

# 高等数值算法与应用(1)

- 数值计算基础、Matlab介绍 (chap1) -

喻文健

# Outline

- ▶ 课程简介
- ▶ 数值计算的基础知识
- ▶ Matlab与数值计算软件
- ▶ 浮点算术体系与误差分析



# 课程简介

## ▶ Who

- 喻文健(东主楼8区403室), 对数值计算感兴趣的你们

## ▶ When

- 周一下午1:30-4:00

## ▶ Where

- 六教6B312

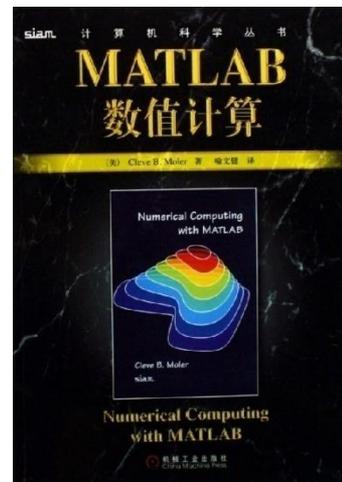
## ▶ What

- 《Matlab数值计算》, 补充内容

## ▶ Why

- 数值计算方法、如何用它们解问题

网络学堂”教学资源”-”课本勘误”



更多信息参考课程网站:

<http://learn.tsinghua.edu.cn:8080/2003990088/anaa.htm>

# 课程简介

## ▶ How

- 课堂讲授、作业(上机解题)、期末考核

## ▶ 如何评分?

- 课程参与: 5%
- 作业(5次): 60%
- 期末大作业/Project: 40%

## ▶ 有问题请及时沟通: [yu-wj@tsinghua.edu.cn](mailto:yu-wj@tsinghua.edu.cn)

## ▶ 教学资源、通知、作业、答疑见“网络学堂”

## ▶ 交互演示网站: <http://learn.tsinghua.edu.cn:8080/2003990088/advNum/index.html>

## ▶ Moler's Blog: <http://blogs.mathworks.com/cleve/>

# 课程简介

## ▶ 教学大纲

- 第一章 MATLAB简介与误差
- 第二章 线性方程组
- 第三章 插值
- 第四章 非线性方程求根与局部极值
- 第五章 最小二乘问题
- 第十章 特征值和奇异值
- 第六章 数值积分
- 第七章 常微分方程
- 第八章 傅立叶分析
- 第九章 随机数
- 第十一章 偏微分方程

重点：1.超出**本科教学范围**的内容  
2.应用实例 (计算思维)  
3.用Matlab编程

## 可能的补充内容：

- 稀疏线性方程组的直接解法
- 解线性方程组的Krylov子空间迭代法

# 数值计算基础知识

## » 概览

“十大算法”

问题、以及求解步骤

误差分析基本概念回顾

# 数值分析、科学计算、数值计算

Numerical analysis => Scientific computing

数值计算是当今科学研究的三种基本手段之一。它随着计算机的普及和各门类科学技术的迅速发展日益受到人们的重视。发达国家普遍比较重视数值计算在科学与工程中的研究和应用，甚至将其作为衡量国家综合实力的一个重要方面。

—— 译者序

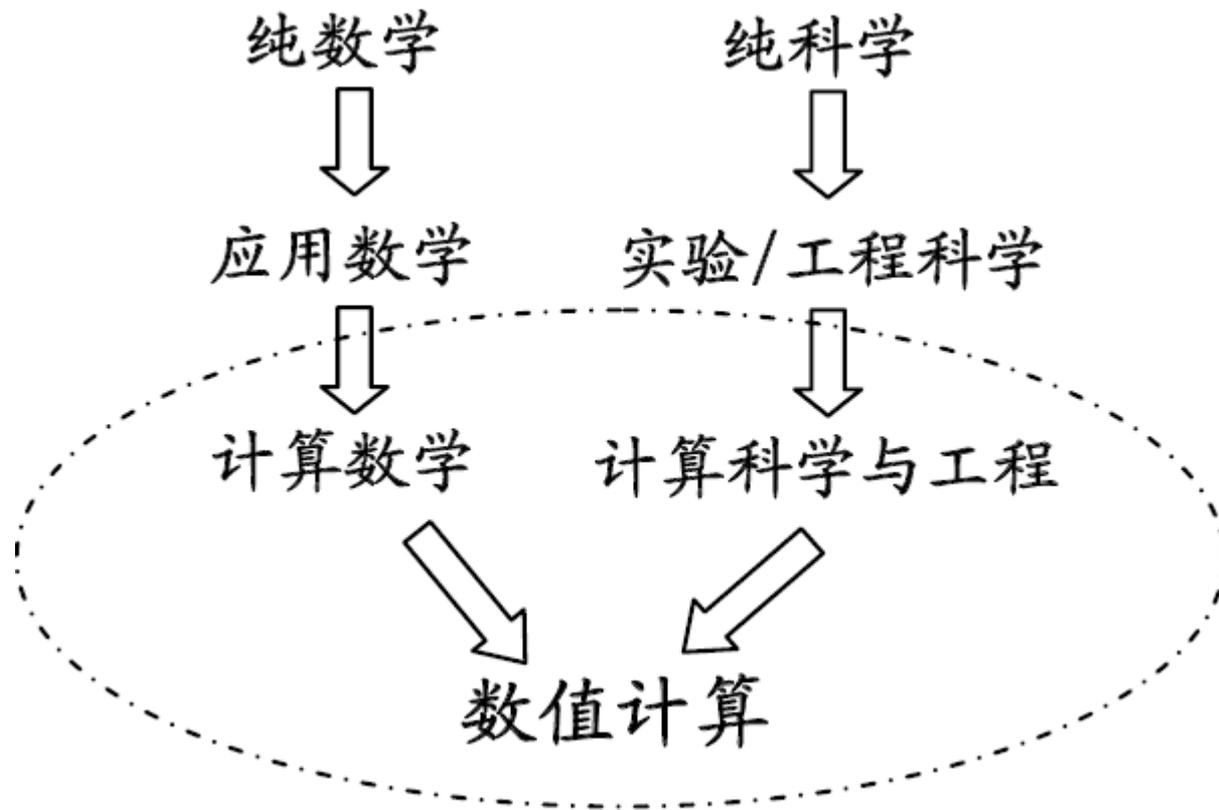
科学计算的含义也可包括软件和硬件两方面，其中软件方面很重要的基础是数值计算的有关算法（数值算法）

对象

核心

数值计算：主要研究求解连续数学问题的算法（其中包含、但不局限于计算误差的研究）

# Evolution of scientific computing from other sciences and engineering disciplines



# 何谓“算法” – Algorithm

## ▶ “Algorithm”

- 源于九世纪阿拉伯学者的名字al-Khwarizmi
- 其著作《al-jabr wa'l muqabalah》是现代高中代数(algebra)课本的雏形
- 他强调解决问题时，系统的过程的重要性

□ *计算机程序 = 数据结构 + 算法*

□ “Computing in Science & Engineering”(IEEE Computer Society期刊)在2000年第1期公布了其评选的20世纪“十大算法”

原始资料见网络学堂“教学资源”



About SIAM

Membership

Journals

Conferences

Books

SIAM News

Cust. Service

Join | Renew | C

Society for Industrial and Applied Mathematics

SIAM

Search

Previous Issue

Calendar

Member News

Federal Agency News

Announcements

Subscriptions



# SIAM News

Volume 33, Number 4, May 2000

[The Best of the 20th Century: Editors Name Top 10 Algorithms \(PDF File\)](#)

[Listening In on a Cryptography Seminar \(PDF File\)](#)

[Parallel Parameter Estimation in Full Motor Vehicle Dynamics \(PDF File\)](#)

[Is It Boole that Makes the World Go Round?](#)

[Progress in Robotics: An Accelerated Rerun of Natural History? \(PDF File\)](#)

DJC

# Top ten algorithms of the century

“We tried to assemble the 10 algorithms with the **greatest influence** on the **development** and **practice** of science and engineering in the 20th century”

— Editors

- ▶ 1. **1946** Los Alamos科学实验室的*J. von Neumann*, *S. Ulam*和*N. Metropolis*提出的Metropolis算法 (Monte Carlo方法中最成功、最有影响的一个)
- ▶ 2. **1947** RAND公司的*G. Dantzig*创造的线性规划的单纯型算法 (simplex method)
- ▶ 3. **1950** 美国UCLA大学与美国国家标准局数值分析所的*M. Hestenes*, *E. Stiefel*和*C. Lanczos*开创的Krylov子空间迭代法 (*Lanczos过程*, *CG算法*)
- ▶ 4. **1950's** 橡树岭(Oak Ridge)国家实验室的*A. Householder*形式化的矩阵分解方法 (*表示成矩阵分解起了革命性作用*)

# Top ten algorithms of the century

- ▶ 5. **1957** IBM的*J. Backus*领导的小组开发的Fortran优化编译器
- ▶ 6. **1959–61** 伦敦Ferranti Ltd.的*J.G.F. Francis*发明的QR算法, *稳定地计算中、小规模矩阵的所有特征值*
- ▶ 7. **1962** 伦敦Elliott Brothers, Ltd.的*Tony Hoare*提出快速排序算法(Quicksort)
- ▶ 8. **1965** IBM Watson研究中心的*J. Cooley*与Princeton大学及AT&T Bell实验室的*J. Tukey*共同提出了的FFT算法
- ▶ 9. **1977** Brigham Young大学的*H. Ferguson*和*R. Forcade*提出的整数关系侦察算法 (*实验数学, 简化量子场理论的计算*)
- ▶ 10. **1987** Yale大学的*L. Greengard*和*V. Rokhlin*发明了快速多极算法(*fast multipole algorithm, 多体问题, 线性复杂度*)

大多数都属于或涉及数值计算的范畴!

# 数值计算的问题与步骤

## ▶ 数值计算（科学计算、计算模拟、数值模拟）

- 没有解析解的问题： $33x^5 + 3x^4 - 17x^3 + 2x^2 + 4x - 39 = 0$
- 虽有解析表达式(解)，但无法(难以)计算： $\sin(x)$
- 模拟通常难以达到的实验条件（时间、金钱成本）
  - 天体物理研究
  - 汽车碰撞实验，芯片投片前的性能仿真

## ■ 数值模拟的步骤

- 建立数学模型（需要相关学科背景）
- 研究数值求解方程的算法
- 通过计算机程序(软件)实现算法
- 在计算机上运行程序(软件)，进行数值实验

课程重点



# 数值计算的问题

## ■ 数值模拟的步骤（续）

- 将计算结果用较直观的方式输出，如图形可视化方法
- 解释和验证计算结果。若需要，重复上面的某些步骤

*上述各步骤相互间紧密地关联，影响着最终的计算结果和效率（问题的实际应用背景也左右着方法的选择）*

## ▶ 问题的适定性（well-defined, well-posed）

- 问题的解连续地依赖于问题数据
- 问题数据的小改变不会导致解的突然(大)变化 反例：地震的物理模型
- 在数值计算中该条件非常重要，因为数据扰动必然存在

## ▶ 充分考虑扰动因素的计算：Stochastic Computation (Uncertainty Quantification)

# 误差

## ▶ 近似造成误差

- 模型误差：建模时的近似
- 数据误差：测量等原因造成误差、前一步计算的结果
- 截断误差（方法误差）：数学上的近似
- 舍入误差：实数用有限位来表示（数值计算中总存在）

计算地球的表面积：▶ 将地球近似成球体

$$A = 4\pi r^2$$

▶ 取半径  $r \approx 6370\text{km}$

▶  $\pi$  的值只能截取到有限位

▶ 计算  $4\pi r^2$  涉及到浮点数的乘法

模型误差

数据误差

数据误差

舍入误差

# 误差

- ▶ (绝对)误差, 相对误差  $e(\hat{x}) = \hat{x} - x$   $e_r(\hat{x}) = \frac{\hat{x} - x}{x}$
- ▶ 数据误差, 计算误差
- ▶ 以计算 $f(x)$ 为例:  $\hat{f}(\hat{x}) - f(x)$
- ▶ 计算误差由**截断误差**与**舍入误差**组成

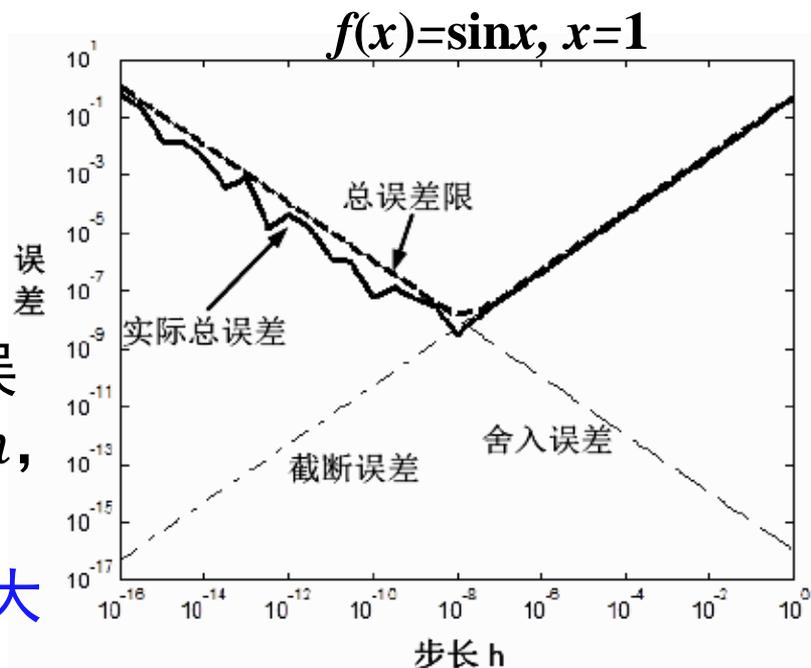
数据误差引起的

- 例: 用差商近似一阶导数

$$f'(x) \approx \frac{f(x+h) - f(x)}{h}$$

- 截断误差估计为 $Mh/2$ ,  $M$ 是 $|f''(\xi)|$ 的上界; 计算 $f$ 函数值的误差主要是舍入误差, 估计为 $2\epsilon/h$ ,  $\epsilon$ 是算一次 $f$ 函数值的误差

随着 $h$ 减小, 总误差**先减小后增大**



# 问题敏感性与算法稳定性

## ▶ 问题的敏感性 (衡量数据传递误差)

- 条件数  $\text{cond} = \frac{\|\text{问题的解的相对变化量}\|}{\|\text{输入数据的相对变化量}\|}$

病态问题

- 例如函数求值问题,  $\text{cond} = \left| \frac{[f(\hat{x}) - f(x)]/f(x)}{(\hat{x} - x)/x} \right|$  (相对条件数)

- 条件数是针对问题而言的, 不涉及求解的具体方法

## ▶ 算法的稳定性

- 反映了“计算过程中”的扰动对计算结果的影响程度, 也称为数值稳定性, 是算法的一个重要属性

- 若计算结果对舍入误差不敏感, 则算法稳定

- 对于含一系列步骤的计算过程, 若计算中的小扰动不被放大(传播)、或放大不严重, 则算法稳定

# Matlab与数值软件

- » 准备知识
  - 黄金分割比
  - 斐波那契数
  - 分形蕨
  - 幻方
  - 数值软件

# MATLAB简介(1)

- ▶ 强大的快速原型编程工具
- ▶ **MAT**rix **LAB**oratory (矩阵实验室), 计算可视化
- ▶ 各种针对不同应用功能的工具箱(Toolbox)
- ▶ 与其他编程语言比较

	Matlab(作为编程语言)	C, C++, Fortran
	第四代编程语言	第三代编程语言
编译方式	解释器, 或JIT加速器(v 6.5后)	编译器
申明变量?	不需要	需要
开发时间	较快	较慢
运行时间	较慢	较快
开发环境	集成环境(编辑器、调试器、命令历史、变量空间、profiler、编译器)	--

# MATLAB简介(2)

其他窗口

The screenshot shows the MATLAB 7.8.0 (R2009a) interface. The Command Window on the left contains the following text:

```
>> 3*4
ans =
    12
>> sin(4)
ans =
-0.7568
fx>>
```

A callout bubble points to the Command Window with the text: **命令窗口，将Matlab作为计算器使用**

The Workspace window on the right shows the following table:

Name	Value	Size	Mir
ans	-0.7568	1x1	-0.7
f	[1;2;3;5;8]	5x1	1

A callout bubble points to the Workspace window with the text: **变量空间**

The Command History window at the bottom right shows the following text:

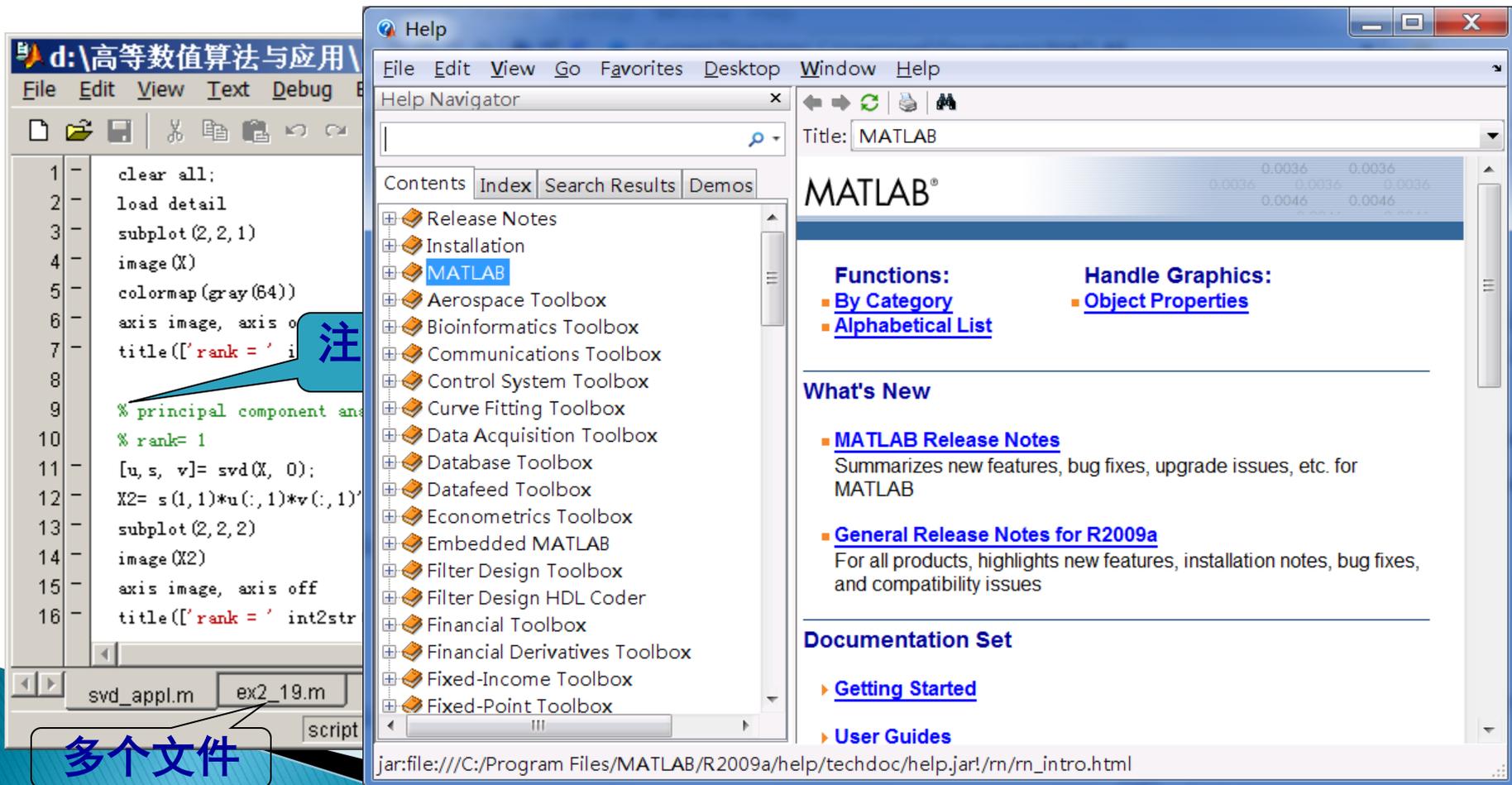
```
%-- 11-8-26 上午10:16 --%
edit fibonacci.m
f=fibonacci(5)
edit
edit fibonacci.m
tic, fibnum(24), toc
tic, fibnum(50), toc
clc
3*4
sin(4)
```

A callout bubble points to the Command History window with the text: **命令历史**

# MATLAB简介(3)

## ▶ 程序编辑窗口

## 帮助窗口



# MATLAB简介(4)

## ▶ 如何“上手”？

- 安装Matlab软件，版本R2009a/b或更新的
- 获取NCM程序包，并设置路径  
(网络学堂-”教学资源”第1项)

```
>> ncmgui
```

- 阅读课本第1章
  - 多动手实验
  - 熟悉“命令历史”键：“↑”，“↓”，以及help, doc命令
- ## ▶ 通过求解几个例子问题熟悉Matlab
- 第1.1 ~ 1.4节

# 黄金分割比(1)

- ▶ 0.6180... , 可能是世界上最有趣的数
- ▶ ?  $\phi = 1 / 0.6180... \approx 1.6180$
- ▶ Matlab功能
  - 表示、显示数的格式
  - 数值、符号求解多项式方程
  - 符号变量转为数值
  - 函数句柄与函数的绘制
  - 脚本程序与绘图功能
  - 一个较复杂的函数程序(截断n项的连分数)

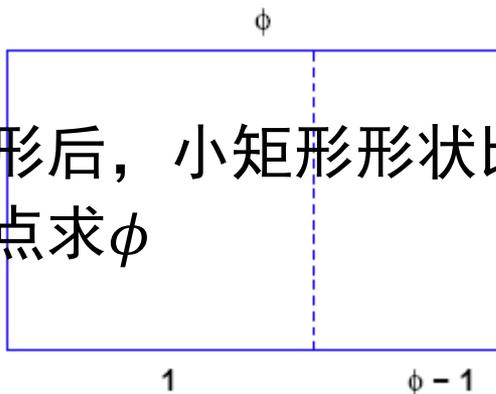
# 黄金分割比(2)

- ▶  $\phi = \frac{1+\sqrt{5}}{2}$
- ▶ Matlab默认采用**双精度浮点数**，但有多种显示格式
- ▶ format命令，或通过File-Preferences菜单设置

```
>> phi = (1+sqrt(5)) / 2  
>> format long
```

## ▶ 黄金矩形

- 裁去正方形后，小矩形形状比例不变
- 根据这一点<sup>1</sup>求 $\phi$



- 解方程:

$$\frac{\phi - 1}{1} = \frac{1}{\phi} \iff \phi^2 - \phi - 1 = 0$$

# 黄金分割比(3)

- ▶ 求解多项式方程  $\phi^2 - \phi - 1 = 0$ 
  - 用降次排列的系数表示多项式，再用 `roots` 命令数值求解

```
>> p = [1 -1 -1];  
>> r = roots(p)
```

- 用 Symbolic Math Toolbox

- 用 `solve` 命令求解方程:  $\frac{\phi - 1}{1} = \frac{1}{\phi}$

- 符号运算时间较长

- 无法求解复杂方程

```
>> r = solve('x-1=1/x')  
>> phi = r(1)
```

- ▶ 符号变量转化为数值

- `vpa`, `double` 命令

- 注意：抑制输出符“;”  
字符串的表示，  
不需申明变量类型

```
>> vpa(phi, 50)  
>> phi = double(phi)
```

# 黄金分割比(4)

## ▶ 函数句柄与函数的绘制

- 有时要将函数作为参数传递，例如积分、绘曲线、解方程
- 匿名函数@ (注: inline命令在新版Matlab中已废除)

- $f(x) = x - 1 - \frac{1}{x}$  ,  $f(\phi) = 0$  `>> f = @(x) x-1-1/x`

- 在[0, 4]区间绘制 $f(x)$ 的曲线 `>> ezplot(f, [0, 4])`

- 使用数值求解非线性方程的命令fzero求 $\phi$

- **注意:** ezplot自动避开奇异点  
plot, fzero命令

- 图形窗口(figure)的编辑、拷贝

```
>> phi = fzero (f, 1)
>> hold on
>> plot (phi, 0, 'o')
```

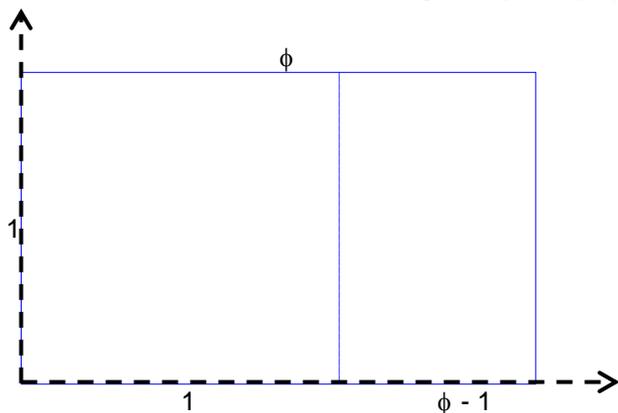
# 黄金分割比(5)

## ▶ 脚本程序与绘图功能

- Matlab程序.m文件有两种: 脚本(script), 函数(function)
- 一个脚本程序实现绘制黄金矩形的功能

程序内容为:

```
>> edit goldrect.m
```



注释、plot, text, axis,  
set命令

```
%GOLDRECT Golden Rectangle  
...  
plot(x,y,'b',u,v,'b--')  
text(phi/2, 1.05, '\phi')  
...  
axis equal           % 遵循真实尺寸  
axis off            % 删除坐标轴  
set(gcf, 'color', 'white')
```

→ get current figure

# 黄金分割比(6)

## ▶ 一个较复杂的函数程序

- $\phi$ 的另一种定义是无限阶连分数 (continued fraction) 的值

$$\phi = 1 + \frac{1}{1 + \frac{1}{1 + \frac{1}{1 + \dots}}}$$

- 函数goldfract.m: 显示截断n项的连分数及其化简分式, 并计算它的值及近似 $\phi$ 的误差

- 连分数化简:  $\phi_1 = 1 + 1/1$ , 设  $\phi_k = \frac{p}{q}$ , 则  $\phi_{k+1} = 1 + \frac{1}{p/q} = \frac{p+q}{p}$

- 注意

- 用“[ ]”构造数组, 也实现字符串的合并

- sprintf命令实现带格式输出 (同C语言)

- eval命令执行字符串中的Matlab命令 (类似于C中的system)

- for循环语句, “:”形成等差数列的值

- 随着项数n增加, goldfract的结果越来越接近 $\phi$ 吗?

# 斐波那契数

## ▶ 斐波那契序列 (Leonardo P. Fibonacci与他的兔子问题)

- $f_1=1, f_2=2, f_n=f_{n-1}+f_{n-2}$

- 第一个函数fibonacci.m, 计算出前n个斐波那契数

- 生成全零向量函数: zeros

```
>> f = zeros (n, 1)
```

- 第二个函数fibnum.m, 输出第n个斐波那契数

- recursive(递推/递归): 递归函数形式简洁、但运行开销大!

- 计时命令: tic, toc

试试fibnum(50)?

```
>> tic, fibnum(24), toc
```

```
>> tic, f=fibonacci(24); f(24), toc
```

- 相邻斐波那契数之比 ~ 截断的连分数 ~ 黄金分割比

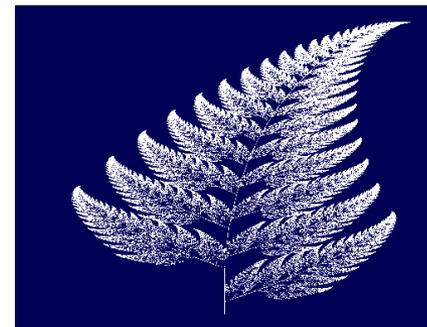
逐项运算符./

```
>> f = fibonacci(40);  
>> f (2:40) ./ f (1:39)
```

- “神奇”的通式

$$f_n = \frac{1}{2\phi - 1} (\phi^{n+1} - (1 - \phi)^{n+1})$$

# 分形蕨



## ▶ 原理

- 平面上精心策划的点集合、无限的随机生成点的过程
- $x \rightarrow Ax + b$ , 有4种变换, 第1种是  $A = \begin{bmatrix} 0.85 & 0.04 \\ -0.04 & 0.85 \end{bmatrix}$ ,  $b = \begin{bmatrix} 0 \\ 1.6 \end{bmatrix}$
- 每次随机地应用4种变换中的一种, 生成新的点

- 无限生长程序: fern
- 含n个点的函数: finitefern

```
>> fern  
>> finitefern(1000, 's')
```

## ▶ Matlab命令

- 程序中含较复杂的绘图命令
- 定义矩阵: ‘;’ while语句
- rand命令生成0~1之间伪随机数
- 用imread, image可看图片

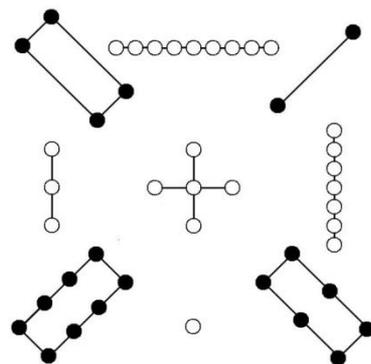
```
>> A1=[.85 .04 ; -.04 .85];  
>> r = rand
```

```
>> F= imread('fern.png');  
>> image(F)
```

# 幻方 (1)

- ▶ 特殊的矩阵~magic命令
  - “洛书”的传说
  - 行, 列, 正 / 反对角线元素之和均相等
- ▶ Matlab命令
  - 验证magic(3)是幻方
  - 这些和等于多少?
  - 三阶幻方矩阵有哪几种?

```
for k = 0:3
    rot90(A, k)      % 旋转矩阵
    rot90(A', k)
end
```

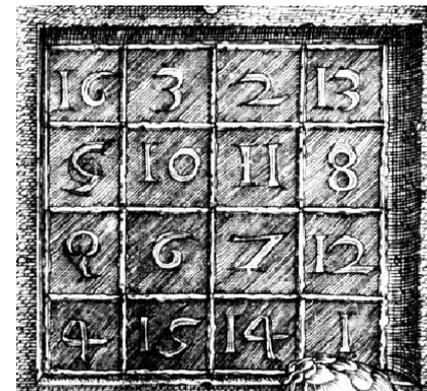


```
>> A = magic(3)
>> sum(A)
>> sum(A')
>> sum(diag(A))
>> sum(diag(flipud(A)))
```

- 矩阵的行列式、逆矩阵

```
>> det(A)
>> X= inv(A)
>> format rat; X
```

# 幻方 (2)



## ▶ 三阶幻方

- 看看矩阵的范数、特征值

```
>> norm(A)
>> eig(A)
```

为何总有15 ?

## ▶ 更高阶幻方

(丢勒)

- Albrecht Durer的版画Melancholia II
- 4阶幻方有880个! (5阶~275305224)
- 矩阵的秩
- 生成幻方的算法

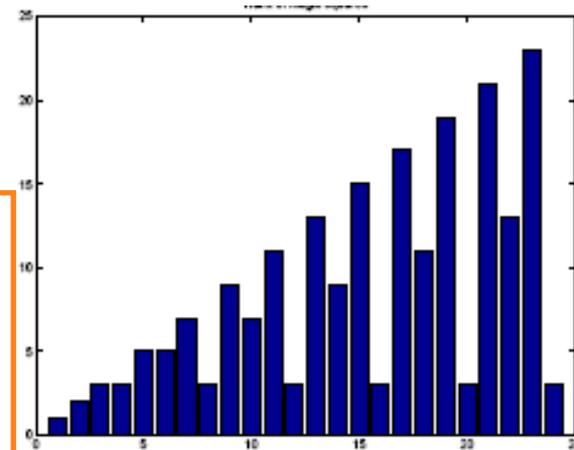
```
>> inv(A)
>> rank(A)
```

```
>> load durer
>> image(X)
>> colormap(map)
>> axis image
```

```
>> edit magic.m
```

- 幻方矩阵的三维表面图

```
surf(magic(n));
axis off;
set(gcf, 'doublebuffer', 'on')
cameratoolbar
```



# 一些常用的matlab命令

- ▶ 常数: pi, i, j
- ▶ 算术运算: +, -, \*, /, ^, .\*, ./, .^
- ▶ 逻辑关系: ==, ~=, >, >=, <, <=
- ▶ 逻辑运算: &, |, ~
- ▶ 帮助: help, doc, lookfor, demo
- ▶ 输出: disp, fprintf, ; (抑制输出), format
- ▶ 初等函数: sin, cos, tan, sinh, asin, exp, log, log10, sqrt
- ▶ 变量记录: who, whos, clear, save, load, ans, diary
- ▶ 向量: [..., ...], rand, length
- ▶ 矩阵: [..., ...; ..., ...], ones, eys, rand, size, diag, tril, triu
- ▶ 绘图: plot, subplot, loglog, ezplot, hold, plot3, figure, close
- ▶ 文件: edit, type, ls, path
- ▶ 编程: function, if, for, while, end, @
- ▶ 退出: quit, exit

# 数值软件 / 程序包

- 数值计算的软件与程序包
  - 解决一些共同问题，促进各个科研领域的工作
  - 了解基本原理，学习算法设计和实现技巧
  - 成为聪明的软件 / 程序包使用者
- 存在形式和资源
  - 互联网，免费 / 商业代码
  - Fortran, C, C++, Matlab
  - 源代码使用，或API调用
  - 交互式集成环境的软件

名称	内容说明	商业/免费	网址
<b>CMLIB</b>	美国国家标准技术协会(NIST)的数学与统计软件虚拟仓库和检索系统	免费	<a href="http://gams.nist.gov">gams.nist.gov</a>
FMM	Computer Methods for Mathematical Computations 一书[2]所附软件	免费	<a href="http://www.netlib.org">www.netlib.org</a>
HSL	英国 Science and Technology Facilities Council 提供的科学计算程序包	免费	<a href="http://www.hsl.rl.ac.uk">www.hsl.rl.ac.uk</a>
IMSL	Visual Numerics 公司提供的数学与统计程序软件库	商业	<a href="http://www.vni.com">www.vni.com</a>
NAG	NAG 公司提供的数值算法程序集	商业	<a href="http://www.nag.com">www.nag.com</a>
NAPACK	Applied Numerical Linear Algebra 一书[3]所附软件	免费	<a href="http://www.netlib.org">www.netlib.org</a>
<b>NETLIB</b>	汇集各种免费数值计算软件的网站	免费	<a href="http://www.netlib.org">www.netlib.org</a>
NR	Numerical Recipes 系列书[4]所附软件	部分免费	<a href="http://www.nr.com">www.nr.com</a>
NUMERALGO	期刊 Numerical Algorithms 中的软件	免费	<a href="http://www.netlib.org">www.netlib.org</a>
MATLAB	MathWorks 公司出品的著名数学软件	商业	<a href="http://www.matlab.com">www.matlab.com</a>
PORT	贝尔实验室开发的软件	部分免费	<a href="http://www.netlib.org">www.netlib.org</a>
SLATEC	美国政府实验室联合汇编的软件	免费	<a href="http://www.netlib.org">www.netlib.org</a>
SOL	美国斯坦福大学系统优化实验室开发的优化及相关问题的软件	免费	<a href="http://www.stanford.edu/group/SOL/">www.stanford.edu/group/SOL/</a>
TOMS	期刊 ACM Transactions on Mathematical Software 中的算法程序, 用名字和数字编号两种方式标识	免费	<a href="http://www.netlib.org">www.netlib.org</a>

# 浮点算术体系与误差

- » 计算机中的浮点数
- 浮点数与舍入误差
- 数值计算中如何减小误差

# 计算机中的浮点数

- ▶ 用机器中的浮点数 $x$ 表示实数 (2进制)

$$x = \pm \left( d_0 + \frac{d_1}{2} + \frac{d_2}{2^2} + \dots + \frac{d_p}{2^p} \right) \cdot 2^E = \pm(1 + f) \cdot 2^E$$

- ▶ **基数**:2; **指数** $E$ 的范围: 下限值 $L$ , 上限值 $U$ ; **尾数** $f$ 位数 $p$
- ▶ IEEE浮点数系统已成为标准, 分**单精度**和**双精度**
- ▶ Matlab默认用双精度, 但也支持单精度  $\rightarrow [-1022, 1023]$
- ▶ 64位中, **1**位存正负号, **11**位存 $E$ ,  $p=52$ 位存 $f$  ( $d_1 d_2 \dots d_{52}$ )

$$x = [b_1 b_2 b_3 \dots b_{12} b_{13} \dots b_{64}] \quad (d_0 = 1, \text{不用存储})$$

$\underbrace{\hspace{10em}}_{E+1023}$

```
realmin=2-1022  
realmax=(2-eps)21023
```

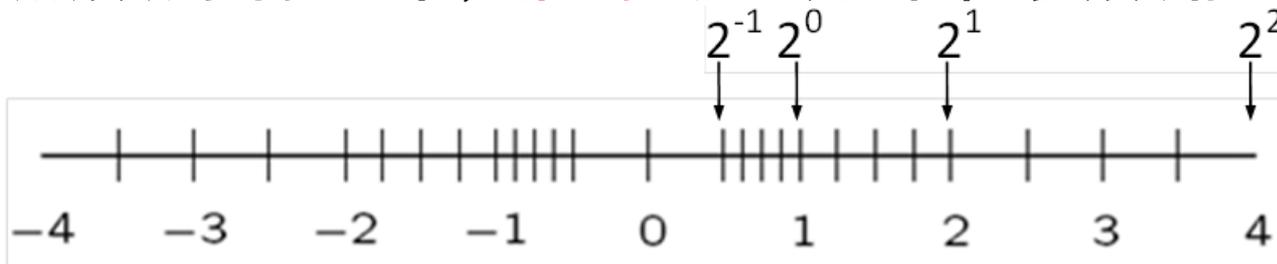
```
>> realmin  
ans =  
2.2251e-308
```

```
>> realmax  
ans =  
1.7977e+308
```

# 计算机中的浮点数

floatgui.m

- ▶ 浮点数为有限个, **非均匀**地分布在实数轴上



例: 一个简单浮点数系统,  $p=2, L=-1, U=1$

- ▶ **机器精度**  $\epsilon_{\text{mach}}$ : 1与下一个浮点数的间隔的**一半**

$$\epsilon_{\text{mach}} = \frac{1}{2} \cdot \frac{1}{2^p} = 2^{-(p+1)} \quad \text{Matlab中, } \text{eps} = 2\epsilon_{\text{mach}}$$

浮点数系统	$p$	$L$	$U$	$\epsilon_{\text{mach}}$
IEEE单精度	23	-126	127	$5.960 \times 10^{-8}$
IEEE双精度	52	-1022	1023	$1.110 \times 10^{-16}$

# 浮点数与舍入误差

## 数的表示造成误差

- **定理1**: 用浮点数  $fl(x)$  近似表示实数  $x$ , 其相对误差

$$\left| \frac{fl(x) - x}{x} \right| \leq \varepsilon_{\text{mach}} = \text{eps} / 2 \approx 1.110 \times 10^{-16}$$

- 例:  $t = 0.1$

```
>> format hex; t
```

$$\frac{1}{10} = 2^{-4} \cdot \left( 1 + \frac{9}{16} + \frac{9}{16^2} + \frac{9}{16^3} + \frac{9}{16^4} + \dots \right), \quad t_1 < 1/10 < t_2$$

$$t_1 = 2^{-4} \cdot \left( 1 + \frac{9}{16} + \frac{9}{16^2} + \frac{9}{16^3} + \dots + \frac{9}{16^{12}} + \frac{9}{16^{13}} \right), \quad t_2 = 2^{-4} \cdot \left( 1 + \frac{9}{16} + \frac{9}{16^2} + \frac{9}{16^3} + \dots + \frac{9}{16^{12}} + \frac{10}{16^{13}} \right)$$

- 十个0.1相加  $\neq 1$  (3ff0000000000000)

- $0.3 / 0.1$  成为 4007ffffffffff

$E+1023$  尾数f

- 评估机器精度的程序  $\rightarrow$

```
>> a = 4/3  
>> b = a - 1  
>> c = 3*b  
>> e = 1 - c
```

# “大数吃掉小数”与抵消

- ▶ **定理2**:  $x_1, x_2 \in \mathbb{R}$ , 若  $\left| \frac{x_2}{x_1} \right| \leq \frac{1}{2} \varepsilon_{\text{mach}}$ , 则  $x_2$  的值对浮点运算  $x_1 \pm x_2$  的结果毫无影响
  - “大数吃掉小数”现象
- ▶ 两个符号相同、值相近的  $p$  位数相减可能使结果的有效数字远少于  $p$  位. 称之为**抵消**(cancellation)
  - **例**:  $x = 1.92305 \times 10^3, y = 1.92137 \times 10^3$ , 则  $x - y = 1.68$
  - 未发生**舍入**, 单独看这次计算无误差
  - 结果的有效数字**位数减少**, 意味着**相对**误差的放大, 进而给后续计算带来较大误差
  - 抵消现象是发生**信息丢失**、**误差变大**的信号!

# 抵消现象

- ▶ 一元二次方程求根公式的例子

$$ax^2 + bx + c = 0$$

解为：

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

若 $|4ac| \ll b^2$ ，出现抵消现象。 $b > 0$ ，则计算 $x_1$ 时误差大  
解决办法：

$$x_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a} \cdot \frac{-b - \sqrt{b^2 - 4ac}}{-b - \sqrt{b^2 - 4ac}} = \frac{2c}{-b - \sqrt{b^2 - 4ac}}$$

计算 $x_2$ 时，可能的问题可类似地解决

# 舍入误差与Matlab

(默认是IEEE双精度浮点数)

$\sim 9.0 \times 10^{15}$

- ▶ Matlab用浮点数表示整数 尾数有52位, 保证 $1 \sim 2^{53} - 1$ 可精确表示
  - 浮点整数, 只要“结果不太长”, 浮点整数的运算没有误差
  - 加、减、乘, 若除法、平方根的结果也是整数
  - 命令log2, pow2能实现分解、构造浮点数的功能
- ▶ 由于舍入误差, 奇异矩阵在计算时可能并不奇异
  - 见课本pp.33~34
- ▶ 一个特殊的7次多项式

```
>> [F, E] = log2(11)
F =
    0.6875
E =
    4
```

```
>> x = 0.988: .0001: 1.012;
>> y = x.^7 - 7*x.^6 + 21*x.^5 - 35*x.^4 + 35*x.^3 - 21*x.^2 + 7*x - 1;
>> plot(x, y)
```

发生了严重的抵消现象

$(x-1)^7$

# 数值计算中如何减小误差

	总误差		
	计算误差		数据传递误差
	截断误差	舍入误差	
如何评估大小?	根据不同问题和方法进行讨论	向后误差分析; 区间分析法; 很难定量分析	问题敏感性(条件数); 直接近似分析
如何减小误差?	计算方法的 选择	选稳定的算法; 减小舍入误差的建议; 采用更高精度浮点数	变换问题形式, 改善敏感性

**参考文献:** 喻文健, 《数值分析与算法》, 清华大学出版社, 2012年1月

