有限长序列线性相关的快速算法研究

刘志君[1]，戚晨皓[2]

(1. 东南大学 吴健雄学院，江苏 南京 211189;
2. 东南大学 信息科学与工程学院，江苏 南京 211189)

摘要: 本文根据线性相关的定义，给出了基于快速傅立叶变换 FFT(Fast Fourier Transform) 直接计算线性相关的快速算法; 随后针对长度相差较大的序列，提出了一种分段求和 FFT 算法，其运算量比直接 FFT 算法更小。仿真结果表明，若参与线性相关的两个序列长度相差较大，相比于直接 FFT 算法，分段求和 FFT 算法具有更小的运算量，且序列长度差距越大，改善效果越好。

关键词: 数字信号处理; 线性相关; 快速傅里叶变换; 分段求和

中图分类号: TN911.72 **文献标识码**: A **文章编号**: 1008-0686(2021) 05-0120-04

Fast Algorithm Research on Linear Correlation of Finite Length Sequences

LIU ZHi-jun[1], **QI CHen-hao**[2]

(1. *Chien-Shiung Wu College*, *Southeast University*, *Nanjing* 211198, *China*;
2. *School of Information Science and Engineering*, *Southeast University*, *Nanjing* 211198, *China*)

Abstract: In this paper, according to the definition of linear correlation, an algorithm based on fast Fourier transform (FFT) for direct calculation of linear correlation is given. To reduce the computational complexity, a piecewise summation FFT algorithm is proposed for linear correlation between two finite length sequences with large length differences. Simulation results show that if the length of the two sequences involved in the linear correlation is quite different, the piecewise summation FFT algorithm has lower computation than the direct FFT algorithm, which becomes much clearer with the larger difference between the lengths of two sequences.

Keywords: digital signal processing; linear correlation; fast Fourier transform; stepwise summation

0 引言

在“数字信号处理”中，相关是一个十分重要的信号分析与处理的工具，在时延估计、随机信号的统计特性分析以及随机信号的功率谱估计等方面有着重要的应用[1]，例如平稳随机信号的功率谱密度就是其自相关函数的傅里叶变换[2]。因此计算两个有限长序列的线性相关是十分重要的内容，而相关文献对于线性相关的 FFT 算法研究甚少，所以有必要对其快速运算作一些探讨和研究，特别是长序列数字信号处理的快速算法，以期达到实时性的目的[3]。本文首先介绍了已有的直接 FFT 算法快速计算线性相关，而当两序列长度相差较大时，直接 FFT 算法的快速性不够明显，因此本文重点研究了如何利用分段求和 FFT 算法来计算线性相关，该算法相比于直接 FFT 算法显著减少了运算量。

收稿日期: 2021-04-14; 修回日期: 2021-05-07

基金项目: 东南大学校级教学改革与研究项目(项目编号: 2019-031、2019-029、2019-026) ，教育部电子信息类教指委教改项目(项目编号: 2020-YB-44) ，江苏省高等教育教改课题(项目编号: 2019JSJG069)

第一作者: 刘志君(1999-) ，男，本科生，研究方向: 信号处理方面的研究，E-mail: 2874361431@ qq. com

1　直接 FFT 算法计算线性相关

1.1　直接 FFT 算法

设两有限长序列 $x(n)$、$h(n)$ 的长度分别为 N、M，则 $x(n)$ 与 $h(n)$ 之间的线性相关的结果（又称互相关函数）$r_{xh}(m)$ 和 $r_{hx}(m)$ 定义为：

$$r_{xh}(m)=\sum_{n=0}^{M-1}x(n-m)h(n)$$

$$=\sum_{n=0}^{N-1}x(n)h(n+m) \tag{1}$$

$$r_{hx}(m)=\sum_{n=0}^{N-1}h(n-m)x(n)$$

$$=\sum_{n=0}^{M-1}h(n)x(n+m) \tag{2}$$

观察式（1）和（2）我们可以发现两种不同互相关函数之间的关系：

$$r_{hx}(m)=\sum_{n=0}^{M-1}x(n-(-m))h(n)=r_{xh}(-m) \tag{3}$$

根据定义可以发现，互相关函数的长度为 $N+M-1$，且两个有限长序列的互相关函数有两个[4]，但是二者之间有明显的关联性，即 $r_{hx}(m)=r_{xh}(-m)$。如果直接使用定义计算线性相关，其运算量为 NM 次乘法，时间复杂度为 $O(NM)$。

为了利用 FFT 快速计算线性相关，我们需要用到线性卷积的相关内容[2]，将 $r_{xh}(m)$ 的公式与线性卷积的公式相比较，可以得到二者的时域关系为：

$$r_{xh}(m)=\sum_{n=0}^{M-1}x(n-m)h(n)$$

$$=\sum_{n=0}^{M-1}x(-(m-n))h(n)$$

$$=x(-m)*h(m) \tag{4}$$

根据式（4），我们就可以利用线性卷积的 FFT 算法[1]快速计算线性相关。这里我们还需要用到循环相关的概念，对于长度分别为 N 和 M 的有限长序列 $x(n)$、$h(n)$，其 L 点循环相关的结果（$L\geqslant\max\{N,M\}$）定义为：

$$\sum_{n=0}^{N-1}h(n)x((n-m))_L R_L=\mathrm{IDFT}\left[X^*(k)Y(k)\right] \tag{5}$$

式（5）中：$((\cdot))_L$ 表示对 L 求余数，$R_L(n)$ 为矩形序列，$X(k)$ 与 $Y(k)$ 均为 L 点 FFT 的结果。

循环相关和线性相关的等价关系[1]为 $L\geqslant N+M-1$，故直接 FFT 算法计算线性相关的过程如下：①取 $L=N+M-1$；②对 $x(n)$ 和 $h(n)$ 做 L 点 FFT 得到 $X(k)$ 与 $H(k)$；③然后将 $X^*(k)$ 与 $H(k)$ 相乘得到 R_{xh}；④对 R 做 L 点 IFFT 得到 $\hat{r}_{xh}$。

但是，此时得到的 $\hat{r}_{xh}$ 并不是最后线性相关的结果 $r_{(}\mathrm{m})$。要注意的是[5]：$r_{xh}(m)$ 的 m 值范围不同于卷积的 $0\leqslant n\leqslant N+M-2$，而为 $-M+1\leqslant m\leqslant N-1$。因此 $\hat{r}_{xh}$ 的后 N－1 项对应于 $r_{xh}(m)$，$-M+1\leqslant m\leqslant N+M-1$；$\hat{r}_{xh}$ 的前 M 项，对应于 $r_{xh}(m)$，$N-M\leqslant m\leqslant N-1$。因此还需要将 $\hat{r}_{xh}$ 的数据重新拼接才能得到 r_{xh}。

从上述过程来看需要 3 次 *FFT* 运算，但是在实际运用中，h(n) 是设计好的参数，在设计时直接给出 H(k)，因此只需要 2 次 *FFT* 计算和第三步的 L 次乘法，因此得到直接 *FFT* 算法的运算量为 $(L\log_2 L+L)$ 次乘法[1]，时间复杂度为 $O(L\log_2 L)$。

1.2　直接 FFT 算法运算量的改进

前文讨论线性相关的运算时并没有考虑到 N 和 M 的关系对于直接 FFT 算法改进程度的影响。因此需要定义一个比值

$$\mathcal{R}_1=\frac{\text{直接 FFT 算法的运算量}}{\text{使用定义计算的运算量}} \tag{6}$$

通过式（6）来讨论 N 和 M 的关系对于直接 FFT 算法改进程度的影响[3]。根据已经得到的运算量的结果我们可以得到：

$$\mathcal{R}_1=\frac{L\log_2 L+L}{NM} \tag{7}$$

当 $N\approx M$ 时，可近似认为 $L=N+M-1\approx 2N$，将两种算法的运算量进行比较（表 1）。

表 1　N 与 M 接近时运算量的比较

长度 $N=M$	使用定义计算的运算量/乘法次数	直接 FFT 算法的运算量/乘法次数	R_1
8	64	80	1.250
16	256	192	0.750
32	1024	448	0.438
64	4096	1024	0.250
128	16384	2304	0.141
256	65536	5120	0.078
512	262144	11264	0.043

从表 1 中可以看出，当 $N=M$ 时，N 越大，直接 FFT 算法的运算量改善效果越好，在 $N=M\geqslant 16$ 时，直接 FFT 算法的运算量就已经明显小于使用定义计算的运算量。

但是在实际的信号处理中，两序列的长度相差较大，一般数字信号处理的单位冲激响应 $h(n)$ 较短，而数字信号 $x(n)$ 的长度较长。如果使用直接FFT算法进行计算，$h(n)$ 必须补很多个零值点，这样一来很不经济，二来快速性不明显[3]。在式(7)中，如果 $N \gg M$，可以近似认为 $L=N+M-1\approx N$，此时 $R_1=\frac{log_2 N+1}{M}$，如果 M 不变，随着 N 增加到大于 2^M，R_1 值反而会增大到超过1，这意味着直接FFT算法不仅没有起到显著减少运算量的作用，反而会在 N 大于 2^M 时增加运算量，这是我们所不期望的，因此需要改善FFT算法，这就是以下将重点介绍的分段求和FFT算法。

2 分段求和FFT算法计算线性相关

2.1 分段求和FFT算法

对于FFT算法来说，先分段计算最后求和是一种很典型的改进计算量的方法[1]，其核心就在于将一部分乘法变为加法，从而达到减小计算量的目的，因此我们考虑设计分段求和FFT算法来快速计算长度相差较大的两序列的互相关函数。

从线性相关的定义式(1)中可以发现，对于任意 $r_{xh}(m)$，由于 $h(n)$ 的长度为 $M(M\ll N)$，因此求和公式中只有 $n=0$ 到 $n=M-1$ 共 M 点值不为零，因此 $x(n)$ 也只有 M 点的值与 $r_{xh}(m)$ 有关，故可以在长度为 N 的长序列 $x(n)$ 末尾补最少数量的零，使其补零后的长度 $\tilde{\boldsymbol{N}}$ 为 M 的整数 k 倍，即 $\boldsymbol{k=\left\lceil\frac{N}{M}\right\rceil=\frac{\tilde{N}}{M}}$。然后将长度为 $\tilde{\boldsymbol{N}}$ 的序列分为 k 段长度为 M 的序列 $x_i(n)$，由此我们可以得到如下关系：

$$x_i(n)=x(n+(i-1)M)$$
$$i=1,2\ldots k,0\le n\le M-1 \tag{8}$$

之后利用直接FFT算法分别计算 $x_i(n)$ 与 $h(n)$ 的互相关函数 $r_{x_ih}(n)$，但是每个 $r_{x_ih}(n)$ 的长度都为 $2M-1$，而最后需要得到的结果 $r_{xh}(m)$ 长度为 $N+M-1$，如果计算补零后的长度则为 $\tilde{\boldsymbol{N}}+M-1=(k+1)M-1$，因此在最后求和时，相邻两个 $r_{x_ih}(n)$ 必然有 $(M-1)$ 个点的值要重叠相加。

经过检验，需要重叠相加的部分为 $r_{x_ih}(n)$ 的后 $M-1$ 项与 $r_{x_{i+1}h}(n)$ 的第2项到第 M 项，将相加后的值重新组成 $k+1$ 个长度为 M 的序列 $\tilde{\boldsymbol{r}}_{\boldsymbol{x_1h}}(n)$，具体关系如下：

$$\tilde{r}_{x_ih}(n)=\begin{cases}r_{x_ih}(n),i=1,1\le n\le M.\\ r_{x_ih}(n),i=2\ldots k,n=1.\\ r_{x_ih}(n)+r_{x_{i-1}h}(n+M-1),i=2\ldots k,2\le n\le M.\\ 0,i=k+1,n=1.\\ r_{x_{i-1}h}(n+M-1),i=k+1,2\le n\le M.\end{cases} \tag{9}$$

然后从 $\tilde{\boldsymbol{r}}_{\boldsymbol{x_{k+1}h}}(n)$ 开始到 $\tilde{\boldsymbol{r}}_{\boldsymbol{x_1h}}(n)$ 结束，将每个序列的结果顺序排列得到长度为 $(k+1)M$ 的结果，再去除前面为零的几项，即可得到长度为 $N+M-1$ 的最后结果 $r_{xh}(m)$。为方便理解，这里通过图1矩阵的方式来具体解释重叠相加和排列过程。

$$\begin{bmatrix} r_{x_1h}(1) & \ldots & r_{x_1h}(2M-1)\\ r_{x_2h}(1) & & r_{x_2h}(2M-1)\\ \vdots & \ddots & \vdots\\ r_{x_{k-1}h}(1) & & r_{x_{k-1}h}(2M-1)\\ r_{x_kh}(1) & \ldots & r_{x_kh}(2M-1)\end{bmatrix}$$

(a) k 个长度为 $2M-1$ 的互相关函数排为 k 行矩阵

$$\begin{bmatrix} \tilde{r}_{x_1h}(1) & \tilde{r}_{x_1h}(2) & \cdots & \tilde{r}_{x_1h}(M)\\ \tilde{r}_{x_2h}(1) & \tilde{r}_{x_2h}(2) & \cdots & \tilde{r}_{x_2h}(M)\\ \vdots & \vdots & \ddots & \vdots\\ \tilde{r}_{x_kh}(1) & \tilde{r}_{x_kh}(2) & \cdots & \tilde{r}_{x_kh}(M)\\ \tilde{r}_{x_{k+1}h}(1) & \tilde{r}_{x_{k+1}h}(2) & \cdots & \tilde{r}_{x_{k+1}h}(M)\end{bmatrix}$$

(b) 重叠相加为 $k+1$ 个长度为 M 的序列排为 $k+1$ 行矩阵

$$[\tilde{r}_{x_{k+1}h}(1)\quad \tilde{r}_{x_{k+1}h}(2)\quad \ldots\quad \tilde{r}_{x_{k+1}h}(M)\quad \tilde{r}_{x_kh}(1)\ldots\quad \tilde{r}_{x_1h}(M-1)\quad \tilde{r}_{x_1h}(M)]$$

(c) 重新排列为长度为 $(k+1)M$ 的序列

图1 重叠相加和排列过程

根据求解过程可以得到分段求和FFT算法总的运算量为 $k(W\log_2 W+W)$ 次乘法，其中 $W\triangleq 2M-1$，时间复杂度为 $O(N\log_2 M)$。

2.2 分段求和FFT算法运算量的改进

为反映分段求和FFT算法的改进程度，我们再定义一个比值：

$$\mathcal{R}_2=\frac{\text{分段求和 FFT 算法的运算量}}{\text{直接 FFT 算法的运算量}} \tag{9}$$

根据已经得到的运算量的结果我们可以得到：

$$\mathcal{R}_2=\frac{k(W\log_2 W+W)}{L\log_2 L+L} \tag{10}$$

当 $N\gg M$ 时，可近似认为 $L\approx N$，$N\approx\tilde{\boldsymbol{N}}=kM$，将两种算法的运算量进行比较(表2)。

从表2中可以看出，当 $N\gg M$ 时，如果 M 不变，N 越大，分段求和FFT算法的改善效果越好。在 $N=7$，$M=2$，即 $x(n)$ 长度约为 $h(n)$ 长度的4倍时，分段求和FFT算法的运算量与直接FFT算法的运算量相当。

(下接第126页)

干扰识别的方法和基于角闪烁转发式假目标鉴别方法,并在微波暗室中进行了辐射式仿真试验验证,结果表明信噪比高于 15dB 时,目标鉴别成功率超过 90%,能显著提高我雷达对转发干扰的适应能力,具有较大的推广应用价值。

4 结语

将组网雷达对抗仿真平台应用于雷达及其对抗相关课程的创新实践,能够帮助学生深刻理解课程中的物理概念和理论方法,还能够帮助学生迅速建立系统性概念,有针对性地围绕某一处理环节或某一算法进行创新。自 2014 年教学改革创新实践开展以来,大量学生通过该系统提高了动手实践能力,创新提出了多种干扰、抗干扰、侦察等方面的算法,取得了一系列研究成果。后续还将进一步升级系统能力,拓展应用范围,为进一步增强我校信息与通信工程学科类学生的科研创新能力提供平台支撑。

参考文献:

[1] 王伟,徐振海,傅其祥,肖顺平."雷达原理与系统"课程教学实验平台建设[J].南京:电气电子教学学报,2010 ,32(5):93-95.

[2] 赵锋,刘进,肖顺平."雷达原理与系统"课程教学实验平台建设[J].南京:电气电子教学学报 2012,34(1):76-78.

[3] 赵锋,毕莉,王伟,肖顺平."电子系统仿真"课程设计实验教学体系建设[J].南京:电气电子教学学报,2012,34(3):64-65.

[4] 刘金林,吴杰长,曾凡明.实战化教学质量控制模型构建方法及应用[J].长沙:高等教育研究学报,2017,40(4):79-85.

[5] 付强,何峻,谢华英.新型作战力量人才培养研究与实践-以精确制导教学体系创建为例[J].长沙:高等教育研究学报,2017,40(4):20-24.

[6] 王国玉,汪连栋等.雷达电子战系统数学仿真与评估[M].北京:国防工业出版社,2004.

(上接第 122 页刘志君等文)

在 $N=15$,$M=2$,即 $x(n)$ 长度约为 $h(n)$ 长度的 8 倍时,分段求和 FFT 算法的运算量就已经明显小于直接 FFT 算法的运算量。

表 2 N 远大于 M 时运算量的比较

序列 $h(n)$ 长度 M	序列 $x(n)$ 长度 N	直接 FFT 算法的运算量/乘法次数	分段求和 FFT 算法的运算量/乘法次数	R_2
2	7	32	32	1.000
2	15	80	62	0.775
2	31	192	124	0.646
2	63	448	248	0.554
2	127	1024	496	0.484
2	255	2304	992	0.431
2	511	5120	1986	0.388

3 结语

本文重点研究了如何利用 FFT 快速计算两个有限长序列的线性相关,介绍了直接 FFT 算法及其运算量的改进,以及当两个序列长度相差较大时分段求和 FFT 算法及其运算量的改进,并综合比较了两种算法的运算量。结果表明,相比于根据定义直接计算线性相关,直接 FFT 算法显著减少了运算量,且序列长度越长,改善效果越明显;若参与线性相关的两个序列长度相差较大,则相比于直接 FFT 算法,分段求和 FFT 算法具有更小的运算量,且序列长度差距越大,改善效果越好。

参考文献:

[1] 吴镇扬.数字信号处理[M].第 3 版.北京:高等教育出版社,2016.

[2] 管致中,夏恭恪,孟桥.信号与线性系统[M].第 6 版.北京:高等教育出版社,2015.

[3] 虞湘宾,毕光国.长序列信号快速相关及卷积的算法研究[J].广州:电路与系统学报,2001,6(4):78-83.

[4] 王华奎.数字信号处理及应用[M].北京:高等教育出版社,2009.

[5] 陈楚,吕石磊,徐梅宣,代芬.基于 FFT 算法的长序列线性卷积及相关函数实现[J].上海:电子技术,2018,47(04):22-25.