


東 海 大 學

工業工程與經營資訊研究所

碩士論文

蜂群最佳化之實數問題求解研究



研 究 生：莊惟丞
指 導 教 授：張炳騰 教授
曾宗瑤 教授

中 華 民 國 九 十 八 年 六 月

Study of Real Function Problem Solution of MBO

By
Wei-Chen Chuang

Advisor: Prof. Ping Teng Chang
Prof. Tsueng-Yao Tseng

A Thesis
Submitted to the Institute of Industrial Engineering and Enterprise
Information at Tunghai University
in Partial Fulfillment of the Requirements
for the Degree of Master of Science
in
Industrial Engineering and Enterprise Information

June 2009
Taichung , Taiwan , Republic of China

蜂群最佳化之實數問題求解研究

研究生：莊惟丞

指導教授：張炳騰 教授

曾宗瑤 教授

東海大學工業工程與經營資訊研究所

摘要

啟發式演算法發展至今已有許多相當成熟的模型方法，然而尚無一通用的演算法可解決所有的問題函數，每種方法皆有其利弊。故在實際應用上為了截長補短，衍生出許多複合型態的方法。而蜂群最佳化則是藉由模仿蜜蜂繁衍的行為模式所發展的一套演算法則，利用蜜蜂繁衍的機制達到演化最佳。目前被大量應用在最佳化、排程、分群、資料探勘等議題。其結合了退火函數、交配操作，如此全域搜尋的特性，保有求解的複雜度；另與其他單純的搜尋方法結合，在區域搜尋上做競爭，面對不同問題能有更好的適應能力，達到快速收斂的效果。

本研究針對不同函數特性與數學環境，探討蜂群最佳化的演化方法，歸納其求解特性，瞭解其中不同啟發式演算法組合邏輯。並提出實數問題的參數設定，經由實驗驗證的結果顯示，蜂群最佳化在實數變數底下也能有相當好的收斂效果。證明如此的結合方法，不論單一或多重區域解，皆能擁有優秀的全域搜尋能力以及穩定的最佳解收斂效率，是一優良的演算法。

關鍵字：蜂群演算法、複合型啟發式演算法、函數特性、全域搜尋、區域搜尋

Study of Real Function Problem Solution of MBO

Student : Wei-Chen Chuang

Advisor : Prof. Ping-Teng Chang
Prof. Tsueng-Yao Tseng

Institute of Industrial Engineering and Enterprise Information
Tunghai University

ABSTRACT

Heuristic algorithm develop have a lot of familiar model method already so far, but does not have a common algorithm that can solve all problem functions yet. Each kind of method all has its advantages and disadvantages. In order to draw on the strong points to offset the weaknesses in practical application, derive a lot of methods of compound attitude out. Marriage in Honey Bees Optimization (MBO) is by imitating a set of algorithms of development of manner that the honeybee multiplies. The mechanism of utilizing honeybees to multiply reaches and evolves the best. Apply optimization to, production schedule, clustering, data mining topic that in a large amount at present. It has combined annealing function and mating operation, such characteristic of global searched, keep the complexity solved. Bind with other simple search patterns separately, do the competition in local searching, there can be better adaptive capacity in the face of different problems, get the result of convergence fast.

This research directs against different function characteristics and mathematics environment, probe into the evolution method that MBO, sum up it ask characteristic of solving and understand with heuristic to perform algorithm make logic up it. The parameter of putting forward the real number question is set up, the result that proves via the experiment reveals, the MBO in the real number parameter there can be quite good convergence results too. Prove the combination method like MBO, no matter solve in the single or multiple local solutions, there can all be outstanding global search capability and steady solving efficiency. It is a fine algorithm of performing.

Keywords: Marriage in Honey Bees Optimization; Multi-Heuristic algorithm; Function Characteristic; Global search; Local search.

誌謝

論文得以順利完成，首先要感謝我的指導教授張炳騰老師，在他細心的指導下，給予我在寫論文時有相當大的幫助及想法，另外對於未來人生的方向與規劃，也給予我諸多的指引，亦師亦友的情誼與師恩，永誌難忘。口試期間，承蒙曾宗瑤老師、洪堯勳老師、白炳豐老師及時序時老師在百忙之中特別撥空審閱論文，提供寶貴的意見，使我的論文得以更加完整，也讓我瞭解做研究所必須有的嚴謹態度，由衷的感謝各位老師。

在研究過程中，感謝研究室同窗的幫忙及相互扶持，一起走過最艱難的時光，讓我從一個陌生的環境中成長與學習。此外，感謝學弟妹的鼓勵及支持，使我在低潮時依舊能擁有歡笑。

最後，特別要感謝父母及家人的關懷與照顧，使我能在無憂無慮的環境下，順利完成學業。

在此，僅以本論文獻給所有關心我及支持我的人，謝謝你們。

莊惟丞 謹誌於

東海大學工業工程與經營資訊學系

民國九十八年六月

目錄

摘要.....	i
ABSTRACT.....	ii
誌謝.....	iii
目錄.....	iv
圖目錄.....	vi
表目錄.....	vii
第一章 緒論.....	1
1.1 研究背景與動機.....	1
1.2 研究目的.....	1
1.3 研究流程與架構.....	2
1.4 研究工具.....	3
第二章 文獻討論.....	4
2.1 演化式計算(Evolutionary Computation; EC)相關文獻.....	4
2.1.1 演化式演算法(Evolutionary Algorithm; EA).....	6
2.1.2 群智慧(Swarm Intelligence).....	11
2.2 蜂群最佳化(Marriage in Honey Bees Optimization; MBO).....	13
第三章 實數型蜂群最佳化.....	16
3.1 MBO 之步驟.....	16
3.2 MBO 之結構.....	22
第四章 實驗結果與效能分析.....	27
4.1 測試函數.....	27
4.2 MBO 參數實驗.....	30
4.2.1 參數設計.....	30
4.2.2 實驗結果.....	31
4.3 比較對象與之參數設定.....	34
4.3.1 與 $(\mu + \lambda)$ ES 比較.....	36

4.3.2	小結.....	37
4.4	MBO 於各函數特性分析.....	37
4.4.1	Function 1 : Sphere Model.....	38
4.4.2	Function 2 : Schwefel's Problem 1.2	40
4.4.3	Function 3 : Rotated High conditioned Elliptic Function.....	43
4.4.4	Function 4 : Generalized Rosenbrock's Function	45
4.4.5	Function 5 : Generalized Rastrigin's Function	46
4.4.6	Function 6 : Ackley's Function	49
4.4.7	Function 7 : Generalized Griewank Function	52
4.4.8	小結.....	54
第五章	結論.....	56
5.1	研究總結.....	56
5.2	後續研究建議.....	56
	參考文獻.....	58

圖目錄

圖 1. 1 研究方法與步驟.....	3
圖 2. 1 演化式計算之發展結構.....	5
圖 2. 2 蜂群優化演算法運作流程	14
圖 3. 1 MBO 之架構圖	17
圖 3. 2 蜂群最佳化的虛擬碼.....	22
圖 3. 3 MBO 結構概念	23
圖 4. 1 Sphere Model 三度空間二維示意圖	38
圖 4. 2 Function 1 各方法收斂比較	39
圖 4. 3 Finction 1MBO 各階段收斂情形	40
圖 4. 4 Schwefel's Problem 1.2 三度空間二維示意圖	41
圖 4. 5 Function 2 各方法收斂比較	42
圖 4. 6 Function 2 MBO 各階段收斂情形	42
圖 4. 7 Rotated High conditioned Elliptic Function 三度空間二維示意圖 ...	43
圖 4. 8 Function 3 各方法收斂比較	44
圖 4. 9 Finction 3 MBO 各階段收斂情形	44
圖 4. 10 Generalized Rosenbrock's Function 三度空間二維示意圖	45
圖 4. 11 Function 4 各方法收斂比較	46
圖 4. 12 Generalized Rastrigin's Function 三度空間二維示意圖	47
圖 4. 13 Function 5 各方法收斂比較	48
圖 4. 14 Finction 5 MBO 各階段收斂情形	48
圖 4. 15 Ackley's Function 三度空間二維示意圖	49
圖 4. 16 Function 6 各方法收斂比較	51
圖 4. 17 Finction 6 MBO 各階段收斂情形	51
圖 4. 18 Generalized Griewank Function 三度空間二維示意圖	52
圖 4. 19 Function 7 各方法收斂比較	53
圖 4. 20 Finction 7 MBO 各階段收斂情形	54

表目錄

表 2.1 MBO 相關應用文獻	14
表 3.1 MBO 與 SA、GA 的異同.....	26
表 4.1 測試函數與函數特性.....	27
表 4.2 最佳解、搜尋範圍定義.....	29
表 4.3 MBO 實驗比較參數表	30
表 4.4 子代數對 MBO 的影響	33
表 4.5 各函數速度遞減比例與初始能量收斂時間	34
表 4.6 PLES 實驗參數	35
表 4.7 MBO 實驗參數	35
表 4.8 MBO 實驗結果	36
表 4.9 ES、PLES、MBO 平均效能	37
表 4.10 MBO 於 Sphere Model 各階段效能.....	38
表 4.11 MBO 於 Schwefel's Problem 1.2 各階段效能	41
表 4.12 MBO 於 Rotated High conditioned Elliptic Function 各階段效能....	44
表 4.13 MBO 於 Generalized Rosenbrock's Function 各階段效能.....	46
表 4.14 MBO 於 Generalized Rastrigin's Function 各階段效能	47
表 4.15 MBO 於 Ackley's Function 各階段效能.....	50
表 4.16 MBO 於 Generalized Griewank Function 各階段效能.....	53
表 4.17 函數特性與 MBO 效能比較表	55

第一章 緒論

1.1 研究背景與動機

現實世界中眾多的問題最終仍歸類為最佳化問題，最佳化問題存在於各個領域，對於其求解的方式，過去已有不少經典的演算法被提出，對於某些特定的問題的確能得到較好的結果。

近年來快速發展的演化計算，為一種借鑑於達爾文進化論而模擬出的一套隨機演算法，舉凡觀察自然、生物、與生命現象為手段，得到可應用於搜尋(Search)、設計(Design)、與最佳化(Optimization)等各類問題之演算法，皆屬於演化計算的研究領域。演化計算著眼於從一群潛在解中尋找問題的最佳解，透過在當前的潛在解之間反覆地進行如篩選、重組、突變等遺傳操作，來對於最佳解所在的位置逐步逼近。演化計算在於應用問題方面，至今已成功的運用在許多不同領域的最佳化問題上，如藥物設計、生物資訊、演化硬體、工業設計、電路設計、資料探勘、財經分析等。

蜂群最佳化(Marriage in Honey Bees Optimization; MBO)則是近年來新興發展的演算法之一，自提出後廣泛應用在排程、分群、資料探勘等議題，皆有不錯表現。然而蜂群最佳化自 2001 年 Abbass and Teo 提出以來皆應用在離散型變數之問題求解，尚無連續型實數問題之應用。

1.2 研究目的

啟發式演算法發展至今已有許多相當成熟的模型方法，然而至今尚無一通用的演算法可解決所有的問題函數，每種方法皆有其利弊。故在實際應用上為了截長補短，衍生出許多複合型態的方法。而蜂群最佳化則是近年來新興的演算法之一，其結合了退火函數、交配操作，如此全域搜尋的特性，保有求解的複雜度；另與其他單純的搜尋法則結合，達到在區域搜尋上快速收斂的效果。

模擬退火利用溫度高低決定全域與區域搜尋的階段改變，基因演算法透過多點搜尋與隨機的方式達到區域解的跳脫，然而面對不同的數學函數環境特性，兩者皆得做參數上的設計與改善。蜂群最佳化結合兩種演算法特性，使之在全域(廣度)與區域(深度)兩個搜尋方向達到有效平衡。

本研究將針對蜂群最佳化求解原理與各項特性，提出連續型實數問題之參數設定；透過實證函數探討的過程，歸納此演算法特性及適用的數學環境，佐以實數變數的函數驗證，以利將來實數應用領域之延伸。

1.3 研究流程與架構

本研究分為以下幾個步驟，研究的流程與架構如圖 1.1：

1. 確立研究目標

確立研究範圍與目標。

2. 文獻探討

藉由相關文獻的回顧，了解目前相關研究領域上的成果，作為研究過程的參考。

3. 研究方法

本章討論蜂群最佳化之原理結構，並詳細描述其模型之步驟。

4. 模式實作與結果分析

本研究以7個多變數函數範例來進行實驗驗證與實驗設計，實驗結果也將與近年來發表在國際期刊上之結果一併做比較。並探討蜂群最佳化的演算邏輯，歸納其搜尋法則。

5. 結論與建議

本章為本研究論文之結論，並對本研究之建議及未來之研究方向作一概括描述。

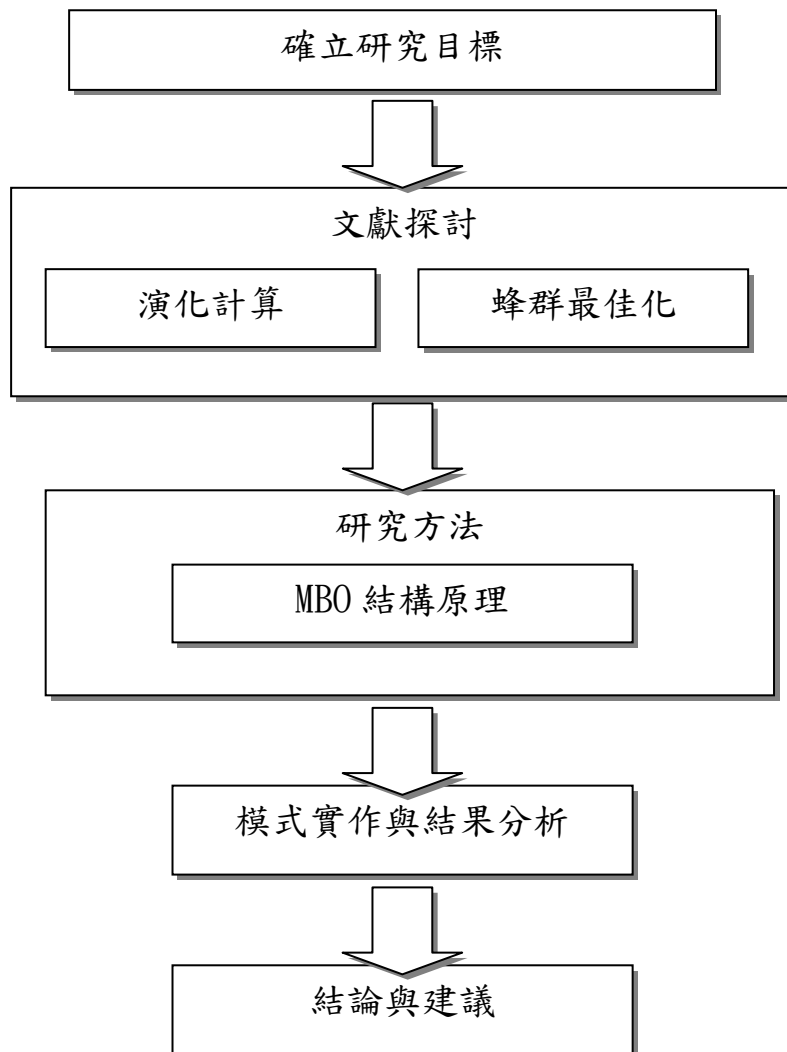


圖 1.1 研究方法與步驟

1.4 研究工具

本研究使用 Matlab 來作為運算工具，利用 Matlab 的特性，開發出一套合理且可行的程式，並利用本研究中的範例求出最佳的解。Matlab 是 Matrix Laboratory 的簡寫，它是一個直譯式的語言程式，與其他的程式語言比較起來，因為語法較於單純，所以較容易學習與瞭解。Matlab 的主要特色在於數值分析、矩陣的運算與資料型態的轉換，它能夠讓使用者即使面對大量的資料以及冗長的運算問題時，也能有效的解決，因此，本研究選擇 Matlab 來作為演化最佳化問題的解決工具。

第二章 文獻討論

2.1 演化式計算(Evolutionary Computation; EC)相關文獻

理論基礎可回到自 1859 年達爾文(Charles Darwin)的「物種演化」(On the Origin of Species by Means of Nature Selection)書中的「物競天擇，適者生存」的演化及淘汰觀念。60 年代於三個不同的領域各自獨立發展。在美國，L. Fogel 提出演化式規劃，Holland 提出遺傳演算法；在德國，Rechenberg 則稱作為演化策略。這些領域個別發展了約 15 年，直至 90 年代初期才統稱此類的方法為演化計算。同時基因規劃也才產生。

由數值分析的角度來看，演化式計算可視為類似隨機搜尋(random search)的最佳化演算法(optimization)。因為其本質是非確定性的(non-deterministic)，所以能搜尋問題空間裡所有的可行解。而在演化式計算的群體中，大量個體追隨環境利基(niche)，聚集而形成各式族群(species)。不同族群也就代表了用以解決問題的不同途徑與其資訊——這就是演化式計算含有多樣性(diversity)以解決問題的能力。尤其在大部份複雜非線性的數學問題上，因為缺乏解析決定的求解方法，演化式計算便成為最有效的數值分析工具。

另外由自然演化的角度來看，演化是在動態變化的環境下的一種適應(adaptation)過程，相對於靜態環境，演化過程中的系統會趨向複雜(complexity)。因此演化可視為適應性複雜系統(adaptive complex system)的變化過程，同時也是在生命與智慧本質中的必要基本過程。

在大自然裡的生物並非由單一簡單的機能形成，而是由大量的個體聚集的複雜系統所構成。生物形成的複雜系統會隨著環境改變，而尋求生存以適應環境。所以生命現象以及由其衍生的智慧行為，可以視為是適應性複雜系統的表現。由適應性複雜系統的觀點，可以歸納出四項生命智能的基本特質——複雜、演化、交感、適應。這四項特質彼此相依相存，只是生命本質不同呈現方式不同。而人工智慧(artificial intelligence)的各種模型就包含了這些特質的簡化與結合。

1. 複雜結構：非線性個體組成的資訊結構。藉由非線性結構，表現複雜的環境條件，即組合最佳化問題(combinatorial optimization problems)。

2. 演化過程：自發突現的形態變化。效仿生命演變形態，於可行解空間做全域搜尋(global search)。比之傳統的數學線性方法求解，能有效節省運算時間。
3. 交感機制：協調一致的交互感應。自然界中，除了生命本質之演化能力，物種個體同時也存在著環境適應方法。而人工智慧則多以區域搜尋(local search)來顯現。
4. 適應行為：適應環境變遷以求生存。經過前面兩個過程與機制產生新世代，就達爾文適者生存之原理，加以篩選淘汰，進而留下最佳適應函數，達到求解最佳化之目的。

演化計算主要的主流可分為演化式演算法(Evolutionary Algorithms)與群智慧(Swarm Intelligence)。

- 演化式演算法：底下模型包含有基因演算法(Genetic Algorithms)、演化規劃(Evolution Programming; EP)、演化策略(Evolution Strategy)、與基因規劃(Genetic Programming)。
- 群智慧：包括粒子群最佳化(Particle Swarm Optimization; PSO)、蟻群最佳化(Ant Colony Optimization; ACO)。

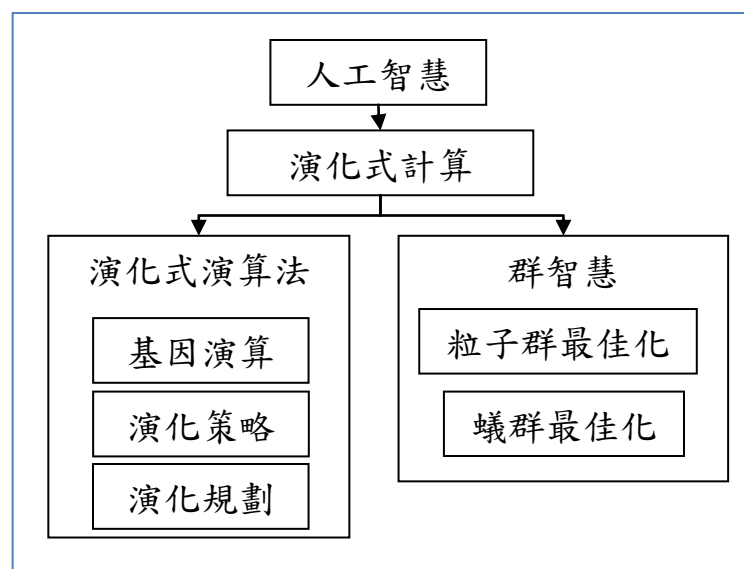


圖 2.1 演化式計算之發展結構

2.1.1 演化式演算法(Evolutionary Algorithm; EA)

• 基因演算法(Genetic Algorithm; GA)

基因演算法於 1975 年由密西根大學教授 John Holland 所提出。當初主要的訴求為：(1)抽象且嚴厲地解釋自然系統適應過程；(2)設定人工系統軟體，保留自然系統的重要機制。藉著生物對複雜自然環境的適應度以及相應產生的演化機制，模仿自然界生物基因遺傳與淘汰過程，建立一個具備自然界演化機制：交配、突變與自我複製的人工系統。使得系統本身具備自我演化的能力，朝著最佳解演化。生物演化的目的在於比上一代更適應目前的生活環境，而不是朝著一個已知的最佳解移動。基因演算法就是要把生物界演化的機制抽象出來應用在解決搜尋最佳解的問題上面，讓系統朝著更佳的解自我演化。

以基因演算法來搜尋最佳解的問題時，已有眾多學者討論過，尤其是於基因演算法運作步驟也有許多學者進行探討，如：決定參數編碼(Encode)方式(Michalewicz, 1992)；設定控制參數，如：交配率、突變率以及停止運算的條件等。(Michalewicz et al., 1994, Murata and Ishibuchi, 1994)

基因演算法主要運用三種基本運算來模仿自然界的生物演化過程，包括天擇(Selection)、交配(Crossover)及突變(Mutation)，而在進行這些運算時，必須先決定編碼(Encoding)與適應函數，各步驟分別說明如下。

1. 編碼：在進行遺傳演算法之前，第一個步驟是將現實的問題表示為演算法所能處理的型態，編碼的設計就好比生物的染色體結構，在據此設計適應函數。一般的編碼方式係採二進位(Binary)，另外尚有整數(Integer)、實數(Real)及文字(Alphabet)等編碼方式。在 GA 中需將問題的解答表示成可以運算的形式，即染色體編碼。大部分最佳化問題都有固定數目的變數，最普遍的編碼方是就是將這些變數對應到某個字元或整數，將其編成固定數目的位元，這些位元組合起來後就成為一個染色體，GA 即運用此編碼後的染色體進行後續運算。
2. 適應函數：染色體解碼後代入適應函數即可求得適應值，適應函數主要用以評估每個染色體的優劣，並指出族群演化的發展方向，通常適應函數為問題最佳化的目標函數，但有時目標函數需經過正規化處理。
3. 天擇：天擇運算子隨機從母代族群中挑選染色體，以繁衍子代。染色體

被選中的機率與其適合度函數值成比例。適合度函數值越高，其被選擇的機率就會越高；反之，則被選擇的機率越低。選擇機制係模擬自然界適者生存的機制，根據染色體的適應值來決定參與交配的染色體族群。常見的方法有：輪盤法(Roulette Wheel Selection)、期望值法(Integral Roulette)、均勻法(Uniform Selection)、競賽法(Tournament Selection)、穩態法(Steady-state Selection)與菁英法(Elitism)。

4. 交配：生物界中，較高等的生物都是藉由交配來產生下一代。基因演算法中，藉由母代染色體間的基因互換，來模仿生物交配的行為。在交配池中的母代染色體會隨機的配對，互換基因(Gene)，以繁衍下一代。藉由這種方式，染色體間可互換資訊，以使得子代染色體能有更佳的適合度函數值。交配是將兩個染色體的遺傳作隨機交換，以產生新的子代(Offspring)，其效果猶如在解答空間中作大幅度的跳躍搜尋。
5. 突變：在自然界中個體有時會因為外在因素的刺激，而使本身的基因產生改變。突變可避免個體在交配的過程當中，失去一些重要的資訊，除此之外，藉由突變，也可以增加個體間的變異性，它讓生物有機會發展出具新特性的個體，使其更具多樣性。

基因演算法本身為一最佳化求解工具，也是一種極有效率的搜尋程序。GA 特別適合於有效解答空間很大、非線性、可能有雜訊、甚至無法確定可解的問題上，是傳統決定性最佳化(Deterministic Optimization)或貪婪法則(Greedy Heuristics)無法做到的，而且具有高度的問題獨立性，這點與依附於問題模式的傳統演算法有所不同。

基因演算法在尋找最佳解時，每次以一個族群為單位，因此可平行搜尋數個答案，而且 GA 的運算過程大多會加入隨機值的影響，因此有較高的機會避開區域最佳解(Local Optimum)的陷阱。GA 的限制在搜尋解答空間時，不保證一定可以找到真正的最佳解，而是在搜尋最佳解到某種程度的可能最適解，或稱為滿意解。

• 演化策略(Evolution Strategy; ES)

1965 年，德國 Rechenberg 為了解決流體力學中，模型控制裡實數參數最佳化的問題。發展了第一個實驗性的演化策略方法，這就是演化策略的

開端。因為體認到工程上的一些非線性模型的數值問題，無法用傳統的數學線性方法求解。後來他與 Schwefel 合作發明了一種新的方法運用電腦以求解決問題，就是『演化策略』。

其研究發展完全建立在嚴密的數值分析與機率理論上，每個演化策略單元由一組實數數值參數所構成。因為 60 年代的電腦記憶容量還很小，所以最早的演化策略非常簡單，類似隨機搜尋，被稱為「1 + 1 演化策略」，其中只有兩個單元。Rechenberg 並以機率的觀點，發展了關於突變的 1/5 成功率規則。隨著電腦記憶的增大，又再發展了多元演化策略——「 $\mu + \lambda$ 演化策略」和「 μ, λ 演化策略」。更使控制演化的策略參數，也加入隨著各單元一同演化，至此奠定雛型。以後便在德國獨立發展為一種以演化解非線性數值問題的方法，直到 1990 年才透過與遺傳演算法研究團體交流傳到美國，又到 1992 年才與研究演化規劃的團體展開交流。

根據生物演化的現象，Rechenberg 歸納以下的結論：『演化會使生物過程達到最佳化，而演化本身也是一種生物過程，所以演化必然使本身也達到最佳化。』這種探討關於演化本身的演化，也就是考慮到演化的策略。在演化策略的機制中，單元染色體的重組是發生在表現型，而非在基因型。其實在演化策略中，所謂的表現型就是意味各單元中的實數參數，而基因型是指在電腦中構成實數的位元，其中的重組是以實數參數為基本單位。而非如遺傳演算法一般，以位元為基本單位。這種觀點導致關於在實數參數中，突變強度的探討，引出了演化窗(evolution window)的觀念。因此在演化策略的程序裡，不止是群體中各單元參數演化，也將突變強度的策略加入演化，調整使其趨入演化窗的區間。這種演化過程被稱為移變演化(meta-evolution)或第二類最佳化(optimization of second kind)。

演化策略與二位元基因演算法(Binary GAs)之間最大的不同在於演化策略的操作使用實數參數(real parameter value)，以及初期的演化策略只作天擇(selection)的操作而不使用基因交換相關的操作方式。演化策略是從母代(parent)以一固定的突變強度(mutation strength)產生子代(offspring)後再作天擇，而為了使演化策略的演化更有彈性，發展出自我適應能力的演化策略(self-adaptation evolution strategies)，近來演化策略發展出一些類似基因交換的操作方式，稱為重構(recombination)，主要目的在整個族群中製造差異，藉以產生優良且具適應能力的下一代，因此，對於演化策略的討論主

要可以分為天擇、自我適應能力與重構三個部分。

天擇(selection)最初的演化策略的結構為(1+1)-ES (Rechenberg, 1973, 1994)，由一個母代使用高斯突變操作(Gaussian mutation operator)產生一個子代，在母代與子代中選取較佳的值成為新的母代，而 Schwefel(1981)提出調整天擇中 σ 的更新規則，Schwefel and Bäck (1998)提出天擇建議參數。

(μ, λ)-ES 與 ($\mu + \lambda$)-ES 是演化策略中基本的兩種天擇(selection)的形式， μ 代表母代(parent)的大小(size)， λ 代表每一代的演化產生的子代(offspring)數量，且 $\lambda > \mu$ 。(μ, λ)-ES 與 ($\mu + \lambda$)-ES 之間最大的差異點在於選取新的母代時，(μ, λ)-ES 是由 λ 個子代中選取最好的 μ 個作為新的母代；($\mu + \lambda$)-ES 是由 μ 個母代與 λ 個子代中選取最好的 μ 個作為新的母代。Schwefel 與 Rudolph(1995)提出(μ, κ, λ)-ES 架構， κ 代表演化代數，若母代存活代數大於 κ 則保留母體， $1 \leq \kappa \leq \infty$ 。若 $\kappa = 1$ ，則天擇方法(selection method) 會成為(μ, λ)-ES 的型態；若 $\kappa = \infty$ ，則天擇方法會成為($\mu + \lambda$)-ES 的型態。Schwefel 與 Bäck (1998)提出的建議為 $\kappa \geq 1$ ，用以允許生命的展延。

重構(recombination)中常使用($\mu / \rho, \sigma$)-ES(Beyer, 1995)的結構作天擇方法，其中 ρ 代表作重構操作時，由 ρ 個母代重構出 λ 個子代，再由 μ 個母代與 λ 個子代中選取最好的 μ 個作為新的母代。近來有些學者使用實參數基因演算法 (real-parameter genetic algorithms) 的重構方法 (recombination operator)，例如 Eshelman and Schaffer(1993)提出的 BLX(blend crossover)、Deb and Agrawal(2002)提出的 SBX(simulated binary crossover)、Ono and Kobayashi(1997)提出的 UNDX(unimodal normally distributed crossover)、Higuchi (2000) 提出的 SPX(simplex crossover)、Deb (2002) 提出的 PCX(parent-centric recombination)。

自我適應能力(Self-Adaptation)主要的自我適應能力的演化策略可以分為以下四類：

1. 等向的自我適應能力(Isotropic Self-Adaptation)

等向的自我適應能力主要使用一個學習參數 τ_0 以及突變強度的更新規則增強演化效率 Schwefel(1987)提出學習參數 τ_0 最佳設定值應為： $\tau_0 \propto n^{-1/2}$ ， n 為變數的數量。Beyer (1995)提出在 sphere model 中，對於(1, λ)-ES 最佳的學習參數 $\tau_0 = c_{1,\lambda} / \sqrt{n}$ ， $c_{1,\lambda}$ 為行進係數。

2. 非等向的自我適應能力(Non-Isotropic Self-Adaptation)

非等向的自我適應能力對於每個不同的變數使用不同的突變強度。Beyer(1995)提出對於球體模型(sphere model)最佳的學習參數設定為 $\tau_0 \propto 1/\sqrt{n}$ ，可得到最大的收斂速率。

3. 關聯式自我適應能力(Correlated Self-Adaptation)

關聯式自我適應能力對於每個不同的變數使用不同的突變強度，因此最多存在 $\binom{n}{2}$ 個共變數，而共有 $\left(\binom{n}{2} + n\right)$ 個外部的策略參數(exogenous strategy parameter)需作更新與調整，關聯式自我適應能力的更新規則為：在 n 個變數、 n 個突變強度 σ_i 與 $\binom{n}{2}$ 個 α_i 的旋轉角度的設定下，Schwefel(1987)與 Bäck (1996)提出參數 β_α 固定為 0.0873(或 5°)。

4. CMA-ES

CMA(The Covariance Matrix Adaptation) 由 Hansen and Ostermeier (1996)提出，以 Covariance matrix 記錄整個族群數代演進的歷史，計算出目標變數變異數與共變數，而搜尋成果受變異數的影響。Hansen and Ostermeier(2001)在接受 Ingo Rechenberg(1998)的建議，從 λ 個子代中以權重重構(weighted recombination)的方法找出最佳的 μ ，將 CMA 改寫為 (μ_w, λ) -CMA_ES。

• 演化規劃(Evolution Programming; EP)

在 1960 年代，L. Fogel(1962)由於想要發展和傳統人工智能中專家系統不同的模型，以便能消除系統對人為設計的依賴，而能自我調適。由演化的觀點出發，他將智能視為一種天擇的產物。所以不像專家系統需要模擬人類的思考行為，而是直接讓系統演化出所須的行為模式。

因為由行為的角度進行演化，所以他以有限狀態機(finite state machines)的模型來構成表現型單元。有限狀態機的邏輯機制就代表其行為，完全不考慮基因型的問題。L. Fogel 所考慮的是行為演化，因此他在早期只採用突變來產生各單元行為的變化。在演化機制上，L. Fogel 特別強調演化規劃是由上而下的演化方法，和遺傳演算法是由下而上的演化方法有所不同。

L. Fogel 將智慧行為視為『可以預測環境，並根據預測採取適當反應以達成目的地的能力。』早期的演化規劃因此採用有限狀態機作為預測及反應的單元，而其環境就是一連串可供有限狀態機操作的符號序列。根據輸入符號，單元必須反應一輸出符號。單元對符號的反應，將可以自環境中獲得報酬。而各單元必須演化其有限狀態機使所獲得的報酬趨向最大，否則就可能在天擇中被淘汰。因此留存的單元會愈來愈適應環境，而且其複雜度也會愈來愈增加。

在 1960 到 1976 年間，演化規劃雖然在研究應用上有些進展。但人工智慧學界對於演化規劃大多抱以懷疑的態度。由 1977 到 1985 年間，只有極少量的研究被發表。近年來經由 D. Fogel 的大力提倡，現今的演化規劃不再限定以有限狀態機作為單元結構。而是根據所對應的問題，來設定單元結構和突變操作。因此在處理數值問題的演化規劃，所採取的機制就和演化策略很類似。現在已被廣泛用在組合最佳化、函數最佳化、自動控制、遊戲理論、路徑規劃等問題上。

2.1.2 群智慧(Swarm Intelligence)

- **蟻群最佳化(Ant Colony Optimization; ACO)**

蟻群最佳化在 Dorigo et al.(1991, 1992)提出，稱之為蟻群系統(Ant System)。是藉由自然界螞蟻尋找食物的精神發展出一套演算法則，並求解蟻群系統驗證旅行銷售員 traveling sales man (TSP)問題。蟻群系統發表後，被廣泛的應用在許多離散事件的最佳化問題上。經過改良演進，產生許多強化的蟻群系統。重要的有 Gambardella and Dorigo(1995)提出 Ant-Q 演算法，採用不同於蟻群系統的狀態轉移法則，費洛蒙軌跡更新方面採用 Q-learning 的方式，更有效的強化費洛蒙值的影響。Bullnheimer et al.(1997)提出選擇固定數量優秀螞蟻，來更新費洛蒙軌跡的 ASrank 演算法。綜合許多對蟻群系統的研究，Dorigo and Maria (1996、1997)提出改良於螞蟻系統的蟻群法 Ant Colony System (ACS)，至此蟻群法成為最廣泛被使用於求解各種最佳化問題。Stutzle and Hoos(2000)提出費洛蒙值受到上下界的限制，以減少停滯現象(Stagnation)的費洛蒙軌跡更新演算法 Max-Min Ant System (MMAS)。有鑒於蟻群系統變化與衍伸出來的許多演算法已經可以

有效的求解許多類型組合最佳化問題，Dorigo et al.(1999)將蟻群系統、蟻群法與相關應用的法則歸納成一套啟發式演算法 Ant Colony Optimization Meta-Heuristic。以這套演算法為精神的統稱為蟻群最佳化 Ant Colony Optimization(ACO)。

另外關於連續性問題的應用包括有：1995年 Bilchev 與 Parmee 首先提出以分段向量定義連續型的 ACO；2000年 Mathur et al.將原本的 ACO 搜尋切割為全域與區域搜尋，之後陸續還有 Monmarche et al.的 API(2000)、Dreo 和 Siarry 的 Continuous Interacting Ant Colony; CIAC(2002)，但以上方法所應用之概念皆與原本的 ACO 不同，直至 2004年 Socha 提出了連續與混合變數的蟻群最佳化，2008年 Socha and Dorigo 做出最新的統整，應用機率密度函數(probability density function)之概念，將原本的蟻群最佳化帶入高斯函數將離散變數予以連續值，提出 AOC for continuous domains(ACOR)。

- **粒子群最佳化(Particle Swarm Optimization; PSO)**

最早是由 Kennedy and Eberhart(1995)提出來的最佳化推測演算技術，其概念乃是模仿自然界中鳥類搜尋食物的機制而設計出的演算法。後續不斷有相關的改良方法提出，例如，Shi and Eberhart(1998)提出降低粒子移動速度的方法，在演化更新時乘上一個慣性因子(inertia weight)，該研究指出當慣性因子大於 1.2 時較具有探索的能力，當小於 0.8 時較容易產生群聚效應。之後在 2002年 Fourie and Groenwold 將動態(Dynamic)的觀念引入慣性因子以及速度上限(Maximum velocity)，加速其收斂效果(Fourie and Groenwold, 2002)；同年另外有人提出一個方法，以緊縮因子(constriction factor)來取代之前所提出的慣性因子，亦可達到不錯效果(Clerc and Kennedy, 2002)；後續另有 Bergh and Engelbrecht(2004)以協同作業的概念改良 PSO 的方法。

粒子群演算法建立在群體(Swarm)間協同呼應的組織性運作機制上，演化的過程中群體的每個粒子(Particle)會不斷地在解空間內探索，而且本身能記憶自己在嘗試過程中最好的值，我們稱之為最佳值(Particle best value; pbest)，此外，粒子群彼此之間也能夠傳遞群體中最好的粒子的位置，這種全域最佳粒子我們稱之為群體最佳值(Global best value; gbest)，或是鄰近地

區最好的粒子所在位置，這種區域最佳粒子我們稱之為區域最佳值(Local best value; lbest)，最佳粒子群集演算法尋優步驟就是依據 pbest 及 gbest(或 lbest)來決定每個粒子所要探索的方向，藉由尋優步驟使得整體的粒子群集逐漸接近在解空間內的目標位置。

2.2 蜂群最佳化(Marriage in Honey Bees Optimization; MBO)

在自然界中，許多生物展現許多不可思議的社會行為與自我組織的智慧，例如：螞蟻群落、候鳥遷徙與覓食、魚群為了逃避捕食者所產生的群流效應等。這些行為簡單個體所組成的群落中，並沒有領導者更沒有集中式的管理，靠的只是個體在環境裡與其他個體互動行為的簡單規則。而蜂群最佳化演算法(MBO)也算其中之一。蜂群優化演算法在 2001 年由 H.A. Abbass (2001)所提出，是藉由觀察蜂群的繁殖行為所建立的尋找最佳解的一種演算法。其主要特色有三：工蜂分工照顧卵巢、至少兩個世代重疊與生殖。典型的蜂群只有一隻蜂后，由零至上千隻的雄風與上萬隻的工蜂環繞，一個族群可能經歷一任或數任的蜂后。Abbass 所提出的蜂群最佳化演算法中，蜂后的特性是利用精巢(spermatheca)，經過一段飛行沿途蒐集各雄蜂的精子，之後回至蜂巢中結合產生下個世代，新一代的幼蟲在蜂巢中，經過工蜂看護與照料成長茁壯；其中結果若優於蜂后則取而代之，若無則全部子代皆被殺滅。。

蜜蜂巢中由蜂后(Queens)、雄蜂(Drones)、工蜂(Workers)與幼蟲(Broods)所組成，藉由蜂后與雄蜂的交配行為，來產生下一代蜂卵。再經由不同工蜂的照顧(區域搜尋)，使蜂卵變成幼蟲且逐漸茁壯。如果新的幼蟲在經過工蜂的照顧後，能力(目標值)比原本的蜂后好，即馬上取代原本的蜂后。蜂群優化演算法(MBO)的運作流程如下圖 2.2。

Problem	Authors	Contents	Year
Data-mining 、 Max-Sat problem	Karima, Lotfi and Mouloud	Using Bees to Solve a Data-Mining Problem Expressed as a Max-Sat One	2005
Benchmark mathematical problems	Hadda, Afshar and Marino	Honey-Bees Mating Optimization (HBMO) Algorithm A New Heuristic Approach for Water Resources Optimization	2006
Reservoir optimization problem	Haddad and Afshar	Honey-bee mating optimization (HBMO) algorithm for optimal reservoir operation	2007
Partitioning and scheduling	Koudil, Benatchba, Tarabet, and Sahraoui	Using artificial bees to solve partitioning and scheduling problems in codesign.	2007
Clustering	Fathian, Amiri, and Maroosi	Application of honey-bee mating optimization algorithm on clustering	2007
Vehicle Routing problems	Marinakis, Marinaki and Dounias	Honey Bees Mating Optimization Algorithm for the Vehicle Routing Problem	2008

此外，Karaboga et al.(2005)與 Teodorovic et al.(2005)個別提出了另外兩種模擬蜜蜂行為的演算法。Karaboga 效仿蜜蜂階級社會的資訊回饋機制於食物搜尋之行為模式，提出人造蜂群演算法(Artificial bee colony; ABC)。Teodorovic 所提出的蜂群最佳化(Bee colony optimization; BCO)，同樣效仿蜜蜂搜尋食物的過程，透過每一階段蜂群回巢後交換資訊來決定下一階段搜尋的方向。

由文獻指出利用蜂群優化演算法來解決資源分配、排程、分群等問題，與過去些傳統的演化計算比較，如：GA、TS、ACO、SA，得到較佳的比較結果。然而目前尚無對連續型實數問題之探討研究。

第三章 實數型蜂群最佳化

3.1 MBO 之步驟

生物學家 Rinderer and Collins 於 1986 年提出蜜蜂的行為模式透過遺傳、環境與社會階層三者交互影響，每隻蜜蜂會依據本身的基因遺傳、環境影響以及社會地位產生行動，每次行之動結果會成為環境的一部份，並影響隨後的蜜蜂。

一般來說蜜蜂的社會結構包含蜂后、雄蜂、工蜂與卵。蜂后最主要的功能便是產卵繁衍後代。雄蜂則是提供不同的遺傳因子給蜂后，其遺傳因子除非發生突變，否則其特徵源自於母代，且為單倍體(haploid)，因此雄蜂可被視為專為蜂后產卵多樣化的而生。工蜂被賦予的職責是照顧子代，也就是卵。卵又可分為受精與未受精，受過精的卵將來可能長成蜂后或工蜂，未受精的卵則會變成雄蜂。

交配旅程(mating-flight)，是由蜂后為了與雄蜂達到交配目的而行成的一種漫步飛行，當蜂后離開巢穴展開旅程時，雄蜂會跟隨其後，兩者在空中完成交配。一般典型的交配旅程，蜂后会與七到二十隻不等的雄蜂交配。蜂后透過這樣的活動蒐集蜂群中雄蜂的遺傳因子(精子)並儲存於精巢內，之後，每當蜂后要產卵時，便會隨機混合精巢中的精子產生受精卵。

蜂群演算法主要的步驟流程大至可分為：(1)蜂后與雄蜂的交配旅程；(2)蜂后產生子代；(3)工蜂改進子代之適應值；(4)工蜂的適應值；(5)最佳子代替換蜂后。其中又以蜂后的交配旅程為此演算法最主要的核心，在此階段中，蜂后執行交配旅程，並隨機與雄蜂相遇，當交配事件發生，雄蜂的基因便會記錄於蜂后的精巢內，並於稍後與蜂后的基因交叉產生新的子代。蜂群最佳化發展雖只短短七、八年，應用於許多領域，然而其求解模式結構並沒改變。圖 3.1 為其流程架構。

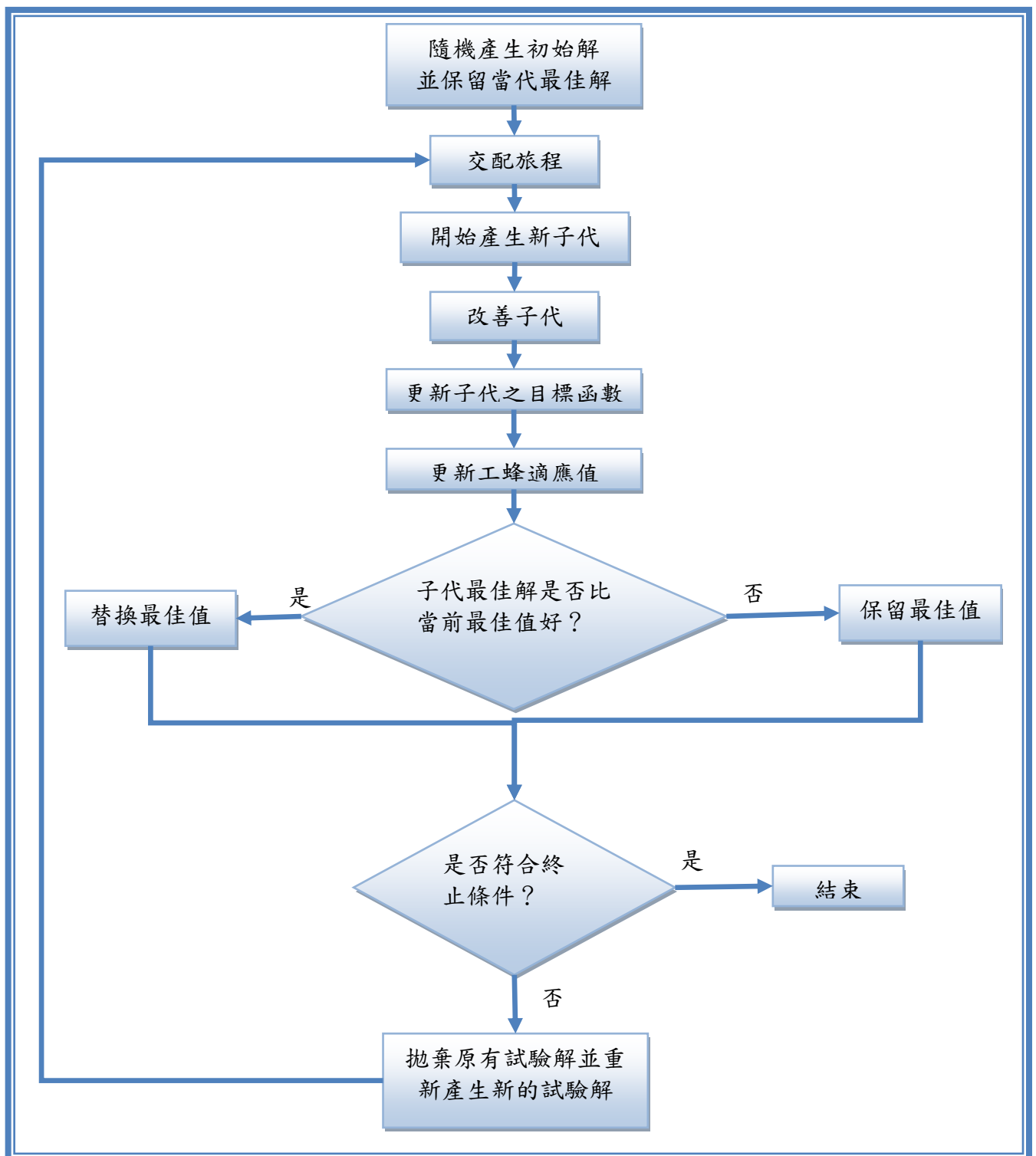


圖 3.1 MBO 之架構圖

操作步驟，步驟內容詳述如下：

Step 1. 蜂后與雄蜂的交配旅程(The mating-flight of the queen bees with drones)

交配旅程是蜂后在蜂群最佳化求解最佳化問題之狀態空間(state-space)轉移步驟，蜂后在可允諾之飛行距離內，透過單純探索策略踏遍達四周新的狀態空間。雄蜂則產生於蜂后所經過之位置，在退火函數的變異機率下與蜂后交配。

每隻蜂后皆有基因型態(genotype)、精巢(spermatheca)、能量(energy)與速度(speed)四個參數；蜂后的基因為該蜂群中最強之代表，透過隨機漫步飛行與雄蜂相遇，交配後留下雄蜂之基因於精巢內。其交配的機率如下：

Queen bee 交配機率(即為 SA)

$$P(Q, D) = e^{-\frac{\Delta(f)}{S(t)}} \quad (1)$$

表示雄蜂 D 的遺傳因子與蜂后 Q 的交配機率， $\Delta(f)$ 表示雄蜂與蜂后適應值(即目標函數)的差異， $S(t)$ 表 t 時間點蜂后的飛行速度；因此，在適應直差異小或蜂后速度較快時，交配成功的機率較高。當開始交配旅程時，蜂后的能量與速度將會如下遞減：

$$\text{Energy: } E(t+1) = E(t) - \text{step} \quad (2)$$

$$\text{Speed: } S(t+1) = a * S(t) \quad \text{where } 0 < a < 1 \quad (3)$$

$$\text{step} = \frac{0.5 \times \text{Initial energy}}{\text{Size of the spermatheca}}$$

其中 step 為每回能量的遞減量。當蜂后的能量降為零或精巢已滿，交配旅程便會結束，蜂后回到蜂巢進行產卵的動作。

Step 2. 蜂后產生子代(The creation of new broods by the queen bees)

經過交配旅程後回到蜂巢，蜂后隨機選取精巢中的雄蜂遺傳因子，進行單倍體交叉生下卵(即子代)，精巢內雄蜂遺傳因子將被隨機選取，與蜂后基因結合生下新的卵。

單倍體交叉(Haploid-Crossover)

假設雄蜂精子皆為單倍體(haploid)，其基因遮蔽記號(genotype marker)如下：

基因遮蔽記號(Genotype-marker)

m	u	m	m	u	m	m	u	u	u
---	---	---	---	---	---	---	---	---	---

假設蜂后與雄蜂的基因個別如下。

蜂后的基因(Queen's genotype)

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
-------	-------	-------	-------	-------	-------	-------	-------	-------	----------

雄蜂的基因(Drone's genotype)

y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8	y_9	y_{10}
-------	-------	-------	-------	-------	-------	-------	-------	-------	----------

透過基因遮蔽記號構成雄蜂的遺傳因子(精子)，*代表不存在的基因，在此演算法中代表雄蜂一半遺傳因子。

雄蜂的精子(Drone sperm)

*	y_2	*	*	y_5	*	*	y_8	y_9	y_{10}
---	-------	---	---	-------	---	---	-------	-------	----------

之後再結合蜂后的基因，取代遮蔽位置，產生子代。

子代(Child)

x_1	y_2	x_3	x_4	y_5	x_6	x_7	y_8	y_9	y_{10}
-------	-------	-------	-------	-------	-------	-------	-------	-------	----------

Step 3. 工蜂改進子代之適應值(The improvement of the broods' fitness by workers)

蜂后產完卵後，形成一群新子代，由工蜂進行照料，在演算法中視作改善子代的適應值(即區域搜尋)；常用的方法有 GSAT(greedy local search procedure), Random Walk, Probabilistic Greedy(一定機率接受最佳 or 比最佳

差的解), One-point Crossover, and WalkSAT, HSAT(hill-climbing procedures), SLS(simple local search), GLS(Guided local search), 其中以 HSAT 結果最好 (Mouloud Koudil, 2007)。

本研究採用的改善方法為 One-point Crossover、Random Walk 和 Hill-climbing procedures，參數上個別操作方法如下說明：

1. One-point Crossover

假設蜂后與雄蜂的基因個別如下。

蜂后的基因(Queen's genotype)

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
-------	-------	-------	-------	-------	-------	-------	-------	-------	----------

子代的基因(Brood's genotype)

b_1	b_2	b_3	b_4	b_5	b_6	b_7	b_8	b_9	b_{10}
-------	-------	-------	-------	-------	-------	-------	-------	-------	----------

隨機決定一個交配點。

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
-------	-------	-------	-------	-------	-------	-------	-------	-------	----------

b_1	b_2	b_3	b_4	b_5	b_6	b_7	b_8	b_9	b_{10}
-------	-------	-------	-------	-------	-------	-------	-------	-------	----------

交換兩者後段之基因，並隨機選取其中一個作為新的子代。

新子代的基因(New brood's genotype)

x_1	x_2	x_3	x_4	x_5	b_6	b_7	b_8	b_9	b_{10}
-------	-------	-------	-------	-------	-------	-------	-------	-------	----------

b_1	b_2	b_3	b_4	b_5	x_6	x_7	x_8	x_9	x_{10}
-------	-------	-------	-------	-------	-------	-------	-------	-------	----------

2. Random Walk

隨機漫步(Random Walk)是一種不規則的變動形式。在種變動過程中，

每一部的變化都純粹是隨機性的。計算的公式如下表示：

$$\begin{aligned} b'_{t,i} &= \alpha + b_{t,i} + \varepsilon_i \\ \varepsilon_i &\in iid N(0, \sigma^2) \end{aligned} \quad (4)$$

$b_{t,i}$ 代表第 t 的疊代第 i 個子代， $b'_{t,i}$ 為第 t 的疊代第 i 個工蜂改善後的子代， α 是一個常數， ε_i 為誤差項。

3. Hill-climbing procedures

登山法(Hill-climbing)只是一個持續往能夠讓直增加的方向移動的簡單回圈。此處我們的操作如下：

(1) 產生新子代 $b'_{t,i} = b_{t,i} + \varepsilon_i$, $\varepsilon_i \in iid N(0, \sigma^2)$

(2) 如果 $f(b'_{t,i}) \leq f(b_{t,i})$ ，則回到上一步

Step 4. 工蜂的適應值(The adaptation of the workers' fitness)

工蜂個別的適應值會隨其改善子代的程度排序。在實數問題的討論下，本實驗以當次疊代工蜂改善子代的幅度作為下一次該方法改善機會的依據。

評估值函數設計如下：

$$\text{Worker's fitness : } Fitness(t + 1) = \frac{f(b_{t,i}) - f(b'_{t,i})}{\sum_i^w f(b_{t,i}) - f(b'_{t,i})} \quad (5)$$

$b_{t,i}$ 代表第 t 的疊代第 i 個子代， $b'_{t,i}$ 為第 t 的疊代第 i 個工蜂改善後的子代， w 為工蜂(即改善方法)總數量。

Step 5. 最佳子代替換蜂后(The replacement of the least fittest queen(s) with the fittest brood(s))

最後適應值最差的蜂后將被最佳的子代給取代，直至沒有子代比蜂后的值更好為止，完成後便會將所有的卵銷毀，重新回到步驟一，展開另一次交配旅程。

Original MBO algorithm

Initialize workers randomly generate the queens apply local search to get a better queen for a pre-defined maximum number of mating-flights

```
for each queen in the queen list
  initialize energy, speed and position
  while the queen's spermatheca is not full and energy > 0
    the queen moves between states and probabilistically chooses drones
    if a drone is selected, then
      add its sperm to the queen's spermatheca
    end if
    update the queen's internal energy and speed
  end while
end for each
generate broods by crossover and mutation
use workers to improve broods
update workers' fitness
while the best brood is better than the worst queen
  replace least-fittest queen with the best brood
  remove the best brood from the brood list
end while
kill all broods
```

圖 3.2 蜂群最佳化的虛擬碼

3.2 MBO 之結構

Abbass and Teo (2001)提出蜂群演算法，藉由模仿蜜蜂繁衍的行為模式所發展的一套演算法則。如同現今一般混合型啟發式演算法，蜂群最佳化演算法結合模擬退火的退火機率與基因演算法的交配機制，並在區域搜尋上做傳統數學方法競爭。求解最佳化問題時，演化式演算法與傳統數學方法的應用；前者目的在於維持求解複雜性，避免過早落入區域最佳解並縮短求解時間，後者則是希望能在小範圍裡增加求解精準度。

綜觀整個蜂群最佳化的過程，大致可分為三個部分：模擬退火(Simulated Annealing; SA)、基因演算法與區域搜尋。

模擬退火—交配飛行利用退火函數控制蜂后與雄蜂交配之機率。此函數會隨飛行時間造成能量與速度的遞減，在適應直差異小或蜂后速度較快時，交配成功的機率較高。藉此達到全域搜尋之效果。

基因演算法—此階段為演化過程中保持複雜度的部分，由基因演算法的交配與突變機制，產生不同子代，保持演化過程的複雜度。

區域搜尋法則—不論是 WalkSAT、random walk，還是後來 Mouloud Koudil 所延伸的 GSAT 與 HSAT，皆為傳統且過程較單純的數學方法，其目的不外乎希望能在小範圍內做最快速、有效的搜尋。

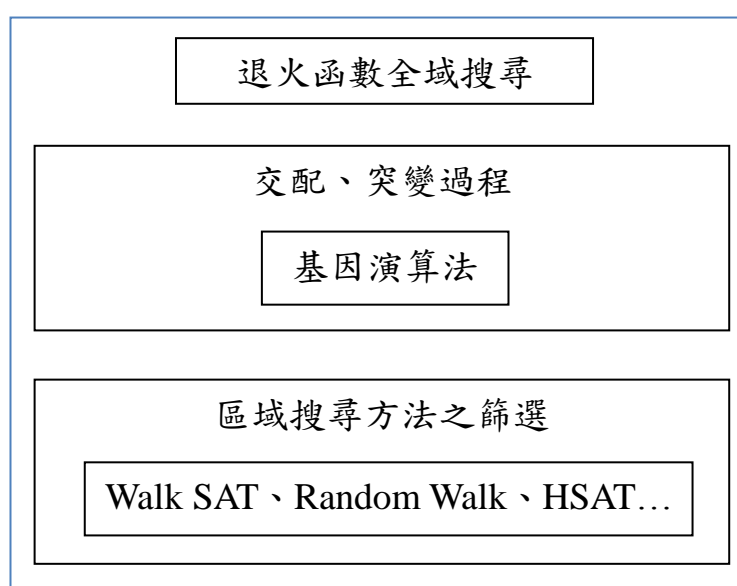


圖 3.3 MBO 結構概念

- **模擬退火(Simulated Annealing; SA)的特性**

在傳統模擬退火之方法 Abboud et al.(1998)提出包含了以下四個基本要素：

1. 系統狀態(Configuration)：即在某一溫度下，系統產生的初始解，並當作目前的現行解。
2. 搜尋法則(Move Set)：在退火過程中，由目前的系統狀態經隨機擾動以致產生變化至另一種狀態。SA 常用的搜尋法則有梯度搜尋法(Gradient Type)和疊代改善法(Iterative Improvement)。
3. 成本函數(Cost Function)：衡量某一系統狀態下之能量函數(即目標函數

值)。

4. 退火程序(Annealing Schedule)：退火程序中包含的參數有初始溫度、降溫時機、冷卻率和終止溫度。

依 Kirkpatrick et al.(1983)的發表指出模擬退火演算法求解機制：找一起始解，並依據此解，隨機產生一鄰近解，若此解比現行解好即接受，並移動過去；若該鄰近解較差，則給定一個機率，依照此機率大小判斷是否接受，若接受則移動過去，若不接受則回到目前的解繼續搜尋。當接受的解達到預定的數目時，就需降溫，並以上一個溫度的最佳解為起始解繼續搜尋，直到符合終止條件。當在高溫時，隨機尋找的鄰近解容易被接受，此時近似隨機的搜尋。

• 基因演算法的特性

基因演算法同時以多點方式搜尋最佳解，而非點對點的搜尋，對於多峰谷之函數(Multimodal)而言，基因演算法較傳統演算法更可以較快找出整體最佳解(Global Optimum)，同時也能避免陷入區域最佳解(Local Optimum)。此特性是基因演算法的最大優點。不過由於計算所需的時間過長、不易掌控，也是另一個缺陷。Goldberg(1989)說明了基因演算法不同於傳統的最佳化方法主要可歸納為下列幾項特性：

1. 基因演算法的運作是關於編碼後的參數，而不是參數的本身。因此可使適用的範圍更廣。
2. 基因演算法在搜尋的過程中是對整個母體做搜尋，也就是對多個點同時做搜尋，而不是單點而已。因此能避免落入區域最佳解(Local Optimum)。
3. 基因演算法使用的是目標函數(Objective Function)的資訊，不需要微分或其他輔助的資訊。因此能使用在各種型態的適應函數，例如多目標或非線性函數等。
4. 基因演算法利用機率性(Probabilistic)方式來求解，而不是使用明確性的(Deterministic)規則。這樣一來可使它更具彈性。

- **區域搜尋法則**

蜂群最佳化演算法除了退火與交配之外，另一項核心精神便是工蜂的競爭，也就是傳統數學方法透過衡量子代改善幅度大小，來決定彼此的優劣。從文獻中得知在這裡 Hill-climbing procedures 為最快速也最有效果的工蜂改善搜尋法則。

Hill-climbing 是一種疊代增進的演算法，它利用單一解在解空間作搜尋，並在每一次疊代中，在目前解的鄰近解空間選擇出一個鄰近解。當鄰近解的目標函數值比目前解的目標函數值來的佳時，就以鄰近解取代目前解；否則，就重新在目前解的鄰近解空間選擇一個鄰近解。如此一步一步沿著求解圖形往上攀升直至目標值最大為止。Hill-climbing 是挑選鄰近點中最好的點，但這樣會有局部最大值的問題。本研究所測試之函數為目標函數值最小化問題，故在操作上相對於所期望的函數值應為往下降，而不是往上攀升。

本研究探討 MBO 結合上述方法的機制，以表 3.1 來整理其與一般 SA 與 GA 的異同。MBO 與 SA 皆有一個機率公式，可使演算法的搜尋過程中有機會接受較差的函數值，主要目的在於藉此來提升全域搜尋的廣度。兩者差異點在於，MBO 另外設定了飛行能量與精巢大小，分別限制交配飛行的時間與搜尋個數，此動作有助於控制交配飛行的時間。

GA 的部分，GA 已經是個相當成熟的演算法，有許多延伸應用，這裡 MBO 主要是利用其交配操作的部分產生下一代的卵，提供之後的工蜂作為改善的起始依據。透過交配操作產生新一子代，GA 交配的方法有很多，本研究沿用單倍體交配的方式。

表 3. 1 MBO 與 SA、GA 的異同

	MBO	SA	GA
搜尋法則	以交配飛行做全域搜尋，速度越快、差異值越小接受比例越高。另藉由不同區域搜尋改善方式彼此競爭，不同方式可達到不同的操作效果。	由目前的系統狀態經隨機擾動以致產生變化至另一種狀態。	搜尋的過程中是對整個母體做搜尋，也就是對多個點同時做搜尋，而不是單點而已。
天擇	經過改善後的子代適應值若比蜂后好，則替換之。除此之外，工蜂的適應值也會隨改善子代幅度作改變，以利下一疊代的改善幅度調整。	退火程序中包含的參數有初始溫度、降溫時機、冷卻率和終止溫度。因系統經隨機擾動以致產生變化至另一種狀態，溫度高時，有較高的比例接受較差的值，此時近似隨機搜尋。	適合度函數值越高，其被選擇的機率就會越高；反之，則被選擇的機率越低。選擇機制模擬自然界適者生存的機制，根據染色體的適應值來決定參與交配的染色體族群。
交配	將兩個染色體的遺傳作隨機交換，以產生新的子代，其效果猶如在解答空間中作大幅度的跳躍搜尋。	—	在交配池中的母代染色體會隨機的配對，互換基因，以繁衍下一代。藉由這種方式，染色體間可互換資訊，以使得子代染色體能有更佳的適合度函數值。
突變	—	—	藉由突變增加個體間的變異性，讓生物有機會發展出具新特性的個體，使其更具多樣性。
搜尋個數	以精巢大小限制每次交配飛行的交配次數，精巢越大蜂后產的卵越多，表示工蜂改善子代能多點方向同時搜尋。	固定搜尋個數，當接受的解達到預定的數目時，就需降溫，並以上一個溫度的最佳解為起始解繼續搜尋。	—
搜尋時間	設定飛行初始能量，其隨時間遞減，能量歸零時，交配飛行結束，藉此控制全域搜尋的時間。	—	—

第四章 實驗結果與效能分析

4.1 測試函數

本研究擬比較 MBO 針對連續型變數最佳化問題與其他的演算法探討。為了公平及有效地衡量上述方法在求解問題上的效能，故以 CEC 2005 special session (IEEE Congress on Evolutionary Computation of the Special Session on Real-Parameter Optimization) 在 2005 年所提供的標準測試函數來做為測試對象，基於這個測試平台的評判標準下，目前已有釋出在此評判標準下所得的相關結果。故可將本研究的比較對象與在這個 session 已接受的結果進行比較與分析，進而觀察 MBO 於搜尋效能上所提升的成度。

本研究以最小化問題做為主要探討對象，測試的實驗將測試函數歸納為兩大類，分別為單一區域解函數與多重區域解函數，透過這兩大類函數的實驗以測試本研究所提方法於各種函數類型之穩定性，詳細的函數定義與特性見表 4.1。

表 4.1 測試函數與函數特性

	測試函數	特性
單一 區域解 函數 (unimodal)	$f_1 = \sum_{i=1}^m x_i^2$ (Sphere Model)	<ul style="list-style-type: none"> • Unimodal • Separable • Scalable
	$f_2 = \sum_{i=1}^m \left(\sum_{j=1}^i x_j \right)^2$ (Schwefel's Problem)	<ul style="list-style-type: none"> • Unimodal • Non-separable • Scalable
	$f_3 = \sum_{i=1}^m \left(10^6 \right)^{\frac{i-1}{m-1}} x_i^2$ (Rotated High conditioned Elliptic Function)	<ul style="list-style-type: none"> • Unimodal • Rotated • Non-separable • Scalable

	測試函數	特性
	$f_4 = \sum_{i=1}^{m-1} (100(x_i^2 - x_{i+1})^2 + (x_i - 1)^2)$ (Rosenbrock's Function)	<ul style="list-style-type: none"> • Multi-modal • Non-separable • Scalable • Having a very narrow valley form local optimum to global optimum
	$f_5 = \sum_{i=1}^m [x_i^2 - 10 \cos(2\pi x_i) + 10]$ (Generalized Rastrigin's Function)	<ul style="list-style-type: none"> • Multi-modal • Separable • Scalable • Local optima's number is huge
多重 區域解 函數 (multi-modal)	$f_6 = 20 + e - 20e^{-0.2\sqrt{\frac{\sum_{i=1}^m x_i^2}{m}}} - e^{\sum_{i=1}^m \frac{\cos(2\pi x_i)}{m}}$ (Ackley's Function)	<ul style="list-style-type: none"> • Multi-modal • Rotated • Non-separable • Scalable • Global optimum on the bound • If the initialization procedure initializes the population at the bounds, this problem will be solved easily
	$f_7 = \frac{1}{4000} \sum_{i=1}^m x_i^2 - \prod_{i=1}^m \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$ (Generalized Griewank Function)	<ul style="list-style-type: none"> • Multi-modal • Rotated • Non-separable • Scalable • No bounds for variables

此 7 組函數皆為實數參數最小化問題，每一個測試函數皆有其不同特性。在解空間的搜尋範圍除了 Function 7 沒有限制外，其餘皆有固定的搜尋範圍，範圍設定明確定義並整理於表 4.2。實驗數據取得為針對每一個函數執行 1000 次函數計算(Function Evaluations)，並重複執行 25 次實驗。在演化次數(FES)達到 1000 時，分別記錄錯誤值(error value) $f(x) - f(x^*)$ 中每一個 function 在 25 次實驗所得的平均值與標準差。當終止條件達成時，依照問題的困難度分別定義一個已求解的臨界值(Accuracy)，若最佳化的結果其錯誤率小於所定義的臨界值 $f(x) - f(x^*) \leq Accuracy$ ，則我們是此問題為可被解決的問題。表 4.2 為 7 個 test functions 的求解範圍與臨界值定義。

表 4.2 最佳解、搜尋範圍定義

Function number	$f(x^*)$	Search Range	Accuracy
F1	0	$[-100,100]^m$ $m = 10$	1.00e-6
F2	0	$[-100,100]^m$ $m = 10$	1.00e-6
F3	0	$[-100,100]^m$ $m = 10$	1.00e-6
F4	0	$[-100,100]^m$ $m = 10$	1.00e-2
F5	0	$[-5,5]^m$ $m = 10$	1.00e-2
F6	0	$[-32,32]^m$ $m = 10$	1.00e-2
F7	0	$[-\infty, \infty]^m$ $m = 10$	1.00e-2

4.2 MBO 參數實驗

4.2.1 參數設計

1. 參數：精巢大小

交配飛行為 MBO 最主要的操作機制之一，精子儲存巢大小直接影響每次交配飛行後，蜂后的產卵量與每個世代單一隻工蜂改善子代數的個數，故對 MBO 不管在全域搜尋或區域搜尋皆具有重要影響力。蜂后產卵數量越多，代表之後工蜂可改善的子代越多，對演算法的搜尋深度有影響。

2. 參數：蜂后飛行速度遞減比

蜂后交配飛行過程中，決定交配成功機率，在目標函數值差異相同的情況下，飛行速度越快，則交配成功的機率越大。整個交配飛行過程的飛行速度成等比例遞減，遞減比例大小影響飛行速度下降的快慢，下降越快表示越早拒絕接受函數值較差的雄蜂遺傳因子。對演算法全域搜尋有直接影響。

3. 參數：蜂后飛行能量初始值

飛行能量與飛行速度相同，兩者接會隨著時間遞減。飛行速度控制交配與否之機率，飛行能量則決定飛行過程的時間長短。此兩個參數則是對搜尋廣度的影響。

在求解問題、測試範圍相同的條件下，測試 MBO 各參數之間彼此對求解速度與效能之影響。欲實驗比較的參數有子代數與精巢大小、蜂后飛行速度遞減比與飛行能量初始值，各比較參數之詳細設定如表 4.3。

表 4.3 MBO 實驗比較參數表

比較參數	參數值
精巢大小	10、20、30、40
蜂后飛行速度遞減比	0.5、0.7、0.9
蜂后飛行能量初始值	1、5、10

本研究將根據程式執行的結果，依據下列幾項指標做為比較點：

- (1) 達收斂時之平均計算時間：在 25 次的測試中，將收斂且達臨界值的次數計算其演算時間，並取平均值。
- (2) 達收斂時之平均計算時間之標準差：在(1)中計算標準差。
- (3) 達收斂時之最佳解平均值：將每次測試後的最好的值取平均值。
- (4) 達收斂時之最佳解標準差：將(3)計算標準差。
- (5) 收斂百分比(Convergence Percentage)：此指標的計算方式為：能夠達到收斂條件的次數/設定的測試次數)*100%。

4.2.2 實驗結果

1. 精巢大小

本實驗設定比較的精巢大小為 10、20、30、40。其對應的飛行次數也有所改變，分別為 600(對應子代數 10)、300(對應子代數 20)、200(對應子代數 30)與 150(對應子代數 40)。如此，可保持整個求解過程擁有固定 6000 次的整體搜尋效果。本實驗固定蜂后的飛行速度遞減比率為 0.9，飛行能量初始值為 1，與改善後子代數為 200，討論子代數對 MBO 演算法求解效能之影響。表 4.8 整理四組不同參數環境底下 MBO 的實驗結果。

(1) 單一區域最佳解(Function 1~3)

單一區域最佳解的函數中，MBO 的求解時間與函數值皆有不錯的表現，唯 Function 3 的收斂比率較不理想。Function 1 子代數越高計算時間雖然越長，然而差異性並不大；相對目標誤差值的標準差(Standard deviation, STD)較小，求解品質較為穩定。整體而言，MBO 子代數在 30 的求解時間標準差最小，在 20 的時候收斂比例最佳。

(2) 多重區域最佳解(Function 4~7)

多重區域最佳解的部分，雖然 MBO 在本次實驗的 Function 4 中皆無法達到臨界值，但面對其他三個測試函數收斂比例為百分之百，且誤差值皆可達到理想值。計算時間上，在子代數為 10 的時候表現最

佳，平均時間最短、標準差最小，求解品質最穩定。

由於改變子代數除了直接影響每個世代交配產生的子代數量，更間接影響到區域搜尋時單一工蜂改善子代的次數，故會影響區域搜尋的求解敏感度。整體觀察下來，精巢大小的確影響 MBO 搜尋的穩定性，當精巢大小設定在 20 時，求解品質較佳，收斂百分比較高。瞭解 MBO 最佳化的求解效能後，我們接著從搜尋時間來探討蜂后交配飛行時重要的參數。

2. 蜂后飛行速度遞減比與初始能量

交配飛行為 MBO 最主要的操作機制之一，與傳統模擬退火相同的地方在於飛行速度快時，其可允許接受較差函數值之機率較高，使得演算法擁有全域搜尋搜尋能力，維持其搜尋上的廣度；隨著速度遞減，接受機率降低，此時則轉為深度，即區域搜尋的能力。不同的地方在於 MBO 另外擁有飛行能量與精子儲存巢大小的限制，此方法能夠有效控制蜂后交配飛行的時間與搜尋的數量。

當初始能量越高時，蜂后有較多的時間執行交配飛行，而能量上限控制停止交配飛行動作，有助於退火函數時間上的掌控。飛行速度則是控制接受較差解的機率，速度越高接受度越高；反之亦然，速度越低接受度越低。前段飛行速度高的過程，有機會接觸更多元的求解空間，此為 MBO 在全域搜尋廣度的表現；MBO 是以等比例遞減的模式來降低飛行速度，即控制接受可行解的機率。兩個參數為交配飛行模仿退火函數應用在全域搜尋的主要機制。

本實驗固定精巢大小為 20，討論在速度遞減比為 0.5、0.7、0.9，與初始能量為 1、5、10，對函數搜尋能力個別表現與彼此間的交互作用，得到最佳參數組合。由於前一項精巢大小實驗結果，瞭解 MBO 的求解效能，本實驗觀察重點在求解時間的表現。各組參數組合於各函數收斂的計算時間整理於表 4.5。

表 4.5 是先將各函數底下之各組參數計算時間表現排名(Rank)，再累計各組參數於所有函數之排名總和(Total Rank)，總和越低者表示其時間之效

表 4.4 子代數對 MBO 的影響

spermatheca size		10			20			30			40		
F#		value	CPUTime	Con.	value	CPUTime	Con.	value	CPUTime	Con.	value	CPUTime	Con.
F1	Avg.	4.10e-08	4.7503	88%	0.00e+00	2.9327	92%	0.00e+00	4.4722	100%	0.00e+00	6.0596	100%
	STD	7.10e-08	3.8335		0.00e+00	0.0378		0.00e+00	0.051		0.00e+00	0.1038	
F2	Avg.	0.00e+00	3.2139	100%	0.00e+00	4.691	100%	0.00e+00	4.6093	100%	0.00e+00	5.9533	100%
	STD	0.00e+00	1.195		0.00e+00	2.5606		0.00e+00	0.0733		0.00e+00	0.2756	
F3	Avg.	0.00e+00	7.172	36%	0.00e+00	3.01	28%	0.00e+00	4.1015	20%	0.00e+00	5.75	20%
	STD	0.00e+00	6.5664		0.00e+00	0.1129		0.00e+00	0.0771		0.00e+00	0.3097	
F4	Avg.	8.82e+00	8.1015	0%	8.68e+00	11.6356	0%	8.65e+00	15.6043	0%	8.66e+00	20.099	0%
	STD	2.50e-01	0.0544		2.78e-02	0.0235		5.15e-03	0.1607		3.34e-02	0.1189	
F5	Avg.	0.00e+00	3.8737	100%	0.00e+00	5.3731	100%	0.00e+00	8.1162	100%	0.00e+00	10.3731	100%
	STD	0.00e+00	2.4217		0.00e+00	2.2015		0.00e+00	2.5936		0.00e+00	3.0443	
F6	Avg.	0.00e+00	2.9275	100%	0.00e+00	3.7325	100%	0.00e+00	5.6856	100%	0.00e+00	7.6987	100%
	STD	0.00e+00	1.8631		0.00e+00	1.4699		0.00e+00	1.4537		0.00e+00	1.6126	
F7	Avg.	0.00e+00	1.8119	100%	0.00e+00	2.96308	100%	0.00e+00	4.6988	100%	0.00e+00	7.1512	100%
	STD	0.00e+00	0.8575		0.00e+00	0.1109		0.00e+00	0.2011		0.00e+00	0.4014	

註：con. = convergence

能表現越好。本實驗結果可知，飛行速度遞減比與初始能量在 0.5 與 10 時，整體函數之表現最佳。

表 4.5 各函數速度遞減比例與初始能量收斂時間

	(速度遞減比例_初始能量)								
	(0.5_1)	(0.5_5)	(0.5_10)	(0.7_1)	(0.7_5)	(0.7_10)	(0.9_1)	(0.9_5)	(0.9_10)
F 1	7.332	6.951	7.125	7.328	7.359	7.362	7.641	7.344	7.730
Rank	5	1	2	4	7	8	9	6	3
F 2	29.797	10.875	28.906	28.89	29.016	38.094	41.031	40.266	36.703
Rank	5	1	3	2	4	7	9	8	6
F3	7.297	7.187	6.297	5.266	7.344	7.156	6.781	7.281	7.375
Rank	7	5	2	1	8	4	3	6	9
F4	26.98	41.43	41	34.46	41.90	43.10	26.04	41.48	40.51
Rank	2	6	5	3	8	9	1	7	4
F5	33.3	25.97	28.89	31.7	34.18	32.17	45.9	34.56	37.13
Rank	5	1	2	3	6	4	9	7	8
F6	37.62	34.74	33.54	34.55	31.47	71.37	41.89	36.57	34.27
Rank	7	5	2	4	1	9	8	6	3
F 7	25.73	16.39	19.89	16.07	7.69	37.23	15.73	11.35	42.46
Rank	7	5	6	4	1	8	3	2	9
Total Rank	33	24	18	21	35	50	42	42	42

4.3 比較對象與之參數設定

本文比較 $(\mu + \lambda)$ ES 與 MBO 在搜尋效能上的差異。此部分的觀察重點，著重在比較在實數函數的環境下原本的 MBO，其效能相較於 $(\mu + \lambda)$ ES 所提升的成度。對於 $(\mu + \lambda)$ ES 在上述函數的效能表現，在 CEC05 Special Session 所提供的相關文獻中，Lino Costa 所提的 PLES(A Parameter-less Evolution Strategy for Global Optimization)裡已有完整的實驗數據與結果，此外，由於 PLES 也是基於 $(\mu + \lambda)$ ES 的架構下，對於其參數調整方面作改良，故也將之列入本實驗的比較對象。

本實驗參數的設定部分與 PLES 所設定的條件有些不同，PLES 的母體數(μ)為 10，子代數(λ)為 100，參與重組的候選個數(ρ)為 10，突變強度的全域學習率與區域學習率分別為 $\tau' \propto 1/\sqrt{2n}$ 與 $\tau \propto 1/\sqrt{2\sqrt{n}}$ ，整理於表 4.6。

MBO 的參數設定，蜂后數為 1，子代數為 20，飛行速度遞減比為 0.9，飛行能量初始值為 1；另又針對子代數做 10、20、30、40 的參數比較，相對應的飛行次數(終止疊代數及演化次數)為 1000 次。改善子代數皆為 200，為區域搜尋方法改善次數之總和。詳見表 4.7。

表 4.6 PLES 實驗參數

參數	參數值
母體數 (μ)	10
子代數 (λ)	100
候選個數 (ρ)	10
全域學習率 (τ')	$1/\sqrt{2n}$
區域學習率 (τ)	$1/\sqrt{2\sqrt{n}}$

表 4.7 MBO 實驗參數

參數	參數值
蜂后數	1
工蜂數	傳統數學方法(3)
子代數	20
交配飛行次數	1000
蜂后的精巢大小	20
改善後的子代數	200
蜂后飛行速度遞減比	0.5
蜂后飛行能量初始值	10

4.3.1 與 $(\mu+\lambda)$ ES 比較

MBO 基於上述的參數設定完整實驗數據整理於表 4.8。表 4.9 為結果比較，此部分重點觀察 7 個測試函數分別在 25 次獨立試驗，終止條件為 1000 次疊代的平均效能表現，觀察 MBO 效能提升的成度。由實驗結果可知，在 3 個單一區域最佳解的測試函數下，MBO 與的表現皆明顯優於傳統 ES 與 PLES。在多重區域最佳解的結果部分，Function 4 雖無法達到最佳值，但求解能力還是優於比較對象。此外，其餘問題也是同樣現象。整體來說，在應用 MBO 的交配飛行概念底下，這組測試函數所衡量的結果，對於其效能確實能達到有效的提升。

由實驗數據可知，MBO 應用退火函數在全域搜尋中，的確能避免過早陷入區域解而達全域最佳，於整體效能提升有很好表現。另外，由於 MBO 在子代改善有改善代數的限制，每次交配飛行完畢後，工蜂必定會做 200 次區域搜尋，即便改善子代已達臨界值標準卻仍持續改善操作，故最佳解會有 0 值的產生。

表 4.8 MBO 實驗結果

F#	MBO	
	Mean	Std.
F1	0.00e+00	0.00e+00
F2	0.00e+00	0.00e+00
F3	0.00e+00	0.00e+00
F4	8.63e+00	1.01e-01
F5	0.00e+00	0.00e+00
F6	0.00e+00	0.00e+00
F7	0.00e+00	0.00e+00

表 4.9 ES、PLES、MBO 平均效能表現

F#	$(\mu + \lambda)$ ES 排名	PLES 排名	MBO 排名
F1	9.86e-09 (3)	8.4e-09 (2)	0.00e+00 (1)
F2	2.9e-06 (3)	9.65e-09 (2)	0.00e+00 (1)
F3	3.52e-05 (3)	1.18e-05 (2)	0.00e+00 (1)
F4	7.49e+01 (3)	3.05e+01 (2)	8.63e+00 (1)
F5	4.49e+01 (3)	1.67e+01 (2)	0.00e+00 (1)
F6	2.04e+01 (2)	2.04e+01 (2)	0.00e+00 (1)
F7	1.18e+00 (2)	4.09e+00 (3)	0.00e+00 (1)

4.3.2 小結

為了有效及公平地衡量 MBO 演算法在搜尋以及求解問題上的能力，使用 CEC 2005 Special Session 所提的測試函數做為我們衡量演算法效能的測試依據，實驗結果說明，MBO 的求解能力確明顯優於傳統 ES；由於 MBO 的特性保持整個求解過程擁有固定 6000 次的全域搜尋，每次全域搜尋底下又各做 200 次的區域搜尋(工蜂改善子代)，故整個求解過程至多會有 12000 的求解組合。每個世代的交配飛行是 MBO 在全域搜尋的特點，而單純的區域搜尋法則能快速反應該區域的最佳解位置，故求解效能較傳統 $(\mu + \lambda)$ ES 為佳。

由本次實驗研究結果可知，蜂群最佳化在連續型變數的函數中表現亦相當突出。而其中利用到的求解機制令人好奇，下一節我們將從另一個角度—函數評估次數與誤差值收斂情形，來分析 MBO 最佳化過程的操作邏輯。

4.4 MBO 於各函數特性分析

依據本實驗組測試函數所擁有的不同的特徵，例如：區域解個數、可分割性(Separable)、連續性、理想值所在區域等。越多的函數特性造成越複雜的求解難度。本小節探討 MBO 求解邏輯的原理與特性，瞭解 MBO 於不同

測試環境底下的優劣性質。

4.4.1 Function 1 : Sphere Model

$$F_1(\mathbf{x}) = \sum_{i=1}^m x_i^2, \mathbf{x} = [x_1, x_2, \dots, x_m]$$

m : dimensions.

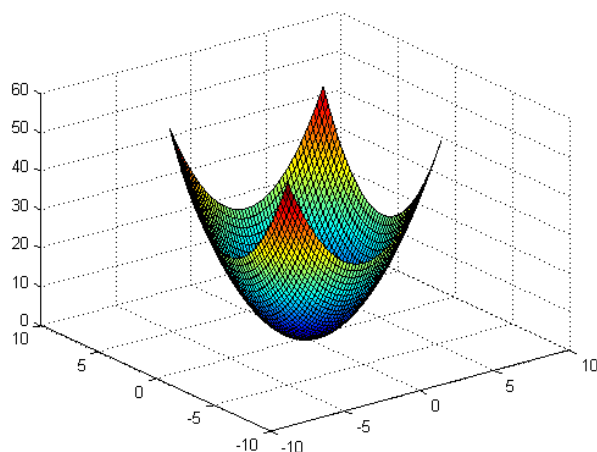


圖 4. 1 Sphere Model 三度空間二維示意圖

函數特性：

- Unimodal
- Separable
- Scalable
- $\mathbf{x} \in [-100, 100]^m$, Global optimum : \mathbf{x}^* , $F_1(\mathbf{x}^*) = 0$

此函數為單一區域解，具有可分割性，變數維度 $m = 10$ ，範圍為[-100, 100]，函數之理想(Ideal)最佳值為 0。

表 4. 10 MBO 於 Sphere Model 各階段效能

交配飛行		交配操作		工蜂改善子代	
誤差值改善	評估次數	誤差值改善	評估次數	誤差值改善	評估次數
3%	4%	0%	1%	97%	95%

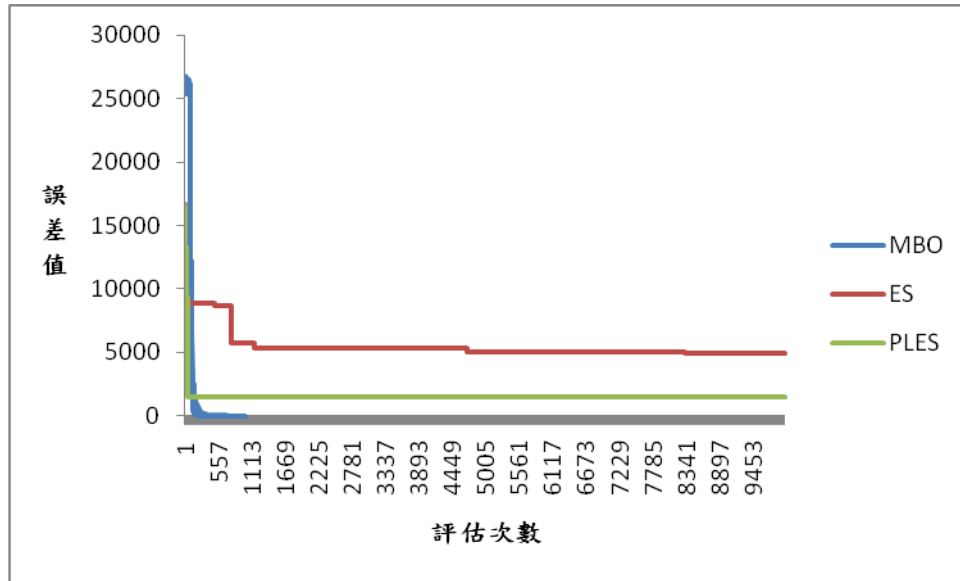


圖 4.2 Function 1 各方法收斂比較

由於不同演算法每次疊代針對函數的評估次數不同，故即便不考慮演算法本身操作原理，評估次數不同單一疊代的表現也會有所差異。MBO 每交配飛行一次為一個疊代，疊代中會進行許多次的區域搜尋改善動作，故以觀察評估次數來觀察其收斂情形的比較，較為客觀。圖 4.2 為 MBO 經過一次疊代演算後所得到的評估次數情形，比較對象是 ES 與 PLES。由圖形可以明顯觀察出 MBO 與其他演算法的收斂速度差異，證明 MBO 在此函數收斂能力上的確顯著優於比較對象，在相對極短的評估次數內達到理想值。

MBO 擁有模擬退火、交配操作與區域搜尋競爭機制。在此函數下，各階段的收斂情況如圖 4.3。藍線為模擬退火中對雄蜂的評估值，紅線為交配完後所產生的子代誤差值綠色則為工蜂對子代改善的評估，紫色線為當前最佳值。一開始 SA 與 GA 的交配操作完畢之後，誤差值並沒有明顯改變，下一階段進入區域搜尋則很快便逼近最佳值，最後達到收斂結束此次搜尋。

模擬退火函數是依據蜂后與雄蜂的目標函數值差異與具有遞減特性的飛行速度，搜尋時間由能量的限制，做為納入精巢的篩選機率。在初始解環境底下，由於雄蜂是以隨機的方式在蜂后四周產生，此時的蜂后所代表的函數解尚未達到有效改善，蜂后所代表的函數值與隨機搜尋到的雄蜂並沒有太大差異。故退火函數並沒有明顯的收斂效果，整個退火程序緊隨飛行速度遞減與能量下降至零而結束，進入下一階段—交配。

如同交配飛行的退火程序，一開始搜尋到的雄蜂基因與蜂后沒有太大差異，此時 MBO 的單體交配操作對目標函數值沒有明顯差異。此時所得到的子代，主要是做為接下來區域搜尋改善的起始依據。在進入區域搜尋階段時，一開始評估值下降相當快，一段時間後趨緩。觀察圖形可發現靠近函數邊界的地形較陡峭，越接近函數理想值則越緩和，此現象可以說明 MBO 所應用的區域搜尋法則對於地形四周環境有相當敏感度。

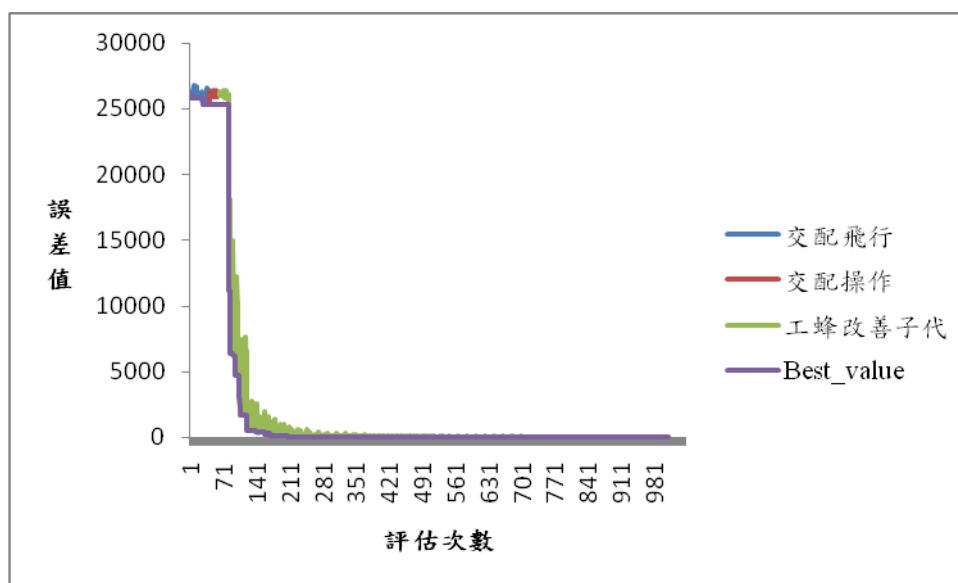


圖 4.3 Function 1 MBO 各階段收斂情形

此函數為典型的單一區域最佳解問題，目標式沒有複雜的計算結構，考驗的是演算法區域搜尋的能力。MBO 工蜂改善子代(區域搜尋)中，其每次改善皆朝鄰近值最佳的方向前進，對環境變化的反應快速且正確，故在此函數中能有較佳的效能表現。

4.4.2 Function 2 : Schwefel's Problem 1.2

$$F_2(x) = \sum_{i=1}^m \left(\sum_{j=1}^i x_j \right)^2, x = [x_1, x_2, \dots, x_m]$$

m : dimensions.

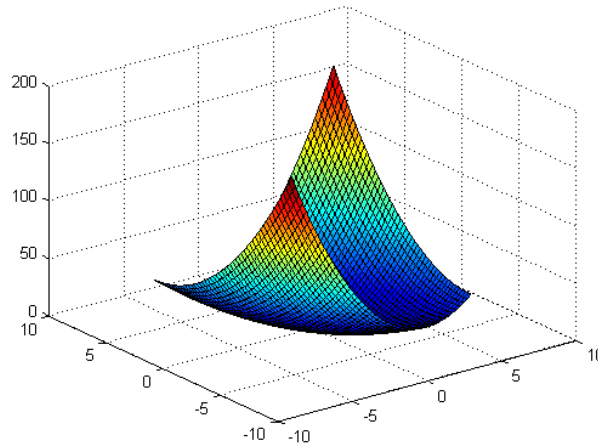


圖 4.4 Schwefel's Problem 1.2 三度空間二維示意圖

函數特性：

- Unimodal
- Non-separable
- Scalable
- $x \in [-100, 100]^m$, Global optimum : x^* , $F_2(X^*) = 0$

此函數為單一區域解，函數不具有可分割性，變數維度 $m = 10$ ，範圍為 $[-100, 100]$ ，函數之理想(Ideal)最佳值為 0。同樣為單一區域解的問題，與 Function 1 不同在於此數學函數具有不可分割的特性。目標函數值受變數個數影響，圖形本身對稱。MBO 在此函數的表現相當好，表示分割性不足以影響 MBO 在區域搜尋的求解效能。

表 4.11 MBO 於 Schwefel's Problem 1.2 各階段效能

交配飛行		交配操作		工蜂改善子代	
誤差值改善	評估次數	誤差值改善	評估次數	誤差值改善	評估次數
50%	7%	0%	2%	50%	91%

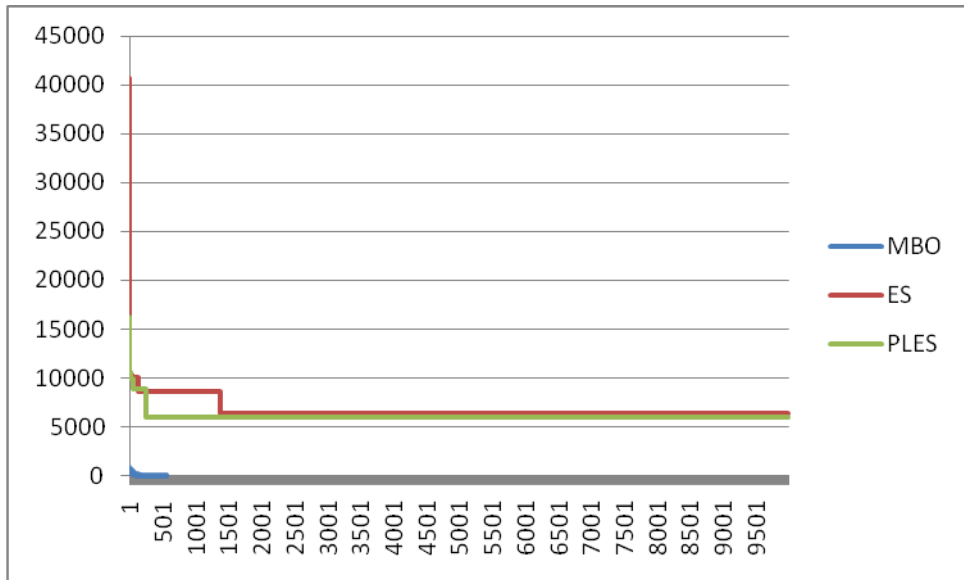


圖 4.5 Function 2 各方法收斂比較

本函數的實驗結果，MBO 僅需一次疊代便能找到最佳值而收斂，然而由於函數本身的特性，從三度空間的圖形可以看出，兩個角落函數值的落差相當明顯，使得初始階段的模擬退火在搜尋改善比例較前一個函數突出。另外兩個角落則因與全域最佳值的落差較小，工蜂改善子代時相較於前個函數的表現則多費較多評估次數。

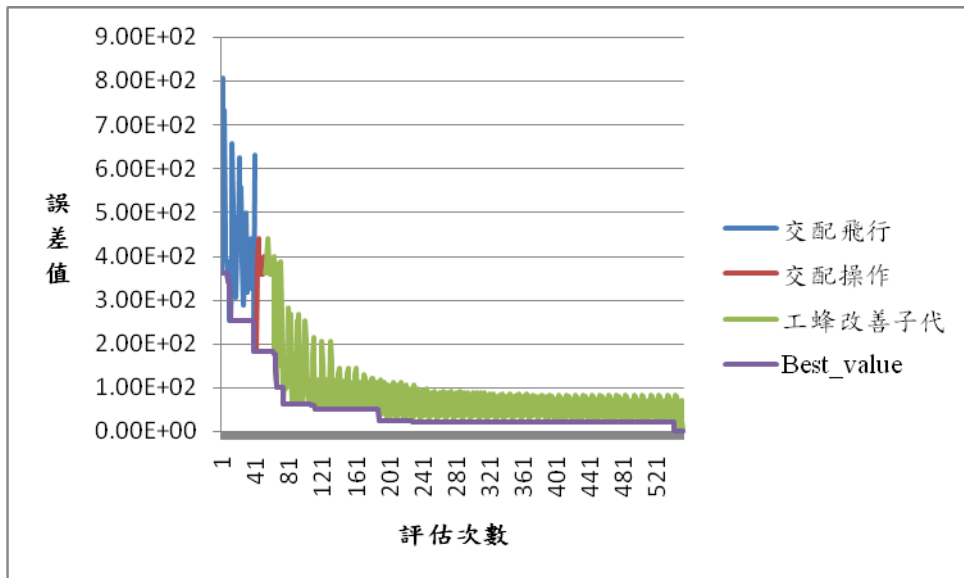


圖 4.6 Function 2 MBO 各階段收斂情形

4.4.3 Function 3 : Rotated High conditioned Elliptic Function

$$F_3(\mathbf{x}) = \sum_{i=1}^m \left(10^6\right)^{\frac{i-1}{m-1}} x_i^2, \mathbf{x} = [x_1, x_2, \dots, x_m]$$

m : dimensions.

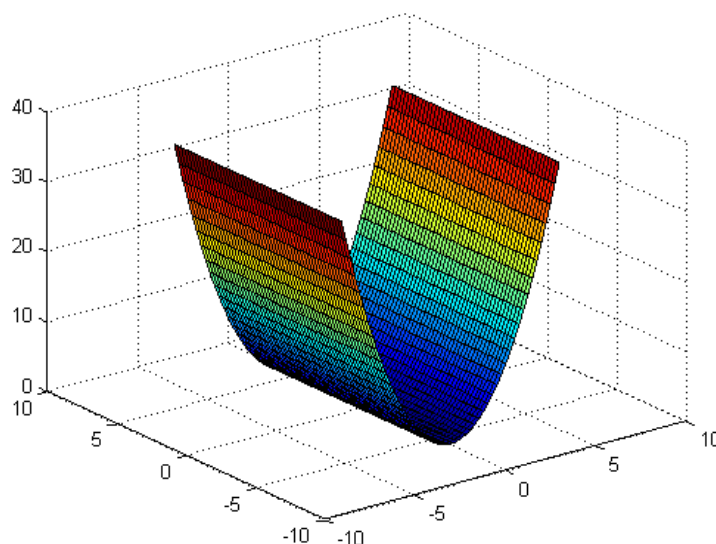


圖 4.7 Rotated High conditioned Elliptic Function 三度空間二維示意圖

函數特性：

- Unimodal
- Rotated
- Non-separable
- Scalable
- $\mathbf{x} \in [-100, 100]^m$, Global optimum : \mathbf{x}^* , $F_3(\mathbf{x}^*) = 0$

此函數為單一區域解，函數不具有可分割性，變數維度 $m = 10$ ，範圍為 $[-100, 100]$ ，函數之理想(Ideal)最佳值為 0。

此函數的特點在於加了旋轉(Rotated)的特性，從收斂圖形上可看出，蜂后在交配操作上會產生極差的子代，且圖形底部極為平緩。工蜂改善子代在此的表現會因此現象造成收斂速度較慢的影響。主要是因為子代改善的移動方向，在圖形平緩環境中搜尋較為不利；MBO 求解過程，區域搜尋

表 4.12 MBO 於 Rotated High conditioned Elliptic Function 各階段效能

交配飛行		交配操作		工蜂改善子代	
誤差值改善	評估次數	誤差值改善	評估次數	誤差值改善	評估次數
2%	3%	0%	2%	98%	95%

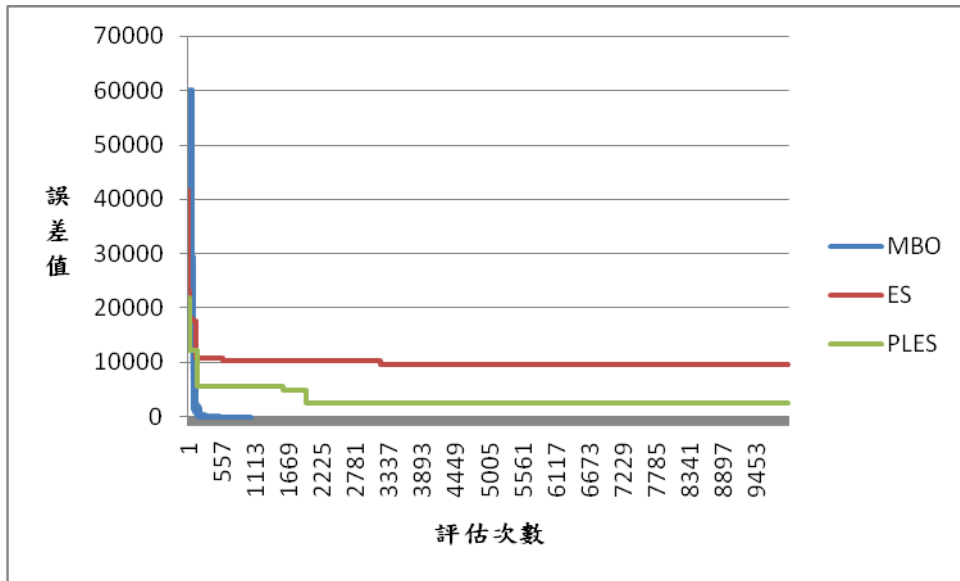


圖 4.8 Function 3 各方法收斂比較

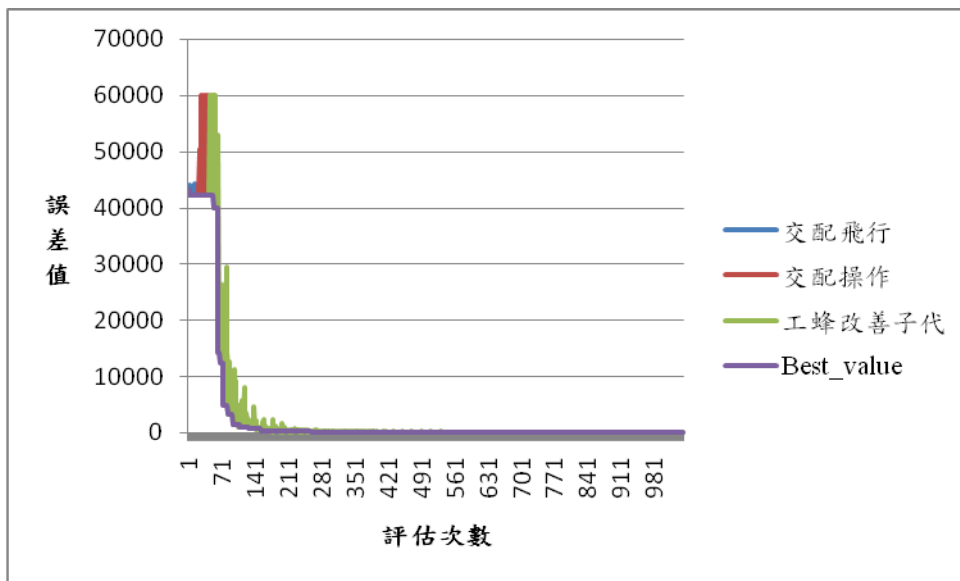


圖 4.9 Finction 3 MBO 各階段收斂情形

每隔固定的子代改善數後，便會跳離該區域的搜尋，蜂后重新一次交配飛行，這點對於 MBO 在太過平緩的區域也是一項不利搜尋的因素。然而針對此函數，只要演化代數夠多，MBO 還是能達到最佳解。

4.4.4 Function 4 : Generalized Rosenbrock's Function

$$F_4(\mathbf{x}) = \sum_{i=1}^{m-1} (100(x_i^2 - x_{i+1})^2 + (x_i - 1)^2), \mathbf{x} = [x_1, x_2, \dots, x_m]$$

m : dimensions.

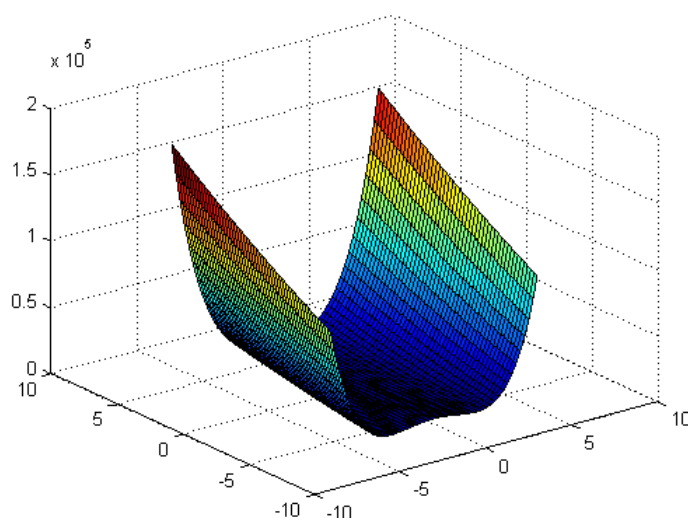


圖 4. 10 Generalized Rosenbrock's Function 三度空間二維示意圖

函數特性：

- Multi-modal
- Non-separable
- Scalable
- Having a very narrow valley form local optimum to global optimum
- $\mathbf{x} \in [-100, 100]^m$, Global optimum : \mathbf{x}^* , $F_4(\mathbf{x}^*) = 0$

此函數為為一個多重區域解的函數，不具有可分割性，變數維度 $m = 10$ ，範圍為 $[-100, 100]$ ，函數之理想(Ideal)最佳值為 0。此函數最特殊的地方在於，區域最佳解與全域最佳解之間有一個小小的緩坡。

表 4.13 MBO 於 Generalized Rosenbrock's Function 各階段效能

交配飛行		交配操作		工蜂改善子代	
誤差值改善	評估次數	誤差值改善	評估次數	誤差值改善	評估次數
10%	7%	0%	3%	90%	90%

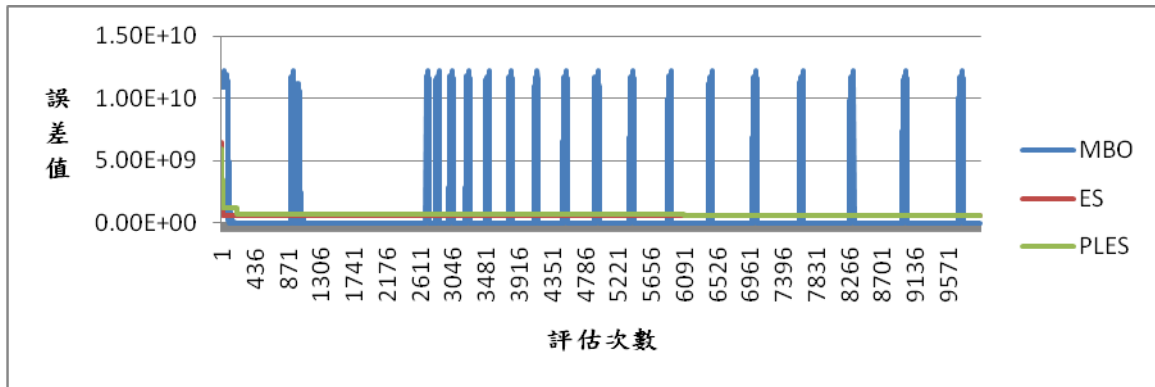


圖 4.11 Function 4 各方法收斂比較

MBO 在全域搜尋上利用的是結合基因演算法的交配操作與退火函數，然而每次交配飛行所蒐集到的雄蜂基因，為了考量每代的改善狀況，故是以蜂后當下的位置為起始出發點，精巢所能蒐集的區域有限；在面對多重區域解彼此間高低落差太過平緩時，因距離另一坡谷的距離太遠，蜂后本身的能量一次到達不了，無法有效發揮跳脫區域解的效果。再加上工蜂改善子代擁有保持對四周環境敏感的特性，在圖形上只會往當下四周更低的方向前進，故很容易陷入區域最佳解中無法跳脫，造成此函數無法達到收斂的情況。

4.4.5 Function 5 : Generalized Rastrigin's Function

$$F_5(x) = \sum_{i=1}^m [x_i^2 - 10 \cos(2\pi x_i) + 10], x = [x_1, x_2, \dots, x_m]$$

m : dimensions.

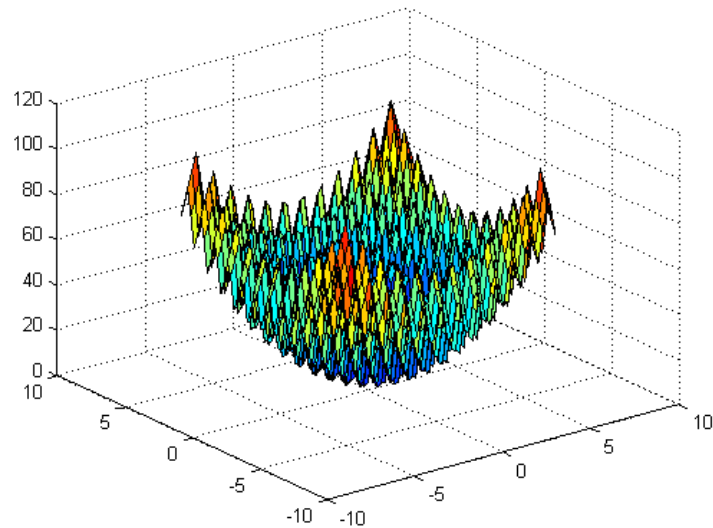


圖 4. 12 Generalized Rastrigin's Function 三度空間二維示意圖

函數特性：

- Multi-modal
- Separable
- Scalable
- Local optima's number is huge
- $x \in [-5, 5]^m$, Global optimum : x^* , $F_5(x^*) = 0$

此函數為多重區域解，函數具有可分割性，變數維度 $m = 10$ ，範圍為 $[-5, 5]$ ，函數之理想(Ideal)最佳值為 0。由 3D 圖形可看出此函數的區域解數量相當龐大，為其最主要之特性。從整體的角度觀察，MBO 在區域解數量龐大的情況底下，評估的疊代數較多，但相較於比較對象，其收斂的速度仍然較為理想。

表 4. 14 MBO 於 Generalized Rastrigin's Function 各階段效能

交配飛行		交配操作		工蜂改善子代	
誤差值改善	評估次數	誤差值改善	評估次數	誤差值改善	評估次數
29%	6%	8%	2%	63%	92%

由於函數的特徵在於有數量相當龐大的區域解，為其主要特性，若全域搜尋機制不好，很容易陷入區域最佳解的窘境。MBO 在全域搜尋與區域搜尋有一個平衡機制，每隔一段子代改善後，必須會離開區域搜尋，在當代最佳值為基準做全域探索，開放一段短暫時間提供蜂后做飛行運動，蒐集其他區域可用之雄蜂基因組合，故可輕易避開如此窘境。且配合區域搜尋敏感的特性，每次子代改善皆能快速找到該區域最佳值。故在本函數中的誤差值、計算時間與收斂比例，表現皆相當理想。

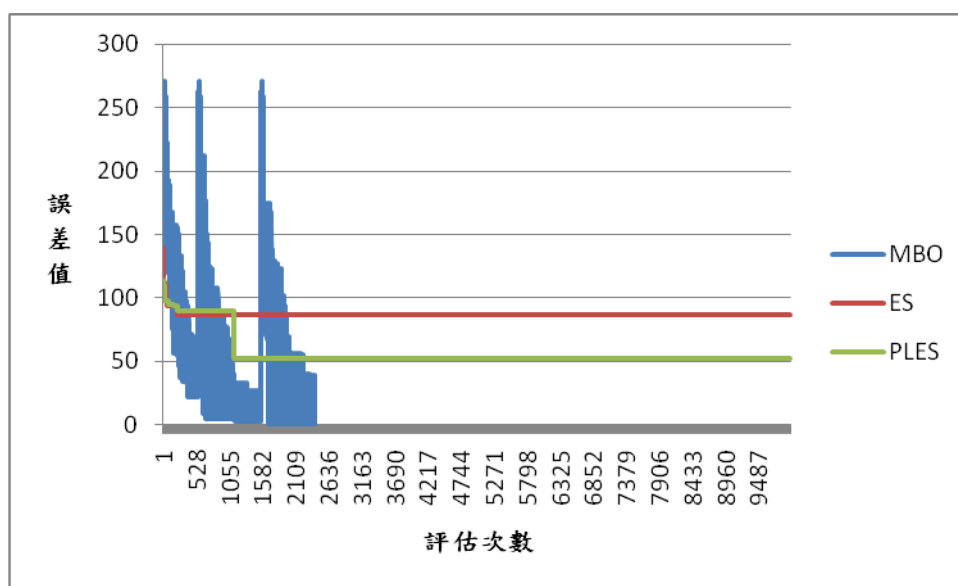


圖 4.13 Function 5 各方法收斂比較

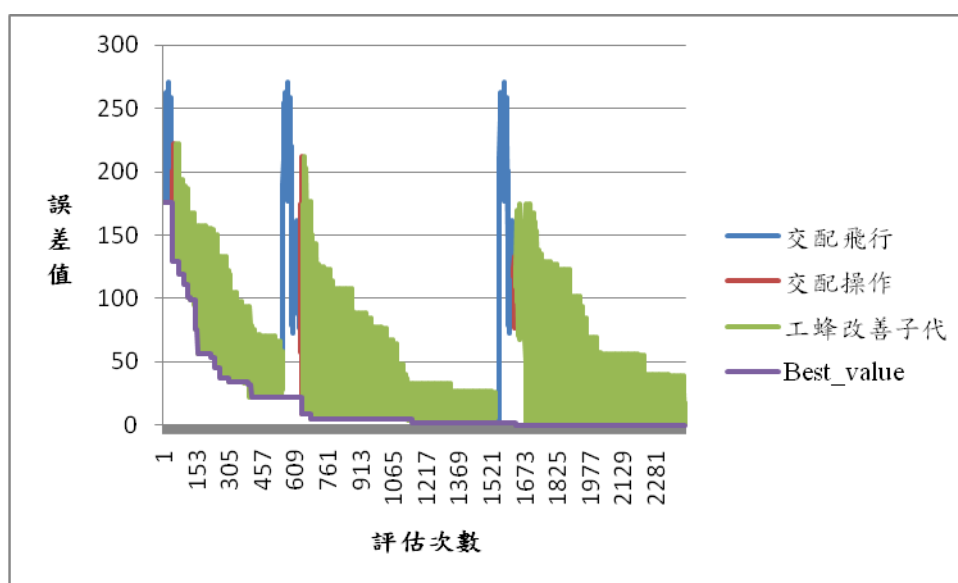


圖 4.14 Function 5 MBO 各階段收斂情形

傳統基因演算法同時以多點方式搜尋最佳解，而非點對點的搜尋，對於多峰谷之函數而言，基因演算法較傳統演算法更可以較快找出整體最佳解，同時也能避免陷入區域最佳解。此特性是基因演算法的最大優點。不過由於計算所需的時間過長、不易掌控，也是另一個缺陷。面對這個缺點 MBO 結合了模擬退火的退火函數，模仿蜜蜂在飛行時的自然現象，以能量守恆定律，隨著飛行時間增加與精巢大小的變化，飛行速度與蜜蜂本身的交配活動能量會跟著減少，來達到模擬退火中溫度下降的效果。如此加快基因演算法在全域搜尋收斂速度。

4.4.6 Function 6 : Ackley's Function

$$F_6(x) = 20 + e - 20e^{-0.2\sqrt{\frac{\sum_{i=1}^m x_i^2}{m}}} - e \sum_{i=1}^m \frac{\cos(2\pi x_i)}{m}, x = [x_1, x_2, \dots, x_m]$$

m : dimensions.

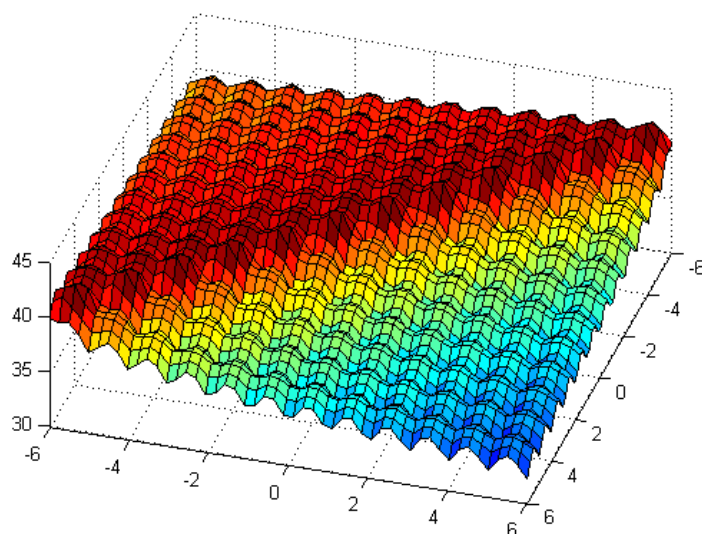


圖 4.15 Ackley's Function 三度空間二維示意圖

函數特性：

- Multi-modal
- Non-separable

- Scalable
- Global optimum on the bound
- If the initialization procedure initializes the population at the bounds, this problem will be solved easily
- $x \in [-32, 32]^m$, Global optimum : x^* , $F_6(x^*) = 0$

此函數為多重區域解，函數本身不具可分割性，變數維度 $m = 10$ ，範圍為 $[-32, 32]$ ，主要差異在於理想值函數之理想(Ideal)最佳值落在邊界上。故若全域搜尋範圍不夠廣泛，則不容易到達全域最佳值的區域。

表 4. 15 MBO 於 Ackley's Function 各階段效能

交配飛行		交配操作		工蜂改善子代	
誤差值改善	評估次數	誤差值改善	評估次數	誤差值改善	評估次數
8%	6%	6%	2%	86%	92%

就 MBO 各階段收斂情形觀察，第一次疊代蜂后交配飛行與交配操作，由於起始值尚未做改善，此時所蒐集到的雄蜂函數值與蜂后並沒有太大差異，即便藉由交配操作來擴充基因型態的多元性，亦無法擁有改善函數值的能力。此時 MBO 著重在於工蜂改善子代的操作，由圖形可知，透過第一階段(疊代)區域搜尋的改善，最後一段的函數值已趨於穩定，找到初始解區域的最佳值位置，系統判斷尚未達到臨界值，此時蜂后便會再度進入交配飛行的操作。

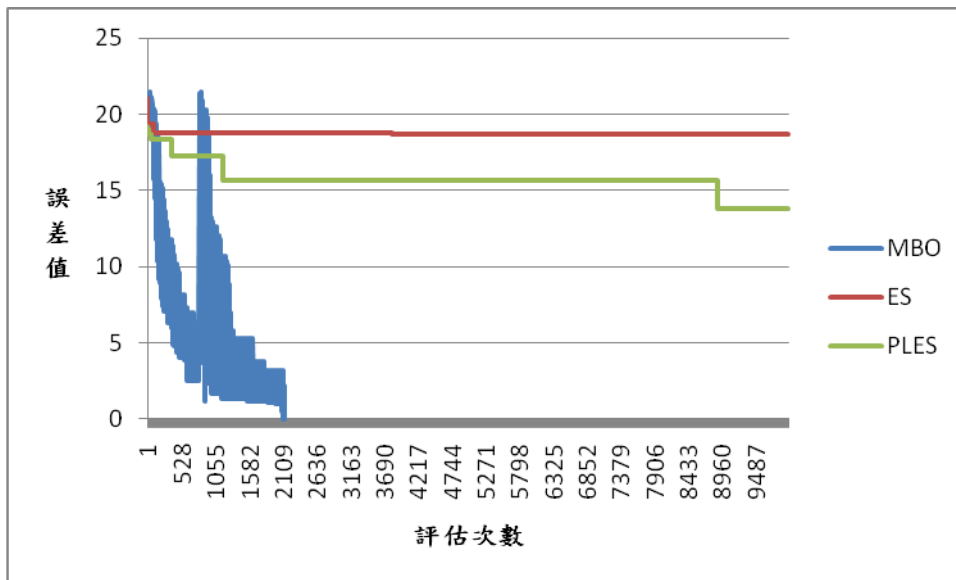


圖 4.16 Function 6 各方法收斂比較

在第二次疊代交配飛行，收斂圖形上形成兩個群落，較高的一群表示 MBO 允許接受較差函數值，相當於初始解的狀況，藉此來跳脫該區域的搜尋；另外隨著蜂后飛行速度的遞減，可接受的函數值相對較小，如此保證此次飛行所蒐集到的解的品質。之後再考慮當前蜂后優良的基因組合，透過交配操作得到另一組卵，作為下一階段工蜂改善子代的起始依據，此次疊代的區域搜尋能很快的便突破前一次疊代所處於的穩定狀況，最後達到全域最佳值的位置。

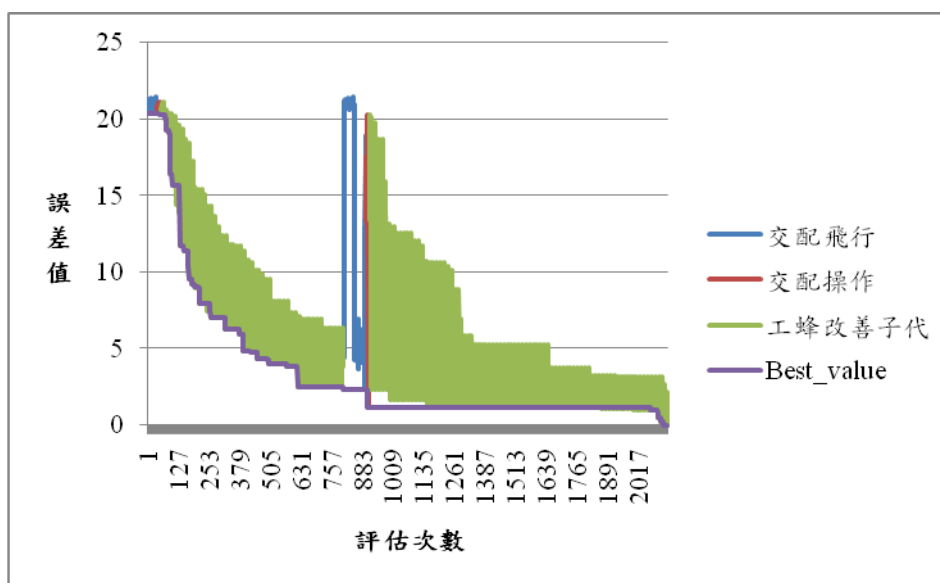


圖 4.17 Function 6 MBO 各階段收斂情形

此函數的全域最佳值在邊界上，但若起始值在邊界附近的話，求解便顯得簡單許多。雖然此函數為多重區域解，且拒不可分割的特性，由實驗結果看來，不論單一區域解或多重區域解的分割性特徵，皆不會對 MBO 搜尋帶來問題。由於 MBO 固定子代改善數量的限制，有助於脫離區域解的環境，在頻繁的全域搜尋次數與區域搜尋迅速的特性下，最佳值發生在邊界並不影響其求解效能。

4.4.7 Function 7 : Generalized Griewank Function

$$F_7(x) = \frac{1}{4000} \sum_{i=1}^m x_i^2 - \prod_{i=1}^m \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1, x = [x_1, x_2, \dots, x_m]$$

m : dimensions.

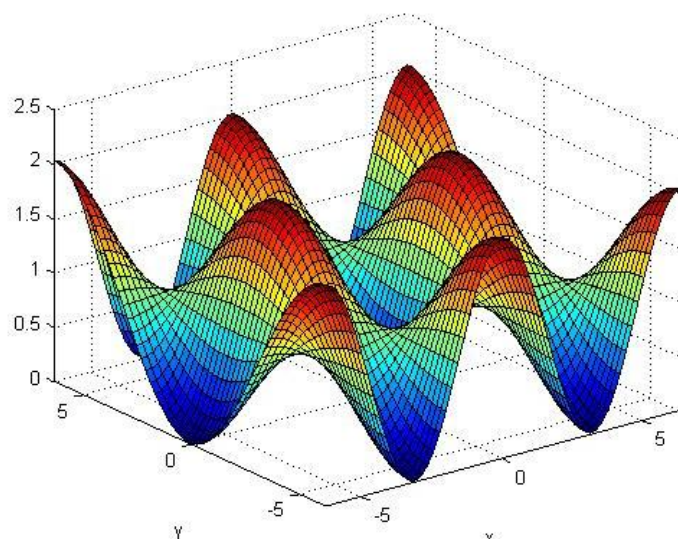


圖 4.18 Generalized Griewank Function 三度空間二維示意圖

函數特性：

- Multi-modal
- Non-separable
- Scalable
- No bounds for variables
- $x \in [-\infty, \infty]^m$, Global optimum : x^* , $F_7(x^*) = 0$

此函數為多重區域解，函數本身不具可分割性，變數維度 $m = 10$ ，範

圍為沒有限制為主要特性，理想值函數之理想(Ideal)為 0。

表 4.16 MBO 於 Generalized Griewank Function 各階段效能

交配飛行		交配操作		工蜂改善子代	
誤差值改善	評估次數	誤差值改善	評估次數	誤差值改善	評估次數
2%	2%	0%	2%	98%	96%

此函數的主要特點除了多重區域解之外，變數範圍沒有限制。MBO 蜂后在執行交配飛行時，會以本身的目標函數值為比較對象，來決定對該雄蜂交配的機率，所以即便搜尋範圍沒有限制，每一世代的交配飛行，蜂后還是會朝著全域最佳方向躍進。又雖然頻繁地做全域搜尋，但因有飛行速度與能量遞減的限制，有助於縮短全域搜尋的時間。在區域搜尋夠快速的情況下，即便變數範圍無限大也能有很好的求解效能。

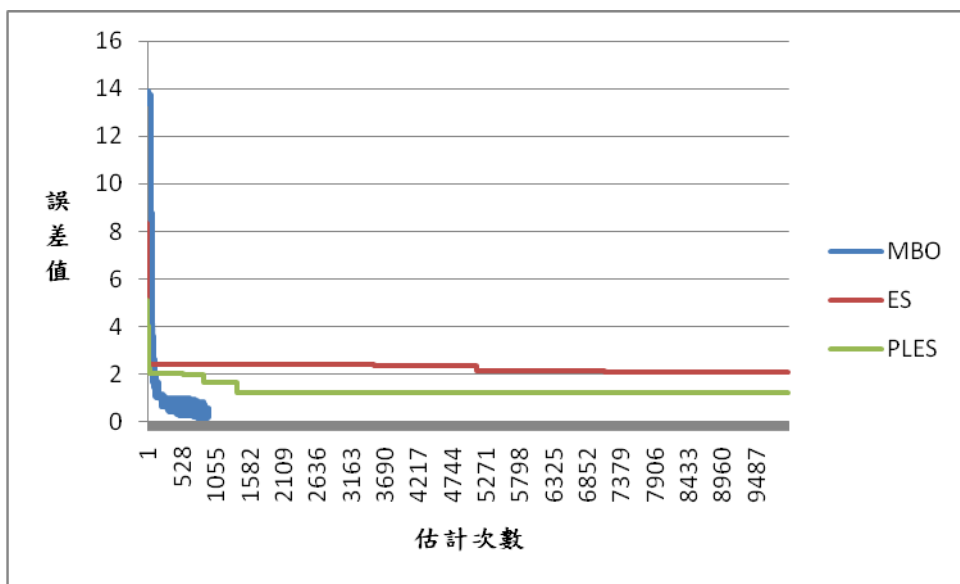


圖 4.19 Function 7 各方法收斂比較

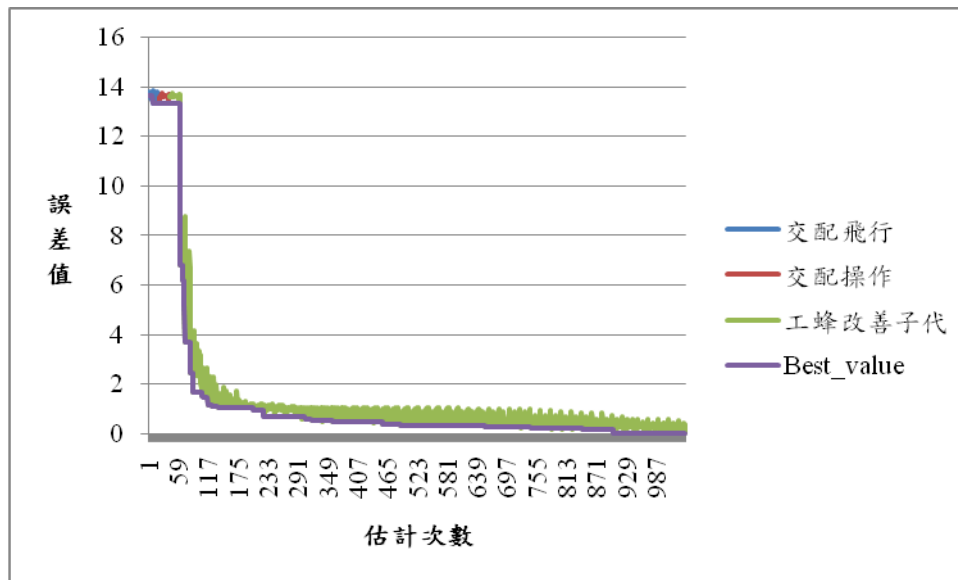


圖 4.20 Function 7 MBO 各階段收斂情形

4.4.8 小結

整體來看，面對不同的函數特性，MBO 皆能有效達到理想值。不論在多重區域解數量龐大、全域最佳值在邊界或變數範圍不限制等情況，求解效能皆有不錯表現。在單一區域解的問題中，MBO 只需一次疊代的評估便能達到收斂；面對多重區域解，透過交配飛行允許接受較差函數值之行為，跳脫區域最佳解的問題。

從演算方法過程可以知道，MBO 結合了模擬退火、基因演算法與單純的區域搜尋法則。如何效果顯著地結合各種方法，底下歸納其幾點特性：

1. 頻繁的交配飛行：經由比較函數值的關係，並考量能量與飛行速度遞減形成退火函數的降溫機制，有效提升求解品質與縮短全域搜尋操作時間。
2. 區域解跳脫機制：透過工蜂改善子代數量的限制，有效平衡全域與區域兩個方向的搜尋。
3. 敏感的區域搜尋：每次改善皆朝鄰近解中最佳的方向前進，能有效且迅速地到達該區域最佳值。

表 4.10 整合上述的函數特性與 MBO 效能表現，從整體的角度來看，MBO 各階段的操作評估次數面對不同函數特性沒有特別差異。若另外從誤

差值改善的比例則有軌跡可循，Function 5 最主要的特性在於多重區域解的數量龐大，故若演算法之搜尋廣度的能力不夠，很容易陷入區域解之中，而透過表列發現，Function 5 中交配飛行的評估次數雖然沒有比較多，但他在此函數誤差值改善貢獻卻特別顯著，表示蜂群透過交配飛行，在多重區域解裡頭能有相當好的表現。

表 4.17 函數特性與 MBO 效能比較表

		函數						
		F 1	F 2	F 3	F 4	F 5	F 6	F 7
特性								
Unimodal		O	O	O				
Multimodal					O	O	O	O
Separable		O				O		
Scalable		O	O	O	O	O	O	O
Rotated				O				O
Others					*	**	***	****
MBO 效能								
誤差值改善比例	交配飛行	3%	50%	2%	10%	29%	8%	2%
	交配操作	0%	0%	0%	0%	8%	6%	0%
	工蜂改善子代	97%	50%	98%	90%	63%	86%	98%
評估次數比例	交配飛行	4%	7%	3%	7%	6%	6%	2%
	交配操作	1%	2%	2%	3%	2%	2%	2%
	工蜂改善子代	95%	91%	95%	90%	92%	92%	96%

註：O = 有此項特性，* = Having a very narrow valley form local optimum to global optimum., ** = Local optima's number is huge., *** = Global optimum on the bound., **** = No bounds for variables.

第五章 結論

5.1 研究總結

本研究除了針對 MBO 做實數函數效能實驗；另外，探討不同函數特性對 MBO 所帶來的影響。本研究的研究結論可分為下列幾點：

1. 啟發式演算法的結合模式，MBO 應用退火函數、交配操作與單純的搜尋法則。先是蜂后交配飛行，效仿能量與飛行速度的概念一再地做全域搜尋，此特點有助於求解多重區域最佳解問題，可避免陷入區域解中無法跳脫的問題。而退火函數考量飛行能量的限制，使退火程序能有效縮短全域搜尋操作時間，且能得到不錯的搜尋品質，因此下一代子代改善除可跳脫區域搜尋的困境，還能有較佳的起始條件。
2. 由第四章的實驗結果得知 MBO 在實數函數應用也有不錯表現，特別是在多重區域解數量龐大時，交配飛行的機制能迅速地找到全域最佳之區域。唯一遺憾是當最佳值附近地形過於平坦時，工蜂改善子代的操作無法在有限代數內改善子代至該區域最佳，在工蜂改善子代中有其他如 random-walk、單點交配等方法，保持求解的多元性，透過實驗得知雖然無法達到收斂效果，但比起比較對象仍可更接近全域最佳解。同樣現象在單一區域解的函數裡，但因沒有跳脫區域解的問題，故只要演化代數夠多，MBO 還是能達到理想的求解。
3. 本研究提出 MBO 在實數問題的參數設定，有效發揮其理論架構，擁有優秀的多重區域解函數的全域搜尋能力以及穩定的最佳解收斂效能，提供往後研究方法一個參考方向。

5.2 後續研究建議

近年來複合型啟發式演算法已成為趨勢，蜂群最佳化提供了一個有建設性的結合模式，提升面對多種不同函數環境適應能力。在瞭解其操作與結合邏輯之後，本研究歸納出幾項值得探討的未來研究方向，依重要性序列如下：

1. 區域搜尋法則之搭配

本研究實驗證實 MBO 全域搜尋穩健的特性，工蜂改善的法則不需過

於繁雜，若針對不同數學函數搭配不同區域搜尋的改善機制，應皆可有不錯的搜尋效率。

2. 啟發式演算法的結合

本研究歸納 MBO 應用基因演算法與模擬退火的組合邏輯，在全域與區域搜尋上達到有效協調，同樣的模式也可考慮在其他啟發式演算法共同的應用上。

3. 區域搜尋與參數的交互作用

實驗證實 MBO 的確是個優秀的演算法則，本研究透過實驗設計也設定出表現不錯的實數參數，但若要更深入研究，建議可朝工蜂改善法則與交配飛行參數之間的關係來做探討。

参考文献

- [1] Abbass, H.A., “MBO: Marriage in Honey Bees Optimization – A Haplometrosis Polygynous Swarming Approach,” The Congress on Evolutionary Computation (CEC2001), Seoul, Korea, pp.207-214, May 2001.
- [2] Abbass, H.A., “A Single Queen Single Worker Honey Bees Approach to 3-SAT,” The Genetic and Evolutionary Computation Conference, GECCO2001, San Francisco, USA, 2001.
- [3] Abbass, H.A. and Teo, J., “A True Annealing Approach to the Marriage in Honey-Bees Optimization Algorithm,” The Inaugural workshop on Artificial Life, AL'01, Adelaide, Dec. 2001.
- [4] Abbass, H.A., “A monogenous MBO approach to satisfiability,” in: Proceeding of the International Conference on Computational Intelligence for Modelling, Control and Automation, CIMCA'2001, Las Vegas, NV, USA, 2001.
- [5] Bäck, T., Evolutionary Algorithms in Theory and Practice. New York: Oxford Univ. Press, 1996.
- [6] Benatchba, K., Admane, L. and Koudil, M., “Using Bees to Solve a Data-Mining Problem Expressed as a Max-Sat One,” IWINAC Vol 2, pp.212-220, 2005.
- [7] Bergh, F. v. d. and Engelbrecht, A. P., “A cooperative approach to particle swarm optimization.,” IEEE Transactions on evolutionary Computation, Vol.8, pp.225-239, 2004.
- [8] Beyer, H.-G., “Toward a theory of evolution strategies: On the benefit of sex—the $(\mu/\mu, \sigma)$ -Theory,” Evolutionary Computation, vol. 3, no. 1, pp. 81–111, 1995.
- [9] Beyer, H.-G., “Toward a theory of evolution strategies: Self-adaptation,” Evolutionary Computation, vol. 3, no. 3, pp. 311–348, 1995.
- [10] Bullnheimer, B., Hartl, R.F., and Strauss, C., “A new rank-based version of the Ant system: a computational study,” Central European Journal Operational Research Economic, Vol. 7, pp.25-38, 1997.
- [11] Clerc, M. and Kennedy, J., “he particle swarm-explosion, stability and convergence in a multidimensional complex space,” IEEE Transactions on Evolutionary Computation, Vol.6, pp.58-73,2002.
- [12] Costa, L., “A Parameter-less Evolution Strategy for Global Optimaztion,” Universidade fo Minho, Escola de Engenharia, Protugal, 2005.
- [13] Colorni, A., Dorigo, M. and Maniezzo, V., “An investigation of some properties of an ant algorithm,” Proceedings of the Parallel Problem Solving from Nature Conference, R. Manner and B. Manderick Eds. Brussels, Belgium: Elsevier, pp.509-520, 1992.
- [14] Colorni, A., Dorigo, M. and Maniezzo, V., “Distributed optimization by ant colonies,” Proceedings of the First European Conference Artificial Life, F. Varela and P. Bourgine,

- Eds. Paris, France: Elsevier, pp.134-142, 1991.
- [15] Deb, K., Anand, A. and Joshi, D., "A Computationally Efficient Evolutionary Algorithm for Real-Parameter Optimization," *Evolutionary Computation* 10(4), 371-395, 2002.
- [16] Dorigo, M. and Maria, L., "Ant colony system: a cooperative learning approach to the traveling salesman problem," *IEEE Transactions evolution Computer*, 1, pp.53-66, 1997.
- [17] Dorigo, M., Bonabeau, E. and Theraulaz, G., "Ant algorithm and stigmergy," *Future Generation Computer Systems*, Vol. 16, pp. 851-871, 2000.
- [18] Dorigo, M., Caro, G.D. and Gambarsella, L.M., "Ant algorithms for discrete optimization," *Artificial Life*, Vol. 5, pp. 137-172, 1999.
- [19] Dorigo, M., Maniezzo, V. and Colorni, A., "Positive feedback as a research strategy," *Technology Report*, pp. 91-016, Politecnico di Milano, 1991.
- [20] Eberhart, R.C. and Kennedy, J. "A new optimizer using particle swarm theory," *Proc. 6th Int. Symp. Micro Machine and Human Science*, Nagoya, Japan, pp.39-43.1995.
- [21] Fathian, M., Amiri, B., Maroosi, A., "Application of honey-bee mating optimization algorithm on clustering," *Applied Mathematics and Computation*. Vol. 190, 2, pp.1502-1513, 2007.
- [22] Fogel, L. J., "Autonomous automata," *Ind. Res.*, Vol. 4, pp. 14-19, 1962.
- [23] Fogel, L. J., "On the organization of intellect," Ph.D. dissertation, University of California, Los Angeles, 1964.
- [24] Fogel, L. J., Owens, A. J., & Walsh, M. J., "Artificial intelligence through simulated evolution." New York: Wiley, 1966.
- [25] Fourie P, and Groenwold, A., "The particle swarm optimization algorithm in size and shape optimization," *Struct Multidiscip Optimiz*, Vol.23, No.4, pp.259-67, 2002.
- [26] Gambardella, L. and Dorigo, M., "Ant-Q: A Reinforcement Learning approach to the traveling salesman problem," *Proceedings ML-95, Twelfth International Conference on Machine Learning*, Tahoe City, A. Prieditis and S. Russell Eds., Morgan Kaufmann, pp.252-260.1995.
- [27] Goldberg, D. E. and Deb, K., A comparison of selection schemes used in genetic algorithms. In *Foundations of Genetic Algorithms 1 (FOGA-1)*, pp. 69-93, 1991.
- [28] Goldberg, D. E., Deb, K. and Korb, B., Messy genetic algorithms revisited: Nonuniform size and scale. *Complex Systems* 4(4), pp. 415-444, 1990.
- [29] Goldberg, D. E., *Genetic Algorithms in Search, Optimazation and Machine Learning*, Addison-Wesley, Reading, MA, 1989.
- [30] Goldberg, D. E., Korb, B. And Deb, K., Messy genetic algorithms: Motivation, analysis and first results. *Complex Systems* 3(5), pp. 493-530, 1989.
- [31] Goldberg, D. E., Real-coded genetic algorithms, virtual alphabets, and blocking. *Complex Systems* 5(2), 139-168, 1991.
- [32] Deb, K., Anand, A. and Joshi, D., "A Computationally Efficient Evolutionary Algorithm

- for Real-Parameter Optimization,” *Evolutionary Computation* 10(4), 371-395, 2002.
- [33] Eshelman, L. J. and Schaffer, J. D., Real-coded genetic algorithms and intervalschemata. In *Foundations of Genetic Algorithms 2 (FOGA-2)*, pp.187-202, 1993.
- [34] Haddad, Bozorg., Afshar, A. and Barry J.Adams, “HBMO in optimal reservoir operation,” ninth international water technology conference, IWTC9 2005, Sahrn El-Sheikh, Egypt, pp.999-1009, 2005.
- [35] Haddad, B., Afshar, A. and Marino, M.A., “Honey bees mating optimization algorithm (HBMO); a new heuristic approach for engineering optimization,” in: *Proceeding of the First International Conference on Modeling, Simulation and Applied Optimization (ICMSA0/05)*, Sharjah, UAE, 1-3 February, 2005.
- [36] Haddad, B., Afshar, A., “MBO (Marriage Bees Optimization), A new heuristic approach in hydrosystems design and operation,” in: *Proceedings of 1st International Conference On Managing Rivers In The 21st Century: Issues and Challenges*, Penang, Malaysia, 21–23 September 2004, pp. 499-504.
- [37] Haddad, O.B., Afshar, A., Marino, M.A., “Honey-Bees Mating Optimization (HBMO) Algorithm: A New Heuristic Approach for Water Resources Optimization”, *Water Resources Management*, Vol. 20, No. 5., pp. 661-680, (22 October 2006).
- [38] Hansen, N. and Ostermeier, A., “Adapting arbitrary normal mutation distributions in evolution strategies: The covariance matrix adaptation.” In *Proceedings of the IEEE International Conference on Evolutionary Computation*, pp. 312-317, 1996.
- [39] Hansen, N. and Ostermeier, “Completely Derandomized Self-Adaptation in Evolution Strategies,” *Evolutionary Computation* 9(2), 159-195, 2001.
- [40] Higuchi, T., Tsutsui, S., and Yamamura, M., "Theoretical analysis of simplex crossover for real-coded genetic algorithms." In Schoenauer, M. et al., editors, *Parallel Problem Solving from Nature (PPSN-VI)*, pages 365-374, Springer, Berlin, Germany, 2000.
- [41] Holland, J. H. and Reitman, J. S., "Cognitive systems based on adaptive algorithms," in *Pattern-Directed Inference Systems*, D. A. Waterman and F. Hayes-Roth, Eds. New York: Academic, 1978.
- [42] Holland, J. H., "Outline for a logical theory of adaptive systems," *J. Assoc. Comput. Mach.*, Vol. 3, pp. 297-314, 1962.
- [43] Holland, J. H., *Adaptation in Natural and Artificial Systems*. Ann Arbor, MI: Univ. of Michigan Press, 1975.
- [44] Kirkpatrick, S., Gelatt, C.D., and Vecchi, M.P., "Optimization by simulated annealing," *Science*, Vol. 220, No. 4598, pp.671-680., 1983.
- [45] Koudil, M., Benatchba, K., Tarabet, A., Sahraoui, E. B., “Using Artificial Bees to Solve Partitioning and Scheduling Problems in Codesign,” *Applied Mathematics and Computation*, col. 186, pp.1710-1722, 2007.
- [46] Michalewicz, Z., “Genetic Algorithms + Data Structures = Evolution Programs,”

Springer Verlag, New York, NY., 1992.

- [47] Monmarche, N., Venturini, G. and Slimane, M., "On How Pachycondyla Apicalis Ants Suggest a New Search Algorithm," *Future Generation Computer Systems* Vol. 16, pp. 937-946, 2000.
- [48] Murata, T. and Ishibuchi, H., "Performance evaluation of genetic algorithms for flowshop scheduling problems," *Proceedings of 1st IEEE Conf. Evolutionary Computation*, Orlando., 1994.
- [49] Ono, I. and Kobayashi, S., "A real-coded genetic algorithm for function optimization using unimodal normal distribution crossover." In Bäck, T., editor, *Proceedings of the Seventh International Conference on Genetic Algorithms (ICGA-7)*, pp. 246–253, Morgan Kaufmann, San Francisco, California, 1997.
- [50] Rechenberg, I., *Evolutionsstrategie: Optimierung technischer Systeme nach Prinzipien der biologischen Evolution*. Stuttgart, Germany: Frommann-Holzboog, 1973.
- [51] Schwefel, H.-P. "Collective intelligence in evolving systems," in W. Wolff, C. J. Soeder and F. Drepper (Eds), *Ecodynamics - Contributions to Theoretical Ecology*, pp. 95-100. Berlin: Springer, 1987a.
- [52] Schwefel, H.-P. and Bäck, T. *Artificial evolution: How and why?* In D. Quagliarella, J. Périaux, C. Poloni and G. Winter (Eds), *Genetic Algorithms an Evolution Strategies in Engineering an Computer Science: Recent Advances and Industrial Applications*, pp. 1-19. Chichester, UK: Wiley, 1998.
- [53] Schwefel, H.-P., "Kybemetische evolution als strategie der experimentellen forschung in der strmungstechnik," *Diploma thesis*, Technical Univ. of Berlin, 1965.
- [54] Schwefel, H.-P., *Evolution and Optimum Seeking*, Wiley , New York, 1995.
- [55] Schwefel, H.-P., *Numerical Optimization of Computer Models*. Chichester: Wiley, 1981.
- [56] Schwefel, H.-P.,and Rudolph, G., "Contemporary evolution strategies," in *Advances in Artificial Life*. 3rd Int. Conf. on Artificial Life (Lecture Notes in Artificial Intelligence, vol. 929), F. Mor´an, A. Moreno, J. J. Merelo, and P. Chac´on, Eds. Berlin, Germany: Springer, pp. 893–907, 1995.
- [57] Shi, Y.-H. and Eberhart, R.-C., "A Modified Particle Swarm Optimizer," *IEEE International Conference on Evolutionary Computation*, Anchorage, Alaska, May 4-9, 1998.
- [58] Socha, K. and Dorigo, M., "Ant Colony Optimization for Continuous Domains," *European Journal of Operational Research*, Vol.185, pp. 1155-1173, 2008.
- [59] Stutzle, T. and Hoos, H. H., "MAX-MIN Ant System," *Future Generation Computer System*, Vol. 16, pp. 889-914, 2000.