

## 模擬退火於單頻道廣播傳輸架構之平行化

洪宗貝<sup>1</sup> 溫添成<sup>2</sup> 游坤明<sup>2</sup> 王慶堯<sup>1</sup>

<sup>1</sup>義守大學資訊工程研究所

<sup>2</sup>中華大學資訊工程研究所

tphong@csa500.isu.edu.tw

### 摘要

模擬退火法是常用於解決最佳化問題中的其中一種近似解法。模擬退火法就是模擬化學結晶過程中，先加至高溫，再慢慢降溫以求得穩定狀態，而將這個過程的觀念，運用於電腦演算法中，以求得近似最佳解。然而，模擬退火法必須要求溫度參數必需降得非常慢，以求得較好的近似結果，但也導致一些問題在應用這種方法時，求解的過程中花費的時間相對較冗長。因此，在這篇論文中，我們嘗試利用廣播傳輸模型平行化模擬退火演算法，以加快其執行速度。最後，我們以旅行者問題 (Travelling Salesman Problem) 作實驗，來驗證我們所提出之平行方法的有效性。

關鍵詞：模擬退火法、平行化、廣播傳輸模式、狀態、接受機率。

### 1. 緒論

日常生活中有許多的問題是最佳化的問題，但是當用電腦來解決這些問題時，常遇到在求最佳化解時，其時間複雜度相當高。譬如說圖派決策問題 (Clique Decision Problem)，節點考慮問題(Node Consideration Problem)，旅行推銷員問題 (Travelling Salesman Problem) [1][4]，共用部份運算式的代碼產生問題(Code Generation with Common Sub-expressions Problem)等均是NP-hard的問題，這些問題都將隨所解問題變數的稍微增大，而導致非常多的執行時間。因此，遇到這些問題時，常是退而求其次，設計出好的啟示演算法 (heuristic algorithm) 以用較低的時間複雜度，來求出好的近似最佳解。

在目前已提出的啟示演算法中，模擬退火法 (simulated annealing) [5, 6] 是相當著名的一種搜尋方法。其適用的範圍相當廣，因此很多問題用模擬退火法常能求到好的近似最佳解，而且和求最佳解的方法相比較能有較低的時間複雜度。

模擬退火法主要是模擬化學結晶過程中，先加至高溫，再慢慢降溫的過程，而將這個過程運用於電腦演算法中，以求得近似最佳解。然而模擬退火法必須要求每次降溫間隔非常小，才有較好的結果，因此溫度參數降得非常慢，也使得其計算時間隨溫度降低的間距縮小而變大。

近年來，由於硬體技術的進步以及硬體價格的滑落，使得對於利用平行技術來解決複雜的問題相當可行。而單頻道廣播模式在平行架構中是相當簡單也相當容易建構的一種方法，因此在本篇論文中，我們嘗試在廣播傳輸模式 (broadcast communication model) 下提出一新的平行模擬退火演算法，以加快求解的速度。我們希望我們

所提的平行演算法，與傳統模擬退火法相比，能對原問題有類似的近似結果，但卻有較短的執行時間。最後，我們以旅行推銷員問題 (Travelling Salesman Problem) 作實驗，來驗證我們所提平行方法的效果。

本篇論文的編排如下：在第二節中我們將對模擬退火演算法，做一簡單的複習，包括基本專有名詞，以及以後將用到的基本演算法。第三節中，我們則簡單介紹將在第四節運用到的單頻道廣播通訊結構基本概念。第四節則闡述我們所提出之平行模擬退火之相關觀念及演算法。而在第五節中，我們舉一實例，利用所提的平行模擬退火演算法，來求解旅行推銷員問題。在第六節中，我們將設計程式並實驗比較傳統模擬退火演算法與我們所提及的平行模擬退火演算法的優劣。最後，在第七節中我們則針對本研究做一結論。

### 2. 模擬退火演算法回顧

在傳統模擬退火演算法中，每一次降溫，便會根據事先訂定的規則從舊狀態 (old state) 更改至一個新狀態 (new state)。若產生新狀態的能量較原狀態低，則以新狀態取代舊狀態；若產生的新狀態 (new state) 較舊狀態 (old state) 的能量大，則以一機率值  $e^{-\frac{\Delta E}{T}}$  來判定是否接受，其中  $T$  為目前的溫度參數，而  $\Delta E$  為新舊系統間的能量差。因此若一新狀態不被接受，則所花的時間白白浪費，無法得到任何的效果。因此在本篇論文中，我們將利用廣播傳輸模式 (broadcast communication model)，以快速找到合適並被接受的狀態，以節省不必要的檢查。

首先，我們對幾個模擬退火法的專有名詞包括狀態 (state)、能量 (energy)、接受機率 (acceptance probability)、降溫公式 (decreasing temperature formula)、全域最佳解 (global optimal)、區域最佳化 (local optimal) 作一簡單的介紹。接著我們將介紹基本的模擬退火演算法。

#### 2.1 名詞介紹

##### (1) 狀態 (state)

在模擬退火中，每一合法的解 (不一定是最佳解) 被稱為狀態。以旅行推銷員問題為例，一種可能的合法走法，就稱為一個狀態。例如有 7 個城市 (編號為 1~7)，其中  $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7$  的走法，稱為一個狀態。所有可能的狀態集合起來即稱為狀態空間。在模擬退火法中，每一次動作，將產生一個新的狀態，再判斷此新狀態是否取代舊狀態。

##### (2) 能量 (energy)

模擬退火法必須定義一個能量函數以評估每一個狀態的好壞。通常較低的能量代表較穩定的狀態，也就是較

好的解。因此，對於求最大值的問題，如求總利益最大的問題，則常在能量函數中加一負號，以使得能量最低的解最好。而在旅行推銷員問題中，我們可定每一種走法的總長度為能量。例如上述  $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7$  的走法中，假若總長度為36，則此走法的能量為36，亦稱此狀態的能量為36。

### (3) 接受機率 (acceptance probability)

在模擬退火法中，剛開始時，給定的溫度參數都設在某一起始值，然後逐漸降低溫度值。每一次的動作中，便產生一個新的狀態，若產生的新狀態能量較原來狀態的能量為低，則新狀態將一定取代舊狀態，在此情況下我們稱其接受機率 = 1。反之，若新產生的狀態能量，較原來的狀態的能量高，則我們以一接受機率  $e^{-\frac{\Delta E}{T}}$  來判斷是否接受此新狀態，其中，T為當時的溫度參數，而 $\Delta E$ 為新舊狀態間的能量差。接受機率  $e^{-\frac{\Delta E}{T}}$  主要是應用在傳統化學結晶的退火過程中求化學狀態的變化機率。在此模擬退火法中，則被用來跳離區域最佳解以求得近似最佳解。

[例一]: 假設目前的溫度  $T=98$ 。

目前的狀態:

$1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7$ , 能量=54。

產生新的狀態:

$3 \rightarrow 2 \rightarrow 1 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7$ , 能量=80。

所以其接受機率  $e^{-\frac{\Delta E}{T}} = e^{-\frac{80-54}{98}} = 0.738$ 。

### (4) 降溫公式

在前人的研究中，模擬退火法中有四個常用的降溫公式被採用。列舉如下:

公式一:  $T=T-K$

公式二:  $T=T*r$

公式三:  $T=T*e^{-0.1k}$

公式四:  $T=T/\log(k+1)$ 。

其中K為調溫的參數。以公式一:  $T=T-K$ 來說，表示在每個模擬退火的步驟中，都是降低K度。

### (5) 區域最佳化 (local optimal) 及廣域最佳化 (global optimal)

在化學結晶領域中的退火方式，通常先加至高溫，主要目的是在打破落在區域最佳化中的陷阱，而能在廣域中移動，之後再慢慢降溫，以求得最小能量的化學結晶格子，此乃慢慢地往較有可能的區域中尋找。透過全域及區域求解的相互配合，希望能將近似最佳解求出。在電腦模擬退火演算法中，也是先加至高溫，此時接受函數  $e^{-\frac{\Delta E}{T}}$  中，由於T是很大值，因此接受機率值多接近1。亦即在高溫時，接受機率 $\approx 1$ 。因此，在高溫時，狀態變換在各處跳來跳去，當慢慢降溫時，接受機率  $\frac{1}{T}$  由於T越來越小，接受機率也越來越小，變換狀態越來越不易。而在前人文獻中記載[6]，若是每次降溫間隔小，可達到接近最佳化的效果，亦即接近廣域最佳化。

### 2.2. 模擬退火演算法

根據上述的觀念，模擬退火演算法敘述如下:

模擬退火演算法:

步驟一: 設定起始溫度 T及終止溫度T' ;

步驟二: 設定起始狀態 S ;

步驟三: 當目前溫度T大於或等於終止溫度T'時:

(a) 透過特定的狀態產生方法(根據題目而定)產生一個新的狀態 S', 並計算其能量;

(b) 計算新舊狀態間的能量差  $\Delta E$ , 即:

$\Delta E = \text{energy}(S') - \text{energy}(S)$  ;

(c) 假設 $\Delta E$ 小於或等於 0 , 則以新狀態 S' 為目前狀態;

(d) 假設 $\Delta E$ 大於 0 , 則以一機率值  $e^{-\frac{\Delta E}{T}}$  來隨機判斷是否接受新狀態;

步驟四: 依照所選定的降溫公式降溫;

步驟五: 重複步驟三及步驟四直至低於終止溫度為止, 最後的狀態即為問題的近似解。

[例二]: 當在解TSP問題時，假設起始溫度為10，終止溫度為0，並且用公式1:  $T = T-2$  來降溫。城市個數為7，分別編號為1、2、3、4、5、6、7，以模擬退火法執行過程如下:

步驟一: 設定起始溫度  $T=10$  ;

步驟二: 隨機產生一個起始狀態7654321並計算其能量，假設能量為89;

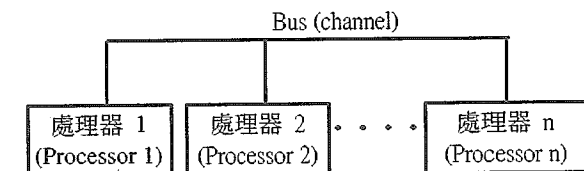
步驟三: 調換第4個和第5個城市而產生一個新狀態: 7645321，假設能量=72，因為  $72 < 89$ ，所以接受機率 = 1。

步驟四:  $T = 10 - 2 = 8$ ，溫度降為 8。

步驟五: 重複步驟三及步驟四直至溫度降至 0 為止，輸出最後結果。

### 3. 單頻道廣播通訊模式(Single Channel Broadcast Communication Model)

在本篇論文中，我們採用單頻道廣播通訊模式[3]來平行化模擬退火演算法，該模式以圖(一)表示。



圖(一): 單頻道廣播通訊模式

此種模式含有一些處理元件 (processing element) 或處理器(processor), 這些處理元件共同分享一個共用的通訊頻道 (bus or channel), 彼此之間互相通訊。此模式採用時間槽 (time slot) 的觀念，每一個時間槽是代表傳播單一信息 (message) 的時間。所有的處理元件都嘗試以此共同的通訊頻道來送信息。而在同一時間中，只允許一個處理元件來廣播。當一個處理元件佔住此大家共同使用的通訊頻道，這一個處理元件就廣播其信息並告訴其它的處理元件已佔住此共同的通訊頻道。此時，若其它的處理元件要傳送信息時，必須偵測此一共同的通訊頻道是否也處於閒置的狀態。因此一個處理單元可將其信息放在公共的通訊路線上，其它的處理單元可至此通訊線路將信息拿走。如果在上一個時間槽中，沒有處理單元在廣播，則被傳送的信息是空的。如果兩個以上的處

理單元同時搶單一的通訊頻道時，衝突已經發生，此時則利用解決衝突的方法來解決（這不在本論文討論範圍中）。

在一廣播通訊模式下，時間複雜度是由所有計算時間、廣播時間以及解決衝突的時間三者之和所算出。我們可以如下的公式表達出：

$$T(N) = B(N) * \gamma + C(N) * \tau + R(N)$$

其中N表示所用處理器的個數，T(N)表示平行演算法的全部執行時間，B(N)表示廣播的次數，C(N)表示計算的次數，R(N)表示解決衝突的總時間， $\gamma$ 表示廣播的單位時間， $\tau$ 表示計算的單位時間。假設當所有的處理單元完成了一個計算便緊接著一個廣播信息的動作，而且解決衝突的時間相當小，則T(N)可約略估計為O(B(N))。

#### 4. 平行模擬退火演算法

在本節中，我們首先提及採用平行模擬退火演算法的基本想法，接著我們採用單頻道廣播架構的方式，來規劃平行模擬退火演算法。

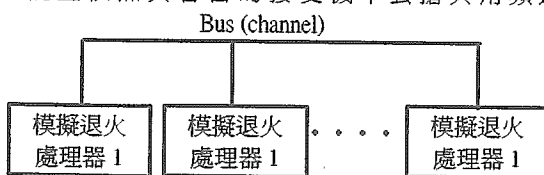
##### 4.1 基本想法

在模擬退火演算法中，當在高溫中，由於接受機率通常很高，因此此時較強調廣域間的跳躍，所以若能以平行方式來增加廣域區間內不同區域出現的可能性，將可以較大的溫度間隔來降溫。而在低溫時，由於接受機率跟著變低，因此新狀態要取代舊狀態的機率逐漸變低，因而若新狀態無法取代舊狀態，則產生和檢查此新狀態時間都浪費了，因此若能利用平行化的方式，快速找出可取代舊狀態的新狀態，將可加速模擬退火的求解過程。合併上述兩個想法，我們利用前述所提的單頻道廣播傳輸架構，將相鄰近時間的N個降溫步驟，合併為一個步驟由N個處理器來執行，並由幾個處理器當中要取代舊狀態的新狀態來爭取廣播權，並傳遞至其它處理器，以避免不必要的檢查動作。例如在極端的情況下，假如N個處理器中只有一個處理器其狀態須要更新，則利用此平行架構，在一個單位時間內即可變換狀態，然而如果在傳統模擬退火演算法中，則平均需要N/2個單位時間才可找出變換的狀態出來。根據上述的想法，我們希望所提的平行模擬退火演算法可在同一時間內做較大的降溫動作，且和傳統模擬退火演算法相比，會有較快收斂速度及差不多效果的解。

##### 4.2 平行模擬退火模式

在本節中的平行模擬退火模式，採取第三節中單頻道廣播傳輸結構的觀念，其基本運作模式如下圖(二)所示。

上圖中，每一模擬退火處理器，依照其變換狀態的方法來產生一個新狀態，並算出這一個新狀態的機率，以判斷是否接受此一新狀態。此外在每一次降溫的動作下，同一時間中每一模擬退火處理器所產生個別的新狀態及能量依照其各自的接受機率去搶共用頻道



圖(二)：平行模擬退火模式

(bus)，搶到公共頻道的處理器，將其狀態廣播給其它處理器。在每一時間槽內，最多只有一個模擬退火處理器可搶到公共頻道，並傳播其狀態給其他處理器，而各個處理器便以此狀態為其新狀態，再做下一個降溫的相關動作，直到終止溫度時才停止，所得到的最後狀態，即為所解問題的近似解。根據上述的想法，其演算法敘述如下。

##### 平行模擬退火演算法：

輸入：起始溫度、終止溫度、N個處理器；

輸出：最終狀態及其能量；

步驟一：設定起始溫度  $T$ ，終止溫度  $T'$ ；

步驟二：設定起始狀態  $S$ ，並放置於每個處理器當中；

步驟三：當目前溫度  $T$  大於或等於終止溫度  $T'$  時，在每個處理器  $P_i$  中，執行下列子步驟：

(a) 根據產生新狀態之方法產生一個新的狀態  $S_i'$ ，並計算其能量  $energy(S_i')$ ；

(b) 計算新舊狀態間的能量差  $\Delta E_i$ ，即：

$$\Delta E_i = energy(S_i') - energy(S)；$$

(c) 假設  $\Delta E_i$  小於或等於 0，則  $P_i$  競爭公用頻道廣播權；

(d) 假設  $\Delta E_i$  大於 0，則以一機率值  $e^{-\frac{\Delta E_i}{T}}$  來隨機判斷是否競爭公用頻道廣播權；

步驟四：獲得廣播權的處理器  $P_j$ ，將其新狀態  $S_j$  送至其它處理器，並設為其狀態  $S$ ；

步驟五：依照所選定的降溫公式作  $N$  次降溫；

步驟六：重覆步驟三至五直至溫度小於或等於終止溫度；

步驟七：輸出最終狀態及其能量為所求問題的近似解。

#### 5. 舉例說明

以下我們以旅行推銷員問題(travelling salesman problem)為例，來說明平行模擬退火演算法。假設有7個城市(編號為0~6)，兩兩城市間的距離用一矩陣表示如下表(一)所示。

城市編號	0	1	2	3	4	5	6
0	0	26	60	56	65	49	67
1	26	0	1	62	21	26	14
2	60	1	0	37	82	24	72
3	56	62	37	0	43	73	6
4	65	21	82	43	0	44	87
5	49	26	24	73	44	0	74
6	67	14	72	6	87	74	0

表(一)：城市間相互距離表示矩陣

上述的矩陣表示法，表示城市0與城市1之間的距離是26，城市0與城市2之間的距離是60，其餘以此類推。在模擬退火法中，如何從舊狀態中改變而產生一新狀態的方法是相當重要，在此例子，我們採用江和金[6]針對旅行推銷員問題所提出的方法來產生新狀態，其方法說明如下：

江和金狀態產生方法：

輸入：目前的狀態  $S$  及位置  $f$ ;

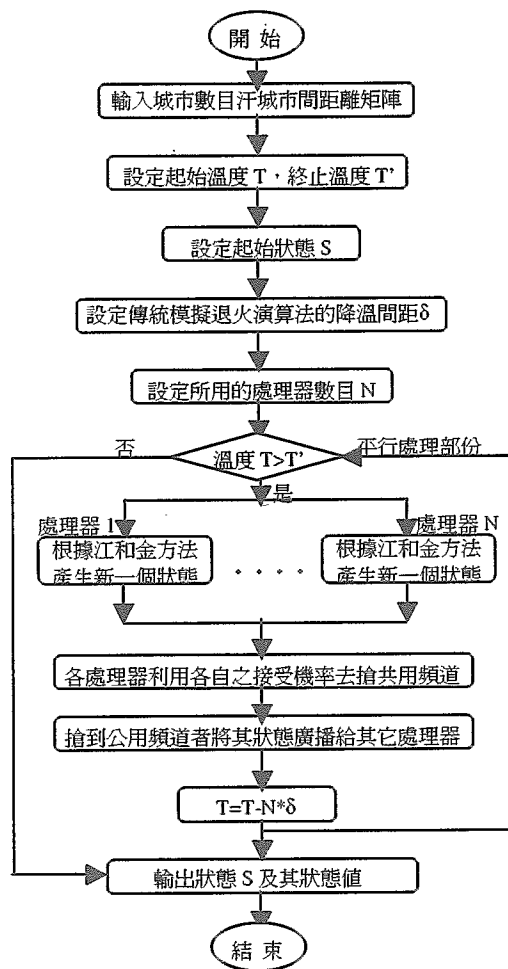
輸出：新狀態  $S'$ ;

步驟一：隨機產生一個指標  $f$  代表第  $f$  個城市  
 $0 \leq f \leq n-1$

步驟二：產生一個隨機變數值  $r$ ,  $0 \leq r \leq n$ ;

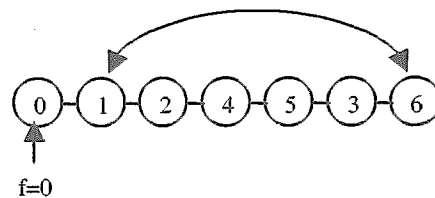
步驟三：對於  $i=0$  到  $\lfloor r/2 \rfloor$ , 將節點編號為  
 $((f - \lfloor (r-1)/2 \rfloor + i) \bmod n)$  和另一節點編號為  
 $((f + \lfloor (r-1)/2 \rfloor - i) \bmod n)$  對調, 即為新狀態  $S'$ ;

在此例中, 我們以3個平行處理器來說明, 而  $n$  表示為城市數目之總數。以等間距降溫的平行模擬退火演算法來解決旅行推銷員問題的流程图, 如圖(三)所示。



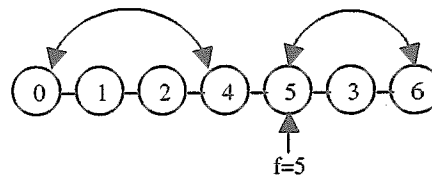
圖(三)：平行模擬退火演算法流程图

假設起始溫度為20, 終止溫度為0, 傳統的降溫間距  $\delta=2$ , 則上述之平行模擬退火演算法步驟進行如下。假設起始狀態  $S$  為0124536則可算出其能量為299, 將其分別放入3個處理器; 接著此3個處理器分別產生下一個狀態  $S'$ , 假設處理器 P1 所隨機產生之  $f=0$ 、 $r=3$ , 則利用江和金的方法, 所產生的新狀態之轉換過程如圖(四)所示。



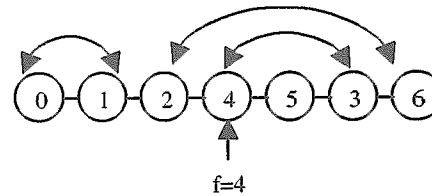
圖(四)： $f=0, r=3$ 之狀態轉換過程

因此其產生的新狀態為 0624531, 其能量 = 426。因為  $426 > 299$ , 所以其接受機率 =  $e^{-\frac{426-299}{20}} = 0.00166$ ; 同理處理器 P2, 假設其  $f=5$ 、 $r=5$ , 則其狀態轉換過程如圖(五)所示。



圖(五)： $f=5, r=5$ 之狀態轉換過程

產生的新狀態為 4120635, 其能量 = 272, 因為  $272 < 299$ , 其接受機率 = 1; 處理器 P3, 假設其  $f=4$ 、 $r=7$ , 轉換過程如圖(六)所示。



圖(六)： $f=4, r=7$ 之狀態轉換過程

產生的新狀態為 1063542, 其能量 = 299。因為  $299=299$ , 其接受機率 = 1。因此3個處理器分別產生新狀態, 並且各自依其接受機率來搶公用頻道, 此作法主要是由各處理器各自產生一隨機變數介於  $[0..1]$  之間, 假設此變數的值小於其接受機率, 則此處理器便去競爭廣播權, 否則不競爭。假設在此例中, 第三個處理器搶到公用頻道, 則將其狀態 1063542 廣播出去, 而每一其它處理器接到後, 則利用此狀態來更新其值。

接著,  $T = 20 - 3 * 2 = 14$ , 返回步驟二執行。然因為  $14 > 0$ , 因此上述步驟繼續重覆執行直至溫度降到 0 以下為止, 此時表示終止溫度已經到達, 所以輸出最終狀態。

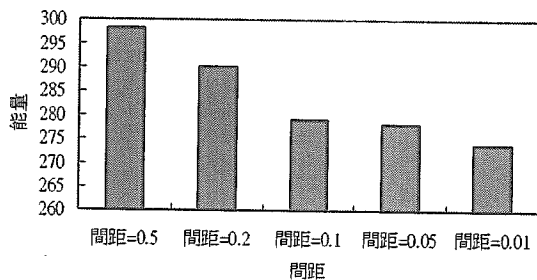
## 6. 程式設計與實驗結果

在本節中我們將依照上述的平行模擬退火演算法, 設計實驗程式來解旅行推銷員問題以驗證我們所提方法的有效性。我們分別採用公式一:  $T' = T - K$ , 公式二:  $T = T * r$ , 公式三:  $T = T * \exp^{-0.01K}$ , 公式四:  $T = T / \log(k+1)$ , 規劃 4 組不同降溫公式的實驗。我們做此實驗的目的, 是比較模擬退火程式與平行模擬退火程式效果及時間上的差異。我們的實驗環境是在 PC-586 上以 C 語言來撰寫並執行, 我們先以 9 個城市為實驗, 隨機產生了一組城市與城市之間的距離矩陣, 如表(二)所示。

城市編號	0	1	2	3	4	5	6	7	8
0	0	40	20	50	49	53	74	41	68
1	40	0	19	77	48	9	5	40	25
2	20	19	0	54	44	3	57	20	40
3	0	77	54	0	31	61	77	48	53
4	49	48	44	31	0	40	68	78	43
5	53	9	3	61	40	0	80	36	47
6	74	5	57	77	68	80	0	71	73
7	41	40	20	48	78	36	71	0	45
8	68	25	40	53	43	47	73	45	0

表(二)：城市間相互距離表示矩陣

我們首先分別對四個不同的降溫公式，不同的降溫參數，及不同的終止溫度來做實驗，以解釋模擬退火中降溫間距對解答的影響。針對降溫公式一： $T' = T - k$ ，溫度從950降至0度的500次實驗平均結果如圖(七)所示。

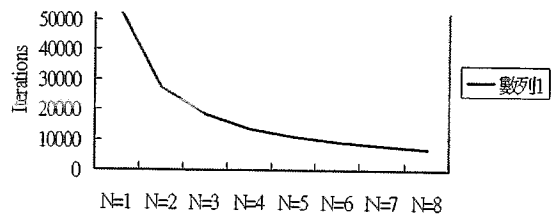
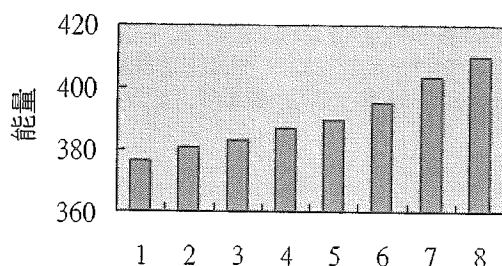


圖(七)：針對降溫公式一之實驗結果

我們由上面的圖示中可以發現，當降溫間距越小時，其所獲得的解越佳，對其它降溫公式所得的實驗結果亦類似，因此我們可知不管使用哪種降溫公式，溫度降的越慢，效果越好，但其執行時間也越久。

接著我們實驗比較傳統模擬退火程式與平行模擬退火程式之效果，在實驗中，非平行模擬退火程式與平行模擬退火程式皆採用相同的城市個數與相同的城市與城市之間的距離，亦即一個相同的距離矩陣分別輸入到傳統模擬退火演算法與平行模擬退火演算法。我們以20個城市為實驗，處理器個數設為1至8個，在PC上模擬，而降溫公式為  $T = T - N * \delta$ ， $\delta$ 設為0.01，其所需要的執行次數如圖(八)所示。其收斂時的能量如圖(九)所示。

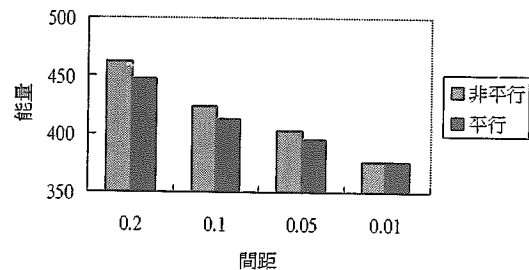
圖(八)：針對不同的處理器個數在間距為0.01之執行次數



圖(九)：針對不同的處理器個數收斂時的能量實驗結果

由圖(八)及圖(九)可知，根據我們的平行模擬退火演算法，當處理器個數越多時，所須的執行次數越少，且效果和非平行模擬退火方法比較(即處理器只有一個)，誤差在很小的比例以內。因此利用平行方法可大幅降低執行時間且可獲得不錯效果的解，而可達到時間與準度之間的折衷。

接著我們以同樣的執行次數來比較平行與非平行方法的效果，我們針對不同的降溫間距，在平行程式部份主要模擬8個處理器，因此在同樣執行次數下，其間距可縮小為傳統模擬退火演算法的1/8，500次實驗平均結果如圖(十)所示。



圖(十)：針對傳統模擬退火演算法與平行模擬退火演算法比較之實驗結果

在圖(十)中的間距是以傳統的模擬退火演算法所用的間距表示，而平行退火演算法為其1/8。根據圖(十)的結果可知在相同的執行次數下，利用平行模擬退火演算法可得出比傳統模擬退火演算法更好的結果。

## 7. 結論

本篇論文主要的目的是嘗試將模擬退火演算法平行化，同時以實際程式實驗，比較了非平行方法與平行方法的效果。我們將平行模擬退火演算法建構在一個相當簡單的平行模式，單頻道廣播模式上，並將傳統模擬退火演算法中，相鄰近時間的  $N$  個降溫步驟，合併為一個步驟由  $N$  個處理器來執行，以增加廣域的搜尋及避免不必要的檢查時間。我們亦以江和金產生狀態方法來解決旅遊者問題，藉由於參數的適當調整，成功的使得平行模擬退火演算法的輸出結果 優於傳統非平行模擬退火演算法。

## 參考文獻

- [1] B. Angeniol, G.Vaubois and J. Texier "Self-Organizing Feature Maps and the Travelling Salesman Problem," *Neural Networks*, Vol. 1, pp 289-293, 1988.
- [2] T. P. Hong and S. S. Tseng, "Parallel perceptron learning on a single-channel broadcast communication model," *Parallel Computing*, Vol. 18, No. 2, 1992, pp. 133-148.
- [3] T. P. Hong and J. J. Lee, "A nearly optima back-propagation learning algorithm on a bus-based architecture," *Parallel Processing Letters*, Vol. 8, No. 3,

- 1998, pp. 297-306.
- [4] V. Cerny "Thermo-dynamical approach to the traveling salesman problem: an efficient simulation algorithm." *Journal of Optimal Theory with Applications*, Vol. 45, 1985.
- [5] C. S. Jeong and M. H. Kim. "Fast parallel simulated annealing for traveling salesman problem on SIMD machines with linear interconnections." *Parallel Computing*, 1991.
- [6] S. Kirkpatrick, C. D. Gelatt and M. P. Vecchi, "Optimization by simulated annealing", *Science*, Vol. 220, pp. 671-680, 1983.