

# 统计推断在数模转换系统中的应用

组号：016 姓名：彭勋 学号：5140309395 ， 姓名：阳展韬 学号：5140309401

**摘要：**本文是上海交通大学电子信息与电气工程学院课程设计《统计推断在模数，数模转换系统中的应用》的课程论文。本文通过多项式拟合的方式，采用模拟退火算法，运用 matlab 工具，对标准库中的 400 组数据进行分析，从而选出最合适的取点方案，使得定标过程中定标精度和定标成本得到兼顾，最终给出了合适的定标方式<sup>[1]</sup>。

**关键字：**模拟退火算法，多项式拟合

## 1 引言

### 1.1 课题背景

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<sup>[2]</sup>。

### 1.2 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。

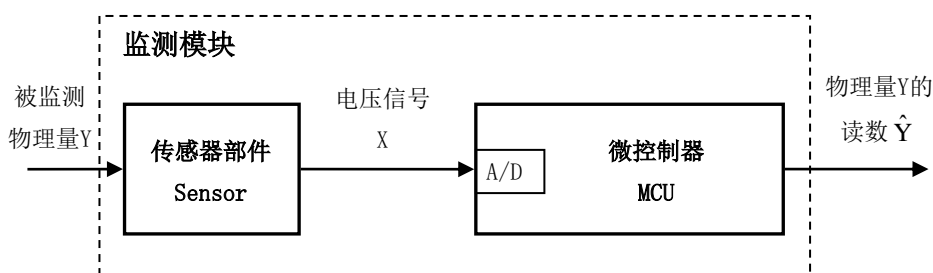


图 1 监测模块组成框图

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

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

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

### 1.3 样本分析

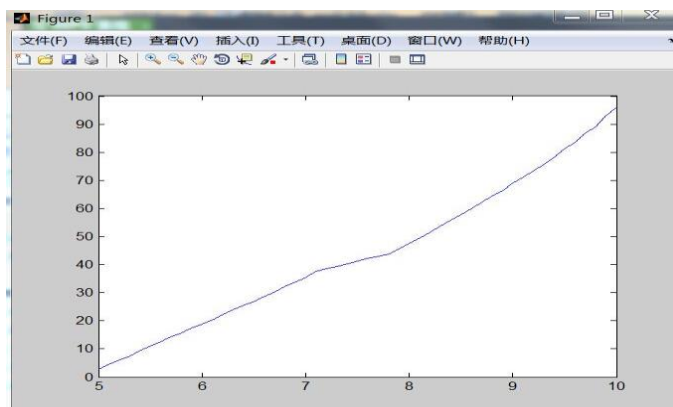


图 2 模块传感部件特性图示

由此可见，样本的基本特性如下：

1. Y 值随 X 值的增大而单调增大。
2. 图像大致可分为三段，前段和后段 Y 值增大的幅度较大，而中段 Y 值增大的幅度较小。
3. Y 值的取值范围为  $[0, 100]$ ，X 值的取值范围为  $[5, 10]$ 。
4. 不同的样本中段的起始位置和终止位置存在个体差异。
5. 不同样本各段的增幅也存在一定的个体差异。

### 1.4 标准样本数据库

在前期的研究中，对 400 个样本进行了较为详细的测量，每个样本测得了 51 个值。得到了 400 组数据，每一组数据有 51 个值，本课题将完全基于这 400 组数据。

### 1.5 评价函数

为了对测量的方案的准确度和成本两方面进行合理的评估，我们采用以下的评价函数：

#### 1. 单点测量成本

实施一次单点测量的成本为 12。

#### 2. 单点定标误差成本

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

#### 3. 单个样本个体的定标成本

$$S_i = \sum_{j=1}^{51} s_{i,j} + 12N_i \quad (1-2)$$

#### 4.定标总成本

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

## 2 问题解决方案

### 2.1 拟合方案选取

考虑到传感部件的特性，我们小组采用了多项式拟合的方式来拟合实验数据，多项式拟合的表达式如下所示：

$$y=p_0x^0+p_1x^1+p_2x^2+\dots+p_nx^n \quad (2-1)$$

由表达式容易看出，低次多项式是高次多项式高次项取 0 的特例。所以，拟合时采用的表达式次数越高，拟合效果越好，然而多项式次数越高，测量次数就越大，会产生更多的测量成本，所以，应当综合考虑测量成本和误差大小两方面的因素。具体采用几次项多项式进行拟合，我们将会在后面讨论说明。

### 2.2 插值方案选取

我们小组采用多项式插值的方式，即将拟合得到的函数在某一非测量点的值作为该点的值。

### 2.3 特征点选取算法（模拟退火算法）<sup>[3]</sup>

由于穷举法在大数据的生产中不适用，我们必须寻求另外的选取特征点的算法。最终我们小组选取了模拟退火算法，下面对模拟退火算法进行详细的介绍：

模拟退火算法主要模拟了固体降温的原理，先将固体加热到充分的高温，再使其冷却。加温时，固体内部的粒子内能增大，呈无序状态。冷却时粒子渐趋有序，最后在常温时达到基态，内能减为最小。

模拟退火算法借鉴了这样的一个过程，它会在自变量不断变化时，接受更优的函数值，并且会以一个逐渐减小的概率去接受更差的函数值，这样做是为了防止模拟退火算法陷入一个局部的最优值，而不能找到全局的最优解。

在本课题中，由于我们要探究取几个点和用几次项的多项式拟合最为合适，所以这里设定为测量  $n$  个点的值，接下来我们要用模拟退火算法来找到最优的测量方案，下图是模拟退火算法的具体实现思想和算法流程框图。

1. 选取最前面的  $n$  个点作为初始的测量方案，将成本计算函数作为评价函数，算出其成本值  $c_0$ 。

2. 将初始的温度  $T$  设定为 1000，每次循环降温 0.1，换句话说，程序被设定为循环 10000 次。

3. 每次循环，将 1-51 这 51 个数据重新随机排列，选取最前面的  $n$  个点，作为新的取点方案，即从 51 个数据点中随机抽取  $n$  个点作为新的取点方案，算出其成本值  $c_1$ ，并算出  $\Delta t'$ 。

$$\Delta t' = c_1 - c_0 \quad (2-2)$$

#### 4. 算出概率 $p$

$$p = \exp(-\Delta t' / T) \quad (2-3)$$

5. 若新方案的成本值低于原来的成本值，则接受这个方案，丢掉原来的方案。若高于原来的成本值，则以  $p$  的概率接受这个方案。具体实现可采用随机数的方法，由计算

机产生一个随机数，若该数小于  $\exp(-\Delta t' / T)$ ，则接受这个较差的解，否则丢弃。

6. T 降低 0.1，重复 3-6 步。

\*7.（注：这一点并不是模拟退火算法的内容，是我们组对于模拟退火算法的一点小小的改进）由于模拟退火算法会在一定概率内接受错误答案，所以我们将运行途中最佳的答案记录下来，即在第 5 步中，若成本值低于原来的成本值，接受新方案的同时，将新方案另外储存起来。与模拟退火算法最后得出的答案进行比较，选取较小的一个。

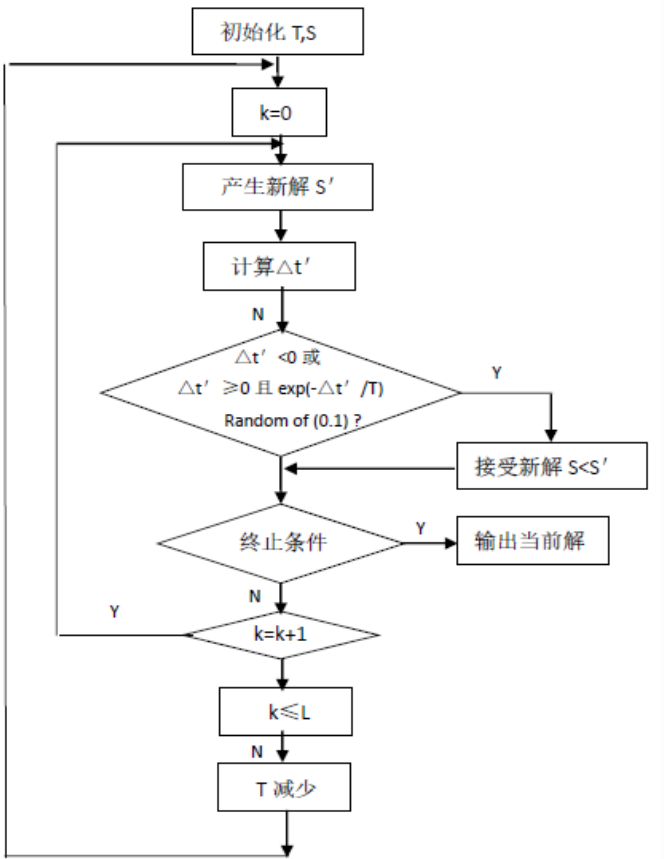


图 3 模拟退火算法的算法流程框图

运用以上的模拟退火算法，把 n 取为不同的值，来看最后的总成本。（下表的最低成本 为考虑了 400 组数据的平均值）

表 1 400 组样本的最低平均成本及其取点方案

选取点的个数	最优点选取方案	最低成本	误差成本
4	4 20 41 47	136.4	88.4
5	1 12 22 40 49	127.4	67.4
6	3 11 21 33 43 50	112.4	40.4
7	2 9 14 29 37 42 51	130.4	46.4
8	2 7 12 14 29 38 48 51	130.4	34.4
9	2 3 6 18 22 34 43 45 51	130.0	22.0
10	2 6 9 17 23 28 40 44 47 49	146.4	26.4
11	1 4 7 11 19 26 33 36 46 47 51	145.2	13.2

可以看到，当综合考虑取点成本和误差成本时，取 6 个点的成本是最小的，换句话说，用 5 次函数来拟合是最合适的。

### 3 结论

由表 1 可以明显看出，最终我们选取的是使用 5 次函数拟合，取点方案是[3 11 21 33 43 50]，最终得出的 400 组数据的平均最低成本为 112.4，其中取样成本为 72，误差成本为 40.4

### 4 参考文献

- [1] “统计推断”课程设计的要求 V2.2 2015-9-22
- [2] 上海交大电子工程系. 统计推断在数模转换系统中的应用课程讲义  
[EB/OL].ftp://202.120.39.248.
- [3] 百度词条“模拟退火算法”

### 附录（代码清单）

#### 1. main.m

```
function [ x0,q] = main(m )
%UNTITLED3 此处显示有关此函数的摘要
% 此处显示详细说明
T=1000;
a=1:51;
s0=1000;
xx=1:m+1;
s1=chengben(1:m+1,m);
while T>0
    x1=randperm(51);
    x=x1(1:m+1);
    for k=1:m+1
        a1(k)=a(x(k));
    end
    s=chengben(a1,m);
    dt=s-s1;
    mm=exp(-2*dt/T);
    if dt<0
        s1=s;
        if s0>s1
            s0=s1;
            xx=x;
        end
        x0=x;
        q=s;
    else
        kk=rand(1,1);
        if kk<mm;
```

```

        s1=s;
        x0=x;
        q=s;
    end
end
T=T-0.5;
end
if s0<q
    q=s0;
    x0=xx;
end

```

## 2. chengben. m

```

function [ cost] = chengben( a ,m)
%UNTITLED4 此处显示有关此函数的摘要
% 此处显示详细说明

my_answer=a;%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('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
    % 请把你的定标计算方法写入函数 mycurvefitting
    y1(j,:)=mycurvefitting(x_optimal(j,:),y0_optimal(j,:),m);
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*(le
2_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;
cost=sum(si)/nsample;
end

```

### 3. mycurvefitting.m

```

function y1 = mycurvefitting( x_premea,y0_premea,m )

x=[5.0:0.1:10.0];

p=polyfit(x_premea,y0_premea,m);
y1=polyval(p,x);

end

```

### 4. test\_ur\_answer.m

```

%%%%%%%% 答案检验程序 2015-11-04 %%%%%%%%%

my_answer=[ 2,9,14,29,37,42,51 ];%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('20150915dataform.csv');
[M,N]=size(minput);
nsample=M/2; npoint=N;
x=zeros(nsample,npoint);

```

```

y0=zeros(nsampl e,npoint);
y1=zeros(nsampl e,npoint);
for i=1:nsampl e
    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:nsampl e
    % 请把你的定标计算方法写入函数 mycurvefitting
    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*(le
2_0-le1_0)+6*(le3_0-le2_0)+12*(le5_0-le3_0)+25*g5_0;
si=sum(sij,2)+Q*ones(nsampl e,1)*my_answer_n;
cost=sum(si)/nsampl e;

% 显示结果
fprintf('\n 经计算, 你的答案对应的总体成本为%5.2f\n',cost);

```