从“穿越沙漠”看离散优化问题的求解

浙 江 大 学 数 学 科 学 学 院
浙江大学数学建模实践基地
谈之奕
tanzy@zju.edu.cn

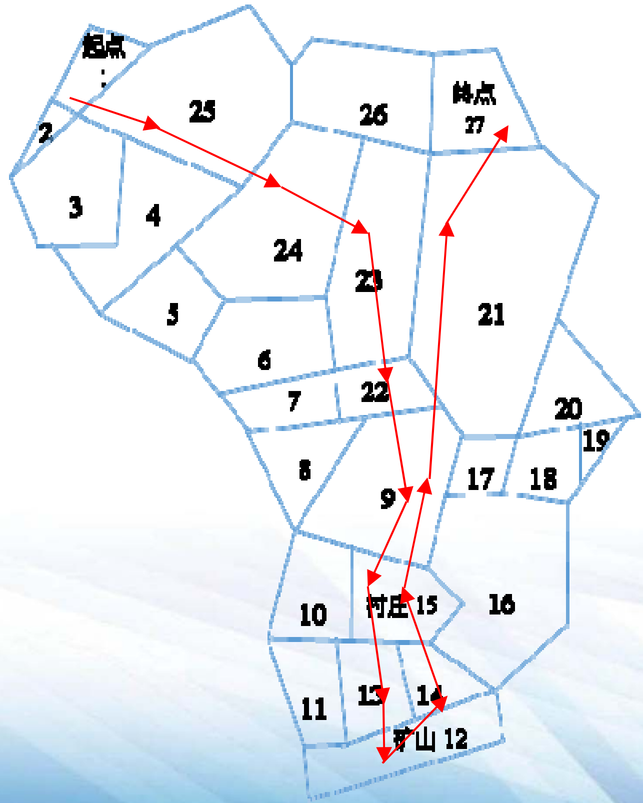
穿越沙漠

- 穿越沙漠
  - 玩家凭借一张地图，从起点出发，在沙漠中行走，途中会遇到不同的天气，可能经过矿山、村庄。目标是在规定时间内到达终点，并保留尽可能多的资金
  - 玩家在起点处获得初始资金。在矿山停留时，可通过挖矿补充资金
  - 玩家每天行动需水和食物两种资源，可以在起点或村庄以不同价格购买。每天玩家需有足够的水和食物以供行动时消耗，但拥有的水和食物质量之和不能超过负重上限
  - 玩家每天可从地图中的某个区域到达与之相邻的另一个区域，也可在原地停留。不同天气、玩家不同行动消耗的水和食物数量也有所不同
  - 针对玩家数量和未来天气情况是否已知的不同设定，给出相应的玩家策略

穿越沙漠

负重上限		1200千克	初始资金		10000元
截止日期		第30天	基础收益		1000元
资源	每箱质量（千克）	基准价格（元/箱）	基础消耗量（箱）		
			晴朗	高温	沙暴
水	3	5	5	8	10
食物	2	10	7	6	10

日期	1	2	3	4	5	6	7	8	9	10
天气	高温	高温	晴朗	沙暴	晴朗	高温	沙暴	晴朗	高温	高温
日期	11	12	13	14	15	16	17	18	19	20
天气	沙暴	高温	晴朗	高温	高温	高温	沙暴	沙暴	高温	高温
日期	21	22	23	24	25	26	27	28	29	30
天气	晴朗	晴朗	高温	晴朗	沙暴	高温	晴朗	晴朗	高温	高温



主要问题

- 主要问题
  - 对离散优化的基本理论和基本方法掌握不够
  - 求解和计算机应用能力缺乏，难以走完解决问题的“最后一公里”
  - 局限于具体实例的求解，缺少一般情况的模型或算法
  - 模型、算法、结果三者割裂，模型堆砌，逻辑混乱

全国数模竞赛中的离散优化问题

- 优化问题
  - （2020B）穿越沙漠
  - （2018B）智能RGV的动态调度策略
  - （2018D）汽车总装线的配置问题
  - （2017D）巡检线路的排班
  - （2014D）储药柜的设计
  - （2011B）交巡警服务平台的设置与调度
- 部分涉及优化
  - （2017B）“拍照赚钱”的任务定价
  - （2016D）风电场运行状况分析及优化
  - （2013B）碎纸片的拼接复原
  - （2012B）太阳能小屋的设计

组合优化

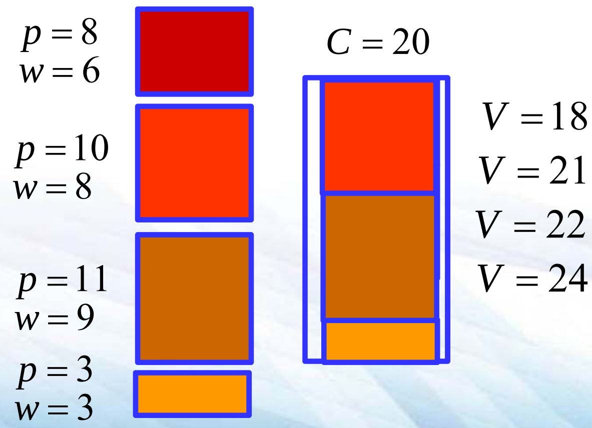
- 组合优化（Combinatorial Optimization）
  - 应用于离散对象的，从有限多个可行解中找出使某个目标函数达到最优的解的优化问题
- 组合优化是离散数学与最优化的交叉学科分支
  - 连续优化（**Continuous Optimization**）与离散优化（**Discrete Optimization**）
    - 连续优化：决策变量为连续变量的优化问题
    - 离散优化 = 组合优化 + 整数规划（+......）

- 组合优化问题
  - 旅行售货商问题（TSP）
  - 最小生成树（MST）
  - 最短路
  - 背包问题
  - 指派问题
  - 装箱问题
  - 车辆路径问题（VRP）

离散与连续

- 模型
  - 现有 n 件物品，物品 j 的价值为 p_j ，大小为 w_j 。物品质地均匀，可任意切割
  - 将若干物品的全部或部分放入容量为 C 的背包中，在放入背包物品大小之和不超过背包容量前提下，使放入背包物品价值之和尽可能大
- 求解思路
  - 称 $\frac{p_j}{w_j}$ 为物品 j 的价值密度
  - 将物品按价值密度从大到小的顺序排列，优先放入价值密度大的物品，直至第一个不能完整放入背包的物品。将该物品的一部分放入背包，使得背包没有剩余空间

$$
\begin{aligned}
\max \quad & \sum_{j=1}^{n} p_j x_j \\
s.t. \quad & \sum_{j=1}^{n} w_j x_j \le C \\
& 0 \le x_j \le 1,\ j = 1, \cdots, n
\end{aligned}
$$

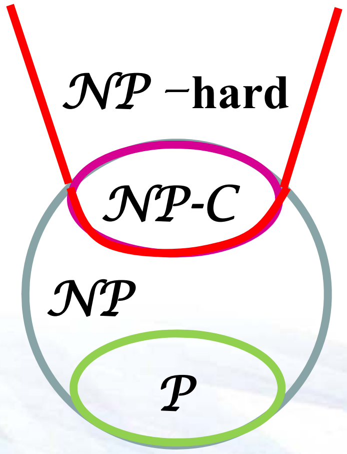


组合优化问题的求解

- 穷举
  - 组合优化问题通常不能通过穷举所有可能的解加以比较来求解，因为可行解的数目可能是一很大的数，以致于当前或相当长的一段时间内人力或计算机不能承受
  - 部分问题，如最小生成树、指派问题等，可以不通过穷举或类似于穷举的方法求得最优解，从而使求解时间大幅下降
  - 部分问题，如背包问题、TSP等，目前还没有找到可以快速求得最优解的方法
- 计算复杂性（computational complexity）
  - 理论计算机科学的一个重要分支，主要研究如何确定一个问题的所有可能的算法解的最优效率或最优效率的界限
  - 运用计算复杂性理论，可以将多数组合优化问题分为$\mathcal{P}$ 问题和$\mathcal{NP}$-难问题两类

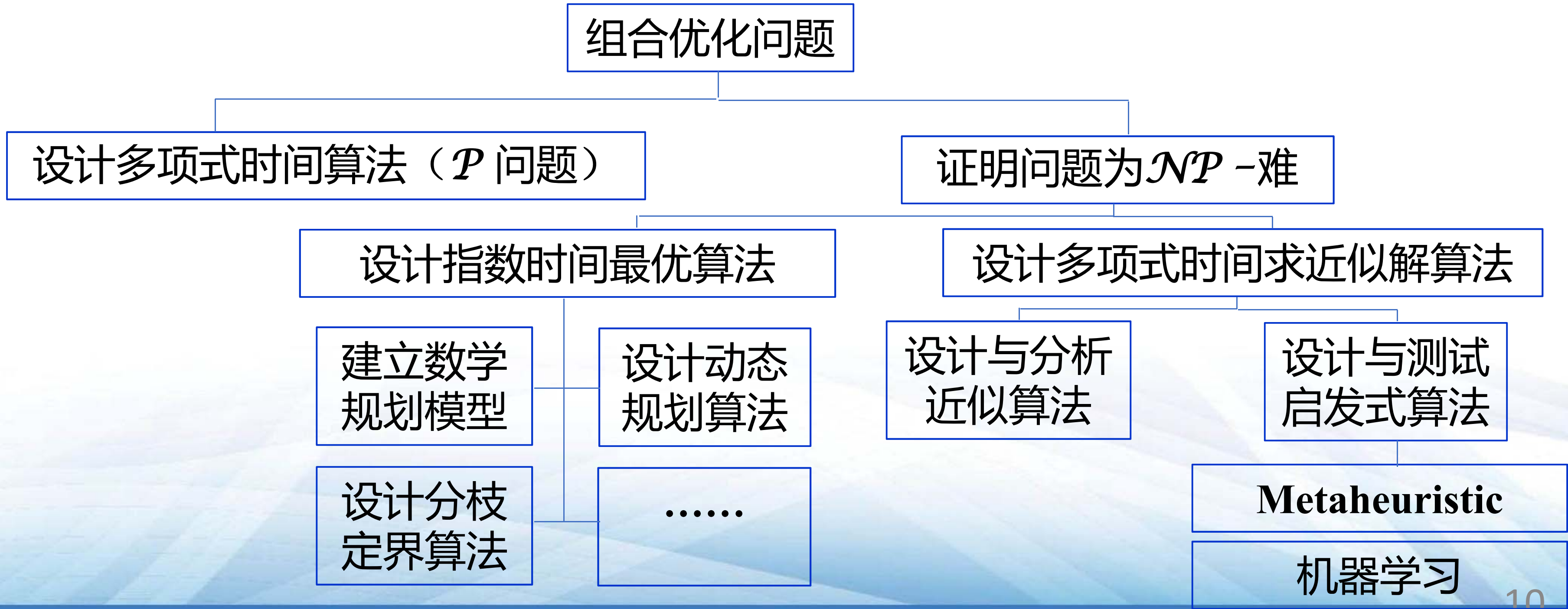
计算复杂性

- $\mathcal{P}$ vs. $\mathcal{NP}$ 问题
  - $\mathcal{P}=\mathcal{NP}$ 与 $\mathcal{P}\subset\mathcal{NP}$ 中何者成立
  - 多数学者相信 $\mathcal{P}\neq\mathcal{NP}$
- **$\mathcal{P}$, $\mathcal{NP}$, $\mathcal{NP}$ –完全与$\mathcal{NP}$–难的通俗解释**
  - $\mathcal{P}$：有（确定性）多项式时间算法的问题
  - $\mathcal{NP}$：有非确定性多项式时间算法的问题
    - 确定性算法是一种特殊的非确定性算法
  - $\mathcal{NP}$ -$\mathcal{C}$： $\mathcal{NP}$类中最难的问题
    - 若一个$\mathcal{NP}$ -$\mathcal{C}$问题有多项式时间算法，所有$\mathcal{NP}$问题都有多项式时间算法
  - $\mathcal{NP}$ -hard：不比$\mathcal{NP}$ -$\mathcal{C}$问题容易的问题
    - 在 $\mathcal{P}\neq\mathcal{NP}$ 假设下，$\mathcal{NP}$-难问题不存在多项式时间最优算法



$\mathcal{P}\neq\mathcal{NP}$

组合优化问题的求解方法



组合优化的发展

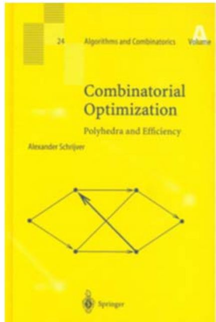
- 组合优化的发展
  - 1950s 最优算法
  - 1960s 近似算法 Lack of unbounded computational resource
  - 1990s 在线问题 Lack of information
  - 2000s 算法博弈论 Lack of coordination

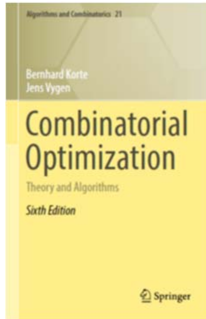
Gödel奖由欧洲理论计算机协会和ACM协会算法和计算理论研究组联合颁发，奖励理论计算机科学领域卓越论文的作者。三篇算法博弈论领域奠基性论文的五位作者Elias Koutsoupias, Christos Papadimitriou, Noam Nisan, Amir Ronen, Tim Roughgarden和Éva Tardos获得了2012年的Gödel奖

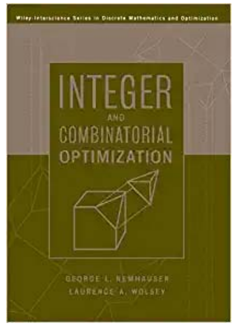
Koutsoupias E, Papadimitriou CH, Worst-case equilibria. ***Proceeding of the 16th Annual Symposium on Theoretical Aspects of Computer Science*****, 404–413, 1999. Also in** ***Computer Science Review*****, 3, 65-69, 2009**

参考资料

- 组合优化
  - Ausiello G, Crescenzi P, Gambosi G, Kann V, Protasi M. *Complexity and Approximation: Combinatorial Optimization Problems and Their Approximability Properties*. Springer, 1999.
  - Cook WJ, Cunningham WH, Pulleyblank WR, Schrijver A. *Combinatorial Optimization*. Wiley, 1997
  - 谈之奕，林凌. 组合优化与博弈论. 浙江大学出版社，2015.
- 算法
  - Cormen TH, Leiserson CE, Rivest RL, Stein C. Introduction to Algorithms (3rd). The MIT Press, 2009.
  - Roughgarden T. Algorithms Illuminated (I-IV). Soundlikeyourself Publishing, 2020.







Schrijver A, *Combinatorial Optimization: Polyhedra and Efficiency*. Springer, 2003.

Korte B, Vygen J, *Combinatorial Optimization: Theory and Algorithms* (6th), Springer, 2017.

Wolsey LA, Nemhauser GL, *Integer and Combinatorial Optimization*, Wiley, 1999.

数学规划

- 数学规划（Mathematical Programming）
  - 若干个变量在满足一些等式或不等式限制条件下，使目标函数取得最大值或最小值
- 建立数学规划模型的主要步骤
  - 确定决策变量
  - 给出目标函数
  - 列出约束条件

$$
\begin{aligned}
\min \quad & f(\mathbf{x}) \\
s.t. \quad & g_i(\mathbf{x}) \geq 0 \quad i = 1, \cdots, s \\
& h_j(\mathbf{x}) = 0 \quad j = 1, \cdots, t \\
& \mathbf{x} \in \mathbb{R}^n
\end{aligned}
$$

目标函数

min 极小
max 极大

subject to
约束条件

决策变量取值范围约束

等式约束

不等式约束

数学规划的求解软件

- 数学规划的求解软件
  - LINGO：Optimization Modeling Software for Linear, Nonlinear, and Integer Programming
  - IBM ILOG CPLEX：High-performance mathematical programming solver for linear programming, mixed-integer programming and quadratic programming
  - Gurobi：The Fastest Mathematical Programming Solver
  - 杉数求解器（Cardinal Optimizer，COPT）：针对大规模优化问题的高效数学规划求解器套件。目前已经可以提供线性、整数规划、非线性问题等多种数学规划求解方案

LINGO the modeling language and optimizer

www.lindo.com

CPLEX Optimizer

www.ibm.com/analytics/cplex-optimizer

GUROBI OPTIMIZATION

www.gurobi.com

杉数科技 Cardinal Operations

www.shanshu.ai/copt

数独

- 数独（Sudoku）
  - 81个方格排列成 9 行 9 列的方块，该方块可划分成 9 个小方块，每个小方块由相邻 3 行 3 列共 9 个方格构成
  - 在每个方格中填入1,2,3,4,5,6,7,8,9共 9 个数字之一，使得每行、每列、每个小方块中填入的数字各不相同
- 数学规划建模
  - 决策变量：第 i 行第 j 列所填的数字
    - $x_{ij} = k$ ：第 i 行第 j 列所填的数字为 k

如何表示 $x_{11}, x_{12}, x_{13}, x_{14}, x_{15}, x_{16}, x_{17}, x_{18}, x_{19}$ 各不相同 $\cancel{x_{11} \neq x_{13}}$

	4		2	8				6
		6	9					5
7			5				8	
			6			4		
	9				8		5	
		2						
	8				2			1
2					9	3		
5				7	3		4	

数独

- 数学规划建模
  - 决策变量：$x_{ijk}=\begin{cases}1 & 第\,i\,行第\,j\,列所填的数字为\,k\\ 0 & 其他\end{cases}$
  - 约束条件
    - 每个格子中填入的数字是唯一的 $\cancel{x_{111}=1,\ x_{112}}=1$
      - 对任意 i,j，$x_{ij1},x_{ij2},x_{ij3},x_{ij4},x_{ij5},x_{ij6},x_{ij7},x_{ij8},x_{ij9}$ 中恰有一个取值1
      - $\sum_{k=1}^{9} x_{ijk}=1,\ i,j=1,\cdots,9$
    - 每一行填入的数字各不相同
      - 对任意 $k,k=1,\cdots,9$，第 i 行各列中只有一个数字为 k
      - 对任意 $k,k=1,\cdots,9$，$x_{i1k},\cdots,x_{i9k}$ 中恰有1个取值1
      - $\sum_{j=1}^{9} x_{ijk}=1,\ i,k=1,\cdots,9$

第4行1,2列中所填数之和大于5

$$\sum_{k=1}^{9} kx_{41k}+\sum_{k=1}^{9} kx_{42k}>5$$

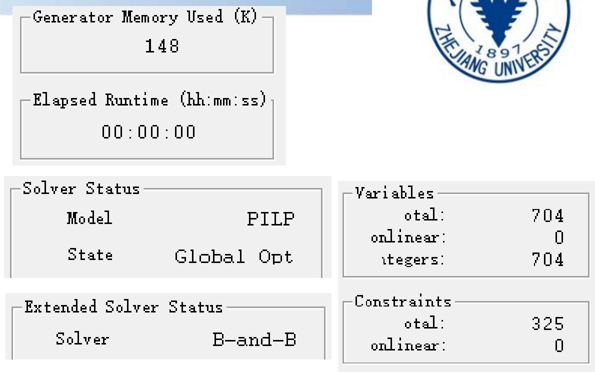
$$x_{124}=1$$

	4		2	8				6
		6	9					5
7			5				8	
			6			4		
	9				8		5	
		2						
	8				2			1
2					9	3		
5				7	3		4	

数独

- 数学规划建模

数学规划可以没有目标函数



- s.t. $\sum_{k=1}^{9} x_{ijk} = 1,\ i, j = 1,\cdots,9$ 每个格子中填入的数字唯一

$\sum_{j=1}^{9} x_{ijk} = 1,\ \ i,k = 1,\cdots,9$ 每行各列中的数各不相同

$\sum_{i=1}^{9} x_{ijk} = 1,\ \ j,k = 1,\cdots,9$ 每列各行中的数各不相同

$\sum_{i=3s+1}^{3s+3} \sum_{j=3t+1}^{3t+3} x_{ijk} = 1,\ \ s,t = 0,1,2, k = 1,\cdots,9$ 每个小方块中的数各不相同

$x_{ijk} = 1,\ \ (i, j,k) \in P$

$x_{ijk} = 0,1,\ \ i, j,k = 1,\cdots,9$

P 由预置的数字及其位置确定，如 $x_{124} = 1$ ，$x_{142} = 1$等

1	4	5	2	8	7	9	3	6
8	3	6	9	4	1	7	2	5
7	2	9	5	3	6	1	8	4
3	1	8	6	2	5	4	9	7
4	9	7	3	1	8	6	5	2
6	5	2	7	9	4	8	1	3
9	8	3	4	6	2	5	7	1
2	7	4	1	5	9	3	6	8
5	6	1	8	7	3	2	4	9

数学规划求解的常见问题

- 认识误区
  - 数学规划只能停留在纸面上，无法真正求解
  - 只要建立数学规划模型，就能通过计算机求出最优解
  - 数学规划是求解优化问题的唯一方法或最有效方法
  - 只有经典或简单问题才能建立数学规划模型
- 方法误区
  - 混淆了数学规划求解与遍历、搜索等其他求解方法
  - 数学规划描述错误、要素不全、徒具形式、无法求解
  - 建立数学规划模型是机械性工作，不需要技巧，模型没有优劣之分

数学规划

- 玩家的策略
  - 位置
    - 停留
    - 行走及到达的区域
  - 资金资源
    - 在矿山停留时是否挖矿
    - 起点购买的资源数量
    - 在村庄经过或停留时是否购买资源及购买的数量

$$x_{ti}=\begin{cases}1, & \text{若第 } t \text{ 天玩家位于区域 } i, \\ 0, & \text{其他,}\end{cases} \quad t=0,1,\ldots,T, i=1,\ldots,n$$

$$z_t=\begin{cases}1, & \text{若第 } t \text{ 天玩家挖矿,} \\ 0, & \text{其他,}\end{cases} \quad t=1,\ldots,T$$

$$b_{tl}\in\mathbb{N}: \text{玩家在第 } t \text{ 天购买的第 } l \text{ 种资源数量,}$$

$$t=0,1,\ldots,T, l=1,2$$

数学规划

- 目标函数：剩余资金=初始资金−购买资源消耗+挖矿收益
- 约束条件
  - 变量定义的必然要求
    - 每一天玩家只能位于一个区域
    - 玩家必须在矿山才能挖矿
    - 玩家必须在起点或村庄才能购买资源
    - ……
  - 具体问题的特定要求
    - 沙暴天不能行走
    - 每天需有足够的资源以供行动时消耗
    - 每天拥有的资源质量之和不能超过负重上限
    - ……

$$x_{ti} = \begin{cases} 1, & \text{若第 } t \text{ 天玩家位于区域 } i \\ 0, & \text{其他} \end{cases}$$

$$\sum_{i=1}^{n} x_{ti} = 1, \quad t = 1, \cdots, T$$

数学规划

- 约束条件

$$x_{ti} = \begin{cases} 1, & \text{若第 } t \text{ 天玩家位于区域 } i \\ 0, & \text{其他} \end{cases} \qquad z_t = \begin{cases} 1, & \text{若第 } t \text{ 天玩家挖矿} \\ 0, & \text{其他} \end{cases}$$

  - 玩家在矿山**停留**时，可通过挖矿获得资金；也可仅在矿山停留时而不挖矿。到达矿山当天不能挖矿。
    - $y_{ti} = \begin{cases} 1, & \text{若第 } t \text{ 天玩家在区域 } i \text{ 停留} \\ 0, & \text{其他} \end{cases}$ $\quad t = 1, 2, \ldots, T, i = 1, \ldots, n$
    - x_{ti} 与 y_{ti} 的联系
      - $y_{t+1,i} = 1 \Leftrightarrow x_{ti} = 1 \wedge x_{t+1,i} = 1$
      - $y_{t+1,i} = x_{ti} x_{t+1,i}$

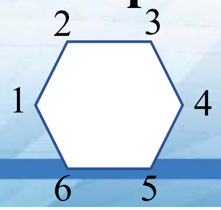
$$\begin{cases} y_{t+1,i} \le x_{ti} \\ y_{t+1,i} \le x_{t+1,i} \\ x_{ti} + x_{t+1,i} \le y_{t+1,i} + 1 \end{cases}$$

    - z_t 与 y_{ti} 的联系
      - $z_t = 1 \Rightarrow \sum_{i \in U} y_{ti} = 1$

$$z_t \le \sum_{i \in U} y_{ti}$$

动态规划

- **动态规划**（**dynamic programming，DP**）
  - 动态规划是求解多阶段决策优化问题的一种数学方法和算法思想
  - 动态规划的主要步骤
    - 将原问题划分为若干个有序的阶段
    - 确定状态和状态变量、建立状态转移方程
    - 给出初始条件，并求解
  - 动态规划求解的问题需满足最优性原理（**Principle of Optimality**）



从1至4的最短路为1-2-3-4 从1至4的最长路为1-2-3-4

从2至4的最短路为2-3-4 从2至4的最长路不为2-3-4

Richard Ernest Bellman
(1920-1984)
美国运筹学家

动态规划

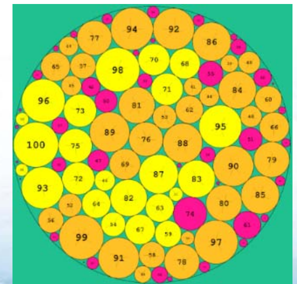
- 动态规划
  - 玩家的状态由当前日期、所处区域、剩余的资源数量所决定
    - $M(t,j,f,w)$：第 t 天位于区域 j，当前剩余食物量和水量分别为 f 和 w 时，可能拥有的最多剩余资金。
  - 状态转移方程
    - 第 t 天不为沙暴，区域 j 不为矿山或村庄。

$$M(t,j,f,w)=\max\left\{\underbrace{M(t-1,j,f+cf_t,w+cw_t)}_{\text{停留}},\ \max_{i\in N(j)\backslash\{n\}}\underbrace{\{M(t-1,i,f+2cf_t,w+2cw_t)\}}_{\text{行走}}\right\}$$

  - 初始条件与最优值
  - 动态规划的算法实现

启发式算法

- 启发式算法（heuristic）
  - 基于某种直观想法、合理假定，或者借助物理、化学、生命科学中的一些原理而设计的算法
  - 体现了在求解的最优性、精确性与求解资源之间的权衡
- **Metaheuristic**
  - 遗传算法
  - 模拟退火算法
  - 禁忌搜索
  - 蚁群算法



- 启发式算法的评估
  - 算法的普适性和解的可行性
  - 算法解的近似性能
    - 一定数量实例的数值模拟
    - 算法值与最优值的比较

启发式算法

- 最优解的性质
  - 存在最优解满足的性质
    - 玩家到达终点时，水和食物恰好耗尽
    - 玩家在路线中相邻两个功能性区域之间的路线为两者的最短路
  - 未必成立的性质
    - 玩家不会多次经过同一个村庄或矿山。
    - 玩家在规定时间的最后一天到达终点
- 基于路线分类的启发式算法
  - 贪婪、局部搜索、动态规划

第二问

- 第二问
  - 只有一名玩家，玩家仅知道当天的天气情况，据此决定当天的行动方案

不确定性

实时性

- 主要问题
  - 缺少清晰的整体求解思路
    - 天气怎么样，策略怎么选，每天怎么办
  - 解决方案未完整体现问题特点和要求
    - 只考虑不确定性，未考虑实时性
    - 只考虑路线，未考虑初始购买资源的数量及其影响
    - 只考虑某种特定情况，未进行参数化分析

第三问

- 第三问
  - 有多名玩家，有相同的初始资金，且同时从起点出发
  - 若有多名玩家在同日有相同的行动，每名玩家消耗资源的数量、购买资源的价格和挖矿的收益与只有单名玩家时不同
- 玩家的目标
  - 所有玩家组成一个团体，团体的目标为所有玩家的剩余资金之和尽可能多 单目标优化
  - 每名玩家的目标均为自身剩余资金尽可能多 **Decentralization** 博弈问题

谢　谢