统计推断在数模转换系统中的应用

组号：35

姓名：李豪（组长） 学号：5140309185

姓名：王楠 学号：5140309176

摘要：本次数模转换的实验老师一共提供了 400 组数据，由于数据庞大，不可能把每个数据都考虑在内，所以我们需要用统计学推断的方法缩小数据样本。其中电子器件的性能参数构成的非线性特性图形的测绘可以通过先对此类器件进行一定数量的抽样检测，得到一组参数的样本后通过数理统计理论进行统计推断。之后的电子器件的性能参数特性曲线的测绘可以通过这若干个点的测量进行拟合曲线。实现以少量数据反映整体系统特性的效果，从而有效降低工程设计上的成本，提高效率。本文阐述了一种通过统计推断获得电子器件的性能参数特性曲线上若干关键点并拟合出特性曲线的一种方案。

关键词：统计推断，样本，三次样条差值拟合，遗传算法，退火算法。

ABSTRACT :The professor provide 400 statistics for this experiment.Because of the enormous amount of statistics,it's impossible to consider each data. A nonlinear characteristic curve of parameters of an electronic element can be mapped by sampling couples of this type of elements to obtain several samples for statistical inference. Then several key points can be obtained. Testing and mapping the characteristic curve of parameters of every element that is the same type of the sampled ones become testing and fitting these key points. Thus, an enough accurate curve can be obtained with a lower cost and less time is consumed. This paper describes a scheme to obtain a characteristic curve of parameters of an electronic element by fitting several key points of the curve statistical inference from statistical inference.

Key words: statistical inference, sample, curve fitting，GA，SAA

△． 引言

实际生产中，产品往往被要求达到一定的精度，为此需对生产出的产品进行测定。本报告中传感器部件是一种典型的需要多次测量的产品，部分的模拟量函数关系可以通过数学关系直接推导出来，但实际电路的误差，通过数学推导并不可靠。因此研究实测的一组数据之间的函数关系，能够比较准确地反映出输出和输入之间的关系。而对于大规模批量生产，减少测定的数量即可以节约大量生产成本。因此探究如何选定尽可能少的点达到推定整体曲线的误差尽可能小是有重要的现实意义的。。本次统计推断的报告是通过分析大量的数据得出尽可能优的解，来说明统计推断在获得输入输出关系时的作用。

一．模型

为了对本课题展开有效讨论，需建立一个数学模型，对问题的某些方面进行必要的描述和限定。

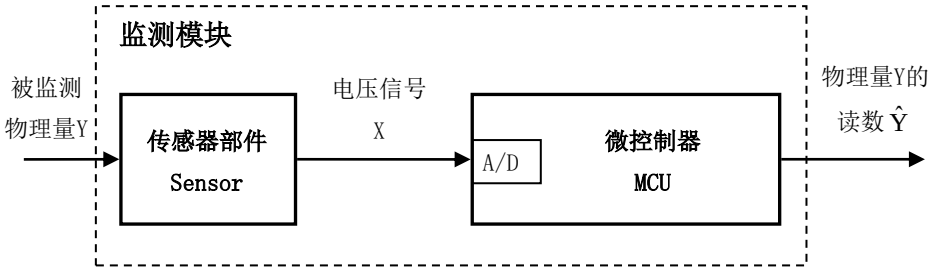


图 1 监测模块组成框图

监测模块的组成框图如图 1。其中，传感器部件（包含传感器元件及必要的放大电路、调理电路等）的特性是我们关注的重点。传感器部件监测的对象物理量以符号 Y 表示；传感部件的输出电压信号用符号 X 表示，该电压经模数转换器（ADC）成为数字编码，并能被微处理器程序所读取和处理，获得信号 $\hat{Y}$ 作为 Y 的读数（监测模块对 Y 的估测值）。

所谓传感特性校准，就是针对某一特定传感部件个体，通过有限次测定，估计其 Y 值与 X 值间一一对应的特性关系的过程。数学上可认为是确定适用于该个体的估测函数 $\hat{y}=f(x)$ 的过程，其中 x 是 X 的取值，$\hat{y}$ 是对应 Y 的估测值。

考虑实际工程中该监测模块的应用需求，同时为便于在本课题中开展讨论，我们将问题限于 X 为离散取值的情况，规定

$$X \in \{x_1, x_2, x_3, ..., x_{50}, x_{51}\} = \{5.0, 5.1, 5.2, ..., 9.9, 10.0\}$$

相应的 Y 估测值记为 $\hat{y}_i = f(x_i)$，Y 实测值记为 y_i，$i = 1, 2, 3, ..., 50, 51$。

1.1 传感部件特性

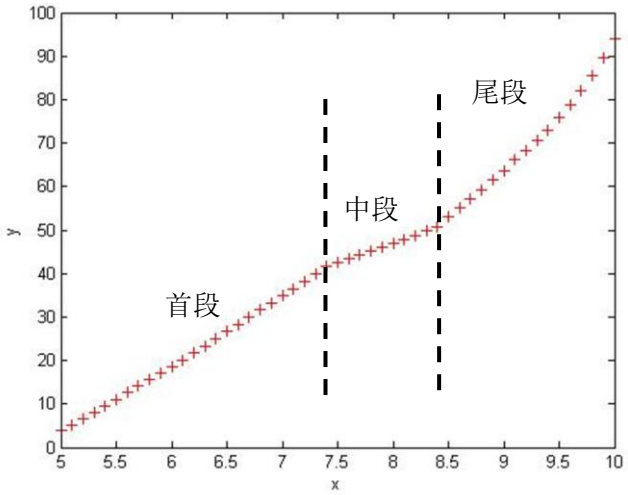


图 2 传感特性图示

一个传感部件个体的输入输出特性大致如图 2 所示，有以下主要特征：

- Y 取值随 X 取值的增大而单调递增；
- X 取值在[5.0,10.0]区间内，Y 取值在[0,100]区间内；

- 不同个体的特性曲线形态相似但两两相异；
- 特性曲线按斜率变化大致可以区分为首段、中段、尾段三部分，中段的平均斜率小于首段和尾段；
- 首段、中段、尾段单独都不是完全线性的，且不同个体的弯曲形态有随机性差异；
- 不同个体的中段起点位置、终点位置有随机性差异。

为进一步说明情况，图 3 对比展示了四个不同样品个体的特性曲线图示。

1.2 标准样本数据库

前期已经通过试验性小批量生产，制造了一批传感部件样品，并通过实验测定了每个样品的特性数值。这可以作为本课题的统计学研究样本。数据被绘制成表格，称为本课题的“标准样本数据库”。

该表格以 CSV 格式制作为电子文件。表格中奇数行存放的取值，偶数行存放对应的取值。第 $2i-1$ 行存放第 i 个样本的 X 数值，第 $2i$ 行相应列存放对应的实测 Y 数值。

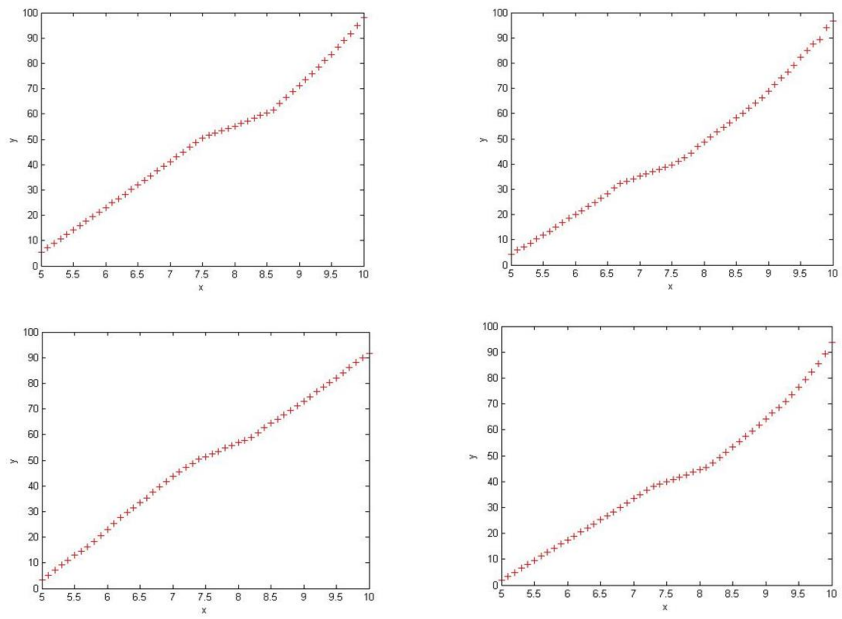


图 3 四个不同样本个体特性图示对比

1.3 成本计算

为评估和比较不同的校准方案，特制定以下成本计算规则。

- 单点定标误差成本

$$s_{i,j}=\begin{cases}0 & \text{if } \left|\hat{y}_{i,j}-y_{i,j}\right|\le 0.4\\ 0.1 & \text{if } 0.4<\left|\hat{y}_{i,j}-y_{i,j}\right|\le 0.6\\ 0.7 & \text{if } 0.6<\left|\hat{y}_{i,j}-y_{i,j}\right|\le 0.8\\ 0.9 & \text{if } 0.8<\left|\hat{y}_{i,j}-y_{i,j}\right|\le 1\\ 1.5 & \text{if } 1<\left|\hat{y}_{i,j}-y_{i,j}\right|\le 2\\ 6 & \text{if } 2<\left|\hat{y}_{i,j}-y_{i,j}\right|\le 3\\ 12 & \text{if } 3<\left|\hat{y}_{i,j}-y_{i,j}\right|\le 5\\ 25 & \text{if } \left|\hat{y}_{i,j}-y_{i,j}\right|>5\end{cases} \tag{1}$$

单点定标误差的成本按式（1）计算，其中 $y_{i,j}$ 表示第 i 个样本之第 j 点 Y 的实测值，$\hat{y}_{i,j}$ 表示定标后得到的估测值（读数），该点的相应误差成本以符号 $s_{i,j}$ 记。

- 单点测定成本

实施一次单点测定的成本以符号 q 记。本课题指定 q=12。

- 某一样本个体的定标成本

$$S_i = \sum_{j=1}^{51} s_{i,j} + q \cdot n_i \qquad (2)$$

对样本 i 总的定标成本按式（2）计算，式中 n_i 表示对该样本个体定标过程中的单点测定次数。

- 校准方案总成本

按式（3）计算评估校准方案的总成本，即使用该校准方案对标准样本库中每个样本个体逐一定标，取所有样本个体的定标成本的统计平均。

$$C = \frac{1}{M} \sum_{i=1}^{M} S_i \qquad (3)$$

总成本较低的校准方案，认定为较优方案。

注：模型背景引用“**课程设计课题和要求**”中模型创建背景。

二．任务及设计思路

2.1 任务概述

假定有某型投入批量试生产的电子产品，其内部有一个模块，功能是监测某项与外部环境有关的物理量（可能是温度、压力、光强等）。该监测模块中传感器部件的输入输出特性呈明显的非线性。本课题要求为该模块的批量生产设计一种成本合理的传感特性校准（定标工序）方案。

传感器部件监测的对象物理量以符号 Y 表示；传感部件的输出电压信号用符号 X 表示，该电压经模数转换器（ADC）成为数字编码，并能被微处理器程序所读取和处理，获得信号 $\hat{Y}$ 作为 Y 的读数（监测模块对 Y 的估测值）。

2.2 设计思路

基本任务中主要有三个方面的问题需要考虑：

（1）尽量减少测定点，节约成本。

（2）保证估算的合理性，合理取样本。

（3）估算的越多则误差越大。

这是一个组合优化问题，但备选的组合方案数量较大，目前的计算机尚难以对以上问题采取穷举的方式进行求解，故采用恰当的算法进行优化较为合理。

三．数学模型

这个问题抽象为一个数学问题就是在一个部件特性曲线未知的前提下，求取 n 个电压值进行测量，得出 6 个样本数据点，通过这 n 个点拟合出部件的特性曲线且测量成本与误差成本之和要尽量的小。

四．具体实现方案

4.1 三次样条插值法

样条插值是使用一种名为样条的特殊分段多项式进行插值的形式。由于样条插值可以使用低阶多项式样条实现较小的插值误差，这样就避免了使用高阶多项式所出现的龙格现象。

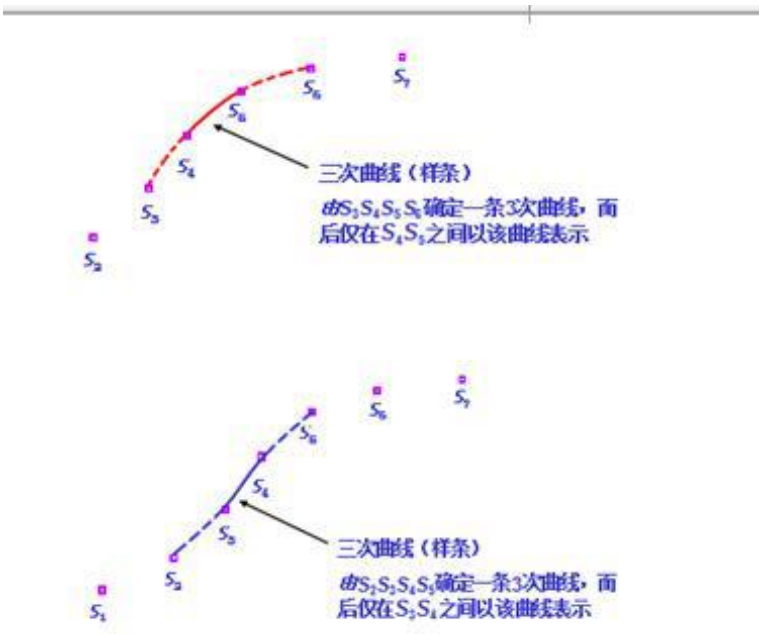
三次样条插值（简称 Spline 插值）是通过一系列形值点的一条光滑曲线，数学上通过求解三弯矩方程组得出曲线函数组的过程。

三次样条函数:

定义:函数 S(x)∈C2[a,b] ，且在每个小区间[xj,xj+1]上是三次多项式，其中 a =x0 <x1<...< xn=b 是给定节点，则称 S(x)是节点 x0,x1,...xn 上的三次样条函数。

若在节点 xj 上给定函数值 Yj= f(Xj).(j=0,1,...,n)，并成立 S(xj)=yj(j=0,1,...,n)，则称 S(x)为三次样条插值函数。

实际计算时还需要引入边界条件才能完成计算。边界通常有自然边界（边界点的二阶导为 0），夹持边界（边界点导数给定），非扭结边界（使两端点的三阶导与这两端点的邻近点的三阶导相等）。一般的计算方法书上都没有说明非扭结边界的定义，但数值计算软件如 Matlab 都把非扭结边界条件作为默认的边界条件。



4.2 遗传算法

遗传算法的概念：遗传算法（Genetic Algorithm）是模拟达尔文生物进化论的自然选择和遗传学机理的生物进化过程的计算模型，是一种通过模拟自然进化过程搜索最优解的方法。

4.2.1 遗传算法中的几个基本概念：

遗传算法：一种基于生物自然选择与遗传机理的随机搜索算法。

种群：遗传算法从样本子空间或通过上一代的选择得到的一组随机产生的解空间。

染色体：种群中的每个个体，每个染色体都是问题的一个解。染色体是一串符号，比如一个二进制字符串。

遗传：即染色体在后续迭代中不断进化。

适值：每一代中用来测量染色体好坏的值。

后代：生成的后一代染色体。

4.2.2 遗传算法的基本运算过程如下：

a)初始化：设置进化代数计数器 t=0，设置最大进化代数 T，随机生成 M 个个体作为初始群体 P(0)。

b)个体评价：计算群体 P(t)中各个个体的适应度。

c)选择运算:将选择算子作用于群体。选择的目的是把优化的个体直接遗传到下一代或通过配对交叉产生新的个体再遗传到下一代。选择操作是建立在群体中个体的适应度评估基础上的。

d)交叉运算：将交叉算子作用于群体。遗传算法中起核心作用的就是交叉算子。

e)变异运算：将变异算子作用于群体。即是对群体中的个体串的某些基因座上的基因值作变动。

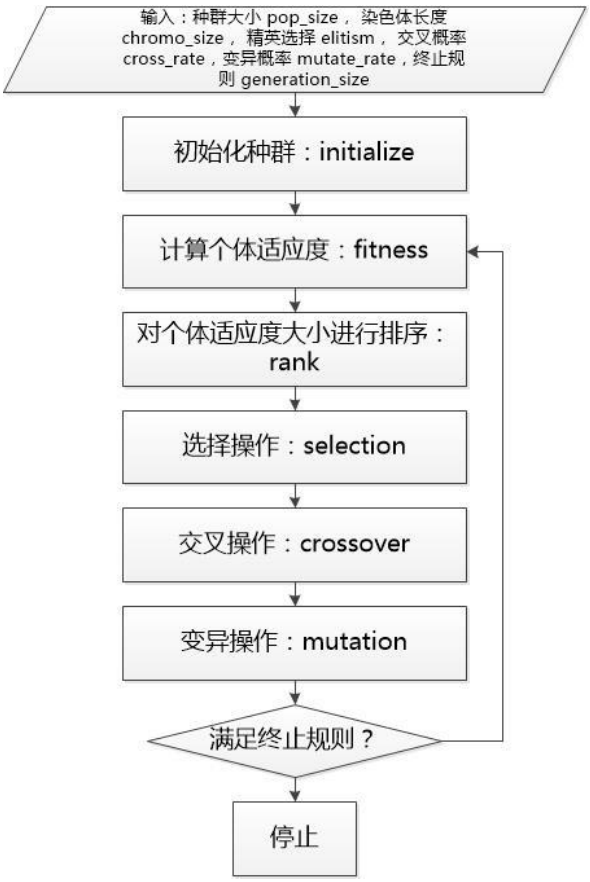
群体 P(t)经过选择、交叉、变异运算之后得到下一代群体 P(t+1)。

f)终止条件判断:若 t=T,则以进化过程中所得到的具有最大适应度个体作为最优解输出，终止计算。

4.2.3 遗传算法流程

遗传算法分为 1 编码　2 解码　3 初始化　4 选择操作　5 交叉操作　6 编译操作

遗传算法流程图如下图所示：



4.3 模拟退火算法

模拟退火算法(Simulated Annealing，SA)最早的思想是由 N. Metropolis[1]等人于 1953 年提出。1983 年,S. Kirkpatrick 等成功地将退火思想引入到组合优化领域。它是基于 Monte-Carlo 迭代求解策略的一种随机寻优算法，其出发点是基于物理中固体物质的退火过程与一般组合优化问题之间的相似性。模拟退火算法从某一较高初温出发，伴随温度参数的不断下降,结合概率突跳特性在解空间中随机寻找目标函数的全局最优解，即在局部最优解能概率性地跳出并最终趋于全局最优。模拟退火算法是通过赋予搜索过程一种时变且最终趋于零的概率突跳性，从而可有效避免陷入局部极小并最终趋于全局最优的串行结构的优化算法。

这里编者借鉴一个形象的描述来粗略讲述模拟退火算法，模拟退火算法本质上是爬山算法，爬山算法是兔子朝着比现在高的地方跳去。它找到了不远处的最高山峰。但是这座山不一定是珠穆朗玛峰。这就是爬山算法，它不能保证局部最优值就是全局最优值。而模拟退火则是兔子喝醉了。它随机地跳了很长时间。这期间，它可能走向高处，也可能踏入平地。但是，它渐渐清醒了并朝最高方向跳去。这就是模拟退火算法。

4.3.1 一般步骤

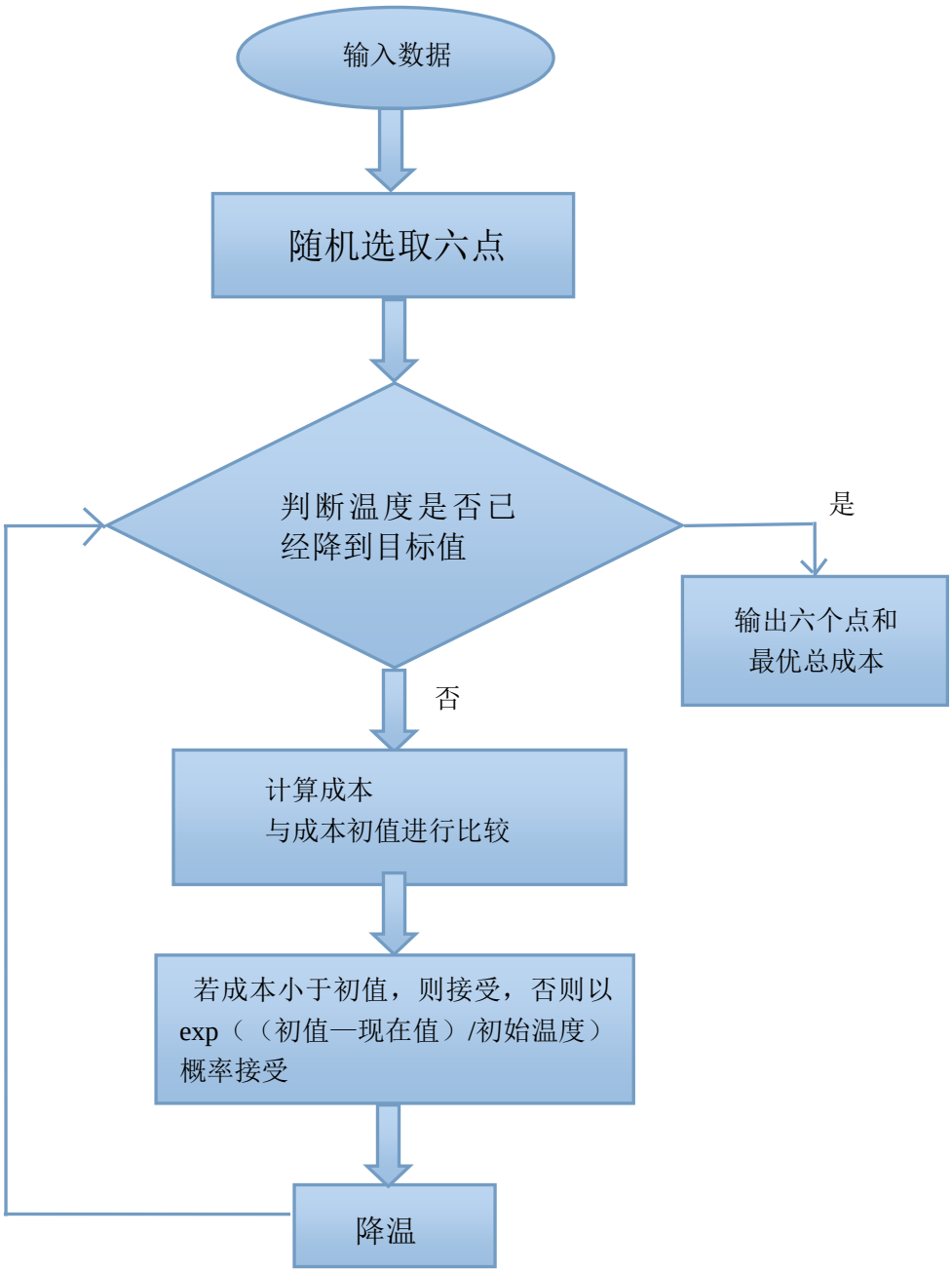
它可以分解为解空间、目标函数和初始解三部分。

（1）初始化：初始温度 T(充分大)，初始解状态 S(是算法迭代的起点)， 每 个 T 值的迭代次数 L

（2）对 k=1，……，L 做第(3)至第 6 步：产生新解 S′

（3）（计算增量 Δt′ =C(S′)-C(S)，其中 C(S)为评价函数

（4）若Δt′<0 则接受 S′作为新的当前解，否则以概率 exp(-Δt′/T)接受 S′作为新的当前解.
（5）如果满足终止条件则输出当前解作为最优解，结束程序。终止条件通常取为连续若干个新解都没有被接受时终止算法。
（6）T 逐渐减少，且 T->0，然后转第 2 步。
以下为本次试验流程：



4.3.2 算法分析

由于选取的降温幅度较小，同时为了减少时间成本，温度的设定我们选择设置初始温度为 10℃，末温为 0.01℃，降温幅度为 0.99，仅通过降温，可进行约 400 组实验数据的测定，这个数据量可以充分找到最优解。

在对比实验中改变初始温度，可以更改初始温度，使实验数据测定量增加，来探究能否找出更有效的取点，并且使得得到的结果更优。

有时在选点后，若不进行进一步处理可能会产生局部最优解。每次产生新的 6 个点时，我们处理的机制是随机替换掉最优取点中的某一个，这样在 400 组数据取点的累积过程中，可以使得总体上保持平均，不会出现局部最优解的情况，而且可以使每次取点的成本逐渐降低，更容易找到最优解。但是如果每次均在该点附近取点进行替换，则很容易造成局部最优解，用一个随机的点进行替换，从某种程度上降低了局部最优解的概率。此外，为了证明最后得出模拟退火算法不易产生局部最优解，后面的分析中引入了设定一个新的初始值再进行实验，从而进一步证明。

五．实验结果及分析

5.1 实验结果

编号	运行选取的六个点	成本	运行时间
1	**3 11 21 31 41 49**	**95.43**	**83.608484**
2	**3 12 22 32 43 50**	**95.18**	**86.550097**
3	**3 11 22 31 43 50**	**94.92**	**87.083758**
4	**3 12 22 31 43 50**	**94.90**	**91.346134**
5	**3 12 22 31 43 50**	**94.90**	**101.893316**
6	**3 11 22 31 43 50**	**94.92**	**102.619204**
7	**2 10 21 30 40 49**	**95.49**	**101.813536**
8	**3 12 22 32 43 50**	**95.18**	**103.099500**
9	**3 11 22 31 43 49**	**95.24**	**103.151436**
10	**3 12 22 31 43 50**	**94.90**	**101.356420**

5.2 实验结果分析

由运行的结果可以得出，样条插值 6 点模拟退火的运行成本平均值为 95.106，最低大概为 94.90，最高大概为 95.49；运行时间最少大概为 83.608484 秒，最多大概为 103.151436 秒。

由实验结果可以看出，6 个点大概可以较为精确的模拟出监测模块中传感器部件的输入输出特性，且用时较短，大概不到 2 分钟就能退火完成。这次实验统一要求选取 6 个点，可是 6 个点不一定是最优的取点个数，6 个点和 7 个点得出结果的区别在哪里呢？增加初始温度是否可以更加降低成本呢？为此，我们又讨论了选取 7 个点和初始温度为 100 度时的情况。

5.3.1 选取七个点实验数据

编号	运行选取的七个点	成本	运行时间
1	3 10 20 27 34 44 50	95.24	121.464828
2	2 9 20 27 34 44 49	95.77	118.827154
3	2 9 20 27 35 45 50	95.79	117.729270

5.3.2 结果分析

由表与表对比可知，用 6 个点运行模拟退火算法，总成本更低，普遍成本比 7 个点低一些，所耗费的时间也更少一些，所以 6 个点效果更优。

结果分析：由于曲线呈分段线性特性，且一共分为三段，由两点确定一条直线的原理，一共确定 6 个点就能较为精确的将曲线趋势表现出来；此外，减少一个测试点能较大的降低总成本，故 6 个点从原理上成本可以优于 7 个点。

5.4.1 选取初始温度为 100 度

编号	运行选取的六个点	成本	运行时间
1	3 11 22 31 43 50	94.92	159.393302
2	2 10 21 30 40 49	95.49	158.466887
3	3 10 21 31 41 50	95.47	157.082305

5.4.2 结果分析

由表、表与表对比可知：100 度时，随着退火次数的增加，找到的解成本虽然会低一点点，但是普遍比较接近，可是所花费时间要多一些，故对成本要求较高时可以考虑进行适当升温处理。

5.5 原本我们组尝试进行了使用遗传算法得出相应的实验结果，但是多次调试后实验结果仍然不理想，难以调试成功，且运行较慢，故最后我们舍弃了遗传算法，仅用退火算法来分析。

六 实验总结

通过实验本身以及上述所有对比试验，我们发现，在当前的评价函数基础上，6 个点是成本最低的取点个数，三次样条插值法是最合适的拟合方法，从达到最优解同时节省时间两方面考虑，不用对温度进行升温或者降温操作，且该算法不易陷入局部最优解中，无须固定初始值，或者加入恒温退火过程。

故采用模拟退火算法选择 6 个点取样，运用三次样条插值法进行数据拟合，是最节省成本，最高效的定标方式。

七．参考文献：

1. 袁琰 课堂 PPT 资料

2. 百度百科，退火算法及遗传算法

http://www.baidu.com/link?url=yYzpwRYyNAyiJpsxA6RoI4wHwAzRaC9Vx8rNDBpPY3N24cwcTYpZG5tTr2M71xY560_V8GENt7EkEl8yt6Eu6q&wd=&eqid=a26cd53f0009381f000000035662f64f

3. 往届报告

4. MATLAB 教程　北京航空航天大学出版社

八．附录（部分程序代码）

```
data=csvread('c:/20150915dataform.csv',0,0,[0 0 799 50]);%读入数据表中的数据
x=data(1:2:end,1:end);
y=data(2:2:end,1:end);%将表格中的 x,y 分别取出
rand=randperm(51);%随机打乱 51 个点
choice=sort(rand(1:6)); %随机取 6 个点，并进行排序
min=0;%总分数
money=0;%当次成本
```

```
savemoney=0;%上一个成本
bestchoice=choice;%最优解
origin_temperature=10;%初始温度
last_temperature=0.01;%末温度
num=0;%计数模拟的次数


tic;%
while origin_temperature>last_temperature
num=num+1;
left=setdiff(rand,choice);%寻求 rand,choice 差集
tmp1=left(randperm(45));%再次打乱顺序
tmp2=randperm(6);
tmpchoice=choice;
tmpchoice(1,tmp2(1))=tmp1(1,tmp2(1)+1);
tmpchoice=sort(tmpchoice);%随机替换 6 个数中的一个数字




%以下代码取自老师给的测试函数
my_answer=tmpchoice;%把你的选点组合填写在此
my_answer_n=size(my_answer,2);
% 标准样本原始数据读入
minput=dlmread('c:/20150915dataform.csv');
[M,N]=size(minput);
nsample=M/2; npoint=N;
x=zeros(nsample,npoint);
y0=zeros(nsample,npoint);
y1=zeros(nsample,npoint);
for i=1:nsample
x(i,:)=minput(2*i-1,:);
y0(i,:)=minput(2*i,:);
end
my_answer_gene=zeros(1,npoint);
my_answer_gene(my_answer)=1;
% 定标计算
index_temp=logical(my_answer_gene);
x_optimal=x(:,index_temp);
y0_optimal=y0(:,index_temp);
for j=1:nsample
% 通过调用 mycurvefitting 函数，实现三次样条插值拟合
y1(j,:)=mycurvefitting(x_optimal(j,:),y0_optimal(j,:));
```

```
end
% 成本计算
Q=12;
errabs=abs(y0-y1);
le0_4=(errabs<=0.4);
le0_6=(errabs<=0.6);
le0_8=(errabs<=0.8);
le1_0=(errabs<=1);
le2_0=(errabs<=2);
le3_0=(errabs<=3);
le5_0=(errabs<=5);
g5_0=(errabs>5);
sij=0.1*(le0_6-le0_4)+0.7*(le0_8-le0_6)+0.9*(le1_0-le0_8)+1.5*(le2_0-le1_0)+6*(le3_0-le2_0)+12*(le5
_0-le3_0)+25*g5_0;
si=sum(sij,2)+Q*ones(nsample,1)*my_answer_n;
money=sum(si)/nsample;
% 显示结果
fprintf('\n 经计算，你的答案对应的总体成本为%5.2f\n',money);
%以上代码取自老师给的测试函数



if num==1 %第一次计算成本
min=money;
savemoney=money;
bestchoice=tmpchoice;
choice=tmpchoice;
elseif money<min %成本小于上一次
fprintf('\n 经计算，当前最优总成本为%5.2f\n',money);
min=money;
savemoney=money;
bestchoice=tmpchoice;
choice=tmpchoice;
elseif rand<exp((savemoney-money)/origin_temperature)%决定点是否变化
savemoney=money;
choice=tmpchoice;
end
origin_temperature=origin_temperature*0.99;%降温
end
fprintf('\n 经计算，最终最优总成本为%5.2f\n',min);
bestchoice %输出最优取点方案
Toc;
```

```
function y1 = mycurvefitting(x_premea,y0_premea)
x = [5.0:0.1:10.0];
y1=interp1(x_premea,y0_premea,x,'spline');


end
```