

# 信号与图像的稀疏分解及初步应用

王建英 尹忠科 张春梅 著

西南交通大学出版社  
· 成 都 ·

## 内 容 提 要

信号与图像的稀疏分解是信号与图像的一种新的分解方法,在信号与图像的压缩编码、去噪、信号的时频分析与信号识别等方面有着极为广阔的应用前景,是信号与图像处理研究领域一个新的很有意义的研究方向。本书总结了国际上在这一研究方向的研究进展以及作者多年来的研究成果。在稀疏分解方法方面,重点介绍了作者关于信号与图像稀疏分解快速算法的研究成果。在稀疏分解应用方面,重点介绍了作者在信号处理及图像压缩编码方面的研究成果。

本书适合于从事信号与信息处理、图像处理与压缩编码等方面工作的科研工作人员和研究生学习、研究中使用。

---

### 图书在版编目(CIP)数据

信号与图像的稀疏分解及初步应用 / 王建英, 尹忠科, 张春梅著. — 成都: 西南交通大学出版社, 2006.7  
ISBN 7-81104-279-7

I. 信... II. ①王...②尹...③张... III. ①信号处理②图像处理 IV. TN911.7

中国版本图书馆 CIP 数据核字 (2006) 第 045313 号

---

### 信号与图像的稀疏分解及初步应用

王建英 尹忠科 张春梅 著

责任编辑 黄淑文  
责任校对 韩松云  
封面设计 本格设计

西南交通大学出版社出版发行  
(成都二环路北一段 111 号 邮政编码: 610031 发行部电话: 028-87600564)  
<http://press.swjtu.edu.cn>  
E-mail: cbsxx@swjtu.edu.cn

四川森林印务有限责任公司印刷

成品尺寸: 170 mm × 230 mm 印张: 12.375  
字数: 227 千字  
2006 年 7 月第 1 版 2006 年 7 月第 1 次印刷  
ISBN 7-81104-279-7  
定价: 29.00 元

图书如有印装问题 本社负责退换  
版权所有 盗版必究 举报电话: 028-87600562

# 序 言

在信号处理和图像处理中,对信号和图像的表示方式的研究是一个根本性的问题,关系到信号处理与图像处理的多个方面。用尽可能简洁的方式表示信号或图像,将为信号或图像的处理带来很大的方便。信号与图像的稀疏表示就是力求得到信号与图像的一种最简洁的表示方式,而得到信号或图像稀疏表示方式的过程,就是信号或图像的稀疏分解。

近年来,信号和图像的稀疏表示与稀疏分解,在信号处理和图像处理的某些方面已经获得了很好的应用效果,引起了信号处理与图像处理国际学术界的广泛关注,如近年来国际图像处理年会(IEEE ICIP)对此方面的研究每年均进行专题讨论,著名刊物《Signal Processing》也以专刊的形式予以报道。而国内此方面研究起步比较晚,报道比较少。因此,作为前期研究成果的总结,作者写作此书,抛砖引玉,以引起国内研究人员对此方面研究的关注,进而推动、促进此项研究在国内的开展。

本书第1章简单介绍在信号与图像稀疏分解中常用的基础知识,即信号空间理论,这一章是本书的基础。第2章总结目前信号与图像稀疏分解的研究现状,并结合信号与图像稀疏分解的MP和BP算法,分析了稀疏分解中常见的研究问题和存在的研究难题。第3章分析信号与图像稀疏分解中最常用的一种算法(MP算法)以及该算法常用的实现方法和存在的问题。第4、第5两章介绍信号的稀疏分解快速算法及在信号处理中的应用。第6、第7和第8章分别介绍图像稀疏分解快速算法及在图像处理中的应用研究。其中快速算法主要是根据过完备原子库的结构特性、原子的能量特性和稀疏分解过程的特性而提出的。在应用研究中,主要介绍了稀疏分解在图像去噪和压缩编码中的应用。特别是在图像压缩中的应用,是值得读者重点关注的,这是因为基于稀疏分解的图像压缩编码表现出了良好的视觉特性和巨大的压缩潜力,很可能成为未来图像低比特率压缩的标准。

本书作者近年来专心从事信号与图像的稀疏分解及应用的研究。三位作者均受到国家留学基金委员会的资助,到国外学习研究信号与图像的稀疏分解。此外,关于此方面的研究,王建英同志先后获得了四川省应用基础研究项目和重点科技攻关项目资助,尹忠科同志获得了教育部留学回国人员科研启动基金资助,张春梅同志获得了国家民族事务委员会的资助。作为科研成果的总结,

在重要学术期刊上已经发表论文（含正式录用论文）30余篇。本书就是结合三位作者的研究成果而编著的。

阅读本书的读者需要具有“信号分析”、“数字信号处理”、“高等代数”等方面的基础。另外，如果要学习研究图像稀疏分解，还需要有“数字图像处理”的基础。但阅读本书并不要求读者学习过“信号空间理论”，有关信号空间分析的内容在本书中将结合第一章内容自成体系。当然，熟悉信号空间理论，将更有利于读者学习。本书适合于从事信号与信息处理、图像处理与压缩编码的科研工作人员和研究生学习、研究中使用。

作 者

2006年4月

# 目 录

<b>第 1 章 信号空间理论基础</b> .....	1
1.1 信号集及其映射 .....	1
1.2 信号空间的基本概念 .....	22
参考资料 .....	48
<b>第 2 章 信号和图像的基于过完备原子库的         稀疏表示与稀疏分解</b> .....	49
2.1 引 言 .....	49
2.2 信号的稀疏逼近 .....	51
2.3 稀疏信号的精确重构条件 .....	52
2.4 过完备不相干级联原子库 .....	56
2.5 结束语 .....	57
参考资料 .....	57
<b>第 3 章 基于 MP 的信号与图像稀疏分解方法</b> .....	62
3.1 引 言 .....	62
3.2 基于 MP 的信号稀疏分解 .....	63
3.3 MP 算法的两大类实现方法 .....	67
3.4 MP 算法改进算法——OMP 算法 .....	69
参考资料 .....	70
<b>第 4 章 信号稀疏分解快速算法</b> .....	72
4.1 引 言 .....	72
4.2 原子能量特性及在快速算法中的应用 .....	72
4.3 基于智能计算和 MP 的信号稀疏分解 .....	76
4.4 基于原子库集合划分和 FFT 的信号稀疏分解 MP 算法 .....	86
4.5 总 结 .....	92
参考资料 .....	93

<b>第 5 章 信号稀疏分解在信号处理中的应用</b>	<b>94</b>
5.1 稀疏分解在信号去噪中的应用	94
5.2 稀疏分解在微弱信号检测中的应用	100
5.3 稀疏分解在阵列信号处理中的应用	104
5.4 结 论	114
参考资料	114
<b>第 6 章 基于 MP 的图像稀疏分解快速算法</b>	<b>117</b>
6.1 图像稀疏分解的原子库	117
6.2 基于原子能量特性的图像稀疏分解算法	118
6.3 在低维空间实现的图像稀疏分解算法	124
6.4 基于智能计算的图像 MP 稀疏分解	127
参考资料	138
<b>第 7 章 稀疏分解在图像去噪中的应用</b>	<b>140</b>
7.1 稀疏分解图像去噪原理	140
7.2 稀疏分解中图像上信息与噪声的区分	141
7.3 稀疏分解图像去噪与最佳模板去噪的比较	148
参考资料	151
<b>第 8 章 稀疏分解在图像低比特率压缩中的应用</b>	<b>152</b>
8.1 引 言	152
8.2 基于稀疏分解的图像压缩国内外研究现状及分析	153
8.3 基于稀疏分解的图像压缩方法	155
8.4 基于排序差分和稀疏分解的图像压缩编码方法	160
8.5 结 论	165
参考资料	166
<b>致 谢</b>	<b>169</b>
<b>附录 A 基于 MP 的信号稀疏分解参考程序</b>	<b>170</b>
<b>附录 B 基于 MP 的信号参数估计参考程序</b>	<b>176</b>
<b>附录 C 基于 GA 和 MP 的信号稀疏分解参考程序</b>	<b>179</b>
<b>附录 D 利用 FFT 实现基于 MP 的信号稀疏分解参考程序</b>	<b>186</b>

# 第1章 信号空间理论基础

在数字信号处理、数字图像处理等课程中，所研究的信号或图像通常是用低维简单空间中的复杂点集来表示。例如，通常表示一维信号用  $s(t)$ ，即信号幅度  $s$  是某一自变量  $t$  的函数；这样表示信号时，信号是二维空间中的点集  $\{(t, s)\}$ 。通常表示一幅图像用  $f(x, y)$ ，图像灰度  $f$  是图像行  $x$  和列  $y$  的二维函数；这样表示图像时，图像是三维空间中的点集  $\{(x, y, f)\}$ 。用低维简单空间中的复杂点集来表示信号或图像，形象直观，易于理解。但是用低维简单空间的复杂点集表示信号或图像，不利于采用现代分析数学的工具来研究信号或图像。为了充分利用现代分析数学的工具来研究信号或图像，把数学方面的新研究成果引入到信号或图像处理中，必须把信号或图像看成是高维复杂空间中的一个简单的点。为了便于读者理解信号或图像作为高维复杂空间中的点的表示方法，本书的第1章介绍信号与图像的空间理论，在此基础上讨论信号分解问题。这部分是全书的基础，内容可参考本章参考资料[1]。熟悉信号与图像空间理论的读者，可以直接阅读第2章的内容。

## 1.1 信号集及其映射

### 1.1.1 引言

在研究信号空间之前，我们先把任一特定的信号看作是信号集中的一个元素。这样，便可容易地形成把信号看作是信号空间中的一个点的概念。

在日常生活中常用到集合的概念，如多面体的面的集合、教室里听众的集合、直线上点的集合、自然数的集合……作为数学领域的集合概念，是数学上最广泛和最基本的概念，不能直接给出定义，只能用公理法展开。我们这里只能依赖我们的直观来描述这一概念。按照 Cantor 的说法，集合是由“确定的、

各别的对象  $m$  组成一个整体 (记为  $M$ ), 而这些对象是我们感觉到的或我们想象到的”。用来形成集合的对象称为该集合的元素或元。因此, 集合可以理解为“元素的汇合”、“元素的总体”, 等等。显然, 这些说法只不过是集合的同义语而已。

一般说来, 用大写字母  $A, B, C, S, \dots$  表示集合, 用小写字母  $a, b, c, x, \dots$  表示集合的元素。我们可以用两种方式表示一个集合的元素是什么。如果一个集合的元素不多, 可以直接列出它的全部元素。例如, 7 到 9 的自然数集可表示为  $A = \{7, 8, 9\}$ 。如果集合的元素很多, 我们可以用写出集合全体元素都满足的共同性质的办法来表示集合。例如, 上例中的集合可表示为  $A = \{x; x \text{ 是大于 } 6 \text{ 小于 } 10 \text{ 的整数}\}$ 。用这个办法来表示信号集  $S_p = \{x; p\}$ , 就意味着  $S_p$  是所有具有特性  $p$  的信号  $x$  的集合。

某元是一个集合的元素可以写作形如  $8 \in A$  的形式; 读作“8 是集合  $A$  的元素”或“8 属于  $A$ ”。如果一个元素不属于该集合, 可以记为形如  $6 \notin A$  的形式, 读作“6 不属于  $A$ ”, 或“6 不是  $A$  的元素”。

如果对于每一个  $x \in A$  都有  $x \in B$ , 则称集合  $A$  是集合  $B$  的子集, 或称  $B$  包含  $A$ , 记为  $A \subseteq B$  或  $B \supseteq A$ 。例如, 整数集便是实数集的一个子集。若  $A \subseteq B$  与  $B \subseteq A$  均成立, 则集合  $A$  与  $B$  的元素完全相同, 记为  $A = B$ 。如果  $A \subseteq B$ , 但  $A \neq B$ , 即  $B$  里至少有一个元素不属于  $A$ , 则称  $A$  是  $B$  的真子集, 记为  $A \subset B$ 。

不包含任何元素的集合叫空集, 记为  $\emptyset$ 。例如, 考察方程式根的集合, 如果该方程无根, 则其根的集合为空集。注意, 空集  $\emptyset$  与数  $0$  是完全不同的。 $0$  是一个元素, 如方程式唯一的根是  $0$ , 此时根的集合不是空集。显然, 任何集合包含空集  $\emptyset$  作为其子集。在信号理论中, 常遇到零信号  $\{0\} = \{x; x(t) = 0, t \in \mathbf{R}\}$ , 零信号是信号集中的一个元素, 与空集的概念完全不同。

有了上述关于集合的基本概念, 就很容易地了解任一信号不过是特定信号集中的一个元素, 而最常用的信号集如下节所述。

## 1.1.2 常用的信号集

### (1) 矩形信号集 $S_\tau$

矩形信号集  $S_\tau$  可表示为

$$S_\tau = \left\{ x; x(t) = A \prod \left( \frac{t - t_0}{\tau} \right), t_0 \in \mathbf{R}; A, \tau > 0 \right\} \quad (1.1)$$

$$\text{式中 } \prod(t) = \begin{cases} 1, & |t| \leq 1/2 \\ 0, & \text{其他} \end{cases} \quad (1.2)$$



显然,  $A$  为矩形信号幅度,  $\tau$  为信号持续期,  $t_0$  为矩形信号中心位置。给定  $A$ 、 $\tau$ 、 $t_0$  值, 则得到矩形信号集中的一个元素。矩形信号的傅里叶变换为

$$X(f) = \int_{-\infty}^{\infty} x(t) e^{-j2\pi ft} dt = A \int_{t_0 - \frac{\tau}{2}}^{t_0 + \frac{\tau}{2}} e^{-j2\pi ft} dt = A\tau \cdot \text{sinc}(f\tau) \cdot e^{-j2\pi ft_0} \quad (1.3)$$

式中,  $\text{sinc}$  函数定义为

$$\text{sinc}(z) = \frac{\sin(\pi z)}{\pi z} \quad (1.4)$$

## (2) 正弦信号集 $S_c$

正弦信号集  $S_c$  可表示为

$$S_c = \{x; x(t) = \text{Re}[e^{a+j(2\pi ft+\theta)}], \alpha, \theta, f \in \mathbf{R}\} \quad (1.5)$$

式中,  $e^a$ 、 $f$ 、 $\theta$  分别表示正弦信号的幅度、频率和相位, 显然正弦信号集  $S_c$  包含所有可能的幅度、频率和相位的正弦信号。

## (3) 对称信号集

对称信号集可分为偶对称信号集  $S_{ev}$  和奇对称信号集  $S_{od}$ , 并分别表示如下

$$S_{ev} = \{x; x(t) = x(-t), -\infty < t < \infty\} \quad (1.6)$$

$$S_{od} = \{x; x(t) = -x(-t), -\infty < t < \infty\} \quad (1.7)$$

偶对称信号的傅里叶变换只有实部, 即

$$\text{Im} X_{ev}(f) = 0 \quad (1.8)$$

而奇对称信号的傅里叶变换只有虚部, 即

$$\text{Re} X_{od}(f) = 0 \quad (1.9)$$

## (4) 周期信号集 $S_p(T)$

周期信号集  $S_p(T)$  是所有周期为  $T$  之信号的集合, 它可表示为

$$S_p(T) = \{x; x(t+T) = x(t)\} \quad (1.10)$$

周期信号具有离散的傅里叶谱, 可用傅里叶级数表示为

$$x(t) = \sum_{n=-\infty}^{\infty} C_n e^{j2\pi n \frac{t}{T}}, \quad -\infty < t < \infty \quad (1.11)$$

式中

$$C_n = \frac{1}{T} \int_{t_0}^{t_0+T} x(t) e^{-j2\pi n \frac{t}{T}} dt \quad (1.12)$$

(5) 幅度有界信号集  $S_m(K)$

幅度有界信号集  $S_m(K)$  表示信号幅度的瞬时值总不大于某一正实数  $K$  之信号集合, 它可表示为

$$S_m(K) = \{x; |x(t)| \leq K, K > 0\} \quad (1.13)$$

显然, 如果  $K_2 \geq K_1$ , 则

$$x \in S_m(K_1) \Rightarrow x \in S_m(K_2)$$

(6) 能量有限信号集  $S_e(K)$

能量有限信号集  $S_e(K)$  可表示为

$$S_e(K) = \left\{ x; \int_{-\infty}^{\infty} x^2(t) dt \leq K \right\} \quad (1.14)$$

能量有限信号又称为平方可积信号。

(7) 时限信号集  $S_d(T)$

时限信号集表示在区间  $-T \leq t \leq T$  之外信号为零的所有信号的集合, 其数学表达式为

$$S_d(T) = \{x; x(t) = 0, |t| > T\} \quad (1.15)$$

显然, 如  $T_2 \geq T_1$ , 则

$$x \in S_d(T_1) \Rightarrow x \in S_d(T_2)$$

(8) 带限信号集  $S_b(B)$

带限信号集  $S_b(B)$  表示信号频谱在区间  $[-B, B]$  之外为零的所有信号的集合, 其数学表达式为

$$S_b(B) = \left\{ x; X(f) = \int_{-\infty}^{\infty} x(t) e^{-j2\pi f t} dt = 0, |f| > B \right\} \quad (1.16)$$

(9) 时域离散信号集  $S_s(\tau)$

时域离散信号集  $S_s(\tau)$  表示采样周期为  $\tau$  的所有时域离散信号的集合, 其数学表达式为

$$S_s(\tau) = \begin{cases} x(t) = x(n\tau) = x(n); & t = n\tau, n \in I \\ \text{无定义}; & t \neq n\tau \end{cases} \quad (1.17)$$

### 1.1.3 信号集的运算

信号集的运算是指已知若干信号集，通过运算得到新的信号集。集的基本运算有三种。

第一种运算是集合的并，定义为

$$A \cup B = \{x; x \in A \text{ 或 } x \in B\} \quad (1.18)$$

并集  $A \cup B$  由属于集合  $A$  或属于集合  $B$  的一切元素组成。

第二种运算是集合的交，定义为

$$A \cap B = \{x; x \in A \text{ 与 } x \in B\} \quad (1.19)$$

交集  $A \cap B$  由既属于集合  $A$  又属于集合  $B$  的一切元素组成。例如， $A$  表示偶数集， $B$  表示能被 3 整除的数集，则  $A \cup B$  为被 2 或被 3 整除的一切整数的集合，而  $A \cap B$  为被 6 整除的一切整数的集合。

集合的并与交运算有交换律、结合律以及分配律。交换律即

$$\begin{aligned} A \cup B &= B \cup A \\ A \cap B &= B \cap A \end{aligned}$$

结合律即

$$\begin{aligned} (A \cup B) \cup C &= A \cup (B \cup C) \\ (A \cap B) \cap C &= A \cap (B \cap C) \end{aligned}$$

分配律即

$$\begin{aligned} (A \cap B) \cup C &= (A \cup C) \cap (B \cup C) \\ (A \cup B) \cap C &= (A \cap C) \cup (B \cap C) \end{aligned}$$

第三种集合的运算是集合的差，定义为

$$S = A - B = \{x; x \in A \text{ 且 } x \notin B\} \quad (1.20)$$

差集  $A - B$  是由属于集合  $A$  但不属于集合  $B$  的一切元素构成的集合。例如， $A$  是整数集， $B$  是奇数集，则  $A - B$  表示偶数集。

在研究集合（包括信号集）时，采用集合之间的关系图，即 Euler-Venn 图是有帮助的。通常将基础集合画成一个大的矩形，用  $U$  表示，而把所讨论的集合画在矩形内。图 1.1 给出了并集、交集和差集的 Euler-Venn 图。

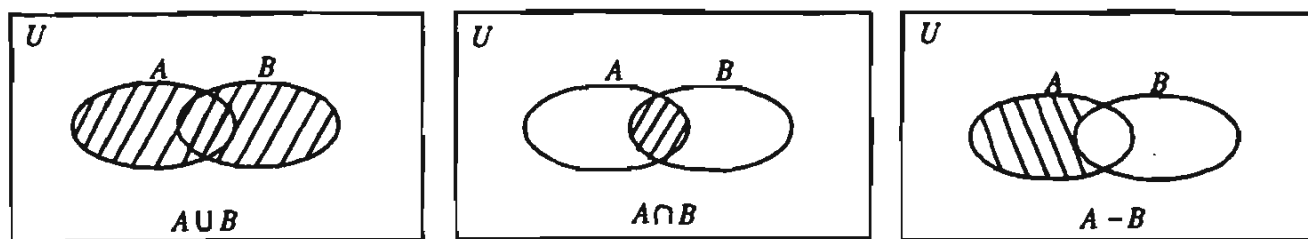


图 1.1 两集合的并集、交集和差集的 Euler-Venn 图

下面举出两个信号集运算的例子。

例 1.1 已知信号集  $S_a$  为

$$S_a = \{x_n(t); n=1, 2, 3, \dots\} \quad (1.21)$$

式中 
$$x_n(t) = \begin{cases} ne^{-nt}, & t \geq 0 \\ 0, & t < 0 \end{cases}$$

试求  $S = S_a \cap S_m(12) \cap S_e(3)$ 。

由式 (1.21) 有  $\max |x_n(t)| = n$ , 且  $\int_{-\infty}^{\infty} x_n^2(t) dt = \frac{n}{2}$ , 显然, 因  $S_m(12)$  要求  $n \leq 12$ ,  $S_e(3)$  要求  $\frac{n}{2} \leq 3$ , 故有  $n \leq 6$ , 则

$$S = \{x_n(t); n=1, 2, \dots, 6\}$$

例 1.2 众所周知, 一个非零的信号不可能既是带限的又是时限的。这是因为, 时限信号的傅里叶变换为  $\int_{-T}^T x(t) e^{-j2\pi ft} dt$ , 它只能在一些  $f$  的孤立点上为零, 除非当  $|t| \leq T$  时,  $x(t) \equiv 0$ 。也就是说, 只有当  $|t| \leq T$  时,  $x(t) \equiv 0$ , 才可能使上述积分在一个  $f$  的区间上为零, 因此, 时限信号中只有零信号才是带限的, 即

$$S_d(T) \cap S_b(B) = \{0\} = \{x; x(t) = 0, t \in \mathbf{R}\} \quad (1.22)$$

#### 1.1.4 信号集的划分与等价关系

为便于掌握一个信号集, 常常需要把信号集划分成一些互不相交的子集, 如图 1.2 所示。

从数学上讲, 把集合  $S$  划分为  $\{S_1, S_2, \dots\}$ , 可表示为

$$S = S_1 \cup S_2 \cup S_3 \cup \dots \quad (1.23a)$$

和

$$S_i \cap S_j = \emptyset, \quad i \neq j \quad (1.23b)$$

例如，在人口统计中，按出生年把成都市的居民进行编组，即得出居民集合中一系列互不相交的子集，以便掌握人口的年龄结构。又如，三维空间点的集合，可以划分为不同半径的球面组成的子集。总之，集合划分的目的是为了便于处理集内的元素，也是为了产生一系列便于处理的信号子集。例如，通常一个信号集内包含了不可数无限多个元素，如果我们把这个集合划分成可数个，甚至有限个子集，则处理这些信号就方便得多。下面我们将给出一些在工程中常用的划分信号集的例子。

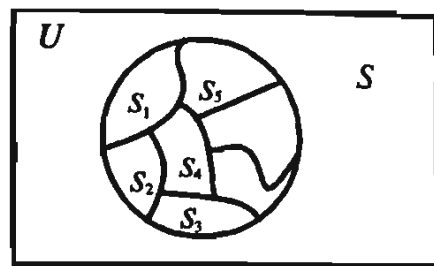


图 1.2 信号集的划分

为了给出集合  $M$  的一个划分，其规则不可能是任意的。通常，一个划分是由集内元素的等价关系产生的。两个元素  $a, b$  等价记为  $a \sim b$ ，必须要下述三个公设成立：

$$\left. \begin{array}{l} \text{① 自反性: } a \sim a; a \in M \\ \text{② 对称性: } a \sim b \Rightarrow b \sim a; a, b \in M \\ \text{③ 传递性: } a \sim b \text{ 和 } b \sim c \Rightarrow a \sim c; a, b, c \in M \end{array} \right\} \quad (1.24)$$

例如，在实数集合中不能定义  $b > a$  作为  $b \sim a$ 。因为  $a \not> a$ ，故  $a \sim a$  不成立，无自反性。同时， $b > a \Rightarrow a \not> b$ ， $a \sim b$  不成立，无对称性。因此，用  $b > a$  作为等价关系是不成立的，也不能形成一种划分。又如，是否能把成都市的居民按互相认识作为等价关系？ $a$  认识  $a$ ，意味着  $a$  认识  $a$ ， $a$  认识  $b$  意味着  $b$  认识  $a$ ，所以有自反性和对称性。但如  $a, b$  互相认识， $b, c$  互相认识，并不意味着  $a, c$  互相认识，无传递性，故不能用互相认识作为等价关系，因而也不能形成一种划分。

不难证明，满足式 (1.23) 的任何一个划分产生一个等价关系，反之，满足式 (1.24) 的任何一个等价关系产生一个划分。因此，一个划分就是把集合  $S$  分解为互不相交的等价子集，其中每一个等价子集可表示为

$$S_x = \{y; y \sim x\} \quad (1.25)$$

也就是说，一种划分的每个子集内各元素等价，反之，等价元素组成的集合为一种划分的一个子集。

划分和等价关系在信号理论中有着广泛的用途，下面是一些十分有用而且有趣的例子。

**例 1.3** 在数论中，通常把整数集合  $\{n; n = 0, \pm 1, \pm 2, \dots\}$  划分为有限个 ( $m$  个) 等价集：

$$S_i = \{n; n = pm + i\}, i = 0, 1, 2, \dots, m-1 \quad (1.26)$$

式中,  $p$  为任意整数。此时, 等价关系就是模  $m$  同余, 即  $n_1 \sim n_2 \Rightarrow n_1 - n_2 = pm \Rightarrow n_1 = n_2 \pmod{m}$ 。例如, 模 2 同余把整数集划分为奇数和偶数两个等价子集, 而模 3 同余把整数集划分为与 0、1、2 三个数同余的三个子集, 后者如图 1.3 所示。

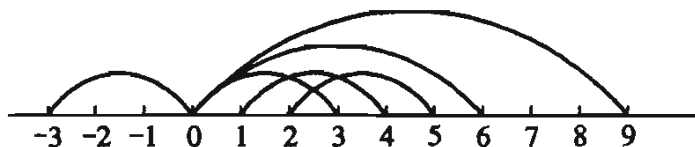


图 1.3 模 3 同余的等价关系示意图

**例 1.4** 对于信号集  $S$ , 我们可以建立等价关系  $x(t_0) \cdot y(t_0) > 0 \Rightarrow x \sim y$ , 即  $x(t)$  与  $y(t)$  在  $t_0$  时刻同号, 则认为  $x$ 、 $y$  等价。按此等价关系, 可把信号集  $S$  划分为如下两个等价集:

$$\begin{aligned} S_+ &= \{x; x(t_0) > 0\} \\ S_- &= \{x; x(t_0) < 0\} \end{aligned} \quad (1.27)$$

上述划分排除了  $x(t_0) = 0$  的情况, 因为  $x(t_0)$  恰好为零的概率极小。如果把  $x(t_0) = 0$  的情况考虑在内, 则可把它附加在  $S_+$  或  $S_-$  的等价集内, 此时, 式 (1.27) 可改写为如下形式:

$$\begin{aligned} S_+ &= \{x; x(t_0) \geq 0\} \\ S_- &= \{x; x(t_0) < 0\} \end{aligned} \quad (1.28)$$

这种划分广泛地应用于二进制数据传输系统, 此时,  $S_+$  集合中所有信号看作是由发射端的  $x_1(t)$  信号产生, 而  $S_-$  集合中所有信号看作是由发射端的  $x_2(t)$  信号产生。图 1.4 给出了这种系统的典型框图。

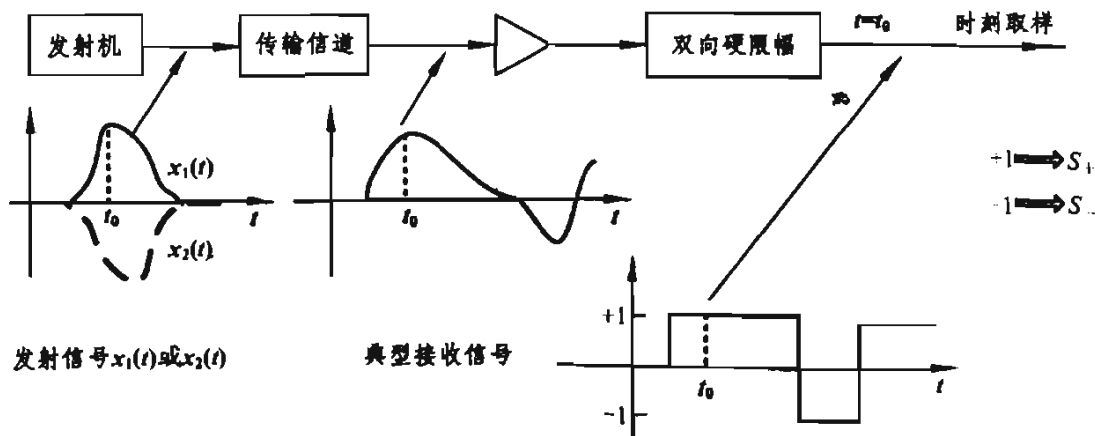


图 1.4 二进制数据传输系统

从图 1.4 可看出, 虽然发射机仅可能发射两种信号  $x_1$  或  $x_2$ , 但由于信道对信号的畸变及各种干扰的影响, 在接收机输入端的信号是不可预计的, 有无限多种可能的波形。信号处理系统就是对这些接收波形进行划分, 从而推测发射波形是  $x_1$  或  $x_2$ 。如果接收信号是属于集合  $S_+$ , 则判决发射端信号为  $x_1$ ; 反之, 如果接收信号是属于集合  $S_-$ , 则判决发射端信号为  $x_2$ 。显然, 这种判决会出现错误, 如何选择判决准则和计算其错误概率, 那是检测理论所要解决的问题。这里我们要指出的是, 像“检测”这样的信号处理问题, 本质上就是对接收信号集进行一种特定的划分。

**例 1.5** 一种抗干扰性能较好的二进制数据传输系统是采用相关接收机把接收信号划分为两个子集  $S_1$  和  $S_2$ 。同上例一样, 接收信号  $y(t)$  属于  $S_1$  或  $S_2$  表明对发射端信号为  $x_1$  或  $x_2$  的判决。此时, 集合  $S_1$  和  $S_2$  可写为:

$$\begin{aligned} S_1 &= \left\{ y; \int_0^T y(t)\varphi(t)dt \leq K \right\} \\ S_2 &= \left\{ y; \int_0^T y(t)\varphi(t)dt < K \right\} \end{aligned} \quad (1.29)$$

式中,  $\varphi(t)$  为参考信号,  $K$  为某一预置门限。完成式 (1.29) 的运算设备可由一乘法器、积分器、取样器和门限设备构成, 如图 1.5 所示。

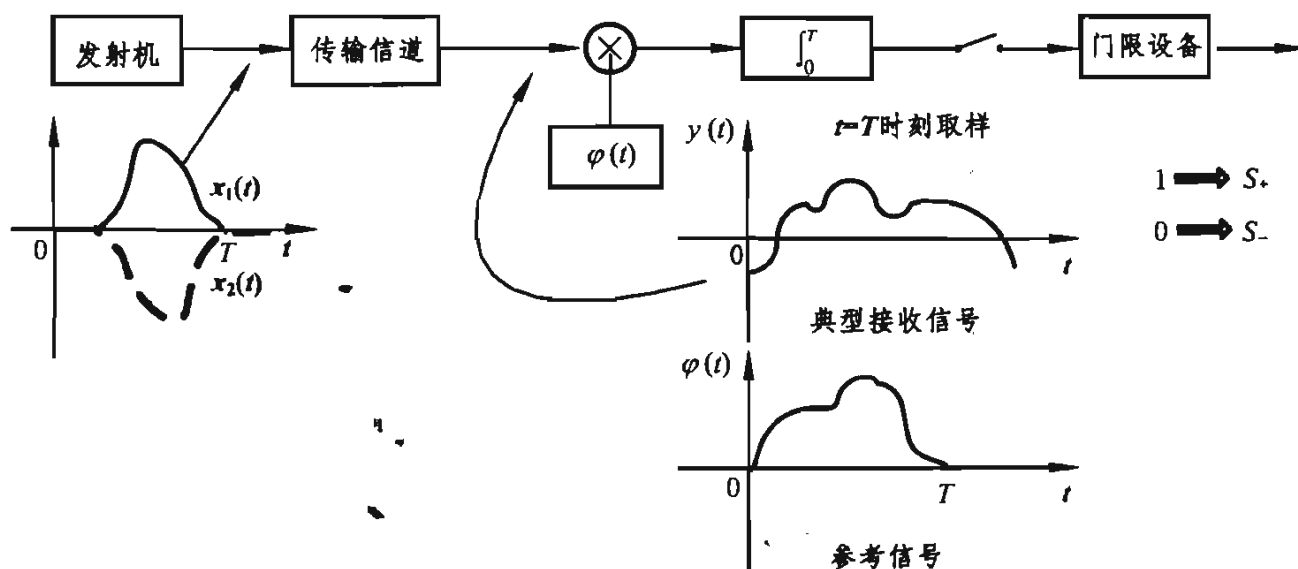


图 1.5 使用相关接收机的二进制数据通信系统

显然, 这个系统本质上是对接收信号进行相关运算, 并与一门限进行比较, 从而把接收信号集划分为两个子集, 以便判决发射信号是  $x_1$  或  $x_2$ 。有关最佳地选择  $\varphi(t)$  和  $K$  的问题是信号检测理论的重要课题。这里只需说明, 如果选择  $\varphi(t) = x(T-t)$ , 相关接收就完全与匹配滤波接收等效。

**例 1.6** 如果我们事先给出一个函数集  $\{\varphi_i; i=1, 2, \dots, n\}$ , 今后将会看到, 这个函数集内的元素通常是线性无关的, 则我们可定义函数  $x, y$  的等价关系为

$$x \sim y \Rightarrow \int_{-\infty}^{\infty} x(t)\varphi_i(t) dt = \int_{-\infty}^{\infty} y(t)\varphi_i(t) dt, \quad i=1, 2, \dots, n \quad (1.30)$$

显然, 式 (1.30) 可以改写为

$$x \sim y \Rightarrow x - y \in M \Rightarrow x = y \pmod{M} \quad (1.31)$$

式中,  $M$  定义为一个函数子集

$$M = \left\{ z; \int_{-\infty}^{\infty} z(t)\varphi_i(t) dt = 0, \quad i=1, 2, \dots, n \right\} \quad (1.32)$$

即  $M$  中的元素  $z(t)$  和  $\varphi_i(t)$  乘积之积分均为零, 后面一章的分析中将会看到, 这是指  $z(t)$  和  $\varphi_i(t)$  正交。在式 (1.31) 中, 我们实质上已定义  $x - y \in M$  为  $x, y$  模  $M$  广义同余, 因为式 (1.31) 在形式上与例 1.3 中  $n_1 - n_2 = pm \Rightarrow n_1 = n_2 \pmod{m}$  完全一致。

按式 (1.30) 形成的每个等价子集中, 可以挑选出一个典型元素  $\hat{x}$

$$\hat{x}(t) = \sum_{k=1}^n a_k \varphi_k(t) \quad (1.33)$$

则每个等价子集可表示为

$$S_{\hat{x}} = \{x; x \sim \hat{x}\} = \{x; x = \hat{x} + z, z \in M\} \quad (1.34)$$

这个表示法的重大意义在于, 只要选定的集合  $\{\varphi_i\}$  满足一定条件, 则可在等价子集  $S_{\hat{x}}$  和一个  $n$  元有序数组  $\{a_1, a_2, \dots, a_n\}$  之间建立一一对应关系, 因而每个等价子集  $S_{\hat{x}}$  可以看作  $n$  维向量空间中的一个点。注意, 这里  $n$  元有序数组是指  $n$  个有序的数的集合, 今后简称为  $n$  维数组。当然, 要完全解决这个问题, 必须进一步学习信号空间理论中一系列其他基本概念, 但这里至少可看出, 这种等价集 的表示法, 可以让我们更方便地表示和处理信号。

应当指出, 等价关系是一种特殊的关系, 即满足式 (1.24) 所列的三个公设关系。一般的, 在集合  $A$  上的关系, 可定义为序偶  $\{a, b\}$  所形成的一个集合, 此处,  $a \in A, b \in A$ , 一般常用  $R$  表示关系, 用  $aRb$  表示序偶  $\{a, b\}$  是关系集合的元素。注意, 这里序偶是指一对有序的元素。用这个观点来解释等价关系, 则意味着  $xRx$  (自反性),  $xRy \Rightarrow yRx$  (对称性),  $xRy$  和  $yRz \Rightarrow xRz$ 。

还应说明, 如果两个集合的元素之间可以建立一一对应关系, 则称它们互相等势。如果一个集合的某个子集与正整数集等势, 则称这个集合是无限集。



如果一个集合本身与正整数集等势，则称之为可数无限集。不是无限的集合称为有限的。有限集与可数无限集统称为可数的。不是可数的集合称为不可数无限集。我们前面所讨论的信号集，大多数是不可数无限集。

### 1.1.5 信号集的映射

1.1.4 节我们用等价关系和划分来表征一个信号集中元素的特征，本节我们将用元素间更一般的关系形式，即映射来表征信号。映射是函数概念的拓展。在数学分析里，函数关系是这样的，即设  $A$ 、 $B$  为实数集合  $\mathbf{R}$  中的两个子集，如果对每一个元素  $x \in A$ ，有一个确定的  $y = f(x)$ ， $y \in B$  与之对应，则在集  $A$  上定义了一个函数  $f$ ， $A$  为函数的定义域， $B$  为函数取值的集合，叫函数的值域。但是，信号集的元素并不一定是一个实数，它可能是一个实数序列（时域离散信号），也可能信号本身就是一个时间函数（时域连续信号）。把函数的概念推广到一般的集合，就是一个集合的元素到另一个集合的元素的映射。从数学上讲，设  $A$ 、 $B$  为两个非空集，如果存在一个规则  $f$ ，使  $A$  中任一元素  $x$ ，在规则  $f$  下，确定  $B$  里的一个元素  $y$  与之对应，即  $f: A \rightarrow B$ ，则称此规则为映射。映射也可以记为

$$y = f(x); x \in A \text{ 和 } y \in B \quad (1.35)$$

并称元素  $y$  为元素  $x$ （在映射  $f$  下）的象，称集  $A$  为映射  $f$  的定义域，所有元素  $x$  的象  $y$  的全体记为  $B'$  ( $B' \subseteq B$ )，称之为映射  $f$  的值域。如果  $B' = B$ ，则称  $f$  为  $A$  到  $B$  上的映射。若  $B' \subsetneq B$ ，则称  $f$  为  $A$  到  $B$  中的映射。

注意，映射总是单值的，即  $A$  中的每一个  $x$ ，只有唯一的一个象  $y$  与之对应。但反过来，并不要求所有  $x$  一定对应于不同的  $y$ 。如果可能有多个  $x$  的象为同一个元素  $y$ ，则称映射是多对一的。如果集  $A$  到集  $B$  的映射  $f$  为一对一的映射，即集  $A$  与集  $B$  中的元素存在着一一对应关系，则可以定义  $f$  的逆映射  $f^{-1}: B \rightarrow A$ 。

映射是数学中的基本概念之一，例如，实变函数是实数集到实数集的映射，积分可以看作是可积函数集到数集上的映射，求导可以看成是可导函数集到函数集的映射。广义来说，任何一种运算也可以看成是一种映射。信号处理系统就是对信号进行某些运算，因此信号处理系统的任何一个环节，都可以看成是一种信号集到信号集的映射。例如，采样过程是时域连续信号集到时域离散信号集的映射；幅度量化过程就是实数集到有限状态集的映射；前面所述相关接收机是输入信号集到相关系数（实数集）的映射。总之，任何一个信号处理系统可以看成如图 1.6 所示的系统，其输入为  $x \in X$ ，输出为  $y \in Y$ ，则它们可以

表示为一种映射  $f: X \rightarrow Y$ 。应当指出, 图 1.6 中集合  $X$  和  $Y$ , 可以是时域连续信号集, 也可以是时域离散信号集或者有限状态的码组, 因此, 它的应用是十分广泛的。

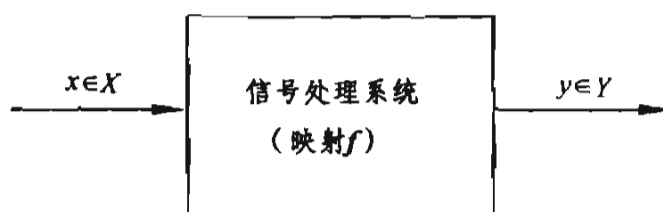


图 1.6 信号处理系统

一个信号处理系统可能包括两个或多个子系统, 此时, 用复合映射的观点来分析系统是十分方便的, 如图 1.7 所示。显然映射  $f: X \rightarrow Z$  是由两个映射  $f_1: X \rightarrow Y$  和  $f_2: Y \rightarrow Z$  复合而形成, 可记为  $f = f_2 f_1$ , 它表示

$$f: X \rightarrow Z \Rightarrow z = f_2(y) = f_2[f_1(x)] = f(x), x \in X \quad (1.36)$$

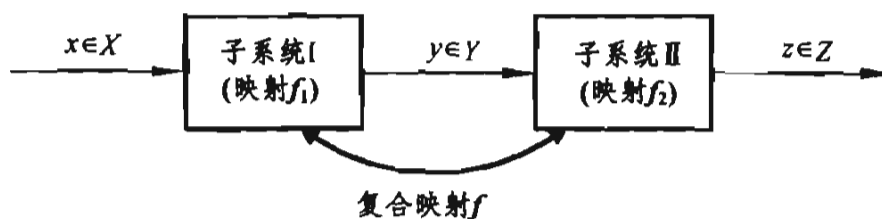


图 1.7 复合系统看作复合映射

为了进一步说明复合映射概念在信号处理中的应用, 我们回到例 1.5 所讨论的使用相关接收机的二进制数据通讯系统。此时图 1.7 中的子系统 I 完成相关运算, 即  $f_1$  为输入信号集  $Y$  到数集  $\mathbf{R}$  的映射:

$$f_1: Y \rightarrow \mathbf{R} \quad (1.37)$$

式中

$$Y = \{y; y(t), t \in \mathbf{R}\}$$

$$R = \left\{ r; r = \int_0^T y(t) \phi(t) dt \right\}$$

而  $f_2$  即子系统 II 完成的映射, 为集合  $\mathbf{R}$  到集合  $\{+1, 0\}$  的映射:

$$f_2: R \rightarrow \{+1, 0\} \quad (1.38)$$

式中

$$f_2(r) = \begin{cases} 1, & r \geq K \\ 0, & r < K \end{cases}$$

所以, 复合映射  $f = f_2 f_1$  可表示为

$$f(y) = \begin{cases} 1, & \int_0^T y(t)\varphi(t)dt \geq K \\ 0, & \int_0^T y(t)\varphi(t)dt < K \end{cases} \quad (1.39)$$

显然, 式 (1.39) 与式 (1.29) 完全一致。

上例已经说明, 任何一个划分或等价关系  $S_x = \{y; y \sim x\}$  可以表示为映射  $f_-$ , 且有

$$f_-: \{x\} \rightarrow \{S_x\} \Rightarrow f_-(x) = S_x \quad (1.40)$$

同样, 任何一个映射给出一种划分和等价关系, 例如,  $f: A \rightarrow B$ , 则有

$$f(x_1) = f(x_2) \Rightarrow x_1 \sim x_2 \quad (1.41)$$

式中,  $x_1, x_2 \in A$ 。例如, 求信号的能量可以看作能量有限信号集  $S_1$  到正实数集  $S_2$  的映射, 数学上可表示为

$$f(x) = \int_{-\infty}^{\infty} x^2(t) dt \quad (1.42)$$

图 1.8 形象地说明了这种映射关系。显然, 这种映射具有多对一的性质, 而由这个多对一映射定义了一种划分与等价关系, 即把信号能量相等作为信号的等价关系, 把能量有限信号按能量划分等价子集。

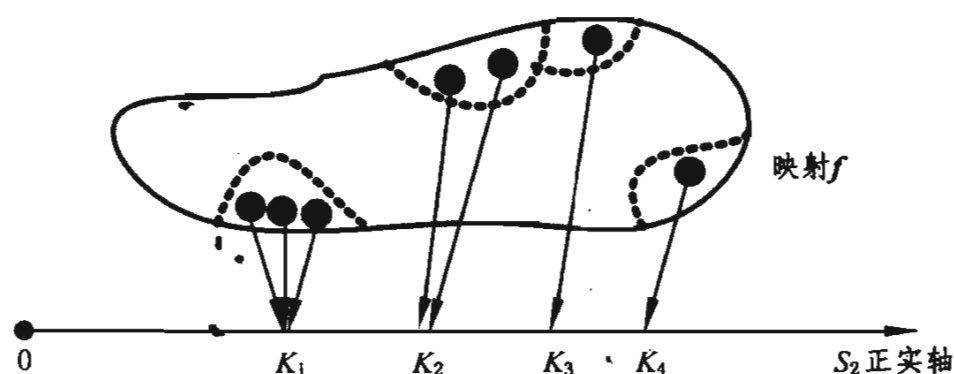


图 1.8 能量有限信号集到正实轴上的映射

例 1.7 信号分析中常用的傅里叶变换是平方可积函数 (能量有限信号) 集  $S_1$

$$S_1 = \left\{ x; \int_{-\infty}^{\infty} x^2(t) dt < \infty \right\}$$

向另一个平方可积函数集  $S_2$

$$S_2 = \left\{ X; \int_{-\infty}^{\infty} |X(f)|^2 df < \infty \right\}$$

的映射，记为：

$$F: S_1 \rightarrow S_2 \Rightarrow X(f) = \int_{-\infty}^{\infty} x(t)e^{-j2\pi ft} dt \quad (1.43)$$

严格说来，这个叙述还不够完全，因为  $x(t)$  平方可积是其傅里叶变换存在的充分条件，而不是必要条件。但是，就一般而言，我们最关心的首先是能量有限信号。

应当指出，傅里叶变换是多对一的映射，因为如果两个时间函数在任何有限区间内仅有可数个点取值不同，则其傅里叶变换是相同的，如图 1.9 所示。

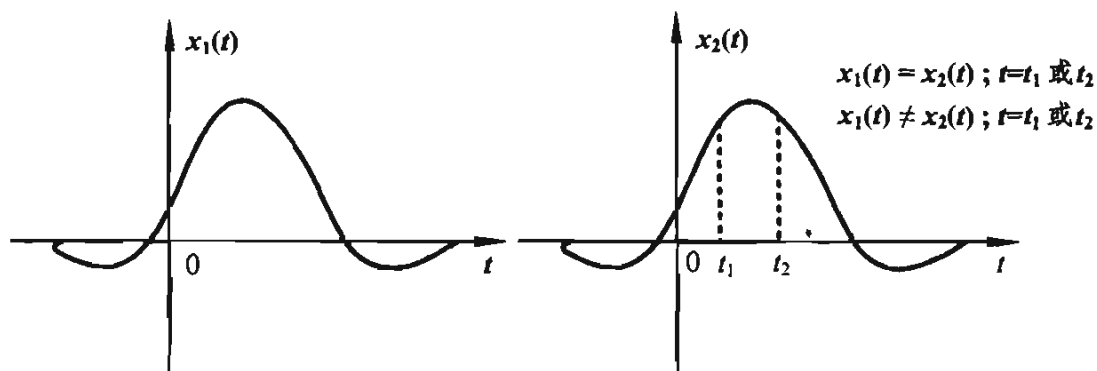


图 1.9 具有相同傅里叶变换  $X(f)$  的两个时间函数

因此，傅里叶变换  $F$  决定了时间函数的一个等价关系，即具有相同傅里叶变换  $X(f)$  的信号  $x(t)$  形成一个等价子集  $S_x = \{y; Y(f) = X(f)\}$ 。显然，在这个等价子集中的信号，只有可数个时刻出现差异，我们称之为“几乎处处相等”。在实际工程上，离散点上的差异并不重要，因而通常不再区分等价集中的信号，而用这个等价集中的任一信号来代表这个子集。于是，傅里叶变换就变成一对一映射，因而存在着逆映射  $F^{-1}$ ，即逆傅里叶变换：

$$F^{-1}: S_2 \rightarrow S_1 \Rightarrow x(t) = \int_{-\infty}^{\infty} X(f)e^{j2\pi ft} df \quad (1.44)$$

式 (1.43) 和式 (1.44) 定义的关系则称为傅里叶变换对。

应当指出，本书定义的傅里叶变换  $X(f)$ ，其变量为频率  $f$ ，而不是角频率  $\omega = 2\pi f$ 。这样做的好处在于傅里叶变换对具有相同的系数，便于记忆。同时，傅里叶变换的对偶性质亦具有简洁的形式。

如果已知

$$x(t) = q(t)$$

则傅里叶变换的对偶性质可表示为

$$X(f) = Q(f) = \int_{-\infty}^{\infty} q(t) e^{-j2\pi ft} dt \quad (1.45)$$

此时, 如果

$$X(f) = q^*(f)$$

则

$$x(t) = \int_{-\infty}^{\infty} q^*(f) e^{j2\pi ft} df = \left\{ \int_{-\infty}^{\infty} q(f) e^{-j2\pi ft} df \right\}^* = [Q(t)]^* \quad (1.46)$$

式 (1.45) 和式 (1.46) 表明, 信号及其傅里叶变换具有明显的共轭对偶性质。

### 1.1.6 信号集的泛函

从信号集到数集的映射在工程上十分重要, 因为我们通常希望得到信号的特定测量值, 从而可能定量地描述一个信号。信号的所有测量值, 例如信号的功率、某瞬时的幅度、某频率分量的大小等, 都可看作信号到数集的映射。从一般集合到数集的映射, 通常也可称为函数。但由于信号集的元素本身是函数 (如时间函数、频率函数等), 为了避免混淆, 我们通常把由集合到数集的映射称为泛函。所以, 可以把泛函理解为“一般集上的函数”, 对于函数集, 可以把它理解为“函数的函数”。泛函一般记为  $f(x)$ , 其中,  $x$  表示信号集上的元素, 而  $f(x) \in \mathbf{R}$ 。

常用的信号集上的泛函有

$$f_1(x) = \int_{-\infty}^{\infty} x(t) \varphi(t) dt \quad (\text{信号与 } \varphi(t) \text{ 的相关})$$

$$f_2(x) = \int_{-\infty}^{\infty} w(t) x^2(t) dt \quad (\text{信号在时窗 } w(t) \text{ 内的能量})$$

$$f_3(x) = \int_{-\infty}^{\infty} x(t) e^{-j2\pi f_0 t} dt \triangleq X(f_0) \quad (\text{信号在 } f_0 \text{ 处的频谱密度})$$

$$f_4(x) = \int_{-\infty}^{\infty} x(t) \delta(t - t_0) dt \triangleq x(t_0) \quad (\text{信号在 } t_0 \text{ 时的采样值})$$

$$f_5(x) = \int_{-\infty}^{\infty} x(t) \delta^{(n)}(t - t_0) dt \triangleq (-1)^n \left. \frac{d^n x}{dt^n} \right|_{t=t_0}$$

(信号  $n$  阶导数在  $t_0$  时的采样值)

$$f_6(x) = \max \{ |x(t)|; t_1 \leq t \leq t_2 \} = \lim_{n \rightarrow \infty} \left[ \int_{t_1}^{t_2} |x(t)|^n dt \right]^{1/n}$$

(信号在  $[t_1, t_2]$  内的最大绝对值) (1.47)

在上述泛函数表示中,  $\triangleq$  表示“按定义等于”。因此  $f_4(x)$  和  $f_5(x)$  给出了广义函数  $\delta(t)$  及其  $n$  阶导数的定义。应当指出, 泛函作为一种映射的值域是数集, 一般指的是实数集  $\mathbf{R}$ 。但我们可以把它推广到复数集合  $\mathbf{C}$ , 例如  $f_3(x) \in \mathbf{C}$ , 因为时间函数在  $f_0$  处的傅里叶谱是一个复数。一个复数本质上是由两个实数组合而成, 例如  $X(f_0)$  可以写成  $|X(f_0)|e^{j\varphi(f_0)}$ , 其中  $|X(f_0)|$  表示信号  $x(t)$  在  $f_0$  处频率分量的幅度,  $\varphi(f_0)$  为其相位, 均具有明显的物理意义, 而且都是可以测量的。因此, 今后我们在讨论泛函时, 认为它是一般集合向复数集合的映射, 而实数泛函仅看作其特例。

应用泛函的概念, 就很容易把信号近似地表示为级数的形式。设  $\{\varphi_k(t); k=1, 2, 3, \dots\}$  为一个给定的基本信号集, 其元素是可数的。此时, 信号可近似表示为

$$x(t) \cong \sum_k f_k(x) \varphi_k(t); t \in T \quad (1.48)$$

式中,  $\{f_k; k=1, 2, 3, \dots\}$  为泛函的一个可数序列。这样, 信号集中的每一个元素就可用可数个泛函来表示, 因而可用可数维空间中的一个点来表示, 从而大大方便了对信号的分析与处理。

信号级数表示法的一个最常见的例子, 就是信号的时间级数表示法。此时, 基本函数集  $\{\varphi_k(t)\}$  是由一个脉冲型的信号  $\varphi(t)$  通过一系列延时形成:

$$\{\varphi_k(t); \varphi_k(t) = \varphi(t - k\tau), k = 0, \pm 1, \pm 2, \dots\} \quad (1.49)$$

如果  $\varphi(t)$  具有如下特性

$$\left. \begin{aligned} \varphi(0) &= 1 \\ \varphi(k\tau) &= 0, k \neq 0 \end{aligned} \right\} \quad (1.50)$$

则称  $\varphi(t)$  为内插脉冲。显然, 应用内插脉冲作基信号, 任意连续信号可以近似用时间级数表示为

$$x(t) \cong \sum_k f_k(x) \varphi_k(t) = \sum_k x(k\tau) \varphi(t - k\tau) \quad (1.51)$$

式中的泛函  $f_k(x)$  可简单地用信号采样值  $x(k\tau) = x(t)|_{t=k\tau}$  表示。显然, 由于  $\varphi(t)$  具有式 (1.50) 的性质, 当  $t = k\tau$  时, 式 (1.51) 准确成立。由于事先假定  $x(t)$  为连续函数, 当  $t \neq k\tau$  时, 式 (1.51) 也近似成立, 而且当  $\tau$  越小时, 式 (1.51) 精度越高。式 (1.51) 的实质, 是用  $x(k\tau)$  值内插产生  $t \neq k\tau$  时的  $x(t)$  近似值, 因而  $\varphi(t)$  称为内插脉冲是合理的。

图 1.10 给出了三种内插脉冲实例, 用图 1.10 所示的三种内插脉冲作基信号近似表示一个连续信号, 如图 1.11 所示。应当说明, 图 1.11 是各采样点用直线相连接近似表示一个连续信号, 而图 1.11 是用一系列与采样值等幅的矩形来近似表示信号。这后一种情况, 正是工程上常用采样保持电路交换时域连续信号时的波形。可以看出, 信号的时间级数表示法在  $t \neq k\tau$  时只是一种近似的表示法。下节我们将说明, 在什么条件下, 这是一种准确的信号表示法。

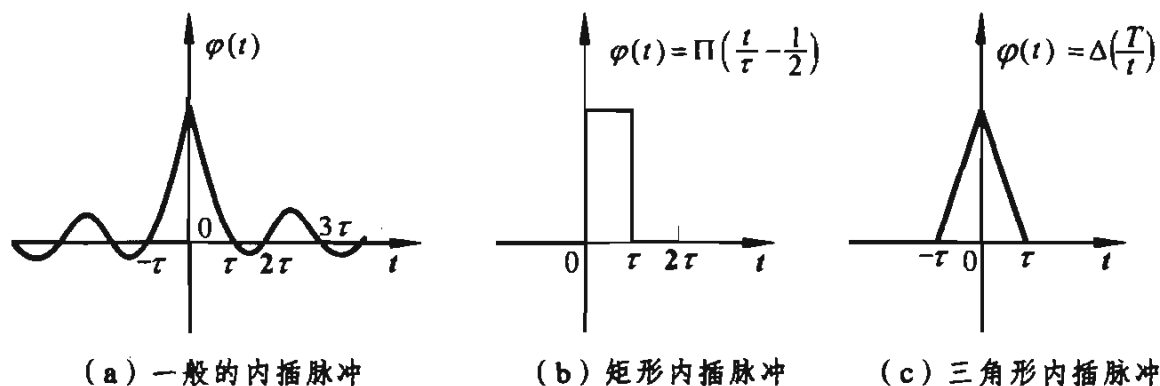


图 1.10 内插脉冲实例

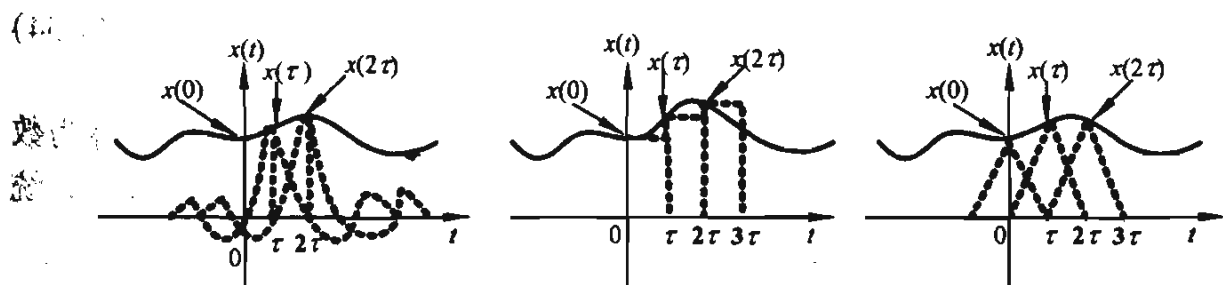


图 1.11 信号的时间级数表示法

信号的另一种重要的级数表示法是傅里叶级数表示法。如果  $x \in S_d(T/2)$ , 则有

$$x(t) = \sum_{n=-\infty}^{\infty} C_n e^{j2\pi n \frac{t}{T}}; \quad |t| \leq T/2 \quad (1.52)$$

式中, 级数的展开系数  $C_n$  (即傅里叶系数) 由如下泛函给出

$$f_n(x) = C_n = \frac{1}{T} \int_{-T/2}^{T/2} x(t) e^{-j2\pi n \frac{t}{T}} dt; \quad n = 0, \pm 1, \pm 2, \dots \quad (1.53)$$

显然，时限信号的傅里叶级数表示法式 (1.52) 和式 (1.53)，与周期信号的傅里叶级数表示式 (1.11) 和式 (1.12) 几乎完全相同，唯一的差别是，式 (1.52) 的成立区间为  $|l| \leq T/2$ ，而式 (1.11) 的成立区间为  $-\infty < l < \infty$ 。

下节我们将进一步讨论信号的时间级数表示法，因为这种表示法，建立了时域离散与时域连续系统的桥梁，在理论和实践上有重大意义。

### 1.1.7 信号的时域离散处理

利用时域连续信号的时间级数表示法，就很容易分析信号的时域离散处理系统，这个系统的一般框图如图 1.12 所示。

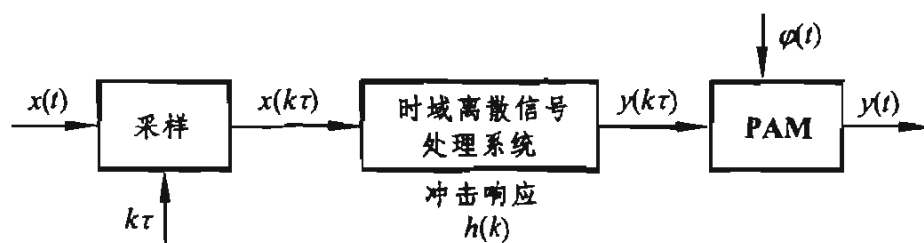


图 1.12 信号的时域离散处理系统

图 1.12 中，采样电路是求得信号时间级数表示式中的泛函，从而用离散信号  $x(k) = x(k\tau)$  表示  $x(t)$ 。时域离散信号处理系统一般为一个冲激响应为  $h(k)$  的时域离散滤波器，其频率特性为

$$H(e^{j2\pi f'}) = \sum_{k=-\infty}^{\infty} h(k)e^{-j2\pi kf'} \quad (1.54)$$

式中， $f' = f/f_s = \tau f$  表示对采样频率  $f_s = 1/\tau$  的归一化频率，有的作者称为数字频率。时域离散信号处理系统的输出  $y(k) = y(k\tau)$ ，应为输入序列  $x(k)$  与系统冲激响应的卷积

$$y(k) = x(k) * h(k) = \sum_{n=-\infty}^{\infty} x(k-n)h(n) \quad (1.55)$$

对式 (1.55) 两边取傅里叶变换，则

$$Y(e^{j2\pi f'\tau}) = X(e^{j2\pi f'\tau}) \cdot H(e^{j2\pi f'\tau}) \quad (1.56)$$

式 (1.56) 中，离散序列  $x(k)$  的傅里叶变换按定义为

$$X(e^{j2\pi f'\tau}) = \sum_{k=-\infty}^{\infty} x(k)e^{-j2\pi kf'\tau} \quad (1.57)$$



图 1.12 中, PAM 表示脉冲幅度调制器, 即用采样序列  $y(k)$  去控制内插脉冲  $\varphi(t-k\tau)$  的幅度:

$$y(t) = \sum_k y(k\tau) \cdot \varphi(t-k\tau) \quad (1.58)$$

对式 (1.58) 两边取傅里叶变换, 则

$$\begin{aligned} Y_a(f) &= \int_{-\infty}^{\infty} y(t) e^{-j2\pi ft} dt = \sum_k y(k\tau) \int_{-\infty}^{\infty} \varphi(t-k\tau) e^{-j2\pi ft} dt \\ &= \left[ \sum_k y(k\tau) e^{-j2\pi k\tau f} \right] \cdot \int_{-\infty}^{\infty} \varphi(t-k\tau) e^{-j2\pi f(t-k\tau)} dt = Y(e^{j2\pi f\tau}) \cdot \Phi(f) \end{aligned} \quad (1.59)$$

式中,  $\Phi(f)$  是选定的内插脉冲  $\varphi(t)$  的傅里叶变换。

把式 (1.56) 代入式 (1.59), 得

$$Y_a(f) = \Phi(f) \cdot H(e^{j2\pi f\tau}) \cdot X(e^{j2\pi f\tau}) \quad (1.60)$$

式 (1.60) 给出了图 1.12 系统输出的傅里叶变换与输入信号  $x(t)$  的采样  $x(k)$  的傅里叶变换之间的关系。为了得到  $y(t)$  与  $x(t)$  两者傅里叶变换之间的关系, 我们需要推导一个重要的关系式——泊松求和公式, 以便把  $x(t)$  的傅里叶变换与其采样  $x(k)$  的傅里叶变换联系起来。

按定义,  $x(t)$  的采样序列  $x(k)$  应满足

$$\begin{aligned} x(k) &= \int_{-\infty}^{\infty} X_a(f) e^{j2\pi k\tau f} df = \sum_{l=-\infty}^{\infty} \int_{(2l-1)/2\tau}^{(2l+1)/2\tau} X_a(f) e^{j2\pi k\tau f} df \\ &= \sum_{l=-\infty}^{\infty} \int_{-\frac{1}{2\tau}}^{\frac{1}{2\tau}} X_a\left(f - \frac{l}{\tau}\right) e^{j2\pi k\tau f} e^{-j2\pi lk} df = \sum_{l=-\infty}^{\infty} \int_{-\frac{1}{2\tau}}^{\frac{1}{2\tau}} X_a\left(f - \frac{l}{\tau}\right) e^{j2\pi k\tau f} df \end{aligned} \quad (1.61)$$

又由于  $x(k)$  为  $X(e^{j2\pi f\tau})$  的逆傅里叶变换, 即

$$x(k) = \int_{-1/2}^{1/2} X(e^{j2\pi f\tau}) e^{j2\pi k\tau f} df' = \tau \int_{-1/2}^{1/2} X(e^{j2\pi f\tau}) e^{j2\pi k\tau f} df \quad (1.62)$$

对比式 (1.61) 和式 (1.62), 显然有

$$X(e^{j2\pi f\tau}) = \frac{1}{\tau} \sum_{l=-\infty}^{\infty} X_a\left(f - \frac{l}{\tau}\right) \quad (1.63)$$

这就是有名的泊松求和公式, 它也可写为数字频率的形式

$$X(e^{j2\pi f\tau}) = \frac{1}{\tau} \sum_{l=-\infty}^{\infty} X_a\left(\frac{f'}{\tau} - \frac{l}{\tau}\right)$$

式 (1.63) 说明, 一个模拟信号的离散采样序列的谱, 为原模拟信号的谱按  $\frac{1}{\tau}$  周期重复后的叠加, 如图 1.13 所示。

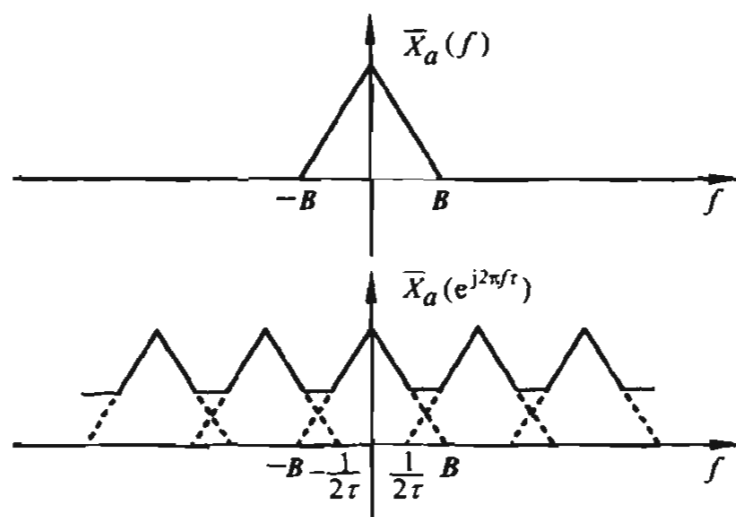


图 1.13 采样信号谱与原信号谱的关系

从图 1.13 可以看出, 如果采样率  $\frac{1}{\tau} < 2B$ , 则采样信号谱与原信号在  $[-B, B]$  区间内也会出现重大差异, 即出现混叠效应, 此时, 就不可能通过采样信号恢复原信号。因此, 对于一个带限信号  $x \in S_b(B)$ , 如果采样率  $\frac{1}{\tau} \geq 2B$ , 则可避免混叠效应, 因而可用一个通带为  $[-B, B]$  的理想低通滤波器滤波, 由采样信号恢复原模拟信号。通常称 2 倍于信号带宽的采样率为奈奎斯特速率。

有了泊松求和公式 (1.63), 就可由式 (1.60) 得到图 1.12 所示系统的输入输出关系式

$$Y_a(f) = \frac{1}{\tau} \Phi(f) \cdot H(e^{j2\pi f\tau}) \sum_{l=-\infty}^{\infty} X_a\left(f - \frac{l}{\tau}\right) \quad (1.64)$$

式 (1.64) 表明, 对模拟信号的时域离散处理系统, 其输出的谱为输入信号的周期重复叠加, 乘以内插脉冲的谱和时域离散系统的传递函数。

下面研究信号的时间级数表示法, 即式 (1.51) 在什么条件下准确成立, 使图 1.12 所示系统的输出  $y(t)$  准确复现输入信号。由式 (1.64) 可知, 如果我们令  $H(e^{j2\pi f\tau}) = 1$ , 同时要求  $x(t) \in S_b(B) = \{x; X(f) = 0, |f| > B\}$ , 且选  $\frac{1}{\tau} = 2B$ , 则有

$$Y_a(f) = \frac{1}{\tau} \Phi(f) X_a(f), \quad |f| < B \quad (1.65)$$

如果式 (1.65) 中  $\Phi(f)$  为一个理想的矩形函数, 即

$$\Phi(f) = \prod \left( \frac{f}{2B} \right) = \begin{cases} 1, & |f| \leq B \\ 0, & |f| > B \end{cases} \quad (1.66)$$

则式 (1.65) 可写为

$$Y_a(f) = \frac{1}{\tau} X_a(f), \quad -\infty < f < \infty \quad (1.67)$$

对式 (1.67) 两边进行傅里叶逆变换, 则有  $y(t) = \frac{1}{\tau} x(t)$ 。这说明, 只要  $x(t) \in S_b(B)$ ,  $\frac{1}{\tau} = 2B$ , 且  $\Phi(f)$  满足式 (1.66), 就可以用  $x(t)$  的采样信号  $x(k)$  完全恢复  $x(t)$ 。显然, 应用傅里叶逆变换于  $\Phi(f)$  的表达式 (1.66), 便可得到内插脉冲为

$$\varphi(t) = \text{sinc}(2Bt) = \frac{\sin(\pi 2Bt)}{\pi 2Bt} \quad (1.68)$$

因而,  $x(t)$  可根据信号的时间级数表示式 (1.58) 准确表示为

$$x(t) = \sum_{k=-\infty}^{\infty} x\left(\frac{k}{2B}\right) \text{sinc}(2Bt - k) \quad (1.69)$$

这就是有名的采样定理。它说明, 对带限信号  $x(t) \in S_b(B)$ , 可以利用式 (1.69) 所示的时间级数表示法, 准确用信号的采样值表示信号。

利用式 (1.46) 所表明的傅里叶变换的共轭对偶性, 则可把式 (1.69) 应用到时限信号  $x \in S_d(T)$ , 即有

$$X(f) = \sum_{m=-\infty}^{\infty} X\left(\frac{m}{2T}\right) \text{sinc}(2Tf - m); \quad -\infty < f < \infty \quad (1.70)$$

同样, 对于  $x \in S_d(T)$ , 有式 (1.52) 和式 (1.53) 所示的傅里叶级数表达式。应用式 (1.46) 所表明的傅里叶变换对的共轭对偶性, 则可把式 (1.52) 及式 (1.53) 应用到带限信号  $x \in S_b(B)$ , 即有

$$X(f) = \sum_{n=-\infty}^{\infty} d_n e^{-j\pi n \frac{f}{B}}; \quad |f| < B \quad (1.71)$$

式中

$$d_n = \frac{1}{2B} \int_{-B}^B X(f) e^{j\pi n \frac{f}{B}} df = \frac{1}{2B} x\left(\frac{n}{2B}\right) \quad (1.72)$$

## 1.2 信号空间的基本概念

### 1.2.1 距离空间

在1.1节中,我们把具有一定特性的信号归纳在一个信号集或一个等价子集中。合乎逻辑的下一步,显然是考察一个集合的各信号元素的差异,而且希望这个差异是可测量的。例如, $n$ 个采样值形成的一个时域离散序列,写作 $x(k) = \{x_1, x_2, \dots, x_n\}$ ,可以看成 $n$ 维空间 $R^n$ 中的一个点。为了量度 $x(k)$ 与另一个 $n$ 点序列 $y(k) = \{y_1, y_2, \dots, y_n\}$ 的差异,最简单的办法就是求这两个点的距离

$$\rho(x, y) = \left( \sum_{i=1}^n |x_i - y_i|^2 \right)^{\frac{1}{2}} \quad (1.73)$$

如果只考虑两点序列 $x = \{x_1, x_2\}$ 和 $y = \{y_1, y_2\}$ ,则它们的距离为 $\rho(x, y) = [(x_1 - y_1)^2 + (x_2 - y_2)^2]^{\frac{1}{2}}$ 。容易看出,这里的距离 $\rho(x, y)$ 具有如下性质:

- (1)  $\rho(x, y) \geq 0$ , 且 $x$ 和 $y$ 相同为 $\rho = 0$ 的充要条件;
- (2)  $\rho(x, y) \leq \rho(x, z) + \rho(y, z)$ 。

后一个不等式表示三角形两边之和不小于第三边,通常称为三点不等式,或三角形不等式。

$n$ 维欧氏空间引入的距离的概念,不仅便于衡量长度为 $n$ 的序列的差异,而且用来表示点列的极限。大家知道,点列 $\{x^i\}$ 趋向于极限点 $x$ 的充要条件是:

$$\rho(x^{(i)}, x) \rightarrow 0 \quad (i \rightarrow \infty)$$

对于时域连续信号,同样可以引入上述距离的概念。设 $C[a, b]$ 表示在区间 $[a, b]$ 上连续的时间函数的信号集,而 $x, y \in C[a, b]$ ,我们定义 $x, y$ 的距离为

$$\rho(x, y) = \max |x(t) - y(t)| \quad t \in [a, b] \quad (1.74)$$

读者可以自行证明,式(1.74)所定义的连续函数空间 $C[a, b]$ 的距离,完全满足式(1.73)定义的 $n$ 维空间的距离所具有的性质。这个距离的定义直观地表示在图1.14上。

显然,若 $x_i(t)$ ,  $x(t) \in C[a, b]$ ,则 $x_i(t)$ 均匀收敛于 $x(t)$ ,即点列 $x_i(t)$ 以 $x(t)$ 为极限地充要条件是

$$\rho(x_i, x) \rightarrow 0 \quad (i \rightarrow \infty)$$

在上面两种信号的情况下,  $\rho$  的意义是不同的, 但是它们有相同的特点。若是把  $R^n$ 、 $C[a, b]$  等看成抽象的空间, 把其中的函数或序列看成是空间中的点, 那么  $\rho(x, y)$  可以看成是两点  $x$  和  $y$  之间的距离。从上面也可看到, 极限过程能够用距离来描述, 而且在我们描述与极限相关联的概念时, 实质上也仅仅利用了距离和它的有关性质。因此, 有必要把这些具体信号空间中所定义的距离  $\rho$  抽象化, 对一般的集引进距离, 这就产生了距离空间的概念。

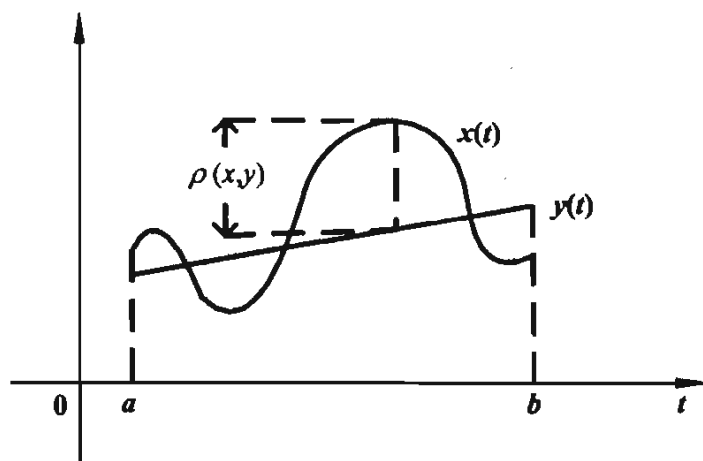


图 1.14 连续函数空间  $C[a, b]$  中两点的距离

定义: 设  $M$  为一非空集。假如对于  $M$  中任意的元素  $x$ 、 $y$  给定一个实数  $\rho(x, y)$  与之对应, 且满足以下条件:

$$\textcircled{1} \quad \rho(x, y) \geq 0, \text{ 而且 } \rho(x, y) = 0 \text{ 的充要条件是 } x = y \quad (1.75a)$$

$$\textcircled{2} \quad \text{三角不等式 } \rho(x, y) \leq \rho(x, z) + \rho(y, z) \quad (z \in M) \quad (1.75b)$$

则称  $\rho(x, y)$  为点  $x$  与  $y$  之间的距离, 而称  $M$  按距离  $\rho(x, y)$  成为距离空间, 记为  $(M, \rho)$ 。

这样定义的距离空间具有广泛的意义, 在各种信号空间中具有广泛的应用。

应当指出, 由式 (1.75a) 和式 (1.75b) 容易证明, 距离有对称性, 即对任意的  $x, y \in M$ , 有

$$\rho(x, y) = \rho(y, x) \quad (1.75c)$$

事实上, 在式 (1.75b) 中取  $z = x$ , 就有  $\rho(x, y) \leq \rho(x, x) + \rho(y, x)$ , 由式 (1.75b) 可知

$$\rho(x, y) \leq \rho(y, x)$$

再由式 (1.75c) 中  $x$ 、 $y$  可互换, 得到  $\rho(y, x) \leq \rho(x, y)$ , 这两式结合起来, 必有

式 (1.75a)。

应用 1.1.6 节所讲的泛函的概念, 距离  $\rho(x, y)$  可以看作序偶  $\{x, y\}$  的泛函, 即  $\rho: \{x, y\} \rightarrow \mathbf{R}$ 。或看作元素序偶  $\{x, y\}$  形成的集合  $M^2$  向实数集合的映射, 即  $\rho: M^2 \rightarrow \mathbf{R}$ 。

在举例说明信号理论中常用的一些距离空间之前, 还需强调如下两个重要概念。

设  $A$  是距离空间  $(M, \rho)$  的子集。对  $A$  中的任意两点, 用它们在  $M$  中的距离作为在  $A$  中的距离。这样,  $A$  也成为距离空间  $(A, \rho)$ , 称  $(A, \rho)$  为  $(M, \rho)$  的子空间。

若在一个集合  $M$  中同时定义了两种距离  $\rho_1(x, y)$  及  $\rho_2(x, y)$ , 且  $\rho_1(x, y) \neq \rho_2(x, y)$ , 那么  $M$  按  $\rho_1(x, y)$  形成的距离空间  $(M, \rho_1)$  与  $M$  按  $\rho_2(x, y)$  形成的距离空间  $(M, \rho_2)$ , 应当看作不同的距离空间。

**例 1.8** 对于任意一个集合  $M$ , 可以定义其元素间的距离为

$$\rho(x, y) = \begin{cases} 0, & (x = y) \\ 1, & (x \neq y) \end{cases} \quad (1.76)$$

得到一个距离空间  $(M, \rho)$ 。显然式 (1.76) 所定义的距离满足距离的两个公理。这个距离空间的特点是, 距离只取 0 和 1 两个值, 因而叫离散距离空间。由于任何一个集带上式 (1.76) 所定义的距离都成为一个距离空间, 因而有着广泛的用途。

**例 1.9** 实轴上的点集, 即实数集合  $\mathbf{R}$ , 通常定义距离

$$\rho(x, y) = |x - y|; \quad x, y \in \mathbf{R}$$

从而形成距离空间  $(\mathbf{R}, \rho)$ 。显然, 上式定义的距离满足距离的两个定理, 它就是通常我们所说的直线上两点之间的距离。这个例子之所以重要, 是因为很多距离空间的距离, 可以看作这个距离的推广。

**例 1.10**  $n$  个实数形成的序列 (即  $n$  个采样值形成的离散信号) 集合  $R^n$  可以以多种方式形成距离空间。如果  $x = \{\alpha_1, \alpha_2, \dots, \alpha_n\}$  和  $y = \{\beta_1, \beta_2, \dots, \beta_n\}$ , 则  $n$  维空间  $R^n$  作为距离空间可以带有下列几种距离:

$$\rho_1(x, y) = \sum_{i=1}^n |\alpha_i - \beta_i| \quad (1.77a)$$

$$\rho_2(x, y) = \left[ \sum_{i=1}^n |\alpha_i - \beta_i|^2 \right]^{\frac{1}{2}} \quad (1.77b)$$

$$\rho_3(x, y) = \max\{|\alpha_i - \beta_i|; i = 1, 2, \dots, n\} \quad (1.77c)$$

式 (1.77) 所定义的  $n$  维空间中的三种距离, 都可以看作是例 1.9 所定义的直线上的距离的直接推广。而其中只有式 (1.77b) 所定义的距离与三维空间中的通常距离概念一致, 我们称之为欧几里得距离。带有欧几里得距离的  $n$  维空间成为欧几里得空间。

应当指出, 上述  $n$  维空间  $R^n$  的几种距离定义式 (1.77), 都可以直接应用到  $n$  维复空间, 即由  $n$  个复数形成的序列作为元素的空间  $C^n$ 。此时,  $\alpha_i$ 、 $\beta_i$  都是复数, 因而式 (1.77) 中的求绝对值应改为求复数的模。

式 (1.77) 定义的距离, 同样可以推广到无限长序列形成的空间, 即为  $R^\infty$  和  $C^\infty$  空间提供距离定义。这正是时域离散信号, 包括实序列和复序列的情况。在这种情况下, 式 (1.77c) 定义的距离要略做修改, 即不是求集合  $\{|\alpha_i - \beta_i|; i = 1, 2, \dots\}$  的最大值, 而是求其最小上限, 记为

$$\rho_3(x, y) = \sup\{|\alpha_i - \beta_i|; i = 1, 2, \dots\}$$

**例 1.11** 在给定区间  $T = \{t; a \leq t \leq b\}$  上定义的复时间函数的任意集合, 可以带有与式 (1.77) 相类似的距离:

$$\rho_1(x, y) = \int_T |x(t) - y(t)| dt \quad (1.78a)$$

$$\rho_2(x, y) = \left[ \int_T |x(t) - y(t)|^2 dt \right]^{\frac{1}{2}} \quad (1.78b)$$

$$\rho_3(x, y) = \sup\{|x(t) - y(t)|; t \in T\} \quad (1.78c)$$

从而形成距离空间, 带有距离式 (1.78c) 的时间函数空间显然是前面所述连续函数空间  $C[a, b]$  的推广。为了说明式 (1.78c) 是一个合理的距离, 只需证明式 (1.78c) 满足距离的两个公理。显然, 当且仅当  $x(t) \equiv y(t)$  时, 有  $\rho_3(x, y) = \sup\{|x(t) - y(t)|\} = 0$ , 因而它满足式 (1.78a)。又因为

$$\begin{aligned} \rho_3(x, y) &= \sup\{|x(t) - y(t)|\} = |x(t_0) - y(t_0)| \\ &= |x(t_0) - z(t_0) + z(t_0) - y(t_0)| \\ &\leq |x(t_0) - z(t_0)| + |z(t_0) - y(t_0)| \\ &\leq \sup\{|x(t) - z(t)|\} + \sup\{|z(t) - y(t)|\} \\ &= \rho_3(x, z) + \rho_3(y, z) \end{aligned}$$

因而它亦满足式 (1.75b)。

式 (1.78a) 和式 (1.78b) 定义的距离对于函数空间有一个明显的困难, 就是当  $x(t)$  和  $y(t)$  在  $T$  区间内有可数个相异点, 仍有  $\rho(x, y) = 0$ 。这就违反了距离公理式 (1.75a) 中的要求: 仅当  $x(t) = y(t)$  时  $\rho(x, y) = 0$ 。为了克服这一困难, 今后我们认为, 如果  $x(t)$ 、 $y(t)$  在区间  $T$  内仅有可数个相异点, 即如果  $x(t)$ 、 $y(t)$  在区间  $T$  内“几乎处处相等”, 我们就把它们看作是距离空间中的同一个点, 即认为  $x(t) = y(t)$ 。这一做法正好与 1.1 节中把“几乎处处相等”的信号看作一个等价集, 从而使傅里叶变换具有一对一映射性质相一致。正因为如此, 这两种定义的距离在信号理论中广泛采用。由于式 (1.78b) 所定义的平方还可表示  $x$  与  $y$  的差信号的能量, 具有明显的物理意义, 因而式 (1.78b) 所定义的  $\rho_2$  应用最为广泛。

**例 1.12** 二进制码组广泛地应用于通信和计算机领域。设  $A = \{0, 1\}$ , 集合  $A$  的元素 (亦称码元) 只有 0 和 1 两个, 故叫二进制码。 $n$  个码元形成一个码组, 用  $W = A^n$  表示码组的集合, 用  $w$  表示码组的元素。例如,  $A^3 = \{000, 001, \dots, 111\}$  表示  $A^3$  集合中的元素有 8 个不同的码组。几个码元组成的码组集合  $A^n$  有  $2^n$  个元素。我们可以把长度为  $n$  的码组看作是  $n$  个实数形成的序列的特例, 此时, 这些实数只能取 0 或 1。

为了便于区分这些码组, 一个最好的办法是定义码组间的距离, 从而形成距离空间。其中一个较好的距离定义是例 1.10 中的式 (1.77a), 即

$$\rho(x, y) = \sum_{i=1}^n |a_i - b_i|$$

式中,  $x = \{a_1, a_2, \dots, a_n\}$ ,  $y = \{b_1, b_2, \dots, b_n\}$ , 此时  $a_i$ 、 $b_i$  只能取 0 或 1。在二进制码组形成的距离空间中, 这个距离定义等效于

$$\rho(x, y) = \sum_{i=1}^n [(a_i + b_i) \text{ 模 } 2] \quad (1.79)$$

式 (1.79) 定义的距离, 称为海明距离。其物理意义是, 为了量度长度为  $n$  的两个二进制码组的距离, 只需对同码位的码元进行模 2 相加, 然后按码位把这些结果进行算术相加。显然, 只有两码组完全相同, 其距离才为 0, 而不同码组间的最小距离为 1。

海明距离的概念已经广泛应用于检错码和纠错码。为了能发现码组在传输中出现的错误, 必须增加码组间的最小距离。如果使码组间的最小距离为 2, 就可以发现 1 位的错误。表 1.1 给出了一种检错码的例子。大家知道, 长度为 4 的码组集合有 16 种不同的元素, 我们可以在其中选出 8 个元素, 使它们间的最小距离为 2。



表 1.1 具有最小距离的码组集合

码组元素	信息码位			奇偶校验码位 $a_4$
	$a_1$	$a_2$	$a_3$	
$w_1$	0	0	0	0
$w_2$	0	0	1	1
$w_3$	0	1	0	1
$w_4$	0	1	1	0
$w_5$	1	0	0	1
$w_6$	1	0	1	0
$w_7$	1	1	0	0
$w_8$	1	1	1	1

这些经过挑选的 4 位码组可以看作是 3 位信息码加上 1 位校验码形成的，如表 1.1 所示。校验码形成的规则是

$$a_4 = (a_1 + a_2 + a_3) \text{模} 2 \quad (1.80)$$

显然， $(a_1 + a_2 + a_3 + a_4) \text{模} 2 = 0$ ，也就是说，每个码组具有的 1 为双数。如果码组中 1 位发生错码，则  $(a_1 + a_2 + a_3 + a_4) \text{模} 2 = 1$ 。因此，只要把码组各位相加后求模 2，则可以判定码组在传输过程中某一位发生了错误。

如果在信息码位后再增加一些校验码位，我们可以得到码组元素间最小距离为 3 的码组集合。显然，这时的码组不仅具有检测出 1 位错误的能力，还有纠正 1 位错误的能力，这就叫纠错码。因为这时如果传输的码组发生 1 位错误，它与原码组的距离为 1，与其他所有可能传输的码组的距离均大于或等于 2，因而可恢复原码组，从而实现纠错。表 1.2 给出了一种纠错码的例子，它具有 4 位信息码和 3 位校验码。校验码产生的规则如式 (1.81) 所述。

$$\left. \begin{aligned} a_5 &= (a_1 + a_2 + a_3) \text{模} 2 \\ a_6 &= (a_1 + a_3 + a_4) \text{模} 2 \\ a_7 &= (a_2 + a_3 + a_4) \text{模} 2 \end{aligned} \right\} \quad (1.81)$$

表 1.2 最小距离为 3 的码组集合

码组元素	信 息 码 位				奇偶校验码位		
	$a_1$	$a_2$	$a_3$	$a_4$	$a_5$	$a_6$	$a_7$
$w_1$	0	0	0	0	0	0	0
$w_2$	0	0	0	1	1	1	1
$w_3$	0	0	1	0	0	1	1
$w_4$	0	0	1	1	1	0	0
$w_5$	0	1	0	0	1	0	1
$\vdots$							
$w_{16}$	1	1	1	1	1	1	1

## 1.2.2 距离空间的特性

### (1) 距离空间的收敛性

1.2.1 节已经说明, 对一个集合  $M$  内的元素间引入距离, 使这个集合具有一定几何特性, 从而形成距离空间。这样, 欧几里得空间中的很多概念可以引入到距离空间中来。此时, 集合  $M$  内的元素称为距离空间  $(M, \rho)$  的点。

距离空间  $M$  中, 满足条件  $\rho(x, x_0) < r$  的所有点的总体, 称为空间  $M$  中以  $x_0$  为球心的开球, 记为  $S(x_0, r)$ ,  $x_0$  为球心,  $r$  为半径。同样, 满足条件  $\rho(x, x_0) \leq r$  的点  $x \in M$  的总体, 称为以  $x_0$  为球心的闭球, 记为  $S[x_0, r]$ 。以  $x_0$  为球心, 以  $\varepsilon$  为半径的开球亦称为点  $x_0$  的  $\varepsilon$ -邻域, 记为  $O_\varepsilon(x_0)$ 。

如果点  $x$  的任何邻域至少包含集  $M$  的一个点, 则称  $x$  为  $M$  的接触点。集  $M$  的一切接触点的总体, 称为  $M$  的闭包, 用  $[M]$  表示。显然, 属于  $M$  的任一点必为  $M$  的接触点, 因此, 任一点必被它本身的每一个邻域所包含。因而有  $M \subseteq [M]$  和  $[M] = [M]$ 。

如果  $x$  的任何邻域总包含集  $M$  的无限多个点, 就称  $x$  是集  $M$  的极限点。极限点  $x$  可以属于  $M$ , 也可以不属于  $M$ 。如果点  $x$  属于  $M$ , 同时, 存在  $x$  的某一邻域  $O_\varepsilon(x)$ , 它不包括含  $M$  中与  $x$  不同的点, 则称  $x$  为  $M$  的孤立点。可以证明, 集  $M$  的任一接触点, 或者是  $M$  的极限点, 或者是  $M$  的孤立点。

有了上述概念以后, 我们可以引入点列的收敛和极限等一系列距离空间的性质。设距离空间  $(M, \rho)$  中有一个无限点列  $\{x_n; x_n \in M, n=1, 2, \dots\}$ , 如果存在一个  $x$ , 使  $x$  的每一个邻域  $O_\varepsilon(x)$  总包含着点列中某个  $n_0$  开始的一切点  $x_n$ , 从数学上来讲, 对于任意的  $\varepsilon > 0$ , 如果存在一个正整数  $n_0$ , 使得

$$n \geq n_0 \Rightarrow \rho(x_n, x) < \varepsilon$$

则称点列  $\{x_n\}$  收敛于  $x$ ，或称  $\{x_n\}$  的极限为  $x$ ，记为

$$\lim_{n \rightarrow \infty} x_n = x \quad (1.82)$$

建立了距离空间的另一重要结果是，可以把通常的函数的连续概念推广到一个距离空间到另一个距离空间的映射。

设  $A = (X, \rho_1), B = (Y, \rho_2)$  为两个距离空间， $f: (X, \rho_1) \rightarrow (Y, \rho_2)$ 。如果对于任意的  $\varepsilon > 0$ ，总存在一个  $\delta > 0$ ，使得

$$\left. \begin{aligned} \rho_1(x, x_0) < \delta \Rightarrow \rho_2(y, y_0) < \varepsilon, \quad x \in X, y \in Y \\ y = f(x), \quad y_0 = f(x_0) \end{aligned} \right\} \quad (1.83)$$

则称映射  $f$  在点  $x_0$  连续。如果映射  $f$  在空间  $A$  的一切点连续，则称  $f$  为连续映射。

**例 1.13** 为了说明连续映射的概念，下面研究由实函数空间向实数集  $\mathbb{R}$  的映射，即泛函。对于实函数空间，我们采用的距离定义为

$$\rho_1(x, y) = \left[ \int_T |x(t) - y(t)|^2 dt \right]^{\frac{1}{2}}$$

$\mathbb{R}$  中的距离定义为  $\rho_2(a, b) = |a - b|$ 。映射定义为

$$f_\varphi(x) = \int_{-\infty}^{\infty} x(t) \varphi(t) dt \quad (1.84)$$

对于任意  $x_0$ ，我们有

$$\rho_2(f_\varphi(x), f_\varphi(x_0)) = |f_\varphi(x) - f_\varphi(x_0)| = \left| \int_{-\infty}^{\infty} [x(t) - x_0(t)] \varphi(t) dt \right| \quad (1.85)$$

应用许瓦兹不等式于式 (1.85)，则

$$\begin{aligned} \rho_2(f_\varphi(x), f_\varphi(x_0)) &= \left| \int_{-\infty}^{\infty} [x(t) - x_0(t)] \varphi(t) dt \right| \\ &\leq \left[ \int_{-\infty}^{\infty} [x(t) - x_0(t)]^2 dt \right]^{\frac{1}{2}} \left[ \int_{-\infty}^{\infty} \varphi^2(t) dt \right]^{\frac{1}{2}} \end{aligned} \quad (1.86)$$

如果  $\varphi(t)$  为平方可积函数，即

$$\left[ \int_{-\infty}^{\infty} \varphi^2(t) dt \right]^{\frac{1}{2}} < K$$

若式中  $K$  为一个正实数，则

$$\rho_1(x, x_0) < \delta = \frac{\varepsilon}{K} \Rightarrow \rho_2(f_\varphi(x), f_\varphi(x_0)) < K \rho_1(x, x_0) < \varepsilon \quad (1.87)$$

因而  $f_\varphi$  是一个连续泛函。

连续映射的下面两种特殊情况值得注意：第一，如果映射  $f: A \rightarrow B$  是一一映射，且双向连续，即  $f$  和  $f^{-1}$  均连续，则称  $f$  是同胚映射，称空间  $A$ 、 $B$  拓扑同构；第二，设  $(A, \rho_1)$  及  $(B, \rho_2)$  是两个距离空间， $f: A \rightarrow B$  是一一映射，如果对于任意  $x, y \in A$  有下式成立

$$\rho_1(x, y) = \rho_2(f(x), f(y))$$

则称  $f$  为  $(A, \rho_1)$  到  $(B, \rho_2)$  上的等距映射，而称这两个空间为等距同构。如果我们仅讨论空间与距离有关的属性，如收敛性、完备性等，则可以把等距同构的空间看成完全一致的。同时还应指出，等距同构是拓扑同构的特例。

## (2) 距离空间的完备性

下面讨论距离空间的另一个特性，即完备性问题。首先需要定义距离空间中的柯西点列。如果距离空间  $A$  的点列  $\{x_n\}$  满足柯西准则，即如果对于任意  $\varepsilon > 0$ ，总存在这样一个正整数  $n_0$ ，使得对一切  $m, n \geq n_0$ ，都有  $\rho(x_m, x_n) < \varepsilon$ ，则称  $\{x_n\}$  为  $A$  中的本来收敛点列或柯西点列。直观地讲，柯西点列就是当  $n$  增加时，点列  $\{x_n\}$  的点互相无限靠近的序列。

显然，距离空间中任何收敛点列一定是柯西点列。这是因为，如果  $\{x_n\}$  收敛于  $x$ ，那么，对于给定的  $\varepsilon > 0$ ，总存在一个正整数  $n_0$ ，使  $n \geq n_0$  时有  $\rho(x_n, x) < \frac{\varepsilon}{2}$ ，因而有  $m, n \geq n_0$  时

$$\rho(x_m, x_n) \leq \rho(x_m, x) + \rho(x_n, x) < \varepsilon$$

即  $\{x_n\}$  为柯西点列。

但是，柯西点列不一定是收敛点列，因为柯西点列的极限点  $x$ ，可能不属于所讨论的距离空间。例如，有理数集合可以形成一个距离空间，本来收敛的有理数序列在有理数域内就不一定有极限。

如果距离空间  $A$  中的任何柯西点列，都收敛于  $A$  中的一个点，则称距离空间  $A$  是完备的。显然， $\mathbf{R}$  和  $\mathbf{R}^n$  空间是完备的。有理数空间是不完备的。下面，我们将给出几个在信号理论中常用的距离空间的例子，并考察其完备性。

**例 1.14**  $C[T]$  表示定义在  $T = \{t; a \leq t \leq b\}$  上的连续实函数集。如果我们采 (1.78b) 所定义的距离  $\rho_2^2(x, y) = \int_a^b |x(t) - y(t)|^2 dt$ ，则形成一个距离空间  $(C[T], \rho_2)$ 。我们将证明  $(C[T], \rho_2)$  是不完备的。为了证明这一点，只需找一个

不收敛的柯西点列即可。图 1.15 给出了属于这个距离空间的序列  $\{x_n(t)\}$ ，其数学表达式为

$$x(t) = \begin{cases} -1, & t < -\frac{1}{n} \\ nt, & |t| \leq \frac{1}{n} \\ +1, & t > \frac{1}{n} \end{cases}$$

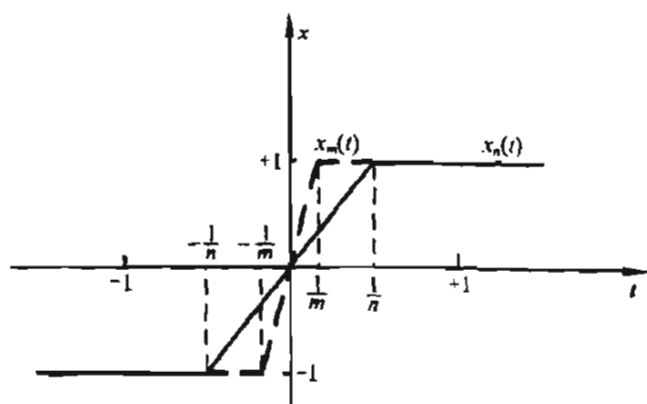


图 1.15 连续函数空间的柯西点列

令  $a = -1$ ,  $b = +1$ , 则当  $m > n$  时有

$$\rho_2(x_m, x_n) = \left[ \int_{-1}^1 |x_m(t) - x_n(t)|^2 dt \right]^{\frac{1}{2}} = \sqrt{\frac{2}{3n} \left( 1 - \frac{n}{m} \right)}$$

显然, 当  $n \rightarrow \infty$  时,  $\rho_2(x_m, x_n) \rightarrow 0$ , 因而  $\{x_n(t)\}$  为柯西点列。但  $\{x_n(t)\}$  当  $n \rightarrow \infty$  时的极限为  $\text{sgn}(t) = t/|t|$ , 是不连续函数, 不属于连续函数集  $C[T]$ 。因而  $(C[T], \rho_2)$  是不完备的距离空间。

例 1.15 上例中连续函数集  $C[T]$ , 如果采用式 (1.78c) 所定义的距离  $\rho_3(x, y) = \sup\{|x(t) - y(t)|, t \in T\}$ , 则形成距离空间  $(C[T], \rho_3)$ 。下面将证明这个空间是完备的。

设  $\{x_n(t); n = 1, 2, 3, \dots\}$  为  $C[T]$  任一柯西点列, 即对于任意  $\varepsilon > 0$ , 只要  $n, m$  足够大, 则有

$$\rho_3(x_m, x_n) = \sup\{|x_m(t) - x_n(t)|; t \in T\} < \varepsilon$$

对给定的  $t$ ,  $x_n(t)$  可以看作实数空间  $\mathbb{R}$  中的柯西序列, 它收敛于  $x(t)$ , 即对于足够大的  $n$  有

$$|x_n(t) - x(t)| < \frac{\varepsilon}{3}$$

下面只需证明  $x(t)$  是一个连续函数, 即证明  $x(t) \in C[T]$ , 即证明了  $(C[T], \rho_3)$  的完备性。也就是说, 只需证明对于任何  $\varepsilon > 0$  和  $t_0 \in T$ , 我们可以找到  $\delta > 0$ , 使得当  $|t - t_0| < \delta$  时, 有

$$|x(t) - x(t_0)| < \varepsilon \quad (1.88)$$

因为

$$x_n(t) \text{ 连续} \Rightarrow |x_n(t) - x_n(t_0)| < \frac{\varepsilon}{3}$$

因而

$$\begin{aligned} |x(t) - x(t_0)| &= |[x(t) - x_n(t)] + [x_n(t) - x_n(t_0)] + [x_n(t_0) - x(t_0)]| \\ &\leq |x(t) - x_n(t)| + |x_n(t) - x_n(t_0)| + |x_n(t_0) - x(t_0)| \\ &< \frac{\varepsilon}{3} + \frac{\varepsilon}{3} + \frac{\varepsilon}{3} = \varepsilon \end{aligned}$$

所以,  $x(t)$  为连续函数, 因而  $(C[T], \rho_3)$  空间是完备的距离空间。例 1.14 和例 1.15 说明, 距离空间的完备性与所定义的距离有关。

**例 1.16**  $L^p[a, b]$  ( $p \geq 1$ ) 表示在  $[a, b]$  上  $p$  方可积函数  $x(t)$  全体所组成的空间, 即对于  $x \in L^p$ , 有

$$\int_a^b |x(t)|^p dt < \infty \quad (1.89)$$

$L^p$  中元素  $x(t)$ 、 $y(t)$  的距离定义如下:

$$\rho(x, y) = \int_a^b |x(t) - y(t)|^p dt \quad (1.90)$$

任何一本泛函教程都有证明, 式 (1.90) 定义的距离满足距离公理 (1.75) 式要求。亦可证明,  $L^p[a, b]$  是一个完备的距离空间。这表明  $L^p$  空间中的柯西点列收敛于  $p$  方可积函数。

这里需要强调指出, 当  $p=2$  时,  $L^p[a, b]$  变为  $L^2[a, b]$ , 即平方可积函数空间, 这正是信号理论中常用的信号空间, 它的元素为平方可积信号, 亦即能量有限的信号。通常  $L^2[a, b]$  空间采用式 (1.78b) 所定义的距离, 这正是式 (1.90) 中  $p=2$  的特殊情况。因而  $L^2[a, b]$  也是一个完备的距离空间。

**例 1.17** 类似于  $L^p$  空间, 可引入  $l^p$  ( $p \geq 1$ ) 空间, 它是由所有满足条件

$$\sum_{k=1}^{\infty} |\varepsilon_k|^p < \infty \quad (1.91)$$

的实数序列  $\{\xi_1, \xi_2, \dots\}$  的全体所组成。对于  $l^p$  中任意两个元素  $x = \{\xi_1, \xi_2, \dots\}$  和  $y = \{\eta_1, \eta_2, \dots\}$ , 引入距离

$$\rho(x, y) = \left( \sum_{k=1}^n |\xi_k - \eta_k|^p \right)^{\frac{1}{p}} \quad (1.92)$$

与例 1.16 一样, 可以验证式 (1.92) 是一个合理的距离, 亦可证明  $l^p$  是一个完备的距离空间。

当  $p=2$  时,  $l^p$  变为  $l^2$  空间, 它在信号理论中有着广泛的应用。 $l^2$  空间的每一个元素, 是一个平方可和的序列, 即能量有限的时域离散信号。因为  $l^2$  为  $l^p$  的特例, 因而  $l^2$  空间也是一个完备的距离空间。

### (3) 距离空间的可分性

设  $A, B$  是距离空间  $M$  里的两个子集。如果  $[A] \supseteq B$ , 则称集  $A$  在集  $B$  里稠密。如果  $[A] = M$ , 则称集  $A$  在集  $M$  里处处稠密。例如, 一切有理数的集合在实轴上处处稠密。

距离空间  $M$  叫做可分的, 是指在  $M$  中存在一个处处稠密的可数子集。也就是说, 如果对于任意  $\varepsilon > 0$ , 总可以找到可数个元素  $\{x_1, x_2, \dots\}$ , 使得对任意  $x \in M$  和某一个  $i$ , 有  $\rho(x, x_i) < \varepsilon$ , 则称  $M$  是可分的。形象地说来, 可分空间就是可以用可数个半径为  $\varepsilon$  的球覆盖的距离空间。

欧氏空间  $R^n$  是可分的, 因为坐标为有理数的点的全体构成它的一个可数稠密子集。 $C[T]$  和  $L^p[a, b]$  是可分的。对于后者, 容易证明具有有理数系数的多项式全体组成  $L^p[a, b]$  中的可数稠密子集。由此可知, 信号分析中常用的  $L^2[a, b]$  也是可分的。

### (4) 距离空间的列紧性

下面简单介绍距离空间内集合的列紧性或致密性。距离空间  $M$  中的集合  $A$  叫列紧的, 是指  $A$  中任一无限子集都含有一个收敛子点列。如果  $A$  中所有收敛点列的极限都属于  $A$ , 则称  $A$  是自列紧的。如果空间  $M$  是自列紧的, 则叫做列紧空间。显然, 列紧空间是完备的, 但完备空间不一定是列紧的。

如果距离空间  $M$  里存在一个包含  $A$  的闭球, 则称集合  $A$  称是有界的。如果设  $X$  是距离空间  $M$  中的点集, 且  $\varepsilon > 0$ , 如果存在  $M$  中的另一点集  $A$ , 使得对  $X$  中任一元素  $x$ , 存在元素  $a \in A$ , 满足  $\rho(x, a) < \varepsilon$ , 则称集合  $A$  是  $X$  的  $\varepsilon$ -网。例如,  $R^2$  中坐标为整数的一切点就是平面上的一个  $\sqrt{2}/2$ -网。如果对于任何  $\varepsilon > 0$ , 总有关于集合  $X$  的有穷  $\varepsilon$ -网, 则称集合  $M$  全有界。显然, 全有界集必有界, 全有界的距离空间必可分。

列紧性或致密性和全有界性之间有密切的关系。在完备距离空间中, 集  $A$  为列紧的充分必要条件是集  $A$  全有界, 因此, 列紧空间或致密空间就是可以用有限个半径为  $\varepsilon$  的球覆盖的距离空间。例如,  $R^n$  空间是完备的, 可分的, 但不是全有界的, 故它本身不是列紧空间。但  $R^n$  空间任一有界闭子集是列紧的, 且  $R^n$  中任一闭子集是完全有界的。在信号理论中, 致密空间具有重要意义,

因为这种空间可用有限维空间以任意精度逼近, 从而可用  $n$  维数组近似表示其中的元素, 大大便利了信号的分析与处理。

### 1.2.3 线性空间

#### (1) 线性空间的定义

1.2.1 和 1.2.2 节中, 我们对讨论的信号集定义了距离, 从而使信号集具有一定几何特性, 形成距离空间。事实上, 常用的信号集还具有一定的代数特性, 这正是本节所要讨论的问题。也就是说, 如果在信号集上附加上一定的代数结构, 就形成了线性空间。

线性空间又称线性向量空间, 其元素亦称为向量。今后, 用  $\bar{x}$ 、 $\bar{y}$ 、 $\bar{z}$  等表示线性空间中的向量。对于  $n$  维向量空间,  $\bar{x}$  表示  $n$  维列向量:

$$\bar{x} = [x_1 \ x_2 \ \cdots \ x_n]^T$$

或  $n$  维数组  $[x_1 \ x_2 \ \cdots \ x_n]$ 。对于函数空间,  $\bar{x}$  表示时间函数:

$$\bar{x} = \begin{cases} x(t), & 0 \leq t \leq T \\ 0, & \text{其他} \end{cases}$$

线性空间中的零向量用  $\bar{0}$  表示, 对于  $n$  维向量空间,  $\bar{0}$  表示所有分量为 0 的向量, 即

$$\bar{0} = [0, 0, \cdots, 0]^T$$

对于函数空间,  $\bar{0}$  表示  $t$  为任何值时  $x(t) = 0$ , 即

$$\bar{0} = 0, \quad -\infty < t < +\infty$$

在线性空间理论中, 还会遇到另一类型的量, 即采用一个域  $K$  的标量, 通常用小字母  $a, b, c, \cdots$  或  $\alpha, \beta, \gamma, \cdots$  表示。标量通常用来表示次序和大小, 也用来作为向量的系数, 以表示向量的尺寸。例如, 可以说  $2\bar{x}$  的尺寸是  $\bar{x}$  的 2 倍。

线性空间在数学上定义为具有如下特性的元素集合  $E$ :

①  $E$  是一加法群。在  $E$  内定义了一种算法, 叫做加法, 即对于任何  $\bar{x}$ 、 $\bar{y}$  确定了一元素  $\bar{x} + \bar{y} \in E$ , 且满足

(交换律);

$(\bar{y} + \bar{z})$  (结合律);

元素 (即零向量)  $\bar{0}$ , 对任何  $\bar{x} \in E$ , 有  $\bar{0} + \bar{x} = \bar{x}$ 。 $\bar{0}$  又

是, 存在唯一的向量  $-\bar{x}$  使  $\bar{x} + (-\bar{x}) = \bar{0}$ 。 $-\bar{x}$  称为  $\bar{x}$  的向量加



法逆元素。

② 定义了数域  $K$  中的数  $\alpha$  与  $\bar{x} \in E$  的数乘 (或称标乘), 即  $\alpha\bar{x} \in E$ , 且满足

- ⑤.  $\alpha(\beta\bar{x}) = (\alpha\beta)\bar{x}$  (标乘有结合律);
  - ⑥.  $1\bar{x} = \bar{x}$  (1 为数域  $K$  的乘法恒等元);  
 $0\bar{x} = \bar{0}$  (0 为数域  $K$  的加法恒等元);
  - ⑦.  $\alpha(\bar{x} + \bar{y}) = \alpha\bar{x} + \alpha\bar{y}$
  - ⑧.  $(\alpha + \beta)\bar{x} = \alpha\bar{x} + \beta\bar{y}$
- (标乘分配律)。

应当说明, 线性向量空间前四条定义, 实质上就是近世代数中在  $(+)$  运算下的一个交换群的公理。线性向量空间的后四条定义, 引入了标乘运算及其性质。众所周知, 标量域通常应定义两种运算, 即加法与乘法, 但其加法恒等元 0 没有乘逆。因此, 请注意区分标量域的加法恒等元和乘法恒等元, 亦应注意区分标量域的加法恒等元 0 与向量的加法恒等元  $\bar{0}$ 。

在线性空间的后四条定义中, 并未指明标量域  $K$  的具体性质。 $K$  可以是实数集合  $\mathbf{R}$ , 也可以是复数集合  $\mathbf{C}$ 。在信号理论中, 特别是在编码理论中, 常常用到的标量域  $K$  是标量的有限域。例如, 二进制集  $\{0, 1\}$  可以形成一个有限域, 其元素仅有 0 与 1 两个。在这个域上定义的运算可以是通常的模 2 运算, 从而形成一个合理标量域。这在通信系统和计算机系统中有着广泛的应用。

如果所使用的标量域为实数域, 则形成的线性空间称为实线性空间。如果使用复数作为标量, 我们就得到一个复线性空间。

设  $A$  是线性空间  $E$  的一个子集。若  $A$  中任何两个元素  $\bar{x}$ 、 $\bar{y}$  的和  $\bar{x} + \bar{y}$  属于  $A$ , 任意数集  $\alpha\bar{x}$  也属于  $A$ , 则称  $A$  为  $E$  的一个线性子空间, 简称子空间。显然, 如  $A$ 、 $B$  是  $E$  的子空间, 则  $A$ 、 $B$  之交集  $C = A \cap B$  及  $A$ 、 $B$  之并集  $D = A \cup B$  均为  $E$  的子空间。

如果线性空间  $E$  中的向量  $\bar{x}$ , 可以表示为  $E$  中的  $n$  个特殊向量  $\{\bar{x}_i\}$  的标乘之和

$$\bar{x} = \sum_{i=1}^n \alpha_i \bar{x}_i \quad (1.93)$$

则称  $\bar{x}$  为  $\{\bar{x}_i\}$  的线性组合。显然, 所有  $\{\bar{x}_1, \bar{x}_2, \dots, \bar{x}_n\}$  的线性组合的集合形成一个线性空间。而且, 如果我们取  $\{\bar{x}_1, \bar{x}_2, \dots, \bar{x}_n\}$  的一个子集, 例如  $\{\bar{x}_1, \bar{x}_2, \dots, \bar{x}_m\}$ ,  $m < n$ , 则  $\{\bar{x}_1, \bar{x}_2, \dots, \bar{x}_m\}$  的线性组合的集合也形成一个线性空间, 它是  $\{\bar{x}_1, \bar{x}_2, \dots, \bar{x}_n\}$  形成的线性空间的子空间。

在线性代数中, 已讨论过向量空间中一组向量是否线性无关的问题。如果

对于一个向量集  $\{\bar{x}_i; i=1, 2, \dots, n\}$ , 仅当所有  $\alpha_i = 0$  时, 才有

$$\sum_{i=1}^n \alpha_i \bar{x}_i = \bar{0} \quad (1.94)$$

则称该向量集线性无关。也就是说, 线性无关向量集中任一向量, 不能表示为其他向量的线性组合。

如果  $M$  是  $n$  个线性无关向量  $\{\bar{x}_i\}$  的线性组合形成的空间, 则  $M$  中每一个向量  $\bar{x}$  是  $\{\bar{x}_i\}$  的唯一的线性组合, 即  $\bar{x}$  由唯一的一组标量  $\{a_i\}$  决定, 因而称  $M$  为  $n$  维线性空间。 $\{\bar{x}_i\}$  称为  $M$  的基集,  $\{a_i\}$  为点  $\bar{x}$  关于基集  $\{\bar{x}_i\}$  的坐标, 而  $M$  称为由基集  $\{\bar{x}_i\}$  生成。

下面是信号理论中常用的几个线性空间的例子。

**例 1.18**  $n$  个标量的序列之集合  $R^n$  或  $C^n$ , 形成一个线性空间  $E^n$ , 设  $\bar{x} = \{\alpha_1, \alpha_2, \dots, \alpha_n\}$  和  $\bar{y} = \{\beta_1, \beta_2, \dots, \beta_n\}$ , 则向量的加法定义为

$$\bar{x} + \bar{y} = \{\alpha_1 + \beta_1, \alpha_2 + \beta_2, \dots, \alpha_n + \beta_n\} \quad (1.95)$$

而标乘定义为

$$\alpha \bar{x} = \{\alpha \alpha_1, \alpha \alpha_2, \dots, \alpha \alpha_n\} \quad (1.96)$$

显然,  $\bar{x} + \bar{y} \in E^n$ ,  $\alpha \bar{x} \in E^n$ , 所以  $E^n$  是一个合理的线性空间。

在  $E^n$  空间中, 每一个向量可表示为基集  $\{\bar{e}_i\}$  的线性组合:

$$\bar{x} = \sum_{i=1}^n \alpha_i \bar{e}_i \quad (1.97)$$

式中,  $n$  个线性无关向量  $\{\bar{e}_i\}$  给定为

$$\begin{aligned} \bar{e}_1 &= \{1, 0, 0, \dots, 0\} \\ \bar{e}_2 &= \{0, 1, 0, \dots, 0\} \\ &\vdots \\ \bar{e}_n &= \{0, 0, 0, \dots, 1\} \end{aligned} \quad (1.98)$$

**例 1.19** 前面讨论过  $M$  是由  $n$  个线性无关向量  $\{\bar{x}_i\}$  的线性组合形成的空间, 任意  $\bar{x} \in M$  可唯一地表示为

$$\bar{x} = \sum_{i=1}^n \alpha_i \bar{x}_i$$

因而,  $M$  中的向量  $\bar{x}$  可以与  $n$  个标量的序列  $\{\alpha_i\}$  建立一一对应关系。也就是说, 可以把  $R^n$  或  $C^n$  空间的点的表示法  $\{\alpha_i\}$ , 作为  $M$  中向量  $\bar{x}$  相对于基集  $\{\bar{x}_i\}$  的一种表示法。后面将说明, 任意  $n$  维线性空间  $M$  与  $R^n$  或  $C^n$  空间线性同构。 $n$  维线性空间  $M$  和  $E^n$  空间可形象地用图 1.16 表示。

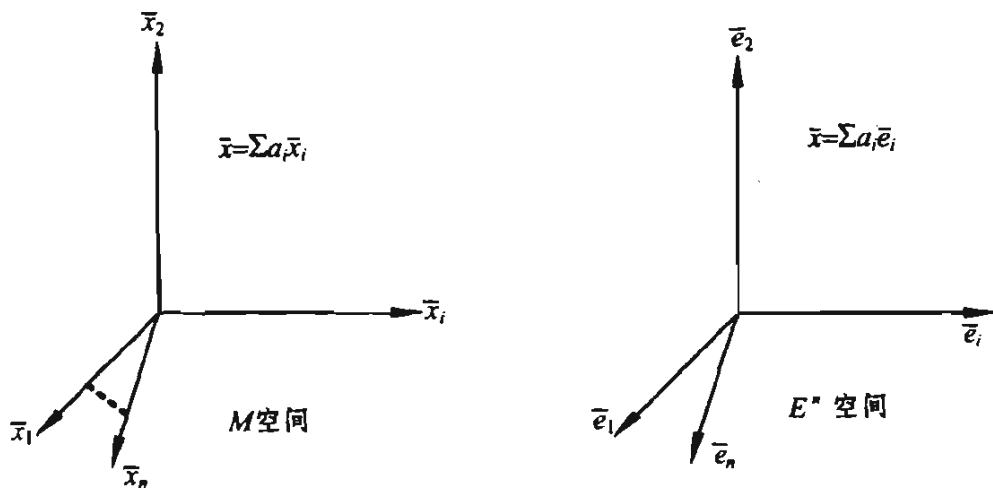


图 1.16  $n$  维线性空间  $M$  与  $E^n$  空间同构关系示意图

**例 1.20** 定义在区间  $T = \{t; a \leq t \leq b\}$  上的实函数或复函数集合形成线性空间, 其中, 向量加运算及标乘运算以如下逐点方式定义:

$$\left. \begin{aligned} \bar{z} = \bar{x} + \bar{y} &\Rightarrow z(t) = x(t) + y(t) \\ \bar{z} = \alpha \bar{x} &\Rightarrow z(t) = \alpha x(t) \end{aligned} \right\} \quad (1.99)$$

$t \in T$

这样定义的线性空间称为函数空间, 记为  $Q[T]$ 。显然, 根据式 (1.99), 如  $\bar{x} \in Q[T]$ , 则有  $\bar{x} + \bar{y} \in Q[T]$ ; 如果  $\bar{x} \in Q[T]$ , 则有  $\alpha \bar{x} \in Q[T]$ , 因而  $Q[T]$  是合理的线性空间。

通常, 函数空间是无限维的, 因为不可能找到有限个线性无关的向量  $\{\bar{x}_i\}$ , 使所有  $\bar{x} \in Q[T]$  都表示为  $\{\bar{x}_i\}$  的线性组合。

## (2) 线性空间的同构映射与同态映射

如果能在两个线性空间  $M_1$  和  $M_2$  之间建立一对一映射  $f: M_1 \rightarrow M_2$ , 而且  $M_1$  的元素  $\bar{x}_1, \bar{y}_1$  分别与  $M_2$  的元素  $\bar{x}_2, \bar{y}_2$  对应时,  $M_1$  的元素  $a\bar{x}_1 + b\bar{y}_1$  ( $a, b$  为标量) 对应于  $M_2$  的元素  $a\bar{x}_2 + b\bar{y}_2$ , 则称映射  $f: M_1 \rightarrow M_2$  为线性同构映射,  $M_1$  与  $M_2$  两空间线性同构。也就是说,  $f: M_1 \rightarrow M_2$  为线性同构映射的条件是一一对应,  $f$  有逆映射, 同时, 映射过程保持原有线性关系。当两个空间线性同构时, 它们的代数性质完全一样, 因而必要时可以视为同一个线性空间而予以区别。

同态映射是  $f: M_1 \rightarrow M_2$ ，并保证

$$f(a\bar{x}_1 + b\bar{y}_1) = af(\bar{x}_1) + bf(\bar{y}_1)$$

成立的映射，但并不要求  $M_1$  与  $M_2$  一一对应，即不一定存在逆映射。显然，同构是同态的特例。

**例 1.21** 在数字信号处理中常用的同态映射（或称同态变换）的例子如图 1.17 所示。设输入空间  $M_1$  的元素为实序列  $\{x(n); n=0,1,2,\dots\}$ ，并且  $x(n)$  表示一个图像信号，因而它只取大于或等于 0 的数值。设  $x_1(n)$  表示图像信号中真实反应信息的部分，而  $x_2(n)$  表示照射不均匀引入的干扰部分，因而有  $x(n) = x_1(n) \cdot x_2(n)$ 。

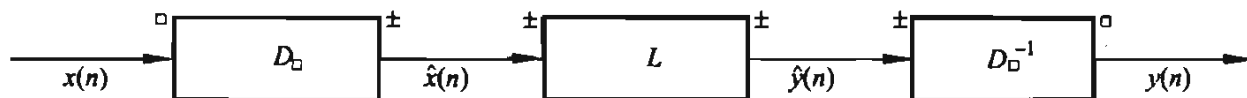


图 1.17 数字信号处理的同态系统

为了减弱或消除  $x_2(n)$  这种乘积型干扰，一个简单的办法是取对数  $\hat{x}(n) = \ln[x(n)] = \ln[x_1(n)] + \ln[x_2(n)]$ ，然后用线性滤波器  $L$  减弱或消除  $\ln x_2(n)$  分量，再用  $D_0^{-1} = \exp[\ ]$  恢复原信号  $x_1(n)$ 。下面将说明  $D_0$  即“取对数”就是一个同态映射。这里，输入空间  $M_1$  的元素是正的实序列  $\bar{x} = x(n)$ ，而在  $M_1$  中定义的向量加法运算  $\square$  为序列的逐点相乘，即  $\bar{x} \square \bar{y} = x(n) \cdot y(n)$ ，定义的标乘运算  $\alpha \bar{x}$  是  $[x(n)]^\alpha$ 。由于两个正实序列之积为正实序列，正实序列的  $\alpha$  次方（ $\alpha$  为实数）仍为正实序列，所以  $M_1$  是一个合理的线性空间。而  $D_0$  同态变换后的输出空间  $M_2$  的元素为  $\hat{x}(n) = \ln[x(n)]$ ，在  $M_2$  中定义的向量加法就是一般的代数加（+），即  $\bar{x} + \bar{y} = \hat{x}(n) + \hat{y}(n)$ ，定义的标乘为  $\alpha \bar{x} = \alpha \hat{x}(n)$ 。显然  $M_2$  是一个线性空间。设  $\hat{x}_1(n) = \ln x_1(n)$ ， $\hat{x}_2(n) = \ln x_2(n)$ ，显然有：

$$\ln \left\{ [x_1(n)]^\alpha \cdot [x_2(n)]^\beta \right\} = \alpha \ln x_1(n) + \beta \ln x_2(n)$$

因而， $D_0$  是一个同态映射。

上面的例子称为乘法同态系统。如果输入空间  $M_1$  上的向量加定义为序列的卷积：

$$x_1(n) \square x_2(n) = x_1(n) * x_2(n)$$

则称之为卷积同态系统，这种系统用来减弱或消除卷积型干扰。

## 1.2.4 线性赋范空间

在研究信号空间时，常常既要考虑到它作为距离空间的几何特性，也要考虑到它作为线性空间的代数特性。完成这一点，是通过对线性空间中的每一元确定一个非负实数  $\|\bar{x}\|$ ，称为范数，它反映了该向量的“尺寸”。

定义：设  $E$  是复（或实）线性空间，对于  $E$  中的每个元素  $\bar{x}$  按照一定的法使其与一非负实数  $\|\bar{x}\|$  相对应，且  $\|\bar{x}\| \geq 0$ ，且  $\|\bar{x}\| = 0$  的充要条件为

$$\bar{x} = \bar{0} \quad (1.100a)$$

$$\|\bar{x} + \bar{y}\| \leq \|\bar{x}\| + \|\bar{y}\| \quad (\text{三角形不等式}) \quad (1.100b)$$

$$\|\alpha \bar{x}\| \geq |\alpha| \|\bar{x}\|, \quad \alpha \in C \quad (1.100c)$$

么，称  $E$  为复（或实）线性赋范空间， $\|\bar{x}\|$  为元素  $\bar{x}$  的范数。

对于线性赋范空间  $E$ ，可以定义其中元素  $\bar{x}$  和  $\bar{y}$  的距离为

$$\rho(\bar{x}, \bar{y}) = \|\bar{x} - \bar{y}\| \quad (1.101)$$

容易证明，式 (1.101) 定义的距离满足距离公理式 (1.75)，因而是一个距离， $E$  是一个距离空间。今后约定，凡讲到线性赋范空间时，总认为是距离空间，且距离由式 (1.101) 来定义。

应用式 (1.101)，则

$$\|\bar{x}\| = \|\bar{x} - \bar{0}\| = \rho(\bar{x}, \bar{0}) \quad (1.102)$$

出一个向量的范数就是它到原点的距离。

有了距离，便可在线性赋范空间中引入收敛性概念。点列  $\bar{x}_n \in E$  称为收敛是指

$$\lim_{n \rightarrow \infty} \|\bar{x}_n - \bar{x}\| = 0$$

这种收敛常称为依范数收敛。

一个线性赋范空间作为距离空间是完备的，则称之为巴拿赫空间。下面是理论中常用的巴拿赫空间的例子。

例 1.22 例 1.18 讨论的  $n$  维复欧氏空间  $C^n$  已证明是线性空间，如引进

$$\|\bar{x}\| = \left[ \sum_{i=1}^n |\alpha_i|^2 \right]^{\frac{1}{2}} \quad (1.103)$$

$\bar{x} = \{\alpha_1, \alpha_2, \dots, \alpha_n\}$ ，则容易证明， $C^n$  是一个巴拿赫空间。式 (1.103) 定

义的范数称为欧氏范数。

注意,对同一线性空间可以用不同的方式引进范数。例如,在  $C^n$  空间也可以用

$$\|\bar{x}\| = \max_i |\alpha_i| \quad (1.104)$$

来定义范数。这时,它仍是巴拿赫空间。

**例 1.23** 连续时间信号空间  $C[a, b]$ , 其元素  $x(t)$  是定义在  $T = \{t; a \leq t \leq b\}$  上的连续复值函数。显然  $C[a, b]$  为一线性空间, 若定义范数为

$$\|\bar{x}\| = \sup \{ |x(t)|; t \in T \} \quad (1.105)$$

则  $C[a, b]$  便成一线性赋范空间, 这种范数称为切比雪夫范数。由本章例 1.15 可知,  $C[a, b]$  空间便是一个巴拿赫空间。

**例 1.24**  $p$  方可积函数空间  $L^p$  ( $p \geq 1$ ), 如果元素的加法与标乘与例 1.20 一样, 按逐点方式定义, 则为一线性空间。若定义范数为

$$\|\bar{x}\| = \left[ \int_a^b |x(t)|^p dt \right]^{\frac{1}{p}} \quad (1.106)$$

则因它满足范数三公理, 且  $L^p$  是完备的, 从而是巴拿赫空间。

在信号分析中常用的  $L^2(T)$  空间, 它的元素是能量有限信号, 是  $L^p$  空间的特例, 因而也是巴拿赫空间。此时, 信号元素的范数为

$$\|\bar{x}\| = \left[ \int_T |x(t)|^2 dt \right]^{\frac{1}{2}} \quad (1.107)$$

式 (1.107) 定义的范数在信号分析中最常用, 一方面它具有明显的物理意义, 即表示信号能量的开方, 或信号的均方根值; 另一方面, 它将很方便地与下节内积的概念联系起来。采用式 (1.107) 作为范数的  $L^2$  空间的原点 (即空间的向量加法恒等元), 是在区间  $T$  上几乎处处为零的信号。

**例 1.25** 类似  $L^p$  空间, 本例讨论  $p$  方可和数列空间  $l^p$  ( $p \geq 1$ )。其中元素  $\bar{x} = \{\xi_1, \xi_2, \dots\}$  为满足  $\sum_{k=1}^{\infty} |\xi_k|^p < \infty$  的序列。如果定义向量加法为序列中相应位置的数值相加, 定义标乘为序列中所有数值的标乘, 则  $l^p$  形成一个线性空间。此时范数定义为

$$\|\bar{x}\| = \left[ \sum_{k=1}^{\infty} |\xi_k|^p \right]^{\frac{1}{p}} \quad (1.108)$$

则可证明  $l^p$  也是巴拿赫空间。

在信号分析中常用的平方可和序列空间  $l^2$  是  $l^p$  空间的特例，因而  $l^2$  也是一个巴拿赫空间，其元素  $\bar{x}$  的范数

$$\|\bar{x}\| = \left[ \sum_{k=1}^{\infty} |\xi_k|^2 \right]^{\frac{1}{2}} \quad (1.109)$$

然表示序列的能量开方。

## 2.5 内积空间

(1) 我们研究和发展信号空间基本概念的最后一步骤，是对信号空间提供附加的几何结构——内积。我们将主要讨论复线性空间，因为实线性空间只不过是其特例而已。

内积定义为复线性空间  $E$  的向量序偶  $(\bar{x}, \bar{y})$  到复平面  $C$  的一种映射，其在  $C$  平面的象记为  $(\bar{x}, \bar{y})$ ，其满足如下特性：

$$(\bar{x}, \bar{y}) = (\bar{y}, \bar{x}) \quad (1.110a)$$

$$(\alpha\bar{x} + \beta\bar{y}, \bar{z}) = \alpha(\bar{x}, \bar{z}) + \beta(\bar{y}, \bar{z}) \quad (1.110b)$$

$$(\bar{x}, \bar{x}) \geq 0; (\bar{x}, \bar{x}) = 0 \text{ 当且仅当 } \bar{x} = \bar{0} \quad (1.110c)$$

此时，称  $E$  为复内积空间，并称  $(\bar{x}, \bar{y})$  为  $x, y$  的内积。内积亦称为点积或标量积，在后面将会看到，它是欧氏空间中点积的概念向一般复线性空间的推广。

由内积的定义式 (1.110) 可以得出内积具有如下性质：

(a) 11

$$(\alpha\bar{x}, \bar{y}) = \alpha(\bar{x}, \bar{y}) \quad (1.111a)$$

11.11

$$(\bar{x}, \alpha\bar{y}) = \alpha^* (\bar{x}, \bar{y}) \quad (1.111b)$$

$$(\bar{x}, \bar{y} + \bar{z}) = (\bar{x}, \bar{y}) + (\bar{x}, \bar{z}) \quad (1.111c)$$

$$(\bar{x}, \bar{x}) = (\bar{x}, \bar{x})^* \quad [\text{即 } (\bar{x}, \bar{x}) \text{ 为实数}] \quad (1.111d)$$

内积空间一定是线性赋范空间，因为如果我们定义内积空间的范数为

$$\|\bar{x}\| = (\bar{x}, \bar{x})^{\frac{1}{2}} \quad (1.112)$$

可以证明式 (1.112) 是一个有效的范数。首先，由内积公设 (1.110c) 可知，

$\|\bar{x}\| = (\bar{x}, \bar{x})^{\frac{1}{2}} \geq 0$ ，当且仅当  $\bar{x} = \bar{0}$  时且有  $\|\bar{x}\| = 0$ ，因而  $\|\bar{x}\|$  满足范数的第一条公设式 (1.100a)。由内积公设式 (1.110b) 可知：

$$\|\alpha \bar{x}\| = (\alpha \bar{x}, \alpha \bar{x})^{\frac{1}{2}} = \left[ |\alpha|^2 (\bar{x}, \bar{x}) \right]^{\frac{1}{2}} = |\alpha| \|\bar{x}\|$$

因而  $\|\bar{x}\|$  满足范数第三条公设 (1.100c)。为了证明  $\|\bar{x}\|$  满足范数第二条公设式 (1.100b)，首先需要引入一个重要的关系式——许瓦兹不等式。

内积空间中的许瓦兹不等式可写为

$$|(\bar{x}, \bar{y})|^2 \leq (\bar{x}, \bar{x})(\bar{y}, \bar{y}) \quad (1.113)$$

其意义是，内积空间中两向量内积之平方，小于或等于两向量自身内积之积。为了证明式 (1.113)，首先将内积公设式 (1.110c) 中的  $\bar{x}$  代为  $\bar{x} + \alpha \bar{y}$ ，则

$$(\bar{x} + \alpha \bar{y}, \bar{x} + \alpha \bar{y}) = (\bar{x}, \bar{x}) + \alpha (\bar{y}, \bar{x}) + \alpha^* (\bar{x}, \bar{y}) + \alpha^2 (\bar{y}, \bar{y}) \geq 0 \quad (1.114)$$

令式 (1.114) 中  $\alpha = -(\bar{x}, \bar{y})/(\bar{y}, \bar{y})$ ，则

$$\begin{aligned} & (\bar{x}, \bar{x}) - \frac{(\bar{x}, \bar{y})}{(\bar{y}, \bar{y})}(\bar{y}, \bar{x}) - \frac{(\bar{x}, \bar{y})^*}{(\bar{y}, \bar{y})}(\bar{x}, \bar{y}) + \frac{|(\bar{x}, \bar{y})|^2}{(\bar{y}, \bar{y})} \\ &= (\bar{x}, \bar{x}) - \frac{|(\bar{x}, \bar{y})|^2}{(\bar{y}, \bar{y})} \geq 0 \end{aligned} \quad (1.115)$$

因而式 (1.113) 成立。应当着重指出，许瓦兹不等式 (1.113) 中，只有当  $\bar{x} = \alpha_1 \bar{y}$  ( $\alpha_1$  为任意复数) 时等式成立。为了说明这个问题，只需将式 (1.114) 变为等式的条件写为  $\bar{x} + \alpha \bar{y} = \bar{0}$ ，且令  $\alpha_1 = -\alpha$  即证得。用数学的语言来讲，当且仅当  $\bar{x} = \alpha_1 \bar{y}$  时，即  $\bar{x}$ 、 $\bar{y}$  线性相关时，有

$$|(\bar{x}, \bar{y})|^2 = (\bar{x}, \bar{x})(\bar{y}, \bar{y}) \quad (1.116)$$

下面使用许瓦兹不等式证明式 (1.112) 定义的内积空间的范数满足三角形不等式。因为  $\|\bar{x}\|^2 = (\bar{x}, \bar{x})$ ，则

$$\begin{aligned} \|\bar{x} + \bar{y}\|^2 &= (\bar{x} + \bar{y}, \bar{x} + \bar{y}) \\ &= (\bar{x}, \bar{x}) + (\bar{x}, \bar{y}) + (\bar{y}, \bar{x}) + (\bar{y}, \bar{y}) \\ &\leq \|\bar{x}\|^2 + \|\bar{y}\|^2 + 2\|\bar{x}\|\|\bar{y}\| \\ &= (\|\bar{x}\| + \|\bar{y}\|)^2 \end{aligned}$$

因而，

$$\|\bar{x} + \bar{y}\| \leq \|\bar{x}\| + \|\bar{y}\| \quad (1.117)$$



至此, 我们证明了式 (1.112) 是一个有效的范数, 因而, 任何内积空间是一个线性赋范空间, 从而也是一个距离空间。一个内积空间, 若它作为距离空间是完备的, 则称为希尔伯特空间, 一般用  $H$  表示。

为了进一步说明内积的几何意义, 有时我们可以认为内积提供了向量间的角度关系。许瓦兹不等式 (1.113) 可写为

$$|(\bar{x}, \bar{y})| \leq \|\bar{x}\| \cdot \|\bar{y}\| \quad (1.118)$$

因而可以定义  $\bar{x}$ 、 $\bar{y}$  之间的夹角为  $\theta$ , 且

$$\cos \theta = \frac{|(\bar{x}, \bar{y})|}{\|\bar{x}\| \|\bar{y}\|} \quad (1.119)$$

当且仅当  $(\bar{x}, \bar{y}) = 0$  时,  $\theta = \pm \frac{\pi}{2}$ , 因而称  $\bar{x}$ 、 $\bar{y}$  正交。当且仅当  $\bar{x} = d_1 \bar{y}$  时, 据式 (1.116),  $\theta = 0$ , 表示  $\bar{x}$ 、 $\bar{y}$  平行, 其夹角为零, 但是, 由于  $|(\bar{x}, \bar{y})|$  只能为正数或零, (1.119) 式定义的角度不可能处于第二和第三象限。因此, 今后我们在考虑内积提供的向量间的角度信息时, 主要考虑  $\bar{x}$ 、 $\bar{y}$  是否正交或平行。

下面介绍几种信号分析中常用的内积空间的例子。

**例 1.26** 例 1.25 所研究的  $l^2$  空间是一个线性赋范空间, 如果对  $l^2$  中任意两个元素  $\bar{x} = \{\xi_1, \xi_2, \dots\}$  和  $\bar{y} = \{\eta_1, \eta_2, \dots\}$  规定内积

$$(\bar{x}, \bar{y}) = \sum_{k=1}^{\infty} \xi_k \eta_k^* \quad (1.120)$$

则可证明式 (1.120) 满足内积三个公设, 因而  $l^2$  还是一个内积空间。同时亦可证明  $l^2$  是一个希尔伯特空间。

容易说明, 式 (1.120) 定义的内积给出范数

$$\|\bar{x}\| = (\bar{x}, \bar{x})^{\frac{1}{2}} = \left[ \sum_{k=1}^{\infty} \xi_k \xi_k^* \right]^{\frac{1}{2}} = \left[ \sum_{k=1}^{\infty} |\xi_k|^2 \right]^{\frac{1}{2}}$$

与式 (1.109) 完全一致。由于  $n$  维欧氏空间是  $l^2$  空间的特例, 因而也是一个希尔伯特空间。对  $n$  维欧氏空间  $C^n$ , 内积表示式 (1.120) 蜕化为

$$(\bar{x}, \bar{y}) = \sum_{k=1}^n \xi_k \eta_k^* = \bar{x}^T \bar{y} \quad (1.121)$$

式中,  $\bar{x}$ 、 $\bar{y}$  均为列向量表示法。即  $\bar{x} = [\xi_1, \xi_2, \dots, \xi_n]^T$ ,  $\bar{y} = [\eta_1, \eta_2, \dots, \eta_n]^T$ 。

**例 1.27** 例 1.26 所研究的  $L^2(T)$  空间是一个线性赋范空间。如果对  $L^2$  中任意两个元素  $\bar{x} = x(t)$  和  $\bar{y} = y(t)$  规定内积

$$(\bar{x}, \bar{y}) = \int_T x(t) y^*(t) dt \quad x, y \in L^2(T) \quad (1.122)$$

则可证明式 (1.122) 满足内积三个公设, 因而  $L^2$  空间是一个内积空间。同时亦可证明  $L^2$  空间是一个希尔伯特空间。

容易证明, 式 (1.122) 定义的内积给出范数

$$\|\bar{x}\| = (\bar{x}, \bar{x})^{\frac{1}{2}} = \left[ \int_T x(t) x^*(t) dt \right]^{\frac{1}{2}} = \left[ \int_T x^2(t) dt \right]^{\frac{1}{2}}$$

与式 (1.107) 完全一致。

**例 1.28** 内积概念在信号分析中的一个重要应用是研究  $L^2(-\infty, \infty)$  空间 (即能量有限信号空间) 中信号的起伏特性。图 1.18 给出了快起伏和慢起伏两种信号。直观上来看, 对于慢起伏信号, 如果在时间上作一个小的位移形成一个新信号, 它与原信号的差别很小, 也就是这两个信号在信号空间中的距离很小。反之, 对于一个快起伏信号, 这两个信号在信号空间中的距离较大。现设  $\bar{x}_\tau$  表示  $\bar{x}$  在时间上位移  $\tau$ , 即令  $x_\tau(t) = x(t + \tau)$ , 则有

$$\begin{aligned} d^2(\bar{x}, \bar{x}_\tau) &= \|\bar{x} - \bar{x}_\tau\|^2 = (\bar{x} - \bar{x}_\tau, \bar{x} - \bar{x}_\tau) \\ &= (\bar{x}, \bar{x}) + (\bar{x}_\tau, \bar{x}_\tau) - (\bar{x}_\tau, \bar{x}) - (\bar{x}, \bar{x}_\tau) \\ &= \|\bar{x}\|^2 + \|\bar{x}_\tau\|^2 - 2\operatorname{Re}(\bar{x}, \bar{x}_\tau) \end{aligned} \quad (1.123)$$

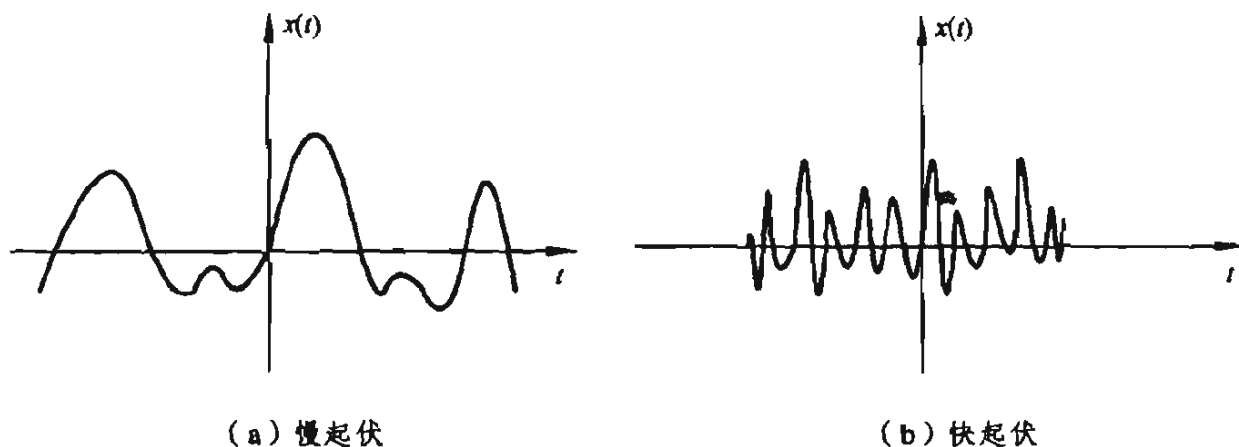


图 1.18 两种起伏特性的信号

由于时移不改变信号能量, 即  $\|\bar{x}\|^2 = \|\bar{x}_\tau\|^2$ , 因而

$$\begin{aligned} d^2(\bar{x}, \bar{x}_\tau) &= 2[\|\bar{x}\|^2 - \operatorname{Re}(\bar{x}, \bar{x}_\tau)] \\ &= 2[\gamma_x(0) - \operatorname{Re}\gamma_x(\tau)] \end{aligned} \quad (1.124)$$

式 (1.124) 中

$$\gamma_x(\tau) = (\bar{x}, \bar{x}_\tau) = \int_{-\infty}^{\infty} x(t)x^*(t+\tau)dt \quad (1.125)$$

式 (1.125) 定义的  $\gamma_x(\tau)$  是信号  $x(t)$  的自相关函数, 即信号  $x(t)$  与其时移信号  $x(t+\tau)$  之内积, 而自相关函数的实部  $\operatorname{Re}[\gamma_x(\tau)]$  称为时间模糊函数。由式 (1.124) 可以看出, 对于给定  $\tau$ , 时间模糊函数  $\operatorname{Re}[\gamma_x(\tau)]$  越大,  $x(t)$  与  $x(t+\tau)$  之间的距离越小, 因而模糊性越大。时间模糊函数在雷达理论中起着重要作用。如果我们想区分距离上相近的两个目标, 则希望它们的回波  $x(t)$  和  $x(t+\tau)$  之间的距离足够大, 因而要求时间模糊函数  $\operatorname{Re}[\gamma_x(\tau)]$  在给定的  $\tau$  上足够小。通常在雷达上是通过发射极短脉冲或发射极大带宽的信号来获得良好距离分辨力的。

应当说明, 对于实信号  $x(t)$ , 自相关函数与时间模糊函数一致, 且  $\gamma_x(\tau)$  是  $\tau$  的偶函数。典型的实信号自相关函数如图 1.19 所示。从图中可看出, 当  $\tau \rightarrow \pm\infty$  时, 一般有  $\gamma_x(\tau) \rightarrow 0$ 。对式 (1.125) 两边取傅里叶变换, 则得  $\gamma_x(\tau)$  的傅氏变换  $R_x(f)$  的表达式

$$\begin{aligned} R_x(f) &= \int_{-\infty}^{\infty} \gamma_x(\tau) e^{-j2\pi f\tau} d\tau \\ &= \iint_{-\infty}^{\infty} x(t) \cdot x(t+\tau) e^{-j2\pi f\tau} d\tau dy \\ &= \int_{-\infty}^{\infty} x(t) e^{-j2\pi ft} dt \int_{-\infty}^{\infty} x(t+\tau) e^{-j2\pi f(t+\tau)} d(t+\tau) \\ &= X(f) \cdot X^*(f) = |X(f)|^2 \end{aligned} \quad (1.126)$$

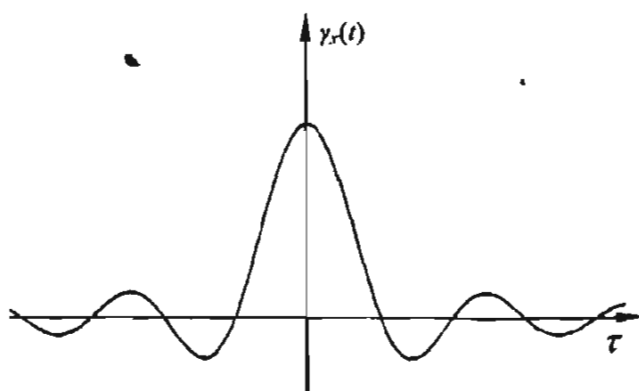


图 1.19 典型的实信号自相关函数

由式 (1.126) 可看出, 实信号的自相关函数仅仅由信号的幅频特性决定, 而与信号的相频特性无关。同时可以看出, 实信号自相关函数的傅里叶变换是  $f$  的实函数, 这是  $\gamma_x(\tau)$  是偶对称函数决定的。

### 1.2.6 有限维内积空间中信号的表示法

引入内积空间的一个直接好处, 就是能方便地表示有限维内积空间中的信号。设  $M$  为一个  $n$  维线性赋范空间, 由基底  $\{\bar{\varphi}_i; i=1, 2, \dots, n\}$  生成, 则  $\bar{x} \in M$  由下式给出

$$\bar{x} = \sum_{i=1}^n \alpha_i \bar{\varphi}_i \quad (1.127)$$

式 (1.127) 两边取内积, 则

$$(\bar{x}, \bar{\varphi}_j) = \sum_{i=1}^n \alpha_i (\bar{\varphi}_i, \bar{\varphi}_j); \quad j=1, 2, \dots, n \quad (1.128)$$

式 (1.128) 给出了  $n$  元联立方程组, 从而可解出  $\bar{\alpha} = (\alpha_1, \alpha_2, \dots, \alpha_n)$ , 提供了  $\bar{x}$  表达式 (1.127) 中的系数。这样,  $\bar{\alpha}$  则称为  $n$  维内积空间中信号  $\bar{x}$  相对于基集  $(\bar{\varphi}_i)$  的表示法。

另一种更方便的表示法, 对内积空间  $M$  建立基集  $(\bar{\varphi}_i)$  两两正交的基集  $(\bar{\theta}_j)$ , 即

$$(\bar{\varphi}_i, \bar{\theta}_j) = \delta_{ij} = \begin{cases} 1, & i=j \\ 0, & i \neq j \end{cases} \quad i, j=1, 2, \dots, n \quad (1.129)$$

式中,  $\delta_{ij}$  称为 Kronecker- $\delta$  函数。满足式 (1.129) 的基集称为对偶基集。为了进一步直观看出对偶基集的性质, 我们举出如下二维空间的例子。设  $\bar{\varphi}_1 = (2, 0)$ ,  $\bar{\varphi}_2 = (2, 3)$ , 则其对偶基集为  $(\bar{\theta}_1) = \left(1, -\frac{2}{3}\right)$ ,  $(\bar{\theta}_2) = \left(0, \frac{1}{3}\right)$ 。读者可以自行验证上式中,  $(\bar{\varphi}_1, \bar{\theta}_2) = (\bar{\varphi}_2, \bar{\theta}_1) = 0$ ,  $(\bar{\varphi}_1, \bar{\theta}_1) = (\bar{\varphi}_2, \bar{\theta}_2) = 1$ 。图 1.20 给出了该例图示, 可以看出,  $\bar{\theta}_1$  与  $\bar{\varphi}_2$  正交,  $\bar{\theta}_2$  与  $\bar{\varphi}_1$  正交。

有了对偶基集的概念后, 就可以更简洁地表示希尔伯特空间的信号。由式 (1.127) 和式 (1.129), 有

$$(\bar{x}, \bar{\theta}_j) = \sum_{i=1}^n \alpha_i (\bar{\varphi}_i, \bar{\theta}_j) \Rightarrow \alpha = (\bar{x}, \bar{\theta}_j); \quad j=1, 2, \dots, n \quad (1.130)$$

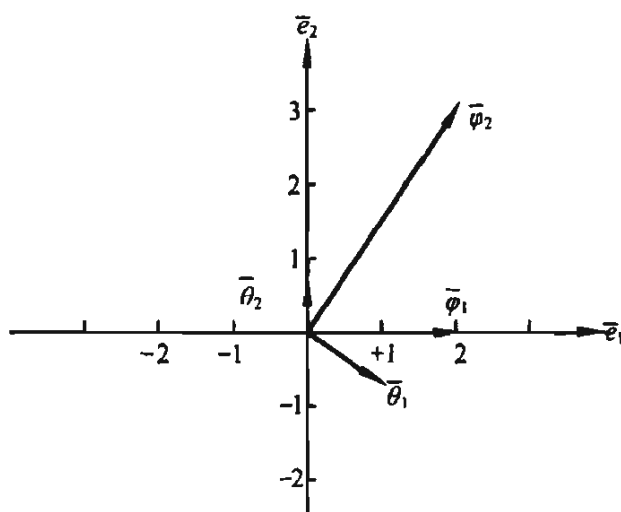


图 1.20 二维空间中的对偶基集举例

将式 (1.130) 代入式 (1.127), 则

$$\bar{x} = \sum_{i=1}^n (\bar{x}, \bar{\theta}_j) \bar{\varphi}_i = \sum_{i=1}^n (\bar{x}, \bar{\varphi}_i) \bar{\theta}_j \quad (1.131)$$

式 (1.131) 表示,  $n$  维内积空间中信号可以表示为基向量的线性组合, 而系数由该信号与相应的对偶基向量的内积给出。

在信号分析中, 使用归一化正交集作为基集更为方便。此时基集  $(\bar{\varphi}_i)$  与它对偶, 即  $(\bar{\varphi}_i)$  为自对偶基集:

$$(\bar{\varphi}_i, \bar{\varphi}_j) = \delta_{ij} \quad (1.132)$$

由式 (1.132) 可以看出, 自对偶与归一化正交的概念完全一致。在二维空  
正交基集的例子是  $\bar{\varphi}_1 = (1, 0)$ ,  $\bar{\varphi}_2 = (0, 1)$ 。当使用归一化正交集作为基集  
式 (1.131) 变为

$$\bar{x} = \sum_{i=1}^n (\bar{x}, \bar{\varphi}_i) \bar{\varphi}_i \quad (1.133)$$

内积空间  $M$  中的信号  $\bar{x}$  的表达式更为方便。使用归一化正交集的好处是,  
可以建立  $M$  空间与  $C^n$  空间中点的一一对应关系, 而且两空间中对应点的  
相等。这可证明如下。

设  $\bar{x} = \sum_{i=1}^n \alpha_i \bar{\varphi}_i$ 、 $\bar{y} = \sum_{i=1}^n \beta_i \bar{\varphi}_i$ , 显然,  $\bar{x}, \bar{y} \in M$ ,  $\bar{\alpha} = (\alpha_1, \alpha_2, \dots, \alpha_n)$ ,  
 $(\beta_1, \beta_2, \dots, \beta_n) \in C^n$ , 则我们有

$$\begin{aligned}
(\bar{x}, \bar{y}) &= \left( \sum_{i=1}^n \alpha_i \bar{\varphi}_i, \sum_{i=1}^n \beta_i \bar{\varphi}_i \right) \\
&= \sum_{i=1}^n \sum_{j=1}^n \alpha_i \beta_j^* (\bar{\varphi}_i, \bar{\varphi}_j) \\
&= \sum_{i=1}^n \alpha_i \beta_i^* = (\bar{\alpha}, \bar{\beta})
\end{aligned} \tag{1.134}$$

在证明上式的最后一步，显然应用了例 1.26 中关于  $C^n$  空间的内积定义式 (1.121)。

由于归一化正交基集在分析问题中极为方便，因而下面研究把  $n$  维空间中  $n$  个线性无关向量  $\{\bar{u}_i; i=1,2,\dots,n\}$  形成的基集变换产生归一化正交基集的办法。这个办法叫格拉姆-施密特 (Gram-Schmidt) 正交归一化步骤。由于格拉姆-施密特步骤具有迭代性质，因而便于用计算机计算。这些迭代公式首先由  $M$  中的  $n$  个线性无关向量  $\{\bar{u}_i; i=1,2,\dots,n\}$  产生正交集  $\{\bar{v}_i; i=1,2,\dots,n\}$ ，然后归一化  $\{\bar{v}_i\}$  得到归一化正交集  $\{\bar{\varphi}_i; i=1,2,\dots,n\}$ ，具体如下：

$$\begin{aligned}
\bar{v}_1 &= \bar{u}_1 \\
\bar{v}_2 &= \bar{u}_2 - (\bar{u}_2, \bar{\varphi}_1) \bar{\varphi}_1 \\
\bar{v}_3 &= \bar{u}_3 - (\bar{u}_3, \bar{\varphi}_1) \bar{\varphi}_1 - (\bar{u}_3, \bar{\varphi}_2) \bar{\varphi}_2 \\
&\vdots \\
\bar{v}_i &= \bar{u}_i - \sum_{k=1}^{i-1} (\bar{u}_i, \bar{\varphi}_k) \bar{\varphi}_k
\end{aligned} \tag{1.135}$$

式中

$$\bar{\varphi}_i = \frac{\bar{v}_i}{\|\bar{v}_i\|}; \quad i=1,2,\dots,n \tag{1.136}$$

应当说明，由格拉姆-施密特步骤产生的归一化正交基集不是唯一的，只要把集合  $\{\bar{u}_i\}$  重新排列，使用格拉姆-施密特步骤，就会产生不同的归一化正交基集。

### 参 考 资 料

- [1] 向敬成，刘醒凡. 信号理论. 成都：电子科技大学出版社，1988
- [2] L.E.Franks. Signal Theory. Prenlic-hall, Inc, Englewood Cliffs, N.J.U.S.A.1981
- [3] Urkowitz, Harry. Signal Theory and random processes. N.J.Artech House 1983

## 第2章 信号和图像的基于过完备原子库的稀疏表示与稀疏分解

### 2.1 引言

在众多信号处理的应用中,都希望找到一种稀疏的数据表示。用稀疏逼近代替原始数据表示可从实质上降低信号处理的成本,提高压缩效率。传统的信号表示理论基于正交线性变换,但许多信号是各种自然现象的混合体,这些混合信号在单一的正交基变换中不能非常有效地表现出来。例如,一个含有脉冲和正弦波形的混合信号,既不能用单一的脉冲基函数,也不能用单一的正弦基函数有效地表示。在这个例子中,有两种结构类型同时出现在信号里,但它们完全不同,没有其中哪一个能有效地模拟另一个。所以,人们希望寻找一种能同时建立在两种基函数之上的信号表示,其结果应该比采用其中任一种基函数有效得多。

在图像和视频处理方面,常用的信号分解方式通常是非冗余的正交变换,如离散余弦变换、小波变换等。离散余弦变换其基函数缺乏时间/空间分辨率,不能有效地提取具有时频局部化特性的信号特征。小波分析在处理一维和具有点状奇异性的对象时,表现出良好的性能,但图像边缘的不连续性空间分布的,小波分析在处理这种线状奇异性时效果并不是很好。因而说,分析对于多维信号来说并不是最优的,它们不能稀疏地捕捉到图像结构的特征,因此在图像和多维编码方面的新突破,必定取决于信号表示的深刻变革。

最近几年,研究者在改变传统信号表示方面的研究有了很大的进展。新信号表示理论的基本思想就是:基函数用称之为原子库的过完备的冗余函数所取代,原子库的选择应尽可能好的符合被逼近的信号的结构,其构成可以没有任何限制,原子库中的元素被称为原子。从原子库中找到具有最佳线性组

合的  $m$  项原子来表示一个信号，被称作是信号的稀疏逼近 (Sparse Approximation) 或高度非线性逼近 (Highly Nonlinear Approximation) <sup>[1]</sup>。

从非线性逼近的角度来讲，高度非线性逼近包含两个层面：一是根据目标函数从一个给定的基库中挑选好的或最好的基；二是从这个好的基中挑选最好的  $m$  项组合。利用贪婪算法和自适应追踪从一个冗余函数系统中进行  $m$  项逼近或阈值逼近也属此例。我们对高度非线性逼近方法的理解只是些零星的片段，用高度非线性方法以指定的逼近速率来描述函数仍然是一个富有挑战的问题 (R.A.DeVore 语) <sup>[2]</sup>。

从基函数的形成来讲，在图像表示方面体现为多尺度几何分析<sup>[3]</sup>，curvelets<sup>[4]</sup>、bandlets<sup>[5]</sup> 和 contourlets<sup>[6]</sup>都强调，基函数应具备下述特点：① 多分辨率分析；② 时频定位能力；③ 全角度分析（方向性）；④ 各向异性的尺度变换。这些新的冗余函数系统的不断涌现，使信号稀疏表示的方法更加成为研究的热点。

过完备信号稀疏表示方法肇始于 20 世纪 90 年代，1993 年，S. Mallat 和 Z. Zhang 首次提出了应用过完备冗余原子库对信号进行稀疏分解的思想，并引入了匹配追踪 (Marching Pursuit, MP) 算法<sup>[7]</sup>。在本章参考资料[7]中，作者举自然语言表述做浅显的类比，说明过完备冗余原子库对信号表示的必要性，同时强调原子库的构成应较好地符合信号本身所固有的特性，以实现 MP 算法的自适应分解。

新思想的提出引起人们极大的关注，但由于算法所涉及的计算量庞大，计算任务十分繁重，因而早期研究的焦点集中在如何实现该算法的快速计算，降低算法的复杂度，以及选择何种类型的原子构造合适的原子库两方面。这期间，许多音视频信号处理方面的实验都对 MP 算法做出了有利的支持，尤其在甚低码率视频编码方面，MP 算法更显示出极大的优越性<sup>[8]</sup>。

1999 年，D. Donoho 等人又另辟蹊径，提出了基追踪 (Basis Pursuit) 算法<sup>[9]</sup>，并从实验的角度举证了 MP、MOF<sup>[10]</sup>和 BOB<sup>[11]</sup>算法各自的优劣。稍后，又在 2001 年发表的另一篇重要文章<sup>[12]</sup>中，给出了基于 BP 算法的稀疏表示具有唯一解的边界条件，并提出了原子库的互不相干性 (incoherent dictionaries) 的概念。

本章以稀疏信号的精确重构条件为主线，综述了基于冗余原子库的信号稀疏分解和非线性逼近领域的一系列最新理论成果。本章结构如下：2.2 节引入稀疏分解的数学表示和非相干原子库的定义；2.3 节介绍满足 BP 和 MP 算法重构条件的相关定理；2.4 节介绍级联原子库 (concatenate dictionaries) 的概念及其发展前景。



## 2 信号的稀疏逼近

### 2.1 稀疏逼近的定义

高度非线性逼近理论主要关注下述问题：给定一个集合  $\{g_k; k=1,2,\dots,K\}$ ，其元素是张成整个 Hilbert 空间  $\mathcal{H}=\mathbf{R}^N$  的单位矢量， $K \geq N$ ，我们称集合  $\mathcal{D}$  为原子库，其元素为原子。对于任意给定的信号  $f \in \mathcal{H}$ ，我们希望能在  $\mathcal{D}$  中自适应地选取  $m$  个原子对信号  $f$  做  $m$  项逼近：

$$f_m = \sum_{\gamma \in I_m} c_\gamma g_\gamma \quad (2.1)$$

中， $I_m$  是  $g_\gamma$  的下标集， $\text{card}(I_m)=m$ ，则  $\mathcal{B}=\text{span}(g_\gamma, \gamma \in I_m)$  就是由  $m$  个原子原子库  $\mathcal{D}$  中张成的最佳子集。我们定义逼近误差为

$$\sigma_m(f, \mathcal{D}) = \inf_{f_m} \|f - f_m\| \quad (2.2)$$

由于  $m$  远小于空间的维数  $N$ ，这种逼近也被称作稀疏逼近。鉴于原子库的冗余性 ( $K > N$ )，矢量  $g_k$  不再是线性独立的，因而 (2.1) 式有多种解答，不同组合中原子所张成的子空间  $\mathcal{B}$  构成原子库  $\mathcal{D}$  中不同的向量基。从稀疏逼近的角度，我们希望在满足 (2.2) 式的前提下，从各种可能的组合中，挑选出分数最为稀疏的一个，或者说  $m$  取值为最小的一个。当  $\mathcal{D}$  是  $\mathcal{H}$  空间的一个正交基时，那么如何得到信号  $f$  的最佳  $m$  项逼近就是一件显而易见的事：即保留的内积  $\langle f, g_\gamma \rangle$  最大的  $m$  个基原子。然而对于一个随机的冗余原子库来说，这是一个 NP 难问题<sup>[13]</sup>。

### 2.2 非相干原子库

要想在冗余原子库中获得高度非线性逼近的建设性结果，我们必须首先将注意力集中在某些特别的原子库上。许多研究人员瞄准了非相干原子库，也就是相干系数  $\mu$  小于某个常数的原子库。相干系数的定义为

$$\mu = \sup_{\substack{i,j \\ i \neq j}} |\langle g_i, g_j \rangle| \quad (2.3)$$

当相干系数较大时，原子间的相互关联也较强，如果  $\mu=1$ ，则意味着原子库中至少包含了两个一模一样的原子。反之，当相干系数较小时，我们就称原子库是非相干的，正交基的相干系数为零。

相干系数为原子库的冗余性提供了另一种可能的测度手段。式 (2.3) 说明：当  $\mu$  充分小时，原子库  $\mathcal{D}$ （虽然可能是过完备的）接近一个正交基。

## 2.3 稀疏信号的精确重构条件

20 世纪 90 年代初期，MP 作为一种寻找信号稀疏表示的策略首先被引入，以代替框架表示方法。很快，研究者们便在实验中发现：如果  $f$  在  $\mathcal{D}$  中有非常稀疏的扩展，并且  $\mathcal{D}$  是由三角函数和 Dirac 函数组成的原子库，那么通过 BP 算法，这个稀疏扩展就可以完全重构。实验的观察结果很快转入一系列理论的证明，首先得到验证的是两个正交基的联合<sup>[12, 14, 15]</sup>，然后是数个不相干基的联合以及更普遍的准不相干原子库<sup>[16~20]</sup>。

### 2.3.1 基追踪

找到最好的、也就是最稀疏的信号表示，等同于解决下述问题：

$$\text{minimize } \|c\|_0 \quad \text{subject to } f = \sum_{k=0}^{K-1} c_k g_k \quad (2.4)$$

其中， $\|c\|_0$  是序列  $\{c_k\}$  中非零项的个数。如前所述，从一个随机冗余原子库中寻找信号的稀疏扩展是一个 NP 难问题，为解决这一难点，Chen、Donoho 和 Saunders 提出解决下述稍有差别的问题：

$$\text{minimize } \|c\|_1 \quad \text{subject to } f = \sum_{k=0}^{K-1} c_k g_k \quad (2.5)$$

这一微妙的修改，转化了问题的性质。的确，最小化  $\ell_1$  范数，被称为 Basis Pursuit 或 BP，是较为简单的问题，可以通过线性规划的方法解决<sup>[9, 21, 22]</sup>。本章参考资料[9]所提出的 BP 算法结合了当前线性规划以及基于原子库结构的特定快速变换，但计算复杂度仍很大，对于结构不好的原子库来说算法也不可靠。最近，由 J.A.Tropp 提出的凸松弛法(convex relaxation approach)<sup>[23]</sup>和 Z. S. Wang 等人提出的 MF 神经网络的方法似乎是一个很好的方向<sup>[24]</sup>。

在 Donoho 和 Huo<sup>[12]</sup>的研究基础上，Elad 和 Bruckstein<sup>[14]</sup>证明了下述引人注目的定理。

**定理 1** 让  $\mathcal{D}$  是一个相干系数为  $\mu$  的原子库，如果一个信号  $f \in \mathbb{R}^N$  可以表示为：

$$f = \sum_{i=0}^m c_i g_i \quad (2.6)$$

并且  $\|c\|_0 < 1/\mu$ ，则式 (2.6) 就是  $f$  在  $\mathcal{D}$  中的唯一的最稀疏的扩展式。

这一结果说明，虽然冗余性排除了扩展的唯一性，但如果原子库是不相干的，仍然可以找到充分稀疏的唯一解。也就是说，如果  $f$  在  $\mathcal{D}$  中有非常稀疏的扩展，那么通过 BP 算法，这个稀疏扩展就可以精确重构。

**定理 2** 如果信号  $f$  在原子库  $\mathcal{D}$  中有一个稀疏表示，并且满足

$$\|c\|_0 < \frac{\sqrt{2}-0.5}{\mu}$$

则最小化问题  $\ell_1$  有一个唯一解，它也是最小化问题  $\ell_0$  的解。

这个结果意味着，我们可以通过 BP 用较为简单的问题  $\ell_1$ ，取代原始的寻找  $f$  最稀疏表示的组合优化问题。换句话说，原子库  $\mathcal{D}$  的非相干性帮助我们解决了 NP 难问题。

### 3.2 匹配追踪 (Matching Pursuit) 及其变种

尽管使用了线性规划方法，基追踪法由于要在所有的原子库向量的不同组中寻求满足算式 (2.5) 成立的极小化  $\|c\|_1$  的解，其计算花销仍是昂贵的，人转而采用能够求解局部最优的贪婪算法以取代求  $\|c\|_0$  的全局最优，以减少计算复杂性。MP 是一个迭代的贪婪算法，它是在每一次迭代过程中，从原子选择最能匹配信号结构的一个原子而构建的一种逼近过程。在统计学界，被称作投影跟踪 (Projection Pursuit) [25, 26]，在逼近学领域，被称为纯贪婪 (Pure Greedy Algorithm) [27]。

假设在  $\mathcal{D}$  中所有原子的范数都为 1，算法在初始化时，设  $R_0 = f$ ，我们可先将信号分解为：

$$R_0 = \langle g_n, R_0 \rangle g_n + R_1$$

中， $\langle g_n, R_0 \rangle g_n$  是  $R_0$  对  $g_n$  的投影， $R_1$  是剩余量。显然， $g_n$  是与  $R_1$  正交的，我们可以得到下式：

$$\|R_0\|^2 = |\langle g_n, R_0 \rangle|^2 + \|R_1\|^2 \quad (2.7)$$

如果我们想使剩余量  $R_1$  的能量最小化，就必须使投影值  $|\langle g_n, R_0 \rangle|$  极大化。我们简单地对剩余量实施同样的步骤，并设定：

$$|\langle g_{\gamma_m}, R_m \rangle| = \sup_{\gamma \in I} |\langle g_{\gamma}, R_m \rangle| \quad (2.8)$$

$$f_{m+1} = f_m + \langle g_{\gamma_{m+1}}, R_{m+1} \rangle g_{\gamma_{m+1}} \quad (2.9)$$

$$R_{m+1} = f - f_{m+1} \quad (2.10)$$

则在第  $M$  步, 可得到下面的逼近:

$$f = \sum_{m=0}^{M-1} \langle g_{\gamma_m}, R_m \rangle g_{\gamma_m} + R_M \quad (2.11)$$

其中剩余量的范数 (逼近误差) 满足:

$$\|R_M\|^2 = \|f\|^2 - \sum_{m=0}^{M-1} |\langle g_{\gamma_m}, R_m \rangle|^2 \quad (2.12)$$

对于无限维空间的原子库, (2.8) 式中的上确界不一定能达到, 因而可以考虑下述选择规则, 使  $|\langle g_{\gamma_m}, R_m \rangle|$  几乎为最优:

$$|\langle g_{\gamma_m}, R_m \rangle| \geq \alpha \sup_{\gamma \in I} |\langle g_{\gamma}, R_m \rangle|, \quad 0 < \alpha \leq 1 \quad (2.13)$$

这个算法被称为弱匹配追踪 (Weak Matching Pursuit)<sup>[28]</sup>, 其中,  $\alpha \in (0, 1]$  是一个最优性因子。

纯贪婪算法的一个明显缺点就是, 在已选原子组成的子空间  $\mathcal{V}_m := \text{span}(g_{\gamma_1}, \dots, g_{\gamma_m})$  上, 信号的扩展可能不是最好的, 因为它不是一个正交投影。正交匹配追踪 (Orthogonal Matching Pursuit)<sup>[29]</sup> 递归地对所选原子的集合进行正交化, 解决了这个问题。它采用下述规则代替式 (2.9):

$$f_{m+1} = P_{\mathcal{V}_{m+1}} f \quad (2.14)$$

其中,  $P_{\mathcal{V}}$  是到无限维子空间  $\mathcal{V}$  上的正交投影算子。

### 2.3.3 MP 的收敛与重构

在一系列精确重构稀疏信号的理论证明中, J.A.Tropp 提出了一个较为通用的理论<sup>[16]</sup>, 他将 OMP 和 BP 的重构问题统一在一个单一的充分条件中。

**定理 3** 给定一个原子库  $\mathcal{D}$ , 其相干系数为  $\mu$ , 只要下述条件满足:

$$m < \frac{1}{2}(\mu^{-1} + 1) \quad (2.15)$$

则 OMP 和 BP 二者都可以从原子库  $\mathcal{D}$  中重构  $m$  项原子的组合。

与 OMP 比较, 在理论方面对 MP 性能的论证似乎更困难一些。虽然在有限维空间, MP 算法的逼近误差按指数级衰减:

$$\|R_m\| \leq (1 - \alpha^2 \beta^2)^{m/2} \|f\| \quad (2.16)$$

$$\|f_m - f\|^2 \leq \varepsilon^m \|f\|^2 \quad (2.17)$$

中, 参数  $\beta$  取决于原子库的结构, 它表示原子库函数捕捉任意输入信号特征能力:

$$\sup_{\gamma \in I_m} |\langle f, g_\gamma \rangle| \geq \beta \|f\| \quad (2.18)$$

对于无限维空间, Jones<sup>[30]</sup>证明 MP 仍然是收敛的, 但没给出收敛速率。Vore 和 Temlyakov 给出的反例显示<sup>[28]</sup>: 对于一个“好”的函数, 可以找到一“坏”的原子库, 使 MP 做出很“差劲”的表现。这说明, 需要对原子库  $\mathcal{D}$  (或) 信号  $f$  做某种限制, 以使 MP 具备好的逼近行为。诚然, Villemoes<sup>[31]</sup>给出的正面的例子验证了 MP 在 Walsh 小波包原子库中表现良好, 但直到 R. Monval 和 P. Vandergheynst 对 MP 算法在理论上做出了相应的证明<sup>[32]</sup>, 才使 MP 算法有了充分的理论依据。

**定理 4** 设  $\mathcal{D}$  是有限或无限维  $\mathcal{H}$  空间的一个原子库, 其相干系数为  $\mu$ , 对满足任意有限下标集  $\text{card}(I_m) = m < (1 + \mu^{-1})/2$  的函数  $f = \sum_{\gamma \in I_m} c_\gamma g_\gamma$

$m(g_\gamma, \gamma \in I_m)$ , MP 可以:

- ① 在每一步拣选出唯一“正确”的原子  $g_{k_n}$  ( $\forall n, k_n \in I_m$ );
- ②  $f_n$  按指数率收敛于  $f$

$$\|f_n - f\|^2 \leq ((1 - 1/m)(1 + \mu))^n \|f\|^2 \quad (2.19)$$

上述定理解释了在  $f$  可以表示为精确扩展的条件下 MP 的性能。但大多数下, 信号或图像在实际的原子库中只能表示为某些原子线性组合的逼近, 情况可由下述定理描述:

**定理 5** 设  $\{f_n\}$  是  $f \in \mathcal{H}$  在 MP 算法下经过  $n$  次迭代产生的逼近序列, 设  $m < (1 + \mu^{-1})/4$ , 并且,  $\tilde{f}_m = \sum_{k \in I_m} c_k g_k$  是  $f$  在原子库  $\mathcal{D}$  中的最佳  $m$  项逼近, 即:

$$\|\tilde{f}_m - f\| = \sigma_m(f) := \inf\{\|f - D_I c\|, \text{card}(I) \leq m, c \in C^I\} \quad (2.20)$$

那么, 存在一个整数  $N_m$ , 使得:

① 在第  $N_m$  次迭代时, 满足

$$\|f_{N_m} - f\| \leq \sqrt{1+4m} \sigma_m \quad (2.21)$$

② 在第  $N_m$  次迭代过程中, MP 从最佳  $m$  项逼近  $k_n \in \tilde{I}_m$  中挑选出相应的原子;

③ 如果  $\sigma_m^2 < 3\sigma_1^2/m$ , 那么

$$N_m \leq 2 + m \cdot \frac{4}{3} \cdot \ln \frac{3\sigma_1^2}{m\sigma_m^2} \quad (2.22)$$

上述定理在稀疏逼近的收敛率上为 MP 算法提供了很好的理论依据。

## 2.4 过完备不相干级联原子库

在  $N$  维有限空间, 两个正交基级联的原子库, 其相干系数  $\mu > \frac{1}{\sqrt{N}}$ , 如果两个正交基包含一个公共原子, 则其相干系数  $\mu = 1$ , 因此, 如果一个完全由正交基级联而成的原子库其相干系数为  $\mu = 1/\sqrt{N}$ , 则这个级联原子库就被认为是(完全)不相干的。

Dirac 函数和三角函数是两类最基本的正交基, 任意一个 Dirac 函数与任意一个三角函数的点积等于  $1/\sqrt{N}$ , 因而将这两个基联合构成一个级联原子库<sup>[12]</sup>, 其相干系数为  $\mu = 1/\sqrt{N}$ 。由这一对正交基联合而成的不相干级联原子库, 是最早被证明符合精确重构条件的。还有一些由其他对不相干的正交基级联的例子, 例如, 基于  $L^2(0,1)$  空间上的 Haar 和 Walsh 函数<sup>[33, 34]</sup>。对于由  $L$  个正交基联合而成的级联原子库  $\mathcal{D} = [B_1, B_2, \dots, B_L]$ , 存在下述定理<sup>[35]</sup>:

**定理 6** 设  $\mathcal{D}$  是一个由  $L$  个正交基联合而成的级联原子库,  $f = \sum_{k \in I} c_k g_k \in \text{span}(g_k, k \in I)$ , 如果

$$\text{card}(I) < \left( \sqrt{2} - 1 + \frac{1}{2(L-1)} \right) \mu^{-1} \quad (2.23)$$

则  $c_0(f) = c_1(f) = c$  就同时是最小化  $\ell_0$  问题和  $\ell_1$  问题的唯一解。

与 2.3 节讨论稀疏信号的精确重构条件一样, 上述定理也涉及原子库的相干系数问题。那么, 多个正交基的联合组成的原子库是否也是非相干的呢? 我们可以应用下述定理建立这样的原子库<sup>[36, 37]</sup>:

**定理 7** 设  $N = 2^{j+1}$ ,  $j \geq 0$ , 并且  $\mathcal{D} = [B_1, B_2, \dots, B_L]$  是空间  $\mathcal{H} = \mathbb{R}^N$  上的原子库, 其中  $L = 2^j = N/2$ , 则对任意一对原子  $u, v \in \mathcal{D}$ ,  $u \neq v$ :  $|\langle u, v \rangle| \in \{0, N^{-1/2}\}$ 。

这个定理告诉我们, 对于空间维数  $N = 2^{j+1}$  的  $\mathcal{H} = \mathbb{R}^N$ , 我们可以选取一个由  $N/2$  个正交基联合组成的原子库, 原子库所包含元素的最大个数为  $N^2/2$ , 其相干系数仍然为  $\mu = 1/\sqrt{N}$ 。然而, 目前对于非正交基联合而成的级联原子库还有待于进一步的研究。

## 2.5 结束语

目前, 基于过完备原子库的稀疏表示与稀疏分解所涉及的应用已扩展到很方面。神经科学的理论研究指出, 过完备表示更符合哺乳动物视觉系统的生物学背景<sup>[38]</sup>; 非线性逼近理论也给出令人信服的实例, 证明过完备系统的逼近于已知的正交基<sup>[39]</sup>; 在图像处理方面, 已有报告声称, 使用这样的信号表示式, 可将图像分割成互不相关的信号类型<sup>[40, 41]</sup>。信号处理的研究表明, 稀疏解对信号压缩和均衡都有较大改善<sup>[42~44]</sup>。

目前, 较为有效的算法是基追踪 (Basic Pursuit)<sup>[9]</sup>、匹配追踪 (Matching Pursuit)<sup>[7]</sup>及其变种<sup>[28, 29]</sup>。原子库所涉及的主要类型为: 多尺度 Gabor 函数<sup>[15]</sup>, 各向异性的精细原子<sup>[45]</sup>, Grassmannian 框架<sup>[37]</sup>, 小波和正弦函数的级联<sup>[13, 16, 20]</sup>, 多尺度脊波函数<sup>[39]</sup>等。如何实现上述算法的快速计算, 降低算法复杂度; 选择何种类型的原子构造合适的原子库或级联原子库族 (concatenate dictionaries family); 或者何种结构的信号适用于哪一类原子库以获得好的逼近, 这些问题一直是该领域研究的热点, 学者们对此已做了一些探索<sup>[12, 14~20]</sup>, 中, 以非相干原子库 (incoherent dictionaries) 为基础的一系列理论证明得到进一步改进<sup>[14, 15, 18~20]</sup>, 这些理论的证明, 反过来为 BP、MP 算法的具体实施提供了充分的理论依据。然而, 需要研究的问题仍很多, 除算法的改进和应用领域的开拓外, 非相干原子库的理论研究虽趋成熟但远非完善; 在高度非线性逼近领域, 各算法的一般性理论问题的研究才刚刚开始, 误差与逼近速率的关系仍是一个开放的话题<sup>[1, 2]</sup>。目前, 国内对此领域的理论研究已有所关注, 应用 MP 对信号进行压缩编码方面也做了一些努力<sup>[46~51]</sup>。

## 参 考 资 料

Temlyakov V. Nonlinear Methods of Approximation. IMI Research Reports, Dept. of

Mathematics, University of South Carolina, 2001, 01: 09

- [ 2 ] DeVore RA. Nonlinear Approximation. Acta Numerica, 1998, 7: 51~150
- [ 3 ] 焦李成, 谭山. 图像的多尺度几何分析: 回顾和展望. 电子学报, 2003, 12A: 1975~1981
- [ 4 ] Candes EJ, Donoho D. Curvelets - a surprisingly effective nonadaptive representation for objects with edges. Technical Report, Department of Statistics, Stanford University, 1999
- [ 5 ] Pennec E, Mallat S. Sparse Geometric Image Representation with Bandelets. Submitted to IEEE Transactions on Image Processing, 2003
- [ 6 ] Do MN, Vetterli M. The Contourlet Transform: An Efficient Directional Multiresolution Image Representation. Submitted to IEEE Transactions on Image Processing, Submitted to 2003
- [ 7 ] Mallat S, Zhang Z. Matching pursuits with time-frequency dictionaries. IEEE Trans. Signal Process, 1993, 41: 3397~3415
- [ 8 ] Neff R, Zakhor A. Very low bit rate video coding based on matching pursuits. IEEE Trans. Circuits and Systems for Video Technology, 1997, 7 (1): 158~171
- [ 9 ] Chen S, Donoho D, Saunders M. Atomic decomposition by basis pursuit. SIAM Journal on Scientific Computing, 1999, 20: 33~61
- [10] Daubechies I. Time-frequency localization operators: A geometric phase space approach. IEEE Trans. Inform. Theory, 1988, 34: 605~612
- [11] Coifman R, Wickerhauser M. Entropy-based algorithms for best-basis selection. IEEE Trans Inform. Theory, 1992, 38: 713~718
- [12] Donoho D, Huo X. Uncertainty principles and ideal atomic decompositions. IEEE Trans Inform. Theory, 2001, 47: 2845~2862
- [13] Davis G, Mallat S, Avellaneda M, Adaptive greedy approximation. Constr. Approx., 1997, 13 (1): 57~98
- [14] Elad M, Bruckstein A M. A generalized uncertainty principle and sparse representation in pairs of bases. IEEE Trans. On Information Theory, 2002, 48: 2558~2567
- [15] Feuer, Nemirovsky A. On sparse representation in pairs of bases. IEEE Trans. on Information Theory, 2003, 49: 1579~1581
- [16] Tropp J. Greed is good: Algorithmic results for sparse approximation. Texas Institute for Computational Engineering and Sciences, Tech. Rep., 2003
- [17] Donoho D, Elad M. Maximal sparsity representation via  $l_1$  minimization. the Proc.



- Nat. Aca. Sci., 2003, 100: 2197~2202
- Gribonval R, Nielsen R. Sparse decompositions in unions of bases. IEEE Trans. Inf. Th., 2003, 49: 3320~3325
- Gilbert A, Muthukrishnan S, Strauss M. Approximation of functions over redundant dictionaries using coherence. SODA'03
- Gilbert A, Muthukrishnan S, Strauss M, Tropp J. Improved sparse approximation over quasi-coherent dictionaries. ICIP'03
- Bertsekas D. Non-Linear Programming. Athena Scientific, Belmont, MA, 2<sup>nd</sup> edition, 1995
- Shrijver A. Theory of Linear and Integer Programming. John Wiley, 1998
- Tropp JA. Just relax: Convex programming methods for subset selection and sparse approximation. ICES Report 04-04, UT-Austin, February
- Wang Z, Cheung J, Xia Y, Chen J. Minimum fuel neural networks and their application to overcomplete signal representations. IEEE Trans. Circuits and systems-I: Fundamental Theory and Applications, 2000, 47: 1146~1159
- Friedman J H, Stuetzle W. Projection pursuit regression. Journal of the American Statistical Association, 1981, 76 (376): 817~823
- Huber P J. Projection pursuit. The annals of statistics. 1985, 13 (2): 435~475
- Temlyakov V. Weak greedy algorithms. Advances in Computational Mathematics, 2000, 12 (2, 3): 213~227
- DeVore R, Temlyakov V. Some remarks on greedy algorithms. Advances in Computational Math., 1996, 5: 173~187
- Pati Y C, Rezaiifar R, Krishnaprasad P S. Orthogonal matching pursuits: recursive function approximation with applications to wavelet decomposition. Proceedings of the 27th Asilomar Conference in Signals, Systems, and Computers, 1993
- Jones L.K. On a conjecture of Huber concerning the convergence of Projection Pursuit Regression. Ann-Stat., 1987, 15 (2): 880~882
- Villemoes L. Nonlinear approximation with walsh atoms. Proc. Surface Fitting and Multiresolution Methods, Chamonix 1996, Vanderbilt University Press, A. Le M'ehaut'e, C. Rabut, L. Schumaker, editors. 1997, 329~336
- Gribonval R, Vandergheynst P. Exponential convergence of matching pursuit in quasi-incoherent dictionaries. Tech. rep., IRISA, 2003
- Gribonval R, Nielsen M. Nonlinear approximation with dictionaries. II. Inverse estimates. Tech. Rep R-02-2019, 2002, Department of Mathematical Sciences,

Aalborg University, Fredrik Bajers Vej. 7G, 9220 Aalborg East, Denmark

- [34] Golubov B, Emov A, Skvortsov V. Walsh Series and Transforms. Kluwer Academic Publishers, 1991
- [35] Gribonval R, Nielsen M. Approximation with highly redundant dictionaries. Proc. SPIE'03, volume 5207: 216~227
- [36] Calderbank A, Cameron P, Kantor W, Seidel J. Z4-Kerdock codes, orthogonal spreads, and extremal Euclidean line-sets. Proc. London Math. Soc., 1997, 75 (3): 436~480
- [37] Strohmer T, Heath R. Grassmannian frames with applications to coding and communications. Appl. Comp. Harm. An. al., 2003, 14 (3): 257~275
- [38] Olshausen B, Field D. Sparse coding with an overcomplete basis set: A strategy employed by V1? Vision Research, 1997, 37: 311~325
- [39] Candes E, Donoho D. New tight frames of curvelets and optimal representations of objects with  $C^2$  singularities. Tech. rep., Stanford University, 2002, 37
- [40] Starck J, Elad M, Donoho D. Image Decomposition: Separation of Texture from Piecewise Smooth Content. SPIE meeting, 2003, San-Diego
- [41] Starck J, Candes E, Donoho D. The curvelet transform for image denoising. IEEE Transactions on Image Processing, 2002, 11 (6): 670~684
- [42] Berg A, Mikhael W. A survey of mixed transform techniques for speech and image coding. Proc. IEEE International Symposium on Circuits and Systems, 1999, 4: 106~109
- [43] DeBrunner V, Chen L, Li H. Lapped multiple basis algorithms for still image compression without blocking effect. IEEE Trans. Image Proc., 1997, 6: 1316~1321
- [44] Cotter S, Rao B. Sparse channel estimation via matching pursuit with application to equalization. IEEE Trans. Communications 2002, 50: 374~377
- [45] Vanderghelynst P, Frossard P. Efficient image representation by anisotropic refinement in matching pursuit. Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, 2001, vol.3: 1757~1760
- [46] 焦李成, 侯彪, 刘芳. 基函数网络逼近-进展与展望. 工程数学学报, 2002, 19 (1): 21~36
- [47] 廖斌, 许刚, 王裕国. 基于非抽样小波原子库的低速率视频编码. 软件学报, 2004, 15 (2): 221~228
- [48] 张文耀, 许刚, 王裕国. 基于匹配追踪的感知梯度正弦建模方法. 软件学报, 2003, 14 (3): 467~472

- [49] 尹忠科, 王建英, Pierre Vandergheynst. 在低维空间实现的基于 MP 的图像稀疏分解. 电讯技术, 2004, 44 (3): 12~15
- [50] ZHANG Chuanmei, YIN Zhongke, CHEN Xiangdong, XIAO Mingxia. Signal overcomplete representation and sparse decomposition based on redundant dictionaries. Chinese Science Bulletin, 2005, 50 (23): 2672~2677
- [51] 张春梅, 尹忠科, 肖明霞. 基于冗余字典的信号超完备表示与稀疏分解. 科学通报, 2006, 51 (6): 628~633

## 第3章 基于MP的信号与图像 稀疏分解方法

### 3.1 引言

在信号与图像的稀疏分解方法中, Matching Pursuit (MP, 匹配追踪) 算法是较早提出的。MP 算法原理简单, 算法易于理解, 算法的计算复杂度也是所有稀疏分解算法中最低的, 因此也是目前最为广泛应用的稀疏分解方法。本章以 MP 算法为例, 具体说明信号与图像的稀疏分解方法。

Mallat 和 Zhang<sup>[1]</sup>首先于 1993 年提出了信号在过完备原子库 (over-complete dictionary of atoms) 上分解的思想。通过信号在过完备库上的分解, 用来表示信号的基可以自适应地根据信号本身的特点灵活选取。分解的结果, 将可以得到信号非常简洁的表达 (即稀疏表示 Sparse representation)。而得到信号稀疏表示的过程称为信号的稀疏分解 (Sparse decomposition)。由于信号稀疏表示的优良特性, 信号稀疏表示已经被应用到信号处理的许多方面, 如信号去噪<sup>[1]</sup>、信号编码<sup>[2]</sup>和识别<sup>[3]</sup>等。其中, 在信号时频分布研究方面的应用特别值得关注<sup>[4]</sup>。

由于图像可以作为二维信号看待, 所以信号稀疏分解和稀疏表示的概念与方法, 一般情况下可以直接推广到图像的稀疏分解与稀疏表示方面, 正如 Mallat 在本章参考资料[5]中所做的那样。

当我们用信号空间理论的观点来看待一维信号和二维图像时, 它们在信号空间中都只是一个点而已, 或者说都是信号空间中的一个向量而已。所以, 从信号空间理论的观点出发, 一维信号和二维图像没有什么本质的区别, 所有的处理方法都是相同的。所以后面的内容中, 主要以信号为例来说明信号与图像的基于 MP 的稀疏分解方法。

目前, 信号和图像的稀疏分解在实际应用中很难被推广而产业化。阻碍其研究及应用发展的关键因素是信号和图像的稀疏分解的计算量十分巨大, 计算

间在现有计算条件下令人无法忍受。国内有研究人员指出,信号长度为 1024 样点时,信号的稀疏分解的难度将十分巨大<sup>[4]</sup>。当研究对象为图像时,对应计算量将更加巨大。在后面的内容中,读者将会看到,即使是作为稀疏分解基中算法复杂度最低的 MP 算法,其计算量也是相当惊人的。

本章中详细介绍 MP 算法,侧重分析其算法复杂度高的原因,后面部分章再构思降低计算复杂度的方法。

## 2 基于 MP 的信号稀疏分解

### 2.1 MP 算法基本思想

假设研究的信号为  $f$ , 信号长度为  $N$ , 则  $f \in \mathcal{H} = \mathbf{R}^N$ ,  $\mathcal{H} = \mathbf{R}^N$  是  $N$  维 Hilbert 空间。若将信号分解在一组完备正交的基上,则这组基的数目应为  $N$ 。由于正交基在由信号所组成的空间中的分布是稀疏的,从而,信号的能量在分解后将分散分布在不同的基上。这种能量分布的分散最后将导致用基的组合表量时表达的不简洁性,即信号表示不是稀疏的。非稀疏的表示,不利于信号处理,如识别和压缩等。为了得到信号的稀疏表示,基的构造必须使得基号组成的空间中足够的密。由此,基的正交性将不再被保证,所以此时的不再是真正意义上的基了,而改称为原子。由这些原子组成的集合,是过完备的,被称为过完备原子库 (over-complete dictionary of atoms)。信号在过完备原子库上的分解结果一定是稀疏的<sup>[1]</sup>。关于原子库过完备的具体概念,会在后面结合一个例子说明。

设  $\mathcal{D} = \{g_\gamma\}_{\gamma \in \Gamma}$  为用于进行信号稀疏分解的过完备原子库,  $g_\gamma$  为由参数组  $\gamma$  构造的原子。用不同的方法构造原子,参数组  $\gamma$  所含有的参数及参数个数也不相同。原子  $g_\gamma$  的长度与信号本身长度相同,但原子应作归一化处理,即  $\|g_\gamma\| = 1$ ,  $\|f\|$  是信号的范数,其定义是在  $\mathcal{H} = \mathbf{R}^N$  信号空间中信号 (该空间的一个点) 到空间坐标原点 (对应于 0 信号) 的距离。 $\Gamma$  为参数组  $\gamma$  的集合。由库的过完备性可知,参数组  $\gamma$  的个数应远远大于信号的长度,即若用  $L_D$  表示过完备库中原子的个数,则  $L_D$  应远远大于信号长度  $N$ 。MP 方法分解信号过程如下<sup>[1]</sup>: 首先,从过完备库中选出与待分解信号最为匹配的原子 (也称最佳原子) 其满足以下条件:

$$|\langle f, g_{\gamma_0} \rangle| = \sup_{\gamma \in \Gamma} |\langle f, g_\gamma \rangle| \quad (3.1)$$

$\langle f, g_{\gamma_0} \rangle$  是信号  $f$  与原子  $g_{\gamma_0}$  的内积。信号因此可以分解为在最佳原子  $g_{\gamma_0}$

上的分量和残余两部分，即为：

$$f = \langle f, g_n \rangle g_n + R^1 f \quad (3.2)$$

其中， $\langle f, g_n \rangle g_n$  是信号  $f$  在原子  $g_n$  上的分量或投影， $R^1 f$  是用最佳原子对原信号进行最佳匹配后的残余。显然， $R^1 f$  与  $g_n$  是正交的，即：

$$\|f\|^2 = |\langle f, g_n \rangle|^2 + \|R^1 f\|^2 \quad (3.3)$$

如果我们想使  $\langle f, g_n \rangle g_n$  是  $f$  的一个很好的逼近或近似，则要求剩余量  $R^1 f$  的能量最小化，从而就必须使投影值  $|\langle f, g_n \rangle|$  极大化。即要求式 (3.1) 成立。从信号空间的角度出发，投影值  $|\langle f, g_n \rangle|$  极大化意味着  $g_n$  是信号所在的  $N$  维 Hilbert 空间中和  $f$  方向最靠近的过完备原子库中的单位向量，即意味着  $g_n$  是信号所在的  $N$  维 Hilbert 空间中和  $f$  最为匹配的过完备原子库中的单位向量。从以上分析可知，这种要求就是 Matching Pursuits (MP) 算法中所谓的 matching 的含义。

在后面对原子库的过完备性的分析中我们会发现，即使是信号长度很有限，过完备原子库也是非常巨大的。如果信号长度很长，则过完备原子库的大小使得用式 (3.1) 代表的最优解问题很难求解，或根本就是不可能的。在实际实现该算法时，式 (3.1) 代表的最优解问题往往转化为以下的次优解问题，或局部最优解问题：

考虑下述选择规则，使  $|\langle f, g_n \rangle|$  几乎为最优，即

$$|\langle f, g_n \rangle| \geq \alpha \sup_{r \in \Gamma} |\langle f, g_r \rangle|, \quad 0 < \alpha \leq 1 \quad (3.4)$$

这个算法被称为 Weak Matching Pursuit，其中， $\alpha \in (0, 1]$  是一个最优性因子。针对 Weak Matching Pursuit 算法，由于不是求全局最优解，而是求次优解，所以可以利用最优化方法，特别是基于智能计算的最优化方法来求解，从而使 Weak Matching Pursuit 算法的计算复杂度较 matching pursuit 算法的计算复杂度急剧下降，使进行稀疏分解的计算成本降低，从而使实际计算成为可能。

对最佳匹配后的残余可以不断进行上面同样的分解过程，即

$$R^k f = \langle R^k f, g_n \rangle g_n + R^{k+1} f \quad (3.5)$$

其中  $g_n$  满足：

① 当采用 MP (Matching Pursuit) 方法时：

$$|\langle R^k f, g_n \rangle| = \sup_{r \in \Gamma} |\langle R^k f, g_r \rangle| \quad (3.6)$$

② 当采用 Weak Matching Pursuit 方法时：

$$|\langle R^k f, g_{\gamma_k} \rangle| \geq a \sup_{\gamma \in \Gamma} |\langle R^k f, g_{\gamma} \rangle| \quad (3.7)$$

③ 无论采用何种算法，均有：

$$\|R^k f\|^2 = |\langle R^k f, g_{\gamma_k} \rangle|^2 + \|R^{k+1} f\|^2 \quad (3.8)$$

这样的过程可以一直进行下去。所谓的 MP (matching pursuit) 中的 pursuit 定义是：只要信号的稀疏表示跟信号的近似程度不够好，即

$\|R^k f\|^2 = \|f\|^2 - \sum_{m=0}^{k-1} |\langle R^m f, g_{\gamma_m} \rangle|^2$  不够小，就要对  $R^k f$  重复进行上面的分解过程，一直到  $\|R^k f\|$  足够小为止。

结合式 (3.2)、(3.5) 及式 (3.3)、(3.8)，经过  $M$  步分解后，信号被分解为

$$f = \sum_{k=0}^{M-1} \langle R^k f, g_{\gamma_k} \rangle g_{\gamma_k} + R^M f \quad (3.9)$$

$$\|f\|^2 = \sum_{m=0}^{M-1} |\langle R^m f, g_{\gamma_m} \rangle|^2 + \|R^M f\|^2 \quad (3.10)$$

中， $R^M f$  为原信号分解为  $M$  个原子的线性组合后，用这样的线性组合表示所产生的误差。由于每一步分解中，所选取的最佳原子满足公式 (3.1 或 3.2)，所以分解的残余  $R^n f$  随着分解的进行迅速地减小。已经证明<sup>[1]</sup>，在信号长度有限的条件下（对数字信号而言，这是完全可以而且一定满足的）， $\|R^k f\|$  随  $k$  的增大而指数衰减为 0。

$\|R^k f\|$  的衰减特性依赖于原子库的特性和每一步分解中使用的最优化方法，用以下公式来描述。

1. MP 算法的逼近误差按指数级衰减：

$$\|R^k f\| \leq (1 - \alpha^2 \beta^2)^{k/2} \|f\| \quad (3.11)$$

中， $\alpha \in (0, 1]$  是 MP 分解中每一步的最优性因子，参数  $\beta$  取决于原子库的结构，它表示原子库函数捕捉任意输入信号特征的能力：

$$\sup_{\gamma \in \Gamma} |\langle f, g_{\gamma} \rangle| \geq \beta \|f\| \quad (3.12)$$

1.1.3 一般而论，用少数的原子（与信号长度相比较而言）就可以表示信号的主要成分，即

$$f \approx \sum_{k=0}^{M-1} \langle R^k f, g_n \rangle g_n \quad (3.13)$$

其中,  $M \ll N$ 。公式 (3.13) 和条件  $M \ll N$  集中体现了稀疏表示的思想。

以上就是基于 MP 的信号稀疏分解算法的描述, 公式 (3.13) 代表了信号稀疏分解的结果, 即信号的稀疏表示。实际上, 公式 (3.13) 也给出了从信号的稀疏表示重建信号的方法。一般而言, 在信号稀疏分解中, 往往关注的是信号中的稀疏成分, 而其他成分不太关注而被忽略, 所以从稀疏分解的结果重建信号, 往往只能是近似的。一定条件下从信号的稀疏分解结果也可以精确地重建原来的信号, 但这并不常用。精确重建信号的内容参考第 2 章中的内容。

### 3.2.2 MP 算法计算复杂性分析

虽然基于 MP 的稀疏分解是目前信号稀疏分解的最常用方法, 也是几乎所有算法中速度最快的, 但和其他的信号稀疏分解方法一样, 其存在的关键问题仍是计算量十分巨大。在基于 MP 的信号稀疏分解中, 每一步都要完成信号或信号分解的残余在过完备库中的每一个原子上的投影计算。按式 (3.1) 所要求, 每一步分解实际上要进行的内积计算  $\langle R^k f, g_r \rangle$  是一个在很高维 ( $N$  维) 空间的内积计算, 而且要进行很多次 ( $L_D$  次), 这是 MP 信号稀疏分解计算量巨大的根本原因所在。

为了使读者对 MP 算法的计算复杂度有一个具体的认识, 以下结合一个具体的过完备原子库来说明。

不失一般性, 引用本章参考资料[3]中过完备原子库的形成方法, 这种过完备原子库的在多个文献中被引用, 因此是颇具代表性的。过完备原子库由 Gabor 原子组成。一个 Gabor 原子由一个经过调制的高斯窗函数构成:

$$g_r(t) = \frac{1}{\sqrt{s}} g\left(\frac{t-u}{s}\right) \cos(vt + w) \quad (3.14)$$

其中,  $g(t) = e^{-\pi t^2}$  是高斯窗函数,  $\gamma = (s, u, v, w)$  是时频参数。时频参数可以按以下方法离散化:  $\gamma = (a^j, pa^j \Delta u, ka^{-j} \Delta v, i \Delta w)$ , 其中,  $a = 2$ ,  $\Delta u = 1/2$ ,  $\Delta v = \pi$ ,  $\Delta w = \pi/6$ ,  $0 < j \leq \log_2 N$ ,  $0 \leq p \leq N 2^{-j+1}$ ,  $0 \leq k < 2^{j+1}$ ,  $0 \leq i \leq 12$ 。

通过对式 (3.14) 和离散化的情况分析, 可以推导出过完备原子库  $\mathcal{D}$  中原子的个数  $L_D$ 。其计算公式如下:

$$L_D = 52(N \log_2 N + N - 1) \quad (3.15)$$

设待分解信号的长度  $N = 1024$ , 通过计算可知,  $L_D = 585\,676$ 。即当信号长



度为 1 024 时, 信号稀疏分解使用的过完备原子库中含有的原子个数为 585 676 个。当信号正交分解时, 如果信号长度为 1 024, 则基的个数也是 1 024; 而稀疏分解中, 如果信号具有同样长度, 和正交分解中基相对应的原子的个数则多达 585 676 个, 这是稀疏分解的计算复杂度远远高于正交分解的根本原因。对于一般的计算机而言, 其内存一般不能存储 585 676 个原子。生成 585 676 个原子和求取信号或信号残差在其上的分量, 对计算机的计算速度提出了极高的要求。

### 3.3 MP 算法的两大类实现方法

很多高校和科研机构 (国外如麻省理工学院、纽约大学、瑞士联邦高工和等; 国内如清华大学、电子科技大学和西南交通大学等) 对基于 MP 的信号稀疏分解进行了大量的研究, 其具体算法大致可分为两类。图 3.1 中分别给出两类算法的粗略的算法流程图。

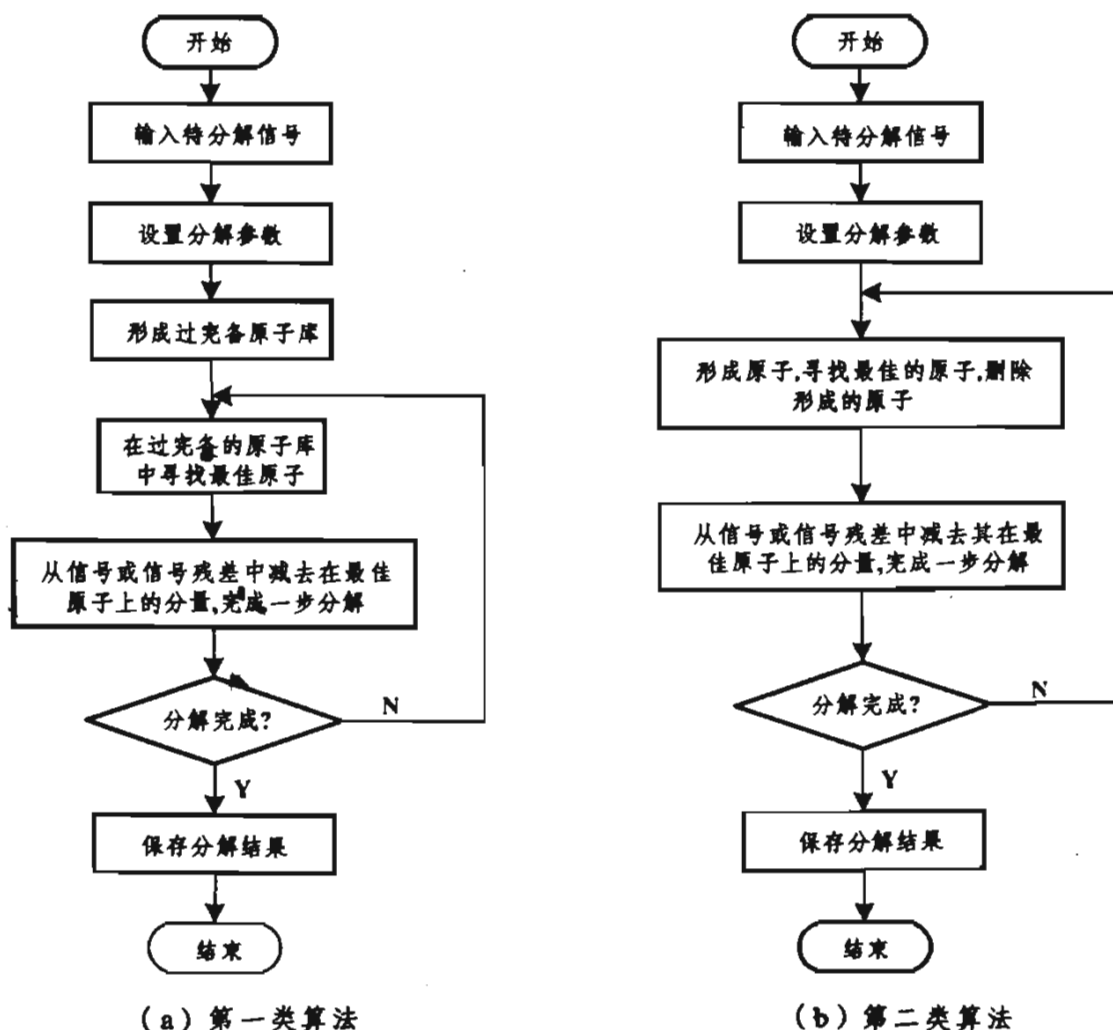


图 3.1 基于 MP 的信号稀疏分解算法流程图

我们先对此信号稀疏分解的计算流程进行说明，后面再分析两种算法的异同。我们以第一类算法为例说明如下：

① 设置分解参数。首先，为了对信号进行稀疏分解，根据信号本身的特性和信号稀疏分解的用途及要求，设计、选取进行信号稀疏分解的参数。这里参数主要是指原子库的结构、相干性、原子库中原子在信号空间分布的稠密程度，分解每一步中的最优性因子等。

② 形成过完备原子库。设定分解参数后，后续工作就是形成原子库。根据分解参数，可以形成一系列的原子，最后形成整个原子库。为了提高计算速度，在第一类算法中，往往需要把过完备原子库存储在内存中，以便在每一步分解中实现快速的内积计算。但实际上，过完备原子库往往非常巨大，一般计算机内存不能够存储这样的原子库。解决此问题的方法有两种：一种是增加内存容量，如国外部分研究机构在进行稀疏分解时，计算机的内存往往增加到数十个 G；另一种是把形成的原子库存在硬盘上，但这样的话，在每一步分解中，需要反复地从硬盘上把原子调入到内存中，计算非常耗时。

③ 在过完备的原子库中寻找最佳原子。如果采用常规 MP 算法，解公式 (3.1) 代表的最优化问题；或采用 Weak MP 算法，解公式 (3.4) 代表的次最优化或局部最优化问题。其解的结果可得到在过完备原子库中与信号或信号残差最匹配的原子。

④ 完成一步分解。利用③中寻找到的最佳原子，从信号或信号的残差中把信号或信号的残差在此原子上的分量去除，就可完成一步分解。记录下此步的最佳原子的参数，作为此步分解的结果。

⑤ 分解完成的判定。由于信号的稀疏表示是一个相对的概念，所以稀疏分解的完成可以根据不同的标准来断定。例如，可以用分解步数（对应于表示信号的原子数目）来判定，如果分解达到了一定的步数  $M$ ，就停止分解，用此

时的  $\sum_{k=0}^{M-1} \langle R^k f, g_n \rangle g_n$  来近似地表示信号  $f$ 。另外，也可以根据信号分解的残差与原子库的相干性来判定是分解是否完成。这方面的内容还将在后面的内容中涉及。

以上以第一类算法为例，介绍了 MP 分解的实现。下面我们具体分析两类算法的差别。从算法流程图可以看出，两者本质的区别在于过完备原子库的形成方法上。其中第一类算法要求在进行稀疏分解之前，先形成用于进行信号稀疏分解的过完备原子库，在整个分解的过程中，程序使用同一个已经生成的过完备原子库，可以形象地称其为“一次生成，终身使用”。在第二类方法中，不要求在稀疏分解之前形成原子库。实质上，在第二类方法中，从来没有形成

一个完整的过完备原子库。它一边生成原子，一边进行最佳原子的搜索，一边删除生成的原子。

两类方法之所以有如此大的差别，关键在于两类方法是针对不同的计算条件设计的。为了说明基于 MP 的信号稀疏分解所要求的计算条件，必须对信号稀疏分解中所要生成的过完备原子库有一个明确的概念。前面 3.2.1 节中稀疏分解的概念说明，得到信号的稀疏分解的前提条件是形成一个过完备的库，即用  $L_D$  表示过完备库中原子的个数，则  $L_D$  应远远大于信号长度  $N$ 。过完备库中原子的个数  $L_D$  的值，是非常非常大的，这使得第一类算法为了“一次生成，身使用”原子库，必须要使用巨大的计算机内存。即第一类算法相对于第二类算法，对计算机的要求是：计算机必须有巨大的内存。第二类算法实际上是对计算机内存不足但又必须进行稀疏分解的情况而设计的。由于没有生成过完备原子库，而要进行稀疏分解又必须使用过完备原子库，所以，第二类算法进行分解的每一步中，一边生成原子，一边比较搜索最佳的原子，一边还要删除生成的原子。所以第二类算法对计算机的内存没有太高的要求。但从上面分析可知，第二类算法每分解一步，实际上都产生了过完备原子库，只是没保存而已，所以第二类算法的计算量是十分惊人的。尽管第一类算法的计算也是很巨大的，但第二类算法的计算量相对第一类算法而言，还要大得多。从以上分析可知，从计算理论上讲，两者的主要区别在于对计算量和存储之间关系的不同处理上。第一类算法使用巨大的内存，第二类算法几乎对内存没有特别的要求，但计算速度很慢。两类算法在内存的使用上极不平衡，造了两者均不适合在普通的计算机上执行。找到在内存和计算速度两个方面均适合应用的方案，一直是研究的方向。在第 4 章信号与图像稀疏分解快速算中，从分析稀疏分解中过完备原子库的结构特性出发，较好地解决了此问题。

## 4 MP 算法改进算法 —— OMP 算法

MP 算法 (Matching pursuit, 匹配跟踪算法) 是目前进行信号稀疏分解的常用算法，OMP 算法 (Orthogonal Matching Pursuit, 正交匹配跟踪)<sup>[6]</sup> 是在 MP 算法基础上的一种改进算法。此算法选取最佳原子的方法和 MP 算法一样，都从过完备库中找出与待分解信号或信号残差最为匹配的原子。不同的是，MP 算法需要将所选原子利用 Gram-Schmidt 正交化方法进行正交化处理，再信号在这些正交原子构成的空间上投影，得到信号在各个已选原子上的分量残余分量；然后用相同方法分解残余分量。经过  $M$  次分解，原信号被分解为

$M$  个原子的线性组合。在每一步分解中,所选取最佳原子均满足一定条件,因此残余分量随着分解过程迅速减小,这样,用少量原子就可以表示原始信号,而且经过有限次迭代就可以收敛。

从上面的算法介绍中可以看出,MP 算法和 OMP 算法在分解过程中的区别,在于 OMP 算法在分解的每一步中都要对所选的全部原子进行正交化处理,而 MP 算法没有这样做。从算法复杂度上讲,在分解的初始阶段,OMP 算法每一步的正交化处理,并没有使它算法复杂度比 MP 算法复杂度有显著的增加。但随着分解过程的进行,分解出的原子越来越多,所有原子的正交化处理的计算量会逐渐增加。从分解效果上讲,OMP 的收敛速度比 MP 算法更快,因此,在稀疏表示精度相同的情况下,OMP 算法所选的原子应更少,即对信号的表示更加稀疏。或者说,如果用相同个数的原子表示原始信号,即稀疏性相同时,OMP 算法得到的稀疏表示的精度要高于 MP 算法得到的稀疏表示的精度。最后,对有限长信号,OMP 算法在有限步长内收敛,而 MP 算法往往较难收敛,特别是在采用 Weak Matching Pursuit 的情况下。虽然从稀疏表示和稀疏分解的概念出发,在有限步长内收敛并不重要,也没有太大的实际意义,但从数学的角度想,OMP 算法在收敛性方面显得更加完美。

图 3.2 中给出了 OMP 和 MP 算法收敛性的比较,其中信号长度为 256。原子库的形成和构造参见 3.2.2 节,最佳原子的选取过程参见第 4 章快速算法部分。

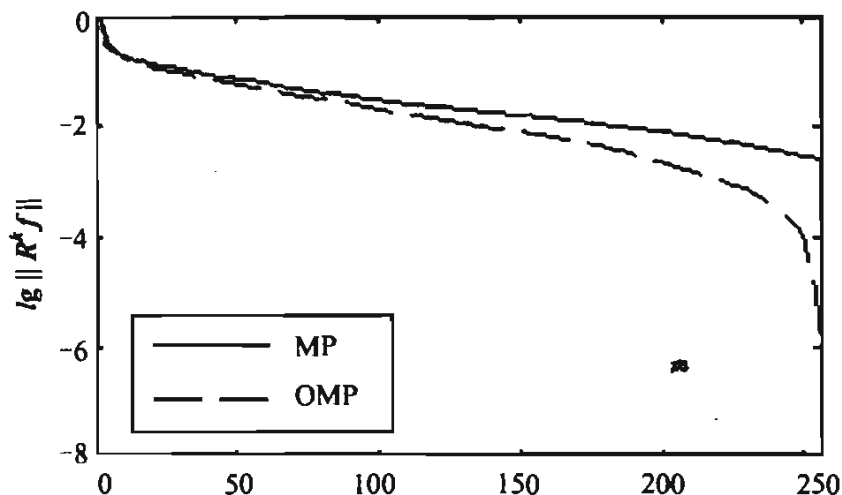


图 3.2 MP 算法与 OMP 算法收敛特性对比

#### 参 考 资 料

- [1] Mallat S, Zhang Z. Matching pursuit with time-frequency dictionaries [J]. IEEE Trans. On Signal Processing, 1993, 41 (12): 3397~3415

张文耀. 基于匹配跟踪的低位率语音编码研究[D]. 北京: 中国科学院研究生院(软件研究所), 2002

Arthur P L, Philipos C L. Voiced/unvoiced speech discrimination in noise using gabor atomic decomposition[A]. Proc of IEEE ICASSP[C]. Hong Kong: IEEE Press: 2003, I (4): 820~828

邹红星, 周小波, 李衍达. 时频分析: 回溯与前瞻[J]. 电子学报, 2000, 28 (9): 78~84

Bergeau F, Mallt S. Match pursuit of images[A]. Proceedings of IEEE-SP[C], Philadelphia, PA, USA, 1994, 330~333

Y.C. Pati, R. Rezaifar, P.S. Krishnaprasad. Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition[C]. In 27th Asilomar Conf. on signals, Systems and Comput. November, 1993

## 第4章 信号稀疏分解快速算法

### 4.1 引言

在信号分析与信号处理中，以下三个方面具有重要的研究意义：① 信号的表示方式和分解方法；② 信号分解的快速算法；③ 信号表示在信号处理中的应用。以传统的信号分析为例，DFT (Discrete Fourier Transform) 在信号分析与处理中处于核心地位，在所有的傅里叶变换中，它是最重要的一种变换。原因之一是该变换在时域和频域都是离散的，因此，可用计算机或其他数字设备进行此变换。DFT 在信号处理中处于核心地位的另一个重要原因是它存在着快速算法，即著名的 FFT 快速算法。因此可以说，如果没有 FFT 快速算法，就形不成 DFT 在信号处理中的核心地位，信号处理就不能形成十分完善的一门学科。

从前面内容可知，信号的稀疏表示与稀疏分解，是信号的一种新的、十分简洁的表示方式和分解方式，有巨大的研究价值和广阔的应用前景。但是信号的稀疏分解的计算复杂度非常高，是阻碍其发展的关键因素。为了促进信号稀疏表示及稀疏分解的研究与应用，研究其快速算法是十分必要的。否则，信号稀疏分解和信号稀疏表示也就无法实用化，而只能停留在研究阶段。

本章分析了进行信号稀疏分解时使用的过完备原子库中原子的能量特性和原子库的结构特性，提出了多种信号稀疏分解的快速算法。这些快速算法各有特点，适合不同的情况下使用，某些快速算法可以复合使用，从而极大地提高了信号稀疏分解的速度。

### 4.2 原子能量特性及在快速算法中的应用

#### 4.2.1 一个具体的过完备原子库描述

信号在过完备库上分解的计算量主要由两部分组成：形成原子（最终形成

原子库)的计算量和信号在过完备原子库上分解的计算量。

为了清楚表明信号在过完备库上分解的困难所在,必须对过完备库的大小有一个明确量化概念。不失一般性,引用本章参考资料[1]中原子库的形成方法。

Gabor 原子由一个经过调制的高斯窗函数构成:

$$g_{\gamma}(t) = \frac{1}{\sqrt{s}} g\left(\frac{t-u}{s}\right) \cos(vt + w) \quad (4.1)$$

中,  $g(t) = e^{-\pi t^2}$  是高斯窗函数,  $\gamma = (s, u, v, w)$  是时频参数, 其中  $s$  为尺度因子,  $u$  为位移因子,  $v$  为频率因子,  $w$  为相位因子。时频参数可以按以下方法离散:  $\gamma = (a^j, pa^j \Delta u, ka^{-j} \Delta v, i \Delta w)$ , 其中,  $a = 2$ ,  $\Delta u = 1/2$ ,  $\Delta v = \pi$ ,  $\Delta w = \pi/6$ ,  $j < \log_2 N$ ,  $0 \leq p \leq N 2^{-j+1}$ ,  $0 \leq k < 2^{j+1}$ ,  $0 \leq i \leq 12$ 。若用  $L_D$  表示过完备库  $D$  原子的个数, 则:

$$L_D = \sum_{j=1}^{\log_2 N} \sum_{p=0}^{N 2^{-j+1}} \sum_{k=0}^{2^{j+1}-1} \sum_{i=0}^{12} 1 \quad (4.2)$$

化简得:

$$L_D = 52(N \log_2 N + N - 1) \quad (4.3)$$

对一般长度的信号而言, 过完备库中原子个数  $L_D$  是一个很大的数。例如, 信号长度  $N$  为 256, 则  $L_D = 119\,756$ 。巨大的原子库是造成分解困难的根本原因。

## 原子形成快速算法

### (1) 原子的分类

利用原子  $g_{\gamma}(t)$  的尺度参数  $s$ , 可以把原子分成两大类, 小原子类和大原子类。取一个适当的  $s_T$ , 若  $s < s_T$ , 则  $g_{\gamma}(t)$  属于小原子类; 反之, 则  $g_{\gamma}(t)$  属于大原子类。不失一般性, 小原子和大原子的波形如图 4.1 所示。图中原子均依据其最大值做了归一化。从图 4.1 和公式 (4.1) 可知, 小原子的能量集中分布在一个小的范围 (范围:  $c = [u - t_0, u + t_0]$ ,  $u$  代表原子位移参数, 也是实际的子中心点,  $t_0 \ll N$ ,  $t_0$  值由尺度参数  $s$  决定) 内, 如图 4.1 中的区域  $[100, 156]$ 。而大原子的能量则分散分布在整个区间  $[0, N-1]$  上。

### (2) 小原子的形成算法

信号在过完备库上分解中, 小原子代表的是信号中在时域上分布比较短的成分, 即信号的细节。从图 4.1 可以看出, 小原子的能量分布在一个小的

区域中，其他区域几乎为 0。由公式 (4.1) 可知，它的波形从波形中心到周围按指数进行衰减，所以从实际有效波形考虑，除中心区域外，其他区域的  $g_{\gamma}(t)$  值可认为是 0。如果  $s_T$  选取得当，并从  $s$  计算出原子能量范围  $c$ ，此时小原子生成过程可以转换成在  $c$  内的小原子生成过程。由于  $c$  标定的范围往往小于范围  $[0, N-1]$ ，所以这样计算将减少计算量。如果  $s_T$  选取得当，这样做不会造成足以影响信号分解结果的误差。

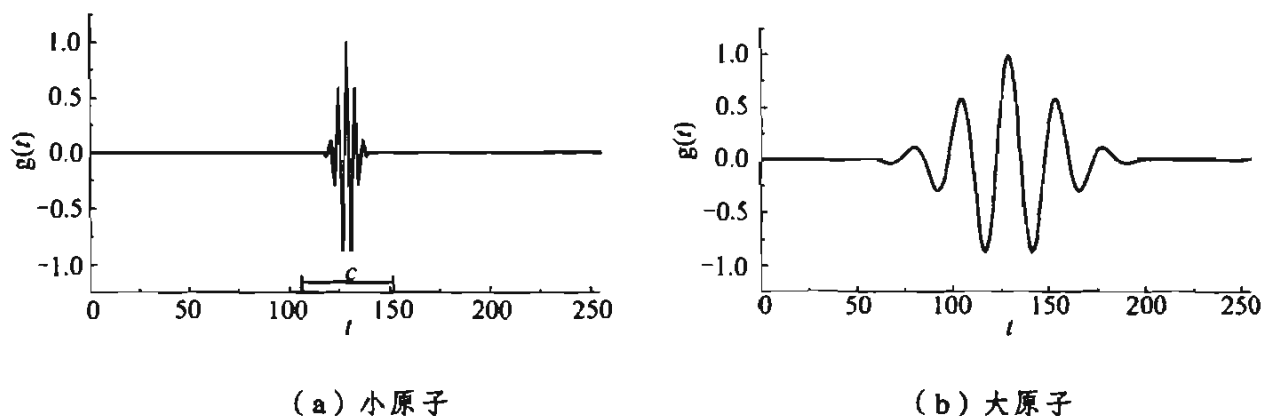


图 4.1 原子波形示意图

### (3) 大原子的形成算法

信号在过完备库上分解中，大原子代表的是信号中在时域上分布比较长的信息成分。由于大原子能量分布比较分散，所以不能用前述的小原子的形成算法。为了提高大原子形成速度，首先考查由时频参数  $\gamma=(s,u,v,w)$  和  $\gamma'=(s',u,v,w)$  定义的两个原子  $g_{\gamma}(t)$  和  $g_{\gamma'}(t)$ 。这两个原子只有尺度不一样，其他参数都一样，所以它们的波形完全一样，区别在于在时域分布上有长短。若  $s'=s/2$ ，则原子  $g_{\gamma'}(t)$  就是原子  $g_{\gamma}(t)$  的抽样。据此，提出大原子形成的快速算法，算法的思路是：要生成一个由  $\gamma=(s,u,v,w)$  定义的原子  $g_{\gamma}(t)$  时，可以先生成由  $\gamma'=(s',u,v,w)$ （其中  $s'=s/2$ ，或  $s'=s/4$ ）定义的原子  $g_{\gamma'}(t)$ ，然后通过对  $g_{\gamma'}(t)$  进行线性插值，就可生成  $g_{\gamma}(t)$ 。从公式 (4.1) 可以看出，原子形成中主要计算是指数运算和余弦运算，与此相比较，线性插值运算计算量几乎可以忽略不计。

### (4) 原子形成快速算法流程

所有原子生成的过程中，要根据不同原子的具体情况，采用不同的形成算法。只有这样，每个原子的形成速度才是最快的，从而使总的过完备原子库的形成速度达到最快。综合前述小原子和大原子的形成算法，可得出原子库中的原子形成算法流程图，如图 4.2 所示。



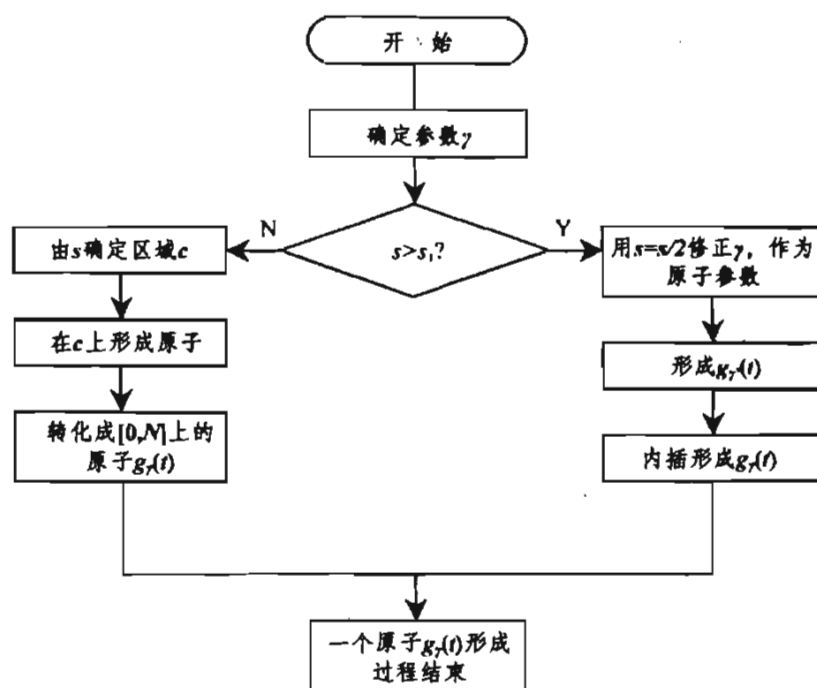


图 4.2 原子形成快速算法流程图

## 2.3 实验结果

实验中采用长度为 256、取值范围为  $[0, 255]$  的实际数字信号。过完备库中的构造方法按本章参考资料[1]。当信号长度为 256 时, 由公式 (4.2) 可知, 库含有 119 756 个原子, 以满足库的过完备性。用本文提出的快速算法生成 119 756 个原子, 比用传统算法<sup>[1, 2]</sup>的计算速度提高约 4.7 倍。

图 4.3 中给出了实际信号及在利用不同方法形成的原子库上分解为 30 个原子重建的信号, 其中分解和重建算法按本章参考资料[2]。图中用  $f(n)$  代表原始数字信号,  $f'(n)$  代表重建的数字信号, 其中,  $n$  代表数字信号  $f(n)$  或  $f'(n)$  的样点。从图中可以看出, 在利用本文快速算法形成的原子库上分解重建的信号与利用传统算法<sup>[1, 2]</sup>形成的原子库上分解重建的信号是一样的, 说明本文快速算法没有对原子的形成质量造成影响。

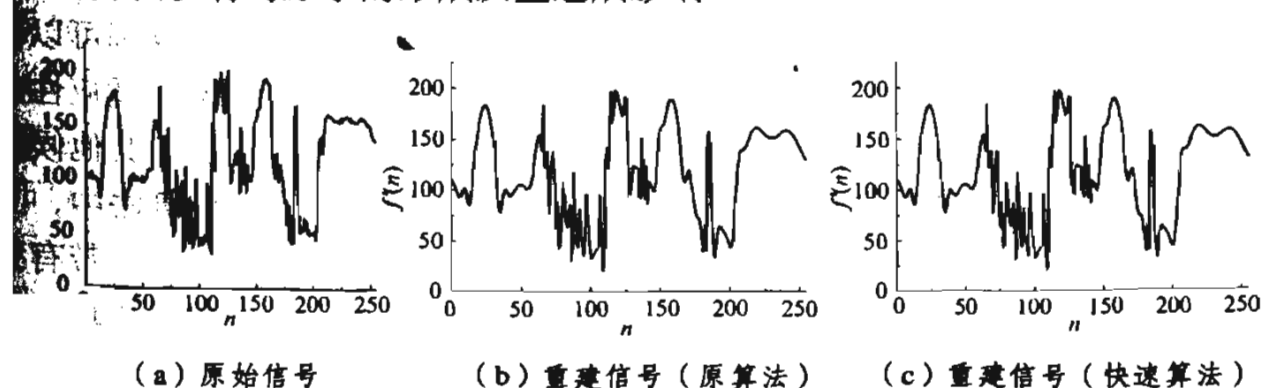


图 4.3 信号和在不同算法形成的原子库上分解重建的信号

## 4.3 基于智能计算和 MP 的信号稀疏分解

### 4.3.1 智能计算概述

随着计算机技术的飞速发展,智能计算方法的应用领域也越来越广泛。智能计算也有人称之为“软计算”,是人们受自然(生物界)规律的启迪,根据其原理,模仿求解问题的算法。从自然界得到启迪,模仿其结构进行发明创造,这就是仿生学。这是我们向自然界学习的一个方面。另一方面,我们还可以利用仿生原理进行设计(包括设计算法),这就是智能计算的思想。这方面的内容很多,如遗传算法、模拟退火算法和群集智能技术等。

#### (1) 遗传算法

遗传算法(Genetic Algorithms)是基于生物进化理论的原理发展起来的一种广为应用的、高效的随机搜索与优化的方法。其主要特点是群体搜索策略和群体中个体之间的信息交换,搜索不依赖于梯度信息。它是在 19 世纪 70 年代初期由美国 Michigan 大学的霍兰(Holland)教授发展起来的。1975 年,霍兰教授出版了第一本比较系统论述遗传算法的专著——《自然系统与人工系统中的适应性》(《Adaptation in Natural and Artificial Systems》)。遗传算法最初被研究的出发点不是为专门解决最优化问题而设计的,它与进化策略、进化规划共同构成了进化算法的主要框架,都是为当时人工智能的发展服务的。迄今为止,遗传算法是进化算法中最广为人知的算法。

近几年来,遗传算法主要在复杂优化问题求解和工业工程领域应用方面,取得了一些令人信服的结果,所以引起了很多人的关注。在发展过程中,进化策略、进化规划和遗传算法之间差异越来越小。遗传算法成功的应用包括:作业调度与排序、可靠性设计、车辆路径选择与调度、成组技术、设备布置与分配、交通问题等。

遗传算法是解决搜索问题的一种通用算法<sup>[4]</sup>,对于各种通用问题都可以使用。搜索算法的共同特征为:① 首先组成一组候选解;② 依据某些适应性条件测算这些候选解的适应度;③ 根据适应度保留某些候选解,放弃其他候选解;④ 对保留的候选解进行某些操作,生成新的候选解。在遗传算法中,上述几个特征以一种特殊的方式组合在一起:基于染色体群的并行搜索,带有猜测性质的选择操作、交换操作和突变操作。这种特殊的组合方式将遗传算法与其他搜索算法区别开来。

遗传算法还具有以下几个方面的特点<sup>[5]</sup>:

① 遗传算法从问题解的串集开始搜索，而不是从单个解开始。这是遗传算法与传统优化算法的极大区别。遗传算法从串集开始搜索，覆盖面大，利于全局优化。② 许多传统搜索算法都是单点搜索算法，容易陷入局部的最优解。遗传算法同时处理群体中的多个个体，即对搜索空间中的多个解进行评估，减少了陷入局部最优解的风险，同时算法本身易于实现并行化。③ 遗传算法基本上不用领域空间的知识或其他辅助信息，而仅用适应度函数值来评估个体，在此基础上进行遗传操作。适应度函数不仅不受连续可微的约束，而且其定义域可以任意设定。这一特点使得遗传算法的应用范围大大扩展。④ 遗传算法不是采用确定性规则，而是采用概率的变迁规则来指导它的搜索方向。⑤ 遗传算法具有自组织、自适应和自学习性。遗传算法利用进化过程获得的信息自行组织搜索时，硬度大的个体具有较高的生存概率，并获得更适应环境的基因结构。

前面描述是简单的遗传算法模型，可以在这一基本型上加以改进，使其在科学和工程领域得到广泛应用。下面列举了一些遗传算法的应用领域：① 优化：遗传算法可用于各种优化问题，既包括数量优化问题，也包括组合优化问题。② 程序设计：遗传算法可以用于某些特殊任务的计算机程序设计。③ 机器学习：遗传算法可用于许多机器学习的应用，包括分类问题和预测问题等。④ 经济学：应用遗传算法对经济创新的过程建立模型，可以研究投标的策略，模拟建立市场竞争的模型。⑤ 免疫系统：应用遗传算法可以对自然界中免疫系统的多个方面建立模型，研究个体的生命过程中的突变现象以及发掘进化中的基因资源。⑥ 进化现象和学习现象：遗传算法可以用来研究个体是学习生存技巧的，一个物种的进化对其他物种会产生何种影响等。⑦ 社会经济问题：遗传算法可以用来研究社会系统中的各种演化现象，例如在一个主体系统中，协作与交流是如何演化出来的。

## 5(2) 群体（群集）智能

受社会性昆虫行为的启发，计算机工作者通过对社会性昆虫的模拟产生了一系列对于传统问题的新的解决方法，这些研究就是群集智能的研究。群集智能（Swarm Intelligence）中的群体（Swarm），指的是“一组相互之间可以进行通信或者间接通信（通过改变局部环境）的主体，这组主体能够合作进行问题求解”。而所谓群集智能，指的是“无智能的主体通过合作表现出智能行为的特性”。群集智能在没有集中控制并且不提供全局模型的前提下，为复杂的分布式问题的解决方案提供了基础。

群集智能的特点和优点：群体中相互合作的个体是分布式的（Distributed），更能够适应当前网络环境下的工作状态；没有中心的控制与数据，这样的

系统更具有鲁棒性 (Robust), 不会由于某一个或者某几个个体的故障而影响整个问题的求解。可以不通过个体之间直接通信而是通过非直接通信 (Stimergy) 进行合作, 这样的系统具有更好的可扩充性 (Scalability)。由于系统中个体的增加而增加的系统的通信开销在这里十分小。系统中每个个体的能力十分简单, 这样每个个体的执行时间比较短, 并且实现也比较简单, 具有简单性 (Simplicity)。因为具有这些优点, 虽说群集智能的研究还处于初级阶段, 并且存在许多困难, 但是可以预言群集智能的研究代表了以后计算机研究发展的一个重要方向。

在计算智能 (Computational Intelligence) 领域有两种基于群集智能的算法, 蚁群算法 (Ant Colony Optimization) 和粒子群算法 (Particle Swarm Optimization), 前者是对蚂蚁群落食物采集过程的模拟, 该算法已经成功运用在很多离散优化问题上。

#### ① 蚁群优化算法<sup>[6]</sup>

受蚂蚁觅食时的通信机制的启发, 20 世纪 90 年代, Dorigo 提出了蚁群优化算法 (Ant Colony Optimization, ACO) 来解决计算机算法学中经典的“货郎担问题”。如果有  $n$  个城市, 需要对所有  $n$  个城市进行访问且只访问一次的最短距离。

在解决货郎担问题时, 蚁群优化算法设计虚拟的“蚂蚁”将摸索不同路线, 并留下会随时间逐渐消失的虚拟“信息素”。虚拟的“信息素”也会挥发, 每只蚂蚁每次随机选择要走的路径, 它们倾向于选择路径比较短的、信息素比较浓的路径。根据“信息素较浓的路线更近”的原则, 即可选择出最佳路线。由于这个算法利用了正反馈机制, 使得较短的路径能够有较大的机会得到选择, 并且由于采用了概率算法, 所以它能够不局限于局部最优解。

蚁群优化算法对于解决货郎担问题并不是目前最好的方法, 但首先, 它提出了一种解决货郎担问题的新思路; 其次由于这种算法特有的解决方法, 它已经被成功用于解决其他组合优化问题, 例如, 图的着色 (Graph Coloring) 以及最短超串 (Shortest Common Supersequence) 等问题。

#### ② 粒子群优化算法<sup>[7]</sup>

粒子群优化算法 (PSO) 是一种进化计算技术 (Evolutionary Computation), 由 Eberhart 博士和 Kennedy 博士发明。PSO 同遗传算法类似, 是一种基于迭代的优化工具。系统初始化为一组随机解, 通过迭代搜寻最优值, 但是并没有遗传算法用的交叉 (crossover) 以及变异 (mutation), 而是粒子在解空间追随最优的粒子进行搜索。

同遗传算法比较, PSO 的优势在于容易实现并且没有许多参数需要调整。

该算法目前已广泛应用于函数优化、神经网络训练、模糊系统控制以及其他遗传算法的应用领域。粒子群优化算法 (PSO) 也是起源对简单社会系统的模拟, 最初设想是模拟鸟群觅食的过程, 但后来发现 PSO 是一种很好的优化工具。

鸟群的捕食行为: 一群鸟在随机搜索食物, 在这个区域里只有一块食物。所有的鸟都不知道食物在哪里, 但是它们知道当前的位置离食物还有多远。那么找到食物的最优策略也即最简单有效的方法, 就是搜寻目前离食物最近的鸟的周围区域。

PSO 从这种模型中得到启示并用于解决优化问题。PSO 中, 每个优化问题的解都是搜索空间中的一只鸟。我们称之为“粒子”。所有的粒子都有一个由被优化的函数决定的适应值 (fitness value), 每个粒子还有一个速度决定它们飞翔的方向和距离。然后粒子们就追随当前的最优粒子在解空间中搜索。

PSO 初始化为一群随机粒子 (随机解), 然后通过迭代找到最优解, 在每一次迭代中, 粒子通过跟踪两个“极值”来更新自己: 第一个就是粒子本身所找到的最优解, 这个解叫做个体极值 *pbest*; 另一个极值是整个种群目前找到的最优解, 这个极值是全局极值 *gbest*。另外也可以不用整个种群而只是用其中一部分最优粒子的邻居, 那么在所有邻居中的极值就是局部极值。

PSO 算法过程如下:

① 种群随机初始化。

② 对种群内的每一个个体计算适应值 (fitness value)。适应值与最优解的精度直接有关。

③ 种群根据适应值进行复制。

④ 如果终止条件满足的话, 就停止, 否则转步骤②。

从以上步骤, 我们可以看到 PSO 和遗传算法有很多共同之处: 两者都随机初始化种群, 都使用适应值来评价系统, 而且都根据适应值来进行一定的随机选择; 两个系统都不是保证一定找到最优解。但是, PSO 没有遗传操作如交叉 (crossover) 和变异 (mutation), 而是根据自己的速度来决定搜索。粒子还有一个重要的特点, 就是有记忆。

与遗传算法比较, PSO 的信息共享机制是很不同的。在遗传算法中, 染色体 (chromosomes) 互相共享信息, 所以整个种群的移动是比较均匀的向最优区域移动。在 PSO 中, 只有 *gbest* (or *lBest*) 发出信息给其他的粒子, 这单向的信息流动。整个搜索更新过程是跟随当前最优解的过程。与遗传算法比较, 在大多数的情况下, 所有的粒子可能更快地收敛于最优解。现在已有一些利用 PSO 代替反向传播算法来训练神经网络的论文。研究表明, PSO

是一种很有潜力的神经网络算法,同时 PSO 速度比较快而且可以得到比较好的结果。

### 4.3.2 基于遗传算法和 MP 的信号稀疏分解

#### (1) 利用遗传算法寻找最佳原子及其改进

在基于 MP 的信号稀疏分解中,每一步都要完成信号或信号分解的残余在过完备库中的每一个原子上的投影计算。按式 (3.1) 所要求,每一步分解实际上要进行的内积计算  $\langle R^k f, g_r \rangle$  是一个在很高维 ( $N$  维) 空间的内积计算,而且要进行很多次 ( $L_D$  次),这是 MP 信号稀疏分解计算量巨大的根本原因所在。公式 (3.1) 所代表的问题,即 MP 每一步分解所需解决的问题,实际上是一个最优化问题,直接的求解方法是全局搜索,这将非常费时。对此问题,利用遗传算法可以很好地快速解决。

遗传算法是一种高效并行全局搜索算法<sup>[8]</sup>。由于基于 MP 的信号稀疏分解的每一步,实际上是解由公式 (2.8) 所表示的最优化问题,所以我们可以利用遗传算法来解决这个问题。把定义一个原子的参数组  $\gamma$  作为待寻优参数 (或称为遗传算法中的个体或染色体),遗传算法的编码采用直接值编码方式 (Value Encoding),信号或信号残余与原子的内积的绝对值  $|\langle R^k f, g_r \rangle|$  作为适应度函数,利用遗传算法在过完备库中选取最佳匹配原子的过程可以用图 4.4 所示的流程图来表示。

在图 4.4 所示的利用遗传算法求取最佳原子框图中,进化过程达到稳定的时候,采用生成新的初始种群的方法,来避免遗传算法求得的解为局部最优解。但是,由于进化代数的限制 (实质为计算量的限制),我们最终所求得的解仍只能是局部最优的解 (只不过程度上更趋近于全局最优解)。因此,如何在解的质量方面和计算的复杂性方面取得一个最佳的平衡,仍值得进行深入研究。

利用遗传算法寻找最佳原子,极大地减少了计算量,是对公式 (3.1) 所表示的最优化问题的一个较令人满意的解决方案。但其计算量仍是很大的,仍有必要对其进行优化。以下是我们对其进行优化的一种方法。首先具体分析利用遗传算法寻找一个最佳匹配原子的计算量:在遗传算法中,主要的计算花费在计算适应度函数上,即计算信号或信号的残余与原子的内积的绝对值 (实际上,如果要计算适应度函数,首先还要利用原子参数形成原子本身)。其他包含在遗传算法中的计算量和此计算量相比,都是微不足道的。

在遗传算法中完全避免计算适应度函数是不可能的,但为了减少计算量,我们根据遗传算法的特点,可以避免相当一部分适应度函数的计算,这对提高

最佳原子的遗传算法的计算速度是很有意义的。我们方法的依据是遗传算法中的遗传特性。具体说来就是，在遗传算法中，同一代的个体之间或相邻的个体之间，有一定的相同性。这种相同性在初始种群或进化不充分的种群中并不明显；但在进化趋向稳定的种群中，这种不同个体之间的相同性非常显著。利用这种相同性，我们把先计算的个体的适应度函数值进行存储，对于后面出现的相同个体，将不再对其进行适应度函数计算，而直接利用存储的结果。

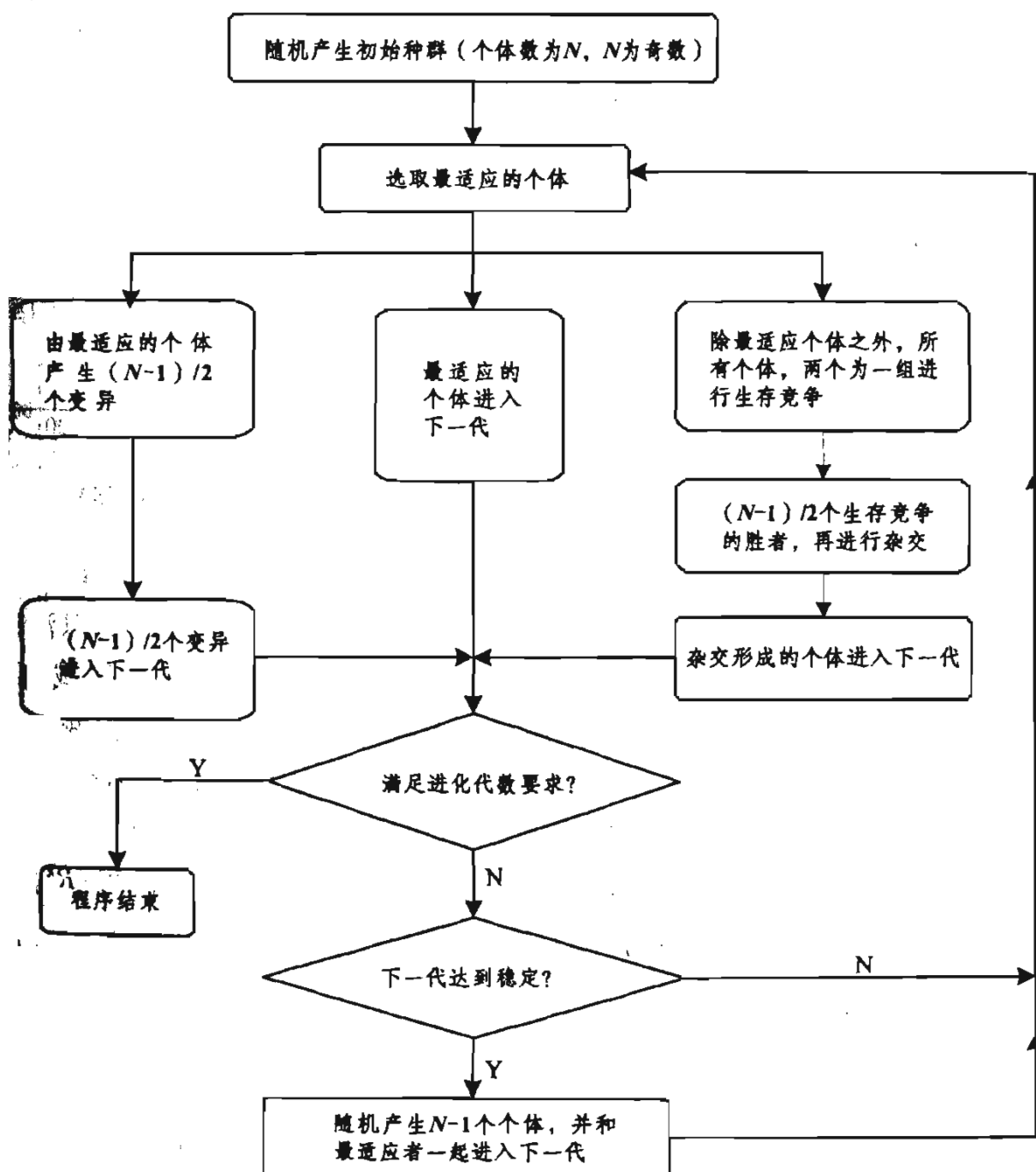


图 4.4 寻找最佳原子的遗传算法流程图

## (2) 利用原子特性对基于 GA 和 MP 的信号稀疏分解算法进行进一步优化

在基于 GA 和 MP 的信号稀疏分解过程中, 尽管我们采用了上面的算法, 以尽量避免计算作为适应度函数的信号或信号残余与原子的内积的绝对值  $|\langle R^k f, g_\gamma \rangle|$ , 但是计算内积的绝对值  $|\langle R^k f, g_\gamma \rangle|$  是无法完全避免的。当我们必须计算内积的绝对值  $|\langle R^k f, g_\gamma \rangle|$  时, 我们利用原子的特性, 可以使这一计算的计算量尽可能的小。以下我们来研究原子的特点和根据其特点减少计算  $|\langle R^k f, g_\gamma \rangle|$  的计算量的方法。

根据本章参考资料[1], 可以得到在信号稀疏分解中使用的较一般的原子。图 4.5 是一个原子的示意图, 原子长度选为 256。从图 4.5 可以看出, 在信号稀疏分解中使用的原子, 一般能量集中在原子的中央, 而原子的其他地方能量为零。这是一个有代表性的原子, 其他原子可能能量更加集中, 或能量较为分散。利用其他文献中提供的原子生成方法, 也会有相似的结论。原子能量的这种集中的特性, 是由形成原子的公式中都包含一个高斯窗函数引起的<sup>[1]</sup>。

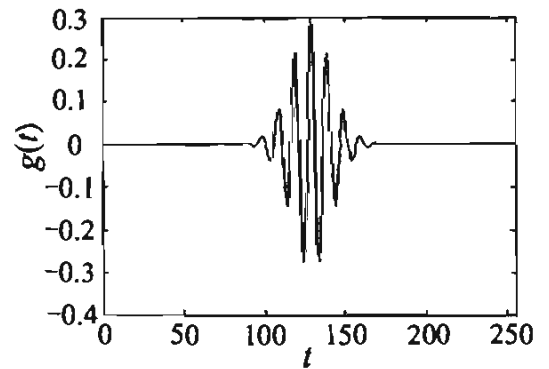


图 4.5 一个原子的示意图

利用以上原子的结构特性, 可以简化  $|\langle R^k f, g_\gamma \rangle|$  的计算。理论上讲, 内积  $\langle R^k f, g_\gamma \rangle$  的计算量随着信号长度  $L$  的增大而急剧增大。但实际上, 由于原子本身一般具有能量集中的特性, 所以没有必要按信号的长度来计算  $|\langle R^k f, g_\gamma \rangle|$ 。设信号的能量集中在长度为  $c$  的区域内, 其他部分为零。则我们只需要计算在  $c$  的区域内的内积值  $\langle R^k f, g_\gamma \rangle$ , 以此作为在信号长度  $L$  上计算的内积  $\langle R^k f, g_\gamma \rangle$  的值。只要正确确定能量集中的区域  $c$ , 则即可做到  $c$  区域内的内积  $\langle R^k f, g_\gamma \rangle$  值和信号的长度  $L$  上计算的  $\langle R^k f, g_\gamma \rangle$  值几乎完全相等。因为, 一般而言  $c$  要小于  $L$ , 所以用计算  $c$  区域内的内积  $\langle R^k f, g_\gamma \rangle$  代替计算信号的长度  $L$  上的内积  $\langle R^k f, g_\gamma \rangle$ , 则可以有效地降低计算量。

## (3) 实验结果与分析

实验中采用长度为 256、取值范围为  $[0, 255]$  的实际数字信号。过完备库中原子的构造方法按本章参考资料[1]。当信号长度为 256 时, 根据此文献, 库的大小为含有 126 490 个原子, 以满足库的过完备性。由于信号稀疏分解的速度依赖于计算条件 (硬件条件和软件条件), 所以给出实际的稀疏分解时间是没有太大的参考意义的 (几乎没有文献给出信号稀疏分解的计算时间)。我们把



基于 MP 的信号稀疏分解的速度设为 1, 表 4.1 中给出的本文中的算法的速度是  
基于 MP 信号稀疏分解的速度的倍数。

表 4.1 信号稀疏分解相对速度比较

算 法	计算速度 (倍)
基于 MP 的信号稀疏分解	1
MP + 改进的遗传算法	96*
MP + 改进的遗传算法+原子特性	128*

\*注: 由于遗传算法本身的随机性, 这里的倍数为平均值。

图 4.6 中给出了原始信号及利用不同方法稀疏分解为 30 个原子后重建的信号。可以看出, 重建信号较好地近似表示了原信号。本文所使用的方法由于基于遗传算法, 而遗传算法所求结果只能是局部最优的, 所以基于全局搜索的 MP 稀疏分解重建信号较本文方法重建的信号更接近于原始信号。但两种方法重建的信号差别并不明显。

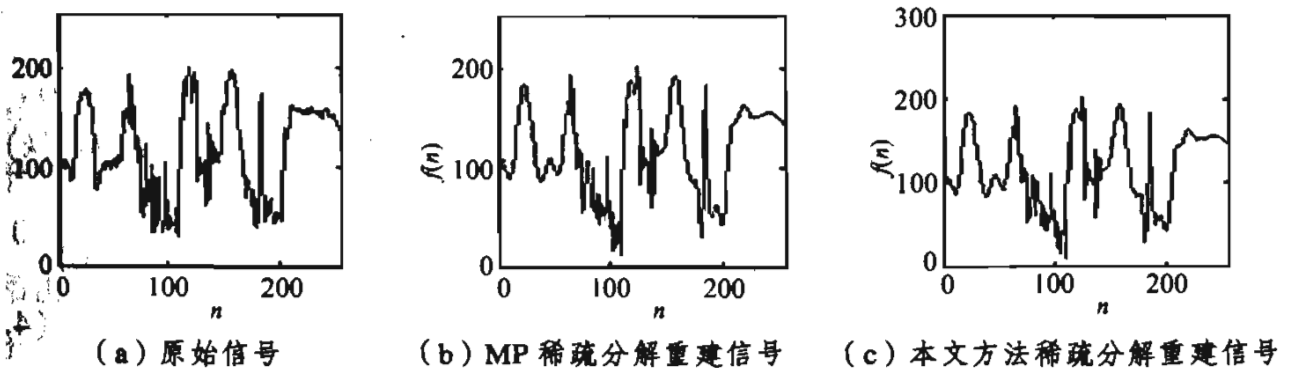


图 4.6 信号及稀疏分解后重建的信号

实验中用 30 个原子即可表示原始信号, 也足以说明这种表示的稀疏性。  
同时, 用 30 个原子即可表示原信号主要特征, 表明这种稀疏表示为进行信号  
处理 (如压缩、识别等) 提供了极其有利的条件。

### 4.3.3 蚁群算法在信号稀疏分解中的应用

#### (1) 利用蚁群算法寻找最佳原子

蚁群算法是一种新型模拟进化算法<sup>[9]</sup>。求解连续优化问题的蚁群算法思路  
为: 首先根据问题的性质估计最优解的范围, 并估计出各变量的取值范围,  
 $x_{jl} < x_j < x_{ju}$  ( $x_j$  表示第  $j$  个自变量,  $x_{jl}$  为自变量下界,  $x_{ju}$  为自变量上界)。然  
后把每个变量划分为若干个分块, 人工蚂蚁在各个分块之间移动并留下信息  
素, 以此影响后续蚂蚁的移动方向。循环一段时间后, 目标函数值大的分块信

息量较大。根据信息量，找出信息量大的分块，缩小变量范围，进行下一轮的循环<sup>[10]</sup>。经过若干次循环，最终达到稳定解。

由于基于 MP 的信号稀疏分解的每一步，实际上是解由公式 (3.1) 所表示的最优化问题，所以在过完备库中选取最佳匹配原子的过程，可转化为利用蚁群算法近似求函数最优化问题。其中，原子的参数组  $\gamma(s, u, v, w)$  作为待寻优参数，信号或信号残余与原子的内积的绝对值  $|\langle R^k f, g_r \rangle|$  作为目标函数。

若蚂蚁  $k$  当前所在分块为  $Tabu(k)$ ，则根据公式 (4.4) 选择<sup>[10]</sup>它下一步要到的分块。

$$Tabu(k) = \begin{cases} \arg(\max\{\tau_i\}), & q < Q_0 \\ S_r, & \text{其他} \end{cases} \quad (4.4)$$

其中， $q$  为随机数； $Q_0$  是一个  $[0, 1]$  上的常数，用于确定伪随机选择的概率。 $S_r$  表示用伪随机选择来确定下一步要走到的分块，也就是根据公式 (4.5)

$$p_{ij} = \frac{\tau_{ij}}{\sum_{i=1}^N \tau_{ij}} \quad (4.5)$$

来计算从第  $i$  分块到第  $j$  分块的概率， $\tau_{ij}$  理解为第  $j$  级第  $i$  个节点的吸引力，这一点与 TSP (Travel Salesman Problem) 问题是有区别的。式中  $i = Tabu(k)$ 。然后按此概率用转轮赌原则来确定要走到的分块。

蚂蚁在分块间寻优的过程中，在已到达的分块上不断地按公式 (4.6)

$$\tau_i = (1 - \rho) \times \tau_i + \rho \times \tau_0 \quad (4.6)$$

来减弱上面残留的信息，这样可减小下一只蚂蚁选择同样分块的概率，除非是经过多次循环已确定为极优的路径。这个过程称为残留信息的局部更新<sup>[11]</sup>。

当所有蚂蚁都按上面步骤完成一次循环后，这时对分块上的信息作全局更新。首先找到目标函数值最大的蚂蚁，设序号为  $k$ ，所在位置为  $Tabu(k)$ ，则对这只蚂蚁到达的分块按公式 (4.7) 作全局更新<sup>[11]</sup>：

$$\tau(Tabu(k)) = (1 - \alpha) \times \tau(Tabu(k)) + \alpha \times f(k) \quad (4.7)$$

其中， $\alpha$  为  $(0, 1)$  上的常数。至此完成了一个循环。反复进行上面的步骤直到指定的循环次数或目标函数达到稳定。当循环到一定次数或目标函数达到稳定后，取出信息素最大的分块，即为最优原子的参数  $(s, u, v, w)$ ，把它作为 MP 问题每一步的最优解。

## (2) 实验结果与分析

实验中，原始信号如图 4.7 (a) 所示，原始信号长度为 256。蚁群算法中

参数设置为： $\alpha=0.8$ ， $\rho=0.8$ ， $Q_0=0.4$ ， $\tau_0=0.01$ ， $N_c=100$ ， $Ant\_num=20$ 。

当原子数目取为 80 时，实验所得信号残差  $SNR=2.380$ 。原子数目取为 50 时，实验所得信号残差  $SNR=4.076$ 。

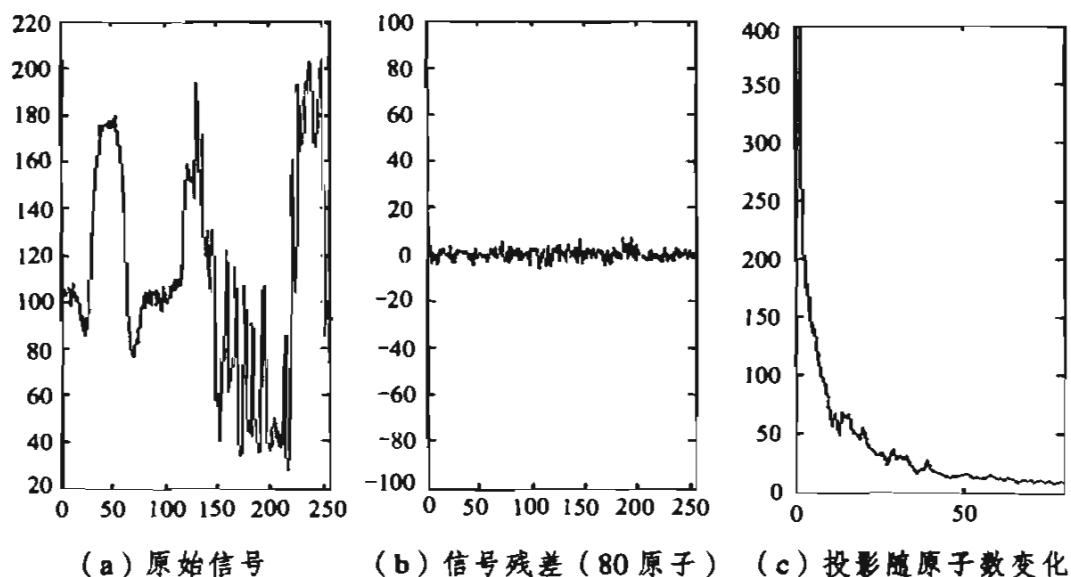


图 4.7 信号残差与投影

图 4.7 中给出了原始信号、分解 80 步时信号残差曲线与信号在最佳原子上投影绝对值曲线。通过投影曲线可看出，随着原子数目的增加，当原子数目大于 50 时，信号残差与最佳原子的内积（即投影值）已趋稳定。当原子数大于 50 时，投影衰减不明显。这从图 4.8 也可得到验证，由 50 个原子重建的信号和由 80 个原子重建的信号与原始信号很接近，只是在细节表现方面，80 个原子重建的信号略微好些。这一实验结果证明，利用蚁群算法所选取的近似最佳原子，可作为式 (3.1) 最优化问题的解，满足实际需要。

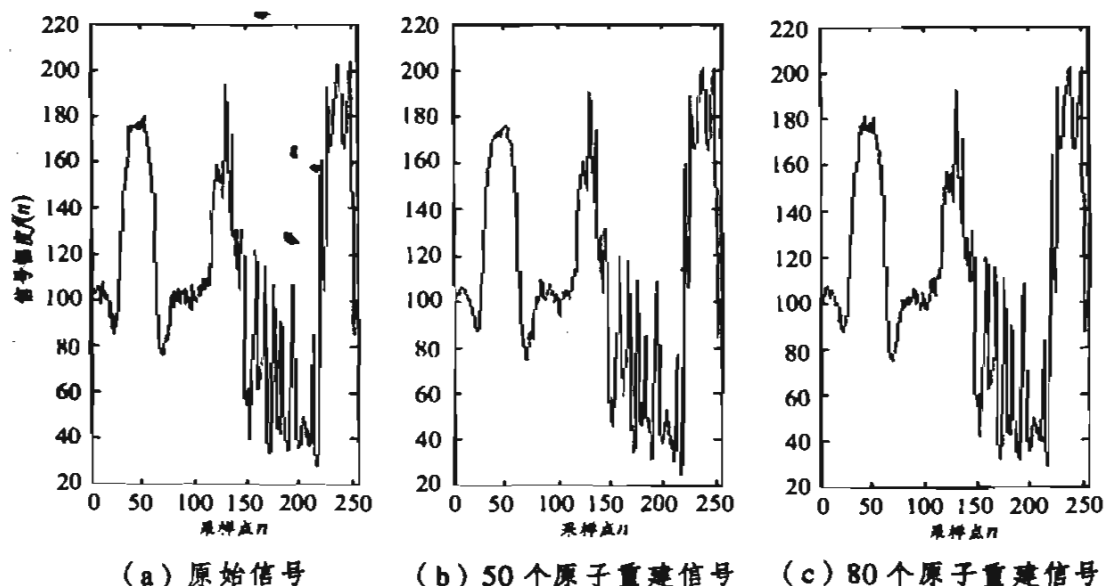


图 4.8 原始信号与重建信号

在实验中, MP 分解每一步中, 即选取一个最佳原子, 蚁群在搜索过程中每只蚂蚁需要 100 次循环, 即 100 次内积计算。本文方法蚂蚁数取为 20, 共需要 2 000 次内积计算。而一般基于 MP 的信号稀疏分解, 对于同样信号长度, 则需要 126 490 次内积计算<sup>[3]</sup>。故采取本文方法分解的速度提高了 60 倍。仿真结果也证明了以上的分析结论。因此, 利用蚁群算法可大大提高基于 MP 的信号稀疏分解速度, 为信号的压缩、识别等稀疏分解产业化应用提供了可靠的保证。

## 4.4 基于原子库集合划分和 FFT 的信号稀疏分解 MP 算法

### 4.4.1 原子库集合划分及在信号稀疏分解中的应用

信号稀疏分解中最为棘手的就是过完备原子库的形成方法、结构特性、优劣评价等, 但对过完备原子库的研究几乎没见文献报道。在本章参考资料[3]中, 我们初步分析了过完备原子库的结构特性。下面利用信号集合划分的方法<sup>[12]</sup>对其从另一个角度进行分析研究。

从数学上讲, 任何一个集合划分产生一个等价关系, 反之, 任何一个等价关系产生一个集合划分。因此, 一个集合划分就是把集合分解为互不相交的等价子集, 其中每一个等价子集是由满足等价关系的元素组成<sup>[12]</sup>。

利用信号集合划分的方法来研究过完备原子库  $\mathcal{D} = \{g_\gamma\}_{\gamma \in \Gamma}$ 。对于过完备原子库  $\mathcal{D}$ , 存在着多种不同的等价关系, 从而存在着对它的多种不同的划分方法。在稀疏分解中有用的、能够用来降低稀疏分解计算复杂度的是如下的等价关系。一个原子  $g_\gamma$  由 4 个时频参数  $\gamma = (s, u, v, w)$  决定, 其中  $s$  是伸缩因子 (尺度因子),  $u$  是原子的平移因子 (位移因子),  $v$  是原子的频率,  $w$  是原子的相位。从上面的参数  $\gamma = (s, u, v, w)$  可知: 原子  $g_\gamma$  波形由三个参数  $(s, v, w)$  决定, 而参数  $u$  (位移因子) 只决定原子的出现位置 (相当于信号中的时延)。如果把过完备库  $\mathcal{D} = \{g_\gamma\}_{\gamma \in \Gamma}$  看成是一个集合, 可以证明“波形相同”满足等价关系要求的中的三个条件 (自反性, 对称性和传递性), 因而“波形相同”是集合中元素 (原子) 之间存在的一个等价关系。因此可以利用“波形相同”这一等价关系形成过完备原子库的一个划分。

令  $(s, v, w) = \beta$ ,  $\Gamma_\beta = \{\beta_i | i = 1, 2, \dots\}$ , 则可以利用“波形相同”这一等价关

把过完备原子库  $\mathcal{D} = \{g_\gamma\}_{\gamma \in \Gamma}$  进行如下划分：

$$\left. \begin{aligned} \mathcal{D} &= \mathcal{D}_{\beta_1} \cup \mathcal{D}_{\beta_2} \cup \mathcal{D}_{\beta_3} \cdots \\ \mathcal{D}_{\beta_i} \cap \mathcal{D}_{\beta_j} &= \emptyset, \quad i \neq j \end{aligned} \right\} \quad (4.8)$$

而过完备原子库中的每一个等价子库为

$$\mathcal{D}_{\beta_i} = \{g_\gamma | \gamma = (s, u, w, v) \in \Gamma, (s, w, v) = \beta_i\} \quad (4.9)$$

从利用集合划分方法对过完备原子库进行划分的结果看，在一个等价子库所有原子的波形相同，不同原子之间的差别只是位移因子（时延）不同。对一个等价子库  $\mathcal{D}_{\beta_i}$  来说，在信号稀疏分解中只要生成和存储一个原子 [其中  $\gamma_i = (s, u = N/2, w, v)$ ,  $(s, w, v) = \beta_i$ ] 即可，等价子库中的其他原子可由  $g_{\gamma_i}$  平移来生成。而原子的平移几乎不需要额外的计算量。所以，通过对过完备  $\mathcal{D}$  的集合划分，然后用一个原子  $g_{\gamma_i}$  代表一个等价子库  $\mathcal{D}_{\beta_i}$ ，可大大降低信号稀疏分解的计算复杂度。

划分后等价子库个数（或需要生成和存储的原子数目） $L'_D$  为

$$L'_D = 13[\log_2 N + 4(N - 1)] \quad (4.10)$$

当待分解信号的长度  $N = 1024$  时， $L'_D = 53326$ 。对比前面的 4.2 节中过完备原子库中原子个数  $L_D = 585676$ ，可以看出，使用过完备原子库的集合划分，降低计算复杂度的巨大作用。

实验中采用长度为 1024、幅度变化范围为  $[0, 255]$  的实际数字信号。信号稀疏分解方法采用 Matching pursuit (MP) 方法<sup>[2]</sup>。实验结果表明，利用信号集合划分的方法对过完备原子库进行处理，可以使空间计算复杂度减少到原来 1/10 左右，和理论分析结果相一致。由于时间计算复杂度依赖于具体的信号稀疏分解算法或同一算法中不同设计方法，所以本方法对时间计算复杂度的作用也随算法的不同而改变。如果信号稀疏分解采用 MP 算法中的第二类算法<sup>[12]</sup>，本文方法可使时间计算复杂度降低到原来的 19% 左右。图 4.9 中给出实验用的原始信号；图 4.10 为没有对过完备原子库进行集合划分，直接在过完备原子库上利用 MP 进行分解后重建的信号；图 4.11 为对过完备原子库进行集合划分，在此基础上利用 MP 进行分解后重建的信号。图 4.9 和图 4.10 均是把原始信号分解在 100 个原子上。从图上可以看出，图 4.9 和图 4.10 重建信号相同，且都高度逼近实验用原始信号。说明利用集合划分方法对过完备原子库进行处理，在大大降低计算复杂度的同时，保证了信号稀疏分解质量。

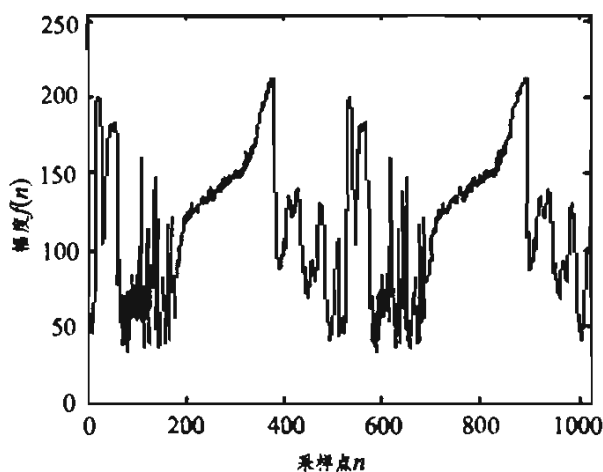


图 4.9 原始信号

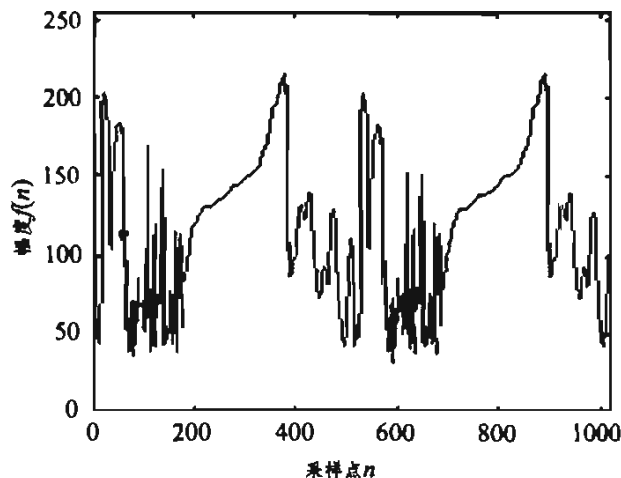


图 4.10 没有利用集合划分时  
分解后重建信号

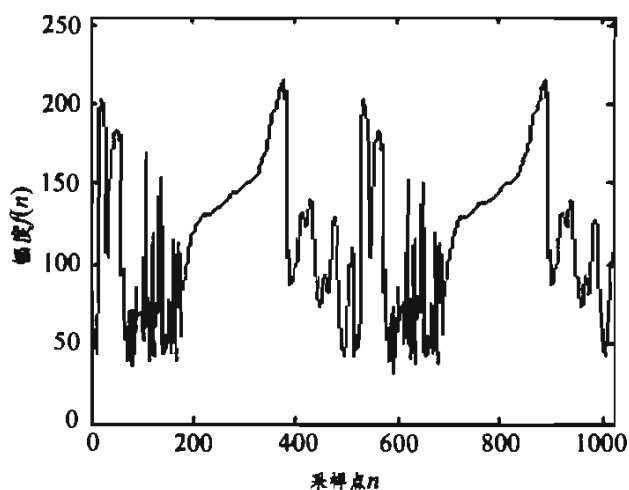


图 4.11 利用集合划分时分解后重建信号

## 4.4.2 利用 FFT 实现信号稀疏分解

### (1) 信号稀疏分解算法流程回顾

为了读者阅读方便，并方便说明、分析下面的内容，先把 MP 算法的两种实现方法部分的内容重复写在下面。两类算法的流程图见图 3.1。

两类方法之所以如此大的差别，关键在于两类方法是针对不同的计算条件设计的。从计算理论上讲，两者的主要区别在于对计算量和存储量之间关系的不同处理上。第一类算法使用巨大的内存，第二类算法几乎对内存没有特别的要求，但计算速度很慢。两类算法在内存的使用上极不平衡，造成了两者均不适合在普通的计算机上执行。

### (2) 稀疏分解中过完备原子库的结构特性

参见 4.2.1 节内容，我们可以知道，如果信号长度  $N$  为 256，则过完备原子

原子的个数为 126 490 个。对一般的计算机内存而言，这样的数据量太大，根本无法存储这样的一个原子库。只有高档的计算机才能有此存储能力。基于 MP 的信号稀疏分解的第一类算法只有高档的计算机才能实现。第二类用一般的计算机进行稀疏分解，由于无法存储如此巨大的原子库，只有解每一步时，临时生成 126 490 个原子以便选取最佳的原子，所以速度很慢。

在过完备原子库中，一个原子由 4 个参数  $(s, u, v, w)$  决定，其中  $s$  是伸缩（尺度因子）， $u$  是原子的平移因子（位移因子）， $v$  是原子的频率， $w$  是原子的相位。在稀疏分解理论中，原子库中原子的个数是一个重要的因素，为重要的一个因素是过完备原子库必须有好的结构（可惜的是，后者的几乎没有见到报道）。而原子库的结构就是由以上 4 个参数决定的。为评价一个原子库结构的好坏，我们必须定义原子库具有良好结构的含义。认为，一个原子库具有好的结构，包含两个方面的含义：一是原子库中含尽可能多的原子个数和种类，以达到稀疏分解的目的，获得稀疏分解好效果；二是原子库中应尽可能不包含相近似的原子，以满足存储量和量方面的要求。以上两个方面是相互矛盾的，前者更侧重于理论，后者重于实际计算。当二者达到一个非常好的平衡时，原子库的结构将是最

通过研究，我们发现，如果按照上面原子库良好结构的定义，现在已经发现的原子库结构都是不太合理的。以上面的由本章参考资料[1]给出的原子库为我们说明这种结构的不合理性。参数  $s$ 、 $v$  和  $w$  定义了一个原子的形状， $u$  定义了一个原子的中心位置。如果原子库中的不同原子具有相同的参数  $s$ 、 $v$ 、 $w$  和不同的参数  $u$ ，这样的原子各方面的特征都相同，只有中心点位置而已。从纯理论上讲，这种不同的原子包含在原子库中是完全必要的，实际计算的角度讲，原子库中包含众多这样的原子是根本没有必要的。它违背了上述原子库良好结构定义中的第 2 个含义。因此，一个具有良好结构的原子库中，参数  $u$  应是不变的，且  $u = N/2$ 。

### (3) 利用 FFT 实现基于 MP 的信号稀疏分解

利用信号稀疏分解中的过完备库的结构特性和 FFT 的快速计算的特性，我提出了以下 3 个算法，这 3 个算法相互关联，后面的算法以前面的算法为基础，且后面算法较前面算法速度一步步提高。

#### 算法 1

根据对具有良好结构的原子库的分析，可以构造出基于 MP 的信号稀疏

分解的快速算法。快速算法的基本思想如下：① 原子库中，原子的参数  $s$ 、 $v$  和  $w$  的选取应和其他方法一样，参数  $u$  应是不变的，且  $u=N/2$ 。这样原子库的大小将大大减小，适合一般计算机内存的要求，克服了第一类算法的缺点；② MP 稀疏分解过程：在分解的每一步，对于参数为  $s_i$ 、 $v_i$ 、 $w_i$  和  $u_i$  的原子 ( $u_i \neq N/2$ )，从原子库中取出参数为  $s_i$ 、 $v_i$ 、 $w_i$  和  $u=N/2$  的原子，通过平移，即可得到参数为  $s_i$ 、 $v_i$ 、 $w_i$  和  $u_i$  的原子 ( $u_i \neq N/2$ )。由于原子的平移几乎不需要计算量，所以，整个稀疏分解过程的速度和第一类方法几乎一样。从上面的描述可以看出，我们提出的算法，达到了计算量和存储量之间一个很好的平衡。

### 算法 2

上面的算法 1 大大提高了基于 MP 的信号稀疏分解第二类算法的速度，而稀疏分解的效果保持不变。沿着上面的思路继续思考下去，我们提出了一种新的算法，该算法不但提高了信号稀疏分解的速度，而且同时提高了信号稀疏分解的效果。

算法 2 中原子库的形成方法和算法 1 中一样，因此，存储量能够适合一般计算机的性能。在稀疏分解的过程中，对于原子库中的一个原子 (参数： $s_i$ 、 $v_i$ 、 $w_i$  和  $u=N/2$ )，让  $u_i$  取所有可能的值  $[0, N-1]$ 。这样相当于原子库的大小增加了，因此，信号稀疏分解的效果要好一些。这样做将增加计算量，但通过下面的方法，增加的计算量不会影响总体计算速度的提高。在 2.1 节中我们已经讨论过，在原子库中寻找最佳原子的过程中，主要的计算量是花费在计算内积  $\langle R^k f, g_\gamma \rangle$  上。对于具有参数  $s_i$ 、 $v_i$ 、 $w_i$  的原子库中的一个原子  $g_\gamma$ ，如果要想  $u_i$  取所有可能的值  $[0, N-1]$ ，则该原子要和信号或信号的残差作  $N$  次内积  $\langle R^k f, g_\gamma \rangle$  计算。由于  $u_i$  连续取值，且从 0 到  $N-1$ ，所有  $N$  次内积  $\langle R^k f, g_\gamma \rangle$  计算可以转换成一次  $R^k f$  和  $g_\gamma$  的互相关运算  $r_{R^k f, g_\gamma}$ 。虽然从理论上讲，这样的转换，计算量几乎没有变化，但是计算效率却大大提高。

### 算法 3

算法 3 是利用 FFT 算法对算法 2 的进一步改进。在算法 2 中，我们已经把基于 MP 的信号稀疏分解中花费绝大部分计算时间的内积  $\langle R^k f, g_\gamma \rangle$  运算，转变成了互相关运算  $r_{R^k f, g_\gamma}$ ，其中， $N$  次内积  $\langle R^k f, g_\gamma \rangle$  计算可以转换成一次  $R^k f$  和  $g_\gamma$  的互相关运算  $r_{R^k f, g_\gamma}$ 。由于利用 FFT 算法可以快速实现互相关运算，所以我们利用 FFT 算法对算法 2 进行进一步的改进。这样的改进，丝毫不会影响信号稀疏分解的效果，但却可以大大提高稀疏分解的速度。因为一般而言，用 FFT 算法实现互相关运算，可以使计算速度提高 1~2 个数量级。



#### 4.4 实验结果与分析

实验中采用长度为 256、取值范围为[0, 255]的实际数字信号。过完备库中的构造方法按本章参考资料[1]。当信号长度为 256 时, 根据此资料, 库的大小为含有 126 490 个原子, 以满足库的过完备性。由于信号稀疏分解的速度依赖于计算条件 (硬件条件和软件条件), 所以给出实际的稀疏分解时间是没有太大的参考意义的 (几乎没有资料给出信号稀疏分解的计算时间)。我们把基于 MP 的信号稀疏分解第 2 类算法的速度设为 1, 表 4.2 给出的本文中的算法速度是基于 MP 信号稀疏分解第二类算法速度的倍数。由于基于 MP 的信号稀疏分解的第一类算法需要的存储量远远超出我们实验用的计算机内存, 所以没有做关于第一类算法的实验。

表 4.2 信号稀疏分解相对速度比较

算 法	计算速度 (倍)
基于 MP 的信号稀疏分解 (第二类算法)	1
算法 1	5.2
算法 2	7.9
算法 3	29.5

从经典 FFT 理论上分析, 算法 3 应比算法 2 速度快至少 1~2 个数量级, 算法 3 的速度是基于 MP 的信号稀疏分解第二类算法速度的数百倍。但是, 如果严格按照以上方案, 实际的实验结果并不是如此。这是由于算法的特性没有很好地体现出来。我们对此问题展开分析和研究, 认为进行 FFT 计算时一定要转化成基 2 的 FFT 计算, 才能实现上面的理论计算速度。

图 4.12 中给出了原始信号及利用不同方法稀疏分解为 30 个原子后的重建信号。可以看出, 重建信号较好地近似表示了原始信号。几种分解方法重建的信号从直观上讲相差无几, 但如果用重建信号与原始信号的均方误差来衡量的话, 基于 MP 的信号稀疏分解 (第二类算法) 和算法 1, 两者重建的信号是一样的, 而算法 2 和算法 3 重建的信号完全相同, 且重建的信号更好, 即更接近于原始信号。(由于算法 2 和算法 3 重建的信号完全一样, 图中没有给算法 3 重建的信号)。

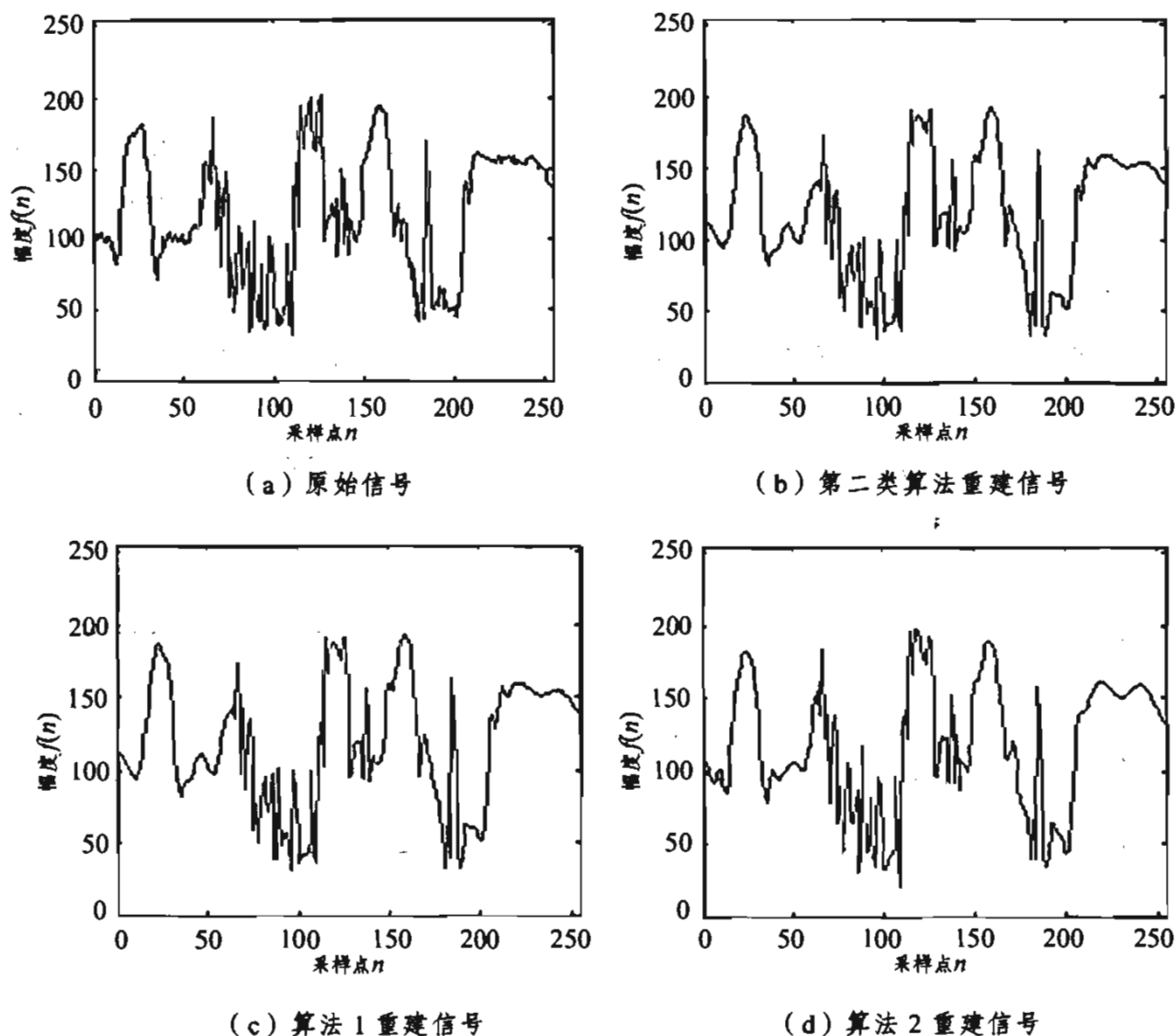


图 4.12 信号和稀疏分解后重建的信号

实验中用 30 个原子即可表示原始信号，也足以说明这种表示的稀疏性。同时，用 30 个原子即可表示原始信号的主要特征，表明这种稀疏表示为进行信号处理（如压缩、识别等）提供了极其有利的条件。

## 4.5 总 结

信号在过完备库上的分解已经被应用到信号处理的许多方面。该方法现存主要问题就是分解速度十分缓慢，因此提出的上述算法是很有意义的。现在国际上已经对信号在过完备库上的分解展开了广泛的研究，随着计算速度的提高，其理论研究和实际应用将会取得长足的进展。

在上述算法中，由于智能计算中有一定的随机性，所以基于智能计算的信

稀疏分解算法分解的结果,有时可能并不太适合某些场合应用。基于原子能特性、原子库的结构特性与集合划分、FFT 算法的信号稀疏分解算法没有随机性问题,可以用在这样的场合中。

## 参 考 资 料

- 1] Arthur P L, Philipos C L. Voiced/unvoiced speech discrimination in noise using gabor atomic decomposition[A], Proc. Of IEEE ICASSP[C], Hong Kong, April, 2003, Vol I: 820~828
- 2] Mallat S, Zhang Z. Matching pursuit with time-frequency dictionaries[J], IEEE Trans. On Signal Processing, 1993, 41 (12): 3397~3415
- 3] 尹忠科, 王建英, 邵君. 基于原子库结构特性的信号稀疏分解[J]. 西南交通大学学报, 2005, 40 (2): 174~177
- 4] 姚新, 陈国良, 徐惠敏, 等. 进化算法研究进展[J]. 计算机学报, 1995, 18 (9): 694~706
- 5] 张晓, 戴冠中, 徐乃平. 一种新的优化搜索算法—遗传算法. 控制理论与应用 [J]. 1995, 12 (3): 265~273
- 6] 李士勇, 等. 蚁群算法及其应用[M]. 哈尔滨: 哈尔滨工业大学出版社, 2004
- 7] Kennedy, J. and Eberhart, R. C, Particle swarm optimization. Proc. IEEE int'l conf. on neural networks Vol.IV, pp.1942~1948. IEEE service center, Piscataway, NJ, 1995
- 8] Davis L, Handbook of Genetic Algorithms[M], Van Nostrand, 1991
- 9] Colomi A, Dorigo M. Maniezzo V. Distributed optimization by ant colonies[C]. Proc 1st European conf artificial life. Paris, France: Elsevier, 1991, 134~142
- 10] 陈焯. 用于连续函数优化的蚁群算法[J]. 四川大学学报 (工程科学版), 2004, 36 (6): 118~120
- 11] 向敬成, 刘醒凡. 信号理论[M]. 成都: 电子科技大学出版社, 1988

## 第5章 信号稀疏分解在 信号处理中的应用

信号稀疏分解及稀疏表示是信号分解与表示方面的新方法，和传统的信号分解与表示方法有着本质的不同。稀疏分解及稀疏表示在信号处理中有着广泛的应用，已经用到诸如信号压缩编码、信号识别、信号的时频分布研究等方面。作者利用信号稀疏分解及稀疏表示进行信号去噪、微弱信号检测、阵列信号参数估计，取得了一定的成效。本章介绍作者在这些方面的工作成果。

### 5.1 稀疏分解在信号去噪中的应用

在实际的信号传输与处理中，由于信道、器件等各种因素的影响，实际信号往往含有各种不同的噪声。噪声严重影响了信号的质量，给后续的处理过程带来诸多不便，也降低了信号的使用价值。因此，对信号进行去噪处理是非常必要的，也是非常普遍的作法。本节介绍了两种基于稀疏分解的方法，即基于硬门限的稀疏分解去噪方法和基于相干比阈值的稀疏分解去噪方法。

一般的，非加性噪声可以通过某种变换转化为加性噪声，而加性噪声的消除又以消除宽带噪声最难，因此对宽带加性噪声的研究最为广泛。

加性噪声模型中，令  $f_s$  为无噪信号， $f_n$  为独立分布的 Gaussian 随机白噪声，那么含噪声的信号  $f$  可以表示为

$$f = f_s + f_n \quad (5.1)$$

从  $f$  中提取  $f_s$  的标准方法是设计一个适当的滤波器，通过滤除噪声分量  $f_n$ ，从而获得期望的信号  $f_s$ 。这样做，往往要求信号的频带与噪声的频带是分离的，但是，实际中信号和噪声的频谱往往是重叠的。为此，必须在噪声的降低与信号的失真之间进行平衡。在具体应用中，尽可能多地降低噪声是以增加

的失真为代价的；相反，如果要求期望信号尽可能地不失真，那么输出信号将不得不残留部分噪声。

常见的宽带加性噪声消除方法还有：① 最优估计或最优滤波法。这种方法包括最小二乘法、Wiener 滤波和 Kalman 滤波<sup>[1]</sup>等。② 基于小波变换的门限<sup>[2]</sup>。其基本过程是先对含噪声信号进行小波变换，运用软门限或硬门限将域中小于一定幅度的小波系数设置为 0，最后对经过门限处理后的小波系数进行逆小波变换得到去噪信号。③ 谱相减方法<sup>[3]</sup>。谱相减方法 (Spectral Subtraction Method) 主要是针对语音信号的增强发展起来的。其原理如图 5.1 所示，从带噪语音谱中减去平均噪声谱，得到降噪语音谱，然后通过逆 FFT 算得到降噪后的语音信号。

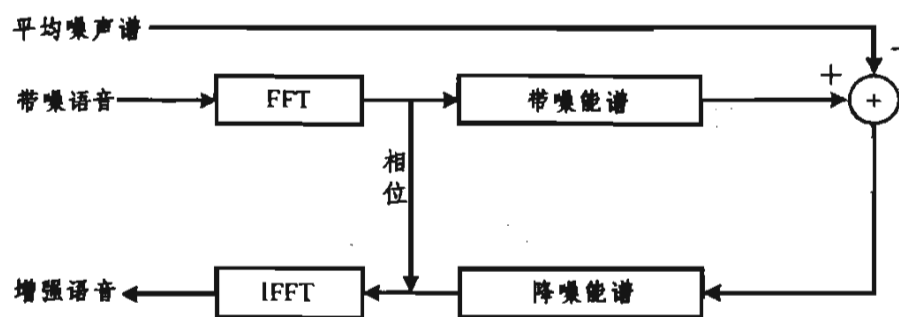


图 5.1 谱相减去噪方法的原理

以上的这些方法在各自的适应范围内都实现了信号去噪的目的，并且得到实际应用，但是它们都依赖于信号或噪声的统计特性。实际应用中往往无法获知信号或噪声的统计特性，因此这些方法的应用受到限制。

为此，人们一直在探索新的信号去噪办法。基于稀疏分解的信号去噪就是这种探索的结果之一。基于匹配跟踪的信号稀疏分解作为一种自适应的信号分解与表示方法，其最早的应用就是信号去噪。基于匹配跟踪的信号稀疏分解去噪由于不需要先验地获取信号和噪声的统计特性，并能够广泛应用于各种不同类型的信号，因而引起了研究人员的极大兴趣。下面将分析基于匹配跟踪信号稀疏分解的去噪原理、方法及其实现。

## 1.1 稀疏分解去噪的基本思想

从稀疏分解与稀疏表示的角度看，含噪声的信号包括两部分，即信号与噪声，而信号是含噪声信号中的稀疏成分。信号是有一定结构的，其结构特性与基特性相吻合。噪声是随机的，不相关的，因此是没有结构特性的。如果能从信号中提取出有意义（相对于原子的能量足够大）的原子，则提取出的部分作为信号。如果不能继续从信号残差中提取出有意义的原子，则认为信号

残差中全部是噪声。

在信号稀疏分解的迭代过程中，每次分解迭代过程都是选择与信号或信号残差  $R^k f$  内积最大的原子向量  $g_n$ 。这实际上相当于原子向量  $g_n$  与信号或残差信号  $R^k f$  相关性最大， $g_n$  最能逼近  $R^k f$ 。稀疏分解就是不断地跟踪并提取最能匹配于原始信号及其残差信号的原子向量的过程，这些被提取的原子向量可以看作是原始信号的分量，即信号。

基于稀疏分解去噪的关键是如何判定什么样的原子才是代表信号的，才是有意义的。基于 MP 的信号稀疏分解过程中，这个问题就是迭代停止条件的确定。一般来说，迭代停止的确定有两种方法，一种方法是采用硬门限，即把稀疏分解的迭代停止条件设置为一个迭代次数的上限  $M$ 。然后用  $M$  个原子的线性组合近似表示原含噪声信号，将信号的近似表示作为原含噪声信号去噪后的信号，而残差信号作为噪声信号。

另一种方法是相干比阈值法，即稀疏分解迭代停止条件设置为一个相干比的下限。下面分析基于相干比阈值的去噪方法。

在基于匹配跟踪的信号稀疏分解中，信号  $f$  在过完备原子库上分解，经过  $n+1$  次迭代后能量衰减程度  $\Delta R^{n+1}$  定义为<sup>[4]</sup>：

$$\Delta R^{n+1} = \|R^n f\|^2 - \|R^{n+1} f\|^2 = \omega^2 \|R^n f\|^2 \quad (5.2)$$

其中， $\|R^n f\|^2$  和  $\|R^{n+1} f\|^2$  分别是信号  $f$  的  $n$  阶和  $n+1$  阶残余能量； $\omega^2 = \lambda^2(R^n f)$  表示信号衰减的速率。 $\lambda(R^n f)$  称为信号  $f$ （或残差信号  $R^n f$ ）与原子库  $\mathcal{D}$  的相干比<sup>[4~6]</sup>（Coherent Ratio），简称为相干比。它取决于信号残差  $R^n f$  与原子向量  $g_r$  之间的相关性。相干比定义为<sup>[6]</sup>：

$$\lambda(R^n f) = \sup_{r \in \Gamma} \frac{|\langle R^n f, g_r \rangle|}{\|R^n f\|} \quad (5.3)$$

一般无噪信号  $f_s$  的各个分量与原子库存在很强的相干性，相干性越强，则相干比越大。而对于随机噪声  $f_n$ ，则由于原子库中不含有随机原子向量，其相干性就很弱。对  $f$  进行稀疏分解，每次提取能量最大的分量，使得残差  $R^k f$  与原子库的相干性越来越低，从而信号残差与原子库的相干比随着分解的进行越来越小。如果能够提取  $f$  中相干性大的分量而去除了相干性小的分量，则可以达到保留信号  $f_s$  而去除噪声  $f_n$  的目的。

如果被分解的信号  $f$  由原子库中少量原子信号的和构成，那么该信号及其残差信号与原子库的相干比将比较高。因为在稀疏分解的过程中将按照能量的大小与相关程度的高低依次来选择这些原子。这些被选择的分量可以看作该信

相对于给定原子库的“相干结构 (Coherent Structure)”。

## 1.2 计算机仿真

计算机仿真中, 采用本章参考资料[7]中原子参数的离散化方法, 生成原子  $\mathcal{D}$ , 对带噪信号  $f_s$  进行稀疏分解。

### (1) 基于硬门限的信号去噪

实验中, 原正弦信号的幅度为 1, 噪声为零均值、标准差为 0.1 高斯白噪如图 5.2 所示。图 5.3 采用的是小波包中标准的测试信号。对于 1024 长信, 设置迭代的上限为 50 步。

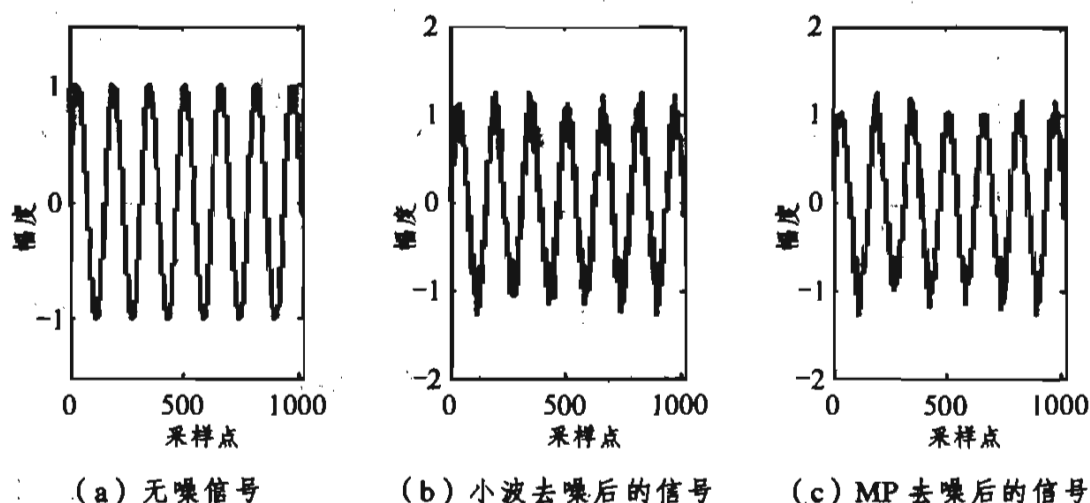


图 5.2 基于硬门限的稀疏分解去噪——正弦测试信号

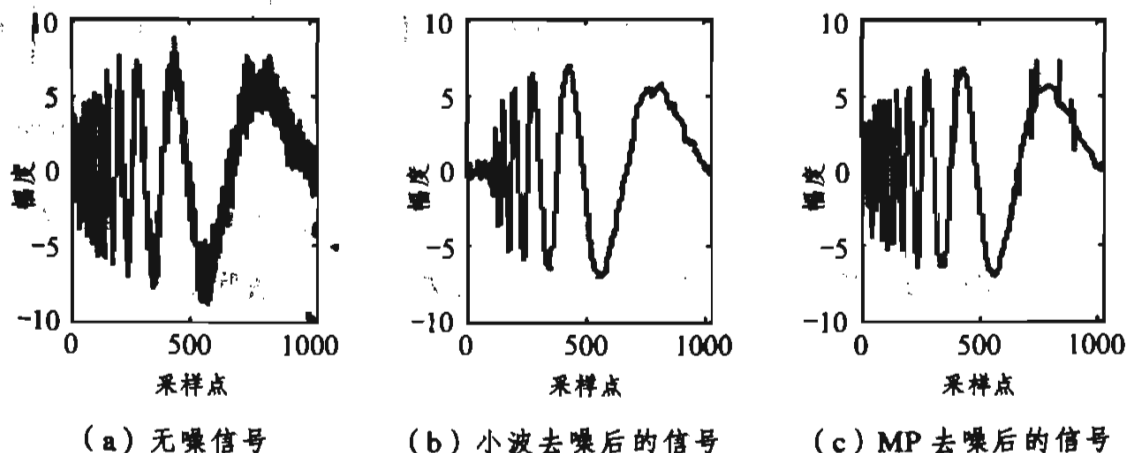


图 5.3 基于硬门限的稀疏分解去噪与小波去噪的比较

### (2) 基于相干比阈值的信号去噪

为了便于和基于硬门限的稀疏分解信号去噪方法比较, 这里的仿真信号仍采用上面的信号。迭代停止的条件为相干比阈值为 0.33。

从仿真结果可以看出，基于稀疏分解的去噪比小波分析去噪效果好，在高频部分的分辨率好，而且不需要信号和噪声的统计特性。从图 5.2 和图 5.4 可以看出，基于相干比阈值的稀疏分解信号去噪方法的效果要比基于硬门限的去噪效果好。下面将分析其原因。令：

$$Rf_s = f_s - f_{apro}$$

观察稀疏分解过程中  $Rf_s$  中的标准差  $\delta_{Rf_s}$  的变化过程（如图 5.6 所示），可以发现： $\delta_{Rf_s}$  开始随着迭代分解次数的增加而逐渐递减，在达到某个最小值  $\delta_{min}$  之后逐渐上升，最后趋于噪声的标准差  $\delta_{f_n}$ 。

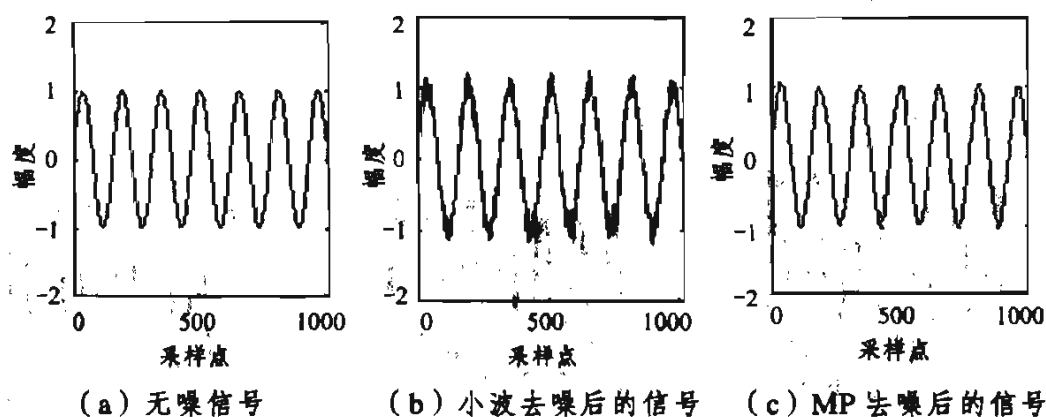


图 5.4 基于相干比阈值的稀疏分解去噪

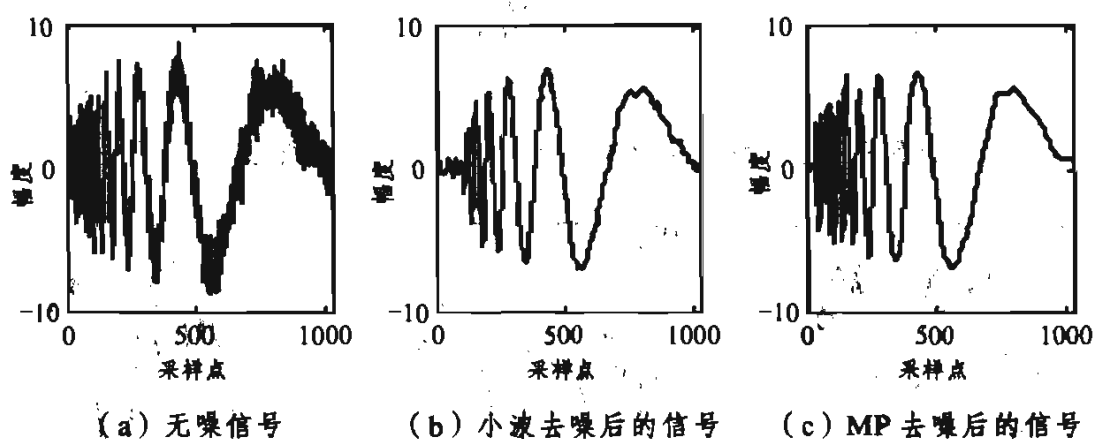


图 5.5 基于相干比阈值的稀疏分解去噪与小波去噪比较

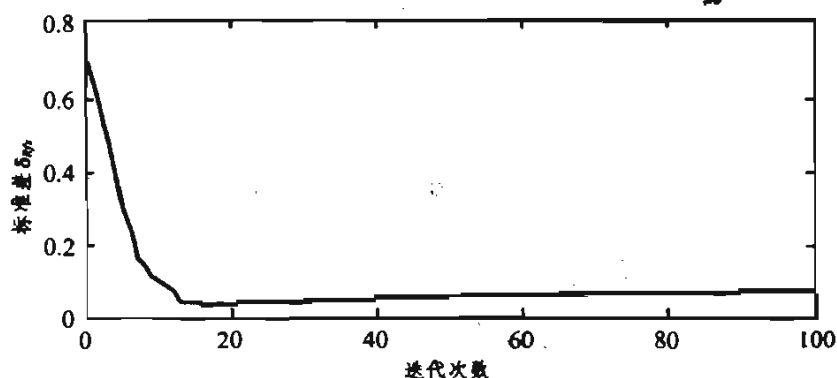


图 5.6  $\delta_{Rf_s}$  变化曲线



选取信噪比 (SNR) 作为衡量标准, 将  $f_{apro}$  看作是对信号的逼近, 那么当  $\delta = \delta_{\min}$  时, 信噪比达到最大值:

$$SNR_{\max} = 20 * \log_{10} \left( \frac{\delta_{Rf_i}}{\delta_{\min}} \right)$$

设此时所对应的迭代次数为  $n$ , 那么在第  $n$  次分解所对应的点将是这个分解迭代过程中的最佳降噪点。如果能够在未知  $f_s$  和  $f_n$  的情况下找到最佳降噪点, 那么就可以达到自适应的去噪目的。为寻找最佳的降噪点, 跟踪相干比的变化过程如图 5.7 所示。

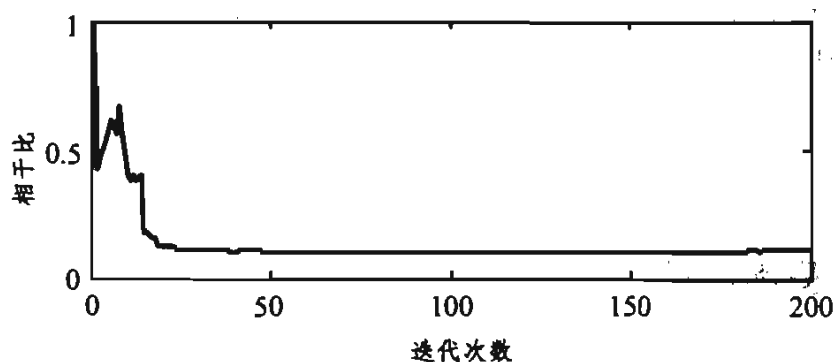


图 5.7 相干比的变化曲线

从图 5.7 中可以看到, 从最佳降噪点附近开始, 相干比  $\lambda(R_n f)$  呈现一种平缓的水平波动, 而且波动的幅度在比较小的范围内。这是因为在后阶段的分解过程中, 残差信号类似于噪声信号, 而噪声信号是无规律的, 与原子库的相干比最低。我们就可以把最佳去噪点之前的匹配跟踪分解看作是对信号  $f_s$  的分解, 之后则是对噪声  $f_n$  的分解。若在此时迭代终止, 则可以实现从带噪的  $f$  中提取期望的信号  $f_s$  的目的。

### (3) 相干比阈值的确定

通过前面的分析可知, 在基于稀疏分解的去噪算法中, 最主要的问题是关于相干比阈值的确定。通过对本章参考资料[5]和[8]及作者的仿真分析, 作者认为相干比的判决阈值是由原子库决定的。对于固定的原子库, 该值应该是一个合适的常量。原子库的构造灵活多样, 如果原子库确定了, 就可以通过一些已知的信号或构造的测试信号, 去确定最佳降噪点, 确定一个合适的相干比阈值。

本节仿真结果证明, 用相干比阈值作为迭代停止的原则对信号进行去噪有如下优点: 随迭代次数的增加, 该比值在分解前部分呈下降趋势, 有最小值; 不受信号幅值的影响; 对于相同的原子库信号, 相干比阈值一致; 判断本身不需要增加计算量 ( $\langle R_n f, g \rangle$ ,  $\|R_n f\|$  均在迭代中计算), 反而 (作为一个好的迭

代停止原则) 还能减少迭代的次数, 达到降低计算量的目的。

## 5.2 稀疏分解在微弱信号检测中的应用

由于器件本身噪声和各种电磁干扰的存在, 使得原本微弱的信号完全淹没在噪声中。在强噪声背景下提取微弱信号一直是信号处理的重点和难点。微弱信号检测<sup>[9, 10]</sup>在通信、雷达、声纳等领域有着重要的意义, 因此, 对微弱信号的检测已成为迫切需要解决的问题。本节将信号稀疏分解思想应用于信号检测, 通过对正弦波模型伸缩和平移形成过完备原子库, 将接收信号在原子库上作稀疏分解, 由 MP 分解结果, 检测出淹没在强噪声环境中的微弱正弦信号的幅度、频率和初相位参数, 从而恢复出待检测的微弱正弦信号。本节所述方法在 -40 dB 极低信噪比环境下可以同时检测多个正弦信号。

### 5.2.1 算法基本思想

假设接收到淹没在噪声中的  $K$  个正弦信号

$$x(t) = \sum_{k=1}^K \alpha_k \cos(2\pi f_k t + \phi_k) + n(t) \quad (5.4)$$

其中,  $\alpha_k$ 、 $f_k$  和  $\phi_k$  ( $k=1, 2, \dots, K$ ) 分别为第  $k$  个正弦信号的幅度、频率和初相位, 信号个数  $K$  已知;  $n(t)$  为加性高斯白噪声。

我们的目的是从接收信号  $x(t)$  的  $N$  个采样数据  $x(n)$  中, 估计出  $K$  个正弦信号的幅度、频率和初相位, 从而恢复出淹没在强噪声中的  $K$  个正弦信号。

设信号如式 (5.4) 所示, 采样信号长度为  $N$ , 采样信号用  $x(n)$  表示, 当讨论在信号空间进行时, 采样信号用  $x$  表示。  $\mathcal{D} = \{g_\gamma\}_{\gamma \in \Gamma}$  为用于进行信号稀疏分解的过完备库,  $g_\gamma$  为由参数组  $\gamma$  定义的原子, 用不同方法构造原子, 参数组  $\gamma$  所含有的参数及参数个数也不一样。原子应作归一化处理, 即  $\|g_\gamma\| = 1$ 。  $\Gamma$  为参数组  $\gamma$  的集合。MP 方法分解信号过程如下<sup>[6]</sup>:

首先, 从过完备库中选出与待分解信号  $x$  最匹配的原子  $g_{\gamma_0}$ , 使其满足

$$|\langle x, g_{\gamma_0} \rangle| = \sup_{\gamma \in \Gamma} |\langle x, g_\gamma \rangle| \quad (5.5)$$

因此, 信号  $x$  可以分解为在最佳原子  $g_{\gamma_0}$  上的分量和残余两部分, 即

$$x = \langle x, g_{\gamma_0} \rangle g_{\gamma_0} + R^1 x \quad (5.6)$$

其中,  $R^1 x$  是用最佳原子对原信号进行最佳匹配后的残余。对最佳匹配后的残余可以不断进行上面同样的分解过程, 即

$$R^i x = \langle R^i x, g_{\gamma_i} \rangle g_{\gamma_i} + R^{i+1} x \quad (5.7)$$

其中  $g_{\gamma}$  满足

$$|\langle R^i x, g_{\gamma_i} \rangle| = \sup_{\gamma \in \Gamma} |\langle R^i x, g_{\gamma} \rangle| \quad (5.8)$$

由公式 (5.6) 和 (5.7) 可知, 经过  $L$  步分解后, 信号被分解为

$$x = \sum_{i=0}^{L-1} \langle R^i x, g_{\gamma_i} \rangle g_{\gamma_i} + R^L x \quad (5.9)$$

由于  $\|R^L f\|$  的衰减特性, 一般而论, 用少数的原子 (相对信号长度而言) 就可表示信号的主要成分, 即

$$x \approx \sum_{i=0}^{L-1} \langle R^i x, g_{\gamma_i} \rangle g_{\gamma_i} \quad (5.10)$$

其中  $L \ll N$ 。式 (5.10) 和条件  $L \ll N$  集中体现了信号稀疏表示和信号稀疏分解的思想。

根据信号模型 (5.4) 式, 设计如下的过完备原子库:

$$g_{\gamma} = \cos(2\pi f_m n + \phi_j) \\ (n=1, 2, \dots, N; m=1, 2, \dots, M; j=1, 2, \dots, J) \quad (5.11)$$

其中,  $f_m$  为频率参量, 按照需要的搜索精度均匀取值;  $M$  为频率搜索个数, 在同样的频率搜索范围内, 搜索精度越高,  $M$  越大;  $\phi_j \in (-\pi, \pi)$  为相位参量, 同样按照需要的精度均匀取值。整个库的原子数为  $MJ$  个。基于这  $MJ$  个原子的原子库作 MP 分解可以求得  $K$  个正弦信号的频率  $f_k$ 、相位  $\phi_k$  和幅度  $\alpha_k$  ( $k=1, 2, \dots, K$ ), 从而提取混杂在噪声中的  $K$  个正弦信号。

由于信号稀疏分解的特性, 正弦信号在进行稀疏分解时, 将在某一分量上达到最大匹配。而噪声不具有正弦信号的特征, 因此在这一分量上投影接近零, 从而达到滤除噪声的目的。参照原子的基本参数, 可以估计得到待检测正弦信号的参数。基于这种思想, 将淹没在噪声中的多个正弦信号  $x(t)$  进行稀疏分解, 将会在不同分量上得到最大匹配, 根据这些与信号最大匹配的原子参数, 可估计出每个正弦信号的频率、初相位和幅度, 从而恢复出多个正弦信号。

设经过 MP 分解, 从过完备库中选出与待分解信号  $x$  最为匹配的原子  $g_{\gamma}$ , 即

$$\langle x, g_{\gamma} \rangle = \sup_{\gamma \in \Gamma} |\langle x, g_{\gamma} \rangle| \quad (5.12)$$

其中,  $g_n = \cos(2\pi f_k n + \phi_k)$ , 且  $x$  在  $g_n$  上的分量为:

$$\alpha_k = \langle x, g_n \rangle \quad (5.13)$$

则这一接收信号中包含正弦信号:

$$s_k(n) = \alpha_k \cos(2\pi f_k n + \phi_k) \quad (5.14)$$

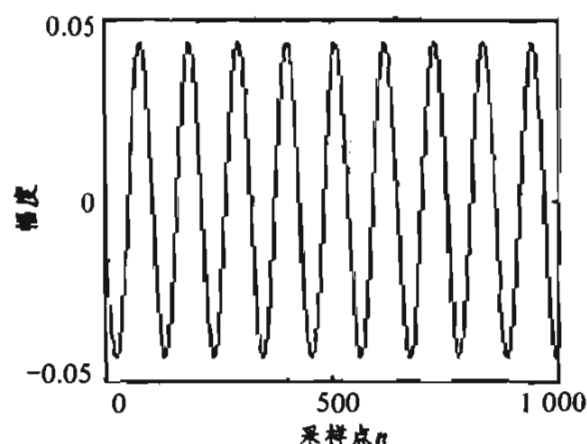
如果接收信号中正弦信号的个数  $K$  已知, 则找出前  $K$  个最匹配的原子  $g_n, \dots, g_{n_K}$ , 即可恢复出  $K$  个正弦信号。

信号 MP 分解必须在巨大的原子库中逐个搜索与信号最匹配的原子, 这就决定了 MP 分解巨大的运算量。但 MP 分解有别于其他信号分解方法的地方是其原子库可以随着信号特性及分解过程自适应地改变, MP 分解的这种特性决定了算法可以更灵活, 可以应用快速算法。这里以频率为例说明文中所用的快速算法。假设频率的搜索范围是  $0 \sim 100$  Hz, 而搜索精度是  $0.1\%$  Hz, 则搜索量为  $10^5$ 。文中所用的算法是, 先在  $0 \sim 100$  Hz 中每隔  $1$  Hz 搜索一次, 确定进一步搜索的初步范围。然后在相应的范围中做更加精细的搜索, 每隔  $0.1$  Hz 搜索一次, 从而求得小数点后第一位。依此类推, 直到求得相应的精度。以文中的精度为例, 采用这种算法计算量约为基本算法的  $0.2\%$ 。其他参量也可以采用类似的快速算法。

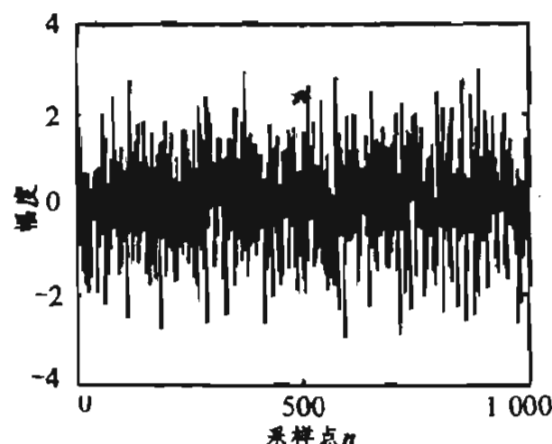
## 5.2.2 计算机仿真

### (1) 单个正弦信号的情况

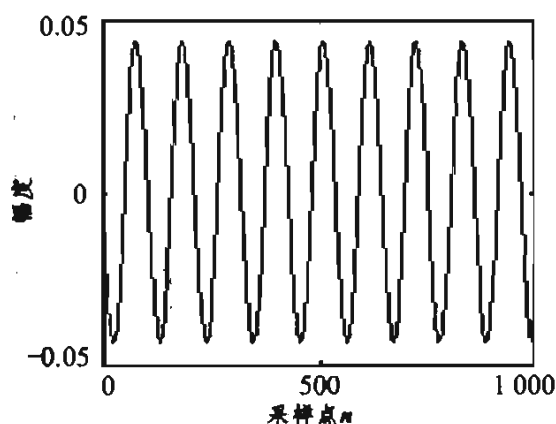
实验中采用单个正弦信号, 其幅度、频率和初相位分别为  $\alpha = 0.0434$ ,  $f = 9.2510$  Hz 和  $\phi = 0.5\pi$ 。噪声为零均值高斯白噪声。图 5.8 为信噪比为  $-40$  dB 时的仿真结果。



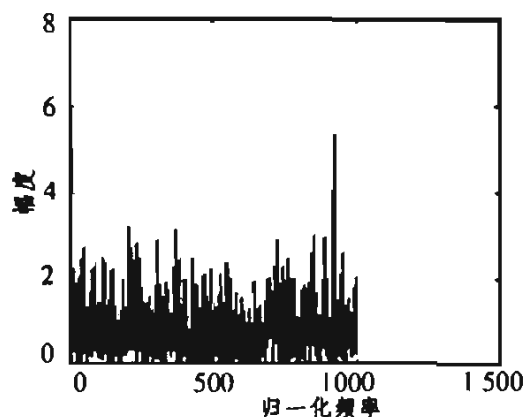
(a) 原始信号



(b) 带噪声信号



(c) 恢复信号



(d) MP 分解系数

图 5.8 单个正弦信号提取结果

信噪比从  $-40$  dB 到  $-50$  dB，每隔  $2$  dB 进行  $1\,000$  次实验，表 5.1 为不同信噪比时的检测概率。

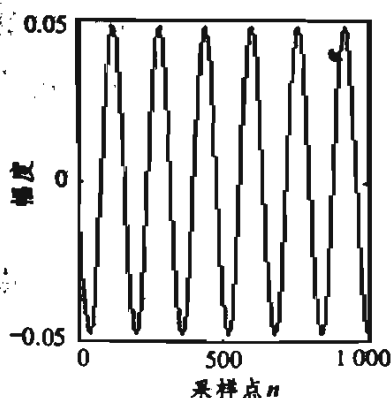
表 5.1 不同信噪比时的检测概率

信噪比 (dB)	-40	-42	-44	-46	-48	-50
识别率 (%)	100.0	99.2	91.6	68.5	44.5	16.3

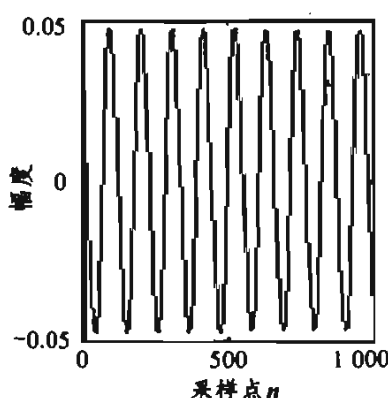
## (2) 多正弦信号的情况

两个正弦信号参数分别为  $\alpha_1 = 0.043\,9$ 、 $f_1 = 6.251\,0$  Hz、 $\phi_1 = 0.5\pi$  和  $\alpha_2 = 0.043\,9$ 、 $f_2 = 9.251\,0$  Hz、 $\phi_2 = 0.3\pi$ 。噪声为零均值高斯白噪声，信噪比  $-40$  dB。仿真结果如图 5.9 所示。

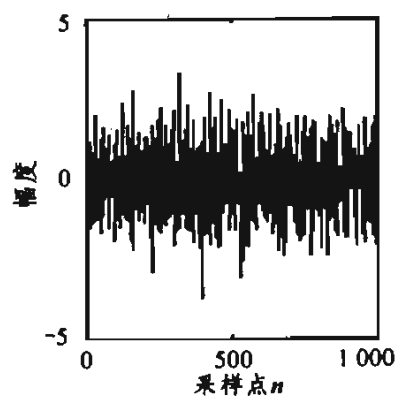
从仿真结果可以看出，在信噪比大于  $-40$  dB 以上时，算法可以可靠精确地同时恢复出单个或多个正弦信号。在信噪比小于  $-42$  dB 时，恢复效果开始下降，但在  $-44$  dB 以上时，仍可较好地恢复出原正弦信号；当信噪比降到  $-46$  dB 以下时，识别概率迅速下降，到  $-50$  dB 以下，算法失效。



(a) 原始信号 1



(b) 原始信号 2



(c) 带噪声输入信号

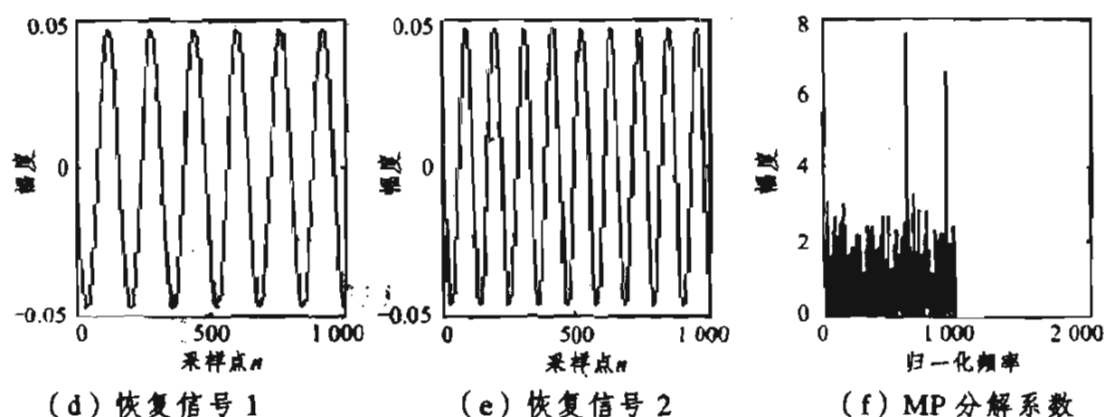


图 5.9 多个正弦信号提取结果

## 5.3 稀疏分解在阵列信号处理中的应用

### 5.3.1 频率估计算法

精确估计淹没在加性噪声中的正弦波频率是信号处理的主要内容之一，在雷达、通信和声纳等领域有广泛的应用。自 20 世纪 50 年代末开始，学者们对频率估计的问题进行了广泛研究，提出了多种算法，如取样自相关法<sup>[11]</sup>、最大似然法<sup>[12]</sup>、傅里叶分解法<sup>[13]</sup>、Pisarenko 谐波分解以及基于特征分解等<sup>[14~26]</sup>的频率估计算法等。

以上这些方法经过多次改进，在信号环境理想的情况下取得了较好的频率估计精度。但其中多数算法都基于信号正交分解，即离散傅里叶分解 (DFT)，DFT 不可逾越的鸿沟是其固有的最小频率分辨率，这个特性决定了这类算法不可能有很高的频率分辨率，尤其在采样数据少的情况下，这个矛盾尤为突出。因此，寻求一种可以克服最小频率分辨率的算法成为学者们不断追求的目标。信号稀疏分解为这个目标提供了有力的理论依据。

本节将信号稀疏分解应用于参数估计领域<sup>[27, 28]</sup>，分析了可以在低信噪比环境下同时精确估计多个正弦信号频率的算法。这种算法较基于正交分解的算法有更高的频率分辨率，可以在更低的信噪比环境下估计信号频率，并且可以直接应用于欠采样的情况。

#### (1) 算法基本思想

假设接收数据中包含  $K$  个淹没在加性噪声中的正弦信号， $N$  次快拍后得到  $N$  个采样数据， $x(0), x(1), \dots, x(N-1)$ ，则信号模型可表示为：

$$x(n) = \sum_{k=1}^K \alpha_k e^{j(2\pi f_k n + \phi_k)} + w(n) \quad (5.15)$$

其中,  $\alpha_k$  和  $f_k$  ( $k=1,2,\dots,K$ ) 分别为第  $k$  个正弦信号的幅度和频率, 这两个参数都是未知常量。  $\phi_k$  为初相位, 假设初相位在  $[0,2\pi)$  内均匀分布。信号个数  $K$  已知。  $w(n)$  为零均值方差为  $\sigma^2$  的加性高斯白噪声。

我们的目的是从接收信号的  $N$  个采样数据  $x(n)$  中, 估计出  $K$  个正弦信号的频率。下面将分析如何利用信号的 MP 分解理论估计信号频率。

根据待估频率的信号特征, 设计如下的过完备原子库:

$$g_\gamma(n) = \cos(2\pi f_m n) \quad (5.16)$$

其中,  $n=0,1,\dots,N-1$ ;  $m=0,1,\dots,M-1$ ;  $f_m$  为频率参量, 按照需要的搜索精度均匀取值;  $M$  为频率搜索个数, 原子参数  $\gamma$  由  $m$  单独决定。整个原子库的原子数为  $M$  个。基于上面构造的原子库做 MP 分解可以估计出所有  $K$  个正弦信号的频率  $f_k$  ( $k=1,2,\dots,K$ )。

由于信号稀疏分解的特性, 正弦信号在进行 MP 分解时, 将在某一分量上达到最大匹配。而噪声不具有正弦信号的特征, 因此在这一分量上投影接近零, 从而达到滤除噪声的目的。参照该原子的参数, 可以估计得到待检测正弦信号的参数。基于这种思想, 将淹没在噪声中的多个正弦信号  $x(t)$  进行稀疏分解, 将会在不同分量上得到最大匹配, 根据这些与信号最大匹配的原子参数, 可估计出每个正弦信号的频率。

设经过 MP 分解, 从过完备库中选出与待分解信号  $x$  最为匹配的原子  $g_{\gamma_k}$ , 即:

$$\langle x, g_{\gamma_k} \rangle = \sup_{\gamma \in \Gamma} |\langle x, g_\gamma \rangle| \quad (5.17)$$

其中,  $g_{\gamma_k} = \cos(2\pi f_k n)$ , 则这一接收信号中包含频率为  $f_k$  的正弦信号。

如果接收信号中正弦信号的个数  $K$  已知, 则找出前  $K$  个最匹配的原子  $g_{\gamma_1}, \dots, g_{\gamma_K}$ , 即可估计出  $K$  个正弦信号的频率。

从以上的算法分析可以看出, MP 分解和 DFT 均可以估计出信号频率, 但由于 MP 分解是基于原子库的非正交性, 文中所提基于 MP 分解的算法和基于 DFT 的算法有本质的不同, 在频率估计结果中, 基于 MP 的算法明显优于基于 DFT 的算法。下面分析在频率分辨率方面和在欠采样情况下两种算法的比较。

频率分辨率: 通过信号的 DFT 分解求得信号的频率时, 固有的最小频率分辨率  $\Delta f$  为:

$$\Delta f = \frac{f_s}{N} \quad (5.18)$$

其中,  $f_s$  为采样频率,  $N$  为采样数据长度。从上式可以看出, 在采样率较低和数据长度较少的情况下, DFT 的频率分辨率较低。而基于 MP 分解的方法估计

频率时，如果不计噪声影响，频率分辨率可以无限高，因为其频率分辨率仅取决于原子库构造的冗余度。

欠采样情况：在欠采样情况下，基于 DFT 理论求得的频谱会发生混叠而无法求得正确的信号频率估计值，而基于 MP 分解的方法由于是寻找最大匹配，因此，求得频率正确估计值的概率有大幅度提高。

### (2) 计算机仿真

为了验证文中所提算法，进行了计算机模拟。在模拟中为了更好地表述算法估计性能，我们定义标准差为：

$$std = \left( \frac{1}{n} \sum_n (x_{g_n} - x_z)^2 \right)^{1/2}$$

其中， $x_{g_n}$  为第  $n$  次的频率估计值， $x_z$  为频率真值。

原子库构造过完备性直接影响方法的估计精度和估计效率。过完备性程度越高，则估计精度越好，但估计效率越差。而库的过完备性由参数  $f_m$  稠密过采样情况决定。为了既保证估计精度，又保证估计效率，在仿真中设计了一种快速的算法。其基本思想是由粗到细的快速搜索算法。

#### ① 频率估计结果和信噪比的关系。

实验中采用单个正弦信号，其频率为  $f = 9.1410 \text{ Hz}$ 。噪声为零均值高斯白噪声。信噪比  $SNR$  从  $-20 \text{ dB}$  开始每增加  $2 \text{ dB}$  作 200 次 Monte Carlo 实验。图 5.10 所示是快拍次数  $N=128$ 、 $M=10\,000$ 、频率搜索间隔取  $0.001 \text{ Hz}$  时频率估计值的标准差，及两种算法频率估计值与真值的偏差随信噪比  $SNR$  的变化曲线。

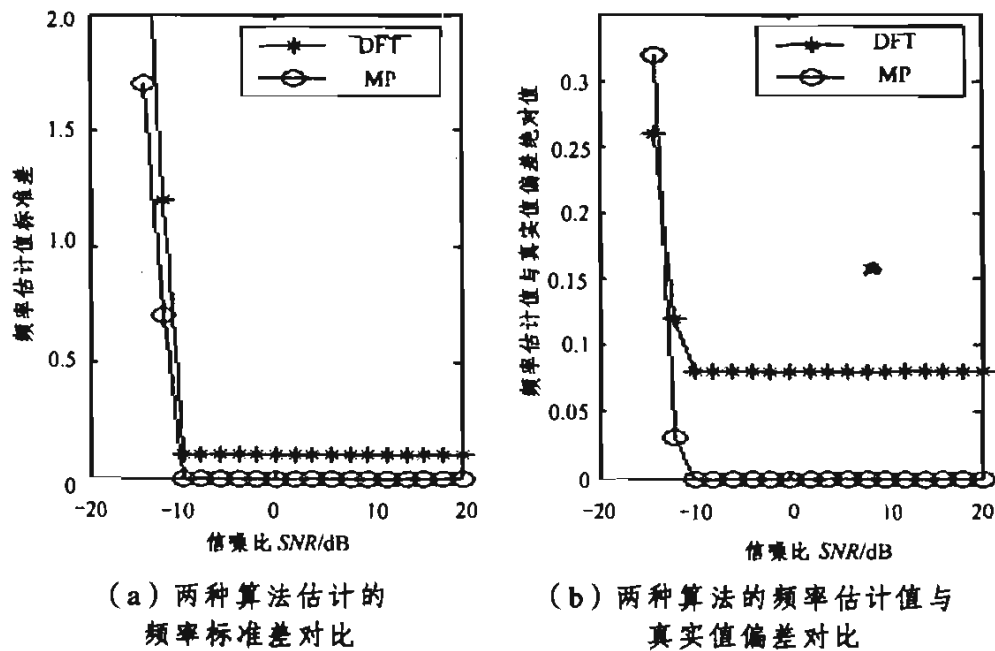


图 5.10 单个正弦信号估计结果



从图 5.10 看出, 基于 MP 分解的算法不但比基于 DFT 的算法具有更高的频率分辨率, 而且可以在更低的信噪比环境下获得较好的频率估计结果。

## ② 欠采样情况。

我们在欠采样情况下对两种算法进行仿真, 我们选取单个正弦信号, 其频率为  $f = 9.1410 \text{ Hz}$ 。噪声为零均值高斯白噪声。我们取采样点数  $N=256$ ,  $M=10\,000$ , 频率搜索间隔取  $0.001 \text{ Hz}$ , 采样频率为  $f_s = 15 \text{ Hz}$ 。信噪比  $SNR$  从  $-30 \text{ dB}$  开始每增加  $2 \text{ dB}$  做 200 次 Monte Carlo 实验。图 5.11 所示是频率估计值的标准差及两种算法频率估计值与真值的偏差随信噪比  $SNR$  的变化曲线。

从图 5.11 可以看出, 在欠采样情况下, DFT 算法基本失效, 而基于 MP 的算法仍然可以得到正确的频率估计, 尤其在信噪比较低情况下, 仍然可以得到很好的频率估计结果。

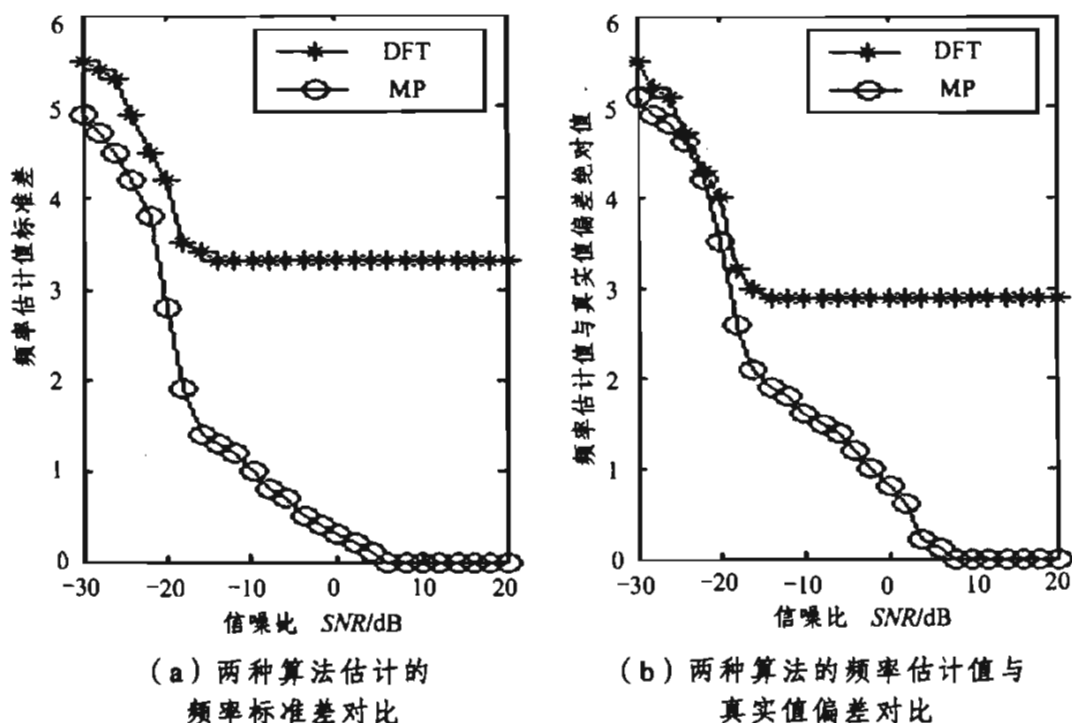
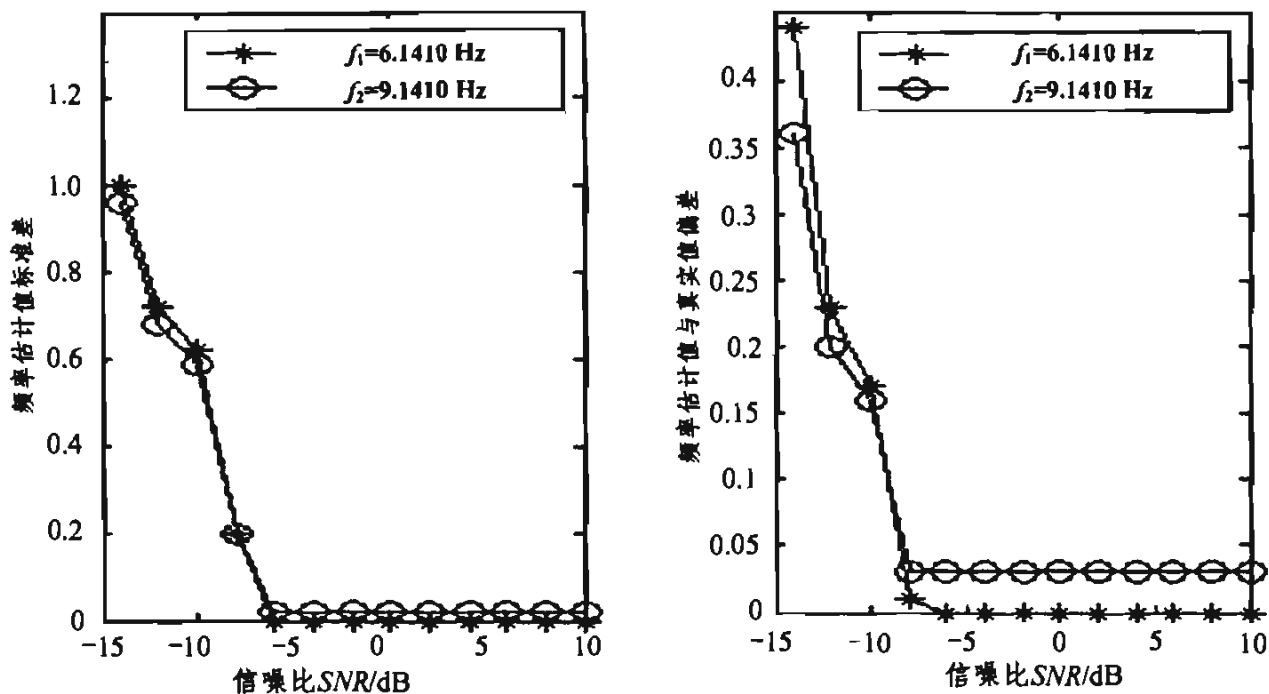


图 5.11 欠采样情况下单个正弦信号估计结果

## ③ 多信号情况。

取两个正弦信号频率参数分别为  $f_1 = 6.1410 \text{ Hz}$  和  $f_2 = 9.1410 \text{ Hz}$ 。噪声为零均值高斯白噪声。信噪比  $SNR$  从  $-14 \text{ dB}$  开始每增加  $2 \text{ dB}$  做 200 次 Monte Carlo 实验。图 5.12 所示是快拍次数  $N=256$ 、 $M=10\,000$ 、频率搜索间隔取  $0.001 \text{ Hz}$ , 采样频率为  $20 \text{ Hz}$  时, 同时对两个正弦频率进行估计的标准差及两种算法频率估计值与真实值的偏差随  $SNR$  的变化曲线。

从仿真结果可以看出, 基于 MP 分解的频率估计算法可以同时获得多信号高分辨频率估计。



(a) 两种算法的频率估计值标准差对比 (b) 两种算法的频率估计值与真实值偏差对比

图 5.12 多个信号源情况下的频率估计结果

### 5.3.2 DOA 估计算法

阵列信号处理是近 30 年来发展很快的一个技术领域，在电子战系统和广域无线网络系统中的应用十分广泛。利用阵列信号进行雷达辐射源信号参数估计是电子侦察系统的重要内容，是战场信息获取的主力军，是决定电子侦察系统性能、影响电子战成败的关键环节。因此，从 20 世纪 60 年代以来，阵列信号参数估计问题就引起了人们极大的研究兴趣。Capon 的最大似然 (MVM) 法<sup>[29]</sup>和 Burg 的最大熵 (MEM) 法<sup>[30]</sup>开创了高分辨 DOA 估计的新纪元，一度得到了广泛研究，而这些方法由于计算量等方面的缺陷，限制了其发展应用。现在应用最广泛的是子空间分析法，包括 Schmidt 提出的 MUSIC 法<sup>[15]</sup>和 Roy 提出的 ESPRIT 法<sup>[16]</sup>。

目前，应用最广泛的子空间分析方法及小波分析方法等大多是基于信号的正交分解，将阵列信号正交投影到完备的子空间上。由于信号的正交分解将信号分解为在各个正交基函数上投影分量的加权和，故存在如基函数系的正交性、完备性保证及信号的固有最小分辨率等诸多的限制。本节将信号非正交分解引入到阵列信号处理领域，通过阵列信号的 MP 分解实现空间谱的超分辨估计，分析了波达方向 (DOA) 估计算法。在一定的过完备库原子密度的条件下，本节所述算法波达方向估计的分辨率高于传统的波达方向估计算法，尤其在低信噪比情况下，算法仍然有良好的性能。

### (1) 算法基本思想

考查在空间分布的  $D$  个远场窄带信号源。并利用一个  $M$  阵元的均匀线性阵列天线（见图 5.13）对这些信号进行接收。窄带信号源可以用如下的复包络形式来表示：

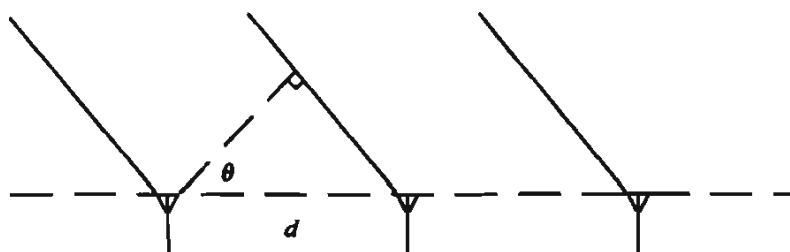


图 5.13 阵列结构

$$s(t) = u(t)e^{j(\omega_0 t + \varphi(t))} \quad (5.19)$$

式中， $u(t)$  是接收信号的幅度， $\varphi(t)$  是接收信号的相位， $\omega_0$  是接收信号的中心频率。

用上述均匀线性阵列进行接收，两个相邻阵元子间隔为  $d$ ，因电波传播延迟  $\tau$ ，两个相邻阵元间接收的第  $i$  个信号可以分别表示为：

$$s_i(t) = u_i(t)e^{j(\omega_0 t + \varphi(t))} \quad (5.20)$$

$$s_i(t - \tau) = u_i(t)e^{j(\omega_0(t - \tau) + \varphi(t))} \approx s_i(t)e^{-j\omega_0 \tau} \quad (5.21)$$

由图 5.13 所示阵列天线结构易知，信号波达方向  $\theta_i$  与传播时延  $\tau$  间存在如下关系：

$$\tau = \frac{d \sin \theta_i}{c} \quad (5.22)$$

以第一个阵元为参考阵元，可以得到第  $l$  个阵元接收信号为：

$$x_l(t) = \sum_{i=1}^D a_{li} s_i(t - \tau_{li}) + n_l(t) \quad (5.23)$$

式中， $a_{li}$  为第  $l$  个阵元对第  $i$  个信号的接收增益，对于各向同性的理想阵元子， $a_{li}$  可归一化为 1。 $n_l(t)$  表示第  $l$  个阵元在  $t$  时刻的噪声， $\tau_{li}$  表示第  $i$  个信号到达第  $l$  个阵元时相对于参考阵元的时延。

整个阵列的输出写成矢量形式如下：

$$\mathbf{X}(t) = \mathbf{A}\mathbf{S}(t) + \mathbf{N}(t) \quad (5.24)$$

式中， $X(t)$  为阵列接收信号的  $M \times 1$  维数据快拍矢量， $S(t)$  为空间分布的  $D$  个窄带信号源的  $D \times 1$  维数据矩阵， $N(t)$  为阵列的  $M \times 1$  维噪声数据矢量。 $A$  为阵列信号的  $M \times D$  维导向矢量矩阵：

$$A = [a_1(\omega_0) \ a_2(\omega_0) \ \cdots \ a_D(\omega_0)] \quad (5.25)$$

对于第  $i$  个信号，我们可以得到其导向矢量为：

$$a_i(\omega_0) = [\exp(-j\omega_0\tau_{1i}), \exp(-j\omega_0\tau_{2i}), \cdots, \exp(-j\omega_0\tau_{Mi})]^T$$

$$i = 1, 2, 3, \cdots, D \quad (5.26)$$

式中， $\tau_{ki}$  为第  $i$  个信号到达第  $k$  个阵元时的传播延迟。由式 (5.22) 易知：

$$\tau_{ki} = \frac{(k-1)d \sin \theta_i}{c} \quad (5.27)$$

由上面的阵列信号模型可知，只要得到传播时延  $\tau_{ki}$ ，然后根据式 (5.27)，就可以得到对信号的波达方向角度的估计。

传统的阵列信号处理方法往往是将信号分解在正交完备的子空间上，存在诸多限制。本节突破信号在正交基函数上投影的思想，将信号 MP 分解的方法引入到上节所述的阵列信号模型中，对阵列信号进行非正交分解。由于信号稀疏分解投影的基函数可以根据信号本身的特性灵活地选取，因此，通过过完备向量原子库的建立，可以将阵列信号投影到任意逼近期望信号的基函数上，并利用分解系数的特性得到我们感兴趣的信息。

考查阵列信号的波达方向估计，由阵列信号模型可知，只要得知阵元间电波传播时延  $\tau_{ki}$ ，根据式 (5.27) 即可得到波达方向角的估计。传播时延  $\tau_{ki}$  包含在式 (5.26) 所示的阵列导向矢量中，因此，为了得到我们感兴趣的信号波达方向信息，首先建立如式 (5.26) 所示形式的过完备向量原子库：

$$G_r \{a_i(\omega_0)\} = \begin{bmatrix} 1 \\ \exp(-j\omega_0 d \sin \theta_m / c) \\ \vdots \\ \exp(-j\omega_0 (M-1) d \sin \theta_m / c) \end{bmatrix}, \quad m = 1, 2, \cdots, M \quad (5.28)$$

由式 (5.28) 可知，通过给定原子向量的参量  $m$  的步长即可设置原子向量库的密度，形成一个过完备的向量投影空间。理论上可以通过简单的向量投影计算将阵列信号投影到每个原子向量上。

假定已知空间分布信号的信息，由式 (5.24) 和式 (5.28)，可以得到过完备原子向量库：

$$G_r\{a_i(\omega_0)\} = S(t) \begin{bmatrix} 1 \\ \exp(-j\omega_0 d \sin \theta_m / c) \\ \vdots \\ \exp(-j\omega_0 (M-1)d \sin \theta_m / c) \end{bmatrix}, \quad m=1,2,\dots,M \quad (5.29)$$

中,  $\theta_m$  为要估计的波达方向参量, 按照需要的搜索精度均匀取值,  $M$  为角度搜索个数, 原子参数  $\gamma$  由  $m$  单独决定。整个原子库的原子数为  $M$  个。基于面构造的原子库对阵列信号做上节所示 MP 分解。

在正交分解中, 每个向量都可以唯一的表示成子空间  $M$  与  $M^\perp$  的两个向量之和, 即:

$$x = P_M x + (I - P_M)x \quad (5.30)$$

中,  $P_M$  为投影映射算子<sup>[31]</sup>。

根据稀疏分解理论可知, 可以将式 (5.30) 所示的向量投影推广到过完备中, 从而将阵列信号  $x$  投影到过完备向量原子库的各原子向量上, 达到阵列信号的稀疏表达。首先定义  $P_H$  为过完备向量空间上的投影映射算子, 则有:

$$P_H = \frac{\langle y, x \rangle}{\|y\|^2} \quad (5.31)$$

式 (5.31) 可以得到向量  $x$  到向量  $y$  上的投影, 它的长度等于  $x$  在  $y$  方向上长度:

$$P_H x = \frac{\langle y, x \rangle}{\|y\|^2} y \quad (5.32)$$

由本章参考资料[31]知, 对于矩阵投影算子, 有:

$$P_H = H(H^T H)^{-1} H^T \quad (5.33)$$

式中,  $H = \text{span}\{G_r\{a_i(\omega_0)\}\}$ 。

选取如式 (5.29) 所示的过完备向量原子库, 则可以将向量分解为:

$$x = P_H x + Rx \quad (5.34)$$

式中,  $x$  为待分解的阵列接收信号,  $P_H x$  为  $x$  在最佳原子向量上的投影,  $Rx$  为用最佳原子向量对  $x$  进行最佳匹配后的残余。

如前所述, 为得到阵列信号的波达方向信息, 需要建立如式 (5.29) 所示的原子向量库,  $\theta_m$  为要估计的波达方向参量, 按照需要的搜索精度均匀取值,

$M$  为角度参量搜索个数, 原子参数  $\gamma$  由  $m$  单独决定。在进行 MP 分解的过程中, 我们注意到式 (5.34) 给出了最佳原子向量的选取约束, 在此约束条件下, 阵列接收信号在进行 MP 分解时, 将在某一原子向量上的投影分量上达到最大匹配。而噪声不具有阵列信号的特征, 因此在这一分量上投影接近零, 从而达到滤除噪声的目的。显然, 通过对波达方向参量  $\theta_m$  的搜索, 可以得到一个最逼近原始阵列信号导向矢量的原子向量, 此向量与原始阵列信号方向最为一致, 投影取得最大值。由此获得了原子参量  $m$ , 由  $m$  即可获得阵列信号的波达方向估计。

由上述分析知, 为得到阵列信号的波达方向信息, 要考虑式 (5.34) 的约束条件, 由向量投影定理知, 在 Hilbert 空间  $H$  的闭子空间  $M$  中, 对所有  $x \in M$ , 存在一个唯一的元素  $\bar{x} \in M$  使得:

$$\|x - \bar{x}\| = \inf_{y \in M} \|x - y\|; \quad (5.35)$$

并且,  $\bar{x}$  满足式 (5.35), 当且仅当  $\bar{x} \in M$  和  $(x - \bar{x}) \perp M$ 。

式 (5.35) 给出了式 (5.33) 的约束条件, 即:

$$\|x - G_m\| = \inf_{y \in M} \|x - G_y\| \quad (5.36)$$

## (2) 算法总结

综合上面的分析, 可以得到阵列信号 DOA 估计的 MP 分解算法, 该算法的基本流程如下:

① 建立如式 (5.29) 所示的过完备向量原子库, 并按照需要的搜索精度对  $\theta_m$  均匀取值进行设置,  $m$  为角度参量搜索个数。

② 按式 (5.36) 给出的约束, 计算阵列信号  $X(t)$  与原子向量的距离  $l$ :

$$l = \inf_{y \in M} \|x - G_y\| \quad (5.37)$$

从而得到最佳原子参量  $m$ 。

③ 由原子参量  $m$  得出阵列信号的波达方向估计。

## (3) 计算机仿真

为了验证上述算法的有效性并和 ESPRIT、MUSIC 算法比较, 进行了计算机模拟。模拟采用均匀线性天线、阵元间隔为  $1/2$  载波波长即  $d = \lambda/2$ 、噪声为加性高斯白噪声。取信号为正弦信号, 来波的真实方向为  $60^\circ$ , 采样长度为 128。信噪比  $SNR$  从  $-10$  dB 到  $30$  dB 每增加  $2$  dB 做 200 次 Monte Carlo 实验, 取 200 次估计值的平均值得到最终的估计。

15 ① 3 阵元情况下的结果。

图 5.14 所示是 3 阵元情况下 DOA 估计值的标准差及随信噪比  $SNR$  的变化曲线。结果表明，基于 MP 的算法在少阵元情况下的估计结果优于 ESPRIT 算法和 MUSIC 算法。

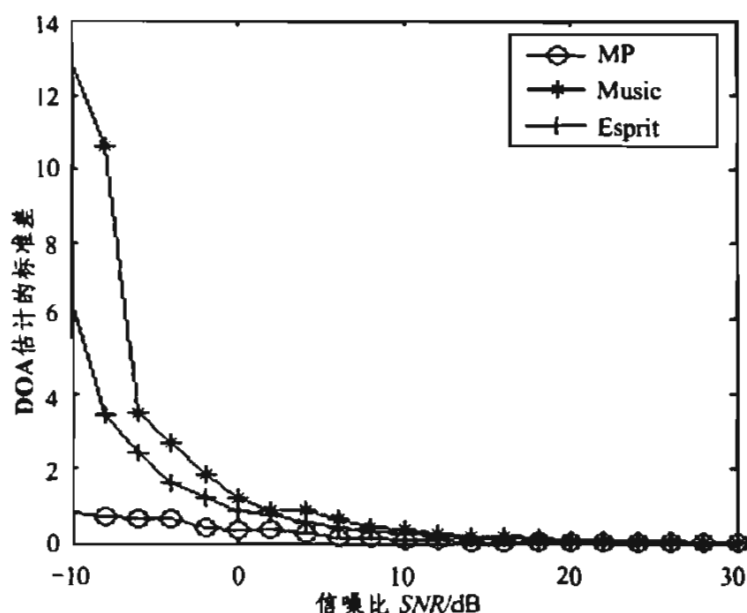


图 5.14 DOA 估计值的标准差及随信噪比  $SNR$  的变化

② 低信噪比情况下的结果。

为了验证算法在低信噪比下的 DOA 估计效果，采用一个 10 阵元天线进行仿真，仿真结果如图 5.15 所示。

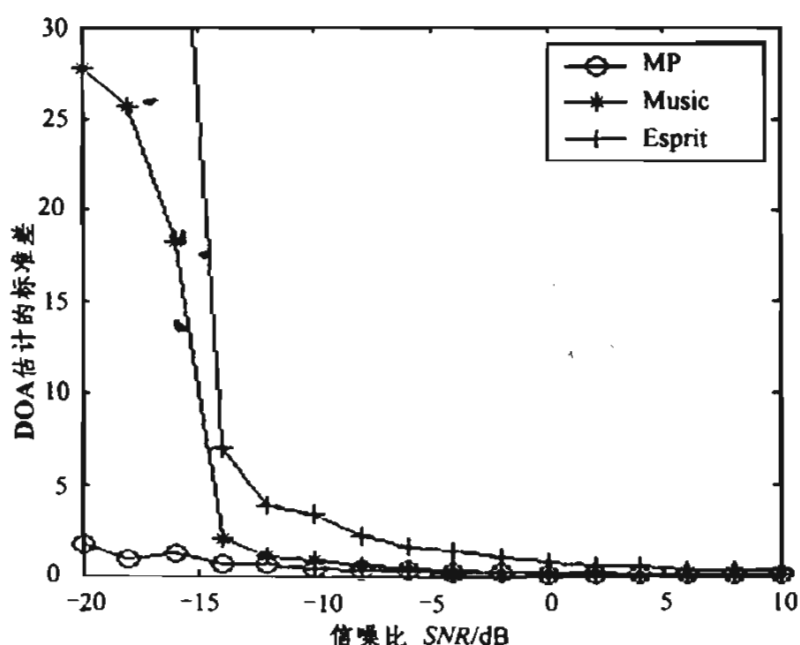


图 5.15 DOA 估计值的标准差及随信噪比  $SNR$  的变化

仿真结果表明,基于 MP 的算法在低信噪比情况下的估计结果优于 ESPRIT 算法和 MUSIC 算法,尤其在低于  $-15\text{ dB}$  以下时,ESPRIT 算法和 MUSIC 算法基本失效,而基于 MP 的算法仍然有很好的结果。

值得提出的是,由于基于 MP 的算法是采用阵列信号的功率,而不是信号子空间,因而这类算法可以直接用来处理不同到达方向的相关信号。

## 5.4 结 论

本章推导了信号稀疏分解在信号去噪、微弱信号检测以及阵列信号参数估计中的应用方法。从本章的分析和仿真结果可以看出,信号稀疏分解理论在信号去噪以及在低信噪比环境下对信号的检测、识别处理等方面表现出了优良的特性,显示了稀疏分解理论在信号处理应用领域的发展潜力。并得出了以下结论:

① 在去噪方面,应用稀疏分解理论的相干比阈值去噪的优点如下:随迭代次数的增加,该相干比值在分解前部分呈下降趋势,有最小值。不受其他因素影响,对于相同的原子库和不同的类型信号,相干比阈值一致,所以这种去噪方法具有自适应的特性。作为一个好的迭代停止原则,还能减少迭代的次数达到降低计算量的目的,而迭代停止时,其去噪效果达到最佳。

② 在微弱检测方面,应用稀疏分解的方法可以在  $-40\text{ dB}$  以下的恶劣信号环境中检测出多个正弦信号。

③ 在频率估计方面,可以取得比 DFT 更高的频率分辨率,而且可以在更低的信噪比环境下获得较好的频率估计结果。

④ 在阵列信号 DOA 估计方面,基于阵列信号稀疏分解的算法可以在少阵元情况下获得良好的参数估计性能,从而减少硬件成本和阵列误差对算法的影响;而且,基于阵列信号稀疏分解的算法可以直接适用于相干信号处理。此方法可以适用于  $-20\text{ dB}$  以下的低信噪比环境。

当然,稀疏分解理论及其在信号处理中的应用仍然属于发展的起始阶段,我们做的也只是很初步的工作,但已表现出了许多优良特性。有关信号稀疏分解在信号处理领域的进一步研究仍然有大量的工作需要完成。

## 参 考 资 料

- [1] 周崇经. 滤波理论与波形编码[M]. 辽宁科学技术出版社, 1986
- [2] D.L.Donoho, I.Johnstone. Ideal spatial adaptation via wavelet shrinkage [J]. Biometrika, 1994, 81 (11): 425~455



- [3] RR Coifamn and D.L.Donoho, Transaction-Invariant De-Noising. Technical Report, Yale University and Stanford University
- [4] 刘利雄, 贾云得, 廖斌, 张敏. 一种改进的最佳时频原子搜索策略[J]. 中国图像图形学报, 2004, 9 (7): 873~877
- [5] 张文耀. 基于匹配跟踪的低位率语音编码研究[D], 博士学位论文, 北京: 中国科学院研究生院 (软件研究所), 2002
- [6] Mallat S, Zhang Z. Matching Pursuits with Time-frequency Dictionaries [J], IEEE Trans. Signal Processing, 1993, 41: 3397~3415
- [7] Arthur P L, Philipos C L. Voiced/unvoiced speech discrimination in noise using gabor atomic decomposition [A], Proc. Of IEEE ICASSP[C], Hong Kong, April, 2003, VoL I: 820~828
- [8] 姜玉亭. 基于稀疏分解的图像去噪. 硕士学位论文, 成都: 西南交通大学, 2005
- [9] Tuteur F B. Wavelet Transforms in Signal Detection [A], Proc. of IEEE ICASSP[C], New York, USA, 1988: 1435~1438
- [10] Wang G, Chen D, Lin J, et al. The Application of Chaotic Oscillators to Weak Signal Detection [J], IEEE Trans. On Industrial Electronics, 1999, 45 (2): 440~444
- [11] Bjorn Volcker, Peter Handel. Frequency Estimation From proper Sets of Correlations [J]. IEEE Trans. on signal processing 2002, 50 (4): 791~802
- [12] Steven Kay and Supratim Sala Mean Likelihood Frequency Estimation [J]. IEEE Trans. on signal processing, 2000, 48 (7): 1937~1946
- [13] D.C.Rife, R.R.Boorslyn. Single tone parameter estimation from discrete-time observation [J]. IEEE Trans. Info.Theory, 1974, IT-20 (5): 591~598
- [14] R.D.Degroat. Nofiterative subspaces tracking [J]. IEEE Trans. Signal processing, 1992, 40 (3): 571~577
- [15] Schmidt R O. Multiple emitter location and signal parameter estimation [J]. IEEE Trans. Antennas and propagation, 1986, 34, (3): 276~280
- [16] Roy R. and Kailath T. ESPRIT-Estimation of Signal Parameters via Rotational Invariance Techniques [J]. IEEE Trans. Acoustics, Speech, and Signal processing, 1989, 37 (7): 984~995
- [17] Wang J.Y., Chen T.Q. Joint frequency, 2D AOA and polarization estimation using fourth-order cumulants[J], SCIENCE IN CHINA (E), 2000, 43 (3): 297~303
- [18] Wang Jianying, Chen Tianqi, Joint frequency. 2-D AOA and polarization estimation in broad-band [J]. Science In China (series F), 2001, 44 (03): 161~167
- [19] Wang Jianying, Chen Tianqi, Joint frequency. 2-D AOA and polarization estimation

- using forth-order cumulants [J]. Science In China (series E), 2000, 43 (3): 297~303
- [20] Wang Jianying, Chen Tianqi. Performance analysis of joint estimation of frequency, 2-D AOA and polarization under array error[C]. proceedings of ICCAS 2002
- [21] 王建英, 曾庆华, 陈天麒. 用任意阵实现二维到达角和极化的联合估计[J]. 电波科学学报, 1999, 14 (4): 410~415
- [22] 王建英, 陈天麒. 频率、二维到达角和极化联合估计[J]. 电子学报, 1999, 27 (11): 74~76
- [23] 王建英, 陈天麒. 用四阶累积量实现频率、二维到达角和极化的联合估计[J]. 中国科学 (E 辑), 2000, 30 (5): 424~429
- [24] 王建英, 王激扬, 陈天麒. 宽频段频率、二维到达角和极化联合估计[J]. 中国科学 (E 辑), 2001, 31 (12): 526~532
- [25] 王建英, 陈天麒. 用 L 阵实现频率、二维到达角和极化的联合估计[J]. 电波科学学报, 2001, 16 (1): 30~33
- [26] 黄佑勇, 王激扬, 陈天麒. 基于欠采样的宽频段信号频率估计技术[J]. 电波科学学报, 2001, 16 (2): 275~279
- [27] 王建英, 尹忠科, 陈磊. 基于非正交分解的频率估计算法[J]. 电波科学学报, 已经正式录用, 将于 2006 年上半年发表
- [28] 王建英, 尹忠科. 基于非正交分解的频率估计算法[J]. 铁道学报, 2006, (4)
- [29] Capon J. High-resolution frequency-wavenumber spectrum analysis [J]. Proc. Of IEEE, 1969, 57 (8): 1408~1418
- [30] Burg J P. Maximum entropy spectral analysis [C]. PhD Thesis, Stanford University, Stanford, USA, 1975
- [31] 张贤达. 信号处理中的线性代数[M]. 北京: 科学出版社, 1997

## 第6章 基于MP的图像稀疏 分解快速算法

在图像和视频处理中,常用的信号分解方式是非冗余的正交变换。例如,现行的 JPEG 国际静止图像压缩标准采用的变换为离散余弦变换;国际视频压缩标准(如 MPEG 系列)采用的变换往往也是离散余弦变换。新近推出的 JPEG2000 国际静止图像压缩标准采用的变换为小波变换。离散余弦变换由于其基函数缺乏时间/空间分辨率,故而不能有效地提取具有时频局部化特性的信号特征。小波分析在处理一维和二维的具有点状奇异性的对象时,表现出良好的性能,但图像边缘的不连续性是按空间分布的,小波分析在处理这种线状奇异性时效果并不是很好。因而,小波分析对于多维信号处理来说并不是最优的,它们不能稀疏地捕捉到图像结构的轮廓特征,因此,在图像和多维编码方面的新突破,必定取决于信号表示的深刻变革。所以,与一维信号稀疏分解、稀疏表示相比,二维图像的稀疏分解和稀疏表示更加重要,意义更加突出。

从第6章开始,讨论图像的稀疏分解、稀疏表示和在图像处理中的初步应用。根据信号空间理论,图像与一维信号在信号空间中并没有本质的区别,图像的稀疏分解理论与信号的稀疏分解理论是一致的,所以这里不再详细地给出图像稀疏分解的理论,以免重复。本章结合图像的特点,介绍了作者在图像稀疏分解方面的工作。后面两章中分别介绍了稀疏分解在图像去噪和图像低比特率压缩中的应用。

### 6.1 图像稀疏分解的原子库

在信号与图像稀疏分解和稀疏表示中,原子库具有决定性的作用。原子库的完备性决定了稀疏分解过程的计算复杂度,同时也决定了稀疏分解结果稀

疏表示的简洁性或稀疏性。原子库中原子的选择也对图像表示的稀疏性有着重要的影响。

针对过完备原子库，为了更好地表示图像内容，已经提出了非对称原子库<sup>[1]</sup>等，而新发展的 Ridgelet、Curvelet、Bandelet 和 Contourlet<sup>[2~5]</sup>等也都可以作为原子模型而形成原子库。多个原子库组合应用是一个有前景的研究方向，库的结构特性也正在引起人们越来越多的关注<sup>[6]</sup>。

## 6.2 基于原子能量特性的图像稀疏分解算法

### 6.2.1 原子能量分布

设研究的图像为  $f$ ，图像大小为  $M_1 \times M_2$ ， $M_1$  和  $M_2$  分别为图像的长和宽。若将图像分解在一组完备正交的基上，则这组基的数目应为  $M_1 \times M_2$ 。由于基的正交性，因而基在由图像所组成的空间中的分布是稀疏的，从而，图像的能量在分解以后将分散分布在不同的基上。这种能量分布的分散最后将导致用基的组合表示图像时表达的不简洁性，即图像表示不是稀疏的。非稀疏的表示，不利于图像的处理，如识别和压缩等。为了得到图像的稀疏表示，基的构造必须使得基在图像组成的空间中足够的密。由此，基的正交性将不再被保证，所以此时的基也不再是真正意义上的基了，而改称为原子。由这些原子组成的集合，是过完备的，被称为过完备库。图像在过完备库上的分解结果一定是稀疏的<sup>[7, 8]</sup>。

设  $\mathcal{D} = \{g_\gamma\}_{\gamma \in \Gamma}$  为用于进行图像稀疏分解的过完备库， $g_\gamma$  为由参数组  $\gamma$  定义的原子。用不同的方法构造原子，参数组  $\gamma$  所含有的参数及参数个数也不一样。原子  $g_\gamma$  的大小与图像本身大小相同，但原子应作归一化处理，即  $\|g_\gamma\| = 1$ 。 $\Gamma$  为参数组  $\gamma$  的集合。由库的过完备性可知，参数组  $\gamma$  的个数应远远大于图像的大小，即若用  $P$  表示过完备库  $\mathcal{D} = \{g_\gamma\}_{\gamma \in \Gamma}$  中原子的个数，则  $P$  应远远大于图像大小  $M_1 \times M_2$ 。

从信号空间理论出发，信号稀疏分解与图像稀疏分解的方法是一样的，不同的是用以进行分解的原子不同。所以这里不再讨论分解的具体过程。基于 MP 的图像稀疏分解中每一步中的核心问题仍是求解式 (6.1) 所示的最优化问题。

$$|\langle R^k f, g_{\gamma_k} \rangle| = \sup_{\gamma \in \Gamma} |\langle R^k f, g_\gamma \rangle| \quad (6.1)$$

中,  $R^k f$  为图像基于 MP 的稀疏分解过程中第  $k$  步分解的残差, 并且  $R^k f = f$ 。  $g_n$  为第  $k$  步分解中选取的最优或最佳原子,  $\langle \cdot \rangle$  表示  $M_1 \times M_2$  维空间的内积。

由于原子库的过完备性, 原子库中原子的个数非常巨大, 使得解式 (6.1) 表的最优化问题非常困难, 几乎是不可能完成的任务。具体来说, 其困难主要来源于两个方面: ① 由于原子库中原子的个数非常多, 所以要进行非常多的内积计算。② 由于图像的数据一般远远大于一维信号的数据,  $M_1 \times M_2$  非常大, 使得计算一次内积的计算量也相当大。针对第一个问题, 在本章的后面给出一些切实可行的快速计算方法。针对第二个问题, 下面分析原子的能量特性, 从而使图像或图像残差与原子的内积计算大大简化。

在物理世界中, 原子是组成物质的基本单位之一。各种不同的物质由原子根据不同的组合方式而形成。与此很类似, 在图像稀疏分解中, 各种不同内容的图像可由原子根据不同的方式组合来合成。物理世界的一个原子, 原子中的大部分空间是空的, 即原子的质量和能量绝大部分集中在原子核处。与此相似, 用以进行图像稀疏分解的原子, 其能量也绝大部分集中在原子的中心处。根据不同的参考文献给出的原子, 均可发现此特性。图像稀疏分解在部分参考文献中也称为原子分解, 可能跟图像稀疏分解中使用的原子与物理世界的原子相似这一点有关。

不失一般性, 取本章参考资料[1]中的非对称原子为例来说明。非对称原子的数学模型如下:

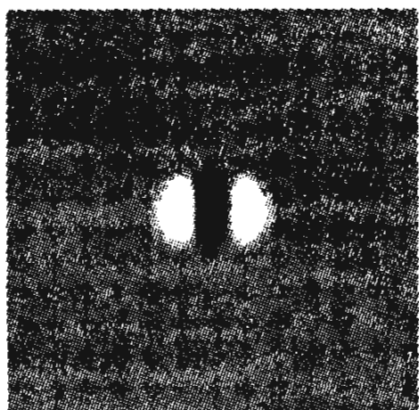
$$g(x, y) = (4x^2 - 2)e^{-(x^2 + y^2)} \quad (6.2)$$

根据式 (6.2) 所示的原子模型, 对其进行平移、旋转和伸缩, 就可以得到一系列不同的原子, 从而形成过完备原子库。

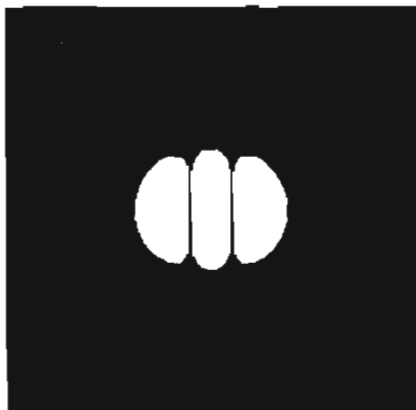
图 6.1 (a) 中表示的是一个一般性的非对称原子, 原子是经过归一化处理, 其幅度最大值为 1。图 6.1 (b) 中表示的是该原子中能量特性。其中白色部分表示原子中幅度值大于 0.01 的部分, 黑色部分表示原子中幅度值小于 0.01 的部分。如果用幅度的平方表示能量的话, 则白色部分集中了原子中的绝大部分的能量。

原子能量的这种分布特性, 主要是因为形成原子的数学模型中, 往往含有一个指数函数, 如式 (6.2) 所示, 使原子的能量随着离原子中心的距离而呈指数衰减。原子模型中含有指数函数, 是利用原子有效表示图像的要求, 即这种原子的能量分布特性对于利用原子来表示图像是有利的, 因而是必须的。原子能量的衰减特性, 使原子能够表示图像的细节。虽然需要利用部分大原子来

表示图像的背景，但这样的大原子的需要量是很小的。为了更加精准地表示图像，稀疏分解中使用的原子大部分应是很小的原子，也是能量非常集中的原子。



(a) 一个原子示意图



(b) 原子能量分布示意图

图 6.1 原子能量的分布特性

### 6.2.2 基于原子能量特性设计图像稀疏分解快速算法

原子能量分布特性给快速解决式 (6.1) 代表的最优化问题提供了可能。式 (6.1) 中的内积计算，由于是在高维 ( $M_1 \times M_2$  维) 空间的内积计算，所以计算十分耗时，利用原子能量的集中特性，可以把高维空间的内积计算，转化为低维空间的内积计算，从而实现高效的计算。

根据以上思路，提出了两种快速解决式 (6.1) 代表的最优化问题的方法。分别如下：

#### 算法 1

由于原子的能量集中分布在原子的中心，在离中心位置较远的地方，原子的能量几乎为 0。所以，图像或图像残差与原子的内积计算（即计算图像或图像残差在原子上的投影），可以转化为原子中心区域与图像或图像残差上相对应的一部分的内积计算。

设原子的中心区域为  $C$ ，中心区域下对应的图像像素个数为  $N_c$ 。则一般而论有

$$N_c \ll M_1 \times M_2 \quad (6.3)$$

且：

$$\langle R^k f, g_r \rangle \approx \langle R^k f|_C, g_r|_C \rangle \quad (6.4)$$

式 (6.4) 中， $g_r|_C$  表示原子的中心区域， $R^k f|_C$  表示图像或图像残差上和原子中心区域相对应的区域。

根据式 (6.4)，可以把很高维空间 ( $M_1 \times M_2$  维空间) 的内积计算，转化为

相对低得多的低维空间 ( $N_c$  维空间) 的内积计算, 使基于 MP 的图像稀疏分解速度相应提高许多倍。

式 (6.4) 是一个近似转化公式, 肯定会有误差。如果单独从式 (6.4) 分析, 可能会认为这个误差是不可接受的。实际分析会发现, 这种近似转化, 在实现快速计算的同时, 最终稀疏分解结果没有任何改变。根据我们的分析, 按照原因的重要次序, 这主要是因为如下几个方面: ① MP 算法的前后步之间有误差相互抵消的作用。② 解式 (6.1) 的过程, 是原子的选择过程, 在完成选择后, 我们得到了原子的参数, 即最佳原子的参数。由最佳原子的参数形成最佳原子时, 可以不根据能量特性只计算原子中心部分, 而是计算整个原子。这样在获取图像分解残差时就不会有这种误差。而每一步分解才有一个最佳原子, 所以这种最佳原子的形成方法, 在计算复杂度方面不会有任何问题。③ 解式 (6.1) 的目的, 并不是为了求图像或图像残差与原子的内积的精确值, 而是为了求其相对比较值, 从而求出一个最大值。综上所述, 这种误差并不会影响图像稀疏分解的最终结果。我们的实验结果也验证了这样的近似, 在大大提高了计算速度的同时, 不会给最后的分解结果即稀疏表示带来任何影响。

最后一个没有讨论的问题是: 对于任一给定的原子, 如何快速确定中心区域  $C$ ? 为了实现上面的想法, 必须有确定一个给定原子的中心区域  $C$  的快速方法。

给定一个原子, 则原子的参数也就确定了, 其中, 包括原子的平移参数、伸缩参数和旋转参数。原子的平移参数、伸缩参数和旋转参数中的任一参数不同, 则原子的中心区域  $C$  也应是不同的。也就是说, 任意两个不同的原子, 其中心区域  $C$  应是不同的。这就要求对每一个原子必须分别计算求得其中心区域  $C$ 。这样做的话, 将是非常费时的。

通过研究我们发现, 中心区域  $C$  的形状及空间分布由伸缩参数决定, 中心区域  $C$  在图像平面上的位置由平移参数决定, 中心区域  $C$  的方向由旋转参数决定。由此, 可以构造出一种求得任一原子中心区域  $C$  的快速算法。

快速算法的思想是: 把所有原子按原子的伸缩参数分类, 具有相同伸缩参数的原子作为一类原子, 具有不同伸缩参数的原子作为不同类的原子。对于同一类的原子, 为了确定每一个原子中心区域  $C$ , 可以先确定一个代表性的原子的中心区域  $C$ , 然后通过对此中心区域  $C$  的旋转与平移, 得到该类原子中每一个原子的中心区域  $C$ 。

为了进一步提高求得每一个原子中心区域  $C$  的算法的速度, 可以在分解程序开始时, 根据不同的原子伸缩参数, 求得每种伸缩参数下的原子中心区域  $C$ , 然后把原子伸缩参数与中心区域  $C$  的关系形成一个表格。在分解的过程中, 求

某个原子的中心区域  $C$  时, 根据该原子的伸缩参数, 在这个表格中查找其中心区域  $C$ , 然后根据该原子的平移参数和旋转参数, 对查找到的中心区域  $C$  进行平移和旋转, 就可以很方便、快速地得该具体原子的中心区域  $C$ 。

### 算法 2

图像稀疏分解过程的每一步中, 所需要的用来最佳表示图像或图像残差的原子是不同的。在基于 MP 的图像稀疏分解的初始阶段, 表示图像或图像残差的最佳原子应是比较大的原子, 以捕捉图像或图像残差上分布比较广的图像背景信息。随着基于 MP 的图像稀疏分解过程的继续, 图像上的广泛分布的信息被基本提取完了, 稀疏分解过程开始提取局部分布的图像细节信息, 而用来表示细节信息最佳原子应是小原子, 即中心区域  $C$  很小的原子。总之, 在基于 MP 的图像稀疏分解过程中, 最佳原子的伸缩参数 (或尺度) 在分解过程的不同阶段有不同的特性。

根据以上原子的能量特性和图像基于 MP 的稀疏分解过程的特点, 提出以下的快速算法。把图像基于 MP 的稀疏分解过程划分成不同的阶段, 如 1~4 个阶段, 针对每一个阶段, 确定可能的最佳原子的伸缩参数的范围。据此, 把原子库划分成不同的子集合。在图像基于 MP 的稀疏分解过程的某一个阶段, 寻找最佳原子只在原子库中的某个小的子集中寻找, 这样就大大缩小了寻找最佳原子的范围, 从而大大降低求解式 (6.1) 代表的最优化问题的难度。

## 6.2.3 实验结果与分析

对以上的算法, 进行了计算机模拟。实验中采用的图像为 Lena 国际标准测试图像。基于 MP 的图像稀疏分解算法中每一步的最优化问题的求解采用遗传算法 (Genetic Algorithm, GA), 关于此算法后面还要具体介绍。

对于算法的性能的优劣, 我们从两个方面进行评价, 所以实验也从两个方面研究: 第一个方面是算法的速度; 第二个方面是算法的效果, 即图像稀疏分解的好坏。

对于算法的速度, 严格地说, 应分析标准的基于 MP 的图像稀疏分解算法中乘法运算次数是多少, 而本文提出的快速算法的乘法运算次数又是多少。但是, 这种分析是有很难度的, 也是不太现实的。这是因为对于稀疏分解一幅图像来说, 原子库的大小将是一个天文数字。所以对于图像稀疏分解的算法速度的讨论, 往往不是以此分析, 而是讨论不同算法之间的相对速度是多少。为了使对比分析具有可比性, 要在计算的硬件环境和软件环境一致的情况下测试不同的算法。把标准的基于 MP 的图像稀疏分解算法的速度设为 1, 把其他算法的速度设为它的速度的倍数, 作为分析算法速度的方法。



表 6.1 中给出了算法速度的实验结果。由于实验中的基于 MP 的稀疏分解算法中，总体上最优化问题的解决使用了遗传算法，因而算法速度带有一定的随机性，所以表中给出的实验结果是一个 5 次平均计算结果。从实验结果中可以看出，对于不同大小的 Lena 图像，算法的速度的改进是不同的。图像越大，本文算法的速度的提高越明显。

表 6.1 不同图像大小情况下新算法的速度

图像大小	128×128	256×256	512×512
速度提高（倍数）	12	20	31

上面从算法的速度方面讨论了本文提出的基于原子能量特性的图像稀疏分解快速算法，下面从稀疏分解效果方面，实验验证并讨论分析本文算法。

对于稀疏分解的结果，即稀疏表示的优劣，根据稀疏表示的含义，可以用以下标准来衡量：① 在分解结果中，给定用来表示图像的原子的数目，利用这些分解出来的原子来重建（或表示）图像，如果重建图像质量较好，则说明稀疏分解的效果较好。② 在分解结果中，确定重建图像的质量，然后依据获得该重建图像质量所需要的原子数目来确定稀疏分解的效果，获得该重建图像质量所需要的原子数目越小，则稀疏分解效果越好。

以上两个标准实质是一致的，在图像稀疏分解的文献资料中，往往用第一个标准。因此，本文也使用第一个标准来测试算法。在使用第一个标准时，遇到的一个问题是重建图像的质量如何评价。重建图像质量评价一般有主观评价与客观评价两种方法，其中主观评价由人观察图像，然后根据观察时的感觉，输出评价结果，有较大的主观性。客观评价根据重建图像与原始图像计算获得，不受观测人的主观影响，但得到的结果有时并不符合人的视觉特性。

传统的图像质量评价标准是用重建图像与原始图像的均方误差或峰值信噪比，其中峰值信噪比最为常用。均方误差（Mean Square Error, MSE）公式如下：

$$MSE = \frac{1}{M_1 \times M_2} \sum_{i=0}^{M_1-1} \sum_{j=0}^{M_2-1} [f(i, j) - \hat{f}(i, j)]^2 \quad (6.5)$$

式 (6.5) 中， $f(i, j)$  代表原始图像上第  $i$  行第  $j$  列的一个像素灰度值， $\hat{f}(i, j)$  代表重建图像上第  $i$  行第  $j$  列的一个像素灰度值， $M_1$ 、 $M_2$  分别代表图像的长和宽。最常用的峰值信噪比（Peak Signal to Noise Ratio, PSNR）如下：

$$PSNR = 10 \lg \left( \frac{G_f^2}{MSE} \right) \quad (6.6)$$

式 (6.6) 中,  $G_f$  代表图像  $f(i, j)$  上像素灰度值的最大可能值,  $R_{\text{SNR}}$  的单位是 dB。一般文献上取为  $G_f = 255$ , 这种情况实际上是默认为图像是 8 位量化的。

图 6.2 给出了基于 MP 的图像稀疏分解算法和本文利用原子的能量特性实现的基于 MP 的图像稀疏分解快速算法的稀疏分解结果。图中的图像为对  $256 \times 256$  大小的 Lena 原始图像进行稀疏分解, 分解出 500 个原子, 然后利用这 500 个原子重建的图像。图 6.2 (a) 为基于 MP 的图像稀疏分解算法利用 500 个原子重建的图像, 图 6.2 (b) 为利用原子的能量特性实现的基于 MP 的图像稀疏分解算法 500 个原子重建的图像。从重建图像上可以看出, 两种算法重建的图像质量, 主观上没有任何区别。两幅图像的  $R_{\text{SNR}}$  值也只有 0.26 dB 的差别, 这是由于解最优化问题时由遗传算法本身的随机性造成的。



(a) 标准 MP 算法分解重建图像

$P_{\text{SNR}} = 26.67 \text{ dB}$



(b) 本方算法分解重建图像

$P_{\text{SNR}} = 26.93 \text{ dB}$

图 6.2 不同算法分解重建图像的效果

需要指出的是, 图像的稀疏分解特别适合人眼的视觉特性。虽然用 500 个原子表示  $256 \times 256$  大小的 Lena 原始图像, 客观图像质量只有 26~27 dB, 但是主观图像质量却是比较好的。如果图像分解是利用正交分解的话, 重建图像质量在 26~27 dB 时, 重建图像主观质量将是难以让人接受的。这是图像稀疏分解与正交分解的一个显著的差别, 是图像稀疏分解的一个优点。

## 6.3 在低维空间实现的图像稀疏分解算法

6.2.2 节中我们利用原子的能量特性, 实现基于 MP 的图像稀疏分解的快速算法。如果从信号空间理论的角度分析, 即是把高维空间的内积计算转化为低维空间的内积计算。这一节的基本思想与上一节类似, 只是转化的方法不同而

已。在这一节中，我们利用抽样的方法，把大的图像变为小的图像，从空间理论的角度分析，即是把高维空间的问题转化为低维空间的问题，从而降低问题的复杂度，实现快速计算的目的。

### 6.3.1 利用低维空间的搜索实现高维空间的搜索

基于 MP 的稀疏分解，算法易于理解，是目前图像稀疏分解的最常用方法。但和其他的图像稀疏分解方法一样，其存在的关键问题也是计算量十分巨大。在基于 MP 的图像稀疏分解中，每一步都要完成图像或图像分解的残余在过完备库中的每一个原子上的投影计算。按公式 (6.1) 所要求，每一步分解实际上要进行的内积计算  $\langle R^k f, g_r \rangle$ （或投影运算）是一个在很高维（ $M_1 \times M_2$ ）空间的内积计算，而且要进行很多次（ $P$  次），这是 MP 图像稀疏分解计算量巨大的根本原因所在。

实质上，公式 (6.1) 实现的是在由原子张成的高维（ $M_1 \times M_2$ ）空间中，寻找、搜索一个和图像或图像的残余（相当于该空间的一个向量）最匹配的原子（相当于该空间的一个单位向量）。假设我们可以在一个低维空间上，寻找到一个最佳的“小”原子，然后再把该“小原子”放大到同图像一样的大小，把放大后的该原子作为公式 (6.1) 所寻找的最佳原子，这样我们就用在低维空间的搜索实现了在高维空间的搜索。由于搜索最佳原子过程中主要涉及内积计算，而内积计算的计算量主要依赖于空间的维数，因此，用在低维空间的搜索代替实现在高维空间的搜索，其计算量将大幅减少。

实现的具体方法如下，对图像  $f$  或图像分解的残余  $R^k f$  进行抽样，抽样后的图像或图像分解的残余分别用  $f_s$  和  $R^k f_s$  表示。把过完备原子库  $\mathcal{D} = \{g_r\}_{r \in \Gamma}$  转化为过完备原子库  $\mathcal{D}_s = \{g_{s,r}\}_{r \in \Gamma}$ ， $\mathcal{D}_s$  中的原子  $g_{s,r}$  是  $\mathcal{D}$  中原子  $g_r$  的抽样，大小与抽样图像  $f_s$  或抽样的图像分解残余  $R^k f_s$  相同。公式 (6.1) 所要表达的问题在低维空间转化为：

$$|\langle R^k f_s, g_{s,r_n} \rangle| = \sup_{r \in \Gamma} |\langle R^k f_s, g_{s,r} \rangle| \quad (6.7)$$

经过搜索得到的满足公式 (6.7) 的原子  $g_{s,r_n}$  是抽样图像  $f_s$  或抽样图像分解的残余  $R^k f_s$  的一个最好匹配。根据图像和抽样图像之间的相似性和对应关系，原子  $g_{s,r_n}$  对应的上采样（即内插）原子可以近似地被认为是与图像  $f$  或图像分解的残余  $R^k f$  最匹配的原子。这样我们就可以求得与图像或图像分解残余相匹配的最佳原子  $g_{r_n}$ ，也就实现了用低维空间的搜索代替高维空间的搜索。

### 6.3.2 存在的问题与改进工作

利用 6.3.1 中提出的方法,可以大幅度降低 MP 图像稀疏分解的计算复杂性。但问题是,过完备原子库  $\mathcal{D}_i$  中的原子  $g_i$  和过完备原子库  $\mathcal{D}$  中原子  $g$  并不存在一一对应的关系。实质上,在过完备原子库  $\mathcal{D}_i$  中搜索到的与图像  $f_i$  或图像分解的残余  $R^k f_i$  匹配的最佳原子  $g_{j_n}$ ,在高维 ( $M_1 \times M_2$ ) 空间中,严格说来,并不是对应于其上采样原子  $g_n$ ,而是对应于以  $g_n$  为中心的一个邻域。因此,在以低维空间的搜索实现高维空间的搜索时,如果以低维空间中的最佳原子  $g_{j_n}$  的上采样原子  $g_n$  作为满足公式 (6.1) 的解,有可能将导致一定的偏差,从而影响图像稀疏分解的质量。

为了纠正可能出现的偏差,我们对 6.3.1 中提出的方法进行了改进。我们并不把低维空间中的最佳原子  $g_{j_n}$  的上采样原子  $g_n$  作为公式 (6.1) 的解,而是把它作为高维空间中存在最佳原子的一个邻域的中心。在此邻域中进行二次搜索,搜索得到的最佳原子,才作为满足公式 (6.1) 的原子。这样处理将有助于提高图像稀疏分解质量,但由此也将增加额外的计算量。

### 6.3.3 实验结果与分析

实验中采用  $512 \times 512$  大小的标准 Pepper 图像。过完备库中原子的构造方法按式 (6.2)。在实验中,当仅用 6.3.1 所述方法时,基于 MP 的图像稀疏分解速度将提高 3 倍多,但图像稀疏分解质量不是很理想。采用 6.3.2 所述的改进方法,速度的提高将有所下降,但图像稀疏分解的质量有所上升。综合权衡考虑,我们认为 6.3.2 中所述的改进方法,适合于在图像稀疏分解中采用。

图 6.3 中分别给出了标准 Pepper 图像和利用上述方法把该图像分解成 500 个原子后,用这 500 个原子重建的图像。可以看出,重建图像具有很好的主观



(a) 标准 Pepper 图像



(b) 用 500 个原子重建的图像

(图像大小为  $512 \times 512$ )

图 6.3 标准 Pepper 图像及用改进方法重建的图像

图像质量。用 500 个原子即可表示  $512 \times 512$  大小的原始图像,也足以说明这种表示的稀疏性。同时,用 500 个原子即可表示原图像,也表明这种表示抓住了图像的关键视觉特征,为进行图像处理提供了极其有利的条件。

## 6.4 基于智能计算的图像 MP 稀疏分解

本节集中讨论智能计算方法在基于 MP 的图像稀疏分解中的应用问题。此部分中智能计算方法主要包括遗传算法、量子遗传算法和粒子群算法等。

### 6.4.1 基于遗传算法的图像稀疏分解

虽然图像的稀疏分解在图像处理的多个方面的应用是成功的,但图像的稀疏分解在图像处理的实际应用很难被推广而产业化。阻碍图像稀疏分解研究及应用发展的关键因素之一是图像稀疏分解的计算量十分巨大,计算时间在现有条件下令人无法忍受。本文中采用 MP 的方法进行图像的稀疏分解,在 MP 的每一步中,用遗传算法快速实现在过完备库中选取最佳的原子。根据寻找最佳原子的遗传算法的内在特点,提出了一种优化的快速算法。利用我们提出的方法,较好地解决了图像稀疏分解计算复杂这一难题。算法的有效性为实验结果所证实。

在基于 MP 的图像稀疏分解中,每一步都要完成图像或图像分解的残余在过完备库中的每一个原子上的投影计算。按公式 (6.1) 所要求,每一步分解实际上要进行的内积计算  $\langle R^k f, g_r \rangle$  是一个在很高维 ( $M_1 \times M_2$ ) 空间的内积计算,而且要进行很多次 ( $P$  次),这是 MP 图像稀疏分解计算量巨大的根本原因所在。公式 (6.1) 所代表的问题,即 MP 每一步分解所需解决的问题,实际上是一个最优化问题,直接的求解方法是全局搜索的方法,这将非常费时。对此问题,利用遗传算法可以很好地快速解决。

#### (1) 利用遗传算法实现基于 MP 的图像稀疏分解

大自然中的生物始终是人类灵感产生的重要源泉,模仿生物的形态、现象和过程,人们可以得到很多重要的启发和教诲,而遗传算法就是人类学习、模仿和研究生物界的一个重要成果。遗传算法最早由 Holland 提出,是一种仿生优化算法,以生物进化理论和遗传学为理论基础,并在科学研究和工程实践中有着广泛的应用。具体来说,它模拟自然界中生物从低级、简单,到高级、复杂的这样一个漫长而绝妙的进化过程,借鉴于达尔文的优胜劣汰、适

者生存的自然选择和 Mendel 的自然遗传的机理, 是一种高效并行全局搜索算法<sup>[9]</sup>。

由于基于 MP 的图像稀疏分解的每一步, 实际上是解由公式 (6.1) 所表示的最优化问题。我们提出利用遗传算法来解决这个问题。把定义一个原子的参数组  $\gamma$  作为待寻优参数 (或称为遗传算法中的个体或染色体), 遗传算法的编码采用直接值编码方式 (Value Encoding), 图像或图像残余与原子的内积的绝对值  $|\langle R^k f, g_\gamma \rangle|$  作为适应度函数, 利用遗传算法在过完备库中选取最匹配原子的过程可以用图 4.4 所示的流程图来表示。只是在图 4.4 中遗传算法中的个体是一个一维信号原子的参数组, 这里遗传算法中的个体是一个二维图像原子的参数组。

在图 4.4 所示的利用遗传算法求取最佳原子框图中, 进化过程达到稳定的时候, 采用生成新的初始种群的方法, 来避免遗传算法求得的解为局部最优解。但是, 由于我们进化代数的限制 (实质为计算量的限制), 我们最终所求得的解仍只能是局部最优的解 (只不过程度上更趋近于全局最优解)。因此, 这里的问题是, 如何在解的质量方面和计算的复杂性方面取得一个最佳的平衡。

## (2) 寻找最佳原子的遗传算法的进一步优化

利用遗传算法寻找最佳原子, 极大地减少了计算量, 是对公式 (6.1) 所表示的最优化问题的一个较令人满意的解决方案。但其计算量仍是很大的, 仍有必要寻求对其进行优化。以下是我们对其进行优化的一种方法。首先具体分析利用遗传算法的计算量: 在遗传算法中, 主要的计算花费在计算适应度函数上, 即计算图像或图像的残余与原子的内积的绝对值 (实际上, 如果要计算适应度函数, 首先还要利用原子参数形成原子本身)。其他包含在遗传算法中的计算量和此计算量相比, 都是微不足道的。若图像大小仍用  $M_1 \times M_2$  表示, 进行 1 次利用遗传算法寻找最佳原子, 此计算至少包括  $(N_l \times N_g \times M_1 \times M_2)$  次指数运算, 及  $O(N_l \times N_g \times M_1 \times M_2)$  次乘法运算, 其中,  $N_l$ 、 $N_g$  分别为遗传算法中一个种群中的个体数目和进化的代数。

在遗传算法中完全避免计算适应度函数是不可能的, 但为了减少计算量, 我们根据遗传算法的特点, 可以避免相当一部分适应度函数的计算, 这对提高寻求最佳原子的遗传算法的计算速度是很有意义的。我们的方法的依据是在遗传算法中的遗传特性。具体说来就是, 在遗传算法中, 同一代的个体之间, 相邻的两代的个体之间, 有一定的相同性。这种相同性在初始种群或进化不充分的种群中, 并不明显, 但在进化趋向稳定的种群中, 这种不同个体之间的相同

性非常明显。利用这种相同性，我们把先计算的个体的适应度函数值进行存储，对于后面出现的相同个体，将不再对其进行适应度函数计算，而直接利用存储的结果。

### (3) 实验结果与分析

实验中采用  $256 \times 256$  大小的标准 Lena 图像。遗传算法中采用的一个种群个体数为 21，进化的代数选为 40。过完备库中原子的构造方法按本章参考资料[1]。表 6.2 中给出了 MP 图像稀疏分解的每一步中，利用遗传算法和改进的遗传算法所需要计算图像或图像的残余与原子内积的次数。表中没有给出直接利用全局最优的方法的计算量，实际上由于一幅中等大小的图像，用以对它进行稀疏分解的原子库中原子的个数是一个天文数字，所以基于 MP 的图像稀疏分解一步的内积计算次数也相应的是一个天文数字，可见我们的方法极大地提高了 MP 图像稀疏分解的速度。

表 6.2 MP 图像稀疏分解每一步中不同方法的计算量

算 法	计算图像或图像的残余与原子内积次数
遗传算法	$21 \times 40 = 840$
改进的遗传算法	630*

\*注：由于遗传算法本身的随机性，这里的次数为平均计算次数。

图 6.4 中给出了把标准 Lena 图像分解成 500 个原子后，用这 500 个原子重建的图像。可以看出，重建图像具有较好的主观图像质量。用 500 个原子即可表示原图像，也足以说明这种表示的稀疏性。



图 6.4 用 500 个原子重建的 Lena 图像

## 6.4.2 基于量子遗传算法的图像稀疏分解的实现

量子计算利用了量子理论中有关量子态的叠加、纠缠和干涉等特性,通过量子并行计算解决经典计算中的 NP 难题。量子遗传算法 (Quantum Genetic Algorithm, QGA) 是新发展起来的一种基于量子计算原理<sup>[10, 11]</sup>的概率优化方法。它以量子计算的一些概念和理论为基础,用量子位编码表示染色体,使得一个染色体可以表达多个态的叠加,比传统 GA 更具并行性。同时,利用当前最优个体的信息更新量子旋转门可加速算法收敛,引入量子交叉和利用量子非门变异可克服早熟收敛现象<sup>[12]</sup>,具有种群规模小而不影响算法性能、收敛速度快和全局寻优能力强的特点。

### (1) 量子计算的基本概念

在量子计算机中,充当信息存储单元的物理介质是一个双态量子系统,称为量子比特 (qubit)。比如一个双能级系统或具有两个自旋态的电子、质子等。为了叙述方便,通常把自旋  $\hbar/2$  的电子作为量子比特。量子比特与经典比特不同之处,就在于它可以同时处在两个量子态的叠加态中。例如:

$$|\psi\rangle = \alpha|1\rangle + \beta|0\rangle \quad (6.8)$$

其中,  $|0\rangle$  和  $|1\rangle$  分别表示自旋向下和自旋向上态,所以一个量子比特可同时包含态  $|0\rangle$  和  $|1\rangle$  的信息;  $\alpha$ 、 $\beta$  分别是  $|1\rangle$  和  $|0\rangle$  的概率幅,且满足下列归一化条件:

$$|\alpha|^2 + |\beta|^2 = 1 \quad (6.9)$$

若干个量子比特的有序集合构成一个量子寄存器。一个二位量子寄存器有 4 个独立的态矢量  $|00\rangle$ 、 $|01\rangle$ 、 $|10\rangle$ 、 $|11\rangle$ , 它们组成四维矢量空间。因此,它可以同时包含这 4 个态的信息。类似地,三位量子寄存器的态矢量,组成 8 维矢量空间。一般,一个  $L$  位量子寄存器态矢量张起  $2^L$  维矢量空间,因而可以存储  $2^L$  个不同的经典信息。

对量子比特进行最简单的么正操作称为基本量子门。任何复杂的么正操作,都可通过量子门的“完备集合”来组合实现。而一个量子门的“完备集合”可以由少数几个基本量子门组成。现已经证明<sup>[13]</sup>, 为了对量子态执行任意么正变换,只要能够实现对单个量子比特在 Hilbert 空间中的任意转动和一对量子比特间的控制转动操作就足够了,比如一个 NOT 门和一个一位门就足以构成这样的集合。如果记一个量子比特的两个态分别为  $|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$  和  $|1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$  (称为基态), 量子 NOT 门执行的运算可以由下面的么正矩阵来表示:



$$U_{\text{NOT}} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad (6.10)$$

在  $L$  位的量子寄存器中存储了  $2^L$  个经典态。由于可以存储不同经典态的叠加态，量子计算机对叠加态的演化就是对其中各个叠加成分的演化。也就是说，对一个叠加态的演化就是对  $2^L$  个态进行演化，这就是量子计算中的高度并行性。

## (2) 量子交叉和变异

遗传算法中，交叉的作用是实现个体之间结构信息的交换。通过交换可使具有低阶、短距、高平均适应度的模式能够合并而产生高阶、高适应度的个体。量子演化过程中，所有个体朝一个目标演化，如果没有交叉操作，极有可能陷入局部最优。利用量子的相干特性，构造了各个个体的联合交叉<sup>[14]</sup>——量子交叉。具体过程如下：

- a. 将种群中全部个体随机排序；
- b. 取第一个个体的第一个基因作为新个体的第一个基因，取其相邻个体的第二个基因作为新个体的第二个基因，循环往复，直到新个体具有与父代个体相同的基因数；
- c. 以此类推，直到新的种群具有与父代种群相同的规模。

变异的作用主要在于防止“早熟”和提供算法的局部搜索能力。在量子遗传算法中，我们通过量子非门设计了一种量子变异操作。具体方法如下：

- ① 以一定的概率从种群中选取若干个个体；
- ② 对选中的个体按确定的概率确定一个或者多个变异位；
- ③ 对选中位量子比特的几率幅执行量子非门操作，即完成该量子比特的变异操作。

量子变异操作，实际上是更改了该量子比特态叠加的状态，使得原来倾向于坍缩到状态“1”的变异倾向于坍缩到状态“0”，或者相反。显然，该变异操作对染色体的所有叠加态均同时有效。

## (3) 量子旋转门旋转角的旋转策略

染色体的调整是通过量子旋转门实现的，旋转门的工作原理如下：

$$\begin{pmatrix} \alpha'_i \\ \beta'_i \end{pmatrix} = \begin{pmatrix} \cos \theta_i & -\sin \theta_i \\ \sin \theta_i & \cos \theta_i \end{pmatrix} \begin{pmatrix} \alpha_i \\ \beta_i \end{pmatrix} \quad (6.11)$$

其中， $(\alpha_i, \beta_i)$  为染色体中的第  $i$  个量子比特， $\theta_i = s(\alpha_i, \beta_i) \Delta \theta_i$ ， $s(\alpha_i, \beta_i)$  和  $\Delta \theta_i$  的值根据表 6.3 策略事先确定。

表 6.3 旋转角选择策略

$x_i$	$b_i$	$f(x) \geq f(b)$	$\Delta\theta_i$	$s(\alpha_i, \beta_i)$			
				$\alpha_i \beta_i > 0$	$\alpha_i \beta_i < 0$	$\alpha_i = 0$	$\beta_i = 0$
0	0	False	0	0	0	0	0
0	0	True	0	0	0	0	0
0	1	False	Delta	0	0	0	0
0	1	True	Delta	-1	+1	$\pm 1$	0
1	0	False	Delta	-1	-1	$\pm 1$	0
1	0	True	Delta	+1	-1	0	$\pm 1$
0	1	False	0	+1	-1	0	$\pm 1$
0	1	True	0	+1	-1	0	$\pm 1$

表 6.3 中, Delta 为每次调整的角步长。该调整策略的思想就是将个体  $q'_j$  当前的测量值的适应度  $f(x_j)$  与该个体当前的目标值的适应度  $f(b_j)$  进行比较, 如果  $f(x_j) > f(b_j)$ , 则调整  $q'_j$  中相应位量子比特 ( $x_i \neq b_j$ ), 使得概率幅对  $(\alpha_i, \beta_i)$  向着有利于  $x_i$  出现的方向演化; 反之, 如果  $f(x_j) < f(b_j)$ , 则调整  $q'_j$  中相应位量子比特 ( $x_i \neq b_j$ ), 使得概率幅对  $(\alpha_i, \beta_i)$  向着有利于  $b_j$  出现的方向演化。Delta 的值太小将影响收敛速度; 太大可能会使结果发散, 或早熟收敛到局部最优解。本文采取动态调整 Delta 的策略, 取值动态范围是  $0.01\pi - 0.08\pi$ 。

#### (4) 利用 QGA 算法寻找最佳原子的实现

由于基于匹配追踪的图像稀疏分解的每一步, 实际上可以归结为由公式 (6.1) 所表示的最优化问题, 可以利用粒子群优化算法来解决这个问题。把定义一个原子的参数组  $\gamma$  作为待寻优参数组, 图像或图像残余与原子的内积的绝对值  $\langle R^k f, g_\gamma \rangle$  作为适应度函数。适应度函数的参数取值是在一系列有限的离散整数值, 而基本的粒子群算法针对的是连续问题的求解, 故在计算适应度函数之前, 需要对参数做约束域的判断以及取整操作。用量子遗传算法寻找最佳匹配原子进行稀疏分解的过程如下:

① 初始化种群  $Q(t_0)$ : 种群中全部染色体的所有基因  $(\alpha'_i, \beta'_i)$  均被初始化为  $1/\sqrt{2}$ , 这意味着一个染色体所表达的是其全部可能状态的等概率叠加。

② 对初始种群中个体进行一次测量, 以获得确定的解。  $P(t_0) = \{p_1^0, p_2^0, \dots, p_n^0\}$ , 其中,  $p_j^t$  为第  $t$  代种群中第  $j$  个解 (第  $j$  个个体的测量值), 表现的形式为长度为  $m$  的二进制串, 其中每一位为 0 或 1 是根据量子比特的概率 ( $|\alpha'_{ij}|^2$  或  $|\beta'_{ij}|^2$ ,  $i=1, 2, \dots, m$ ) 测量得到的。

- ③ 对各状态进行适应度评估，计算各个状态的适应度函数值。
- ④ 保留最佳个体状态及其适应度值，把最佳适应度个体作为该个体下一代演化的目标值。
- ⑤ While 非结束状态 do  
 Begin  
 a.  $t = t + 1$ ;  
 b. 对种群实施一次测量  $Q(t)$ ，得到一组状态  $P(t)$ ；  
 c. 对各个状态进行适应度评估；  
 d. 根据一定的调整策略，对种群中的个体执行量子交叉和变异，利用量子旋转门  $U(t)$  对种群进行更新，得到子代群体  $Q(t+1)$ ；  
 e. 记录最佳个体状态及其适应度值；  
 End
- ⑥ 利用式 (6.12) 新图像残差，转至步骤①寻找残差图像的下一个最匹配的原子。

$$R^{k+1}f = R^k f - \langle R^k f, g_n \rangle g_n \quad (6.12)$$

### (5) 实验结果

由于图像稀疏分解的复杂度比较高，实验时采用  $256 \times 256$  大小的标准 Lena 图像的三个  $64 \times 64$  的小块，量子交叉的概率为 0.05，变异的概率是 0.75，这三个分块图像的复杂度分别是低、中、高，完备原子库的构建采用式 (6.2)。

复杂度低的图像块，种群规模取 75，迭代次数是 40，重建图像时用 10 个原子得到的图像， $P_{\text{SNR}}$  是 31.75 dB，如图 6.5 (a) 所示。复杂度中等的图像块，种群规模取 75，迭代次数是 80，重建图像时用 45 个原子得到的图像， $P_{\text{SNR}}$  是 27.52 dB，如图 6.5 (b) 所示。复杂度高的图像块，种群规模取 100，迭代次数是 100，重建图像时用 100 个原子得到的图像， $P_{\text{SNR}}$  是 27.78 dB，如图 6.5 (c) 所示。

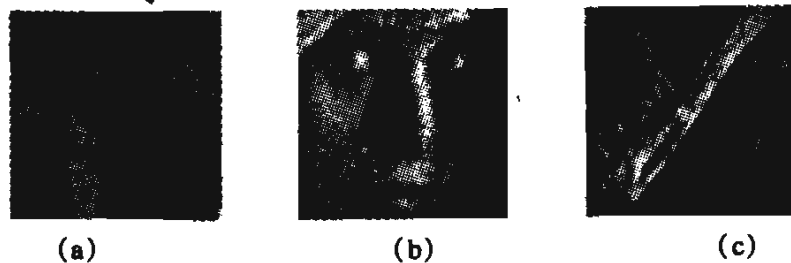


图 6.5 重建图像子块

### 6.4.3 基于粒子群算法的图像稀疏分解

针对图像稀疏分解计算量巨大的问题，本节采用 MP 的方法进行的图像稀

疏分解, 在 MP 的每一步中, 利用了粒子群优化算法快速实现在过完备库中选取次佳原子代替最佳的原子, 在实现图像的稀疏分解的同时在一定程度上避免了巨大的计算量和存储量。

在基于 MP 的图像稀疏分解的每一步都要求取图像或者残余图像在过完备库中所有原子的投影。换句话说, 每一步分解都要计算内积  $\langle R^k f, g_r \rangle$ , 而计算  $\langle R^k f, g_r \rangle$  是在一个很高维 ( $M_1 \times M_2$ ) 的空间上进行的, 而且还要计算很多次 ( $P$  次)。另外, 计算过程中需要存储整个过完备原子库, 这是 MP 图像稀疏分解计算量和存储量巨大的根本原因所在。匹配追踪过程的每一步所需要解决的问题, 实际上是一个寻优问题, 如果直接求解, 那将非常费时。对此, 退而求其次, 提出利用粒子群优化 (PSO, Particle Swarm Optimization) 算法解此最优化问题, 用次优解代替最优解, 则寻优问题可以快速地得到解决。

PSO 算法是由美国普渡大学的 Eberhart 和 Kennedy 受鸟群和鱼群群体运动行为方式启发于 1995 年提出的一种具有代表性的集群智能方法。其起源于对一个简化社会模型的仿真<sup>[15, 16]</sup>。与基于达尔文“适者生存, 优胜劣汰”进化思想的遗传算法不同, 粒子群优化算法的基本思想是通过群体中个体之间的协作和群体中信息的共享进化优势, 进而达到寻找最优解的目的。

粒子群算法在本质上是一种并行的全局性的随机搜索算法。它原理简单、容易实现、需要调整的参数少、搜索速度快、搜索范围大、精度高, 和其他优化算法相比, 优点非常突出。

### (1) PSO 的算法原理与数学描述

PSO 算法是进化智能算法, 可以用来解决优化问题。每个待优化问题的解可以抽象为搜索空间中的一只鸟, 它在搜索空间中以一定的速度飞行, 这个速度根据它本身的飞行经验和同伴的飞行经验来动态调整。鸟被抽象为没有质量和体积的微粒 (点), 在  $N$  维空间里, 第  $i$  个粒子的位置表示为矢量  $X_i = (x_1, x_2, \dots, x_N)$ , 飞行速度表示为矢量  $V_i = (v_1, v_2, \dots, v_N)$ 。每个粒子都有一个被优化的函数决定的适应值 (Fitness Value), 并且知道自己到目前为止发现的最好位置 ( $pbest$ ) 和现在的位置  $X_i$ , 这两个参数值可以理解成粒子本身的飞行经验。除此之外, 每个粒子还知道到目前为止整个群体中所有粒子发现的最好位置  $gbest$  ( $gbest$  是在  $pbest$  中的最好值), 可以理解成粒子的同伴的经验。每个粒子根据下列信息改变自己的当前位置和速度 (矢量包括大小和方向): ① 当前位置; ② 当前速度; ③ 当前位置与自己最好位置之间的距离; ④ 当前位置与群体最好位置之间的距离。也就是说, 粒子根据自己的经验和同伴中最好的经验来决定下一步的运动。

PSO 算法作为一种迭代优化工具, 对于第  $k$  次迭代, 每个粒子是按照式

(6.13)、(6.14) 进行变化的<sup>[17]</sup>。

$$v_{id}^{k+1} = w \times v_{id}^k + c_1 \times rand() \times (p_{id} - x_{id}^k) + c_2 \times rand() \times (p_{gd} - x_{id}^k) \quad (6.13)$$

$$x_{id}^{k+1} = x_{id}^k + v_{id}^{k+1} \quad (6.14)$$

如果待求解的问题在一个约束域内，还需要做式 (6.15)、(6.16) 的判断处理：

$$v_{id} = \begin{cases} V_d^{\max}, & v_{id} > V_d^{\max} \\ -V_d^{\max}, & v_{id} < -V_d^{\max} \end{cases} \quad (6.15)$$

$$x_{id} = \begin{cases} X_d^{\max}, & x_{id} > X_d^{\max} \\ X_d^{\min}, & x_{id} < X_d^{\min} \end{cases} \quad (6.16)$$

在式 (6.13)、式 (6.14) 中， $i=1,2,\dots,M$ ， $M$  是该群体中粒子的总数； $v_{id}^k$  为第  $k$  次迭代粒子  $i$  飞行速度矢量的第  $d$  维分量； $x_{id}^k$  为第  $k$  次迭代粒子  $i$  位置矢量的第  $d$  维分量； $p_{id}$  为粒子  $i$  个体最好位置  $pbest$  的第  $d$  维分量； $p_{gd}$  为群体最好位置  $gbest$  的第  $d$  维分量； $c_1$ 、 $c_2$  为学习因子； $rand()$  是随机函数，产生取值为  $[0,1]$  的随机数；粒子的搜索区域（即  $x_{id}^k$  的取值范围  $[X_d^{\min}, X_d^{\max}]$ ）由具体的问题给定。 $w$  称为惯性权重函数，一般在  $0.1 \sim 0.9$  之间取值； $v_{\max}$  为最大速度，决定粒子在一个循环中的最大移动距离，通常设定为粒子移动的范围宽度，即  $V_d^{\max} = X_d^{\max} - X_d^{\min}$ ，而  $v_{id} \in [-V_d^{\max}, V_d^{\max}]$ 。

如果从社会学的角度看，公式 (6.13) 中第二部分（粒子  $i$  当前位置与自己最好位置之间的距离）为“认知 (cognition)”部分<sup>[18]</sup>，表示粒子的动作来源于自己经验的部分；第三部分（粒子  $i$  当前位置与群体最好位置之间的距离）为“社会 (social)”部分，表示粒子的动作来源于群体中其他粒子经验的部分，表现为知识的共享和合作。粒子就是通过自己的经验和同伴中最好的经验来决定下一步的运动，这与人类的决策非常相似，人们通常也是通过综合自身已有的信息和从外界得到的信息来做出决策的。

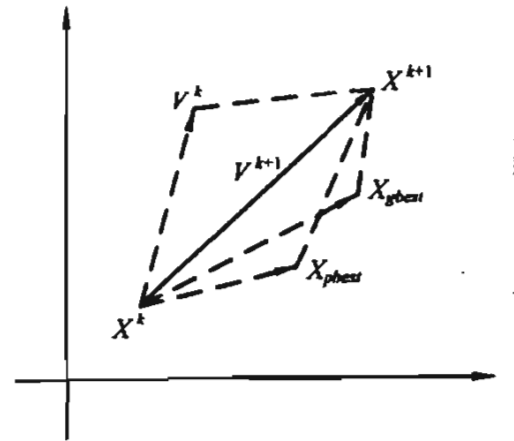


图 6.6 粒子移动原理图

在图 6.6 中，以二维空间为例描述了粒子根据式 (6.13)、式 (6.14) 从位置  $X^k$  到  $X^{k+1}$  的移动原理。

## (2) 利用 PSO 算法寻找最佳原子的实现

由于基于匹配追踪的图像稀疏分解的每一步，实际上可以归结为由公式

(6.1) 所表示的最优化问题, 可以利用粒子群优化算法来解决这个问题。把定义一个原子的参数组  $\gamma$  作为待寻优参数组, 图像或图像残余与原子的内积的绝对值  $|\langle R^k f, g_\gamma \rangle|$  作为适应度函数。适应度函数的参数的取值是在一系列有限的离散整数值, 而基本的粒子群算法针对的是连续问题的求解, 故在计算适应度函数之前, 需要对参数做约束域的判断以及取整操作。利用粒子群优化算法在过完备库中选取一个最佳匹配原子的过程如下:

① 初始化所有的粒子。在允许范围内随机设置粒子的初始位置和速度, 通常采用四舍五入的方法对粒子的位置取整操作, 每个粒子的  $pbest$  设为初始值,  $pbest$  中的最好设为  $gbest$ , 并计算每个粒子的适应度函数。

② 根据公式 (6.13) 和式 (6.14) 计算粒子新的飞行速度和位置, 计算粒子的新位置是否在约束区域内, 如果是, 则将粒子的位置更新为新计算出的值; 否则取边界值。计算粒子的新的飞行速度, 如果飞行速度的绝对值小于其最大值, 则将粒子的速度更新为新计算出的值; 否则不更新。

③ 评价每个粒子的适应值, 计算每个粒子的适应度函数, 如果优于  $pbest$ , 则  $pbest$  被当前位置替换; 如果所有粒子中有  $pbest$  优于  $gbest$ , 则重新设置  $gbest$ 。

④ 以一定的概率对种群中最好的两个个体以随机概率进行交叉, 以保持群体的优良特性。

⑤ 检查是否满足终止条件, 如果达到最大的迭代次数, 则立即终止迭代, 输出  $gbest$  和相应的位置参数, 同时利用公式的变形式 (6.17) 更新图像或者图像残差, 转向步骤①在残余图像匹配追踪下一个原子; 否则转向步骤②。

$$R^{k+1}f = R^k f - \langle R^k f, g_n \rangle g_n \quad (6.17)$$

由于粒子的惯性因子  $w$  对算法的优化性能有很大的影响,  $w$  取值较大则算法具有较强的全局搜索能力,  $w$  取值较小则算法倾向于局部搜索。在公式中引入一个惯性权重因子  $w$ , 以起到权衡全局搜索和局部搜索能力。本章参考资料 [16] 通过大量实验证明, 如果  $w$  随算法迭代的进行而线性减小, 将显著改善算法的收敛性能。设  $w_{\max}$  为惯性因子的最大值,  $w_{\min}$  为惯性因子的最小值,  $iter$  为当前迭代次数,  $iter_{\max}$  为算法迭代的总次数, 则有:

$$w = w_{\max} - iter \times \frac{(w_{\max} - w_{\min})}{iter_{\max}} \quad (6.18)$$

### (3) 实验结果及分析

实验采用  $256 \times 256$  大小的标准 Lena 图像, 过完备原子库的构建采用公式 (6.2)。粒子群优化算法中各个参数的选择依赖于具体的问题, 经过多次试验和比较, 在本实验中, 粒子群算法中的参数做如下选择: 种群交叉概率取 0.05,

学习因子  $c_1 = c_2 = 2.1$ ，初始惯性权重  $w(0) = 0.9$ ，并随着迭代次数线性递减至 0.1，种群规模是 100，以最大迭代次数 30 作为算法停止搜索的条件。

图 6.7 是本文和本章参考资料[19]在重建图像时都用 500 个原子得到的图像，图 6.7 (a) 是用本章参考资料[19]的遗传算法重建的图像， $P_{SNR}$  是 27.54 dB；图 6.7 (b) 是本文的粒子群优化算法的重建的图像， $P_{SNR}$  是 28.50 dB。很明显，用粒子群优化算法获得的图像的峰值信噪比(客观评价图像的质量)大于遗传算法。从图 6.7 (a) 与图 6.7 (b) 直观比较也可看出，本文提出用粒子群优化算法做的图像稀疏分解方法的重建图像的主观视觉质量有略优于本章参考资料[19]。



(a) 基于 GA 算法稀疏分解重建图像



(b) 基于 PSO 算法稀疏分解重建图像

图 6.7 GA 算法和 PSO 算法重建的图像

图 6.8 是粒子群算法与遗传算法在稀疏分解中寻找最佳原子后重建图像的结果示意图。从图 6.8 中也可以看出，粒子群优化算法的效果始终优于遗传算法。

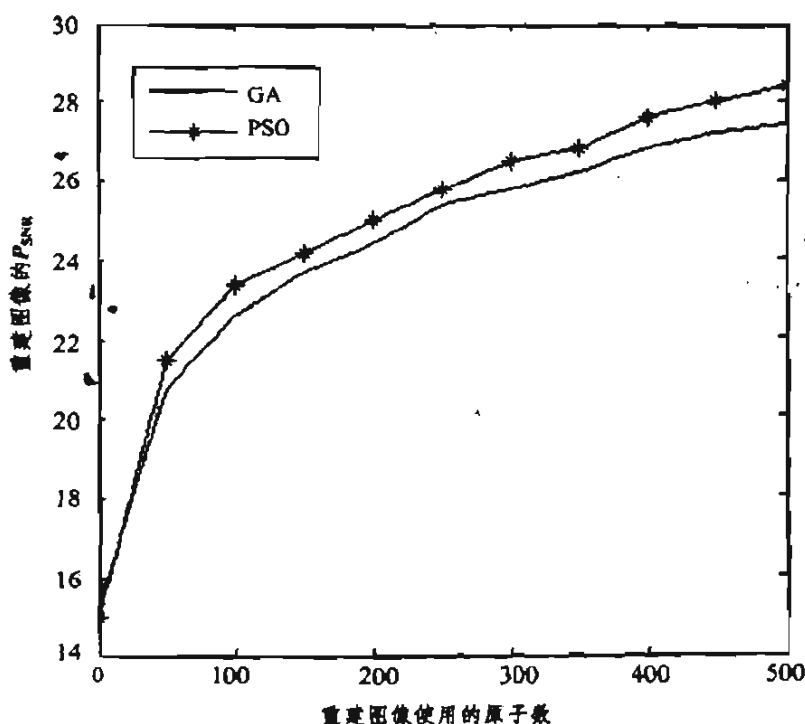


图 6.8 GA 算法和 PSO 算法重建图像质量对比

## 参 考 资 料

- [1] Vandergheynst P, Frossard P. Efficient image representation by anisotropic refinement in matching pursuit[A], Proceedings of IEEE on ICASSP[C], Salt Lake City, UT, USA, May, 2001, 3: 1757~1760
- [2] Cand E, Ridgelet: Theory and applications [D], USA: Department of Statistics, Stanford University, 1998
- [3] Cand E, Donoho D. Curvelets[R]. USA: Department of Statistics, Stanford University, 1999
- [4] Pennec E, Mallat S. Image compression with geometrical wavelets[A], Proceeding of IEEE ICIP[C], Vancouver, Canada, 2000. 661~664
- [5] Do M, Vetterli M, Contourlet[A], J Stoeckler, GV Welland. Beyond Wavelets[C]. Academic Press. 2002
- [6] Gribonval R, Nielsen M. Sparse decompositions in "incoherent" dictionaries[A], Proceedings of IEEE ICIP[C], 2003. 33~36
- [7] Mallat S, Zhang Z. Matching pursuits with time-frequency dictionaries[J], IEEE Trans. Signal Processing, 1993, 41: 3397~3415
- [8] Bergeau F, Mallat S. Matching pursuit of images [A]. Proceedings of IEEE-SP[C]. USA: Philadelphia, 1994. 330~333
- [9] L.Davis. Handbook of Genetic Algorithms, Van Nostrand, 1991
- [10] Hey T. Quantum computing: An introduction [J]. Computing & Control Engineering Journal, 1996, 10 (3): 105~112
- [11] Narayanan A. An introductory tutorial to quantum computing [A]. Proc Of IEEE Colloquium on Quantum Computing: Theory, Applications and Implications[C]. London: IEE Press, 1997. 1/1~1/3
- [12] Yang J A, Li B, Zhuann Z Q. Quantum genetic algorithm and its application research in blind source separation[J], mini-micro systems, 2003, 24 (8): 1518~1523
- [13] D Vincenzo D P. Two-bit gates are universal for quantum computation [J]. Phys, Rev, A, 1995, 51 (2): 1015~1022
- [14] Narayanan A, Moore M. Quantum inspired genetic algorithms [A]. In: Proceeding; of the 1996 IEEE International Conference on Evolutionary Computation (ICEC96), Nogaya, Japan, 1996. 41~46
- [15] J Kennedy, R C. Particle swarm optimization [A]. Proc IEEE international conference on Neural Networks[C]. USA: IEEE Press, 1995, 4: 1942~1948



- [16] Shi Y, Eberhart R. A modified particle swarm optimizer [C].In. IEEE World Congress On Computational Intelligence. 1998, 69~73
- [17] Eberhart R C, Shi Y. Particle swarm optimization: development, applications and resources[C]. In: Proc Congress on Evolutionary Computer 2001. Piscataway, NJ:IEEE Press. 2001, 81~86
- [18] Kennedy J. The particle swarm: social adaptive of knowledge [A].Proc. IEEE Int Conf. on Evolutionary Computation [C]. 1997
- [19] 尹忠科, 王建英, Vandergheynst P. 一种新的图像稀疏分解快速算法[J], 计算机应用, 2004, 24 (10): 92~96

## 第7章 稀疏分解在图像去噪中的应用

在图像处理实际应用中,由于多种因素的影响,图像上往往都含有噪声,因此有必要对图像进行去噪处理。从作用域出发,传统图像去噪方法可分为空间域图像去噪和频率域图像去噪两种方法。一般来说,图像信息主要集中在低频部分,噪声则是较均匀地分布在整个频段。传统的图像去噪方法,无论是在空间域进行还是在频率域进行,大部分都是将图像中的信息和噪声从频率上分开。但是,在许多情况下,图像信息中也有高频成分,如图像的细节和边缘等,噪声在低频部分也有一定的分量。所以在频率域图像有用信息和噪声并不是完全分开的,从而传统的图像去噪方法在去除噪声的同时,往往也对图像信息构成一定程度的损害。为了尽量减少这种损害而设计的方法,在相当程度上又使得噪声去除不够干净。常用的图像去噪方法,往往在去除噪声和保护有用图像信息之间进行艰难的平衡<sup>[1, 2]</sup>。

稀疏分解作为图像分解的一种新方法,肯定会给图像处理的许多方面带来新的思路。本章利用图像的稀疏分解,得到一种全新的图像去噪方法。基于稀疏分解的图像去噪按照是否是图像中的稀疏成分把图像中的信息和噪声分开。一个原子是有特定结构的,图像中有一定结构的成分构成有用信息,是能够用原子来表示的。但图像中的噪声是没有结构的,所以不能用原子来表示。这样可以将图像与噪声区别开来,以达到去噪的目的。

### 7.1 稀疏分解图像去噪原理

假设图像  $f$  由两部分组成:

$$f = f_s + f_n \quad (7.1)$$

其中

$$f_s = \sum_{k=0}^{n-1} \langle R^k f, g_{\gamma_k} \rangle g_{\gamma_k}$$

$$f_n = \sum_{k=n}^{\infty} \langle R^k f, g_{\gamma_k} \rangle g_{\gamma_k}$$

式 (7.1)、(7.2) 和 (7.3) 中,  $f_s$  表示图像的稀疏成分 (Sparse Component),  $f_n = f - f_s$  表示图像中的其他成分, 即图像中去除稀疏成分后的残差部分或说是噪声。

匹配跟踪算法, 是从完备原子库中选择与图像  $f$  内积最大的原子向量  $g_{\gamma_0}$ , 相当于原子向量  $g_{\gamma_0}$  与图像信号最为匹配, 即从图像信号中提取出最能匹配原始图像的原子——图像有用信息。图像  $f$  减去最佳匹配的原子向量  $g_{\gamma_0}$ , 得到图像残余  $R^1 f$ , 对图像残余  $R^1 f$  继续上面的计算过程, 提取出最佳匹配原子  $g_{\gamma_1}$ 。这样不断地迭代下去, 提取出的有用信息越来越多, 图像残余越来越少, 如图 7.1 所示。分解到某一程度, 可以近似地认为图像的有用信息被提取出来, 因此剩下的残余就是噪声, 公式 (7.2) 可以表示去噪后的图像。上面所述, 是匹配跟踪图像稀疏分解的去噪原理。

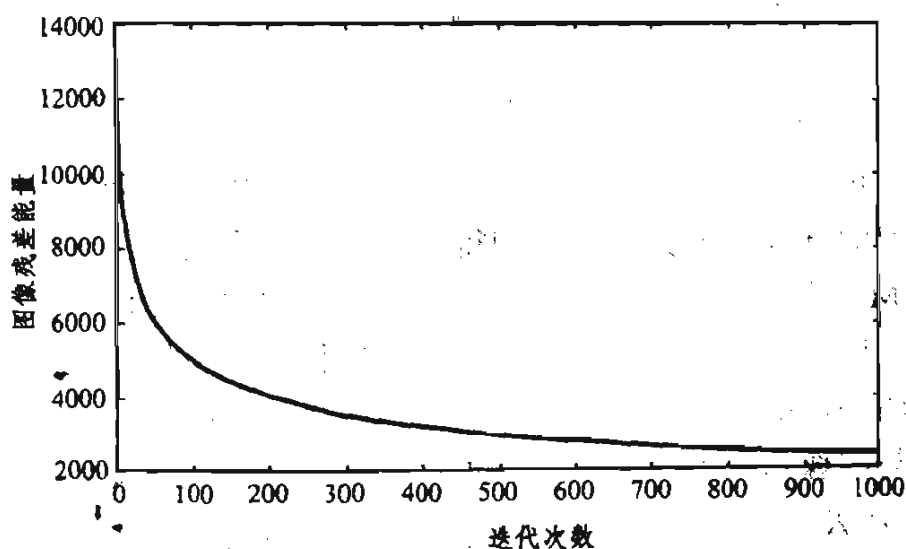


图 7.1 图像稀疏分解中残差能量的分布规律

## 7.2 稀疏分解中图像上信息与噪声的区分

由 7.1 节的匹配跟踪图像去噪原理可知, 基于稀疏分解的图像去噪方法先从图像中提取出图像的稀疏成分, 然后利用提取出的图像稀疏成分重建图像, 则重建的图像即为去除噪声后的图像。然而图像的稀疏成分和图像中的

他成分具有相对性，这体现在式 (7.2) 和式 (7.3) 中的  $n$  没有明确的取值，如果  $n$  取得太小，图像中的有用信息可能还没有被完全提取出来；如果  $n$  取得太大，则有可能将部分噪声也提取出来，被当成图像的有用信息。因此，不论上述哪种结果都将影响图像去噪效果。图像去噪中的难点正在于怎样区分图像中的有用信息与噪声。具体到基于稀疏分解的图像去噪方法，就是怎样确定式 (7.2) 和式 (7.3) 中的  $n$  值。

一个比较直观的想法是：利用残差图像的某个能量值作阈值，作为图像稀疏分解的终止条件，区分图像中的有用信息与噪声。但从图 7.1 可以看出，残差图像的能量是单调递减的，没有一个拐点，所以这个值的确定是比较困难的。本节利用相干比阈值确定区分图像中的有用信息与噪声的  $n$  值，达到图像去噪的目的。

## 7.2.1 图像稀疏分解中的相干比

根据 5.1.1 节信号稀疏分解中相干比的定义，在此可以定义图像稀疏分解中原始图像或图像残余与原子库的相干比：

$$\lambda(R^k f) = \sup_{\gamma \in \Gamma} \frac{|\langle R^k f, g_\gamma \rangle|}{\|R^k f\|} \quad (7.4)$$

其中， $R^k f$  表示图像或图像残余， $\{g_\gamma\}_{\gamma \in \Gamma}$  表示过完备原子库。

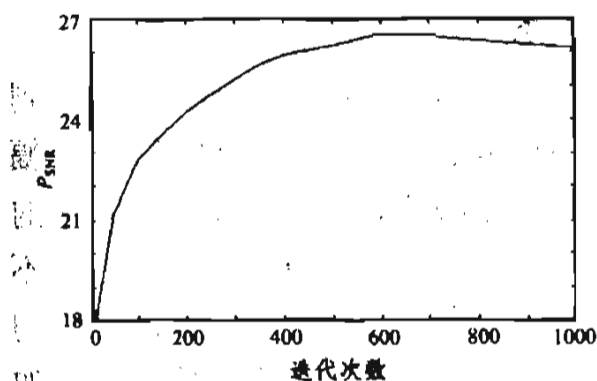
现在来分析相干比随着分解过程的变化规律。由于直接根据 MP 算法进行图像稀疏分解的计算量十分巨大，使得直接计算几乎是不可能的，所以本节算法中的基于 MP 的图像稀疏分解都是采用第 6 章中的智能计算方法来实现的。图 5.7 是信号残差与原子库的相干比随分解过程的变换规律。由图 5.7 发现，信号残余与原子库的相干比并不是递减的，而是波动的。再看看图像稀疏分解中图像残余与图像原子库的相干比（本节使用的是基于遗传算法的图像稀疏分解<sup>[3,4]</sup>，遗传算法是局部最优的），相干比曲线波动要更大一些，因此给我们分析问题带来一定的困难。为了将本章参考资料[5]中一维信号的去噪方法扩展到作为二维信号的图像去噪上，作者对图像分解中图像残余与原子库的相干比的关系做了大量的实验。

## 7.2.2 重构图像 $P_{\text{SNR}}$ 的变化 —— 确定相干比阈值的方法

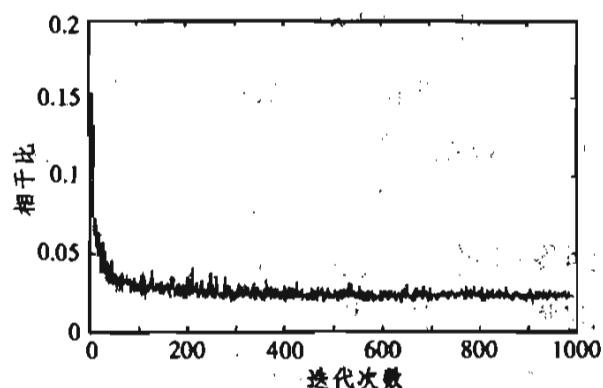
图像稀疏分解中，我们使用本章参考资料[6]给出的原子库。实验中，我们采用  $256 \times 256$  大小的 Lena 标准测试图像，将 Lena 图像加入不同级别的噪声，构造出不同的含噪声图像。对噪声图像进行匹配跟踪稀疏分解，每一步迭代都

计算重构图像的峰值信噪比 ( $P_{\text{SNR}}$ ) 和图像残余与原子库的相干比  $\lambda$ ; 跟踪其变化规律, 记录最大的  $P_{\text{SNR}}$  值和与其对应的相干比 [ $P_{\text{SNR}}$  和相干比的变化曲线分别如图 7.2 (a) 和图 7.2 (b) 所示]。

由图 7.2 (a) 可以看出, 随着图像稀疏分解的进行, 重构图像的  $P_{\text{SNR}}$  值从一个较小的值开始慢慢变大, 当分解到某一步时到达最大, 如果继续分解下去, 重构图像的  $P_{\text{SNR}}$  值又逐渐减小。通过上述分析, 我们可以认为图像稀疏分解中, 开始时提取出的是图像的有用信息; 当重构图像的  $P_{\text{SNR}}$  值到达最大时, 以后的分解提取的是残差图像的噪声, 因此重构图像的  $P_{\text{SNR}}$  值又开始减小。这样跟踪重构图像的  $P_{\text{SNR}}$  值, 可以认为  $P_{\text{SNR}}$  值最大的时候, 图像信息提取得最合适, 去噪效果最佳, 因此可以在  $P_{\text{SNR}}$  值最大的时候停止分解。



(a) 稀疏分解过程中重构图像  $P_{\text{SNR}}$  的变化曲线



(b) 稀疏分解过程中图像残余与原子库相干比的变化曲线

图 7.2 图像稀疏分解过程中  $P_{\text{SNR}}$  与相干比的变化规律

$P_{\text{SNR}}$  是我们用以衡量图像去噪效果好坏的评价标准, 只是作为分析用, 并不能依此作为结束条件 (因为实际应用中, 进行图像去噪时, 我们并不知道原始的不含噪声图像, 因此无法知道噪声图像的  $P_{\text{SNR}}$  值), 但可以通过  $P_{\text{SNR}}$  推导出结束条件。信号去噪中相干比阈值的成功应用为图像去噪创造了条件, 我们可以用图像残余与原子库的相干比作为结束条件。对  $256 \times 256$  大小含有不同程度噪声的 Lena 图像, 在跟踪过程中发现, 对应于  $P_{\text{SNR}}$  值最大的相干比在某一个小的范围内波动。为了选择一个合适的相干比阈值, 在这个小的波动范围内取一组相干比阈值作为结束条件, 选择去噪效果最好的一个相干比阈值作为最佳阈值。

### 7.2.3 利用实验方法研究用于图像去噪的相干比阈值的选取

对于一幅大小为  $256 \times 256$  的 Lena 含噪声图像, 跟踪图像稀疏分解中重构图像  $P_{\text{SNR}}$  的变化情况以及相干比与  $P_{\text{SNR}}$  的对应关系。选择与最大  $P_{\text{SNR}}$  值相对应的相干比作为结束条件, 称为相干比阈值。然而, 本实验中, 我们对图像的稀疏分

解是用遗传算法来实现的，相干比的变化波动比较大，所以相干比阈值的选择并不能简单地认为是与最大  $P_{\text{SNR}}$  值相对应的相干比，而是在与最大  $P_{\text{SNR}}$  值相对应的相干比的一个小的波动范围内来选取。具体方法是，在这个小的波动范围内，选择一组相干比作为分解的结束条件，重构图像的  $P_{\text{SNR}}$  值最大时所对应的一个相干比作为相干比阈值。本实验中，对  $P_{\text{SNR}}=18.70$  dB 的噪声图像，重构图像的最大  $P_{\text{SNR}}=25.68$  dB，其所对应的相干比为 0.018。但是如果我们以 0.018 作为相干比阈值对图像进行分解重构，则重构图像的  $P_{\text{SNR}}=24.79$  dB；而以 0.015 作为相干比阈值的话，则重构图像的  $P_{\text{SNR}}=25.42$  dB，这与图像最佳降噪点对应的最大  $P_{\text{SNR}}$  值很接近。所以这里选择相干比阈值为 0.015 较合适。造成这种情况的原因在于图像稀疏分解过程中，相干比的变化并不是递减的，而是波动的。

### (1) 相干比阈值的选择与噪声水平没有关系

对于其他的噪声水平，相干比阈值的选择是不是不同的呢？如果不同，到底有什么规律呢？为了研究这个问题，本实验选取了 5 个相干比阈值和 15 幅含不同噪声水平的 Lena 图像，对于每一个相干比阈值，计算与此阈值对应的这组图像的  $P_{\text{SNR}}$  值。通过比较分析得到的这 5 组  $P_{\text{SNR}}$  值，可以认为对应于不同的噪声水平，相干比阈值选择为  $\lambda_c=0.015$  比较合适。这就是说稀疏分解的图像去噪方法，其相干比阈值的选择与图像的噪声水平没有关系。实验结果如表 7.1 所示，图像去噪的主观效果，如图 7.3 所示。

表 7.1 256×256 大小的 Lena 图像相干比阈值为 0.015 的去噪效果与最佳处理结果

被处理图像 $P_{\text{SNR}}/\text{dB}$	最佳处理结果 $P_{\text{SNR}}/\text{dB}$	阈值处理结果 $P_{\text{SNR}}/\text{dB}$
24.55	28.81	28.70
23.98	28.33	28.32
23.02	27.93	27.81
22.24	27.62	27.42
21.62	27.12	27.06
21.02	26.76	26.76
20.50	26.53	26.44
20.07	26.28	26.14
19.65	26.24	25.93
19.28	26.03	25.70
18.69	25.68	25.36
17.82	25.36	24.73
17.23	25.03	24.35
16.30	24.59	23.70
15.57	24.15	23.14



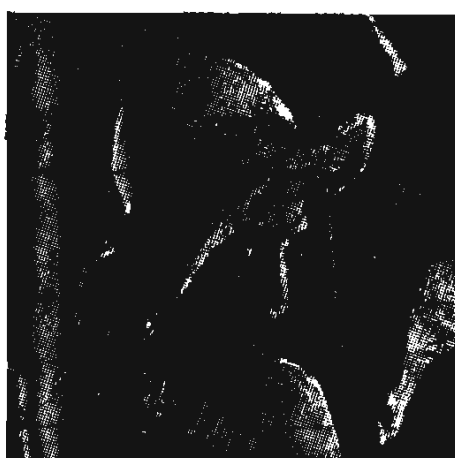
(a) 原始图像



(b) 含噪声图像



(c) 最佳去噪结果图



(d)  $\lambda_k = 0.015$  的去噪结果

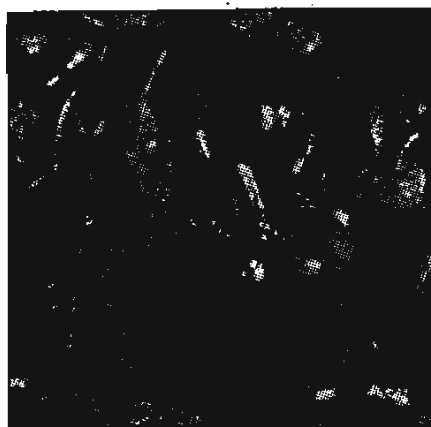
图 7.3 原始图像和去噪结果

## (2) 相干比阈值的选择与图像的类型无关

相干比阈值的选择是不是针对某一幅特定图像的呢？为了说明这个问题，此处选取了其他几幅标准测试图像（图 7.4 是实验中用到的标准测试图像）进行仿真，结果数据如表 7.2 所示。通过实验我们发现，相干比阈值的选择与图像的类型也是无关的。



(a) Cameraman 图像



(b) pepper 图像



(c) house 图像



(d) woman 图像

图 7.4 几种标准测试图像

表 7.2 256 × 256 大小的其他标准测试图像 (相干比阈值为 0.015)

图像类型	被处理图像 $P_{\text{SNR}}/\text{dB}$	最佳处理结果 $P_{\text{SNR}}/\text{dB}$	阈值处理结果 $P_{\text{SNR}}/\text{dB}$
cameraman 图像	23.66	26.82	26.64
	21.37	25.60	25.56
pepper 图像	24.02	27.10	26.69
	21.84	25.60	25.56
house 图像	24.00	26.26	25.91
	21.59	25.10	25.28
woman 图像	24.01	23.87	23.44
	21.60	22.56	22.48

既然相干比阈值的选择与图像类型或噪声水平没有关系,那么我们可以说,基于稀疏分解的图像去噪方法与图像和噪声的统计特性无关,这正说明了稀疏分解图像去噪方法的自适应性。

### (3) 相干比阈值的选择与图像大小有关

对  $512 \times 512$  大小和  $128 \times 128$  大小的图像做了大量的实验,通过实验可以得出与  $256 \times 256$  大小的图像同样的实验结果:相干比阈值的选择与图像类型和噪声水平没有关系。只是  $512 \times 512$  大小的图像,其相干比阈值选择要小一些,  $\lambda_c = 0.008$ ;而  $128 \times 128$  大小的图像,其相干比阈值选择得要大一些,  $\lambda_c = 0.022$ 。具体结果如表 7.3、表 7.4 所示。



表 7.3 128 × 128 大小的 Lena 标准图像 (相干比阈值为 0.022)

被处理图像 $P_{\text{SNR}}/\text{dB}$	最佳处理结果 $P_{\text{SNR}}/\text{dB}$	阈值处理结果 $P_{\text{SNR}}/\text{dB}$
24.59	26.36	26.24
24.00	25.77	25.76
22.23	24.97	25.02
21.50	24.78	24.71
20.51	24.40	24.29
19.77	23.84	23.83
18.66	23.33	23.28
17.28	22.75	22.54
16.27	22.24	22.21

表 7.4 512 × 512 大小的 Lena 标准图像 (相干比阈值为 0.008)

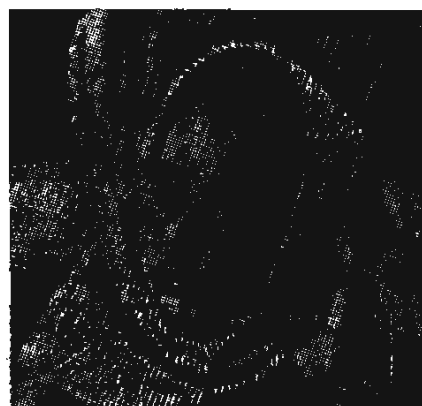
被处理图像 $P_{\text{SNR}}/\text{dB}$	最佳处理结果 $P_{\text{SNR}}/\text{dB}$	阈值处理结果 $P_{\text{SNR}}/\text{dB}$
22.25	28.28	28.19
21.58	28.13	28.13
21.03	28.00	27.87
20.50	27.97	27.84
20.09	27.88	27.80
19.65	27.78	27.66
18.64	27.30	27.20
17.86	27.00	26.84
16.29	26.78	26.66

#### 7.2.4 稀疏分解图像去噪的局限性

表 7.2 中的 woman 图像,其噪声图像比处理后图像的  $P_{\text{SNR}}$  值还要大,这是因为 woman 图像的纹理性比较明显,这种类型的图像其结构与原子库的相干性小,在图像稀疏分解中将其作为噪声来处理,因此效果并不好;也就是说稀疏分解图像去噪具有一定的局限性。但是我们看到去噪处理的视觉特性还不错(见图 7.5),这正验证了稀疏分解具有良好的视觉特性。



(a) 噪声图像



(b) 阈值去噪图像

图 7.5 woman 图像

### 7.3 稀疏分解图像去噪与最佳模板去噪的比较

将稀疏分解的图像去噪与本章参考资料[7]所提出的平滑去噪的最佳模板去噪方法进行比较（如表 7.5、7.6、7.7 所示）。可以看出噪声图像的  $P_{\text{SNR}}$  值较

表 7.5  $256 \times 256$  大小的 Lena 图像去噪效果比较

被处理图像 $P_{\text{SNR}}/\text{dB}$	最佳模板图像去噪 $P_{\text{SNR}}/\text{dB}$	稀疏分解图像去噪 $P_{\text{SNR}}/\text{dB}$
24.55	28.70	27.68
23.98	28.32	27.28
23.02	27.81	27.12
22.24	27.42	27.05
21.62	27.06	26.44
21.02	26.76	26.31
20.50	26.44	26.26
20.07	26.14	26.17
19.65	25.93	26.03
19.28	25.70	25.73
18.69	25.36	25.42
17.82	24.73	25.01
17.23	24.35	24.85
16.30	23.70	24.27
15.57	23.14	23.86

大时,就客观评价标准( $P_{\text{SNR}}$ 准则)来说,基于稀疏分解的图像相干比阈值噪声方法不如最佳模板效果好;而噪声水平较高时,稀疏分解图像去噪的效果好于最佳模板的去噪效果。然而从视觉效果来看,即使基于稀疏分解的图像去噪结果的  $P_{\text{SNR}}$  值要小于最佳平滑模板的图像处理结果的  $P_{\text{SNR}}$  值,但稀疏分解图像去噪的视觉效果仍要好于最佳平滑模板去噪的效果,图 7.6、7.7、7.8 是种方法的图像去噪视觉效果图(针对的是  $256 \times 256$  大小的 Lena 噪声图像)。

表 7.6  $128 \times 128$  大小的 Lena 图像去噪效果比较

被处理图像 $P_{\text{SNR}}/\text{dB}$	最佳模板图像去噪 $P_{\text{SNR}}/\text{dB}$	稀疏分解图像去噪 $P_{\text{SNR}}/\text{dB}$
22.23	25.50	25.02
21.50	25.16	24.77
20.95	24.87	24.57
20.12	24.35	23.94
19.28	23.94	23.83
18.67	23.53	23.56
17.72	23.03	22.79
16.27	22.18	22.34
15.64	21.72	21.98

表 7.7  $512 \times 512$  大小的 Lena 图像去噪效果比较

被处理图像 $P_{\text{SNR}}/\text{dB}$	最佳模板图像去噪 $P_{\text{SNR}}/\text{dB}$	稀疏分解图像去噪 $P_{\text{SNR}}/\text{dB}$
22.25	29.65	28.19
21.58	29.20	28.13
21.03	28.78	27.87
20.50	28.39	27.84
20.09	28.11	27.80
19.65	27.77	27.66
18.64	26.92	27.20
17.86	26.28	26.84
16.29	24.92	26.66



(a) 最佳模板去噪 ( $P_{\text{SNR}} = 27.81 \text{ dB}$ )



(b) 稀疏分解去噪 ( $P_{\text{SNR}} = 27.12 \text{ dB}$ )

图 7.6 噪声图像  $P_{\text{SNR}}$  值为 23.02 dB 时去噪效果比较



(a) 最佳模板去噪 ( $P_{\text{SNR}} = 27.06 \text{ dB}$ )



(b) 稀疏分解去噪 ( $P_{\text{SNR}} = 26.44 \text{ dB}$ )

图 7.7 噪声图像  $P_{\text{SNR}}$  值为 21.62 dB 时去噪效果比较



(a) 最佳模板去噪 ( $P_{\text{SNR}} = 26.14 \text{ dB}$ )



(b) 稀疏分解去噪 ( $P_{\text{SNR}} = 26.14 \text{ dB}$ )

图 7.8 噪声图像  $P_{\text{SNR}}$  值为 20.07 dB 时去噪效果比较

## 参 考 资 料

- [1] R. C. Gonzalez, R. E. Woods, 阮秋琦, 阮宇智等译. 数字图像处理. 北京: 电子工业出版社, 2005
- [2] K. R. Castleman, 朱志刚等译. 数字图像处理. 北京: 电子工业出版社, 2000
- [3] 尹忠科, 王建英, Vandergheynst P. 在低维空间实现的基于 MP 的图像稀疏分解[J]. 电讯技术, 2004, 44 (3): 12~15
- [4] 尹忠科, 王建英, Vandergheynst P. 一种新的图像稀疏分解快速算法[J]. 计算机应用, 2004, 24 (10): 92~96
- [5] 张文耀. 基于匹配跟踪的低位率语音编码研究[D]. 北京: 中国科学院研究生院(软件研究所), 2002
- [6] Vandergheynst P, Frossard P. Efficient image representation by anisotropic refinement in matching pursuit[A], Proceedings of IEEE on ICASSP[C], Salt Lake City, UT USA, May, 2001, 3: 1757~1760
- [7] 姜玉亭, 尹忠科. 图像平滑去噪最佳模板的选取[J]. 交通运输工程与信息学报, 2004, 2 (3): 68~71

# 第 8 章 稀疏分解在图像低比特率 压缩中的应用

## 8.1 引 言

目前, 数字图像在科学研究、经济建设和日常生活的各个方面都得到了广泛的使用。数字图像的数据量特别巨大, 所以在各个方面的应用中, 对图像数据进行压缩是非常必要的。图像压缩作为多媒体技术的关键环节, 一直是信息处理技术中研究最活跃的领域之一。随着社会信息化程度的提高, 图像信息的应用将越来越广泛, 因此对图像压缩进行更加深入的研究具有特别重要的意义。

对图像数据压缩的研究, 可以追溯到数字图像处理学科的产生。20 世纪末到现在, 图像数据压缩研究及应用取得了长足的进展, 并已经形成几种重要的国际图像压缩标准 (包括静态图像压缩标准和运动图像压缩标准, 如 JPEG、JPEG2000、MPEG4、H.264 等)。但是, 目前在低比特率条件下的图像压缩仍是有待做进一步研究的难题。在不同的国际静态图像压缩标准 (如 JPEG, JPEG2000) 中, 在低比特率条件下 (如静止灰度图像, 压缩倍数大于 64 倍, 或比特率小于 0.125 bpp 时), 压缩后重建图像质量都不太令人满意。为了在低比特率条件下获得较好的解压图像质量, 对此条件下的图像压缩编码方法进行重新认识和研究是非常必要的。低比特率图像压缩的新研究进展, 将克服相关产业 (如无线视频通信) 中的技术瓶颈, 强有力地推动其发展, 具有巨大的经济效益和社会效益。

图像压缩编码效果依赖于三个方面: 图像变换、系数量化和编码。其中图像变换最具决定性作用。例如, JPEG 中变换是 DCT, 所以称其为基于 DCT 的压缩编码方法; JPEG2000 中变换是 WT, 所以称其为基于 WT 的压缩编码方法。这样称呼国际图像压缩标准, 说明图像变换在图像压缩编码方法中起着决定性

的作用。为了实现在低比特率的情况下的图像压缩编码,必须从全新的角度对图像变换(或图像表示方法)进行研究。目前图像压缩编码中采用的变换均为正交变换,这使得图像的表示显得不够精简,从而限制了图像压缩的压缩比。为了从根本上解决图像低比特率压缩问题,研究图像的简洁表示是十分必要的。

近年来兴起的图像稀疏分解是一种非正交的分解,分解结果(即图像的稀疏表示)非常简洁。另外,图像的稀疏表示非常符合人眼的视觉特性,在相同的峰值信噪比( $P_{\text{SNR}}$ )的条件下,稀疏分解表示的图像比正交分解表示的图像具有更好的主观视觉效果(参见第6章、第7章中的众多稀疏分解的重建图像)。图像稀疏分解的优良特性,使其成为进行低比特率图像压缩的新途径。基于稀疏分解的图像压缩已经成为实现低比特率图像压缩最有潜力的可能解决方案。

## 8.2 基于稀疏分解的图像压缩国内外研究现状及分析

为了从根本上探索图像压缩方法,有必要从图像处理中的最底层问题,即图像表示方法出发展开研究。在数字图像处理和编码的理论研究和实际工程应用中,数字图像的表达(representation)或分解(decomposition)研究是一个关键性的环节,因而具有非常重要的意义。经典的 Fourier 分解及其变种 DCT 分解在图像处理和编码中有着重要的应用,并曾经有力地推动了图像处理和编码的发展,形成了基于 DCT 的国际图像压缩标准 JPEG。但是经典的 Fourier 分解用以表示图像时,只能把图像信号分解成一个个具有不同强度和不同频率的分量的组合,而不能给出频率分量在图像上出现的位置,即这种表示没有给出图像的空间特性。图像的小波分解很好地解决了这个问题,它对图像的表达既含有图像的频率特性,又含有图像的空间特性。国际图像压缩标准 JPEG2000 主要就是基于小波变换的。图像小波分解的局限性在于,在进行小波分解时,常规的小波基是确定的,这限制了小波分解的灵活性。特别是用小波变换表示图像数据并不是最佳的或最为“稀疏的”。为了实现对信号更加灵活、更加简洁和自适应的表示,在小波分析的基础上, Mallat 和 Zhang<sup>[1]</sup>总结前人研究成果,于 1993 年提出了信号在过完备原子库上分解的思想。通过信号在过完备原子库上的分解,用来表示信号的基可以自适应地根据信号本身的特点灵活选取。分解的结果,将可以得到信号的一个非常简洁的表达,即稀疏表示。而

到信号稀疏表示的过程称为信号的稀疏分解。由于信号稀疏表示的优良特性,使信号稀疏表示的研究很快从一维信号推广到作为二维信号的图像表示研究上。1994年, Mallat 等人提出了图像稀疏分解的 Matching Pursuit (MP) 方法<sup>[2]</sup>。殊途同归, Coifman 和 Wickerhauser 等人从另外角度也提出了稀疏分解的初步概念<sup>[3]</sup>。人们往往从数学上证明图像信息表示时的稀疏性,但由于图像信息本身是视觉信息,只从数学方面进行证明是不太有说服力的。1996年,《Nature》等著名刊物发表了在生物视觉的初级过程中找到这种“过完备——稀疏”表达的证据的文章<sup>[4, 5]</sup>,从而从另一个侧面推动了图像稀疏分解的研究进展。

目前,国际上针对图像稀疏分解的研究主要集中在以下三个方面:

- ① 用于进行图像稀疏分解的、适合于图像内容表述的过完备原子库;
- ② 在原子库形成基础上的图像稀疏分解算法;
- ③ 图像稀疏分解结果在图像处理各个方面的应用。

针对过完备原子库,为了更好地表示图像内容,已经提出了非对称原子库<sup>[6]</sup>等,而新发展的 Ridgelet、Curvelet、Bandelet 和 Contourlet<sup>[7~10]</sup>等也都可以作为原子模型而形成原子库。多个原子库组合应用是一个有前景的研究方向,库的结构特性也正在引起人们越来越多的关注<sup>[11]</sup>。针对稀疏分解过程,已经发展了多种算法,如 MP 算法、BP 算法、MOF 算法和 BOB 算法等<sup>[1, 2, 3, 12, 13]</sup>,其中最常用的即为 MP 算法及其变种 Orthogonal Matching Pursuit (OMP) 算法<sup>[14]</sup>。而图像稀疏分解也已经被成功应用于图像处理的多个方面,如图像压缩、图像去噪和识别等方面<sup>[2, 15~18]</sup>。特别是在图像压缩方面,无论是静态图像压缩,还是运动图像压缩,图像稀疏表示都显示出了巨大潜力,在低比特率条件下(对图像序列而言,在低数码率条件下)很多研究结果都优于现存的图像压缩国际标准<sup>[16~8]</sup>。

虽然图像的稀疏分解在图像压缩编码方面的应用研究是成功的,但图像的稀疏分解在图像压缩中的实际应用目前还很难被推广而产业化。阻碍图像稀疏分解研究及应用发展的最关键因素是:图像稀疏分解的计算量十分巨大,计算时间在现有条件下令人无法忍受。国内有研究人员<sup>[19]</sup>指出,一维数字信号长度仅为 1024 采样点时,信号稀疏分解的难度已经十分巨大。对于一般图像而言,其数据量远远大于 1024,图像稀疏分解的计算量之大可见一斑。阻碍图像稀疏分解研究及应用发展的另一关键因素,是用以进行图像稀疏分解的过完备原子库的结构和构造,这是实现图像稀疏分解的基础和关键。为了得到最稀疏的图像分解结果,往往要求原子库包含尽可能多的波形,这将导致分解难度的加剧;但如果原子库包含的波形不够,则会导致分解结果不好。这是稀疏分解中突出的一对矛盾。



但这些问题并没有能够阻碍人们对图像稀疏分解的研究热情,国际图像处理年会 (IEEE ICIP) 从 2003 年开始均对图像稀疏分解的研究进展进行专题讨论<sup>[20]</sup>。其他国际著名刊物如《Signal Processing》也分别对图像稀疏分解方法以专刊的形式进行追踪报道。

对图像稀疏分解,国内研究相对较少<sup>[21~24]</sup>。比较而言,国内关于信号稀疏分解的研究近年已经开始增加<sup>[25~27]</sup>。并且国家自然科学基金 2005 年资助了两项关于信号稀疏分解或信号稀疏表示的课题,但我们没有检索到资助图像稀疏分解或图像稀疏表示研究的信息。我们在信号稀疏分解方面和图像稀疏分解方面的研究也取得初步的进展<sup>[28~34]</sup>。

## 8.3 基于稀疏分解的图像压缩方法

图像稀疏分解是一种新的图像分解方法,可以将图像分解为很简洁的近似表达形式。本节在图像稀疏分解的基础上,对分解出的数据进行压缩。通过对分解结果数据分布规律进行分析,提出了量化和编码的方案,期望在低比特率情况下,得到较好的解码图像质量。实验结果表明,本节基于稀疏分解的图像压缩方法得到的解压图像有较好的主观图像质量。

### 8.3.1 图像稀疏分解的结果——图像的稀疏表示

设研究的图像为  $f$ ,  $f$  大小为  $M_1 \times M_2$ ,  $M_1$  和  $M_2$  为图像的长和宽。设  $D = \{g_\gamma\}_{\gamma \in \Gamma}$  为用于图像稀疏分解的过完备库,  $g_\gamma$  为由参数组  $\gamma$  定义的原子,用不同方法构造原子,参数组  $\gamma$  所含有的参数及参数个数也不一样。原子  $g_\gamma$  的大小与图像本身大小相同,但原子应作归一化处理,即  $\|g_\gamma\| = 1$ 。 $\Gamma$  为参数组  $\gamma$  的集合。由库的过完备性可知,参数组  $\gamma$  的个数应远远大于图像的大小,即若用  $P$  表示过完备库  $D = \{g_\gamma\}_{\gamma \in \Gamma}$  中原子的个数,则有  $P \gg M_1 \times M_2$ 。通过图像稀疏分解,可以得到图像的一个线性表示<sup>[1~2]</sup>:

$$f = \sum_{k=0}^{\infty} \langle R^k f, g_n \rangle g_n \quad (8.1)$$

式中,  $\langle R^k f, g_n \rangle$  表示图像  $f$  或图像的残余  $R^k f$  在对应原子  $g_n$  上的分量。由于  $\|R^n f\|$  的衰减特性<sup>[1, 2]</sup>, 用少数的原子 (与图像大小相比较而言) 就可以表示图像的主要成分, 即:

$$f \approx \sum_{k=0}^{n-1} \langle R^k f, g_n \rangle g_n \quad (8.2)$$

其中,  $n \ll M_1 \times M_2$ 。公式 (8.2) 和条件  $n \ll M_1 \times M_2$  集中体现了稀疏表示的思想。

图像稀疏分解的关键难题是计算量十分巨大。我们利用遗传算法和空间转换的方法, 把图像稀疏分解的速度提高了 2 个数量级以上<sup>[33, 34]</sup>。在本书中, 这部分内容参考第 6 章中的快速算法部分。本节中的量化编码是在本章参考资料 [33, 34] 和第 6 章的基础上的进一步工作。

图像  $f$  稀疏分解的结果, 即图像  $f$  的稀疏表示, 一般以公式 (8.2) 来表达。基于稀疏分解的图像压缩, 就是对图像用此种方式表达时的数据进行量化和编码, 以达到图像压缩的目的。为了说明下文中的数据量化和编码, 先对图像的稀疏表示即式 (8.2) 进行具体说明。非对称原子在图像稀疏表示中显示出良好性能, 所以我们在稀疏分解中采用非对称原子<sup>[6]</sup>。非对称原子的基本形式表示如下 (没有经过归一化处理):

$$g(x, y) = (4x^2 - 2)e^{-(x^2 + y^2)} \quad (8.3)$$

通过对基本非对称原子进行旋转、平移和伸缩变换, 可以得到一系列原子  $g_\gamma$ , 从而形成原子库  $\mathcal{D} = \{g_\gamma\}_{\gamma \in \Gamma}$ 。

$$g_\gamma = g_\theta \left( \frac{x-u}{s_x}, \frac{y-v}{s_y} \right) \quad (8.4)$$

其中,  $\gamma = (\theta, u, v, s_x, s_y)$ , 而  $\theta$ 、 $u$ 、 $v$ 、 $s_x$  和  $s_y$  分别代表原子的旋转、原子在  $x$ 、 $y$  方向上的平移以及原子在  $x$ 、 $y$  方向上的伸缩 (或称为尺度)。由式 (8.2)、(8.3) 和 (8.4) 可知, 图像稀疏分解的最终结果数据为:

$$\{(\langle R^k f, g_n \rangle, \theta_k, u_k, v_k, s_{xk}, s_{yk}) | k = 0, 1, 2, \dots, n-1\} \quad (8.5)$$

基于稀疏分解的图像压缩, 就是研究由图像稀疏分解得到的数据 (8.5) 式, 找出其分布范围和规律, 然后按照其分布范围和规律进行量化和编码, 以实现数据压缩的目的。

### 8.3.2 图像稀疏分解结果数据的分布规律

为了对图像稀疏分解的最终结果进行量化和编码, 需要分析其分布范围和分布规律。稀疏分解结果数据的分布规律取决于分解所使用的算法和原子, 本节使用 MP 算法<sup>[1, 2, 33, 34]</sup>和非对称原子<sup>[6]</sup>, 因此分解出的结果的数据为式 (8.5)。

进一步分析可发现,随着 MP 迭代过程的进行,图像中有意义的内容在逐渐减少,因此  $\langle R^k f, g_n \rangle$  的值应该随着迭代过程逐渐减小。在迭代过程的最初阶段 MP 分解过程总是首先捕捉图像中空间分布范围较大的信息;随着迭代的进行图像中信息逐渐减少,MP 分解过程开始捕捉图像中的细节信息,所以伸缩分量  $s_x$  和  $s_y$  随着迭代进行呈衰减分布。因为图像的信息在图像的绝大部分位置随机分布,所以匹配图像信息的原子的位置也呈随机分布,即平移分量  $u$ 、 $v$  的取值呈均匀分布。

为了验证上述分析,本节将分解出的各个参数分别按顺序进行排列,其横轴为原子数,也就是迭代次数,纵轴表示相应分量的取值,如图 8.1 所示。

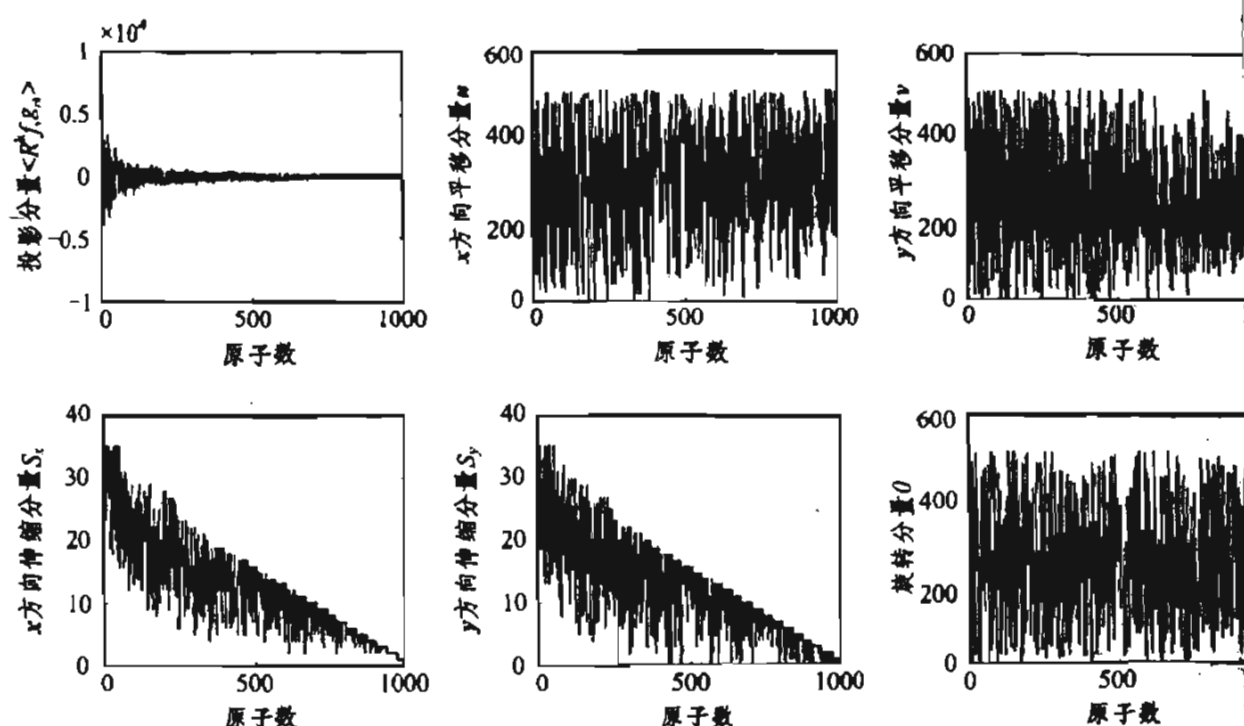


图 8.1 稀疏分解结果原子参数分布规律

从图中可以看到各个参数分布规律:旋转角度  $\theta$  和平移量  $u$ 、 $v$  呈均匀分布,其取值范围和图像的大小相同,即:从 0 到图像的长度或宽度。投影分量  $\langle R^k f, g_n \rangle$  ( $k=0,1,2,\dots,n-1$ ) 的取值范围变化较大,但是随着图像分解进程,随着  $k$  从 0 变化到  $n-1$ ,其取值趋于 0。原子的伸缩  $s_x$ 、 $s_y$  的变化规律类似  $\langle R^k f, g_n \rangle$ , ( $k=0,1,2,\dots,n-1$ ) 的变化规律,亦随着  $k$  从 0 变化到  $n-1$ ,其取值趋于 0;区别在于  $\langle R^k f, g_n \rangle$  可取负值,而  $s_x$ 、 $s_y$  总是取正值。从图 8.1 可试验结果与理论分析一致。实际上,上面所有参数的分布规律一方面是由稀疏分解内在规律决定的<sup>[1, 2, 6]</sup>,另一方面是由本节使用的分解程序决定的<sup>[33, 34]</sup>。特别是,为了提高稀疏分解的速度,本节中使用的方法对稀疏分解程序加以特殊限制,这是造成伸缩分量上限按线性衰减的原因。

### 8.3.3 图像稀疏分解结果数据的量化和编码

根据图像稀疏分解结果的分布规律，我们设计了对分解结果的量化编码方案。该方案采用均匀量化，当原始图像大小为  $512 \times 512$ 、8 bit 量化时，图像稀疏分解结果中每个参数量化编码方案如表 8.1 所示。

表 8.1 图像稀疏分解结果数据的量化方案

参 数	$\langle R^k f, g_n \rangle$	$\theta$	$u$	$v$	$S_x$	$S_y$
量化位长 (bit)	14	10	10	10	6	6

由于参数  $\langle R^k f, g_n \rangle$ 、 $S_x$  和  $S_y$  有图 8.1 所示的变化规律，所以在量化编码的过程中，可以据此进行改进。即以上三个参数的量化位长随着编码的进行不断地变小。这种量化位长的动态调整，可以达到提高压缩比的目的。

### 8.3.4 实验结果与结论

实验中我们采用  $512 \times 512$  大小、8bit 量化的标准 Lena 测试图像。稀疏分解算法采用 MP 算法，原子库采用非对称原子库。实验分为两部分：第一部分是量化和编码的实验；第二部分是本节方法和最新发表本章参考资料[16]上均为基于稀疏分解的图像压缩编码方法效果的对比实验。

在匹配追踪算法中，迭代过程的停止准则是一个重要的问题，它影响着图像表示的精度和数据压缩的效率。由于本节的压缩编码方法并不只是针对匹配追踪算法设计的，实际上它对所有的图像稀疏方法（如 BP 等）的分解数据均有效，所以本节对匹配追踪算法中迭代过程的停止准则不做重点分析。我们已经研究了此问题，认为原子库的相干性可作为图像匹配追踪算法中迭代过程的停止准则（此研究内容可参考第 7 章）。本节试验中用原子个数作为 MP 算法中迭代停止的依据，对同一幅图像用不同的原子个数进行表示，从而得到不同质量和不同压缩比的重建图像。其中，本节使用峰值信噪比（ $P_{\text{SNR}}$ ）来衡量图像质量，其计算公式为：

$$P_{\text{SNR}} = 10 \times \lg \left[ \frac{255 \times 255}{\frac{1}{M \times N} \sum_{j=0}^{M-1} \sum_{i=0}^{N-1} (f_{i,j} - \hat{f}_{i,j})^2} \right] \quad (8.6)$$

式中， $f$  代表压缩前的原始图像， $\hat{f}$  代表通过压缩后重建的图像。

### (1) 量化编码前后重建图像质量的比较

根据图像稀疏分解结果, 直接利用公式 (8.2) 重建的图像如图 8.2 (a) 所示。对分解出的图像数据进行量化和编码, 得到一个压缩后的图像数据文件。通过解码, 从压缩图像文件中近似恢复出原图像, 如图 8.2 (b) 所示。图 (a) 和图 8.2 (b) 在视觉上并没有显著的差别, 在信噪比方面也没有显著差别 (信噪比相差为 0.002 5 dB), 说明我们的量化和编码过程是正确的。



(a) 根据稀疏分解数据直接重建的图像



(b) 根据量化的稀疏分解数据重建的图像  
(压缩比为 43:1)

图 8.2 量化编码前后重建图像质量的比较

### (2) 基于稀疏分解的图像压缩编码方法之间的对比

在图 8.3 中给出了本节基于稀疏分解的图像压缩编码方法和本章参考资料[16]中基于稀疏分解的图像压缩编码方法压缩效果的对比情况。由于本章参考资料[16]中采用的压缩图像为  $128 \times 128$  大小、8 bit 量化的标准 Lena 测试图像, 所以在此项对比实验中采用  $128 \times 128$  大小的 Lena 测试图像。从图中可以看出, 本节的方法在低比特率 (如  $< 0.4$  bpp) 时的实验结果要优于本章参考资料[16]中的结果。但因为本章参考资料[16]采用了把量化误差反馈到正在分解的图像上, 从而在分解的下一步中对此量化误差进行校正, 即此种方法是把分解过程和量化过程合并在一起, 从而达到减小

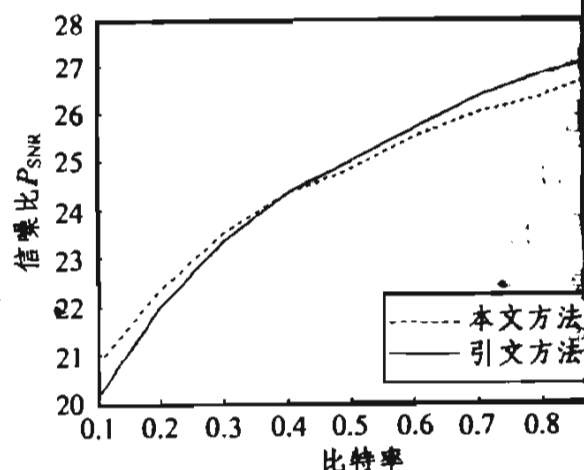


图 8.3 低比特率情况下两种基于稀疏分解的图像压缩方法压缩效果的比较

体量化误差的目的；本节方法没有考虑这种量化误差减小方法，所以压缩编码效果在比特率增加到一定程度时（如 $>0.4$  bpp）要稍差于本章参考资料[16]中方法的效果。本节方法在低比特率（如 $<0.4$  bpp）时的实验结果要优于本章参考资料[16]中的结果，主要是因为本节中采用的分解算法<sup>[33, 34]</sup>要优于本章参考资料[16]中的分解算法。而此时在分解下一步中对量化误差进行校正的优化作用效果并不明显。只有在分解进行到一定程度时，量化误差校正的优势才体现出来，这时本章参考资料[16]中算法的压缩编码效果才优于本文算法的压缩编码效果。

由于我们的结果是一个初步的，尚没有新的 JPEG2000 算法效果好。我们现在正在做进一步的研究，以获得更好的压缩效果。通过分析，我们认为压缩效果不如 JPEG2000 算法的原因在于：① 由于计算条件的限制，实验中用于图像稀疏分解的过完备原子库大小不可能接近于理论大小；② 基于同样原因，稀疏分解算法往往用近似算法来实现，如 MP 算法的每一步理论上是解一个全局最优化问题，但往往用局部最优化算法来近似解决<sup>[33, 34]</sup>；③ 在量化和编码中，本节利用了分解结果数据的分布规律，但对分解结果数据没有进行组合优化，特别是没有利用投影分量之间的相关性。目前正对以上问题进行研究，并已经取得了一定的成果。

## 8.4 基于排序差分 and 稀疏分解的图像压缩编码方法

现有的基于稀疏分解的图像压缩编码方法对分解出的数据直接进行量化编码，因而没有消除分解结果数据之间的冗余，由此得到的压缩结果并不完美。为此，有必要对图像稀疏分解数据的特性进行深入研究。

本节研究基于稀疏分解的图像压缩编码方法，根据对稀疏分解结果数据的分析，提出了新的压缩编码方案。本方案对稀疏分解结果数据进行差分预处理，大幅度缩小了投影分量的分布范围，提高了编码效率。对比实验表明，本方法在低比特率压缩时重建图像有较好的主观质量和客观质量。

### 8.4.1 基于稀疏表示的图像压缩中存在的关键难题

设研究的图像为  $f$ ， $f$  大小为  $M_1 \times M_2$ ， $M_1$  和  $M_2$  为图像的长和宽。设  $\mathcal{D} = \{\mathbf{g}_\gamma\}_{\gamma \in \Gamma}$  为用于图像稀疏分解的过完备库。图像稀疏分解的结果即稀疏表示用式 (8.2) 表示。不失一般性，原子库用非对称原子库，用式 (8.3)、(8.4)

表示。分解结果用式 (8.5) 表示。

图像稀疏分解结果数据的动态范围和分布规律，一方面是由稀疏分解内在规律决定的<sup>[1, 2]</sup>，另一方面是由本节使用的分解方法决定的<sup>[33, 34]</sup>，同时也跟原子库的结构有关。原子库的结构由  $\theta$ 、 $u$ 、 $v$ 、 $s_x$  和  $s_y$  的取值情况决定，本节中使用的  $\theta$ 、 $u$ 、 $v$ 、 $s_x$  和  $s_y$  取值如式 (8.7)。

$$\left. \begin{aligned} u &\in [1, M_1] \\ v &\in [1, M_2] \\ \theta &= \frac{\theta'}{\max(M_1, M_2)} 360, \quad \theta' \in [1, \max(M_1, M_2)] \\ s_x &= 2^{\frac{stx}{NN}}, \quad stx \in [0, NN \log_2^{M_1} - NN] \\ s_y &= 2^{\frac{sty}{NN}}, \quad sty \in [0, NN \log_2^{M_2} - NN], NN = 5 \end{aligned} \right\} \quad (8.7)$$

式 (8.7) 中， $u$ 、 $v$ 、 $\theta'$ 、 $stx$ 、 $sty$  均取整数值。在基于稀疏分解的图像压缩编码中，由于  $u$ 、 $v$ 、 $\theta'$ 、 $stx$ 、 $sty$  均取整数，所以一般无需进行量化，并且也简化了编码过程。所以基于稀疏分解的图像压缩编码中，关键是投影分量  $\langle R^k f, g_n \rangle$  的量化和编码，这也是一个难以解决的问题。

为了对图像稀疏分解的最终结果进行量化和编码，需要分析其分布范围和分布规律。通过分析发现：① 平移量  $u$ 、 $v$  呈均匀分布，其取值范围和图像的大小相同，即：从 1 到图像的长度或宽度。② 旋转角度  $\theta$  呈均匀分布，其取值范围和图像的大小相同，即：从 1 到图像的长度或宽度。③ 投影分量  $\langle R^k f, g_n \rangle$

( $k=0,1,2,\dots,n-1$ ) 的取值范围变化较大，但随着图像分解进程，即随着  $k$  从 0 变化到  $n-1$ ，其取值按指数衰减快速趋于

0。④ 原子的伸缩  $s_x$ 、 $s_y$  的变化规律与  $\langle R^k f, g_n \rangle$  ( $k=0,1,2,\dots,n-1$ ) 的变化规律类似，区别在于  $\langle R^k f, g_n \rangle$  可取负值，而  $s_x$ 、 $s_y$  总是取正值，且是整数。图 8.4 显示了当原始图像大小为  $512 \times 512$ 、8 bit 量化时， $\langle R^k f, g_n \rangle$  ( $k=0,1,2,\dots,n-1$ ) 的分布规律。

从图像稀疏分解结果数据分布规律中可以看出，投影分量的值域远大于其他分量的值域，而且在此范围内取任意

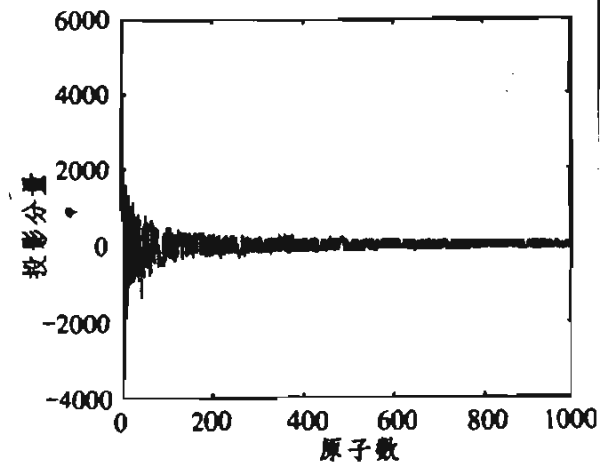


图 8.4 图像稀疏分解结果中投影分量的分布规律

实数（而其他分量只取整数）。相应的，如果要对它较准确地量化并编码，其所占用的比特数也远大于其他分量所占用的比特数。因此，如果能够大幅度提高投影分量的编码效率，则能够相应的提高基于稀疏分解的图像压缩整体编码效率。本节讨论的编码方法也主要针对稀疏分解的投影分量。

#### 8.4.2 图像稀疏分解数据基于差分的编码方案

从图像稀疏分解结果数据的分布规律可以看出，投影分量 $\langle R^k f, g_n \rangle$  ( $k=0,1,2,\dots,n-1$ )的取值范围变化很大，随着分解的进行呈波动递减。如果直接对其进行量化和编码，则会产生大量的编码冗余，故有必要对结果数据进行预处理。本节引入差分方法对分解数据中投影分量进行预处理，以达到提高编码效率的目的。本节使用 MP 算法对图像进行稀疏分解<sup>[1, 2, 33, 34]</sup>，投影参数并没有严格按递减规律分布，而是略有波动。如用其他图像稀疏分解算法，也有类似结果。若直接对其进行差分预处理，差分结果值的动态范围较大，则会产生新的编码冗余。为了解决这个问题，本节先将结果数据按投影分量 $\langle R^k f, g_n \rangle$ 的值由大到小进行排序，使投影分量 $\langle R^k f, g_n \rangle$ 呈单调递减排列，如图 8.5 所示。根据投影分量进行重新排序，则其他分量的顺序也相应发生改变。由于原子的平移分量  $u$ 、 $v$  为均匀分布，重新排序对其分布并无影响，而原子伸缩分量  $s_x$ 、 $s_y$  与投影分量分布类似，又由于其取值范围较投影分量 $\langle R^k f, g_n \rangle$ 取值范围小得多，打乱其分布规律对整体编码效率并无太大影响。至此，对排序后的投影分量进行差分处理，得到其分布规律如图 8.6 所示。

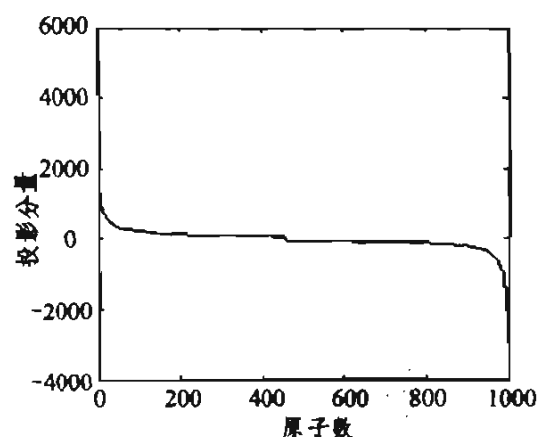


图 8.5 排序后投影分量的分布规律

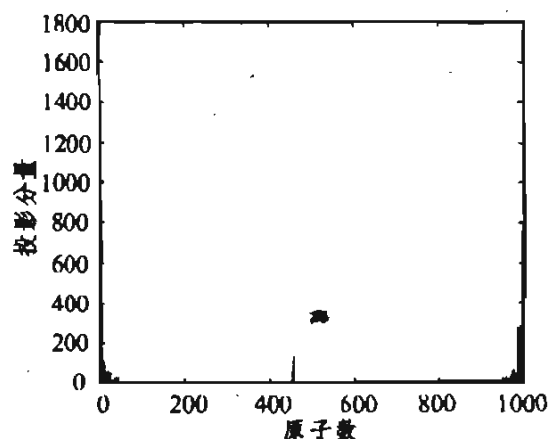


图 8.6 差分后投影分量的分布规律

从图 8.4 和图 8.6 的比较可以看出，经预处理后的投影分量（投影分量的排序差值结果数据）的分布范围较处理之前有大幅度减小，达到了预期的效果。根据排序差分结果，本节设计了量化编码方案（实验图像为  $512 \times 512 \times 8$  的 Lena 图像），如表 8.2 所示。



表 8.2 稀疏分解结果数据的编码方案

原子数	$\langle R^k f, g_n \rangle$	$\theta$	$u$	$v$	$S_x$	$S_y$
1 ~ 20	11	9	9	9	6	6
21 ~ 160	4	9	9	9	5	5
161 ~ 922	2	9	9	9	5	5
923 ~ 978	4	9	9	9	5	5
979 ~ 1 000	11	9	9	9	6	6

注:  $S_x$ 、 $S_y$  的值分别由  $stx$ 、 $sty$  唯一确定。

### 8.4.3 实验结果及讨论

实验中采用  $512 \times 512$  大小、8 bit 量化的标准 Lena 测试图像和 Peper 测试图像。稀疏分解算法采用 MP 算法<sup>[1, 2, 33, 34]</sup>, 原子库采用非对称原子库<sup>[6]</sup>。实验分为三部分: 第一部分是量化和编码的实验; 第二部分是本节方法和最新发表本章参考资料[16]上均为基于稀疏分解的图像压缩编码方法效果的对比实验; 第三部分是在低比特率的情况下不同方法压缩效果的对比实验。

#### (1) 量化编码前后重建图像质量的比较

根据图像稀疏分解结果直接重建图像, 如图 8.7 (a) 所示。对分解出的图像数据进行量化和编码, 得到一个压缩后的图像数据文件; 通过解码, 从压缩图像文件中近似恢复出原图像, 如图 8.7 (b) 所示。图 8.7 (a) 和图 8.7 (b) 在视觉上并没有显著的差别, 在信噪比方面也没有显著差别 ( $P_{\text{SNR}}$  信噪比相差为 0.002 5 dB), 说明量化和编码过程是正确的。



(a) 根据稀疏分解数据直接重建的图像



(b) 根据量化编码数据重建的图像  
(压缩比为 54:1)

图 8.7 量化编码前后重建图像质量的比较

## (2) 基于稀疏分解的图像压缩编码方法之间的对比

在图 8.8 中给出了本节基于稀疏分解的图像压缩编码方法和本章参考资料[16]中基于稀疏分解的图像压缩编码方法压缩效果的对比情况。由于本章参考资料[16]中采用的压缩图像为  $128 \times 128$  大小、8 bit 量化的标准 Lena 测试图像，所以在此项对比实验中均采用  $128 \times 128$  大小的 Lena 测试图像。从图 8.8 可以看出，本节方法的实验结果要优于本章参考资料[16]中的结果。

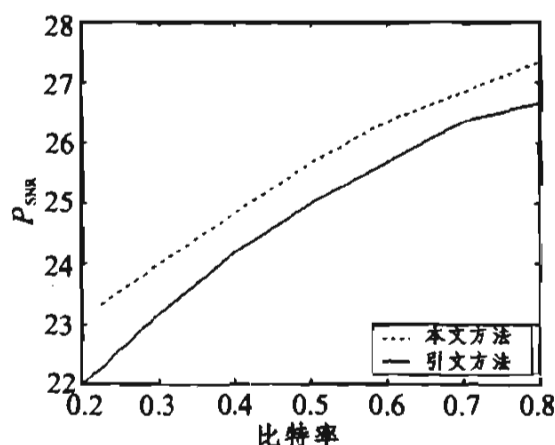


图 8.8 低比特率情况下两种基于稀疏分界的图像压缩方法压缩效果的比较

## (3) 本节方法和其他压缩编码方法比较

### 比较之一：重建图像客观质量比较

客观质量比较采用信噪比 ( $P_{SNR}$ ) 为评价标准。表 8.3 给出了多种压缩方法在低比特率情况下的表现。注意：实验中此部分采用  $512 \times 512$  大小、8 bit 量化的标准 Lena 测试图像和 Peper 图像，结果有可能和本书第 7 章结果不一致，这是由于实验中采用的图像大小不一致造成的。

从表 8.3 可以看出，基于稀疏分解的图像压缩方法在图像低比特率压缩时，其重建图像的客观质量要优于现有的多种图像压缩方法的重建图像客观质量。

表 8.3 本节实验结果与多种图像压缩方法的比较

Lena	本文方法	自适应零树小波编码	本文方法	IWT+EWZ	边缘自适应小波变换	JPEG2000	双正交多小波
比特率/bpp	0.062 5	0.062 5	0.123	0.125	0.125	0.125	0.125
$P_{SNR}/dB$	27.32	26.37	29.15	27.29	25.39	25.61	29.13
文献		[35]		[36]	[37]	[37]	[38]
Peper	本文方法	浮点 DCT	binDCT	本文方法	浮点 DCT	binDCT	
比特率/bpp	0.062 5	0.062 5	0.062 5	0.122	0.125	0.125	
$P_{SNR}/dB$	26.247 2	25.6	25.31	28.238 4	28.45	28.3	
文献		[39]	[39]		[39]	[39]	

### 比较之二：重建图像主观质量比较

由于图像的最终接受者往往是人而不是计算机或其他设备，所以，和图像的客观质量评价相比较，人的主观图像质量评价显得更加重要。在图 8.9 中分别显示了恢复图像具有相同信噪比 ( $P_{\text{SNR}}$ ) 时的解码图像。其中，图 8.9 (a) 为 JPEG2000 重建图像，图 8.9 (b) 为使用本节方法压缩图像的重建结果。从主观上比较，图 8.9 (b) 较图 8.9 (a) 有更好的主观视觉效果。JPEG2000 算法恢复图像的边缘和图像细节有较明显的失真，让人感到不舒服。基于稀疏分解压缩后的重建图像虽然也有失真，但人的视觉系统对其失真不太敏感，从而获得更好的视觉效果。所以，基于稀疏分解的图像压缩，在大的压缩比时，恢复图像有较好的主观质量。



(a) JPEG2000 重建图像



(b) 本节方法重建图像

图 8.9 低比特率情况下客观质量相同时不同压缩方法重建图像主观质量比较

从以上实验结果可以看出，在低比特率的情况下，基于稀疏分解的图像压缩可以得到较好的压缩效果，为低比特率图像压缩提供了一种新的可能解决方案。

## 8.5 结 论

对图像稀疏分解已经进行了较深入研究，而基于图像稀疏分解的压缩编码已经报道的结果并不太令人满意。我们在图像稀疏分解的基础上，通过对图像稀疏分解结果数据进行差分预处理，有效地提高了编码效率。实验证明，本章所提出的图像压缩方法在图像低比特率压缩时，其重建图像有较好的客观质量和主观质量。低比率图像压缩的研究成果，将有力地推动窄带图像通信的发展。

## 参 考 资 料

- [ 1 ] Mallat S, Zhang Z. Matching pursuits with time-frequency dictionaries[J]. IEEE Trans. Signal Processing, 1993, 41: 3397~3415
- [ 2 ] Bergeau F, Mallat S. Matching pursuit of images [A]. Proceedings of IEEE-SP[C]. USA: Piladephia, 1994, 330~333
- [ 3 ] Coifman R, Wickerhauser M. Entropy-based algorithms for best basis selection[J]. IEEE Trans. Information Theory, 1992, 38: 1713~1716
- [ 4 ] Olshausen B, Field D. Emergence of simple-cell receptive field properties by learning a sparse code for natural images[J]. Nature, 1996, 381: 607~609
- [ 5 ] Olshausen B, Field D. Learning efficient linear codes for natural images: the roles of sparseness, overcompleteness, and statistical independence[A]. Proceedings of SPIE [C]. 1996, 2657: 132~138
- [ 6 ] Vanderghelynst P, Frossard P. Efficient image representation by anistropic refinement in matching pursuit[A], Proceedings of IEEE on ICASSP[C]. Salt Lake City, UT, USA, May, 2001, 3: 1757~1760
- [ 7 ] Cand E. Ridgelet: Theory and applications [D]. USA: Department of Statistics, Stanford University, 1998
- [ 8 ] Cand E, Donoho D. Curvelets[R]. USA: Department of Statistics, Stanford University, 1999
- [ 9 ] Pennec E, Mallat S. Image compression with geometrical wavelets[A]. Proceeding of IEEE ICIP[C]. Vancouver, Canada, 2000. 661~664
- [10] Do M, Vetterli M. Contourlet[A]. J Stoeckler, GV Welland. Beyond Wavelets[C]. Academic Press. 2002
- [11] Gribonval R, Nielsen M. Sparse decompositions in 'incoherent' dictionaries[A], Proceedings of IEEE ICIP[C], 2003: 33~36
- [12] Chen S, Donoho D, Sauners M. Atomic decomposition by basis pursuit[J]. SIAM REVIEW, 2001, 43 (1): 129~159
- [13] Daubechies I. Time-frequency localization operator: A geometric phase space approach[J]. IEEE Trans. Information Theory, 1988, 34: 605~612
- [14] Y. C. Pati, R. Rezaiifar, P. S. Krishnaprasad. Orthogonal matching pursuits: recursive function approximation with applications to wavelets decomposition[A]. Proceedings of the 27<sup>th</sup> Asilomar Conference in Signals, Systems and Computers[C]. 1993, 40~44

- [15] Thang V. Pham, Arnold W. M. Smeulders. Sparse representation for coarse and fine object recognition[J]. IEEE Transaction on pattern analysis and machine intelligence. 2006, 28 (4): 555~587
- [16] Frossard P, Vandergheynst P, Rosa Maria Figueras i Ventura, Kunt M, A Posteriori Quantization of Progressive Matching Pursuit Streams[J]. IEEE Trans. on Signal Processing, 2004, 52 (2): 525~535
- [17] Neff R, Zakhor A. Very low bit-rate video coding based on matching pursuit[J]. IEEE Trans. Circuits and Systems for Video Tech., 1997, 7 (1): 158~171
- [18] Rosa M. R. I. V., Pierre Vandergheynst, Pascal Frossard, Low-rate and flexible image coding with redundant representations[J]. IEEE trans. On Image Processing, 2006, 15 (3): 726~741
- [19] 邹红星, 周小波, 李衍达. 时频分析: 回溯与前瞻[J]. 电子学报, 2000, 28 (9): 78~84
- [20] SESSION. MA-S2 : Redundant Representations for Visual Communications. Proceedings of IEEE ICIP[C], 2003, Barcelona, Spain, Sept. 2003, 3: 33~64
- [21] 焦李成, 谭山. 图像的多尺度几何分析: 回顾和展望[J]. 电子学报, 2003, 31(12A): 1975~1981
- [22] 孙俊, 王文渊, 卓晴. 基于稀疏编码的提取人脸整体特征算法[J]. 清华大学学报(自然科学版), 2002, 42 (3): 411~413
- [23] 刘利雄, 贾云得, 廖斌, 张敏. 一种改进的最佳时频原子搜索策略[J]. 中国图像图形学报, 2004, 9 (7): 873~877
- [24] 廖斌, 许刚, 王裕国. 一种基于混合变换的分层图像表示与编码[J]. 通信学报, 2004, 25 (6): 120~125
- [25] 傅霆, 尧德中. 稀疏分解的加权迭代方法及其初步应用[J]. 电子学报, 2004, 32 (4): 567~570
- [26] 杜小勇, 胡卫东, 郁文贤. 高分辨雷达一维距离像稀疏表示技术[J]. 系统工程与电子技术, 2005, 27 (6): 968~970
- [27] 杜小勇, 胡卫东, 郁文贤. 推广的正则化 FOCUSS 算法及收敛性分析[J]. 系统工程与电子技术, 2005, 27 (5): 924~925
- [28] Zhang Chunmei, Yin Zhongke, Chen Xiangdong, Xiao Mingxia. Signal overcomplete representation and sparse decomposition based on redundant dictionaries[J]. Chinese Science Bulletin, 2005, 50 (23): 2672~2677
- [29] Yin Zhongke, Xie Mei, Wang Jiangying. Fast MP of images based on atom energy property[J]. Journal of UESTC, 2005, 34 (2): 156~159

- [30] 张春梅, 尹忠科, 肖明霞. 基于冗余字典的信号超完备表示与稀疏分解[J]. 科学通报, 2006, (7)
- [31] 尹忠科, 邵君, Pierre Vandergheynst. 利用 FFT 实现基于 MP 的信号稀疏分解[J]. 电子与信息学报. 2006, (4)
- [32] 张跃飞, 姜玉亭, 王建英, 尹忠科. 基于稀疏分解的图像压缩[J]. 系统工程与电子技术, 2006
- [33] 尹忠科, 王建英, Vandergheynst P. 在低维空间实现的基于 MP 的图像稀疏分解[J]. 电讯技术, 2004, 44 (3): 12~15
- [34] 尹忠科, 王建英, Vandergheynst P. 一种新的图像稀疏分解快速算法[J]. 计算机应用, 2004, 24 (10): 92~96
- [35] 彭进业, 俞卞章, 张烨, 刘宏兵. 一种改进的图像自适应零树编码方法[J]. 电子学报, 2002, 30 (4): 591~593
- [36] 丁绪星, 朱日宏, 李建欣. 基于整数小波变换和改进嵌入零树编码的图像压缩[J]. 电子与信息学报, 2004, 26 (7): 1064~1069
- [37] 汪雪林, 刘忠轩, 彭思龙. 基于边缘自适应小波变换的低比特率图像压缩算法[J]. 中国图像图形学报, 2005, 10 (2): 150~157
- [38] 高西奇, 甘露, 邹采荣. 对称—反对称多小波滤波器组的参数化构造及其在图像压缩中的应用[J]. 电子学报, 2001, 29 (6): 796~798
- [39] 钟广军, 成礼智, 陈火旺. 双正交重叠变换的整数实现算法与图像压缩[J]. 电子学报, 2001, 29 (11): 1499~1502

## 致 谢

近年来,三位作者从事信号与图像的稀疏分解及初步应用方面的研究,得到了四川省重点科技攻关计划项目(批准号:04GG006-025,04GG021-020-05)、四川省应用基础研究项目(批准号:04JY029-059-2,03JY029-048-2)、教育部留学回国人员科研启动基金项目(批准号:[2004]527)、国家民族事务管理委员会 2004 年度重点项目的资助,在此表示衷心感谢!另外,作者还得到了西南交通大学校基金的资助,使得在其他项目资金没有落实的情况下,即可着手开展工作,在此特别表示感谢!

在从事此项研究工作和编写本书的过程中,硕士研究生参与了部分工作,他们是:邵君、尹明、吕雪、陈磊、袁龙、张悦庭、朱宏、于洋、孟晓锋、杨美英、王峰等同学。另外,博士研究生李恒建、张跃飞也参与了部分工作。本书能够顺利出版,西南交通大学出版社做出了很大的贡献。在从事此项工作的过程中,作者与电子科技大学尧德中教授、清华大学邹红星教授进行了深入讨论,并得到了两位非常有益的帮助。在此向以上各位表示衷心感谢!

在作者从事研究的过程中,得到了西南交通大学科研处、西南交通大学信息科学与技术学院各位领导的大力支持,使此项工作能够成功开展,在此表示衷心感谢!

由于本书内容的需要,在本书的第一部分,主要参考了电子科技大学出版社出版的向敬成老师、刘醒凡老师的书《信号理论》的内容,在此向两位老师表示深深的谢意。

由于作者水平有限,时间仓促,书中错误在所难免,请读者多提宝贵意见,我们将衷心感谢!

作 者

2006 年 5 月

## 附录 A 基于 MP 的信号稀疏分解参考程序

```
% function: MP algorithm applied to decomposition of one signal named bat
% This program is for explaining the theory of signal sparse decomposition by
% MP, to help the readers to understand the concept and the process of MP
% method.
% the over-complete dictionary used in the program is the Gabor atom ditionary

% ATTENTION: it's maybe not possible to construct the dictionary at one time and to use %it
forever.
% because the memory is so limited compared with the memory needed.


% part one: read signal data (to be processed )
% Here we can input different types of data, to check the program.
% In this example, we read one signal data from disk, which is saved in the
% format: save bat.dat bat -ascii

load bat.dat -ascii;

% part 2:

% part 2.1

%determine the decomposition parameters

% the matching pursuit processing iterative number

iterative_number=10;
```



% the length of signal and the length of atoms; N

```
[a,N]=size(bat);
```

%signal\_reconstruct is the signal reconstructed by the sparse parameters

%signal\_r is the residual signal , that is the left signal of the original

%signal reduced by the signal\_reconstruct

```
signal_reconstruct=zeros(1,N);
```

```
signal_r=bat;
```

% the following is the Gabor atom parameters which are closely related to the size of the %  
over-complete atom dictionary.

% scale: the scale is determined by a and j;

% I think that the j should include the 0

```
a_base=2;
```

```
j_min=0;
```

```
j_max=log2(N);
```

% the transmission or displacement u should be determined as following

```
u_base=1/2;
```

```
p_min=0;
```

% p\_max can be determined by its relationship with j

% the frequency v

```
v_base=pi;
```

```
k_min=0;
```

% k\_max can be determined by its relationship with j;

% the phase w

w\_base=pi/6;

i\_min=0;

i\_max=12;

%part 2.2

% the matching pursuit process

for n=1:iterative\_number

% the following program uses one subroutine to select the best atom;

[proj,scale,translation,freq,phase]=select\_best(signal\_r,N,a\_base,j\_min,j\_max,u\_base,  
p\_min,v\_base,k\_min,w\_base,i\_min,i\_max);

%reconstruct the best atom from the parameters gotted by the above subroutine

t=0:N-1;

t=(t-translation)/scale;

g=(1/sqrt(scale))\*exp(-pi\*t.\*t).\*cos(freq\*t+phase);

g=g/sqrt(sum(g.\*g));

% reducing the best atom part from the residual signal and adding it to the reconstructed  
%signal

signal\_reconstruct=signal\_reconstruct+proj\*g;

signal\_r=signal\_r-proj\*g;

```
% at each step of MP, we display the figures of the original signal, the best atom
% selected,the residual signal and the reconstructed signal at window 1, 2
% ,3, 4, respectively
```

```
subplot(221);
plot(bat);
subplot(222);
plot(g);
subplot(223);
plot(signal_r);
subplot(224);
plot(signal_reconstruct);
drawnow;
```

```
% n is the MP process number, or the number of atoms selected
n
end
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
```

```
% the above program is the main program for signal decomposition by mp
%
```

```
% the following program is the subroutine program (function program),
% select_best
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
Function [proj,scale,translation,freq,phase]=select_best(signal_r,N,a_base, j_min, j_max,
u_base, p_min,v_base,k_min,w_base,i_min,i_max);
```

```
% this subroutine is to select in the dictionary the best atom suited the signal or the residual
of the signal
```

```
% INPUT
```

```

% the signal_r: the signal or the residual of the signal to be decomposed
% the N: the length of the signal or of the residual of the signal or the length of the atoms

% parameters : the parameter to construct the dictionary , it has much influence on the
% speed of the decomposition
% a_base=2
% j_min=0;
% j_max=log2(N);

%OUTPUT
% proj: the projection of the signal or the residual of the signal on the best atom
% the scale: the scale of the best atom (s in the formula)
% the translation : the translation of the best atom (u in the formula)
% the freq: the frequency of the best atom (v in the formula)
% phase: the phase of the best atom (w in the formula)

% proj_trans : to determine which projection is biggest

proj_trans=0;
proj=0;

% size_dic is one parameter to show the size of the over-complete dictionary used

size_dic=0

for j=j_min:j_max
    for p=p_min:N*2^(-j+1)
        for k=k_min:2^(j+1)
            for i=i_min:i_max

                size_dic=size_dic+1;
                s=a_base^j;
                u=p*s*u_base;
                v=k*(1/s)*v_base;

```

```

w=i*w_base;
t=0:N-1;
t=(t-u)/s;

g=(1/sqrt(s))*exp(-pi*t.*t).*cos(v*t+w);

g=g/sqrt(sum(g.*g));

proj_trans=sum(signal_r.*g);

if abs(proj_trans)>abs(proj)
    proj=proj_trans;
    scale=s;
    translation=u;
    freq=v;
    phase=w;
end
end
end
end
end

% size_dic :    display the size of the dictionary
size_dic

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% end of appendix A %%%%%%%%%%

```

## 附录 B 基于 MP 的信号参数估计参考程序

```
% MATLAB script for estimation the parameter of the cosinoidal signal by Matching
% Pursuit (MP)
%
%
%***** Preparation part *****
%
% *****set parameter*****
f=9.251; % frequency of the signal
fai=0.512*pi; % phase of the signal
snr=-20; % SNR in dB (logarithmic)
t=0:0.001:32; % time vector
s=cos(2*pi*f*t+fai); % signal with the Amplify of 1
%
%***** noise generation *****
%
tr=randn(size(s));
Amplifytr=max(abs(tr));
ampsig=Amplifytr*10^(snr/20);
s=ampsig*s;
%
%***** signal with noise generation *****
%
s_o=s+tr;
%
%***** dictionary generation *****
%
error=zeros(101,201);
```

```

for fatom=0:1000                                % frequency of the atom
    for faiatoml=0:1:200
        faiatom=faiatoml*0.01*pi;                % phase of the atom
        fatoml=fatom*0.01;
        s1=cos(2*pi*fatoml*t+faiatom); % generate the atom
        s_s1=sqrt(sum(s1.*s1));
        s1=s1/s_s1;                               % normalizes to 1 the norm of atom
    % **calculate the decomposition coefficients of all the atoms and the signal**
        error(fatom+1,faiatoml+1)=sum(s_o.*s1);
    end
end

yabs=abs(error);                                % save all the expansion coefficients
[m,n]=max(yabs);                                % choose the of beat atom
[k,l]=max(m);
result_f=(n(l)-1)/100;                          % save the frequency parameter of beat atom
result_fai=(l-1)*0.01;
result_a=error(n(l),l);

if result_a<0
    result_a=-result_a;
    result_fai=result_fai-1;
end

result_a=result_a/s_s1                          % save the amplify parameter of beat atom
result_fai;                                     % save the phase parameter of beat atom
%
%***** renew the cosinoidal signal *****
%
sr=result_a*cos(2*pi*result_f*t+result_fai*pi);
%
%***** the plots of the results *****
%
tt=0:1:1000;
subplot(2,2,1);

```

```

plot(tt,s(1:1001));          % original siganl
subplot(2,2,2);
plot(tt,s_o(1:1001));       % signal with noise
subplot(2,2,3);
plot(tt,sr(1:1001));        % renewed signal
subplot(2,2,4);
plot(yabs(:,1-1));          % decomposed coefficients of frequency

% ***** end of file *****

```



## 附录 C 基于 GA 和 MP 的信号稀疏分解参考程序

```
%% MATLAB script for MP algorithm based on Genetic Algorithm (GA)
%
%
%***** Preparation part *****
%

%***** part 1:input signal *****
%read signal data (to be processed )
% Here we can input different types of data, to check the program.
% In this example, we read one signal data from disk, which is saved in the
% format: save bat.dat bat -ascii

load bat.dat -ascii;
Im=bat;

%***** part 2: set parameters *****
f1=50;
f2=160;
fs=200;

iterative_number=100; % the matching pursuit processing iterative number
[a,N]=size(Im); % the length of signal and the length of atoms; N
a_base=2; % scale: the scale is determined by a and j;
j_min=0;
j_max=log2(N);
u_base=1/2; % the transmission or displacement u should determined as following
v_base=pi; % the frequency v
k_min=0;
```

```

w_base=pi/6; % the phase w
i_min=0;
i_max=12;

```

```

%%***** initialization *****

```

```

signal=Im;
signal_reconstruct=zeros(1,N);
signal_r=Im;

```

```

%***** wipe off the direct current vector *****

```

```

signal_reconstruct=(1/N)*sum(signal);
signal_r=signal-signal_reconstruct;

```

```

%%*****start calculation*****

```

```

for n=1:iterative_number

```

```

% *****the following program uses one subroutine to select the best atom*****
%

```

```

[proj,scale,translation,freq,phase]=gas(signal_r,N,a_base,j_min,j_max,u_base,p_min,
v_base ,k_min,w_base,i_min,i_max);

```

```

%*****reconstruct the best atom from the parameters got by above subroutine*****

```

```

t=0:N-1;
t=(t-translation)/scale;
g=(1/sqrt(scale))*exp(-pi*t.*t).*cos(freq*t+phase);
g=g./sqrt(sum(g.*g)); %normalization the atom

```

```

%*****reconstruct signal and renew the residual*****

```

```

signal_reconstruct=signal_reconstruct+proj*g;
signal_r=signal_r-proj*g;

%***** the figure of result *****

subplot(221);
plot(signal);          % original signal
subplot(222);
plot(g);              %the atom
subplot(223);
plot(signal_r);       %the residual
subplot(224);
plot(signal_reconstruct); %reconstruct signal
end

% ***** end of main program file *****

% ***** The following is the subroutine program searching for the best atom. *****

function [proj,scale,translation,freq,phase]=gas(signal_r,N,a_base,j_min, j_max,u_base,
p_min,v_base,k_min,w_base,i_min,i_max)

% this subroutine is to select in the dictionary the best atom suited the signal or the residual
of the signal

% the searching method used is Genetic Algorithm (GA).
% the following parameters are used in GA
% n : number of Chromosomes
% Ngen: the number of generation
% vec: the solution for one atom
% v(1) : the scale
% v(2) : location or translation
% v(3) : frequency

```

```

% v(4) : phase
% Genetic algorithm : INITIALIZATION
% proj_trans :to determine which projection is biggest

%***** set parrameter*****
n=21;
Ngen=30;

pool=zeros(4,n);

sig=[1
     1
     1
     1];
proj_trans=zeros(1,n);
res=ones(1,n);
proj=0;
bi=ones(4,1); % bi: lower bounds
bs=ones(4,1); % bs: upper bounds

bi(1)=j_min;
bi(2)=p_min;
bi(3)=k_min;
bi(4)=i_min;

bs(1)=j_max;
bs(2)=N;
bs(3)=N;
bs(4)=i_max;

og=ones(4,n); % og(i,:) : vector of n i-genes
ng=ones(4,n); % ng: new generation vector

%***** create the initial population*****

```

```

og(1,:)=round((bs(1)-bi(1))*rand(1,n)+bi(1));
og(2,:)=round((bs(2)-bi(2))*rand(1,n)+bi(2));
og(3,:)=round((bs(3)-bi(3))*rand(1,n)+bi(3));
og(4,:)=round((bs(4)-bi(4))*rand(1,n)+bi(4));

```

**%\*\*\*\*\*the main part of Genetic algorithm \*\*\*\*\***

```

for c=1:Ngen
    for d=1:n
        s=a_base^og(1,d);
        u=og(2,d);
        v=og(3,d)*(1/s)*v_base;
        w=og(4,d)*w_base;
        t=0:N-1;
        t=(t-u)/s;
        g=(1/sqrt(s))*exp(-pi*t.*t).*cos(v*t+w);
        g=g/sqrt(sum(g.*g));
        proj_trans(d)=sum(signal_r.*g);
        res(d)=abs(proj_trans(d));
    end
end

```

**%\*\*\*\*\* place the best atom (biggest projection in the first position) \*\*\*\*\***

```

[best, e]=max(res);
ng(:,1)=og(:,e);
og(:,e)=og(:,n);
og(:,n)=ng(:,1);
temp_proj=proj_trans(e);
temp_res=res(e);
proj_trans(e)=proj_trans(n);
proj_trans(n)=temp_proj;

res(e)=res(n);
res(n)=temp_res;

```

```
%*****competition between the adjacent atoms*****
```

```
for d=2:2:n-1
    [best,e]=max(res(d-1:d));
    pool(:,d/2)=og(:,d-rem(e,2));
end;
```

```
%*****creation of the pool for crossover *****
```

```
for d=1:(n-1)/2
    f=ceil((n-1)/2*rand);
    Inter=round(rand(4,1));
    ng(:,d+1)=pool(:,d).*Inter+pool(:,f).*(1-Inter);
end
```

```
%***** Mutations of the winner *****
```

```
sigm=sig(:,ones(n-((n-1)/2+1),1));
ngm=ng(:,1);
ngm=ngm(:,ones(n-((n-1)/2+1),1));
ng(:,(n-1)/2+2:n)=round(ngm+(rand(4,n-((n-1)/2+1))-0.5).*sigm);
```

```
%***** bounding into the limits *****
```

```
for l=1:4
    while max(ng(l,:))>bs(l)
        [rw,lw]=max(ng(l,:));
        ng(l,lw)=bs(l);
    end
    while min(ng(l,:))<bi(l)
        [rw,lw]=min(ng(l,:));
        ng(l,lw)=bi(l);
    end
end
end
```

```

og=ng;

%***** stable or not? *****:

nog=og(:,1);
nog=nog(:,ones(1,n));
nog=abs(og-nog);
nog=max(max(nog));

%***** if yes, create a new generation *****:

if nog<4
    og(1,2:n)=round((bs(1)-bi(1))*rand(1,n-1)+bi(1));
    og(2,2:n)=round((bs(2)-bi(2))*rand(1,n-1)+bi(2));
    og(3,2:n)=round((bs(3)-bi(3))*rand(1,n-1)+bi(3));
    og(4,2:n)=round((bs(4)-bi(4))*rand(1,n-1)+bi(4));
end
end

%***** output the results*****

s=a_base^ng(1,1);
u=ng(2,1);
v=ng(3,1)*(1/s)*v_base;
w=ng(4,1)*w_base;
t=0:N-1;
t=(t-u)/s;
g=(1/sqrt(s))*exp(-pi*t.*t).*cos(v*t+w);
g=g/sqrt(sum(g.*g));

proj=sum(signal_r.*g);
scale=a_base^ng(1,1);
translation=ng(2,1);
freq=v;

```

## 附录 D 利用 FFT 实现基于 MP 的信号稀疏分解参考程序

```
%% MATLAB script for MP algorithm based on FFT
%
%
%***** Preparation part *****
%
%***** part 1:input signal *****
%
%read signal data (to be processed )
% Here we can input different types of data, to check the program.
% In this example, we read one signal data from disk, which is saved in the
% format: save bat.dat bat -ascii

load bat.dat -ascii;
signal=bat;

%***** part 2: set parameters *****

iterative_number=30; % the matching pursuit processing iterative number
[a,N]=size(signal); % the length of signal and the length of atoms; N

signal_reconstruct=zeros(1,N);
signal_r=signal;

a_base=2; % scale: the scale is determined by a and j;
j_min=0;
j_max=log2(N);
u_base=1/2; % the transmission or displacement u should determined as following
```



```

v_base=pi;    % the frequency v
k_min=0;
w_base=pi/6;  % the phase w
i_min=0;
i_max=12;

%***** wipe off the direct current vector *****

signal_reconstruct=(1/N)*sum(signal);
signal_r=signal-signal_reconstruct;

%***** create dictionary *****

[atoms]=dic_a(signal_r,N,a_base,j_min,j_max,u_base,v_base,k_min,w_base,i_min,i_max)

%***** start calculation*****

for n=1:iterative_number
    n

% *****the following program uses one subroutine to select the best atom*****

[proj,scale,translation,freq,phase]=select_best(signal_r,N,a_base,j_min,j_max,u_base,
        v_base,k_min,w_base,i_min,i_max,atoms);

%*****reconstruct the best atom from the parameters gotted by above subroutine*****

    t=0:N-1;
    t=(t-translation)/scale;
    g1=(1/sqrt(scale))*exp(-pi*t.*t).*cos(freq*t+phase);
    g=g1/sqrt(sum(g1.*g1));

%*****reconstruct signal and renew the residual*****

```

```

signal_reconstruct=signal_reconstruct+proj*g;
signal_r=signal_r-proj*g;

%***** the figure of result *****

subplot(221);
plot(signal);          % original signal
subplot(222);
plot(g);              %the atom
subplot(223);
plot(signal_r);       %the residual
subplot(224);
plot(signal_reconstruct); %reconstruct signal
end

% ***** end of main program file *****

% ***** the following is the atom dictionary creation program*****

function [atoms]=dic_a(signal_r,N,a_base,j_min,j_max,u_base,v_base,k_min,w_base, i_min,
                        i_max)

% this subroutine is to create the dictionary
% INPUT
% the signal_r: the signal or the residual of the signal to be decomposed
% the N: the length of the signal or of the residual of the signal or the length of the atoms

% parameters :the parameter to construct the dictionary , it has much influence on the
%speed of the decomposition

%OUTPUT
% atoms:the dictionary

```

```

size_dic=1;
atoms=zeros(size_dic,N);

%*****creat dictionary*****
for j=j_min:j_max
    for k=k_min:2^(j+1)
        for i=i_min:i_max
            size_dic=size_dic+1; %the munber of atoms
            s=a_base^j;          %scale of the atom
            u=N/2;                % transmission of the atom
            v=k*(1/s)*v_base;    %frequency of the atom
            w=i*w_base;          %phase of the atom
            t=0:N-1;
            t=(t-u)/s;
            g=(1/sqrt(s))*exp(-pi*t.*t).*cos(v*t+w); %generate the atom
            g=g/sqrt(sum(g.*g)); %normalization
            atoms(size_dic,:)=g;
        end
    end
end

%***** end of the dictionary creation subroutine program *****

% *****the following is the best atom selecting subroutine program using FFT*****

Function [proj,scale,translation,freq,phase]=select_best(signal_r,N,a_base,j_min,j_max,u_base, v_base,k_min,w_base,i_min,i_max,atoms)

% this subroutine is to select in the dictionary the best atom suited the signal or the residual of the signal using FFT to reduce computational time

% INPUT
% the signal_r: the signal or the residual of the signal to be decomposed

```

```
% the N: the length of the signal or of the residual of the signal or the length of the atoms
% parameters :the parameter to construct the dictionary , it has much influence on the
% speed of the decomposition
```

```
%OUTPUT
```

```
% proj: the projection of the signal or the residual of the signal on the best atom
% the scale: the scale of the best atom (s in the formula)
% the translation : the translation of the best atom (u in the formula)
% the freq: the frequency of the best atom (v in the formula)
% phase: the phase of the best atom (w in the formula)
```

```
% proj_trans :to determine which projection is biggest
```

```
%*****initialization*****
```

```
[m n]=size(atoms);
re=zeros(5,m);
proj_trans=0;
proj=0;
size_dic=0;
```

```
%*****selcet the best atom*****
```

```
for j=j_min:j_max
    for k=k_min:2^(j+1)
        for i=i_min:12
            size_dic=size_dic+1;
            g=[atoms(size_dic,N:-1:1)];
            co=real(ifft(fft(signal_r,2*N).*fft(g,2*N)));
            recon=co((N/2+1):3*N/2);
            [proj_trans e]=max(abs(recon));
            proj_trans=recon(e);           %choose the biggest projection
            re(:,size_dic)=proj_trans
        end
    end
end
```

```

                                e
                                k
                                i];

                                end

                                end

                                end

                                se=re(1,:);
                                [proj_trans e]=max(abs(se));
                                proj=se(e);
                                scale=re(2,e);
                                translation=re(3,e);
                                freq=re(4,e);

                                %*****compute the parameter of the selected atom*****

                                phase=re(5,e);
                                scale=a_base^scale;
                                freq=freq*(1/scale)*v_base;
                                phase=phase*w_base;

```