介紹

魚群演算法是於2002年由浙江大學的李曉磊博士根據自然界中具有慣性的魚群行為模擬動物行為的適應性、自治性、盲目性、突現性和並行性等慣性所推演出的一種新式群體智能優化算法。

魚群算法是模仿魚類行為方式提出的一種基於動物自體的優化方法，是仿生智慧思想的一個具體應用， 總體的設計理念到具體的實施算法，都不同於傳統的設計和解決方法，同時它又具有與傳統方法相融合的基礎 魚群算法已經在類神經網路優化、參數計算、發電系統排程計算、邊波穩定、組合優化、非線性方程組求解等方面都有很好的應用，並且取得了不錯的效果。

魚群演算法強健性高、對初始參數不敏感、且簡單易上手，又有著優異的求取全局極值的能力。

找尋最佳解的四大行為

隨機行為：

魚在水中悠閒地自由活動，基本上是隨機的，其實他們也是為了搜尋更大範圍的食物或同伴。

覓食行為：

這是生物的一種最基本的行為，也就是趨向食物的一種活動；一般可以認為這種行為是透過 視覺或味覺感知水中的食物量來選擇前進的方向。

群聚行為：

這是魚類常見的一種現象，大量或少量的魚都能群聚成群，這是他們在進化過程中行程的一 種生存方式。

追尾行為：

當某一條魚或幾條魚發現食物時，牠們 附近的魚會尾隨其後快速游過來，進而導致遠處的魚 也尾隨過來。

魚群演算法參數

魚群總數量：fishNum

終止條件：疊代次數

試探次數：tryNumber

可視範圍：visual

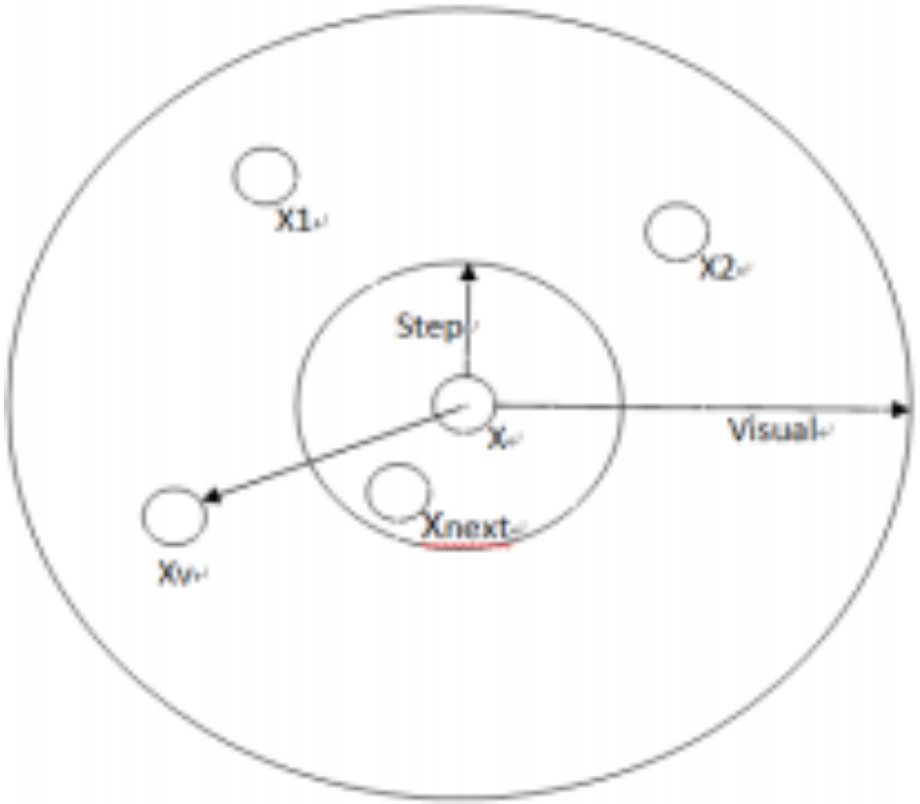
群聚因子：delta

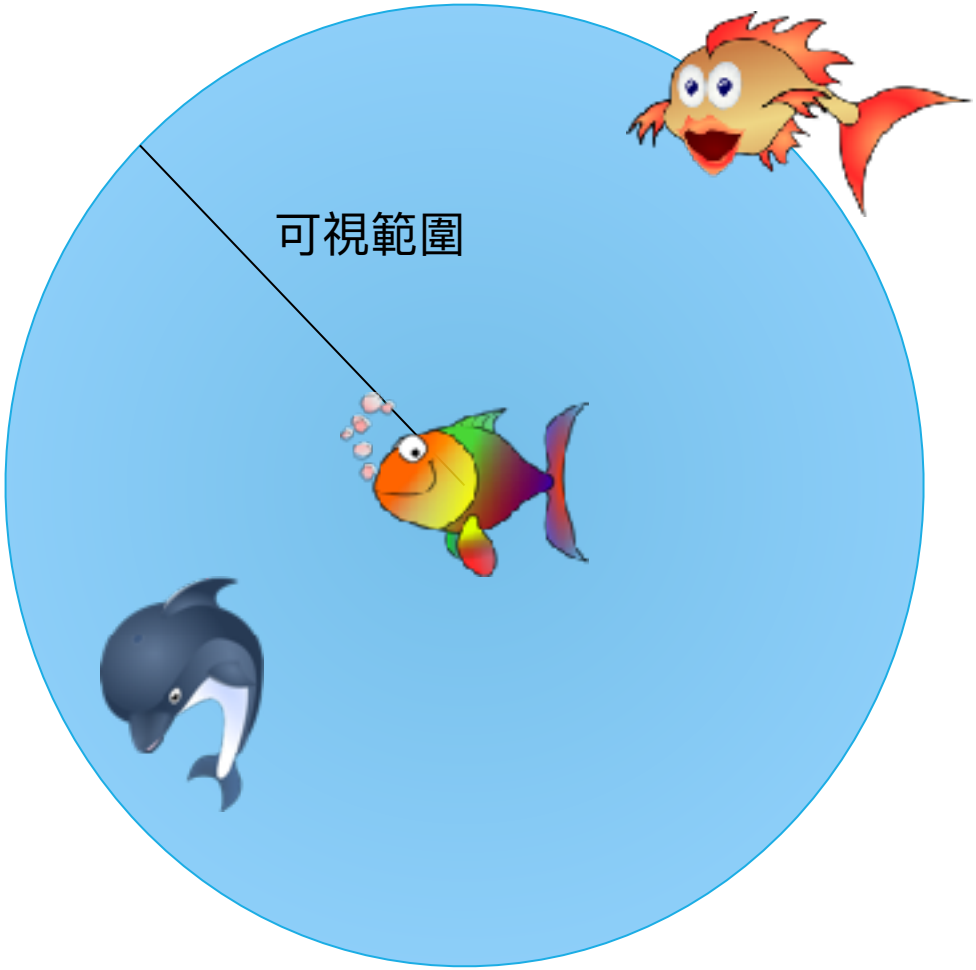
步長：step

X：位置

Y (X)：位置的食物數量

視野與步長

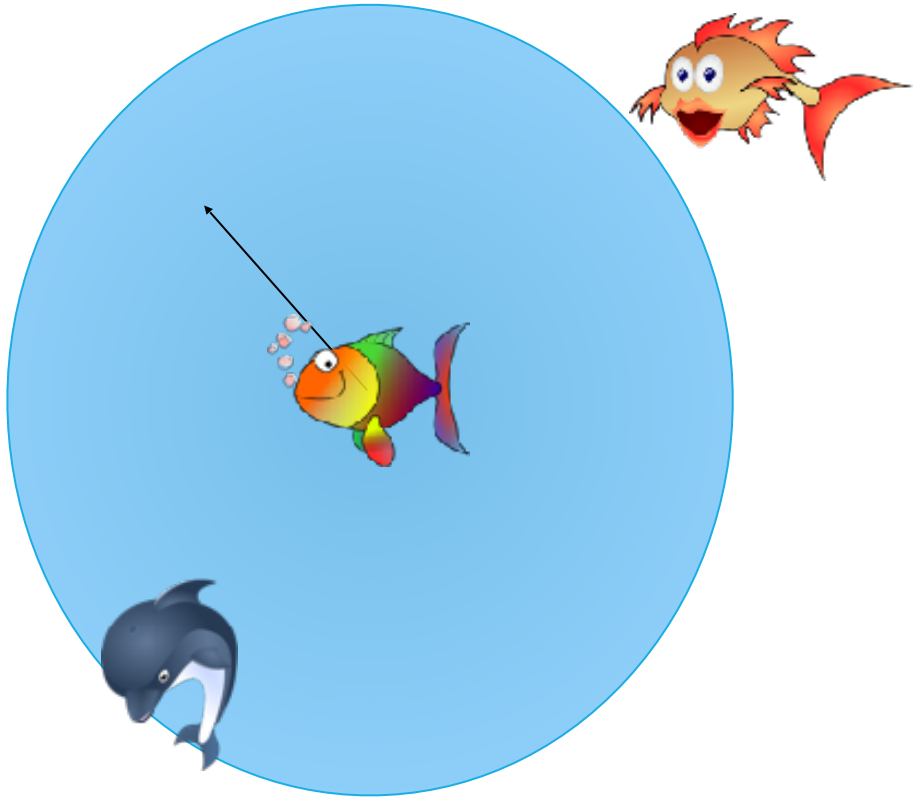




隨機行為

首先，隨機行為是最基礎的動作，當無法滿足其他行為時，就隨機移動，rand 是 0 至 1 的隨機數，step 是步長，以此擴大範圍，避免落入區域解。

$$X_{i\,next} = X_i + rand \times Step$$



覓食行為

覓食行為是先隨機找尋一個位置，若該地方食物值更大，就往該方向去，移動的向量是正歸化後的值。

while tryCount<= tryNumber:

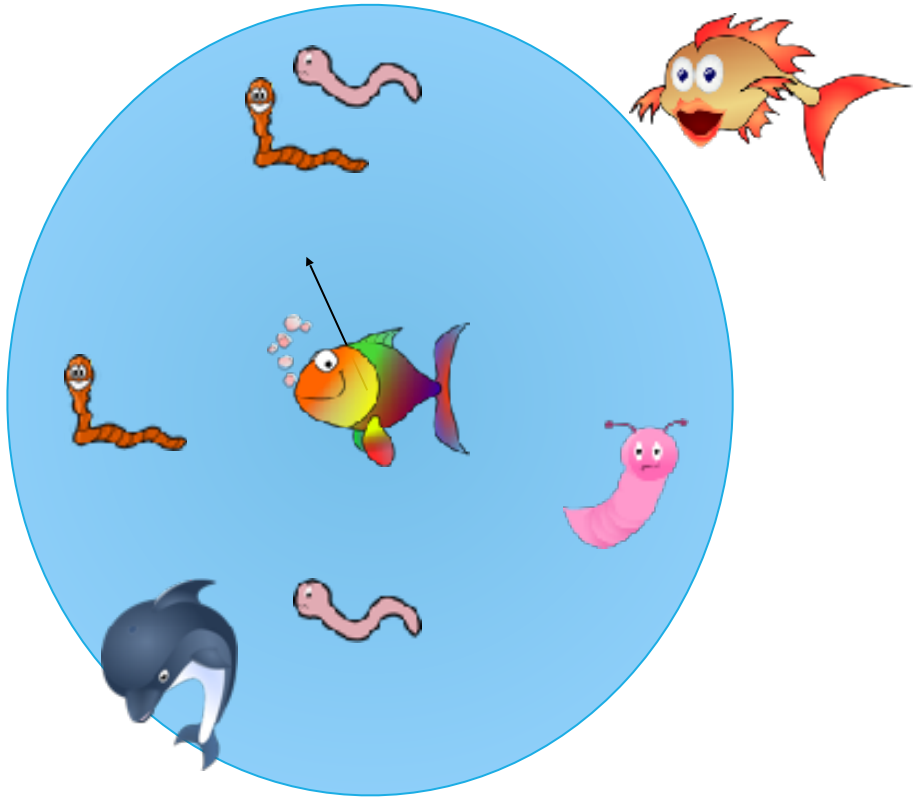
if Y(Xj)>Y(Xi):

$$X_{i|next} = X_i + rand \times Step \times \frac{X_j - X_i}{\|X_j - X_i\|}$$

tryCount+=1

.

.



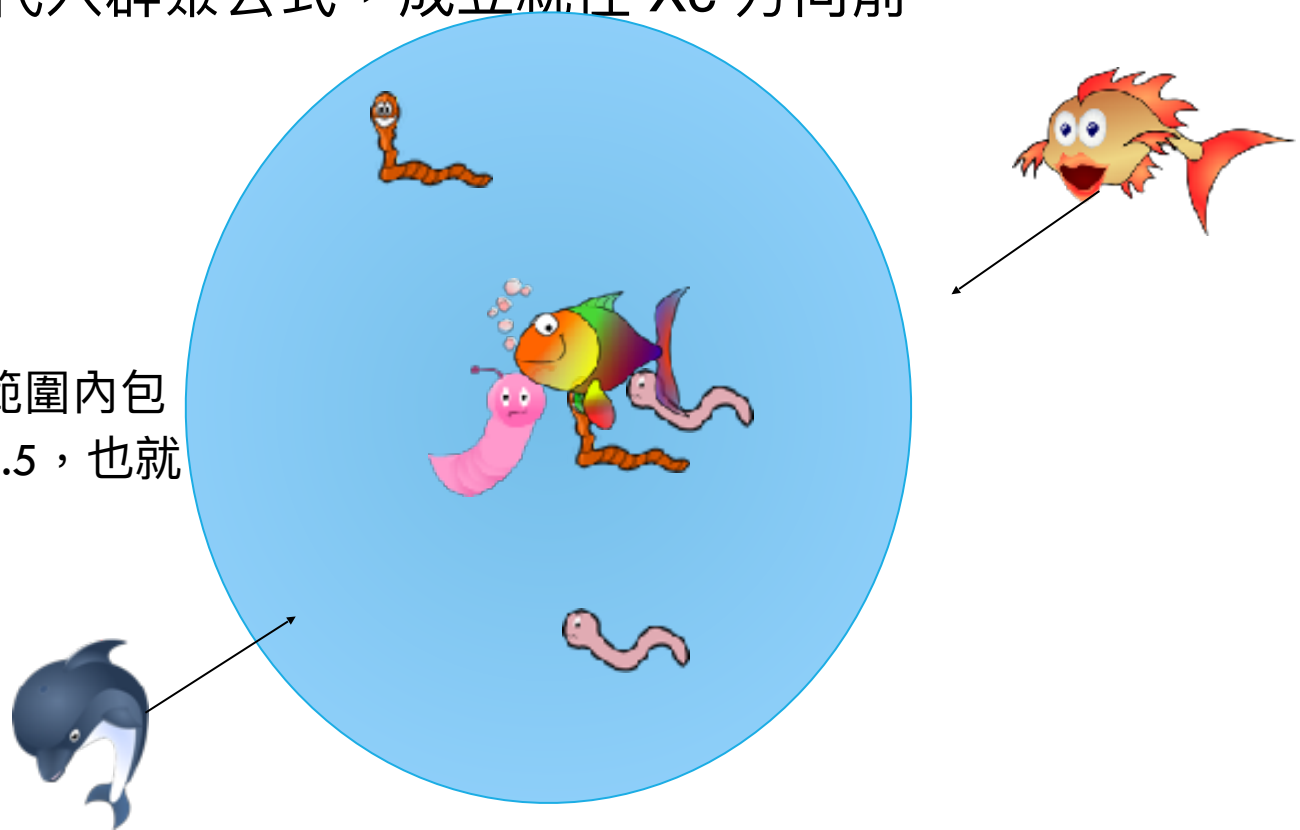
群聚行為

群聚行為的過程，尋找當前位置 Xi 視野內的所有座標，取mean 函數，當作中心位置 Xc，取出 Xc 之食物波度 Yc 代入群聚公式，成立就往 Xc 方向前進，反之執行覓食行為。

If Y(Xc)>Y(Xi) && nf/ fishNum <delta:

$$X_{i|next} = X_i + rand \times step \times \frac{X_c - X_i}{\|X_c - X_i\|}$$

群聚因子delta的意義：假設delta為0.5，nf為可視範圍內包含自己的魚群數量，在這個例子中為 1 ，則1/3<0.5，也就是並非非常密集，則代表這個移動客觀。



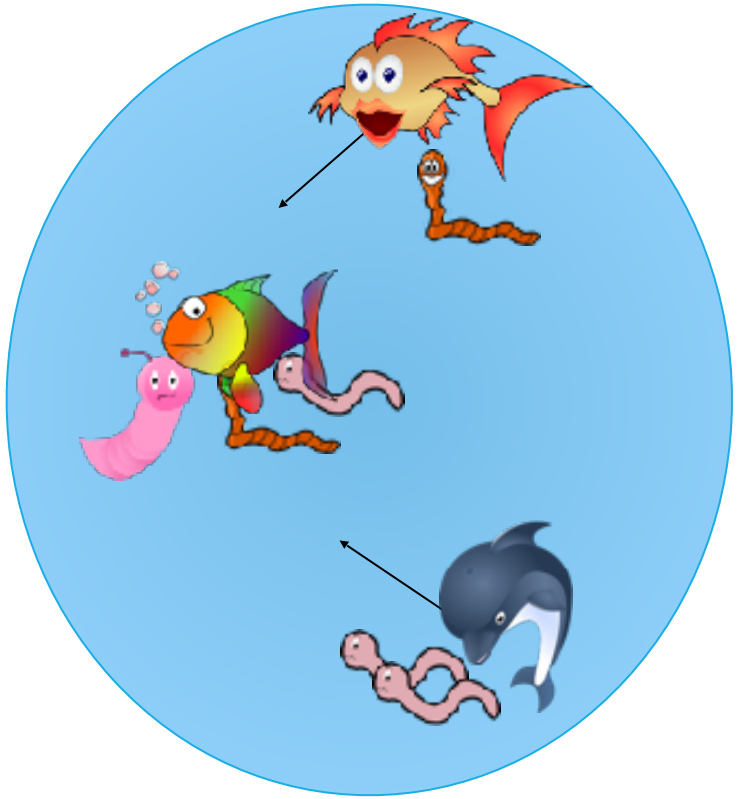
追尾行為

尋找當前位置 Xi 視野內的所有座標，並找出食物波度最大的座標Xmax其食物波度為Ymax，將Ymax帶入追尾公式，成立就往Xmax方向前進一步，反之執行覓食行為

Find maxFood in Current view:

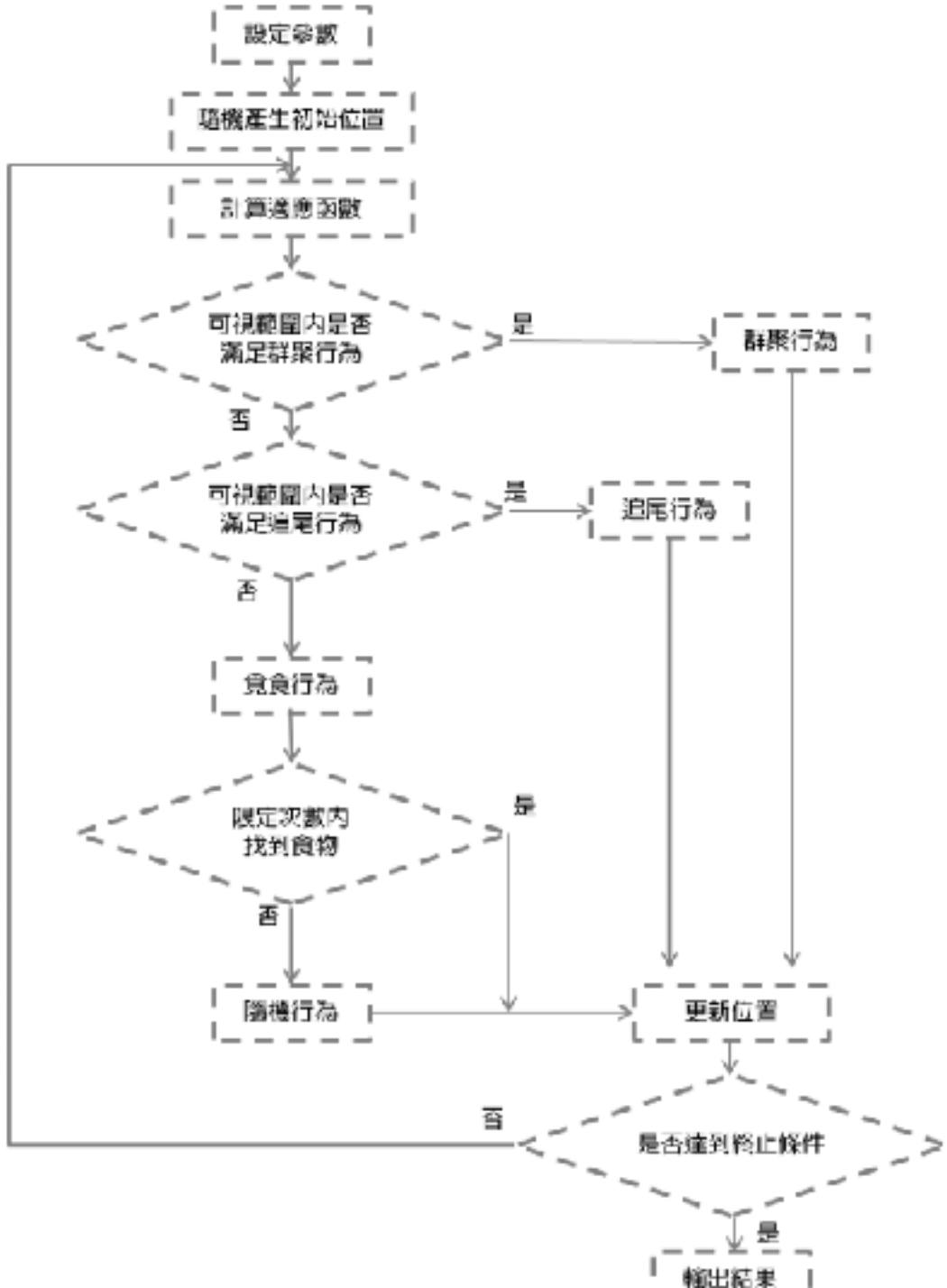
If maxFood >Y(Xi) && nf/ fishNum <delta:

$$X_{i|next} = X_i + rand \times step \times \frac{X_{max} - X_i}{\|X_{max} - X_i\|}$$



流程圖

其中追尾和群聚並無一定先後順序之規定



實驗說明

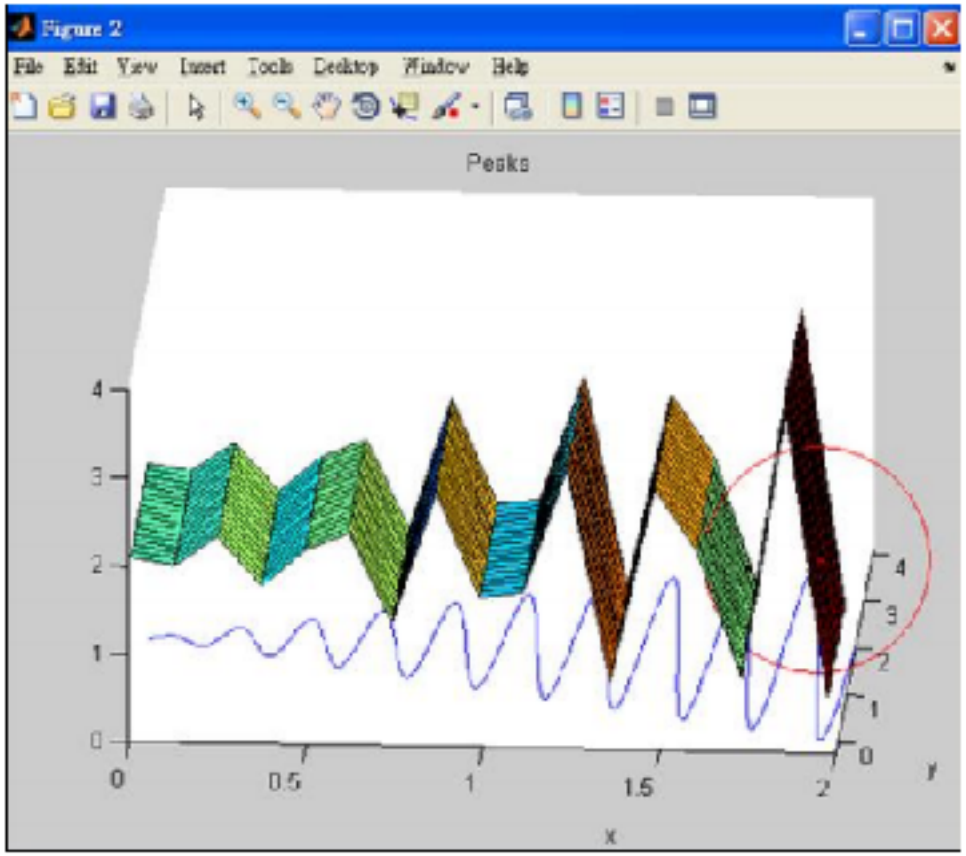
以下實驗結果皆擷取自[3]粒子群演算法與魚群 演算法之比較研究，然而文中並無詳述其實驗流程與參數設定，因此，若以文中提出之表格推論其實驗流程，表格中已提供正確的全域極值，推測，演算法皆設定相同的有限迭代次數，但各個演算法因邏輯不同而有不同的計算時間，當到達最後一次的迭代，AFSA和PSO雖較消耗運算時間，但有助達到真實的全域解，而GA雖有時較快達到最後一次迭代（端看函數的複雜度與初始值），但未必能達真實的全域解。

EXAMPLE

Case1、Constrains: $0 \le x \le 2$

$$\max \ f(x) = x \sin(10\pi x) + 2$$

演算法	次數	1	2	3	4	5	6	7	8	9	10	平均
AFSA	f 極值	3.85	3.85	3.85	3.85	3.85	3.85	3.85	3.85	3.85	3.85	
	X_1	1.85	1.85	1.85	1.85	1.85	1.85	1.85	1.85	1.85	1.85	
	et(秒)	3.24	3.14	3.14	3.14	3.17	3.11	3.14	3.14	3.17	3.16	3.16
GA	f 極值	3.62	3.64	3.54	3.82	3.59	3.28	3.84	3.65	3.67	3.83	
	X_1	1.87	1.87	1.66	1.84	1.83	1.44	1.85	1.65	1.84	1.86	
	et(秒)	3.91	3.92	2.51	1.96	2.20	2.67	2.34	1.85	1.96	2.34	2.57
PSO	f 極值	3.85	3.85	3.85	3.85	3.85	3.85	3.85	3.85	3.85	3.85	
	X_1	1.85	1.85	1.85	1.85	1.85	1.85	1.85	1.85	1.85	1.85	
	et(秒)	4.17	4.14	4.07	4.08	4.33	4.01	4.47	5.00	4.27	4.37	4.29

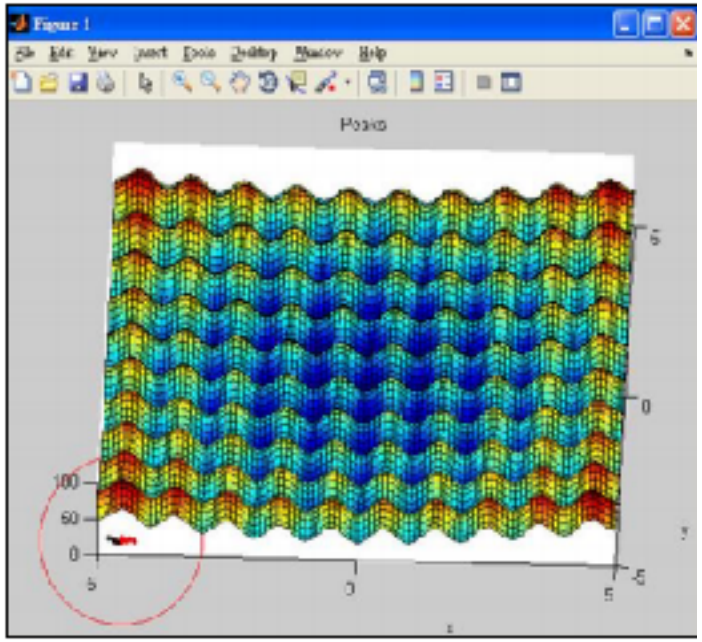


EXAMPLE

Case4、Constrains: $-5 \le x_1, x_2 \le 5$

$$\max\ f(x_1, x_2) = \frac{\sin\sqrt{x_1^2 + x_2^2}}{\sqrt{x_1^2 + x_2^2}} + \exp\frac{\cos 2\pi x_1 + \cos 2\pi x_2}{2} - 2.71289$$

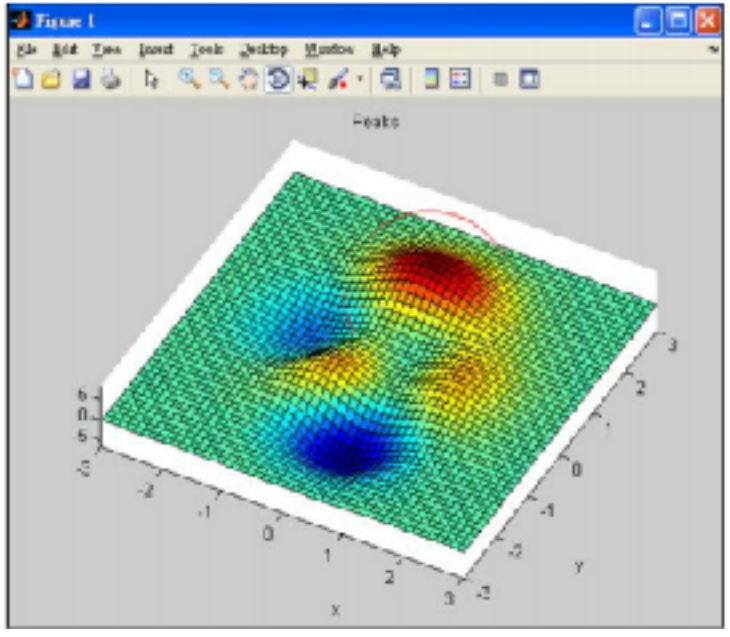
演算法	次數	1	2	3	4	5	6	7	8	9	10	平均
AFSA	f極值	0.92	0.94	0.98	1.00	0.98	0.85	0.97	0.95	1.00	0.97	
	X1	0.0586	0.0396	-0.0108	0.0029	0.0282	0.9954	0.0351	-0.0309	0.0088	0.0067	
	X2	-9.88	-0.03	0.03	-0.01	-8.27	0.00	8.98	-0.03	0.01	-0.03	
	ct(秒)	2.93	2.79	2.83	3.04	2.99	3.21	3.10	3.12	3.09	2.79	2.99
GA	f極值	0.88	0.96	1.00	0.98	1.00	1.00	1.00	0.98	0.85	0.85	
	X1	0.07	0.00	0.00	-0.02	0.01	0.02	0.01	0.03	1.00	0.08	
	X2	0.01	-0.04	0.00	-0.03	0.00	0.00	0.00	0.00	0.00	0.00	
	ct(秒)	0.63	0.64	0.64	0.64	0.63	0.61	0.64	0.64	0.63	0.64	0.63
PSO	f極值	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	
	X1	-3.66	1.10	1.38	2.40	2.53	1.05	1.78	1.20	1.17	8.49	
	X2	-2.29	-2.20	-1.60	-1.63	-3.45	-2.60	-1.44	-2.94	-1.47	-3.54	
	ct(秒)	1.19	1.02	1.03	1.02	1.02	1.02	1.02	1.02	1.03	1.03	1.04



EXAMPLE

Case7、Constrains: $-10 \le x_1, x_2 \le 10$

$$\max f(x_1, x_2) = 3(1-x_1)^2 exp(x_2^2+1) - 10\left(\frac{x_1}{3} - x_1^3 - x_2^3\right) exp(-x_1^2 - x_2^2) - \frac{1}{3} exp((-x_1+1)^2 - x_2^2)$$



演算法	次數	1	2	3	4	5	6	7	8	9	10	平均
AFSA	f 極值	5.82	5.79	5.75	5.81	5.81	5.81	5.81	5.81	5.81	5.82	
	X1	1.4175	1.4197	1.4187	1.3962	1.4221	1.4215	1.4155	1.4134	1.4145	1.4181	
	X2	-0.29	-0.25	-0.30	-0.30	-0.30	-0.30	-0.31	-0.29	-0.28	-0.30	
	時(秒)	11.77	10.12	10.57	11.04	12.08	11.67	13.02	11.46	11.06	13.11	11.59
GA	f 極值	5.78	5.79	5.77	5.64	5.67	5.79	5.74	5.75	5.79	5.79	
	X1	1.46	1.38	1.47	1.47	1.40	1.40	1.36	1.47	1.46	1.41	
	X2	-0.25	-0.32	-0.30	-0.19	-0.19	-0.25	-0.36	-0.25	-0.30	-0.34	
	時(秒)	38.28	44.70	41.44	48.31	45.04	51.14	37.65	41.24	42.84	44.69	43.53
PSO	f 極值	5.82	5.82	5.82	5.82	5.82	5.82	5.82	5.82	5.82	4.52	
	X1	1.42	1.42	1.42	1.42	1.42	1.42	1.42	1.42	1.42	-0.44	
	X2	-0.30	-0.30	-0.30	-0.30	-0.30	-0.30	-0.30	-0.30	-0.30	-0.45	
	時(秒)	6.31	3.70	4.45	4.00	4.73	5.20	4.14	4.84	5.29	5.07	4.77

人工魚群演算法與基因演算法和粒子群演算法比較

GA	**隨機多點交配**		
PSO	個體極值	全域極值	
AFSA	群聚行為	追尾行為	覓食行為、隨機行為

以 $f(x,y)=\frac{\sin(x)}{x}\frac{\sin(y)}{y}$，其中 $-10 \le x \le 10, -10 \le y \le 10$ 為例

模擬情型來看，如果我們是以準確率為標準的話，我們可以看的出來，PSO 與 AFSA 這兩個演算法，都能準確的找到最佳解,雖然AFSA 所用的時間最久。如果我們是以速度為標準，其次才是準確度的話，GA 更完美，只需 0.02 秒，即可趨近最佳解。而 AFSA，相較其他演算法，屬於平庸型的。

參考

[1] 人工魚群AFSA https://drive.google.com/file/d/0B88DFhUzRqSdTk0xRXlwOFpNR0k/view

[2] 人工魚群演算法應用於混合式再生能源之系統規劃

https://eportal.stust.edu.tw/eshare/EshareFile/2013_12/2013_12_37af05be.pdf

[3]粒子群演算法與魚群 演算法之比較研究

http://www.admin.ltu.edu.tw:8000/Public/Upload/files/%E8%B3%87%E7%AE%A1%E7%B3%BB/102%E5%AD%B8%E5%B9%B4%E5%BA%A6%E5%9B%9B%E6%8A%80%E7%95%A2%E6%A5%AD%E5%B0%88%E9%A1%8C/10241.pdf

[4] Springer Science+Business Media B.V. 2012, Artificial fish swarm algorithm: a survey of the stateof-the-art, hybridization, combinatorial and indicative applications.

[5]李晓磊,邵之江,钱积新.一种基于动物自治体的寻优模式:鱼群算法［J］.系统工程理论与实践, 2002,22(11):32-38.