

统计推断在数模转换系统中的应用

组号 58 童仙雨 5130309652, 张颖 5130309459

摘要: 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。而后得到若干个关键点。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。这样既能测绘得足够精确的曲线, 又能降低成本, 节约时间。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, 曲线拟合, 遗传算法

ABSTRACT: 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

1 引言

在工程上, 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、噪声等影响较大, 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。而对于工业大规模批量生产, 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现实意义的。

1.1 任务概述

为某型产品的一个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。

1.2 设计思路

基本任务中主要有两个方面的问题需要考虑:

- (1) 测定点数量越多, 测定成本越高;
- (2) 估算点数量越多, 误差损失(成本)越高。

这是一个组合优化问题, 但备选的组合方案数量巨大, 是典型的 NP-hard 问题, 目前的计算机尚难以对以上问题采取暴力穷举的方式进行求解, 故采用遗传算法。

2 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 7 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。

3 方案概述

3.1 拟合方式

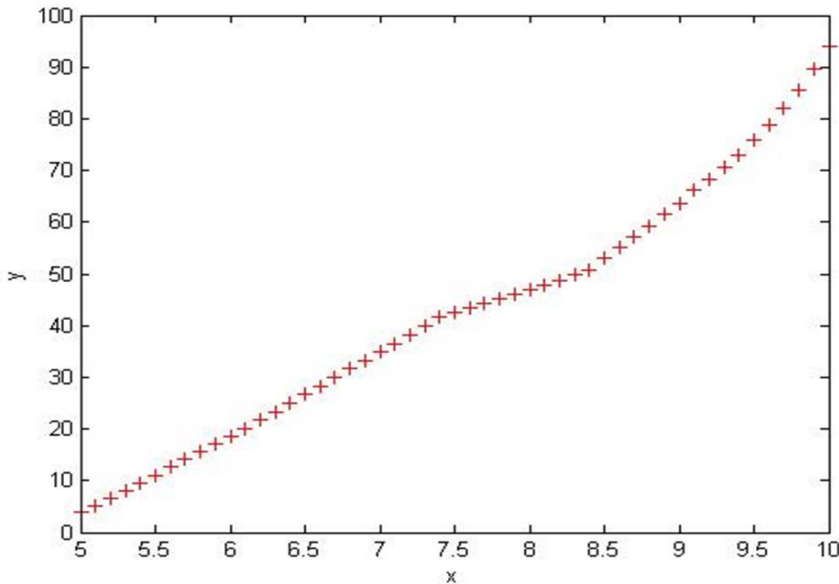


图 1 样本的特性曲线

分析此样品的特性曲线，有以下特点：

- (1) 特性曲线是单调递增的；
- (2) 大致可以分为首中尾三段，三段都不是完全线性的，有一定弯曲度；
- (3) 中段的斜率小于首段和尾段的斜率，且中段的起点位置和长度都带有随机性。

因此，我们将特性曲线分三段分别拟合。那如何确定中段的起始位置呢？由观察可以知道首中尾段的斜率有明显的不同。因此我们计算了样本值任意相邻两点间的斜率，并比较大小，其中有两点前后的斜率变化最大即为中段的起始点。遂以此两点为分界，将特性曲线分为三段，用 Matlab 自带的 `polyfit` 函数拟合。

3.2 遗传算法

即引入生物进化论的概念，仿照生物进化过程中染色体的交叉、变异与自然选择过程，对算法进行优化，以更快的达到“最优解”。由于暴力穷举法中待研究的情况超出可承受能力，故采取上述人为设定限制因素求解的方法。但不可避免的引入较大的人为干扰，必然会对结果的准确性产生影响。而采用遗传算法，借助其特有的交叉、变异和选择功能，可最大范围且更具针对性的讨论可能组合情况，并最终逼近最优解。

3.2.1 遗传算法中的基本概念

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。

3.2 遗传算法求“最优解”步骤

3.2.1 初始化种群

选取 100 个样本点作为初始种群，并对每组进行多项式拟合，得到拟合函数 F 。
将成本函数的倒数作为适应度函数（ Q 值）。

3.2.2 选择

从初始种群的 N 个父代染色体中选择出 $N \cdot p$ 个（ $N \cdot p$ 为期望值）进行交叉和变异。这个规则必须有利于选择更为接近适值，即 Q 值最小的若干父代。需要说明的是，一个染色体可能被选择多次，但保持最终 K 个染色体数不变。

3.2.3 交叉

已选择出的父代中随机产生 $N \cdot pc$ 个染色体进行交叉。交叉时，将染色体转化为二进制数，随机选取交叉位点，将两个染色体位点之后的部分进行交叉互换，产生新的染色体作为子代染色体。此处需要对产生的子代染色体进行判断，已确定其是否超出研究范围，即是否合法。

3.2.3 变异

设变异概率 $pe=0.025$ 依然需要将所有染色体转化为二进制数，二进制位数为 M 。此时选择对象为基因，即从所有 $N \cdot M$ 基因中随机选择 $N \cdot M \cdot pe$ 个基因，这些基因将发生变异，即从 1 变为 0，从 0 变为 1，变异时，需判断变异位点的位置；变异后，需检查子代的合法性。

之后，通过父代和通过交叉、变异得到的子代的竞争，即从中选择 K 个 Q 值最小的染色体组成新的种群，然后继续产生下一代，以此类推，直到若干代后，种群收敛于以特定种群，即为问题的最优解。

4 结论

最终借助 MATLAB 编程实现以上步骤，我们得出以下结果：

取点方案：2,19,31,32,33,46

最小成本值：136.8891

我们计算出来的这个成本值相对来说还是偏高，分析可能是因为特性分三段以同一种方式拟合所致。进一步研究的话将可以组合多种不同的拟合方式并比较得出来结果，但由于时间和技术原因本次未能实现，实为一憾。

5 致谢：

感谢袁焱老师的讲座指导，还有助教们的耐心答疑与指导。

6 参考文献：

《统计推断在数模转换中的应用》课件 袁焱老师

《遗传算法及其应用》（陈国良等编著）

《matlab 6.5 辅助优化计算与设计》（飞思科技产品研发中心编著）

附录:

MATLAB 实现代码

cal.m

```
function f=fun(x,y)
s=0;
for i=1:51
    if (abs(x(i)-y(i))>0.5)
        s=s+0.5;
    end
    if (abs(x(i)-y(i))>1)
        s=s+1;
    end
    if (abs(x(i)-y(i))>2)
        s=s+4.5;
    end
    if (abs(x(i)-y(i))>3)
        s=s+6;
    end
    if (abs(x(i)-y(i))>5)
        s=s+13;
    end
end
f=s;
end
```

calsum.m

```
function f=fun(x,m)
n=0;
for i=1:51
    n=n+x(i);
end
if (n>=3)
x1=zeros(n,1);x2=zeros(n,1);
y1=zeros(n,1);
t=0;
for i=1:51
    if (x(i)==1)
        t=t+1;
        x1(t)=i/10+5;
        x2(t)=i;
    end
end
```

```

end
s=0;
for i=1:469
    for j=1:n
        y1(j)=m(i*2,x2(j));
    end
    y2=draw(x1,y1,n);
    t=calc(y2,m(i*2,1:51));
    s=s+t;
end
s=s/469;
s=s+n*12;
else
    s=500;
end
f=s;
end

```

draw

```

function f=fun(x,y,n)
min=999999;mi=2;max=0;ma=n-1;
for i=2:n-1
    t=(y(i+1)-y(i))/(x(i+1)-x(i))-(y(i)-y(i-1))/(x(i)-x(i-1));
    if t<min
        min=t;mi=i;
    end
    if t>max
        max=t;ma=i;
    end
end
x1=zeros(mi,1);
y1=zeros(mi,1);
for i=1:mi
    x1(i)=x(i);y1(i)=y(i);
end
f1=polyfit(x1,y1,1);
x2=zeros(ma-mi+1,1);
y2=zeros(ma-mi+1,1);
for i=mi:ma
    x2(i-mi+1)=x(i);y2(i-mi+1)=y(i);
end
f2=polyfit(x2,y2,1);
x3=zeros(n-ma+1,1);

```

```

y3=zeros(n-ma+1,1);
for i=ma:n
    x3(i-ma+1)=x(i);y3(i-ma+1)=y(i);
end
f3=polyfit(x3,y3,1);
y4=zeros(51,1);
for i=1:51
    xt=i;xt=xt/10+5;
    if (xt<x(mi))
        yt=f1(1)*xt+f1(2);
    end
    if (xt>=x(mi)&&xt<x(ma))
        yt=f2(1)*xt+f2(2);
    end
    if (xt>x(ma))
        yt=f3(1)*xt+f3(2);
    end
    y4(i)=yt;
end
f=y4;
end

```

gene.m

```

function f=fun(m)
x=zeros(20,51);
x1=zeros(20,51);
for i=1:20
    for j=1:51
        t=rand;
        if (t<0.1)
            x(i,j)=1;
        end
    end
end
y=zeros(21);
for p=1:30
    st=0;
    for i=1:20
        y(i)=1.0/calcsun(x(i,1:51),m);
        st=st+y(i);
    end
end

```

```

y(20)=999999999;
for q=1:20

    t1=rand;t1k=1;
    while (t1>y(t1k)/st)
        t1=t1-y(t1k)/st;
        t1k=t1k+1;
    end
    t2=rand;t2k=1;
    while (t1>y(t2k)/st)
        t2=t2-y(t2k)/st;
        t2k=t2k+1;
    end
    t3=rand;t3=t3*51;
    t3k=1;
    while (t3k<t3)
        x1(q,t3k)=x(t1k,t3k);
        t3k=t3k+1;
    end
    while (t3k<=t1)
        x1(q,t3k)=x(t2k,t3k);
        t3k=t3k+1;
    end
    t4=rand;
    if (t4<0.025)
        t5=rand*50+1;
        t5=floor(t5);
        x1(q,t5)=1-x1(q,t5);
    end
end
x=x1;
p

end
min=99999999;mi=0;
for i=1:20
    t=calcsun(x(i,1:51),m);
    if (t<min)
        min=t;
        mi=i;
    end
end
min
x(mi,1:51)
end

```

