

國立交通大學

土木工程學系碩士班

碩士論文

改良式粒子群優化演算法於
桁架結構最佳化設計之應用



**Application of an Improved Particle Swarm Optimization
Method in Optimization Design of Truss Structures**

研究生：洪冠豪

指導教授：洪士林 博士

中華民國 九十九年八月

改良式粒子群優化演算法於
桁架結構最佳化設計之應用

研究生：洪冠豪

Student：Guan-Hao Hung

指導教授：洪士林

Advisor：Dr. Shih-Lin Hung

國立交通大學

土木工程學系碩士班

碩士論文

The logo of National Chiao Tung University is a circular emblem. It features a gear-like outer border. Inside, there is a shield with a book and a compass. The letters 'NCTU' are prominently displayed in the center, with '1896' at the bottom. The text 'A Thesis' is positioned above the shield, and 'Submitted to Department of Civil Engineering' and 'College of Engineering' are written below it.

A Thesis
Submitted to Department of Civil Engineering
College of Engineering
National Chiao Tung University
In Partial Fulfillment of the Requirements
for the Degree of
Master of Science
in
Civil Engineering

August 2010

Hsinchu, Taiwan, Republic of China

中華民國 九十九 年 八 月

改良式粒子群優化演算法於 桁架結構最佳化設計之應用

學生：洪冠豪

指導教授：洪士林 博士

國立交通大學土木工程學系碩士班

摘 要

最近十年，粒子群優化演算法(Particle Swarm Optimization, PSO)是一款在結構最佳化設計問題上被探討的新方法。它是一種模擬群體智慧概念的仿生演算法，粒子群先是以全域搜尋的方式在搜尋空間內移動並且蒐集資訊，再逐漸轉成局部搜尋，最後則收斂到粒子群經驗中最好解的位置。粒子群優化演算法與其他仿生演算法相比具有參數設定簡單與快速收斂的優點，但是太過快速收斂有可能造成引導到局部最佳解。為了改善粒子群優化演算法在桁架結構最佳化設計問題上的效能，本論文提出兩個策略有效率的控制粒子群收斂過程。第一個策略取名「邊界移動法」，它是利用結構設計最佳解常位於搜尋空間內合理解與不合理解交界的特性，將全域搜尋階段的粒子瞬間移動到鄰近的交界區，因此降低粒子群在不重要區域搜尋的時間。第二個策略稱做「粒子飛離法」，將靠近收斂中心的粒子隨機拋離有限的距離，使得粒子與收斂中心的距離在指定的搜尋半徑內均勻分布，藉此避免粒子群過早收斂。為了測試改良式粒子群優化演算法在桁架結構最佳化設計問題上的效能，數個經典的桁架設計問題在本論文中被測試。測試結果指出，改良式粒子群優化演算法較傳統粒子群優化演算法能更有效率找到桁架設計最佳解。

關鍵字：粒子群優化演算法、仿生演算法、桁架結構最佳化設計

Application of an Improved Particle Swarm Optimization Method in Optimization Design of Truss Structures

Student : Guan-Hao Hung

Advisor : Dr. Shih-Lin Hung

**Department of Civil Engineering
College Engineering
National Chaio Tung University**

Abstract

The particle swarm optimization (PSO), simulating the flying of avian for searching food, is one of bionic algorithms and a popular method for finding the optimal design of structures in the last decade. The convergence speed of PSO is fast for the PSO gradually turns global search into local search. In the early search stage of PSO, a population particles move in the different regions of search space to collect information. Then, the particles converge to the same search region to search solutions according to their best experience and the best solution of swarm. However, the PSO may converge to the local minimum if its convergence speed is too fast. Therefore, how to balance the global search and local search in the PSO is an important research issue. This work proposes two novel strategies to improve the capacity of PSO for solving truss optimization design problems. The first strategy is quickly moving the particle to the boundary between feasible and constraints region. The other strategy is forcing the particle to a random position with random distance which closes to convergence center. Four examples of optimal design of truss structures are employed to verify the performance of the proposed improved-PSO algorithm. The analytic results reveal that particles can move quickly in early search stage based on the first strategy. Moreover, based on the second strategy, particles can keep converging to the global minimum when bounding to local minimum. The analytic results also expose that the proposed PSO algorithm outperforms standard PSO algorithm in convergence speed and the obtained optimal solutions.

Keyword: particle swarm optimization (PSO), bionic algorithm, optimal design of truss structures

誌謝

在研究所中兩年的時間不算長也不算短，但卻已足夠讓人學會許多事情、獲得許多成長。然而在這段期間內，真的是受了許許多多人的照顧及幫助，若不是這些人，恐怕這本論文會難產；若不是這些人，恐怕學生是無法學習到這許多事物；若不是這些人，恐怕學生的人生中就會缺少了這兩年的歡笑聲。

需要感謝的人太多太多了，首要感謝的是恩師洪士林教授，在萬般忙碌之中還是給予學生最棒的指導與建議，讓學生像隻無頭蒼蠅般昏頭時也能相信一定不會迷途，且教授的慈祥總能讓學生感到許多溫暖，真的很感謝能夠遇到洪士林教授，也讓學生更加的堅信遇到一個對的老師對一個學生來說是多麼重要的事情；感謝詹君治副教授的從旁指導以及許多建議，讓學生得以在兩年的期限內順利產出這本論文，沒有因為惰性而延誤畢業的時間，也讓學生知道在重重的要求下都是為了學生的成長；感謝黃炯憲教授及鄭復平副教授兩位口試委員，在百忙之中抽空參加學生的口試，不但給了學生許多寶貴的意見，使本論文更加趨於完整，也讓學生見識到教授們的細心、用心以及經驗的累積；感謝博士班子軒學長、勇奇學長、玉邦學長以及碩士班志銘學長、彥伶學姐在這兩年間給予我的許多指導、建議以及鼓勵，讓我學習了許多新的東西，見識了許多新的事物；感謝與我一同抗戰的家宇、夙修、麗松，有你們當我的同屆戰友讓我感到十分開心，因為我的人生中又多了三個不可抹滅的好朋友；感謝學弟妹阿槍、阿祥、孟軒及思伶，不但可以一同覓食、一同出團、一同出遊，甚至在我忘了拔鑰匙的時候也能有個照應(笑)，更因為你們的關係讓研究室內增添了許多的歡笑及回憶；感謝我的家人，在我就學的這段期間給予我莫大的鼓勵與期許，讓我知道我還有個「家」可以放心停靠；感謝我的女友，雖然真的又傻又遲鈍，但卻也傻傻的一路陪著我，不管在我生氣、落寞還是開心時，總是傻傻的對我好，真的很感謝妳。

感謝再感謝似乎是這篇致謝的重點，但卻又好像不是重點，因為要感謝的人真的太多太多了，多到我其實不知道該把重點放哪邊。不過，我唯一且肯定知道的是「感謝」！感謝周遭一切與我有關的人，因為你們才會有我；因為你們才會讓我像個人一樣有喜怒哀樂；因為你們才會讓我學會成長。在此謹以此篇論文獻給所有愛我及我愛的人。

目錄

摘要.....	I
Abstract.....	II
誌謝.....	III
目錄.....	IV
表目錄.....	VI
圖目錄.....	VIII
符號說明.....	IX
第一章 緒論.....	1
1.1 研究背景.....	1
1.2 研究動機與目的.....	2
1.3 研究步驟.....	4
1.4 論文章節及架構.....	5
第二章 文獻回顧.....	6
2.1 粒子群優化演算法.....	6
2.1.1 原始粒子群優化演算法.....	6
2.1.2 慣性權重(Inertia Weight).....	7
2.1.3 壓縮因子(Constriction factor).....	9
2.1.4 被動聚集因子(Passive congregation).....	9
2.1.5 粒子群蟻群優化演算法.....	10
2.1.6 啟發式粒子群優化演算法.....	11
2.1.7 啟發式粒子群蟻群優化演算法.....	12
2.2 粒子群優化演算法流程.....	13
第三章 研究方法.....	15
3.1 桁架結構最佳化設計.....	15
3.1.1 最佳化設計概念.....	15
3.1.2 二維桁架結構分析.....	16

3.1.3 三維桁架結構分析.....	19
3.2 改良式粒子群優化演算法.....	20
3.2.1 改良式粒子群優化演算法概念.....	20
3.2.2 邊界移動法.....	21
3.2.3 粒子飛離法.....	24
3.2.4 飛回機制(Fly back mechanism).....	27
3.2.5 停止條件.....	28
3.2.6 改良式粒子群優化演算法流程.....	28
第四章 數值案例測試.....	30
4.1 例一：平面 10 根桿件桁架結構(單載，載重情況一).....	30
4.2 例二：平面 10 根桿件桁架結構(單載，載重情況二).....	33
4.3 例三：空間 25 根桿件桁架結構(多載).....	34
4.4 例四：空間 72 根桿件桁架結構(多載).....	35
4.5 例五：空間 120 根桿件桁架結構(單載).....	36
第五章 結論與建議.....	40
5.1 結論.....	40
5.2 建議與展望.....	41
參考文獻.....	43
附表.....	46
附圖.....	78

表目錄

表 4-1 粒子群優化演算法於本研究中的參數設定.....	46
表 4-2 例一，各桿件起始節點及終止節點之編號.....	46
表 4-3 例一，各組設計斷面中所包含之桿件編號.....	47
表 4-4 例一，桁架受力情形.....	47
表 4-5 例一，不同粒子群優化演算法所找到的最佳設計斷面.....	48
表 4-6 例一，最佳設計斷面經結構分析後之各節點位移.....	48
表 4-7 例一，最佳設計斷面經結構分析後之各桿件應力.....	49
表 4-8 例一，最佳設計斷面比較表.....	50
表 4-9 例二，桁架受力情形.....	51
表 4-10 例二，不同粒子群優化演算法所找到的最佳設計斷面.....	51
表 4-11 例二，最佳設計斷面經結構分析後之各節點位移.....	52
表 4-12 例二，最佳設計斷面經結構分析後之各桿件應力.....	52
表 4-13 例二，最佳設計斷面比較表.....	53
表 4-14 例三，各桿件起始節點及終止節點之編號.....	54
表 4-15 例三，各組設計斷面中所包含之桿件編號.....	54
表 4-16 例三，桁架受力情形.....	55
表 4-17 例三，各組設計斷面中桿件之最大容許應力.....	55
表 4-18 例三，不同粒子群優化演算法所找到的最佳設計斷面.....	56
表 4-19 例三，最佳設計斷面經結構分析後之各節點位移.....	57
表 4-20 例三，最佳設計斷面經結構分析後之各桿件應力.....	58
表 4-21 例三，最佳設計斷面比較表.....	59
表 4-22 例四，各桿件起始節點及終止節點之編號.....	60
表 4-23 例四，各組設計斷面中所包含之桿件編號.....	61
表 4-24 例四，桁架受力情形.....	62
表 4-25 例四，不同粒子群優化演算法所找到的最佳設計斷面.....	62
表 4-26 例四，最佳設計斷面經結構分析後之各節點位移(第一組載重)....	63
表 4-27 例四，最佳設計斷面經結構分析後之各節點位移(第二組載重)....	64

表 4-28 例四，最佳設計斷面經結構分析後之各桿件應力(第一組載重)	65
表 4-29 例四，最佳設計斷面經結構分析後之各桿件應力(第二組載重)	66
表 4-30 例四，最佳設計斷面比較表	67
表 4-31 例五，各桿件起始節點及終止節點之編號	68
表 4-32 例五，各組設計斷面中所包含之桿件編號	70
表 4-33 例五，桁架受力情形	70
表 4-34 例五，不同粒子群優化演算法所找到的最佳設計斷面	70
表 4-35 例五，最佳設計斷面經結構分析後之各節點位移	71
表 4-36 例五，最佳設計斷面經結構分析後之最大容許應力	73
表 4-37 例五，最佳設計斷面經結構分析後之各桿件應力	74
表 4-38 例五，最佳設計斷面經 SAP2000 結構分析後之各節點位移	75
表 4-39 例五，最佳設計斷面經 SAP2000 結構分析後之各桿件應力	77



圖目錄

圖 2-1 標準粒子群優化演算法的粒子速度及位置更新示意圖	78
圖 2-2 含被動聚集因子之粒子群優化演算法的粒子速度及位置更新示意圖	78
圖 2-3 粒子群優化演算法流程圖	79
圖 3-1 局部座標及整體座標下的二維桁架桿件示意圖	80
圖 3-2 局部座標及整體座標下的三維桁架桿件示意圖	81
圖 3-3 標準之粒子搜尋示意圖	82
圖 3-4 使用邊界移動法之粒子搜尋示意圖	82
圖 3-5 邊界移動法使用不同次數情況下的收斂情形	83
圖 3-6 邊界移動法使用不同次數情況下的收斂情形	83
圖 3-7 粒子飛離法示意圖	84
圖 3-8 飛離空間縮小的情形	84
圖 3-9 改良式粒子群優化演算法流程圖	85
圖 4-1 平面 10 根桿件桁架結構	86
圖 4-2 空間 25 根桿件桁架結構	86
圖 4-3 空間 72 根桿件桁架結構	87
圖 4-4 空間 120 根桿件桁架結構	88
圖 4-5 例一，不同的粒子群優化演算法收斂結果	89
圖 4-6 例一，三種改良式粒子群優化演算法之比較	89
圖 4-7 例二，不同的粒子群優化演算法收斂結果	90
圖 4-8 例三，不同的粒子群優化演算法收斂結果	90
圖 4-9 例四，不同的粒子群優化演算法收斂結果	91
圖 4-10 例五，不同的粒子群優化演算法收斂結果	91

符號說明

- a : 最大容許應力除以各桿件應力中的最大值
- A_i : 第 i 根桿件的斷面積
- A^* : 經由邊界移動法後得到之新的斷面積
- b : 最大容許位移除以各節點位移中的最大值
- BS_u : 基準空間的上限
- BS_l : 基準空間的下限
- c_1 、 c_2 : 學習因子
- c_3 : 被動聚集因子係數
- C_C : 臨界長細比
- D : 變數數量
- E : 桿件材料的楊式係數
- f : 外力點
- $f(x)$: 適應度函數
- F_y : 鋼材的降伏應力
- FS : 飛離空間
- $\{F\}$: 整體座標下的桿件內力
- $\{\bar{F}\}$: 整體座標下完整的節點受力
- k : 迭代數
- k_{max} : 最大迭代數
- $[k]$: 局部座標下的桿件勁度矩陣
- $[K]$: 整體座標下的桿件勁度矩陣
- $[\bar{K}]$: 整體座標下的完整勁度矩陣
- L_i : 第 i 根桿件的桿件長度
- n : 桿件個數
- m : 自由度個數
- M : 群體規模
- P_i : 第 i 個粒子所經歷過的最佳位置向量



P_i^0 : 第 i 個粒子的初始最佳位置向量

P_g : 所有粒子經歷過的最佳位置向量

P_g^0 : 所有粒子的初始最佳位置向量

$\{Q\}$: 局部座標下的桿件內力

r : 迴轉半徑

r_1 、 r_2 、 r_3 : 為 $[0,1]$ 之間的隨機均勻亂數

R_i : 粒子群中隨機選擇的一個粒子

s : 反力點

$smax$: 搜尋空間上限

$smax^*$: 縮小後的搜尋空間上限

$smin$: 搜尋空間下限

$smin^*$: 縮小後的搜尋空間下限

$[T]$: 座標轉換矩陣

V_i : 第 i 個粒子的速度向量

V_{max} : 最大速度

w : 慣性權重

X_i : 第 i 個粒子的位置向量

X_i^0 : 第 i 個粒子的初始位置向量

$\{\delta\}$: 局部座標下的節點位移

Δ_j : 第 j 個自由度的位移

$\Delta_{j,a}$: 第 j 個自由度的最大容許位移

$\{\Delta\}$: 整體座標下的節點位移

$\{\bar{\Delta}\}$: 整體座標下完整的節點位移

λ : 長細比

ρ_i : 第 i 根桿件的材料密度

σ_i : 第 i 根桿件所受到的應力

$\sigma_{i,a}$: 第 i 根桿件的最大容許應力



第一章 緒論

1.1 研究背景

最佳化(Optimization)是指在一個有許多限制和條件相互衝突的環境下，找尋一個最適當答案的過程，而最適當的答案則表示最好的妥協。然而在實際應用上，改進是相當重要的，甚至連極微小的改進都是非常值得的，因此才會有最佳化設計的出現。最佳化設計是根據最佳化原理及方法，並綜合各方面的因素，以「自動搜尋」或「人機配合」的方式，在計算機上進行自動或半自動設計，以選出在現有條件下之最佳設計方案的一種現代設計方法【1】。最佳化的方法在許多不同領域皆有廣泛的應用，如：數學、應用科學、工程、經濟、統計學及醫學等。而在土木工程方面，通常則以最少的成本為目標來達到最佳化目的。

桁架結構為工程上常見的一種結構系統，其力學分析過程只需考慮到軸力，因此結構設計過程較其它結構系統來的簡易。桁架結構最佳化通常可分為最佳桿件斷面尺寸設計(Size optimization)、最佳結構配置(Configuration optimization)、最佳拓撲設計(Topology optimization) 等三類，其中又以桿件斷面尺寸最佳化最為普遍【2】。

在早期，最佳化設計大多利用梯度做為搜尋方向的資訊，對問題極值求解有很好的計算效率。此種方法的特點在於初始點位，若初始點位在問題的全域最佳解附近，則可以很快速的搜尋到全域最佳解；反之，則會陷入局部最佳解中而無法跳脫【3】。因此，後來就有許多學者為了改善傳統梯度法的這項缺點，而陸續發展出許多不需依賴梯度資訊的最佳化演算法，如：基因遺傳演算法(Genetic Algorithms, GA)(John Holland,1975)、模擬退火法(Simulate Anneal, SA)(S.Kirkpatrick,1983)、禁忌搜尋演算法(Tabu Search, TS)(Glover,1986)、螞蟻演算法(Ant Colony Optimization, ACO)(M. Dorigo,1992)、粒子群優化演算法(Particle Swarm Optimization, PSO)(J. KennedyR. & C. Eberhar,1995)及和諧搜尋演算法(Harmony search, HS)(Geem et al,2001)等。上述這些新的演算法有一個共通點，就是都能在不用梯度資訊的情況下，還能擁有比傳統梯度法更好的全域搜尋能力，因此在無梯度資訊下也能找到比使用傳統梯度法要更好的最佳解【4】。粒子群優化演算法為一門新的、一種具有群體智慧概念的仿生演算法，在問題空間中透過

迭代的運算來搜尋全域最佳解，其主要特色在於較少的參數設定及收斂速度快，且具有分散式搜尋、記憶性及容易與其他演算法結合等特性，故粒子群優化演算法極具求解組合最佳化問題的潛力【4】。

1.2 研究動機與目的

粒子群優化演算法是一種前期為全域搜尋，而後期轉為局部搜尋的最佳化演算法，因此若在前期的全域搜尋中沒有獲得較佳的搜尋資訊，則粒子群會容易因收斂過快，過早進入局部搜尋導致陷入局部最佳解中而無法跳脫。

為了改善粒子群的這種問題，自 1995 年粒子群優化演算法被提出後，已有許多學者提出一些參數來修改它。Shi 等人【5】於 1998 年提出了慣性權重(Inertia Weight)的想法，藉由慣性權重的加入來改變粒子本身的移動速度，使粒子能在全域搜尋及局部搜尋兩者間，取得一個較佳的平衡點，使能更有效的找到全域最佳解。結果顯示其慣性權重的值在 0.9 至 1.2 之間，雖然需要較多的迭代次數，但卻能有較佳的搜尋效果。另外，根據文獻【5】資料表示，當慣性權重的值為 1.2 時的迭代次數是 0.8 時的 3 倍，顯示當慣性權重越大，其時間成本也會相對增加，但作者在文中只針對一個例子來做測試，因此若要有慣性權重對搜尋特性其更完整的了解，就應多測試各種不同的例子【5】。

郭信川等人【1】於 2004 年提到，Shi 和 Eberhart【6】在 2001 年針對粒子群優化演算法的發展概況與應用方面做一較完整的整理，文中對於改良的粒子群優化演算法分慣性權重(Inertia Weight)、壓縮因子(Constriction Factors)及軌跡動態系統(Tracking Dynamic System)等三類來探討，而在相關的文獻【6、7、8】研究中，則大多都以慣性權重為探討的重點。

He 等人【9】於 2004 年提出了被動聚集因子(passive congregation)的想法，即在粒子的速度更新中，增加了一個被動聚集的部分。這個部分代表了粒子群中任一粒子對其他粒子的影響。其結果顯示，藉由被動聚集因子的加入，使粒子群能在搜尋過程當中，由於認知學習及社會學習部分的值為零或一很小的數，而導致粒子的搜尋停滯時，該方法能夠改善粒子的停滯不前，使粒子具有較大搜尋能力，增加找到最佳解的機率。

另外，亦有學者將粒子群優化演算法與其他最佳化演算法融合使用。Shelokar

等人【10】於 2007 年提出了 PSACO(Particle Swarm Ant Colony Optimization)演算法，PSACO 為一種結合粒子群優化演算法及蟻群演算法之混合式演算法，其主要構想為結合粒子群的全域搜尋特性及蟻群的局部搜尋特性。該演算法的最佳解尋找過程分為兩部分，第一部分為個別粒子以粒子群優化演算法的速度更新機制，來促使粒子做全域搜尋，其後粒子再以蟻群演算法的賀爾蒙引導機制，對各粒子當前的所在位置附近進行局部搜尋。其結果顯示，加入蟻群演算法的賀爾蒙引導機制後，能明顯改善粒子群優化演算法的效能，使其不但能有效地執行全域搜尋，並且達到最佳解或接近最佳解。

Li 等人【11】於 2007 年提出了一種結合粒子群優化演算法及和諧搜尋演算法的混合式演算法，稱為 HPSO(Heuristic Particle Swarm Optimizer)。粒子在搜尋空間中飛行搜尋時，常會因粒子飛出搜尋空間，而需要花費一些時間(迭代次數)來使粒子飛回到搜索空間中做有效的搜尋，因此造成一些多餘的浪費。故 Li 等人的主要想法為粒子群在飛行搜尋過程中，配合使用和諧搜尋來再生這些飛出搜尋空間的粒子，使其能更充分地利用粒子的搜尋行為，有效地提升粒子的搜尋效率。其結果顯示，相較於 PSO 及 PSOPC，HPSO 於搜尋前期擁有更快速的收斂速度且具有很好的搜尋結果。

Kaveh 等人【12】更於 2009 年提出了 HPSACO 演算法，HPSACO 為一種結合粒子群優化演算法、蟻群演算法及和諧搜尋演算法的混合式演算法。就如同 PSACO 之概念，第一步以粒子群優化演算法來對問題做全域搜尋，第二步則以蟻群演算法來做各粒子周圍的局部搜尋，但不同於 PSACO 的是，HPSACO 又加入了和諧演算法來處理一些飛出搜尋範圍的粒子，使其減少粒子在搜尋範圍之外所造成不必要的時間成本浪費。結果顯示 HPSACO 不但於前期就具有很好的收斂速度，且因蟻群演算法的緣故，亦能擁有很不錯的最佳解，大幅的改善粒子群優化演算法的效能。

這些學者不論是藉由修改粒子群優化演算法，或者是與其他最佳化演算法結合使用，不外乎就是希望可以使粒子群擁有較大的機率來跳脫出局部最佳解，或者配合利用各種演算法的優點，如多點局部搜尋或邊界問題的處理，使粒子能找到更多可能的解。

因此在本研究中，亦希望能發展一種改良式的粒子群優化演算法，使其能具

有下列的能力：

- 一、 以改良後的粒子群優化演算法來解決結構最佳化問題。
- 二、 使改良後的粒子群優化演算法具有半全域搜尋之特性。
- 三、 使改良後的粒子群優化演算法具有更快的收斂速度，並同時擁有更佳的精度。

1.3 研究步驟

本研究的研究過程步驟簡述如下：

1. 蒐集粒子群優化演算法及其應用在桁架方面的相關資料及文獻

資料蒐集分成兩部分：在粒子群優化演算法方面，蒐集國內外粒子群優化演算法及其應用在桁架方面的相關期刊、論文及研討會等資料，以瞭解目前學術界對此問題的研究狀況及成果；而在另一方面，基於演算法撰寫之需求，蒐集其所需用到之指令集和工具書，以利演算法之撰寫。

2. 建立粒子群優化演算法及結構矩陣之演算法，並加入新的概念將其設計組合。

本研究之最佳化設計主要係針對桁架結構之桿件斷面尺寸進行調整，因此需先確認並建立粒子群優化演算法，以及桁架結構之結構矩陣演算法。但因同時考慮其束制條件時，演算法執行的步驟、流程及新的概念都會成為影響效能及成果的關鍵，故需將演算法給予適當的設計組合以達到期望之結果。

3. 進行數值案例分析，並適當調整演算法。

透過數個文獻中所提的桁架最佳化設計問題，對本人所撰寫好的演算法進行測試與分析，以驗證改良後演算法的可行性，並適當調整演算法的參數、流程及錯誤發生處。

4. 撰寫論文

最後彙整本研究之相關理論、成果、結論與參考文獻後，將其整理並撰寫成論文。

1.4 論文章節及架構

本研究之論文架構分為五個章節，各章節的內容如下所述：

第一章為緒論，說明本研究的背景、動機、目的及研究步驟。

第二章為文獻回顧，一開始先介紹粒子群優化演算法，從原始粒子群優化演算法開始，再慢慢的回顧粒子群優化演算法的演化，演化的趨勢大約可分成兩種類別，一種是針對粒子群優化演算法本身的參數或概念來做調整及修改，另一種則是與其他最佳化演算法結合使用，而這兩種類別的方法其主要概念都是要補強粒子群優化演算法自身的強健性，最後則介紹標準粒子群優化演算法搜尋最佳解的完整流程。

第三章為研究方法，首先介紹桁架結構最佳化設計的概念及結構分析的過程，接著介紹在本研究中所提出改良式粒子群優化演算法的概念，及所使用到的兩個新的策略，同時也介紹本研究中處理最佳化邊界問題的方法，最後則是介紹改良後粒子群優化演算法其搜尋最佳解的完整流程。

第四章為數值案例測試，蒐集數個桁架結構最佳化設計文獻中的經典案例，並根據本研究所提出的改良式粒子群優化演算法進行數值案例測試，藉此驗證本研究中所提出的兩個策略，相較於標準的粒子群優化演算法，能有效的改善粒子群的收斂速度及解的品質。

第五章為結論與建議，從本文中數個數值案例最佳化後的結果做一些結論，並針對不同粒子群優化演算法的收斂結果，來討論本研究中所提出的兩個策略對於結構最佳化問題之所以能改良的原因，最後則介紹本研究結束之後的未來展望。

第二章 文獻回顧

2.1 粒子群優化演算法

2.1.1 原始粒子群優化演算法

粒子群優化演算法(Particle Swarm Optimization, PSO)是由 Eberhart 和 Kennedy 【13、14】於 1995 年所提出的一種具有群體智慧概念的仿生演算法。粒子群優化演算法的起源為 Eberhart 和 Kennedy 兩位學者經由觀察鳥類的覓食行為得到啟發，進而將其模擬成一最佳化演算法，藉由個體與個體間一些互動的規則，從而產生出特殊的群體行為，來達到最佳化的目標。

粒子群優化演算法的主要概念為假設有一鳥群於存在食物的空間中隨機飛行覓食，一開始並不知道最佳的覓食地點在哪個位置，所以每隻鳥兒則憑藉著各自以往的經驗或直覺，飛往他們覺得較佳的地點進行覓食，而當某些鳥兒發現了更好的覓食地點而前去覓食時，其他的鳥兒則會像接收到信息般一同飛往更好的覓食地點進行覓食【15】。

在粒子群優化演算法的模擬理論中，將「粒子(particle)」模擬成解空間中的一隻鳥，每個粒子所在的位置，都是最佳化問題的一個解，每個解都可對應一個答案，稱之為「目標函數值(objective function value)」或「適應值(fitness value)」【1、16】。每個粒子都擁有其各自的速度，利用本身以往的速度、各自經歷過的最佳解(稱做 pbest)及群體經歷過的最佳解(稱做 gbest)來做為更新的依據，依此決定粒子的移動方向及移動距離，來調整粒子當前的位置，並依靠各粒子所在位置的適應值來判別目前找到解的好壞，利用這樣的機制在解空間中做反覆的迭代搜尋，以找到最佳化後的解答。【4、17】假設有一 N 維(N 個變數)的最佳化問題，則粒子群優化演算法中速度及位置的更新法則如(2-1)及(2-2)式所示。

$$V_i^{k+1} = V_i^k + c_1 r_1 (P_i^k - X_i^k) + c_2 r_2 (P_g^k - X_i^k) \quad (2-1)$$

$$X_i^{k+1} = X_i^k + V_i^{k+1} \quad (2-2)$$

其中：

V_i ：第 i 個粒子的速度向量(註： V_i 中每個元素都有上界)

X_i ：第 i 個粒子的位置向量

P_i ：第 i 個粒子所經歷過的最佳位置向量

P_g ：所有粒子經歷過的最佳位置向量

k ：迭代數

c_1 、 c_2 ：學習因子

r_1 、 r_2 ：為 $[0, 1]$ 之間的隨機均勻亂數

其中學習因子 c_1 及 c_2 則分別代表將粒子推向各自及群體所經歷過最佳解的權重。其值較低時，即允許粒子在到達各自或群體所經歷過的最佳解之前，可在目標區域外做較多次的徘徊搜尋，增加搜尋到全域最佳解的機率，但需花費較多的運算及時間成本；其值較高時，即允許粒子以較快的速度前往各自或群體所經歷過的最佳解，可節省一些不必要的運算及時間以提高收斂速度，但卻容易因此導致粒子突然衝過全域最佳解而發生找不到的狀況。而許多文獻【6、13】中則建議將 c_1 及 c_2 的值取為 2，故本研究中即沿用該數值。

對於(2-1)式可分成三個部分來解讀，第一部分為粒子先前的慣性，即自身先前的速度；第二部分為「認知(cognition)」部分，表示粒子本身的思考；第三部分為「社會(social)」部分，表示粒子間的訊息共享及相互合作。【1、18】故粒子群優化演算法的核心即是利用這三部分加以線性組合來更新粒子速度及位置，並算出適應值以完成問題的最佳化。其粒子的速度及位置更新示意圖如圖 2-1 所示。自從粒子群優化演算法被提出後，就因為該演算法只有少數的參數需要做調整，且粒子的收斂速度很快，因而吸引了許多學者的關注。這些學者針對該演算法做了一些相關的測試，也從粒子群的收斂特性中發現了一些問題，所以學者們做了些相關的研究，希望能藉由對粒子群優化演算法加以改良，進而提升求解的速度及解的品質。學者們也因此而陸續找出了一些改善的方法，改善的方法大致上可分成兩個類別，一是藉由增加參數來修改標準粒子群優化演算法的更新機制，使其擁有更完善的搜尋能力；另一則是將粒子群優化演算法與其他最佳化演算法一同使用，配合著最佳化演算法各自的優點相輔相成，藉此來找到全域最佳解。

2.1.2 慣性權重(Inertia Weight)

Shi 等人【5】於 1998 年提到，將粒子群優化演算法在更新粒子速度時所使用的(2-1)式分成三部分來看，則第一部分為粒子本身先前的速度，即本身以往

的記憶。其功用就在於開發搜尋空間，也就是擴展粒子的搜尋空間，藉此增加粒子搜尋到全域最佳解的機率。若沒有此項，則粒子群會因此而無法記憶先前的速度，只能靠著自身所經歷過的最佳位置及所有粒子所經歷過的最佳位置來做移動與搜尋，使得粒子會很快的往這兩個位置的加權中心來做收斂，而落在加權中心上的粒子，則會因速度變為零而停止搜尋的動作，這使得該粒子群優化演算法變的像是一種局部搜尋的演算法，故(2-1)式中第一部分的加入，使得粒子群優化演算法能擁有全域搜尋之能力，讓演算法成為一種全域搜尋之演算法。在解最佳化問題過程中，很難肯定的說局部搜尋演算法或全域搜尋演算法哪個比較好，只能說該最佳化問題比較適合用這兩種演算法中的哪一個，也就是說在眾多的最佳化問題中，兩種演算法各有各適合使用的地方。因此 Shi 等人【5】提出了一個稱做慣性權重(Inertia Weight)的想法，在加入慣性權重w後，粒子群優化演算法中的速度更新公式則由(2-1)式修正為如下所示：

$$V_i^{k+1} = wV_i^k + c_1r_1(P_i^k - X_i^k) + c_2r_2(P_g^k - X_i^k) \quad (2-3)$$

其中：

w：慣性權重

慣性權重來自於速度遞減的觀念，當慣性權重較大時，粒子第 k+1 代的速度會較接近第 k 代的速度，因此粒子的搜尋方向較為穩定，而搜尋過程中則變成較注重於全域搜尋，但也因粒子本身具有較高的自由搜尋能力，使粒子群能擁有較高的多樣性。不過當慣性權重過大時，則會容易造成粒子群散佈在搜尋空間中，導致粒子群無法收斂至最佳解；反之若慣性權重較小時，粒子第 k+1 代的速度則會由三個部分來共同影響，因而造成粒子在搜尋過程中的速度方向較不穩定，此時粒子所搜尋到的解則有可能因多樣性不足而無法保證該解為問題之最佳解【19、20】。

因此使用慣性權重的最大問題就是在於如何定出一個較為合理的數值，使粒子能夠在全域搜尋與局部搜尋中取得一個較佳的平衡。因此 Shi 等人【21】於 1998 年提出常數式的慣性權重之後，於同年亦接著提出線性遞減式的慣性權重，其主要概念為在搜尋前期時給予較大的慣性權重，讓粒子能擁有較大的速度以進行全域搜尋，其後隨著代數增加而使慣性權重遞減，使其慢慢的降低粒子速度，讓粒

子從全域搜尋漸漸的轉為局部搜尋，藉此達到收斂的效果。故慣性權重可以為一大於零的常數，也可以為一對於時間之正線性或非線性函數，藉由慣性權重的加入來改變粒子本身的移動速度，使粒子能在全域搜尋及局部搜尋兩者間，取得一個較佳的平衡點，使能更有效的找到全域最佳解。而此後的相關研究，則將加入慣性權重後的粒子群優化演算法視為標準粒子群優化演算法來做為研究並加以比較。

2.1.3 壓縮因子(Constriction factor)

Clerc【22】於1999年時，針對粒子群優化演算法的收斂情形，提出了一個稱做壓縮因子(Constriction factor)的想法，此種含壓縮因子的粒子群優化演算法稱為CPSO(Constriction Particle Swarm Optimization)。為了增加粒子群的局部搜尋能力，藉由使用壓縮因子來縮短各粒子的移動速度，並藉此控制粒子的位置，使粒子在整個搜尋過程中，能夠加強並改善粒子的局部搜尋能力。在使用壓縮因子後一開始的研究及應用中，粒子的最大速度被認為是不重要的，因此不需要特別的選擇並設定它，但Eberhart等人【23】在後續的研究及應用中，證實了如果在使用壓縮因子的情況下想要有更好的最佳解，就必須限制粒子的最大速度，而依靠經驗法則所得到的最佳情況即為粒子的最大速度剛好為粒子的最大搜尋空間。另外Eberhart等人【23】亦表示了使用慣性權重的粒子群優化演算法，只要將一些參數做適當的調整，則該速度的更新公式即與使用壓縮因子之粒子群優化演算法的速度更新公式相同，也就是說此種使用壓縮因子的方法，是為慣性權重方法下的一種特例。

2.1.4 被動聚集因子(Passive congregation)

He 等人【9】於 2004 年藉由一些有關群體動物行為的文獻【24】中發現，群體動物中的成員可以經由鄰近的成員來得到一些必要或重要的訊息，進而產生一些對應的行為，不一定需藉由直接接收到某訊息後才會產生相對應的反應，也就是說群體內的每一個個體，對於群體內的其他個體皆含有隱藏性的資訊或訊息，可以幫助每一個個體判斷或進行某些行為，因此 He 等人【9】則藉由此種現象並基於標準粒子群優化演算法下，提出了被動聚集因子(passive congregation)的構

想，此種含被動聚集因子的粒子群優化演算法稱為 PSOPC(Particle Swarm Optimization with Passive Congregation)。在加入被動聚集因子後，標準粒子群優化演算法中的速度更新公式則由(2-3)式修正為如下所示：

$$V_i^{k+1} = wV_i^k + c_1r_1(P_i^k - X_i^k) + c_2r_2(P_g^k - X_i^k) + c_3r_3(R_i^k - X_i^k) \quad (2-4)$$

其中：

R_i ：粒子群中隨機選擇的一個粒子

c_3 ：被動聚集因子係數

r_3 ：為[0,1]之間的隨機均勻亂數

被動聚集因子的概念來自於群體動物的行為而得，He 等人【9】藉由此種行為，於標準粒子群優化演算法的速度更新公式中增加了一個被動聚集的部分。這個部分代表了粒子群中任一粒子對其他粒子的影響。讓粒子速度的更新機制由原先的三個部分修正為如(2-4)式所示，由四個部分來共同影響，其粒子的速度更新示意圖如圖 2-2 所示。後來的許多學者在相關的研究中也都將含被動聚集因子的粒子群優化演算法與學者們各自所提出的改良式粒子群優化演算法一同做比較及討論。

2.1.5 粒子群蟻群優化演算法

Shelokar 等人【10】於 2007 年時提到 Angeline 在文獻【25】中表示，粒子群優化演算法在搜尋效能上有兩個特性，第一個特性是粒子群優化演算法的速度更新機制相較於其他的進化演算法，能更快速的搜尋到一合理之優質解；第二個特性是如果粒子群在平衡狀態下，也就是位於收斂中心時，粒子群會因速度變緩而很難再去搜尋到其他有可能為更佳解的答案，因此停滯了最佳解的更新。於是 Shelokar 等人【10】基於上述第一個特性，採用了粒子群優化演算法的全域搜尋能力，並基於上述第二個特性，Shelokar 等人【10】選擇了蟻群演算法來彌補粒子群優化演算法局部搜尋的弱點，因此 Shelokar 等人【10】提出了一種結合粒子群優化演算法及蟻群演算法的混合式演算法，稱為粒子群蟻群優化演算法(Particle Swarm Ant Colony Optimization, PSACO)。該演算法為一種基於標準粒子群優化演算法概念的改良式混合演算法，其最佳解尋找過程分為兩部分進行，

第一部分為個別粒子以粒子群優化演算法的速度更新機制，來促使粒子做全域搜尋，其後粒子再以蟻群演算法的賀爾蒙引導機制，對各粒子當前的所在位置附近進行局部搜尋，進而提升搜尋到全域最佳解的機率及品質，以這兩步驟持續進行疊代的搜尋，藉此得到全域最佳解。

2.1.6 啟發式粒子群優化演算法

Li 等人【11】於 2007 年提到，在大多數的結構最佳化問題中都含有許多束制條件，但傳統的數學最佳化演算法卻很難用來處理這些含有束制條件的結構最佳化問題，其後陸續有許多處理這些束制條件的方法被提出，而其中最常被使用的就是懲罰函數(penalty functions)的方法【26】。然而也有文獻【27,28】指出，就如同使用其他處理束制條件的方法一樣，使用懲罰函數的方法也有它的缺點，其中最主要的缺點有二，第一是在使用懲罰函數時必須增加一些調整用的參數至演算法中，才能控制懲罰的大小以達到想要的目標；第二就是這些參數加入之後，還必須將其調整到使目標函數及懲罰函數達到一個平衡為止，才能算是一個適當的懲罰函數。如果不能提供適當的懲罰係數，那問題在最佳化過程中將會不容易得到其最佳解，但是在使用懲罰函數時要如何定義一個適當的懲罰係數，往往都是最困難也最不容易達到的。因此有學者【26】則提出另一種處理含束制條件之邊界問題的新方法，稱為飛回機制(fly-back mechanism)。且 Li 等人指出文獻中【26】也提到在含束制條件的結構最佳化設計中，全域最佳解通常都會靠近或在合理解的邊界上，也因此當粒子群一開始隨機初始在合理解的區域中，並在合理解的範圍內進行搜尋，粒子會因為越來越靠近合理解的邊界，而有機會越過邊界並搜尋到一不合理解。在這種情況下會有兩種情形發生，一種是因為違反了最佳化問題具體的限制而發生的不合理解，另一種則是因為違反了變數上下限的限制而發生的不合理解。

Li 等人【11】則提到，由於在大多數結構最佳化設計中，尤其是一些大型或者複雜的結構，找到一個滿意解所花費的時間也是一個主要考量的因素，故如何減少搜尋到全域最佳解的時間也是個很重要的問題。因此 Li 等人【11】於 2007 年提出了一種結合粒子群優化演算法及和諧搜尋演算法的混合式演算法，稱為 HPSO(Heuristic Particle Swarm Optimizer)。其概念在於粒子於搜尋空間中飛行搜

尋時，常會因粒子飛出合理解區域，而發生了上述兩種情形並需要花費一些時間(迭代次數)來使粒子飛回到搜索空間中做有效的搜尋，因此造成一些多餘的浪費。故 Li 等人【11】希望粒子群在飛行搜尋的過程中，使用飛回機制來處理最佳化問題的具體限制，並配合和諧搜尋演算法來處理違反變數上下限之粒子，藉此再生這些粒子使其能更充分地利用粒子的搜尋行為，有效地提升粒子的搜尋效率。

2.1.7 啟發式粒子群蟻群優化演算法

Kaveh 等人【12】於 2009 年時提到，標準粒子群優化演算法如何在全域搜尋及局部搜尋中取得一個平衡是有一定的難度【25】。因此為了改善標準粒子群優化演算法的這種問題，有些學者試著將粒子群優化演算法與其他演算法一同使用，型成不同的混合式演算法，例如 Shelokar 等人【10】於 2007 年將粒子群優化演算法與蟻群演算法混合而成的粒子群蟻群優化演算法(PSACO)，以粒子群優化演算法進行全域搜尋後，再使用蟻群演算法對每個粒子的所在位置附近進行局部搜尋，也就是取兩者之優點將其一同使用以找到全域最佳解；以及 Li 等人【11】於 2007 年將粒子群優化演算法與和諧搜尋演算法混合而成的啟發式粒子群優化演算法(HPSO)，在粒子群搜尋過程中，配合使用和諧搜尋演算法來處理最佳化問題的束制條件問題，以增加粒子群的搜尋效率，並同時降低搜尋到最佳解的時間。因此 Kaveh 等人【12】則於 2009 年提出了結合粒子群優化演算法、蟻群演算法及和諧搜尋演算法的混合式最佳化演算法，稱為啟發式粒子群蟻群優化演算法(Heuristic Particle Swarm Ant Colony Optimization, HPSACO)。其主要的構想為結合粒子群優化演算法的全域搜尋能力、蟻群演算法的局部搜尋能力以及和諧搜尋演算法處理束制條件問題的能力等各種優點於一身，使其能在整個搜尋過程中都能夠利用各種最佳化演算法的優點來促使粒子達到最佳解的搜尋。不同於 Shelokar 等人【10】所提出的粒子群蟻群優化演算法，啟發式粒子群蟻群優化演算法除了多了和諧搜尋演算法來處理束制條件的問題外，另一個不同的地方就是最佳化演算法中所使用的粒子群優化演算法。粒子群蟻群優化演算法所根據的是標準粒子群優化演算法，而啟發式粒子群蟻群優化演算法所根據的則是含被動聚集因子的粒子群優化演算法，Kaveh 等人【12】表示，He 等人【9】於 2004 年提出含被動聚集因子之粒子群優化演算法的同時，也以數個基準函數來測試該演

算法的成效，其測試結果也明顯的表示含被動聚集因子的粒子群優化演算法相較於標準粒子群優化演算法，能擁有更好的收斂速度同時，也擁有更高的準確性。

以啟發式粒子群優化演算法搜尋最佳解的過程如下所述，粒子於一開始隨機產生之後，經由粒子群優化演算法的速度更新機制來使粒子移動到下一個位置以進行全域搜尋的動作，隨後即判斷粒子是否有違反束制條件，如果有違反束制條件，則使用飛回機制或和諧搜尋演算法來重生粒子的位置，接著在每個粒子的所在位置附近進行蟻群演算法，搜尋附近是否有更佳的問題解，接著再一次的判斷粒子是否有違反束制條件，如果有違反束制條件，則再一次的使用飛回機制或和諧搜尋演算法。持續這樣的循環使其能在每一次的迭代中，都能讓粒子進行充分的搜尋，直至達到終止條件或者找到全域最佳解為止。

2.2 粒子群優化演算法流程

標準粒子群優化演算法的演算程序說明如下，其流程圖如圖 2-3 所示【6】：

1. 給定相關設計資料：群體規模(M)、最大速度(V_{max})、最大迭代數(k_{max})、慣性權重(w)、學習因子(c_1 、 c_2)、目標函數、限制函數及結構系統的相關資料。
2. 在解空間中隨機產生粒子群的初始位置 X_i^0 及初始速度 V_i^0 。
3. 對各粒子所在位置的解做結構矩陣分析，並根據目標函數及限制函數計算初始位置的適應值 $f(X_i^0)$ ，且令 $pbest_i^0$ 為各粒子的初始位置 X_i^0 ，而 $gbest^0$ 則為所有 $pbest_i^0$ 中之最佳解。如(2-5)及(2-6)式所示。

$$P_i^0 = X_i^0, \quad i = 1, 2, \dots, M \quad (2-5)$$

$$P_g^0 = \min P_i^0, \quad i = 1, 2, \dots, M \quad (2-6)$$

其中：

P_i^0 ：第 i 個粒子的初始最佳位置向量

P_g^0 ：所有粒子的初始最佳位置向量

X_i^0 ：第 i 個粒子的初始位置向量

M ：群體規模

4. 將各粒子所求出來的 P_i^k 及 P_g^k ，代入(2-3)式中來更新各粒子下一代的搜

尋速度 V_i^{k+1} ，並利用此更新後的速度，代入(2-2)式以更新各粒子下一代的位置 X_i^{k+1} 。

5. 對更新位置後的每個粒子，再進行結構矩陣分析，配合目標函數及限制函數計算 $k+1$ 代的適應值 $f(X_i^{k+1})$ 後，與各粒子自身最佳解的適應值 $f(P_i^k)$ 及所有粒子中最佳解的適應值 $f(P_g^k)$ 做比較，繼而更新 P_i^{k+1} 及 P_g^{k+1} ，如(2-7)及(2-8)式所示。

$$P_i^{k+1} = \begin{cases} P_i^k, & f(X_i^{k+1}) > f(P_i^k), i = 1, 2, \dots, M \\ X_i^{k+1}, & f(X_i^{k+1}) \leq f(P_i^k), i = 1, 2, \dots, M \end{cases} \quad (2-7)$$

$$P_g^{k+1} = \begin{cases} P_g^k, & f(P_i^{k+1}) > f(P_g^k), i = 1, 2, \dots, M \\ P_i^{k+1}, & f(P_i^{k+1}) \leq f(P_g^k), i = 1, 2, \dots, M \end{cases} \quad (2-8)$$

6. 若演算法滿足終止條件時(如：搜尋次數達最大迭代數)，則停止搜尋，否則需重複步驟4至6，直到滿足終止條件為止。



第三章 研究方法

3.1 桁架結構最佳化設計

3.1.1 最佳化設計概念

基本上，一項工程設計的進行必須經由工程師依照相關規範及其他規定，配合著工程師的專業知識、以往的經驗累積以及該工程所要求的功能需求，來做出符合該工程之設計。在工程設計中，一個符合所有應滿足之束制條件的設計，即為一個可行設計。而大多時候的工程設計皆同時擁有多種或無限多種的可行設計，在這麼多的可行設計中，都分別有各種程度不同的差異，以及設計結果的優劣之分，而所謂的最佳化設計，就是利用一系列有系統的方法，在這些可行設計中尋求最適合者。

最早的結構最佳化設計概念，是在 1904 年由 Michell【29】提出的桁架理論中所出現。一般而言，結構最佳化設計可因設計目標的不同而將其區分成三個領域，分別是尺寸最佳化設計(Size optimization design)、形狀最佳化設計(Shape optimization design)及拓樸最佳化設計(Topology optimization design)等。尺寸最佳化主要改變結構中的斷面大小；型狀最佳化主要改變節點連接的位置；而拓樸最佳化則主要改變結構中的材料分配，包含節點以及桿件數目。尺寸最佳化設計於 1960 年由 Schmit【30】提出後，成為了這三個領域中最早被發展的一支。

一般而言，結構最佳化的三個領域中，以尺寸最佳化設計屬最為普遍，而尺寸最佳化設計的操作就是針對一已知外型之結構物，在滿足應力、位移或變形等束制條件下，透過改變桿件斷面的寬度、厚度或斷面積等方式，使該結構達到我們所設定的最小總重或最小成本等設計目標。對於桁架結構而言，其設計目標通常就是求該桁架總重之最小值，而控制目標函數值的限制函數，則是各桿件所受到的應力需在該桿件的最大容許應力內，以及各自由度的位移需在該自由度的最大容許位移內。對於一個具有 n 根桿件及 m 個自由度的桁架結構而言，假設所有桿件及自由度皆需滿足最大容許應力及最大容許位移之限制，則目標函數及限制函數可分別表示成如(2-1)、(2-2)及(2-3)式所示。

目標函數：

$$\min f(x) = \sum_{i=1}^n \rho_i A_i L_i \quad (3-1)$$

限制函數：

$$\sigma_i \leq \sigma_{i,a}, \quad i = 1, 2, \dots, n \quad (3-2)$$

$$\Delta_j \leq \Delta_{j,a}, \quad j = 1, 2, \dots, m \quad (3-3)$$

其中：

ρ_i ：第 i 根桿件的材料密度

A_i ：第 i 根桿件的斷面積

L_i ：第 i 根桿件的桿件長度

σ_i ：第 i 根桿件所受到的應力

$\sigma_{i,a}$ ：第 i 根桿件的最大容許應力

Δ_j ：第 j 個自由度的位移

$\Delta_{j,a}$ ：第 j 個自由度的最大容許位移

3.1.2 二維桁架結構分析

在桁架結構分析部分，採用直接勁度法來分析該桁架結構的整體勁度、節點位移及桿件應力。其分析步驟如下所敘：

由材料力學的觀念可知，單一桿件內力與節點位移的關係如下：

$$\{Q\} = [k]\{\delta\} \quad (3-4)$$

由圖 3-1 所示，(3-4)式展開後為

$$\begin{Bmatrix} Q_1 \\ Q_2 \\ Q_3 \\ Q_4 \end{Bmatrix} = \frac{EA}{L} \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{Bmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \\ \delta_4 \end{Bmatrix} \quad (3-5)$$

其中：

$Q_1 \sim Q_4$ ：局部座標下的桿件內力

$\delta_1 \sim \delta_4$ ：局部座標下的節點位移

E ：桿件材料的楊式係數

A ：桿件斷面積

L：桿件長度

由圖 3-1 中可推得桿件內力於局部座標及整體座標間轉換的關係如(3-6)及(3-7)式所示：

$$\begin{Bmatrix} Q_1 \\ Q_2 \\ Q_3 \\ Q_4 \end{Bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta & 0 & 0 \\ -\sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & \cos\theta & \sin\theta \\ 0 & 0 & -\sin\theta & \cos\theta \end{bmatrix} \begin{Bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \end{Bmatrix} \quad (3-6)$$

$$\begin{Bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \end{Bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & \cos\theta & -\sin\theta \\ 0 & 0 & \sin\theta & \cos\theta \end{bmatrix} \begin{Bmatrix} Q_1 \\ Q_2 \\ Q_3 \\ Q_4 \end{Bmatrix} \quad (3-7)$$

其中：

$F_1 \sim F_4$ ：整體座標下的桿件內力

因此由(3-6)式可定義座標轉換矩陣[T]如下：

$$[T] = \begin{bmatrix} \cos\theta & \sin\theta & 0 & 0 \\ -\sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & \cos\theta & \sin\theta \\ 0 & 0 & -\sin\theta & \cos\theta \end{bmatrix} \quad (3-8)$$

並由(3-6)及(3-7)式可推得

$$[T]^T = [T]^{-1} \quad (3-9)$$

再由圖 3-1 中亦可得知局部座標與整體座標下節點位移的關係如下：

$$\{\delta\} = [T]\{\Delta\} \quad (3-10)$$

由(3-5)及(3-6)式可得

$$\{Q\} = [k]\{\delta\} = [T]\{F\} \quad (3-11)$$

將(3-10)式代入(3-11)式，可得

$$[k][T]\{\Delta\} = [T]\{F\} \quad (3-12)$$

將(3-12)式等號兩邊同乘 $[T]^{-1}$ ，可得

$$\{F\} = [T]^{-1}[k][T]\{\Delta\} = [T]^T[k][T]\{\Delta\} = [K]\{\Delta\} \quad (3-13)$$

其中 $[K]$ 為一根桿件在整體座標下之勁度矩陣，矩陣內容如下：

$$[K] = \frac{EA}{L} \begin{bmatrix} \cos^2\theta & \sin\theta\cos\theta & -\cos^2\theta & -\sin\theta\cos\theta \\ \sin\theta\cos\theta & \sin^2\theta & -\sin\theta\cos\theta & -\sin^2\theta \\ -\cos^2\theta & -\sin\theta\cos\theta & \cos^2\theta & \sin\theta\cos\theta \\ -\sin\theta\cos\theta & -\sin^2\theta & \sin\theta\cos\theta & \sin^2\theta \end{bmatrix} \quad (3-14)$$

再將各桿件所求出來的 $[K]$ ，依照自由度的編號，放置並加總到相對應的位置，即可組成完整結構在整體座標下之勁度矩陣 $[\bar{K}]$ ，矩陣內容如下：

$$[\bar{K}] = \begin{pmatrix} K_{11} & \cdots & K_{1m} \\ \vdots & \ddots & \vdots \\ K_{m1} & \cdots & K_{mm} \end{pmatrix}_{m \times m} \quad (3-15)$$

其中：

$[\bar{K}]$ ：完整結構在整體座標下之勁度矩陣

m ：自由度的個數

將每個節點上所受的力及位移，按照自由度列成行向量，並配合 $[\bar{K}]$ 矩陣後，可將其表示成如(3-13)式之格式如下：

$$\{\bar{F}\} = [\bar{K}]\{\bar{\Delta}\} \quad (3-16)$$

$$\{\bar{F}\} = \begin{Bmatrix} F_1 \\ \vdots \\ F_m \end{Bmatrix}$$

$$\{\bar{\Delta}\} = \begin{Bmatrix} \Delta_1 \\ \vdots \\ \Delta_m \end{Bmatrix}$$

其中：

$\{\bar{F}\}$ ：所有節點所受力之行向量

$\{\bar{\Delta}\}$ ：所有節點所受位移之行向量

m ：自由度的個數

若再將(3-16)式以外力點及反力點來區分的話，則可表示成如下：

$$\begin{Bmatrix} \bar{F}_f \\ \bar{F}_s \end{Bmatrix} = \begin{bmatrix} \bar{K}_{ff} & \bar{K}_{fs} \\ \bar{K}_{sf} & \bar{K}_{ss} \end{bmatrix} \begin{Bmatrix} \bar{\Delta}_f \\ \bar{\Delta}_s \end{Bmatrix} \quad (3-17)$$

其中：

f ：外力點

s ：反力點

因為支承點不會有位移，所以(3-17)式中的 $\bar{\Delta}_s = 0$ ，因此(3-17)式可簡寫如下：

$$\{\bar{F}_f\} = [\bar{K}_{ff}]\{\bar{\Delta}_f\} \quad (3-18)$$

$$\{\bar{F}_s\} = [\bar{K}_{sf}]\{\bar{\Delta}_f\} \quad (3-19)$$

將(3-18)式等號兩邊同乘 $[\bar{K}_{ff}]^{-1}$ ，即可得到節點位移如下：

$$\{\bar{\Delta}_f\} = [\bar{K}_{ff}]^{-1}\{\bar{F}_f\} \quad (3-20)$$

求出節點位移之後，將(3-13)式代入(3-11)式中，即可得到桿件之內力如下：

$$\{Q\} = [T][K]\{\Delta\} \quad (3-21)$$

最後再將(3-21)式中所求得的桿件內力，除上桿件的斷面積，即可得到桿件所受到的應力如下：

$$\sigma = \frac{Q}{A} \quad (3-22)$$

最後，即可將(3-20)及(3-22)式所求得之節點位移及桿件應力，拿來與最大容許位移及最大容許應力比較，檢查該組桿件斷面是否滿足束制條件。

3.1.3 三維桁架結構分析

三維桁架結構與二維桁架結構的直接勁度法觀念，主要差異在於節點的自由度數目。二維桁架結構的節點自由度為 2，但三維桁架結構的節點自由度為 3，如圖 3-2 所示。且由圖 3-2 中可推得在局部座標及整體座標下桿件內力的關係如下所示：

$$\begin{Bmatrix} Q_1 \\ Q_2 \end{Bmatrix} = \begin{bmatrix} c_x & c_y & c_z & 0 & 0 & 0 \\ 0 & 0 & 0 & c_x & c_y & c_z \end{bmatrix} \begin{Bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \\ F_5 \\ F_6 \end{Bmatrix} \quad (3-23)$$

其中：

c_x ：桿件與 X 軸夾角之 cos 值，即 $\cos\theta_x$

c_y ：桿件與 Y 軸夾角之 cos 值，即 $\cos\theta_y$

c_z ：桿件與 Z 軸夾角之 cos 值，即 $\cos\theta_z$

因此可由(3-23)式定義座標轉換矩陣 $[T]$ 如下：

$$[T] = \begin{bmatrix} c_x & c_y & c_z & 0 & 0 & 0 \\ 0 & 0 & 0 & c_x & c_y & c_z \end{bmatrix} \quad (3-24)$$

而一根桿件在整體座標下的勁度矩陣 $[K]$ 則如下：

$$[K] = \frac{EA}{L} \begin{bmatrix} c_x^2 & c_x c_y & c_x c_z & -c_x^2 & -c_x c_y & -c_x c_z \\ c_x c_y & c_y^2 & c_y c_z & -c_x c_y & -c_y^2 & -c_y c_z \\ c_x c_z & c_y c_z & c_z^2 & -c_x c_z & -c_y c_z & -c_z^2 \\ -c_x^2 & -c_x c_y & -c_x c_z & c_x^2 & c_x c_y & c_x c_z \\ -c_x c_y & -c_y^2 & -c_y c_z & c_x c_y & c_y^2 & c_y c_z \\ -c_x c_z & -c_y c_z & -c_z^2 & c_x c_z & c_y c_z & c_z^2 \end{bmatrix} \quad (3-25)$$

除了上述的式子外，其餘的分析計算過程則與二維桁架結構相似，故在此就不再加以贅述。

3.2 改良式粒子群優化演算法

傳統的標準粒子群優化演算法憑藉著圖2-3之流程來使粒子趨向並得到最佳解，但如 1.2 節所述，粒子群若沒有於前期的全域搜尋中獲得較佳的搜尋資訊，則會因過早進入局部搜尋階段，而容易陷入局部最佳解中並無法脫離，因此將在本節中介紹本研究所提出的改良式粒子群優化演算法。

3.2.1 改良式粒子群優化演算法概念

在結構最佳化設計中，最佳解通常會位於解空間中合理解(feasible region)與不合理解(infeasible region)的邊界上，因此粒子群一開始雖然是均勻隨機分佈在解空間中，但是在搜尋最佳解的過程當中，粒子群會逐漸趨向合理解與不合理的邊界。在這個過程當中，粒子是先經過結構矩陣分析後，讓得到的結果成為下一次搜尋的資訊，然後將粒子移動至新的位置，再重複結構矩陣的運算。利用此循環一代又一代慢慢的將粒子推向合理解與不合理的邊界，最後才得到問題的

最佳解。

我們知道粒子於搜尋前期是經由一步步的移動才會趨向合理解與不合理的邊界，在這一步步的過程中，粒子其實停留了很多不必要的位置，也做了很多無謂的結構矩陣運算，浪費了不少時間成本。然而，既然我們知道粒子一定會往合理解與不合理的邊界過去，那何不於一開始隨機產生粒子群後，就直接將粒子們送往合理解與不合理的邊界上呢？

此外，由於粒子群的群體行為，會促使所有粒子逐漸趨向各自及群體所經歷過最佳解的加權中心位置，導致粒子群由前期的全域搜尋在後期轉變為局部搜尋，也因此喪失了問題解的多樣性。倘若粒子群不是收斂在全域最佳解的附近，而是落在局部最佳解的周圍，則粒子群就會很難找到全域最佳解。

基於上述兩個標準粒子群優化演算法的問題，本研究提出了兩個相應的策略來將其做修正，取名後分別為「邊界移動法」及「粒子飛離法(fly away)」。

針對粒子的移動問題，採用邊界移動法將前期全域搜尋階段的粒子瞬間移動到鄰近的邊界上，使粒子群能有效降低在不重要區域的搜尋時間；並使用粒子飛離法將靠近收斂中心的粒子隨機拋離有限的距離，使得粒子在收斂中心與指定的搜尋半徑內均勻分布，藉此避免粒子群因過早收斂而喪失問題解的多樣性，失去找到全域最佳解的可能性。針對這兩項策略，於下一節中做更詳盡的說明。

3.2.2 邊界移動法

在結構最佳化設計中，我們知道最佳解通常是位於合理解與不合理的邊界上，因此我們將這塊區域視為重要區域，而越遠離這塊區域的地方則視為不重要區域。在搜尋前期於解空間中隨機分佈的粒子群，會花費一些時間逐步更新各自及群體所經歷過的最佳解，使粒子群由不重要區域逐漸趨向重要區域，如圖 3-3 所示。因此本研究中提出了一種取名為「邊界移動法」的策略，該策略的主要構想為使前期全域搜尋階段的粒子能在短時間內就可快速移動到鄰近的重要區域上，藉此降低粒子群於不重要區域裡的搜尋時間及一些不必要的運算，該示意圖如圖 3-4 所示。

以結構矩陣做結構分析時，很重要的一個方程式如(3-26)式所示：

$$\{F\} = [K]\{\Delta\} \quad (3-26)$$

其中：

$\{F\}$ ：節點所受的外力矩陣

$[K]$ ：結構之整體勁度矩陣

$\{\Delta\}$ ：節點變位矩陣

由(3-26)式我們知道 $[\Delta] \propto \frac{1}{[K]}$ ，且由(3-25)式亦可知 $[K] \propto A$ ，可得

$$[\Delta] \propto \frac{1}{A} \quad (3-27)$$

同時由(3-22)式中，亦可得到

$$\sigma \propto \frac{1}{A} \quad (3-28)$$

由(3-27)及(3-28)式我們得知，當桁架結構中所有桿件的斷面積同時發生線性改變時，各節點的位移及各桿件的應力也會呈現線性改變。另外，由於桁架結構的尺寸最佳化設計中，需在結構的任一節點位移達到最大容許位移或任一桿件應力達最大容許應力等條件下，得到一組最經濟斷面，即為最佳設計斷面。對於一個具有 n 根桿件及 m 個自由度的桁架結構而言，邊界移動法的主要概念步驟如下：

1. 將一開始隨機產生之粒子，進行結構矩陣分析並求出該組斷面相對應的各節點位移及各桿件應力。
2. 將最大容許應力除以各桿件應力中的最大值，可得到一比值如下所示：

$$a = \frac{\sigma_{i,a}}{\sigma_i}, i = 1, 2, \dots, n \quad (3-29)$$

其中：

$\sigma_{i,a}$ ：第 i 根桿件的最大容許應力

σ_i ：第 i 根桿件的應力

3. 將最大容許位移除以各節點位移中的最大值，可得到另一比值如下所示：

$$b = \frac{\Delta_a}{\max(\Delta_j)}, j = 1, 2, \dots, m \quad (3-30)$$

其中：

Δ_a ：節點的最大容許位移

Δ_j ：第 j 個自由度的位移

4. 將該組桿件斷面積除以比值 a 及 b 的較小者，即可得到一組更新後的斷面積如下所示：

$$A^* = A/\min(a, b) = A/\min\left(\frac{\sigma_{i,a}}{\sigma_i}, \frac{\Delta_a}{\max(\Delta_j)}\right) \quad (3-31)$$

其中：

A^* ：經由邊界移動法後得到之新的斷面積

A ：桿件的斷面積

5. 更新桿件的斷面積後，藉由(2-3)及(2-2)式來更新粒子下一代的速度及位置，並做完結構矩陣分析，再重複步驟 2 至 5，直到迭代次數達邊界移動法之使用代數為止。

在經過上述邊界移動法的步驟後，可使一開始均勻隨機分布在解空間中的粒子群，在很短的時間內直接移動到離該粒子最近的合理解與不理解的邊界上，有效的增加前期的收斂速度並減少不必要的運算。此外，由於邊界移動法是為了讓粒子於搜尋前期可以很快速的移動至合理解與不理解的邊界上，藉此降低粒子於搜尋前期時在不重要區域中的搜尋時間，因此當粒子群整體相當靠近收斂中心時，卻仍然使用此策略的話，則有可能會減緩甚至阻止粒子收斂至收斂中心。所以在這個考慮下，本人在此做了一個測試，其設定如下所示：

1. 以標準粒子群優化演算法為測試標準。
2. 使用 10 根平面桁架做為測試案例。
3. 使用邊界移動法的次數分別為不使用、1 次、5 次、10 次及 20 次等五種案例。
4. 每種案例皆個別獨立執行 30 次，取最常出現的收斂類型為該種案例的代表，藉此比較邊界移動法的最佳使用次數。

圖 3-5 及圖 3-6 所示即為標準粒子群優化演算法以 10 根平面桁架為測試案例，在邊界移動法使用不同次數下的收斂情形。由圖中可以看出粒子群優化演算法在加入邊界移動法後，其初始的適應值相對於沒有使用邊界移動法的粒子群優化演算法是好許多的，而隨後幾次迭代搜尋的收斂速度也會因此而有大幅的提升，讓粒子群在開始搜尋後沒多久的時間內，即找到一定程度的問題解。

在這個測試中，我們得知若於第一代搜尋時使用邊界移動法，可以有效的降低初始的適應值，而其中使用 5 次、10 次及 20 次邊界移動法的測試案例，則於第一代搜尋後仍然以很快的收斂速度在進行搜尋，並且獲得更好的問題解。然而，由於是以隨機亂數來產生初始位置的關係，所以在使用這三種次數限制做測試時，都有機率可以在很少的迭代次數中就獲得很好的問題解，但是在各自經過 30 次的獨立測試後，發現使用 20 次的邊界移動法時，平均上能有比較好的收斂結果，因此在本研究中，邊界移動法的使用代數則設定為只在演算法開始搜尋時的前 20 代為止。

3.2.3 粒子飛離法

粒子群優化演算法的一大特色為擁有很快的收斂速度，能在該更新機制下很快的引導粒子趨向各自及群體所經歷過最佳解的加權中心位置，但也因為擁有這樣快速的收斂速度，導致粒子群在收斂過程中容易忽略了一些重要資訊並且喪失問題解的多樣性。在文獻中也指出，粒子群優化演算法容易因為它的快速收斂，使得粒子群落入局部最佳解中，並因失去問題解的多樣性而無法跳脫【18】。

在許多關於粒子群優化演算法的文獻中，大部分的學者為了使粒子群能擁有跳脫局部最佳解的能力，而對標準粒子群優化演算法做了修改。在 1.2 節中提過，學者對於粒子群優化演算法的主要調整方法有二，第一種方法是針對(2-3)式下去做調整，亦即改變粒子的移動速度，增加一隨機亂數，讓粒子有機率能依靠這項亂數值來增加問題解的多樣性，使其能跳脫出局部最佳解；另一種方法則是將粒子群優化演算法結合其他最佳化演算法，截取演算法各自的長處來加以共同利用，藉此補強粒子群優化演算法的強健性，使其較不容易因陷入局部最佳解中而失去搜尋能力。

本研究中提出的另一個策略取名為「粒子飛離法」，其主要構想是針對粒子群於搜尋期間，因為快速收斂而失去了問題解的多樣性，同時也有可能錯過了全域最佳解而收斂至局部最佳解中。因此為了加強找到全域最佳解的機率，使用粒子飛離法將靠近收斂中心的粒子隨機拋離有限的距離，使得粒子在收斂中心與指定的搜尋半徑內均勻分布，藉此可避免粒子群因收斂過快的問題而喪失問題解的多樣性，導致粒子群困在局部最佳解中。

與其他文獻不同的是，其他文獻是針對粒子的速度下去做修正，而粒子飛離法的主要概念則是針對粒子的位置直接做修正，其示意圖如圖 3-7 所示且主要的步驟如下所示：

1. 設定一個基準空間，即執行粒子飛離法的空間。

粒子飛離法的概念是為了阻止粒子的收斂，亦即為了增加問題解的多樣性而想出一種策略。所以假如粒子尚位於離收斂中心很遠的地方，也就是問題解還具有多樣性時，就對粒子進行該策略，則會因為粒子還具有多樣性而失去了該策略的意義。因此需要以收斂中心為基準設定一個基準空間，使粒子飛進這個基準空間後才會有一固定機率啟動這個策略。在本研究中，對於一個 D 維的最佳化問題，其基準空間的設定如下：

$$BS_u = (BS_{u,1}, BS_{u,2}, \dots, BS_{u,D}) \quad (3-32)$$

$$BS_{u,i} = P_{g,i}^k + 0.1 \times (smax_i^k - P_{g,i}^k), \quad i = 1, 2, \dots, D$$

$$BS_l = (BS_{l,1}, BS_{l,2}, \dots, BS_{l,D}) \quad (3-33)$$

$$BS_{l,i} = P_{g,i}^k - 0.1 \times (P_{g,i}^k - smin_i^k), \quad i = 1, 2, \dots, D$$

其中：

BS_u ：基準空間的上限

BS_l ：基準空間的下限

$smax$ ：搜尋空間上限

$smin$ ：搜尋空間下限

P_g ：所有粒子所經歷過的最佳位置向量

k ：迭代數

2. 設定一個飛離空間，即粒子飛離時的最大空間。

粒子群在執行粒子飛離法時，是隨機飛到一個有限距離的地方，而這個有限距離地方的最遠處所包圍起來的空間，即為粒子的飛離空間。

由於希望粒子在搜尋過程中能在不失粒子多樣性的同時，也能集中在較可能出現全域最佳解的附近做搜尋，因此假如讓粒子一直飛離至完整的解空間中，雖然能夠增加粒子的多樣性，但卻也會造成粒子的分散而無法集中搜

尋可能性較大的區域，導致可能無法找到全域最佳解；又假如讓粒子一直飛離在一小區域中，則雖然能做集中搜尋，但卻可能會有多樣性不足的問題發生，因此也失去了粒子飛離法的用意，故需要設定一個適當的粒子飛離空間。在本研究中，對於一個 D 維的最佳化問題，且每一維搜尋空間的上下限分別為 $smax$ 及 $smin$ ，其飛離空間的設定如下：

$$FS = ([smin_1^k, smax_1^k], [smin_2^k, smax_2^k], \dots, [smin_D^k, smax_D^k]) \quad (3-34)$$

其中：

k ：迭代數

$smax$ ：搜尋空間上限

$smin$ ：搜尋空間下限

為了達到上述的目的，這個適當的飛離空間在本研究中則設定為一線性縮小的空間，然而飛離空間如果一直無限制的縮小，到最後粒子一樣會失去其多樣性，也就失去了使用粒子飛離法的意義，故需設定一個最小的限度來限制飛離空間的持續縮小，使其不會因飛離空間無限制的縮小而發生上述問題，因此在本研究中所使用限制條件則設定為當粒子群的迭代數達最大迭代數的 80% 時，飛離空間剛好為最初搜尋空間的 20%。另外由於飛離空間在縮小的同時也就意味著將捨棄一些搜尋空間，使粒子群能夠更集中的搜尋某個區塊，因此本人希望在縮小飛離空間前，能有一小段時間進行較完整的全域搜尋，使其能夠在進行局部搜尋之前，可以充分獲得相關的資訊，提升搜尋到最佳解的機率，故本研究中則設定粒子飛離法在執行之後，當粒子群的迭代數達最大迭代數的 20% 時，才開始執行飛離空間的縮小。其飛離空間縮小的情形如圖 3-8 所示，而縮小的關係式則如下所示：

$$smax_i^{k+1} = \begin{cases} smax_i^*, & P_{g,i}^k < (smax_i^k - smin_i^k)/2 \\ smax_i^k, & P_{g,i}^k \geq (smax_i^k - smin_i^k)/2 \end{cases} \quad (3-35)$$

$$smax_i^* = (0.8 \times (smax_i^1 - smin_i^1)) / (0.6 \times k_{max})$$

$$smin_i^{k+1} = \begin{cases} smin_i^k, & P_{g,i}^k < (smax_i^k - smin_i^k)/2 \\ smin_i^*, & P_{g,i}^k \geq (smax_i^k - smin_i^k)/2 \end{cases} \quad (3-36)$$

$$smin_i^* = (0.8 \times (smax_i^1 - smin_i^1)) / (0.6 \times k_{max})$$

3. 設定一個啟動機率，即粒子於基準空間中執行粒子飛離法的機率。

雖然設定了一基準空間，使得粒子飛進基準空間後才會執行粒子飛離法，但如果粒子一飛進基準空間後即一定執行粒子飛離法，則粒子會因一直飛離收斂中心而導致無法收斂，因此也一樣會有找不到全域最佳解的可能性。所以需設定一個啟動機率，讓粒子於基準空間內，可以以一定的機率來執行粒子飛離法。由於考慮到粒子的收斂速度很快，因此在本研究中這個啟動機率設定為 0.2，即 20%。

在經過上述粒子飛離法的步驟後，可使原先粒子群因過早收斂而喪失問題解的多樣性，並失去找到全域最佳解之可能性的問題加以改善。另外，由於粒子飛離法的加入，使得原先粒子群優化演算法的搜尋特性有了些不同的改變，即由全域搜尋轉為局部搜尋的特性，改變成由全域搜尋轉為半全域與局部共存的搜尋模式，藉此加強或改善粒子群的搜尋品質。

3.2.4 飛回機制(Fly back mechanism)

在含束制條件的最佳化問題中，會有另一個重要問題就是束制條件的邊界問題。在許多含束制條件之最佳化問題的文獻中，最常見也是目前最為廣泛用來處理邊界問題的方法就是懲罰函數(Penalty Function)法。

然而一些實驗結果表示，懲罰函數的結果會將位於不合理的空間裡的粒子重置到該粒子所經歷過的最佳位置，所以有時候反而會制止粒子搜尋至全域最佳解，因此懲罰函數的使用會降低粒子群優化演算法的搜尋效率【26】。為了改善粒子群優化演算法在使用懲罰函數時的問題，He 等人【26】於 2004 年提出了另一種處理邊界問題的方法，稱為「飛回機制(fly back mechanism)」。

如 3.2.1 節中所提到在結構的最佳化設計時，最佳解往往都在合理解與不合理的邊界上，然而不只是結構最佳化設計，許多最佳化的問題也都有此特性，因此粒子在搜尋最佳解的過程當中，很有可能因為速度過快衝過了合理解的邊界，使得問題解成為不合理的【26】。

粒子飛回機制的構想為當粒子在合理的空間飛行搜尋時，雖然有最大速度的限制，但依然會有因為速度過快而飛出合理解邊界的可能，然而一旦有粒子飛

出合理解邊界，即飛入不合理解區域時，粒子飛回機制將會強迫使粒子返回前一次的位置，以確保粒子仍然為合理解，而粒子在經由這個動作之後，於下一次的搜尋中又有機會可以到達或更靠近邊界，藉此提升搜尋到全域最佳解的機率。

一些實驗結果顯示，這種飛回機制對於處理含束制條件的最佳化問題，比起其他處理邊界問題的技術，更能在較少的迭代次數中取得較好的問題解，因此這種飛回機制很適合用來處理含束制條件之最佳化邊界的問題【26】。故本研究中亦引用了這種飛回機制來處理最佳化的邊界問題。

3.2.5 停止條件

在粒子群優化演算法這種需經過疊代搜尋後才能得到最佳解的最佳化演算法中，要啟動演算法的搜尋很容易，但要如何使其停止以及什麼時候停止卻是一個很重要部份，在這個部分中，常見的停止條件有下列幾種：

- i. 一固定之最大迭代數。
- ii. 連續某一定量之迭代數皆沒有得到更好的問題解。

然而在本研究中所使用的停止條件是為上述第一種方法，也就是當粒子群優化演算法啟動並開始迭代搜尋後，必須迭代至一個預設的最大迭代數後才會停止搜尋，在本研究中則設定此最大迭代數為 400 代。

3.2.6 改良式粒子群優化演算法流程

改良式粒子群優化演算法的演算程序，因為加入了兩個新的改良策略，因此與 2.2 節中所介紹的標準粒子群優化演算法的演算程序略有不同，改良式粒子群優化演算法的演算程序說明如下，其流程圖如圖 3-9 所示：

1. 給定相關設計資料：群體規模(M)、最大速度(V_{max})、最大迭代數(k_{max})、慣性權重(w)、學習因子(c_1 、 c_2)、目標函數、限制函數、結構系統的相關資料、邊界移動法的使用代數、粒子飛離法的基準空間及飛離空間。
2. 在解空間中隨機產生粒子群的初始位置 X_i^0 及初始速度 V_i^0 。
3. 對各粒子所在位置的解做結構矩陣分析，並求得該組斷面下各桿件應力及各節點位移，與其相對應之最大容許應力及最大容許位移的比例 a 、 b 後，即可使用邊界移動法將粒子瞬間移動至合理解與不合理的邊界上，

以加速粒子的收斂。

4. 根據目標函數及限制函數計算該位置的適應值 $f(X_i^0)$ ，且令 $pbest_i^0$ 為各粒子的初始位置 X_i^0 ，而 $gbest^0$ 則為所有 $pbest_i^0$ 中之最佳解。如(2-3)及(2-4)式所示。
5. 將各粒子所求出來的 P_i^k 及 P_g^k ，代入(2-1)式中來更新各粒子下一代的搜尋速度 V_i^{k+1} ，並利用此更新後的速度，代入(2-2)式以更新各粒子下一代的位置 X_i^{k+1} 。
6. 利用兩個條件判斷粒子更新後的位置 X_i^{k+1} 是否滿足粒子飛離法的執行條件，判斷條件如下所示：
 - i. 粒子的位置是否落於粒子飛離法的基準空間內。
 - ii. 是否達到啟動機率。若 X_i^{k+1} 滿足上述兩個條件後，則執行粒子飛離法讓粒子飛離至飛離空間中的任一隨機位置，並使這個位置為該粒子的 X_i^{k+1} 。
7. 對更新位置後的每個粒子，再進行結構矩陣分析，配合目標函數及限制函數計算 $k+1$ 代的適應值 $f(X_i^{k+1})$ 後，與各粒子自身最佳解的適應值 $f(P_i^k)$ 及所有粒子中最佳解的適應值 $f(P_g^k)$ 做比較，繼而更新 P_i^{k+1} 及 P_g^{k+1} ，如(2-5)及(2-6)式所示。
8. 若演算法滿足終止條件時(如：搜尋次數達最大迭代數)，則停止搜尋，否則需重複步驟3及5至8，直到滿足終止條件為止。

第四章 數值案例測試

在本研究中，經了文獻蒐集了數個桁架結構尺寸最佳化設計的經典案例，藉此測試改良式粒子群優化演算法的可行性，並驗證本研究中所提出兩種策略的效果。本文會在完成各案例之最佳化設計後，以圖示來說明其收斂的過程，並以表格方式來呈現各節點位移及各桿件應力，以證明最佳設計斷面為一組滿足束制條件的桿件斷面。最後再將最佳設計斷面與其所相對的整體重量，利用表格來與其他文獻中所找出的最佳設計斷面加以比較。然而，除了與其他文獻所找到的最佳設計斷面加以比較之外，亦會將本研究中所提出的改良式粒子群優化演算法，與本人所一同撰寫的標準粒子群優化演算法(PSO)及含被動聚集因子的粒子群優化演算法(PSOPC)做比較，並一併將搜尋時的收斂過程以圖示的方式來呈現，藉此驗證改良式粒子群優化演算法之效果。

在本研究中所提出的改良式粒子群優化演算法中，使用了 50 個粒子來做最佳化的動作，並讓每個粒子做了 400 次的迭代，亦即粒子群在搜尋過程中總共做了 20000 次的結構矩陣運算分析。另外，慣性權重 w 設為 0.9，學習因子 c_1 及 c_2 則設為 2。對於標準粒子群優化演算法的設定則一概相同，至於含被動聚集因子的粒子群優化演算法除了上述之設定相同之外，被動聚集因子 c_3 則設定為 0.6，如表 4-1 所示。以下所列即為本研究所使用及測試之數值案例：

例一：平面 10 根桿件桁架結構，如圖 4-1 所示(單載，載重情況一)。

例二：平面 10 根桿件桁架結構，如圖 4-1 所示(單載，載重情況二)。

例三：空間 25 根桿件桁架結構，如圖 4-2 所示(多載)。

例四：空間 72 根桿件桁架結構，如圖 4-3 所示(多載)。

例五：空間 120 根桿件桁架結構，如圖 4-4 所示(單載)。

4.1 例一：平面 10 根桿件桁架結構(單載，載重情況一)

本研究測試的第一個案例為如圖 4-1 所示的一平面 10 根桿件懸臂桁架。此案例於先前的文獻中已有許多學者將其做過最佳化設計，如 Li 等人【11】、Kaveh 等人【12、32】、Perez 等人【31】。故本人將此案例做為本研究中的第一個測試案例。此案例所使用的材料密度 $\rho = 0.1 \text{ lb/in}^3$ ，彈性係數 $E = 10000 \text{ ksi}$ ，該桁架

結構共有 10 根桿件及 6 個節點，其各桿件起始節點及終止節點之編號如表 4-2 所示，而所有桿件共分成 10 組設計斷面，即此案例為含 10 個變數之最佳化設計問題，各組設計斷面所包含的桿件如表 4-3 所示，且每一組設計斷面中桿件斷面積的最小值為 0.1 in^2 ，最大值為 35.0 in^2 。此案例屬單組載重受力之問題，其受力情形為 $P_1 = 100 \text{ kips}$ 且 $P_2 = 0$ ，即節點 2 及節點 4 各受一 100 kips 向下的垂直力，如表 4-4 所示，而各組設計斷面中桿件的最大容許應力為 $\pm 25.0 \text{ ksi}$ ，及各節點的最大容許位移 $\pm 2.0 \text{ in}$ (於水平與垂直方向)。

圖 4-5 為標準粒子群優化演算法、含被動聚集因子的粒子群優化演算法及本研究所提出的改良式粒子群優化演算法於此案例中粒子群的收斂結果。可以發現標準粒子群優化演算法及含被動聚集因子的粒子群優化演算法在收斂過程中，其收斂情形如圖 4-5 所示呈現階梯狀收斂，且由圖中可看出粒子群在搜尋時所擁有的兩個問題如下：

1. 粒子群於搜尋前期是一步一步慢慢的收斂，並收斂至某值後則會因暫時找不到較佳解而使粒子的收斂停滯，因此整體拉長了粒子群收斂至全域最佳解的時間。
2. 粒子群在找到下一個較佳解之前，會因落入局部最佳解中，而需經過一段較長時間的搜尋，才會有另一較佳解出現。此外，粒子群也需在一個較長的迭代次數下，才有可能找到全域最佳解或者近似全域最佳解的答案。

於圖 4-5 中可看到本研究中所提出的改良式粒子群優化演算法，相較於標準粒子群優化演算法及含被動聚集因子的粒子群優化演算法，不但在前期就能以極快的速度收斂至全域最佳解附近，也能在較少的迭代次數中，搜尋到更佳的問題解，如表 4-5 所示即為本案例中使用不同粒子群優化演算法所找到的最佳設計斷面。此外，為了驗證本研究中所提出的兩種策略能分別解決上述之問題，本人亦做了一個簡單的測試藉此比較，其設定如下：

1. 只使用邊界移動法的改良式粒子群優化演算法
2. 只使用粒子飛離法的改良式粒子群優化演算法
3. 同時使用邊界移動法及粒子飛離法的改良式粒子群優化演算法

利用上述三種改良的粒子群優化演算法來對此案例進行最佳化設計，其搜尋

的收斂結果如圖 4-6 所示。針對第一條曲線，即只使用邊界移動法的改良式粒子群優化演算法，可以看到的是該曲線於搜尋前期即以很快的速度收斂至全域最佳解附近，省了非常多粒子群在不重要區域的搜尋及運算時間，但是卻因為未改善第二個問題，所以在一開始收斂停滯後，則因一直沒有找到更佳的問題解而使得第一條曲線一直持平；針對第二條曲線，即只使用粒子飛離法的改良式粒子群優化演算法，可以看到的是整條曲線都在不斷的進步，因找不到更佳的問題解而產生的水平線也變得沒那麼明顯，甚至於搜尋後期也更能找到趨於全域最佳解的答案，但是卻因為未改善第一個問題，導致搜尋時依然是一步一步慢慢的靠近，所以整體上依舊是拉長了粒子群收斂至全域最佳解的時間。因此，假如能合併這兩種方法，如同第三條曲線般，則不但可以在很短的時間內就讓粒子收斂，且是將粒子收斂至全域最佳解。

此案例在經過本研究所提出的方法來進行最佳化設計後，得到的最佳設計斷面為 30.447、0.100、23.596、15.029、0.100、0.566、7.420、20.988、21.524、0.100，單位為 in²，在此組斷面下的桁架總重為 5061.36 lb。將此組最佳設計斷面經結構分析之後可得到其各節點位移如表 4-6 所示，其中最大位移出現在第 1 個節點的 Y 方向上，該值為 -2.000 in，剛好符合最大容許位移 ± 2.0 in；而表 4-7 為各桿件應力，其中最大應力出現在第 5 根桿件，該值為 24.9961 ksi，也小於最大容許拉應力 25.0 ksi，故此組設計斷面為可行解。表 4-8 為最佳設計斷面比較表，將一些有做過此案例之文獻值與本研究中提出之改良式粒子群優化演算法所得到的最佳設計斷面做比較。在表中可看到 Li 等人【11】利用 HPSO、Perez 等人【31】利用 PSO 及 Kaveh 等人【12、32】利用 PSACO 與 HPSACO 所得到最佳設計斷面下的結構總重分別為 5060.92 lb、5024.21 lb、5057.36 lb 及 5056.56 lb，雖然比本研究中所得到的 5061.36 lb 來的小，但本人將此四組最佳設計斷面拿來做結構矩陣分析後，發現此案例在這四組最佳設計斷面下，若將節點位移或桿件應力的精度取至小數點下 2 至 6 位數時，其結構中的最大節點位移或最大桿件應力則會不滿足最大容許位移或最大容許應力之限制。另外在圖 4-5 中可以發現，在相同初始位置的情況下，改良式粒子群優化演算法的搜尋結果較另外兩種最佳化演算法來的好很多，約在經過 245 次的迭代(即 12,250 次的結構矩陣分析)後即得到最佳解，而標準粒子群優化演算法與含被動聚集因子的粒子群優化演算法則在經過

400 次的迭代之後，卻尚未獲得一較好之最佳解，由此可以看出本研究中所提出的方法在擁有高精度的情況下，同時也能擁有非常好的收斂速度。

4.2 例二：平面 10 根桿件桁架結構(單載，載重情況二)

此案例與例一只有整個結構的受力不同，其他如桁架結構配置、材料性質、桿件與節點編號、最大容許應力及最大容許位移等皆完全相同，也有一些學者對此案例做過最佳化設計，如 Li 等人【11】、Kaveh 等人【12,32】。此案例之受力情形為 $P_1 = 150$ kips 且 $P_2 = 50$ kips，即節點 1 及節點 3 各受一 50 kips 向上的垂直力，節點 2 及節點 4 各受一 150 kips 向下的垂直力，如表 4-9 所示。

圖 4-7 所示為標準、含被動聚集因子以及改良式等三種不同的粒子群優化演算法於搜尋過程中的收斂結果，可以看到與圖 4-5 有著相同的結果，表 4-10 即為三種不同粒子群優化演算法所找到的最佳設計斷面。而此案例在經過本研究所提出的方法來進行最佳化設計後，得到的最佳設計斷面為 24.1567、0.1141、24.7967、14.4889、0.1000、1.9742、12.4771、12.7048、20.2231、0.1010，單位為 in^2 ，在此組斷面下的桁架總重為 4679.4847 lb。將此組最佳設計斷面經結構分析之後可得到其各節點位移如表 4-11 所示，其中最大位移出現在第 2 個節點的 Y 方向上，該值為 -2.000 in，剛好符合最大容許位移 ± 2.0 in；而表 4-12 為各桿件應力，其中最大的應力出現在第 5 根桿件，該值為 25.0 ksi，亦剛好為最大容許拉應力 25.0 ksi，故此組設計斷面為可行解。而表 4-13 即為最佳設計斷面比較表，為一些此案例之文獻值與本研究中所求得最佳設計斷面來做比較。在表中可看到，Li 等人【11】利用 PSOPC 與 HPSO 及 Kaveh 等人【12,32】利用 PSACO 與 HPSACO 所得到最佳設計斷面下的結構總重分別為 4677.70 lb、4677.29 lb、4676.05 lb 及 4675.78 lb，雖然比本研究所得到的 4679.4847 lb 來的小，但本人將此四組最佳設計斷面拿來做結構矩陣分析後，發現此案例在這四組最佳設計斷面下，若將節點位移或桿件應力的精度取至小數點下 3 至 4 位數時，其結構中的最大節點位移或最大桿件應力則會不滿足最大容許位移或最大容許應力之限制。另外在圖 4-7 中也可以發現，改良式粒子群優化演算法約在經過 95 次的迭代(即 4,750 次的結構矩陣分析)後即得到最佳解，而標準粒子群優化演算法與含被動聚集因子的粒子群優化演算法則在經過 400 次的迭代之後，亦尚未搜尋到一較好之最佳解，因

此也可看再次出本研究中所提出的方法於最佳化設計中的搜尋效能。

4.3 例三：空間 25 根桿件桁架結構(多載)

此案例如圖 4-2 所示為一空間 25 根桿件的桁架結構，共有 10 個節點。有許多學者也對此案例做過最佳化設計，如 Li 等人【11】、Kaveh 等人【12、32】。此案例所使用的材料密度 $\rho = 0.1 \text{ lb/in}^3$ ，彈性係數 $E = 10000 \text{ ksi}$ ，其各桿件起始節點及終止節點之編號如表 4-14 所示，而所有桿件共分成 8 組設計斷面，即此案例為含 8 個變數之最佳化設計問題，各組設計斷面所包含的桿件如表 4-15 所示，且每一組設計斷面中桿件斷面積的最小值為 0.01 in^2 ，最大值為 3.4 in^2 。不同於平面 10 根桿件桁架結構，此案例屬多組載重受力之問題，需滿足兩組載重受力，第一組載重組合為節點 1 於 Y 方向受 20 kips 及 Z 方向受 -5 kips 與節點 2 於 Y 方向受 -20 kips 及 Z 方向受 -5 kips，第二組載重組合為節點 1 於 X 方向受 1 kips、Y 方向受 10 kips 及 Z 方向受 -5 kips；節點 2 於 Y 方向受 10 kips 及 Z 方向受 -5 kips；節點 3 及節點 6 皆於 X 方向受 5 kips，如表 4-16 所示，而各組設計斷面中桿件的最大容許應力如表 4-17 所示，各節點的最大容許位移則為 $\pm 0.35 \text{ in}$ (於 X、Y 及 Z 方向)。

圖 4-8 所示為標準、含被動聚集因子以及改良式等三種不同的粒子群優化演算法於搜尋過程中的收斂結果，而表 4-18 即為三種不同粒子群優化演算法所找到的最佳設計斷面。此案例在經過本研究所提出的方法來進行最佳化設計後，得到的最佳設計斷面為 0.0102、1.9895、2.9893、0.0100、0.0100、0.6865、1.6766、2.6602，單位為 in^2 ，在此組斷面下的桁架總重為 545.1648 lb。將此組最佳設計斷面經結構分析之後，所得到各節點位移如表 4-19 所示，其中第一組載重組合下的最大位移發生在節點 1 及節點 2 的 Y 方向上，該值分別為 0.3500 及 -0.3500 in，剛好符合最大容許位移 $\pm 0.35 \text{ in}$ ；第二組載重組合下的最大位移亦發生相同位置上，其值皆為 0.3499 in，亦小於最大容許位移 $\pm 0.35 \text{ in}$ ，因此兩組載重下的最大位移皆符合最大容許位移之限制；而表 4-20 為各桿件應力，比較各桿件所受到的應力與各桿件所屬設計斷面之最大容許應力後，可以發現最接近最大容許應力的桿件為第一組載重下的第 18 及 21 根桿件，應力皆為 -6.9588 ksi，此兩根桿件皆所屬為第 7 個設計斷面，該設計斷面之最大容許壓應力為 6.959，因此所受應

力皆小於最大容許應力，故此組設計斷面為可行解。另外亦將一些學者對於此案例所做的最佳化結果，與本研究中所求得之最佳斷面值一同比較，表 4-21 即為最佳設計斷面比較表。在表中可看到，Kaveh 等人【12,32】利用 PSACO 與 HPSACO 所得到最佳設計斷面下的結構總重分別為 545.04 lb 及 544.99 lb，雖然比本研究中所得到的 545.1648 lb 來的小，但本人將此兩組最佳設計斷面拿來做結構矩陣分析後，發現此案例在這兩組最佳設計斷面下，若將節點位移或桿件應力的精度取至小數點下 3 位數時，其結構中的最大節點位移或最大桿件應力則會不滿足最大容許位移或最大容許應力之限制。另外在圖 4-8 中也可以發現，改良式粒子群優化演算法約在經過 30 次的迭代(即 1,500 次的結構矩陣分析)後即得到最佳解，而標準粒子群優化演算法與含被動聚集因子的粒子群優化演算法則在經過 400 次的迭代之後所得到的最佳解，亦與改良式粒子群優化演算法所得到的最佳解有些差距，且在表 4-21 中也可發現在滿足最大容許位移及最大容許應力的限制下，此案例中所找到的最佳設計斷面其總重量亦比表中文獻值來的更進步一些，由此也可再次證明改良式粒子群優化演算法的搜尋效能。

4.4 例四：空間 72 根桿件桁架結構(多載)

此案例如圖 4-3 所示為一空間 72 根桿件的桁架結構，共有 20 個節點。一些學者如 Li 等人【11】、Perez 等人【31】，也對此案例做過最佳化設計。此案例所使用的材料密度 $\rho = 0.1 \text{ lb/in}^3$ ，彈性係數 $E = 10000 \text{ ksi}$ ，其各桿件起始節點及終止節點之編號如表 4-22 所示，而所有桿件共分成 16 組設計斷面，即此案例為含 16 個變數之最佳化設計問題，各組設計斷面所包含的桿件如表 4-23 所示，且每一組設計斷面中桿件斷面積的最小值為 0.1 in^2 ，最大值為 4.0 in^2 。此案例與例三同屬於多組載重之問題，需滿足兩組載重受力如表 4-24 所示，而各組設計斷面中桿件的最大容許應力皆為 $\pm 25 \text{ ksi}$ ，各節點的最大容許位移則為 $\pm 0.25 \text{ in}$ (於 X、Y 及 Z 方向)。

圖 4-9 所示為標準、含被動聚集因子以及改良式等三種不同的粒子群優化演算法於搜尋過程中的收斂結果，而表 4-25 即為三種不同粒子群優化演算法所找到的最佳設計斷面。此案例在經過本研究所提出的方法來進行最佳化設計後，得到的最佳設計斷面為 1.8428、0.4999、0.1039、0.1000、1.2211、0.5486、0.1000、

0.1000、0.4897、0.496、0.1033、0.1502、0.1558、0.5747、0.4331、0.5219，單位為 in^2 ，在此組斷面下的桁架總重為 381.6105 lb。將此組最佳設計斷面經結構分析之後，在兩組載重下所得到各節點位移分別如表 4-26 及表 4-27 所示。其中第一組載重組合下的最大位移發生在節點 17 的 X 方向及 Y 方向上，其值皆為 0.25 in，剛好符合最大容許位移 ± 0.25 in 的限制；第二組載重組合下的最大位移發生在節點 17 至節點 20 等四個節點的 Z 方向上，其值皆為 -0.25 in，亦符合最大容許位移 ± 0.25 in 的限制。而表 4-28 及表 4-29 則分別為在兩組載重組合下的各桿件應力，其中第一組載重組合下的最大應力發生在第 55 根桿件上，其值為 -16.1115 ksi，小於最大容許壓應力 25 ksi；而第二組載重組合下的最大應力則發生在第 55 根至第 58 根等四根桿件上，其值皆為 -24.6374 ksi，亦小於最大容許壓應力 25 ksi，故此組設計斷面為可行解。表 4-30 為最佳設計斷面比較表，將一些有做過此案例之文獻值與本研究中提出之改良式粒子群優化演算法所得到的最佳設計斷面做比較。在表中可看到 Li 等人【11】利用 PSOPC 及 HPSO 所得到最佳設計斷面下的結構總重皆為 369.65 lb，雖然比本研究中所得到的 381.6105 lb 來的小，但本人將此兩組最佳設計斷面拿來做結構矩陣分析後，發現此案例在 HPSO 所找到的最佳設計斷面下，若將節點位移的精度取至小數點下 6 位數時，其結構中的最大節點位移則會不滿足最大容許位移之限制；且這兩組最佳設計斷面在表 4-24 中第二組載重下的最大節點位移及最大桿件應力，皆早已超過節點的最大容許位移 0.25 in 及桿件的最大容許應力 25 ksi。另外在圖 4-9 中也可以發現，改良式粒子群優化演算法約在經過 345 次的迭代(即 17,250 次的結構矩陣分析)後即得到最佳解，而標準粒子群優化演算法與含被動聚集因子的粒子群優化演算法則在經過 400 次的迭代之後所得到的最佳解，亦與改良式粒子群優化演算法所得到的最佳解有些差距，且在表 4-30 中也可發現在滿足最大容許位移及最大容許應力的限制下，此案例中所找到的最佳設計斷面其總重量亦比表中文獻值來的更進步一些，由此也可再次證明改良式粒子群優化演算法的搜尋效能。

4.5 例五：空間 120 根桿件桁架結構(單載)

此案例如圖 4-4 所示為一空間 120 根桿件的桁架結構，共有 49 個節點。一些學者如 Kaveh 等人【12,33】等，也對此案例做過最佳化設計。此案例所使用

的材料密度 $\rho = 0.288 \text{ lb/in}^3$ ，彈性係數 $E = 30450 \text{ ksi}$ 以及鋼材的降伏應力 $F_y = 58 \text{ ksi}$ ，其各桿件起始節點及終止節點之編號如表 4-31 所示，而所有桿件共分成 7 組設計斷面，即此案例為含 7 個變數之最佳化設計問題，各組設計斷面所包含的桿件如表 4-32 所示，且每一組設計斷面中桿件斷面積的最小值為 0.775 in^2 ，最大值為 20.0 in^2 。此案例屬單組載重之問題，受力情形為 -13.49 kips 作用在節點 1、 -6.744 kips 作用在節點 2 至節點 13 以及 -2.248 kips 作用在節點 14 至節點 37，且作用力的方向皆在結構的垂直方向，即作用在 Z 軸上如表 4-33 所示，其各節點的最大容許位移則為 $\pm 0.1969 \text{ in}$ (於 X、Y 及 Z 方向)。

另外，此案例中桿件的最大容許應力與前幾個案例略有不同。在前面的案例中，桿件的最大容許應力皆為一個定值，不會因為設計斷面的改變而造成最大容許應力的改變；在此案例中，桿件的最大容許拉應力及最大容許壓應力則分別如下：

$$\begin{cases} \sigma_i^+ = 0.6F_y, & \text{for } \sigma_i \geq 0 \\ \sigma_i^-, & \text{for } \sigma_i < 0 \end{cases} \quad (4-1)$$

而

$$\sigma_i^- = \begin{cases} \left[\left(1 - \frac{\lambda_i^2}{2C_c^2} \right) F_y \right] / \left(\frac{5}{3} + \frac{\lambda_i}{8C_c} - \frac{\lambda_i^3}{8C_c^3} \right), & \text{for } \lambda_i < C_c \\ \frac{12\pi^2 E}{23\lambda_i^2}, & \text{for } \lambda_i \geq C_c \end{cases} \quad (4-2)$$

$$C_c = \sqrt{(2\pi^2 E / F_y)}$$

$$\lambda_i = kL_i / r_i$$

$$r_i = aA_i^b$$

$$a = 0.4993$$

$$b = 0.6777$$

其中：

E：桿件材料的楊式係數

F_y ：鋼材的降伏應力

L_i ：第 i 根桿件的桿件長度

在(4-2)式中， r_i 為第 i 根桿件的迴轉半徑。迴轉半徑的原始計算公式為 $r = \sqrt{I/A}$ ，因為需要計算到慣性矩，所以會增加一些計算上的負擔也使計算過程較為麻煩。因此有學者【34】找出了迴轉半徑與桿件斷面積的另一個關係為 $r_i = aA_i^b$ ，其中 a 及 b 在不同斷面如管狀、L 型或 T 型斷面下，皆為不同之常數，在此案例中可將桿件視為管狀斷面，因此 a 及 b 則分別為 0.4993 及 0.6777【12、34】。在此案例中，將鋼材的降伏強度 $F_y = 58$ ksi 帶入(4-1)式中，可算出最大容許拉應力為一定值，該值為 $0.6*58 = 34.8$ ksi。

圖 4-10 所示為標準、含被動聚集因子以及改良式等三種不同的粒子群優化演算法於搜尋過程中的收斂結果，而表 4-34 即為三種不同粒子群優化演算法所找到的最佳設計斷面。此案例在經過本研究所提出的方法來進行最佳化設計後，得到的最佳設計斷面為 3.3436、2.7146、3.8777、2.5662、1.2460、3.4320、2.8302，單位為 in^2 ，在此組斷面下的桁架總重為 20122.9726 lb。將此組最佳設計斷面經結構分析之後可得到其各節點位移如表 4-35 所示，而各設計斷面的最大容許應力及各桿件所受應力則分別為表 4-36 及表 4-37 所示。

此外，本人在進行此案例的最佳化搜尋後，發現了一些與前面數個案例略為不同的情況發生。此案例在文獻【12】中於同時滿足位移條件(X、Y 及 Z 方向)與應力條件的最佳設計斷面下，其結構總重為 33248.9 lb，與本研究中在滿足相同的位移及應力條件下，所找到最佳設計斷面的總重 20122.9726 lb 有一定的差距，在正常的情況及考慮下，加上此案例已被一些學者【12、32、33、35、36】做過最佳化設計，其最佳化後的總重量也都在 33000 lb 以上，因此應該不太可能其他學者一致的都找不到更好的答案而落在這個答案附近，卻只有本人找到更輕的最佳設計斷面，而進步如此的多以至於造成這麼大的差距。

為了檢查究竟是哪裡出了問題，本人先反覆多次的比對文獻上所給予的材料性質與基本結構資料如材料密度、彈性係數、鋼材的降伏強度、節點座標、節點編號、桿件編號、設計斷面所包含之桿件與外力載重設定等，以及束制條件如設計斷面的斷面積上下限、節點最大容許位移與桿件最大容許應力等所有資料，待確認無誤後，接著逐步檢查本人所撰寫的演算法，從讀取結構資料及束制條件資料、計算最大容許應力與適應度函數值、結構矩陣運算、檢查節點位移與節點

最大容許位移及桿件應力與桿件最大容許應力之關係、邊界移動法與粒子飛離法等兩個新的策略，以及其它一些零零總總的小地方。在這樣反覆檢查之後，可以確定本人在此案例中的所有設定，皆符合文獻中所提供一切設定，且沒有發現本人所撰寫的演算法中有任何錯誤的地方。後來與指導教授及學長們的討論之後，為了更進一步的想要確認本人所撰寫的演算法是否真的無誤，即利用了一套常用的結構分析套裝軟體 SAP2000 V.11 來建立了一個此案例的桁架結構模型，並將上述所說的材料性質、基本結構資料及本人於最佳化搜尋後所找到的最佳設計斷面一同輸入至此模型中，進行三維桁架結構分析，並於結構分析後，將各節點的位移及各桿件的應力輸出如表 4-38 及表 4-39 所示，與本人所撰寫的演算法於最佳設計斷面下所求得的節點位移及桿件應力來做比較，藉此驗證本人所撰寫之演算法是否有錯誤的地方，並驗證所找到的最佳設計斷面是否滿足束制條件之限制。

由表 4-35 及表 4-37 對照表 4-38 及表 4-39，可以發現在本人所找到之最佳設計斷面下，利用本人所撰寫之演算法所得到的節點位移與桿件應力，與藉由 SAP2000 結構分析後所得到的節點位移與桿件應力幾乎一樣，也就是可以藉此驗證本人所撰寫之演算法是正確的，並沒有邏輯上或撰寫上的錯誤。另外，在表 4-35 或表 4-38 中，可以看到節點的最大位移出現在節點 5 及節點 11 的 Y 方向上且值皆為 -0.1967 in，剛好小於最大容許位移 ± 0.1969 in；應力方面將表 4-37 或表 4-39 配合著表 4-36 來對照，可以發現應力值最接近最大容許應力值的桿件為第 38 根與第 59 根，這兩根桿件的應力皆為 -2.2504 ksi，剛好小於該兩根桿件的最大容許壓應力 2.2514 ksi，因此此組設計斷面為一可行解。由於此案例發生了如上述不同於前面數個案例的問題，因此就不將其它學者所得到的最佳化設計斷面放進本研究中並做比較。

第五章 結論與建議

本論文提出一個改良式的粒子群優化演算法，可以有效的解決桁架結構斷面尺寸的最佳化問題。並已經過了數個文獻中常見的經典案例來證實，本文所提出的改良式粒子群優化演算法與其它以粒子群優化演算法為基礎的最佳化演算法所設計之結果相比，其搜尋的效能有很大的進步。以下則為本論文研究的結論、建議及展望。

5.1 結論

1. 相較於其它以粒子群優化演算法為基礎的最佳化演算法，本文中所提出的改良式粒子群優化演算法能在較少量的迭代次數及結構分析次數中完成最佳化設計，相較之下是非常有效率的。
2. 邊界移動法的使用可使改良式粒子群優化演算法在桁架結構最佳化設計中，於搜尋開始時就能把粒子從隨機初始位置，迅速的移動至鄰近的重要區域內，使粒子能有效的提升搜尋前期的收斂速度。
3. 邊界移動法的加入只是加速粒子群由隨機初始位置移動至鄰近重要區域的收斂速度，使粒子群不用浪費多餘的時間及不必要的運算於不重要區域內做搜尋，藉此降低粒子群整體的收斂時間，因此並不受搜尋起始值之影響。
4. 在粒子群優化演算法的搜尋過程中，粒子群會因為逐漸收斂至收斂中心而使粒子速度趨緩，進而降低粒子全域搜尋的能力，隨之失去問題解的多樣性而可能造成收斂至局部最佳解中，因此粒子飛離法的使用可以延長粒子全域搜尋狀態的時間，使其對問題解能於較長時間中擁有較高的多樣性，提升搜尋到全域最佳解的機率。
5. 當粒子群搜尋一段時間而轉變為局部搜尋時，粒子群會由全域探索轉變為集中挖掘，雖然會陸續的找到更佳解，但卻因越來越靠近收斂中心而使得粒子速度趨緩，導致每當找到一個更佳解後，都需花費更多的時間才能再找到下

一個更佳解，因此拉長整體的收斂時間，在加入粒子飛離法之後，粒子於集中挖掘階段依然會有一個小範圍的飛離空間，讓部分粒子在挖掘的同時，也有部分的粒子在進行粒子飛離，使得粒子因速度趨緩而導致搜尋更佳解之時間變長的問題得到改善。

6. 在幾個案例進行最佳化設計後的結果中，例一及例二雖然沒有找到較優的桁架斷面尺寸，但相較於過去文獻所得到的結果，相對的誤差都可以在 0.01% 以內，因此客觀上是可以被接受的；而例三及例四則在本文所提出方法下得到了較文獻值好的結果，因此可以證明此方法的效能及可行性。
7. 在本文中的例五出現了其他案例沒有出現的問題，即最佳化後的結果與文獻值差了許多，而在反覆檢查之中並沒有發現任何錯誤的地方，同時也使用了許多人常用的結構分析軟體 SAP2000 來建構該結構模型，並且將最佳化後得到的最佳設計斷面輸入後分析，其結果亦與本人所撰寫之演算法進行分析後得到的結果一模一樣，因此可以佐證本人所撰寫之演算法的正確性，也可確定在文獻中給定的資料情況下，本人所找到的最佳化設計斷面是可行的。

5.2 建議與展望

1. 在本研究中，針對邊界移動法的使用次數做了個測試，證明邊界移動法除了在第一代中的使用是有很大的效果，而不同的使用次數如 5、10 及 20 次等，也都可以看出在前期的貢獻。然而測試期間發現在這三種次數限制中，都有機率可以得到很好的問題解，只是若以平均值來看的話，使用 20 次邊界移動法下能有較好的收斂結果，因此本文建議邊界移動法使用次數的範圍可在 5 至 20 次之間。
2. 由於本文中邊界移動法的使用次數只在 0 至 20 次之間且只使用一個案例來做測試，因此建議其它學者可以嘗試利用不同的使用次數以及不同案例來測試邊界移動法的最佳使用次數或範圍。
3. 在本文中所使用的粒子飛離法並沒有特別提出討論，只有基於這個概念下做

了一些試誤的測試而決定在本文中所使用的數值，因此建議其他學者可以針對粒子飛離法的一些相關設定做較詳細的探討。

4. 由於邊界移動法與粒子飛離法的概念分別是為促使粒子收斂及阻止粒子收斂的兩種方法，因性質不太相同故一起使用的意義較不大，所以在本研究中粒子飛離法的使用是於邊界移動法停止後才開始執行，因此建議此兩種方法可以接續使用而不要同時使用。
5. 在例五中所發生的問題，由於在多方面已反覆檢查並確認無誤，故本人懷疑可能是文獻中所給的相關結構資料及設定，或者束制條件等有誤，因此建議其他學者若要對此案例進行最佳化設計，可以再多方面的檢查或者比對其他文獻中關於此案例的相關設定，以免發生與本文中相同的問題，而造成無法比較的情況。



參考文獻

- 【1】 郭信川，張建仁，劉清祥，「粒子群演算法於最佳化問題之研究」，第一屆台灣作業研究學會學術研討會暨 2004 年科技與管理學術研討會，419~432 頁，民國九十三年。
- 【2】 鄭博育，「啟發式桁架斷面尺寸最佳化設計」，國立交通大學，碩士論文，民國九十四年。
- 【3】 郭信川，官佳慶，「隨機搜尋法於多極值最佳化問題之應用」，中國造船暨輪機工程師學刊，第十九卷，第四期，頁 33~40，民國八十九年。
- 【4】 顏上堯，李旺蒼，施佑林，「路徑基礎類粒子群最佳化演算法於求解含凹形節線成本最小成本轉運問題之研究」，運輸計劃季刊，第三十六卷，第三期，393~424 頁，民國九十六年九月。
- 【5】 Y. Shi, R.C. Eberhart, "A modified particle swarm optimizer", Proceedings of IEEE International Conference on Evolutionary Computation, Anchorage, Alaska, 1998.
- 【6】 R.C. Eberhart, Y. Shi, "Particle Swarm Optimization Developments, Applications and Resources", In: Proceedings of IEEE International Conference on Evolutionary Computation, pp.81-86, 2001.
- 【7】 P. N. Suganthan, "Particle Swarm Optimizer with Neighbourhood Operator", Proc. Congress on Evolutionary Computation, pp.1958-1962, 1999.
- 【8】 Y. Shi, R.C. Eberhart, "Empirical study of particle swarm optimization", Proceedings of the Evolutionary Computation 1999 Congress, 3, pp.1945-1950, 1999.
- 【9】 S He, QH Wu, JY Wen, JR Saunders, RC Paton, "A particle swarm optimizer with passive congregation", Biosystem, 78, pp.135-47, 2004.
- 【10】 PS Shelokar, P Siarry, VK Jayaraman, BD Kulkarni, "Particle swarm and ant colony algorithms hybridized for improved continuous optimization", Applied Mathematics and Computation, 188, pp.129-42, 2007.
- 【11】 LJ Li, ZB Huang, F Liu, QH Wu, "A heuristic particle swarm optimizer for optimization of pin connected structures", Computers and Structures, 85, pp.340-9, 2007.
- 【12】 A Kaveh, S Talatahari, "Particle swarm optimizer, ant colony strategy and harmony search scheme hybridized for optimization of truss structures", Computers and Structures, 87, pp.267-83, 2009.

- 【13】 R.C. Eberhart, J. Kennedy, “Particle Swarm Optimization”, Proceedings of IEEE International Conference on Neural Networks, IV, pp.1942-1948, 1995.
- 【14】 R.C. Eberhart, J. Kennedy, “A New Optimizer Using Particle Swarm Theory”, Proceedings of the Sixth International Symposium on Micro Machine and Human Science, IEEE Service Center, Piscataway, NJ, Nagoya, Japan, pp.39-43, 1995.
- 【15】 R.C. Eberhart, J. Kennedy, Y. Shi, “Swarm Intelligence”, Evolutionary Computation, Morgan Kaufmann, Los Altos, CA, 2001.
- 【16】 李維平，黃郁授，戴彰廷，「自適應慣性權重改良粒子群演算法之研究」，中原大學資訊管理所。
- 【17】 R.C. Eberhart, Y. Shi, “Comparison between genetic algorithms and particle swarm optimization”, Annual Conference on Evolutionary Programming, San Diego, CA, 1998.
- 【18】 李維平，王雅賢，江正文，「粒子群最佳化演算法改良之研究」，中原大學資訊管理研究所。
- 【19】 林大為，「結合模擬退火之改良粒子群演算法於結構最佳化設計的研究」，國立中央大學，碩士論文，民國九十八年。
- 【20】 莊玟珊，「PSO-SA 混合搜尋法與其他結構最佳化設計之應用」，國立中央大學，碩士論文，民國九十六年。
- 【21】 Y. Shi, R.C. Eberhart, “Parameter Selection in Particle Swarm Optimization”, V. W. Porto, N. Saravanan, D. Waagen, and A. E. Eiben (eds), Lecture Notes in Computer Science, 1447, Evolutionary Programming VII, Springer, Berlin, pp. 591–600, 1998.
- 【22】 M. Clerc, “The Swarm and the Queen: Towards a Deterministic and Adaptive Particle Swarm Optimization”, Proceedings of the Congress on Evolutionary Computation, 3, pp.1951–1957, 1999.
- 【23】 R.C. Eberhart, Y. Shi, “Comparing Inertia Weights and Constriction Factors in Particle Swarm Optimization”, Proceedings of the 2000 Congress on Evolutionary Computation, 1, pp.84–88, 2000.
- 【24】 J.K. Parrish, W.M. Hamner, “Animal Groups in Three Dimensions”, Cambridge University Press, Cambridge, UK, 1997.
- 【25】 P.J. Angeline, “Evolutionary optimization versus particle swarm optimization: philosophy and performance difference”, in: V.W. Porto et al. (Eds.), Proceedings of 7th Annual Conference on Evolutionary Programming, Lecture Notes in Computer Science, vol. 1447, Springer, Berlin, pp.601–610, 1998.
- 【26】 S He, E Prempan, QH Wu. “An improved particle swarm optimizer for mechanical design optimization problems”, Eng Optimiz, 36, 5, pp.585–605, 2004.

- 【27】 L Davis, “Genetic algorithms and simulated annealing”, London: Pitman, 1987.
- 【28】 Le Riche RG, Knopf-Lenoir C, Haftka RT. “A segregated genetic algorithm for constrained structural optimization”, In: Sixth International Conference on Genetic Algorithms, University of Pittsburgh: Morgan Kaufman, pp.558–65, 1995.
- 【29】 A. G. M. Michell, “The Limits of Economy of Material in Framed Structures”, Philosophical Magazine Series 6, 8, 47, pp.589-597, 1904.
- 【30】 L.A. Schmit, “Structural Design by Systematic Synthesis”, Proceedings of the 2nd Conference on Electronic Computation, ASCE, New York, pp.105-122 1960.
- 【31】 R.E. Perez, K. Behdinan, “Particle swarm approach for structural design optimization”. Computers and Structures, 85, pp.1579–88, 2007.
- 【32】 A. Kaveh, S. Talatahari, “A hybrid particle swarm and ant colony optimization for design of truss structures”, Asian J Civil Eng, 9, 4, pp.329–48, 2008.
- 【33】 A. Kaveh, S. Talatahari, “Size optimization of space trusses using Big Bang–Big Crunch algorithm”, Computers and Structures, 87, 17–18, pp.1129–1140, 2009.
- 【34】 M.P. Saka, “Optimum design of pin-jointed steel structures with practical applications”, J Struct Eng ASCE, 116, 10, pp.2599–620, 1990.
- 【35】 O. Keleşoğlu, M. Ülker, “Fuzzy optimization geometrical nonlinear space truss design”, Turkish Journal of Engineering and Environmental Science, 29, pp.321-9, 2005.
- 【36】 A Kaveh, B Farahmand Azar, S Talatahari, “Ant colony optimization for design of space trusses”. Int J Space Struct, 23, 3, pp.167–81, 2008.

附表

表 4-1 粒子群優化演算法於本研究中的參數設定

項目	符號	設定值
群體規模(population)	M	50
最大迭代數(generations)	k_{max}	400
慣性權重(inertia weight)	w	0.9
學習因子(acceleration constants)	c_1 、 c_2	2
被動聚集因子 (passive congregation)	c_3	0.6

表 4-2 例一，各桿件起始節點及終止節點之編號

桿件編號	起始節點編號	終止節點編號
1	5	3
2	3	1
3	6	4
4	4	2
5	4	3
6	2	1
7	5	4
8	6	3
9	3	2
10	4	1

表 4-3 例一，各組設計斷面中所包含之桿件編號

群組編號	所包含之桿件編號
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	10

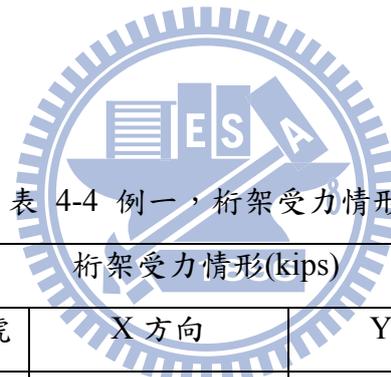


表 4-4 例一，桁架受力情形

桁架受力情形(kips)		
節點編號	X 方向	Y 方向
1	0.0	0.0
2	0.0	-100.0
3	0.0	0.0
4	0.0	-100.0

表 4-5 例一，不同粒子群優化演算法所找到的最佳設計斷面

Area (in ²)	PSO	PSOPC	Improved PSO
A ₁	29.2002	34.5899	30.447
A ₂	7.7670	0.1660	0.100
A ₃	34.9995	34.9999	23.596
A ₄	7.2611	13.2248	15.029
A ₅	0.1003	0.1031	0.100
A ₆	14.9028	4.5191	0.566
A ₇	26.4979	6.2689	7.420
A ₈	11.8001	15.0432	20.988
A ₉	9.0385	25.4301	21.524
A ₁₀	10.7008	0.3675	0.100
Weight (lb)	6347.0850	5552.1330	5061.36
結構矩陣 分析次數	20,000	20,000	20,000

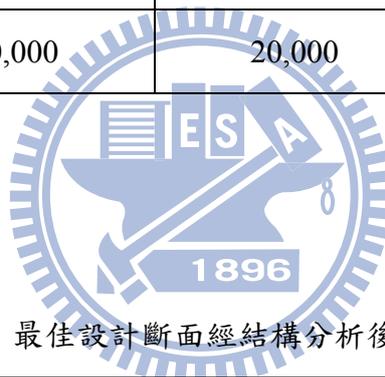


表 4-6 例一，最佳設計斷面經結構分析後之各節點位移

各節點位移(in)		
節點編號	X 方向	Y 方向
1	1.9306E-01	-2.0000E+00
2	-5.4097E-01	-1.9917E+00
3	2.3959E-01	-7.3749E+01
4	-3.0112E-01	-1.6374E+00
5	0.0000E+00	0.0000E+00
6	0.0000E+00	0.0000E+00

表 4-7 例一，最佳設計斷面經結構分析後之各桿件應力

各桿件應力(ksi)	
桿件編號	桿件應力
1	6.6552
2	-1.2923
3	-8.3646
4	-6.6623
5	24.9961
6	-0.2285
7	18.5587
8	-6.9153
9	6.5789
10	1.8276

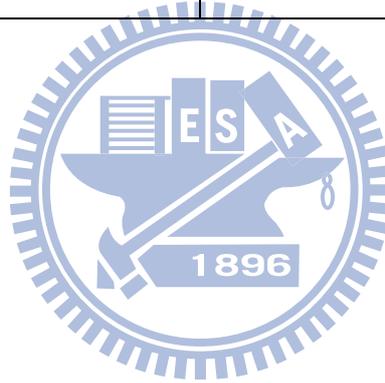


表 4-8 例一，最佳設計斷面比較表

Area (in ²)	Li et al. 【11】			Perez et al. 【31】	Kaveh et al. 【32】	Kaveh et al. 【12】	This work
	PSO	PSOPC	HPSO	PSO	PSACO	HPSACO	Improved PSO
A ₁	33.469	30.569	30.704	33.500	30.068	30.307	30.447
A ₂	0.110	0.100	0.100	0.100	0.100	0.100	0.100
A ₃	23.177	22.974	23.167	22.766	23.207	23.434	23.596
A ₄	15.475	15.148	15.183	14.417	15.168	15.505	15.029
A ₅	3.649	0.100	0.100	0.100	0.100	0.100	0.100
A ₆	0.116	0.547	0.551	0.100	0.536	0.524	0.566
A ₇	8.328	7.493	7.460	7.534	7.462	7.437	7.420
A ₈	23.340	21.159	20.978	20.467	21.228	21.079	20.988
A ₉	23.014	21.556	21.508	20.392	21.630	21.229	21.524
A ₁₀	0.190	0.100	0.100	0.100	0.100	0.100	0.100
Weight (lb)	5529.50	5061.00	5060.92*	5024.21*	5057.36*	5056.56*	5061.36

註：* 表示該組設計斷面下，若將節點位移及桿件應力的精度取至小數點下第 2 至 6 位時，其最大節點位移或最大桿件應力就會不滿足最大容許位移或最大容許應力之限制。

** 表示該文獻中找不到有關於結構矩陣分析次數的相關說明。

表 4-9 例二，桁架受力情形

桁架受力情形(kips)		
節點編號	X 方向	Y 方向
1	0.0	50.0
2	0.0	-150.0
3	0.0	50.0
4	0.0	-150.0

表 4-10 例二，不同粒子群優化演算法所找到的最佳設計斷面

Area (in ²)	PSO	PSOPC	Improved PSO
A ₁	34.9999	24.9456	24.1567
A ₂	0.1000	0.1000	0.1141
A ₃	28.5146	19.4945	24.7967
A ₄	15.9205	13.7102	14.4889
A ₅	12.0559	0.1050	0.1000
A ₆	6.5039	2.7461	1.9742
A ₇	4.9454	17.5269	12.4771
A ₈	25.8923	28.6679	12.7048
A ₉	21.6755	15.7415	20.2231
A ₁₀	0.2384	0.1002	0.1010
Weight (lb)	6217.0910	5358.0330	4679.4847
結構矩陣 分析次數	20,000	20,000	20,000

表 4-11 例二，最佳設計斷面經結構分析後之各節點位移

各節點位移(in)		
節點編號	X 方向	Y 方向
1	-1.9358E-02	-1.1025E+00
2	-6.0860E-01	-2.0000E+00
3	2.2844E-01	-6.5550E-01
4	-3.5818E-01	-1.5555E+00
5	0.0000E+00	0.0000E+00
6	0.0000E+00	0.0000E+00

表 4-12 例二，最佳設計斷面經結構分析後之各桿件應力

各桿件應力(ksi)	
桿件編號	桿件應力
1	6.3455
2	-6.8832
3	-9.9495
4	-6.9560
5	25.0000
6	24.9289
7	16.6293
8	-5.9314
9	7.0480
10	10.9968

表 4-13 例二，最佳設計斷面比較表

Area (in ²)	Li et al. 【11】			Kaveh et al. 【32】	Kaveh et al. 【12】	This work
	PSO	PSOPC	HPSO	PSACO	HPSACO	Improved PSO
A ₁	22.935	23.473	23.353	23.052	23.194	24.1567
A ₂	0.113	0.101	0.100	0.100	0.100	0.1141
A ₃	25.355	25.287	25.502	25.601	24.585	24.7967
A ₄	14.373	14.413	14.250	15.139	14.221	14.4889
A ₅	0.100	0.100	0.100	0.100	0.100	0.1000
A ₆	1.990	1.969	1.972	1.969	1.969	1.9742
A ₇	12.346	12.362	12.363	12.206	12.489	12.4771
A ₈	12.923	12.694	12.894	12.568	12.925	12.7048
A ₉	20.678	20.323	20.356	20.330	20.952	20.2231
A ₁₀	0.100	0.103	0.101	0.100	0.101	0.1010
Weight (lb)	4679.47	4677.70*	4677.29*	4676.05*	4675.78*	4679.4847

註：* 表示該組設計斷面下，若將節點位移及桿件應力的精度取至小數點下第 3 至 4 位時，其最大節點位移或最大桿件應力就會不滿足最大容許位移或最大容許應力之限制。

表 4-14 例三，各桿件起始節點及終止節點之編號

桿件編號	起始節點編號	終止節點編號	桿件編號	起始節點編號	終止節點編號
1	1	2	14	3	10
2	1	4	15	6	7
3	2	3	16	4	9
4	1	5	17	5	8
5	2	6	18	3	8
6	2	5	19	4	7
7	2	4	20	6	9
8	1	3	21	5	10
9	1	6	22	3	7
10	3	6	23	4	8
11	4	5	24	5	9
12	3	4	25	6	10
13	5	6			

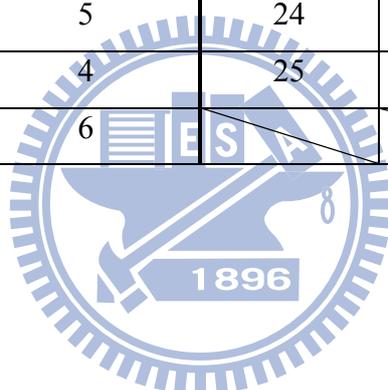


表 4-15 例三，各組設計斷面中所包含之桿件編號

群組編號	所包含之桿件編號
1	1
2	2、3、4、5
3	6、7、8、9
4	10、11
5	12、13
6	14、15、16、17
7	18、19、20、21
8	22、23、24、25

表 4-16 例三，桁架受力情形

桁架受力情形(kips)				
載重組合	節點編號	X 方向	Y 方向	Z 方向
載重組合 1	1	0.0	20.0	-5.0
	2	0.0	-20.0	-5.0
載重組合 2	1	1.0	10.0	-5.0
	2	0.0	10.0	-5.0
	3	0.5	0.0	0.0
	6	0.5	0.0	0.0

表 4-17 例三，各組設計斷面中桿件之最大容許應力

各組設計斷面之最大容許應力(ksi)		
群組編號	容許壓應力	容許拉應力
1	35.092	40.0
2	11.590	40.0
3	17.305	40.0
4	35.092	40.0
5	35.092	40.0
6	6.759	40.0
7	6.959	40.0
8	11.082	40.0

表 4-18 例三，不同粒子群優化演算法所找到的最佳設計斷面

Area (in ²)	PSO	PSOPC	Improved PSO
A ₁	0.0522	0.3942	0.0102
A ₂	1.7926	1.8508	1.9895
A ₃	3.3822	3.2477	2.9893
A ₄	2.2788	0.7730	0.0100
A ₅	0.9779	0.0459	0.0100
A ₆	0.9212	0.6415	0.6865
A ₇	1.6649	1.6912	1.6766
A ₈	2.3521	2.6504	2.6602
Weight (lb)	600.2526	561.1104	545.1648
結構矩陣 分析次數	20,000	20,000	20,000

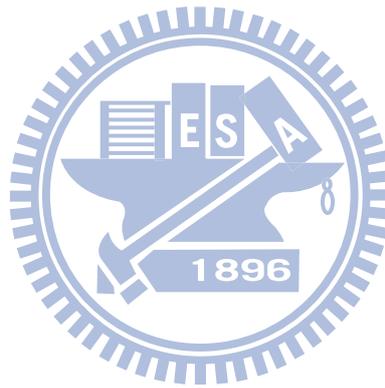


表 4-19 例三，最佳設計斷面經結構分析後之各節點位移

各節點位移(in)						
節點編號	載重組合 1			載重組合 2		
	X 方向	Y 方向	Z 方向	X 方向	Y 方向	Z 方向
1	-1.9868E-02	3.500E-01	-2.8956E-02	6.4247E-03	3.4999E-01	-2.2737E-02
2	1.9868E-02	-3.500E-01	-2.8956E-02	3.3273E-02	3.4999E-01	-3.2606E-02
3	1.1179E-01	-4.0304E-02	-9.9818E-02	1.8072E-02	-3.7977E-02	-1.2253E-01
4	9.7752E-02	2.6874E-02	5.6951E-02	-1.1001E-02	-3.9509E-02	-1.3026E-01
5	-1.1179E-01	4.0304E-02	-9.9818E-02	3.8496E-03	-2.4511E-02	7.9597E-02
6	-9.7752E-02	-2.6874E-02	5.6951E-02	3.2220E-03	-2.6044E-02	8.7324E-02
7	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
8	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
9	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
10	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00

表 4-20 例三，最佳設計斷面經結構分析後之各桿件應力

桿件編號		載重組合 1		載重組合 2	
		桿件應力(ksi)		桿件應力(ksi)	
1	14	5.2983	-2.6017	3.5797	-4.9814
2	15	-6.9791	0.9998	-3.0305	3.8140
3	16	6.9318	0.9998	-2.5932	-5.4158
4	17	6.9318	-2.6017	2.1240	3.3796
5	18	-6.9791	-6.9588	2.5613	-3.7682
6	19	-6.6193	5.3200	2.4755	-3.6782
7	20	4.8587	5.3200	-4.2444	2.0302
8	21	-6.6193	-6.9588	-4.0063	2.1203
9	22	4.8587	-0.2673	2.7136	-4.9122
10	23	-1.7906	-1.1756	-1.5911	-5.5404
11	24	-1.7906	-0.2673	-1.9997	3.4735
12	25	-1.8712	-1.1756	-3.8764	4.1017
13		-1.8712		0.0837	

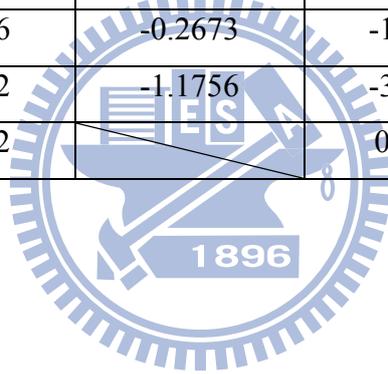


表 4-21 例三，最佳設計斷面比較表

Area (in ²)	Li et al. 【11】			Kaveh et al. 【32】	Kaveh et al. 【12】	This work Improved PSO
	PSO	PSOPC	HPSO	PSACO	HPSACO	
A ₁	9.863	0.010	0.010	0.010	0.010	0.0102
A ₂	1.798	1.979	1.970	2.052	2.054	1.9895
A ₃	3.654	3.011	3.016	3.001	3.008	2.9893
A ₄	0.100	0.100	0.010	0.010	0.010	0.01
A ₅	0.100	0.100	0.010	0.010	0.010	0.01
A ₆	0.596	0.657	0.694	0.684	0.679	0.6865
A ₇	1.659	1.678	1.681	1.616	1.611	1.6766
A ₈	2.612	2.693	2.643	2.673	2.678	2.6602
Weight (lb)	627.08	545.27	545.19	545.04*	544.99*	545.1648

註：* 表示該組設計斷面下，若將節點位移及桿件應力的精度取至小數點下第 3 至 4 位時，其最大節點位移或最大桿件應力就會不滿足最大容許位移或最大容許應力之限制。

表 4-22 例四，各桿件起始節點及終止節點之編號

桿件編號	起始節點編號	終止節點編號	桿件編號	起始節點編號	終止節點編號
1	1	5	30	8	9
2	2	6	31	9	10
3	3	7	32	10	11
4	4	8	33	11	12
5	5	2	34	12	9
6	1	6	35	9	11
7	2	7	36	12	10
8	6	3	37	9	13
9	3	8	38	10	14
10	7	4	39	11	15
11	8	1	40	12	16
12	4	5	41	13	10
13	5	6	42	9	14
14	6	7	43	10	15
15	7	8	44	14	11
16	8	5	45	11	16
17	5	7	46	15	12
18	8	6	47	16	9
19	5	9	48	12	13
20	6	10	49	13	14
21	7	11	50	14	15
22	8	12	51	15	16
23	9	6	52	16	13
24	5	10	53	13	15
25	6	11	54	16	14
26	10	7	55	13	17
27	7	12	56	14	18
28	11	8	57	15	19
29	12	5	58	16	20

表 4-22 例四，各桿件起始節點及終止節點之編號(續)

桿件編號	起始節點編號	終止節點編號	桿件編號	起始節點編號	終止節點編號
59	17	14	66	16	17
60	13	18	67	17	18
61	14	19	68	18	19
62	18	15	69	19	20
63	15	20	70	20	17
64	19	16	71	17	19
65	20	13	72	20	18

表 4-23 例四，各組設計斷面中所包含之桿件編號

群組編號	所包含之桿件編號
1	1、2、3、4
2	5、6、7、8、9、10、11、12
3	13、14、15、16
4	17、18
5	19、20、21、22
6	23、24、25、26、27、28、29、30
7	31、32、33、34
8	35、36
9	37、38、39、40
10	41、42、43、44、45、46、47、48
11	49、50、51、52
12	53、54
13	55、56、57、58
14	59、60、61、62、63、64、65、66
15	67、68、69、70
16	71、72

表 4-24 例四，桁架受力情形

桁架受力(kips)				
載重組合	節點編號	X 方向	Y 方向	Z 方向
載重組合 1	17	5.0	5.0	-5.0
載重組合 2	17	0.0	0.0	-5.0
	18	0.0	0.0	-5.0
	19	0.0	0.0	-5.0
	20	0.0	0.0	-5.0

表 4-25 例四，不同粒子群優化演算法所找到的最佳設計斷面

Area (in ²)	PSO	PSOPC	Improved PSO
A ₁	3.5468	1.4818	1.8428
A ₂	0.2427	0.5274	0.4999
A ₃	0.6712	1.4041	0.1039
A ₄	0.1123	0.9059	0.1000
A ₅	1.0396	2.6571	1.2211
A ₆	0.5078	0.4267	0.5486
A ₇	0.5042	0.1061	0.1000
A ₈	1.6069	1.1063	0.1000
A ₉	0.4148	0.5278	0.4897
A ₁₀	0.8059	0.5486	0.4960
A ₁₁	0.7777	0.1000	0.1033
A ₁₂	1.5967	0.4056	0.1502
A ₁₃	0.1000	0.1892	0.1558
A ₁₄	1.0250	0.5860	0.5747
A ₁₅	0.1024	0.8597	0.4331
A ₁₆	1.0500	0.2686	0.5219
Weight (lb)	646.3323	550.4602	381.6105
結構矩陣 分析次數	20,000	20,000	20,000

表 4-26 例四，最佳設計斷面經結構分析後之各節點位移(第一組載重)

各節點位移(in)			
節點編號	載重組合 1		
	X 方向	Y 方向	Z 方向
1	0.0000E+00	0.0000E+00	0.0000E+00
2	0.0000E+00	0.0000E+00	0.0000E+00
3	0.0000E+00	0.0000E+00	0.0000E+00
4	0.0000E+00	0.0000E+00	0.0000E+00
5	4.9167E-02	4.9167E-02	1.6986E-02
6	5.0453E-02	4.1032E-02	-5.1077E-03
7	4.7066E-02	4.7066E-02	-2.2974E-02
8	4.1032E-02	5.0453E-02	-5.1077E-03
9	1.1404E-01	1.1404E-01	3.0699E-02
10	1.0064E-01	8.6315E-02	-1.1756E-02
11	9.8415E-02	9.8415E-02	-4.7027E-02
12	8.6315E-02	1.0064E-01	-1.1756E-02
13	1.5952E-01	1.5952E-01	2.5775E-02
14	1.9980E-01	1.4952E-01	-2.5977E-02
15	1.5023E-01	1.5023E-01	-7.7699E-02
16	1.4952E-01	1.9980E-01	-2.5977E-02
17	2.4999E-01	2.4999E-01	-7.0894E-02
18	2.1310E-01	1.9978E-01	-2.7574E-02
19	2.1167E-01	2.1167E-01	-1.0655E-01
20	1.9978E-01	2.1310E-01	-2.7574E-02

表 4-27 例四，最佳設計斷面經結構分析後之各節點位移(第二組載重)

各節點位移(in)			
節點編號	載重組合 2		
	X 方向	Y 方向	Z 方向
1	0.0000E+00	0.0000E+00	0.0000E+00
2	0.0000E+00	0.0000E+00	0.0000E+00
3	0.0000E+00	0.0000E+00	0.0000E+00
4	0.0000E+00	0.0000E+00	0.0000E+00
5	-6.1828E-03	-6.1828E-03	-1.6098E-02
6	6.1828E-03	-6.1828E-03	-1.6098E-02
7	6.1828E-03	6.1828E-03	-1.6098E-02
8	-6.1828E-03	6.1828E-03	-1.6098E-02
9	-1.9962E-03	-1.9962E-03	-4.0056E-02
10	1.9962E-03	-1.9962E-03	-4.0056E-02
11	1.9962E-03	1.9962E-03	-4.0056E-02
12	-1.9962E-03	1.9962E-03	-4.0056E-02
13	-3.1330E-02	-3.1330E-02	-1.0214E-01
14	3.1330E-02	-3.1330E-02	-1.0214E-01
15	3.1330E-02	3.1330E-02	-1.0214E-01
16	-3.1330E-02	3.1330E-02	-1.0214E-01
17	-8.6880E-03	-8.6880E-03	-2.4997E-01
18	8.6880E-03	-8.6880E-03	-2.4997E-01
19	8.6880E-03	8.6880E-03	-2.4997E-01
20	-8.6880E-03	8.6880E-03	-2.4997E-01

表 4-28 例四，最佳設計斷面經結構分析後之各桿件應力(第一組載重)

桿件編號			載重組合 1		
			桿件應力(ksi)		
1	25	49	2.8310	2.4282	3.3563
2	26	50	-0.8513	-2.2426	0.0590
3	27	51	-3.8290	-2.2426	0.0590
4	28	52	-0.8513	2.4282	3.3563
5	29	53	-2.7116	2.4734	-0.7742
6	30	54	3.1933	-3.0456	4.1895
7	31	55	2.3719	-1.1167	-16.1115
8	32	56	-2.9058	1.0084	-0.2661
9	33	57	-2.9058	1.0084	-4.8087
10	34	58	2.3719	-1.1167	-0.2661
11	35	59	3.1933	-1.3021	-4.8438
12	36	60	-2.7116	1.1938	1.7936
13	37	61	0.1071	-0.8205	1.4575
14	38	62	0.5028	-2.3703	-1.6323
15	39	63	0.5028	-5.1121	-1.6323
16	40	64	0.1071	-2.3703	1.4575
17	41	65	-0.1751	-2.6743	1.7936
18	42	66	0.7850	3.8278	-4.8438
19	43	67	2.2855	2.0629	-3.0747
20	44	68	-1.1080	-2.7055	0.9913
21	45	69	-4.0088	-2.7055	0.9913
22	46	70	-1.1080	2.0629	-3.0747
23	47	71	-3.0456	3.8278	-3.1935
24	48	72	2.4734	-2.6743	1.1102

表 4-29 例四，最佳設計斷面經結構分析後之各桿件應力(第二組載重)

桿件編號			載重組合 2		
			桿件應力(ksi)		
1	25	49	-2.6831	-0.2533	5.2217
2	26	50	-2.6831	-0.2533	5.2217
3	27	51	-2.6831	-0.2533	5.2217
4	28	52	-2.6831	-0.2533	5.2217
5	29	53	-0.1244	-0.2533	5.2217
6	30	54	-0.1244	-0.2533	5.2217
7	31	55	-0.1244	0.3327	-24.6374
8	32	56	-0.1244	0.3327	-24.6374
9	33	57	-0.1244	0.3327	-24.6374
10	34	58	-0.1244	0.3327	-24.6374
11	35	59	-0.1244	0.3327	-2.2596
12	36	60	-0.1244	0.3327	-2.2596
13	37	61	1.0305	-10.3482	-2.2596
14	38	62	1.0305	-10.3482	-2.2596
15	39	63	1.0305	-10.3482	-2.2596
16	40	64	1.0305	-10.3482	-2.2596
17	41	65	1.0305	0.1521	-2.2596
18	42	66	1.0305	0.1521	-2.2596
19	43	67	-3.9929	0.1521	1.4480
20	44	68	-3.9929	0.1521	1.4480
21	45	69	-3.9929	0.1521	1.4480
22	46	70	-3.9929	0.1521	1.4480
23	47	71	-0.2533	0.1521	1.4480
24	48	72	-0.2533	0.1521	1.4480

表 4-30 例四，最佳設計斷面比較表

Area (in ²)	Li et al. 【11】			Perez et al. 【31】	This work
	PSO	PSOPC	HPSO	PSO	Improved PSO
A ₁	41.794	1.855	1.857	1.7427	1.8428
A ₂	0.195	0.504	0.505	0.5185	0.4999
A ₃	10.797	0.100	0.100	0.1000	0.1039
A ₄	6.861	0.100	0.100	0.1000	0.1000
A ₅	0.438	1.253	1.255	1.3079	1.2211
A ₆	0.286	0.505	0.503	0.5193	0.5486
A ₇	18.309	0.100	0.100	0.1000	0.1000
A ₈	1.220	0.100	0.100	0.1000	0.1000
A ₉	5.933	0.497	0.496	0.5142	0.4897
A ₁₀	19.545	0.508	0.506	0.5464	0.496
A ₁₁	0.159	0.100	0.100	0.1000	0.1033
A ₁₂	0.151	0.100	0.100	0.1095	0.1052
A ₁₃	10.127	0.100	0.100	0.1615	0.1558
A ₁₄	7.320	0.525	0.524	0.5092	0.5747
A ₁₅	3.812	0.394	0.400	0.4967	0.4331
A ₁₆	18.196	0.535	0.534	0.5619	0.5219
Weight (lb)	6818.67	369.65*	369.65*	381.91	381.6105

註：* 表示該組設計斷面下，若將節點位移及桿件應力的精度取至小數點下第 6 位時，其最大節點位移或最大桿件應力就會不滿足最大容許位移或最大容許應力之限制。且在第二組載重下，最大節點位移及最大桿件應力則超出容許值許多。

表 4-31 例五，各桿件起始節點及終止節點之編號

桿件編號	起始節點編號	終止節點編號	桿件編號	起始節點編號	終止節點編號
1	1	2	31	8	26
2	1	3	32	9	28
3	1	4	33	10	30
4	1	5	34	11	32
5	1	6	35	12	34
6	1	7	36	13	36
7	1	8	37	2	15
8	1	9	38	3	15
9	1	10	39	3	17
10	1	11	40	4	17
11	1	12	41	4	19
12	1	13	42	5	19
13	2	3	43	5	21
14	3	4	44	6	21
15	4	5	45	6	23
16	5	6	46	7	23
17	6	7	47	7	25
18	7	8	48	8	25
19	8	9	49	8	27
20	9	10	50	9	27
21	10	11	51	9	29
22	11	12	52	10	29
23	12	13	53	10	31
24	13	2	54	11	31
25	2	14	55	11	33
26	3	16	56	12	33
27	4	18	57	12	35
28	5	20	58	13	35
29	6	22	59	13	37
30	7	24	60	2	37

表 4-31 例五，各桿件起始節點及終止節點之編號(續)

桿件編號	起始節點編號	終止節點編號	桿件編號	起始節點編號	終止節點編號
61	14	15	91	26	44
62	15	16	92	28	45
63	16	17	93	30	46
64	17	18	94	32	47
65	18	19	95	34	48
66	19	20	96	36	49
67	20	21	97	15	38
68	21	22	98	15	39
69	22	23	99	17	39
70	23	24	100	17	40
71	24	25	101	19	40
72	25	26	102	19	41
73	26	27	103	21	41
74	27	28	104	21	42
75	28	29	105	23	42
76	29	30	106	23	43
77	30	31	107	25	43
78	31	32	108	25	44
79	32	33	109	27	44
80	33	34	110	27	45
81	34	35	111	29	45
82	35	36	112	29	46
83	36	37	113	31	46
84	37	14	114	31	47
85	14	38	115	33	47
86	16	39	116	33	48
87	18	40	117	35	48
88	20	41	118	35	49
89	22	42	119	37	49
90	24	43	120	37	38

表 4-32 例五，各組設計斷面中所包含之桿件編號

群組編號	所包含之桿件編號
1	1、2、...、12
2	13、14、...、24
3	25、26、...、36
4	37、38、...、60
5	61、62、...、84
6	85、86、...、96
7	97、98、...、120

表 4-33 例五，桁架受力情形

桁架受力(kips)			
節點編號	X 方向	Y 方向	Z 方向
1	0.0	0.0	-13.49
2,3,...,14	0.0	0.0	-6.744
15,16,...,37	0.0	0.0	-2.248

表 4-34 例五，不同粒子群優化演算法所找到的最佳設計斷面

Area (in ²)	PSO	PSOPC	Improved PSO
A ₁	6.2016	3.7825	3.3436
A ₂	4.1650	8.6086	2.7146
A ₃	4.3157	3.8915	3.8777
A ₄	2.6396	2.7589	2.5662
A ₅	1.0626	1.9051	1.2460
A ₆	14.516	3.3822	3.4320
A ₇	2.6757	2.8560	2.8302
Weight (lb)	30477.9549	24360.4800	20122.9726
結構矩陣 分析次數	20,000	20,000	20,000

表 4-35 例五，最佳設計斷面經結構分析後之各節點位移

各節點位移(in)			
節點編號	X 方向	Y 方向	Z 方向
1	0.0000E+00	-6.2594E-03	-1.0859E-02
2	0.0000E+00	-5.5184E-02	-1.6572E-01
3	2.2785E-02	-4.7880E-02	-1.8746E-01
4	4.1919E-02	-3.0007E-02	-1.9544E-01
5	4.8949E-02	-4.7918E-03	-1.9670E-01
6	4.2042E-02	1.9911E-02	-1.9544E-01
7	2.4006E-02	3.7399E-02	-1.9392E-01
8	0.0000E+00	4.3657E-02	-1.9330E-01
9	-2.4006E-02	3.7399E-02	-1.9392E-01
10	-4.2042E-02	1.9911E-02	-1.9544E-01
11	-4.8949E-02	-4.7918E-03	-1.9670E-01
12	-4.1919E-02	-3.0007E-02	-1.9544E-01
13	-2.2785E-02	-4.7880E-02	-1.8746E-01
14	0.0000E+00	-8.1736E-02	-1.4622E-01
15	-1.9191E-02	1.34369E-02	1.1582E-02
16	-1.9461E-02	2.0960E-02	8.0340E-03
17	-4.6652E-02	4.2874E-02	3.9023E-02
18	-2.7199E-02	1.3191E-02	8.0413E-03
19	-7.1823E-02	1.7659E-02	5.0639E-02
20	-3.3392E-02	-1.3956E-03	1.0810E-02
21	-7.6635E-02	-2.1740E-02	5.6866E-02
22	-3.0622E-02	-1.8839E-02	1.3216E-02
23	-5.8233E-02	-5.9329E-02	6.0663E-02
24	-1.8327E-02	-3.2830E-02	1.4816E-02
25	-2.1705E-02	-8.2065E-02	6.2581E-02

表 4-35 例五，最佳設計斷面經結構分析後之各節點位移(續)

各節點位移(in)			
節點編號	X 方向	Y 方向	Z 方向
26	0.0000E+00	-3.8177E-02	1.5381E-02
27	2.1705E-02	-8.2065E-02	6.2581E-02
28	1.8327E-02	-3.2830E-02	1.4816E-02
29	5.8233E-02	-5.9329E-02	6.0663E-02
30	3.0622E-02	-1.8839E-02	1.3216E-02
31	7.6635E-02	-2.1740E-02	5.6866E-02
32	3.3392E-02	-1.3956E-03	1.0810E-02
33	7.1823E-02	1.7659E-02	5.0639E-02
34	2.7199E-02	1.3191E-02	8.0413E-03
35	4.6652E-02	4.2874E-02	3.9023E-02
36	1.9461E-02	2.0960E-02	8.0340E-03
37	1.9191E-02	3.4369E-02	1.1582E-02
38	0.0000E+00	0.0000E+00	0.0000E+00
39	0.0000E+00	0.0000E+00	0.0000E+00
40	0.0000E+00	10.0000E+00	0.0000E+00
41	0.0000E+00	0.0000E+00	0.0000E+00
42	0.0000E+00	0.0000E+00	0.0000E+00
43	0.0000E+00	0.0000E+00	0.0000E+00
44	0.0000E+00	0.0000E+00	0.0000E+00
45	0.0000E+00	0.0000E+00	0.0000E+00
46	0.0000E+00	0.0000E+00	0.0000E+00
47	0.0000E+00	0.0000E+00	0.0000E+00
48	0.0000E+00	0.0000E+00	0.0000E+00
49	0.0000E+00	0.0000E+00	0.0000E+00

表 4-36 例五，最佳設計斷面經結構分析後之最大容許應力

群組編號	容許壓應力(ksi)	容許拉應力(ksi)
1	2.6162	34.8
2	7.5632	34.8
3	4.0564	34.8
4	2.2514	34.8
5	3.2463	34.8
6	6.5465	34.8
7	2.8197	34.8

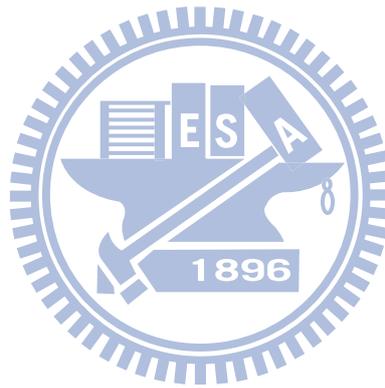


表 4-37 例五，最佳設計斷面經結構分析後之各桿件應力

桿件編號				桿件應力(ksi)			
1	31	61	91	-2.5229	-2.7704	-2.7525	-3.1432
2	32	62	92	-1.9710	-2.7570	0.8921	-3.1328
3	33	63	93	-1.9079	-2.7174	0.8921	-3.1020
4	34	64	94	-1.9690	-2.6485	1.7293	-3.0484
5	35	65	95	-2.0508	-2.5184	1.7293	-2.9472
6	36	66	96	-2.1125	-2.1464	2.0222	-2.6579
7	37	67	97	-2.1350	-0.4330	2.0222	-1.0677
8	38	68	98	-2.1125	-2.2504	2.1772	-2.7418
9	39	69	99	-2.0508	-1.2378	2.1772	-1.6728
10	40	70	100	-1.9690	-1.4456	2.2664	-2.1367
11	41	71	101	-1.9079	-1.4423	2.2664	-1.7718
12	42	72	102	-1.9710	-1.2410	2.2966	-2.0377
13	43	73	103	-5.1447	-1.4762	2.2966	-1.8037
14	44	74	104	-5.6333	-1.2071	2.2664	-2.0059
15	45	75	105	-5.6348	-1.4433	2.2664	-1.8375
16	46	76	106	-5.5214	-1.2400	2.1772	-1.9721
17	47	77	107	-5.4074	-1.3788	2.1772	-1.8809
18	48	78	108	-5.3405	-1.3045	2.0222	-1.9286
19	49	79	109	-5.3405	-1.3045	2.0222	-1.9286
20	50	80	110	-5.4074	-1.3788	1.7293	-1.8809
21	51	81	111	-5.5214	-1.2400	1.7293	-1.9721
22	52	82	112	-5.6348	-1.4433	0.8921	-1.8375
23	53	83	113	-5.6333	-1.2071	0.8921	-2.0059
24	54	84	114	-5.1447	-1.4762	-2.7525	-1.8037
25	55	85	115	-4.0271	-1.2410	-6.0975	-2.0377
26	56	86	116	-2.1464	-1.4423	-2.6579	-1.7718
27	57	87	117	-2.5184	-1.4456	-2.9472	-2.1367
28	58	88	118	-2.6485	-1.2378	-3.0484	-1.6728
29	59	89	119	-2.7174	-2.2504	-3.1020	-2.7418
30	60	90	120	-2.7570	-0.4330	-3.1328	-1.0677

表 4-38 例五，最佳設計斷面經 SAP2000 結構分析後之各節點位移

各節點位移(in)			
節點編號	X 方向	Y 方向	Z 方向
1	-2.8190E-17	-6.2590E-03	-1.0859E-02
2	-7.8060E-18	-5.5184E-02	-1.6572E-01
3	2.2785E-02	-4.7880E-02	-1.8746E-01
4	4.1919E-02	-3.0007E-02	-1.9544E-01
5	4.8949E-02	-4.7920E-03	-1.9670E-01
6	4.2042E-02	1.9911E-02	-1.9544E-01
7	2.4006E-02	3.7399E-02	-1.9392E-01
8	-1.9300E-17	4.3657E-02	-1.9330E-01
9	-2.4006E-02	3.7399E-02	-1.9392E-01
10	-4.2042E-02	1.9911E-02	-1.9544E-01
11	-4.8949E-02	-4.7920E-03	-1.9670E-01
12	-4.1919E-02	-3.0007E-02	-1.9544E-01
13	-2.2785E-02	-4.7880E-02	-1.8746E-01
14	-1.7350E-18	-8.1735E-02	-1.4622E-01
15	-1.9191E-02	1.34369E-02	1.1582E-02
16	-1.9461E-02	2.0960E-02	8.0340E-03
17	-4.6652E-02	4.2874E-02	3.9023E-02
18	-2.7199E-02	1.3191E-02	8.0410E-03
19	-7.1822E-02	1.7659E-02	5.0639E-02
20	-3.3392E-02	-1.3960E-03	1.0811E-02
21	-7.6635E-02	-2.1740E-02	5.6866E-02
22	-3.0622E-02	-1.8839E-02	1.3216E-02
23	-5.8233E-02	-5.9329E-02	6.0663E-02
24	-1.8327E-02	-3.2830E-02	1.4816E-02
25	-2.1705E-02	-8.2065E-02	6.2581E-02

表 4-38 例五，最佳設計斷面經 SAP2000 結構分析後之各節點位移(續)

各節點位移(in)			
節點編號	X 方向	Y 方向	Z 方向
26	-1.7350E-18	-3.8177E-02	1.5381E-02
27	2.1705E-02	-8.2065E-02	6.2581E-02
28	1.8327E-02	-3.2830E-02	1.4816E-02
29	5.8233E-02	-5.9329E-02	6.0663E-02
30	3.0622E-02	-1.8839E-02	1.3216E-02
31	7.6635E-02	-2.1740E-02	5.6866E-02
32	3.3392E-02	-1.3960E-03	1.0811E-02
33	7.1822E-02	1.7659E-02	5.0639E-02
34	2.7199E-02	1.3191E-02	8.0410E-03
35	4.6652E-02	4.2874E-02	3.9023E-02
36	1.9461E-02	2.0960E-02	8.0340E-03
37	1.9191E-02	3.4369E-02	1.1582E-02
38	0.0000E+00	0.0000E+00	0.0000E+00
39	0.0000E+00	0.0000E+00	0.0000E+00
40	0.0000E+00	10.0000E+00	0.0000E+00
41	0.0000E+00	0.0000E+00	0.0000E+00
42	0.0000E+00	0.0000E+00	0.0000E+00
43	0.0000E+00	0.0000E+00	0.0000E+00
44	0.0000E+00	0.0000E+00	0.0000E+00
45	0.0000E+00	0.0000E+00	0.0000E+00
46	0.0000E+00	0.0000E+00	0.0000E+00
47	0.0000E+00	0.0000E+00	0.0000E+00
48	0.0000E+00	0.0000E+00	0.0000E+00
49	0.0000E+00	0.0000E+00	0.0000E+00

表 4-39 例五，最佳設計斷面經 SAP2000 結構分析後之各桿件應力

桿件編號				桿件應力(ksi)			
1	31	61	91	-2.5230	-2.7705	-2.7528	-3.1434
2	32	62	92	-1.9709	-2.7570	0.8925	-3.1329
3	33	63	93	-1.9078	-2.7173	0.8925	-3.1020
4	34	64	94	-1.9691	-2.6485	1.7295	-3.0484
5	35	65	95	-2.0508	-2.5185	1.7295	-2.9473
6	36	66	96	-2.1124	-2.1464	2.0225	-2.6579
7	37	67	97	-2.1351	-0.4329	2.0225	-1.0678
8	38	68	98	-2.1124	-2.2504	2.1774	-2.7419
9	39	69	99	-2.0508	-1.2376	2.1774	-1.6727
10	40	70	100	-1.9691	-1.4457	2.2665	-2.1366
11	41	71	101	-1.9078	-1.4422	2.2665	-1.7720
12	42	72	102	-1.9709	-1.2411	2.2970	-2.0377
13	43	73	103	-5.1448	-1.4761	2.2970	-1.8038
14	44	74	104	-5.6332	-1.2072	2.2665	-2.0059
15	45	75	105	-5.6347	-1.4434	2.2665	-1.8373
16	46	76	106	-5.5213	-1.2400	2.1774	-1.9719
17	47	77	107	-5.4074	-1.3787	2.1774	-1.8808
18	48	78	108	-5.3404	-1.3047	2.0225	-1.9285
19	49	79	109	-5.3404	-1.3047	2.0225	-1.9285
20	50	80	110	-5.4074	-1.3787	1.7295	-1.8808
21	51	81	111	-5.5213	-1.2400	1.7295	-1.9719
22	52	82	112	-5.6347	-1.4434	0.8925	-1.8373
23	53	83	113	-5.6332	-1.2072	0.8925	-2.0059
24	54	84	114	-5.1448	-1.4761	-2.7528	-1.8038
25	55	85	115	-4.0271	-1.2411	-6.0973	-2.0377
26	56	86	116	-2.1464	-1.4422	-2.6579	-1.7720
27	57	87	117	-2.5185	-1.4457	-2.9473	-2.1366
28	58	88	118	-2.6485	-1.2376	-3.0484	-1.6727
29	59	89	119	-2.7173	-2.2504	-3.1020	-2.7419
30	60	90	120	-2.7570	-0.4329	-3.1329	-1.0678

附圖

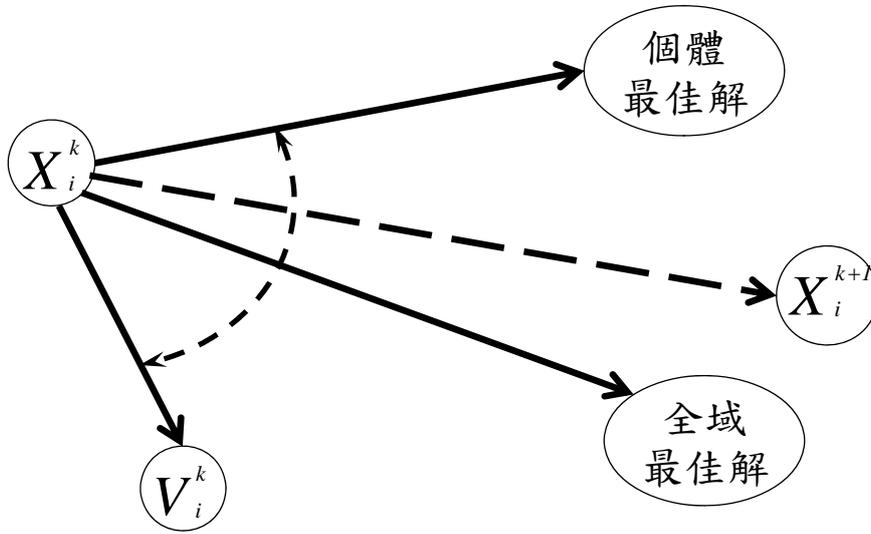


圖 2-1 標準粒子群優化演算法的粒子速度及位置更新示意圖

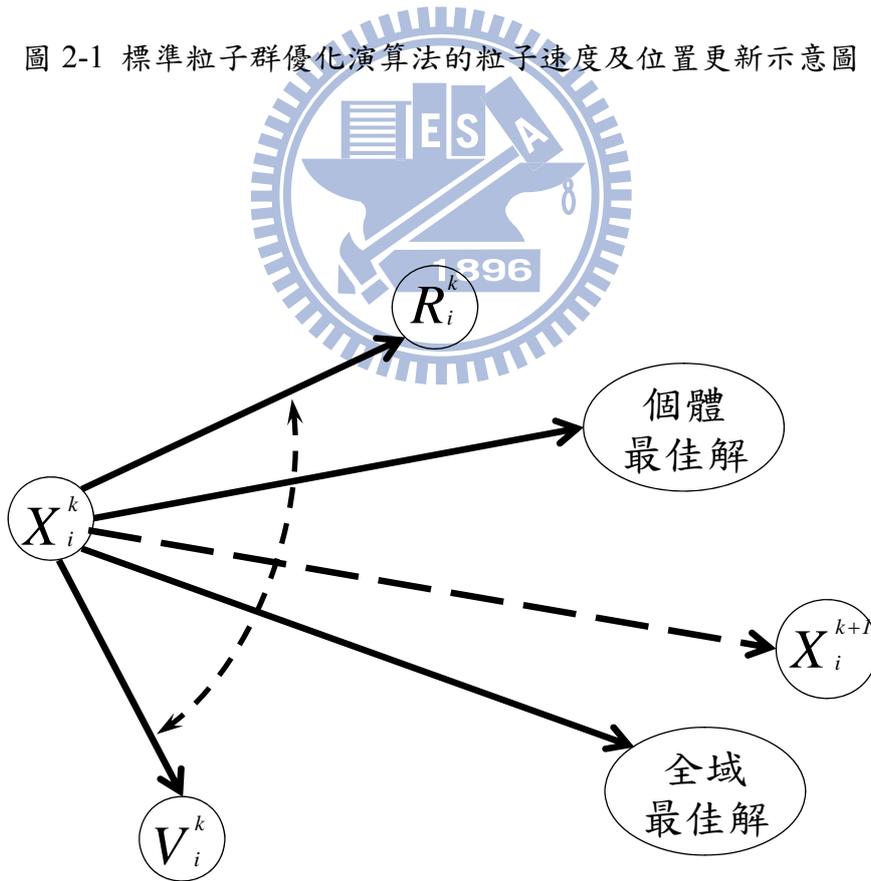


圖 2-2 含被動聚集因子之粒子群優化演算法的粒子速度及位置更新示意圖

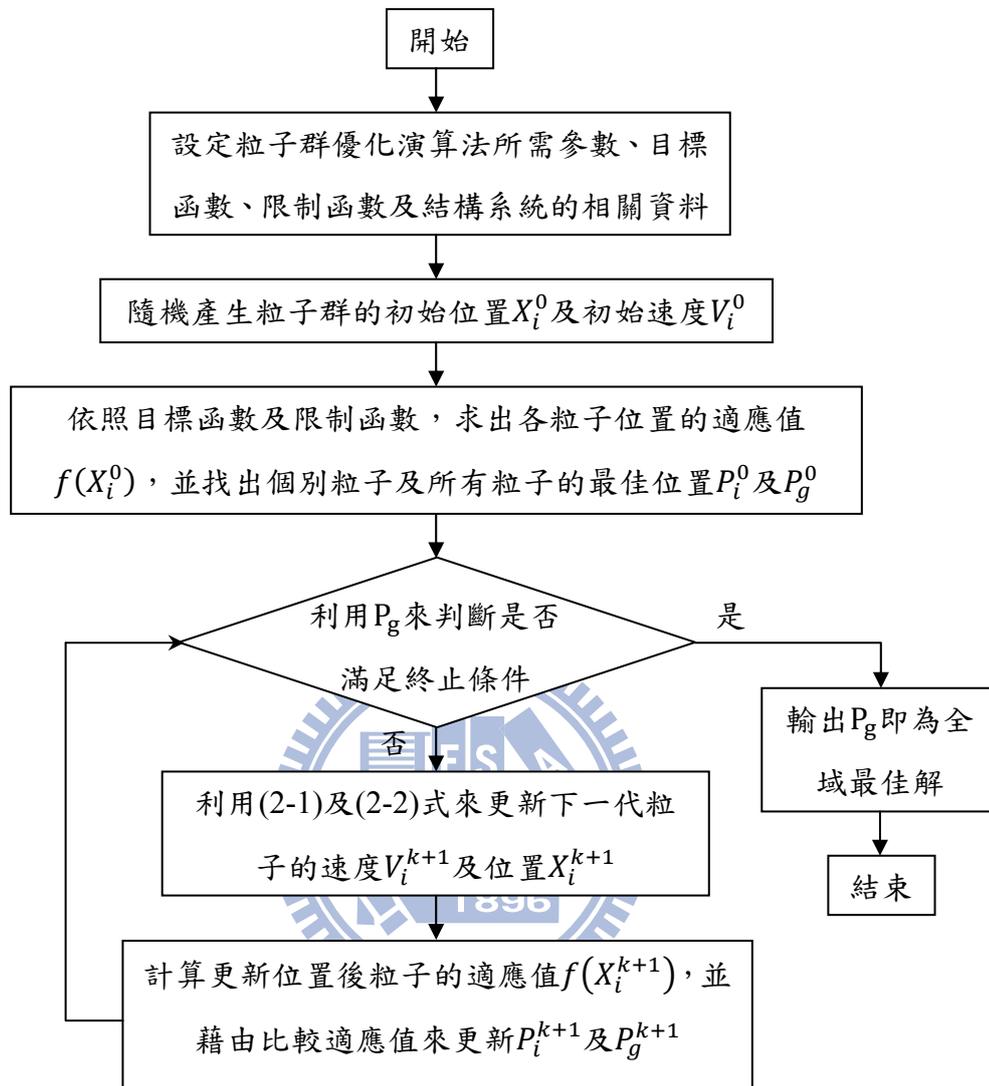


圖 2-3 粒子群優化演算法流程圖

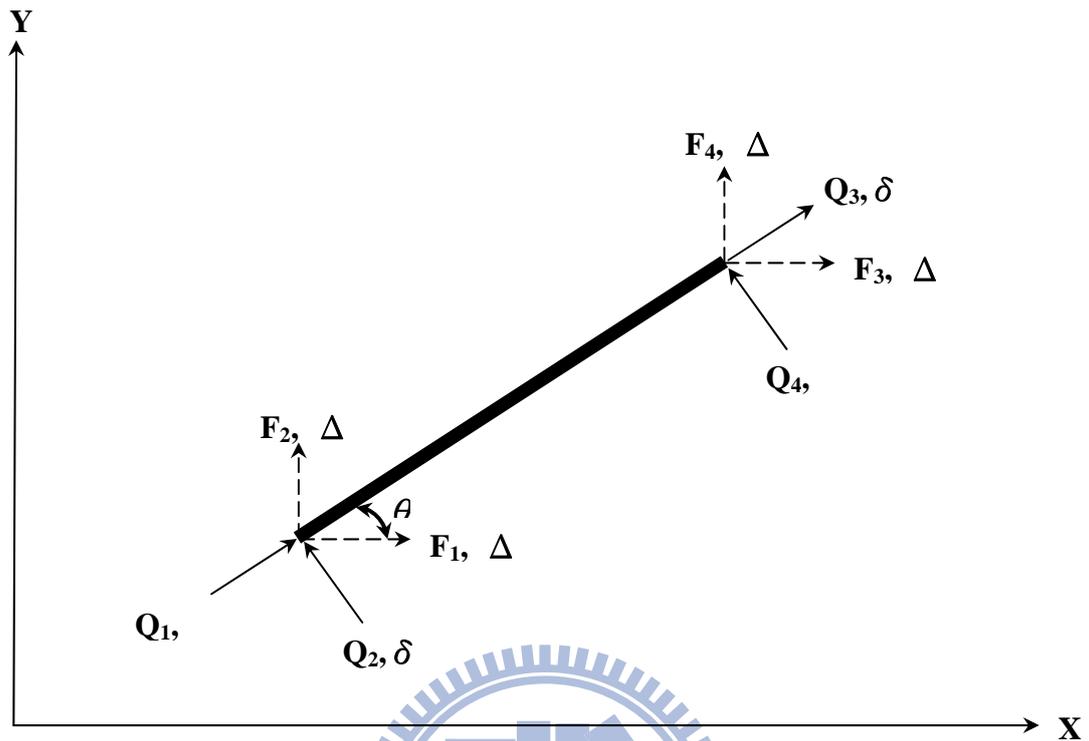


圖 3-1 局部座標及整體座標下的二維桁架桿件示意圖

註：

$Q_1 \sim Q_4$ ：局部座標下的桿件內力

$\delta_1 \sim \delta_4$ ：局部座標下的節點位移

$F_1 \sim F_4$ ：整體座標下的桿件內力

$\Delta_1 \sim \Delta_4$ ：整體座標下的節點位移

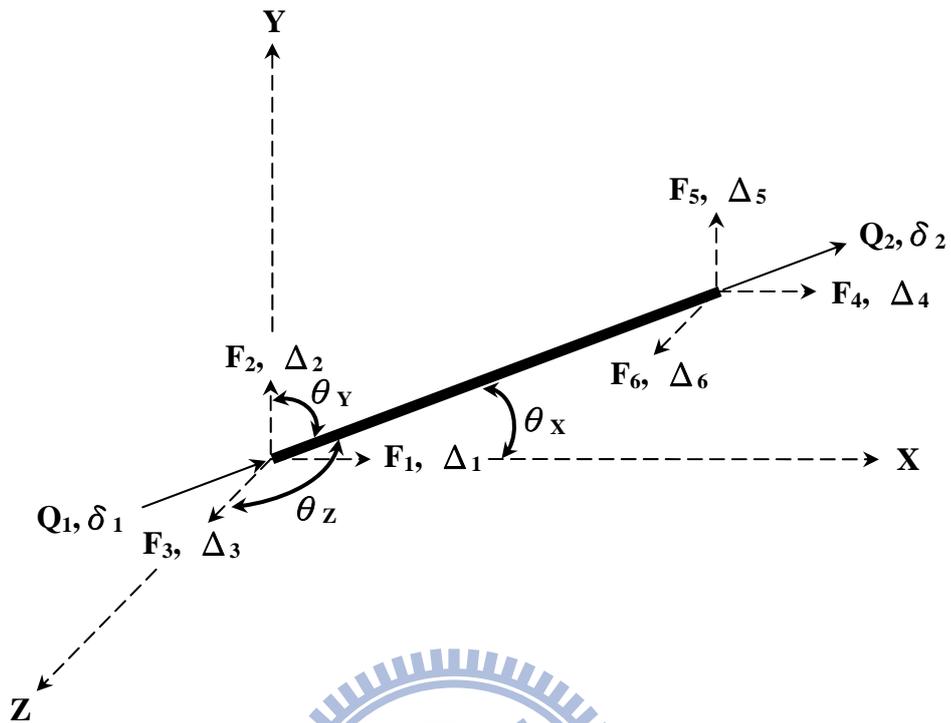


圖 3-2 局部座標及整體座標下的三維桁架桿件示意圖

註：

$Q_1 \sim Q_2$ ：局部座標下的桿件內力

$\delta_1 \sim \delta_2$ ：局部座標下的節點位移

$F_1 \sim F_6$ ：整體座標下的桿件內力

$\Delta_1 \sim \Delta_6$ ：整體座標下的節點位移

θ_x ：桿件與 X 軸之夾角

θ_y ：桿件與 Y 軸之夾角

θ_z ：桿件與 Z 軸之夾角

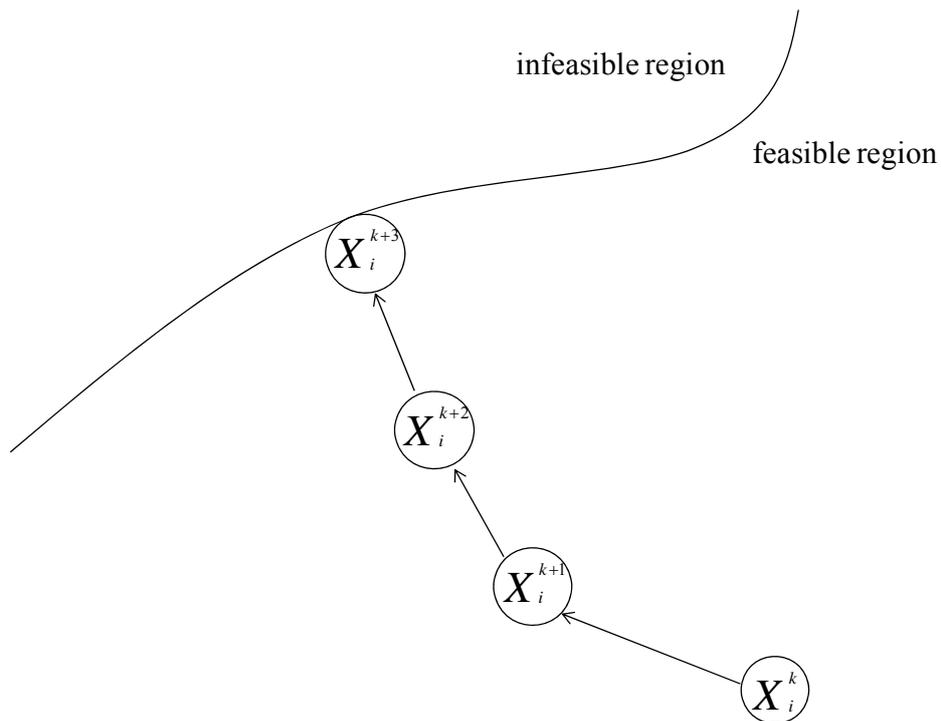


圖 3-3 標準之粒子搜尋示意圖

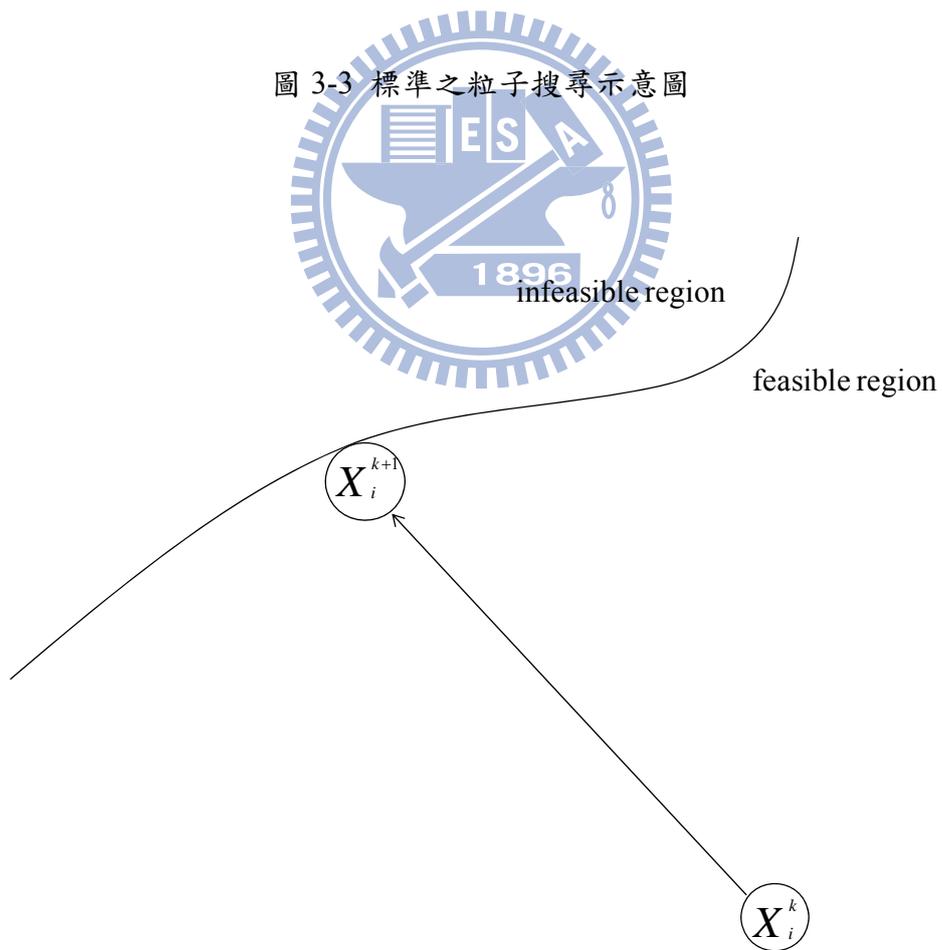


圖 3-4 使用邊界移動法之粒子搜尋示意圖

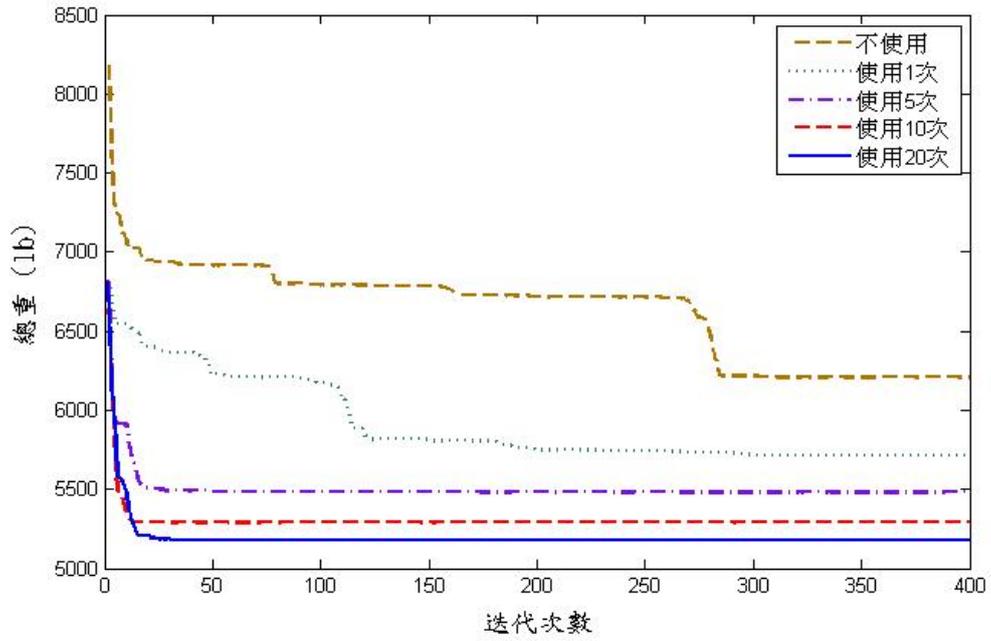


圖 3-5 邊界移動法使用不同次數情況下的收斂情形

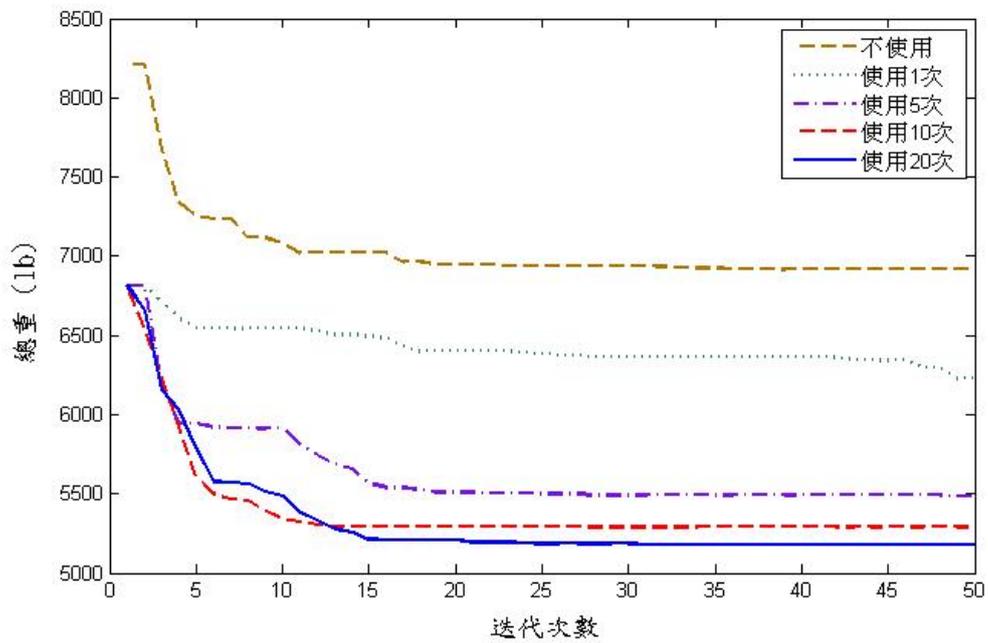


圖 3-6 邊界移動法使用不同次數情況下的收斂情形

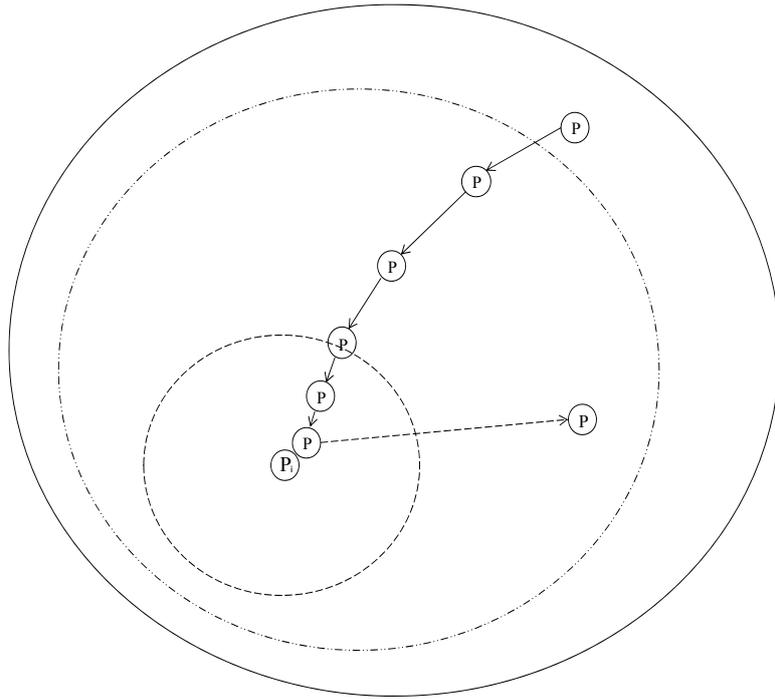


圖 3-7 粒子飛離法示意圖

註：

- 解空間
- - - 基準空間
- · - · 飛離空間
- 粒子搜尋路徑
- - - - 粒子飛離路徑
- ⊙(P) 粒子
- ⊙(P_i) 粒子收斂中心

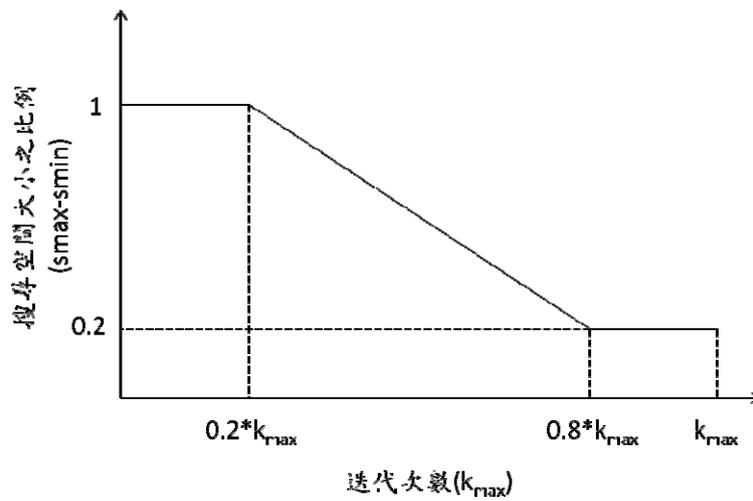


圖 3-8 飛離空間縮小的情形

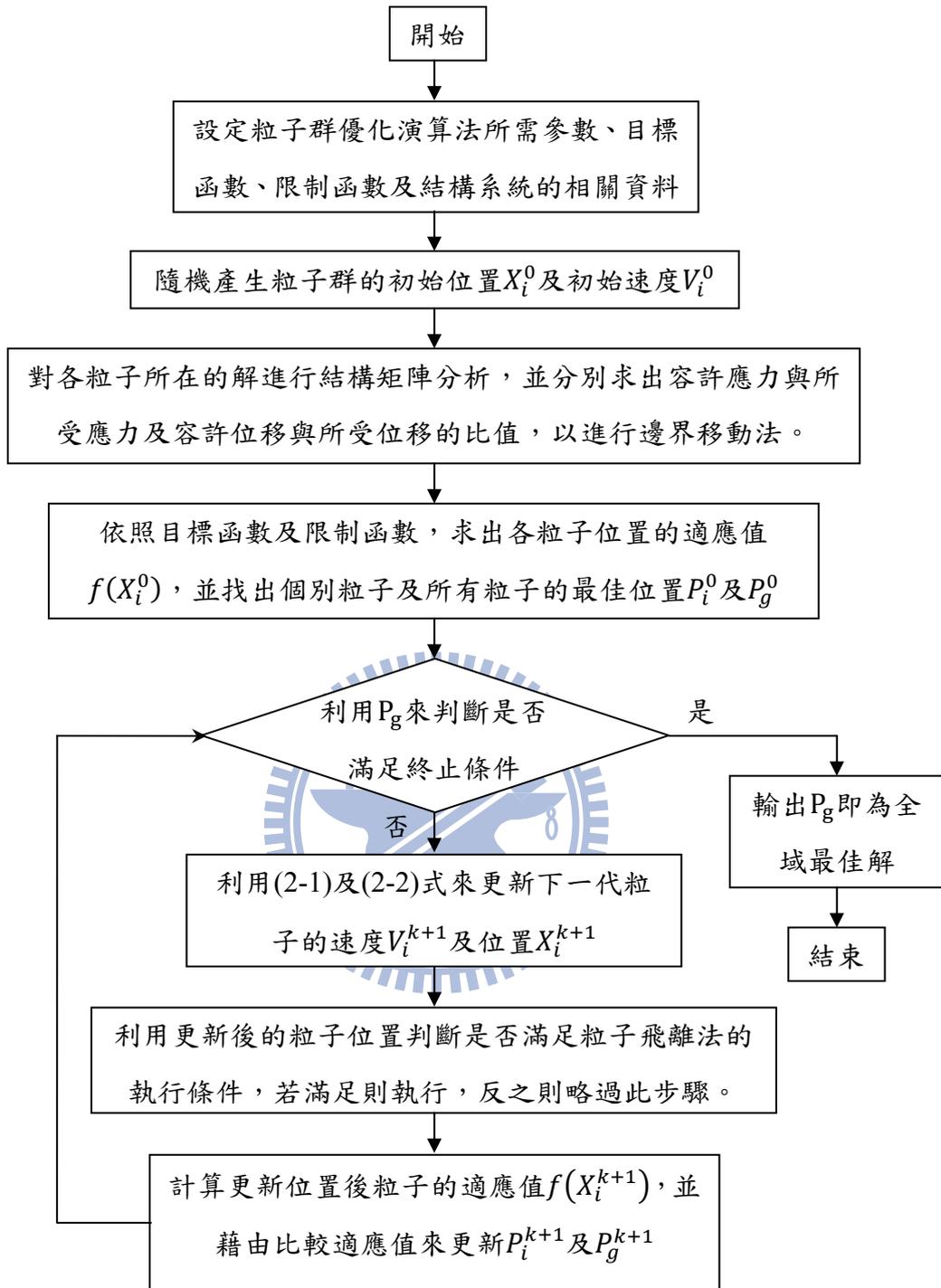


圖 3-9 改良式粒子群優化演算法流程圖

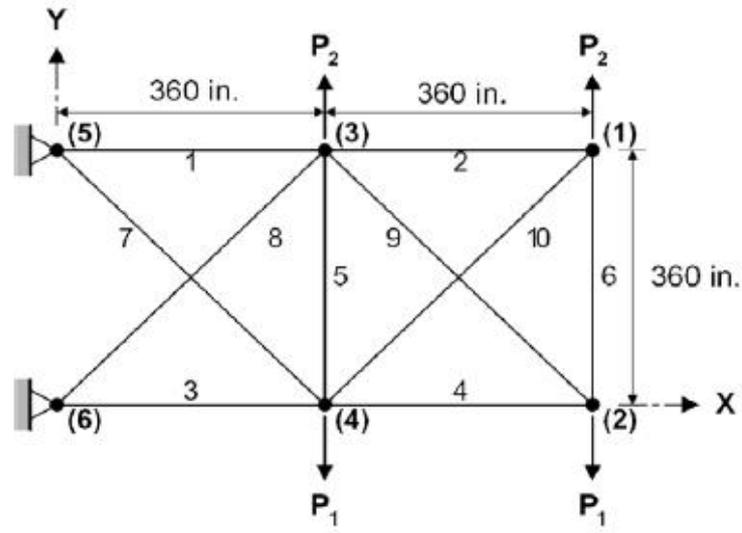


圖 4-1 平面 10 根桿件桁架結構

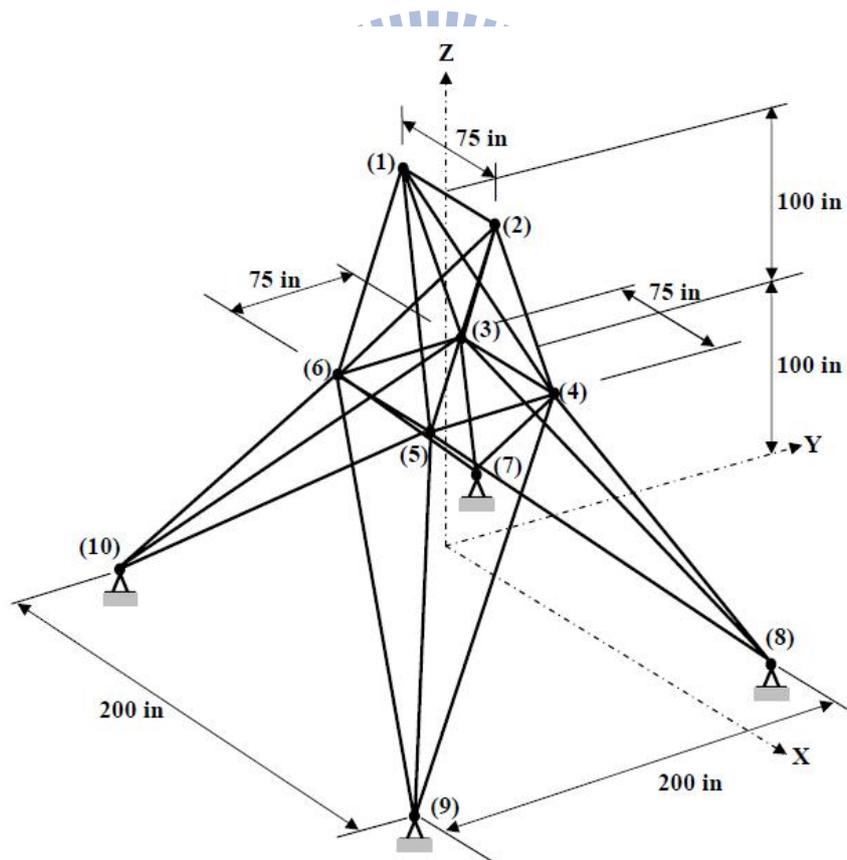


圖 4-2 空間 25 根桿件桁架結構

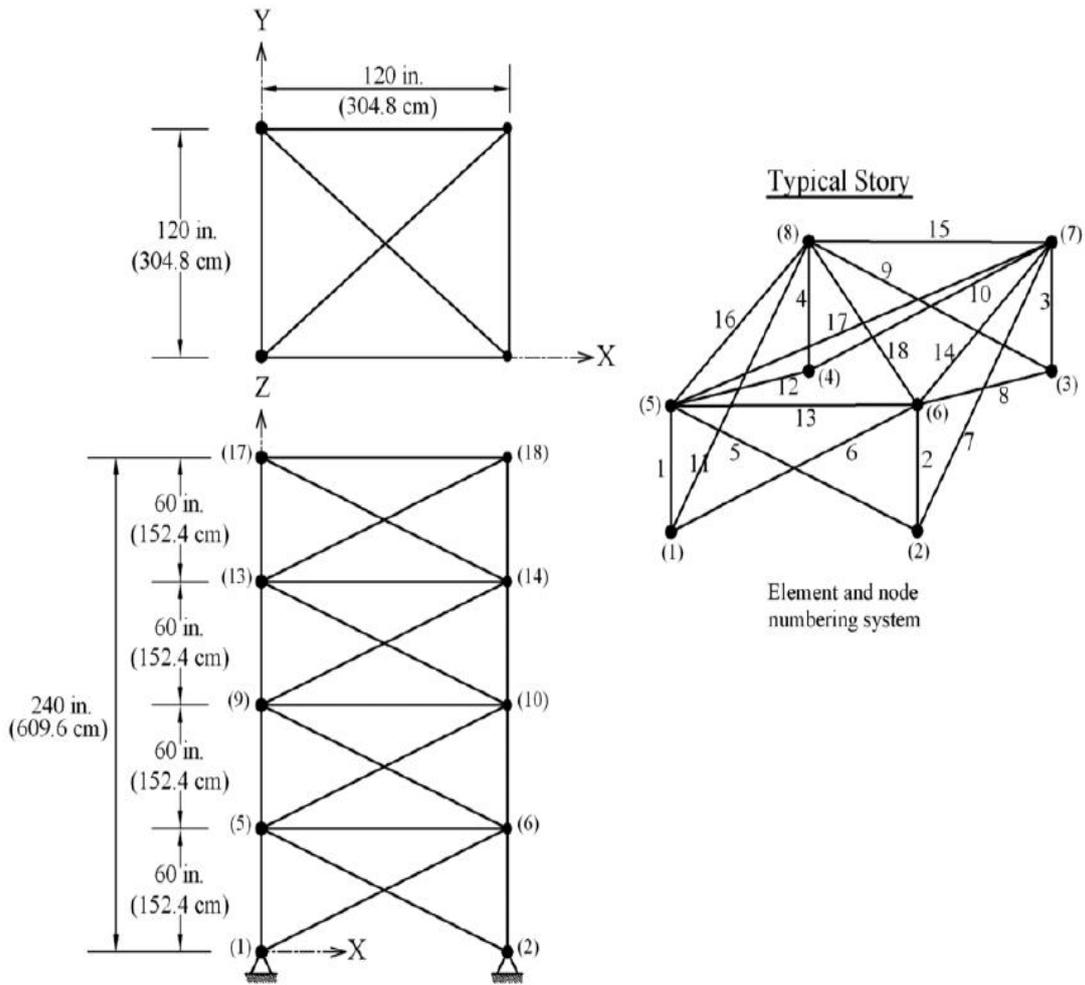


圖 4-3 空間 72 根桿件桁架結構

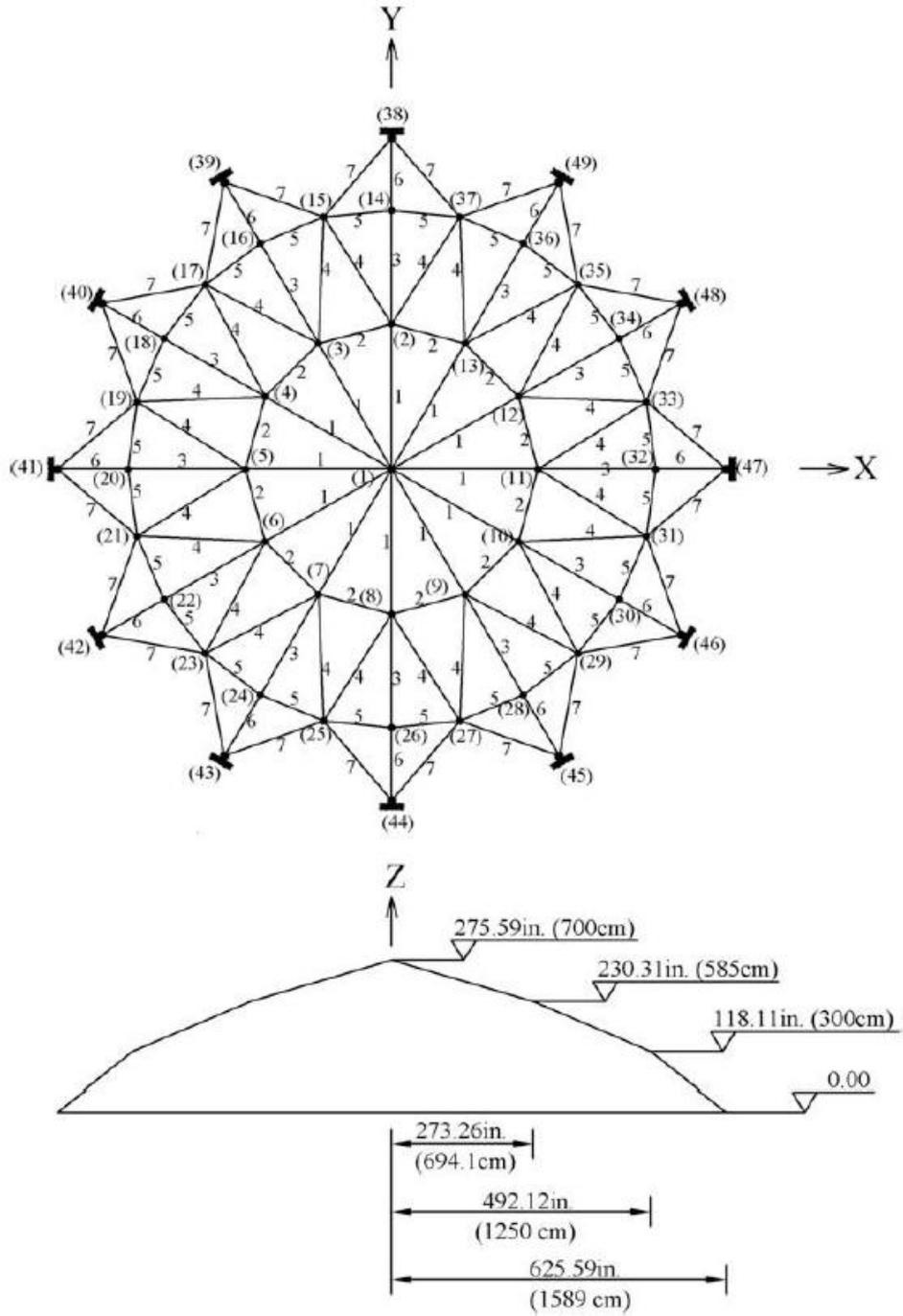


圖 4-4 空間 120 根桿件桁架結構

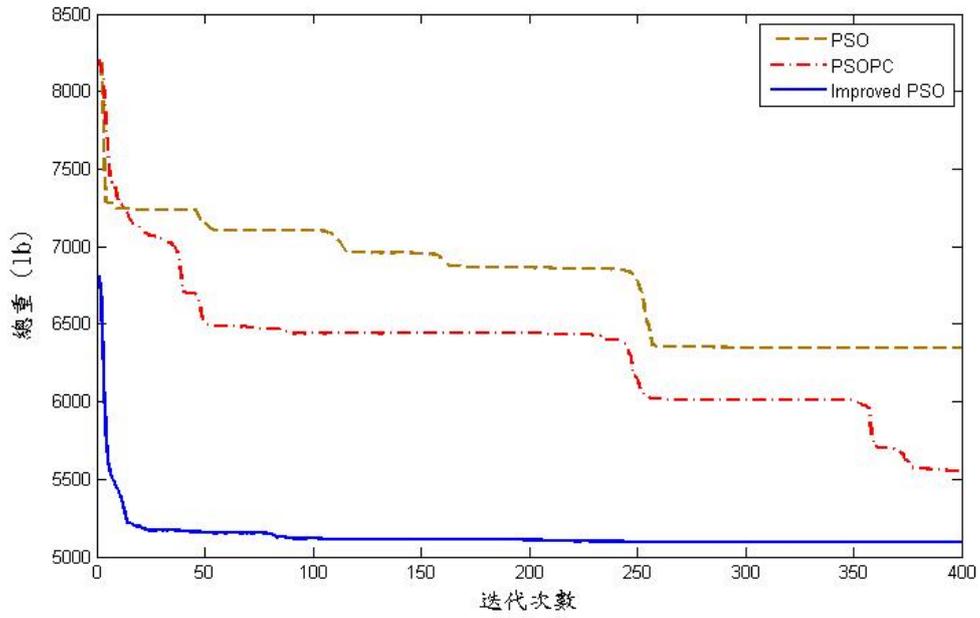


圖 4-5 例一，不同的粒子群優化演算法收斂結果

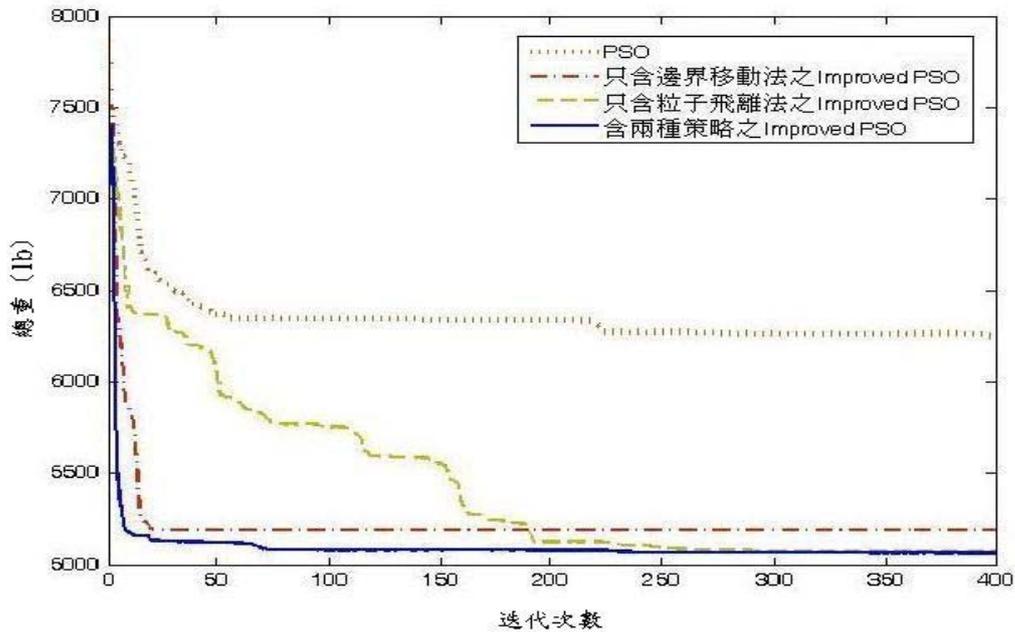


圖 4-6 例一，三種改良式粒子群優化演算法之比較

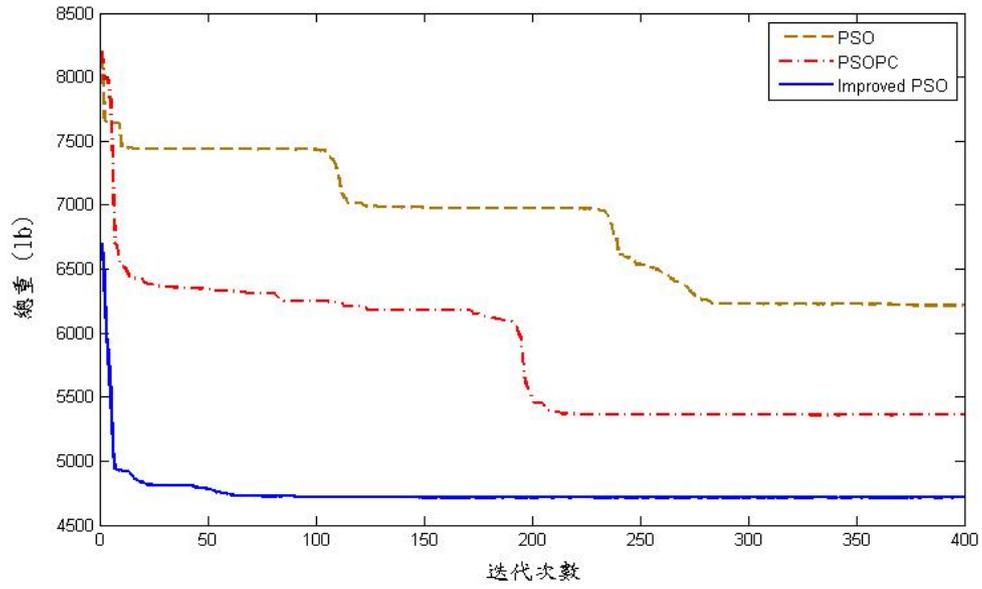


圖 4-7 例二，不同的粒子群優化演算法收斂結果

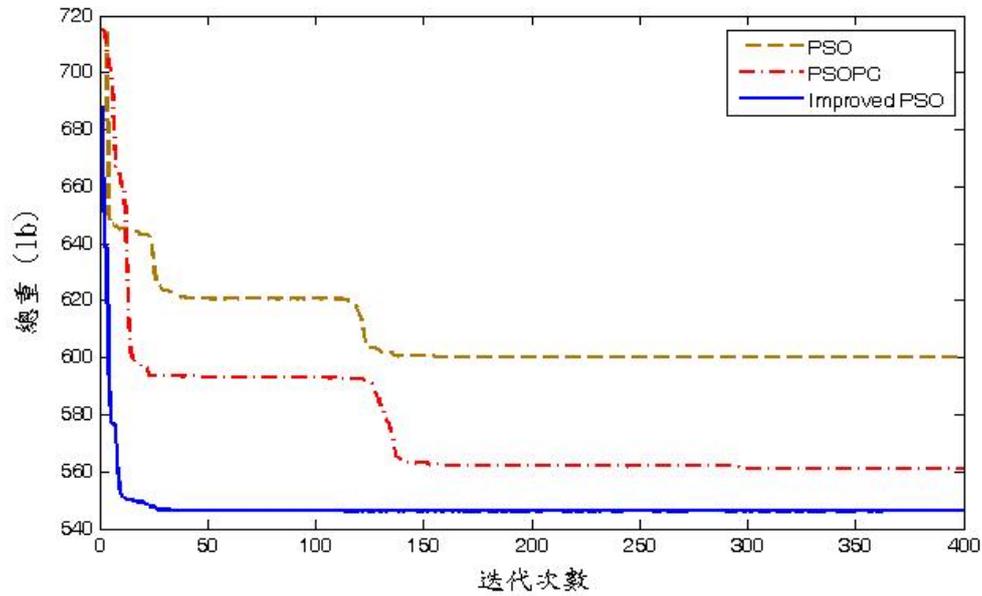


圖 4-8 例三，不同的粒子群優化演算法收斂結果

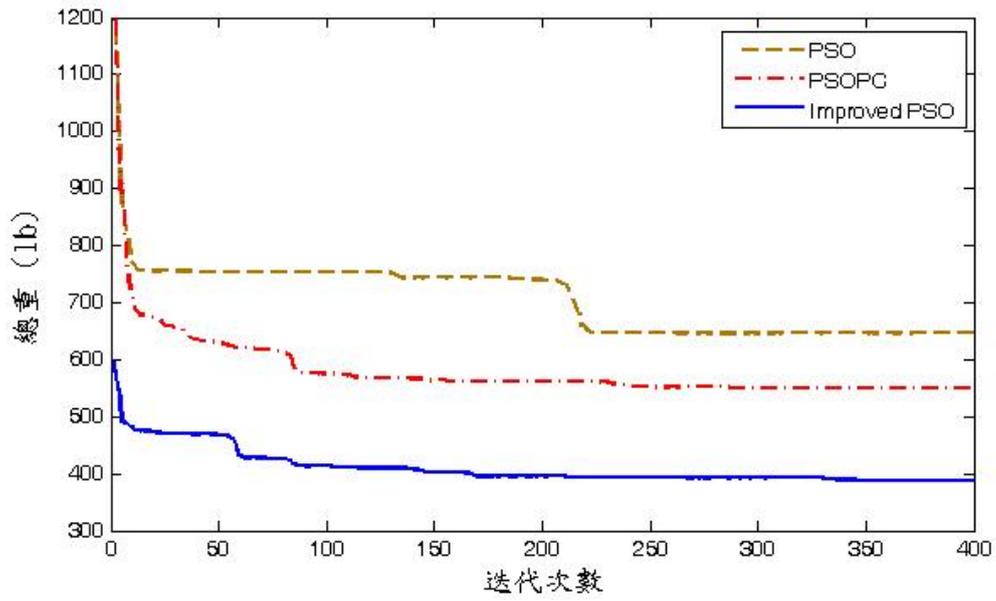


圖 4-9 例四，不同的粒子群優化演算法收斂結果

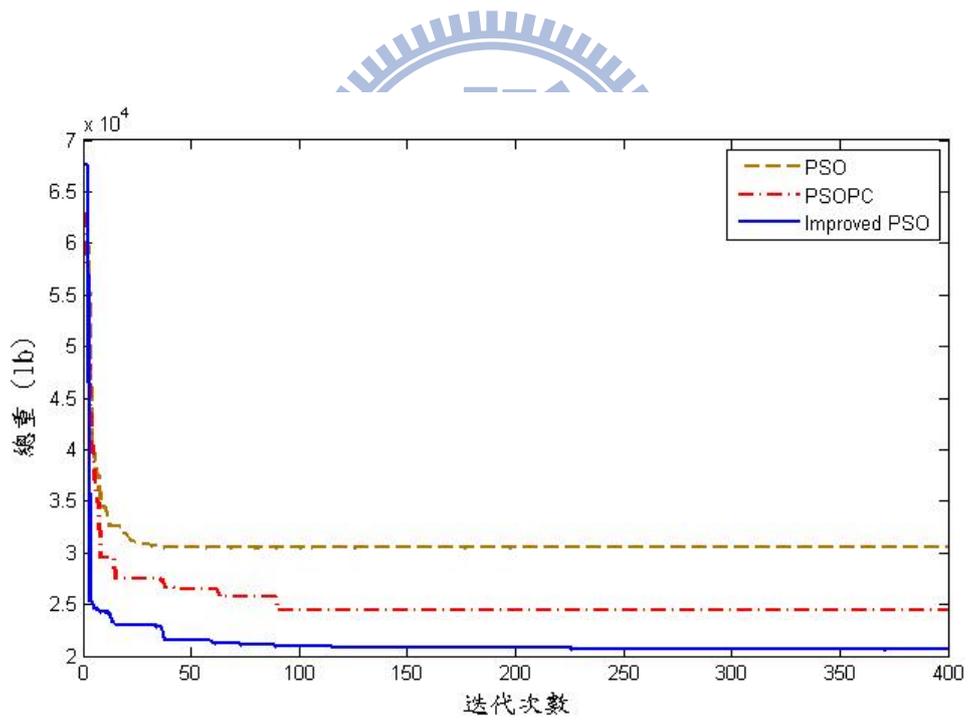


圖 4-10 例五，不同的粒子群優化演算法收斂結果