OR 第六周上机作业

20123101 李昀哲

2023.1.5

1、给分支定界法求解（A 包含一个单位矩阵）：

接口函数[xstar,fxstar] = BranchBound(A,b,c)

判断整数条件可用：abs(round(x) –x) < 1e-3

```
A = [-1 3 1 0; 7 1 0 1];
b = [6 35]';
c = [7 9 0 0]';
```

准备部分：

```matlab
% Operational Research
% @author 李昀哲 20123101
% Jan 5, 2023
A = [-1 3 1 0; 7 1 0 1];
b = [6; 35];
c = [-7; -9;0;0];%标准格式是求min

lb = [0; 0;0;0]; %x值的初始范围下界
ub=[inf;inf;inf;inf];%x值的初始范围上界

optX = [0; 0];
optVal = 0;
```

BranchBound 函数：

```matlab
% Operational Research
% @author 李昀哲 20123101
% Jan 5, 2023
%Aeq    等式约束系数矩阵；
%Beq    等式约束常数向量；
%vlb    定义域的下界；
%vub    定义域的上界；
%optXin    每次迭代的最优x
%optF   每次迭代最优的f值
%iter   迭代次数
function [xstar, fxstar] = BranchBound(A, b, c, Aeq, Beq, vlb, vub, optXin, optF, iter)
    global optX optVal optFlag;
    iter = iter + 1;
    optX = optXin; optVal = optF;

    [x, fit, status] = linprog(c, A, b, Aeq, Beq, vlb, vub, []);

    if status ~= 1%没有找到最优解
        xstar = x;
        fxstar = fit;
        flagOut = status;
        return;
    end
```

```
    if max(abs(round(x) - x)) >= 1e-3%找到的函数最优解仍不是整数解
        if fit > optVal
            xstar = x;
            fxstar = fit;
            flagOut = -100;
            return;
        end

    else%此时解得的函数解为整数解
        if fit > optVal
            xstar = x;
            fxstar = fit;
            flagOut = -101;
            return;
        else
            optVal = fit;
            optX = x;
            optFlag = status;
            xstar = x;
            fxstar = fit;
            flagOut = status;
            return;
        end
    end
end
    midX = abs(round(x) - x);
    notIntV = find(midX > 1e-3);
    pXidx = notIntV(1);
    tempVlb = vlb;
    tempVub = vub;
    if vub(pXidx) >= fix(x(pXidx)) + 1
        tempVlb(pXidx) = fix(x(pXidx)) + 1;
        [~, ~] = BranchBound(A, b, c, Aeq, Beq, tempVlb, vub, optX, optVal, iter + 1);
    end

    if vlb(pXidx) <= fix(x(pXidx))
        tempVub(pXidx) = fix(x(pXidx));
        [~, ~] = BranchBound(A, b, c, Aeq, Beq, vlb, tempVub, optX, optVal, iter + 1);
    end
    xstar = optX;
    fxstar = optVal;
    flagOut = optFlag;
end
```

函数调用：

```
[xstar, fxstar] = BranchBound(A, b, c, [], [], lb, ub, optX, optVal, 0)
```

运行结果：

```
xstar =

    4
    3
    0
    0


fxstar =

   -55
```