高级人工智能

沈华伟
中国科学院计算技术研究所
2020.11.17

课程回顾

三大学派
会发生什么?
行为主义
联结主义
符号主义
强化学习
涌现智能
……
神经网络
深度学习
……
符号逻辑
专家系统
……
DEEP BLUE
AlphaGo
Lee Sedol
发展阶段
规则驱动的
确定性智能
数据驱动的
不确定性智能
交互驱动的
涌现智能

群体智能

- 群体智能指的是无智能或者仅具有相对简单智能的主体通过合作涌现出更高智能行为的特性
  - 其中的个体并非绝对的无智能或只具有简单智能，而是相对于群体表现出来的智能而言是简单的。
- 单个复杂个体可以实现的功能，同样可以由大量简单的个体通过群体合作实现，后者的优势在于它更健壮、灵活和经济。
- 群体智能利用群体优势，在没有中心控制的条件下，寻找解决复杂问题的新思路

群体智能

- 集群智能
  - 众多无智能的个体，通过相互之间的简单合作所表现出来的智能行为
- 博弈
  - 具备一定智能的理性个体，按照某种机制行动，在群体层面体现出的智能
- 众包
  - 设计合适的机制，激励个体参与，从而实现单个个体不具备的社会智能

集群智能

- 集群智能是分布式、自组织的（自然/人造）系统表现出的一种群体智能
- 集群智能系统一般由一群简单的智能体构成，智能体按照简单的规则彼此进行局部交互，智能体也可以环境交互
- 灵感通常来自生物系统
  - 蚁群、鸟群、兽群
  - 粒子群

集群智能

鸟群

http://www.scs.carleton.ca/~arpwhite/courses/95590Y/notes/SI%20Lecture%203.pdf

集群智能

鱼群

http://www.scs.carleton.ca/~arpwhite/courses/95590Y/notes/SI%20Lecture%203.pdf

集群智能

蜂群

http://www.scs.carleton.ca/~arpwhite/courses/95590Y/notes/SI%20Lecture%203.pdf

集群智能

蚁群

http://www.scs.carleton.ca/~arpwhite/courses/95590Y/notes/SI%20Lecture%203.pdf

集群智能

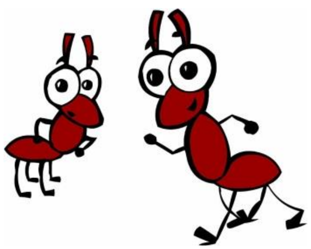
- 特点
  - 分布式：无中心控制
  - 随机性：非确定性
  - 自适应：个体根据环境进行策略调整
  - 正反馈：个体好的尝试会对个体产生正反馈
  - 自发涌现：会在群体层面涌现出一种智能

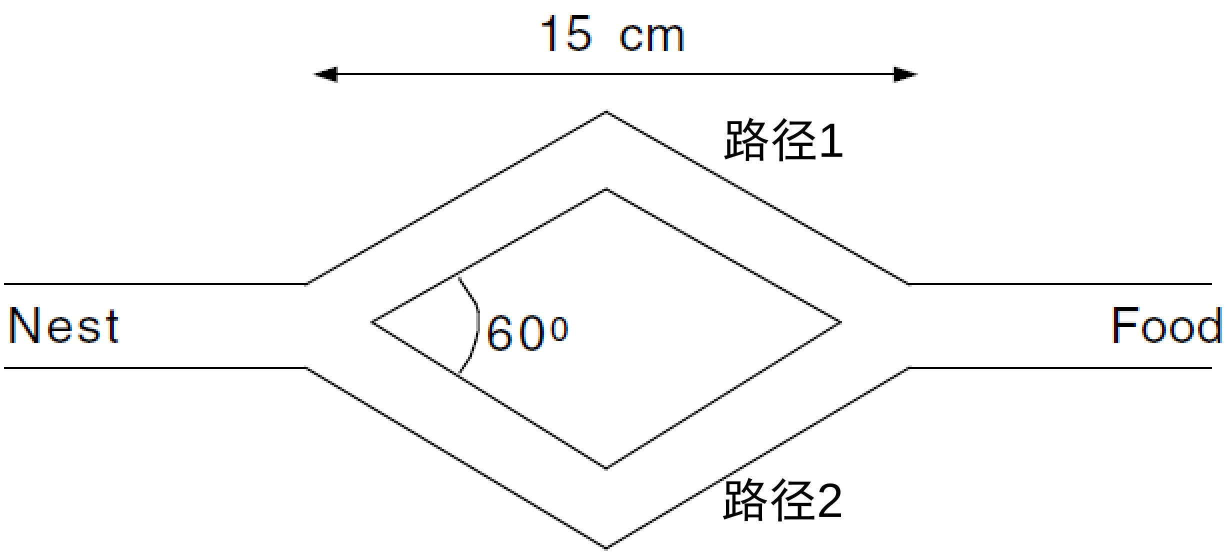
集群智能

- 代表性方法
  - 蚁群优化算法
  - 粒子群优化算法

蚁群寻食

- 等长路径的情形

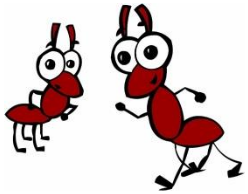


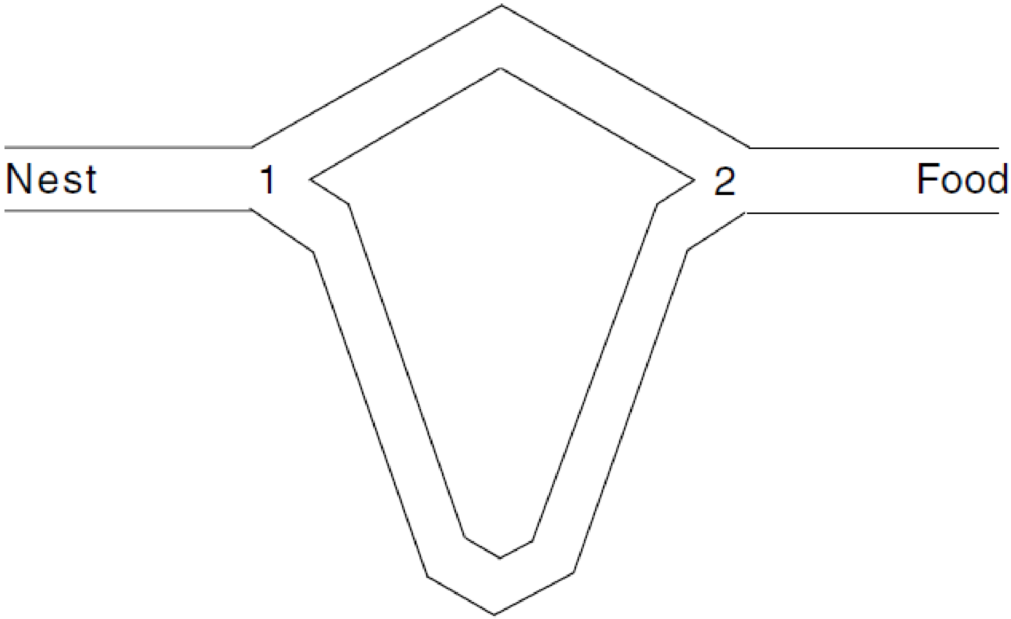


选择走路径1的蚂蚁和选择走路径2的蚂蚁数目相近

蚁群寻食

- 不等长路径的情形

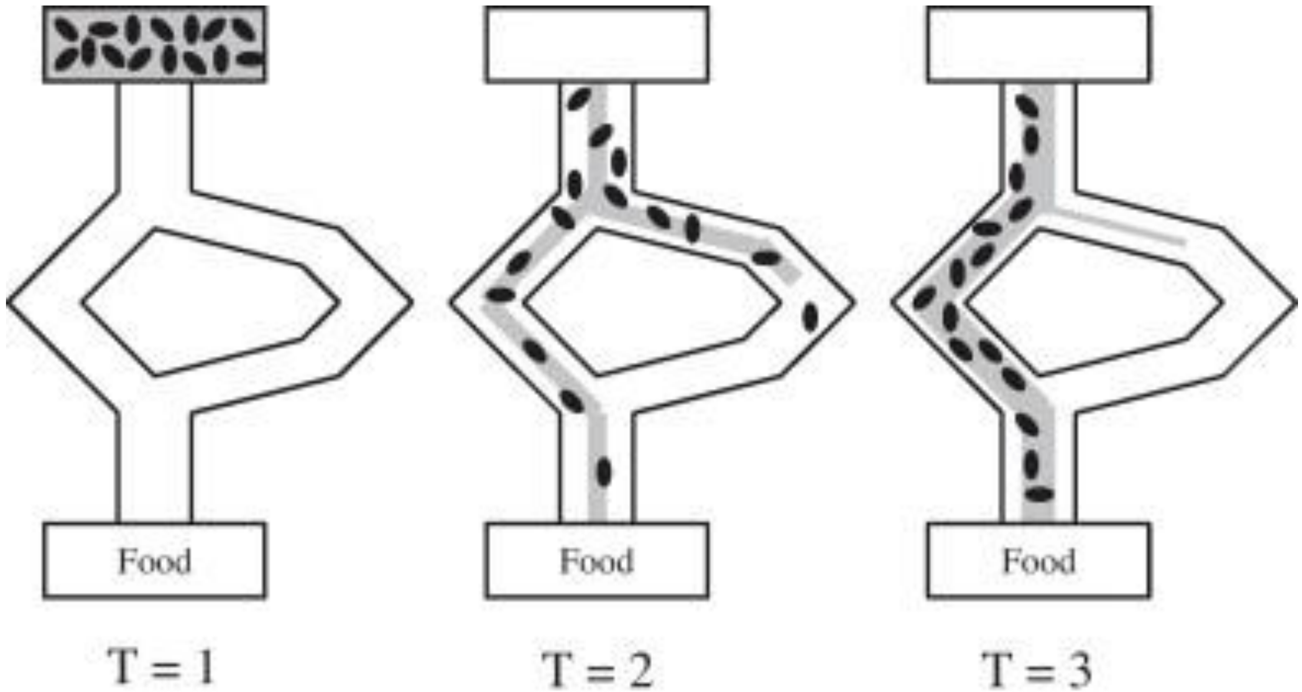




结果如何呢？

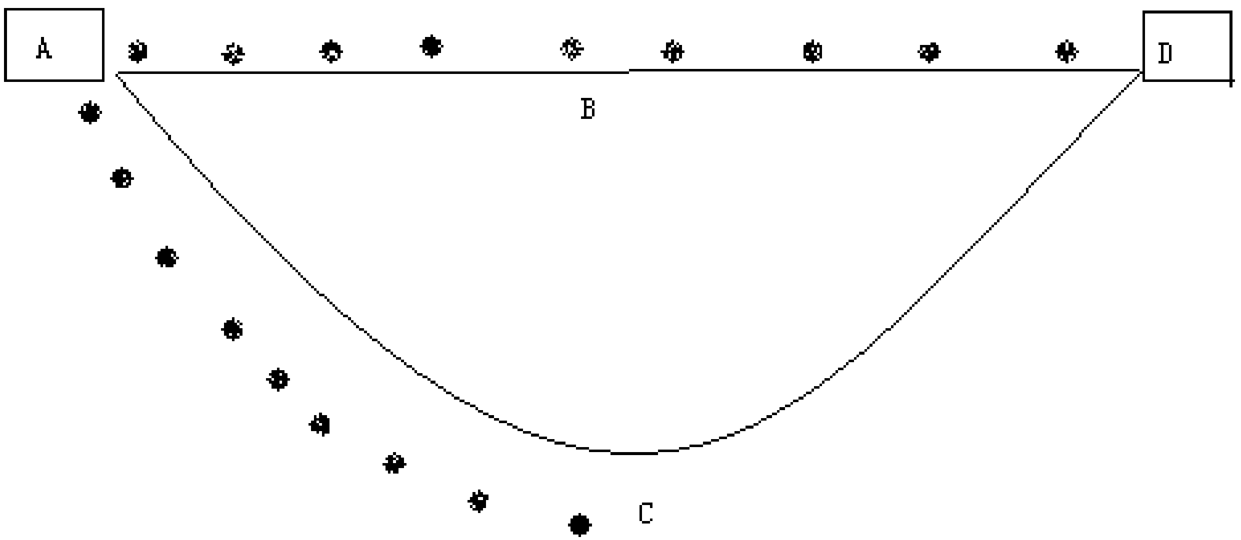
蚁群寻食

- 不等长路径的情形



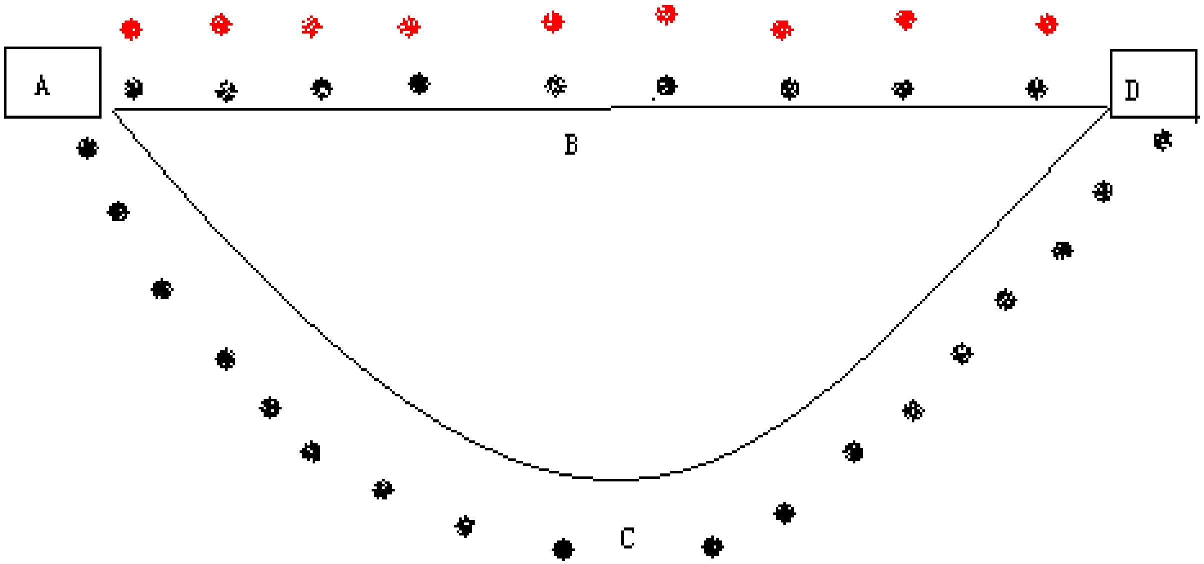
绝大多数蚂蚁选择长度较短的路径

蚁群寻食过程分析



- 蚂蚁从A点出发，速度相同，食物在D点，可能随机选择路线ABD或ACD。
- 假设初始时每条路线分配一只蚂蚁，每个时间单位行走一步，本图为经过9个时间单位时的情形
- 走ABD的蚂蚁到达终点，而走ACD的蚂蚁刚好走到C点，为一半路程。

蚁群寻食过程分析



经过18个时间单位时的情形：

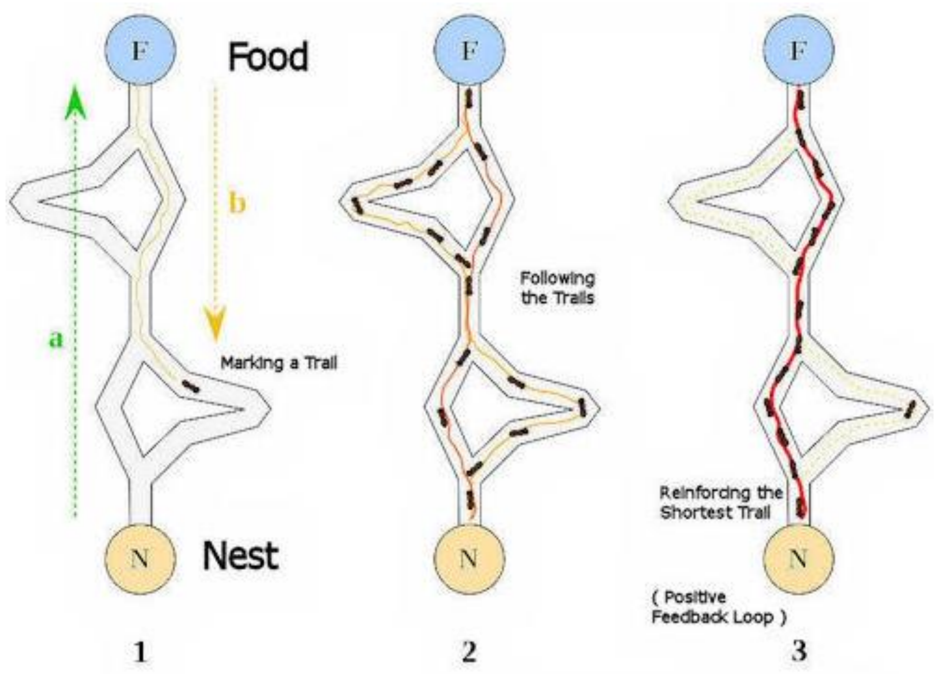
走ABD的蚂蚁到达终点后得到食物又返回了起点A，而走ACD的蚂蚁刚好走到D点。

蚁群寻食过程分析

- 假设蚂蚁每经过一处所留下的信息素为一个单位，则经过36个时间单位后，所有开始一起出发的蚂蚁都经过不同路径从D点取得了食物，此时ABD的路线往返了2趟，每一处的信息素为4个单位，而 ACD的路线往返了一趟，每一处的信息素为2个单位，其比值为**2：1**。
- 寻找食物的过程继续进行，则按信息素的指导，蚁群在ABD路线上增派一只蚂蚁（共2只），而ACD路线上仍然为一只蚂蚁。再经过36个时间单位后，两条线路上的信息素单位积累为12和4，比值为**3：1**。
- 若按以上规则继续，蚁群在ABD路线上再增派一只蚂蚁（共3只），而ACD路线上仍然为一只蚂蚁。再经过36个时间单位后，两条线路上的信息素单位积累为24和6，比值为**4：1**。
- 若继续进行，则按信息素的指导，最终所有的蚂蚁会放弃ACD路线，而都选择ABD路线。

蚁群优化算法

- **ACO: Ant Colony Optimization**
  - 一种解空间搜索方法
  - 适用于在图上寻找最优路径



A. Colorni, M. Dorigo et V. Maniezzo, Distributed Optimization by Ant Colonies, actes de la première conférence européenne sur la vie artificielle, Paris, France, Elsevier Publishing, 134-142, 1991.

蚁群优化算法

- 形式化
  - 每个蚂蚁对应一个计算智能体
  - 蚂蚁依概率选择候选位置进行移动
  - 在经过的路径上留下“信息素”（Pheromone）
  - “信息素”随时间挥发
  - “信息素”浓度大的路径在后续的选择中会以更高的概率被选取

旅行商问题的蚁群优化求解

- 旅行商问题（TSP: Traveling Salesman Problem）
  - n个城市的有向图$G=(V,E)$

$$V=\{1,2,\cdots,n\} \qquad \mathrm{E}=\{(i,j)|i,j\in V\}$$

  - 城市之间的距离表示为

d_{ij} 为节点i和j之间的距离

  - 目标函数

$$f(w)=\sum_{l=1}^{n} d_{i_l i_{l+1}}$$

$w=(i_1,i_2,\cdots,i_n)$ 为TSP问题的任意可行解，其中 $i_{n+1}=i_1$

旅行商问题的蚁群优化求解

首先将m只蚂蚁随机放置在n个城市，位于城市i的第k只蚂蚁选择下一个城市j的概率为:

差一个更新规则

$$p_{ij}^{k}(t)=\begin{cases}\dfrac{\left(\tau_{ij}(t)\right)^{\alpha}\left(\eta_{ij}(t)\right)^{\beta}}{\sum_{k\in allowed}\left(\tau_{ik}(t)\right)^{\alpha}\left(\eta_{ik}(t)\right)^{\beta}} & j\in allowed\\ 0, & otherwise\end{cases} \tag{1}$$

$\tau_{i,j}(t)$ 表示边(i,j)上的信息素浓度

$\eta_{i,j}(t)=1/d_{ij}$　是根据距离定义的启发信息

α和β反映了信息素与启发信息的相对重要性

旅行商问题的蚁群优化求解

当所有蚂蚁完成周游后，按以下公式进行信息素更新

$$\Delta\tau_{ij}^{k} = f(x) = \begin{cases} \dfrac{Q}{L_k}, & (i,j) \in w_k \\ 0, & otherwise \end{cases} \tag{2}$$

$$\tau_{ij}(t+1) = \rho \cdot \tau_{ij}(t+1) + \Delta\tau_{ij}$$

$$\Delta\tau_{ij} = \sum_{k=1}^{m} \Delta\tau_{ij}^{k}$$

其中：Q为常数, w_k表示第k只蚂蚁在本轮迭代中走过的路径，L_k为路径长度，ρ为小于1的常数，反映信息素挥发速度

旅行商问题的蚁群优化求解

TSP问题蚁群算法流程

⑴初始化　随机放置蚂蚁，

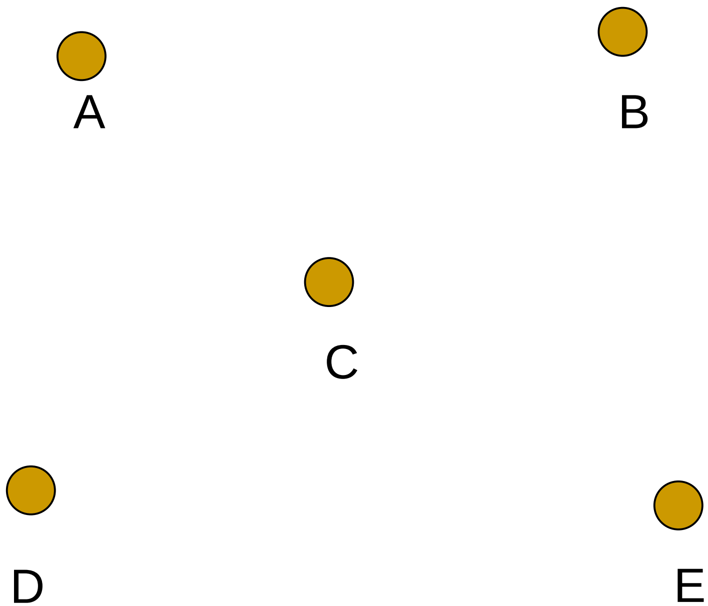
⑵迭代过程

```
k=1
while k=<ItCount do (执行迭代)
      for i = 1 to m do  (对m只蚂蚁循环)
        for j = 1 to n - 1 do  (对n个城市循环)
            根据式(1)，采用轮盘赌方法在窗口外选择下一个城市j;
        将j置入禁忌表,蚂蚁转移到j;
     end for
  end for
  计算每只蚂蚁的路径长度;
  根据式(2)更新所有蚂蚁路径上的信息量;
  k = k + 1;
end while
```

⑶输出结果,结束算法.

旅行商问题的蚁群优化求解

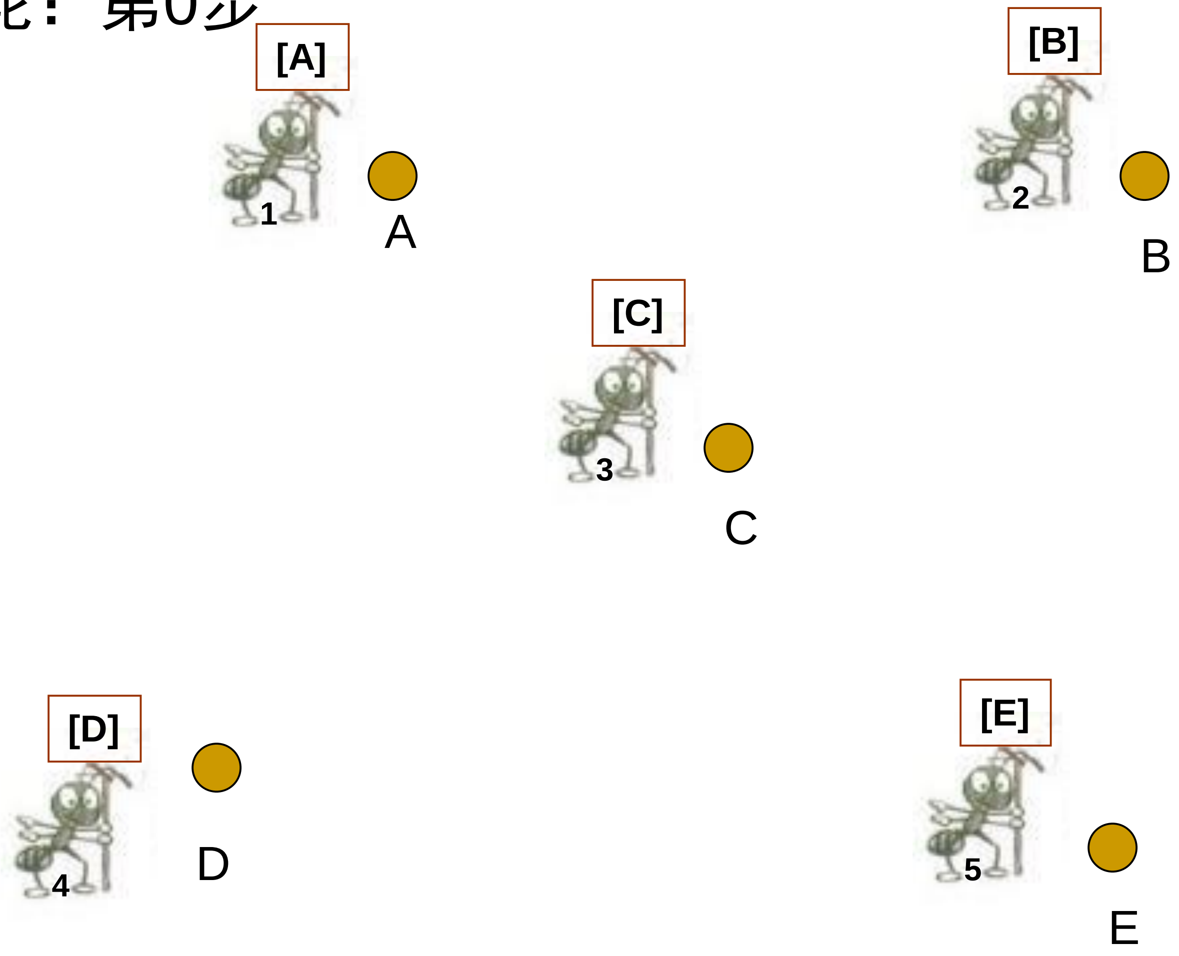
- 旅行商问题（TSP: Traveling Salesman Problem）



节点间距离： $d_{AB}=100; d_{BC}=60\ldots; d_{DE}=150$

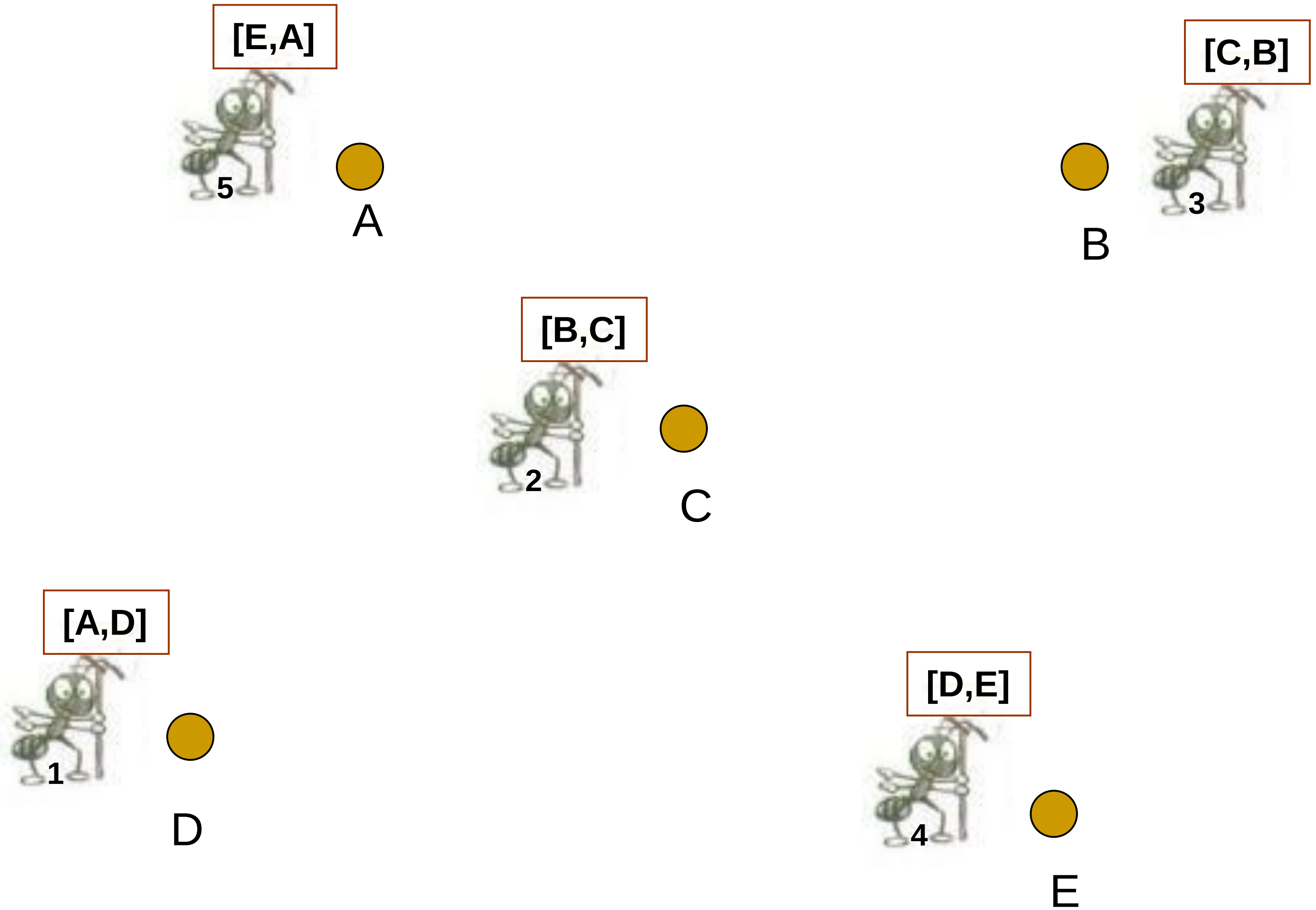
旅行商问题的蚁群优化求解

- 第t轮：第0步



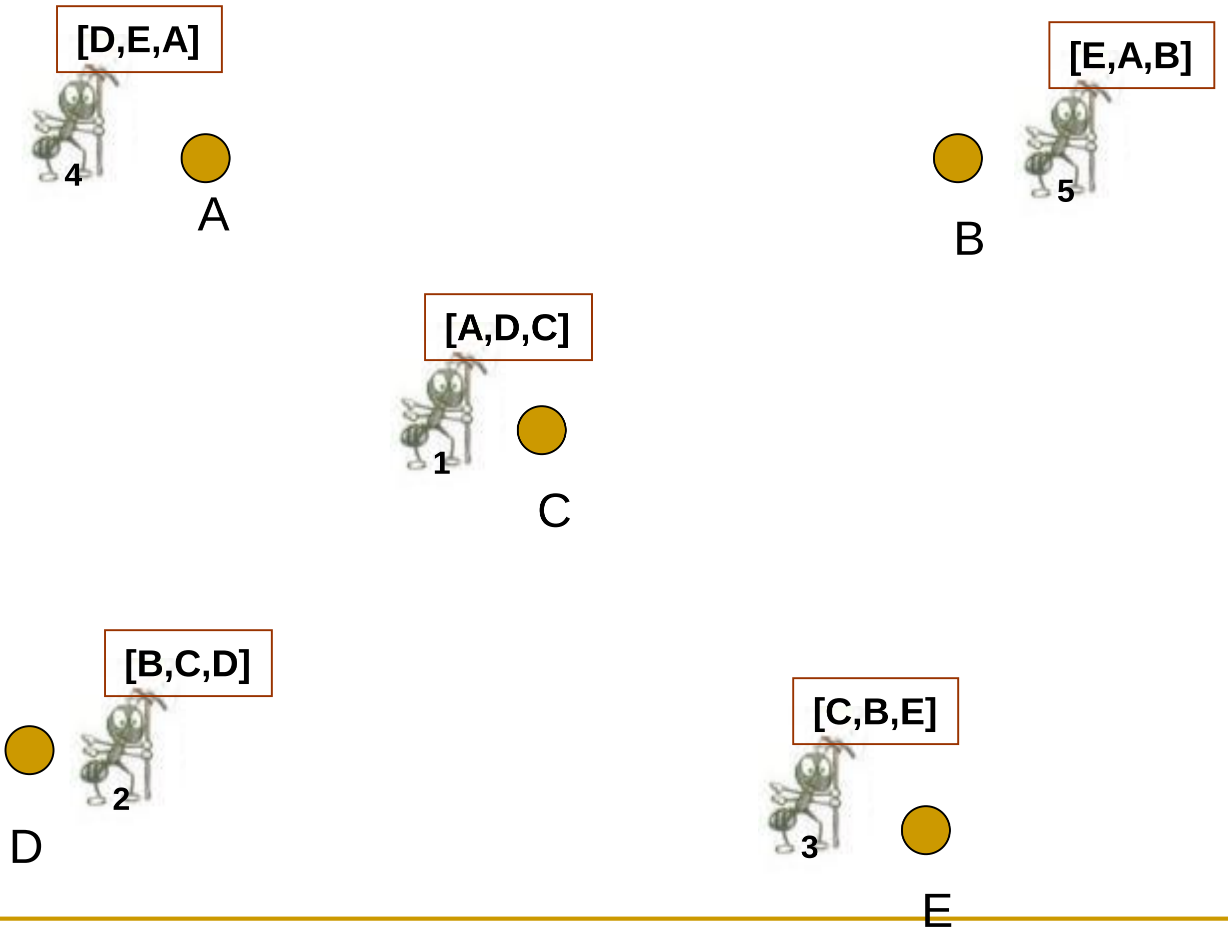
旅行商问题的蚁群优化求解

- 第t轮：第1步



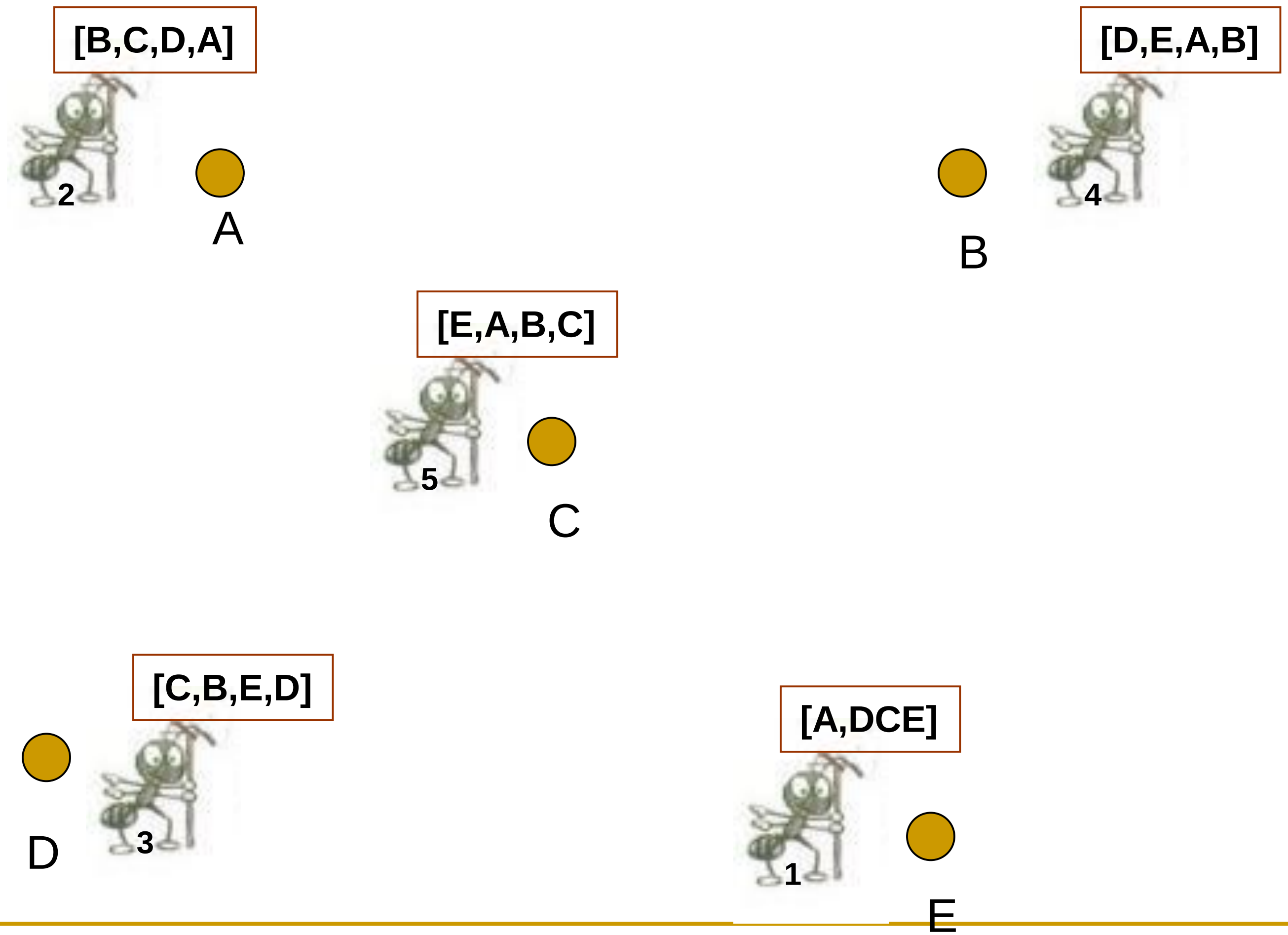
旅行商问题的蚁群优化求解

- 第t轮：第2步



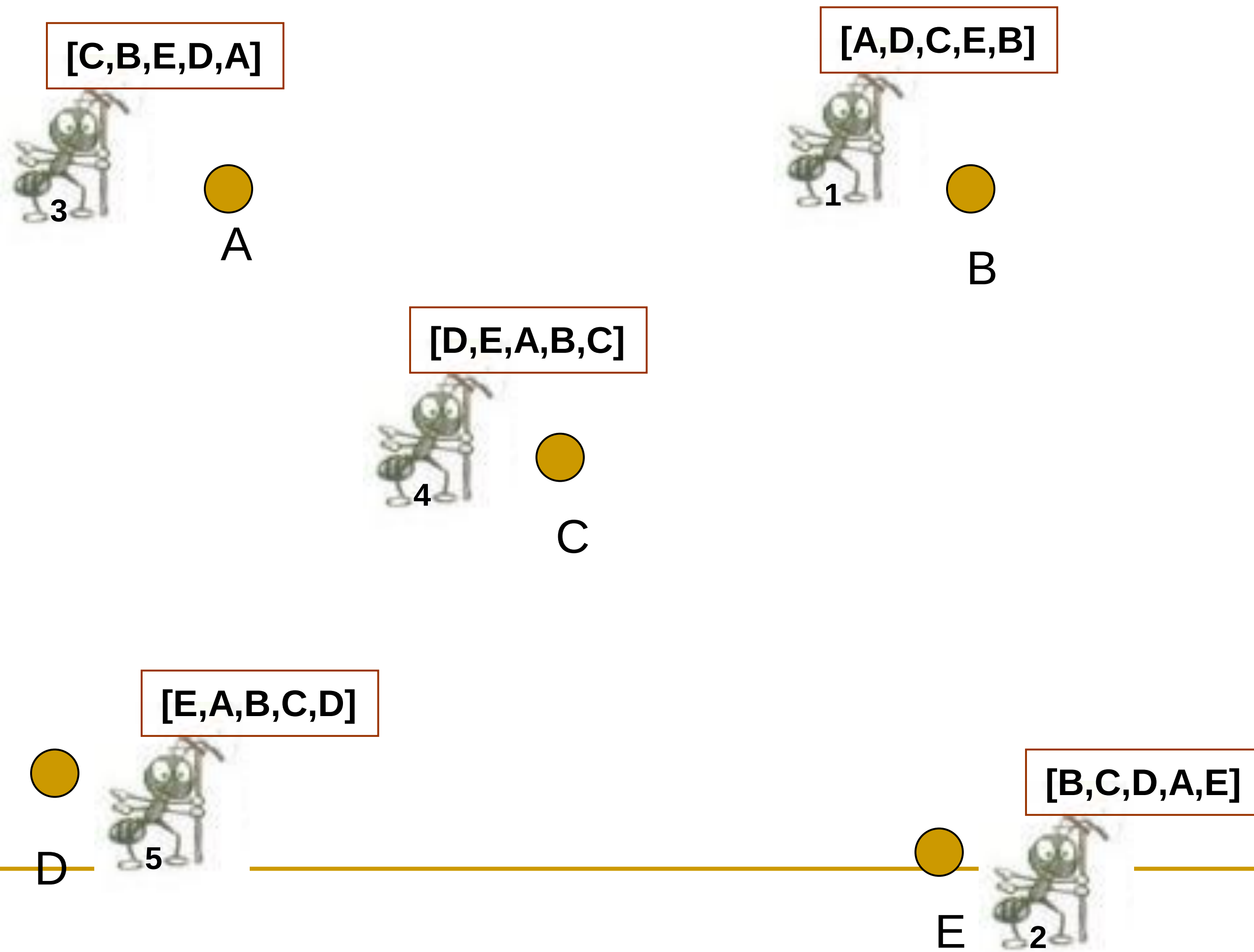
旅行商问题的蚁群优化求解

- 第t轮：第3步



旅行商问题的蚁群优化求解

- 第t轮：第4步



旅行商问题的蚁群优化求解

- 计算路径长度

[A,D,C,E,B]

$$L_1 = d_{AD} + d_{DC} + d_{CE} + d_{EB}$$

[B,C,D,A,E]

$$L_2 = d_{BC} + d_{CD} + d_{DA} + d_{AE}$$

[C,B,E,D,A]

$$L_3 = d_{CB} + d_{BE} + d_{ED} + d_{DA}$$

[D,E,A,B,C]

$$L_4 = d_{DE} + d_{EA} + d_{AB} + d_{BC}$$

[E,A,B,C,D]

$$L_5 = d_{EA} + d_{AB} + d_{BC} + d_{CD}$$

旅行商问题的蚁群优化求解

- 蚁群大小
  - 一般情况下，蚁群中的蚂蚁个数不超过TSP图中节点的个数
- 终止条件
  - 设定迭代轮数
  - 设定最优解连续保持不变的迭代轮数

蚁群优化算法小结

- 思想
  - 局部随机搜索+自增强
  - 鲁迅：“世界本无路，走的人多了也就有了路”

- 缺点
  - 收敛速度慢
  - 易于陷入局部最优
  - 对于解空间为连续的优化问题不适用

课间休息

集群智能

- 代表性方法
  - 蚁群优化算法
  - 粒子群优化算法

粒子群优化算法

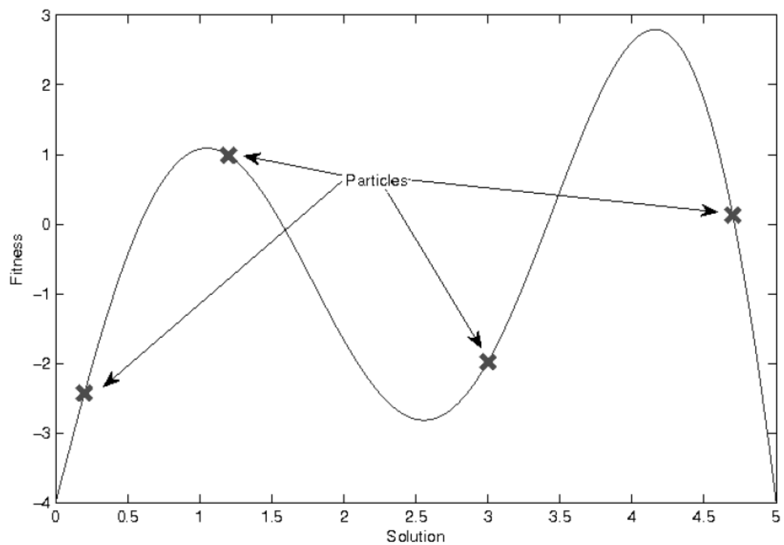
- 粒子群优化算法是一种基于种群寻优的启发式搜索算法。在1995年由Kennedy 和Eberhart 首先提出来的。
- 它的主要启发来源于对鸟群群体运动行为的研究。我们经常可以观察到鸟群表现出来的同步性，虽然每只鸟的运动行为都是互相独立的，但是在整个鸟群的飞行过程中却表现出了高度一致性的复杂行为，并且可以自适应的调整飞行的状态和轨迹。
- 鸟群具有这样的复杂飞行行为的原因，可能是因为每只鸟在飞行过程中都遵循了一定的行为规则，并能够掌握邻域内其它鸟的飞行信息。

粒子群优化算法

- 粒子群优化算法借鉴了这样的思想，每个粒子代表待求解问题搜索解空间中的一个潜在解，它相当于一只鸟，“飞行信息”包括粒子当前的位置和速度两个状态量。
- 每个粒子都可以获得其邻域内其它个体的信息，对所经过的位置进行评价，并根据这些信息和位置速度更新规则，改变自身的两个状态量，在“飞行”过程中传递信息和互相学习，去更好地适应环境。
- 随着这一过程的不断进行，粒子群最终能够找到问题的近似最优解。

粒子群优化算法

- **PSO: Particle Swarm Optimization**
  - 一种随机优化方法
  - 通过粒子群在解空间中进行搜索，寻找最优解（适应度最大的解）



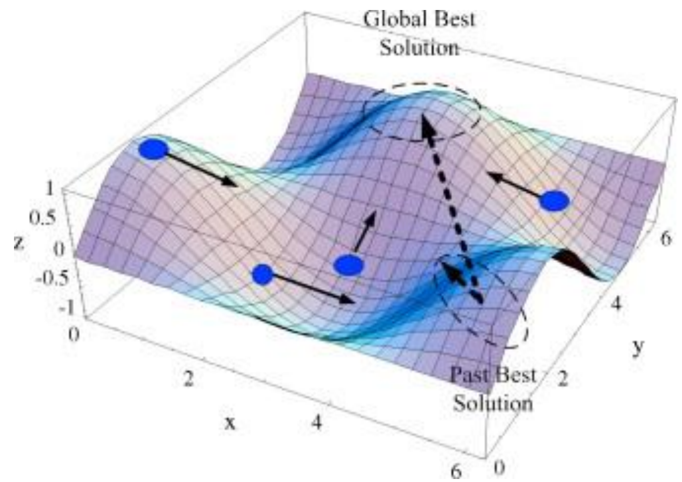
没有梯度
对每个解
打分

James Kennedy and Russell Eberhart. Particle swarm optimization. Proceedings of the IEEE International Conference on Neural Networks, pp. 1942–1948, Piscataway, NJ, 1995.

粒子群优化算法

- 构成要素
  - 粒子群
    - 每个粒子对应所求解问题的一个可行解
    - 粒子通过其位置和速度表示
      - 粒子i在第n轮的位置：$x_n^{(i)}$
      - 粒子i在第n轮的速度：$v_n^{(i)}$
  - 记录
    - $p_{best}^{(i)}$：粒子i的历史最好位置
    - g_{best}：全局历史最好位置
  - 计算适应度的函数
    - 适应度：$f(x)$

fitness



粒子群优化算法

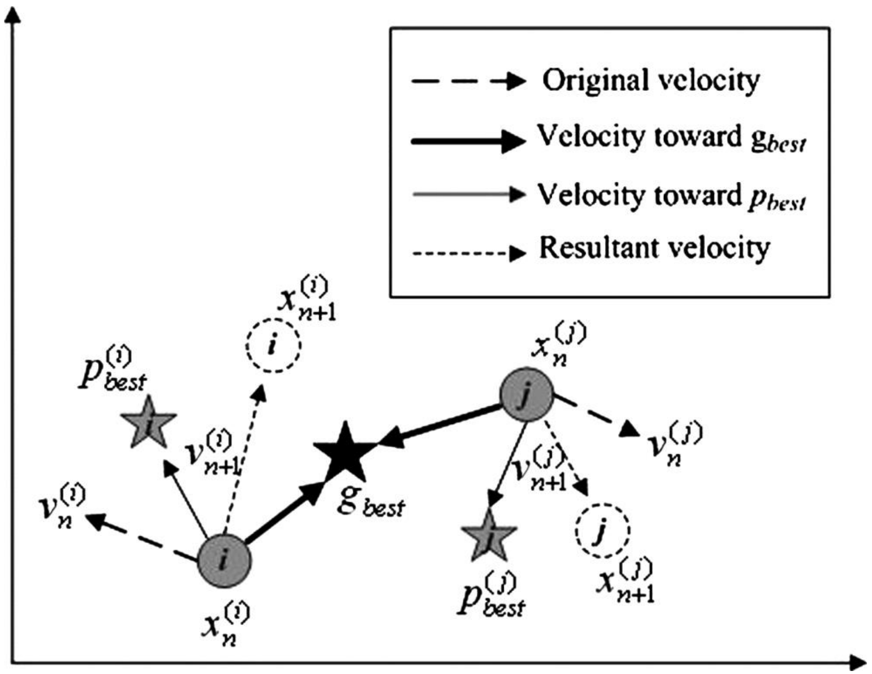
- 算法过程描述
  - 初始化
    - 初始化粒子群：每个粒子的位置和速度，即$x_0^{(i)}$和$v_0^{(i)}$
    - $p_{best}^{(i)}$和g_{best}
  - 循环执行如下三步直至满足结束条件
    - 计算每个粒子的适应度：$f\left(x_n^{(i)}\right)$
    - 更新每个粒子历史最好适应度及其相应的位置，更新当前全局最好适应度及其相应的位置
    - 更新每个粒子的速度和位置

$$v_{n+1}^{(i)} = v_n^{(i)} + c_1 * r_1 * \left(p_{best}^{(i)} - x_n^{(i)}\right) + c_2 * r_2 * \left(g_{best} - x_n^{(i)}\right)$$

$$x_{n+1}^{(i)} = x_n^{(i)} + v_{n+1}^{(i)}$$

粒子群优化算法

- 粒子位置和速度更新示例



粒子群优化算法

- 粒子速度更新公式解读

$$v_{n+1}^{(i)} = \boxed{v_n^{(i)}} + c_1 * r_1 * \boxed{\left(p_{best}^{(i)} - x_n^{(i)}\right)} + c_2 * r_2 * \boxed{\left(g_{best} - x_n^{(i)}\right)}$$

①惯性项

保持原速度不变的倾向

②记忆项

回到历史最好位置的倾向

③社会项

走向粒子群全局最好位置的倾向

粒子群优化算法

- 粒子速度更新公式解读

$$v_{n+1}^{(i)} = v_n^{(i)} + \boxed{c_1} * \boxed{r_1} * \left(p_{best}^{(i)} - x_n^{(i)}\right) + \boxed{c_2} * \boxed{r_2} * \left(g_{best} - x_n^{(i)}\right)$$

权重参数：一般取值为2

随机参数：0和1之间的随机数

粒子群优化算法

- 算法终止条件
  - 迭代的轮数
  - 最佳位置连续未更新的轮数
  - 适应度函数的值到达预期要求

粒子群优化算法

- 速度更新参数分析
  - 又称加速度参数，用来控制粒子当前最优位置$p_{best}^{(i)}$和粒子群当前最优位置g_{best}对粒子飞行速度的影响

- $c_1 > 0, c_2 = 0$： 每个微粒执行局部搜索；
- $c_1 = 0, c_2 > 0$： 微粒群转化为一个随机爬山法；
- $c_1 = c_2 > 0$： 微粒逐渐移向 $\vec{p}_g$ 和 $\vec{p}_i$ 的加权均值；
- $c_2 > c_1$： 算法比较适合于单峰优化问题；
- $c_1 > c_2$： 算法比较适合于多峰优化问题。

粒子群优化算法改进

- 惯性权重
  - 速度冲量导致微粒按照先前速度方向继续移动。Yuhui Shi[1]提出一个惯性权重w来控制先前微粒速度的影响

惯性权重

$$v_{n+1}^{(i)} = w * v_n^{(i)} + c_1 * r_1 * \left(p_{best}^{(i)} - x_n^{(i)}\right) + c_2 * r_2 * \left(g_{best} - x_n^{(i)}\right)$$

[1] Y. Shi, R. Eberhart. “A modified particle swarm optimizer,” Proceedings of IEEE World Congress on Computational Intelligence, Anchorage, AK, 1998, pp. 69-73.

粒子群优化算法

- 和遗传算法相比
  - 遗传算法强调“适者生存”，不好的个体在竞争中被淘汰；**PSO**强调“协同合作”，不好的个体通过学习向好的方向转变。
  - 遗传算法中最好的个体通过产生更多的后代来传播基因；**PSO**中的最好个体通过吸引其它个体向它靠近来施加影响。
  - 遗传算法的选择概率只与上一代群体相关，而与历史无关，群体的信息变化过程是一个**Markov**链过程；而**PSO**中的个体除了有位置和速度外，还有着过去的历史信息（**pBest**、**gBest**）。

粒子群优化算法

- 优点
  - 易于实现；
  - 可调参数较少；
  - 所需种群或微粒群规模较小；
  - 计算效率高，收敛速度快。
- 缺点
  - 和其它演化计算算法类似，不保证收敛到全局最优解

粒子群优化算法小结

- 一种随机优化算法
- 适用于求解连续解空间的优化问题

课后作业

- 实现一个粒子群优化算法，求解函数

$$f(x) = x^3 - 5x^2 - 2x + 3$$

在取值范围[-2,5]之间的最小值和最大值