SJTU-CUMCM2021

运筹优化模型概览

主讲：张嘉乐
IEEE试点班 2018级
tt885.github.io

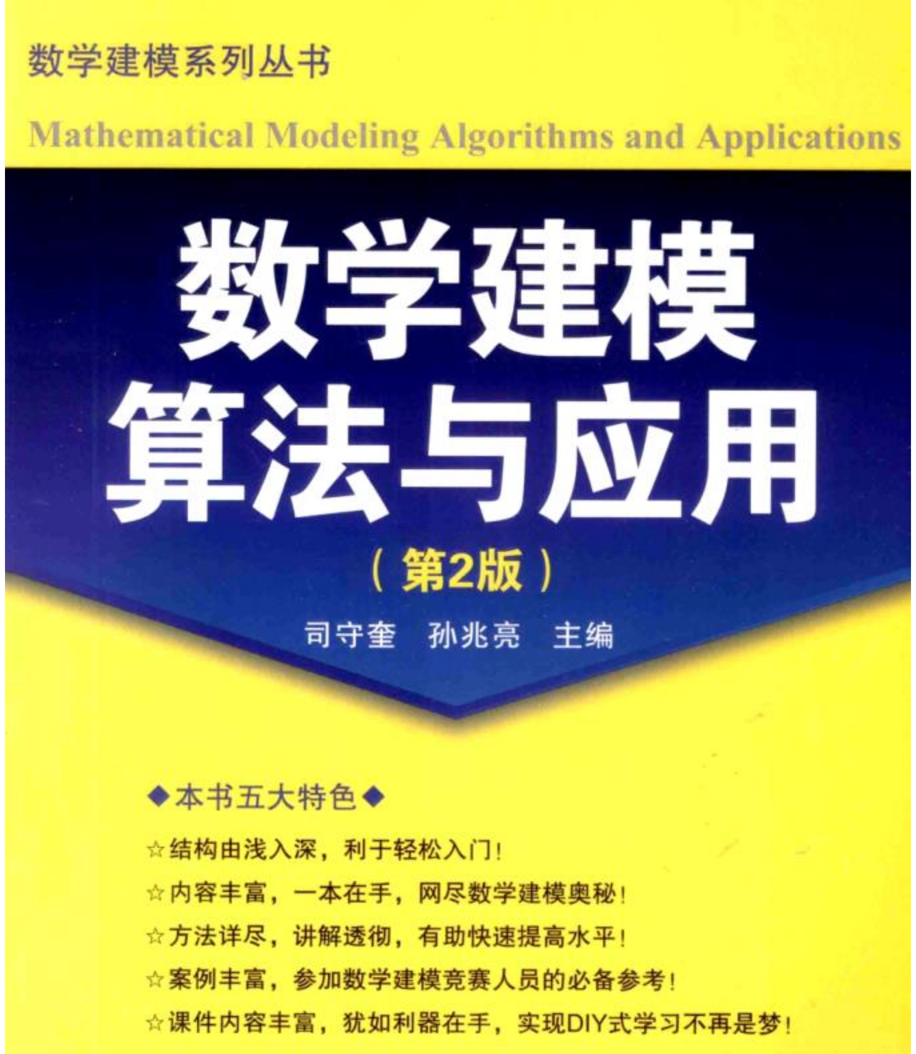
2021.6

十年B题 | 运筹优化，数据分析，物理题

年份	题目	模型/备注	类型
2010	上海世博评价	灰色预测，熵权法	数据分析
2011	交警平台调度	图论最短路，0-1规划	运筹学
2012	太阳能小屋	/	物理题
2013	碎纸片拼接	图像处理算法，匹配	运筹学
2014	平板折叠桌	/	物理题
2015	出租车配置	层次分析，模糊综合评价，经济模型	数据分析
2016	小区道路交通	聚类，元胞自动机，最短路，评价	运筹学+数据分析
2017	拍照赚钱定价	聚类，灰色关联分析	数据分析
2018	RGV调度	智能优化，调度模型，整数规划	运筹学
2019	同心协力	2019年C题是运筹学(排队论)	物理题
2020	穿越沙漠	动态规划，博弈论，统计模型	运筹学

接下来概览运筹优化，数据分析两大类模型

运筹学 | 研究各种系统的最优化问题



运筹学 | 研究各种系统的最优化问题

- 运筹学研究的对象是一类有目的性、可控制的过程或行动
- 凡是安排、调度、筹划、控制等问题均属于运筹学范畴
- 运筹学的目标是找到最优方案
  - 资源分配 —— 规划论
  - 随机聚散 —— 排队论
  - 竞争现象 —— 博弈论
  - 网络优化 —— 图论
  - ……

夫运筹策帷帐之中,决胜于千里之外。

——《史记·高祖本纪》

规划论 | 概述

- 线性规划 Linear Programming
- 整数规划 Integer Programming
- 目标规划 Goal Programming
- 动态规划 Dynamic Programming

……

$$
\begin{aligned}
\max \quad & f(\mathbf{x}) \\
s.t. \quad & g_i(\mathbf{x}) \le 0, \quad i = 1, 2, \cdots, m \\
& h_i(\mathbf{x}) = 0. \quad i = m+1, m+2, \cdots, n
\end{aligned}
$$

- 例题：生产两种产品，求最大产值

资源\产品	甲	乙	可用资源限制
材料1	1	0	4
材料2	0	2	12
材料3	3	2	18
售价	\$3	\$5	

生产甲、乙分别为x_1, x_2个

$$
\begin{aligned}
\max f(x) &= 3x_1 + 5x_2 \\
s.t.\quad x_1 &\le 4 \\
2x_2 &\le 12 \\
3x_1 + 2x_2 &\le 18 \\
x_1, x_2 &\ge 0
\end{aligned}
$$

解得 $x_1 = 2, x_2 = 6, f(x) = 36$

- 一般形式

$$
\begin{aligned}
\max \quad & \sum_{j=1}^{n} c_j x_j \\
s.t. \quad & \sum_{j=1}^{n} a_{ij} x_j \le b_i, \quad i = 1, 2, \cdots, m \\
& x_j \ge 0. \quad j = 1, 2, \cdots, n
\end{aligned}
\qquad
\begin{aligned}
\max \quad & \mathbf{c}^T \mathbf{x} \\
s.t. \quad & \mathbf{A}\mathbf{x} \le \mathbf{b}, \\
& \mathbf{x} \ge \mathbf{0}.
\end{aligned}
$$

```
% max  f(x)=3x1+5x2
% s.t.  x1         <=4
%              2x2 <=12
%         3x1 + 2x2 <=18
%          x1        >=0
%               x2 >=0
f=[3 5];
A=[1 0;0 2;3 2];
b=[4;12;18];
Aeq=[];beq=[];
lb=[0 0]';ub=[];
[x val]=linprog(-f,A,b,Aeq,beq,lb,ub)
```

$$\max \quad \mathbf{c}^T\mathbf{x}$$
$$s.t. \quad \mathbf{A}\mathbf{x} \le \mathbf{b}, \quad \mathbf{x} \ge \mathbf{0}.$$

结果x =[2.0000, 6.0000]
val = -36

```
% 松弛后问题如下
% max  f(x)=3x1+5x2
% s.t.  x1         + x3            =4
%            2x2        + x4       =12
%       3x1 + 2x2            + x5 =18
%        x1,x2,x3,x4,x5 >=0
f2=[3 5 0 0 0];
Aeq=[1 0 1 0 0;0 2 0 1 0;3 2 0 0 1];
beq=[4 12 18]';
lb=zeros(1,5);
[x2 val2]=linprog(-f2,[],[],Aeq,beq,lb,ub)
```

x2 = [2.0000, 6.0000, 2.0000, 0, 0]
val2 = -36

- 决策变量、可行解、约束条件
- **标准形式**及矩阵表示
- 对偶理论(了解即可)
  - **对偶形式**及矩阵表示
  - 弱对偶定理、强对偶定理
- 求解算法(不用掌握)：单纯形法 对偶单纯形法、椭球算法、内点法
- 应用：工业生产，交通运输，经济决策，政治选举 ……

$$
\begin{aligned}
\max \quad & \sum_{j=1}^{n} c_j x_j \\
s.t. \quad & \sum_{j=1}^{n} a_{ij} x_j \le b_i, \quad i = 1, 2, \cdots, m \\
& x_j \ge 0. \quad j = 1, 2, \cdots, n
\end{aligned}
\qquad
\begin{aligned}
\max \quad & \mathbf{c}^T \mathbf{x} \\
s.t. \quad & \mathbf{A}\mathbf{x} \le \mathbf{b}, \\
& \mathbf{x} \ge \mathbf{0}.
\end{aligned}
$$

$$
\begin{aligned}
\min \quad & \sum_{i=1}^{m} b_i y_i \\
s.t. \quad & \sum_{i=1}^{m} a_{ij} y_i \ge c_j, \quad \forall j \\
& y_i \ge 0. \quad \forall i
\end{aligned}
\qquad
\begin{aligned}
\min \quad & \mathbf{y}^T \mathbf{b} \\
s.t. \quad & \mathbf{y}^T \mathbf{A} \ge \mathbf{c}^T, \\
& \mathbf{y} \ge \mathbf{0}.
\end{aligned}
$$

- 和线性规划类似，决策变量取值为整数
  - 纯整数线性规划
  - 混合整数线性规划
  - 0 - 1线性规划 背包问题、指派问题…

$$\max z = \sum_{j=1}^{n} c_j x_j$$

$$\begin{cases} \sum_{j=1}^{n} a_{ij} x_j \le b_i & (i = 1,2,\cdots,m) \\ x_j \ge 0 & (j = 1,2,\cdots,n) \\ x_1, x_2, \cdots, x_n \text{部分或全部取整数} \end{cases}$$

例题https://acm.sjtu.edu.cn/OnlineJudge/problem/3030

背包问题
有n个物品，第i个重 w_i,价值v_i,选总重不超过W的物品使总价最大


Online Judge 首页 题库 比赛 作业 评测状态 用户名

3030. mushroom

【问题描述】
从前有座山，山上有座庙，庙里有个老和尚，老和尚让小和尚采蘑菇... “采蘑菇是一门博大精深的学问”老和尚如是说。 采摘不同的蘑菇，需要不同的时间；采集不同的蘑菇，有不同的价值。 小和尚苦恼地揉了揉眉心，他想要在规定地时间中，采集价值最大地蘑菇，却苦无良策， 只能打电话询问正在学习编程的你，期望你能帮他解决这个问题。

【输入文件】
输入的第一行有两个整数T（1 <= T <= 1000）和M（1 <= M <= 100），用一个空格隔开，T代表总共能够用来采蘑菇的时间，M代表山里的蘑菇的数目。接下来的M行每行包括两个在1到100之间（包括1和100）的整数，分别表示采摘某蘑菇的时间和蘑菇的价值。

【输出文件】
输出在规定的时间内，可以采到的蘑菇的最大总价值。

【样例输入】
```
100 4
99 100
101 1
97 3
2 3
```

【样例输出】
```
100
```

解：第i个物品选x_i个(整数)

$$\max f(x)= x_i\ v_i$$

$$\text{s.t.}\ \sum_{i=1}^{n} x_i\ w_i \le W$$

```
N=4;W=100;
f2=[100 1 3 3]; %v_i
A2= [99 101 97 2];  %w_i
b2=[W]
Aeq=[];beq=[];
lb=zeros(N,1)';
ub=ones(N,1);
intcon=[1 2 3 4];
[x,value]= intlinprog(-f2,intcon,A2,b2,
                      Aeq,beq,lb,ub);
```

输出x =[1 0 0 0], value =-100

- 决策变量取值为整数 （NP难，而线性规划是P问题）
- 整数**线性**规划 Integer Linear Programming
    - 纯整数线性规划
    - 混合整数线性规划
    - 0-1线性规划 背包问题、指派问题、旅行商问题…

•求解算法(不用掌握)：穷举法、分支定界法、割平面法
匈牙利法(适用于指派问题)、近似算法、概率算法……

- 是一种建模方式，并非新的数学形式
- 用于多目标优化问题，允许不达标

•例题：

资源\产品	甲	乙	可用资源限制
钢铁	2	3	100
工时	4	2	120
单位利润	6	4	

双目标 P_1：利润超过280。

P_2 :钢材，工时不超100、120，

其**权重比**5:1

生产甲,乙分别为x_1, x_2个

$$\min f(x) = P_1 d_1^- + P_2(5d_2^+ + d_3^+)$$

$$\text{s.t. } \begin{aligned} 6x_1 + 4x_2 + d_1^- - d_1^+ &= 280 \\ 2x_1 + 3x_2 + d_2^- - d_2^+ &= 100 \\ 4x_1 + 2x_2 + d_3^- - d_3^+ &= 120 \\ x_1, x_2, d_i^-, d_i^+ &\geq 0 (i = 1,2,3) \end{aligned}$$

引入了 d_i^-, d_i^+ ,P_1, P_2， 5:1

- 多目标决策，线性规划
- 偏差变量 (盈亏) d^+ d^-
- 系统约束、目标约束
- 优先因子 p
- 权系数 ω
- **线性**目标规划
  - 图解法、单纯形法
  - ……

$$\min \quad \sum_{l=1}^{L} p_l \sum_{k=1}^{K} (\omega_{lk}^- d_k^- + \omega_{lk}^+ d_k^+)$$

$$\varphi_i(x_1, x_2, \cdots, x_n) \le g_i \, (i = 1,2,\cdots,m)$$

$$f_i + d_i^- - d_i^+ = f_i^* \, (i = 1,2,\cdots,K)$$

规划论 | 动态规划

- 是一种求解方法
- 需要问题具有2个性质：

  ① 重叠子问题

  ② 最优子结构
- 动态规划算法构成

  最优值函数dp[][][]、状态转移方程、边界条件

•玩具例子：计算斐波那契数列第k项，

$$f[k]=f[k-1]+f[k-2]$$

•基础例子:数字三角形最大和、背包问题、旅行商问题

•可求解问题非常多……

例子：斐波那契数列

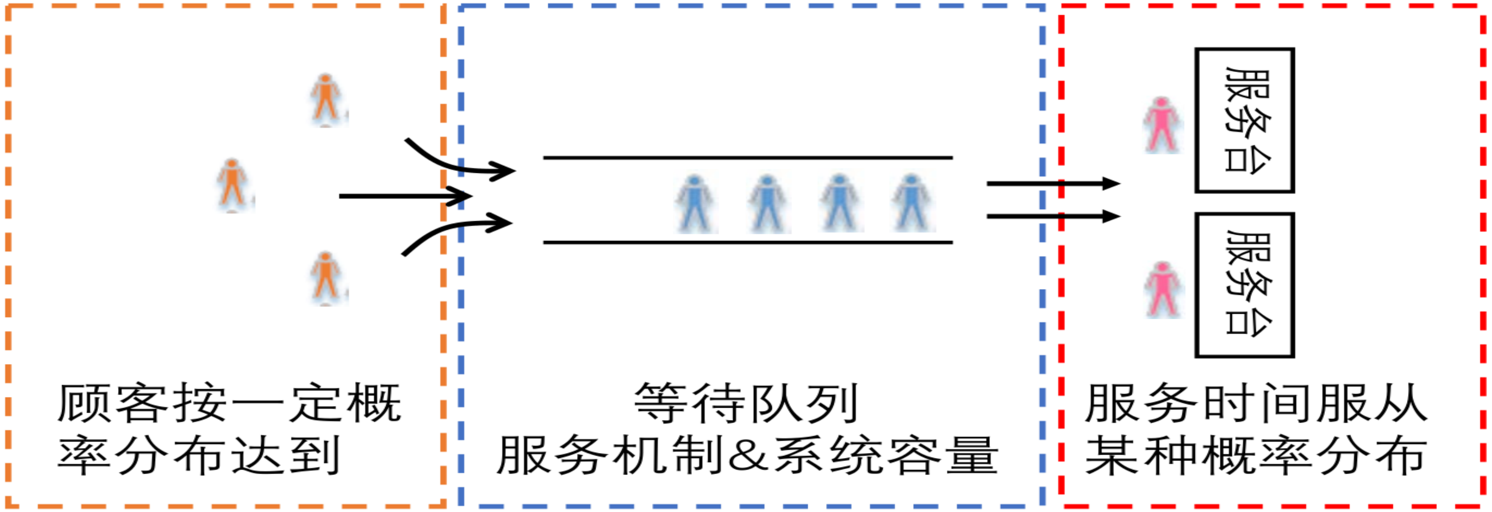
```
def fib1(k):
    if k<=2:
        return 1;
    return fib1(k-1)+fib1(k-2);

def fib_dp(k):
    dp=[1]*k;
    for i in range(2,k):
        dp[i]=dp[i-1]+dp[i-2];
    return dp[k-1];

print(fib_dp(35));
print(fib1(35));
```

排队论 | 概述

•排队论是研究排队系统的数学理论方法，丹麦**电话工程师**埃尔朗为解决自动电话设计在1909年开始形成的一套理论

•可以排队的东西： 生产线上原材料、半成品； 等待修理的机器、船食堂吃饭、车站等车、医院挂号；打电话、12306网上买票……

排队系统：



排队论 | 各种指标

•例题： 有一个银行柜台，平均每分钟到达λ个顾客，服务μ个顾客($\mu \geq \lambda$)，均为泊松流(每分钟有λ概率来一个顾客，有μ概率完成一个顾客服务)，求：

1） 平均队长 $L = \dfrac{\lambda}{\mu - \lambda}$

2) 平均排队长 $L_q = \dfrac{\lambda^2}{\mu(\mu - \lambda)}$

3） 平均逗留时间 $W = \dfrac{1}{\mu - \lambda}$

4) 平均等待时间 $W_q = \dfrac{\lambda}{\mu(\mu - \lambda)}$

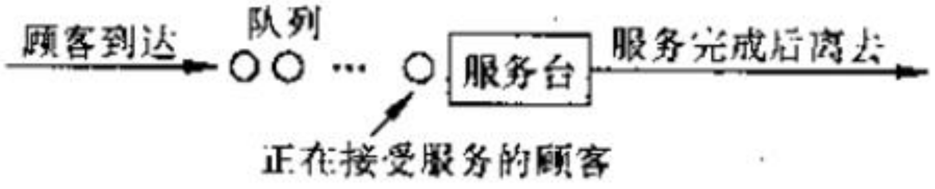


图 10-1 单服务台排队系统

排队系统Litter 公式: $L = \lambda W, L_q = \lambda W_q$

•这种排队设定称为M/M/1/∞ 排队系统

排队论肯德尔记号

一般的排队模型可以表示为模板“X/Y/Z/A/B/C”：

X—顾客相继到达的间隔时间的分布；

Y—服务时间的分布；

（X和Y的取值可以为：M-指数分布，G-一般分布）

Z—服务台个数；

A—系统容量限制（默认为∞）；

B—顾客源数目（默认∞）；

C—服务规则（默认为FCFS”First Come, First Service”）。

经典排队论类型：M/M/1,M/M/K,M/D/1……

排队论 | 多种多样的排队系统

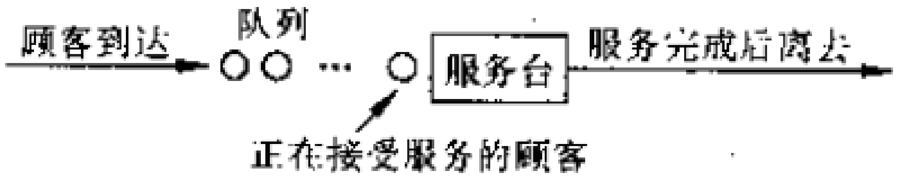


图 10-1 单服务台排队系统

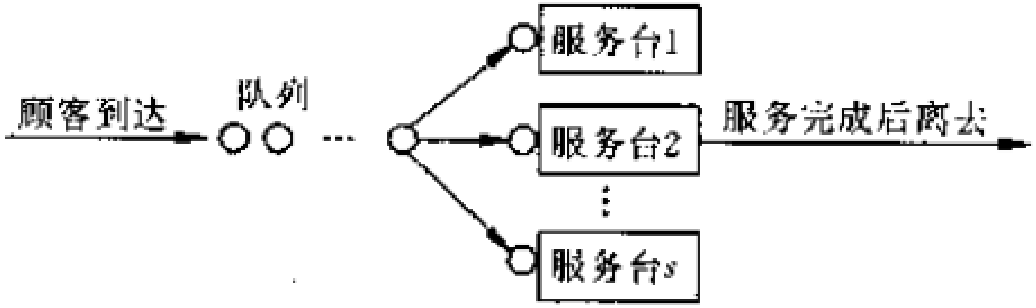


图 10-2 s 个服务台，一个队列的排队系统

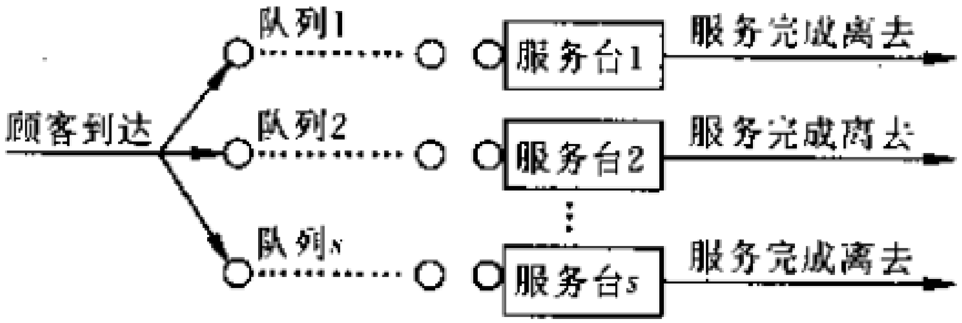


图 10-3 s 个服务台，s 个队列的排队系统

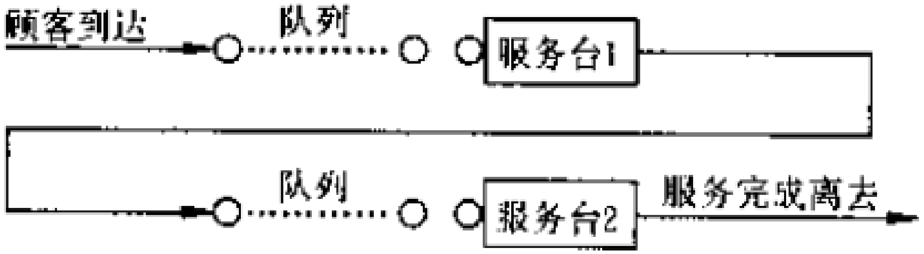


图 10-4 多个服务台的串联排队系统

怎么办： 写程序模拟

排队论 | 小结

- 问题背景　超市排队、电话占线、打出租车排队……
- 排队系统三个组成部分：输入过程、排队规则、服务机制
  - 1.输入过程：　顾客源数量，到达间隔，单个还是成批
  - 2.排队规则：　允许队列最大长度

    先到先服务 (FCFS)、后到先服务 (LCFS)、优先队列
  - 3.服务机制：服务时间的概率分布

    服务台数量与结构
- 四个衡量指标　平均队长、平均排队长，平均逗留时间、平均等待时间

博弈论 | 概述

- 又称对策论、竞赛论。研究具有竞争、对抗、利益分配的问题。
- 理论最早1921年由博雷尔提出，在我国相关思想可追溯到孙子兵法

•例题： 1943年2月，日军舰队策划了一次军事行动，盟军安排空中打击。

•路线分南、北两线，时间3天。南线晴朗，北线阴雨，盟军飞机部署在南线。

- •局势1：舰队走北线，侦察机先搜北线，可以炸2天
- •局势2：舰队走南线，侦察机先搜北线，可以炸2天
- •局势3：舰队走北线，侦察机先搜南线，可以炸1天
- •局势4：舰队走南线，侦察机先搜南线，可以炸3天

博弈论 | 概述

- 局势1：舰队走北线，侦察机先搜北线，可以炸2天
- 局势2：舰队走南线，侦察机先搜北线，可以炸2天
- 局势3：舰队走北线，侦察机先搜南线，可以炸1天
- 局势4：舰队走南线，侦察机先搜南线，可以炸3天

飞机 \ 舰队	**走北线**	**走南线**	Min
先搜北线	2	2	2*(飞机策略)
先搜南线	1	3	1
max	2*(舰队策略)	3	min max =max min=2

- 史实就是局势1，因为有**鞍点**

- 博弈三要素
  - 玩家 (player)
  - 策略 (strategy)
  - 赢得矩阵 (winning matrix)
- 分类
  - **纯策略**、混合策略
  - **完全信息**、不完全信息
  - **静态**、动态
  - **零和**、非零和
  - 合作、**非合作**

1P\2P	**石头**	**剪刀**	**布**
石头	0	1	-1
剪刀	-1	0	1
布	1	-1	0

- 没有**鞍点**,由于min max> max min，只有混合策略
- 纳什均衡解（$\frac{1}{3}, \frac{1}{3}, \frac{1}{3}$），等概率地采用3种手势
    - 1P采取此策略期望收益最高
    - 2P采取此策略期望收益最低(也就是对2P越有利)

- 博弈三要素
    - 玩家 (player)
    - 策略 (strategy)
    - 赢得矩阵 (winning matrix)
- 分类
    - 纯策略、**混合策略**
    - **完全信息**、不完全信息
    - **静态**、动态
    - **零和**、非零和
    - 合作、**非合作**

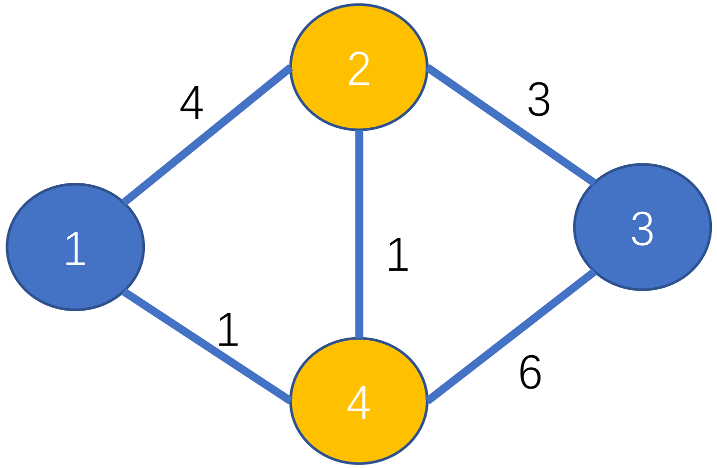
博弈论 | 小结

- 问题背景：竞争现象 棋牌、体育竞技、商业竞争……
- 博弈三要素 玩家 (player)、策略 (strategy)、赢得矩阵 (winning matrix)
- 基本假设：玩家是**理性**的，每个玩家都想尽可能取得最大收益
- 分类: 纯策略、混合策略;
  - 完全信息、不完全信息；
  - 静态、动态
  - ……
- eg：二人零和博弈 —— 田忌赛马 · 续
  - 纳什均衡 (Nash Equilibrium)
    - ➢ 所有玩家都不愿意主动改变现状的局势

齐王 \ 田忌	ABC	ACB	BAC	BCA	CAB	CBA
ABC	3	1	1	1	-1	1
ACB	1	3	1	1	1	-1
BAC	1	-1	3	1	1	1
BCA	-1	1	1	3	1	1
CAB	1	1	1	-1	3	1
CBA	1	1	-1	1	1	3

图论 | 概述

例：



G=(V,E)
点集 V={1,2,3},
边集E= {(1,2,4), (1,4,1), (2,4,1), (2,3,3), (3,4,6)}

邻接矩阵W

$$\begin{matrix} 0 & 4 & 0 & 1 \\ 4 & 0 & 3 & 1 \\ 0 & 3 & 0 & 6 \\ 1 & 1 & 6 & 0 \end{matrix}$$

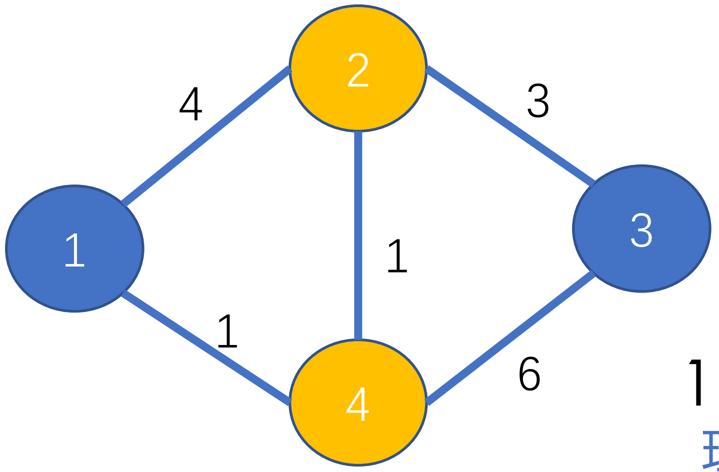
几种含义的理解

1. 点表示地点，边权重表示路线长度
   - 最短路，最小生成树，最短哈密顿回路(TSP)
2. 点表示地点，边权表示水管容量：网络流
   - 最小费用流，最小割，匹配问题
3. 蓝点表示工人，黄点表示工作(去掉2-4连边)
   - 最大匹配，稳定匹配
4. 点表示工作，边权表示完成时间
   - 拓扑排序，统筹安排
5. 点表示人，边权表示合作次数
   - 社交关系分析聚类

……

1.点表示地点，边权重表示路线长度

例：



G=(V,E)
点集 V={1,2,3},
边集E= {(1,2,4), (1,4,1), (2,4,1), (2,3,3), (3,4,6)}

1.1最短路

理论上我们要：Dijkstra、Bellman-Ford (SPFA)

实际上：`graphallshortestpaths(W);`

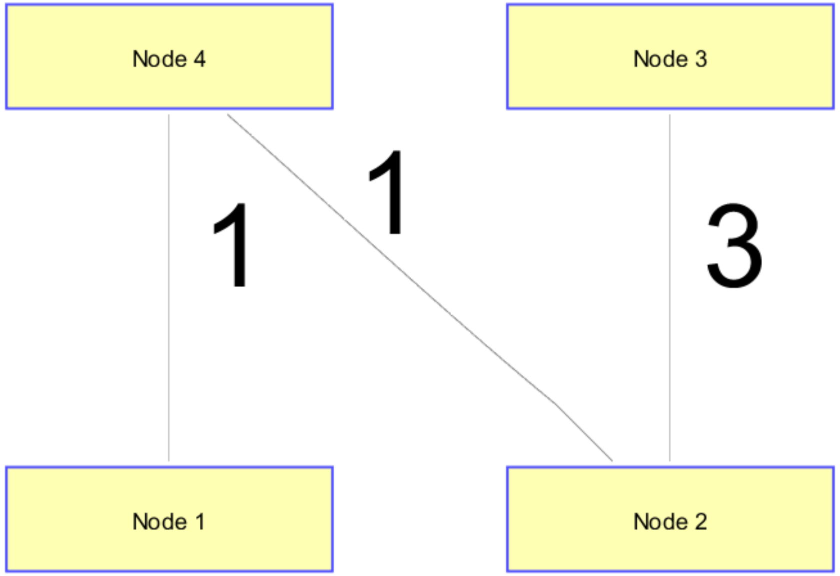
要路径的话： `graphshortestpath(W,1,3);`

$$\begin{matrix} 0 & 2 & 5 & 1 \\ 2 & 0 & 3 & 1 \\ 5 & 3 & 0 & 4 \\ 1 & 1 & 4 & 0 \end{matrix}$$

1.2最小生成树

理论上我们要： Kruskal、Prim

实际上： `graphminspantree(W);`



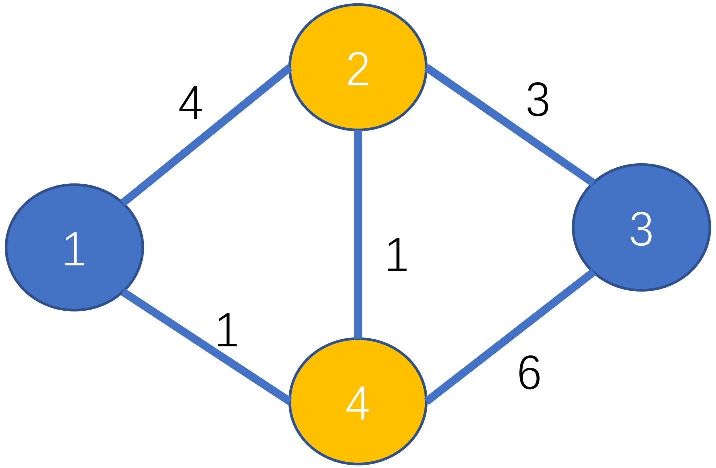
1.3 TSP 问题

小规模可以动态规划

大规模用最小生成树得近似解

2.点表示地点，边权表示水管容量：网络流

例:



G=(V,E)
点集 V={1,2,3},
边集E= {(1,2,4), (1,4,1), (2,4,1), (2,3,3), (3,4,6)}

理论上我们要： Ford-Fulkerson, Edmonds-Karp , Dinic

实际上： `graphmaxflow(W, 1, 3);` 或 线性规划

最大流为5:

FlowMatrix =

(1,2)	4.0000
(2,3)	3.0000
(4,3)	2.0000
(1,4)	1.0000
(2,4)	1.0000

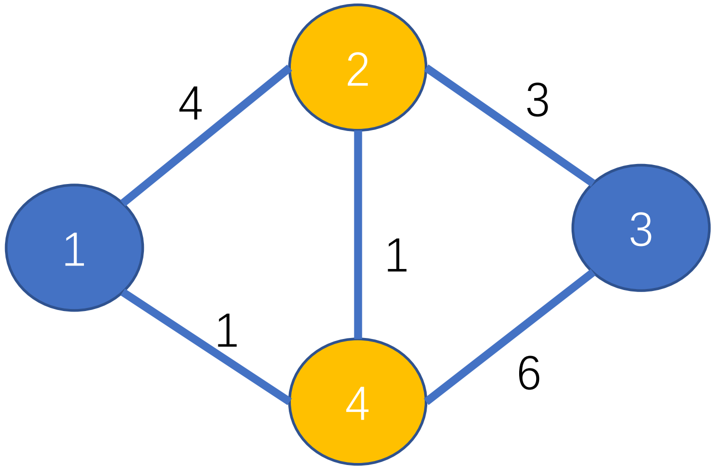
设边流量分别为$x_{12}, x_{14}, x_{24}, x_{23}, x_{43}$

$$\max f(x) = x_{23} + x_{43}$$
$$\text{s.t.}\quad x_{12} - x_{24} - x_{23} = 0$$
$$x_{14} + x_{24} - x_{43} = 0$$
$$x_{ij} \le cap_{ij} (i = 1,2,3,4)$$

网络流模型延伸(均可规划)
- 匹配问题
- 最小费用流
- 最小割问题

最短路、最小生成树、网络流MATLAB代码

例:

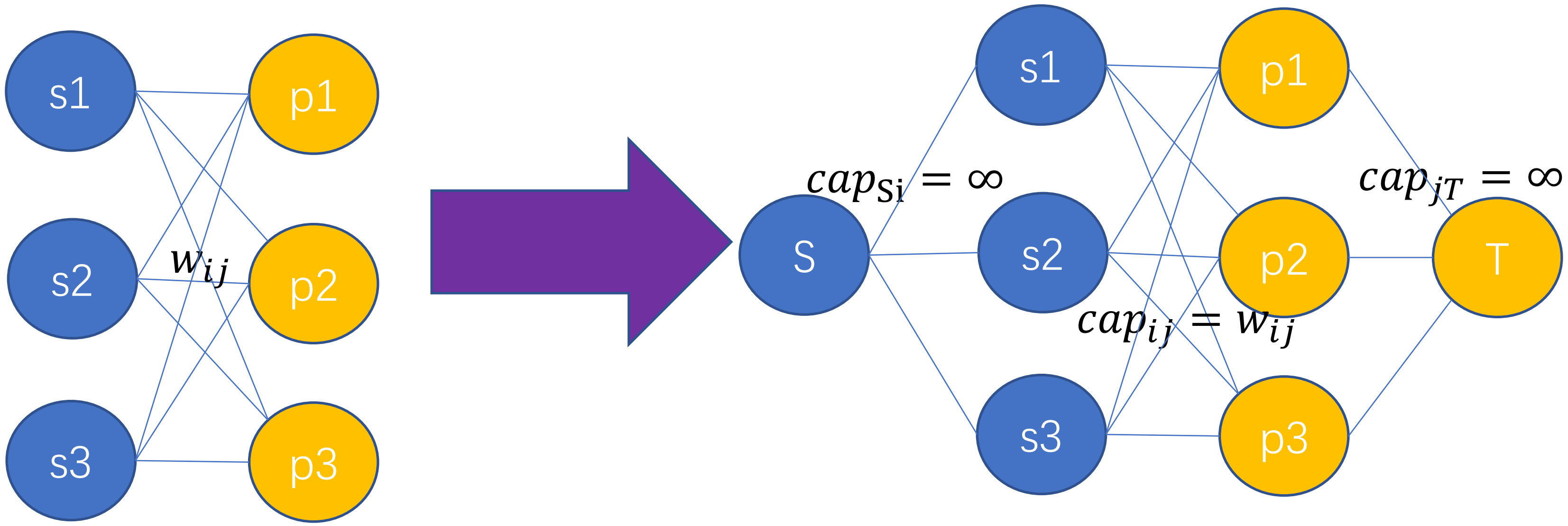


G=(V,E)
点集 V={1,2,3},
边集E= {(1,2,4), (1,4,1), (2,4,1), (2,3,3), (3,4,6)}

```
vi=[1,1,2,2,4];
vj=[2,4,4,3,3];
weight=[4,1,1,3,6];
W=sparse (vi,vj,weight,4,4);
graph= full(W);%邻接矩阵
[MaxFlow, FlowMatrix, Cut] = graphmaxflow(W, 1, 3);
%最大流
W=W+W‘;
[dist,path,pred]=graphshortestpath(W,1,3); %单源最短路
dist2= graphallshortestpaths(W); %多源最短路
[Tree, pred] = graphminspantree(W); %最小生成树
view(biograph(Tree,[],'ShowArrows','off','ShowWeight
s','on','EdgeFontSize',45))
```

3. 蓝点表示工人，黄点表示工作
权重表示匹配收益，求收益总和最大的匹配

用网络流解最大权匹配



$$cap_{ij} = w_{ij}$$
$$cap_{Si} = cap_{jT} = \infty$$

图论　| 小结

- 图的表示法　邻接矩阵、邻接表
- 最短路径算法
  - 单源　Dijkstra（正权图）、Bellman-Ford (SPFA)
  - 多源　Floyd-Warshall、Johnson’s
- 最小生成树　Kruskal、Prim
- 网络最大流　Ford-Fulkerson
- 图搜索算法　DFS、BFS
- 统筹安排　拓扑排序，关键路径
- 欧拉回路、中国邮路问题、旅行商问题、图着色问题……

其他运筹学问题与方法

- 其他规划：
  - 非线性规划　Nonlinear Programming　(二次规划、凸规划……)
  - 参数规划　Parametric Programming
  - 随机规划　Stochastic Programming
  - ……
- 存储论 Inventory Theory　最小化储存成本、防止缺货和积压
- 决策论 Decision Theory　方案择优方法
- 搜索论 Search Theory　在状态空间中找最优解

  启发式搜索　模拟退火、蚁群算法、遗传算法、粒子群优化……
- 模拟　Simulation　实际比赛中很常用
- ……

运筹学 | 总结

- 资源分配 —— 规划论
    - 线性规划；整数规划；目标规划；动态规划
- 随机聚散 —— 排队论
    - 排队系统；平均队长、平均时间等指标；程序模拟
- 竞争现象 —— 博弈论
    - 赢得矩阵；纯/混合策略、静/动态等概念；纳什均衡
- 网络优化 —— 图论
    - 图的建模与表示；最短路；最小生成树；网络流，匹配

- 2020国赛B题，穿越沙漠多人游戏：图论+动态规划，博弈论
- 2019国赛C题，机场出租车：排队论
- 2018国赛B题，智能RGV调度：蒙特卡洛模拟，搜索
- 2021美赛B题，无人机救火： 图论覆盖
- 2019美赛D题，卢浮宫逃生：图论网络流，排队论
- 2017美赛D题，机场安检： 排队论