

第 1 章

信号处理概况

1.1 引言

信号处理是一种广泛应用于电子工程、计算机科学、通信技术等领域的核心技术。它涉及对信号进行采集、滤波、放大、转换、分析、识别和压缩等一系列操作，旨在提高信号的质量、提取有用信息、降低噪声干扰、实现信号的有效传输和存储。

信号处理的基本流程包括以下几个环节。信号采集：通过传感器或其他设备将现实世界的物理信号转换为电信号，以便进行处理。信号预处理：对采集到的信号进行滤波、去噪、放大等操作，以提高信号的质量。信号转换：将模拟信号转换为数字信号，便于计算机进行处理。这一过程通常涉及采样、量化和编码。信号分析：对信号进行时域、频域、时频域等多角度分析，提取信号的特征参数。信号处理算法：根据具体应用需求，采用各种算法对信号进行处理，如滤波、预测、调制、解调等。信号识别与分类：通过对信号特征的分析，实现对信号的识别、分类和预测。信号压缩：为了节省存储空间和传输带宽，对信号进行有损或无损压缩。信号恢复与重建：在信号传输或处理过程中，尽量恢复原始信号，以减少失真。

信号处理技术在诸多领域发挥着重要作用，如语音处理、图像处理、视频处理、无线通信、生物医学信号处理等。随着人工智能、大数据等技术的发展，信号处理技术在不断进步，为人类社会的发展提供了有力支持。

1.2 MATLAB 信号处理工具箱

MATLAB 是 Matrix Laboratory（矩阵实验室）的缩写，是一款由美国 The MathWorks 公司开发的商业数学软件。MATLAB 是一种用于算法开发、数据可视化、数据分析以及数值计算的高级技术计算语言和交互式环境。除了矩阵运算、绘制函数 / 数据图像等常用功能外，MATLAB 还可以用来创建用户界面及调用其他语言（包括 C、C++、Java、Python 和 FORTRAN）编写的程序。

尽管 MATLAB 主要用于数值运算，但通过利用众多的附加工具箱（Toolbox），它也适合不同领域的应用，如控制系统设计与分析、图像处理、信号处理与通信、金融建模和分析等。另外，MATLAB 还有一个配套软件包 Simulink，其可提供一个可视化开发环境，常用于系统模拟、动态 / 嵌入式系统开发等方面。

MATLAB 的 30 多个工具箱大致可分为两类：功能型工具箱和领域型工具箱。功能型工具箱主要用来扩充 MATLAB 的符号计算功能、图形建模仿真功能、文字处理功能以及与硬件实时交互功能，能用于多种学科。领域型工具箱具有很强的专业性。MATLAB 的信号处理工具箱主要用来对已产生的信号进行分析和处理，包括滤波、去重、频率分析等，是一种专业性很强的领域型工具箱。

MATLAB 信号处理和无线通信类工具箱簇如表 1.1 所示。

表 1.1 MATLAB 信号处理和无线通信类工具箱簇

序号	MATLAB 系列	Simulink 系列
1	5G Toolbox	Audio Toolbox
2	Antenna Toolbox	Communications Toolbox
3	Audio Toolbox	DSP HDL Toolbox
4	Bluetooth Toolbox	DSP System Toolbox
5	Communications Toolbox	Mixed-Signal Blockset
6	DSP System Toolbox	Phased Array System Toolbox
7	LTE Toolbox	Radar Toolbox
8	Phased Array System Toolbox	RF Blockset
9	Radar Toolbox	SerDes Toolbox
10	RF PCB Toolbox	Wireless HDL Toolbox
11	RF Toolbox	
12	Satellite Communications Toolbox	
13	Sensor Fusion and Tracking Toolbox	
14	SerDes Toolbox	
15	Signal Integrity Toolbox	
16	Signal Processing Toolbox	
17	Wavelet Toolbox	
18	Wireless Testbench	
19	WLAN Toolbox	

高校信号处理类课程和实验主要涉及 Signal Processing Toolbox 和 DSP System Toolbox 这两个工具箱。

1.2.1 Signal Processing Toolbox

MATLAB Signal Processing Toolbox 提供了一些函数和App，用来分析、预处理及提取均匀和非均匀采样信号的特征。该工具箱包含可用于滤波器设计和分析、重采样、平滑处理、去趋势和功率谱估计的工具，并提供提取特征（如变化点和包络）、寻找波峰和信号模式、量化信号相似性以及执行 SNR 和失真等测量的功能。通过该工具箱还可以对振动信号执行模态和阶次分析。使用信号分析器，可以在时域、频域和时频域同时预处理和分析多个信号，而无须编写代码；探查长信号；提取感兴趣的区域。通过滤波器设计工具，可以从多种算法和响应中进行选择来设计和分析数字滤波器。这两个 App 都生成 MATLAB 代码。

MATLAB Signal Processing Toolbox 的体系架构和函数、实例数量统计如表 1.2 所示。

表 1.2 MATLAB Signal Processing Toolbox 的体系架构和函数、实例数量统计

序号	目 录	内 容	函数数量	实例数量
1	Signal Processing Toolbox 快速入门	Signal Processing Toolbox 基础知识学习		18
2	信号分析和可视化	使用信号分析器来可视化、预处理和探查信号	1	6
3	信号生成和预处理	对信号进行创建、重采样、平滑、去噪和去趋势处理	23	22
4	测量和特征提取	波峰、信号统计、脉冲和瞬态指标、功率、带宽、失真	55	22
5	变换、相关性和建模	互相关、自相关、傅里叶变换、DCT、Hilbert、Goertzel、参数化建模、线性预测编码	63	39
6	数字和模拟滤波器	FIR 和 IIR、单速率和多速率滤波器设计、分析和实现	106	33
7	频谱分析	功率谱、相干性、窗	47	19
8	时频分析	频谱图、同步压缩、重排、Wigner-Ville、时频边缘、数据自适应方法	22	10
9	振动分析	阶数分析、时间同步平均、包络频谱、模态分析、雨流计数	13	5
10	信号的机器学习和深度学习延伸	信号标注、特征工程、数据集生成	35	17
11	代码生成和 GPU 支持	生成可移植的 C/C++/MEX 函数，并使用 GPU 来部署或加速处理		7
小计			365	198

1.2.2 DSP System Toolbox

MATLAB DSP System Toolbox 提供算法、应用程序和示波器，用于在 MATLAB 和

Simulink 中设计、仿真和分析信号处理系统，可以为通信、雷达、音频、医疗设备、物联网和其他应用的实时 DSP 系统建模。使用 DSP System Toolbox，可以设计和分析 FIR、IIR、多速率、多级和自适应滤波器；可以采样流式传输来自变量、数据文件和网络设备的信号，以进行系统开发和验证。时间示波器（The Time Scope）、频谱分析仪（Spectrum Analyzer）和逻辑分析仪（Logic Analyzer）可以动态地可视化和测量流信号。对于桌面原型设计和部署到嵌入式处理器（包括 ARM Cortex 架构），该工具箱支持 C/C++ 代码生成。另外，它还支持从滤波器、FFT、IFFT 和其他算法生成精确的定点建模和 HDL 代码。算法包括 MATLAB 函数、System 对象和 Simulink 模块。

MATLAB DSP System Toolbox 的体系架构和函数、模块、实例数量统计如表 1.3 所示。

表 1.3 MATLAB DSP System Toolbox 的体系架构和函数、模块、实例数量统计

序号	目 录	内 容	函数数量	模块数量	实例数量
1	DSP System Toolbox 快速入门	DSP System Toolbox 基础知识学习			12
2	信号生成、操作和分析	创建、导入、导出、显示和管理信号	54	60	10
3	滤波器设计与分析	FIR、IIR、频率变换	130	25	23
4	滤波器实现	单速率、多速率和自适应滤波器	64	43	32
5	变换和频谱分析	FFT、DCT、频谱分析、线性预测	9	32	10
6	统计和线性代数	测量、统计、矩阵数学、线性代数	9	56	4
7	定点设计	浮点到定点转换，定点算法设计	34	27	7
8	代码生成	ARM Cortex-M 处理器和 ARM Cortex-A 处理器的仿真加速、代码生成和优化	1	1	16
9	应用	模拟雷达、通信和生物医学系统	13		6
10	DSP System Toolbox 支持的硬件	支持第三方硬件，如 ARM Cortex-M 和 ARM Cortex-A 处理器	35		
小计			349	244	120

1.3 Python 信号处理工具箱

1.3.1 Python 简介

Python 由荷兰数学和计算机科学研究学会的 Guido van Rossum 于 20 世纪 90 年代初设计，作为一门叫作 ABC 语言的替代品。Python 提供了高效的高级数据结构，还能简单地面向对象编程。Python 语法和动态类型以及解释型语言的本质，使它成为多数平台上写脚本和快速开发应用的编程语言，随着版本的不断更新和语言新功能的添加，逐渐被用于独立的、大型项目的开发。Python 解释器易于扩展，可以使用 C 或 C++（或者其他可

以通过 C 调用的语言) 扩展新的功能和数据类型。Python 也可用于可定制化软件中的扩展程序语言。Python 丰富的标准库, 提供了适用于各个主要系统平台的源码或机器码。

由于 Python 语言的简洁性、易读性以及可扩展性, 在国外用 Python 做科学计算的研究机构日益增多, 一些知名大学已经采用 Python 来讲授程序设计课程。例如卡内基梅隆大学的编程基础、麻省理工学院的计算机科学及编程导论就使用 Python 语言讲授。Python 拥有一个强大的标准库。Python 语言的核心只包含数字、字符串、列表、字典、文件等常见类型和函数, 而由 Python 标准库提供系统管理、网络通信、文本处理、数据库接口、图形系统、XML 处理等额外的功能。Python 标准库命名接口清晰、文档良好, 很容易学习和使用。Python 社区提供了大量的第三方模块, 使用方式与标准库类似。它们的功能覆盖科学计算、Web 开发、数据库接口、图形系统等领域, 并且大多成熟而稳定。众多开源的科学计算软件包提供了 Python 的调用接口, 如著名的计算机视觉库 OpenCV、三维可视化库 VTK、医学图像处理库 ITK。

1.3.2 Python 库简介

Python 专用的科学计算扩展库就更多, NumPy (Numerical Python)、SciPy 和 Matplotlib 就是 3 个十分经典的科学计算扩展库, 它们分别为 Python 提供了快速数组处理、数值运算以及绘图功能。

NumPy 是 Python 语言的一个扩展程序库, 支持大量的维度数组与矩阵运算, 此外也针对数组运算提供大量的数学函数库。SciPy 是世界上著名的 Python 开源科学计算库, 建立在 NumPy 之上, 用于数学、科学、工程学等领域。SciPy 函数库在 NumPy 库的基础上增加了众多的数学、科学以及工程计算中常用的库函数, 如线性代数、常微分方程求解、信号处理、图像处理、稀疏矩阵等。它用于有效计算 numpy 矩阵, 使 NumPy 和 SciPy 协同工作, 高效解决问题。Matplotlib 是 Python 编程语言及扩展包的通用图形用户界面工具包。因此, Python 语言及其众多的扩展库所构成的开发环境十分适合工程技术、科研人员处理实验数据、制作图表, 甚至开发科学计算应用程序。

1.3.3 信号处理工具箱

目前除了 MATLAB 的一些专业性很强的工具箱还无法被替代之外, MATLAB 的大部分常用功能可以在 Python 中找到相应的扩展库。与 MATLAB 相比, 用 Python 做科学计算有如下优点。首先, MATLAB 是一款商用软件, 并且价格不菲。而 Python 完全免费, 众多开源的科学计算库提供了 Python 的调用接口。用户可以在任何计算机上免费安装 Python 及其绝大多数扩展库。其次, 与 MATLAB 相比, Python 是一门更易学、更严谨的程序设计语言, 它能让用户编写出更易读、易维护的代码。最后, MATLAB 主要专注于工程和



视频讲解



科学计算。然而即使在计算领域，也经常会遇到文件管理、界面设计、网络通信等需求。而 Python 有着丰富的扩展库，可以轻易完成各种高级任务，开发者可以用 Python 实现完整应用程序所需的各种功能。

本书参考 MATLAB Signal Processing Toolbox 的体系架构及其函数和实例组织方式，对应整理并编写了基于 Python 的信号处理工具箱（Python Signal Processing Toolbox, PySPT）函数和实例。重点整理编写了如下 5 章的对应函数和实例：第 3 章信号生成和预处理、第 4 章测量和特征提取、第 5 章变换、相关性和建模、第 6 章数字和模拟滤波器、第 7 章频谱分析。



函数和实例都配以对应 MATLAB 的 Python 代码为示例，引导读者通过编程的方式来准确地理解信号处理的相关知识及其应用。

GitHub 是基于 Git 的一个代码托管平台，因为只支持 Git 作为唯一的版本库格式进行托管，故名 GitHub。GitHub 于 2008 年 4 月 10 日正式上线，除了 Git 代码仓库托管及基本的 Web 管理界面以外，还提供了订阅、讨论组、文本渲染、在线文件编辑器、协作图谱、代码片段分享等功能。开发者可以将代码在 GitHub 上开源；也可以浏览其他项目的代码，fork 到自己名下做修改，clone 回本地使用；还可以发起 pull request 向上游提交自己的修改。Gitee（码云）是开源中国社区推出的代码托管协作开发平台，支持 Git 和 SVN，提供免费的私有仓库托管。Gitee 专为开发者提供稳定、高效、安全的云端软件开发协作平台，无论是个人、团队还是企业，都能够实现代码托管、项目管理、协作开发功能。