统计推断在数模转换系统中的应用

组号：37　　姓名：张义诚 5130309443　　高鹏　5130309438

摘要：本文以某型产品内部的检测模块为研究对象，在确保测量精度的前提下，运用统计推断方法，以 MATLAB 为工具，通过选取原始数据库中 469 组密集选点作为实验数据进行分析，来寻找校准工序的优化方案，同时尽量降低测定成本。

关键词 :统计推断 多项式拟合 样条函数插值法 Hermite 插值法　模拟退火算法

The Application of Statistical Inference in the Digital-analog Conversion System

Group No.037　　Yicheng Zhang　5130309443　　Peng Gao　5130309438

ABSTACT

The report is based on a certain type of products within the detection module as the research object.On the premise of assuring the accuracy of measurement, we use statistical inference method, based on the MATLAB tool.We choose 469 pairs of intensive study in the original database as experimental data analysis, to find out the optimized plan for the calibration process and at the same time to reduce the measurement cost as much as possible.

KER WORDS

Statistical Inference , Polynomial fitting , Spline interpolation , Cubic pchip interpolation , Simulated Annealing Algorithm

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）。工业产品对精度有严格的要求，而其标准化测量不能够精确到每一个点，输入输出的受控关系。经过抽样研究，发现有如下特点：

（1）有确定的对应关系

（2）非线性或局部非线性

（3）个体差异较大

所以需要通过几个特殊的测定点来有效拟合出产品的特定曲线（本实验中为所测物理量与检测模块的特性曲线））并由此来对出厂产品进行标定。本实验中，我们通过原始数据库中的 469 组有效实验数据，讨论如何在限定误差的范围内完成标定，需要讨论测量点的个数、测量点的选取、受控曲线的表达式确定方法。由于数据组数较为庞大，我们不再可能使用暴力穷举等传统方法来得出结论，考虑到拟合效果与运行时间最终我们选定多项式与三次样条插值两种方式来拟合出最佳匹配曲线。接下来为了能够高效而准确地筛选出测

量点，我们将选择使用退火法拟合并对之进行了适当的改进，以得到最优结果。

1.2 模型评判标准

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

$$s_{i,j}=\begin{cases}0 & \text{if } \left|\hat{y}_{i,j}-y_{i,j}\right|\le 0.5\\ 0.5 & \text{if } 0.5<\left|\hat{y}_{i,j}-y_{i,j}\right|\le 1\\ 1.5 & \text{if } 1<\left|\hat{y}_{i,j}-y_{i,j}\right|\le 2\\ 6 & \text{if } 2<\left|\hat{y}_{i,j}-y_{i,j}\right|\le 3\\ 12 & \text{if } 3<\left|\hat{y}_{i,j}-y_{i,j}\right|\le 5\\ 25 & \text{if } \left|\hat{y}_{i,j}-y_{i,j}\right|>5\end{cases} \quad (1)$$

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

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

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。

- 校准方案总体成本

按式（3）计算评估校准方案的总体成本，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。

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

总体成本较低的校准方案，认定为较优方案。

2. 拟合方法的选取

2.1 选取的拟合方式

拟合的方式有很多种，常用的有多项式拟合，指数函数拟合，插值法拟合，高斯拟合，洛伦兹拟合，傅里叶拟合等，在对本次需要分析处理的数据进行分析后，认为本次数据两边平缓而中间趋于线性，所以我们决定采取多项式拟合以及插值拟合两种方法，并从中选取的一种最优的拟合方案。

2.2 多项式拟合

2.2.1 概述

在广泛观察了大量的实验数据之后，我们总结出数据实验曲线在两端的数据变化值偏大，而在中部的数据具有一个较为良好的线性关系，如图 2-1 所示的第 1 组数据点图像。

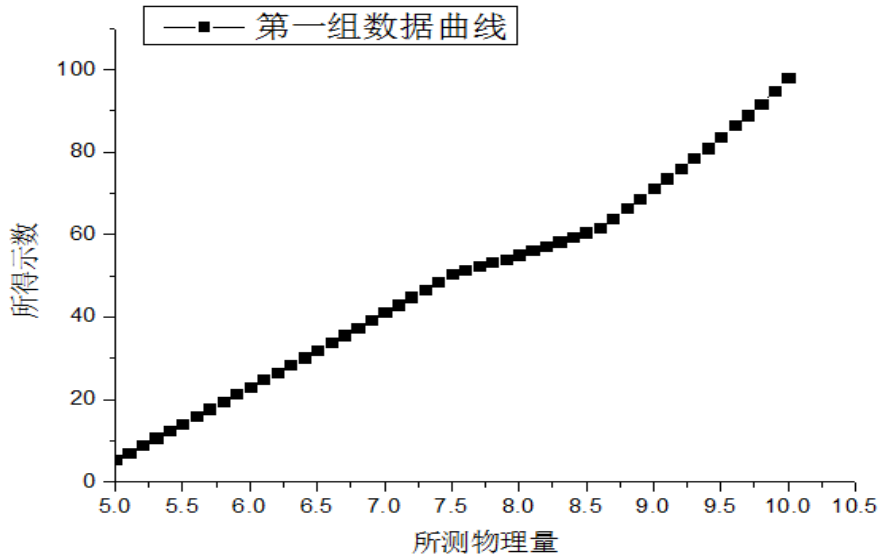


图 2-1 第 1 组数据点图像

因此我们认为可以通过多项式的性征来拟合出数据曲线，然而当我们想要通过平面上两点作一条曲线时，希望这条曲线能通过"A"和“B”的中点来满足拟合精度。对于低次多项式，曲线将没有很大波动，而能通过中点（对于一次多项式，甚至能保证肯定通过中点）。但是对于高次多项式，情况就不是这样了，高次多项式曲线往往可能有很大或者很小的幅值。因此认为可以运用低次多项式来拟合数据。考虑到以七个点来拟合的数据点数量不足，我们对二次至七次的各阶多项式进行了尝试。

2.2.2 各阶多项式的拟合效果比较

通过对二次至七次曲线的数据进行分析，得到如表格 2-1 中的数据。

表 2-1 各阶多项式的拟合结果

0

拟合项目	拟合组数	平均成本
二次多项式	51	248.5
三次多项式	51	107.4
四次多项式	51	112.3
五次多项式	51	130.5
六次多项式	51	185.1

通过对表格中数据进行分析可知，当拟合阶数为三阶、四阶、五阶时，其平均成本处于140左右，说明其拟合效果较好，能较为准确的反映出应有的数据走势，并且成本控制的较好。而二阶、六阶的曲线，其拟合所得成本明显偏高，故在实验中应当选取三、四、五次的多项式，进行数据拟合。

同时可从表中观察到，对于不同阶次的多项式而言，其运行时间也有较大的差异。对二阶、三阶的拟合，其用时较短，而四五六阶时间均在 0.6s 以上，目前已知的可能原因为评分函数中分数分段的判断语句结构影响以及多项式次数升高导致的运算复杂度不同，从

而增加了拟合时间然而具体原因仍在进一步研究当中。

2.2.3 结果分析

根据已有的数据进行分析，在多项式拟合曲线中，能够良好的反应出曲线的性状并保证运行速度的应为三、四、五次多项式，所以在寻找最优解的过程中，我们应当以上述三种多项式来进行拟合，从而达到最好的效果。然而同时多项式你和也反映出了其精度不高的问题，即便是效果最好的三次、四次多项式，随机取点的样本中拟合出来并进行自行评估后所得到的最低成本也没未达到期望值，因此我们认为需要考虑更为精确的拟合手段。

2.3 插值法拟合

2.3.1 Hermite 插值法

为了保证插值函数能够更好地逼近原函数，不仅要求两者在节点上有相同函数值，而且要求在节点上有相同的导数值。这类插值称为 Hermite 插值。Hetmite 插值法具有精度高，拟合出的曲线光滑程度好；且计算简单易于实现，同时又具有一定的局限性，如果想要部分修改数值是，并不会影响全局。

Matlab 关于插值法有现成的插值函数可以调用：y=interp1(x,y, i x ,method)

其中，x 和 y 均为数组，在本文附带程序中便是我们选取的 7 个点对应的占空比 D 和输出电压 U；xi 可以是一个数值也可以是一组数据；method 是所采用的插值方法，包括：

' nearest' ：最近邻点插值；

' linear' ：线性插值；

' spline' ：三次样条函数插值；

' pchip' ：分段三次 Hermite 插值。

其中，适合本次曲线拟合的是分段三次 Hermite 插值法（ “pchip” ）或者是三次样条插值法（ “spline” ）。pchip 与 spline 调用方式完全相同，为了保证运行结果与时间准确，且可进行横向对比，我们分别采用分段三次 Hermite 插值法“pchip”和三次样条函数插值法“spline”计算。

2.3.2 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。对于个给定点的数据集，我们可以用段三次多项式在数据点之间构建一个三次样条。用公式(2-3)[表示对函数 f 进行插值的样条函数，需要：

（1） 插值特性，

（2） 样条相互连接，

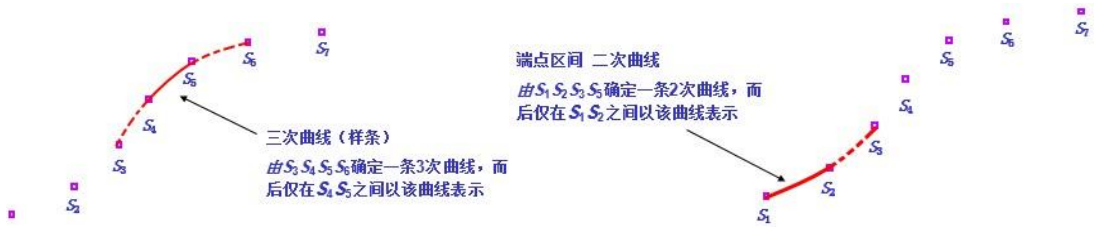
（3） 两次连续可导，以及

$$S(x)=\begin{cases} S_0(x), & x\in[x_0,x_1] \\ S_1(x), & x\in[x_1,x_2] \\ \cdots & \\ S_{n-1}(x), & x\in[x_{n-1},x_n] \end{cases} \tag{2-3}$$

由于每个三次多项式需要四个条件才能确定曲线形状，所以对于组成的个三次多项式来说，这就意味着需要个条件才能确定这些多项式。但是，插值特性只给出了个条件，内部数

据点给出 个条件，总计是个条件。我们还需要另外两个条件，根据不同的因素我们可以使用不同的条件。

在本课题中我们使用六段三次曲线样条来近似表达曲线性状，取法如图 2-2 所示。



(a) (b)

图 2-2 三次样条曲线的拟合样图（中间区间）

运用 MATLAB 软件自带的三次样条拟合函数 spline()，其输入值为两组已知的数据以及所求点，返回值为所求点在已知数据的三次样条拟合下的数值。MATLAB 自带的函数可以直接解决在非中间点的数据拟合，故可直接使用。然而运行 spline()函数需要很长的时间。

可见 MATLAB 自带三次样条插值的运行结果数据以及估算出的评分函数调用次数可以看出，直接使用 MATLAB 系统自带的函数是难以在短时间内得出有效的结果。即此种拟合方式。

三次样条函数插值法，精度高，可以避免龙格现象（指在插值过程中，插值函数的两个端点处发生巨大波动而产生较大误差的想象）因此可以用它来拟合函数，以较为理想地逼近原函数，可以在 matlab 中直接可调用 spline 函数进行拟合。

三次样条插值的算法复杂化程度很高，尤其是 MATLAB 自带的 spline()函数，很难满足本课题研究的要求，必须要进行改进。现在正在进行的工作为对自编的三次样条插值函数的边缘点拟合方式以及三次方程解法进行优化，目前在简单的尝试中改进后的函数在运行时间上得到了有效的缩减。

2.5 最优拟合方案的确定

通过观察表格中的数据，我们可以清楚的看到插值法的平均得分基本在 90 分以上，与 n 次多项式拟合法相比，优势是不存在警告且速度较快，劣势是评分不如 n=4 时 n 次多项式的拟合评分高，但差距不大。

我们综合考虑，因为求得的函数是要经用于工业大规模生产的定标，在及其大量的数据中，实验中的运行速度在投入生产中会被无限放大，进而影响经济效益，而且 n 次多项式拟合中产生的警告可能会造成拟合结果的巨大误差，所以我们认为牺牲少许的评分以提高速度和稳定性是值得的，于是最终决定在插值法之中的选取一种方法作为我们所使用的拟合方法。而在分段三次 Hermite 插值法和三次样条函数插值法两种方法中，在试验的六组 7 点集合中，分段三次 Hermite 插值法在均得分和总耗时上均略胜一筹。于是我们决定采用三次
Hermite 插值法作为拟合方案。

3. 特征点寻找

3.1 问题起源

本课程设置本质上是一个搜索问题，如果对全部数据进行搜索，所需搜索的范围巨大，为种组合，穷举搜索理论上将花费大量时间，实际上是不可能实现的 ，属于所谓的 NP 问题。所以我们决定通过几个特征点的拟合曲线来标定 D-U 曲线。

3.2 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。

3.2.1 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：

(3) 产生新解 S′

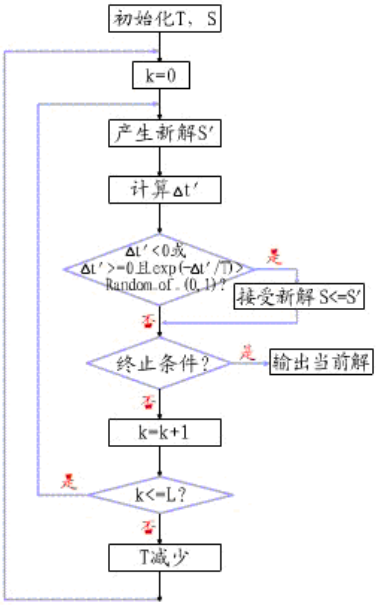
(4) 计算增量**错误!未找到引用源。**=**错误!未找到引用源。**-**错误!未找到引用源。**，其中**错误!未找到引用源。**为评价函数

(5) 若**错误!未找到引用源。**<0 则接受 S′作为新的当前解，否则以概率**错误!未找到引用源。**接受 S′作为新的当前解.

(6) 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。

(7) T 逐渐减少，且 T->0，然后转第 2 步，流程图可见图 3.1

图 3.1 模拟退火算法流程图



3.2.2 退火算法的具体实现

1） 初始化部分

由于在初始温度足够大的情况下，初始温度实际上是对于最后结果没有影响的，所以我们暂定为 T=100，之后 r=0.99，并且 Tf=0.01，这样保证了循环次数。

2） 产生新解

在本课程中，新解即为新的特征点组合，这里我们采用了随机某点随意左右摆动 1 的方案，并且由于为了是代码更加简练，这里我们限制了随机产生的特征点组合的特性，即每组点只能从小到大排列且每个点均需要在 1-51 的范围之内，这样做可以大大方便我们之后进行的计算。

3） 计算增量

因为目标函数差仅由变换部分产生，所以目标函数差的计算最好按增量计算。事实表明，对大多数应用而言，这是计算目标函数差的最快方法。

4） Metropolis 准则判断是否接受

这里我们选用了模拟退火算法的经典判断条件来决定是否接受所产生的新的特征点组合，若 Δt′ <0 则接受 S′ 作为新的当前解 S，否则以概率 exp(-Δt′ /T)接受 S′ 作为新的当前解 S。

5） 退火循环

当连续几次拒绝接受新解时，开始退火，即 T=T*0.99，然后重新开始 1)-4)循环直至 T<0.01 从而输出结果。

表 4-1 模拟退火算法得到的组合及其均成本（Hermite 插值法）

第一组	8,28,35,44,46,50	均成本	88.8262
第二组	4,13,14,16,26,35	均成本	88.8804
第三组	19,20,21,31,45,48	均成本	88.2537
第四组	17,24,29,40,44,45	均成本	88.5512
第五组	20,33,41,46,49,51	均成本	88.2953

表 4-2 模拟退火算法得到的组合及其均成本（Spline 插值法）

第一组	8,16,18,28,36,40	均成本	97.8849

第二组	14,15,29,32,36,43	均成本	96.6535
第三组	4,13,19,26,32,51	均成本	94.8497
第四组	14,15,22,25,32,43	均成本	99.5906
第五组	11,12,14,22,35,42	均成本	97.4552

3.2.3 结论

通过以上 10 组数据可以分析得出，取 6 个点时可以得到最优解，而且 Hermite 插值法普遍比 Spline 插值法得出的解要更优。

对于 Hermite 插值法，我们得出的比较满意的取点方法是[8 28 35 44 46 50]。

对于三次样条插值法，我们得出的比较满意的取点方法是[4 13 19 26 32 51]。

4. 致谢

感谢袁焱老师在课堂上深入浅出的讲解，启发了我们的思维，并在这段时间为我们答疑解惑，提供帮助！我们能够顺利完成全部工作离不开老师的悉心指导，再次感谢老师的帮助。

5. 参考文献

1. 统计推断提供的课程 ppt

2.http://www.cnblogs.com/heaad/archive/2010/12/20/1911614.html

大白话解析模拟退火算法

3.http://baike.baidu.com/link?url=wRRyvhwwJVFJxLCRrt67ZZdjHiLRd6ReqyD73idwol2W9803RICkGti54cziDIAL

模拟退火算法百度百科

4.http://hi.baidu.com/yangchenhao/item/c2153975ed35ca2b5c17898f

数值分析—三次样条插值

5.http://baike.baidu.com/link?url=ai4dBOJAl7RpZdOgEwKJXJHnAUyS0fojygCPvMdkzAdsVSsTYjC8pxKOTyNjP42MIA2C6DedpIKyd9pV4FEeT_

三次样条插值百度百科

附录：matlab 程序（R2014a 环境）

```
data=xlsread('20141010dataform.csv');%读入数据表中的数据
D=data(1:2:end,1:end);
U=data(2:2:end,1:end);%将表格中的 D 和 U 分别取出
J=10000;
Tf=0.01;
T=100;
tic;
while T>Tf
%n=0;
n=6;
%while n<Tk/10;
%n=n+1;
A=randperm(51);
B=sort(A(1:n)); %首次整理取点
K=B;%误差最小的情况
remain=setdiff(A,B);%寻求差集
E=remain(randperm(51-n));
F=randperm(n);
S=B;
S(1,F(1))=E(1,F(1)+1);
S=sort(S);%以上代码用于随机替换 n 个数中的一个数字
cost=zeros(1,469);
M=zeros(469,51);
for i=1:469
X=D(i,S);
Y=U(i,S);
%p=polyfit(X,Y,3);
%yi=polyval(p,X);
%plot(X,Y,yi,'r*');
M(i,:)=U(i,:)-interp1(X,Y,D(i,:),'pchip');%Hermite 插值
for j=1:51
if abs(M(i,j))<=0.5
cost(1,i)=cost(1,i)+0;
elseif abs(M(i,j))<=1 && abs(M(i,j))>0.5
cost(1,i)=cost(1,i)+0.5;
elseif abs(M(i,j))<=2 && abs(M(i,j))>1
cost(1,i)=cost(1,i)+1.5;
elseif abs(M(i,j))<=3 && abs(M(i,j))>2
cost(1,i)=cost(1,i)+6;
elseif abs(M(i,j))<=5 && abs(M(i,j))>3
cost(1,i)=cost(1,i)+12;
else cost(1,i)=cost(1,i)+25;
```

```
end %以上代码用于计算一组 n 个点拟合 1~51 每个点所得分数并加入
end
cost(1,i)=cost(1,i)+12*n;
if (cost(1,i)<0)
cost(1,i)=0;
end%以上用于求出一组数据的分值
end
ave_cost=sum(cost,2)/469;%求 469 组的平均分值
if ave_cost<J
J=ave_cost;
cost_save=ave_cost;
K=S;
B=S;
elseif rand<exp(-(ave_cost-cost_save)/T)%决定点是否变化
cost_save=ave_cost;
B=S;
end
J;ave_cost;
%end
T=T*0.99;
disp(J);
end
J
K
toc;
```