

基于 LMS 算法的多点滑动 DFT 方法

李春宇¹, 张晓林²

(1. 北京电子技术应用研究所, 北京 100091; 2. 北京航空航天大学电子信息工程学院, 北京 100191)

摘 要: 根据自适应 LMS 法, LMS 谱分析器可以通过递归运算完成滑动窗口中数据的 DFT 运算. 本文推导了 LMS 算法及多点滑动 DFT 运算之间的关系式, 并由此提出了一种基于 LMS 算法的多点滑动 DFT 运算方法. 文章在理论推导的同时, 进行了计算机仿真验证. 该方法使用方便, 可灵活适用于不同的滑动窗口大小及滑动步长参数.

关键词: DFT; 最小均方算法; 滑动窗

中图分类号: TN911.72 **文献标识码:** A **文章编号:** 0372-2112 (2010) 10-2422-04

Multisample-Based Sliding-DFT Method Based on LMS Algorithm

LI Chun-yu¹, ZHANG Xiao-lin²

(1. Beijing Institute of Electronics Technology Application, Beijing 100091, China;

2. School of Electronic Engineering, Beijing University of Aeronautics and Astronautics, Beijing 100191, China)

Abstract: Based on the adaptive least mean squares algorithm, the LMS spectrum analyzer can be applied to recursively compute the discrete Fourier transform of a sliding window of data. In this paper, a LMS-based method for calculating multisample-based sliding-DFT is proposed. The method establishes the relationship between LMS algorithm and sliding-DFT when new multisamples come into the sliding window for DFT. The inferential formula is deduced, meanwhile, the computer simulation is performed for validation. The method is effective and available for arbitrary transform length and sliding step size.

Key words: DFT; LMS algorithm; sliding window

1 引言

在谱分析等数字信号处理技术应用中, 使用滑动离散傅立叶变换 (Discrete Fourier Transform, DFT) 算法可以大幅度节约计算量. 根据滑动 DFT 算法计算两个连续时刻的信号谱时, 若已知前一时刻的信号谱, 则可通过简单的递推运算得到后一时刻的信号谱, 这种方法在连续的实时谱分析中具有重要的意义^[1]. 滑动 DFT 是一种反复递推的运算, 数字信号处理是通过硬件处理器来实现的, 这就限制了可表示的数值是有限字长的, 所以无法用无限精度来表示一个数. 因此, 前一个点的值并不完全精确, 故存在着舍入误差, 而这种舍入误差会随着递推次数的增加而不衰减的传播, 且随着时间逐步积累. Widrow 等人推导了最小均方 (Least Mean Square, LMS) 自适应算法与 DFT 运算之间的关系式, 并提出了基于自适应 LMS 算法, 应用 LMS 谱分析器来计算 DFT 的方法^[2]. 因为采用了自适应技术, 所以应用该方法进行滑动 DFT 计算时可以有效避免舍入误差的传播^[3]. 在 Widrow 等人工作的基础上, 研究者们提出了许多新的数字信号处理方法, 如离散哈特列/沃尔什/余弦/正弦变换 (Discrete Hartley/Walsh/Cosine/Sine Transform,

DHT/DWT/DCT/DST)^[4~8]. 在文献 [2] 中介绍了滑动 DFT 运算时滑动步长是 1 个点, 即逐点滑动的情形. 然而, 在一些实际应用中, 滑动窗的滑动步长是多点而非单点. 因此, 本文针对该种情形, 建立了 LMS 算法及多点滑动 DFT 运算之间的联系, 并由此提出了一种基于 LMS 算法的多点滑动 DFT 运算方法.

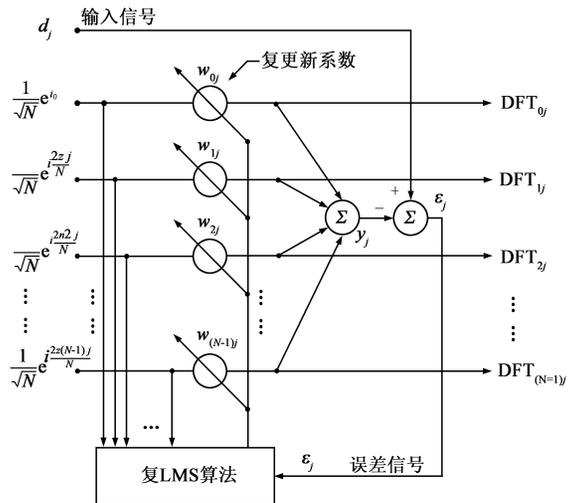


图1 LMS 谱分析器结构

2 LMS 谱分析器

基于 LMS 的正交谱分析器的结构如图 1 所示. 欲进行傅立叶变换的信号 d_j 为线性自适应滤波器的输出期望值, 滤波器在时刻 j 的输入复数向量 \mathbf{X}_j 可以表示为:

$$\mathbf{X}_j = \frac{1}{\sqrt{N}} [1 \quad e^{i\frac{2\pi j}{N}} \quad e^{i\frac{2\pi 2j}{N}} \cdots e^{i\frac{2\pi(N-1)j}{N}}]^T \quad (1)$$

其中 $i = \sqrt{-1}$, N 表示 DFT 的点数(即滑动窗长度), T 表示转置. \mathbf{X}_j 构成 N 维空间的规范正交基.

滤波器权向量随 LMS 算法做自适应调整, 系数更新公式为:

$$\mathbf{W}_{j+1} = \mathbf{W}_j + 2\mu\epsilon_j\bar{\mathbf{X}}_j \quad (2)$$

式中 $\bar{\mathbf{X}}_j$ 为 \mathbf{X}_j 的共轭. Widrow 等人证明了如果 μ 取 1/2, 从初始状态开始迭代 j 次, 则有^[2,3]:

$$\mathbf{W}_j = \sum_{n=j-N}^{j-1} d_n \bar{\mathbf{X}}_n \quad (3)$$

同样, 对于采样点 d_{j-N}, \dots, d_{j-1} 的 DFT 可以表示为:

$$\text{DFT}_j = \sum_{n=j-N}^{j-1} d_n \bar{\mathbf{X}}_{n-j} \quad (4)$$

所以联立式(3)、(4), 可建立 LMS 与 DFT 之间的关系式为^[2]:

$$\text{DFT}_{j-1} = \mathbf{B}_j \mathbf{W}_j = \frac{1}{\sqrt{N}} \begin{bmatrix} 1 & & & & \\ & e^{i\frac{2\pi j}{N}} & & & \\ & & e^{i\frac{2\pi(2)j}{N}} & & \\ & & & \ddots & \\ & & & & e^{i\frac{2\pi(N-1)j}{N}} \end{bmatrix} \mathbf{W}_j \quad (5)$$

LMS 谱分析器就可以根据式(5)所表示的 LMS 算法与 DFT 之间的关系完成 DFT 运算. 即在 j 时刻, 对角阵 \mathbf{B}_j 与系数向量 \mathbf{W}_j 相乘, 可完成 $j-1$ 时刻输入向量的 DFT 运算.

3 基于 LMS 算法的多点滑动 DFT 方法

依据文献[2]的思路, 本文提出了基于 LMS 算法的多点滑动 DFT 方法, 下面对该方法进行详细的推导.

假设滑动窗步长为 p (这里 $0 < p < N$), 如图 2 所示, 前一滑动窗中的输入信号的采样序列为 $d_{j-N+1} \cdots d_j$; 后一窗中的采样序列为 $d_{j+p-N+1} \cdots d_{j+p}$. 当采样序列点 $d_{j+1} \cdots d_{j+p}$ 准备好后, 滑动窗滑动 p 点, 同时滑动窗舍弃采样序列点 $d_{j-N+1} \cdots d_{j+p-N+1}$, 新的采样序列点 $d_{j+1} \cdots d_{j+p}$ 进入滑动窗口.

下面, 通过建立 DFT_{j+p} 与 \mathbf{W}_j 之间的联系, 将多点滑动 DFT 运算与 LMS 算法结合在一起. 由式(5)可得:

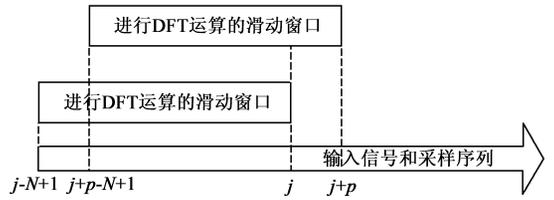


图2 滑动步长为 p 时的滑动 DFT 示意图

$$\text{DFT}_{j+p-1} = \mathbf{B}_{j+p} \mathbf{W}_{j+p}$$

$$= \frac{1}{\sqrt{N}} \begin{bmatrix} 1 & & & & \\ & e^{i\frac{2\pi(j+p)}{N}} & & & \\ & & e^{i\frac{2\pi(2)(j+p)}{N}} & & \\ & & & \ddots & \\ & & & & e^{i\frac{2\pi(N-1)(j+p)}{N}} \end{bmatrix} \mathbf{W}_{j+p}$$

$$= \frac{1}{N} \begin{bmatrix} \sum_{n=0}^{N-1} d_{n+j+p-N} \\ \sum_{n=0}^{N-1} d_{n+j+p-N} e^{-i\frac{2\pi n}{N}} \\ \vdots \\ \sum_{n=0}^{N-1} d_{n+j+p-N} e^{-i\frac{2\pi n(N-1)}{N}} \end{bmatrix} \quad (6)$$

同时, DFT_{j+p-1} 也可以表示为:

$$\text{DFT}_{j+p-1} = \mathbf{B}_{j+p} [\mathbf{W}_j + \mathbf{Q}_j]$$

$$= \frac{1}{\sqrt{N}} \begin{bmatrix} 1 & & & & \\ & e^{i\frac{2\pi(j+p)}{N}} & & & \\ & & e^{i\frac{2\pi(2)(j+p)}{N}} & & \\ & & & \ddots & \\ & & & & e^{i\frac{2\pi(N-1)(j+p)}{N}} \end{bmatrix}$$

$$\left(\frac{1}{\sqrt{N}} \begin{bmatrix} \sum_{n=0}^{N-1} d_{n+j-N} \\ \sum_{n=0}^{N-1} d_{n+j-N} e^{-i\frac{2\pi(n+j-N)}{N}} \\ \vdots \\ \sum_{n=0}^{N-1} d_{n+j-N} e^{-i\frac{2\pi n(n+j-N)(N-1)}{N}} \end{bmatrix} + \frac{1}{\sqrt{N}} \begin{bmatrix} \sum_{n=j}^{j+p-1} (d_n - d_{n-N}) \\ \sum_{n=j}^{j+p-1} (d_n - d_{n-N}) e^{-i\frac{2\pi n}{N}} \\ \vdots \\ \sum_{n=j}^{j+p-1} (d_n - d_{n-N}) e^{-i\frac{2\pi n(N-1)}{N}} \end{bmatrix} \right) \quad (7)$$

式(7)中

$$\mathbf{Q}_j = \frac{1}{\sqrt{N}} \begin{bmatrix} \sum_{n=j}^{j+p-1} (d_n - d_{n-N}) \\ \sum_{n=j}^{j+p-1} (d_n - d_{n-N}) e^{-i\frac{2\pi n}{N}} \\ \vdots \\ \sum_{n=j}^{j+p-1} (d_n - d_{n-N}) e^{-i\frac{2\pi n(N-1)}{N}} \end{bmatrix} = \begin{bmatrix} q_{0j} \\ q_{1j} \\ \vdots \\ q_{(N-1)j} \end{bmatrix} \quad (8)$$

联立式(6)、(7)可得:

$$\text{DFT}_{j+p-1} = \mathbf{B}_{j+p} \mathbf{W}_{j+p} = \mathbf{B}_{j+p} (\mathbf{W}_j + \mathbf{Q}_j) \quad (9)$$

根据 LMS 算法系数更新式(2)及(9),并与文献[2~8]同样取 μ 为 1/2,则:

$$\mathbf{Z}_j = \begin{bmatrix} z_{0j} \\ z_{1j} \\ \vdots \\ z_{(N-1)j} \end{bmatrix} = \sqrt{N} \begin{bmatrix} q_{0j} + \frac{1}{N} \sum_{n=0}^{N-1} w_{nj} e^{\frac{2\pi nj}{N}} \\ q_{1j} e^{i\frac{2\pi j}{N}} + \frac{1}{N} \sum_{n=0}^{N-1} w_{nj} e^{\frac{2\pi nj}{N}} \\ \vdots \\ q_{(N-1)j} e^{i\frac{2\pi(N-1)j}{N}} + \frac{1}{N} \sum_{n=0}^{N-1} w_{nj} e^{\frac{2\pi nj}{N}} \end{bmatrix} \quad (15)$$

$$\mathbf{W}_{j+p} = \mathbf{W}_j + \mathbf{Q}_j = \mathbf{W}_j + \boldsymbol{\varepsilon}_j \bar{\mathbf{X}}_j \quad (10)$$

式中,

$$\boldsymbol{\varepsilon}_j = \mathbf{Z}_j - y_j \mathbf{I} \quad (11)$$

$$y_j = \mathbf{X}_j^T \mathbf{W}_j \quad (12)$$

式(11)中, \mathbf{I} 为 $N \times N$ 的单位矩阵, $\boldsymbol{\varepsilon}_j$ 和 \mathbf{Z}_j 为对角矩阵. \mathbf{Z}_j 中包含了新的采样序列输入点 $d_{j+1} \cdots d_{j+p}$ 的计算信息.

从式(10)、(11)可以推导得到:

$$\mathbf{Q}_j = \boldsymbol{\varepsilon}_j \bar{\mathbf{X}}_j = (\mathbf{Z}_j - \mathbf{X}_j^T \mathbf{W}_j) \bar{\mathbf{X}}_j \quad (13)$$

整理可得:

$$\mathbf{Z}_j \bar{\mathbf{X}}_j = \mathbf{Q}_j + \mathbf{X}_j^T \mathbf{W}_j \bar{\mathbf{X}}_j \quad (14)$$

两边同时乘以 $\bar{\mathbf{X}}_j^{-1}$, 则:

根据式(16),可以得到图 3 所示的多点滑动 DFT 运算器结构,该运算器融入了自适应 LMS 算法.在 $j+p-1$ 时刻,滑动窗口中的输入序列点为 $d_{j+p-N+1} \cdots d_{j+p}$, DFT 的运算结果即为图 3 中的 $\text{DFT}_{(j+p-1)0} \cdots \text{DFT}_{(j+p-1)(N-1)}$.

4 计算机仿真实验

从第 3 部分算法推导描述可知,本文通过建立 LMS 算法与多点滑动 DFT 之间的联系,推导出基于 LMS 算法的多点滑动 DFT 运算关系式,应用本文方法所计算的 DFT 结果与直接对数据序列进行 FFT 计算所得的结果一致.这里用计算机仿真的方法验证以上分析,来说明本文方法的正确性.仿真时,连续输入的 32 点信号序列如图 4 所示.滑动窗从第 1 个序列点向第 32 个序列点方向滑动,滑动窗口大小 $N=8$,滑动步长分别取为 P

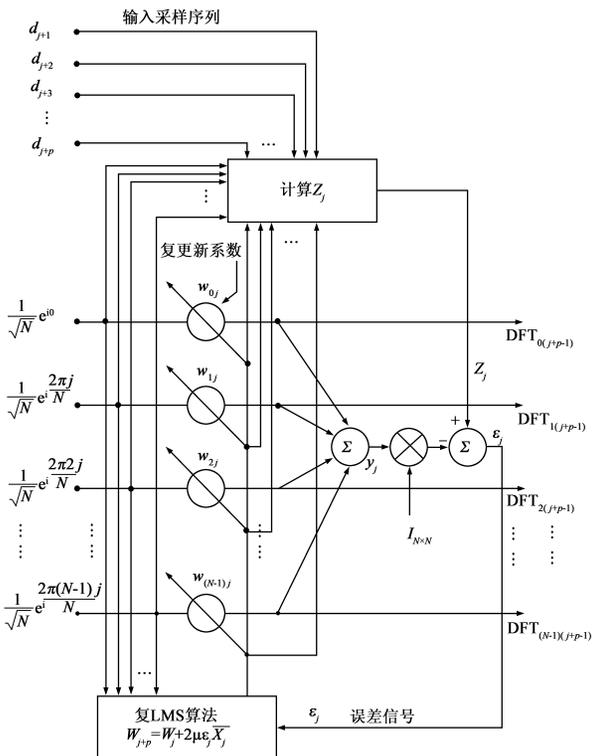


图3 基于LMS算法的多点滑动DFT运算器结构图

这样,根据式(11)便可以计算出 $\boldsymbol{\varepsilon}_j$. 由此,根据式(9)、(10)可以得到多点滑动 DFT 的计算公式:

$$\text{DFT}_{j+p-1} = \mathbf{B}_{j+p} [\mathbf{W}_j + \boldsymbol{\varepsilon}_j \bar{\mathbf{X}}_j], \quad 0 < p < N \quad (16)$$

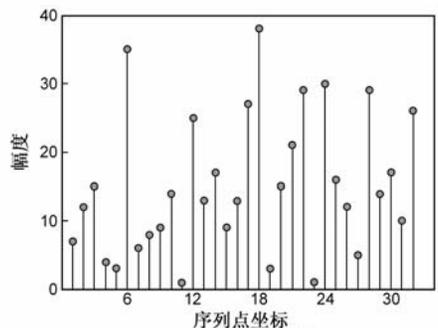


图4 32点输入信号序列

$= 3, P = 4$, 进行两次实验. 当 $P = 3$ 时, 滑动窗经过 7 次滑动, 完成最后 8 点的 DFT 运算; 当 $P = 4$ 时, 滑动窗经过 6 次滑动, 完成最后 8 点的 DFT 运算.

直接通过 FFT 计算得到的最后 8 点输入序列幅度谱序列如图 5(a) 所示, 相位谱序列如图 6(a) 所示. $P = 3$ 时, 通过本文方法计算得到的最后 8 点输入序列幅度谱

序列如图 5(b) 所示, 相位谱序列如图 6(b) 所示. $P = 4$ 时, 通过本文方法计算得到的最后 8 点输入序列幅度谱序列如图 5(c) 所示, 相位谱序列如图 6(c) 所示. 从图 5、图 6 可以看出, 无论是幅度谱还是相位谱, 三种情况下的计算结果完全相同, 从而验证了本文方法的正确性及对于不同参数适用性.

5 结论

本文通过建立自适应 LMS 算法与滑动 DFT 算法之间的联系来完成多个数据点同时进入滑动窗口的多点滑动 DFT 运算. 该方法的提出借鉴了文献[2]的推导思路及相关结论, 所以方法中 LMS 算法的自适应步长与文献[2]选择一致, 即 $\mu = 1/2$. 理论推导及计算机仿真实验均表明, 本文提出的基于 LMS 算法的多点滑动 DFT 方法有效且可适用于不同的滑动窗口大小及滑动步长参数.

参考文献:

- [1] Jacobsen E, Lyons R. The Sliding DFT[J]. IEEE Signal Processing Magazine, 2003, 20(2): 74 – 80.
- [2] Widrow B, Baudrenghien P, et al. Fundamental relations between the LMS algorithm and the DFT[J]. IEEE Transactions on Circuits and Systems, 1987, 34(7): 814 – 820.
- [3] Beaufays F, Widrow B. On the Advantages of the LMS Spectrum Analyzer Over Nonadaptive Implementations of the Sliding-DFT[J]. IEEE Transactions on Circuits and Systems-I: fundamental theory and applications, 1995, 42(4): 218 – 220.
- [4] Liu J C, Lin T P. LMS-Based DHT analyzer[J]. Electronics Letters, 1988, 24(8): 483 – 485.
- [5] Xi J T, Chicharo J F. Computing running discrete hartley trans-

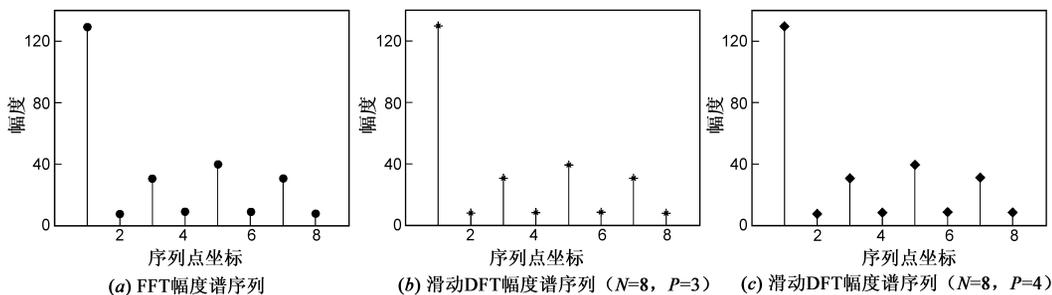


图5 后8点输入信号序列的DFT幅度谱序列

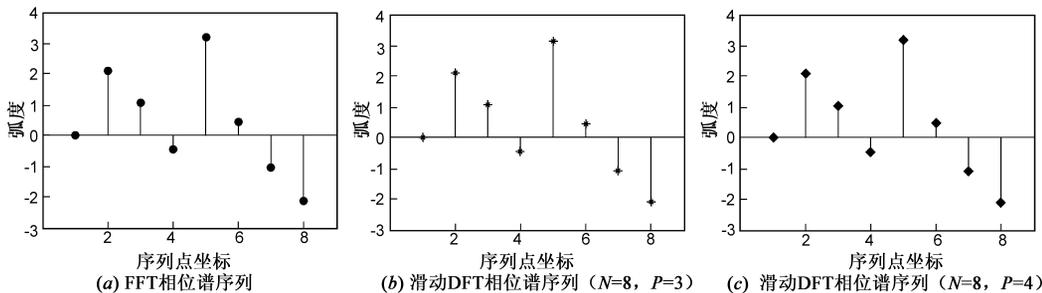


图6 后8点输入信号序列的DFT相位谱序列

form and running discrete W transforms based on the adaptive LMS algorithm[J]. IEEE Transactions on Circuits and Systems-II: analog and digital signal processing, 1997, 44(3): 257 – 260.

- [6] Vasanthan Raghavan, Prabhu K M M, Sommen Piet C W. An analysis of real-Fourier domain-based adaptive algorithms implemented with the Hartley transform using cosine-sine symmetries[J]. IEEE Transactions on signal processing, 2005, 53(2): 622 – 629.
- [7] Xi J T, Chicharo J F. On the relationship between discrete Walsh transform and the adaptive LMS algorithm[J]. IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, 1994, E77-A(7): 1199 – 1201.
- [8] Xi J T, Chicharo J F. Computing running discrete cosine/sine transforms based on the adaptive LMS algorithm[J]. IEEE Transactions on circuits and systems for video technology, 1998, 8(1): 31 – 35.

作者简介:



李春宇 男, 1981年4月生于辽宁鞍山, 博士. 主要研究方向为数字信号处理、卫星导航接收机系统、数字电视系统设计等.
E-mail: lichunyu@ee.buaa.edu.cn

张晓林 男, 1951年1月生于北京, 教授, 博士生导师, 博士. 长期从事卫星导航系统、数字信号处理、集成电路设计、数字电视系统设计的科研与教学工作. E-mail: zxl@buaa.edu.cn