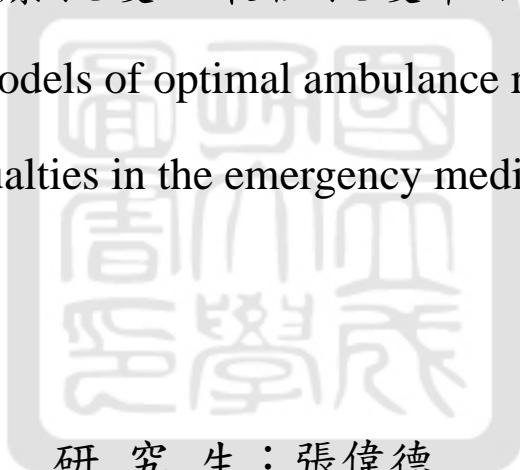


國立成功大學
工業與資訊管理研究所
碩士論文

大量傷患緊急醫療救護之最佳救護車派遣數學模式研究

Mathematical models of optimal ambulance redeployment for
shipping mass casualties in the emergency medical services system



研究生：張偉德

指導教授：王逸琳 博士

中華民國一百零五年九月

國立成功大學

碩士論文

大量傷患緊急醫療救護之最佳救護車派遣數學模式研究

Mathematical models of optimal ambulance
redeployment for shipping mass casualties in the
emergency medical services system

研究生：張偉德

本論文業經審查及口試合格特此證明

論文考試委員：王逸琳
張秀雲
蔡青志

指導教授：王逸琳

系(所)主管：李洪殿

中華民國 105 年 7 月 19 日

摘要

發生大規模災害後，可能會同時在多個地方發生事故，而每事故點若有大量傷患的話，如何有效率地使用有限的救護車資源，將各事故點的傷患儘快送至合適的醫院便成為一個重要議題。現今由緊急醫療救護系統之救護車指揮中心所調度的救護車派遣方式，通常是取決於決策者的自身經驗，其直覺的派遣決策缺乏整體性的評估，可能導致傷患等待過久而死亡。本研究擬探討緊急醫療救護系統面對多事故點的大量傷患之救護車派遣問題，以最小化傷患總等候時間為目標，決定哪台救護車應去哪個事故點將傷患送至哪家醫院的最佳派遣路徑。本研究依救護車之初始位置狀態，分別研究「已知初始發車處」及「可彈性調整初始發車處」等兩類救護車派遣問題。兩者皆假設各事故點位址及其傷患數量、各醫院位址等資訊皆已知，只是前者假設各救護車發車處為已知，著重在如何計算車輛途徑；而後者則假設各救護車初始發車處可彈性調整，除了車輛途徑決策外還必須處理初始的車輛配置方式（每個事故點應配置幾台）。

在求解已知初始發車處之救護車派遣問題時，我們先以時空網路圖為基礎發展一整數規劃模式，求解最小化傷患總等候時間的救護車最佳路徑，然而此種作法經常耗時甚久，因此我們提出以傷患找尋最近救護車的貪婪演算法；而為避免派遣決策過度集中處理單一事故點而忽略其它事故點傷患，我們亦提出另一考量「公平性」的派遣演算法來比較。可彈性調整初始發車處之救護車派遣問題主要是著眼於現實的派遣決策幾乎無法即時使用電腦求解，而較直覺的派遣決策通常會以事故點與醫院之責任劃分方式來指派救護車，因此該如何指定哪些醫院負責哪些事故點，再依此指派各醫院的初始救護車數，以最小化傷患總等候時間。

最後本研究針對兩問題中建構出 8 個與 24 個不同的 Case 進行測試，證實本研究之數學模式可有效地使傷患總等候時間最小化，而提出之演算法亦能在短時間內得到求解品質佳的解。

關鍵字：救護車派遣；緊急醫療救護；整數規劃；時空網路。

Mathematical models of optimal ambulance redeployment for shipping mass casualties in the emergency medical services system

Wade Chang

I-lin Wang

Department of Industrial and Information Management

SUMMARY

The aim of this study included developing mathematical models of optimal ambulance redeployment for shipping mass casualties in an emergency medical services system by using integer programming. Additionally, the objectives of the study involved minimizing the total waiting time for casualties. In the study, two ambulance redeployment problems, namely (1) Ambulance Dispatching Problem with the Initial Location provided, and (2) Ambulance Dispatching Problem with a Flexible initial Location, were investigated. It was assumed that the number of event points, hospitals, and patients were given for both problems. Although both problems required solving for optimal ambulance routings, the latter problem focused more on initial ambulance deployment. An integer programming model was formulated on a time space network for the first problem. However, the model was extremely time consuming. This was followed by developing a greedy heuristics based on a nearest first concept, in which the ambulance closest to a casualty was assigned to the casualty. In order to derive a fair waiting time, another heuristics that allowed each casualty to have a more equal waiting time was also employed. With respect to the second problem, certain properties of optimal redeployment were first derived for a hospital case, and then greedy heuristics were derived for other general cases. Unfortunately, a specific relationship between the optimal number of ambulances and the distance of the number of casualties to event points could not be derived from the results of the study.

Keyword : Ambulance redeployment, Emergency Medical Services, Integer Programming, Time space network

INTRODUCTION

The most important objective of decision makers in emergency medical services (EMS) involves saving human lives by making correct decisions to provide ambulances in the shortest possible time as a response to the arriving emergency calls. Decision makers determine the dispatches and the routing of ambulances based on past experiences. However, this is not effective in terms of an optimal dispatching strategy. Additionally, it does not involve any systematic calculations. In this study, a time space network was established, and an integer programming model was used to determine optimal dispatching strategies. Both ambulance response time and hospital arrival time were integrated into the total waiting time of patients. The objective of this study involved minimizing the total waiting time of casualties. Furthermore, an effective algorithm and a greedy algorithm were proposed to improve the model solving time.

MATERIALS AND METHODS

The problem was separated into two parts, namely (1) an Ambulance Dispatching Model with the Initial Location provided (which corresponded to a normal situation in which an ambulance starts from a station), and (2) an Ambulance Dispatching Model with a Flexible Location (in which infinity patients were simulated to find a flexible ambulance location). The first part involved formulating an integer programming model to solve an Ambulance Dispatching problem with the Initial Location provided based on a time space network (AD_{TS}^{IL}) to minimize the total waiting time of patients.

This was followed by two algorithms, denoted by GDA_d^{IL} and BDA^{IL} , which were proposed to improve the efficiency of the integer programming model. In the second part, two models, namely AD_{TS}^{FL} and AD_S^{FL} , were formulated from different viewpoints. Specifically, AD_{TS}^{FL} corresponded to Ambulance Dispatching with a Flexible Location based on a time Space network. This involved solving the problem by using integer programming based on a time space network with flexible locations given by hospitals instead of stations. Furthermore, AD_S^{FL} corresponded to Ambulance Dispatching with Flexible Location based on a dispatching sequence, which has an advantage as it was not affected by the distance between the hospitals

and the event points. The study discussed properties and validated scenarios ranging from a hospital to several event points, and used these to develop algorithms referred to as GDA_{f1}^{FL} and GDA_{f2}^{FL} .

The basic concept of algorithm GDA_d^{IL} in the first part involved shipping patients by using the closest ambulance, wherein the “d” at the right corner indicates “distance” and corresponded to the fact that patients called for ambulances based on the “distance” (i.e., closest ambulance is chosen by patients when patients are shipped given the occurrence of a mass casualty incident). The concept of algorithm BDA^{IL} involved balancing the number of patients at each event point, which in addition to providing a degree of fairness also accounted for the patients’ mood. In the second part of the study, the concept of GDA_{f1}^{FL} and GDA_{f2}^{FL} algorithms involved dispatching all ambulances to the same event point after locating all the ambulances at a given hospital, and the “f” denoted “a fleet of ambulances”. Both GDA_{f1}^{FL} and GDA_{f2}^{FL} were the same at the first dispatch, but differed at the next dispatch. The rule of algorithm GDA_{f1}^{FL} included the following: (1) dispatching ambulances to the closest event point at a hospital, (2) dispatching ambulances to the closest hospital at an event point. Algorithm GDA_{f2}^{FL} involved first sorting the event point, and then this order was followed when the ambulances were dispatched.

RESULT AND DISCUSSION

In the experiment, eight cases were generated with each case involving 50 different random tests to test the performances of AD_{TS}^{IL} , GDA_d^{IL} , and BDA^{IL} . Additionally, the experiment also generated 24 cases with each case involving 40 different random tests to test performances of AD_{TS}^{FL} , AD_s^{FL} , GDA_{f1}^{FL} , and GDA_{f2}^{FL} . All the experiments were conducted on a personal computer using Windows 7, 8GB RAM, and Intel Corei7. The results revealed that the AD_{TS}^{IL} model took a long time to solve all the cases. In contrast, algorithm GDA_d^{IL} could solve a case within 1 s and also indicated a good performance with a GAP less than 4%. Although algorithm BDA^{IL}

exhibited a poor performance with respect to the solving time, it exhibited an excellent performance with respect to the patient standard deviation of waiting time. The MAX standard deviation of the waiting time of BDA^{IL} was less than those of AD_{TS}^{IL} and GDA_d^{IL} . Therefore, a good solution could be obtained in a short time by algorithm GDA_d^{IL} , and the BDA^{IL} algorithm effectively balanced the patients at each event point. The AD_S^{FL} model was better than the AD_{TS}^{FL} model because the AD_S^{FL} model in addition to obtaining an optimal solution, could also obtain the solution in a short time. Algorithms GDA_{f1}^{FL} and GDA_{f2}^{FL} could obtain optimal solutions in situations that involved only a single hospital, and the gap increased as the number of hospitals increased.

CONCLUSION

The problem of ambulance redeployment for shipping mass casualties in an emergency medical services system is very important since it is related to the probability of patient survival. A literature survey in the present study indicated that previous researches only focused on the impact of factors including “response time” or “arrival time” on patient survival probability. This study proposed integer programming models based on a time space network to minimize the total waiting time (that included the total of response time and arrival time). Moreover, algorithms were constructed to improve solving effectiveness. In the first part of the model, the performances of AD_{TS}^{IL} , GDA_d^{IL} , and BDA^{IL} were tested. Algorithm GDA_d^{IL} exhibited a good performance and also obtained a solution in a short time. Algorithm BDA^{IL} performs well with respect to the patient standard deviation of waiting time. In the second part, the performances of AD_S^{FL} , AD_{TS}^{FL} , GDA_{f1}^{FL} , and GDA_{f2}^{FL} were tested. The time taken by model AD_S^{FL} to obtain a solution was shorter than that taken by model AD_{TS}^{FL} . This was because model AD_S^{FL} was not affected by the distance between the hospital and the event point. Algorithms GDA_{f1}^{FL} and GDA_{f2}^{FL} performed well in a situation involving a single hospital but were not suitable for situations involving more than one hospital.

誌謝

這一頁，是最輕鬆的一頁，也是最辛苦的一頁。

輕鬆的是，這一頁所有的內容已經不需要再與老師咪聽、討論，也不需要寫程式來證明，更不需要畫圖講解了；但辛苦的是，為了這一頁，我足足花費了一千多個日子才能開始落筆書寫。

一千多個日子以來，最感謝的還是逸琳老師，如果沒有老師的幫助，這篇論文絕對不可能誕生，儘管時常達不到老師的要求，但在緊要關頭，老師依然會提供適時的協助，老師就像一個站在懸崖邊伸手相救的人，拯救那個只剩一隻手抓在懸崖，快掉下深谷的我。感謝老師在程式、數學模型、論文修改、PPT 製作與做事態度上的指導，沒有老師就沒有現在這篇論文，沒有老師就沒有現在的我。

一千多個日子中的前三分之二，還記得當初與琮閔一起為了趕報告去吃勝早看日出，感謝琮閔經常在弄完自己的東西後，會來協助其他人，不管是報告或是 PPT 的製作上；另外還有熱心助人的超級里長伯「阿百」(妹子王)在碩零、碩一、碩二的各個時期一起看 Lab 日出，感謝阿百對待他人的熱忱與付出，LAB 大小事情交給你就對了。新進的學弟妹，小秉、Tyra、貞泰(元忠)不但為 LAB 增添了許多熱鬧的色彩，同時也幫助了我許多，感謝小秉和貞泰在寫程式和寫網頁上的協助，感謝 Tyra 成為史上第一位被我的魔術嚇到叫出來後，還不斷慇懃我去嚇別人的觀眾。後來進 LAB 的富元，你不但永遠在臉上保持著招牌笑容，散播歡愉的氣氛，而且還為 LAB 準備了多款桌遊，讓大家能夠一起在 LAB 同樂，請容我尊稱你一聲「桌遊王」！感謝你為大家提供了一個排解煩憂苦悶生活的方法。最後要感謝的是我的家人，媽媽、爸爸、哥哥，感謝你們讓我毫無後顧之憂的研究碩士，感謝你們時常關心我的論文進度、身體健康，讓我知道，家永遠是我最後的避風港，家永遠能讓我的身心靈得到最充分的休息，使我蓄積更大的能量。

光陰似箭，歲月如梭。離開成大的生活也不遠了，這意味著我即將由學生身分邁入下一個階段。期許自己能夠運用所學，回饋社會、回饋世界，讓世界變得更好，讓每個人變得更好，讓自己變得更好。

目錄

摘要.....	I
ABSTRACT.....	II
誌謝.....	VI
目錄.....	VII
表目錄.....	XI
圖目錄.....	XII
第一章 緒論.....	1
1.1 研究背景.....	1
1.2 研究動機與目的.....	3
1.3 研究問題.....	4
1.4 論文架構.....	5
第二章 文獻探討.....	6
2.1 緊急醫療救護系統與大量傷患事件.....	6
2.1.1 緊急醫療救護服務系統.....	6
2.1.2 大量傷患事件.....	8
2.2 救護車派遣與緊急醫療救護系統佈署問題.....	10

2.2.1 救護車基地佈署問題.....	10
2.2.1.1 確定性靜態佈署模式.....	10
2.2.1.2 隨機性靜態佈署模式.....	14
2.2.1.3 動態佈署模式.....	16
2.3 救護車輛派遣問題.....	17
2.4 小結.....	21
 第三章 已知初始發車處之救護車派遣問題.....	22
3.1 問題描述與相關假設.....	22
3.1.1 問題描述.....	22
3.1.2 問題假設.....	25
3.2 數學模式.....	25
3.2.1 已知初始發車處之救護車派遣之時空網路圖模式.....	25
3.3 貪婪演算法與傷患平衡之救護車派遣法.....	30
3.3.1 GDA_d^L 演算法介紹.....	30
3.3.2 BDA^L 演算法介紹	34
3.3.3 小結.....	36
3.4 數值分析.....	36
3.4.1 數據產生方式.....	36

3.4.2 數值測試.....	38
3.4.3 小結.....	42
第四章 可彈性調整初始發車處之救護車派遣問題.....	44
4.1 問題描述與假設.....	45
4.1.1 問題描述.....	45
4.1.2 問題假設.....	45
4.2 時空網路圖之救護車派遣整數規劃模式(AD_{TS}^{FL})	46
4.2.1 參數與變數定義.....	46
4.2.2 模式說明.....	47
4.2.3 範例說明.....	49
4.3 順序式救護車派遣之整數規劃模式(AD_S^{FL})	52
4.3.1 基本概念.....	52
4.3.2 參數與變數定義.....	53
4.3.3 順序式模式說明.....	54
4.3.4 範例說明.....	55
4.4 貪婪演算法.....	58
4.4.1 基本想法.....	58
4.4.2 GDA_{f1}^{FL} 、 GDA_{f2}^{FL} 演算法說明	60

4.5 數值分析.....	66
4.5.1 數據產生方式.....	66
4.5.2 數值測試.....	67
4.5.3 小結.....	74
第五章 結論與未來方向.....	76
5.1 研究結論.....	76
5.1.1 已知初始發車處之救護車派遣問題.....	76
5.1.2 可彈性調整初始發車處之救護車派遣問題.....	77
5.2 未來研究方向.....	77
參考文獻.....	81

表目錄

表 1.1 台灣各地救護車分配數量表.....	2
表 1.2 反應時間與死亡率統計.....	3
表 2.1 派遣順序清單.....	15
表 3.1 變數索引與集合說明表.....	26
表 3.2 參數說明表.....	26
表 3.3 變數定義表.....	27
表 3.4 測試 Case 分類表.....	38
表 3.5 求解時間與 Gap 比較表格.....	39
表 3.6 標準化標準差統計表格.....	42
表 4.1 變數索引與集合說明.....	46
表 4.2 參數說明.....	46
表 4.3 變數說明.....	47
表 4.4 集合與索引介紹.....	53
表 4.5 參數說明.....	53
表 4.6 變數說明.....	54
表 4.7 符號說明表.....	59
表 4.8 測試 Case 分類表(一).....	66
表 4.9 測試 Case 分類表(二).....	67
表 4.10 平均求解時間 (單位：秒).....	69
表 4.11 演算法與修改模式 Gap 比較表.....	70
表 4.12 救護車放置規則測試 Case 表.....	71
表 4.13 護車放置數量表(一).....	72
表 4.14 護車放置數量表(二).....	73
表 4.15 護車放置數量表(三).....	74

圖目錄

圖 2.1 緊急醫療救護流程圖	8
圖 2.2 佈署問題示意圖	11
圖 2.3 DSM 示意圖	13
圖 2.4 部分覆蓋模式示意圖	14
圖 2.5 救護車重新配置示意圖	16
圖 2.6 醫院緊急疏散示意圖	18
圖 2.7 避難醫院與疏散醫院	19
圖 2.8 疏散醫院時間擴張網路圖 (Time-expanded Network)	19
圖 3.1 事故點與各層救護車關係示意圖	23
圖 3.2 醫院與各層救護車關係示意圖	23
圖 3.3 單一救護車時空網路圖	24
圖 3.4 傷患流量網路圖	24
圖 3.5 GDA_d^{IL} 演算步驟流程圖	31
圖 3.6 範例說明示意圖	32
圖 3.7 GDA_d^{IL} 演算流程圖(一)	33
圖 3.8 GDA_d^{IL} 演算流程圖(二)	33
圖 3.9 GDA_d^{IL} 演算流程圖(三)	33
圖 3.10 GDA_d^{IL} 演算流程圖(四)	33
圖 3.11 GDA_d^{IL} 演算流程圖(五)	33
圖 3.12 GDA_d^{IL} 演算流程圖(六)	33
圖 3.13 GDA_d^{IL} 演算流程圖(七)	33
圖 3.14 GDA_d^{IL} 演算流程圖(八)	33
圖 3.15 BDA^{IL} 演算步驟流程圖	34
圖 3.16 BDA^{IL} 演算流程圖(一)	35
圖 3.17 BDA^{IL} 演算流程圖(二)	35
圖 3.18 BDA^{IL} 演算流程圖(三)	35

圖 3.19 BDA^{IL} 演算流程圖(四).....	35
圖 3.20 BDA^{IL} 演算流程圖(五).....	36
圖 3.21 BDA^{IL} 演算流程圖(六).....	36
圖 3.22 BDA^{IL} 演算流程圖(七).....	36
圖 3.23 BDA^{IL} 演算流程圖(八).....	36
圖 3.24 座標圖示意圖.....	37
圖 3.25 Case3 盒鬚圖.....	40
圖 3.26 Case6 盒鬚圖.....	40
圖 3.27 標準化傷患等候時間標準差比較圖.....	42
圖 4.1 範例說明圖.....	49
圖 4.2 救護車路徑網路連接圖(一).....	50
圖 4.3 救護車路徑網路連接圖(二).....	50
圖 4.4 單一救護車網路圖.....	51
圖 4.5 傷患流向網路圖.....	51
圖 4.6 範例說明示意圖.....	56
圖 4.7 去程指派表.....	56
圖 4.8 回程指派表.....	56
圖 4.9 傷患抵達時刻記錄圖.....	57
圖 4.10 基本想法範例圖.....	58
圖 4.11 傷患總等候時間函數圖形.....	60
圖 4.12 GDA_{f1}^{FL} 演算步驟流程圖.....	61
 圖 4.13 GDA_{f1}^{FL} 範例說明示意圖.....	62
 圖 4.14 GDA_{f1}^{FL} 演算流程圖(一).....	62
 圖 4.15 GDA_{f1}^{FL} 演算流程圖(二).....	62
 圖 4.16 GDA_{f1}^{FL} 演算流程圖(三).....	62
 圖 4.17 GDA_{f1}^{FL} 演算流程圖(四).....	62
 圖 4.18 GDA_{f1}^{FL} 演算流程圖(五).....	63
 圖 4.19 GDA_{f1}^{FL} 演算流程圖(六).....	63
 圖 4.20 GDA_{f1}^{FL} 演算流程圖(七).....	63

圖 4.21 $GDA_{f_2}^{FL}$ 演算步驟流程圖 64

圖 4.22 $GDA_{f_2}^{FL}$ 演算流程圖(一) 65

圖 4.23 $GDA_{f_2}^{FL}$ 演算流程圖(二) 65

圖 4.24 $GDA_{f_2}^{FL}$ 演算流程圖(三) 65

圖 4.25 $GDA_{f_2}^{FL}$ 演算流程圖(四) 65

圖 4.26 $GDA_{f_2}^{FL}$ 演算流程圖(五) 65

圖 4.27 $GDA_{f_2}^{FL}$ 演算流程圖(六) 65

圖 4.28 $GDA_{f_2}^{FL}$ 演算流程圖(七) 65



第一章

緒論

1.1 研究背景

近年來，影響人們生活的大量傷患事件不斷發生，如：高雄氣爆事件、八仙塵爆事件、美國 911 恐怖攻擊事件以及區域大且嚴重的地震事件，或是影響區域較為局部的災害如：火災、土石流、重大車禍等災害事件都造成了人民生命財產上嚴重的損失。而地理位置特殊的台灣，重大的天然災害如颱風或是地震更是沒有少過，有時候甚至連救災人員都無法自保。因此，大量傷患事件發生時，緊急醫療服務系統(Emergency Medical System, EMS)該如何提供有效的救災決策便顯得相當關鍵。

EMS 中的救災任務可分為檢傷、治療以及後送，「檢傷」為對現場傷患依照傷重程度進行分類的稱呼，並依照此分類排定優先順序；「治療」為現場救護人員提供簡單的醫療處置，減緩傷患的疼痛或暫時維持傷患的生命；「後送」即為救護車依照傷患順序將傷患送至合適的醫療院所。過往發生重大災難事故時，EMS 成立緊急的應變指揮中心，處理及分配不同的救災任務，而指揮中心在派遣救護車後送傷患或是派遣救護車至事故點時，經常是依照決策者過去的經驗派遣救護車，但是決策者的救護車派遣經驗往往缺乏有系統的最佳化分析，因此雖然決策可行，但並不保證是最有效率的救護車派遣決策。

目前台灣的緊急醫療系統中的救護車分屬下述單位：消防站、醫院、衛生機關與民營救護車公司，而各機構的救護車使用方式也不盡相同，一般而言，醫院、衛生機關與民營救護車公司的救護車通常負責一般慢性傷病患的轉院與轉診，並不負責緊急的醫療事故，因此醫院、衛生機關與民營救護車公司所擁有的救護車數量並不多，且僅具基本設備；而台灣大部分的救護車皆是由台灣各地的消防機構所擁有，負責緊急事故的救援，例如突發的車禍事件或是偶發的火災事件，緊

急事故的救援除了救護車以外，同時也需要擁有急救證照的急救人員配置於救護車上，因此發生重大事故時，能夠使用的救護車資源僅有消防隊本身擁有的救護車。表 1.1 為台灣各縣市的救護車數量表，以臺南市為例，救護車在消防機關的比例佔臺南市所有救護車中的 60% 以上。

表 1.1 台灣各地救護車分配數量表

縣市別	合計	衛生機關	消防機關	醫院		衛生所	救護車營業機構	其它
				公立	私立			
新北市	226	-	130	5	14	11	50	16
臺北市	164	-	74	24	16	-	44	6
桃園市	176	-	129	2	2	-	30	12
臺中市	209	6	113	8	29	5	36	9
臺南市	138	1	87	8	4	-	27	11
高雄市	288	-	178	14	8	-	60	25
宜蘭縣	59	4	26	6	6	2	13	2
新竹縣	53	1	30	-	-	3	16	2
苗栗縣	55	3	29	-	6	-	12	5
彰化縣	90	1	53	1	27	-	-	7
南投縣	72	1	29	4	13	15	10	-
雲林縣	72	-	36	1	3	9	17	6
嘉義縣	53	1	31	3	-	7	10	1
屏東縣	107	4	54	3	21	13	7	3
臺東縣	66	1	28	6	4	11	7	9
花蓮縣	92	2	35	12	8	9	18	8
澎湖縣	23	1	16	3	1	-	-	2
基隆市	33	1	27	3	-	-	-	1
新竹市	18	2	11	1	-	-	-	4
嘉義市	36	-	18	2	7	2	6	-
金門縣	18	-	10	4	-	-	-	4
連江縣	11	-	10	1	-	-	-	-

資料來源：(衛生福利部統計處, 民 103)

1.2 研究動機與目的

長久以來，多事故點、多傷患的緊急大量傷患事件之救護車派遣與送醫問題，皆是由決策者的自身經驗來決定救護車的派遣路徑，而在大量傷患與大量事故點的情況下，若僅僅憑著決策者自身經驗指派救護車，恐怕不容易提升救護車派遣的效率，更有甚者，指派救護車時若是缺乏有系統地最佳化指派，將會使得傷患的等候時間大幅增加，而 Larsen et al. (1993) 認為在大量傷患事件發生後，傷患的病情將隨時間的增加而持續惡化，此外，Blackwell and Kaufman (2002) 則針對所有到院前緊急醫療救護的傷病患，收集美國南卡羅萊納州 Mecklenburg County 的 5424 筆資料進行統計，傷患的死亡率與反應時間(救護車抵達事故點的時間)的統計資料如表 1.2 所示，可以發現當救護車的反應時間增加時，傷病患的死亡率也隨之增高。除了反應時間對傷患死亡率有影響外，Inoue et al. (2006) 亦提出到院時間與存活率的關係，研究中發現傷患的到院時間越長，傷患的死亡率將隨之增高。因此，本研究將綜合救護車派遣問題中的反應時間與到院時間，以最小化傷患的總等候時間當作目標值，找出救護車的最佳派遣路徑。

表 1.2 反應時間與死亡率統計

反應時間(分)	死亡率(%)	死亡人數	樣本數
0~1	0	0	47
1~2	0	0	70
2~3	0.57	1	176
3~4	0.48	2	419
4~5	0.6	4	669
5~6	1.82	16	879
6~7	1.52	14	921
7~8	1.57	11	701
8~9	1.92	10	522
9~10	2.02	7	347
10~11	1.34	3	224
11~12	2.16	3	139
12 以上	0	0	310

資料來源：(Blackwell & Kaufman, 2002)

本研究將探討如何在多個事故點、多個消防站、多台救護車、多名傷患與多間醫院的情況下，並將消防站、事故點與醫院彼此的行車時間以及救護車在事故點與醫院的處理時間列入考慮後，提供正確且有效率的救護車派遣決策，期望在有限的資源下，將資源做最妥善的利用，並最小化傷患的總等候時間。另外，本第二部分的研究不再假設救護車固定從消防站出發，而是假設救護車的放置處可於各間醫院彈性調整，藉此模擬無限多個傷患的救護車派遣情境，期望找出救護車的數量與醫院、事故點的配對關係與配置比例。

1.3 研究問題

本研究針對大量傷患事件之緊急救護車派遣的相關問題分為已知初始發車處之派遣問題與可彈性調整初始發車處之救護車派遣問題兩個部分：

1. 已知初始發車處之救護車派遣問題：

大量傷患事件發生時，緊急應變指揮中心會將救護車由消防站派遣至合適的事故點，並將傷患送往合適的醫院，若此時事故點還有傷患，則指揮中心將繼續指派救護車至事故點，直到傷患全數送至醫院，救護車才結束派遣任務，回到原來的消防站待命，在此問題中，救護車的數目以及救護車的初始位置都是已知的且固定的。

本研究在已知初始發車處之救護車派遣問題中，考慮在已知數量的消防站、已知數量的救護車、已知數量的事故點、已知各個事故點的傷患數目與已知數量的醫院下，欲決定救護車在各個時刻下派遣的事故點與載送的傷患，並將傷患送至最適當的醫院，使得傷患總等候時間最小。而本問題除了以整數規劃的數學模式求解以外，我們亦發展出提升求解效率的相關演算法。

2. 可彈性調整初始發車處之救護車派遣問題：

當大量傷患事件發生時，假設救護車除了一開始由消防站出發，以及結束時回到消防站之外，其餘的時間皆是在事故點與醫院之間來回地載送傷患。而當傷患數量越多時，消防站的位置對救護車派遣所帶來的影響會越來越小；

當傷患數量達到無限多個時，救護車來回的事故點與醫院將趨於穩定，此時可以發現救護車、事故點與醫院彼此間的配對關係與分群組合，舉例來說，派遣路徑中可能會看到某台救護車只在某幾間醫院與某幾個事故點之間來回，並不會派遣至其它的事故點與醫院，而其它的事故點與醫院將會交給其它合適的救護車負責。但實務上不可能有無限多個傷患，因此本研究假設救護車的初始發車數並非消防站，而是由醫院出發，其中救護車的初始發車處可彈性調整，藉此模擬傷患無限多的情境，以求得救護車、事故點與醫院彼此間的配對關係與分群組合，期望提供決策者更有效率的救護車派遣機制。

本問題取消了救護車初始位置固定於消防站的限制，並將救護車改成由醫院出發，且救護車放置的醫院與放置數量可由決策者彈性調整。在已知醫院位置、醫院數量、醫院距離、事故點距離、事故點數量與傷患數量的情況下，本研究欲決定最佳的救護車擺放醫院與擺放數量，同時決定救護車最佳的派遣路徑，使得傷患總等候時間最小化。

1.4 論文架構

本研究第一部分為緒論，介紹大量傷患緊急救護之救護車派遣問題的研究背景、研究動機、研究目的與研究問題簡介。第二部分為文獻回顧，首先介紹現今緊急醫療服務系統與大量傷患事件的概況，再來介紹救護車基地佈署與救護車派遣策略兩大類問題，最後介紹文獻與本研究的異同處。第三部分詳細介紹已知初始發車處之救護車派遣問題的整數規劃模式及相關的假設，設計兩個演算法得到較好的求解效率，並針對演算法與模式做出相關的數值測試。第四部份為可彈性調整初始發車處之救護車派遣問題，提出了兩個不同觀點的整數規劃模式求解，同時也提出兩個不同的演算法，最後進行不同模式與不同演算法的求解比較，並探討救護車初始配置方式的規則。第五部分總結本研究成果與貢獻，及提供未來可研究方向與建議。

第二章

文獻探討

本章一開始將回顧與介紹現有救護車的緊急醫療服務系統做，再介紹救護車靜態與動態的配置模式，接著介紹救護車的相關派遣問題，最後提出本研究與過去研究的異同處。

2.1 緊急醫療救護系統與大量傷患事件

2.1.1 緊急醫療救護服務系統

緊急醫療服務系統之救護指揮中心(Dispatch Center)接獲大量傷患事件報案時，救護派遣員即需利用相關資訊決定出勤單位、數量與視需求請求其它縣市提供資源，並將救護車派遣至合適的責任醫院。胡勝川(2002)認為緊急醫療救護系統的精神在於以下幾點：(1)接到報案能快速反應、(2)在現場施行必要的救護處置、(3)有急診醫師參與、(4)統一支配運用緊急醫療資源。

蘇喜等人(2001)指出，「美國在 1970 年代開始發展緊急醫療救護系統，1973 年將緊急救護定義為，在緊急情況下，有效地整合各式醫療服務，其中包括人員、醫療設備與各種醫療機構……」，因此緊急醫療救護系統為一綜合性的組織，藉由有效地管理人員、器材與資源，以達到提供民眾急難協助及緊急醫療處置之目的。依我國目前的緊急醫療救護法規定，緊急醫療救護系統由救護隊、救護指揮中心以及急救責任醫院組成，而其任務為「緊急或大量傷患之現場醫療處理」、「送醫途中緊急救護」、「離島與偏遠地重大傷病患之轉診」。

當緊急重大事故發生時，執行醫療任務的人員稱為緊急救護技術員(Emergency Medical Technicians, EMT)，依照訓練的等級分為初級救護技術員(EMT-1)、中級救護技術員(EMT-2)與高級救護技術員(EMT-P)，其中初級救護技術員與中級救護技術員可進行基本生命急救術(Basic Life Support, BLS)，而高級

救護技術員可執行高級生命急救術(Advanced Life Support, ALS)。一般而言，ALS 技術人員的人數遠少於 BLS 的技術人員，因為 ALS 的訓練成本較高再加上實際需要 ALS 的傷病患比例偏低，因此若派遣過多的 ALS 人員將不符合緊急醫療救護的實質效益。緊急醫療救護系統中，依照救護人員的派遣方式可分為單軌系統(Single-tiered)與多軌系統(Multi-tiered)，其中多軌系統為依照事情的嚴重程度派遣適當等級的救護人員出勤，而單軌則不分等級，使用多軌系統能將有限的 ALS 救護人員使用於真正需要的傷患(Kuehl, 2002)。

關於緊急醫療救護處理流程，如圖 2.1 吳青翰(2006)所示，Spaite et al. (1993) 提出以下的工作階段：

1. 接獲報案：緊急事故或重大災害發生時，指揮中心接獲報案及通話過程所需的時間。
2. 派遣出勤：指揮中心人員依照患者傷重情況，按照相關規範派遣適合的救護人員。其所需時間包含判斷出勤分隊及人員、通知分隊時間與分隊出勤延滯時間。
3. 去程：救護人員由消防站出發，至到達現場之路程，其時間為接獲報案至到達現場所花的時間，又稱為「反應時間(Response Time)」。
4. 接觸患者：指到達現場至實際接觸患者所花的時間。
5. 評估患者：救護人員對患者施以初步評估以及二次評估所花的時間。
6. 現場處置：救護人員依初步評估以及二次評估的生命徵象，施行適當處置所花的時間。
7. 移動患者：救護人員完成現場處置，開始移動患者至救護車所花的時間。而步驟 4~7 這段時間稱為「現場時間」。
8. 送醫旅程：將患者由案發地點送至醫院之路程。步驟 3~8 合稱為「到院前時間」。
9. 交接患者：患者送至醫院，交由護理人員處理，並完成相關事項交接及

更換器材所需時間。

10. 返隊旅程：救護人員與救護車離開醫院返回至原消防分隊的路程。

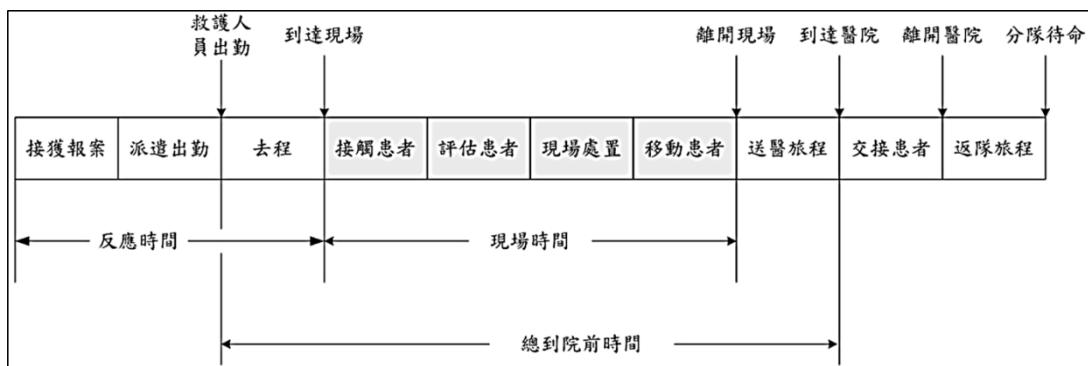


圖 2.1 緊急醫療救護流程圖

通常在大量傷患事件發生時，評估傷患的行為將交由現場駐點的醫護人員集中處理，且救護車載送傷患至醫院後，並不會回到原消防站待命，而是回到事故點繼續執行載送傷患的任務。

2.1.2 大量傷患事件

目前在大量傷患事件的主題上，尚未有明確的定義，經文獻回顧後發現，大量傷患的領域大致可以分為以傷患數量來定義的類型以及以是否需額外支援的類型，其中如我國的【衛生機關及醫療機構處理大量傷病患醫療救護作業要點】第二條中則具體以傷患人數來定義：「大量傷病患緊急醫療救護，是指單一事故、災害發生之傷病患人數達 15 人以上，或是預判可能發生 15 人以上傷病患之緊急醫療救護」，此定義方式的優點是清楚明確，當第一線緊急醫療救護人員或救護指揮中心決策者僅需以傷患人數即可啟動大量傷病患事故應變的相關措施。而吳青翰(2010)在文獻回顧中提到，除了以傷患人數定義大量傷患事故之外，亦有文獻提到相關的定義：「當傷患數量超過當地緊急醫療資源所能負荷之緊急事故，導致短時間內緊急醫療資源供需失衡」，此外也有文獻指出，大量傷患必須取決於傷患數量與事故地點是否會需要跨縣市的支援來決定之。

吳青翰(2010)在文獻回顧中提到，在大量傷患事件發生時，因為現場混亂、傷患嚴重程度不一且傷患眾多，因此傷患的資訊通常是不完整且不精確的，而事故點的狀況甚至會隨著時間而改變。因此在實務上，最先到達的救護人員必須評估現場的傷患數量、傷重程度與事故類型，並將此資訊回報給緊急應變指揮中心，而根據 Koenig et al. (1996)的 PICE(Potential Injury-Creating Event)命名規則，事故的後續發展可能是穩定的，也可能是正在發展中的，因此事故現場的指揮官必須隨時將現場狀況的變化回報給指揮中心的決策者，以便根據事故點傷患情況變化調整最適合的救護車派遣準則。

大量傷患事件對緊急醫療系統的衝擊，不僅僅是以醫療資源使用的數量或是傷患人數來衡量，舉例來說，當大量傷患事故發生在資源稀少的偏遠山區時，即便是少量的傷患，亦有可能瞬間癱瘓當地的緊急醫療救護系統。反之，若同樣的事件發生在緊急醫療資源密集度高的市區，可能當地的醫療資源即可自行處理。因此，若以醫療資源供給與資源需求間的關係，又可以分類為以下三種類型：「可控制的」(Controlled)、「破壞性的」(Disruptive)與「癱瘓性的」(Paralytic)，其中「可控制的」指的是當地醫療資源可自行處理，且不會對緊急醫療救護系統的日常運作造成影響；「破壞性的」代表雖然當地的緊急醫療資源可以處理，但對當地的緊急醫療救護系統已造成極大的影響；「癱瘓性的」代表當地緊急醫療資源已無法自行處理，必須藉由外在的資源來協助救災。

日常生活的一般醫療事故的處理流程與大量傷患事件的處理流程並不相同，針對一般事件而言，若短時間內發生兩件以上的事故或同一事故點有兩名以上的傷患，則依照事故的危急程度來派遣救護車，較危急的事故點將會優先處理。但是當大量傷患事故發生時，極短的時間內將累積大量的傷患，因此以有限的醫療資源不但無法同時處理這樣大量的傷患，而且也沒辦法將所有傷患及時地送至合適的醫院。因此，當大量傷患事故發生時，如何妥善安排並有效利用現有的緊急醫療資源，並將資源投注於最需要的傷患，並使得全數傷患在最小的等候時間內後送至適合的醫院便成為一個非常重要的課題。

在本研究中，我們綜合上述文獻對於大量傷患的各種定義以及各種特性的描

述，將大量傷患事故定義為「一個人為或天然所造成的災害，使得短時間內產生多個事故點且各事故點有著大量的傷患，且傷患人數與事故點數目已達到無法以一般日常緊急醫療救護系統之措施處理的情況，必須投入大量資源才能順利地安置所有事故點的傷患。」，而本研究期望以最小的傷患總等候時間為目標值，提出最佳的救護車派遣決策。

2.2 救護車派遣與緊急醫療救護系統佈署問題

2.2.1 救護車基地佈署問題

緊急醫療救護系統之中，反應時間對傷病患的存活率有最直接的影響，因此我們希望救護車皆能配置於距離事故發生地點最近的消防分隊(亦即救護車之醫療服務基地)，以縮短救護車的反應時間。一般而言，此類佈署問題大多將緊急事件發生點簡化為離散的需求點，例如：以鄰里或行政區作為緊急事件發生點，在此區域中發生的緊急事件皆視為此點的需求。在此問題中，決策者已知的資訊如下：需求點個數、救護車之醫療基地的候選點個數、救護車數量以及各個救護車之醫療基地所負責的需求點，各類文獻將會再依據不同的假設條件或目標函式，決定最佳的候選點放置救護車並決定各個候選點應該擺放的救護車數量。

救護車之醫療基地佈署問題可以再進一步分為以下三類：確定性靜態模式、機率靜態模式、動態模式三個類別。早期的研究大多為確定性靜態模式，此類模式主要應用於規劃階段，並假設救護車每次呼叫都一定能及時到達；機率靜態模式則依照救護車所在的消防站，給定各救護車能夠提供救援的機率，故病患有時並不能及時呼叫我到救護車；而動態模式強調指定區域之內的救護車必須重新佈署，以填補執勤任務的救護車所留下的空位。

2.2.1.1 確定性靜態佈署模式

緊急醫療救護系統之救護車醫療基地佈署集合覆蓋問題(Location Set Covering Problem, LSCP)由 Toregas et al. (1971)首先提出，此問題希望以最少的救護車基地數目，在各個需求點必須被覆蓋一次的情況下，決定救護車應該放置於

那些救護車基地中，如圖 2.2 所示，其中三角形代表醫療服務站，圓圈代表需求點，以三角形為圓心所畫出的半徑則是救護車基地提供的服務範圍，以 S3、S4 為圓心的兩個圓亦為此問題的一組可行解。

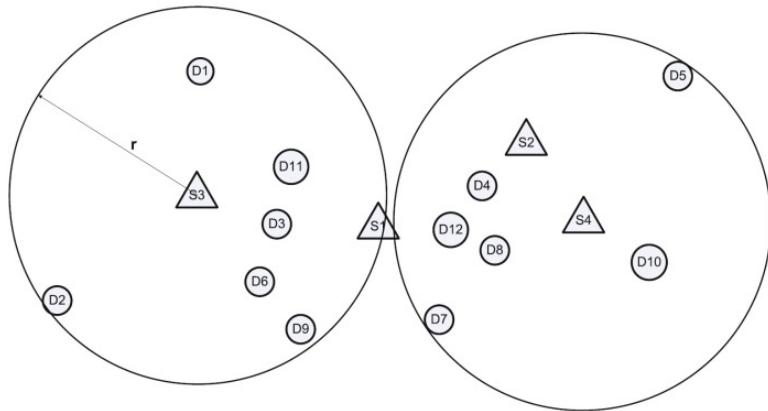


圖 2.2 佈署問題示意圖

資料來源：(Toregas et al, 1971)

為解釋不同文獻之不同數學模式，各模式所使用的索引、集合、參數與決策變數統一說明如下：

索引

i 需求點的索引

j 救護車基地的索引

集合

V 需求點的集合

W 可放置救護車的地點集合

N_i 可服務需求點*i* 的救護車基地集合

參數

d_i 需求點*i* 的需求量

p 救護車基地放置的數量

決策變數

$$x_j = \begin{cases} 1, & \text{基地 } j \text{ 放置救護車} \\ 0, & \text{ow} \end{cases}$$

$$y_i = \begin{cases} 1, & \text{需求點 } i \text{ 在救護車服務範圍內} \\ 0, & \text{ow} \end{cases}$$

救護車基地佈署集合覆蓋問題的數學模式(LSCP)如下：

$$\text{Min} \sum_{j \in W} x_j \quad (2.1)$$

$$\text{s.t. } \sum_{j \in N_i} x_j \geq 1, \quad i \in V \quad (2.2)$$

$$x_j \in \{0, 1\}, \quad j \in W. \quad (2.3)$$

式(2.1)為目標函式，最小化救護車基地總數。式(2.2)限制各個需求點必須至少被救護車基地覆蓋一次。式(2.3)為二元變數，決定救護車是否放置於地點 j 。

緊急醫療之最大化覆蓋佈署問題(Maximal Covering Location Problem,MCLP)後來由 Church and ReVelle (1974)提出，此模式在各個需求點加入需求量，此需求量可類比為某地區發生緊急事件的頻率，數學模式如下：

MCLP :

$$\text{Max} \sum_{j \in W} d_i y_i \quad (2.4)$$

$$\text{s.t. } \sum_{j \in N_i} x_j \geq y_i, \quad i \in V \quad (2.5)$$

$$\sum_{j \in W} x_j \geq p \quad (2.6)$$

$$x_j, y_i \in \{0, 1\}, \quad j \in W, i \in V. \quad (2.7)$$

其中式(2.4)為最大化總需求，變數 y_i 表示需求點 i 被覆蓋與否，亦即需求點 i 服務與否，式(2.5)限制服務站與需求點的關係，式(2.6)表示在可放置救護車

的地點中，總救護車數至少要放置 p 台。此問題欲決定各個事故需求點被哪些緊急醫療服務站覆蓋，以及救護車所放置的服務站位置。

在上述兩個模式中，各需求點僅被一個服務站服務，因此當救護車出勤時，該服務站勢必無法對需求點提供服務，Daskin and Stern (1981) 延伸 LSCP，於目標式中加入條件，使得服務站不只服務一個需求點，並希望最大化服務站覆蓋需求點的次數，此模式稱為分層目標集合覆蓋問題(Hierarchical Objective Set Covering Problem, HOSCP)。不過就算需求點被多個服務站覆蓋，部分事故頻繁的區域卻極可能僅由特定幾個服務站負責，導致其它服務站閒置比例的增高，Eaton and Morgan (1986) 針對此問題應用多目標規劃決定緊急醫療服務站的佈署問題，目標式期望使用最少的救護車基地，且最大化各個救護車基地服務的需求點個數。

需求點的覆蓋模式中，Gendreau et al. (1997) 建構了一個雙重標準模式(Double Standard Model, DSM)，此模式希望使被覆蓋兩次的需求點的數量越多越好，目的在於決定哪些需求點會被覆蓋兩次、哪些需求點僅會被覆蓋一次以及各個需求點將由哪些服務站提供服務。如圖 2.3 所示，其為問題的其中一個可行解，以三角形為圓心，半徑較大的 r_2 是服務站基本的服務範圍，本問題希望在半徑較小的 r_1 之內，能被覆蓋兩次的需求點越多越好，以需求點 D11 為例，可以發現其在 S3 及 S1 的 r_1 之內，故此需求點即為被覆蓋兩次的需求點；相對的，需求點 D7 僅被 S1 於 r_1 之內覆蓋一次。

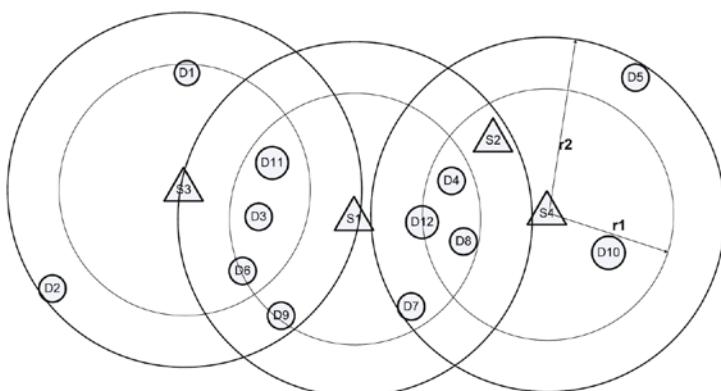


圖 2.3 DSM 示意圖

資料來源：(Gendreau et al., 1997)

縱使已有多種問題模式被發展，但是這些模式的服務站皆有固定的服務距離，若需求點不在服務範圍內，則不提供服務，但若剛好某些需求點的位置雖不在服務範圍內，但卻在服務半徑的不遠處，若因此而無法服務到該需求點，將會使醫療基地的服務功能大大地降低。Karasakal and Karasakal (2004)將此距離定義轉換成一個距離範圍，若需求點在此範圍之內，則規定可被服務，且被稱為部分覆蓋 (Partial Coverage)。如圖 2.4 呈現了部分覆蓋模式所帶來的影響，若現在有兩個候選點選擇其中一個地點放置救護車，實心的線代表固定的服務距離，若以最大化覆蓋需求點為目標，選擇 Y_1 可以覆蓋最多 6 個需求點，但若使用虛線所代表的部分覆蓋模式的距離範圍，則可發現選擇 Y_2 才是最佳解，覆蓋了 12 個需求點。

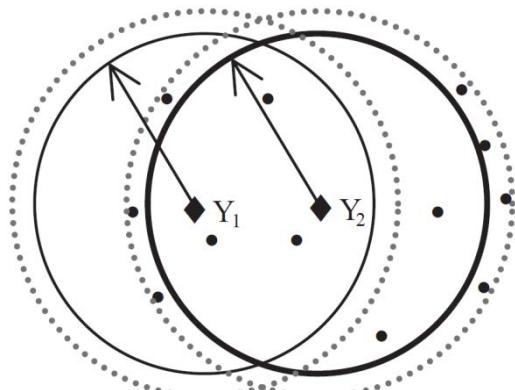


圖 2.4 部分覆蓋模式示意圖

資料來源：(Karasakal and Karasakal, 2004)

Berman et al. (2010) 定義出一個特殊的服務站覆蓋機制，首先想像各個服務站都會發射一個「訊號」，並由需求點接收之，此訊號強度依據距離增加而降低，而每一個需求點會接收各個救護車基地所射出的訊號，唯一不同的點在於，接收到的訊號強度會因為距離各服務站的遠近而有所不同，在此為需求點定義出門檻值，並將需求點向各個服務站所接收到的訊號強度加總，高於門檻值則服務該需求點，低於則否，此模式稱為合作覆蓋模式(Cooperative coverage model)。

2.2.1.2 隨機性靜態佈署模式

由於確定性的救護車基地佈署問題並未將救護車的值勤機率列入考慮，所以

Daskin (1983) 將救護車忙碌的機率加入模式的假設中，當救護車處於忙碌的狀態則無法對需求點提供服務，其中救護車忙碌的機率是由報案速率 λ 與服務速率 μ 輔以二項式分配所計算出的，並以最大化期望覆蓋當作目標函式，求解救護車基地位置與救護車基地所覆蓋的需求點。而 Repede and Bernardo (1994) 針對各個需求點在不同時間點上可能有不同的需求量，加入了時間的維度，且發現需求點覆蓋的比例由 36% 增加至 84%，而反應時間則下降了 36%。

Larson (1974) 首先提出超立方等候模式(Hypercube Queuing Model)應用於多服務者的緊急救護問題，此問題中假設需求點發生狀況時，皆派遣最近的救護車至現場執勤，其中將救護車狀態分為忙碌(1)與空閒(0)兩種，又因為此模式為一有限狀態的連續時間馬可夫過程，故依此性質可計算出各個狀態的穩態機率。Toro et al. (2013) 以超立方模式為基礎延伸出聯合救護車派遣與佈署的研究，此模式以最小化平均反應時間為目標，不但決定救護車放置的位置與各服務站所轄的需求點而且決定出派遣的規則並依此規則列出一個清單，此規則針對各個需求點而言，列出一個派遣清單，排序方式即為救護車出勤順序。舉例來說，表 2.1 為某一個需求點求解出的救護車派遣順序清單，當此需求點發生事故時，清單上派遣順序一號的救護車將會前往值勤，但若此時該救護車處於忙碌狀態，則會派出派遣順序二號的救護車出勤，以此邏輯推演，即成為救護車的派遣規則，而各個需求點皆會算出一個這樣的派遣清單，提供醫療服務站一個最佳的救護車派遣策略。

表 2.1 派遣順序清單

派遣順序	醫療服務站
1	S4
2	S2
3	S1
4	S3

2.2.1.3 動態佈署模式

靜態的模式適用於規劃的階段，但缺點是缺少了實際運作時靈活性的調動，動態的模式中，為了達到比較好的覆蓋效果，在一天當中，救護車所在的位置可以重新配置，所以此類模式中除了救護車的佈署之外，救護車該如何重新配置位置也是動態佈署模式中重要的課題。此類模式可分成以下三種類型，即時求解的模式(Gendreau, Laporte, and Semet, 2001)、預先計算出最佳的救護車位置的模式(Gendreau, Laporte, and Semet, 2006)、綜合系統隨機性同時應用馬可夫性質的模式或是在特定的條件做出重新配置的決策的模式(Andersson and Varbrand, 2007)。

即時求解模式中，Gendreau et al. (2001) 提出以即時的方式求解緊急醫療救護車基地的重新配置，此模式係根據前述的 DSM 模式 (Gendreau et al., 1997)，引入新的變數，並依據各個不同時段的需求進而求解得出救護車最佳的重新配置方式，稱為「動態雙重標準模式」(Dynamic Double Standard Model, DDSM)。Gendreau et al. (2006) 則是依照前述的 LSCP 模式(Toregas et al., 1971) 為基礎，再針對系統的歷史資料，計算出需求點平均的報案速率以及服務站的平均服務速率，並且以系統中空閒的救護車數量給定系統不同的狀態，最後求解得到各個狀態的救護車所應該配置的服務站，而此模式是以最大化覆蓋為目標值求解。如圖 2.5 所示，一開始有四台救護車，S3 派出一台之後，經由模式計算之後使 S1 移至 S3 待命，使系統目標值保持最佳。

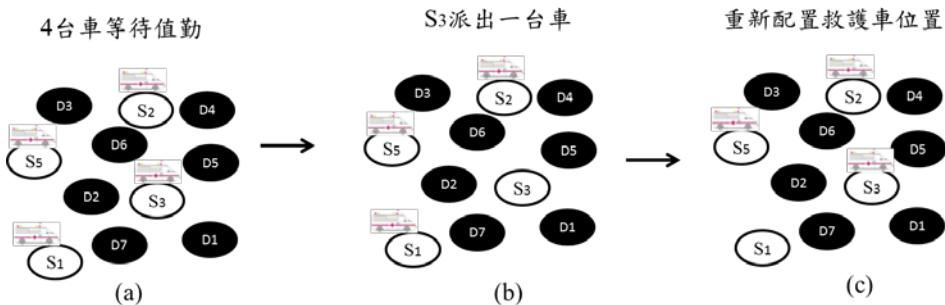


圖 2.5 救護車重新配置示意圖

Andersson and Varbrand (2007) 針對每一個需求點定義出各自的「準備值」，

並以此值計算救護車的配置方式。更詳細地說，此模式將一個城市，劃分為多個區域，每個區域依據歷史資料可以得知單位時間內的需求量，即事故發生頻率，例如：50 件事故/天。假設我們已知救護車的數量，而救護車所座落的位置也是已知的資訊，同時，每個需求點被哪些救護車所服務亦是已知的資訊。在此情況之下可以為各個需求點計算出一個量化的資訊，稱為「準備值(Preparedness)」，此數值亦代表服務站對某需求點的服務能力，此數值越高代表服務能力越佳。準備值的計算公式如式(2.8)， c_j 為單位時間內的報案頻率， l 為依照與需求點的距離排序好的救護車順序， t_j^l 為第 l 台救護車到需求點的行車時間， γ^l 則是決策者自行給定的計算因子。此模式中每一個需求點皆會事先給定一個準備值的最低門檻值，當事故發生後，系統將會少一台救護車，必會使得需求點的準備值下降，此時若發現有某些需求點低於門檻值，則會藉由此模式計算出最佳解，並移動救護車的位置使得所有需求點至少都滿足準備值的門檻值。

$$p_j = \frac{1}{c_j} \sum_{l=1}^{L_j} \frac{\gamma^l}{t_j^l} \quad (2.8)$$

$$\text{其中 } t_j^1 \leq t_j^2 \leq \dots \leq t_j^{L_j} \quad (2.9)$$

$$\gamma^1 \geq \gamma^2 \geq \dots \geq \gamma^{L_j} \quad (2.10)$$

2.3 救護車輛派遣問題

在大量傷患問題中，包含了派遣哪些車(位置)及多少車(數量)到事故點。大部分的文獻中雖然以巨觀的方式估計各個事故點的資源需求，或是估計各事故點中傷患的存活率後，指派適當的資源及種類至相關的事故點，但是卻較少針對救護車抵達事故點後，應該送往哪間醫院，做出明確的描述，同時也並未針對個別傷患之嚴重程度做區別及探討個別傷患之送醫問題。

Tayfur and Taaffe (2009) 提出了一個以整數規劃模式為架構的醫院緊急疏散計畫，目的是藉由救護車的來回載送，將院內所有病患在一定的時間之內全數

送至安全的醫院或安全的避難所。如圖 2.6 為醫院緊急疏散示意圖，當緊急災害發生時，多台救護車將會聚集於醫院，並將院內病患一一載送至安全的避難地點，直到院內所有病患與醫護人員全部疏散完畢後，醫院緊急疏散才告一段落。救護車將會依照模式中給定的目標函式以及避難所可容納的病患數目決定適當的避難醫院，並在完成一趟載送任務之後再次回到醫院，準備載送下一個病患至合適的避難醫院，且在此問題中假設救護車到達避難醫院後必須立即返回原醫院，繼續進行疏散，而不能夠行駛至其它的避難醫院。

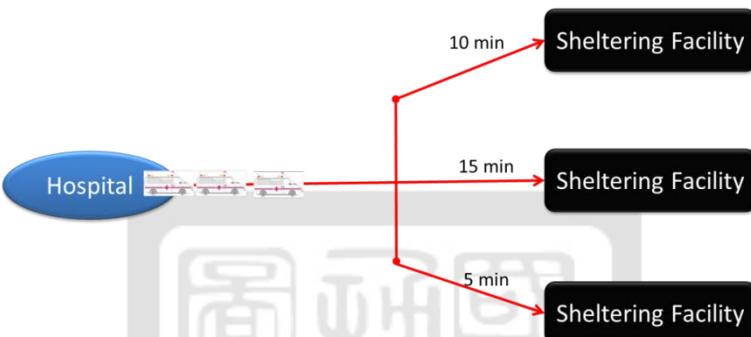


圖 2.6 醫院緊急疏散示意圖

醫院疏散問題中會給定一個期望最大疏散時間 T (ex : $T = 4$ 小時)，醫院管理者希望能在此時間之內找到一個最佳的疏散模式。Tayfur et al. (2009) 以最小化醫院總成本為目標提出一個整數規劃求解模式，其中目標式的成本包含車輛運輸成本、車輛租用成本、護理人員成本以及留在院內病患所造成的懲罰性成本，此研究提供了醫院管理者評估是否能在自身有限的醫療資源中，完成有效的醫院疏散。而 Bish et al. (2011) 則是以最小化風險為目標，以整數規劃模式進行醫院疏散問題的求解，其中最小化風險的直觀意義即為病患的死亡機率或病患病情惡化的機率最小化，此風險值視作給定的參數，此風險值的計算牽涉到病患病情嚴重程度、使用的救護車種類不同(不同救護車配置等級的救護員)、行車時間等而有所不同。

此類醫院緊急疏散問題具有特殊形式的車輛途程問題 (Vehicle Routing Problem ; VRP) 的架構，如圖 2.7 所示， H^0 表示需要疏散的醫院，可以將其想像成一個 VRP 問題中的 Depot；弧線上的數字代表行車所需耗費的單位時間； H^1 、

H^2 代表可接收病患的避難醫院；因為假設救護車到達避難醫院後，必須再次回到原醫院，故此為一簡化版的 VRP。因為疏散工作必須在有限時間 T 之內完成，故此問題可以一個有限的時間網路圖表示之，如圖 2.8 所示，結點上的數字代表在醫院時的各個時間點，節點 0 代表 H^0 醫院疏散的起始點，而節點 9 代表 H^0 醫院疏散的終止時間點，節線代表救護車的流量，其中節點上方的節線代表前往 H^1 醫院後再返回至 H^0 醫院的救護車數量，由於上載病患與卸載病患各需要花 1 單位時間，加上去回總共 2 單位時間，故救護車將會在節點 4 重新回到 H^0 醫院；同樣的道理，於節點下方的節線則代表前往 H^2 的救護車數量；而介於節點之間的節線代表在該期處於閒置狀態的救護車。最後，在各個節點之間的救護車流量必須保持流量守恆。

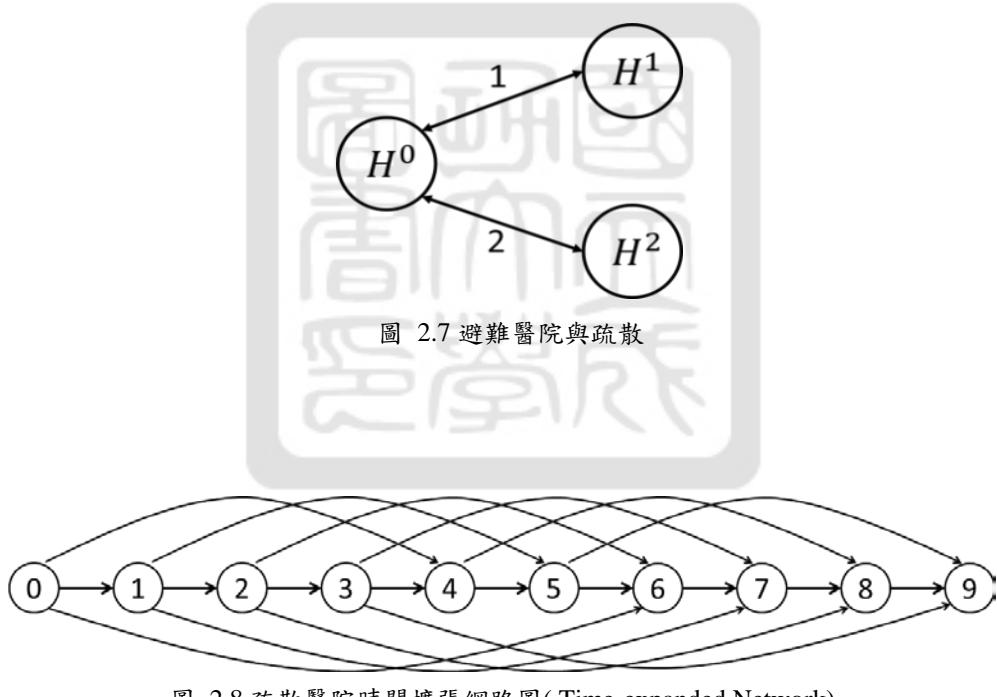


圖 2.8 疏散醫院時間擴張網路圖 (Time-expanded Network)

除了醫療救護車基地的佈署與覆蓋問題外，我們也關心救護車本身的救援動向，這就關係到指派中心的如何提供一個有效率的調派策略。Carter et al. (1972) 發表了最常見且最為直覺的救護車派遣規則，即最小化反應時間，但此模式的缺點是，不能保證每次的派遣都可以達到最佳解。當考慮兩個責任區域，假設兩區域分別為區域 A 與區域 B，若區域 A 的報案數顯著高於區域的時候，Cunningham

and Harries (1988) 提出了區域 B 的救護車可前往區域 A 支援，以降低平均反應時間，而 Repede and Bernardo (1994) 則發現，當區域 A 已經在執行救援任務時，可派遣區域 C 的車輛支援區域 A，而不是派遣較近的區域 B，不過此情況只適用於區域 B 報案頻率高於區域 C 時使用。Weintraub et al. (1999) 在智利的聖地牙哥提出一個車輛服務的電子系統，救護車輛將會在報案地點之間移動，在此同時，調度者在決定車輛即將前往下個報案點之時，亦會試著維持充足的醫療準備，以應付較為嚴重的病患。

Gong and Batta (2007) 針對大規模的災難所造成的多處事故，以最小化最大處理時間以及總處理時間為目標，發展兩階段的救護車配置模式。第一階段為在事故初期配置適當的救護車數量至每一事故點，而第二階段為當有新的事故產生時，重新配置救護車，以提高救護車的利用率。文獻中定義，第一階段為事故初期，此時一部救護車有可能需要再送一名以上的傷患，而為了能夠最小化傷患搜尋時間同時最大化救護車利用率，因此將救護車派至某個傷患的群集(Cluster)，再由搶救人員在群集內尋找傷重程度高的傷患，由於事故初期各群集內的傷患數量未知，因此文獻中假設傷患增加率為時間的函數，再配合初始傷患數量以及服務率，提出了一個決定性模式，估計救護車數量與處理完成時間之關係。該研究最後提出兩種疊代求解方式，以求得最佳救護車派遣方式。而傷患的數量可能會隨著災難事件的演進而增加，並產生新的傷患群集，此時即進入第二階段，進入第二階段後，必須重新調整救護車的配置，而文獻中將此重新派遣的方式為將處理現有事故群集的救護車，調派至新增的傷患群集，因此依據重新配置的時間點、各群集當下的傷患數量與救護車數量以及傷患增加率，提出模式估計救護車數量變更的兩個群集之處理完成時間。

Jotshi, Gong and Batta (2009) 針對大規模災難造成多處事故，且考量資訊不完整的情形下，為求在最少服務時間內，將高優先順序的傷患送醫，其中傷患的優先順序分為四級，第四級為預期死亡傷患，而第一級為輕微受傷傷患，第一級與第四級傷患皆不送至醫院；作者運用資料融合(Data Fusion)的方法，提出緊急救護車輛派遣之決策評估指標，期望在災難發生後能夠有效地執行緊急救護車派遣及路徑指派。而車輛的派遣又分為兩階段：載運傷患(Patient Pickup)以及傷患

送醫(Patient Delivery)，其中在第一階段載運傷患時期，採用了 Gong and Batta (2007) 將鄰近傷患視為一個群集的觀念，以(1)群集內各嚴重等級的傷患數量來計算該群集的權重、與(2)群集與搶救車輛之距離，來決定救護車派遣的先後順序；而第二階段的傷後送醫問題則為，在車輛載運傷患之後，就需要指派其送往合適醫院，其中相關的參數有醫院的容量、傷患在醫院急診室的等候時間以及事故地點與醫院的距離。

2.4 小結

由前述的文獻回顧可知，當緊急的大量傷患事故發生時，對緊急醫療救護系統所帶來的影響並非一般日常醫療事故可比擬，當大量傷患事故發生時，醫療單位的決策內容包括了救護車的出勤單位、傷患的後送醫院、傷患的派送順序及合適的醫院，其中事故點的位置、救護車的位置與傷患的嚴重程度，將影響到救護車整體的派遣決策。

目前現有的文獻中針對大量傷患事故或災難事件的著眼點大多是救護車的「反應時間」或是救護車的「到院時間」，而在大量傷患事件發生時，必須同時考慮救護車多趟的載送，但是文獻中大多缺乏了對救護車「反應時間」與「到院時間」的整合，且並未同時針對「反應時間」與「到院時間」提出一套有系統的決策模式。因此本研究將整合救護車的「反應時間」與「到院時間」，並在考量整體可用的醫療資源後，做出大量傷患之緊急醫療救護事件的最佳決策。此外，由文獻中可得知，不管是「反應時間」或「到院時間」，只要者兩者時間增大，即會造成傷患的死亡率升高，是故，本研究將提出一最小化總傷患等候時間之數學規劃模式，以決定救護車的派遣數量與派遣路徑並決定傷患最佳的派遣醫院。

第三章

已知初始發車處之救護車派遣問題

本章將詳細描述並定義救護車派遣問題的整數規劃模式以及相關假設，首先將會詳細描述救護車調派問題的基本架構與假設。第二節將呈現已知初始發車處之救護車派遣問題的整數規劃模式 AD_{TS}^{IL} ，其中 AD 代表 Ambulance Dispatching；右上角的 IL 代表 Initial Location，意思是救護車初始發車處已知；右下角的 TS 代表此模型是以時空網路圖(Time Space network)為基礎建構而成的數學模式。第三節將介紹兩種不同求解概念的演算法，分別是傷患尋找距離最近之救護車的貪婪演算法 GDA_d^{IL} 與傷患平衡演算法 BDA^{IL} ，其中 GDA 代表 Greedy Dispatching Algorithm；下標 d 代表此演算法是以距離的遠近當作準則來派遣救護車，而 BDA 則代表 Balanced Dispatching Algorithm，此演算法考量了傷患等候的公平性，以平衡各事故點傷患的等候時間當作救護車的派遣準則。第四節將針對整數規劃模式與不同的演算法之求解效率與品質做出相關比較與分析。

3.1 問題描述與相關假設

3.1.1 問題描述

本研究以時空網路圖(Time Space Network)為基礎，將連續的時間離散化後，在網路圖上的各個時刻做出決策，決定救護車派遣的路徑，而離散化的單位時間長度則可由使用者自行定義，例如：若以 5 分鐘為一期，代表每 5 分鐘做一次決策；另外網路圖的大小 T 則是代表所有事故點的傷患，必須在時間 T 之內全數送至醫院，而網路圖中的時刻為 $t = 0 \sim T$ 。網路圖中的節點分別為消防站(S)、事故點(e)、醫院(H)與傷患數量(P)，連接彼此的節線則是依照事故點、消防站與醫院間的距離連結，因此，距離越大將會使得 T 變大，進而使網路圖增大。本研究期望在網路圖上的各時刻決定出最佳的救護車派遣決策，並最小化傷患總等候時

間。

本研究中的時空網路圖分為兩個種類，第一種類為救護車的配送路徑所構成的時空網路圖，第二種為各時刻的傷患流量時間網路圖。在救護車的派遣時空網路圖中，一台救護車即成為一「層」網路，各層之間由事故點節點與醫院節點串鍊之，圖 3.1、圖 3.2 為各層救護車網路圖與事故點、醫院的簡單示意圖，三層網路圖代表三台救護車，其中節點 S1、S2 與 S3 分別代表三台救護車由消防站 1、消防站 2 以及消防站 3 出發，而在本研究中，假設每間消防站都只有一台救護車。另外，節點 e 為發生事故的地點，節點 H 代表醫院，每一層連接的節線代表救護車可以走的其中一個路徑之一，而救護車將傷患送至醫院後，若事故點還有傷患滯留，救護車將會返回事故點載送傷患至醫院，反之，若事故點已無傷患，救護車將回到消防站待命。

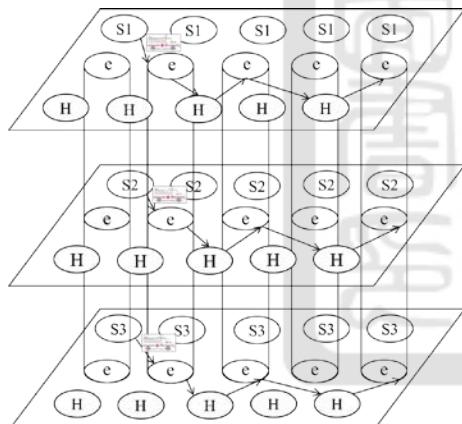


圖 3.1 事故點與各層救護車關係示意圖

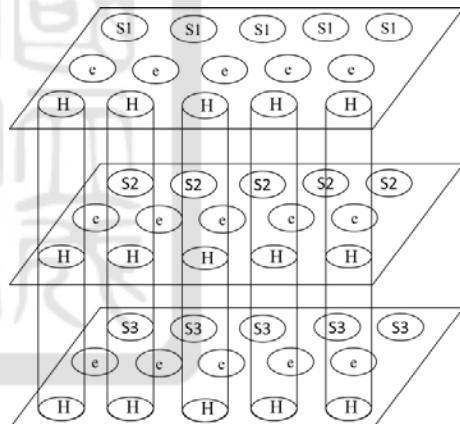


圖 3.2 醫院與各層救護車關係示意圖

除了救護車與救護車之間的連接關係外，各層救護車同時也會依照與事故點以及與醫院的距離繪製出時空網路圖，假設目前有一輛救護車、一間消防站、一個事故點與一間醫院，而消防站距離事故點 1 單位時間，事故點距離醫院 1 單位時間，在此例中假設事故點有 2 個傷患，救護車網路圖的連接方式如圖 3.3 所示，在此例中，T 為 8 單位時間，各時刻 $t=0 \sim 8$ ，其中不同的節點 S 代表不同時刻的消防站，不同的節點 e 代表不同時刻的事故點，不同的節點 H 代表不同時刻的醫

院，而 S 旁的數字「1」則表示消防站 S 有一台救護車可供使用。在救護車的派遣網路圖中，不同節點之間的節線連接方式是依照事故點、醫院與消防站彼此之間的距離(即救護車行車時間)連接，若是將救護車在事故點與醫院處理傷患所花的時間考慮進去，則會有相同節點之間的節線，這些節線的連接代表處理傷患所需要花的時間，而在救護車完成傷患救援任務之後，救護車將由醫院返回原消防站，最後由 S 節點在 $t=8$ 時流出網路圖。

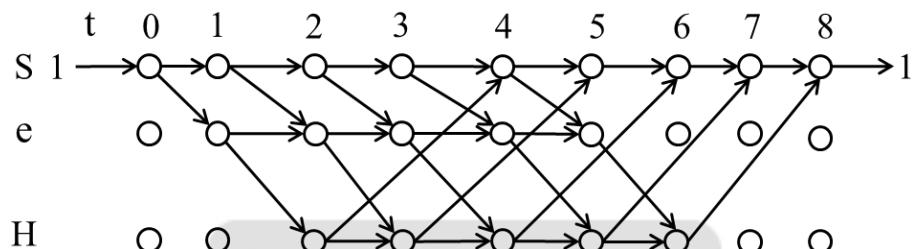


圖 3.3 單一救護車時空網路圖

圖 3.4 為傷患的流量網路圖，字母 e 那一列代表事故點的節點，字母 H 那一列代表醫院的節點，而字母 e 旁的數字「2」則代表事故點的傷患數量是 2 位。傷患網路圖中的節線代表傷患的流向，若節線連接的是同一個事故點，代表傷患停留於同一個事故點，若節線連接的是醫院 H 則代表傷患被送至醫院 H。而在傷患網路流量圖中 T 的大小與救護車網路圖中 T 的大小同樣都是 8 單位時間，亦即救護車必須在 8 單位時間內將傷患全數送至醫院，因此節點 e 在時刻 $t=8$ 時並未被任何節線連接。

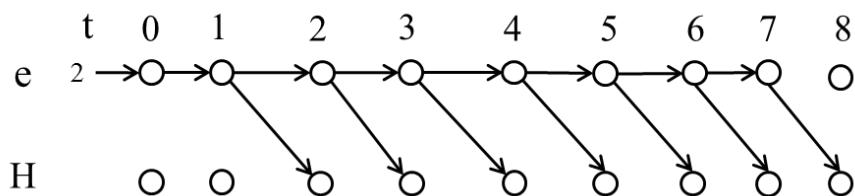


圖 3.4 傷患流量網路圖

3.1.2 問題假設

以下是本研究所需要的相關假設及限制。

1. 救護車的行車速率皆相等。
2. 假設各消防隊與事故點之間皆是行走最短距離，且不會有道路上的交通阻塞，亦即行車時間固定，事故點與醫院之間亦有相同的假設。
3. 救護車不閒置。舉例來說，若消防站最快可以在 2 單位時間內支援某事故點，則此消防隊不會拖延到 2 單位時間以上才派出救護車。
4. 假設事故點的傷患在救護車來了以後，不管傷重程度，一定會上車，並且被載到醫院治療。
5. 救護車若被派到事故點，一定不能空車去醫院，或是不能空車回到原本的消防隊，亦即，派出去的救護車到了事故點後不能有閒置的情況發生。
6. 假設在時刻 T 內要將所有傷患全數送至醫院。
7. 假設救護車到事故點與醫院的行車時間(距離)皆為整數。
8. 一台救護車只能載一位傷患。
9. 醫院沒有容量限制。因為在實務上，大量傷患派送時，若醫院真的額滿了，救護車還是會繼續將傷患送至該醫院，故有此假設。
10. 不論傷患的傷重程度，醫院都必須接收傷患。
11. 每個消防站只有一台救護車，因此救護車編號即為消防站編號。

3.2 數學模式

3.2.1 已知初始發車處之救護車派遣之時空網路圖模式

3.2.1.1 參數與變數定義

本研究使用的變數索引、參數、集合定義及變數定義詳細列於表 3.1、表 3.2、

表 3.3。

表 3.1 變數索引與集合說明表

符號	說明	符號	說明
k	救護車與消防站編號	K	救護車的數量集合
i	事故點編號	H	網路圖上可達的醫院集合
j	醫院編號	A_k	網路圖上可達消防站節點 k 的時刻集合
t	時空網路圖內的各個時刻	A_i	網路圖上可達事故點節點 i 的時刻集合
t'	特定變數由事故點或醫院出發的時刻	A_j	網路圖上可達醫院節點 j 的時刻集合
I	網路圖上可達的事故點集合	A_T	網路圖上所有可達節點的時刻集合

表 3.2 參數說明表

參數符號	參數說明
P_{i0}	事故點 i 在時刻 $t = 0$ 時的傷患數量。
T	接獲報案後，將傷患全數送至醫院，所估計出的最長時間。
d_{ij}	由事故點 i 到醫院 j 的行車時間。
d_{ki}	由消防站 k 到事故點 i 的行車時間。

表 3.3 變數定義表

變數符號	定義說明
RP_{it}	第 i 事故點在時刻 t 所剩下的傷患數量。
p_{ijt}^k	第 k 台車在時刻 t 由事故點 i 送至醫院 j 的傷患數量。
w_{it}^k	$\begin{cases} 1, & \text{第 } k \text{ 台車在時刻 } t \text{ 由消防站 } k \text{ 派至事故點 } i。 \\ 0, & \text{ow} \end{cases}$
x_{ijt}^k	$\begin{cases} 1, & \text{第 } k \text{ 台車在時刻 } t \text{ 將傷患由事故點 } i \text{ 送至醫院 } j。 \\ 0, & \text{ow} \end{cases}$
y_{jit}^k	$\begin{cases} 1, & \text{第 } k \text{ 台車在時刻 } t \text{ 將傷患由醫院 } j \text{ 送至事故點 } i。 \\ 0, & \text{ow} \end{cases}$
z_{jt}^k	$\begin{cases} 1, & \text{第 } k \text{ 台車在時刻 } t \text{ 完成任務後，由醫院 } j \text{ 返回消防站 } k。 \\ 0, & \text{ow} \end{cases}$
e_{it}^k	$\begin{cases} 1, & \text{第 } k \text{ 台車在時刻 } t \text{ 繼續待在事故點 } i。 \\ 0, & \text{ow} \end{cases}$
h_{jt}^k	$\begin{cases} 1, & \text{第 } k \text{ 台車在時刻 } t \text{ 繼續待在醫院 } j。 \\ 0, & \text{ow} \end{cases}$
st_t^k	$\begin{cases} 1, & \text{第 } k \text{ 台車在時刻 } t \text{ 繼續待在消防站 } k。 \\ 0, & \text{ow} \end{cases}$

3.2.1.2 整數規劃模式

目標式

$$\text{Min} \quad \sum_{k=1}^K \sum_{t=1}^T \sum_{i \in I} \sum_{j \in J} (t + d_{ij}) x_{ijt}^k \quad (3.1)$$

式(3.1)考慮所有在時刻 t 由事故點派至醫院的救護車，計算所有救護車抵達醫院的時刻後加總即為傷患的總等候時間，其中 $t + d_{ij}$ 代表救護車抵達醫院的時刻。

限制式

$$p_{ijt}^k \leq x_{ijt}^k \quad \forall k \in K, i \in I, j \in H, t \in A_i \quad (3.2)$$

式(3.2)限制每台救護車最多可容納的傷患數量。

$$\sum_{i \in I} w_{it}^k + st_t^k = \sum_{j \in H} \sum_{t' \in A_j} z_{jt'}^k \quad \forall k \in K, t \in A_k \quad (3.3)$$

$$\sum_{i \in I} w_{it}^k + st_t^k = 1, \quad \forall k \in K, t \in A_k \quad (3.4)$$

$$\sum_{j \in H} z_{jt}^k + st_{(t-1)}^k = 1, \quad \forall k \in K, t \in A_k \quad (3.5)$$

式(3.3)、式(3.4)與式(3.5)針對網路圖中的消防站節點，限制各個節點的救護車流量平衡，確保進入節點的救護車流量等於離開節點的救護車流量，其中式(3.3)為包含由醫院回到消防站的救護車流量平衡，而式(3.4)與式(3.5)分別代表消防站節點中的第一個節點與最後一個節點的救護車流量平衡。

$$\sum_{j \in H} x_{ijt}^k + e_{it}^k = \sum_{j \in H} \sum_{t' \in A_j} y_{j�t'}^k + w_{i(t-d_{ki})}^k + e_{i(t-1)}^k \quad \forall k \in K, i \in I, t \in A_i \quad (3.6)$$

式(3.6)限制事故點的救護車流量平衡，確保進入各事故點節點的救護車流量與離開各事故點節點的救護車流量相等。

$$\sum_{i \in I} y_{j�t}^k + z_{j�t}^k + h_{j�t}^k = \sum_{i \in I} \sum_{t' \in A_i} x_{ijt'}^k + h_{j(t-1)}^k \quad \forall k \in K, j \in H, t \in A_j \quad (3.7)$$

式(3.7)限制救護車在各個醫院節點的流量平衡，確保進入醫院節點的救護車流量與離開醫院節點的救護車流量相等。

$$RP_{it} + \sum_{j \in H} p_{ijt}^k = RP_{i(t-1)} \quad \forall k \in K, i \in I, t \in A_i \quad (3.8)$$

式(3.8)對各時刻事故點本身所剩下的病患作限制，確保上一期所剩的傷患數目與當期剩下的傷患數目加上當期被載走的病患數目相等。

$$\sum_{k \in K} \sum_{j \in H} \sum_{t' \in A_j} y_{jit'}^k \leq RP_{it} \quad \forall k \in K, i \in I, t \in A_i \quad (3.9)$$

當事故點還留有傷患時，若此時醫院有救護車可用，式(3.9)將迫使救護車至事故點載送傷患；反之，若某事故點已無傷患，救護車將不會前往該事故點救援。

$$w_{it'}^k \leq e_{it}^k \quad \forall k \in K, i \in I, t \in A_i, t' \in A_k \quad (3.10)$$

$$y_{jit'}^k \leq e_{it}^k \quad \forall k \in K, i \in I, t \in A_i, t' \in A_j \quad (3.11)$$

$$x_{ijt'}^k \leq h_{jt}^k \quad \forall k \in K, j \in J, t \in A_j, t' \in A_i \quad (3.12)$$

式(3.10)、式(3.11)限制救護車在事故點所需停留的期數，當有救護車從消防站或醫院前往事故點時，必須予以限制；式(3.12)則是針對救護車停留於醫院的時間做限制。

$$x_{ijt}^k, y_{jit}^k, w_{it}^k, z_{jt}^k, e_{it}^k, h_{jt}^k, st_t^k, \in \{0, 1\} \quad \forall k \in K, i \in I, j \in H, t \in A_T \quad (3.13)$$

$$p_{ijt}^k, RP_{it} \geq 0, \text{ and integer.} \quad \forall k \in K, i \in I, j \in H, t \in A_T \quad (3.14)$$

式(3.13)、式(3.14)分別為二元變數限制與變數的非負及整數限制。

3.3 貪婪演算法與傷患平衡之救護車派遣法

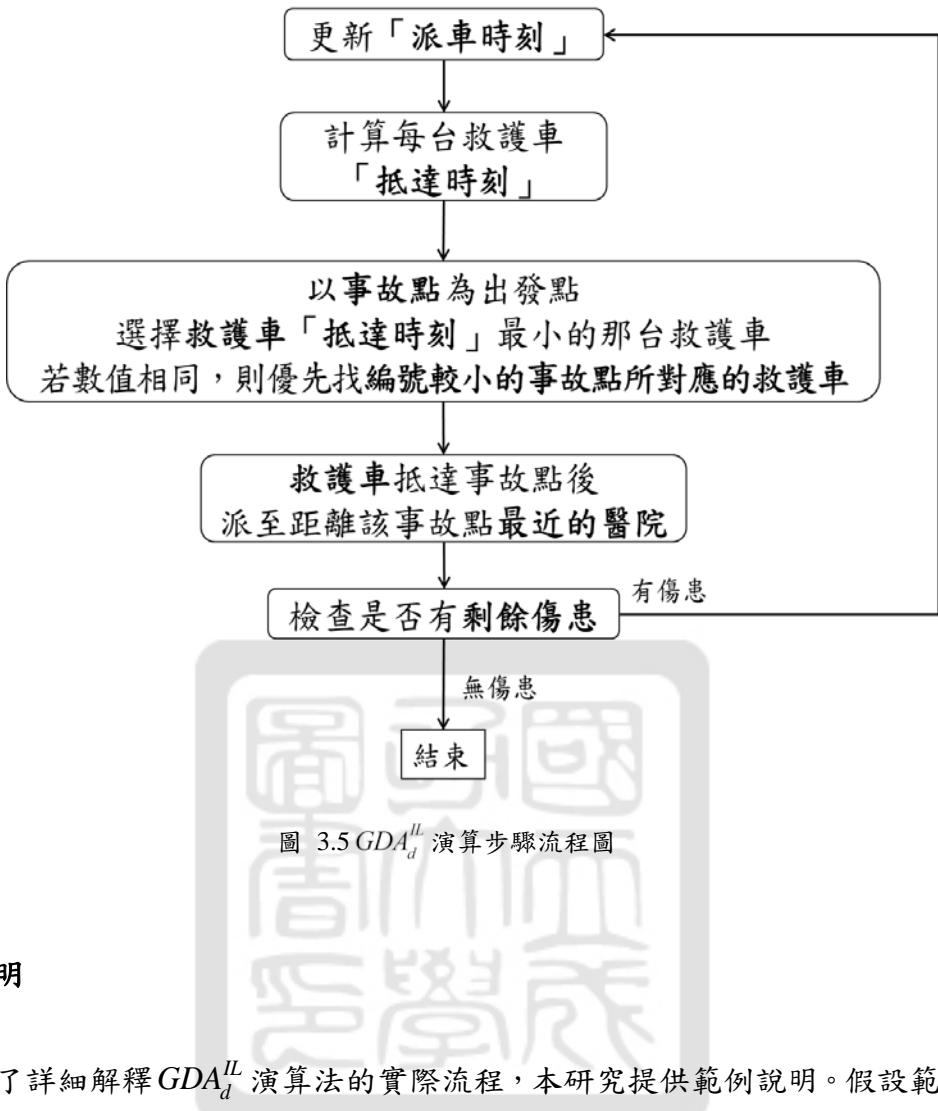
當使用 3.2 節提到的整數規劃模式求解時，求解時間會隨著問題規模的擴大而增加(包括傷患數量、救護車數量、事故點的數量、事故點距離等等不同的參數增加)。為了提升求解效率，本小節提出了兩種不同的演算法，分別為貪婪演算法 GDA_d^{IL} 以及平衡傷患數量之救護車派遣法 BDA^{IL} ，希望提升救護車派遣模式的求解效率。兩種演算法的概念不盡相同，相關的求解概念將於下面小節解釋。

3.3.1 GDA_d^{IL} 演算法介紹

GDA_d^{IL} 的基本概念是以傷患本身當作出發點，以傷患找救護車的概念為出發點，每位傷患依據自己所在的事故點，尋找距離自己最近的救護車。

首先定義「抵達時刻」與「派車時刻」，顧名思義，抵達時刻指的是救護車抵達事故點的時刻，而派車時刻則是代表救護車可派遣的時刻，亦即救護車處於完成任務且正在等待下一個指派任務的時刻，此外，當救護車抵達事故點後便直接將傷患送至最近的醫院。

GDA_d^{IL} 演算法的派車流程如下圖 3.5 所示，救護車開始派遣任務前，所有救護車的「派車時刻」皆為零，接下來依照救護車所在位置以及事故點所對應的距離計算每台救護車抵達事故點的「抵達時刻」，再來找尋每一個事故點最近的救護車，比較彼此距離後找出距離最近的救護車派至該事故點，當距離有相同數值時，優先選擇事故點編號最小的或編號最小的救護車。救護車抵達事故點後，自動將傷患送至距離該事故點最近的醫院，抵達醫院後檢查各個事故點是否有剩餘的傷患，若有剩餘的傷患，則重新更新「派車時刻」並重複上述流程，直到事故點沒有傷患為止。



範例說明

為了詳細解釋 GDA_d^{LL} 演算法的實際流程，本研究提供範例說明。假設範例如下圖 3.6 所示，總共有 2 個消防站(S1、S2)、2 個事故點(e1、e2)、2 間醫院(H1、H2)，而每個消防站皆有 1 台救護車可供使用，每個事故點的傷患人數皆為 2 人，消防站距離事故點分別是 1 單位時間、2 單位時間與 3 單位時間、4 單位時間，醫院距離事故點分別是 1 單位時間、2 單位時間與 2 單位時間、1 單位時間。

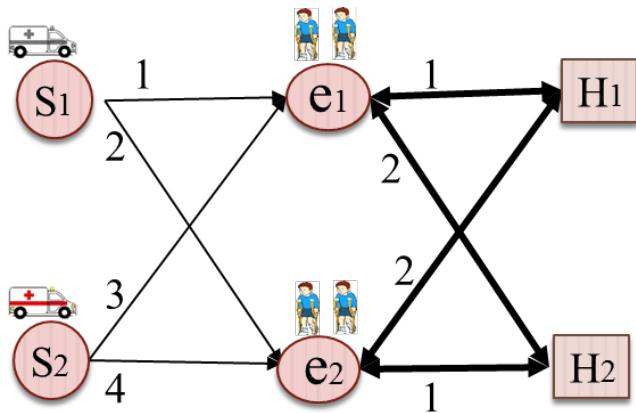


圖 3.6 範例說明示意圖

為了方便說明，本研究將從 1 號消防站(S1)出發的救護車稱為 1 號車，而從 2 號消防站(S2)出發的救護車稱為 2 號車，另外此範例假設救護車抵達事故點或醫院後，不需要任何處理傷患的時間，可以立即載傷患上車並立即開往醫院或是立即讓傷患下車並從醫院開至事故點。救護車派送的流程如下圖 3.7 ~ 圖 3.14，如圖 3.7 所示，時刻 $t=0$ 為救護車的「派車時刻」，此時救護車在消防站待命；如圖 3.8 所示，對 e_1 而言最近的救護車是 1 號車，故在時刻 $t=1$ 時，1 號車抵達 e_1 ，並將一位傷患載送至最近的 H_1 ，如圖 3.9 所示，救護車抵達後，兩台車的「派車時刻」分別為 0 與 2，重新計算「抵達時刻」後，距離 e_1 最近的救護車是 1 號車，故 $t=3$ 時 1 號車抵達 e_1 並將傷患送至最近的醫院 H_1 ，如圖 3.10；而圖 3.11 時，因為 e_1 已無傷患，故只需要計算兩台救護車至 e_2 「抵達時刻」，因為兩台車的派車時刻分別是 $t=4$ 與 $t=0$ ，因此兩台車的「抵達時刻」分別是 $t=6$ 與 $t=4$ ，因此在 $t=4$ 時，2 號車已經抵達 e_2 並將傷患載至最近的醫院 H_2 ，如圖 3.12 所示；此時兩台車的「派車時刻」分別為 $t=4$ 與 $t=5$ ，而 e_2 的「抵達時刻」分別為 $t=6$ 與 $t=7$ 故將 1 號車派至 e_2 ，如圖 3.13；最後將傷患送至最近的醫院 H_2 後，完成救護車派遣，如圖 3.14 所示。

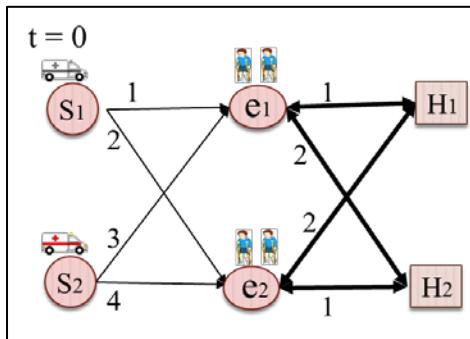


圖 3.7 GDA_d^{IL} 演算流程圖(一)

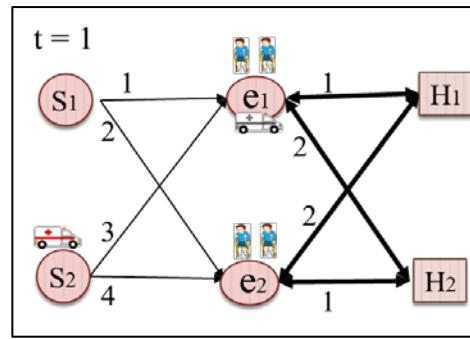


圖 3.8 GDA_d^{IL} 演算流程圖(二)

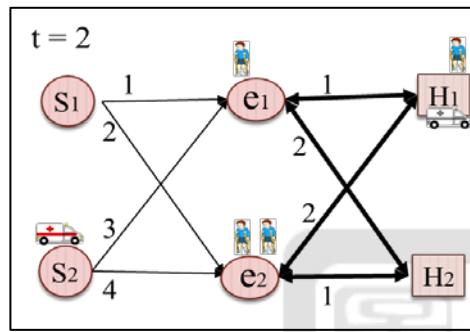


圖 3.9 GDA_d^{IL} 演算流程圖(三)

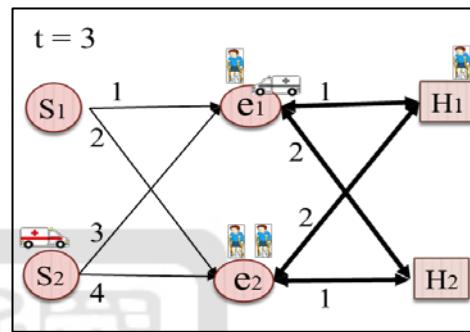


圖 3.10 GDA_d^{IL} 演算流程圖(四)

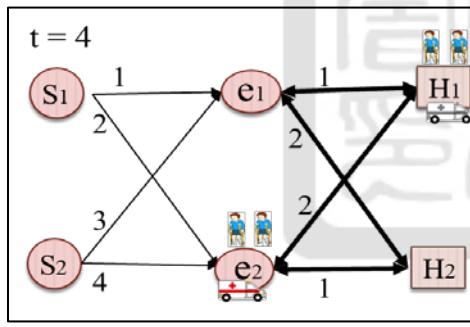


圖 3.11 GDA_d^{IL} 演算流程圖(五)

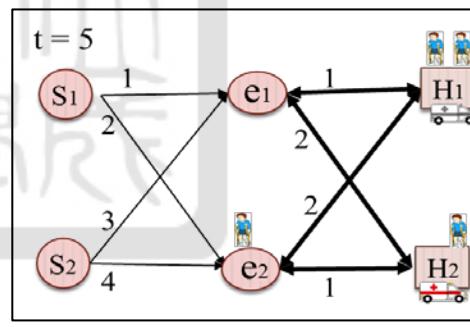


圖 3.12 GDA_d^{IL} 演算流程圖(六)

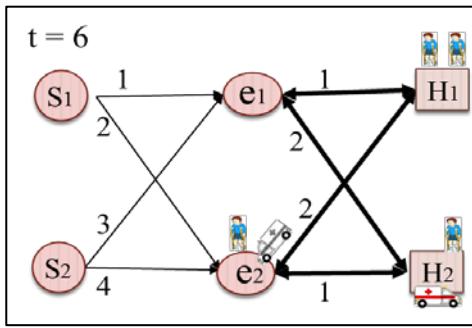


圖 3.13 GDA_d^{IL} 演算流程圖(七)

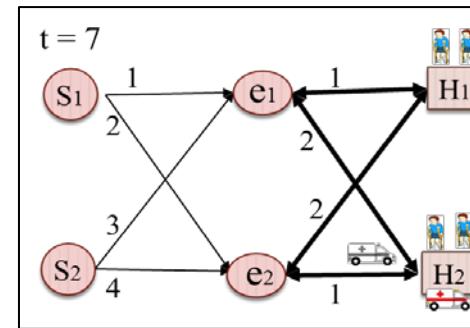


圖 3.14 GDA_d^{IL} 演算流程圖(八)

3.3.2 BDA^{IL} 演算法介紹

BDA^{IL} 演算法以事故點傷患數量當作事故點選擇的優先考慮條件，救護車將會派至傷患最多的事故點，這種派遣方式會使得各事故點的傷患數量以平衡、相等的速率下降，故取名為「Balance」。

演算法的派車流程如圖 3.15 所示：

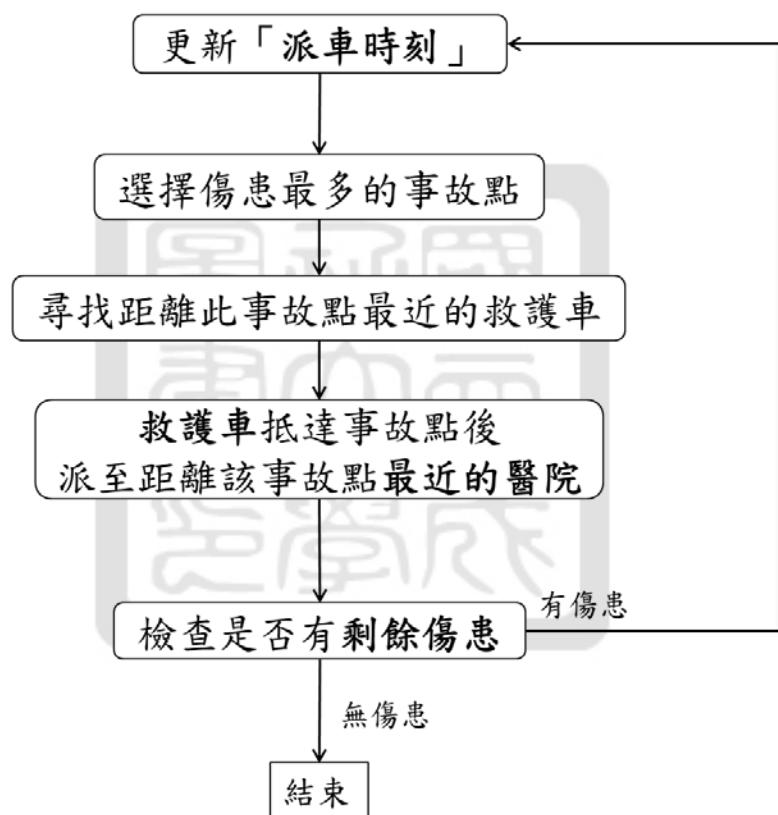


圖 3.15 BDA^{IL} 演算步驟流程圖

範例說明

說明範例如上圖 3.6，本節以相同的例子演示不同的演算法，並沿用上一節的名稱，包括 1 號車、2 號車分別代表 1 號消防站與 2 號消防站派出的救護車。

圖 3.16 ~ 圖 3.23 為救護車的派遣流程，在 $t=0$ 時兩事故點傷患數量一樣，所以先將救護車送至編號較小的事故點 e_1 ，再將傷患送至最近醫院 H_1 ，如圖 3.16、圖 3.17 所呈現，而圖 3.18 ~ 圖 3.21 中，傷患抵達 H_1 後，重新計算各救護車的「派車時刻」分別為 $t=2$ 、 $t=0$ ，此時選擇事故點傷患較多的 e_2 派送，對 e_2 來說，兩台車的「抵達時刻」都是 $t=4$ ，故選擇編號較小的 1 號車派至 e_2 ，而 1 號車派至 e_2 後送至最近的醫院 H_2 後(如圖 3.21)，此時 1 號車的「派車時刻」為 $t=5$ ，而 2 號車的「派車時刻」為 $t=0$ ，此時兩事故點傷患數量相同，故選擇編號較小的事故點 e_1 ，而 1 號車與 2 號車到 e_1 的「抵達時刻」分別是 $t=3$ 與 $t=7$ 所以將 2 號車派至 e_1 後再送到最近醫院 H_1 ，如圖 3.19、圖 3.20 所示；圖 3.21 再度更新兩台車的「派車時刻」，分別為 $t=5$ 與 $t=4$ ，此時僅剩 e_2 有傷患，而兩台車派遣至 e_2 的「抵達時刻」分別為 $t=6$ 與 $t=6$ ，在此選擇編號較小的 1 號車派至事故點 e_2 ，最後送至最近的 H_2 結束派送，如圖 3.22 與圖 3.23。

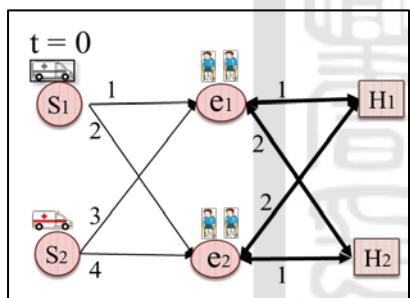


圖 3.16 BDA^H 演算流程圖(一)

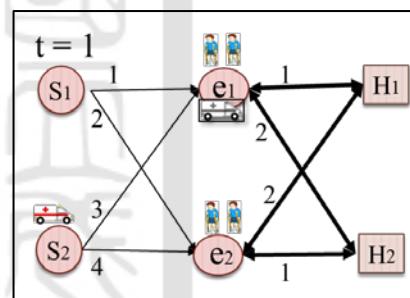


圖 3.17 BDA^H 演算流程圖(二)

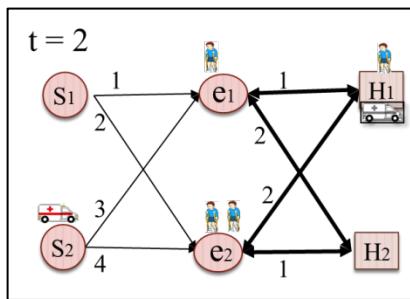


圖 3.18 BDA^H 演算流程圖(三)

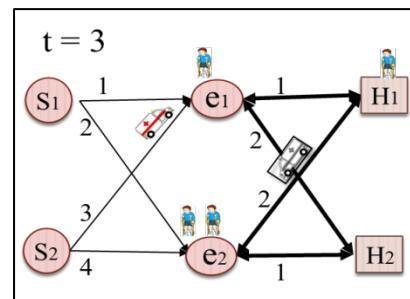


圖 3.19 BDA^H 演算流程圖(四)

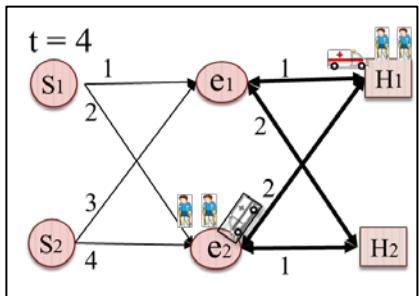


圖 3.20 BDA^{IL} 演算流程圖(五)

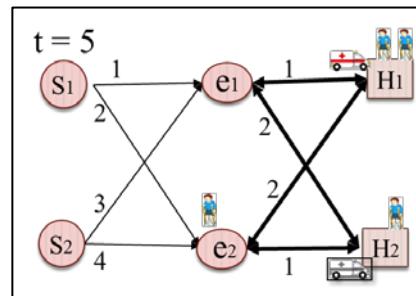


圖 3.21 BDA^{IL} 演算流程圖(六)

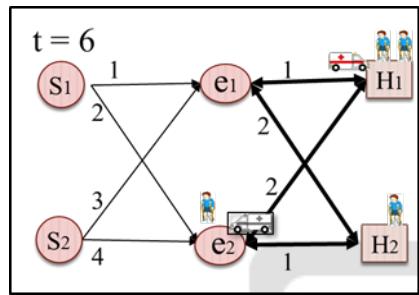


圖 3.22 BDA^{IL} 演算流程圖(七)

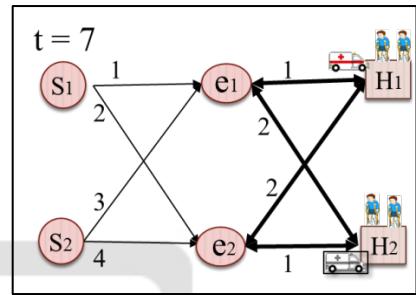


圖 3.23 BDA^{IL} 演算流程圖(八)

3.3.3 小結

本節介紹兩種演算法，提升原模式的求解效率，兩種演算法分別為 GDA_d^{IL} (人找車)與 BDA^{IL} (傷患平衡下降)的方式，兩種演算法有其不同的優缺點。而在演算法 GDA_d^{IL} 中我們發現人找車與車找人的方法其實是等價的，差別只在於排序「抵達時間」方式先後順序不同，但最終結果都只是找出「抵達時間」的最小值。

下一小節將針對演算法 GDA_d^{IL} 和演算法 BDA^{IL} 求解方法與原始整數規劃模型求解方法進行數值比較。

3.4 數值分析

3.4.1 數據產生方式

本研究定義了一個 20×20 的座標，並在座標上隨機產生不同數量的消防站、事故點、醫院與傷患人數，如圖 3.24 為示範圖例，而各節點彼此之間的距離直

接以直線距離計算之。

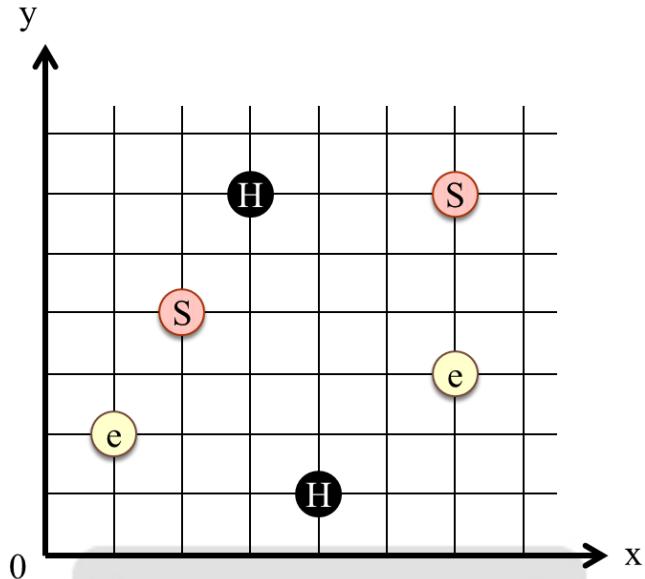


圖 3.24 座標圖示意圖

本研究測試了 400 種不同的例子後，再記錄下求解時間與 Gap，而每一個例子會在座標上隨機產生不同的救護車數量(K)、事故點數量(e)、傷患總數(P)與醫院數量(H)，其中各參數的數量區間為：救護車 2 台~ 9 台、事故點 2~ 6 個、傷患總數 1 個~ 40 個、醫院的數目 2~ 4 個。為了方便觀察各參數與求解時間和 Gap 之間的關係，本研究將 400 個例子進行分類，如表 3.4 所示，依照救護車數量(K)、事故點數量(e)與傷患總數(P)進行分類，總共組成 8 種不同的 Case，其中救護車數量的分類為大(2 ~ 5 台車)、小(6 ~ 9 台車)兩類，事故點數量的分類為大(2 ~ 4 個事故點)、小(5 ~ 6 個事故點)兩類，傷患總數的分類為大(1 ~ 20 個傷患)、小(21 ~ 40 個傷患)兩類。舉例來說，當隨機產生的例子為：救護車 3 台，事故點 5 個，傷患總數 35 個時，此例子將被分類為 Case 4，而本研究於每個 Case 都測試了 50 個不同的例子。

表 3.4 測試 Case 分類表

Case (K,e,P)	救護車數量(K)	事故點數量(e)	傷患總數(P)
Case 1 (小,小,小)	2 ~ 5	2、3、4	1 ~ 20
Case 2 (小,小,大)	2 ~ 5	2、3、4	21 ~ 40
Case 3 (小,大,小)	2 ~ 5	5、6	1~20
Case 4 (小,大,大)	2 ~ 5	5、6	21 ~ 40
Case 5 (大,小,小)	6 ~ 9	2、3、4	1 ~ 20
Case 6 (大,小,大)	6 ~ 9	2、3、4	21 ~ 40
Case 7 (大,大,小)	6 ~ 9	5、6	1 ~ 20
Case 8 (大,大,大)	6 ~ 9	5、6	21 ~ 40

3.4.2 數值測試

本節將上述 8 個 Case 所產生的 400 個例子分別以整數規劃模式 AD_{TS}^{FL} 、演算法 GDA_d^{IL} 與演算法 BDA^{IL} 求解後，將求解時間(單位：秒)與最佳解的 Gap(單位：%)列出。本研究使用 Corei7 處理器之個人電腦，並在 Windows7 的作業系統下，使用最佳化軟體 Gurobi 5.6.0 版以及演算法 GDA_d^{IL} 、演算法 BDA^{IL} 求解。

本研究中，若最佳解的求解時間超過 2 小時 (7200 秒)，模式將停止求解，若此時尚未解出最佳解，則演算法的 Gap 可能會出現小於零的情況。另外，因為 Gap 的計算皆是以最佳化軟體所求得的目標值為基準，因此模式 AD_{TS}^{FL} 所陳列出的 Gap 皆為零，而本研究使用的貪婪演算法，求解時間皆在 1 秒以內，求解相當迅速，故求解時間直接以 0 秒紀錄。

求解時間與 Gap

表 3.5 將模式 AD_{TS}^{FL} 的求解時間與演算法 GDA_d^{IL} 、演算法 BDA^{IL} 的 Gap(%)一一列出比較。表中可以發現，模式 AD_{TS}^{FL} 的求解時間依不同的 Case 而有極大的差異，其中最快的是 Case5，只需要平均 12 秒左右，由此得知當救護車數量多且

事故點數量與傷患數量小時，求解時間將大幅降低，其餘求解時間至少都需要 500 秒以上。演算法 GDA_d^{IL} 在各個 Case 中的求解品質表現都不錯，且平均求解 Gap 都在 4% 以下，值得一提的是，模式 AD_{TS}^{FL} 在 Case4 中的求解時間需要 6000 秒左右，但此時演算法 GDA_d^{IL} 的 Gap 僅僅只有 0.8%，求解表現非常好。此外，演算法 GDA_d^{IL} 不但求解迅速，而且除了 Case3 與 Case6 以外，其它 Case 的平均求解 Gap 都在 3% 以下。演算法 BDA^{IL} 的平均 Gap 明顯高於演算法 GDA_d^{IL} ，顯然此演算法無法有效地減少傷患的總等候時間，但可以發現此演算法的 Gap 表現受到傷患數量影響，當傷患數量少時(Case1、Case3、Case5、Case7)，Gap 明顯較低，另外可以發現當救護車數量上升時， BDA^{IL} 有下降的趨勢。

表 3.5 求解時間與 Gap 比較表格

	AD_{TS}^{FL}		GDA_d^{IL}		BDA^{IL}	
	求解時間(s)	Gap(%)	求解時間(s)	Gap(%)	求解時間(s)	Gap(%)
Case 1	367.72	0%	0	2.5%	0	13%
Case 2	3721	0%	0	1.9%	0	23.2%
Case 3	3335	0%	0	3.8%	0	11.2%
Case 4	5904	0%	0	0.8%	0	18.7%
Case 5	12.6	0%	0	2.2%	0	7.8%
Case 6	1202.4	0%	0	3%	0	15.1%
Case 7	629.6	0%	0	2.4%	0	8.4%
Case 8	1176.1	0%	0	2.5%	0	14.4%

演算法 GDA_d^{IL} 在 Case3 與 Case6 的 Gap 表現比較差，因此本研究列出這兩個 Case 求解時間與 Gap 的盒鬚圖，試圖探究相關的影響原因。圖 3.25 與圖 3.26 為 Case3 與 Case6 的求解時間與 Gap 的盒鬚圖，其中圖上的 A 點代表最大值、B 點代表平均值、C 點代表中位數。Case3 中，可以發現演算法 GDA_d^{IL} 的 Gap 最大值為 37%，而平均值與中位數分別為 3.8% 與 0.6%，而模式 AD_{TS}^{FL} 的平均求解時間需要 3335 秒且求解時間的最大值為 7204 秒，雖然 GDA_d^{IL} 的 Gap 最大值為 37% 並不是個好的求解 Gap，但考量此值為 Gap 分佈中的離群值，並非正常的分佈，

而且 GDA_d^{IL} 最好的求解 Gap 為 0%，再加上模式 AD_{TS}^{FL} 需要較久求解時間，因此演算法 GDA_d^{IL} 在 Case3 仍然是個不錯的求解方法。在 Case6 中，最差的 Gap 是 19%，而 Gap 的平均值與中位數分別是 3% 與 2.1%，模式的求解時間中，求解時間最大值為 7200 秒而平均求解時間為 1202 秒，雖然 AD_{TS}^{FL} 的平均求解時間優與於 Case3 很多，但考量過多離群值且 1202 秒也並非迅速的求解時間，再加上演算法 GDA_d^{IL} 的求解 Gap 僅為 3%，因此在 Case6 中，演算法 GDA_d^{IL} 表現依然不差。

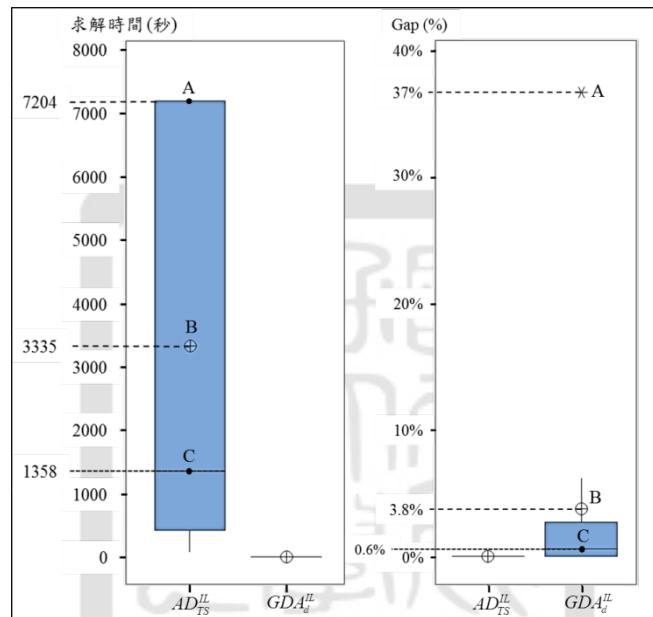


圖 3.25 Case3 盒鬚圖

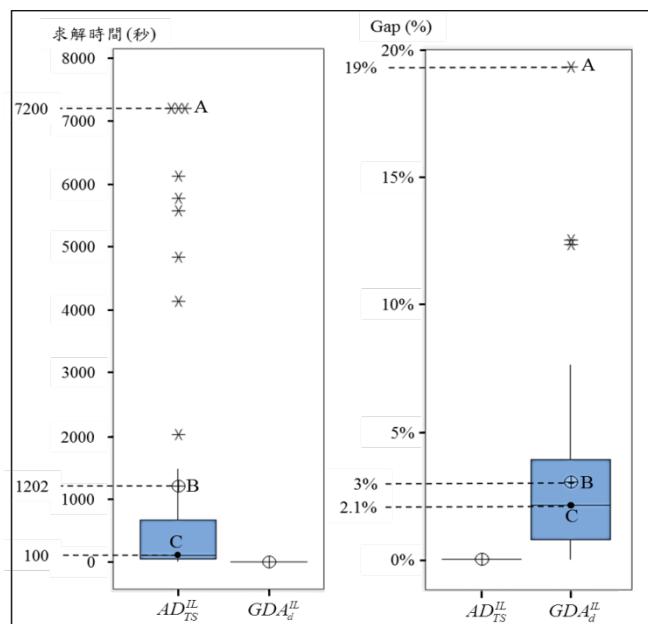


圖 3.26 Case6 盒鬚圖

演算法 BDA^{IL} 是以各事故點的傷患平衡下降當作救護車的派遣準則，因此本研究期望演算法 BDA^{IL} 在傷患等候時間的標準差有較好的表現，亦即各個傷患彼此的等候時間差距較小。為了比較模式 AD_{TS}^{FL} 、演算法 GDA_d^{IL} 與演算法 BDA^{IL} 的傷患等候時間標準差，本研究將傷患等候時間的標準差除以每個傷患的平均等候時間，使得三種求解方法有標準化的比較基準，為了方便解釋，本研究將傷患等候時間標準差與傷患平均等候時間相除的數值統一稱為「標準化傷患等候時間標準差」或是簡稱「標準化標準差」。

表 3.6 與圖 3.27 分別列出 400 個測試例子中各個例子的「標準化傷患等候時間標準差」，其中表 3.6 列出 400 個例子中，不同求解方式之標準化標準差的平均值、標準差、最大值與最小值等相關的統計量；而圖 3.27 則是呈現所有例子的資料點，其中藍色的是模式 AD_{TS}^{FL} 、紅色的是演算法 GDA_d^{IL} 、綠色的是演算法 BDA^{IL} 。相較於模式 AD_{TS}^{FL} 與演算法 GDA_d^{IL} ，圖 3.27 可以發現演算法 BDA^{IL} 的傷患等候時間的「標準化傷患等候時間標準差」有較低的趨勢，而演算法 GDA_d^{IL} 的標準化標準差偏高，另外表 3.6 中可以發現模式 AD_{TS}^{FL} 與演算法 BDA^{IL} 的標準化標準差的平均值剛好都是 0.381，都優於演算法 GDA_d^{IL} ，而在標準化標準差的最大值中，演算法 BDA^{IL} 為 0.767，明顯優於演算法 GDA_d^{IL} 的 0.914 與模式 AD_{TS}^{FL} 的 0.905；因為演算法 BDA^{IL} 的概念是盡量使所有傷患的等候時間相同，所以會使本來等候時間較低的傷患變得等候時間比較長，而原本等候時間長的傷患變得等候時間短，導致標準化標準差的最小值為 0.174，並未低於模式 AD_{TS}^{FL} 與演算法 GDA_d^{IL} ，但仍然在不錯的範圍內。

表 3.6 標準化標準差統計表格

	AD_{TS}^{FL}	GDA_d^{IL}	BDA^{IL}
Avg.	0.381	0.401	0.381
Std.	0.104	0.092	0.077
Max	0.905	0.914	0.767
Min	0.158	0.158	0.174

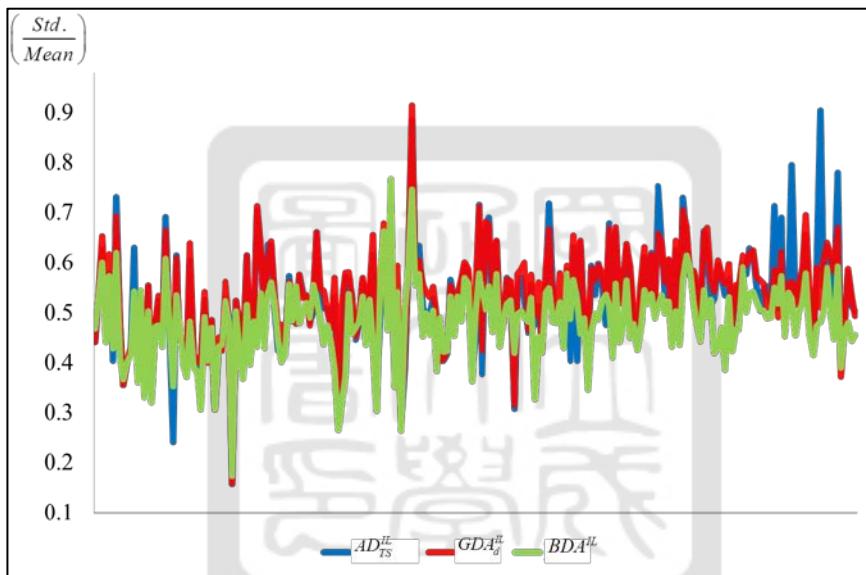


圖 3.27 標準化傷患等候時間標準差比較圖

3.4.3 小結

本節先在自行定義的座標上，隨機產生救護車派遣的例子，其中包括救護車數目、事故點數目、醫院數目與傷患的數目等參數，並同時將不同的參數分類成 8 種 Case，每個 Case 隨機產生 50 個例子測試，因此總共有 400 個例子進行數值測試。

數值測試中，首先針對問題的目標值--「傷患總等候時間」比較求解時間與演算法 GDA_d^{IL} 、演算法 BDA^{IL} 的求解品質，再來針對演算法與傷患等候時間的「標

準化標準差」來比較是否有差異。第一部分數值測試中發現演算法 GDA_d^{IL} 在各 Case 的求解 Gap 皆低於 4%，且模式的求解時間普遍偏高，因此決策者若使用演算法 GDA_d^{IL} 做為決策方針，不但可以在短時間內立刻得到解，同時也可以得到品質不錯的解，就算在表現最差的 Case3 與 Case6 中，演算法 GDA_d^{IL} 仍然是不錯的求解方法。演算法 BDA^{IL} 雖然在總傷患等候時間的表現不佳，但若希望傷患彼此的等候時間較為相近，並考量傷患的等候心情，演算法 BDA^{IL} 可使最大的傷患等候時間標準差最小，同時可以有效地降低「標準化傷患等候時間標準差」。



第四章

可彈性調整初始發車處之救護車派遣問題

本研究在第三章的問題中，假設救護車的初始位置是已知的，且都是由消防站出發，而在研究中發現，除了一開始是由消防站出發，以及結束回到消防站之外，救護車其它的派遣路徑都在事故點與醫院之間來回，直到所有傷患皆送至醫院為止，而在研究測試中發現，在大量傷患的情況下，救護車派遣的事故點與醫院將趨於固定，因此本章將在忽略消防站的假設下，模擬出無限多個傷患之救護車派遣情境，並期望在此假設下找出救護車與醫院之配對關係以及各醫院、事故點和救護車的分群機制，如此一來便可提供給決策者一個救護車派遣的遵循法則。

本章假設救護車的初始位置是未知的，在大量傷患事件下，救護車的位置可以放置於任意醫院，以推算出救護車與醫院的配對關係，因此除了求解救護車的派遣路徑之外，本章節將同時探討救護車擺放的位置與數量。本章先定義可彈性調整初始發車處之問題後，再提出相關假設與限制，同時在此假設與限制下建立可彈性調整初始發車處之時空網路圖救護車派遣模式 AD_{TS}^{FL} ，發展一個以最小化總等候時間為目標值的整數規劃模式，提供決策者救護車的最佳派遣路徑與最適當的救護車放置醫院，其中 AD 代表 Ambulance Dispatching；右上角 FL 代表 Flexible Initial Location，意思是救護車初始位置可彈性調整；右下角 TS 代表 Time Space network。除了時空網路圖的模式外，本章亦以不同的觀點建構可彈性調整初始發車處之順序式救護車派遣模式 AD_S^{FL} ，其中右下角 S 代表 Sequential，代表救護車依順序派遣。另外，本研究也提供了兩種貪婪演算法提升求解速度。本章的最後提供不同求解模式的求解時間與求解效率的數值比較，並歸納救護車分群(救護車擺放的醫院)趨勢與影響分群的原因。

4.1 問題描述與假設

4.1.1 問題描述

在本研究中，假設醫院的總數為集合 J ，事故點的數量為集合 I ，可用的救護車總數集合為 K ，各事故點 i 的傷患人數為 P_i 。而救護車派遣時，將連續的時間離散化，並將派遣的時刻分為 $t = 0 \sim T$ ，本研究將在時刻 $t = 0$ 時，決定哪些醫院該放置救護車，並決定救護車放置的數量，同時在 $t = 0 \sim T$ 之間，以最小的傷患等候時間為目標決定在各個時刻 t 之救護車最佳派遣路徑。

4.1.2 問題假設

本研究的相關假設如下：

1. 每台救護車的行車速率皆一樣。
2. 假設各事故點與醫院之間皆是行走最短距離，且不會有道路上的交通阻塞，亦即行車時間固定。
3. 救護車不閒置。舉例來說，若救護車最快可以在 2 單位時間內支援某事故點，則此救護車不會拖延到 2 單位時間以上才派出救護車。
4. 事故點的傷患在救護車來了以後，不管傷重程度，一定會上車然後被載到醫院治療。
5. 救護車若被派到事故點，不能夠空車去醫院。
6. 在 T 時間內要將所有病患都送至醫院。
7. 救護車到事故點與醫院的行車時間(距離)皆為整數。
8. 一台救護車只能載一位傷患。
9. 醫院沒有傷患容量限制。
10. 不論傷患的傷重程度，醫院都必須接收傷患。
11. 每間醫院可以放置零到多台救護車。

4.2 時空網路圖之救護車派遣整數規劃模式(AD_{TS}^{FL})

4.2.1 參數與變數定義

本節所使用的變數與參數定義如下表 4.1、表 4.2、表 4.3 所示：

表 4.1 變數索引與集合說明

符號	說明	符號	說明
k	救護車編號	K	救護車的總數
i	事故點編號	J	醫院的總數
j	醫院編號	A_i	網路圖上可到達事故點 i 的時刻集合
t	時空網路圖的各個時刻	A_j	網路圖上可到達醫院 j 的時刻集合
t'	特定變數由事故點或醫院出發的時刻	A_T	網路圖上 T 時間內的時刻集合
I	事故點的總數		

表 4.2 參數說明

參數符號	參數說明
RP_{i0}	事故點 i 於 $t = 0$ 時的傷患數量。
T	接獲報案後，估計出的最長救援時間。
d_{ji}	由醫院 j 到事故點 i 的時間。
d_{ij}	由事故點 i 到醫院 j 的時間。

表 4.3 變數說明

變數符號	定義說明
x_{kijt}	$= \begin{cases} 1, & \text{第 } k \text{ 台車在時刻 } t \text{ 由醫院 } j \text{ 派去第 } i \text{ 個事故點} \\ 0, & \text{ow} \end{cases}$
y_{kijt}	$= \begin{cases} 1, & \text{第 } k \text{ 台車在時刻 } t \text{ 由第 } i \text{ 個事故點派去醫院 } j \\ 0, & \text{ow} \end{cases}$
h_{kjt}	$= \begin{cases} 1, & \text{第 } k \text{ 台車在時刻 } t \text{ 留在醫院 } j \\ 0, & \text{ow} \end{cases}$
e_{kit}	$= \begin{cases} 1, & \text{第 } k \text{ 台車在時刻 } t \text{ 留在事故點 } i \\ 0, & \text{ow} \end{cases}$
Ih_{kj}	$= \begin{cases} 1, & \text{第 } k \text{ 台車由醫院 } j \text{ 出發} \\ 0, & \text{ow} \end{cases}$
p_{kijt}	第 k 台車在時刻 t 由事故點 i 到醫院 j 所載的傷患數目
RP_{it}	事故點 i 在時刻 t 所剩的傷患數量

4.2.2 模式說明

目標式

$$\text{Min} \quad \sum_{k=1}^K \sum_{t=1}^T \sum_{i \in I} \sum_{j \in J} (t + d_{ij}) y_{kijt} \quad (4.1)$$

目標式(4.1)考慮所有在時刻 t 由事故點派遣至醫院的救護車，計算其抵達醫院的時刻後加總，即得到傷患總等時間，其中 $t + d_{ij}$ 為救護車抵達醫院的時刻。

限制式

$$p_{kijt} \leq y_{kijt} \quad \forall k = 1, \dots, K, i = 1, \dots, I, j = 1, \dots, J, t = 0, 1, \dots, T \quad (4.2)$$

式(4.2)限制每台救護車載送傷患的數量限制，每台車最多只可載一個傷患。

$$\sum_{j=1}^J Ih_{kj} = 1 \quad \forall k = 1, \dots, K \quad (4.3)$$

$$\sum_{k=1}^K \sum_{j=1}^J Ih_{kj} = K \quad (4.4)$$

式(4.3)、式(4.4)為決定救護車的起始醫院之限制式，式(4.3)限制每台救護車必須放置於某一間醫院，式(4.4)限制總共有 K 台救護車被放置於醫院。

$$h_{kj(t-1)} + \sum_{i=1}^I \sum_{t' \in A_i} y_{kijt'} = h_{kjt} + \sum_{i=1}^I x_{kijt} \quad \forall k = 1, \dots, K, j = 1, \dots, J, t \in A_j \quad (4.5)$$

式(4.5)為救護車在醫院的流量限制，對每個醫院節點而言，救護車流入量等於救護車流出量。

$$e_{ki(t-1)} + \sum_{j=1}^J \sum_{t' \in A_j} x_{kijt'} = e_{kit} + \sum_{j=1}^J y_{kijt} \quad \forall k = 1, \dots, K, i = 1, \dots, I, t \in A_i \quad (4.6)$$

式(4.6)為救護車在事故點的流量限制，對每個事故點的節點而言，救護車流入量等於救護車流出量。

$$RP_{i(t-1)} = RP_{it} + \sum_{k=1}^K \sum_{j=1}^J \sum_{t' \in A_j} p_{kijt'} \quad \forall i = 1, \dots, I, t = 0, \dots, T \quad (4.7)$$

式(4.7)為傷患在傷患事故點的流量平衡限制，對每個傷患事故點的節點而言，傷患的流入量等於傷患的流出量。

$$x_{kijt} \in \{0, 1\} \quad \forall k = 1, \dots, K, i = 1, \dots, I, j = 1, \dots, J, t = 0, 1, \dots, T \quad (4.8)$$

$$y_{kijt} \in \{0, 1\} \quad \forall k = 1, \dots, K, i = 1, \dots, I, j = 1, \dots, J, t = 0, 1, \dots, T \quad (4.9)$$

$$h_{kjt} \in \{0, 1\} \quad \forall k = 1, \dots, K, j = 1, \dots, J, t = 0, 1, \dots, T \quad (4.10)$$

$$e_{kit} \in \{0, 1\} \quad \forall k = 1, \dots, K, i = 1, \dots, I, t = 0, 1, \dots, T \quad (4.11)$$

$$Ih_{kj} \in \{0, 1\} \quad \forall k = 1, \dots, K, j = 1, \dots, J \quad (4.12)$$

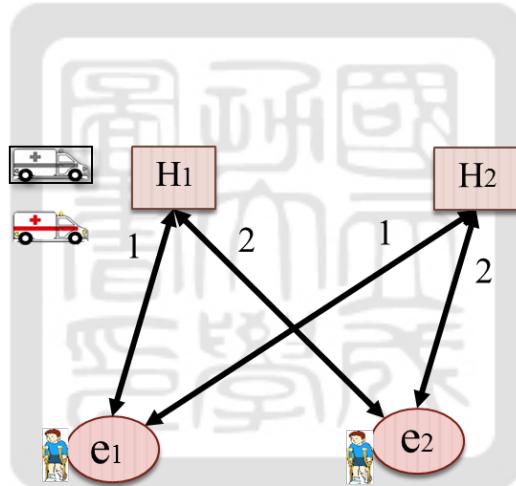
$$p_{kijt} \geq 0 \quad \forall k = 1, \dots, K, i = 1, \dots, I, j = 1, \dots, J, t = 0, 1, \dots, T \quad (4.13)$$

$$RP_{it} \geq 0 \quad \forall i = 1, \dots, I, t = 0, 1, \dots, T \quad (4.14)$$

式(4.8)~式(4.14)為本模式的非負限制，其中式(4.8)~式(4.12)為二元變數，式(4.13)、式(4.14)為非負之實數。

4.2.3 範例說明

以圖 4.1 為範例，說明時空網路圖模式，在本例中假設總共有 2 台救護車可供使用 ($K = 2$)，醫院與事故點數量皆為 2 ($J = I = 2$)，而每個事故點的傷患人數皆是 1 個傷患 ($RP_{10} = RP_{20} = 1$)。



首先，依照醫院與事故點的距離計算出網路圖的 $T = 8$ ，時空網路圖可分為「救護車網路圖」與「傷患網路圖」兩種，救護車網路圖以一台救護車當作一「層」網路，每一層的救護車路徑網路圖彼此以事故點節點與醫院節點將相連，為了方便解釋，本研究將事故點節點連接與醫院節點連接分開來繪製，如圖 4.2、圖 4.3 所示。

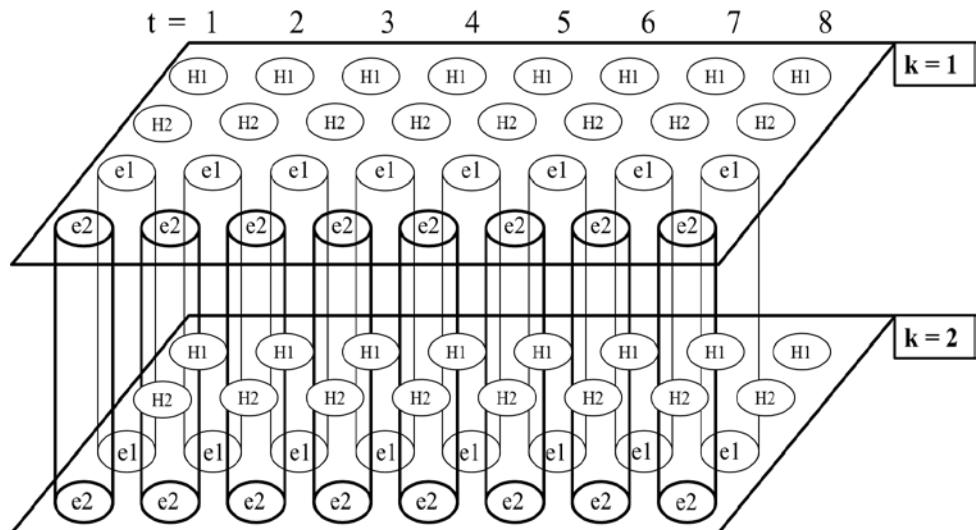


圖 4.2 救護車路徑網路連接圖(一)

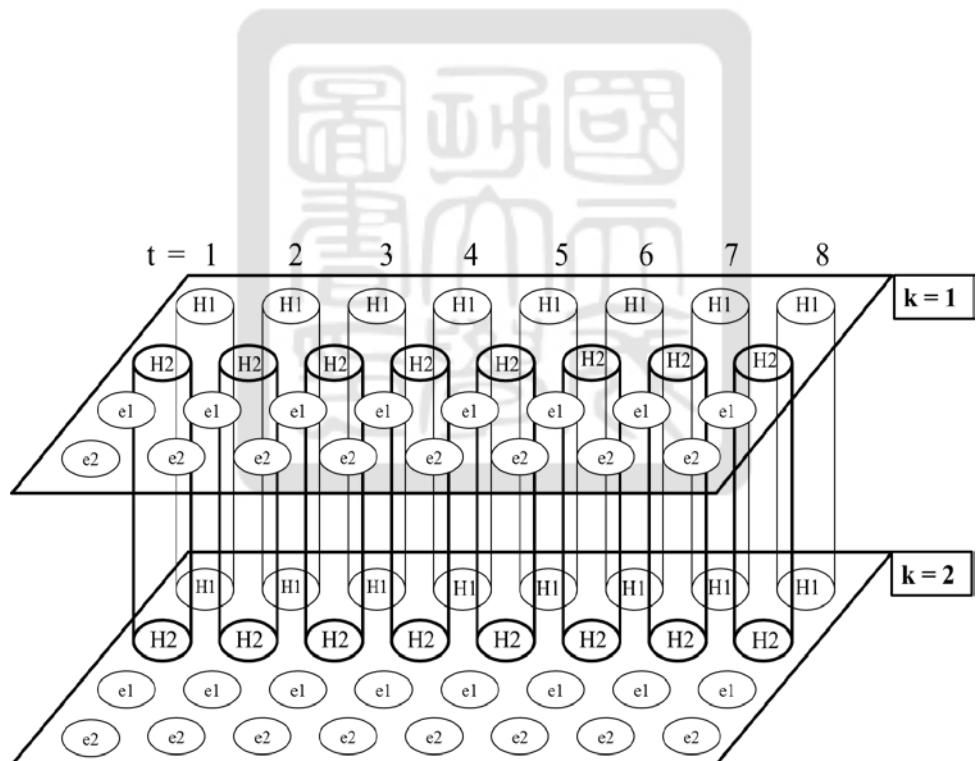


圖 4.3 救護車路徑網路連接圖(二)

每一台車的時空網路圖如圖 4.4 所示，為了決定救護車的初始醫院，網路圖中定義一個虛擬的救護車節點(Dummy)，以虛線表示之，並以節線連結至兩間醫院。圖中醫院之間的節線以及事故點之間的節線分別代表救護車駐留於醫院與駐留於事故點，而醫院與事故點的節線連接則是依照彼此之間的距離連接，此外，因為 $T = 8$ ，因此所有傷患最晚必須在時刻 $T = 8$ 時送至醫院。

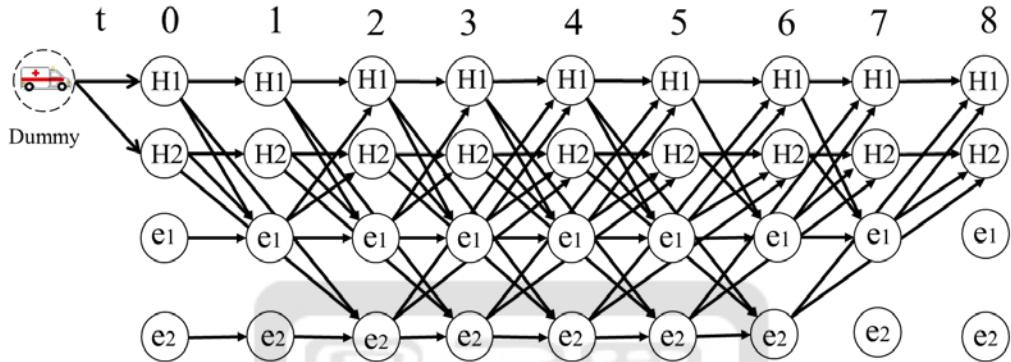


圖 4.4 單一救護車網路圖

圖 4.5 為傷患流向網路圖，此例中每個事故點皆有 1 個傷患流入，事故點之間的節線表示傷患滯留於事故點，事故點與醫院之間的節線代表傷患被載送至醫院。

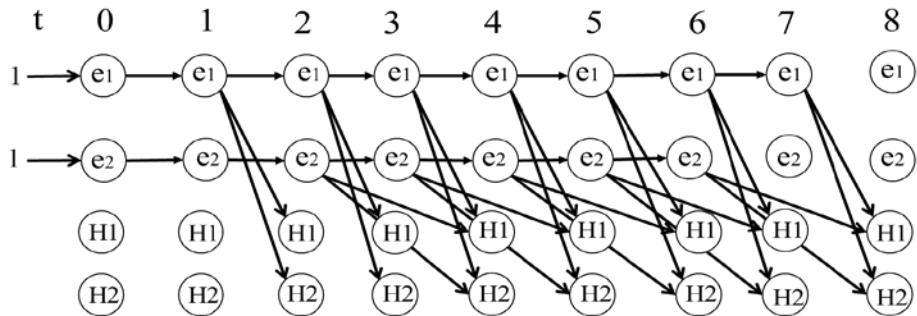


圖 4.5 傷患流向網路圖

4.3 順序式救護車派遣之整數規劃模式(AD_s^{FL})

4.3.1 基本概念

以時空網路圖的方法建立模式時，網路圖的大小會因為不同節點之間的距離而有影響，就算傷患人數不多，但是當距離過大時，一樣會導致網路圖過大，進而影響求解時間。有別於時空網路圖的模式建立方式，本章提出可彈性調整初始發車處之順序式救護車派遣模式(AD_s^{FL})，係以不同的觀點建構求解模式，不同於時空網路圖在各時刻 t 決定救護車的路徑， AD_s^{FL} 針對救護車的「每一趟」，決定派遣的路徑，而本模式之所以取名為「順序式救護車派遣」，是因為傷患抵達醫院的時刻完全由救護車的派遣順序與每趟指派的事故點來決定，因此不同的派遣順序會有不同的傷患抵達時刻，而相較於時空網路圖的模式， AD_s^{FL} 的救護車派遣模式之求解時間不會受到醫院與事故點之間的距離影響。

在 AD_s^{FL} 中，本研究除了假設救護車編號為 $k = 1 \sim K$ 、事故點編號為 $l = 1 \sim ne$ 、醫院編號為 $j = 1 \sim J$ 之外，另外也將傷患編號為 $i = 1 \sim P$ ，而傷患的編號方式將會依照事故點的編號的大小由小到大排序，比如說 1 號事故點與 2 號事故點各有 2 個與 3 個傷患，則 1 號事故點的傷患編號為 1 號與 2 號，而 2 號事故點的傷患編號為 3、4、5 號，此外，傷患到達醫院的時刻則會依照救護車所在醫院與第 v 趟所選擇的傷患而有不同，因此本研究將決定第 k 台救護車在第 v 趟的派遣路徑，並最小化傷患總等候時間。

而在救護車派遣的趟數方面，因為我們無法得知每台救護車分別需要多少趟才能將傷患送完，所以本研究假設每一台車的派遣趟數皆為最高，不過這樣將會出現所有救護車趟數總和與總傷患數目不合的情況，故本研究在各醫院旁邊加入一個距離大小為 0 的虛擬事故點，並依照缺少的傷患數目在虛擬事故點上設置相同數量的虛擬傷患，同時，將虛擬傷患的載送限制條件設為可送與可不送，如此一來，不但能使救護車派遣的趟數與傷患的數量相當，同時亦不會影響到傷患總等候時間的計算。

4.3.2 參數與變數定義

下表 4.4 為相關的集合、索引介紹。

表 4.4 集合與索引介紹

符號	說明	符號	說明
k	救護車編號	j	醫院編號
v	第 k 台救護車派出的第 v 趟	K	救護車總數的集合
i	傷患編號	I	傷患總數的集合
l	事故點編號	J	醫院總數的集合

下表 4.5 為參數說明。

表 4.5 參數說明

參數符號	參數說明
d_{jl}^{For}	第 j 醫院到第 l 事故點的距離
d_{lj}^{Back}	第 l 事故點到第 j 醫院的距離
P	實際的傷患總數
ne	事故點的總數
$VirP$	虛擬的傷患數量
$Vire$	虛擬事故點的數量
V	每台救護車 k 派遣次數的最大值(v 的最大值)

下表 4.6 為變數說明。

表 4.6 變數說明

變數符號	定義說明
x_{kvji}	$= \begin{cases} 1, & \text{第 } k \text{ 台車在第 } v \text{ 順序由醫院 } j \text{ 派去載第 } i \text{ 位傷患} \\ 0, & \text{ow} \end{cases}$
y_{kvij}	$= \begin{cases} 1, & \text{第 } k \text{ 台車在第 } v \text{ 順序由將第 } i \text{ 位傷患送至醫院 } j \\ 0, & \text{ow} \end{cases}$
α_{kv}	第 k 台車在第 v 順位由醫院派出後，再次抵達醫院所花的時間

4.3.3 順序式模式說明

目標式

$$\text{Min } Z = \sum_{k=1}^K \sum_{v=1}^V (V - v + 1) \alpha_{kv} \quad (4.15)$$

式(4.15)為目標式，紀錄傷患的總等候時間。 α_{kv} 為救護車單趟(醫院派至事故點，再回到醫院)所花費的時間，而傷患總等候時間的計算必須加上前一趟所花費的時間，因此 $V - v + 1$ 代表累積的趟數的相加數目，更詳細的解釋將在範例說明中介紹。

限制式

$$\sum_{k=1}^K \sum_{v=1}^V \sum_{j=1}^J x_{kvji} = 1 \quad \forall i = 1, 2, \dots, P \quad (4.16)$$

$$\sum_{k=1}^K \sum_{v=1}^V \sum_{j=1}^J y_{kvij} = 1 \quad \forall i = 1, 2, \dots, P \quad (4.17)$$

$$\sum_{k=1}^K \sum_{v=1}^V \sum_{j=1}^J x_{kvji} \leq 1 \quad \forall i = P + 1, P + 2, \dots, P + VirP \quad (4.18)$$

$$\sum_{k=1}^K \sum_{v=1}^V \sum_{j=1}^J y_{kvij} \leq 1 \quad \forall i = P + 1, P + 2, \dots, P + VirP \quad (4.19)$$

對實際的傷患而言，式(4.16)、式(4.17)分別為救護車由醫院駛向事故點與事故點駛向醫院載送傷患的限制，救護車必須載送每一個實際的傷患。對虛擬傷患而言，式(4.18)、式(4.19)限制救護車的去程及回程中可選擇載送傷患亦可不載。

$$\sum_{j=1}^J \sum_{i=1}^{VirP} x_{kyji} \leq 1 \quad \forall k = 1, \dots, K, v = 1, \dots, V \quad (4.20)$$

$$\sum_{j=1}^J \sum_{i=1}^{VirP} y_{kvij} \leq 1 \quad \forall k = 1, \dots, K, v = 1, \dots, V \quad (4.21)$$

對每一台救護車的每一趟來說，式(4.20)、式(4.21)限制救護車於去程及回程時，可以選擇載傷患或不載傷患。

$$\sum_{j=1}^J x_{kvji} = \sum_{j=1}^J y_{kvij} \quad \forall k = 1, \dots, K, v = 1, \dots, V, i = 1, \dots, VirP + P \quad (4.22)$$

式(4.22)限制救護車去回的關係式，當救護車由醫院駛至某事故點，救護車必須由該事故點駛向下一個醫院。

$$\sum_{i=1}^{VirP+P} y_{kvij} = \sum_{i=1}^{VirP+P} x_{k(v+1)ji} \quad \forall k = 1, \dots, K, j = 1, \dots, J, v = 1, \dots, V - 1 \quad (4.23)$$

式(4.23)限制救護車出發的醫院，當救護車由事故點行駛至醫院，則下一趟出發時，必須由同樣的醫院出發。

$$\begin{aligned} \alpha_{kv} = & \sum_{j=1}^J \sum_{l=1}^{ne+Vire} \sum_{i \in Set_i_l} d_{jl}^{For} x_{kyji} \\ & + \sum_{j=1}^J \sum_{l=1}^{ne+Vire} \sum_{i \in Set_l} d_{lj}^{Back} y_{kvij} \quad \forall k = 1, \dots, K, v = 1, \dots, V \end{aligned} \quad (4.24)$$

式(4.24)依照事故點與醫院之間的距離，定義第 k 台救護車在第 v 趟花費的時間。

4.3.4 範例說明

為了提供詳細的解釋，本研究提供範例說明。範例說明如圖 4.6 所示，假設救護車(k)、醫院(H)、事故點(e)皆為 2 個，虛擬事故點(e3、e4)的數目則是與醫院數量相等，每個事故點傷患人數為 2 位，傷患的編號放置於傷患圖示下面，實

實際傷患的編號為 1 號~4 號，而根據救護車數量與傷患數目計算出的虛擬傷患總共是兩個，虛擬事故點 3 與虛擬事故點 4 分別有 1 位虛擬傷患，編號分別是 5 號及 6 號，此外，虛擬事故點與實際事故點的距離如圖所呈現。

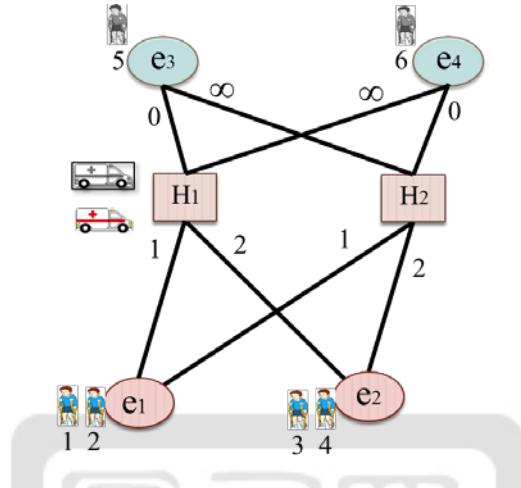


圖 4.6 範例說明示意圖

說明變數前，先定義去程代表救護車由醫院駛向事故點，回程代表救護車由事故點載送傷患至醫院。為了方便說明，本研究將模式中的去、回程的二元變數求解結果表格化，如圖 4.7 與圖 4.8 分別代表去程與回程的救護車指派，在去程指派表中，空格中的數字若等於 1 代表第 i 位傷患被編號為 k 的救護車第 v 趟，且是由第 j 醫院所派出的救護車；而回程指派表代表第 i 位傷患編號為 k 的救護車第 v 趟送至第 j 醫院治療。

	k	1			2				
v	1	2	3	1	2	3			
i	j	1	2	1	2	1	2	1	2
1			1						
2						1			
3							1		
4				1					
5					1				
6	1								

圖 4.7 去程指派表

	k	1			2				
v	1	2	3	1	2	3			
i	j	1	2	1	2	1	2	1	2
1			1						
2						1			
3							1		
4				1					
5					1				
6	1								

圖 4.8 回程指派表

圖 4.7、圖 4.8 的指派表中，1 號救護車與 2 號救護車的第一趟皆是派遣至虛擬事故點載送虛擬傷患且虛擬傷患抵達醫院時刻皆為 0，因此可將虛擬傷患的等候時間忽略，故在計算傷患總等候時間時，可將兩台救護車的第一趟派遣忽略，而實際的第一趟派遣則是由兩台救護車的第二趟開始算起。圖 4.9 為傷患抵達醫院的時刻記錄圖，詳細記錄了 4 位傷患抵達醫院的時刻，同時也記錄了每一位傷患所搭乘的救護車，例如 1 號傷患與 3 號傷患分別是由第 1 台的救護車的第一趟與第二趟載送；因為 1 號傷患與 3 號傷患分別處於兩個不同的事故點，所以兩位傷患與醫院的距離並不相等，如圖所示，1 號傷患與 2 號傷患的去、回程總時間為 2 單位時間($\alpha_{k_1} = 2$)，而 3 號傷患與 4 號傷患去、回程總時間為 4 單位時間($\alpha_{k_2} = 4$)。計算傷患的等候時間時，除了傷患本身去、回程的時間外，還需要加上該台救護車「前一趟」所花的時間，舉例來說，1 號救護車在第一趟($v=1$)與第二趟($v=2$)分別載送了 1 號傷患與 3 號傷患，再加上救護車在時刻等於零時出發，所以 1 號救護車載的 1 號傷患抵達醫院的時刻為 $0 + \alpha_{11} = 2$ ，而救護車的第二趟必須以上一趟的抵達時刻為基準往上加，因此 1 號救護車所載的 2 號傷患的抵達時刻為 $\alpha_{11} + \alpha_{12} = 2 + 4 = 6$ 。

最後將所有傷患的等候時間加總即為傷患的總等候時間(即為圖上面積)：

$$2\alpha_{11} + \alpha_{12} + 2\alpha_{21} + \alpha_{22} = 16$$

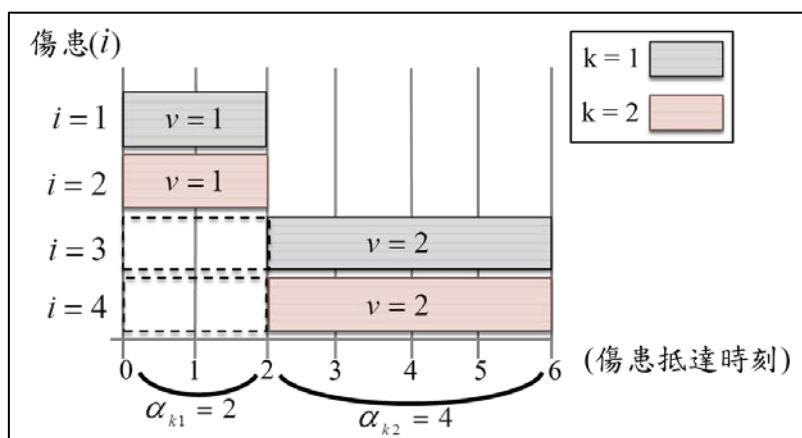


圖 4.9 傷患抵達時刻記錄圖

4.4 貪婪演算法

4.4.1 基本想法

當問題只有一間醫院時，本研究發現了特別的派遣規則，且提出相關證明。下圖 4.10 為問題描述，假設救護車共有 k 台、1 間醫院(H)與 2 個事故點(e1、e2)，其中事故點傷患數目分別是 P_1 與 P_2 ，而事故點距離醫院分別是 d_1 與 d_2 ，在此定義 e1 是距離醫院較近的事故點，e2 是距離醫院較遠的事故點，因此 $d_1 < d_2$ 。在給定參數值之後，期望找到一個派遣方式，使傷患總等候時間最小化，而在此所提的派遣方式指的是救護車第一趟派遣的事故點與各事故點派遣的救護車數量。

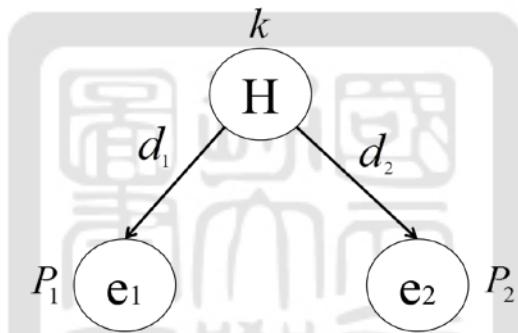


圖 4.10 基本想法範例圖

最初求解此問題時，本研究使用列舉法，將所有救護車可能的第一趟派遣方式窮舉出來，並找出最小的傷患總等候時間。列舉法發現，在傷患總數大於救護車總數的情況下，救護車第一趟的派遣方式皆是將所有救護車派送至距離醫院最近的事故點，而發現此規則後，本研究試圖提出相關的數學證明。

為了提出相關數學證明，以下是本問題相關假設與定義：

1. 定義 e1 距離醫院較近
2. 兩事故點的傷患數目皆高於救護車總數
3. 救護車須將事故點的傷患全數送完，才能指派至下一個事故點
4. 救護車派至事故點的數量與事故點傷患數量剛好可以整除

下表 4.7 定義相關的參數與符號：

表 4.7 符號說明表

符號	定義說明
k	救護車總數
x_i	救護車第一趟派至第 i 事故點的數量
D_i	救護車派至第 i 事故點來回一趟的時間($D_i = 2d_i$)
$e_i(\tilde{x})$	第 i 事故點的傷患全送至醫院後，所花的傷患總等候時間
$N(\tilde{x})$	表有多餘救護車可供支援的週期
$f(x)$	傷患總等候時間

對 e1 與 e2 的傷患而言，各事故點的傷患等候時間可以下式(4.25)、(4.26)表示。

$$e_1(\tilde{x}) = D_1x_1 + 2D_1x_1 + \dots + P_1D_1x_1 \quad (4.25)$$

$$e_2(\tilde{x}) = D_2x_2 + 2D_2x_2 + \dots + \frac{P_1D_1}{x_1}x_2 + N(x) \quad (4.26)$$

最小化傷患總等候時間問題可以下列非線性規劃模式表示：

$$\text{Min} \quad f(\tilde{x}) = e_1(\tilde{x}) + e_2(\tilde{x}) \quad (4.27)$$

$$\text{s.t.} \quad x_1 + x_2 \leq k \quad (4.28)$$

$$x_1, x_2 \geq 0 \quad (4.29)$$

式(4.27)為目標值傷患總等候時間，式(4.28)為救護車總數限制，最後式(4.29)為救護車數量的非負限制。

雖然此模式無法以最佳化軟體 Gurobi 求解，但是將目標式展開並將 $x_2 = k - x_1$ 代入目標式後，即可對其微分尋找區域極小值，如下式(4.30)與式(4.31)。因為 $D_2 \geq D_1$ ，且方程式符號為負，因此可以知道目標式的一階微分小於零，如

式(4.32)所示，亦即此方程式為遞減函數，在 x_1 不等於零的情況下， x_1 越大，傷患總等候時間將隨之增大，而當 x_1 到達最大值 $x_1 = k$ 時將可得到最小的傷患總等候時間。

$$f(x) = \frac{P_1(kD_1D_2 - kD_1) + x_1[D_1^2P_1^2 + D_2^2P_2^2 + P_2(kD_2 + 2D_1D_2P_1) + kD_1D_2P_1]}{2kD_2x_1} \quad (4.30)$$

$$f'(x) = -\frac{D_1P_1^2(D_2 - D_1)}{2D_2x_1^2} \quad (4.31)$$

$$-\frac{D_1P_1^2(D_2 - D_1)}{2D_2x_1^2} \leq 0 \quad (4.32)$$

將實際的數值代入方目標式後，可繪製出如下圖 4.11 的函數圖形，可明顯看出當 $x_1 > 0$ 時，函數為遞減函數，因此當 $x_1 = k$ 時，函數有最小值。

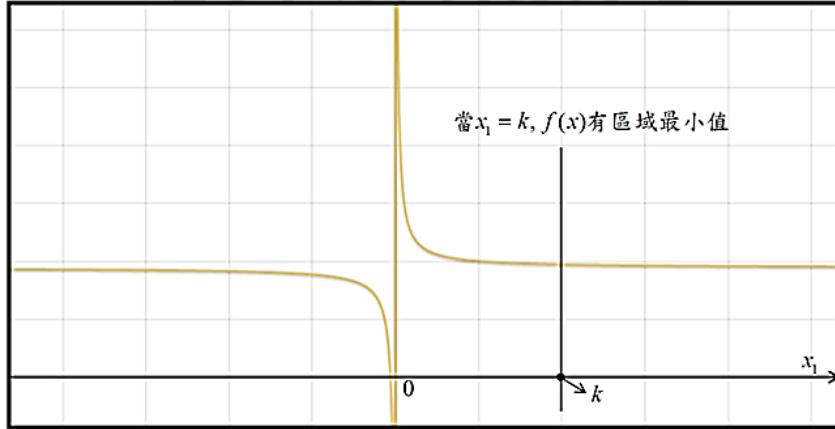


圖 4.11 傷患總等候時間函數圖形

4.4.2 GDA_{f1}^{FL} 、 GDA_{f2}^{FL} 演算法說明

發現了單一醫院的派遣規則後，本節以此為基礎提出 GDA_{f1}^{FL} 與 GDA_{f2}^{FL} 兩種貪婪演算法，其中 GDA 代表 Greedy Dispatching Algorithm；在右下角的 f 皆代表 a fleet of ambulance，而編號 1、2 代表派法 1 與派法 2。這兩種貪婪演算法的概念皆是將所有救護車送往同一個事故點，而當醫院只有一個時，兩種演算法的救

護車派送路徑完全一樣，而當醫院數量大於一個時，兩種演算方法除了初始指派的事故點一樣之外，之後派遣的醫院與事故點則會因為派遣的準則不同而有不一樣的路徑。兩種演算法最初皆是將所有救護車擺放於同一間醫院，而該醫院與事故點間的距離是所有事故點距離中最近的一間醫院。本節將依序介紹 GDA_{f1}^{FL} 、 GDA_{f2}^{FL} 的流程與範例說明

GDA_{f1}^{FL} 演算法

GDA_{f1}^{FL} 演算法派遣的「派遣規則」如下：

1. 救護車在醫院時，派遣至當下最近事故點。
2. 救護車在事故點時，將救護車派至距離該事故點最近的醫院。

圖 4.12 為 GDA_{f1}^{FL} 演算法的流程圖：

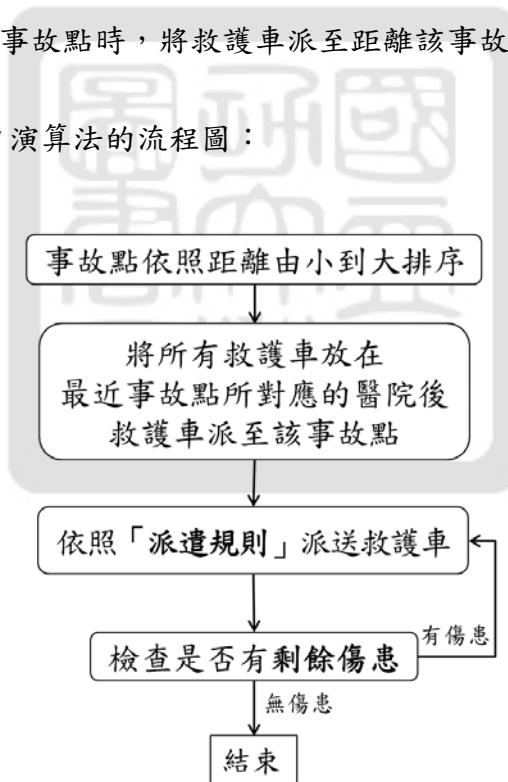


圖 4.12 GDA_{f1}^{FL} 演算步驟流程圖

範例說明

範例如圖 4.13 所示，假設總共有 2 台救護車、2 間醫院、2 個事故點與 6 位傷患平均分部於各個事故點，一個事故點有 2 位傷患，事故點與醫院的距離如圖所示。

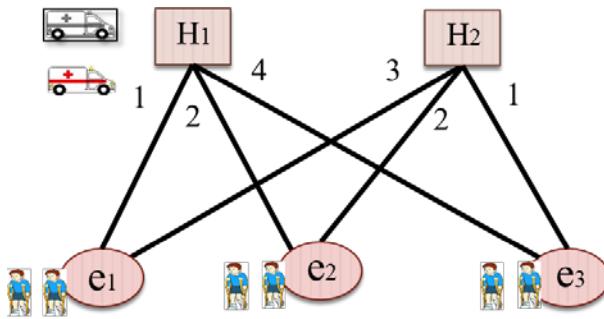


圖 4.13 GDA_{f1}^{FL} 範例說明示意圖

GDA_{f1}^{FL} 演算法的救護車派遣路徑如下圖 4.14 ~ 圖 4.20，在此例中排序最近的醫院分別為 H1 到 e1 與 H2 到 e3，在此選擇編號最小的醫院 H1，因此所有救護車一開始派至事故點 e1，如圖 4.14 ~ 圖 4.16；當 $t = 4$ 時選擇對 H1 而言最近的事故點 e2，然後再選擇距離 e2 最近的醫院 H2，如圖 4.17、圖 4.18；到 H2 之後選擇最近的事故點 e3，最後再將傷患送回至最近醫院 H2，如圖 4.19 與圖 4.20，最後結束派送。

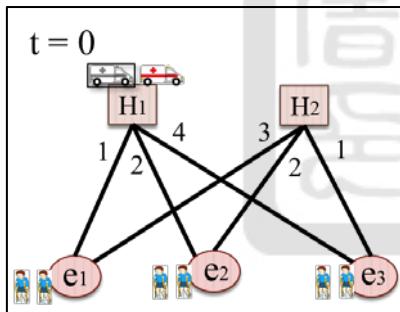


圖 4.14 GDA_{f1}^{FL} 演算流程圖

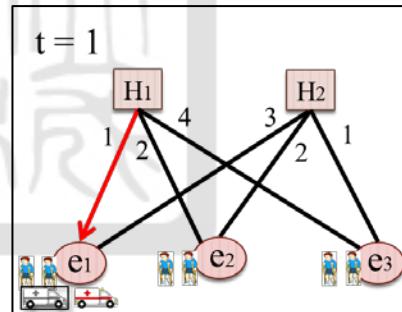


圖 4.15 GDA_{f1}^{FL} 演算流程圖

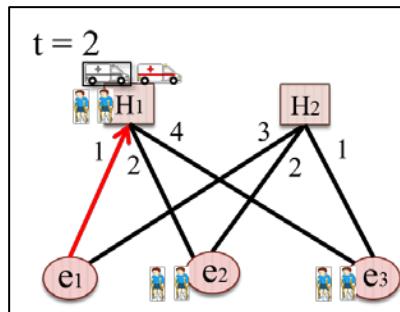


圖 4.16 GDA_{f1}^{FL} 演算流程圖

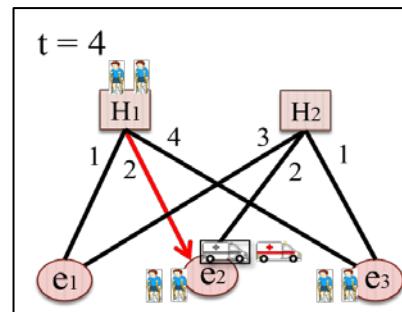


圖 4.17 GDA_{f1}^{FL} 演算流程圖

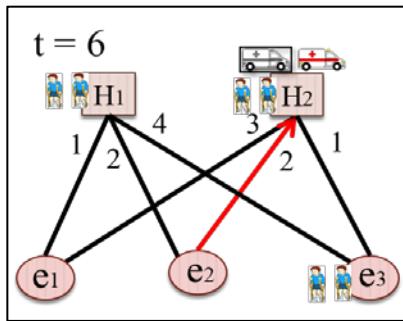


圖 4.18 GDA_{f1}^{FL} 演算流程圖

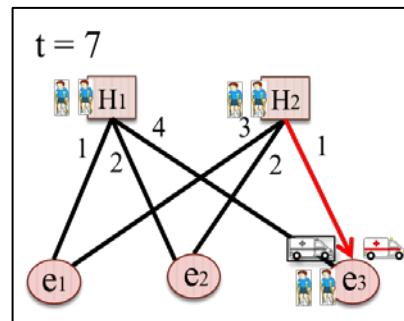


圖 4.19 GDA_{f1}^{FL} 演算流程圖

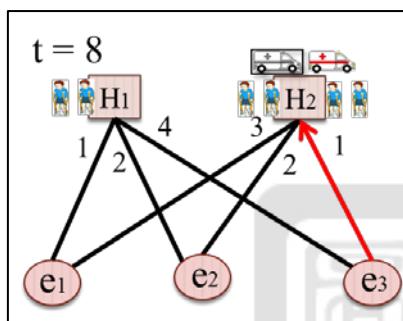
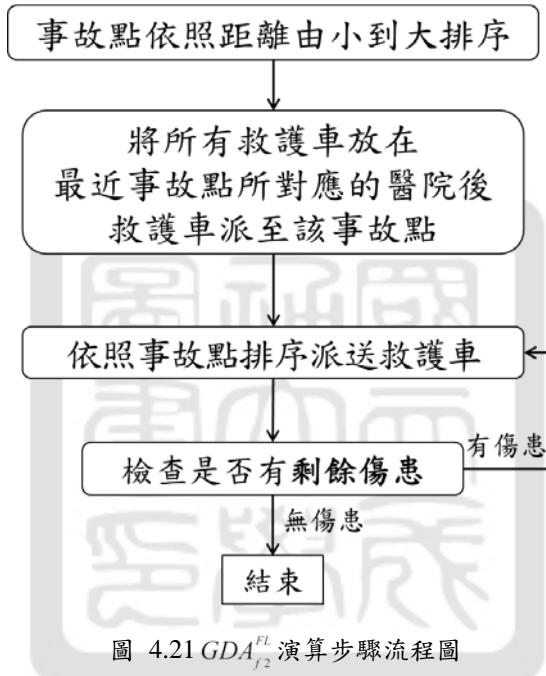


圖 4.20 GDA_{f1}^{FL} 演算流程圖

GDA_{f2}^{FL} 演算法

GDA_{f2}^{FL} 演算法派遣規則與 GDA_{f1}^{FL} 的救護車派遣差異在於， GDA_{f2}^{FL} 要指派的事故點已經事先決定好，事故點將依照與醫院的距離由近到遠排序，而救護車的派送順序將依照此排序依序派遣，而救護車在事故點時，將會派至最近的醫院。

下圖 4.21 為 GDA_{f2}^{FL} 的派遣流程：



範例說明

此範例說明沿用圖 4.13 的範例，藉此可看出 GDA_{f1}^{FL} 與 GDA_{f2}^{FL} 的差異。圖 4.22 ~ 圖 4.28 為救護車派遣流程，在此例子中，事故點依照距離的排序分別是 e1、e3、e2，故救護車將會依照此順序派車。一開始全把救護車擺在編號最小且最近的醫院 H1，並將救護車派至 e1 再送回 H1，如圖 4.22 ~ 圖 4.24；救護車第二趟派出時，選擇第二近的事故點 e3，到達事故點後再選擇最近的醫院 H2，如圖 4.25、圖 4.26；最後救護車選擇第三近的事故點 e2，到事故點後兩間醫院距離一樣，則選擇編號較小的 H1，如圖 4.27 與 4.28。

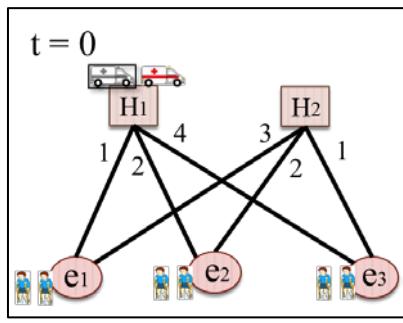


圖 4.22 GDA_{f2}^{FL} 演算流程圖

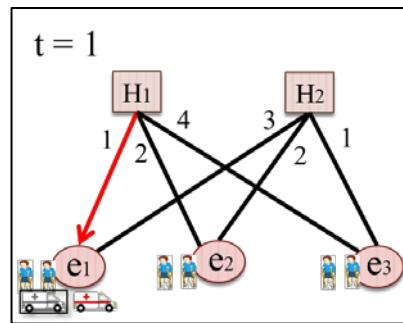


圖 4.23 GDA_{f2}^{FL} 演算流程圖

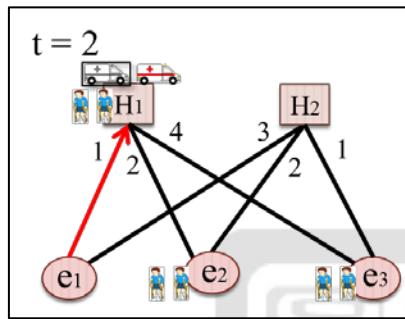


圖 4.24 GDA_{f2}^{FL} 演算流程圖

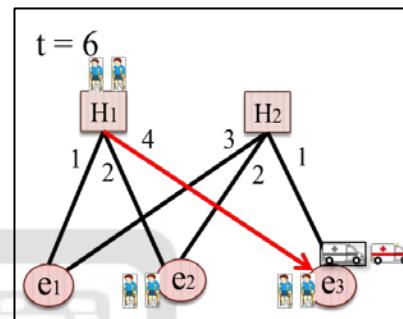


圖 4.25 GDA_{f2}^{FL} 演算流程圖

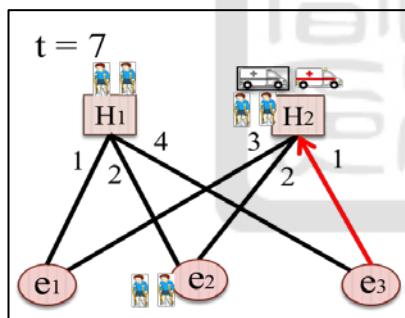


圖 4.26 GDA_{f2}^{FL} 演算流程圖

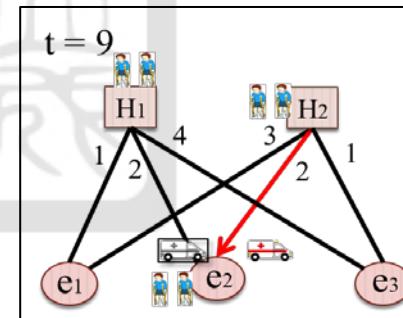


圖 4.27 GDA_{f2}^{FL} 演算流程圖

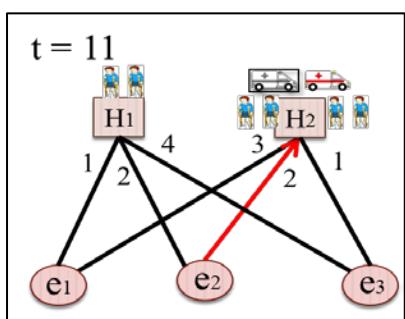


圖 4.28 GDA_{f2}^{FL} 演算流程圖

4.5 數值分析

4.5.1 數據產生方式

本章參數產生的方式與第三章相同，一樣是在自行定義的座標軸上隨機產生不同的參數，差別在於本章不需產生消防站的節點，其餘的事故點、醫院與事故點距離皆在座標上隨機產生並以直線距離計算彼此的距離。

下表 4.8 與表 4.9 為測試的 24 種 Case，依救護車數量(K)、總傷患數(P)、事故點數量(e)與醫院數量(H)來分類，而每一個 Case 測試了 40 個例子，其中救護車數量分為小(2 ~ 5 台車)與大(6 ~ 9 台車)兩種情況，總傷患數分為小(總傷患小於 30 人)與大(總傷患大於 30 人)兩種情況，事故點分為小(2 ~ 4 個事故點)與大(5 ~ 7 個事故點)兩種情況，醫院分為小(1 間醫院)、中(2、3 間醫院)與大(4、5 間醫院)三種情況。

表 4.8 測試 Case 分類表(一)

Case (K,P,e,H)	救護車(K)	傷患總數(P)	事故點(e)	醫院數量(H)
Case 1 (小,小,小,小)	2 ~ 5	1 ~ 30	2、3、4	1
Case 2 (小,小,小,中)	2 ~ 5	1 ~ 30	2、3、4	2,3
Case 3 (小,小,小,大)	2 ~ 5	1 ~ 30	2、3、4	4,5
Case 4 (小,小,大,小)	2 ~ 5	1 ~ 30	5、6、7	1
Case 5 (小,小,大,中)	2 ~ 5	1 ~ 30	5、6、7	2,3
Case 6 (小,小,大,大)	2 ~ 5	1 ~ 30	5、6、7	4,5
Case 7 (小,大,小,小)	2 ~ 5	31 ~ 60	2、3、4	1
Case 8 (小,大,小,中)	2 ~ 5	31 ~ 60	2、3、4	2,3
Case 9 (小,大,小,大)	2 ~ 5	31 ~ 60	2、3、4	4,5
Case 10 (小,大,大,小)	2 ~ 5	31 ~ 60	5、6、7	1
Case 11 (小,大,大,中)	2 ~ 5	31 ~ 60	5、6、7	2,3
Case 12 (小,大,大,大)	2 ~ 5	31 ~ 60	5、6、7	4,5

表 4.9 測試 Case 分類表(二)

Case (K,P,e,H)	救護車(K)	傷患總數(P)	事故點(e)	醫院數量(H)
Case 13 (大,小,小,小)	6 ~ 9	1 ~ 30	2、3、4	1
Case 14 (大,小,小,中)	6 ~ 9	1 ~ 30	2、3、4	2,3
Case 15 (大,小,小,大)	6 ~ 9	1 ~ 30	2、3、4	4,5
Case 16 (大,小,大,小)	6 ~ 9	1 ~ 30	5、6、7	1
Case 17 (大,小,大,中)	6 ~ 9	1 ~ 30	5、6、7	2,3
Case 18 (大,小,大,大)	6 ~ 9	1 ~ 30	5、6、7	4,5
Case 19 (大,大,小,小)	6 ~ 9	31 ~ 60	2、3、4	1
Case 20 (大,大,小,中)	6 ~ 9	31 ~ 60	2、3、4	2,3
Case 21 (大,大,小,大)	6 ~ 9	31 ~ 60	2、3、4	4,5
Case 22 (大,大,大,小)	6 ~ 9	31 ~ 60	5、6、7	1
Case 23 (大,大,大,中)	6 ~ 9	31 ~ 60	5、6、7	2,3
Case 24 (大,大,大,大)	6 ~ 9	31 ~ 60	5、6、7	4,5

4.5.2 數值測試

數值測試中，若使用原來的模式 AD_S^{FL} 進行求解，因為每台救護車派遣的趟數是未知的，因此當傷患、救護車數量比較多的時候，為了將每一台救護車的趟數補到一樣的數量，會使得每一台車的派遣趟數與虛擬傷患的數量大幅增加，如此一來將會使決策變數的數量上升，導致求解效率大幅降低，因此本研究針對救護車的派遣趟數對模式 AD_S^{FL} 做出改變。

為了降低決策變數的數量，本研究針對救護車派遣趟數的計算方式，將模式 AD_S^{FL} 改成 AD_{SAT}^{FL} 與 AD_{SFT}^{FL} ，其中 AD_{SAT}^{FL} 右下角的 SAT 代表 Sequential Average Trip，也就是依據傷患的數量平均分配給每一台救護車，因此每台救護車派遣的趟數都差不多，例如：若總共有 10 位傷患、2 台車，則平均每台車派出 $10 \div 2 = 5$ 趟，若無法整除，則將多餘的傷患依照救護車編號依序分配，如此一來可降低二元變數的數量，期望能夠降低求解時間；而 AD_{SFT}^{FL} 中右下角的 SFT 代表 Sequential Fixed Trip，也就是依據每台救護車平均派遣的趟數再加上固定的 3 趟後，再測試此演算方法的求解時間。

首先針對模式 AD_{TS}^{FL} 、 AD_{SAT}^{FL} 與 AD_{SFT}^{FL} 的平均求解時間做比較，如表 4.10 為各 Case 的平均求解時間，表格中可以發現模式 AD_{TS}^{FL} 的求解時間皆高於 AD_{SAT}^{FL} 與 AD_{SFT}^{FL} ，而在 Case3 、 Case6 、 Case9 、 Case12 、 Case18 、 Case24 中最為明顯，這些 Case 共通的特性為，醫院數量的分類皆為「大」，而當醫院數量增多時，時空網路圖的節點數量亦會隨之上升，因此影響模式 AD_{TS}^{FL} 有的求解時間，而在這其中 Case12 的求解時間最長，模式 AD_{TS}^{FL} 、 AD_{SAT}^{FL} 與 AD_{SFT}^{FL} 的求解時間分別為 300.2189 秒、 31.8094 秒與 20.3741 秒，若與 Case24 比較後，可以發現兩者唯一不同的是救護車的數量，當救護車數量少的時候，將使得求解時間大幅上升，因此可以得知救護車的數量亦會影響求解時間；而從表格中也可以發現，在相同的救護車數量、傷患數量與事故點數量(如 Case1~3 、 Case4~6 、 Case7~9 、 、 Case22~Case24)，醫院的數量將影響模式的求解時間，因為當醫院數量上升時，不但救護車擺放於醫院的可能性變多了，而且載送傷患時有更多的醫院可以選擇，導致變數與限制式增加，使得模式 AD_{TS}^{FL} 、 AD_{SAT}^{FL} 與 AD_{SFT}^{FL} 的求解時間都有增加的趨勢；另外表格中可以發現 AD_{SFT}^{FL} 的求解時間略高於 AD_{SAT}^{FL} 的求解時間，因為 AD_{SFT}^{FL} 所產生的變數比 AD_{SAT}^{FL} 還要多，使得 AD_{SFT}^{FL} 的求解時間略高於 AD_{SAT}^{FL} 。

表 4.10 平均求解時間 (單位：秒)

Case	AD_{TS}^{FL}	AD_{SAT}^{FL}	AD_{SFT}^{FL}
1	0.194411	0.01048	0.020961
2	1.854409	0.188103	0.390114
3	3.729532	0.672526	1.458864
4	0.636661	0.014001	0.027877
5	5.666608	0.402356	0.595891
6	10.88166	1.277304	2.483142
7	0.262873	0.027716	0.05486
8	1.660137	0.279558	0.488611
9	6.544179	1.497285	3.591274
10	2.043724	0.05111	0.094827
11	18.12139	1.687076	3.067577
12	300.2189	20.3741	31.8094
13	0.153259	0.010751	0.033419
14	1.057573	0.106069	0.488128
15	3.595278	0.196297	1.63295
16	0.587783	0.018752	0.046253
17	4.545593	0.201067	0.944054
18	19.11009	0.434581	2.357858
19	0.391897	0.034502	0.085256
20	1.688397	0.308484	0.97819
21	4.658745	0.983103	3.865599
22	1.60655	0.048211	0.116798
23	10.75758	2.082628	3.573059
24	36.94137	7.466037	10.71346

本研究在表 4.11 列出 AD_{SAT}^{FL} 、 AD_{SFT}^{FL} 、 GDA_{f1}^{FL} 與 GDA_{f2}^{FL} 24 個 Case 的平均求解 Gap(%)比較，表格中明顯可以看出 GDA_{f1}^{FL} 與 GDA_{f2}^{FL} 的求解品質比 AD_{SAT}^{FL} 、 AD_{SFT}^{FL} 還要差，但是當醫院只有 1 間時(Case1、Case4、Case7、.....、Case22)， GDA_{f1}^{FL} 與 GDA_{f2}^{FL} 皆能求得最佳解，但是求解 Gap 將會隨著醫院數量增多而增加，其中 GDA_{f2}^{FL} 的表現又比 GDA_{f1}^{FL} 還要差，因為 GDA_{f2}^{FL} 先將事故點排序後再指派救護車，但這種派遣方式並未考量後續的傷患載送，導致求解品質不佳。另外比較

AD_{SAT}^{FL} 與 AD_{SFT}^{FL} 的求解品質後發現， AD_{SFT}^{FL} 除了 Case24 是 0.01% 以外，其它 Gap 皆是 0%，所以 AD_{SFT}^{FL} 雖然在求解時間上略遜色於 AD_{SAT}^{FL} ，但求解時間不長而且求解品質非常好，因此是個不錯的求解方法；而醫院的數量對 AD_{SAT}^{FL} 的 Gap 也有影響，表格中可以發現，當醫院數量增加時， AD_{SAT}^{FL} 的求解 Gap 也隨之增加，但 AD_{SAT}^{FL} 的優點在於，就算是最差的 Gap，也是在 0.6% 以內，因此 AD_{SAT}^{FL} 也是個不錯的求解方法。

表 4.11 演算法與修改模式 Gap 比較表

Case	AD_{SAT}^{FL}	AD_{SFT}^{FL}	GDA_{f1}^{FL}	GDA_{f2}^{FL}
1	0.0%	0.00%	0.0%	0.0%
2	0.2%	0.00%	4.0%	5.1%
3	0.3%	0.00%	8.6%	9.6%
4	0.0%	0.00%	0.0%	0.0%
5	0.3%	0.00%	8.4%	11.4%
6	0.5%	0.00%	19.6%	25.8%
7	0.0%	0.00%	0.0%	0.0%
8	0.0%	0.00%	3.1%	3.2%
9	0.4%	0.00%	7.6%	7.9%
10	0.0%	0.00%	0.0%	0.0%
11	0.2%	0.00%	7.7%	9.9%
12	0.2%	0.00%	21.2%	29.2%
13	0.0%	0.00%	0.0%	0.0%
14	0.3%	0.00%	13.5%	14.4%
15	0.5%	0.00%	20.7%	29.3%
16	0.0%	0.00%	0.0%	0.0%
17	0.4%	0.00%	16.6%	27.6%
18	0.3%	0.00%	35.7%	49.2%
19	0.0%	0.00%	0.0%	0.0%
20	0.3%	0.00%	7.1%	10.3%
21	0.6%	0.00%	24.2%	42.1%
22	0.0%	0.00%	0.0%	0.0%
23	0.3%	0.00%	13.7%	21.0%
24	0.4%	0.01%	31.5%	52.5%

為了找出救護車放置於醫院的規則與比例，本研究以 10 台救護車、2 個事故點、2 間醫院與 30 位傷患當作測試參數，而為了測試不同距離對救護車放置數量的影響，我們設定的距離為分別為：1 單位時間、2 單位時間、3 單位時間與 4 單位時間。若事故點的傷患數量少於救護車總數，則會影響到救護車正確的放置比例，因此在測試時，本研究設定各個事故點的傷患數量不少於救護車總數，以確保測試的正確性。

表 4.12 為不同的測試 Case，在距離的欄位中，括號內的數字代表醫院與事故點的距離，其中左邊數過來的前兩個數字分別代表 H1 到 e1 與 H1 到 e2 的距離，而左邊數過來第三與第四個數字代表 H2 到 e1 與 H2 到 e2 的距離，舉例來說，如 Case1 中的(1,2,3,4)代表 H1 到 e1 為 1 單位時間，H1 到 e2 為 2 單位時間，H2 到 e1 為 3 單位時間，H2 到 e2 為 4 單位時間；而傷患的欄位中，括弧內左邊的數字代表 e1 的傷患人數，括弧內的右邊代表 e2 的傷患人數，e1 的傷患人數從 10 位一直到 20 位，反之，e2 的傷患人數則由 20 位遞減至 10 位，因此每個 Case 依照兩個事故點傷患的數量不同而有 11 個不同的測試例子。舉例來說，Case1 將在(1,2,3,4)的距離下，測試傷患數分別為(10,20)、(11,19)、……、(19,11)、(20,10)等 11 種不同的例子。

表 4.12 救護車放置規則測試 Case 表

Case	距離	傷患(P1,P2)
1	(1,2,3,4)	(10,20)~(20,10)
2	(1,2,4,3)	(10,20)~(20,10)
3	(1,3,2,4)	(10,20)~(20,10)
4	(1,3,4,2)	(10,20)~(20,10)
5	(1,4,2,3)	(10,20)~(20,10)
6	(1,4,3,2)	(10,20)~(20,10)
7	(2,1,3,4)	(10,20)~(20,10)
8	(2,1,4,3)	(10,20)~(20,10)
9	(2,3,1,4)	(10,20)~(20,10)
10	(2,3,4,1)	(10,20)~(20,10)
11	(2,4,1,3)	(10,20)~(20,10)
12	(2,4,3,1)	(10,20)~(20,10)

表 4.13、表 4.14 與表 4.15 為不同 Case 測試的結果，表格中列出各個 Case 在不同傷患配置下，救護車擺放於 H1 與 H2 的數量。如表 4.13 所示，在 Case1 中，第一列代表在 e1、e2 分別有 10、20 位傷患的情況下，H1 放置了 10 台救護車，而 H2 沒有放置救護車。

表 4.13 中，Case1、Case2 與 Case3、Case4 的相同處在於，H1 與 e1、e2 的距離一樣，而 H2 與 e1、e2 的距離不一樣。表中可以發現 Case1 與 Case2 把所有救護車都擺放於 H1，因為距離的關係，救護車在事故點時並不會將救護車送至 H2，因此救護車全數放置於 H1，而 Case1、Case2 可視為只有 1 間醫院的情況；在 Case3 中，救護車全數放置於 H1，但在 Case4 中卻因為 H2 與 e1、e2 的距離不同，並考量到救護車後續的派遣路徑，導致救護車放置的數量與 Case3 有極大的差異，而在 Case4 中並沒辦法從距離或是傷患數目找出救護車放置的比例與規則。在此表中除了 Case1、Case2 可以視為 1 間醫院的情境之外，其它兩個 Case 並沒辦法看出救護車的放置比例。

表 4.13 救護車放置數量表(一)

(P1,P2)	Case1(1,2,3,4)		Case2(1,2,4,3)		Case3(1,3,2,4)		Case4(1,3,4,2)	
	H1	H2	H1	H2	H1	H2	H1	H2
(10,20)	10	0	10	0	10	0	5	5
(11,19)	10	0	10	0	10	0	5	5
(12,18)	10	0	10	0	10	0	6	4
(13,17)	10	0	10	0	10	0	6	4
(14,16)	10	0	10	0	10	0	7	3
(15,15)	10	0	10	0	10	0	7	3
(16,14)	10	0	10	0	10	0	8	2
(17,13)	10	0	10	0	10	0	8	2
(18,12)	10	0	10	0	10	0	9	1
(19,11)	10	0	10	0	10	0	9	1
(20,10)	10	0	10	0	10	0	10	0

表 4.14 中，Case7 與 Case8 可以視為 1 間醫院的情境，救護車全數擺放於 H1 後，會將 e1 的傷患送完後再送 e2 的傷患，而過程中因為距離的關係，不會將傷患送至 H2；而 Case11、Case12 則因為 H2 與 e1、e2 的距離影響救護車初始擺放的位置，不論傷患數目，Case11 一開始所有救護車放置於 H2，但 Case12 却因為 e1、e2 不同的傷患數量以及事故點的距離而影響救護車初始放置的數量，但在此並無一個特定的數值可推得救護車初始放置數量。

表 4.14 護車放置數量表(二)

(P1,P2)	Case7(2,1,3,4)		Case8(2,1,4,3)		Case11(2,4,1,3)		Case12(2,4,3,1)	
	H1	H2	H1	H2	H1	H2	H1	H2
(10,20)	10	0	10	0	0	10	0	10
(11,19)	10	0	10	0	0	10	0	10
(12,18)	10	0	10	0	0	10	1	9
(13,17)	10	0	10	0	0	10	2	8
(14,16)	10	0	10	0	0	10	2	8
(15,15)	10	0	10	0	0	10	2	8
(16,14)	10	0	10	0	0	10	3	7
(17,13)	10	0	10	0	0	10	4	6
(18,12)	10	0	10	0	0	10	4	6
(19,11)	10	0	10	0	0	10	4	6
(20,10)	10	0	10	0	0	10	5	5

表 4.15 中，Case5、Case6 的 H1 與 e1、e2 的距離在距離大小的排序中分別是最遠與最近的，而 H2 與 e1、e2 的距離則是在距離排序中的第 2 與第 3 位，Case5、Case6 中可以發現當 H2 與 e1、e2 的距離交換，將會對救護車初始放置的數量有顯著的影響。除此之外，在同樣的距離下，不同數量的傷患也會使救護車放置的數量有所差異，表中雖然可以看出救護車放置的數量可能與距離、傷患數有著潛在規則，但因為救護車後續的派遣路徑極為複雜，所以並不容易觀察出其中的比例與派遣規則。

表 4.15 護車放置數量表(三)

(P1,P2)	Case5(1,4,2,3)		Case6(1,4,3,2)		Case9(2,3,1,4)		Case10(2,3,4,1)	
	H1	H2	H1	H2	H1	H2	H1	H2
(10,20)	8	2	3	7	2	8	0	10
(11,19)	7	3	3	7	3	7	1	9
(12,18)	7	3	4	6	2	8	2	8
(13,17)	8	2	4	6	3	7	3	7
(14,16)	8	2	4	6	3	7	3	7
(15,15)	8	2	5	5	2	8	4	6
(16,14)	8	2	5	5	2	8	4	6
(17,13)	8	2	5	5	2	8	5	5
(18,12)	9	1	6	4	2	8	5	5
(19,11)	9	1	6	4	1	9	5	5
(20,10)	10	0	6	4	0	10	6	4

4.5.3 小結

本研究首先針對時空網路模式(AD_{TS}^{FL})、平均型順序式派遣模式(AD_{SAT}^{FL})與固定型順序式派遣模式(AD_{SFT}^{FL})的求解時間做比較，測試中發現醫院的數量對求解時間影響甚大，當醫院數量增多時，求解時間將隨之增加，另外當救護車數量少時，亦會使求解時間增加。另外 AD_{SFT}^{FL} 的求解時間略高於 AD_{SAT}^{FL} 的求解時間，因為 AD_{SFT}^{FL} 所產生的變數比 AD_{SAT}^{FL} 還要多，使得 AD_{SFT}^{FL} 的求解時間略高於 AD_{SAT}^{FL} 。

第二部分測試了演算法 GDA_{f1}^{FL} 、 GDA_{f2}^{FL} 與 AD_{SAT}^{FL} 、 AD_{SFT}^{FL} 的求解品質，在醫院數量為 1 間醫院時， GDA_{f1}^{FL} 、 GDA_{f2}^{FL} 皆能求解至最佳解，但是當醫院數量增加時， GDA_{f1}^{FL} 與 GDA_{f2}^{FL} 的 Gap 將變得非常差，因為這兩種方法並未考慮救護車後續的派遣，其中 GDA_{f2}^{FL} 因為對後續的派遣路徑考量的更少，使得 GDA_{f2}^{FL} 的表現比 GDA_{f1}^{FL} 還要更差，因此演算法 GDA_{f1}^{FL} 、 GDA_{f2}^{FL} 僅適合使用於一間醫院的情境，當多間醫院時，求解品質並不令人滿意。而單獨比較 AD_{SAT}^{FL} 與 AD_{SFT}^{FL} 時，發現 AD_{SAT}^{FL} 會受到醫院數目、傷患數目與事故點數目的影響，使得 Gap 變高，但

AD_{SFT}^{FL} 只有在問題規模最大時(Case24)，才會有些微的 Gap 產生，總體來說 AD_{SFT}^{FL} 是個同時兼具求解時間與求解品質的演算法。

第三部分針對救護車擺放的位置做測試，測試後發現，雖然各個 Case 可以經由事故點的距離來判斷是否能簡化為 1 間醫院的情境，但若是事故點的距離無法簡化為 1 間醫院的情境時，目前並不能由傷患數量或事故點的距離來計算出救護車初始的擺放數量，推斷原因是因為救護車初始放置的數量必須要考慮救護車在後續週期的派遣狀況，無法只憑著初始的傷患人數或者事故點的距離，得出救護車確切的放置比例，而當問題規模越大，此情況將顯得越嚴重，且任何的傷患或距離的變動，都會影響救護車的放置醫院。



第五章

結論與未來方向

5.1 研究結論

本研究主要討論在緊急情況下的大量傷患之救護車派遣問題，相關的決策複雜，且在分秒必爭的狀況下，單憑救護派遣員的經驗及人為的判斷，難以在有限的時間內，做出最佳的救護車派遣決策。本研究將救護車派遣問題分成「已知初始發車處之救護車派遣問題」與「可彈性調整初始發車處之救護車派遣問題」兩類後，進行救護車派遣與救護車放置於醫院時的配置數量探討，同時提供不同觀點的數學模式，並提出相關的求解貪婪演算法，提升求解品質與求解效率。

5.1.1 已知初始發車處之救護車派遣問題

由於現有的緊急救護車指派系統中，皆是以決策者的自身經驗派遣救護車，因此並不能保證最小化傷患的總等候時間。故本研究在第三章提出已知初始發車處之救護車派遣問題，並將連續的時間離散化，架構出時空網路圖(Time Space Network)以建構救護車派遣之整數規劃模式，而隨著時空網路圖的擴大，求解時間將會隨之上升，甚至高於 2 小時以上，故本研究提出傷患找車(GDA_d^{IL})與平衡傷患數量(BDA^{IL})兩種演算法，希望提升求解效率。數值測試中發現， GDA_d^{IL} 能夠有效降低求解時間，並且得到還不錯的解，扣除離群值後，平均求解 Gap 皆在 4% 以內，且最好的 Gap 可以達到 0%，可說明 GDA_d^{IL} 的確是不錯的求解演算法；而 BDA^{IL} 雖然在傷患總等候時間的求解表現不佳，但卻能夠使最大的傷患等候時間差距最小化，因此，若以考量各個傷患等候時間的公平性為目標，此演算方法有不錯的表現。本研究有系統地提供給決策者求解品質佳且求解時間快速的演算法做出相關的派遣比較，以決定最適當的派遣決策。

5.1.2 可彈性調整初始發車處之救護車派遣問題

當傷患數量與事故點數量增加至非常大的時候，救護車的初始位置已非救護車派遣的關鍵了，當大量傷患與事故點的情況時，大多的時間裡，救護車都在醫院與事故點之間來回載送傷患。因此本研究發展可彈性調整初始發車處之救護車派遣模式，省略救護車初始皆置於消防站的限制，而是將救護車全數置於醫院，目的是希望能夠模擬傷患數量無限多的情況，進而決定救護車與醫院、事故點的配對關係與放置比例，本研究希望提供決策者救護車初始的放置數量與初始的放置醫院，因此在本模式中，除了救護車的最佳派遣路徑外，同時還需要求解救護車擺放的最佳位置。本研究以時空網路圖(AD_{TS}^{FL})與順序式救護車派遣(AD_S^{FL})兩種不同的觀點建構整數規劃模式，並以最小化傷患總等候時間當作目標值，決定救護車的最佳派遣路徑與救護車最佳的放置醫院和數量。此外，本研究在測試中發現了單一醫院、多事故點時優先處理最近事故點傷患的救護車派遣規則，並以此規則發展出演算法 GDA_{f1}^{FL} 與 GDA_{f2}^{FL} 。數值測試中，時空網路圖的數學模式消耗最多求解時間，而求解速度最快的是平均型的順序式派遣模式(AD_{SAT}^{FL})，但其求解品質劣於固定型順序式派遣模式(AD_{SFT}^{FL})； AD_{SFT}^{FL} 模式有不錯的求解品質，且其求解時間也都在可以接受的範圍內；而演算法 GDA_{f1}^{FL} 、 GDA_{f2}^{FL} 在 1 家醫院時有不錯的表現，可是當醫院數量增加時，其求解 Gap 將急遽上升，因此 GDA_{f1}^{FL} 、 GDA_{f2}^{FL} 不適合求解醫院數量過多的問題。最後在救護車擺放位置的測試中發現，救護車的初始擺放位置雖然可以藉由事故點與醫院距離得知，但其數量卻無法估算出來研究發現救護車的初始位置除了與傷患、距離有關外，更與救護車之後的派遣路徑息息相關，牽涉層面複雜，並無法以單純的參數得知放置數量。

5.2 未來研究方向

關於大量傷患事故的救護車派遣問題，在後續研究中可深入探討的課題如下：

1. 以最小化傷患等候時間之標準差為目標值

在已知初始發車處之救護車派遣模型中，本研究是以最小傷患總等候時間為目標，但若以此為目標值可能會使得不同事故點的傷患等候時間差距過大，因此後續研究中或可試圖讓傷患等候時間互相不會相差太多為目標，求解救護車派遣路徑，相關限制僅需將目標式換成式(5.1)、並將式(5.2)加入原限制中即可。而數學式子中索引、參數與變數的定義與第三章相同，其中式(5.2)中的 nP 代表傷患總人數， $t + d_{ij}$ 代表傷患抵達醫院的時刻。

目標式

$$\text{Min } Z = \sum_{k=1}^K \sum_{i=1}^I \sum_{j=1}^J \sum_{t=1}^T |x_{ijt}^k(t + d_{ij}) - x_{ijt}^k \mu| \quad (5.1)$$

限制式

$$\mu = \text{傷患平均等候時間} = \frac{1}{nP} \left(\sum_{k=1}^K \sum_{i=1}^I \sum_{j=1}^J \sum_{t=1}^T x_{ijt}^k (t + d_{ij}) \right) \quad (5.2)$$

而為了將使模式以線性規劃方式求解，可將非線性之目標式使用輔助變數 SD_{kijt}^+ 、 SD_{kijt}^- 並加入式(5.3)與式(5.4)，即可將原始之非線性模式轉成線性模式。

線性目標式

$$\text{Min } Z = \sum_{k=1}^K \sum_{i=1}^I \sum_{j=1}^J \sum_{t=1}^T (SD_{kijt}^+ - SD_{kijt}^-) \quad (5.3)$$

限制式

$$x_{ijt}^k(t + d_{ij}) - x_{ijt}^k \mu = SD_{kijt}^+ - SD_{kijt}^- \quad \forall k \in K, i \in I, j \in J, t \in T \quad (5.4)$$

2. 考量傷患依受傷程度分級

本研究假設傷患的傷重程度皆一樣，因此救護車無須依其傷重程度選擇傷患，亦不用考慮派遣的醫院是否有適合重傷患者的醫療設備。然而，實務上在事故發生現場的傷患通常會依不同傷重程度決定上車的順序，而且有些

醫院並沒有足夠的醫療設施能夠治療嚴重創傷的傷患，因此傷患的傷重程度與類別亦會影響傷患實際派遣的救護車與醫院；此外，傷患的傷重程度也會影響救護車的派遣，當傷患皆為輕傷時，僅須派出配有基本醫護人員的救護車(BLS)即可，甚至可以派遣小巴士，一次載送多名傷患；但是當傷患為重傷且性命垂危時，則需派出配有高級醫護人員的高等級救護車(ALS)，因此在傷患分級的情況下，決定救護車派遣順序以及傷患派遣的醫院，亦是大量傷患事件的重要課題，後續的研究可針對此問題做進一步的研究。

3. 考量空窗期之救護車派遣問題

當緊急大量傷患事故發生時，日常的緊急救護案件仍可能同時會發生，因此若將全部救護車調去處理緊急大量傷患，可能會顧此失彼，讓其原本負責的區域發生救護「空窗期」。因此，後續研究中可參考各地區發生事故的機率或發生事故的死亡率來決定救護車是否需定期回到消防站待命或控制救護車派遣的總數量。

4. 參數與求解時間之關係

本研究除了測試數學模式的求解時間，同時使用了迴歸分析來探討問題中的參數與參數間交互作用項(如：救護車數量、事故點數量、醫院數量與傷患數量)對於求解時間所造成的影響，測試結果中發現，多項參數和多項參數間的交互作用與求解時間呈現出顯著的相關性，未來的研究中，可深入探討這些參數影響求解時間的原因，期望可為緊急醫療派遣之決策人員帶來可供參考的資訊。

5. 救護車之行車速率推估與初始狀態之改變

本研究假設每一台救護車的行車速率皆為定值，但實務上救護車的行駛速率可能受到環境或駕駛對路線的熟悉程度而有變化，因此後續的研究亦可將救護車的行車速率加入救護車派遣問題，擬定更精確行車速率，如此一來便能夠更符合實務上的救護車派遣模式。另一方面，本研究假設救護車皆是

由消防站或是醫院出發至事故點，但在實務上，事故發生時，救護車可能正處於行駛中的狀態，其中包括正在前往事故點路上、完成任務回消防站路上或是正載送傷患至醫院的路上，因此未來研究中亦可將救護車的不同狀態加入問題中考慮，並進行相關的測試與分析。

6. 順序式派遣模型應用於 AD_{TS}^{IL}

本研究在第四章的可彈性調整初始發車處之救護車派遣問題中，發展出順序式救護車派遣模式(AD_{seq}^{FL})在求解時間上有不錯的表現，後續研究中，可修改模式相關限制，使用於第三章的已知初始發車處之救護車派遣模式，並進行相關的數值測試與分析。

7. 初始之救護車配置方式測試

針對小例子的初始救護車配置比例測試，本研究發現一旦救護車數量、傷患數量、事故點與醫院距離等若有小幅度的更動，就會對救護車放置的比例有極大影響，未來研究中，可增加傷患的數量，使得傷患數量遠大於救護車數量後，再做相關的測試與比較，找出其中的規律。

參考文獻

中文文獻

吳青翰(2006)。緊急醫療救護系統資源配置之模擬研究-以臺南市為例。國立成功大學交通管理科學研究所碩士論文。

吳青翰(2010)。最小化整體死亡人數之大量傷病患事故救護車派遣模式。國立成
功大學交通管理科學研究所博士論文。

胡勝川(2002)。「急診醫師與緊急醫療救護」。新北市：金名。

蘇喜、石崇良及陳麗華 (2001)。運用模擬技術建立緊急救護模式及效益評估。
行政院國家科學委員會專題研究計畫：NSC89-2314-B-002-383。

英文文獻

Andersson, T., & Vaerband, P. (2007). Decision support tools for ambulance dispatch and relocation. *Journal of the Operational Research Society*, 58(2), 195-201.

Aytug, H., & Saydam, C. (2002). Solving large-scale maximum expected covering location problems by genetic algorithms: A comparative study. *European Journal of Operational Research*, 141, 480-494.

Berman, O., Drezner, Z., & Krass, D. (2011). Discrete cooperative covering problems. *Journal of the Operational Research Society*, 62, 2002-2012.

Bish, D. R., Agca, E., & Glick, R. (2011). Decision support for hospital evacuation and emergency response. *Annals of Operations Research*, 221, 89-106.

Carter, G. M., Chaiken, J.M., & Ignall, E. (1972). Response areas for two emergency units. *Operations Research*, 20, 571-594.

Brotcorne, L., Laporte, G., & Semet, F. (2003). Ambulance location and relocation models. *European Journal of Operational Research*, 147, 451-463.

Church, R. L., & ReVelle, C. (1974). The Maximal Covering Location Problem. *Papers of Regional Science Association*, 32(1), 101-118.

Cuningham, R. A., & Harries, G. (1988). Nearest-neighbour rules for emergency services. *Zeitschrift für Operations Research*, 32(5), 299-306.

Daskin, M. (1983). A maximum expected covering location model: Formulation, properties and heuristic solution. *Transportation Science*, 17, 48-68.

Gendreau, M., Laporte, G., & Semet, F. (1997). Solving an Ambulance Location Model by Tabu Search. *Location Science*, 5(2), 75-88.

Gendreau, M., Laporte, G., & Semet, F. (2001). A dynamic model and parallel tabu search heuristic for real-time ambulance relocation. *Parallel Computing*, 27, 1641-1653.

Gendreau, M., Laporte, G., & Semet, F. (2006). The maximal expected coverage relocation problem for emergency vehicles. *Journal of the Operational Research Society*, 57, 22-28

Gong, Q., & Batta, R. (2007). Allocation and reallocation of ambulances to casualty clusters in a disaster relief operation. *IIE Transactions*, 39(1), 27-39.

Haghani, A., Hu, H., & Tian, Q. (2003, January). *An optimization model for real-time emergency vehicle dispatching and routing*. Paper presented at the meeting of the Transportation Research Board, Washington, DC.

Huang, B., & Pan, X. (2007). GIS coupled with traffic simulation and optimization for incident response. *Computers Environment and Urban Systems*, 31, 116-132.

Ibri, S., Noureldath, M., & Drias, H. (2012). A multi-agent approach for integrated emergency vehicle dispatching and covering problem. *Engineering Applications of Artificial Intelligence*, 25(3), 554-565.

- Ibri, S., Nourelfath, M. & Drias, H. (2010). On the integration of dispatching and covering for emergency vehicles management system. *Machine and Web Intelligence (ICMWI), 2010 International Conference on*, 198-204.
- Inoue, H., Yanagisawa, S., & Kamae, I. (2006). Computer-simulated assessment of methods of transporting severely injured individuals in disaster—case study of an airport accident. *Computer Methods and Programs in Biomedicine*, 81(3), 256-265.
- Jacobson, E. U., Argon, N. T., & Ziya, Serhan. (2012). Priority Assignment in Emergency Response. *Operation Resaerch*, 60(4), 813-832.
- Jagtenberg, C. J., Bhulai, S., & Mei, R. D. (2015). An efficient heuristic for real-time ambulance redeployment. *Operation Research for Health Care*, 4, 27-35.
- Jotshi, A., Gong, Q., & Batta, R. (2009). Dispatching and routing of emergency vehicles in disaster mitigation using data fusion. *Socio-Economic Planning Sciences*, 43(1), 1-24.
- Karasakal, O., & Karasakal, E. K. (2004). A maximal covering location model in the presence of partial coverage. *Computer & Operation Research*, 31, 1515-1526.
- Koenig, K. L., Dinerman, N., & Kuehl, A. E. (1996). Disaster nomenclature — a functional impact approach: the PICE system. *Academic Emergency Medicine*, 3(7), 723-727.
- Larson, R. C. (1974). A hypercube queuing model for facility location and redistricting in urban emergency services. *Computers & Operations Research*. 1, 67-95.
- Lee, S. (2011). The role of preparedness in ambulance dispatching. *Journal of the Operational Research Society*, 62, 1888-1897.
- Li, X., Zhao, Z., & Zhu, X. (2011). Covering models and optimization techniques for emergency response facility location and planning: a review. *Mathematical Methods of Operations Research*, 74, 281-310.

- Maleki, M., Majlesinasab, N., & Sepehri, M. M. (in press). Two new models for redeployment of ambulances. *Computers & Industrial Engineering*, 78, 271-284.
- Repede, J. F., & Bernardo, J. J. (1994). Developing and validating a decision support system for locating emergency medical vehicles in Louisville, Kentucky. *European Journal of Operational Research*, 75, 567-581.
- Schilling, D. A., Elzinga, D. J., Cohon, J., Church, R. L., & ReVelle, C. (1979). The team/fleet models for simultaneous facility and equipment Siting. *Transportation Science*, 13(2), 163-175.
- Spaite, D. W., Valenzuela, T. D., Meislin, H. W., Criss, F. A., & Hinsberg, P. (1993). Prospective validation of a new model for evaluating emergency medical services systems by in-field observation of specific time intervals in prehospital care. *Annals of Emergency Medicine*, 22, 638-645.
- Tayfur, E., & Taaffe, K. (2009). A model for allocating resources during hospital evacuations. *Computers & Industrial Engineering*, 57, 1313-1323.
- Toregas, C., Swain, R., ReVelle, C., & Bergman, L. (1971). The location of Emergency service facilities. *Operations Research*, 19 (6), 1363-1373.
- Toro-Diaz, H., Mayorga, M. E., Chanta, S., & McLay, L. A. (2013). Joint location and dispatching decisions for Emergency Medical Services. *Computers & Industrial Engineering*, 64, 917-928.
- Weintraub, A., Aboud, J., Fernandez, C., Laporte, G., & Ramirez, E. (1999). An emergency vehicle dispatching system for an electric utility in Chile. *Journal of the Operational Research Society*, 50, 690-696.
- Yang, S., Hamed, M., & Haghani, A. (2007). Integrated approach for Emergency Medical Service location and assignment problem. *Transportation Research Record: Journal of the Transportation Research Board*, 1882, 184-192.