

第3章 粒子群优化算法

3.1 群集智能算法基本思想

3.2 PSO基本算法

3.3 PSO标准算法

3.4 PSO收敛性

3.4 PSO改进算法

3.5 PSO应用

群体智能（Swarm Intelligence）



生物学家研究表明：在这些群居生物中虽然每个个体的智能不高，行为简单，也不存在集中的指挥，但由这些单个个体组成的群体，似乎在某种内在规律的作用下，却表现出异常复杂而有序的群体行为。

群体智能

- 群 (swarm)定义为某种交互作用的组织或agent的结构集合。
- 群集中个体行为简单，但群体行为相当复杂；
- 社会组织中全局群行为由群内个体行为以非线性方式出现，个体行为与全局群行为之间存在某种紧密的联系。个体的集体行为构成和支配了群行为；群行为决定个体执行其作用的条件。
- 由群行为决定的条件包括空间和时间两种模式。

群体智能

- 群行为不能仅仅由独立于其他个体的个体行为所决定，个体间的交互作用在构建群行为中起到重要作用；
- 个体间的交互作用帮助改善对环境的经验知识，增强到达优化的群进程。
- 个体间的交互作用或合作由遗传学或通过社会交互确定。
- 社会交互可以是直接或间接的。
- 直接交互通过视觉、听觉或化学接触；间接交互是在某个个体改变环境，其他个体反应该新环境时出现的。

PSO基本思想

- 粒子群优化(Particle Swarm Optimization, PSO)最早是由心理学研究人员Kenned 博士和计算智能研究人员Eberhart博士于1995年提出的，它是源于对鸟群觅食过程中的迁徙和群居的模拟。
- 相较于其它的演化式寻优法则，PSO常被归属于非常有效率的启发式算法的一种。

粒子群最佳化



Russ Eberhart

整合群體行為、人類決策與鳥群行為發展而成稱為
粒子群演算法。

【Eberhart, Kennedy, 1995】



PSO——基于群体的优化工具

- PSO算法是基于群智能(Swarm Intelligence)方法的演化计算技术，引入了“群”的概念。PSO是一种基于群体的优化工具，同时也是一种基于迭代的优化工具。
- 系统初始化为一组随机解，通过迭代搜寻最优值，粒子（潜在的解在解空间追随最优的粒子进行搜索。在PSO中，采用信息共享机制，它有着简单容易实现同时又有深刻的智能背景的特点。

鸟群的飞行行为

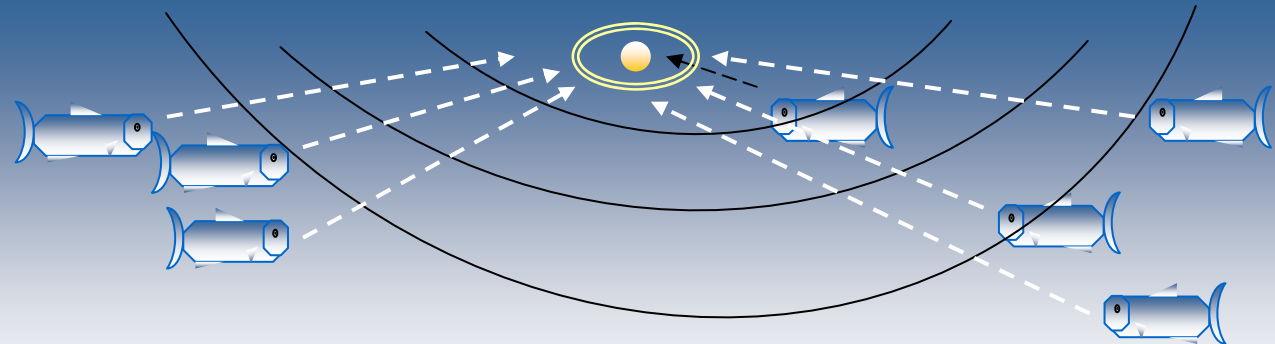


- ❖ 避免碰撞
- ❖ 速度匹配
- ❖ 中心聚集

2012-3-19

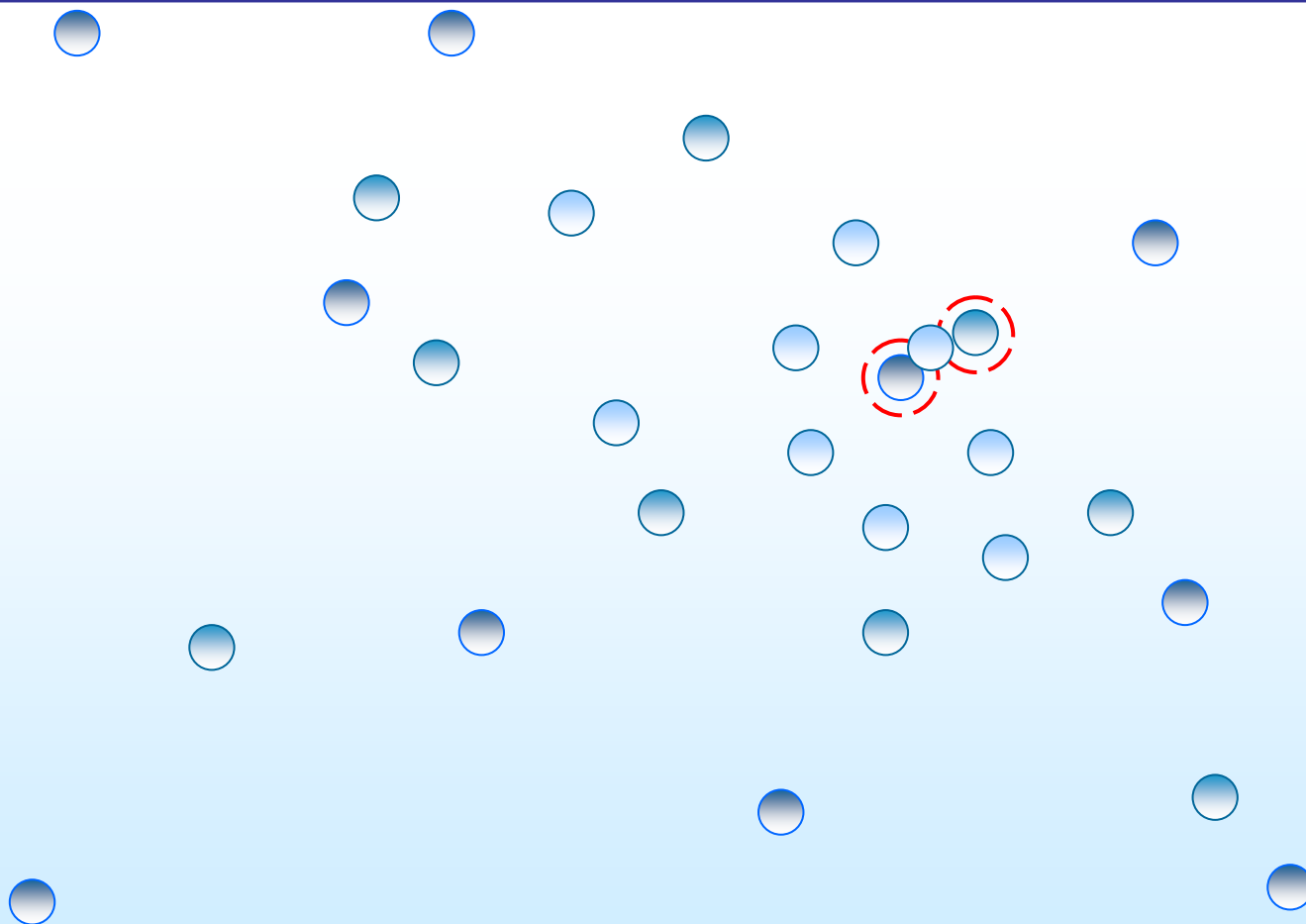
鳥群(魚群)行為

師法大自然

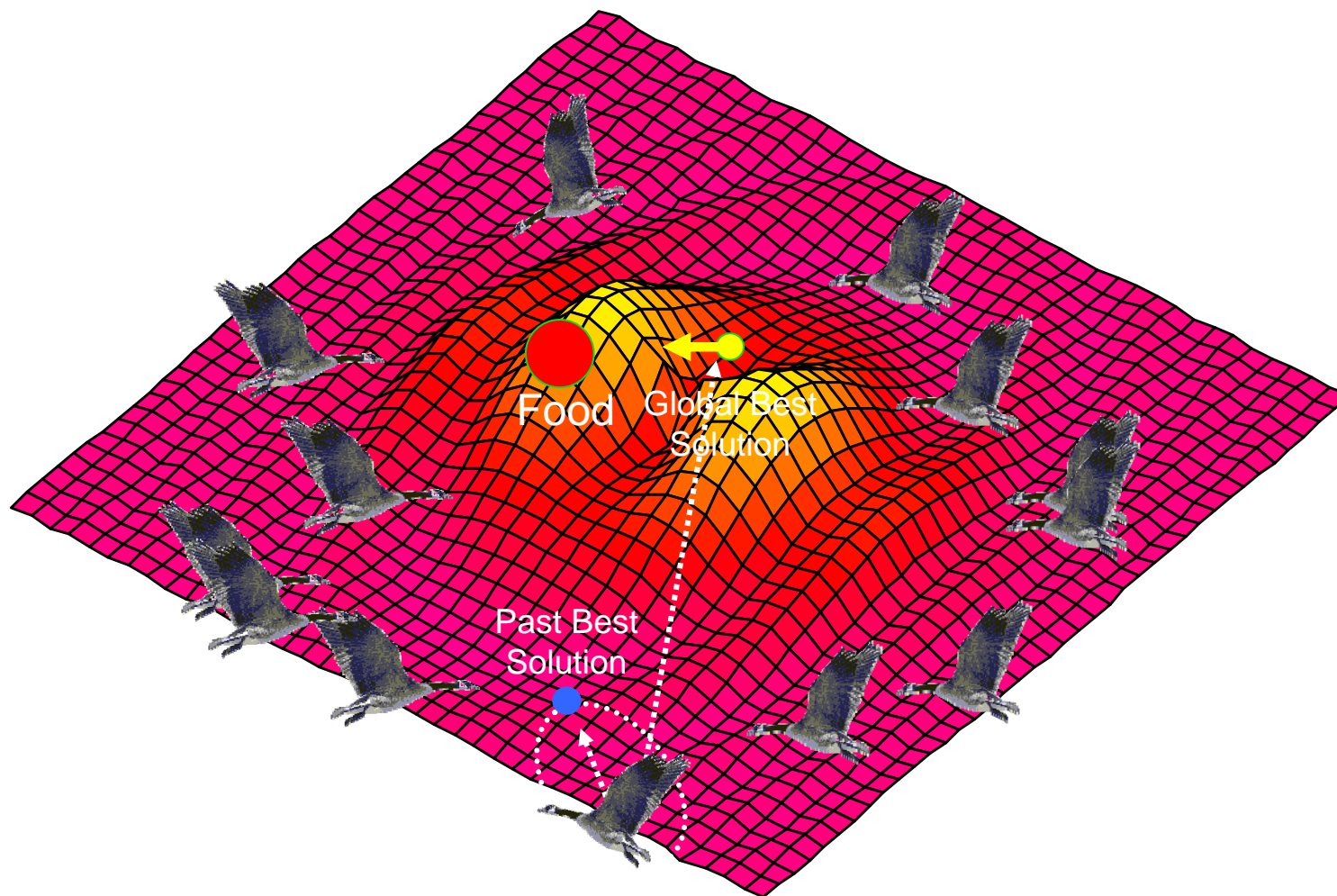


鳥群(魚群)行為

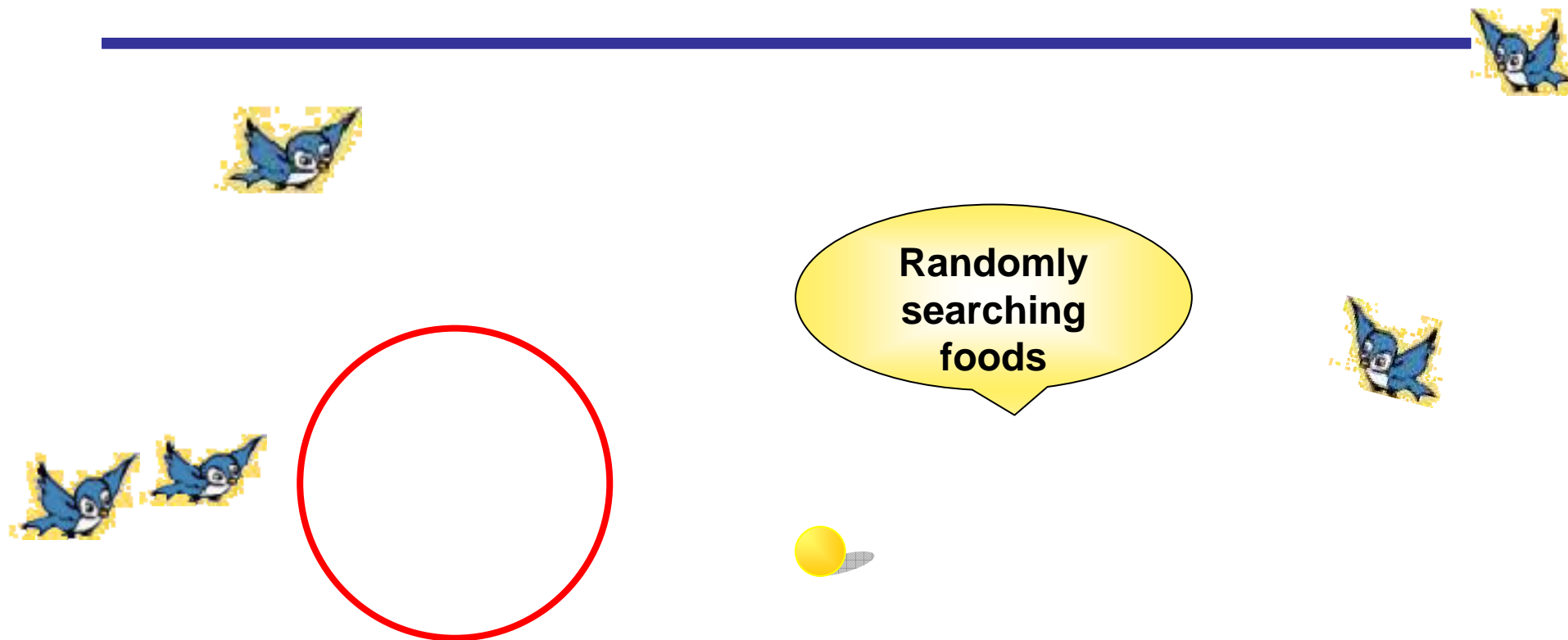
粒子群特性



鸟群觅食模型

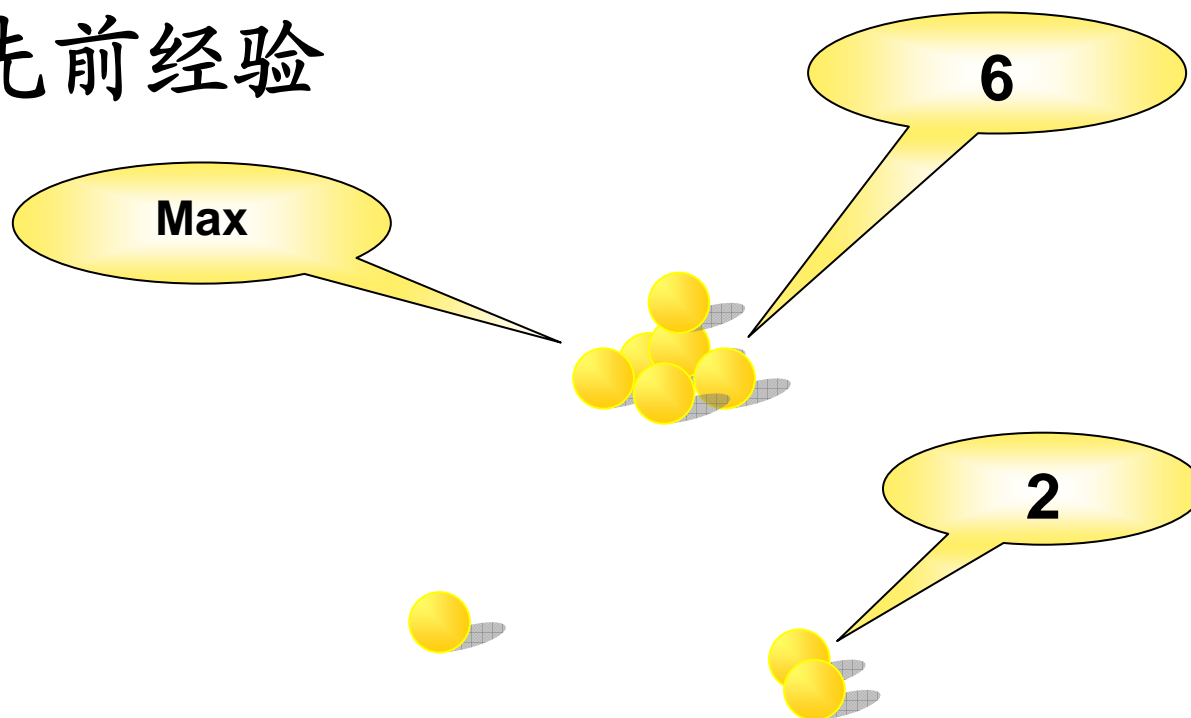


社会型行为的模拟



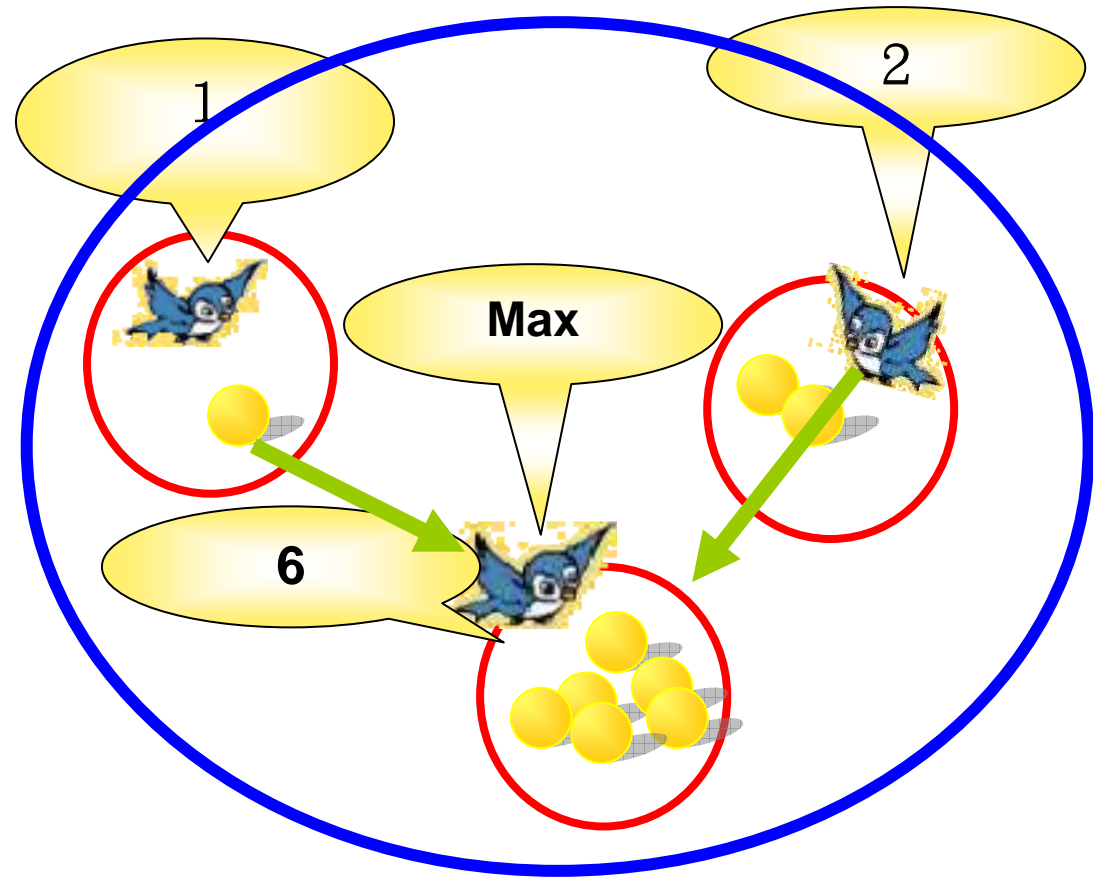
认知行为 (Cognition Behavior)

▶ 先前经验



社会行为 (Social Behavior)

- We tend to adjust our beliefs and attitudes to conform with those of our social peers.



人类社会系统

粒子群算法介绍

- ❖ 每个寻优的问题解都被想像成一支鸟，也称为“Particle”。
- ❖ 所有的Particle 都有一个fitness function 以判断目前的位置之好坏，
- ❖ 每一个Particle具有记忆性，能记得所搜寻到最佳位置。
- ❖ 每一个Particle 还有一个速度以决定飞行的距离与方向。

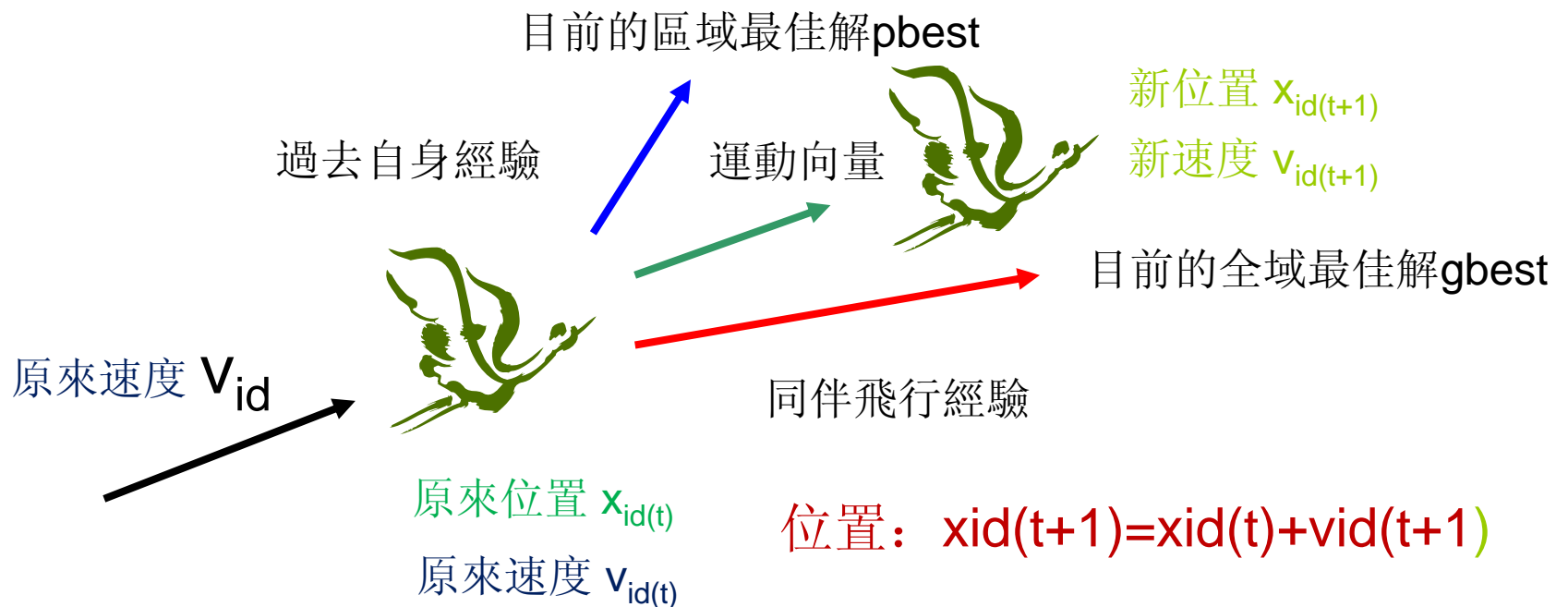
PSO算法描述

假设在一个 D 维的目标搜索空间中，有 m 个粒子组成一个群落，其中第 i 个粒子表示为一个 D 维的向量 $X_i = (x_{i1}, x_{i2}, \dots, x_{iD})$ ， $i = 1, 2, \dots, m$ 即第 i 个粒子在 D 维搜索空间中的位置是 X_i 。换言之，每个粒子的位置就是一个潜在的解。将 X_i 带入一个目标函数就可以计算出其适应值，根据适应值的大小衡量解的优劣。第 i 个粒子的“飞翔”速度也是一个 D 维的向量，记为 $V_i = (v_{i1}, v_{i2}, \dots, v_{iD})$ 。记第 i 个粒子迄今为止搜索到的最优位置为 $P_i = (p_{i1}, p_{i2}, \dots, p_{iD})$ ，整个粒子群迄今为止搜索到的最优位置为 $P_g = (p_{g1}, p_{g2}, \dots, p_{gD})$ 。对粒子可按下列公式操作：

PSO 目標式

$$\text{速度: } v_{id}(t+1) = w \times v_{id}(t) + c_1 \times \text{rand}() \times [p_{id}(t) - x_{id}(t)] + c_2 \times \text{rand}() \times [P_{gd}(t) - x_{id}(t)]$$

v-速度
w-慣性權重
C-學習因子
pid-區域最佳解
Pgd-全域最佳解



速度与位置更新

$$v_{id}(t+1) = v_{id}(t) + c_1 \cdot rand() \cdot (p_{id} - x_{id}(t)) + c_2 \cdot rand() \cdot (p_{gd} - x_{id}(t))$$

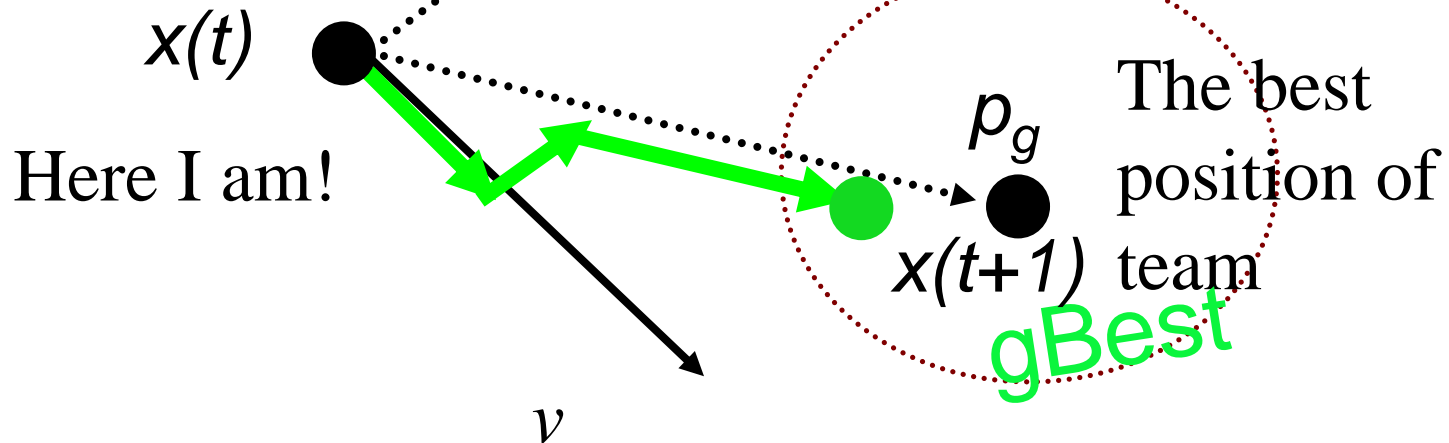
$$x_i(t+1) = x_i(t) + v_i(t)$$

Study Factor

$$V_i = \{V_{i1}, V_{i2}, \dots, V_{id}\}$$

$$X_i = \{X_{i1}, X_{i2}, \dots, X_{id}\}$$

My best position



動能向量概論

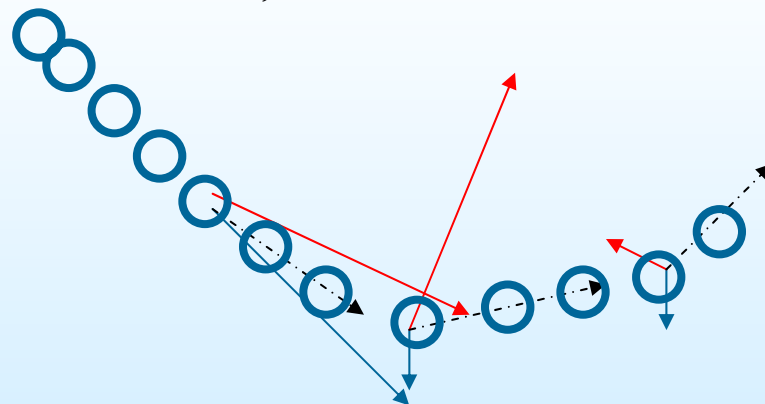
$$V_i^{t+1} = \omega V_i^t + C_1 * \phi^*(P_i^t - X_i^t) + C_2 * \phi^*(P_g^t - X_i^t)$$

$$X_i^{t+1} = X_i^t + V_i^{t+1}$$

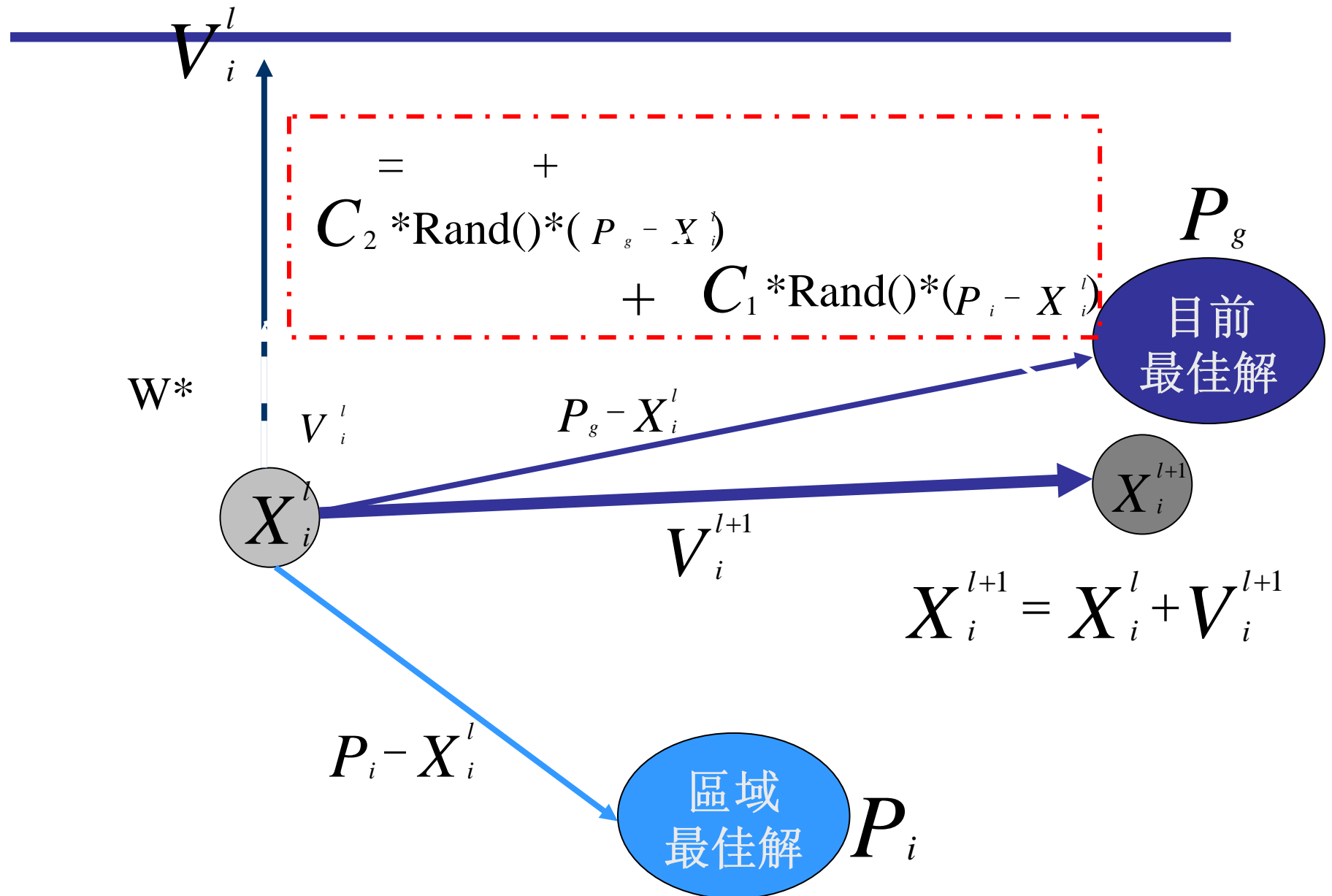
craziness

$$V_i = \{V_{i1}, V_{i2}, \dots, V_{iN}\}$$

$$X_i = \{X_{i1}, X_{i2}, \dots, X_{iN}\}$$

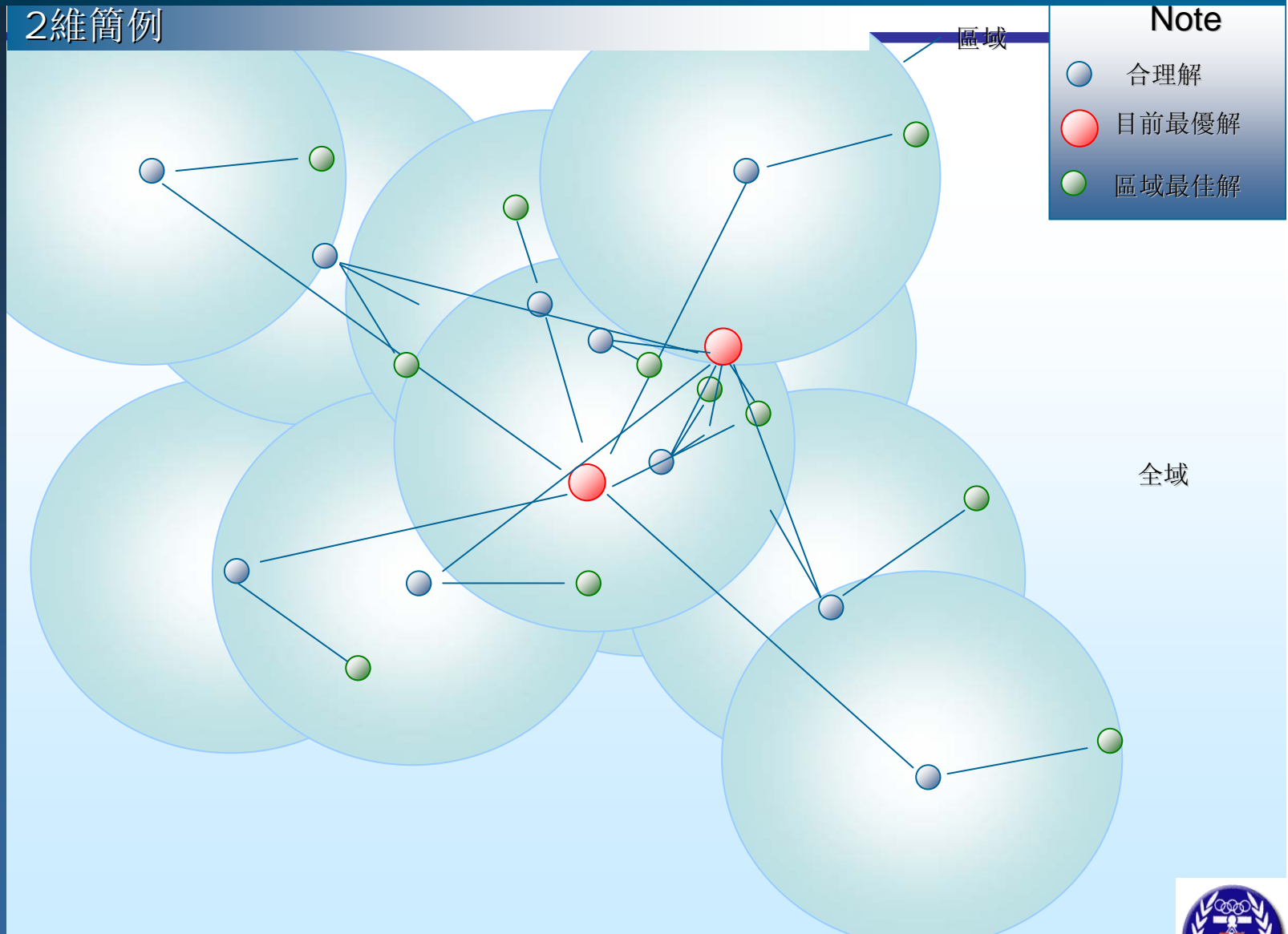


PSO 向量示意圖



粒子群最佳化

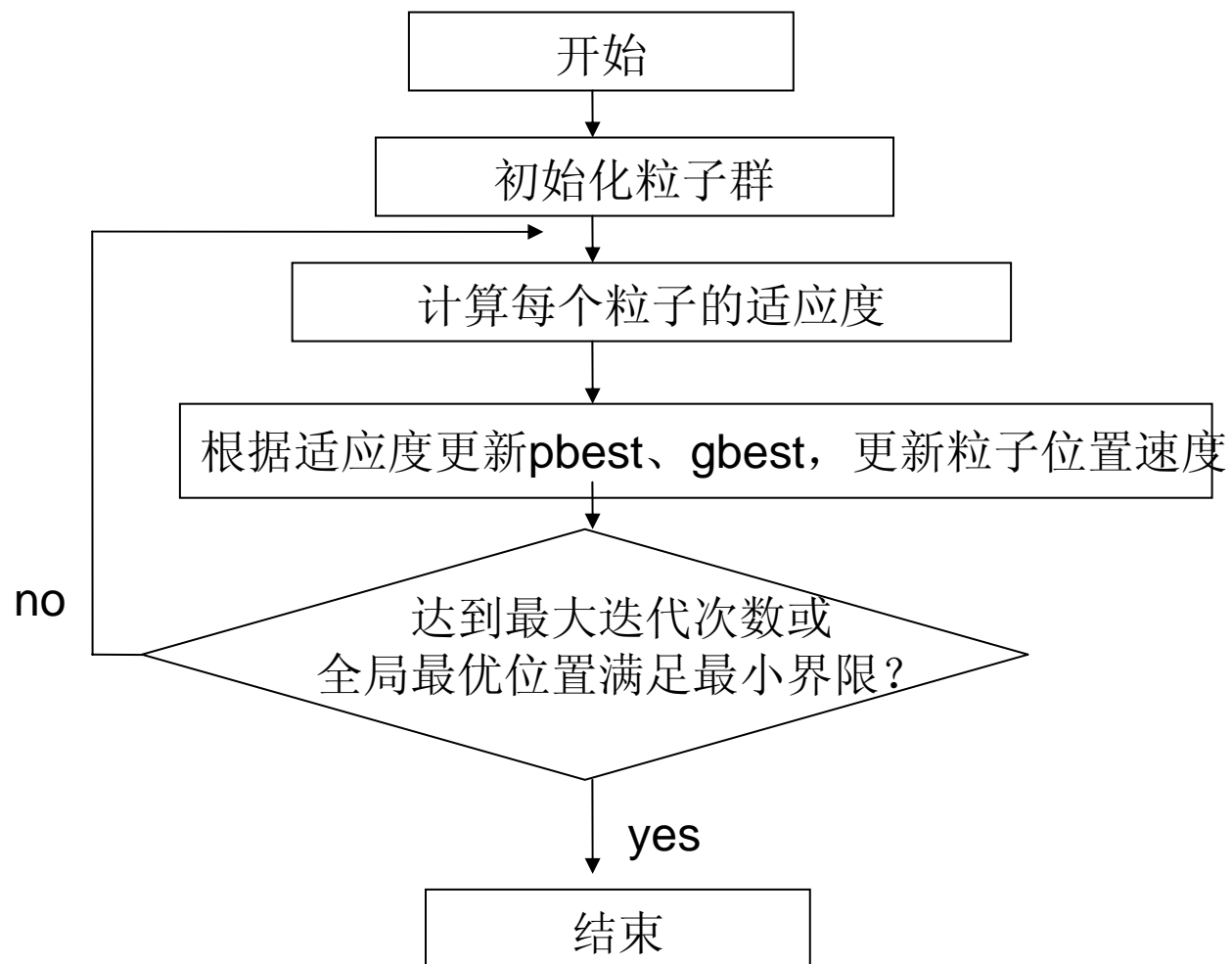
2維簡例



算法流程

- ❖ **Initialization** : 将群族做初始化, 以随机的方式求出每一Particle 之初始位置与速度。
- ❖ **Evaluation**: 依据fitness function 计算出其fitness value 以作为判断每一个Particle之好坏。
- ❖ **Find Pbest** : 找出每一个Particle 到目前为止的搜寻过程中最佳解, 这个最佳解称之为Pbest。
- ❖ **Find the Gbest**: 找出所有群体中的最佳解, 此最佳解称之为Gbest。
- ❖ **Update the Velocity and position**: 根据速度与位置公式 更新每一Particle的速度与位置。
- ❖ **Termination**. 返回步骤2继续执行, 直到获得一个令人满意的结果或符合终止条件为止。

基本粒子群优化算法流程图



粒子群优化算法的流程描述伪代码

```
Do
  for  $i=1$  to  $n$  //  $n$  is the number of particles
    if  $G(X_i) > G(P_i)$  do
      for  $d=1$  to  $m$  //  $m$  is the dimension of particle
         $P_{id} = x_{id}$ 
      end for
       $g = i$  // index of the best particle
      for  $j=1$  to  $n$ 
        if  $G(P_j) > G(P_i)$  do
           $g = j$ 
        end if
      end for
      for  $d=1$  to  $m$ 

$$V_i(k+1) = wV_i(k) + c_1r_1(P_i - X_i(k))/\Delta t + c_2r_2(P_g - X_i(k))/\Delta t$$

$$X_i(k+1) = X_i(k) + V_i(k+1)\Delta t$$
 // according to (2.18) and (2.19)
      end for
    end for
  end for
while (算法停止条件不满足)
```

PSO算法分析

从社会学的角度来看，粒子速度更新公式

$$v_{id}(t+1) = v_{id}(t) + c_1 \cdot rand() \cdot (p_{id} - x_{id}(t)) + c_2 \cdot rand() \cdot (p_{gd} - x_{id}(t))$$

可看成由三部分组成：

- (1) “惯性”部分。粒子先前的速度乘以一个权值进行加速，表示粒子对当前自身运动状态的信任，依据自身的速度进行惯性运动。

PSO算法分析

（（2）“认知”部分。粒子当前位置与自身最优位置之间的距离，表示粒子本身的思考，即粒子的运动来源于自己经验的部分。

$$v_{id}(t+1) = v_{id}(t) + c_1 \cdot rand() \cdot (p_{id} - x_{id}(t))$$

——仅有此性能变差，不同微粒间缺乏信息交流，使规模为N的群体等效运行N个单个微粒，获得最优解的概率非常小。

PSO算法分析

(3) “社会”部分。粒子当前位置与群体最优位置之间的距离，表示粒子间的信息共享与相互合作，即粒子的运动来源于群体中其他粒子经验的部分。

$$v_{id}(t+1) = v_{id}(t) + c_2 \cdot rand() \cdot (p_{gd} - x_{id}(t))$$

只有社会的模型，有能力到达新的搜索空间。对复杂问题，容易陷入局部最优点。

参数选择

- ❖ **粒子数**: 一般取 20 – 40. 其实对于大部分的问题10个粒子已经足够可以取得好的结果, 不过对于比较难的问题或者特定类别的问题, 粒子数可以取到100 或 200
- ❖ **粒子的维数**: 这是由优化问题决定, 就是问题解的长度
- ❖ **粒子的范围**: 由优化问题决定, 每一维可是设定不同的范围
- ❖ **Vmax**: 最大速度, 决定粒子在一个循环中最大的移动距离, 通常设定为粒子的范围宽度
- ❖ **学习因子**: $c1$ 和 $c2$ 通常等于 2. 不过在文献中也有其他的取值. 但是一般 $c1$ 等于 $c2$ 并且范围在0和4之间
- ❖ **中止条件**: 最大循环数以及最小错误要求.

PSO与遗传算法的比较

➤ 相同点

- 都是基于种群的
- 都需要适应度函数.
- 都是随机计算技术
- 不能保证100%收敛

➤ 不同点

- PSO没有交叉变异等进化操作.
- PSO中通过粒子的竞争与协作实现种群进化
- 粒子具有记忆能力

➤ 优点

- PSO 容易实现具有较小的调整参数
- 收敛速度快、解质量高、鲁棒性好

PSO/GA广义理解

- 具有隐含的选择机制
- 进化方程与实数编码的算术交叉向类似，“认知”与“社会”部分可理解为两个父代产生一个子代的算术交叉运算。
- 速度更新方程也可理解为一个变异算子，其变异强度取决于两个父代微粒间的距离，即个体最优和全局最优的两个微粒的距离；
- $W \times v_d$ 可看作另一种变异形式，其大小与微粒在前代进化中的位置相关。

PSO/GA广义理解2

- 进化类算法——子代替代父代 只利用位置信息

PSO进化为自适应过程，微粒位置 x 根据速度向量自适应更新； 同时利用位置与速度信息；

- 如果将PSO位置更新公式看作变异算子，则PSO与进化规划相似；

PSO：微粒向群体经验给出的最好方向飞行

进化规划：随机变异到任何方向。

- PSO为“意识（Conscious）”的变异，如果“意识”可提供有用信息，则PSO有更多机会更快飞到很好解的区域。

PSO的两种进化模型

(1) Gbest模型（全局最优模型）

$P_g(t)$ ，全局最优点

$$v_{id}(t+1) = v_{id}(t) + c_1 \cdot rand() \cdot (p_{id} - x_{id}(t)) + c_2 \cdot rand() \cdot (p_{gd} - x_{id}(t))$$

(2) Lbest模型（局部最优模型）

多吸引子代替Gbest模型中的单一吸引子

将整个粒子群分成若干个子群，每一子群中保留其局部最好粒子 $P_{id}(t)$

*：子群中的微粒与其所在搜索空间的位置无关，仅与微粒的索引或编码有关。

$$v_{id}(t+1) = v_{id}(t) + c_1 \cdot rand() \cdot (p_{id} - x_{id}(t)) + c_2 \cdot rand() \cdot (p_{ld} - x_{id}(t))$$

PSO标准算法

带惯性加权因子的PSO算法

$$v_{id}(t+1) = w \cdot v_{id}(t) + c_1 \cdot rand() \cdot (p_{id} - x_{id}(t)) + c_2 \cdot rand() \cdot (p_{gd} - x_{id}(t))$$

$$x_i(t+1) = x_i(t) + v_i(t)$$

带收缩因子（constriction factor）的PSO算法

$$v_{id}^{k+1} = \lambda \left[v_{id}^k + c_1 r_1 (p_{id} - x_{id}^k) + c_2 r_2 (p_{gd} - x_{id}^k) \right]$$

$$\lambda = \frac{2}{\left| 2 - \varphi - \sqrt{\varphi^2 - 4\varphi} \right|} \quad \varphi = c_1 + c_2 \quad \varphi > 4$$

收缩因子

- 在使用收缩因子时，通常取 $\phi = 4$ ，从而使收缩因子为**0.729**。从数学上分析，惯性权重和收缩因子这两个参数是等价的。

W的意义

- 探测（**exploration**）是指粒子在较大程度上离开原先的寻优轨程，偏到新的方向进行搜索；
- 开发（**exploitation**）则指粒子在较大程度上继续原先的寻优轨程进行细部搜索，
- w 的大小可以用来控制算法的开发和探索能力。
较大的惯性权重将使粒子具有较大的速度，从而具有较强的探索能力；
- 较小的惯性权重将使粒子具有较大的开发能力，
因此选择适合的惯性权重可以使粒子具有均衡的探测和开发能力。

线性递减权值（LDW）策略

$$w^k = \frac{(w_{ini} - w_{end})(K_{max} - k)}{K_{max}} + w_{end}$$

- 其中， K_{max} 是PSO算法设定的最大迭代次数， w_{ini} 是初始惯性权重， w_{end} 是最大迭代终止时的惯性权重， k 是当前迭代次数。

指数递减策略

$$w^k = (w_{ini} - w_{end})(\exp(-a * k)) + w_{end}$$

- 其中， K_{\max} 是PSO算法设定的最大迭代次数， w_{ini} 是初始惯性权重， w_{end} 是最大迭代终止时的惯性权重， k 是当前迭代次数。

随机惯性权重取值策略

惯性权重在 $[0, 1]$ 随机取值，效果优于LDW。

- 通过标准测试函数Rosenbrock和Rastrigin 对上述策略进行试验，
- 其中Rosenbrock是单峰函数，而Rastrigin是多峰函数。将50次试验的最优结果
- 取均值列于表4.1和.42。试验中速度上限 V_{max} 取各函数初始范围的上限，
- $C1=c2=2.0$ 。

表3.1Rosnebockrk函数平均最优适应值

粒子数	维数	最大代数	LDW 法	RIW 法
20	10	1000	106.63370	65.28474
	20	1500	180.17030	147.52372
	30	2000	458.28375	409.23443
40	10	1000	61.36835	41.32016
	20	1500	171.98795	95.48422
	30	2000	289.19094	253.81490
80	10	1000	47.91896	20.77741
	20	1500	104.10301	82.75467
	30	2000	176.87379	156.00258

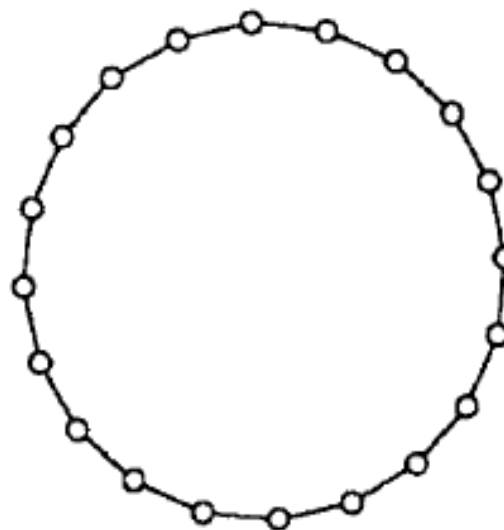
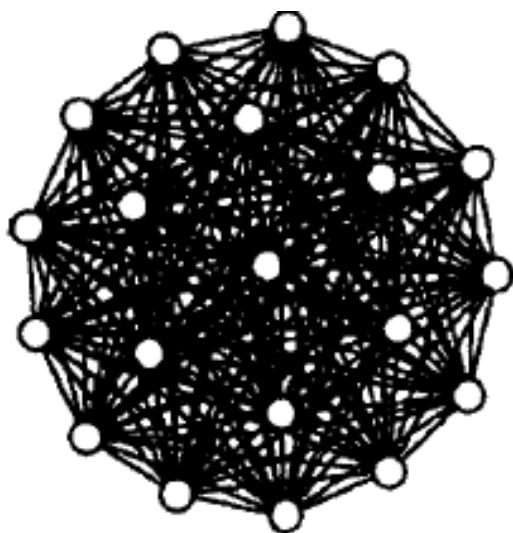
表3.2 Rastrigni函数平均最优适应值

粒子数	维数	最大代数	LDW 法	RNW 法
20	10	1000	5.25230	5.04258
	20	1500	22.92156	20.31109
	30	2000	49.21827	42.58132
40	10	1000	3.56574	3.22549
	20	1500	17.74121	13.84807
	30	2000	38.06483	32.15635
80	10	1000	2.37332	1.85928
	20	1500	13.11258	9.95006
	30	2000	30.19545	25.44122

粒子群优化算法种群拓扑结构研究

- 种群拓扑结构是指整个种群所有粒子之间的连接方式(相互连接的粒子进行通信)。而邻域结构则是单个粒子与它通信粒子的连接方式。
- 粒子的行为主要由其局部邻域结构影响，该局部邻域可视为种群拓扑结构中的部分区域。种群邻域结构的限定可阻止信息在整个种群中的流动，从而保持种群多样性，它可控制算法的探测和开发能力。
- 种群拓扑结构对PSO算法性能的影响有两个层面：其一，可选取不同粒子的局部邻域结构；其二，可定义不同的局部邻域之间的通信方式。

几种典型种群拓扑结构



- (a)全局模型;
- (b)局部模型
- 全局(gbest)种群中所有个体都相互通信
- 局部(lbest)种群队列中相邻个体之间通信,

几种典型种群拓扑结构

- (3) 4类(four cluster)整个粒子群由4个类组成，4个类内部个体相互完全通信，类之间通信较少，见图3.2(a)所示。
- (4) 金字塔(pyramid)三角形线框的金字塔结构；
- (5) 冯.诺以曼(von Neumman)四方网格，顶点相连形成环面，见图3.2(e)。

PSO结构示意图

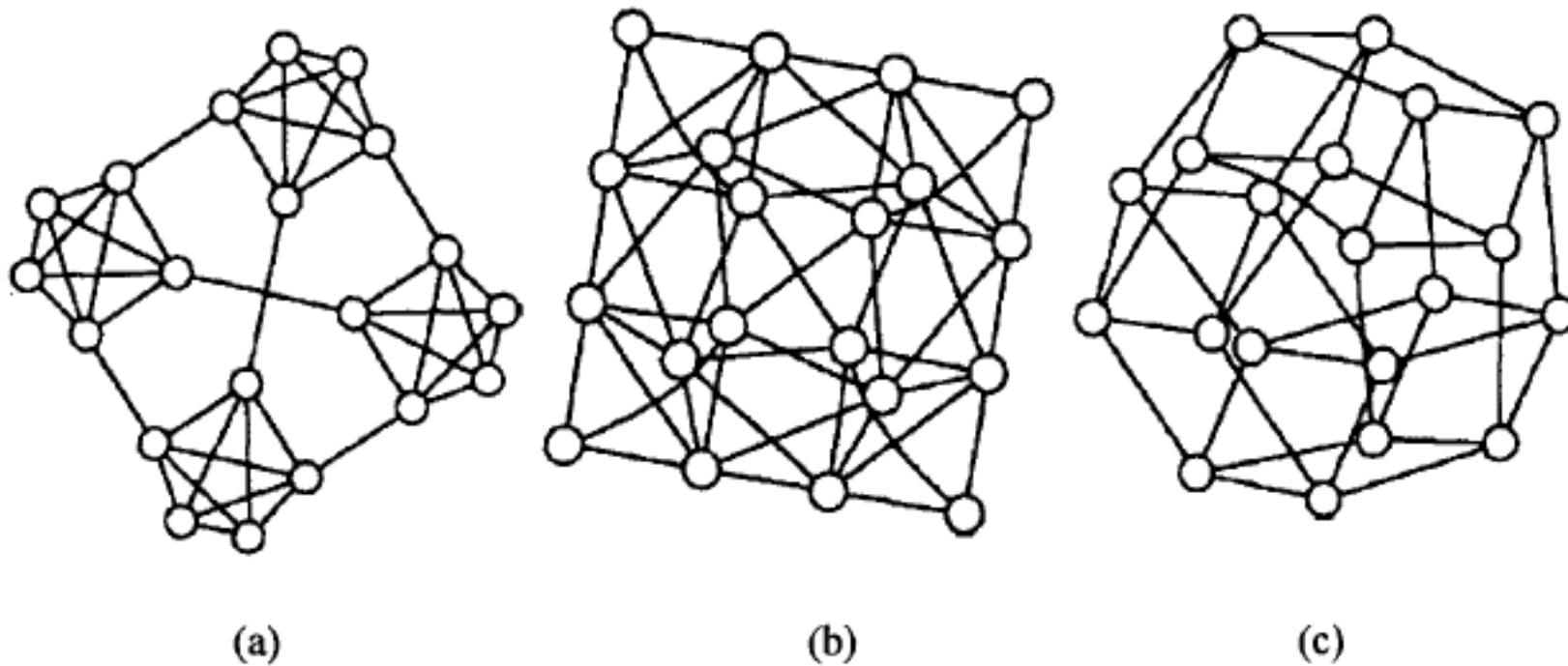


图 3.2 PSO 算法拓扑结构图 (a) 4 类; (b) 金字塔; (c) 冯·诺以曼

PSO公式变换

令 $\varphi = \varphi_1 + \varphi_2$, $P_m = (\varphi_1 \cdot P_i + \varphi_2 \cdot P_l) / \varphi$ 代表粒子在搜索空间中的收敛点, 则对 (3.6) 式的粒子群优化算法迭代式进行变换可得:

$$v(t+1) = \chi(v(t) + \varphi(P_m - X(t))) \quad (3.8)$$

其中 φ_{\max} 为 φ 的最大值, φ_1 和 φ_2 分别为 0 至 $\varphi_{\max} / 2$ 之间随机数, P_i 是第 i 粒子达到的最好位置, P_l 是其邻居粒子中达到的最好位置, φ_1 代表粒子向自身最优值推进的随机加速权值, φ_2 代表粒子向局部邻居最优值推进的随机加速权值。

Pm计算方法

➤ PSO: 每个解都同等考虑

$$\varphi_k = U[0, \frac{\varphi_{\max}}{m_n}] \quad \forall k \in m_n$$

$$\varphi = \sum_{k \in m_n} \varphi_k$$

$$P_m = \frac{\sum_{k \in m_n} \varphi_k P}{\varphi}$$

PSO: 加权方法

$$P_m = \frac{\sum_{k \in N} \frac{\varphi_k P_k}{f_k}}{\sum_{k \in N} \frac{\varphi_k}{f_k}}$$

其中 f_k 为第 k 个粒子的最优适应值

性能比较

表3.1 测试函数参数

函数	维数	初始范围	停止标准
sphere	30	± 100	0.01
Rastrigin	30	± 5.12	100
Griewank	10	± 600	0.05
Griewank	30	± 600	0.05
Rosenbrock	30	± 30	100
Shaffer f6	2	± 100	0.0001

测试方法

- 为比较粒子群优化算法采用不同拓扑结构时的寻优性能，可从两个方面来进行研究。其一，在给定迭代次数的情况下，比较各函数获得最优函数值。这里设定最大迭代次数Tmax为1000。
- 其二，通过比较在给定计算精度情况下，比较各算法所需平均迭代次数。同时，设定迭代上限为10000次。如此一来，算法或者达到计算精度，或者达到10000次计算终止。。

表 3.2 测试结果

方法	Neighborhood	Proportion	PropRank	CritRank	PerfRank
PSOWN	lbest	100.00%	1	19	22
PSO	vneumm	98.75%	2	4	1
PSO	slbest	98.75%	2	11	14
PSOWN	slbest	98.33%	4	5	6
PSO	lbest	96.67%	5	22	23
PSOWN	vneumm	96.25%	6	3	2
PSO	vneumm	92.50%	7	10	10
PSO	four clusters	92.08%	8	9	15
PSO	pyramid	91.67%	9	6	5
PSOWN	four clusters	91.67%	9	2	9
PSO	four clusters	91.25%	11	7	7
PSO	slbest	91.25%	11	13	20
PSO	lbest	90.83%	13	12	12
PSOWN	pyramid	90.00%	14	1	8
PSO	svneumm	87.50%	15	18	16
PSO	pyramid	87.08%	16	17	13
PSOWN	svneumm	86.67%	17	8	3
PSO	spyramid	85.42%	18	16	17
PSO	svneumm	81.25%	19	20	4
PSOWN	spyramid	78.33%	20	21	11
PSO	gbest	75.42%	21	15	19
PSO	sgbest	75.42%	21	14	21
PSO	spyramid	70.42%	23	23	18
PSOWN	gbest	22.08%	24	25	25
PSOWN	sgbest	18.33%	25	26	26
PSO	gbest	16.67%	26	24	27
PSO	sgbest	13.75%	27	27	24

分级性能指标

- ①运行成功达到各函数优化精度的等级称之为比例等级(**PropRank**);
- ②达到各函数优化精度所需平均迭代次数的等级称之为达标等级(**CrtiRank**);
- ③取迭代次数**1000**时的各函数寻优平均性能的等级称之为性能等级吃**PerfRank**)。

在表中有些结构名称前加了**s**:是指拓扑结构中
包含自身，也就是该粒子包含在它的邻居
中。

结果分析

- 所有结构当中，每次都能达到给定精度是局部近邻PSOWN，但是其性能等级是22。换句话说，该结果总是能找到全局最优解，但是需花费很长时间。
- 如果性能为首要考虑因素，也就是在1000代时函数最优值，则冯.诺以曼结构的PSON最好。这种结构也有很高的成功概率，比例等级是第2，成功比例是98.75%，并且是最快算法之一，达标所需平均迭代数排第4。

其他分析

- 成功概率性能最好的近邻结构是局部模型；
- 根据比例等级和性能等级两项指标则冯·诺以曼模型最优。后者是一个全面较优的算法，应当考虑替换当前采用较多的其它模型。
- 全局模型是一个较差的模型，无论PSON还是PSOWN。
- 究其原因，如果每个个体都被整个种群所影响，则由种群所导出的信息是相同的，都在种群的重心，从而由于缺少多样性阻碍了探测的进行。
- 如果每个个体的近邻粒子较少，则每个个体将有不同的搜索空间，故可保持良好的性能。

-
- **PSON和PSOWN所得结果都比标准PSO要好。**
 - **标准PSO所得最好结果为冯·诺以曼结构，比例等级第7，性能等级第10。**
 - **尤其，PSON和PSOWN能够使10维Griewank函数收敛，而标准PSO则不能。**

表 3.3 采用不同结构的算法对各函数寻优结果的平均性能比较

方法	Neigh	Sph	Ros	Gri30	Rast	Gri10	Sch
PSO	vneumm	2.62e-13	55.28001	0.004797	18.55624	0.014131	0.000863
PSOWN	vneumm	3.24e-15	41.07052	0.001905	19.45144	0.009542	0.002397
PSOWN	slbest	1.99e-14	56.63672	0.006164	67.66419	0.021161	0.001492
PSO	vneumm	2.92e-06	112.8217	0.012733	66.43264	0.049096	0.001946
PSO	lbest	0.000262	129.2118	0.009787	80.90703	0.046919	0.001945
PSO	slbest	3.11e-13	55.18398	0.009572	90.0651	0.043936	0.001952
PSO	svneumm	3.14e-08	93.4288	0.014248	74.79059	0.066009	0.002916
PSO	gbest	1.34e-09	77.63808	0.030719	70.24394	0.08057	0.004615
PSO	slbest	7.34e-06	117.461	0.010339	84.50814	0.051867	0.005105
PSO	sgbest	1.09e-08	129.6299	0.811115	78.22841	0.094794	0.004372
PSOWN	lbest	36.50434	3584.292	1.377846	182.3623	0.152637	0.001006
PSO	lbest	60.28522	12801.69	1.552355	178.3093	0.172	0.001592
PSO	sgbest	777.2214	66402.33	15.4314	106.8297	0.39283	0.007165
PSOWN	gbest	1197.985	104401.3	14.79815	85.97243	0.165539	0.00785
PSOWN	sgbest	1244.63	105420.1	14.325	103.51	0.190038	0.006315
PSO	gbest	1106.068	81166.54	16.07105	78.22476	0.259984	0.008197

推荐

- 局部模型和冯•诺以曼模型是两种较优的拓结构，在今后的应用中，值得进一步的推广。

小结

- PSO原理
- PSO算法中信息交换的拓扑结构