

COMP9414：人工智能讲座3b。规

划

韦恩-沃布克

电邮：w. wobcke@unsw.edu.au

本讲座

- 关于行动的推理
- STRIPS规划师
- 图形计划
- 作为约束条件满足的规划

规划代理人

- 环境因行动的实施而改变
- 规划方案
 - ▲ 代理人可以控制其环境
 - ▲ 只有原子行动，而不是有期限的过程
 - ▲ 环境中只有单一制剂（无干扰）。
 - ▲ 只有由于代理人执行行动而产生的变化（没有进化）。
- 更复杂的例子
 - ▲ 机器人杯狗
 - ▲ 送货机器人
 - ▲ 自动驾驶汽车

关于行动的推理

- 语义学。将世界划分为一连串的（名义上的）时间点
 - ▲ 情况是世界在某一时间点的（完整）状态
 - ▲ 行动是一种情况之间的过渡
 - ▲ 在两种情况之间没有发生（相关的）事情
- 规划师。保持对情况的**不完整**描述
 - ▲ 令人困惑的是，也被称为世界的一种**状态**



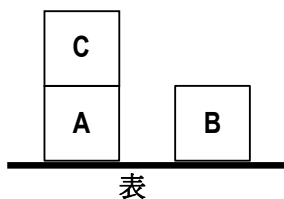
寻找从初始状态到目标状态的路径

▲ 状态转换对应于行动

主要问题是指定行动

块状世界

- 积木可以放在桌子上，可以互相堆叠
- 所有区块的尺寸相同，桌子大到足以容纳所有区块



状态： $on(C, A)$, $on(A, Table)$, $on(B, Table)$, $clear(B)$, $clear(C)$

块状世界行动 (STRIPS)

- 动作描述： $move(x, y, z)$ ($x \neq y \neq z$)
- 前提条件： $on(x, y)$, $clear(x)$, $clear(z)$
- 删除列表： $clear(z)$, $on(x, y)$
- 添加列表： $on(x, z)$, $clear(y)$, $clear(table)$
 - ▲ 添加 $clear(Table)$ ，以确保表始终是清空的。

指定行动 (STRIPS)

- 行动描述--行动的名称
- 前提条件--
只有在行动执行前的情况下，前提条件成立，行动才能在
情况下执行。
- 删除列表 - 在执行行动后要从状态（描述）中删除的字词
- 添加列表 - 在执行行动后要添加到状态（描述）中的字词
- STRIPS 假设--
在执行动作后，状态（描述）中不包含在删除列表中的任
何字词都保持不变（例如，框架问题）。

关于行动推理的问题

假设行动被完美执行（对计划来说是合理的？）

■ 框架问题

- △ 如何描述状态中哪些东西不会因为执行一个动作而改变
 - 问题是有很多这样的事实
 - 既有 "认识论 "问题，也有 "计算学 "问题

■ 夯实问题

- △ 执行一项行动的直接和间接影响是什么？
 - 问题是，间接效应取决于初始情况

■ 资格问题

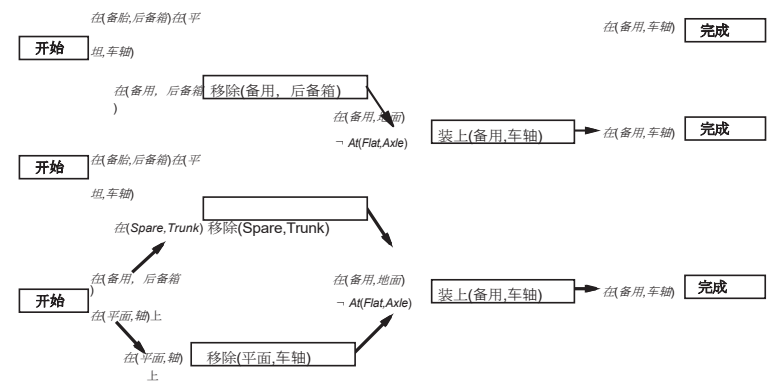
- △ 在行动的规范中需要哪些先决条件？
 - 问题是，资格取决于背景

规划

- 计划--为实现某些目标而采取的行动的顺序（或有序的集合）。
- 规划师--产生计划的问题解决者
- 目标--通常是一个连词
- 初始状态--通常是一个字词的组合
- 块状世界的例子，目标是 $on(B, C) \wedge on(C, Table)$
 - ▲ $move(C, A, Table), move(B, Table, C)$

非线性规划

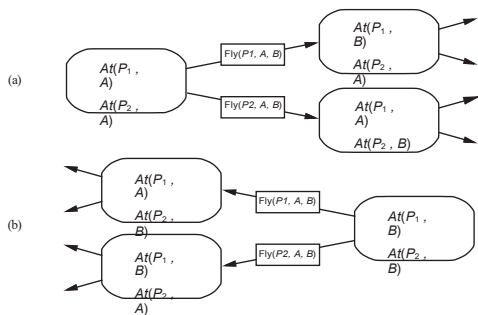
- 从目标倒退开始，尝试修复计划中的 "缺陷"。



- 最低承诺：执行可以按任何允许的顺序进行

简单的规划算法

- 正向搜索和目标回归

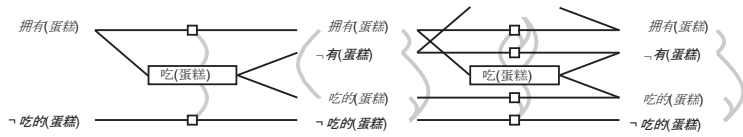


使用计划图的前向搜索

- 只考虑 "命题式 "计划
- S_i 包含所有在时间*i*可能存在的字词
- 一个 S_i ，包含了所有 可能在时间*i*满足前提条件的行动
- 与先决条件有关的行动
- 从时间*i*到时间*i+1*持续存在的字词通过行动联系起来
- 在同一时间内，动作/字词之间的互斥（mutex）链接



- 前向搜索的问题是状态空间可能非常大
- 回归的问题是，它很难，而且并不总是有效。



相互排斥

- 行动
 - ▲ 不一致的效果。一个行动否定了另一个行动的效果
 - ▲ 干扰。一个行动的效果是对另一个行动的前提条件的否定。
 - 互相竞争的需要。一个行动的前提条件与另一个行动的前提条件是相互排斥的。
- 字面意义
 - ▲ 一个字词是另一个字词的否定。
 - ▲ 不一致的支持。可能实现这两个字的每一对行动都是相互排斥的

GraphPlan算法

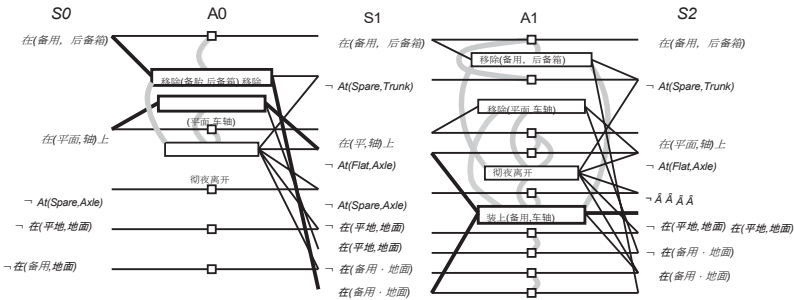
- 图形 = 带有初始状态 S 的初始计划图形 o
- nogoods = 空集
- 对于 $t = 0, - \dots$
 - ▲ 如果所有目标都是 S 中的非Mutex目标,
 - 从图表中提取解决方案
 - 图为CSP, 计划中的行动有变量T/F。
 - 或启发式地引导回归, 从 S_t 到 S_0
 - 如果解决方案有效, 则返回解决方案
 - ▲ 如果图和nogoods没有变化, 则返回失败。
 - ▲ 将图表扩展到下一个层次

GraphPlan扩展步骤

- 将行动添加到 A_i , 其前提条件是在 S_i
- 在 A 中添加 "持久性行动" i , 用于 S 中的字词。 i
- 为不能一起发生的动作添加突变链接到 A_i
- 将 A 的所有行动的效果 i 到 S 中。 $i+1$
- 将字词添加到 S_{i+1} , 用于来自 A 的持久性行动。 i
- 在 S 中添加突变体链 用于不能同时出现的字词接 $i+1$

GraphPlan实例

- 在扩展到2级后



作为约束条件满足的规划

每个规划阶段 k 的CSP（根据需要改变 k ）。

■ 变量

- ▲ 为每个字词和时间 $0, \dots, k$ 创建一个变量
- ▲ 为每个行动和时间 $0, \dots, k-1$ 创建一个变量

■ 限制条件

- ▲ 状态约束：时间 t 的字词
- ▲ 前提条件约束：时间 t 的行动和状态
- ▲ 效果约束：时间 t 的行动，时间 t 和 $t+1$ 的字词
- ▲ 行动约束：在时间 t 的行动（相互排斥）。
- ▲ 初始状态约束：时间 0 时的字词
- ▲ 目标约束：时间 k 的字词

总结

- 从哲学的角度来看，对行动的推理很有趣
- 规划方面的最新进展使效率有了很大提高
- 规划使用带有启发式的CSP框架
- 多Agent系统，动态世界更加复杂