

逢 甲 大 學
交通工程與管理學系碩士班
碩士論文

螞蟻記憶系統應用於旅行推銷員問題

**The Application of Ant Memory System
on Traveling Salesman Problems**

指導老師：林大傑 教授

研 究 生：陳冠樺

中 華 民 國 九 四 年 七 月

誌 謝

兩年研究所生活即將尾聲，學生的身份也將告一段落，而人生另一挑戰也即將展開。回首這兩年的研究所生活，首先感謝恩師 林大傑博士悉心指導，當我在嘗試不同想法進行測試時，恩師總能在最適當的時機給我一臂之力使我論文得以順利完成，並且在做事之態度上也感謝恩師寶貴意見，讓我在這兩年中，除了知識的增長外，對於做人處事更具另一種體認。論文口試期間，承蒙口試委員台灣大學土木工程學系暨研究所 張堂賢博士以及淡江大學運輸管理學系 陳敦基博士給予許多寶貴意見及經驗，使得本論文得以更臻完善。論文撰寫期間，感謝 胡大瀛老師在論文構架上的指正以及 李克聰老師傳授文獻撰寫的技巧。

研究所的兩年期間，感謝同窗好友秋評、宇軒、阿斐、曜彰、小龜、光鎧、tt、阿潘、俊宏、自強、雲慶、表弟、品翰、裕雯、怡芳、怡如、偉強、孟宗以及宗憲；學弟妹阿展、馬祖等讓我研究所的生活不會乏味；羽球隊學弟妹讓我的體重不會直線往上飆。感謝研究室學弟妹張耿禎、賴志豪、洪韻珏在計劃案的協助，使我在執行計劃案之餘還有充足的時間完成論文。感謝這二年來一直陪伴我的秋評，當我遇到挫折時鼓勵我、當我壓力大到喘不過氣時協助我，感謝她的體諒、支持，使我的論文得以順利完成。

最後謹將此份論文獻給我最親愛的爸爸 陳漢南；媽媽 劉彩霜；姐姐 陳彥雯；哥哥 陳俊良，感謝您們一路上支持我，讓我可以無憂無慮地專心求學、順利完成學業，唯有你們當我精神上的後盾，我才能成就這份論文，謝謝你們！

陳冠樺 謹誌于逢甲
中華民國九 四年七月

摘 要

旅行推銷員問題(Traveling Salesman Problem, TSP)為網路型態問題之一重要類型，也是許多問題之基礎，例如VRP問題、物流配送之路線安排問題、通訊網路測試等，都是旅行推銷員問題之延伸。TSP已被證明是屬於NP-Complete,其所需之計算時間將隨節點數之增加成指數成長。因此，近幾年來開始有學者從觀測動物習性、生物演化與物理化學現象，建構出啟發式解法來求取近似解，以期能兼顧求解速度與準確性。著名之啟發式演算法包括了基因演算法(genetic algorithm,GA)、禁忌搜尋法(Tabu Search, TS)、螞蟻群聚最佳化(Ant Colony Optimization,ACO)...等。

螞蟻群聚最佳化為近年來最著名之巨集啟發式演算法之一，且已成功的被廣泛應用於組合最佳化問題。從螞蟻群聚最佳化第一個模型：螞蟻系統(Ant System, AS)發展至今已有多位學者針對AS提出不同改良之方法來提升求解品質。本研究提出一改良式螞蟻演算法，稱之為螞蟻記憶系統(Ant Memory System, AMS)以求解旅行推銷員問題。AMS最特別之處在於具有記憶功能之記憶盒，可以有效控制費落蒙更新方式。AMS利用記憶盒中最小值進行費落蒙更新將可避免一直採用目前最佳解以及一直採用疊代最佳解所帶來陷入局部解風險。

本研究採用螞蟻群聚最佳化最常測試之七題國際例題，兩種題型(對稱性 TSP 以及非對稱性 TSP)來測試 AMS，並且採用螞蟻系統(Ant System, AS)、螞蟻傑出系統(Ant System elitist, ASe)、螞蟻群落系統(Ant Colony System, ACS)、分等螞蟻系統(Rank-Based Version of Ant System, ASrank)以及極大-極小螞蟻系統(MAX-MIN Ant System, MMAS)之文獻解來比較本研究所提出螞蟻記憶系統(Ant Memory System, AMS)之求解效率及品質。在測試結果發現 AMS 在與其他演算法(除 MMAS+pts)相較下，都有大幅度的改善。

關鍵字：旅行推銷員問題、巨集啟發式演算法、螞蟻記憶系統、螞蟻演算法、螞蟻群落系統、分等螞蟻系統、極大-極小螞蟻系統。

Abstract

Traveling Salesman Problem (TSP) is an important type of network problems and the foundation of several applications such as vehicle routing problems (VRP), communication network testing, etc. Recently, animal habits, biological evolutions and physical phenomenon have inspired various heuristic algorithms that are expected to be more efficient and robust. Examples include Genetic Algorithm (GA), Tabu Search(TS), Ant Colony Optimization(ACO).

Recently, Ant Colony Optimization(ACO) is one of the most popular Meta-Heuristics and has been successfully applied to several NP-hard combinatorial optimization problems. Since Ant System was developed, there were many scholars devoted to improve the robustness and efficiency of AS. This study developed a new improved AS algorithm Ant Memory System(AMS), that has a memory box to efficiently control pheromone updating avoid local minima.

Commonly used the benchmark problems were tested in this study and the result are compared with the results from other revised algorithms. The comparisons show that AMS provides better approximation solution than the other algorithms. The study compare the performance of AMS to that obtained with Ant System(AS), Ant System elitist (ASe), Rank-Based Version of Ant System(ASrank), Ant Colony System(ACS) , MAX-MIN Ant System(MMAS). The computational results was that generally AMS achieves the best performance. The only exception is ATSP instance ft70, for which MMAS+pts has a better average performance.

Key word : Traveling Salesman Problem、Meta-Heuristic、Ant Memory System、Ant System、Ant System elitist、Rank-Based Version of Ant System、Ant Colony System、MAX-MIN Ant System.

目 錄

誌 謝	I
摘 要	II
Abstract.....	III
目 錄	IV
圖 目 錄	VII
表 目 錄	VIII
第一章 緒論.....	1
1.1 研究動機.....	1
1.2 研究目的.....	2
1.3 研究範圍.....	2
1.4 研究方法與流程.....	4
第二章 文獻回顧.....	6
2.1 旅行推銷員問題.....	6
2.1.1 TSP 之數學模式	6
2.1.2 TSP 解法之敘述	7
2.2 巨集啟發式解法.....	8
2.2.1 基因演算法基本原理及在 TSP 上之應用	9
2.2.2 模擬退火法基本原理及在 TSP 上之應用	11
2.2.3 門檻接受法基本原理及在 TSP 上之應用	12
2.2.4 禁制搜尋法基本原理及在 TSP 上之應用	15
2.3 螞蟻演算法發展之回顧.....	17
2.3.4 螞蟻演算法基本原理	17
2.3.5 螞蟻群聚最佳化	18

1.螞蟻系統.....	20
2.螞蟻群落系統.....	21
3.極大-極小螞蟻系統	22
4.分等螞蟻系統.....	24
5. 螞蟻群落最佳化結合多種螞蟻群落	24
2.4 小結.....	29
第三章 螞蟻記憶系統.....	30
3.1 螞蟻記憶系統之介紹.....	30
3.2 螞蟻記憶系統變數定義.....	30
3.3 螞蟻記憶系統運算流程.....	32
3.4 螞蟻記憶系統操作流程說明.....	33
3.5 螞蟻記憶系統環境設定.....	35
第四章 參數設定.....	36
4.1 α 、 β 、 σ 、 ρ 之參數設定.....	36
4.2 λ_0 、 λ 參數之設定.....	40
4.3 螞蟻數量之設定.....	43
4.4 記憶盒容量之設定.....	44
第五章 測試結果.....	50
5.1 螞蟻記憶系統之測式.....	50
5.2 比較其他改良式螞蟻演算之結果.....	52
第六章 結論與建議.....	56
6.1 結論.....	56
6.2 建議.....	56

參考文獻	58
附錄一、 AMS 參數組合	71



圖 目 錄

圖 1.1	研究流程圖	4
圖 2.1	常見 TSP 解法分枝說明圖	8
圖 2.2	局部解與全域最佳解之關係	9
圖 2.3	Dueck 與 Scheuer 所採用的門檻數列	14
圖 2.4	TA 與 SA 接受暫劣解機率比較	15
圖 2.5	螞蟻演算法原理	17
圖 2.6	ACO 演算法	19
圖 3.1	螞蟻記憶系統運算流程	32
圖 3.2	記憶盒操作流程(以記憶盒容量為 5).....	34
圖 4.1	kroA100 記憶盒容量為 1 時費落蒙更新之情形	46
圖 4.2	kroA100 記憶盒容量為 5 時費落蒙更新之情形	46
圖 4.3	kroA100 記憶盒容量為 10 時費落蒙更新之情形	46
圖 4.4	kroA100 記憶盒容量為 15 時費落蒙更新之情形	47
圖 4.5	kroA100 記憶盒容量為 20 時費落蒙更新之情形	47
圖 4.6	kro124p 記憶盒容量為 1 時費落蒙更新之情形	47
圖 4.7	kro124p 記憶盒容量為 5 時費落蒙更新之情形	48
圖 4.8	kro124p 記憶盒容量為 10 時費落蒙更新之情形	48
圖 4.9	kro124p 記憶盒容量為 15 時費落蒙更新之情形	48
圖 4.10	kro124p 記憶盒容量為 20 時費落蒙更新之情形	49
圖 5.1	各演算法 GAP 比較圖	54

表 目 錄

表 2.1	螞蟻群聚最佳化於 TSP 之發展	27
表 4.1	eil51 國際例題選定誤差值為 0.4%之 α 、 β 、 σ 、 ρ 參數組合	37
表 4.2	kroA100 國際例題選定誤差值為 0.75%之 α 、 β 、 σ 、 ρ 參數組合	38
表 4.3	d198 國際例題選定誤差值為 1.5%之 α 、 β 、 σ 、 ρ 參數組合	39
表 4.4	符合三題國際例題圍之 α 、 β 、 σ 、 ρ 參數組合	39
表 4.5	TSP 之 λ_0 、 λ 參數組合	41
表 4.6	ATSP 之 λ_0 、 λ 參數組合	42
表 4.7	AMS 於 TSP 問題之螞蟻數量設定	43
表 4.8	ATSP 問題螞蟻參數之設定	44
表 4.9	記憶盒容量 1、5、10 所測試之結果	45
表 4.10	記憶盒容量 15、20 所測試之結果	45
表 5.1	螞蟻記憶系統測試七個國際例題之結果	51
表 5.2	AMS、AS、ACS 測試國際例題之結果	52
表 5.3	ASrank、ASe、MMAS 測試國際例題之結果	53
表 5.4	MMAS+pts、ACOMAC、ACOMAC+NN 以及 ACOMAC+DNN 測試國際例題之結果	53
表 5.5	AMS、AS、ACS、ASrank、ASe、MMAS、MMAS+pts 測試國際例題誤差值比較	54

第一章 緒論

1.1 研究動機

旅行推銷員問題(Traveling Salesman Problem, TSP)為網路型態問題之一重要類型，也是許多問題之基礎，例如車輛巡迴問題(VRP)、物流配送之路線安排問題、生產排程...等，都是旅行推銷員問題之延伸。此外，研究旅行推銷員問題之學者也廣泛分佈在不同領域，例如：人工智慧、生物學、數學、物理以及作業研究...等，都是應用不同專業角度建構不同求解模型。

雖然TSP定義 分簡單，但是求解的搜尋空間則是非常大為 $O(TSP)=n!$ ，以問題規模30個城市為例，如果電腦一秒內可處理一百萬次交換，其需要 4×10^{18} 年來完成所有的交換。因此近年來著重於利用啟發式解法(Heuristic Algorithm)來求解此問題，以克服電腦沉長的運算時間以及電腦硬體設備限制情況，其著名之啟發式演算法包括了基因演算法(Genetic Algorithm, GA)[1,5-6,23,28]、門檻接受法(Threshold Accepting, TA)[2,4,7,13]、禁忌搜尋法(Tabu Search, TS)[6,24-26]、螞蟻群聚最佳化(Ant Colony Optimization, ACO)[19-21]...等。

螞蟻系統(Ant System, AS)為Dorigo[14-16]於1996年所提出之第一個螞蟻群聚最佳化(Ant Colony Optimization, ACO)模型並應用於旅行推銷員問題，其演算法原理是依據模仿自然界螞蟻搜尋食物時，會遺留特有的分泌物-費洛蒙(Pheromone)，以吸引其他螞蟻可以追尋先前螞蟻所探索之路徑前往。其後，Dorigo[17-18,27]、Bullnheimer[8-10]、Stützle [32-35]、Fa Tsai[22]...等學者分別在1997-2004年之間提出改善AS之模型。無論AS或其他改良AS模型都已成功的被應用在組合最佳化問題，例如：二次指派問題(Quadratic Assignment problem)[30]、資料探勘(Data Mining)[31]、空間規劃(space-planning)[12]...等。

從螞蟻系統發展至今已有多位學者針對此演算法提出改良之模

型，其目的在於研究出最優良的螞蟻模型來求解相關NP-Complete問題。有鑑於此，本研究也嘗試提出一改良式螞蟻演算法來求解旅行推銷員問題，以期能建構出一較佳模型。

1.2 研究目的

基於以上動機，本研究之研究目的如下：

1. 彙析旅行推銷員問題求解方法。
2. 本研究以其他啟發式演算法跳脫局部解其需接受較劣解之想法為基礎，建立螞蟻記憶系統(Ant Memory System,AMS)，以期望建構出一較其他改良式螞蟻演算法優良之演算法。
3. 本研究使用文獻中最常測試之七題國際例題二種不同問題型態，進行不同求解方法的效度比較，包括本研究所建立之AMS以及其他改良式螞蟻演算法：螞蟻演算法(Ant System, AS)、螞蟻群落系統(Ant Colony System, ACS)、分等螞蟻系統(Rank-Based Version of Ant System, ASrank)、極大-極小螞蟻系統(MAX-MIN Ant System, MMAS)以及螞蟻群落最佳化結合多種螞蟻群落(Ant Colony Optimizaition with multiple ant clans, ACOMAC)。
4. 以往其他啟發式演算法在求解組合最佳化問題時，都會隨著某一問題之不同而跟隨著規劃不同模式，並且有著實用上的限制。螞蟻演算法與其他演算法不同之處為其應用彈性大，只需改變求解問題之變數以及參數調整，就可廣泛應用於組合最佳化問題。因此本研究提出一較其他改良式螞蟻演算法為優之模型，以期能廣泛應用於其他組合最佳化問題。

1.3 研究範圍

本研究的研究範圍如下：

1. 目標函數值以歐基里德距離(Euc-lidean distance)為基準。
2. 本研究為配合其他改良式螞蟻演算法常測試之國際例題，測試問題

型態包含了對稱性(symmetric)以及非對稱性(asymmetric)TSP。對稱性TSP為從節點 i 到節點 j 之距離等於從節點 j 到節點 i 之距離 ($node_{ij} = node_{ji}$)。反之，非對稱性TSP為從節點 i 到節點 j 之距離不等於從節點 j 到節點 i 之距離 ($node_{ij} \neq node_{ji}$)。

3. 本研究採用其他螞蟻演算法常測試之七題國際例分別為對稱性TSP：eil51、kroA100、d198以及非對稱性TSP問題：ry48p、ft70、kro124p、ftv170，來驗證本研究所提出之模型。
4. 本研究顧及在撰寫其他改良式螞蟻演算法時可能無法完全考慮原作者之涵意，使得撰寫結果可能會與文獻解有所差異。因此，為求比較之公平性，本研究皆採用文獻解來比較不同演算法之求解品質。



1.4 研究方法與流程

本研究之研究流程如圖 1.1 所示：

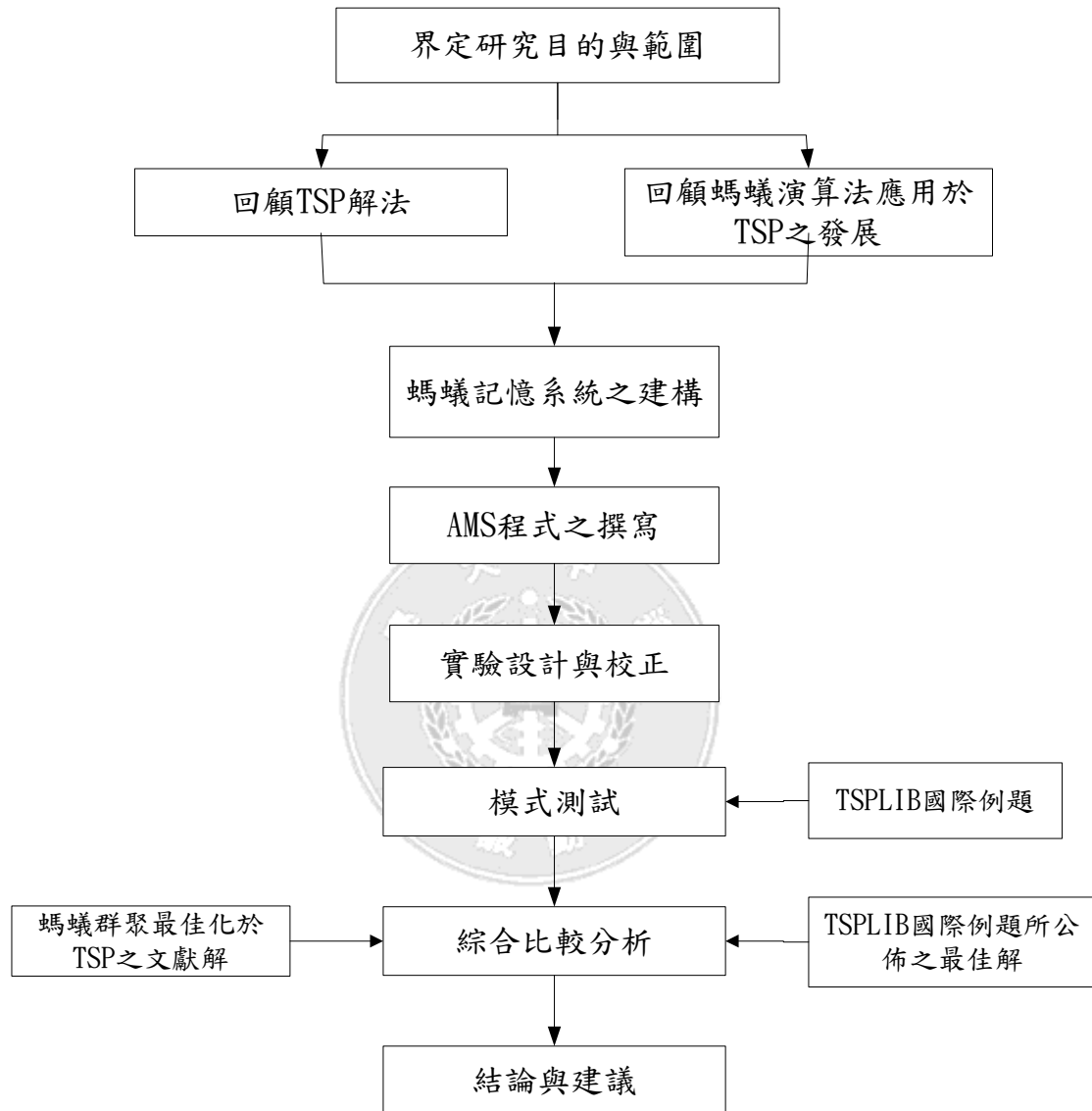


圖 1.1 研究流程圖

根據研究流程圖之內容可分述如下：

1. 界定研究目的與範圍

本研究以其他巨集啟發式演算法跳脫局部解其需接受較劣解之概念為基礎，建立螞蟻記憶系統 (Ant Memory System, AMS)，且以 TSPLIB 國際例題為基準，比較分析其他改良式螞蟻

演算法之求解效度以及TSPLIB所提供之最佳解。

2. 回獻相關文獻

回顧旅行推銷員問題(TSP)求解之方法，且詳細介紹巨集啟發式解法求解之概念以及回顧螞蟻群聚最佳化應用於TSP之發展。

3. 螞蟻記憶系統模型建構

以其他巨集啟發式演算法跳脫局部解其需接受較劣解的概念為基礎，建立出能在適當時機接受劣解，且能進而逼近全域最佳解模式，稱之為螞蟻記憶系統(Ant Memory System, AMS)。

4. 題庫之蒐集

蒐集TSPLIB國際例題、TSPLIB所提供之已知最佳解以及其他改良式螞蟻演算法所測試之結果。

5. 實驗設計與測試

此部份工作重點為針對AMS參數加以調整，以構建出一穩定模式。

6. 綜合比較分析：

將本研究所設計之AMS演算法與其他改良式螞蟻演算法，進行求解品質比較，並探討本研究所設計之演算法在未來發展的可行性和應用潛力。

7. 結論與建議

根據前述綜合比較所得之結果提出研究結論，並對本研究所建構之AMS提出建議。

第二章 文獻回顧

本章將對TSP解法做一簡單闡述並且對於近年來所興起之常見巨集啟發式(Meta-Heuristic)演算法做詳細介紹。最後，將回顧螞蟻群聚最佳化應用於TSP之發展。本章節內容安排如下：2.1旅行推銷員問題之介紹。2.2巨集啟發式解法之介紹。2.3螞蟻群聚最佳化應用於TSP之發展回顧。

2.1 旅行推銷員問題

一路網 G 中由節點 (Vertices, V) 與節線 (Edges, E) 所組成，即為 $G=(V,E)$ 。其中 $V=\{\text{該路網中所有節點(Vertices)}\}$ ， $E=\{\text{該路網中所有節線(Edges)}\}$ 。旅行推銷員問題之目標函數為在總距離最短之情況下，求得在此路網中連接所有節點之順序。

2.1.1 TSP之數學模式

本研究TSP類型之數學規劃模式如下[3]：

$$\text{Minimize } \sum_{i=1}^n \sum_{j=1}^n c_{ij} x_{ij} \dots\dots\dots(2-1)$$

Subject

$$\sum_{i=1}^n x_{ij} = 1(j = 1, \dots, n) \dots\dots\dots(2-2)$$

$$\sum_{j=1}^n x_{ij} = 1(i = 1, \dots, n) \dots\dots\dots(2-3)$$

$$X = (x_{ij}) \in S \dots\dots\dots(2-4)$$

$$x_{ij} = 0, 1 \dots\dots\dots(2-5)$$

其中，決策變數 x_{ij} 表示是否行經節線 (i, j) ，等於1時代表有節點 i 連接節點 j ，等於0時代表路網中不含 (i, j) 路線。其中 C_{ij} 代表由節點 i 至節點 j 的成本或時間， n 為節點個數。式子(2-1)為數學規劃模式之目標函式，即總路線總成本極小化。式子(2-2)、(2-3)為限制每節點只能經過一次。式子(2-5)為雙元整數限制式。至於式子(2-4)中之 s 則為避免產生子迴路的限制式， s 通常可以從下列三種形式中擇一使用：

$$S = \left\{ (x_{ij}) : \sum_{i \in Q} \sum_{j \notin Q} x_{ij} \geq 1 \text{ for every nonempty proper subset } P \text{ of } V \right\} \dots\dots\dots(2-6)$$

$$S = \left\{ (x_{ij}) : \sum_{i \in k} \sum_{j \in k} x_{ij} < |k| \text{ for every nonempty subset } R \text{ of } R \text{ of } V \setminus 1 \right\} \dots\dots\dots(2-7)$$

$$S = \left\{ (x_{ij}) : y_i - y_j + N \cdot x_{ij} \leq N - 1, \text{ for } 2 \leq i \neq j \leq N, y_i \text{ 與 } y_j \text{ 皆為實數} \right\} \dots\dots\dots(2-8)$$

式子(2-6)為在TSP之解集合X集合中，每個節點子集合Q內至少有一節點與Q外的某一節點相連，以避免產生子迴路。式子(2-7)中， $|k|$ 為節點集合V之非空子集合k的節點個數。由於k中所有節點若要形成一個迴路，需要 $|k|$ 條節線，因此式(2-7)限制k中的節線個數不得超過 $|k|-1$ 條，以避免子迴路的產生。式子(2-8)則需要 $N^2 - 3N + 2$ 條限制式，其中 $y_i = t$ ，表示節點 i 在TSP路線的第 t 站被經過。

2.1.2 TSP解法之敘述

求解TSP之方式可分為最佳解解法(exact solution)以及近似解解法(approximation)兩大類。最佳解是追求正確解，其缺點為沒有效率執行。近似解是追求有效率且能逼近正確解為目標，當問題特性不同時，所設計出之啟發式解法便有所差異。下圖為常見最佳解解法以及近似解解法之演算法。

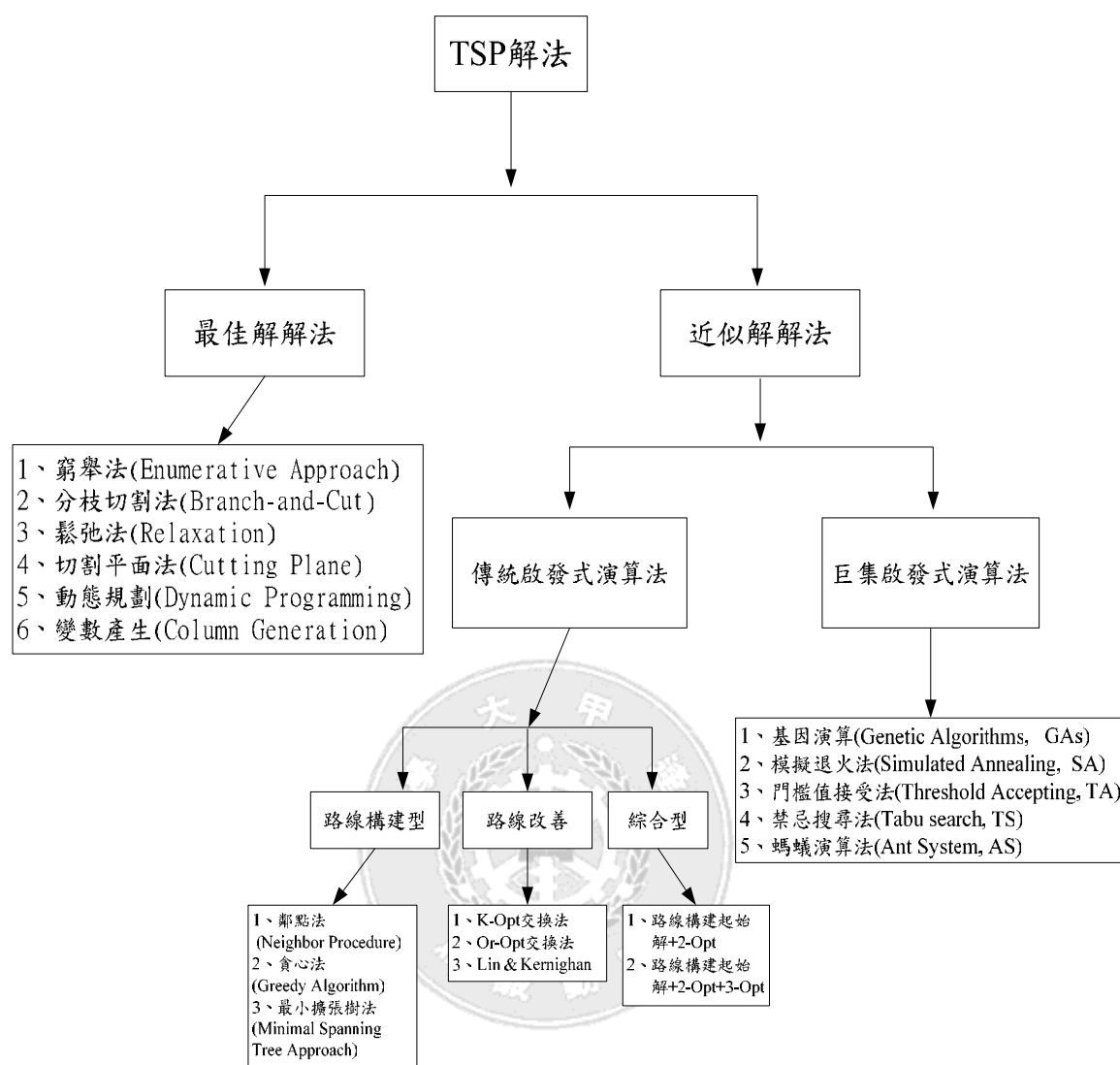


圖 2.1 常見 TSP 解法分枝說明圖

由圖2.1所示，近似解解法又可分為傳統啟發式解法與巨集啟發式解法，而單用傳統啟發式解法求解TSP已被證實會陷入局部解，所以近年來才有巨集演算法之興起。因此下一節本研究將針對近年來所興起之常見巨集啟發式解法應用於TSP做一簡單之介紹。

2.2 巨集啟發式解法(Meta-Heuristic)

一般學者在求解最小化問題時都會企圖嘗試尋找最小值，而傳統啟發式解法經常會有陷入局部最佳解而無法跳脫之窘境，因此如欲得到全域最佳解之前題下就是必須跳脫局部解。以圖2.5為例，A點為局

部解，C點為全域最佳解，如要從A點到C點則必須經過B點，而其中最好之想法就是在適當之情況下接受較劣解，以致有機會達到全域最佳解。

巨集啟發式解法其發展重點就是著重於如何跳脫局部解，一般巨集啟發式解法有兩種方式跳脫局部解分別為機率尋優法以及接受劣解法。

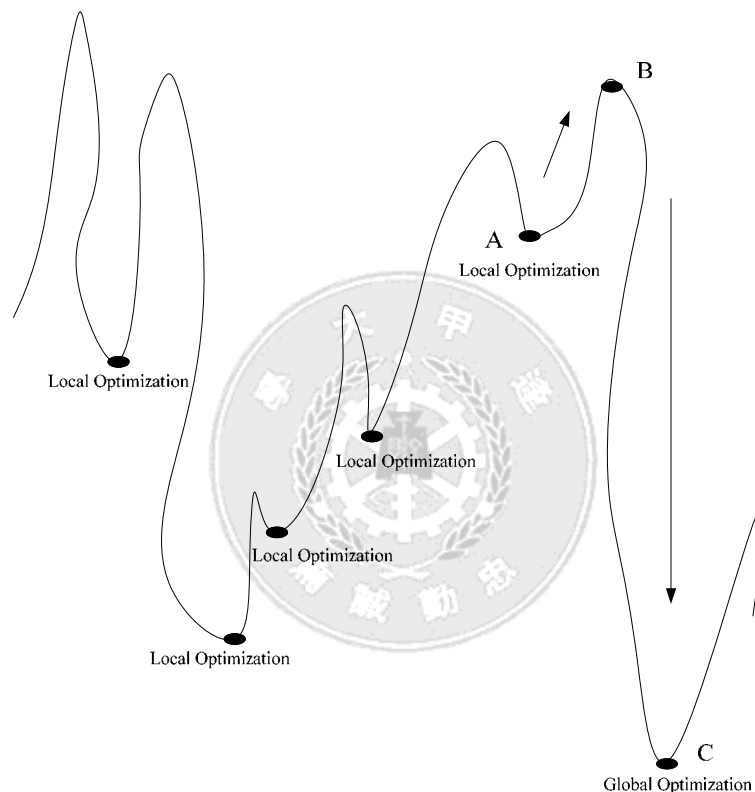


圖 2.2 局部解與全域最佳解之關係

2.2.1 基因演算法基本原理及在TSP上之應用

基因演算法 (Genetic Algorithm, GA)[1,5-6,23,28] 最早是由 Holland[28]於1975年所研究出來。其基本理論是模擬達爾文所提出之「物競天擇，適者生存」的生物進化搜尋法則為基根，選擇族群中環境適應能力較強之個體，當作繁殖下一代之種子，經過複製、交配、突變等演化過程，產生新的下一代；如此反覆進行，最後得到適應環

境最強的下一代。

基因演算法之基本單位為染色體(chromosome)，而染色體則由許多遺傳基因(genes)組成，每個基因對應著此物種某些特性。因此，染色體會因遺傳基因組合方式不同，而對環境產生不同適應度。基因演算法以適者生存的法則，讓環境適應性越強之染色體產生越多子代，運用特殊交配方式交換優良的基因片段，並以突變方式增加世代間的變異性，最後獲得對環境適應性最佳之物種。

基因演算法主要三個遺傳法則如下：

(1)選擇或稱複製(Selection)

從族群中選出染色體予以交配，且適合度較高之染色體被選中用來產生下一代的機率亦較高，故使得較「優良」之遺傳因子得以流傳與繼承。

(2)交配(Crossover)

交配乃是將經過複製過程篩選出之成對染色體以一設定機率(交配率)決定是否進行配對，再經由彼此間所進行之基因交換行為產生子代。

(3)突變(Mutation)

突變乃是針對子代之遺傳基因以一定機率，更動某一基因值，以防止染色體於複製及交配過程中，落入局部最佳解，亦即預防族群內各染色體僵化。

GA的操作步驟如下(求解最小化問題)：

步驟 1：產生初始族群：利用亂數產生器或其他方式產生初始族群，以作為演化基礎。

步驟 2：評估：計算族群中每一條染色體之適合度值。如果該染色體為不可行解，一般會降低其適合度值。

步驟 3：選擇：隨機選擇成對染色體，如符合交配率則進行交配產生子代，否則將置回母代族群。

步驟4：交配：將兩母代染色體利用隨機產生之交配點進行交配，以產生子代。計算子代的適合度值後，置回族群。

步驟五：突變：隨機改變子代某一基因值，且計算突變後子代適合度。

步驟六：測試停止條件：測試是否符合停止條件。如已達停止條件，完成尋優動作，若否則回至步驟三。

早期GA應用範圍多在人工智慧領域，從1985年開始Brady[23]才首次將GA應用於組合最佳化TSP問題。從GA發展至今，已被證明為一有效最佳化搜尋方法，且目前已有數種改良式的衍生方法被發展出來。將GA技術應用於TSP時，染色體將被視為求解TSP之任一可行路線，且在求解過程中必須先確定下列事項：(1)路線表現方式。(2)初始解產生方式。(3)基本三個步驟：複製、交配、突變。

2.2.2 模擬退火法基本原理及在TSP上之應用

模擬退火法(Simulation Annealing, SA)[6,11,29]其基本觀念最早是由Metropolis等人於1953年所提出。SA原理源自當固體加熱至一定溫度後，會由固態結構瓦解變為液態結構，當完全冷卻時能重新排列成預期的穩定晶體型態。以求解最小化問題為例，目標值往較低處移動之情況可視為物體降溫結晶的過程，最後結晶型態即為一個最佳解。SA最大特色即是在尋優過程中若發生某一狀態較前一狀態惡化時，SA會以一設定機率決定是否接受，而不是直接放棄。

以求解最小化問題之前題下，在尋優過程中如果K狀態高於前一之狀態(K-1)時，則SA會以下列公式產生一機率值

$$P(\Delta E) = e^{-\Delta E / T_k}$$

$P(\Delta E)$ ：是否接受K狀態之機率

ΔE : K狀態之目標值 - K-1狀態之目標值

T_k : K狀態時之溫度

此時，利用亂數產生器產生一個介於0,1之數值R，以判斷是否接受該狀態。當 $R > P(\Delta E)$ 時，則放棄狀態K。反之，當 $R \leq P(\Delta E)$ 時，則接受狀態K。因此， $P(\Delta E)$ 值愈高，狀態K被接受的機率也愈高。

SA的操作步驟如下(求解最小化問題)：

步驟 1：產生起始狀態：產生一個起始可行解 x_0 、狀態值 $f(x_0)$ 、起始溫度 T_0 。令最佳狀態 $x_b = x_0$ 、最佳狀態值 $f(x_b) = f(x_0)$ 、現有溫度 $T_c = T_0$ 、尋優次數 $k=0$ 。

步驟 2：執行鄰域搜尋(傳統節線交換法)。

步驟 3：判斷是否接受劣解。當 $f(x_k) \leq f(x_b)$ 時，令 $x_b = x_k$ 、 $f(x_b) = f(x_k)$ 。
當 $f(x_k) > f(x_b)$ 時，則利用亂數產生器產生一個介於0,1之數值R，以判斷是否接受該狀態。若 $R > P(\Delta E)$ 則放棄狀態K。若 $R \leq P(\Delta E)$ 則接受狀態K $x_b = x_k$ 、 $f(x_b) = f(x_k)$ 。

步驟 4：執行降溫：令 $T_{k+1} = \alpha T_k$ 或 $T_{k+1} = T_k - \delta$ (α 、 δ 為一係數)。

步驟 5：測試停止條件：(1)運算次數已達設定遞迴次數。(2)最佳狀態為 x_b 及 $f(x_b)$ 。

Kirkpatrick[29]將SA應用在對稱性的TSP問題上，成功的解出了20*20方格矩陣。

2.2.3 門檻接受法基本原理及在TSP上之應用

門檻接受法 (Threshold Accepting, TA)[2,4,7,13] 為 Dueck 與 Scheuer[13]於1990年所發表的一個新啟發式解題架構。門檻接受法(TA)與SA類似，當搜尋的暫時解成本 $C(k)$ 比現有解成本 $C(k-1)$ 較大時，仍有被接受更新的機會。然而TA與SA不同處在於TA的接受法則是確定性而非機率性，若令 T_k 為時間k之接受門檻值，則TA搜尋求解

的基本程序如下：

$$\begin{array}{ll} \text{接受機率}=1 & \text{若 } C(k) - C(k-1) \leq T_k \\ \text{接受機率}=0 & \text{若 } C(k) - C(k-1) > T_k \end{array}$$

門檻數列 $(T_k)_{k=0}^K$ 是由非負實數組成的非遞增數列。由於TA是個改善起始解之方法論，因此在應用時必須先有一個起解，然後建立一個門檻數列。TA將會根據門檻數列決定是否接受劣解並依據門檻數列決定何時終止。當舊的解轉換到新的解時必須要有一個交換法，此交換法可以是傳統的交換法的任何一種方法。一般而言，核心方法(節線交換法)通常是執行效率快、效果良好的方法(一般使用2-Opt)。

門檻數列之決定與下列三個因素有關：起始門檻 (T_0) 、數列長度 (k) 、以及數列的收斂形態。

1. 起始門檻 (T_0) 設定

Dueck與Scheuer[13]在提出TA時並未說明如何設定起始門檻，其所採用之起始門檻值約最佳成本的0.25%。

2. 門檻數列遞減形態

TA之門檻數列必須由起始門檻 (T_0) 逐漸減至0，而門檻數列長度及門檻數列遞減形態會影響求解精確度以及速度。Dueck與Scheuer[13]將TA解求節點數442的TSP時，採用一個不規則遞減門檻數列形態；起始門檻值為0.13；門檻數列的前三分之一是屬於等差遞減形態；中間約占二分之一的部分呈現門檻值重複後再遞減數列；末段又呈現等差遞減狀態，其所採用的門檻數列，如圖2.6所示。

3. 數列長度 (k)

門檻數列長度是指門檻數列之中有幾個門檻值，門檻數列長度與起始門檻、門檻數列遞減形態有關，並且會影響求解精確度

及運算時間。Dueck與Scheuer[13]所使用門檻數列長度為 $k=30$ 。

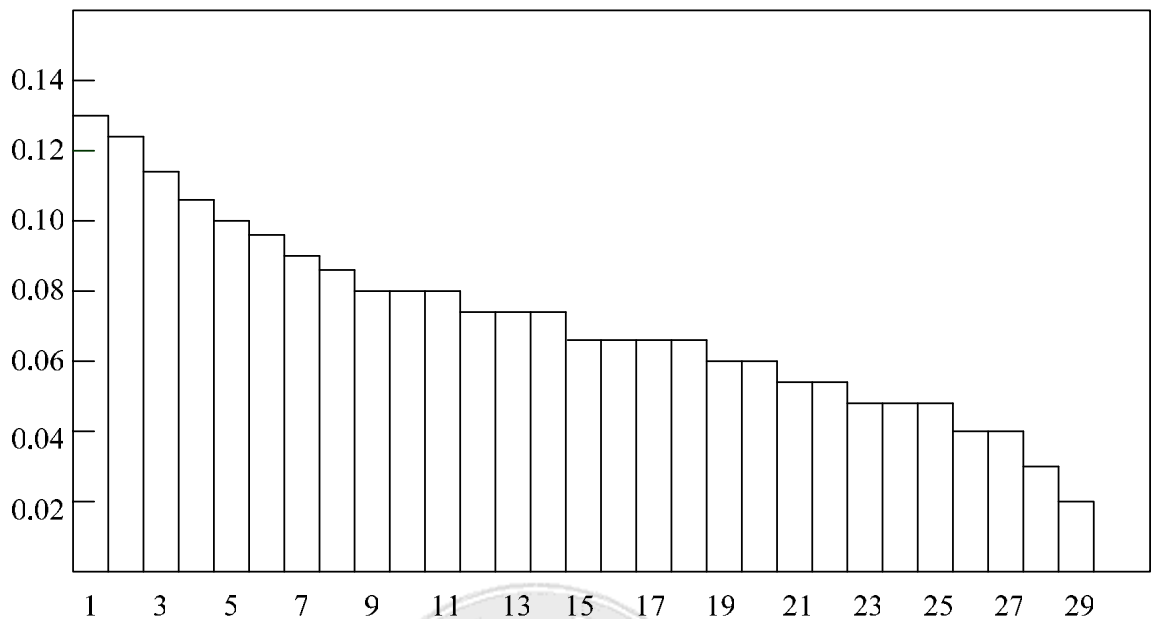


圖 2.3 Dueck 與 Scheuer 所採用的門檻數列

TA的操作步驟如下(求解最小化問題)：

步驟 1：產生起始狀態：產生一個起始可行解 x_0 ，狀態值 $f(x_0)$ ，設定起始門檻值 T_k 。並令最佳狀態 $x_b = x_0$ ，最佳狀態值 $f(x_b) = f(x_0)$ 。

步驟 2：選定核心交換法(例如2-Opt、3-Opt)。

步驟 3：判斷是否接受劣解。當 $f(x_k) \leq f(x_b)$ 時，令 $x_b = x_k$ 、 $f(x_b) = f(x_k)$ 。
當 $f(x_k) > f(x_b)$ 時，則判斷是否接受劣解。若 $f(x_k) - f(x_{k-1}) < T_k$ 則接受劣解，即 $x_b = x_k$ 、 $f(x_b) = f(x_k)$ 。

步驟 4：執行降低門檻值：若 $f(x_k) \leq f(x_b)$ 則降低門檻值並令 $x_b = x_k$ 、 $f(x_b) = f(x_k)$ 。

步驟 5：測試停止條件：檢查門檻數列是否已用盡。若非符合停止條件則繼續執行步驟二。

在起始解與交換法方面，TA與SA的架構皆相似，而其中最大不

同之處在於其接受法則。SA是根據一個與其溫度序列有關的機率決定是否接受成本較差解，而TA 是根據門檻數列中門檻值大小來決定是否接受成本較差解。SA與TA接受劣解不同之處如圖2.7所示。

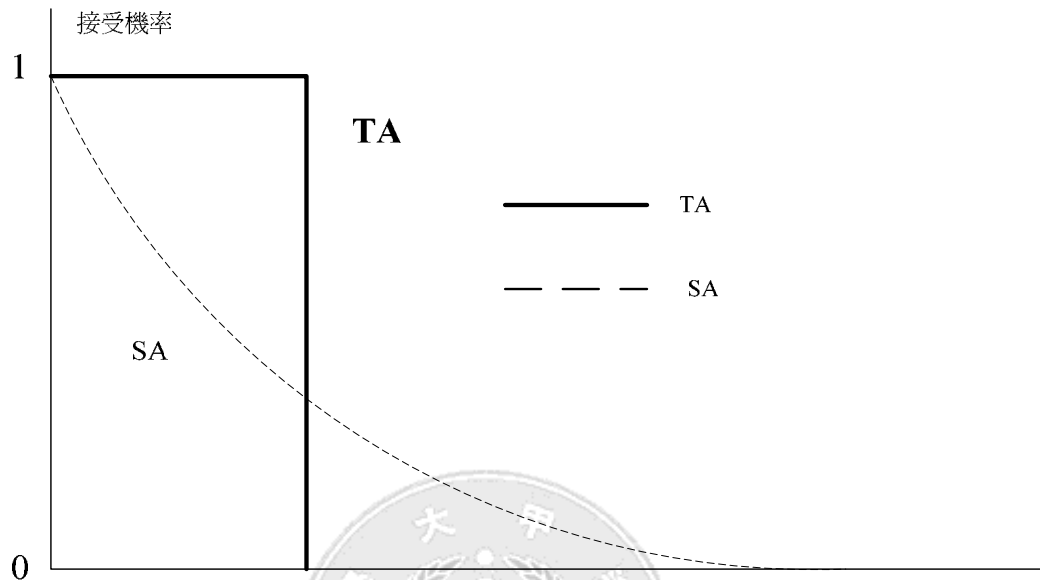


圖 2.4 TA 與 SA 接受暫劣解機率比較

2.2.4 禁制搜尋法基本原理及在TSP上之應用

禁制搜尋法(Tabu Search, TS) [6,24-26]的觀念最早由Glover[24]所提出，其方法利用目前解在鄰域進行搜尋，並應用人工智慧之記憶機制，將已經搜尋過的解記錄於「禁制名單(Tabu List)」，以避免重覆搜尋或毫無目的搜尋情況發生，以提昇搜尋之效率。在禁制搜尋法中，主要包含移步(Move)、禁制名單(Tabu List)、渴望準則(AspirationLevel)、候選名單(Candidate List)以及搜尋停止準則(Stopping Criterion)五大模組。以下將針對五大模組做介紹：

1. 移步(Move)

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

2. 候選名單(Candidate List)

候選名單即為可以進行移步屬性的集合，候選名單內的每一個被允許的移步皆對應一鄰解。禁制搜尋法中之移步，即是由這些名單中挑選最好鄰解，進行移步動作。

3. 禁制名單(Tabu List)

在進行移步同時，禁忌名單會將其移步屬性紀錄下來，以作為未來進行移步時參考依據，而禁忌名單大小，將決定該移步被禁制的期間。當禁制期間愈長時，愈能避免在求解過程中可能發生的循環，因而較不容易陷入區域最佳解。

4. 渴望準則(Aspiration Level)

當某一移步被禁制時，其所獲得之解優於目前搜尋的最佳解時（設定為當目標函數值獲得改善），則可透過渴望準則將其禁制狀態解除，使該移步可以進行。

5. 搜尋停止準則(Stopping Criterion)

搜尋停止準則通常分為下列幾種：(1)預設運算過程中，最大允許遞迴數 (2)預設目標函數值持續未改善之最大允許遞迴數。(3)預設允許CPU 計算時間。(4)預設達到可接受目標函數值範圍，一旦達到預設範圍即停止搜尋，而所搜尋到之最佳尋獲解即為最佳解。

TS的操作步驟如下(求解最小化問題)：

步驟 1：設定起始狀態：由其他方法找到一起始解 x_0 ，並令 x_0 為目前最佳解 x^* ，目標值為 $f(x^*)$ 。設定起始遞迴次數 $k=0$ ，禁忌名單 $T = \phi$ 。

步驟 2：移步：選取能使目標值最低之鄰近解 x_k 。如果鄰近解集合與禁止名單中(T 集合)相同時則跳至步驟四。

步驟 3：更新：當 $f(x_k) < f(x^*)$ 時，則目前最佳解由 x_k 所取代，並列入禁忌名單中(T 集合)。反之，當 $f(x_k) > f(x^*)$ 時，也需列入禁忌名單中(T 集合)。

步驟 4：測試停止條件：若符合搜尋停止準則測試停止。

2.3 螞蟻演算法發展之回顧

2.3.4 螞蟻演算法基本原理

螞蟻演算法最早是由 Dorigo[14-16]在觀察真正螞蟻時所得到靈感，Dorigo[14-16]發現螞蟻可以在食物和巢穴間找到最短路徑。因此，就將此想法應用於旅行推銷員問題。當螞蟻在覓食時，在走過的路徑上會留下一種化學物質，稱之為費洛蒙，這些化學物質就像是留下路標一樣，讓後來的螞蟻沿著有費洛蒙路徑行走，因而形成了費洛蒙路徑。當費洛蒙強度愈強，則吸引螞蟻前來的能力就愈大。下圖為 Dorigo[16]在觀察真正螞蟻後，所模擬出的螞蟻行為，也說明著其演算法之原理。

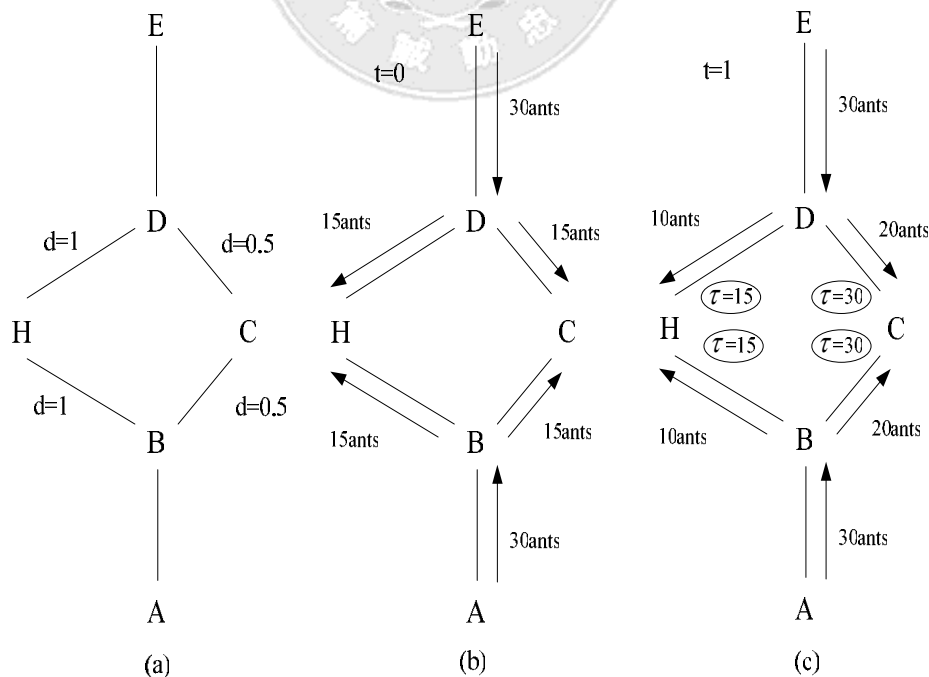


圖 2.5 螞蟻演算法原理

以圖 2.8 為例，假設螞蟻每單位時間 $t(t=1,2,\dots)$ 行走距離 1($d=1$)，每隻螞蟻在時間 t 時，行走過之路徑會留下費落蒙數量為 1($\tau=1$)。圖 2.8-(a)中 H-D 與 H-B 中的 $d=1$ 代表路徑長度為 1。同理，C-D 與 C-B 中的 $d=0.5$ 代表路長度為 0.5。圖 2.8-(b)為 $t=0$ 時，放置 30 隻螞蟻從 A 到 E，同時也放置 30 隻螞蟻從 E 到 A。當 30 隻螞蟻到達 B 處時，因無先前螞蟻所留下之費落蒙數量，所以螞蟻在選擇 B-H 與 B-C 的機率同時為 $1/2$ ，這時會有 15 隻螞蟻通往 B-H 路徑；15 隻螞蟻往 B-C 路徑前進。同理 30 隻螞蟻從 E 出發遇 D 處時，也將會有 15 隻螞蟻往 D-H 路徑；15 隻螞蟻往 D-C 路徑。當 t 介於 0 與 1 時，因路徑 B-H-D 長度為 2($d=2$)，B-C-D 長度為 1($d=1$)，這代表著當螞蟻在每單位時間走完 B-C-D 時，B-C 與 D-C 有 30 隻螞蟻走過，留下費落蒙數量為 30($\tau=30$)。而另一路徑 B-H-D 所行走之螞蟻，在一單位時間結束時，不管由 B 或 D 處出發之螞蟻，此時只能到達 H，所以 B-H 與 D-H 路徑只有 15 隻螞蟻走過，留下費落蒙數量為 15($\tau=15$)。圖 2.8-(c) $t=1$ 時，又在 A 與 E 處放置 30 隻螞蟻，當螞蟻到達 B 時，螞蟻將會追隨先前螞蟻所留下之費落蒙而有選擇機率之發生。選擇 B-C 之機率為 $\frac{\tau_{BC}(30)}{\tau_{BC}(30)+\tau_{BH}(15)} = \frac{2}{3}$ ，選擇 B-H 之機率為 $\frac{\tau_{BC}(15)}{\tau_{BC}(30)+\tau_{BH}(15)} = \frac{1}{3}$ ，因此會有 $20(30 \times \frac{2}{3})$ 隻螞蟻往 B-C 前進， $10(30 \times \frac{1}{3})$ 隻螞蟻往 B-H 前進。同理，選擇路徑 D-C 之機率為 $\frac{2}{3}$ ，選擇路徑 D-H 之機率為 $\frac{1}{3}$ ，有 20 隻螞蟻往路徑 D-C 前進，10 隻螞蟻往路徑 D-H 前進。隨著單位時 t 一直增長，最後螞蟻將會一直往 B-C-D 前進。

2.3.5 螞蟻群聚最佳化

螞蟻群聚最佳化(Ant Colony Optimization,ACO)[19-21]為巨集演算法之一，且已成功的解決很多組合最佳的問題(Hard Combinatorial Optimization Problems)。在螞蟻群聚最佳化中所含括之演算法除傳統螞蟻系統(Ant System, AS)外，也包含其他由螞蟻系統所進化之演算法，例如：ACS[17-18,27]、ASrank[8-10]、MMAS [32-35]...等，其

概念都是仿效真實螞蟻的行為模式求解組合最佳化問題。在螞蟻群聚最佳化(ACO)中所包含之演算法皆有一致性之主體架構，其架構敘述如下[19-21]：

Procedure ACO algorithm for TSP

Set parameters, initialize pheromone trails。

While(termination condition not met)**do**

Construct Solutions

Apply Local Search #optional

Pheromone update

End

end ACO algorithm for TSP

圖 2.6 ACO 演算法

- 建構解(Construct Solutions)

此部份主要功能為提供螞蟻選擇下一節點之機率。當每隻螞蟻選擇完所有節點時，即建構出一個解。

- 費落蒙更新(Pheromone update)

螞蟻在走過的路徑中，選擇比較好的路徑增加較多費落蒙數量。且每一路徑上之費落蒙會隨時間蒸發掉一部份。

- 鄰近搜尋(Apply Local Search)

有些改良式螞蟻演算法之設計會加入鄰近搜尋法以減少計算時間及得到更好的解。Stützle[35]提及將機率建構解(probabilistic solution construcion)結合鄰近搜尋，稱之為混合式演算法(hybrid algorithms)。

下一小節將介紹ACO求解TSP問題之演算法。首先將介紹通用之參數：

$p_{ij}^k(t)$:在 t 疊代時，第 k 隻螞蟻由節點 i 到節點 j 之機率。

$\tau_{ij}(t)$ ：在第 t 次疊代時，節點 i 與節點 j 之間所累積的費落蒙數量。

η_{ij} ：節點 i 到節點 j 距離之倒數。

$J_k(i)$ ：第 k 隻螞蟻在節點 i 時未走過節點之集合。

α 、 β ：分別為費落蒙資訊與距離倒數(η_{ij})之重要參數。

δ 、 ρ ：蒸發係數，即費落蒙數量會隨每次疊代而有所揮發。

$s^{ib}(t)$ ：第 t 疊代所得到之最佳解。

$s^{gb}(t)$ ：為至目前 t 疊代以前所得到之最佳解。

q ：由均一分配所求得之亂數值

q_0 ：決定螞蟻路徑選取方法之參數。

Q ：影響費落蒙之參數。

1. 螞蟻系統

螞蟻系統(Ant System, AS) [14-16]為 ACO 中最早被提出之最佳化模型且成功應用於旅行員銷售問題(Traveling Salesman Problem)。

➤ 建構解(solution construction)：

$$p_{ij}^k(t) = \left\{ \begin{array}{ll} \frac{[\tau_{ij}(t)]^\alpha * [\eta_{ij}]^\beta}{\sum_{u \in J_k(i)} [\tau_{iu}(t)]^\alpha * [\eta_{iu}]^\beta}, & \text{if } j \in J_k(i) \\ 0, & \text{otherwise} \end{array} \right\}$$

當螞蟻從節點 i 選擇下一節點 j 時， $[\tau_{ij}(t)]^\alpha$ 乘以 $[\eta_{ij}]^\beta$ 之值愈大，則路徑 ij 被選擇之機率就會愈大。

➤ 費落蒙更新(Pheromone update)

$$\tau_{ij}(t) = (1 - \delta) * \tau_{ij}(t) + \sum_{k=1}^m \Delta \tau_{ij}^k$$

$$\Delta \tau_{ij}^k = \begin{cases} \frac{Q}{L_k}, & \text{if } k\text{th ant uses edge } ij \text{ in its tour} \\ 0, & \text{otherwise} \end{cases}$$

在完成建構解後，將螞蟻走過之路徑予以費落蒙更新即為 $\sum_{k=1}^m \Delta \tau_{ij}^k$ ， L_k 為第 k 隻螞蟻所得到之解。此外，Dorigo[16]也嘗試使用最傑出之螞蟻 ($s^{gb}(t)$) 進行費落蒙更新，稱之為螞蟻系統傑出策略 (Ant System elitist, ASe)。

2. 螞蟻群落系統

螞蟻群落系統 (Ant Colony System, ACS) [17-18,27] 為 Dorigo 改良先前所提之螞蟻系統[14-16]。其演算法流程如下：

➤ 建構解(solution construction)：

當 $q \leq q_0$

$$p_{ij}^k(t) = \begin{cases} 1 & \text{if } j = \arg \max_{u \in J_k(i)} [\tau_{iu}(t)]^\alpha [\eta_{iu}]^\beta \\ 0 & \text{otherwise} \end{cases}$$

當 $q > q_0$

$$p_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha * [\eta_{ij}]^\beta}{\sum_{u \in J_k(i)} [\tau_{iu}(t)]^\alpha * [\eta_{iu}]^\beta}, & \text{if } j \in J_k(i) \\ 0, & \text{otherwise} \end{cases}$$

當 $q \leq q_0$ 時，螞蟻從節點 i 選擇下一節點 j 時，螞蟻將選擇 $[\tau_{ij}(t)]^\alpha$ 乘以 $[\eta_{ij}]^\beta$ 之值最大的 ij 路徑。反之，當 $q > q_0$ 時， $[\tau_{ij}(t)]^\alpha$ 乘以 $[\eta_{ij}]^\beta$ 之值愈大，則路徑 ij 被選擇之機率就愈大。

➤ 費落蒙更新(Pheromone update)

此更新方式又分為局部費落蒙更新以及全域費落蒙更新二種方式。局部費落蒙更新為當第 k 隻螞蟻完成下一節點選擇時，對走過之節點進行更新。全域費落蒙更新為當每隻螞蟻走完所有節點時，選擇最優之螞蟻進行更新。

● 局部費落蒙更新(Local Pheromone update)

$$\tau_{ij}(t) = (1 - \delta) * \tau_{ij}(t) + \delta * \Delta \tau_{ij}$$

$$\Delta \tau_{ij} = (L_m * n)^{-1}$$

局部費落蒙更新主要之目的為降低螞蟻所走過路徑之費落蒙數量，使其他螞蟻不會遵循同一路線前進。

● 全域費落蒙更新(Global Pheromone update)

$$\tau_{ij}(t) = (1 - \rho) * \tau_{ij}(t) + \rho * \Delta \tau_{ij}$$

$$\Delta \tau_{ij} = \begin{cases} (s^{gb}(t))^{-1}, & \text{if } ij \in \text{global-best-tour} \\ 0, & \text{otherwise} \end{cases}$$

利用全域更新將至目前疊代為止最好螞蟻所走之路徑增加費落蒙數量。

3. 極大-極小螞蟻系統

極大-極小螞蟻系統 (MAX-MIN Ant System, MMAS) [32-35]想法為將每條路徑之費落蒙數量控制在一範圍內。當 $\tau_{ij} > \tau_{\max}$ 時，則令

$\tau_{ij} = \tau_{\max}$ 。反之，當 $\tau_{ij} < \tau_{\max}$ 時，則令 $\tau_{ij} = \tau_{\max}$ 。此演算法之設計其目的在於讓螞蟻在搜尋解時不會過早收斂在某個範圍內。

➤ 建構解(solution construction)：

從節點 i 到節點 j ，如 j 屬於 i 最近之 20 節點內，則使用下式：

$$p_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha * [\eta_{ij}]^\beta}{\sum_{u \in J_k(i)} [\tau_{iu}(t)]^\alpha * [\eta_{iu}]^\beta}, & \text{if } j \in J_k(i) \\ 0, & \text{otherwise} \end{cases}$$

反之，下一節點之選擇不屬於最近 20 節點內時，則使用 $\max [\tau_{ij}(t)]^\alpha [\eta_{ij}]^\beta$ 。

➤ 費落蒙更新(Pheromone update)

使用疊代最佳解增加費落蒙數量。並且將所有路徑上之費落蒙數量限制在 $[\tau_{\min}, \tau_{\max}]$ 範圍內。其費落蒙更新方式如下：

$$\tau_{ij}(t+1) = \rho \cdot \tau_{ij}(t) + \Delta \tau_{ij}^{best}$$

$$\Delta \tau_{ij}^{best} = \begin{cases} (s^{ib}(t))^{-1}, & \text{if } (i, j) \in \text{iteration} - \text{best} - \text{tour} \\ 0, & \text{otherwise} \end{cases}$$

Stützle[35]提及 pts(pheromone trail smoothing) 主要之目的為增加尋搜之機率。其公式為

$$\tau_{ij}^*(t) = \tau_{ij}(t) + \delta(\tau_{\max}(t) - \tau_{ij}(t))$$

$\tau_{ij}^*(t)$ 為加入 pts 後， $\tau_{ij}(t)$ 為加入 pts 前

4. 分等螞蟻系統

分等螞蟻系統(Rank-Based Version of Ant System, ASrank)[8-10]演算法之設計為當每個疊代結束後，先將 m 隻螞蟻所得求之值依大小排序。然後對第 $w = \sigma - 1$ 隻螞蟻排列編號(ranking index)以前之螞蟻進行費落蒙更新，且排在愈前面之螞蟻所增加的費落蒙數量就愈多。

➤ 建構解(solution construction)：

$$p_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha * [\eta_{ij}]^\beta}{\sum_{u \in J_k(i)} [\tau_{iu}(t)]^\alpha * [\eta_{iu}]^\beta}, & \text{if } j \in J_k(i) \\ 0, & \text{otherwise} \end{cases}$$

當螞蟻從節點 i 選擇下一節點 j 時， $[\tau_{ij}(t)]^\alpha$ 乘以 $[\eta_{ij}]^\beta$ 之值愈大，則路徑 ij 被選擇之機率就會愈大。

➤ 費落蒙更新(Pheromone update)

$$\tau_{ij}(t+1) = \rho * \tau_{ij}(t) + \Delta\tau_{ij} + \Delta\tau_{ij}^*$$

$$\Delta\tau_{ij}^\mu = \begin{cases} (\sigma - \mu) * Q / L_\mu & \text{if the } \mu\text{-th best ant travels on edge}(i, j) \\ 0 & \text{otherwise} \end{cases}$$

$$\Delta\tau_{ij}^* = \begin{cases} \sigma * Q / s^{gb} & \text{if edge } (i, j) \text{ is part of the best solution} \\ 0 & \text{otherwise} \end{cases}$$

μ ：螞蟻排列編號。 $\Delta\tau_{ij}^\mu$ ：第 μ 隻螞蟻在經過路徑 ij 所增加之費落蒙數量。 L_μ ：第 μ 隻螞蟻所走之路徑長度。 $\Delta\tau_{ij}^*$ ：至目前疊代為止之最佳解所增加的費落蒙數量。 σ ：傑出螞蟻數量。

5. 螞蟻群落最佳化結合多種螞蟻群落

螞蟻群落最佳化結合多種螞蟻群落(Ant Colony Optimization with

Multiple Ant Clans, ACOMAS)[22]採用平行基因演算法(Parallel Genetic Algorithm, PGA)之觀念,搜尋不同且獨立之求解空間以避免螞蟻在搜尋時陷入局部解。ACOMAS[22]設計之想法為將螞蟻分成數個群落,除群落有自行費落蒙更新方式外,且群落與群落之間也有其溝通方式(費落蒙更新)。

➤ 建構解(solution construction)

當 $q \leq q_0$

$$p_{ij}^k(t) = \begin{cases} 1 & \text{if } j = \arg \max_{u \in J_k(i)} [\tau_{iu}(t)]^\alpha [\eta_{iu}]^\beta \\ 0 & \text{otherwise} \end{cases}$$

當 $q > q_0$

$$p_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha * [\eta_{ij}]^\beta}{\sum_{u \in J_k(i)} [\tau_{iu}(t)]^\alpha * [\eta_{iu}]^\beta}, & \text{if } j \in J_k(i) \\ 0, & \text{otherwise} \end{cases}$$

當 $q \leq q_0$ 時,螞蟻從節點 i 選擇下一節點 j 時,螞蟻將選擇 $[\tau_{ij}(t)]^\alpha$ 乘以 $[\eta_{ij}]^\beta$ 之值最大的 ij 路徑。反之,當 $q > q_0$ 時, $[\tau_{ij}(t)]^\alpha$ 乘以 $[\eta_{ij}]^\beta$ 之值愈大,則路徑 ij 被選擇之機率就愈大。

➤ 費落蒙更新(Pheromone update)

除使用ACS [17-18,27]之局部費落蒙與全域費落蒙更新外,群落之間還有不同之溝通方式,其溝通方式如下所示:

$$\begin{cases} Clan_i = Clan_k * (1 - w) + Clan_i * w, & \text{if } i = 1 \\ Clan_i = Clan_{i-1} * (1 - w) + Clan_i * w, & \text{otherwise} \end{cases}$$

i 為第幾個群落, k 為群落數目, w 為權重值

Fa Tsai[22]除使用 ACOMAS 外，還加入最近鄰近法(nearest neighbor, NN)以及對偶最近鄰近法(Dual nearest neighbor approach, DNN)兩種初始費落蒙路徑方法。最近鄰近法(nearest neighbor, NN)為利用多種最近鄰近法(Multiple nearest neighbor approach)將螞蟻放置不同節點以建構出螞蟻初始費落蒙路徑。DNN(Dual nearest neighbor approach)為隨機選擇一點設定為起始點，然後再選擇該起始點之最遠距離的節點為另一起點，最後再跟據這兩個起始節點選擇最近節點進行增加費落蒙之數量動作以建構出螞蟻初始費落蒙路徑。

茲將上述之改良式螞蟻演算法應用於TSP的相關研究彙整如表 2.1 所示。



表 2.1 螞蟻群聚最佳化於 TSP 之發展

作者	演算法名稱	建構解	費落蒙更新	測試資料來源	測試結果
Dorigo[16]	AS	使用機率選擇模式	對螞蟻所走過之路徑皆加入費落蒙	使用 Oliver30 測試一個小時	AS 較 TA 及 SA 兩者優良
Dorigo and Gambardel [17]	ACS	使用機率選擇模式並且加入轉換規則	利用區域費落蒙更新以及全域費落蒙更新	1、設計出 50、75 以及 100 個節點 2、TSPLIB 國際例題	ACS 較 GA、EP 以及 SA 優良。
Dorigo and Gambardel [18]	ACS	使用機率選擇模式並且加入轉換規則	利用區域費落蒙更新以及全域費落蒙更新	1、設計出 50 節點 2、TSPLIB 國際例題	ACS 較 SA、EN、SOM、F1、GA、EP、AG 優良
Bullnheim, Hartl and Strauss[8]	ASrank	使用機率選擇模式	將 m 隻螞蟻所得求之值依大小排序，然後對第 w 隻螞蟻排列編號以前之螞蟻做費落蒙更新	1、使用 Oliver30 2、使用產業界真實案例	ASrank 較 SA、SAnn、GA、AS、ASe 優良
Stützle, Hoos[33]	MMAS	使用機率選擇模式	限制費落蒙路徑之數量	1、TSPLIB 國際例題 2、使用 ACS 求解 TSPLIB 國際例題之文獻解	MMAS 較 ACS 優良

表 2.1(續) 螞蟻群聚最佳化於 TSP 之發展

作者	演算法名稱	建構解	費落蒙更新	測試資料來源	測試結果
Stutzle, Hoos[35]	MMAS、MMAS+pts	當選擇下一節點為最近之 20 節點內時使用機率選擇模式。反之，不在 20 節點內時，使用轉換規則	限制費落蒙路徑數量以及加入 pts	1、TSPLIB 國際例題。 2、使用 ACS 求解 TSPLIB 國際例題之文獻解	MMAS+pts 較 ACS、ASrank、ASe、AS MMAS 優良
Fa Tsai, Wei Tsai, Chang Tseng.[22]	ACOMAS	使用機率選擇模式並且加入轉換規則	除使用區域費落蒙更新以及全域更新法外，還將螞蟻分成數個群落，除群落有自行之費落更新方式，且群落與群落之間也有費落蒙更新方式	5. TSPLIB 國際例題。	ACOMAS 較 ACS 優良

2.4 小結

1. 巨集啟發式解法其發展重點就是著重於如何跳脫局部解，進而求得全域最佳解。常見跳出局部解之方法為接受劣解以及隨機尋優。
2. 由螞蟻演算法應用於TSP之發展可發現從最早改良式螞蟻演算法 ASe到ACOMAS，都是從費落蒙更新方式進行改良。
3. 因此本研究提出能有效控制費落蒙更新且能適時接受劣解之演算法，稱之為螞蟻記憶系統(Ant Memory System,AMS)。



第三章 螞蟻記憶系統

本章節將介紹本研究所提出之螞蟻記憶系統(Ant Memory System, AMS)。本章節內容安排如下：3.1 螞蟻記憶系統之介紹。3.2 螞蟻記憶系統變數定義。3.3 螞蟻記憶系統運算流程。3.4 螞蟻記憶系統操作流程說明。3.5 螞蟻記憶系統環境設定。

3.1 螞蟻記憶系統之介紹

螞蟻記憶系統(Ant Memory System, AMS)最主要之功能為能有效控制螞蟻對於費落蒙的更新方式。螞蟻透過記憶功能進行費落蒙更新，可避免陷入局部解，進而達到最佳解。AMS 之記憶方式，即是利用記憶盒記憶每次疊代最佳解，然後再對記憶盒中最小值進行費落更新，其目的可避免螞蟻一直接受至 t 疊代為止所得到之最佳解進行費落蒙更新而有陷入局部解的危險，且利用此記憶功能將有效控制使用疊代最佳解之更新方式，相較於一直使用疊代最佳解進行更新時，可能會因每個疊代值差異太大，而對較差的路徑進行費落更新。在記憶盒之設計方面，主要有兩種功能：

- 具更新功能。當記憶容量大小已不能容下時，將會把最早之疊代最佳解剔除，更新並儲存最新疊代最佳解。
- 具選擇性。如果疊代最佳解已存在記憶盒裡，將不更新記憶。此功能有效防止螞蟻對某一最佳解之偏好，而陷入局部解。

3.2 螞蟻記憶系統變數定義

t ：第 t 次疊代。

$s^{gb}(t-1)$ ： *global-best-solution*，至目前 $t-1$ 疊代為止，所得到之最佳解。

$s^{ib}(t)$ ： *iteration-best-solution*，第 t 次疊代完成時，該疊代所得到之最佳解。

$p_{ij}^k(t)$ ：第 t 次疊代時，第 k 隻螞蟻從節點 i 選擇下一節點 j 所發生之機率。

$\eta_{ij}(t)$ ：第 t 次疊代時，路徑 ij 所累積之費落蒙數量。

η_{ij} ：距離 ij (d_{ij}) 之倒數，即為 $1/d_{ij}$ 。

$u \in J_k(i)$ ：第 k 隻螞蟻在節點 i 時，未走過節點之集合。

m ：螞蟻數目。

n ：節點數目。

M_s ：記憶盒容量大小。

$M_i (i \in 1.2....M_s)$ ：記憶盒資料位置。

M^b ：記憶盒中最小值。

σ ：區域費落蒙更新之蒸發係數。

ρ ：記憶盒費落蒙更新之蒸發係數。

q ：由均一分配所求得之亂數值

q_0 ：決定螞蟻路徑選取方法之參數。

L_{nn} ：最佳鄰近解值。

τ_0 ：費落初始值， $\tau_0 = 1/(n * L_{nn})$ 。

α 、 β ：影響費洛蒙路徑強度 (τ) 與距離倒數 (η) 之重要性參數值。

λ_0 、 λ ：分別為當 $s^{gb}(t-1) > s^{ib}(t)$ 時影響增加費落蒙數量之參數以

及當 $s^{gb}(t-1) \leq s^{ib}(t)$ 時影響增加費落蒙數量之參數。

3.3 螞蟻記憶系統運算流程

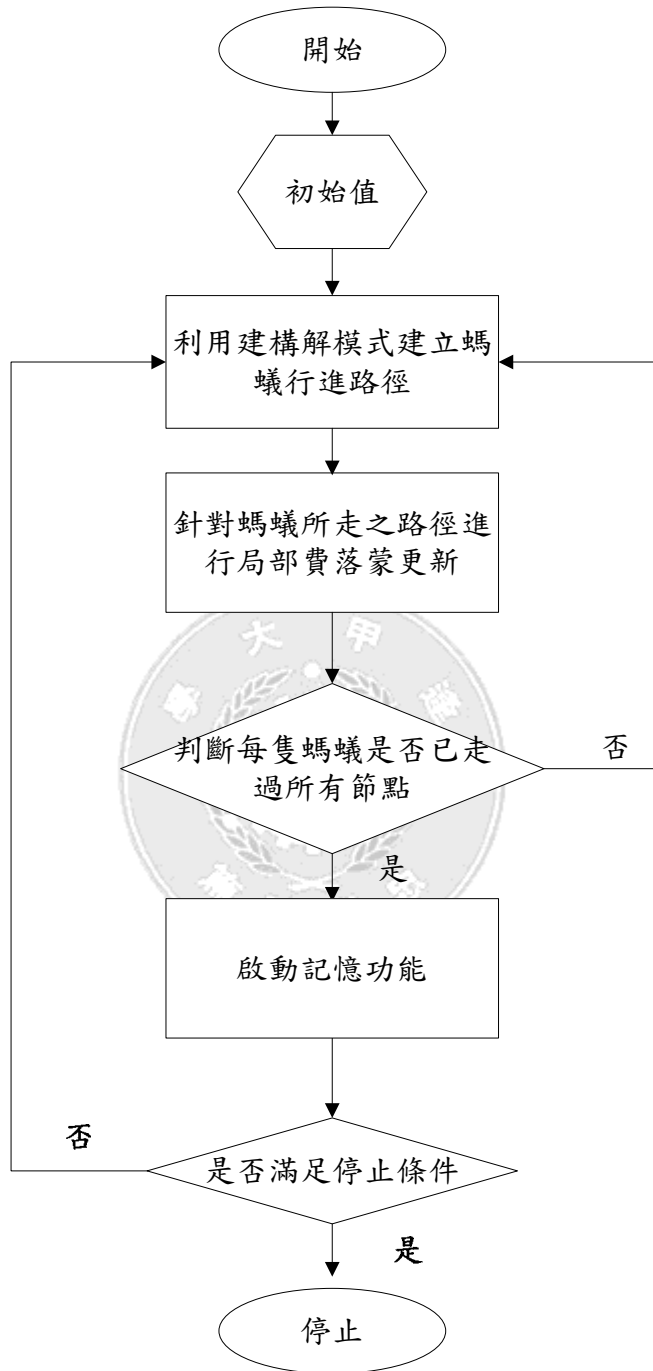


圖 3.1 螞蟻記憶系統運算流程

3.4 螞蟻記憶系統操作流程說明

本小節將對本研究所提出之螞蟻記憶系統做作流程做說明

步驟 1：初始值設定(Initialize setting)

包含螞蟻數(m)、記憶容量大小(M_s)、區域和記憶盒費落蒙蒸發係數(σ 、 ρ)、影響費洛蒙路徑強度(τ)以及距離倒數(η)之重要性參數值(α 、 β)、決定螞蟻路徑選取方法之參數($q_0=0.9$)、影響增加費落蒙數量之參數(λ_0 、 λ)、初始費落蒙設定($\tau_0=1/(n*L_m)$)、初始解(L_m)。

步驟 2：建構解(Construct Solutions)

當 $q \leq q_0$

$$p_{ij}^k(t) = \begin{cases} 1 & \text{if } j = \arg \max_{u \in J_k(i)} [\tau_{iu}(t)]^\alpha [\eta_{iu}]^\beta \\ 0 & \text{otherwise} \end{cases}$$

當 $q > q_0$

$$p_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha * [\eta_{ij}]^\beta}{\sum_{u \in J_k(i)} [\tau_{iu}(t)]^\alpha * [\eta_{iu}]^\beta}, & \text{if } j \in J_k(i) \\ 0, & \text{otherwise} \end{cases}$$

當 $q \leq q_0$ 時，螞蟻從節點 i 選擇下一節點 j 時，螞蟻將選擇 $[\tau_{ij}(t)]^\alpha$ 乘以 $[\eta_{ij}]^\beta$ 之值最大的 ij 路徑。反之，當 $q > q_0$ 時， $[\tau_{ij}(t)]^\alpha$ 乘以 $[\eta_{ij}]^\beta$ 之值愈大，則路徑 ij 被選擇之機率就愈大。

步驟3：局部費落蒙更新(Local Pheromone update)

$$\tau_{ij}(t) = (1 - \sigma) * \tau_{ij}(t) + \sigma * \Delta \tau_{ij}$$

$$\Delta\tau_{ij} = (L_{nn} * n)^{-1}$$

局部費落蒙更新主要之目的為降低螞蟻所走過路徑之費落蒙數量，使其他螞蟻不會遵循同一路線前進。

步驟4: 啟動記憶功能(Operate Memory Function)

記憶盒操作可分成二個階段，第一階段主要為記憶盒內資料之初始化，第二階段主要為當記憶盒容量容納不下時所採取之動作。其操作流程如圖3.2，記憶盒容量以5為例。

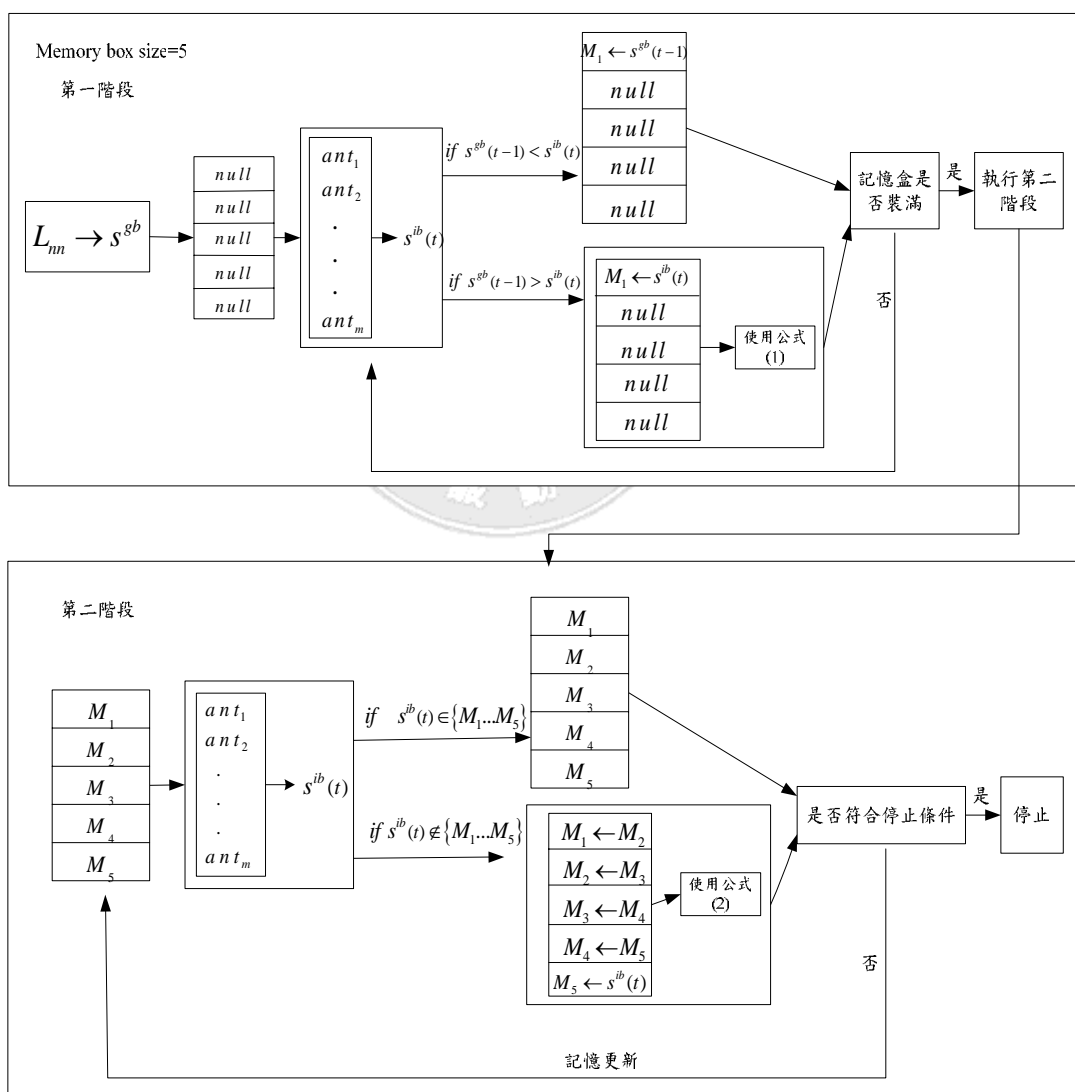


圖 3.2 記憶盒操作流程(記憶盒容量為 5)

圖 3.2 之第一階段說明了當記憶盒容量未滿時，如果 t 疊代之最佳解 $s^{ib}(t)$ 大於 $s^{gb}(t-1)$ 時，則將 $s^{gb}(t-1)$ 值放入記憶盒中，且不做任何費落蒙更新動作。反之，如果 t 疊代之最佳解 $s^{ib}(t)$ 小於 $s^{gb}(t-1)$ 時，除了將 $s^{ib}(t)$ 值放入記憶盒內，且使用公式(1)進行費落蒙更新。

當記憶盒容量已滿時將進行第二階段，如果 t 疊代之最佳解 $s^{ib}(t)$ 已出現在記憶盒中時，則不進行任何費落蒙更新動作。如果 t 疊代之最佳解 $s^{ib}(t)$ 不曾出現在記憶盒中，則將 t 疊代之最佳解 $s^{ib}(t)$ 放入記憶盒中，且將最早進入記憶中之資料剔除，然後對記憶盒之最小值進行費落蒙更新，執行式子(2)。

➤ 記憶盒初始化

$$\tau_{ij} = (1-\rho) * \tau_{ij} + \rho * \Delta\tau_{ij} \dots\dots\dots(1)$$

$$\Delta\tau_{ij} = \begin{cases} \lambda_0 * (M^b)^{-1}, & i, j \in M^b \\ 0, & otherwise \end{cases}$$

➤ 記憶盒容量已滿

$$\tau_{ij} = (1-\rho) * \tau_{ij} + \rho * \Delta\tau_{ij} \dots\dots\dots(2)$$

$$\Delta\tau_{ij} = \begin{cases} \lambda_0 * (M^b)^{-1}, & \text{if } s^{gb}(t-1) > s^{ib}(t) \& i, j \in M^b \\ \lambda * (M^b)^{-1}, & \text{if } s^{gb}(t-1) < s^{ib}(t) \& i, j \in M^b \end{cases}$$

3.5 螞蟻記憶系統環境設定

本研究所有之測試環境皆採用中央處理器 (CPU)Pentium4 2.4GHz，記憶體(Ram)採用 DDR 512MB 單插孔，軟體方面作業系統屬於 Microsoft 2000。參數設定以及測試結果皆採用 Java2 程式語言撰寫。在結果比較部份，本研究採用誤差率準則為 $GAP\% = (\text{演算法解} - \text{已知最佳解(opt)}) / \text{最佳解} * \%)$ 。

第四章 參數設定

本章將對本研究所提出之 AMS 做參數設定，以期能得到最好參數組合。首先，本研究將對直接影響建構解模式之費落蒙數量與距離倒數之參數以及費落蒙蒸發係數 α 、 β 、 σ 、 ρ 一起討論，之後對影響增加費落蒙數量之參數 λ_0 、 λ 一起進行探討，然後再加以探討會影響搜尋空間之螞蟻數量(m)，最後將探討記憶盒容量之大小對接受劣解的影響。本研究在測試參數皆採用 5000 次疊代，執行 10 次之平均值。本章節內容安排如下：4.1 α 、 β 、 σ 、 ρ 之參數設定。4.2 λ_0 、 λ 參數之設定。4.3 螞蟻數量之設定。4.4 記憶盒容量之設定。

4.1 α 、 β 、 σ 、 ρ 之參數設定

α 、 β 為影響建構解模式之重要參數， σ 、 ρ 為費落蒙蒸發係數，其四參數並不受 TSP 以及 ATSP 問題型態而有直接之影響，因此本研究將直接利用 TSP 所測試之參數直接延用至 ATSP 問題。本研究將選定其他改良式螞蟻演算常測試之 TSP 國際例題：eil51、kroA100 以及 d198 三題進行測試。設定 α 值界於 1~3、 β 值界於 2~5、 σ 與 ρ 值界於 0.01~0.20、 $\lambda_0 = 2, \lambda = 1$ 、螞蟻數量為 10、記憶容量為 5。每一題將嘗試 300 種參數組合，詳細請參閱附錄一。在判定最好之參數組合方面，首先，將視參數組合之數量以及誤差值大小來界定每一題可被允許之誤差值，然後再選定三題國際例題都表現較優者為考慮之對象。eil51 使用誤差值在 0.4% 範圍之參數組合有 60 種，如表 4.1 所示。kroA100 使用誤差值在 0.75% 範圍之參數有 34 種組合，如表 4.2 所示。因 d198 無差值在 1% 之參數組合，因此選用使用誤值在 1.5% 範圍之參數有 9 種組合。表 4.1-4.3 中之陰影代表三題國際例題都存在之參數組合。

表 4.1 eil51 國際例題選定誤差值為 0.4% 之 α 、 β 、 σ 、 ρ 參數組合

$\alpha = 1, \beta = 3$	$\sigma = 0.01, \rho = 0.20$	427.3	$\alpha = 1, \beta = 4$	$\sigma = 0.05, \rho = 0.10$	427.3
	$\sigma = 0.05, \rho = 0.05$	426.9		$\sigma = 0.05, \rho = 0.15$	427.1
	$\sigma = 0.05, \rho = 0.10$	426.7		$\sigma = 0.05, \rho = 0.20$	426.9
	$\sigma = 0.05, \rho = 0.15$	427.1		$\sigma = 0.10, \rho = 0.15$	427.6
	$\sigma = 0.10, \rho = 0.10$	427.2		$\sigma = 0.10, \rho = 0.20$	426.6
	$\sigma = 0.10, \rho = 0.15$	427.1	$\alpha = 2, \beta = 2$	$\sigma = 0.05, \rho = 0.20$	427.5
	$\sigma = 0.10, \rho = 0.20$	426.9	$\alpha = 2, \beta = 4$	$\sigma = 0.05, \rho = 0.05$	426.9
	$\sigma = 0.15, \rho = 0.20$	427.1		$\sigma = 0.05, \rho = 0.10$	427.2
$\alpha = 1, \beta = 5$	$\sigma = 0.05, \rho = 0.15$	426.5	$\alpha = 2, \beta = 4$	$\sigma = 0.05, \rho = 0.15$	427.4
	$\sigma = 0.05, \rho = 0.20$	426.8		$\sigma = 0.05, \rho = 0.20$	427.4
$\alpha = 2, \beta = 3$	$\sigma = 0.05, \rho = 0.05$	426.8		$\sigma = 0.10, \rho = 0.05$	427.2
	$\sigma = 0.05, \rho = 0.10$	427.0		$\sigma = 0.10, \rho = 0.10$	426.8
	$\sigma = 0.05, \rho = 0.20$	427		$\sigma = 0.10, \rho = 0.15$	426.5
	$\sigma = 0.10, \rho = 0.05$	427.4		$\sigma = 0.10, \rho = 0.20$	426.9
	$\sigma = 0.10, \rho = 0.15$	427.4		$\sigma = 0.15, \rho = 0.10$	427.6
	$\sigma = 0.10, \rho = 0.20$	427.3	$\alpha = 3, \beta = 2$	$\sigma = 0.05, \rho = 0.01$	427.3
$\alpha = 2, \beta = 5$	$\sigma = 0.05, \rho = 0.05$	426.7	$\alpha = 3, \beta = 4$	$\sigma = 0.05, \rho = 0.20$	427.3
	$\sigma = 0.05, \rho = 0.10$	426.6		$\sigma = 0.05, \rho = 0.05$	427.4
	$\sigma = 0.05, \rho = 0.15$	427.3		$\sigma = 0.05, \rho = 0.10$	427.5
	$\sigma = 0.05, \rho = 0.20$	427.6		$\sigma = 0.05, \rho = 0.20$	427.7
	$\sigma = 0.10, \rho = 0.10$	426.8		$\sigma = 0.10, \rho = 0.10$	427.2
	$\sigma = 0.10, \rho = 0.15$	426.4		$\sigma = 0.10, \rho = 0.20$	427
	$\sigma = 0.10, \rho = 0.20$	426.8		$\sigma = 0.15, \rho = 0.05$	427.4
	$\sigma = 0.15, \rho = 0.10$	427.2	$\alpha = 3, \beta = 5$	$\sigma = 0.05, \rho = 0.05$	427.5
	$\sigma = 0.15, \rho = 0.15$	426.9		$\sigma = 0.05, \rho = 0.10$	426.7
$\alpha = 3, \beta = 3$	$\sigma = 0.05, \rho = 0.01$	427.7		$\sigma = 0.05, \rho = 0.15$	427.2
	$\sigma = 0.05, \rho = 0.05$	426.8		$\sigma = 0.05, \rho = 0.20$	427.5
	$\sigma = 0.10, \rho = 0.05$	426.9		$\sigma = 0.10, \rho = 0.05$	427
	$\sigma = 0.10, \rho = 0.15$	427.2		$\sigma = 0.10, \rho = 0.10$	427.3
	$\sigma = 0.10, \rho = 0.20$	427		$\sigma = 0.10, \rho = 0.20$	427.6

表 4.2 kroA100 國際例題選定誤差值為 0.75% 之 α 、 β 、 σ 、 ρ 參數組合

$\alpha=1, \beta=2$	$\sigma = 0.01, \rho = 0.15$	21383.3		$\sigma = 0.01, \rho = 0.01$	21384.5
	$\sigma = 0.01, \rho = 0.20$	21401		$\sigma = 0.01, \rho = 0.05$	21414.5
$\alpha = 1, \beta = 3$	$\sigma = 0.01, \rho = 0.05$	21389.2	$\alpha = 3, \beta = 3$	$\sigma = 0.05, \rho = 0.01$	21414.5
	$\sigma = 0.01, \rho = 0.10$	21372.1		$\sigma = 0.05, \rho = 0.05$	21410.1
	$\sigma = 0.01, \rho = 0.15$	21387.8		$\sigma = 0.05, \rho = 0.10$	21433.1
	$\sigma = 0.01, \rho = 0.20$	21362.8		$\sigma = 0.05, \rho = 0.15$	21384.7
	$\sigma = 0.05, \rho = 0.15$	21420.4		$\sigma = 0.05, \rho = 0.20$	21407.8
	$\sigma = 0.05, \rho = 0.20$	21429.6		$\sigma = 0.01, \rho = 0.01$	21388.9
$\alpha = 1, \beta = 4$	$\sigma = 0.01, \rho = 0.20$	21417.3	$\alpha = 3, \beta = 4$	$\sigma = 0.05, \rho = 0.05$	21435.3
	$\sigma = 0.05, \rho = 0.20$	21433.5		$\sigma = 0.05, \rho = 0.10$	21423.5
$\alpha = 2, \beta = 2$	$\sigma = 0.05, \rho = 0.15$	21424		$\sigma = 0.05, \rho = 0.15$	21431.2
$\alpha = 2, \beta = 4$	$\sigma = 0.01, \rho = 0.01$	16053.8		$\sigma = 0.05, \rho = 0.20$	21400.4
	$\sigma = 0.05, \rho = 0.05$	21430.2	$\alpha = 3, \beta = 5$	$\sigma = 0.01, \rho = 0.01$	21398.1
	$\sigma = 0.05, \rho = 0.10$	21420.5	$\alpha = 3, \beta = 2$	$\sigma = 0.01, \rho = 0.01$	21415.1
	$\sigma = 0.05, \rho = 0.15$	21422.9	$\alpha = 2, \beta = 3$	$\sigma = 0.01, \rho = 0.01$	21386.7
	$\sigma = 0.05, \rho = 0.20$	21432.7		$\sigma = 0.05, \rho = 0.10$	21413.4
				$\sigma = 0.05, \rho = 0.15$	21376
				$\sigma = 0.05, \rho = 0.20$	21368

表 4.3 d198 國際例題選定誤差值為 1.5%之 α 、 β 、 σ 、 ρ 參數組合

$\alpha = 1, \beta = 3$	$\sigma = 0.01, \rho = 0.15$	16003.9
	$\sigma = 0.01, \rho = 0.20$	15994.2
$\alpha = 2, \beta = 2$	$\sigma = 0.05, \rho = 0.05$	15984.6
	$\sigma = 0.05, \rho = 0.15$	16015
$\alpha = 2, \beta = 3$	$\sigma = 0.01, \rho = 0.01$	16009.2
	$\sigma = 0.05, \rho = 0.05$	15998.1
	$\sigma = 0.05, \rho = 0.10$	15994.8
	$\sigma = 0.05, \rho = 0.15$	15998.7
$\alpha = 2, \beta = 4$	$\sigma = 0.05, \rho = 0.05$	16011.3

表 4.4 符合三題國際例題圖之 α 、 β 、 σ 、 ρ 參數組合

國際例題	已知最佳解	參數組合		平均值	GAP(%)
Eil51	426	$\alpha = 1, \beta = 3$	$\sigma = 0.01, \rho = 0.20$	427.3	0.28%
		$\alpha = 2, \beta = 3$	$\sigma = 0.05, \rho = 0.10$	427.1	0.26%
		$\alpha = 2, \beta = 4$	$\sigma = 0.05, \rho = 0.05$	426.9	0.21%
kroA100	21282	$\alpha = 1, \beta = 3$	$\sigma = 0.01, \rho = 0.20$	21362.8	0.38%
		$\alpha = 2, \beta = 3$	$\sigma = 0.05, \rho = 0.10$	21413.4	0.62%
		$\alpha = 2, \beta = 4$	$\sigma = 0.05, \rho = 0.05$	21430.2	0.70%
d198	15780	$\alpha = 1, \beta = 3$	$\sigma = 0.01, \rho = 0.20$	15994.2	1.36%
		$\alpha = 2, \beta = 3$	$\sigma = 0.05, \rho = 0.10$	15994.8	1.36%
		$\alpha = 2, \beta = 4$	$\sigma = 0.05, \rho = 0.05$	16011.3	1.47%

由表 4.1-4.3 之參數組合表，整理出 eil51、kroA100、d198 三題都存在之參數組合，其參數組合為 $\alpha = 1, \beta = 3, \sigma = 0.01, \rho = 0.20$ 、 $\alpha = 2, \beta = 3, \sigma = 0.01, \rho = 0.20$ 、 $\alpha = 2, \beta = 4, \sigma = 0.05, \rho = 0.05$ ，如表 4.4 所示。當 $\alpha = 1, \beta = 3, \sigma = 0.01, \rho = 0.20$ 時，有 kroA100 以及 d198 二題國際例較其他兩參數組合優良，且在 kroA100 國際例題所得之誤差值遠

遠優於其他兩者。因此本研究將 $\alpha, \beta, \sigma, \rho$ 參數分別設定為 $\alpha = 1, \beta = 3, \sigma = 0.01, \rho = 0.20$ 。

4.2 λ_0 、 λ 參數之設定

本小節將針對影響增加費落蒙之參數，包括了當 $s^{gb}(t-1) > s^{ib}(t)$ 時，影響記憶盒最小值增加費落蒙數量之參數 λ_0 ；當 $s^{gb}(t-1) \leq s^{ib}(t)$ 時，影響記憶盒最小值所增加費落蒙數量之參數 λ 。費落蒙路徑之數量將影響螞蟻尋優動作，因此 λ_0 、 λ 將是影響螞蟻尋找最佳解之重要參數。TSP與ATSP為兩種不同問題型態。TSP問題之節線 ij 距離等於 ji 距離，其 ij 費落蒙數量也等於 ji 費落蒙數量。而ATSP問題之節線 ij 距離不等於節線 ji 距離，其 ij 節線費落蒙數量也不等於 ji 路徑費落蒙數量。因此， λ_0 、 λ 兩參數設定將針對兩種問題型態進行探討。 $\alpha, \beta, \sigma, \rho$ 值分別採用上一節測試之結果 $\alpha = 1$ 、 $\beta = 3$ 、 $\sigma = 0.01$ 、 $\rho = 0.20$ ，螞蟻數量為10，記憶盒容量為5。由表4.5之測試結果可看出在使用TSP型態時， $\lambda_0 = 2, \lambda = 1$ 所得之結果為最好。由表4.6之測試結果可看出在使用ATSP型態時， $\lambda_0 = 2, \lambda = 2$ 所得之結果為最好。

表 4.5 TSP 之 λ_0 、 λ 參數組合

例題名稱	eil51		kroA100		d198	
已知最佳解	426		21282		15780	
$\lambda_0 = 1, \lambda = 1$	429.4	0.80%	21386.8	0.49%	16008	1.44%
$\lambda_0 = 1, \lambda = 2$	428.7	0.63%	21417.1	0.63%	16053.8	1.74%
$\lambda_0 = 1, \lambda = 3$	429	0.70%	21353.9	0.34%	16034.3	1.61%
$\lambda_0 = 1, \lambda = 4$	428.3	0.54%	21378.6	0.45%	15960.1	1.14%
$\lambda_0 = 1, \lambda = 5$	429.4	0.80%	21418.3	0.64%	15994	1.36%
$\lambda_0 = 2, \lambda = 1$	427.4	0.33%	21337.2	0.26%	15992.9	1.35%
$\lambda_0 = 2, \lambda = 2$	429.3	0.77%	21429.2	0.69%	15991.5	1.34%
$\lambda_0 = 2, \lambda = 3$	428.8	0.66%	21400.8	0.56%	15992.4	1.35%
$\lambda_0 = 2, \lambda = 4$	428.2	0.52%	21467.5	0.87%	15995	1.36%
$\lambda_0 = 2, \lambda = 5$	431.9	1.38%	21402	0.56%	16039	1.64%
$\lambda_0 = 3, \lambda = 1$	429	0.70%	21404.9	0.58%	16043.6	1.67%
$\lambda_0 = 3, \lambda = 2$	427.7	0.40%	21483.4	0.95%	16009.9	1.46%
$\lambda_0 = 3, \lambda = 3$	428.3	0.54%	21431	0.70%	16001.2	1.40%
$\lambda_0 = 3, \lambda = 4$	428.5	0.59%	21422.6	0.66%	15994.5	1.36%
$\lambda_0 = 3, \lambda = 5$	427.8	0.42%	21419.8	0.65%	16062.9	1.79%
$\lambda_0 = 4, \lambda = 1$	427.6	0.38%	21423.3	0.66%	16009	1.45%
$\lambda_0 = 4, \lambda = 2$	427.5	0.35%	21487.6	0.97%	16007.4	1.44%
$\lambda_0 = 4, \lambda = 3$	430.6	1.08%	21365.7	0.39%	16030.2	1.59%
$\lambda_0 = 4, \lambda = 4$	428.3	0.54%	21359.3	0.36%	16026.9	1.56%
$\lambda_0 = 4, \lambda = 5$	432.3	1.48%	21449.1	0.79%	16051	1.72%
$\lambda_0 = 5, \lambda = 1$	428.9	0.68%	21396	0.54%	16017.3	1.50%
$\lambda_0 = 5, \lambda = 2$	429	0.70%	21430.3	0.70%	16033.5	1.61%
$\lambda_0 = 5, \lambda = 3$	429	0.70%	21349.5	0.32%	16093.8	1.99%
$\lambda_0 = 5, \lambda = 4$	428.6	0.61%	21464	0.86%	16039	1.64%
$\lambda_0 = 5, \lambda = 5$	428.2	0.52%	21459.6	0.83%	16036.3	1.62%

表 4.6 ATSP 之 λ_0 、 λ 參數組合

例題名稱	ry48p		ft70		kro124p	
已知最佳解	14422		38673		36230	
$\lambda_0 = 1, \lambda = 1$	14590	1.16%	39855	3.06%	38084.4	5.12%
$\lambda_0 = 1, \lambda = 2$	14636.4	1.49%	39757.4	2.80%	38259.6	5.60%
$\lambda_0 = 1, \lambda = 3$	14546.7	0.86%	39694.6	2.64%	38458.7	6.15%
$\lambda_0 = 1, \lambda = 4$	14790.3	2.55%	39920.8	3.23%	38279	5.66%
$\lambda_0 = 1, \lambda = 5$	14557.9	0.94%	39984.2	3.39%	38329.3	5.79%
$\lambda_0 = 2, \lambda = 1$	14552.3	0.90%	39956.9	3.32%	38955	7.52%
$\lambda_0 = 2, \lambda = 2$	14543	0.84%	39663.5	2.56%	37864.3	4.51%
$\lambda_0 = 2, \lambda = 3$	14552	0.90%	39753.6	2.79%	37913	4.65%
$\lambda_0 = 2, \lambda = 4$	14550	0.89%	39948	3.30%	38068.9	5.08%
$\lambda_0 = 2, \lambda = 5$	14605.6	1.27%	39782	2.87%	38679	6.76%
$\lambda_0 = 3, \lambda = 1$	14593	1.19%	39922	3.23%	38385.3	5.95%
$\lambda_0 = 3, \lambda = 2$	14649.9	1.58%	39834	3.00%	37971	4.81%
$\lambda_0 = 3, \lambda = 3$	14623.2	1.40%	39817.9	2.96%	38184	5.39%
$\lambda_0 = 3, \lambda = 4$	14577.8	1.08%	39752.5	2.79%	38409	6.01%
$\lambda_0 = 3, \lambda = 5$	14555.4	0.92%	40304.5	4.22%	38617.4	6.59%
$\lambda_0 = 4, \lambda = 1$	14632	1.46%	40046	3.55%	37888	4.58%
$\lambda_0 = 4, \lambda = 2$	14544.9	0.85%	39747.2	2.78%	37895.8	4.60%
$\lambda_0 = 4, \lambda = 3$	14546.3	0.86%	39878	3.12%	38283	5.67%
$\lambda_0 = 4, \lambda = 4$	14682.9	1.81%	39740	2.76%	38357	5.87%
$\lambda_0 = 4, \lambda = 5$	14868.4	3.10%	39677.8	2.60%	38318.3	5.76%
$\lambda_0 = 5, \lambda = 1$	14542.8	0.84%	39827.3	2.98%	38302.7	5.72%
$\lambda_0 = 5, \lambda = 2$	14589.4	1.16%	39676.6	2.60%	38315	5.75%
$\lambda_0 = 5, \lambda = 3$	14580.8	1.10%	39687.3	2.62%	38280	5.66%
$\lambda_0 = 5, \lambda = 4$	14629	1.44%	39670.9	2.58%	38747.9	6.95%
$\lambda_0 = 5, \lambda = 5$	14599.9	1.23%	39812.2	2.95%	38774.4	7.02%

4.3 螞蟻數量之設定

螞蟻數量關係求解搜尋空間，螞蟻數量愈多，則求解搜尋空間就愈大。以 TSP 問題為例，5 個節點將有 $5!$ 之路徑組合。以 ATSP 問題為例，5 個節點將有 $(5)^5$ 之路徑組合。因此本節將針對 TSP 問題以及 ATSP 問題兩種不同問題型態進行螞蟻數量測試，其中螞蟻數量設定之最大值不大於節點數。 $\alpha, \beta, \sigma, \rho, \lambda_0, \lambda$ 值分別採用上一節測試之結果 $\alpha = 1$ 、 $\beta = 3$ 、 $\sigma = 0.01$ 、 $\rho = 0.20$ ， $\lambda_0 = 2, \lambda = 1$ (TSP)， $\lambda_0 = 2, \lambda = 2$ (ATSP)，記憶盒容量為 5。使用 AMS 於 TSP 問題之螞蟻數量設定如下表 4.7。使用 AMS 於 ATSP 問題之螞蟻數量設定如下表 4.8。

表 4.7 AMS 於 TSP 問題之螞蟻數量設定

	eil51	GAP(%)	kroA100	GAP(%)	d198	GAP(%)
已知最佳解	426		21282		15780	
Ant10	427.3	0.31%	21361.2	0.37%	15994.2	1.36%
Ant20	427.3	0.31%	21299.8	0.08%	15929.8	0.95%
Ant30	427.1	0.26%	21328.2	0.22%	15993.2	1.35%
Ant40	427.2	0.28%	21293.2	0.05%	15921.8	0.90%
Ant50	426.9	0.21%	21327	0.21%	15934.8	0.98%
Ant60	-	-	21313.4	0.15%	15885.8	0.67%
Ant70	-	-	21309.4	0.13%	15920.4	0.89%
Ant80	-	-	21303.2	0.10%	15890.6	0.70%
Ant90	-	-	21306.6	0.12%	15921.4	0.90%
Ant100	-	-	21308.7	0.13%	15914.7	0.85%
Ant110	-	-	-	-	15933.1	0.97%
Ant120	-	-	-	-	15937.7	1.00%
Ant130	-	-	-	-	15921.7	0.90%
Ant140	-	-	-	-	15927.5	0.93%
Ant150	-	-	-	-	15940.6	1.02%
Ant160	-	-	-	-	15955.7	1.11%
Ant170	-	-	-	-	15909.6	0.82%
Ant180	-	-	-	-	15930.8	0.96%
Ant190	-	-	-	-	15947.2	1.06%
Ant200	-	-	-	-	15966.4	1.18%

由表 4.7 可發現, *eil51* 在螞蟻數量為 50 時所得之結果為最好, *kroA100* 在螞蟻數量為 40 時所得之結果為最好, *d198* 在螞蟻數量 60 時所得之結果為最好, 因此本研究根據實驗之結果設計出之螞蟻數量為 $m=n*1/4+2000/n$ (n 為節點數)。

表 4.8 ATSP 問題螞蟻參數之設定

	ry48p	GAP(%)	ft70	GAP(%)	kro124p	GAP(%)
已知最佳解	14422		38673		36230	
Ant10	14562	0.97%	39663.5	2.56%	37893.3	4.59%
Ant20	14532.5	0.77%	39460.4	2.04%	37715.1	4.10%
Ant30	14523.3	0.70%	39378.2	1.82%	37088	2.37%
Ant40	14510.9	0.62%	39310.7	1.65%	37121.8	2.46%
Ant50	14504.8	0.57%	39275.7	1.56%	36832.1	1.66%
Ant60	-		39244.5	1.48%	36880.1	1.79%
Ant70	-		39182.8	1.32%	36710.1	1.33%
Ant80	-		-		36889.5	1.82%
Ant90	-		-		36746.5	1.43%
Ant100	-		-		36484.2	0.70%

由表 4.8 可發現, *ryp48* 在螞蟻數量為 50 時所得之結果為最好, *ft70* 在螞蟻數量為 70 時所得之結果為最好, *kro124p* 在螞蟻數量 100 時所得之結果為最好。本研究根據實驗之結果可知在面對 ATSP 問題時, 需要龐大之搜尋空間。因此, 將螞蟻數等於城市數之結果為較優 ($m = n$)。

4.4 記憶盒容量之設定

記憶盒主要功能為能有效的控制費蒙更新, 而記憶盒容量之設定將會影響螞蟻接受較劣解進行費落蒙更新, 以下就針對記憶盒容量 1、5、10、15、20 對 TSP 以及 ATSP 兩種問題類型進行測試。 $\alpha, \beta, \sigma, \rho, \lambda_0, \lambda, m$ 值分別採用上一節測試之結果 $\alpha = 1$ 、 $\beta = 3$ 、 $\sigma = 0.01$ 、 $\rho = 0.20$ 、 $\lambda_0 = 2, \lambda = 1$ (TSP), $\lambda_0 = 2, \lambda = 2$ (ATSP), $m = 1/4 * n + 2000/n$ (TSP)、 $m = n$ (ATSP)。表 4.9-4.10 為測試之結果, 圖 4.1-4.5 分別為 TSP 國際

例題 kroA100 記憶盒容量 1、5、10、15、20 接受較劣解進行費落蒙更新之情形。圖 4.6-4.10 分別為 ATSP 國際例題 kro124p 記憶盒容量 1、5、10、15、20 接受較劣解進行費落蒙更新之情形。

表 4.9 記憶盒容量 1、5、10 所測試之結果

國際例題	已知最佳解	1	GAP(%)	5	GAP(%)	10	GAP(%)
eil51	426	426.6	0.14%	427	0.23%	427.2	0.28%
kroA100	21282	21398.2	0.55%	21321.4	0.19%	21378.8	0.45%
d198	15780	16008.4	1.45%	15925.4	0.92%	15959.2	1.14%
ry48p	14422	14467.6	0.32%	14503.5	0.57%	14491.4	0.48%
ft70	38673	39134.6	1.19%	39094.6	1.09%	39136.2	1.20%
Kro124p	36230	36580.3	0.97%	36345	0.32%	36486.8	0.71%

表 4.10 記憶盒容量 15、20 所測試之結果

國際例題	已知最佳解	15	GAP(%)	20	GAP(%)
eil51	426	427.2	0.28%	427.4	0.33%
kroA100	21282	21357.2	0.35%	21532.8	1.18%
d198	15780	15962.6	1.16%	15967.6	1.19%
ry48p	14422	14498.7	0.53%	14501.1	0.55%
ft70	38673	39202.8	1.37%	39237	1.46%
Kro124p	36230	36645	1.15%	36885	1.81%

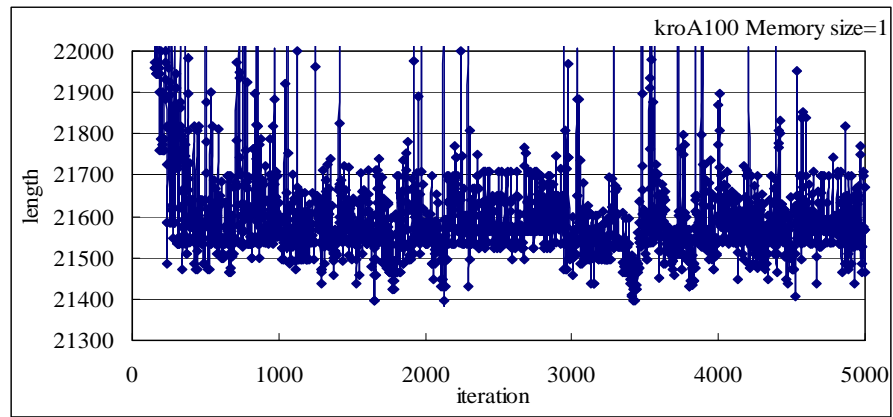


圖 4.1 kroA100 記憶盒容量為 1 時費落蒙更新之情形

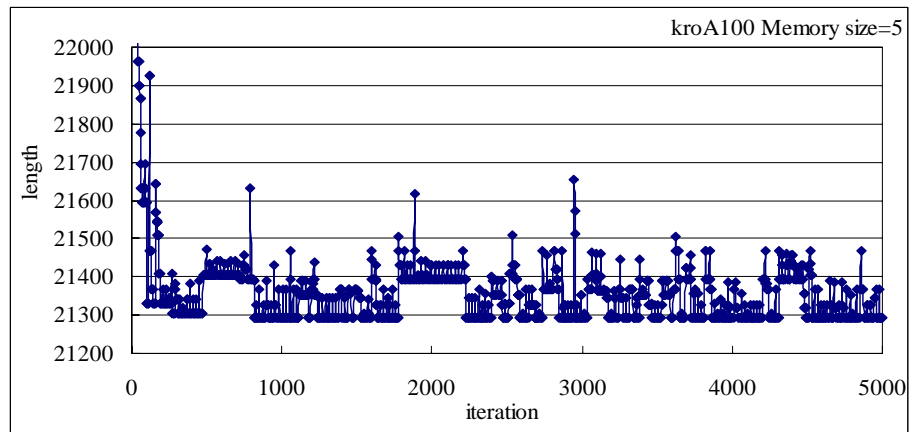


圖 4.2 kroA100 記憶盒容量為 5 時費落蒙更新之情形

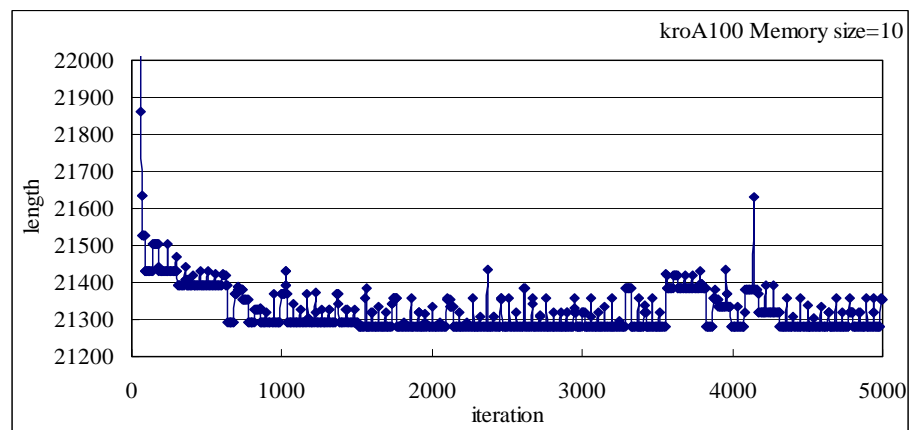


圖 4.3 kroA100 記憶盒容量為 10 時費落蒙更新之情形

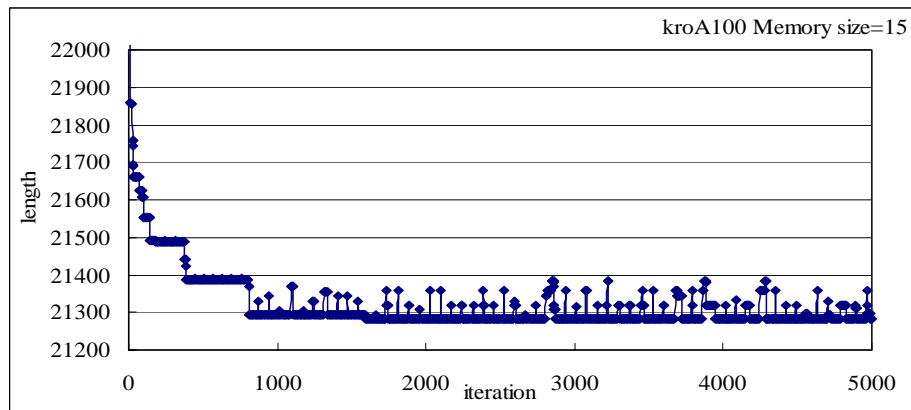


圖 4.4 kroA100 記憶盒容量為 15 時費落蒙更新之情形

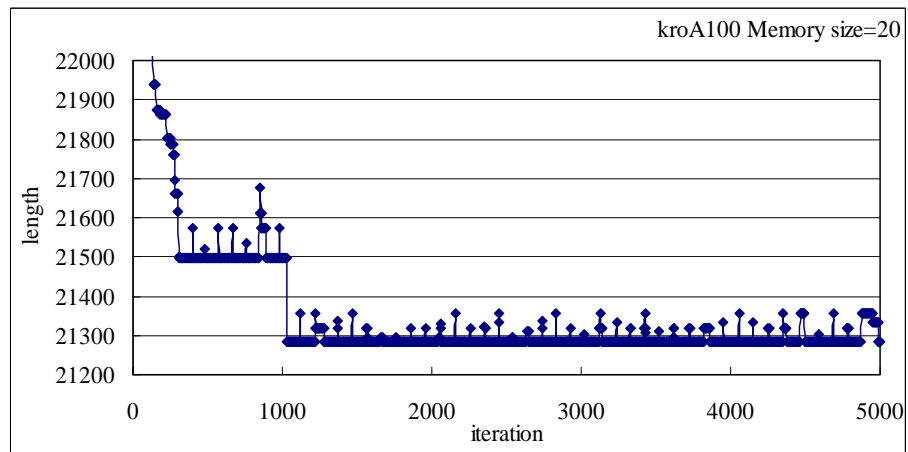


圖 4.5 kroA100 記憶盒容量為 20 時費落蒙更新之情形

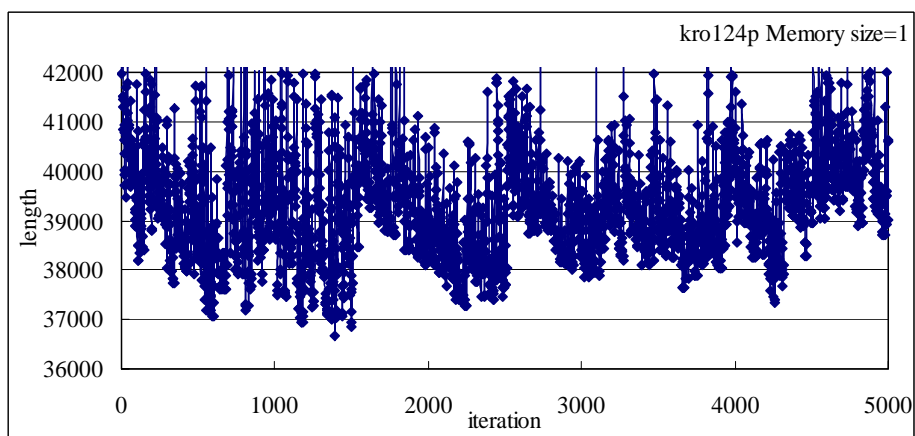


圖 4.6 kro124p 記憶盒容量為 1 時費落蒙更新之情形

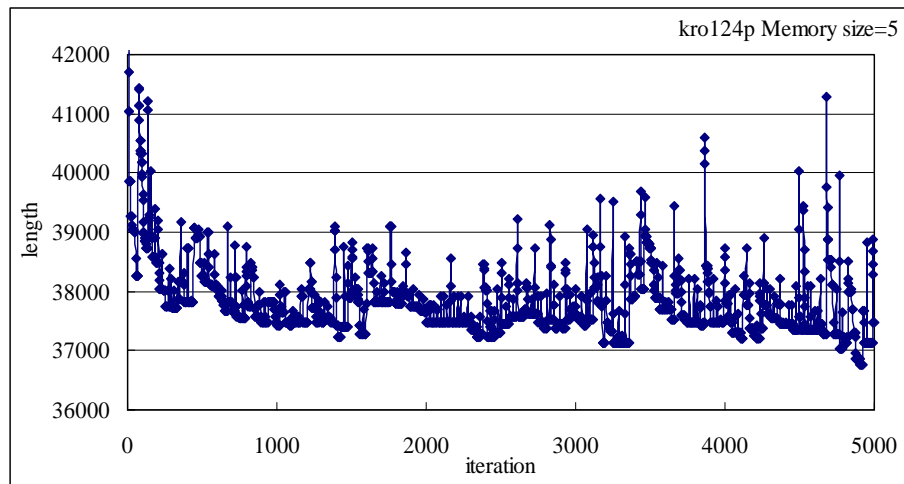


圖 4.7 kro124p 記憶盒容量為 5 時費落蒙更新之情形

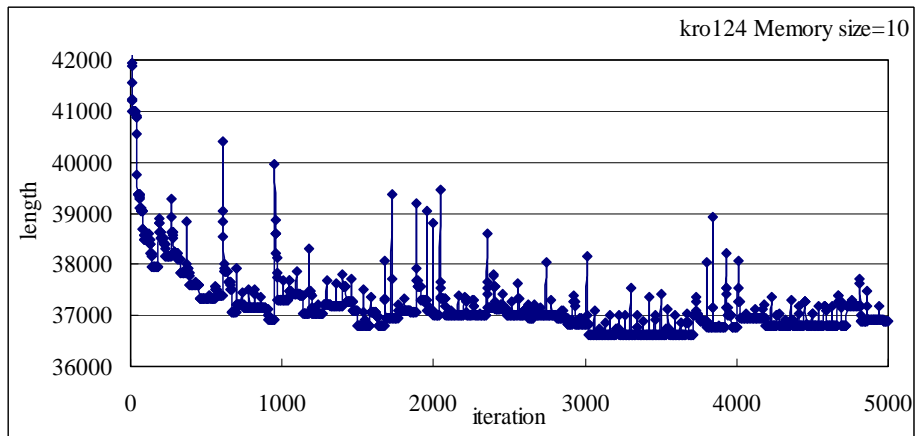


圖 4.8 kro124p 記憶盒容量為 10 時費落蒙更新之情形

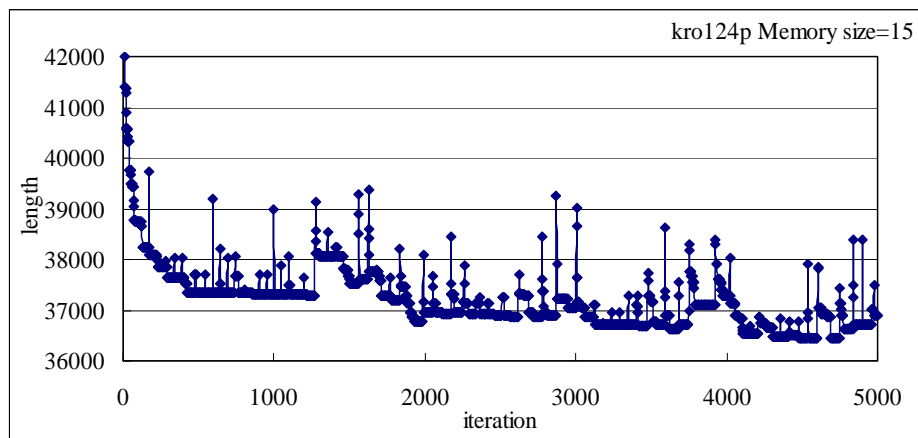


圖 4.9 kro124p 記憶盒容量為 15 時費落蒙更新之情形

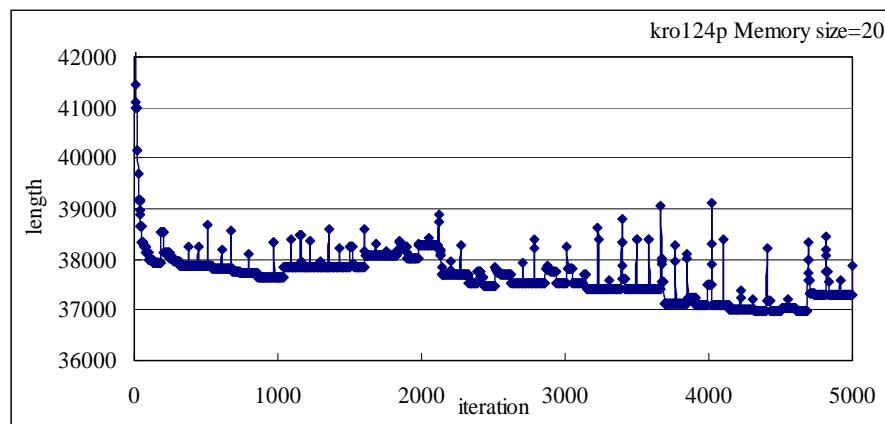


圖 4.10 kro124p 記憶盒容量為 20 時費落蒙更新之情

由以上記憶盒容量大小測試之結果可發現，當記憶容量愈小時，接受暫劣解進行費落更新之範圍愈大(幅度愈大)，如圖 4.1、圖 4. 所示，此情形就如一直接受疊代最佳解進行費落蒙更新，可能會對不該走之路徑進行費落蒙更新。當記憶盒容量愈大時，則接受劣解進行費落蒙更新之範圍愈小(幅度愈小)，如圖 4.5、圖 4.10 所示，此情形就如一直接受至 t 疊代之最佳解進行費落蒙更新，無法對適當劣解進行費蒙更新。而由表 4.9 以及表 4.10 之結果可發現當記憶盒容量為 5 時所到之結果為最好，且由圖 4.2、圖 4.7 發現記憶盒容量為 5 時，其接受劣解之跳動最為平順(不會過於緩和，也不會過於跳動太大)。

第五章 測試結果

本章節將針對第三章所提出之 AMS 演算法進行測試。5.1 節為 AMS 測試之結果。5.2 節為比較其他改良式螞蟻演算法應用於 TSP 問題之求解品質。

5.1 螞蟻記憶系統之測試

本研究所測試之停止條件為參考Dorigo[27]以及Stützle[35]文獻的測試停止條件。Dorigo[27]以及Stützle[35]在測試TSP問題時，是採用 $n*10000$ 之建構解，在測試ATSP時則採用 $n*20000$ 之建構解(n 為城市數目)為停止條件。本研究在測試TSP問題時採用 $m*10000$ 之建構解(m 為螞蟻數目， $m=1/4*n+2000/n$)，即建構解數目小於文獻中之建構解數目。在測試ATSP問題時則採用和文獻建構解數目一樣 $n*20000$ 。

在參數設定部份，使用第四章所測試的最好參數組合，即為 $\alpha = 1$ 、 $\beta = 3$ 、 $\sigma = 0.01$ 、 $\rho = 0.20$ ， $\lambda_0 = 2, \lambda = 1$ (TSP)， $\lambda_0 = 2, \lambda = 2$ (ATSP)， $m=1/4*n+2000/n$ (TSP)， $m=n$ (ATSP)，記憶容量 $M_s=5$ 。初始費落蒙(τ_0)為 $1/(L_m*n)$ ，初始解採用 L_m 。每題國際例題皆執行30次，其平均測試結果如表5.1所示：

表 5.1 螞蟻記憶系統測試七個國際例題之結果

國際例題 [節點數目]	已知 最佳解	AMS 最佳結果 (GAP(%))[建構解數目]	AMS 平均值 (GAP(%))	標準差	完成一建解 所需時間 (秒)
eil51 [51]	426	426(0.00%) [50592]	426.93(0.22%)	0.45	6.16E-05
kroA100 [100]	21282	21282(0.00%) [22185]	21291.93(0.05%)	17.88	2.29E-04
d198 [198]	15780	15820(0.25%) [571061]	15890.17(0.70%)	27.80	9.25E-04
ry48p [48]	14422	14422(0.00%) [699936]	14487.30(0.45%)	23.55	6.09052E-05
ft70 [70]	38673	38722(0.13%) [1088640]	38975.90(0.78%)	104.95	1.20E-04
kro124p [100]	36230	36230(0.00%) [281900]	36293.77(0.18%)	111.41	2.28E-04
ftv170 [171]	2755	2758(0.11%) [1386981]	2773.27(0.66%)	16.11	6.80E-04

5.2 比較其他改良式螞蟻演算之結果

本小節將比較其他改良式螞蟻演算法(AS, ASe, ASrank, MMAS, MMAS+pts, ACS and ACOMAS)之文獻結果。在測試例題上, 本研究將採用其他改良式螞蟻演算法常用之對稱性(Symmetric)以及非對稱性(Asymmetric)TSP的國際例題。AS[14-16]、ASe[16]以及ASrank[8-10]原作者皆無使用國際例題, 因此採用由Stützle[35]測出之結果, 而MMAS[35]、ACS[27]以及ACOMAC[22]皆採用原作者之文獻解。在路徑建構數目方面AS、ASe、ASrank、MMAS、MMAS+pts、ACS皆採用kn10000為基準, 在求解TSP時, $k=1$; 在求解ATSP時, $k=2$ 。執行次數以30次為基準。在ACOMAC[22]文獻中未提及使用疊代(iteration)數目以及建構(tour structions)數目, 因此本究只列出ACOMAC之求解之結果, 不做任何比較。

表 5.2 AMS、AS、ACS 測試國際例題之結果

國際例題	opt	AMS	GAP(%)	AS	GAP(%)	ACS	(GAP(%)
eil51	426	426.93	0.22%	437.3	2.65%	428.1	0.49%
kroA100	21282	21291.93	0.05%	22247.1	4.53%	21420.0	0.65%
d198	15780	15890.17	0.70%	16702.1	5.84%	16054.0	1.74%
ry48p	14422	14487.30	0.45%	15296.4	6.06%	14565.4	0.99%
ft70	38673	38975.90	0.78%	39596.3	2.39%	39099.0	1.10%
kro124p	36230	36293.77	0.18%	38733.1	6.91%	36857.0	1.73%
ftv170	2755	2773.27	0.66%	3154.5	14.50%	2826.5	2.60%

* Opt 為已知最佳解。ACS 採用 Dorigo[27]測式之結果。AS 採用 Stützle [35]測試結果。

表 5.3 ASrank、ASe、MMAS 測試國際例題之結果

國際例題	opt	ASrank	GAP(%)	ASe	GAP(%)	MMAS	GAP(%)
eil51	426	434.5	2.00%	428.3	0.54%	427.6	0.38%
kroA100	21282	21746.0	2.18%	21522.8	1.13%	21320.3	0.18%
d198	15780	16199.1	2.66%	16205.0	2.69%	15972.5	1.22%
ry48p	14422	14511.4	0.62%	14685.2	1.82%	14553.2	0.91%
ft70	38673	39410.1	1.91%	39261.8	1.52%	39040.2	0.95%
kro124p	36230	36973.5	2.05%	37510.2	3.53%	36773.5	1.50%
ftv170	2755	2854.2	3.60%	2952.4	7.17%	2828.8	2.68%

* Opt 為已知最佳解。ASrank、ASe 以及 MMAS 採用 Stützle [35]之測試結果。

表 5.4 MMAS+pts、ACOMAC、ACOMAC+NN 以及 ACOMAC+DNN 測試國際例題之結果

國際例題	opt	MMAS +pts	GAP (%)	ACOMAC	GAP (%)	ACOMAC +NN	GAP (%)	ACOMAC +DNN	GAP (%)
eil51	426	427.1	0.26%	430.684	1.10%	430.0437	0.95%	430.0076	0.94%
kroA100	21282	21291.6	0.05%	21457.953	0.83%	21433.33	0.71%	21408.23	0.59%
d198	15780	15956.8	1.12%	16558.7	4.93%	16034	1.61%	15955.6	1.11%
ry48p	14422	14523.4	0.70%	-	-	-	-	-	-
ft70	38673	38922.7	0.65%	-	-	-	-	-	-
kro124p	36230	36573.6	0.95%	-	-	-	-	-	-
ftv170	2755	2817.7	2.28%	-	-	-	-	-	-

* Opt 為已知最佳解。pts 為 pheromone trail smoothing。MMAS、MMAS+pts 採用 Stützle [35]之測試結果。ACOMAC、ACOMAC+NN、ACOMAC+DNN 採用 Fa Tsai[22] 之測試結果。

表 5.5 AMS、AS、ACS、ASrank、ASe、MMAS、MMAS+pts 測試國際例題誤差值比較

國際例題	AMS	AS	ACS	ASrank	ASe	MMAS	MMAS+pts
eil51	0.22%	2.65%	0.49%	2.00%	0.54%	0.38%	0.26%
kroA100	0.05%	4.53%	0.65%	2.18%	1.13%	0.18%	0.05%
d198	0.70%	5.84%	1.74%	2.66%	2.69%	1.22%	1.12%
ry48p	0.45%	6.06%	0.99%	0.62%	1.82%	0.91%	0.70%
ft70	0.78%	2.39%	1.10%	1.91%	1.52%	0.95%	0.65%
kro124p	0.18%	6.91%	1.73%	2.05%	3.53%	1.50%	0.95%
ftv170	0.66%	14.50%	2.60%	3.60%	7.17%	2.68%	2.28%

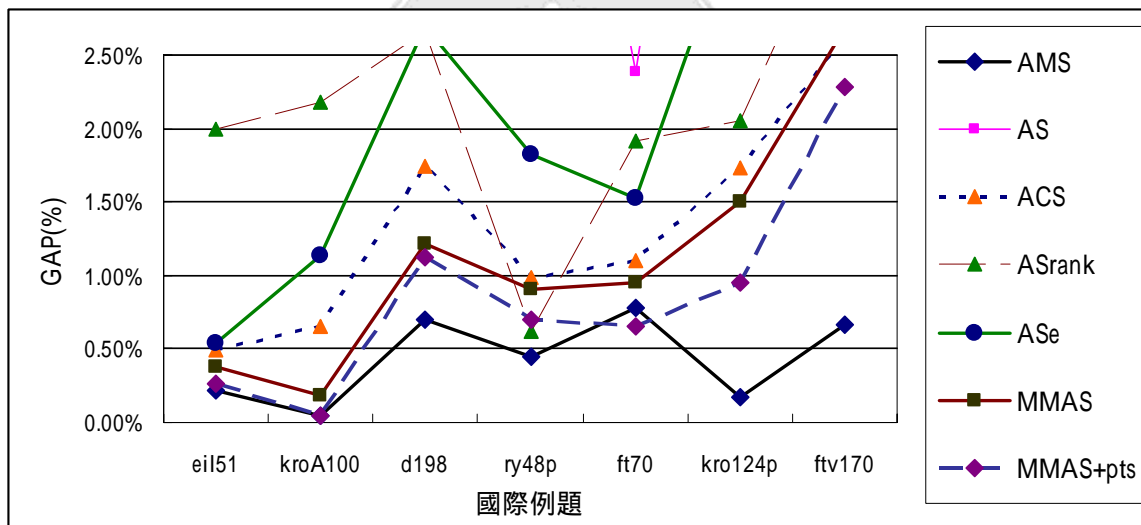


圖 5.1 各演算法 GAP 比較圖

由表 5.5 以及圖 5.1 之測試結果可發現，AMS 之測試結果除了在 ft70 國際例題較 MMAS+pts 表現些微差外(ft70 誤差值較 MMAS+pts 多出 0.13%)，其餘皆優於 MMAS+pts，包括 eil51 AMS 較 MMAS+pts 優 0.04%、d198 AMS 較 MMAS+pts 優 0.42%、ry48p AMS 較 MMAS+pts 優 0.25%、kro124p AMS 較 MMAS+pts 優 0.77%以及 ftv170 AMS 較

MMAS+pts 優 1.62%。此外, AMS 之求解表現皆較 AS、ACS、ASrank、ASe、MMAS 等改良式螞蟻演算法具大幅度改良。



第六章 結論與建議

6.1 結論

1. 本研究提出一新啟發式解法稱之為螞蟻記憶系統(Ant Memory System, AMS)求解對稱性(Symmetric)TSP 問題以及非對稱性(Asymmetric)TSP 二種問題類型七個國際例題。
2. 從文獻中可發現從最早改良式螞蟻演算法 ASe 到 ACOMAS, 都是從費落蒙更新方式進行改良, 因此可想而知費落蒙更新方式將是影響求解之品質之重要關鍵。
3. 本研究所提及之螞蟻記憶系統(Ant Memory System, AMS)其目的將能有效控制費落蒙更新方式, 以提升其求解品質。記憶盒容量為 5 時, 所得之結果為最佳。當記憶盒容量為 1 時, 因接受劣解之範圍太大, 較易往較差之路徑行走, 此情形就好像一直採用疊代最佳解進行費落蒙更新, 而當記憶盒容量為 20 時, 則因接受劣解之範圍太小, 不易跳脫局部解。
4. AMS 除了在 ft70 國際例題較 MMAS+pts 表現些微差外(ft70 誤差值較 MMAS+pts 多出 0.13%), 其餘皆優於 MMAS+pts, 包括 eil51 AMS 較 MMAS+pts 優 0.04%、d198 AMS 較 MMAS+pts 優 0.42%、ry48p AMS 較 MMAS+pts 優 0.25%、kro124p AMS 較 MMAS+pts 優 0.77%以及 ftv170 AMS 較 MMAS+pts 優 1.62%。此外, AMS 之求解表現皆較 AS、ACS、ASrank、ASe、MMAS 等改良式螞蟻演算法具大幅度之改良。

6.2 建議

1. 至今, 螞蟻演算法已被廣泛應用於組合最佳化問題上, 例如車輛巡迴問題(VRP)、物流配送之路線安排問題、生產排程...等相關排程問題, 在未來或許也能將本篇論文所提出之螞蟻記憶系應用於相關排序課題上。

2. 本研究所提出之螞蟻記憶系統，雖比其他改良式螞蟻演算法優，但其求解正確性還有改善空間，因此建議未來對螞蟻理論有興趣之學者，建議以費落蒙更新著手，以期望有更好之螞蟻模型誕生。



參考文獻

1、中文部份

- [1] 許為元，民 89 年，「複合式自我學習之基因演算法應用於旅行推銷員問題」，國立臺灣大學資訊工程學研究所碩士論文，台北市。
- [2] 陳隆熙，民 91 年，「一個解決 TSP 問題最佳解的穩定方法—以 TA 演算法為例」，私立大業大學工業工程學系碩士班碩士論文，彰化縣。
- [3] 陳惠國、林正章、汪進財、卓訓榮、顏上堯、李宗儒、許巧鶯、韓復華、李治綱、蘇雄義、陳春益合著，民 90 年，「運輸網路分析」，五南圖書出版有限公司。
- [4] 陳建緯，民 90 年，「大規模旅行推銷問題之研究：鄰域搜尋法與巨集啟發式解法之應用」，國立交通大學運輸工程與管理學系碩士論文，新竹市。
- [5] 曾國雄、王日昌、黃明居，「以基因演算法與槎板路徑求解旅行推銷員問題」，運輸計劃季刊，第二 五卷第三期，民國八 五年九月，pp.493-516。
- [6] 馮正民、邱裕鈞，民 93 年，「研究分析方法」，建都文化事業股份有限公司。
- [7] 韓復華，楊智凱，「門檻接受法在 TSP 問題上之應用」，運輸計劃季刊，第二 五卷第二期，民國八 五年六月，pp.163-188。

2、英文部份

- [8] Bullnheimer, B., Hartl, R. F. and Strauss, C., 1997, “A new rank-based version of the ant system: a computational study,” Technical Report POM-03/97, Institute of Management science, University of Vienna.
- [9] Bullnheimer, B., Hartl R.F. and C. Strauss, 1999, “A new rank-based version of the ant system:a computational study”, *Central European Journal of Operations Research* 7 (1), pp.25–38.
- [10] Bullnheimer, B., Hartl, R. F. and Strauss, C., 1999,“An improved ant system algorithm for the vehicle routing problem,” *Annals of Operations Research*,Vol.89, pp.319-328.
- [11] Belegundu, A. D. and Chadrapatla T. R., 1999, “Optimization concepts and application in engineering,” Prentice Hall.
- [12] Bland, J.A., 1999, “Space-planning by ant colony optimization,” *International Journal of Computer Applications in Technology* 12 (6), pp.320–328.
- [13] Dueck, G. and Scheuer, T., 1990, “Threshold accepting : A general purpose optimization algorithm appearing superior to simulated annealing,” *Journal of Computational Physics*, Vol. 90, pp161-175.
- [14] Dorigo, M., Maniezzo, V. and Colorni, A., 1991, “Positive feedback as a search strategy,” Technical Report 91-016, Dipartimento Elettronica, Politecnico di Milano, Italy.
- [15] Dorigo, M., 1992, “Optimization learning and natural algorithms,” PhD Thesis, Dip. Elettronica, Politecnico di Milano, Italy.
- [16] Dorigo, M., Maniezzo, V. and Colorni, A., 1996, “Ant system :

- Optimization by a colony of cooperating agents,” *IEEE Transactions on Systems, Man, and Cybernetics-Part B*, vol. 26(1), pp.29-41.
- [17] Dorigo, M. and Gambardella, L. M., 1997, “Ant colony system : a cooperative learning approach to the traveling salesman problem,” *IEEE Transactions on Evolutionary Computation*, Vol. 1(1), pp.53-66.
- [18] Dorigo, M. and Gambardella, L. M., 1997, “Ant colonies for the traveling salesman problem,” *BioSystems*, Vol.43, pp.73-81.
- [19] Dorigo, M., Caro, G. D., Gambardella, L. M., 1999, “Ant algorithms for distributed discrete optimization,” *Artificial Life*, Vol.5(2), pp.137-172.
- [20] Dorigo, M., Caro, G. D., 1999, “New Ideas in Optimization,” McGraw-Hill.
- [21] Dorigo M., Bonabeau E., Theraulaz G., 2000, “Ant algorithms and stigmergy”, *Future Generation computer Systems 16* , pp.889-914.
- [22] Fa Tsai, C., Wei Tsai, C., Chang Tseng, C., 2004, “A new hybrid heuristic approach for solving large traveling salesman problem,” *Information Sciences 166*, pp.67-81.
- [23] Goldberg, D. and Lingle R. A., 1985, “Locs and thhe traveling saleman problem”, Proceedings of the 1st International Conference on Genetic Algorithms, Lawrence Erlbaum Associates, Hillsdale, NJ, pp.154-159.
- [24] Glover, F. and McMillan, C., 1986, “The general employee Scheduling Problem : An integration of MS and AI.” *Computers and Operations Research*, Vol. 13, pp. 563-573.

- [25] Glover, F., 1989, Tabu search-part I, *ORSA Journal of Computing* 1(3), pp.190–206.
- [26] Glover, F. , 1990, Tabu search-part II, *ORSA Journal of Computing* 12 (1), pp.4–32.
- [27] Gambardella L. M. and Dorigo M., 1996, “Solving symmetric and asymmetric TSPs by ant colonies,” in: Proceedings of the IEEE International Conference on Evolutionary Computation (ICEC’96), IEEE Press, Piscataway, pp.622–627.
- [28] Holland, J.H., 1975, *Adaptation in Natural and Artificial System*, University of Michigan Press, Ann Arbor.
- [29] Kirkpatrick, S., Gelatt Jr., C.D. and Vecchi, M.P. , 1983, ”Optimization by simulated annealing,” *Science*, Vol.220,pp.671-680.
- [30] Maniezzo, V. and Colorni A., 1999, “The ant system applied to the quadratic assignment problem,” *IEEE Transactions on Knowledge and Data Engineering* 11(5), pp.769–778.
- [31] Parpinelli, R.S., Lopes H.S. and Freitas, A.A., 2002, “Data mining with an ant colony optimization algorithm,” *IEEE Transactions on Evolutionary Computation* 6 (4), pp.321–332.
- [32] Stützle, T. and Hoos, H., 1997, “MAX-MIN Ant System for quadratic assignment problems.” Technical Report AIDA-97-04, Intellectics Group, Department of Computer Science, Darmstadt University of Technology, Germany, July.
- [33] Stützle, T., and Hoos, H.H., 1997, “The MAX-MIN ant system and local search for the traveling salesman problem.” in Bäck, T., Michalewicz, Z., Yao, X.(editors), Proceedings of the IEEE International Conference on Evolutionary Computation (ICEC’97),

IEEE press, Piscataway, USA, pp.309-314.

- [34] Stützle, T., and Hoos, H.H., 1998, "Improvements on the Ant system: introducing the MAX-MIN Ant System," in: proceedings of the International Conference on Artificial Neural Networks and Genetic Algorithms, Springer, Vienna, pp.245-249.
- [35] Stützle T., and Hoos H.H., 2000, "MAX-MIN Ant System," *Future Generation computer Systems* 16, pp.889-914.



附錄一、 AMS 參數組合

Eil51	$\alpha=1, \beta=2$	$\alpha=1, \beta=3$	$\alpha=1, \beta=4$	$\alpha=1, \beta=5$
$\sigma=0.01, \rho=0.01$	428.3	429.5	430.89	433
$\sigma=0.01, \rho=0.05$	429.3	428.2	427.9	428.7
$\sigma=0.01, \rho=0.10$	429.4	428.5	427.9	429.9
$\sigma=0.01, \rho=0.15$	428.7	428.2	428.3	431.1
$\sigma=0.01, \rho=0.20$	429.4	427.3	428	430.7
$\sigma=0.05, \rho=0.01$	436.9	441.7	440.8	428
$\sigma=0.05, \rho=0.05$	429.00	426.9	428.1	429.1
$\sigma=0.05, \rho=0.10$	428.2	426.7	427.3	427.9
$\sigma=0.05, \rho=0.15$	428.5	427.1	427.1	426.5
$\sigma=0.05, \rho=0.20$	428.1	428.2	426.9	426.8
$\sigma=0.10, \rho=0.01$	448.1	442.9	448.6	429
$\sigma=0.10, \rho=0.05$	430.5	431.2	430.6	433.4
$\sigma=0.10, \rho=0.10$	428.6	427.2	428.6	428.9
$\sigma=0.10, \rho=0.15$	430	427.1	427.6	428.5
$\sigma=0.10, \rho=0.20$	428.8	426.9	426.6	428.3
$\sigma=0.15, \rho=0.01$	449.6	447.5	453.3	428.1
$\sigma=0.15, \rho=0.05$	435.9	434.3	433.1	434.5
$\sigma=0.15, \rho=0.10$	431.2	429.3	430.4	431.8
$\sigma=0.15, \rho=0.15$	430.7	427.8	428.9	430.2
$\sigma=0.15, \rho=0.20$	430.2	427.1	428.3	429.4
$\sigma=0.20, \rho=0.01$	455.3	450.9	453.3	428.1
$\sigma=0.20, \rho=0.05$	435.7	434.9	438.8	440.8
$\sigma=0.20, \rho=0.10$	432.7	433.8	430.7	432.3
$\sigma=0.20, \rho=0.15$	430.6	430.3	431.6	433.1
$\sigma=0.20, \rho=0.20$	432.7	428.9	430.1	433.1

Eil51	$\alpha = 2, \beta = 2$	$\alpha = 2, \beta = 3$	$\alpha = 2, \beta = 4$	$\alpha = 2, \beta = 5$
$\sigma = 0.01, \rho = 0.01$	429.2	428	428.4	428.7
$\sigma = 0.01, \rho = 0.05$	430	428	428.5	428.1
$\sigma = 0.01, \rho = 0.10$	429.9	428.8	428	427.9
$\sigma = 0.01, \rho = 0.15$	429.7	429.4	428.9	429.1
$\sigma = 0.01, \rho = 0.20$	430	429.4	429.3	428.8
$\sigma = 0.05, \rho = 0.01$	431.7	428.6	431.3	431.8
$\sigma = 0.05, \rho = 0.05$	428.2	426.8	426.9	426.7
$\sigma = 0.05, \rho = 0.10$	428.1	427.0	427.2	426.6
$\sigma = 0.05, \rho = 0.15$	428.4	428.1	427.4	427.3
$\sigma = 0.05, \rho = 0.20$	427.5	427.0	427.4	427.6
$\sigma = 0.10, \rho = 0.01$	435.6	432.5	434.8	438.1
$\sigma = 0.10, \rho = 0.05$	429.9	427.4	427.2	428
$\sigma = 0.10, \rho = 0.10$	428.7	427.8	426.8	426.8
$\sigma = 0.10, \rho = 0.15$	429.6	427.4	426.5	426.4
$\sigma = 0.10, \rho = 0.20$	428.3	427.3	426.9	426.8
$\sigma = 0.15, \rho = 0.01$	441.1	439.4	439.33	442.8
$\sigma = 0.15, \rho = 0.05$	431.4	428.5	427.9	430.4
$\sigma = 0.15, \rho = 0.10$	431	428.3	427.6	427.2
$\sigma = 0.15, \rho = 0.15$	432.3	428.4	428.8	426.9
$\sigma = 0.15, \rho = 0.20$	430	428.4	428.6	428.3
$\sigma = 0.20, \rho = 0.01$	444.67	442.8	443.2	446.4
$\sigma = 0.20, \rho = 0.05$	432.8	429.5	431.1	432
$\sigma = 0.20, \rho = 0.10$	431.7	430.2	429.8	431.1
$\sigma = 0.20, \rho = 0.15$	434.7	427.9	428	428.5
$\sigma = 0.20, \rho = 0.20$	435.9	429.3	429.1	427.9

Eil51	$\alpha = 3, \beta = 2$	$\alpha = 3, \beta = 3$	$\alpha = 3, \beta = 4$	$\alpha = 3, \beta = 5$
$\sigma = 0.01, \rho = 0.01$	428.3	428	428.5	428.4
$\sigma = 0.01, \rho = 0.05$	428.1	428.7	429	428.3
$\sigma = 0.01, \rho = 0.10$	428.3	429.5	430.2	428.7
$\sigma = 0.01, \rho = 0.15$	428.2	429.5	428.4	429.5
$\sigma = 0.01, \rho = 0.20$	428.6	429.8	428.6	428.3
$\sigma = 0.05, \rho = 0.01$	427.3	427.7	428.7	428.7
$\sigma = 0.05, \rho = 0.05$	429	426.8	427.4	427.5
$\sigma = 0.05, \rho = 0.10$	428	427.8	427.5	426.7
$\sigma = 0.05, \rho = 0.15$	427.8	427.8	427.8	427.2
$\sigma = 0.05, \rho = 0.20$	427.3	427.9	427.7	427.5
$\sigma = 0.10, \rho = 0.01$	431.7	429.1	433.1	433.1
$\sigma = 0.10, \rho = 0.05$	431.7	426.9	428	427
$\sigma = 0.10, \rho = 0.10$	428.1	427.9	427.2	427.3
$\sigma = 0.10, \rho = 0.15$	428.9	427.2	427.9	427.9
$\sigma = 0.10, \rho = 0.20$	428.3	427	427	427.6
$\sigma = 0.15, \rho = 0.01$	435.6	436.2	438.8	438.8
$\sigma = 0.15, \rho = 0.05$	434.5	428.6	427.4	428.2
$\sigma = 0.15, \rho = 0.10$	431.4	429.8	428.7	428
$\sigma = 0.15, \rho = 0.15$	431.4	428.7	428.7	429.3
$\sigma = 0.15, \rho = 0.20$	428.6	428.5	428	428.6
$\sigma = 0.20, \rho = 0.01$	439.3	439.7	440.6	440.6
$\sigma = 0.20, \rho = 0.05$	437.7	430.6	430.1	430.3
$\sigma = 0.20, \rho = 0.10$	431.8	430.8	429.1	429.4
$\sigma = 0.20, \rho = 0.15$	432.4	431	429.3	429.5
$\sigma = 0.20, \rho = 0.20$	431.2	432.1	429.9	428.5

kroA100	$\alpha=1, \beta=2$	$\alpha=1, \beta=3$	$\alpha=1, \beta=4$	$\alpha=1, \beta=5$
$\sigma=0.01, \rho=0.01$	21645.9	21533.8	21752.8	22155.3
$\sigma=0.01, \rho=0.05$	21487	21389.2	21485.5	21529.8
$\sigma=0.01, \rho=0.10$	21449.6	21372.1	21456	21527.4
$\sigma=0.01, \rho=0.15$	21383.3	21387.8	21454.9	21529.3
$\sigma=0.01, \rho=0.20$	21401	21362.8	21417.3	21481.1
$\sigma=0.05, \rho=0.01$	22547.4	22317.2	22436.7	22625.2
$\sigma=0.05, \rho=0.05$	21726.20	21500.4	21649.1	21690.3
$\sigma=0.05, \rho=0.10$	21583.9	21458.4	21516.3	21596.4
$\sigma=0.05, \rho=0.15$	21471.1	21420.4	21532.9	21577.9
$\sigma=0.05, \rho=0.20$	21447	21429.6	21433.5	21506.7
$\sigma=0.10, \rho=0.01$	23114.8	22696.3	22699	23029.4
$\sigma=0.10, \rho=0.05$	22173.7	21766.8	21984.3	22103
$\sigma=0.10, \rho=0.10$	21947	21589	21593.8	21753.1
$\sigma=0.10, \rho=0.15$	21761.4	21501.9	21547.6	21606.7
$\sigma=0.10, \rho=0.20$	21721.2	21466.6	21509.8	21537.8
$\sigma=0.15, \rho=0.01$	23229.4	22988.8	23069.9	23081.5
$\sigma=0.15, \rho=0.05$	22374.1	22024.1	22241.3	22297
$\sigma=0.15, \rho=0.10$	22151.6	21867.5	21865.6	22076.3
$\sigma=0.15, \rho=0.15$	22049.6	21743.2	21778	21886
$\sigma=0.15, \rho=0.20$	21923.4	21660.2	21662.7	21796.2
$\sigma=0.20, \rho=0.01$	23253.4	23220	23134.4	23097.4
$\sigma=0.20, \rho=0.05$	22633.5	22319.5	22290.6	22593.5
$\sigma=0.20, \rho=0.10$	22249.2	21888.9	22087.6	22177.9
$\sigma=0.20, \rho=0.15$	22305.3	21931.1	21946.1	22158
$\sigma=0.20, \rho=0.20$	22330.2	21839.3	21859.7	22146.1

kroA100	$\alpha = 2, \beta = 2$	$\alpha = 2, \beta = 3$	$\alpha = 2, \beta = 4$	$\alpha = 2, \beta = 5$
$\sigma = 0.01, \rho = 0.01$	21465.1	21386.7	16053.8	21521.22
$\sigma = 0.01, \rho = 0.05$	21849.9	21562.7	21584.1	21798.3
$\sigma = 0.01, \rho = 0.10$	22436.4	22060.4	22112.1	22351.8
$\sigma = 0.01, \rho = 0.15$	22675	22412.3	22463.6	22600.3
$\sigma = 0.01, \rho = 0.20$	23045.2	22581.4	22720.2	22754.6
$\sigma = 0.05, \rho = 0.01$	21849.9	21562.7	21584.1	21798.3
$\sigma = 0.05, \rho = 0.05$	21455.3	21444.4	21430.2	21471.5
$\sigma = 0.05, \rho = 0.10$	21494.8	21413.4	21420.5	21441
$\sigma = 0.05, \rho = 0.15$	21424	21376	21422.9	21442.5
$\sigma = 0.05, \rho = 0.20$	21459.2	21368	21432.7	21473.2
$\sigma = 0.10, \rho = 0.01$	22436.4	22060.4	22112.1	22351.8
$\sigma = 0.10, \rho = 0.05$	21776.5	21514.4	21545.6	21556.5
$\sigma = 0.10, \rho = 0.10$	21659.3	21462.1	21499.9	21523.9
$\sigma = 0.10, \rho = 0.15$	21663.4	21481.9	21512.7	21513.2
$\sigma = 0.10, \rho = 0.20$	21623.1	21544.9	21462.5	21499.3
$\sigma = 0.15, \rho = 0.01$	22675	22412.3	22463.6	22600.3
$\sigma = 0.15, \rho = 0.05$	22062	21708.6	21647.1	21765.5
$\sigma = 0.15, \rho = 0.10$	21931.9	21601.6	21588.9	21642.2
$\sigma = 0.15, \rho = 0.15$	21855.4	21641	21539.8	21553.5
$\sigma = 0.15, \rho = 0.20$	21936.9	21682.3	21594.7	21552.8
$\sigma = 0.20, \rho = 0.01$	23045.2	22581.4	22720.2	22754.6
$\sigma = 0.20, \rho = 0.05$	22295	21828.4	21937.4	21908.2
$\sigma = 0.20, \rho = 0.10$	22211.1	21802.9	21737.5	21805.6
$\sigma = 0.20, \rho = 0.15$	22346.1	21787.7	21654.6	21736.9
$\sigma = 0.20, \rho = 0.20$	22334.3	21758.6	21654.9	21644.1

kroA100	$\alpha = 3, \beta = 2$	$\alpha = 3, \beta = 3$	$\alpha = 3, \beta = 4$	$\alpha = 3, \beta = 5$
$\sigma = 0.01, \rho = 0.01$	21415.1	21384.5	21388.9	21398.1
$\sigma = 0.01, \rho = 0.05$	21560.6	21414.5	21461.6	21500.1
$\sigma = 0.01, \rho = 0.10$	22156.4	21727.1	21777.5	21888.1
$\sigma = 0.01, \rho = 0.15$	22314.9	22006.2	22060.6	22208.2
$\sigma = 0.01, \rho = 0.20$	22467.6	22233	22256	22426.2
$\sigma = 0.05, \rho = 0.01$	21560.6	21414.5	21461.6	21500.1
$\sigma = 0.05, \rho = 0.05$	21465.9	21410.1	21435.3	21458.9
$\sigma = 0.05, \rho = 0.10$	21494.1	21433.1	21423.5	21443.1
$\sigma = 0.05, \rho = 0.15$	21444.2	21384.7	21431.2	21454.7
$\sigma = 0.05, \rho = 0.20$	21475.8	21407.8	21400.4	21467
$\sigma = 0.10, \rho = 0.01$	22156.4	21727.1	21777.5	21888.1
$\sigma = 0.10, \rho = 0.05$	21800.4	21488.9	21514.4	21510.7
$\sigma = 0.10, \rho = 0.10$	21683.5	21533.6	21521.4	21480.2
$\sigma = 0.10, \rho = 0.15$	21691.3	21530.1	21496.5	21527.5
$\sigma = 0.10, \rho = 0.20$	21661.7	21537.2	21477.3	21488.6
$\sigma = 0.15, \rho = 0.01$	22314.9	22006.2	22060.6	22208.2
$\sigma = 0.15, \rho = 0.05$	22085.2	21687.1	21602.1	21572.3
$\sigma = 0.15, \rho = 0.10$	22097.6	21684.9	21598	21568.7
$\sigma = 0.15, \rho = 0.15$	22065.3	21672.2	21692.5	21595.3
$\sigma = 0.15, \rho = 0.20$	21922.6	21678.9	21654.8	21624.7
$\sigma = 0.20, \rho = 0.01$	22467.6	22233	22256	22426.2
$\sigma = 0.20, \rho = 0.05$	22363.2	21858.2	21774.7	21826.2
$\sigma = 0.20, \rho = 0.10$	22184.4	21901.6	21802.3	21698.4
$\sigma = 0.20, \rho = 0.15$	22186.4	21872.8	21699.8	21622.5
$\sigma = 0.20, \rho = 0.20$	22082.9	21810.9	21768.9	21689.2

d198	$\alpha=1, \beta=2$	$\alpha=1, \beta=3$	$\alpha=1, \beta=4$	$\alpha=1, \beta=5$
$\sigma=0.01, \rho=0.01$	16744.5	16232.8	16198.1	16436
$\sigma=0.01, \rho=0.05$	16152.6	16226.1	16161.5	16117.9
$\sigma=0.01, \rho=0.10$	16188.7	16076.1	16201.7	16098
$\sigma=0.01, \rho=0.15$	16156.6	16003.9	16216.5	16126.4
$\sigma=0.01, \rho=0.20$	16139.2	15994.2	16185.5	16171.8
$\sigma=0.05, \rho=0.01$	17541.3	16833.2	16766.9	16716.5
$\sigma=0.05, \rho=0.05$	17075.30	16483	16259.5	16444.2
$\sigma=0.05, \rho=0.10$	16644.7	16336	16173.4	16328.9
$\sigma=0.05, \rho=0.15$	16177.9	16235.5	16192.4	16240.4
$\sigma=0.05, \rho=0.20$	16072.9	16187.6	16149.5	16194.8
$\sigma=0.10, \rho=0.01$	17641.6	17040.6	16932.2	16838.7
$\sigma=0.10, \rho=0.05$	17497.4	16702.8	16472	16696.2
$\sigma=0.10, \rho=0.10$	17106	16512	16386.4	16464.9
$\sigma=0.10, \rho=0.15$	16940.1	16415.1	16316.4	16460.1
$\sigma=0.10, \rho=0.20$	16775.2	16392.9	16291.8	16420.1
$\sigma=0.15, \rho=0.01$	17655	17207.8	17027.3	16927.5
$\sigma=0.15, \rho=0.05$	17510.8	16840.9	16643.1	16761.1
$\sigma=0.15, \rho=0.10$	17310.4	16627.7	16522.8	16686.1
$\sigma=0.15, \rho=0.15$	17201.1	16587.7	16427.9	16586
$\sigma=0.15, \rho=0.20$	16985.3	16524.5	16374.1	16505.1
$\sigma=0.20, \rho=0.01$	17673	17278.8	17047.1	16895.8
$\sigma=0.20, \rho=0.05$	17496.1	16872.2	16783.6	16827.1
$\sigma=0.20, \rho=0.10$	17354.6	16747.2	16551.1	16714.6
$\sigma=0.20, \rho=0.15$	17323.7	16595.5	16523.6	16681.6
$\sigma=0.20, \rho=0.20$	17244.7	16660.8	16446.3	16607.8

d198	$\alpha = 2, \beta = 2$	$\alpha = 2, \beta = 3$	$\alpha = 2, \beta = 4$	$\alpha = 2, \beta = 5$
$\sigma = 0.01, \rho = 0.01$	16042.7	16009.2	16040.3	16105
$\sigma = 0.01, \rho = 0.05$	16062.6	16041.2	16032.2	16114.6
$\sigma = 0.01, \rho = 0.10$	16109.2	16043	16145.7	16176
$\sigma = 0.01, \rho = 0.15$	16092.8	16061.3	16135.2	16138.8
$\sigma = 0.01, \rho = 0.20$	16091.6	16098.4	16094.1	16144.6
$\sigma = 0.05, \rho = 0.01$	16956.8	16389.6	16243.4	16220.8
$\sigma = 0.05, \rho = 0.05$	15984.6	15998.1	16011.3	16076
$\sigma = 0.05, \rho = 0.10$	16024.2	15994.8	16048	16052.6
$\sigma = 0.05, \rho = 0.15$	16015	15998.7	16026.1	16047.8
$\sigma = 0.05, \rho = 0.20$	16017.6	16045.4	16076.8	16091.1
$\sigma = 0.10, \rho = 0.01$	17399.7	16681	16479.2	16457.3
$\sigma = 0.10, \rho = 0.05$	16451.9	16226.2	16197.7	16160.8
$\sigma = 0.10, \rho = 0.10$	16300.1	16085.9	16082.6	16114.8
$\sigma = 0.10, \rho = 0.15$	16379.6	16139.7	16095	16067.2
$\sigma = 0.10, \rho = 0.20$	16312	16115.9	16055.1	16115.5
$\sigma = 0.15, \rho = 0.01$	17592.1	16855	16626.6	16629.5
$\sigma = 0.15, \rho = 0.05$	16727	16379.2	16304.1	16321.3
$\sigma = 0.15, \rho = 0.10$	16771.8	16287.2	16262.5	16222
$\sigma = 0.15, \rho = 0.15$	16574.8	16220.5	16194.1	16174.3
$\sigma = 0.15, \rho = 0.20$	16494.7	16211.3	16164.8	16136.9
$\sigma = 0.20, \rho = 0.01$	17658.2	16921.6	16721.1	16727.4
$\sigma = 0.20, \rho = 0.05$	16866.3	16425.1	16402.1	16348.6
$\sigma = 0.20, \rho = 0.10$	16815.5	16314.9	16285.1	16306.7
$\sigma = 0.20, \rho = 0.15$	16881.3	16433.4	16286.8	16263.6
$\sigma = 0.20, \rho = 0.20$	16789.7	16355.6	16212.2	16228.1

d198	$\alpha = 3, \beta = 2$	$\alpha = 3, \beta = 3$	$\alpha = 3, \beta = 4$	$\alpha = 3, \beta = 5$
$\sigma = 0.01, \rho = 0.01$	16047.67	15974.9	16034	16064.7
$\sigma = 0.01, \rho = 0.05$	16126.7	16098	16111.7	16129.6
$\sigma = 0.01, \rho = 0.10$	16140.8	16067	16142.4	16210.6
$\sigma = 0.01, \rho = 0.15$	16213.8	16172	16246	16310.6
$\sigma = 0.01, \rho = 0.20$	16191	16054	16235.2	16300.1
$\sigma = 0.05, \rho = 0.01$	16126.7	16034	16062.3	16130.5
$\sigma = 0.05, \rho = 0.05$	16024	16035.7	16039.7	16031.9
$\sigma = 0.05, \rho = 0.10$	16070.6	16033.1	16032.8	16092.3
$\sigma = 0.05, \rho = 0.15$	16081.8	16129.6	16058.1	16107.7
$\sigma = 0.05, \rho = 0.20$	16121.2	16082.6	16077.4	16187.5
$\sigma = 0.10, \rho = 0.01$	16140.8	16470.1	16336.9	16289.9
$\sigma = 0.10, \rho = 0.05$	16409	16166.7	16107.4	16120.7
$\sigma = 0.10, \rho = 0.10$	16468.2	16180.8	16132	16107.8
$\sigma = 0.10, \rho = 0.15$	16465.8	16182	16118.4	16135.4
$\sigma = 0.10, \rho = 0.20$	16580.8	16229.5	16131.7	16102.8
$\sigma = 0.15, \rho = 0.01$	16213.8	16638.6	16462.3	16420.9
$\sigma = 0.15, \rho = 0.05$	16767.4	16370.1	16212.9	16209
$\sigma = 0.15, \rho = 0.10$	16567.6	16228.4	16224	16115.4
$\sigma = 0.15, \rho = 0.15$	16695.4	16324	16174.7	16147.4
$\sigma = 0.15, \rho = 0.20$	16854.8	16333.7	16242.5	16122.9
$\sigma = 0.20, \rho = 0.01$	16191	16832.6	16582.2	16537.5
$\sigma = 0.20, \rho = 0.05$	16748.2	16431	16281.1	16240.5
$\sigma = 0.20, \rho = 0.10$	16916	16471.6	16287.9	16241.2
$\sigma = 0.20, \rho = 0.15$	16903.2	16465.7	16212.2	16223.1
$\sigma = 0.20, \rho = 0.20$	16898.4	16413.6	16231.7	16196.4

d198	$\alpha = 3, \beta = 2$	$\alpha = 3, \beta = 3$	$\alpha = 3, \beta = 4$	$\alpha = 3, \beta = 5$
$\sigma = 0.01, \rho = 0.01$	16047.67	15974.9	16034	16064.7
$\sigma = 0.01, \rho = 0.05$	16126.7	16098	16111.7	16129.6
$\sigma = 0.01, \rho = 0.10$	16140.8	16067	16142.4	16210.6
$\sigma = 0.01, \rho = 0.15$	16213.8	16172	16246	16310.6
$\sigma = 0.01, \rho = 0.20$	16191	16054	16235.2	16300.1
$\sigma = 0.05, \rho = 0.01$	16126.7	16034	16062.3	16130.5
$\sigma = 0.05, \rho = 0.05$	16024	16035.7	16039.7	16031.9
$\sigma = 0.05, \rho = 0.10$	16070.6	16033.1	16032.8	16092.3
$\sigma = 0.05, \rho = 0.15$	16081.8	16129.6	16058.1	16107.7
$\sigma = 0.05, \rho = 0.20$	16121.2	16082.6	16077.4	16187.5
$\sigma = 0.10, \rho = 0.01$	16140.8	16470.1	16336.9	16289.9
$\sigma = 0.10, \rho = 0.05$	16409	16166.7	16107.4	16120.7
$\sigma = 0.10, \rho = 0.10$	16468.2	16180.8	16132	16107.8
$\sigma = 0.10, \rho = 0.15$	16465.8	16182	16118.4	16135.4
$\sigma = 0.10, \rho = 0.20$	16580.8	16229.5	16131.7	16102.8
$\sigma = 0.15, \rho = 0.01$	16213.8	16638.6	16462.3	16420.9
$\sigma = 0.15, \rho = 0.05$	16767.4	16370.1	16212.9	16209
$\sigma = 0.15, \rho = 0.10$	16567.6	16228.4	16224	16115.4
$\sigma = 0.15, \rho = 0.15$	16695.4	16324	16174.7	16147.4
$\sigma = 0.15, \rho = 0.20$	16854.8	16333.7	16242.5	16122.9
$\sigma = 0.20, \rho = 0.01$	16191	16832.6	16582.2	16537.5
$\sigma = 0.20, \rho = 0.05$	16748.2	16431	16281.1	16240.5
$\sigma = 0.20, \rho = 0.10$	16916	16471.6	16287.9	16241.2
$\sigma = 0.20, \rho = 0.15$	16903.2	16465.7	16212.2	16223.1
$\sigma = 0.20, \rho = 0.20$	16898.4	16413.6	16231.7	16196.4