

# 机器计算复数傅里叶级数的一种算法

James W.Cooley Jhon W.Tukey

2015 年 1 月 21 日

一个  $2^m$  阶的交互析因实验的有效方法被 Yates 提出，并以他的名字命名。它的  $3^m$  阶的一般化方法被 Box et al. [1] 提出。Good [?] 概括总结了这些方法并给出了简洁的算法用于一类傅里叶级数的计算的应用。在他们的所有概论中，Good 的方法必须应用到一个  $N$  维向量乘以一个可以被分解为  $m$  个稀疏矩阵乘积的  $N \times N$  矩阵，其中  $m = O(\log N)$ 。这个结果使得一个程序需要的计算量为  $O(N \log N)$  而不是  $O(N^2)$ 。这些理论被用于计算傅里叶级数。当数据点个数是或者能选择一些数据是高度复合数时，它们是有效的。这个算法在这里表现为一种不同的形式。注意到选择的  $N$ ，在一个二进制计算机上使用  $N = 2^m$ ，能展现出很大的优势以及  $N$  个数据存储位置内用于给定傅里叶系数能让整个计算可执行。

考虑如下的复数傅里叶级数计算的问题：

$$X(j) = \sum_{k=0}^{N-1} A(k) \cdot W^{jk}, j = 0, 1, \dots, N-1, \quad (1)$$

其中给定的傅里叶系数  $A(k)$  为复数， $W$  为  $N$  次单位本原根，即

$$W = e^{2\pi i/N} \quad (2)$$

使用 (1) 的直接计算需要  $N^2$  次数的操作，这里的操作是指，一个复数乘法接着一个复数的加法，这个定义贯穿本文。

这里描述的这个算法遍历整个给出的傅里叶振幅数组，产生结果需要的操作次数小于  $2N \log N$  相比给定的数组  $A$  不需要额外的数据存储量。下面来得到这个算法，假设  $N$  为一个合数，不妨设  $N = r_1 \cdot r_2$ 。令 (1) 中下标如下给出：

$$\begin{aligned} j &= j_1 r_1 + j_0, j_0 = 0, 1, \dots, r_1 - 1, j_1 = 0, 1, \dots, r_2 - 1, \\ k &= k_1 r_2 + k_0, k_0 = 0, 1, \dots, r_2 - 1, k_1 = 0, 1, \dots, r_1 - 1. \end{aligned} \quad (3)$$

于是，我们可以得到，

$$X(j_1, j_0) = \sum_{k_0} \sum_{k_1} A(k_1, k_0) \cdot W^{j k_1 r_2} W^{j k_0} \quad (4)$$

由于

$$W^{jk_1r_2} = W^{j_0k_1r_2} \quad (5)$$

对于内部求和与  $k_1$  无关, 仅仅决定于  $j_0$  和  $k_0$ , 可以被定义为一个新数组:

$$A_1(j_0, k_0) = \sum_{k_1} A(k_1, k_0) \cdot W_0^j k_1 r_2 \quad (6)$$

于是结果可以写为:

$$X(j_1, j_0) = \sum_{k_0} A_1(j_0, k_0) \cdot W^{(j_1r_1+j_0)k_0} \quad (7)$$

## 参考文献

- [1] G.E.P.Box,L.R.Connor,W.R.Cousins,O.L.Davies(Ed.),F.R.Hirnsworth& G.P.Silitto,The Design and Analysis of Industrial Experiments,Oliver& Boyd,Edinburgh, 1954.