

國立成功大學  
工業與資訊管理學研究所  
碩士論文

最大化最小瞬時通訊量之無人機飛行路線規劃問題研究

On the optimal UAV flight path planning  
for maximum min data transmission



研究生：李蒿仔

指導教授：王逸琳 博士

中華民國一百一十一年六月

國立成功大學

碩士論文

最大化最小瞬時通訊量之無人機飛行路線規劃問題  
研究

On the optimal UAV flight path planning for  
maximum min data transmission

研究生：李蒿仔

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

丁慶榮 林立凡

周詩梵

指導教授：王逸琳

系(所)主管：翁美京

(單位主管是否簽章授權由各院、系(所、學位學程)自訂)

中華民國 111 年 6 月 17 日

## 摘要

無線網路在急難救災應用可扮演重要角色，特別當因災難而導致原本定點建置的通訊基地台毀損時，以無人機權充之行動通訊基地台有更佳的機動、即時與便利性，更適合應用於急難救災時，成為災區臨時搭建之無線通訊蜂巢網路，方便搜救通訊使用。無人機在服務過程中，若每一時刻都能盡其所能地提高其用戶所接收之資訊量，便能保障用戶之間通訊接收量之公平性及品質；然而此問題牽涉到無人機的飛行路線、通訊電力以及其對各用戶的頻寬指派排程等諸多決策，而不同決策之間又會互相影響，導致最佳決策求解困難。

本研究將無人機飛行範圍離散化，假設其可移動之路線、用戶需求點位置等皆為已知，以時空網路圖為基礎，先簡化無人機與用戶通訊量計算方式以構建混整數規劃模型來求解最佳之無人機飛行路線和通訊設定排程，期望能權衡無人機之間互相產生之雜訊干擾和本身給予需求點之訊號強度，計算出無人機之最佳飛行路線與最好的通訊設定方式，以在服務期間最大化其每單位時間每位使用者可收送之瞬時最小資訊傳輸量，使災區內的通訊服務能更通順公平可靠。為加速求解並能處理較大規模的路線與通訊設定規劃，本研究亦提出滾動式求解演算法機制，將規劃期間切割成數個時段，每次使用混整數規劃模式求解部分連續時段內之決策，於固定其較早時段之決策後，再重複此過程以規劃後續時段，直至所有時段皆完成規劃。此作法雖以粗略的通訊量公式為基礎，但仍可求解出一組不錯的無人機飛行路線當候選路線。為計算更精確的通訊量以及無人機飛行路線，本研究亦提出一個外層粒子位置決定飛行路線、內層粒子位置計算功率與頻寬比的兩階層粒子群演算法，其架構在於給定外層粒子位置（即飛行路線）之下，收斂內層粒子的最佳位置（即各時段的無人機功率與其對各用戶的頻寬比設定），再調整外層粒子之位置，重複上述步驟直至收斂。藉由敏感度分析，我們可找到用戶間隔距離所對應無人機派遣之最佳數量及最佳懸停位置，並可依此法則設計貪婪演算法。數值測試結果顯示，結合滾動式求解機制之混整數規劃模式雖然效能較差，但其所算出的無人機飛行路線可當兩階層粒子群演算法的外層粒子初始位置以加速收斂過程，然而貪婪演算法所得之無人機飛行路線若再配合內層粒子群演算法，則又比上述作法有更佳之效能。

關鍵詞：無人機；路線規劃；混整數規劃；無線網路；粒子群演算法

# On the optimal UAV flight path planning for maximum min data transmission

Author: Chiao-Yu Li

Advisor: I-Lin Wang

Department of Industrial and Information Management, National Cheng-Kung University

## SUMMARY

This research uses UAVs as mobile base stations to provide temporary wireless relay networks in disaster areas and maximize the total data transmission during the service period. To ensure the fairness and quality of data transmission for each user, it is necessary to reasonably plan each UAV's flight path, communication power, and bandwidth assignment schedule. A time-space network as a framework is used to model flight paths and scheduled communication settings. To consider the noise interference among UAVs and the signal strength provided to users, we simplify the data transmission calculation to construct a mixed integer programming model (MIP) and then design a rolling-horizon mechanism that solves a smaller MIP at each iteration. The rolling-horizon MIP solution can be used as good initial flight paths for our two-stage nested particle swarm optimization algorithm, where the outer-stage particle position converges a UAV's flight path, and the inner-stage particle position converges the UAV's power and bandwidth ratios. The sensitivity analysis helps us deisgn a greedy heuristic that directly derives locally optimal quantity and hovering positions of UAVs based on the users' mutual distances. Finally, the computational experiments show that our greedy heuristic calculates the best UAV flight paths and settings of power and bandwidth.

**Key words:** UAVs, Flight Path, Mixed integer Programming, Wireless Relay Network, Particle Swarm Optimization

## INTRODUCTION

Wireless communication has become a necessity in our daily life and is crucial in post-disaster communication when wired communication becomes infeasible due to disasters. In that situation, we may treat unmanned aerial vehicles (UAVs) as base stations that facilitate wireless communication in disaster areas. In other words, we need to calculate each UAV's flight path, communication power, and bandwidth assignment schedules. The calculation of data transmission involves nonlinear functions of these decision variables to model Shannon's Capacity Formula and interference effects caused by different base stations. These complicate the design of linear formulation based on mathematical programming techniques. Most literature solves this problem by the Block Coordinate Descent algorithm, which fixes variables of most types except one at each iteration. In particular, it selects one of the three variables in turn (flight paths, power, and bandwidth) to optimize while keeping the remaining two variables fixed. The above three steps are repeated until a termination condition is met. We agree with optimizing the communication settings based on given flight paths but not vice versa. In this thesis, we have proposed three solution methods to calculate the UAVs' flight paths and communication settings on a time-space framework: (1) a mixed integer programming model (MIP) that simplifies and linearizes the complicated Shannon's Capacity Formula and other communication settings; (2) a two-stage nested Particle Swarm Optimization (PSO) heuristic algorithm; and (3) a greedy heuristic based on the observation from sensitivity analysis.

## MATERIALS AND METHODS

We assume that the starting position of UAVs and the locations of each user are known and fixed on a grid network. A UAV can only move one unit along the grid or hover over a grid point at each unit time. The pair of UAV-user must comply with the following communication protocol: (1) Each user can only be served by one of its nearest UAVs. (2) Each UAV can serve multiple users simultaneously, but the UAV has a power and bandwidth capacity limit. We have applied some integer programming formulation techniques to linearize Shannon's Capacity Formula for constructing a mixed integer programming model (MIP) on a time-space network. We seek the optimal flight path and the best communication settings for each UAV to maximize instantaneous minimum data transmission per unit of time during the service period so that the communication service is provided with both quality and fairness consideration.

Since solving the MIP is time-consuming for larger cases, we propose a rolling-horizon mechanism that divides the entire planning horizon into several consecutive segments of overlapped shorter periods. We then solve the same MIP but on one segment at one time, fix

some of its decision variables of earlier periods, and then select the next segment that overlaps with the previous one in the latter periods. We repeat the above procedures in a rolling fashion so that all the decisions on the entire planning horizon are finally calculated. Since this method does not accurately follow Shannon's Capacity Formula and the nonlinear interference relation, its flight paths can be referred to but not the communication settings.

On the other hand, our two-stage nested PSO algorithm and greedy heuristic can more accurately calculate the communication settings using Shannon's Capacity Formula. In particular, the outer-stage PSO procedure converges good flight paths for UAVs. The inner-stage PSO procedure converges each UAV's power and bandwidth settings for a given set of UAV flight paths by the outer-stage PSO procedure. The greedy heuristic is designed based on sensitivity analysis. We have identified a threshold distance (e.g., 14470m from our testing) between two bases (i.e., UAVs) for wireless interference to take effect. For example, if two UAVs keep a distance longer than 14470m from each other, their interference can be neglected. Then we would deploy two UAVs, one UAV for one user, to hover above the users so they can have the best data transmission. Otherwise, we would deploy one UAV to hover above any of the two users. The greedy heuristic and MIP can calculate a good set of UAV flight paths as the initial outer-stage particle position for the two-stage nested PSO algorithm.

## RESULTS AND DISCUSSION

Our computational experiments are performed on a personal computer with Windows 10, Intel Core i9-10900, 2.80GHZ Processors, and 16GB RAM. Gurobi 9.1 version is used for solving MIPs. Our MIP models and algorithms are implemented in Python. The parameter settings for the two-stage nested PSO algorithm are determined from the sensitivity analysis. The results of the computational experiments indicate the two-stage nested PSO algorithm with initial outer-stage particle positions set by the greedy heuristic gives the best result. The MIP result can only help provide another option for setting the initial UAV flight paths, whose quality is worse than the one by our greedy heuristic. The interference between different UAVs does affect the transmission, should the distance between two UAVs be closer than 14470m.

## CONCLUSIONS

The greedy heuristic has the best performance among the three proposed solution methods and is quite intuitive to implement. This shows the importance of wireless interference of base stations that are too close. It is interesting that initially, we expected the

UAVs to move around users to provide better transmission quality. However, the effects of interferences overthrow our expectations. If we ignore the interferences, the linearized MIP solution method may be more effective. Also, if we assume the communication settings are determined only according to the number of users serviced, the entire problem would become much easier to solve.

Nevertheless, this thesis considers all these practical factors (e.g., nonlinear Shannon's Capacity Formula and interference between UAVs), making our problem much more challenging. Our computational experiments support the effectiveness of our proposed greedy heuristic. For future research, we suggest further investigation on the modifications of our greedy heuristic for more general graphs.



## 誌謝

本論文得以順利完成，首先最感謝就是指導教授王逸琳老師的指導，在研究所兩年的期間，不管是學術研究或是生活，恩師總是毫不保留的給予許多幫助。很榮幸可以跟著恩師的腳步，讓我在專業知識更佳的專精，也在為人處事和做事的態度上有深遠的影響。在此向恩師致上最誠摯的感激。Proposal 與口試期間，承蒙書審委員呂執中老師與利德江老師以及口試委員丁慶榮老師、林仁彥老師與周詩梵老師的建議與指正，使本論文更完備，感謝各位教授。

感謝書桓學長和承中學長一直以來的關心與照顧，你們真的是史上最棒的學長了！真的好懷念以前一起在學校吃飯、聊天、玩桌遊的日子喔！謝謝你們總是在我需要幫忙的時候主動的關心我並給予我協助。感謝威銓、郁恩從碩一開始一起討論課業、準備助教課，到後面一起處理論文，都幫了我不少的忙，感謝碩班這兩年有你們的陪伴。Rani 和 Ocha，雖然我們比較少聊天，但還是很謝謝你們每次都很努力地想懂的破英文、和我交流，希望在印尼的妳們未來一切順利。

學弟妹彥融、凱翔、怡玲和閔樺，雖然因為疫情的關係比較少接觸，但還是很開心你們能加入 Lab 61205，如果以後有問題也隨時歡迎你們來找我們。祝你們的論文也能順利完成，找到理想的工作。另外，在碩班時，很開心有姿勻、培峯、宜真、萱如、宜萱、致恆不時約出來聊天聚一聚，更謝謝儼蒨總是陪我一起讀書、陪我度過最不知所措最難熬的時光，每次都很用心地聽我說、給我意見，希望我們未來都能越來越好！

最後感謝我的媽媽，謝謝妳讓我能無後顧之憂的完成學業，更尊重、支持我每一個選擇。希望在接下來的旅程中我也能成為讓你驕傲、讓妳依靠的孩子！

## 目錄

摘要.....	I
誌謝.....	VI
目錄.....	VII
表目錄.....	IX
圖目錄.....	XI
第一章 緒論.....	1
1.1 研究背景與動機.....	1
1.2 研究目的.....	3
1.3 研究範圍與限制.....	4
1.4 研究架構.....	4
第二章 文獻探討.....	6
2.1 無人機種類及特色.....	6
2.2 多台無人機基地台提供無線網路之優勢.....	7
2.3 無人機提供無線網路之模式.....	9
2.3.1 定點設置.....	9
2.3.2 機動式飛行.....	9
2.4 通訊信道.....	9
2.4.1 夏農公式參數應用.....	13
2.5 過往文獻研究比較.....	13
2.5.1 問題設定.....	14
2.5.2 求解方式.....	14
2.6 小結.....	15
第三章 無人機最佳飛行路線與通訊設定之混整數規劃模型.....	16
3.1 問題描述.....	16
3.2 問題假設與限制.....	17
3.3 數學模式.....	18
3.3.1 符號定義.....	18
3.3.2 目標式和非線性轉換架構.....	19
3.3.3 限制式.....	22
3.4 變數與限制式個數估算.....	27
3.5 小結.....	29
第四章 無人機最佳飛行路線與通訊設定之演算法.....	30
4.1 滾動式求解機制.....	30
4.1.1 滾動式求解概念.....	30
4.1.2 滾動式求解機制虛擬碼.....	31
4.2 雙階層粒子群演算法.....	32

4.2.1 粒子群演算法概念.....	33
4.2.2 雙階層粒子群演算法虛擬碼.....	36
4.2.3 $PSO_{traj}$ 虛擬碼.....	38
4.2.4 $PSO_{pb}$ 虛擬碼.....	41
第五章 數值分析.....	44
5.1 粒子群演算法參數之敏感度分析.....	44
5.2 問題設定之敏感度分析.....	48
5.3 用戶分群之求解結果比較.....	50
5.4 不同用戶數及無人機數之求解結果比較.....	54
5.5 小結.....	63
第六章 結論與未來研究建議.....	65
6.1 結論與貢獻.....	65
6.2 未來研究方向.....	67
參考文獻.....	69
中文.....	69
英文.....	69



## 表目錄

表 2-1 無人機種類及性能 .....	6
表 2-2 決策變數設定比較 .....	11
表 2-3 夏農公式參數 .....	13
表 2-4 夏農公式參數(統一單位 $w$ ) .....	13
表 2-5 問題設定比較 .....	14
表 2-6 文獻之求解方式比較表 .....	15
表 3-1 估算變數與限制式個數之參數設定值 .....	27
表 3-2 變數總個數估算 .....	27
表 3-3 限制式總個數比較 .....	28
表 4-1 執行時間比較 .....	31
表 4-2 滾動式求解方法設計邏輯 .....	32
表 4-3 $PSO_{traj}$ 和 $PSO_{pb}$ 比較 .....	36
表 4-4 $PSO_{traj}$ 函式內容及目的 .....	39
表 4-5 $PSO_{traj}$ 初始化函式 .....	39
表 4-6 $PSO_{traj}$ 適應值函式 .....	40
表 4-7 $PSO_{traj}$ 更新函式 .....	40
表 4-8 $PSO_{traj}$ 位置轉換函式 .....	41
表 4-9 $PSO_{traj}$ 粒子群演算法函式 .....	41
表 4-10 $PSO_{pb}$ 初始化函式 .....	42
表 4-11 $PSO_{pb}$ 適應值函式 .....	43
表 4-12 $PSO_{pb}$ 更新函式 .....	43
表 4-13 $PSO_{pb}$ 粒子群演算法函式 .....	43
表 5-1 參數設定 .....	44
表 5-2 文獻參數比較 .....	45
表 5-3 $PSO_{pb}$ 參數敏感度分析 .....	46
表 5-4 $PSO_{traj}$ 參數敏感度分析 .....	48
表 5-5 用戶距離及無人機數比較 .....	49
表 5-6 求解方法與說明 .....	51
表 5-7 粒子群演算法與數學模式之最佳值比較 .....	52
表 5-8 求解方法與說明 .....	54
表 5-9 測試資料參數設定 .....	55
表 5-10 不同狀態及求解方法之最佳值比較 .....	55
表 5-11 求解方法與說明 .....	60



## 圖目錄

圖 1-1 固定式基地台 .....	1
圖 1-2 視距(LOS)和非視距(NLOS)差異 .....	2
圖 1-3 Sunglider 示意圖 .....	2
圖 1-4 研究架構與流程 .....	5
圖 2-1 地面通訊漏洞 .....	7
圖 2-2 單台無人機系統(左)和多台無人機系統(右)通訊盲點之比較 .....	8
圖 2-3 NOMA(下)和 OFDMA(上)比較圖 .....	10
圖 3-1 服務範圍切割示意圖 .....	16
圖 3-2 時空網路圖架構 .....	17
圖 4-1 滾動式方法示意圖 .....	31
圖 4-2 粒子群演算法示意圖 .....	33
圖 4-3 粒子方向示意圖 .....	33
圖 4-4 粒子群演算法流程圖 .....	35
圖 4-5 雙階層 PSO 邏輯圖 .....	36
圖 4-6 $PSO_{traj}$ 示意圖 .....	37
圖 4-7 服務範圍 .....	38
圖 5-1 $PSO_{pb}$ 參數敏感度分析趨勢圖 .....	47
圖 5-2 $PSO_{traj}$ 參數敏感度分析趨勢圖 .....	48
圖 5-3 一台無人機兩用戶最佳服務位置 .....	49
圖 5-4 兩台無人機兩用戶最佳服務位置 .....	50
圖 5-5 情境 1 .....	51
圖 5-6 情境 2 .....	51
圖 5-7 情境 3 .....	51
圖 5-8 情境 4 .....	51
圖 5-9 情境 5 .....	51
圖 5-10 粒子群演算法與數學模式之軌跡比較 .....	53
圖 5-11 一台無人機之飛行軌跡 .....	57
圖 5-12 兩台無人機之飛行軌跡 .....	58
圖 5-13 三台無人機之飛行軌跡 .....	59
圖 5-14 9 組測資之飛行軌跡 .....	63

# 第一章 緒論

本研究探討如何以多台無人機進行路徑規劃及調整頻寬和功率之通訊設定，以最大化每一個時刻的最小瞬時通訊量。本章第一節介紹本研究之背景與動機，於第二節說明研究目的，第三節則解釋本研究在模擬時相對於實際情況的範圍與限制，最後一節介紹論文架構。

## 1.1 研究背景與動機

隨著科技發展的進步，無線網路大大的突破資訊傳遞間的時間和空間限制，其廣泛的應用及所帶來的便利性逐漸成為大多數人日常生活中不可或缺的部分，尤以急難救災之情況更備受關注。

山難防救的過程中，無線網路是救援的關鍵。網路不僅可以使災民回報身邊災情狀況，進而使災害應變中心得以掌控資訊、做出最佳決策，更能提供搜救團隊間通訊及災民 GPS 定位等相關功能。目前，無線網路通常仰賴固定式基地台（如圖 1-1）來為需求範圍提供網路服務。但網路傳播條件會因為地形地物的影響被區分成視距 (Line of Sight, 簡稱為 LOS) 和非視距(Non-Line of Sight, 簡稱為 NLOS) 兩種環境（如圖 1-2），而影響需求點接收到網路的通訊品質，更嚴重甚至造成通訊漏洞。在地震、颱風都頻繁發生的台灣，固定式基地台也存在著崩塌而無法繼續提供網路之風險，如 2009 年莫拉克風災連日挾帶的豪大雨造成山區民眾網路通訊中斷，導致形成孤島現象，使搜救團隊無法得知山區災情狀況，更無法在搜救過程保持彼此穩定通訊和資訊流通，進而使救援困難重重。

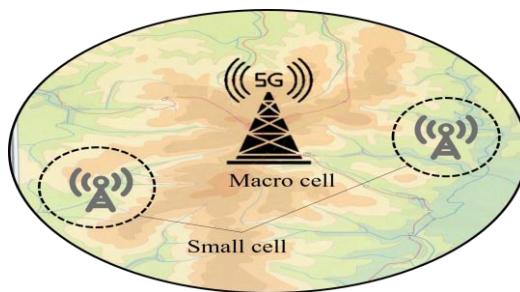


圖 1-1 固定式基地台

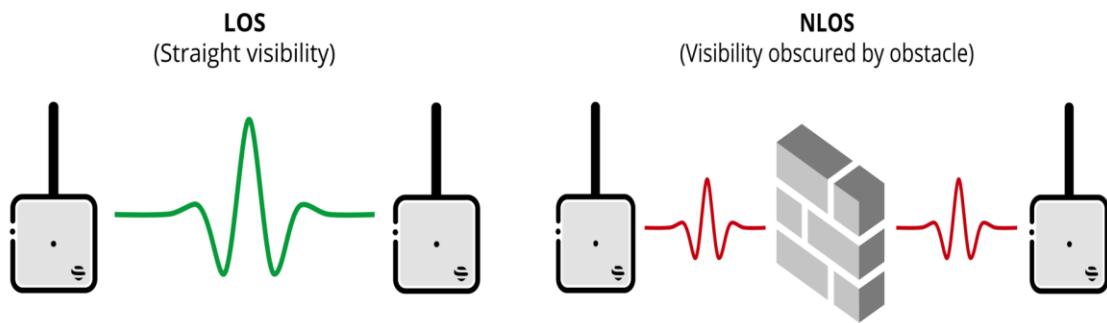


圖 1-2 視距(LOS)和非視距(NLOS)差異

圖片來源：<https://docs.sewio.net/docs/los-vs-nlos-25593229.html>

在目前這無人機盛行的年代中，由於無人機擁有機動性、便利性高及成本低等特性，因此常被廣泛的應用於軍事防護、建築建設、航拍、農業、通訊等生活中的諸多領域。而在無線通訊領域中，Shi et al. (2019) 更指出當通訊網路受制於地形地物或基地台崩壞等不可抗力因素而造成通訊盲區、無法持續為特定區域提供網路時，無人機便能充當一個飛行基地台(Flying base stations, 簡稱 FBSs)或無人機基地台(Unmanned Aerial Vehicle Mounted base stations, 簡稱 UAV-BSs)，以定點懸停或機動性飛行的模式來輔助原本固定式基地台，並快速地建立臨時蜂巢式網路、提供特定區域之網路需求。

近年來針對定點和機動性飛行的服務模式，各地也紛紛有實際行動與突破，如圖 1-3 所示，由各地電信業者合組的 HAPS 聯盟，推出 Sunglider 無人機希望藉由高空、定點向下提供無線網路，來解決偏僻地區或部分山區通訊聯繫的困難度，並於 2020 年 7 月進行超過 20 小時滯空測試。

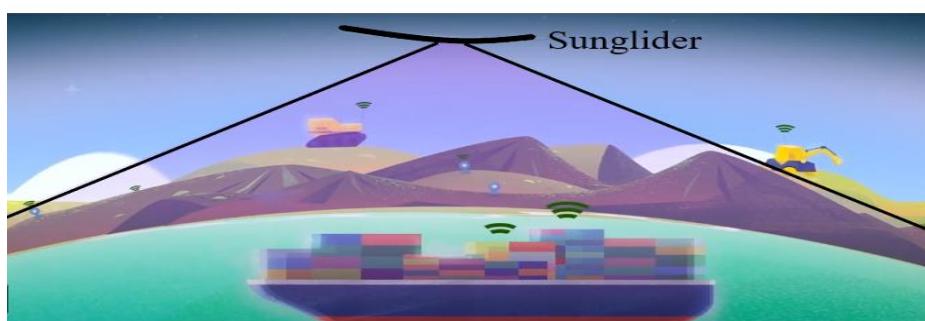


圖 1-3 Sunglider 示意圖

圖片來源：<https://udn.com/news/story/7086/4939452>

而我國國家通訊委員會更致力於研發空中基地台，使災情發生時能臨時建置定點和機動式飛行基地台，以保證災區緊急通訊需求。

無人機基地台提供無線網路的應用中，不管是定點或機動性飛行模式，除了倚靠完善、精密的硬體設備，通訊設定及路線規劃的決策問題也同樣是成功之關鍵，而每一時刻無人機之飛行軌跡、多台無人機和多個需求點之間分配方式和資源給予的量，其問題本質是一個 NP-hard 的路線規劃與功率、頻寬之排程決策問題。本研究建構時空網路圖探討路線規劃問題，並找出每台無人機對每個需求點最佳通訊設定量，使服務區域的每一時刻最小傳輸量能盡可能地極大化，以確保能提供更高品質之網路。

## 1.2 研究目的

在以往大多數以無人機群搭建臨時蜂巢無線網路相關文獻中，不會同時考慮多台無人機間的同頻干擾問題，本研究欲找出無人機的最佳路線及通訊設定方式，來最大化客戶每一時刻可收送的最小瞬時通訊量。由於無線網路資訊量會依不同用戶所接收到其服務的無人機功率以及該無人機配給該客戶的頻寬大小而改變，且該資訊量亦會受到其它無人機的通訊干擾。因此如何有效分配通訊設定、並避免其它無人機造成的干擾十分重要。

本研究之目的可歸納如下：

1. 提高每一時段，所有用戶中最小瞬時通訊量。
2. 找出每台無人機的最佳飛行路線及最佳通訊設定（發射功率、頻寬分配）方式，以減少信號干擾，提高每位用戶所接收之實際通訊量。

### 1.3 研究範圍與限制

本研究期望可以尋找無人機最佳的飛行路線和最佳通訊設定的方式，其問題本質屬於 NP-hard 的路線規劃與功率、頻寬之排程決策問題，本研究將會限縮研究範圍以簡化問題、方便建模與求解，研究範圍如下：

1. 假設使用同性能之無人機，即所提供之頻寬和功率上下限相同。
2. 忽視外在環境對於傳輸無線網路之干擾。
3. 忽視無人機本身飛行及傳輸無線網路之耗能。
4. 飛行路線的規劃只著重於二維空間，即無人機飛行過程之飛行高度皆固定。
5. 本研究會將無人機群服務範圍均分為若干格點，格點間距為無人機在單位時段以定速飛行之距離。而無人機的移動方式將被限縮於單位時間內可以上、下、左、右位移一格，或者滯空待在原格點位置。

### 1.4 研究架構

本研究之研究架構與流程如下：

#### 1. 確立研究主題

無人機漸漸受到大眾重視，在近幾年更是蓬勃發展。由於無人機本身高機動性的優點，因此常被廣泛應用在各行各業。在爭分奪秒的救災現場，無人機更能充當一個臨時通訊網路基地台，在黃金 72 小時中，保持暢通的網路以提高搜救效率。而在無人機的服務期間，除了無人機本身硬體性能外，其移動路線與頻寬和功率的相關通訊設定將決定客戶所能傳輸收送的通訊量，該如何計算最佳路線與通訊設定的參數實屬一個 NP-hard 的決策問題，因此將其設為研究之主題，期望在緊迫的時間、惡劣的地形中，最大化其效益。

## 2. 文獻回顧與整理

為了更了解無人機，研究初期參閱了多篇有關無人機種類及其特色之相關文獻，並去探討其它學者們如何將無人機應用於無線通訊領域，最後經由統合整理以界定本論文之研究範圍。

## 3. 撰寫混整數規劃模型與啟發式演算法

本研究在遵守通訊協定的前提下，經由混整數規劃模型及啟發式演算法進行求解，並進行分析。

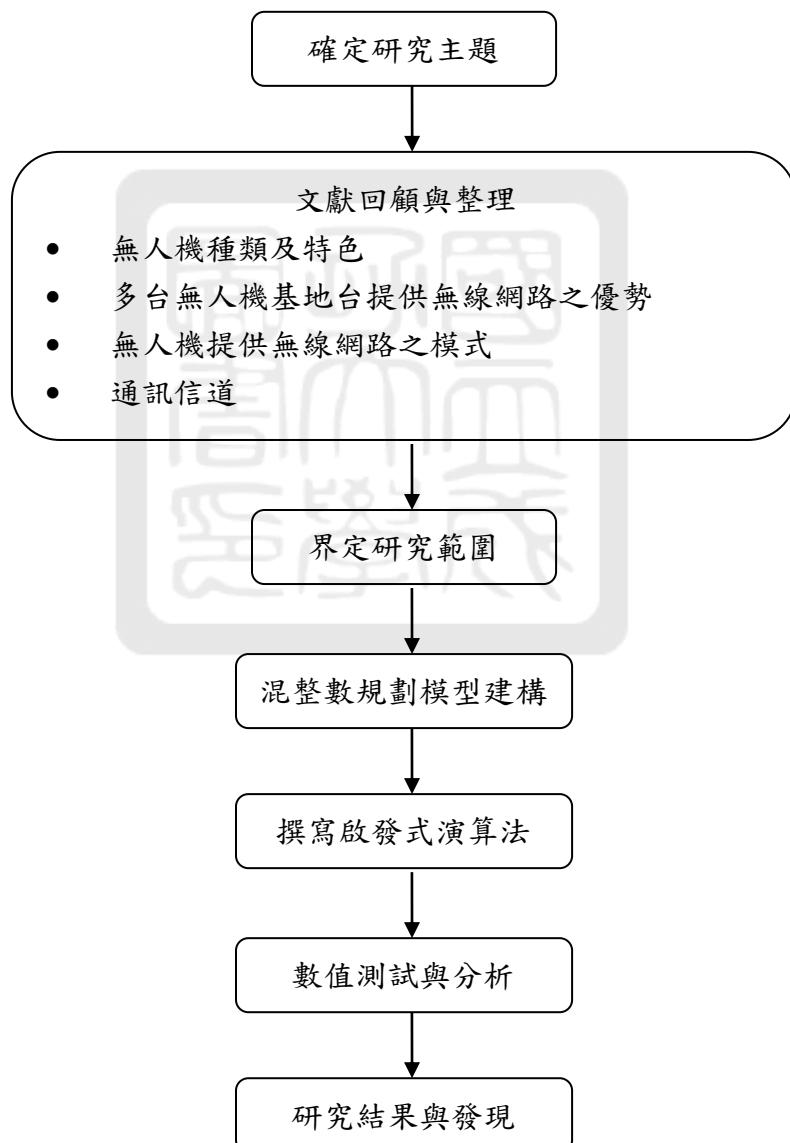


圖 1-4 研究架構與流程

## 第二章 文獻探討

本章分為四節，第一節探討貨無人機不同種類間的特性，第二節說明無人機群提供無線網路之優勢，第三節針對提供無線網路的模式進行說明，第四節回顧大部分文獻的問題設定，第五節以文獻和本研究進行比較，並於最後一節總結。

### 2.1 無人機種類及特色

近年來無人機的使用日益增加，除了軍事應用外，無人機更被應用於越來越多的民用領域。為了滿足每一個領域的要求及目標，近幾年陸續發展出許多不同種類、特色之無人機。Xu et al. (2018) 分別列出如表 2-1 各類無人機特色及差異比較。

表 2-1 無人機種類及性能

種類	固定翼式	旋翼式
優點	1. 高負載 2. 穩穩定性高 3. 繳航力佳 4. 速度快	飛行動作可懸停 和垂直起降
限制	飛行動作只能 向前	1. 低負載 2. 速度慢 3. 服務範圍小
速度(m/hr)	10-75	0-40
服務半徑(m)	5-25	0-8
續航力(hr)	1-4	1/6-1

由表中可以看出雖然旋翼式比起固定翼式速度較低、服務範圍較小且航程也較短，但相較於固定翼式只能水平起飛、飛行動作又只局限於維持向前之特性，在大部分領域之應用範圍，旋翼式機高機動性、高彈性的飛行動作卻是固定翼式所無法取代之優勢。因此，在機動性飛行模式中的無人機基地台大多會選擇旋翼式，以增加服務期間之決策彈性。

## 2.2 多台無人機基地台提供無線網路之優勢

無人機基地台(Unmanned Aerial Vehicle Mounted base stations，簡稱 UAV-BSS)，又稱為飛行基地台(Flying base stations，簡稱 FBSs)是近幾年來新的概念，期望以無人機高機動性等特性補足傳統固定式基地台之缺陷，為人們帶來更便利的生活，同時縮短人與人之間時空上的差距。

Shi et al. (2019) 提到不管是都市或鄉下常常會因為基礎建設的缺乏或障礙物的阻擋，形成如圖 2-1 的地面通訊漏洞(coverage holes，簡稱 CH)，為了解決此問題，決策者會採用無人基地台取代建設固定式基地台之方案，不僅解決無線通訊不良之問題，也不會存在花費高額費用、服務少數人，使機會成本過高的憂慮。

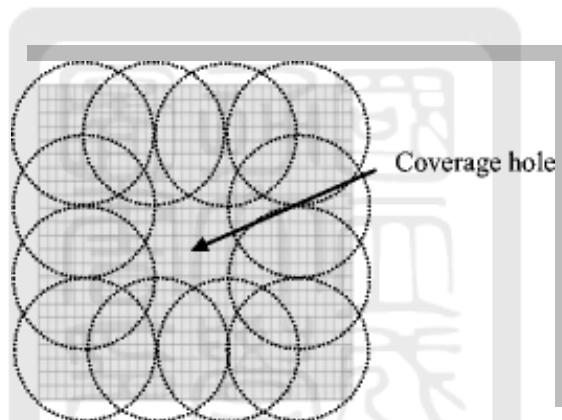


圖 2-1 地面通訊漏洞  
圖片來源： Amgoth & Jana (2015)

Wu & Zhang (2018) 認為在網路不斷發展且需求日漸增加的現在，無人機的發展足以成為一個新的通訊平台，以加強原本的蜂窩式網路，更針對無人機基地台列出以下優勢：

1. 由於無人機基地台所提供之更高視角的網路服務，降低障礙物所導致的信道衰弱，以更多機率的可視距(LOS)建立更可靠的空對地傳播通訊信道。
2. 無人機基地台可以在三維度的空間中更靈活的佈署和飛行移動，提供更有彈性的傳播模式。

3. 無人機基地台能通過所配置之控制、決策模組，做出智能化、自主性飛行決策及通訊設定方式，增強連接之信道訊號並避免不需要之干擾。

Hu et al. (2020) 指出目前最常見的飛行策略為人為遙控式單台無人機系統，操作者會針對一台無人機做監督並控制，但這種端對端的通訊方式卻存在以下的缺點：

1. 通訊信道的品質取決於一台無人機的飛行距離，無人機和用戶間距離增加會直接導致用戶所接收到的訊號品質較差。
2. 由於一台無人機有限的服務範圍，使得無人機在需求分散的情況需要飛行相對長的距離來接近需求點。
3. 無人機無法自主性地因應外界變化而採取及時決策，導致無人機可能因為意外而中斷指揮的通道，而無法回到操作者手中。

為改善以上這些缺點，多台無人機系統逐漸盛行。圖 2-2 針對單台和多台無人機系統比較其通訊盲點，其中單台無人機系統可能會在服務期間產生通訊盲點，然而多台無人機系統卻能運用數台小型無人機同時飛行相互覆蓋，減少通訊盲點的範圍。

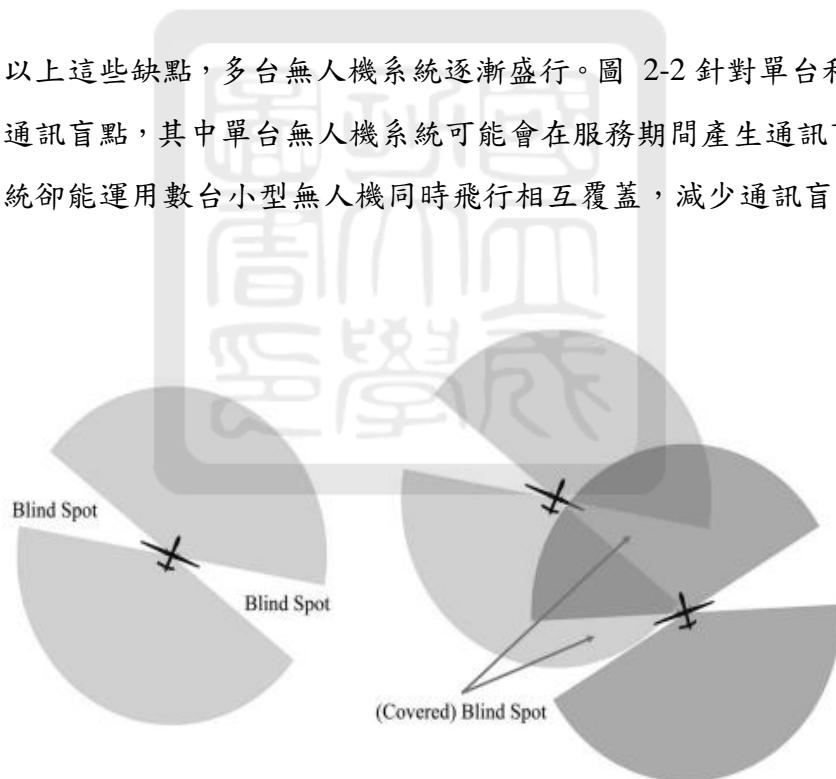


圖 2-2 單台無人機系統(左)和多台無人機系統(右)通訊盲點之比較  
圖片來源： Hu et al. (2020)

Campion et al. (2018) 也指出由於無人機群間的協調與任務分配，可以大大降低尺寸、重量和功率(SWAP)的限制，進而提高其通訊可靠性。

## 2.3 無人機提供無線網路之模式

在過往有關無人機提供無線網路之相關文獻中，分別有 Al-Hourani et al. (2014)、Alzenad et al. (2018) 和 Chou et al. (2020) 提出的「定點設置」與 Mozaffari et al. (2016)、Shi et al. (2019)、Sun et al. (2019) 相關文獻提出的「機動性飛行」兩種模式。

### 2.3.1 定點設置

在定點設置無人機來提供穩定無線網路的模式中，又細分為低空空中平台 (Low-altitude aerial platforms, 簡稱 LAPs) 與高空空中平台 (High-altitude aerial platforms, 簡稱 HAPs) 兩大類。Zeng et al. (2018) 說明當無人機作為低空空中平台時，無人機會於離地表約 600~3000 公尺的高度找到最佳懸停點，再以定點方式為需求範圍提供無線網路之服務；由於高度低，因此有快速佈署和通訊強度高的優點。而當無人機作為高空空中平台時，它們通常會在離地表數十公里的平流層中運行。與低空空中平台相比，有著覆蓋範圍更廣和續航時間更長。因此，高空空中平台通常更適合為比較大的區域長期提供可靠的無線網路。

### 2.3.2 機動式飛行

以定點設置無人機的方式來提供無線網路是個很好的無人機應用，但為了更加充分地利用無人機的高機動特性，也為了降低無人機先天的限制，機動性飛行的方式也是許多學者所探討的議題。透過無人機群服務一個指定服務範圍，可以更加彈性的調度無人機，使無人機在服務範圍中可以更充分的被利用，不會有資源被浪費或需求點因為沒在服務範圍中而沒被服務等問題。

## 2.4 通訊信道

近年來在通訊系統的相關領域中，由於非正交多址接入(Non-Orthogonal Multiple Access, NOMA)相較於一些通訊設定相關文獻中，學者所採取的正交分頻多重進接

(Orthogonal Frequency Division Multiple Access, OFDMA)系統而言，有同時服務較多用戶之優點，已廣泛應用於 5G 通訊系統上。NOMA 圖 2-3(下)的技術是 OFDMA 圖 2-3(上)的延伸，系統同樣會將所傳輸之網路頻道分割成數個較小的子載波來傳輸，使不同需求點的網路可以在同一環境下以不同子載波同時傳遞而不會互相干擾。但由於 NOMA 相較於 OFDMA 系統有不同的子載波可疊加的特性，當兩個子載波於同一頻率下傳遞資訊則會產生干擾。

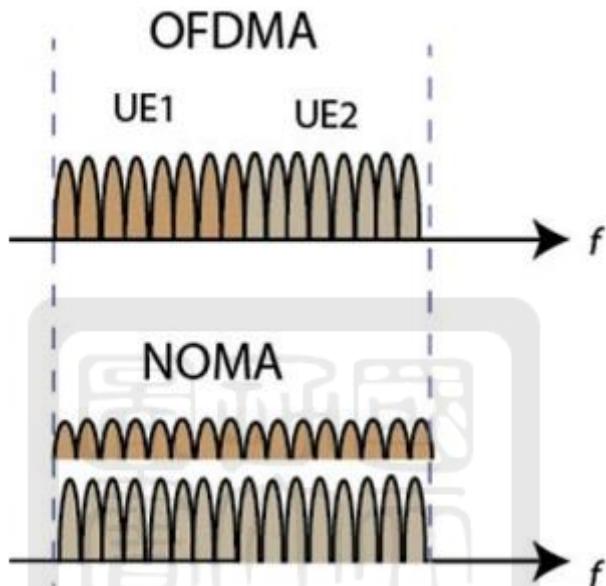


圖 2-3 NOMA(下)和 OFDMA(上)比較圖

圖片來源：<https://ithelp.ithome.com.tw/articles/10245366?sc=iThomeR>

基於 NOMA 的基本架構下，要考慮同一位用戶在同頻情境中多台無人機間的同頻干擾，而無人機在做通訊設定時除了調整發射功率以控制傳播至用戶的網路大小，更能調整給予每一位用戶的頻寬比例，使無人機可以在同一時段同時傳播給多位用戶時不會互相干擾。

下表為多篇論文針對發射功率及頻寬兩大變數的設定比較，可發現大部分學者都選擇將每台無人機之發射功率設定為連續變數，對於不同的需求點給予不同量的發射功率。而頻寬設定上，雖然 Wu et al. (2018)、Xu et al. (2018)和 Sun et al. (2019)都將其變數視為二元變數，但文獻的前提是為了避免干擾而規定每一時刻每一台無人機最多只能服務一個需求點。因此，變數分配可以視為一個頻寬比例的概念，對於每一需求點不是給予 100%的無線網路就是完全不服務此需求點。其本質和其他文獻一樣都是

將頻寬變數的分配當成一個頻寬比例的概念。本研究綜合所有文獻，將其功率設定為一連續變數，每台無人機所發射給不同需求點的功率相同。至於頻寬變數，則將其離散成若干等級的比例(為將夏農公式線性化)，針對不同需求點給予不同的頻寬比例。

表 2-2 決策變數設定比較

	每台無人機之發射功率	每台無人機之頻寬
Shi et al. (2010)	<ul style="list-style-type: none"> <li>• 離散成若干等級</li> <li>• 不同需求點給予不 同等級的發射功率</li> </ul>	<ul style="list-style-type: none"> <li>• 離散成若干子載波</li> <li>• 一子載波服務一需 求點</li> </ul>
Wu et al. (2018)	<ul style="list-style-type: none"> <li>• 為連續變數</li> <li>• 不同需求點給予不 同量的發射功率</li> </ul>	<ul style="list-style-type: none"> <li>• 為二元變數</li> <li>• 最多只有一組 UAV-User 之限制</li> <li>• 紿 100% 的網路或 完全不給</li> </ul>
Xu et al. (2018)	<ul style="list-style-type: none"> <li>• 為連續變數</li> <li>• 不同需求點給予相 同量的發射功率</li> </ul>	<ul style="list-style-type: none"> <li>• 為二元變數</li> <li>• 最多只有一組 UAV-User 之限制</li> <li>• 紿 100% 的網路或 完全不給</li> </ul>
Wu & Zhang (2018)	<ul style="list-style-type: none"> <li>• 為連續變數</li> <li>• 不同需求點給予不 同量的發射功率</li> </ul>	<ul style="list-style-type: none"> <li>• 為連續變數</li> <li>• 不同需求點給予不 同量的頻寬比例</li> </ul>
Wu & Zhang (2017)	<ul style="list-style-type: none"> <li>• 為連續變數</li> <li>• 不同需求點給予不 同量的發射功率</li> </ul>	<ul style="list-style-type: none"> <li>• 為連續變數</li> <li>• 不同需求點給予不 同量的頻寬比例</li> </ul>
Chou et al. (2020)	<ul style="list-style-type: none"> <li>• 為參數</li> <li>• 不同需求點給予相 同量的發射功率</li> </ul>	<ul style="list-style-type: none"> <li>• 為參數</li> <li>• 所有需求點會平均 分得無人機總頻寬</li> </ul>
Sun et al. (2019)	<ul style="list-style-type: none"> <li>• 為連續變數</li> <li>• 不同子載波給予不 同量的發射功率</li> </ul>	<ul style="list-style-type: none"> <li>• 為二元變數</li> <li>• 最多只有一組 UAV-User 之限制</li> <li>• 紉 100% 的網路或 完全不給</li> </ul>
本研究	<ul style="list-style-type: none"> <li>• 為連續變數</li> <li>• 不同需求點給予相 同量的發射功率</li> </ul>	<ul style="list-style-type: none"> <li>• 離散成若干等級</li> <li>• 不同需求點給予不 同量的頻寬比例</li> </ul>

而在計算每一位用戶在每一時段所接收到的網路瞬時通訊量，不僅要考慮無人機和用戶間的距離、發射功率大小，更要關注其所分得的頻寬比例，而大部分在探討通訊系統通訊設定的議題，通常會以信道容量(Channel capacity)來表達一個頻道中穩定傳輸資訊時的可達速率之上界，如 Ahmed et al. (2020)、Fallgren (2010)、Fu et al. (2018)、Kim et al. (2020) 等文獻都使用「夏農公式」(Shannon equation)計算求得此上界。夏農公式列於式(2.1)至式(2.3)。首先，式(2.1) 中分子部分的  $\rho_0$  表示距離為 1 公尺處的信道功率常數，此數學式會先經由無人機高度  $H$  及無人機與用戶間的水平距離  $d_{kut}$ ，計算無人機及用戶間的直線距離，以計算功率增益 (power gain) 的值  $h_{kut}$ 。

$$h_{kut} = \frac{\rho_0}{H^2 + d_{kut}} \quad (2.1)$$

接著需要去關注每位用戶在每時段實際所接收到之「信噪比」  $r_{kut}$ ，而每一位用戶所接收到之信噪比通常受主要服務此用戶之無人機的訊號，即發射功率  $p_{kt}$ 、功率增益  $h_{kut}$  和用戶接收截面積  $\sigma$  相乘，再加上其它無人機干擾訊號影響。而在干擾訊號部分，模型仍需考慮白噪音在自由空間中的影響。其中，用戶接收的可加性「高斯白雜訊」(Additive white Gaussian noise，簡稱 AWGN)之功率譜密度為  $N_0$ ，此 AWGN 會乘上無人機最大頻寬容量  $B$ 。由於並非所有用戶都會接收到來自無人機 100% 之頻寬比，因此會乘上用戶實際接收頻寬比大小  $b_{kut}$ ，以計算用戶所接收之白噪音  $b_{kut}BN_0$ ，如式(2.2) 所示。

$$r_{kut} = \frac{p_{kt} \times h_{kut} \times \sigma}{\sum_{k' \in N_{uav} - \{k\}} p_{k't} \times h_{k'ut} \times \sigma + b_{kut}BN_0} \quad (2.2)$$

最後，經由式(2.3)計算出當一頻道中穩定傳輸網路時可達速率的上限值  $R_{ut}$ ，此客戶  $k$  可傳輸的上限速率可由各台無人機對該客戶的信噪比  $r_{kut}$  取  $\log$  值再乘以頻寬比例  $b_{kut}$  再加總而得。

$$R_{ut} = \sum_{k \in N_{uav}} b_{kut} \times \log_2(1 + r_{kut}) \quad (2.3)$$

#### 2.4.1 夏農公式參數應用

參考多篇文獻後，本研究統整如表 2-3 之參數值設定。其中在功率的相關參數上，文獻曾用  $w$ 、 $dBm$  和  $dB$  等三個不同的單位來表示：單位  $w$  為國際單位制的功率單位；單位  $dBm$  通常定義為功率之絕對值；而  $dB$  則是任兩個功率間的比值。為統一單位以方便計算，本研究將會把參數統一以國際單位制的功率( $w$ )，如表 2-4 來計算。

表 2-3 夏農公式參數

	值	單位
$N_0$	-170	$dBm / Hz$
$\rho_0$	-50	$dB$
$B$	$5 \times 10^7$	$Hz$
$P_{max}$	0.5	$w$
$H$	100	$m$
$\sigma$	1	$m^2$

表 2-4 夏農公式參數(統一單位  $w$ )

	值	單位
$N_0$	$3.98 \times 10^{-21}$	$w / Hz$
$\rho_0$	$10^{-5}$	$w$
$B$	$5 \times 10^7$	$Hz$
$P_{max}$	0.5	$w$
$H$	100	$m$
$\sigma$	1	$m^2$

#### 2.5 過往文獻研究比較

近年來諸多學者投入研究無人機相關議題，因此也有文獻探討利用無人機來提供無線通訊網路之作法，在 2.5.1 節及 2.5.2 節將簡易比較文獻之問題設定和求解方式。

### 2.5.1 問題設定

從表 2-5 可看到在過去無人機提供通訊的文獻中，可以以提供無線網路的模式粗略分為「定點設置」和「機動性飛行」兩大類，由於機動性飛行相較於定點設置更能善用無人機高彈性、高機動性、低成本之特性，因此本研究著眼於「機動性飛行模式」。觀察有關機動性飛行之相關文獻，可以發現很少學者會考慮多台無人機群間同頻干擾的問題，因此本研究將依據其它無人機（即非主要服務之無人機）所給予用戶的資源和其相對距離計算出每一時段所發出的干擾訊號，以減少其接收強度。

表 2-5 問題設定比較

	模式	無人機群	干擾
Al-Houraniet et al. (2014)	定點設置		
Alzenad et al. (2017)			
Alzenad et al. (2018)			
Chou et al. (2020)		V	V
Mozaffari et al. (2016)	機動性 飛行	V	
Shi et al. (2019)		V	
Sun et al. (2019)			
Wu et al. (2017)			
Wu et al. (2018)			
Xu et al. (2018)		V	
本研究		V	V

### 2.5.2 求解方式

由於機動性飛行之相關文獻在處理通訊設定及規劃路線時，資源變數及無人機和用戶之間的相對距離會存在相互依存之關係，進而使模型呈現非線性，因此相關文獻都是使用「塊座標下降演算法」(Block Coordinate Descent)進行求解，如表 2-6 所示。其演算法原理是使用迭代的方式，在每次迭代只優化一項變數，直至收斂。舉例來說，若用此演算法來收斂路徑、功率、頻寬比的話，它可能會先固定路徑與功率，試圖收斂頻寬比；再固定其收斂結果的功率與頻寬比，調整並收斂路徑；最後再固定其收斂

結果的路徑與頻寬比，調整並收斂功率；重複這些步驟直至滿意為止。然而，我們認為此作法有些不甚合理之處，譬如在固定無人機任一時刻的功率與頻寬比之後，再反向去收斂路徑這個步驟，似乎「倒果為因」了；亦即，我們認為路徑的決策應該在功率與頻寬比之前才合理（由路徑來推論功率與頻寬比，而不是反過來做），因此本研究於第四章提出的演算法設計，就是先決定路徑、再依此決定對應的功率與頻寬比。此外，塊座標下降演算法極有可能只收斂於局部最佳解。因此本研究在第三章想以不同角度切入，以「夏農公式」(Shannon equation) 的相似邏輯來設計數學規劃模型，並將軌跡離散化，以數學模型來模擬其變數之間的變動趨勢。

表 2-6 文獻之求解方式比較表

	求解方式	
	演算法	數學模型
Mozaffari et al. (2016)	V	
Shi et al. (2019)	V	
Sun et al. (2019)	V	
Wu et al. (2017)	V	
Wu et al. (2018)	V	
Xu et al. (2018)	V	
本研究	V	V

## 2.6 小結

本章節一開始介紹各類無人機之特色、強調無人機群協同提供無線網路之優勢，更分別解釋其提供無線網路的模式，因此將研究設定為一群旋翼式無人機群以機動性飛行的模式來為用戶提供一臨時蜂巢網路結構。而本研究也會針對 2.4 節所提出之信道結構及夏農容量於下一章進行對應的數學編寫。

### 第三章 無人機最佳飛行路線與通訊設定之混整數規劃模型

本章探討無人機群之最佳服務路線規劃及通訊設定（發射功率、頻寬比）問題，3.1 節將描述研究問題。3.2 節將詳細說明本研究之假設與網路架構，為了方便數學模式的建立，本研究將原始問題之網路圖架構轉成以時空網路為架構的混整數規劃模式。3.3 節將說明模式之參數、變數、目標式及限制式，接著針對模型於 3.4 節提出範例演練，最後於 3.5 節進行變數與限制式個數估算。

#### 3.1 問題描述

本研究以時空網路圖為基礎。使用者可以自行將一段連續的服務時間離散化成  $T$  個相同單位長度的時段。舉例來說，假設無人機總服務時間為 10 分鐘，如欲分為分成 6 個時段來為地面用戶服務，每一單位時段將會花費 100 秒。由於無人機的飛行模式主要分為「懸停滯空」及「定速飛行」兩類，故將無人機總服務時間之飛行範圍平均切割成  $N_{node}$  個節點所組成，如圖 3-1，每兩個節點間的距離為無人機在單位時段的可達距離。在每單位時段內，無人機可選擇懸停在原本節點之上，也可以以定速沿著格點飛行到其上、下、左、右之鄰近節點。

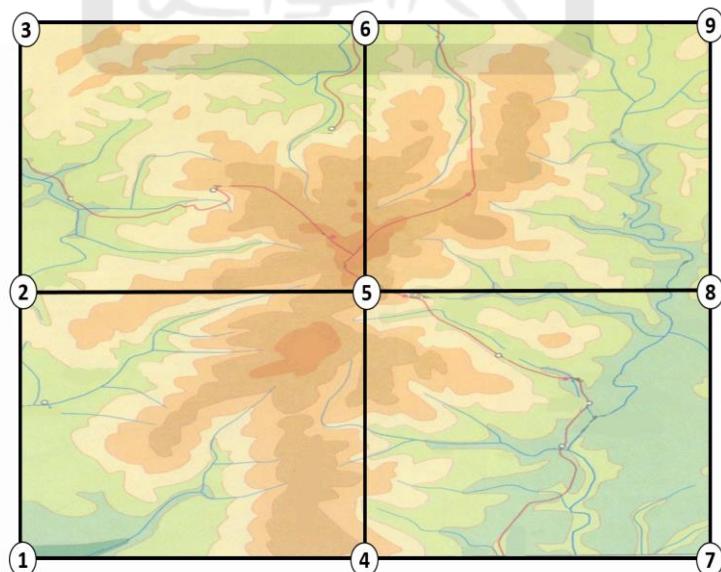


圖 3-1 服務範圍切割示意圖

時空網路圖以  $T$  個時段當垂直層和  $N_{node}$  個節點當水平層互相堆疊建構，如圖 3-2 所示。網路圖中，每時段的每一個（空間）節點都是決策點，決定無人機在每一時段的所在位置，經由串連每時段的位置即可得知無人機服務期間的飛行軌跡。

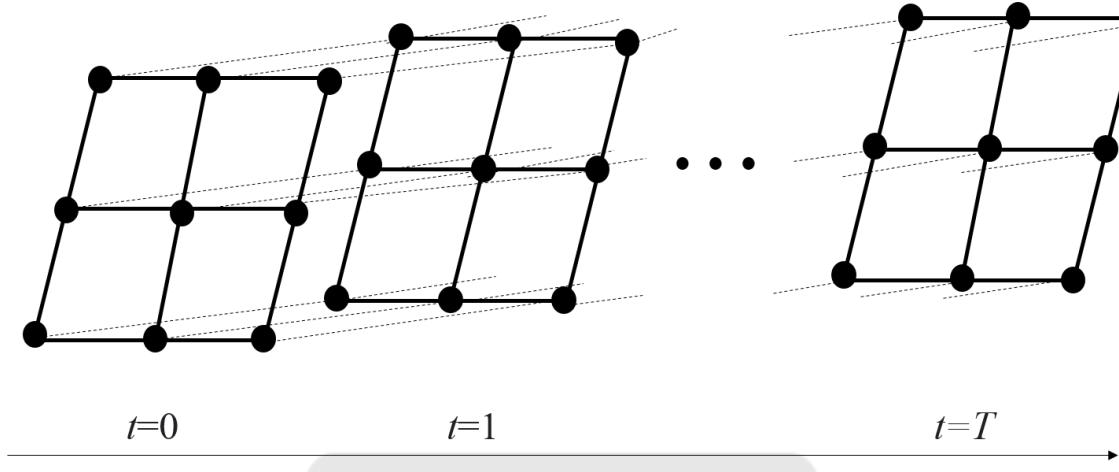


圖 3-2 時空網路圖架構

無人機服務期間，整個模型需嚴格遵守在每一時段裡，用戶只能找距離自己最近之無人機服務，而同一台無人機卻可以同時服務多人。因此，當無人機位置若固定已知，其功率以及對個別服務客戶的頻寬比例為已知的話，即可計算出每客戶可傳輸的通訊量。

### 3.2 問題假設與限制

為方便將問題使用混整數規劃模型來配置，本研究進行下述基本假設與限制：

1. 網路圖為網格狀。
2. 服務時間無人機的續航力無虞，亦即其電量皆充足，足以負荷飛行與通訊需求。
3. 每台無人機的性能相同，即所提供之頻寬和功率上下限皆相同。
4. 每架無人機都是由同一起始點出發。

5. 為方便計算無人機路線，規定各無人機每單位時間只能在平面上往橫向、直向移動一單位，禁止斜行，但允許停留在原地。
6. 假設無人機飛行時，不考量飛行時的外在天氣、干擾等不可控制因素影響。
7. 無人機及用戶間的配對須符合用戶只能被距離自己最近之一台無人機服務，而無人機可以同時服務多客戶，依其服務頻寬總和上限的通訊協定計算其傳輸資料量。

### 3.3 數學模式

3.3.1 小節將介紹此次數學模式所用到之符號定義，3.3.2 小節說明目標式設定，最後 3.3.3 小節說明限制式。

#### 3.3.1 符號定義

集合

$N_{node}$  所有節點之集合

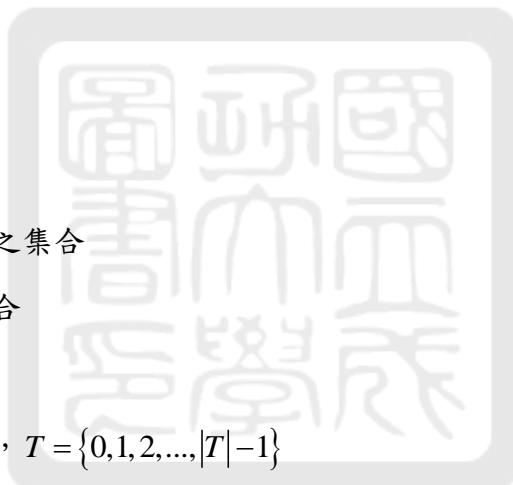
$N_{uav}$  無人機集合

$N_{user}$  用戶集合

$T$  時段集合， $T = \{0, 1, 2, \dots, |T|-1\}$

$L$  頻寬等級集合

$FP$  可行路線集合



## 參數

$P_{\max}$	無人機所發射的最大功率
$H$	無人機飛行高度
$ST$	無人機總服務時間
$D_{ui}$	每用戶和各格點間相對距離
$B_l$	每個等級 $l$ 的頻寬大小， $B_l = \{0, 0.2, 0.4, 0.6, 0.8, 1\}$
$M$	極大值

## 變數

$x_{kijt}$	無人機 $k$ 在時段 $t$ 走 $(i, j)$ 線段為 1，反之為 0
$y_{kut}$	無人機 $k$ 在時段 $t$ 服務用戶 $u$ 為 1，反之為 0
$a_{kul}$	無人機 $k$ 在時段 $t$ 以等級 $l$ 服務用戶 $u$ 為 1，反之為 0
$SNR_{ut}$	用戶 $u$ 在時段 $t$ 所接收的訊號比雜訊大為 1，反之為 0
$w_{ut}$	用戶 $u$ 在時段 $t$ 所接收的瞬時通訊量最小為 1，反之為 0
$b_{kut}$	無人機 $k$ 在時段 $t$ 服務用戶 $u$ 的頻寬
$p_{kt}$	無人機 $k$ 在時段 $t$ 發射功率
$d_{kut}$	無人機 $k$ 在時段 $t$ 與用戶 $u$ 的距離
$\underline{d}_{ut}$	用戶 $u$ 在時段 $t$ 的最小距離
$r_{kut}$	時段 $t$ 用戶 $u$ 被無人機 $k$ 服務所收信噪比
$z_{kut}$	無人機 $k$ 在時段 $t$ 服務用戶 $u$ 的頻寬乘功率
$s_{kut}$	無人機 $k$ 在時段 $t$ 服務用戶 $u$ 的訊號
$R_{ut}$	時段 $t$ 用戶 $u$ 所收到瞬時通訊量
$MinR_t$	時段 $t$ 所收到最小瞬時通訊量

### 3.3.2 目標式和非線性轉換架構

此模型的目標是期望在總服務時間內，最大化每一時段之最小傳輸量，以保障用戶間通訊公平性。如式(3.1)所示。

$$\text{Max } ST \times \sum_{t \in T} \text{Min}R_t \quad (3.1)$$

如 2.4 節所提及，用戶在每一時段所接收之瞬時通訊量須由夏農公式計算頻寬瞬時可達速率之上限，如下面(3.2)至(3.4)三條數學式，步驟如下：

1. 利用距離為 1 公尺處的信道功率常數  $\rho_0$  除以由無人機高度  $H$  平方及無人機與用戶間的水平距離平方  $d_{kut}$ ，得到由式(3.2)所求之功率增益值  $h_{kut}$ 。
2. 經由發射功率  $p_{kt}$  乘以功率增益  $h_{kut}$ ，得到每一台無人機對於此用戶之信號大小，其主要服務此用戶之無人機所給予之訊號除以非主要服務此用戶之無人機所給予之雜訊  $\sum_{k' \in N_{uav} - \{k\}} (p_{k't} \times h_{k'ut})$  為此用戶所接收之信噪比，而在干擾訊號部分，模型仍需考慮白噪音在自由空間中的影響，其中，用戶接收的加性高斯白雜訊(AWGN)之功率譜密度為  $N_0$ ，此 AWGN 會乘上無人機最大頻寬容量  $B$ 。由於並非所有用戶都會接收到來自無人機全部頻寬，因此會乘上用戶實際接收頻寬比大小  $b_{kut}$ ，求得用戶所接收之白噪音  $b_{kut}BN_0$ ，最後得到如式(3.3)所計算之實際接收信噪比  $r_{kut}$ 。
3. 最後，經由信噪比  $r_{kut}$  與頻寬比例  $b_{kut}$  計算得到如式(3.4)可達速率的上限值  $R_{ut}$ 。

$$h_{kut} = \frac{\rho_0}{H^2 + d_{kut}} \quad (3.2)$$

$$r_{kut} = \frac{p_{kt} \times h_{kut} \times \sigma}{\sum_{k' \in N_{uav} - \{k\}} (p_{k't} \times h_{k'ut} \times \sigma) + b_{kut}BN_0} \quad (3.3)$$

$$R_{ut} = \sum_{k \in N_{uav}} b_{kut} \times \log_2 (1 + r_{kut}) \quad (3.4)$$

由於式(3.2)至(3.4)的變數之間存在相互依存關係，導致問題成為非線性。本研究觀察其變數之間的消長關係，先將之簡化如式(3.5)，因此每一用戶在每一時段的瞬時通訊量，經由主要服務此用戶之無人機所提供的訊號和他台無人機所造成的雜訊相減得到其值。請注意式(3.5)的值當然與原始式(3.4)不相同，我們的目的主要是想藉由簡化的線性函式來處理複雜的路線規劃問題。因此當求解出路線之後，各時刻所對應的

無人機之功率以及頻寬應該還要再用別的方法（譬如本研究提出的 PSO 演算法）去收斂計算而得。

$$R_{ut} = \sum_{k \in N_{uav}} \left[ b_{kut} \times \left( \frac{p_{kt}}{d_{kut}} - \sum_{k' \in N_{uav} - \{k\}} \frac{p_{k't}}{d_{k'ut}} \right) \right] \quad (3.5)$$

即使式(3.5)已是簡化版本，但仍存在變數間相乘、相除的非線性關係。因此會在接下來數學式中解釋一連串非線性轉換成線性的過程架構。

首先，在計算訊號時，可發現訊號會正比於頻寬與功率相乘，由於變數相乘為非線性關係，因此引用變數  $z_{kut}$  做替換，而此三個變數之間的關係，如式(3.6)所示，在任一時段每一位用戶只會被一台無人機服務，因此每一用戶在每一時段只會接收到一組來自主要服務此用戶無人機的頻寬比例，而每台無人機對於此用戶的訊號則會是當台無人機之發射功率再乘以用戶接收頻寬比例。

$$z_{kut} = p_{kt} \times \sum_{k' \in N_{uav}} b_{k'ut} \quad \forall k \in N_{uav}, \forall u \in N_{user}, \forall t \in T \quad (3.6)$$

但是變數  $z_{kut}$  仍為兩個連續變數相乘之關係，因此我們嘗試將頻寬的連續變數離散成若干等級，並利用分等級之二元變數和其相對應之頻寬量參數來替換掉連續變數，如式(3.7)所示。

$$b_{kut} = a_{kul} \times B[l] \quad \forall k \in N_{uav}, \forall u \in N_{user}, \forall t \in T, \forall l \in L \quad (3.7)$$

當解決頻寬與功率相乘之關係，(3.5)的關係式則會變成如式(3.8)。因此第二步將會緊跟著處理距離變數之問題，而處理方式和第一步相似，同樣是將距離連續變數拆分成一個二元變數和一個距離常數相乘，如式(3.9)，每一時段  $t$ ，式(3.9)都會找尋無人機  $k$  的飛行路線，並計算此路線起點和用戶  $u$  之間的相對距離。

$$R_{ut} = \frac{z_{kut}}{d_{kut}} - \sum_{k' \in N_{uav} - \{k\}} \frac{z_{k'ut}}{d_{k'ut}} \quad (3.8)$$

$$d_{kut} = \sum_{i \in N_{node}} D_{ui} \sum_{(i,j) \in FP} x_{kijt} \quad (3.9)$$

將式(3.9)帶入第(3.8)式，便可展開成如式(3.10)。

$$R_{ut} = \frac{z_{kut}}{\sum_{i \in N_{node}} D_{ui} \sum_{(i,j) \in FP} x_{kijt}} - \sum_{k' \in N_{uav} - \{k\}} \frac{z_{k'ut}}{\sum_{i \in N_{node}} D_{ui} \sum_{(i,j) \in FP} x_{k'ijt}} \quad (3.10)$$

最後引用代表信號的  $s_{kut}$  變數，替換掉非線性的部分，如式(3.11)所示。

$$R_{ut} = s_{kut} - \sum_{k' \in N_{uav} - \{k\}} s_{k'ut} \quad (3.11)$$

### 3.3.3 限制式

本研究考量無人機之路線規劃和資源配，限制式主要分為非線性處理、軌跡、通訊協定、資源限制和求最小瞬時通訊量五大部分。

#### 非線性處理：

數學式(3.12) 表示當無人機在任一時段有服務用戶，那它就應該分配一個等級的頻寬比例給此用戶。當  $y_{kut} = 1$  時表示無人機  $k$  在時段  $t$  會服務用戶  $u$ ，因此無人機  $k$  就會於時段  $t$  分配一個等級  $l$  來服務用戶  $u$ ，使  $\sum_{l \in L} a_{kutl} = 1$ 。而無人機  $k$  在時段  $t$  若沒有服務用戶  $u$ ，即  $y_{kut} = 0$ ，則無人機將不會給此用戶任何等級的頻寬，因此  $\sum_{l \in L} a_{kutl} = 0$ 。

其邏輯可轉換成如限制式(3.12\_1)所示。

$$\begin{aligned} y_{kut} = 1 &\Leftrightarrow \sum_{l \in L} a_{kutl} = 1 \\ y_{kut} = 0 &\Leftrightarrow \sum_{l \in L} a_{kutl} = 0 \end{aligned} \quad (3.12)$$

$$\begin{aligned}
& \sum_l a_{kutl} + M(1 - y_{kut}) \geq 1 \\
& \sum_l a_{kutl} - M(1 - y_{kut}) \leq 1 \\
& \sum_l a_{kutl} \leq My_{kut} \\
& \forall k \in N_{uav}, \forall u \in N_{user}, \forall t \in T, \forall l \in L
\end{aligned} \tag{3.12_1}$$

數學式(3.13)表示當無人機有分配一個頻寬等級給某一用戶，此等級便會對應到相應之頻寬量。因此當  $a_{kutl} = 1$ ，模型將會於  $B$  集合中找到相對應於等級  $l$  的頻寬量並將其值指派給連續變數  $b_{kut}$ ，而當  $\sum_{l \in L} a_{kutl} = 0$  則  $b_{kut}$  也會為 0，表示無人機  $k$  於時段  $t$  沒有給予用戶  $u$  任何頻寬量。

其邏輯可轉換成如限制式(3.13\_1)所示。

$$\begin{aligned}
a_{kutl} = 1 &\Leftrightarrow b_{kut} = B[l] \\
\sum_{l \in L} a_{kutl} = 0 &\Leftrightarrow b_{kut} = 0 \\
b_{kut} + M(1 - a_{kutl}) &\geq B[l] \\
b_{kut} - M(1 - a_{kutl}) &\leq B[l] \quad \forall k \in N_{uav}, \forall u \in N_{user}, \forall t \in T, \forall l \in L \\
b_{kut} \leq M \sum_{l \in L} a_{kutl} &\quad \forall k \in N_{uav}, \forall u \in N_{user}, \forall t \in T
\end{aligned} \tag{3.13_1}$$

數學式(3.14)確保當無人機有分配頻寬給某一用戶，便會計算一個  $z_{kut}$  變數。當  $\sum_{l \in L} a_{kutl} = 1$  時，連續變數  $z_{kut}$  將會等於對應等級  $l$  的頻寬量  $B[l]$  乘以無人機  $k$  於時段  $t$  所發射之功率大小  $p_{kt}$ 。

其邏輯可轉換成如限制式(3.14\_1)所示。

$$\begin{aligned}
\sum_{l \in L} a_{kutl} = 1 &\Leftrightarrow z_{kut} = B[l] \times p_{kt} \\
z_{kut} + M(1 - \sum_{k' \in N_{uav}} a_{k'u'utl}) &\geq B[l] \times p_{kt} \\
z_{kut} - M(1 - \sum_{k' \in N_{uav}} a_{k'u'utl}) &\leq B[l] \times p_{kt} \\
\forall k \in N_{uav}, \forall u \in N_{user}, \forall t \in T, \forall l \in L
\end{aligned} \tag{3.14_1}$$

限制式(3.15)利用二元變數  $x_{kijt}$  找出無人機  $k$  任一時段  $t$  之軌跡，並利用其軌跡找到預先計算好之距離參數，進而得出訊號值。當二元變數  $\sum_{(i,j) \in FP} x_{kijt} = 1$  表示無人機  $k$  在時段  $t$  走  $(i, j)$  線段，因此本研究嘗試找出無人機  $k$  每個時段  $t$  所飛行之線段，並找出其所對應之距離參數  $D_{ui}$ 。

其邏輯可轉換成如限制式(3.15\_1)，利用夾擠的方式求解。

$$\sum_{(i,j) \in FP} x_{kijt} = 1 \Leftrightarrow s_{kut} = \frac{z_{kut}}{D_{ui}} \quad (3.15)$$

$$s_{kut} + M(1 - \sum_{(i,j) \in FP} x_{kijt}) \geq \frac{z_{kut}}{D_{ui}} \quad (3.15\_1)$$

$$s_{kut} - M(1 - \sum_{(i,j) \in FP} x_{kijt}) \leq \frac{z_{kut}}{D_{ui}}$$

$$\forall k \in N_{uav}, \forall u \in N_{user}, \forall i \in N_{node}, \forall t \in T$$

最後可由限制式(3.16\_1)針對每一用戶計算每一時段所獲得的瞬時通訊量。此限制式會先找出在時段  $t$  哪台無人機  $k$  為用戶  $u$  的主要服務無人機，並將主要服務無人機之訊號  $s_{kut}$  減掉其它非主要服務無人機之雜訊干擾總和  $\sum_{k' \in N_{uav} - \{k\}} s_{k'ut}$ ，但由於本研究為了適用於線性求解，而將信噪比公式由原本的信號除以干擾噪音的概念線性轉換為信號減掉干擾噪音，因此使得信噪比有負數值的可能，進而造成模型找不到可行解，為解決此問題，本研究規定當信噪比為負數值時，模型將會指定此信噪比為 0。

其邏輯可轉換成如限制式(3.16\_1)，利用夾擠的方式求解。

$$R_{ut} = \max\{s_{kut} - \sum_{k' \in N_{uav} - \{k\}} s_{k'ut}, 0\} \quad (3.16)$$

$$R_{ut} + M(1 - SNR_{ut}) \geq s_{kut} - \sum_{k' \in N_{uav} - \{k\}} s_{k'ut} \quad (3.16\_1)$$

$$R_{ut} - M(1 - SNR_{ut}) \leq s_{kut} - \sum_{k' \in N_{uav} - \{k\}} s_{k'ut}$$

$$R_{ut} + M(SNR_{ut}) \geq 0$$

$$R_{ut} - M(SNR_{ut}) \leq 0$$

$$\forall k \in N_{uav}, \forall u \in N_{user}, \forall t \in T$$

接著限制式(3.16\_2)則針對訊號及信噪比之間的關係做限制，規定當訊號比干擾大時（即  $s_{kut} - \sum_{k' \in N_{uav} - \{k\}} s_{k'u't} > 0$  時），則指定  $SNR_{ut}$  為 1。

$$\begin{aligned} s_{kut} - \sum_{k' \in N_{uav} - \{k\}} s_{k'u't} &< SNR_{ut} \\ s_{kut} - \sum_{k' \in N_{uav} - \{k\}} s_{k'u't} &\leq M(SNR_{ut} - 1) \\ \forall k \in N_{uav}, \forall u \in N_{user}, \forall t \in T \end{aligned} \quad (3.16_2)$$

**軌跡限制：**

限制式(3.17)限制每一架無人機都由原點出發。 $x_{kijt}$  表示無人機  $k$  在時段  $t$  是否走  $(i, j)$  線段，因此本研究於時段 0 時限制無人機  $k$ ，只能走  $(1, j)$  線段。

$$\sum_{(1,j) \in FP} x_{k1j0} = 1 \quad \forall k \in N_{uav} \quad (3.17)$$

限制式(3.18)限制每一架無人機的移動只會有一種可能。在還沒有此限制條件前，因為飛行模式能沿著格點飛行也能懸停於原格點，故模型中無人機  $k$  在時段  $t$  飛行路線會有 3 至 5 種可能，因此加總所有  $(i, j)$  線段，規定無人機只能選擇其中一種路線飛行。在實例演練中，由於我們無法確定無人機起點位置，所以會藉由加總  $i$ ，找到其起始位置。

$$\sum_{i \in N_{node}} \sum_{(i,j) \in FP} x_{kijt} = 1 \quad \forall k \in N_{uav}, \forall t \in T \quad (3.18)$$

限制式(3.19)為流量守恆，無人機進入節點的數量與飛出的會相同。當時段  $t$  時無人機  $k$  會由  $i$  飛向  $j$ ，在時段  $(t+1)$  必會再從  $j$  點飛出。

$$\sum_{(i,j) \in FP} x_{kijt} = \sum_{(j,i) \in FP} x_{kji(t+1)} \quad \forall k \in N_{uav}, \forall j \in N_{node}, \forall t \in T \quad (3.19)$$

**通訊協定限制：**

限制式(3.20)規定每一時段  $t$ ，每一用戶  $u$  都要被某架無人機  $k$  服務。

$$\sum_{k \in N_{uav}} y_{kut} = 1 \quad \forall u \in N_{user}, \forall t \in T \quad (3.20)$$

限制式(3.21)計算時刻  $t$  每架無人機  $k$  和用戶  $u$  距離。當二元變數  $x_{kijt} = 1$  表示無人機  $k$  在時段  $t$  走  $(i, j)$  線段，因此起始點  $i$  便會對應到相對的  $D_{ui}$  參數。

$$d_{kut} = \sum_{i \in N_{node}} D_{ui} \sum_{(i,j) \in FP} x_{kijt} \quad \forall k \in N_{uav}, u \in N_{user}, \forall t \in T \quad (3.21)$$

限制式(3.22)主要要找出每時段每台無人機和用戶間最短距離  $d_{kut}$ ，並進行無人機和用戶的配對。當時段  $t$  用戶  $u$  找到離他距離最近之無人機，便會指派此無人機  $k$  來服務用戶  $u$ ，使  $y_{kut} = 1$ 。其邏輯可轉換成如限制式(3.22\_1)，利用夾擠的方式求解。

$$y_{kut} = 1 \Leftrightarrow d_{ut} = d_{kut} \quad (3.22)$$

$$\begin{aligned} d_{ut} - M(1 - y_{kut}) &\leq d_{kut} \\ d_{ut} + M(1 - y_{kut}) &\geq d_{kut} \\ d_{ut} &\leq d_{kut} \end{aligned} \quad \forall k \in N_{uav}, \forall u \in N_{user}, \forall t \in T \quad (3.22\_1)$$

**通訊協定限制：**

限制式(3.23)限制發射功率的上下限。其表示無人機  $k$  於時段  $t$  所發射之功率  $p_{kt}$  至少要大或等於 0，而最多不能超過功率上限參數  $P_{\max}$ 。

$$0 \leq p_{kt} \leq P_{\max} \quad \forall k \in N_{uav}, \forall t \in T \quad (3.23)$$

限制式(3.24)規定每時段每台無人機服務用戶的頻寬加總的上限。模型會針對每台無人機  $k$  於時段  $t$  所分配給每一位用戶  $u$  的頻寬量  $b_{kut}$  進行加總，並限制頻寬加總要等於 1，以最大化利用其頻寬。

$$\sum_{u \in N_{user}} b_{kut} = 1 \quad \forall k \in N_{uav}, \forall t \in T \quad (3.24)$$

**求最小瞬時通訊量：**

限制式(3.25)限制模型在每一時段  $t$  只會有一個接收瞬時通訊量最小之用戶  $u$ 。

$$\sum_{u \in N_{user}} w_{ut} = 1 \quad \forall t \in T \quad (3.25)$$

限制式(3.26)會比對每位用戶  $u$  在每一時段  $t$  的瞬時通訊量  $R_{ut}$ ，並找出最小接收瞬時通訊量  $MinR_t$ ，指定其  $w_{ut} = 1$ 。

其邏輯可轉換成如限制式(3.26\_1)，利用夾擠的方式求解。

$$w_{ut} = 1 \Leftrightarrow MinR_t = R_{ut} \quad (3.26)$$

$$\begin{aligned} MinR_t + M(1 - w_{ut}) &\geq R_{ut} \\ MinR_t - M(1 - w_{ut}) &\leq R_{ut} \\ MinR_t &\leq R_{ut} \\ \forall u \in N_{user}, t \in T \end{aligned} \quad (3.26\_1)$$

### 3.4 變數與限制式個數估算

本節以  $4 \times 4$  的網格圖（共有 16 個節點）、2 台無人機及 5 位用戶，時空網路圖時長設定 10 為例，參數相關設定如表 3-1 所示，進行如表 3-2 及表 3-3 的數學模式變數與限制式個數估算。

表 3-1 估算變數與限制式個數之參數設定值

節點個數	可行路線數	無人機	頻寬等級	用戶	網路時長
$ N_{node} $	$ FP $	$ N_{uav} $	$ L $	$ N_{user} $	$ T $
16	64	2	6	5	10

表 3-2 變數總個數估算

本研究		
Variable	Quantity Formula	Numerical Value
$x_{kijt}$	$ N_{uav}  \cdot  FP  \cdot  T $	1280
$y_{kut}$	$ N_{uav}  \cdot  N_{user}  \cdot  T $	100
$a_{kutl}$	$ N_{uav}  \cdot  N_{user}  \cdot  T  \cdot  L $	600
$w_{ut}$	$ N_{user}  \cdot  T $	50
$b_{kut}$	$ N_{uav}  \cdot  N_{user}  \cdot  T $	100
$p_{kt}$	$ N_{uav}  \cdot  T $	20

$d_{kut}$	$ N_{uav}  \cdot  N_{user}  \cdot  T $	100
$d_{ut}$	$ N_{user}  \cdot  T $	50
$r_{kut}$	$ N_{uav}  \cdot  N_{user}  \cdot  T $	100
$z_{kut}$	$ N_{uav}  \cdot  N_{user}  \cdot  T $	100
$s_{kut}$	$ N_{uav}  \cdot  N_{user}  \cdot  T $	100
$R_{ut}$	$ N_{user}  \cdot  T $	50
$MinR_t$	$ T $	10
<b>變數總個數</b>		2660

表 3-3 限制式總個數比較

本研究		
Constraint ID	Quantity Formula	Numerical Value
(3.12)	$ N_{uav}  \cdot  N_{user}  \cdot  T  \cdot  L $	600
(3.13)	$ N_{uav}  \cdot  N_{user}  \cdot  T  \cdot  L  +  N_{uav}  \cdot  N_{user}  \cdot  T $	700
(3.14)	$ N_{uav}  \cdot  N_{user}  \cdot  T  \cdot  L $	600
(3.15)	$ N_{uav}  \cdot  N_{user}  \cdot  N_{node}  \cdot  T $	1600
(3.16)	$ N_{uav}  \cdot  N_{user}  \cdot  T $	100
(3.17)	$ N_{uav} $	2
(3.18)	$ N_{user}  \cdot  T $	50
(3.19)	$ N_{uav}  \cdot  N_{node}  \cdot  T $	320
(3.20)	$ N_{user}  \cdot  T $	50
(3.21)	$ N_{uav}  \cdot  N_{user}  \cdot  T $	100
(3.22)	$ N_{uav}  \cdot  N_{user}  \cdot  T $	100
(3.23)	$ N_{uav}  \cdot  T $	20
(3.24)	$ N_{user}  \cdot  T $	50
(3.25)	$ N_{user}  \cdot  T $	50
(3.26)	$ N_{user}  \cdot  T $	50
<b>限制式總個數</b>		4392

### 3.5小結

現實的決策問題蠻容易遇到變數與變數之間存在非線性的關係，或者是限制式與目標式含有非線性的項目。非線性項目的處理方式並無定論，在本章節中，我們觀察項目之間的消長關係，試圖利用相似邏輯重新編寫成線性的數學關係式，希冀能透過線性轉換來求得相似解。然而，因為線性轉換的過程中會衍生多項的變數和限制式，因此隨模型規模變大，也會耗費大量求解時間，效率不佳。為處理此效率問題，第四辛將設計並實作更有效率的啟發式演算法，以作為求解此問題的替代方案。



## 第四章 無人機最佳飛行路線與通訊設定之演算法

在第三章中，本研究通過 Gurobi 求解器進行混整數規劃模型求解，但隨著問題規模的擴大，譬如擴大服務覆蓋範圍、增加需求點及無人機數等，最佳化軟體的求解時間會大幅增加，實用性將大打折扣。因此本章節在觀察無人機飛行方式和參考多篇文獻後，提出滾動式求解機制(Rolling-horizon Mechanism)，使問題能在短時間內便能獲得品質不錯的解。此外，第二章提及大部分文獻採用塊座標下降演算法(Block Coordinate Descent)進行求解，為模仿該演算法的部分機制作為對照，本研究同時提出粒子群演算法(Particle Swarm Optimization, PSO)來求解。

### 4.1 滾動式求解機制

由於一次求解整個完整問題所花費的時間過長，本研究嘗試將總服務時長進行分組批次求解，期望藉由減少每次求解問題的規模，使執行時間得以加速。但此求解方式同時也會影響求解的效果，無法保證得到最佳解。

#### 4.1.1 滾動式求解概念

當求解整個完整的服務時長時，由於模型可以綜觀全局，因而可以找到全局最佳解，只是過大的規模會導致執行時間過長。相反地，當每次求解的總服務時長越調越少，理應會越快得到其解，但解的品質也會隨之降低。本研究為了在求解時間和求解品質之間取得平衡，以 2 台無人機服務 4 位用戶(服務範圍被平分為 9 個格點)為例，分別以不同的總服務時間段測試（如表 4-1），來比較其程式執行時間，可從表格清楚看見當總服務時段大於 4 時，程式執行時間則超過本研究可承受範圍，因此將四個時間段劃分為一組來求解較為合適。為避免時段切割方式的求解會因為計劃區間太短（僅四個時段）而落入局部最佳解，本研究採取「看四取一」的策略，亦即在每一次求解四段區間之後，固定其第一個時段所求解的軌跡和通訊設定，讓其餘三個時段的軌跡將成為下一次迭代中的初始解（但可再被重算調整），再重複此機制持續進行優化求解。如圖 4-1，在第一次的迭代中，模型會求解時間段  $t=1$  到  $t=4$  這四個時段的

軌跡和通訊設定，再將所求得的  $t=1$  的軌跡和通訊設定固定，讓求得的  $t=2$  到  $t=4$  所求之軌跡解當其第二階段的初始解，再求解  $t=2$  到  $t=5$  等四個時間段。

表 4-1 執行時間比較

	總服務時長				
	$ T =1$	$ T =2$	$ T =3$	$ T =4$	$ T =5$
<b>Running Time</b>	0.05s	0.2s	16.03s	543.29s	9559.68s

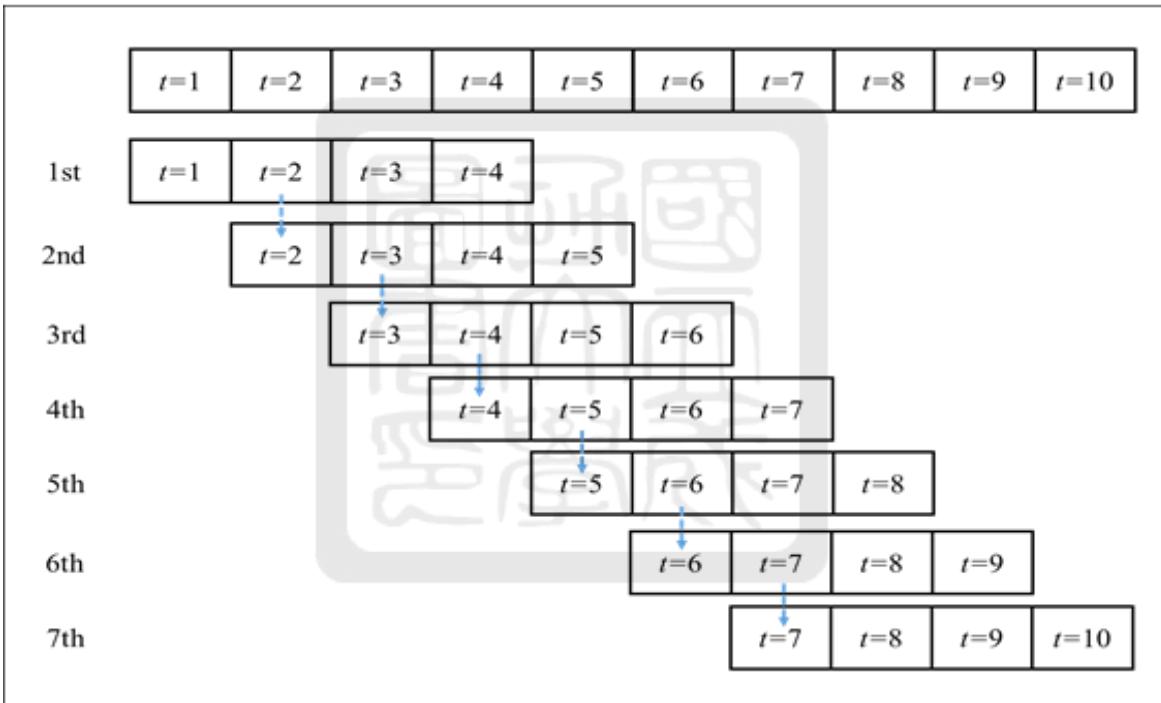


圖 4-1 滾動式方法示意圖

#### 4.1.2 滾動式求解機制虛擬碼

本研究設計之滾動式求解如表 4-2 所示，首先輸入資料要須要告知模型欲求解之總時長  $T$  及每一次迭代中欲求解的時長  $T_p$ 。經由第 1 列計算，便能得知需要進行的迭代次數  $iter$ ，同時程式會初始化一個軌跡  $Traj$ ，使每一台無人機初始位置都位於原點。在每一次迭代中，模型會經由列 3、4 計算出此次迭代中的初始時刻  $Stime$  及結束

時刻  $Etime$ ，並於列 6 紿定軌跡初始解  $init\_pos$  來提高求解品質。為避免時刻間飛行軌跡的不連貫，模型將會於列 7~10 固定上一時刻的結束位置，當作此刻的出發位置再來求解其軌跡。譬如當無人機 1 於  $t=1$  時會由點 1 飛向點 4， $t=2$  時無人機 1 則會由點 4 出發向其它格點飛行。當設定完成後，程式會經由數學模型求解並記錄每一時刻的軌跡  $Traj_t$ 、通訊設定  $p_t$ ,  $b_t$  及最小瞬時通訊量， $MinR_t$ 。

表 4-2 滾動式求解方法設計邏輯

---

### **Rolling Solution**

---

**Data :**  $T, T_p$

1. Initialize  $Traj$ . Let  $iter = T - T_p + 1$
  2. **for**  $i = 0$  to  $iter-1$  **do**
  3.      $S_{time} = i$
  4.      $Etime = i + T_p - 1$
  5.     **for**  $t = S_{time}$  to  $Etime$  **do**
  6.          $init\_pos_t = Traj_t$
  7.         **if**  $t = 0$  **then**
  8.              $S_{node} = \text{original node}$
  9.         **else**
  10.          $S_{node} = Traj_{t-1, Etime}$
  11.          $Model(i, S_{node}, S_{time}, Etime)$
  12.         update  $Traj_t, p_t, b_t, MinR_t$ , where  $\forall t \in [S_{time}, \dots, Etime]$
- 

**Result :**  $Traj, p, b, MinR$

---

## 4.2 雙階層粒子群演算法

本研究使用粒子群演算法，主要是認為大多數文獻所提出之塊座標下降演算法(Block Coordinate Descent)不甚合理，但其調整解再收斂的方式的確符合直覺。因此本研究擬採取具有類似「調整解再收斂」機制，且考慮無人機軌跡（因）與其功率、頻寬比（果）的關係，以雙階層粒子群演算法（外層先決定軌跡後，再由內層決定其功率、頻寬比），分別優化完整總時長  $T$  的無人機軌跡及通訊設定，並經由一次次的迭代中相互影響而逐漸收斂。

#### 4.2.1 粒子群演算法概念

粒子群演算法(Particle Swarm Optimization, PSO)始於 1995 年，由 Kennedy & Eberhart (1995)提出，為一種啟發式演算法。此演算法是模擬鳥群覓食行為設計而得，常被應用於求解最佳化問題上。

在粒子群演算法中，如圖 4-2 所示，每個問題的解都是搜尋空間中的一隻鳥，模型中將其視為粒子。由於粒子會依照自身經驗和參考並學習其它粒子的經驗來決定飛行方向，因此演算法的全部粒子將會漸漸地往較好的方向移動。

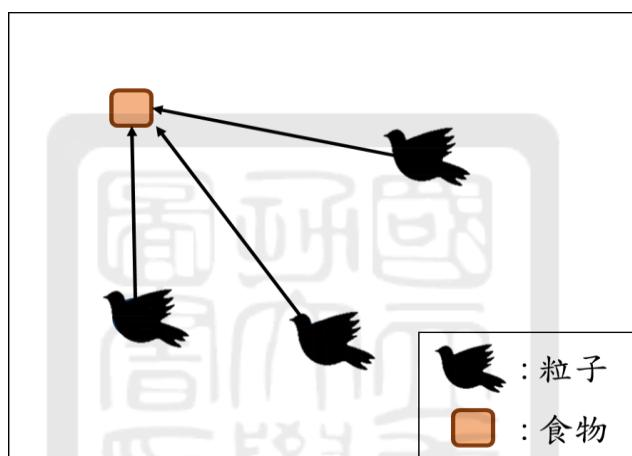


圖 4-2 粒子群演算法示意圖

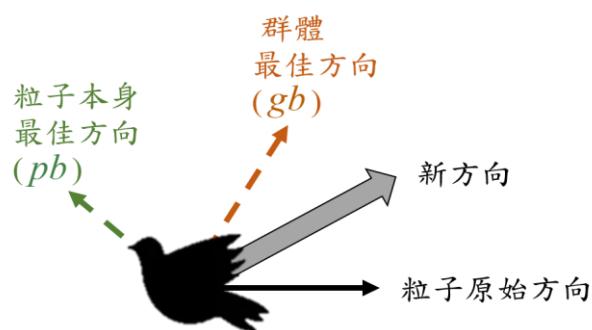


圖 4-3 粒子方向示意圖

每一個粒子都擁有兩大屬性及  $d$  個維度。粒子的兩大屬性分為位置和速度，其中速度代表指引粒子移動的方向，如圖 4-3，粒子每次的移動除了會從自己自身的經驗

中找到最好的位置(*pb*)，更會因為發現有別的粒子比自己更好(*gb*)，就去參考並調整方向，得到一全新的移動方向。而一個粒子在粒子群演算法中的*d*個維度，表示粒子群*I*會在*d*維度的解空間中進行移動，找到每一個粒子在*d*維度的位置。

每次迭代的速度轉換公式，如式(4.1)所示。其中 $v_{id}^t$ 表示第*t*次迭代中，第*i*個粒子裡第*d*維度的速度，而這個新速度主要會受到以下三大因素所影響：

- (1)原本速度慣性：上一次迭代中原本速度方向。
- (2)自身經驗(*pb*)：綜合自己每次迭代經驗中最好的移動方向。
- (3)其它粒子的經驗(*gb*)：綜觀所有粒子目前所有迭代經驗中最好的移動方向。

在第一大因素中， $v_{id}^{t-1}$ 代表在上一次迭代中，第*i*個粒子裡第*d*維度原本的速度，*w*則代表此慣性因素的權重值，通常設定為1，因素二和三中， $c_1$ 和 $c_2$ 分別定義為認知常數和社會常數，其意義同等於被自身經驗(*pb*)及其它粒子的經驗(*gb*)影響之權重，通常設定為2，而rand()則是一個介於[0,1]的隨機變數。

$$v_{id}^t = \underbrace{w \times v_{id}^{t-1}}_1 + \underbrace{c_1 \times \text{rand}() \times (pb_{id} - x_{id}^{t-1})}_2 + \underbrace{c_2 \times \text{rand}() \times (gb_d - x_{id}^{t-1})}_3 \quad (4.1)$$

每次迭代下，粒子的位置如式(4.2)所示。 $x_{id}^t$ 表示第*t*次迭代中，第*i*個粒子裡第*d*維度的位置，其位置會經由上一階段迭代的最終位置 $x_{id}^{t-1}$ 出發，考慮此階段經由學習、調整的新速度 $v_{id}^t$ 引導而得。

$$x_{id}^t = x_{id}^{t-1} + v_{id}^t \quad (4.2)$$

如圖 4-4 所示，粒子群演算法會經由一次一次的迭代重複以下步驟，來找尋最佳位置：

1. 初始化粒子：隨機分配每個粒子的位置和速度，並將其位置設定為此粒子之個別最佳位置 *pb*，再找出目前群體最佳位置 *gb*。
2. 計算各粒子適應值：每個粒子在*d*維度空間所求之解都會丟入所設定的目標式中計算適應值。

3. 更新  $pb$  和  $gb$ ：經由第 2 步，可獲得每個粒子所對應之適應值。每個粒子就會各自和這個新的位置所對應之適應值做比較，若數值較原本好，則會更新個別粒子位置  $pb$ 。除此之外，演算法也會在每一次迭代中比較群體最佳位置  $gb$ ，因此，只要有粒子在此次迭代中新位置所對應之適應值比之前還好，則會更新群體最佳位置  $gb$ 。
4. 更新粒子位置、速度：如下圖所述，每個粒子都會因為本身經驗和別人的影響而改變其移動速度，因此藉由第 3 步驟可以清楚的了解此次迭代中的  $pb$  和  $gb$ ，進而透過公式計算下一次迭代的位置和速度。
5. 終止條件：終止條件用來控制整個程式何時結束，如果程式未達終止條件則會持續迭代直到符合終止條件為止，在此將以迭代次數上限當成終止條件。

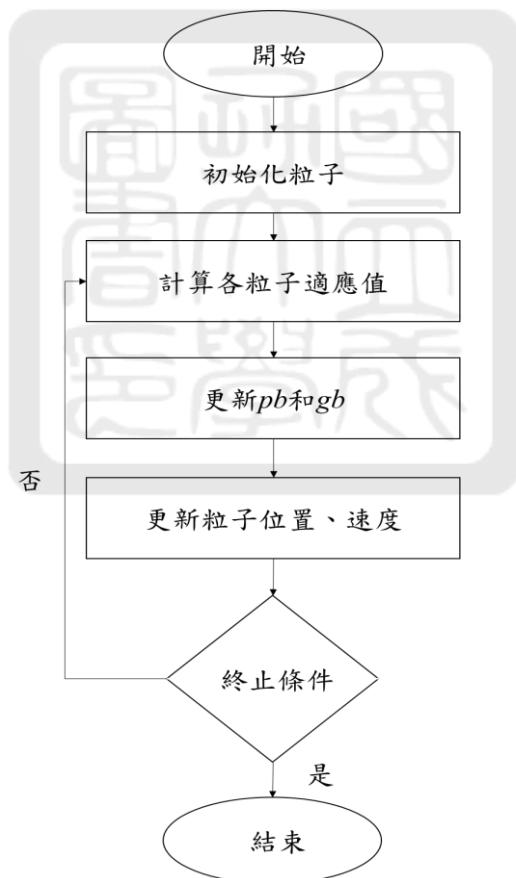


圖 4-4 粒子群演算法流程圖

#### 4.2.2 雙階層粒子群演算法虛擬碼

本研究所設計之粒子群演算法主要由兩層 PSO 建構而成，分別為  $PSO_{traj}$  及  $PSO_{pb}$ 。如圖 4-5 所示，本研究會先在解空間中投入  $ParN_{traj}$  個粒子進入第一層  $PSO$ ，即  $PSO_{traj}$  中，求解出每一時刻  $t$  無人機  $k$  之飛行模式，經轉換得到各無人機  $k$  每刻之停留格點，接著，針對每一時刻  $t$  演算法會呼叫第二層的  $PSO_{pb}$ ，來收斂出基於時刻  $t$  每一台無人機位置所需之通訊設定。

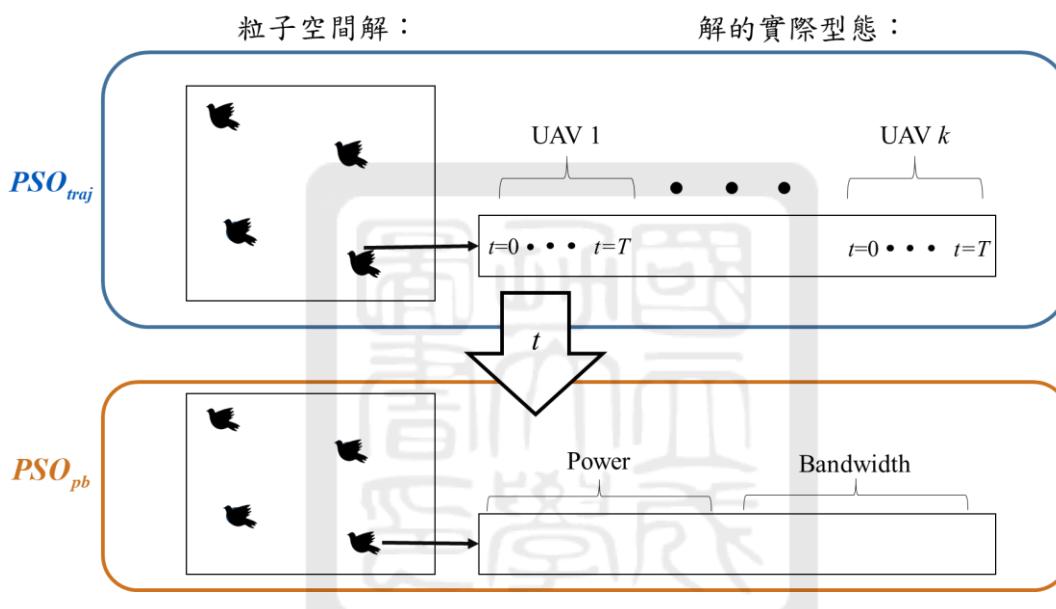


圖 4-5 雙階層 PSO 邏輯圖

表 4-3  $PSO_{traj}$  和  $PSO_{pb}$  比較

	$PSO_{traj}$	$PSO_{pb}$
目的	收斂總服務時長內所有無人機之飛行軌跡	收斂某一時刻所有無人機之通訊設定（發射功率和頻寬比）
適應值	最大化總服務時長之最小瞬時通訊量總合	最大化個別時刻之最小瞬時通訊量

$PSO_{traj}$  和  $PSO_{pb}$  間之求解目的與比較如表 4-3 所述， $PSO_{traj}$  目的是收斂總服務時長內的飛行軌跡，而  $PSO_{pb}$  則會針對每一時刻求得較佳的通訊設定，並找到最小瞬時通訊量，每一時刻之最小瞬時通訊量再經由  $PSO_{traj}$  加總，以達到最大化最小瞬時通訊量總和之目標。

以兩台無人機 ( $N_{uav} = 2$ ) 在  $|T| = 3$  ( $t = 0, 1, 2$ ) 的總服務時長裡服務三位用戶 ( $N_{user} = 3$ ) 為例，來解說雙層 PSO 的主要任務。外層 PSO ( $PSO_{traj}$ )，在每一次迭代中，每個粒子表示所有無人機在總服務時長裡的軌跡路線，含有  $|N_{uav}| \times |T|$  個維度，即  $2 \times 3 = 6$  個維度。如圖 4-6，第一次迭代中，無人機 1 的飛行軌跡為  $1 \rightarrow 2 \rightarrow 3$ ，無人機 2 的飛行軌跡為  $1 \rightarrow 4 \rightarrow 7$ ，經過  $Maxiter_{traj}$  次迭代與優化，無人機 1 的飛行軌跡會更新為  $1 \rightarrow 2 \rightarrow 5$ ，無人機 2 的飛行軌跡則變為  $1 \rightarrow 4 \rightarrow 5$  以更接近最佳解。

每次以  $PSO_{traj}$  求解完當次迭代之所有軌跡後，會針對每一個時刻所有無人機的位置進行內層 PSO ( $PSO_{pb}$ )，做無人機和用戶間的配對並找到有限迭代次數中較佳的通訊設定。以圖 4-6 第一次迭代中  $t=0$  為例，可看見無人機 1 和 2 所在位置都在格點 1，因此粒子群演算法會基於此位置做  $PSO_{pb}$  找到兩台無人機和三位用戶間要如何配置，無人機又該發射多少的發射功率和頻寬比大小。

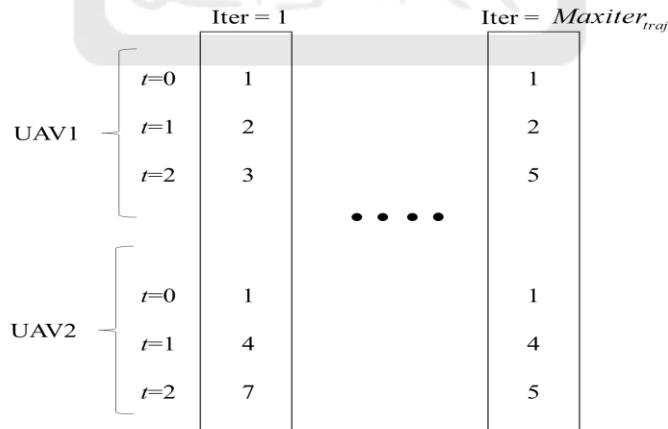


圖 4-6  $PSO_{traj}$  示意圖

#### 4.2.3 $PSO_{traj}$ 虛擬碼

接著本研究將細說外層  $PSO$  ( $PSO_{traj}$ ) 的設計構想， $PSO_{traj}$  主要目的為求解總服務時長內所有無人機飛行軌跡，來達到最大化總服務時長中最小瞬時通訊量的總和。

由於在  $PSO_{traj}$  收斂飛行軌跡時，其設定仍須遵循數學模型之問題設定，因此我們仍需將無人機飛行範圍平均劃分成網格狀，此設定會導致無人機只能沿著格點移動至相鄰格點，而非在任兩點間都可隨意飛行，因此為了方便建構演算法，本研究的粒子將不會直接設定為總服務時長內飛行所經格點，而是無人機在每個時刻之行為模式，此行為模式代表的就是懸停、向上、下、左、右移動這五種飛行模式。我們會先去界定一個有  $Nb_i$  個飛行模式的格點  $i$  中，每個機率範圍內所對應之飛行模式，接著演算法會隨機生成一個介於  $[0,1]$  範圍內的值，依其值對應至無人機下一時刻的飛行模式。

如圖 4-7 所示，假設我們將服務範圍平分成  $3 \times 3$  的格點，當無人機位於格點 1 時，在下一時刻無人機可以懸停於原本格點，也可向上或向右沿格點飛行一單位距離，因此，有  $Nb_1 = 3$  個行為模式。當粒子的解坐落於  $[0,1/3)$  區間，下一時刻無人機將懸停於格點 1，當解於  $[1/3,2/3)$  區間，無人機將會於下一時刻向上飛到格點 2，如果粒子解在  $[2/3,1]$  區間中，無人機則會在下一時刻向右飛向格點 4。

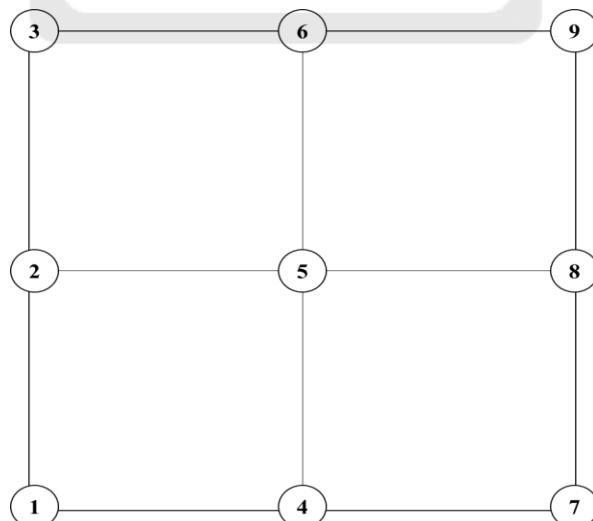


圖 4-7 服務範圍

整個  $PSO_{traj}$  中包含五個函式，每個函式有其不同之功能，可參考如表 4-4 之比較。

表 4-4  $PSO_{traj}$  函式內容及目的

$PSO_{traj}$	
函式	目的
<code>__init__( )</code>	初始化函式，隨機生成粒子所對應位置及速度。
<code>fitness( )</code>	適應值函式，根據目標式設定， $PSO_{traj}$ 目的為最大化總服務時長之最小瞬時通訊量加總。
<code>update( )</code>	更新粒子位置、速度函式，於每次迭代中根據學習及經驗重新更新粒子位置及速度。
<code>gBehavior( )</code>	位置轉換函式，依演算法生成的值，界定下一時刻無人機所在位置。
<code>psol( )</code>	PSO 主函式，在每一次迭代中，找 $pb$ 及 $gb$ 。

表 4-5 的函式主要目的在初始化所有粒子。如第 3、4 列所示，我們需要依照其粒子位置及速度上下限，隨機產生  $j$  個維度解空間的  $i$  個粒子及其移動速度，接著，藉由 5 到 9 列先把每個粒子目前的位置設定為個別最佳  $pb$ ，並找出所有粒子中適應值最高的位置設定為群體最佳  $gb$ 。

表 4-5  $PSO_{traj}$  初始化函式

---

**Procedure `__init__( )`**

---

1. **for**  $i = 1$  to  $ParN_{traj}$  **do**
  2.     **for**  $j = 1$  to  $Dim_{traj}$  **do**
  3.          $x_{i,j} = rand(l\_traj, u\_traj)$
  4.          $v_{i,j} = rand(vL, uL)$
  5.          $pb_i = x_i$
  6.          $fit = fitness(pb_i)$
  7.         **if**  $fit > opt$  **then**
  8.              $gb = pb_i$
  9.          $opt = fit$
- 

表 4-6 的函式主要目的在計算每一個粒子的適應值，此適應值即為本研究之目標，最大化總服務時長之最小瞬時通訊量總和。首先，函式會先藉由第 2 列確定每個

粒子中每台無人機的飛行軌跡，再經由第 5、6 列來呼叫  $PSO_{pb}$  來求解每時刻每台無人機的通訊設定，並加總服務時長內之所有最小瞬時通訊量當適應值。

表 4-6  $PSO_{traj}$  適應值函式

<b>Procedure</b> fitness( $x_i$ )
1. <b>for</b> $j = 1$ to $Dim_{traj}$ <b>do</b>
2.     Assign $GridPosition_j$ by $gBehavior(x_{i,j}, GridPosition_{j-1})$
3. <b>for</b> $t = 0$ to $T$ <b>do</b>
4.     Assign the position to $UAVPosition_t$
5. $MinR_t = PSO_{pb}(UAVPosition_t)$
6. $SumMinR += MinR_t$
7. <b>return</b> $SumMinR$

表 4-7 的函式主要目的為更新每次迭代之粒子位置及速度，如第 2 列。接著，我們將會針對每個粒子的位置及速度中每個維度檢查是否超出其所限制之上下限。如果超出其限制，則會強制其值等於上\下限數值。本研究限制位置和速度範圍分別為 [0,1] 和 [-1,1]，因此，當位置中任一維度解超出 1，則會將其值強制指定為 1，而當有任一維度解小於 0，此函式則會將其值指定為 0。更新完的位置解同樣需藉由列 5 到 8 找出新一輪的  $pb$  及  $gb$ 。

表 4-7  $PSO_{traj}$  更新函式

<b>Procedure</b> update( )
1. <b>for</b> $i = 1$ to $ParN_{traj}$ <b>do</b>
2.     Calculate (4.1) and (4.2) to get the $v_i$ and $x_i$
3. <b>for</b> $j = 1$ to $Dim_{traj}$ <b>do</b>
4.         Adjust $x_{i,j}$ and $v_{i,j}$ when they exceed the upper and lower bound
5. <b>if</b> $fitness(x_i) > fitness(pb_i)$ <b>then</b>
6. $pb_i = x_i$
7. <b>if</b> $fitness(x_i) > fitness(gb)$ <b>then</b>
8. $gb = x_i$

表 4-8 的函式主要目的就是要將粒子解中的行為模式轉換成總服務時長內每個時刻飛行所經的格點。首先，我們必須告訴此函式無人機目前所停格點  $GridPosition_{j-1}$ ，再針對不同格點所會產生的不同行為模式  $x_{i,j}$ ，來判斷粒子解所坐落區間並告知下一時刻無人機所在格點  $GridPosition_j$ 。

表 4-8  $PSO_{traj}$  位置轉換函式

---

<b>Procedure</b> $gBehavior(x_{i,j}, GridPosition_{j-1})$
1. <b>for</b> $n = 1$ to $ N_{node} $ <b>do</b>
2. <b>if</b> $GridPosition_{j-1} = n$ <b>then</b>
3.         Assign the next position $GridPosition_j$ based on the probability of the action

---

表 4-9 為粒子群演算法的主體，粒子群演算法會在粒子的更新、計算適應值和找  $pb$  和  $gb$  的循環中不斷進行，直到到達最大迭帶次數  $Maxiter_{traj}$  而終止。

表 4-9  $PSO_{traj}$  粒子群演算法函式

---

<b>Procedure</b> $psol()$
1. <b>for</b> $gen = 1$ to $Maxiter_{traj}$ <b>do</b>
2. $update()$
3. <b>if</b> $fitness(gb) > fitness(Best_{final})$ <b>then</b>
4. $Best_{final} = gb$
5. $optValue = fitness(Best_{final})$

---

#### 4.2.4 $PSO_{pb}$ 虛擬碼

內層的  $PSO_{pb}$  設計構想與外層的  $PSO_{traj}$  類似。只是  $PSO_{pb}$  相較於  $PSO_{traj}$  已經不需要去收斂軌跡，因此也不需要位置轉換函式  $gBehavior()$ ，而其餘函式照舊。 $PSO_{pb}$  主要目的為在每一時刻無人機位置已知的情況下，求解每一台無人機之通訊設定，得其每一時刻最小瞬時通訊量。

$PSO_{pb}$  中每一個粒子會含有  $(|N_{uav}| + |N_{user}|)$  個維度，表示  $|N_{uav}|$  台無人機所發射功率及  $|N_{user}|$  個用戶所接收到頻寬比大小。發射功率及頻寬比的決策變數上下限都與數學模型一樣，最大的不同在於粒子群演算法不再受限於線性求解，因此，我們在求解時不需將頻寬比離散為若干等級，而是可以選擇更符合現實的連續性頻寬比。

表 4-10 的函式主要目的在初始化所有粒子的位置及其速度。首先，我們會接收到来自  $PSO_{traj}$  所得到的當時刻無人機位置  $UAVPosition_t$ ，在初始化過程中函式要針對每台無人機及用戶位置，找到對於每位用戶距離最近的無人機進行配對。接著依照其粒子位置及速度上下限，隨機產生  $j$  個維度解空間的  $i$  個粒子位置及其移動速度。

由於每台無人機要符合分配給用戶之總頻寬比剛好等於 1 的通訊協定，因此我們

將在第 9 至 12 列進行頻寬的調整。起初，函式會檢查每一台無人機所分配出去的頻寬比總和，如不等於 1，則藉由無人機目前發射頻寬比總和  $TransmitB$  等比例調整用戶所接收之頻寬比。以一台無人機同時服務用戶一和二，而兩用戶所接收之頻寬比分別為 0.7 和 0.5 為例，無人機分配出去的頻寬比總和為  $0.7+0.5=1.2$ ，不等於 1，經列 12 調整，用戶一將收到  $\frac{0.7}{1.2}=0.583$  的頻寬比，而用戶二則收到  $\frac{0.5}{1.2}=0.417$  的頻寬比。

當所有粒子都調整完，最後會藉由第 13 至 17 列把每個粒子目前的位置設定為  $pb$  並找出所有粒子中適應值最高的位置設定為  $gb$ 。

表 4-10  $PSO_{pb}$  初始化函式

---

**Procedure  $_init_$ ( $UAVPosition_t$ )**

---

1. *Find the pair with the shortest distance between the UAV and the user*
  2. **for**  $i = 1$  to  $ParN_{pb}$  **do**
  3.     **for**  $j = 1$  to  $Dim_{pb}$  **do**
  4.         **if**  $j \leq N_{uav}$  **then**
  5.              $x_{i,j} = rand(L_p, U_p)$
  6.         **else**
  7.              $x_{i,j} = rand(L_b, U_b)$
  8.          $v_{i,j} = rand(L_v, U_v)$
  9.         **for**  $k = 1$  to  $|N_{uav}|$  **do**
  10.             **for**  $u = 1$  to  $|N_{user}|$  **do**
  11.                 **if** the total bandwidth transmitted by UAV > upper bound **then**
  12.                  $x_{i,N_{uav}+u-1} = x_{i,N_{uav}+u-1} / TransmitB$
  13.      $pb_i = x_i$
  14.      $fit = fitness(pb_i)$
  15.     **if**  $fit > optValue$  **then**
  16.          $gb = pb_i$
  17.      $optValue = fit$
- 

下表的函式主要目的在計算每一個粒子的適應值，我們會在一開始用夏農公式計算每一用戶在那一時刻裡所接收到的瞬時通訊量，並找到其最小的瞬時通訊量做為  $PSO_{pb}$  之適應值。

表 4-11  $PSO_{pb}$  適應值函式

---

**Procedure**  $fitness(x_i)$

---

1. Get the communication settings according to the particle  $x_i$  and calculate the Shannon Capacity  $R_{ut}$ , where  $\forall u \in N_{uav}, \forall t \in T$
  2.  $MinR_t = \min(R_{ut})$
  3. **return**  $MinR_t$
- 

表 4-12 的函式主要目的為更新每次迭代之粒子位置及速度，如列 2。接著，如初始函式一樣，為了符合通訊協定規則，我們會經由列 5 到列 8 對頻寬比進行調整，最後再藉由列 9 到 12 找出新一輪的  $pb$  和  $gb$ 。

表 4-12  $PSO_{pb}$  更新函式

---

**Procedure**  $update()$

---

1. **for**  $i = 1$  to  $ParN_{pb}$  **do**
  2.     Calculate (4.1) and (4.2) to get the  $v_i$  and  $x_i$
  3.     **for**  $j = 1$  to  $Dim_{pb}$  **do**
  4.         Adjust  $x_{i,j}$  and  $v_{i,j}$  when they exceed the upper and lower bound
  5.         **for**  $k = 1$  to  $|N_{uav}|$  **do**
  6.             **for**  $u = 1$  to  $|N_{user}|$  **do**
  7.                 if the total bandwidth transmitted by UAV > upper bound **then**
  8.                      $x_{i,N_{uav}+u-1} = x_{i,N_{uav}+u-1} / TransmitB$
  9.             if  $fitness(x_i) > fitness(pb_i)$  **then**
  10.                  $pb_i = x_i$
  11.             if  $fitness(x_i) > fitness(gb)$  **then**
  12.                  $gb = x_i$
- 

表 4-13 目的在未達最大迭代次數  $Maxiter_{pb}$  前，不斷執行粒子的更新、計算適應值和找  $pb$  和  $gb$  的循環。

表 4-13  $PSO_{pb}$  粒子群演算法函式

---

**Procedure**  $pso2()$

---

1. **for**  $gen = 1$  to  $Maxiter_{pb}$  **do**
  2.      $update()$
  3.     if  $fitness(gb) > fitness(Best_{final})$
  4.          $Best_{final} = gb$
  5.          $optValue = fitness(Best_{final})$
-

## 第五章 數值分析

本章將對於第三章提出之數學模式與第四章提出之演算法進行測試與分析。本研究的測試環境為 Windows10 作業系統，Intel Core i9-10900， 2.8GHz 處理器，與 16GB 記憶體，以 Python 程式語言撰寫數學模型與求解演算法，並使用 Gurobi9.1 版求解器，以下將分別介紹如何藉由敏感度分析找出測試資料的參數設定及不同情況下無人機的飛行軌跡及通訊設定比較。

### 5.1 粒子群演算法參數之敏感度分析

本研究在執行粒子群演算法時，需決定內外層 PSO 要執行的迭代次數及粒子個數，如表 5-1。可想而知，過多的迭代次數及粒子數會造成求解時間過長；反之，當迭代次數及粒子數不夠時，則會使粒子不能進行足夠地探索及移動，導致陷入局部最佳解中。因此參數的設定值會直接的影響其程式執行的效率及所求得解的品質。

表 5-1 參數設定

參數設定			
$PSO_{traj}$ 迭代次數	$PSO_{traj}$ 粒子個數	$PSO_{pb}$ 迭代次數	$PSO_{pb}$ 粒子個數
$Maxiter_{traj}$	$ParN_{traj}$	$Maxiter_{pb}$	$ParN_{pb}$

如表 5-2 所示，本研究參考多篇以粒子群演算法收斂無人機飛行路徑之文獻，可發現學者通常把最大迭代次數定為 100 或 20000，而粒子個數則為 30 或 100。因此本研究決定先以 100、1000 和 10000 作為最大迭代次數測試之樣本，粒子個數則選用 30、40、50、60、70、80、90、100 分別做敏感度測試，欲在敏感度測試中找出最適合本研究之參數設定。

表 5-2 文獻參數比較

	參數設定	
	Max iteration	Particle
Fan & Akhter. (2021)	100	100
Song et al. (2021)	20000	30
Guo et al. (2020)	100	30
Tang et al. (2021)	100	100
本研究	100,1000,10000	30,40,50,60,70,80,90,100

由於本研究之雙層粒子群演算法是由外層  $PSO_{traj}$  包內層  $PSO_{pb}$  所組成，因此本研究會先單獨測試內層  $PSO_{pb}$  的最大迭代數和粒子個數等參數，得到最適合的參數值後，便固定內層  $PSO_{pb}$  的參數，繼續做雙層粒子群演算法的敏感度分析，進而來找到對應於外層  $PSO_{traj}$  的最佳參數（最大迭代次數和粒子個數）。

在做  $PSO_{pb}$  的敏感度分析時，本研究會固定一組由滾動式方法所求得之飛行軌跡，基於此飛行軌跡進行  $PSO_{pb}$  演算法，來求解每一單位時刻每台無人機所發射之功率及每位用戶所分配之頻寬比，期望找出每一單位時刻最小瞬時通訊量。

實驗過程我們會觀察兩種情況：第一種情況中，我們會效法大部分粒子群演算法之作法，於初始化時在一個解空間中隨機撒入  $ParN_{pb}$  個粒子，然而當所有粒子都是由隨機機制來產生位置時，很容易因位置設置不佳，而造成只找到局部最佳解或求解過久的問題。在第二種情況中，我們利用滾動式方法所求得之通訊設定值當成  $PSO_{pb}$  其中一個粒子之位置解（類似聯賽中的「種子隊」角色），其餘  $ParN_{pb} - 1$  個粒子仍保持由隨機機制來產生初始解位置，藉由此方式則可以讓粒子群演算法更有效率的找到好品質的解。

本研究以兩台無人機於  $|T|=5$ 、 $6 \times 6$  的網路中為四位用戶服務當作範例，綜合兩種情況（有無種子球員）、三種固定之隨機數種子(Random Seed=1,2,3)來為 3 種迭代

數、8 種不同粒子個數進行求解時間與其最適值之比較、分析。

而固定隨機數種子之目的是為使兩種情況得以在相同環境基礎上進行測試，不會因為不同的隨機值影響測試結果。

在最適值的部分，由於本研究之目標為最大化最小瞬時通訊量總和，因此針對每一組測試組合中，我們會基於每時刻做兩種情況和三個隨機數種子的平均，之後加總每時刻之最小瞬時通訊量進行最適值的比較。

表 5-3 為不同迭代次數  $Maxiter_{pb}$  及粒子個數  $ParN_{pb}$  組合的敏感度分析，為更清楚其趨勢，本研究將其轉換為圖 5-1 之趨勢圖。由表 5-3 及圖 5-1 相互對照後，可看見在適應值部分， $Maxiter_{pb} = 100$ 、 $ParN_{pb} = 90$  及  $Maxiter_{pb} = 1000$ 、 $ParN_{pb} = 80$  這兩個組合是所有測試組合中最高者，都為 2.68。而在求解時間的比較中，明顯可以發現迭代次數多寡與求解時間有密切的關係，因此我們採用求解時間相對較少之組合做為內層  $PSO_{pb}$  之參數設定，即  $Maxiter_{pb} = 100$ 、 $ParN_{pb} = 90$ 。

表 5-3  $PSO_{pb}$  參數敏感度分析

$Maxiter_{pb} = 100$								
	$ParN_{pb} = 30$	$ParN_{pb} = 40$	$ParN_{pb} = 50$	$ParN_{pb} = 60$	$ParN_{pb} = 70$	$ParN_{pb} = 80$	$ParN_{pb} = 90$	$ParN_{pb} = 100$
<b>Fitness</b>	2.59	2.54	2.52	2.56	2.58	2.63	<b>2.68</b>	2.56
<b>Time(s)</b>	0.15	0.17	0.17	0.18	0.20	0.21	<b>0.23</b>	0.24
$Maxiter_{pb} = 1000$								
	$ParN_{pb} = 30$	$ParN_{pb} = 40$	$ParN_{pb} = 50$	$ParN_{pb} = 60$	$ParN_{pb} = 70$	$ParN_{pb} = 80$	$ParN_{pb} = 90$	$ParN_{pb} = 100$
<b>Fitness</b>	2.58	2.65	2.63	2.63	2.62	<b>2.68</b>	2.67	2.59
<b>Time</b>	0.83	0.94	1.07	1.19	1.32	<b>1.42</b>	1.55	1.68
$Maxiter_{pb} = 10000$								
	$ParN_{pb} = 30$	$ParN_{pb} = 40$	$ParN_{pb} = 50$	$ParN_{pb} = 60$	$ParN_{pb} = 70$	$ParN_{pb} = 80$	$ParN_{pb} = 90$	$ParN_{pb} = 100$
<b>Fitness</b>	2.62	2.65	2.62	2.63	2.60	2.59	2.67	2.65
<b>Time(s)</b>	7.62	8.81	10.04	11.31	12.60	13.79	14.92	16.24

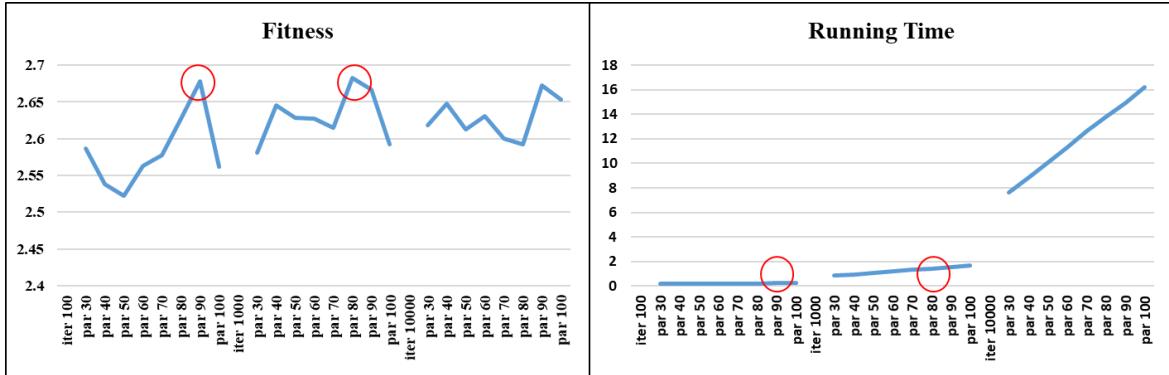


圖 5-1  $PSO_{pb}$  參數敏感度分析趨勢圖

接著，固定內層  $PSO_{pb}$  的參數為  $Maxiter_{pb} = 100$ 、 $ParN_{pb} = 90$  後，進而採用雙階層 PSO 的架構找尋對應於外層  $PSO_{traj}$  之參數  $Maxiter_{traj}$  及  $ParN_{traj}$ 。在內層 PSO 的敏感度分析中，我們選用最大迭代次數為 100,1000,10000 和粒子個數分別為 30,40,50,60,70,80,90,100 作為實驗對象進行測試，來找尋最適合本研究內層粒子群演算法之參數組合。然而在執行雙階層粒子群演算時，每一次迭代裡，外層  $PSO_{traj}$  的每一個粒子都會呼叫內層  $PSO_{pb} |T|$  次，而  $PSO_{pb}$  又會自行進行 100 次的迭代來收斂其通訊設定，因此求解時間會超出本研究可接受之範圍。為了更有效率的執行敏感度分析，本研究先觀察當外層  $PSO_{traj}$  參數設定為  $Maxiter_{traj} = 1000$  及  $ParN_{traj} = 30$  時所花費之求解時間，經實驗發現其求解時間會高達 31502 秒，因此本研究欲將測試參數縮小範圍來節省實驗時間，最終我們採最大迭代次數為 100,1000 和粒子個數分別為 30,60,90 為範例來測試比較。

表 5-4 為  $PSO_{traj}$  在不同迭代次數  $Maxiter_{traj}$  及粒子個數  $ParN_{traj}$  組合的敏感度分析，為更清楚其趨勢，本研究同樣將其轉換為圖 5-2 的趨勢圖。由表 5-4 及圖 5-2 相互對照後，可以看見不管是適應值或是求解時間的比較裡， $Maxiter_{traj} = 100$ 、 $ParN_{traj} = 90$  對於外層  $PSO_{traj}$  是最佳組合，因此我們採用此組合當作本研究外層  $PSO_{traj}$  的參數設定值。

表 5-4  $PSO_{traj}$  參數敏感度分析

	$Maxiter_{traj} = 100$		
	$ParN_{traj} = 30$	$ParN_{traj} = 60$	$ParN_{traj} = 90$
<b>Fitness</b>	3.05	3.05	<b>3.17</b>
<b>Time(s)</b>	3454.41	6301.38	<b>8050.79</b>
	$Maxiter_{traj} = 1000$		
	$ParN_{traj} = 30$	$ParN_{traj} = 60$	$ParN_{traj} = 90$
<b>Fitness</b>	3.07	3.06	3.14
<b>Time(s)</b>	31502.53	57012.72	85524.32

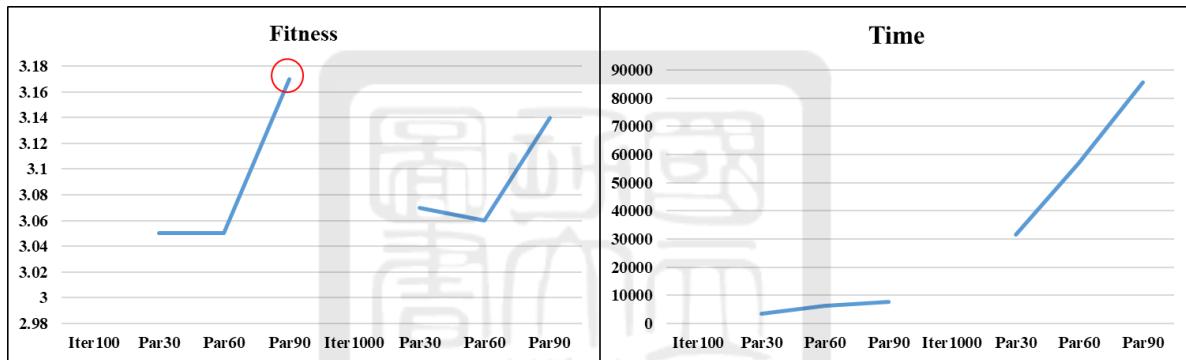


圖 5-2  $PSO_{traj}$  參數敏感度分析趨勢圖

## 5.2 問題設定之敏感度分析

本研究在問題設定上考慮到無人機機群之間的同頻干擾影響，此干擾訊號會直接的降低用戶接收之通訊量大小，因此本小節將針對用戶和無人機最佳放置位置及派遣之無人機數進行敏感度分析，找出最佳的問題設定。

首先，本研究將針對兩位用戶的距離做敏感度分析，找出相對於距離的最佳派遣無人機數量。由於兩位用戶最多只會派遣兩台無人機來服務用戶，因此會觀察不同距離下，一台和兩台無人機服務用戶之最佳值。如表 5-5 所示，在一台無人機服務兩位用戶的情境下，當距離拉大，通訊量會因為無人機離用戶越來越遠而隨之降低。當有兩台無人機同時服務兩位用戶時，用戶則會受到非主要無人機之干擾訊號影響其通訊

量大小，當用戶間距離越拉越大，用戶所接收之干擾訊號則會越來越少，使通訊量逐漸增加。經測試可得兩用戶在相距 14470 公尺時，為一台和兩台無人機之平衡距離。亦即，當用戶間距離大於 14470 公尺時，派遣兩台無人機會優於一台無人機；而當用戶間距離小於 14470 公尺時，則會選擇派遣一台無人機來得到更高的通訊量。

表 5-5 用戶距離及無人機數比較

	距離 (m)	OBJ	功率	頻寬		距離 (m)	OBJ	功率	頻寬
1 UAV	10000	7.40	[0.5]	[0.62, 0.38]	2 UAV	10000	6.66	[0.5, 0.5]	[1, 1]
	13000	7.25	[0.5]	[0.63, 0.37]		13000	7.03	[0.5, 0.5]	[1, 1]
	14000	7.21	[0.5]	[0.63, 0.37]		14000	7.14	[0.5, 0.5]	[1, 1]
	14470	7.19	[0.5]	[0.63, 0.37]		14470	7.19	[0.5, 0.5]	[1, 1]
	15000	7.17	[0.5]	[0.63, 0.37]		15000	7.24	[0.5, 0.5]	[1, 1]
	20000	7.00	[0.5]	[0.64, 0.36]		20000	7.65	[0.5, 0.5]	[1, 1]

得知距離和最佳派遣無人機數量的關係後，本研究將會利用雙層粒子群演算法（此處演算法之外層粒子解為最佳位置，而非如第四章所述的飛行軌跡），收斂出不同無人機數在服務兩用戶時的最佳位置。

舉例來說，當一台無人機服務兩位相距  $10000 < 14470$  公尺之用戶時（如圖 5-3 所示），無人機最佳的服務位置會懸停於某一用戶上方，再藉由分配較多的頻寬比給偏離的用戶，以提高用戶通訊量。

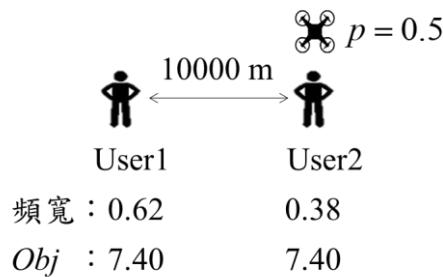


圖 5-3 一台無人機兩用戶最佳服務位置

在兩用戶相距 14470 公尺以上時，派遣兩台無人機服務兩位用戶相較於一台無人

機更優，如圖 5-4，兩台無人機最佳的服務位置會分別懸停於一用戶上方，各自服務主要服務之用戶，以最大化瞬時通訊量。

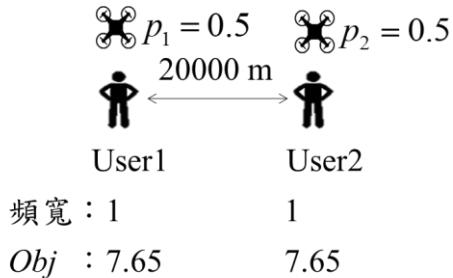


圖 5-4 兩台無人機兩用戶最佳服務位置

綜合以上，可得知在派遣無人機時，如果用戶間距離足夠遠，即干擾訊號不會過度影響主要訊號值，則會盡其所能的出動  $|N_{user}|$  台無人機，且無人機最佳的服務位置會懸停於用戶上方，來服務主要服務之用戶，以最大化瞬時通訊量。

### 5.3 用 戶 分 群 之 求 解 結 果 比 較

本小節以 5.2 節所求之用戶距離，將用戶分群成不同的情境來和不同求解方法之最小瞬時通訊量總和進行測試與比較。本研究以兩台無人機於  $|T|=10$ 、 $6 \times 6$  的網路中為 4 位用戶服務當作範例，本研究將每一單位距離視為 5000 公尺，以五種不同分佈位置來比較其求解結果，此五個情境分別為：

1. 情境 1：如圖 5-5 的用戶分配位置，主要是將四位用戶分成四群，即每一位用戶都自成一群，分別分布在服務範圍內的四個角落。
2. 情境 2：如圖 5-6 的用戶分配位置，將四位用戶分成三群，分別為用戶 1、3、4 聚集成一群，而用戶 2 單獨一群。
3. 情境 3：如圖 5-7 的用戶分配位置，將四位用戶分成兩群，分別為用戶 1、3、4 集結為一群，而用戶 2 自己單獨一群。

4. 情境 4：如圖 5-8 的用戶分配位置，同樣是將四位用戶分成兩大族群，而分配方式為用戶 1、4 一群，而用戶 2、3 為另一群。

5. 情境 5：如圖 5-9 的用戶分配位置，將四位用戶集中成一大群。

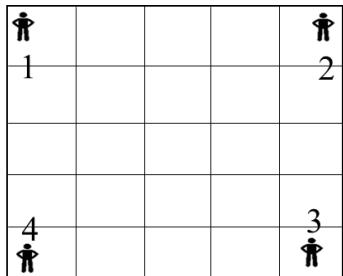


圖 5-5 情境 1

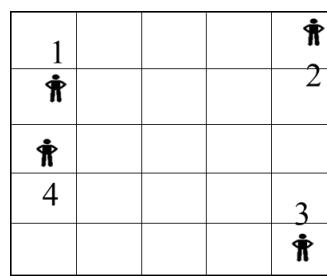


圖 5-6 情境 2

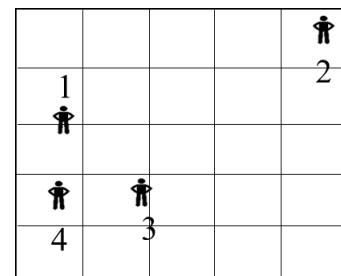


圖 5-7 情境 3

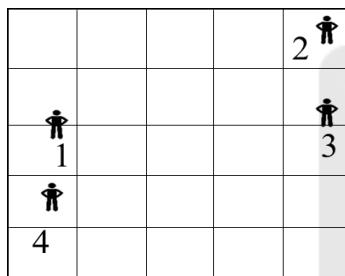


圖 5-8 情境 4

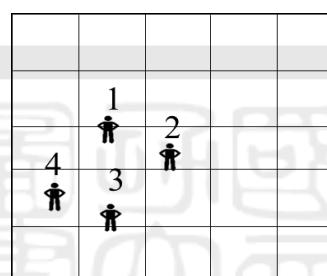


圖 5-9 情境 5

針對此五種情境，我們使用表 5-6 中四個方法來測試與分析，並比較他們之間的差異。

表 5-6 求解方法與說明

方法	說明
<b>RM</b>	單純以第三章之數學模型搭配 4.1 節所提出之滾動式方法加速求解。
<b>PSO<sub>pb</sub></b>	以 RM 求解出的飛行軌跡為基礎，用 4.2.4 小節所開發之內層粒子群演算法 PSO <sub>pb</sub> 求解對應位置上的通訊設定。
<b>PSO<sub>1</sub></b>	為 4.2 小節所開發之雙層粒子群演算法，演算法中所有粒子皆為隨機生成。
<b>PSO<sub>2</sub></b>	為 4.2 小節所開發之雙層粒子群演算法，內外層皆有一個粒子為參考 5.2 節實驗所得之貪婪解當作種子球員，其餘粒子則同樣由隨機產生器隨機生成。

由於在建立數學模型  $RM$  時，為了適用於 Gurobi，我們將夏農公式轉換成一個估計的線性公式，因此當  $RM$  求解出一組飛行軌跡及通訊設定時，本研究會將其值代入原始公式進行精算，並與其它方式比較。而粒子群演算法的測試中為了使不同方式在相同環境基礎上進行測試，不會因為不同的隨機數影響測試結果，我們會固定三種隨機數種子(Random Seed=1,2,3)，並在飛行過程中每一單位時刻做三個隨機數種子的平均，之後進行加總得最小瞬時通訊量總和的比較。

表 5-7 粒子群演算法與數學模式之最佳值比較

	<i>Obj</i>			
	$RM$	$PSO_{pb}$	$PSO_1$	$PSO_2$
情境 1	5.25	6.95	7.52	<b>9.18</b>
情境 2	5.13	7.17	8.21	<b>10.55</b>
情境 3	4.87	10.16	14.42	<b>15.35</b>
情境 4	5.20	11.36	15.10	<b>16.76</b>
情境 5	5.18	27.73	29.50	<b>30.99</b>

觀察表 5-7 可得知數學模式  $RM$  目標值遠低於其它求解方法，猜測是因為計算通訊量的公式在線性轉換的過程中低估了干擾訊號的影響，即使數學模式目標值不是很好但數學模式的解卻可以當作種子球員，成為  $PSO_2$  裡其中一個粒子的初始位置，使  $PSO_2$  相較於其它方法能收斂到更好的解。

而細看情境 1 到 5，可發現用戶間的距離會由一開始分佈在四個角落慢慢的拉近，直到情境 5 集結成一大群，我們可以從表 5-7 看見當用戶彼此距離靠近，則會因為  $RM$  的線性公式與原始公式計算通訊量的部分有差異，使其低估了干擾的影響進而導致目標式最佳值低於演算法。但當用戶間距離越遠， $RM$  和粒子群演算法  $PSO_{pb}$ 、 $PSO_1$  及  $PSO_2$  的目標值差異就會越少，因此針對用戶獨立分散分布的情況， $RM$  或粒子群演算法  $PSO_{pb}$ 、 $PSO_1$  及  $PSO_2$  都可以應用在山區來搭建臨時性蜂巢網路。

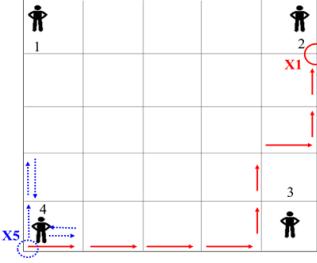
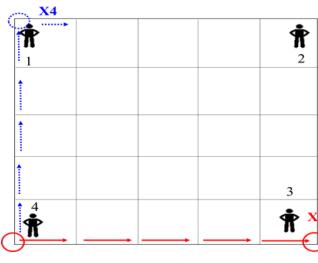
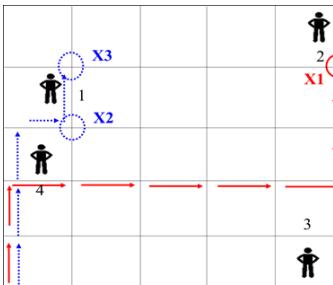
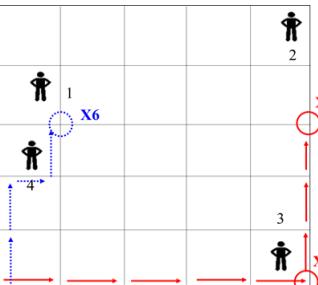
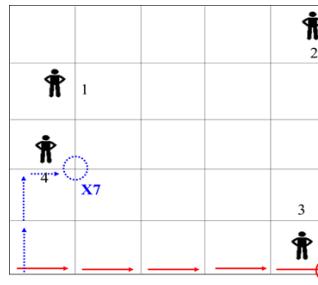
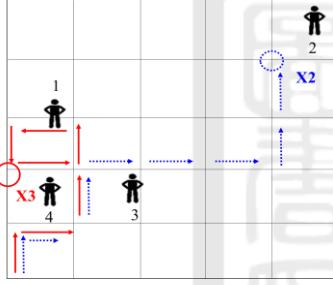
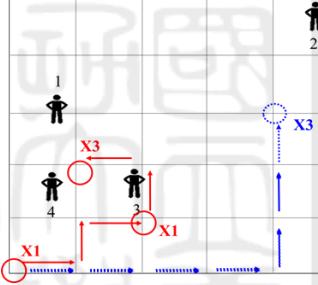
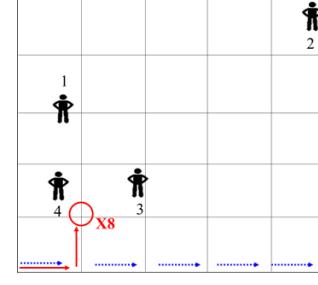
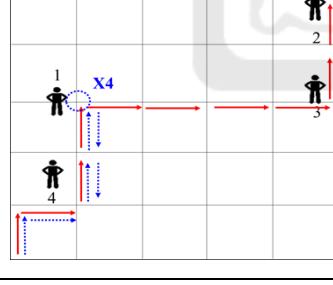
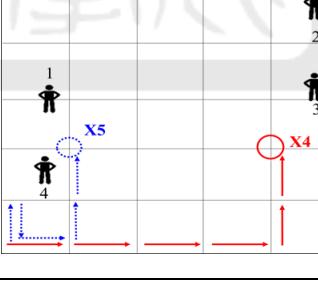
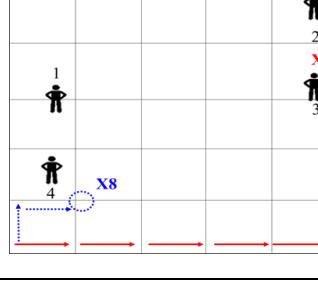
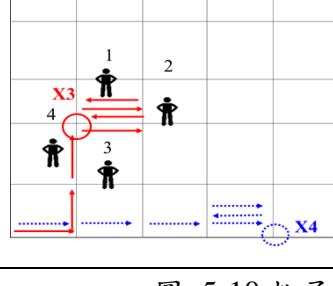
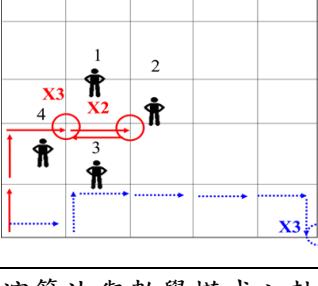
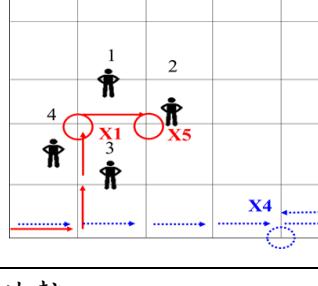
	$RM + PSO_{pb}$	$PSO_1$	$PSO_2$
情境 1			
情境 2			
情境 3			
情境 4			
情境 5			

圖 5-10 粒子群演算法與數學模式之軌跡比較

由圖 5-10 可觀察五種情境之兩台無人機飛行軌跡。其中，小黑人代表著用戶的分佈位置，不同型態的箭頭代表不同無人機在每時刻的飛行方向。而無人機於某格點懸停則以圓環型箭頭代表，若停留三單位時間則在該格點旁邊以 X3 表示。

在情境 1 到情境 4 的飛行路線圖中，用戶群體數  $G_{user}$  皆大於無人機數  $|N_{uav}|$ ，故派出兩台無人機分別服務並靠近  $\lfloor G_{user} / |N_{uav}| \rfloor$  和  $G_{user} - \lfloor G_{user} / |N_{uav}| \rfloor$  群用戶。當四位用戶因彼此距離較近而被視為同一群用戶時（情境五），則所有解法都只會指派其中一台無人機來靠近四位用戶，而另一台無人機則為了降低干擾盡可能地遠離用戶。

#### 5.4 不同用戶數及無人機數之求解結果比較

5.3 節中，本研究透過距離將用戶進行分群並比較目標值大小和飛行軌跡，可觀察發現 RM 因為計算通訊量的公式在線性轉換的過程中嚴重低估了干擾訊號的影響，而使目標值遠低於其它求解方法，並不具比較的價值，因此本小節會拿掉 RM 並效法 Chou et al. (2020) 的作法進行求解比較，如下表，在每一單位距離視為 10000 公尺、服務時長  $|T|$  為 10 的基礎上，以表 5-9 之不同測資觀察不同無人機數在不同用戶數的狀態下目標值和飛行軌跡的差異。

表 5-8 求解方法與說明

求解方法	
$PSO_{pb}$	以 RM 求解出的飛行軌跡為基礎，用第四章 4.2.4 小節所開發之內層粒子群演算法 $PSO_{pb}$ 求解對應位置上的通訊設定。
$PSO_1$	為第四章 4.2 小節所開發之雙層粒子群演算法，演算法中所有粒子皆由隨機產生器隨機生成。
$PSO_2$	為第四章 4.2 小節所開發之雙層粒子群演算法，內外層皆有一個粒子為參考 5.2 節實驗所得之貪婪解當作種子球員，其餘粒子則由隨機產生器生成。
$PSO_3$	效法 Chou et al. (2020) 的作法。將發射功率設定為參數，而每一需求點的頻寬為無人機總頻寬之平均。用外層粒子群演算法求解其對應軌跡。

表 5-9 測試資料參數設定

名稱	$ N_{node} $	$ N_{uav} $	$ N_{user} $
Case 1	36	1	1
Case 2	36	1	2
Case 3	36	1	3
Case 4	36	2	1
Case 5	36	2	2
Case 6	36	2	3
Case 7	36	3	1
Case 8	36	3	2
Case 9	36	3	3

經測試可得如表 5-10 之最佳值比較，以及圖 5-11、圖 5-12、圖 5-13 的飛行路線圖。

表 5-10 不同狀態及求解方法之最佳值比較

	<i>Obj</i>			
	<i>PSO<sub>pb</sub></i>	<i>PSO<sub>1</sub></i>	<i>PSO<sub>2</sub></i>	<i>PSO<sub>3</sub></i>
Case 1	<b>173.2</b>	172.5	<b>173.2</b>	113.8
Case 2	59.1	53.8	<b>62.8</b>	54.5
Case 3	57.5	52.0	<b>61.6</b>	39.1
Case 4	<b>172.8</b>	171.0	<b>172.7</b>	111.4
Case 5	69.2	62.9	<b>83.5</b>	54.3
Case 6	59.8	46.7	<b>64.2</b>	35.1
Case 7	170.2	169.3	<b>171.9</b>	109.3
Case 8	67.5	61.3	<b>82.7</b>	56.6
Case 9	61.4	45.3	<b>66.7</b>	27.4

觀察表 5-10 可發現  $PSO_2$  之目標值通常會是最優而  $PSO_3$  則會是所有求解方法中目標值最小者，原因將由圖 5-11、圖 5-12、圖 5-13 個別飛行路線圖詳細說明。

在不同 Case 中，當無人機數相同時(如 Case 1、2、3)，如果用戶人數增加，最佳值則會隨之遞減。而用戶人數相同時，若無人機數一增加，則會因為無人機數及用戶人數之間的關係產生兩種情況：如果無人機數多於用戶人數(如 Case 1、4、7)，無人機數增加時，目標值則會因為干擾問題而跟著遞減；但如果無人機數少於用戶人數(如 Case 3、6、9)，無人機數增加時，目標值則會因為用戶因收到更多頻寬而跟著遞增。

圖 5-11、圖 5-12、圖 5-13 中的符號與前述圖 5-10 相同。以圖 5-11 為例，當無人機數為一台時，在只有一個用戶的 case 1 中，四個求解方式皆會飛向離用戶最近的格點進行服務；而 case 2 和 case 3 中，無人機數多於用戶人數時， $PSO_{pb}$  和  $PSO_2$  都會飛向某一用戶上方進行服務，譬如 5.2 節所找到的最佳位置。而  $PSO_1$  則會飛往用戶間的中點； $PSO_3$  則比較沒有規律，因此才會如表 5-10 所示， $PSO_{pb}$  和  $PSO_2$  的目標值會比  $PSO_1$  更好，而  $PSO_3$  則是最小的。

如圖 5-12，在無人機數為兩台時，當無人機數多於用戶人數時（譬如 case 4），則會只派遣  $|N_{user}|$  台無人機去服務並靠近  $|N_{user}|$  位用戶，其餘  $|N_{uav}| - |N_{user}|$  台無人機則屬於閒置狀態。當無人機數等於用戶人數時（譬如 case 5），則會派遣所有的無人機去服務所有人，而無人機位置如 5.2 節所測試之結果顯示，直接懸停在用戶上方之最佳值較好；因此  $PSO_{pb}$  和  $PSO_2$  的飛行路線將會比  $PSO_1$  和  $PSO_3$  好。反之，當無人機數少於用戶人數時（譬如 case 6），會指派所有無人機去服務用戶，飛行路線同樣是如  $PSO_{pb}$  和  $PSO_2$  一樣懸停於用戶之上較佳。

	<b>Case1</b>	<b>Case2</b>	<b>Case3</b>
$PSO_{pb}$			
$PSO_1$			
$PSO_2$			
$PSO_3$			

圖 5-11 一台無人機之飛行軌跡

	Case4	Case5	Case6
$PSO_{pb}$			
$PSO_1$			
$PSO_2$			
$PSO_3$			

圖 5-12 兩台無人機之飛行軌跡

	Case7	Case8	Case9
$PSO_{pb}$	<p>Flight path diagram for Case 7 using PSO<sub>pb</sub>. It shows three drones (X3, X5, X8) moving from user locations (1, 3, 5) to a central point (15).</p>	<p>Flight path diagram for Case 8 using PSO<sub>pb</sub>. It shows three drones (X5, X2, X1) moving from user locations (1, 3, 5) to a central point (15).</p>	<p>Flight path diagram for Case 9 using PSO<sub>pb</sub>. It shows three drones (X5, X3, X5) moving from user locations (1, 3, 5) to a central point (15).</p>
$PSO_1$	<p>Flight path diagram for Case 7 using PSO<sub>1</sub>. It shows three drones (X6, X3, X4) moving from user locations (1, 3, 5) to a central point (15).</p>	<p>Flight path diagram for Case 8 using PSO<sub>1</sub>. It shows three drones (X1, X4, X2) moving from user locations (1, 3, 5) to a central point (15).</p>	<p>Flight path diagram for Case 9 using PSO<sub>1</sub>. It shows three drones (X3, X1, X6) moving from user locations (1, 3, 5) to a central point (15).</p>
$PSO_2$	<p>Flight path diagram for Case 7 using PSO<sub>2</sub>. It shows three drones (X6, X6, X1) moving from user locations (1, 3, 5) to a central point (15).</p>	<p>Flight path diagram for Case 8 using PSO<sub>2</sub>. It shows three drones (X7, X3, X2) moving from user locations (1, 3, 5) to a central point (15).</p>	<p>Flight path diagram for Case 9 using PSO<sub>2</sub>. It shows three drones (X5, X6, X5) moving from user locations (1, 3, 5) to a central point (15).</p>
$PSO_3$	<p>Flight path diagram for Case 7 using PSO<sub>3</sub>. It shows three drones (X5, X6, X1) moving from user locations (1, 3, 5) to a central point (15).</p>	<p>Flight path diagram for Case 8 using PSO<sub>3</sub>. It shows three drones (X8, X2, X5) moving from user locations (1, 3, 5) to a central point (15).</p>	<p>Flight path diagram for Case 9 using PSO<sub>3</sub>. It shows three drones (X6, X6, X5) moving from user locations (1, 3, 5) to a central point (15).</p>

圖 5-13 三台無人機之飛行軌跡

如圖 5-13，在無人機數為三台時，若無人機數少於用戶人數（譬如 case 7）、8，

則會只派遣  $|N_{user}|$  台無人機去服務並靠近  $|N_{user}|$  位用戶，其餘  $|N_{uav}|-|N_{user}|$  台無人機則屬於閒置狀態。反之，當無人機數等於用戶人數時（譬如 case 9），則會派遣所有的無人機去服務所有人。四個 case 之最佳飛行路線同樣是如  $PSO_{pb}$  和  $PSO_2$  一樣懸停於用戶之上。由表 5-10 可發現以貪婪解當作種子球員的  $PSO_2$  目標值最佳，為了進一步探討貪婪法所得飛行軌跡之效能，我們在單位距離 10000 公尺、服務時長  $T$  為 10 的基礎上，以表 5-9 之測資多做了下表的  $PSO_{2-pb}$  進行測試及比較。

表 5-11 求解方法與說明

求解方法	說明
$PSO_2$	為第四章 4.2 小節所開發之雙層粒子群演算法，內外層皆有一個粒子為參考 5.2 節實驗所得之貪婪解當作種子球員，其餘粒子則同樣由隨機產生。
$PSO_{2-pb}$	以貪婪法得出無人機飛行軌跡，固定其軌跡並使用內層粒子群演算法求解其對應通訊設定。

表 5-12 9 組測資之最佳值比較

	<i>Obj</i>	
	$PSO_2$	$PSO_{2-pb}$
<b>Case 1</b>	173.2	173.2
<b>Case 2</b>	62.8	62.8
<b>Case 3</b>	61.6	61.4
<b>Case 4</b>	172.7	173.6
<b>Case 5</b>	83.5	83.5
<b>Case 6</b>	64.2	64.1
<b>Case 7</b>	171.9	173.0
<b>Case 8</b>	82.7	83.6
<b>Case 9</b>	66.7	66.9

觀察表 5-12 及圖 5-14，可發現  $PSO_2$  及  $PSO_{2-pb}$  之目標值和無人機飛行軌跡都差不多，而相較於還要多執行一次外層粒子群演算法來收斂飛行軌跡的  $PSO_2$ ， $PSO_{2-pb}$  可以在更短的求解時間裡求得一樣好的最佳解。

	$PSO_2$	$PSO_{2-pb}$
Case 1		
Case 2		
Case 3		

<b>Case 4</b>		
<b>Case 5</b>		
<b>Case 6</b>		
<b>Case 7</b>		

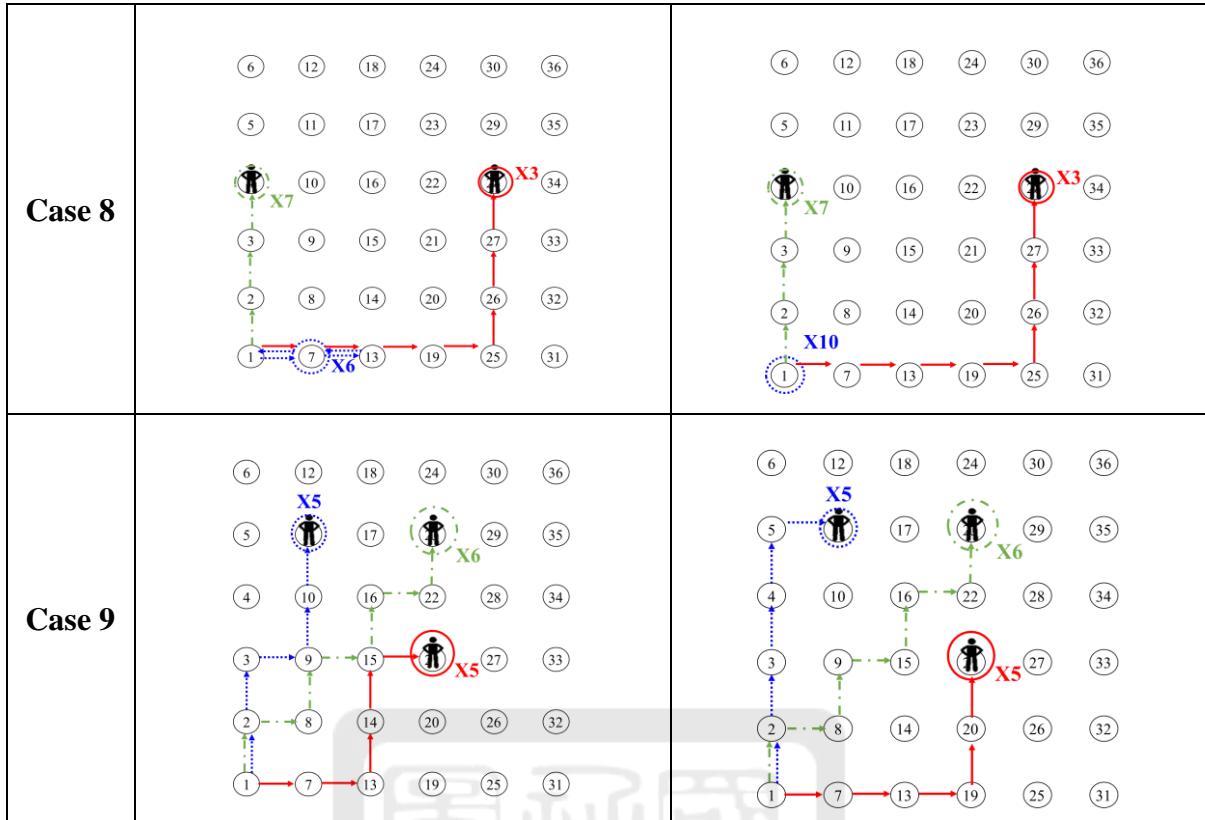


圖 5-14 9 組測資之飛行軌跡

## 5.5小結

由於粒子群演算法有關於迭代次數和粒子數等參數會直接影響演算法求解之效率以及品質，因此本章節先進行相關參數之敏感度分析，找到最適參數組合，使粒子群演算法在求解過程中可以在求解品質和執行時間中取得平衡。由於計算通訊量的公式中，距離會直接影響訊號及干擾的大小，因此本研究同樣為用戶間的距離進行敏感度分析，找到用戶間隔距離所對應無人機派遣之最佳數量及懸停位置。接著再進行演算法及數學模式之比較，可以發現數學模式和  $PSO_2$  的無人機都會收斂至敏感度分析所測出的最佳懸停位置； $PSO_1$  則飛向用戶中點來進行服務。而  $PSO_3$  效法文獻固定發射功率、平均分配頻寬的作法則沒有固定的規律，其最佳值也是最差的。反之，有種子球員協助的  $PSO_2$  則能得到最好的最佳目標值。

最後經測試  $PSO_2$  及  $PSO_{2pb}$  後，可發現其實以貪婪法所得之無人機飛行路線配合

內層粒子群演算法的  $PSO_{2pb}$ ，相較於有種子球員協助的  $PSO_2$ ，更能在短時間內求得相同品質的最佳解。



## 第六章 結論與未來研究建議

針對本研究之多台無人機權充當行動基地台來提供無線網路問題中，我們希望在基地台因像地震等因素而崩塌，無法繼續提供用戶網路時，無人機便能當作移動基地台快速建立臨時蜂巢網路。本研究針對此議題分別於第三章及第四章中使用數學模型與演算法來進行求解，並在第五章中進行數值測試與比較。因此我們將於本章節做最後的結論並探討外來之研究方向。

### 6.1 結論與貢獻

由於無人機本身高機動性與便利性的特質，促使無人機成為各行各業的新寵兒，全球都卯足了全力往無人機普及的時代努力。近幾年，無線網路對於每一個人的生活漸漸地成為不可或缺的必需品，無人機在無線網路之領域發展也更受關注。尤其在災難發生之際，無線網路不僅可以幫助搜救人員加速救援更能使受難者得以和外界聯繫並求救，而這時無人機便是形成行動基地台、迅速架構臨時網路的關鍵。然而分散的用戶，常常造成無人機在提供網路時效率非常低，故本研究期望能提出如何讓多台無人機合作找到最佳的飛行路線及通訊設定來服務用戶，並提高每一時刻所有用戶間最小的瞬時通訊量，以保障每位用戶接收之公平性。

為能更了解無人機的基本知識與其在無線網路領域上之應用，本研究參考多篇文獻，並發現大部分文獻並未考慮多台無人機在提供無線網路時所造成之干擾。雖然多數文獻以塊座標下降演算法(Block Coordinate Descent)求解，但本研究認為該方法有因果邏輯錯置的疑慮，因此想以不同角度出發來設計求解方法。

本研究在第三章開發了一個線性的混合整數數學規劃模式，試圖將非線性的通訊量計算公式進行線性簡化、轉換，並以 Gurobi 進行求解。接著，為了處理規模較大的問題，本研究嘗試在第四章中撰寫滾動式演算法來分組逐批求解小規模的混整數規劃模式。本研究亦參考塊座標下降演算法的想法，設計雙階層粒子群演算法來收斂、求解無人機的軌跡與其通訊設定。其中，外階層之粒子群演算法旨在收斂總服務時長內的飛行軌跡，而針對給定的一組飛行軌跡則再會給內階層之粒子群演算法去收斂每

一時刻每一台無人機的最佳通訊設定，以計算各使用者之最大瞬時通訊量，同一時刻各使用者中最小的瞬時通訊量再經由外層粒子群演算法加總，以達到最大化最小瞬時通訊量總和之目標。

最後，本研究在第五章中針對雙層粒子群演算法進行敏感度分析，找到最適合此演算法的粒子數及最大迭代次數設定。我們也對第三章的數學模式以及第四章演算法進行數值分析，比較了飛行軌跡及最小順時通訊總量，從中發現第三章所建構的數學模式低估了干擾的影響，但針對當無人機服務於空曠且用戶分散的山區仍能適用（與粒子群演算法求解結果類似）。而在本研究所提出的三種求解方式中，以貪婪法所得之無人機飛行路線配合內層粒子群演算法的效果最佳。

綜合上述，針對本研究的具體貢獻進行以下整理說明：

#### **1. 考慮多台無人機同頻干擾和最大化最小資料傳輸量：**

本研究在設定問題時為更符合現實狀況，將多台無人機間同頻干擾的影響納入計算用戶所接收到瞬時通訊量之公式之中，而我們目的在於盡其所能地提高每一時刻之最小資料傳輸量，以保障通訊網路之品質和用戶間的公平性。

#### **2. 時空網路圖建構數學模型：**

如第二章所述，大部分文獻中都採用塊座標下降演算法(Block Coordinate Descent)進行求解，因此本研究以不同角度切入，成為第一篇利用時空網路圖建構數學模型進行模擬、求解之論文。

#### **3. 滾動式求解加速數學模型：**

由於橫跨整個規劃時長的數學規劃模式求解太過耗時，因此本研究提出滾動式求解機制，將問題拆分成數個規模較小（較短的規劃時長）的問題，以分段逐批方式每次求解部分規劃時長的問題，固定較早期的決策結果、將較晚期的決策視為下一段的初始解，如此可依時間逐次求解方式來處理規劃時程較長的問題。

#### **4. 使用粒子群演算法進行收斂：**

大部分文獻所使用之塊座標下降演算法在優化路徑、頻寬比及發射功率時，會使用迭代的方式，在每次迭代只優化一項變數，直至收斂。此作法在路徑固定情況下優化頻寬比或發射功率是合理的。但如果固定頻寬比及發射功率的前提下，再反過來收斂路徑則會因為「倒果為因」而不甚合理。即便如此，此演算法「先調整解再收斂」的機制則是符合直覺，因此本研究便依照此機制提出兩階層粒子群演算法。其架構主要由外層粒子位置調整每台無人機之飛行路線，在每一時刻無人機位置已知的情況下，收斂內層粒子最佳位置來求得各時段無人機的功率與其對各用戶的頻寬比設定，重複上述步驟直至收斂。

## 5. 敏感度分析：

本研究針對我們自己所開發之粒子群演算法，經由實驗測試來找尋適合之最大迭代數及粒子個數參數值，來平衡演算法的求解品質和求解時間。

## 6.2 未來研究方向

直至目前為止，本研究雖對於多台無人機權充行動基地台來提供無線網路之飛行路線與通訊設定問題做了一些具體的貢獻和探討，但仍尚有未臻完善之處，以下將列舉說明。

### 1. 自然環境因：

本研究目前把問題設定成在不考慮外在天氣、干擾等不可控制因素的前提下來建立模型與演算法，然而在現實狀況下這些因素常常會導致無人機在服務期間遇上困難，例如陰天的雲層會減弱用戶所接收之訊號，或是當無人機遇到強風時，無人機將會花費比平時更多飛行時間來抵達下一格點，因此，我們可多加限制使問題更符合現實狀況。

### 2. 無人機電量限制：

本研究目前假設服務時間無人機電量足以負荷飛行，現實中的無人機則會有電池電量與續航力的限制，使得無人機必須在電量耗盡前進行充換電再繼續服務，因此，

未來更貼近實務的研究可考慮將無人機的電量耗損與充換電加入整體路線規劃中。

### 3. 數學模式改進：

在現在數學模式的設計中，我們無法將線性轉換後的公式完全還原為原本公式的效用，因此會造成用戶間距離越靠近，數學模式的解就會和原本的解差距越大，未來希望可以開發出一個更接近原本夏農公式的線性轉換數學式。

### 4. 重現塊座標下降演算法(Block Coordinate Descent)：

本研究目前進行分析的方法都是我們自己開發的數學模型與模仿其它文獻而改寫的粒子群演算法，雖然我們認為塊座標下降演算法有因果邏輯上的缺失，但其實仍有待更多實驗來比較與分析。

### 5. 決策變數設定：

本研究目前對於變數的設定為每台無人機每一時刻所發射給所有需求點之功率量相同，但會針對不同需求點給予不同的頻寬比例。未來亦可效仿 Shi et al. (2010) 使每台無人機之發射功率依需求點不同給予不同的功率量，而頻寬變數則同樣可以依需求點不同給予不同的頻寬頻率範圍，當不同頻寬範圍互相重疊才計算其干擾訊號。

## 參考文獻

### 中文

聯合新聞網，楊又肇(2020)。HAPS 無人飛機 1 萬 9 千公尺高空完成超過 20 小時網路測試。上網日期：2021 年 12 月 04 日。檢自 <https://udn.com/news/story/7086/4939452>

吳珣君(2020)。防救災行動通訊再升級 無人機變身空中基地台。上網日期：2021 年 12 月 04 日。檢自 <https://www.rti.org.tw/news/view/id/2048870>

VITO 雜誌。看懂 dB、dBm、dBw 的區別。上網日期：2022 年 02 月 22 日。檢自 <https://vitomag.com/tech/cwhzy.html>

### 英文

Ahmed, S., Chowdhury, M. Z., & Jang, Y. M. (2020). Energy-efficient UAV-to-user scheduling to maximize throughput in wireless networks. *IEEE Access*, 8, 21215-21225.

Al-Hourani, A., Kandeepan, S., & Lardner, S. (2014). Optimal LAP altitude for maximum coverage. *IEEE Wireless Communications Letters*, 3(6), 569-572.

Alzenad, M., El-Keyi, A., Lagum, F., & Yanikomeroglu, H. (2017). 3-D placement of an unmanned aerial vehicle base station (UAV-BS) for energy-efficient maximal coverage. *IEEE Wireless Communications Letters*, 6(4), 434-437.

Alzenad, M., El-Keyi, A., & Yanikomeroglu, H. (2018). 3-D placement of an unmanned aerial vehicle base station for maximum coverage of users with different QoS requirements. *IEEE Wireless Communications Letters*, 7(1), 38-41.

- Amgoth, T., & Jana, P. K. (2017). Coverage hole detection and restoration algorithm for wireless sensor networks. *Peer-to-Peer Networking and Applications*, 10(1), 66-78.
- Campion, M., Ranganathan, P., & Faruque, S. (2018, May). A Review and Future Directions of UAV Swarm Communication Architectures. In *EIT* (pp. 903-908).
- Chou, S. F., Pang, A. C., & Yu, Y. J. (2019). Energy-aware 3D unmanned aerial vehicle deployment for network throughput optimization. *IEEE Transactions on Wireless Communications*, 19(1), 563-578.
- Fallgren, M. (2010, June). On the complexity of maximizing the minimum Shannon capacity in wireless networks by joint channel assignment and power allocation. In 2010 IEEE 18th International Workshop on Quality of Service (IWQoS) (pp. 1-7). IEEE.
- Fu, S., Zhao, L., Su, Z., & Jian, X. (2018). UAV based relay for wireless sensor networks in 5G systems. *Sensors*, 18(8), 2413.
- Guo, X., Ji, M., Zhao, Z., Wen, D., & Zhang, W. (2020). Global path planning and multi-objective path control for unmanned surface vehicle based on modified particle swarm optimization (PSO) algorithm. *Ocean Engineering*, 216, 107693.
- Hu, F., Ou, D., & Huang, X. L. (Eds.). (2020). *UAV Swarm Networks : Models, Protocols, and Systems*. CRC Press.
- Huang, H., & Savkin, A. V. (2021). Energy-efficient decentralized navigation of a team of solar-powered UAVs for collaborative eavesdropping on a mobile ground target in urban environments. *Ad Hoc Networks*, 117, 102485.
- Kennedy, J., & Eberhart, R. (1995, November). Particle swarm optimization.

- In Proceedings of ICNN'95-international conference on neural networks (Vol. 4, pp. 1942-1948). IEEE.
- Kim, K., Park, Y. M., & Hong, C. S. (2020, January). Machine learning based edge-assisted UAV computation offloading for data analyzing. In 2020 International Conference on Information Networking (ICOIN) (pp. 117-120). IEEE.
- Mozaffari, M., Saad, W., Bennis, M., & Debbah, M. (2016, December). Mobile Internet of Things : Can UAVs provide an energy-efficient mobile architecture?. In 2016 IEEE global communications conference (GLOBECOM) (pp. 1-6). IEEE.
- Nayeem, G. M., Fan, M., & Akhter, Y. (2021, January). A time-varying adaptive inertia weight based modified PSO algorithm for UAV path planning. In 2021 2nd International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST) (pp. 573-576). IEEE.
- Shi, W., Li, J., Cheng, N., Lyu, F., Zhang, S., Zhou, H., & Shen, X. (2019). Multi-drone 3-D trajectory planning and scheduling in drone-assisted radio access networks. *IEEE Transactions on Vehicular Technology*, 68(8), 8145-8158.
- Shi, Y., Hou, Y. T., Kompella, S., & Sherali, H. D. (2010). Maximizing capacity in multihop cognitive radio networks under the SINR model. *IEEE Transactions on Mobile Computing*, 10(7), 954-967.
- Song, B., Wang, Z., & Zou, L. (2021). An improved PSO algorithm for smooth path planning of mobile robots using continuous high-degree Bezier curve. *Applied Soft Computing*, 100, 106960.
- Sun, Y., Xu, D., Ng, D. W. K., Dai, L., & Schober, R. (2019). Optimal 3D-trajectory

- design and resource allocation for solar-powered UAV communication systems. *IEEE Transactions on Communications*, 67(6), 4281-4298.
- Tang, Y., Miao, Y., Barnawi, A., Alzahrani, B., Alotaibi, R., & Hwang, K. (2021). A joint global and local path planning optimization for UAV task scheduling towards crowd air monitoring. *Computer Networks*, 193, 107913.
- Wu, Q., & Zhang, R. (2017, December). Delay-constrained throughput maximization in UAV-enabled OFDM systems. In *2017 23rd Asia-Pacific Conference on Communications (APCC)* (pp. 1-6). IEEE.
- Wu, Q., Zeng, Y., & Zhang, R. (2018). Joint trajectory and communication design for multi-UAV enabled wireless networks. *IEEE Transactions on Wireless Communications*, 17(3), 2109-2121.
- Wu, Q., & Zhang, R. (2018). Common throughput maximization in UAV-enabled OFDMA systems with delay consideration. *IEEE Transactions on Communications*, 66(12), 6614-6627.
- Xu, Y., Xiao, L., Yang, D., Wu, Q., & Cuthbert, L. (2018). Throughput maximization in multi-UAV enabled communication systems with difference consideration. *IEEE Access*, 6, 55291-55301.
- Yin, H., & Alamouti, S. (2006, March). OFDMA : A broadband wireless access technology. In *2006 IEEE sarnoff symposium* (pp. 1-4). IEEE.
- Zeng, Y., Zhang, R., & Lim, T. J. (2016). Wireless communications with unmanned aerial vehicles : Opportunities and challenges. *IEEE Communications Magazine*, 54(5), 36-42.
- Zorbas, D., & Douligeris, C. (2018, April). Computing optimal drone positions to wirelessly recharge IoT devices. In *IEEE INFOCOM 2018-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)* (pp.

628-633). IEEE.

