第6卷 增刊2 控制理论与应用 Vol.6 Suppl.I. 2
1989年4月 CONTROL THEORY AND APPLICATIONS April, 1989

控制系统CAD软件包的误差分析*

喻铁军 戴冠中 王为真

（西北工业大学计算机系，西安）

摘 要

本文讨论了一种有效的软件误差分析法一摄动法，以及它在控制系统CAD软件包误差分析中的应用．文中首先给出了应用摄动法求任意数制有效位数的估算公式，然后利用本方法对MCSS**软件包中的主要算法进行了误差分析．经在PDP—11/23计算机上反复计算表明，用本方法来分析软件误差是有效的，所得的结果与实际情况相符．不仅如此，利用本方法还改进了矩阵奇异性的判据以及迭代算法的终止条件，从而改善了MCSS的软件质量．

一、引 言

软件的计算误差是软件研制人员和软件用户所共同关心的问题．软件研制人员在设计软件时，不但要考虑软件所占的存贮量和计算量，更要考虑软件的计算精度．

一个计算软件运行时，每一次运算操作均有可能产生运算误差[1,2]，运算误差的积累最终将导致计算结果有效位的损失．然而，对于一个复杂的计算软件，怎样求得计算结果的有效位，即确定计算结果的精度，是一个相当困难的问题．本文所研究的摄动法是解决本问题的一种有效的方法．利用摄动法，不仅能给出计算结果的有效位数，还能给出计算结果的置信区间．

二、摄 动 法 原 理

摄动法的基本思想是：对一个数值计算过程软件，我们对每步运算的结果进行随机摄动，得到该计算过程软件在计算机上实现时的结果集，然后再用该结果集来估计计算值的精度．

设一个实际的数值计算过程为

$$\text{procedure}(d,\ r,\ +,\ -,\ \times,\ \div,\ \text{funct}), \qquad (1)$$

其中，$d\subset R$（实数集）为原始数据集，$r\subset R$为计算过程的结果，funct为计算机所具有的数学函数．

为了简便起见，我们假设r是唯一的．上述数值计算过程在计算机上实现时，相应软件的算法可表示为：

* 中国科学院科学基金资助的课题．

** 西北工业大学研制的“多变量控制系统计算机辅助设计软件包（简称MCSS）”．

本文于1986年12月23日收到．1988年6月29日收到修改稿．

$$\text{PROCEDURE}(D,\ R,\ +,\ -,\ *,\ /,\ \text{FUNCT}) \tag{2}$$

其中，$D\subset F$（浮点数集），$R\subset F$，FUNCT为计算机库函数。

我们知道（2）是过程（1）在计算机上的一个实现，但不是唯一的一个，而是（1）式实现集中的一个。

于是，设Σ为对应于（1）式实现的结果集，对Σ集中的元素R_i，我们认为是一个均值为$\overline{R}$，标准差为δ的高斯随机变量[3]，则采用t－分布区间估计有：准确结果r的显著性水平为p、置信区间为I_C的概率为：

$$P_r[r\in I_C]=1-p, \tag{3}$$

式中，$I_C=[\overline{R}-t_p\cdot\delta/\sqrt{N},\ \overline{R}+t_p\cdot\delta/\sqrt{N}]$，其中$t_p$为自由度为$N-1$的$t$－分布值。

于是，当取$R_i\in\Sigma$且$R_i\in I_C$为过程（1）的计算值时，其计算结果的绝对误差满足如下不等式：

$$\delta R_i=|R_i-r|\leqslant 2\cdot t_p\delta/\sqrt{N} \tag{4}$$

Maille指出[4]计算值的最好估计是其均值$\overline{R}$，于是把$\overline{R}$作为计算结果时，有：

$$\delta\overline{R}=|\overline{R}-r|\leqslant t_p\delta/\sqrt{N}. \tag{5}$$

根据实际计算，一般情况下，$N=3$时即可给出良好的结果。于是取$N=3$和$p=5\%$时，有$t_p=4.303$。从而真值r的置信区间为

$$I_C=[\overline{R}-2.48434\delta,\ \overline{R}+2.48434\delta]. \tag{6}$$

对$R_i\in\Sigma$且$\in I_C$时，有

$$\delta R_i\leqslant 2t_P\delta/\sqrt{N}\approx 5\delta. \tag{7}$$

以$\overline{R}$作为计算结果时，有

$$\delta\overline{R}\leqslant t_P\delta/\sqrt{N}\approx 2.5\delta. \tag{8}$$

三、利用摄动法分析计算软件误差

1．计算值有效位的确定

为了使讨论具有一般性，我们研究任意数制（β进制）的情况。设某计算过程的计算结果的真值为r，近似计算值为$\overline{r}$，计算绝对误差为$\delta_r=|r-\overline{r}|$，其规范形式分别为

$$r=\pm 0\cdot r_1r_2\cdots\times\beta^{E_1},$$

$$\overline{r}=\pm\ 0\cdot\overline{r}_1\overline{r}_2\cdots\times\beta^{E_2},$$

$$\delta_r=0\cdot d_1d_2\cdots\times\beta^{E_3},$$

其中，r_i、$\overline{r}_i$和d_i为正整数，且

$$\begin{aligned}&1\leqslant r_1<\beta,\quad 1\leqslant\overline{r}_1<\beta,\quad 1\leqslant d_1<\beta\\&0\leqslant r_i<\beta,\quad 0\leqslant\overline{r}_i<\beta,\quad 0\leqslant d_i<\beta\ \ (i=2,3,\cdots)\end{aligned} \tag{9}$$

于是，在β进制的情况下，$\overline{r}$的有效位数为

$$C = E_2 - E_3。 \tag{10}$$

又令

$$C^* = \log_\beta |\overline{r}|/\delta_r = \log_\beta \frac{0\cdot\overline{r_1}\,\overline{r_2}\cdots}{0\cdot d_1 d_2\cdots} \times \beta^{E_2 - E_3}$$

$$= E_2 - E_3 + \log_\beta \frac{0\cdot\overline{r_1}\,\overline{r_2}\cdots}{0\cdot d_1 d_2\cdots}$$

注意到（9）式，则有

$$-1 < \log_\beta \frac{0\cdot\overline{r_1}\,\overline{r_2}\cdots}{0\cdot d_1 d_2\cdots} < 1。$$

又考虑到C为整数时，有

$$C = E_2 - E_3 = C^* - \log_\beta \frac{0\cdot\overline{r_1}\,\overline{r_2}\cdots}{0\cdot d_1 d_2\cdots}$$

$$= \begin{cases} \mathrm{AINT}(C^*) + 1 & \text{当} 0\cdot\overline{r_1}\,\overline{r_2}\cdots < 0\cdot d_1 d_2\cdots \text{时} \\ \mathrm{AINT}(C^*) & \text{当} 0\cdot\overline{r_1}\,\overline{r_2}\cdots \geqslant 0\cdot d_1 d_2\cdots \text{时} \end{cases}, \tag{11}$$

其中，AINT(·)为取整函数。

显然当$C<1$时，计算结果无有效位。

（11）式即为计算结果有效位数的计算公式。直接利用(11)式是困难的，因为真值r无法确定，误差δ_r也就无法知道。然而，我们可利用上节的（4）或（5）式来估计误差δ_r，从而可利用（11）式来估算计算结果的有效位数。

2．方法的程序实现

上面讨论了计算结果的绝对误差和有效位数的估算方法，这些估算公式是比较容易在计算机上实现的。我们在PDP－11/23计算机上编制了计算绝对误差和有效位数的程序，并应用这些程序对MCSS软件包中的一些主要算法进行了分析，所得结果和实际情况非常相符。

在计算机上实现摄动法时，主要解决了下列问题：

1）结果集的选取。我们知道，结果集Σ的元素一般是很多的，要全部求出这些元素是不可能，也没有必要。我们采用的方法是：找出一个反映结果集Σ特征的子集，然后用该子集来估计计算值的绝对误差或有效位数；而子集产生的方法是逐步产生Σ集中的元素，直到结果的有效位数达到一个稳定状态。

2）摄动量的产生。为了求得结果集Σ，我们需要对每次运算结果进行随机摄动。设计算结果为(PDP－11/23的浮点数是一个带隐含位的双字节数，其中23位尾数位，8位指数

位，1位符号位）

$$x = \pm 0.1 d_1 d_2 \cdots d_{23} \times 2^E$$

则产生的摄动量为

$$dx = \mathrm{sign}(x) \times 2^{E-24},$$

式中，sign(·)为取符号函数.

在计算机上实现时，首先产生一个在（0，1）区间均匀分布的随机变量y，然后根据y值按下规则对结果进行摄动：

$$\overline{x} = \begin{cases} x - dx & \text{当} y < 0.5\text{时}, \\ x & \text{当} y = 0.5\text{时}, \\ x + dx & \text{当} y > 0.5\text{时}, \end{cases}$$

这样产生的随机摄动，不难知道其均值为零，这对浮点运算来说是合适的. 另外，当计算过程的计算量较大时，根据中心极限定理，每步运算结果经过摄动后所得的最后结果，将近似服从高斯分布，这保证了第二节假设的合理性.

四、摄动法在控制系统CAD软件包设计中的应用

1. 软件误差的估计

我们采用上面讨论的摄动法，在PDP11/23计算机上对MCSS软件包的主要算法进行了误差分析. 作为例子，表1给出了矩阵求逆和奇异值分解的计算结果. 从这些结果可以看出，用摄动法来估计计算结果的精度是有效的. 另一方面，我们也看到了MCSS的算法是可靠的，一般情况下均能给出有效位为5至6位的计算结果.

表 1　误差估计结果（矩阵求逆误差 $\times 10^{-8}$，奇异值分解误差 $\times 10^{-5}$）

	N	2	3	4	5	6	7	8	9	10
矩阵求逆	E_a	0.1184	0.3059	2.4505	0.7695	2.1700	2.9630	0.2454	1.0130	0.5957
	E_e	0.3437	0.8556	10.8269	2.4970	2.0958	8.1412	0.4343	1.4887	1.5026
	N_S	6	6	5	6	6	5	6	6	6
奇异值分解	E_a	2.0728	2.2376	2.6553	3.5009	3.0807	4.1326	4.0973	5.2058	5.5883
	E_e	3.0719	4.7234	6.4699	5.9938	6.2000	9.1933	8.5385	10.538	10.949
	N_S	6	6	6	6	6	6	6	6	6

表中：N为矩阵阶数，N_S为计算值的有效位数，E_a为用双精度运算结果作为真值所得的计算值绝对误差，E_e为采用摄动法所得计算值的估计绝对误差.

2. 判据条件的改善

在计算软件的设计中，许多地方都要遇到对某些计算条件进行判别，以确定其程序的流程. 由于计算机的运算操作总是在有限位字长的条件下进行的，这样每次运算都有可能产生

误差，随着运算操作的增加，运算误差也会随着积累．这些积累的误差轻则影响最终结果的精度，重则产生完全错误的流程．所以，怎样选择合适的判据条件，目前是计算软件设计中的一个重要问题．把摄动法应用到确定判据条件时，将会使软件的计算结果得到很大的改善．我们知道，矩阵的奇异性是控制系统CAD软件包设计中遇到最多的一个判据条件．下面，我们就以判别矩阵的奇异性为例来进行说明．

矩阵的奇异性是由其行列式是否为零来确定的．由于计算机的计算误差，采用上述判据条件来判断，几乎是不可能的．在软件的设计中，一般采用如下判据：

判据1　设计算机的零点为EPS，则当$|\det(A)|\leqslant$EPS时，认为A是奇异的；否则，认为是非奇异的．

经过多次验算，我们发现随着矩阵阶数的增加，采用上述判据所得的结果非常不理想．下面，我们给出一个新的判据．

判据2　当一个矩阵的行列式是零或无有效位时，则认为该矩阵是奇异的；否则，认为是非奇异的．

表2　给出了分别采用判据1和判据2来判断矩阵奇异性所得的结果．矩阵元素除最后一行外，均是在（－100，100）上均匀分布的随机数，而最后一行是由矩阵前$n-1$行线性组成的．显然，这样的矩阵必是奇异的．从表2可以看出，采用判据1来判断矩阵的奇异性，几乎得不到正确的结果（特别是矩阵阶数较大时）；而采用判据2来判断矩阵的奇异性时，其结果是很理想的．

表 2　分别采用两种判据条件判矩阵奇异性的结果

零　点	N	2	3	4	5	6	7	8	9	10
1·E－6	m_1	8	4	3	0	2	4	1	0	1
	m_2	10	10	10	10	10	10	10	10	10
1·E－7	m_1	2	2	2	0	0	2	0	0	0
	m_2	10	10	10	10	10	10	10	10	10

表中，N为矩阵阶数，m_1为采用判据1正确判断矩阵奇异性的次数，m_2为采用判据2正确判断矩阵奇异性的次数（每阶矩阵均计算了十组数据）．

3．迭代终止条件的改善

在软件的设计中，经常会遇到许多迭代算法，它们在迭代过程中满足一定的迭代终止条件后就停止．

设迭代序列为$\{x_n\}$，且$x_n\to x$，传统的迭代终止条件为

$$\|x_n-x_{n-1}\|\leqslant\varepsilon \quad 或 \quad \|x_n-x_{n-1}\|\leqslant\varepsilon\|x_n\|,$$

其中，ε为计算机的相对零点．

原MCSS软件包的设计中，均采用上述迭代终止条件．在对MCSS作了大量数值计算和

软件误差分析后，我们发现把 ε 取为固定不变的计算机相对零点并不合理．我们对上述条件作了一些修改，修改后的传统迭代终止条件为

$$\|x_n - x_{n-1}\| \leqslant 10^{-N_s} \text{ 或 } \|x_n - x_{n-1}\| \leqslant 10^{-N_s}\|x_n\|,$$

其中，N_s 为计算结果的平均有效位数．

又设 x 满足如下条件：$F(x)=0$，则 Wilkinson 提出的最佳迭代终止条件[5]为 $F(x_n)=0$，即当 x_n 满足条件 $F(x_n)=0$ 时，就停止迭代，把 x_n 作为其最终迭代结果．

从表面上看，以上三种迭代终止条件中，似乎采用最佳迭代终止条件最为合适．但实际情况并不是这样，而是由于计算误差，要判断最佳迭代条件是否满足是比较困难的．所以，尽管最佳迭代终止条件在六十年代初就提出来了，但在实际计算中，采用得很少．本文所给出的摄动法能解决上述困难．采用摄动法，修改后的最佳迭代终止条件为

表 3　采用传统迭代终止条件的计算结果

	符号函数法		试探—迭代法	
N	ITM	ERM ($\times 10^{-6}$)	ITM	ERM ($\times 10^{-6}$)
2	5	0.3263354	11	0.2525747
3	5	3.997992	10	440.8254
4	8	0.5657785	13	3.100187
5	7	0.4897863	15	13.34628
6	8	1.196657	15	54.24961
7	9	0.9019004	15	218.5133
8	9	1.423527	15	37.785
9	13	3.988537	15	162.1268
10	10	0.6732903	15	38.85891

表中：N为系统阶数，ITM为平均迭代次数，ERM为代数 Riccati 方程的平均误差。
（本程序中我们规定的最大迭代次数为15次）

表 4　采用修改最佳终止条件的结果

	符号函数法		试探—迭代法	
N	ITM	ERM ($\times 10^{-6}$)	ITM	ERM ($\times 10^{-6}$)
2	4	0.2413988	3	0.2063811
3	4	1.852690	3	14.1406
4	4	0.4053116	3	0.3376044
5	4	1.817316	3	0.5621985
6	4	5.143643	3	1.281593
7	5	1.291839	3	0.7366318
8	5	1.301523	3	0.7179653
9	5	1.466661	3	0.6100431
10	5	0.3932603	3	0.5139876

表中：各列的含意同表 3

表 5　采用修改传统终止条件的结果

	符号函数法		试探—迭代法	
N	ITM	ERM ($\times 10^{-6}$)	ITM	ERM ($\times 10^{-6}$)
2	5	0.3263354	4	0.257045
3	5	3.997992	6	42.67913
4	5	0.3666617	4	0.3451482
5	5	0.5804077	4	0.5993694
6	5	1.747834	4	2.193937
7	6	0.817866	4	0.9134564
8	6	1.383169	4	0.9642943
9	6	2.053417	5	0.597855
10	6	0.4510395	4	0.6527454

表中：各列的含意同表 3

设 $\rho_n = F(x_n)$，则当 $\rho_n = 0$ 或 $C\rho_n < 1$（$C\rho_n$ 为 ρ_n 的有效位数）时，停止迭代，取 x_n 为 x 的计算值．

表3～5分别给出了采用传统迭代终止条件、修改后的传统迭代终止条件以及修改后的最

佳迭代终止条件来解代数 Riccati 方程的结果，这里解代数 Riccati 方程采用了两种方法，即符号函数法和试探一迭代法[6]。从表中可看到，采用修改后的传统迭代终止条件和最佳迭代终止条件，与采用传统迭代终止条件相比，不但能减少迭代次数，提高运算速度，而且还能改善结果的精度，从而提高了程序算法的有效性和可靠性。

五、结　论

本文讨论的摄动法，是一种非常有效的软件计算误差的估算方法。把它应用到控制系统CAD软件包的设计中，一方面能对软件包算法的计算误差给出很好的估计，给算法的选择提供有力的依据；另一方面还能改善软件的判据条件和迭代终止条件等等，从根本上提高软件的运算速度和可靠性。经大量数值计算表明，采用摄动法改善后的MCSS软件包，无论是在软件的有效性和可靠性上，还是在软件的运算速度方面，都比原MCSS软件包有所改善和提高。

主要参考文献

〔1〕 Wilkinson, J. H., The Algebraic Eigenvalue Problem, Oxford Univ. Press, Claredon,(1965).

〔2〕 Miller, W., Software for Roundoff Analysis, ACM TOMS, 1:2, (1975), 108—128.

〔3〕 Tsao, N.K., On the Distributions of Significant Digits and Roundoff Errors, Comm. ACM, 17:5, (1974), 269—271.

〔4〕 Vignes, J., New Methods for Evaluating the Validity of the Results of Mathematical Computation, Mathematics & Computers in Simulation XX, (1979), 227—249.

〔5〕 Bois, P., and Vignes, J., A Software for Evaluating Local Accuracy in the Fourier Transform, Mathematics & Computers in Simulation XXII, (1980),140—150.

〔6〕 干华雷，LQGSP——线性二次型高斯控制系统计算机辅助设计软件包，信息与控制，6，(1982)，11—16.

Error Analysis for Control Systems CAD Software Packages

Yu Tiejun, Dai Guanzhong, Wang Weizhen

(Department of Computer Northwestern Polytechnical University, Xian)

Abstract

An effective error analytical method, the perturbation method, and its application to the control systems CAD software packages are discussed in this paper. Using the perturbation method, we first present the formula for estimating the significant digits in any numeration systems, and then analyse the errors of the main algorithms in MCSS. The results of repeated computations in PDP—11/23 show that the perturbation method is valid for software error analysis. Moreover, the detection condition of the matrix singularity and the stopping rule of the iterative algorithms are improved after using the perturbation method, and, hence, the quality of MCSS is improved.