（-△Z/Ti）＞u，則接受新解 S_{k+1} 。
5. 檢查是否符合停止準則。

SA 可視為是鄰近搜尋法的一種變形，對一個最小化問題而言，往較低的目標值移動可視為是物質降溫結晶的過程，而最後結晶的狀態是一個局部最佳解，不同於傳統鄰域搜尋法的是，當落於局部最佳解的時候，SA 可藉由重新加熱的過程（機率性接受暫劣解）使他跳出目前局部最佳解，而有一個達到另一個更佳解的機會，由上數大略的解體架構可看出，模擬退火法其接受暫劣解是透過一個機率法則。由此可知，SA 的關鍵在於「降溫過程（Cooling Process）」與「接受法則（Acceptance Rule）」的設計。

SA 自 80 年代開始被運用在旅行銷售員問題 TSP 及 VRP 等問題上，由於具跳脫區域最佳解的特性，其解題效率比一般交換啟發式法來得快。Osman(1993)使用禁忌搜尋法與模擬退火法於交換車輛途程中的顧客，求解 VRP 問題。Chiang 及 Russell(1996)運用 SA 求解 VRPTW 問題。

表 2-4 整理此節介紹 Gendreau *et al.*(1993)、Osman (1993)與 Taillard (1993) 之人工智慧演算法比較。

表 2-4 人工智慧啟發式解法比較(Fisher,1995)

problem	tabu search					simulated annealing	
	Gendreau *et al.*		Osman		Taillard	Osman	
	cost	Cpu[a]	cost	Cpu[b]	cost	cost	Cpu[b]
1	524.61	6	524	2	524.61	528	3
2	835.32	53.8	844	3	835.26	538	107
3	826.14	18.4	835	26	826.14	829	156
4	1031.07	58.8	1044	59	1028.42	1058	84
5	1311.35	90.9	1334	54	1298.79	1378	39
6	819.56	16	819	15	819.56	826	11

[a]Silicon Graphics 36MHZ workstation minutes [b]VAX 8600 minute

2.3 集合涵蓋模式

2.3.1 集合涵蓋問題

集合涵蓋問題(SCP)可以被定義為如下，讓 $A=(a_{ij})$ 是一個 0-1 的 $m \times n$ 矩陣，讓 $c=(c_j)$ 是一個 n 維的向量。以下泛指 A 的行(row)列(column)為行列，讓 $M=\{1,...,m\}$ 以及 $N=\{1,...,n\}$，$c_j(j \in N)$ 的值代表列 j 的成本，同時假設 $c_j > 0$，對 $j \in N$，定義如果一個列 $j \in N$ 涵蓋了一個行 $i \in M$，則 a_{ij}=1，集合涵蓋問題是將全部項目 $i \in M$，利用不同的集合將其涵蓋，每個項目被涵蓋之次數至少一次也就是每個行 $i \in M$ 都被至少一個列 $j \in N$ 涵蓋到，其中每一個集合由不同項目所構成，並有其相對應的成本。SCP 需要一個列的最小成本的子集合 $S \subseteq N$，目的為求取在符合涵蓋所有項目的限制下，所選取集合之總成本為最小。

SCP 數學模式如下：

$$\min \sum_{j \in N} c_j x_j \tag{2-14}$$

subject to

$$\sum_{j \in N} a_{ij} x_j \geq 1, \qquad i \in M \tag{2-15}$$

$$x_j \in \{0,1\}, \qquad j \in N \tag{2-16}$$

SCP 數學模式類似於 2.2 節所提之 SPP 數學模式(式(2-10)~式(2-13))，目標式(2-14)類似式(2-10)設定極小化總成本之目標，限制式(2-15)為所有顧客皆需要被涵蓋的次數，值得注意的是式(2-15)中是一個≥不等式，因此顧客可被涵蓋一次以上，是標準的集合涵蓋問題，如果此不等式為=號，就是 2.2 節所謂的集合分割問題(set partitioning problem)之限制式(2-12)，限制顧客只可被涵蓋一次，如果改成≤不等式，則是所謂的集合裝運問題(set packing problem)，限制式(2-16)表示所有 x_j 為 0-1 整數變數。。

SCP 明顯是一個 NP-hard 的問題，而且有許多實務上的應用，很多演算法都已經被建議在尋找此種問題確切解的文獻，這些確切演算法可以解決多至上百個的行以及上千個列的的例子，但當處理更大規模的 SCP 問題時，便需要啟發式演算法。

2.3.2 集合涵蓋模式在 VRP 之應用

過去有相當多研究將各類問題轉換為 SCP，相似的方法被學者應用在許多排班(scheduling)問題，以及路由(routing)問題如車輛派遣等，皆可以集合涵蓋問題描述。例如 Hoffman 及 Padberg(1993)以集合涵蓋模式應用在車隊排程問題（crew scheduling problem）。對於多場站之車輛排程問題，Ribeiro 及 Soumisu(1994)也成功應用集合涵蓋模式求解。Irnich(2000)將一多點位撿貨運送問題，轉化成集合涵蓋問題後再進行求解。張育彰（2001）應用基因演算法於台

鐵列車駕駛員排班與輪班整合問題之研究中，亦使用集合涵蓋模式概念，在排班與輪班整合求解階段，將排班問題視為一集合涵蓋問題，從可行工作班集合中選擇能涵蓋到所有乘務的工作班解集合。

有關集合涵蓋法之數學模式在車輛定線問題的應用，最先被 Cullen *et al.*(1981)成功將此法應用設計車輛定線問題的啟發式解法，另外尚有 Agarwal *et al.*(1989) 基於此法發展求解車輛定線問題之精確的演算法，近年來 Bixby(1998)，Hadjiconstantinou *et al.*(1995)亦對演算法有相當程度的發展，此法普遍應用在求解數量大之問題。

對於含時間窗限制但無容量限制之車輛定線問題，Desrosiers *et al.*(1984)應用集合涵蓋問題模式求解此類問題得出最佳解。對於有容量限制車輛定線問題（CVRP）及有時間窗限制車輛定線問題（VRPTW），Desrosiers *et al.*(1992)引伸一定限分支演算法來求解 Solomon(1986)的部分含時間窗限制問題，亦可達到將近最佳或是最佳解。

應用集合涵蓋問題可以找到 VRP 的最佳解，但前提是候選車輛路線解須產生所有可行解，但對於大型問題，所有可行的候選車輛路線解數目龐大，以致求解沒有效率，甚至無法產生所有的可行候選車輛路線解。

2.3.3 集合涵蓋問題之啟發式解法

集合涵蓋問題的啟發式解法通常使會用貪心法則，(例如在每一個循環 iteration 裡，短淺地去選擇最好的下一步)；也使用交換的方法，也就是交換一個或是多個以上的列，只要目標式的解有所改善，在實務上非常快速，但是特徵是不能提供高品質的解(Fisher and Wolsey ,1982)。其他更新的啟發式解法方式還有 Feo *et al.*(1989)使用機率搜尋，Johnson *et al.*(1989)使用模擬退火法，Levineg(1992)使用基因演算法，Aourid 及 Kaminska(1994)使用類神經網路法等，但是這些應用發法之間尚未出現比較的標準，來決定出在何情況下是哪個方法比較優秀。

集合涵蓋問題亦可以用上下限夾集的方法加以尋找確切解，由前段所提之啟發式解法皆可得到好的問題解上限，通常問題解的下限值是藉由放鬆限制式而求解來的，有兩種經典的放鬆集合涵蓋問題的方法，第一是拉格蘭式放鬆法(可行解集合還是保有 0-1 的可行解性質，但是會有限制式被移到目標式去)，第二是線性放鬆法(也就是整數限制式被放鬆，而目標式則依然和原本相同)。

第一種採用拉式放鬆法(Lagrangian relaxation)以及次梯度最佳化(subgradient optimization) 是非常具有效率的的 SCP 啟發式解法，也就是跟隨 Balas 及 Ho(1980)的發展研究、Beasley(1990)、Balas 及 Carrera(1996)的改善研究。

第二種採用放鬆集合涵蓋問題線性限制（也就是移走 x_r 變數的整數限制），因為集合涵蓋問題中含有大量的變數，所以我們可使用Gimore及Gomory（1961,1963)

發展的列運算技巧(column generation，以下泛稱CG)方法求解。

以下我們參考Bramel及Simchi-Levi(1995)，將一次車輛路線巡迴視為一個集合，轉換為類似集合涵蓋問題(set covering problems, SCP)的數學規劃模式，放鬆整數限制式後，以列運算求解，詳細介紹如下：

將一次車輛路線巡迴視為一個組合，轉換為類似集合涵蓋問題(set covering problems, SCP)的數學規劃模式，以 P 為原模式代號：

i ：顧客之編號，$i=1\sim n$ ； I：所有顧客所成的集合。

j ：指派車輛之編號，$j=1\sim m$ ； J：為所有可指派車輛 j 的集合。

r ：車輛路線巡迴組合之編號； $r\in R$ ； R：可行的車輛路線巡迴組合空間(所有車輛路線巡迴組合所成之集合)。

K ：最多可派遣的車輛。

c_r ：每個車輛路線巡迴組合 r 所對應的成本。

a_{ir}：當顧客 i 包含於車輛路線巡迴組合 r 之內時，$a_{ir}=1$；否則為 0。

x_r ：0-1 整數變數。當車輛路線巡迴組合 r 被選取時，$x_r=1$，否則為 0。

(P)

目標式：

$$Min\sum_{r\in R} c_r x_r \qquad (2\text{-}17)$$

限制式：

$$\sum_{r\in R} a_{ir} x_r \ge 1 \qquad \forall i\in I \qquad (2\text{-}18)$$

$$\sum_{r\in R} x_r \le K, \qquad (2\text{-}19)$$

$$x_r \in \{0,1\} \qquad \forall r\in R. \qquad (2\text{-}20)$$

將原問題(P)之式(2-20)整數限制式放鬆後，並產生部份集合 $R'\subseteq R$ ，並以此部份集合解代入放鬆整數限制後之模式(以 P' 為代號)：

(P')

目標式：

$$Min\sum_{r\in R} c_r x_r \qquad (2\text{-}21)$$

限制式：

$$\sum_{r\in R} a_{ir} x_r \ge 1 \qquad \forall i\in I \qquad (2\text{-}22)$$

$$\sum_{r\in R} x_r \le K, \qquad (2\text{-}23)$$

$$x_r \ge 0 \qquad \forall r\in R' \qquad (2\text{-}24)$$

使用這些部分集合解代入後，讓$\bar{x}$為P'問題中的最佳解，且讓$\bar{\pi}=\{\bar{\pi}_1,\bar{\pi}_{2,\ldots,}\bar{\pi}_n\}$為限制式(2-22)相對的最佳對偶變數，讓$\bar{\theta}$為限制式(2-23)的最佳對偶變數，我們會想得知$\bar{x}$是否為原問題$P$之最佳解，為了求解此問題，觀察以下原問題$P$放鬆後的對偶形式(以$P_D$為代號)：

(P_D)

目標式：

$$\max\sum_{i\in I}\pi_i-K\theta \qquad (2\text{-}25)$$

限制式：

$$\sum_{i\in I}a_{ir}\pi_i-\theta\le c_r, \qquad \forall r\in R \qquad (2\text{-}26)$$

$$\pi_i\ge 0 \qquad \forall i\in I \qquad (2\text{-}27)$$

$$\theta\ge 0 \qquad (2\text{-}28)$$

很清楚地，如果$(\bar{\pi},\bar{\theta})$滿足限制式(2-26)，那麼就會是問題形式(P_D)的最佳解，同時$\bar{x}$就會是線性放鬆問題模式(P')的最佳解，我們如何確認$(\bar{\pi},\bar{\theta})$滿足問題形式P_D的每個限制式？可以靠著觀察一個限制式(2-29)如下，如果滿足此限制式，就知道$(\bar{\pi},\bar{\theta})$不可行：

$$\sum a_{ij}\bar{\pi}>c_r+\bar{\theta} \qquad (2\text{-}29)$$

反覆地，如果我們發現有一個列r最小化$c_r-\sum_{i\in I}a_{ir}\bar{\pi}_i$的值，而這個值比$-\bar{\theta}$小，那麼就找到違反的限制式，如此的話，$(\bar{\pi},\bar{\theta})$就不是問題$P_D$的最佳解，而符合的列可以代回問題求解，一直到沒有違反限制式(負的減少成本列)被發現，如此的話，我們就找到線性放鬆問題(P')的最佳解$\bar{x}$，也找到對偶問題形式(P_D)的最佳解$(\bar{\pi},\bar{\theta})$。

列運算再去確認一個可行的車輛路線$r\in R$滿足限制式(2-29)，定義$\bar{c}_r$為減少成本的列r(reduced cost of column)，例如：$\bar{c}_r=c_r+\bar{\theta}-\sum_{i\in S_r}\bar{\pi}_i$，對每個$r\in R$，同時定義$d(S)=\sum_{i\in S}d_i$，對每個$S\subseteq I$，$S_r$為車輛路線$r$中所出現的顧客集合，$d_i$為顧客$i$的需求量，$C$為車輛容量限制。艱難之處便在於求解式(2-30)這個列運算問題(代號為CG)：

$$\text{(CG)} \qquad \bar{c}_{\min}=Min\{\bar{c}_r:d(S_r)\le C\} \qquad (2\text{-}30)$$

關於這個列運算問題(CG)如何求解尚未明確，CG 問題本身是一個 NP-hard 的問題，因為即使給定S_r，計算$\bar{c}_r$或是c_r需要求解旅行推銷員問題(TSP)。有許多學者朝求解此問題努力，其中關於 CVRP 問題的有 Agarwal *et al.*(1989)發展一

個分枝定限法求解 CG，Bixby *et al.*(1997)發展一個切割平面法(cutting plane algorithm)求解問題 CG，此法還可以詳見 Bixby(1998)的研究。

有些執行的小技巧可以幫助我們改善求解列運算問題時的收斂，通常列運算裡的步驟三是最耗時間的，為了減少此步驟所花費的時間，我們可以採行以下方法，第一是：在步驟一時，產生一個好的起始解是非常重要的，而要辦到如此，已經有許多 CVRP 演算法可以使用了，事實上，如果可得一個好的對偶解，那麼此解便可以拿來幫助產生負減少成本的車輛路線(指在對偶問題裡)，有一些評估良好對偶變數的方法可見 Agarwal *et al.*(1989)的研究，第二是：在步驟三的每個循環裡，產生較多的負減少成本之車輛路線解，不要只產生一個。

整理 Column generation 列運算演算法，求解線性放鬆問題(P')的步驟如下：

1. 產生初始部分路線集合 R' 。
2. 以部分解 R' 代入求解 P' ，並得到最佳原始變數解，及最佳對偶變數解。
3. 解問題 CG，找出滿足減少路線成本(reduced cost)小於零的路線集合。
4. 對每個 $r \in R$ ，且減少路線成本(reduced cost)小於零的 r ，將 r 加入 R' ，回到步驟 2。
5. 如果沒有任何 r 的減少路線成本 $\bar{c}_r$ 小於零，則停止。

簡單來說，我們使用代入部分解所產生最佳的對偶變數，判斷是否有集合解應該被涵蓋，以此求解較為簡單的最佳化問題，接著再繼續求解直到沒有任何集合解可以再更減低目標值，當這個使用部分解出發的線性放鬆問題(P')的最佳解 $\bar{x}$ 被找到，目標式的值 $\sum_{r \in R'} c_r \bar{y}_r$ (應為分數形式)也就是此 CVRP 問題的下界值。

從現有產生的集合出發(或許只是總集合的一小部份集合)，再使用如切割平面法(cutting plane approach)或是分支法(branch-and-cut approach)求解為整數可行解，此法不保證會是所有集合中最佳的整數解，但是非常容易趨近於收斂。

第三章 模式建立

3.1 啟發式解法架構

車輛定線問題由於本身的計算規模與複雜度，需要一個更為簡易計算的方式，本研究參考各類文獻，欲試將此車輛定線問題以一「數學規劃基礎」的整數規劃模式-「集合涵蓋模式」，集合涵蓋模式可以較簡單地掌控成本，這也同樣是集合涵蓋模式的優點。

本研究並參考紀玟豪(2004)有關航空貨運承攬業之貨物併裝問題模式(詳見文獻注釋)，以「拉式鬆弛法(Lagrangian relaxation)」為基礎，結合貪心法則、次梯度法之啟發式解法，並結合列運算技巧之觀念：只產生「部分車輛路線巡迴解」加以求解，綜合以上發展出本研究之啟發式解法。

啟發式解法之架構圖詳見圖 3-1，本研究之啟發式解法細節於此後各節所述。

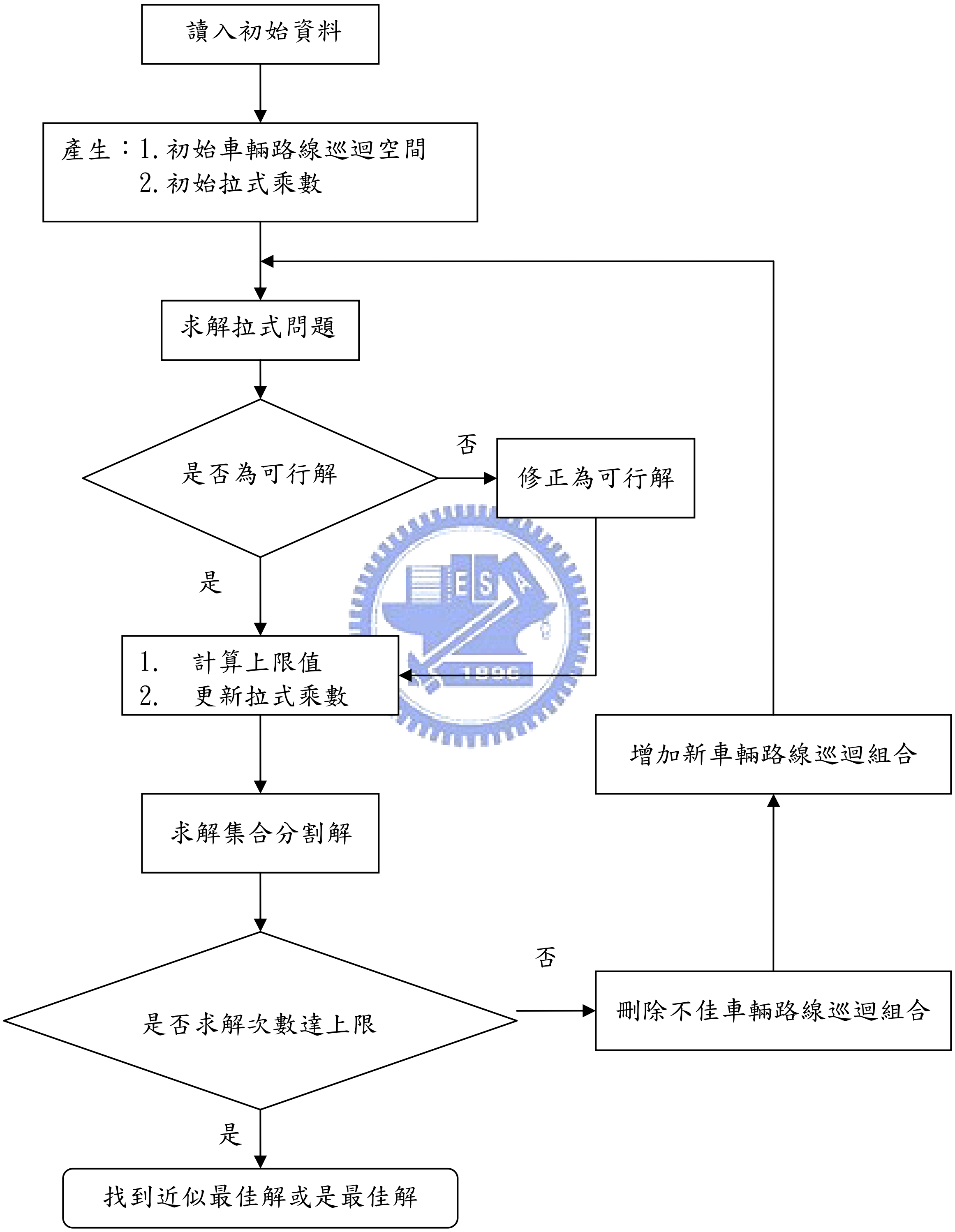


圖 3-1 啟發式解法之架構圖

3.2 拉式鬆弛模型

(一) 轉換車輛定線問題為集合涵蓋問題

將一次車輛路線巡迴視為一個組合，轉換為類似集合涵蓋問題(set covering problems, SCP)的數學規劃模式如下：

i ：顧客之編號，$i=1\sim n$ ； I：所有顧客所成的集合。

j ：指派車輛之編號，$j=1\sim m$ ； J：為所有可指派車輛 j 的集合。

r ：車輛路線巡迴組合之編號； $r\in R$ ； R：可行的車輛路線巡迴組合空間(所有車輛路線巡迴組合所成之集合)。

c_r ：每個車輛路線巡迴組合r所對應的成本。

a_{ir}：0-1 整數變數。當顧客i包含於車輛路線巡迴組合r之內時，$a_{ir}=1$；否則為 0。

x_r ：0-1 整數變數。當車輛路線巡迴組合r被選取時，$x_r=1$，否則為 0。

目標式：

$$Min\sum_{r\in R}c_r x_r \tag{3-1}$$

限制式：

$$\sum_{r\in R}a_{ir}x_r \geq 1 \qquad \forall i\in I \tag{3-2}$$

$$x_r,\ binary \tag{3-3}$$

目標式(3-1)設定極小化總運輸成本之目標，限制式(3-2)為所有顧客皆需要被涵蓋的限制式。限制式(3-2)中所使用的是不等號而非等號，也因此並未限制顧客只可被涵蓋一次。因此當求解發生顧客被涵蓋兩次以上的情形時，便需要進行修正，此即為集合涵蓋與集合分割問題(Set Partitioning Problem – SPP)之差別。限制式(3-2)使用不等式而非等式是因為一般而言求解 SCP 會較 SPP 容易。而且，即使所求得之解違反顧客僅能被涵蓋一次的限制，仍可於其後輕易藉由刪除車輛路線巡迴組合中重複之顧客，將顧客修正為僅被涵蓋一次。限制式(3-3)表示所有 x_r 為 0-1 整數變數。

本研究主題是對於式(3-1)到式(3-3)之拉式放鬆法模式，對於式(3-2)使用拉式放鬆法後的問題如下，其中以u_i代表拉式乘數，$\boldsymbol{u}$代表所有u_i所形成之向量。

目標式：

$$L(\boldsymbol{u}) = Min\sum_{r \in R} c_r(\boldsymbol{u})x_r + \sum_{i \in I} u_i \qquad (3\text{-}4)$$

限制式：

$$x_r,\ binary \qquad (3\text{-}5)$$

其中 $c_r(\boldsymbol{u}) = c_r - \sum_{i \in I_r} u_i \qquad \forall r \in R$

I_r：針對車輛路線巡迴組合r，所有被涵蓋顧客所成之集合。(I_r ={ $i \in I$：a_{ir}=1})。

求解上述拉式問題時，當$c_r(\boldsymbol{u})$ <0 則 x_r = 1；反之，$c_r(\boldsymbol{u})$ >0 或是$c_r(\boldsymbol{u})$ =0 則 x_r = 0。詳見圖 3-2 說明如下：

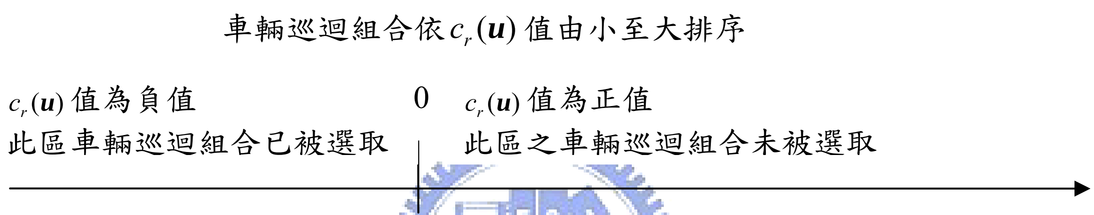


圖 3-2 起始車輛路線巡迴組合挑選圖

對於幾近最佳的拉式乘數u_r而言，拉式成本$c_r(\boldsymbol{u})$在選擇集合r的效用上可以作為一個可靠的資訊，基於這個特性，在啟發式解法過程中使用拉式成本作為參考指標，也就是依據它們可能被選為最佳解的可能性，用來當做一個貪婪的方法以找尋潛在的優良 SCP 解。

以此方式可得到一個鬆弛解$L(\boldsymbol{u})$值，因為求解出的$L(\boldsymbol{u})$值是根據放鬆限制式後求得，就該車輛路線巡迴解之集合空間所對應之問題而言，其值為下限值。但鬆弛解通常為不可行解，當發生此情形時，必須根據原來限制式修正為可行解，才能得到上限值(Upper Bound, *UB*)。於得到 *UB* 與$L(\boldsymbol{u})$值後，即可根據兩值夾出最佳解所在的區間並得到誤差範圍，其結果可分為三種：

1. 上下限值相等，即無誤差。
2. 上下限值之誤差範圍在容忍值內。
3. 上下限值之誤差範圍在容忍值外。

第一種情形表示最佳解產生，第二種則表示雖僅得到近似最佳解，但為一個可接受的近似解，該解與最佳解的最大差距會等於容忍值。最後一種情形則表示，尚未找到可接受的解，需要再繼續求解，但並不保證一定能收斂找到近似最佳解。因此若演算法設計得當，通常會找到不錯的近似解或者是最佳解，否則便會有無法收

斂的情形；此亦為拉式鬆弛法判斷是否繼續求解的停止機制。

(二)修正拉式解為可行解

因拉式問題為放鬆限制式後再進行求解，所以極可能求出不可行解。欲求得可行解，一般而言可透過拉式解的修正。然而該修正程序的設計上必須注意，一方面希望造成目標式值增加最少的情況下完成，但是又不宜太複雜，以免大幅增加運算的負荷。

本研究之演算法主要係利用 $c_r(\boldsymbol{u})$ 值的觀念來進行修正，如前述求解拉式問題時，已求出各車輛路線巡迴組合的 $c_r(\boldsymbol{u})$ 值並進行由小到大排序，在本研究所建構之模式下，$c_r(\boldsymbol{u})$ 值為負值之車輛路線巡迴組合已被挑選，此時若尚有顧客未被涵蓋，代表為不可行解，所以再從 $c_r(\boldsymbol{u})$ 值為正值之車輛路線巡迴組合中挑選，修正為可行解，挑選的遵循法則有兩個：一是其正值越小越好，二是必須涵蓋未被涵蓋的顧客，持續選取車輛路線巡迴組合，直至所有車輛路線巡迴組合涵蓋所有的顧客為止，依據以下之範例修正可行解方式由於不複雜，故較不會大幅增加運算的負荷。本研究修正可行解方法詳見圖 3-3。

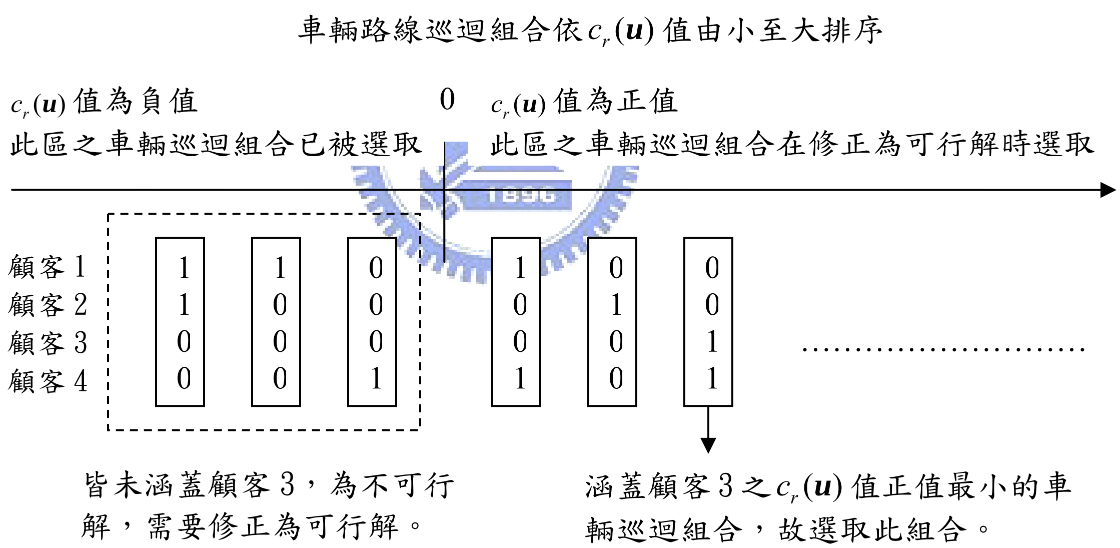


圖 3-3 修正為可行解圖

(三) 拉式乘數更新

通常被使用在短計算時間裡，尋找接近最佳乘數向量的方法是使用次梯度法，這個方法產生一序列 u^1、u^2... 等的非負拉式乘數向量，採用次梯度法後，可一次調整多個拉式乘數，對於較大規模的問題，可以得到較快的求解速度，但是不保證利用此方法求解會收斂，因此求解能否收斂，是一項值得注意的問題。

故有關拉式鬆弛法遞迴運算過程中，拉式乘數之修正，可參考(3-6)、(3-7)式

之運算(Held & Karp, 1970)，式中 t 代表第 t 次遞迴運算。

$$u_i^{t+1} = \max\left\{ u_i^t + \lambda \frac{UB - L(\boldsymbol{u}^t)}{\left\| s(\boldsymbol{u}^t) \right\|^2} s_i(\boldsymbol{u}^t),\ \ 0 \right\} \qquad \forall i \in I \tag{3-6}$$

$$s_i(\boldsymbol{u}^t) = 1 - \sum_{r \in R_i} x_r(\boldsymbol{u}^t) \qquad \forall i \in I \tag{3-7}$$

R_i：針對顧客i，所有涵蓋此顧客之車輛路線巡迴組合所成的集合，$\{ i \in I : a_{ir} = 1\}$。

a. UB所代表的是遞迴運算中最佳上限值，可利用修正不可行解後，由可行解之目標值求出。$L(\boldsymbol{u}^t)$為第t次求得之下限值，為拉式放鬆問題的現行解。經由兩值之所夾出的區間，代表最佳解所在的區間範圍。其原則是在上下限值接近時(UB -$L(\boldsymbol{u}^t)$變小時)，用來縮小u_i值調整的步幅。一般而言，如此有助於求解的收斂。

b. λ為更新$\boldsymbol{u}$值的調整係數，λ＞0 是一個給定的步幅參數(step-size parameter),主要在針對UB、$L(\boldsymbol{u}^t)$和$s_i(\boldsymbol{u}^t)$三項值做出修正細部調整，避免修正幅度過大或過小，參照文獻所述，λ值可根據求解次數或解的品質來做調整。藉由改變其值以加快收斂速度或減緩調整幅度以利於求得較佳的解，對於不同問題，也可依自行需求作設定。

c. 方程式(3-7) $s_i(\boldsymbol{u}^t)$值大小之涵意為項目i被涵蓋之次數，當其值為負代表被涵蓋超過一次，越負代表被涵蓋越多次；其值等於 0 時，表示該項目僅被涵蓋一次；而其值等於 1 時，表示該項目未被涵蓋，詳見圖 3-4。由此可知，方程式(3-7)與限制式(3-2)有相當的關係。

d. $\boldsymbol{s}(\boldsymbol{u}^t)$所代表的是$\boldsymbol{u}$值的修正方向，$\| \boldsymbol{s}(\boldsymbol{u}^t) \|^2$所表示的是行向量$\boldsymbol{s}(\boldsymbol{u}^t)$的長度平方，所影響的是調整的步幅。透過適度的修正，使得每一次求解，逐漸的往最佳解方向前進，其原則是，當該向量長度平方大時，代表方向的誤差仍大，將之置於分母，以降低步幅的大小，避免在方向誤差仍大時(也就是限制式(3-2)對大多數項目i仍不滿足)，過度地修正$\boldsymbol{u}$值。

關於 $\boldsymbol{u}$ 值的部分，於計算上必須由初始值開始，而後才有辦法做後續的更新動作，本研究將每個顧客 i 之起始u_i值，設定為由場站原點出發到顧客 i 之距離，u_i值的設定影響解收斂的狀態，但當遞迴次數持續增加此值亦會趨向平穩的值。

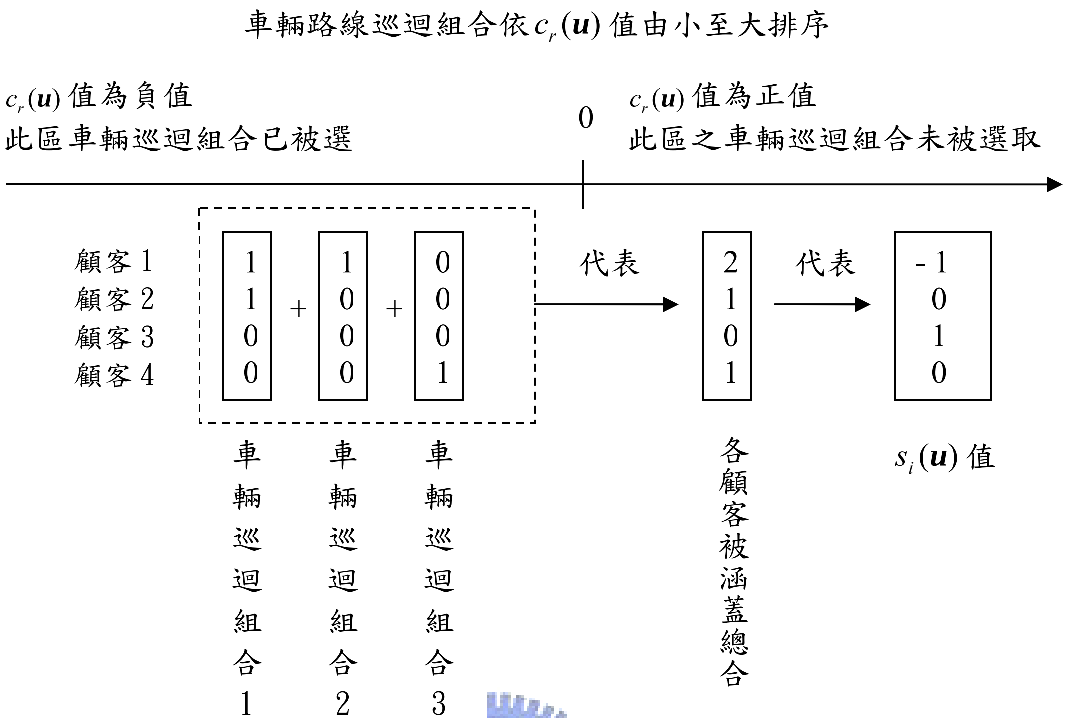


圖 3-4 $s_i(\boldsymbol{u})$ 值圖

(四)將集合涵蓋問題之可行解轉為為集合分割解

在演算法流程中，解集合分割問題的方式是對集合涵蓋問題之解，檢查是否有顧客被重複選取，若有重複，則將重複的顧客保留在 $c_r(\boldsymbol{u})$ 值最負者之車輛路線巡迴組合，再將其他車輛路線巡迴組合中含有此重複的顧客予以剔除，求解出集合分割解值。

3.3 車輛路線巡迴空間的調整

3.3.1 初始車輛路線巡迴空間

原顧客車輛路線巡迴問題雖然可藉由 SCP 加以描述，但當問題規模加大時，其車輛路線巡迴組合空間將會極大，因此本研究先產生一個初始車輛路線巡迴組合的空間，以各個顧客單趟來回(產生 C_1^n 個，n 為顧客數)以及挑選兩個顧客來回(產生 C_2^n 個，n 為顧客數)為起始車輛路線巡迴空間，因為當車輛路線巡迴組合裡只含有一個顧客或是兩個顧客時，對應 TSP 問題相當簡單，亦可輕易求出各車輛路線巡迴組合所對應的成本。

3.3.2 刪除不佳車輛路線巡迴組合

當顧客數增加，車輛路線巡迴組合空間劇烈增大，不可能保留所有車輛路線巡迴加以運算，故本研究在每次遞迴運算刪除不佳的車輛路線巡迴組合，保留一

定數目的車輛路線巡迴組合數，配合拉式鬆弛法的遞迴運算，以一個有效合理的方式，將不合適的予以淘汰，保留的車輛路線巡迴組合數大小需考慮：一是為了運算上的效率，不能保留太多，二是需保留適當的車輛路線巡迴組合數，預期好的車輛路線巡迴組合，會由這些保留下來的車輛路線巡迴組合繼續產生。

保留車輛路線巡迴組合的選取方式為，依照 $c_r(\boldsymbol{u})$ 值由小到大的排序，選取排序在前面的固定車輛路線巡迴組合數 m 個，依顧客數大小不同而調整保留固定的車輛路線巡迴組合數(m)，以此車輛路線巡迴組合數的空間為基礎，此後將還有車輛路線巡迴調整的機制。詳見圖 3-5 說明如下：

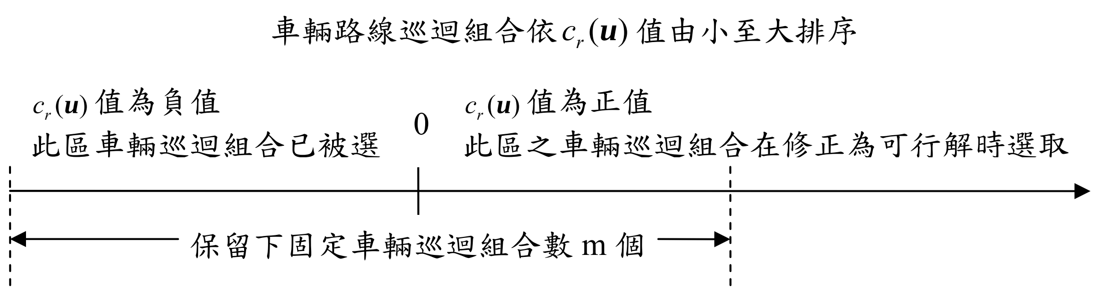


圖 3-5　車輛路線巡迴組合空間之調整圖

3.3.3 產生新的車輛路線巡迴之演算法

本研究新增具有潛力之優良車輛路線巡迴組合，以改進現行車輛路線巡迴組合的空間，逐步求得近似最佳解。產生新的車輛路線巡迴方式主要分為二種，一是在車輛路線巡迴組合刪減掉顧客的方式，二是在車輛路線巡迴組合增加顧客進去的方式。

增加或是刪減顧客的選取，是參考式(3-7)之 $s_i(\boldsymbol{u})$ 值(定義詳見 3.2 之(三))。挑選顧客 $s_i(\boldsymbol{u})$ 值為 1 者作為「加入的顧客」，挑選顧客 $s_i(\boldsymbol{u})$ 值最負者作為「刪減的顧客」，當遇上 $s_i(\boldsymbol{u})$ 值相同時，則以隨機方式挑選，隨機方式為產生一個 0~1 之間的亂數，將 0~1 之區間依 $s_i(\boldsymbol{u})$ 相同的顧客數大小均分，再檢視此亂數落於何區間，就挑選哪一個顧客。

依此方式之理由，主要是基於式 $s_i(\boldsymbol{u})$ 值所代表的不僅是顧客涵蓋限制式(3-2)是否滿足，同時也是拉式乘數 $\boldsymbol{u}$ 依據式(3-6)修正時的數值。因此依據加項產生新的車輛路線巡迴組合，因為其 $s_i(\boldsymbol{u})$ 值最大，相對 u_i 的搭配也最大，潛在性地就有可能使得新增加之車輛路線巡迴組合在下一個遞迴計算 $c_r(\boldsymbol{u})$ 值時，其 $c_r(\boldsymbol{u})$ 值越負，因而產生出較好的車輛路線巡迴組合。反之，利用刪除顧客所產生的新車輛路線巡迴組合，其 $c_r(\boldsymbol{u})$ 值也應該越負。

$$c_r(\boldsymbol{u}) = c_r - \sum_{i \in I_r} u_i \qquad \forall r \in R$$

I_r：針對車輛路線巡迴組合r，所有被涵蓋顧客所成之集合。(I_r ={ $i \in I$：a_{ir}=1})。

$$u_i^{t+1} = \max\left\{u_i^t + \lambda \frac{UB - L(\boldsymbol{u}^t)}{\left\|s(\boldsymbol{u}^t)\right\|^2} s_i(\boldsymbol{u}^t),\ \ 0\right\} \qquad \forall i \in I \qquad (3\text{-}6)$$

$$s_i(\boldsymbol{u}^t) = 1 - \sum_{r \in R_i} x_r(\boldsymbol{u}^t) \qquad \forall i \in I \qquad (3\text{-}7)$$

R_i：針對顧客i，所有涵蓋此顧客之車輛路線巡迴組合所成的集合，{ $i \in I$：a_{ir} =1}。

以下詳細說明在車輛路線巡迴組合裡，刪減及增加顧客的方法：

(一)在車輛路線巡迴組合刪減顧客

檢視當次遞迴裡顧客的$s_i(\boldsymbol{u})$值是否為負值(代表有顧客被涵蓋一次以上)，值得注意的是有此情況發生，這個在車輛路線巡迴組合刪減顧客之產生新車輛路線巡迴組合機制才會執行。滿足此前提後，再檢視$s_i(\boldsymbol{u})$值最負者(代表此顧客被涵蓋最多次)，如果有兩個以上$s_i(\boldsymbol{u})$值一樣的負值，則隨機挑選一個，將此顧客視為「要刪除的顧客」，上述已經談過，本研究中參考總顧客數大小而調整保留固定的車輛路線巡迴組合數，假設共留下 m 個車輛路線巡迴組合，在留下來的 m 個車輛路線巡迴組合集合裡全部刪去此顧客，故新產生了車輛路線巡迴組合集合，這些新產生的車輛路線巡迴組合數至多 m 個，因為以下限制之關係：

1. 首先檢視原本留下的車輛路線巡迴組合，原本就不含要刪除的顧客的車輛路線巡迴組合，則無法執行刪除顧客的機制，唯有含有要被刪除的顧客之車輛路線巡迴組合，才需要被執行刪除顧客的機制。

2. 若要刪減的顧客恰巧為車輛路線巡迴組合中唯一包含的顧客，則此車輛路線巡迴組合亦不需要被執行刪除顧客的機制。

3. 被挑選執行刪除顧客機制之車輛路線巡迴組合，尚需檢查是否與「保留下之固定車輛路線巡迴組合」重複，如果重複，則不算是新產生的車輛路線巡迴組合。

剩下之車輛路線巡迴組合就是真正因刪減顧客點後，所產生的新車輛路線巡迴組合，本研究稱這些新產生的車輛路線巡迴組合為「減點的集合」，將此併入原本保留之固定車輛路線巡迴組合裡，目前為止，倘若刪減顧客點的機制有被執行的話，保留的車輛路線巡迴組合數增長到至多 2m 個。

在車輛路線巡迴組合裡刪除顧客時，巡迴組合變動方式是直接刪除該顧客，其他剩下的巡迴組合排列不變，例如車輛路線巡迴組合為[0、5、6、3、0]，其中 0 代表場站原點，由原點 0 出發，依序經過了顧客點 5、6、3，最後回到原點 0，而如果要刪除的顧客為顧客點 3，則此車輛路線巡迴組合變為[0、5、6、0]，不再

另外求解 TSP 最佳走法，詳見圖 3-6。

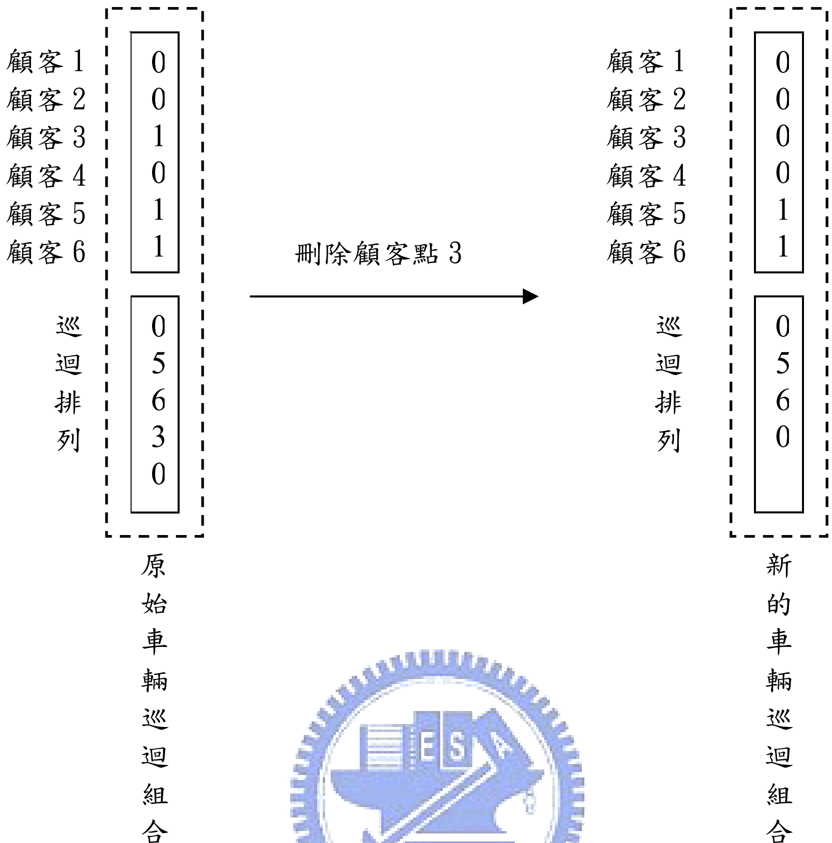


圖 3-6 車輛路線巡迴組合刪除顧客點圖

(二)在車輛路線巡迴組合增加顧客進去

檢視當次遞迴裡顧客的 $s_i(\boldsymbol{u})$ 值是否有 1 的情況發生(代表有顧客未被涵蓋)，值得注意的是有此情況發生，這個加顧客進去車輛路線巡迴組合之產生新車輛路線巡迴組合機制才會執行。滿足此前提後，再檢視 $s_i(\boldsymbol{u})$ 值為 1 的顧客為何，如果同時有兩個以上 $s_i(\boldsymbol{u})$ 值為一的顧客，則隨機挑選一個，將此顧客視為「要加入的顧客」，把此顧客加入「所有原本保留的 m 個車輛路線巡迴」裡，產生了新的車輛路線巡迴組合，這些新產生的車輛路線巡迴組合至多 m 個，因為新產生的車輛路線巡迴組合有以下限制：

1. 加入該顧客於車輛路線巡迴組合後，檢視車輛路線巡迴組合是否超出車輛容量限制，超出容量即為不可行解，不能算是新產生的車輛路線巡迴組合。

2. 產生的新車輛路線巡迴組合需檢查是否與「目前保留下之車輛路線巡迴組合」重複，如果重複，則不算是新產生的車輛路線巡迴組合。

滿足以上限制條件後所留下之車輛路線巡迴組合，本研究簡稱為「加點的集合」，再將這些加點的集合一同合併進去保留的車輛路線巡迴組合，故倘若只有加顧客點的機制有被執行的話，此時留下的車輛路線巡迴組合數為至多 2m 個，倘若刪除顧客點及加顧客點的機制皆被執行，此時留下的車輛路線巡迴組合數已經增長到至多 3m 個。

在車輛路線巡迴組合裡增加顧客時，巡迴組合變動方式是試著插入顧客於該巡迴組合裡的顧客點與顧客點之間，尋找出最小成本的巡迴組合，依此為新的車輛路線巡迴組合，例如車輛路線巡迴組合為[0、5、6、0]，其中 0 代表場站原點，由原點 0 出發，依序經過了顧客點 5、6，最後回到原點 0，而如果要增加的顧客為顧客點 3，則插入此顧客後將會有三種巡迴組合：[0、3、5、6、0]、[0、5、3、6、0]和[0、5、6、3、0]，計算此三種巡迴組合的成本，假設[0、5、6、3、0]為其中成本最小巡迴組合，則挑選此巡迴組合為[0、5、6、0]插入顧客點 3 後的新車輛巡迴組合，並不再另外求解 TSP 最佳走法，詳見圖 3-7。

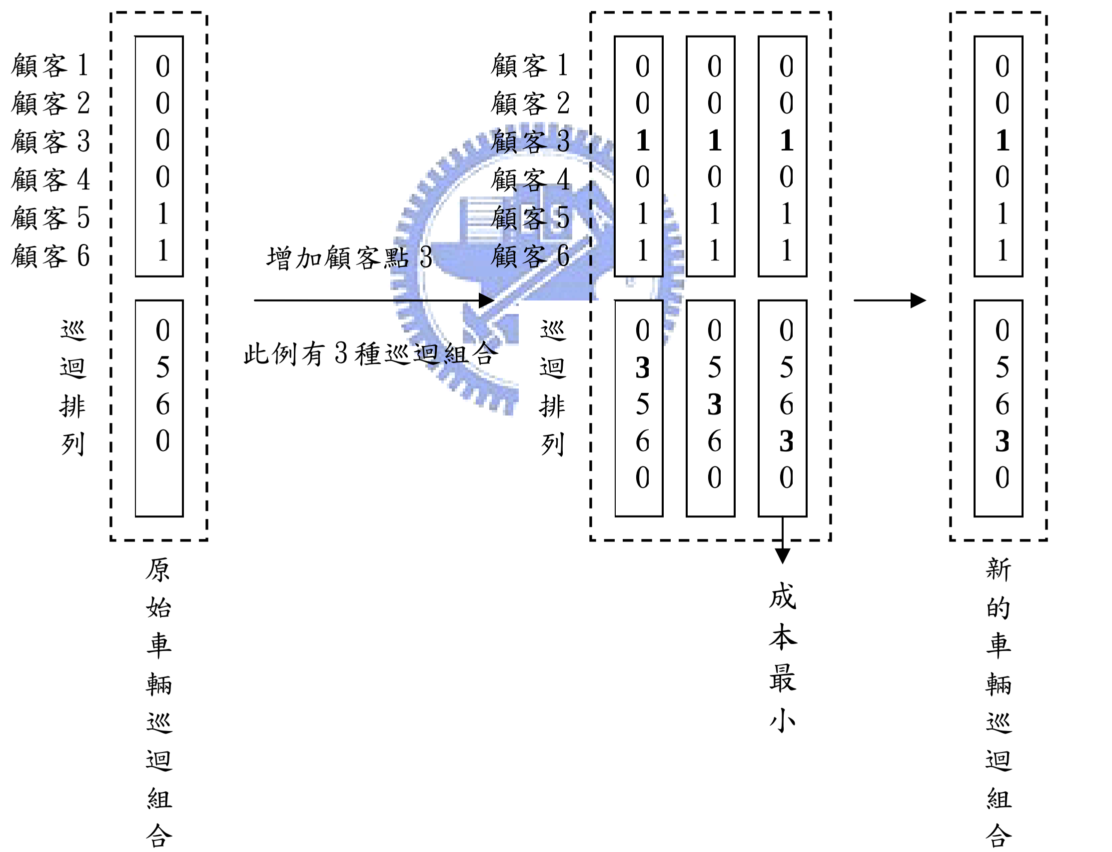


圖 3-7 車輛路線巡迴組合增加顧客圖

3.3.4 反覆遞迴裡所紀錄的最佳解

保留的車輛路線巡迴組合空間最後還會加上:「反覆遞迴裡所紀錄的最佳解

之車輛路線巡迴組合」，此舉是由於本研究預期從這些組合裡，可以調整產生更好的車輛路線巡迴解，也使得下次遞迴修正需要可行解時，車輛路線巡迴組合空間裡一定含有可行解供挑選。

這些全部整理保留下來的車輛路線巡迴組合，將提供作為下次遞迴運算之起始車輛路線巡迴解。

3.4 上限值、下限值的意義與停止機制

原本拉式鬆弛法求解集合涵蓋問題的方法，是產生全部可行之車輛路線巡迴組合解的集合空間，當上限值 UB 及下限值 $L(\boldsymbol{u})$ 兩解相等或兩解之差值在容忍範圍內時，即停止求解。但本研究以部分車輛路線巡迴組合空間，並在遞迴運算的過程中加以調整的方式求解，就 *UB* 值與 $L(\boldsymbol{u})$ 值的意義上，已經與原本拉式鬆弛法求解集合涵蓋問題略有不同，造成求解之停止機制，也隨之改變。

以下將先說明本研究演算法中，求解拉式問題、集合涵蓋問題與原本拉式鬆弛法求解集合涵蓋問題不同意義之處，而後再敘述如何決定停止機制。

a. 本研究演算法求解拉式鬆弛問題所得到的解，在遞迴運算中針對該次之車輛路線巡迴組合空間，確實仍是一個有效的下限值，但與原來車輛定線問題的下限已有所差異。主要是因為本研究僅產生部份部分的車輛路線巡迴組合空間，所得之下限僅是針對目前車輛路線巡迴組合空間求解得到的一個下限值，不確保車輛定線問題的最佳解必定高於此 $L(\boldsymbol{u})$ 值。(當然，若原來的拉式鬆弛問題中，是利用所有可能車輛路線巡迴組合空間來求得下限值，則可保證最佳解值必定高於 $L(\boldsymbol{u})$ 值。)
b. 根據拉式問題的解如 3.2 之(一)部分，可求得集合涵蓋問題的可行解及上限值。此上限值是以部份車輛路線巡迴組合空間所求得，因此可視為一個區域最佳解。由於最佳解必小於或等於區域最佳解，所以此上限值(區域最佳解)，雖然意義上僅是根據部分車輛路線巡迴組合空間所求得，但對於原來車輛定線問題來說仍是一個有效的上限值。

由此可知，雖然採用與原先拉式鬆弛法同樣方式，但本研究之演算法無法利用上下限值夾擠得到確切的最佳解所在區間，僅能根據上下限值持續改進有限的車輛路線巡迴組合空間，再反饋至上下限值，使得集合分割問題的解能夠持續獲得改進。所以在最後停止機制部分，因為 *UB* 值與 $L(\boldsymbol{u})$ 值的區間，僅做為提供遞迴下次求解時的參數之用，使得真正的求解機制僅能利用「集合分割問題的解趨於穩定」，或是「當求解次數達到所設定次數」，來做為停止條件。圖 3-8 為 21 個顧客點之集合分割解圖，可見集合分割問題的解趨於穩定，且於程式設定求解一千次未改善解後停止求解。

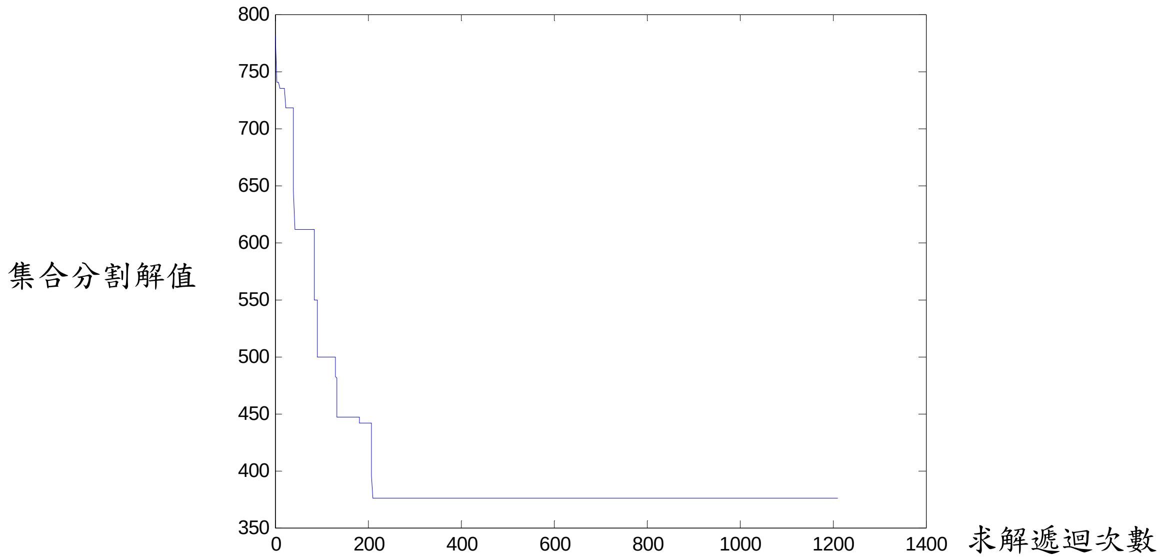


圖 3-8 21 個顧客點之集合分割解遞迴圖

3.5 演算法流程及細部設定說明

本研究演算法之主要流程敘述詳見圖 3-9。

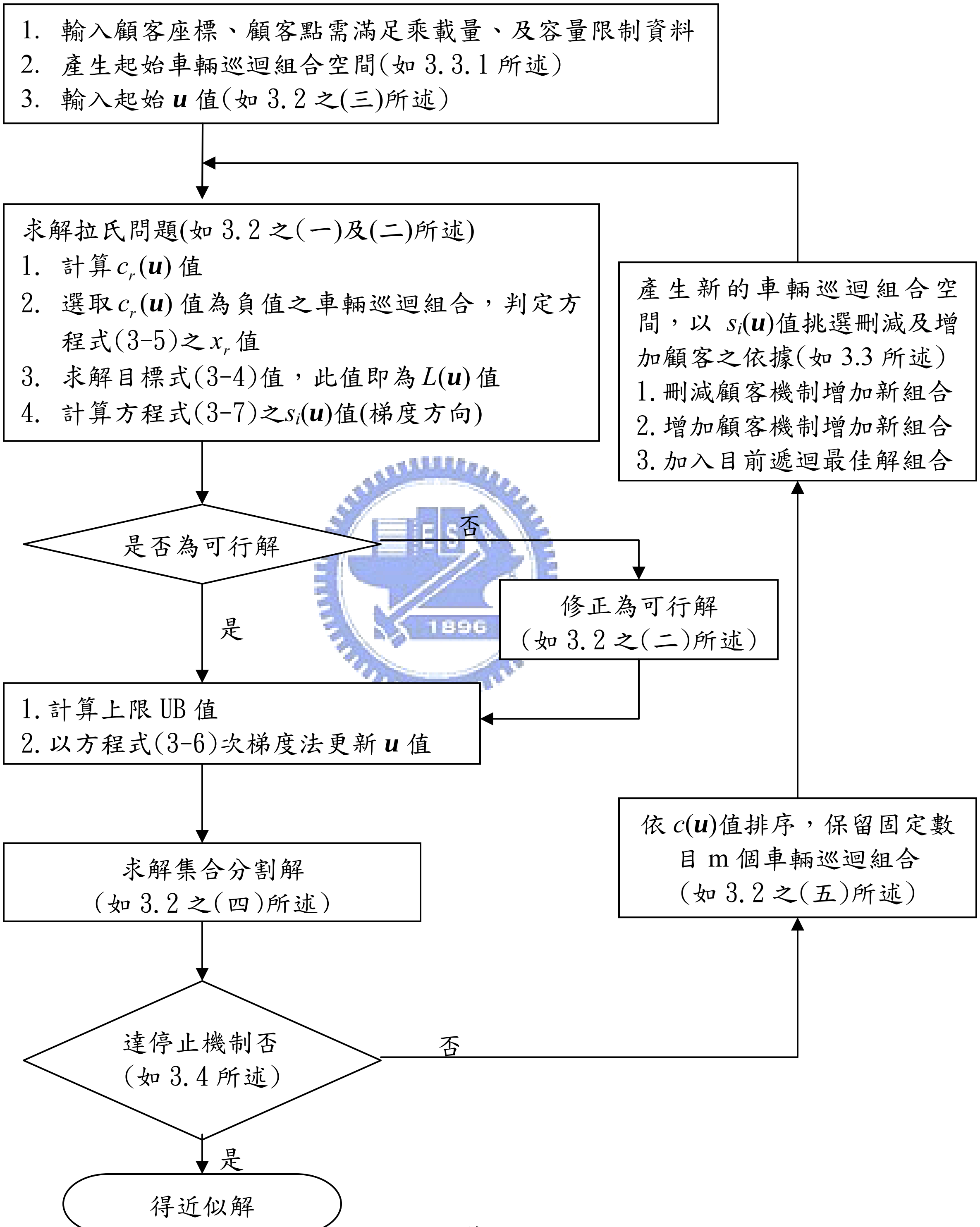


圖 3-9 演算法詳細流程圖

將演算法之主要流程敘述過後，其後再針對演算法的一些細部設定做說明。

1. 有關保留車輛路線巡迴組合數的數值(m)

在前述內容中是以固定數值的方式，但也可採用變動數值方式，隨問題規模增大自由增減。選擇變動數值方式，有利於產生足夠的車輛路線巡迴組合空間，但會將降低求解效能；而固定數值的方式，則是選定適合之限制個數，不因問題規模而改變，但有可能會加長求解過程。但因為多數的車輛路線巡迴組合對於原問題並沒有貢獻，所以即使問題規模加大，也不需要產生太多的集合，因此採用固定數值的方式，會比變動數值方式來的好。

2. 有關調整係數λ值方面

於 ***u*** 值更新方程式中的調整係數λ值，在 Carprara *et al*. (1999)中，另外提及以上下限值的差距對λ值做小幅度調整的方式，由於在本研究中之上下限值區間，不一定為最佳解存在之區間，因此在實例測試方面，對於調整係數λ值與方向長度影響等議題將不再深入，而採參考過往文獻的經驗法則來設定。

總合第三章內容所述，本研究之演算法採用拉式放鬆法後，往最佳解修改之方向是使用一般貪心法則、次梯度法，以此啟發式解法挑選及產生集合反覆求解，依然以部分集合為起始解求解問題，不產生所有可能的集合，遞迴過程中也同樣是以部分解求解，每個求解循環之中，核心機制在著重於如何產生好的有限車輛路線巡迴組合空間，進一步利用此有限車輛路線巡迴組合空間，來求得近似解。因此在產生新的車輛路線巡迴組合空間方面，設定了兩項機制可增加新車輛路線巡迴組合，為調整車輛路線巡迴組合空間的動作，可避免車輛路線巡迴組合空間變化不足，而使得好的解無法產生的問題。

本研究與紀玟豪(2004)集合涵蓋模式由於求解不同類型問題，所以在成本結構與限制皆有所不同，例如紀玟豪之研究中限定每個貨物集合只能上一個航班，此限制在本研究並不存在，另外「修改為可行解」與「調整新增集合的方式」則是最主要之差異。

第四章 演算法測試與結果分析

4.1 測試問題

本研究由 Solomon 學者之網站(註一)蒐集測試例題，其中有容量限制之車輛路線問題之例題眾多，本研究挑選 Christofides 和 Eilon 學者之車輛路線例題題庫加此測試演算法，經初步數值測試後，可詳細觀察本研究演算法之績效，表 4-1 為例題整理表。

註一：http://neo.lcc.uma.es/radi-aeb/WebVRP/index.html?/results/resultsSolom.htm

表 4-1 Christofides 及 Eilon 學者之 VRP 例題整理表

例題	顧客數	容量限制	目前最佳解
E-n13-k4	12	6000	247
E-n22-k4	21	6000	375
E-n23-k3	22	4500	569
E-n30-k3	29	4500	534
E-n33-k4	32	8000	835
E-n51-k5	50	160	521

4.2 處理同一問題之車輛路線規劃比較表現

本研究演算法之車輛路線規劃結果將與最佳解比較外，由於本研究之啟發式解法屬於 Fisher 學者所分類的第二類數學規劃基礎啟發式解法，所以亦與和本研究所測試例題相同之四篇其他分類之論文比較，此四篇論文如下：

1. Laporte 及 Semet (1998)改善 Clarke 及 Wright(1964)之經典節省法後的 Parallel 節省法，此法屬於 Fisher 學者所分類的第一類簡單啟發式解法。

2. Mole 及 Jameson(1976)的插入法，此法屬於 Fisher 學者所分類的第一類簡單啟發式解法。

3. Osman(1993a)的模擬退火法，此法屬於 Fisher 學者所分類的第三類-人工智慧啟發式解法。

4. Von Zuben Gomes(2002)的類神經模糊法加以比較，此法屬於 Fisher 學者所分類的第三類-人工智慧啟發式解法。

本研究與其他啟發式解法處理同一車輛路線規劃問題之比較表現如表 4-2 所示。

表 4-2 處理同一車輛路線規劃問題之比較表現

例題	顧客數	Laporte 及 Semet	Mole 及 Jameson	Osman	Gomes 及 Von Zuben	本研究	目前最佳解	Gap%
		Parallel 節省法	插入法	模擬退火法	類神經模糊法	數學規劃法		
E-n13-k4	12	-	-	-	-	247	247	0
E-n22-k4	21	-	-	-	383.51	375	375	0
E-n23-k3	22	-	-	-	-	569.75	569	0.1
E-n30-k3	29	-	-	-	534[a]	**507.1**	534	-
E-n33-k4	32	-	-	-	-	861.87	835	3
E-n51-k5	50	585	575	528	525.74[a]	533.18	521	2

[a] 由於其啟發式解法之特性，此解超出車輛容量限制 0.5%~2.8%。

4.3 綜合分析

本研究求解到 50 個顧客數的車輛路線規劃績效良好，且求解品質穩定。

測試 12 個及 21 個顧客點，本研究車輛路線規劃最佳解同於目前最佳解，圖 4-1 為 21 個顧客-目前與本研究最佳解車輛路線規劃圖。

測試 22 個顧客點，本研究車輛規劃最佳解與目前最佳解僅僅相差 0.1%，幾乎相等，圖 4-2 為 22 個顧客-目前最佳解車輛路線規劃圖，圖 4-3 為 22 個顧客-本研究最佳解車輛路線規劃圖。

測試 29 個顧客點之例題 E-n30-k3，在本研究啟發式解法下，目前最佳解為 534，使用車輛數為 3 台，而本研究求解之最佳解為 507.1，使用車輛數為 4 台，因為本研究並未限制車輛數，純粹以最小總旅行距離為目標，圖 4-4 為 29 個顧客-目前最佳解車輛路線規劃圖，圖 4-5 為 29 個顧客-本研究最佳解車輛路線規劃圖。

測試 32 個顧客點，本研究車輛規劃最佳解與目前最佳解僅僅相差 3%，圖 4-6 為 32 個顧客-目前最佳解車輛路線規劃圖，圖 4-7 為 32 個顧客-本研究最佳解車輛路線規劃圖。

測試 50 個顧客點，本研究車輛規劃最佳解與目前最佳解僅僅相差 2%，圖 4-8 為 50 個顧客-目前最佳解車輛路線規劃圖，圖 4-9 為 50 個顧客-本研究最佳解車輛路線規劃圖。

本研究之啟發式解法最佳解與目前最佳解之誤差，推斷原因如下：

1. 要增加的顧客和要刪減的顧客在遞迴裡一直重複選到同一個顧客，導致無法產生其他更好的車輛路線巡迴路線解。

2. 保留 m 個車輛路線巡迴組合數，m 值的設定不妥當，導致好的車輛路線巡迴組合被刪去。

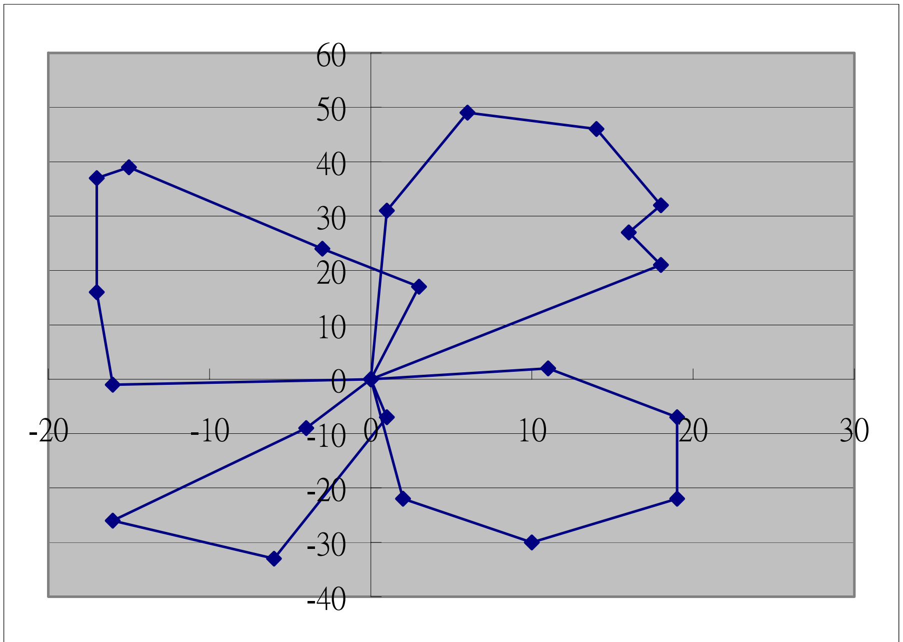


圖 4-1 21 個顧客-目前與本研究最佳解車輛路線規劃圖

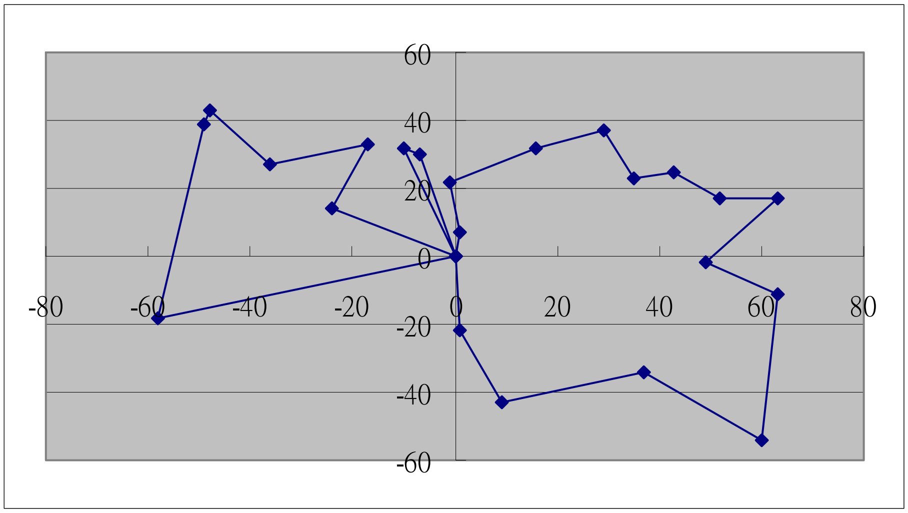


圖 4-2 22 個顧客-目前最佳解車輛路線規劃圖

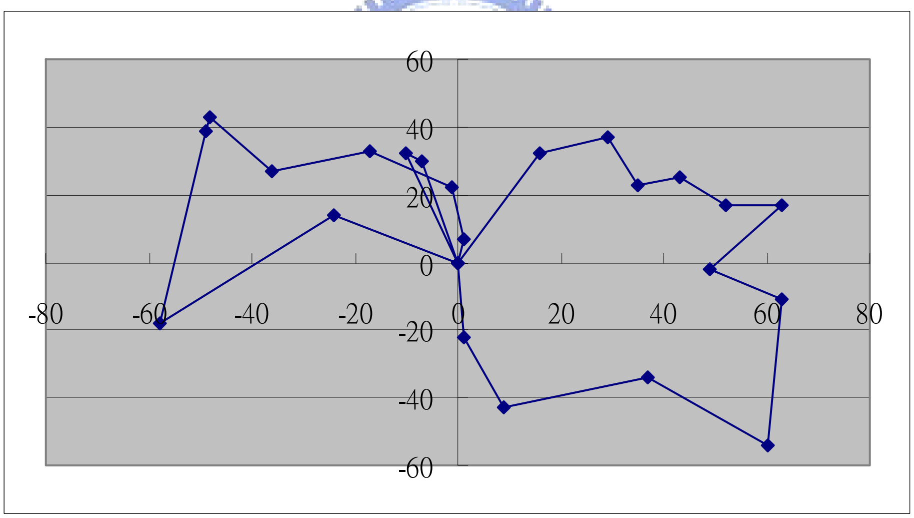


圖 4-3 22 個顧客-本研究最佳解車輛路線規劃圖

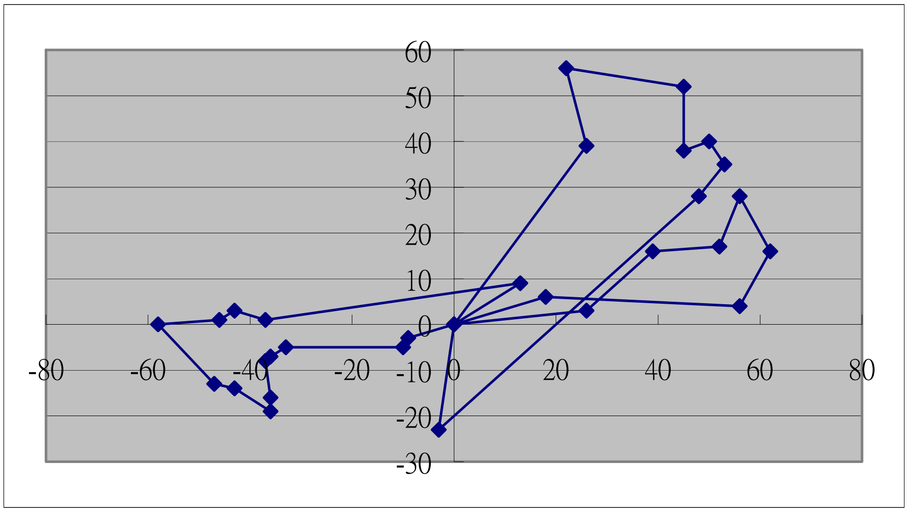


圖 4-4 29 個顧客-目前最佳解車輛路線規劃圖

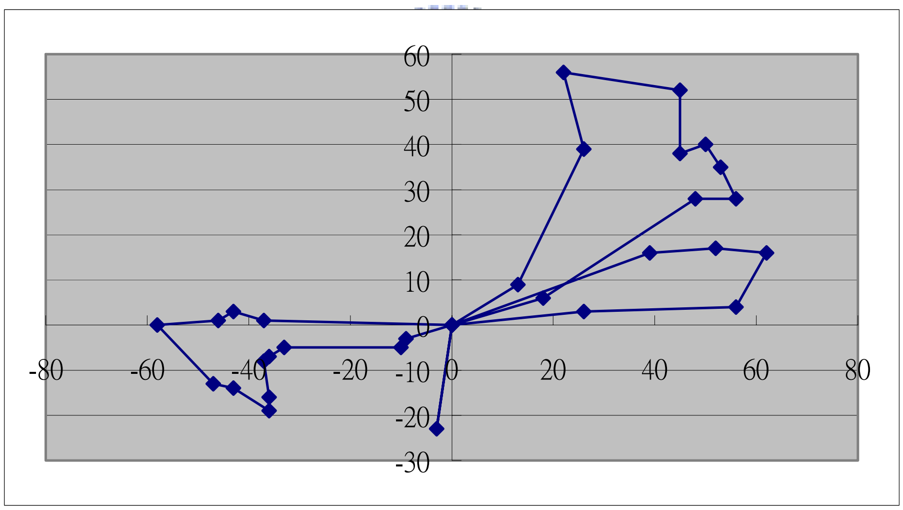


圖 4-5 29 個顧客-本研究最佳解車輛路線規劃圖

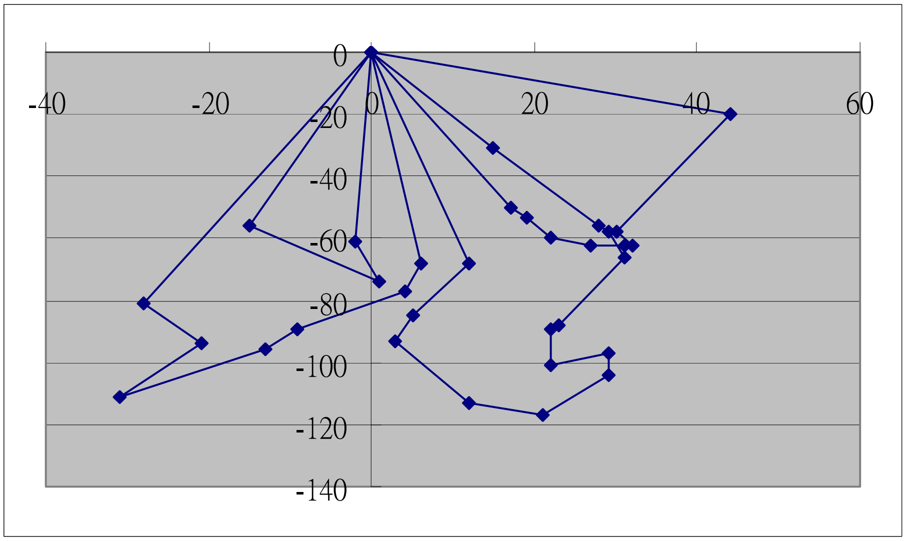


圖 4-6 32 個顧客-目前最佳解車輛路線規劃圖

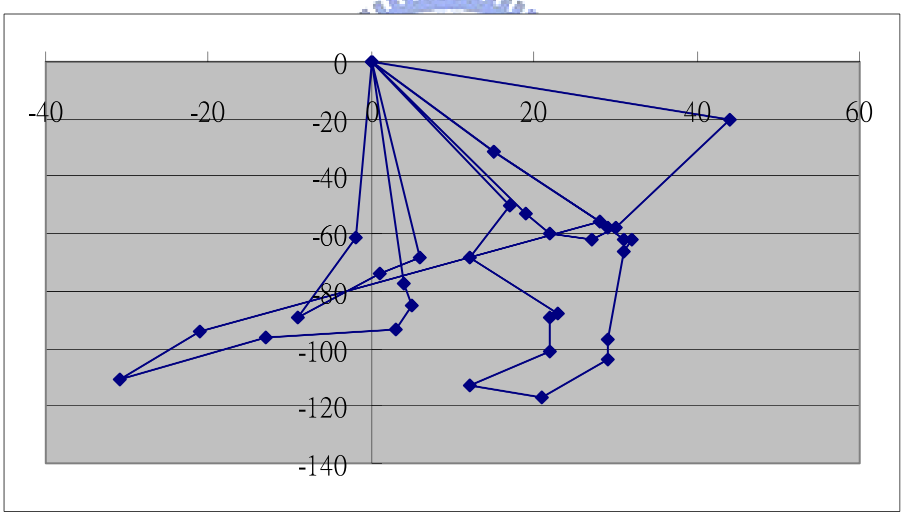


圖 4-7 32 個顧客-本研究最佳解車輛路線規劃圖

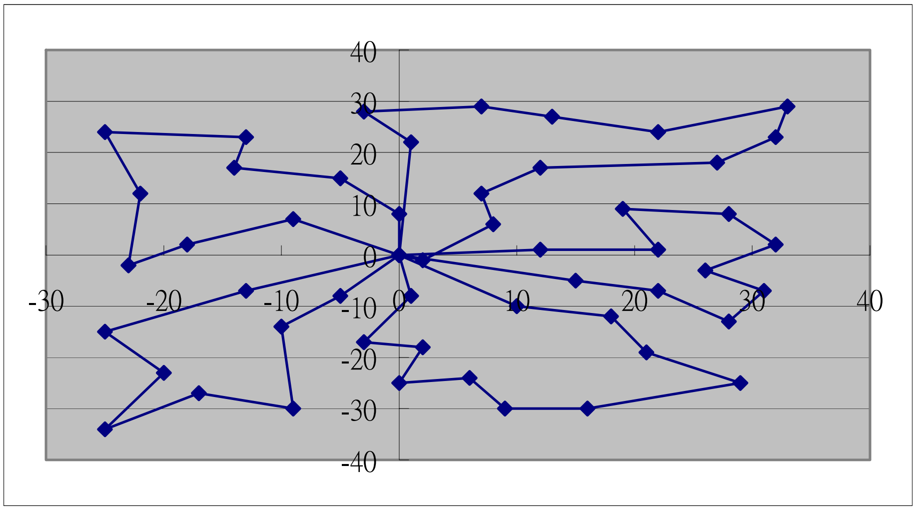


圖 4-8 50 個顧客-目前最佳解車輛路線規劃圖

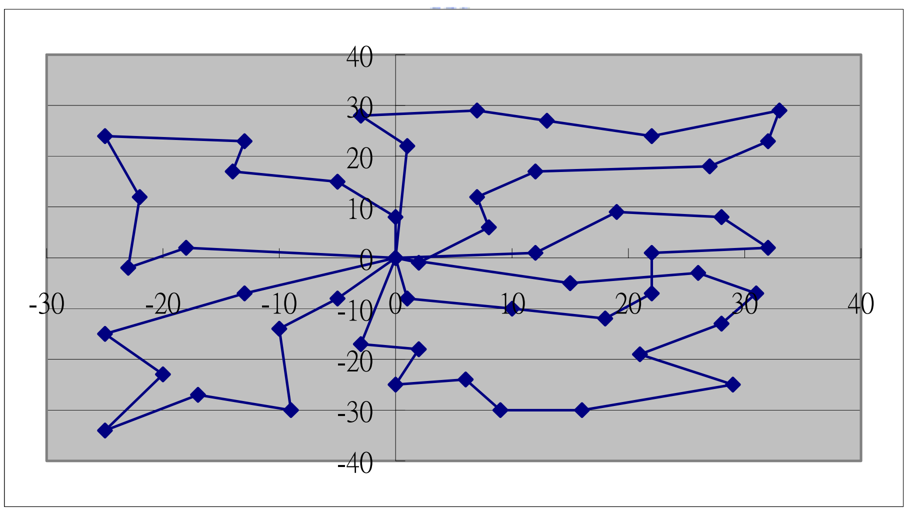


圖 4-9 50 個顧客-本研究最佳解車輛路線規劃圖

第五章 結論與建議

5.1 結論

混合整數規劃之車輛定線問題模式，在處理較大規模的問題時，其求解效能並不理想，如果逕行以其他方式求解，則其求解效果未知。而藉由文獻了解，大多數路由以及排班問題，可藉由集合涵蓋問題來求解，並且集合涵蓋問題已有相當成熟且效能不錯之求解方法，如拉式鬆弛法即是其中之一。由於每個顧客僅需要一台車輛加以服務，所以原車輛定線問題在本質上應屬於集合分割問題，但是過往文獻皆顯示集合涵蓋問題較集合分割問題容易求解，而且將集合涵蓋問題之解修正為集合分割問題之解時，其步驟相當容易。

因此本研究決定將車輛定線問題模式，以集合涵蓋問題描述，而後以常用之拉式鬆弛法為基礎，結合貪心法則、次梯度法之啟發式解法，並結合列運算技巧之觀念：只產生「部分車輛路線巡迴解」進行第一階段求解，因集合涵蓋問題的求解，必須利用一有效的集合空間，對車輛定線問題而言，可能的集合(車輛路線巡迴組合)空間會由於問題規模增加而過大，而使得求解相當困難。因此本研究透過僅以部分集合空間，來達到求解的目的，並於過程中，利用 $c_r(\boldsymbol{u})$ 值、$s_i(\boldsymbol{u})$值對集合空間進行調整，使得演算法解能逐步逼近最佳解，故本研究最主要之貢獻在於新建一簡單之啟發式解法，顧客分群後並未求解TSP問題，以刪減車輛路線巡迴組合調整保持部分車輛路線巡迴組合，在每次遞迴中加以求解。

第二階段再根據拉式解求得集合涵蓋問題之解，最後利用集合涵蓋問題之解修正為集合分割解，倘若未達停止求解機制，產生新的車輛路線巡迴組合併入下次遞迴求解。

而停止機制方面，因為依上述方式求解出之 UB 與 $L(\boldsymbol{u})$ 值，僅能做為求解迴圈下次運算之參數，造成停止機制必須重新設定，本研究即以集合分割問題之解趨於穩定，或設定求解次數做為停止機制。

本研究之演算法經數值範例測試求解後，演算法在 50 個顧客數以下具穩定之求解品質，且與最佳解皆相當接近。

5.2 未來研究方向

本研究之演算法目前僅著重於解的品質，對於求解時間並未有很大的著墨，僅設定在合理時間內求解，主要原因為研究目標之設定在於以演算法架構為主，參考一已知的求解方法，根據問題特質修正解法內容來求解。在各個求解步驟裡，希望皆採取較簡單且具有一定效果的方式，增加求解效率，並由問題特性著手，觀察求解的過程，逐步測試多項不同的解題機制，而後定出演算法之最後架構。

本研究在相關參數部分，僅參考文獻之經驗法則，以可以穩定求解之目標為主，未做深入探討，若能繼續深入搭配更細緻的解題機制，相信會有更佳的求解品質。

本研究演算法尚有許多可持續進行之議題，因此未來研究方向，將有以下幾點：

1. 部分解集空間的調整機制。如將一次調整加入或是刪減一個顧客，可修改成調整多個；或是遞迴中設定不同 m 值的機制，來保留適合的車輛路線巡迴組合作為解集合空間。

2. 可以更細緻的方式來修正,如參照挑選集合加入後可增加涵蓋多少原本不涵蓋的顧客。

3. 參數部分的設定。可再多加嘗試不同的數值，來加快求解，在穩定求解品質與速度上做權衡。

參考文獻

物流業，作者：鐘榮欽、鄭永侃、楊緝熙、何薇立、蘇雄義，民國 84 年 5 月出版。

澳門生產力暨技術轉移中心發表的生產力論壇。

研究分析方法，馮正民、邱裕鈞，建都文化事業股份有限公司，民國 93 年 6 月。

張育彰，應用基因演算法於台鐵列車駕駛員排班與輪班整合問題之研究，國立成功大學交通管理科學研究所碩士論文，民國 92 年 6 月。

Agarwal, Y., Mathur, K., and Salkin H.M. (1989) A set-partition-based exact algorithm for the vehicle routing problem. **Networks, 19**, 731-749.

Altinkemer, K., and B. Gavish. (1991) Parallel savings based heuristics for the delivery problem. **Oper. Res., 39**, 456-469.

Aourid, M. and Kaminska, B. (1994) Neural Networks for the Set Covering Problem: An Application to the Test Vector Compaction. **IEEE International Conference on Neural Networks Conference Proceedings, 7**, 4645-4649.

Baker, B. M., and Sheasby, J. (1999) Extensions to the generalized assignment heuristic for vehicle routing. **European Journal of Operational Research, 119**, 147-157.

Balas E. and Ho. (1980) Set Covering Algorithms Using Cutting Planes, Heuristics, and Subgradient Optimization: A Computational Study. **Mathematical Programming, 12**, 37-60.

Balas E. and Carrera M.C. (1996) A Dynamic Subgradient-based Branch-and-Bound Procedure for Set Covering. **Oper. Res., 44**, 875-890.

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

Barrie M. Baker, M.A. Ayechew (2003) A genetic algorithm for the vehicle routing problem. **Computers & Operations Research, 30,** 787-800.

Beasley, J.E. (1990) A Lagrangian Heuristic for Set-Covering Problems. **Naval Research Logistics, 37**, 151-164.

Billy E. Gillett & Leland R. Miller (1974) A Heuristic Algorithm for the Vehicle-Dispatch Problem. **Oper. Res. 22**, 340-349.

Bixby A., Coullard C. and Simchi-Levi D. (1997) The capacitated prize-collecting traveling salesman problem. Working paper, Department of Industrial Engineering and Engineer Management, Northwestern University, Evanston, IL.

Bixby A. (1998) Polyhedral analysis and effective algorithm for the capacitated vehicle routing problem. Ph. D. dissertation, Northwestern University, Evanston, IL.

Bodin Lawrence and Bruce Golden（1981）Classification in vehicle routing and scheduling. **Network, 11**, 1981.

Bramel, J., Simchi-levi, D. (1996) Probabilistic analysis and practical algorithms for the vehicle routing problem with time windows. **Oper. Res., 44**, 501-509.

Bramel, J., Simchi-levi, D. (1997) On the effectiveness of set covering formulations for the vehicle routing problem with time windows. **Oper. Res., 45**, 295-301.

Caprara, A., Fischetti M., and Toth, P., (1999) A heuristic method for the set covering problems. **Oper. Res., 47**, 730-743.

Chiang, W., Russell, R. (1996) Simulated annealing metaheuristics for the vehicle routing problem with time windows. **Annals of Operations Research, 63**, 3-27.

Christofides, N., and Eilon, S., (1969) An algorithm for the vehicle dispatching problem. **Oper. Res. Q. 20**, 309-318.

Christofides, N., (1985) Vehicle routing, In: Lawler, E. L., J. K. Lenstra, A. H. G. Rinnooy Kan, and Shmoys, D. B. (eds). The traveling Salesman Problem, John Wiley and Sons Ltd., **New York**, 431-448.

Christofides, N., Mingozi, A., and Toth, P. (1979) The vehicle routing problem. **Combinational Optimization**, John Wiley & Sons Ltd., New York, 318-338.

Clarke, G., and Wright, J., (1964) Scheduling of vehicles from a central depot to a number of delivery points, **Oper. Res., 12**, 568-581.

Cullen, F., J. Jarvis and D. Ratliff, (1981) Set partitioning based heuristics for interactive routing. **Networks, 11**, 125-144.

Desrochers, M., Desrosiers, J. Soumis, M. (1984) Routing with time windows by column generation. **Networks, 14**, 545-565.

Desrochers, M., Desrosiers, J. Solomon, M. (1992) A new optimization algorithm for the vehicle routing problem with time windows. **Oper. Res., 40**, 342-354.

Gendreau, M., Hertz, A., and Laporte, G.. (1992) A tabu search heuristic for the vehicle routing problem. Working Paper, ORWP 92/14. Department de Mathematiques. Ecole Polytechnique Fe'de'rale de Lausanne.

Glover, F., (1989) Tabu search-Part Ⅰ, **ORSA Journal on Computing, 1(3)**, 190-206.

Feo, A. and Mauricio, G.C. and Resende, A. (1989) A Probabilistic Heuristic for a Computationally Difficult Set Covering Problem. **Oper. Res. Lett., 8**, 67-71.

Fisher, M., Vehicle routing, in: Ball, M., Magnanti, T., Monma, M., Nemhauser (Eds.), G. (1995) **Handbooks in Operations Research and Management Science, Vol. 8**: Network Routing, North-Holland, Amsterdam, 1-33.

Fisher, M. L., and Jaikumar, R., (1981) A generalized assignment heuristics for vehicle routing, **Networks, 11**, 109-124.

Fisher, M. L., and Wolsey L., (1982) On the Greedy Heuristic for Covering and Packing Problems, **SIAM Journal on Algebraic and Discrete Methods, 3**, 584-591.

Fisher, M. L., and Rosenwein, M. B., (1989) An interactive optimization system for bulk cargo ship scheduling. **Naval Research Logistics, 36**, 27-42.

Fisher, M., Jornsten, K.O., Madsen, O.B.G. (1997) Vehicle routing with time windows: two optimization algorithms. **Oper. Res., 4**5, 488-492.

Fleuren, H. (1988) A computational study of the set partitioning approach for vehicle routing and scheduling problems. Ph. D. Dissertation, University of Twente, Enshede.

Foster, B.A. and Ryan, D.M. (1976) An integer programming approach to the vehicle scheduling problem. **Oper. Res., 27**, 367-384.

Gaskell, T.J.(1967) Basis for the vehicle fleet scheduling. **Oper. Res., Q. 18**, 281.

Gendreau, M., Hertz, A., and Laporte, G., (1994) A tabu search heuristic for the vehicle routing problem. **Management Science, 40(10)**, 1276-1290.

Gillett, B. E., and Miller, L. R., (1974) A heuristic algorithm for the vehicle dispatch problem. **Oper. Res., 22**, 340-349.

Golden, B., L. Bodin, AL. Assad and M. Ball. (1983) Routing and scheduling of vehicle and crew: the state of art. **Special Issue of Computers and Operations Research, Vol. 10**, No.2, 63-211.

Golden, B., (1975) Vehicle routing problem: Formulations and heuristic solution techniques. ORC Tech. M.I.T., Rep. 113.

Golden, B., (1977) T.L. Magnanti and H.Q. Nguyen 1977. Implementing vehicle routing algorithms. **Networks, 7**, 113-148.

Gomes, L.C.T., Von Zuben, F.J. (2002) A Neuro-Fuzzy Approach to the Capacitated Vehicle Routing Problem. **Proceedings of the IEEE International Joint Conference on Neural Networks (IJCNN'2002), vol. 2,** 1930-1935, in the 2002 IEEE World Congress on Computational Intelligence (WCCI'2002), Honolulu, Hawaii, May 12-17.

Hoffman K.L. and Padberg M. (1993) Solving airline crew scheduling problems by branch and cut. **Management Science, 39**, 657-682.

John H. Holland. (1975) Adaptation in Naturaland Artificial Systems. University of Michigan Press, AnnArbor.

Kohl, N., Madsen, O.B.G. (1997) An optimization algorithm for the vehicle routing problem with time windows based on lagrangian relaxation. **Oper. Res., 45**, 395-406.

Kolen, A.W.J., Rinnooy Kan, A.H.G., Trienkens, H.W.J.M. (1987) Vehicle routing with time windows. **Oper. Res., 35**, 266-273.

Koskosidis, A., Powell, W.B., Solomon, M.M. (1992) An optimization-based heuristic for vehicle routing and scheduling with soft time window constrains. **Transportation Science, 26**, 69-85.

Kontoravdis, G., and Bard, J., (1992) Improved heuristics for the vehicle routing problem with time windows. Working Paper, Operations Research Group, Department of Mechanical Engineering, University of Texas, Austin.

Koskosidis, Y. A., Powell W.B. and Solomom M.M. (1989) An optimization based heuristic for vehicle routing and scheduling with time window constraints. No. TMS-88-09-1, The Institute for Transportation Systems, The City Colledge of CUNY, New York, NY.

Laporte G. and Semet F. (1998) Classical Heuristics for the Vehicle Routing Problem. Les Cahiers du GERAD, G98-54, Group for Research in Decision Analysis, Montreal, Canada. Lenstra, J., and Rinnooy K., (1981) Complexity of vehicle routing and scheduling problem. **Network, 11(2)**, 221-227.

Lin, S., (1965) Computer solutions of the traveling salesman problem. **Bell System Tech. J, 44**, 2245-2269.

Lin, S., and Kernighan, B., (1973) An effective heuristic algorithm for the traveling salesman problem. **Oper. Res., 21**, 498-516.

Mole R. H., Jameson S. R., (1976) A sequential route-building algorithm employing a generalised savings criterion. **Oper. Res., Q. 27**, 503.

Nygard, K.E., Greenberg, P., Bolkan, W.E. and Swenson, E.J., (1988) Generalized assignment methods for the deadline vehicle routing problem, in B.L. Golden and A.A. Assad (eds.), **Vehicle routing: Methods and Stuides**, Elsevier Science Publishers B.V., North-Holland.

Osman, I.H. (1991) Metastrategy simulated annealing and tabu search algorithms for combinatorial optimization problem. Ph.D. Dissertation, The Management School, Imperial Collage, London.

Osman, I.H. (1993a) Metastrategy simulated annealing and tabu search algorithms for the vehicle routing problem. **Annals of Operational Research, 41**, 421-451.

Potvin, J., Rousseau, J. (1995) An exchange heuristic for routing problems with time windows. **Journal of the Operational Research Society, 46**, 1433-1446.

Pureza, V.M., and Franca, P.M. (1991) Vehicle routing problem via tabu search metaheuristic. Publication CRT-747, Centre de recherche sur les transports, Montreal.

Renaud J, Boctor FF, Laporte G. (1996) A fast composite heuristic for the symmetric traveling salesman problem. **INFORMS Journal on Computing; 8**,134-43.

Ribeiro C. and Soumis F. (1994) A column generation approach to the multi-depot vehicle scheduling problems. **Oper. Res., 42**, 41-52.

Russel, R.A. (1977) An effective heuristic for the M-tour traveling salesman problem with some side conditions. **Oper. Res., 25**, 517-524.

Russel, R.A. (1995) Hybrid heuristics for the vehicle routing problem with time windows. **Transportation Science, 29**, 156-166.

Savelsbergh, M. (1985) Local search in routing problems with time windows. **Oper. Res., 4**, 285-305.

Solomon, M., (1987) Algorithm for the vehicle routing and scheduling problem with time window constrain. **Oper. Res., 35**, 254-265.

Solomon, M., (1986) On the worst-case performance of some heuristics for the vehicle routing and scheduling problems with time windows constraints. **Networks, 16**, 161-174.

Stefan Irnich, (2000) A multi-depot pickup and delivery problem with a single hub and heterogeneous vehicles, **European Journal of Operational Research, 122**, 310-328.

Taillard, E. (1993) Parallel iterative search methods for vehicle routing problems. **Networks, 23**, 661-673.

Taillard, E., Badeau, P., Gendreau, M.M., Guertin, F., Potvin, J. (1997) A tabu search heuristic for the vehicle routing problem with soft time windows. **Transportation Science, 31**, 170-186.

Thompson, P.M. (1988) Local search algorithms for vehicle routing and other combinatorial problems. Submitted in partial fulfillment of the requirements of the degree of Doctor of Philosophy in Operations Research at the Masachusetts Institute of Technology.

Van Breedam, A. (2001) Comparing descent heuristics and metaheuristics for the vehicle routing problem. **Computers & Operations Research, 28**, 289-315.

Willard, J. A. G. (1989) Vehicle Routing Using r-Optimal Tabu Search. M.Sc. Dissertation. The Management School, Imperial College, London.

Yello, P. (1970) A computation modification to the savings method of vehicle scheduling. **Oper. Res., Q. 21**, 281-283.

文獻注釋

紀玟豪, 以拉格蘭式鬆弛法求解航空貨運承攬業之並裝決策問題, 國立交通大學，碩士論文，民國 93 年

紀玟豪（2004）應用集合涵蓋模式及拉式放鬆法求解航空貨運承攬貨物之併裝問題，基於航空貨運計費機制的複雜性，航空貨運承攬業的貨物併裝問題是一個相當困難的凹性極小化（concave minimization）問題。此研究將該問題轉換成集合涵蓋問題，再以拉式鬆弛法（Lagrange relaxation）為基礎，透過併裝組合空間的調整，發展一遞迴性的啟發式演算法。數值測試的結果顯示該演算法求解的品質相當理想，在所有的測試問題中，演算法求解之目標式值與最佳值均相當接近。

航空貨運承攬業經營的成效對整個航空貨運業有著關鍵的影響，其中航空貨運的計價方式包含有兩個特性，一是同時考慮重量與體積，一是有數量折扣，不同重量級距之費率不相同，且各航班之費率曲線亦不同。航空貨運承攬業必須要能非常有技巧地對所承攬的貨物進行併裝，並向航空公司訂取適當航班的艙位，才能在滿足貨主要求下，降低其支付給航空公司的運價。此研究之目的即是希望利用數學模式，發展具備求解較大規模問題之演算法，做為航空貨運承攬業併裝作業決策輔助系統的核心模組。

由於本研究之解題構想與紀玟豪(2004)相似，故摘錄其演算法發展步驟如下：

(一) 轉換併裝決策問題為集合涵蓋問題

將一個貨物併裝組合視為一個集合，轉換為類似集合涵蓋問題(set covering problems, SCP)的數學規劃模式如下：

i ：貨物之編號，$i=1\sim n$ ； I：所有貨物所成的集合。

j ：指派航班之編號，$j=1\sim m$ ； J：為所有可指派航班 j 的集合。

k ：併裝組合之編號； $k\in K$ ； K：併裝組合空間(所有併裝組合所成之集合)。

N_j：航班j之併裝組合所成的集合。

c_k ：每個併裝組合k所對應的成本。

a_{ik}：當貨物i包含於併裝組合k之內時，$a_{ik}=1$；否則為 0。

Xr ：0-1 整數變數。當併裝組合k被選取時，$Xr=1$，否則為 0。

目標式：

$$Min\sum_{k\in K} c_k x_k \tag{1}$$

限制式：

$$\sum_{k\in K} a_{ik} x_k \geq 1 \qquad \forall i\in I \tag{2}$$

$$\sum_{k\in N_j} x_k \leq 1 \qquad \forall\, j\in J \tag{3}$$

$$Xr,\ binary \tag{4}$$

目標式(1)設定極小化總運價之目標，限制式(2)為所有貨物皆需要被涵蓋的限制式。限制式(2)中所使用的是不等號而非等號，也因此並未限制貨物只可被涵蓋一次。因此當求解發生貨物被涵蓋兩次以上的情形時，便需要進行修正，此即為集合涵蓋與集合分割問題(Set Partitioning Problem – SPP)之差別。限制式(2)使用不等式而非等式是因為一般而言求解 SCP 會較 SPP 容易。而且，即使所求得之解違反貨物僅能被涵蓋一次的限制，仍可於其後輕易藉由刪除併裝組合中重複之貨物，將

貨物修正為僅被涵蓋一次。限制式(3)為一般典型之 SCP 所沒有之限制式，其用途為限制每個航班最多僅能有一個併裝組合出現。限制式(4)表示所有 Xr 為 0-1 整數變數。

(二)求解拉式鬆弛問題

上述問題以拉式鬆弛法放鬆限制式(2)後，其模式如下，其中以u_i代表拉式乘數，$\boldsymbol{u}$代表所有u_i所形成之向量。

目標式：

$$L(\boldsymbol{u}) = Min\sum_{k\in K} c_k(\boldsymbol{u})x_k + \sum_{i\in I} u_i \tag{5}$$

限制式：

$$\sum_{k\in N_j} x_k \le 1 \qquad \forall\, j \in J \tag{6}$$

$$Xr\text{ , }binary \tag{7}$$

$$c_k(\boldsymbol{u}) = c_k - \sum_{i\in I_k} u_i \qquad \forall k \in K \tag{8}$$

I_k：針對併裝組合k，所有被涵蓋貨物所成之集合。(I_k ={ $i \in I$ ：a_{ik}=1})。

轉化後模式與典型集合涵蓋問題的拉式鬆弛問題相近，僅在於有無限制式(6)之差別。求解上述拉式問題時，若僅考慮目標式(14)與限制式(7)，則當$c_k(\boldsymbol{u})<0$ 則$Xr = 1$；反之，$c_k(\boldsymbol{u})>0$ 則$Xr = 0$；當$c_k(\boldsymbol{u})=0$ 則$Xr = 1$ 或 0。但考量限制式(6)，若某一航班所屬的併裝組合有多個$c(\boldsymbol{u})$值小於 0 時，則僅能選擇$c(\boldsymbol{u})$值最負者，至於實際運算上，則可透過$c(\boldsymbol{u})$值的排序來完成。

(三)修正拉式解為可行解及拉式乘數更新

因拉式問題為放鬆限制式後再進行求解,所以極可能求出違反限制式(6)(有貨物未被涵蓋)之不可行解。欲求得可行解，一般而言可透過拉式解的修正。然而該修正程序的設計上必須注意，一方面希望造成目標式值增加最少的情況下完成，但是又不宜太複雜，以免大幅增加運算的負荷。此研究之演算法主要係利用 $c(\boldsymbol{u})$值的觀念來進行修正，如前述求解拉式問題時，已求出各併裝組合的 $c(\boldsymbol{u})$值並進行排序，若將拉式解所選擇併裝組合之 $c(\boldsymbol{u})$值與所屬航班排序順位第二之 $c(\boldsymbol{u})$值相減，所得之差值代表將目前拉式解(排名首位)之併裝組合，置換為排序第二的併裝組合後，原目標值與新目標值之差。

因為希望在目標式增加最少的情況下找到可行解，因此在所有航班中應考慮以差值最小者嘗試加以置換。然而，為有利於盡快找到涵蓋全部貨物的可行解，是否進行置換則視以下檢查流程的結果而定。必須在置換後，原不可行解是否所涵蓋之貨物仍繼續被涵蓋，並增加一個以上未被涵蓋之貨物，才進行置換；否則，就放棄該併裝組合，並選取下一順位之併裝組合，對該航班重新計算 $c(\boldsymbol{u})$值之差，再於所有航班中選取 $c(\boldsymbol{u})$差值最小者進行置換檢查程序。重複上述流程，直至所有航班的所有併裝組合皆測試過。

在上述嘗試置換的過程中，併裝組合被選取進行置換檢驗過後即不再被保留做置換選擇，因此整個修正過程執行檢驗的次數大約僅為併裝組合數。上述方式並不保證定會產生可行解，若過程中若仍無法找到可行解時，在本次運算迴圈中，即逕以目前已發現的最佳可行解予以替換，若發現可行解，則更新目前最佳可行解的紀錄。

前述所求得之拉氏解與可行解之目標值可分別以 $L(\boldsymbol{u})$ 及 UB 來代表。有關拉式鬆弛法遞迴運算過程中，拉式乘數之修正，則可參考(9)、(10)式之運算(Held & Karp(1970)，式中 t 代表第 t 次遞迴運算，UB 代表的是目前最佳的上限值。

$$u_i^{t+1} = \max\left\{u_i^t + \lambda \frac{UB - L(\boldsymbol{u}^t)}{\left\|s(\boldsymbol{u}^t)\right\|^2} s_i(\boldsymbol{u}^t),\ \ 0\right\} \forall i \in I \tag{9}$$

$$s_i(\boldsymbol{u}^t) = 1 - \sum_{k \in K_i} x_k(\boldsymbol{u}^t) \qquad \forall i \in I \tag{10}$$

K_i：針對貨物i，所有涵蓋此項目之集合所成的集合，$\{\, i \in I：a_{ik} = 1\}$。

(四)併裝組合空間之調整機制：

原貨物併裝問題雖然可藉由 SCP 加以描述，但當問題規模加大時，其併裝組合空間將會極大，因此此研究先產生一個初始併裝組合的空間，配合拉式鬆弛法的遞迴運算，以一個有效合理的方式，將不合適的予以淘汰，並新增具有潛力之併裝組合，以改進現行併裝組合的空間，逐步求得近似最佳解。

- 併裝組合之刪減

如前述求解拉式解時，對各航班，係挑選 $c(\boldsymbol{u})$值小於 0 最負的併裝組合。由此可知，當 $c(\boldsymbol{u})$值越負時，代表該併裝組合越值得被挑選，也因此 $c(\boldsymbol{u})$值可以做為一個併裝組合是否應該被保留的指標。然而，在遞迴運算的過程中，隨拉式乘數 $\boldsymbol{u}$ 之更新，各併裝組合的 $c(\boldsymbol{u})$值也持續在變動中，為求得一穩定且較具參考價值的基準，本研究利用一個類似指數平滑法(exponential smoothing)之觀念，在每一次求解迴圈中，將每個併裝組合原分數乘上一個權重值 α，再加上該次的 $c(\boldsymbol{u})$值乘上$(1-\alpha)$後，做為參考分數。刪選併裝組合的方式，係藉由限制每航班之併裝組合數，來控制總併裝組合的數量。因此，對各個航班所屬的併裝組合，在每次的遞迴運算中，依據參考分數高低，僅保留一特定比例(如每航班 20 個)併裝組合，其餘的則刪除。

- 併裝組合空間之增加

主要係依據求解拉式解對併裝組合$c(\boldsymbol{u})$值的排序，配合特定貨物依序針對現有併裝組合進行增刪來產生。有關特定貨物的選取，係參考(10)式之$s_i(\boldsymbol{u})$值。挑選$s_i(\boldsymbol{u})$值最大者作為「加項」，挑選$s_i(\boldsymbol{u})$值最負者作為「減項」，當遇上$s_i(\boldsymbol{u})$值相同時，則以隨機方式挑選。其中，加項之意義為依$c(\boldsymbol{u})$值之排序逐一檢視現有併裝組合，若不含有加項，則將加項加入以產生新的併裝組合；反之，減項為針對現有併裝組合中若含有減項，則由併裝組合中剔除以產生新的併裝組合。

依此方式之理由，主要係基於$s_i(\boldsymbol{u})$值所代表的不僅是貨物涵蓋限制式(6)是否滿足，同時也是拉式乘數$\boldsymbol{u}$依據(9)式修正時的數值。因此依據加項產生新的併裝組合，因為其$s_i(\boldsymbol{u})$值最大，相對u_i的搭配也最大，潛在性地就有可能使得新增加之併裝組合在下一個迴圈依據(8)式計算$c(\boldsymbol{u})$時，其值越負，因而產生出較好的併裝組合。反之，利用刪除減項所產生的新併裝組合，其$c(\boldsymbol{u})$值也應該越負。

上述增加併裝組合的方式，對每一航班均設定了一個併裝組合數目的最大值，當達到最大值時即不再增加併裝組合。其理由一方面在於縮短運算的時間；另一方面因為加項減項的調整機制是依 $c(\boldsymbol{u})$的排序來進行，排序較後的併裝組合一般是較差的組合，其透過加項減項而產生好的組合之可能性亦較低。

另外，在集合涵蓋問題之解求解集合分割問題的方式時，針對重複涵蓋的貨物僅保留其在 $c(\boldsymbol{u})$值最負者之併裝組合中，在其他併裝組合中重複之貨物則予以剔

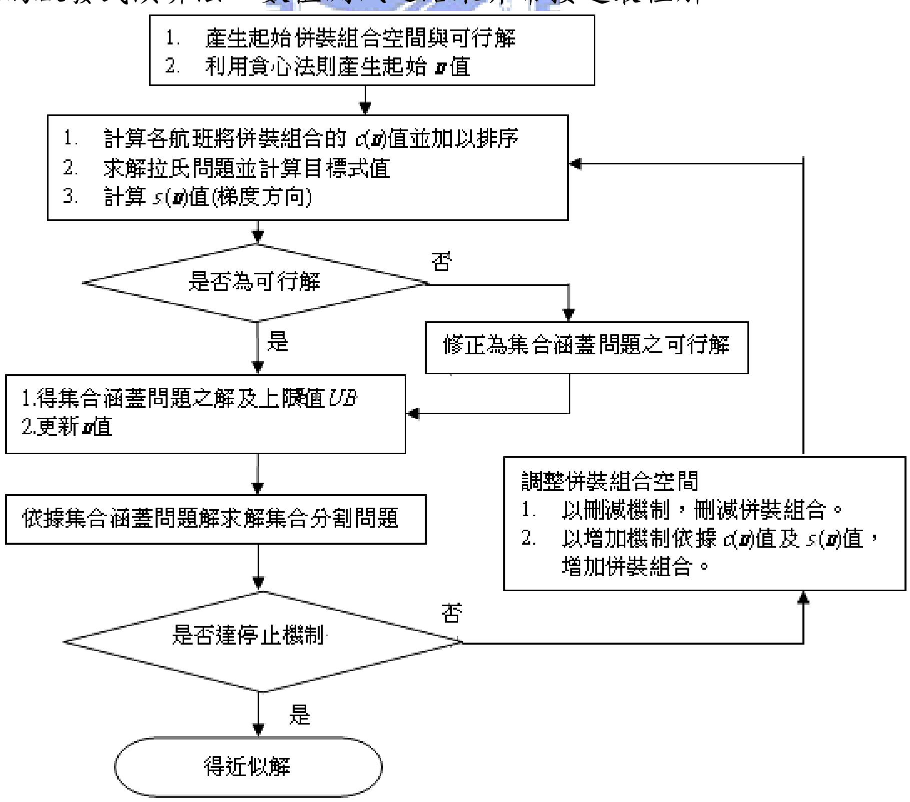
除。於其過程中，所產生之新併裝組合亦將之加入於並裝組合空間之中。

（五）停止機制與演算法流程摘要

一般以拉式鬆弛法求解典型之集合涵蓋問題時，是針對以固定的集合空間，因此當上下限值相等或差值在容忍範圍內時，即停止求解。但此演算法採用部分併裝組合空間在遞迴運算的過程中加以調整的方式求解，就上下限值的意義上已經有所不同，因此求解之停止機制也隨之改變。拉式鬆弛問題的目標式值 $L(\boldsymbol{u})$，針對該次迴圈之併裝組合空間，仍是一個有效的下限值，但是因為該併裝組合空間僅包含部份的併裝組合，其並不是原來貨物併裝問題的有效下限值。至於針對 SCP 所得的上限值 UB，因求解併裝組合空間僅包含部份的併裝組合，因此可視為一個區域最佳解。由於最佳解必小於或等於區域最佳解，所以此上限值對於原來貨物併裝問題來說仍是一個有效的上限值。

由此可知，此研究之演算法無法利用上下限值夾擠得到確切的最佳解所在區間，上下限值 UB 值與 $L(\boldsymbol{u})$值僅能僅做為下次迴圈參數修正之用。所以在停止機制，僅能利用「SCP 的解趨於穩定」，或是「當求解次數達到所設定次數」，來做為停止條件。以後述之數據測試實驗為例，演算法由最初求解至產生近似解的過程，大約需經過 500 次的求解，且隨問題規模的加大，求解次數略有需要增加趨勢。有關整個演算法之流程，如圖 3-1。

基於航空貨運計費機制的複雜性，航空貨運的併裝問題是一個相當困難的混合整數規劃問題，處理較大規模的問題時，其求解時間過長並不能符合業界作業的需求。此研究將該問題轉換成類似集合涵蓋問題的形式，再以拉式鬆弛法為基礎，發展一遞迴性的啟發式演算法，數值測試之結果非常接近最佳解。



紀玟豪求解併裝問題之演算法流程圖

簡　歷

姓名：曾筠予

學歷：

國立交通大學運輸科技與管理所畢業	民國 94 年 6 月
國立交通大學科技管理輔所畢業	民國 94 年 6 月
國立交通大學運輸科技與管理學系畢業	民國 92 年 6 月
國立交通大學財務工程學程	民國 92 年 6 月
台北市立松山高級中學畢業	民國 88 年 6 月

著作：

2004 年工研院創新與科技管理研討會

-台灣地區商業用電量預測模型研究

2003 海峽兩岸智慧運輸系統學術研討會

-都市高齡化社會智慧型運輸系統應用之研究-以台北市與新竹市為例

九十一年度國科會大專學生參與專題研究計畫

-都市高齡化社會智慧型運輸系統應用之研究

榮譽：

九十一年度國科會大專學生參與專題研究計畫研究創作獎

電子郵件信箱：doff.tem92g@nctu.edu.tw