EnergyPlus和Dakota集成进行优化的方法

目录

[EnergyPlus和Dakota集成进行优化的方法 1](#_Toc20449)

[概述 2](#_Toc30524)

[1. Dakota生成的params.in文件分析 2](#_Toc17114)

[2. Dakota所需的结果文件results.out文件格式 3](#_Toc3230)

[3. Input\_filter输入过滤器的分析方法 3](#_Toc5226)

[4. EnergyPlus自动运行新的IDF文件 4](#_Toc19699)

[4.1复制生成新生成的输入文件 4](#_Toc30205)

[4.2 EnergyPlus驱动运行文件RunEP.bat文件更改 4](#_Toc31646)

[4.3使用python运行EnergyPlus 5](#_Toc15988)

[5. 采用outputfileter对输出数据进行后处理 5](#_Toc14580)

[6. 设定Dakota的运行文件 5](#_Toc1501)

[7. 案例分析——优化电力尖峰前期和期间的空调设定温度 6](#_Toc6554)

[建筑简介 6](#_Toc8172)

[优化目标 6](#_Toc30087)

[优化变量 7](#_Toc29549)

[优化限制 8](#_Toc5192)

[优化算法 8](#_Toc24061)

[分析结果 8](#_Toc18495)

[8. 案例分析——保温板最优厚度 9](#_Toc3785)

[建筑简介 9](#_Toc22764)

[优化目标 10](#_Toc22577)

[优化变量 10](#_Toc23690)

[优化算法 10](#_Toc14197)

[结果分析 11](#_Toc25475)

# 前言

商业优化工具，如 ModelCenter、modeFRONTIER 和 MATLAB 不适合于开发， 免费软件 GenOpt 不能进行多目标优化， jEPlus二次开发难度较高， Topgui 是 MATLAB 的优化界面， 难以和能耗模拟软件集成。 优化工具Dakota 具有功能强大、免费、开源、可扩展性强和开发难度小等优点， 非常适合于二次开发应用， 本研究选择 Dakota 作为优化工具。

Dakota（Design Analysis Kit for Optimization and Terascale Applications） 是由美国桑迪亚国家实验室开发的能够进行多目标优化、 参数预测、不确定性分析和敏感性分析工具。 Dakota提供的优化方法包括基于梯度的优化法、 无导数局部优化法和全局优化法， Dakota 还提供更为高级的多目标优化和代理模型最小化方法。 Dakota 为开源软件，提供可扩展接口， Dakota 对使用者的专业能力要求极高，需要掌握一定的编程能力。参考其官网https://dakota.sandia.gov。

# EnergyPlus集成Dakota概述

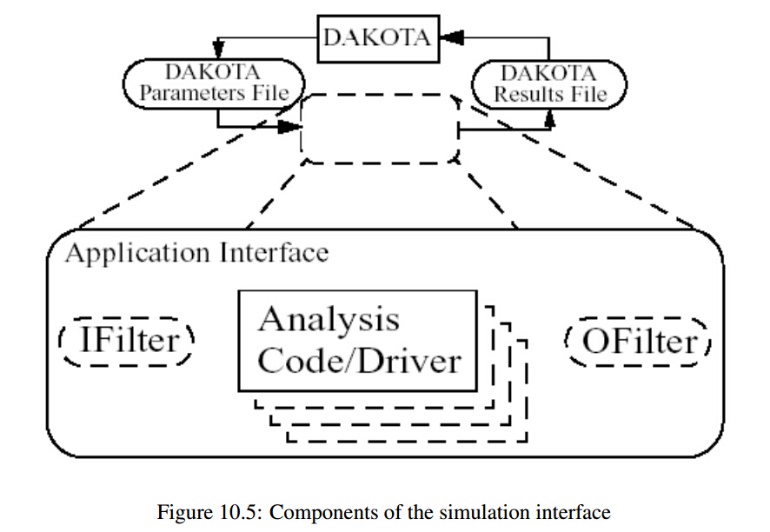


图 1 Dakota和外部软件集成运行数据流程图

如上图所示

1. Dakota能够产生Parameters File，即为params.in文件，Dakota需要的结果文件为results.out。
2. 前处理。计算过程在EnergyPlus中进行，计算使用的输入文件要根据parameters file和模版文件生成。
3. 调用EnergyPlus计算。调用EnergyPlus进行计算要使用RunEPlus.bat，后更改为RunEPlusPython.bat文件。
4. 后处理。EnergyPlus计算结果要进行后处理生成Dakota的结果文件results.out。

上述的b), c)和d)通过DOS批量处理文件统一进行处理。下面对他们分别进行叙述。

# Dakota生成的params.in文件分析

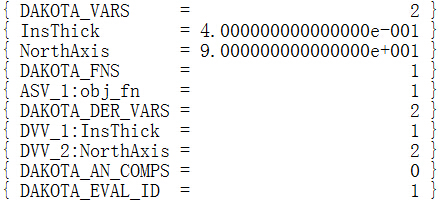


图 2 params.in文件示意图

DAKOTA\_VARS表示的是变量的个数。

InsThick和NorthAxis是设定参数

DAKOTA\_FNS表示目标函数（objective function）的个数。

ASV\_1:obj\_fn表示需要向参数返回的数据，用整数代表。

# Dakota所需的结果文件results.out文件格式

Dakota唯一支持的文件格式如下图所示。下图中黑色字体显示的是目标函数的值（requested function），图中的蓝色字体现实的是梯度值（requested gradients），红色字体显示的是Hessians值（requested Hessians）。如果输出的数据没有按照下图的结果格式现实，程序将会报错。

如下图所示，function的值是不需要有中括号的，其后面可以跟字符串string，表示标注其名字。如果采用标签，标签和函数值之间要用空格隔开（可以是一个或多个空格），标签名中不要有空格。

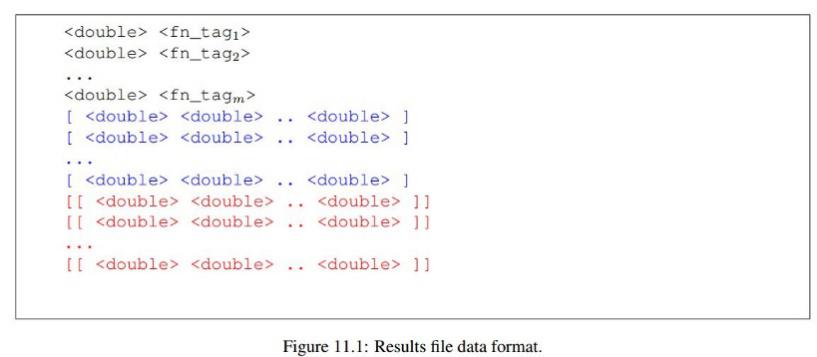


图 3 Dakota支持的结果文件格式

梯度值用中括号，在梯度值的后面不能有标签（tag）。可以使用空格分割这些数据。

Hessian矩阵使用双中括号和其他数据进行分割。同样不能使用tag标签。

数值的部分可以是浮点数或采用科学计数法。如果采用C/C++语言编写的，要将d D等表示指数的值转换为e和E。

# Input\_filter输入过滤器的分析方法

Input\_filter的源代码及使用说明见。

input\_filter的作用是生成新的输入文件。在模版文件中（templatedir）变化的变量采用中括号代替，如{x1}。Input\_filter处理程序将params\_in中的相应数值替换{x1}，从而生成新的输入文件。

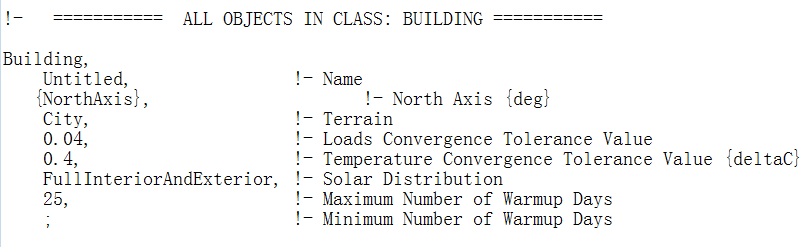


图 4 EnergyPlus模版输入文件举例

图4是EnergyPlus输入文件的一段，图中{NorthAxis}表示会变化的值，input\_filter读取params\_in的数据，如下图5所示，读取NorthAxis等号后的数据，替换模版中的数据生成新的输入文件，新的输入文件举例如图6所示。

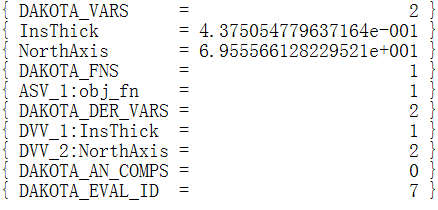


图 5 Dakota的params\_in文件示意图

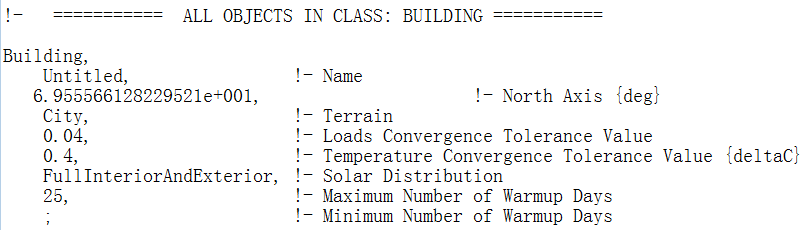


图 6 新的EnergyPlus输入文件示意图

# EnergyPlus自动运行新的IDF文件

Input\_filter, analysis\_driver和output\_filter在同一个文件中组织，命名为simulator\_script\_EP.bat的批量处理文件。

## 4.1复制生成新生成的输入文件

由于新生成的EnergyPlus的输入文件有时要保存在不同的文件夹下面，而EnergyPlus不方便到不同的文件夹中提取输入文件，故将新生成的文件复制为work.idf的文件，每次覆盖work.idf的内容，EnergyPlus运行work.idf的文件。见simulator\_script\_EP.bat下面一句代码：python %~dp0/inputfilter.py %1% in.idf work.idf。

其他内容读代码后的备注。

## 4.2 EnergyPlus驱动运行文件RunEP.bat文件更改

调用EnergyPlus并非直接调用EnergyPlus.exe，也不是调用EP-Launch.exe而是调用RunEPlus.bat文件。批量处理文件RunEPlus.bat用于打开EnergyPlus.exe以及其他文件。更改了输入文件的路径以及天气文件所在的路径。如下图所示，另存为RunEPlusPython.bat（名字自己定）。

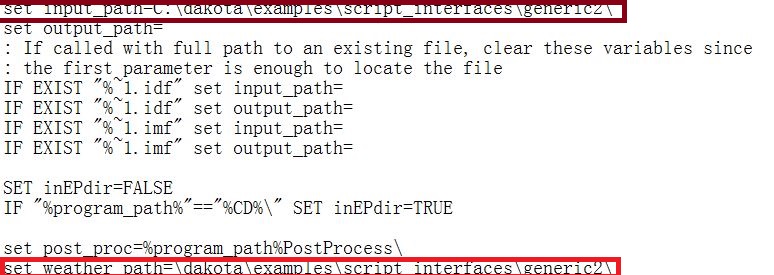


图 7 RunEPlusPython.bat截图

## 4.3使用python运行EnergyPlus

我们使用python的os模块的system函数运行EnergyPlus（见runEPlusScript.py），代码如下：

import os

os.system ("C:/EnergyPlusV8-7-0/RunEPlusPython.bat work beijing");

其中C:/EnergyPlusV8-7-0/RunEPlusPython.bat是我们在上一节设定运行EnergyPlus的脚本，work是work.idf，beijing是beijing.epw。可见向RunEPlusPython.bat传递2个参数，一个是idf文件，另外一个是epw气象文件。

# 采用outputfileter对输出数据进行后处理

将运行输出的结果提取出需要的数据保存到results.tmp，最终保存到results.out文件中。也就是我们保存到results.out即可。writeoutput.py中的下面的代码即用于将能耗费用和PMV写入到results.out。

with open(outputfile,"a+") as output\_file:

output\_file.write(str(energy\_cost)+"\n")

output\_file.write(str(pmvSum))

本案例使用Python提取EnergyPlus输出报告的workTable.csv文件中的用电电费和work.csv文件加中的PMV数据。

# 设定Dakota的运行文件

Dakota运行文件（参考dakota\_EnergyPlus\_cygwin.in）内分为environment, method, variables, interface, responses几个部分，参数设定可以参考官网的reference.pdf和手册manual.pdf。

* Environment，用于设定结果的显示，可不设定。
* Method，用于设定不同的算法。
* Variables，用于设定优化的变量，可以设置多种形式的变量。
* Interface，用于设定和其他程序的交互，这里使用fork即第三方的程序。
* Responses，用于设定dakota接受的参数。

# 案例分析——优化电力尖峰前期和期间的空调设定温度

## 建筑简介

建筑是一栋一层的办公建筑，建筑面积497m2，共被分成5个热区。

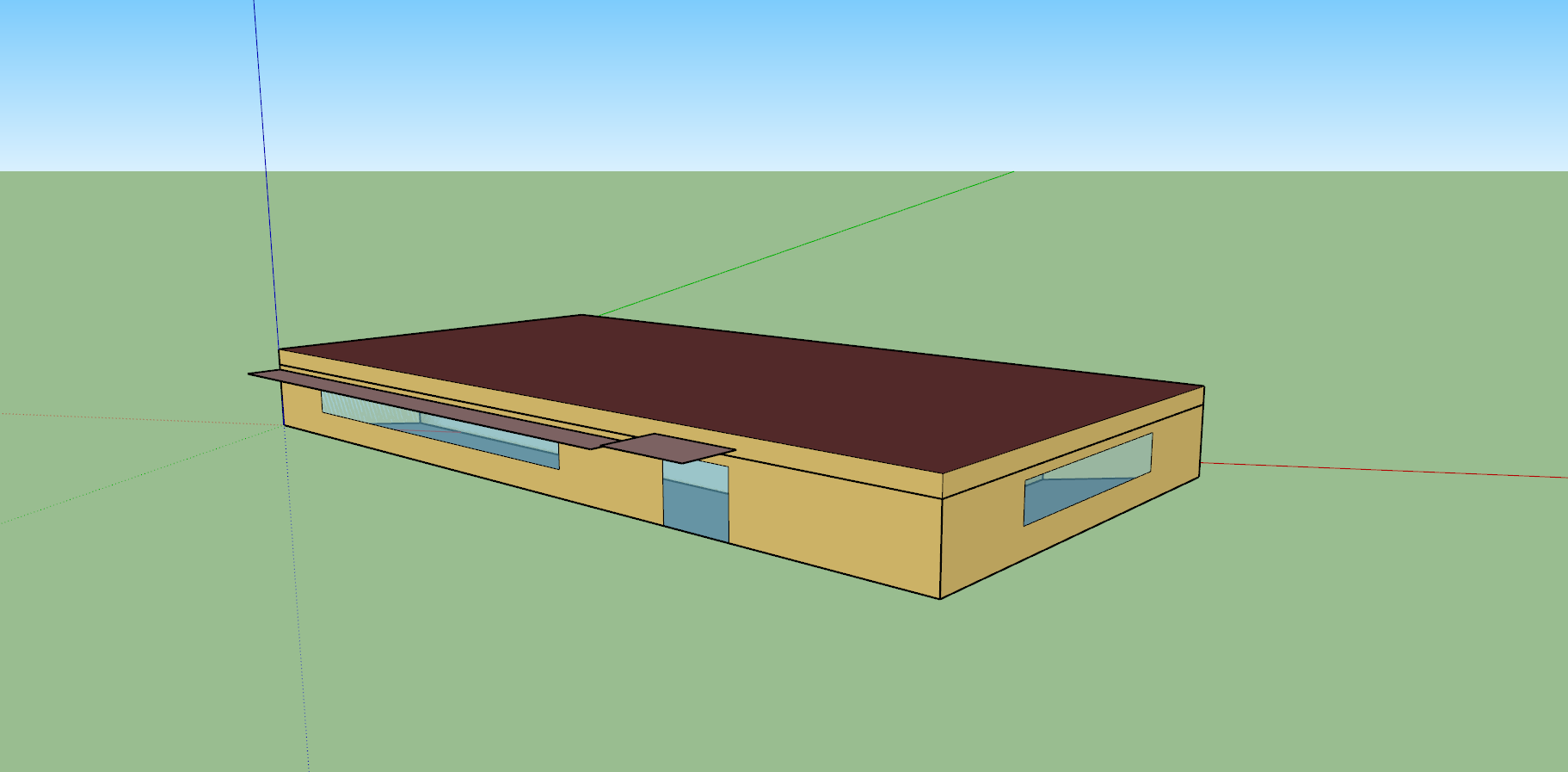


图 8 案例建筑示意图

## 优化目标

优化的目标是24小时内运行电费的最低。不同城市夏季某日的电力价格如图所示。从图中可以看出在尖峰价格区间，价格非常贵。

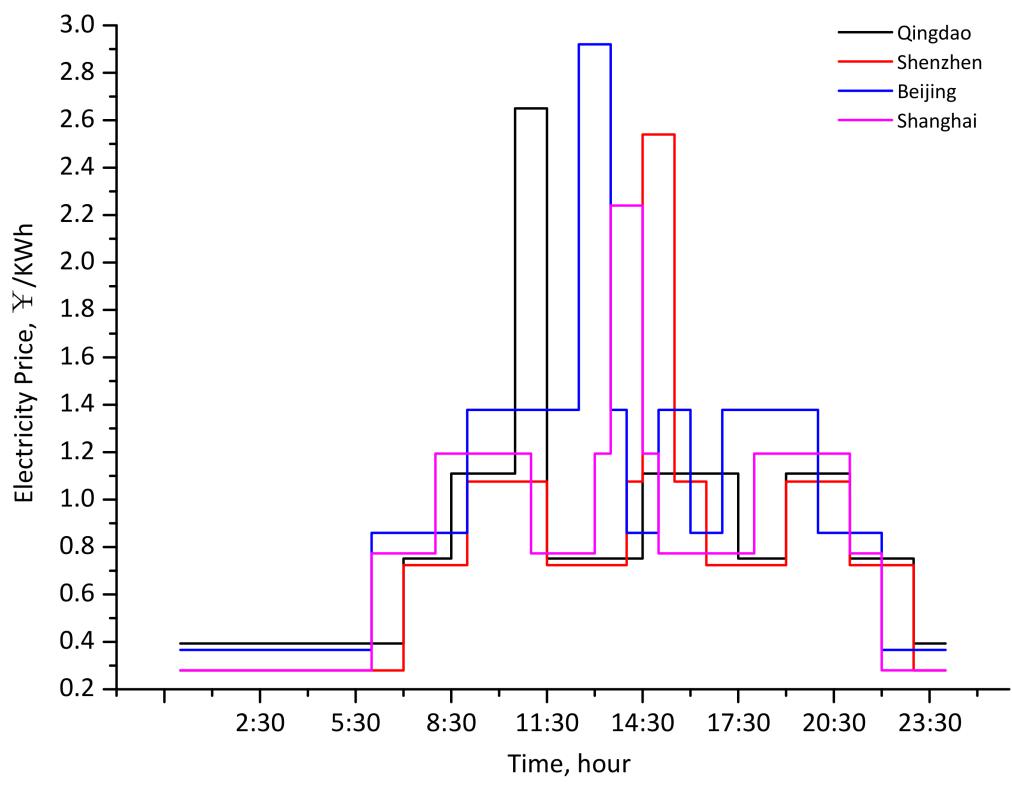


图 9 逐时电力价格

## 优化变量

为了降低运行的价格一个措施是在尖峰电力期间关闭空调，开启风扇。参考下面的优化限制，一旦房间内的PMV超过1，则重新开启空调，关闭风扇。

优化的变量是尖峰电价前1小时和期间的空调温度设定。设定了这段时间的温度设定值，如下图所示。这是在IDF中更改实现的。

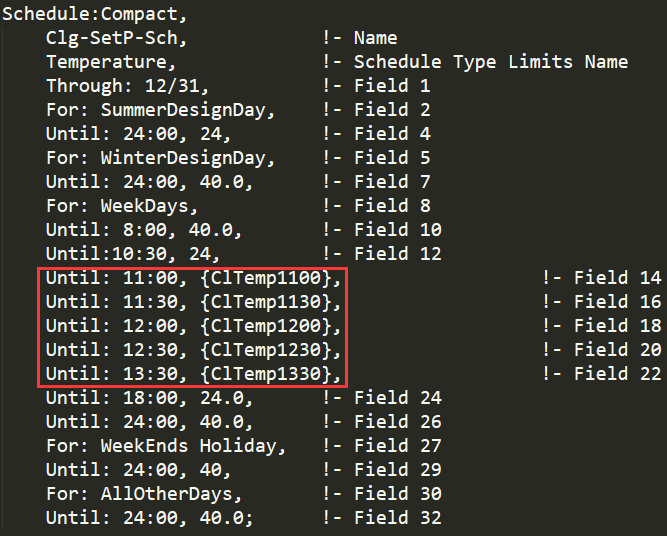


图 10 在template文件夹下设定模板idf中的变量

## 优化限制

房间内的热舒适性是限制条件，即输出的是PMV值，保证PMV控制在[-1,+1]。

## 优化算法

遗传算法。

## 分析结果

优化的最终结果是在尖峰电力来临前，应将建筑进行遇冷，如下图

最终结果是采用最优的措施，**可以节能33.6%，建筑在尖峰电价期间电力负荷降低44.0%。**

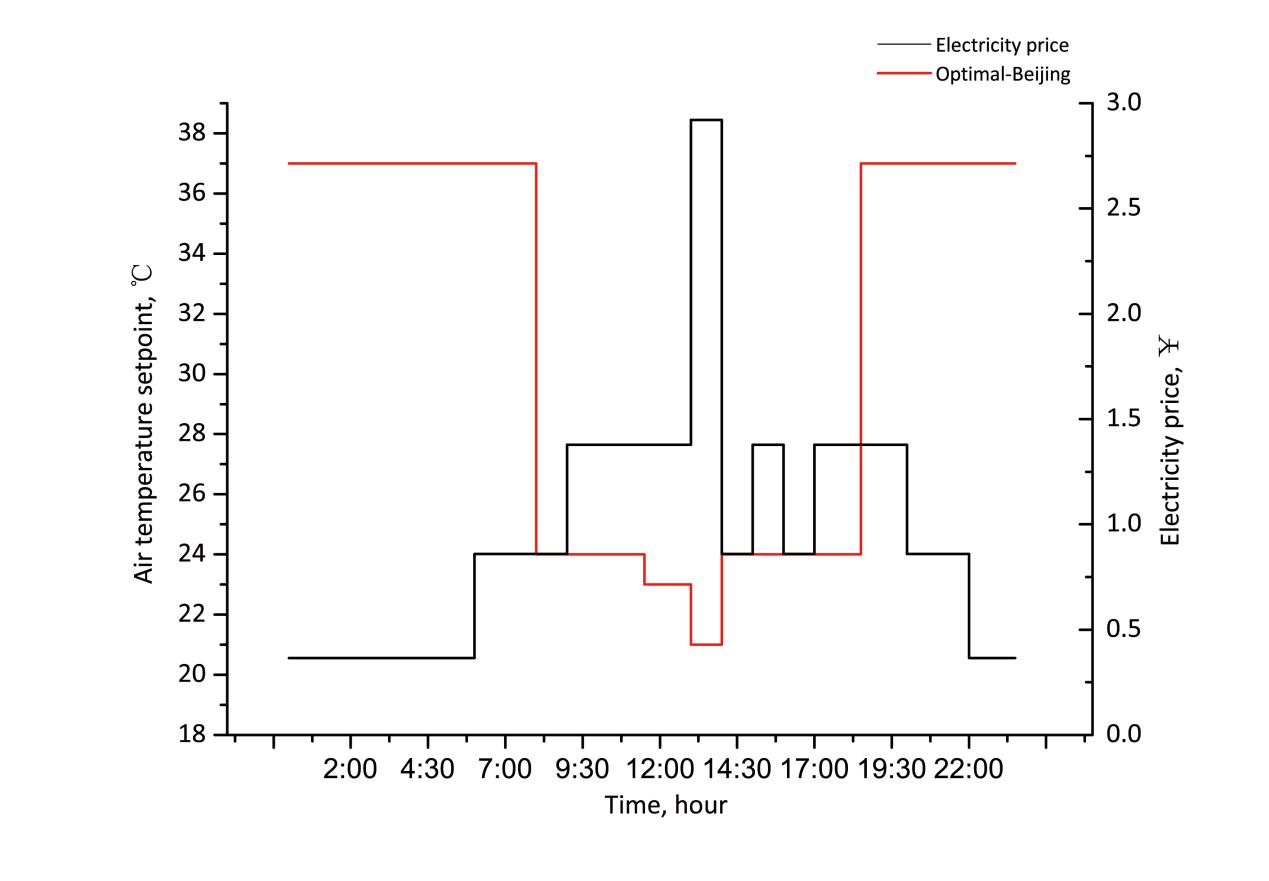


图 11 北京地区夏季某日最优的温度设定值

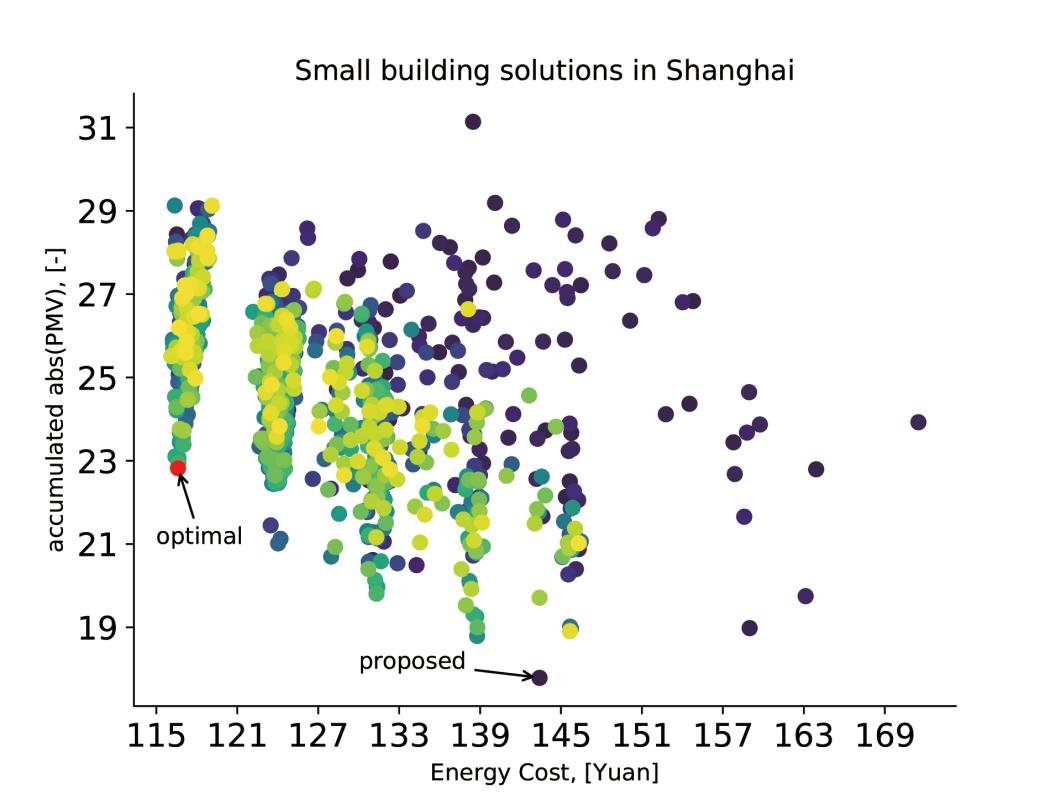


图 12最终所有方案的能源费用和累计PMV

# 案例分析——保温板最优厚度

## 建筑简介

建筑为一层L型办公建筑。

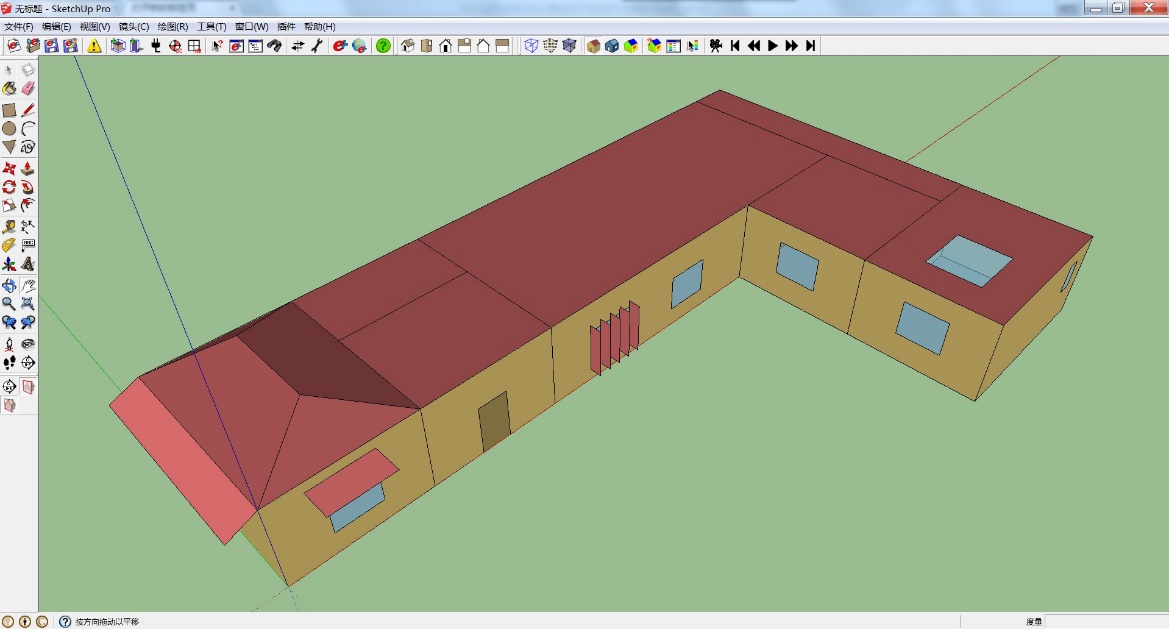


图 13 L型一层建筑SketchUp图

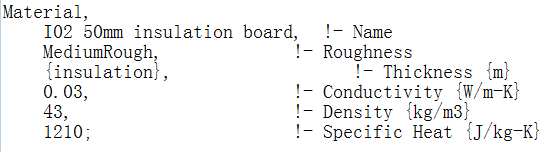
## 优化目标

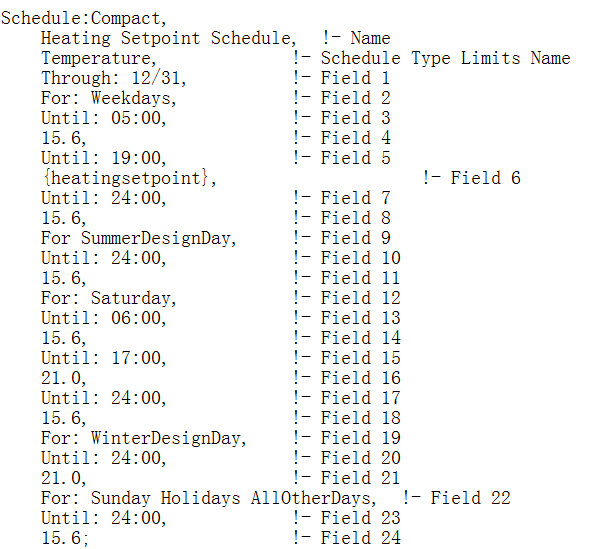
建筑全年用电成本最小。

电费0.55/kWh

## 优化变量

变量1：保温板厚度，变量2冬季供暖温度。

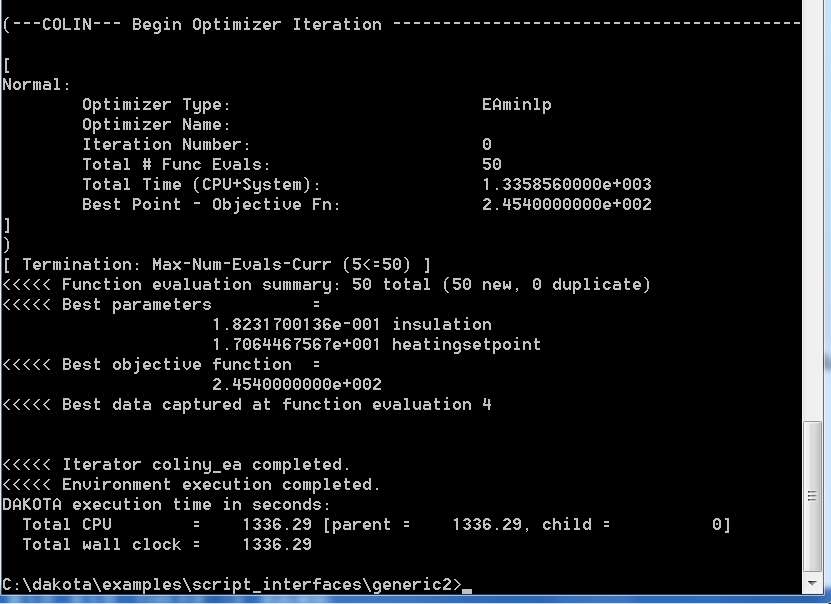




## 优化算法

进化算法（evolutionary Algorithm）。

## 结果分析



从图中可以看出最后时保温板厚度为0.1823m，供暖温度为1.7064℃，此时全年单位面积能耗245.4kWh/m3。

# 作者简介



田志超

东南大学建筑学院博士研究生

微信：zicao423，公众号：领绿建筑性能设计

版权声明：本文由田志超原创，转载请联系tzchao123@qq.com.