西安交通大學
XI'AN JIAOTONG UNIVERSITY

Systems Engineering Institute
Ministry of Education Key Lab for Intelligent Networks and Network Security

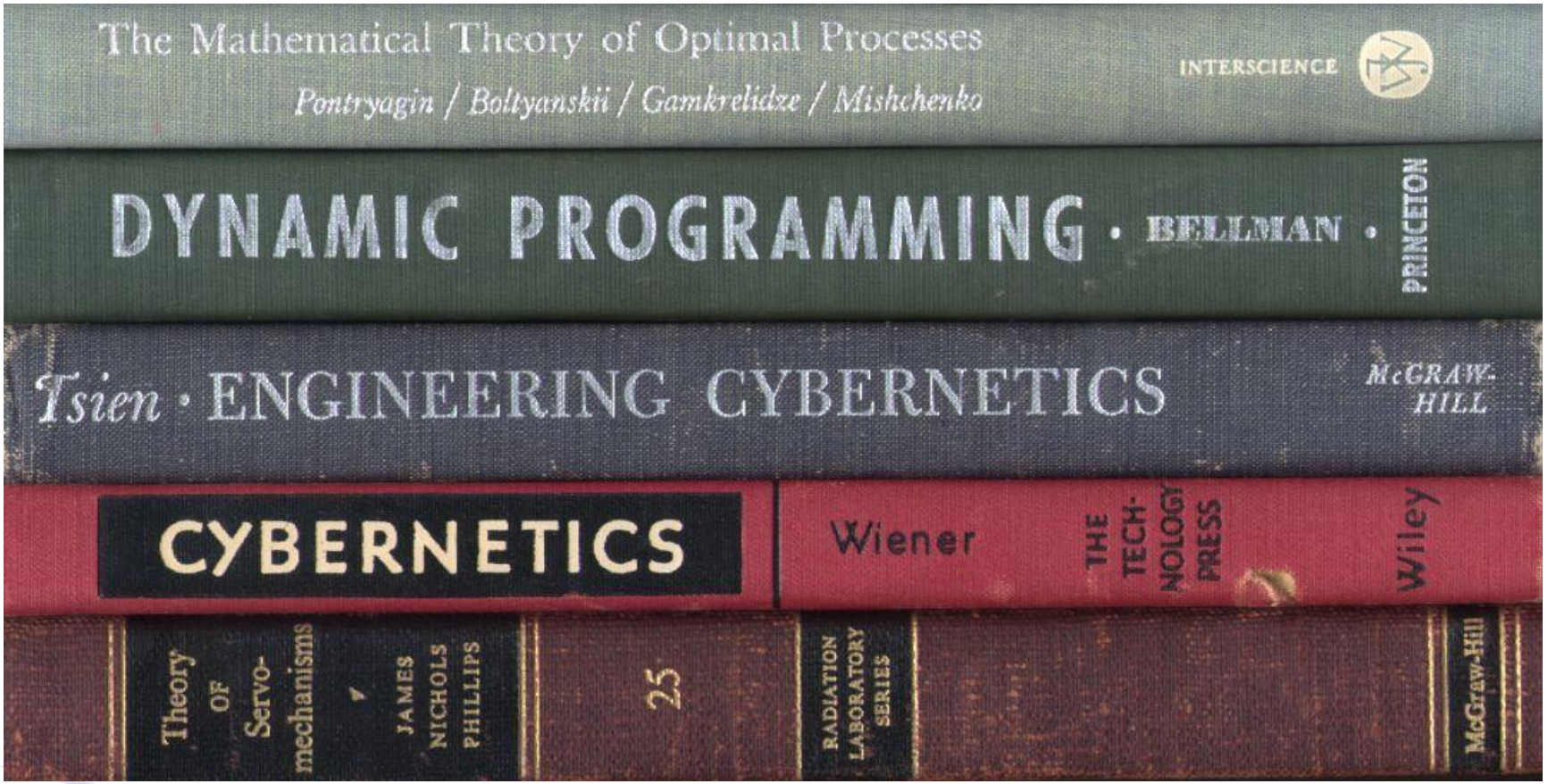
动态规划
Dynamic Programming

电信学院·自动化科学与技术系
系统工程研究所
吴江

Outline

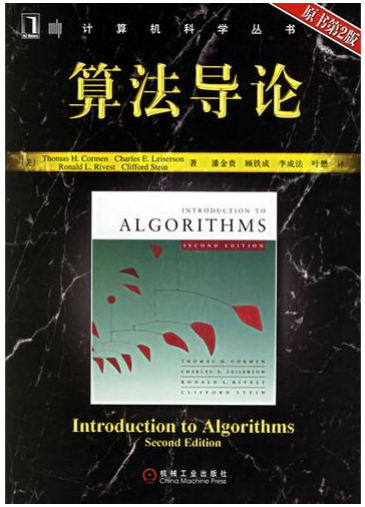
- 多阶段决策问题的例子
- 动态规划的基本概念

The Beginning - - KARL J. ÅSTRÖM



动态规划

贝尔曼, R.



Introduction to Reinforcement Learning with David Silver

Mastering the game of Go with deep neural networks and tree search

David Silver[1]*, Aja Huang[1]*, Chris J. Maddison[1], Arthur Guez[1], Laurent Sifre[1], George van den Driessche[1], Julian Schrittwieser[1], Ioannis Antonoglou[1], Veda Panneershelvam[1], Marc Lanctot[1], Sander Dieleman[1], Dominik Grewe[1], John Nham[2], Nal Kalchbrenner[1], Ilya Sutskever[2], Timothy Lillicrap[1], Madeleine Leach[1], Koray Kavukcuoglu[1], Thore Graepel[1] & Demis Hassabis[1]

Lecture 1: Introduction to Reinforcement Learning

Lecture 2: Markov Decision Processes

Lecture 3: Planning by Dynamic Programming

Lecture 4: Model-Free Prediction

Lecture 5: Model-Free Control

Lecture 6: Value Function Approximation

Lecture 7: Policy Gradient Methods

Lecture 8: Integrating Learning and Planning

Lecture 9: Exploration and Exploitation

Lecture 10: Case Study: RL in Classic Games

https://deepmind.com/learning-resources/-introduction-reinforcement-learning-david-silver

动态规划(Dynamic Programming, DP):

- 求解和时间有关的动态系统多阶段决策的有力工具
- 美国，Bellman, 1957
- 也适用于与时间无关的静态系统最优决策

动态规划是一种求解最优决策问题的重要思想，
而不是一种具体的算法

- 不能建立一个标准的DP算法流程，具体算法与问题结构有关
- 理论上，DP可以求解所有优化问题(离散、连续、无穷维)
- 很多重要应用问题由于采用了DP思想得以成功解决

多阶段决策问题

动态规划是一种解决最优决策问题的**重要思想**，而不是一种具体的算法

$$\min \quad z = f(x)$$
$$s.t \quad x \in D, D \subset R^n$$

$$\min \quad z = c^T x$$
$$s.t. \quad A^{(1)} x \le b^{(1)}$$
$$A^{(2)} x = b^{(2)}$$

$$\min \quad z = f(x)$$
$$s.t. \quad A^{(1)} x = b^{(1)}$$
$$x_j \in I$$

$$\min \quad z = f(x)$$
$$s.t. \quad g_i(x) \le 0$$
$$h_j(x) = 0$$

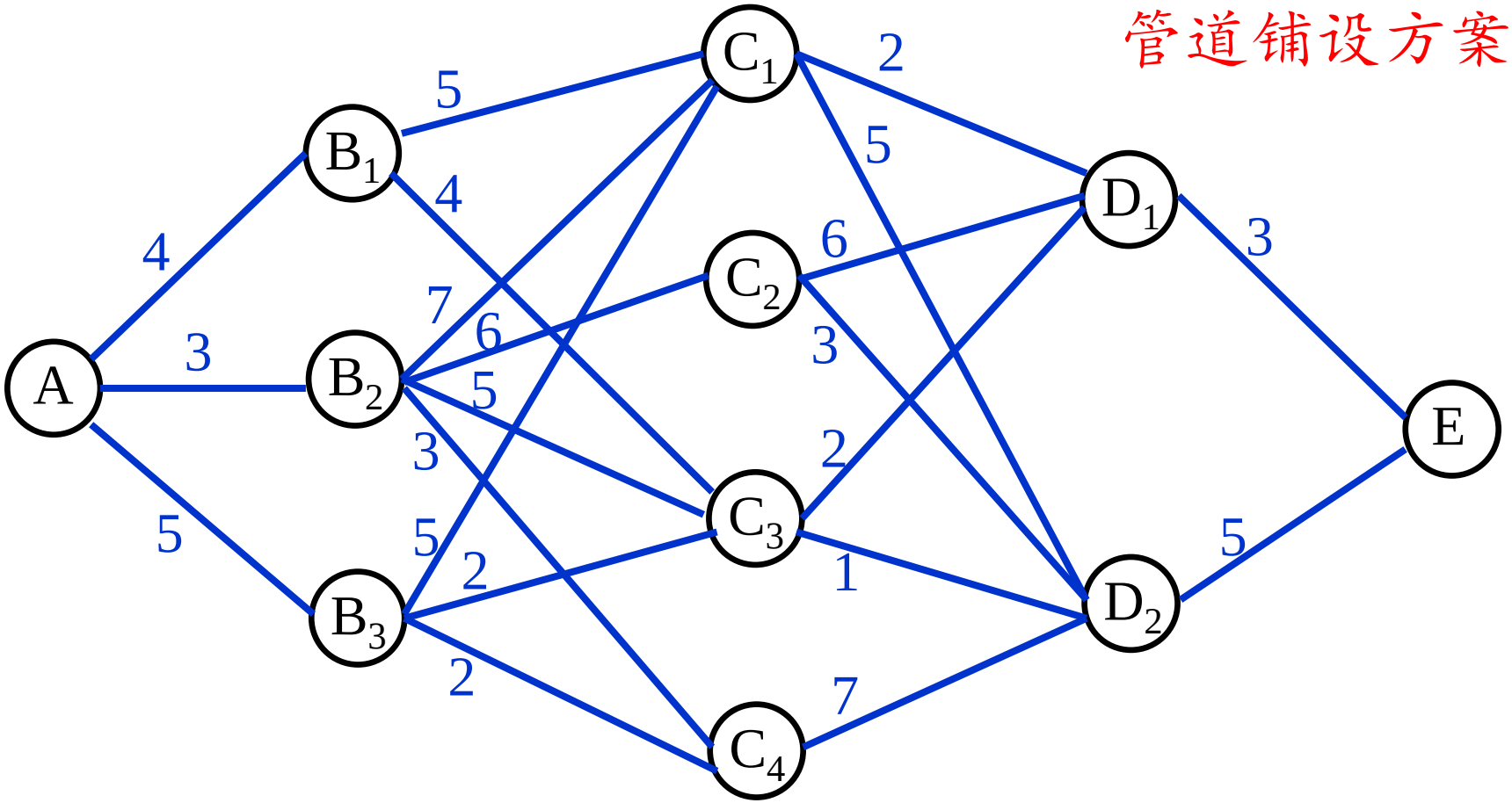
多阶段决策问题:

决策的全过程依据时间或空间划分若干个联系的阶段。在各阶段中，人们都需要作出方案的决策，并且当一个阶段的决策之后，常常影响到下一个阶段的决策，从而影响整个过程的活动。

多阶段决策问题举例

例1. 管道铺设问题(纯离散问题)。

求总费用最少的
管道铺设方案



源头　　第一中间站候选位置　　第二中间站候选位置　　第三中间站候选位置　　终点

多阶段决策问题举例

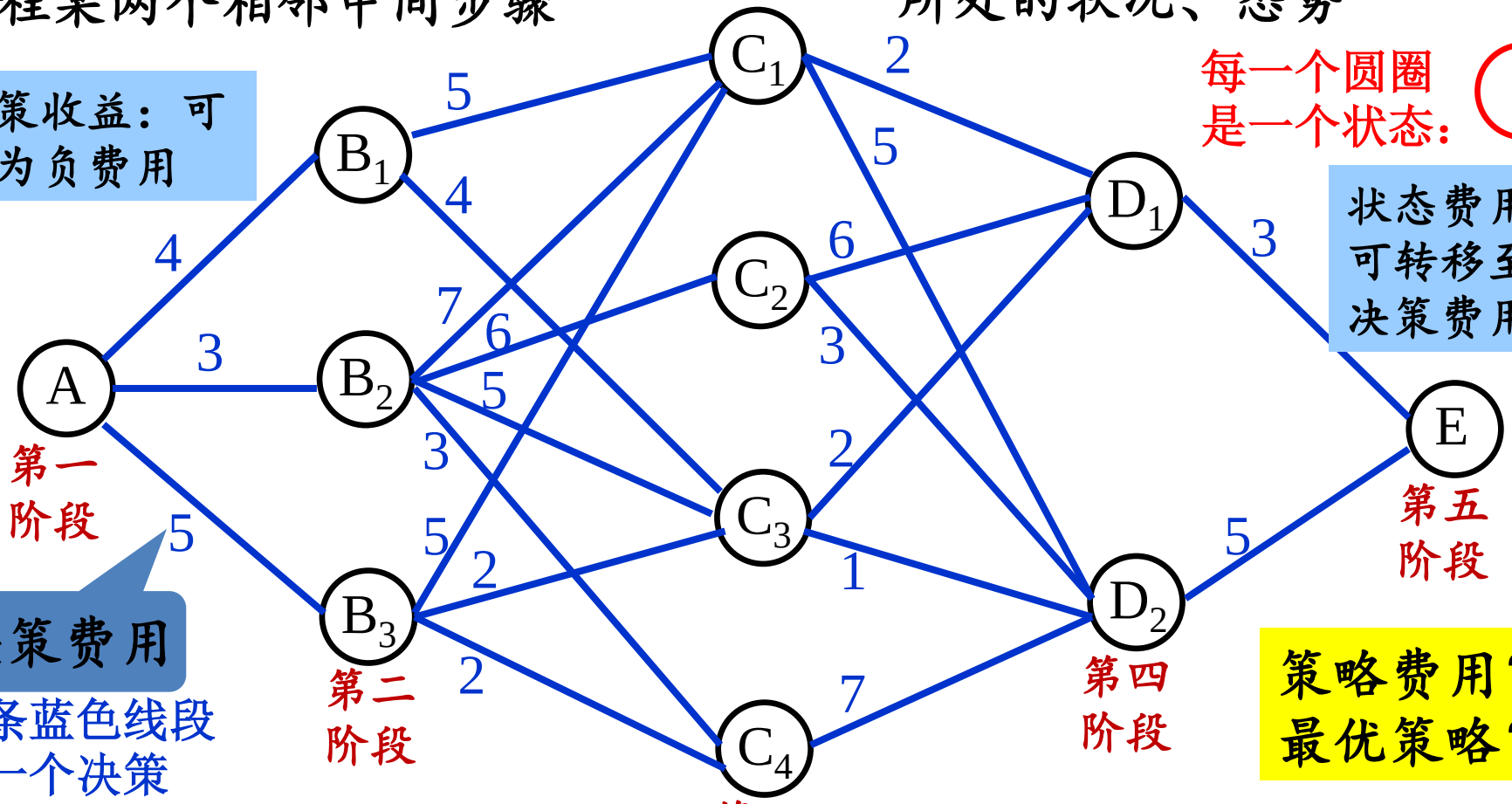
阶段(Stage)：用来划分决策过程某两个相邻中间步骤

状态(State)：每个阶段系统所处的状况、态势

决策收益：可视为负费用

每一个圆圈是一个状态：*

状态费用：可转移至决策费用



决策费用

每条蓝色线段是一个决策

策略费用？最优策略？

决策(Decision)：一个状态演变至下阶段另一状态的方式

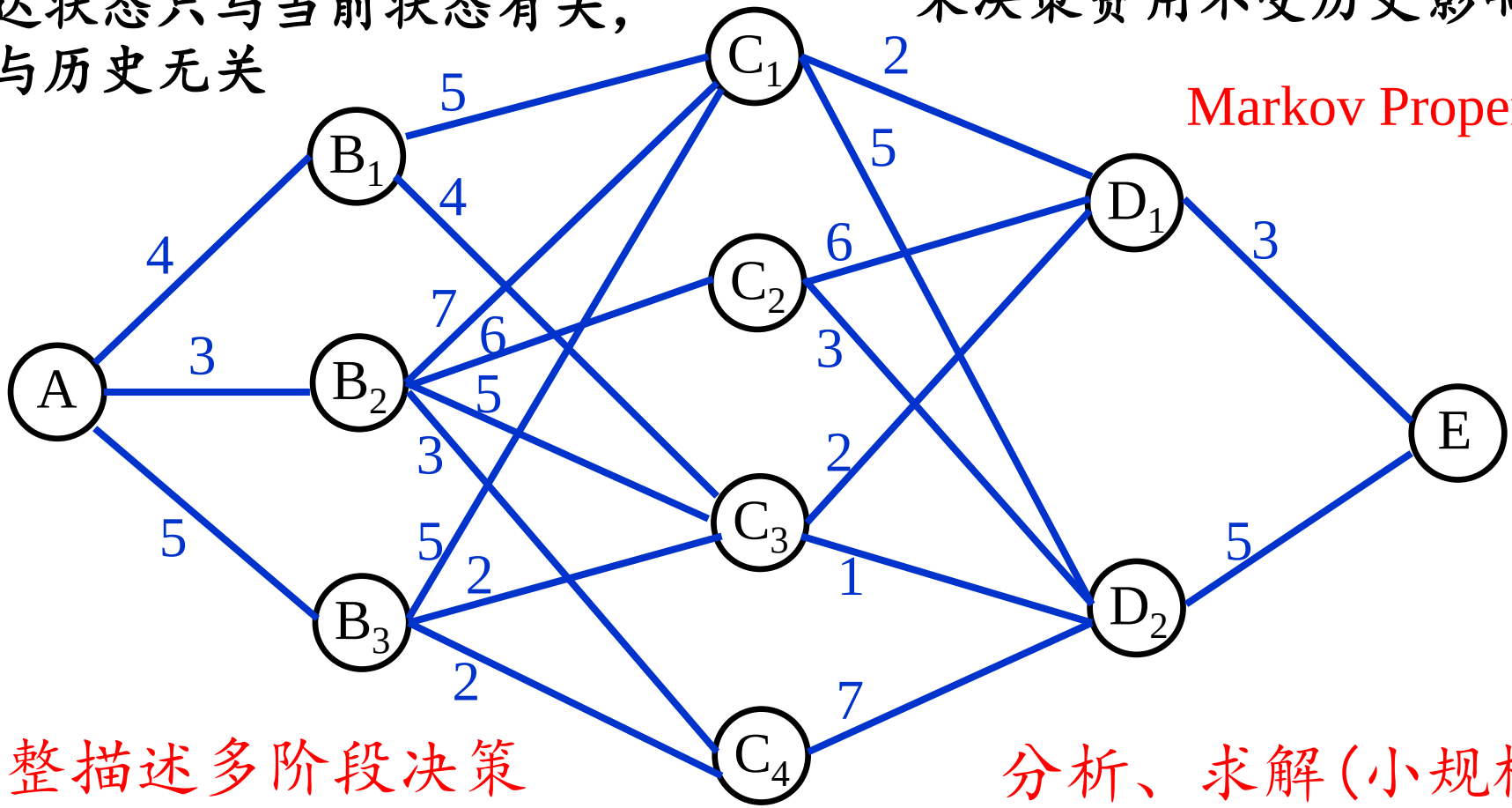
策略(Policy)：首尾相接的完整决策链

多阶段决策问题举例

状态演变的**无后效性**：未来可达状态只与当前状态有关，而与历史无关

决策费用的**无后效性**：未来决策费用不受历史影响

Markov Property



完整描述多阶段决策过程的重要辅助工具

分析、求解(小规模问题)的重要途径

状态转移图

多阶段决策问题举例

例2. 多阶段资源分配问题(时间离散、状态连续问题)。

某种资源，初始拥有量x_0，投入A，B两种生产。以y_0投入A，x_0-y_0投入B，利润为$g(y_0)+h(x_0-y_0)$ $(g(0)=h(0)=0)$。两种生产的资源回收率分别为a, b $(a, b\in(0, 1))$。回收的资源再投入生产：$x_1=ay_0+b(x_0-y_0)$，以y_1投入A，x_1-y_1投入B，……，共重复生产n次，如何安排各次的资源分配以使总利润最大？

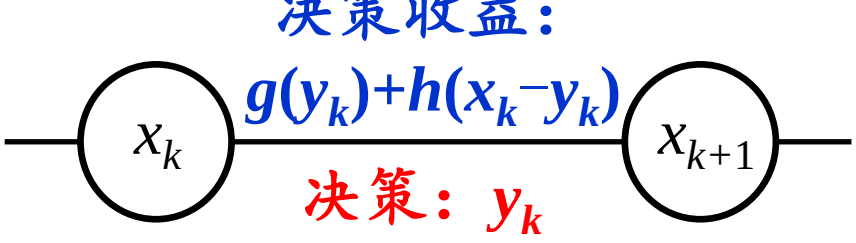
解：

$$\max_{x_k,\,y_k}\sum_{k=0}^{n-1}[g(y_k)+h(x_k-y_k)]$$

$$s.t.\quad x_{k+1}=a\cdot y_k+b\cdot(x_k-y_k);\qquad k=0,1,\cdots,n-2$$

$$0\le y_k\le x_k;\qquad k=0,1,\cdots,n-1$$

$$x_0=C$$



状态转移示意图

思考：阶段、状态、决策、策略、无后效性？

多阶段决策问题分类

基本概念：阶段、状态、决策、策略、无后效性

- **确定性/不确定性**：不含/含有随机因素
- **定期/不定期**：有多少个阶段事先已知/未知(或为无穷)
- **离散时间/连续时间**：决策过程中的时间是离散/连续变化的
- **离散/连续/混合状态**：状态向量各分量是哪一类型的变量

各种组合下的问题在现实世界中均存在

例1(管道铺设)、例2(多阶段资源分配)、生产调度、排队系统、下棋、工程项目管理、机器人控制、……

重点讨论确定性、离散时间问题

确定性定期离散时间问题

时间(阶段)：$t_0, t_1, \cdots, t_N$，简记为0, 1, 2, …, N

状态：$x_k : k = 0,1,\cdots N.\quad x_k \in \Omega_k \subset R^{n_k}$，$\Omega_k$为状态空间

不失一般性，假定初始和最终阶段仅包含一个状态，
可以通过引入虚拟阶段及虚拟状态实现此转化

决策：$u_k : k = 0,1,\cdots N-1.\quad u_k \in D_k(x_k) \subset R^{m_k}$，$D_k(x_k)$为决策空间

允许决策集合(决策空间)受当前状态影响

状态转移方程：$x_{k+1} = \phi(x_k, u_k, k);\quad k = 0,1,\cdots N-1.$

决策费用：$G(x_k, u_k, k);\quad k = 0,1,\cdots N-1.$

控制/决策目标：$\min \sum_{k=0}^{N-1} G(x_k, u_k, k).$

建模关键：满足无后效性的阶段划分及状态定义

确定性定期离散时间问题

确定性定期离散时间多阶段决策问题基本模型

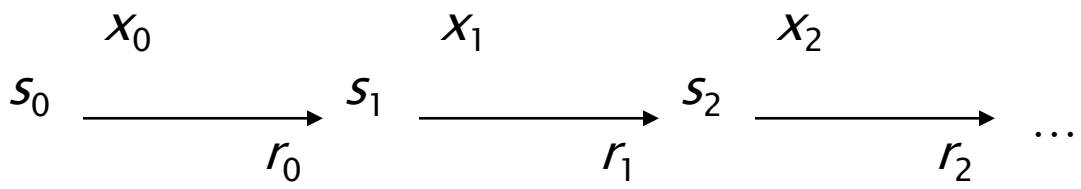
$$
\begin{cases}
\min \sum_{k=0}^{N-1} G(x_k, u_k, k). & \text{决策目标} \\
s.t. \quad x_{k+1} = \phi(x_k, u_k, k); \quad k = 0,1,\cdots N-1. & \text{状态转移约束} \\
x_k \in \Omega_k \subset R^{n_k}; \quad k = 0,1,\cdots N. & \text{状态空间约束} \\
u_k \in D_k(x_k) \subset R^{m_k}; \quad k = 0,1,\cdots N-1. & \text{决策空间约束}
\end{cases}
$$

思考：和一般非线性规划相比，有何结构特点？

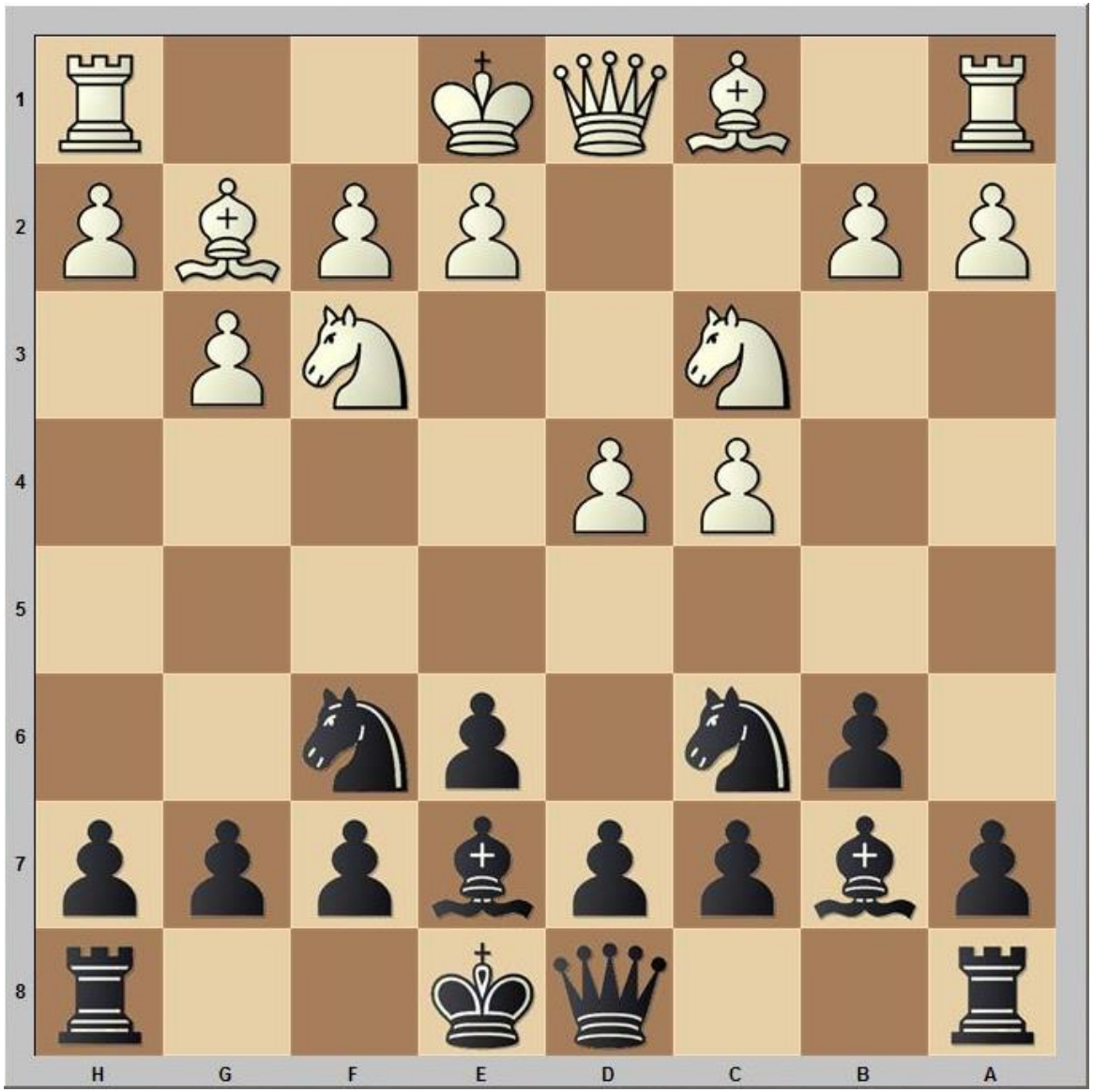
建模关键：满足无后效性的阶段划分及状态定义

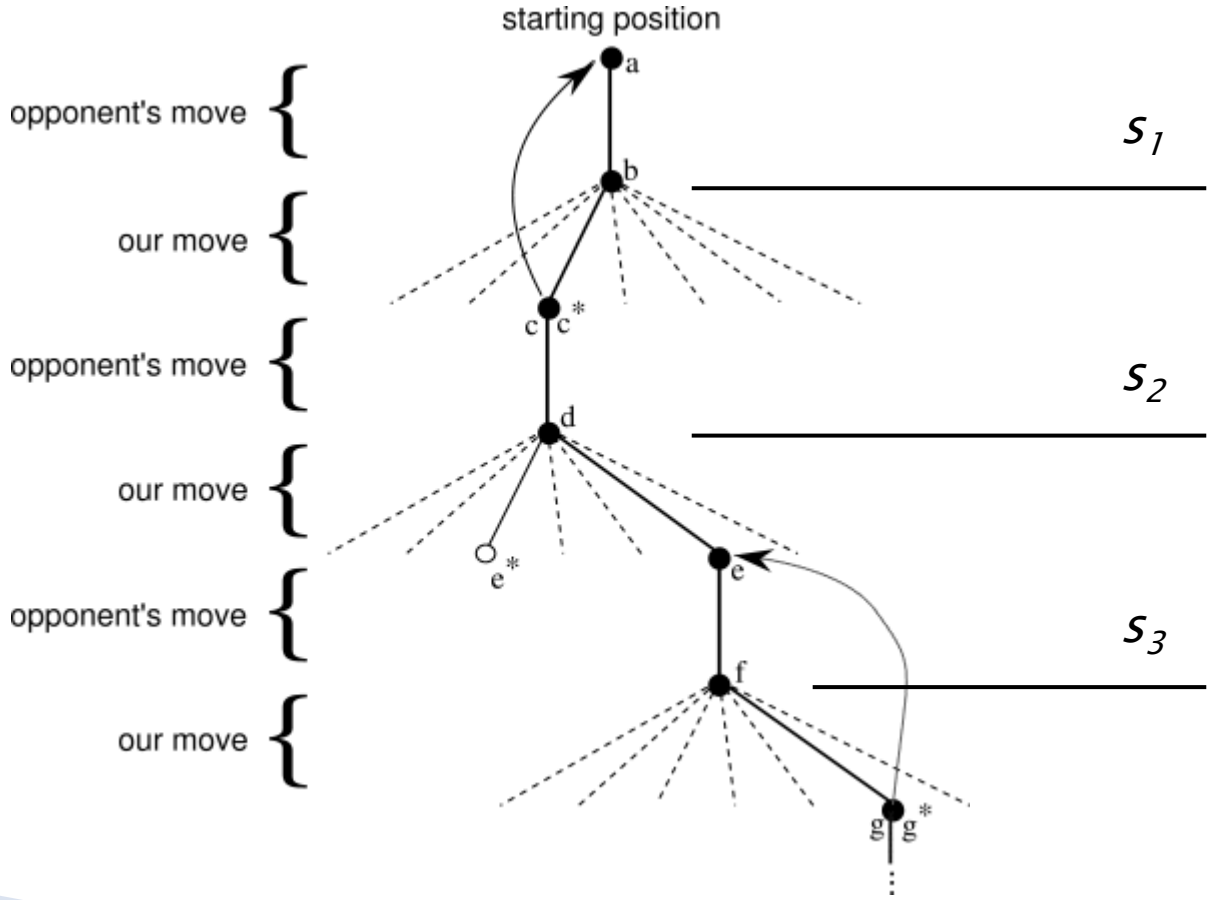
小结

- 两个问题都可以看成一个**多阶段的决策问题**
- 各个阶段**决策的选取**不是任意的，它依赖于**当前**面临的状态，又**给以后的发展**以影响
- 当各个阶段的决策确定之后，就组成了一个**决策序列**，也就决定了整个过程的一条活动路线
- 这种把一个问题变成一个前后关系具有链状态结构的多阶段决策过程，也称为**序贯决策过程**



例:




starting position
opponent's move
our move
opponent's move
our move
opponent's move
our move
a
b
c
c*
d
e*
e
f
g
g*
s1
s2
s3

例：自动驾驶

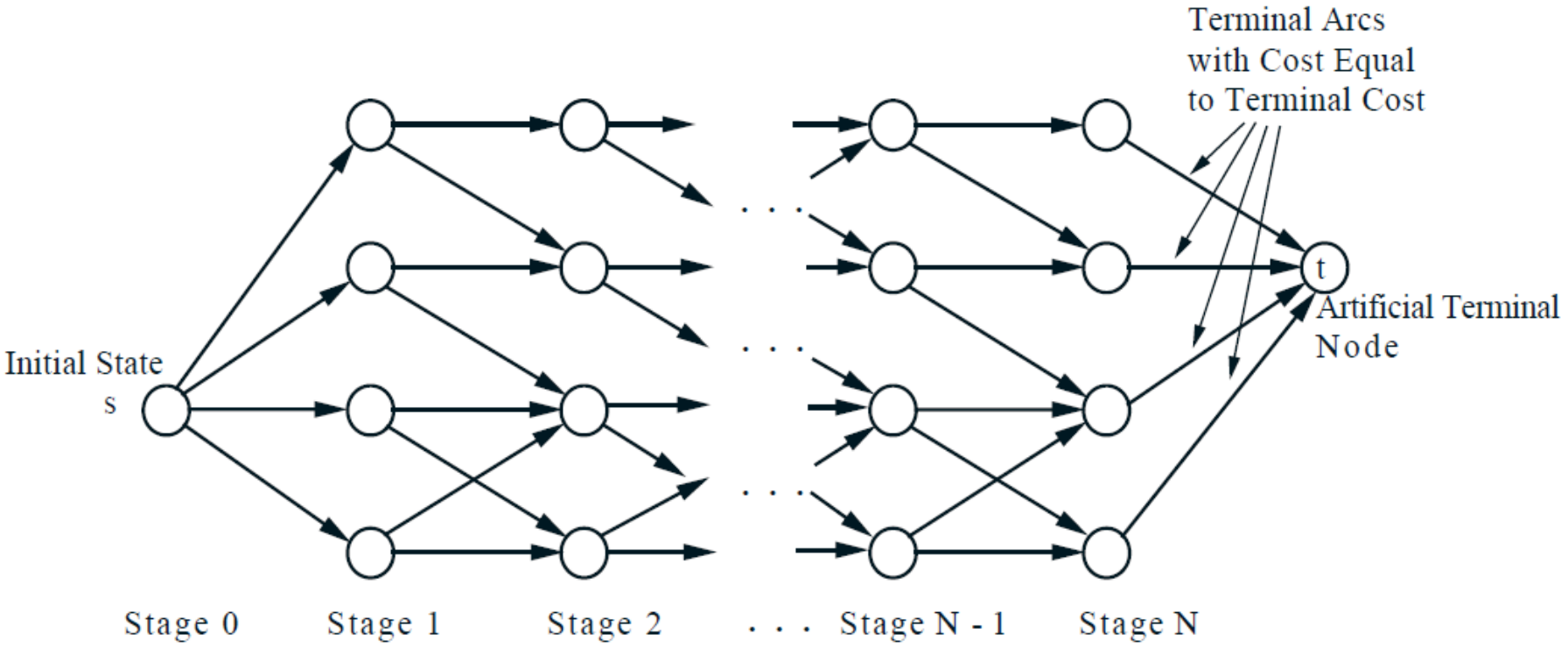
Self-driving Uber Crash, in 2018

两个重要概念

- **无后效性(Markov性)**:某阶段的状态一旦确定，则此后过程的演变不再受此前各种状态及决策的影响，简单的说，就是“未来与过去无关” 。无后效性是一个问题可以用动态规划求解的标志之一。

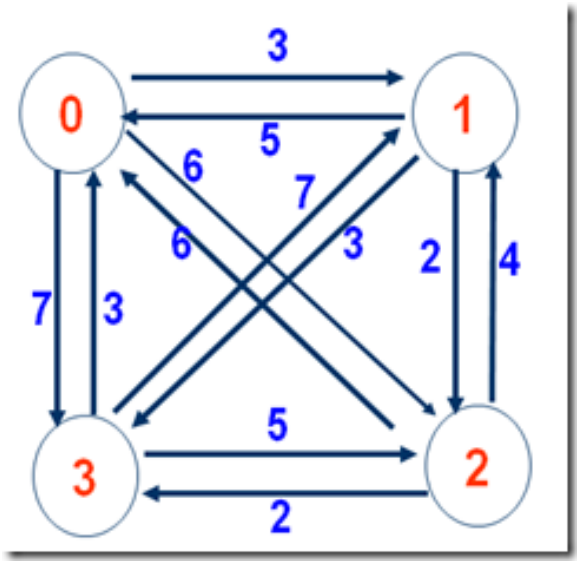
- **状态转移方程**:第k阶段到第$k+1$阶段的状态转移规律，称为状态转移方程或状态转移函数。

状态转移图


Terminal Arcs
with Cost Equal
to Terminal Cost
t
Artificial Terminal
Node
Initial State
s
Stage 0
Stage 1
Stage 2
. . .
Stage N - 1
Stage N

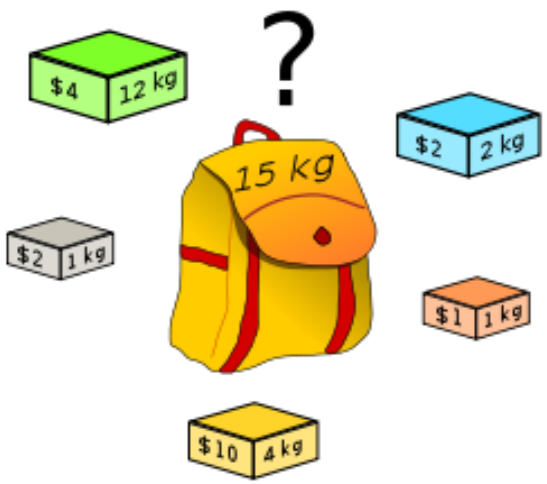
旅行商问题(TSP)

- 有*n*个城市，一个推销员要从其中某一个城市出发，唯一走遍所有的城市，再回到他出发的城市，求最短的路线



0-1背包问题(Knapsack problem)

- 给定 n 种物品，物品 j 的重量为 w_j，价格为 c_j，在限定的总重量 W 内，我们如何选择，才能使得物品的总价格最高。



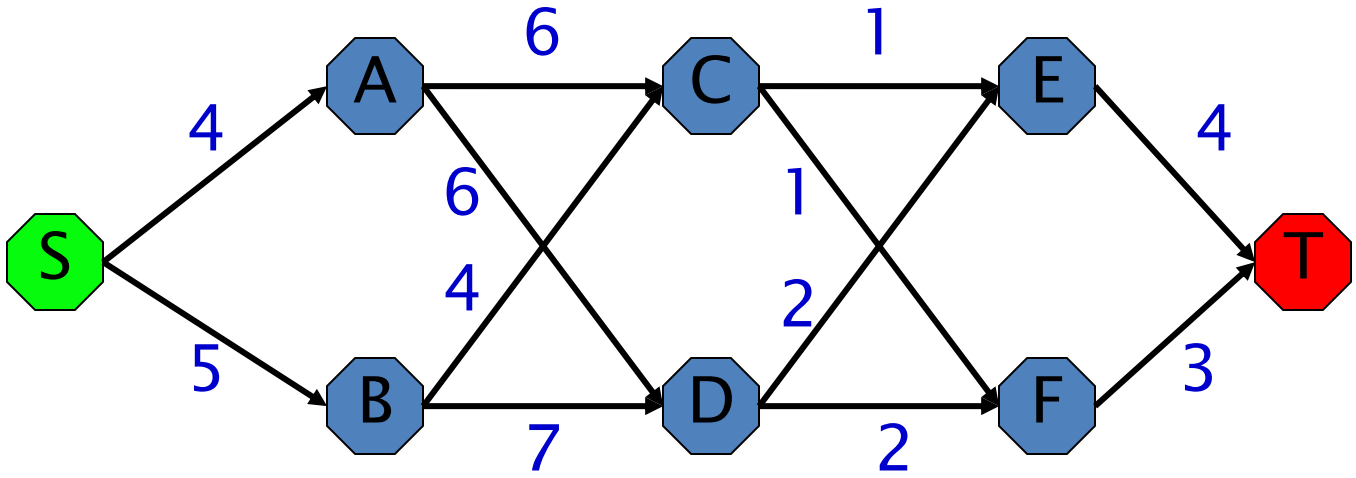
$$\max \quad z = \sum_{j}^{n} c_j x_j$$

$$s.t. \quad \sum_{j=1}^{n} w_j \le W$$

$$x_j \in \{0,1\}. j = 1, \ldots, n$$

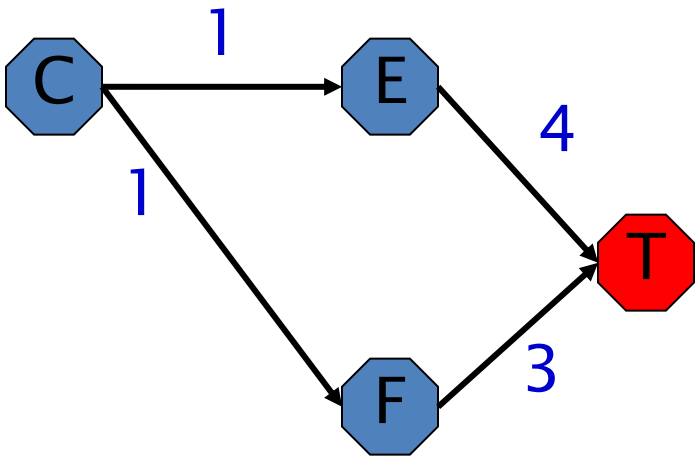
例：最短路径问题

$$\min \quad D(p)$$
$$p \in Path(S,T)$$

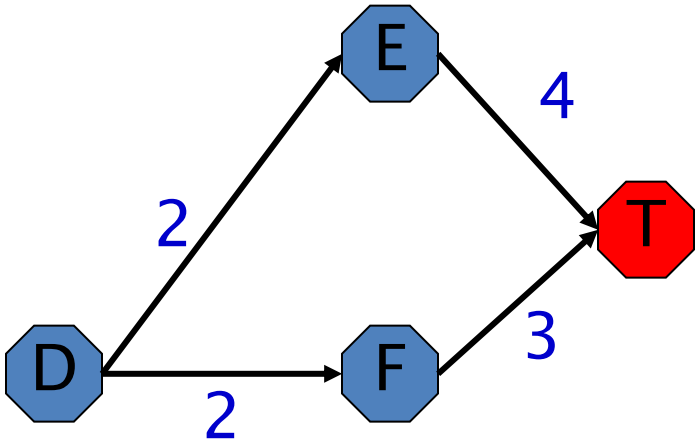


汽车从S站出发，到T站终止，全程分为4段，如何选择路线使S->T时间最短？（图中数字为各段所需时间。）

第三阶段决策：E or F

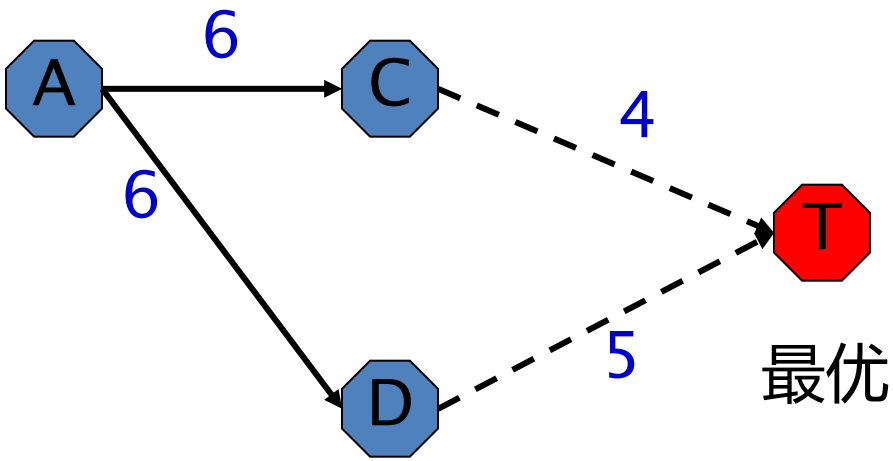


最优：C->F->T (4)

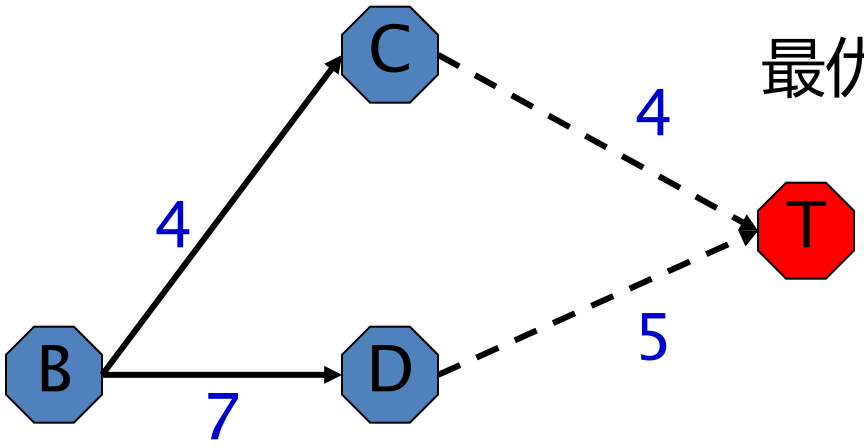


最优：D->F->T (5)

第二阶段决策：C or D

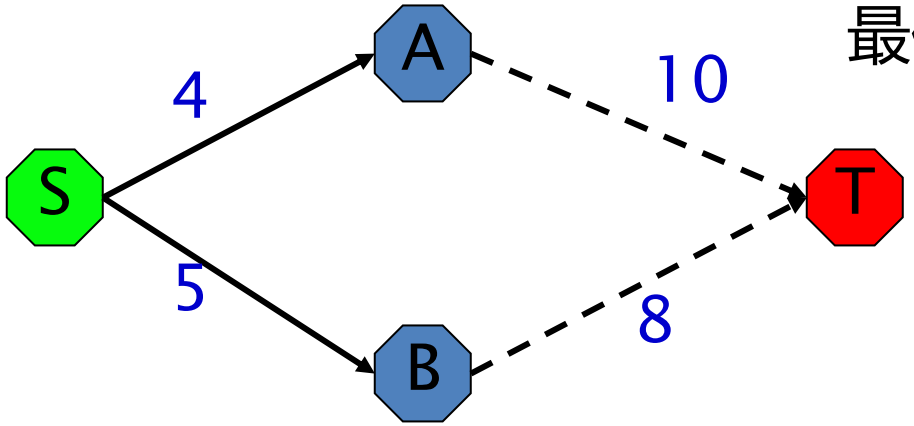


最优：A->C……>T (10)



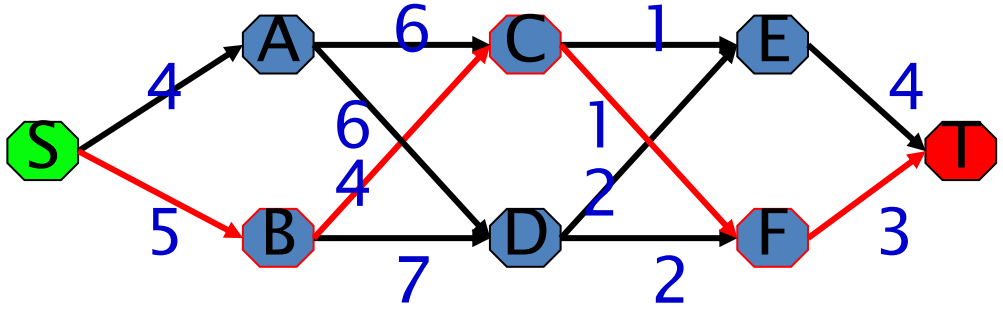
最优：B->C……>T (8)

第一阶段决策：A or B

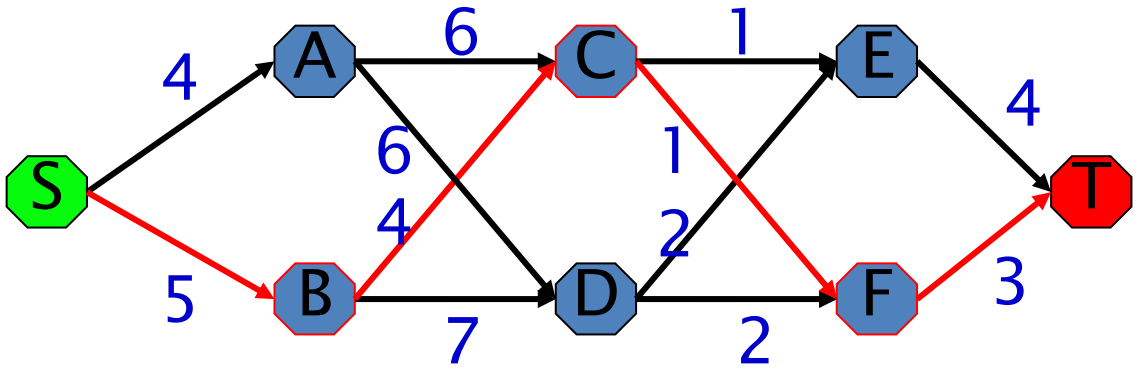


最优：S->B……>T (13)

最优决策：S->B->C->F->T （13）



特点



- 为找到S->T的最优路线，先找出各站到终点T的最优路线。
  - 表面上，多作了计算，但在迭代求解过程中，每一步只需作很简单的计算。
- 最优决策S->B->C->F->T中，从任一点开始到T的任一段，如C->F->T也是从C到T的最优决策。