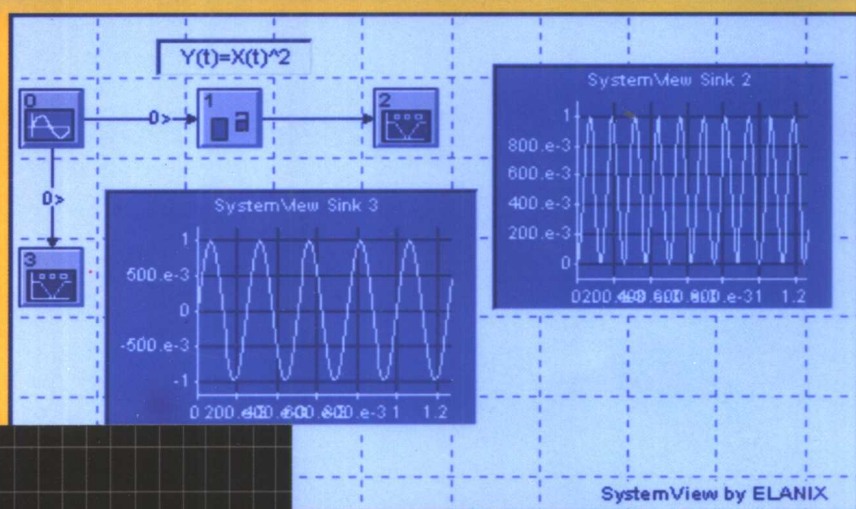


现代通信

实验系统的计算机仿真

XIANDAI TONGXIN

SHIYAN XITONG DE JISUANJI FANGZHEN



陈 萍 等 编著

国防工业出版社

现代通信实验系统的 计算机仿真

陈萍 韩玉芬 谢文苗 徐春秀 编著
刘文京 张瑞芹 杨鸿文

国防工业出版社

·北京·

图书在版编目(CIP)数据

现代通信实验系统的计算机仿真/陈萍等编著. —北京:国防工业出版社, 2003.4

ISBN 7-118-03064-3

I. 现... II. 陈... III. 通信系统—计算机仿真
IV. TN914

中国版本图书馆 CIP 数据核字(2003)第 009237 号

国防工业出版社 出版发行

(北京市海淀区紫竹院南路 23 号)

(邮政编码 100044)

新艺印刷厂印刷

新华书店经售

*

开本 787×1092 1/16 印张 16½ 378 千字

2003 年 4 月第 1 版 2003 年 4 月北京第 1 次印刷

印数:1—4000 册 定价:23.00 元

(本书如有印装错误,我社负责调换)

前 言

目前,我国的教育改革面临着四大转变,即从应试教育到素质教育的转变,从注重书本知识到注重实验能力的转变,从注重知识接受能力到注重创新能力的转变,从培养个人奋斗到培养团队工作精神的转变。而实验教学则是目前教育改革中最重要的突破口,它能够培养学生的理论联系实际的能力、综合设计能力和创新意识。根据我们实验教学的经验,我们认为设计性实验比验证性实验更能激发学生的学习兴趣,更能使学生得到锻炼。

通信实验系统的计算机仿真在现代通信的教学中具有十分重要的意义,它不仅能够提高学生对所学理论知识的理解能力,更重要的是能够提高和挖掘学生对所学知识的实际运用能力即创新意识,为学生将来进入社会从事相关工作奠定较好的“能力”基础。

本书的基本内容是从教学实践和科研课题中提炼出来的,初稿曾在北京邮电大学电信工程学院用作通信原理实验讲义和课程设计指导书。为了满足更多读者的需要,我们将原有实验教材作了全面系统的整理,编写了本书。本教材力图体现三新:内容新,手段新,体系新。在内容上我们在原有实验的基础上扩充了 CDMA、纠错编码等实验内容;在实验手段上,我们引入了目前国际上流行的仿真软件,实现了实验系统设计的软硬件结合;在实验体系上,我们注重引入“理工融合、理势工发”的教学理念,旨在高校实验教改中做一些探索性的工作。

全书共分为 3 部分。

第 1 部分共 4 章,第 1、2 章介绍 MATLAB 软件和基本使用,第 3 章介绍仿真思想和数字基带传输系统的计算机仿真,第 4 章讲述脉冲编码数字通信系统的计算机仿真。

第 2 部分共 2 章,第 5 章内容包括 SystemView 软件介绍,第 6 章以实例的方式详细介绍了 SystemView 软件在通信仿真系统中的应用。

第 3 部分共 7 章,第 7 章内容包括 MAX + plus II 和硬件实验环境的介绍,第 8 章讲述 PCM 14/16 路数字终端机定时系统的硬件实现及计算机仿真,第 9 章为串行口的数据传输实验,第 10 章讲述用 FPGA 实现 CDMA 数字基带收发实验系统的思想和方法,第 11

章叙述用 CPLD 实现 CDMA 扩频通信中的同步实验系统的思想和方法,第 12 章叙述用 CPLD 仿真通信信道中的纠错编码实验系统,第 13 章是关于“GMSK 调制器”基带硬件实验系统的介绍。

本书适用对象是通信与电子工程类高年级本科生,假定读者已具有电路、信号与系统以及通信原理方面的基础知识。

本书由陈萍担任主编,第 1 部分由陈萍、杨鸿文编写;第 2 部分由谢文苗编写;第 3 部分的第 7、8 章由韩玉芬编写,第 13 章由徐春秀编写,第 9 章由刘文京编写,第 10、11 章由陈萍、张瑞芹编写,第 12 章由陈萍编写。由于作者水平有限,书中难免存在一些缺点和错误,殷切希望广大读者批评指正。

在本书出版之际,我们要特别感谢北京邮电大学的林家儒教授、庞沁华教授对本书的贡献!感谢他们为本书编写提供的宝贵素材和建设性意见。我们还要感谢做创新实验和毕业设计的同学们,许多高效的程序就出自他们。最后我们要感谢国防工业出版社的责任编辑和其他工作人员为书稿的完成所作的努力。

作 者

2003 年 1 月

于北京邮电大学

内 容 简 介

本书以数字通信原理为基础,用大量实例系统介绍了现代通信系统仿真的手段和方法。

全书共分为三大部分,第1部分介绍 MATLAB 软件及在数字基带传输系统和脉冲编码数字通信系统中的仿真应用。第2部分介绍 System View 软件及在通信仿真系统中的应用。第3部分介绍实验软硬件的实验环境以及通信系统中较典型的几个系统实验。

本书可作为高等学校工科无线电技术、通信与电子系统等专业的本科生实验教材,也可作为科研工作人员的参考书。

目 录

第 1 部分 基于 MATLAB 的数字通信系统的仿真

第 1 章 引言	1
第 2 章 MATLAB 初步	3
2.1 命令的执行与显示	3
2.2 MATLAB 基本命令	4
2.3 矩阵、矢量和标量	6
2.4 基本运算	7
2.5 函数	9
2.6 MATLAB 编程	16
2.7 循环与分支	18
2.8 输入语句	21
2.9 其他基本操作	21
第 3 章 数字基带传输系统的计算机仿真	22
3.1 信号及系统在计算机中的表示	22
3.2 随机信号的产生	24
3.3 眼图的仿真	27
3.4 误码率的仿真	28
3.5 范例	28
3.6 上机实验要求	29
附录	30
第 4 章 脉冲编码调制系统的计算机仿真	42
4.1 实验设置的意义	42
4.2 实验目的	42
4.3 线性编译码的计算机仿真	42
4.4 对数编译码的计算机仿真的实现	47
4.5 线性 PCM 与对数 PCM 的性能比较	53
4.6 实验内容	58
附录	59

第 2 部分 SystemView 在通信仿真系统中的应用

第 5 章 SystemView 简介	62
5.1 SystemView 简介	62

5.2	SystemView 的用户环境	63
第 6 章	实例详解	84
6.1	信号的产生实例	84
6.2	数字调制与解调实例	90
第 3 部分 实验系统		
第 7 章	实验系统软硬件开发环境	110
7.1	MAX + plusII 软件	110
7.2	实验系统硬件环境说明	138
第 8 章	PCM 14/16 路数字终端机定时系统的硬件实现及计算机仿真	148
8.1	实验设置的意义和目的	148
8.2	帧结构及主时钟频率的确定	148
8.3	原理、设计及硬件实现	149
8.4	计算机软件仿真与下载	163
8.5	实验要求	165
第 9 章	串行口数据传输实验	166
9.1	实验目的	166
9.2	数字系统的设计方法	166
9.3	实验内容及设计任务	167
9.4	设计示例	168
9.5	实验内容及要求	173
附录		174
第 10 章	用 FPGA 实现 CDMA 数字基带收发实验系统	176
10.1	实验设置的意义及目的	176
10.2	CDMA 基本原理及核心技术	176
10.3	CDMA 数字基带收发实验系统的构成	181
10.4	实验内容及要求	186
附录 A		186
附录 B		190
附录 C		206
第 11 章	用 CPLD 实现 CDMA 扩频通信中的同步实验系统	214
11.1	实验设置的意义和目的	214
11.2	基本理论及系统框图	214
11.3	同步系统在 MAX + plus II 中的实现	218
11.4	MAX + plus II 对系统的仿真结果及分析	226
11.5	实验要求	227
附录		228
第 12 章	用 CPLD 仿真通信信道中的纠错编码实验系统	229
12.1	实验设置的意义和目的	229

12.2	用循环码实现纠错编码实验系统·····	229
12.3	用卷积码实现纠错编码实验系统·····	235
12.4	循环码与卷积码纠错性能比较·····	238
12.5	实验内容及要求·····	240
附录	·····	240
第 13 章	“GMSK 调制器”基带硬件实验系统 ·····	243
13.1	实验设置的意义和目的·····	243
13.2	实验目的·····	243
13.3	实验内容·····	243
13.4	实验原理·····	244
13.5	实验步骤·····	247

第 1 部分 基于 MATLAB 的 数字通信系统的仿真

第 1 章 引 言

通信原理课的主要内容是论述怎样可靠而有效地实现信息的传输。为此目的, 该课从理论上提出了许多可供选择的传输方法。要使这些传输方法成为现实, 就需要制作出相应的发送设备及接收设备。然后在发送端, 我们把欲传送的信息转换成某种适宜的信号并将之馈入传输媒体(电缆、光缆、无线电波等)。在接收端, 信号又从媒体馈入接收设备, 我们再以同发送端相反的过程恢复原来所发送的信息。根据通信原理课所学的知识, 可知道在什么样的情况下应该选择什么样的传输方式, 并能判断出噪声、信道、传输方式等因素将会怎样影响对我们来说非常重要的一些通信指标, 如信噪比、误码率、发送频谱等等。

要想充分地了解某种传输方式的特性, 最理想的办法莫过于先把它做出来, 再对其进行实际测量。不过实际制作一套通信系统所需的周期长、代价高, 有时要改变系统的某一两个参数就可能意味着整个系统需要重做。故此对于研究设计人员来说, 通过把所有的设计方案、研究问题等都做成硬件, 再对硬件进行测量从而获得系统性能的研究方法应属于不得已而为之。较好的研究方法应该是在硬件制作以前就先设法对系统特性做尽可能充分的了解, 然后再去做硬件。

某种传输方法, 当我们把它做成实际系统后, 它理应符合通信原理的理论对它的描述。如果用通信仪表对实际系统进行实际测量的话, 实测结果应该同理论分析给出一致的结果。如果这两个结果不一致, 一般只可能是这样两种情况: 一是理论中所假设的某些模型并不能完全正确地反映实际情况, 比如通信原理课中经常假设信道是一个理想滤波器, 而实际信道则有可能是一个多径信道; 二是硬件制作可能存在实现上的误差, 也就是说制作的某一个或某几个部件的实际性能与对它期望的性能之间有一些差距。换句话说, 如果在理论阶段所依赖的模型是充分可信的, 如果将要制作的硬件确实能够按我们的要求去工作, 那么就可以在制作硬件之前, 先对将要制作的实际传输系统做充分的研究, 并据此对系统设计是否合理做出评判, 如果设计是合理的我们就可以放心地去实现系统, 不然就需要修改现有设计再做分析。

然而在通信系统中, 除非个别极简单的情况外, 一般很难用推公式这种解析的办法得到系统的各项性能指标。比如无码间干扰时, 双极性最佳基带系统的误码率公式是

$P_e = \frac{1}{2} \operatorname{erfc} \left(\sqrt{\frac{E_b}{n_0}} \right)$, 但如果信道中确实存在码间干扰, 那么对应的误码率公式基本上是不可能得到的。不过这并不意味着就没有别的办法, 在计算机技术十分发达的今天, 通信

中的许多问题都可以通过计算机仿真的办法来进行研究。

计算机仿真实质上就是把硬件实验搬进了计算机, 可以把它看成是一种软件实验。在硬件实验系统中, 用各种电子元器件制作出通信原理中的理论模型所规定的各个模块, 再把它们通过导线或电缆等接在一起, 然后再用示波器、频谱仪、误码仪等通信仪表做各种测量, 最后分析测量结果。在软件实验中我们也是这样做, 只不过所有通信模块及通信仪表的功能都是用程序来实现的, 通信系统的全过程在计算机中仿真运行。

虽然软件实验不像硬件实验那样让人感到“真实”, 但对于许多通信问题的研究来说的确非常有效。与硬件实验相比, 软件实验具有如下一些优点:

(1) 软件实验具有广泛的适应性和极好的灵活性。在硬件实验中改变系统参数也许意味着要重做硬件, 而在软件实验中则只是改一两个数据, 甚至只是在屏幕上按几下鼠标。

(2) 软件实验更有助于我们较为全面地研究通信系统。有许多问题, 通过硬件实验来研究可能非常困难, 但在软件实验中却易于解决。

(3) 硬件实验的精确度取决于元器件及工艺水平, 软件实现的精度取决于 CPU 的运算速度或者说是程序的运算量。

(4) 软件实验建设开发周期短, 成本低。

在本部分实验中我们使用的软件工具是 MATLAB, 期望本部分实验能带给你如下收获:

(1) 学会 MATLAB 软件的最基本运用。

MATLAB 是一种很实用的数学软件, 它易学易用。MATLAB 对于许多的通信仿真类问题是很合适的。

(2) 了解计算机仿真的基本原理及方法, 知道怎样通过仿真的方法去研究通信问题。

(3) 加深对通信原理课相关内容的理解。

通信原理是通信工程专业的一门重要专业基础课。从教学实践中我们注意到, 此课的某些内容对初学者来说往往不大好把握。如果您能自己动手, 通过仿真的办法对这些问题进行一些研究和观察, 自然会有助于解除您的困惑。

第2章 MATLAB 初步

2.1 命令的执行与显示

在 Windows 中,用鼠标双击 MATLAB 图标即可启动 MATLAB for Windows。MATLAB 的命令提示符为“?”。MATLAB 的命令也称语句或函数,在本文中名词“命令”“语句”及“函数”在许多情况下并没有多大差别。

在命令行中,键入命令并按回车(Enter)就可以执行命令。执行结果或其他与命令执行相关的信息将随后显示在屏幕上。例如

```
?why
```

```
Pete wanted it that way.
```

这里,“?”是命令行提示符,“why”是命令,“Pete wanted it that way.”是执行结果。在 MATLAB 中,命令 why 的功能是向屏幕输出一个随机字符串。

又比如

```
?y=log(0)
```

```
Warning: Log of zero
```

```
y =
```

```
-Inf
```

语句 $y=\log(0)$ 的意思是对 0 求自然对数,对数函数 $\log$ 的返回值被赋给变量 y 。MATLAB 在运行时除给出了运算结果为负无穷大 ($-\text{Inf}$) 外,还会显示出对零求对数的告警。

在命令的末尾加上分号“;”,表示不显示函数的返回值或语句的运算结果。例如

```
?y=log(0);
```

```
Warning: Log of zero
```

只显示告警信息而不显示运行结果。此时 y 仍然被赋值为 $-\text{Inf}$ 。

一行中可以有多条命令,命令与命令之间应用逗号“,”或分号“;”隔开。以逗号结尾的命令显示结果,以分号结尾的命令不显示结果。例如

```
?y=log(0);b=1/0,a=atan(b)*180/pi;a
```

```
Warning: Log of zero
```

```
Warning: Divide by zero
```

```
b =
```

```
Inf
```

```
a =
```

```
90
```

这里,“ $y=\log(0);$ ”只显示告警信息而不显示运行结果。“ $b=1/0,$ ”既显示告警信息(被

零除: Divide by zero), 也显示运行结果 (无穷大: Inf)。语句 “a=atan(b)*180/pi;” 被运行但结果未显示。在 MATLAB 中 atan 是反正切函数, pi 是圆周率。语句 “a” 的作用是显示 a 的值: $a=90^\circ$ 。

如果语句有运算结果但此结果未被赋值, 那么 MATLAB 自动将此结果赋值给变量 “ans”, 它是英语单词 answer 的前 3 个字母。例如:

```
?0.5*erfc(sqrt(4.77))
ans =
    0.0010
```

此处, MATLAB 把 $\frac{1}{2}\text{erfc}(\sqrt{4.77})$ 的运算结果赋值给 ans。erfc 是互补误差函数, sqrt 是求平方根。

有些命令除了在命令行状态下有显示外, 还会激活其它图形窗口。如命令 intro、demo 等。intro 是 MATLAB 简介, demo 是 MATLAB 的演示程序。这两个命令都是用 MATLAB 编写而成的。

在 MATLAB 中标识符的大小写是有区别的。例如:

```
?a=6
a =
    6
?A
??? Undefined function or variable 'A'.
```

2.2 MATLAB 基本命令

2.2.1 帮助命令 help

help 命令是一个非常有用的帮助命令, 它用于帮助用户了解 MATLAB 命令与函数的用法或者其他 MATLAB 用到的一些内容, 例如: 运算符等。如果你忘记了一个命令的用法, 只要敲入 “help 命令”, MATLAB 就会显示有关这个命令的详细信息。MATLAB 分大小写, 但有一些命令不分大小写, help 就是其中的一个。

2.2.2 显示目录内容命令 what

what 用于列出目录里的 M, MAT 和 MEX 文件, 其中 M 文件就是 MATLAB 程序文件。例如输入: what work, 那么 MATLAB 就会显示 work 目录下的所有 M 文件。如果用 DOS 里的 dir 或其他类似的命令列出 work 目录的内容, 这时目录下会列出以 m 为后缀的文件。what 命令区分大小写。

2.2.3 寻找函数命令 which

which 命令显示函数所在文件的位置, 给出文件位置的路径。如果是内部函数则显示此函数是内部的。例如:

```
?which max
```

```
max is a built-in function.
```

表明函数max是一个内置的函数。而

```
?which mean
```

```
\matlab\toolbox\matlab\datafun\mean.m
```

表明函数mean是由M文件mean.m定义的，以及该文件存放路径的位置。

2.2.4 路径显示命令 path

path 显示 MATLAB 的搜寻路径。MATLAB 寻找命令或 M 文件时，先查找当前目录，如果找不到，则按 path 设定的搜寻路径去找。

P=path 返回一个字符串 P，P 的内容是搜寻路径。

path(P)把字符串 P 所描述的路径作为当前的搜寻路径。path(path,P)把 P 追加到当前搜寻路径中，P 在最后。path(P,path)把 P 加入到当前搜寻路径中，P 在最前。

2.2.5 显示变量命令 who、whos 与清除变量命令 clear

who 与 whos 命令都用于显示当前变量，但是 whos 给出的信息更加详细。

MATLAB 自动记住用过的变量，但是，如果变量太多，就会占用过多的内存，运行速度就会下降，这时就可以用 clear 命令把一些不要的变量从内存中清除出去。如果要清除所有的变量，那么只需输入 clear 即可，此时在用 whos 命令查看变量的话，将不会得到任何消息。如果只想清除部分变量，那么需在 clear 后加上要清除的变量。例如：

```
? who
```

```
Your variables are:
```

```
a          c          cc         d2
ans        c1         d          dd
b          c2         d1         e
```

```
clear a b    %清除变量a、b
```

```
clear        %清除所有变量
```

2.2.6 改变目录命令 cd 与显示目录命令 dir

cd 命令用于改变当前的目录，dir 命令用来显示目录里的文件。它们与 DOS 或 Unix 里的 cd 命令和 dir 命令是类似的。

2.2.7 窗口编辑命令

clc: 清除屏幕上的内容，但并不清除变量。

home: 移动光标到左上角。

format: 设置输出格式，例如小数点位数等。

echo: 开或关显示命令。

more: 控制多页显示。

2.3 矩阵、矢量和标量

2.3.1 矩阵

MATLAB 的基本运算对象是矩阵。方括弧“[]”用来定义矩阵。逗号“,”或空格用来分隔一行中的不同元素,分号“;”用来分隔不同的行。例如

```
?a=[1,2,3;4,5,6;7,8,9]
```

```
a =
```

```
    1    2    3
    4    5    6
    7    8    9
```

是一个 3×3 的矩阵。

2.3.2 矢量

只有一行或一列的矩阵就是矢量,我们分别称其为行矢量及列矢量。例如

```
?b=[1,4,8,2] %b是一个行矢量。
```

```
b =1      4      8      2
```

```
?c=[1;4;8;2] % c是一个列矢量。
```

```
c =
```

```
1
4
8
2
```

考虑到通信实验内容的具体需要,今后我们主要只考虑行矢量。

2.3.3 计算矢量长度命令 length

length 用于计算矢量的长度,如果 x 是一个常数,则 $\text{length}(x)$ 返回 1; 如果 x 是一个矢量,则 $\text{length}(x)$ 返回矢量的长度; 如果 x 是一个 $M \times N$ 的矩阵, $\text{length}(x)$ 返回 M 、 N 中较大的一个值。例如

```
?length(c)
```

```
ans =
```

```
4
```

2.3.4 标量

只有一个元素的矢量或矩阵就是标量。例如 $d=[1]$ 或 $d=1$ 的意思是把 d 定义为一个值为 1 的标量。

2.3.5 数据类型

1) 实数

MATLAB 不需要对实数作类型说明,所有的数都按双精度浮点方式保存。对很大及

很小的数可采用科学记数法：例如 $1.3\text{e-}18$ 表示 1.3×10^{-18} 。

2) 字符串

MATLAB 也支持字符串变量。字符串由一对单引号给定。字符串在 MATLAB 中被看作是一个矢量，每个字符构成矢量的元素。例如 `str='hello'` 等价于 `str=['h','e','l','l','o']`。

3) 复数

在 MATLAB 中如果标识符 `i` 或 `j` 没有被定义过，那么一个复数可以简单地用两个实数 `a`, `b` 组成：`d=a+j*b` 或 `d=a+i*b` 表示复数。许多人编程时习惯用 `i`、`j` 作循环变量，如果你的程序涉及复数运算的话，最好避免使用它们。

2.3.6 冒号 “:”

设 `a1`, `a2`, `step` 是 3 个实数标量。则 `a1:step:a2` 表示一组实数，其中第 1 个是 `a1`，第 2 个是 `a1+step`，依此类推，最后一个数小于等于 `a2`。`step=1` 时可省略成 `a1:a2`。例如矢量 `[1:3]` 就是矢量 `[1,2,3]`，而矢量 `[3.3:-1.25:0]` 等于矢量 `[3.3,2.05,0.8]`。

2.3.7 空矢量

没有元素的矢量称空矢量。用 `[]` 来表示。例如：

```
?a=[3:-2:5]
```

```
a =
```

```
[]
```

`a` 的结果是一个空矢量。

2.3.8 矢量的引用

若 `A` 是一个矢量，`A(k)` 是 `A` 的第 `k` 个元素，其中 `k` 的最小值为 1，最大值为 `length(A)`。所以函数 `length(A)` 的返回值又是矢量 `A` 的长度。

若 `A` 是一个行矢量，`B=A(2:7)` 是由 `A` 的第 2 个~第 7 个元素组成的行矢量。`C=[A,B]` 也是一个矢量，表示 `C` 的前 `length(A)` 个元素正好是矢量 `A`，后 `length(B)` 个元素正好是矢量 `B`。

若 `A` 是一个矢量，`B=A(1:2:length(A))` 是所有 `A` 的奇序数元素。例如，`A=[1:100]`，则矢量 `A(1:2:100)` 的元素是 1 到 100 之间的所有奇数。矢量 `A(30:30:100)` 等价于 `[30,60,90]`。矢量 `A([1,9,67])` 等价于 `[A(1),A(9),A(67)]`。

2.4 基本运算

2.4.1 矢量与标量的加减乘除

矢量加 (+)、减 (-)、乘以 (*) 或除以 (/) 一个标量时，其结果是该矢量的每一个元素加、减、乘、除以同一标量。例如 `A=[1:5]+1` 的结果是 `[2,3,4,5,6]`，`c=[3,1,1]*3+j` 的结果是复数矢量 `[9+j,3+j,3+j]`。

标量加(+)、减(-)或乘以(*)矢量的结果与矢量加(+)、减(-)或乘以(*)标量的结果相同,即矢量与标量的加、减、乘(+、-、*)运算满足交换律。但请注意:标量除以(/)矢量(a/A)是非法的。

2.4.2 矢量与矢量之间的加减

两个相同长度的矢量相加或相减是其对应的元素相加或相减。例如:

```
?a=[1:3]*5-[6:6:20]/3
```

```
a =          3          6          9
```

2.4.3 矢量与矢量之间的点乘“.”与点除“./”

若 A 与 B 是两个长度相同的矢量,则 A.*B 是 A 与 B 的对应元素相乘, A./B 是 A 与 B 的对应元素相除。例如 C=[1:3].*[2,5,5]的结果是[2,10,15]。D=[4,10,2]./[2,2,1]的结果是[2,5,2]。

2.4.4 幂“^”与点幂“.^”

对于标量 a、b, a^b 是以 a 为底, b 为指数的幂运算。例如 2^3=8;

对于标量 x 及矢量 A, A.^x 是对矢量 A 的每一个元素求以 x 为指数的幂运算。例如: [-1,4,9].^0.5 的结果是[j,2,3];

对于标量 x 及矢量 A, x.^A 是一个矢量,它的每一个元素是以 x 为底、以 A 的对应元素为指数的幂。例如 10.^[-2:2]的结果是[0.01,0.1,1,10,100]。

对于两个相同长度的矢量 A、B, A.^B 的结果是一个与 A 同长度的矢量。其元素是以 A 的对应元素为底,以 B 的对应元素为指数的幂。

2.4.5 inf 与 eps

一个标量除以零得 inf。inf 自身是一个标量,它加减乘除以任何实标量仍为 inf。但 inf/inf, inf-inf 及 inf*0 的结果是 NaN(Not a Number),它表示结果不确定。

标量 eps 相当于无穷小,它反映了运算的精度。对于 MATLAB for Windows, eps 的初始值为 2.2204e-16。eps 的值可任意设置,例如可设 eps=1e-20 或其他值。有时为了避免程序运行中除以零引起麻烦,我们可给除数加上 eps,例如

```
?x=[-2:2]*pi;y=sin(x)./x
```

```
Warning: Divide by zero
```

```
y =
```

```
1.0e-016 *
```

```
-0.3898    0.3898    NaN    0.3898   -0.3898
```

此处的显示表示 y 的值是 1.0e-16*[-0.3898,0.3898,NaN,0.3898,-0.3898]。由于 x(3) 是零,所以 sin(x(3))/x(3)发出告警,并给出结果为 NaN。将 x 加上 eps,前面的例子便成为

```
?x=[-2:2]*pi+eps;y=sin(x)./x
```

```
y =
```