

第四章 通勤交通車路線問題啟發式解法構建

由於通勤交通車問題屬於大規模數學規劃問題，求解小型問題尚可於有效時間內求解，但對於大規模之問題則無法於有效率的時間內求解出最佳解，故本章將根據通勤交通車之特性，以總營運成本最小為目標構建一啟發式解法求解大規模之問題，以便實務上之應用。第一節中將說明啟發式解法之架構，第二節說明路線起始解之建構方法，第三節則說明路線改善之方法，第四節則進行啟發式解法成效之評估。

4.1 啟發式解法之架構

本研究所探討之問題屬於多對多起迄點之路線問題，同一起點至不同迄點之需求視為不同之需求，與多場站車輛路線問題之性質不同，故不屬於多場站車輛路線問題。因此，本研究針對多對多起迄點及多車種之通勤交通車路線問題提出一啟發式解法，期能有效率地求解。本研究首先建構路線起始解，接著進行路線改善，以求得較佳之結果。整個啟發式解法執行架構如圖 4.1。

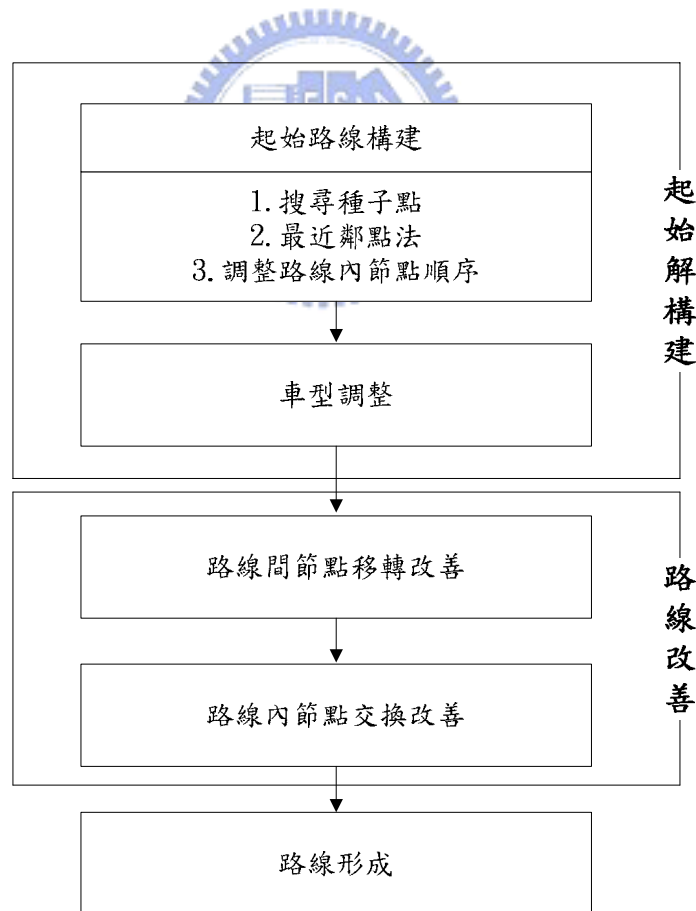


圖 4.1 啟發式解法執行架構

1. 起始解構建

起始解之構建分為二個部分，第一個部份進行起始路線構建，先搜尋種子點，並以最近鄰點法為基礎將各節點插入路線中；第二個部份則針對各路線實際乘載人數進行車型調整(縮小)，使得各路線都能利用最適合其乘載人數之車種型態進行服務，避免成本之浪費。

2. 路線改善

起始解構建完成後，進行路線改善。首先，考量起始解之各路線剩餘時間與剩餘容量，對起始解進行路線間節點移轉，以改善路線，降低成本。路線間節點移轉改善後，必須再次確認路線內節點順序，以確保經過路線內各節點所需之距離最短，故進行路線內節點交換改善。

4.2 路線起始解模組構建

本研究以總營運成本最小為目標規劃啟發式解法，而影響營運成本最重要之關鍵為車輛之固定成本，每增加使用一輛車，營運成本就必須增加一固定成本，因此，在滿足車容量限制與路線時間限制下，若使用愈少之車輛，則總營運成本會愈小。故本研究先以最大車型之車容量作為車容量限制，儘量降低車輛數，然後再進行車輛型態之調整。而路線構建方式則以搜尋種子點及最近鄰點法來執行。

1. 路線構建

本研究所探討之問題雖屬於多對多起迄點之路線問題，但因以最小成本為目標，路線數愈少則固定成本愈少，且路線無重疊則可減少路線之變動成本，故在起始解之路線構建時，先考量每個節點經過一次，將同一起點至不同迄點之需求指派至同一路線上。其執行架構如圖 4.2，執行步驟說明如下：

步驟一：選擇起點與迄點間路徑距離最長之迄點 D_1 當作搜尋種子點之迄點。

步驟二：搜尋離迄點最遠之未指派節點為種子點，並以種子點當作路線 k 之起點。

步驟三：搜尋離種子點最近之節點 i ，檢查節點 i 插入路線 k 是否滿足路線時間限制與最大車型之車容量限制。若是，則將節點 i 插入路線 k ，並將種子點更新為 i ；若否，則回到步驟一。

步驟四：檢查是否有未指派之節點。若有，則回到步驟二；若無，則進行下一步驟。

步驟五：將其他迄點選擇距離增加最小之位置加入至各路線中。

步驟六：分別檢查各路線之距離是否為最短，若是，則路線構建完成；若否，則須先調整路線中節點順序，路線才算路線構建完成。

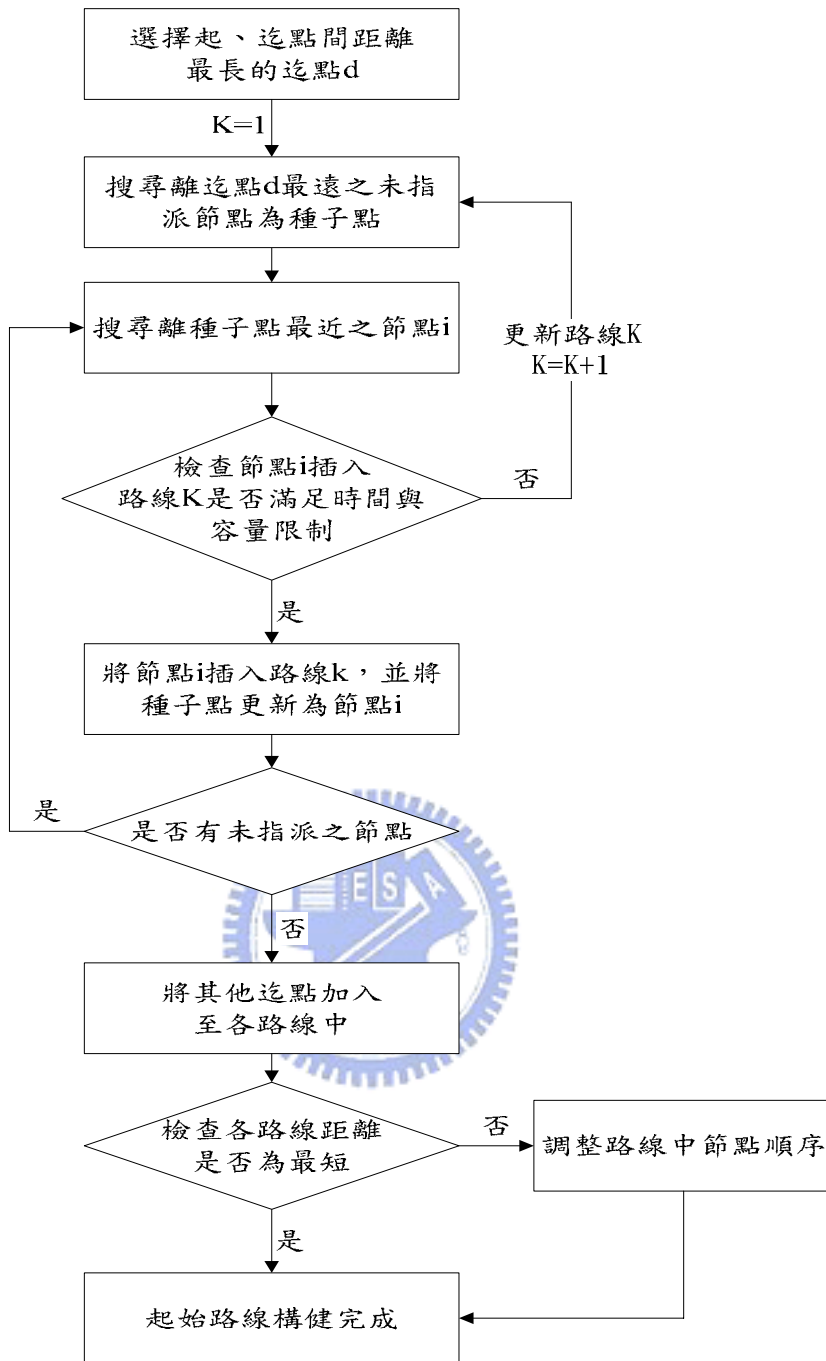


圖 4.2 起始路線構建執行架構

因本研究有多個迄點，故本研究在搜尋種子點前，須先決定以哪一個迄點為主進行種子點搜尋。本研究是以起點與迄點間路徑距離最長之迄點當作搜尋種子點之迄點。而路線一開始之種子點則是以距離該迄點最遠之未指派起點當作種子點，再以此種子點當作路線之起點，並搜尋離種子點最近之節點插入路線中，直至路線時間超過限制或乘載人數超過最大車型之車容量為止。不斷反覆搜尋種子點及最近鄰點插入，直至所有節點皆指派至路線上後，則將其他迄點選擇距離增加最小之位置插入。

因起始路線構建開始時只以搜尋種子點之迄點為考量，直至最後才將其他迄點加入

路線中，故在進行最近鄰點插入時，需將路線時間限制作調整，以預留時間讓其他迄點加入。路線時間限制之調整值為搜尋種子點之迄點行經其他所有迄點所需之最短時間。例如：原本之路線時間限制為 60 分鐘，搜尋種子點之迄點為 a，其他仍有二個迄點 b 與 c，迄點 a 行經迄點 b 與 c 之最短路徑為 a→c→b，所需之旅行時間為 10 分鐘，故路線時間限制則減少 10 分鐘，調整為 50 分鐘，亦即若最近鄰點插入某一路線會超過 50 分鐘的話，則該節點不能插入該路線中。

在其他迄點皆加入各路線後，需要針對每條路線進行檢查，進而調整路線內之節點順序，以確保經過該路線內之節點所需之距離為最短。因為通勤交通車之路線屬於起點至迄點單方向之路徑(path)型態，且先前路線構建之方法係從最遠點尋找其最近鄰點插入，故本研究按照路線節點順序，從路線起點開始，將此節點與其下一節點交換，檢查交換後的路線距離是否小於交換前的路線距離，若是，則更新路線；否則路線維持原解。持續進行節點之交換，直至所有節點與其下一節點都交換檢查完畢為止。

2. 車型調整(縮小)

先前路線構建時，各路線車容量限制皆以最大車型之車容量作為限制依據，故所有路線皆使用最大車容量之車輛型態，而此階段最主要之目的是考量不同車型有不同之車容量，依據各路線之乘載人數，選擇最適合之車型，以增加各路線之車輛乘載率，避免成本浪費。



4.3 路線改善

由於本研究係探討多車種之通勤交通車路線問題，不同之車型有不同之固定成本及不同之變動成本係數，故路線使用之車種型態對於總營運成本最小之目標影響甚大，若能在路線數固定之情況下，使各路線儘量使用較小型之車輛型態，則能有效降低營運成本。有鑑於此，本節路線改善最主要之目的是在不改變其他路線使用之車種型態下，透過路線間節點移轉之動作，將部分需求移轉至其他路線，一方面使該路線可使用較小之車輛型態，而減少車輛之固定成本；另一方面亦可增加其他路線之乘載率，避免座位之閒置。

1. 路線間節點移轉改善

本階段主要針對路線之剩餘車容量來考量可否進行節點之移轉，而使部分路線可以從大型車種改變成小型車種，以減少成本。其主要概念說明如下，並以圖 4.3 說明剩餘容量與可移轉量之定義：

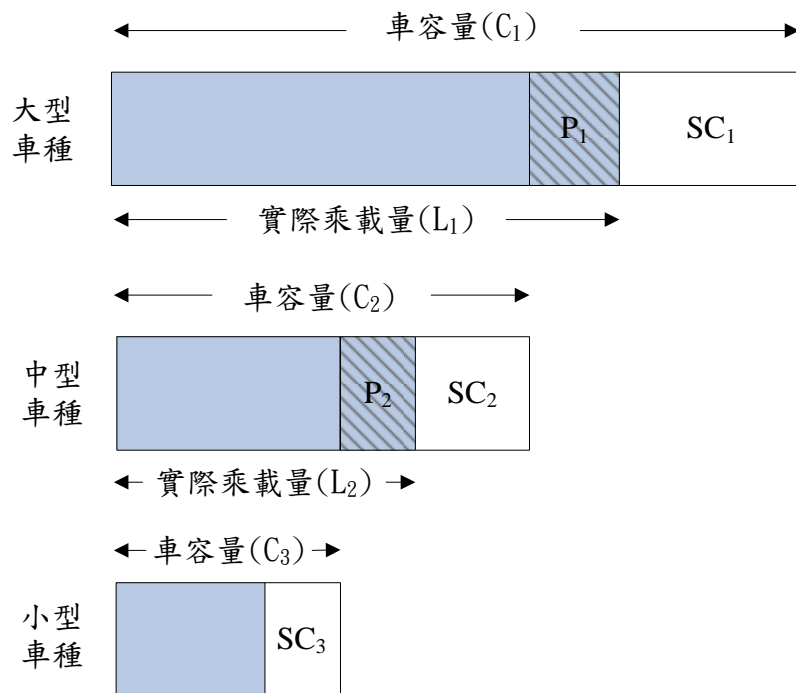


圖 4.3 路線剩餘容量與可移轉量示意圖

在圖 4.3 中，三個長方形代表三種車種型態，有顏色的部份代表車輛乘載人數，白色的部份代表車輛剩下之容量 SC ，有顏色之斜線部份代表較大車種若要換成較小車種需要移轉之人數 P 。在起始解構建完成後，每條路線即可計算出該路線之 SC ，使用大型車種之路線可將實際乘載量減去中型車種之車容量計算出該路線之 P ，即 $P_1 = L_1 - C_2$ ，使用中型車種之路線亦可將實際乘載量減去小型車種之車容量計算出該路線之 P ，即 $P_2 = L_2 - C_3$ ；接著根據各路線 SC 與 P 進行節點移轉之檢查與移轉。例如：假定大型車輛之車容量有 40 人，中型車輛之車容量有 25 人，路線 1 使用大型車輛，乘載人數共有 30 人，則路線 1 之剩餘容量 SC 有 10 個座位，可移轉量 P 則有 5 人，故若路線 1 可將 5 人移轉至其他路線，則可將路線 1 使用之車輛型態更改為中型車輛。路線間節點移轉改善之流程圖如圖 4.4，執行步驟說明如下：

步驟一：計算起始解中每條路線之剩餘車容量 SC 、剩餘路線時間 ST 與可移轉量 P 。

步驟二：計算各路線將其 P 移轉至其他路線所能減少之車輛固定成本 Δf ，選擇 $\Delta f/P$ 最大之路線 k 開始。

步驟三：檢查路線 k 可移轉量 P 是否小於其他路線之剩餘車容量總和。若是，則進行下一步驟；若否，則跳至步驟九。

步驟四：選擇路線 k 離迄點最近之起點 j 進行下列步驟。

步驟五：檢查路線 k 以外之路線是否皆檢查完畢。若是，則 $j=j+1$ ，並回到步驟四；

若否，則選擇最靠近 j 之路線 l 。

步驟六：檢查路線 l 之剩餘車容量 SC 是否大於 0。若是，則進行下一步驟；若否，則回到步驟五。

步驟七：檢查將 j 點插入路線 l 是否滿足路線 l 之剩餘車容量 SC 與剩餘路線時間 ST 。若是，則將 j 點選擇距離增加最少之位置插入，並更新路線 l 之 SC 、 ST 與 P ，以及路線 kl 之 SC 、 ST 與 P ；若否，則回到步驟五。

步驟八：檢查路線 k 之可移轉量 P 是否小於或等於 0。若是，則將路線 k 之使用車輛變更為較小車型之車輛；若否，則 $j=j+1$ ，並回到步驟四。

步驟九：檢查所有路線是否皆檢查完畢。若是，路線間節點移轉完成；若否，則回到步驟二。

因本研究係研究多對多起迄點之通勤交通車路線問題，每一個起、迄對間之需求是獨立的，故本研究對於路線間節點交換之程序執行二次，第一次先將同一起點至不同迄點之需求同時考量，亦即節點移轉時之移轉人數為該節點至所有迄點之需求總和；第二次進行路線間節點交換時，則將同一起點至不同迄點之需求分開考量，亦即先檢查至某一迄點之需求可否移轉，若是，則進行節點移轉；若否則檢查至另一迄點之需求可否移轉。

2. 路線內節點交換改善

節點移轉改善完成後，需要再針對每條路線進行檢查，進而調整路線內之節點順序，以確保該路線內經過之節點所需距離為最短。因為通勤交通車之路線屬於起點至迄點單方向之路徑(path)型態，故本研究按照路線節點順序，從路線起點開始，將此節點與其下一節點交換，檢查交換後的路線距離是否小於交換前的路線距離，若是，則更新路線；否則路線維持原解。持續進行節點之交換，直至所有節點與其下一節點都交換檢查完畢為止。

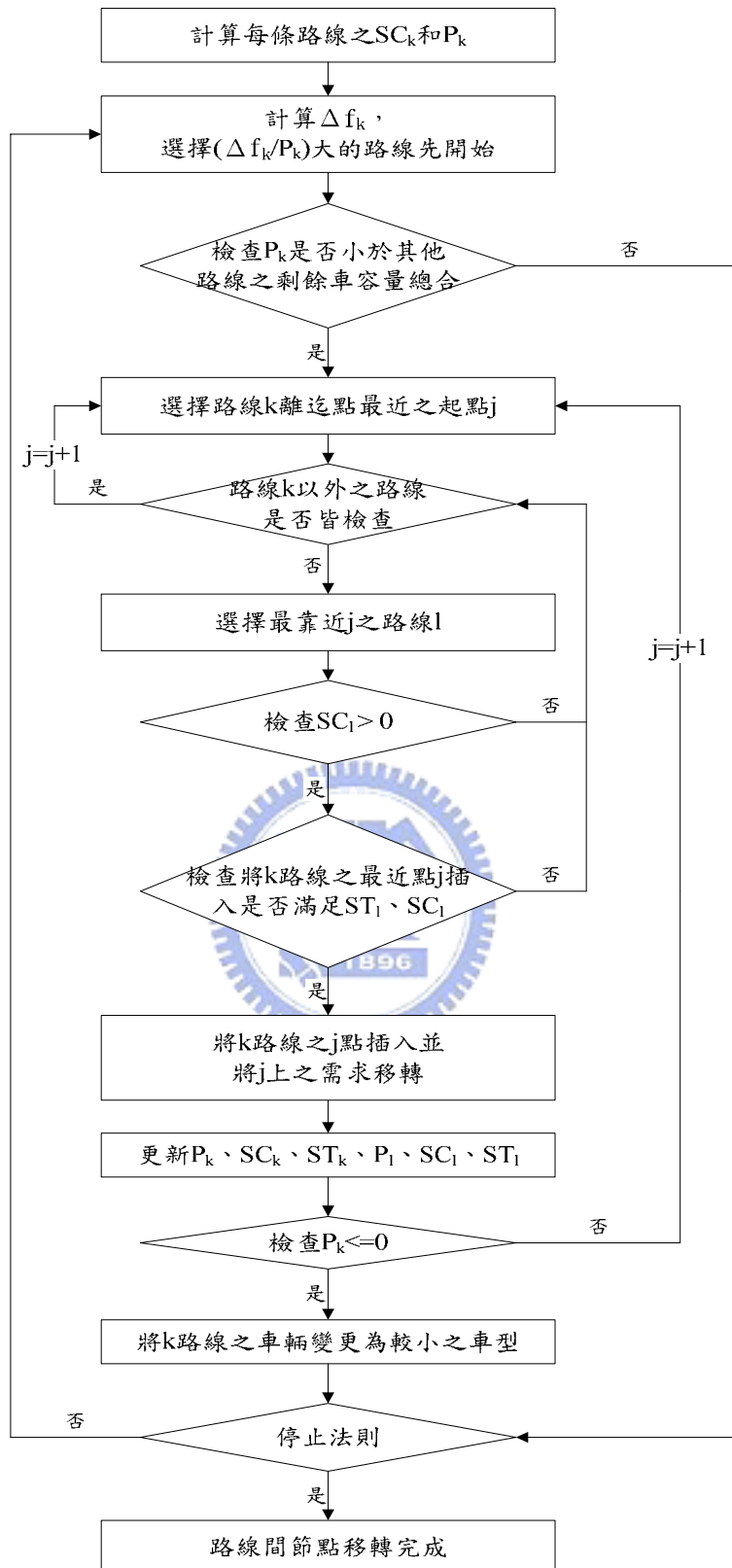


圖 4.4 路線間節點移轉改善流程圖

4.4 啟發式解發之測試與分析

由於國、內外較少有學者進行類似研究，本研究所提出之啟發式解法之成效不易評估，故本研究針對第三章設計之小型例題，分別利用第三章之數學模式及本章之啟發式解法求解並比較二方法求解結果，以作為評估啟發式解法優劣之依據。二者分別利用 ILOG OPL Development Studio 4.2 軟體及撰寫 C++ 程式於 Windows XP 作業系統、Pentium IV 2.8GHz CPU 及 1GHz RAM 之個人電腦求解。茲將二種方法求解結果整理如表 4.1，詳細結果參照附錄二。

表 4.1 啟發式解法求解結果與最佳解之比較

例題標號	目標值		求解時間	
	最佳解	啟發式求解	最佳解	啟發式求解
CC1	1752.7	1752.7	1 小時 14 分 15 秒	0.16 秒
CC2	1752.7	1752.7	2 小時 23 分 16 秒	0.14 秒
CC3	1524.5	1524.5	6 分 26 秒	0.14 秒
CC4	1524.5	1524.5	7 分 49 秒	0.14 秒
CC5	1705.4	1705.4	3 分 1 秒	0.1 秒
CC6	1705.4	1705.4	6 小時 3 分 59 秒	0.15 秒
CC7	2328.2	2328.2	2 天 18 小時 6 分 25 秒	0.15 秒
CC8	1791	1791	21 分 53 秒	0.17 秒
CR1	1683.7	1683.7	2 小時 15 分 39 秒	0.15 秒
CR2	1683.7	1683.7	6 小時 13 分 6 秒	0.17 秒
CR3	1599.8	1599.8	21 小時 28 分 33 秒	0.11 秒
CR4	1355	1355	44 分 09 秒	0.12 秒
CR5	1772.4	1772.4	19 分 54 秒	0.1 秒
CR6	1772.4	1772.4	10 小時 28 分 08 秒	0.12 秒
IC1	1677.45	1677.45	15 分	0.21 秒
IC2	1677.45	1677.45	6 分 26 秒	0.29 秒
IC3	1617.47	1617.47	11 分 57 秒	0.29 秒
IC4	1617.47	1617.47	16 分 38 秒	0.21 秒
IC5	1959.32	1959.32	17 分 19 秒	0.37 秒
IC6	1870.33	1870.33	18 分 14 秒	0.33 秒
IC7	1870.33	1870.33	29 分 37 秒	0.31 秒
IR1	1682.05	1682.05	10 分 5 秒	0.21 秒
IR2	1586.19	1586.19	1 小時 11 分 42 秒	0.23 秒
IR3	1586.19	1586.19	41 分 27 秒	0.31 秒

從表 4.2 中可以看出本研究之啟發式解法求解小型例題時，皆可以求解出最佳解，且求解時間皆在 1 秒內；反觀整數規劃之求解效率則低許多，例題 CC7 之求解時間甚至超過 2 天。上述之求解結果顯示本研究提出之啟發式解法對於通勤交通車問題可以有效率地求解，且對於小型問題皆可求解出最佳解，故此啟發式解法之成效良好。

