

云南大学数学与统计学院  
《运筹学通论实验》上机实践报告

课程名称：运筹学实验	年级：2015 级	上机实践成绩：
指导教师：李建平	姓名：刘鹏	专业：信息与计算科学
上机实践名称：求给定线性规划问题	学号：20151910042	上机实践日期：2018-03-28
上机实践编号：2	组号：	

一、 实验目的

通过编程解决简单的线性规划问题，了解线性规划问题求解的步骤。

二、 实验内容

对于线性规划问题：

$$\max z = c x = c_1 x_1 + c_2 x_2 + \cdots + c_n x_n,$$

s.t.

$$a_1 x_1 + a_2 x_2 + \cdots + a_n x_n = b$$

其中， $a_i \in \mathbb{R}_0^+$ ， $x_i \geq 0$ 。用 C 语言编程实现求 $z$ 的最大值。

三、 实验平台

Microsoft Windows 10 Pro Workstation 1803;  
Microsoft Visual Studio 2017 Enterprise。

四、 算法设计<sup>1</sup>

**Algorithm:** Simple linear Programming  
**Input:**  $A = (a_1, a_2, \cdots, a_n) > 0$ ,  $c = (c_1, c_2, \cdots, c_n)$ ,  $b$   
**Output:**  $z, x$   
**Begin**  
**Step 1:** if  $\exists a_i = 0, c_i \neq 0$ , output  $\infty$ , GOTO **End**  
**Step 2:**  $\mathbf{tmp} = b \cdot c ./ A$   
**Step 3:**  $z = \text{MAX}(\mathbf{tmp})$   
**Step 4:** Output  $z$ , together with all satisfied  $x_i = (0, \cdots, b/A_i, \cdots, 0)$  where  $c x_i = z$   
**End**

五、 程序代码

当 $n = 2$ 时，约束条件是笛卡尔坐标系 $Ox_1x_2$ 下的一条直线 $a_1x_1 + a_2x_2 = b$ ，可行解必然在角点上，所

<sup>1</sup> 此处的伪代码中，矩阵运算符的意义均与 MATLAB 语言一致，如矩阵的左除、右除和点除等。

以带入 $(0, \frac{b}{a_2})$ ,  $(\frac{b}{a_1}, 0)$ ,  $(0, 0)$ , 求出最大值即可。所以把 $n$ 推广到任意大也基本是这个道理。如果目标函数中出现了约束条件中不存在的变量, 说明这个变量不受约束, 自然可以往无穷大方向发展, 这时候输出无穷大。

## 5.1 程序描述

这个程序模拟了一个十分简单的 linear programming 算法。

## 5.2 程序代码

```

1  /*
2  * Copyright (c) 2018, Liu Peng, School of Mathematics and Statistics, YNU
3  * Apache License.
4  *
5  * 文件名称: Source.cpp
6  * 文件标识: 见配置管理计划书
7  * 摘 要: 对一个仅有一个约束的线性规划问题用遍历法求解。
8  *
9  * 当前版本: 1.0
10 * 作 者: 刘鹏
11 * 创建日期: 2018 年 6 月 25 日
12 * 完成日期: 2018 年 7 月 07 日
13 *
14 * 取代版本:
15 * 原作者 : 刘鹏
16 * 完成日期:
17 */
18
19 /*
20 * A function can solve a simple LP problem with only one
21 * condition.
22 */
23
24 #include<stdio.h>
25 #include<stdlib.h>
26
27 #define INFINITY -1
28
29 int main(int argc, char *argv[]) {
30
31     double A[] = { 1, 2 }; // x + 2y = b
32     double b = 3;
33     double c[] = [1];
34
35     int i;
36     for (i = 0; i < 2; i++) {
37         if (A[i] == 0 && c[i] != 0) {
38             return INFINITY;
39         }
40     }
41

```

```
42     double d[] = [1];
43     // generate divisions
44
45     if (d[0] > d[1]) {        // sort
46         printf("x = (%2.0f, 0)\nMAX z = %2.0f\n", d[0] / c[0], d[0]);
47     }
48     else {
49         printf("x = (0, %2.0f)\n, MAX z = %2.0f\n", d[1] / c[1], d[1]);
50     }
51     system("pause");
52     return 0;
53 }
```

程序代码 1

## 六、 运行结果



```
D:\Nutstore\myStudyMaterial\Grade_3_Term_2\Operations_Research\Operations_Research_Report\Code\02.linprog\Debug\linprog.exe
x = ( 3, 0)
MAX z = 3
请按任意键继续. . .
```

运行结果 1

### 6.1 代码分析

代码虽然简单，但是五脏俱全。因为是练习性实验，主力放在调用 MATLAB 的 linprog.m 上，这里仅是一个示例。

## 七、 实验体会

通过思考，比较清楚地了解了求 BFS 的步骤<sup>[1]</sup>。

## 八、 参考文献

- [1] HILLIER F S, LIEBERMAN G J. 运筹学导论 [M]. 9th ed. 北京: 清华大学出版社, 2010.