第 6 章　模拟生物群体行为的群智能算法

教材：

王万良《人工智能应用教程》（2版）

清华大学出版社，2025

第 6 章　模拟生物群体行为的群智能算法

- 智能优化方法中受动物群体智能启发的算法称为群智能算法。
- 本章首先简要介绍群智能算法产生的背景，然后详细介绍粒子群优化算法、蚁群算法等群智能算法及其应用。

蜜蜂采蜜路径优化

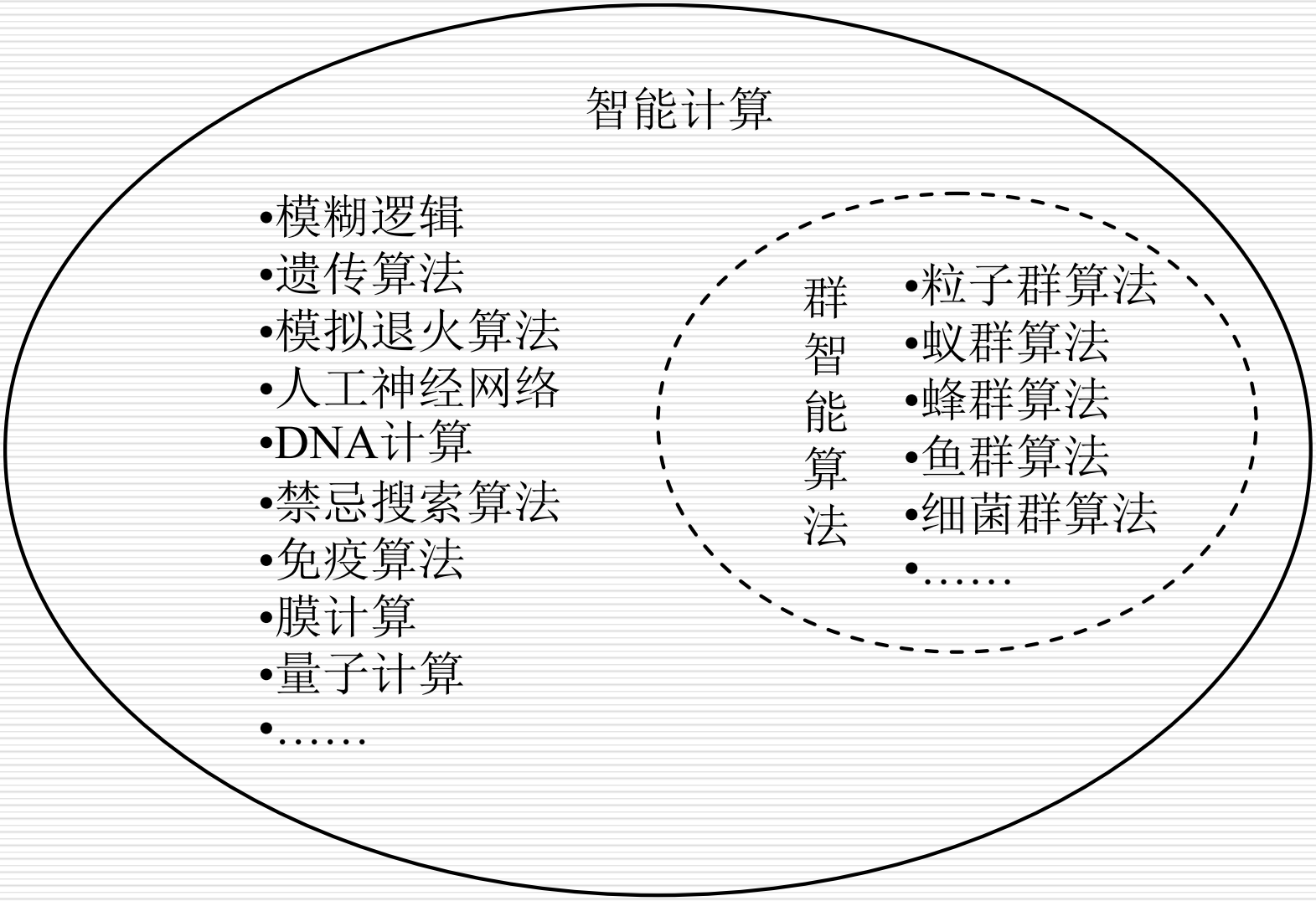
第 6 章　模拟生物群体行为的群智能算法

- 6.1　群智能算法的生物学背景
- 6.2　模拟鸟群行为的粒子群优化算法
- 6.3　模拟蚁群行为的蚁群优化算法

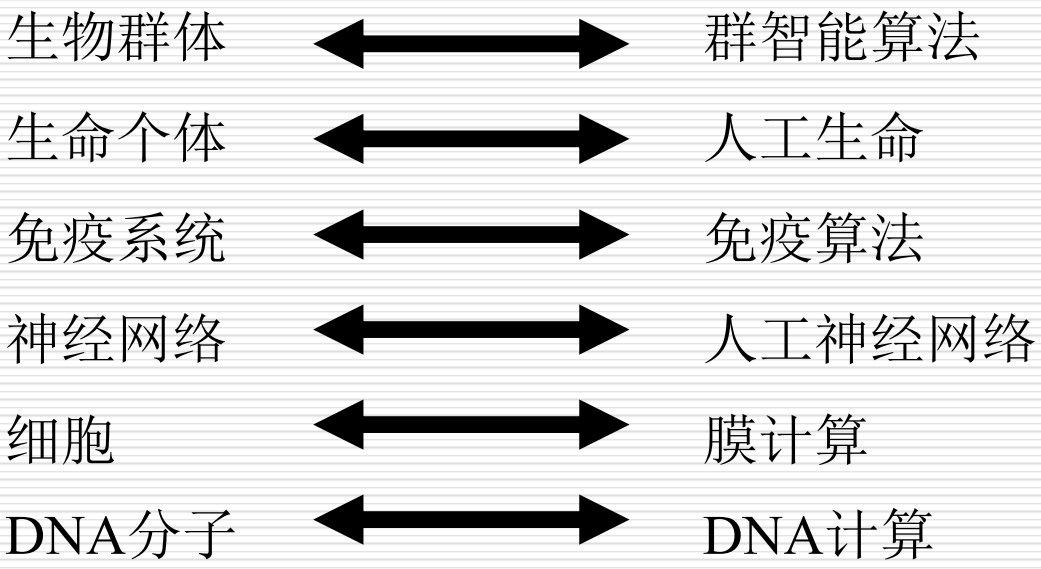
6.1 群智能算法的生物学背景

- 群智能算法（swarm algorithms，SI）：受动物群体智能启发的算法。
- 群体智能：由简单个体组成的群落与环境以及个体之间的互动行为。
- 群智能算法包括：粒子群优化算法、蚁群算法、蜂群算法、……

6.1 群智能算法的生物学背景



6.1 群智能算法的生物学背景



生物层次与仿生智能计算的对应关系

第 6 章　模拟生物群体行为的群智能算法

- 6.1 群智能算法的生物学背景
- 6.2 模拟鸟群行为的粒子群优化算法
- 6.3 模拟蚁群行为的蚁群优化算法

6.2 模拟鸟群行为的粒子群优化算法

- **产生背景：**粒子群优化（Particle Swarm Optimization, PSO）算法是由美国普渡大学的Kennedy和Eberhart于1995年提出，它的基本概念源于对鸟群觅食行为的研究。

- **设想这样一个场景：**一群鸟在随机搜寻食物，在这个区域里只有一块食物，所有的鸟都不知道食物在哪里，但是它们知道当前的位置离食物还有多远。那么找到食物的最优策略是什么呢？

最简单有效的就是搜寻目前离食物最近的鸟的周围区域。

6.2.1 基本粒子群优化算法

□ 基本思想

将每个个体看作*n*维搜索空间中一个没有体积质量的粒子，在搜索空间中以一定的速度飞行，该速度决定粒子飞行的方向和距离。所有粒子有一个由优化函数决定的适应值。

□ 基本原理

PSO初始化为一群随机粒子，然后通过迭代找到最优解。在每一次迭代中，粒子通过跟踪两个“极值”来更新自己。第一个就是粒子本身所找到的最优解，这个解称为个体极值。另个一是整个种群目前找到的最优解，这个解称为全局极值。

6.2.1 基本粒子群优化算法

□ 算法定义

在n 维连续搜索空间中，对粒子群中的第i (i=1, 2, …, m)个粒子进行定义：

- $x^i(k)=\begin{bmatrix} x_1^i & x_2^i & \cdots & x_n^i \end{bmatrix}^T$ ：表示搜索空间中粒子的当前位置。

- $p^i(k)=\begin{bmatrix} p_1^i & p_2^i & \cdots & p_n^i \end{bmatrix}^T$ ：表示该粒子至今所获得的具有最优适应度 $f_p^i(k)$ 的位置。

- $v^i(k)=\begin{bmatrix} v_1^i & v_2^i & \cdots & v_n^i \end{bmatrix}^T$ ：表示该粒子的搜索方向。

6.2.1 基本粒子群优化算法

群体经历过的最优位置(gbest)记为 $p^g(k)=\begin{bmatrix} p_1^g & p_2^g & \cdots & p_n^g \end{bmatrix}^T$，则

基本的PSO算法为：

$$v_j^i(k+1)=\omega(k)v_j^i(k)+\varphi_1 rand(0,a_1)\left(p_j^i(k)-x_j^i(k)\right)+\varphi_2 rand(0,a_2)\left(p_j^g(k)-x_j^i(k)\right)$$

——（6.1a）

$$x_j^i(k+1)=x_j^i(k)+v_j^i(k+1) \quad \text{——（6.1b）}$$

$$i=1,2,\cdots,m; \quad j=1,2,\cdots,n$$

其中，ω 是惯性权重因子。φ_1 ，φ_2 是加速度常数，均为非负值。$rand(0,a_1)$和$rand(0,a_2)$为[0, a_1]、[0, a_2]范围内的具有均匀分布的随机数，a_1 与 a_2 为相应的控制参数。

6.2.1 基本粒子群优化算法

$$v_j^i(k+1)=\omega(k)v_j^i(k)+\varphi_1 rand(0,a_1)\left(p_j^i(k)-x_j^i(k)\right)+\varphi_2 rand(0,a_2)\left(p_j^g(k)-x_j^i(k)\right)$$

- 第1部分是粒子在前一时刻的速度；
- 第2部分为个体“认知”分量，表示粒子本身的思考，将现有的位置和曾经经历过的最优位置相比。
- 第3部分是群体“社会(social)”分量，表示粒子间的信息共享与相互合作。
- φ_1， φ_2分别控制个体认知分量和群体社会分量相对贡献的学习率。
- 随机系数增加搜索方向的随机性和算法多样性。

6.2.1 基本粒子群优化算法

基于学习率 φ_1, φ_2，

Kennedy给出以下4种类型的PSO模型：

- 若 $\varphi_1 > 0$，$\varphi_2 > 0$，则称该算法为PSO全模型。

- 若 $\varphi_1 > 0$，$\varphi_2 = 0$，则称该算法为PSO认知模型。

- 若 $\varphi_1 = 0$，$\varphi_2 > 0$，则称该算法为PSO社会模型。

- 若 $\varphi_1 = 0$，$\varphi_2 > 0$且$g \neq i$，则称该算法为PSO无私模型。

6.2.1 基本粒子群优化算法

粒子群优化算法的流程：

（1）初始化每个粒子。在允许范围内随机设置每个粒子的初始位置和速度。

（2）评价每个粒子的适应度。计算每个粒子的目标函数。

（3）设置每个粒子的P_i。对每个粒子，将其适应度与其经 历过的最好位置 P_i进行比较，如果优于 P_i，则将其作为该粒子的最好位置 P_i。

6.2.1 基本粒子群优化算法

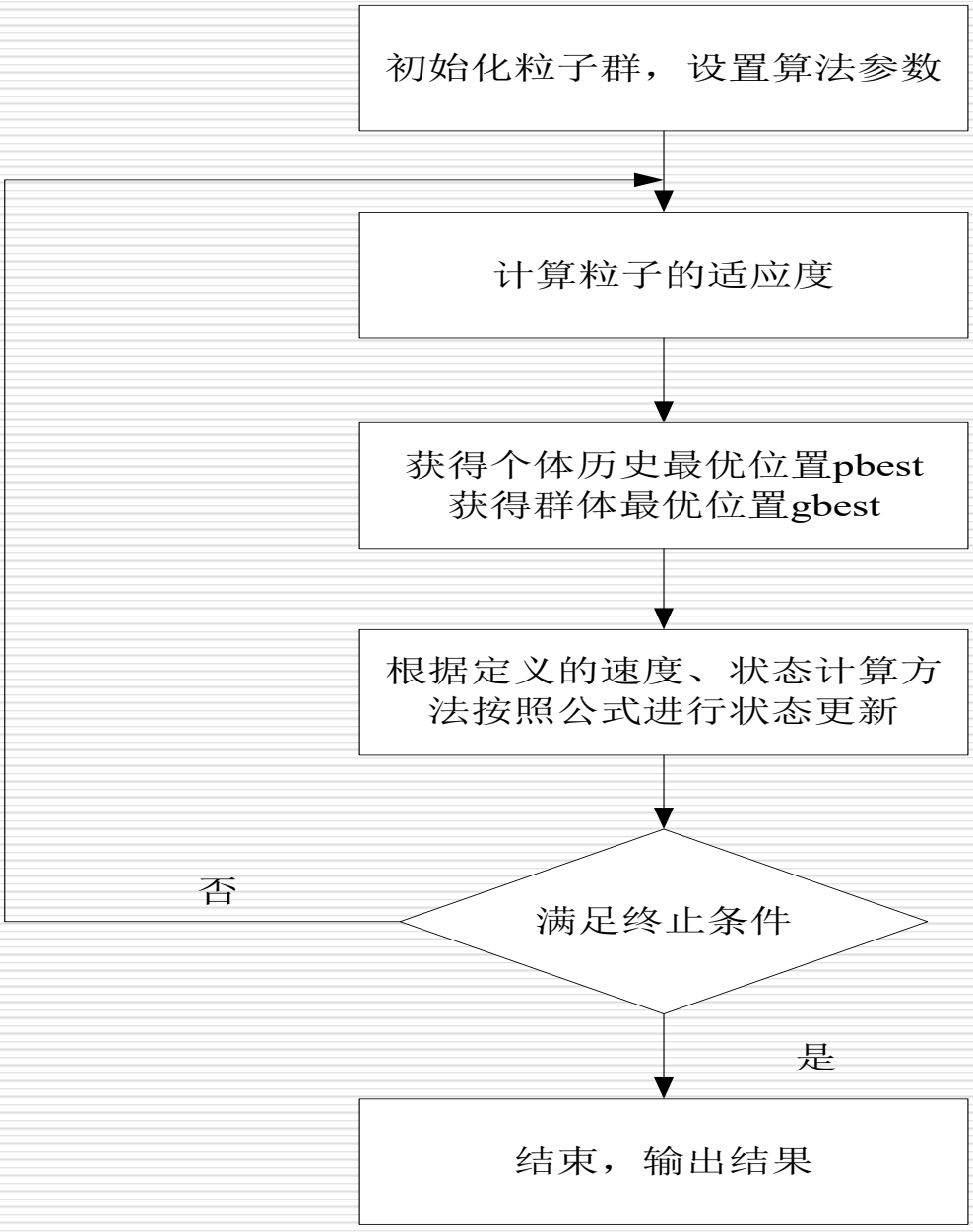
粒子群优化算法的流程：

（4）设置全局最优值 P_g **。**对每个粒子，将其适应度与群体经历过的最好位置 P_g 进行比较，如果优于 P_g，则将其作为当前群体的最好位置 P_g。

（5）更新粒子的速度和位置。根据式（6.20）更新粒子的速度和位置。

（6）检查终止条件。如果未达到设定条件（预设误差或者迭代的次数），则返回第（2）步。

粒子群优化算法流程图


初始化粒子群，设置算法参数
计算粒子的适应度
获得个体历史最优位置pbest
获得群体最优位置gbest
根据定义的速度、状态计算方法按照公式进行状态更新
满足终止条件
否
是
结束，输出结果

6.2.2 粒子群优化算法的应用

粒子群优化算法已在诸多领域得到应用，归纳如下：

（1）神经网络训练

（2）化工系统领域

（3）电力系统领域

（4）机械设计领域

（5）通讯领域

（6）机器人领域

（7）经济领域

（8）图像处理领域

（9）生物信息领域

（10）医学领域

（11）运筹学领域

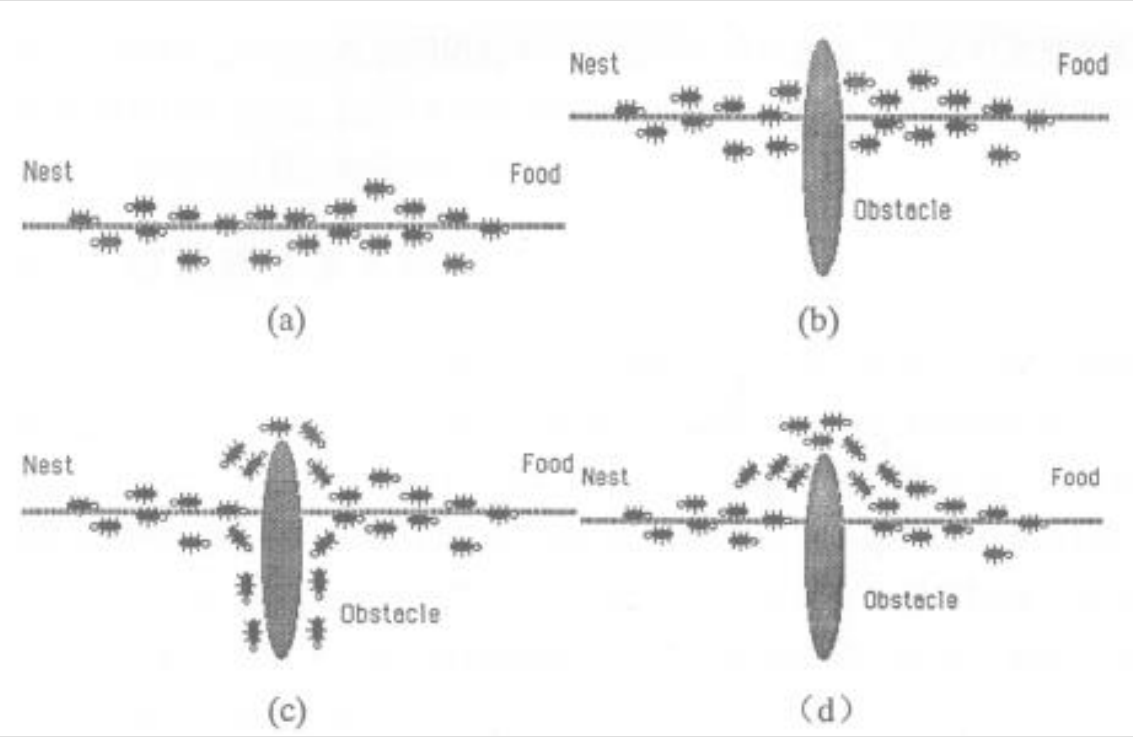
……

第 6 章　模拟生物群体行为的群智能算法

- 6.1　群智能算法的生物学背景
- 6.2　模拟鸟群行为的粒子群优化算法
- 6.3　模拟蚁群行为的蚁群优化算法

6.3.1 蚁群算法的生物学背景

- **20世纪90年代初，意大利科学家Marco Dorigo等受蚂蚁觅食行为的启发，提出蚁群算法(Ant Colony Optimization，ACO)。**

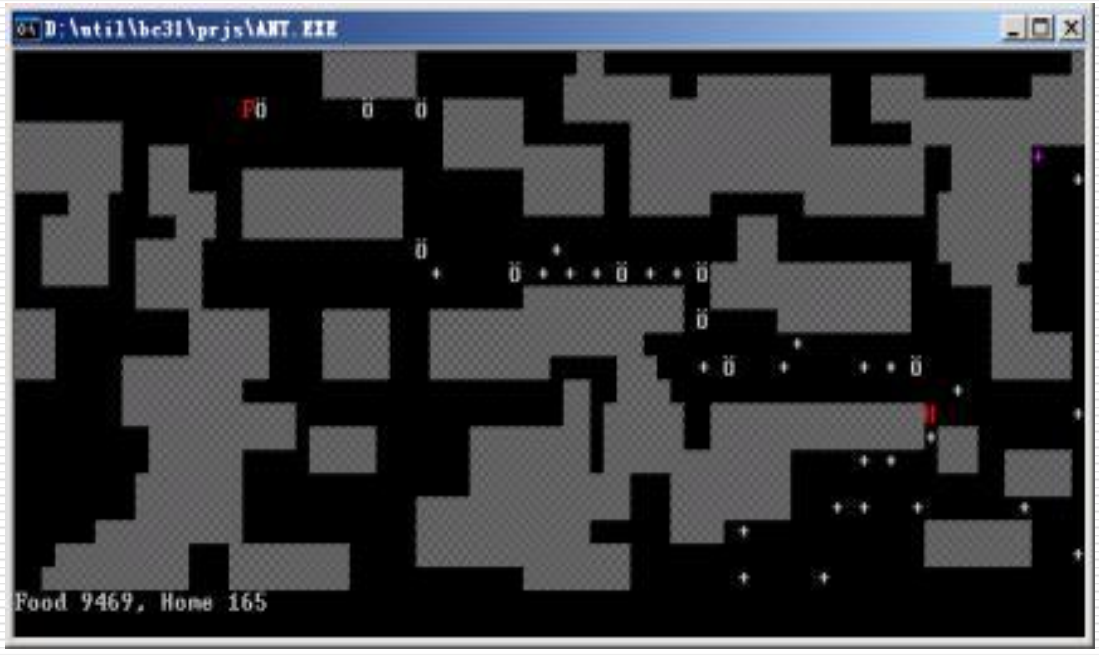


6.3.1 蚁群算法的生物学背景

基本思想

●**信息素跟踪**：按照一定的**概率**沿着信息素较强的路径觅食。

●**信息素遗留**：会在走过的路上会释放信息素，使得在一定的范围内的其他蚂蚁能够觉察到并由此影响它们的行为。



6.3.1 蚁群算法的生物学背景

（1）环境：有障碍物、有其他蚂蚁、有信息素。

（2）觅食规则：范围内寻找是否有食物，否则看是否有信息素，每只蚂蚁都会以小概率犯错。

（3）移动规则：都朝信息素最多的方向移动，无信息素则继续朝原方向移动，且有随机的小的扰动，有记忆性。

（4）避障规则：移动的方向如有障碍物挡住，蚂蚁会随机选择另一个方向。

（5）信息素规则：越靠近食物播撒的信息素越多，越离开食物播撒的信息素越少。

6.3.2 基本蚁群优化算法

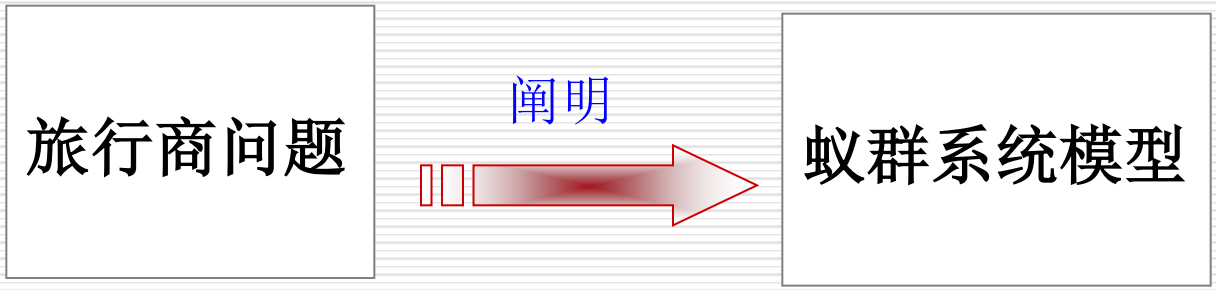
蚁群优化算法的第一个应用是著名的旅行商问题。

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

在寻求单一旅行者由起点出发，通过所有给定的需求点之后，最后再回到原点的最小路径成本。

蚂蚁搜索食物的过程：

通过个体之间的信息交流与相互协作最终找到从蚁穴到食物源的最短路径。



6.3.2 基本蚁群优化算法

蚁群系统的模型

m 是蚁群中蚂蚁的数量

$d_{xy}(x, y = 1, ..., n)$ 表示元素(城市) 和元素(城市) 之间的距离

$\eta_{xy}(t)$ 表示能见度，称为启发信息函数，等于距离的倒数，即 $\eta_{xy}(t) = \dfrac{1}{d_{xy}}$

$b_x(t)$ 表示t时刻位于城市x的蚂蚁的个数，$m = \sum_{x=1}^{n} b_x(t)$

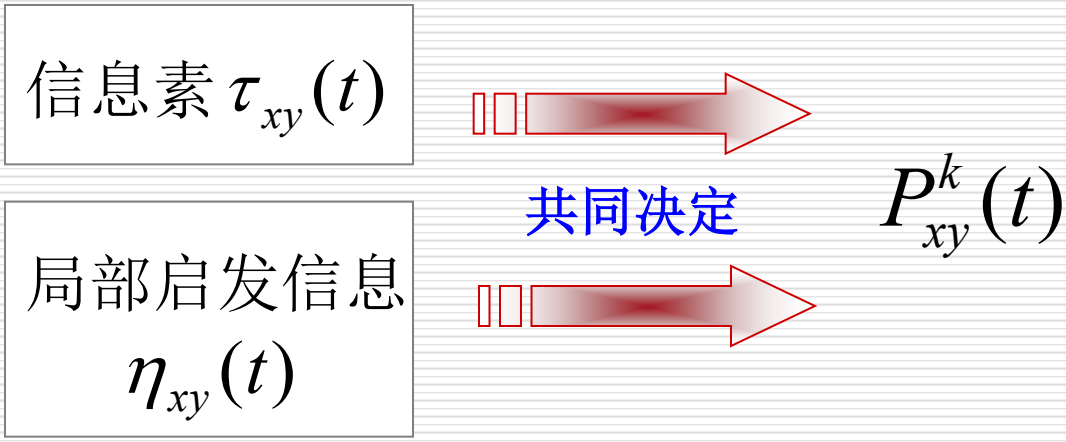
$\tau_{xy}(t)$ 表示t时刻在xy连线上残留的信息素，初始时刻，各条路径上的信息素相等即 $\tau_{xy}(0) = C(const)$

蚂蚁k在运动过程中，根据各条路径上的信息素决定转移方向。

6.3.2 基本蚁群优化算法

蚁群系统的模型

$P_{xy}^{k}(t)$ 表示在t时刻蚂蚁 k 选择从元素(城市) x 转移到元素(城市) y 的概率，也称为随机比例规则。



6.3.2 基本蚁群优化算法

$P_{xy}^{k}(t)$ 表示如下：

$$P_{xy}^{k}(t)=\begin{cases}\dfrac{\left|\tau_{xy}(t)\right|^{\alpha}\left|\eta_{xy}(t)\right|^{\beta}}{\sum_{y\in allowed_k(x)}\left|\tau_{xy}(t)\right|^{\alpha}\left|\eta_{xy}(t)\right|^{\beta}} & if \quad y\in allowed_k(x)\\ 0 & \text{其他}\end{cases} \tag{6.2}$$

其中：

$allowed_k(x)=\{0,1,...,n-1\}-tabu_k(x)$ 表示蚂蚁k下一步允许选择的城市

$tabu_k(x)(k=1,2,...,m)$ 记录蚂蚁k当前所走过的城市

α 是信息素启发式因子，表示轨迹的相对重要性

6.3.2 基本蚁群优化算法

$P_{xy}^{k}(t)$ 表示如下：

$$P_{xy}^{k}(t)=\begin{cases}\dfrac{\left|\tau_{xy}(t)\right|^{\alpha}\left|\eta_{xy}(t)\right|^{\beta}}{\sum_{y\in allowed_k(x)}\left|\tau_{xy}(t)\right|^{\alpha}\left|\eta_{xy}(t)\right|^{\beta}} & if \quad y\in allowed_k(x)\\ 0 & 其他\end{cases} \tag{6.2}$$

其中：

α 值越大	该蚂蚁越倾向于选择其它蚂蚁经过的路径，该状态转移概率越接近于贪婪规则。
当 $\alpha=0$时	不再考虑信息素水平，算法就成为有多重起点的随机贪婪算法。
当 $\beta=0$时	算法就成为纯粹的正反馈的启发式算法。

6.3.2 基本蚁群优化算法

用参数1-ρ表示信息素消逝程度，蚂蚁完成一次循环，各路径上信息素浓度消散规则为：

$$\tau_{xy}(t) = \rho\tau_{xy}(t) + \Delta\tau_{xy}(t) \qquad (6.3)$$

蚁群的信息素浓度更新规则为：

$$\Delta\tau_{xy}(t) = \sum_{k=1}^{m} \Delta\tau_{xy}^{k}(t) \qquad (6.4)$$

M. Dorigo给出$\Delta\tau_{xy}^{k}(t)$的三种不同模型

6.3.2 基本蚁群优化算法

1. 蚂蚁圈系统（Ant-cycle System）

单只蚂蚁所访问路径上的信息素浓度更新规则为：

$$\Delta\tau_{xy}^{k}(t)=\begin{cases}\dfrac{Q}{L_k} & \text{若第}k\text{只蚂蚁在本次循环中从}x\text{到}y\\ 0 & \text{否则}\end{cases} \qquad (6.5)$$

其中：$\tau_{xy}(t)$ 为当前路径上的信息素

$\Delta\tau_{xy}(t)$ 为路径(x,y)上信息素的增量

$\Delta\tau_{xy}^{k}(t)$ 第k只蚂蚁留在路径(x,y)上的信息素的增量

Q 为常数

L_k 为优化问题的目标函数值，表示第k只蚂蚁在本次循环中所走路径的长度

6.3.2 基本蚁群优化算法

2. 蚂蚁数量系统（Ant-quantity System）

$$\Delta\tau_{xy}^{k}(t)=\begin{cases}\dfrac{Q}{d_k} & \text{若第}k\text{只蚂蚁在本次循环中从}x\text{到}y\\ 0 & \text{否则}\end{cases} \tag{6.6}$$

3. 蚂蚁密度系统（Ant-density System）

$$\Delta\tau_{xy}^{k}(t)=\begin{cases}Q & \text{若第}k\text{只蚂蚁在本次循环中从}x\text{到}y\\ 0 & \text{否则}\end{cases} \tag{6.7}$$

6.3.2 基本蚁群优化算法

三种模型比较

效果最好，通常作为蚁群优化算法的基本模型。

蚂蚁圈系统	利用的是全局信息 Q/L_k，即蚂蚁完成一个循环后，更新所有路径上的信息。
蚂蚁数量系统	利用的是局部信息 Q/d_{xy}，即蚂蚁每走一步都要更新残留信息素的浓度。
蚂蚁密度系统	利用的是局部信息 Q，即蚂蚁每走一步都要更新残留信息素的浓度。

6.3.3 蚁群优化算法的应用

例6.1 柔性作业车间调度问题：某加工系统有6台机床，要加工4个工件，每个工件有3道工序，如表6.1所示。

6.3.3 蚁群优化算法的应用

表6.1 柔性作业车间调度事例

工序选择		加工机床及加工时间					
		1	2	3	4	5	6
J_1	p_{11}	2	3	4			
	p_{12}		3		2	4	
	p_{13}	1	4	5			
J_2	p_{21}	3		5		2	
	p_{22}	4	3		6		
	p_{23}			4		7	11
J_3	p_{31}	5	6				
	p_{32}		4		3	5	
	p_{33}			13		9	12
J_4	p_{41}	9		7	9		
	p_{42}		6		4		5
	p_{43}	1		3			3

6.3.3 蚁群优化算法的应用

由图6.5可以看出机器6并没有加工任何工件。分析其原因为它虽然可以加工工序p_{23}，p_{33}，p_{42}，p_{43}但从表6.5可知机器6的加工时间大于其他可加工机器，特别是p_{23}，p_{33}的加工时间，因此机器6并未分到任何加工任务。

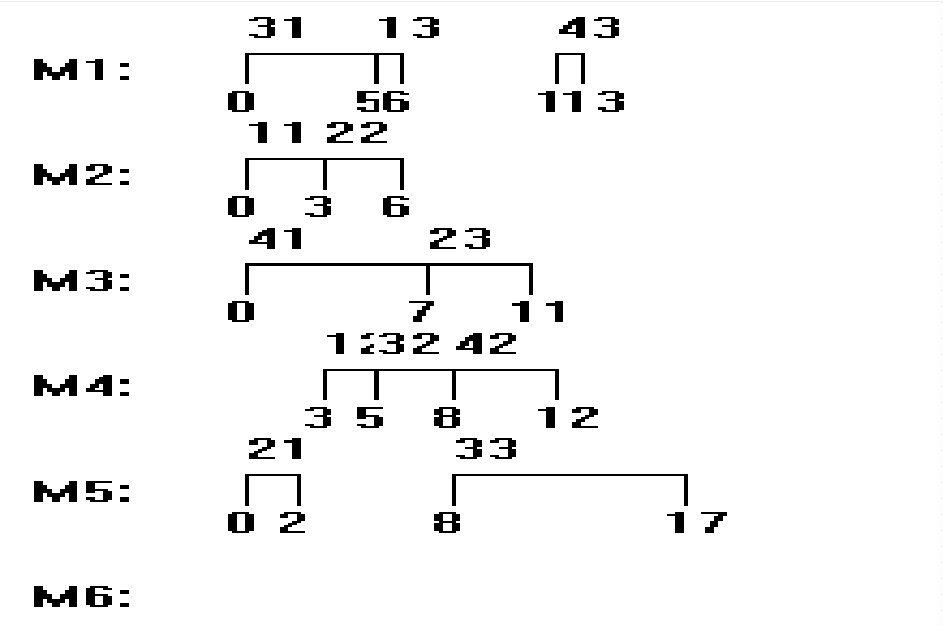
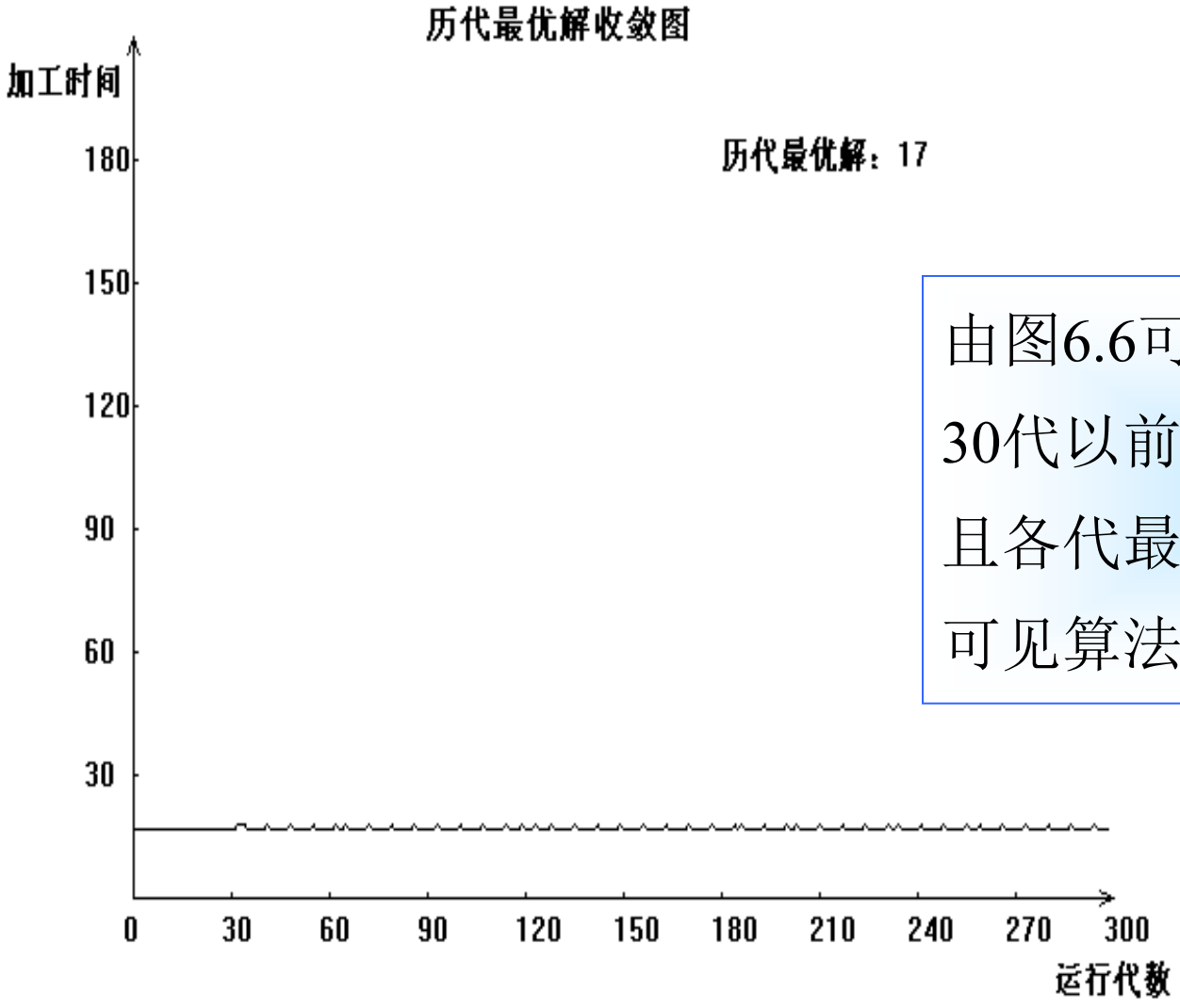


图6.5 最优解甘特图

6.3.3 蚁群优化算法的应用



由图6.6可知，算法在大约30代以前就收敛到最优解，且各代最优解相差不大，可见算法较为稳定。

图6.6 历代最优解收敛图

THE END