

基于 ADSP - 2106x 的 DFT 改进算法

徐硕¹, 徐习东¹, 刘浏²

(1. 浙江大学电气工程学院, 浙江 杭州 310027; 2 浙江科技学院信息学院, 浙江 杭州 310023)

摘要: 离散傅氏算法 DFT(Discrete Fourier Transform)计算量随着采样频率的增加而增加,给实时在线频谱增加了困难。而递推算法虽然能大幅减少运算量,但会存在累计误差,影响了算法的可靠性。提出了 DFT 的一种改进算法,该算法采用分组求和的方式,能大幅地减少全周傅氏算法运算量,同时不会发生累积误差,最后通过基于 ADSP - 2106x 的编程,对两种算法进行了对比分析。

关键词: ADSP - 2106x; 离散傅里叶变换; 算法; 继电保护

中图分类号: TM74 **文献标识码:** A **文章编号:** 1003-4897(2006)19-0026-03

0 引言

离散傅里叶变换(DFT)是利用采样数据计算信号的频谱的一种数学工具,在生产和工程中有重要的应用。数字式保护是将连续的模拟量经采样和 A/D 转换变换成离散的数字量后,由微机对数字量进行计算和判断。傅氏算法的基本原理是建立一个傅氏数字滤波系统,滤取电流、电压中的基波或谐波分量。由于对继电保护装置有速动性及可靠性要求,数字式保护不仅要实时计算各谐波分量,而且要对各元器件进行自检。随着采样频率增加,采样间隔时间越来越小,而 DFT 运算量却大大增加。如采用递推算法,可大大减少运算量,但一旦发生一次存取错误,该错误将一直保留在结果中,随着保护装置的长期运行,错误将会积累,影响了算法的可靠性。

本文提出了 DFT 的一种改进算法,能在大大减少 DFT 计算量的同时,消除误差累计的影响,提高了可靠性,最后利用 32 位浮点 DSP 实现了该算法,且对该算法进行了评估。

1 算法简介

1.1 传统全周傅氏算法

设 $x(n)$ 为采样序列, $W_N^{kn} = e^{-j\frac{2\pi}{N}nk}$, $x(k)$ 为 $x(n)$ 的离散傅里叶变换, N 为一周期内采样点数,全周傅氏算法的公式为式 (1)^[1]

$$x(k) = \text{DFT}[x(n)] = \sum_{n=0}^{N-1} x(n)W_N^{kn} \quad (1)$$

其中正弦、余弦分量为式 (2)、(3):

$$X_r(k) = \frac{2}{N} \sum_{n=0}^{N-1} x(n) \cos(n\frac{2\pi}{N}) \quad (2)$$

$$X_i(k) = \frac{2}{N} \sum_{n=0}^{N-1} x(n) \sin(n\frac{2\pi}{N}) \quad (3)$$

将傅氏算法分解成余弦和正弦分量之后,如果要实时对电力系统基波或者谐波进行处理,在一个采样周期内就应该进行一次 DFT,如图 1。那么在一个采样周期内,要计算出某次谐波的正弦和余弦分量,需要进行 $2N$ 次实数乘法和 $(2N - 2)$ 次实数加法。由于 ADSP2106X 能进行并行加乘运算,且具有桶形寻址能力,因此能非常简洁地实现式 (2)、(3) 算法,且占用内存空间少。但缺点在于:每个采样周期内,都要进行 $2N$ 次加乘运算,这要占用很多时间。以基波为例,假设采样点数为一个周波 64 点,则一次 DFT 滤过基波要进行 128 次浮点乘法法和 126 次浮点加法运算。

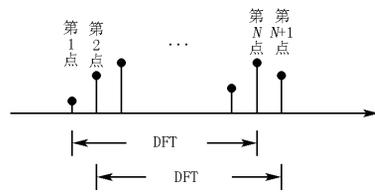


图 1 DFT 数据窗

Fig 1 The window of DFT

1.2 递推算法

根据 DFT 的循环移位(圆周移位)定理^[1],可得递推公式

$$X_m(k) = X_{m-1}(k) + [x(m) - x(m-n)]W_N^{kn} \quad (4)$$

根据式 (4) 算法大大减少了计算量,在一个采样周期内,进行一次 DFT,计算出正弦余弦量只要 2 次浮点乘法法和 4 次浮点加法运算,因此计算速度快,而且占用内存量也很少。但是,它的不足之处是会带来一定的不可靠性。如果在存取数据过程中出现

错误,比如存取值 X_m 与实际值 X_m 相比出现了一个误差,即 $X_m = X_m + X$;那么运用递推算法进行 DFT,这个积累误差 X 将会一直存在于计算结果中,长期运行,随着存取错误的积累,将导致误差越来越大,影响了算法的可靠性。为此也可以采用定期自检的方法,比如定期利用公式(2)、(3)对 DFT 计算结果进行对比或替代等方式,以判断其结果是否正确,而这又增加了运算量和复杂性。

1.3 改进 DFT算法

为了达到快速、可靠的目的,可以对式(2)、(3)的计算做一些改进。根据循环移位定理,如果将每次采样点乘以正余弦常数后的滤波值,例如式(2)中的 $\frac{2}{N}x(n)\cos(n\frac{2}{N})$ 存储下来,并且将新采样点的滤波值代替上一个周波采样点对应的滤波值,也就是将公式(2)中的 $x(n-N)\cos(n\frac{2}{N})$ 换成 $x(n)\cos(n\frac{2}{N})$,然后对一个周波的滤波值进行累加,得到最终结果。采用这种方法可以使得一次 DFT 的计算量减少为 2 次浮点乘法和 $(2N-2)$ 次浮点加法。运算量有一定程度减少,但是考虑到仍然存在 $(2N-2)$ 次加法,实际运算量还是非常可观的。

为了进一步地节省计算时间,可以对上述算法进一步的改进。令式(2)中 $\frac{2}{N}x(n)\cos(n\frac{2}{N})$ 为 $real(n)$, $\frac{2}{N}x(n)\sin(n\frac{2}{N})$ 为 $imag(n)$,该算法将 $real(n)$ 、 $imag(n)$ 分成若干小组,每小组的和值构成一个新的数组,每次采样周期内只需计算当前采样点小组的和值以及重新计算总和即可,这样可以有效地减少加法的次数。例如将 $real(n)$ 分为若干小组: $real(1)$, $real(2)$, ..., $real(m)$; $real(m+1)$, $real(m+2)$, ..., $real(2m)$; ...; $real(n-m)$, $real(n-m+1)$, ..., $real(n)$ 。令 $s(1) = real(1) + real(2) + \dots + real(m)$; $s(2) = real(m+1) + \dots + real(2m)$; ...; $s(k) = real(n-m+1) + \dots + real(n)$;在内存中存入一组 s 值,每个新采样周期只需重新计算每个小组的 DFT 值 $s(x)$,然后再将所有 s 值重加,便可得到新的 DFT 值 $X_r(k)$ 。这种方法随着采样频率的增加,计算量将显著减少。以一个周波 64 个采样点为例。我们可以将 64 个数据分成 8 小组,每组 8 个数据,同时每次采样周期用 $x(n)$ 替换成 $x(n-64)$ 之后,将小组内的数据重新计算,然后每组数据的总和得出最终 DFT 值,这样每个采样周期 DFT 的运算量为

2 次浮点乘法和 28 次浮点加法,相比传统全周傅氏算法而言大大节省了运算量,提高了 DFT 的运算速度。而相比递推算法而言,虽然增加了运算量,但不会存在长期的累积误差。

2 基于 32 位浮点 ADSP - 2106x 的算法实现

ADSP2106x 运算单元包括并行工作的一个加法器、乘法器和移位逻辑电路,一个指令周期内,能进行一次 32 位浮点的加乘运算。以下分别为实现式(2)、(3)的传统全周傅氏算法和改进傅氏算法的程序。

2.1 传统全周傅氏算法

设采样频率为 3 200 Hz, $\frac{2}{N}\sin(n\frac{2}{N})$, $\frac{2}{N}\cos(n\frac{2}{N})$ ($n=0, 1, \dots, 63$) 这两组常数分别存在 $_{\sin64}$ [64]、 $_{\cos64}$ [64] 数组中,采样点保存在 $_{\text{Sample}}$ [64] 数组中,通过以下程序可计算出正弦分量和余弦分量。

```
B9 = _sin64; //get the address of sin64
m9 = 1;
B1 = _cos64; //get the address of cos64
m1 = 1;
b0 = _Sample;
m0 = 1;
F7 = PM (B,M9);
LCNTR = R4, DO NNER_1 UNTL LCE;
F14 = F0 * F7, F10 = F10 + F13, F5 = DM (I,
M1);
NNER_1: F13 = F0 * F5, F9 = F9 - F14, F0 = DM (I,
M0), F7 = PM (B,M9);
```

此算法程序简洁明了,设 ADSP2106x 的一个指令周期为 $1t$,通过 ADSP environment emulate 仿真可测得上述计算需要 $200t$,其中运行循环体需要 $192t$,若模拟量为 n 路时, DFT 计算就至少需要 $192 * n$ 个机器周期。

2.2 改进傅氏算法

改进型 DFT 算法程序框图如图 2。

在改进算法中,要保存临时中间计算结果将 $sample[n] * \frac{2}{N}\cos(n\frac{2}{N})$, $sample[n] * \frac{2}{N}\sin(n\frac{2}{N})$ 结果分别保存在 $_{\text{real}}$ [64]、 $_{\text{imag}}$ [64] 数组中,对 $_{\text{real}}$ 、 $_{\text{imag}}$ 数组进行分组,各分成 8 组,每组之和保存在 $_{\text{real_sum}}$ [8]、 $_{\text{imag_sum}}$ [8] 数组中,则可如下编制程序。

程序如下:

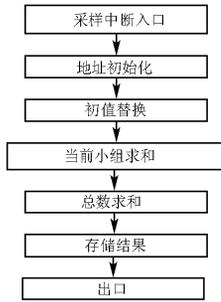


图 2 程序流程图

Fig 2 Program flow chart

//初始化

```

b1 = _real; // _real, _imag 起始地址
b2 = _real_sum; // _real_sum, _imag_sum 起始地址
b3 = _real_sum_end;
i1 = 0;
i2 = 0;
i3 = 0;
i4 = _sample;
i5 = _cos64;
R1 = DM(_seq);
R7 = 64; // 采样数
m4 = r7;
r0 = 8; // 组数. 每组个数
m0 = 8;
M1 = R1;
M2 = R7;
// 替换
modify(i5, m1);
F2 = DM(i4, m5);
F4 = DM(i5, m5);
F8 = F2 * F4, F5 = DM(m2, i5);
DM(m1, i1) = F8;
F8 = F2 * F5, modify(i1, m1);
DM(m2, i1) = F8;
// 计算一组合值
R6 = LSH IFT R1 BY - 3;
m3 = R6;
R9 = R6 * R0(ssi), r3 = b1;
R3 = R9 + R3;
B1 = R3;
F9 = F9 - F9, F3 = DM(m4, i1);
F8 = F8 - F8, F2 = DM(i1, m6);
LCNTR = r0, DO sum1 until LCE;
F9 = F3 + F9, F3 = DM(m4, i1);
sum1: F8 = F2 + F8, F2 = DM(i1, m6);
DM(m3, i2) = F8;
modify(i2, 8);
DM(m3, i2) = F9;
  
```

//计算最终值

```

modify(i2, - 8);
F9 = F9 - F9, f3 = DM(m0, i2);
f8 = f8 - f8, F2 = DM(i2, m6);
LCNTR = r0, DO sum3 until LCE;
F9 = F9 + F3, F3 = DM(m0, i2);
sum3: F8 = F2 + F8, F2 = DM(i2, m6);
DM(B, m6) = F8;
// 顺序 ++
rts(DB);
R1 = R1 + 1, DM(B, m5) = F9;
dm(_seq) = R1;
  
```

通过 ADSP environment emulate 的硬件仿真, 该程序每计算一个相量只需要 $70t$, 程序虽不如传统全周傅氏算法的程序简洁, 但却大幅地提高了运算速度。

3 结束语

本文在传统全周傅氏算法和递推算法的基础上, 给出了一种改进的 DFT 算法, 在大幅提高传统全周傅氏算法计算速度的基础上, 又不存在累积误差, 提高了算法的可靠性, 适用于数字式继电器。若以 ADSP2106x 为核心, 一个 DSP 指令周期为 $1t$, 假设采样频率为 3200 Hz , 即每周波采样点数为 64 点。采用非递推全周傅氏算法, 每计算一个相量至少需要 $192t$, 而采用改进型的傅氏算法, 虽然程序不如全周傅氏算法简洁, 但每计算一个相量只需要 $70t$, 可节约时间 63.5% 。如果采用其他无加乘指令的 CPU, 节省时间将更加可观。

参考文献:

- [1] 丁玉美, 高西全, 彭学愚. 数字信号处理 [M]. 西安: 西安电子科技大学出版社, 1994.
DING Yu-mei, GAO Xi-quan, PENG Xue-yu. Digital Signal Processing [M]. Xi'an: Xidian University Press, 1994.
- [2] 胡广书. 数字信号处理——理论、算法与实现 [M]. 北京: 清华大学出版社, 1997.
HU Guang-shu. Digital Signal Processing—Theory, Algorithm and Implementation [M]. Beijing: Tsinghua University Press, 1997.
- [3] 孙仲康. 快速傅里叶变化及其应用 [M]. 北京: 人民邮电出版社, 1988.
SUN Zhong-kang. Variation and Application of FFT [M]. Beijing: People's Posts & Telecom Press, 1988.

(下转第 41 页 continued on page 41)

[3] 张义辉. 一种符合电力系统动态频率变化的软件测量算法[J]. 重庆科技学院学报(自然科学版), 2005, 7(2): 73-75.

ZHANG Yi-hui The Algorithm for Dynamical Frequency Based on Software of Power System [J]. Journal of Chongqing University of Science and Technology (Sci & Tec), 2005, 7(2): 73-75.

[4] Moore P J, Carranza R D, Johns A T. A New Numeric Technique for High-speed Evaluation of Power System

Frequency[J]. IEE Proc on Gener, Transm and Distrib, 1994.

收稿日期: 2006-06-09; 修回日期: 2006-08-02

作者简介:

孙莉(1969-),女,教师,主要从事计算机及自控专业设计。E-mail: bhcpkfb@sina.com

An improved method of frequency measurement for power system

SUN Li

(Information Security Department, Henan College of Public Security, Zhengzhou 450002, China)

Abstract: Frequency is a very important characteristic quantity in power systems, and an important criterion for relay protection, measurement and control, etc. So Frequency measuring plays a very important role in power system measuring equipments. Discrete Fourier algorithm is a common software frequency measurement in power system, this paper describes a new improved method which uses Piecewise Cubic Hermite Interpolating Polynomial to enhance the effect of Discrete Fourier algorithm. A useful method for frequency measuring is proposed, which has the advantages of high precision, small computation and a good real time performance. By numerical simulation, this paper gives the data windows of time response, the absolute error and the effects of harmonics. The results prove that the method has some practical value.

Key words: frequency; real-time measurement; Fourier algorithm; piecewise cubic hermite interpolating polynomial

(上接第 28 页 continued from page 28)

[4] IEEE Tutorial Course Computer Relaying[Z]. IEEE Inc, 1979.

[5] ADSP-2106x SHARC User's Manual[Z].

徐硕(1981-),男,硕士研究生,主要从事电力系统继电保护方面的研究工作; E-mail: coppola@163.com

徐习东(1966-),男,副教授,主要从事电力系统继电保护方面的研究工作;

刘浏(1977-),女,主要从事电子技术的教学与实验工作。

收稿日期: 2006-04-18; 修回日期: 2006-05-08

作者简介:

An improved DFT algorithm based on ADSP - 2106x

XU Shuo¹, XU Xi-dong¹, LU Liu²

(1. School of Electrical Engineering, Zhejiang University, Hangzhou 310027, China;

2. School of Information Engineering, Zhejiang College of Science and Technology, Hangzhou 310023, China)

Abstract: The calculation of full-wave DFT(Discrete Fourier Transform) algorithm increases with the rising of the sampling frequency, but it is not very suitable for the real-time processing circumstance. The recursive of DFT algorithm reduce the calculation dramatically, but once storage errors occur, an accumulated error will be made which will affect the reliability of the algorithm. This paper presents a new improved algorithm of DFT which use the method of group calculation. This algorithm can reduce the calculation of non-recursive DFT obviously and increase the reliability. ADSP-2106x-based programs are proposed to compare the improved algorithm with the non-recursive algorithm.

Key words: ADSP-2106x; DFT; algorithm; relay protection