

注：

XXXX： 表示老师上课提示的内容，必看

XXXX： 表示可能出现的名词解释

其余颜色越深，颜色越多 个人认为越重要

编辑：1603011_zr

一 绪论

1.1 人工智能的主要学派人工智能的主要学派：

答：人工智能经过发展，形成了许多学派。不同学派的研究方法、学术观点、研究重点有所不同。主要形成了以下学派：符号主义学派，连接主义学派，行为主义学派。

- 符号主义学派：认为人工智能来源于数理逻辑；基于物理符号系统假设和有限合理性原理，以知识的符号表达为基础，利用推理进行问题求解，发展了启发式算法→专家系统→知识工程理论与技术；(研究方法：功能模拟法)
- 连接主义学派：认为人工智能来源于仿生学；基于神经网络及其间的连接机制和学习算法，以人工神经网络为代表，侧重于模拟和实现人的认识过程中的感知过程、形象思维、分布式记忆、自学习自组织的过程。(研究方法：结构模拟法)
- 行为主义学派：智能只是在与环境的交互作用中产生。基于控制论和感知-动作型控制系统，主张从行为中模拟、扩展、延伸出智能；Brooks 提出了无需知识表示的智能，无需推理的智能。他认为智能只是在与环境的交互作用中表现出来，其基本观点：(1) 到现场去；(2) 物理实现；(3) 初级智能；(4) 行为产生智能。(研究方法：行为模拟法)

1.2 传统 AI 的三大核心研究内容

- **知识表示**：其是基础。其是把人类知识概念化、形式化或模型化。即运用符号知识、算法和状态图等来描述待解决的问题。主要包括符号表示法和神经网络表示法，涉及到状态空间法、问题规约法、谓词演算法、语义网络法、框架表示法、本体表示法、过程表示法和神经网络表示法。
- **知识推理**：其实现问题求解。推理即是从一些已知判断或前提推导出一个新的判断或结论的思维过程。形式逻辑中的推理分为演绎推理、归纳推理和类比推理。知识推理，包括不确定性推理和非经典推理。
- **知识应用**：其是目的。人工智能能否获得广泛应用是衡量其生命力和检验其生存力的重要标志。专家系统，机器学习，自然语言理解的发展促进了 AI 的进步。

1.3 人工智能的应用

问题求解与博弈，逻辑推理与定理证明，计算智能，分布式人工智能 Agent, 自动程序设计，专家系统，机器学习，自然语言处理，机器人学，模式识别，机器视觉，神经网络，智能控制，智能调度与指挥，智能检索，系统和语言工具。

对于**人工智能研究的展望**（自己发挥，送分题？）**人工智能学术派别之争趋于平和，走向集成和合作。**

二 知识表示

2.1 知识、信息、数据及其关系

知识是人们在改造客观世界的实践中积累起来的认识和经验。可以是对信息的关联，也可以是对已有知识的再认识，有价值的信息沉淀并结构化后就形成了知识。

数据：是信息的载体，事实或观察的结果，是对客观事物的逻辑归纳，用于表示客观事物未经加工的原始素材。本身无确切含义。如：水的温度是 100°C ，

信息：是数据的关联，赋予数据特定的含义，其比数据更抽象，是隐藏在数据背后的规律，需要分析和挖掘方能得到，仅可理解为描述性知识。数据是没有联系的，孤立的，只有**当数据用来描述一个客观事物和客观事物的关系，形成有逻辑的数据流，他们才能被称为信息。**

2.2 知识表示：

知识表示：是研究用机器表示知识的可行性、有效性的一般方法，是一种数据结构与控制结构的统一体，既考虑知识的存储又考虑知识的使用。

是对知识的一种描述，一种计算机可以接受的用于描述知识的数据结构。

状态空间法是一种基于解答空间的问题表示和求解方法，它是以状态和操作符为基础的。在利用状态空间图表示时，我们从某个初始状态开始，每次加一个操作符，递增地建立起操作符的试验序列，直到达到目标状态为止。由于状态空间法需要扩展过多的节点，容易出现“组合爆炸”，因而只适用于表示比较简单的问题。

问题归约法从目标(要解决的问题)出发，逆向推理，通过一系列变换把初始问题变换为子问题集合和子-子问题集合，直至最后归约为一个平凡的本原问题集合。这些本原问题的解可以直接得到，从而解决了初始问题，用与或图来有效地说明问题归约法的求解途径。

谓词逻辑法采用谓词合式公式和一阶谓词演算把要解决的问题变为一个有待证明的问题，然后采用消解定理和消解反演来证明一个新语句是从已知的正确语句导出的，从而证明这个新语句也是正确的。

语义网络是知识的一种图解表示，它由节点和弧线或链线组成。节点用于表示实体、概念和情况等，弧线用于表示节点间的关系。语义网络通过**继承**和**匹配**进行推理。

框架是一种结构化表示方法。框架通常由指定事物各个方面的槽组成，每个槽拥有若干个侧面，而每个侧面又可拥有若干个值。大多数实用系统必须同时使用许多框架，并可把它们联成一个框架系统。

剧本是框架的一种特殊形式，它使用一组槽来描述事件的发生序列。剧本表示特别适用于描述顺序性动作或事件，但使用不如框架灵活，因此应用范围也不如框架那么广泛。

过程是一种知识的过程式表示，它将某一有关问题领域知识同这些使用方法一起，隐式地表示为一个问题求解过程。过程表示用程序来描述问题，具有很高的问题求解效率。由于知识隐含在程序中难以操作，所以适用范围较窄。

状态空间法的三要素

(1) 状态 (state)：描述某类不同事物间的差别而引入的一组最少变量 q_0, q_1, \dots, q_n 的有序集合。有序集合中每个元素 q_i ($i = 0, 1, \dots, n$) 为集合的分量，称为状态变量。给定每个分量的一组值就得到一个具体的状态。

(2) 算符 (operator): 使问题从一种状态变化为另一种状态的手段称为操作符或算符。

(3) 状态空间方法: 是一个表示该问题全部可能状态及其关系的图, 它包含三种说明的集合, 即三元状态 (S, F, G)。

搜索:

根据问题的实际情况, 不断寻找可利用的知识, 构造出一条代价较少的推理路线, 使得问题能够圆满解决的过程。

搜索分类:

❖ 搜索的类型

➤ **盲目搜索:** 按预定的控制策略进行搜索, 在搜索过程中获得的中间信息并不改变控制策略。

- 宽度优先搜索
- 深度优先搜索
- 等代价搜索

➤ **启发式搜索:** 在搜索中加入了与问题有关的启发性信息, 用于指导搜索朝着最有希望的方向前进, 加速问题的求解过程并找到最优解。

- 有序搜索
- A*算法

启发式搜索:

采用问题自身的特性信息, 以指导搜索朝着最有希望的方向前进, 加速问题的求解并且找到最优解。

启发性信息的概念: 启发性信息是指那种与具体问题求解过程有关的, 并可指导搜索过程朝着最有希望方向前进的控制信息。启发信息的启发能力越强, 扩展的无用结点越少。

搜索的完备性:

对于一类可解的问题和一个搜索过程, 如果运用该搜索过程一定能求得该类问题的解, 则称该搜索过程为完备的, 否则为不完备的。

注: 广度优先搜索、等代价搜索、有界深度优先搜索以及 A*算法都是完备的搜索过程, 其它搜索过程都是不完备的。

示例学习:

示例学习 (Learning from Examples) 则是这一种归纳学习方法。

(归纳学习: 研究一般性 概念的描述和 概念聚类??)

估价函数和启发函数:

用于评估节点重要性的函数称为估价

函数。估价函数的一般形式为：

$$f(x) = g(x) + h(x)$$

✓ $g(x)$ 表示从初始节点 S_0 到节点 x 的代价；

✓ $h(x)$ 是从节点 x 到目标节点 S_g 的最优路径的代价的估计，它体现了问题的启发性信息。

✓ $h(x)$ 称为启发函数。

文字和子句：

子句定义为由文字的析取组成的公式(一个原子公式和原子公式的否定都叫做文字)。当消解可使用时,消解过程被应用于母体子句对,以产生一个导出子句。

盲目搜索和启发式搜索的区别：

① 不需要重新安排 OPEN 表的搜索叫做无信息搜索或盲目搜索，盲目搜索按规定的控制策略进行搜索，在搜索过程中获得的中间信息并不改变控制策略。它的代表是宽度优先搜索、深度优先搜索和等代价搜索。盲目搜索只适用于求解比较简单的问题。

② 启发式搜索在搜索过程中加入了与问题有关的启发式信息,用于指导搜索朝着最有希望的方向前进，加速问题的求解过程并找到最优解。它的代表是有序搜索和 A* 算法。

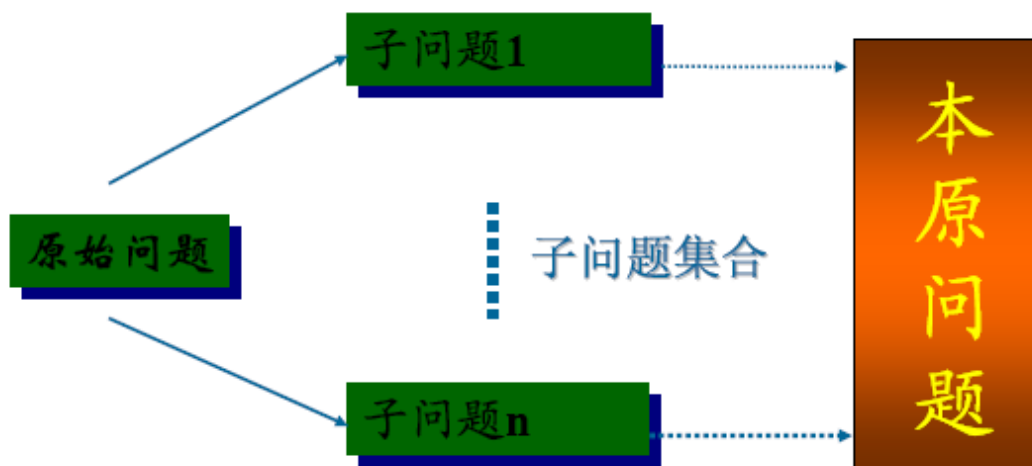
问题归约 (Problem Reduction)

✓ 是另外一种基于状态空间的问题描述与求解方法

✓ 已知问题的描述，通过一系列变换把此问题变为一个子问题集合

✓ 这些子问题的解可以直接得到 (本原问题)，从而解决了初始问题

本原问题：



置换与合一：

- **置换的定义：**置换是用变元、常量、函数来替换变元，使该变元不在公式中出现。置换是形如 $\{t_1/x_1, t_2/x_2, \dots, t_n/x_n\}$ 的有限集合， t_i/x_i 表示用 t_i 项替换变元 x_i ，不允许 t_i 和 x_i 相同，也不允许变元 x_i 循环地出现在另一个 t_j 中

- 合一的定义：寻找项对变量的置换，以使两表达式一致。

公式集的合一：设有公式集 $F=\{F_1, F_2, \dots, F_n\}$ ，若存在一个代换 λ 使得 $F_1\lambda=F_2\lambda=\dots=F_n\lambda$ ，则称 λ 为公式集 F 的一个合一，且称 F_1, F_2, \dots, F_n 是可合一的。

最一般的合一：设 σ 是公式集 F 的一个合一，如果对任一个合一 θ 都存在一个代换 λ ，使得 $\theta=\sigma\lambda$ 则称 σ 是一个最一般的合一。

- (1) 代换过程是一个用项代替变元的过程，因此是一个从一般到特殊的过程。
- (2) 最一般合一是唯一的。

归结：

设 C_1 和 C_2 是子句集中的任意两个子句，如果 C_1 中的文字 L_1 与 C_2 中的文字 L_2 互补，那么可从 C_1 和 C_2 中分别消去 L_1 和 L_2 ，并将 C_1 和 C_2 中余下的部分按析取关系构成一个新的子句 C_{12} ，则称这一过程为**归结**，称 C_{12} 为 C_1 和 C_2 的归结式，称 C_1 和 C_2 为 C_{12} 的亲本子句。

命题逻辑的归结：

为证明子句集 S 的不可满足性，只要对其中可进行归结得子句进行归结，并把归结式加入到子句集 S 中，或者用归结式代替他的亲本子句，然后对新的子句集证明其不可满足性就可以了。

子句集 S 是不可满足的，当且仅当存在一个从 S 到空子句的归结过程。

归结演绎推理 归结反演

如欲证明 Q 为 P_1, P_2, \dots, P_n 的逻辑结论，只需证 $(P_1 \wedge P_2 \wedge \dots \wedge P_n) \wedge \neg Q$ 是不可满足的，或证明其子句集是不可满足的。而子句集的不可满足性可用归结原理来证明。

应用归结原理证明定理的过程称为**归结反演**。

设 F 为已知前提的公式集， Q 为目标公式(结论)，用归结反演证明 Q 为真的步骤是：

1. 否定 Q ，得到 $\neg Q$ ；
2. 把 $\neg Q$ 并入到公式集 F 中，得到 $\{F, \neg Q\}$ ；
3. 把公式集 $\{F, \neg Q\}$ 化为子句集 S ；
4. 应用归结原理对子句集 S 中的子句进行归结，并把每次归结得到的归结式都并入 S 中。如此反复进行，若出现了空子句，则停止归结，此时就证明了 Q 为真。

归结策略可分为两大类：

一类是**删除策略**；删除某些无用的子句来缩小归结的范围。

纯文字删除法，重言式删除法，包孕删除法

一类是**限制策略**。通过对参加归结的子句进行种种限制，尽可能减小归结的盲目性，使其尽快地归结出空子句。

支持集策略，线性输入策略，单文字子句策略，祖先过滤策略

归结演绎推理的优点：

✓ 简单，便于在计算机上实现。

❖ 归结演绎推理的不足：

✓ 必须把逻辑公式化成子句集。

- ✓ 不便于阅读与理解：
- ✓ 可能丢失控制信息

不确定性推理:

- 不确定性推理是建立在非经典逻辑基础上的一种推理,它是对不确定性知识的运用与处理。
- 具体地说,不确定性推理就是从不确定性的初始证据(即事实)出发,通过运用不确定性知识,最终推出具有一定程度不确定性的结论。

七、归纳推理和演绎推理的区别与联系

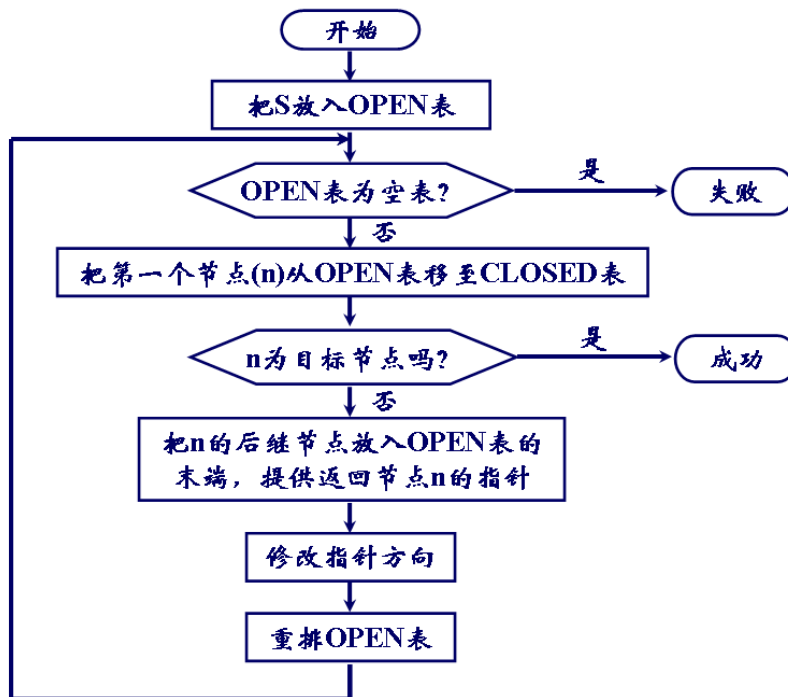
(1) 区别

- ①思维进程不同。归纳推理的思维进程是从个别到一般,而演绎推理的思维进程不是从个别到一般,是一个必然地得出的思维进程。
- ②对前提真实性的要求不同。演绎推理要求大前提,小前提必须为真。归纳推理则没有这个要求。
- ③结论所断定的知识范围不同。演绎推理的结论没有超出前提所断定的知识范围。归纳推理除了完全归纳推理,结论都超出了前提所断定的知识范围。
- ⑤前提与结论间的联系程度不同。演绎推理的前提与结论间的联系是必然的,也就是说,前提真实,推理形式正确,结论就必然是真的。归纳推理除了完全归纳推理前提与结论间的联系是必然的外,前提和结论间的联系都是或然的,也就是说,前提真实,推理形式也正确,但不能必然推出真实的结论。

(2) 联系

- ①演绎推理如果要以一般性知识为前提,则通常要依赖归纳推理来提供一般性知识。
- ②归纳推理离不开演绎推理。其一,为了提高归纳推理的可靠程度,需要运用已有的理论知识,对归纳推理的个别性前提进行分析,把握其中的因果性,必然性,这就要用到演绎推理。其二,归纳推理依靠演绎推理来验证自己的结论。

图搜索的一般策略：



深度优先搜索与广度优先搜索的唯一区别：广度优先搜索是将节点 n 的子节点放入到 OPEN 表的尾部，而深度优先搜索是把节点 n 的子节点放入到 OPEN 表的首部。

$\alpha - \beta$ 剪枝的条件是什么？

α 剪枝：若任一极小值层节点的 β 值小于或等于它任一先辈极大值节点的 α 值，即 α （先辈层） $\geq \beta$ （后继层），则可中止该极小值层中这个 MIN 节点以下的搜索过程。这个 MIN 节点最终的倒推值就确定为这个 β 值。

β 剪枝：若任一极大值层节点的 α 值大于或等于它任一先辈极小值层节点的 β 值，即 α （后继层） $\geq \beta$ （先辈层），则可以中止该极大值层中这个 MAX 节点以下的搜索过程。这个 MAX 节点的最终倒推值就确定为这个 α 值。

专家系统的一般步骤有哪些？专家系统的开发与一般的软件系统开发相比较, 有哪些共同点和特点？

专家系统与一般的软件系统开发无异，其开发过程同样要遵循软件工程的步骤和原则，即也要进行系统分析、系统设计等几个阶段的工作。

但由于它是专家系统，而不是一般的软件系统，所以，又有其独特的地方，主要包括以下几个步骤：

- 系统总体分析与设计；
- 知识获取；
- 知识表示与知识描述语言设计；
- 知识库设计、知识库管理系统设计；
- 推理机与解释模块设计；
- 总控与界面设计
- 编程与调试

- 测试与评价
- 运行与维护

可以看出它有如下特点：

- 知识获取和知识表示设计是一切工作的起点；
- 知识表示与知识描述语言确定后，其他设计可同时进行；

盲目搜索：按照预定的控制策略进行搜索，在搜索过程中获得的中间信息不用来改进控制策略。

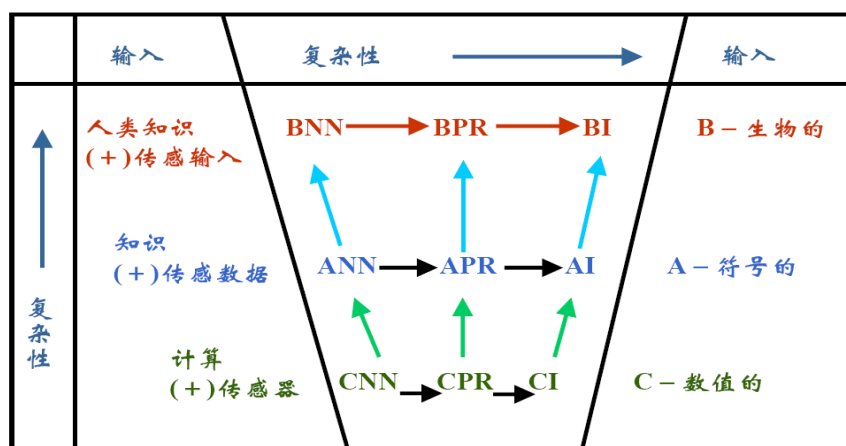
启发式搜索：在搜索中加入了与问题有关的启发性信息，用以指导搜索朝着最有希望的方向前进，加速问题的求解过程并找到最优解。

信号累积：

计算智能：

- 计算智能是学科交叉研究过程中出现的一个重要研究方向。
- 计算智能涉及神经网络、模糊逻辑、进化计算和人工生命等领域，它的研究和发展正是反映了当代科学技术多学科交叉与集成的重要发展趋势。

计算智能和人工智能的区别和联系：



计算智能是一种智力方式的低层认知，它与人工智能的区别只是认知层次从中层下降至低层而已。中层系统含有知识（精品），低层系统则没有。

生物神经特性：

- (1) 并行分布处理的工作模式
- (2) 神经系统的可塑性和自组织性。
- (3) 信息处理与信息存储合二为一。
- (4) 信息处理的系统性
- (5) 能接受和处理模糊的、模拟的、随机的信息。
- (6) 求满意解而不是精确解。
- (7) 系统具有鲁棒性和容错性

人工神经网络的特性

1. 可以充分逼近任意复杂的非线性关系
2. 所有定量或定性的信息都等势分布贮存于网络内的各神经元，故有很强的鲁棒性和容错

性

3. 采用并行分布处理方法，使得快速进行大量运算成为可能
4. 可学习和自适应不知道或不确定的系统
5. 能够同时处理定量、定性知识。
6. 可以通过软件和硬件实现。

人工神经网络：

人工神经网络是反映人脑结构及功能的一种抽象数学模型，是由大量神经元节点互连而成的复杂网络，用以模拟人类进行知识的表示与存储以及利用知识进行推理的行为。

人工神经网络（ANN）可以看成是以人工神经元为结点，用有向加权弧连接起来的有向图。在此有向图中，人工神经元就是对生物神经元的模拟，而有向弧则是轴突—突触—树突对的模拟。有向弧的权值表示相互连接的两个神经元间相互作用的强弱。

学习算法：

学习就是对信息进行编码，其目的就是通过向有限个例子（训练样本）的学习来找到隐藏在例子背后（即产生这些例子）的规律（如函数形式）。

分为三种学习算法：

1. 有师学习（监督学习）

有师（监督）就是对每一个输入 X_i ，都假定我们已经知道它的期望输出 Y_i ，这个 Y_i 可以理解为监督信号

2. 无师学习（非监督学习）

无师学习不需要知道期望输出。

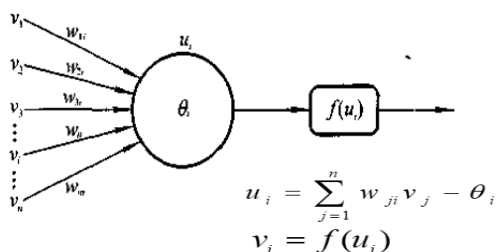
3. 强化学习

指从环境状态到行为映射的学习，以使系统行为从环境中获得的累积奖励值最大。

监督学习与非监督学习的区别：

- 在监督学习中，假定我们知道每一输入对应的期望输出，并利用学习系统的误差，不断校正系统的行为；
- 在非监督学习中，我们不知道学习系统的期望输出。

标准MP模型



w_{ij} ——代表神经元 i 与神经元 j 之间的连接强度(模拟生物神经元之间突触连接强度)，称之为连接权；

u_i ——代表神经元 i 的活跃值，即神经元状态；

v_j ——代表神经元 j 的输出，即是神经元 i 的一个输入；

θ_i ——代表神经元 i 的阈值。

常用激励函数:

线性函数, 阈值型函数, Sigmoid 型函数, 双曲正切函数

进化计算包括:

- 遗传算法(genetic algorithms, GA)
- 进化策略(evolution strategies)
- 进化编程(evolutionary programming)
- 遗传编程(genetic programming)。

遗传算法是一种全局随机优化技术。Darwin 进化论最重要的是适者生存原理。Mendel 遗传学说最重要的是基因遗传原理。

遗传算法 GA:

把问题的解表示成“染色体”, 在算法中即是以一定方式编码的串。并且, 在执行遗传算法之前, 给出一群“染色体”, 也即假设解。然后, 把这些假设解置于问题的“环境”中, 并按适者生存的原则, 从中选择出较适应环境的“染色体”进行复制, 再通过交叉, 变异过程产生更适应环境的新一代“染色体”群。这样, 一代一代地进化, 最后就会收敛到最适应环境的一个“染色体”上, 它就是问题的最优解。

遗传算法由四个部分组成:

编码机制、控制参数、适应度函数、遗传算子

编码机制(encoding mechanism): 用遗传算法解决问题时, 首先要对待解决问题的模型结构和参数进行编码, 一般用字符串表示。

适应性强弱是通过计算适应度函数 $f(x)$ 的值来判别的, 这个值称为适应值。

控制参数, GA 的实际操作时, 需适当确定某些参数的值以提高选优的效果。

遗传算法中的三种算子:

(1) 选择算子(Selection/Reproduction): 从群体中按某一概率选择个体, 某个体 x_i 被选择的概率 P_i 与其适应度值成正比。最通常的实现方法是轮盘赌(roulette wheel)模型。

(2) 交叉算子(Crossover): 将被选中的两个个体的基因链按概率 P_c 进行交叉, 生成两个新的个体, 交叉位置是随机的。其中 P_c 是一个系统参数。

(3) 变异算子(Mutation): 将新个体的基因链的各位按概率 P_m 进行变异, 对二值基因链(0,1 编码)来说即是取反。

简单遗传算法的遗传操作主要有三种:

选择(selection)、交叉(crossover)、变异(mutation)。

- 选择操作也叫复制操作, 根据个体的适应度函数值所度量的优、劣程度决定它在下一代是被淘汰还是被遗传。遗传算法本质上是一种随机搜索, 选择算子则将遗传搜索的方向引向最优解所在区域!
- 交叉: 对于选中用于繁殖下一代的个体, 随机地选择两个个体的相同位置, 按交叉概率 P_c 在选中的位置实行交换。这个过程反映了随机信息交换

- 变异：根据生物遗传中基因变异的原理，以变异概率 P_m 对某些个体的某些位执行变异

遗传算法的主要特点

- 遗传算法一般是直接在解空间搜索，而不像图搜索那样一般是在问题空间搜索，最后才找到解。
- 遗传算法的搜索随机地始于搜索空间的一个点集，而不像图搜索那样固定地始于搜索空间的初始节点或终止节点，所以遗传算法是一种随机搜索算法。
- 遗传算法总是在寻找优解，而不像图搜索那样并非总是要求优解，而一般是设法尽快找到解，所以遗传算法又是一种优化搜索算法。
- 遗传算法的搜索过程是从空间的一个点集(种群)到另一个点集(种群)的搜索，而不像图搜索那样一般是从空间的一个点到另一个点地搜索。因而它实际是一种并行搜索，适合大规模并行计算，而且这种种群到种群的搜索有能力跳出局部最优解。
- 遗传算法的适应性强，除需知适应度函数外，几乎不需要其他的先验知识。

■ 5.3.2 进化编程的步骤

进化编程可繁殖出新的计算机程序以解决问题，它分为三个步骤：

(1) 产生出初始群体，它由关于问题（计算机程序）的函数随机组合而成。

(2) 迭代完成下述子步骤，直至满足选种标准为止：

(a) 执行群体中的每个程序，根据它解决问题的能力，给它指定一个适应值。

(b) 应用变异等操作创造新的计算机程序群体。

(3) 在后代中适应值最高的计算机程序个体被指定为进化编程的结果。进化编程的基本过程如右图。



进化编程的基本过程

■ 进化策略和遗传算法的区别

除了研究和应用领域外，进化策略和遗传算法还有以下区别：

- (1) 进化策略和遗传算法表示个体的方式不同，进化策略在浮点矢量上运行，而遗传算法一般运行在二进制矢量上。
- (2) 进化策略和遗传算法的选择过程不同。
- (3) 进化策略和遗传算法的复制参数不同，遗传算法的复制参数(交叉和变异的可能性)在进化过程中保持恒定，而进化策略时时改变它们。

