

理解离散傅立叶变换（一）

-----傅立叶变换的由来

关于傅立叶变换，无论是书本还是在网上可以很容易找到关于傅立叶变换的描述，但是大都是些故弄玄虚的文章，太过抽象，尽是一些让人看了就望而生畏的公式的罗列，让人很难能够从感性上得到理解，最近，我偶尔从网上看到一个关于数字信号处理的电子书籍，是一个叫 Steven W. Smith, Ph.D. 外国人写的，写得非常浅显，里面有七章由浅入深地专门讲述关于离散信号的傅立叶变换，虽然是英文文档，我还是硬着头皮看完了有关傅立叶变换的有关内容，看了有茅塞顿开的感觉，在此把我从中得到的理解拿出来跟大家分享，希望很多被傅立叶变换迷惑的朋友能够得到一点启发，这电子书籍是免费的，有兴趣的朋友也可以从网上下载下来看一下，URL 地址是：

<http://www.dspguide.com/pdfbook.htm>

要理解傅立叶变换，确实需要一定的耐心，别一下子想着傅立叶变换是怎么变换的，当然，也需要定的高等数学基础，最基本的是级数变换，其中傅立叶级数变换是傅立叶变换的基础公式。

一、傅立叶变换的提出

让我们先看看为什么会有傅立叶变换？傅立叶是一位法国数学家和物理学家的名字，英语原名是 Jean Baptiste Joseph Fourier(1768-1830), Fourier 对热传递很感兴趣，于 1807 年在法国科学学会上发表了一篇文章，运用正弦曲线来描述温度分布，论文里有个在当时具有争议性的决断：任何连续周期信号可以由一组适当的正弦曲线组合而成。当时审查这个论文的人，其中有两位是历史上著名的数学家拉格朗日(Joseph Louis Lagrange, 1736-1813)和拉普拉斯(Pierre Simon de Laplace, 1749-1827)，当拉普拉斯和其它审查者投票通过并要发表这个论文时，拉格朗日坚决反对，在近 50 年的时间里，拉格朗日坚持认为傅立叶的方法无法表示带有棱角的信号，如在方波中出现非连续变化斜率。法国科学学会屈服于拉格朗日的威望，拒绝了傅立叶的工作，幸运的是，傅立叶还有其它事情可忙，他参加了政治运动，随拿破仑远征埃及，法国大革命后因会被推上断头台而一直在逃避。直到拉格朗日死后 15 年这个论文才被发表出来。

谁是对的呢？拉格朗日是对的：正弦曲线无法组合成一个带有棱角的信号。但是，我们可以用正弦曲线来非常逼近地表示它，逼近到两种表示方法不存在能量差别，基于此，傅立叶是对的。

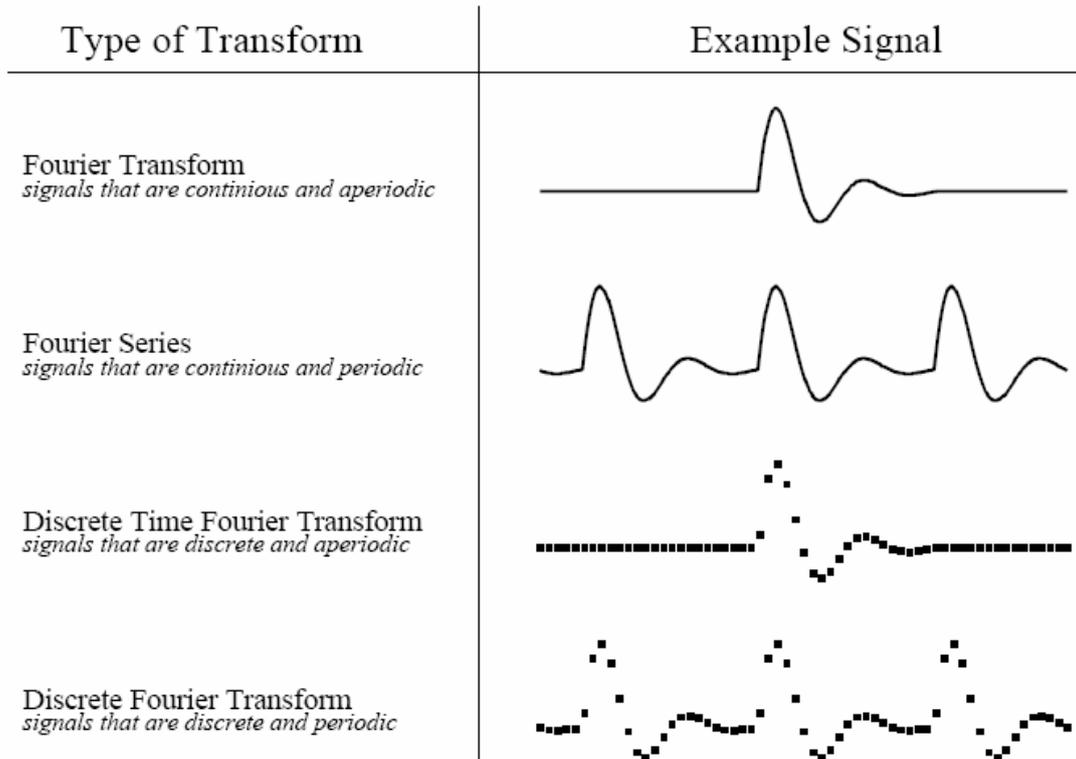
为什么我们要用正弦曲线来代替原来的曲线呢？如我们也还可以用方波或三角波来代替呀，分解信号的方法是无穷的，但分解信号的目的是为了更加简单地处理原来的信号。用正余弦来表示原信号会更加简单，因为正余弦拥有原信号所不具有的性质：正弦曲线保真度。一个正弦曲线信号输入后，输出的仍是正弦曲线，只有幅度和相位可能发生变化，但是频率和波的形状仍是一样的。且只有正弦曲线才拥有这样的性质，正因如此我们才不用方波或三角波来表示。

二、傅立叶变换分类

根据原信号的不同类型，我们可以把傅立叶变换分为四种类别：

1	非周期性连续信号	傅立叶变换 (Fourier Transform)
2	周期性连续信号	傅立叶级数(Fourier Series)
3	非周期性离散信号	离散时域傅立叶变换 (Discrete Time Fourier Transform)
4	周期性离散信号	离散傅立叶变换(Discrete Fourier Transform)

下图是四种原信号图例：



这四种傅立叶变换都是针对正无穷大和负无穷大的信号，即信号的的长度是无穷大的，我们知道这对于计算机处理来说是不可能的，那么有没有针对长度有限的傅立叶变换呢？没有。因为正余弦波被定义成从负无穷小到正无穷大，我们无法把一个长度无限的信号组合成长度有限的信号。面对这种困难，方法是把长度有限的信号表示成长度无限的信号，可以把信号无限地从左右进行延伸，延伸的部分用零来表示，这样，这个信号就可以被看成是非周期性离散信号，我们就可以用到离散时域傅立叶变换的方法。还有，也可以把信号用复制的方法进行延伸，这样信号就变成了周期性离散信号，这时我们就可以用离散傅立叶变换方法进行变换。这里我们要学的是离散信号，对于连续信号我们不作讨论，因为计算机只能处理离散的数值信号，我们的最终目的是运用计算机来处理信号的。

但是对于非周期性的信号，我们需要用无穷多不同频率的正弦曲线来表示，这对于计算机来说是不可能实现的。所以对于离散信号的变换只有离散傅立叶变换 (DFT) 才能被适用，对于计算机来说只有离散的和有限长度的数据才能被处理，对于其它的变换类型只有在数学演算中才能用到，在计算机面前我们只能用 DFT 方法，后面我们要理解的也正是 DFT 方法。这里要理解的是我们使用周期性的信号目的是为了能够用数学方法来解决问题，至于考虑周期性信号是从哪里得到或怎样得到是无意义的。

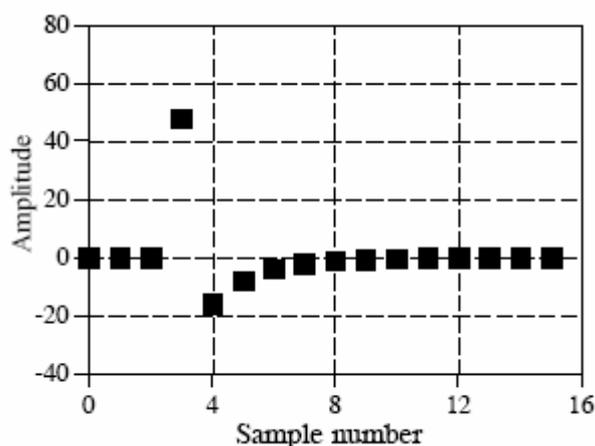
每种傅立叶变换都分成实数和复数两种方法，对于实数方法是最好理解的，但是复数方

法就相对复杂许多了，需要懂得有关复数的理论知识，不过，如果理解了实数离散傅立叶变换(real DFT)，再去理解复数傅立叶就更容易了，所以我们先把复数的傅立叶放到一边去，先来理解实数傅立叶变换，在后面我们会先讲讲关于复数的基本理论，然后在理解了实数傅立叶变换的基础上再来理解复数傅立叶变换。

还有，这里我们所要说的变换(transform)虽然是数学意义上的变换，但跟函数变换是不同的，函数变换是符合一一映射准则的，对于离散数字信号处理 (DSP)，有许多的变换：傅立叶变换、拉普拉斯变换、Z 变换、希尔伯特变换、离散余弦变换等，这些都扩展了函数变换的定义，允许输入和输出有多种的值，简单地说变换就是把一堆的数据变成另一堆的数据的方法。

三、一个关于实数离散傅立叶变换(Real DFT)的例子

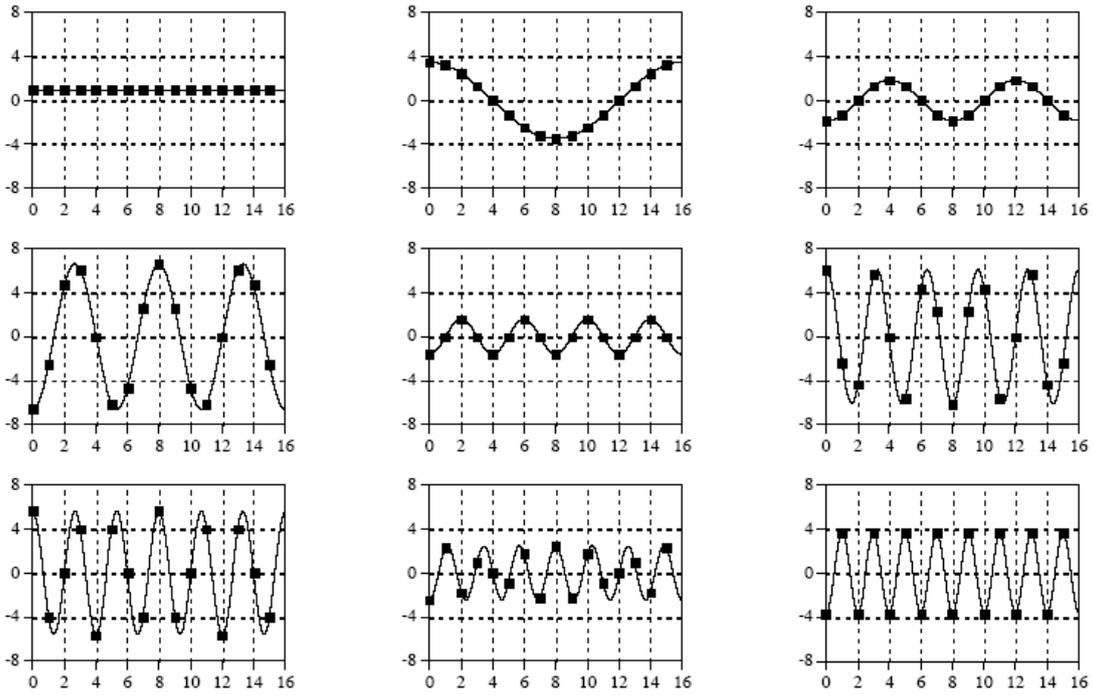
先来看一个变换实例，下图是一个原始信号图像：



这个信号的长度是 16，于是可以把这个信号分解 9 个余弦波和 9 个正弦波（一个长度为 N 的信号可以分解成 $N/2+1$ 个正余弦信号，这是为什么呢？结合下面的 18 个正余弦图，我想从计算机处理精度上就不难理解，一个长度为 N 的信号，最多只能有 $N/2+1$ 个不同频率，再多的频率就超过了计算机所能处理的精度范围），如下图：

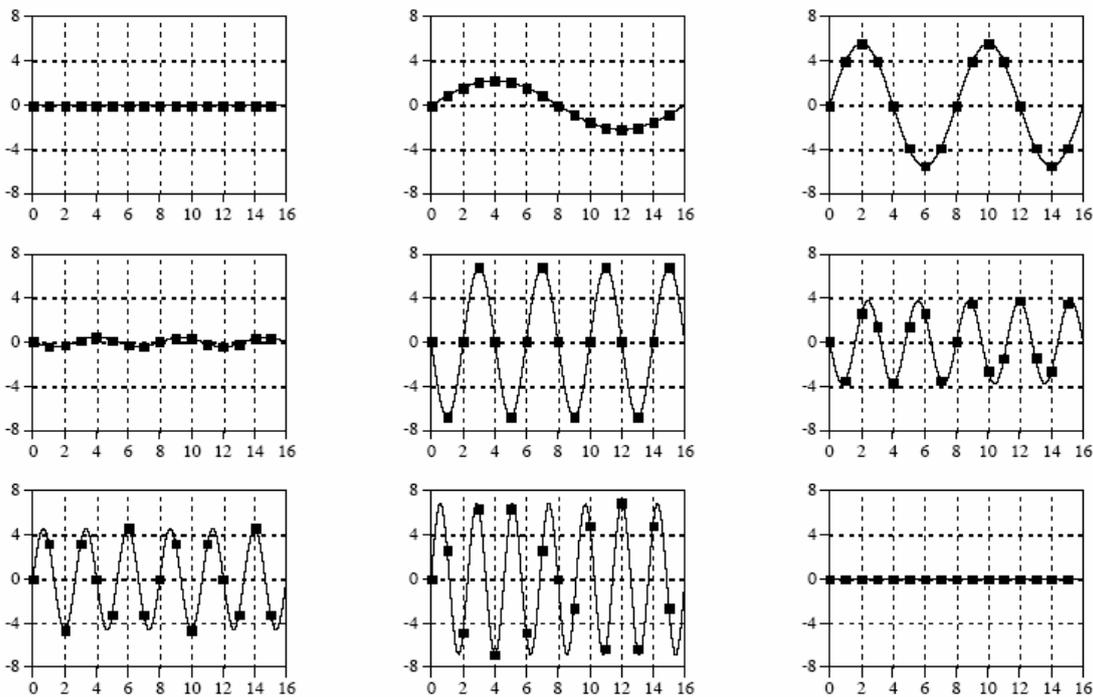
9 个余弦信号：

Cosine Waves

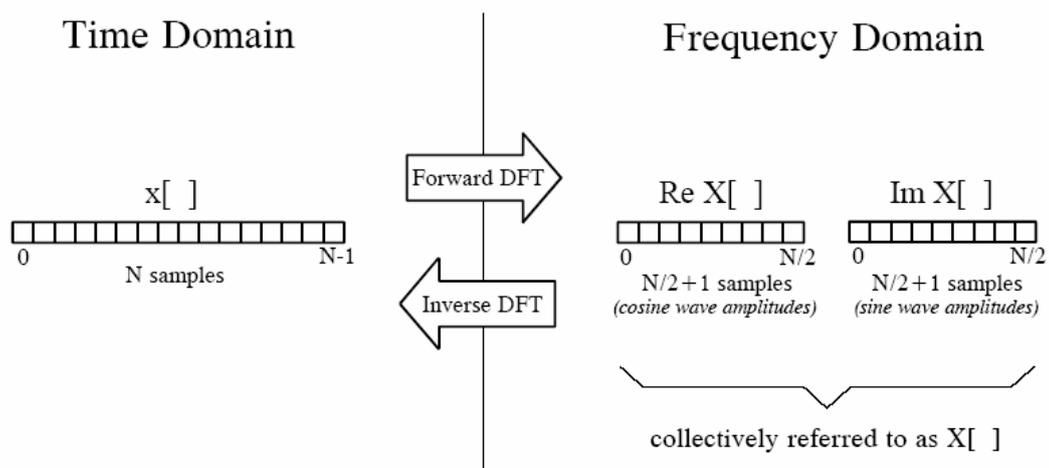


9 个正弦信号:

Sine Waves



把以上所有信号相加即可得到原始信号，至于是怎么分别变换出 9 种不同频率信号的，我们先不急，先看看对于以上的变换结果，在程序中又是该怎么表示的，我们可以看看下面这个示例图：



上图中左边表示时域中的信号，右边是频域信号表示方法，从左向右表示正向转换 (Forward DFT)，从右向左表示逆向转换 (Inverse DFT)，用小写 $x[n]$ 表示信号在每个时间点上的幅度值数组，用大写 $X[n]$ 表示每种频率的幅度值数组，因为有 $N/2+1$ 种频率，所以该数组长度为 $N/2+1$ ， $X[n]$ 数组又分两种，一种是表示余弦波的不同频率幅度值： $\text{Re } X[n]$ ，另一种是表示正弦波的不同频率幅度值： $\text{Im } X[n]$ ， Re 是实数 (Real) 的意思， Im 是虚数 (Imagine) 的意思，采用复数的表示方法把正余弦波组合起来进行表示，但这里我们不考虑复数的其它作用，只记住是一种组合方法而已，目的是为了便于表达（在后面我们会知道，复数形式的傅立叶变换长度是 N ，而不是 $N/2+1$ ）。

下一节我们将来看一下实数傅立叶变换的具体方法。

理解离散傅立叶变换（二）

-----实数形式离散傅立叶变换（Real DFT）

上一节我们看到了一个实数形式离散傅立叶变换的例子,通过这个例子能够让我们先对傅立叶变换有一个较为形象的感性认识,现在就让我们来看看实数形式离散傅立叶变换的正向和逆向是怎么进行变换的。在此,我们先来看一下频率的多种表示方法。

一、 频域中关于频率的四种表示方法

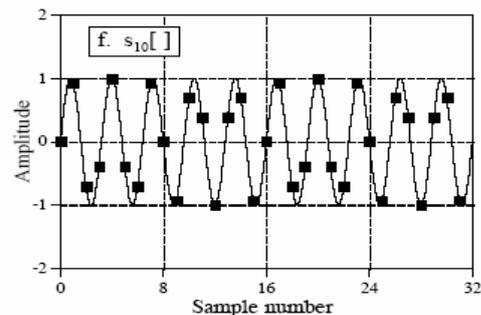
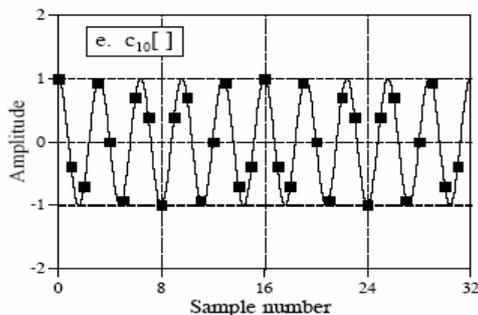
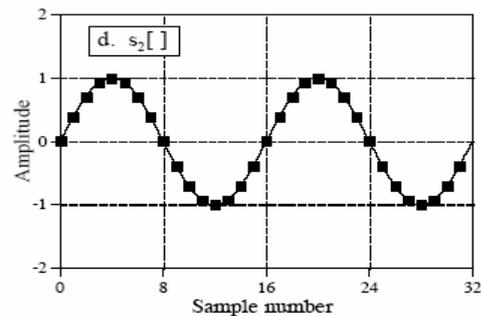
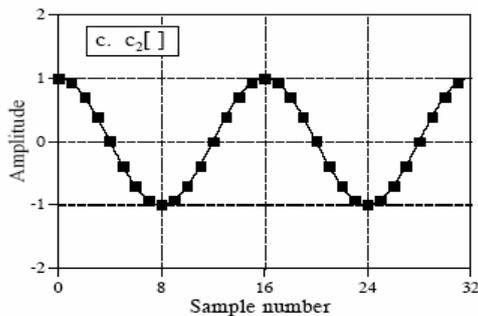
- 1、 序号表示方法, 根据时域中信号的样本数取 $0 \sim N/2$, 用这种方法在程序中使用起来可以更直接地取得每种频率的幅度值, 因为频率值跟数组的序号是一一对应的: $X[k]$, 取值范围是 $0 \sim N/2$;
- 2、 分数表示方法, 根据时域中信号的样本数的比例值取 $0 \sim 0.5$: $X[f]$, $f = k/N$, 取值范围是 $0 \sim 1/2$;
- 3、 用弧度值来表示, 把 f 乘以一个 2π 得到一个弧度值, 这种表示方法叫做自然频率 (natural frequency): $X[\omega]$, $\omega = 2\pi f = 2\pi k/N$, 取值范围是 $0 \sim \pi$;
- 4、 以赫兹(Hz)为单位来表示, 这个一般是应用于一些特殊应用, 如取样率为 10 kHz 表示每秒有 $10,000$ 个样本数: 取值范围是 0 到取样率的一半。

二、 DFT 基本函数

$$C_k[i] = \cos(2\pi ki/N)$$

$$S_k[i] = \sin(2\pi ki/N)$$

其中 k 表示每个正余弦波的频率, 如为 2 表示在 0 到 N 长度中存在两个完整的周期, 10 即有 10 个周期, 如下图:



上图中至于每个波的振幅(amplitude)值(Re X[k],Im X[k])是怎么算出来的,这个是 DFT 的核心,也是最难理解的部分,我们先来看看如何把分解出来的正余弦波合成原始信号(Inverse DFT)。

三、 合成运算方法(Real Inverse DFT)

DFT 合成等式:

$$x[i] = \sum_{k=0}^{N/2} \text{Re}\bar{X}[k] \cos(2\pi ki/N) + \sum_{k=0}^{N/2} \text{Im}\bar{X}[k] \sin(2\pi ki/N)$$

如果有学过傅立叶级数,对这个等式就会有似曾相识的感觉,不错!这个等式跟傅立叶级数是非常相似的:

$$f(x) = \frac{a_0}{2} + \sum_{k=1}^{\infty} (a_k \cos kx + b_k \sin kx)$$

当然,差别是肯定存在的,因为这两个等式是在两个不同条件下运用的,至于怎么证明 DFT 合成公式,这个我想需要非常强的高等数学理论知识了,这是研究数学的人的工作,对于普通应用者就不需要如此的追根究底了,但是傅立叶级数是好理解的,我们起码可以从傅立叶级数公式中看出 DFT 合成公式的合理性。

DFT 合成等式中的 $\text{Im } \bar{X}[k]$ 和 $\text{Re } \bar{X}[k]$ 跟 $\text{Im } X[k]$ 和 $\text{Re } X[k]$ 是不一样的,下面是转换方法:

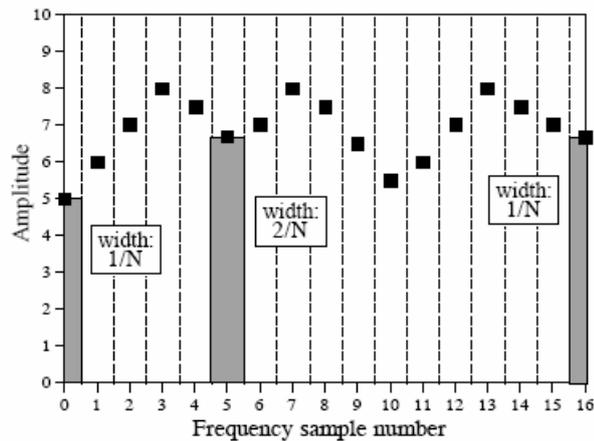
$$\begin{aligned} \text{Re}\bar{X}[k] &= \frac{\text{Re}X[k]}{N/2} \\ \text{Im}\bar{X}[k] &= -\frac{\text{Im}X[k]}{N/2} \end{aligned}$$

但 k 等于 0 和 $N/2$ 时,实数部分的计算要用下面的等式:

$$\begin{aligned} \text{Re}\bar{X}[0] &= \frac{\text{Re}X[0]}{N} \\ \text{Re}\bar{X}[N/2] &= \frac{\text{Re}X[N/2]}{N} \end{aligned}$$

上面四个式中的 N 是时域中点的总数, k 是从 0 到 $N/2$ 的序号。

为什么要这样进行转换呢?这个可以从频谱密度(spectral density)得到理解,如下图就是个频谱图:



这是一个频谱图，横坐标表示频率大小，纵坐标表示振幅大小，原始信号长度为 N （这里是 32），经 DFT 转换后得到的 17 个频率的频谱，频谱密度表示每单位带宽中为多大的振幅，那么带宽是怎么计算出来的呢？看上图，除了头尾两个，其余点的所占的宽度是 $2/N$ ，这个宽度便是每个点的带宽，头尾两个点的带宽是 $1/N$ ，而 $\text{Im } X[k]$ 和 $\text{Re } X[k]$ 表示的是频谱密度，即每一个单位带宽的振幅大小，但 $\text{Im } \bar{X}[k]$ 和 $\text{Re } \bar{X}[k]$ 表示 $2/N$ （或 $1/N$ ）带宽的振幅大小，所以 $\text{Im } \bar{X}[k]$ 和 $\text{Re } \bar{X}[k]$ 分别应当是 $\text{Im } X[k]$ 和 $\text{Re } X[k]$ 的 $2/N$ （或 $1/N$ ）。

频谱密度就象物理中物质密度，原始信号中的每一个点就象是一个混合物，这个混合物是由不同密度的物质组成的，混合物中含有的每种物质的质量是一样的，除了最大和最小两个密度的物质外，这样我们只要把每种物质的密度加起来就可以得到该混合物的密度了，又该混合物的质量是单位质量，所以得到的密度值跟该混合物的质量值是一样的。

至于为什么虚数部分是负数，这是为了跟复数 DFT 保持一致，这个我们将在后面会知道这是数学计算上的需要（ $\text{Im } X[k]$ 计算时加上了一个负号， $\text{Im } \bar{X}[k]$ 再加上负号，结果便是正的，等于没有变化）。

如果已经得到了 DFT 结果，这时要进行逆转换，即合成原始信号，则可按如下步骤进行转换：

1、先根据上面四个式子计算得出 $\text{Im } \bar{X}[k]$ 和 $\text{Re } \bar{X}[k]$ 的值；

2、再根据 DFT 合成等式得到原始信号数据。

下面是用 BASIC 语言来实现的转换源代码：

100 'DFT 逆转换方法

110 'XX[] 数组存储计算结果（时域中的原始信号）

120 'REX[] 数组存储频域中的实数分量，IMX[] 为虚分量

130 '

140 DIM XX[511]

```

150 DIM REX[256]
160 DIM IMX[256]
170 '
180 PI = 3.14159265
190 N% = 512
200 '
210 GOSUB XXXX '转到子函数去获取 REX[]和 IMX[]数据
220 '
230 '
240 '
250 FOR K% = 0 TO 256
260   REX[K%] = REX[K%] / (N%/2)
270   IMX[K%] = -IMX[K%] / (N%/2)
280 NEXT k%
290 '
300 REX[0] = REX[0] / N
310 REX[256] = REX[256] / N
320 '
330 ' 初始化 XX[]数组
340 FOR I% = 0 TO 511
350   XX[I%] = 0
360 NEXT I%
370 '
380 '
390 '
400 '
410 '
420 FOR K% =0 TO 256
430   FOR I%=0 TO 511
440 '
450     XX[I%] = XX[I%] + REX[K%] * COS(2 * PI * K% * I% / N%)
460     XX[I%] = XX[I%] + IMX[K%] * SIN(2 * PI * K% * I% / N%)
470 '
480   NEXT I%
490 NEXT K%
500 '
510 END

```

上面代码中 420 至 490 换成如下形式也许更好理解，但结果都是一样的：

```

420 FOR I% =0 TO 511
430   FOR K%=0 TO 256
440 '
450     XX[I%] = XX[I%] + REX[K%] * COS(2 * PI * K% * I% / N%)
460     XX[I%] = XX[I%] + IMX[K%] * SIN(2 * PI * K% * I% / N%)

```

```
470 ‘  
480 NEXT I%  
490 NEXT K%
```

四、 分解运算方法 (DFT)

有三种完全不同的方法进行 DFT: 一种方法是通过联立方程进行求解, 从代数的角度看, 要从 N 个已知值求 N 个未知值, 需要 N 个联立方程, 且 N 个联立方程必须是线性独立的, 但这是这种方法计算量非常的大且极其复杂, 所以很少被采用; 第二种方法是利用信号的相关性 (correlation) 进行计算, 这个是我们后面将要介绍的方法; 第三种方法是快速傅立叶变换 (FFT), 这是一个非常具有创造性和革命性的方法, 因为它大大提高了运算速度, 使得傅立叶变换能够在计算机中被广泛应用, 但这种算法是根据复数形式的傅立叶变换来实现的, 它把 N 个点的信号分解成长度为 N 的频域, 这个跟我们现在所进行的实域 DFT 变换不一样, 而且这种方法也较难理解, 这里我们先不去理解, 等先理解了复数 DFT 后, 再来看一下 FFT。有一点很重要, 那就是这三种方法所得的变换结果是一样的, 经过实践证明, 当频域长度为 32 时, 利用相关性方法进行计算效率最好, 否则 FFT 算法效率较高。现在就让我们来看一下相关性算法。

利用信号的相关性 (correlation) 可以从噪声背景中检测出已知的信号, 我们也可以利用这个方法检测信号波中是否含有某个频率的信号波: 把一个待检测信号波乘以另一个信号波, 得到一个新的信号波, 再把这个新的信号波所有的点进行相加, 从相加的结果就可以判断出这两个信号的相似程度。如下图:

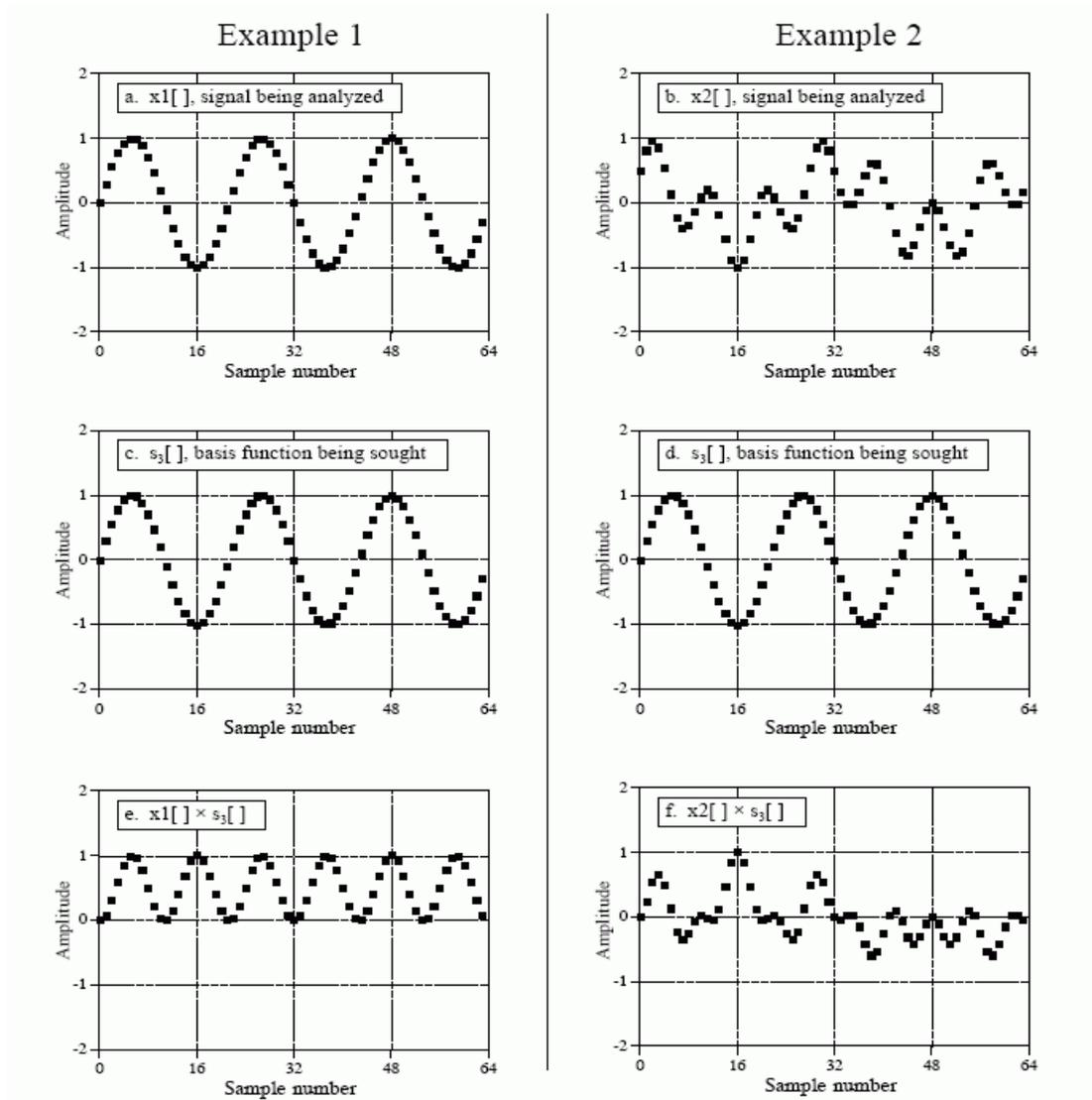


FIGURE 8-8
 Two example signals, (a) and (b), are analyzed for containing the specific basis function shown in (c) and (d). Figures (e) and (f) show the result of multiplying each example signal by the basis function. Figure (e) has an average of 0.5, indicating that $x1[]$ contains the basis function with an amplitude of 1.0. Conversely, (f) has a zero average, indicating that $x2[]$ does not contain the basis function.

上面 a 和 b 两个图是待检测信号波,图 a 很明显可以看出是个 3 个周期的正弦信号波,图 b 的信号波则看不出是否含有正弦或余弦信号,图 c 和 d 都是个 3 个周期的正弦信号波,图 e 和 f 分别是 a、b 两图跟 c、d 两图相乘后的结果,图 e 所有点的平均值是 0.5,说明信号 a 含有振幅为 1 的正弦信号 c,但图 f 所有点的平均值是 0,则说明信号 b 不含有信号 d。这个就是通过信号相关性来检测是否含有某个信号的方法。

相应地,我也可以通过把输入信号和每一种频率的正余弦信号进行相乘(关联操作),从而得到原始信号与每种频率的关联程度(即总和大小),这个结果便是我们所需要的傅立叶变换结果,下面两个等式便是我们所要的计算方法:

$$\operatorname{Re}X[k] = \sum_{i=0}^{N-1} x[i] \cos(2\pi k i / N)$$

$$\operatorname{Im}X[k] = - \sum_{i=0}^{N-1} x[i] \sin(2\pi k i / N)$$

第二个式子中加了个负号,是为了保持复数形式的一致,前面我们知道在计算 $\operatorname{Im} \bar{X}[k]$ 时又加了个负号,所以这只是个形式的问题,并没有实际意义,你也可以把负号去掉,并在计算 $\operatorname{Im} \bar{X}[k]$ 时也不加负号。

这里有一点必须明白一个正交的概念:两个函数相乘,如果结果中的每个点的总和为 0,则可认为这两个函数为正交函数。要确保关联性算法是正确的,则必须使得跟原始信号相乘的信号的函数形式是正交的,我们知道所有的正弦或余弦函数是正交的,这一点我们可以通过简单的高数知识就可以证明它,所以我们可以通过关联的方法把原始信号分离出正余弦信号。当然,其它的正交函数也是存在的,如:方波、三角波等形式的脉冲信号,所以原始信号也可被分解成这些信号,但这只是说可以这样做,却是没有用的。

下面是实域傅立叶变换的 BASIC 语言代码:

```

100 'THE DISCRETE FOURIER TRANSFORM
110 'The frequency domain signals, held in REX[ ] and IMX[ ], are calculated from
120 'the time domain signal, held in XX[ ].
130 '
140 DIM XX[511]           'XX[ ] holds the time domain signal
150 DIM REX[256]         'REX[ ] holds the real part of the frequency domain
160 DIM IMX[256]         'IMX[ ] holds the imaginary part of the frequency domain
170 '
180 PI = 3.14159265      'Set the constant, PI
190 N% = 512             'N% is the number of points in XX[ ]
200 '
210 GOSUB XXXX           'Mythical subroutine to load data into XX[ ]
220 '
230 '
240 FOR K% = 0 TO 256   'Zero REX[ ] & IMX[ ] so they can be used as accumulators
250   REX[K%] = 0
260   IMX[K%] = 0
270 NEXT K%
280 '
290 '                   'Correlate XX[ ] with the cosine and sine waves, Eq. 8-4
300 '
310 FOR K% = 0 TO 256   'K% loops through each sample in REX[ ] and IMX[ ]
320   FOR I% = 0 TO 511 'I% loops through each sample in XX[ ]
330     '
340     REX[K%] = REX[K%] + XX[I%] * COS(2*PI*K%*I%/N%)
350     IMX[K%] = IMX[K%] - XX[I%] * SIN(2*PI*K%*I%/N%)
360     '
370   NEXT I%
380 NEXT K%
390 '
400 END

```

到此为止,我们对傅立叶变换便有了感性的认识了吧。但要记住,这只是在实域上的离

散傅立叶变换，其中虽然也用到了复数的形式，但那只是个替代的形式，并无实际意义，现实中一般使用的是复数形式的离散傅立叶变换，且快速傅立叶变换是根据复数离散傅立叶变换来设计算法的，在后面我们先来复习一下有关复数的内容，然后再在理解实域离散傅立叶变换的基础上来理解复数形式的离散傅立叶变换。

理解离散傅立叶变换（三）

-----复数

复数扩展了我们一般所能理解的数的概念，复数包含了实数和虚数两部分，利用复数的形式可以把由两个变量表示的表达式变成由一个变量(复变量)来表达，使得处理起来更加自然和方便，我们知道傅立叶变换的结果是由两部分组成的，使用复数形式可以缩短变换表达式，使得我们可以单独处理一个复变量（这个在后一节的描述中我们就可以更加确切地知道），而且快速傅立叶变换正是基于复数形式的，所以几乎所有描述的傅立叶变换形式都是复数的形式。但是复数的概念超过了我们日常生活中所能理解的概念，要理解复数是较难的，所以我们在理解复数傅立叶变换之前，先来专门复习一下有关复数的知识，这对后面的理解非常重要。

一、 复数的提出

在此，先让我们看一个物理实验：把一个球从某点向上抛出，然后根据初速度和时间来计算球所在高度，这个方法可以根据下面的式子计算得出：

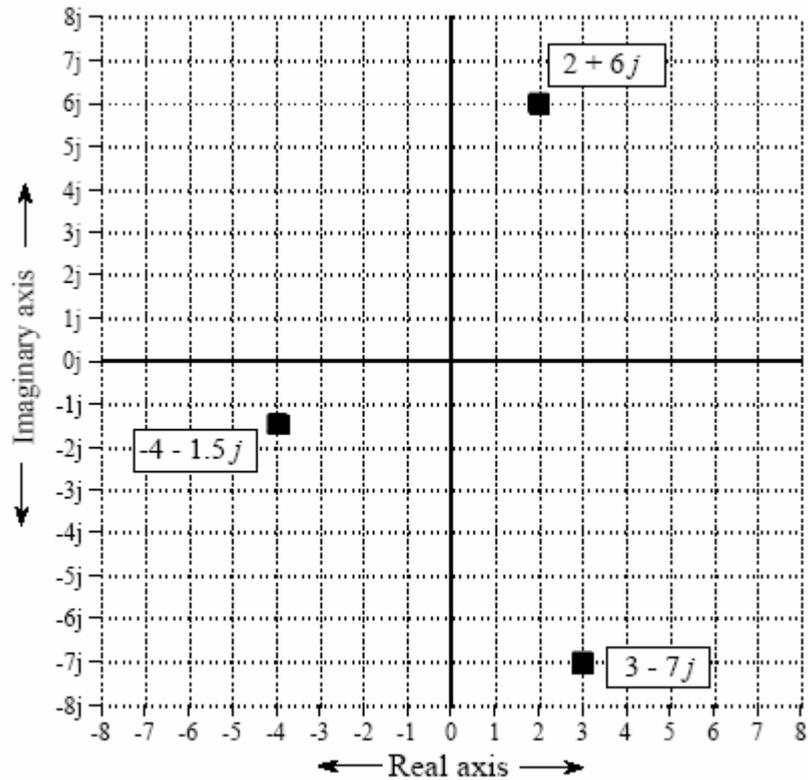
$$h = \frac{-gt^2}{2} + vt$$

其中 h 表示高度， g 表示重力加速度(9.8m/s^2)， v 表示初速度， t 表示时间。现在反过来，假如知道了高度，要求计算到这个高度所需要的时间，这时我们又可以通过下式来计算：

$$t = 1 \pm \sqrt{1 - h/4.9}$$

经过计算我们可以知道，当高度是 3 米时，有两个时间点到达该高度：球向上运动时的时间是 0.38 秒，球向下运动时的时间是 1.62 秒。但是如果高度等于 10 时，结果又是什么呢？根据上面的式子可以发现存在对负数进行开平方运算，我们知道这肯定是不现实的。第一次使用这个不一般的式子的人是意大利数学家 Girolamo Cardano (1501-1576)，两个世纪后，德国伟大数学家 Carl Friedrich Gause (1777-1855) 提出了复数的概念，为后来的应用铺平了道路，他对复数进行这样表示：复数由实数 (real) 和虚数(imaginary)两部分组成，虚数中的根号负 1 用 i 来表示（在这里我们用 j 来表示，因为 i 在电力学中表示电流的意思）。

我们可以把横坐标表示成实数，纵坐标表示成虚数，则坐标中的每个点的向量就可以用复数来表示，如下图：



上图中的 ABC 三个向量可以表示成如下的式子：

$$A = 2 + 6j$$

$$B = -4 - 1.5j$$

$$C = 3 - 7j$$

这样子来表达方便之处在于运用一个符号就能把两个原来难以联系起来的数组合起来了，不方便的是我们要分辨哪个是实数和哪个是虚数，我们一般是用 $\text{Re}()$ 和 $\text{Im}()$ 来表示实数和虚数两部分，如：

$$\text{Re } A = 2 \quad \text{Im } A = 6$$

$$\text{Re } B = -4 \quad \text{Im } B = -1.5$$

$$\text{Re } C = 3 \quad \text{Im } C = -7$$

复数之间也可以进行加减乘除运算：

$$(a + bj) + (c + dj) = (a + c) + j(b + d)$$

$$(a + bj) - (c + dj) = (a - c) + j(b - d)$$

$$(a + bj)(c + dj) = (ac - bd) + j(bc + ad)$$

$$\frac{(a + bj)}{(c + dj)} = \left(\frac{ac + bd}{c^2 + d^2} \right) + j \left(\frac{bc - ad}{c^2 + d^2} \right)$$

这里有个特殊的地方是 j^2 等于 -1, 上面第四个式子的计算方法是把分子和分母同时乘以 $c - dj$, 这样就可消去分母中的 j 了。

复数也符合代数运算中的交换律、结合律、分配律:

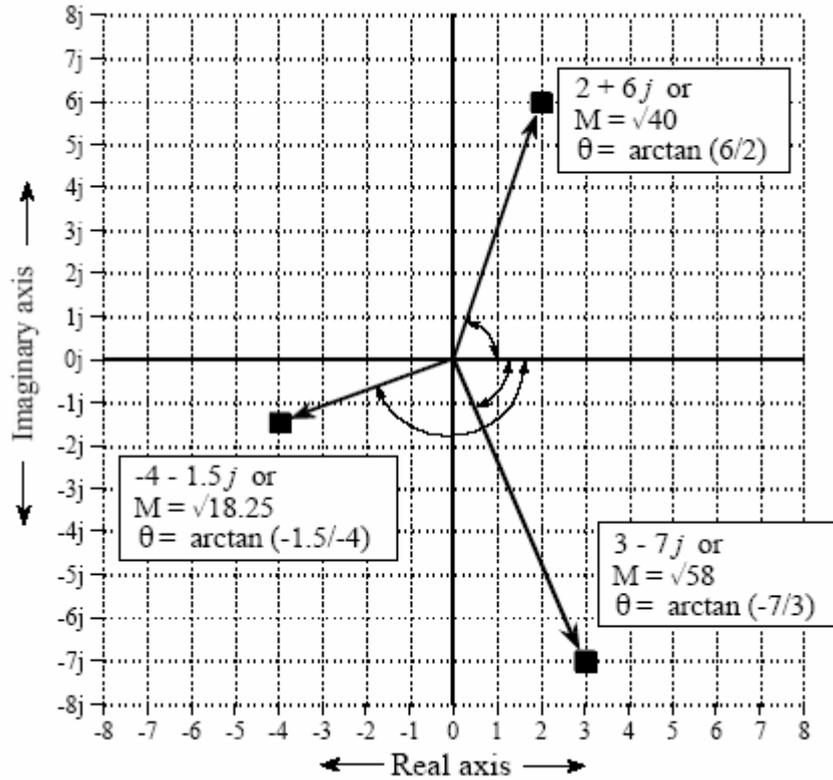
$$AB = BA$$

$$(A + B) + C = A + (B + C)$$

$$A(B + C) = AB + AC$$

二、复数的极坐标表示形式

前面提到的是运用直角坐标来表示复数, 其实更为普遍应用的是极坐标的表示方法, 如下图:



上图中的 M 即是数量积(magnitude)，表示从原点到坐标点的距离， θ 是相位角(phase angle)，表示从 X 轴正方向到某个向量的夹角，下面四个式子是计算方法：

$$M = \sqrt{(Re A)^2 + (Im A)^2}$$

$$\theta = \arctan \left[\frac{Im A}{Re A} \right]$$

$$Re A = M \cos(\theta)$$

$$Im A = M \sin(\theta)$$

我们还可以通过下面的式子进行极坐标到直角坐标的转换：

$$a + jb = M (\cos \theta + j \sin \theta)$$

上面这个等式中左边是直角坐标表达式，右边是极坐标表达式。

还有一个更为重要的等式——欧拉等式（欧拉是瑞士的著名数学家，Leonhard Euler, 1707-1783）：

$$e^{jx} = \cos x + j \sin x$$

这个等式可以从下面的级数变换中得到证明：

$$e^{jx} = \sum_{n=0}^{\infty} \frac{(jx)^n}{n!} = \left[\sum_{k=0}^{\infty} (-1)^k \frac{x^{2k}}{(2k)!} \right] + j \left[\sum_{k=0}^{\infty} (-1)^k \frac{x^{2k+1}}{(2k+1)!} \right]$$

上面中右边的两个式子分别是 $\cos(x)$ 和 $\sin(x)$ 的泰勒(Taylor)级数。

这样子我们又可以把复数的表达式表示成指数的形式了：

$$a + jb = M e^{j\theta} \quad (\text{这便是复数的两个表达式})$$

指数形式是数字信号处理中数学方法的支柱，也许是因为用指数形式进行复数的乘除运算极为简单的缘故吧：

$$M_1 e^{j\theta_1} M_2 e^{j\theta_2} = M_1 M_2 e^{j(\theta_1 + \theta_2)}$$

$$\frac{M_1 e^{j\theta_1}}{M_2 e^{j\theta_2}} = \left[\frac{M_1}{M_2} \right] e^{j(\theta_1 - \theta_2)}$$

三、复数是数学分析中的一个工具

为什么要使用复数呢？其实它只是个工具而已，就如钉子和锤子的关系，复数就象那锤子，作为一种使用的工具。我们要解决的问题表达成复数的形式（因为有些问题用复数的形式进行运算更加方便），然后对复数进行运算，最后再转换回来得到我们所需要的结果。

有两种方法使用复数，一种是用复数进行简单的替换，如前面所说的向量表达式方法和前一节中我们所讨论的实域 DFT，另一种是更高级的方法：数学等价(mathematical equivalence)，复数形式的傅立叶变换用的便是数学等价的方法，但在这里我们先不讨论这种方法，这里我们先来看一下用复数进行替换中的问题。

用复数进行替换的基本思想是：把所要分析的物理问题转换成复数的形式，其中只是简单地添加一个复数的符号 j ，当返回到原来的物理问题时，则只是把符号 j 去掉就可以了。

有一点要明白的是并不是所有问题都可以用复数来表示，必须看用复数进行分析是否适用，有个例子可以看出用复数来替换原来问题的表达方式明显是谬误的：假设一箱的苹果是 5 美元，一箱的橘子是 10 美元，于是我们把它表示成 $5 + 10j$ ，有一个星期你买了 6 箱苹果和 2 箱橘子，我们又把它表示成 $6 + 2j$ ，最后计算总共花的钱是 $(5 + 10j)(6 + 2j) = 10 + 70j$ ，

结果是买苹果花了 10 美元的，买桔子花了 70 美元，这样的结果明显是错了，所以复数的形式不适合运用于对这种问题的解决。

四、用复数来表示正余弦函数表达式

对于象 $M \cos(\omega t + \phi)$ 和 $A \cos(\omega t) + B \sin(\omega t)$ 表达式，用复数来表示，可以变得非常简洁，对于直角坐标形式可以按如下形式进行转换：

$$A \cos(\omega t) + B \sin(\omega t) \Leftrightarrow a + jb$$

(conventional representation) (complex number)

上式中余弦幅值 A 经变换生成 a ，正弦幅值 B 的相反数经变换生成 b ： $A \Leftrightarrow a$ ， $B \Leftrightarrow -b$ ，但要注意的是，这不是个等式，只是个替换形式而已。

对于极坐标形式可以按如下形式进行转换：

$$M \cos(\omega t + \phi) \Leftrightarrow M e^{j\theta}$$

(conventional representation) (complex number)

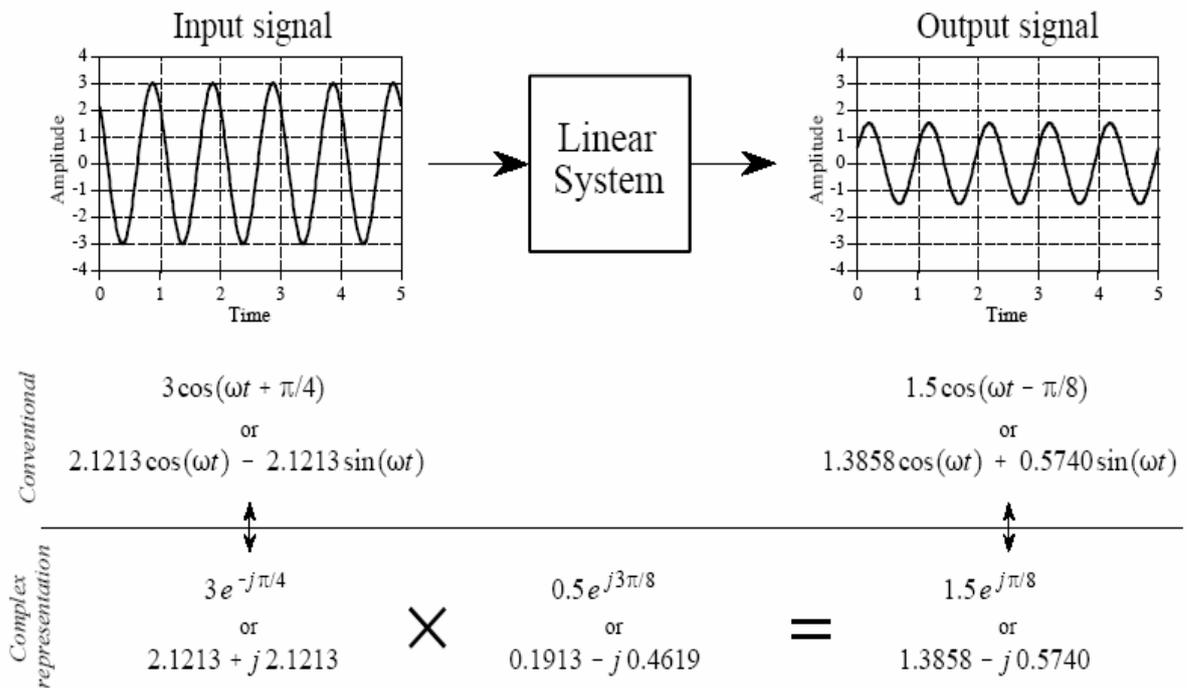
上式中， $M \Leftrightarrow M$ ， $\theta \Leftrightarrow -\phi$ 。

这里虚数部分采用负数的形式主要是为了跟复数傅立叶变换表达式保持一致，对于这种替换的方法来表示正余弦，符号的变换没有什么好处，但替换时总会被改变掉符号以跟更高级的等价变换保持形式上的一致。

在离散信号处理中，运用复数形式来表示正余弦波是个常用的技术，这是因为利用复数进行各种运算得到的结果跟原来的正余弦运算结果是一致的，但是，我们要小心使用复数操作，如加、减、乘、除，有些操作是不能用的，如两个正弦信号相加，采用复数形式进行相加，得到的结果跟替换前的直接相加的结果是一样的，但是如果两个正弦信号相乘，则采用复数形式来相乘结果是不一样的。幸运的是，我们已严格定义了正余弦复数形式的运算操作条件：

- 1、参加运算的所有正余弦的频率必须是一样的；
- 2、运算操作必须是线性的，如两个正弦信号可以进行相加减，但不能进行乘除，象信号的放大、衰减、高低通滤波等系统都是线性的，象平方、缩短、取限等则不是线性的。要记住的是卷积和傅立叶分析也只有线性操作才可以进行。

下图是一个相量变换(我们把正弦或余弦波变成复数的形式称为相量变换，Phasor transform)的例子，一个连续信号波经过一个线性处理系统生成另一个信号波，从计算过程我们可以看出采用复数的形式使得计算变化十分的简洁：



前一节中我们描述的实数形式傅立叶变换也是一种替换形式的复数变换,但要注意的是那还不是复数傅立叶变换,只是一种代替方式而已。下一节我们就会知道复数傅立叶变换是一种更高级的变换,而不是这种简单的替换形式。

理解离散傅立叶变换（四）

-----复数形式离散傅立叶变换

复数形式的离散傅立叶变换非常巧妙地运用了复数的方法,使得傅立叶变换更加自然和简洁,它并不是只是简单地运用替换的方法来运用复数,而是完全从复数的角度来分析问题,这一点跟实数 DFT 是完全不一样的。

一、 把正余弦函数表示成复数的形式

通过欧拉等式可以把正余弦函数表示成复数的形式:

$$\cos(x) = 1/2 e^{j(-x)} + 1/2 e^{jx}$$

$$\sin(x) = j(1/2 e^{j(-x)} - 1/2 e^{jx})$$

从这个等式可以看出,如果把正余弦函数表示成复数后,它们变成了由正负频率组成的正余弦波,相反地,一个由正负频率组成的正余弦波,可以通过复数的形式来表示。

我们知道,在实数傅立叶变换中,它的频谱是 $0 \sim \pi$ ($0 \sim N/2$),但无法表示 $-\pi \sim 0$ 的频谱,可以预见,如果把正余弦表示成复数形式,则能够把负频率包含进来。

二、 把变换前后都看成复数的形式

复数形式傅立叶变换成原始信号 $x[n]$ 当成是一个用复数来表示的信号,其中实数部分表示原始信号值,虚数部分为 0,变换结果 $X[k]$ 也是个复数的形式,但这里的虚数部分是有值的。在这里用复数的观点来看原始信号非常的关键,是理解复数形式傅立叶变换的关键(如果有学过复变函数则可能更好理解,即把 $x[n]$ 看成是一个复数变量,然后象对等实数那样对这个复数变量进行相同的变换)。

三、 对复数进行相关性算法(正向傅立叶变换)

从实数傅立叶变换中可以知道,我们可以通过原始信号把乘以一个正交函数形式的信号,然后进行求总和,最后就能得到这个原始信号所包含的正交函数信号的分量。现在我们的原始信号变成了复数,我们要得到的是复数的信号分量,我们是不是可以把它乘以一个复数形式的正交函数呢?答案是肯定的,正余弦函数都是正交函数,变成如下形式的复数后,仍旧还是正交函数(这个从正交函数的定义可以很容易得到证明):

$$\cos x + j \sin x, \cos x - j \sin x, \dots$$

这里我采用上面的第二个式子进行相关性求和,为什么用第二个式,我在后面会知道,正弦函数在虚数中变换后得到的是负的正弦函数,这里我们再加上一个负号,使得最后得到的是正的正弦波,根据这个于是我们很容易就可以得到了复数形式的 DFT 正向变换等式:

$$X[k] = \frac{1}{N} \sum_{n=0}^{N-1} x[n] \left(\cos(2\pi kn/N) - j \sin(2\pi kn/N) \right)$$

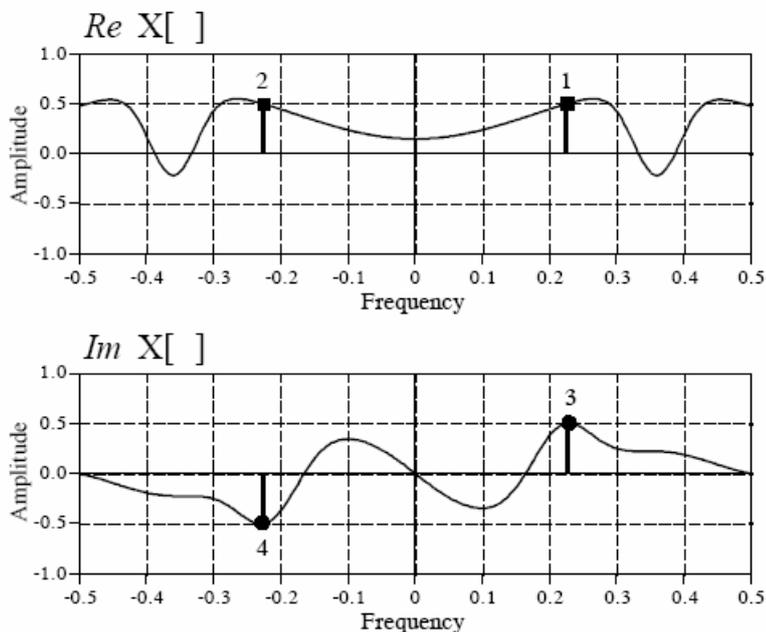
这个式子很容易可以得到欧拉变换式子：

$$X[k] = \frac{1}{N} \sum_{n=0}^{N-1} x[n] e^{-j2\pi kn/N}$$

其实我们是为了表过上的方便才用到欧拉变换式，在分析问题是我们还是较多地用到正余弦表达式。

对于上面的等式，我们要清楚如下几个方面（也是区别于实数 DFT 的地方）：

- 1、 $X[k]$ 、 $x[n]$ 都是复数，但 $x[n]$ 的虚数部分都是由 0 组成的，实数部分表示原始信号；
- 2、 k 的取值范围是 $0 \sim N-1$ (也可以表达成 $0 \sim 2\pi$)，其中 $0 \sim N/2$ (或 $0 \sim \pi$) 是正频部分， $N/2 \sim N-1$ ($\pi \sim 2\pi$) 是负频部分，由于正余弦函数的对称性，所以我们将 $-\pi \sim 0$ 表示成 $\pi \sim 2\pi$ ，这是出于计算上方便的考虑。
- 3、其中的 j 是一个不可分离的组成部分，就象一个等式中的变量一样，不能随便去掉，去掉之后意义就完全不一样了，但我们知道在实数 DFT 中， j 只是个符号而已，把 j 去掉，整个等式的意义不变；
- 4、下图是个连续信号的频谱，但离散频谱也是与此类似的：



上面的频谱图把负频率放到了左边，是为了迎合我们的思维习惯，但在实际实现中我们一般是把它移到正的频谱后面的。

从上图可以看出，时域中的正余弦波（用来组成原始信号的正余弦波）在复数 DFT 的频谱中被分成了正、负频率的两个组成部分，基于此等式中前面的比例系数是 $1/N$ （或 $1/2\pi$ ），而不是 $2/N$ ，这是因为现在把频谱延伸到了 2π ，但把正负两个频率相加即又得到了 $2/N$ ，又还原到了实数 DFT 的形式，这个在后面的描述中可以更清楚地看到。由于复数 DFT 生成的是一个完整的频谱，原始信号中的每一个点都是由正、负两个频率组合而成的，所以

频谱中每一个点的带宽是一样的，都是 $1/N$ ，相对实数 DFT，两端带宽比其它点的带宽少了一半；

复数 DFT 的频谱特征具有周期性： $-N/2 \sim 0$ 与 $N/2 \sim N-1$ 是一样的，实域频谱呈偶对称性（表示余弦波频谱），虚域频谱呈奇对称性（表示正弦波频谱）。

四、 逆向傅立叶变换

假设我们已经得到了复数形式的频谱 $X[k]$ ，现在要把它还原到复数形式的原始信号 $x[n]$ ，当然应该是把 $X[k]$ 乘以一个复数，然后再进行求和，最后得到原始信号 $x[n]$ ，这个跟 $X[k]$ 相乘的复数首先让我们想到的应该是上面进行相关性计算的复数： $\cos(2\pi kn/N) - j \sin(2\pi kn/N)$ ，但其中的负号其实是为了使得进行逆向傅立叶变换时的正弦函数变为正的符号，因为虚数 j 的运算特殊性，使得原来应该是正的正弦函数变为了负的正弦函数（我们后面的推导会看到这一点），所以这里的负号只是为了纠正符号的作用，在进行逆向 DFT 时，我们可以把负号去掉，于是我们便得到了这样的逆向 DFT 变换等式：

$$x[n] = \sum_{k=0}^{N-1} X[k] (\cos(2\pi kn/N) + j \sin(2\pi kn/N))$$

我们现在来分析这个式子，会发现这个式其实跟实数傅立叶变换是可以得到一样结果的。我们先把 $X[k]$ 变换一下：

$$X[k] = \text{Re } X[k] + j \text{Im } X[k]$$

这样我们就可以对 $x[n]$ 再次进行变换，如：

$$\begin{aligned} x[n] &= \sum_{k=0}^{N-1} (\text{Re } X[k] + j \text{Im } X[k]) (\cos(2\pi kn/N) + j \sin(2\pi kn/N)) \\ &= \sum_{k=0}^{N-1} (\text{Re } X[k] \cos(2\pi kn/N) + j \text{Im } X[k] \cos(2\pi kn/N) + \\ &\quad j \text{Re } X[k] \sin(2\pi kn/N) - \text{Im } X[k] \sin(2\pi kn/N)) \\ &= \sum_{k=0}^{N-1} (\text{Re } X[k] (\cos(2\pi kn/N) + j \sin(2\pi kn/N)) + \dots \text{---(1)} \end{aligned}$$

$$\text{Im } X[k] (-\sin(2\pi kn/N) + j \cos(2\pi kn/N))) \text{---(2)}$$

这时我们就把原来的等式分成了两个部分，第一个部分是跟实域中的频谱相乘，第二个部分是跟虚域中的频谱相乘，根据频谱图我们可以知道， $\text{Re } X[k]$ 是个偶对称的变量， $\text{Im } X[k]$ 是个奇对称的变量，即

$$\text{Re } X[k] = \text{Re } X[-k]$$

$$\text{Im } X[k] = -\text{Im } X[-k]$$

但 k 的范围是 $0 \sim N-1$ ， $0 \sim N/2$ 表示正频率， $N/2 \sim N-1$ 表示负频率，为了表达方便我们把 $N/2 \sim N-1$ 用 $-k$ 来表示，这样在从 0 到 $N-1$ 的求和过程中对于(1)和(2)式分别有 $N/2$ 对的 k 和 $-k$ 的和，对于(1)式有：

$$\text{Re } X[k] (\cos(2\pi kn/N) + j \sin(2\pi kn/N)) + \text{Re } X[-k] (\cos(-2\pi kn/N) + j \sin(-2\pi kn/N))$$

根据偶对称性和三角函数的性质，把上式化简得到：

$$\operatorname{Re} X[k] (\cos(2\pi kn/N) + j \sin(2\pi kn/N)) + \operatorname{Re} X[k] (\cos(2\pi kn/N) - j \sin(2\pi kn/N))$$

这个式最后的结果是：

$$2 \operatorname{Re} X[k] \cos(2\pi kn/N)$$

再考虑到求 $\operatorname{Re} X[k]$ 等式中有个 $1/N$ ，把 $1/N$ 乘以 2，这样的结果不就是跟实数 DFT 中的式子一样了吗？

对于(2)式，用同样的方法，我们也可以得到这样的结果：

$$-2 \operatorname{Im} X[k] \sin(2\pi kn/N)$$

注意上式前面多了个负符号，这是由于虚数变换的特殊性造成的，当然我们肯定不能把负符号的正弦函数跟余弦来相加，还好，我们前面是用 $\cos(2\pi kn/N) - j \sin(2\pi kn/N)$ 进行相关性计算，得到的 $\operatorname{Im} X[k]$ 前面有个负的符号，这样最后的结果中正弦函数就没有负的符号了，这就是为什么在进行相关性计算时虚数部分要用到负符号的原因（我觉得这也许是复数形式 DFT 美中不足的地方，让人有一种拼凑的感觉）。

从上面的分析中可以看出，实数傅立叶变换跟复数傅立叶变换，在进行逆变换时得到的结果是一样的，只不过是殊途同归吧。