091M4041H - Algorithm Design and Analysis

Assignment 4

2019 年 8 月 30 日

目录

1 Linear-inequality feasibility

Given a set of m linear inequalities on n variables $x_1, x_2, ..., x_n$, the **linear-inequality feasibility problem** asks if there is a setting of the variables that simultaneously satisfies each of the inequalities.

Show that if we have an algorithm for linear programming, we can use it to solve the linear-inequality feasibility problem. The number of variables and constraints that you use in the linear-programming problem should be polynomial in n and m.

1.1 Formulation of Linear Programing

设共有 m 个线性不等式和 n 个 x_i 变量 $(1 \leq i \leq n)$，转化成线性规划问题求解，即对 m 个线性不等式中每个都增加一个变量 $x_j(n+1 \leq j \leq n+m)$，且 $x_j \leq 0$，求解目标函数 $\max \sum -x_j$，若其值为 0，则原线性不等式是可满足的。

$$\begin{gathered}\max \sum -x_j \\ \text{s.t. } \sum_i a_i x_i - x_j \leq b_j \\ x_i \geq 0 \\ x_j \geq 0\end{gathered}$$

2 Interval Scheduling Problem

A teaching building has m classrooms in total, and n courses are trying to use them. Each course $i(i = 1, 2, ..., n)$ only uses one classroom during time interval $[S_i, F_i)(F_i > S_i > 0)$. Considering any two courses can not be carried on in a same classroom at any time, you have to select as many courses as possible and arrange them without any time collision. For simplicity, suppose 2n elements in the set $\{S_1, F_1, ..., S_n, F_n\}$ are all different.

Please use ILP to solve this problem, then construct an instance and use GLPK or Gurobi or other similar tools to solve it.

2.1 Formulation of Linear Programing

设 x_i 表示第 i 门课是否选择，选择为 1，否则为 0，将 $S_1, F_1, ..., S_n, F_n$ 排序使得一天的课程时间被划分为 2n-1 个区间 $T_1, ..., T_{2n-1}$，用 N_k 表示一门课所占用的所有区间, 使得所有选中课所占的相同总区间个数小于 m，即相同时间的课程不多于 m 个。

$$\max \sum_{i=1}^{n} x_i$$
$$\text{s.t. } \sum_{i \in N_k} x_i \leq m$$
$$x_i \in \{0, 1\}, \text{ for all i = 1,2,...,n}$$

2.2 Soving by GLPK

```
var x1 binary;
var x2 binary;
var x3 binary;
var x4 binary;
var m = 2;

maximize z: x1 + x2 + x3 + x4;

s.t. con1 : x1 <= m;
s.t. con2 : x1 + x2 <= m;
s.t. con3 : x2 <= m;
s.t. con4 : x2 + x3 <= m;
s.t. con5 : x2 + x3 + x4 <= m;
s.t. con6 : x3 <= m;

```

图 1: problem2.mod

```
Problem:    lp_problem2
Rows:       7
Columns:    5 (4 integer, 4 binary)
Non-zeros:  20
Status:     INTEGER OPTIMAL
Objective:  z = 3 (MAXimum)

   No.   Row name        Activity     Lower bound   Upper bound
------ ------------    ------------- ------------- -------------
     1 z                           3
     2 con1                       -1                          -0
     3 con2                        0                          -0
     4 con3                       -1                          -0
     5 con4                        0                          -0
     6 con5                        0                          -0
     7 con6                       -1                          -0

   No. Column name       Activity     Lower bound   Upper bound
------ ------------    ------------- ------------- -------------
     1 x1           *              1             0             1
     2 x2           *              1             0             1
     3 x3           *              1             0             1
     4 x4           *              0             0             1
     5 m                           2             2             =

Integer feasibility conditions:

KKT.PE: max.abs.err = 0.00e+00 on row 0
        max.rel.err = 0.00e+00 on row 0
        High quality

KKT.PB: max.abs.err = 0.00e+00 on row 0
        max.rel.err = 0.00e+00 on row 0
        High quality

End of output
```

图 2: problem2.sol

3 Gas Station Placement

Let's consider a long, quiet country road with towns scattered very sparsely along it. Sinopec, largest oil refiner in China, wants to place gas stations along the road. Each gas station is assigned to a nearby town, and the distance between any two gas stations being as small as possible. Suppose there are n towns with distances from one endpoint of the road being $d_1, d_2, ..., d_n$. n gas stations are to be placed along the road, one station for one town. Besides, each station is at most r far away from its correspond town. $d_1, ..., d_n$ and r have been given and satisfied $d_1 < d_2 < ... < d_n, 0 < r < d_1 and d_i + r < d_{i+1} - r$ for all i. The objective is to find the optimal placement such that the maximal distance between two successive gas stations is minimized.

Please formulate this problem as an LP.

3.1 Formulation of Linear Programing

设 x_i 表示加油站与路起始点的距离，目标函数为最小化所有加油站间最大距离。加油站与城镇的距离小于 r，x_i 应落在城镇半径为 r 的区间内，且 x_i 应大于 0。

$$\min \max\{x_{i+1} - x_i\} \text{ i = 0,1,...,n}$$
$$\text{s.t. } x_i - d_i \leq r$$
$$d_i - x_i \leq r$$
$$x_i \geq 0$$

4 Stable Matching Problem

n men $(m_1, m_2, ..., m_n)$ and n women $(w_1, w_2, ..., w_n)$, where each person has ranked all members of the opposite gender, have to make pairs. You need to give a stable matching of the men and women such that there is no unstable pair. (A matching is unstable if: there is an element A of the first matched set which prefers some given element B of the second matched set over the element to which a is already matched, and B also prefers A over the element to which B is already matched.) Please choose one of the two following known conditions, formulate the problem as an ILP, construct an instance and use GLPK or Gurobi or other similar tools to solve it.

1. You have known that for every two possible pairs (man m_i and woman w_j, man m_k and woman w_l), whether they are stable or not. If they are stable, then $S_{i,j,k,l} = 1$; if not, $S_{i,j,k,l} = 0$. $(i, j, k, l \in \{1, 2, ..., n\})$

2. You have known that for every man m_i, whether m_i likes woman w_j more than w_k. If he does, then $p_{i,j,k} = 1$; if not, $p_{i,j,k} = 0$. Similarly, if woman w_i likes man m_j more than m_k, then $q_{i,j,k} = 1$, else $q_{i,j,k} = 0$. $(i, j, k \in \{1, 2, ..., n\})$

4.1 Formulation of Linear Programing

1. 设 x_{ij} 表示 man_i 与 $women_j$ 是否彼此喜欢，若是则为 1，否则为 0。对于一个男人，他只能选择所有女人中的一个，对于一个女人，她也只能选择所有男人中的一个。若 x_{ij} 与 x_{kl} 同时为 1，则 $S_{i,j,k,l} = 1$，若 x_{ij} 与 x_{kl} 中有且仅有一个为 1，则 $S_{i,j,k,l} = 1$ 仍旧稳定，当 x_{ij} 与 x_{kl} 同时为 0 时，$S_{i,j,k,l}$ 可能为 1 或 0。因此可以总结如下

$$\begin{gathered}
\min 0 \\
\text{s.t. } \textstyle\sum_{i=1}^{n} x_{ij} = 1 \\
\textstyle\sum_{j=1}^{n} x_{ij} = 1 \\
x_{ij} + x_{kl} \le S_{i,j,k,l} + 1 \\
x_{ij} \in \{0, 1\}
\end{gathered}$$

2. 当 $p_{i,j,l}, p_{j,i,l}$ 均为 1 时，$x_{ij} = 1$,$x_{kl} = 0/1$, 当 $p_{i,j,l}, p_{j,i,l}$ 均为 0 时，$x_{ij} = 0$,$x_{kl} = 0/1$, 当 $p_{i,j,l} = 1, p_{j,i,l} = 0$ 时，$x_{ij} = 1$,$x_{kl} = 0/1$，当 $p_{i,j,l} = 0, p_{j,i,l} = 1$ 时，$x_{ij} = 0$,$x_{kl} = 0/1$。因此可以总结如下

$$\begin{gathered}
\min 0 \\
\text{s.t. } \textstyle\sum_{i=1}^{n} x_{ij} = 1 \\
\textstyle\sum_{j=1}^{n} x_{ij} = 1 \\
x_{ij} + x_{kl} \le 3 - p_{i,j,l} - p_{j,i,l} \\
x_{ij} \in \{0, 1\}
\end{gathered}$$

4.2 Soving by GLPK

```
var xij binary;
var xil binary;
var xkj binary;
var xkl binary;
var Sijkl binary;

minimize z: 0;

s.t. con1 : xij + xil = 1;
s.t. con2 : xkj + xkl = 1;
s.t. con3 : xij + xkj = 1;
s.t. con4 : xil + xkl = 1;
s.t. con5 : xij + xkl <= Sijkl + 1;

```

图 3: problem4_1.mod

```
Problem:    lp_problem4
Rows:       6
Columns:    5 (5 integer, 5 binary)
Non-zeros:  11
Status:     INTEGER OPTIMAL
Objective:  z = 0 (MINimum)

   No.   Row name        Activity     Lower bound   Upper bound
------ ------------    ------------- ------------- -------------
     1 z                           0
     2 con1                        1             1             =
     3 con2                        1             1             =
     4 con3                        1             1             =
     5 con4                        1             1             =
     6 con5                        0                           1

   No. Column name       Activity     Lower bound   Upper bound
------ ------------    ------------- ------------- -------------
     1 xij          *              0             0             1
     2 xil          *              1             0             1
     3 xkj          *              1             0             1
     4 xkl          *              0             0             1
     5 Sijkl        *              0             0             1

Integer feasibility conditions:

KKT.PE: max.abs.err = 0.00e+00 on row 0
        max.rel.err = 0.00e+00 on row 0
        High quality

KKT.PB: max.abs.err = 0.00e+00 on row 0
        max.rel.err = 0.00e+00 on row 0
        High quality

End of output

```

图 4: problem4_1.sol

```
var xij binary;
var xil binary;
var xkj binary;
var xkl binary;
var pijl binary;
var pjil binary;

minimize z: 0;

s.t. con1 : xij + xil = 1;
s.t. con2 : xkj + xkl = 1;
s.t. con3 : xij + xkj = 1;
s.t. con4 : xil + xkl = 1;
s.t. con5 : xij + xkl <= 3 - pijl - pjil;
```

图 5: problem4_2.mod

```
Problem:    lp_problem4_2
Rows:       6
Columns:    6 (6 integer, 6 binary)
Non-zeros:  12
Status:     INTEGER OPTIMAL
Objective:  z = 0 (MINimum)

   No.   Row name        Activity     Lower bound   Upper bound
------ ------------    ------------- ------------- -------------
     1 z                          0
     2 con1                       1             1             =
     3 con2                       1             1             =
     4 con3                       1             1             =
     5 con4                       1             1             =
     6 con5                       0                           3

   No. Column name       Activity     Lower bound   Upper bound
------ ------------    ------------- ------------- -------------
     1 xij          *             0             0             1
     2 xil          *             1             0             1
     3 xkj          *             1             0             1
     4 xkl          *             0             0             1
     5 pijl         *             0             0             1
     6 pjil         *             0             0             1

Integer feasibility conditions:

KKT.PE: max.abs.err = 0.00e+00 on row 0
        max.rel.err = 0.00e+00 on row 0
        High quality

KKT.PB: max.abs.err = 0.00e+00 on row 0
        max.rel.err = 0.00e+00 on row 0
        High quality

End of output
```

图 6: problem4_2.sol

5 Duality

Please write the dual problem of the MultiCommodityFlow problem in Lec8.pdf, and give an explanation of the dual variables.

Please also construct an instance, and try to solve both primal and dual problem using GLPK or Gurobi or other similar tools.

5.1 Formulation of Linear Programing

primal:

$$\max 0$$
$$\text{s.t.} \textstyle\sum_{i=1}^{k} f_i(u,v) \le c(u,v)$$
$$\textstyle\sum_{v,(u,v)\in E} f_i(u,v) - \sum_{v,(v,u)\in E} f_i(v,u) = 0$$
$$\textstyle\sum_{v,(s_i,v)\in E} f_i(s_i,v) - \sum_{v,(v,s_i)\in E} f_i(v,s_i) = d_i$$
$$f_i(u,v) \ge 0$$

duality: 用 x_{uv} 代表第一条约束，y_{iu} 代表第二，三条约束:

$$\min c(u,v)x_{uv} + d_i y_{is_i}$$
$$\text{s.t. } x_{uv} + y_{iu} - y_{iv} \ge 0$$
$$x_{ut_i} + y_{iu} \ge 0$$
$$x_{t_iv} - y_{iv} \ge 0$$
$$x_{uv} \ge 0$$

6 Dual Simplex Algorithm

For the problem

minimize

$$-7x_1 + 7x_2 - 2x_3 - x_4 - 6x_5$$

subject to:

$$\begin{gathered} 3x_1 - x_2 + x_3 - 2x_4 = -3 \\ 2x_1 + x_2 + x_4 + x_5 = 4 \\ -x_1 + 3x_2 - 3x_4 + x_6 = 12 \\ x_i \geq 0, (i = 1, ..., 6) \end{gathered}$$

Implement dual simplex algorithm with your favorate language to solve this problem, and make comparison result with GLPK or Gurobi or other similar tools.

6.1 Formulation of Linear Programing

对偶方程可表示为

$$\begin{gathered} \max\ -3y_1 + 4y_2 + 12y_3 \\ 3y_1 + 2y_2 - y_3 \leq -7 \\ -y_1 + y_2 + 3y_3 \leq 7 \\ y_1 \leq -2 \\ -2y_1 + y_2 - 3y_3 \leq -1 \\ y_2 \leq -6 \\ y_3 \leq 0 \end{gathered}$$

6.2 Soving by GLPK

```
var y1;
var y2;
var y3;

maximize z: -3 * y1 + 4 * y2 + 12 * y3;

s.t. con1 : 3 * y1 + 2 * y2 - y3 <= -7;
s.t. con2 : -1 * y1 + y2 + 3 * y3 <= 7;
s.t. con3 : y1 <= -2;
s.t. con4 : -2 * y1 + y2 - 3 * y3 <= -1;
s.t. con5 : y2 <= -6;
s.t. con6 : y3 <= 0;

```

图 7: problem6.mod

```
Problem:    lp_problem6
Rows:       7
Columns:    3
Non-zeros:  15
Status:     OPTIMAL
Objective:  z = -16.5 (MAXimum)

   No.   Row name   St   Activity     Lower bound   Upper bound    Marginal
------ ------------ -- ------------- ------------- ------------- -------------
     1 z            B          -16.5
     2 con1         B          -19.5                          -7
     3 con2         B           -3.5                           7
     4 con3         B           -2.5                          -2
     5 con4         NU            -1                          -1           1.5
     6 con5         NU            -6                          -6           2.5
     7 con6         NU             0                          -0          16.5

   No. Column name  St   Activity     Lower bound   Upper bound    Marginal
------ ------------ -- ------------- ------------- ------------- -------------
     1 y1           B           -2.5
     2 y2           B             -6
     3 y3           B              0

Karush-Kuhn-Tucker optimality conditions:

KKT.PE: max.abs.err = 0.00e+00 on row 0
        max.rel.err = 0.00e+00 on row 0
        High quality

KKT.PB: max.abs.err = 0.00e+00 on row 0
        max.rel.err = 0.00e+00 on row 0
        High quality

KKT.DE: max.abs.err = 0.00e+00 on column 0
        max.rel.err = 0.00e+00 on column 0
        High quality

KKT.DB: max.abs.err = 0.00e+00 on row 0
        max.rel.err = 0.00e+00 on row 0
        High quality
```

图 8: problem6.sol