

ICEE

小型及微型计算机应用论文选

上海科学技术文献出版社

前 言

美国电气、电子工程师学会 (IEEE) 所属工业电子学与控制装置 (IECI) 学会第六届年会, 1980 年 3 月 17 日至 20 日于美国宾州费城召开。这次年会的主题是“小型和微型计算机应用”。会议首先用两整天的时间作了数据采集系统和小型、微型计算机应用的指导性宣讲, 然后分 18 个专题作普通的小会介绍, 另有 2 次是特别会议。年会的论文达 80 篇以上, 而其中一半是来自美国以外的 16 个国家。会议期间还特邀英特尔公司 (Intel Corp.) 微处理机及外围设备部总经理 D. L. House 作了题为“微型计算机发展方向”的重点演说。18 个专题所涉及的范围十分广泛, 它包括了小型和微型计算机的许多侧面及其应用, 代表了国际上本专业有关大学、研究院所及大型工厂企业研究和应用的较新水平。现组织上海市电气自动化研究所、上海交通大学、上海市计算技术研究所、上海电器科学研究所和华东师范大学等五个单位的部分同志选译了该论文集集中的 30 篇文章, 内容包括: 过程控制、辅助设计、电动机控制、大功率逆变器、智能测试系统、分散系统、电站应用、生产制造自动化、能源管理、仪表接口、数据采集、通信技术、生物医学、控制器、信号处理、交通运输等十六个方面的微型计算机应用有关的硬件配置、软件开发、系统设计和理论分析等主要专题。

我国目前已经发展生产了几种微型计算机系列及其必要的外部设备, 且正朝着扩大应用面、提高我们使用技术的方向大踏步前进, 若本论文选的出版能有助于国内同行作参考则幸甚。由于论文的专业面广而译者水平有限, 不妥之处望读者指正。

编者

1982 年 4 月

目 录

使用微型计算机改进过程控制的测量精度	1
以微处理机为基础的工业顺序控制系统的设计自动化	9
一种简化微机控制系统程序编制的高级语言	15
采用多微处理机系统的直流电机测速及控制	23
微处理机三相逆变器用于感应电动机的速度控制	27
微机数字式 PLL 速度控制系统性能的改善	33
按稳定性及品质考虑的数字式 PID 调节器参数选择	39
用微处理机的高级可编程控制器	45
通用型过程控制接口	53
用微处理机的适应控制系统	61
可控硅变流器包括高可靠性的快速微处理机控制	66
发电机组实时数字控制器的分析、设计和实验结果	72
一个采用微处理机的轴承动态监控系统	75
一种基于数据流的分布计算系统	79
分布式智能多微计算机系统用于超高压电站控制——其软硬件设计效率	86
低损耗光纤接插件修整机的微处理机自动化控制系统	92
分散型能源控制系统的功能分配	95
用于运动过程控制的微型计算机系统设计	101
一种通用异步串行接口的设计方案	107
一种以微处理机为基础的声速、温度、深度和含盐量测量设备	114
用于 TM 990/101 微型计算机的高速数据采集和硬件信号处理器	119
多终端分时显示系统	125
IEEE-488 接口总线的可编程规约翻译机的设计	131
呼叫改善系统——多微型计算机的一种应用	137
一种远程连接单元——多处理机远程终端	141
用于定量色谱数据分析的微处理机系统	147
微型计算机机器人控制系统	153
使用多路复用 I/O 的计算机控制系统	160
实现快速沃尔什变换的微处理机	164
一种用于孤立交叉点的混合型交通控制器	169

使用微型计算机改进过程控制的测量精度

W. J. Barnett R. W. Gilchrist

本文叙述了减少非随机误差的一个联机系统的研制成果,非随机误差是由传感器和测量仪器的偏移、非线性和滞后作用等因素造成的,而这些因素又是由温度和湿度等测量参数的改变所产生的。这个系统由在通用机上运行的脱机分析程序和微型计算机上运行的联机误差减少程序组成。

引言

在很多过程监督和过程控制系统中,日益需要较好的测量精度。核材料加工工业就有这种特殊需要的一个领域。 $\pm 1\%$ 的满刻度的标准精度,对于通常情况已是很好了,但尚不能满足某些特殊问题的要求。所以,我们不得不寻找测量精度在 0.1% 和 0.01% 之间的方法^[1]。有一些方法能够用来解决这个问题。目前正在大力进行研究的是使用一种综合系统的统计分析方法来有效地减少随机误差^[2]。这些方法可能很有效,但它们通常只适于处理随机误差,而在大多数过程监测传感器和过程监控仪器中却往往要遇到非随机误差。

本文探讨用微型计算机来大大减少非随机误差的问题。至于其它误差,如果确实是随机误差,则可以采用别的方法使之进一步降低。

本文所考虑的非随机误差,涉及下面几个因素:

1. 偏移和非线性;
2. 由于诸如温度、湿度、传导率和压力等测量参数的改变而引起的仪器响应特性的改变;
3. 滞后现象。

通常的方法是对若干种条件(即测量参

数)采用一系列定标测试来收集一种仪器如何对若干种条件(即测量参数)作出响应的数据,这些数据用脱机的计算机程序进行分析,以便建立一组方程,用来描述仪器对这些不同条件的响应。这组方程存放在一台联机微型计算机内。这个系统从测量参数的目标仪器和传感器接受测量值,并根据本项研究工作所研制的算法计算其改进值。

在测试仪器(模拟式伏特计)上所测得的实验数据表明,本系统能使偏移和非线性误差从平均值为 0.33% FS、标准差为 2.51% FS减少到平均值为 0.00% FS、标准差为 0.19% FS,与测量参数(温度和相对湿度)变化有关的误差将由平均值为 1.01% FS减少到 0.29% FS。同时,在一系列测试条件下,平均滞后误差从 0.18% FS降低到 0.08% FS。

本系统的一个主要优点是采用了一台微型计算机来控制一台或数台仪器。本方法适用于正在研制的能用于很多领域的智能仪器,同时也适用于以监督很多仪器的大型计算机为基础的系统。

先前的工作

在收集数据记录和控制系统以及新近研制的、采用微处理器的控制仪器中,已普遍地使用计算机校正测量数据^[3]。对于仪器响应

1110282

误差(例如,偏移和非线性)的校正方法,大多数都不计影响被测对象的那些参数(例如,温度、密度等)的变化。有一些仪器确能校正某些参数,但是,所用方法常常是以表达一类仪器的响应方程(例如,通过小孔的具有不同密度的液体流动 δ 压力的方程)为基础的,而不是针对一种特定的仪器。本项研究工作就是以上述方法为基础的。也就是说:我们假设,以这类仪器为基础的校正已经做好,但是有些误差对于一个特定的仪器仍然存在。另外,本项工作已经研究了一个减少滞后误差均值的方法,这个课题以前几乎很少受到注意。

系统的描述

在使用本系统时,共分为三步:

- a. 收集定标数据;
- b. 执行脱机分析程序;
- c. 装入和执行联机误差校正程序。

我们要求在实际操作时对规定的测量参数范围,用一系列定标测试来收集系统所必须的数据。

在一台模拟式伏特计的实验测试中,温度和湿度有着重要的影响。因此,伏特计应安放在温度和湿度可调的房间内,然后进行一系列定标测试。因为要考虑滞后现象的影响,所以信号从0%连续上升到满刻度100%以及从100%连续地下降到0%过程中的定标数据都应收集。定标数据输入脱机的数据分析程序。对于每个参数条件,程序根据最小均方差准则,对下列三个数据集计算一、二、三次多项式方程:

- a. 上升信号数据(即UP曲线);
- b. 下降信号数据(即DOWN曲线);
- c. 前两者的组合(即COMBINATION曲线)。

• 2 •

这个程序还计算每个方程的拟合质量,并通过权衡提高方程次数从而增加计算复杂性的得失,用来指出这些方程中哪一个方程的曲线最好。

对于每个测量参数条件,可选择UP、DOWN和COMBINED方程中的一个,将其系数加入微型计算机联机误差减少程序。这个程序通过模数转换器,接收待校正的仪器或传感器输出数据及影响仪器的参数的测量值。

被测物理量的校正值由联机程序用三个与上述三类误差有关的算法进行计算。这三个算法本质上是分级的,减少滞后误差的算法要求对每个UP、DOWN和COMBINED曲线求得“真值”。对当前的参数条件,这三个“真值”由用于定标测试过程中表示参数条件的若干个“线性化”值之间的插值得到,这些“线性化”值由所选择的减少偏移误差和非线性误差的多项式方程得到。

下面将分别讨论每一个算法和它的实验结果。

减少偏移和非线性误差

如果一台仪器在指定测量域内的响应曲线相当平滑,那末,三次或低于三次的多项式方程就能很好地模拟这种响应。其实,大多数实际装置都有很“平滑”的响应曲线,而不是非规则的响应曲线,并且最小均方差方程

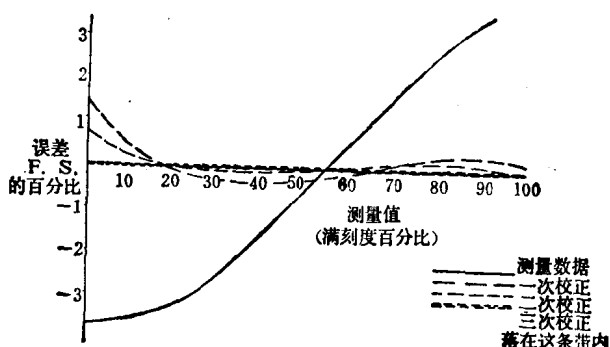


图1 采用回归方程的误差减少图

基本上不受定标数据中随机性误差的影响。

脱机分析程序用经典的回归分析法计算多项式方程。并打印其均值、标准差和最大误差,以指出方程对数据的拟合质量。(在实验测试中一般采用三次方程。)

用图 1 所示的由于实验仪器的偏移和非线性所引起的剩余误差,作为由多项式方程获得改进的一个说明。

减少测量参数误差

大多数仪器都在一定程度上受到某些参数条件或数量的影响,只有许多现代化的仪器这种影响才能减少到可以忽略的程度。对于很多其它的仪器,这些影响带来了很大的误差。

用户必须确定什么参数是重要的,从而必须规定其该取的定标数据的数值。例如,在一个实验测试中,必须确定温度和相对湿度是最重要的参数,因而规定定标数据必须取在温度 10°C 到 50°C 的范围以及相对湿度 30% 到 60% 的范围,间隔分别是 10°C 和 10%。因此,要取 20 组定标数据。脱机分析程序确定表示每种条件响应的最佳多项式方

程。

校正算法的第一步是确定限制被测参数值的定标条件,对于每个限制值,采用有关的方程系数来计算“真值”;其次,完成一系列插值来决定现时参数值上的“真值”。

为了测试系统中减少的测量参数误差分量,在一个温度和湿度可调的房间内放一只伏特计来收集所需的定标数据。脱机分析程序对定标数据进行运算。同时,把限制条件集的三次方程的系数加入联机程序*。这些限制条件集表示如下:

测试集	有限制的参数条件				被限制的条件	
	温度	相 对 湿 度	温度	相 对 湿 度	温度	相 对 湿 度
A	10°	30%	30°	50%	20°	40%
B	10°	40%	30°	60%	20°	50%
C	20°	30%	40°	50%	30°	40%
D	20°	40%	40°	60%	30°	50%
E	30°	30%	50°	50%	40°	40%
F	30°	40%	50°	60%	40°	50%

于是,由限制条件数据得到的测量值和参数值,作为联机程序的输入。请注意,这样产生的测试条件通常恰好位于最坏限制值的中间。

然后,分析联机程序的输出可以检验校

表 1 测量参数误差减少方法的实验结果

参数条件		参数误差减少的结果		对于最密切的参数条件简单误差减少程序的结果							
温 度	相对湿度	误 差 均 值		高 温 高相对湿度 误差均值		低 温 高相对湿度 误差均值		高 温 低相对湿度 误差均值		低 温 低相对湿度 误差均值	
		(伏)	(% 满度)	(伏)	(% 满度)	(伏)	(% 满度)	(伏)	(% 满度)	(伏)	(% 满度)
20	40%	-.010	0.125	-.077	0.963	.073	0.913	-.102	1.275	.067	0.838
20	50%	-.021	0.263	-.096	1.200	.095	1.188	-.125	1.563	.043	0.538
30	40%	.015	0.188	-.078	0.975	.125	1.563	-.093	1.308	.107	1.338
30	50%	-.013	0.163	-.103	1.288	.092	1.150	-.120	1.500	.078	0.975
40	40%	.041	0.513	-.009	0.113	.121	1.513	-.046	0.575	.098	1.225
40	50%	.040	0.500	-.022	0.275	.015	0.188	-.046	0.575	.080	1.225
%FS (% 满度)的平均值=0.292 上面 24 个 %FS 的平均值=1.011											

* 这些都是定标数据的 UP 位置。这样做使得计算中消除了滞后影响。

正方法的有效性。对包含参数的校正值和由原始数据得到的真值(对于情况a)作了比较,结果(见表1)表明:带有参数校正的平均误差远远优于未经参数校正的平均误差。不考虑减少由于温度和相对湿度改变所引起的误差时,所有FS平均误差百分比的绝对值的平均值是1.011%,而考虑误差减少的系统后,此平均百分比降低为0.292%。

减少滞后误差

1. 滞后误差的描述

滞后现象所引起的仪器误差,是由于仪器内部部件的摩擦和阻尼所引起的。有时候,把外摩擦和死区效应也作为滞后的一部分。一个典型仪器的滞后特性如图2所示。通常是用UP和DOWN曲线在整个曲线范围内的最大偏差来规定滞后误差。

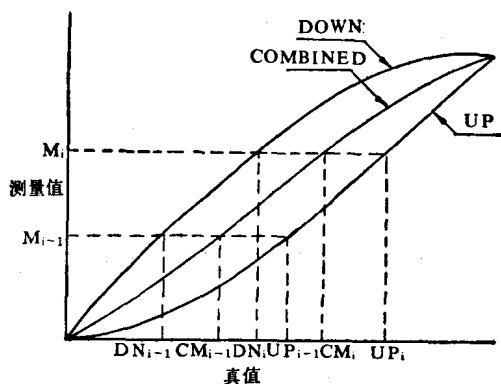


图2 局部滞后校正算法

校正滞后误差的一种方法是在滞后回路内以某种方式对所有可能的路径建立被测量值与真值的关系方程。于是,我们可用输入初始测量值和各个逐次测量值以及跟踪滞后回路内所遵循的路径的轨迹的办法来实现校正。采用这个方法的主要困难在于推导方程组。方程的个数从理论上看是无限的,而实际上可能穿过的路径也是非常大的,所以方程组的个数也是很大的。另外,还必须解决误差积累等问题。由于存在上述困难,对于

完整的滞后校正,这个方法就无能为力了。

2. 局部校正滞后误差的一般方法

然而,滞后误差的局部校正还是可能的。这种局部校正方法是依据滞后曲线的下列特性进行的:

(1) 测量值总是比真值滞后(也就是说,当真实信号增加时,测量值总是要比它低,而当真实信号减少时,测量值比它高)。

(2) 延迟量(误差)依赖于信号的变化量。

基于这些特性的方法是:首先,用前面所讨论的不考虑滞后的方法对给定的测量值确定一个校正值;然后,从最近的测量点据根改变的方向和改变的数量,对那个校正值进行修正。

为了实现这个方法,还需要关于特定仪器的滞后特性的信息和先前的测量值的信息。

仪器的特征信息由定标数据得到,这些定标数据已用于前面所讨论的方法,并带有附加要求:对每个参数条件,取两个分开的数据集,其中一个用于“UP”滞后曲线,而另一个用于“DOWN”滞后曲线。例如,有20组定标数据,每组由取自上升信号(即UP)的一组点以及取自下降信号(即DOWN)的一组点组成。

3. 脱机定标数据分析

脱机回归分析的结果类同于前面所描述的结果,但是带有附加信息。对于每个参数条件,均要对UP数据、DOWN数据和COMBINED数据进行完整的分析。“Combined”方程刻划了对所有被考虑点的最佳拟合曲线,它十分接近UP曲线和DOWN曲线的平均值曲线。

除了前面所描述的拟合质量之外,还能打印输出每条曲线滞后量的信息,这个信息包括最大滞后误差(正象用UP和DOWN曲线的最大偏差来进行度量一样)和它所出现的点。

4. 局部滞后校正的算法

局部滞后校正所需的数据涉及到仪器操作的最近的情况, 仪器操作可由下列因素确定: (1) 前面的测量值; (2) 上述值的变化方向。如果从最近数据点以来输入数据仅有微小变化, 那末, 这个信息对于局部校正已足够, 也就是说, 测量的采样频率已足够快了。

在讨论算法以前, 需引进下面一些定义:

UP_i : UP 曲线上的一个特殊点;

DN_i : DOWN 曲线上的一个特殊点;

CM_i : COMBINED 曲线上的一个特殊点;

M_i : 最近(或当前)测量值;

M_{i-1} : M_i 前一点的测量值;

M_{i-2} : M_{i-1} 前一点的测量值;

F_i : 值 M_i 的真实信号的第一(或初始)估计值;

B_i : 值 M_i 的真实信号的改进(或“最佳”)估计值*;

K : 仪器满刻度值的固定额定值。但有时为了提供较好的结果需对该值加以修改。

图 3 表示一条典型的滞后曲线的这些曲线和点。

假设所需信息都能得到, 则实现算法的步骤如下:

第一步:

作用: 算法预置初值(或复位)(a), 当进行第一次测量时(例如, 在接通电源以后)或较一般的情况是在每次当前点改变的方向总是不同于前面点改变的方向时, 在开始点上, 真实信号的“最佳”估计值是对被测量的值由 COMBINED 曲线所确定的值, 即对于数值 M_0 , 有 $B_0 = CM_0$ 。

这一步的作用是在当被测量的值改变方向时, 则算法复位, 使得初始校正值落在 Combined 曲线上。

第二步:

作用: 如果在点 (i) 上改变的方向和在前一

点 ($i-1$) 上改变的方向相同, 那末, 第一个值 F_i 根据前面的值 B_{i-1} 估算。

$$F_i = UP_i - \frac{(UP_i - DN_i)}{(UP_{i-1} - DN_{i-1})} \times (UP_{i-1} - B_{i-1})$$

或等价地

$$F_i = DN_i + \frac{(UP_i - DN_i)}{(UP_{i-1} - DN_{i-1})} \times (B_{i-1} - DN_{i-1})$$

这一步的作用是在上、下运动时保持滞后回路的相对位置。

第三步:

作用: 根据最后测量值的距离变化作一次改进。对于向上运动:

$$B_i = F_i + (UP_i - F_i) \frac{(M_i - M_{i-1})}{K}$$

(不可能有 $F_i > UP_i$)

对于向下运动:

$$B_i = F_i - (F_i - DN_i) \frac{(M_{i-1} - M_i)}{K}$$

(不可能有 $F_i < DN_i$)

这一步的作用是根据滞后曲线的一般特性改进“最佳”估计值, 方法是对于上升信号向右移, 而对于下降信号则向左移。

该算法的图解说明示于图 3。下面的讨论根据图 3 进行。

测量第一步: 假设 B_i 的斜率由输入值 M_i 从 Combined 曲线找到。

测量第二步: 斜率和前一步相同, 所以, F_2 用算法方程计算, 这些方程实现下列图示的比值

$$F_2 = \frac{D_1}{W_1} \cdot W_2$$

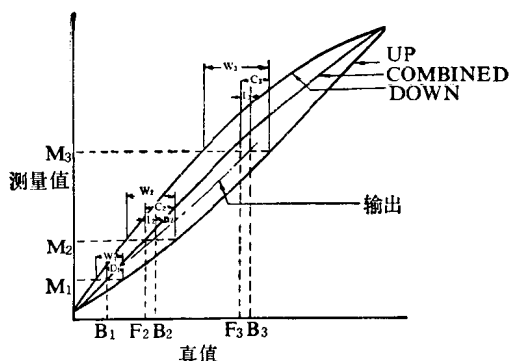
于是, 可算出 B_2 : $B_2 = F_2 + I_2$ 。

测量第三步: 斜率不改变, 所以, F_3 根据图示的比值

$$F_3 = \frac{D_2}{W_2} \cdot W_3$$

进行计算。于是, 可算出: $B_3 = F_3 + I_3$ 。

* 这项工作由能源部研究合同 FY-77-S-09-1011-Mod.1 支持。



测量次数	m_i	m_{i-1}	F_i	B_i
1	m_1	—	—	B_1
2	m_2	m_1	F_2	B_2
3	m_3	m_2	F_3	B_3

图3 局部滞后校正的图示说明

该算法的元素之一是常数 K ，它可额定为满刻度测量值，其单位量纲和被测量值的量纲相同。然而，对于一台给定的仪器， K 值可以进行调整，使得从平均计算的角度，由算法所得的校正值更紧密地和仪器的滞后特性相匹配。

5. 局部滞后校正联机程序的描述

图4是程序的一个简单框图。程序存放三种方程 (UP, DOWN, COMBINED) 的所有回归方程的系数，这三种方程表达了在所有参数条件下带滞后的仪器响应值。子程序由输入测量值和两个参数值开始它的常规运算。程序的下面三步根据表示仪器滞后响应的三条曲线 (UP, DOWN, COMBINED) 计算“真值”。采用在测量参数误差校正程序中所描述的参数条件之间的插值子程序来完成这项工作。在程序的这一步，局部滞后误差减少了子程序开始工作。如果最近一点测量值改变的方向不同于前一点，或者，由于电源接上，程序第一次执行，那末，从 COMBINED 方程得到的计算值作为最佳值“TRUE”。可是，如果测量值改变的方向和前面相同，则执行局部滞后算法。最

后，输出最佳值“TURE”，并存贮供下一输入点使用的若干个变量。

6. 实验结果

与前面减少误差的方法相比，要确定滞后误差局部校正的有效性是比较困难的。如果一台仪器有了滞后，理论上，这个测量值就有无限多个“真值”。实际的“真值”依赖于仪器的过去的操作情况和测量值。这个问题的真正结果是：局部滞后校正的有效性度量，必须在统计大量现象的基础上进行。

这种有效性度量的统计方法包括下列步骤：

(1) 选择一组操作条件，代表一台仪器的典型操作条件如下：

a. 开始测量值；

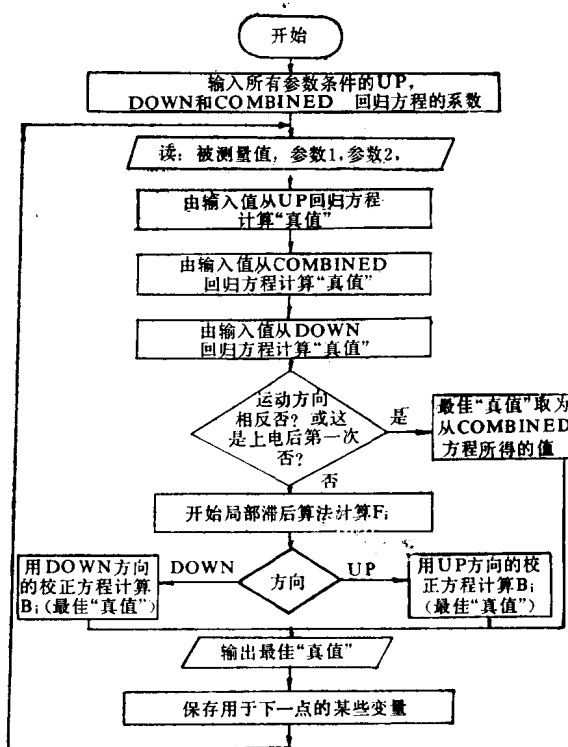


图4 联机局部滞后校正程序的一个简单框图

b. 改变测量值的步长；

c. 步数。

(2) 对每个开始测量值，估算真值的概率密度函数。

表2 局部滞后校正程序的结果

测试次数	测量范围		模拟结果 (平均值)	未经滞后校正的结果		局部滞后校正的结果(K=4)	
	开始(%FS)	结束(%FS)		(值)	(差, %FS)	(值)	(差, %FS)
1	10	50	4.1093	4.086	0.29%	4.105	.05%
2	20	50	4.1021	4.086	0.20%	4.101	.01%
3	30	50	4.0935	4.086	0.09%	4.097	.04%
4	40	50	4.0908	4.086	0.06%	4.092	.02%
5	10	30	1.9176	1.906	0.15%	1.913	.06%
6	20	30	2.6784	2.643	0.44%	2.650	.36%
7	30	40	3.8718	3.368	0.05%	3.375	.04%
				平均误差 0.18%		平均误差 0.08%	

(3) 在测量值规定改变量的末端, 确定真值的概率密度函数。

为了确定整个方法的有效性, 这种分析可分三步实现。第一, 寻找实验仪器的实际响应; 第二, 寻找未作滞后校正的响应值; 第三, 寻找在局部滞后校正程序执行后的响应值。

所选择的用来代表典型操作条件的那种条件, 是假设算法刚开始执行。同时, 假设上升方向是 UP, 然后规定以后的开始值和终止值。

对于每个开始测量值其真值的分布, 在 UP 和 DOWN 滞后曲线之间假设是正态概率函数。两条曲线间的中心点取为这个分布的均值, 并把标准差取得使 3σ 点落在曲线上。

实验仪器的“平均”响应值根据实验确定的滞后曲线用蒙特卡洛模拟法计算。

如果对于测量值没有应用任何校正滞后误差的方法, 那末, 输出的最佳值将由 COMBINED 响应曲线算得。这个计算可用前一节所讨论的减少测量参数误差的程序来完成。

最后, 对测试数据用联机误差减少程序进行分析。这些数据的一个比较如表 2 所示。模拟的结果是对实际仪器如何响应的一个估计值。未作滞后校正的数值就是根据 COMBINED 曲线由联机误差减少程序所计

算的数值。带有局部滞后的结果是由完整的联机误差减少程序所计算的。应该注意, 仪器的平均值和计算值的平均值之间的差值, 对于每次测试都得到了改进。

计算机系统的考虑

脱机回归分析程序是一个相当小的 FORTRAN 程序, 它实际上能在现有的任何通用计算机和大多数小型计算机上运行。

有两种硬件配置方案能实现这个系统。第一种配置由一台通用计算机组成(典型的情形是一台小型计算机), 它采用一个输入通道的多路扫描装置从大量的传感器或仪器中采集数据。这个系统能以某种速率一次一个地扫描各个通道, 这种速率允许在规定的时间内完成全部测量工作。由于整个系统的成本为多台仪器所分摊, 故每台仪器的费用很低。第二种硬件配置由一台用于改进一台仪器或一只传感器的精度的专用微型计算机组成。这种配置的优点是从经济和操作观点来看同其它仪器无关。因此, 这个系统能够应用于一台或数台位置相隔很远的装置。这个系统的缺点是整个成本与一种成本较高的部件有关。

在确定采用微型计算机实现联机算法的可行性工作方面, 对这种设备的典型例子做了些实验。微型计算机是 Cromemco 公司制

造的 Z₂-D 型计算机。这是一台采用 S-100 总线以及由 Z-80 型八位微处理器构成的微型计算机。虽然它是一台通用计算机,但其操作特性实际上与用同样部件组成的专用计算机系统的特性相同。

与微型计算机系统有关的两个主要问题是存储器容量和运算速度。速度必须足够快,使能实时处理测量值;存储器容量必须适当,限制在微处理器的地址空间范围内,即 64K 字以内。

程序用 FORTRAN 语言编写,以便减少软件研制时间和工作量。虽然,采用高级语言编写的程序与采用汇编语言编写的程序相比要求较多的存储量且运行速度慢,但是这些实验确实提供了最坏情况下的运算速度和存储容量。最大的程序需要 13K 字节的内存容量,每次测量使用机器的时间大约 250 毫秒。

参考文献 4 种(略)

严洪范译 江善标 白英彩校

以微处理机为基础的工业顺序 控制系统的设计自动化

J. A. Talavera M. Collado

为了便于在工程技术人员之间交换信息而研制出一种新的令人感兴趣的过程描述语言。一个计算机辅助设计系统能接受工业顺序控制过程的描述语言,并在向操作人员提问后绘出一种以微处理机为基础的控制装置的输入/输出接线图。辅助设计系统还根据这些图生成各种内部程序(当前只能生成8080微处理机用的程序)。

引 言

在设计机器或顺序装置时可以分成两个阶段:基本结构设计阶段和控制设备设计阶段。为了设计控制设备,专业工程师既要通晓所设计的机器的实际结构,又要了解整个受控过程。传递这类信息的标准和格式早已有了,而且现在已成为经典的了。例如,在那些大家较熟知的形式中间可列举出功能图(按照VDI标准第3260号)或流程图(按照DIN标准第6601号)来。其它形式,例如逻辑图或阶梯图(ladder diagrams)等,则与控制装置的工艺联系更为密切。

近来已提出不少有关规范的新的图解形式,其中包括功能逻辑框图(functional logic diagrams)、以Petri网络为基础的语言(诸如GRAFCET语言)、Noe和Nutt的宏观E-网络以及Nos和Musgrave的S.F.网络等。从另一个不同的观点出发则构想出各种微量转换图(differential transition diagrams)如Naslin所提出的那种。

其它一些建议则取短语结构语言(phrase structure language)的形式,例如Kuznetsov所提出的YARUS语言;也有取某些专用编程语言形式的,例如Rukl的

CONTROL语言或Feldman的AUTOMAT语言。

在能描述顺序过程的各种短语结构语言中,有一大批硬件描述语言(HDL),诸如Hirsh和其他人所提出的那种语言。

尽管已有许许多多不同的说明方式,人们发现仍有大量的设计时间花费在精确地澄清要设计的是什么这一点上。人们发现继续在部份地或全部地使用自然语言来描述过程。从普及的角度看,用自然语言描述过程要比用形式语言容易理解,但它有一个重大缺点:语义含糊,容易产生歧义。

鉴于上述理由,研制了一种S语言,它从自然语言出发,不断增加句法和语义上的限制条件直到含义不明确的情况完全消除为止,同时还提供了一些使描述更加简明的技巧。

如果用新的S语言所作的描述足够精确的话,在理论上就可能用计算机来进行分析,并且能在一切情况下都作出控制设备的设计,尽管实际存在的工业过程千差万别。

本文将简要地说明应用和发展上述概念所得的一些方法和结果。这些结果最初表现为一种以微型计算机为基础的设计,使这成为可能的基点在于:几乎所有的工业过程形

式上都能描述成一个顺序系统, 而与工业过程的工艺和实际变量的特性无关。

用 S-BA'SICO 语言 描述工业过程*

原来是把 S 语言设想成为工程技术人员之间交换信息的一种工具; 为了使计算机能有效地对它进行分析, S 语言还要在句法上受到相当的限制。

已研制出两种文本: 一种是基本文本, 在这种文本上已搞成一种专用于各种顺序过程的自动设计系统; 另一种是更广泛的文本, 企图使它适用于一切工业过程。

从图 1 所示的例子中可以看出, 用 S-BA'SICO 语言的描述由两部份构成: 第一部份是由输入和输出变量的说明构成的。不仅要规定 I/O 的物理概念而且要确定其类型, 这在正常情况下应当是逻辑型的或是数值型的。

在这一部份中还要说明一些其他概念, 例如用 DISPLAY (显示) 语句来说明一些要显示的变量, 又如可利用 INHIBITORS (禁止符) 的功能来使构成各种下文的说明变得很简洁。这种用法的一个实例如下:

INHIBITORS' MANUAL;

STOP ROLLERS, ...

这一说明的含义是: 如果主变量为真值,

```

      * COLD STEEL TUBES WITHDRAWAL *
      -----
      OUTPUTS
      ROLLER TABLE (S)
      TABLE UPWARDS, TABLE DOWNWARDS
      RIGHT GUIDE, LEFT GUIDE
      RIGHT EJECTOR UPWARDS, RIGHT EJECTOR DNWDS
      LEFT EJECTOR UPWARDS, LEFT EJECTOR DOWNWARDS

      INPUTS
      START ROLLERS, STOP ROLLERS
      TABLE UP, TABLE DOWN
      RIGHT EJECTOR UP, RIGHT EJECTOR DOWN
      LEFT EJECTOR UP, LEFT EJECTOR DOWN
      TUBE GOING IN, UNEXPECTED TUBE OUT
      RIGHT, LEFT
      MANUAL, LOWER TABLE
      LEFT EJECTOR, LOWER EJECTOR

      INHIBITORS 'SYSTEM ON
      INHIBITORS 'MANUAL; STOP ROLLERS, LEFT TABLE, LOWER TABLE, LEFT EJECTOR, LOWER EJECTOR

      DISPLAY ROLLER TABLE(S), TABLE UP, RIGHT EJECTOR UP, AUTO CONDITION
      DISPLAY LEFT EJECTOR UP, TUBE GOING IN, UNEXPECTED TUBE OUT

      PROCESS
      ---- MANUAL OPERATION ----
      START ROLLERS =: ROLLER TABLE(S)
      STOP ROLLERS =: TABLE UPWARDS
      LEFT TABLE =: TABLE DOWNWARDS
      TABLE UP =: TABLE DOWNWARDS
      LOWER TABLE =: TABLE UP
      TABLE DOWN =: RIGHT EJECTOR UPWARDS
      RIGHT LEFT EJECTOR, RIGHT EJECTOR DOWN =: RIGHT EJECTOR DNWDS
      RIGHT EJECTOR UP =: RIGHT EJECTOR UPWARDS
      RIGHT LOWER EJECTOR, RIGHT EJECTOR UP =: RIGHT EJECTOR DNWDS
      RIGHT EJECTOR DOWN =: LEFT EJECTOR UPWARDS
      LEFT LEFT EJECTOR, LEFT EJECTOR DOWN =: LEFT EJECTOR DNWDS
      LEFT EJECTOR UP =: RIGHT EJECTOR UPWARDS
      LEFT LOWER EJECTOR, LEFT EJECTOR UP =: RIGHT EJECTOR DNWDS
      LEFT EJECTOR DOWN =: RIGHT GUIDE:=RIGHT, LEFT GUIDE:=LEFT
      MANUAL

      ---- AUTOMATIC OPERATION ----
      =: AUTO CONDITION := 'MANUAL.ROLLER TABLE(S)

      !AUTO CONDITION.TUBE GOING IN =:
      .TUBE GOING IN.TIME IS =: TABLE UPWARDS
      .TABLE UP =: (RIGHT GUIDE=RIGHT EJECTION).(LEFT GUIDE:=LEFT EJECTION)
      .TIME 2.SS =: TABLE DOWNWARDS
      .TABLE DOWN =:

      RIGHT EJECTION, RIGHT EJECTOR DOWN =: RIGHT EJECTOR UPWARDS
      .RIGHT EJECTOR UP =: RIGHT EJECTOR DNWDS
      .TIME 2.SS =: RIGHT EJECTION
      .RIGHT EJECTOR DOWN =:

      LEFT EJECTION, LEFT EJECTOR DOWN =: LEFT EJECTOR UPWARDS
      .LEFT EJECTOR UP =: LEFT EJECTOR DNWDS
      .TIME 2.SS =: LEFT EJECTION
      .LEFT EJECTOR DOWN =:

      <UNEXPECTEDTUBEOUT+TUBEGOINGIN.(TABLEUP+LEFT EJECTORUP+RIGHT EJECTORUP)> =: 'ROLLERTABLE(S)
      END
  
```

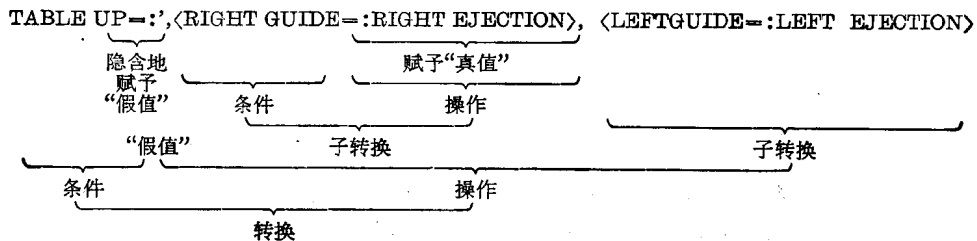
图 1 用 S-BA'SICO 语言描述一个连续无缝钢管厂中的拔管机, * 号表示是注释

* 这里只给出一个报道性的简介。在提供的参考文献中可查阅到更详尽的描述。

则所有的从变量都被禁止了(即使它们为假值),这一说明的优先级别高于任何其它的赋值规定。在这例子中,当 MANUAL 为假值时, STOPSROLLERS, ... 等将被禁止, 因为在这一场合下禁止符是一个变量的补码。如果在说明中不出现从变量, 则它将被认为是一个过程中所有变量都起作用的绝对禁止符。

本语言的重要特点之一是采用与上下文有关的语义, 这一特点在逻辑或数值表达式中也有所体现。含义与上下文有关的符号列在表 1 中。

在 I/O 变量说明区后面是实际过程的描述。它是由一组下列形式的转换(transitions)组成的: (条件) = : <操作>, 下面是一个例子。



这一语句的含义是: 如果条件(它是个逻辑表达式)成立, 则操作(它是一组赋值语句和/或子转换语句)将被执行。

转换语句可以只包含一个条件, 或者只包含一个将被连续执行的、无条件的操作。

这种语言也象许多 HDL 语言一样是以非过程性结构为基础的, 但它具有一种不同的机制 (mechanism): 闭合的“微过程” (closed microprocess), 其结构形式如下:

- 第一转换
- 第二转换
-
- 最后转换

在微过程内部, 只有在前一转换已经完成的情况下才能进行下一个中间转换。只有当微过程的最末一个转换已经完成的情况下, 第一转换才能进行第二次或以后各次。图 1 中包含着闭合微过程的三个例子。

就各种赋值语句而言, 可能有多种表达方法。可以把一个显值赋给一个显变量 (RIGHT GUARD=:RIGHT)。也可以把一个逻辑隐值 (logical implicit value) 赋予一个显变量, 只要写下它的名称就行了 (赋予真值: TABLE UPWARD; 赋予假值: 'LEFT EJECTION)。在赋予假值时变量的名称也可省略不写, 这时可以这样理解: 它就

表 1 语义与上下文有关的各种 S-BASIC 符号

符 号	上下文为数值量		上 下 文 为 逻 辑 量			特殊的上下文
	单 量 的	双 量 的	无量的	单 量 的	双量的	
-	负号 -X	减号 X-7				
+	正号 +52.20	加号 995+X			逻辑“或” A+B	
.	小数点 .32	小数点 923.19		微过程连接 .A	逻辑“与” A.B	逻辑和数值上下文 动态算符 A. 时间 × S
' (引号)			赋值为“假” '	补码或赋值为“假” 'A		在输入变量说明中的信号取反 'B

附注: 变量类型: A、B 为逻辑变量; X 为数值变量

是在所提到的前一转换的赋值表中占有对应位置的那个变量。在这种情况下只需要写出撤消算符 (deactivation operator) 就可以了。这种符号体系可大大简化用 S-BÁSICO 语言的描述, 使它看起来象一种图表, 并推出重要的结论: 没有逻辑显常数。

时间被处理成一个双量算符 (< >TIME < >S), 而它的处理层次 (evaluation hierarchy) 和逻辑交叉 (logical intersection) 一样。左边的运算数是逻辑型的, 而右边的则是数值型的, 结果是一个逻辑值。

逻辑运算数从 0 到 1 的转换标志着一次计时的开始, 而数值运算数则表明应延续的时间。象任何其它算符一样, 时间也能插入到复杂的逻辑表达式 (动态逻辑表达式) 中去。这些表达式的处理与各变量的演变历程有关。

由图 1 所示的例子还可以看出该语言的其它特点, 例如重复约定 (repeated consultation) 指示。它用惊叹号 (!) 来表示, 并把受影响的变量和跟随它的算符都隐含地插入到以后各个转换的所有条件中去。这也是另一种形式的与上下文有关的语义。

控制装置的自动设计

与在过程控制领域方面的效果相当差的

情况不同, 在现有的工程设计领域中自动设计的应用愈来愈成功。

这一领域中的主要限制因素在于工业设备中包含着大量不同的工艺, 而且有许多不同的途径来实施这些工艺。为此, 我们建立了一个模块化的自动设计系统, 同时我们认为一开始该系统不可能适用于所有工艺, 但是它能一步一步地扩展而构成一个通用系统。

那个完善化了的系统是由一组在一个 HP 21 MX 小型计算机用 FORTRAN IV 语言写成的程序构成的, 它可以分成两个方面: 第一, 与工艺无关, 对于一切应用来说均有共性, 并由图 2 中的上面两个方框组成。第二, 与实际的控制装置有关, 已经研制了一种初

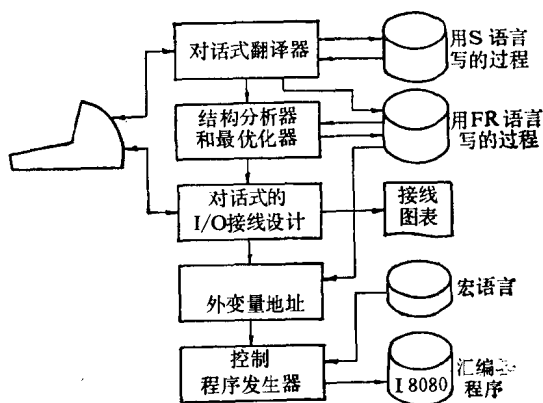


图 2 自动设计系统的功能块结构

A) 用 S-Básico Language 语言写的语句。
 !AUTO CONDITION.TUBE GOING IN =:
 *TUBE GOING IN.TIME IS =: TABLE UPWARDS

B) 翻译成中间过渡的 FR 语言

```
IF (S017) 0016
C0= Y028 A X010
IF (C0) 0016
S065=T
S017=F
0016 CN
S1=MX010
S2= S065 A Y028
S1= S002 A S001
TIMES001, 1
IF (C0) 0017
Y002= T
S066=T
S065=F
0017 CN
```

C) 在 8080 汇编器中生成的程序。

```
LDA S017
ORA A
JZ L0016
LDA Y2502
MOV B,A
LDA X2501
ANA B
ORA A
JZ L0016
MVI A,000H
STA S065
MVI A,000H
STA S017
LDA X2501
CMA
ANI 000H
STA S001
LDA S065
MOV B,A
LDA Y2502
ANA B
STA S002
MOV B,A
LDA S001
ANA B
STA S001
ORA A
JZ L0017

L0016 ORA A
JZ L0016
LDA A
JZ L0016
LHLD A
XRA A
H
CNZ L
CNZ L
JNZ L
JNZ L
MVI A,000H
JMP L
LXI H,1
SHLD 4187
MVI A,000H
STA 4235
JMP L
XRA A
STA 4235
XRA A
ORA A
JZ L0017
MVI A,000H
STA Y2402
MVI A,000H
STA S066
MVI A,000H
STA S065
L0017 EQU $
```

图 3 程序生成实例

始文本，它适用于以 I 8080 微处理机为基础的微型计算机。

该系统是一种对话式系统，能接受用 S-BÁSICO 语言编写的过程描述语句，被作为正确语句接受下来的句子翻译成一种中间的宏语言 FR。这一宏语言是用来作为设计

的两个阶段之间的联系环节。作为一个例子，图 3 所示为某些过程语句翻译成 FR 的情况。

一旦翻译成 FR 后，结构分析器和最优化器就要对它进行检查，目的在于从句法上描述得很好的过程中找出可能导致控制装置

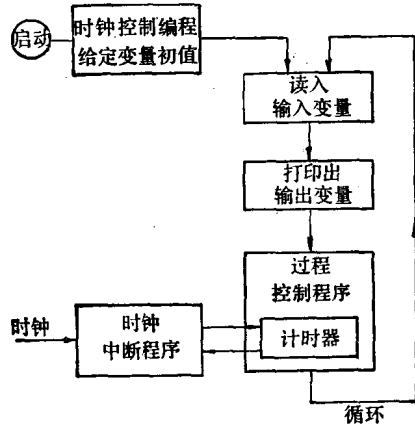
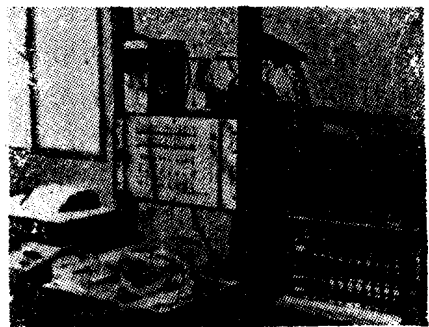


图 4 为微型控制生成程序的功能块的结构图



照片 I 检验自动设计的设备

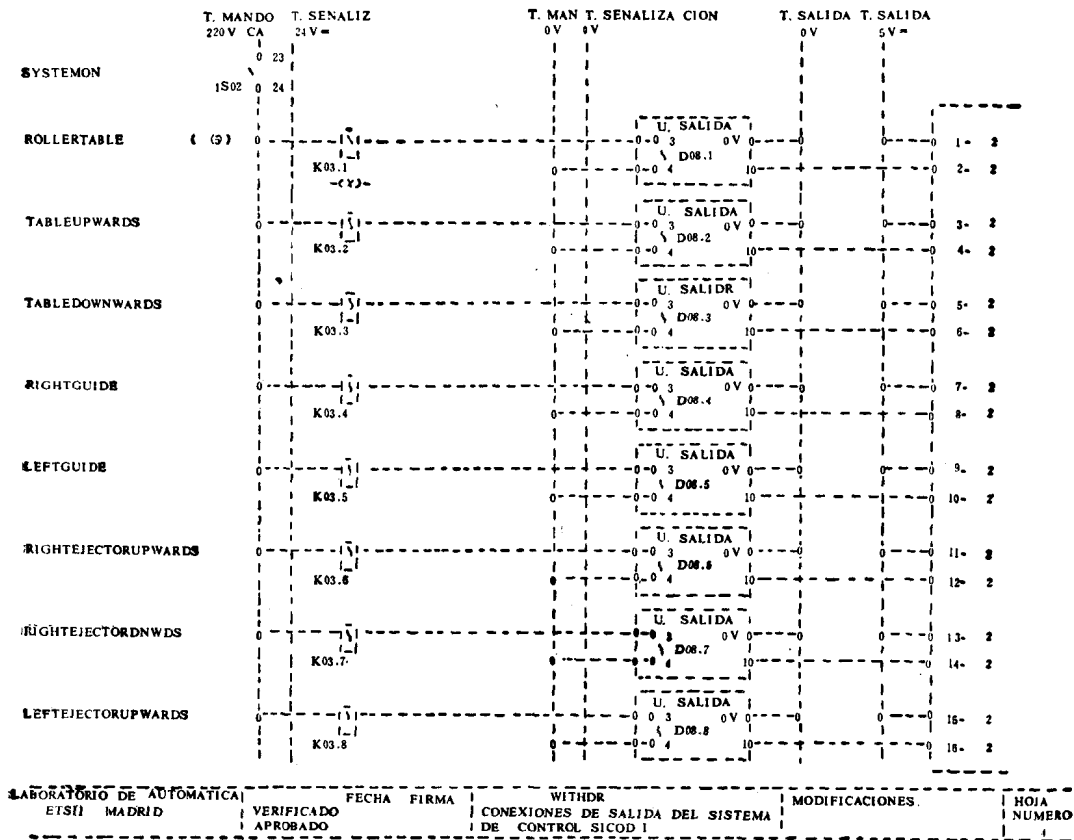


图 5 用计算机绘出的输出接线图表的一个实例

动作不正确的缺陷来, 并把所有找出来的缺陷都告诉操作人员, 而且就由他来决定是接受还是摒弃过程的这些描述。这一功能块还能系统地消除多余的指令。

对过程描述进行分析并通过向操作人员询问有关设备的实际结构, 设计系统的第二部份就产生出控制装置的输入和输出的外接图表。在图 5 中出现的图表按 DIN 40719 标准使用图形符号和名称并可用一个标准打印机打印出来。所列举的图表是专为我们实验室所建造的一个控制装置模型而编制的, 该模型示于照片 I 中。

用 FR 写成的描述和接线图表相结合为各外变量指定恰当的地址。接下来是一个控

制程序发生器, 它把 FR 语言翻译成 Intel 8080 汇编语言。所产生的程序具有图 4 所示的结构, 它在循环重复执行方式 (cyclical execution) 下与许多商用 PLC 相似。

今后打算

我们拟从两个方面继续发展自动设计技术。一方面要扩充工艺模块的数目, 以便包括硬接线形式的电子逻辑和 PLC (BBC 的 Procontio-e); 另一方面要扩展系统, 以便包括具有模拟变量的各种过程。

参考文献 18 种(略)

乐茂生译 白英彩校