数值计算方法

徐欢乐
计算机与网络安全学院，9A304
xuhl@dgut.edu.cn

为什么要开设这个课呢?

- 计算机的出现为大规模的数值计算创造了条件，集中而系统的研究适用于计算机的数值方法变得十分迫切和必要；
- 数值计算是计算机处理实际问题的重要手段，数值算法是进行科学计算必不可少的起码常识；
- 通过对数值计算的讨论，能够使人们掌握设计数值算法的基本方法和一般原理，为在计算机上解决科学计算问题打下基础。

学习数值计算方法的准备知识

- 数学分析(或微积分)
- 线性代数
- 数学软件

我们先来看看学过的一些知识和问题：

$$3x+5=0 \qquad x=-5/3$$

$$2x\wedge 2+3x-8=0 \qquad x_{1,2}=\frac{-3\pm\sqrt{3*3-4*2*(-8)}}{2*2}$$

如果：三次方程呢？
n次方程呢？

$$x^6+3x^5+9x^3+7x^2+x=0$$

X=????

多项式次数一般超过5次，它的根一般已经不能用公式表示

非线性方程呢？

$$e^x - x^2 + 3x - 2 = 0$$

X=???

问题二：

$$\begin{cases} 2x_1 - x_2 - x_3 = 2 & ① \\ x_1 + x_2 - 2x_3 = 1 & ② \\ 2x_1 + 8x_2 - 6x_3 = 6 & ③ \end{cases}$$

$$\begin{cases} x_1 + x_2 - 2x_3 = 1 & ① \\ 2x_1 - x_2 - x_3 = 2 & ② \\ x_1 + 4x_2 - 3x_3 = 3 & ③ \end{cases}$$

$$\begin{cases} x_1 + x_2 - 2x_3 = 1 & ① \\ -3x_2 + 3x_3 = 0 & ② \\ 3x_2 - x_3 = 2 & ③ \end{cases}$$

$$\begin{cases} x_1 + x_2 - 2x_3 = 1 \\ -3x_2 + 3x_3 = 0 \\ 2x_3 = 2 \end{cases}$$

方程的个数是3个

那如果是20个呢？

用什么方法解？

一般的呢？

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = b_2 \\ \quad \dots \qquad \dots \qquad \dots \\ a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nn}x_n = b_n \end{cases}$$

我们在数学分析中学过：用克莱姆法则能解决

$$x_i = \frac{|A_i|}{|A|}, i = 1, 2, \cdots, n.$$

n=20

理论上很“漂亮”的Cramer法则在计算机上并不适用！

问题三

$$A = \begin{pmatrix} 1 & 2 & 2 \\ 1 & 0 & 2 \\ 0 & 1 & 1 \end{pmatrix}$$

求A的特征值和特征向量

$$\mathrm{A} = \begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \cdots & \cdots & \cdots \\ a_{n1} & \cdots & a_{nn} \end{pmatrix}$$

求A的特征值和特征向量?

问题四

两个例子

美国的人口普查每**10**年举行一次，下表列出了从**1940**年到**1990**年的人口（按千人计）

年	1940	1950	1960	1970	1980	1990
人口（千人）	132165	151326	179323	203302	226542	249633

能否利用这些数据合理地估计人口的数量，比如**1965**年的人口，甚至**2015**年的人口。

问题5 数值积分

$$\int_0^1 x dx = ?,$$

$$\int_0^1 \frac{\sin x}{x} dx = ?,$$

$$\int_0^1 x^2 dx = ?$$

$$\int_0^1 \sin x^2 dx = ?$$

$$\int_{-1}^1 \cos x dx = ?,$$

$$\int_{-1}^1 \cos x^2 dx = ?,$$

$$\int_0^1 e^x dx = ?,$$

$$\int_0^1 \frac{1}{\ln x} dx = ?,$$

$$\int_0^1 e^{-x} dx = ?$$

$$\int_0^1 e^{-x^2} dx = ?$$

第一章　绪　论

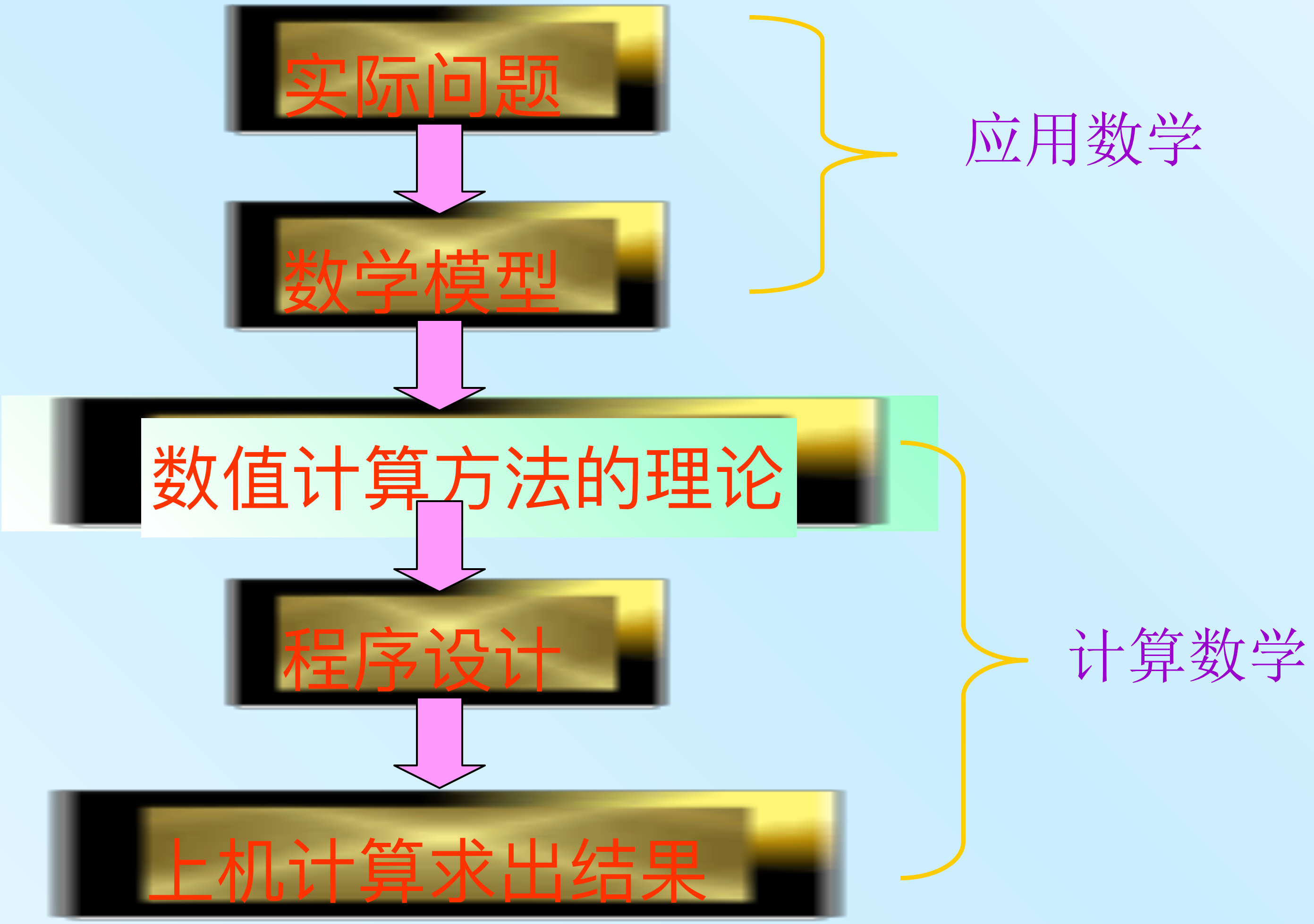
1.1　科学计算的一般过程

1.2　数值计算方法的研究内容和特点

1.3　计算过程的误差及其控制

1.1　科学计算的一般过程

研究求数学问题近似解的方法和过程



•一个科学计算过程主要包括如下几个环节：

1.1.1 数学建模：将工程问题数学化

应用有关学科的知识和数学理论，将实际工程问题，用精炼准确的数学语言对其核心部分进行描述并给出数学模型，这一过程常称为数学建模。一个好的数学模型符合下列两个方面的要求：

1.数学模型能真实准确的反映实际工程问题的本质；

2.数学模型所用的数学算法能再计算机上实现

1.1.2 对数学问题给出数值计算方法

例1 求解线性方程组 $Ax = b$ 是数值问题

求解二次方程 $ax^2 + bx + c = 0$

我们能给出怎样的算法？

什么样的算法才能是好算法呢？

1.算法的速度（算法的收敛速度）；

2.算法所得到结果的精确度；

3.算法所占用的计算机资源；

1.1.3 对数值计算方法进行程序设计

1.熟练掌握一门语言，比如c语言，c+ matlab, fortran

我们在以后用到的是matlab

将数值问题机器化

1.1.4 上机计算并分析结果

数值模拟物理过程，分析计算结果的可靠性，必要时重复上述过程。

其中算法设计是数值计算的核心内容。数值计算方法针对来源于科学与工程中的数学模型问题，介绍计算机上常用的数值方法的算法设计思想并进行算法分析。

§1.2 数值计算方法的研究内容与特点

- 数值计算：常称为数值分析或计算数学或计算方法。主要是研究如何运用计算工具（如计算器、计算机等）去获得数学问题的数值解的理论和方法。

对那些在经典数学中，用解析方法在理论上已作出解的存在，但要求出他的解析解又十分困难，甚至是不可能的这类数学问题，数值解法就显得不可缺少，同时又十分有效。

实践表明：计算方法正在日趋明显地成为数学与计算机科学的交叉科学。

- 数值计算研究内容：对如下五类问题探索数值求解方法及其与算法有关的理论分析

(1) 数值代数（线性方程组、非线性方程及方程组的数值解法）

(2) 数值逼近（各种函数逼近问题的数值解、数值积分和微分）

(3) 常微分方程数值解法

(4) 最优化理论和方法

数值计算的根本任务就是研究算法

算法：从给定的已知量出发，经过有限次四则运算及规定的运算顺序，最后求出未知量的数值解，这样构成的完整计算步骤称为算法。

运算量(计算量)：一个算法所需的乘除运算总次数

计算量是衡量一个算法好坏的重要指标!

- 研究数值算法的任务主要有：

(1) 构造计算机上可执行的算法

计算机上可执行的运算： 四则运算 逻辑运算

计算方法：把求解数学问题转化为按一定次序只进行加、减、乘、除等基本运算——数值方法。

(2) 构造计算复杂性好的算法

尽可能提高数值方法的计算速度和少占存贮空间。

(3) 构造可靠性好的数值方法

选择或研制能达到“数值问题”要求的计算精度的数值方法，为此须研究数值问题的性态及数值方法的稳定性。

例 1.1.1 例如：计算3次多项式 的函数值

$$p_3(x) = a_3x^3 + a_2x^2 + a_1x + a_0$$

直接计算需要6次乘法，3次加法。如果作如下改变:

$$p_3(x) = a_3x^3 + a_2x^2 + a_1x + a_0$$

$$= ((a_3x + a_2)x + a_1)x + a_0$$

只有3次乘法，3次加法。这个算法称作：秦九绍算法。

例1.1.2 解线性方程组 $Ax=b,$

其中，$A=(a_{ij})_{n\times n}, x=(x_1,x_2,\ldots,x_n)^T, b=(b_1,b_2,\ldots,b_n)^T.$

▶ 克兰姆(Cramer)法则：

$$x_i=\frac{|A_i|}{|A|}, i=1,2,\cdots,n.$$

运算量(乘除)： $(n+1)\times n!\times(n-1)\approx(n+1)!\times(n-1)$

▶ 高斯消元法(Gauss)：

运算量(乘

理论上很“漂亮”的Cramer法则

在计算机上并不适用！

取 $n=20$

Cramer: (2 0 +×1) ! (≈2 0 × 1 1)×3 9. ≈1 3 0 1.7 8 (10^12 /)

Gauss: 3060次

例1.1.3 计算积分 $I_n = \int_0^1 \frac{x^n}{5+x} dx \quad n = 0,1,\cdots,20$

的递推关系式，并研究它的误差传递。

解： $I_n + 5I_{n-1} = \int_0^1 \frac{x^n + 5x^{n-1}}{5+x} dx = \int_0^1 x^{n-1} dx = \frac{1}{n}$

可得算法：$\begin{cases} I_0 = \ln 6 - \ln 5 \approx 0.1823 \\ I_n = \frac{1}{n} - 5I_{n-1} (n = 1,2,\ldots\ldots,20) \end{cases}$

具有以下的性质：

1. $I_n > 0$

2. I_n单调递减

3. $\lim\limits_{n\to\infty} I_n = 0$

4. $x \in [0,1]$时 $\dfrac{x^n}{6} < \dfrac{x^n}{5+x} < \dfrac{x^n}{5}$

$$\frac{1}{6(n+1)} < I_n < \frac{1}{5(n+1)}$$

- 将问题可算化的手段：将问题可算化是设计一个算法的第一步

(1) 用有限维空间代替无限维空间

(2) 用有限过程代替无限过程

(3) 用简单问题代替复杂问题

(4) 扰动分析：估计误差或精度

§1.3 计算过程中的误差及其控制

数值方法中的计算公式及参与运算的数，都和数学中的一般情况有所不同，即

计算公式中的运算必须是在计算机上可执行的运算

参与运算的数必须是有限小数或整数

因此，数值方法中的取数和运算往往会出现误差，算得的结果（称为计算值）一般也为近似值。

在任何科学计算中，其解的精确性总是相对的，而误差则是绝对的。

1.3.1、误差的种类及来源

一个物理量的真实值和我们算出的值（即计算值）往往存在差异，它们之差称为误差。

模型误差

在建立数学模型过程中，要将复杂的现象抽象归结为数学模型，往往要忽略一些次要因素的影响，而对问题作一些简化，因此数学模型和实际问题之间有一定的误差。

观测误差

在建模和具体运算过程中所用的数据往往是通过观察和测量得到的，受观测方式、仪器精度以及外部观测条件等多种因素限制，不可能获得精确值，由此而来产生的误差。

截断误差

由于计算机只能完成有限次算术运算和逻辑运算，因此要将有些需用极限或无穷过程进行的运算有限化，对无穷过程进行截断，这就带来误差。

例:

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \ldots\ldots$$

$$\sin x = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \ldots\ldots$$

$$\ln(1+x) = x - \frac{x^2}{2!} + \frac{x^3}{3!} - \frac{x^4}{4!} + \ldots\ldots$$

Taylor展开

若将前若干项的部分和作为函数值的近似公式，由于以后各项都舍弃了，自然产生了误差

舍入误差

在数值计算过程中还会遇到无穷小数，因计算机受到机器字长的限制，它所能表示的数据其位数只能是有限的，如按四舍五入规则取有限位数，由此引起的误差

$$\pi = 3.14159265\ldots\ldots$$

$$\pi \approx 3.1415927$$

$$\sqrt{2} = 1.414213562\ldots\ldots$$

$$\sqrt{2} \approx 1.4142136$$

$$\frac{1}{3!} = \frac{1}{6} = 0.166666666\ldots\ldots$$

$$\frac{1}{3!} \approx 0.16666667$$

另外还有过失误差，这类误差是由于模型错误或方法错误所引起的，一般可以避免。

结论：误差是不可避免的

在实际问题中求精确解是没有意义的，求近似解是正常的。问题是如何尽量减少误差，提高精度。

在 4 种误差中，前 2 种是客观存在的，后 2 种是计算方法引起的。数学模型一旦建立，进入具体计算时所考虑和分析的就是截断误差和舍入误差。因此本课程只涉及这 2 种误差。

经过大量的运算之后，积累的总误差有时会大得惊人，因此如何控制误差的传播也是数值方法的研究对象。

1.3.2、误差与有效数字

设x^*为准确值，x为x^*的一个近似值。称

$$e = x^* \quad x$$

为近似值x的绝对误差，简称误差.

因为准确值 x^*往往是未知甚至是无法知道的，

因此$e = x^* \quad x$往往也无法求出。

而只能知道$e = x^* \quad x$绝对值的某个上界，即

数值 称为近似值x的一个绝对误差限或误差限，

显然 $\varepsilon > 0$

且 $x - \varepsilon \le x^* \le x + \varepsilon$

有时也表示为 $x = x^* \pm \varepsilon$

例 $x = 15 \pm 2$ $y = 1000 \pm 5$

用绝对误差来刻画近似值的精确程度是有限的，因为它没有反映出它相对于精确值的大小或它占精确值的比例。例如两个量 x^* 和 y^* 与它们的近似值 x 和 y 分别为

$$x^* = 10, x = 10 \pm 1$$

$$y^* = 1000, y = 1000 \pm 3$$

则有误差限 $|\varepsilon_x| = |x - x^*| \le 1, \quad |\varepsilon_y| = |y - y^*| \le 3$

虽然 ε_y 是 ε_x 的**3**倍，但在**1000**内差**3**显然比**10**内差**1**更精确些。这说明一个近似值的精确程度除了与绝对误差有关外，还与精确值的大小有关，所以这是可以用相对误差来比较这两个近似数的准确度。

绝对误差和绝对误差限仅考虑了误差值本身的大小，没有考虑准确值的大小。为了能较好地反映近似值的精确程度，还应考虑准确值的大小。

定义1.3 设x^* 0为准确值，x为x^*的一个近似值。称

$$e_r = \frac{e}{x^*} = \frac{x^* \quad x}{x^*}$$

为近似值x的相对误差。若存在正数 $_r$满足

$$|e_r| = \left|\frac{x^* \quad x}{x^*}\right| = \left|\frac{e}{x^*}\right| \qquad {}_r$$

则称 $_r$为近似值x的一个相对误差限。

相对误差限

绝对误差限

$$\varepsilon_r = \frac{\varepsilon}{|x^*|}$$

往往未知

$$e_r = \frac{e}{x} = \frac{x^* - x}{x}$$

代替相对误差

$$\varepsilon_r^* = \frac{\varepsilon}{|x|}$$

代替相对误差限

为了给出近似数的表示方法，使之既能表示大小又能表示精度，引进有效数字的概念。

一个数的近似数往往是通过四舍五入的原则求得的，例如：

$$\pi = 3.1415926\ldots$$

取以下近似数

$$\pi_1 = 3.14,\quad \pi_2 = 3.142,\quad \pi_3 = 3.14159$$

则分别得到这些近似数的绝对误差

$$|\pi - \pi_1| = 0.001\ldots \le 0.005 = \frac{1}{2}\times 10^{-2}$$

$$|\pi - \pi_2| = 0.0004\ldots \le 0.0005 = \frac{1}{2}\times 10^{-3}$$

$$|\pi - \pi_3| = 0.000002\ldots \le 0.000005 = \frac{1}{2}\times 10^{-5}$$

可以发现每一个近似数的绝对误差限都不超过近似数末尾数的半个单位。如果一个近似数满足这个条件，就把这个近似数从末尾到第一位非零数字之间的所有数字叫做有效数字。

定义1.5 设数 x^* 的近似值可以表示为

$$x = \pm 0.\alpha_1\alpha_2\ldots\alpha_n \times 10^m$$

其中 m 是整数，$\alpha_i (i = 1,2,\cdots,n)$ 是0到9 中的一个数字，而且 $\alpha_1 \neq 0$。如果其绝对误差限为

$$|x - x^*| \le \frac{1}{2} \times 10^{m-n}$$

则称近似数 x 具有 n 位有效数字。

1.3.3、误差的传播

1.误差分析的重要性

$$I_n = \frac{1}{n} - 5I_{n-1} \qquad ①$$

$$I_{n-1} = \frac{1}{5n} - \frac{1}{5}I_n \qquad ②$$

针对①，设初始误差 $e(I_0)$

$$e(I_n) = -5e(I_{n-1}) = \ldots. = (-5)^n e(I_0)$$

针对② ，设初始误差 $e(I_{20})$

$$e(I_0) = -\frac{1}{5}e(I_1) = \ldots. = \frac{1}{(-5)^{20}}e(I_{20})$$

2、数据误差的传播

计算机主要是加、减、乘、除四则运算，经过四则运算后，误差会怎么样变化？

由于精确值与误差相差很小，其差可以认为是较小的增量，即可以把误差看作微分

$$e = x^* - x = dx$$

$$e_r = \frac{x^* - x}{x^*} = \frac{dx}{x} = d\ln x$$

1.3.4、误差的控制

1.简化计算步骤，减少运算次数

$$x^{255}=x.x^{2}.x^{4}.x^{8}.x^{16}.x^{32}.x^{64}.x^{128}$$

原先要做254次乘法现只需14次即可

(2) 作减法时应避免两个相近数相减

两个相近的数相减，会使有效数字的位数严重损失！

例1.2.10 用四位浮点数计算 $\dfrac{1}{759}-\dfrac{1}{760}$

解 $$\frac{1}{759}-\frac{1}{760}=0.1318\times10^{-2}-0.1316\times10^{-2}=0.2\times10^{-5}$$

只有一位有效数字，有效数字大量损失，造成相对误差扩大。

$$\frac{1}{759}-\frac{1}{760}=\frac{1}{759\times760}=\frac{1}{0.5768\times10^{6}}=0.1734\times10^{-5}$$

结果仍然有四位有效数字。

这说明了算法设计的重要性。在算法设计中，若可能出现两个相近数相减，则改变计算公式，如使用三角变换、有理化等等。

(3) 防止大数吃小数

主要由计算机的位数引起

计算机中数的计算特点：

- 加法先对阶，后运算，再舍入。
- 乘法先运算，再舍入。
- 不在计算机数系中的数做四舍五入处理。

例　作一个有效数字为4位的连加运算

$$10^4 \times 0.1234 + 0.4987 + 0.4896 + 0.4697 + 0.4012$$

$$= 10^4 \times 0.1234$$

而如果将小数放在前面计算

$$0.4987 + 0.4896 + 0.4697 + 0.4012 + 10^4 \times 0.1234$$

$$= 10^4 \times 0.1236$$

在作连加时，为防止大数吃小数，应从小到大进行相加，如此，精度将得到适当改善。当然也可采取别的方法。

(3) 避免小数作除数

$$d\left(\frac{x}{y}\right)=\frac{ydx-xdy}{y^2} \Rightarrow e\left(\frac{x}{y}\right)=\frac{ye(x)-xe(y)}{y^2}$$

其中 $e\left(\frac{x}{y}\right)$ $e(y)$ $e(x)$ 表示绝对误差

当|y|很小时，$e\left(\frac{x}{y}\right)$很大

§1.3.5 算法的稳定性

一种数值算法，如果其计算舍入误差积累是可控制的，则称其为数值稳定的，反之称为数值不稳定的。数值不稳定的算法没有实用价值。

不稳定的算法可能导致计算结果不可靠甚至严重失真。因此，在计算时，应该采用稳定的数值计算方法。

例　求解线性方程组

$$\begin{cases} x_1 + \frac{1}{2}x_2 + \frac{1}{3}x_3 = \frac{11}{6} \\ \frac{1}{2}x_1 + \frac{1}{3}x_2 + \frac{1}{4}x_3 = \frac{13}{12} \\ \frac{1}{3}x_1 + \frac{1}{4}x_2 + \frac{1}{5}x_3 = \frac{47}{60} \end{cases}$$

如把方程组的系数舍入成两位有效数字

$$\begin{cases} x_1 + 0.50x_2 + 0.33x_3 = 1.8 \\ 0.50x_1 + 0.33x_2 + 0.25x_3 = 1.1 \\ 0.33x_1 + 0.25x_2 + 0.20x_3 = 0.78 \end{cases}$$

其精确解为　$x_1=x_2=x_3=1.$

它的精确解为$x_1 = -6.222\ldots$

$x_2 = 38.25\ldots$

$x_3 = -33.65\ldots$

若对方程组的系数和中间结果均取3位10进制有效数字，然后用Gauss消元法求解，得到计算解为：

$$x_1 \approx 1.09,\ x_2 \approx 0.488,\ x_3 \approx 0.491.$$

显然，该计算解的精度较差。

同样用Gauss消元法求解方程组：

$$\begin{cases} 2x_1 - x_2 + 3x_3 = 1 \\ 4x_1 + 2x_2 + 5x_3 = 4 \\ 2x_1 + 2x_3 = 6 \end{cases}$$

也取3位10进制有效数字，得到计算解为：

$$x_1 = 9.00,\ x_2 = -1.00,\ x_3 = -6.00.$$

容易验证，它是方程组的精确解。

- 算法优劣的标准

  - 从截断误差观点看，算法必须是截断误差小，收敛速速要快。即运算量小，机器用时少。

  - 从舍入误差观点看，舍入误差在计算过程中要能控制，即算法的数值要稳定。

  - 从实现算法的观点看，算法的逻辑结构不宜太复杂，便于程序编制和上机实现.

- 设计算法时应遵循的原则

  – 要具有数值稳定性，即能控制误差的传播。

  – 避免大数吃小数，即两数相加时，防止较小的数加不到较大的数上。

  – 避免两相近的数相减，以免有效数字的大量丢失。

  – 避免分母很小或乘法因子很大，以免产生溢出。