运筹学课程实验说明

宋艳枝

中国科学技术大学
数学科学学院

2022-12-14

Outline

1. 概述
2. 报告说明
   - 网络最优化
   - 动态规划
   - 无约束优化

Outline

1. 概述
2. 报告说明
   - 网络最优化
   - 动态规划
   - 无约束优化

Project

- 如下题目三选二，编程语言不限。
  - 调研并实现最小成本循环流问题的强多项式时间求解算法。
  - 实现动态规划的一种快速算法。
  - 编程实现基于 Wolfe-Powell 准则的非精确一维搜索 (line search) 的某一种（搜索方向）迭代下降算法，并用于求解无约束最优化问题。
- 要求提交程序代码，用户指南及对应的测试报告。

提交要求

- 除程序本体外，需要完成报告文档，报告的提纲（包括但不限于）
  - 问题描述
  - 算法原理
  - 数据集（案例）说明
  - 程序输入输出说明
  - 程序测试结果
  - 分析总结
- 在课程网站上提交一个包含程序代码和报告文档的 zip 压缩包文件。
- 提交截止日期：2023 年 1 月 6 日 23:59:59.
- 提交网址：http://www.smartchair.org/ustc_or_class_2022/

Outline

Outline

1. 概述
2. 报告说明
   - 网络最优化
   - 动态规划
   - 无约束优化

最小成本循环流

考虑有向图 $G=\{V,E\}$，在各边上定义成本为 $c=(c(e)|e\in E)$，并在各边 $e\in E$ 上引入流的下界值 $l=(l(e)|e\in E)$ 和上界值 $u=(u(e)|e\in E)$，从而得到网络 $\mathcal{N}=(G,c,l,u)$. 求在这个网络上使成本达到最小的流，即所谓的最小成本循环流问题:

$$\begin{aligned} &\min \sum_{e\in E} c(e)x(e) \\ &s.t. \sum_{e\in Out(v)} x(e) - \sum_{e\in In(v)} x(e) = 0, \ \ v\in V \\ &\qquad l(e)\le x(e)\le u(e), \ \ e\in E. \end{aligned}$$

作业要求

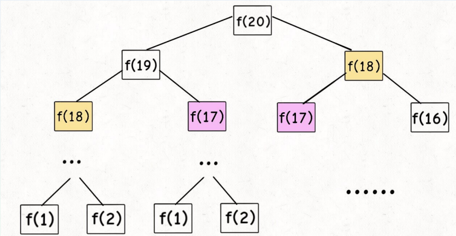
- 调研最小成本循环流问题的强多项式时间求解算法，在报告文末列出参考文献.
- 选定一种调研的算法，在报告中详细写出算法迭代过程，手动实现算法（允许调包）.
- 本作业需至少构造一个数据集，算法程序应至少在该数据集上进行测试；鼓励构造更多实例，充分测试.

Outline

动态规划三要素

下面以斐波那契数列为例辅助理解动态规划的原理，但是斐波那契数列不涉及求最值，所以严格意义上不是动态规划.

- 重叠子问题：划分为子问题时，存在重复求解的情况，需要 DP Table 优化穷举过程。



- 最优子结构：通过求得每个子问题最值，可以得到原问题最值。
- 状态转移方程：

$$f(x) = \begin{cases} 1, & n = 1, 2 \\ f(n-1) + f(n-2), & n > 2 \end{cases}$$

凑零钱问题

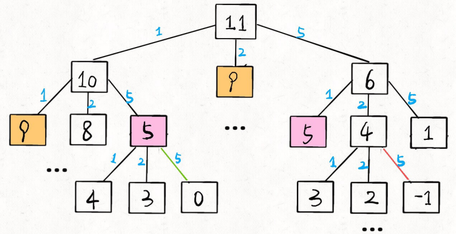
给定 k 种面值的硬币，面值分别为 $c_1, c_2, \cdots, c_k$，每种硬币的数量无限，再给一个总金额 Amount，问最少需要几枚硬币凑出这个金额；如果不可能凑出，算法返回-1。
不妨以 $k=3$，$c_1=1$，$c_2=2$，$c_3=5$，$Amount=11$ 为例。

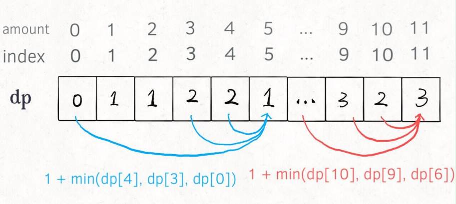
- 具有最优子结构：由于硬币数量无限制，所以子问题之间没有相互限制、相互独立。求 $Amount=11$ 时的最少硬币数（原问题），若已知凑出 $Amount=10$ 的最少硬币数（子问题），只需要把子问题的答案加一（再选一枚面值为 1 的硬币）。
- 状态转移方程

$$dp(n)=\begin{cases}0, & n=0\\ -1, & n<0\\ min\{dp(n-coin)+1 | coin \in coins\}, & n>0\end{cases}$$

凑零钱问题

- 消除重叠子问题：构造 DP Table.





作业要求

- 调研一种动态规划的快速算法，在报告中详细写出算法迭代过程，手动实现算法（允许调包）.
- 本作业需至少构造一个实际案例（多重背包问题/凑零钱问题/投资问题/排序问题等），算法程序应至少在该案例上进行测试；鼓励构造更多实例，充分测试.

  多重背包问题：一位旅行者能承受的背包最大重量是 b 千克，现有 n 种物品供他选择装入背包，第 i 种物品单件重量为 a_i 千克，最多有 s_i 件，每件的价值为 c_i，$1 \leq i \leq n$. 设第 i 种物品装载数量是 x_i，问旅行者应该如何选择所携带的物品件数，以使得总装载价值最大.

Outline

牛顿法

对于如下的无约束优化问题

$$\min_{x} f(x)$$

牛顿法的一般迭代格式

(0) 初始化：选取适当的初始点 $x^{(0)}$，令 $k := 0$.

(1) 计算搜索方向：$d^{(k)} = -\nabla^2 f(x^{(k)})^{-1} \nabla f(x^{(k)})$.

(2) 确定步长因子：采用非精确的一维搜索确定步长因子 α_k.

(3) 更新迭代点：令 $x^{(k+1)} = x^{(k)} + \alpha_k d^{(k)}$. 置 $k := k + 1$, 返回第 (1) 步.

拟牛顿法的一般迭代格式

(0) 初始化：选取适当的初始点 $x^{(0)}$，令 $H_0 = I, k := 0$.

(1) 计算搜索方向：$d^{(k)} = -H_k \nabla f(x^{(k)})$.

(2) 确定步长因子：采用非精确的一维搜索确定步长因子 α_k

(3) 更新迭代点：$x^{(k+1)} = x^{(k)} + \alpha_k d^{(k)}$

(4) 基于 $x^{(k)}$ 到 $x^{(k+1)}$ 的梯度变化, 更新 Hesse 矩阵逆的近似, 即确定满足正割条件的 H_{k+1}. 置 $k := k + 1$, 返回第 (1) 步.

共轭梯度法

共轭梯度法的一般迭代格式

(0) 初始化：给定正定矩阵 G, 选取初始点 $x^{(0)}$, 计算 $g^{(0)} = \nabla f(x^{(0)})$ 并构造 $d^{(0)}$ 使得 ${g^{(0)}}^T d^{(0)} < 0$. 令 $k := 0$.

(1) 确定步长因子：采用非精确的一维搜索确定步长因子 α_k.

(2) 更新迭代点：$x^{(k+1)} = x^{(k)} + \alpha_k d^{(k)}$

(3) 构造 $d^{(k+1)} = -g^{(k+1)} + \beta_k d^{(k)}$ 使得 ${d^{(k+1)}}^T G d^{(j)} = 0,\ j = 0, 1, \cdots, k$, 其中 β_k 选择不唯一. 置 $k := k + 1$, 返回第 (1) 步.

对于非二次函数，共轭梯度法迭代 n 步后所产生的搜索方向 $d^{(k+1)}$ 可能不再是下降方向，需要采用重启动策略.

确定步长因子

- 精确一维搜索：通过求解下面一维最优化问题确定步长

$$\min_{\alpha \leq 0} \phi(\alpha) = f(x^{(k)} + \alpha d^{(k)})$$

- 非精确一维搜索：找出满足某些适当条件的粗略近似解作为步长，提升算法的整体计算效率.
  Wolfe-Powell conditions:

$$\varphi(\alpha) \leq \varphi(0) + \rho\alpha\varphi^{'}(0)$$

$$\varphi^{'}(\alpha) \geq \sigma\varphi^{'}(0)$$

  其中 $\rho \in (0, 1/2)$, $\sigma \in (\rho, 1)$ 是固定参数.

非精确一维搜索算法

设 $\bar{\alpha}_k$ 是使得 $f(x^{(k)}+\alpha d^{(k)})=f(x^{(k)})$ 的最小正数 α.

基于 Wolfe-Powell 准则的非精确一维搜索算法:

(0) 给定初始一维搜索区间 $[0,\bar{\alpha}]$, 以及 $\rho\in(0,\frac{1}{2})$,$\sigma\in(\rho,1)$. 计算 $\varphi_0=\varphi(0)=f(x^{(k)})$, $\varphi_0^{'}=\varphi^{'}(0)=\nabla f(x^{(k)})^T d^{(k)}$. 并令 $a_1=0,\ a_2=\bar{\alpha},\ \varphi_1=\varphi_0,\ \varphi_1^{'}=\varphi_0^{'}$. 选取适当的 $\alpha\in(a_1,a_2)$.

(1) 计算 $\varphi=\varphi(\alpha)=f(x^{(k)}+\alpha d^{(k)})$. 若 $\varphi(\alpha)\leq\varphi(0)+\rho\alpha\varphi^{'}(0)$, 则转到第 (2) 步。否则, 由 $\varphi_1,\varphi_1^{'},\varphi$ 构造二次插值多项式 $p^{(1)}(t)$, 并得其极小点 $\hat{\alpha}$. 令 $a_2=\alpha,\alpha=\hat{\alpha}$, 重复第 (1) 步.

(2) 计算 $\varphi^{'}=\varphi^{'}(\alpha)=\nabla f(x^k+\alpha d^{(k)})^T d^{(k)}$. 若 $\varphi^{'}(\alpha)\geq\sigma\varphi^{'}(0)$, 则输出 $\alpha_k=\alpha$, 并停止搜索. 否则由 $\varphi,\varphi^{'},\varphi_1^{'}$ 构造两点二次插值多项式 $p^{(2)}(t)$, 并求得极小点 $\hat{\alpha}$. 令 $a_1=\alpha,\alpha=\hat{\alpha}$, 返回第 (1) 步.

作业要求

- 实现基于 Wolfe-Powell 准则的非精确一维步长搜索算法.
- 基于非精确一维步长搜索，从牛顿法、拟牛顿类方法 (DFP/BFGS)、共轭梯度法中选择一种算法，手动实现.
- 本作业需至少构造一个函数 (例如 Rosenbrock 函数)，应用算法在不同初始值下求解无约束最优化问题，并分析不同初值点对结果的影响.

$$f(x) = \sum_{i=1}^{d-1}[100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2], \quad x \in \mathbb{R}^d$$

.

Thanks for your attention!