

捷運司機員排班問題之研究

—以台北捷運公司為例

The Study of Mass Rapid Transit Drivers' Scheduling Problems A Case Study of Taipei MRT Company

王晉元¹ 盧宗成²

Abstract

This paper proposes an algorithm to improve the efficiency for solving the drivers scheduling problem of the Taipei MRT company. We formulate this problem as a set-partitioning problem. A three phases algorithm, column generation, branch-and-bound, and heuristic local search, is then developed for solving this model.

We use the real cases of the Taipei MRT company as the numerical examples. The results show that our algorithm could get better solutions with higher operational feasibility than the current one does.

一、前言

在解決都會區交通問題上，發展大眾運輸是一般認為較有效的方法。在各種大眾運輸工具中，大眾捷運系統由於其具有運量大、快速、舒適、安全及對環境污染較小等特性，世界各先進國家紛紛趨向以捷運系統來疏解都市交通問題。台北市自民國八十六年起有數條捷運線相繼通車，至今整個台北捷運系統初期路網已略具雛形，對於改善大台北都會區的交通將有相當的幫助。

目前，對於已完成的捷運系統路網，乃是由台北捷運公司負責實際的營運管理。捷運公司運務部根據法定的營運班距產生列車發車時刻表(Timetable)，列車司機員的排班則是依據列車發車時刻表以人工排班方式產生司機員每日執勤任務卡。然而司機員排班的工作相當繁瑣，除了必須配合時刻表執行所有任務之外，更必須兼顧司機員基本需求（休息與用餐）、及相關法規之工作時數限制。

因此，每當捷運公司必須調整營運班距及列車發車時刻表時，亦將影響到司機員任務排班表的製定，通常捷運公司須指派專人花費一個月的時間去重新產生司機員每日任務排班表。由此可見，捷運公司在司機員排班所投入的成本實在相當可觀。本研究針對台北捷運公司司機員排班問題建構數學模式，並設計一求解演算法，以期能有效提升台北捷運公司司機員排班效率。

1. 王晉元。國立交通大學運輸工程與管理學系 副教授。

2. 盧宗成。國立交通大學運輸工程與管理學系 碩士班研究生。

在本研究中，定義車旅次(vehicle block)為單一車輛(列車)所執行之最小工作量，通常為車輛從某一場站出發至另一場站的旅次，或是回到同一場站的旅次；任務(task)為司機員在同一車旅次內所執行之最小工作量，其端點為換班點(relief point)或是場站；任務鍊(piece)為司機員在同一車旅次內連續服務的一組任務；工作班(workday)或任務卡為單一司機員每日所服務的一組任務鍊。

二、文獻回顧

人員排班問題通常以 0-1 整數規劃問題來加以描述，此類問題屬於 NP-Hard 問題[5]。目前有關人員排班問題的文獻中，求解此類問題通常以在合理時間內找到較佳可行解之啟發式解法為主，最佳解演算法通常用於規模較小的問題上。不過，近來由於計算機技術的進步，在許多較不具即時性之人員排班問題上，最佳解演算法，或是以最佳解為基礎之演算法的應用亦愈趨廣泛。

2.1 啟發式解法 (Heuristics Method)

Run-cutting 啟發式解法(Wren et al.[20])綜合了構建式與改善式啟發式解法，其求解過程可分為兩階段。在第一階段中構建一起始可行解，將車旅次切割成許多任務鍊，並將二或三個任務鍊組合成一個可行的工作班。在第二階段中進行交換法改善，試著在第一階段所產生的工作班間進行任務鍊的交換動作，如果交換後成本有所改善則進行交換，否則不交換，並繼續進行交換，直到都無法改善為止。

HASTUS 啟發式解法(Lessard et al[14], Rousseau et al[17])亦綜合構建式與改善式啟發式解法，求解過程可分為三階段。在第一階段中，利用線性規劃(LP)放鬆人員排班問題之整數解限制，以獲得一近似排班表(an approximate schedule)。在第二階段中，切割車旅次成許多任務鍊，利用配對(matching)的方法將切割出來的任務鍊組合成工作班。在第三階段中，重新考慮切割車旅次成任務鍊，重新組合，並進行區域改善(local improvement)，在工作班中進行任務鍊的交換。

其它大眾運輸人員排班問題解法尚有：Ball、Bodin and Dial[3]應用配對(matching)的方法將車輛及人員的排班問題同時解決。此外，人工智慧型啟發式解法為近來較流行的方法，其觀念為加入人工智慧的特性，使可行解的搜尋過程有機會能跳脫區域最佳解(Local Optimal)的情況。Levine[13]發展出 Hybrid Genetic Algorithm 用來求解航空公司人員排班問題，其概念主要是結合了基因法與區域搜尋法兩種方法。一旦利用基因法在可接受的時間內找到一個較佳解後，再利用區域搜尋法來找尋更好的區域最佳解。文獻上其它較常被使用的人工智慧方法尚有則有禁制搜尋法(Tabu Search, TA)[23]以及模擬退火法(Simulated Annealing, SA)[2]等。

2.2 最佳解解法 (Exact Solution Method)

目前有關大眾運輸人員排班問題最佳解解法的研究，在作法上通常將問題構建成集合涵蓋(Set Covering)或集合分割(Set Partitioning)型式，並配合最佳解的求解策略，我們稱這種方法為集合涵蓋(或集合分割)法。至於在配合集合涵蓋(或集合分割)問題的最佳解求解方法，最常用的是分支定限法(Branch and Bound)。

文獻中利用集合涵蓋(或集合分割)法求取人員排班問題最佳解的例子非常多。舉例如下：Heurgon[12]利用線性規劃(LP)與分支定限法求解集合分割問題。Shepardson and Marsten[18]利用拉式鬆弛法(Lagrangian relaxation)與分支定限法求解集合分割問題。Mittra and Darby-Dowman[16]修正集合分割模式，允許任務的 over(或 under)－covering 情形，但須加上一懲罰值(penalty)。Wren et al[20]利用線性規劃與 Ryan and Foster 所發展的複雜的分支定限法去求解集合涵蓋問題。

2.3 變數產生法 (Column Generation)

在利用集合涵蓋(或集合分割)方法求解人員排班問題時，如果在求解的過程中能證明所有可行的變數(工作班)都已被考慮過，則可以獲得最佳解。由於實際上人員排班問題通常規模相當大，因此，實在不太可能實際考慮(explicit consideration)所有可能的變數，因此許多研究採用可以隱含考慮(implicit consideration)所有可能變數的變數產生法(column generation)[8]來求解。

此法的優點在於求解過程中一次只考慮部份的變數(column)，並利用主、次(子)問題間的訊息傳遞，逐步得到最佳解，如此將可增進求解效率，並可避免因變數過多而導致無法求解。Gilmore and Gomory[11]曾利用變數產生法求解 Cutting Stock 問題，而變數產生法最早用於排班問題上，則可追溯至 Appelgren[1]應用在求解具時間窗限制的船舶排班問題上。

Desrochers and Soumis[9]曾利用變數產生法求解加拿大蒙特婁市(Montreal)公車人員排班問題，其主問題為一集合涵蓋問題，利用簡捷法求解，子問題為有限制的最短路徑問題，利用動態規劃方法求解具負邊際成本的變數(column 或 workday)，加入主問題重新求解，以此步驟逐步改善以獲得最佳解。最後並配合分支定限法以獲得最佳整數解。在子問題的模式與時空網路構建上，Desrochers、Gilbert、Sauve and Soumis[10]有相當詳細的介紹與討論。

此外，變數產生法在空服員排班問題的應用上亦相當普遍。Lavoie et al [13]曾採用變數產生法發展出空服員排班模式，在其求解過程中，先以人工方法產生足夠數量之可行變數(pairing 或 column)，並以此作為起始解，其主問題為集合涵蓋問題，利用簡捷法(simplex method)自現有變數集合中找最佳解；子問題為最短路徑問題，利用 Label Setting Algorithm 找到一個對主問題目標值有貢獻的變數，將其加入主問題中重新求解，以此程序改善目前解直到無法改善為止；

在 12 個測試問題中，有 9 個可以找到最佳解。Crainic and Rousseau[7]利用變數產生法求解空服員排班問題，在其求解過程中，新的負邊際成本的變數(column or pairing)的產生是利用列舉法(enumeration method)，至於整數解則是利用啟發式單一支列舉法(heuristic single branch enumeration method)獲得。顏上堯、湯敦台[22]利用變數產生法求解多基地空服員排班組合最佳化問題，主問題用簡捷法(simplex method)求解，子問題用 Label Correcting Algorithm 求解，若獲得非整數解則利用分支定限法求得最佳整數解。

目前文獻中有關捷運司機員排班問題的文獻甚少。Chu and Chan[6]曾以作業研究模式結合電腦輔助求解香港輕軌捷運(Light Rail transit)司機員排班問題，並發展一決策支援系統，將複雜的排班過程予以自動化，研究結果顯示將可大幅提升排班的效率。

三、捷運司機員排班模式

在本研究中將捷運司機員排班問題構建成集合分割問題(Set Partitioning Problem)型式，其問題定式如下：

$$\text{Min } \sum_{r \in R} c_r x_r \quad (1)$$

Subject to

$$\sum_{r \in R} a_{ir} x_r = 1 \quad \forall i \in N \quad (2)$$

$$\sum_{r \in R} x_r = K \quad r \in R \quad (3)$$

$$x_r = 0,1 \quad r \in R \quad (4)$$

在此集合分割問題中，變數 x_r 代表一「可行(feasible)」之工作班(workday 或 column)，是 0-1 二元變數(Binary Variable)， R 為所有工作班之集合。如果 $x_r=1$ 表示工作班 r 有被選入排班表中，如果 $x_r=0$ 則沒有被選入班表中。 c_r 為工作班 r 之成本，工作班成本可分為固定成本與變動成本兩部份。固定成本代表每一司機員之基本薪資部分；變動成本則包括工時成本與人力閒置時間成本兩部份，工時成本之計算方式為：(每小時工資率) $\times$ (每一工作班之工作時數)；閒置成本則為每一工作班內扣除工作時間、規定休息時間與吃飯時間外之人力閒置時間乘上某一權重值，有效的排班即在於使人力閒置成本最小。捷運司機員排班的目標即在於使所有工作班的總成本最小，如式(1)所示。

限制式(2)表集合分割限制式。每一限制式(row) i 代表一任務(task)， N 為所有任務所成之集合。如果工作班 r 包含任務 i ，則 $a_{ir}=1$ ；如果工作班 r 不包含任務 i ，則 $a_{ir}=0$ 。此一限制式之作用在於限制每一任務只能被執行一次。由限制式(2)與二元變數限制式(4)，可知在集合分割問題模式中，每一任務只能被一工作班執行，而且只能被執行一次，如此正可滿足捷運司機員排班問題之要

求。(3)式為司機員個數限制式， K 為捷運公司編制司機員個數，此限制式限制總任務卡數要與編制司機員個數相等。至於可行(feasible)工作班的定義，則是在一工作班中須滿足休息時間、用餐時間、連續駕車時間與總駕車時間等相關工作時間之規定[21]。

四、捷運司機員排班求解演算法

本研究針對所建構的捷運司機員排班模式設計一求解演算法，此一演算法包含三個階段：變數產生法、分支定限法與啟發式解法。在第一階段中，利用變數產生法產生所有可行的工作班；在第二階段中，則由分支定限法求取整數解；最後，設計一啟發式解法以解決因變數產生法之主問題定式為集合涵蓋問題且放鬆司機員個數限制後，所可能產生之求解結果中任務卡過多，與部份任務卡中有重複涵蓋(Over-Cover)任務之問題。整個演算法求解流程如圖 1 所示。

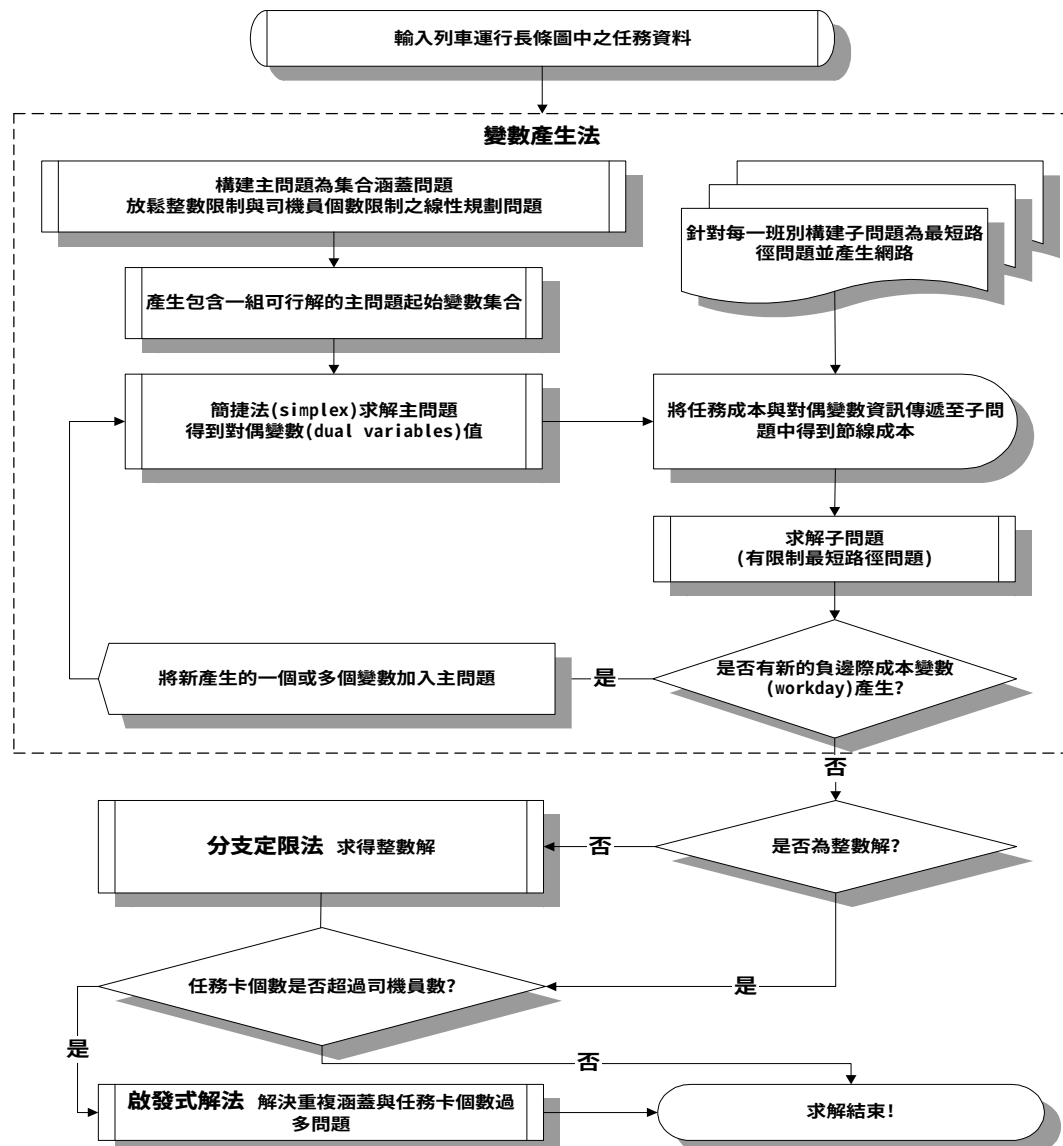


圖 1 捷運司機員排班問題求解演算法

4.1 變數產生法

在應用變數產生法求解集合涵蓋或集合分割問題時，通常將求解過程分為主問題與子問題兩部份。主問題為原問題放鬆整數限制後的線性規劃問題，由於很難去窮舉原來集合涵蓋問題的所有可能變數(workday)，因此先以人工方法或啟發式解法找到一組包含起始可行解的變數，再利用簡捷法(simplex method)求解現有變數的最佳解，並獲得每個任務(task 或 row)所對應的對偶變數。子問題通常為最短路徑問題，或是有限制的最短路徑問題(constrained shortest path problem)，利用來產生新的對目標值有貢獻的變數(工作班)，通常在最短路徑問題中為具負成本的路徑。如果能找到新的變數，則將其加入主問題中重新求解，以此程序逐步改善原有問題的解，直到無法再產生新的變數為止，便得到最佳解

4.1.1 主問題

在本研究中將捷運司機員排班問題構建為集合分割問題，以使所有任務皆能被一個工作班執行，且只能被執行一次。在應用變數產生法求解時，原則上只要將主問題定式為集合分割問題放鬆整數限制後之線性規劃問題即可。但是，根據 Desrochers[9]的研究，若將主問題定式為集合涵蓋問題(Set Covering)放鬆整數限制後之線性規劃問題，會較易獲得可行解與好的整數解，故在此將變數產生法主問題重新定式為集合涵蓋問題放鬆整數限制後之線性規劃問題。此外，為了較易找到可行解，亦放鬆司機員個數限制式。主問題定式如下：

$$\text{Min } \sum_{r \in R} c_r x_r \quad (5)$$

Subject to

$$\sum_{r \in R} a_{ir} x_r = 1 \quad \forall i \in N \quad (6)$$

$$x_r = 0, 1 \quad r \in R \quad (7)$$

此主問題為一線性規劃問題(Linear Programming Problem)。變數 x_r 代表「可行」之工作班， c_r 為工作班 r 之成本， R 為所有工作班之集合。如果 $x_r = 1$ 表示工作班 r 有被選入排班表中，如果 $x_r = 0$ 則沒有被選入班表中。

求解主問題之目標在於選擇總成本最小之可行工作班集合，如式(5)所示。限制式(6)表集合涵蓋限制式。每一限制式 i 代表一任務， N 為所有任務所成之集合。如果工作班 r 包含任務 i ，則 $a_{ir}=1$ ；如果工作班 r 不包含任務 i ，則 $a_{ir}=0$ 。此一限制式之存在使得每一任務至少會被執行一次。限制式(7)則為變數非負限制式。

若針對限制式(6)定義對偶變數 p_k (dual variables)，則可獲得此問題之對偶問題(dual problem)如下：

$$\text{Max } \sum_{i=1}^N \rho_i \quad (8)$$

Subject to

$$\sum_{i=1}^N a_{ri} \rho_i \leq c_r \quad \forall r \in R \quad (9)$$

$$\rho_i \geq 0 \quad (10)$$

根據簡捷法(simplex method)之對偶可行性(dual feasibility)可知：如果可證明對尚未考慮的變數(column)均滿足不等式(11)，則找到最佳解；否則我們可以從這些尚未考慮的變數中找到一個新的變數 r ，使得 $c_r - \sum_{i=1}^N a_{ri} \rho_i < 0$ ，因此式(11)可以幫助我們判斷是否尚有新的變數可以加入，或是已達最佳解。

$$c_r - \sum_{i=1}^N a_{ri} \rho_i \geq 0 \quad \forall r \in R \quad (11)$$

在求解主問題之前，必須先經過一前置處理程序，以產生一組包含起始可行解之變數。至於前置處理的過程，一般有以下三種方法：(一)利用 Run-cutting 啟發式解法[26]找到一組起始可行解；(二)經由修正目前使用之班表而獲得一組起始可行解；(三)加入與總任務個數相同數目之 tripper 變數（每一工作班僅包含單一任務之變數），即可獲得主問題起始可行解。在實作上，方法一與方法二皆須花費相當可觀之處理時間，且變數產生法對於此一起始可行解之品質要求並不高。因此，本研究的前置處理過程中，乃是利用方法三，以僅包含單一任務之變數來產生主問題之起始可行解。

在本研究中，由於主問題經放鬆整數限制後為一線性規劃問題，因此利用簡捷法(Simplex Method)求解，並獲得對偶變數(Dual Variables)資訊。

4.1.2 子問題

為了能夠迅速且有效的產生具有負邊際成本的可行工作班，在本研究中，乃是針對不同的工作班別，將子問題設計為最短路徑問題(Shortest Path Problem)。求解每一工作班別所對應之最短路徑問題，所得之最短路徑即可能是一個具負成本之可行工作班。

在子問題網路建構過程中，首先，依序從列車運行圖中，圈選出每一工作班別所屬工作時間內之任務，每一任務即為網路中之一節點(Node)。在所有的工作班別中，早、午班這兩個工作時段中包含司機員下車用餐時間。在用餐時段內，每位司機應下車吃飯一次，且僅能下車一次。在子問題中對於用餐時間的設計，一般是採用在用餐時段內產生用餐節點(meal node)的方法去完成[10]。但是，在本研究中，則是特別針對早、午班吃飯時段之要求，將子問題網路修改為「二階段子問題網路」。

在此二階段網路中，將原本之工作班別子問題網路以司機員可能下車用餐之時間，分為用餐前與用餐後二階段之子網路。第一階段子問題網路的起點為簽到時間，迄點為用餐開始時間；第二階段子問題網路的起點為用餐結束時間，迄點為簽退時間。分別對此二階段子網路求解最短路徑問題後，將所得之最短路徑加以合併，即可得原來工作班別子問題網路之最短路徑加上用餐時間後之結果。如此，便可處理排班時早、午班司機員用餐時間之要求。二階段子問題網路如圖 2 所示。

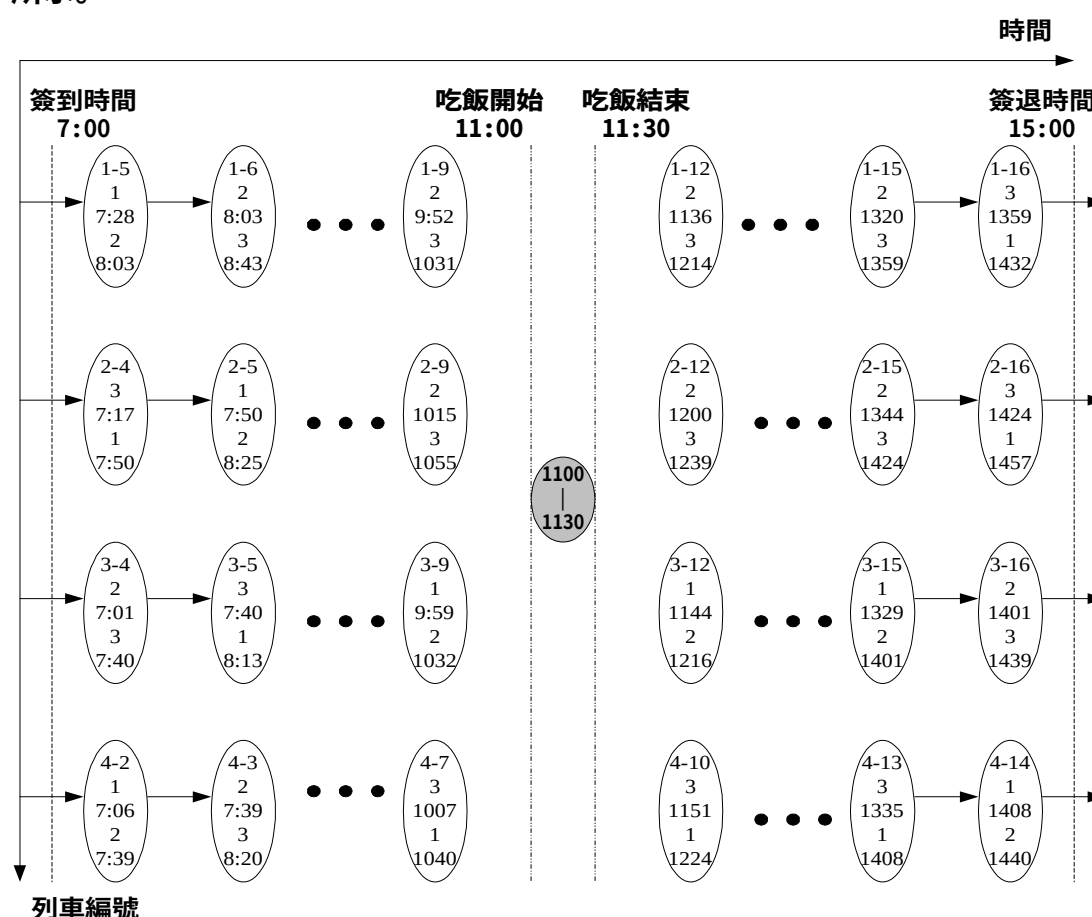


圖 2 二階段子問題網路圖

由於二階段子問題網路之設計，網路中之節點除了原有之任務節點外，尚包含簽到點、用餐時間開始點、用餐時間結束點、簽退點。任何兩個節點間的時間與空間關係如果滿足捷運公司排班基本原則之規定，則可以一節線加以連接。若考慮節線所連接之起迄點不同，可能節線連接情形有：簽到點與任務點、同一列車相鄰之任務點、不同列車間之任務點、任務點與用餐開始點、用餐結束點與任務點、任務點與簽退點。

在子問題網路中，每一節點會有其成本與相對應之對偶變數值，因此在指派節線成本時，須將節點成本與對偶變數值轉移至節線上。假設 $Node_ID$ 表一節線所連接兩節點中之迄點，則將此節線成本定義為： $Cost(Node_ID) - Dual(Node_ID) + time_differ \times W$ 。其中 $Cost(Node_ID)$ 代表任務成本， $Dual(Node_ID)$ 代表任務

所對應之對偶變數值，乃是根據限制式(6)而定義，並經由求解主問題後，將對偶變數值傳遞至子問題中相對應之任務節點上，此一過程稱為 Pricing。 $time_differ$ 則是相連兩節點間之閒置時間成本，其計算方式為：迄點之任務開始時間減去起點之任務結束時間； W 為一權重值，可反映閒置時間成本之重要程度，例如連接不同列車任務之節線，其 $time_differ$ 之權重應大於連接同一列車任務節線之 $time_differ$ 權重值。各類節線成本值如下表所示。

表 1 各類節線成本設定表

類別	節線起點	節線迄點	Cost(Node_ID)	Dual(Node_ID)	time_differ	W
A	簽到點	一般任務	Cost(迄點任務)	Dual(迄點任務)	time_differ	Wa
	用餐點	一般任務	Cost(迄點任務)	Dual(迄點任務)	time_differ	Wa
B	一般任務	簽退點	0	0	time_differ	Wb
	一般任務	用餐點	0	0	time_differ	Wb
C	同一列車相鄰任務		Cost(迄點任務)	Dual(迄點任務)	0	Wc
D	不同列車任務		Cost(迄點任務)	Dual(迄點任務)	time_differ	Wd

在四類節線中，B 類節線之迄點為不具成本與對偶變數值之簽退點或用餐點，故其 $Cost(Node_ID)$ 與 $Dual(Node_ID)$ 部份為零，節線成本僅有 $time_differ \times W$ 。

在本研究之子問題網路中，除了司機員用餐時間與休息時間外，要產生「可行」工作班，尚須考慮到連續駕駛時間與總駕駛時間等限制條件。但是，由於這些條件並不容易從子問題網路設計上去達成，因此，在處理上乃是留到設計求解子問題演算法時再作處理。也正因為這些額外限制條件的存在，在求解困難度與複雜度上，亦相對較高。文獻上[4]，將這類最短路徑問題稱為「有限制之最短路徑問題(Constrained Shortest Path Problem)」，這類問題屬於 NP-Hard[4]。

由於額外限制式的存在，一般求解有負節線成本最短路徑問題的標籤修正法已經不適合用來求解有限制之最短路徑問題。因此，在本研究中參考 Nemhauser[19]提出多重標籤之觀念，設計一多重標籤修正法來求解變數產生法之子問題。

本研究所設計之多重標籤修正法，除了「成本」外，尚有「連續駕駛時間」與「總駕駛時間」兩個標籤。此二標籤可視為「資源(resource)」，在每一節點上，都會有此二資源的使用量(任務執行時間)；在每次利用 Bellman 條件判斷是否更新節點上之「成本」標籤前，尚須檢查另外兩項資源的使用是否滿足限制條件，才能決定是否要更新此一節點之「成本」標籤與前置點，以及「連續駕駛時間」之重設或累積，與「總駕駛時間」之累積。

當目前終點之任一下游節點上「成本」標籤不滿足 Bellman 條件時，則檢查「總駕駛時間」與「連續駕駛時間」兩個標籤。首先，必須取得節點上之資源使用量資訊，也就是任務執行時間(Task Duration)。接著檢查此一資源消耗是否

違反「總駕駛時間」之限制。若違反則停止檢查，不更新節點上之標籤與前置點，繼續檢查下一個下游點；若不違反，則檢查是否違反「連續駕駛時間」之限制。

若違反限制，須判斷目前終點與此下游點之列車編號(Train No)是否相同。若不同表示司機員已下車休息，目前終點與此下游節點間連續駕駛行為不成立，不違反「連續駕駛時間」限制，故更新「成本」、「總駕駛時間」與前置點，歸零「連續駕駛時間」；若車號相同尚有兩種可能情形，一是目前終點與此下游點相鄰，表示兩點之連接為連續駕駛行為，則停止檢查，不更新此下游點上之標籤與前置點，繼續檢查下一個下游點；另一則是目前終點與下游點不相鄰，表示司機員下車休息後，再回到同一列車執行任務，不違反「連續駕駛時間」限制，故更新「成本」、「總駕駛時間」與前置點，歸零「連續駕駛時間」。

若未違反限制，則表示不論目前終點與此下游點之列車編號是否相同，兩點是否相鄰，都要更新「成本」、「總駕駛時間」與前置點；但是，須注意的是在列車編號相同，且目前終點與此下游點亦相鄰之情形時，表示兩點之連接屬於連續駕駛行為，須更新「連續駕駛時間」，也就是將目前終點之「連續駕駛時間」加上此下游點之任務執行時間，而非歸零。

在更新或歸零過此一下游點上的三個標籤後，若此一下游點尚未出現在記錄容器內，則將其加入容器內，以作為下次運算的目前終點，並更新目前列車編號與節點編號。如此依序檢查完目前終點之所有下游點後，再從容器內取出下一點。若容器內已無節點，則演算法停止。

4.2 啟發式解法

由於在變數產生法求解階段中，將主問題定式為集合涵蓋問題，而非集合分割問題，所以在求解結果中可能會有重複涵蓋任務(over-covering task)情形發生。此外，求解過程中亦放鬆司機員個數限制，因此求解產生工作班個數亦可能超過捷運公司編制司機員個數。在本研究中，設計一二階段式啟發式解法以解決這兩個問題。在第一階段中，求解目標為移除過多的任務卡。第二階段則設計來解決重複涵蓋任務問題。

在第一階段中，首先計算每張任務卡中之非重複涵蓋任務個數，如果任務卡張數超過捷運司機員編制人數，則選取非重複涵蓋任務個數最少的那張任務卡 k ，先移除任務卡 k 中所包含之重複涵蓋任務，接著執行取代程序(Replace Procedure)。

在取代過程中，乃是將任務卡 k 中之非重複涵蓋任務，在滿足 $\hat{A}$ 換班地點、 $\hat{C}$ 休息時間、 $\hat{E}$ 連續駕駛時間、 $\hat{N}$ 總駕駛時間與 $\hat{O}$ 月台機廠旅行時間(如果有尖轉離收車或離轉尖發車情形)限制條件下，插入到其它任務卡 k_1 中之可行位置，以取代 k_1 中之重複涵蓋任務。如果可行取代位置不止一個，則優先選擇被取代後有重複涵蓋任務個數最多之位置，接著再考慮取代後閒置時間最少之位置。

在經過以上處理程序後, 如果任務卡 k 中仍有剩下無法處理之非重複涵蓋任務, 則將這些任務記錄在一容器(Container)中。重複執行以上各步驟, 直到任務卡張數等於司機員編制個數後, 求解流程進入第二階段。

首先, 執行移除程序(Remove Procedure)以移除任務卡中之重複涵蓋任務。此一程序設計之原則在於避免破壞原本連續執行之任務鍊, 並使工作班中換車情形較少。接著執行插入法(Insert Procedure)。在插入法過程中, 乃是將第一階段中無法處理而記錄在容器中之非重複涵蓋任務, 在滿足 $\hat{A}$ 換班地點、 $\hat{C}$ 休息時間、 $\hat{E}$ 連續駕駛時間、 $\hat{N}$ 總駕駛時間與 $\hat{O}$ 月台機廠旅行時間(如果有尖轉離收車或離轉尖發車情形)限制條件下, 插入到所有任務卡中之可行位置。如果可行插入位置超過一個, 則選擇插入後閒置時間最少者。將容器中所有任務依序執行插入法後, 如果沒有剩下的任務, 則求解結束, 輸出所有任務卡。

如果尚有剩下的任務, 檢查這些剩下的任務是否能夠組成新的工作班, 如果不能, 將剩下的任務交由領班或助理以備班方式執行; 如果可以, 則執行建構程序(Construct Procedure)以組成新的工作班。建構程序將每一工作班別依不同吃飯時間, 分兩階段建構出工作班, 如果建構出之工作班滿足各項工作時間限制, 則為一可行之工作班, 再將此工作班中所包之任務從容器中移除。如果還有剩下無法組成新工作班的任務亦交由備班執行。

五、實際問題求解

在本研究中所使用測試例題乃是台北捷運公司於 88 年 4 月通過之淡中線(淡水—中和)平常日列車運行時刻表, 與 89 年 1 月通過之板南線(板橋—南港)平常日列車運行時刻表。前者之司機員排班問題有 25 列車, 471 個任務; 後者則有 7 列車, 353 個任務。工作班工作時間參數設定如表一所示:

由表一中可發現在簽到(退)時間、連續駕車時間、總駕車時間、休息時間、用餐時間與早、午班吃飯時段等各項工作班工作時間參數上, 目前人工排班時所設定之參數值均違反排班基本原則之規定。推究其原因, 應為捷運公司司機員編制人數不足, 以致於排班人員在進行排班工作時, 為滿足現有司機員個數限制, 須放寬多項工作班工作時間參數設定標準, 才能夠排出可行班表。就捷運公司而言, 在司機員能夠負荷的範圍內儘可能減少司機員編制數目可以降低人事成本; 但是, 在連續駕駛時間與總駕駛時間上便須加以延長, 在休息時間與用餐時間上更會受到壓縮, 對司機員而言工作負擔更為加重。

表一 工作班工作時間參數設定表

時間項目	排班基本原則規定	本研究設定值	人工排班設定值
簽到(退)時間	5	≥ 5	≥ 0
步行時間 (月台機廠往返)	(中和) 10 (北投) 30	(中和) 10 (北投) 30	(中和) 10 (北投) 30
發車前整備時間	40	40	40
收車後檢視時間	30	30	30
連續駕車時間	30 – 180	30 – 190	30 – 190
總駕車時間	240 – 360	270 – 400	270 – 400
休息時間	15 – 40	10 – 90	8 – 88
用餐時間	30 – 65	30 – 65	28 – 64
用餐時段	(早班) 10:30–13:00 (午班) 16:30–19:00	(早班) 10:00–14:00 (午班) 16:00–20:00	(早班) 10:00–14:00 (午班) 16:00–20:00

因此，在本研究中，在不超過人工排班設定值之原則下，亦適度放寬工作班工作時間參數設定值，以期能在超過司機員負荷最小的目標下，排出可行之班表。經過調整放寬後之工作時間設定值如表一所示。

根據捷運公司所提供之資料，88.4 淡中線平常日司機員排班問題，經人工方式排班之結果共產生 58 張任務卡。人工排班結果整理如表二所示。人工排出之班表，其總任務卡數等於當時捷運公司於北投車班之司機員編制個數，但是在午班所屬工作時段中仍有數個任務無法排入，須由領班或助理以備班方式執行。

表二 88.4 淡中線平常日司機員排班問題求解比較表

求解內容	本研究求解結果	人工排班結果
目標函數值	37857	41802
任務卡個數	58	58
無法排進班表任務個數	6 (早班 5 個，午班 1 個)	4 (午班 4 個)
求解時間	85 小時	7 天

利用本研究演算法求解結果如表二所示。在剩下 6 個無法排進班表的任務中，早班工作時段有 5 個，午班工作時段有 1 個，此 6 個任務可由領班或助理以備班方式執行。將上述求解結果與捷運公司所提供之資料作比較，雖然人工排班結果中，無法排進班表的任務個數(4 個)較本研究演算法求解結果中無法排進班表任務個數(6 個)來得少。但是，若進一步分析人工排班結果，此 4 個任務乃是集中在午班用餐時段，也是運量尖峰時段，捷運公司為因應運量需求增加，須較離峰時段增開列車。在司機員人數固定的限制下，為執行因加開列車所增加之列車駕駛任務，經常造成司機員用餐時間過短(不足 30 分鐘)，紛紛向排班人員反應與抱怨。若無法排進班表的任務又都

集中在午班用餐時段，且要動用至少一名備班司機執行任務，將會更增加排班人員的困擾。

相較之下，本研究求解結果在午班時段只有 1 個任務須由備班執行，且司機員用餐時間遵守規定之標準(最少 30 分鐘)，已能改善現況。至於本研究求解結果中，雖然早班工作時段有較多任務無法排進班表，但由於這些任務較為分散，且早班吃飯時段(10:00-13:00)為運量離峰時段，故影響較少，也易為排班人員所接受。

此外，若不考慮這些無法排進班表任務，而僅依照本研究對於任務卡成本之定義(包括各成本項目與成本參數)來計算，本研究演算法求解所產生之 58 張任務卡成本總和(即司機員排班模式中之目標函數值)為 37857，而人工排班所得班表之總任務卡成本為 41802。在須要相同人數的備班司機來執行那些無法排進班表任務的情況下，本研究演算法求解結果明顯優於捷運公司人工排班之結果。

至於 89.1 板南線平常日司機員排班問題求解結果如表三所示。在剩下 3 個無法處理的任務中，在午班工作時段有 2 個，夜班工作時段有 1 個，此 3 個任務可由領班或助理以備班方式執行。由於捷運公司目前在板南線的司機員排班並非採任務卡方式，而是由 28 名司機員採機動調度方式執行所有任務，故此一測試例題求解結果無法與現況進行比較。

表三 89.1 板南線平常日司機員排班問題求解比較表

求解內容	本研究求解結果	人工排班結果
目標函數值	13413	由 28 名司機員採機動調度方式執行所有任務，故無法比較。
任務卡個數	28	
無法排進班表任務個數	3 (午班 2 個，夜班 1 個)	
求解時間	75 小時	

六、結論與建議

本研究之目的在於提升台北捷運公司司機員排班之效率，並減輕排班人員之工作負擔。在司機員排班模式建立方面，本研究將捷運司機員排班問題建構為集合分割問題模式，並針對此模式設計一演算法求解捷運司機員排班問題。求解演算法包含變數產生法、分支定限法與啟發式解法三階段。

利用演算法求解捷運公司 88.4.26 版列車運行時刻表之測試例題的過程中，發現捷運公司因編制司機員數不足，必須多項放寬排班基本原則中所規定之工作班時間參數設定標準，否則無法在現有司機員個數下排出一可行班表。本研究經過調整測試，在超過司機員負荷最小的原則下，適度放寬數項工作班時間參數設定值後，利用演算法求解獲得捷運司機員排班表。

分析求解結果發現，雖然求解產生的任務卡張數等於司機員編制人數，但是仍有少數幾個任務無法排進班表中，須另外由領班或助理以備班方式執行，此一結果與目前捷運公司以人工方式排班之現況相同。但是深入分析人工排班結果可發現無法排進班表任務全部集中在午班用餐時段，也是運量尖峰時段，為因應尖峰時段加開列車所增加之列車駕駛任務，已造成司機員因用餐時間過短，紛紛向排班人員提出反應，若無法排進班表任務又全部集中在午班用餐時段，而須動用備班司機員，將更加造成排班人員困擾。相較之下，本研究之求解結果已明顯改善此一現象。而且，在需要相同人數備班司機員來執行那些無法排進班表之任務的情況下，本研究演算法所得結果之總任務卡成本將較人工方式排班結果之總任務卡成本為低。

此外，在本研究中將整個求解流程自動化，除了須由排班人員輸入任務資料外，剩餘求解工作皆可由個人電腦執行。且本研究中演算法求解時間約 3-4 日，相較於捷運公司目前以人工方式排班須花費數星期時間，已可大幅提升司機員排班效率，及減輕排班人員之工作負擔。

在本研究中，雖然演算法求解結果中剩餘少數無法排進班表中之任務可仿照台北捷運公司人工方式排班之現況，由領班或助理以備班方式執行，但是對於研究成果而言，在完整性上仍稍有不足。

此一現象，可從變數產生法的基本原理來分析。由於變數產生法乃是利用子問題中所設計之求解機制，將任務組合成對主問題求解有貢獻(負成本)之可行工作班。但是，所謂「有貢獻(負成本)的可行工作班」乃是指在本研究中所設計之子問題網路與其節線成本定義下求解之結果。在本研究的變數產生法求解階段中，已經產生出所有具負成本的有貢獻變數，卻仍然會有重複涵蓋任務與少數任務無法排進班表之情形，可見在本研究目前的子問題網路與節線成本設計下，只產生具負成本的有貢獻變數已不足夠排出完整班表所需之所有可行工作班，也就是可能須進一步考慮產生並加入具正成本(沒有貢獻)之工作班。至於要產生多少的正成本工作班，目前所參考之變數產生法應用文獻中尚無相關之探討；且受限於研究時程，本研究尚無法以數學方法加以推導與驗證。此一部份之研究工作，將有賴後續更廣泛之相關文獻參考，及更深入且嚴謹的數學理論求證。

分析三階段求解過程所花費之運算時間，可發現第二階段分支定限法之求解時間佔所有運算時間的百分之九十以上。由於分支定限法的求解乃是在程式中呼叫 CPLEX 軟體所提供之函式，其求解乃是採用傳統窮舉所有變數之方式，因此變數越多，其運算所須時間便越長。此一問題除了仰賴未來計算機科技的發展外，對於如何提升分支定限法之求解效率，許多相關文獻中皆有相當有效之建議。

這些改善建議主要分為分支策略與節點選擇策略兩部份。在分支策略方面，有研究提出 Constraint Branching 方法，對一組目標變數進行分支，以提升求解效率。在節點選擇策略方面，文獻中提到常用的方式有：最佳目標值法

(Best-Bound Search)、深度搜尋法 (Depth-First Search)、廣度搜尋法 (Breadth-First Search) 以及隨機搜尋 (Random Search)。不同的節點選擇方式對於求解效率亦會有不同程度之影響。在改善分支定限法求解時，上述的方法都值得進行測試，以提高整個演算法的求解效率，縮短排班所需時間。

七、參考文獻

1. Appelgren, L. H., "A Column Generation Algorithm for a Ship Scheduling Problem", *Transportation Science*, 3, pp. 53-68 (1969).
2. Bailey R. N., Garner, K. M., Hobbs M. F., "Using Simulate Annealing and Genetic Algorithm to Solve Staff-Scheduling Problems", *Asia-Pacific Journal of Operational Research*, Vol. 14, No. 2, pp.27-43 (1997).
3. Ball, M., Bodin, L. and Dial, R., "A Matching Based Heuristics for Scheduling Mass Transit Crews and Vehicles", *Transportation Science*,17(1), pp.4-31(1983).
4. Ball, M. O., Magnanti, T. L., Monma, C. L., Nemhauser, G. L. "Ch2. Time Constrained Routing and Scheduling" in *Handbooks in Operations Research and Management Science Volume 8 : Network Routing*.
5. Bartholdi, John, J., "A Guaranteed-Accuracy Round-off Algorithm for Cyclic Scheduling and Set Covering", *Operations Research*, 29, pp. 501-510 (1981).
6. Chu, C. K. Sydney and Chan C. H. Edmond, "Crew Scheduling of Light Rail Transit in Hong Kong : From Modeling to Implementation", *Computers and Operations Research*, Vol. 25, No. 11, pp. 887-894 (1998).
7. Crainic, T. G. and Rousseau, J. "The Column Generation Principle and the Airline Crew Scheduling Problem" *INFOR*, Vol. 25, No. 2, pp. 136-151 (1987).
8. Dantzig, G. B., and P. Wolfe "The Decomposition Algorithm for Linear Programming" *Operations Research*, 8, pp. 101-111 (1960).
9. Desrochers, Martin and Soumis, Francois, "A Column Generation Approach to the Urban Transit Crew Scheduling Problem", *Transportation Science*, Vol. 23, No. 1, pp. 1-13 (1989).
10. Desrochers, M., Gilbert, J., Sauve, M., Soumis, F. "Crew-Opt : Subproblem Modeling in a Column Generation Approach to Urban Crew Scheduling" in *Computer-Aided Transit Scheduling*, pp. 395-406. Lecture Notes in Economics and Mathematical System 386. Springer-Verlag, Berlin, 1992.
11. Gilmore, P. C. and Gomory, R. E., "A Linear Programming Approach to the Cutting-Stock Problem", *Operations Research*, Vol. 9, pp. 849-859 (1961).
12. Heurigon, H., "Preparing Duty Roster for Bus Routes by Computer", in *Preprint Workshop on Automated Technique for Scheduling of Vehicle Operations for Urban Public Transportation Service* (1975).
13. Lavoie, S., M. Minoux and E. Odier, "A new Approach for Crew Pairing

- Problems by Column Generation with an Application to Air Transportation” *European Journal of Operational Research*, Vol. 35, pp. 45-58 (1998).
14. Lessard, R., Rousseau, J. M. and Dupuis, D., “HASTUS I : A Mathematical Programming Approach to the Bus Driver Scheduling Problem”, in *Computer Scheduling of Public Transport : Urban Passenger Vehicle and Crew Scheduling*, ed. A. Wren. North-Holland, Amsterdam, pp. 255-268, (1981).
 15. Levine, D. and Argonne, “Application of a Hybrid Genetic Algorithm to Airline Crew Scheduling”, *Computers and Operations Research*, Vol. 23, pp. 547-558, 1996.
 16. Mitra, G. and Darby-Dowman, K, “CRU-SHED : A Computer Based Bus Crew Scheduling System Using Integer Programming”, in *Computer Scheduling of Public Transport 2, Elsevier, Amsterdam*, 1985.
 17. Rosseau, J. M., Lessard, R. and Blais, J. Y., “Enhancements to the HASTUS Crew Scheduling Algorithm”, in *Computer Scheduling of Public Transport*, Vol. 2, ed. J. M. Rousseau. North-Holland, Amsterdam, pp. 295-310 (1985).
 18. Shepardson, F. and Marsten, R. E., “A Lagrangean relaxation Algorithm for the Two Duty Period Scheduling Problem”, *Management Science*, 26, pp. 274-281 (1980).
 19. Vance, P. H., Barnhart, C., Johnson, E. L., Nemhauser, G. L., “Airline Crew Scheduling : A New Formulation and Decomposition Algorithm”, *Operations Research*, Vol. 45, No. 2 (1997).
 20. Wren, A., Smith, B. M. and Miller, A. J., “Complementary Approach to Scheduling”, in *Computer Scheduling of Public Transport 2, Elsevier, Amsterdam*, 1985.
 21. 台北大眾捷運股份有限公司, 高運量運務人員勤務管理作業要點 (1996).
 22. 顏上堯, 湯敦台 “多基地空服員排班組合最佳化” *中華民國運輸學會第十一屆論文研討會論文集*, pp. 73-84 (1996).
 23. 廖學昌 “公車客運業人員排班問題之研究—以金門縣公車為例”, *國立交通大學交通運輸研究所碩士論文* (1998).