

含凹形節線成本最小成本轉運問題鄰近 搜尋法之研究¹

NEIGHBORHOOD SEARCH ALGORITHMS FOR CONCAVE COST TRANSSHIPMENT PROBLEMS

顏上堯 Shangyao Yan²

陳建榮 Chien-Rong Chen³

湯慶輝 Chin-Hui Tang⁴

(91 年 9 月 9 日收稿，92 年 3 月 5 日修改，93 年 4 月 1 日定稿)

摘 要

傳統在最小成本轉運問題的定式上，常以線性方式來定義運送成本，藉以簡化問題的複雜度。在實務上，貨物運送的單位成本常隨數量的增加而遞減，成本函數曲線為凹形。以往有不少含凹形成本節線之研究，但侷限於不同之特殊網路且方法屬傳統區域搜尋法或傳統啟發解法，近期雖有學者開始以新近鄰近搜尋法求解簡化的運輸問題，以達到較大範圍的搜尋方式，期能找到較優於傳統啟發解法的解，卻忽略運輸網路常見的轉運問題。緣此，本研究針對含凹形節線成本一般性最小成本網路流動問題，參考新近鄰近搜尋法，如門檻值接受法與大洪水法，發展有效率的鄰近搜尋法，以求解問題。在求解的方法上，

-
1. 本研究承行政院國家科學委員會專題研究補助 (NSC90-2416-H-366-009)，特此致謝。作者亦感謝二位審查委員寶貴的修正意見，使本文更為完整與嚴謹。
 2. 國立中央大學土木工程學系教授 (聯絡地址：320 中壢市五權里中大路 300 號中央大學土木工程學系；電話：03-4227151 轉 4141；E-mail：t320002@cc.ncu.edu.tw)。
 3. 國立中央大學土木工程學系碩士。
 4. 國立中央大學土木工程學系博士生。

本研究首先設計適合凹形成本網路流動特性之初始解產生方法，快速得到一組解，再使用新近鄰近搜尋法的改善機制以達到改善效果，以找到近似最佳解。為測試本研究演算法在不同規模及參數的網路問題的求解績效，本研究設計一隨機網路產生器，產生大量的隨機網路，並以 C++ 語言撰寫所有相關的電腦程式，在個人電腦上測試分析。測試結果顯示，本研究採行的方式求解品質良好。

關鍵詞：鄰近搜尋；凹形節線成本；轉運問題；門檻值接受法；大洪水法

ABSTRACT

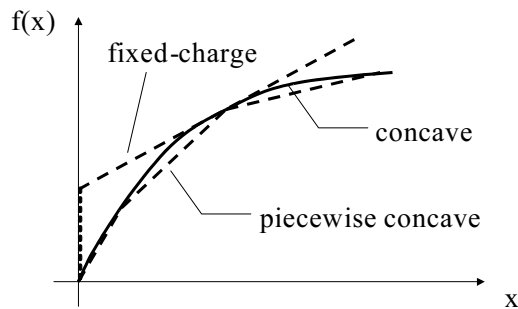
The minimum cost transshipment problems are traditionally defined as a linear cost problem, to reduce problem complexity. In reality, the unit cost decreases as the amount transported increases, resulting in a concave cost function. Great efforts have been devoted to the development of solution algorithms. However, they were confined to special transportation networks. Besides, their methods were focused on local search algorithms or traditional heuristics. Recently, researchers began to use advanced neighborhood search algorithms to solve concave cost bi-partite transportation network problems to enlarge search area and find near-optimal solutions. This type of research, however, neglected flow transfers in transportation networks. We developed two neighborhood search algorithms referring to the threshold accepting algorithm and the great deluge algorithm to efficiently solve transshipment problems. Problem characteristics were first explored to efficiently generate initial solutions, which are then improved by neighborhood search algorithms to near-optimal solutions. To evaluate the proposed neighborhood search algorithms, we designed a randomized network generator to produce many test problems. We employed C++ computer language to code all necessary programs and perform tests on personal computers. The results show that the developed neighborhood search algorithms performed well in the tests.

Key Words: *Neighborhood search; Concave arc cost; Transshipment problem; Threshold accepting algorithm; Great deluge algorithm*

一、緒 論

許多運輸實際問題可使用網路的方式表示並加以求解，例如最短路徑問題、運輸問題（或稱為二分割運輸網路問題）及考慮轉運點的轉運問題。其中，轉運問題為前二者之一般性運送問題。轉運問題的數學定義一般稱為最小成本網路流動問題，以往學界或實務界在最小成本網路流動問題的定式上，常將各節線之單位運送成本予以簡化為固定值，即以線性成本來定式運送成本，其主要原因在於簡化後的線性成本較易於求解。然而，此假設常與許多實際貨物之運送成本不相符合，實際上單位運送成本常隨著運送貨物的增加而減

少，也就是經濟學上常見之規模經濟特性。實務上，貨物的運送成本主要受到物品種類不同與種類多寡、運送方式、運送距離、以及運送數量的影響，而運送方式中包含了裝卸與運送兩個部分。一般而言，高固定成本或低變動成本的運送方式，例如鐵路與水運，通常貨物運送的每單位成本隨貨物量的增加而減少。基本上，由於實務的規模經濟特性，運送成本函數常可定式為凹形成本（如 Yan 與 Luo^[1,2]）。一般凹形成本網路流動問題中，有以下幾種常見的成本類型，如凹形成本（concave）、分段凹形成本（piecewise concave），或是固定費率用的凹形成本（fixed charge cost），如圖 1 所示。



資料來源：Yan 與 Luo^[2]。

圖 1 凹形成本函數示意圖

最小成本網路流動問題為傳統之網路規劃問題，此等問題在多年來已有多種適合之求解方式（Ahuja 等人^[3]）。而含凹形節線成本之最小成本網路流動問題或轉運問題，其意義為「將人或貨物由起點運送至終點，其中可經過數個轉運點，並求總成本和之最小化」，每個起點或終點都可為轉運點。在網路流動規劃問題中則屬於 NP-hard 問題（Garey 與 Johnson^[4]），難以在有限的時間內求得大型問題的最佳解。

傳統上求解含凹形節線成本問題，可分為兩大方向：1.直接將問題形式簡化為線性問題，以線性規劃的方式求解；2.將非線性問題分成數個子問題，以線性近似法求解，如分段線性（piecewise linear）的求解模式，或是用整數規劃（integer programming）的方式求解。此透過簡化後的線性成本，常難以反映實際成本，而難求得近似最佳解。至於以線性近似法求解凹形成本的轉運問題，一旦問題規模變大，則傳統整數規劃模式則將由於問題複雜度的增加而難以求解（Yan 與 Luo^[2]）。

近來 Yan 與 Luo^[1,2] 利用新近組合最佳化的方法，如禁制搜尋法（tabu search, TS）、門檻值接受法（threshold accepting, TA）、模擬退火法（simulated annealing, SA）等區域搜尋法，針對單純之二分割運輸網路問題，發展求解演算法，且經過測試及比較傳統啟發解後，發現效果頗佳。然而，此二分割運輸問題，未考量網路的轉運問題，在求解的設計上，遠較轉運問題為簡單，難以應用到一般性的轉運問題。

本研究的目的是針對含凹形節線成本的最小成本網路流動問題或轉運問題，探討此問題的特性，並據以發展有效率的鄰近搜尋法。為評估此鄰近搜尋法的求解績效，本研究藉由電腦快速的運算能力，發展測試的網路產生器，測試不同規模大小的網路，期以為實務界求解此類轉運問題的參考。

由於本研究問題在數學上可歸類為 NP-hard 問題 (Larsson 等人^[5])，故本研究發展問題導向之啟發式解法以求解。本研究首先從數學上探討含凹形節線成本轉運問題的特性，從理論觀點引用組合最佳化之解法技巧，並運用部分純網路問題解法的技巧，發展合適的鄰近搜尋法。在求解方法上，本研究首先考慮凹形節線成本的特性，設計適合凹形成本特性之初始解，再參考新近鄰近搜尋法中，門檻值接受法 (threshold accepting)、大洪水法 (great deluge algorithm) 與禁制搜尋法 (tabu search)，發展有效的鄰近搜尋法。本文其餘架構如下，第二節進行文獻回顧，第三節介紹含凹形節線成本之轉運問題的數學式，第四節設計求解演算法，第五節進行實驗測試，第六節提出結論與建議。

二、文獻回顧

本節將針對以往凹形成本運送問題之求解方法，以及求解組合最佳化問題所發展的數種啟發式解法做一回顧。之後則對本研究使用的三種鄰近搜尋觀念的啟發式解法做個別介紹。

2.1 凹形成本運送問題

凹形成本運送問題之問題形式如第 1 節中所述，例如鐵路與水運，通常貨物運送的每單位成本隨貨物量的增加而減少，亦即經濟學上之規模經濟特性 (Yan 與 Luo^[1,2])。另外，相對於凹形成本之問題形式，在都市運輸網路常見有凸形成本運輸問題，例如 (Sheffi^[6]) 所述之路段時間，因流量增加時間呈凸形狀成本快速增加，為一種凸形成本運輸問題，與本研究探討的凹形成本運輸問題不同。凸形成本運輸問題在數學上一般屬凸形問題 (convex program)，在求解上較凹形成本運輸問題 (為 non-convex 問題) 為簡單。

早期求解最小化凹形成本網路流動問題，多從理論方面探討。例如 Zangwill^[7] 曾對此問題找出端點解 (extreme point) 的特性，在多元貨品 (multi-commodity) 簡化為單一貨品類型的情況下，建立單一起點至單一迄點的求解模式，並應用在生產和存貨模式上。其運用動態規劃法，搜尋所有的端點解以便尋找最佳解；然而當迄點擴大到 D 個，則每個節點的遞迴次數將擴大到 $2^D - 1$ 。換言之，運算時間將大幅成長。Zangwill 並提及求解凹形成本網路問題最大的困難在於此種問題具有多個區域最佳解，一般搜尋方式無法有效的求得全域最佳解，而必須透過列舉所有區域最佳解的方式，致求解費時，故實用有限。

Gallo 與 Sandi^[8] 以網路圖形鄰接的理論特性，從凹形成本網路最佳化的觀點，由網路上給定的端點解，列舉所有鄰近的端點解 (adjacent extreme flow)，再利用最佳解落在可行

解端點的特性，將原問題簡化後以窮舉方式求解；並指出隨著網路規模擴大，鄰近點之總數將呈指數規模成長。Rech 與 Barton^[9]以運送單一貨物種類從 m 個起點到 n 個終點的非凸性分段線性網路 (non-convex piecewise linear network)，運用分枝界限法求得下解；其方法介於嚴格最佳化和啟發式解法之間。Gallo 等人^[10]以單一起點至多個迄點的最小成本網路流動問題，利用分枝界限法找尋全域最佳解 (global optimum)，同時利用線性鬆弛原問題的限制式以求得下解。Gallo 等人並認為由於成本函數的凹性 (concavity) 與網路複雜度影響，故對欲找到最佳解變得很困難，因此一般解法多採用啟發式解法來搜尋，以改善區域最佳解。

對於凹形節線成本轉運問題，亦有學者從不同觀點進行研究。Jordan^[11]探討區域分布中心 (local distribution center) 和服務網路關連，對於最小化凹形運輸成本的大型網路，採用啟發式解法，計算平均成本並以線性規劃 (linear programming) 來反覆求解。另一方面，Blumenfeld 等人^[12]從生產、運輸、及存貨三方面探討貨物網路成本，而 Hall^[13]探討貨物路線經過聯合轉運站或直接運達目的地的路線選擇問題。此等文獻均將原貨運網路分成數個較小的子網路，每個網路包括一個起點和迄點，列舉所有可行路線組合來找子網路最佳解，再反覆運算所有貨物量組合。Thach^[14]則對有方向性網路之最佳化凹形成本流動問題，從理論觀點目標函數分解成主問題與子問題兩部分，並以價格機制分別求解之。Guisewite 與 Pardalos^[15]針對單一節線為凹形成本，其餘為線性成本，且為單一起點至多個迄點的節線無上限網路，發展一多項式次方時間演算法 (polynomial time algorithm)，將候選解集合從 2^k 降為 $k + 1$ ，以有效求解。

Yaged^[16]曾經針對大型通信網路 (約 100 ~ 200 個節點、200 ~ 300 條節線) 的凹形成本傳輸節線，發展區域最佳解的啟發式解法。Larsson 等人^[5]曾以拉氏鬆弛法 (Lagrangian relaxation) 求解節線有流量限制的最小化凹形成本網路流動問題，提供一下限解，再利用次梯度法 (subgradient method) 修正拉氏乘數並反覆運算。Larsson 等人^[5]並提到要確定這種方法可以找到最佳解，最好伴隨著使用分枝界限法來對照結果。Balakrishnan 和 Graves^[17]也曾使用了相同觀念之拉氏鬆弛法求解 fixed-charge cost 之問題，而 Amiri 和 Pirkul^[18]則延伸改良前一篇之模式，在拉氏鬆弛後之模式中，增加一個階段性變動節線流量的變數，藉此將模式分為兩個子問題並使用次梯度法求解，可較單一模式更快達到收斂。Nourie 與 Guder^[19]曾採用啟發式解法，對於分段線性 (piecewise linear) 之凹形成本運輸問題，將成本分成生產成本與運送成本兩部分，再以限制進入基底 (restricted-entry basis) 的方式改良單體法 (simplex method)，並反覆求解線性問題以找出區域最佳解。

傳統求解分段線性凹形成本 (piecewise linear concave cost) 網路問題時，常依照各節線上之分段 (不同成本函數) 分解為多條節線，但此分解法將使得網路規模擴大，增加求解時間。Dukwon 與 Panos^[20]根據此觀念提出了動態斜率排列方式，並增加一信賴區間之判斷，雖然分解節線使得網路規模擴大，但此求解方式可較以往分段方式的做法更快得收斂。Suwan 等人^[21]曾就整包交換網路 (packet-switched networks) 之節線容量指派問題 (link capacity assignment problem)，針對分段線性凹形成本函數求解。其方法係將不可微分

之函數在轉折點附近以弧線趨近形成可微分函數，並對拉氏乘數作收斂判斷，在達收斂時即可結束。

另外，凹形成本具有節線流量傾向最大流量與最小流量兩極化之特性，Kuhn 與 Baumol^[22]曾提出貪婪法 (greedy method) 等的數種啟發式解法，其求解策略為：指派最大流量至成本最小的節線，在 Yan 與 Luo^[2]針對二分割運輸網路問題中，也使用此觀念發展初始解，各起迄點的供給量與需求量傾向以全有或全無指派 (all or nothing)。

2.2 新近組合最佳化啟發式解法

傳統鄰近搜尋方法為區域搜尋法 (local search)，其經常面臨到陷入「局部最佳解 (local optimum)」的困境，以致無法提供更好的近似解。為克服其缺點，1980 年代開始，新的啟發式解題概念逐漸形成，並造成一股新興的研究風潮。此等新近啟發式方法的主要求解策略均是在可行解區中一步步移動以逐步改善目標函數值，並使演算法具有從局部最佳解脫離的能力，同時又能向最佳解逼近。

自從 Glover 在其文章揭示了「巨集啟發式解法 (meta-heuristics)」的精神後，「巨集啟發式方法」一詞獲得多數學者的認同而延用至今 (Glover 與 Laguna^[23])。所謂巨集啟發式解法乃是以傳統啟發式方法為核心架構，結合高階的搜尋策略 (meta-strategies)，以指導其跳脫局部最佳解的束縛而找到更好的解 (Osman 與 Kelly^[24])。上述的高階搜尋策略大多取材自物理化學、生物自然、人類思考之行為或邏輯。諸如，禁制搜尋法 (tabu search, TS) (Glover^[25,26])、模擬退火法 (simulated annealing, SA) (Kirkpatrick 等人^[27])、門檻值接受法 (threshold accepting, TA) (Dueck 與 Scheuer^[28])、遺傳演算法 (genetic algorithm, GA) (Goldberg^[29])、大洪水法 (great deluge algorithm, GDA) (Dueck^[30])、記錄更新法 (record-to-record travel, RRT) (Dueck^[30])、噪音擾動法 (noising method, NM, 如可參考 Charon 與 Hurdy^[31]或韓復華與卓裕仁^[32])、兩極跳躍法 (flip-flop method, FF, 如可參考韓復華等人^[33,34])、搜尋空間平滑法 (search space smoothing, SSS, 如可參考 Gu 與 Huang^[35]或韓復華與卓裕仁^[32])、類神經網路 (neural network, NN, 如可參考 Reeves^[36])等。此等新近演算法，相較以往的啟發式演算法，可有效的跳脫局部最佳解，使得搜尋範圍更大，因而提高求得最佳解 (或較佳解) 的可能性，因此在不少的應用中，獲得了不錯的成果。近來此等方法也衍生出更多的變形或組合架構，如 Booker^[37]。為了使演算法具有從局部最佳解脫離的能力，同時又能向最佳解逼近，各種方法發展不同的策略，而這些策略亦為這些方法相互之最主要差異所在 (詳見表 1)。

目前罕見有利用新近巨集啟發式解法求解凹形成本的網路問題，其中僅 Yan 與 Luo^[1,2]曾針對二分割 (bipartite) 的網路，證明含凹形節線成本的運輸問題，其最佳解必落在端點，且此端點為一可行伸展樹 (spanning tree)。Yan 與 Luo 依照凹形成本網路問題具有節線流量傾向最大流量與最小流量兩極化分布時，其總成本較小的特性，發展一初始解，並進一步用 TS、TA 和 SA 等演算法改善初始解。之後，並藉由接受率的觀念發展出 linear threshold accepting (LTA) 的演算法，經測試發現其效率優於 TA 或 SA。

表 1 各啟發式解法避免陷入局部最佳解之策略差異

演算法	避免陷入局部最佳解之策略
SA	機率函數 $e^{-\Delta}$ 做為接受暫劣解之機率，以跳脫局部最佳解。
TA	設定起始門檻值 (可接受之劣化範圍)，並根據門檻數列收斂型態調整門檻值，故門檻值會逐漸遞減 (相對門檻值)。
NM	使用隨機產生的噪音量，來擾動原來的目標函數，進而對擾動後的問題重新求解。
SSS	透過一個標準化的轉換機制，將原本高低起伏的搜尋空間 (亦即解空間) 加以平滑，以減少局部最佳解的影響。
GA	解集合 (包含可行解與不可行解) 進行交配，並允許其發生突變產生下一代解集合，進行多點平行式搜尋避免受局部最佳解的吸引。
TS	透過彈性記憶體 (禁忌串列) 之應用，以跳脫局部最佳解。
FF	當演算法進入某一局部最佳解的吸引範圍時，即翻轉目標函數，亦即將極大化的求解目標改為極小化 (或反之) 以快速脫離。
RRT	以目前找到的最佳解成本值為其紀錄值，若鄰近解之目標值在紀錄值與誤差之間即被接受，以避免局部最佳解的吸引。
GDA	具有較包容 (絕對門檻值) 的接受法則可以在傳統搜尋過程中，接受較劣之可行解，跳脫局部最佳解。

資料來源：李其澧^[38]。

2.3 鄰近搜尋法

新近的鄰近搜尋法中，模擬退火法 (SA)、門檻值接受法 (TA) 與大洪水法 (GDA)、禁制搜尋法 (TS) 頗為知名，且在不少的應用中，得到頗佳的結果，例如可參考 (韓復華、卓裕仁^[32]、韓復華、楊智凱^[39]、韓復華等^[33,34]、韓復華、林修竹^[40]、Yan 和 Luo^[2]、顏上堯等^[41])。其中，TA 與 GDA 在很多測試中皆優於 SA，故本研究中以 TA 與 GDA 兩種鄰近搜尋法並配合避免退化情形之 TS 發展區域搜尋法，並做一測試比較。以下先簡要回顧此等演算法的做法：

2.3.1 模擬退火法

模擬退火法 (SA) 又稱模擬鍛鍊法或模擬降溫法，可歸類於機率式尋優法，最初由 Kirkpatrick 等人^[27]所提出，藉由模擬物質系統的鍛鍊過程，以系統達到收斂時的穩定狀態及緩慢降溫的程序，來求解組合最佳化問題。SA 於搜尋鄰近點時，對於目標函數值 (最小化) 的改善之向下的 (downhill) 進展 (move) 必然接受；而避免落入區域最佳解，當新的進展目標值沒有改善時，對此向上的 (uphill) 進展仍以一機率 $\exp(-\Delta/T_m)$ 接受之，其中 Δ 為目標函數值的增量，參數 T_m 為目前的溫度。

其後 Golden 與 Skiscim^[42]曾用 SA 求解排程與區位問題。他們以隨機選取方式產生初始解，對於鍛鍊過程中達到收斂時之降溫方式，是採用等差遞減的降溫程序；結果發現 SA 運算時間長，控制參數對於執行時間和求解精確度影響很大。Robuste 等人^[43]曾利用

SA 求解旅行推銷員問題和車輛路線問題 (vehicle routing problem, VRP) 並認為雖然 SA 已被證明對一般 HP-Hard 問題的解，可收斂於全域最佳點，但隨著問題規模增加，收斂時間將成指數成長，雖非最有效率解法，但仍為求解傳統組合最佳化問題之理想工具。

Alfa 等人^[44]曾以 SA 法結合 3-opt 法求解 VRP，在鍛鍊過程中採用等比數列遞減的方式降溫；並於文獻中提到不同的初始解與不同的降溫程序將有助於改善 SA，但求解大型網路的結果顯示此種方式仍須努力。在國內詹達穎^[45]也對 VRP 進行類似的研究，結果顯示設定效果良好時，與文獻最佳解差很小，但電腦執行時間長，無法快速求解。

2.3.2 門檻值接受法

門檻值接受法 (TA) 最早是由 Dueck 與 Scheuer^[28]發表，其架構可歸類為以明確型的 (deterministic) 求解方式來改善初始解的交換法，初步實證於 TSP，結果發現優於 SA。TA 與 SA 步驟相似，以門檻數列來決定是否接受新的進展，凡是進展後變動成本在門檻值內的解都予以接受，而當系統達到收斂時認為系統將不再大幅變動，因此將門檻值漸次遞減，重複選取新的進展直到用盡門檻數列。

TA 與 SA 最大不同在於：SA 在鄰近點的尋優過程中，係以機率方式作為接受依據；而 TA 則對所有新的進展只要其目標值改變量低於門檻值 (最小化問題) 都接受。Dueck 與 Scheuer^[28]的文獻也提到，由於 TA 的解題特性，因此初始解可以隨機選取方式產生。韓復華、楊智凱^[39]、韓復華等^[33,34]曾以 TA 來求解 TSP 和 VRP，其文獻曾比較不同的門檻數列遞減方式，並建議對該問題以線性遞減的門檻數列求解效果較佳；結果並發現 TA 確能有效應用於 TSP 及 VRP。顏上堯等^[41]曾以 TA 求解計畫評選問題，發現效果顯著，的確較傳統鄰近搜尋法為佳。

2.3.3 大洪水法

大洪水法 (GDA) 由 Dueck^[30]發表，為一明確型求解方法，類似 TA。GDA 與 TA 最大不同處在於 TA 考慮的是與前一解的差異幅度是否落在門檻之內，以作為判斷接受與否的依據。而 GDA 在考慮新鄰近解接受與否時並不考慮前項解，而是與其所設定的標準 (水面) 作一比較，只要在此標準之下 (最小化問題)，不管其正負向改善幅度，均予以接受。與 TA 類似，GDA 開始亦設定起始水面值，之後，逐漸降低水面值，直到設定收斂的水面值而止。韓復華與林修竹^[40]針對 TA 與 GDA 在 VRPTW 問題上之應用作一探討，其測試結果發現 GDA 的解題精度較 TA 來得佳，但所費時間亦較長。

2.3.4 禁制搜尋法

由 Glover^[25,26]所提出的禁制搜尋法 (TS)，其觀念來自於人工智慧，是近幾年相當熱門的組合最佳化啟發式解法。Glover 曾提到傳統的爬山演算法一旦爬到山頂就停止，即使尚未找到全域最佳解也無法再改善；而禁制搜尋法卻不會陷入先前已找過的區域最佳解。

TS 於每次進展時皆選取改善最多的進展方向，並藉由禁制串列來防止搜尋的方向倒轉回來，同時又運用渴望機制 (aspiration) 在搜尋特定方向被禁制時，仍然能彈性選取較好的進展，以避免進展過程發生循環 (cycling)。

禁制串列本身是一種先進先出 (first-in, first-out) 的佇列，其目的在記錄最近幾次進展時的部分屬性。在禁制串列中的解稱為禁制解 (tabu solution)，任何會導致禁制解的進展方向 (tabu move)，當下次進展時若發現相同屬性已出現在禁制串列中則可拒絕此一進展。由於進展時若記錄所有屬性將花費大量時間在判比對上，因此一般只記錄數個重要的屬性。另外，渴望機制是為了讓新的進展在已被設定為禁制進展時，若其成本小於渴望機制函數則仍接受此一進展。這是因為禁制串列只記錄幾個屬性，以渴望機制來避免可行的進展被限制。

禁制搜尋法應用廣泛，在交通運輸方面，Glover^[25]曾實證 TSP 問題，結果良好。在其他領域如 Sinclair^[46]曾將 TS 與其他數種組合最佳化解法相比，並以二次指派問題實證，發現在考慮求解品質的前提下，TS 優於其他啟發式解法。Reeves^[36]曾針對機械工時編排 (machine sequencing) 問題，實證 TS 優於 SA；Hu^[47]則以最佳化設計來實證，發現 TS 在選取鄰近點時結合隨機進展 (random move) 方式，對於求解連續變數之全域最佳化 (global optimal) 問題，是一有效的工具。

2.4 小結

從以上文獻回顧可知，傳統對於凹形節線成本運輸問題，多採用分枝界限法或動態規劃法，而對於較簡易的分段線性 (piecewise linear) 成本問題，則可修改原本適用於解線性運輸問題的網路單體法，或是以啟發式解法求線性近似解，亦可分解節線以產生新的網路求解之。另外適用於含額外限制最小成本網路流動問題之拉氏鬆弛法暨次梯度法，亦可用來求線性近似解。

在新近組合最佳化啟發式解法中，如 TA、SA 等應用在凹形成本運輸問題亦有良好結果 (Yan 與 Luo^[2])，但仍侷限於特殊之二分割網路應用，忽略了網路中重要之轉運特性。此等方法再應用於一般轉運問題上，須針對轉運問題特別設計，有別二分割網路之單純特性。本研究針對多個起迄點，且具有轉運點之最小成本網路研究，利用 TA、GDA 及 TS 等新近鄰近搜尋法，發展有效的求解演算法，求解更為一般化型態的最小成本轉運問題。

三、問題定式

一般在處理運輸問題時，常以線性成本來定式運送成本，其主要原因在於簡化後的線性成本較易於求解。然而，若考慮規模經濟的特性，例如鐵路與水運等具有高固定成本或低變動成本的運送方式，通常貨物運送的每單位成本會隨貨物量的增加而減少，故其運送成本函數常可定式為凹形成本。另外，在都市運輸網路中，若考慮道路擁擠成本的運送方

式，例如 Sheffi^[6]所述之路段旅行時間，因流量增加使得旅行時間呈凸形狀成本快速增加，則可定式此問題為凸形成本函數的運輸問題。理論上，凸形成本運輸問題為 convex 問題^[6]，求解上遠較凹形成本運輸問題（為 non-convex 問題^[2]）為簡單。因此，本節針對一般具方向性之含凹形成本最小成本網路流動問題之數學模式作一描述，且探討其有方向性伸展樹（spanning tree）之相關理論特性，以為後續演算法設計之依據。

3.1 數學定式

令有方向性之網路 $G = (\bar{N}, \bar{A})$ 表示一般性之網路，其中 \bar{N} 為所有節點之集合， \bar{A} 為節線集合，則含凹形節線成本的最小成本網路流動問題或轉運問題可定式為：

◎轉運問題模式

$$\text{Min} \quad \sum_{ij \in \bar{A}} f_{ij}(x_{ij}) \quad (1)$$

$$\text{S.T.} \quad \sum_j x_{ij} - \sum_k x_{ki} = b_i \quad \forall i \in \bar{N} \quad (2)$$

$$x_{ij} \geq 0 \quad \forall (i, j) \in \bar{A} \quad (3)$$

目標式(1)為最小化總成本，其節線 ij 的成本函數型態 $f_{ij}(x_{ij})$ 為凹形成本；限制式(2)為流量守恒限制式，當 i 點為供給點時 $b_i > 0$ 、為需求點時 $b_i < 0$ ，其餘各點（為轉運點） $b_i = 0$ ；限制式(3)為流量非負限制式。集合 \bar{A} 表示有方向性節線之集合，至於不具方向性網路的數學模式與此雷同。為簡化研究測試，本研究假設各節線流量無上限的限制。實際上，有節線流量上限的問題，可透過網路修正改為無節線流量上限的問題（Ahuja 等人^[3]）。值得注意的是，為保證此最小成本網路流動問題有可行解，本研究設定總供給量等於總需求量，因此後節產生測試的網路時必須滿足 $\sum_i b_i = 0$ 之條件。至於在實務上可能面臨的總供需量不均衡的求解方式，一般是以加入虛擬節點及虛擬供需量，使總供需量平衡後求解。

此種轉運問題的定義即是將所有的物資從所有供給點，透過所有供需點及轉運點，運送到所有需求點，並使花費的運送成本最小化。為了便於演算法之發展，本研究對問題形式做了下列假設：

1. 所有節線的流量介於 0 到無窮大之間，亦即節線流量無上限。
2. 由於本研究旨在探討組合尋優啟發解法應用在含凹形節線成本運輸問題上的適用性，因此對於成本函數型態不作深入探討。為簡化問題的複雜度並方便測試，本研究參考 Yan 與 Luo^[1,2]中節線成本的設計，假設各節線成本為 $f_{ij} = c_{ij}\sqrt{x_{ij}}$ ，其中 x_{ij} 為流量， c_{ij} 為成本參數。此函數型態顯示全部運輸成本隨運送量的增加而增加，但單位的運輸成本

($= c_{ij} / \sqrt{x_{ij}}$) 則隨運送量的增加而呈遞減狀態。在實務上，此成本函數型態類似許多運輸問題之成本函數。至於其他類凹形成本函數的測試，可為後續研究。

3. 令節點總供給量等於節點總需求量，以確立網路流量守恒。
4. 二節點間可能存在來回方向之節線，此兩方向節線的成本由隨機產生，不一定相同。

3.2 問題特性說明

本研究網路問題屬於非線性的最小凹形成本網路流動問題。目前一般最小網路流動問題的求解大都利用伸展樹 (spanning tree) 之觀念求解，例如網路單體法 (network simplex method) (Ahuja 等人^[3])。在一般的最小網路流動問題中，一個可行伸展樹 (feasible spanning tree) 代表的是一組可行流量解，相當於單體法中的基本可行解 (basic feasible solution)。而可行伸展樹在線性規劃問題的本質上，亦可表示成一個端點解 (extreme point) (Gallo 等人^[10])。由於本研究問題的各節線成本為凹形成本，故整個目標函數亦形成一凹形成本函數 (Ahuja 等人^[3])，而參考 Yan 與 Luo^[2] 類似的證明，此凹形成本函數之轉運問題的最佳解，必落在可行解區域 (feasible region) 的一端點上，各端點為一區域最佳解。

由於本問題的限制式與一般最小成本網路流動問題的流量守恒限制式相同，因此類似於純網路問題，若所有供需點的供給量或需求量皆為整數，則所有端點解 (伸展樹) 亦為整數解 (Ahuja 等人^[3])。又如上所述，因最佳解必落在端點解上，所以對於凹形成本的最小化網路問題，尋找全域最佳解的方法需要列舉出許多的區域最佳解。因此如何有效率的列舉出端點解來找尋最佳解，將是本研究的焦點。

此外，值得一提的是，在凹形成本網路中給定一個可行伸展樹，必對應唯一的一組路段解流量，滿足供給點與需求點間之運送。說明如下：給定一個可行伸展樹作流量的指派，與目標函數無關，因此凹形成本網路在此與線性成本網路相同。不論針對之網路為具方向性網路或不具方向性網路，可行伸展樹中任意兩點間只有一條路徑連結，網路各路段流量可由所有路徑流量累加得到，因此對應唯一的一組路段解流量。

綜合上述，本問題有下列特性：

1. 最佳解落在端點。
2. 端點解恰為整數解，且對應唯一的一組路段解流量。
3. 最佳解恰為端點及整數解。

本研究網路問題為一非線性規劃問題 (NLP)，若透過鄰近搜尋法尋求最佳解，則理論上需搜尋目前解的所有線性鄰近解，如此需搜尋的數量將變得相當龐大。然而，根據上述最佳解恰為端點解及整數解的特性下，本研究在搜尋的過程中，僅需針對鄰近的端點進行求解。換言之，本研究的可行解區域可從廣闊的線性解集合，濃縮至部分的端點解集合。此等端點解散落在線性解集合中，與一般整數規劃問題的離散解散落在線性解集合中的特性相似，故可利用一般求解整數線性規劃問題 (ILP) 的鄰近搜尋法 (或組合最佳化的啟發式解法) 進行求解。

最後，值得注意的是，網路在求解過程常會出現退化現象 (Yan 與 Luo^[2])，例如在單體法中，一基本可行解中，若有變數為零則表示為退化情形。因此在使用伸展樹求解本問題時，因本問題供需量在各起迄點間之運送，並不需經過所有伸展樹上的節線 (路段)，因此可能會有許多節線上流量為零者，此於網路單體法中可以修正其做法以達到退化的改善 (Ahuja 等人^[3])。然在本研究後節利用區域搜尋法設計演算法上，以交換節線達成改善的方式，且用目標值大小判斷搜尋方向上，可能難以控制退化現象，因此將會使用禁制搜尋的觀念來改善退化現象，此設計將在下節加以介紹。

四、求解演算法設計

本研究參考羅守正^[48] 針對 TA、TS、SA 與各種組合啟發式解法對二分割運輸網路問題的測試結果發現，TS 的求解品質雖頗佳，但求解相當費時，而若以 TA 結合 TS 策略，如能配合良好的系統參數與起始解，則其求解績效常較 TS 為佳。因此，本研究嘗試以啟發式解法中的綜合式解法，首先有效率產生起始解，然後藉由交換網路上的可行伸展樹的節線進行改善，再配合 TA、GDA 等組合尋優啟發法並結合 TS 策略，來判斷是否達到停止條件。

4.1 初始解

根據以往經驗區域搜尋法起始解較好，其後續改善的結果也會較好 (Yan 與 Luo^[2])，因此本研究使用啟發式解法中的綜合式解法有效率地產生好的初始解。本研究根據凹形成本特性發展適合的起始解法。

Kuhn 與 Baumol^[22] 曾提出貪婪法 (greedy method) 等的數種啟發式解法，其求解策略為：指派最大流量至成本最小的節線，在 Yan 與 Luo^[2] 研究中，曾參考其原理提出流量導向的起始解啟發解法，是依據凹形成本運輸問題的特性所發展。本研究依據凹形成本的特性，應用在本問題上。於每回供給點將供給量運送完成後，根據新的 x_{ij} 以成本函數 $c_{ij}\sqrt{x_{ij}}$ 更新經過的節線成本，再做下一個供給點之運送。而供給點的選取順序則依照供給量大小，由大到小選取。此初始解法的詳細步驟如下，流程如圖 2 所示：

步驟 1. 確定供需點關係：以標籤修正法標籤值判斷供給點可到達之需求點，並記錄之。

步驟 2. 將供給點依照供給量由大到小排序。

步驟 3. 依序選取供給點運送至需求點：

3.1 依序選取供給點進行指派。若選取完畢，則進行步驟 5，否則進行步驟 3.2。

3.2 依步驟 3.1 選取之供給點供給量且根據成本函數 $c_{ij}\sqrt{x_{ij}}$ 更新所有路段成本，再以標籤修正法，依據新的路段成本求得供給點到其餘各點之最短路徑，並記錄之。

3.3 將可運送到的需求點依照需求量排序，由大到小接收供給點之運送，並記錄供需

節點對與運送量。若已無可運送之需求點但剩餘供給量 S_i 大於 0，則進行步驟 4。
若供給量等於 0 表示運送完成，則進行步驟 3.5。

3.4 更新路段流量 x_{ij} 。

3.5 以新的 x_{ij} 根據成本函數 $c_{ij}\sqrt{x_{ij}}$ 更新路段成本，回到步驟 3.1。

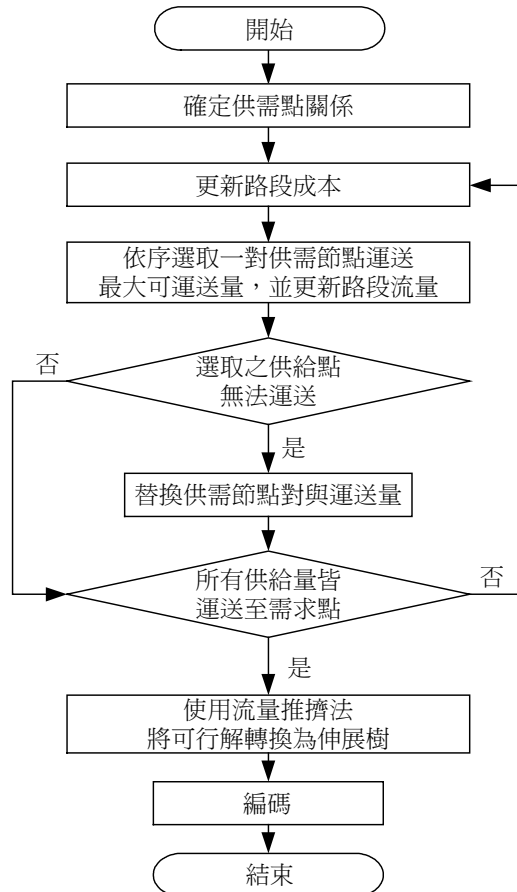


圖 2 初始解流程圖

步驟 4. 替換供需節點對與運送量：

- 4.1 依先前步驟 3.3 之記錄，找出運送至 j 點之供給點 i' 、運送量 x 。若 $S_i < x$ ，則進行步驟 4.2；若 $S_i = x$ ，則進行步驟 4.3；若 $S_i > x$ ，則進行步驟 4.4。
- 4.2 先前記錄 i' 、 j 之運送量改為 $x = x - S_i$ ，並記錄 i 、 j 與運送量 S_i ，更新路段流量，回到步驟 3.4。
- 4.3 刪除先前記錄 i' 、 j 與運送量 x ，記錄 i 、 j 與運送量 S_i ，更新路段流量，回到步驟 3.4。

4.4 刪除先前記錄 $i' \setminus j$ 與運送量 x ，記錄 $i \setminus j$ 與運送量 S_i ，更新路段流量與路段成本，回到步驟 4.1。

步驟 5. 使用流量推擠法 (flow augmentation algorithm) (Yan 與 Young^[49]) 將可行解轉換為伸展樹。其作法為修改 Powell^[50] 之最小成本流量推擠演算法，在網路上尋找最小成本流量推擠路徑 (least cost flow augmenting path)，以可行流動解 (非伸展樹) 為基礎，將此非伸展樹解推擠成一伸展樹解。

5.1 任選一節點 R 為伸展樹之根點 (root)。記錄 R 於一佇列 Q 中。

5.2 讀取 Q 中第一個未標記點標記為 a ，若所有點皆標記過，則進行步驟 5.6。

5.3 將 a 在網路中相連接之節點，點號由小到大依序以 b 表示，表示至最後一個 b ，則回到步驟 5.2。

5.4 若 b 標記過，則回步驟 5.3 更換下一個 b ；若 b 未標記但不在 Q 中，則記錄 b 點前置點為 a ，將 b 記錄在 Q 中，回到步驟 5.3 並更換下一個 b ；若 b 未標記但在 Q 中，則進行步驟 5.5。

5.5 此時有迴圈產生，根據前置點找出此迴圈，並使用流量推擠法改變此迴圈中各節線流量，消除迴圈。回到步驟 5.3 更換下一個 b 。

5.6 各點與前置點間節線流量之集合即為一可行解，若節點無前置點者表示該點為不曾用到之轉運點，則隨機以網路中之一節線連結形成一伸展樹。

4.2 門檻值接受法 (TA)

Dueck 與 Scheuer^[28] 認為門檻值接受法中無論新的進展目標值是否有改善，只要目標值的改變量在門檻值內就予以接受。韓復華與楊智凱^[39] 以起始目標值乘以固定百分比做為起始門檻值。本研究以韓復華與楊智凱^[39] 方式設定初始門檻值，而以新舊目標值差與舊目標值的比率作為判斷是否在門檻值內的依據。

有了起始解便可進行改善步驟，本研究改善方式為選取非伸展樹上節線加入伸展樹中，再使用流量推擠法形成新的可行伸展樹，以達成改善。然而當選取非伸展樹上節線時，若考慮全部可能節線，則每次進展將有 $m - n + 1$ (m 為節線數， n 為節點數) 條節線及同數量的鄰近解需要測試，如此需花費大量時間。值得一提的是，Golden 與 Skiscim^[42] 運用 SA 處理 TSP 問題，曾令 n 次潛在的 swap 為一個 epoch，並記錄其中最佳進展的目標值；每次檢查是否達到收斂時，檢驗目前的目標值是否和以前任何一個 epoch 的目標值相同，若有相符則認為系統已達到收斂，可予以降溫。Yan 與 Luo^[2] 亦參考其作法，利用 epoch 候選解的觀念來搜尋鄰近解，即將欲搜尋的鄰近解 (候選節線) 設定為一個 epoch。因此本研究令候選節線數 epoch = n 的倍數，候選節線數目為 n 、 $2n$ 、 $3n$ 、 \dots 等，並從候選節線中選出改善幅度最大的節線來評估進展方向，以此方式來搜尋鄰近解。

至於系統收斂，Yan 與 Luo^[2] 曾使用兩種方式判斷系統是否達到收斂，一是採回合數 (iteration) 控制，二是以具上次進展後未有改善的未進展次數 (no_update) 控制。本研究

參考其作法應用在本問題上，至於何者較為合適則在下一節中探討。本研究參考 Yan 與 Luo^[2]之 TA 發展適合本問題之演算法，並運用起始門檻值 T_0 、門檻數列個數 num、每次進展時的候選節線數 epoch、回合數 iter 與未進展次數 no_update 等參數。若採回合數 (iteration) 控制系統收斂，則此法之具體步驟如下：

步驟 1. 給定起始解和起始條件：

- 1.1 產生數個初始解選取目標值最佳的解當做起始解。
- 1.2 設定 T_0 、num、iter、epoch、no_update。令 count = 0、 $x = 0$ 。

步驟 2. 選取最佳的鄰近節線進行交換：

- 2.1 以隨機的方式從鄰近的候選解中選取節線，並選出改善幅度最大的節線加入目前的伸展樹。
- 2.2 經由流量推擠法消除迴圈產生新的可行伸展樹，並計算目標值。

步驟 3. 評估新的進展：

- 3.1 計算 $\Delta E = \text{新解的目標值} - \text{目前解的目標值}$ 。若 ΔE 值小於門檻值則接受此一新解，若大於或等於門檻值，則予以拒絕並進行步驟 4。
- 3.2 若接受新的進展，則更新目標值與目前解。

步驟 4. 若達到收斂時 (count > iter) 則降低門檻值 T_i ， $x = x + 1$ ，count = 0；否則令 count = count + 1，回步驟 2。

步驟 5. 若 $x > \text{num}$ 代表門檻數已使用完，則結束；否則回到步驟 2。

注意，若以未進展次數 (no_update) 控制系統收斂，則上述步驟 3.2 需加上 count = 0 指令。步驟 4 的指令 (count > iter)，需改為 (count > no_update)。

4.3 大洪水法 (GDA)

GDA 使用絕對門檻值與 TA 之相對門檻值方式不同，但基本的觀念類似。一般在系統達到收斂時降低洪水面，當找到新解的目標值低於洪水面則接受此一進展，反之，則不接受 (Dueck^[30])。本研究在洪水面的設定上，以目前找到最好的解為基準，並以此基準乘上某一遞減的比率作為洪水面初始與降低的控制。若找到新的解，其目標值低於目前最好的解，則更新目前最好解與其目標值，因此洪水面可能快速降低，其意義為只接受目前最好解之某一目標值範圍內的解，如此作法可避免類似 TA 使用目標值差變化率的作法時，可能所造成目標值持續變壞，致使新的解與目前最好解的目標值相差甚多的情形。

本研究運用起始洪水面比率 G_0 、洪水面數列個數 num、每次進展時的候選節線數 epoch、回合數 iter 與未進展次數 no_update 等參數，若採回合數 (iteration) 控制系統收斂，則此法的具體步驟如下：

步驟 1. 給定起始解和起始條件：

- 1.1 產生數個起始解選取目標值最佳的解當做初始解與目前最好的解。

1.2 設定 G_0 、num、iter、epoch、no_update，令 $\text{count} = 0$ 、 $x = 0$ 。

步驟 2. 選取最佳的鄰近節線進行交換：

2.1 以隨機的方式從鄰近的候選解中選取節線，並選出改善幅度最大的節線加入目前解（伸展樹）。

2.2 經由流量推擠法消除迴圈產生新的可行伸展樹，並計算其目標值。

步驟 3. 評估新的進展：

3.1 計算洪水面 $G = G_{0_i}^*$ 目前最好的目標值 (V_g)，若新解的目標值 (V_n) 小於 G 則接收此一新解，若大於 G ，則予以拒絕並進行步驟 4。

3.2 若 V_n 小於 V_g ，則更新目前最好解與 V_g 。

3.3 若接受新的進展，則更新目前解與其目標值。

步驟 4. 若達到收斂時 ($\text{count} > \text{iter}$) 則降低洪水面 G_i 、 $x = x + 1$ ，令 $\text{count} = 0$ ；否則，令 $\text{count} = \text{count} + 1$ ，回到步驟 2。

步驟 5. 若 $x > \text{num}$ 代表已使用到最低的洪水面值，則結束；否則回到步驟 2。

注意，若以未進展次數 (no_update) 控制系統收斂，則上述步驟 3.3 需加上 $\text{count} = 0$ 指令。步驟 4 的指令 ($\text{count} > \text{iter}$)，需改為 ($\text{count} > \text{no_update}$)。

4.4 結合禁制策略的門檻值接受法 (TTA) 與大洪水法 (TGDA)

為了提升求解品質，避免因退化造成一再搜尋相同方向的影響，以增加新進展的機會，本研究嘗試引用禁制搜尋 (TS) 策略於門檻值接受法與大洪水法中。TS 為 Glover 所提出，對於已搜尋過的進展方向，運用禁制串列在適當的記憶體空間內記錄新進展的部分屬性，不限制新進展的目標值一定要有改善，只要新的方向不在禁制串列中便予以更新，因此 TS 允許新解的目標值較原來的差，如此便不會陷入區域的最佳解。Glover^[26] 曾提到，禁制串列的長度對求解品質並沒有顯著的影響，其文獻中建議禁制串列的長度為 7。而本研究問題在禁制串列的儲存與檢驗方式中，若只記錄過程中節線的編號，則不足以反映新的進展是否已重複；若記錄整個 cycle 的變動部分，則需要記錄大量的伸展樹資訊，而如此將耗費太多時間在檢驗上。因此，本研究參考 Yan 與 Luo^[1] 作法將原目標函數中所有組成節線的成本，各乘上一介於 0 與 1 之間的隨機亂數 P_{ij} ，所得新目標函數如下列所示：

$$\text{原目標函數} \quad f_{ij}(x_{ij}) = c_{ij} \sqrt{x_{ij}} \quad (4)$$

$$\text{新目標函數 1} \quad f_{ij}(x_{ij}) = P_{ij}^1 c_{ij} \sqrt{x_{ij}} \quad (5)$$

$$\text{新目標函數 2} \quad f_{ij}(x_{ij}) = P_{ij}^2 c_{ij} \sqrt{x_{ij}} \quad (6)$$

$$\vdots \quad \quad \quad \vdots$$

以數個不同目標函數計算同一伸展樹，記錄所得各值作為禁制串列的參考值，若進展中找到的兩個伸展樹其各目標函數值皆相同，則表示兩個伸展樹相同的機會很高。本研究曾針對不同組數的係數矩陣進行測試，發現若組數小於 5 時，較會造成禁制解的誤判，而導致搜尋效果較差，亦較難反映禁制的功能；而在 5 組以上，則未發現有誤判禁制解的情況，其求解的效果亦較佳。由於 5 組以上的測試所求得的目标值皆相同，但以 5 組的測試時間最短（因需存取禁制資訊與檢查的時間最短），因此本研究在後節便以 5 組不同的係數矩陣進行測試，即每條節線對應 5 組係數值，以產生五個不同的目標函數值。

另外，由於伸展樹所需的節線數為 $n-1$ 條，但並不需要完全使用 $n-1$ 條節線來滿足供需量運送，因此伸展樹上有可能若干的節線流量為 0，而在節線交換去除迴圈時，每次推擠迴圈上最小之節線流量也因此可能為 0，故會造成目標值無變動的退化現象。因此，本研究在選取節線加入伸展樹進行交換的步驟時，記錄加入之節線為另一種禁制串列，判斷是否接受後續之進展方向。至於此禁制串列的長度對求解的好換，則需要加以測試決定。綜合上述，本研究有兩種禁制串列，一為禁制相同伸展樹的目標函數值矩陣，稱為禁制串列 1，二為禁制退化解的禁制矩陣，稱為禁制串列 2。

本研究除上述二禁制串列的設定外，亦參考 Glover^[26] 設置渴望機制，亦即若新進展的解雖存在禁制串列 1，但若其目標值可改善目前解，則亦接受此新進展。綜合本研究二禁制串列與渴望機制，本研究禁制搜尋的策略如下：本研究以伸展樹為基本解，當一新進展的解與禁制串列 1 的解相同時，表示找到過去的方向，然而若與目前的解（禁制串列 1 中最近的解）相同時，表示遇到退化解。因此，當一新進展的解與禁制串列 1 的解相同時，但不是退化時，則除非滿足渴望機制，否則不予接受。然而，若新進展的解與禁制串列 1 的目前解相同時（即為退化），則此時若交換之節線亦在禁制串列 2 中，則表示此搜尋方向會得到過去相同的伸展樹，因此予以禁制。相反的，若此時交換之節線，不在禁制串列 2 中表示該解雖為退化解，但新進展的伸展樹將不同，因此為使搜尋的解不致過份侷限，本研究仍接受此退化解的搜尋方向。值得一提的是，在加入 TS 策略的考量後，TTA 與 TGDA 需再進一步考慮新的進展方向是否在禁制串列之中，以判斷是否接受此一新解，以有效率地挑選出可能不適合的進展方向。因此，此 TTA 與 TGDA 等混合式演算法，雖稍有別於原本 TA 與 GDA 只要目標值的改變量或新解的目標值在門檻值內就接受的作法，但仍保有 TA 與 GDA 在搜尋上的門檻值接受特質。此 TTA 與 TGDA 的設計係綜合 TS、GDA 及 TS 的優點，可望獲得較佳的求解績效。

最後，本研究運用起始門檻值 T_0 （或起始洪水面比率 G_0 ）、門檻數列（洪水面數列）個數 num、每次進展時的候選節線數 epoch、回合數 iter、未進展次數 no_update、與禁制串列長度 tabu_num 等控制參數，設計 TTA 與 TGDA，其具體的步驟如下：

步驟 1. 給定起始解和起始條件：

- 1.1 產生數個起始解選取目標值最佳的解當做初始解。
- 1.2 給定控制參數值。

步驟 2. 選取最佳的鄰近節線進行交換：

2.1 以隨機的方式從鄰近的候選解中選取節線，並選出改善幅度最大的節線加入目前的伸展樹。

2.2 經由流量推擠法消除迴圈產生新的可行伸展樹，並計算其目標值。

步驟 3. 評估新的進展：

3.1 依據 TA 或 GDA 的方式判斷是否接受新進展，是則進行步驟 3.2；否則予以拒絕，進行步驟 4。

3.2 當新的進展不在禁制串列 1 中，但滿足渴望機制，或在禁制串列 1 的目前解中，但不在禁制串列 2 中時，則接受此一新解；否則予以拒絕，進行步驟 4。

3.3 更新禁制串列。

3.4 若接受新的進展則更新目標值與目前解。

步驟 4. 若達到收斂時，則降低門檻值或洪水面；否則回到步驟 2。

步驟 5. 若門檻數列或洪水面使用完，則結束；否則回到步驟 2。

五、實驗測試

為了評估第四節所提出之演算法之績效，本節針對具方向性含凹形節線成本網路，以電腦產生路網資訊，接著以不同方式產生初始解，再以不同機制控制改善方式，最後配合敏感度測試提出結論分析與適用參數。本研究使用 C++ 語言撰寫程式並測試演算法，測試環境為 K7-700 之個人電腦。本節結構如下：5.1 節說明測試網路之產生步驟；5.2 節測試各演算法求解績效及適合的系統參數；5.3 節以 5.2 節所得的最佳系統參數測試小型網路比較各鄰近搜尋法之正確性。

5.1 網路產生器設計

一般而言對於組合最佳化問題之啟發式解法參數之設定，或比較各演算法的績效，需要大量的網路資料作為測試，因此我們藉助電腦高速運算的功能，編寫網路產生器程式，以亂數的方式隨機地產生大量隨機網路 (randomized network)。Demeulemeester 等人^[51]提到不同演算法在不同規模與參數所得到的績效不同，因此在設計隨機路網以測試演算法時，應以不同規模及參數值的隨機網路為輸入。例如 Klingman 等人^[52]曾針對指派問題、線性成本運輸問題、及最小成本流動問題等設計過隨機性網路產生器程式 (NETGEN) 以測試演算法績效；Yan 與 Luo^[2]針對運輸問題，產生二分割且有方向性的運輸網路進行測試。

本研究的網路產生器設計方式首先確定網路節點數 n ，以節點數多寡表示網路的規模大小。然後選取一個節點以正向連接機率 P_f 控制是否與其他節點達成連結，達成連結則增加一條正向節線於兩點間，同時隨機產生節線成本函數 $c_{ij}(x_{ij})$ 之係數 c_{ij} 值。當正向達

成連結時以反向連接機率 P_b 判斷是否存在反向節線，若存在則產生另一 c_{ji} 值為反向節線成本函數之係數。而節點的選取順序以產生亂數方式隨機選取，並以連接範圍 s 控制兩節點相連的最大範圍，若兩節點號碼超過 s 時，則不進行連結。最後再判斷此網路是否為連續性 (connected) 網路，若為連續性，則網路產生完成；若不為連續性，則表示所有節點被分成數個區域，再以隨機選取不同區域的節點進行連結，進而達成網路連續之目的。此節點節線相連產生法的詳細步驟如下：

步驟 0. 給定初始條件： n 、 P_f 、 P_b 、 s 。令 $i = 1$ ， $a = 2$ 。

步驟 1. 隨機排序節點：

隨機選取不曾被選取的節點，存於一維矩陣 \mathbf{R} 中， R_i 代表第 i 個節點號碼。

步驟 2. 達成節點連結：

2.1 $j = i + 1$

2.2 以 P_f 判斷 $R_i \rightarrow R_j$ 是否連接，若相連接則產生成本係數 c_{ij} ，並以 P_b 判斷 $R_j \rightarrow R_i$ 是否相連，相連則再產生 c_{ji} 值。

2.3 $j = i + 1$ ，若 $j \leq n$ 且 $j - 1 \leq s$ 則進行步驟 2.2，否則 $i = i + 1$ ，進行步驟 2.4。

2.4 $i < n$ 則回到步驟 2； $i \geq n$ 則進行步驟 3。

步驟 3. 檢查網路是否相連，是則結束；否則表示步驟 2 造成 k 個區域節點之連結，但各區域間不相連，進行步驟 4。

步驟 4. 網路達成相連：

4.1 在區域 a 中隨機選取數個節點與區域 1 中隨機選出之節點達成正向連結，再以 P_b 判斷反向連結。

4.2 增加步驟 4.1 達成連接之正反向節線，並給予隨機節線成本係數。

4.3 區域 a 與區域 1 結合為新的區域 1， $a = a + 1$

4.4 若 $a > k$ 則結束，否則回到步驟 4.1。

完成上述步驟可得到一連結的有方向性網路。產生了網路連結節線與節線成本，尚須設置供需點與供需量以完成完整的網路輸入資料。供需點與供需量以隨機方式產生，首先選取一節點為供給點，再由此點任意在網路中移動 Move_Num 次相連的節點，將最後移動到的點設定為需求點，儲存並標記此兩節點。之後，再隨機產生供需量，並累加該兩節點已經存在的供需量，如此完成一回合步驟。重複以上步驟 Ass_Num 回合後，便可完成網路的供需點與供需量產生。其中除了第一回合供給點以均一機率選取各節點外，其他回合供給點之選取以機率 P_k 判斷選擇節點類型，符合機率者由標記過的節點中選取，不符合機率者由其他未標記節點選取。標記過的點表示曾經為供需點之節點。移動次數 Move_Num 與每回供需量皆以亂數在各自設定的上下限內產生，Ass_Num 則以節點數乘以一控制供需點密度的參數 D 得到， D 值範圍介於 $0 \sim 1$ ，在產生網路前給定。此供需量產生法的詳細步驟如下：

步驟 0. 初始值設定：設定 P_k 、 $i = 1$ 、各點供需量為 0。

步驟 1. 選取供給點：

- 1.1 假如 $i = 1$ 則隨機選取網路中節點為供給點，標記供給點，進行步驟 2；否則進行步驟 1.2。
- 1.2 產生一個 $0 \sim 1$ 的亂數，若小於等於 P_k ，則隨機選取標記過的節點；若大於 P_k ，則隨機選擇未標記的節點。
- 1.3 標記供給點， $i = i + 1$ 。

步驟 2. 選取需求點：

- 2.1 由供給點開始，隨機選取相連之節點移動，移動 Move_Num 次停止，停止時移動到的點便為第 i 回合需求點。
- 2.2 標記需求點。

步驟 3. 累加供需量：隨機產生供需量值累加至供需點上。

步驟 4. 若 $i \leq \text{Ass_Num}$ ，則回到步驟 1，否則結束。

以此方式產生的網路必存在有可行解，可避免無解的情形發生。為檢驗本研究提出之演算法，並評比其績效，本研究用來測試的路網，有 10、20、50、100 與 150 等節點數規模，規模 50、100 與 150 的網路各自配合 0.2、0.5、0.8 三個控制供需點密度的參數 D 值，而規模 10 與 20 的網路因節點數過少只測試 D 值為 0.5 的供需點密度，總共有 11 種網路。各不同節點數網路之節線數與網路密度列於表 2。其他設定值，如節線成本係數設定值在 50 至 300 之間，每回合供需量在 1 至 150 之間。例如圖 3 為 20 個節點，參數 $D = 0.5$ 的網路圖，本研究以 20_0.5 表示該網路，其餘網路亦以此類似方式表示。

表 2 測試網路節線數與網路密度

網路節點數	10	20	50	100	150
網路節線數	17	47	220	730	1117
網路密度 (%)	17.0	11.8	8.8	7.3	5.0

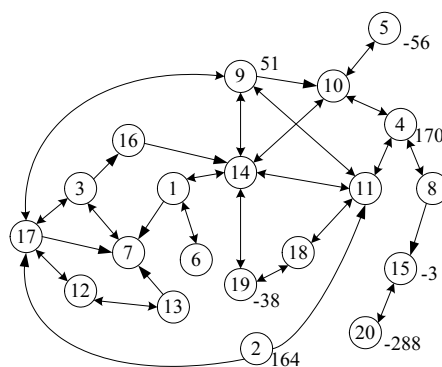


圖 3 20_0.5 之網路圖

5.2 鄰近搜尋法

由於組合最佳化演算法對於不同問題型態，其適用之系統參數或有不同；又對於不同的網路種類及網路規模，參數的變化亦可能影響求解品質，因此必須對各系統參數進行測試，以期能找出優良求解品質的系統參數。值得注意的是，本研究所提出的各區域搜尋法適用的系統參數，係就對本問題形式而言，至於是否能應用於其他問題，則需再加以實證。

在各種鄰近搜尋法的參數方案測試過程中，本研究將比較各方案求得的目標值與最優解目標值相比，並以誤差百分比表示，其目的在評估各解法在不同網路下的實證績效，誤差百分比越小表示該方案參數在該網路求解的結果越好。本研究主要提出可行的求解策略，且僅對各演算法之系統參數做初步測試，找出適用的參數設定，至於更廣泛與更精確的測試則可留待後續研究。

5.2.1 起始解

由許多文獻可發現，區域搜尋法求解品質與起始解相關，由較好的起始解開始搜尋會有較好的結果 (Yan 與 Luo^[2])，因此本研究產生大量的初始解，選取其中目標值最低的解為區域搜尋之起始解，並以此起始解測試不同的演算法，配合參數設定，其意義為皆由同一起點開始搜尋，測試何種演算法配合參數會有較好的結果。故本研究以凹形初始解產生法產生 1,000 個解，選取其中最好的解為區域搜尋法的起始解，對較大型網路測試所得起始解如表 3 所示。由表 3 亦顯示此起始解仍有相當的改善空間。

表 3 區域搜尋法較大型網路起始解

網路	100_02	100_05	100_08	150_02	150_05	150_08	平均
起始解目標值	33927.4	59815.5	70202.8	54988.8	98198.0	106113.1	70540.93
誤差百分比	10.0%	27.1%	30.8%	30.1%	32.2%	35.0%	27.5%

註：誤差百分比 (%) = (起始解目標值 / 本研究各演算法配合各參數所得最小目標值) - 1。

5.2.2 TA 與 GDA 測試與系統參數分析

TA 最重要的系統參數為門檻值遞減方式，本研究採線性遞減數列方式，而起始門檻值的設定方式，在韓復華與楊智凱^[39]的研究中以起始路段成本的固定百分比 0.2% ~ 1% 作為起始門檻值；Yan 與 Luo^[2] 則以目標函數值的變動率來做為門檻值判斷的依據，凡是目標值變動幅度在門檻值之內皆予以接受，因此本研究起始門檻值採起始可接受的目標值變動比率。

GDA 與 TA 的差別在於 GDA 採絕對的門檻值接受方式，與 TA 相對門檻值不同。而絕對的門檻值即洪水面的設定方式，本研究以目前找到最佳的目標值為基準，加上此基準

乘以一比率所得的值的總和為洪水面，起始洪水面為起始解可接受與目前最佳解的差值比率。

本研究以較大型網路為基礎，測試 TA 與 GDA 的系統參數：如起始門檻值、門檻值數列長度、每次進展時的候選節線數。至於是否達到收斂測試兩種參數，第一為固定回合數、第二為距上次進展後連續無改善的次數。各測試方案之系統參數值列於表 4；由於鄰近搜尋方式係採固定次數之隨機搜尋，故各方案重複測試 3 次，取 3 次中最好一次的結果，將 TA 測試結果之目標值誤差百分比列於表 5；GDA 測試結果之目標值誤差百分比則列於表 6。

由測試結果發現，TA 在 0.5 的起始門檻設定值與門檻值數列長度為 60 的方案 B 有較好的結果，GDA 在 0.39 的起始門檻設定值而門檻值數列長度為 100 的方案 E 有較好的結果。TA 的最佳解結果稍優於 GDA 結果。其中，因為 GDA 以目前找到最好的解為基準，可以因最好解的更新而大幅降低洪水面的數列次數，加速收斂效果。至於 TA 則給予較高的起始門檻值較少的門檻值數列長度有好的結果。

表 4 TA 與 GDA 各測試方案之系統參數值

參數名稱 \ 方案	A	B	C	D	E	F	G	H	I
起始門檻值	0.39	0.5	0.1	0.39	0.39	0.39	0.25	0.2	0.39
門檻值數列長度	60	60	60	100	100	30	30	100	100
候選節線數	2n	2n	2n	n	n	n	2n	2n	n
未改善次數	30	30	30	10	50	10	50	0	0
回合數	0	0	0	0	0	0	0	200	100

表 5 TA 測試結果之誤差百分比

	A	B	C	D	E	F	G	H	I
100_02	0.0%	0.0%	0.0%	2.2%	0.0%	2.5%	0.0%	0.0%	0.0%
100_05	0.0%	2.5%	0.9%	6.4%	4.3%	5.3%	4.7%	0.0%	3.5%
100_08	0.0%	1.3%	3.3%	1.9%	1.1%	3.2%	1.9%	3.8%	5.7%
150_02	3.4%	0.3%	0.3%	6.5%	3.2%	4.7%	1.3%	2.4%	4.7%
150_05	4.0%	2.9%	2.8%	3.9%	1.0%	5.4%	2.9%	4.1%	2.6%
150_08	2.5%	0.8%	0.7%	1.6%	0.0%	1.9%	1.8%	2.8%	0.7%
平均	1.65%	1.30%	1.33%	3.75%	1.60%	3.83%	2.10%	2.18%	2.87%

註：誤差百分比 (%) = (該參數下所得之目標值 / 本研究各演算法配合各參數所得最小目標值) - 1。

表 6 GDA 測試結果之誤差百分比

	A	B	C	D	E	F	G	H	I
100_02	0.0%	1.5%	0.0%	0.0%	0.0%	0.0%	0.0%	1.5%	1.6%
100_05	4.8%	4.8%	4.8%	4.9%	4.0%	5.4%	1.7%	6.3%	0.9%
100_08	1.9%	3.2%	1.2%	7.2%	1.9%	7.9%	1.3%	3.1%	2.9%
150_02	2.2%	3.6%	2.4%	2.8%	1.4%	5.7%	1.6%	3.3%	2.9%
150_05	2.5%	2.5%	2.6%	3.6%	0.8%	9.4%	4.7%	4.2%	0.7%
150_08	2.3%	2.7%	0.7%	3.3%	1.8%	4.0%	3.1%	2.9%	4.1%
平均	2.28%	3.05%	1.95%	3.63%	1.65%	5.40%	2.07%	3.55%	2.18%

註：誤差百分比 (%) = (該參數下所得之目標值 / 本研究各演算法配合各參數所得最小目標值) - 1。

5.2.3 TTA 與 TGDA 測試與系統參數分析

由於此兩種方法是架構在 TA 和 GDA 上，因此參數的項目大致與 TA 和 GDA 相同，包括：起始門檻值、門檻值數列長度、每次進展時的候選節線數、兩種收斂測試參數固定回合數與距上次進展後連續無改善的次數；只增加一項禁制串列長度用以討論適用之禁制串列長度。各測試方案之系統參數值列於表 7 (注意，由於組合效應，測試結果表 7 參數與表 4 參數不同)；TTA 測試結果之目標值誤差百分比列於表 8；而 TGDA 測試結果之目標值誤差百分比則列於表 9。

由測試結果可知，TA 與 GDA 加入禁制觀念後方案可找到較小的誤差平均值。值得注意的是 TTA 在禁制串列長度為 3 (方案 G) 時，較一般禁制串列長度為 7 或大於 3 的方案上，有較好的求解結果，可能的原因在於其禁制串列 1 存有不少退化解，而較短的禁制串列可接受較多的退化解搜尋方向，故可增加搜尋範圍。

表 7 TTA 與 TGDA 各測試方案之系統參數值

方案 參數名稱	A	B	C	D	E	F	G	H	I	J
起始門檻值	0.5	0.5	0.5	0.5	0.5	0.5	0.39	0.39	0.39	0.39
門檻值數列長度	60	100	100	30	100	100	100	100	100	100
候選節線數	2n	n	n	2n	2n	n	n	n	n	n
未改善次數	30	10	50	50	0	0	50	50	50	50
回合數	0	0	0	0	200	100	0	0	0	0
禁制串列長度	7	7	7	7	7	7	3	4	5	6

表 8 TTA 測試結果之誤差百分比

	A	B	C	D	E	F	G	H	I	J
100_02	0.0%	2.2%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	1.5%	0.0%
100_05	0.6%	5.1%	3.7%	5.1%	3.3%	2.6%	3.8%	6.1%	3.7%	3.2%
100_08	0.4%	2.7%	3.7%	3.8%	3.8%	2.2%	0.0%	2.8%	1.4%	3.9%
150_02	2.8%	0.3%	0.0%	3.6%	1.6%	4.9%	0.3%	4.5%	2.2%	0.3%
150_05	5.8%	2.5%	4.3%	0.6%	3.2%	3.6%	0.8%	3.0%	3.3%	3.2%
150_08	3.0%	2.5%	2.0%	2.3%	0.5%	3.2%	1.3%	2.9%	0.6%	1.0%
平均	2.10%	2.55%	2.28%	2.57%	2.07%	2.75%	1.03%	3.22%	2.12%	2.10%

註：誤差百分比 (%) = (該參數下所得之目標值 / 本研究各演算法配合各參數所得最小目標值) - 1。

表 9 TGDA 測試結果之誤差百分比

	A	B	C	D	E	F	G	H	I	J
100_02	0.0%	1.5%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
100_05	5.4%	5.0%	0.0%	5.6%	5.1%	2.2%	0.3%	2.0%	0.0%	1.8%
100_08	1.9%	2.7%	3.1%	6.7%	5.3%	4.5%	4.4%	6.2%	1.1%	1.9%
150_02	0.3%	5.3%	0.3%	0.3%	0.3%	0.3%	3.6%	5.7%	0.3%	2.9%
150_05	3.2%	4.7%	4.0%	2.9%	0.0%	3.4%	2.1%	4.1%	3.1%	2.9%
150_08	2.4%	3.5%	0.2%	1.4%	0.7%	0.2%	1.0%	0.9%	3.2%	1.3%
平均	2.20%	3.78%	1.27%	2.82%	1.90%	1.77%	1.90%	3.15%	1.28%	2.20%

註：誤差百分比 (%) = (該參數下所得之目標值 / 本研究各演算法配合各參數所得最小目標值) - 1。

5.2.4 綜合結果分析

對於前述各區域搜尋求解方法的測試過程與結果，本研究歸納出以下特性：

1. 本研究針對具方向網路，配合節線選取與流量推擠還原伸展樹達成的節線交換策略，運用組合最佳化啟發解法來求解含凹形成本的最小網路流動問題，因有退化現象發生，再加入禁制方式控制後求解，結果有改善。
2. 對於以隨機候選節線中最佳的交換節線為進展依據的方式，每次進展時不宜以全部節線作為候選解。若每次進展測試太多節線可能耗費過多的時間在測試迴圈與流量推擠步驟。
3. 禁制串列不一定需設定太長，較短的禁制串列可能讓搜尋方向受到較少的限制，例如經測試結果，TTA 以禁制串列長度為 3 有較好的結果。
4. 將本研究提出的四種區域演算法 TA、GDA、TTA 與 TGDA 各自配合適用之方案，以

目標值誤差百分比顯示測試結果，並與初始解目標值誤差百分比一同列於表 10。其中以 TTA 配合方案 G 的效果最佳，平均誤差百分比最小為 1.03%。另外，TA 較 GDA 佳，TTA 較 TGDA 佳，TA 與 TGDA 相近。

5. 為進一步比較本研究發展的演算法，本研究利用 LINGO7.0 數學規劃軟體進行求解，求解結果列於表 10。注意，由於本研究問題為一 non-convex 非線性規劃問題，故存有多組區域最佳解，因此該軟體所求得的结果可能僅為一局部最佳解（此亦說明一般非線性規劃的最佳化軟體無法確保求得 non-convex 問題的最佳解）。結果顯示四種區域演算法的求解結果均較 LINGO 為佳，相較之下，LINGO 的平均相對誤差約為 10.52%。另外，LINGO 每次的求解時間約在 2~20 秒內，而本研究各演算法每次求解均可在 20~100 秒內完成，顯示就本研究所測試的問題規模，二者均具有良好的求解效率。雖 LINGO 的求解時間較短（以規劃問題而言，此等差異幾可忽略），但相對的目標值誤差太大，仍較不適合求解此 non-convex 問題。

表 10 較大型網路四種平均區域演算法求解結果

演算法	本研究各演算法 所得最小目標值	初始解	TA	TTA	GDA	TGDA	LINGO
方案	—	—	B	G	E	C	—
100_02	30848.7	10.0%	0.0%	0.0%	0.0%	0.0%	10.5%
100_05	47058.1	27.1%	2.5%	3.8%	4.0%	0.0%	10.1%
100_08	53664.0	30.8%	1.3%	0.0%	1.9%	3.1%	14.9%
150_02	42273.1	30.1%	0.3%	0.3%	1.4%	0.3%	11.2%
150_05	74286.7	32.2%	2.9%	0.8%	0.8%	4.0%	13.5%
150_08	78608.1	35.0%	0.8%	1.3%	1.8%	0.2%	9.1%
平均	54456.5	27.53%	1.30%	1.03%	1.65%	1.27%	11.55%

註：誤差百分比 (%) = (該參數下所得之目標值 / 本研究各演算法配合各參數所得最小目標值) - 1。

5.3 小型網路正確性測試

小型網路中 10_05 與 20_05 兩個網路經人工求證，本研所得最小目標值皆為此兩個網路的最佳解目標值，其中，在 20_05 網路中，LINGO 恰可求得最佳解，如表 11 所示。另外，各區域演算法的求解結果亦均較 LINGO 為佳，顯示本研究發展的鄰近搜尋法在求最佳解上，初步效果頗佳。

表 11 小型網路四種區域演算法求解結果

演算法	本研究各演算法 所得最小目標值	初始解	TA	TTA	GDA	TGDA	LINGO
方案	—	—	B	G	E	C	—
10_05	4639.6	0.4%	0.0%	0.0%	0.0%	0.0%	19.3%
20_05	19756.8	0.9%	0.0%	0.0%	0.0%	0.0%	0.0%
50_02	12883.0	7.5%	0.0%	0.0%	0.0%	0.0%	3.0%
50_05	25733.1	14.7%	3.5%	0.0%	4.1%	3.5%	3.5%
50_08	38494.7	24.5%	1.8%	3.0%	0.0%	1.8%	1.8%
平均	20301.44	9.60%	1.06%	0.60%	0.82%	1.06%	5.52%

註：方案 10_05 與 20_05 之誤差百分比 (%) = (該參數下所得之目標值 / 最佳目標值) - 1。

其餘方案的誤差百分比 (%) = (該參數下所得之目標值 / 本研究各演算法配合各參數所得最小目標值) - 1。

六、結論與建議

6.1 結論

傳統上在考量運輸成本時，為求簡便，大多以線性成本來簡化模式；然而實際上貨物運送成本常隨數量的增加而享有遞減的單位運送成本，亦即具運送經濟規模特性，因此各起迄運送成本常隨運送數量而成凹形狀 (concave)。本研究的目的是針對含凹形節線成本的轉運問題，發展有效率的解法，以成本最小的方式，透過轉運，將所有供給點的供給量運送至所有需求點上。

本研究先從數學觀點探討凹形節線成本之運輸問題與組合最佳化問題的關連性，依凹形節線成本特性發展有效的起始解法，再參考新近鄰近搜尋法中的門檻值接受法與大洪水法以及禁制搜尋策略，發展 TA、GDA、TTA、TGDA 等四個適用本問題的改善解法，以求解含凹形節線成本最小成本網路流動問題。為測試各演算法的績效，本研究另設計一隨機網路產生器，產生大量網路資訊進行測試，結果良好。本研究綜合所有測試可得下列幾個較重要的結論：

1. 本研究問題雖然不是一般整數組合最佳化問題，但在最佳解恰為端點解及整數解的特性下，恰可利用組合最佳化的啟發式解法，配合電腦快速運算，使用新近有效的鄰近搜尋法，如 TA 及 GDA，以有效率找出各端點解，進而求得近似最佳解。
2. 由於網路問題在進展過程中常會發生退化現象，因此利用鄰近搜尋法求解時，最好加入防止退化的機制，以減少不必要的測試進展。如本研究在 TA 及 GDA 中加入防止退化的禁制搜尋策略，設計出 TTA 及 TGDA，經測試顯示其求解品質的確均較 TA 及 GDA 為佳。其中，又以 TTA 的效果最好。另外，各演算法與 LINGO 數學規劃軟體的測試結

果相比較，各演算法的求解時間雖稍較 LINGO 為長，但各演算法的求解品質皆明顯較 LINGO 為佳，且求解仍具效率。顯示在現有的解法中，本研究發展的演算法效果頗佳。

3. 隨著網路節點數規模的增加或隨著供需節點密度的增加，各演算法 CPU 時間有增加的趨勢，但 CPU 時間約和節線數或供需節點密度呈線性遞增的關係，表示求解時間不會隨問題規模大幅度的呈指數成長。以本研究測試的問題規模而言，測試效率均甚佳，即使以較大型網路而言，各演算法每次求解均可在 20 ~ 100 秒內完成。求解的品質上，以小網路的正確性測試顯示，本研究的鄰近搜尋法均可找到最佳解。

6.2 建議

本研究對於未來可再繼續研究的方向，建議如下：

1. 對於本研究的假設，目標函數為 $c_{ij}\sqrt{x_{ij}}$ 類型的凹形成本函數，或節線流量無上限等，皆可在未來放鬆而進行更深入的探討。其中，對於其他形式之凹形成本函數，應可以類似本研究發展之演算法進行求解，然而適用之系統參數則可能不同，需要再進一步測試。另外，考量有節線流量上限之網路，在可行伸展樹的構建上，需要做適當的修正，而此將會影響起始解與改善解的設計，至於具體的作法可在未來繼續探討。
2. 本研究以隨機網路產生器來產生不同規模的網路，其中，小型網路經由人工運算實證，確實可以此求得最佳解；而對其他較大型的網路，目前僅由本研究所提的各演算法以及 LINGO 數學規劃軟體相比較求解結果，未來可引用其他的求解方法（如分枝界限法）求上下限解，以驗證本研究演算法的求解績效。
3. 本研究利用 TA 與 TGA 並結合 TS 策略發展四個鄰近搜尋法進行求解，未來可繼續發展其他組合最佳化方法，如 TS 或 GA，以進行求解並比較其求解績效的差異。

參考文獻

1. Yan, S. and Luo, S. C., "A Tabu Search-Based Algorithm for Concave Cost Transportation Network Problems", *Journal of the Chinese Institute of Engineers*, Vol. 21, 1998, pp. 327-335.
2. Yan, S. and Luo, S. C., "Probabilistic Local Search Algorithms for Concave Cost Transportation Network Problems", *European Journal of Operational Research*, Vol. 117, 1999, pp. 511-521.
3. Ahuja, R. K., Maganti, T. L., and Orlin, J. B., *Network Flows, Theory, Algorithms, and Applications*, Prentice Hall, Englewood Cliffs, 1993.
4. Garey, M. R. and Johnson, D. S., *Computers and Intractability: A Guide to the Theory of NP-Completeness*, WH Freeman and Co., San Francisco, 1979.
5. Larsson, T., Migdalas, A., and Ronnqvist, M., "A Lagrange an Heuristic for the Capacitated Concave Minimum Cost Network Flow Problem", *European Journal of Operational Research*, Vol. 78, 1994, pp. 116-129.
6. Sheffi, M. J., *Urban Transportation Networks: Equilibrium Analysis with Mathematical*

- Programming Methods*, Prentice-Hall, 1984.
7. Zangwill, W. I., "Minimum Concave Cost Flows in Certain Networks", *Management Science*, Vol. 14, 1968, pp. 429-450.
 8. Gallo, G. and Sandi, C., "Adjacent Extreme Flows and Application to Min Concave Cost Flow Problems", *Networks*, Vol. 9, 1979, pp. 95-121.
 9. Rech, P. and Barton, L. G., "A Non-Convex Transportation Algorithm", E. M. Beale, Ed., *Applications of Mathematical Programming Techniques*, 1970.
 10. Gallo, G., Sandi, C., and Sodini, C., "An Algorithm for the Min Concave Cost Flow Problem", *European Journal of Operation Research*, Vol. 4, 1980, pp. 248-255.
 11. Jordan, W. C., "Scale Economies on Multi-Commodity Networks", GMR-5579, Operating Systems Research Dept., GM Research Laboratories, 1986.
 12. Blumenfeld, D. E., Burns, L. D., Diltz, J. D., and Daganzo, C. F., "Analyzing Trade-Offs between Transportation, Inventory, and Production Costs on Freight Network", *Transportation Research*, Vol. 19B, 1985, pp. 361-380.
 13. Hall, R. W., "Direct Versus Terminal Freight Routing on Network with Concave Costs", GMR-4517, Transportation Research Dept., GM Research Laboratories, 1983.
 14. Thach, P. T., "A Decomposition Method Using a Pricing Mechanism for Min Concave Cost Flow Problems with a Hierarchical Structure", *mathematical Programming*, Vol. 53, 1992, pp. 339-359.
 15. Guisewite, G. M. and Pardalos, P. M., "A Polynomial Time Solvable Concave Network Flow Problems", *Networks*, Vol. 23, 1993, pp. 143-147.
 16. Yaged, B., "Minimum Cost Routing for Static Network Models", *Networks*, Vol. 1, 1971, pp. 139-172.
 17. Balakrishnan, A. and Graves, S. C., "A Composite Algorithm for a Concave-Cost Network Flow Problem", *Networks*, Vol. 19, 1989, pp. 175-202.
 18. Amiri, A. and Pirkul, H., "New Formulation and Relaxation to Solve a Concave Cost Network Flow Problem", *Journal of the Operational Research Society*, Vol. 48, 1997, pp. 278-287.
 19. Nourie, F. J. and Guder, F., "A Restricted-Entry Method for a Transportation Problem with Piecewise-Linear Concave Cost", *Computer & Operations Research*, Vol. 21, 1994, pp. 723-733.
 20. Dukwon, K. and Panos, M., "Dynamic Slope Scaling and Trust Interval Techniques for Solving Concave Piecewise Linear Network Flow Problems", *Networks*, Vol. 35, 2000, pp. 216-222.
 21. Suwan, R. and Sawased, T., "Link Capacity Assignment in Packet-Switched Networks: The Case of Piecewise Linear Concave Cost Function", *IEICE Trans. Commun.*, Vol. E82-B, No. 10, 1999.
 22. Kuhn, H. W. and Baumol, W. J., "An Approximate Algorithm for the Fixed-Charge Transportation Problem", *Naval Res. Logistics Quarterly*, Vol. 9, 1962, pp. 1-16.
 23. Glover, F. and Laguna, M., *Tabu Search*, Kluwer Academic Publishers, Massachusetts, 1997.
 24. Osman, I. H. and Kelly, J. P., "Meta-Heuristics: An Overview", *Meta-Heuristics: Theory & Applications*, Kluwer Academic Publishers, Boston, London, Dordrecht, 1996, pp. 1-21.

25. Glover, F., "Tabu Search, Part I", *ORSA Journal on Computing*, Vol. 1, No. 3, 1989, pp. 190-206.
26. Glover, F., "Tabu Search, Part II", *ORSA Journal on Computing*, Vol. 2, No. 1, 1990, pp. 4-32.
27. Kirkpatrick, S., Gelatt, C. D., and Vecchi, M. P., "Optimization by Simulated Annealing", *Science*, Vol. 220, 1983, pp. 671-680.
28. Dueck, G. and Scheuer, T., "Threshold Accepting: A General Purpose Optimization Algorithm Appearing Superior to Simulated Annealing", *Journal of Computational Physics*, Vol. 90, 1990, pp. 161-175.
29. Goldberg, D. E., *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, Reading MA, 1989.
30. Dueck, G., "New Optimization Heuristics: The Great Deluge Algorithm and the Record-to-Record Travel", *Journal of Computational Physics*, Vol. 104, 1993, pp. 86-92.
31. Charon, I. and Hurdy, O., "The Noising Method: A New Method for Combinatorial Optimization", *Operations Research Letters*, Vol. 14, 1993, pp. 133-137.
32. 韓復華、卓裕仁，「門檻接受法、成本擾動法與搜尋空間平滑法在車輛路線問題之應用研究與比較分析」，**運輸學刊**，第九卷，第三期，民國八十五年，頁 103-129。
33. 韓復華、陳國清、卓裕仁，「成本擾動法在 TSP 問題之應用」，**中華民國第二屆運輸網路研討會論文集**，民國八十六年，頁 283-292。
34. 韓復華、楊智凱、卓裕仁，「應用門檻接受法求解車輛路線問題之研究」，**運輸計劃季刊**，第二十六卷，第二期，民國八十六年，頁 253-280。
35. Gu, J. and Huang, X., "Efficient Local Search with Search Space Smoothing: A Case Study of the Traveling Salesman Problem (TSP)", *IEEE Transaction on Systems, Man and Cybernetics*, Vol. 24, 1994, pp. 728-739.
36. Reeves, C. R., "Improving the Efficiency of Tabu Search for Machine Sequencing Problems", *Journal of the Operation Research Society*, Vol. 44, No. 4, 1993, pp. 375-382.
37. Booker, L. B., "Improving Search in Genetic Algorithms", L. Davis, Ed., *Genetic Algorithms and Simulated Annealing*, Pitman, London, 1987, pp. 61-73.
38. 李其澧，「中小型交通建設評估模式之研究」，國立中央大學土木工程研究所碩士論文，民國八十九年六月。
39. 韓復華、楊智凱，「門檻接受法在 TSP 問題上之應用」，**運輸計劃季刊**，第二十五卷，第二期，民國八十五年，頁 163-188。
40. 韓復華、林修竹，「TA 與 GDA 巨集啟發式法在 VRPTW 問題上之應用」，**中華民國第四屆運輸網路研討會**，民國八十八年，頁 83-92。
41. 顏上堯、周榮昌、李其澧，「交通建設計畫評選模式及其解法之研究—以中小型交通建設計畫的評選為例」，**運輸計劃季刊**，第三十一卷，第一期，民國九十一年，頁 143-170。
42. Golden, B. L. and Skiscim, C. C., "Using Simulated Annealing to Solve Routing and Location Problems", *Naval Research Logistic Quarterly*, Vol. 33, 1986, pp. 261-279.

43. Robuste, F., Daganzo, C. F., and Souleyrette, R., "Implementing Vehicle Routing Models", *Transportation Research*, Vol. 24B, No. 4, 1990, pp. 263-286.
44. Alfa, A. S., Heragu, S. S., and Chen, M., "A 3-opt Based Simulated Annealing Algorithm for Vehicle Routing Problem", *Computers and Industrial Engineering*, Vol. 21, 1991, pp. 635-639.
45. 詹達穎, 「模擬鍛鍊法求解車輛排程之探討」, **中華民國運輸學會第九屆論文研討會論文集**, 民國八十三年, 頁 185-192。
46. Sinclair. M., "Comparision of the Performance of Modern Heuristic for Combinatorial Optimization on Real Data", *Computer & Operations Research*, Vol. 20, No. 7, 1992, pp. 687-695.
47. Hu, N., "Tabu Search Method with Random Moves for Globally Optimal Design", *International Journal for Numerical Methods in Engineering*, Vol. 35, 1992, pp. 1055-1070.
48. 羅守正, 「含凹形節線成本運輸問題啟發式解法之研究」, 國立中央大學土木工程研究所碩士論文, 民國八十五年六月。
49. Yan, S. and Young, H. F., "A Decision Support Framework for Multi-Fleet Routing and Multi-Stop Flight Scheduling", *Transportation Research*, Vol. 30A, 1996, pp. 379-398.
50. Powell, W. B., "A Review of Sensitivity Results for Linear Networks and a New Approximation to Reduce the Effects of Degeneracy", *Transportation Science*, Vol. 26, No. 3, 1992, pp. 230-245.
51. Demeulemeester, E., Dodin, B., and Herroelen, W., "A Random Activity Network Generator", *Operations Research*, Vol. 41, No. 5, 1993, pp. 972-980.
52. Klingman, D., Gelatt, C. D., and Stutze, J., "NETGEN: A Program for Generating Large Scale Capacitated Assignment, Transportation, and Minimum Cost Flow Network Problem", *Management Science*, Vol. 20, 1974, pp. 814-821.