

國立成功大學
工業與資訊管理研究所
碩士論文

廢棄物物流系統規劃之研究

指導教授：王逸琳 博士

研究生：楊文正

民國九十四年七月

國立成功大學
碩士論文

廢棄物物流系統規劃之研究

研究生：楊文正

本論文業經審查及口試合格特此證明
論文考試委員：

陳文智

陳文智

施勵行

施勵行

王逸琳

王逸琳

指導教授：王逸琳

系(所)主管：王泰裕

中華民國九十四年六月二十五日

博碩士論文授權書

(國科會科學技術資料中心版本 93.2.6)

本授權書所授權之論文為本人在 國立成功 大學(學院) 工業管理 系所
組 93 學年度第 2 學期取得 碩 士學位之論文。
論文名稱: 廢棄物物流系統規劃之研究

☒ 同意 ☐ 不同意

本人具有著作財產權之論文全文資料，授予行政院國家科學委員會科學技術資料中心(或其改制後之機構)、國家圖書館及本人畢業學校圖書館，得不限地域、時間與次數以微縮、光碟或數位化等各種方式重製後散布發行或上載網路。

本論文為本人向經濟部智慧財產局申請專利(未申請者本條款請不予理會)的附件之一，申請文號為：_____，註明文號者請將全文資料延後半年再公開。

☒ 同意 ☐ 不同意

本人具有著作財產權之論文全文資料，授予教育部指定送繳之圖書館及本人畢業學校圖書館，為學術研究之目的以各種方法重製，或為上述目的再授權他人以各種方法重製，不限地域與時間，惟每人以一份為限。

上述授權內容均無須訂立讓與及授權契約書。依本授權之發行權為非專屬性發行權利。依本授權所為之收錄、重製、發行及學術研發利用均為無償。上述同意與不同意之欄位若未鈎選，本人同意視同授權。

指導教授姓名: 王逸琳

研究生簽名:

學號: R36921019

(親筆正楷)

(務必填寫)

日期: 民國 94 年 8 月 8 日

1. 本授權書(得自<http://sticnet.stic.gov.tw/sticweb/html/theses/authorize.html> 下載或至<http://www.stic.gov.tw> 首頁右下方下載)請以黑筆撰寫並影印裝訂於書名頁之次頁。
2. 授權第一項者，請確認學校是否代收，若無者，請自行寄論文一本至台北市(106)和平東路二段 106 號 1702 室 國科會科學技術資料中心 黃善平小姐。(本授權書諮詢電話: 02-27377606 傳真: 02-27377689)

摘要

近年來，由於科技的進步與經濟的發展，讓人們享受著舒適便利的生活，但相對也製造了大量的廢棄物。如何抑制廢棄物的增加與資源回收變成重要的課題，尤其是在地狹人稠、資源缺乏的臺灣，更需要致力於廢棄物的減量與資源回收。行政院環保署大力推行多種物品的資源回收，迄今為止回收項目有廢一般容器、廢機動車輛、廢輪胎、廢潤滑油、廢乾電池、廢鉛蓄電池、農藥廢容器、廢電子電器物品及廢資訊物品、廢玻璃、廢日光燈管等十種類型，這些廢棄物透過社區民眾、地方政府、回收處理商及資源回收基金進行資源回收，建立一個完整的資源回收網路。

由於不同類別的廢棄物，其廠房成本與處理程序各有不同，而先前之研究大都假設所有廠房均可回收或處理所有類別的廢棄物。本研究將首度把新建廠房所需考慮之廠址選擇與廠房類別問題一併列入考慮，針對整個廢棄物回收網路，提出一個混整數規劃數學模型，讓決策者不止可以決定廠房位址，並可同時決定新建廠房可回收或處理之廢棄物類別，因而可做出更佳之決策。

因為本研究之數學模型較傳統問題更為複雜，特別在處理大規模問題時將耗費許多時間。因此本研究亦將採用並延伸 Jayaraman (2003) 所提出的啟發式演算法，提出二種改良式演算法 ($RS+HC+MHE$ 及 $MCC+MHE$)，來加快求解本問題的速度。本研究並依據臺灣的資源回收相關資訊，訂定各類參數並提出數個不同大小的問題來測試數學模型及演算效率。測試結果證實使用本研究之演算法的確可以有效地減少求解時間，且平均而言求出之解的品質與最佳解亦十分接近，因此本研究所提出之數學模型與解法應可供政府機構或廢棄物物流業者，在執行先導規劃時參考之用。

關鍵字：廢棄物回收網路、資源回收、演算法、混整數規劃。

Abstract

Reverse logistics has attracted a lot of research attention recently, especially when the disposition of electrical appliances increases. Different recycled appliances may consume different levels of resources and may require very different recycling processes. Conventional reverse logistics optimization model usually considers only the optimal facility locations and assumes the facility can conduct all different recycling processes. This paper investigates a mixed integer programming model that solves for both optimal facility locations and facility planning for a general reverse logistics network. In particular, in a planning stage for a reverse logistics company, our model considers both location and facility type —a facility that can only process a specific category of recycled products, or a facility that can process more categories of recycled products. Since our mathematical programming model is more complex and NP-hard, we propose several heuristic solution methods based on previous research and conduct computational experiments on several scenarios using data collected from Taiwan's reverse logistics market. The computational tests show that our proposed solution methods can give a high-quality solution in a promising time much shorter than the optimization software CPLEX.

Keywords: Reverse logistics; Facility location; Heuristics; Mixed Integer programming

誌謝

兩年前我一直質疑我真的能完成這篇論文嗎？真的可能嗎？因為那時候我連程式是啥碗糕都不知道，更遑論是要我寫出來，頓時覺得畢業對我來說好像遙遙無期似的，但經過兩年來不斷地學習、遇挫、解惑、豁然終於完成這篇論文，這都該歸功於我的老師－王逸琳老師，在研究領域上循序漸進地指導與嚴謹的研究態度，讓懶散的我能夠快速熱機且更謹慎地做研究。另外還要感謝我的審查老師施勵行老師及陳文智老師，感謝你們在百忙之中抽空審查我的論文，且不吝地給予建議與指教，讓我的論文更趨完善。

在論文的撰寫過程中，當然還要感謝研究室的同學們及學弟妹們，謝謝你們給予我不少的幫助，尤其是筑軍同學在程式撰寫上給了我許多指導與建議，真是感激不盡啊！

再來要感謝我的家人對我的支持與體諒，尤其是爸媽無怨無悔的照料下，讓我能夠在沒有負擔的情況下順利完成我的學業，真是太感謝你們了。至於我的女友佳芬，仔細想了想，她除了陪我吃喝玩樂之外，好像沒有什麼具體的貢獻，不過還是感謝妳陪我走過這六年光陰。最後，再一次謝謝大家，你們辛苦了！

謹將此篇論文獻給曾經陪我一起成長的所有人！

楊文正 2005 年 8 月於成大工管

目錄

摘要	I
目錄	II
表目錄	III
圖目錄	IV
第一章 緒論	1
1.1 研究動機	1
1.2 研究目的	3
1.3 研究流程	3
1.4 論文架構	4
第二章 文獻討探	5
2.1 臺灣目前資源回收的情況	5
2.2 逆向物流數學模型	9
2.2.1 符號定義	12
2.2.2 數學模型	15
2.3 演算法	18
第三章 模式建構與演算法	22
3.1 模式的建構目的與模型假設	22
3.1.1 模式的建構目的	22
3.1.2 模型假設	22
3.2 資源回收網路之數學模型	23
3.2.1 符號定義	23
3.2.2 數學模型	24
3.3 改良之啟發式演算法	30

第四章 例題測試與分析	34
4.1 參數估計	34
4.2 結果分析	38
第五章 結論與建議	59
參考文獻	61

表目錄

表 2-1 廢物品及容器回收量統計表	7
表 4-1 本研究分析的問題	34
表 4-2 各項廢棄物的歷年回收總量	35
表 4-3 出售處理後有用資源的收益和處理無用資源的費用	38
表 4-4 各項廢棄物的補貼金額	38
表 4-5 使用 CPLEX 與 Jayaraman et al. (2003) 演算法求解問題 1 所花費的時間	43
表 4-6 比較 CPLEX、本研究及 Jayaraman et al. (2003) 之演算法求解 問題 2' 的時間	44
表 4-7 比較使用 Jayaraman et al. (2003) 與本研究演算法求解問題 2' 之目標函式值差距	44
表 4-8 使用 CPLEX 與本研究演算法求解問題 1 所花費的時間	45
表 4-9 使用 CPLEX 與本研究演算法求解問題 2 所花費的時間	46
表 4-10 使用 CPLEX 與本研究演算法求解問題 3 所花費的時間	47
表 4-11 使用 CPLEX 與本研究演算法求解問題 4 所花費的時間	48
表 4-12 使用 CPLEX 與本研究演算法求解問題 5 所花費的時間	49
表 4-13 使用 CPLEX 與本研究演算法求解問題 1 之目標函式值差距	50
表 4-14 使用 CPLEX 與本研究演算法求解問題 2 之目標函式值差距	51
表 4-15 使用 CPLEX 與本研究演算法求解問題 3 之目標函式值差距	52
表 4-16 使用 CPLEX 與本研究演算法求解問題 4 之目標函式值差距	53
表 4-17 使用 CPLEX 與本研究演算法求解問題 4 之目標函式值差距	54

圖目錄

圖 1-1	四合一流程圖	2
圖 1-2	研究流程圖	4
圖 2-1	一般廢棄物回收系統圖	7
圖 2-2	回收處理機制運作圖	8
圖 2-3	Shih (2001) 資源回收網路圖	11
圖 3-1	本研究之資源回收網路圖	25
圖 3-2	回收點至新增回收廠之路徑	29
圖 3-3	回收點至新增處理廠廠之路徑	29
圖 3-4	回收廠至新增處理廠廠之路徑	30
圖 4-1	使用CPLEX與Jayaraman et al. (2003) 及本研究演算法求解 問題1所需時間之比較	55
圖 4-2	使用CPLEX與本研究演算法求解問題2所需時間之比較	55
圖 4-3	使用CPLEX與本研究演算法求解問題3所需時間之比較	55
圖 4-4	使用 CPLEX 與本研究演算法求解問題4所需時間之比較	56
圖 4-5	使用 CPLEX 與本研究演算法求解問題5所需時間之比較	56
圖 4-6	本研究之演算法在解問題1時，與CPLEX最佳目標函式值之差距	56
圖 4-7	本研究之演算法在解問題2時，與CPLEX最佳目標函式值之差距	57
圖 4-8	本研究之演算法在解問題3時，與CPLEX最佳目標函式值之差距	57
圖 4-9	本研究之演算法在解問題4時，與CPLEX最佳目標函式值之差距	57
圖 4-10	本研究之演算法在解問題5時，與CPLEX最佳目標函式值之差距	58
圖 4-11	本研究之演算法與 CPLEX 最佳目標函式值之差距的趨勢圖	58
圖 4-12	本研究之演算法與 CPLEX 求解所需時間之趨勢圖	58

第一章 緒論

本章主要在敘述有關本研究之背景與動機、研究目的、研究流程，以說明本研究之內容與架構。

1.1 研究背景與動機

台灣地區十多年來，隨著經濟的持續成長，垃圾與資源問題也日益嚴重。尤其台灣地區資源有限，更應積極考量如何有效將廢棄物轉化為資源並善加利用，所以資源回收更顯得重要。資源回收不僅可做到垃圾污染減量之預防工作，並可藉由資源回收再利用構成一完整資源之生命週期。在我國有限之資源限制及永續發展之原則下，更極需執行資源有效管理及再生利用。環保署一向積極推動資源回收工作，並自民國 86 年 1 月起推動推動資源回收四合一計劃（如圖 1-1），結合社區民眾、地方政府、回收處理商及資源回收基金進行資源回收，透過此四者合一，建立完整回收網路，確保資源垃圾確實回收再利用或妥善處理，並使參與民眾、清潔隊及回收商獲得合理利潤或獎勵，以確保回收體系之完整循環。環保署更於民國 87 年 7 月 1 日成立了「資源回收管理基金管理委員會」（簡稱基管會），推動建制廢一般容器、廢機動車輛、廢輪胎、廢潤滑油、廢乾電池、廢鉛蓄電池、農藥廢容器、廢電子電器物品及廢資訊物品、廢玻璃、廢日光燈管等回清除處理體系，亦即將之前的多個基金管理委員會整合成一個單位，統合國內資源回收的工作。

由於回收率有逐年上升的現象，各類廢棄物的回收數量勢必增加，因此諸如現有的資源回收網路是否足以應付新增的各類廢棄物數量，而新建的回收廠或處理廠之軟硬體是否亦應同時針對不同類別廢棄物而分別處理或綜合考量等等決策問題，益發顯得重要。

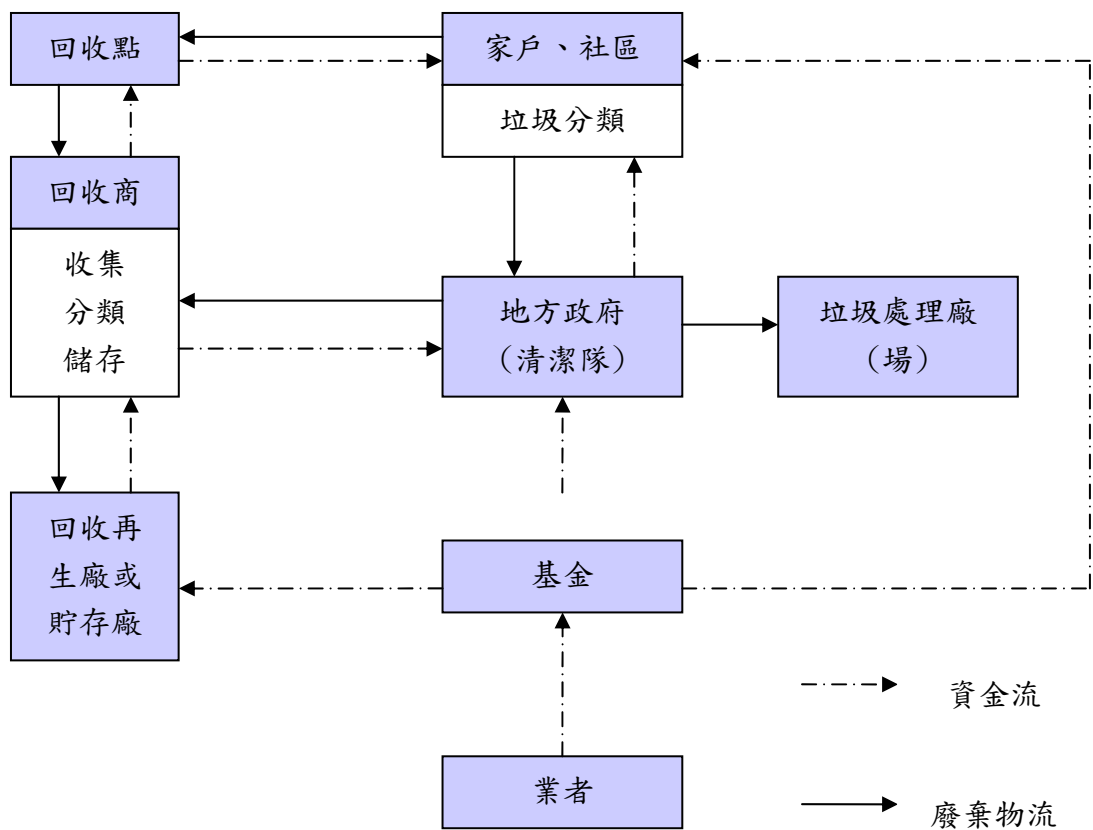


圖 1-1 四合一流程圖（資料來源：環保署回收基管會）

在 Shih (2001) 的研究中，曾提出台灣回收網路的數學模型，並探討在不同的回收率下，北台灣的廢家電與廢資訊產品回收網路是否應新增回收商或處理廠、最後處理廠，以及廢棄物該如何配送，以有效降低整體回收網路之成本，並提高整體回收之效率。由於 Shih (2001) 之研究把廢家電與廢資訊產品等回收物品視為同一類探討，而實際上，回收不同之廢棄物品所需之相關軟硬體設施與處理成本可能相異甚多，因此本研究將更進一步把廢家電與廢資訊產品分為二類不同的回收項目，探討其相關軟硬體設施之決策問題，且針對新增的回收廠或處理廠部分，將之分為兩類：（1）新增可回收或處理二類回收項目的廠房；（2）新增僅可回收或處理其中一類回收項目的廠房。本研究並針對此一多類回收項目之廠址配置與設施規劃問題提出新的數學模型；由於該數學模型為一 NP-hard 的問題，且較傳統的廠址配置及設施規劃問題更為複雜，假若使用 CPLEX 軟體求解

時，可能會花費許多時間在求解過程上。因此假若需要在短時間內得到一個概略的最佳解估計值之情況下，使用 CPLEX 軟體求解將無法滿足此類情況之需求。舉例來說，在初步擬定物流系統規劃之時，有時需提供即時之數據來探討計劃之可行性或其最佳解之特性；有時我們也需要藉由調整模式中之參數，探討某項參數對目標函數值的影響，進而調整該參數使目標函數值或最佳解朝計畫的理想方向趨近。諸如這些需要在短時間內得到一個概略的近似最佳解情況，應該要有比求解真正最佳解更快的方法。因此，本研究亦將試圖提出合適的啟發式演算法，來加快求解速度，以在更短的時間內求得近似最佳解之估計值。

1.2 研究目的

本研究將延伸傳統廢棄物回收網路（或稱逆向回收網路、逆向物流網路）之廠址配置與設施規劃問題，在設置新回收廠或處理廠時加入可以選擇興建可同時回收或處理二類回收項目的廠房，或是僅可回收或處理其中一類回收項目的廠房等決策。針對這個問題，本研究將提出一個新的資源回收網路數學模型，並用 CPLEX 求最佳解。同時，由於 CPLEX 求解該問題的時間偏長，本研究亦將試圖改進 Jayaraman et al.（2003）的啟發式演算法，提出更有效率的啟發式演算法。

1.3 研究流程

本論文的研究流程如圖 1-2 所示：

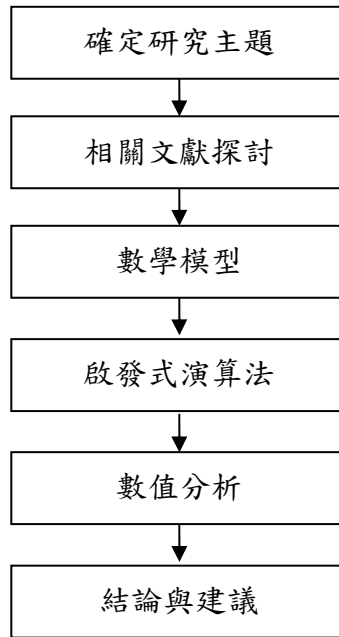


圖 1-2 研究流程圖

1.4 論文架構

本研究之論文共分為五個章節，分別是緒論、文獻探討、模式建構與演算法、例題測試與分析、及結論與建議等。在第一章緒論部份，主要在敘述有關本研究之背景與動機、研究目的、研究流程，以說明本研究之內容與架構。第二章文獻探討部份，將針對台灣目前資源回收的情況、逆向物流數學模型以及啟發式演算法三個部份作相關文獻的回顧。而第三章將建立本研究之資源回收網路的數學模型，並闡述本研究所提出之求解演算法。在第四章例題測試與分析中，本研究將依臺灣之廢棄物相關歷史資料，產生數個不同大小之例題，用以測試本研究提出之數學模式及求解演算法。第五章結論與建議將整理本研究之經驗與心得，提出具體之研究成果與貢獻，並對未來相關研究及待改善之處提出建議。

第二章 文獻探討

本章將針對台灣目前資源回收的情況、逆向物流數學模型以及啟發式演算法三個部份作相關文獻的回顧。

有關廢棄物的定義，章裕民（2003）指出廢棄物即泛指不要而丟棄足以影響環境衛生的物質，因此從習慣用語而言，只要是被丟棄的物品或物質皆可泛稱廢棄物。另再嚴謹的定義可為：「除依資源回收再利用法公告或經申請核准之再生資源項目外，任何物質、材料或物品被所有人、使有人、管理人或關係人丟棄、意欲丟棄、必須丟棄者，與由環境自然產生而足以影響環境衛生者稱之」。依台灣「廢棄物清理法」之明文規定，廢棄物分為下列二種：（1）一般廢棄物，其包含垃圾、糞尿、動物屍體或其他非事業機所產生足以汙染環境衛生之固體或液體廢棄物。（2）事業廢棄物，其又可分為二類：（Ⅰ）有害事業廢棄物，是指由事業機構所產生具有毒性、危險性，其濃度或數量足以影響人體健康或汙染環境之廢棄物。（Ⅱ）一般事業廢棄物，指由事業機構所產生有害事業廢棄物以外之廢棄物。

本研究所提到之廢棄物，主要以事業廢棄物為主。

2.1 台灣目前廢棄物回收制度及回收現況

廢棄物清理的責任可區分為政府責任、消費者責任及業者責任，目前依據我國廢棄物清理法第 15 條規定：物品或其包裝、容器（不易清除、處理；含長期不易腐化之成分；含有害物之成分；具回收再利用之價值）之製造、輸入或原料之製造、輸入業者、負責該物品或其包裝、容器之回收、清除、處理工作，並由販賣業者負責回收、清除工作。也就是採用業者責任制，即由產品的製造業者

(輸入業者)負責產品廢棄後之回收、清除及處理。業者責任制是將廢棄物清理責任由原來的政府或消費者轉由產品的製造、輸入業者負擔，除可降低政府的財政負擔及民眾的廢棄物清理費支出外，亦可促使業者於產品設計上考慮廢棄物處理之因素，而促進廢棄物的減量、再利用與回收再生。業者負責的型式可區分為實際回收清理及付回收清理費二種，多數實施業者責任制的國家採前者(歐、日等國)，德國的包裝回收系統(DSD)，我國則採後者，韓國則是由業者就未達目標回收量的部分付費。

鄭顯榮(2004)指出目前環保署依規定公告應回收之項目包括：(1) 容器類包括廢鐵容器、廢鋁容器、廢玻璃容器、廢紙容器、廢塑膠容器、一般環境衛生用藥廢容器、特殊環境衛生用藥廢容器、農藥廢容器。其中廢塑膠容器包括聚乙烯對苯二甲酸酯，如寶特瓶；聚乙烯，如不透明之清潔瓶；聚氯乙烯，如透明之沙拉油瓶；聚丙烯，如免洗餐具；發泡聚苯乙烯，如免洗餐具；未發泡聚苯乙烯，如養樂多瓶；及其他，但不含塑膠袋。(2) 物品類包括廢輪胎、廢潤滑油、廢乾電池、廢鉛蓄電池、廢日光燈管(直管)、廢機動車輛及廢棄之電視機、電冰箱、洗衣機、冷暖氣機、廢資訊物品及其週邊設備等。

環保署並公告相關之回收清除處理辦法，以執行資源回收業務。民國 87 年 7 月環保署成立「資源回收管理基金管理委員會」，推動資源回收計畫，以回收點、清潔隊、機關團體、社區、學校回收等方式實施回收工作。依環保署統計，91 年之資源回收物質計約 127 萬公噸，回收率達 15.5%總體效益相當於 58.76 億元。其中公告應回收廢棄物認證處理量為 71 萬 6 千公噸，詳如表 2-1，未公告資源回收物(如：廢紙、舊衣等)為 55 萬 5 千公噸，較 90 年分別成長 1.3%及 59%。由於回收處理量逐年增加，使台灣地區平均每人每日垃圾產生量由 86 年之 1.143 公斤、每年 888 萬公噸垃圾產量遞減至 91 年之 0.828 公斤、672 萬公噸。

92 年起環保署又訂定「垃圾資源回收分選廠興建計畫」、「巨大廢棄物回收再利用計畫」、「焚化灰渣資源化建廠計畫」，以進一步加強資源回收。

表 2-1 廢物品及容器回收量統計表

項目	廢鐵容器 (公斤)	廢鋁容器 (公斤)	廢玻璃容器 (公斤)	廢鋁箔包 (公斤)	廢紙容器 紙盒包 (公斤)	紙餐具 (公斤)	農藥廢容 器及特殊 環藥 (公斤)
總計	43,381,075	12,892,207	163,400,351	5,741,595	3,410,302	5,567,629	1,004,430
項目	PET (公斤)	PVC (公斤)	PP/PE (公斤)	PS 未發泡 (公斤)	PS 發泡 (公斤)	廢乾電池 (公斤)	廢照明光 源(公斤)
總計	69,082,969	2,359,921	48,651,578	1,930,186	491,319	1,016,562	7,891,706
項目	廢汽車 (輛)	廢機車 (輛)	廢鉛蓄電池 (公斤)	廢輪胎 (公斤)	廢潤滑油 (公升)	廢電視機 (台)	廢洗衣機 (台)
總計	142,549	182,994	41,778,207	120,541,496	9,008,457	473,564	263,324
項目	廢電冰箱 (台)	廢冷暖氣機 (台)	筆記型電腦 (件)	主機板 (件)	監視器 (件)	印表機 (件)	
總計	318,942	227,383	2,507	680,568	646,771	490,037	

(資料來源：環保署回收基管會之網站)

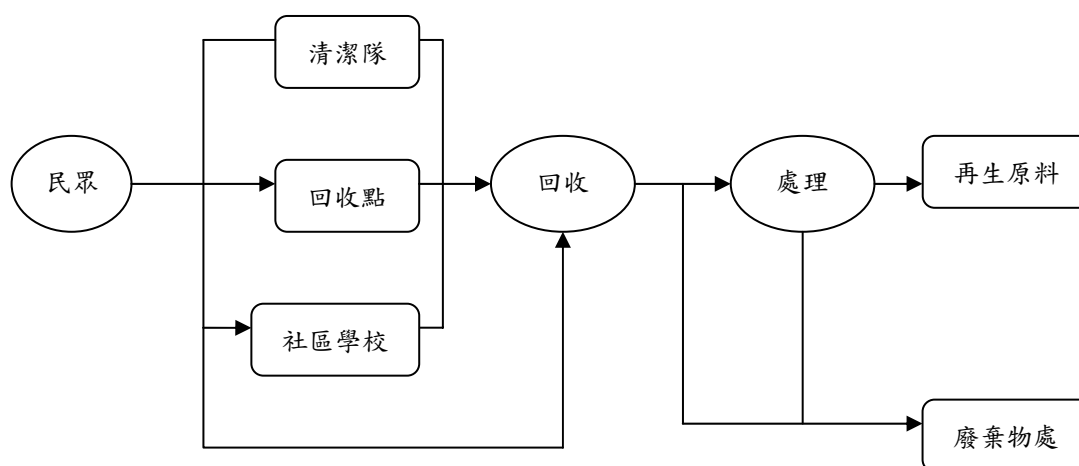


圖 2-1 一般廢棄物回收系統圖 (資料來源：鄭顯榮(2004))

另外，環保署依據「一般廢棄物回收清除處理辦法」第三十四條第一項，

於 92 年 4 月 24 日公告「一般廢棄物-廚餘再利用管理方式」，規範再利用用途包括有機質肥料、培養土、土壤改良之原料及動物飼料，並訂定「廚餘清運回收再利用計劃」，以加強推動廚餘回收再利用，進一步有效利用資源，並減少垃圾量。

圖 2-1 為一般廢棄物回收系統示意圖。前述之物品或其包裝、容器經使用廢棄後，民眾即可經由清潔隊、回收點、社區或直接由回收商予以回收後，送至環保署回收基管會登記有案之處理廠，進行再生處理。

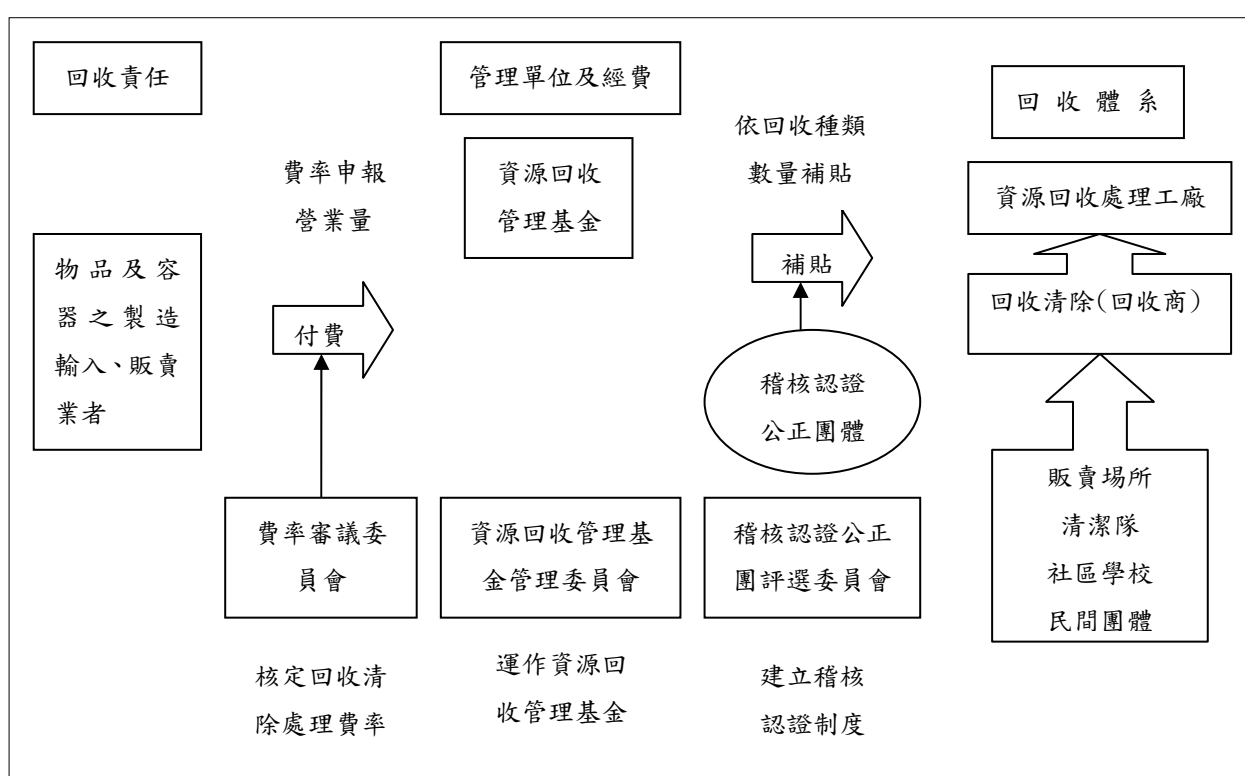


圖 2-2 回收處理機制運作圖 (資料來源：鄭顯榮(2004))

依環保署回收基管會統計資料顯示，93 年登記之「回收商」中，經營廢塑膠容器有 34 家、非塑膠容器有 58 家、廢乾電池有 27 家、廢輪胎有 15 家、廢日光燈管有 21 家、廢潤滑油有 37 家、廢鉛蓄電池有 12 家、廢機動車輛有 90 家、廢棄之電視機、電冰箱、洗衣機、冷暖氣機有 33 家、廢資訊物品及其週邊設備有 35 家。

93 年登記之「處理廠」中，經營廢塑膠容器有 15 家、非塑膠容器有 20 家、廢輪胎有 18 家、廢日光燈管有 4 家、廢潤滑油有 4 家、廢鉛蓄電池有 1 家、廢機動車輛有 3 家、廢棄之電視機、電冰箱、洗衣機、冷暖氣機有 7 家、廢資訊物品及其週邊設備有 8 家。以上之資料顯示我國資源回收處理體系大致已完備。

上述登記有案之回收點及社區學校，以及回收、處理商等則依「稽核認證機構」簽署認證之回收、處理量，向環保署回收基管會申領回收處理補貼費，以支應回收、清除、處理費用。該補貼回收處理費之來源則由依法指定公告之物品或其包裝、容器之製造、輸入或原料之製造、輸入業者，依規定之費率，繳交回收處理費，成立回收基金（信託基金）支應。回收處理機制運作方式如圖 2-2。

2.2 逆向物流數學模型

逆向物流網路（或稱廢棄物物流網路、逆向回收網路）問題有許多文獻，而本研究的問題是關於廠址配置 (facility location) 與設施規劃 (facility planning) 問題，以下將相關之研究文獻加以整理羅列。

Marin and Pelegrin (1998) 提出一個回收廠址配置 (Return Plant Location) 問題，為一個混整數規劃問題，目的是在顧客及供應商之間，設置回收廠，供應商可利用回收廠將產品送至顧客手中；而顧客也可利用回收廠將舊產品退至供應商，故成本包括運作成本、舊產品的回收成本以及新產品的運送成本。演算法根據拉氏分解 (Lagrangian decomposition) 的方法來簡化數學模型，加快求解速度。

Barros et al. (1998) 提出一個建築廢棄物和沙土的二階段逆向物流模型，來解決荷蘭的沙土物流問題，因為問題屬於混整數規劃問題，不易求解，故使用線性鬆弛 (linear relaxation) 的方法來簡化數學模型，以加快求解速度。

Shih (2001) 提出台灣回收家庭電器及資訊產品的逆向物流網路系統的數學模型，他針對北台灣資源回收網路系統進行研究。目標函式包括運輸成本、操作

成本、津貼、固定成本(設立新廠的成本)以出售處理後資源的收益等。限制式包括流量限制，回收廠／處理廠的能力限制、新設回收廠／處理廠／最後處理廠的數量限制等等。問題主要討探在不同回收率下，是否應新增回收廠或處理廠，以及最佳化流量。

Hu et al.(2002)提出一個多重時期多重廢棄物的最小成本逆物流網路系統模型。其目標函式為逆物流網路的最小總操作成本，限制式中考慮到內部因素，例如公司策略；外部因素，例如政府法規等。文中證實使用此模型搭配策略，可以有效降低逆物流網路成本。

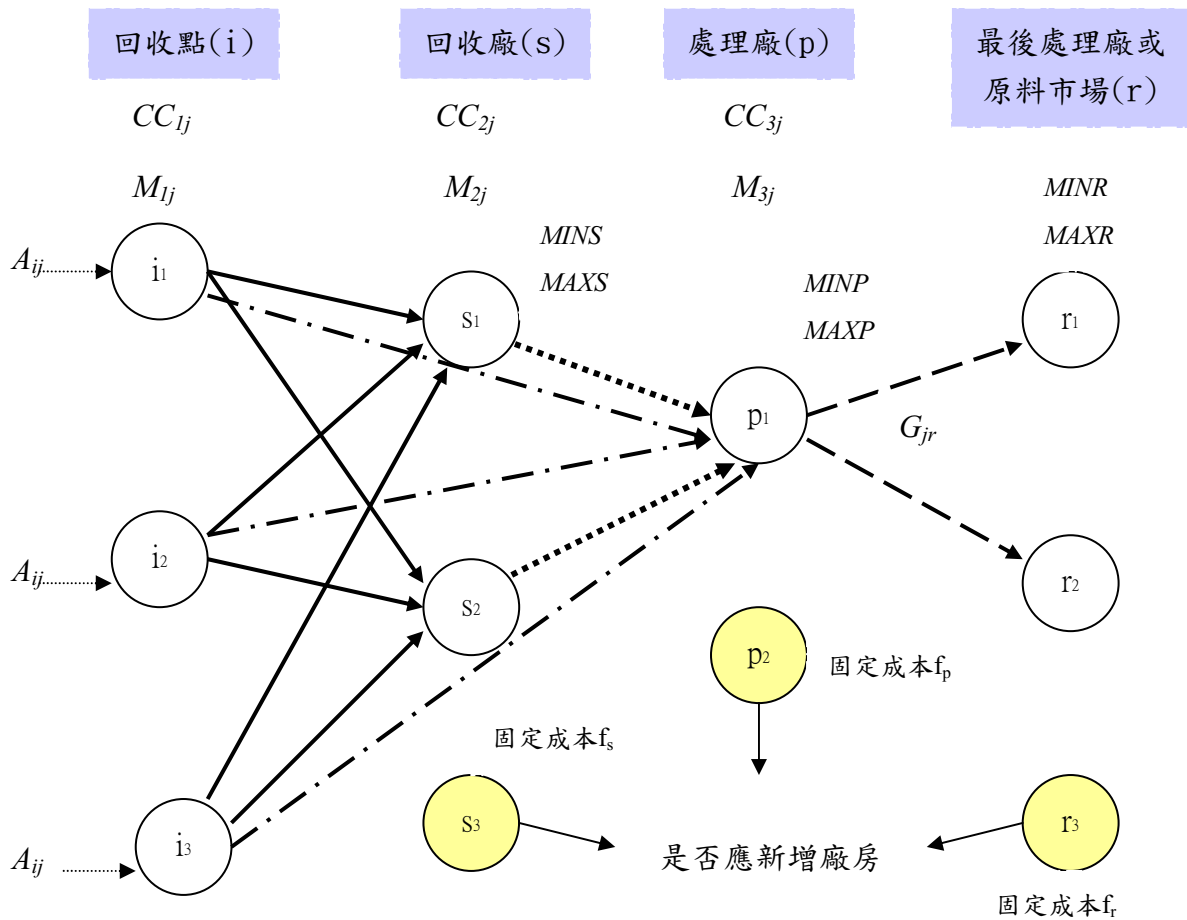
Jayaraman et al. (2003) 提出一個逆向配銷網路的混整數規劃數學模型，為加快求解速度，提出二種演算法：(1) 以隨機選出 (Random selection, *RS*) 搭配 heuristic concentration (*HC*) 以及 heuristic expansion (*HE*)；(2) 以程序 *CC* 來搭配 *RS* 與 *HC* 的方式，來做為演算法，並提出數據顯示其演算法在求解速度上優於 CPLEX。其演算法部份將在本章的下一節中再詳加說明。

Nagurney and Toyasaki (2004)提出一個結合電子廢棄物逆向供應鏈管理和資源回收的體制，它不僅考慮不同的決策者、電子廢棄物的組成成份、資源回收者、加工者，並且進一步考慮需求市場的均衡等等限制條件，進而提出一個數學模型。由於其模型太過複雜，為了求出最佳解進而提出特殊演算法來求解。

由於不良產品或受損產品的回收網路的成本過高，Min et al. (2004) 提出一個回收網路來有效地降低成本，其研究在零售商（或最終顧客）和製造商（或產品修護廠）之間，建立一個大型的集中分類中心，藉由大量的搜集、分類及運送來降低整個回收網路的成本。該研究並提出一個非線性混整數規劃模型，以基因演算法來求解，最後應用在實例上有著明顯的改善。

本研究乃根據Shih (2001) 所提出的數學模型，來作進一步的討探。圖2-3為Shih (2001) 所提出的資源回收網路圖。該資源回收網路圖中共有四類節點 (node)：(一) 回收點：指一般家庭、各縣市清潔隊、各機關學校、以及設有回收點的零售商店等等均視為回收點，在此只有初步收集功能。(二) 回收廠：回

收商到各地回收點收集廢棄物，然後經分類儲存後，再送往各處理廠。(三)處理廠：將廢棄物進一步拆解、處理，以分解出有用資源和無用的廢棄物(四)原料市場／最後處理廠：透過處理廠處理後，將有用資源出售至原料市場，而無用的廢棄物則透過最後處理廠(例如：掩埋廠、焚化廠等)加以處理。



A_{ij} ：估計的回收量。

G_{jr} ：廢棄物的原料比例。

$MINS$ ：回收廠的回收能力下限。

$MAXS$ ：回收廠的回收能力上限。

$MINP$ ：處理廠的處理能力下限。

$MAXP$ ：處理廠的處理能力上限。

$MINR$ ：最後處理廠的處理能力下限。

$MAXR$ ：最後處理廠的處理能力上限。

CC_{aj} ：運作成本， $a: 1, 2, 3$

M_{aj} ：津貼， $a: 1, 2, 3$

路徑	流量	運送成本
\longrightarrow	X_{isj}	$TCIS_j$
$\cdots\cdots\longrightarrow$	X_{spj}	$TCSP_j$
$-\cdot-\cdot\longrightarrow$	X_{ipj}	$TCIP_j$
$-\cdot-\cdot-\cdot\longrightarrow$	Z_{prg}	$TCPR_j$

圖 2-3 Shih (2001) 資源回收網路圖

在圖 2-3 中，主要有四種運送路徑，分別是：從回收點到回收廠、從回收廠到處理廠、從回收點到處理廠以及從處理廠到原料市場／最後處理廠等四種路徑。路徑不同，其運輸成本亦不盡相同。模式中也考慮到津貼部分，其津貼由基管會發放補助金，以回收數量多寡計算。當回收（或處理）量越多時，津貼就越多，且不同的廢棄物有不同的津貼。圖 2-3 中的 CC_{1j} 、 CC_{2j} 、 CC_{3j} 為各回收點、回收廠、處理廠的單位運作成本，與津貼相同，不同產品會有不同的運作成本，而回收（或處理）量越多時，運作成本就越多。每個回收廠、處理廠、最後處理場（或原料市場）均有回收處理能力的上、下限。最後透過處理廠處理後，可以將有用的資源販售至原料市場獲取利潤，而無用資源則運往最後處理廠處理，依照數量多少收取處理費用。

2.2.1 符號定義

以下將對 Shih（2001）之數學模式用到的符號、變數、及參數加以說明。

下標

i ：回收點。

j ：產品的種類（四種家庭電器及資訊用品）。

s ：回收廠。

p ：處理廠。

r ：最後處理廠或原料市場。

g ：原料之種類。

n ：現有之回收廠的個數。

m ：現有之處理廠的個數。

d ：現有之最後處理廠的個數。

q ：潛在回收廠的個數。

k ：潛在處理廠的個數。

v ：潛在最後處理廠 \ 原料市場的個數。

$h1$ ：新設回收廠的限制數量。

$h2$ ：新設處理廠的限制數量。

$h3$ ：新設最後處理廠的限制數量。

決策變數

L_g ：原料 g 的總量。

X_{isj} ：產品 j 從回收點 i 送到回收廠 s 的數量。

X_{spj} ：產品 j 從回收廠 s 送到處理廠 p 的數量。

X_{ipj} ：產品 j 從回收點 i 送到處理廠 p 的數量。

Z_{prg} ：原料 g 從處理廠 p 送到最後處理廠 r 的數量。

W_{sj} ：產品 j 通過回收廠 s 的總數量。

V_{pj} ：產品 j 通過處理廠 p 的總數量。

T_s ：廢棄物通過回收廠 s 的總量。

Q_p ：廢棄物通過處理廠 p 的總量。

U_{rg} ：原料 g 送往最後處理廠的總數量。

S_s ：0-1 變數，指是否應新建回收廠。

P_p ：0-1 變數，指是否應新建處理廠。

R_{rg} ：0-1 變數，指是否應新建最後處理廠。

其它參數

$B1$ ：指具有利益價值的有用原料。

$B2$ ：指需要花費成本來處理的無用物料。

B_g ：賣出每一單位原料的收益。

C_g ：處理每一單位原料的成本。

M_{1j} ：基管會給予回收點針對產品j之津貼。

M_{2j} ：基管會給予回收廠針對產品j之津貼。

M_{3j} ：基管會給予處理廠針對產品j之津貼。

$TCIS_j$ ：產品j從回收點i送往回收廠s的運送成本。

$TCSP_j$ ：產品j從回收廠s送往處理廠p的運送成本。

$TCPR_g$ ：產品g從處理廠p送往最後處理廠r的運送成本。

$TCIP_j$ ：產品j從回收點i送往處理廠p的運送成本。

D_{is} ：從回收點i送往回收廠s的距離。

D_{sp} ：從回收廠s送往處理廠p的距離。

D_{ip} ：從回收點i送往處理廠p的距離。

D_{pr} ：從處理廠p送往最後處理廠r的距離。

f_s ：新建回收廠的固定成本。

f_p ：新建處理廠的固定成本。

f_r ：新建最後處理廠的固定成本。

CC_{1j} ：產品j在回收點上每一單位的運作成本。

CC_{2j} ：產品j在回收廠上每一單位的運作成本。

CC_{3j} ：產品j在處理廠上每一單位的運作成本。

A_{ij} ：估計回收點i上產品j的數量。

G_{jg} ：產品j中原料g的比例。

$MINS$ ：回收廠 s 一年處理數量的下限。

$MAXS$ ：回收廠 s 一年處理數量的上限。

$MINP$ ：處理廠 p 一年處理數量的下限。

$MAXP$ ：處理廠 p 一年處理數量的上限。

$MINR_{rg}$ ：最後處理廠r處理原料g的年處理量之下限。

$MAXR_{rg}$ ：最後處理廠r處理原料g的年處理量之上限。

2.2.2 數學模型

Shih (2001) 提出的數學模型如下：

目標函式中包括運輸成本、操作成本、津貼、固定成本(設立新廠的成本)以及出售處理後有用資源的收益和處理無用資源的費用等等。限制式包括流量限制，回收廠／處理廠的能力限制、新設的回收廠／處理廠／最後處理廠的數量限制、非負變數和零一變數。

目標函數

$$\begin{aligned}
 \text{Max} \quad & \sum_{g \in B1} B_g \times L_g - \sum_{g \in B2} C_g \times L_g + \quad (\text{處理成本及收益}) \\
 & \sum_i \sum_j M_{1j} \times A_{ij} + \sum_i \sum_s \sum_j M_{2j} \times X_{isj} + \sum_s \sum_p \sum_j M_{3j} \times (X_{spj} + X_{ipj}) \quad (\text{津貼}) \\
 & - \sum_i \sum_s \sum_j (TCIS_j \times D_{is} \times X_{isj}) - \sum_p \sum_r \sum_L (TCP R_g \times D_{pr} \times Z_{prg}) \quad (\text{運送成本}) \\
 & - \sum_s \sum_p \sum_j (TCSP_j \times D_{sp} \times X_{spj}) - \sum_i \sum_p \sum_j (TCIP_j \times D_{ip} \times X_{ipj}) \\
 & - \sum_{s=n+1}^q f_s \times S_s - \sum_{p=m+1}^k f_p \times P_p - \sum_{r=d+1}^v f_r \times R_{rg} \quad (\text{固定成本}) \\
 & - \sum_i \sum_j CC_{1j} \times A_{ij} - \sum_i \sum_s \sum_j CC_{2j} \times X_{isj} - \sum_s \sum_p \sum_j CC_{3j} \times X_{spj} \\
 & - \sum_i \sum_p \sum_j CC_{3j} \times X_{ipj} \quad (\text{運作成本})
 \end{aligned} \tag{2-1}$$

限制式 (流量限制)

$$\sum_s X_{isj} + \sum_p X_{ipj} = A_{ij} \quad \text{for all } i, j \quad (2-2)$$

$$\sum_i X_{isj} = W_{sj} \quad \text{for all } s, j \quad (2-3)$$

$$\sum_p X_{spj} = W_{sj} \quad \text{for all } s, j \quad (2-4)$$

$$\sum_s X_{spj} + \sum_i X_{ipj} = V_{pj} \quad \text{for all } p, j \quad (2-5)$$

$$\sum_j V_{pj} \times G_{jg} = \sum_r Z_{prg} \quad \text{for all } p, g \quad (2-6)$$

$$\sum_p Z_{prg} = U_{rg} \quad \text{for all } r, g \quad (2-7)$$

$$\sum_r U_{rg} = L_g \quad \text{for all } r, g \quad (2-8)$$

(回收及處理能力的上下限)

$$\sum_j W_{sj} \geq MINS \times S_s \quad \text{for all } s \quad (2-9)$$

$$\sum_j W_{sj} \leq MAXS \times S_s \quad \text{for all } s \quad (2-10)$$

$$\sum_j W_{sj} = T_s \quad \text{for all } s \quad (2-11)$$

$$\sum_j V_{pj} \geq MINP \times P_p \quad \text{for all } p \quad (2-12)$$

$$\sum_j V_{pj} \geq MAXP \times P_p \quad \text{for all } p \quad (2-13)$$

$$\sum_j V_{pj} = Q_p \quad \text{for all } p \quad (2-14)$$

$$U_{rg} \geq MINR_{rg} \times R_r \quad \text{for all } r, g \quad (2-15)$$

$$U_{rg} \geq MAXR_{rg} \times R_r \quad \text{for all } r, g \quad (2-16)$$

(新增廠房的數量上限)

$$\sum_{s=n+1}^q S_s \leq h1 \quad (2-17)$$

$$\sum_{p=m+1}^k P_p \leq h2 \quad (2-18)$$

$$\sum_{r=d+1}^v R_{rg} \leq h3 \quad (2-19)$$

(非負變數及 0-1 變數)

$$L_g \geq 0, X_{isj} \geq 0, X_{ysj} \geq 0, X_{ipj} \geq 0, W_{sj} \geq 0, Z_{prg} \geq 0, V_{pj} \geq 0 \quad (2-20)$$

$$P_p \geq 0, R_{rg} \geq 0, T_s \geq 0, Q_p \geq 0, U_{rg} \geq 0, S_s \geq 0$$

$$S_s, P_p, R_{rg} \in [0,1] \quad (2-21)$$

在數學模型中，2-1 式為目標函數，求取回收體系之最大化利潤，其中包含了津貼 (M_{ij})、運送成本、運作成本、新建廠房的固定成本 (f_s 、 f_p 、 f_r) 以及出售處理後有用資源的收益 (B_g) 和處理無用資源的費用 (C_g)。在限制式方面，2-2 式為回收點i的流量守恒限制，即表示從回收點i送出的廢棄物總量要等於回收點i的回收總量；同理 2-3 及 2-4 式為回收廠s的流量守恒限制；2-5 式為處理廠的流量守恒限制；2-6 式為廢棄物透過處理廠處理後依照一定比例 (G_{jg}) 產生各種有用的資源及無用或有害的廢棄物；2-7 式表示各個最後處理廠r中第g類原料的數量；2-8 式為所有最後處理廠中第g類原料的總量；2-9、2-10、2-11 式為回收廠s的回收能力上、下限之限制；2-12、2-13、2-14 式為處理廠p的處理能力上、下限之限制；2-15、2-16 式為最後處理廠r的處理能力上、下限之限制；2-17、2-18、2-19 式為新增回收廠s、處理廠p、最後處理廠r的廠房數量上限；2-20、2-21 式表示非負變數以及零一變數。

Shih (2001) 提出的數學模型問題以北台灣的家電及資訊產品回收網路作為實例，討論在不同的回收率下，應新增回收廠、處理場、最後處理場的個數及最佳流量。該研究中僅以軟體求解，並未提出特殊演算法；其研究將家電與資訊產品等回收物品視為同一類，且對於不同類回收物之相關軟硬體設施之規劃決策並無著墨。本研究將更進一步將家電與資訊產品等回收物品分為二類不同回收項

目，來探討其廠址配置與設施規劃問題。由於模型中將考慮二類不同回收項目，故對於新增廠房而言，究竟是新增可回收或處理二類回收項目的廠房較好，抑或是新增僅可回收其中一類回收項目的廠房較佳等相關決策，本研究將進一步討論之。亦即，本研究針對新增的回收廠或處理廠的部分，將其分為兩類加以探討：

(1) 新增可回收或處理二類回收項目的廠房；(2) 新增僅可回收或處理其中一類回收項目的廠房。據此本研究將提出一個新的資源回收網路數學模型，並希望能夠有效地增加整體回收網路之總利益。

2.3 演算法

在求解廠址配置問題最大的難處，在於如何於所有潛在的廠址中選擇應新增之廠房。假如碰到擁有很多潛在廠址的問題時，就會造成可選擇之可能性的組合數過多，而使得求解速度變慢。為了克服這類問題，應設法想出一個方法可以縮小潛在廠址的個數，以加快求解速度但又不影響求得之解的品質。

Rosing and ReVelle (1997) 提出一種heuristic concentration (*HC*) 啟發式解法來求解p-median問題，並於Rosing et al. (1998) 中比較*HC*與禁忌搜尋 (Tabu Search) 法在求解p-median問題，發現 *HC*方法優於禁忌搜尋法。針對求解在逆向配銷網路中的廠址配置問題，Jayaraman et al.(2003)更進一步先用一種Random selection (*RS*) 的方式來縮小潛在回收廠及處理廠個數來求解，再採用*HC*並加上另一種heuristic expansion(*HE*)啟發式解法來改善求得之解。除此之外，Jayaraman et al. (2003) 亦提出一種*CC*啟發式解法取代*RS*與*HC*，並結合*HE*來求解，亦得到不錯的結果。據其測試結果顯示，當例子中潛在的回收廠及處理廠之個數越大時，此二種方法 (法1：*RS+HC+HE*；法2：*CC+HE*) 所求出之解將非常近似由CPLEX求出的最佳解，且其求解時間均明顯小於Weak CPLEX (將數學模式中的限制式鬆弛後，再以CPLEX求解的方法) 的求解時間；而其中更以*CC*搭配*HE*所

需的時間為最短。

由於 Jayaraman et al. (2003) 提出的網路問題及數學模型與本研究近似，且其演算法在求解本類問題的效率及結果均不錯，因此本研究將以其解法為基準，再進一步改良之。該演算法的邏輯乃在使用 *RS* 與 *HC* 有效地縮小潛在回收廠址及處理廠址個數，以便於求解；為防止求出的解有偏頗的情況，故又搭配 *HE* (類似 one-opt greedy heuristic)，以使求出的解更近似 CLPEX 的最佳解。其第一種演算法 (*RS+HC+HE*) 包含三大步驟：首先以 *RS* 法來縮小潛在的回收廠址及處理廠址個數來求解，但因所求出的解可能不太正確，為提高其正確率，該作者搭配了 *HC* 以及 *HE* 的方式，使解出的解更正確。而其第二種演算法 (*CC+HE*) 則包含二大步驟：以程序 *CC* (Procedure *CC*) 來代替 *RS* 及 *HC*，並搭配 *HE* 以求解。下面我們將各別介紹此兩種演算法的詳細步驟。

假設 P_{max} 表示可運作之回收廠的最大個數。而 Q_{max} 表示可運作之處理廠的最大個數，則 Jayaraman et al. (2003) 的第一種解法乃依序執行 *RS*、*HC*、*HE* 三大步驟：

一 · *RS*

假設一個隨機抽出的最大次數 $\text{MaxIterations} = b$ ($b=25, 50$, 或 100)

1. 當 $\text{MaxIterations} = b$ 就執行以下 2~5 的程序 b 次 (若 $b=100$ ，則就執行 100 次)。
2. 分別從回收廠及處理廠，隨機選出 P_{max} 個回收廠與 Q_{max} 個處理廠 (每個廠被抽取的機率均相同)。
3. 將選出的廠址放入數學模式中，利用 AMPL，將目前的問題解到最佳解。
4. 記錄此最好的解，假如求解出的解比先前的解佳時，則更新目前之最好的解。(此時最佳解中回收廠的個數為 P^* 、最佳處理廠的個數為 Q^*)
5. 終止。

二 · HC

6. 由RS求解出來的前5%最好的解中整理出最常出現的回收廠及處理廠的點，依序各取最常出現的 $P_{max} + 2 - P^*$ 個回收廠，及 $Q_{max} + 2 - Q^*$ 個處理廠，將之加到先前RS所選出之 P^* 個回收廠及 Q^* 個處理廠之潛在廠址名單（即，總共取了 $P_{max} + 2$ 、 $Q_{max} + 2$ 個潛在廠址）。
7. 利用AMPL，將目前的問題求解至最佳解。
8. 假如求出的解優於先前的解，則記錄該新求出的解。（此時最佳回收廠的個數為 P^{**} ，而最佳處理廠的個數為 Q^{**} ）
9. 回報目前找到之最好的解。

三 · HE

10. 將 P^{**} 及 Q^{**} 個別加1(由未被選取的回收廠及處理廠中，個別選取出一個)。
11. 利用AMPL,將目前的問題求解至最佳解。
12. 假如求出的解優於先前的解，則記錄此新求出的解。
13. 重複實行11~13的步驟，直到所有的回收廠及處理廠均曾被選取到為止。
14. 假如發現有改善,則使用所選出的回收廠或處理廠，並加以記錄。
15. 重複實行11~15的步驟，直到每次加一個點，而其解並不會有所改善為止。
16. 回報所找到的解

Jayaraman (2003) 的第二種解法，將原來的 RS 與 HC 步驟用另一個新程序 CC 取代之，並搭配 HE。

一 · 程序CC(Procedure CC)

- (1)將所有的回收廠及處理廠的點，依回收或處理產量之單位成本比例，由小至大排序。
- (2)分別從回收廠及處理廠中，以比例較小者選出 $P_{max} + 4$ 個點及 $Q_{max} + 4$ 個點，且並用AMPL求解。

二·HE

（步驟如前）使用程序CC所求出的最佳解作為起始點。

Jayaraman et al. (2003) 提出幾個例子測試其演算法求解的效率，其結果顯示，在品質方面，使用此二種演算法求出的解都有不錯的品質，其求出的解與最佳解的差距都在1%左右，其中以100次RS搭配HC、HE的品質較佳；在求解速度方面，雖然在小例子時，兩者求解速度都不及CPLEX，但當例子變大時，兩者求解時間都遠遠低於CPLEX求解時間，其中以CC搭配HE的求解速度較佳。雖然相較之下，兩種演算法求解效率都不錯，但是假若當潛在廠址之回收或處理產量之單位成本比例相近時，則CC搭配HE演算法求解的品質就會變差，而對RS搭配HC、HE演算法求解品質完全沒有影響。所以CC搭配HE演算法僅侷限於潛在廠址之回收或處理產量之單位成本比例相差很大時，才有效率。

本研究將以Jayaraman et al. (2003) 中的RS搭配HC、HE演算法為基準，進一步改良以求解廢棄物回收網路規劃問題，其更詳細的步驟將於後面章節整理之。

第三章 模式建構與演算法

本章將建立本研究之資源回收網路的數學模型，並闡述吾人提出之求解演算法。

3.1 模式的建構目的與模型假設

3.1.1 模式的建構目的

本研究將延續 Shih (2001) 所提出之資源回收網路的概念，探討在回收率逐年增加的情況下，是否應新增回收廠或處理廠，並計算最佳化流量；對於新增廠房之相關設施，本研究將更進一步加入可選擇興建回收或處理單一類別或是綜合類別廢棄物之設施。因此，本研究將基於 Shih (2001) 之數學模型，新增一些新的決策變數、參數及限制式。

3.1.2 模型假設

為了簡化問題，本研究做了以下假設：

1. 原先的回收廠及處理廠均可回收或處理二類回收項目之廢棄物。
2. 每個最後處理廠及原料市場，僅可處理一種原料或無用廢棄物。
3. 每個回收點均可回收二類回收項目之廢棄物。
4. 每個回收點都可以送至各個回收廠及處理廠，無路徑限制。
5. 每個回收點的回收量為已知的固定常數。

6. 各廢棄物可分解成各原料或無用廢棄物的比例為已知。
7. 各點間的路徑距離均以直線距離計算。
8. 每條路徑均假設無運載能量上限。
9. 假設所有回收物（包括日光燈管等）在運送途中無破損情況。

3.2 資源回收網路之數學模型

3.2.1 符號定義

本研究之數學模型與 Shih (2001) 之模型有類似的基本架構，因此將沿用第 2.2.1 節之定義，茲將本模型與前模型相異及新增之符號定義如下：

決策變數

X_{isj} ：產品j從回收點i送到回收廠s的數量。

X_{spj} ：產品j從回收廠s送到處理廠p的數量。

X_{ipj} ：產品j從回收點i送到處理廠p的數量。

Z_{pr} ：原料從處理廠p送到最後處理廠r的數量。

Y_{isj} ：與 X_{isj} 類似，亦代表廢棄物j從回收點i送到回收廠s的數量，其與 X_{isj} 之差別在於 Y_{isj} 乃針對新增之廠房s為可回收綜合類廢棄物之廠房而設計，而 X_{isj} 是對新增之廠房s為僅可回收單一類廢棄物之廠房而設計。

Y_{ipj} ：與 X_{ipj} 類似，亦代表廢棄物j從回收點i送到處理廠p的數量，其與 X_{ipj} 之差別在於 Y_{ipj} 乃針對新增之廠房p為可處理綜合類廢棄物之廠房而設計，而 X_{ipj} 是對新增之廠房p為僅可處理單一類廢棄物之廠房而設計。

Y_{spj} ：與 X_{spj} 類似，亦代表廢棄物j從回收廠s送到處理廠p的數量，其與 X_{spj} 之差別在於 Y_{spj} 乃針對新增之廠房p為可處理綜合類廢棄物之廠房而設計，而 X_{spj} 是

對新增之廠房 p 為僅可處理單一類廢棄物之廠房而設計。

S_{sj} ：0-1 變數，指是否應新建回收 j 類廢棄物的回收廠。

S_{s0} ：0-1 變數，指是否應新建回收二類廢棄物的回收廠。

P_{pj} ：0-1 變數，指是否應新建處理 j 類廢棄物的處理廠。

P_{p0} ：0-1 變數，指是否應新建處理二類廢棄物的處理廠。

其它參數

B_r ：賣出每一單位原料的收益。

C_r ：處理每一單位原料的成本。

$TCPR_r$ ：原料 r 從處理廠 p 送往最後處理廠 r 的運送成本。

f_{sj} ：新建回收 j 類廢棄物回收廠之固定成本。

f_{s0} ：新建回收二類廢棄物回收廠之固定成本。

f_{pj} ：新建處理 j 類廢棄物處理廠之固定成本。

f_{p0} ：新建處理二類廢棄物處理廠之固定成本。

G_{jr} ：產品 j 中原料 g 的比例。

$MINS_j$ ：回收廠 s 一年回收第 j 類廢棄物數量的下限。

$MAXS_j$ ：回收廠 s 一年回收第 j 類廢棄物數量的上限。

$MINP_j$ ：處理廠 p 一年處理第 j 類廢棄物數量的下限。

$MAXP_j$ ：處理廠 p 一年處理第 j 類廢棄物數量的上限。

3.2.2 數學模型

本研究之數學模式延續 Shih (2001) 所提出回收網路的概念，整個回收網路系統包括：回收點、回收廠、處理廠、最後處理廠（或原料市場）以及四種運送路徑：從回收點至回收廠、從回收點至處理廠、從回收廠至處理廠、從處理廠至最後處理廠（或原料市場）。回收點、回收廠及處理廠每回收一單位廢棄物，除

了可獲得基金管理委員會的津貼另外還有其運作成本。

圖 3-1 為本研究之資源回收網路案例，包含三個回收點、一個回收廠、一個處理廠以及三個原料市場（或最後處理廠），其中原料市場（或最後處理廠）僅能處理一種原料。其回收流程由回收點作初步收集，再送至回收廠做進一步分類、儲存後再送至處理廠或由回收點直接送往處理廠，最後透過處理廠將廢棄物轉換成原料後再運送到原料市場。整個回收路徑總共有四種，每種路徑的運送成本均不相同。

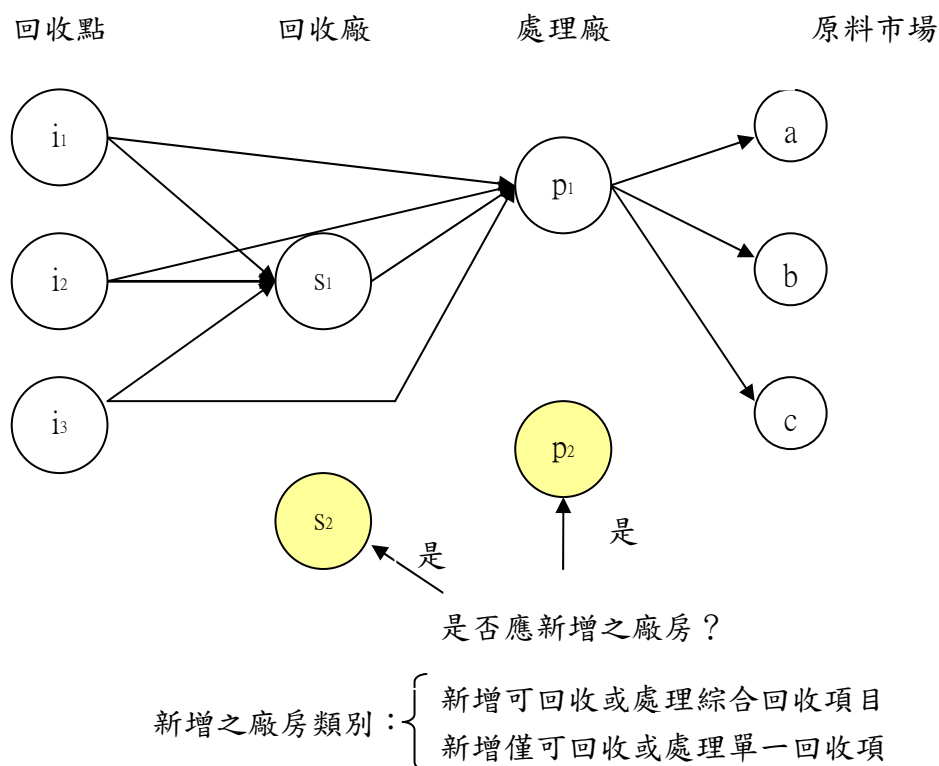


圖 3-1 本研究之資源回收網路圖

本研究之數學問題主要探討整體回收網路是否應新增新的回收廠或處理廠及其種類，並規劃最佳之廢棄物配送路徑，而使整體利益達到最大。其中特別針對新增的回收廠及處理廠方面新增兩個決策加以探討，分別是（1）新增可回收（或處理）二類別回收項目的廠房；（2）新增僅可回收（或處理）其中一類回收項目的廠房。另外，在 Shih (2001) 的模型中，主要在探討是否應新增回收廠、

處理廠、最後處理廠\原料市場等三種階段的廠址來提高資源回收網路的效率，而本研究僅探討回收廠及處理廠二種階段的廠址，並不探討最後處理廠\原料市場。因此我們假設最後處理廠\原料市場的廠址個數為已知的固定常數，且每個最後處理廠及原料市場，僅可處理一種原料\無用廢棄物，也就是說分解後的原料\無用廢棄物之種類（g）會等於最後處理廠\原料市場的廠址個數（r）。針對這些部分，本研究將修改 Shih（2001）限制式中 2-9、2-10、2-12、2-13 等回收廠及處理廠的回收（或處理）能力，將之修改成本研究數學模型中的 3-8~3-19 等限制式。

本研究之數學模式如下：

目標函數

$$\begin{aligned}
 \text{MAX} \quad & \sum_i \sum_j M_{1j} \times A_{ij} + \sum_i \sum_s \sum_j M_{2j} \times X_{isj} + \sum_i \sum_{s=n+1}^q \sum_j M_{2j} \times Y_{isj} & (\text{津貼}) \\
 & + \sum_i \sum_p \sum_j M_{3j} \times X_{ipj} + \sum_i \sum_{p=m+1}^k \sum_j M_{3j} \times Y_{ipj} \\
 & + \sum_s \sum_p \sum_j M_{3j} \times X_{spj} + \sum_s \sum_{p=m+1}^k \sum_j M_{3j} \times Y_{spj} \\
 & - \sum_i \sum_s \sum_j (TCIS_j \times D_{is} \times X_{isj}) - \sum_i \sum_{s=n+1}^q \sum_j (TCIS_j \times D_{is} \times Y_{isj}) & (\text{運送成本}) \\
 & - \sum_s \sum_p \sum_j (TCSP_j \times D_{sp} \times X_{spj}) - \sum_s \sum_{p=m+1}^k \sum_j (TCSP_j \times D_{sp} \times Y_{spj}) \\
 & - \sum_i \sum_p \sum_j (TCIP_j \times D_{ip} \times X_{ipj}) - \sum_i \sum_{p=m+1}^k \sum_j (TCIP_j \times D_{ip} \times Y_{ipj}) \\
 & - \sum_p \sum_r (TCPR_r \times D_{pr} \times Z_{pr}) \\
 & - \sum_i \sum_j CC_{1j} \times A_{ij} - \sum_i \sum_s \sum_j CC_{2j} \times X_{isj} - \sum_i \sum_{s=n+1}^q \sum_j CC_{2j} \times Y_{isj} & (\text{運作成本}) \\
 & - \sum_i \sum_p \sum_j CC_{3j} \times X_{ipj} - \sum_i \sum_{p=m+1}^k \sum_j CC_{3j} \times Y_{ipj} \\
 & - \sum_s \sum_p \sum_j CC_{3j} \times X_{spj} - \sum_s \sum_{p=m+1}^k \sum_j CC_{3j} \times Y_{spj}
 \end{aligned}$$

$$- \sum_{s=n+1}^q \left(\sum_j S_{sj} \times f_{sj} + S_{s0} \times f_{s0} \right) - \sum_{p=m+1}^k \left(\sum_j P_{pj} \times f_{pj} + P_{p0} \times f_{p0} \right) \quad (\text{固定成本})$$

$$+ \sum_{r \in B_1} \sum_p B_r \times Z_{pr} - \sum_{r \in B_2} \sum_p C_r \times Z_{pr} \quad (\text{收益及處理費用}) \quad (3-1)$$

限制式 (流量限制)

$$\sum_s X_{isj} + \sum_{s=n+1}^q Y_{isj} + \sum_p X_{ipj} + \sum_{p=m+1}^k Y_{ipj} \leq A_{ij} \quad \text{for all } i, j \quad (3-2)$$

$$\sum_i X_{isj} = \sum_p X_{spj} + \sum_{p=m+1}^k Y_{spj} \quad \text{for all } s \in [1, n], j \quad (3-3)$$

$$\sum_i X_{isj} + \sum_i Y_{isj} = \sum_p X_{spj} + \sum_{p=m+1}^k Y_{spj} \quad \text{for all } s \in [n+1, q], j \quad (3-4)$$

$$\sum_i X_{ipj} + \sum_s X_{spj} = Q_{pj} \quad \text{for all } p \in [1, m], j \quad (3-5)$$

$$\sum_i X_{ipj} + \sum_i Y_{ipj} + \sum_s X_{spj} + \sum_s Y_{spj} = Q_{pj} \quad \text{for all } p \in [m+1, k], j \quad (3-6)$$

$$\sum_j Q_{pj} \times G_{jr} = Z_{pr} \quad \text{for all } p, r \quad (3-7)$$

(新舊廠房的回收及處理能力的上下限)

$$\sum_i \sum_j X_{isj} \geq MINS_0 \times S_{s0} \quad \text{for all } s \in [1, n] \quad (3-8)$$

$$\sum_i \sum_j X_{isj} \leq MAXS_0 \times S_{s0} \quad \text{for all } s \in [1, n] \quad (3-9)$$

$$\sum_i \sum_j X_{ipj} + \sum_s \sum_j X_{spj} \geq MINP_0 \times P_{p0} \quad \text{for all } p \in [1, m] \quad (3-10)$$

$$\sum_i \sum_j X_{ipj} + \sum_s \sum_j X_{spj} \leq MAXP_0 \times P_{p0} \quad \text{for all } p \in [1, m] \quad (3-11)$$

$$\sum_i X_{isj} \geq MINS_j \times S_{sj} \quad \text{for all } s \in [n+1, q], j \quad (3-12)$$

$$\sum_i X_{isj} \leq MAXS_j \times S_{sj} \quad \text{for all } s \in [n+1, q], j \quad (3-13)$$

$$\sum_i \sum_j Y_{isj} \geq MINS_0 \times S_{s0} \quad \text{for all } s \in [n+1, q] \quad (3-14)$$

$$\sum_i \sum_j Y_{isj} \leq MAXS_0 \times S_{s0} \quad \text{for all } s \in [n+1, q] \quad (3-15)$$

$$\sum_i X_{ipj} + \sum_s X_{spj} \geq MINP_j \times P_{pj} \quad \text{for all } p \in [m+1, k], j \quad (3-16)$$

$$\sum_i X_{ipj} + \sum_s X_{spj} \leq MAXP_j \times P_{pj} \quad \text{for all } p \in [m+1, k], j \quad (3-17)$$

$$\sum_i \sum_j Y_{ipj} + \sum_s \sum_j Y_{spj} \geq MINP_0 \times P_{p0} \quad \text{for all } p \in [m+1, k] \quad (3-18)$$

$$\sum_i \sum_j Y_{ipj} + \sum_s \sum_j Y_{spj} \leq MAXP_0 \times P_{p0} \quad \text{for all } p \in [m+1, k] \quad (3-19)$$

(新增廠房的數量上限)

$$\sum_{s=n+1}^q \sum_j S_{sj} \leq h1 \quad (3-20)$$

$$\sum_{p=m+1}^k \sum_j P_{pj} \leq h2 \quad (3-21)$$

$$\sum_j S_{sj} \leq 1 \quad \text{for all } s \in [n+1, q] \quad (3-22)$$

$$\sum_j P_{pj} \leq 1 \quad \text{for all } p \in [m+1, k] \quad (3-23)$$

(非負變數及 0-1 變數)

$$\begin{aligned} X_{ipj} \geq 0, X_{isj} \geq 0, X_{spj} \geq 0, Z_{pr} \geq 0, Y_{ipj} \geq 0, Y_{isj} \geq 0, Y_{spj} \geq 0 \\ S_{sj} \geq 0, S_{s0} \geq 0, P_{pj} \geq 0, P_{p0} \geq 0 \end{aligned} \quad (3-24)$$

$$S_{sj}, S_{s0}, P_{pj}, P_{p0} \in [0, 1] \quad (3-25)$$

3-1 式為目標函式，求取整體回收體系之最大利潤，其中包含了津貼 (M_{ij})、運送成本、運作成本 (OC_{ij})、新建廠房的固定成本 (fs 、 f_p) 以及出售處理後有用資源的收益 (B_g) 和處理無用資源的費用 (C_g)；3-2 式限制從回收點流出的廢棄物數量不能超過回收點廢棄物回收之數量。3-3、3-4、3-5、3-6 式分別為回收廠、處理廠的流量守恒限制；3-7 式為從處理廠 p 送往第 r 類原料市場的數量，其中 G_{jr} 表示廢棄物中各原料之比例；3-8、3-9 式為舊回收廠的回收能力限制；3-10、3-11 式為舊處理廠的處理能力限制；3-12、3-13、3-14、3-15 式為新增回收廠的回收能力限制；3-16、3-17、3-18、3-19 式為新增處理廠處理能力限制；3-20、3-21 式為新增回收廠、處理廠之廠房數量上限；3-22、3-23 式表示新建的

回收廠或處理廠中，只能選擇其中一種類型（即，綜合類或單一類）的廠房來興建；3-24 式為變數非負之限制；3-25 式為零一變數之限制。

因為新建的回收廠或處理廠只能為一種類型（即，綜合類或單一類），本數學模型針對新增綜合類之回收廠及處理廠設計 Y_{isj} 、 Y_{spj} 、 Y_{ipj} 三種流量決策變數；而針對單一類之回收廠及處理廠則使用 X_{isj} 、 X_{spj} 、 X_{ipj} 三種流量決策變數，且 Y 與 X 這兩類變數不能同時為正。 $Y_{isj} > 0$ 代表 s 新增之回收廠可收取綜合類廢棄物。 $Y_{isj} = 0$ 代表 s 新增之回收廠不能收取綜合類廢棄物，也就是若 s 有新增回收廠的話，該廠僅可收取單一類廢棄物（此時 $X_{isj} > 0$ ）；或是 s 根本沒有新增回收廠（此時 $X_{isj} = 0$ ）。舉例來說，如圖 3-2 所示， s 為新增的回收廠，從回收點 i 到回收廠 s 之間需要另外加上 Y_{isj} 之流量，而限制式 3-8、3-9 式將使得 Y_{isj} 與 X_{isj} 兩者無法同時大於零。亦即當 s 為新建可回收二類回收項目的廠房時，則 Y_{isj} 大於零、 X_{isj} 等於零；而當 s 為新建僅可回收其中一類回收項目的廠房時，則 X_{isj} 大於零、 Y_{isj} 等於零；倘使 s 決定不新建廠房，則 Y_{isj} 、 X_{isj} 兩者均同時等於零。同理 Y_{spj} 、 Y_{ipj} 如圖 3-3、3-4 所示。

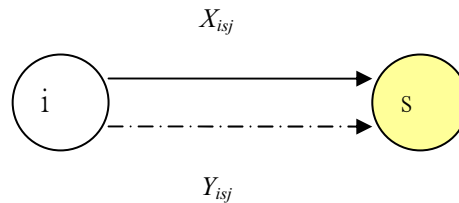


圖 3-2 回收點至新增回收廠之路徑

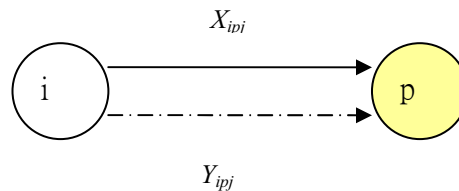


圖 3-3 回收點至新增處理廠廠之路徑

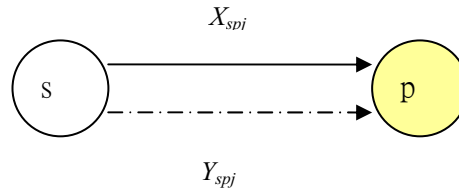


圖 3-4 回收廠至新增處理廠廠之路徑

3.3 改良之啟發式演算法

由於本研究的數學模型較 Jayaraman et al. (2003) 的數學模型更為複雜，因此我們改良 Jayaraman et al. (2003) 所提出的兩種演算法 $RS+HC+HE$ 與 $CC+HE$ 。在改良 $RS+HC+HE$ 演算法過程中，保留其 RS 與 HC 之精髓，大幅修改其 HE 部分，使它更適合本研究之數學模型。修改後之 HE 本研究將它稱為「 MHE 」，以 $RS+HC+MHE$ 演算法取代原來的 $RS+HC+HE$ 。在改良 $CC+HE$ 演算法過程中，除了保留原本 CC 所考慮的每單位固定成本的比例外，本研究又另外考慮每單位固定成本，以二者之總和作為排序的依據。修改後之 CC 本研究將它稱為「 MCC 」，以 $MCC+MHE$ 演算法取代原來的 $CC+HE$ 。

本研究之第一種演算法 $RS+HC+MHE$ 與原來的 Jayaraman et al. (2003) 所提出的 $RS+HC+HE$ 演算法之主要差別，在於 Jayaraman et al. (2003) 提出的 HE 演算法在經過實際測試本研究的數學模型後，會耗費太多的運算次數，導致其求解速度十分緩慢，跟原本的效率預期落差很大。為了使其演算法更適合本研究之數學模型，於是本研究著手進行修改 HE （而成為 MHE ），將其運算次數減少，但為了兼顧其求解品質，本研究也修改原來的 HC ，將其潛在廠址個數提高，以提高 MHE 的起始解品質。這樣不但可以加快整體演算法的速度，且其求出之解之品質也不會太差。

假設 $h1$ 表示可運作之回收廠的最大個數。而 $h2$ 表示可運作之處理廠的最大個數， $RS+HC+MHE$ 演算法之個別流程如下：

一・RS

1. 分別從回收廠及處理廠，隨機選出 $h1$ 個回收廠與 $h2$ 個處理廠(每個廠被抽取的機率均相同)。
2. 將選出的回收廠及處理廠放入本研究的數學模式中，並利用 CPLEX 將此混整數規劃問題解到最佳解。
3. 假如求解出的解比先前的解佳時，則更新目前之最好的解。
4. 反覆執行 1-3 步驟 100 次後，從求出 100 個解中，取出最好的解，此時求出的最好的解稱為「RS 解」，記錄 RS 解最好的解（此時 RS 解中回收廠的個數為 P^* 、處理廠的個數為 Q^* ），並記錄前 8 個最好的解。
5. 將 100 個解中出現的回收廠及處理廠加以記錄其出現的次數，最後依出現次數多寡排序，依序記錄在回收廠及處理廠的陣列中。
6. 終止。

二・HC

7. 將 RS 求解出來的前 8 個最好的解中出現的回收廠及處理廠的點各取 $h1-P^*+6$ 、 $h2-Q^*+6$ 個不同於 P^* 、 Q^* 所出現之點。把取出的廠址個數加到先前 RS 選出之 P^* 、 Q^* 個回收廠及處理廠上，則目前總共取了 $h1+6$ 、 $h2+6$ 個潛在廠址。
8. 將取出的潛在廠址放入數學模式中，並利用 CPLEX 將此混整數規劃問題求解至最佳解。
9. 假如求出的解優於先前 RS 解，則記錄該新求出的解，此時求出的最好的解稱為「HC 解」。（此時 HC 解中佳回收廠的個數為 P^{**} 、最佳處理廠的個數為 Q^{**} ）

10. 回報HC解。

三・MHE

11. 將 P^{**} 及 Q^{**} 各別加 $h1-P^{**}+6$ 、 $h2-Q^{**}+6$ （由未被選入HC解的潛在廠址中的回收廠及處理廠之陣列，各依序選取出6個）。

12. 將 $h1+6$ 及 $h2+6$ 的廠址放入數學模型中，利用CPLEX將目前的問題求解至最佳解。

13. 重複實行11~13的步驟，直到所有的回收廠及處理廠均曾被選取到為止。

14. 假如求出的解優於先前HC解，則記錄此新求出的解，此時求出的最好的解稱為「MHE解」。

15. 回報找到最佳解。

本研究所提出之第二種演算法MCC+MHE與原來的Jayaraman et al. (2003)所提出的CC+HE演算法之主要差別，除了我們用MHE來替代HE之外，我們觀察到原來的CC程序僅參考單位容量之廠房固定成本，然而實際上與廠房相關的成本應該還有運輸成本才對。因此，我們提出的MCC程序可視為同時考慮與廠房相關之單位固定成本與運輸成本，使能應該更接近實例，而給出更好的決策參考。

MCC+MHE演算法的流程說明如下：

一・MCC

(1)將所有的回收廠及處理廠的點，依回收或處理產量之單位固定成本及運送成本之總和，由小至大排序。

(2)分別從回收廠及處理廠中，以比例較小者選出 $P_{max}+4$ 個點及 $Q_{max}+4$ 個點，且並用CPLEX求解。

(3)此時求出的最好的解稱為「MCC解」。

二 · *MHE*

(步驟如前) 使用 *MCC* 所求出的最佳解作為起始點。

在下一章，我們將以數個數值例子來測試我們所提出之兩種演算法 (*RS+HC+MHE* 及 *MCC+MHE*) 的效率與正確程度，並與 Jayaraman et al. (2003) 的數種演算法及最佳化軟體 CPLEX 做比較分析。

第四章 例題測試與分析

本章將依臺灣之廢棄物相關歷史資料，產生數個不同大小之問題，測試本研究提出之數學模式及求解演算法。測試環境為一台使用 Windows XP 為 OS 且搭配 1GB RAM 之 Intel Pentium4 3.20GHz 的個人電腦，所有演算法皆以 C++ 語言撰寫，Visual C ++6.0 版本編譯，使用版本為 6.5.1 的 CPLEX 軟體之 callable library。

4.1 參數估計

本研究將以國內回收廢電子電器物品及廢資訊物品的回收系統做為例子，提出下列 5 個問題做為本研究分析的例子，且每個問題依據本研究的數學模式及演算法，將執行 10 次試驗。

以下描述本研究推估各類參數的原則與方法，基本上，本研究之測試參數例均自該參數之推估範圍內以 uniformly distribution 方式隨機產生。其中 i 表示回收點； $j=1$ 表示回收的廢棄物為廢電子電器、 $j=2$ 表示回收的廢棄物為資訊產品、 $j=0$ 表示同時回收廢電子電器及廢資訊產品二類：

表 4-1 本研究分析的問題

問題	回收點的個數	回收廠的個數	已建的回收廠個數	新建回收廠個數的限制 ($h1$)	處理廠的個數	已建的處理廠個數	新建處理廠個數的限制 ($h2$)
1	40	20	1	3	25	1	5
2	60	30	1	4	35	1	6
3	70	40	1	5	45	1	7
4	100	50	1	6	55	1	8
5	120	60	1	6	65	1	8

(資料來源：本研究自行整理)

表 4-2 各項廢棄物的歷年回收總量

	廢家電 (台)	廢資訊資產品 (台)
92	1,283,213	1,819,883
93	1,285,343	1,930,954
平均量	1,284,278	1,875,519

(資料來源：環保署回收基管會之網站)

1. 廢棄物回收總量 (A_{ij})：由環保署回收基管會統計結果如表 4-2 所示，本研究根據廢電子電器及資訊產品 92、93 年的平均來估計回收總量。其中廢電子電器的每年回收量估計為 1,284,278 件、廢資訊產品每年的回收量估計為 1,875,519 件。依據本研究提出 5 個問題中，回收點最大個數為 120 個，故本研究估計每個回收點的廢電子電器每年之回收量大約 10702 件、每個回收點的廢資訊產品每年之回收量大約 15629 件。則估計回收廠的每年回收量範圍如下所示：

回收點的回收量之範圍： $A_{i1} = [10500, 11500]$ 。

$A_{i2} = [15000, 16000]$ 。

2. 固定成本 (f_{sj} 、 f_{pj}) 及運作成本 (OC_{ij})：根據 Shih (2001) 研究顯示，回收廠每年攤提的固定成本大約 71 萬元，處理廠每年攤提的固定成本大約 2200 萬元。運作成本方面，回收廠平均每件廢電子電器及廢資訊產品的運作成本大約在 50~100 元之間，處理廠平均每件廢電子電器及廢資訊產品的運作成本大約在 150~300 元之間。則估計回收廠及處理廠的固定成本、運作成本範圍如下所示：

回收廠的固定成本之範圍： $f_{sj} = [660000, 760000]$ 。

處理廠的固定成本之範圍： $f_{pj} = [17000000, 27000000]$ 。

回收廠的每件運作處理成本之範圍： $OC_{ij} = [50, 100]$ 。

處理廠的每件公斤運作成本之範圍： $OC_{ij} = [150, 300]$ 。

3. 回收廠的回收能力 ($MINS_j$ 、 $MAXS_j$) 及處理廠的處理能力 ($MINP_j$ 、 $MAXP_j$)：
- 有關可回收綜合類之回收廠及處理廠的回收（處理）能力上限 ($MAXS_0$ 、 $MAXP_0$) 估計方法，本研究依Shih（2001）之研究，以廢電子電器及廢資訊產品其回收廠每年可回收約 60 萬件，及其處理廠每年可處理約 30 萬件的能力來估算。至於有關回收單一類的回收廠的回收能力上限及處理廠的處理能力上限 ($MAXS_1$ 、 $MAXS_2$ 、 $MAXP_1$ 、 $MAXP_2$) 以及各類回收廠及處理廠的回收（處理）能力下限 ($MINS_0$ 、 $MINS_1$ 、 $MINS_2$ 、 $MINP_0$ 、 $MINP_1$ 、 $MINP_2$) 等等參數之估計方法，由於缺乏相關資料，本研究乃自行設定其回收廠的回收能力及處理廠的處理能力範圍估計值如下所示：

回收廠的回收能力之範圍（公斤）：

$$MAXS_0 = [600000, 650000]。$$

$$MAXS_1 = [500000, 550000]。$$

$$MAXS_2 = [450000, 500000]。$$

$$MINS_0 = [300000, 325000]。$$

$$MINS_1 = [250000, 275000]。$$

$$MINS_2 = [225000, 250000]。$$

處理廠的處理能力之範圍（公斤）：

$$MAXP_0 = [300000, 350000]。$$

$$MAXP_1 = [250000, 300000]。$$

$$MAXP_2 = [200000, 250000]。$$

$$MINP_0 = [150000, 175000]。$$

$$MINP_1 = [125000, 150000]。$$

$$MINP_2 = [100000, 125000]。$$

4. 運送成本 ($TCIS_j$ 、 $TCSP_j$ 、 $TCIP_j$ 、 $TCPR_r$)：由於無確實的資料，所以本研究乃自行設定估算。假設由回收點直接送到處理廠的單位運送成本會高於其它階段的單位運送成本。估計各階段之運送成本範圍如下所示：

估計各階段之運送成本（元／件）：

$$TCIS_1 = [60, 80] \times (\text{回收點}i\text{到回收廠}s\text{之距離})。$$

$$TCIS_2 = [20, 30] \times (\text{回收點}i\text{到回收廠}s\text{之距離})。$$

$$TCSP_1 = [60, 80] \times (\text{回收廠}s\text{到處理廠}p\text{之距離})。$$

$$TCSP_2 = [20, 30] \times (\text{回收廠}s\text{到處理廠}p\text{之距離})。$$

$$TCIP_1 = [100, 120] \times (\text{回收點}i\text{到處理廠}p\text{之距離})。$$

$$TCIP_2 = [40, 50] \times (\text{回收點}i\text{到處理廠}p\text{之距離})。$$

5. 出售處理後有用資源的收益 (B_r) 和處理無用資源的費用 (C_r)、津貼 (M_{1j} 、 M_{2j} 、 M_{3j})：廢電子電器及廢資訊產品經過處理可分解成鐵、銅、鋁、塑膠、玻璃、馬達／壓縮機、金屬絲、可回收之冷媒等有用資源，以及潤滑油、石墨、螢光粉、聚氨酯、印刷電路板、樹脂等無用資源。表 4-3 表示各種有用資源的收益以及處無用（有害）資源的費用。在津貼方面，本研究根據環保署回收基管會所公佈各種廢電子電器及廢資訊產品，如表 4-4 所示。估計出售處理後有用資源的收益和處理無用資源的費用、津貼範圍如下示：

$$\text{有用資源的收益（元／公斤）：} Br = [0, 35]。$$

$$\text{處理無用資源的費用（元／公斤）：} Cr = [1, 13]。$$

$$\text{津貼（元／台）：} M_{21}、M_{31} = [346, 636]。$$

$$M_{22}、M_{32} = [182, 303]。$$

表 4-3 出售處理後有用資源的收益和處理無用資源的費用

	原料	(元／公斤)
收益	鐵	2.3
	銅	35
	鋁	25
	塑膠	3.5
	玻璃	0.1
	馬達／壓縮機	4
	金屬絲	10
	可回收之冷媒	12
成本	潤滑油	5
	石墨	1.1
	螢光粉	13
	聚氨酯	2.4
	印刷電路板	13
	樹脂	1.1

(資料來源：Shih (2001))

表 4-4 各項廢棄物的補貼金額

	回收清除處理補貼費用
廢電子電器	1、廢電視機：379.5 元/台
	2、廢洗衣機：346.5 元/台
	3、廢電冰箱：635.5 元/台
	4、廢冷暖氣機：410.5 元/台
廢資訊產品	1、廢主機：182 元/台
	2、廢監視器：215 元/台
	3、廢筆記型電腦：303 元/台
	4、廢印表機：192 元/台

(資料來源：環保署回收基管會之網站)

4.2 測試結果分析

本研究除了產生如表 4-1 所提出的 5 組測試問題之外，另外產生一組與問題 2 有全相同設定之問題 2'，用以測試本研究之二種演算法（ $RS+HC+MHE$ 及 $MCC+MHE$ ）以及 Jayaraman et al. (2003) 提出之演算法（ $RS+HC+HE$ 及 $CC+HE$ ），並與 CPLEX 求解結果相比較其求解時間與品質。

首先，本研究先以問題 1 及問題 2' 為例，比較 $RS+HC+MHE$ 、 $MCC+MHE$ 、 $RS+HC+HE$ 、 $CC+HE$ 及 CPLEX 之求解時間與品質。表 4-5 顯示，使用 Jayaraman et al. (2003) 的演算法（ $RS+HC+HE$ 及 $CC+HE$ ）求解問題 1 時，在 10 次的測試中，除了在第 4 次的測試中，使用 RS 搭配 HC 、 HE 演算法的總時間（322.452 秒）小於使用 CPLEX 求解所需的時間（417.453 秒）以外，其餘均大於 CPLEX 求解所需的時間如圖 4-1 所示。就平均時間而言，使用 $RS+HC+HE$ 演算法平均所需時間為 801.7345 秒、使用 $CC+HE$ 演算法平均所需時間為 832.2432 秒，都遠大於 CPLEX 求解所需的平均時間 303.6857 秒。然而，若使用本研究所提出的二種演算法（ $RS+HC+MHE$ 及 $MCC+MHE$ ）求解問題 1，在 10 次的測試中（其結果見表 4-8），使用（ $RS+HC+MHE$ 及 $MCC+MHE$ ）演算法的總時間均小於使用 CPLEX 求解所需的時間。如圖 4-1 所示，在平均時間方面，使用 $RS+HC+MHE$ 演算法平均所需時間為 106.8434 秒，而使用 $MCC+MHE$ 演算法平均所需時間為 73.6843 秒，都小於 CPLEX 求解所需的平均時間 303.6857 秒。由上述得知，在解問題 1 時， $RS+HC+MHE$ 及 $MCC+MHE$ 二種演算法之求解速度皆優於 $RS+HC+HE$ 及 $CC+HE$ 二種演算法；另外， $RS+HC+HE$ 及 $CC+HE$ 之求解時間又大於 CPLEX 之求解時間，因此針對求問題 1 之最佳解來說， $RS+HC+HE$ 及 $CC+HE$ 二種演算法已無使用上的需要（因為用 CPLEX 即可）。因此，在求解問題 1 時，使用本研究所提出之二種演算法（ $RS+HC+MHE$ 及 $MCC+MHE$ ）將比 Jayaraman et al. (2003) 所提出之二種演算法（ $RS+HC+HE$ 及 $CC+HE$ ）來得更有效率。另外，由表 4-6、4-7 中亦顯示，若再求解更大的例子（例題 2'）

時，本研究的演算法 $RS+HC+MHE$ 及 $MCC+MHE$ 之求解平均時間分別為 879.3768 秒及 532.5762 秒，皆仍優於 Jayaraman et al. (2003) 之演算法 $RS+HC+HE$ 及 $CC+HE$ 的平均時間 3625.129 秒及 3716.423 秒；雖然使用 $RS+HC+HE$ 及 $CC+HE$ 求出的解品質比本研究之 $RS+HC+MHE$ 及 $MCC+MHE$ 略佳，但其差距相當微小。故整體來說，在求解本研究之資源回收網路問題時， $RS+HC+MHE$ 及 $MCC+MHE$ 比 $RS+HC+HE$ 及 $CC+HE$ 更有效率。因此，以下之測試，我們將不再拿 Jayaraman et al. (2003) 之演算法 ($RS+HC+HE$ 及 $CC+HE$) 來比較，而是直接用我們的演算法 ($RS+HC+MHE$ 及 $MCC+MHE$) 與 CPLEX 比較求解效率。

接著，本研究以問題 1、2、3、4、5 等五個問題做為例子，代入數學模式中，使用 CPLEX 及本研究之演算法 (RS 搭配 HC 、 MHE 演算法及 MCC 搭配 MHE 演算法) 各執行 10 次試驗後，其結果顯示於表 4-5 及表 4-8~4-17 中。

表 4-5 及表 4-8~4-12 為使用本研究之二種演算法 ($RS+HC+MHE$ 及 $MCC+MHE$) 以及 Jayaraman et al. (2003) 提出之演算法 ($RS+HC+HE$ 及 $CC+HE$) 求解問題 1、2、3、4、5 所需的時間與使用 CPLEX 求解所需的時間做比較。求解問題 1 的結果 (見表 4-5、4-8 及 4-13)，已於前面討論過。表 4-9 顯示，當求解較大例題 (問題 2) 時，使用 $RS+HC+MHE$ 演算法平均所需的總時間為 882.7098 秒，而使用 $CC+MHE$ 演算法平均所需的總時間為 526.2612 秒，兩者皆少於 CPLEX 求解所需的平均時間 8010.264 秒；且在 10 次的測試當中，二種演算法 ($RS+HC+MHE$ 及 $MCC+MHE$) 求解所需的總時間均小於 CPLEX 求解所需的時間，如圖 4-2 所示。表 4-10 顯示，當求解更大的例題 (問題 3) 時，使用 CPLEX 求解所需的平均時間 (15499.69 秒) 已經遠遠大於使用 $RS+HC+MHE$ 演算法的平均時間 (2593.864 秒) 以及 $MCC+MHE$ 演算法的平均時間 (1873.591 秒)，如圖 4-3 所示。相同地，可由表 4-11 及表 4-12 中看出，使用 ($RS+HC+MHE$ 及 $MCC+MHE$) 在求解問題 4 及問題 5 時，其所需的時間與使用 CPLEX 求解所需的時間，二者之間的差距越來越大。由這些測試數據顯示，當例子越大時，使用

($RS+HC+MHE$ 及 $MCC+MHE$) 二種演算法可以更有效地縮短求解時間。

表 4-13~4-17 分別表示使用本研究之二種演算法 ($RS+HC+MHE$ 及 $MCC+MHE$) 與使用 CPLEX 在求解五個問題時，兩種演算法之目標函式值與最佳目標函式值的差距。由表 4-13 可知，當只使用 RS 演算法求解問題 1 時，其求出的解與最佳解的目標函式值差距平均為 4.64%；以 RS 求得之解當作起始解並搭配 HC 演算法求解問題 1 時，其平均差距降為 2.12%；若再進一步以 RS 搭配 HC 、 MHE 求解問題 1，平均差距則降為 0.798%。當只使用 CC 演算法求解問題 1 時，其求出的解與最佳解的目標函式值差距平均為 1.49%；若再進一步以 CC 搭配 MHE 求解問題 1，平均差距則降為 0.44%，如圖 4-6 所示。類似的情況也出現在求解問題 2、問題 3、問題 4、問題 5 之時。表 4-14 顯示，在求解問題 2 時，依序使用 RS 、 HC 、 MHE 演算法，可使與最佳解的目標函式值差距由平均 6.61% 下降到 3.19%，最後降為 1.1%；而依序使用 MCC 、 MHE 則可使與最佳解的目標函式值差距由平均 3.68% 下降到 1.22%，如圖 4-7 所示。表 4-15 顯示，求解問題 3 時，依序使用 RS 、 HC 、 MHE 演算法，一樣可使其與最佳解的目標函式值差距由平均 7% 下降到 5%，最後降為 1.9%；而依序使用 MCC 、 MHE 則可使與最佳解的目標函式值差距由平均 3.4% 下降到 1.5%，如圖 4-8 所示。同樣地，可由表 4-16 及 4-17 中看出，求解問題 4 及問題 5 時，依序執行 $RS+HC+MHE$ 及 $MCC+MHE$ 二種演算法均能逐漸縮小其與最佳解的目標函數值之差距。由這些數據顯示，使用本研究之二種演算法 ($RS+HC+MHE$ 及 $MCC+MHE$) 求解此五個問題，皆可有效地降低與最佳解的目標函式值差距，使其求出的解更近似最佳解。

另外，由表 4-13~4-17 顯示，當例子越大時，使用 $RS+HC+MHE$ 及 $MCC+MHE$ 二種演算法與 CPLEX 所求出之目標函式值差距由問題 1 的 0.798% ($RS+HC+MHE$) 及 0.44% ($MCC+MHE$)、問題 2 的 1.1% ($RS+HC+MHE$) 及 1.22% ($MCC+MHE$)、問題 3 的 1.9% ($RS+HC+MHE$) 及 1.5% ($MCC+MHE$)、問題 4 的 1.9% ($RS+HC+MHE$) 及 2.3% ($MCC+MHE$)、增大到問題 5 的 2.25%

($RS+HC+MHE$) 及 2.2% ($MCC+MHE$)，如圖 4-11 所示。也就是當例子越大時，使用 $RS+HC+MHE$ 及 $MCC+MHE$ 二種演算法求出的最佳解的品質似乎有變差的趨勢；但以時間而言，由圖 4-12 中可看出當例子越大時，使用 CPLEX 求解所需時間會更大幅度地增加；反之，使用 $RS+HC+MHE$ 及 $MCC+MHE$ 二種演算法求解所需時間僅小幅度地增加，這個變動趨勢可由圖 4-12 看出來。就 $RS+HC+MHE$ 及 $MCC+MHE$ 這兩個演算法的相對求解效率而言，可從圖 4-11 及 4-12 中看出。在圖 4-11 中， $RS+HC+MHE$ 及 $MCC+MHE$ 這兩個演算法與 CPLEX 最佳解之差距互有勝負，無法判定那一個演算法與 CPLEX 最佳解之差距絕對比較小。另外，以求解時間而言，從圖 4-12 中顯示， $RS+HC+MHE$ 及 $MCC+MHE$ 二者之求解時間也相互差異不大。因此 $RS+HC+MHE$ 及 $MCC+MHE$ 二種演算法在求解本研究之資源回收網路問題時，其效率與求解品質均十分類似，互有高下。

由這些測試結果顯示，使用 $RS+HC+MHE$ 及 $MCC+MHE$ 二種演算法在求解本研究之資源回收網路問題時，均可以有效地降低求解時間；且平均而言，其所求出的解之品質亦不錯。雖然當例子越大時，使用 $RS+HC+MHE$ 及 $MCC+MHE$ 二種演算法所求出之最佳解的品質會變較差，但其求解時間卻遠低於 CPLEX 所需之求解時間。另外，使用本研究之 $RS+HC+MHE$ 及 $MCC+MHE$ 兩種演算法亦比 Jayaraman et al. (2003) 所提出的 $RS+HC+HE$ 及 $CC+HE$ 兩種演算法花更少的時間而得到品質相近的解。因此，本研究所提出的啟發式演算法 $RS+HC+MHE$ 及 $MCC+MHE$ 應可在較短的時間內求解出品質不錯的近似最佳解；即使遇到較大的例題，其求解品質與效率亦不錯。

表 4-5 使用 CPLEX 與 Jayaraman et al. (2003) 演算法求解問題 1 所花費的時間

問題 1	CPLEX time	Heuristics time (sec)						
		RS	HC	HE	RS+HC+HE	CC	HE	CC+HE
回收點：40 回收廠：20 (h1：3) 處理廠：25 (h2：5) 最後處理廠：20								
1	187.156	49.078	1.453	815.703	866.234	5.594	811.875	817.469
2	239.453	47.031	0.5	348.796	396.327	7.672	589.578	597.25
3	411.921	47.5	1.609	1310.09	1359.199	5.016	1137.31	1142.326
4	417.453	47.281	0.968	274.203	322.452	10.359	710.64	720.999
5	263.422	48.563	3.156	769.406	821.125	14.625	933.343	947.968
6	171.578	51.125	0.593	817.546	869.624	4.265	886.75	891.015
7	294.687	51.812	1.578	680.093	733.483	3.797	612.265	616.062
8	445.156	50.796	1.5	766.421	818.717	5.953	853.203	859.156
9	362.656	44.843	2.281	996.312	1043.436	4.062	761.5	765.562
10	243.375	51.546	0.547	735.015	787.108	4.547	960.078	964.625
最大值	445.156	51.812	3.156	1310.09	1359.199	14.625	1137.31	1142.326
最小值	187.156	44.843	0.5	274.203	322.452	3.797	589.578	597.25
平均值	303.6857	48.9575	1.4185	751.3585	801.7345	6.589	825.6542	832.2432

(資料來源：本研究自行整理)

表4-6 比較CPLEX、本研究及Jayaraman et al. (2003) 之演算法求解問題2'的時間

問題 2'	CPLEX time	Heuristics time (sec)			
		<i>CC+HE</i>	<i>RS+HC+HE</i>	<i>MCC+MHE</i>	<i>RS+HC+MHE</i>
回收點：60 回收廠：30 （h1：4） 處理廠：35 （h2：6） 最後處理廠：20					
1	9494.2	5056.753	4045.966	1282.532	1000.593
2	2327.36	4538.067	2962.657	382.203	645.03
3	2429.66	3536.51	4281.858	975.702	999.874
4	2272.26	3862.597	3991.309	415.906	752.688
5	9058.3	4503.77	2751.312	329.874	685.813
6	9839.11	3725.811	5521.261	469.797	1334.143
7	4122.13	2567.377	2941.563	245.14	484.234
8	1582.03	2957.683	2915.505	270.375	692.281
9	3836.14	3052.985	3416.675	316.656	785.578
10	16953.1	3362.674	3423.183	637.577	1413.534
平均值	6191.429	3716.423	3625.129	532.5762	879.3768

(資料來源：本研究自行整理)

表4-7 比較使用Jayaraman et al. (2003) 與本研究演算法求解問題2'之目標函式值差距

問題 2'	Optimality Gap =(Optimal obj - Heuristic obj) / Optimal obj			
	<i>CC+HE</i>	<i>RS+HC+HE</i>	<i>MCC+MHE</i>	<i>RS+HC+MHE</i>
回收點：60 回收廠：30 (h1：4) 處理廠：35 (h2：6) 最後處理廠：20				
1	0.001856	0.00053	0.006435	0.00691
2	0.013893	0.01061	0.013338	0.007793
3	0.000907	0.002497	0.012093	0.009801
4	0.004421	0	0.006453	0.007135
5	0.001454	0.003931	0.003843	0.001454
6	0.004951	0.011356	0.011452	0.01147
7	0.002334	0.002715	0.000553	0.006265
8	0.010665	0.009756	0.010841	0.022552
9	0.006994	0.003179	0.008884	0.016564
10	0.000681	0.007068	0.01163	0.003559
平均值	0.004816	0.005164	0.008552	0.00935

(資料來源：本研究自行整理)

表 4-8 使用 CPLEX 與本研究演算法求解問題 1 所花費的時間

問題 1	CPLEX time		Heuristics time (sec)				
	RS	HC	MHE	Total time	MCC	MHE	Total time
回收點：40 回收廠：20 (h1：3) 處理廠：25 (h2：5) 最後處理廠：20							
1	187.156	48.718	1.922	68.688	119.328	5.046	40.656
2	239.453	46.828	0.656	26.156	73.64	12.063	74.938
3	411.921	47.203	1.609	42.844	91.656	8.828	64.859
4	417.453	46.953	0.953	71.125	119.031	3.922	55.172
5	263.422	48.25	3.125	75.719	127.094	11.438	81.313
6	171.578	67.343	0.687	55.657	123.687	15.531	74.844
7	294.687	51.343	1.422	37.953	90.718	2.5	37.688
8	445.156	50.406	0.484	50.656	101.546	14.078	110.234
9	362.656	44.734	2.265	56.422	103.421	5.953	52.39
10	243.375	51.109	0.563	66.641	118.313	13.937	51.453
最大值	445.156	67.343	3.125	75.719	127.094	15.531	110.234
最小值	171.578	44.734	0.484	26.156	73.64	2.5	37.688
平均值	303.6857	50.2887	1.3686	55.1861	106.8434	9.3296	64.3547

(資料來源：本研究自行整理)

表 4-9 使用 CPLEX 與本研究演算法求解問題 2 所花費的時間

問題 2	CPLEX time		Heuristics time (sec)				
	RS	HC	MHE	Total time	MCC	MHE	Total time
回收點：60 回收廠：30 (h1：4) 處理廠：35 (h2：6) 最後處理廠：20							
1	10876.8	72.735	8.015	1367.5	1448.25	22.656	1140.83
2	1275.13	81.469	0.968	205.328	287.765	4.875	80.515
3	1445.23	103.015	1.375	122.078	226.468	10.141	142.031
4	3382.02	69.375	1.765	356.875	428.015	22.906	334.171
5	2081.45	72.421	0.641	377.656	450.718	7.875	719.515
6	18311.5	69.187	7.25	1795.06	1871.497	6.672	214.016
7	2250.45	69.125	1.782	530.39	601.297	9.219	346.359
8	6709.16	70.234	2.14	468.984	541.358	15.094	608.406
9	21849	82.75	5.453	1051.14	1139.343	36.078	1091.83
10	11921.9	93.625	4.922	1733.84	1832.387	26.157	423.266
最大值	21849	103.015	8.015	1795.06	1871.497	36.078	1140.83
最小值	1275.13	69.125	0.641	122.078	226.468	4.875	80.515
平均值	8010.264	78.3936	3.4311	800.8851	882.7098	16.1673	510.0939

(資料來源：本研究自行整理)

表 4-10 使用 CPLEX 與本研究演算法求解問題 3 所花費的時間

問題 3	CPLEX time			Heuristics time (sec)			
	RS	HC	MHE	Total time	MCC	MHE	Total time
回收點：70 回收廠：40 (h1：5) 處理廠：35 (h2：7) 最後處理廠：20							
1	13173.7	151.203	20.266	2671.98	2843.449	76.625	1591.61
2	17661.4	144.156	5.985	3469.47	3619.611	24.64	1986.02
3	2553.77	180.296	2.609	1354.83	1537.735	127.687	732.73
4	22224.5	160.812	3.062	1726.63	1890.504	11.75	1373.67
5	32177.9	142.171	4	2322.44	2468.611	23.703	2553.23
6	33979.8	169.625	5.218	3898.25	4073.093	78.937	2602.63
7	5215.84	146.843	14.531	1894.59	2055.964	32.281	1183.36
8	14092.4	152.531	15.125	2748.63	2916.286	36.859	3574.63
9	8081.39	407.828	11.641	2167.11	2586.579	39.61	980.766
10	5836.16	160.031	12.031	1774.75	1946.812	30.531	1674.64
最大值	33979.8	407.828	20.266	3898.25	4073.093	127.687	3574.63
最小值	2553.77	142.171	2.609	1354.83	1537.735	11.75	732.73
平均值	15499.69	181.5496	9.4468	2402.868	2593.864	48.2623	1825.329

(資料來源：本研究自行整理)

表 4-11 使用 CPLEX 與本研究演算法求解問題 4 所花費的時間

問題 4	CPLEX time	Heuristics time (sec)						
		RS	HC	MHE	Total time	MCC	MHE	Total time
回收點：100 回收廠 50 (h1：6) 處理廠：55 (h2：8) 最後處理廠：20								
1	62384.2	190.031	1.36	1749.59	1940.981	6.344	3362.28	3368.624
2	5906.66	151.281	0.969	529.344	681.594	6.156	516.297	522.453
3	22052.5	152.171	0.453	826.656	979.28	2.266	541.281	543.547
4	1763.55	152.687	1.406	342.406	496.499	7.922	993.531	1001.453
5	23749.8	148.64	0.609	481.031	630.28	2.734	464.375	467.109
6	9177.55	153.843	1.531	703.031	858.405	3.922	480.828	484.75
7	5977.39	157.64	0.75	596.813	755.203	3.422	618.484	621.906
8	8101.23	153.125	0.359	906.422	1059.906	3.719	444.344	448.063
9	8116.11	153.421	1.844	611.984	767.249	9.156	525.75	534.906
10	9682.48	158.484	0.703	537.172	696.359	5.969	540	545.969
最大值	62384.2	190.031	1.844	1749.59	1941.465	9.156	3362.28	3371.436
最小值	1763.55	148.64	0.359	342.406	491.405	2.266	444.344	446.61
平均值	15691.15	157.1323	0.9984	728.4449	886.5756	5.161	848.717	853.878

(資料來源：本研究自行整理)

表 4-12 使用 CPLEX 與本研究演算法求解問題 5 所花費的時間

問題 5	CPLEX time		Heuristics time (sec)				
	RS	HC	MHE	Total time	MCC	MHE	Total time
回收點：12 0 回收廠 60 (h1：6) 處理廠：65 (h2：8) 最後處理廠：20							
1	6170.39	191.64	0.672	574.297	4.765	725.75	730.515
2	30649.4	195.515	1.672	682.906	14.672	1389.88	1404.552
3	5119.39	205.953	1.438	739.5	3.469	601.265	604.734
4	19347.7	213.594	1.031	1509.72	40.094	3445.17	3485.264
5	16952.03	225.031	2.094	2785.75	42.641	3564.84	3607.481
6	78248.1	217.593	2.453	2065.92	2.922	1545.7	1548.622
7	15088.5	210.562	0.719	976.64	8.641	1092.7	1101.341
8	9385.34	192.593	1.062	622.485	4.704	925.484	930.188
9	18925	193.5	0.734	718.312	3.297	1354.3	1357.597
10	4165.72	199.656	0.844	1176.59	3.985	862.062	866.047
最大值	78248.1	225.031	2.453	2785.75	42.641	3564.84	3607.481
最小值	4165.72	191.64	0.672	574.297	2.922	601.265	604.734
平均值	20405.16	204.5637	1.2719	1185.212	12.919	1550.715	1563.634

(資料來源：本研究自行整理)

表4-13 使用CPLEX與本研究演算法求解問題1之目標函式值差距

問題 1					
Optimality Gap =(Optimal obj - Heuristic obj) / Optimal obj					
	RS	HC	MHE	MCC	MHE
回收點：40 回收廠：20 (h1：3) 處理廠：25 (h2：5) 最後處理廠：20					
1	0.021642	0.018032	0.00984504	0	0
2	0.0453	0.027707	0.007309389	0.032046	0.005105336
3	0.049525	0.028375	0.002210752	0.012545	0.00167944
4	0.028106	0.012359	0.000642815	0.01261	0.000809654
5	0.027104	0.017195	0.009534764	0.02421	0.009486495
6	0.047636	0.017999	0.008886852	0.019032	0.006648969
7	0.047302	0.023231	0.00447714	0.011929	0.002067375
8	0.031224	0.011617	0.003903946	0.008268	0.008046955
9	0.12164	0.040149	0.027968515	0.016634	0.010152389
10	0.044051	0.015071	0.005032475	0.011466	0
最大值	0.12164	0.040149	0.027968515	0.032046	0.010152389
最小值	0.021642	0.011617	0.000642815	0	0
平均值	0.046353	0.021173	0.007981169	0.014874	0.004399661

(資料來源：本研究自行整理)

表4-14 使用CPLEX與本研究演算法求解問題2之目標函式值差距

問題 2	Optimality Gap =(Optimal obj - Heuristic obj) / Optimal obj			
	RS	HC	MHE	MCC
回收點：60 回收廠：30 (h1：4) 處理廠：35 (h2：6) 最後處理廠：20				
1	0.047373	0.029771	0.007627968	0.031755
2	0.10471	0.03321	0.009512473	0.018962
3	0.111562	0.062385	0.028584129	0.066481
4	0.03102	0.021133	0.006293758	0.014686
5	0.046995	0.041959	0.01880723	0.046757
6	0.053336	0.028013	0.005224897	0.031659
7	0.049809	0.021565	0.006795983	0.01045
8	0.049726	0.023152	0.00814501	0.046855
9	0.056723	0.022182	0.00732939	0.040316
10	0.109451	0.035171	0.011735638	0.060326
最大值	0.111562	0.062385	0.028584129	0.066481
最小值	0.03102	0.021133	0.005224897	0.01045
平均值	0.066071	0.031854	0.011005648	0.036825

(資料來源：本研究自行整理)

表4-15 使用CPLEX與本研究演算法求解問題3之目標函式值差距

問題 3	Optimality Gap =(Optimal obj - Heuristic obj) / Optimal obj				
	RS	HC	MHE	MCC	MHE
回收點：70 回收廠 40 (h1：5) 處理廠：45 (h2：7) 最後處理廠：20					
1	0.076126	0.041441	0.015810745	0.025115	0.009850312
2	0.108496	0.08997	0.032626172	0.057454	0.020030234
3	0.061107	0.044938	0.016529592	0.025147	0.014374245
4	0.102566	0.06556	0.026305479	0.031375	0.01701135
5	0.071115	0.055564	0.02076723	0.042808	0.027448363
6	0.072259	0.036531	0.018775153	0.063401	0.023039428
7	0.049603	0.035507	0.013790293	0.017265	0.009889635
8	0.034975	0.030643	0.011758646	0.022772	0.009388006
9	0.073926	0.065719	0.0217394	0.030665	0.015223615
10	0.049779	0.033464	0.012194767	0.02131	0.007225282
最大值	0.108496	0.08997	0.032626172	0.063401	0.027448363
最小值	0.034975	0.030643	0.011758646	0.017265	0.007225282
平均值	0.069995	0.049934	0.019029748	0.033731	0.015348047

(資料來源：本研究自行整理)

表4-16 使用CPLEX與本研究演算法求解問題4之目標函式值差距

問題 4	Optimality Gap =(Optimal obj - Heuristic obj) / Optimal obj				
	RS	HC	MHE	MCC	MHE
1	回收點：100 回收廠 50 (h1：6) 處理廠：55 (h2：8) 最後處理廠：20				
2	0.138515	0.070904	0.020057037	0.072042	0.0260387
3	0.133655	0.030747	0.013758612	0.051186	0.017435164
4	0.106692	0.031841	0.00849062	0.056018	0.014464799
5	0.124987	0.026779	0.009714187	0.062824	0.022083814
6	0.119874	0.041745	0.008072322	0.034607	0.012347569
7	0.179479	0.053421	0.020527243	0.074188	0.035155224
8	0.166492	0.056503	0.024800151	0.083247	0.034654029
9	0.182635	0.079082	0.026950212	0.093778	0.034417626
10	0.17142	0.081579	0.028799963	0.053514	0.015196579
最大值	0.188735	0.089284	0.033267676	0.093778	0.035155224
最小值	0.106692	0.026779	0.008072322	0.034607	0.012347569
平均值	0.151248	0.056189	0.019443802	0.064377	0.023414711

(資料來源：本研究自行整理)

表4-17 使用CPLEX與本研究演算法求解問題5之目標函式值差距

問題 5	Optimality Gap =(Optimal obj - Heuristic obj) / Optimal obj				
	RS	HC	MHE	MCC	MHE
回收點：120 回收廠：60 (h1：6) 處理廠：65 (h2：8) 最後處理廠：20					
1	0.120676	0.034117	0.010386237	0.041212	0.01699431
2	0.167846	0.055593	0.020882195	0.04753	0.02133973
3	0.149045	0.054272	0.024670593	0.045319	0.021168628
4	0.149688	0.068658	0.02525742	0.086573	0.045070742
5	0.122339	0.071662	0.030552589	0.08975	0.03282758
6	0.141365	0.064482	0.026296258	0.045561	0.02801272
7	0.16757	0.058492	0.030508292	0.045188	0.016984536
8	0.110227	0.050857	0.024384954	0.035398	0.014366102
9	0.101502	0.036642	0.014896391	0.033557	0.010373719
10	0.176759	0.048446	0.017361378	0.051029	0.012911493
最大值	0.176759	0.071662	0.030552589	0.08975	0.045070742
最小值	0.101502	0.034117	0.010386237	0.033557	0.010373719
平均值	0.140702	0.054322	0.022519631	0.052112	0.022004956

(資料來源：本研究自行整理)

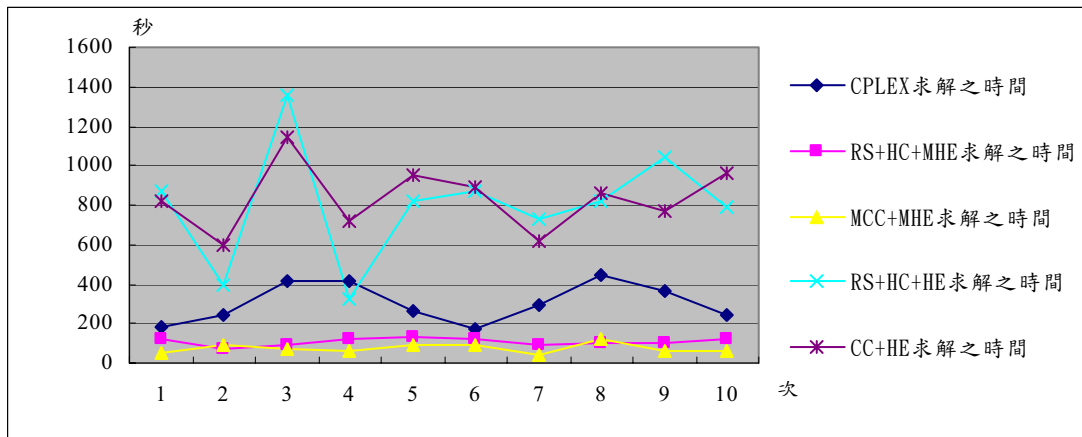


圖4-1 使用CPLEX與Jayaraman et al. (2003) 及本研究演算法求解問題1所需時間之比較

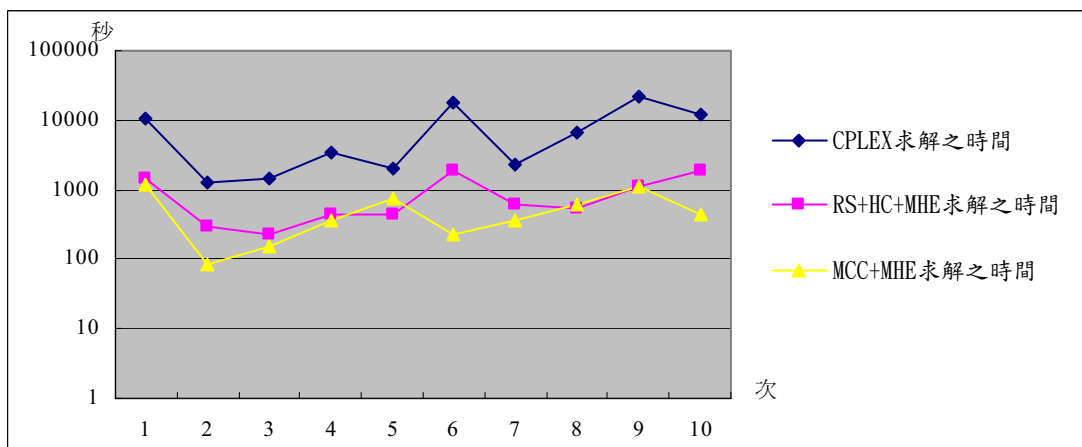


圖4-2 使用CPLEX與本研究演算法求解問題2所需時間之比較

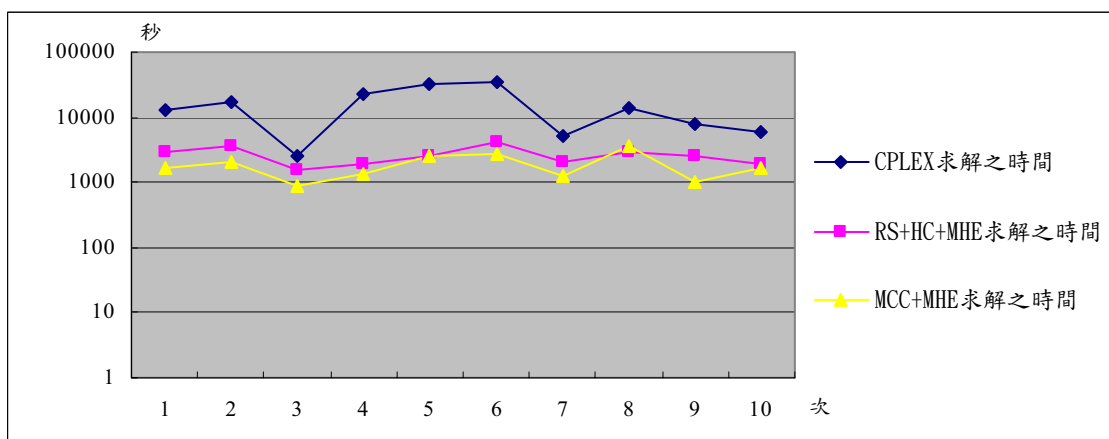


圖4-3 使用CPLEX與本研究演算法求解問題3所需時間之比較

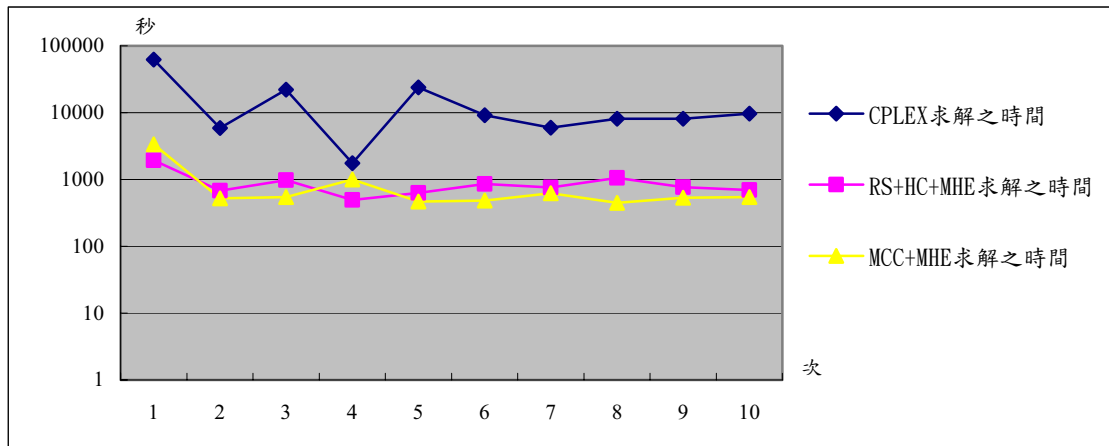


圖4-4 使用CPLEX與本研究演算法求解問題4所需時間之比較

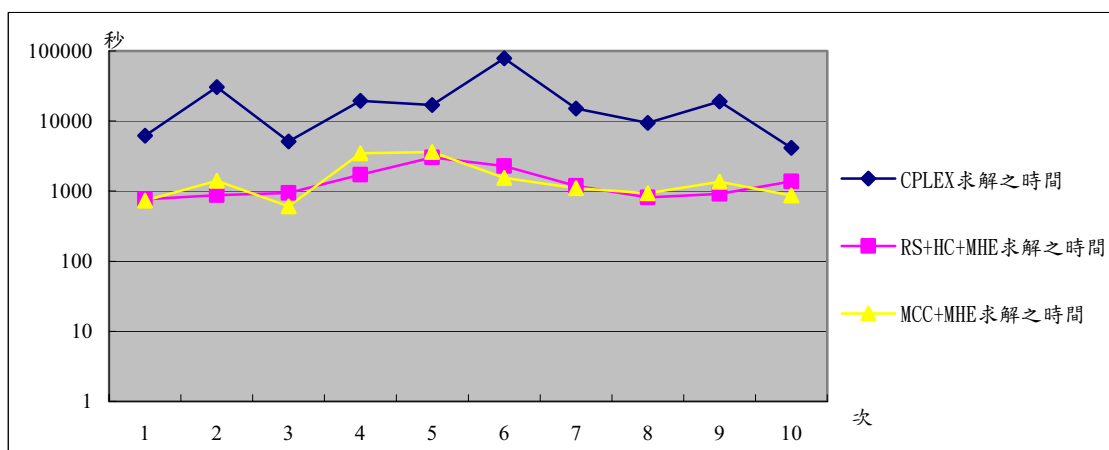


圖4-5 使用CPLEX與本研究演算法求解問題5所需時間之比較

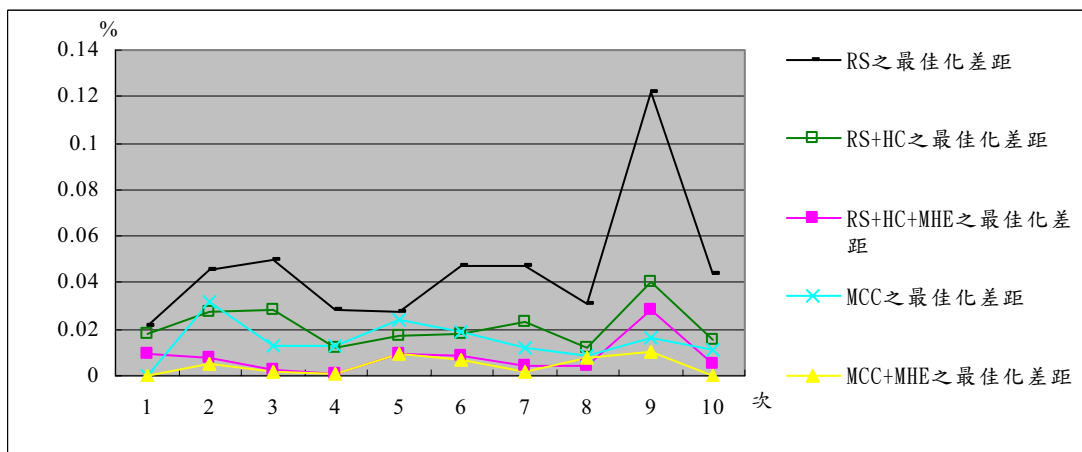


圖4-6 本研究之演算法在解問題1時，與CPLEX最佳目標函式值之最佳化差距

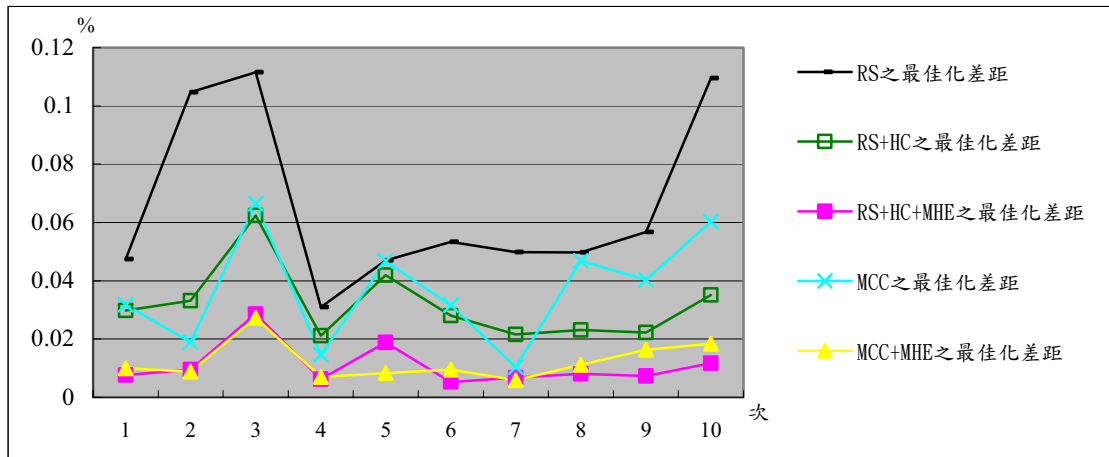


圖4-7 本研究之演算法在解問題2時，與CPLEX最佳目標函式值之最佳化差距

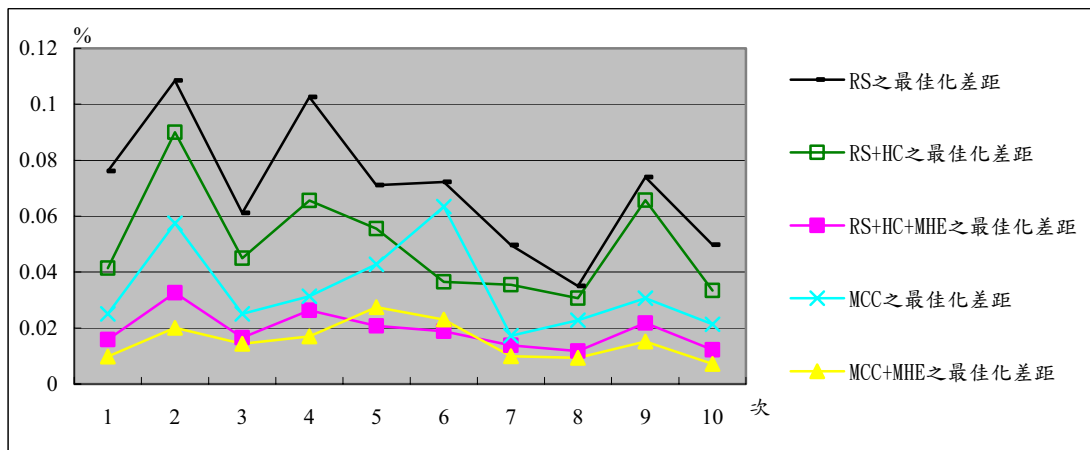


圖4-8 本研究之演算法在解問題3時，與CPLEX最佳目標函式值之最佳化差距

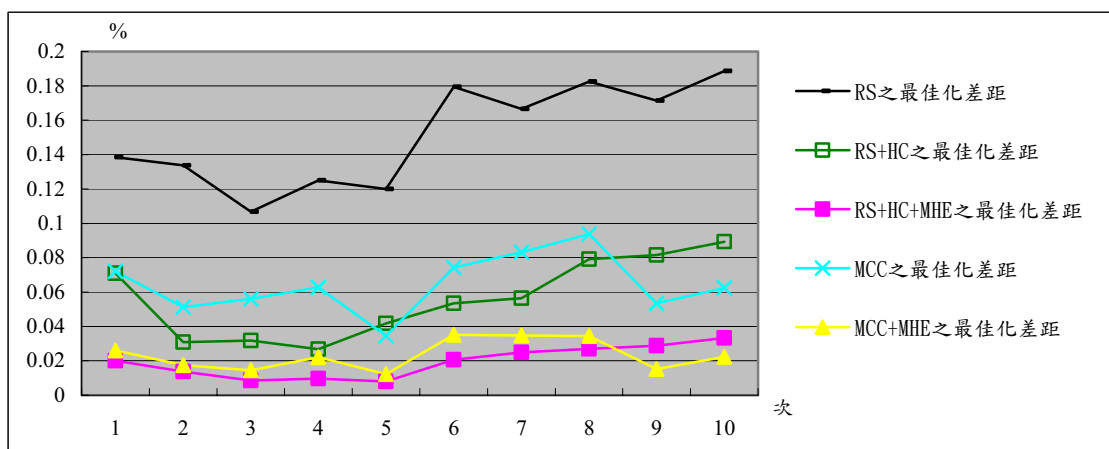


圖4-9 本研究之演算法在解問題4時，與CPLEX最佳目標函式值之最佳化差距

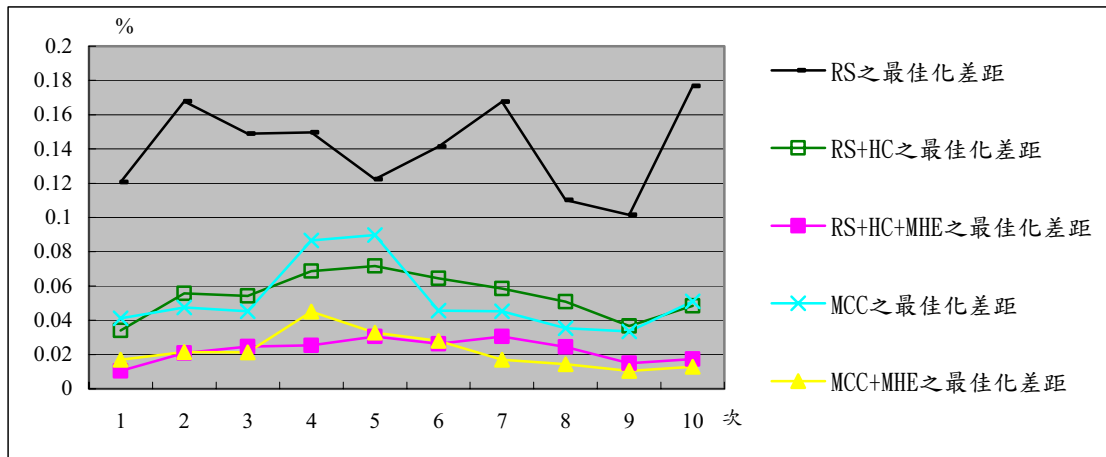


圖4-10 本研究之演算法在解問題5時，與CPLEX最佳目標函式值之最佳化差距

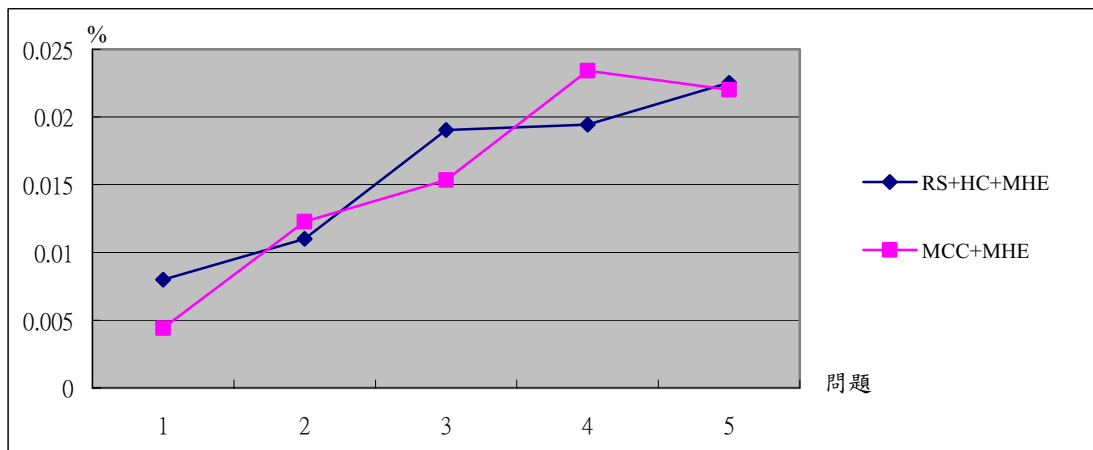


圖4-11 本研究之演算法與CPLEX最佳目標函式值之最佳化差距的趨勢圖

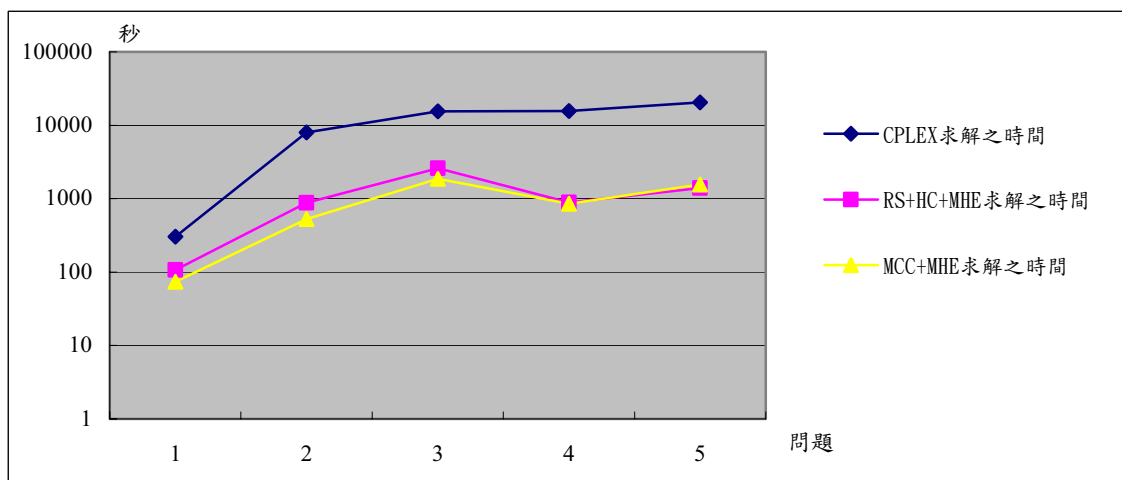


圖4-12 本研究之演算法與CPLEX求解所需時間之趨勢圖

第五章 結論與建議

本研究探討資源回收網路之廠址配置與設施規劃最佳化問題。在數學模式方面，本研究以 Shih (2001) 提出之資源回收網路模式為基礎，進一步容許新建廠房可選擇建構可回收或處理兩類回收物之綜合式設施，抑或建構僅可回收或處理單一類回收物之設施；因此，決策者可以決定新建的廠房之廠址選擇與其設施規劃，進而最大化其回收利潤。

我們提出的數學模式為一個複雜的混整數規劃問題，在問題變大時以 CPLEX 求解最佳解將花費許多時間。因此，本研究採用並延伸 Jayaraman (2003) 所提出的啟發式演算法 ($RS+HC+HE$ 及 $CC+HE$)，提出二種改良式演算法 ($RS+HC+MHE$ 及 $MCC+MHE$)，來加快求解本問題的速度。

本研究依據臺灣的資源回收相關資訊，訂定各類參數並提出數個不同大小的問題來測試數學模型及演算法。測試結果証實使用 $RS+HC+MHE$ 及 $MCC+MHE$ 二種演算法的確可以有效地減少求解時間，且平均而言求出之解的品質與最佳解亦十分接近。

由於本研究之演算法可在較短的時間內求得品質不錯的近似最佳解，因此針對處理某些需要在短時間內得到最佳解估計值的問題而言，本研究之成果應有不錯的應用性。舉例來說，政府機構該訂出何種設施規範或津貼罰鍰細則，才能提高某類廢棄物回收率？廢棄物回收業者在初步規劃其物流系統時，如何與其相關回收或處理商簽訂通路合約，才能獲取最大利益？諸如此類先導系統規劃問題之相關研究，應該都可以使用本研究發展之方法在短時間內求得十分逼近最佳解之近似解，再經由類似敏感度分析等微調參數過程，達到我們想要的目標；或者觀察問題限制式及目標函數之變動情況，幫助我們更進一步了解問題的特性與本質。

本研究目前的成果尚有未臻完備之處，以下將提供數個未來可能的研究方向建議，以供後續研究者參考：

- 一、由於能力及時間之限制下，本研究的廢棄物回收網路，僅討論針對二種不同類型的回收物品相關的設施選擇（可選綜合類、第一類、或第二類），未來可往多類的回收物品來討論。若考慮對於多類回收物品相關設施作選擇，則限制式無疑地將增加許多。以 k 類回收物品而言，總共可能之設施類別就有 $C_0^k + C_1^k + C_2^k + \dots + C_k^k = 2^k$ 種，其中 C_i^k 代表建構可回收或處理 i 類回收物的設施個數。亦即，多考慮一類回收物，其限制式總數將呈指數趨勢增加。
- 二、本研究提出之演算法雖然可有效降低求解的時間，但面對例子越大時，求出的解品質會變差，嚴格來說，此演算法求解效率仍有不少改進空間，未來可往演算法之方向努力，發展出求解效率更佳之演算法。
- 三、由於本研究的演算法可在短時間內求得品質良好之近似最佳解，因此可用來研究先導時期之物流規劃問題，探討影響物流規劃之種種參數及限制條件該如何設定規範，以多次的模擬微調參數，提供管理層面更深層的知識挖掘或驗證。

參考文獻

- 鄭顯榮，環境設施規劃設計與操作管理，高立圖書有限公司，頁 225–228，2004。
- 章裕民，廢棄物處理，第三版，新文京開發出版有限公司，頁 7–8，2003。
- 網址 <http://recycle.epa.gov.tw/>，資源回收管理基金管理委員會。
- Barros, A. I., Dekker, R. and Scholten, V. A two-level network for recycling sand: A case study. *European Journal of Operational Research*, **110**, 199–214, 1998.
- Bloemhof-Ruwaard, J. M., Salomon, M. and Van Wassenhove, L. N. On the coordination of product and by-product flows in two-level distribution networks: Model formulations and solution procedures. *European Journal of Operational Research*, **79**, 325–339, 1994.
- Hu, T. L., Sheu, J. B. and Huang, K. H. A reverse logistics cost minimization model for the treatment of hazardous wastes. *Transportation Research Part E*, **38**, 457–473, 2002.
- Jayaraman, V., Patterson R. A. and Rolland, E. The design of reverse distribution network : Models and solution procedures. *European Journal of Operational Research*, **150**, 128–149, 2003.
- Nagurney, A. and Toyasaki, F. Reverse supply chain management and electronic waste recycling: a multitiered network equilibrium framework for e-cycling . *Transportation Research Part E*, **41**, 1–28, 2005.
- Marin, A. and Pelegrin, B. The return plant location problem: Modeling and resolution . *European Journal of Operational Research*, **104**, 375–392, 1998.
- Min, H., Ko, H. J. and Ko, C. S. A genetic algorithm approach to developing the multi-echelon reverse logistics network for product returns. *Omega*, 2004
Article in press.
- Rosing, K. E., ReVelle, C. S., Rolland, E., Schilling, D. A. and Current, J. R.

- Heuristic concentration and tabu search: A head to head comparison.
European Journal of Operational Research, **104**, 93–99, 1998.
- Rosing, K. E. and ReVelle, C. S. Heuristic concentration: Two stage solution construction. *European Journal of Operational Research*, **97**, 75–86, 1997.
- Shih, L. H. Reverse logistics system planning for recycling electrical appliances and computes in Taiwan. *Resources, conservation and recycling*, **32**, 55-72, 2001.