

逢甲大學
交通工程與管理學系碩士班
碩士論文

禁制搜尋法於動態車輛巡迴路線問題之研究

A Tabu Search Algorithm for The
Dynamic Vehicle Routing Problem

指導教授：胡大瀛

研 究 生：柯景文

中華民國九十一年六月

摘要

隨著物流業的發展，商用車輛營運操作 (Commercial Vehicle Operations, CVO) 也就成為智慧型運輸系統 (Intelligent Transportation Systems, ITS) 發展以來重要的課題之一，由於業者可透過車輛上的定位系統，進行監控營運車輛的位置，並根據即時的交通路網狀況來改善車隊的營運派遣，以降低營運上的成本並提昇顧客貨物運輸的服務水準，因此動態的車輛巡迴路線的問題 (Vehicle Routing Problem, DVRP) 逐漸的受到重視。

國內外一直有許多文獻探討車輛巡迴路線的問題，由於該問題屬於 NP-Hard，因此多以啟發式解法進行問題的求解。本研究將藉由禁制搜尋法 (Tabu Search) 求解動態 VRP 問題，禁制搜尋法中，主要包含了移步 (Move)、禁制名單 (Tabu List)、渴望準則 (Aspiration Level)、候選名單 (Candidate List) 及搜尋停止準則 (Stopping Criterion) 五大模組，其主要透過禁制名單及渴望準則決定出下一次可能移步的候選名單，以進行最佳的移步動作，並在此一動作循環下配合搜尋停止準則以求得最佳解，且該方法不易陷入先前的區域最佳解，並能避免在求解過程中可能發生的循環 (cycling)，進而得到全域的最佳解。

本研究將針對即時資訊下的車輛巡迴路線問題進行研究，為了能快速獲得較佳的解，將藉由禁制搜尋法 (Tabu Search) 求解動態 VRP 問題。研究中提出一擁擠路段的預測方式，並根據即時性與歷史性兩種不同的資訊屬性，設計了空間上及時間上的禁制名單，因此在獲得即時資訊更新巡迴路線解時，能避免行經較擁的區域，而在事故發生時，亦能避開肇事路段，降低路線上的停等時間。本研究將在路段行駛速率隨時間變動下的條件下，藉由動態 VRP 的解法計算出即時性的最佳巡迴路線，並構建一車輛巡迴路線之交通模擬架構，用於未來路線的評估、產生、與動態性資訊的產生與研擬；此一架構將結合 VRP 的演算法與動態模擬指派模式進行動態路線之評估與分析，藉此反應真實的交通特性。

關鍵字：智慧型運輸系統、車輛巡迴路線問題、禁制搜尋法、即時資訊

Abstract

Dynamic vehicle dispatching and routing strategies are important aspects of Commercial Vehicle Operations (CVO) applications of Intelligent Transportation Systems (ITS) technologies. One of the primary operational benefits of real-time information on vehicle locations and demands is the ability to efficiently re-route vehicles according to current traffic conditions. The management can also reduce the operating cost and enhance the service level for the customer by the vehicles assignment under the real-time traffic conditions. Thus the Dynamic Vehicle Routing Problem (DVRP) plays an important role in designing routes under real-time information.

Since no existing techniques could be applied to solve the DVRP directly, thus a heuristic approach based on tabu search is developed to solve the problem. There are five steps in a tabu-search based algorithm: Move, Tabu List, Aspiration Level, Candidate List and Stopping Criterion. The general tabu search explores part of the solution space by moving at each iteration to the best neighbor of the current solution, even if this leads to a deterioration of the objective function. Solutions that were recently considered are made inaccessible for a number of iterations to avoid cycling.

This research considers two different types of information, the historical information and real-time information. This research deploys different types of information to design a temporal and spatial tabu list for route construction and improvement. The re-route vehicles can avoid the congestion and incident under real-time information. The heuristic approach is then applied in an evaluation framework in which dispatching and routing operations could be simulated in a realistic traffic environment. The simulation-assignment model, DYNASMART, is applied to evaluate dispatching and routing strategies in a traffic network. Numerical experiments are conducted in a Taichung City Network to explore the evaluation framework for dynamic fleet management problem under real-time information supply strategies.

Keywords: Tabu Search, Vehicle Routing Problem (VRP), real-time information

目錄

目錄	i
圖目錄	iii
表目錄	v
第一章 緒論	1
1.1 研究背景	1
1.2 研究目的	2
1.3 研究方法與流程	3
1.4 章節敘述	4
第二章 文獻回顧	6
2.1 車輛巡迴路線問題	6
2.2 車輛巡迴路線問題的求解	9
2.3 動態車輛巡迴路線問題	13
2.4 禁制搜尋法	15
2.5 交通模擬指派模式	21
2.5.1 路網模擬指派模式	21
2.5.2 DYNASMART 模擬指派模式	22
第三章 即時資訊下動態車輛路線分析架構	26
3.1 即時資訊下動態 VRP 模擬評估架構之建立	26
3.1.1 動態派遣策略架構	26
3.1.2 即時資訊下動態 VRP 模擬評估架構	28
3.2 禁制搜尋法求解動態車輛巡迴路線問題	31
第四章 禁制搜尋法於靜態問題之測試	34
4.1 實驗設計	34
4.1.1 實驗目的	34
4.1.2 測試例題與參數設定	34

4.2 結果分析	37
4.3 小結	39
第五章 即時資訊下動態 VRP 之分析與實驗	41
5.1 演算法設計	41
5.1.1 考慮即時資訊下巡迴路線演算流程	41
5.1.2 結合即時資訊與歷史資訊下巡迴路線演算流程	44
5.2 實驗設計	50
5.2.1 台中市路網資料	50
5.2.2 參數設定	50
5.3 考慮即時資訊下動態 VRP 之結果分析	52
5.3.1 情境 1	52
5.3.2 情境 2	55
5.4 結合即時資訊與歷史資訊下動態 VRP 之結果分析	58
5.5 考慮事故發生下動態 VRP 之結果分析	62
5.6 小結	65
第六章 結論與建議	66
6.1 結論	66
6.2 建議	67
參考文獻	68
附錄一 動態車輛巡迴路線問題之求解程式流程與程式	71
動態車輛巡迴路線問題之求解程式流程	71
動態車輛巡迴路線問題之求解程式	72
附錄二 動態需求點順序	80
考慮即時資訊下動態需求點順序	80
結合歷史資訊與即時資訊下動態需求點順序	94
事故發生下動態需求點順序	98

圖目錄

圖 1 研究流程圖	4
圖 2.1 掃描法分群圖	10
圖 2.2 節省法示意圖 (原路線)	11
圖 2.3 節省法示意圖 (節省後)	11
圖 2.4 2-opt 交換示意圖 (原路線)	11
圖 2.5 2-opt 交換示意圖 (交換後)	12
圖 2.6 禁制搜尋法之流程圖	16
圖 2.7 短期記憶的使用方式	18
圖 2.8 DYNASMART 模式架構	24
圖 2.9 DYNASMART 功能架構	25
圖 3.1 貨運車輛指派與巡迴路線研擬流程圖	27
圖 3.2 模擬式動態派遣策略研擬之架構	28
圖 3.3 即時資訊下動態派遣策略之演算流程	30
圖 3.4 求解動態 VRP 問題的演算流程	32
圖 3.5 DYNASMART 與運算程式之演算流程	33
圖 4.1 需求點數與運算時間關係圖	39
圖 5.1 考慮即時資訊下巡迴路線演算流程圖	43
圖 5.2 不同公車行駛同路線之時空圖	44
圖 5.3 結合即時資訊與歷史資訊下巡迴路線演算流程	49
圖 5.4 台中市路網圖	50
圖 5.5 情境 1 的車輛產生型態	52
圖 5.6 情境 1-10 個需求點之巡迴路線比較圖	53
圖 5.7 情境 1-20 個需求點之巡迴路線比較圖	54
圖 5.8 情境 2 的車輛產生型態	55
圖 5.9 情境 2-10 個需求點之巡迴路線比較圖	56
圖 5.10 情境 2-20 個需求點之巡迴路線比較圖	58

圖 5.11 即時資訊頻率為 5 分鐘下考慮不同資訊種類之巡迴路線比較圖	59
圖 5.12 即時資訊頻率為 10 分鐘下考慮不同資訊種類之巡迴路線比較圖	60
圖 5.13 即時資訊頻率為 15 分鐘下考慮不同資訊種類之巡迴路線比較圖	61
圖 5.14 事故發生下 10 個需求點之巡迴路線比較圖	63
圖 5.15 事故發生下 20 個需求點之巡迴路線比較圖	64



表目錄

表 2.1 禁制搜尋法於車輛路巡迴路線問題之比較表	20
表 4.1 TSP 測試例題表	35
表 4.2 測試項目及範圍	36
表 4.3 測試結果表	37
表 4.4 最近鄰點法下不同設計準則與執行速度	38
表 4.5 本研究所得之最佳解	40
表 5.1 交通模擬時參數項目、測試範圍及代號	51
表 5.2 10 個需求點之巡迴路線路段數、旅行時間及改善百分比	53
表 5.3 20 個需求點之巡迴路線路段數、旅行時間及改善百分比	55
表 5.4 10 個需求點之巡迴路線路段數、旅行時間及改善百分比	57
表 5.5 20 個需求點之巡迴路線路段數、旅行時間及改善百分比	58
表 5.6 即時資訊頻率為 5 分鐘下考慮不同資訊種類之巡迴路線路段數、路線停等時間、旅行時間及改善百分比	59
表 5.7 即時資訊頻率為 10 分鐘下考慮不同資訊種類之巡迴路線路段數、路線停等時間、旅行時間及改善百分比	60
表 5.8 即時資訊頻率為 15 分鐘下考慮不同資訊種類之巡迴路線路段數、路線停等時間、旅行時間及改善百分比	61
表 5.9 10 個需求點之巡迴路線路段數、旅行時間及改善百分比	63
表 5.10 20 個需求點之巡迴路線路段數、旅行時間及改善百分比	64

第一章 緒論

1.1 研究背景

隨著物流業的發展，商用車輛營運操作 (Commercial Vehicle Operations, CVO) 也就成為智慧型運輸系統(Intelligent Transportation Systems)發展以來重要的課題之一，由於業者可透過車輛上的定位系統，進行監控營運車輛的位置，以達到即時派遣，減少巡迴、載貨、送貨所需的時間，並可透過即時的交通路網狀況來改善車隊的營運派遣，如車輛數目、車輛大小、路線的調整等，因此動態車輛巡迴路線的問題(Dynamic Vehicle Routing Problem, DVRP)逐漸的受到重視；有效的巡迴路線可以降低營運上的成本並提昇顧客貨物運輸的服務水準，但由於現有的車輛旅行途中，其道路行駛速率會隨著突發的交通狀況而產生改變，因此如何在獲得即時性的交通資訊下重新規劃車輛巡迴路線，降低車輛的旅行成本便成為一個重要的課題。

國內外一直有許多的文獻探討車輛巡迴路線的問題，由於該問題屬於 NP-Hard，在透過數學規劃方式進行求解時，其問題規模變大時，需要較長的運算求解時間，因此許多啟發式演算法被應用於解決類似的問題。由於現有的路線規劃模式在最佳路線的計算中其旅行成本多為靜態的假設，因此無法適時反應道路的即時車流狀況，而在實際的運輸系統環境下，這些演算法也只能被應用在顧客需求量、顧客有無服務及車輛旅行時間確定型行業中，對於實際車輛旅行時間不確定的情形下較不適用。

呂英志（2001）利用數學規劃的方式求解即時資訊下的車輛巡迴路線問題，其主要透過數規軟體 CPLEX 進行求解，而由於 CPLEX 是透過分支界定法 (Branch and Bound) 的方式求解整數規劃問題，當求解的問題其需求點過多的情況下，會導致過多的分支，使得求解的速度變慢，甚至於求不出最佳解；為了能快速獲得較佳的解，本研究將藉由禁制搜尋法 (Tabu Search) 的技巧來求解動態 VRP 問題。研究中將在路段行駛速率隨時間變動的條件下，藉由動態旅行時間的 VRP

解法計算出即時性的最佳巡迴路線，並構建一車輛巡迴路線之交通模擬架構，用於未來路線的評估、產生、與動態性資訊的產生與研擬，此一架構將結合 VRP 的演算法與動態模擬指派模式進行動態路線之評估與分析，希望藉此來反應真實的交通特性。

禁制搜尋法為一高階的萬用啟發式解法，專門用來求解組合最佳化的問題。目前已應用的領域如排程、車輛旅途、旅行銷售員問題(TSP)等組合最佳化問題，不同於過去的啟發式演算法，在禁制搜尋法中，主要包含移步 (Move)、禁制名單 (Tabu List)、渴望準則 (Aspiration Level)、候選名單 (Candidate List) 及搜尋停止準則 (Stopping Criterion) 五大模組，其主要透過禁制名單及渴望準則決定出下一次可能移步的候選名單，以進行最佳的移步動作，並在此一動作循環下配合搜尋停止準則以求得最佳解，且該方法不易陷入先前的區域最佳解，並能避免在求解過程中可能發生的循環(cycling)，進而得到全域的最佳解。

1.2 研究目的

本研究探討車輛自配送中心，運送貨物至各需求點時考慮的車輛運送路線問題，由於車輛在行駛途中會因為道路、氣候等狀況、即時交通資訊而使行車時間改變，因此必須在獲得相關交通資訊下重新規劃指派新的巡迴路線，以保持應有的貨物運輸服務水準。本研究主要透過禁制搜尋法求解即時資訊下的車輛巡迴路線問題，並藉由模擬指派模式 DYNASMART 進行評估。

研究中對車輛路線問題的假設如下：

1. 僅考慮某一特定位置的場站中心。
2. 車輛無容量限制，且為單一車輛。
3. 研究中採用不對稱之路網結構。
4. 各需求點及需求量为已知。
5. 車輛在接受即時資訊後，必須到下一個需求點才可進行路線的轉換。

本研究的目的可歸納如下：

1. 探討動態路線問題的觀念與解法。
2. 透過禁制搜尋法求解動態路線問題。
3. 進行路線規劃模式與模擬指派模式之結合，利用模擬指派模式反應現有的交通狀況。
4. 探討在即時資訊下動態 VRP 所產生的路線效益。
5. 討論在事故發生的情況下，動態 VRP 所產生的路線效益。

1.3 研究方法與流程

本研究主要應用禁制搜查法求解即時資訊下的動態車輛路線問題，並透過模擬指派模式 DYNASMART 進行實際的車流及車輛行駛的模擬，其研究流程如圖 1 所示，主要內容及進行流程如下所示：

1. 文獻回顧：除針對傳統車輛巡迴路線問題及其解法進行回顧外，更針對即時資訊下車輛巡迴路線問題，及禁制搜尋法演算法進行回顧。
2. 程式撰寫：由於模擬指派模式 DYNASMART 是由 Fortran 程式語言所撰寫，因此本研究在撰寫程式的部分亦使用相同的程式語言，以除去因不同程式間資料轉換、使用上可能發生的問題。
3. 國際例題測試：從相關文獻及網際網路上搜集相關例題與測試結果，以了解禁制搜尋法演算法的效率及正確性。
4. 實例測試與分析：研究中將以台中市路網進行實例測試，以瞭解禁制搜尋法提供即時路線的效益。
5. 結論與建議：根據分析所得之結果，提出具體的結論與後續研究之建議。

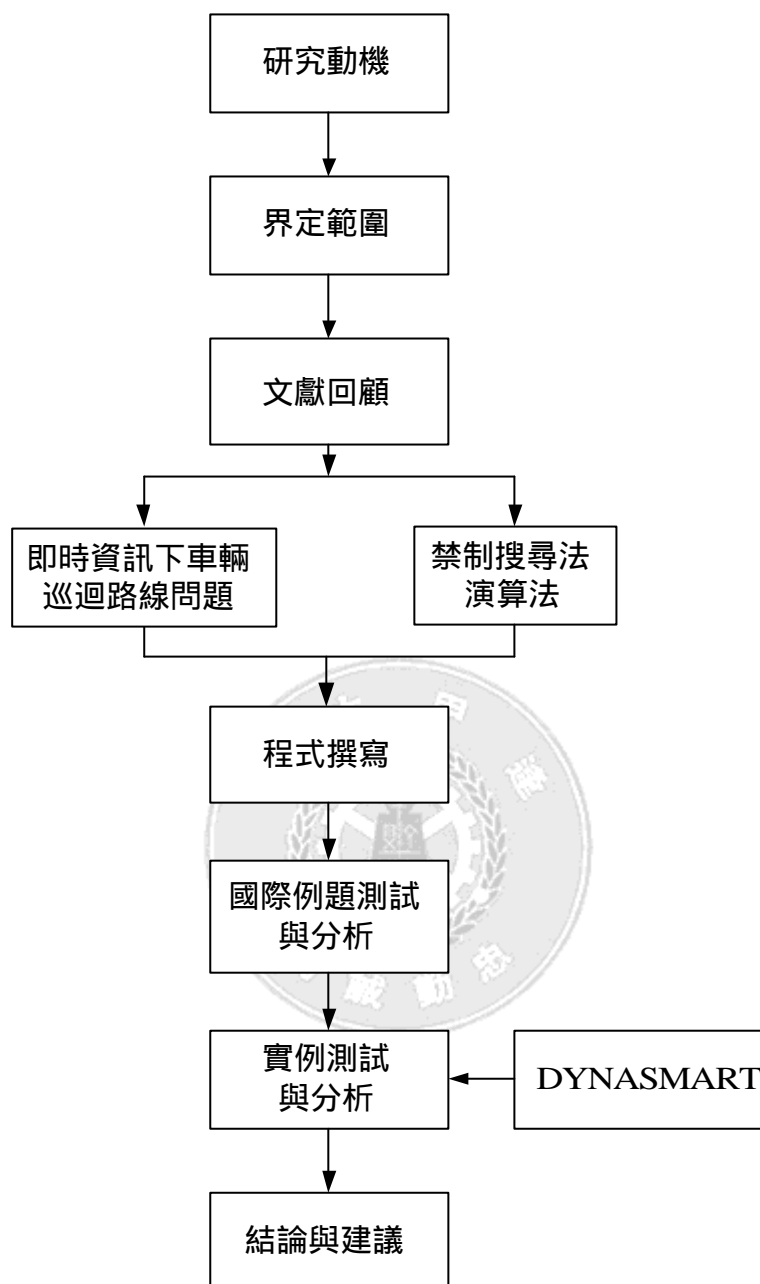


圖 1 研究流程圖

1.4 章節敘述

本節將敘述後續章節內容，第二章主要為文獻回顧，其中針對不同的 VRP 問題及其常用的解法進行回顧，並討論禁制搜尋法在 VRP 問題上的應用、與模擬指派模式 DYNASMART；第三章主要討論即時

資訊下動態 VRP 模擬評估架構之建立與如何應用 Tabu Search 求解動態 VRP 的問題；第四章透過國際例題的求解，以驗證本研究模式的正確性；第五章則針對不同的車輛型態與需求點數，在考慮不同的交通資訊屬性下，進行即時資訊下動態 VRP 之分析與實驗；第六章為結論與建議。



第二章 文獻回顧

本研究主要將文獻回顧分為四部分，第 2.1 節探討靜態的車輛巡迴路線問題；第 2.2 節探討車輛巡迴路線問題常用的解法；第 2.3 節討論動態的車輛巡迴路線問題；第 2.4 節針對禁制搜尋法進行相關文獻的回顧；第 2.5 節則對模擬指派模式 DYNASMART 進行說明。

2.1 車輛巡迴路線問題 (Vehicle Routing Problem)

最基本的車輛巡迴路線問題即為旅行推銷員問題 (Traveling Salesman Problem, TSP), TSP 問題的定義為給定一路網 $G=(N,A,C)$, 其中 N 為節點之集合, A 為節線之集合, $C=(C_{ij})$ 為一成本矩陣, 其中 C_{ij} 代表由節點 i 到節點 j 之成本; TSP 問題即是以最小成本, 求由起始點開始, 通過集合 N 中所有節點恰一次, 並回到起始點的問題。而車輛巡迴路線問題即假設給定一路網 $G=(N,A,C)$, 其中 N 為所有節點 $\{N_0, N_1, N_2 \dots N_n\}$ 的集合, 其中 N_0 表示為場站中心 (depot), $N_1, N_2 \dots N_n$ 代表各需求點, A 為連接各節點的節線集合, $C=(C_{ij})$ 為一成本矩陣, 其中 C_{ij} 代表由需求點 i 到需求點 j 之路線成本。

車輛巡迴路線問題主要假設車輛在有容量限制及可允許的巡迴時間限制下, 求出通過所有需求點且總成本最小的數條巡迴路線, 其中巡迴路線需求點需求量的總和不可超過車輛的容量限制, 巡迴路線的旅行時間不可大於可允許的巡迴時間限制; 同時每輛車必須由場站中心出發, 服務過若干個需求點再回到場站中心; 此外各需求點只能由一輛車來服務, 且每個迴路不可有子迴路存在。

因此根據 Fisher 和 Jaikumar (1981) 該問題可由以下模式所構建 :

$$\text{Min} \sum_{ijk} C_{ij} x_{ijk} \quad (1)$$

Subject to

$$\sum_i a_i y_{ik} \leq b_k, k = 1, \dots, K \quad (2)$$

$$\sum_k y_{ik} = \begin{cases} K, & i = 0 \\ 1, & i = 1, \dots, n \end{cases} \quad (3)$$

$$y_{ik} = 0 \text{ or } 1, \quad \begin{matrix} i = 0, \dots, n \\ k = 1, \dots, K \end{matrix} \quad (4)$$

$$\sum_i x_{ijk} = y_{jk}, \quad j = 0, \dots, n \quad (5)$$

$$\sum_j x_{ijk} = y_{ik}, \quad i = 0, \dots, n \quad (6)$$

$$\sum_{ij \in S \times S} x_{ijk} \leq |S| - 1, \quad \begin{matrix} S \subseteq \{1, \dots, n\} \\ 2 \leq |S| \leq n - 1 \end{matrix} \quad (7)$$

$$x_{ijk} = 0 \text{ or } 1, \quad \begin{matrix} i = 0, \dots, n \\ j = 0, \dots, n \end{matrix} \quad (8)$$

其中各常數及變數定義如下

常數：

K ：車輛數編號

n ：需求點的數目，當 $n = 0$ 時為場站； $n = 1$ 至 n 時，為各需求點。

b_k ：車輛 k 的容量

a_i ：需求點 i 的需求量

c_{ij} ：從需求點 i 到需求點 j 的旅行成本

變數：

$$y_{ik} = \begin{cases} 1, & \text{當需求點 } i \text{ 是由車輛 } k \text{ 服務。} \\ 0, & \text{不符合上列情況時。} \end{cases}$$

$$x_{ijk} = \begin{cases} 1, & \text{當車輛 } k \text{ 從需求點 } i \text{ 服務至需求點 } j。 \\ 0, & \text{不符合上列情況時。} \end{cases}$$

方程式(1)為目標函數，為總成本最小的巡迴路線；方程式(2)為車輛容量限制式，表示各車輛行經的需求點其需求量總和不得超過該輛車的服務容量；方程式(3)表示除場站外，各需求點只能被一輛車服務一次；方程式(4)說明 y_{ik} 為 0,1 變數，當 y_{ik} 為 1 時，代表需求點 i 是由車輛 k 服務；方程式(5)與(6)，限制車輛 k 從需求點 i 服務至需求點

j 時的服務次數；方程式(7)為避免產生子迴路的限制式；方程式(8)說明 x_{ijk} 為 0,1 變數，當 x_{ijk} 為 1 時，代表從需求點 i 服務至需求點 j 的路線是由車輛 k 所服務。

由於 VRP 該問題之複雜度屬於 NP-Hard，在透過數學規劃方式進行求解時，其問題規模變大時，需要較長的運算求解時間，因此多應用啟發式演算法以解決類似的問題。

根據 Bodin、Golden、Assad 及 Ball (1983) 文獻可知，由傳統的 VRP 中，衍生出不同型態的 VRP，以下將簡單的介紹之。

1. 多場站車輛巡迴路線問題 (Multi-Depot Vehicle Routing Problem, MDVRP) :

MDVRP 的問題主要在探討同時擁有 M 個場站中心的 VRP 問題，其目標為求出通過所有需求點且總成本最小的數條巡迴路線，每輛車由場站中心出發後，服務過若干個需求點必須回到原場站中心，而其他的基本假設與限制條件與 VRP 的問題差異不大。

2. 多車種車輛巡迴路線問題 (Fleet Size and Mixed Vehicle Routing Problem, FSMVRP) :

FSMVRP 主要探討在不同型式車輛下的 VRP 問題，其中不同型式的車輛擁有不同的固定成本及巡迴路線成本，而固定成本主要為旅行成本，變動成本則為燃料、維修費用、人事成本等；該問題主要希望各型式的車輛能有效的被使用，資源能夠有效的分配，其目標亦是求出總成本最小的巡迴路線。

3. 隨機型車輛巡迴路線問題 (Stochastic Vehicle Routing Problem, SVRP) :

SVRP 的問題主要是由確定型 VRP (deterministic VRP) 衍生而來的，其主要的限制條件如下：

- (1)顧客的需求為一已知機率分配下的隨機變數。
- (2)巡迴路線必須在需求點由未知變成已知前設計完成。
- (3)問題主要為求解最小化的期望旅行成本，而當某些需求點無法進行服務時，也許會增加額外的成本。

由於隨機需求點的加入，使得該問題在目標式與限制式中可能存在非線性的關係，因此在問題的求解上更為困難；而後續的研究除考

慮需求點為不確定性外，更討論在需求量及旅行時間為不確定性的情況下 VRP 的問題。

除了上述三種不同的 VRP 問題外，其後尚有週期性車輛巡迴路線問題及時窗限制下車輛巡迴路線問題，分別敘述如下：

1. 週期性車輛巡迴路線問題 (Period Vehicle Routing Problem , PVRP):

傳統的 VRP 其車輛服務週期即為一天，而 PVRP 問題主要探討在 M 天的週期內，顧客至少被服務一次以上的 VRP 問題，其中車輛為單一車種 (homogeneous vehicle fleet)，車輛容量為一固定值，並有容量限制；而顧客每日的需求為已知，且每日只能被一輛車服務；PVRP 的問題主要為求解最小化的總旅行成本。

2. 時窗限制下車輛巡迴路線問題 (Time Window Vehicle Routing Problem , TWVRP):

VRPTW 的問題與傳統 VRP 問題最大的差異在於時窗的限制式，其主要限制需求點的服務時間，在時窗的限制下可分為硬時窗與軟時窗兩種，硬時窗是指車輛必須在需求點要求服務時間的上下界之間進行服務，軟時窗則是違反需求點的時窗限制，當違反時窗限制必須加入一懲罰值，其餘的假設則與傳統 VRP 問題差異不大。

2.2 車輛巡迴路線問題的求解

由於 VRP 問題已被證明為 NP-Hard 的問題，其問題的複雜度將隨著節點數的增加而成指數的成長，因此一般皆用啟發式解法進行問題的求解，根據 Bodin、Golden、Assad 及 Ball (1983) 整理將 VRP 求解的方法歸納成七類：

1. 先分群-再排路線 (Cluster First-Route Second)
2. 先排路線-再分群 (Route First-Cluster Second)
3. 節省法/插入法 (Saving/Insertion)
4. 改善法/交換法 (Improvement/Exchange)
5. 數學規劃法 (Mathematical-programming-based)
6. 人機互動法 (Interactive optimization)

7.最佳解解法 (Exact procedures)

其中前 4 種解法和第 7 種解法，為過去較常使用於 VRP 問題的求解，而第 5 及 6 種則為近期所發展的解法。其說明歸納如下：

1.先分群-再排路線 (Cluster First-Route Second)：

即將所有需求點 (或是節線) 分為數個群體後，再從各個群體中設計出較經濟的路線。較為常用的是 1974 年由 Gillett 和 Miller 提出的掃描法 (Sweep Method)，其利用極座標來表示各需求點的位置，以場站中心為中心點，取任意需求點為起始點，以車輛容量為限制條件，依照該需求點之順時鐘 (或逆時鐘) 方向，進行服務區域的分群，再藉由 TSP 的演算法進行巡迴路線的求解；而由於選取不同起始點會得到不同的路線結果，因此可將不同的需求點設為起始點求解，再取其最佳解。

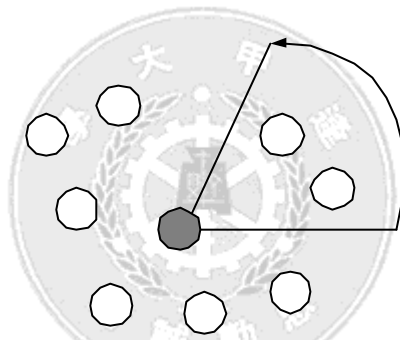


圖 2.1 掃描法分群圖

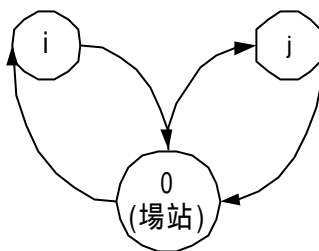
2.先排路線-再分群 (Route First-Cluster Second)：

即先建立一條通過所有需求點的路線後 (通常為不可行解)，接著將該條路線分割成數條較小且為可行解的路線。

3.節省法/插入法 (Saving/Insertion)：

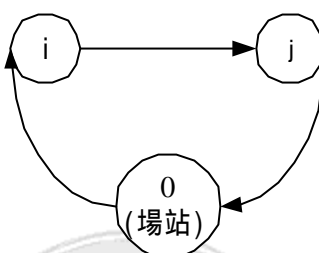
本方法主要先產生一條起始的路線解，計算各需求點插入的後路線成本所節省的數值，而最大的節省值即為插入需求點的依據，根據以上步驟反覆進行已產生一條可行的路線解。圖 2.2 與圖 2.3 為一對稱性路網之節省值示意圖。圖 2.2 為該路網之起始狀態，圖 2.3 則是將 i 、 j 進行配對後的狀況。

節省值 $S_{ij} = (C_{0i} + C_{i0} + C_{oj} + C_{j0}) - (C_{0i} + C_{ij} + C_{j0}) = C_{i0} + C_{oj} - C_{ij}$



路線成本： $C_{0i} + C_{i0} + C_{oj} + C_{j0}$

圖 2.2 節省值示意圖(原路線)



路線成本： $C_{0i} + C_{ij} + C_{j0}$

圖 2.3 節省值示意圖(節省後)

4.改善法/交換法 (Improvement/Exchange) :

在應用改善法 (或是交換法) 前首先產生一可行的起始解, 透過該路線的改變產生新的路線解, 而當所產生的路線其成本較小時, 則進行路線的改善, 並重覆上述步驟直到路線成本無法改善為止; 最常被應用的為 2-opt/3-opt, 2-opt 的交換示意圖如圖 2.4 及圖 2.5 所示。

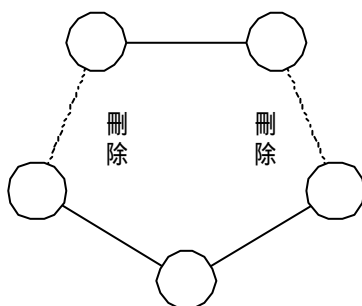


圖 2.4 2-opt 交換示意圖(原路線)

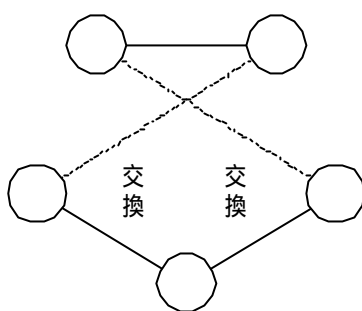


圖 2.5 2-opt 交換示意圖(交換後)

5.數學規劃法 (Mathematical-programming based) :

主要利用數學規劃的方式，透過數學式描述問題並加以求解，並計算鬆弛後的數學式。

6.人機互動法 (Interactive optimization) :

此法在問題的求解上，包含了人機互動的方式，其主要藉由決策者過去的經驗，根據相關知識及直覺的判斷，以進行模式內參數的設定與修正。

7.最佳解解法 (Exact procedure) :

一般使用分支界定法 (Branch and Bound)、動態規劃法 (Dynamic Programming)、切割平面法 (Cutting Plane Algorithm) 等，進行最佳化的求解。

由於過去的啟發式解法都是先產生一起始解，再應用不同改善方式求得較佳解，但往往無法跳脫區域最佳解，於是逐漸發展出可跳脫區域解的巨集啟發式解法 (Meta-Heuristic)，如模擬降溫法 (Simulated Annealing)、門檻接受法 (Threshold Accepting Method)、噪音擾動法 (Noising Method)、搜尋空間平滑法 (Search Space Smoothing) 與禁制搜尋法 (Tabu Search) 等。

以下將就巨集啟發式解法做一簡單的介紹。

1. 模擬降溫法 (Simulated Annealing , SA) : 模擬降溫法對於一個最小化的問題而言，當陷入區域最佳解時，透過一機率性的接受法則，使其接受較差的解以跳脫區域最佳解，以搜尋其他鄰域。
2. 門檻接受法 (Threshold Accepting Method , TA) : 門檻接受法

的解題架構來自於模擬降溫法，兩者最大的差別在於門檻接受法使用門檻值做為確定性的接受法則。

3. 噪音擾動法 (Noising Method , NM) : 噪音擾動法對於一個最小化的問題而言，當陷入區域最佳解時，使用隨機產生的噪音量(Noise)，擾動原本的成本函數，在對擾動後的問題重新求解。(韓復華等人，1996)
4. 搜尋空間平滑法 (Search Space Smoothing , SSS) : 搜尋空間平滑法希望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。
5. 禁制搜尋法 (Tuba Search , TS) : 禁制搜尋法在搜尋完鄰域時，往最佳的方向進行移步(move)，並將搜尋的解記錄在禁制名單(Tabu List)內，每次的移步都需考慮是否被禁制，以避免重覆的搜尋。

2.3 動態車輛巡迴路線問題 (Dynamic Vehicle Routing Problem)

根據 Psaraftis(1995)對靜態及動態車輛路線的定義，當一個 VRP 問題被稱為”動態”時(如：即時性(real-time)、線上型(on-line))，所有資訊在決策者進行路線規劃的同時會即時的更新；反之，當所有資訊在路線決定前已被取得且不再改變，此類問題便稱為靜態。Psaraftis(1995)更歸納出問題本質為靜態卻被誤解為動態的問題，其說明如下：

1. 依時性 TSP 問題 (The Time-Dependent TSP) :

依時性的 TSP 問題與傳統 TSP 問題最大的差別在於其擁有各個時間點的旅行時間矩陣，雖然路線成本會隨著時間改變，但由於其資訊在設計路線前為已知，且路線在決定後並不會改變，不須重新進行求解，因此其被歸為靜態的問題。

2. 機率型 TSP 問題 (The Probabilistic TSP) :

在 PTSP 的問題裡，其旅行時間為一固定值，需求點為一已知的機率值，由於 PTSP 其旅行時間矩陣與機率值皆為已知且不會改變，

而巡迴路線決定後並不會再進行調整，因此被歸為靜態的問題。

3. 隨機旅行時間的 VRP 問題 (The Vehicle Routing Problem with Stochastic Times):

在旅行時間與服務時間為不確定的情況下求解 VRP 問題，即為隨機旅行時間的 VRP 問題，但該路線在設計完後並不會再進行求解 (reoptimize) 的動作，且其獲得的資訊並沒有即時的特性，因此被歸為靜態的問題。

而由於資訊的提供方式對動態問題有明顯的影響，因此 Psaraftis(1995)將資訊依照不同的屬性區分為以下四種：

1. 資訊的演進 (evolution of information): 分為靜態與動態兩種，當資訊在路線決定前已被取得且不再改變為靜態，反之，則稱為動態。
2. 資訊的品質 (quality of information): 主要分為已知確定性 (known-deterministic)、預測(forecast)、機率值(probabilistic)、未知(unknown)。
3. 資訊的可獲得性(available of information): 分為區域性(local)與全域性 (global) 兩種；區域性的資料只有在區域範圍內的使用者才能獲得，而全域性的資料可透過通訊技術（如：無線電廣播）使區域範圍外的使用者獲得相關的資訊。
4. 資訊的處理(processing of information): 分為集中(centralized)與分散 (decentralized) 兩類；集中是指所有的資訊由特定單位集中處理，反之則稱為分散。

由於靜態的問題並無法確實的呈現現實的貨車營運狀況，因此動態車輛巡迴路線的問題逐漸受到重視，目前在求解動態車輛巡迴路線問題上，尚有以下兩種型態：

1. 線上型車輛路線問題 (On-Line VRP):

線上型車輛路線問題其與靜態的問題最大的差別在於求解線上型車輛路線問題的過程中，其顧客的需求與道路狀況是無法事先得知的，必須隨著時間的經過逐一獲得，且對於尚未出現資料並無法事先得知，已出現的顧客需求則依其順序提供車輛進行服務，並滿足需求的服務時間及車輛的最大載重、路線總時間等限制條件。且該問題通常不考慮旅行

時間的動態變化，即表示各路段的旅行時間為一固定值，而僅考慮需求點、需求量變化的情況下，進行問題的求解。

2. 即時資訊下車輛路線問題 (VRP under real-time information) :

即時資訊下車輛路線問題與線上型車輛路線問題最大的差別在於路段的旅行時間會著隨時間的變動而改變，而非一固定值，且路線的旅行時間並無法事先獲得，因此較能符合現實的交通狀況。

以下將就國內外對於求解動態問題的文獻做一簡單的說明。

Taniguchi 等人(1999)根據動態交通模擬模式 BOX model 提出一機率型時窗限制下車輛巡迴路線問題 (probabilistic VRP-TW) 模式，研究中在進行巡迴路線的求解時，考慮了路段旅行時間變動的特性，在測試路網中發現，該模式與預測型時窗限制下車輛巡迴路線問題 (forecast VRP-TW) 相比較後，發現其確實能降低巡迴路線時間。

呂英志(2002)為了考慮路段旅行時間變動的特性，應用機會限制模式 (Chance Constrain Model) 進行模式的構建，透過數學規劃軟體 CPLEX 進行問題的求解，並透過 DYNASMART 進行路線的評估，但當需求點增加時，模式中的子迴路限制式也隨之大幅增加，使得求解的速度變慢，無法在有效的時間內獲得最佳解。

2.4 禁制搜尋法 (Tabu Search)

禁制搜尋法為一高階的萬用啟發式解法，專門用來求解組合最佳化的問題。目前已應用的領域如排程、車輛旅途、旅行銷售員問題(TSP)等組合最佳化問題，不同於過去的啟發式演算法，在禁制搜尋法中，主要包含移步 (Move)、禁制名單 (Tabu List)、渴望準則 (Aspiration Level)、候選名單 (Candidate List) 及搜尋停止準則 (Stopping Criterion) 五大模組，其主要透過禁制名單及渴望準則決定出下一次可能移步的候選名單，以進行最佳的移步動作，並在此一動作循環下配合搜尋停止準則以求得最佳解，且該方法不易陷入先前的區域最佳解，並能避免在求解過程中可能發生的循環(cycling)，進而得到全域的最佳解；圖 2.6 為禁制搜尋法之流程圖。

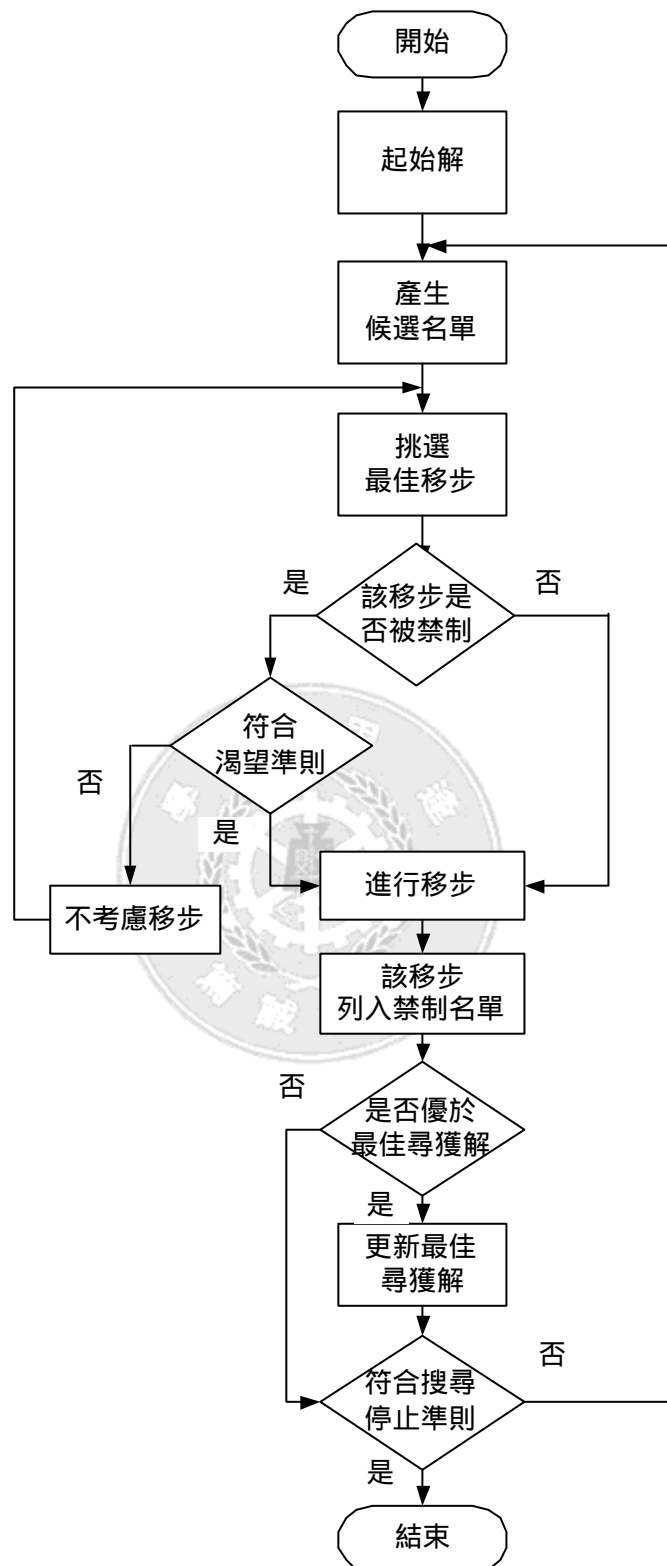


圖 2.6 禁制搜尋法之流程圖

(本研究整理)

以下針對禁制搜尋法中五個模組進行說明。

1. 移步 (Move) : 禁制搜尋法首先藉由隨機或是其他方式產生一起始解，藉由各種移步方式搜尋其鄰近解 (Neighborhood)，並選擇最好的鄰近解進行移步的動作，而目前常用的方式為 2-opt/3-opt。

2. 候選名單 (Candidate List) : 候選名單即為可以進行移步屬性的集合，候選名單內的每一個被允許的移步皆對應一鄰近解，禁制搜尋法中的移步，即是由這些名單中挑選最好的鄰近解，進行移步的動作。

3. 禁制名單 (Tabu List) : 在進行移步的同時，禁忌名單會將其移步的屬性紀錄下來，以作為未來進行移步時參考依據；而禁忌名單的大小，決定該移步被禁制的期間，當禁制的期間愈長時，愈能避免在求解過程中可能發生的循環，因此較不容易陷入區域的最佳解。

4. 渴望準則 (Aspiration Level) : 當某一移步被禁制時，但其所獲得的解優於目前搜尋而的最佳解時（設定為當目標函數值獲得改善），則透過渴望準則將其禁制的狀態解除，使該移步可以進行。

5. 搜尋停止準則 (Stopping Criterion) : 搜尋停止準則通常分為下列幾種：(1) 預設運算過程中，最大允許遞迴數；(2) 預設目標函數值持續未改善之最大允許遞迴數；(3) 預設允許 CPU 計算時間；(4) 預設達到可接受目標函數值範圍；一但達到預設的範圍即停止搜尋，而所搜尋的最佳尋獲解即為最佳解。

在禁制搜尋法中主要透過記憶 (Memory) 來儲存移步的屬性，其記憶可分為短期記憶 (Short Term Memory) 與長期記憶 (Long Term Memory) 兩階段。在短期記憶階段，利用禁制限制 (Tabu Constraints) 來限制搜尋狀態，以避免重複的搜尋增加搜尋時間；而長期記憶階段則使用強化性 (Intensification) 與多樣化性 (Diversification)，其中強化性主要為限制移步的條件，而多樣化性主要根據過去求得的區域最佳解，配合不同的移步策略，將搜尋帶入新的解域以求得全域的最佳解。圖 2.7 為 Fred Glover 與 Manuel Laguna 提出的短期記憶的使用方式 (Short term memory operation)。

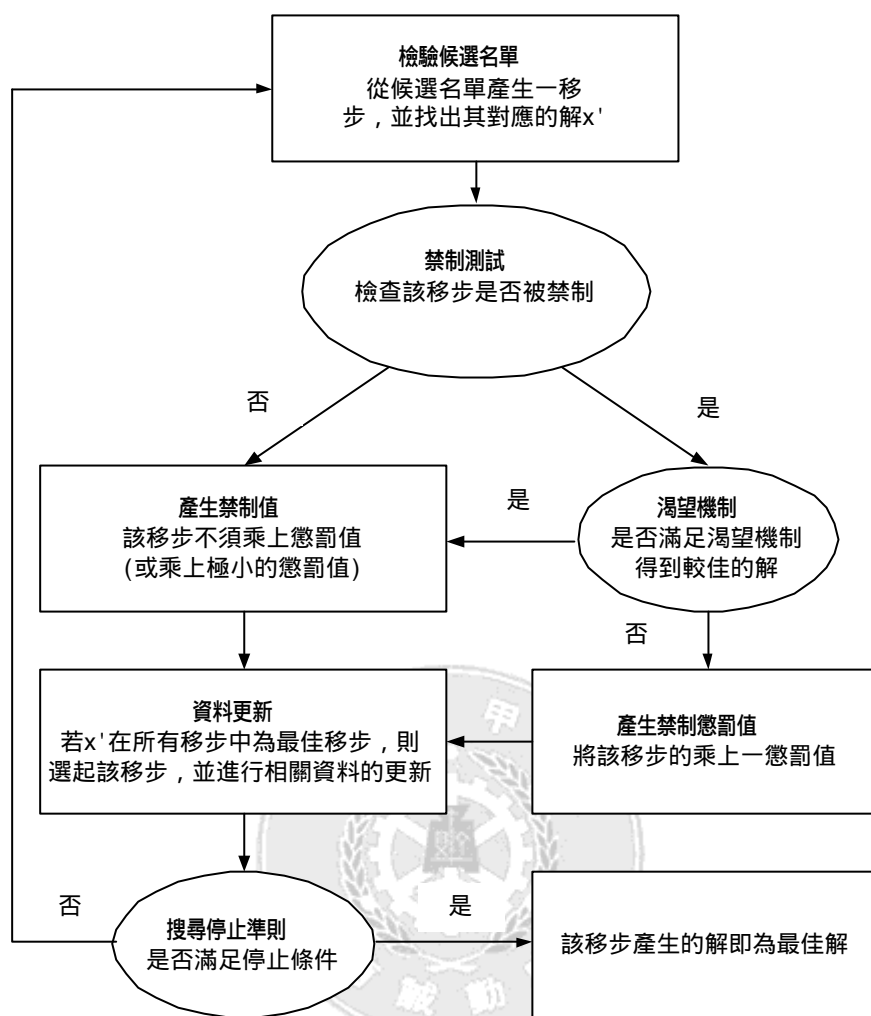


圖 2.7 短期記憶的使用方式

(Glover, and Laguna, 1997)

其短期記憶使用方式是先由候選名單中產生一移步，並找出其對應的解 x' ，檢驗該移步是否被禁制，若被禁制，則考慮符不符合渴望準則，若亦不符合渴望準則，則將該移步的加上一懲罰值，以降低該移步被選取的機會；當該移步並未禁制，則選取該移步，在所有的候選移步中，挑選最佳的移步，如此反覆進行，直到達到停止的準則，以求得最佳解。

以下針對過去應用禁制搜尋法求解的相關文獻進行回顧。

Gendreau, Hertz 和 Laporte (1994) 應用禁制搜尋法求解車輛巡迴路線問題，其使用過去為求解 TSP 問題而發展的兩個方法 GENI 和

US 產生鄰解，GENI 是指需求點插入的路線上必須包含與其最近的點；US 是一種事後最佳化的方式，其使用方法是應用 GENI 針對每一個需求點進行移除及插入的動作；根據結果顯示禁制搜尋法的績效值明顯優於其他種啟發式解法。

Renaud, Laporte 和 Boctor (1996) 考慮在車輛容量及路線長度的限制的情況下，應用禁制搜尋法求解多場站的車輛巡迴路線問題 (Multi-Depot VRP)，並發展一演算法 FIND，在該演算法中包含了 Fast improvement、Intensification 及 Diversification 三個階段，而每個階段都包含了 1-route、2-route 與 3-route 移步法則；該演算法測試了 Chrisofides, Eilon, Gillette 和 Johnson 的 11 個典型的問題，與 Chao et al. 的 12 個新的問題，發現該演算法能計算出較好的解。

Brandao 和 Mercer (1997) 以禁制搜尋法求解多旅次車輛巡迴路線問題與排程問題 (multi-trip vehicle routing and scheduling problem, MTVRSP)，該問題基本的限制與一般車輛巡迴路線問題並沒有太大的不同，額外的限制如下所述：1. 每輛車一天之中允許被使用一次以上；2. 各需求點有時窗限制；3. 考慮不同容量的車種；4. 特定的需求點必須由特定的車輛服務；5. 必須考慮到駕駛人的排程；6. 考慮卸貨的時間等。

Barbarosoglu 和 Ozgur (1999) 應用禁制搜尋演算法求解車輛巡迴路線問題，並在土耳其以一實例進行求解，該篇文獻主要探討一新的鄰近搜尋法稱之 TANE 與 TANEC，TANE 主要是給定一搜尋範圍，以目前的解進行搜尋，其中最佳的非禁制移步 (non-tabu move) 是被決定的；而 TANEC 只要是尋找最佳的非禁制移步，並找出最小的目標函數值，其中移步的方式皆是採用 2-opt 的方式進行。

陳勝男 (1996) 使用兩階段進行求解車輛巡迴路線問題，在第一階段利用空間填滿曲線產生起始解，第二階段則透過禁制搜尋法，結合插入法及路線節點交換法作為改善法，在求解 Fisher, Christofides, Eilon, Mingozzi 和 Toth 提供的 10 個例題中，可得到 4 個最佳解。

林明俊 (1998) 在車輛巡迴路線問題中考慮了隨機的需求、隨機的運送時間和不同的車輛容量，應用禁制搜尋法求解隨機環境下多車種派車的問題，但由於作者未能從模式中找出確切解，因此僅針對較

小的例題進行驗證，但無法得知解的品質。

王文貞(1997)使用兩階段求解一圖書配送車輛排程問題時，第一階段以節省法產生起始解，第二階段則以插入法與節點交換法做為路線改善的方式，在模式正確性的驗證上，作者針對假設性的路網資料，以 LINDO 進行最佳解的求解，進行比對，在四個範例中，有三個可達到最佳解，而未達最佳解的範例，其誤差為 3.1%。

黃金智（1999）求解一隨機的需求量、車輛數有限之多車種隨機車輛途程問題，該研究以最鄰近解法（Nearest Neighbor）作為禁制搜尋法之起始解，而利用 2-exchange、2-swap 或插入法為搜尋鄰解的方式。

表 2.1 禁制搜尋法於車輛巡迴路線問題之比較（本研究整理）

題目	問題型式	起始解	路線改善階段	停止準則
A Tabu Search Heuristic for the Vehicle Routing Problem (Gendreau, Hertz, Laporte, 1994)	靜態 VRP	Clarke and Wright (n 個顧客 n 個起始解)	GENI US	最佳解未改善之遞迴
禁忌搜尋法應用於車輛路線問題之研究(陳勝男, 1996)	靜態 VRP	空間填滿曲線	節點交換法 插入法	最大遞迴
圖書配送車輛排程問題之研究(王文貞, 1997)	靜態 VRP	節省法	插入法 節點交換法	最大遞迴
隨機環境下多車種派車問題之研究(林明俊, 1998)	隨機型 VRP	Clarke and Wright (n 個顧客 n 個起始解)	插入法	最佳解未改善之遞迴
隨機型車輛途程問題解法之研究(黃金智, 1999)	隨機 VRP	最鄰近解法 (Nearest Neighbor)	2-exchange 插入法 2-swap 2-opt	最大遞迴

2.5 交通模擬指派模式 (胡大瀛 , 1995)

由於本研究將透過模擬指派模式反應交通狀況，因此 2.5.1 節將針對現有的路網模擬指派模式進行回顧，其中並針對研究中所採用的 DYNASMART 模擬指派模式將於 2.5.2 節進行較詳細的回顧。

2.5.1 路網模擬指派模式

本節主要參考魏慶地 (1999) 及黃燦煌 (2000) 針對現有路網模擬指派模式的整理結果。

1. CORSIM 模式：

CORSIM (Corridor Traffic Simulation Model) 為美國聯邦公路總署 (FHWA) 所發展之微觀隨機模擬模式，其主要包括高速公路模擬模式 FRESIM，與地面道路的模擬模式 NETSIM，透過介面節點將次路網連結而成；車輛於介面節點離開次路網並進入另一個路網中，每一次路網均有一區域儲存欲離開的車輛，直到能夠進入另一個路網中。CORSIM 利用跟車、車道變換與等候線紓解等方法，模擬車輛於高速公路與匝道中的行為，而個別駕駛人/車輛的特性則是依照所設定之駕駛型態隨機給予。CORSIM 可模擬號誌之運作、公車、車輛偵測器、匝道儀控；此外，對轉向車輛、共用車道 (Shared Lanes) 及溢流等亦有詳細處理。由於 CORSIM 尚缺乏模擬與評估 ATIS 策略的功能，因此目前正發展結合交通指派與控制策略的功能。

2. PARAMICS 模式：

PARAMICS (Parallel Microscopic) 是由英國愛爾蘭愛丁堡平行計算中心 (Edinburgh Parallel Computing Center) 於 1992 年所發展之微觀模擬模式，該模式發展之初是於工作站 UNIX 作業系統下進行，並以網路的方式以平行處理方式對車流進行模擬；PARAMICS 可模擬車輛之跟車行為與車道變換行為，及匝道儀控、號誌運作、路徑資訊等，使用者只需透過適當的參數修訂，即可進行交通系統的模擬，此外，PARAMICS 可將模擬的過程實體化，使模擬的情境更加逼真。

3.CONTRAM 模式：

CONTRAM (Continuous Traffic Assignment Model) 是由英國運輸與道路研究實驗室所發展，主要在評估都會區內的交通管理措施，其模擬的方式有類似「逐步增量指派 (Incremental Assignment)」的觀念，可用以模擬停等車隊、旅行時間、旅行速度、延滯，並可透過模擬所得的車輛行駛時之燃油消耗，以評估交通對環境的影響。

4.CORFLO 模式：

CORFLO 為美國聯邦公路總署 (FHWA) 所發展之模擬模式，其主要包括三個模組：一個高速公路模擬模式 FREEFLO，與兩個地面道路的模擬模式 NETFLO1 與 NETFLO2；CORFLO 可模擬旅行時間、速度、延滯廢棄排放等，但無法模擬高速公路匝道儀控的功能，而目前多應用於評估高速公路與平面道路之交通績效。

5.METACOR/METANET：

METACOR/METANET 是由德國慕尼黑技術大學及法國運輸研究實驗室共同發展，其為一巨觀模擬模式，用以模擬高速公路交通狀況，METACOR 模式是由 METANET 模式所擴充，以進行平行幹道車流之模擬；METACOR/METANET 模式是採用流量守恒與動態的速度-密度關係以進行車流的模擬，該模式中包含了控制與動態交通指派模組，因此可模擬匝道儀控策略與路徑資訊導引。

2.5.2 DYNASMART 模擬指派模式

DYNASMART 為因應美國公路聯邦總署對 ATMS/ATIS 的規劃，所發展的綜合評估模式，其前身是 Chang et al. (1985) 用以模擬兩條平行道路的交通狀況之模擬模式，Jayakrishnan (1992) 將此一模式延伸至路網。Hu et al. (1992) 加強模擬與指派的整合，將交通控制策略加入此一模式，並改善模式運算效率。而根據 FHWA 的構想，模擬指派模式應具備下列的功能：

- 1.同時模擬市區街道與高速公路的交通。
- 2.同時進行多車種的指派。
- 3.反應個別駕駛者對交通資訊的反應。

4.能擷取與產生不同類型的交通資訊。

DYNASMART 主要是利用交通模擬來建立相對應的交通型態，並用以評估交通路網在不同假設下的績效，其採用模擬指派 (Simulation-Assignment) 的方式來指派依時改變的交通需求。迄今，DYNASMART 的應用，主要是以描述與評估在 ATIS/ATMS 架構下路網車流可能的變化；考慮的因素包含了交通資訊的產生與供給、交通控制策略、路線導引與駕駛者個體對系統的反應與決策。

如圖 2.8 所示，DYNASMART 包含了車流模式 (Traffic Flow Models) 的應用，路徑資訊產生 (Path Information Processing)、個體決策行為模式 (Individual Decision Behavioral Rules) 與可能的資訊供給法則 (Information Supply Strategies)。在這個架構下，旅次產生者所決定的出發時間 (Departure Time)，運具 (Model) 與使用路徑 (Route)，構成了一個三維的依時 OD 矩陣。DYNASMART 依時間次序，將私人車輛 (含汽、機車與貨車) 與公車陸續載入交通路網中，並按各車種的特性來移動車輛，直到車輛抵達目的地為止。在車輛移動的過程中，交通資訊的傳輸與使用，亦將針對車輛特性加以區分。DYNASMART 架構中，包含三個主要元素：

1. 交通模擬

模擬階段，分成兩大部份，一是在路段中車輛的移動，另一是車輛進入路口的行為，我們可將兩者定義為路段移動 (Link Movement) 與路口轉換 (Node Transfer)。路段移動的主要目的在於捕捉車輛移動的情形，因此車輛速度的決定將是考慮的重點。路口轉換須能精確的反應交通控制的限制與混合車流組成之特性。DYNASMART 提供了大部份現有交控措施，其中可分為兩大項，市區街道與公路系統。

2. 路徑資訊的產生

根據交通資訊產生的策略，不同的路徑資訊，可由不同的駕駛者於旅次前或行進中使用。在這個元素中，具備考慮轉向成本的 k 條最短路徑將被用在此一模式，以確保模式在使用上之彈性 (Ziliaskopoulos, 1994)。

3. 旅次產生的決策行為

模式中最大的優點是假設旅次產生者能根據自己的行為特性或派

遣中心的指示去選擇路徑，因此在這一個元素中，可以反應路徑或路線之選擇。

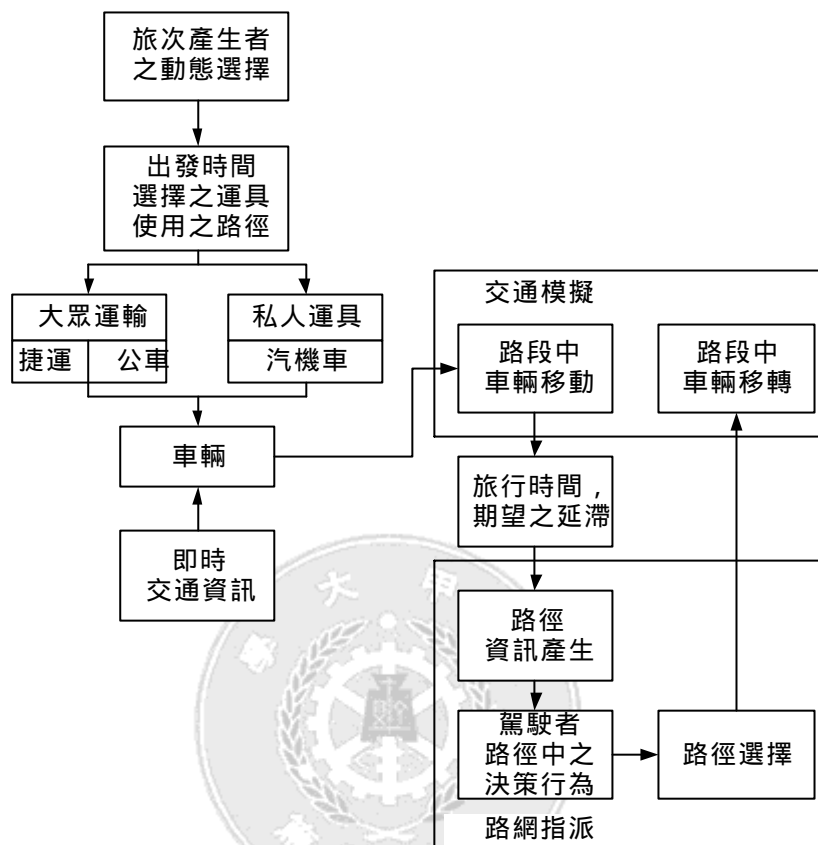


圖 2.8 DYNASMART 模式架構

DYNASMART 的功能結構，如圖 2.9 所示，其中包含了不同元素間的功能關係。圖中最基本的單元是車輛（或駕駛人）。在這個單元中包含了幾個不同的屬性：1.車輛靜態屬性：在車輛產生時建立，用以描述車輛靜態的特性，並且假設在模擬中是固定不變的項目，如車型、車內設備等；2.駕駛人個人社會經濟特性（behavior attributes）：駕駛者行為相關屬性反應出使用者的特性，這些個人社會經濟特性將應用於駕駛者路徑選擇的分析上。個人社會經濟特性如：年齡、性別等；3.執行過程屬性（run-time attributes）：執行中屬性是用於模擬中，登錄並改變駕駛者旅次中的統計資料，可提供事後的分析。

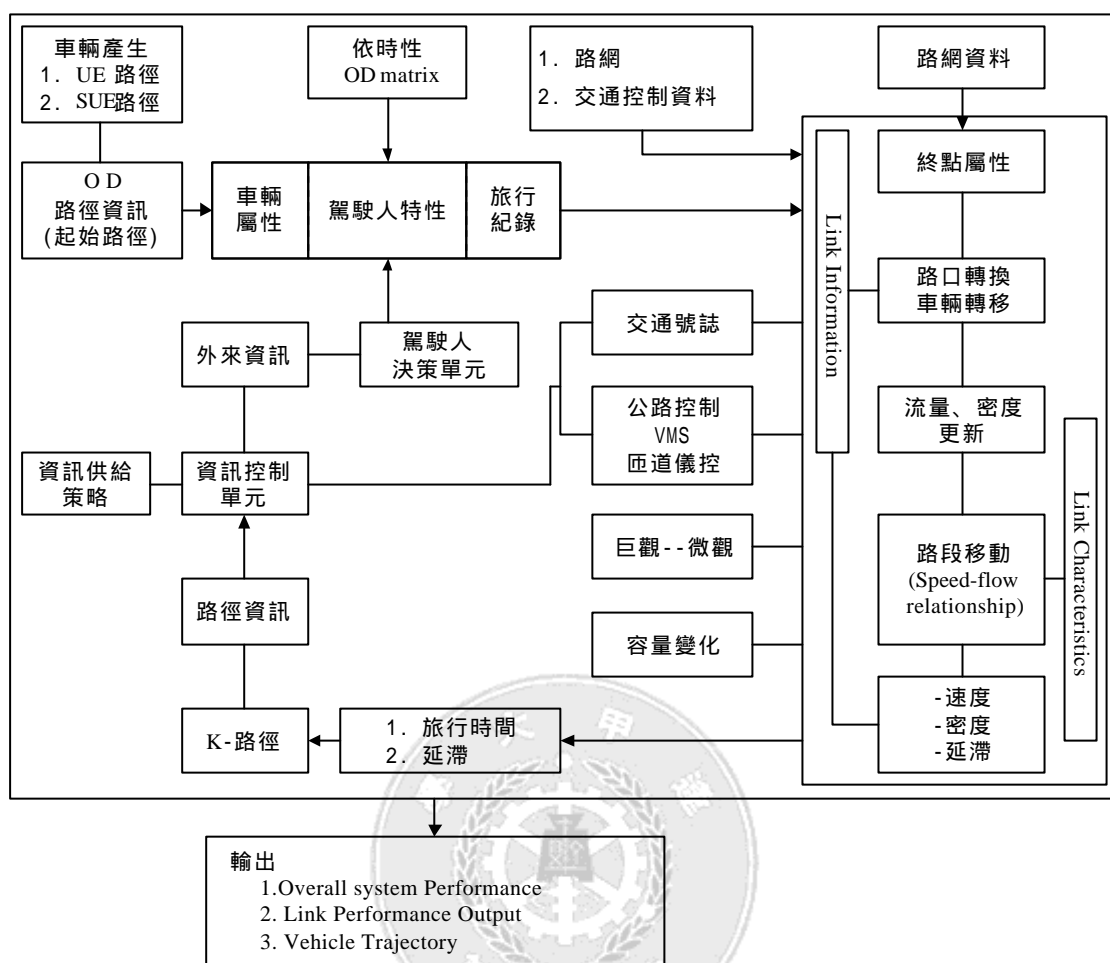


圖 2.9 DYNASMART 功能架構

第三章 即時資訊下動態車輛路線分析架構

本章針對即時資訊下動態車輛路線求解與評估分為兩部份進行探討，3.1 節為即時資訊下動態 VRP 模擬評估架構之建立；3.2 節討論如何應用禁制搜尋法求解動態車輛巡迴路線問題。

3.1 即時資訊下動態 VRP 模擬評估架構之建立

本節將先探討動態派遣策略架構，在構建即時資訊下動態派遣策略後，進而建立即時資訊下動態 VRP 模擬評估架構。

3.1.1 動態派遣策略架構

一般貨運車輛的指派有兩種，一種是出發前的指派，另一種是路途中指派，在出發前指派中，其服務的需求點可分為固定需求和變動需求兩種，在考慮固定需求及變動需求下，其車輛巡迴路線問題可由傳統的 TSP 或 VRP 方式求解。

而路途中的指派，其服務的需求點以即時性的需求為主，管理者在接受即時性的需求資訊後，考慮在滿足車輛容量限制及其允許服務範圍的限制條件下，將該需求點納入原始的路線，以產生新的巡迴路線，或是另行派遣車輛進行服務。因此，一商用車隊在車輛出發前可根據固定及變動需求來規劃起始的巡迴路線，而隨著即時性需求的產生，管理者可依照營運成本、營運服務水準，配合即時交通資訊的使用，重新調整以產生新的巡迴路線。其貨運車輛指派與巡迴路線研擬流程圖如圖 3.1 所示。

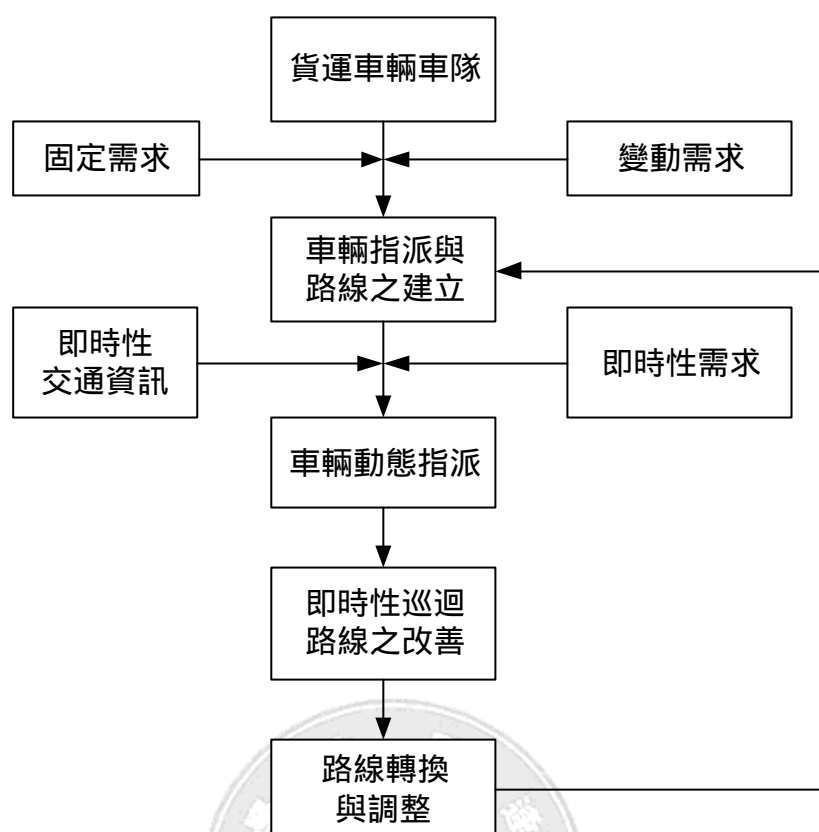


圖 3.1 貨運車輛指派與巡迴路線研擬流程圖

由於一般求解巡迴路線的演算法，無法解釋系統中部分因素的隨機特性（Gendreau et al., 1996），如：隨機需求量、隨機旅行時間等，加上一般 VRP 或隨機性 VRP 問題，並未考慮交通流量大小、交通號誌控制、路口轉向等交通因素，因此本研究採用模擬指派模式進行相關派遣策略的評估。

由於模擬指派係依據時間次序，將車輛陸續載入交通路網中進行模擬移動，直到車輛抵達目的地為止，此設計可以反應不同時間點的交通路網狀況，因此亦考量了一般 VRP 或隨機性 VRP 無法考慮的部分，確實反應交通路網的真實情況；其首先藉由讀入車輛、路網進行模擬移動，並根據貨運車輛出發前的固定需求及變動需求，進行起始路線的規劃與車輛的派遣，在透過交通模擬指派後，利用即時性的交通資訊與即時性的需求，根據即時性路徑與路段研擬原則，重新規劃路徑和路線，以進行車輛的派遣；其模擬式動態派遣策略研擬之架構如圖 3.2 所示。

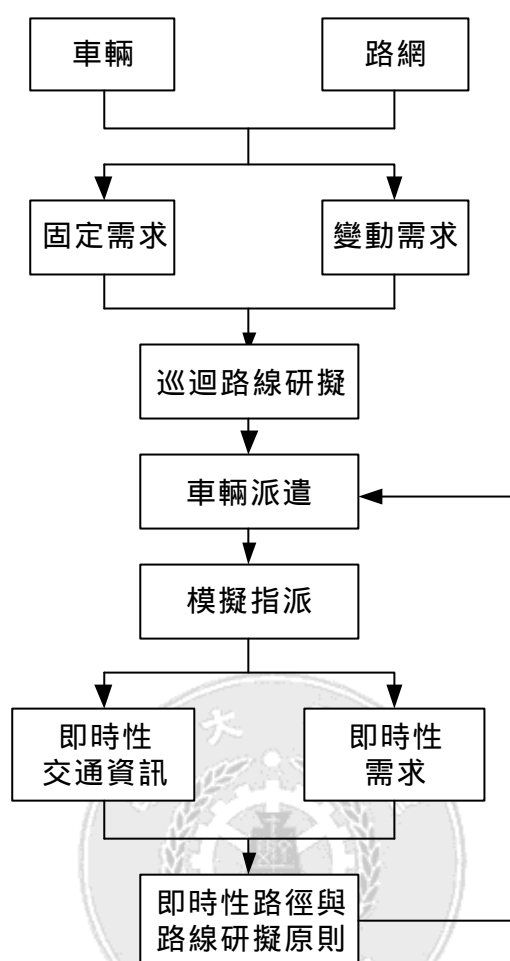


圖 3.2 模擬式動態派遣策略研擬之架構

3.1.2 即時資訊下動態 VRP 模擬評估架構

評估架構中首先設定依時性 OD 旅次，並配合歷史路徑使車輛均有可用路徑，其後將車輛陸續載入交通路網中進行模擬，此處之車輛亦包含商用車隊，進行模擬的過程中可獲得全面性的交通資訊，可視之為現有之車輛偵測器或偵測器與 GPS 車輛回報所蒐集的即時性流量分佈情形與旅行時間。

此一即時性交通分佈情形將配合路徑與巡迴路線研擬的策略，產生即時性的路徑資訊。而在路徑研擬及巡迴路線研擬的計算上是同時進行的，其中在路徑研擬策略部分係利用 K 條最短路徑進行計算，主要提供路網上的非商用車輛使用；而商用車輛的巡迴路線研擬則先以

Floyd-Warshall 演算法計算所有需求點，點到點的最短路徑，並利用禁制搜尋法求解 VRP，以進行巡迴路線的設計。

其得到的即時性資訊將提供給車輛使用，影響車輛的移動行為，進而造成交通狀況的改變，並藉由交通狀況之不同來評估此即時資訊之效果。其模式演算流程圖如圖 3.3 所示；即時資訊下動態派遣策略之演算流程說明如下：

步驟 1：初值設定

根據社經資料產生的屬性，配合歷史資料路徑資料庫，以 P_{rs}^j 代表時間 j 內，於起點 r 到迄點 s 所有可以使用的路徑。

步驟 2：載入車輛

根據車輛 i 之出發時間與起點，從 P_{rs}^j 中選擇路徑並指派給 i ，其中，商用車的路線為固定，將其輸入 DYNASMART。

步驟 3：交通模擬

DYNASMART 根據依時性的 OD 旅次，進行車流之模擬，車輛的移動根據交通路網與實際車流特性進行。

步驟 4：即時性資訊

根據道路上的偵測器與車輛上的定位系統，以獲得路網內相關路段的旅行時間與延滯，重新預估旅行時間矩陣。

步驟 5：路徑及巡迴路線研擬

步驟 5.1：路徑研擬

假設車輛為單一起訖點，以 KSP 演算法 (Ziliaskopoulos, 1994) 產生 K 條替代路線，並將路徑儲存供一般車輛使用。

步驟 5.2：巡迴路線研擬

先以 Floyd-Warshall 演算法計算所有點到點的最短路徑，以此時間矩陣配合禁制搜尋法求解 VRP 問題，以進行巡迴路線的設計，並將該路線提供給商用車輛使用。

步驟 6：回饋

將步驟 5.1 與步驟 5.2 所產生的路徑及巡迴路線交由車輛使用。其中步驟 5.2 中的 Floyd-Warshall Algorithm 為計算 All-pairs Shortest Path 之演算法，複雜度為 $O(n^3)$ ，其假設路網中共有 n 個節點，存在一 $n \times n$ 之矩陣 $d[i, j]$ ，依序檢查矩陣內每一元素 d_{ij} ，若 $d_{ij} > d_{ik} + d_{kj}$ ，

表示經由節點 k 的成本較低，此時令 $d_{ij} = d_{ik} + d_k$ ，繼續檢查其他元素，直到每一元素都檢查過為止。

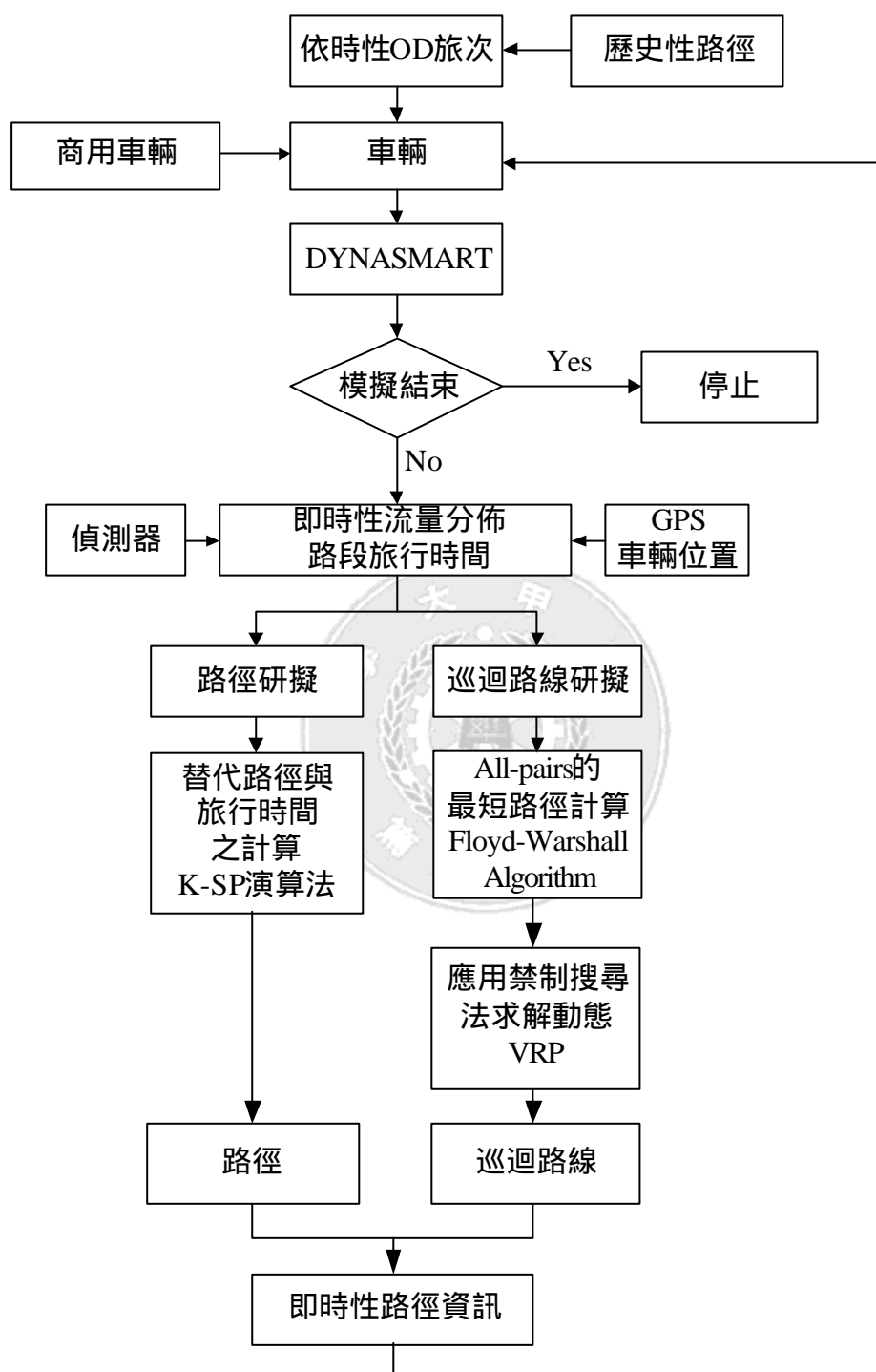


圖 3.3 即時資訊下動態派遣策略之演算流程

3.2 禁制搜尋法求解動態車輛巡迴路線問題

在求解即時資訊下動態車輛巡迴路線問題中，本研究主要應用禁制搜尋法的觀念，配合路線交換法，進行問題的求解，其中路線改善部份則是使用 2-opt 的方式進行。

本研究對於問題的基本假設如下：

1. 僅考慮某一特定位置的場站中心。
2. 車輛無容量限制，且為單一車輛。
3. 路網結構為不對稱路網。
4. 各需求點及需求量为固定且已知。

根據上述假設，本研究在求解動態 VRP 問題的主要演算流程構想如下，進一步的演算法將於第五章中介紹：

步驟 0. 令遞迴次數為 1，沒有任何移步被禁止

步驟 1. 需求點及需求量的確定

步驟 2. 讀入各需求點，點到點之間的旅行時間矩陣

步驟 3. 產生一條通過剩餘需求點的巡迴路線作為起始解

步驟 4. 使用 2-opt 進行移步

步驟 5. 在所有移步解中找出最佳的尋獲解 S

步驟 6. 決定最佳移步

步驟 7. 更新最佳移步解 S^* 、禁制名單、起始路線

步驟 8. 是否滿足停止準則，是，則停止，若否，則更新起始路線，並回到步驟 4

步驟 9. 剩餘需求點數量及位置的確定

步驟 10. 當剩餘的需求點數量為 0 時則結束，若不等於 0 則回到步驟 2

在求解動態 VRP 問題的演算流程圖如圖 3.4 所示

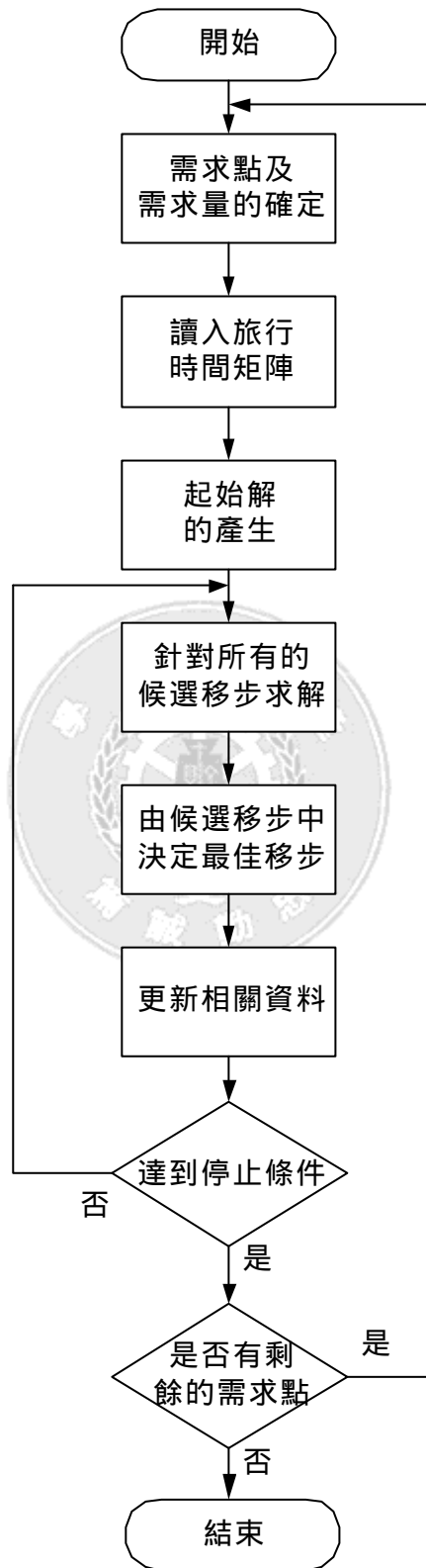


圖 3.4 求解動態 VRP 問題的演算流程

由於 DYNASMART 模擬指派模式，主要是利用交通模擬來建立相對應的交通型態，對於動態車輛路線問題而言，並無法計算得到一最佳的巡迴路線解，因此，本研究所撰寫之程式即為 DYNASMART 之副程式，以利於即時資訊下的交通模擬與問題求解的整合。在求解的演算過程中，首先決定需求點的數目及其需求量，並由模擬指派模式 DYNASMART 產生即時性的旅行時間矩陣，接著呼叫副程式，以進行最佳巡迴路線的求解。在求解的過程中，是以不同的方式產生起始解，透過 2-opt 的方式進行移步，並根據不同的需求點設定不同的最大遞迴數為收斂條件，在最佳巡迴路線產生後，將求得的巡迴需求點順序回傳給 DYNASMART，此時 DYNASMART 在接收到新的資訊後，開始計算交換新路徑，繼續模擬產生下一組的旅行時間矩陣，如此反覆執行，直到模擬結束，其演算流程如圖 3.5 所示。

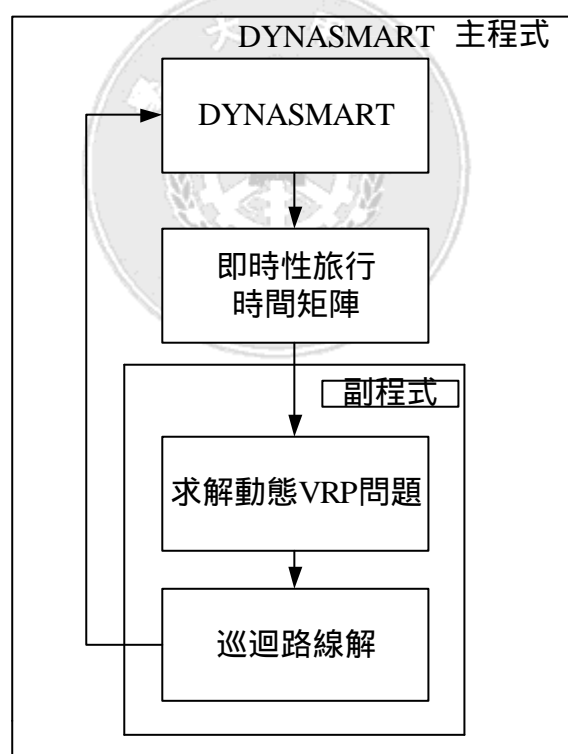


圖 3.5 DYNASMART 與運算程式之演算流程

第四章 禁制搜尋法於靜態問題之測試

本章將針對本研究所發展禁制搜尋法於靜態 TSP 問題之測試，以瞭解其參數設定與效率；第 4.1 節將簡單敘述實驗的主要目的，並介紹選取的測試例題及模式內參數的設定原則，第 4.2 節為測試結果與分析。

4.1 實驗設計

4.1.1 實驗目的

當應用啟發式解法求解 VRP 問題時，並不確定其結果是否為最佳的巡迴路線解，因此為瞭解本研究演算法之正確性及問題求解上的效率，研究中挑選了相關的國際例題進行測試與分析；而由於本研究主要為建立即時資訊下動態 VRP 之模擬評估架構，因此對於 Tabu Search 求解動態 VRP 問題時，相關參數的設定（如禁制名單、停止搜尋準則等），並沒有進行討論，其參數的設定主要參考相關文獻作為設計的準則。

進行相關例題測試主要目的如下：

1. 決定 Tabu Search 於求解即時資訊下動態 VRP 問題時，其參數的設定原則。
2. 在上述參數的設計原則下，驗證演算法在 VRP 問題求解上的有效性與正確性。
3. 根據問題求解的運算速度，瞭解演算法在求解即時資訊下動態 VRP 問題的可行性。

4.1.2 測試例題與參數設定

本研究所測試的例題主要是透過網路查詢公開的國際網路題庫 TSPLIB，TSPLIB 網站內包含了許多 TSP 相關測試例題，如對稱性及

非對稱性 TSP 問題、容量限制的 VRP 問題等，並定期由作者 Gerhard Reinelt 等人予以更新，目前最新的版本為 December 6, 2001，網址為 <http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/>；由於本研究對車輛路線問題的假設為單一車輛，且車輛無容量限制，因此亦即一旅行推銷員問題(Traveling Salesman Problem)，在 TSPLIB 共有 111 題 TSP 例題，本研究蒐集該題庫中需求點小於 100 的 14 個例題，其資料型態有路線成本矩陣及平面空間座標兩種，表 4.1 為這些例題的名稱、節點數、最小成本、及資料型態。

表 4.1 TSP 測試例題表

編號	例題名稱	節點數	最小成本	資料形態
1	bayg29	29	1610	路線成本矩陣
2	bays29	29	2020	路線成本矩陣
3	Berlin52	52	7542	平面空間座標
4	eil51	51	426	平面空間座標
5	eil76	76	538	平面空間座標
6	fir26	26	937	路線成本矩陣
7	gr17	17	2085	路線成本矩陣
8	gr21	21	2707	路線成本矩陣
9	gr24	24	1272	路線成本矩陣
10	kroA100	100	21282	平面空間座標
11	Pr76	76	108159	平面空間座標
12	Rd100	100	7910	平面空間座標
13	St70	70	675	平面空間座標
14	Swiss42	42	1273	路線成本矩陣

在禁制搜尋法中，本研究在移步上是採用 2-opt 交換法，而鄰搜的搜尋上則是選擇全鄰域搜尋法，當所有的鄰域解都被搜尋後，在滿足禁制名單與渴望準則的條件下，選取鄰域解中所搜尋的最佳解當做新的起始解，而該移步則納入禁制名單，以進行下一次的搜尋。

而在路線交換的過程中，當某一個移步已被列入禁制名單內時，即進行渴望準則的判斷，以決定該移步是否能被使用；本研究對於渴望準則的判斷原則如下：當該移步被列入禁制名單，但其移步所獲得的巡迴路線總成本小於目前所尋獲之最佳解時，便進行該移步的使用，並更新尋獲之最佳解及禁制名單。

在參數設定方面，主要探討的有起始解的建構、禁制名單的設計、及搜尋停止準則三部份，在起始解的建構上包含了插入法(Insert)、最近鄰點法(Nearest Neighbor)、及隨機路線法(Random Tour)三種；而禁制名單的設計、及搜尋停止準則這兩部份則是參考過去相關文獻以兩種不同的方式進行設計。

方法 1 是由 John Knox(1994)所提出，Knox 認為禁制名單的設計應為 3 倍的需求點數，最大遞迴為 0.0003 乘上需求點的四次方，在此設計準則之下，當需求點小於 100 時，可獲得較佳的解，並由 6 個測驗例題中發現，誤差皆在 5% 之內；而方法 2 則是參考吳泰熙等人(1997)的研究，吳泰熙等人認為禁制名單應為 2.375 倍的需求點數，當需求點小於 50，選取的遞迴數為 2000，需求點大於 50 時，選取的遞迴數為 4000，並由 8 個測驗例題中發現，平均誤差為 0.96%；以下將以 Knox 代表方法 1，方法 2 以 Wu 稱之；研究中的測試項目及範圍如表 4.2 所示。

表 4.2 測試項目及範圍

項目	測試範圍	
起始解	插入法(Insert)	
	最近鄰點法(Nearest Neighbor)	
	隨機路線法(Random Tour)	
禁制名單與遞迴數	Knox	遞迴數 = $0.003 * N^4$ 禁制名單 = $3N$
	Wu	禁制名單 = $2.375N$ $0 < N \leq 50$ 遞迴數 = 2000 $50 < N$ 遞迴數 = 4000

註：N 為需求點數

4.2 結果分析

本節主要依據 4.1 節所設計的參數組合，進行例題的測試，決定出最佳的設計參數。

測試的過程首先決定起始解的產生方式，再由相關文獻決定禁制名單及遞迴數的大小，所有的例題皆在 Compaq XP1000 UNIX 系統上以 Fortran 程式語言進行測試，其測試結果誤差百分比如表 4.3.所示。

表 4.3 測試結果表

編號	題目名稱	現有 最佳成本	插入法		最近鄰點法		隨機路線	
			誤差		誤差		誤差	
			方法 1	方法 2	方法 1	方法 2	方法 1	方法 2
1	bayg29	1610	6.15%	6.15%	3.04%	3.04%	4.41%	5.78%
2	bays29	2020	0.69%	0.69%	0.00%	0.00%	0.74%	1.93%
3	berlin52	7542	3.29%	3.29%	2.32%	2.32%	0.00%	6.68%
4	eil51	426	2.58%	2.58%	1.41%	1.41%	1.88%	2.82%
5	eil76	538	5.02%	5.02%	0.00%	0.00%	5.58%	1.49%
6	fir26	937	1.71%	1.71%	1.92%	1.92%	2.35%	0.00%
7	gr17	2085	0.00%	0.00%	0.00%	0.00%	0.00%	0.24%
8	gr21	2707	0.00%	0.00%	0.00%	0.00%	0.00%	3.47%
9	gr24	1272	0.00%	0.00%	3.54%	3.54%	0.00%	4.01%
10	kroA100	21282	0.29%	0.29%	2.21%	2.21%	4.94%	4.77%
11	pr76	108159	5.99%	5.99%	1.60%	1.60%	3.22%	5.98%
12	rd100	7910	9.38%	9.38%	1.02%	1.02%	4.01%	5.83%
13	st70	675	0.89%	0.89%	2.52%	2.52%	3.56%	5.04%
14	swiss42	1273	0.86%	0.86%	0.86%	0.86%	2.20%	3.30%
平均誤差			3.35%	3.35%	1.86%	1.86%	2.99%	4.67%

由上表可知當起始解選用最近鄰點法時，其平均誤差較小，而在禁制名單與遞迴數的設計上，兩種設計方式所得的結果並沒有差異，因此在起始解的選擇上將採用最近鄰點法；表 4.4 為起始解為最近鄰點法的情況下，Knox 與 Wu 在禁制名單與遞迴數上的設計準則與其運算時間，圖 4.1 是需求點數與運算時間關係圖。

表 4.4 最近鄰點法下不同設計準則與執行速度

編號	例題名稱	Knox			Wu		
		禁制名單	遞迴數	CPU 時間(sec)	禁制名單	遞迴數	CPU 時間(sec)
1	Bayg29	87	212	0.16sec	68	2000	1.53sec
2	bays29	87	212	0.13sec	68	2000	1.26sec
3	Berlin52	156	2193	3.5sec	123	4000	6.48sec
4	eil51	153	2029	3.16sec	121	4000	6.18sec
5	eil76	228	10008	36.75sec	180	4000	14.55sec
6	fir26	78	139	0.08sec	61	2000	1.13sec
7	gr17	51	25	0.01sec	40	2000	0.73sec
8	gr21	63	58	0.03sec	49	2000	0.93sec
9	gr24	72	99	0.06sec	57	2000	1.10sec
10	kroA100	300	30000	214.43sec	237	4000	28.73sec
11	pr76	228	10008	37.96sec	180	4000	14.86sec
12	rd100	300	30000	216.55sec	237	4000	28.86sec
13	st70	210	7203	22.26sec	166	4000	12.26sc
14	swiss42	126	933	1.06sec	99	2000	2.18sec
平均		38.32sec			8.63sec		

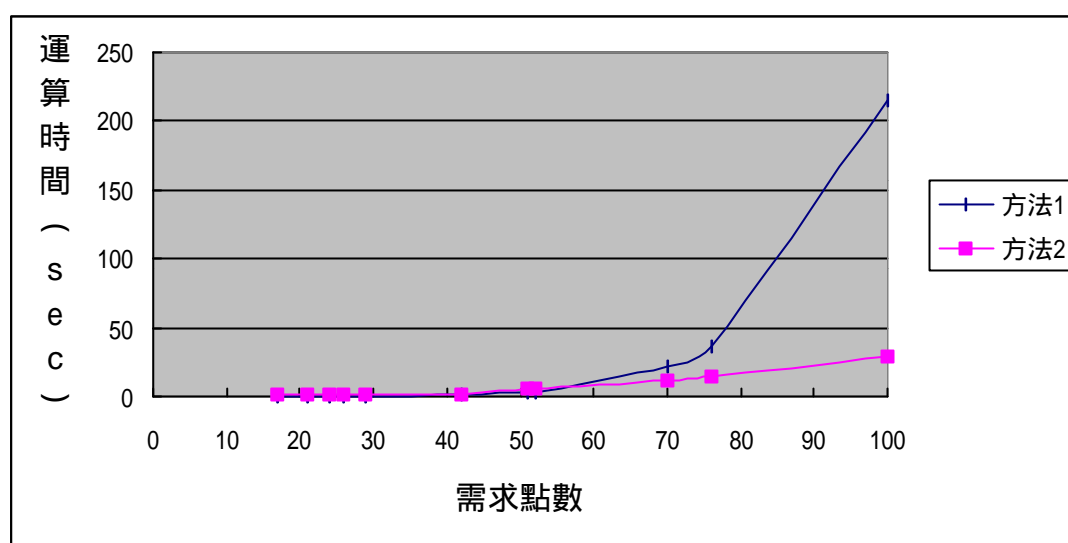


圖 4.1 需求點數與運算時間關係圖

由表 4.4 及圖 4.1 可知，當需求點約小於 55 時，Knox 的設計方式具有較佳的解題速率，但因其遞迴數的設計為 $0.0003 * N^4$ (N 為需求點數)，隨著需求點 N 的增加，運算時間會成幾何的成長，當需求點為 100 時，約需要 215 秒才能達到收斂條件，但若採用 W_u 的方式，僅需耗費 28 秒；由於本研究主要探討即時資訊下的 VRP 問題，為了在有限的時間內獲得巡迴路線解，因此本研究將採用 W_u 的設計方式做為後續實驗的準則。

4.3 小結

傳統啟發式解法，在進行鄰域的搜尋中，只有在目標值改善的情況下才進行路線的交換，因此容易陷入局部的區域解，禁制搜尋法(TS)突破了陷入局部區域解的缺點，因此在求解過程中，可獲得較佳的結果，研究所得之最佳解如表 4.5 所示，在 14 個例題中其平均誤差為 0.65%。

表 4.5 本研所得之最佳解

編號	例題名稱	最小成本	本研究 最小成本	誤差
1	bayg29	1610	1659	3.04%
2	bays29	2020	2020	0.00%
3	berlin52	7542	7542	0.00%
4	eil51	426	432	1.41%
5	eil76	538	538	0.00%
6	fir26	937	937	0.00%
7	gr17	2085	2085	0.00%
8	gr21	2707	2707	0.00%
9	gr24	1272	1272	0.00%
10	kroA100	21282	21343	0.29%
11	pr76	108159	109887	1.60%
12	rd100	7910	7991	1.02%
13	st70	675	681	0.89%
14	swiss42	1273	1284	0.86%
平均誤差				0.65%

由於本研究主要探討即時資訊下的動態 VRP 問題，因此對於 TS 中的參數設計上並沒有太多的討論，僅引用過去相關文獻進行比較；比較各例題的測試結果後，在考慮求解的正確性及計算的運算速度上後，本研究在起始解的建構上將選擇最近鄰點法，而參數設計上將採用 Wu 的方式，將禁制名單設為 2.375 倍的需求點數，當需求點小於 50，選取的遞迴數為 2000，需求點大於 50 時，選取的遞迴數為 4000，以做為後續求解即時資訊下動態 VRP 問題實驗時的準則。

第五章 即時資訊下動態 VRP 之實驗與分析

為驗證模式的適用性，本章將利用台中市路網資料，以數值實驗方式進行模式的測試，以瞭解本研究建構模式之特性及能力，並觀察在不同設計條件下，路網車流的變化情形。第 5.1 節將討論本研究模式在獲得不同屬性的交通資訊下，採用不同的求解演算流程；第 5.2 節將說明實驗中的基本假設、設計參數及設定準則；第 5.3 與 5.4 節為獲得不同屬性的交通資訊之實驗結果分析與討論；第 5.5 節將進行道路上有事故發生時的實驗與分析。

5.1 演算法設計

根據第二章回顧的文獻中可知，資訊的提供方式對動態問題有明顯的影響，因此本研究針對不同屬性的交通資訊設計不同的演算法以進行動態 VRP 的求解。其中根據交通資訊的屬性可分為即時性資訊與歷史性資訊兩類；第 5.1.1 節將討論在僅擁有即時性資訊下，其資料的特性，與求解的演算流程；第 5.1.2 節討論同時擁有即時性資訊與歷史性資訊，資料的應用及求解的過程。根據第四章的例題測試的結果，演算法中起始解的建構將採用最鄰近解法，而禁制名單與停止搜尋準則，則是以吳泰熙等人 (1997) 的文獻作為設計的準則。

5.1.1 考慮即時資訊下巡迴路線演算流程

在即時資訊下巡迴路線演算流程中，主要應用的交通資訊為即時性的路段資料，根據 Psaraftis(1995)對資訊的分類，其屬性說明如下：

1. 資訊的演進為動態屬性，因為在進行巡迴路線求解時，必須依據即時性的交通資訊為基準。

2. 資訊的品質為已知確定性，因為路段上的旅行時間是可藉由道路上的偵測器及車輛上的定位系統所獲得的。

在運算求解過程中，首先在獲得需求點資訊後，透過即時性路段旅行時間的獲得，以最近鄰點法進行巡迴路線起始解的研擬，並應用 2-opt 交換法產生所有的可行移步解，配合禁制名單與渴望準則的判斷，挑選最佳的移步解，此處的禁制名單在設計上是根據第四章的測試結果，以 2.375 倍的需求點數，做為禁制名單的禁制期間；在渴望準則的設計上是以某一移步所獲得的巡迴路線總成本，是否小於目前所尋獲之最佳解為主，當移步位於禁制名單內，但其移步後的巡迴路線總成本小於目前所尋獲之最佳解時，便透過渴望準則將該移步於禁制名單內釋放出來，進行該移步；在挑選完最佳的移步解後，並更新相關的資料，如：禁制名單、起始解等；並判斷是否達到搜尋停止的條件，以決定路線的轉換與否，如此反覆進行，直到所有的需求點皆被服務為止。

在應用禁制搜尋法求解 VRP 問題的過程裡，只須求解在接受即時資訊時，該時間點下的最佳的巡迴路線解，而此時的禁制名單稱之為空間上的禁制名單；演算流程如圖 5.1 所示。

其演算步驟如下：

- 步驟 1. 巡迴路線上需求點及需求量的確定
- 步驟 2. 透過即時性路段旅行時間的取得，更新旅行時間矩陣
- 步驟 3. 產生巡迴路線起始解
- 步驟 4. 利用 2-opt 交換法產生所有可行的巡迴路線解（即候選名單）
- 步驟 5. 在所有可行解（候選名單）中選擇最佳的巡迴路線解
- 步驟 6. 當步驟 5 所選擇的巡迴路線內包含被禁制的路段，考慮是否符合渴望準則（即該移步所獲得的巡迴路線總成本是否小於目前所尋獲之最佳解），若是，到步驟 7；若否，則不考慮該條路線，回到步驟 5

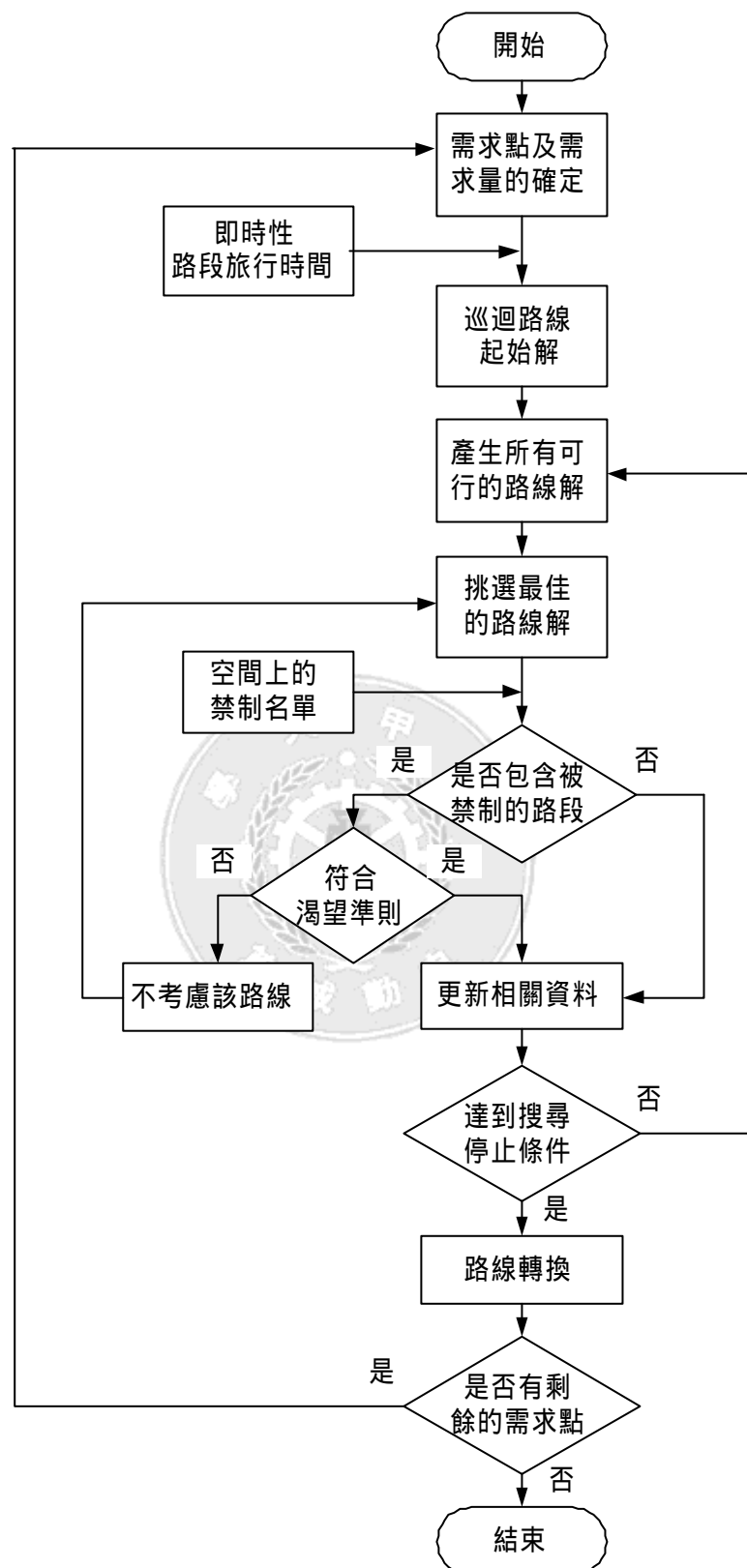


圖 5.1 考慮即時資訊下巡迴路線演算流程

步驟 7.選擇該移步，並更新相關資訊，如：起始解、現有尋獲解、禁制名單等

步驟 8.是否滿足停止準則，是，則停止，否，則回到步驟 4

步驟 9.進行巡迴路線的轉換

步驟 10.是否仍有剩餘的需求點，是，則回到步驟 2；否，則結束

5.1.2 結合即時資訊與歷史資訊下巡迴路線演算流程

根據胡大瀛與洪百賢（2002）在”公車動態到站時間預測成果報告”中指出，若蒐集每日的公車資訊，並將公車行駛狀態繪製成時空圖後，如圖 5.2 所示，可發現在公車行經的路線上，在無特殊交通事故下，其車流為一穩定的狀態，可知公車在不同日期同一時段內行駛於相鄰兩站間的時間可透過歷史資料，配合即時性資訊進行到站時間的預測。

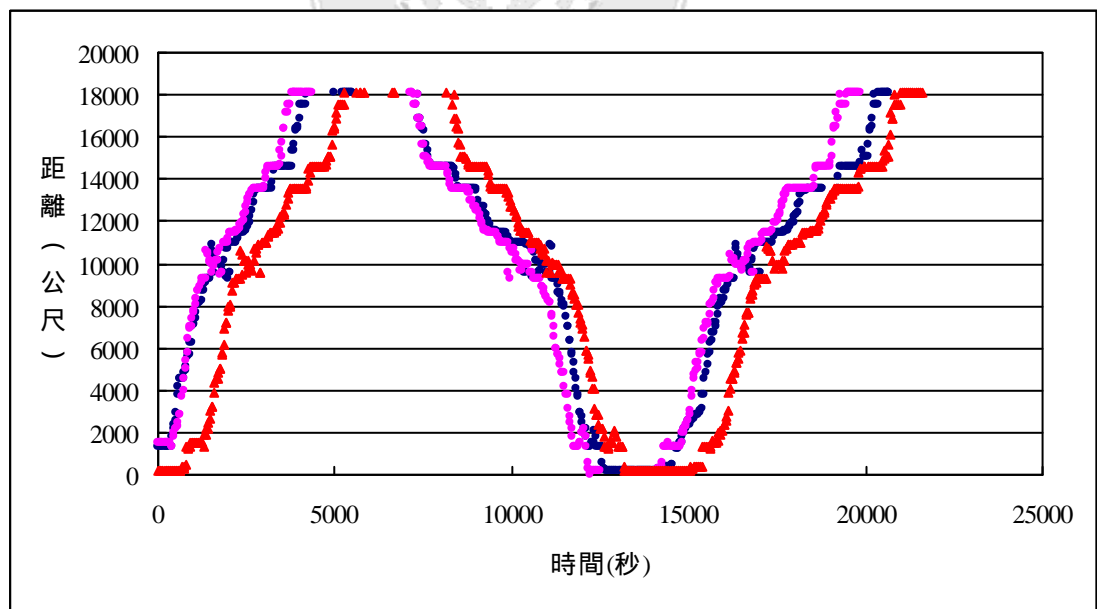


圖 5.2 不同公車行駛同路線之時空圖

資料來源：胡大瀛、洪百賢（2002）

因此，在同時接受即時資訊與歷史資訊下，根據 Psaraftis(1995)對資訊的分類，其屬性說明如下：

1. 資訊的演進為包含了動態與靜態屬性，在進行巡迴路線求解時，除了依據即時性的交通資訊，更必須配合歷史資料進行路線的研擬。
2. 資訊的品質為已知確定性及預測性；路段上的旅行時間除了藉由道路上的偵測器及車輛上的定位系統所獲得外，更透過歷史性資料進行路段旅行時間的預測。

研究中的歷史性資訊是以 DYNASMART 進行完整的交通模擬後，搜集各時間點的資料作為歷史性背景資料的依據，而在實務的應用上，這些資料可透過車輛偵測器或偵測器與 GPS 車輛蒐集每日各時段的流量分佈與旅行時間所獲得；而進行擁擠路段預測時，其預測期間主要配合即時資訊提供的頻率，即當即時資訊提供的頻率為每 5 分鐘時，在該時間點上，則預測未來 5 分鐘內，路網中將產生擁擠的路段；當其頻率為每 10 分鐘或 15 分鐘時，則進行未來 10 分鐘或 15 分鐘內擁擠路段的預測。

當某條路段被預測為擁擠路段時，將被列入時間上的禁制名單中，限制該條路段於某特定時間點上被使用的機會，其名單主要根據歷史性背景資料與即時性資料相比較後產生；在應用禁制搜尋法求解 VRP 問題的過程中，仍需透過 DYNASMART 進行交通模擬，以獲得即時性路段旅行時間，但在進行巡迴路線的研擬，即決定路線的轉換與否的同時，必須透過歷史資訊，同時考慮路段上可能產生擁擠的情況，因此，所求得之解不一定為該時間點的最佳解。

在進行即時性資訊與歷史性資訊比較的過程中，本研究透過以下幾種方式反應未來的路段狀況，其中 t_{ij}^t 代表現有時間點 t 節點 i 到節點 j 的路段旅行時間，因此現有時間點 n 下，節點 i 到節點 j 的路段旅行時間以 t_{ij}^n 表示，而 t_{ij}^f 則表示預測時間點 f 下，節點 i 到節點 j 的路段旅行時間。

1. 旅行時間變化百分比：主要為計算路段上旅行時間變動的程度，當其數值為正時，代表未來該路段的旅行時間將增長，反之，則代表旅行時間將減少，其計算方式如下：

$$\text{旅行時間變化百分比} = \frac{t_{ij}^f - t_{ij}^n}{t_{ij}^n} \times 100\%$$

2. 旅行時間的差值：主要為計算行經某路段可能增加(或減少)的旅行時間成本，當其數值為正時，代表未來該路段的旅行時間將增加，反之，則代表旅行時間將減少，其計算方式如下：

$$\text{旅行時間的差值} = t_{ij}^f - t_{ij}^n$$

3. 旅行時間平均值：主要為獲得未來行經該路段時的期望旅行時間成本，其計算方式為如下：

$$\text{旅行時間平均值} = \frac{\sum_{a=n}^f t_{ij}^a}{f - n}$$

4. 期望旅行時間的差值：計算行經某路段期望增加的旅行時間成本，主要為旅行時間平均值減去現有時間點的路段旅行時間，其公式如下：

$$\text{期望旅行時間的差值} = \frac{\sum_{a=n}^f t_{ij}^a}{f - n} - t_{ij}^n$$

根據本研究的測試發現在獲得旅行時間增加百分比與旅行時間平均值兩種資訊下，對禁制名單的設計上較難控制與掌握，其原因如下：

1. 旅行時間變化百分比：雖然百分比可以反應路段旅行時間的變化量，但就百分比的變化並無法得知旅行時間的差異性，因此較難以進行禁制名單的設計；如路段旅行時間為 1 分鐘，增加百分比為 100%時，其預測的旅行時間差為 1 分鐘；而路段旅行時間為 10 分鐘下，百分比為 50%時，其預測的旅行時間差則為 5 分鐘，後者較前者多了 4 分鐘。
2. 旅行時間平均值：路段旅行時間平均值無法適時反應路段旅行

時間的變化；如旅行時間由 2 分鐘增加至 4 分鐘，與 4 分鐘降至 2 分鐘，其平均值皆相同。

因此，研究中透過實驗測試，藉由旅行時間的差值與期望旅行時間的差值，將預測擁擠路段納入時間上的禁制名單，其設計的基本原則說明如下：

1. 旅行時間的差值：當旅行時間的差值設定過高時，路網中被禁制的路段數較少，因此反應不出路段被禁制所帶來的效用；而當差值選擇過小時，會使路網內過多的路段被禁制，容易解出一條極差的巡迴路線解；透過實驗的分析，當即時資訊提供的頻率為每 5 分鐘及每 10 分鐘的情況下，為其使用的最佳時機，當差值為 1.5 分鐘以上者，建議將該條路段納入時間上的禁制名單，。
2. 期望旅行時間的差值：當預測期間過長時，預測時間點的旅行時間可能已經過了擁擠的時段，而回復到原始的旅行時間，此時若以旅行時間的差值作為基準，便無法顯示路段旅行時間於擁擠情況下的變動情形，因此，研究中建議採用期望旅行時間的差值作為禁制的準則，當其差值 0.75 分鐘以上者，納入時間上的禁制名單，本法在即時資訊提供的頻率為每 10 分鐘及每 15 分鐘的情況下，較能顯示其禁制的效用。

但當巡迴路線為了不行經所預測的擁擠路段，反而使得整體的旅行時間明顯的增加，此時便不將該條路段納入禁制名單內，而該條路段稱為巡迴路線之絕對路段。

本研究在求解結合即時資訊與歷史資訊下巡迴路線的演算流程如圖 5.3 所示，首先在獲得需求點資訊後，透過即時性路段旅行時間的獲得，以最近鄰點法進行巡迴路線起始解的研擬，並應用 2-opt 交換法產生所有的可行移步解，配合禁制名單與渴望準則的判斷，挑選最佳的移步解，此處的禁制名單分為兩種，分別為空間與時間上的禁制名單，空間上的禁制名單設計方式及渴望準則的設計同 5.1.1 節所示，而時間上禁制名單，則是透過歷史性的路段旅行時間進行擁擠路

段的預測，但若該路段為巡迴路線上的必經路段，則將其於時間上禁制名單中釋放出來，以進行該移步；在挑選完最佳的移步解後，並更新相關的資料，如：禁制名單、起始解等；並判斷是否達到搜尋停止的條件，以決定路線的轉換與否，如此反覆進行，直到所有的需求點皆被服務為止；其演算流程如圖 5.3 所示

其演算步驟如下：

- 步驟 1.巡迴路線上需求點及需求量的確定
- 步驟 2.透過即時性路段旅行時間的獲得，更新旅行時間矩陣
- 步驟 3.產生巡迴路線起始解
- 步驟 4.利用 2-opt 交換法產生所有可行的巡迴路線解（即候選名單）
- 步驟 5.透過歷史性資訊進行擁擠路段的預測，並進行時間上禁制名單的設計
- 步驟 6.在所有的可行解（候選名單）中選擇最佳的巡迴路線解
- 步驟 7.當步驟 5 所選擇的路線內包含被禁制的路段，
 - 步驟 7.1 是否為巡迴路線必經路段，是，到步驟 7.2；否，則不考慮該條路線，並回到步驟 6
 - 步驟 7.2 是否符合渴望準則（即該移步所獲得的巡迴路線總成本是否小於目前所尋獲之最佳解），是，到步驟 8；否，則不考慮該條路線，並回到步驟 6
- 步驟 8.選擇該移步更新相關資訊，如：起始解、現有尋獲解、禁制名單等
- 步驟 9.是否滿足停止準則，是，到步驟 9，否，則回到步驟 4
- 步驟 10.進行巡迴路線的轉換
- 步驟 11.是否仍有剩餘的需求點，是，則回到步驟 2；否，則結束

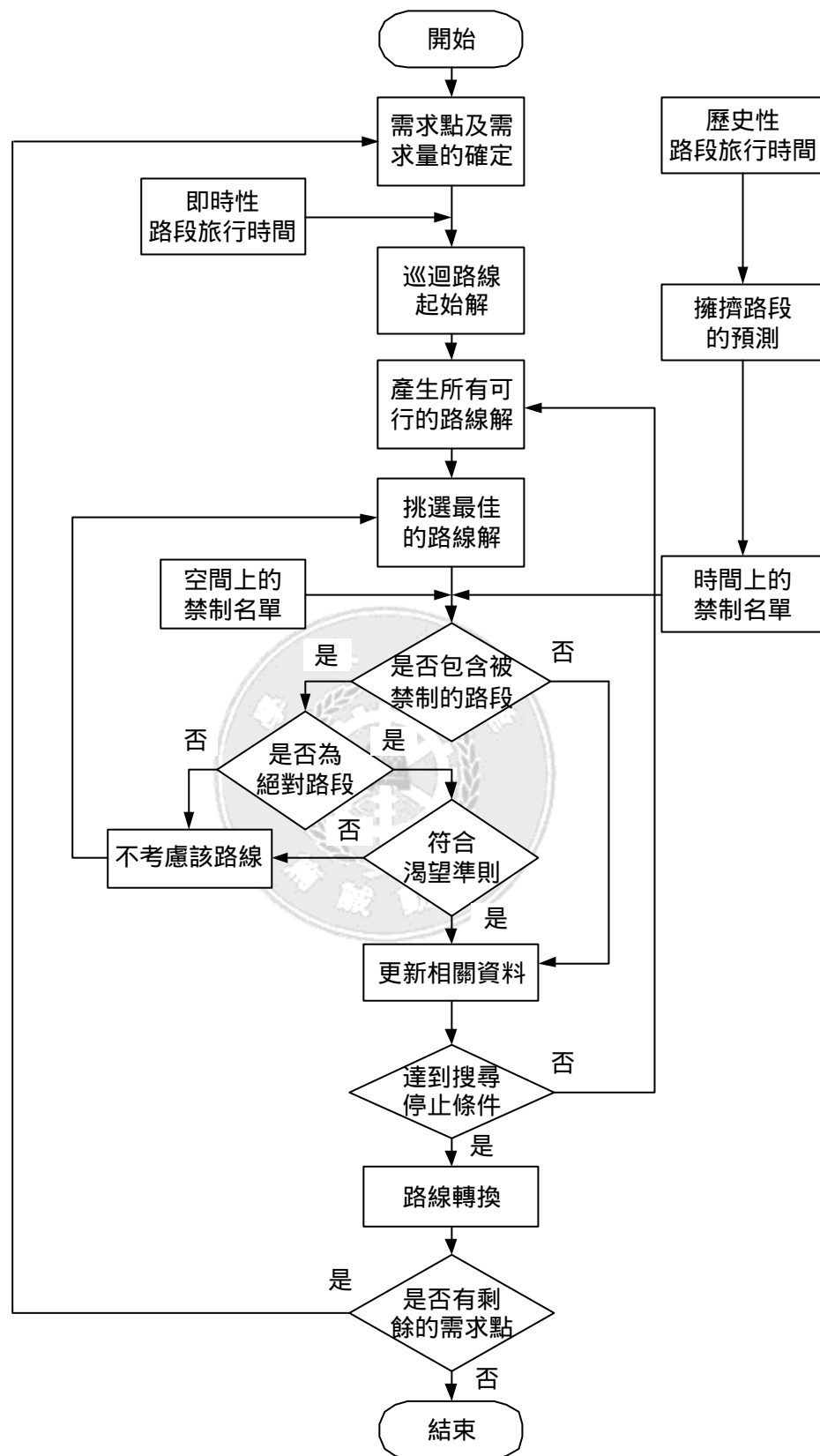


圖 5.3 接受即時性與歷史性資訊下巡迴路線演算流程

5.2 實驗設計

5.2.1 台中市路網資料

目前台中市的道路系統如圖 5.4 所示，主要是由輻射狀連外幹道如中港路、大雅路、北屯路、復興路、國光路加上環狀道路如文心路、進化路、忠明南路結合而成。市中心道路多為雙車道配置，少數為四車道以上之單行道。外環聯絡道路如文心路、忠明路，多為四車道以上的 30 米道路，並有其他市區服務道路相連接提供服務，整個路網相當密集。

在研究中所使用的台中市路網計有 87 個交通分區、574 個節點、1894 條節線；路網資料包含節點編號、節點 XY 座標、節點間距離、及車道數。

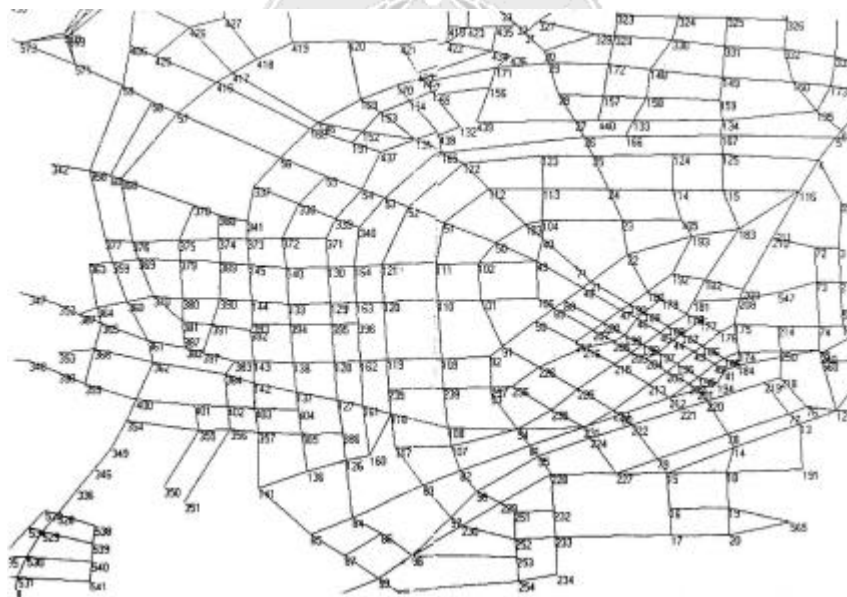


圖 5.4 台中市路網圖

5.2.2 參數設定

本實驗的主要目的為探討貨運業者在接受即時資訊後，重新規劃巡迴路線所帶來的路線效益，並探討於不同的車輛的產生型態及不同

需求點，即時資訊提供的最佳時機；實驗中，商用車輛服務的分區內計有 131 個需求點，實驗的平台為 Compaq XP1000 UNIX 系統。

研究中主要的假設與設計參數及設定準則如下所示

基本假設：

1. 單一車輛，且車輛無容量限制。
2. 駕駛者在接受即時資訊後，必須遵照規劃的巡迴路線行駛。
3. 車輛在接受即時資訊後必須抵達下一個需求點後，才可進行路線的轉換，不允許路途中轉換路線的情形發生。

而實驗中的設計參數及其準則如下：

1. 需求點個數：分為 10 個及 20 個需求點，其位置由電腦以亂數產生，產生完畢後即為一固定且已知的數值。
2. 即時資訊提供的頻率：主要分為 3 種類型，分別為每 5 分鐘、每 10 分鐘、及每 15 分鐘提供一次即時資訊，主要為討論在不同頻率下，提供即時資訊所帶來的效益。
3. 車輛產生型態：實驗中的車輛產生型態有 2 種情境，在 2 種不同的情境下，根據依時性的 OD 旅次，於模擬時間 30 分鐘內，產生約 27500 部車輛；情境 1 假設車輛的產生是均一的，主要為探討交通路網中產生一穩定的車輛數，道路上較無擁擠產生的情境；情境 2 則考慮尖峰的情況，於尖峰時間內同時湧入過多的車輛數，以反應出道路受車輛數增加時所產生擁擠情況。

表 5.1 為進行實驗時設計的參數項目、測試範圍及代號

表 5.1 交通模擬時參數項目、測試範圍及代號

參數項目	測試範圍	代號
需求點數	10	N10
	20	N20
即時資訊提供的頻率	不提供	RI0
	5 分鐘	RI5
	10 分鐘	RI10
	15 分鐘	RI15

5.3 考慮即時資訊下動態 VRP 之結果分析

本節將討論 2 種不同的需求點情況，2 種不同的車輛產生型態，與 4 種即時資訊的提供頻率下，巡迴路線所帶來的效益。

5.3.1 情境 1

在情境 1 中假設車輛的產生型態是均一的，如圖 5.5 所示，主要為探討當交通路網為一般無擁擠情況下，即時資訊的提供對巡迴路線解所帶來的效益；實驗中於模擬時間 30 分鐘內，根據依時性的 OD 旅次，產生 27542 輛車，其平均旅行時間為 14.09 分鐘，平均停等時間為 4.34 分鐘，平均旅行距離為 6.69 公里。

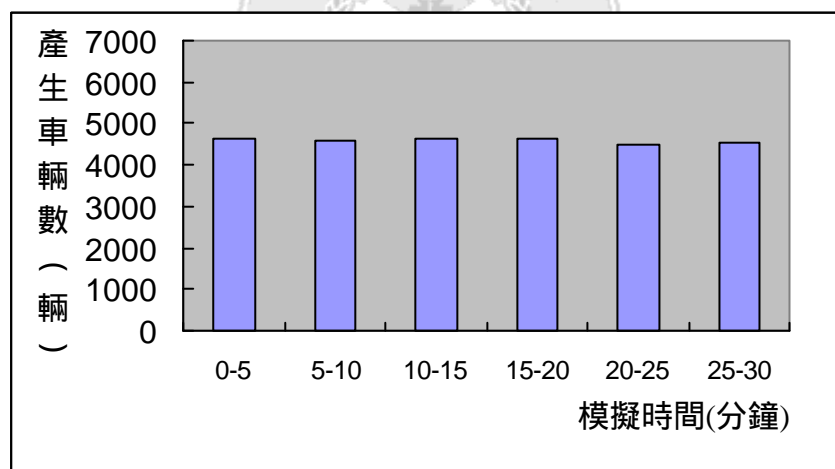


圖 5.5 情境 1 的車輛產生型態

1. 情境 1-10 個需求點

圖 5.6 為 10 個需求點在不同即時資訊下產生的巡迴路線圖，表 5.2 為其行經路段數、旅行時間及巡迴路線改善百分比。由圖中可知在不同頻率的即時資訊下，巡迴路線並沒有太大的差異，而在旅行時間方面，皆為 47.62 分鐘。

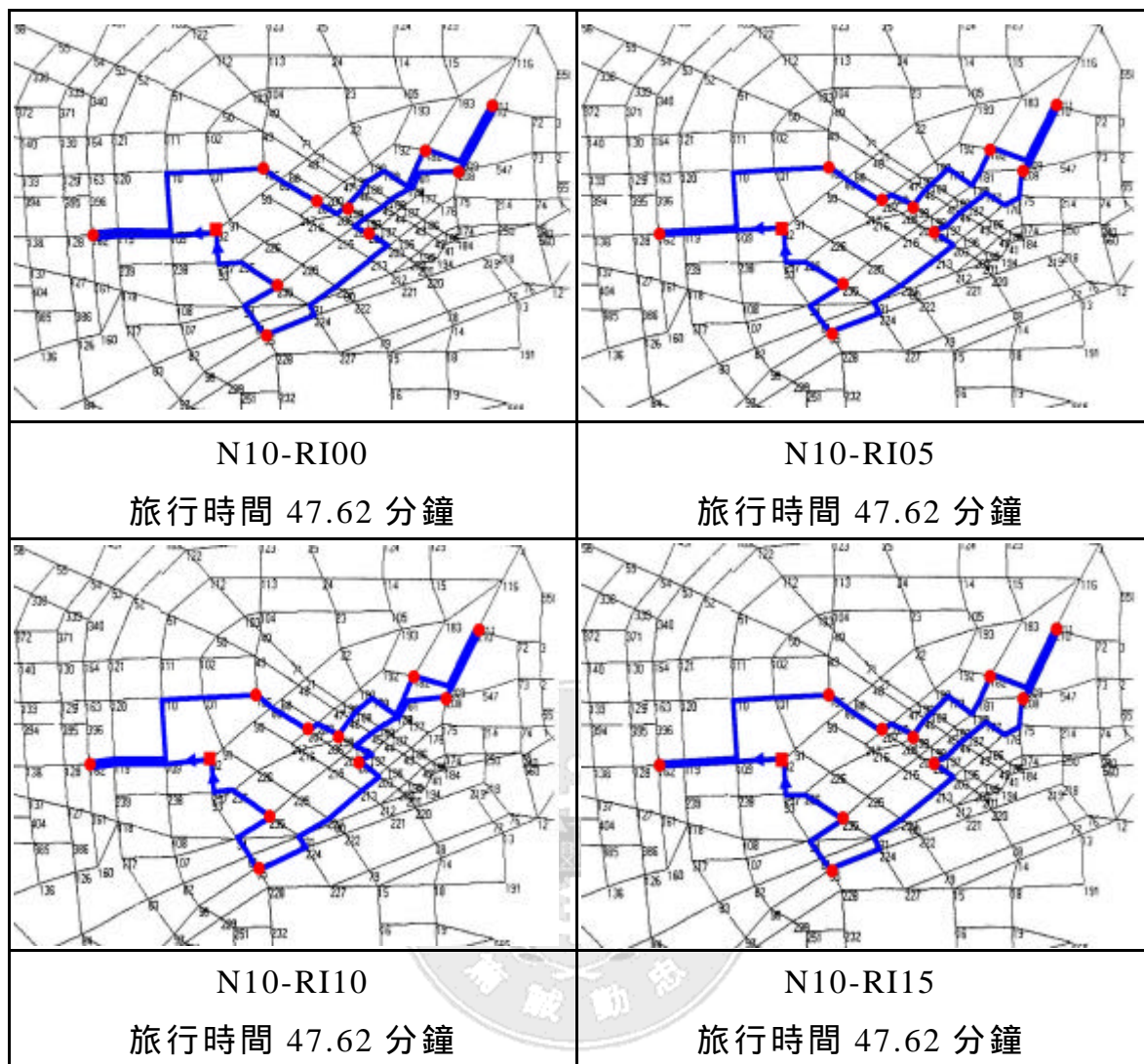


圖 5.6 情境 1-10 個需求點之巡迴路線比較圖

表 5.2 巡迴路線路段數、旅行時間及改善百分比

	路段數	旅行時間(分鐘)	改善百分比
N10-RI00	42	47.62	-
N10-RI05	42	47.62	0.00%
N10-RI10	42	47.62	0.00%
N10-RI15	42	47.62	0.00%

2.情境 1-20 個需求點

在 N20-RI10 與 N20-RI15 兩種情況下，由於即時資訊提供，使得巡視路線陷入較差的路線解，其主要原因是在求解的過程中並未考慮未來路段旅行時間的變化，因此提供了一條較差的路線解，並增加了 1.03% 的旅行時間。圖 5.7 為需求點 15 在不同即時資訊下產生的巡迴路線圖，表 5.3 為其行經路段數、旅行時間及巡迴路線改善百分比。

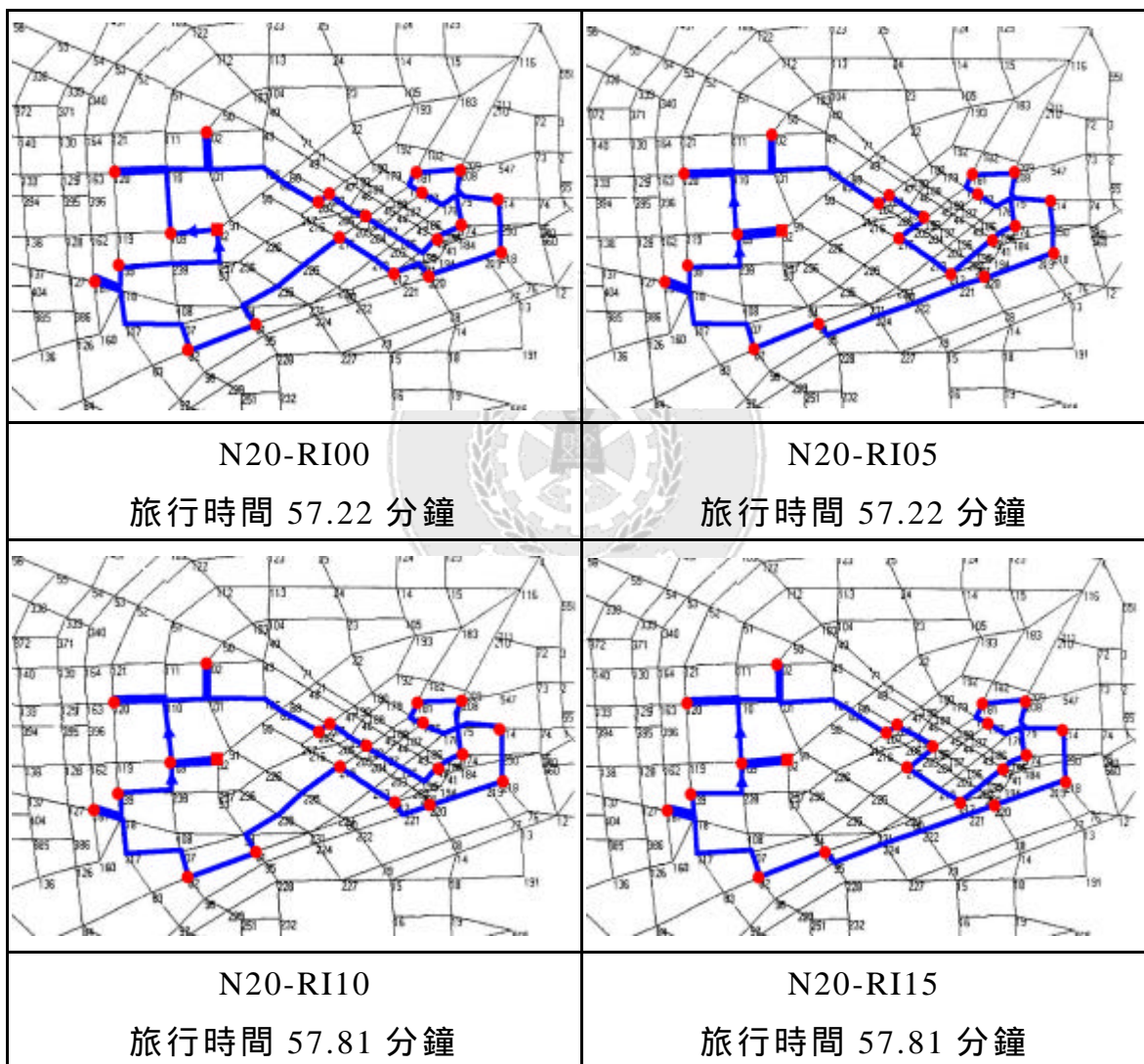


圖 5.7 情境 1-20 個需求點之巡迴路線比較圖

表 5.3 巡迴路線路段數、旅行時間及改善百分比

	路段數	旅行時間(分鐘)	改善百分比
N20-RI00	50	57.22	-
N20-RI05	50	57.22	0.00%
N20-RI10	50	57.81	-1.03%
N20-RI15	50	57.81	-1.03%

5.3.2 情境 2

情境 2 的車輛產生型態考慮了交通的變化及其尖峰性，如圖 5.8 所示，主要為探討當交通路網為產生明顯的擁擠情況下，即時資訊的提供對巡迴路線解所帶來的效益；實驗中於模擬時間 30 分鐘內，根據依時性的 OD 旅次，產生 27490 輛車，其平均旅行時間為 15.72 分鐘，平均停等時間為 5.53 分鐘，平均旅行距離為 6.71 公里。

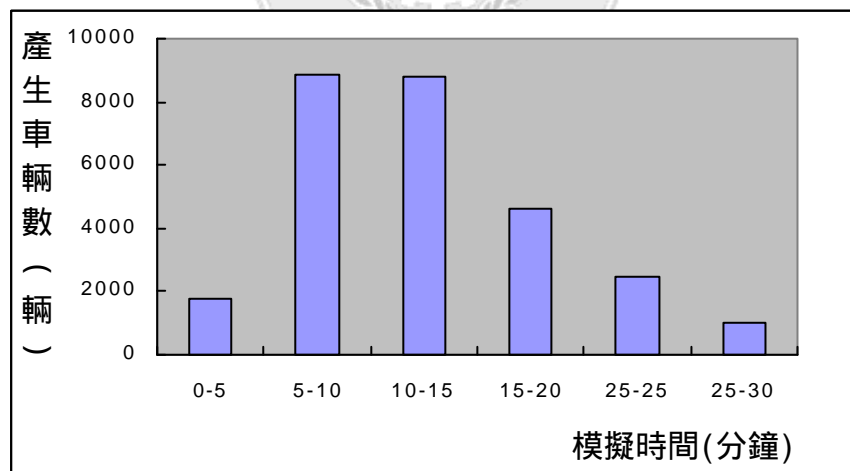


圖 5.8 情境 2 的車輛產生型態

1. 情境 2-10 個需求點

由圖 5.9 與表 5.4 可知，車輛在接受即時資訊後，巡迴路線明顯的改變，雖然其行經的路段數增加，但路線解皆獲得改善，其中改善幅度最大的為 N10-RI15，旅行時間由未提供即時資訊的 48.81 分鐘，降至 44.82 分鐘，改善百分比達 8.17%。

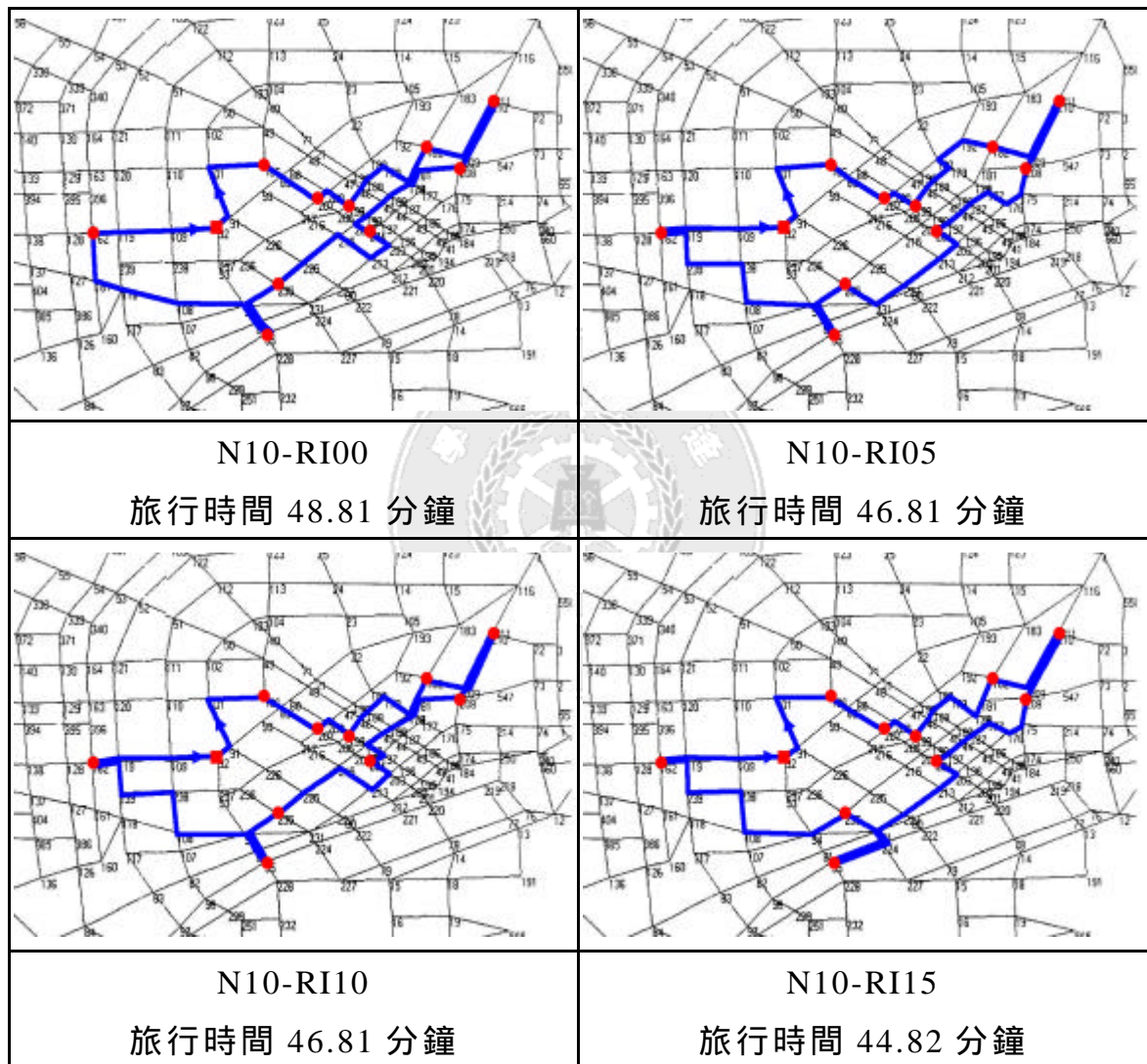


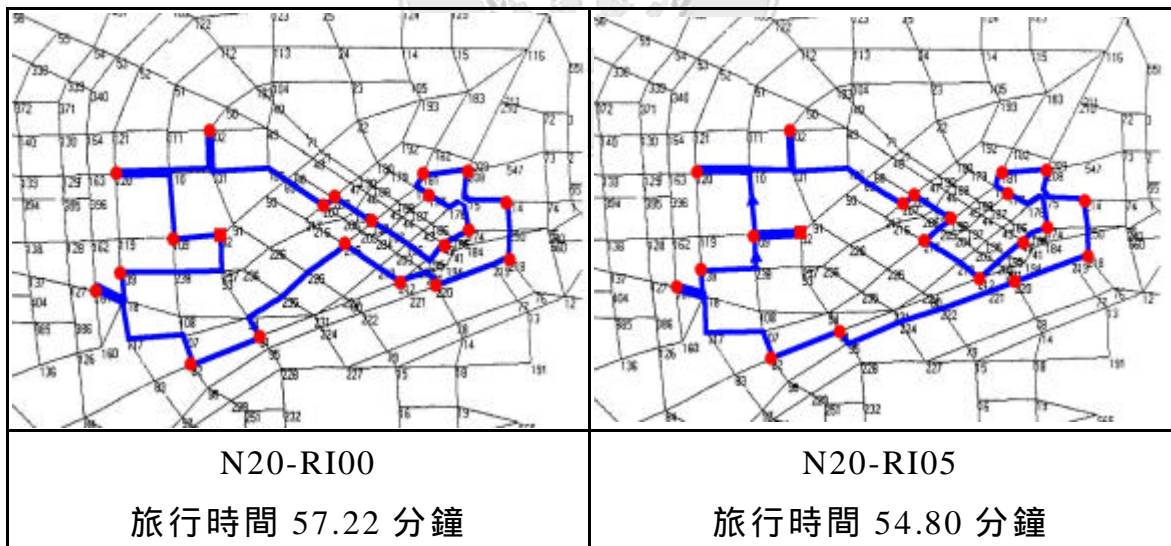
圖 5.9 情境 2-10 個需求點之巡迴路線比較圖

表 5.4 巡迴路線路段數、旅行時間及改善百分比

	路段數	旅行時間(分鐘)	改善百分比
N10-RI00	42	48.81	-
N10-RI05	43	46.81	4.10%
N10-RI10	43	46.81	4.10%
N10-RI15	43	44.82	8.17%

2. 情境 2-需求點 20

由圖 5.10 與表 5.5 可知，在 N20-RI05 及 N20-RI15 兩種情況下，轉換的路線降低了 4.23%與 4.54%的旅行時間，但在 N20-RI10 的情況中，轉換的路線卻陷入較差的解，旅行時間增加了 2.25%。圖 5.7 為需求點 20 之巡迴路線比較圖，表 5.5 為其行經路段數、旅行時間及巡迴路線改善百分比。



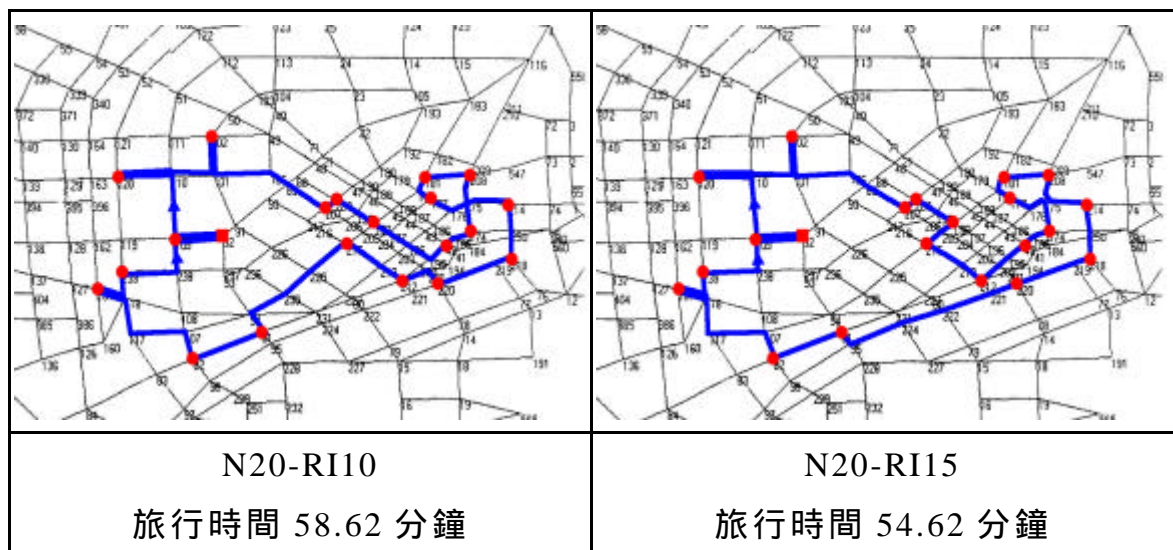


圖 5.10 情境 2-20 個需求點之巡迴路線比較圖

表 5.5 巡迴路線路段數、旅行時間及改善百分比

	路段數	旅行時間(分鐘)	改善百分比
N20-RI00	49	57.22	-
N20-RI05	49	54.80	4.23%
N20-RI10	49	58.62	-2.25%
N20-RI15	49	54.62	4.54%

5.4 結合即時資訊與歷史資訊下動態 VRP 之結果分析

由於 5.3 節中，情境 1 在道路上產生擁擠的情形並不明顯，因此在結合即時資訊與歷史資訊下動態 VRP 的實驗中，其車輛產生型態將選用情境 2；而由於 20 個需求點其對應的路段數較多，在進行擁擠路段的預測與選擇上較具彈性；因此，本節將在情境 2，20 個需求點的情況下，討論結合即時性資訊與歷史資訊，於不同即時資訊提供頻率下所帶來的路線效益，其中為了與 5.3 節的結果進行比較，在即時資訊的提供上將選擇每 5 分鐘、10 分鐘、15 分鐘做為其頻率。

1. 每 5 分鐘提供一次即時資訊：

由圖 5.11 及表 5.6 可知，雖然巡迴路線經過微幅的調整，但兩者在巡迴路線路段數、巡迴路線的停等時間、及旅行時間上並無差異，顯示在考慮即時性資訊下產生的巡迴路線，雖然避開了可能產生擁擠的路段，但其選擇的替代路段並沒有帶來明顯的效果；因此，雖然結合了歷史資訊進行擁擠路段的預測與路線的求解，巡迴路線無法再獲得改善。

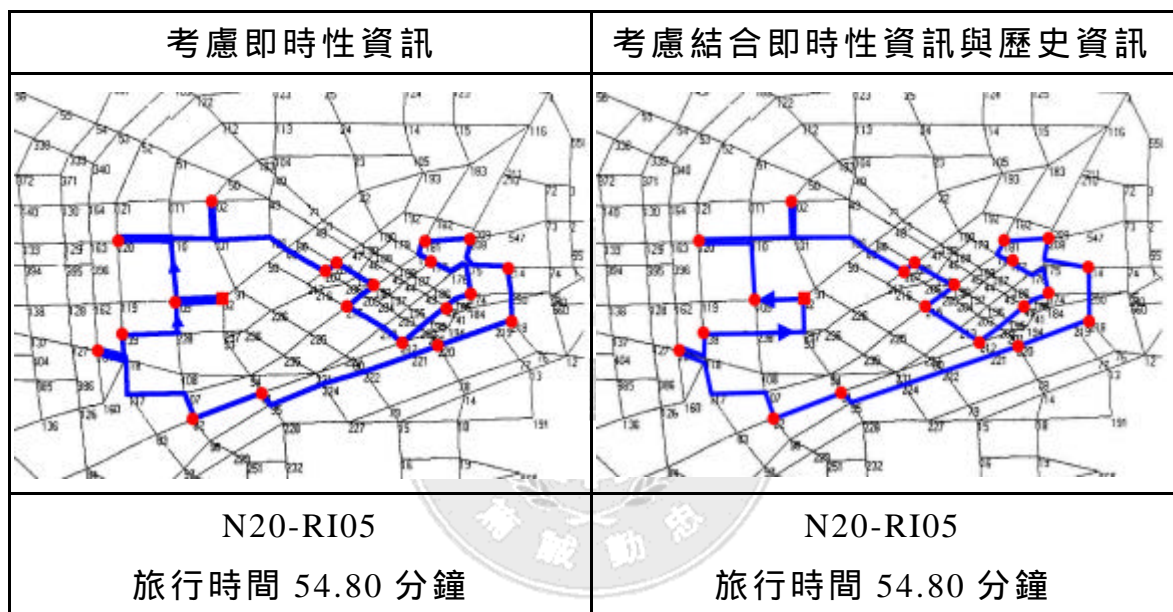


圖 5.11 考慮不同資訊種類下巡迴路線比較圖

表 5.6 考慮不同資訊種類下巡迴路線路段數、路線停等時間、旅行時間及改善百分比

資訊種類	路段數	路線停等時間(分鐘)	旅行時間(分鐘)	旅行時間改善百分比
即時資訊	50	11.61	54.80	-
即時資訊 與歷史資訊	50	11.61	54.80	0%

2. 每 10 分鐘提供一次即時資訊：

由圖 5.12 可知，整體的巡迴路線明顯的改變，而透過表 5.7 可知巡迴路線的停等時間下降了 4.01 分鐘，而旅行時間則由 58.62 分鐘降至 54.62 分鐘，共改善了 6.82%，顯示在結合即時資訊與歷史資訊下，提供的巡迴路段確實能有效避開擁擠的路段，降低路線上的停等時間及巡迴時間。

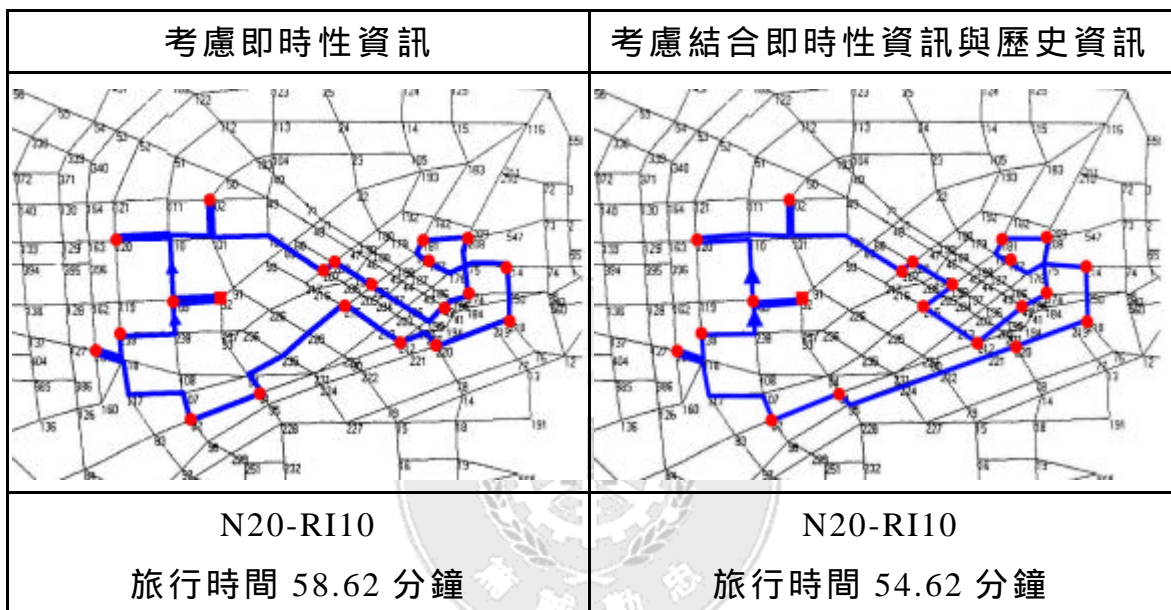


圖 5.12 考慮不同資訊種類下巡迴路線比較圖

表 5.7 考慮不同資訊種類下巡迴路線路段數、路線停等時間、旅行時間及改善百分比

資訊種類	路段數	路線停等時間(分鐘)	旅行時間(分鐘)	旅行時間改善百分比
即時資訊	50	15.30	58.62	-
即時資訊 與歷史資訊	50	11.29	54.62	6.82%

3.每 15 分鐘提供一次即時資訊：

由圖 5.13 可知，巡迴路線僅微幅的調整，根據表 5.8 可知，雖然整體的旅行時間並沒有獲得改善，但由巡迴路線停等時間的下降可瞭解，巡迴路線解確實避開了某些擁擠的路段。

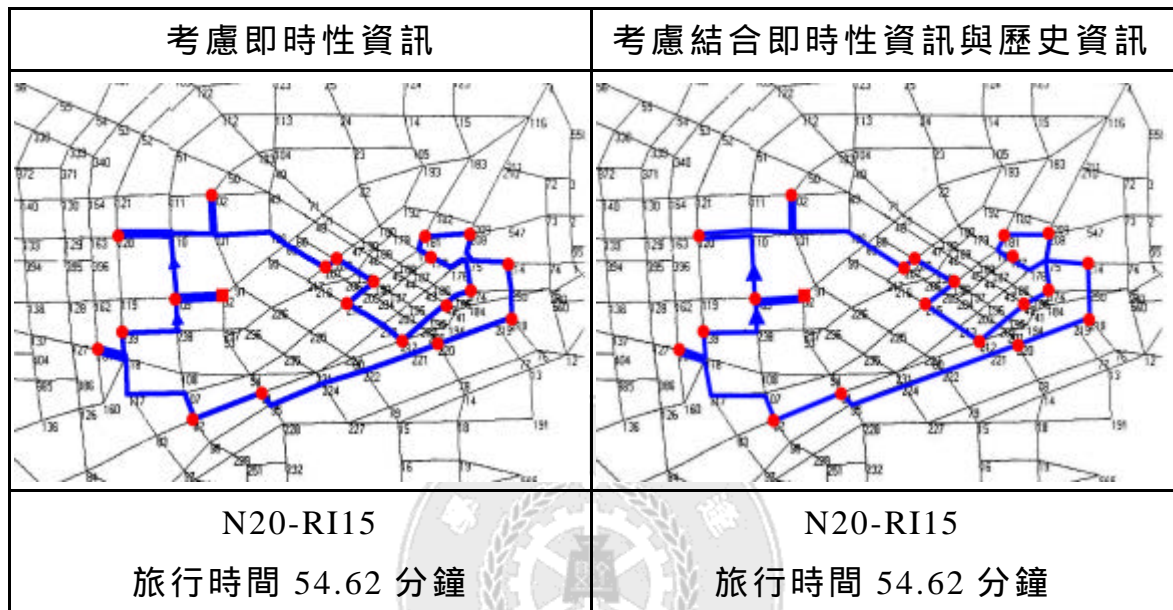


圖 5.13 不同資訊種類下巡迴路線比較圖

表 5.8 不同資訊種類下巡迴路線路段數、路線停等時間、旅行時間及改善百分比

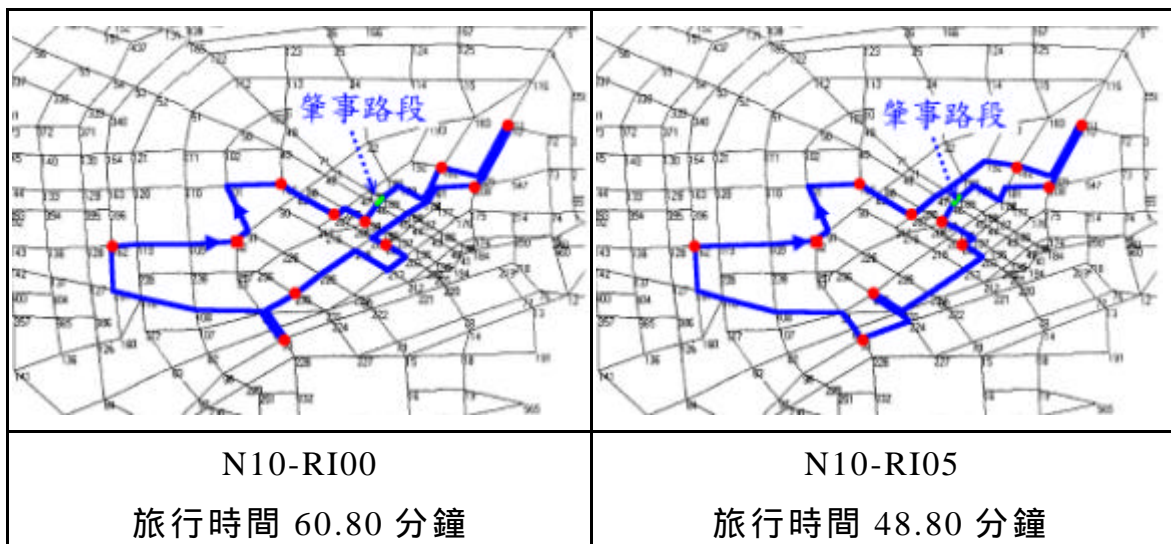
資訊種類	路段數	路線停等時間(分鐘)	旅行時間(分鐘)	旅行時間改善百分比
即時資訊	50	11.42	54.62	-
即時資訊 與歷史資訊	50	11.29	54.62	0%

5.5 考慮事故發生下動態 VRP 之結果分析

本節將討論在事故發生的情形下，商用車量因獲得即時資訊進行路線轉換下的路線效益，由於 5.3 節中，情境 1 在道路上產生擁擠的情形並不明顯，因此較不容易突顯事故發生時對交通影響，在考慮事故下動態 VRP 的實驗中，其車輛產生型態將選用情境 2，而在需求點的選擇上將考慮 10 個及 20 個需求點。

1. 需求點個數 10

由圖 5.14 及表 5.9 可知，在 N10-RI00 的情況下，由於車輛在未獲得即時資訊的情況下行經肇事路段，其總旅行時間為 60.80 分鐘，路線上的停等時間為 20.38 分鐘；在 N10-RI05 的情況下，車輛在接受即時資訊後，避開了肇事的路段，總旅行時間降為 48.80 分鐘，改善百分比達 19.74%，而停等時間則降為 12.57 分鐘；在 N10-RI10 及 N10-RI15 的情況下，雖然車輛有接受即時資訊，但由於車輛在接受即時資訊之前已到達肇事路段，因此並無法即時避開肇事路段，其總旅行時間為 56.80 分鐘，停等時間分別為 16.21 和 16.28 分鐘。



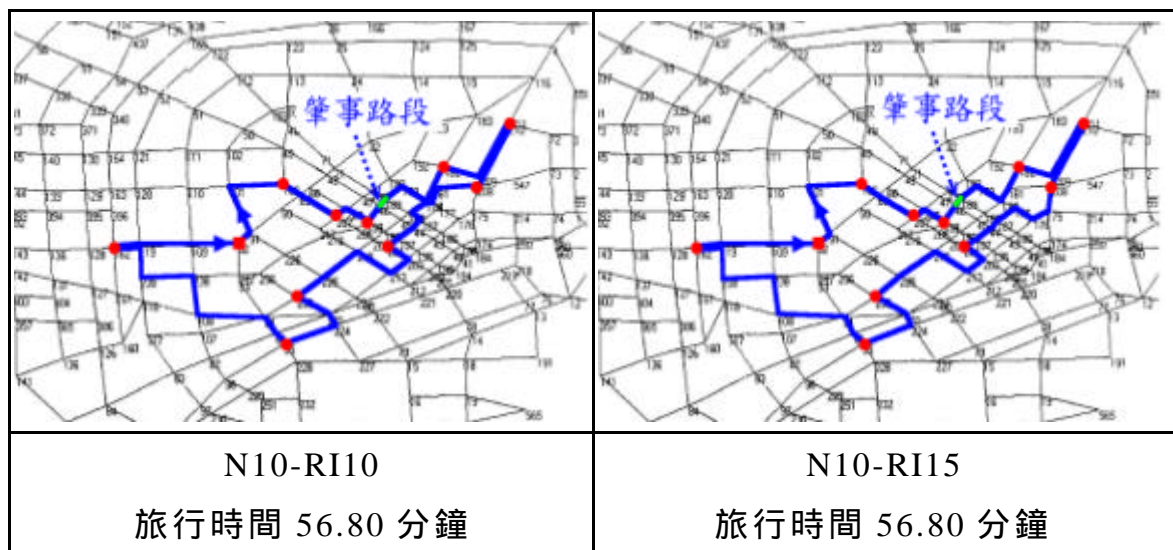


圖 5.14 事故發生下巡迴路線比較圖

表 5.9 巡迴路線路段數、旅行時間、停等時間及改善百分比

	路段數	旅行時間(分鐘)	路線停等 時間(分鐘)	旅行時間 改善百分比
N10-RI00	42	60.80	20.38	-
N10-RI05	42	48.80	12.57	19.74%
N10-RI10	43	56.80	16.21	6.58%
N10-RI15	43	56.80	16.28	6.58%

2.需求點個數 20

由圖 5.15 及表 5.10 可知,車輛在未獲得任何即時資訊的 N20-RI00 情況下行經肇事路段,其總旅行時間為 70.80 分鐘,路線上的停等時間為 19.77 分鐘;在 N10-RI05 的情況下,車輛在獲得即時資訊後,更新了巡迴路線,避開了肇事的路段,總旅行時間降為 48.80 分鐘,改善百分比達 24.48%,而停等時間則降為 14.93 分鐘;在 N20-RI10 及 N20-RI15 的情況下,由於車輛在接受即時資訊之前已到達肇事路段,因此並無法即時避開肇事路段,其總旅行時間為 70.61 分鐘,停等時間皆為 19.74 分鐘。

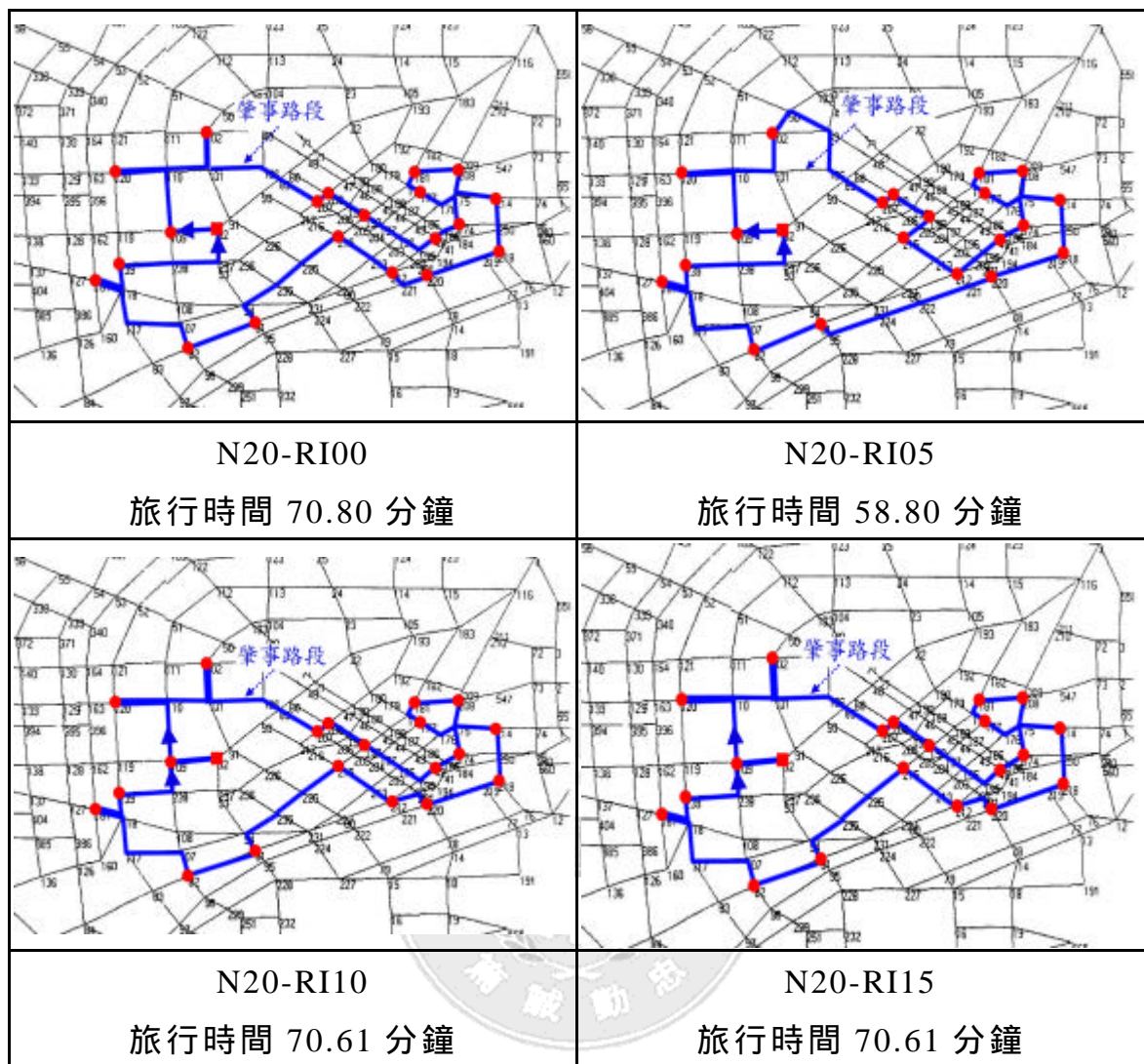


圖 5.15 事故發生下巡迴路線比較圖

表 5.10 巡迴路線路段數、旅行時間、停等時間及改善百分比

	路段數	旅行時間(分鐘)	路線停等 時間(分鐘)	旅行時間 改善百分比
N20-RI00	50	70.80	19.77	-
N20-RI05	51	58.80	14.93	24.48%
N20-RI10	50	70.61	19.74	0.15%
N20-RI15	50	70.61	19.74	0.15%

5.6 小結

根據數值實驗分析的結果，歸納出以下幾點結論：

1. 情境 1 與情境 2 相比較後，由於情境 1 並不具有尖峰的特性，車輛數維持穩定的成長，因此在道路上擁擠的情形並不明顯，使得即時資訊無法發揮其效用。
2. 在情境 2 中可明顯的發現，提供即時性的巡迴路線可以有效的縮短商用車的旅行時間，但在路線轉換的同時，也有可能轉換到較差的巡迴路線。
3. 在考慮即時資訊下動態 VRP 的實驗裡，並無法得知何種即時資訊提供頻率為最佳。
4. 由結合即時資訊與歷史資訊的實驗中可知，擁擠路段預測確實能減少路線上的停等時間，改善現有的巡迴路線解。
5. 本研究對擁擠路段的設計原則，僅適用於研究中的實驗路網，若考慮將其應用於其他路網時，應依據不同的區域特性及不同的車輛產生型態等因素，重新設定設計原則。
6. 在事故發生的情況下，透過即時資訊的提供能避開肇事路段，但其提供的時機必須在到達肇事路段之前，以便於進行路線的轉換。
7. 現有車輛服務分區計有 131 個需求點，測試中對需求點數目的討論只區分為 10 個及 20 個需求點，未來可擴大車輛的服務分區，進行不同需求點個數的測試。

第六章 結論與建議

本研究主要應用禁制搜尋法求解單一車輛且無容量限制的 VRP 問題，研究中考慮了不同時間點所收集的交通資訊，進行路線的規劃，並透過模擬指派模式 DYNASMART 反應即時性路徑與巡迴路線所產生的效益。以下針對研究分析所得之結果，提出相關結論與後續研究之建議。

6.1 結論

1. 本研究應用禁制搜尋法求解車輛巡迴路線問題，並由國際例題的測試中可知，禁制搜尋法在問題的求解上，其平均誤差百分比可達 0.65%。
2. 本研究透過模擬模式 DYNASMART 評估即時性巡迴路線的效益，研究中發現，適時的提供即時資訊，確實能改善商用車的旅行時間，但頻繁的路線轉換，也可能會使巡迴路線陷入較差的路線解，增加車輛的巡迴時間。
3. 由情境設計的實驗中可知，當路段上有擁擠的情形發生時，較能發揮即時資訊的功用。
4. 本研究提出一預測擁擠路段的想法與概念，並將其應用於動態 VRP 問題的求解，透過實驗分析後可知，藉由擁擠路段預測確實能減少巡迴路線上的停等時間，降低車輛的巡迴時間。
5. 透過結合即時資訊與歷始資訊進行 VRP 問題的求解，在不同頻率下所提供的即時性巡迴路線解較儘考慮即時資訊時一致，且其效益亦較高。
6. 在事故發生的情況下，透過即時資訊的提供進行路線轉換，可避免行經肇事路段，降低車輛的巡迴時間。

6.2 建議

1. 研究中對於禁制名單的設計與停止準則僅參考 Knox 及吳泰熙等人的文獻，未來可針對不同的需求點數，設計不同的禁制名單與遞迴數，以提高計算上的精度，並在計算的精度與速度上求得平衡點。
2. 研究中對即時資訊提供的頻率設計有 3 種(每 5 分鐘、10 分鐘、15 分鐘提供一次)，未來對於即時資訊提供的頻率不應以時間考量，應以路線改善幅度為原則，並避免提供重覆的資訊。
3. 從數值實驗中可知，提供即時資訊對巡迴路線而言，有時會產生負效用，因此未來可分析在不同的車輛產生型態、需求點分佈情況、事故發生的時間，再進行分析比較，以求出提供即時資訊最適當的時機。
4. 研究中的車輛在途中接受即時資訊後，必須到達下一個需求點才可進行路線的轉換，未來可考慮允許車輛於途中進行路線轉換。
5. 研究中考慮的需求點為一已知固定值，未來可考慮需求點為變動情形下的 VRP 問題。

參考文獻

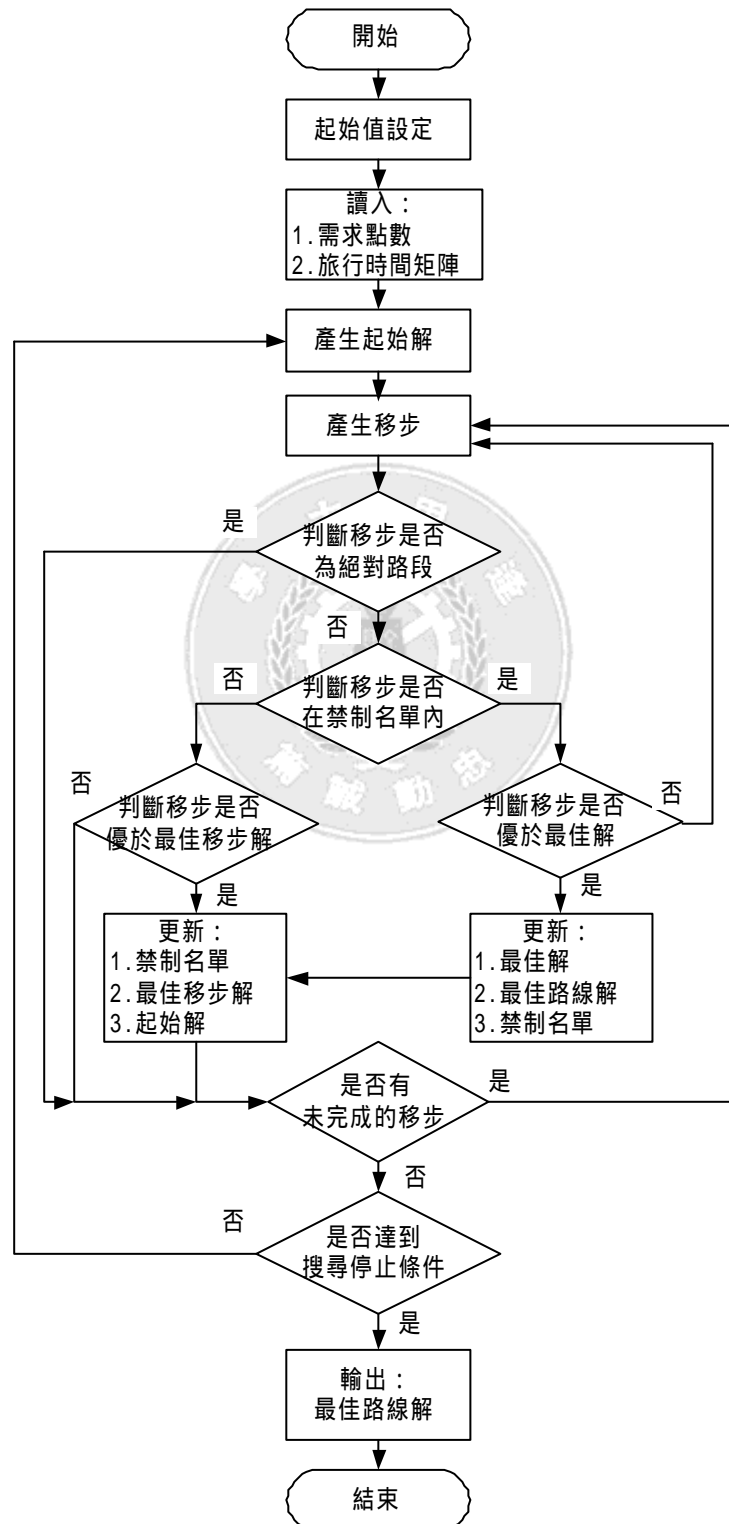
- 王文貞(1997),「圖書配送車輛排程問題之研究」,成功大學交通管理研究所。
- 呂英志(2002),「即時資訊下車輛路線問題之研究」,逢甲大學交通工程與管理研究所碩士論文。
- 林明俊(1998),「隨機環境下多車種車派車問題之研究」,中原大學工業工程研究所碩士論文。
- 卓裕仁,「以巨集啟發式方法求解多車種與週期性車輛路線問題之研究」,交通大學運輸工程與管理研究所博士論文。
- 吳泰熙、張欽智(1997),「以禁制搜尋法則求解推銷員旅行問題」,大業學報,第六卷,第一期,頁 87-99。
- 胡大瀛(1996),「DYNASMART 在動態交通指派模式中之應用」,中華民國運輸學會第 11 屆研討會論文集,頁 873-883。
- 胡大瀛、陳克宇、陳建緯、洪百賢(1999),「VRP 之交通模擬評估架構」,中華民國第四屆運輸網路研討會論文集,頁 62-71。
- 胡大瀛、洪百賢(2002),「公車動態到站時間預測成果報告」。
- 梅明德(1999),「線上型時窗限制車輛路線問題之模式與求解演算法」,中央大學土木工程研究所博士論文。
- 陳建緯(1999),「即時資訊下動態派遣策略之研擬」,逢甲大學交通工程與管理學系,國科會大專學生參與專題計畫研究成果報告。
- 陳建緯(2001),「大規模旅行推銷員問題之研究:鄰域搜尋法與巨集啟發式解法之應用」,交通大學運輸工程與管理研究所碩士論文。
- 陳勝男(1996),「禁忌搜尋法應用於車輛路線問題之研究」,大葉大學工業工程研究所碩士論文。
- 黃金智(1999),「隨機型車輛途程問題解法之研究」,大葉大學工業工程研究所碩士論文。
- 黃燦煌(2000),「即時資訊狀況下駕駛人路線選擇行為之分析」,交通大學交通運輸研究所博士論文。
- 韓復華、卓裕仁(1996),「門檻接受法在 TSP 問題上之應用」,運輸計畫季刊第 25 卷第 2 期,頁 163-188。

- 韓復華、卓裕仁(1996),「門檻接受法、噪音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9卷第3期,頁113-144。
- 韓復華、陳正元(1996),「車輛路線問題啟發式解法在PC上直行績效之研究」,中華民國運輸學會第七屆研討會論文集,第四冊,頁649-662。
- 魏慶地(1999),「行前交通資訊逐日動態模擬之研究」,台灣大學土木工程學研究所博士論文。
- Angeles, E., Speranza, M.G.(2002), "The periodic vehicle routing problem with intermediate facilities," *European Journal of Operational Research*, 139, pp.233-247.
- Barbarosoglu, G. and Ozgur, D.(1999), "A tabu search algorithm for the vehicle routing problem," *Computer & Operations Research*, 26, pp.255-270.
- Bodin, L., Golden, B., Assad, A., Ball, M.(1983), "Routing and scheduling of vehicles and crews: the state of the art," *Computer & Operations Research*, Vol. 10, No. 2, pp. 63-211.
- Brandao, J., Mercer, A.(1997), "A tabu search algorithm for the multi-trip vehicle routing and scheduling problem," *European Journal of Operational Research*, 100, pp.180-191.
- Chang, G.L., Mahmassani, H.S. and R. Herman(1985), "A Macroparticle Traffic Simulation Model to Investigate Peak-Period Commuter Decision Dynamic," *Transportation Research Record* 1005, pp.107-120.
- Fisher, M.L., and Jaikumar, R.(1981), "A generalize assignment heuristic for vehicle routing," *Networks*, Vol. 11, pp.109-124.
- Gendreau, M., Hertz, A., Laporte, G.(1994), "A tabu search heuristic for the vehicle routing problem," *Management Science*, Vol. 40, No. 10, pp.1276-1290.
- Gendreau, M., Laporte, G. and Seguin, R.(1996), "Invited Review Stochastic Vehicle Routing," *European Journal of Operational Research*, 88, pp.3-12.
- Glover, F., and Laguna, M.(1997), *Tabu Search*, Kluwer Academic Publishers.
- Gillett, B., and Miller, L.(1974), "A heuristic algorithm for the vehicle dispatch problem," *Operations Researches*, 22, pp.340-349.
- Hu, T.Y., Rothery, R. W., and Mahmassani, H.S.(1992), "DYNASMART: Dynamic Network Assignment-Simulation Model for Advanced Road Telematics," Technical Report DTFH61-90-C-00074-TWPI, Center for Transportation, The University of Texas at Austin.

- Hu, T.Y.(2001), "Evaluation Framework for Dynamic Vehicle Routing Strategies Under Real-Time Information," Transportation Research Record, 1774, pp.115-122.
- Jayakrishnan, R.(1992), In-vehicle Information Systems for Network Traffic Control: A Simulation Framework to Study Alternative Guidance Strategies, Ph.D. Dissertation, The University of Texas at Austin.
- Knox, J.(1994), "Tabu search performance on the symmetric traveling salesman problem" Computers & Operations Research, Vol 21, No 8, pp.867-876.
- Psaraftis H.N.(1995), "Dynamic vehicle routing: Status and prospect," Annals of Operations Research, 61, pp.143-164.
- Renaud, J., Laporte, G., F. Boctor, F.F.(1996), "A tabu search heuristic for the multi-depot vehicle routing problem," Computer & Operations Research, Vol. 23, No. 3, pp.229-235.
- Tan, K.C., et. al(2001), "Heuristic methods for vehicle routing problem with time windows," Artificial Intelligence in Engineering, 15, pp.281-295.
- Tangiguchi, E., Yamada, T. and Tamagawa, D.(1999), "Modeling Advanced Routing and Scheduling of Urban Pickup/Delivery Trucks," Proc., 6th World Congress on Intelligent Transportation System.
- TSPLIB , <http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/>.

附錄一 動態車輛巡迴路線問題之求解程式流程與程式

動態車輛巡迴路線問題之求解程式流程



動態車輛巡迴路線問題之求解程式

subroutine nearvrp() C- 本程式為 DYNASMART 內 mainvrp.f 之副程式

C- 宣告變數

```
c      implicit none

      include "parmt.inc"

      real    nw(nu_van,nu_van)      C -- 路線成本矩陣

      common /nw/nw

      integer tabulist(nu_van,nu_van)  C -- 空間上的禁制名單

      integer optimal(nu_van)          C -- 運算所得最佳路線解

      integer optimal_near(nu_van)     C -- 運算所得最佳路線解

      common /optimal_near/optimal_near

      integer r(nu_van)                C -- 路線解

      integer t(nu_van)                C -- 路線暫存檔

      real cost,minc,mcost  C -- 路線成本,最小成本,移步時最小成本

      real minc_near                C -- 運算所得最佳解成本

      common /minc_near/minc_near

      integer i,j,k,a,b,c,z

      integer maxtabu                C -- 禁制期間

      common /maxtabu/maxtabu

      integer aaaa,symbol  C -- 最大遞迴數

      common /aaaa/aaaa

      integer tabulist_tabu(nu_van,nu_van)  C -- 禁制名單暫存檔

      integer tabulist_untabu(nu_van,nu_van) C -- 禁制名單暫存檔

      integer nr_tabu(nu_van)  C -- 起始路線暫存檔

      integer nr_untabu(nu_van) C -- 起始路線暫存檔

      REAL soltmp,soli,solitm

      integer soltm(nu_van)

      real n                C -- 需求點數

      integer ii,jj
```

```

real node_number C -- 需求點數

common /node_number/node_number

integer tabulist2(nu_van,nu_van) C -- 時間上的禁制名單

common /tabulist2/tabulist2

C -- 路線解成本預設值
C -- 令 n 為需求點數
n = node_number

C -- 禁制名單歸零
do ii=1,n
do jj=1,n
tabulist(ii,jj) = 0
end do
end do

C -- 最小成本設為 9999
C -- 程式中相關變數、矩陣變更為預設值、
minc = 9999
tabulist=0
optimal=0
optimal_near=0
r=0
t=0
soltm=0
tabulist_tabu=0
tabulist_untabu=0
nr_tabu=0
nr_untabu=0
soltmp=0
soli=0
solitm=0

```



C -- 以最近鄰點法產生起始解

```

    soli=1
    do a=1,n-1
        i=soli
        soltmp=999999
        do j=2, n-1
            if (nw(i,j)/=0) then
                if (nw(i,j)<soltmp .and. soltm(j)==0 ) then
                    soltmp=nw(i,j)
                    solitm=j
                end if
            end if
        end do
        if (n > 2) then
            soli=solitm
            soltm(solitm)=1
            r(a+1)=solitm
        end if
    end do

    r(1)=1
    r(n)=n

```



C -- 起始路線解暫存檔 t=r

```

    t=r

```

C -- 計算起始解成本

```

    cost = 0
    do i=1, n-1
        cost = cost + nw(r(i),r(i+1))
    end do

```

C -- 令最佳解成本 minc 為起始解成本 cost

C -- 令最佳路線解矩陣 optimal 為起始路線解矩陣 r

```
minc = cost
```

```
do ii=1,n
```

```
optimal(ii) = r(ii)
```

```
end do
```

C -- 找出所有的可行移步解

C -- 最大遞迴數 aaaaa 設定

C -- 移步解成本歸回預設值 99999

```
do z=1,aaaaa
```

```
mcost = 99999
```

```
symbol = 0
```

C -- 2-opt 交換法

```
do i=1,n-3
```

```
do j=i+2,n-1
```

```
r = t
```

```
r(i+1)=t(j)
```

```
r(j)=t(i+1)
```

```
b=j
```

```
do a=i+2,j
```

```
r(a)=t(b-1)
```

```
b=b-1
```

```
end do
```

C -- 移步後的路線成本 cost

```
cost = 0
```

```
do a=1, n-1
```

```
cost = cost + nw(r(a),r(a+1))
```

```
end do
```



C -- 判斷該移步是否位於時間上的禁制名單內，若是則選擇下一個移步，若否則繼續

```
do b=1,n-1
    if(tabulist2(r(b),r(b+1))==999)then
        goto 100
    end if
end do
```

C -- 判斷該移步是否位於空間上的禁制名單內

C -- 該移步被禁止

```
if ( tabulist(r(i),r(i+1))> 0 .or. tabulist(r(j),r(j+1))> 0 ) then
```

C -- 判斷是否符合渴望準則，若符是則更新最佳解、最佳路線解、禁制名單暫存檔

```
if (cost < minc)then
    symbol = 1
    minc = cost
    optimal = r
    nr_tabu = r
```

C -- 更新禁制名單

```
tabulist_tabu=tabulist
do a=1,n
do b=1,n
    if(tabulist_tabu(a,b)>0)then
        tabulist_tabu(a,b)=tabulist_tabu(a,b)+1
    end if
end do
end do

tabulist_tabu(r(i),r(i+1))=1
tabulist_tabu(r(i+1),r(i))=1
tabulist_tabu(r(j),r(j+1))=1
tabulist_tabu(r(j+1),r(j))=1
```



C -- 禁制名單若超過禁制期間則歸 0

```

do a=1,n
do b=1,n
if(tabulist_tabu(a,b)>maxtabu)then
tabulist_tabu(a,b)=0
end if
end do
end do
else
goto 100
end if
end if

```

C -- 判斷該移步是否位於空間上的禁制名單內

C -- 該移步不被禁止，若是則更新最佳移步解、禁制名單暫存檔

```

if (tabulist(r(i),r(i+1)))==0 .and. tabulist(r(j),r(j+1)))==0) then
if (cost < mcost)then
mcost = cost
nr_untabu = r
tabulist_untabu=tabulist

```

C -- 更新禁制名單

```

do a=1,n
do b=1,n
if (tabulist_untabu(a,b)>0)then
tabulist_untabu(a,b)=tabulist_untabu(a,b)+1
end if
end do
end do

tabulist_untabu(r(i),r(i+1))=1
tabulist_untabu(r(i+1),r(i))=1
tabulist_untabu(r(j),r(j+1))=1
tabulist_untabu(r(j+1),r(j))=1

```

C -- 禁制名單若超過禁制期間則歸 0

```

do a=1,n
do b=1,n
if(tabulist_untabu(a,b)>maxtabu)then
tabulist_untabu(a,b)=0
end if
end do
end do

if (mcost<minc)then
symbol=0
minc = mcost
optimal=nr_untabu
end if
end if
end if
100 continue
end do
end do

```

C -- 更新起始路線與禁制名單

```

if (symbol==1)then
r = nr_tabu
t = nr_tabu
tabulist = tabulist_tabu
else
r = nr_untabu
t = nr_untabu
tabulist = tabulist_untabu
end if

end do

```



C -- 更新最佳解成本與最佳路線解矩陣

```
do ii=1,n
    optimal_near(ii) = optimal(ii)
end do

minc_near = minc

write (*,*) 'minc_near',minc_near

return

end
```

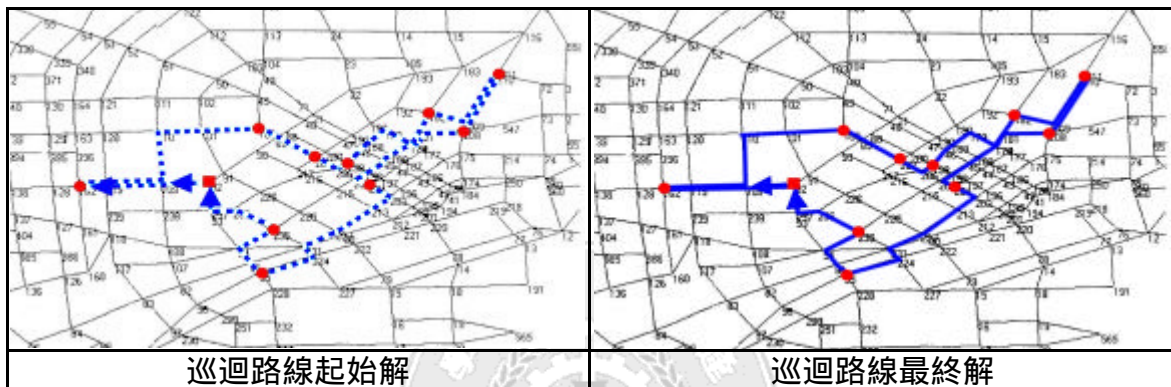


附錄二 動態需求點順序

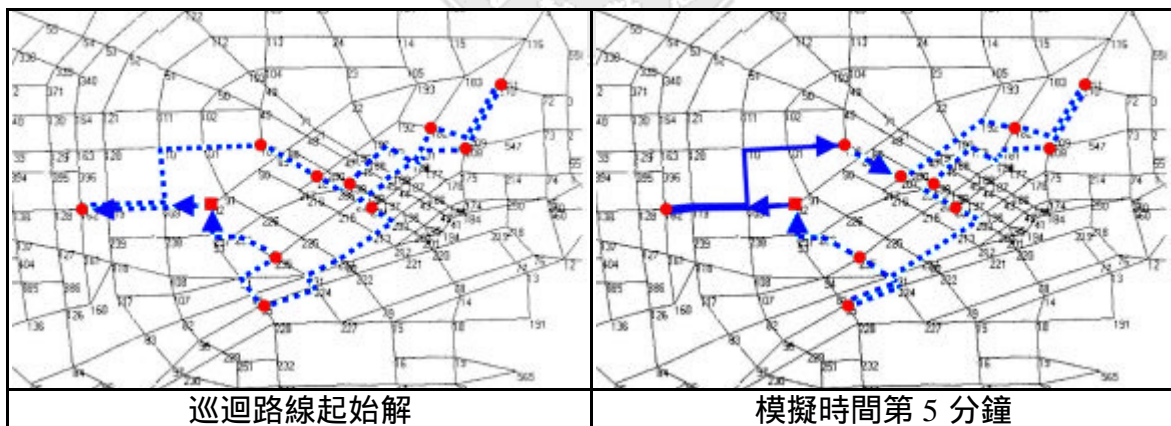
考慮即時資訊下動態需求點順序

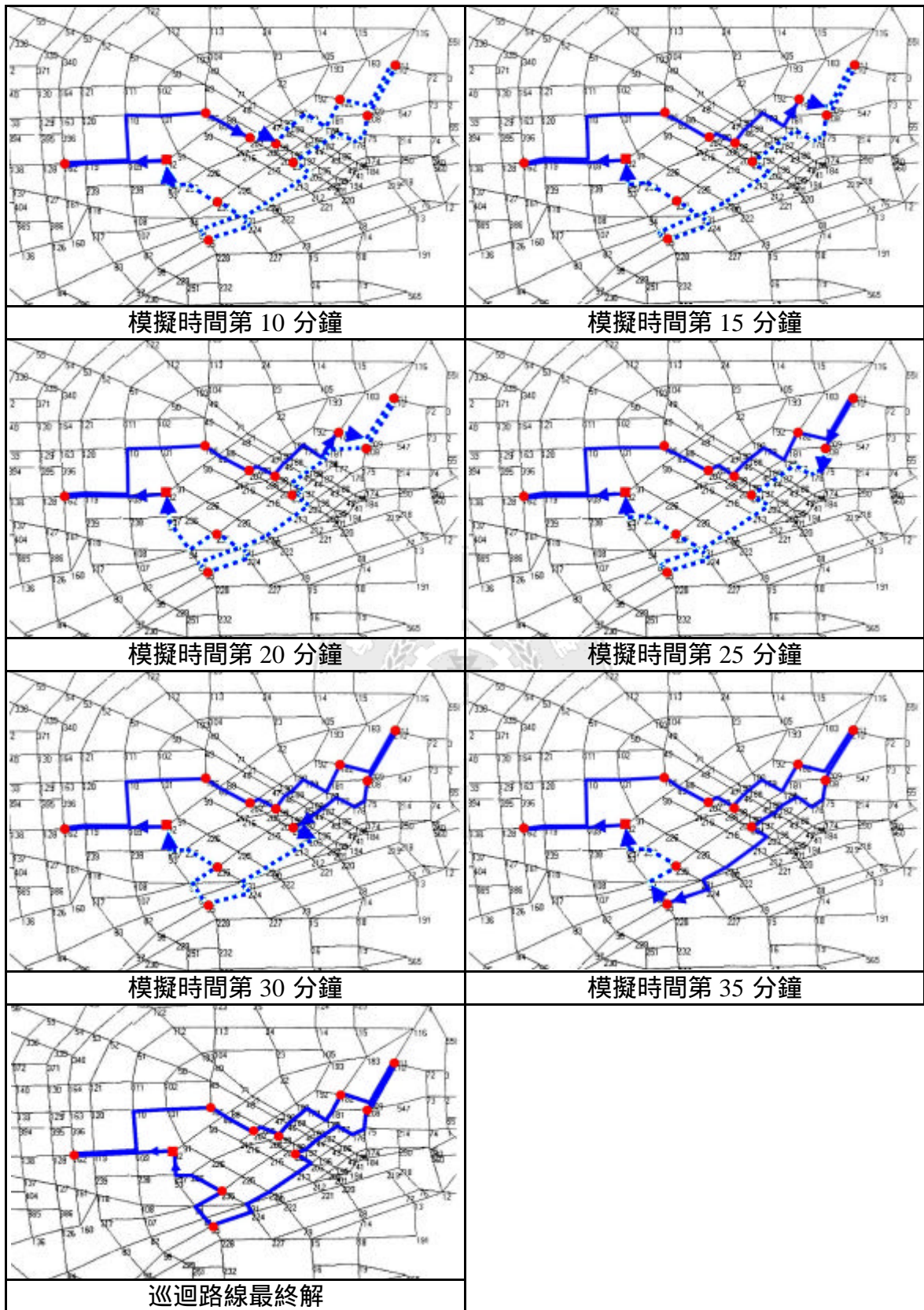
情境 1-10 個需求點

不提供即時資訊

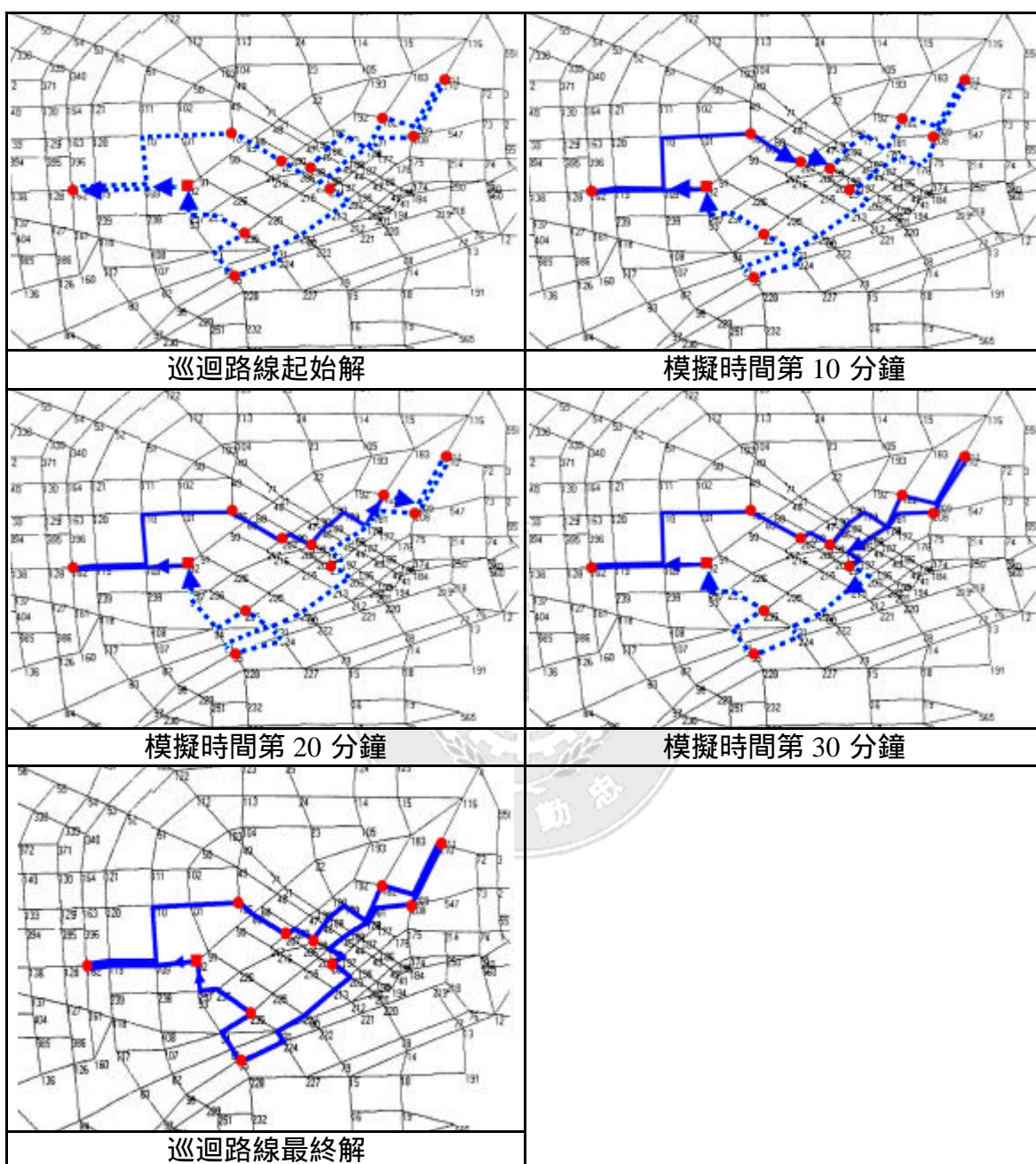


每 5 分鐘提供一次即時資訊

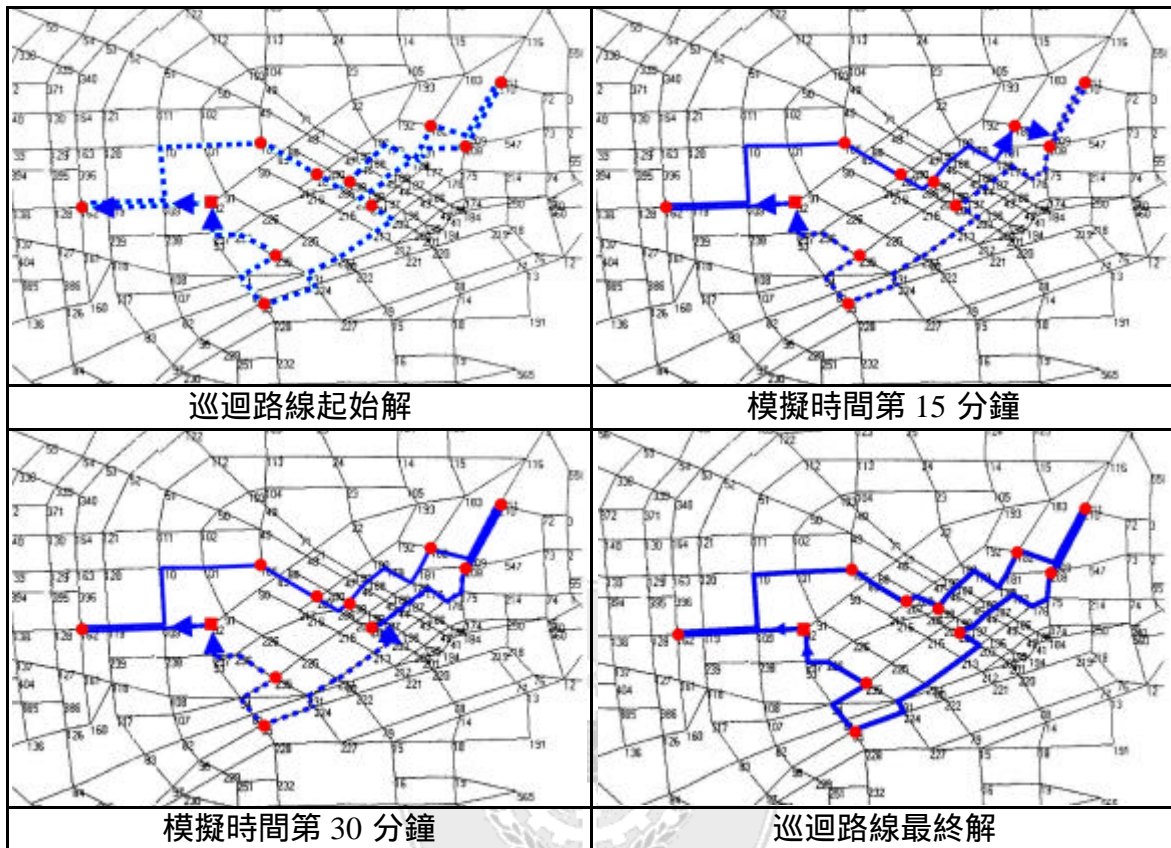




每 10 分鐘提供一次即時資訊

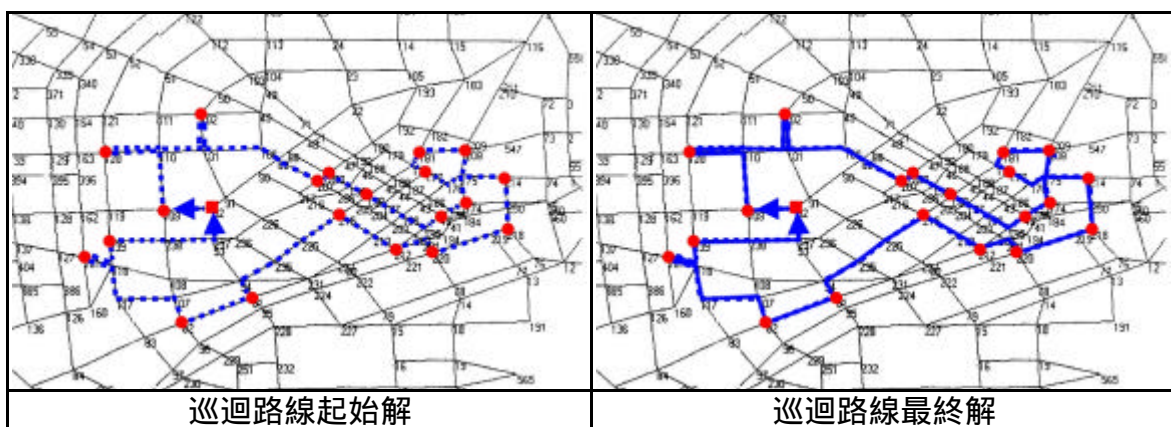


每 15 分鐘提供一次即時資訊

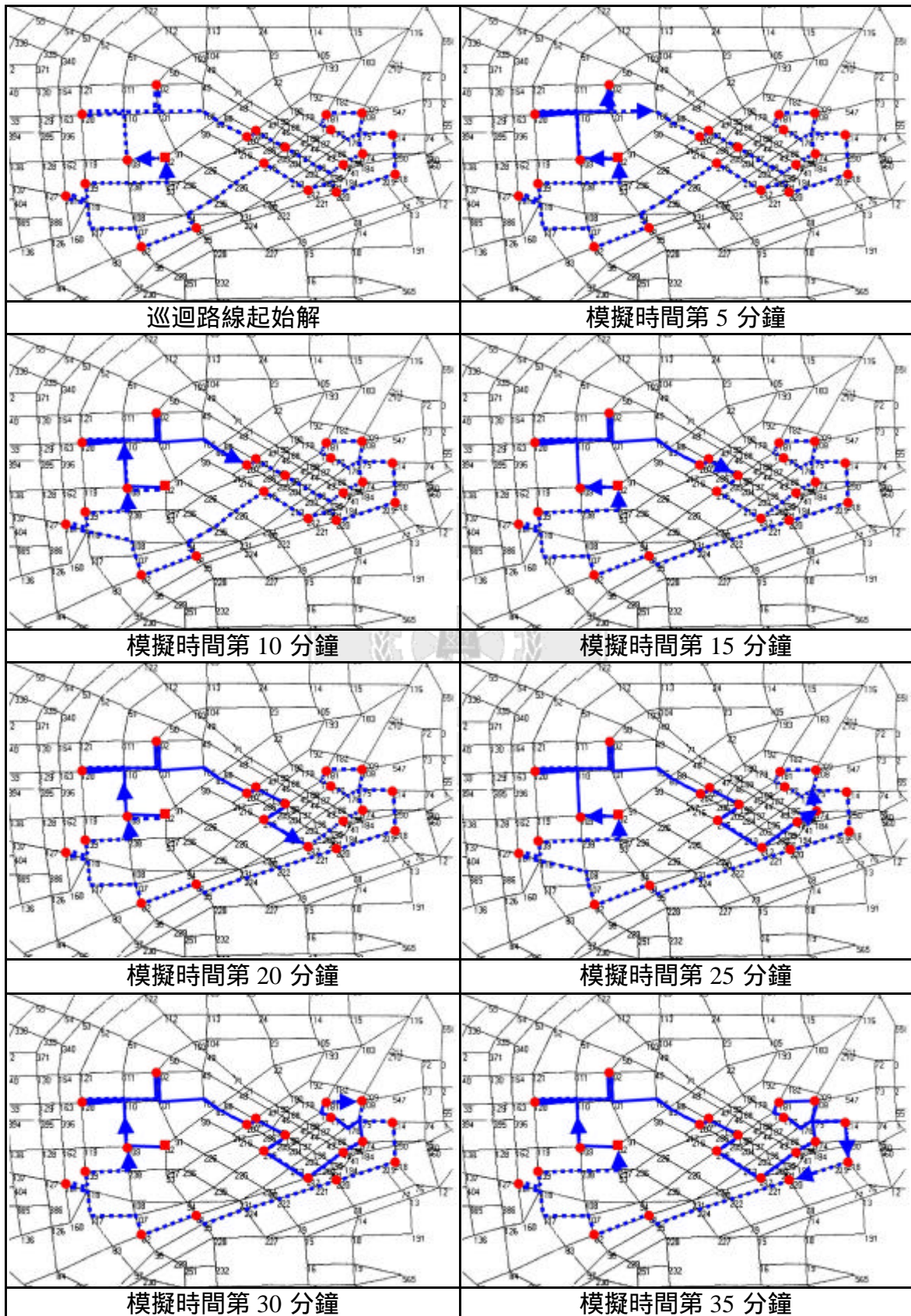


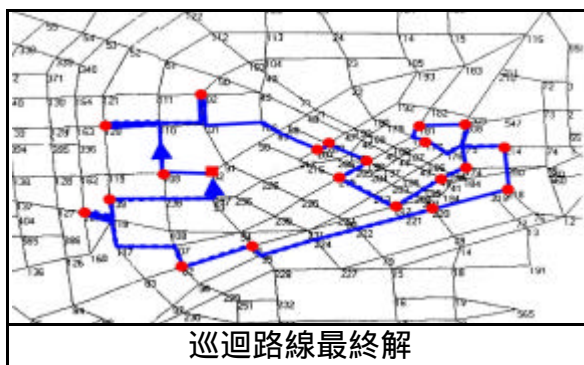
情境 1-20 個需求點

不提供即時資訊

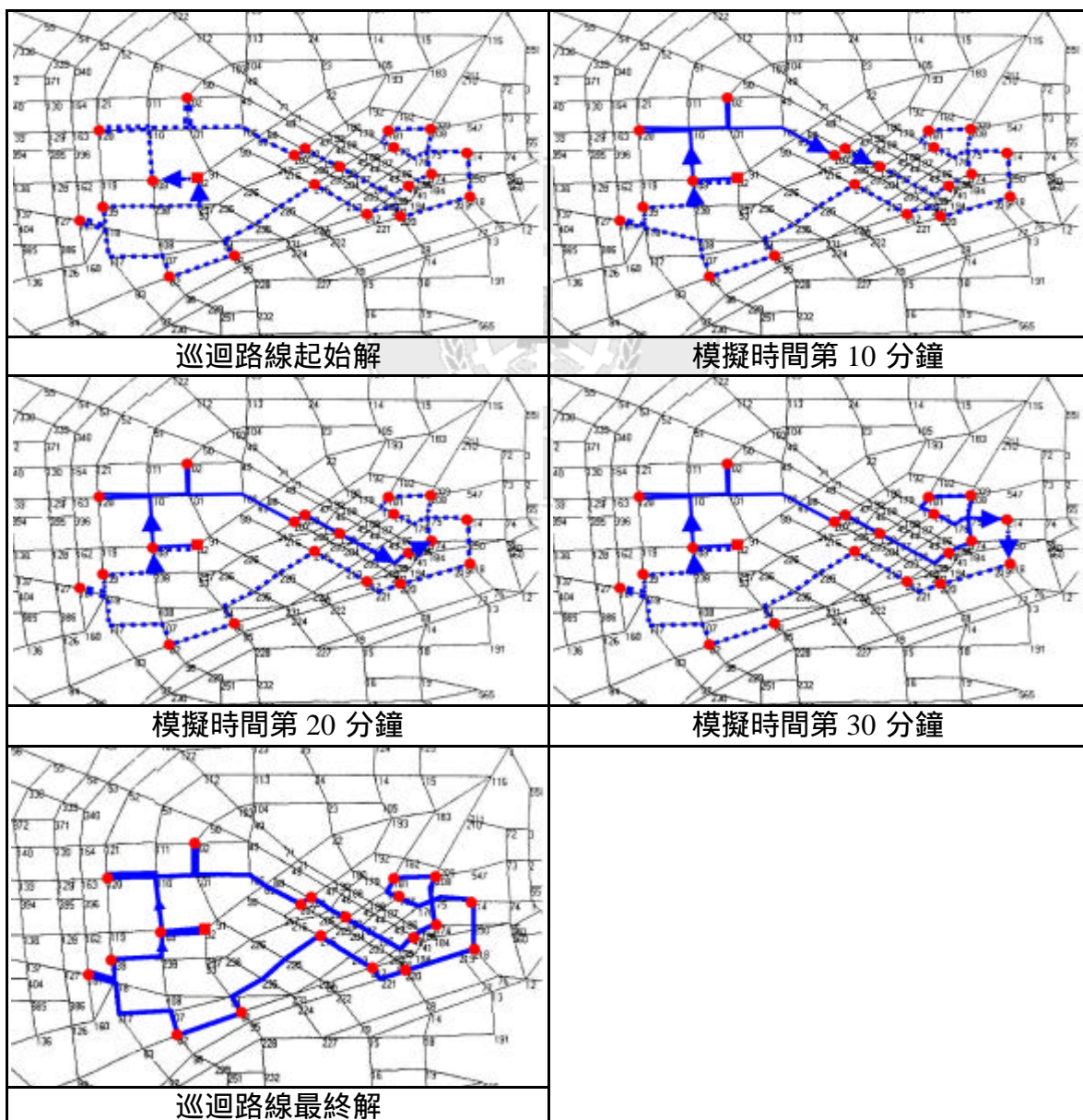


每 5 分鐘提供一次即時資訊

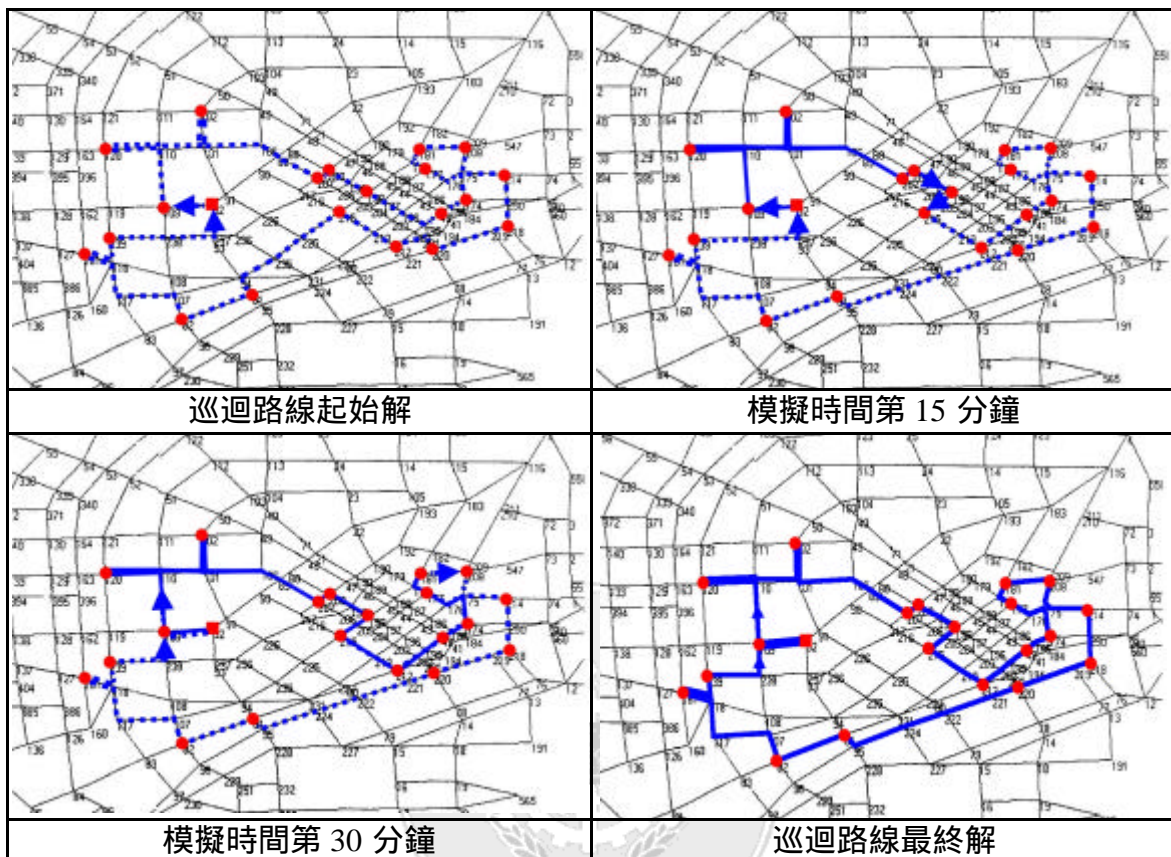




每 10 分鐘提供一次即時資訊

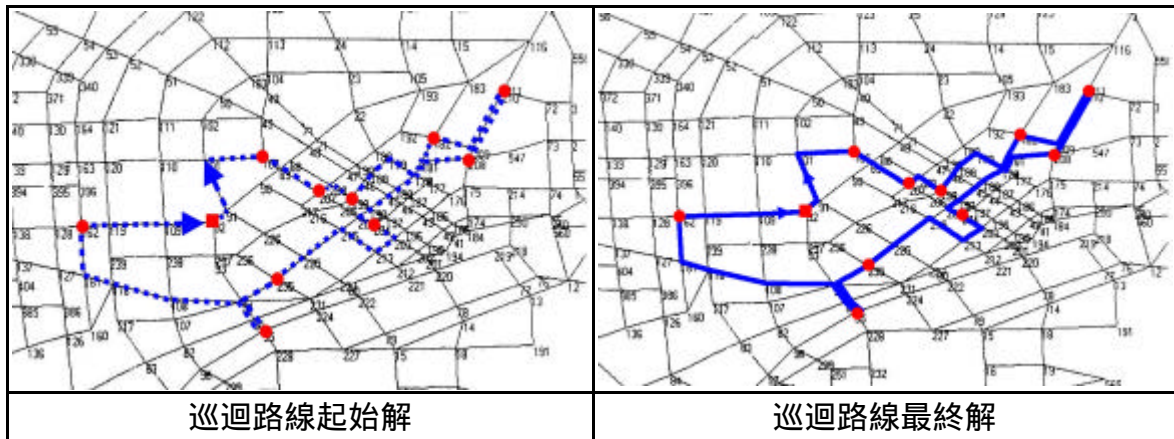


每 15 分鐘提供一次即時資訊

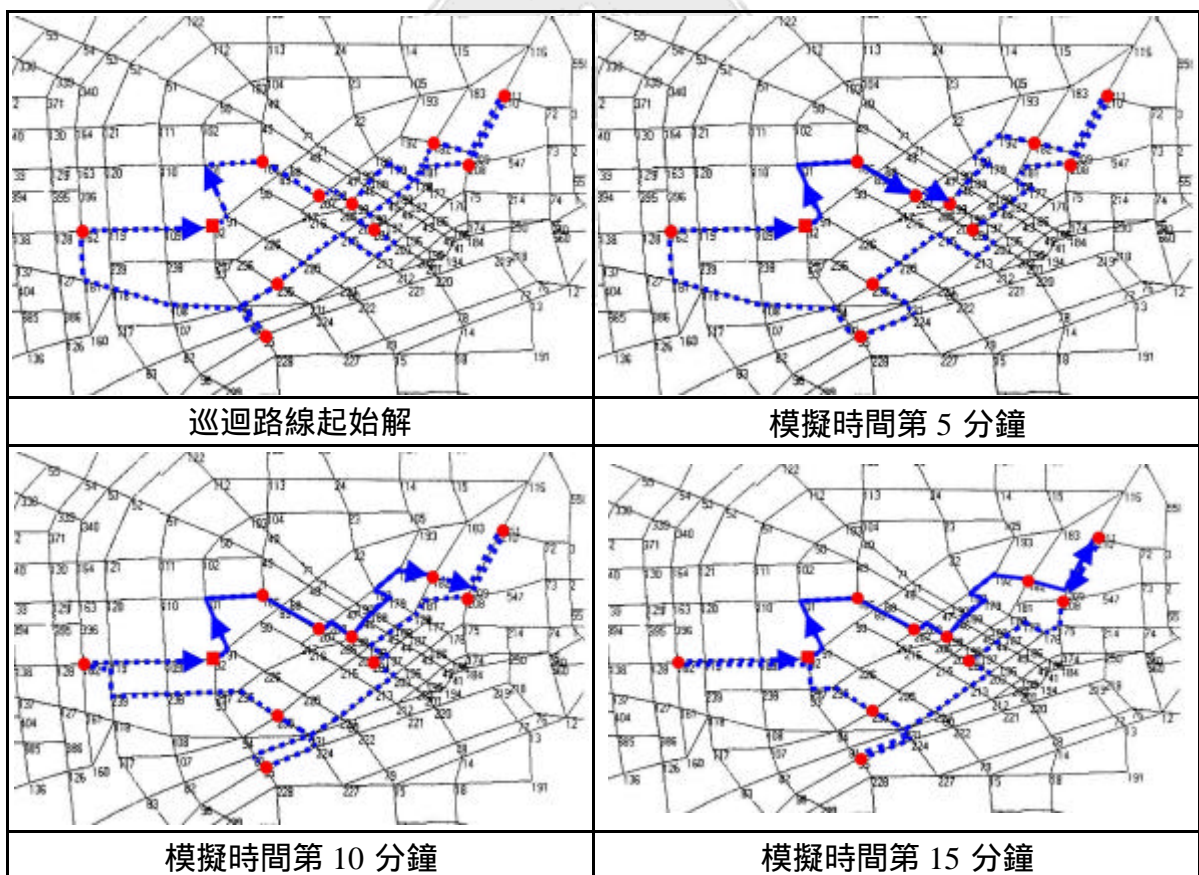


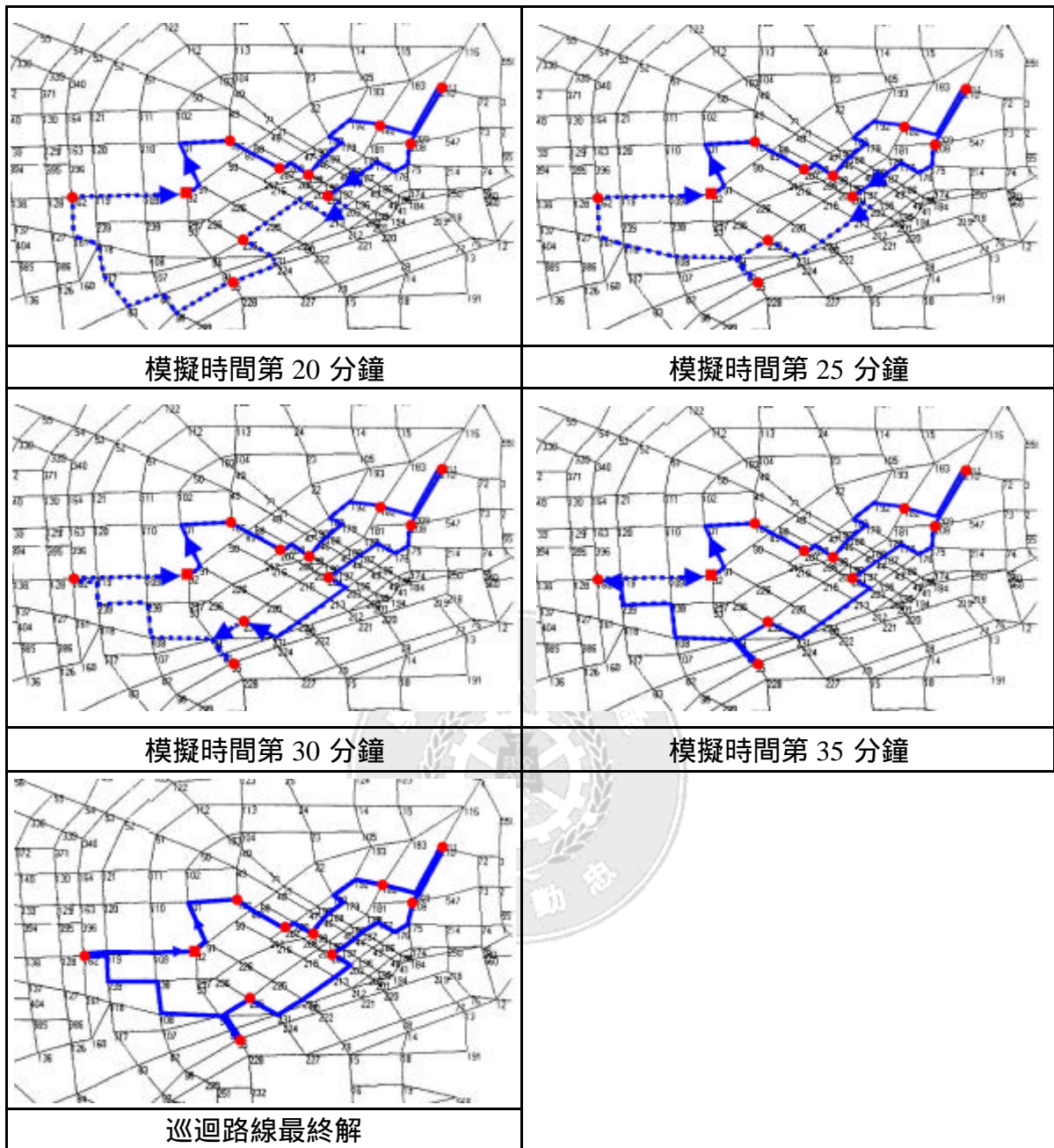
情境 2-10 個需求點

不提供即時資訊

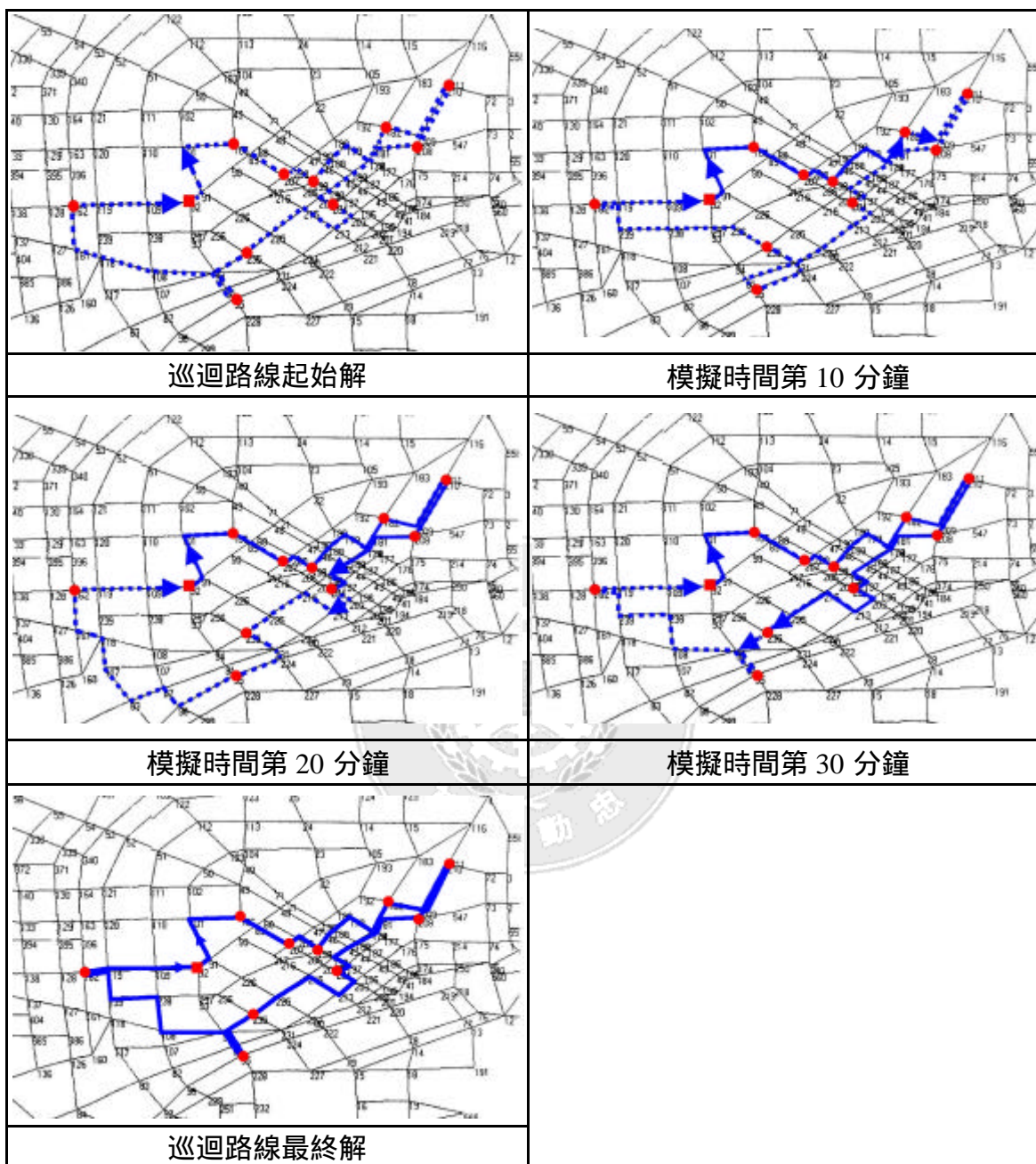


每 5 分鐘提供一次即時資訊

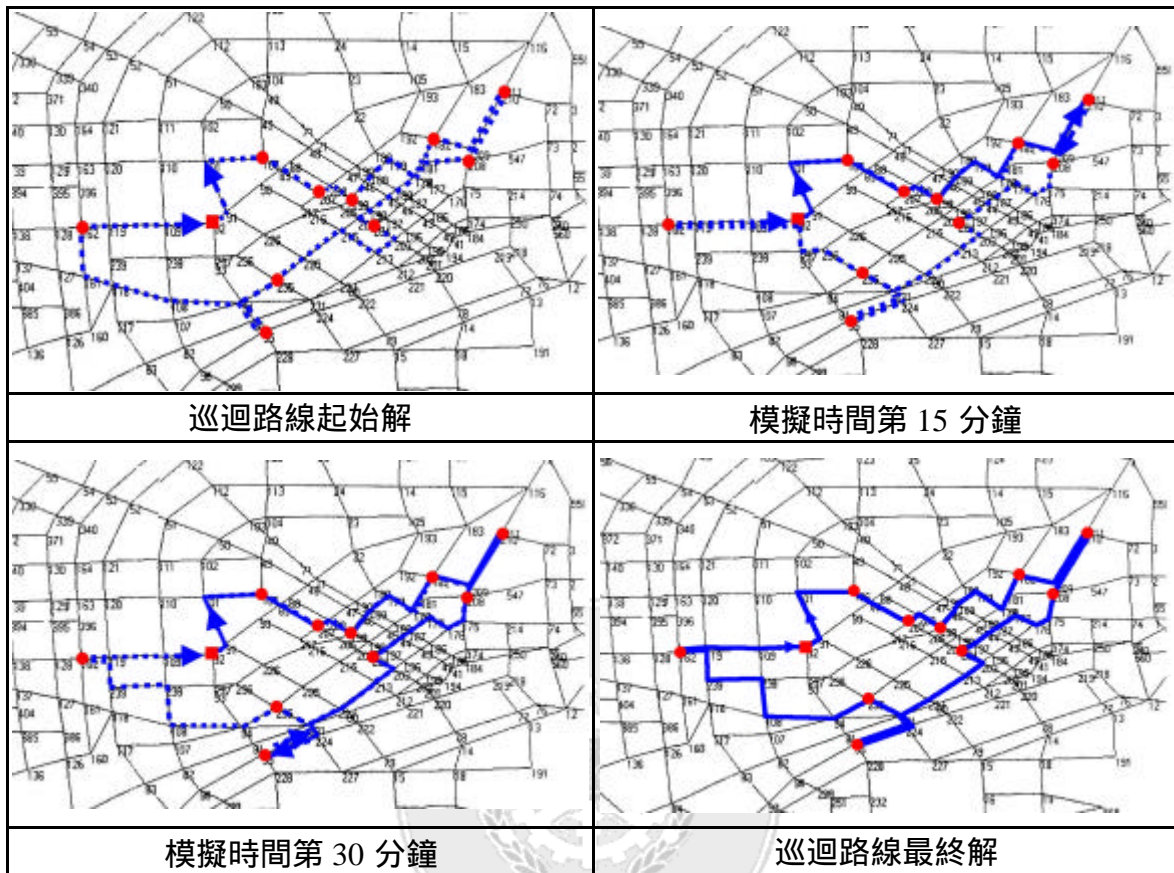




每 10 分鐘提供一次即時資訊

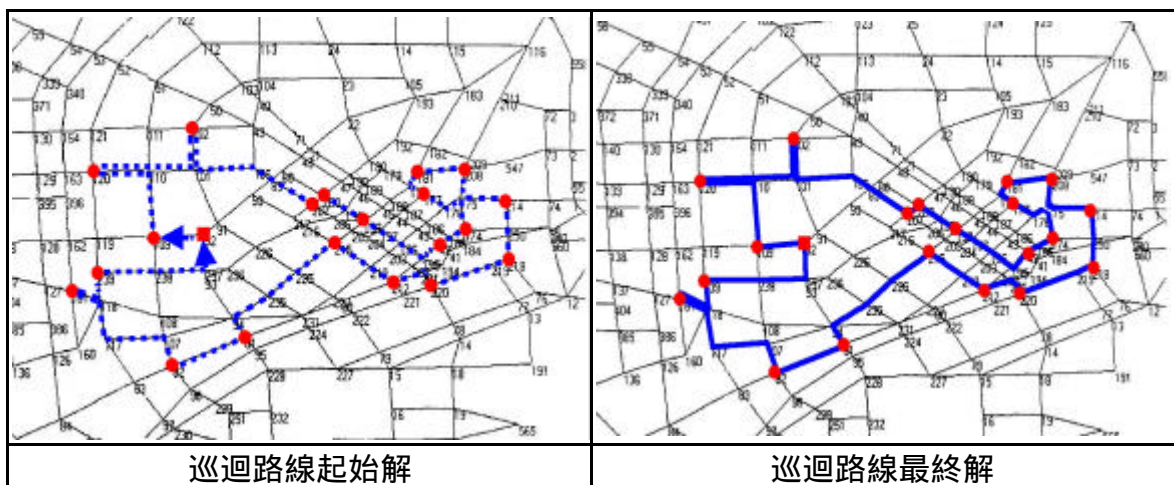


每 15 分鐘提供一次即時資訊

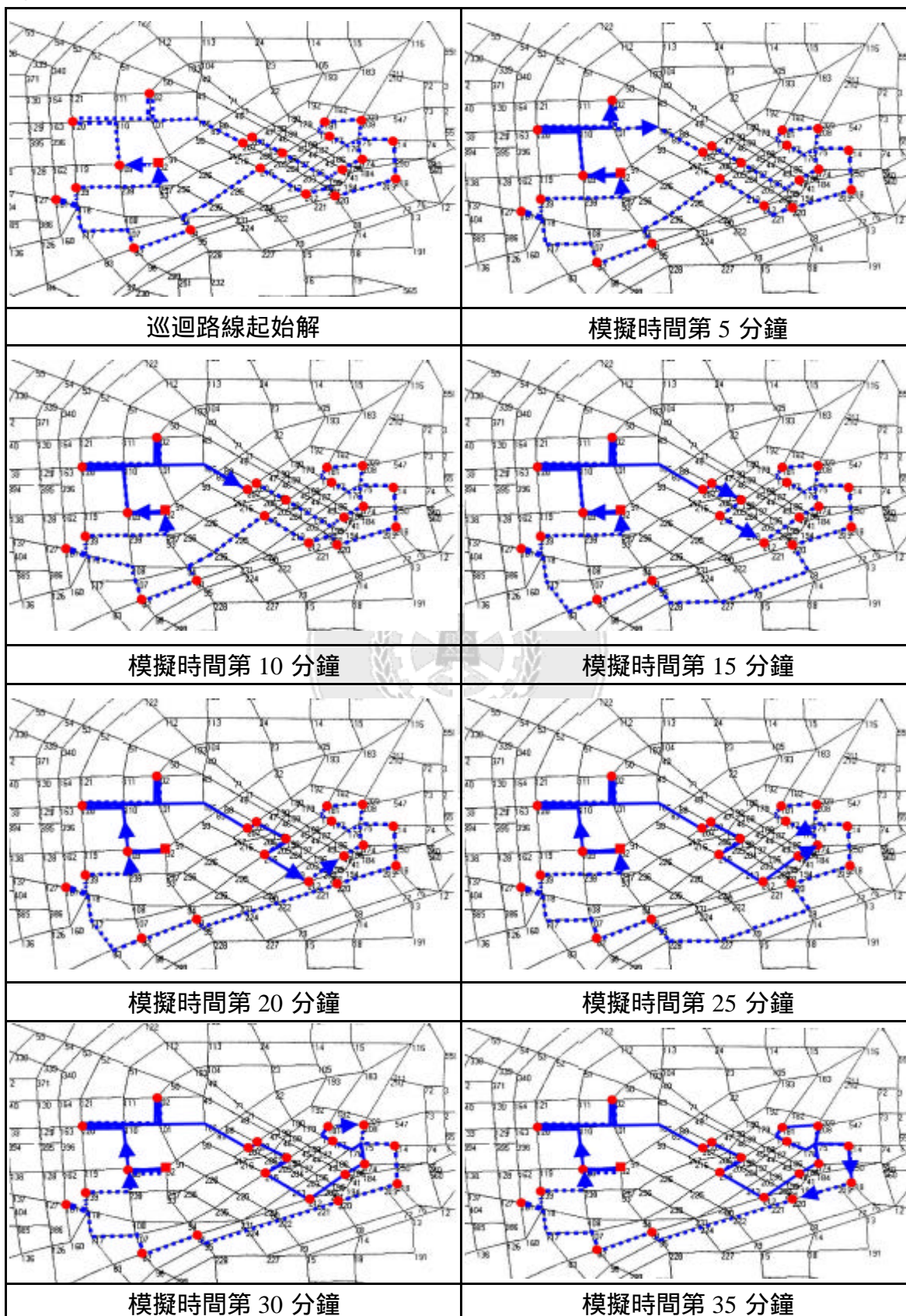


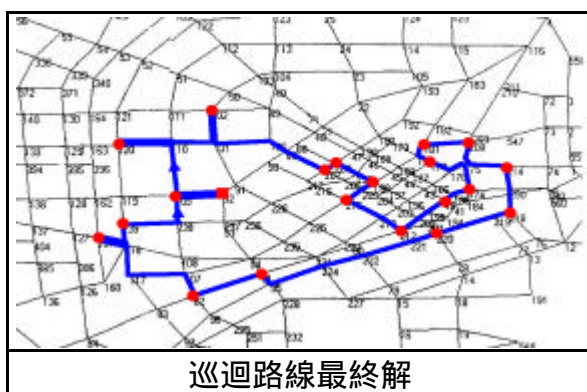
情境 2-20 個需求點

不提供即時資訊

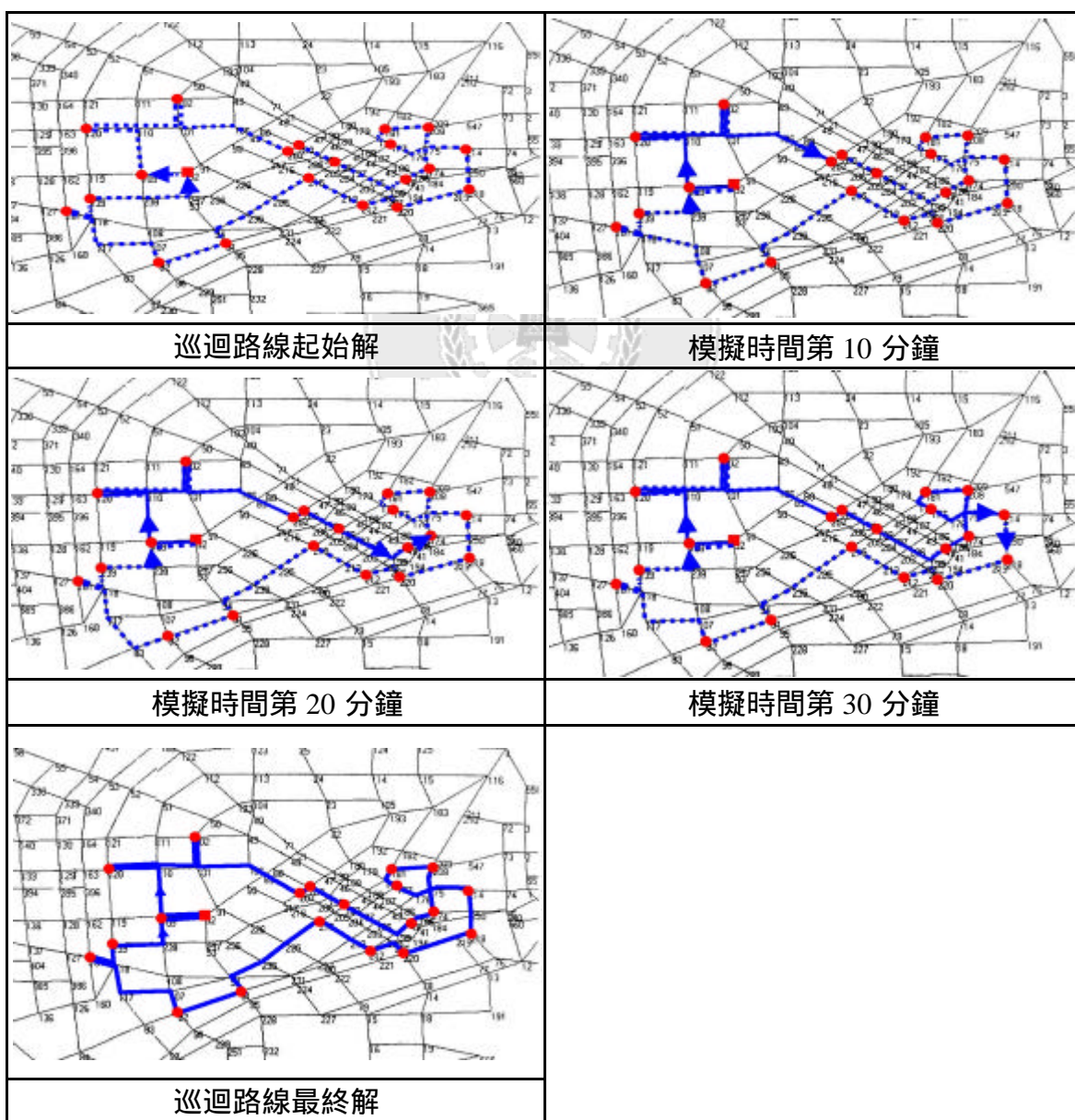


每 5 分鐘提供一次即時資訊

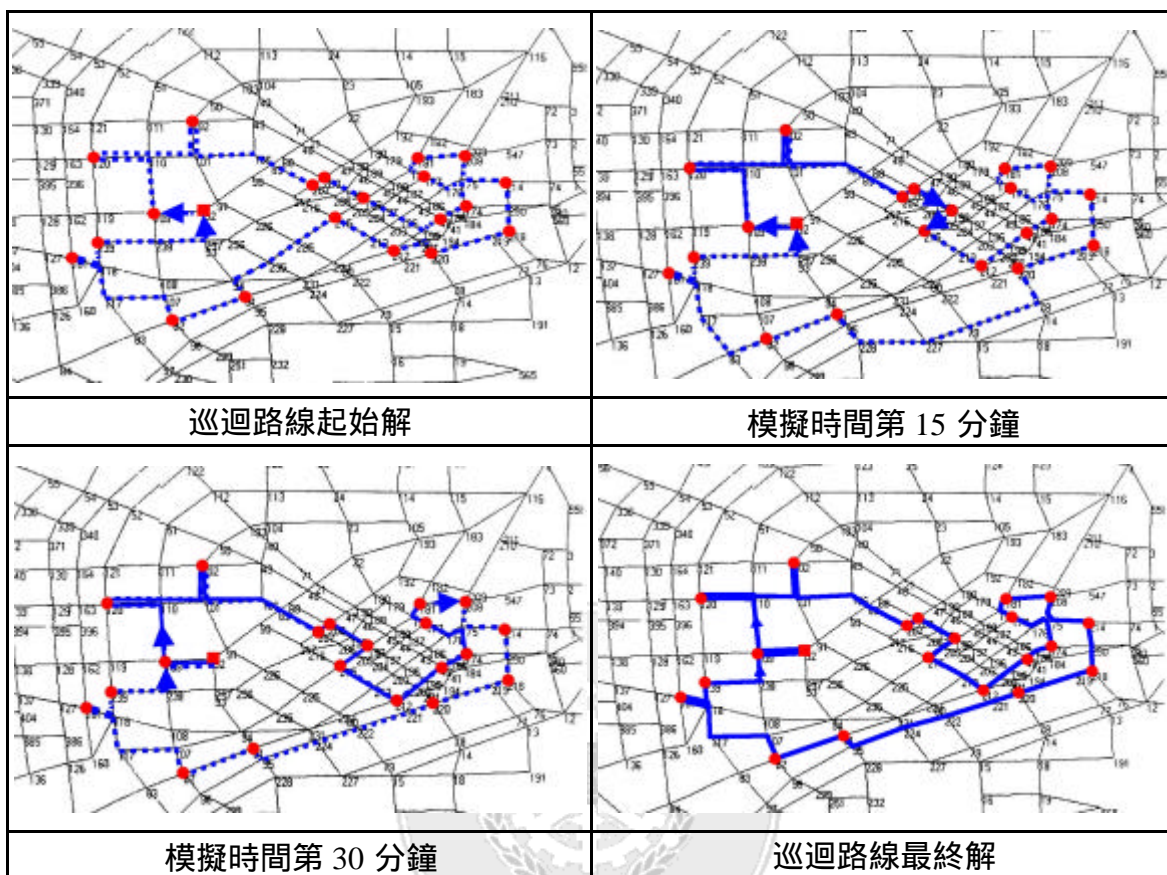




每 10 分鐘提供一次即時資訊

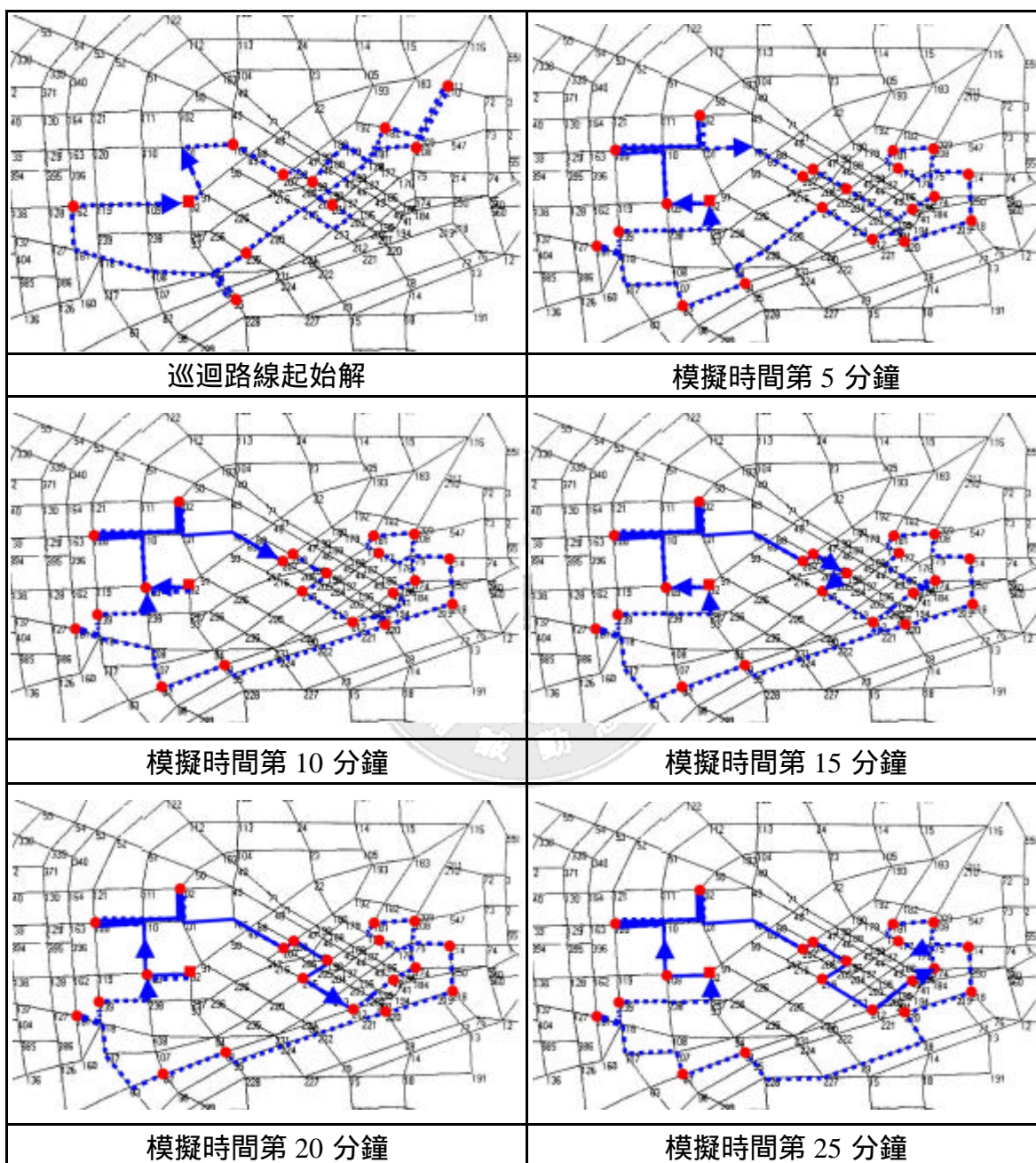


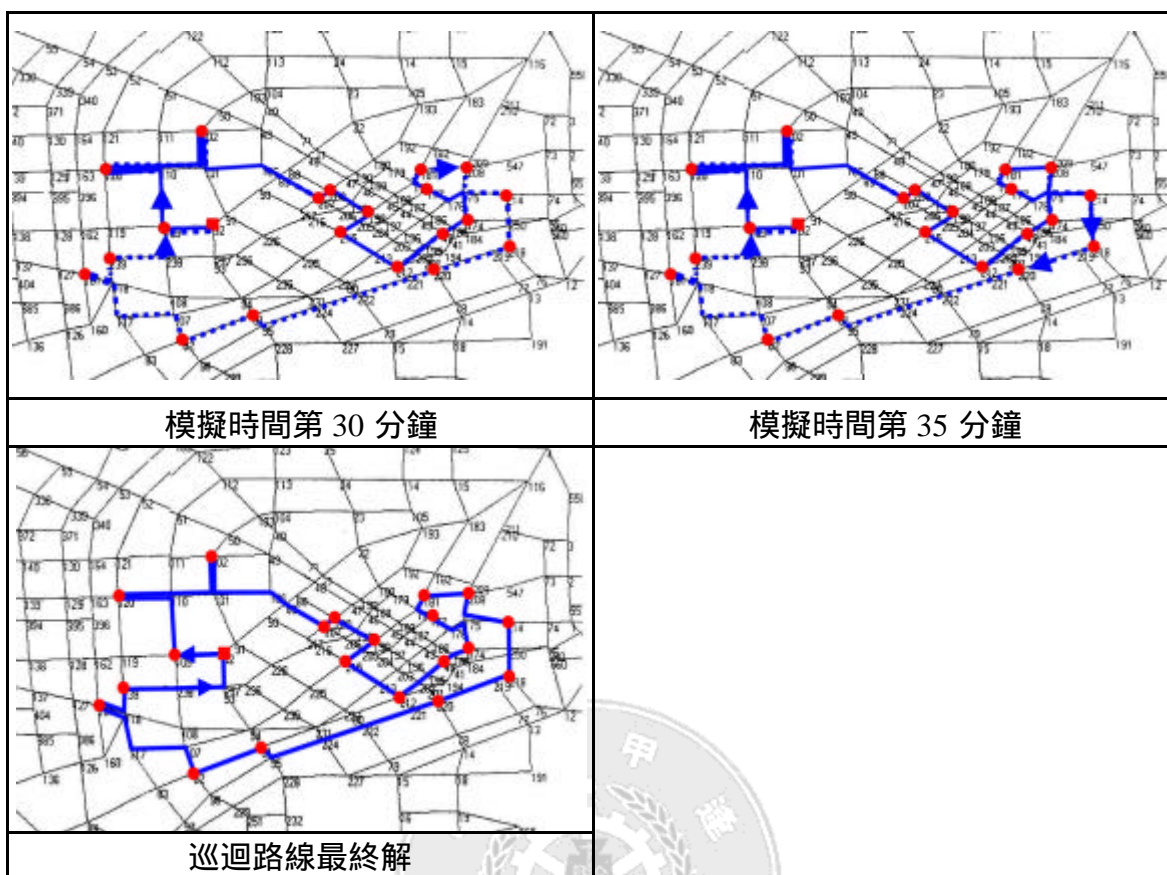
每 15 分鐘提供一次即時資訊



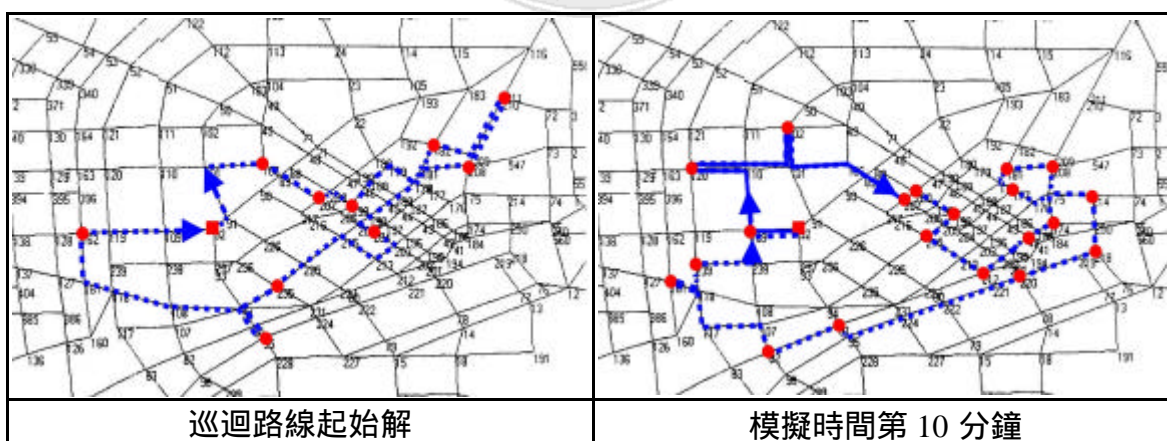
結合歷史資訊與即時資訊下動態需求點順序

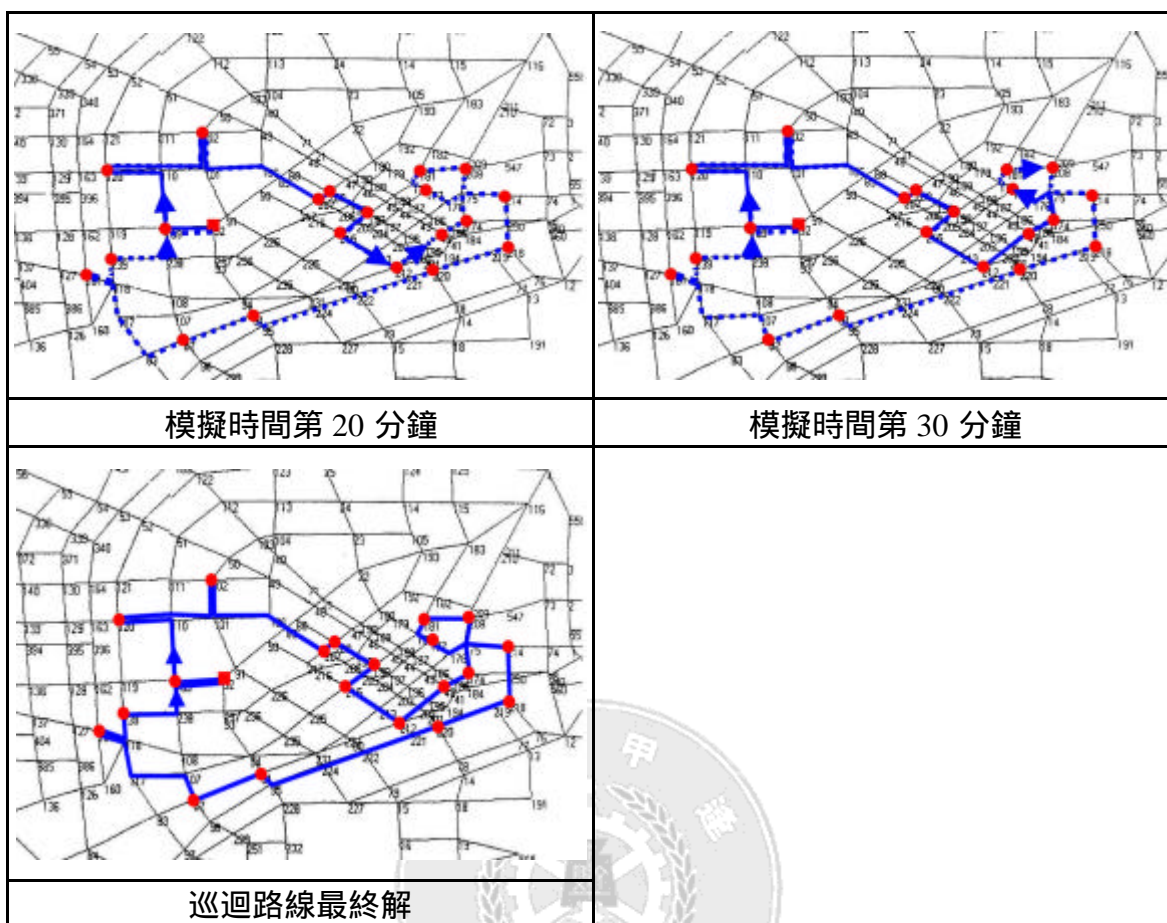
每 5 分鐘提供一次即時資訊



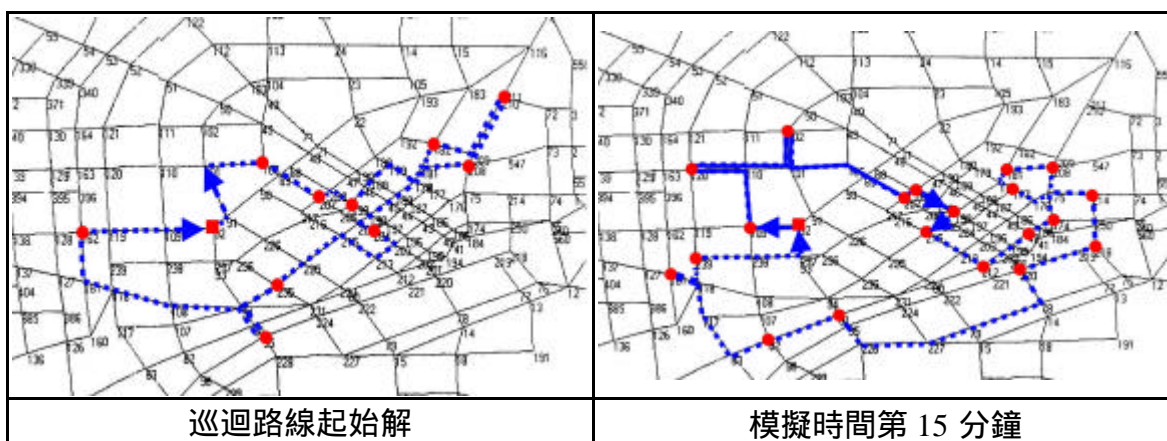


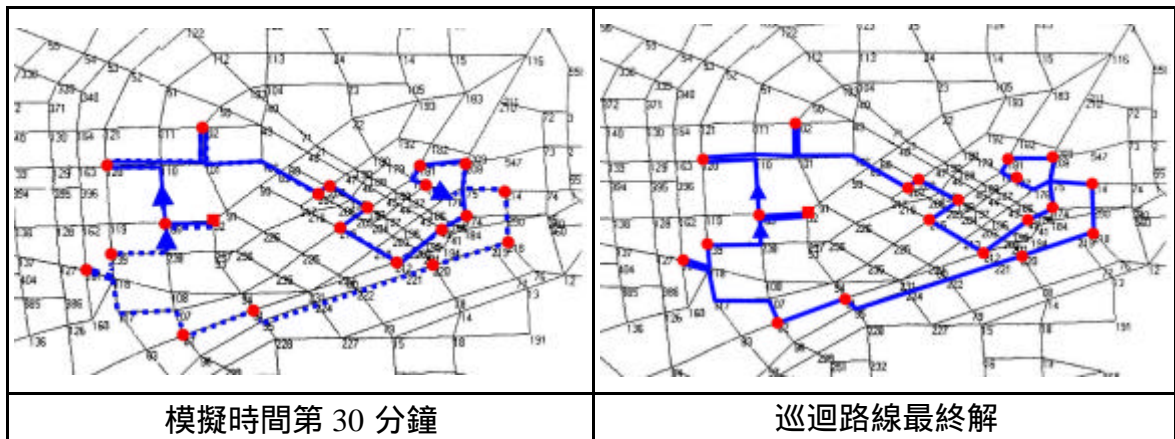
每 10 分鐘提供一次即時資訊





每 15 分鐘提供一次即時資訊

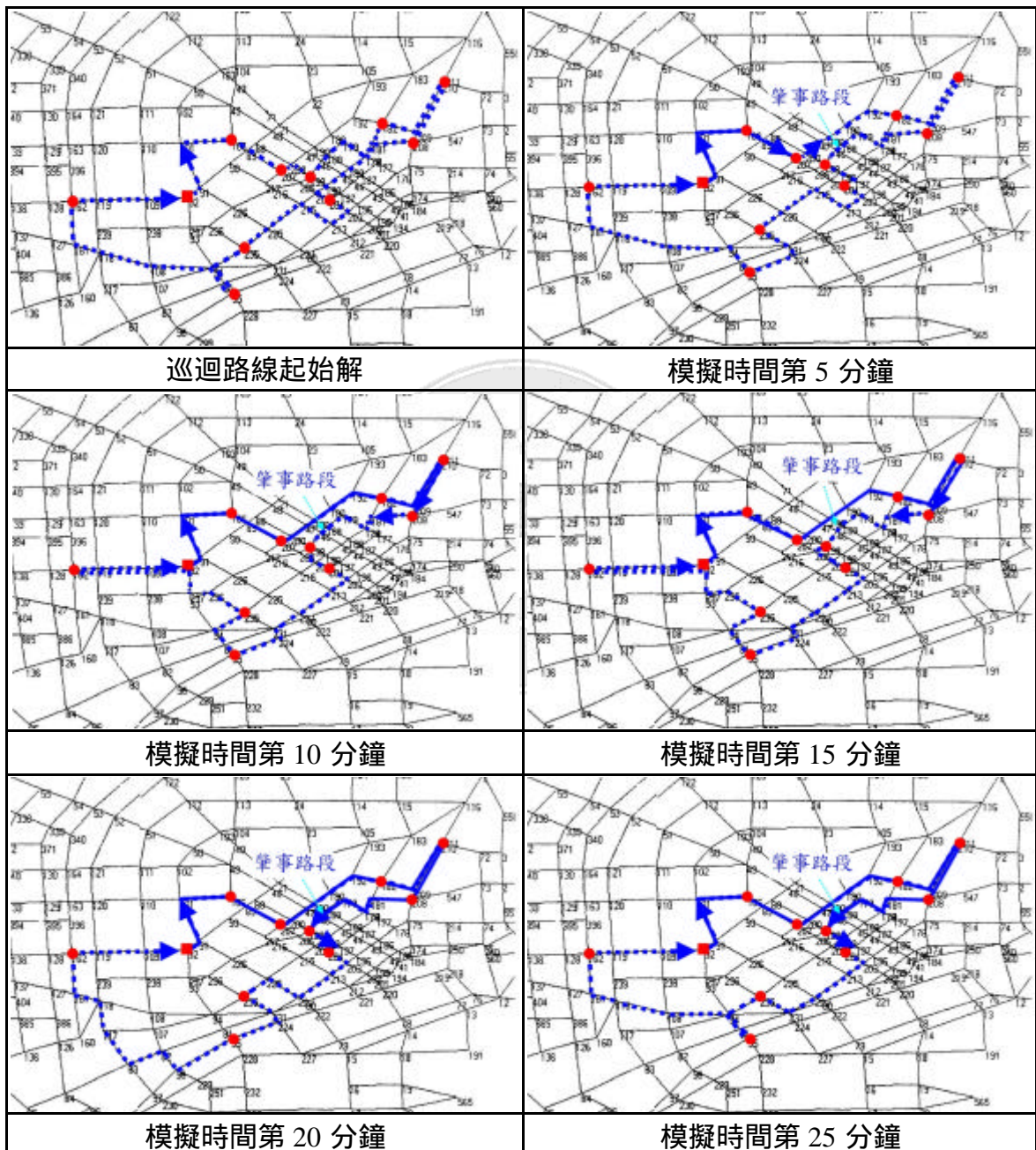


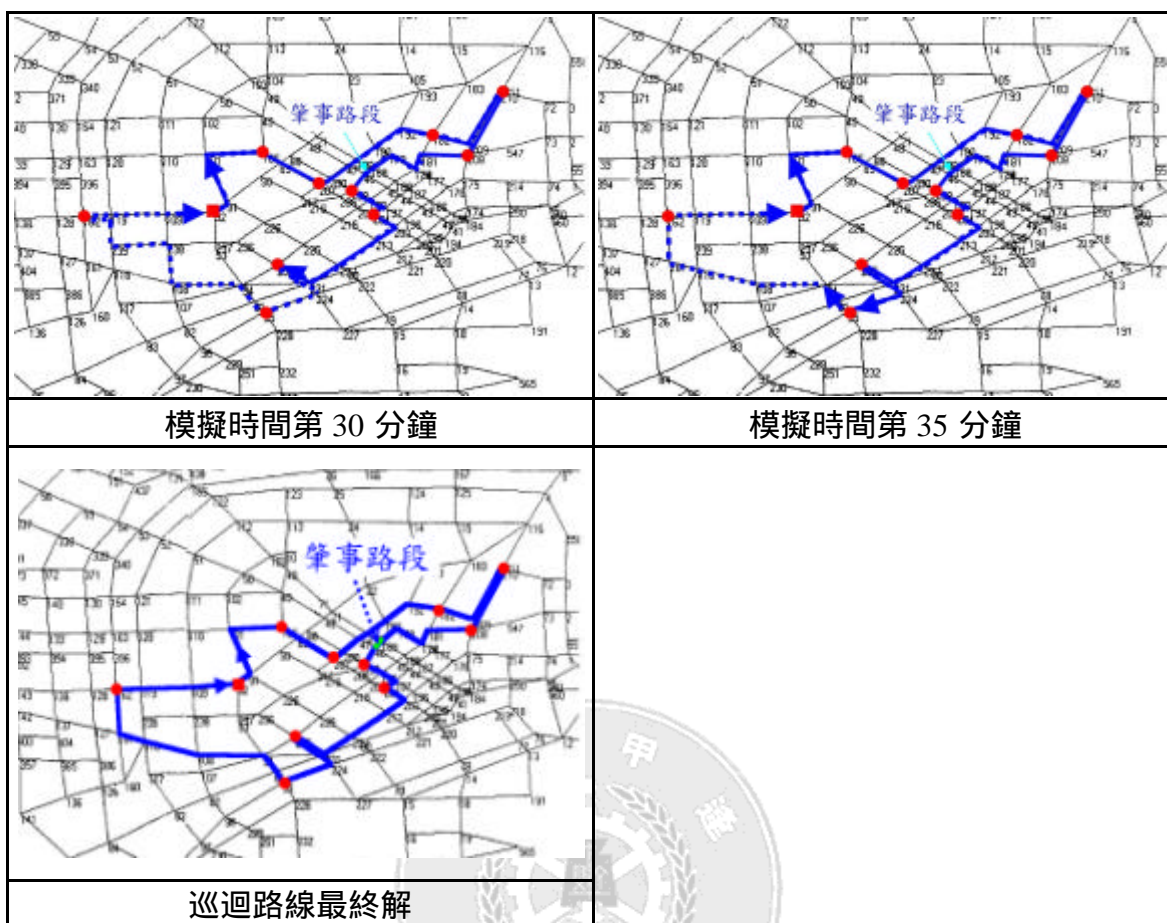


事故發生下動態需求點順序

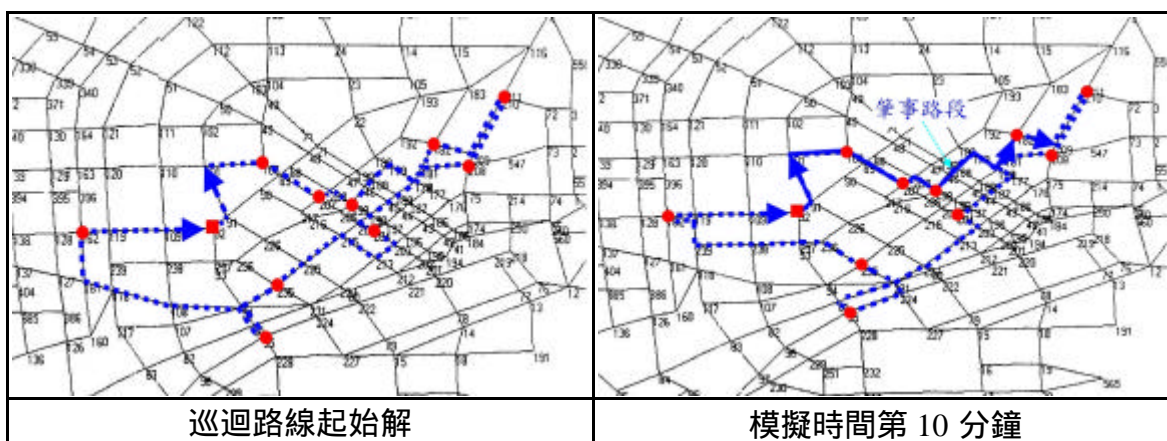
10 個需求點

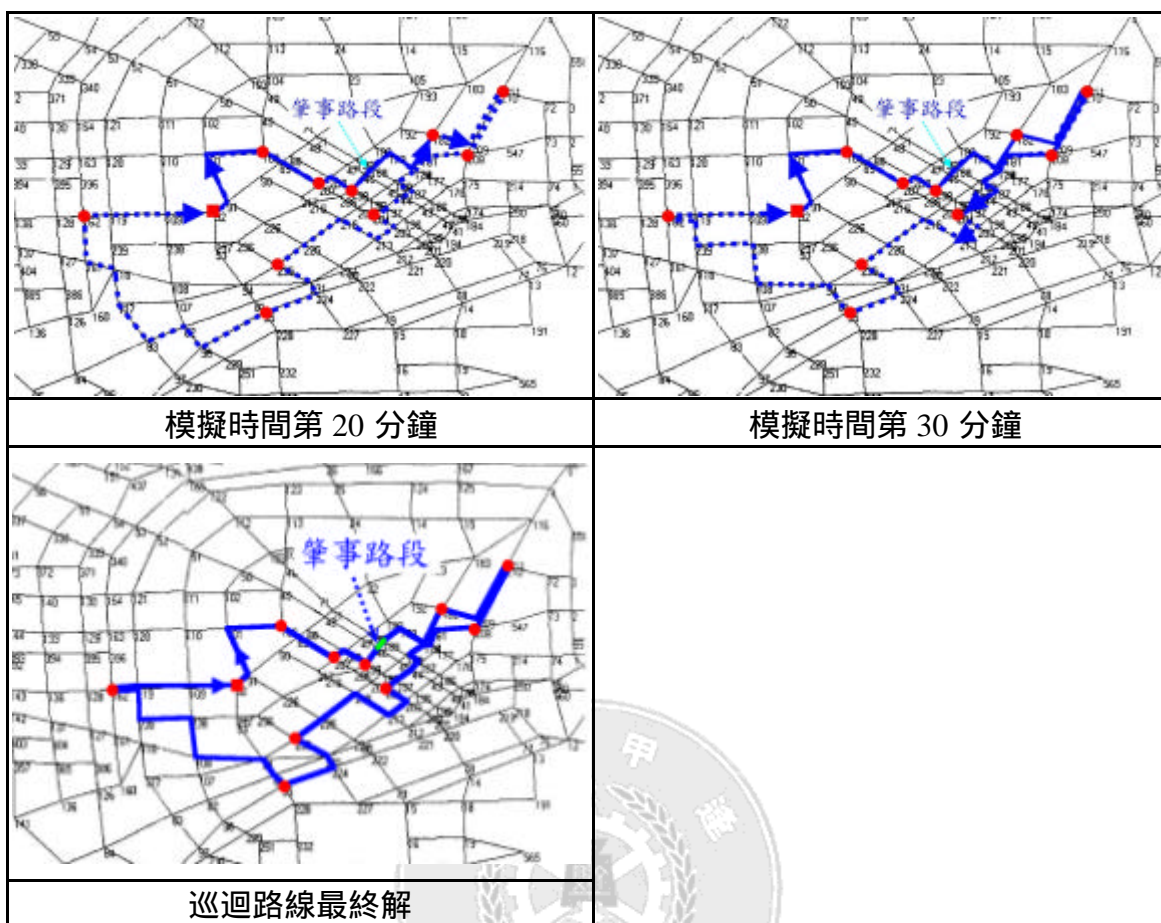
每 5 分鐘提供一次即時資訊



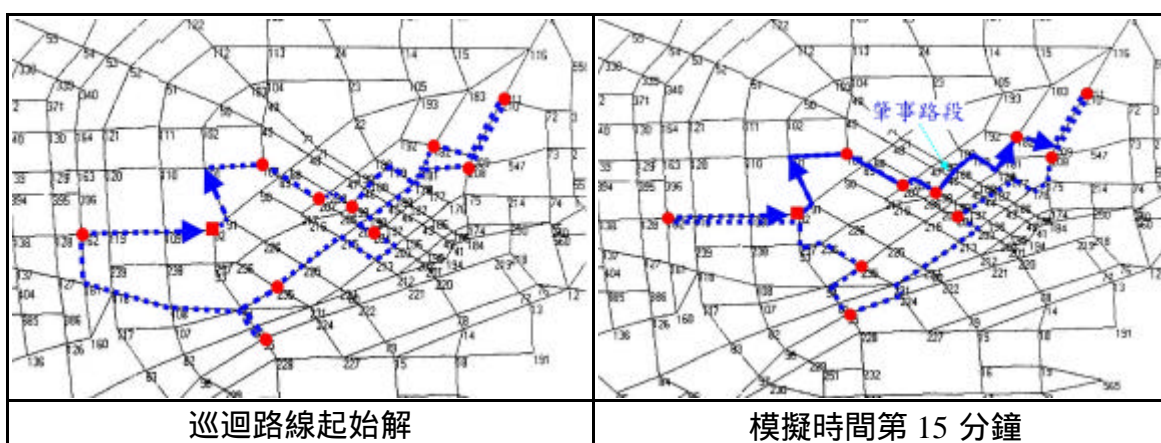


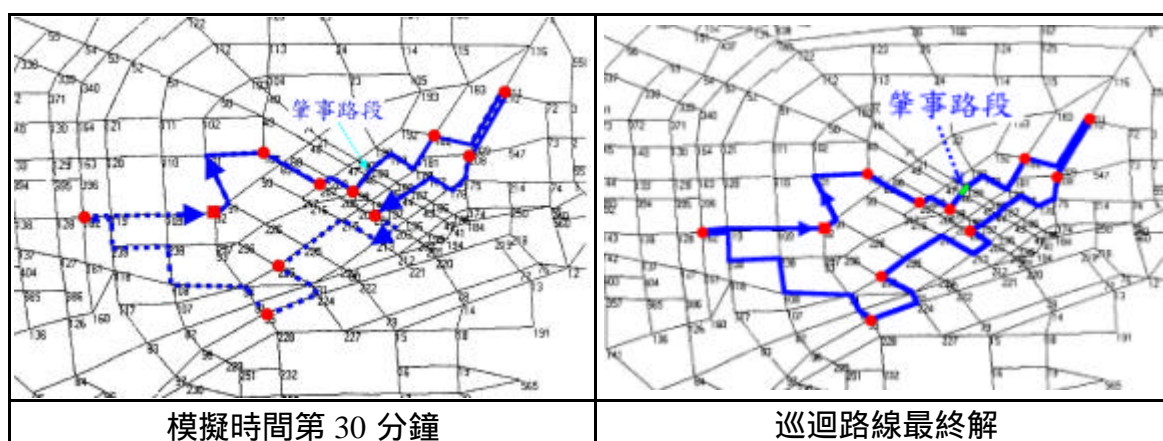
每 10 分鐘提供一次即時資訊





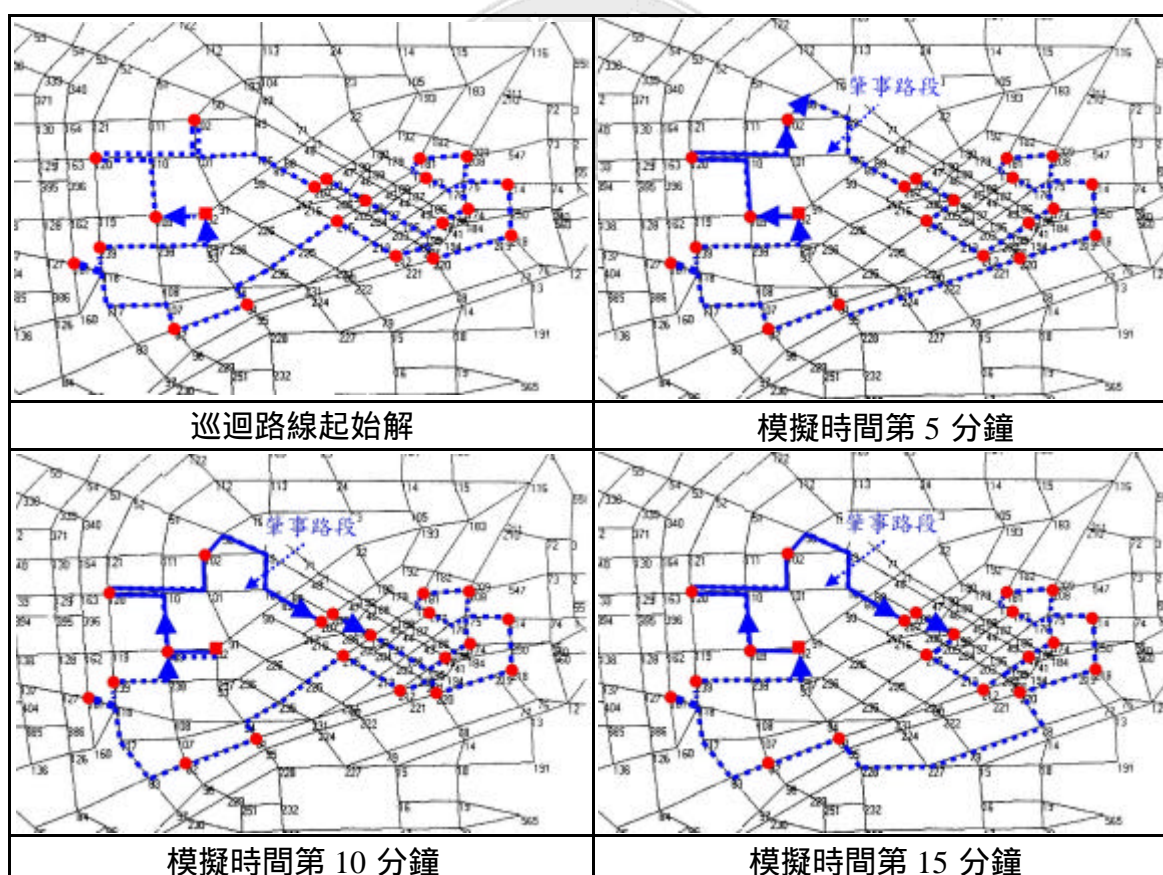
每 15 分鐘提供一次即時資訊

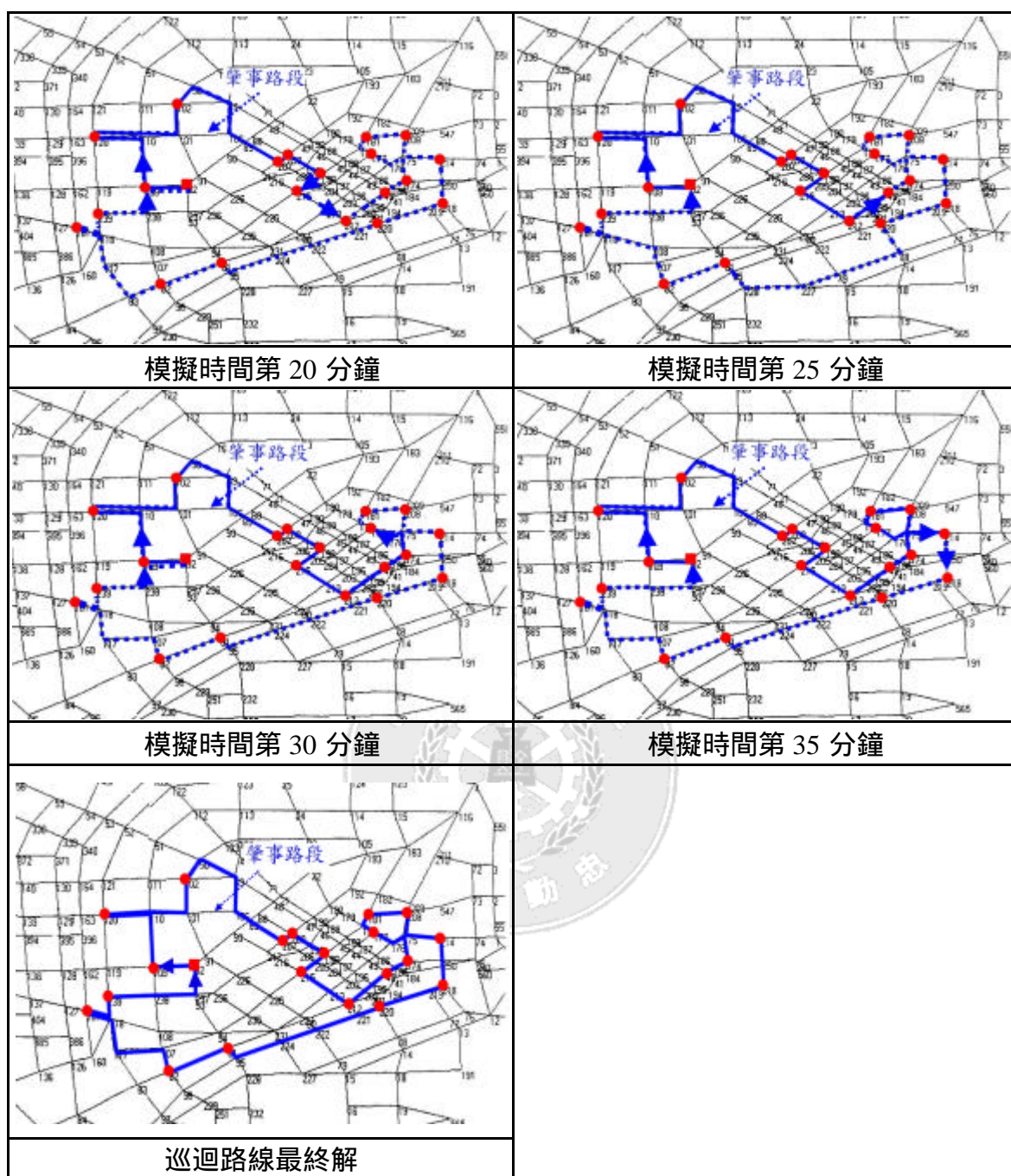




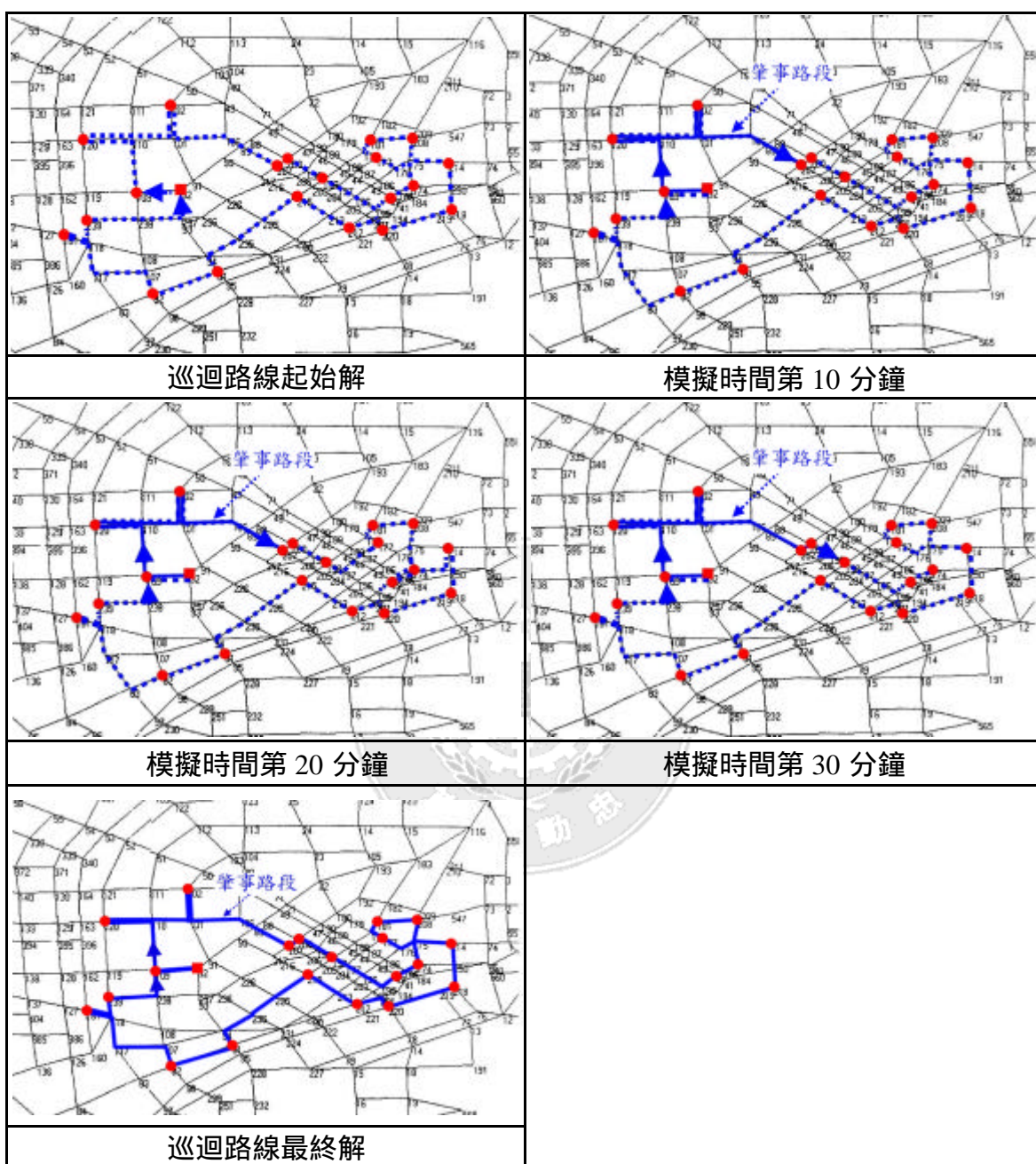
20 個需求點

每 5 分鐘提供一次即時資訊





每 10 分鐘提供一次即時資訊



每 15 分鐘提供一次即時資訊

