

國立臺灣大學工學院土木工程學系

碩士論文

Department of Civil Engineering

College of Engineering

National Taiwan University

Master Thesis

運用基於派翠網之蟻群最佳化法於營建資源之規劃

—以高科技廠房格子版施工為例

Applying Petri-net-based Ant Colony Optimization to

Resource Allocation: The Case of Waffle Slab Construction

吳曜安

Yao-an, Wu

指導教授：張陸滿 教授

Advisor: Luh-Maan, Chang, Ph.D.

中華民國98年6月

June, 2009

誌謝

這本論文代表的是我在台大土木所營建工程與管理組兩年的淬鍊。作研究的過程像品茗，滾水燙過，從壺口濾出茶的一生精華。舉杯聞香，嘗試著將學問的奧秘宇宙於一絲鼻息中凝聚。無法窺其全貌，故就口，卻被其炙熱的內涵震懾而停頓，待波瀾平息方一飲而盡。初入喉是苦澀，不安地於舌尖滾動，波瀾與時間一同減緩，最後靜止於舌根。沈寂，然後豁然開朗，化為甘甜襲來，金絲傾瀉而下，將心湖染成璀璨的金色。

湖心是一株茶樹的倒影。

我最敬愛的張陸滿教授，給予茶樹成長的寬闊天空，是光合作用所需的日光；林建元教授、郭斯傑教授、荷世平教授、曾惠斌教授有如清爽的微風，吹拂枝葉擺向不同的角度；莊子壽學長與沈恆光學長，有如甘霖滋潤了葉脈。倒影中，我見到不僅一株茶樹，思賢、紘瑞、雅晴、那桑提、炳全、詠珉、文守、琮暉、博文、智弘、浩偉、加麗、祐帆、恩霓，在R96這畝田地中，共同扦插下十五株茶苗，與他們共同成長的快樂，如同晨曦下的露珠般耀眼。我的室友兼好友忠毅，是夜空中的星星，時而提點我正確的方向；我親愛的玉芬，是默默支持著這株茶樹的莖；亦兄亦友的哥哥是聯繫枝幹與土壤的根，而紮根的土壤，是我最親的父母。擁有你們的愛，讓我不斷茁壯。

謹以本文獻與我的師長、好友、摯愛與家人。

吳曜安

民國98年6月29日於台灣大學營建工程與管理組研究室



摘要

對高科技產業而言，新廠興建必須考量企業產能規劃和景氣循環，導致初期規劃設計時間短；且工期緊湊，排程容錯率相當低。工程專案排程預定目標之達成，首重營建資源分派之規劃，工項必須給予適當的資源組合，方可得到與排程相符的結果。

為彌補傳統CPM網圖於工作資源需求描述之不足，本研究之目的係嘗試導入派翠網(Petri-net)，藉其善於表現資源與工作連結之特性，增進流程模擬的可信度。並透過蟻群最佳化演算法(Ant Colony Optimization)，建構基於派翠網的智慧型資源規劃模式，進行案例之模擬分析及資源數量分派最佳化。

潔淨室為高科技廠房專案中最具代表性之設施，是機台安裝與廠房氣密的先置條件，而潔淨室各工項中，又以格子版(Waffle Slab)結構工程最為關鍵。本研究以格子版結構工程為例，首先以派翠網建立格子版 workflow 模型，並實地調查格子版各工序之資源需求，分析不同資源組合下之工期與資源使用。與實際施工獲得之數值進行比較，發現派翠網能真實呈現資源配置對工率、工期之影響。建立可信的分析模型後，輔以蟻群最佳化演算法進行資源規劃。結果顯示，蟻群最佳化演算法能夠針對工程目標如工期、出工成本、工率等，迅速找出最適資源配置組合。本研究結合派翠網及蟻群最佳化演算法，提供一種迅速且可信的資源規劃方式，讓規劃者在專案工期與資源間取得平衡、減少資源閒置之浪費。

關鍵字：派翠網; 資源分派; 蟻群最佳化演算法; workflow 管理; 高科技廠房; 營建管理

Abstract

In High-Tech industry, the decision to construct a plant must take the business cycle and capacity requirement of the enterprise into consideration. To facilitate the products getting into market earlier, it often results in the shortening of the duration needed for finishing a plant. The crashing of the scheduling makes the construction less fault-tolerant. Resource planning has significant influence on the activities' completion, and each activity must be given adequate of resource to assure every task been finished as planned. The purpose of this research is to propose a more precisely workflow network to improve the reliability of scheduling, and use artificial intelligence to reduce the time of searching the best combination of resource.

Cleanroom is the most core facility of the High-Tech plan. In a high-tech plant construction, the activity of completing the cleanroom is normally along the project's critical path. Its completion is crucial to the timely delivery of the company's products. To timely complete the cleanroom, adequate resource must be supplied to the construction.

CPM is a widely used method on project scheduling, but it is inadequate to describe the linkage between resources and tasks. In this research, the Petri-net, a graphical tool in the modeling of workflow management and manufacturing process is proposed to improve the reliability of the workflow simulation. Base on the Petri-net workflow simulation and the collaboration of Ant Colony Optimization(ACO), the

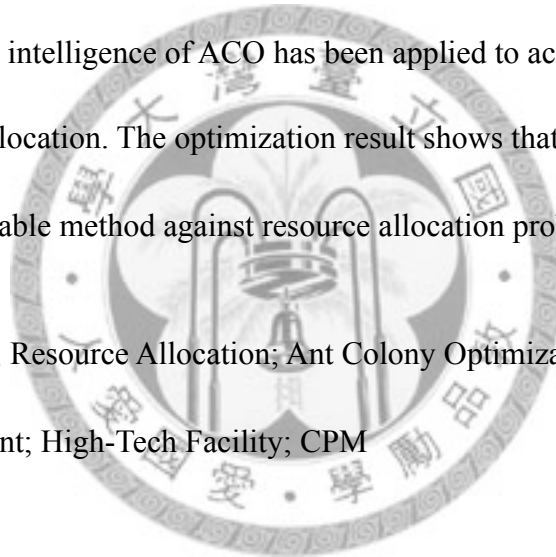
optimization time of resource combination is significantly reduced. Meanwhile, the reliability of simulation and optimization are greatly improved.

Firstly, a Petri-net workflow model of waffle slab is established, then the field data has been surveyed and imported into the simulation process. The results of field data and Petri-net simulation was then compared. According to the outcome, the result of Petri-net simulation has high relativity with the result of field construction work. This fact verifies that Petri-net is a suitable tool for modelling real construction workflow.

Secondly, an artificial intelligence of ACO has been applied to accelerate the planning process of resource allocation. The optimization result shows that the Petri-net based ACO is a fast and reliable method against resource allocation problem.

Key Word : Petri-net; Resource Allocation; Ant Colony Optimization;

Workflow Management; High-Tech Facility; CPM





目 錄

第一章 緒論.....	1
1.1 研究動機.....	1
1.2 定義問題.....	3
1.3 研究目的.....	3
1.4 研究範圍與限制.....	3
1.5 研究方法與流程.....	3
第二章 高科技廠房設施規劃與建設.....	5
2.1 高科技產業及廠房設施概述.....	5
2.2 高科技廠房新建專案特性.....	8
2.3 施工流程.....	10
第三章 文獻回顧.....	13
3.1 派翠網(Petri-nets).....	13
3.1.1 派翠網圖概述.....	13
3.1.2 高階派翠網圖.....	15
3.1.3 派翠網於營建專案管理之應用.....	16
3.1.4 派翠網與CPM法之比較.....	17
3.2 人工智慧演算法於資源規劃之應用.....	19
3.2.1 蟻群最佳化演算法概述.....	20
3.2.2 蟻群最佳化演算法計算流程.....	22
3.2.3 蟻群最佳化演算法於組合問題之求解.....	23

第四章 資源配置方式對工率之影響.....	25
4.1 高科技廠房格子版(Waffle Slab)結構工程案例背景.....	25
4.2 格子版結構工序分析.....	29
4.3 案例資源配置方式分析.....	32
第五章 以派翠網分析資源對工率之影響.....	37
5.1 建置階層式時間派翠網圖.....	37
5.2 輸入資源配置進行派翠網模擬推估.....	41
5.3 模擬結果分析.....	44
5.4 應用派翠網於資源組合分析.....	51
第六章 營建資源配置組合最佳化.....	53
6.1 基於派翠網之蟻群最佳化演算法模式建置.....	53
6.2 蟻群最佳化演算法計算流程.....	58
6.3 最佳化結果分析探討.....	63
第七章 結論與建議.....	71
7.1 結論與建議.....	71
7.2 研究貢獻.....	73
7.3 研究限制.....	74
7.4 未來研究建議.....	74
參考文獻.....	75
附錄：PetriStar程式碼.....	79

圖目錄

圖1.5.1 研究流程圖.....	4
圖2.3.1 半導體廠房基本FAB棟型式示意圖.....	10
圖2.3.2 FAB棟鋼結構施工示意圖.....	11
圖3.1.1.1a~c 派翠網執行範例.....	14
圖3.2.1.1a~d 自然界中螞蟻搜尋最短覓食路徑之原則.....	21
圖3.2.2.1 節點與合法解答示意圖.....	22
圖3.2.2.2 解空間與對應成本.....	23
圖3.2.1.2a~d 用蟻群最佳化演算法解旅行推銷員問題.....	24
圖4.1.1 案例FAB棟施工順序.....	25
圖4.2.1 格子版單格簡圖.....	29
圖4.2.2 格子版施工作業流程.....	30
圖4.2.3a~d 格子版結構場鑄施工流程.....	30
圖4.3.1a~b 灌漿時程及分區圖.....	32
圖4.3.2a~c 案例各時期物料配置簡圖.....	33
圖4.3.3a~f 資源配置與總出工數、工率關係圖.....	36
圖5.1.1 格子版派翠網圖.....	38
圖5.1.2 鋼筋綁紮次階網圖.....	39
圖5.1.3 FRP模組立次階網圖.....	39
圖5.1.4 PetriStar介面.....	40
圖5.2.1 系統循環圖.....	42
圖5.2.2 每小時進度更新次數與計算時間關係圖.....	43

圖5.2.3 片刻長度與計算時間關係圖.....	43
圖5.3.1 鋼筋工率模擬結果與實作比較圖.....	45
圖5.3.2 鋼筋出工數模擬結果與實作比較圖.....	45
圖5.3.3 FRP工率模擬結果與實作比較圖.....	46
圖5.3.4 FRP出工數模擬結果與實作比較圖.....	46
圖5.3.5 鋼筋工期模擬結果與實作比較圖.....	47
圖5.3.6 鋼筋工期模擬結果與實作比較圖.....	47
圖5.4.1 取料距離與每日出工數對工率關係圖.....	51
圖5.4.2 工區大小與每日出工數對總出工需求圖.....	52
圖6.2.1 蟻群最佳化演算法流程圖.....	62
圖6.3.1 PACO模式示意圖.....	63
圖6.3.5a~d 四項指標收斂情形(全體).....	67
附圖1. PetriStar系統架構圖.....	79

表 目 錄

表2.2.1 高科技廠房與商辦大樓建築結構比較表.....	8
表2.2.2 高科技廠房與商辦大樓專案特性比較表.....	9
表2.3.1 FAB建廠里程碑與主結構體工序.....	11
表2.3.2 FAB細部結構工序.....	12
表3.1.2.1 高階派翠網一覽表.....	15
表3.1.4.1 派翠網與CPM法性質比較表.....	18
表4.1.1 案例廠房特性調查表.....	26
表4.1.2 案例FAB包商預定進場時間表.....	28
表4.2.1 格子版工率影響因素調查.....	31
表4.3.1 鋼筋工率統計表(區域依施作先後排序).....	34
表4.3.2 FRP模組工率統計表.....	35
表5.2.1 資源可得性最大化工率調查表.....	41
表5.3.1 模擬測試環境.....	44
表5.3.2 派翠網模擬結果與實作之相關係數表.....	48
表5.3.3 派翠網模擬結果與實作之誤差表.....	49
表6.1.1 鋼筋綁紮資源配置原則與相關工作.....	53
表6.1.2 資源配置績效指標關係表.....	54
表6.1.3 成本函數設置.....	55
表6.2.1 ACO演算範例：步驟1.解空間配置.....	58
表6.2.2 ACO演算範例：步驟2.費洛蒙區域更新.....	59
表6.2.3 ACO演算範例：步驟3.費洛蒙蒸發.....	60

表6.2.4 ACO演算範例：步驟4.迭代結束之費洛蒙分佈.....	60
表6.2.5 ACO演算範例：步驟5.全域蒸發後之費洛蒙分佈.....	60
表6.2.6 ACO演算範例：步驟6.演算結束獲得資源配置建議.....	61
表6.3.1 固定工區面積之工期最佳化(業主).....	64
表6.3.2 固定工區面積之出工數與暫存區需求最佳化(分包商).....	65
表6.3.3 非固定工區面積之暫存區需求、取料距離及工率最佳化.....	66
表6.3.4 固定工區面積之四項指標最佳化(全體).....	67
附表1 Place Data Loader Format (example for steel assembling).....	96
附表2 Place Data Loader Format (example for steel assembling).....	96
附表3 Graph Loader Format (steel assembling).....	97
附表4 Resource Limitation Loader Format (steel assembling).....	97



第一章 緒論

1.1 研究動機

高科技產業是我國重要的經濟命脈，廠房設施則為科技產品之生產基礎，是高科技產業提昇競爭力不可或缺之存在。廠房設施之興建與規劃必須考量景氣與企業產能規劃，在競爭激烈、變化迅速的高科技產業中，廠商皆希望能儘早完成廠房進行生產以提昇市佔率，造就高科技廠房工期短之特性[王明信, 2004]。

營建工程的設計規劃階段，影響工程生命週期的程度最大，對高科技廠房而言，初期的規劃設計時間短，加上土木與機電界面眾多，排程複雜度比一般建築物更高[張書萍, 2000]。在工程執行階段，因生產機具有進場安裝之排程需求，排程緊湊不容延遲，趕工情形頻繁。

工項之所以產生延遲的原因不外乎：排程錯誤、工程風險意外、品質缺失重作、物料不足及其他行政因素導致。現場工程師在訪談中指出，工程延遲的主因為：品質缺失重作、工序安排不當、工人素質不佳、物料不足、出工不足、運補距離過長、施作界面衝突等[現地調查]。

由以上調查發現，除了品質與排程以外，資源規劃不良亦為工程延遲的主因。過往在資源規劃的領域上之主軸有二，一為資源限制專案排程(Resource Constrained Project Scheduling)，二為資源撫平(Resource Leveling)。近年來受到精實營建(Lean Construction)之觀念影響，亦開始提出適時、適量之資源規劃方式。專案資源排程之目標為，使工作項目能適時取得所需資源進行施作[蕭博文, 2002]，換言之，若無足夠資源，工作進度會受到相當的影響。

專案管理常用的排程管理手法是CPM法，然而CPM法對於施工情況的描述較為簡略，尤其在工作執行條件上，僅以先後關係為要件，對於資源限制、作業

間互相影響、專案特性無法詳盡描述[許書凱, 2006]。相對的，在工作流管理中則對工項狀態和需求有詳盡的定義，各種工作流管理手法中，派翠網是一套圖形化的語言，其直覺式理解、邏輯嚴謹的特性，適合運用在流程規劃上。實務上，派翠網廣泛應用於製造流程的建模；學術上，則能配合各種演算法進行流程的最佳化、驗證與績效分析[林訓宇, 2002]，亦有學者研究派翠網在營建排程上的使用方式[A.Sawhney, 1997],[Ron, 1997]。

電腦科技的演進讓人工智慧成為輔助決策的重要工具，成熟的人工智慧可以扮演經驗老道的工程師，作出合理的決策。過往亦有許多軟體嘗試自動進行排程的計算，如GHOST(1987), CMM+(1996), CasePLAN(2004), CLCPM(2004)...等等，皆獲得不錯的成效。在資源規劃的課題上，過去曾透過基因演算法、模擬退火法、線性規劃等等方法求解資源受限排程與資源撫平問題[Yen, 2005]。本研究則著眼於借重人工智慧的速度，進行資源分派之最佳化解題。

綜合以上討論，工程專案排程預定目標之達成，營建資源分派規劃為相當重要之一環，工項必須給予適當的資源組合，方可得到與排程相符的結果。高科技廠房專案中，潔淨室為重點工作項目之一，是機台安裝與廠房氣密的先置條件，而潔淨室各工項中，又以格子版(Waffle Slab)工程最為關鍵。本研究希望透過研究找出更適合的資源配置方式，並輔以人工智慧進行資源之最佳化，針對工程目標作資源最適規劃，在專案工期與資源間取得平衡、減少資源間互相等待之浪費，提昇資源規劃之績效並縮短規劃時間。

1.2 定義問題

本研究中所指的資源規劃課題為「資源分派」(Resource Allocation)，在公共資源、企業管理、電腦系統、供應鏈、營建管理等範疇皆有資源分派不均或者閒置的問題。在過去於營建領域的研究中，排程研究之基礎網圖以CPM為最大宗，以工項為基準，未考慮施工與資源之關係，且工項中的變異性皆以統計之機率模型表示，當估算誤差較大時，動輒歸咎於工程個案性質不同。本研究調查發現，施工績效與資源配置組合高度相關，隨著工程的進步，營建工程必須求取更精確的工作估時，因此要詳細考慮工項中各工序環節施作與等待資源的時間，方可獲得較貼近真實施工的模擬結果。

1.3 研究目的

本研究之目的為，提出可考慮資源配置方式對工作績效影響之網圖，彌補傳統CPM網圖於工作資源需求描述之不足，並以人工智慧演算法建構基於派翠網的智慧型資源規劃模式，針對工程目標提出資源最適規劃方式，在專案工期與資源間取得平衡、減少資源間互相等待之浪費，給予工程師合理的資源規劃建議。

1.4 研究範圍與限制

研究案例為高科技廠房之潔淨室格子版(Waffle Slab)結構工程的工作流程，派翠網分析範圍是鋼筋綁紮及FRP模組立之施作循環，蟻群最佳化演算法則運用於鋼筋綁紮工作之資源分派最佳化。

1.5 研究方法與流程

導入工作流(Workflow)管理中的派翠網(Petri-net)，並進行格子版結構工程之模擬分析。首先以派翠網建立格子版工作流模型，並實地調查格子版各工序之

工率與工作資源需求項目，分析不同資源組合下的工期長度與資源使用率。再藉由蟻群最佳化演算法(Ant Colony Algorithm, ACO)，建置智慧型的資源規劃模式。

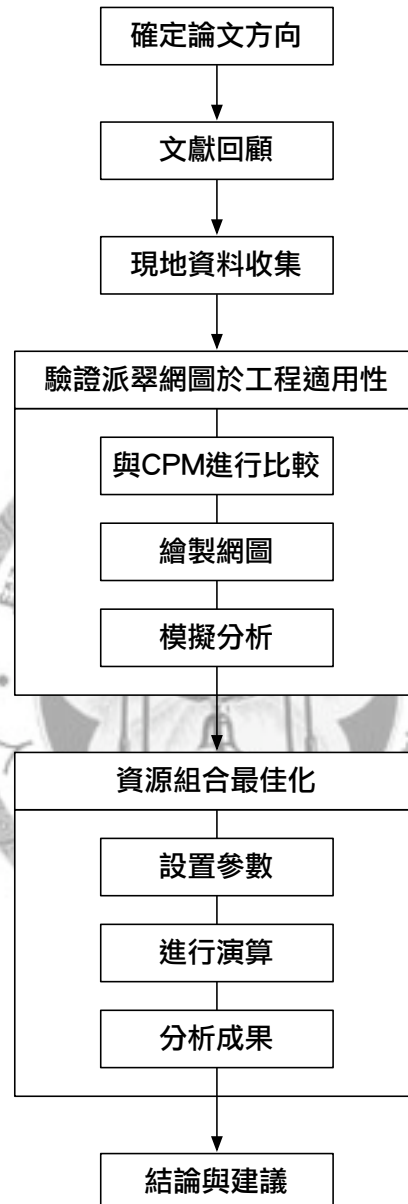


圖1.5.1 研究流程圖

第二章 高科技廠房設施規劃與建設

2.1 高科技產業及廠房設施概述

台灣自1960年代起陸續有跨國企業來台設置工廠，工研院於民國62年成立後，於民國65年引進半導體製程技術。新竹科學園區(Hsinchu Science Park)在民國69年底正式成立，為台灣科技業重鎮，為台灣之技術與經濟命脈。

根據國家科學委員會於民國86年出版之科技白皮書之描述，高科技電子產業具有「資金密集、技術密集、高級人力密集」三項特色。過往的高科技產業約可分為六大項[張書萍, 2000]：(1)積體電路、(2)電腦及週邊設備、(3)通訊、(4)光電、(5)精密機械、(6)生物技術。依照民國97年5月2日所修正之「新興重要策略性產業屬於製造業及技術服務業部份獎勵辦法第五條第一項附表」，台灣目前重點發展之產業可分為以下十項(經本研究整理)：

1. 3C工業：資訊硬體工業、通訊工業。
2. 精密電子元件工業：如印表機、藍光光碟片、液晶投影系統之光閥、彩色平面顯示裝置、液晶顯示器之平面光源、光學膜、氮化鎵藍及綠或白光發光二極體(僅從事封裝者除外)、光通訊主被動元件及裝置、增層式基板、固態晶片型電解電容器、高頻無線通訊主被動元件、電力電子元件、半導體材料及裝置、可充電式鋰電池或其材料、燃料電池、發光二極體磊晶用品棒、晶圓、白光發光二極體照明設備
3. 精密機械設備工業：半導體製程設備、平面顯示器Array段及Cell段製程設備、電腦控制精密切削工具機、伺服精密沖床、電腦數值控制產業機械、精密機械零組件、智慧型機器人、車輛運輸系統
4. 航太工業：飛機機體系統及其零組件、飛機維修與改裝
5. 生醫及特化工業：醫療保健工業、製藥工業、特用化學品

6. 綠色技術工業：環保科技材料、資源化產品、環保處理設備及材料、新及淨潔能源設備及材料。
7. 高級材料工業：高清淨度特殊合金材料、鎂合金材料及其製品、特殊鋁合金擠錠、擠型及其製品、鈦合金材料及其製品、鍍靶材料、機能性高分子材料、高級纖維材料、高科技紡織品。
8. 奈米技術工業：應用或開發奈米技術且符合下列要件之新產品：產品中有一尺度在奈米尺度範圍(100nm以下)，且該產品須因該奈米尺度而顯現新功能特性之產品。
9. 安全產業：多重辨識門禁系統。
10. 技術服務業：數位內容產品及服務、高階積體電路設計、自動化或電子化工程服務、無線及寬頻光纖通訊測試服務、環境保護工程技術服務、生物技術服務、提供屬製造業之溫室氣體排放量減量工程技術服務、節約能源或利用新及淨潔能源工程技術服務、研究發展服務、整合性綠色化設計及製程技術服務業、汽車開發設計、經行政院指定之產品或技術服務項目。

從以上產業的內容可以發現台灣的產業發展方向，除了持續發展精密技術之外，亦投入再生能源及奈米科技之研發，並從純製造業轉型為科技服務業。

高科技產業競爭求新求快，與生產製造最密切相關者即為廠房設施。深入高科技廠房設施的範疇，最獨特者為潔淨室(Cleanroom)，潔淨室對落塵、通風、溫度皆嚴格控制，保護產品製程不受外界污染。從國際標準ISO-14644的定義可知，潔淨室是為了管控空氣中的微粒、溫溼度、氣體運動、振動噪音等等環境因素所建造的。國內的高科技產業有相當大的比例在潔淨室中進行產品製程，其中積體電路、光電、生物科技這三類產業之廠房中，潔淨室面積相當廣大。平均來說，積體電路有10700平方公尺；光電為8300平方公尺；生物科技則為3100平方公

尺，根據潔淨室等級不同，每平方公尺造價為5~30萬新台幣，而廠房中非潔淨室區域造價則為2~3萬，最多不超過10萬新台幣[張書萍, 2000]。

潔淨室的品質關係著產品的良率(Yield)，對高科技廠房建廠專案而言，潔淨室結構體工程是整個專案中出工數最多、施工面最多、且對機具安裝最為關鍵之環節。因此，掌握潔淨室工程是此類型專案中期的重點。



2.2 高科技廠房新建專案特性

基於將高科技廠房與一般工廠做出區隔，本研究中將高科技廠房範圍定為：主製程區位於潔淨室中，且有恆溫、恆濕、恆壓、抗微震需求之積體電路、光電、生技等高科技產品生產工廠。一個完整的高科技廠房會包含三個主結構體：辦公(Office)、製造(Fabrication, FAB)、機電(Central Utility Plant, CUP)三者。其中FAB棟與CUP之屬性較符合本研究對高科技廠房之定義，Office棟則較接近一般辦公大樓之屬性，但仍會用管橋(Bridge)與工廠連結供系統管控及人員進出。從建築工程之角度來看，高科技廠房結構與一般建築不同點可參考表2.2.1及表

2.2.2[本研究整理調查][張書萍, 2000][王維志,周世傑, 2000][楊立華, 2001]：

表2.2.1 高科技廠房與商辦大樓建築結構比較表

項目	高科技廠房	商辦大樓
樓層數目	地下1~2層，地上5層	大多10層以上
樓高	因機具需求，皆挑高(6M以上)	因設計需求，選擇性挑高
跨度	跨度大以便提供更多生產空間	一般建築物之跨度
樓層載重	機具安裝需求，承重能力較高	商場或辦公需求，載重一般
結構體	RC(機台)+鋼構(大跨度需求)	無特殊要求
設計模組化程度	結構模組化以便縮短設計規劃時間，特殊外觀需求少	因外觀設計需求而異，一般而言較多變
結構制震	產品製程需求，必須控制微震	結構安全需求，重視地震安全
特殊結構	潔淨室(Cleanroom)、格子版(Waffle Slab)	挑空(Atrium Space)、露台(Balcony)
結構表面	Epoxy漆	磁磚、油漆
環境綠化	潔淨度考量，較無綠化	造景與綠建築導向之植栽
空調系統	講求恆溫、恆濕、潔淨之系統	以使用者舒適度為導向之系統

表2.2.2 高科技廠房與商辦大樓專案特性比較表

項目	高科技廠房	商辦大樓
工期長度	配合產品週期，約10~12個月	無縮短工期之需求
主導工程	機具工程(受限於機具採購及安裝時程)	結構體工程
建築里程碑	土方→基礎→鋼結構基礎區→鋼結構吊裝→內部樓層結構→潔淨室安裝	土方→基礎→地上結構
品質管理	基於製程水準需求高品質之廠房建築	基於銷售考量品質
成本分佈	生產機具佔絕大部份金額，一般而言總價約數十至數百億	以結構體為主，其次為裝潢
工程界面	工期短造成同步工程較多，加上土建及機電一同施作，界面複雜度較高	施工按部就班，較單純



2.3 施工流程

高科技廠房施工之流程必須考量潔淨室與機台安裝的需求，因此會將結構水密(屋頂密封)列為優先，但結構氣密(外牆密封)受限於機台搬運與管線安裝，必須在內部建築完成後方可施作[本研究調查]。高科技廠房整體結構可用下圖表示(圖中無塵室即為潔淨室)：

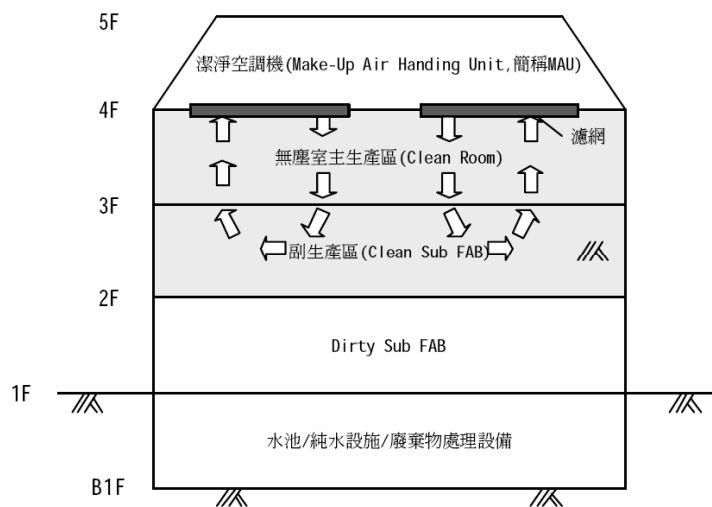


圖2.3.1半導體廠房基本FAB棟型式示意圖[張書萍, 2000]

典型的半導體FAB棟3F是潔淨室區域，放置大量製程機具，而2F則為Sub Fab區域，設置製程機具的附屬設備。潔淨室與Sub Fab之間的機具透過管線連結，加上廠區的迴風需求，因此將3F樓板設計為格子版(Waffle Slab, 俗稱為洞洞版)，以利管線穿越與迴風。各式高科技廠房儘管工廠設計不同，其樓層規劃仍受限於潔淨室的設置而大同小異。本研究於T半導體製造公司訪問規劃設計團隊與建築團隊，確立了高科技廠房之建廠工序(表2.3.1、表2.3.2)。

另外，從訪談中亦得知，高科技廠房之FAB棟建築工程困難點在於，鋼結構之安裝會與RC結構產生重大衝突，因為鋼結構無法直接橫跨廠房，必須由中央SRC柱進行支撐，如圖2.3.2所示之橘色部份。此時期受限於中央柱工程之進行，

大型吊車(400噸以上)與鋼結構材料進入一樓內部進行施工，因而無法進行內部RC結構體。待鋼結構完成，吊車撤場之後方可進行2F及3F的結構體工程。

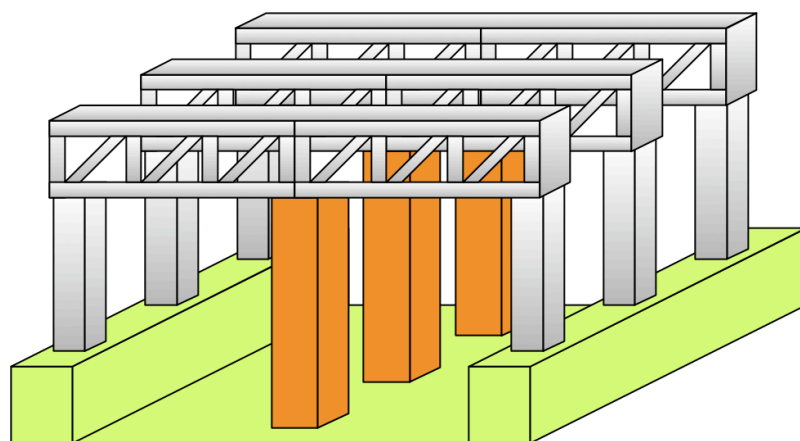


圖2.3.2 FAB棟鋼結構施工示意圖

表2.3.1 FAB建廠里程碑與主結構體工序

FAB建廠重要里程碑	主結構體工序
1. 動土典禮	1. 土方開挖
2. 基礎板進場	2. 基礎板施工(鋼柱區)
3. 土方開挖完成	3. B1~1F樓版
4. 鋼構進場	4. 1F~3F鋼柱區RC結構
5. 基礎板完成	5. 立中央柱
6. 2F板施作	6. 鋼構吊裝
7. 3F板施作	7. 4~5F DECK&GRATING結構
8. 鋼構吊裝完成	8. 中央SRC柱
9. Clean Room進場	9. 2F 非鋼構區RC結構
10. 2F板完成	10. 3F 非鋼構區RC結構
11. 3F板完成	
12. 取得使用執照	

表2.3.2 FAB細部結構工序

RC結構工序	鋼結構工序
梁板	中央SRC柱
<ol style="list-style-type: none"> 1. 取得區域 2. 重型架搭設 3. 釘模板 4. 放樣 5. 鋼筋綁紮 6. 鋼筋綁紮品質檢查 7. 混凝土澆置 8. 重型架拆除 9. 移交區域 	<ol style="list-style-type: none"> 1. 取得區域 2. 建立吊車、運料動線 3. 清除動線上障礙 4. 鋪設覆工板 5. 吊車進入 6. 材料進駐 7. 中央柱吊裝
柱牆	鋼結構吊裝
<ol style="list-style-type: none"> 1. 取得區域 2. 鷹架搭設 3. 鋼筋綁紮 4. 鋼筋綁紮品質檢查 5. 釘模板 6. 混凝土澆置 7. 鷹架拆除 8. 移交區域 	<ol style="list-style-type: none"> 1. 取得區域 2. 建立吊車、運料動線 3. 清除動線上障礙 4. 鋪設覆工板 5. 吊車進入 6. 材料進駐 7. 桁架電焊組裝 8. 桁架吊裝 9. 小梁吊裝 10. 噴防火漆

第三章 文獻回顧

3.1 派翠網(Petri-nets)

在本章節中，將就派翠網(斐氏網、派屈網、Petri網)之發展與特質進行介紹。派翠網是一種圖形化的數學建模工具，特別適合用來描述、研究工作流程。其性質適合用於描述作業之同步、非同步、離散、平行、未確定事件、隨機事件。派翠網圖形化的特質有助於使用者溝通及作業動態模擬；透過其數學性質，派翠網能建立工作流程的數學模型以利分析[Tadao Murata, 1989]。

3.1.1 派翠網圖概述

德國科學家Carl Adam Petri在1962年的論文中首次提出派翠網的理論，其目的是為了在計算機系統中描述物理行為。這套理論在不久之後(1970年代)受到MIT的研究團隊重視，在美洲及歐洲蓬勃發展，為重要的自動化理論之一。

派翠網中有以下四種基本圖形：

1. Place(庫所)：圓形的圖案，用來標示變遷所需要的執行條件。
2. Transition(變遷)：方形的圖案，用來標示作業。
3. Token(令牌)：黑色圓點，用來表示庫所中的工作執行條件到達情形。
4. Arc(有向弧)：庫所與變遷之間用箭頭連結，表達作業需求與產出之關係。

派翠網的基礎執行程序邏輯則如下[Tadao Murata, 1989] [Ashok & Ganesh,1998]：

1. 確認變遷 i 的所有輸入庫所皆持有令牌。
2. 若所有庫所皆持有令牌，該變遷將處於啟動狀態(Enabled)。

3. 處於啟動狀態之變遷，將依照其觸發條件(Triggering Condition)，決定變遷在何種條件下發生(Fire)。
4. 當觸發條件達成時，所有的輸入庫所將被消耗掉一個令牌，發生變遷。
5. 發生變遷後，於所有的輸出庫所產生一個令牌。
6. 當所有的變遷都無法啟動時，流程停止(Dead)。

運用為流程的建模時，四種基本圖形將有以下對映關係：庫所對應至工作資源，變遷對應至工作執行，令牌對應至工作資源抵達之情形，有向弧對應至該工作與資源輸入、輸出之關係。圖3.1.1.1a~c是一個簡單的派翠網工作範例：

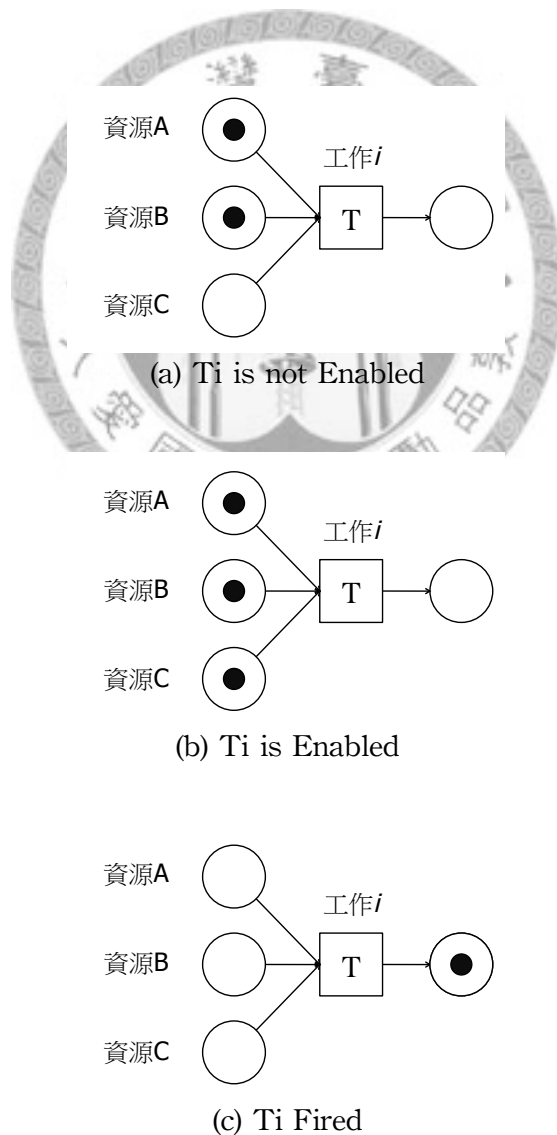


圖3.1.1.1a~c 派翠網執行範例

3.1.2 高階派翠網圖

由於基礎的派翠網圖僅用來表達工作流程的順序、相互關係、資源需求與產出，後續的研究為了更妥善的表現工作性質，加入了時間、顏色、機率、階層等屬性。這類型的派翠網比典型派翠網擁有更詳盡描述流程之能力，因此稱之為高階派翠網(High Level Petri-net)。下表將就各種高階派翠網進行介紹。高階派翠網除了各自的功能外，亦能混合其他類型網圖發揮更強之效用，例如時間著色派翠網(TCPN)或隨機時間派翠網(STPN)等等。

表3.1.2.1 高階派翠網一覽表

名稱	概念	最早出處
時間派翠網, TPN (Timed Petri-nets)	將各個變遷加入發生時間(Firing Time)以利估時。	Ramachandani, "Analysis of synchronous concurrent systems by timed Petri nets", 1974
顏色派翠網, CPN (Colored Petri-nets)	將令牌著色，能於同一庫所中表現不同的工作資源，可收斂資源項目過多之網圖。	C. R. Zervos and K. B. Irani, "Colored Petri nets: Their properties and applications" 1977
隨機派翠網, SPN (Stochastic Petri-nets)	為了表現真實世界中工作流程之不確定性，賦予變遷所需時間、流程路徑選擇、等項目機率。	S. D. Shapiro, "A stochastic Petri net with applications to modelling occupancy times for concurrent task systems", 1979.
階層式派翠網, HPN (Hierarchical Petri-nets)	將網圖階層化，賦予層級關係並收斂過於複雜之網圖。	S. Porat and M. Yoeli, "Towards a hierarchy of nets", 1984

3.1.3 派翠網於營建專案管理之應用

自1962年派翠網發表以來，在流程建模方面已被廣泛運用於通訊系統、電腦軟硬體、製造流程、安全性評估、排隊系統等，而流程分析方面則適用於流程架構分析、績效評估、電腦模擬等[Ron, R. Wakefield, 1997]。

儘管派翠網在資訊管理及工業工程界獲得許多成功，但在營建專案管理方面則始終處於研究階段，1997年Ron, R. Wakefield發表了派翠網於營建專案的建模方法，該論文建立了一個土方工程的隨機時間派翠網進行電腦模擬，Wakefield認為，營建工程之特質如：工作優先權、隨機性、資源多方共享、等待行為、作業間互動等能透過派翠網作出良好的描述[Ron, R. Wakefield, 1997]。同年底，Anil Sawhney亦針對派翠網於營建專案上的適用性發表研究成果。該論文研究派翠網於營建管理中三方面的適用性：(1)營建專案的階層化、(2)加入風險及不確定性、(3)分析資源動態。Sawhney之研究結論指出，透過派翠網，能適切的表達循環及隨機性作業，且庫所(Places)能表現資源共享與動態分派，能夠幫助管理者預見資源之可得性(Availability)問題[Anil Sawhney, 1997][Heng Li, 1998]。

大陸之學者偏向運用於實際工程，針對隧道施工過程[楊學紅等,2002]、大壩混凝土[劉全等, 2005]、鋼結構施工[鄭衛國等, 2006]等作業進行建模，另外亦有學者利用高速公路施工案例研究派翠網之資源瓶頸辨識能力[張紹陽等, 2006]。

3.1.4 派翠網與CPM法之比較

目前在營建專案管理上主要使用的網圖為CPM法，惟CPM法雖簡明易用，仍受限於以下問題，導致實務應用上有不足之處[許書凱, 2006]：

1. 未考量專案資源限制
2. 作業間施作影響關係不明確
3. 未考量專案獨特性之排程因子(工法、工地現場狀況、空間與時間衝突)

過往於營建管理專案導入派翠網之嘗試亦提到，傳統的流程建模方法欠缺動態、工程隨機性質之表現[Anil Sawhney, 1997]。派翠網則提供了一套視覺化、嚴謹的工具，可適應營建工程同步作業多、工項相互影響性高、隨機性高、資源問題複雜的性質。下頁將就CPM法與派翠網兩者的性質作比較。



表3.1.4.1 派翠網與CPM法性質比較表[本研究調查]

項目	派翠網	CPM
表現手法	表現工作(變遷)與資源(庫所)之輸入與產生關係，工作間關係用資源關係表現。	表達工作前置與後續之間的關係，不同工作間的關係以箭頭連結表現。
圖形化程度	用不同圖形表現工作與資源項目，加入顏色後能表示不同資源項目。可表現OR閘、AND閘等路徑選擇。	使用簡明的表格表現工項的起始時間、結束時間等等性質，要徑用不同顏色表示之。
資源關聯性	工作之執行(發生)必須消耗令牌，必須完整連結所有輸入及產出之資源。	改良之CPM法可附加每日消耗之資源於工項上，資源統計後需另表於時間軸上。
動態追蹤	可檢視各個時間點需要的資源抵達情形。	可知各時段資源需求量，但無法從網圖得知抵達情形。
工期估計	需加入資源取得之排程，能考慮工作等待資源取得的情形，亦可加入隨機，獲得更可靠之工期估算。	單時估計法，未考量工作資源取得與否對工期之影響。改良之CPM法可採用PERT的概念進行隨機估時。
網圖複雜度	資源輸入與產出之關係需定義完整，網圖容易膨脹。	簡明表現各工作之前後關係與所需工期。
網圖收斂技術	已發展出階層化、資源著色、節點收斂等技術。	使用WBS技術進行收斂。

3.2 人工智慧演算法於資源規劃之應用

在人工智慧的範疇中，解題式人工智慧旨在所有可能解的集合之中找到最符合需求者。根據Stuart Russell & Peter Norvig 合著之 *Artificial Intelligence: A Modern Approach* 的分類，人工智慧可分為以下六大類[Stuart Russel et al., 2003]：

1. 解題(搜尋最佳解)
2. 知識理解(專家、知識庫)
3. 邏輯應對(規劃)
4. 不確定之知識與推理(機率與決策)
5. 學習型(監督、非監督)
6. 溝通、感知、反應(知覺機器人)

資源分派問題(Resource Allocation Problem)在1960年代起便受到重視 [Wiest, 1964]，其後有許多學者進行資源分派模式的探討，並發展出Integer Programming、Branch-and-bound、Dynamic Programming等模式，但運用到真實的分派問題求解時，則受制於計算不易而顯得不切實際[Tarek, 1999]。自1990年代起，電腦模擬方法與啟發式(Heuristic)人工智慧實用性大幅提昇，學者運用諸如：模擬退火法(Simulated Annealing)、線性規劃(Linear Programming)、基因演算法(Genetic Algorithm)、禁忌搜尋法(Tabu Search)、蟻群最佳化演算法(Ant Colony Optimization)等進行求解，其中以資源撫平(Resource Leveling)、資源限制之工期排程(Resource Constrained Scheduling)為最大宗研究[本研究整理]。

資源撫平的目標是在不影響工期的情況下緩和資源變動性[Tarek, 1999]。資源限制下的工期排程則是在考慮資源有限的情形下進行排程[Merkle, 1999]。這兩個方面的研究皆是在一定的限制中取得最良好的資源配置。儘管已有許多學者進行研究，但研究皆朝向降低資源限制導致的排程衝擊。

回到高科技廠房之興建，資源配置數量需求受限於排程，必要時得擴充資源，也就是進行所謂的趕工，此時資源分派的問題從資源數量變動最小化轉變為：工期最小化、資源可得性最大化。此外，台灣地狹人稠，基地面積往往相當狹隘，物料暫存區不足的問題對施工影響相當大，亦時常有工地過度擁擠的情形，在竹科尤其明顯。歸納高科技廠房專案之營建資源規劃問題，資源最佳化之研究方向應為：在工期限縮的情況下進行資源配置的最佳化，使作業資源充足不間斷，同時避免資源閒置造成的浪費。

儘管高科技廠房的資源分派問題與一般建築截然不同，該問題仍可建置為數學式進行求解，因此本研究將透過解題式人工智慧進行此類型問題的探討。資源配置比例問題屬於組合最佳化問題(Combinational Optimization Problems)，而蟻群最佳化演算法對該類型問題特別拿手[章允建, 2005]，因此本論文選擇蟻群最佳化演算法進行研究。

3.2.1 蟻群最佳化演算法概述

蟻群最佳化演算法是採取隨機、全局式的搜尋演算法，由Dorigo等學者在1991年所提出的一種最佳化演算法，經過數年的發展，該演算法在處理旅行推銷員(Traveling Salesman Problem)、二次分配(Quadratic Assignment)、車間排程(Job Shop Scheduling)等組合最佳化問題取得相當良好的成效[Marco Dorigo et al.,1996]。其演算法精隨為，利用過往的經驗在可能的解空間中不斷探索、開發，提出更好的解答，以逼近最佳解[章允建, 2005]。蟻群最佳化演算法被歸類為群集智能(Swarm Intelligence)，所謂的群集智能是透過群體的智慧與經驗，處理單一個體難以解決的問題。這類型的演算法可以避免演算法常見之弊病，如題目難度提高會導致過度複雜、容易落入區域解等[馬良, 2000]。

蟻群最佳化演算法(Ant Colony Optimization, ACO)是經由觀察自然界中蟻群尋找最短覓食的方法而得到的。當螞蟻發現食物，會前往食物所在的地點進行搬運，在螞蟻移動的路徑上會留下費洛蒙以供後面的螞蟻跟隨其路徑。然而，螞蟻與食物之間不一定是直線、單一的路徑，而是存在各種可能的路線，較晚出發的螞蟻會根據各路徑上費洛蒙強弱程度為路徑選擇依據，即費洛蒙殘存多的路線較容易被選擇。自然界中的費洛蒙會隨著時間蒸發(Evaporation)而減少，由於各路線所花費的搬運時間不同，較長的搬運路線費洛蒙消散較快，螞蟻選擇路徑之天性便能讓螞蟻傾向採取較短的路徑，但仍有機會搜尋未知的路徑。其概念可用下頁組圖作簡單的演示(順序 $a>b>c>d$)：

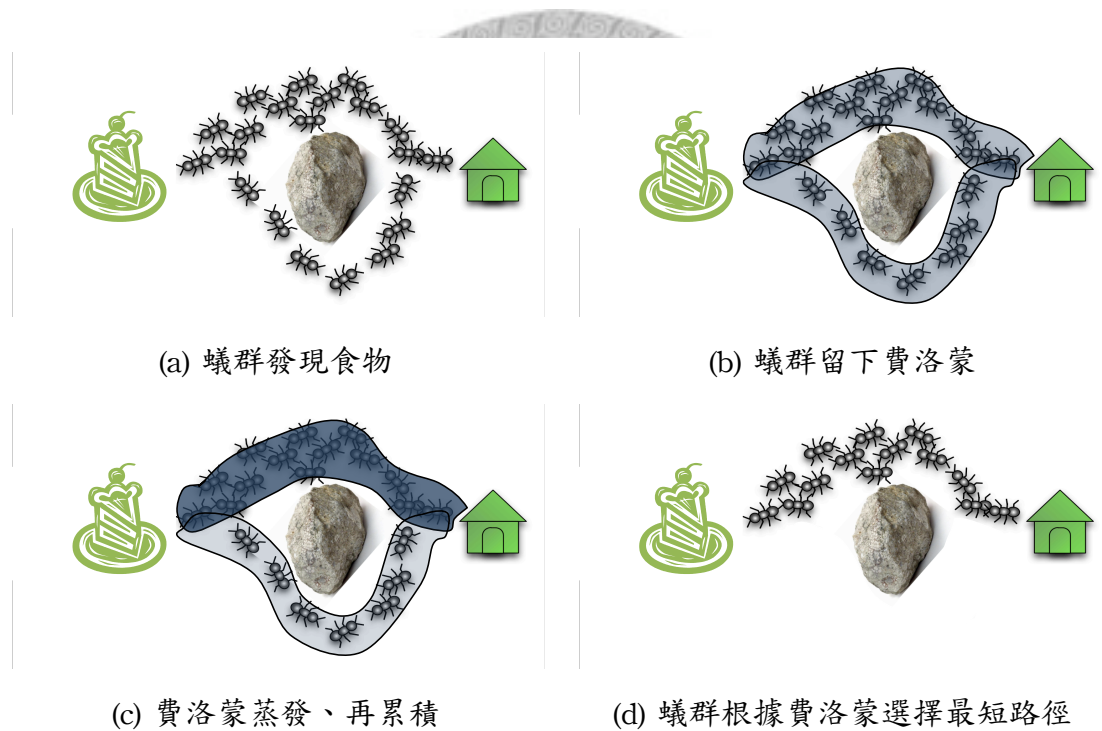


圖3.2.1.1a~d 自然界中螞蟻搜尋最短覓食路徑之原則

3.2.2 蟻群最佳化演算法計算流程

在使用蟻群演算法時，必須進行以下準則設置[章允建, 2005], [Merkle, 2002]：

1. 定義節點(Node)：解空間由眾多節點的合法解答(Eligible Activity)構成，每個節點皆有一定解答範圍(圖3.2.2.1)，各節點之關係可關聯或獨立、選擇順序可依序(Sequenced)或同步(Synchronized)，舉螞蟻覓食的例子來說，即為各個叉路之方向選擇；對於交通號誌管控來說，即為各號誌的時間長度。螞蟻的例子中前一個節點之選擇對會影響下一個節點能選擇的方向，但交通號誌則不受影響。

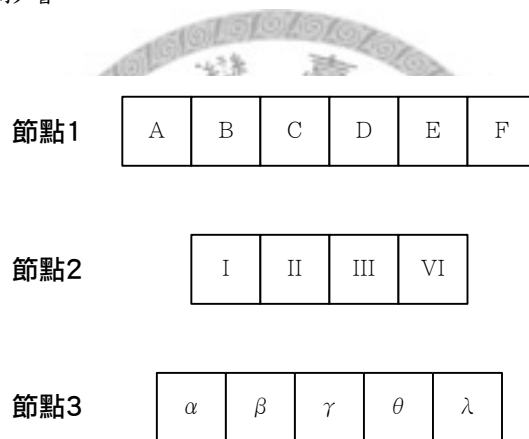


圖3.2.2.1 節點與合法解答示意圖

2. 定義成本(Cost)：成本為最佳化成本的指標，影響到費洛蒙消散的程度。例如在交通號誌管控上，成本為車輛通行的總時間。在多重成本目標的求解上，亦可設定不同權重。圖3.2.2.2為合法解答組合與成本之關係範例，以球體半徑表示成本高低。

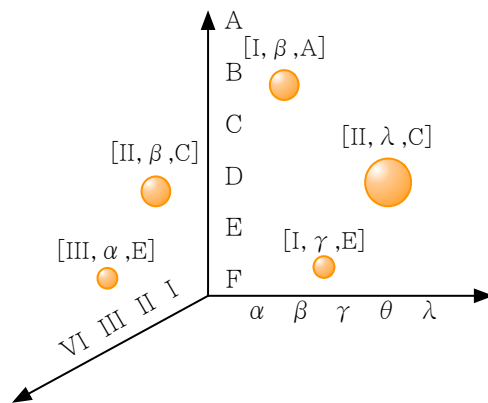
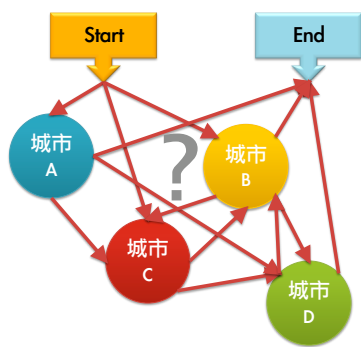


圖3.2.2.2 解空間與對應成本

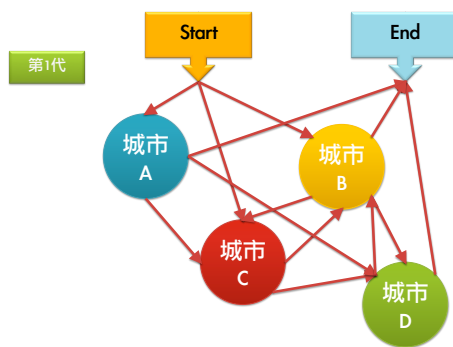
3. 建立限制式(Constrain)：針對不同的問題，選擇合法解答時必須遵循之限制，例如旅行推銷員不能選擇曾走過的城市。
4. 費洛蒙更新(Pheromone)：費洛蒙為螞蟻經過節點時選擇合法解答的依據，在真實世界中費洛蒙會自然蒸發，較短的路徑由於花費時間較短，費洛蒙蒸發較少，而螞蟻亦傾向找尋費洛蒙氣味較強烈的路徑，蟻群演算法便是運用這種特性使演算結果收斂。

3.2.3 蟻群最佳化演算法於組合問題之求解

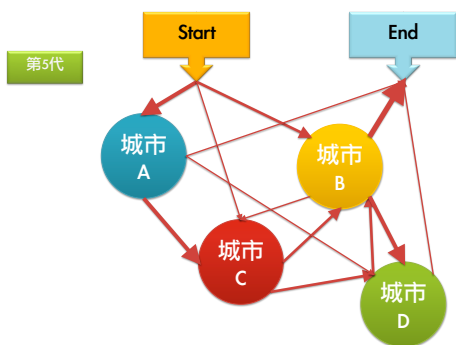
本節以著名的旅行推銷員問題為例，每隻螞蟻可視為一個代理人(Agent)，自由進行搜尋後，依照費洛蒙蒸發的結果改變各路徑的選擇機率，即得到逼近最短路徑的結果。下頁組圖是旅行推銷員問題的範例，代理人必須從起點經過四個城市抵達終點，然而各路徑之旅行成本不盡相同，如何挑選最經濟的一條路徑為其課題。對於簡單的覓食路徑問題，成本僅有時間，但真實問題之求解，則需針對問題定義成本函數，進行費洛蒙蒸發程度的計算，透過多次迭代(Iteration)的演算，即可去蕪存菁留下最理想的路徑(圖中箭頭粗細代表費洛蒙多寡)。



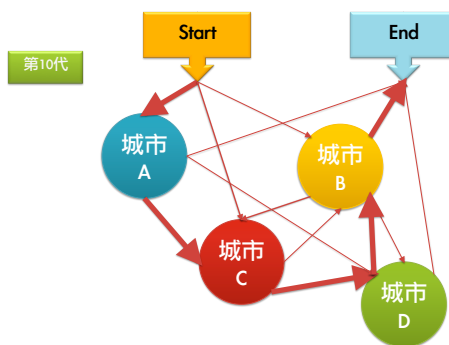
(a) 旅行推銷員問題示意圖



(b) 第一代(各路徑之費洛蒙相同)



(c) 第五代(優秀路徑之費洛蒙較高)



(d) 第十代(不良路徑幾乎無費洛蒙)

圖3.2.1.2a~d 用蟻群最佳化演算法解旅行推銷員問題

除了旅行推銷員問題之外，Dorigo等學者亦嘗試過如Job Shop Problem等等問題之求解。其研究指出此演算法的特性有三：(1)分散(Distributed)、(2)自動催化(Autocatalytic)、(3)貪婪式啟發(Greedy Heuristic)。特性一能避免落入局部解(Local Optimal)，特性二是較優解會具有吸引較多螞蟻進行嘗試，特性三則是事先給予較佳解的範圍，減少無用搜尋[Dorigo, 1996]。

第四章 資源配置方式對工率之影響

4.1 高科技廠房格子版(Waffle Slab)結構工程案例背景

本研究於新竹科學園區T公司之新建半導體晶圓廠案例實地進行調查研究，透過訪談業主及承包商負責人取得各項資料。以下將就該案例之背景、規劃及格子版結構工程進行描述。該案例與一般高科技廠房之異同點可參考表4.1.1。

這個專案是一個完整的廠房，包含辦公棟(OFFICE)、製造棟(FAB)、機電中心(CUP)三棟建築物，中間以管橋連結，自民國97年1月底土方開挖起算，結構工程於97年12月底完工，歷時約11個月。圖4.1.1為本案例中FAB的結構施工順序，可以看到屋頂結構(4F~屋頂)、內部結構(1F~3F之RC結構體)與格子梁(3F潔淨室區域)結構是採取同步施工的方式，實際上，這個時期也是本專案出工、物料、空間使用最密集的時期。根據廠商施工報表，這段期間平均每日出工數高達1500~2000人日，而光是8月份就進場966噸的鋼筋，平均每天消耗掉超過30噸的鋼筋。在結構工程方面，屋頂結構的鋼結構基礎已完成可獨立施作，但格子梁須以內部1~2F結構為基礎，兩者密不可分。表4.1.2為本案例包商進場時程規劃。

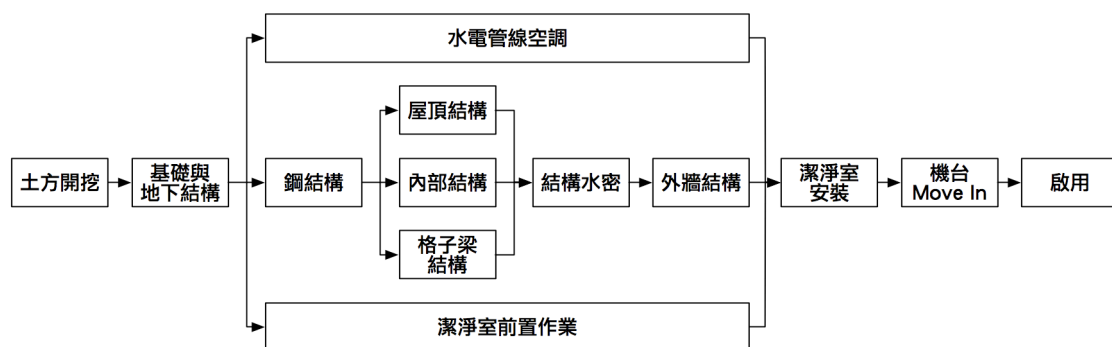


圖4.1.1 案例FAB棟施工順序

表4.1.1 案例廠房特性調查表(與[張書萍, 2000]比較)

類別	項目	本晶圓廠專案	典型半導體廠房	比較	原因
工程專案基本描述	樓層	B1~5F	B1~5F	相同	Sub-Fab設於二樓，地下室設計為系統設備。三樓為主要潔淨室。
	開挖工法	明挖覆蓋	明挖覆蓋	相同	明挖覆蓋法較經濟，適合地質良好者如竹科。
	主結構	RC+SRC	RC+SRC	相同	SRC跨度大且省時，故多數廠房採取此法。
	土建經費	30億	5億~65億	中等	視專案規模而定。
	建築樓地板面積	20萬M ²	~75500M ²	大	視生產或研發而異。
	潔淨室面積	2萬M ²	~10700M ²	大	視生產或研發而異。
	潔淨室等級	V.C.E	視個案	×	視生產或研發而異。
團隊組織	建廠團隊	業主主導	視個案	×	業主主導、規劃設計顧問主導、統包三種。
	營管部門	有	有	相同	
	工安部門	有	少有	較高	本案業主較重視工安。
規劃階段	規劃時間	6個月	60%在6個月以內	相同	專案啟動必須配合企業產能規劃，導致規劃時間較短。
	整合單位	業主	業主	相同	
	土建進廠時機	70%	設計成熟度40%~80%	相對成熟	為了縮短工期必須使用設計施工同步進行的方式，亦導致常有設計變更的情形。

設計階段	設計時間	4個月	57%於3個月以內	稍長	決定執行至實際施工之時間很短，常有設計施工同步的情形。
	設計整合	業主	建築師	不同	本案業主的建廠團隊中有廠務設計部門負責。
	新工法採行	無	無	相同	視成本與工期衡量。
	設計模組化	有	視個案	較多	模組化具有縮短工期及節省成本的優點。
	需求變更	50次	6~15次	較多	此處指廠務需求的變更所導致者，產生原因可能是上級決策或需求的改變，亦可能是設計時未考慮清楚造成。
分標發包	分標策略	專業分包	專業分包	相同	由於各界面設備複雜度高，投資金額大，為了節省經費並落實專業，大多採取專業分包。
	分包數量	建築6包	80%在3標以內	較多	雖多採專業分包，但廠商會合作併標，因此會降低業主分標案總數。
	土建招標	公開招標	邀商比價>公開招標>獨家議價	×	常採邀商比價的方式，獨家議價在業主有良好合作經驗時才考慮。
進度管理	總工期	11個月	10~14個月	相同	配合企業決策，一般要求於10~14個月完工。
	進度管控	每週	雙週	較細	由於工期短，最小管理單位以雙週最多，而每週控管者次之。
人力資源	現場單日最大施工人數	2000人	80%在1000人以內	較高	視工程急迫、專案大小、分標數而定。
	監造人數	50人	10~50人	較多	各專案差異相當大，與業主及承商態度有關。

表4.1.2 案例FAB包商預定進場時間表

FAB進場時程	1	2	3	4	5	6	7	8	9	10	11	12	1	2
土方開挖	█	█	█											
基礎版		█	█	█	█	█								
B2-1FL			█	█	█	█	█							
1-3FL鋼構區				█	█	█	█							
水電管線空調				█	█	█	█	█	█	█	█	█		
中央柱SRC						█	█	█	█					
3-RF鋼構安裝						█	█	█						
1-3FL非鋼構區							█	█	█	█				
EPOXY塗漆								█	█	█	█	█	█	
潔淨室安裝								█	█	█	█	█	█	
機台Move In													█	█
取得使用執照														█

首先從表4.1.1中，本個案與其他高科技廠房之比較可以發現，本工程與其他高科技廠房專案不同處在於工程規模較大，每日總出工人次最高峰時高達2000人，其餘屬性則大致與平均值相同。觀察包商進場時間表(表4.1.2)，1-3FL非鋼構區給定工期僅3個月，配合圖4.1.1可以發現該區位於要徑節點上，必須更精準確實否則會造成後續工程與機台安裝無法進行。

4.2 格子版結構工序分析

本案例之FAB棟結構體工程截至論文撰稿當時已施工完畢，本研究取得案例七月至九月的施工日報表，並實地調查工率與物料安排情形。同時透過對業主、營造廠、建築師、鋼筋包商、模板包商、工人等進行工程實況的訪查，了解工程進行之困難與相關因素。

本研究之目標為FAB中的場鑄格子版(Waffle Slab, 俗稱洞洞版)結構工程，此類型結構為高科技廠房潔淨室獨有之結構體，工法上分為預鑄及場鑄兩種，依據業主需求選擇。根據訪談的結果，除本案例由於成本考量選擇場鑄工法外，過去的高科技廠房興建亦以場鑄居多。為了更進一步調查格子版施工的影響因素，本研究對四位主管級工程師(現場工程師兩位、設計、監造各一位，平均經驗14年)進行訪談，整理出影響格子版施作工率的主要影響因素如表4.2.1。

單格鋼筋與FRP模組裝完成圖可參考圖4.2.1，圖4.2.2為營造廠提供之格子版施工作業流程(實際施工照片4.2.3a~d)。初期規劃所排定之格子版工程主要工項為鋼筋綁紮(3~4d)、FRP模組立(2d)和灌漿(1d)這三個項目。廠商亦針對鋼筋綁紮及FRP組立兩項主要作業之工率影響要素進行說明。其中鋼筋部份以運輸距離及施作品質為主因，前者為物料可得性之要素，後者之不良會導致重作之浪費；FRP模組部份則以運輸動線、堆置區不足為主因。

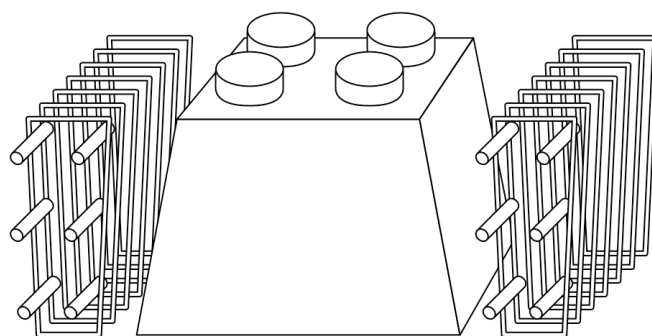


圖4.2.1 格子版單格簡圖

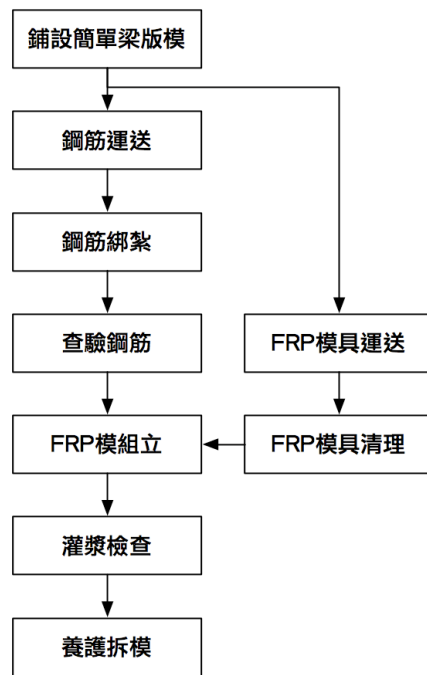


圖4.2.2 格子版施工作業流程



(a) 格子版鋼筋綁紮



(b) FRP模具組裝



(c) 灌漿前檢查並安裝FRP頂板



(d) 養護完畢拆模

圖4.2.3a~d 格子版結構場鑄施工流程

表4.2.1 格子版工率影響因素調查

分類	影響因素
人力	工法不熟悉、出工數不足
品管	放樣不準確、鋼筋缺失過多、監工人數不足
排程	階梯狀施工安排不當導致施工面落差太大
場地	施作區域擁擠、FRP搬運路徑過遠、物料暫存區太小、動線不佳
物料	送料機具不足、FRP模不足、鋼筋不足

歸納上表中的因素，空間、工人、物料之不足與搬運動線為影響場鑄格子版工率的主要因素。徵詢包商與工人意見時，亦提到搬運路徑和暫存區不足影響甚大。過往針對鋼筋及模板工率的建模研究較重視天氣、意外事故、擁擠、工作難度等等因素，施工動線時常被提起，但鮮少被納入模式中。本案例之場鑄格子版工程位於3F，屬於室內工程，不受天氣影響，因此研判本案例之工率影響因素以施作空間與物料之取得為主。

4.3 案例資源配置方式分析

格子版結構以柱位作為最小施工單位，每個柱位為4.8M × 4.8M，由井字形之梁鋼筋構成，隔出4x4 = 16單位的格子，並放置16個FRP模具，鋼筋約需3噸。本案例之格子版寬18柱位、長31柱位，共558柱位(由於有SRC柱等障礙物，約為8900格)。為了更貼近真實施工情形，本研究採用各階段灌漿之區域為分區依據，這是因為實際工程會考量現場動線及物料安排，使工區劃分與初期規劃不同。各區之鋼筋與FRP施作範圍比灌漿範圍略大，但考量數據處理問題，簡化為灌漿區域大小。

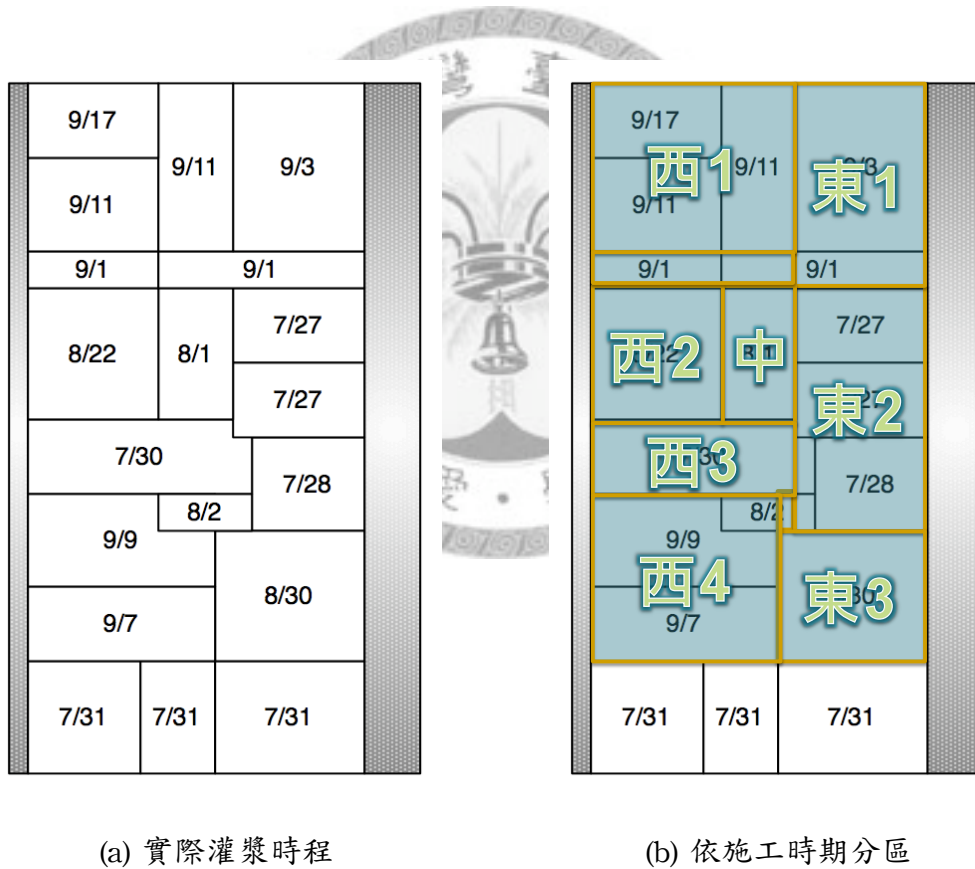


圖4.3.1a~b 灌漿時程及分區圖

經過實地訪查，物料擺放區域有三個原則：離作業越近越好、離進料口越近越好、不與作業區及動線重疊。另外，這個案例中物料之擺放可分為三個時期，初期多數樓板未完成，物料擺放區域相當有限；中期施作空間較大，初期區域已完成可供擺放；晚期受限於初期完成之格子版已開始潔淨室安裝工程，需避免放置於潔淨室施工中的區域。三個時期物料擺放如下：

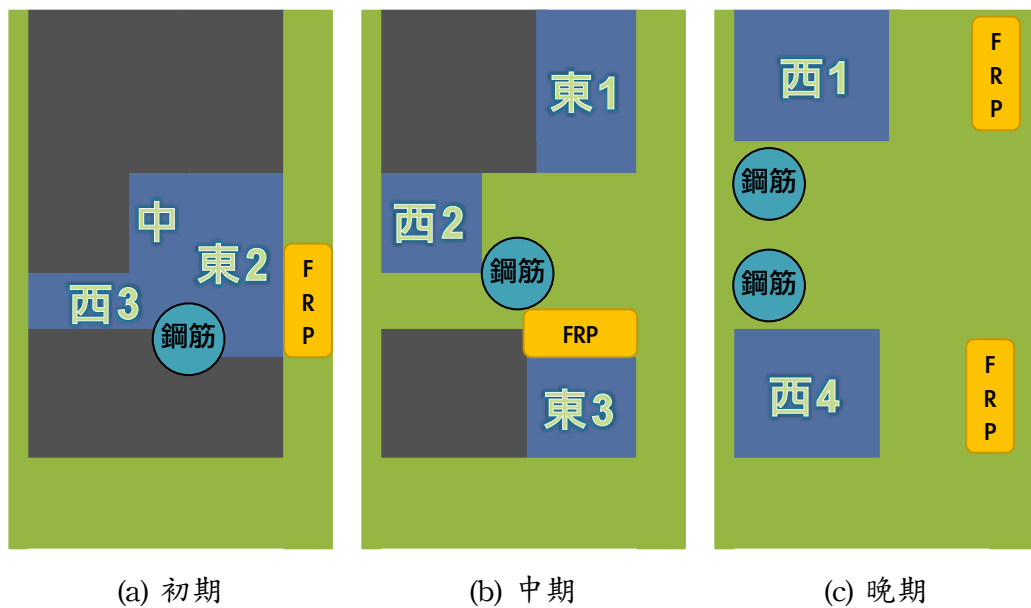


圖4.3.2a~c 案例各時期物料配置簡圖

本研究中將物料搬運距離統一為暫存區與工區中心之X截距加上Y截距，不取直線距離是由於工人需於格子版鋼筋上行走，跨越格子時，因斜向距離過長不易立足，採縱向或橫向跨越較安全且快速。運距計算式如下(D—運距，X—X截距，Y—Y截距，m—工區代號，s—暫存區代號)：

$$D_{ms} = |X_m - X_s| + |Y_m - Y_s| \quad (1)$$

其次由施工日報表中計算每日分區出工數，因施工報表多達90頁，故僅列出計算方式與結果。本工程的施工報表在出工數計算方面以包商作為劃分，即計算該包商當日出工數並進行累計以利計價。另外亦會紀錄當日氣候、進料、並簡

略描述當日工作區域，然而工作區域之描述時有缺漏，該部份配合施工進度表進行修正。

受限於日報表僅紀錄包商出工情形，並未對各工區出工人數作更詳細的紀錄，本研究在訪談工地主管了解工人分佈情形及實地觀察，在避免擁擠之前提下，認為分區出工人數估算可依據各區域比例劃分。每日出工數計算式如下(U—出工數，A—工區面積，m—分區代號，n—工程第n天)：

$$U_{mn} = U \times A_m / A \quad (2)$$

透過(1)、(2)得到每日分區出工情形之後，便可進行該區工率與總出工數之計算。針對鋼筋及FRP模組裝計算各區工量、運距、平均出工數、總出工數、工率之關係如表4.3.1及表4.3.2：

表4.3.1 鋼筋工率統計表(區域依施作先後排序)

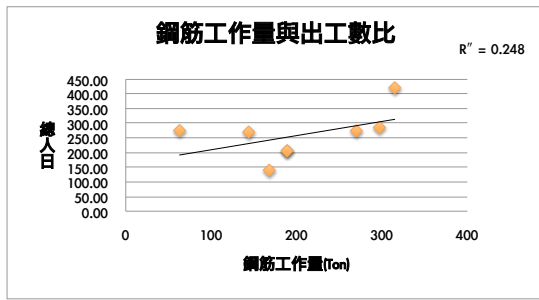
區域	(1) 總格數	(2) 運距	(3) 工量(噸) (1)×3	(4) 平均 出工 (人日)	(5) 工期(天)	(6) 總出工 (人日) (4)×(5)	(7) 工率 (噸/人日) (3)/(6)
東2	105	55.2	315	35.15	12	421.82	0.7468
中	21	43.2	63	39.37	7	275.62	0.2286
西3	48	43.2	144	33.84	8	270.7	0.5319
西2	63	26.3	189	34.4	6	206.4	0.9157
東1	63	26.4	189	25.81	8	206.47	0.9154
西1	99	48	297	26.02	11	286.22	1.038
東3	56	16.8	168	23.37	6	140.21	1.198
西4	90	21.6	270	27.47	10	274.7	0.9829

表4.3.2 FRP模組工率統計表

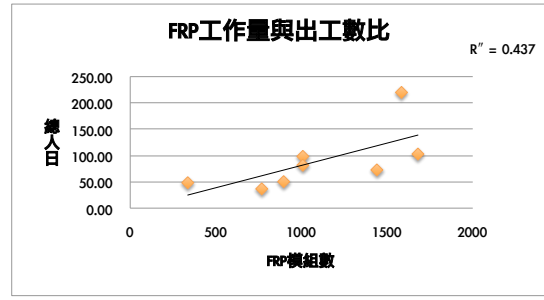
區域	(1) 總格數	(2) 運距	(3) 工量(個) (1)×16	(4) 平均 出工 (人日)	(5) 工期(天)	(6) 總出工 (人日) (4)×(5)	(7) 工率 (個/人日) (3)/(6)
東2	105	48	1680	14.69	7	102.83	16.34
中	21	76.8	336	24.17	2	48.33	6.95
西3	48	67.2	768	12.22	3	36.67	20.94
西2	63	69.6	1008	24.6	4	98.4	10.24
東1	63	38.4	1008	27.17	3	81.5	12.37
西1	99	64.8	1584	24.44	9	220	7.2
東3	56	45.6	896	25.2	3	50.4	17.78
西4	90	84	1440	18.17	4	72.67	19.81

接下來將就資源配置方式對工率的相關性進行討論。首先，傳統的計算方式上，所需出工數與工作量成正比，從本案例分析的結果可以發現(圖4.3.3a~f)，儘管工作量越高出工數亦越高，其線性仍相當不規則，趨勢難以預料。在上一節的專家訪談中提到，空間、出工數、物料補給、搬運動線對於工率有相當大的影響，進一步觀察工率與每日平均出工數、總工作量、取料距離之間的關係，發現確實出現下面趨勢：每日平均出工數越高，工率越低；取料距離越遠，工率越低。從這部份的觀察了解到，每日平均出工數與物料搬運距離這兩個因素確實與工率有關。

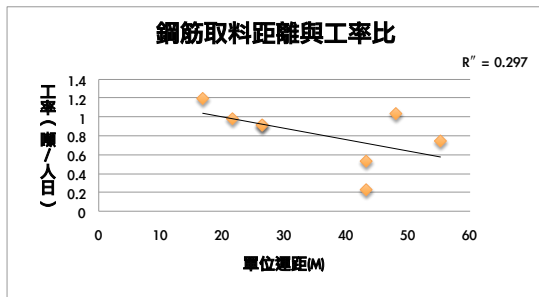
就這兩個因素而言，平均出工數越高代表資源消耗速度越快，容易發生物料運補不足，導致工率下降的情形；物料距離越遠則會造成運補時間拉長，亦會造成工率下降的情形。這兩個因素並非互相獨立，因為前者導致工率下降的情形是限定於物料運補不足時，後者則限定於資源消耗速度太快的情形下，故無法單純以線性表示因素與工率關係。兩者皆列入下一章節輸入派翠網之資源變因。



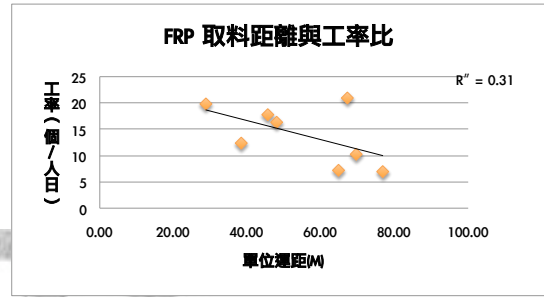
(a)



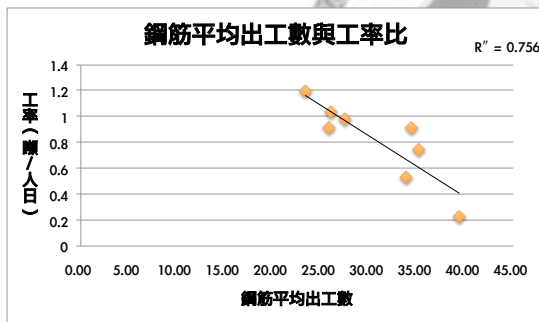
(b)



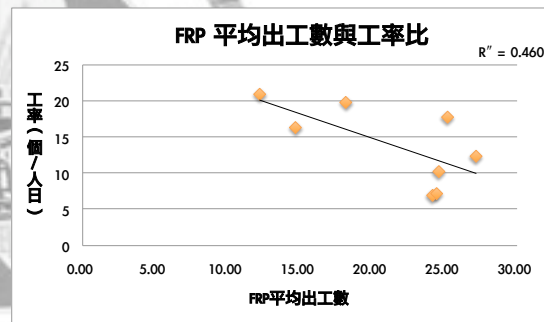
(c)



(d)



(e)



(f)

圖4.3.3a~f 資源配置與總出工數、工率關係圖

從上述分析可觀察出，資源可得性在本案例中對工率的影響相當直接，資源的配置方式如：每日平均出工數、取料距離、暫存區空間大小皆會限制資源可得性，導致工作環節產生等待，工率無法最大化。因此，若要提昇工率，縮短工期，應從調整資源配置比例著手。

第五章 以派翠網分析資源對工率之影響

5.1 建置階層式時間派翠網圖

建立派翠網分析模型的第一步為繪製正確的格子版網圖，繪製流程如下：

1. 將工序與工作資源連結，補齊未列於工序中的資源取得作業
2. 配置各工項(變遷)之發生時間(有固定、與資源相關兩種計算方式)
3. 配置各資源(庫所)之容量(Capacity)
4. 給予初始資源，進行派翠網模擬測試
5. 若測試於未完工前產生鎖死(Deadlock)，回到步驟1重新檢查網圖邏輯
6. 完成網圖

本網圖採用階層式時間派翠網的方式繪製，因此每個變遷皆需經過一段時間進行，為了更精確表現資源對工項時間長短之影響，本研究將時間長度計算分為兩類，一種為固定時間長度，另一種則根據下面公式：

$$\text{資源需消耗數量}/(\text{資源使用者數量}\times\text{工率}) \quad (3)$$

舉例來說鋼筋綁紮時間為： $(\text{鋼筋施作數量})/(\text{鋼筋工每日出工數}\times\text{工率})$ 。接著必須進行階層化，將過於複雜的網圖收斂，由於鋼筋綁紮與FRP模組立兩者可視為獨立，因此將這兩者分別進行階層化收斂。最後替同一類型之變遷與庫所著上相同顏色以利辨識，綠色—工作設計規劃、灰色—鋼筋綁紮、粉紅色—施作空間準備、橘色—FRP模組立、淺藍色—混凝土灌漿，本步驟與著色派翠網概念不同，後者替令牌著色以辨識資源並收斂網圖。由本案例繪製完成之格子版派翠網圖如圖5.1.1，鋼筋綁紮與FRP模組立之網圖分別為圖5.1.2及圖5.1.3。

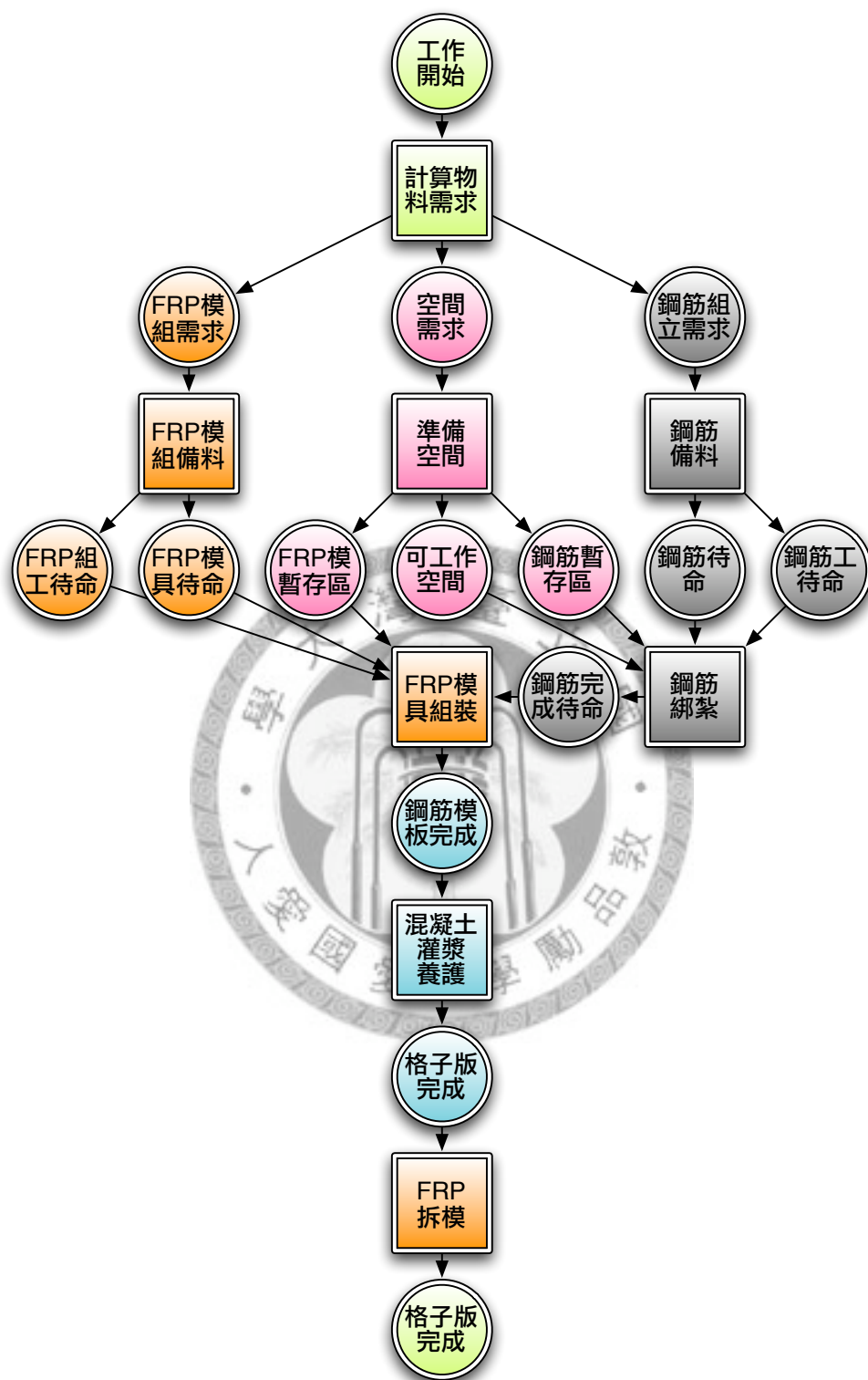


圖5.1.1 格子版派翠網圖

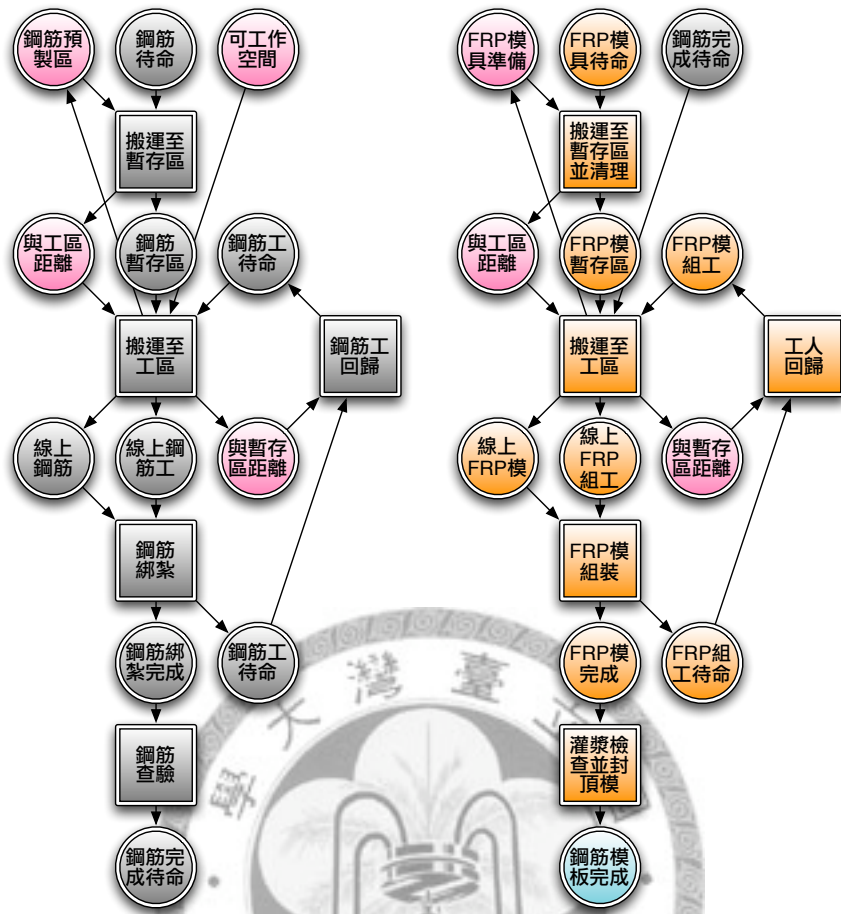


圖5.1.2

圖5.1.3

圖5.1.2 鋼筋綁紮次階網圖

圖5.1.3 FRP模組立次階網圖

在整個格子版結構工程中，鋼筋綁紮與FRP模組立是最費時、消耗人力物力最多的工項，因此後續分析重點為這兩個次階網圖對資源配置方式影響工率的表現能力。從實際完成所花費的工期來看，鋼筋綁紮與FRP模組立分別需要6~12天及3~9天的工期，工期長且變動性大不易估算，所以這兩個部份將根據資源配置方式進行派翠網模擬，除此之外如工廠備料、工作空間準備、灌漿等工期之變動性較低，少則半天多則一天，故採行固定工期。另外，由於本案例位於室內，除颱風停工外，受氣候影響相當小，這段期間聘僱之工班亦相同，工人素質因素受到控制，可排除工率隨機變動的情形，只需考慮工率受資源可得性的影響。

繪製完成之網圖皆以本節一開始所述之步驟進行網圖正確性之檢核，確認網圖皆按照排定之流程達成預計工作數量，從前一章案例分析中，我們了解到每日平均出工數、取料距離這兩個因素對於工作資源可得性有最直接的影響，因此將這兩個資源表現於網圖中工人施作與取料的循環中。

在本研究中，變遷發生時間(各工作耗時)以資源數量與工率進行計算，且需於第二部份的研究中，加入蟻群最佳化演算法進行智慧型資源配置，在針對市面上較常運用於研究領域之的派翠網建模軟體如Netlab、TINA、Snoopy、WoPed等等進行搜尋後，發現沒有符合需求之軟體，因此本研究以MATLAB軟體(版本為R2007b)進行程式撰寫，命名為PetriStar，其具備以下功能：(1)派翠網圖模擬、(2)動畫模擬、(3)計算績效指標並繪圖、(4)可用蟻群最佳化演算法進行派翠網資源調配。PetriStar介面如下圖，程式碼與功能按鈕可參考附錄。

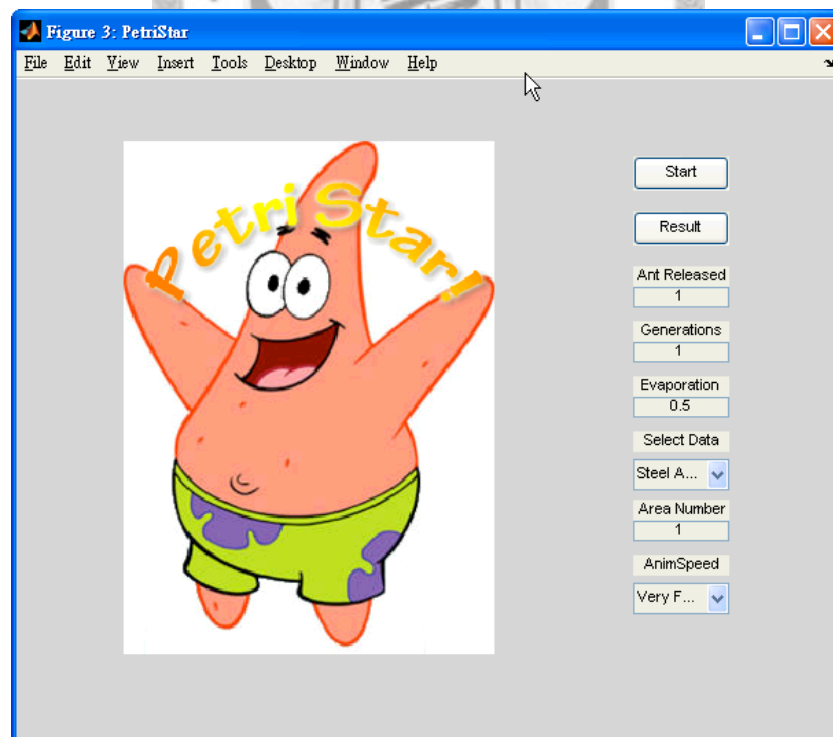


圖5.1.4 PetriStar介面

5.2 輸入資源配置進行派翠網模擬推估

在過去於營建領域的研究中，派翠網模擬結果的比較對象皆限制於初期規劃之排程，未曾對真實案例的施工程果進行檢驗，亦未針對相同工項進行不同資源配置下的排程變動性。本研究著眼於派翠網詳細描述的能力，深入探討營建工程的工程變動性產生原因。過往的排程上的研究，多半以一層樓為一個製造單位來導入製造業的管理手法，但實際上一個樓層內仍包含多個製造單元，如梁鋼筋、梁模板、柱鋼筋、柱模板、混凝土...等等構件，這些構件不盡相同，貿然將其視為同一個單元進行分析，容易產生過高的變動性以致管理不易。因此，本研究認為，若要獲得更精準的工作估時，必須將工程各環節皆詳細納入，並考量資源可得性的問題，因此採用派翠網針對最低階的工作進行估算，並與真實工作結果進行比較。

資源配置比例不當會降低資源可得性，導致工作流程接續不順，本論文旨在研究派翠網是否能夠反映真實施工時，資源配置比例不當造成對於工率的影響。首先需要獲得最佳工作情形下的工率，本研究於現地調查密集工作時段(早上9:00~11:00、下午14:00~16:00)之工率，該時段物料補給和出工人數皆相當穩定，資源可得性最大化，發現皆高於整體的工率計算結果。調查結果如下表：

表5.2.1 資源可得性最大化工率調查表

工項	調查結果	單位
鋼筋綁紮	1.27	噸/人日
FRP模組立	14.4	個/人日
鋼筋取料	0.5	公尺/秒
FRP模取料	0.7	公尺/秒

參照鋼筋綁紮與FRP模組立的次網圖，上表所調查之工率會對以下三種類
型工作有影響：搬運至工區、組立工作、工人回歸。這三種類工作受限於工率與
資源配置方式，可參考5.1的工作時間計算公式(3)。

下一步要將案例中各區的資源數量配置方式(庫所容量及令牌所代表的大
小，參照表4.3.1&表4.3.2)，以及各工項(變遷)之工率，分別輸入鋼筋綁紮及FRP模
組立兩個派翠網圖，進行模擬計算總循環時間。派翠網計算時間的概念可用下面
這個循環圖表示：

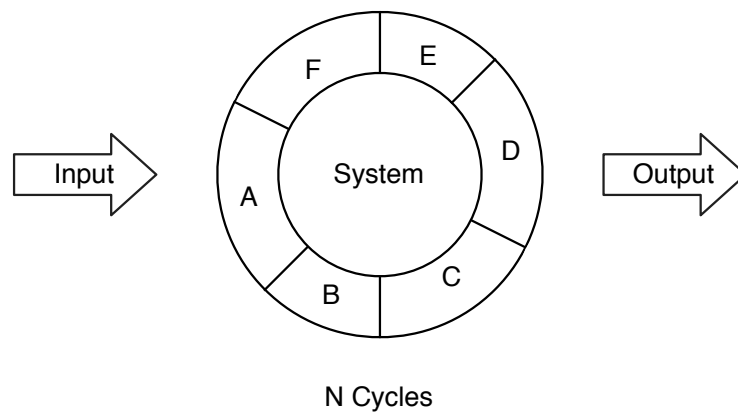


圖5.2.1 系統循環圖

1. 時間經過片刻(Tick)
2. 檢查所有變遷是否啟動(Enabled)
3. 若已啟動之變遷符合發生條件，則計算其累計完成進度
4. 檢查所有變遷累計完成進度，大於或等於100%時視為完成
5. 變遷完成者，所有輸入庫所消耗一個令牌，所有的輸出庫所產生一個令牌
6. 若該片刻無任何變遷可呈現啟動狀態，流程停止
7. 判斷完成數量是否符合本次模擬預期，若不足需重新檢驗網圖邏輯，正確者則可進行時間累計及結果分析

上述步驟中，片刻時間的長度選擇是相當重要的。片刻越長，計算需時越少，片刻越短則反之。另外，由於令牌的更新於該片刻結束時進行，當片刻時間過長，會發生工作完成時僅該片刻前段，時間卻需計為整段片刻的長度的情形。這類型的誤差會在片刻越長的時候越明顯，舉例來說，一個片刻設為一小時的情況下，若工作在某個片刻只剩下10分鐘的工作量，仍需計為一小時，而產生50分鐘的工期損失。為了找到最適合本案例的模擬片刻長度，本研究在模擬前，以FRP模組立之網圖進行測試取得片刻長度和計算時間的關係，結果如下表：

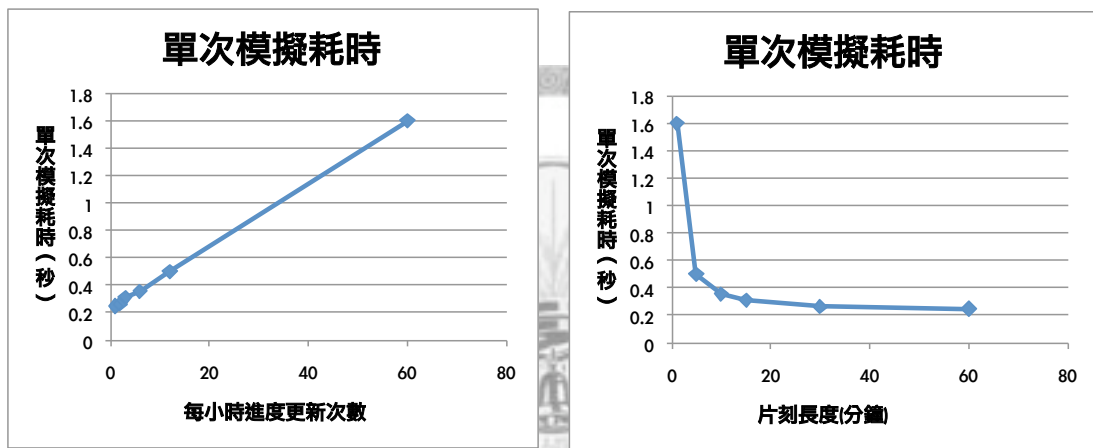


圖5.2.2

圖5.2.3

圖5.2.2 每小時進度更新次數與計算時間關係圖

圖5.2.3 片刻長度與計算時間關係圖

從圖5.2.2可以發現派翠網的模擬計算需時與單位時間內進度更新次數成正比。將單位時間內進度更新次數換算為片刻長度可得到圖5.2.3。觀察發現，片刻時間小於10分鐘時計算量會急遽增加，因此選擇10分鐘為派翠網模擬之片刻時間單位。

5.3 模擬結果分析

本研究的模擬測試環境如表5.3.1，以MATLAB R2007b建立PetriStar模擬平台，變遷發生時間(Transition Firing Time)以資源數量與工率計算，且能運用蟻群最佳算演算法進行資源數量最佳化為其特色。由於PetriStar為針對本研究開發者，故軟體自由度較低，需另外配合Notepad或Excel進行輸入資料之編輯。

表5.3.1 模擬測試環境

項目	敘述
作業系統	Windows XP SP3
CPU	AMD Athlon(tm) 64 X2 Dual Core Processor 5200+, MMX,3DNow (2 CPUs), ~2.7GHz
RAM	3326MB (800Hz,DDRII)
派翠網模擬平台	PetriStar(由MATLAB R2007b開發)
蟻群演算法平台	PetriStar(由MATLAB R2007b開發)

受限於資料取得，現地資訊不足者採以下假設：

1. 同一工區資源配置數量穩定。
2. 由於格子版工程位於室內，故不考慮天氣對工率的影響，且不考慮工安。
3. 受限於資料取得，暫存區大小不納入考慮。

接下來便根據上節的步驟對每個工區進行派翠網模擬後，得到相同的工作資源配置下，派翠網模擬推估結果與實作結果比較圖表如後續圖表：

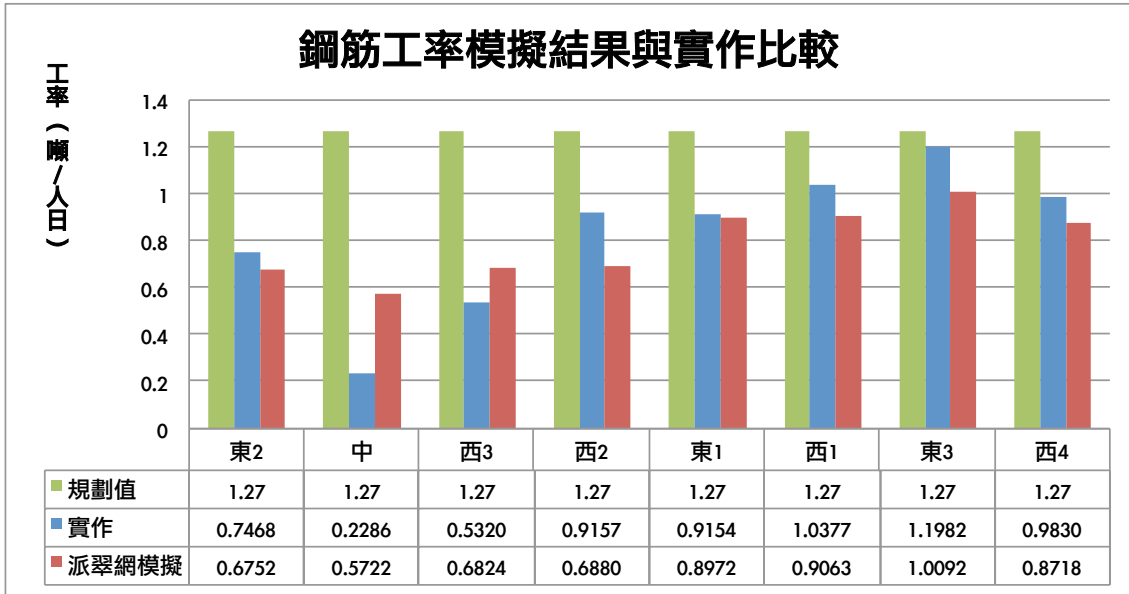


圖5.3.1 鋼筋工率模擬結果與實作比較圖

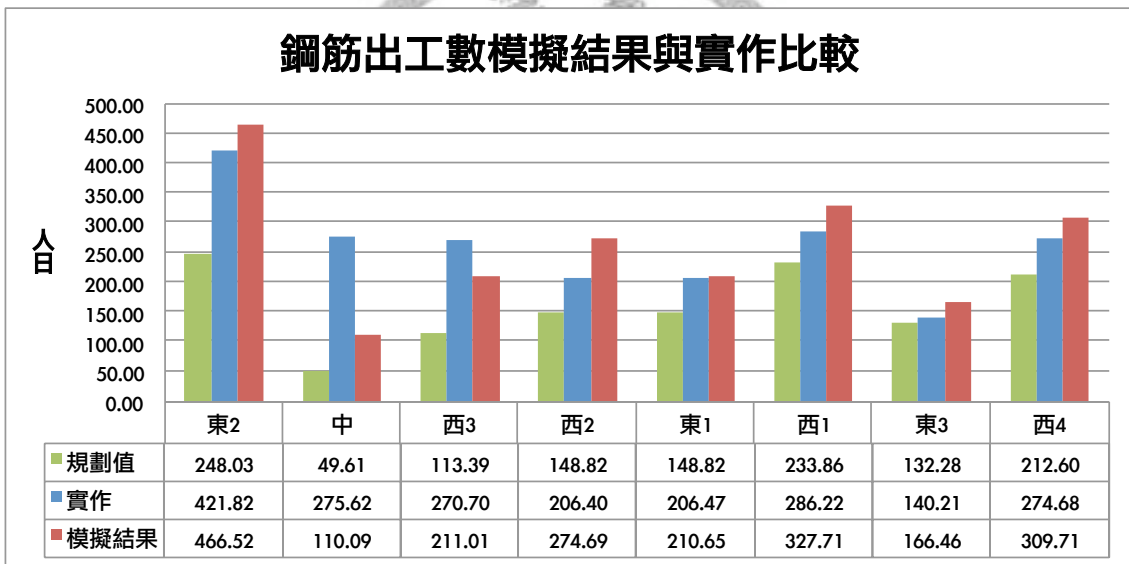


圖5.3.2 鋼筋出工數模擬結果與實作比較圖

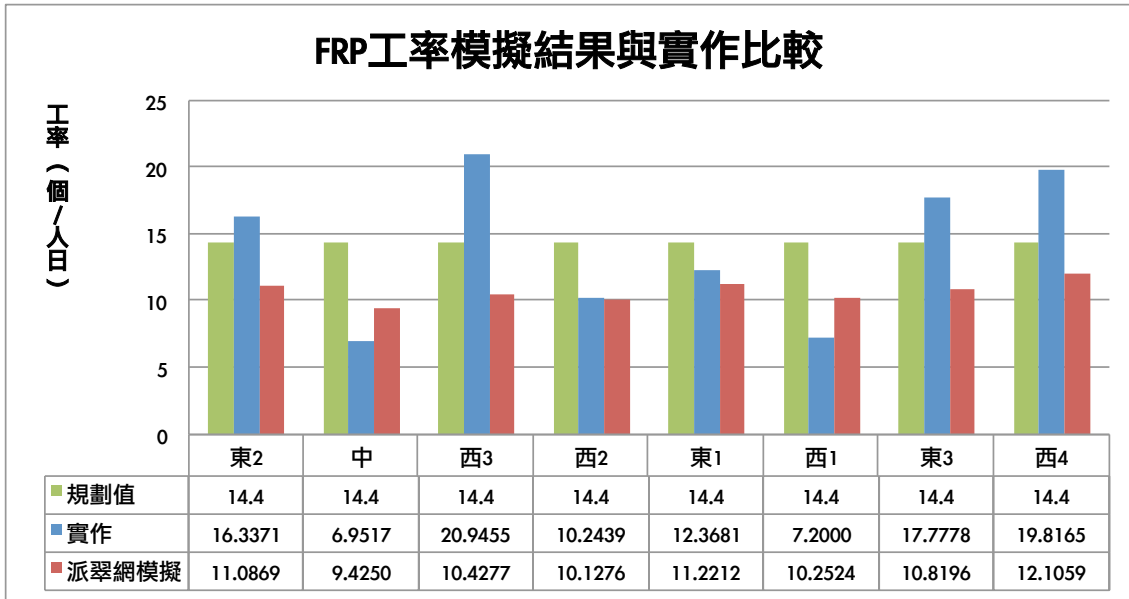


圖5.3.3 FRP工率模擬結果與實作比較圖

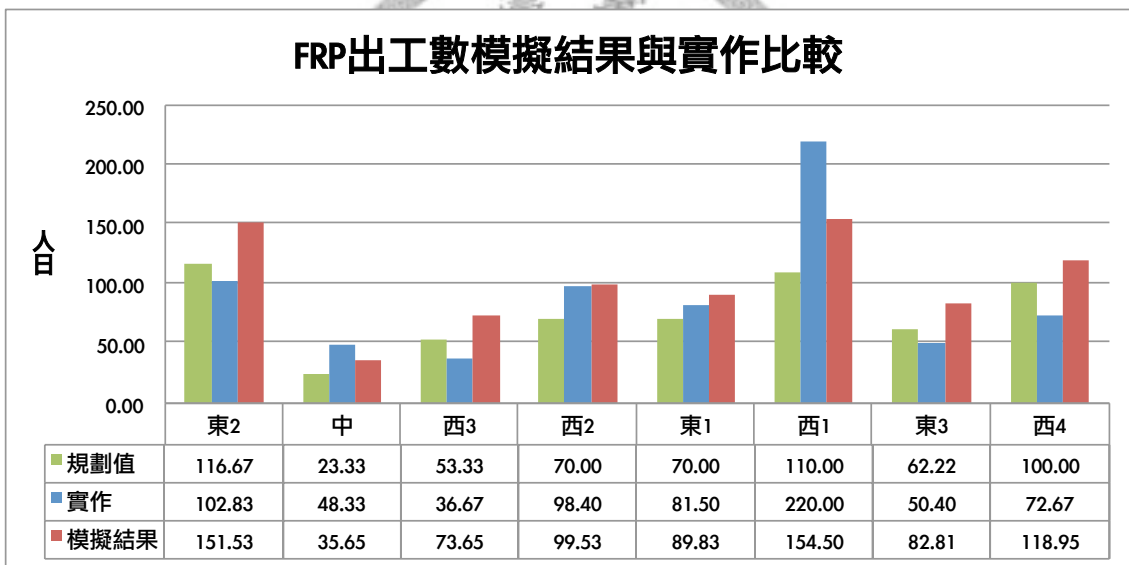


圖5.3.4 FRP出工數模擬結果與實作比較圖

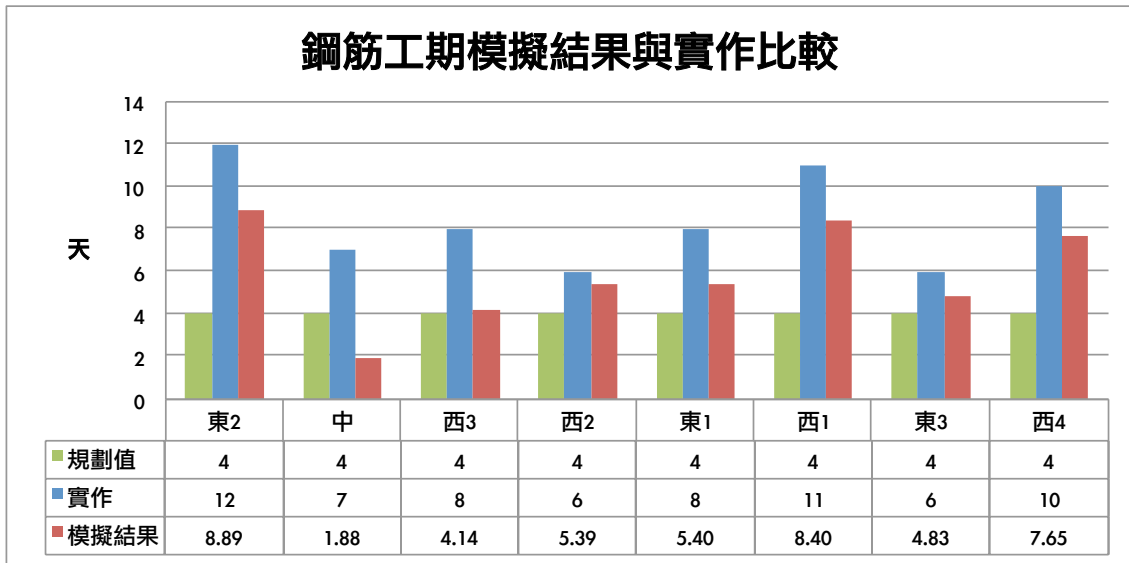


圖5.3.5 鋼筋工期模擬結果與實作比較圖

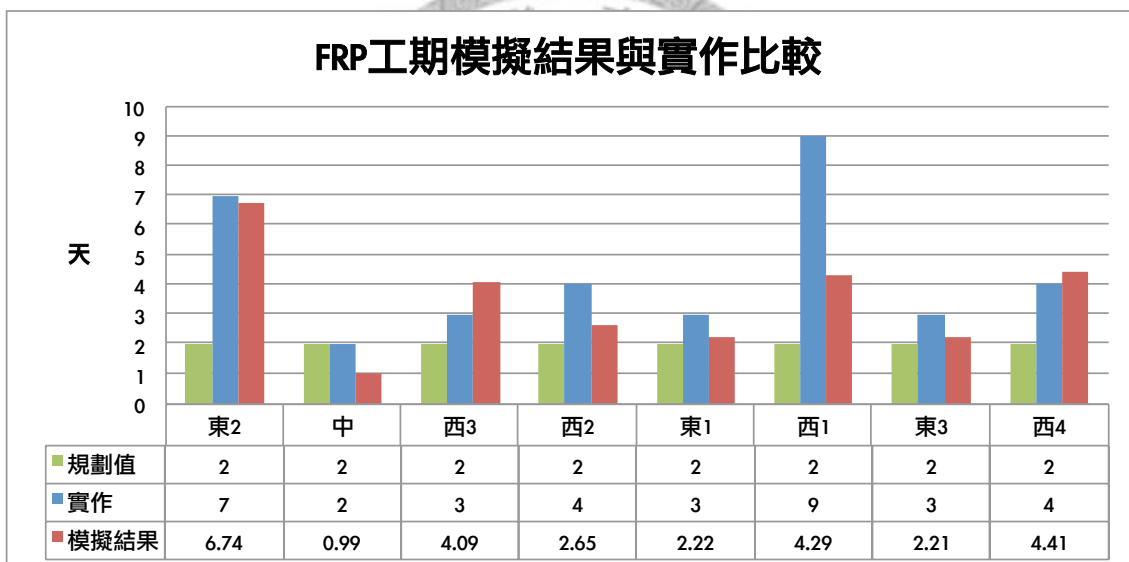


圖5.3.6 鋼筋工期模擬結果與實作比較圖

以上六個圖表從三方面進行估算與實際值的比較，一為派翠網推估之工率與真實工率之比較，這部份為結構分包商績效指標；另一方面為派翠網推估之出工數與真實出工數之比較，此為出工成本績效指標；最後為派翠網推估之工期與真實工期之比較，這部份是高科技廠房新建專案中，業主最重視的指標。

從上面的圖表可以發現，實際執行的績效與理論值差異甚大，在工程規劃設計期，往往套用理論工率及理論工期進行排程，出工數則套用理論工率與施作面積進行計算，忽略現場工作變因，造成工率高估。到了工作執行時期，才發現

估算錯誤，產生工期延宕、出工數不足等等窘境。過往對於工期的變異性採用隨機分佈模型進行估算，僅能了解該工項可能的變異範圍，仍無法確實告知管理人員該工程實際需時多長、需出多少工，導致估算結果不被現場工程師信任。

觀察派翠網模擬結果與實際執行的數值，可以發現兩者具有高連動性，即工率相對較低的區域，派翠網模擬出來的結果亦相對較低，反之亦然。因此在估計的誤差方面亦較單一工率估算的精準度大幅提昇。這部份的連動性在本研究中採相關係數(Correlation)作為評估指標，並配合平均誤差進行分析討論。評估比較的結果如表5.3.2及表5.3.3：

表5.3.2 派翠網模擬結果與實作之相關係數表

與實作值之相關係數	派翠網	理論值
鋼筋工率	0.878	0.000
鋼筋出工數	0.721	0.499
鋼筋工期	0.812	0.000
FRP工率	0.665	0.000
FRP出工數	0.754	0.662
FRP工期	0.694	0.000

表5.3.3 派翠網模擬結果與實作之誤差表

推估值與真值誤差	派翠網	規劃值	改善幅度
鋼筋工率	31.85%	99.93%	68.13%
鋼筋總出工數	21.72%	35.46%	38.74%
鋼筋工期	32.10%	112.50%	71.47%
FRP工率	31.10%	44.20%	29.64%
FRP總出工數	42.95%	33.08%	-29.82%
FRP工期	29.91%	118.75%	74.82%

首先就相關係數進行討論，規劃值為廠商規劃設計時的參考數值，分為工率、出工數、工期三個部份。工率是出工數的計算依據，總出工數計算式為(工量/工率)，工期則是(總出工數/每日出工數)。配合圖5.3.1~6，可以觀察出派翠網模擬推估之數值會隨著實作值變化，具有相同的趨勢。派翠網與實作值兩曲線之間的同步變動情形可以用相關係數進行評估，評估結果如表5.3.2(0：無任何相關，0.5：中度正相關，1.0：高度正相關)。特別在鋼筋綁紮部份獲得相當良好的相關性，FRP模組立的派翠網推估與實作值之相關性雖不如鋼筋綁紮，但仍獲得高相關性的推估成效。反觀理論值，雖運用於廠商規劃設計之參考，但未考慮到資源數量配置對工率、工期之影響，因此在這兩部份無相關性可言，僅出工數因受到實際工量影響，與實作值仍有中度相關。

除了相關性評估之外亦須考量誤差，誤差計算式如下：

$$|\text{估算值} - \text{實作值}| / \text{實作值} \times 100\% \quad (4)$$

上式能反映估算與真值的差異幅度，缺點為當實作值較小時，此值會較大。表5.3.3為派翠網估計成果與理論值之比較，可以發現相較於理論值，派翠網估計有相當大幅度的改善(平均改善幅度42.16%)。下面將就幾個方面討論實驗誤差之產生可能：

1. 本模式之派翠網模擬假設工程進行時資源固定，但實作時仍有變動。
2. 工地施工意外會導致短暫停工，實作工率仍受到工安風險影響。
3. 本模擬受限於資料取得，未將暫存區大小納入考慮，但實務上暫存區大小對物料補給的流暢度有很大的影響。
4. 配合工率統計表(表4.2.1、表4.2.2)中的資源配置情形，可以發現估計誤差最大的中區與西3區，這兩區工量最小，且鋼筋工率過低。根據施工時期之實地觀察，這兩區的鋼筋工時常被調派至其他工區，這類調派未反映在施工日報表中，而導致出工數高估、工率低估的情況，此類型誤差應視為施工日報表紀錄之缺漏導致。
5. 本研究繪製之FRP模組立網圖偏向將FRP模搬運工和組裝工合併為一群，流程為「取料→組裝→回歸」，但實際FRP模組立流程中，會先將一部分的FRP模搬運至格子放置後，再開始上述流程。實驗結果發現FRP模組立之模擬結果不如鋼筋綁紮理想，應為網圖不夠貼近實際施工流程導致。

過往於營建工程領域的派翠網研究中，僅對派翠網的建模、分析與模擬能力進行檢驗，在本研究中則進一步探討，派翠網對於不同資源配置之工程，其績效推估能力。綜合趨勢變動、相關係數與誤差分析的結果，本研究認為，當資源配置情況不同時，派翠網的估算結果比傳統CPM網圖更貼近實際工程績效。

5.4 應用派翠網於資源組合分析

由前一個章節驗證派翠網能反映資源配置方式對施工績效之影響，並了解到派翠網能相當快速的求出任一種資源組合下的工程績效。因此，接下來將嘗試以派翠網進行窮舉法，將各種排列組合繪製成工程資源配置與績效之關係曲面圖，供管理者進行資源配置之參考。圖5.4.1為固定工量下，不同取料距離和每日出工數對應之工率。圖5.4.2則為固定取料距離，不同工區大小和每日出工數對應至所需的總出工數，可協助管理者進行成本分析。

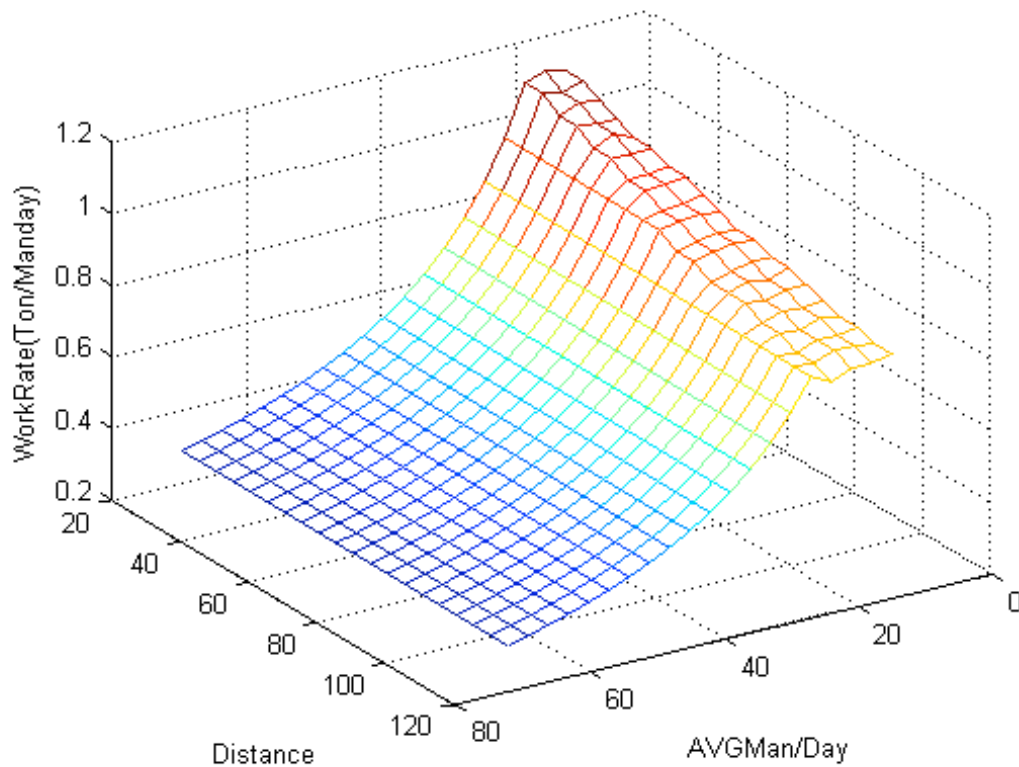


圖5.4.1 取料距離與每日出工數對工率關係圖

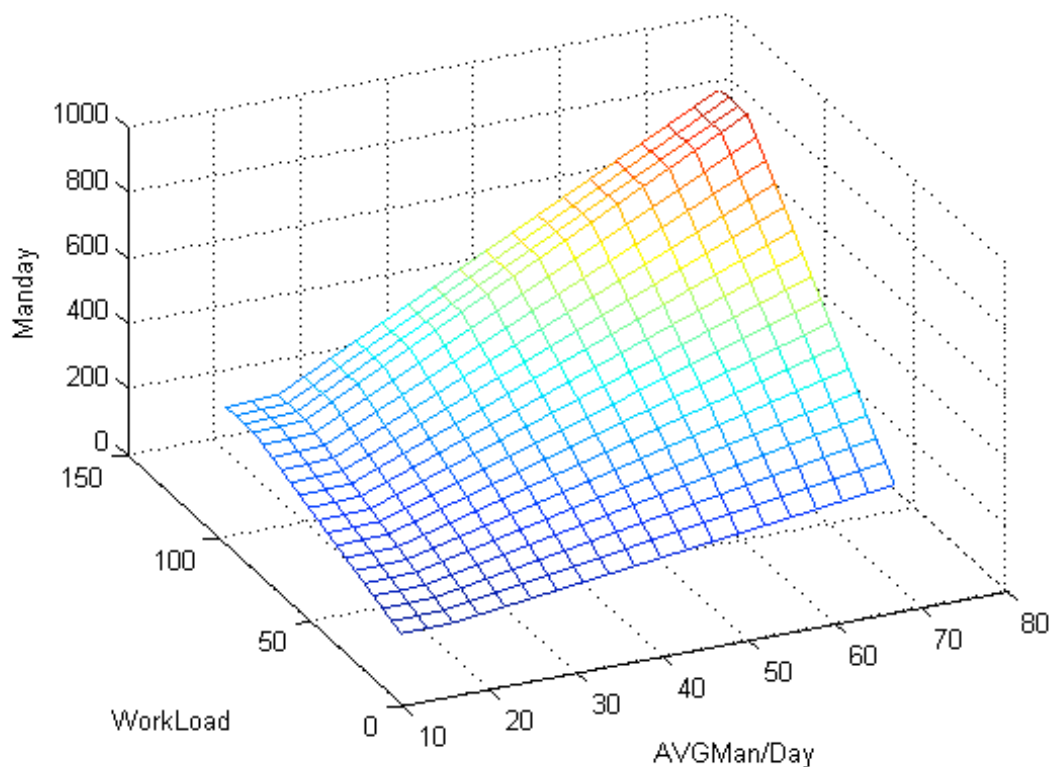


圖5.4.2 工區大小與每日出工數對總出工需求圖

上面兩個例子是以鋼筋綁紮之網圖，各以 20×20 種排列組合進行400次模擬，計算時間約300秒。藉由大量的派翠網模擬，能在很短的時間內，繪製任兩種資源變數及對應之績效曲面區，協助規劃者進行決策分析，缺點是若資源變數達三種以上，其排列組合數量會呈現指數成長，假設三種資源各有20種可能，模擬次數即高達8000次，計算時間約需6000秒(100分鐘)，在規劃時間分秒必爭的高科技廠房專案中，使用窮舉法顯得不盡理想。因此下一章將運用人工智慧，加速找出最佳的資源數量配置方式。

第六章 營建資源配置組合最佳化

6.1 基於派翠網之蟻群最佳化演算法模式建置

從上一章對派翠網的探討及實證，了解到其反映真實施作之能力，基於此網圖能快速求取具可信度之最佳資源組合。然而一旦輸入變數增加，採窮舉法進行最佳組合搜尋所需之時間仍長，無法配合高科技廠房規劃設計時間短的特性。本研究運用自動催化(Autocatalytic)型的人工智慧演算法，進行最佳解之搜尋。從文獻回顧中了解到，蟻群最佳化演算法(Ant Colony Optimization，以下簡稱ACO)在處理組合最佳化問題(Combinational Optimization Problem)的成果相當卓越，因此本研究將採用ACO進行資源數量配置的最佳化。參考前一章的模擬成果比較，選擇模擬成果較佳的鋼筋綁紮網圖作為基準網圖。各項參數之設置法則參考3.2.2。首先建立合法解答，相關資源項目、配置原則及相關工項如下表：

表6.1.1 鋼筋綁紮資源配置原則與相關工作

資源名稱	配置原則	相關工作	合理範圍
暫存區容量	不可配置於動線及工區，僅能選擇閒置之完成區域	搬運至暫存區	1~20組柱位
每日出工數	個人必須擁有最小工作空間(一個格子)，且需預留部份格子作為工區內動線。	鋼筋綁紮、 鋼筋取料	5~55人 (參考案例每日出工極值)
工區面積	灌漿面積太大會造成施工性不佳，及灌漿縫產生。	影響需完成之 工作循環數	20~105 (參考實際規劃)
取料距離	與鋼筋暫存區設置位置遠近及動線規劃有關。	工人取料、 工人回歸	10~85M (潔淨室寬度)

在運用ACO求解資源配置數量組合時，需依照上表之配置原則，以避免無用的解產生，例如工區面積32格，卻配置每天出35位工人便顯得相當不合理。確認合法解答的範圍後，便能形成解集合。

其次建立目標成本函數，這部份可用績效之函數表示。對高科技廠房而言，工期、成本、場地、工率皆為重要評估指標，其中工期用工作天為指標，成本之指標為總出工數，場地指標為暫存區大小需求和取料距離遠近，工率指標則為廠商績效評估指標。各指標與資源配置之優劣關係可參考下表：

表6.1.2 資源配置績效指標關係表

績效指標	優←— —→劣
工期	短←— —→長
總出工數	少←— —→多
容許取料距離	長←— —→短
暫存區需求	小←— —→大
工率	高←— —→低

本研究根據上表訂定績效函數，分別依業主、包商、施工規劃者、全體之考量，制定以下四種績效指標函數，多目標者則由各指標平分權重：

1. 工期：高科技廠房專案業主首要考量。

$$\text{業主成本} = 1 \times \text{工期}$$

2. 總出工數 + 暫存區容量需求：格子版分包商施工成本與施工性考量。

$$\text{分包商成本} = 0.5 \times \text{總出工數} + 0.5 \times \text{暫存區容量需求}$$

3. 暫存區需求 + 容許取料距離 + 工率：營造廠施工規劃考量。

$$\text{營造廠成本} = 0.33 \times \text{暫存區容量需求} + 0.33 \times \text{容許取料距離} + 0.33 \times \text{工率}$$

4. 工期 + 總出工數 + 閒置資源 + 工率：將所有績效一併考慮。

$$\text{全體成本} = 0.25 \times \text{工期} + 0.25 \times \text{總出工數} + 0.25 \times \text{平均閒置資源} + 0.25 \times \text{工率}$$

表6.1.3 成本函數設置

對象	業主	分包商	營造廠	全體
工區面積固定	Yes	Yes	No	Yes
工期	O			O
總出工數		O		O
平均閒置資源				O
工率			O	O
暫存區容量需求		O	O	
容許取料距離			O	

下一步驟為演算法之參數設定，參數設定影響到演算法的收斂速度與求解能力，本研究參考文獻並作適當調整後配置如下：

1. 解空間：將表6.1.1中四個資源變數於限制範圍內各平均切為50等分(暫存區取整數為20等分)，即有 $20 \times 50 \times 50 \times 50 = 2,500,000$ 種可能組合。節點i上之合法解答j被選擇的機率，為費洛蒙係數(τ_{ij})與啟發係數(η_{ij})之函式：

$$P_{ij} = (\tau_{ij}^{\alpha} \cdot \eta_{ij}^{\beta}) / \sum (\tau_{ij}^{\alpha} \cdot \eta_{ij}^{\beta}) \quad (5)$$

2. 最大演算迭代數及每代釋放隻螞蟻數量：越複雜之目標及網圖需要較多迭代數方可收斂，螞蟻數量則影響解答搜尋之廣度，越大越不容易落入區域解，但較難收斂。經試誤法多次測試後發現，螞蟻數量為10隻時，本網圖約於30~60代之間收斂，保守起見最大演算代數設為100代。
3. 費洛蒙函數：每個解答上所含之費洛蒙數量會隨著蟻群的選擇及迭代更新，可分為區域更新、費洛蒙蒸發及全域更新三種情形。費洛蒙更新公式根據各研究的目標不同而有所差異，基本公式如下(本研究整理)：

3.a.區域更新(Local Update)：當合法解答被選擇時，螞蟻留下之費洛蒙。

$$\tau_{ij}' = \tau_{ij} + \theta \quad (6)$$

3.b.費洛蒙蒸發(Pheromone Evaporation)：當螞蟻結束當次搜尋，得到該合法解答組合下的成本後，該組合上的費洛蒙依據相對成本(該螞蟻搜索結果/該代最差結果， T_{hk}/T_h^*)蒸發對應之數量，解答越好蒸發越少。

$$\tau_{hk}' = \tau_{hk} - \rho \cdot (T_{hk}/T_h^*) \quad (7)$$

3.c.全域更新(Global Updating)：每個迭代(Iteration, 每隻螞蟻皆需完成一次搜尋)結束時，所有的費洛蒙皆蒸發一定百分比(λ)，這個步驟是基於前幾個世代的費洛蒙應逐漸降低其影響之強度，讓搜尋的廣度增加。

$$\tau_h' = (1-\lambda) \cdot \tau_h \quad (8)$$

公式(5)之 η_{ij} 、 α 及 β 於本研究中皆設為1，目的為簡化螞蟻搜尋之準則，僅依據費洛蒙之強弱。費洛蒙之計算引用公式(6)及(7)，其中 T_{hk} 為該次獲得之績效指標， T_h^* 為該迭代最佳績效指標。每隻螞蟻會先留下 $\theta = 1$ 的費洛蒙於走過的解答，完成該次搜尋後根據績效計算蒸發量。費洛蒙蒸發係數 $\rho = 1$ ，即最大蒸發量為1。本研究的蒸發參數設定，會使留下的費洛蒙量落於0~1之間，較優解與較差解的費洛蒙量差距較大，且能避免留下負的費洛蒙。其次需設定全域蒸發係數 λ ，此係數代表前一代費洛蒙對後一代之影響強度，本研究設為0.5。

註：符號定義一覽：

1. i ：第 i 個節點
2. j ：第 j 個合法解答
3. τ_{ij} ：費洛蒙數量
4. η_{ij} ：啟發係數
5. α ：費洛蒙重要性指數
6. β ：啟發係數之重要性指數
7. P_{ij} ：該合法解答被選擇機率
8. θ ：蟻群固定留下的費洛蒙數量
9. h ：第 h 迭代
10. k ：第 k 隻螞蟻
11. Th_k ：該螞蟻搜尋所得之成本
12. Th^* ：該迭代中最佳成本
13. ρ ：費洛蒙蒸發係數
14. λ ：全域蒸發係數



6.2 蟻群最佳化演算法計算流程

蟻群最佳化演算法之流程為如下(參照圖6.2.1)：

1. 分析問題：設定各節點(i)的形式、包含之合法解答(j)、成本函數
2. 設定演算法參數：最大迭代數、螞蟻數量、費洛蒙蒸發係數(ρ)
3. 釋放螞蟻：螞蟻依據前一個迭代留下的費洛蒙強度為機率進行組合搜尋。
4. 費洛蒙更新：被選擇之合法解答進行區域更新，搜尋結束後進行費洛蒙蒸發，所有螞蟻皆完成時進行全域更新。
5. 若尚未到達最大演算迭代數或結果未收斂，回到步驟3
6. 演算結束：根據費洛蒙強度輸出優化結果與績效。

釋放蟻群的方式可採取依序釋放(Sequenced Releasing)或批次釋放(Batched Releasing)，兩者差異在於，選擇前者時，該迭代螞蟻間會互相影響。參數之設置已於前一章節中完成。下面將進行單次演算範例：

首先設置解空間的形式，本範例中需求取暫存區、距離、每日平均出工三種資源的配置數量，解答範圍如表6.2.1。

表6.2.1 ACO演算範例：步驟1.解空間配置

$i \setminus j$	1	2	3	4	5	6	7	8
暫存區	5	6	7	8	9	10	11	12
距離	15	20	25	30	35	40	45	50
平均出工	20	22	24	26	28	30	32	34

首先釋放一隻螞蟻A，該螞蟻根據公式(5)計算節點i上解答j被選擇之機率 P_{ij} ，本範例中初始費洛蒙皆為1，例如在節點1(暫存區)上，第3個解答(可儲存7組物料)被選擇之機率為：

$$P_{13} = (1^1 \cdot 1^1) / \sum(1^1 \cdot 1^1) = 0.125$$

螞蟻A根據費洛蒙進行其選擇後，便於各節點上的解答留下定量($\theta = 1$)之費洛蒙，假設螞蟻A選擇了[7 20 28]的組合，費洛蒙改變情形如下表。以 τ_A 代表螞蟻A選擇之資源組合上的費洛蒙量：

$$\tau_A = 1(\text{初始}) + 1(\text{螞蟻留下}) = 2$$

表6.2.2 ACO演算範例：步驟2.費洛蒙區域更新

$i \setminus j$	1	2	3	4	5	6	7	8
暫存區	5	6	7	8	9	10	11	12
距離	15	20	25	30	35	40	45	50
平均出工	20	22	24	26	28	30	32	34

當螞蟻A以該次資源組合完成工作時，便可計算出其績效，並與該迭代中其他螞蟻進行比較，作為費洛蒙蒸發量計算依據。假設螞蟻A耗費60小時完成該工作，而該迭代中工時最長之螞蟻花費100個小時，依公式(7)計算蒸發量如下，費洛蒙改變情形如表6.2.3：

$$\tau_A' = \tau_A - 1 \cdot (60/100) = 2 - 0.6 = 1.4$$

表6.2.3 ACO演算範例：步驟3.費洛蒙蒸發

i \ j	1	2	3	4	5	6	7	8
暫存區	5	6	7	8	9	10	11	12
距離	15	20	25	30	35	40	45	50
平均出工	20	22	24	26	28	30	32	34

當該代所有的螞蟻都結束搜尋，並完成費洛蒙蒸發量計算後，可得到各解答上的費洛蒙量改變情形。接著進行全域蒸發以強調，隔代費洛蒙之影響強度隨著代數遞減的情形。假設該代結束時費洛蒙如表6.2.4，進行全域蒸發後即成為表6.2.5。至此一個迭代的演算結束，下一迭代之螞蟻機率選擇之依據，為步驟5得到的費洛蒙分佈模型。

表6.2.4 ACO演算範例：步驟4.迭代結束之費洛蒙分佈

i \ j	1	2	3	4	5	6	7	8
暫存區	5	6	7	8	9	10	11	12
距離	15	20	25	30	35	40	45	50
平均出工	20	22	24	26	28	30	32	34

表6.2.5 ACO演算範例：步驟5.全域蒸發後之費洛蒙分佈

i \ j	1	2	3	4	5	6	7	8
暫存區	5	6	7	8	9	10	11	12
距離	15	20	25	30	35	40	45	50
平均出工	20	22	24	26	28	30	32	34

不斷重複步驟1~5，經過多個迭代的演算之後，較好的解答會獲得較多費洛蒙，較差的解答則會不斷蒸發減少，最終優劣之間的費洛蒙量差距懸殊，使所有螞蟻都傾向選擇較優的解答，表6.2.6為假設之演算收斂成果，費洛蒙最多的解答為[10 20 32]，即為建議資源組合。

表6.2.6 ACO演算範例：步驟6.演算結束獲得資源配置建議

i \ j	1	2	3	4	5	6	7	8
暫存區	5	6	7	8	9	10	11	12
距離	15	20	25	30	35	40	45	50
平均出工	20	22	24	26	28	30	32	34

以上表6.2.1~6.2.6為一次ACO的演算範例。演算過程採用本研究開發之PetriStar程式，首先用公式(5)計算個別資源項目中，各合法解答(資源配置數量)之機率配合輪盤選擇法(Roulette Wheel Selection)，於四種資源項目各選取一個解答，輸入派翠網中進行模擬，並計算績效，回饋於費洛蒙之區域更新。設定之迭代數達到後，停止釋放螞蟻，此時每個節點上費洛蒙最多者即為建議之解答，最佳化結果詳列於下一小節。

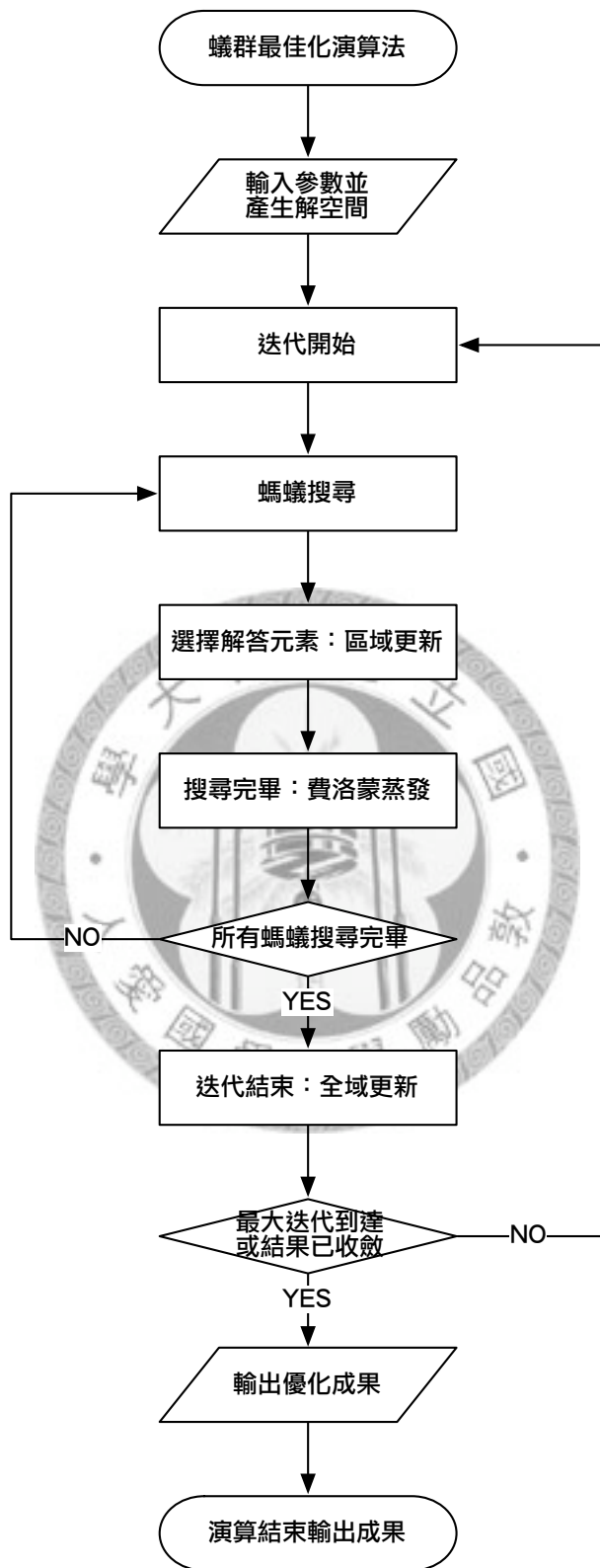


圖6.2.1 蟻群最佳化演算法流程圖

6.3 最佳化結果分析探討

對於高科技廠房新廠專案所採計的四種最佳化指標(業主、包商、規劃者、全體)，本研究發展基於派翠網之蟻群最佳化演算法(Petri-net based Ant Colony Optimization, PACO)，進行資源配置數量求解之最佳化，分別得到收斂後之資源配置建議及對應之最佳化成果。三十次演算結果之平均與標準差依照最佳化目標列於表6.3.1~4，分別繪製其四項績效指標(工期、總出工數、資源閒置率、工率)的平均收斂趨勢如組圖6.3.2~5，並紀錄演算時間與窮舉法需時進行比較。

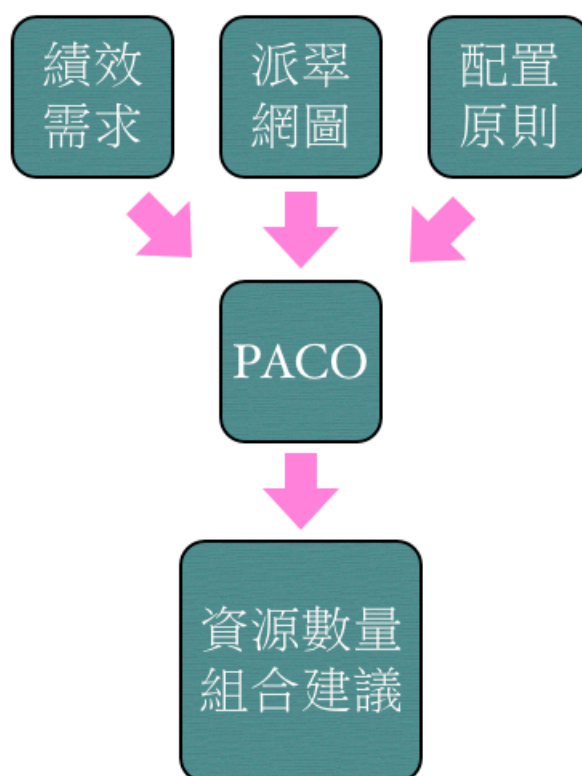
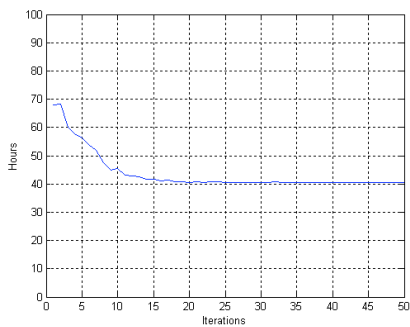


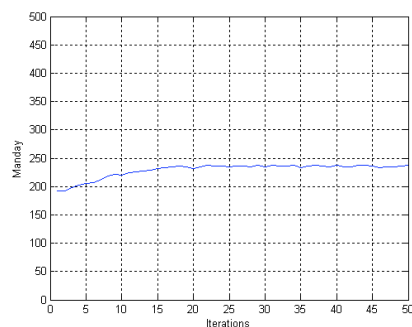
圖6.3.1 PACO模式示意圖

表6.3.1 固定工區面積之工期最佳化(業主)

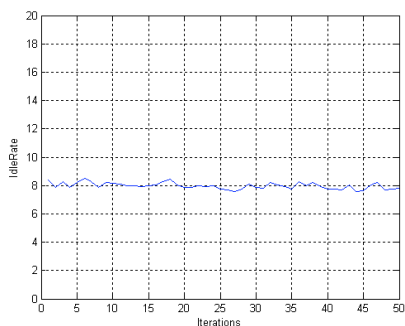
演算時間	212.9866 秒		
工期收斂代數	20 代		
收斂所需搜尋次數	200 次		
窮舉法組合數	$20 \times 50 \times 50 = 50,000$ 組		
節省搜尋次數	49800次(99.6%)		
三十次平均資源建議(標準差)		三十次平均績效指標(標準差)	
工區面積	48(0) 柱位	工期	40.5(0) 小時
容許取料距離	46.38(21.90) M	總出工數	237.11(21.87) 工
暫存區需求	9.77(5.61) 組	閒置物料	7.83(2.51) 組
每日出工數	46.77(6.28) 工	工率	0.6174(0.0574) 噸/工



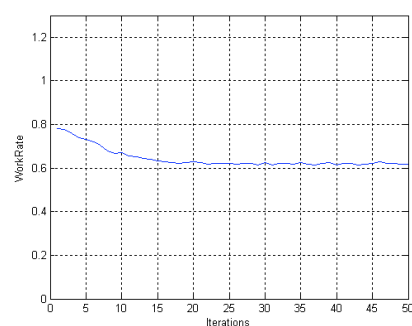
(a)



(b)



(c)

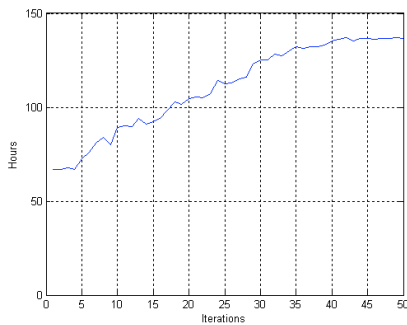


(d)

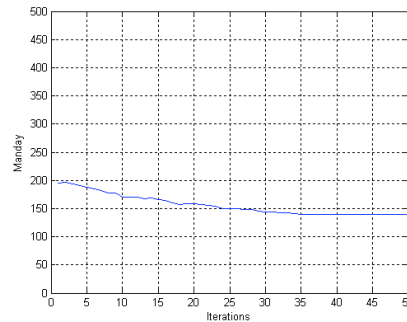
圖6.3.2a~d 四項指標收斂情形(業主)

表6.3.2 固定工區面積之出工數與暫存區需求最佳化(分包商)

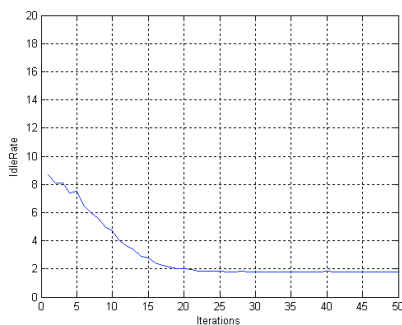
演算時間	435.53782秒		
平均收斂代數	34.25 代		
收斂所需搜尋次數	343 次		
窮舉法組合數	$20 \times 50 \times 50 = 50,000$ 組		
節省搜尋次數	49657次(99.314%)		
三十次平均資源建議(標準差)		三十次平均績效指標(標準差)	
工區面積	48(0) 柱位	工期	136.77(46.62) 小時
容許取料距離	39.90(21.90) M	總出工數	139.05(9.02) 工
暫存區需求	2.03(1.27) 組	閒置物料	1.81(1.18) 組
每日出工數	9.37(4.16) 工	工率	1.0396(0.0642) 噸/工



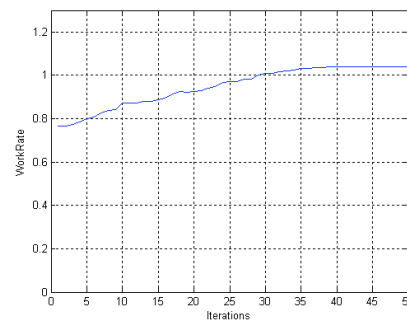
(a)



(b)



(c)

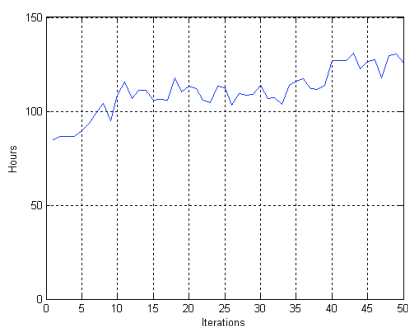


(d)

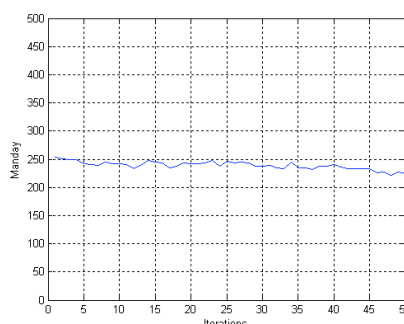
圖6.3.3a~d 四項指標收斂情形(分包商)

表6.3.3 非固定工區面積之暫存區需求、取料距離及工率最佳化(營造廠施工規劃)

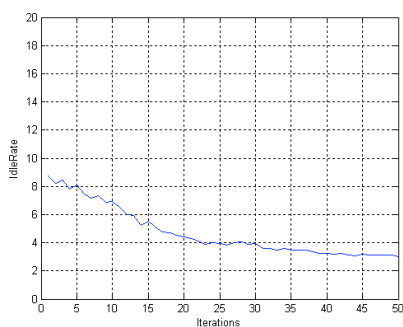
演算時間	380.4339 秒		
平均收斂代數	50 代		
收斂所需搜尋次數	500 次		
窮舉法組合數	$20 \times 50 \times 50 \times 50 = 2,500,000$ 組		
節省搜尋次數	2,499,500次(99.98%)		
三十次平均資源建議(標準差)		三十次平均績效指標(標準差)	
工區面積	65.77(29.19) 柱位	工期	125.50(82.62) 小時
容許取料距離	74.69(10.40) M	總出工數	224.05(88.40) 工
暫存區需求	3.47(2.16) 組	閒置物料	3.04(1.96) 組
每日出工數	19.60(11.35) 工	工率	0.8861(0.0801) 噸/工



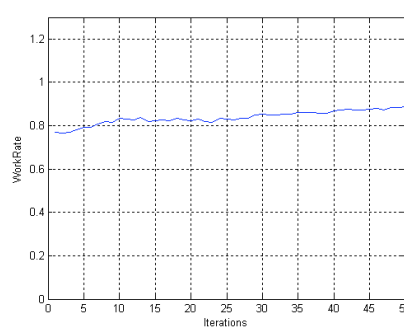
(a)



(b)



(c)

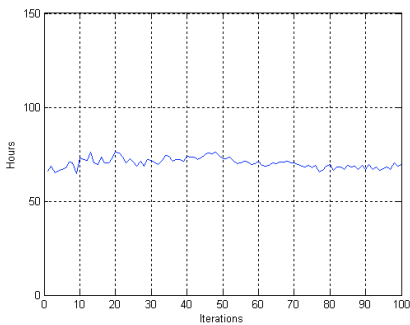


(d)

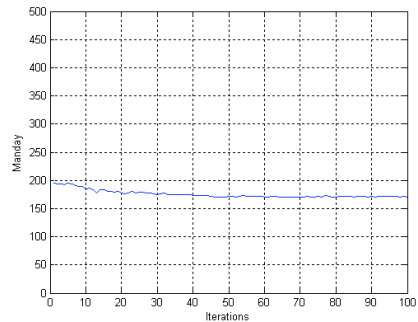
圖6.3.4a~d 四項指標收斂情形(營造廠施工規劃)

表6.3.4 固定工區面積之四項指標最佳化(全體)

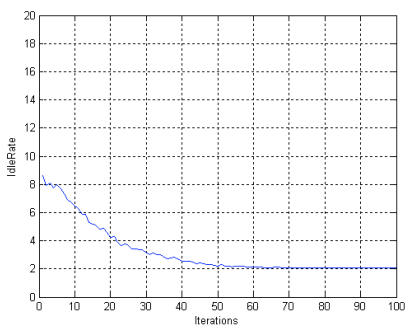
演算時間	629.6289 秒		
平均收斂代數	51.67 代		
收斂所需搜尋次數	517 次		
窮舉法組合數	20×50×50 = 50,000 組		
節省搜尋次數	49483次(98.97%)		
三十次平均資源建議(標準差)		三十次平均績效指標(標準差)	
工區面積	48(0) 柱位	工期	69.30(31.46) 小時
容許取料距離	51.48(23.39) M	總出工數	170.58(23.13) 工
暫存區需求	2.47(1.68) 組	閒置物料	2.05(1.48) 組
每日出工數	23.70(11.41) 工	工率	0.8607(0.1111) 噸/工



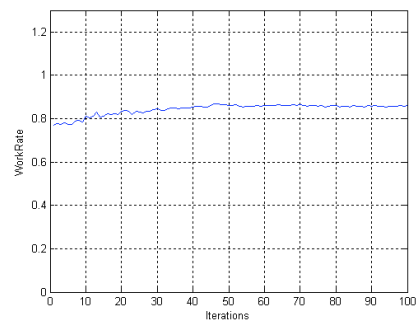
(a)



(b)



(c)



(d)

圖6.3.5a~d 四項指標收斂情形(全體)

首先基於業主對工期的要求，僅對工期進行最佳化。可以發現工期很快就能收斂至最短，但其他指標變動性較大，這是因為純粹以工期為目標時，可能有不同的解決方案(如：提高出工數、縮短距離、增大暫存區...或互相搭配)。在收斂速度方面，由於目標為工期，觀察其圖形發現絕大多數都在15次以內收斂至最短工期(40.5Hr)，每日出工數需求提高(46.77)，伴隨總出工數提高、工率下降的負面效果，對閒置資源則較無影響。

其次是包商之考量，對出工數及暫存區容量需求進行ACO演算，由於最佳化目標有兩項，因此需要較多代的演算，以取得兩者間的平衡。儘管如此，仍可發現這兩者之績效於35代內收斂，並獲得總出工數僅需139.05工，及暫存區需求僅需2.03組的改進，但工期大幅拉長(136.77Hr)。由演算結果發現，以降低總出工數為目標時，會伴隨著工率提高，但工期變長之影響。

第三個目標是針對施工規劃者，考量工區大小與暫存區位置及容量，讓蟻群可以自由選擇工區大小以提高工率並降低暫存區需求，演算目標則為工期、暫存區容量需求、工率。由於多了工區大小的變化，可能的解答組合數目大幅提高，但由於需要考慮的因素更多，以致演算收斂於50代以後。從最佳化建議可觀察出，當工區規模適中(65.77柱位)，出工數略低(每天19.6工)時，暫存區放置距離較自由(74.69M以內)，工率表現亦良好(0.8861噸/工)。由於同時進行三種目標之最佳化，故收斂結果變異性較前兩次演算稍大，但三項績效指標皆大幅改進。

最後則統合業主、包商、規劃者三方面的考量，綜合四個因素進行演算，從三個目標之收斂結果可以發現，運用ACO法仍能迅速地在50代左右收斂，但相對於窮舉法依舊相當迅速。由於需取得四方之平衡，使得工期較難降低至僅考量工期時之成果(69.3：40.5)，但總出工數、工率及資源閒置率仍有不錯的改善。

接下來就演算所需時間進行討論，採窮舉法時，每種資源組合輸入網圖並求取其績效，約需時0.7秒(派翠網模擬所需循環數約與工區大小成正比)。窮舉法

演算所需的時間，可用單次演算時間乘上次數計算。對上面所解的四種問題，節省的時間可參考表6.3.5及表6.3.6。

四個最佳化目標函式中，施工規劃者之解空間雖大，但工區大小影響績效程度較低，故仍能很快速的收斂，不需花費太多時間。綜合以上最佳化的結果得出結論，PACO模式確實能求取目標績效所需之營建資源配置組合，不僅能改善單一目標之規劃績效，對於多目標之最佳化，亦能獲得相當優秀的規劃成果，平均獲得 42.186% 之目標績效提昇。除了提昇規劃水準外，規劃所需時間亦大量減少，符合高科技廠房規劃設計期短的特性。



表6.3.5 ACO演算時間與窮舉法比較

目標	窮舉法需時	ACO需時	搜尋時間比
業主	$50,000 \times 0.7 = 35,000$ 秒	212.99 秒	1/164
包商	$50,000 \times 0.7 = 35,000$ 秒	435.54秒	1/80
施工規劃者	$2,500,000 \times 0.7 = 1,750,000$ 秒	380.43 秒	1/4605
三方	$50,000 \times 0.7 = 35,000$ 秒	629.63 秒	1/56

表6.3.6 ACO績效與隨機組合績效比較表

目標	ACO績效	隨機組合績效	績效提昇
業主	工期—40.5	工期—67.8	工期—67.4%
包商	總出工數—139.05 暫存區需求—2.03	總出工數—196 暫存區需求—9.6	總出工數—40.96% 暫存區需求—83.35%
施工規劃者	暫存區需求—3.47 取料距離—74.69 工率—0.8861	暫存區需求—9.6 取料距離—46.8 工率—0.7667	暫存區需求—63.85% 取料距離—59.59% 工率—12.96%
三方	工期—69.3 暫存區需求—2.47 取料距離—51.48 工率—0.8607	工期—67 暫存區需求—9.8 取料距離—47.3 工率—0.7588	工期—(-3.31%) 暫存區需求—74.79% 取料距離—8.84% 工率—13.43%
平均績效提昇		42.186%	

第七章 結論與建議

7.1 結論與建議

本研究可分為三個部份，(1)案例格子版工作流程與資源配置情形之觀察；(2)派翠網於不同資源配置情形下，施工流程的模擬結果；(3)運用基於派翠網之蟻群最佳化演算法，於資源配置數量組合最佳化的效率。三個部份的結論分別敘述如下：

1. 藉由分析案例日報表，繪製工區大小、取料距離、每日平均出工數與工作績效的關係圖，發現工作量大小、取料距離與每日平均出工數之改變，會對(1)工期(2)總出工數(3)資源閒置率(4)工率這四項的績效產生影響。經由現場觀察與文獻回顧，本研究認為當資源配置數量不同時，工作資源可得性問題亦隨之改變，並導致工作績效之差異。
2. 於派翠網輸入資源配置數量進行模擬的結果，比傳統規劃方式更貼近真實工作績效。與實際施工相比，相關係數平均達到0.754，準確度較傳統估計方式平均提昇42.16%。除了提供更精準的績效推估方式外，亦可藉由輸入不同配置之排列組合進行模擬，繪製出資源與績效之間的關係，提供規劃者參考。
3. 對於單目標及多目標的規劃，ACO搜尋最佳配置組合之速度與績效提昇之成果皆相當優異，績效平均提昇42.186%，時間節省98%以上。加上基於派翠網之準確性，使本模式所獲得的成果更值得信賴。

另外，在派翠網建模的過程中，發現派翠網於營建工程方面之適用範圍，因此做出以下建議：

1. 對於重複性高的流程，派翠網的建模複雜度較低，模擬結果較貼近真實，但對於整個專案之建模，則過度複雜且難以收斂。建議可以針對整個專案中之重複性工項如鋼筋、模板、混凝土、土方等等，進行細部流程建模，估計出較準確的工期後，再回饋於CPM法或其他簡易網圖，即可獲得較準確之估時與較簡單明瞭之網圖。
2. 除了本研究所提到的閒置資源分析、資源與績效相關性分析等等，派翠網亦能用於流程正確性之檢驗，可避免排程規劃之謬誤。
3. 派翠網需進行工程細部流程之建模，故較適合由包商階層著手建置網圖。
4. 營建工程各工項花費的時間受資源與工率影響，於營建工程領域導入派翠網時，應透過各變遷(工項)之資源、工率計算時間。



7.2 研究貢獻

高科技廠房新廠興建專案，受限於景氣與產能規劃，規劃設計與施工之工期皆較一般建築物短，需要更精確的規劃模式。傳統之排程規劃最令人詬病的問題是難以看出資源與工程績效間的關聯性，尤其是工期、總出工數、資源閒置問題及工率四者。此外，傳統排程方法在規劃相同工項時，常假設其工期相同，未就資源配置之特性進行排程工項之工期微調，造成排程不準確，需要大量趕工以符合排程，使臨時出工成本大增，並導致工地現場混亂。

針對以上問題，本研究深入調查高科技晶圓廠之潔淨室格子版結構工程，發展出「基於派翠網之蟻群最佳化演算法」(Petri-net based Ant Colony Optimization, PACO)，結合派翠網能完整連結工作與資源之特性，進行更準確的流程模擬，並藉此進行各項績效指標的分析。其次，基於更準確的派翠網圖，導入ACO，大幅減少搜尋最佳資源組合所需的時間。本模式不僅能運用在高科技廠房格子版工程的範疇，基於派翠網的特性更使本模式擴充性與自由度提高，能運用於各種工作的建模，ACO在面對不同類型的工程問題時，只需改變目標函數，即可求取對應該工程之結果。歸納本研究貢獻如下：

1. 深入調查高科技廠房格子版結構工程，調查分析其工序與工率影響因素。
2. 從施工者角度進行工作流程建模，以派翠網突顯資源使用問題，驗證派翠網於營建工程模擬之實用性與適用範圍。
3. 過去的最佳化演算法基於CPM法，估時變動性過大，導致最佳化成果可信度低，本研究發展基於派翠網之蟻群最佳化演算法，獲得更可靠的最佳化結果。

7.3 研究限制

本研究限制如下：

1. 輸入派翠網之資源數量不可於工程進行中變更，且流程不可中斷。
2. 基準工率固定，排除擁擠、工安等外在因素影響。
3. 不考慮工區間資源共享問題。

7.4 未來研究建議

由於廠商施工日報表紀錄不夠詳盡，其對於分區暫存區容量、資源抵達速度、分區出工數、風險意外事故、停工區域等等未作完整紀錄，本研究對於資源配置組合之分析仍有部份誤差難以消弭。建議未來研究可朝向以下方向：

1. 提供更準確的施工報表，研究增進本網圖準確度的方法，並配合隨機派翠網模擬風險狀況。
2. 將網圖各環節的連接定義的更清楚，減少系統性誤差。
3. 建議與4D工程圖面連結，精確地考量資源組合之範圍限制，建立資源動態模擬系統，使最佳化成果更可信賴。

參考文獻

英文文獻

1. Aalst, W. M. P. v. d. (1998). "The Application of Petri Nets to Workflow Management." *Journal of Circuits, Systems, and Computers (JCSC)* 8(1).
2. Ashok Kumar, V. K. and L. S. Ganesh (1998). "Use of Petri Nets for Resource Allocation in Projects." *IEEE Transactions on Engineering Management* 45(No. 1).
3. Diego, E., C. W. Ibbs, et al. (1991). "Sequencing Knowledge for Construction Scheduling." *Journal of Construction Engineering and Management* 117(1): 118-130.
4. Heng Li. (1998). "Petri net as a formalism to assist process improvement in the construction industry." *Automation in Construction* 7: 8.
5. J. Prashant Reddy, S. K. a. O. V. K. C. (2001). "Application of Petri Nets and a Genetic Algorithm to Multi-Mode Multi-Resource Constrained Project Scheduling." *international journal of advanced manufacturing technology* 17: 10.
6. Khaled Nassar, A. C. (2008). "Analysis of Timed Petri Nets for Reachability in Construction Applications." *Journal of Civil Engineering and Management* 14(3): 189-198.
7. Koo, B. (2004). "Formalizing construction sequence constraint for the rapid generation of scheduling alternatives." Technical Report Nr.155, CIFE, Stanford University. Stanford ,CA.
8. M. D. Jeng, X. X., and S. W. Chou (1998). "Modeling, Qualitative Analysis, and Performance Evaluation of the Etching Area in an IC Wafer Fabrication System

- Using Petri Nets." IEEE Transactions on Semiconductor Manufacturing 11(3): 358-373.
9. Marco Dorigo, V. M., Alberto Colorni (1996). "Ant System: Optimization by a Colony of Cooperating Agents." IEEE Transactions on Systems, Man, and Cybernetic-Part B, Cybernetics 26(1): 13.
 10. Merkle, D. M., M.; Schmeck, H. (2002). "Ant colony optimization for resource-constrained project scheduling." Evolutionary Computation, IEEE Transactions 6(4): 14.
 11. Murata, T. (1989). "Petri nets: Properties, analysis and applications." Proceedings of the IEEE 77(4): 541-580.
 12. Ron, R. Wakefield. and A. S. Glenn (1997). "Petri Nets for Simulation and Modeling of Construction Systems." Journal of Construction Engineering and Management 123(2): 105-112.
 13. Sawhney, A. (1997). "Petri Net Based Simulation Of Construction Schedules." 1997 Winter Simulation Conference (WSC'97): 1111-1118.
 14. Stuart Russell, P. N. (2003). "Artificial Intelligence: A Modern Approach." Prentice Hall/Pearson Education.
 15. Tarek Hegazy. (1999). "Optimization of Resource Allocation and Leveling Using Genetic Algorithms." Journal of Construction Engineering and Management 125(3): 167-175.
 16. Tarek Hegazy, E. Zanelidin, et al. (2001). "Improving Design Coordination for Building Projects. I: Information Model." Journal of Construction Engineering and Management 127(4): 322-329.

17. Thomas, H. R., J. H. Michael, et al. (2003). "Improving Labor Flow Reliability for Better Productivity as Lean Construction Principle." *Journal of Construction Engineering and Management* 129(3): 251-261.
18. Wiest, J. D. (1964). "Some Properties of Schedules for Large Projects with Limited Resources." *Operation Research* 12(3): 24.
19. Yen, C.-I. (2005). "Simulated Annealing For Optimizing Linear Scheduling Projects With Multiple Resource Constraints." Dissertation, Purdue University, West Lafayette, Indiana, USA.

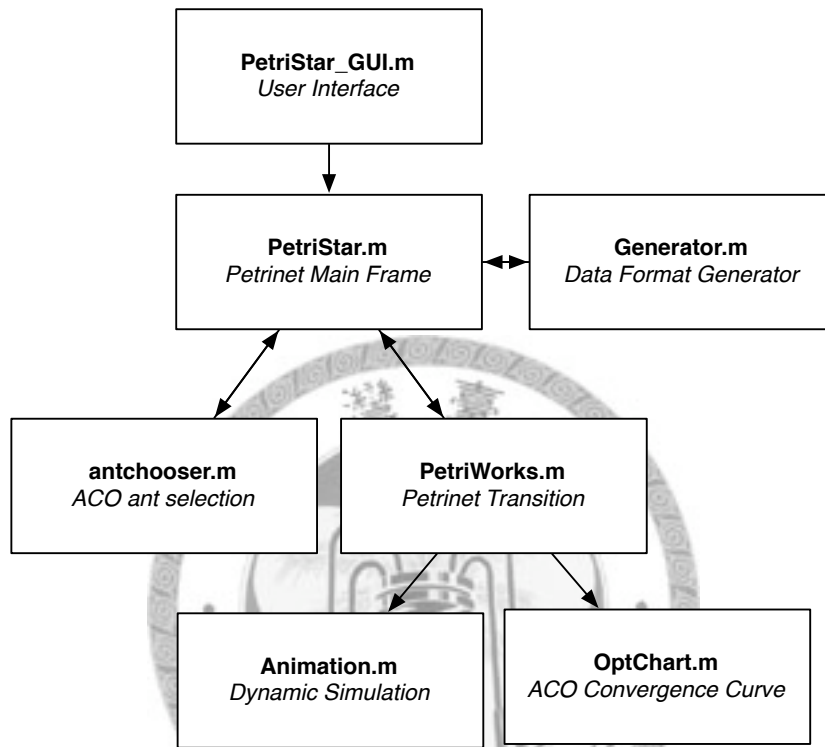
中文文獻

1. 張書萍 (2000). "高科技廠房營建工程特性之調查與分析." 國立交通大學土木工程研究所.
2. 馬良 (2000). "基於螞蟻算法的函數優化." 上海理工大學管理學院.
3. 曾建發 (2002). "應用電腦模擬工具求解營建作業排程最佳工期之研究." 朝陽科技大學營建工程系碩士論文.
4. 林訓宇 (2002). "應用斐氏網路與遺傳演算法於晶圓測試廠之模型建構與排程." 國立台灣大學資訊工程學研究所.
5. 楊學紅, 胡志根等, (2002). "隧洞施工過程的Petri網絡模擬方法." *系統仿真學報* 14(8).
6. 蕭博文 (2002). "應用禁忌搜尋法求解營建工程專案多重資源排程最佳化之研究." 朝陽科技大學營建工程系.
7. 王明信 (2004). "高科技製造廠縮短工期模式之研究--以南部科學工業園區光電、積體電路廠辦為例." 國立成功大學建築研究所碩士論文.

8. 劉全, 楊學紅等, (2005). "基于賦時Petri網大壩混凝土施工過程建模與實現." 水電能源科學 23(4).
9. 孫國勛 (2005). "工作群組式重複性專案排程與最佳化模式之研究." 國立中央大學土木工程研究所博士論文.
10. 張浩偉 (2005). "應用時間派翠網路於DVD光碟製程之分析." 國立中央大學機械工程所碩士論文.
11. 章允建 (2005). "應用蟻群演算法在解決TFT-LCD物料規劃最佳化之研究." 元智大學資訊管理學系碩士論文: 91.
12. 張紹陽, 王選倉 (2006). "基于Petri網的公路施工資源瓶頸識別和消除." 長安大學學報 26: 5.
13. 許書凱 (2006). "應用多維度圖形元件於營建專案排程規劃之研究." 國立成功大學土木工程研究所碩士論文.
14. 鄭衛國, 吳乃森等, (2006). "基於著色Petri網的鋼結構施工過程仿真研究." 工業工程與管理 4.
15. 張陸滿 (2008). "奈米時代之高科技廠房設施工程." 土木水利期刊 1(35): 12.

附錄：PetriStar程式碼

PetriStar程式架構如下圖：



附圖1. PetriStar系統架構圖

程式碼Part A : PetriStar_GUI.m

```
function PetriStar_GUI
% PetriStar is a newly developed program for Petri-net-based dynamic process simulation,
% and automatic resource allocation with ant colony optimization
% All rights reserved to Yao An, Wu, the developer of PetriStar, this version is a public beta
% Any further suggestions or modifications of PetriStar should notify the developer in advance
% r96521702@ntu.edu.tw, Yao An, Wu, last updated on 2009/06/26

f = figure('Visible','off','Position',[360,500,600,480]);
Petrick = imread('PatrickStar.png');

% Construct the components.
Start = uicontrol('Style','pushbutton','String','Start','Position',[450,400,70,25],'Callback',...
{@startbutton_Callback});
Anttext = uicontrol('Style','text','String','Ant Released','Position',[450,330,70,15]);
AntNum = uicontrol('Style','edit','String','1','Position',[450,315,70,15],'Callback',@antnum_callback);
Gtext = uicontrol('Style','text','String','Generations','Position',[450,290,70,15]);
GenNum = uicontrol('Style','edit','String','1','Position',[450,275,70,15],'Callback',@gennum_callback);
Dtext = uicontrol('Style','text','String','Evaporation Rate','Position',[450,250,70,15]);
EvpRate = uicontrol('Style','edit','String','0.5','Position',[450,235,70,15],'Callback',@decay_callback);
stext = uicontrol('Style','text','String','Select Data','Position',[450,210,70,15]);
datamenu = uicontrol('Style','popupmenu','String',{'Steel Assembling','FRP Assembling'},...
'Position',[450,180,70,25],'Callback',{@data_menu_Callback});
Atext = uicontrol('Style','text','String','Area Number','Position',[450,160,70,15]);
Area = uicontrol('Style','edit','String','1','Position',[450,145,70,15],'Callback',@area_callback);
spdtext = uicontrol('Style','text','String','AnimSpeed','Position',[450,120,70,15]);
AnimSpeed = uicontrol('Style','popupmenu','String',{'Very Fast','Fast','Normal','Slow','Very Slow'},...
'Position',[450,90,70,25],'Callback',{@animspeed_callback});
subplot(1,5,[1:3],'replace');
imshow(Petrick);

% align([Start,Result,stext,datamenu,AntNum,GenNum],'Center','None');

% Data Initialize using Steel Assembling
tick = 0.000001; GA = 1; ants = 1; Evaporation = 0.5; number = 1;
```

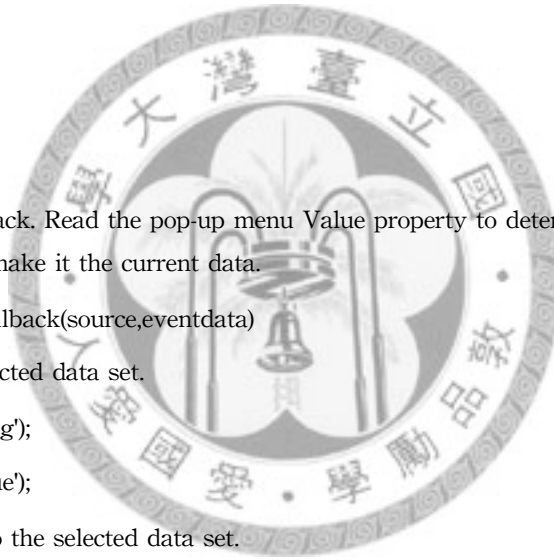
```

p = load('STEEL1.txt');
t = load('TransitionLoader1.2s.txt');
g = load('Graph1.2s.txt');
tstat(1,1) = 1; pstat(1,1) = 1;
pInit = [1 0;2 0;3 0;6 0;12 105;];

% Initialize the GUI.
% Change units to normalized so components resize automatically.
set([Start,stext,Antttext,Gtext,Dtext,AntNum,GenNum,EvpRate,datamenu,Area,...
Atext,spdttext,AnimSpeed],'Units','normalized');
% Assign the GUI a name to appear in the window title.
set(f,'Name','PetriStar')
movegui(f,'center')
set(f,'Visible','on');

% Pop-up menu callback. Read the pop-up menu Value property to determine which item is
currently displayed and make it the current data.
function data_menu_Callback(source,eventdata)
% Determine the selected data set.
str = get(source,'String');
val = get(source,'Value');
% Set current data to the selected data set.
switch str {val};
    case 'Steel Assembling'
        filename = ['Steel' num2str(number) '.txt'];
        p = load(filename);
        t = load('TransitionLoader1.2s.txt');
        g = load('Graph1.2s.txt');
        pInit = [1 0;2 0;3 0;6 0;12 105;]; %Initial Place Marking
    case 'FRP Assembling'
        filename = ['FRP' num2str(number) '.txt'];
        p = load(filename);
        t = load('TransitionLoader1.2pm.txt');
        g = load('Graph1.2p.txt');

```



```

        pInit = [1 0;2 0;3 0;6 0;12 105;]; %Initial Place Marking
    end
end
% InputBox
function animspeed_callback(source,eventdata)
    str = get(source,'String');
    val = get(source,'Value');
    switch str{val};
        case 'Very Fast'; tick = 0.000001;
        case 'Fast'; tick = 0.05;
        case 'Normal'; tick = 0.1;
        case 'Slow'; tick = 0.2;
        case 'Very Slow'; tick = 0.5;
    end
end
% Start button callbacks
function [tstat,pstat] = startbutton_Callback(source,eventdata)
[tstat,pstat,dur] = PetriStar(p,t,g,pInit,tick,GA,ants,Evaporation);
%   Tt = zeros(size(tstat,1),3*dur+1);
%   Pt = zeros(size(pstat,1),2*dur+1);
%   for i = 1:dur+1
%       Tt(:,(i-1)*3+1:i*3) = tstat(:,i);
%       Pt(:,(i-1)*2+1:i*2) = pstat(:,i);
%   end
%   save('tstat.txt','Tt','-ascii');
%   save('pstat.txt','Pt','-ascii');
%   save('dur.txt','dur','-ascii');
end
% Input Variance callbacks
function antnum_callback(source,eventdata)
    ants = str2double(get(source,'String'));
end
function genuum_callback(source,eventdata)
    GA = str2double(get(source,'String'));

```




```
end  
  
function decay_callback(source,eventdata)  
    Evaporation = str2double(get(source,'String'));  
end  
  
function area_callback(source,eventdata)  
    number = str2double(get(source,'String'));  
end  
  
end
```



程式碼Part B：PetriStar.m

```
function [tstat,pstat,t] = PetriStar(Place,Transition,Graph,pInit,tick,GA,ants,EvpRate)

clc;

%=====Load Data=====

Limitation = load('limit1.2s.txt');

t = 0; maxloop = 9000; Availables = 50; DecayRate = 1;

[pdata, tdata, pstat, tstat, RouteCap, RouteUnit, PheromoneCap, PheromoneUnit] =
Generator(Place,Transition,maxloop,Limitation,Availables);

ProgramEnd = max(pdata(:,1));

ResTypes = size(Limitation,1);

tic;

duration = zeros(GA,ants); manday =zeros(GA,ants); workrate = zeros(GA,ants); storage =
zeros(GA,ants); distance = zeros(GA,ants); IdleRes = zeros(GA,ants); OptFunction = zeros(GA,ants);
choosen = zeros(ResTypes,2,ants);
processing = waitbar(0,'Program Executing..');

% Run GA generations of Ant Algorithm, each generation release k ants
for g = 1:GA
    CapPhBegin = PheromoneCap;
    UnitPhBegin = PheromoneUnit;
    for k = 1:ants
        % Choose Resource Set According to CapPhBegin, update Pheromone
        [pdata choosen(:,k)] = antchooser(pdata, RouteCap, RouteUnit, CapPhBegin,...
            UnitPhBegin, ResTypes, Availables);
        % Reset statement
        pstat(:) = 0; tstat(:) = 0; t = 0;
        for i = 1:size(pInit,1)-1
            pstat(pInit(i))= unique(pdata(pdata(:,1)==pInit(i),4)); %Set Initial Tokens
        end

        % PetriWorks
        while pstat(ProgramEnd,1,t+1) ~= pInit(size(pInit,1),2);
            t = t+1;
        end
    end
end
```

```

    [pstat(:,t+1),tstat(:,t+1)] = PetriWorks(pdata,tdata,pstat(:,t),tstat(:,t));
    if t == maxloop; break; end
    if tstat(:,1:2,t+1) == tstat(:,1:2,t); break; end;
end
% Calculate Performance
duration(g,k) = t;
manday(g,k) = t/48*pdata(6,5);
workrate(g,k) = pdata(2,4)*pdata(2,5)/manday(g,k);
distance(g,k) = pdata(4,5)/50;
storage(g,k) = pdata(5,4);
IdleRes(g,k) = pstat(5,2,t+1)/(t+1);
end
% Update Pheromone According to Chosen Route
for k = 1:ants
% Optimizing Function Can Always be Modified to Fit Different Requirements
OptFunction(g,k) = 1 - DecayRate*(0.2*duration(g,k)/max(duration(g,:))+...
0.2*manday(g,k)/max(manday(g,:))+0.2*(1-distance(g,k)/max(distance(g,:)))+...
0.2*storage(g,k)/max(storage(g,:))+0.2*(1-workrate(g,k)/max(workrate(g,:)));
for i = 1:ResTypes
    PheromoneCap(i,chosen(i,1,k)) = PheromoneCap(i,chosen(i,1,k)) + OptFunction(g,k);
    PheromoneUnit(i,chosen(i,2,k)) = PheromoneUnit(i,chosen(i,2,k)) + OptFunction(g,k);
end
end
% Pheromone Decay
PheromoneCap = PheromoneCap * (1 - EvpRate);
PheromoneUnit = PheromoneUnit * (1 - EvpRate);
waitbar(g/GA);
end
close(processing);
efficiency = tstat(:,3,t+1)/(t+1)
efficiencyRes = pstat(:,2,t+1)/(t+1)
toc;
for i = 1:t+1

```

```

if mod(i,5) == 0
    subplot(1,5,[1:3],'replace')
    Animation(tstat(:,i),pstat(:,i),tdata,pdata,Graph,i,tick);
end
end
% Obtain Route Suggestion
RouteSet = [max(PheromoneCap)' max(PheromoneUnit)'];
for i = 1:ResTypes
    for j = 1:Availables
        if PheromoneCap(i,j) == RouteSet(i,1);
            RouteSet(i,1) = RouteCap(i,j);
        end
        if PheromoneUnit(i,j) == RouteSet(i,2);
            RouteSet(i,2) = RouteUnit(i,j);
        end
    end
end
end
OptChart(duration,GA,manday,workrate,storage,IdleRes,distance);

```



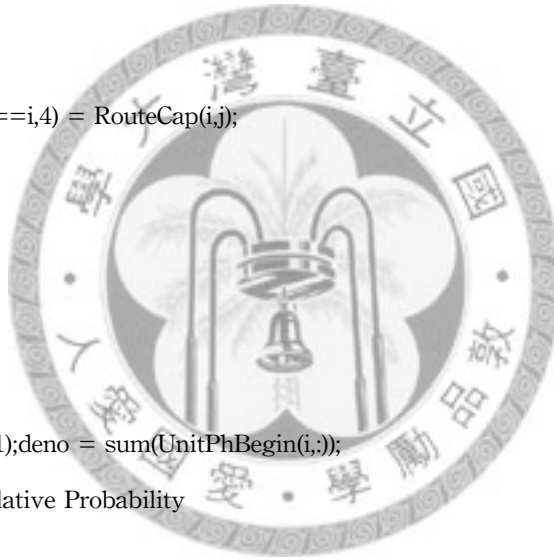
程式碼Part C : Generator.m

```
function [pdata, tdata, pstat, tstat, RouteCap, RouteUnit, PheromoneCap, PheromoneUnit] =  
Generator(Place,Transition,maxloop,limit,Availables)  
  
% pdata = [No(1),From(2),To(3),Cap(4),/Token(5),ResourceType(6),]  
% tdata = [No(1),From(2),To(3),Type(4),Attribute(5),Resource(6),Labor(7)]  
% Limit = [UpperLimit(1),LowerLimit(2),UpperLimit-U(3),LowerLimit-U(4)]  
pstat = zeros(max(Place(:,1)),2,maxloop);  
pdata = Place;  
tstat = zeros(max(Transition(:,1)),3,maxloop);  
tdata = Transition;  
% Adjust Place Capacity  
PheromoneCap = ones(size(limit,1),Availables);  
PheromoneUnit = ones(size(limit,1),Availables);  
RouteCap = zeros(size(limit,1),Availables); % Alternatives  
RouteUnit = zeros(size(limit,1),Availables); % Alternatives  
for i = 1:size(limit,1)  
    for j = 1:Availables  
        adjust = (limit(i,1)-limit(i,2))/(Availables-1);  
        RouteCap(i,j) = round(limit(i,2) + adjust*(j-1));  
    end  
end  
for i = 1:size(limit,1)  
    for j = 1:Availables  
        adjust = (limit(i,3)-limit(i,4))/(Availables-1);  
        RouteUnit(i,j) = round(limit(i,4) + adjust*(j-1));  
    end  
end
```



程式碼Part D : antchooser.m

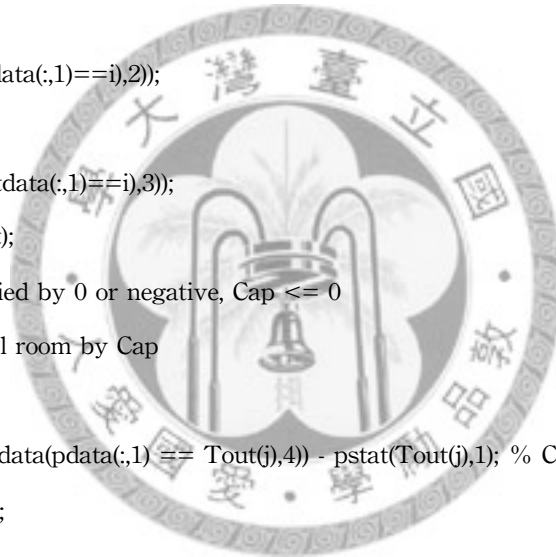
```
function [pdata choosen] = antchooser(pdata, RouteCap, RouteUnit, CapPhBegin, UnitPhBegin, Type,
Availables)
choosen = zeros(Type,2);
for i = 1:Type
    rnd = random('unif',0,1);deno = sum(CapPhBegin(i,:));
    cprob = 0; % Cumulative Probability
    for j = 1:Availables
        cprob = cprob+CapPhBegin(i,j)/deno;
        % Update Pheromone
        if cprob >= rnd
            choosen(i,1) = j;
            pdata(pdata(:,6)==i,4) = RouteCap(i,j);
            break;
        end
    end
end
for i = 1:Type
    rnd = random('unif',0,1);deno = sum(UnitPhBegin(i,:));
    cprob = 0; % Cumulative Probability
    for j = 1:Availables
        cprob = cprob+UnitPhBegin(i,j)/deno;
        % Update Pheromone
        if cprob >= rnd
            choosen(i,2) = j;
            pdata(pdata(:,6)==i,5) = RouteUnit(i,j);break;
        end
    end
end
end
```



程式碼Part E：PetriWorks.m

```
function [pstat,tstat] = PetriWorks(pdata,tdata,pstat,tstat)
% pdata = [No(1),From(2),To(3),Cap(4),Group/Token(5),Unit(6),UpperLimit(7),LowerLimit(8)]
% tdata = [No(1),From(2),To(3),TType(4),Attribute(5),Resource(6),Labor(7)]
% pstat = [Tokens,TotalWaitTime]
% tstat = [Enable,Progress]
% Workload = [Groupsize,WorkRate,Resource]
Tn = max(tdata(:,1));

% Remove/Add Token when tokens < Capacity
for i = 1:Tn
    Tin = unique(tdata((tdata(:,1))==i),2);
    [mTin n] = size(Tin);
    Tout = unique(tdata((tdata(:,1))==i),3);
    [mTout n] = size(Tout);
    Cap = 1; % If multiplied by 0 or negative, Cap <= 0
    % Check if there's still room by Cap
    for j = 1:size(Tout)
        Capleft = unique(pdata(pdata(:,1) == Tout(j),4) - pstat(Tout(j),1); % Cap - Token
        Cap = Cap*Capleft;
    end
    if tstat(i,2) >= 1 && Cap > 0 % Cap cannot be 0 or negative
        for j = 1:mTin
            pstat(Tin(j),1) = pstat(Tin(j),1) - 1;
            tstat(i,1) = 0;
            tstat(i,2) = 0;
        end
        for j = 1:mTout
            pstat(Tout(j),1) = pstat(Tout(j),1) + 1;
        end
    end
end
end
```




```

% Update Progress
for i = 1:Tn
    checker = 1;

    Tin = unique(tdata((tdata(:,1)==i),2));
    [mTin n] = size(Tin);
    Tout = unique(tdata((tdata(:,1)==i),3));
    [mTout n] = size(Tout);

    % if any of Tin place have no token, checker = 0, or checker will >= 1
    % only check when T is not processing
    if tstat(i,2) == 0
        for j = 1:mTin
            checker = checker*pstat(Tin(j),1);
        end
        tstat(i,1) = boolean(checker>=1);
    end
    if tstat(i,1) == 1 && tstat(i,2) <= 1
        tstat(i,3) = tstat(i,3) + 1;
        TriggerType = max(tdata((tdata(:,1)==i),4));
        switch TriggerType
            % Auto--
            case 1
                tstat(i,2) = 1;
            %
            % Timed--
            case 2
                tstat(i,2) = tstat(i,2) + 1/unique(tdata((tdata(:,1)==i),5));
            %
            % Resource--
            case 3
                material = unique(tdata(tdata(:,1)==i,6));
                labor = unique(tdata(tdata(:,1)==i,7));
                Workload(1) = unique(pdata(pdata(:,1)==labor,5)); % Res
                Workload(2) = unique(tdata(tdata(:,1)==i,5)); % Gsize
                Workload(3) = unique(pdata(pdata(:,1)==material,5)); % WorkRate
        end
    end
end

```



```

        tstat(i,2) = tstat(i,2) + Workload(1)*Workload(2)/Workload(3);
%         --disp('res');
    end
end
end
tstat(tstat(:,2) >= 1,2) = 1; % Progress cannot be more than 1;
% Update WaitTime
Pn = max(pdata(:,1));
for i = 1:Pn
    pstat(i,2) = pstat(i,2) + pstat(i,1);
    if unique(pdata(pdata(:,1)==i,3)) ~= 0 && tstat(unique(pdata(pdata(:,1)==i,3))) == 1
        pstat(i,2) = pstat(i,2) - 1;
    end
end
end

```

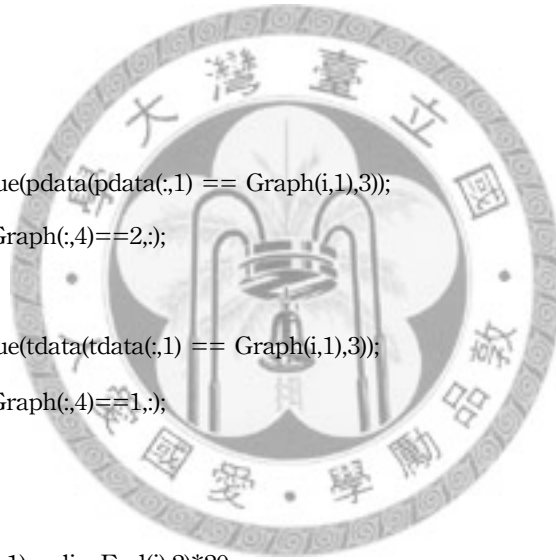


程式碼Part F：Animation.m

```
function Animation(tstat,pstat,tdata,pdata,Graph,time,tick)
% [Node(1),xlevel(2),ylevel(3),NodeType(4)]
[m n]= size(Graph);
hold on;

% Draw Arrows
for i = 1:m
    nodetype = Graph(i,4);
    x = Graph(i,2)*20;
    y = Graph(i,3)*20;
    switch nodetype
        case 1
            lineEnd = unique(pdata(pdata(:,1) == Graph(i,1),3));
            node = Graph(Graph(:,4)==2,:);
        case 2
            lineEnd = unique(tdata(tdata(:,1) == Graph(i,1),3));
            node = Graph(Graph(:,4)==1,:);
        end
        for j=1:size(lineEnd)
            xend = node(node(:,1)==lineEnd(j),2)*20;
            yend = node(node(:,1)==lineEnd(j),3)*20;
            line([x xend],[y yend])
        end
    end
end

% Draw Place & Transition
for i = 1:m
    nodetype = Graph(i,4);
    x = Graph(i,2)*20;
    y = Graph(i,3)*20;
    switch nodetype
        case 1
```



```

plot(x,y,'MarkerSize',15,'Marker','o','MarkerFaceColor','w','MarkerEdgeColor','b');

text(x-2,y,int2str(pstat(Graph(i,1),1)), 'fontsize',9);

case 2

plot(x,y,'MarkerSize',15,'Marker','s','MarkerFaceColor','w','MarkerEdgeColor','r');

if tstat(Graph(i,1),1) == 1

percent = [num2str(tstat(Graph(i,1),2)*100,3) '%'];

rectangle('Position',[x-7,y-6,14,12],'FaceColor',[.8 .9 .9],'LineStyle','none');

rectangle('Position',[x-7,y-6,14,tstat(Graph(i,1),2)*12+0.01],'FaceColor',[.6 .9 .6],'LineStyle','none');

rectangle('Position',[x-7,y-6,14,12]);

text(x-2,y,percent,'fontsize',8);

end

end

hold on;

end

% Update Elapsed Time
text(max(Graph(:,2))*20-20,max(Graph(:,3))*20+20,['time elapsed = ' int2str(time+1)],'BackgroundColor',[.8 .8 .3])

axis([0 max(Graph(:,2))*20+40 0 max(Graph(:,3))*20+40]);

set(gca,'YDir','reverse');

pause(tick); % update every x sec

```



程式碼Part G : optchart.m

```
function OptChart(duration,GA,manday,workrate,storage,idlerate,Distance)

Opt1 = figure;
GD = mean(duration)';
GD = GD/6;
plot(GD)
axis([0 GA min(GD)-10 max(GD)+10]);
grid on; xlabel('Iterations'); ylabel('Hours');

Opt2 = figure;
Man = mean(manday)';
plot(Man)
axis([0 GA min(Man)-10 max(Man)+10]);
grid on; xlabel('Iterations'); ylabel('Manday');

Opt3 = figure;
Rate = mean(workrate)';
plot(Rate)
axis([0 GA 0 max(Rate)+0.2]);
grid on; xlabel('Iterations'); ylabel('WorkRate');

Opt4 = figure;
Store = mean(storage)';
plot(Store)
axis([0 GA 0 max(Store)+5]);
grid on; xlabel('Iterations'); ylabel('Storage');

Opt5 = figure;
IdleRate = mean(idlerate)';
plot(IdleRate)
axis([0 GA 0 max(IdleRate)+5]);
grid on; xlabel('Iterations'); ylabel('IdleRate');

Opt6 = figure;
Distance = mean(Distance)';
plot(Distance)
axis([0 GA 0 max(Distance)+5]);
```



```
grid on; xlabel('Iterations'); ylabel('Distance');
```

```
% show last result
```

```
GD(GA)
```

```
Man(GA)
```

```
Store(GA)
```

```
IdleRate(GA)
```

```
Rate(GA)
```

```
Distance(GA)
```



附表1 Place Data Loader Format (example for steel assembling)

Place No.	from	to	Capacity	Units per Token	Resource Type
1	2	1	105	3	1
2	0	1	105	3	2
3	0	2	105	1	2
4	1	2	105	3312	2
5	1	2	105	3	2
6	3	2	1	35.15	3
7	2	4	105	3	3
8	2	4	1	35.15	3
9	2	3	105	3312	3
10	4	5	105	3	4
11	4	3	1	35.15	4
12	5	0	105	3	2

附表2 Place Data Loader Format (example for steel assembling)

Transition No.	from	to	Transition Type	Unit Rate	Resource Used (Material)	Resource User (Labor)
1	1	4	2	2	0	0
1	2	5	2	2	0	0
2	3	7	3	300	4	6
2	4	8	3	300	4	6
2	5	9	3	300	4	6
2	6	1	3	300	4	6
3	9	6	3	300	9	11
3	11	6	3	300	9	11
4	7	10	3	0.02645833	7	8
4	8	11	3	0.02645833	7	8
5	10	12	2	2	0	0

附表3 Graph Loader Format (example for steel assembling)

Node No.	from	to	Node Type
1	1	1	1
2	2	1	1
3	3	1	1
4	1	3	1
5	2	3	1
6	3	3	1
7	1	5	1
8	2	5	1
9	3	5	1
10	2	7	1
11	3	7	1
12	2	9	1
1	2	2	2
2	2	4	2
3	4	4	2
4	2	6	2
5	2	8	2

附表4 Resource Limitation Loader Format (example for steel assembling)

	Place Capacity – Upper Limit	Place Capacity – Lower Limit	Unit Per Token – Upper Limit	Unit Per Token – Lower Limit
Workload	48	48	3	3
Distance (distance × workers needed to carry one unit)	105	105	4250	500
Storage Size	20	1	3	3
Workers per Day	1	1	55	5