**LPGenerator类API文档**

**使用场景：**

假设有一lp的问题为：

最优化式：c\*x

约束式：a1\*x<=b1

a2\*x>=b2

a3\*x==b3

变量：

X为N个命名方式从x\_0至x\_N-1的变量

整数规划时x仅可以取0或者1，所以没有变量范围可以设置，同时不允许设置变量范围。

线性规划时设变量范围为d1<=x<=d2

**欲使用LPGenerator产生与上述问题相对应的lp文件:**

**第一步：**创建一个LPGenerator的对象

**第二步：**设置参数：(以下方法均为LPGenerator类的方法)

public void setExpression(ExpType type,float cMat[]):

用于设置最优化式，其中type是LPGenerator类中的枚举对象，可以取ExpType.Maximize、ExpType.Minimize，分别表示最大化及最小化表达式的值。cMat[]是问题中的一维系数矩阵。

public void setConstraints(float a1Mat[][],float a2Mat[][],float a3Mat[][],float b1Mat[],float b2Mat[],float b3Mat[])

用于设置约束式，a1Mat[][]对应问题中的二维数组a1,其他参数的对应方式相同。由于题中对于某一种约束式可以没有，所以当a1Mat[][]、b1Mat[]中任意一个为null说明没有小于类型的约束式，a2Mat[][]等类似。

public void setVariable(VarType type,int Num,float d1Mat[],float d2Mat[])

用于设置变量。type参数也是LPGenerator类中的枚举对象，可以取VarType.INT、VarType.FLOAT，分别表示整数及线性规划。Num是变量的个数。特别注意的是整数规划时d1Mat[]、d2Mat[]参数必须取null，否则lp文件将无法求解。而在线性规划中视问题d1Mat[]、d2Mat[]可以取null或者不取null。

需要注意一点的是各个参数的维数一定要同表达式相符，且只能为标准矩阵。

**第三步：**设置输出文件的文件名：

public void generateLPFile(String fileName)

fileName就是文件名，但是并不包括后缀，例如当该参数为”test”时，输出文件全名为”test.lp”，文件的储存路径是项目文件的相对路径” ./lpFiles/”。

**第四步：**如果一切正确，将会在lpFiles目录中找到生成的lp文件。

**实例使用代码：**

public static void main(String args[])

{

LPGenerator lpGenerator=new LPGenerator();

float c[]={3,2};

float a[][]={{1,2},{4,5}};

float b[]={10,20};

float d1[]={0,0};

float d2[]={10000,10000};

lpGenerator.setExpression(ExpType.Maximize,c);

lpGenerator.setConstraints(a,null,null,b,null,null);

lpGenerator.setVariable(VarType.FLOAT,2,d1,d2);

lpGenerator.generateLPFile("test");

}

**使用此代码将得到的lp文件内容为：**

\\* algorithm \*\

Maximize

OBJ: 3.0 x\_0 + 2.0 x\_1

Subject To

\_C0: x\_0 + 2.0 x\_1 <= 10.0

\_C1: 4.0 x\_0 + 5.0 x\_1 <= 20.0

Bounds

0.0 <= x\_0 <= 10000.0

0.0 <= x\_1 <= 10000.0

End