



回顧

2.1

经典概念理论

2.2

数理逻辑

2.3

集合论

2.4

概念的现代表示理论





中國石油大學(华东)
CHINA UNIVERSITY OF PETROLEUM

计算机科学与技术学院
College of Computer Science & Technology

第三章 知识表示

人工智能课程组

智能科学系





目录

CONTENTS

3.1

知识与知识表示的概念

3.2

产生式表示法

3.3

框架表示法

3.4

状态空间表示法





3.1.1 关于知识

- 知识表示的对象：在长期的生活及社会实践中、在科学研究及实验中积累起来的对客观世界的认识与经验。
- 从实践角度来说：知识是把有关信息关联在一起所形成的信息结构。
- 知识反映了客观世界中事物之间的关系，不同事物或者相同事物间的不同关系形成了不同的知识。
- 信息关联的不同形式有：
 - 1、用的最多的是“如果……，则……”，称为**规则**
 - 如果大雁向南飞，则冬天就要来临了。
 - 2、“雪是白色的”，反应事物之间的关系，这个是**事实**



3.1.2 知识的特性

1. 相对正确性

- 任何知识都是在**一定的条件及环境下产生的**，在这种条件及环境下才是正确的。

$$1+1=2 \quad (\text{十进制})$$

$$1+1=10 \quad (\text{二进制})$$

2. 不确定性

知识状态:

- 确定性：“真”、“假”
- 不确定性：**真与假之间的中间状态**
 - ① 随机性引起的不确定性（连环计）
 - ② 模糊性引起的不确定性（张三跑得快）
 - ③ 经验引起的不确定性（专家经验）
 - ④ 不完全性引起的不确定性（火星没水）



3.1.2 知识的特性

3. 可表示性与可利用性

- 知识的可表示性: 知识可以用适当形式表示出来, 如用语言、文字、图形、神经网络等。
- 知识的可利用性: 知识可以被用来解决问题。





3.1.3 知识的表示

知识表示 (knowledge representation) : 将人类知识形式化或者模型化。

- 知识表示是对知识的一种描述, 或者说是一组约定, 一种计算机可以接受的用于描述知识的数据结构。

选择知识表示方法的原则:

- (1) 充分表示领域知识。
- (2) 有利于对知识的利用。
- (3) 便于对知识的组织、维护与管理。
- (4) 便于理解与实现。



3.1.3 知识的表示

知识表示的方法主要有：

3.2 产生式表示法

3.3 框架表示法

3.4 状态空间表示法





3.2 产生式表示法

3.2.1 产生式

3.2.2 产生式系统——动物识别系统

3.2.3 产生式系统

3.2.4 产生式表示法的特点





3.2.1 产生式

- “产生式”：1943年，美国数学家波斯特（E. Post）首先提出。
- 1972年，纽厄尔和西蒙在研究人类的认知模型中开发了基于规则的产生式系统。
- 产生式通常用于表示事实、规则以及它们的不确定性度量，适合于表示事实性知识和规则性知识。

3.2.1 产生式

1. 确定性规则知识的产生式表示

■ 基本形式: IF P THEN Q

或者: $P \rightarrow Q$

■ 例如:

r_4 : IF 动物会飞 AND 会下蛋 THEN 该动物是鸟

2. 不确定性规则知识的产生式表示

■ 基本形式: IF P THEN Q (置信度)

或者: $P \rightarrow Q$ (置信度)

例如: IF 发烧 THEN 感冒 (0.6)

3.2.1 产生式

3. 确定性事实性知识的产生式表示

- 三元组表示：（对象，属性，值）
或者：（关系，对象1，对象2）
- 例：老李年龄是40岁：（*Li*, *age*, 40）
老李和老王是朋友：（*friend*, *Li*, *Wang*）

4. 不确定性事实性知识的产生式表示

- 四元组表示：（对象，属性，值，置信度）
或者：（关系，对象1，对象2，置信度）
- 例：老李年龄很可能是40岁：（*Li*, *age*, 40, 0.8）
老李和老王不大可能是朋友：（*friend*, *Li*, *Wang*, 0.1）



3.2.1 产生式

- 产生式与谓词逻辑中的蕴含式的区别：
 - (1) 除逻辑蕴含外，产生式还包括各种操作、规则、变换、算子、函数等。例如，“如果炉温超过上限，则立即关闭风门”是一个产生式，但不是蕴含式。
 - (2) 蕴含式只能表示精确知识，而产生式不仅可以表示精确的知识，还可以表示不精确知识。蕴含式的匹配总要求是精确的。产生式匹配可以是精确的，也可以是不精确的，只要按某种算法求出的相似度落在预先指定的范围内就认为是可匹配的。



3.2.2 产生式系统——动物识别系统

张三不知道某种动物是什么，通过动物识别系统（AR）查询，过程如下：

AR：你看到的动物有羽毛吗？

张三：有羽毛。

AR：会飞吗？

张三：（经观察后）不会飞。

AR：有长腿吗？

张三：没有。

AR：会游泳吗？

张三：（看到该动物在水中）会。

AR：颜色是黑白吗？

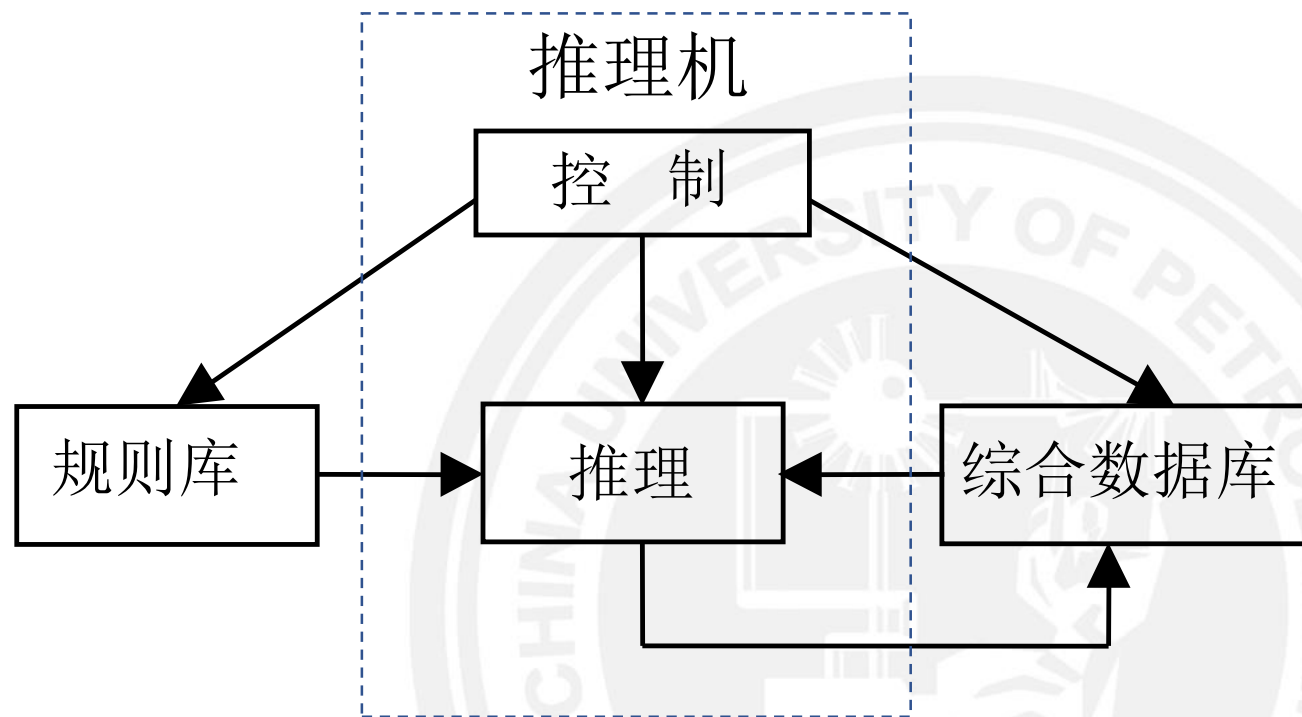
张三：是。

AR：这个动物是企鹅。



3.2.3 产生式系统

- 把一组产生式放到一起，让他们互相配合、协同作用，一个产生式生成的结论可以供另一个产生式作为已知事实使用，以求得问题的解，这样的系统称为产生式系统。
- 一般来说，一个产生式系统由规则库、综合数据库、控制系统（推理机）三部分组成。



图：产生式系统的基本结构



3.2.3 产生式系统

1. 规则库

- **规则库**：用于描述相应领域内知识的产生式集合。

2. 综合数据库

- **综合数据库**（事实库、上下文、黑板等）：一个用于存放问题求解过程中各种当前信息的数据结构。

3. 控制系统

- **控制系统**（推理机构）：由一组程序组成，除了推理算法，还负责整个产生式系统的运行，实现对问题的求解。

3.2.3 产生式系统

3. 控制系统（续）

控制系统要做以下几项工作：

- （1）推理。从规则库中选择与综合数据库中的已知事实进行匹配。
- （2）冲突消解。匹配成功的规则可能不止一条，进行冲突消解。
- （3）执行某一规则时，如果其右部是一个或多个结论，则把这些结论加入到综合数据库中；如果其右部是一个或多个操作，则执行这些操作。
- （4）对于不确定性知识，在执行每一条规则时还要按一定的算法计算结论的不确定性。
- （5）检查推理终止条件。检查综合数据库中是否包含了最终结论，决定是否停止系统的运行。



3.2.4 产生式表示法的特点

1. 适合产生式表示的知识

- (1) 具有因果关系的过程性知识，非结构化的知识表示形式。
- (2) 经验性及不确定性的知识，且相关领域中对这些知识没有严格、统一的理论。
- (3) 领域问题的求解过程可被表示为一系列相对独立的操作，且每个操作可被表示为一条或多条产生式规则。

2. 产生式表示法的优点

- (1) 自然性
- (2) 模块性
- (3) 有效性
- (4) 清晰性

3. 产生式表示法的缺点

- (1) 效率不高
- (2) 不能表达结构性知识



3.3 框架表示法

- **1975年**，美国人工智能学者明斯基提出了框架理论。该理论基于人们对现实世界中各种事物的认识都是以一种类似于框架的结构存储在记忆中的。
- **框架表示法**：一种结构化的知识表示方法，已在多种系统中得到应用。



3.3.1 框架的一般结构

- **框架 (frame)**：一种描述所论对象（一个事物、事件或概念）属性的数据结构。
- 一个框架由若干个被称为“槽” (slot) 的结构组成，每一个槽又可根据实际情况划分为若干个“侧面” (facet) 。
 - 一个槽用于描述所论对象某一方面的属性。
 - 一个侧面用于描述相应属性的一个方面。
 - 槽和侧面所具有的属性值分别被称为槽值和侧面值。



3.3.1 框架的一般结构

<框架名>

槽名1:	侧面名 ₁₁	侧面值 ₁₁₁ , ..., 侧面值 _{11P1}
	侧面名 _{1m}	侧面值 _{1m1} , ..., 侧面值 _{1mPm}
槽名n:	侧面名 _{n1}	侧面值 _{n11} , ..., 侧面值 _{n1P1}
	侧面名 _{nm}	侧面值 _{nm1} , ..., 侧面值 _{nmPm}
约束:	约束条件 ₁	
	约束条件 _n	



3.3.2 用框架表示知识的例子

■ 例1 教师框架

框架名：〈教师〉

姓名：单位（姓、名）

年龄：单位（岁）

性别：范围（男、女）

缺省：男

职称：范围（教授，副教授，讲师，助教）

缺省：讲师

部门：单位（系，教研室）

住址：〈住址框架〉

工资：〈工资框架〉

开始工作时间：单位（年、月）

截止时间：单位（年、月）

缺省：现在



3.3.2 用框架表示知识的例子

■ 例1 教师事例框架

当把具体的信息填入槽或侧面后，就得到了相应框架的一个事例框架。

框架名：〈教师-1〉

姓名：李四

年龄：38

性别：男

职称：副教授

部门：软件工程系

住址：〈adr-1〉

工资：〈sal-1〉

开始工作时间：2004.9

截止时间：2019.9



3.3.2 用框架表示知识的例子

■ 例2 教室框架

框架名：〈教室〉

墙数：

窗数：

门数：

座位数：

前墙：〈墙框架〉

后墙：〈墙框架〉

左墙：〈墙框架〉

右墙：〈墙框架〉

门：〈门框架〉

窗：〈窗框架〉

黑板：〈黑板框架〉

天花板：〈天花板框架〉

讲台：〈讲台框架〉



3.3.2 用框架表示知识的例子

例3 将下列一则**地震消息**用框架表示：“某年某月某日，某地发生6.0级地震，若以膨胀注水孕震模式为标准，则三项地震前兆中的波速比为0.45，水氡含量为0.43，地形改变为0.60。”

解：地震消息用框架如下图所示。



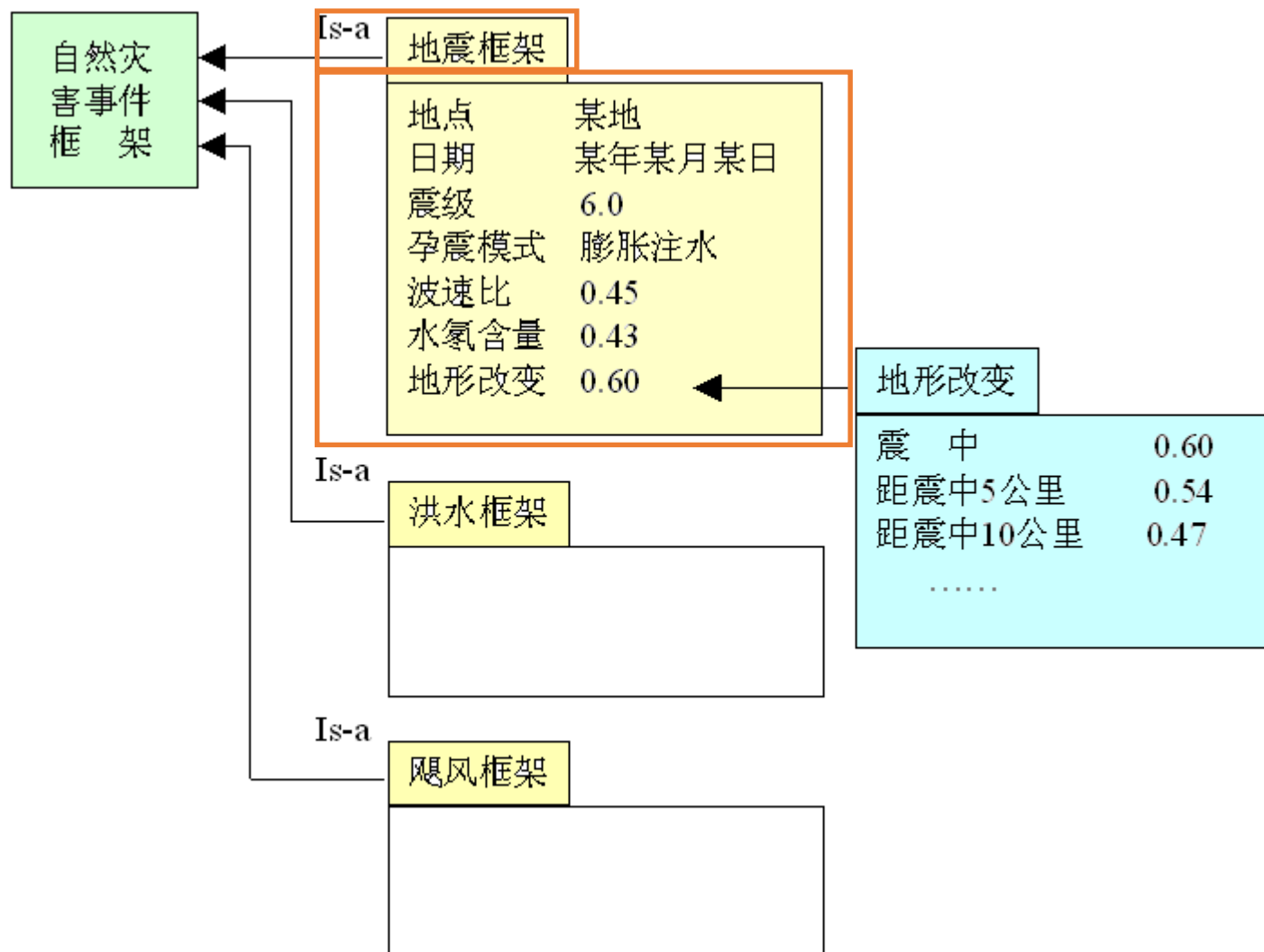
3.3.2 用框架表示知识的例子

例3 将下列一则**地震消息**用框架表示：“某年某月某日，某地发生6.0级地震，若以膨胀注水孕震模式为标准，则三项地震前兆中的波速比为0.45，水氦含量为0.43，地形改变为0.60。”

解：地震消息用框架如下图所示。

框架名：〈地震〉
地 点：某地
日 期：某年某月某日
震 级：6.0
波 速 比：0.45
水 氦 含 量：0.43
地 形 改 变：0.60

3.3.2 用框架表示知识的例子





3.3.2 框架表示法的特点

(1) 结构性

便于表达结构性知识，能够将知识的内部结构关系及知识间的联系表示出来。

(2) 继承性

框架网络中，下层框架可以继承上层框架的槽值，也可以进行补充和修改。

(3) 自然性

框架表示法与人在观察事物时的思维活动是一致的。



3.4 状态空间表示法

3.4.1 状态空间表示

- **状态空间** (state space) 是利用状态变量和操作符号表示系统或问题的有关知识的符号体系。
- **状态** (state) : 表示问题解法中每一步问题状况的数据结构
- **操作符号** (operator) : 把问题从一种状态变换为另一种状态的手段
- 状态空间方法: 基于解答空间的问题表示和求解方法, 它是以状态和算符为基础来表示和求解问题的。

3.4.1 状态空间表示

状态空间可以用一个四元组表示：

$$(S, O, S_0, G)$$

S : 状态集合，每一个元素表示一个状态，状态是某种结构的符号或数据。

O : 操作算子的集合，利用算子可以将一个状态转换为另一个状态。

S_0 : 问题的初始状态集合， $S_0 \subset S$

G : 目的状态的集合，可以是若干具体状态，也可以是满足某些性质的路径信息描述。是 S 的非空子集。



3.4.1 状态空间表示

从 S_0 结点到G结点的路径称为求解路径。求解路径上的操作算子序列为状态空间的一个解。

$$S_0 \xrightarrow{O_1} S_1 \xrightarrow{O_2} S_2 \xrightarrow{O_3} \dots \xrightarrow{O_k} G$$

操作算子序列 $O_1, O_2, O_3, \dots, O_k$ 是状态空间的一个解。通常解不是唯一的。



3.4.1 状态空间表示

- 任何类型的数据结构都可以用来描述状态

如: 符号、字符串、向量、多维数组、树、表格等。

- 所选择的数据结构形式要与状态所蕴含的某些特性具有相似性。

如八数码问题, 3×3 的阵列是合适的状态描述方式。

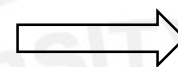


3.4.1 状态空间表示

- 状态空间：8个数码的任何一种摆法是一种状态，所有的摆法构成状态集，即状态空间S。
 - 状态空间有 $9!$ 个状态。
- 操作算子：
 - 数码移动：4（方向） \times 8（数码）=32个算子
 - 空格（0）移动：up, down, left, right. 4个算子

2		3
1	8	4
7	6	5

初始状态



1	2	3
8		4
7	6	5

目标状态

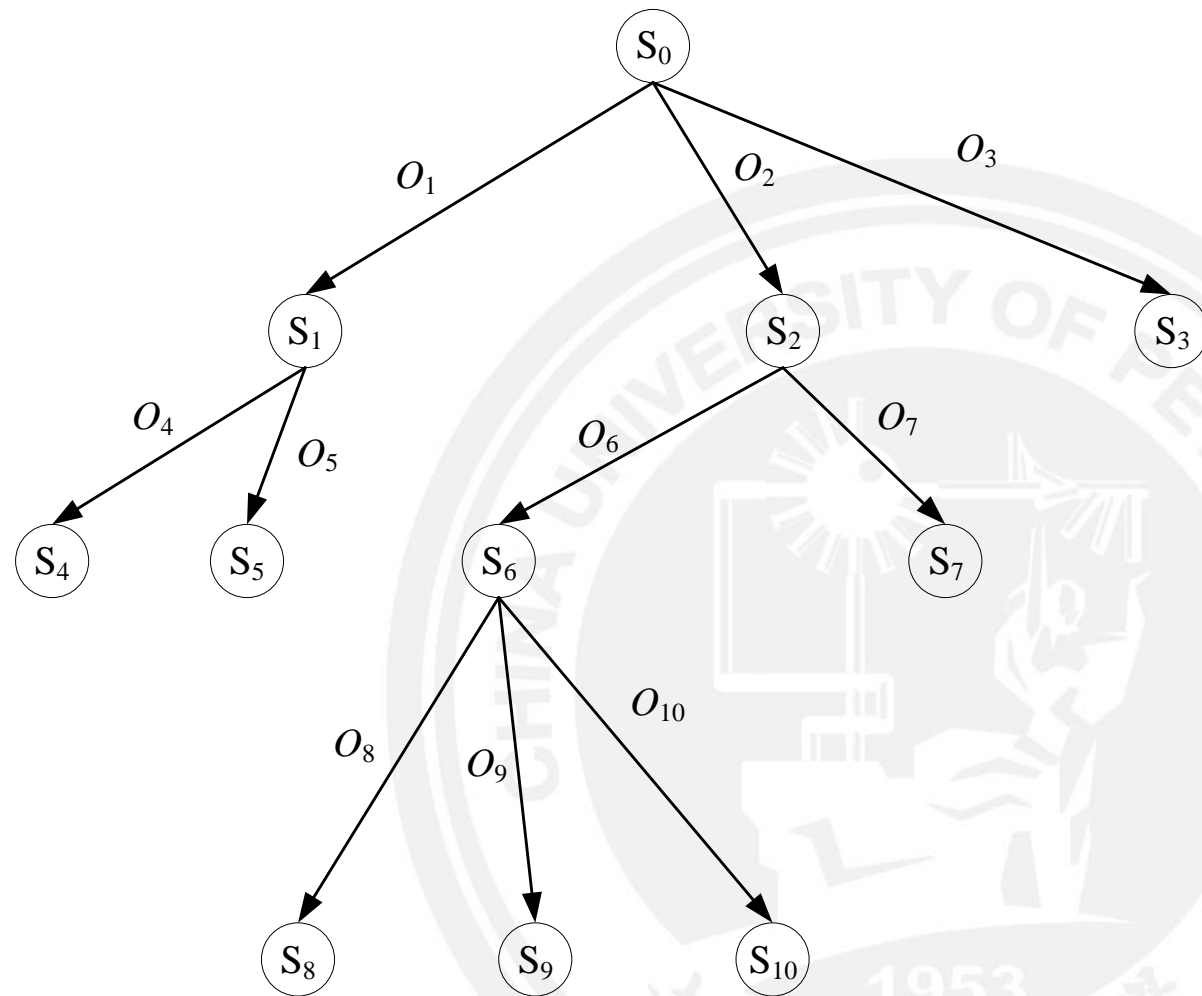


3.4.2 状态空间的图描述

- 状态空间可以用有向图来描述，图的节点表示问题的状态，图的弧表示状态之间的关系。
- 初始状态对应于实际问题的已知信息，是图中的根节点。在问题的状态空间描述中，寻找从一种状态转换为另一种状态的某个操作算子序列等价于在一个图中寻找某一路径。

3.4.2 状态空间的图描述

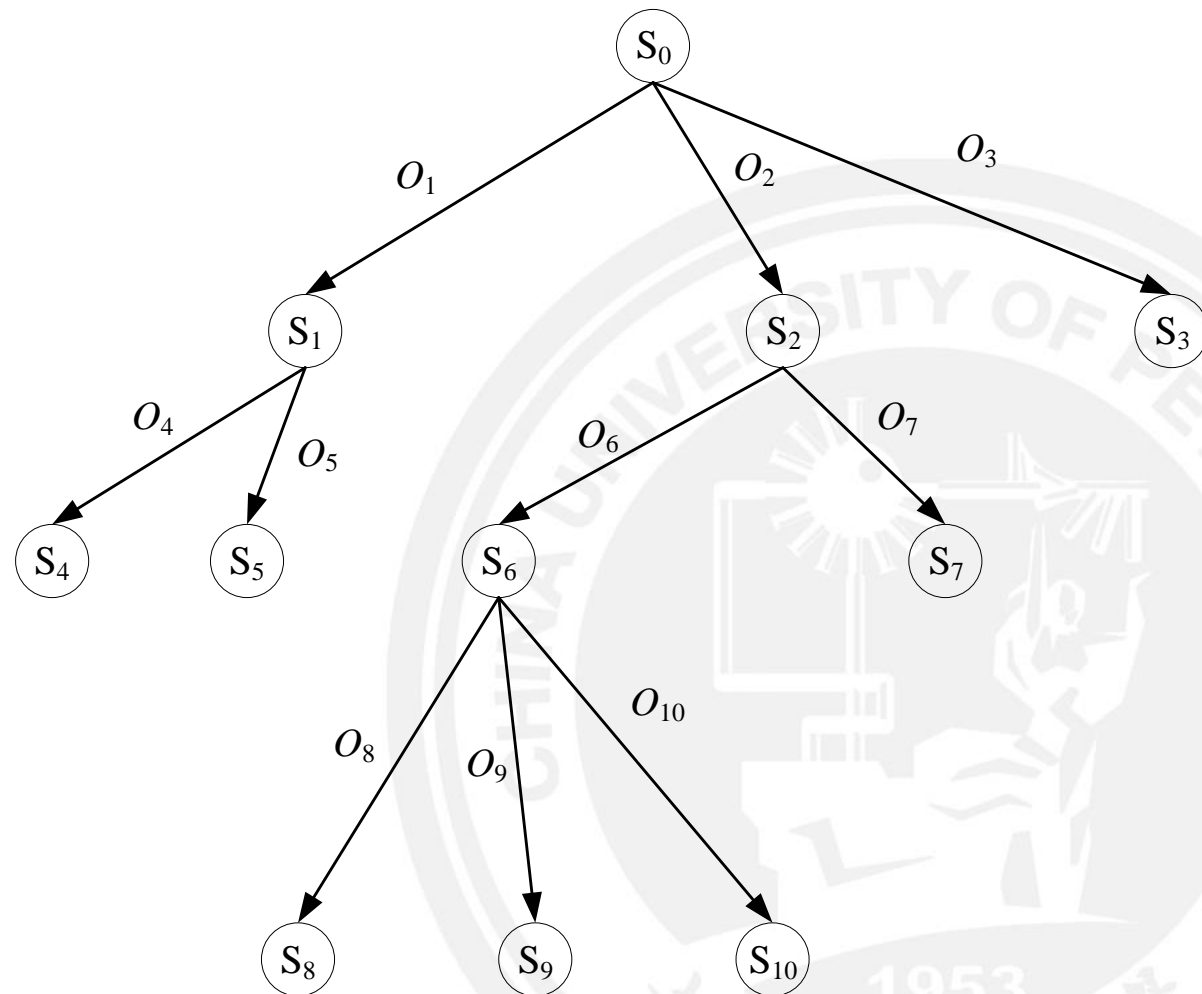
- 对状态 S_0 允许使用操作算子 O_1, O_2, O_3 , 分别转换为 S_1, S_2, S_3 .
- 对于 $S_{10} \in G$, 则解是什么?





3.4.2 状态空间的图描述

- 对状态 S_0 允许使用操作算子 O_1, O_2, O_3 , 分别转换为 S_1, S_2, S_3 .
- 对于 $S_{10} \in G$, 则 O_2, O_6, O_{10} 是一个解



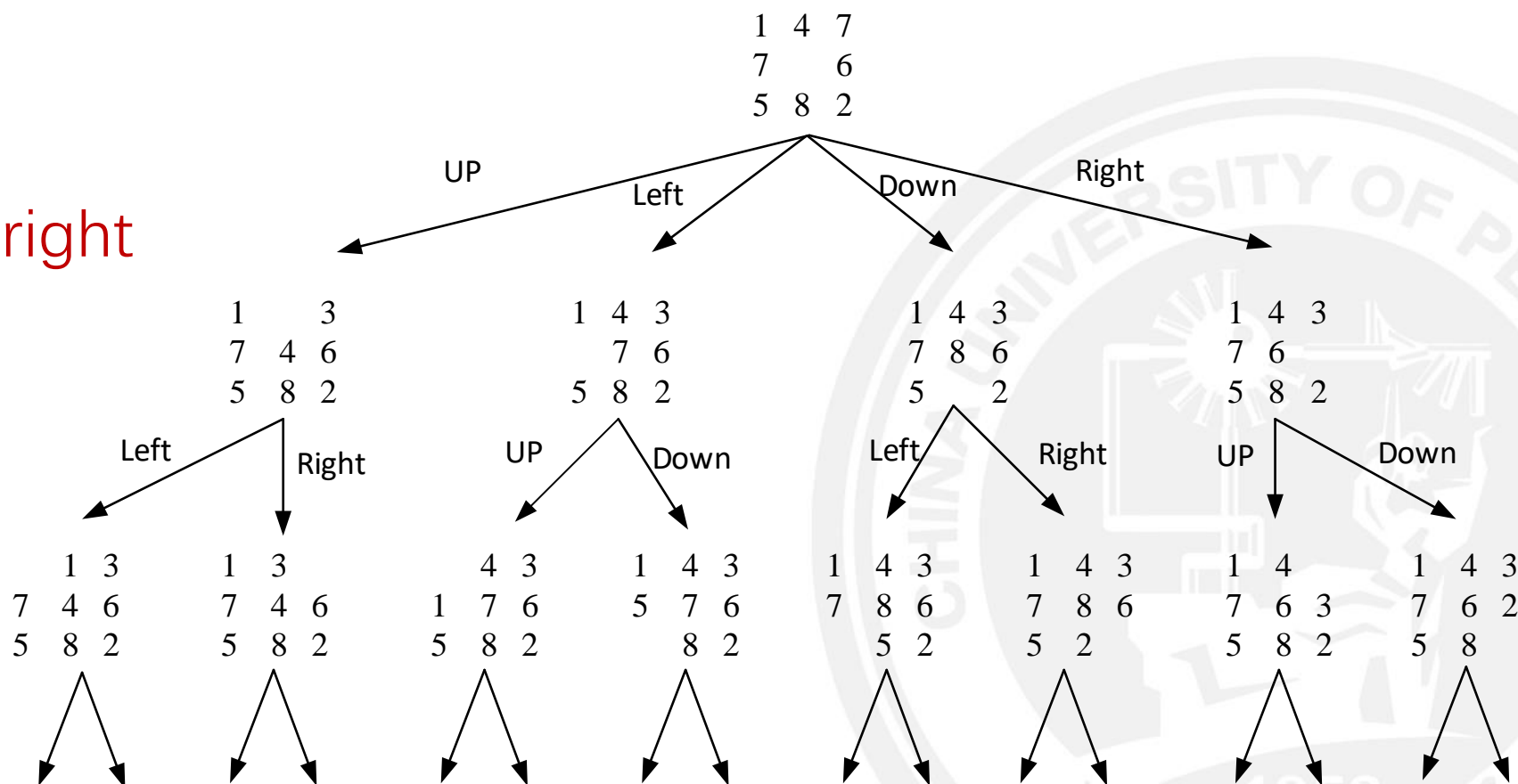


3.4.2 状态空间的图描述

例1、八数码问题

空格 (0) 移动:

- 4个算子
- up, left, down, right

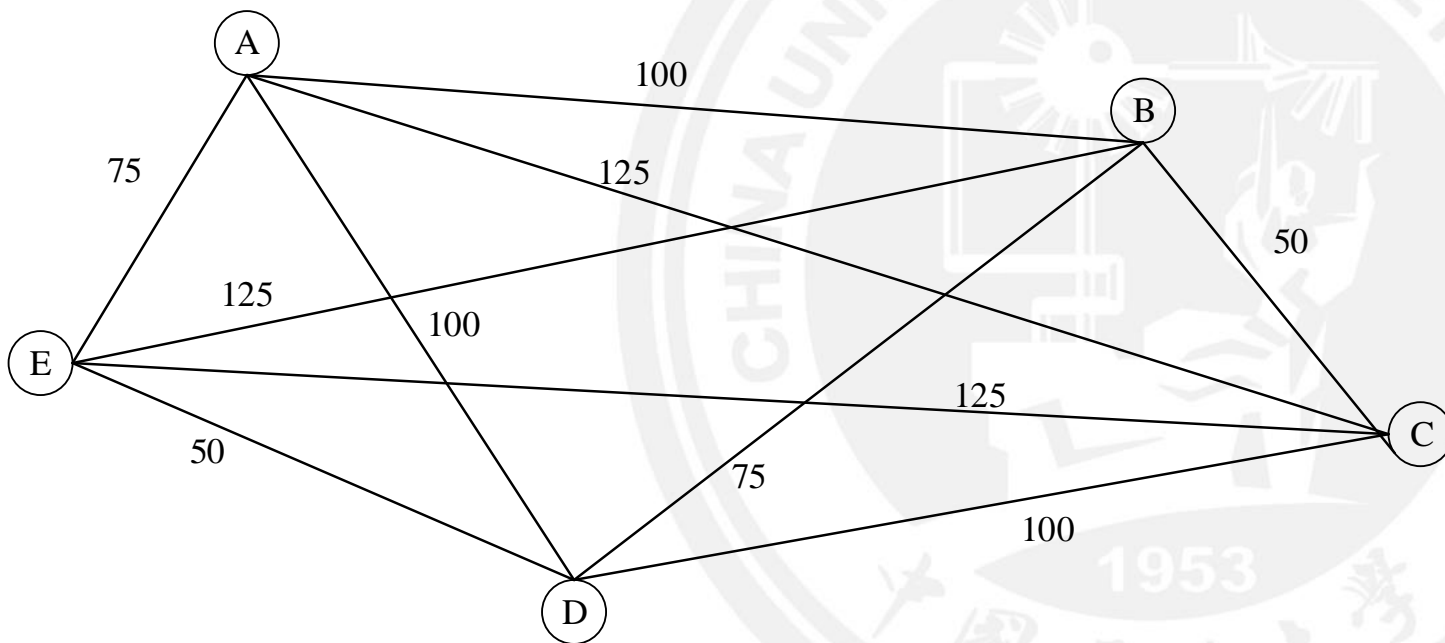




3.4.2 状态空间的图描述

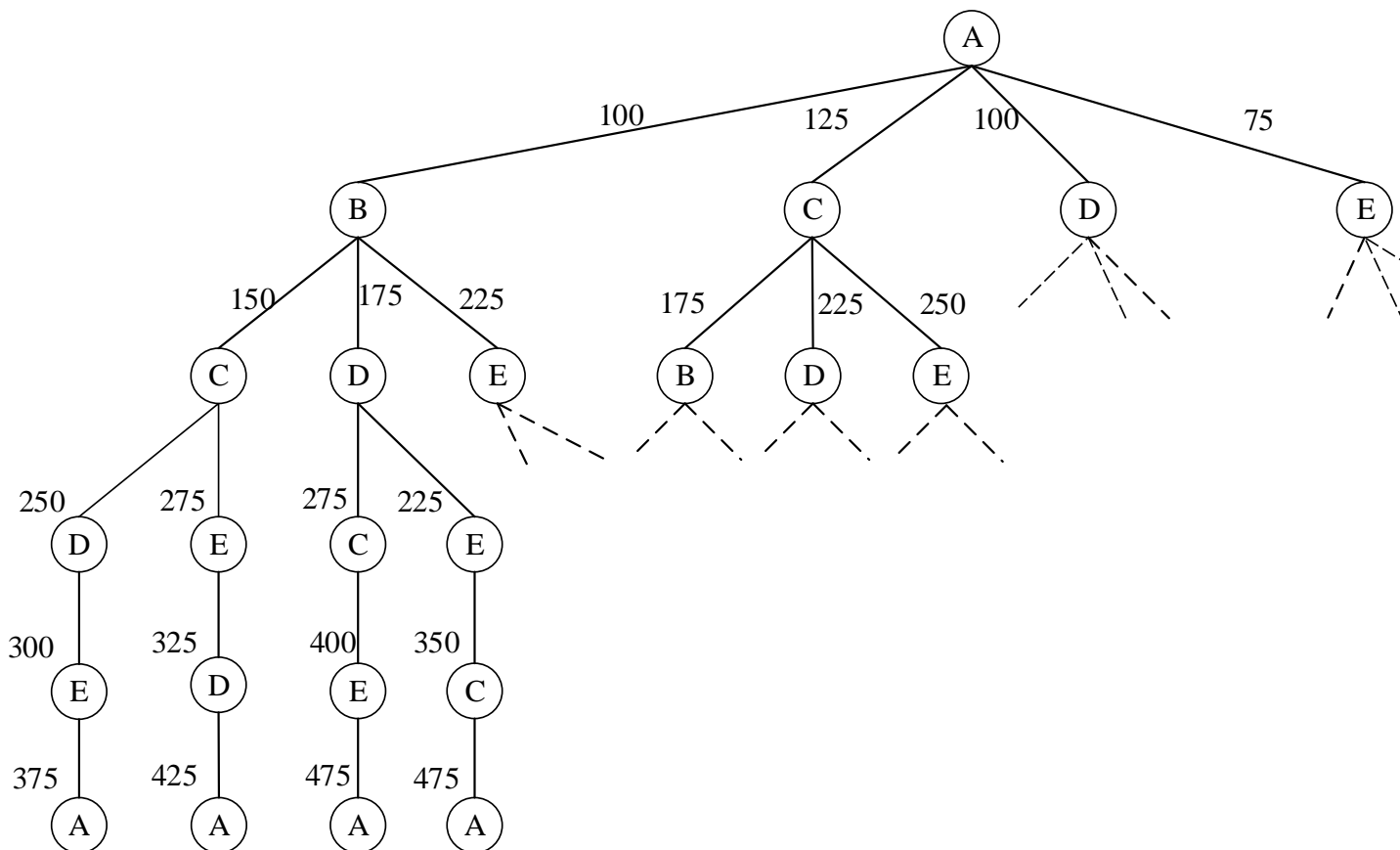
例2、旅行商问题

- 给定一系列城市 and 每对城市之间的距离，求解访问每一座城市一次并回到起始城市的最短回路。
- 节点代表城市，连线代表路径，线上数值是路径的距离（权重）
- 各个算子的开销不同





3.4.2 状态空间的图描述



路径:	路径:	路径:	路径:
ABCDEA	ABCEDA	ABDCE	ABDECA
费用:	费用:	费用:	费用:
375	425	475	475

11

- 最优解: 375 (A, B, C, D, E)

问题

- 城市越多复杂度越高, 开销越大, 问题规模不能太大

需要研究

- 在有限时间内给出具有最优解的搜索算法



THANKS

