

群體智能最佳化演算法之介紹

白子易

朝陽科技大學環境工程與管理系教授

摘 要

群體智能最佳化演算法 (swarm intelligence optimization algorithm, SIOP) 是近年來逐漸獲得重視的研究領域。本文擬介紹幾種常用的 SIOP，包括粒子群最佳化演算法 (particle swarm optimization algorithm, PSO)、螞蟻演算法 (ant algorithm, AA)、克隆選擇演算法 (clonal selection algorithm, CSA)、細菌覓食演算法 (bacterial foraging algorithm, BFA)等典型 SIOP。探討其發展、原理，及其在環境相關領域的應用，以供研究人員參考。

關鍵字：群體智能最佳化演算法、粒子群最佳化演算法、螞蟻演算法、克隆選擇演算法、細菌覓食演算法

一、前 言

人工智慧 (Artificial Intelligence, AI) 於 1956 年問世以來，已經獲得許多重大成就，並於許多領域進行應用。但是，AI 與真正的智能之間的距離還相當遠，因此，研究人員認為，應該重新思考從「物理學」為出發點的 AI，創新的 AI 應該從「生物學」為出發點。事實上，AI 研究一直關注著生命現象和生物智慧行為，尤其是近年來，AI 的重要成果與生物有密不可分的關係，無論是從結構模擬，還是從功能模擬，或以生物進化為機制，都存在生物模擬的痕跡。因此，模仿生物的智慧行為、智慧機制，是 AI 研究領域的重點。

20 世紀 60 年代至今，研究人員不斷開發模擬某種特定自然現象或模擬某種生物系統特性的理論和技術，例如人工神經網絡 (artificial neural network, ANN)、模糊系統 (fuzzy system)、模擬退火演算法 (simulated annealing algorithm, SA)、進化算法等和自然現象有關的計算方法，不但促進智能計算、資訊技術、人工智能等發展，並且應用於程序控制、工程最佳化、機器學習、經濟預測等諸多領域。

在上述與自然現象有關的計算方法中，群體智能最佳化演算法 (swarm



intelligence optimization algorithm, SIOP) 是近年來逐漸獲得重視的研究領域。廣意的 SIOP 包括基因演算法 (genetic algorithm, GA)、粒子群最佳化演算法 (particle swarm optimization algorithm, PSO)、螞蟻演算法 (ant algorithm, AA)、克隆選擇演算法 (clonal selection algorithm, CSA)、細菌覓食演算法 (bacterial foraging algorithm, BFA)等典型 SIOP。

本文擬針對幾種常用的 SIOP，介紹其發展、原理，及其在環境管理、環境科學、環境工程等領域的應用，以供環境相關領域的研究人員參考。

二、群體智能最佳化演算法發展及原理

(一)螞蟻演算法 (AA)

AA 乃由義大利學者 Dorigo et al.於 1991 年提出⁽¹⁾，屬於群體啟發式搜尋法。AA 是模擬真實螞蟻的群體合作行為而發展的隨機搜尋算法，藉由解空間中的群體進化尋求最佳解。解空間中每個候選解根據累積的資訊不斷調整自身結構並且藉由資訊與其他候選解進行交互作用，以產生更好的解。

AA 起初只是隨機地選擇搜索路徑，隨著對解空間的了解，搜尋越來越具有規律性，並逐漸得到全局最佳解。AA 的搜尋機制主要包括下列各項⁽²⁾：

- 1.解空間中，每隻螞蟻可藉由信息素相互交流，信息素是群體中通信的機制。在行經的路徑上，螞蟻能分泌一種化學物質費洛蒙 Pheromone (亦即信息素)，藉由分泌費洛蒙形成信息素軌跡。螞蟻在移動過程中能夠感知費洛蒙的存在及其強度，並藉此引導本身的移動方向，使螞蟻傾向朝著該物質強度高的方向移動。螞蟻利用信息素軌跡找到其返回食物源 (或蟻穴) 的路徑。當螞蟻的其他同伴進行路徑選擇時，會依照路徑上的信息素進行選擇，此信息素成為螞蟻之間通信的媒介。

2.螞蟻的群體活動

每隻螞蟻在尋找食物源的時候只產生極小部分的貢獻，但整個蟻群的行为卻可展現具有尋找最短路徑的能力。大量螞蟻組成蟻群，蟻群的集體行為展現信息的正回饋現象：走過某一路徑上的螞蟻數目越多，信息素強度隨之增大，該路徑對後來的螞蟻就越有吸引力，即某隻螞蟻選擇某一條路徑的機率隨著之前選擇該路徑的螞蟻數量的增加而增大。而通過螞蟻數量較少的某些路徑，路徑上的信息就會隨時間演化而逐漸蒸發。此搜索路徑的過程稱之為螞蟻的自催化行為或正反饋機制。因此，藉由模擬此機制即可使 AA 的搜尋向量朝最佳解前進。

3. 利用當前信息進行路徑選擇的機率選擇法

AA 中，螞蟻從某個節點移動到另一個節點的求解方式，是利用機率選擇法進行的，機率選擇法只利用目前的信息預測未來的狀況，無法利用未來的信息，因此，該選擇法所使用的信息都是當前信息。

螞蟻演算法的特點如下⁽³⁾：

1. 優點

- (1) AA 基本上屬於模擬進化算法，其結合分布式計算、正回饋機制和貪婪式搜尋法，在搜尋過程中不容易陷入局部最佳解，即使在所定義的適應函數不連續、不規則或有噪度的狀況下，發現最佳解的機率也較其他方法大，同時貪婪式搜尋有助於快速找到可行解，縮短搜尋時間。
- (2) AA 採用自然進化機制表現複雜的現象，藉由信息素融合而不是個體之間的通信機制，使演算法的擴充性較好，能夠可靠迅速地解決問題。
- (3) AA 的並行性很高，適合於大量並行計算。

2. 缺點

- (1) AA 的搜尋時間通常較長。由於蟻群中個體隨機性移動，當群體規模較大時，就需要較長的搜尋時間以找到較好的路徑。
- (2) AA 在搜尋過程中容易產生停滯現象，表示當 AA 搜索到一定階段後，所有解趨向一致，已無法進一步搜索解空間，因此無法發現更好的解。

因此，在求解不同的最佳化問題時，就需設計不同的 AA，合理地選擇目標函數，信息更新和群體協調機制，盡量避免算法缺落於局部最佳解。

(二) 粒子群最佳化演算法

PSO 由 Eberhart and Kennedy 於 1995 年提出^(4,5)，屬進化計算技術 (evolutionary computation)。起源於對鳥群捕食的行為研究，與 GA 相似，是另一種選擇子代的最佳化工具。隨機初始化系統的解，藉由選擇子代搜尋最佳值。但未採用 GA 的交叉 (crossover) 以及突變 (mutation) 等演算，而是各個粒子追隨最佳解的粒子在解空間中進行搜尋。

PSO 中，每個最佳化問題的解都是搜尋空間中的一顆沒有質量和體積的微粒，並且擴展到 N 維空間。微粒 i 在 N 維空間裡的位置可表示為向量，每顆微粒子的飛行速度也可表示為向量。

由最佳化的函數決定所有微粒的適應值 (fitness)，每顆微粒飛翔的力向和距離則藉由速度決定。和飛翔的鳥一樣，每一顆微粒記錄本身截至目前發



現的最好位置和現在的位置，此過程可視為每一顆微粒本身的飛行經驗。此外，每顆粒子還記錄截至目前整個群體中所有微粒發現的最好位置，此過程可視為微粒群其他同伴的經驗，如同鳥群藉由特定方式告知同伴食物所在。PSO 就是藉由微粒[本身的經驗和同伴中最好的經驗決定在解空間的下一步移動。

PSO 首先隨機初始化微粒群（隨機解），然後微粒群就追隨在初始狀態的最佳微粒在解空間中搜尋，亦即藉由選擇子代尋找最佳解。假設 d 維搜尋空間中的第 i 個微粒的位置和速度分別為 $\mathbf{X}^i = (x_{i,1}, x_{i,2}, \dots, x_{i,d})$ 和 $\mathbf{V}^i = (v_{i,1}, v_{i,2}, \dots, v_{i,d})$ ，在選擇每一次子代中，每顆微粒藉由追蹤兩個最佳解更新本身，第一個就是微粒本身截至目前所搜尋到的最佳解，即個體極值 \mathbf{pbest} ， $\mathbf{P}^i = (p_{i,1}, p_{i,2}, \dots, p_{i,d})$ ；另一個是整個群體截至目前所搜尋到的最佳解，即全域最佳解 \mathbf{gbest} ， \mathbf{P}_g 。在搜尋到這兩個最佳解後，微粒根據以下的公式更新本身的速度和位置：

$$v_{i,j}(t+1) = wv_{i,j}(t) + c_1r_1[p_{i,j} - x_{i,j}(t)] + c_2r_2[p_{g,j} - x_{i,j}(t)]$$

$$x_{i,j}(t+1) = x_{i,j}(t) + v_{i,j}(t+1), j = 1, 2, \dots, d$$

其中 w 為慣性權因子， c_1 和 c_2 為正的學習因子， r_1 和 r_2 為 0 到 1 之間均勻分佈的隨機數。

PSO 的效能絕大部分取決於演算的控制參數，例如粒子數、最大速度、學習因子、慣性權重等，各項參數的選取原則如下⁽⁶⁾。

1. 粒子數：粒子數的多寡乃根據問題的複雜程度由演算人員自行決定。對於一般的最佳化問題，粒子數取 20 至 40 即可完全獲得良好的結果；對於比較複雜的問題或是特殊類型的問題，可以取 100 以上的粒子數。
2. 粒子的維度：此參數乃由最佳化問題決定，即問題解的維度（解的個數）；
3. 粒子的範圍：此參數亦由最佳化問題決定，亦即每個最佳解的範圍，因此每一維度可以設定不同的範圍；
4. 最大速度 \mathbf{Vmax} ：決定每個粒子在每一次迭代中最大的移動距離，通常設定為粒子的範圍寬度；
5. 學習因子：學習因子使微粒具有自我彙整以及向群體中最優秀個體學習的能力，藉以向最佳解靠近，通常 c_1 和 c_2 為 2；
6. 慣性權重：決定了對粒子當前速度繼承的多少，合適的選擇可以使粒子具有均衡的探索能力和開發能力。

(三)克隆選擇算法

克隆選擇算法 (clonal selection algorithm, CSA) 主要根據來自於免疫系統的克隆選擇理論，是一種模擬免疫系統學習過程的進化算法，由 de Castro and von Zuben (2002) 所提出⁽⁷⁾。CSA 相關計算主要由克隆選擇理論的基本過程所啟發，也就是受微環境下的“天擇說”所啟發，而發展 CSA 的基本流程。CSA 計算過程中主要呈現抗體在受外界抗原刺激後，通過自身的記憶細胞把抗原的特徵“一一記住”，當再次遇見相同的抗原時，記憶細胞就會進行克隆繁殖，達到足以殺死（或清除）入侵抗原的數量。細胞變異和新個體的替換，呈現抗體記憶細胞的多樣性，此顯示 CSA 的主要特徵。

克隆選擇必需經歷親和度成熟 (affinity maturation) 的過程，亦即，在克隆選擇機制的作用下，某個對抗原親和力較低的個體經歷增殖複製和突變過程後，因親和力逐漸增大而慢慢轉變成熟的過程。超突變 (hypermutation)、受體編輯 (receptor editing) 是親和力成熟過程中的重要組成，其保障基本的多樣性。超突變是指在複製過程中對受體編輯時，部分編碼發生突變而生成新細胞群體的過程。受體編輯可進行更大範圍的搜索，可搜尋到較佳的抗體，提供消除局部極值的能力。

CSA 的流程及演算法如下⁽⁸⁾：

1. 在解空間生成候選解集合 P ， P 由記憶單元 (M) 和保留種群 (P_r) 組成，即 $P = P_r + M$ ；
2. 根據親和力測量，選擇 n 個個體 (P_n)；
3. 複製（克隆）種群中此 n 個最好的個體，生成克隆臨時種群 (C)，克隆規模和抗體-抗原的親和力成正比；
4. 對克隆臨時種群進行超突變，此處，超突變和抗體-抗原的親和度相對應，因此獲得突變後的抗體群 (C^*)；
5. 從 C^* 中重新選擇改良的個體組成記憶單元 M ， P 中的某些個體也被 C^* 中其他改良的個體所取代。
6. 利用新產生的抗體代替 d 個舊抗體 N_d (引入多樣性)，親和度低的抗體更容易被取代。

CSA 與一般 GA 相較之下，差異在於：首先，將以機率為基礎的賭博輪盤選擇改變為以抗體-抗原親和度（適應度）為基礎的比例選擇；其次，建構記憶單元，將 GA 記憶單一個最佳個體轉變為記憶一個最佳解的群體；最後，藉由新舊抗體的替代，增加種群多樣性。



(四)細菌覓食演算法

BFA 亦稱為細菌覓食最佳化算法 (bacterial foraging optimization algorithm, BFO or BFOA)，由 Passino (2002) 所提出⁽⁹⁾。BFA 是根據 *E. coli* 大腸桿菌在人體腸道內吞噬食物的行為，所發展的一種新型仿生算法。BFA 因具有 SIOP 並行搜索、易跳出局部極小值等優點，成為生物啟發式計算研究領域的焦點之一。

細菌覓食算法模仿大腸杆菌在人體腸道內覓食行為，屬於仿生最佳化算法。在 BFA 模型中，最佳化問題的解對應成解空間中細菌的狀態，即最佳化函數適應值。BFA 算法包括趨化 (chemotaxis)、複製 (reproduction) 和消散 (elimination- dispersal) 等 3 個步驟，簡述如下。

- 1.細菌朝向營養豐富區域聚集的行為稱為趨化。在趨化過程中，細菌運動模式包括滾動 (tumble) 和游動 (run, swim)。滾動的定義為細菌向任意方向移動單位步長。當細菌完成一次滾動後，若得到較佳的適應值，細菌將沿同一方向繼續移動若干步長，直至適應值無法再改善，或達到預定的移動步數臨界值為止。此過程定義為游動。
- 2.達到臨界趨化次數，亦即當細菌生命周期結束時，細菌將進行繁殖。細菌的繁殖過程遵循自然界“優勝劣汰，適者生存”的原則。將趨化過程中各個細菌的適應值累加和為標準，較差的半數細菌死亡，較好的半數細菌分裂成兩個子細菌。子細菌繼承母細菌的生物特性，具有與母細菌相同的位置及步長。為簡化計算，可規定複製過程中細菌總數維持不變。
- 3.趨化過程可確保細菌的局部搜尋能力，複製過程可加速細菌的搜尋速度，但對於複雜的最佳化問題，趨化和複製無法避免發生細菌陷入局部極小值的現象。BFA 引入消散過程以加強尋找全域最佳解的能力。當細菌完成一定次數的複製後，將以一定機率被消散到搜索空間中的任意位置。

三、結論

本文介紹幾種常用的 SIOP，包括 AA、PSO、CSA 和 BFA，這些 SIOP 皆是模擬特定生物系統特性的理論和技術，這些 SIOP 在搜尋過程中不容易陷入局部最佳解，即使在所定義的目標函數不連續、不規則或有噪度的狀況下，發現最佳解的機率也很高，而且具有快速找到可行解及縮短搜尋時間的特性。環境管理、環境科學、環境工程等領域的最佳化問題，目標函數通常具有高度非線性的特性，適度採用 SIOP，將可快速找到可行解及縮短搜尋時間。本文的相關內容亦可供環境相關領域的研究人員參考。

參考文獻

1. Dorigo, M., Maniezzo, V., and Colorni, A., Positive feedback as a search strategy. (Tech. Rep. 91-016). Milan, Italy: Politecnico di Milano, Dipartimento di Elettronica, 1991.
2. Dorigo, M., Optimization, learning and natural algorithms. Ph.D. Thesis, Politecnico di Milano, Italy, 1992.
3. 楊淑瑩, 「模式識別與智慧計算: Matlab 技術實現」, 電子工業出版社, 2008。
4. Eberhart, R., and Kennedy, J., A new optimizer using particle swarm theory. In: Proceedings of the IEEE Sixth International Symposium on Micro Machine and Human Science, pp. 39-43. 1995.
5. Kennedy, J., and Eberhart, R.C., Particle swarm optimization. In: Proceedings of IEEE International Conference on Neural Networks, pp. 1942-1948, 1995.
6. 龔純、王正林, 「精通 MATLAB 最優化計算」, 電子工業出版社, 2009。
7. De Castro, L.N., and von Zuben, F.J., Learning and optimization using the clonal selection principle. IEEE Transactions on Evolutionary Computation, Special Issue on Artificial Immune Systems (IEEE), 6 (3): 239-251, 2002.
8. 劉希玉, 「人工神經網絡與微粒群優化」, 北京郵電大學, 2008。
9. Passino, K.M., Biomimicry of bacterial foraging for distributed optimization and control, IEEE Control Systems Magazine, Vol. 22, Issue 3, 52-67, 2002.