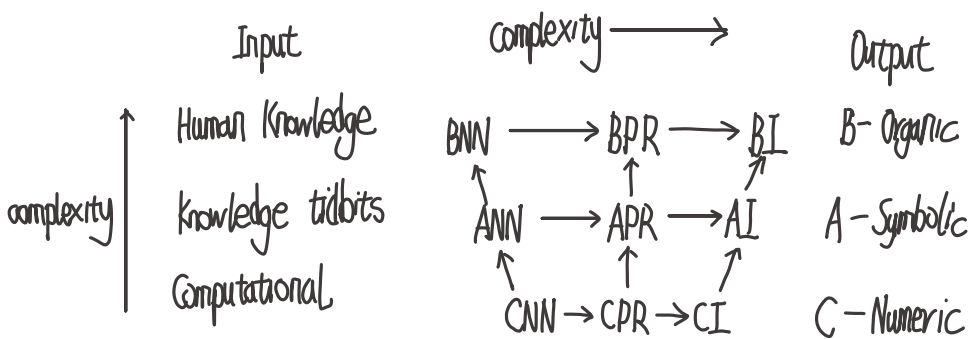


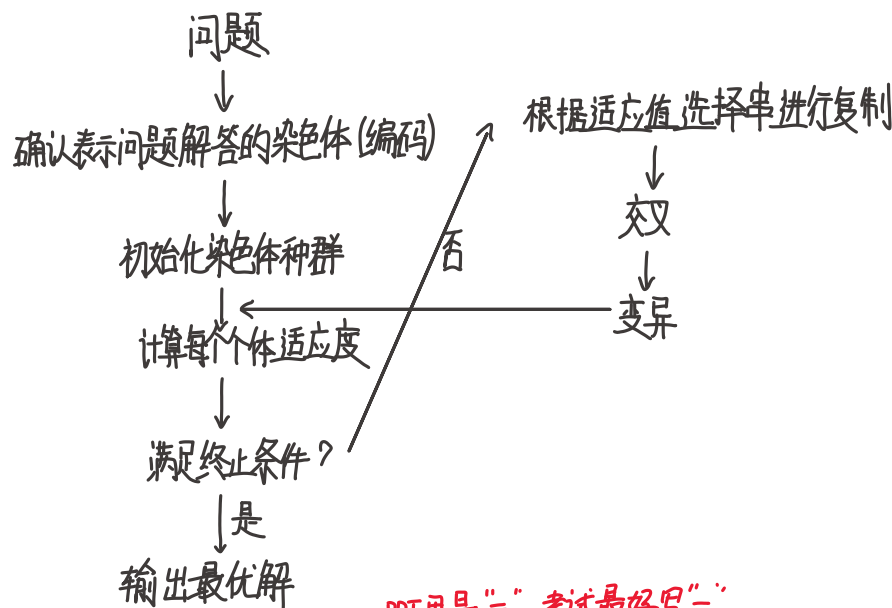
ABC框架和NN、PR、I之间关系



本图由贝兹德克于1994年提出，中间具有9个节点，表示9个研究领域
 ABC三者对应三个系统不同的复杂度级别，复杂度自左向右自下至上
 不断提高，节点间的距离衡量领域间的差距，CNN和CPR的差距远小
 于BNN和BPR之间的差距，CI和AI的差异远小于AI和BI的差距，“→”意味着适当的
 子集，如 $ANN \subset APR \subset AI$ ， $CI \subset AI \subset BI$

以机器人应用为案例：见Notion总结

遗传算法流程



编码计算核心公式 $\delta \geq \frac{U_{max} - U_{min}}{2^L - 1}$

(PPT里是 "=", 考试最好写 "=")

$l = 18 \rightarrow$

$$x = U_{min} + \sum_{i=1}^l b_i 2^{i-1} \frac{U_{max} - U_{min}}{2^L - 1}$$

$$= -30 +$$

$$\begin{matrix} 65536 & 4096 & 256 & 16 & & \\ \uparrow & \uparrow & \uparrow & \uparrow & & \\ 010001001011010000 \end{matrix}$$

$$\frac{70352}{262144} \times 151^{-3}$$

$$65536 \rightarrow 16$$

$$131072$$

$$262144$$

- 遗传算法特点:
1. 对求解问题没有太多数学要求
 2. 能有效地进行概率意义的全局搜索
 3. 算法灵活性高, 对于特殊问题也有效

适应度函数尺度变换
(1) 函数值必须非负

- 选择算子作用: 从当前群体选出优良个体
交叉算子作用: 模拟物种繁衍, 促进个体多样性
变异算子作用: 产生有"个性"的新个体

- 遗传算法步骤
1. 令进化代数 $g=0$, 定义初始化种群 $P(g)$
 2. 对 $P(g)$ 中每一个体计算适应度和选择得到新种群
 3. 按一定方法依适应度选择两个个体进入交叉变异操作, 重复多次得到新一代群体 $P(g+1)$
 4. 满足终止步条件算法结束, 否则回到步骤 2

- 遗传算法5基本要素
1. 个体的参数编码形式
 2. 初始群体的设定
 3. 适应度函数设计
 4. 控制参数的设定 (群体规模 ·)
 5. 遗传操作设定

粒子群算法

基本思想、原理：一群粒子 速度

迭代更新寻找最优 个体极值 全局极值

公式：

i 表示粒子编号， k 表示迭代次数， g 表示 global

连续

$x^i(k)$ $p^i(k)$ $v^i(k)$ $p^g(k)$ ← 向量
↓ 当前位置 个体最优位置 个体方向 群体最优位置

均匀分布随机数

$$v_j^i(k+1) = w(k) v_j^i(k) + \varphi_1 \text{rand}(0, 1) (p_j^i(k) - x_j^i(k)) + \varphi_2 \text{rand}(0, 1) (p_j^g(k) - x_j^i(k))$$

惯性因子

加速度常数

跟踪个体最优 pbest

跟踪全局最优 gbest

$$x_j^i(k+1) = x_j^i(k) + v_j^i(k+1)$$

当前速度惯性

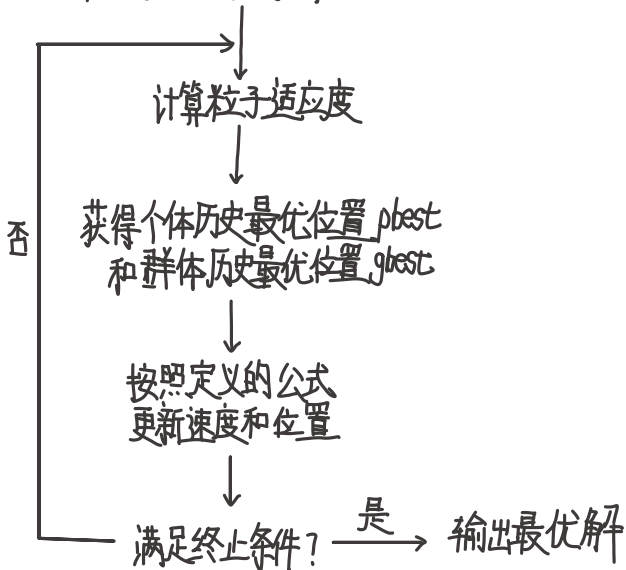
跟踪个体最优 pbest

跟踪全局最优 gbest

向量中的分量值

流程图

初始化粒子群，设置算法参数



速度更新模型 各部分作用

1. 只有第一部分
2. 没有第一部分
3. 没有第一部分
4. 没有第三部分

基本量子粒子群优化算法

1. 粒子不用位置和速度而是波函数来表示

$$\text{波函数 } \psi(x) = \frac{1}{\sqrt{L}} \exp\left(-\frac{\|p-x\|}{L}\right)$$

$$\text{概率密度函数 } Q(x) = \psi^2(x)$$

$\leftarrow p$ 为粒子个体
最好位置

2. 引入 $mbest$ 作为所有微粒的中心

$$mbest = \sum_{i=1}^M \frac{p_i}{M}$$

M 为种群数, p_i 为粒子个体最好位置

基本步骤:

1. 确定种群规模和粒子维数, 初始化粒子群体
2. 计算粒子适应度
3. 计算个体历史最优值 $pbest$ 和全局最优值 $gbest$ (比较、替换)
4. 计算所有粒子重心 $mbest$
5. 更新所有粒子位置, 产生新种群
6. 判断算法是否满足终止条件

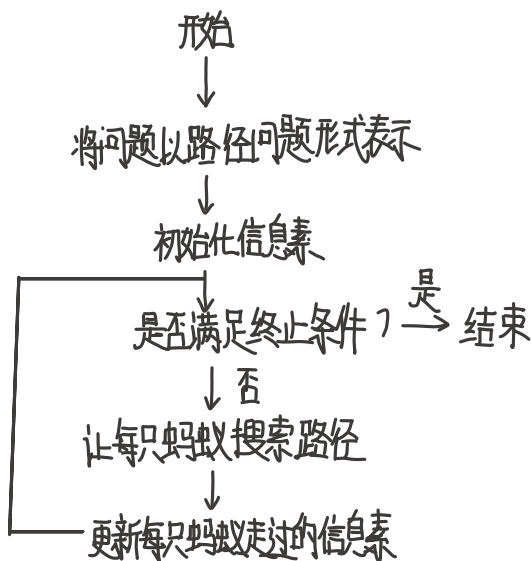
优点: 具有更好收敛性和全局搜索能力

缺点: 求解约束问题会产生大量不可行解, 会破坏种群多样性

蚁群算法：离散

信息素思想：蚂蚁行走留下信息素，其它蚂蚁按一定概率沿信息素较强的路径觅食

流程图



基本蚁群公式

$\tau_{xy}(t)$ → t 时刻留在 xy 连线上信息素 $\tau_{xy}(0) = C$

$\eta_{xy}(t)$ → 启发信息函数，TSP中为距离的倒数 $\frac{1}{d_{xy}}$

蚂蚁在运动过程中，根据路径上信息素决定转移方向

- 1 路径信息素消散规则： $\tau_{xy}(t) = \rho \tau_{xy}(t) + \Delta \tau_{xy}(t)$
蚁群信息素浓度更新规则： $\Delta \tau_{xy}(t) = \sum_{k=1}^m \Delta \tau_{xy}^k(t)$
- 2 t 时刻蚂蚁 k 从城市 x 转到城市 y 概率 $p_{xy}^k(t) = \frac{|\tau_{xy}(t)|^\alpha |\eta_{xy}(t)|^\beta}{\sum |\tau_{xy}(t)|^\alpha |\eta_{xy}(t)|^\beta}$

α 是信息素启发式因子, 反映了随机性因素作用, α 过大陷入局部最优过早

β 是期望值启发式因子, 反映了先验性、确定性因素作用, β 过大陷入局部最优, 加快收敛性

3 计算 $\Delta\tau_{ij}(t)$ 的三种方法

(1) 蚂蚁圈系统: $\Delta\tau_{ij}^k(t) = \begin{cases} \frac{Q}{L_k} & \leftarrow \text{带数} \text{ 第 } k \text{ 只蚂蚁在本次循环中从 } i \text{ 到 } j \\ 0 & \leftarrow \text{该蚂蚁本循环所走路径长度} \end{cases}$
 效果最好
 ↓ 一个循环更新一次

(2) 蚂蚁数量系统: $\Delta\tau_{ij}^k(t) = \begin{cases} \frac{Q}{d_k} & \leftarrow \text{本段路径长度} \\ 0 & \end{cases}$ } 每走一步更新一次

(3) 蚂蚁密度系统: $\Delta\tau_{ij}^k(t) = \begin{cases} Q \\ 0 \end{cases}$

1- ρ 信息素挥发度 过大降低算法随机性和全局搜索能力
 过小降低收敛速度

蚁群算法 A 和遗传算法 G 的区别联系

联系 1 都是不确定性算法, 求解过程有随机性要素

2 都使用了“种群”概念, 多个体搜索, 用统计规律战胜个体局限性

3 使用了“适应度”对个体进行评价

4 去中心化, 算法鲁棒性好

区别 1 GA 处理完整的解, ACO 是一步步构建解

更新环境

2 GA 中信息共享是隐式的 (优势个体存活), ACO 中信息共享是显式的 (信息素)

进化: 种群中遗传性状在世代之间的变化