

# The Algorithm Analysis and Computer Implementation of Digital Filter

Liu Yuying

College of Communication and Electronics  
Jiangxi Science & Technology Normal University  
Nanchang, China  
e-mail: 502625690@qq.com

**Abstract**—As information science, super-large-scale integrated circuit and computer science develop rapidly, digital signal processing (DSP) has become a new and independent discipline since FFT algorithm was put forward in 1965. It permeates almost the all fields of engineering, science and technology. Signal filter is an important part of the application of DSP, so the design of the filter has been the basic content of digital signal processing. Owing to the advances in digital computer technology and software development, this discipline has made great progress and development, and the MATLAB provides a computational simulation platform for it. In this paper, the computability algorithm and transfer function algorithm of digital filter equation are analyzed, and are verified and determined through the MATLAB.

**Keywords**—digital filter; algorithm; MATLAB; simulation; computability; transfer function

## I. INTRODUCTION

Before the implementation of digital filter in the form of hardware, the algorithm must be implemented in software to verify the effectiveness of the algorithm we choose. Firstly, the time domain equations of digital filter should be arranged in a proper order to ensure its computability. Secondly, symbols analysis method or the computer simulation can be used to determine the transfer function. Finally, the structure of filter should be simulated and verified, and make sure that there is no miscalculation or other errors in the process of implementation, while guaranteeing that the transfer function can describe the filter accurately.

## II. THE COMPUTABILITY ALGORITHM ANALYSIS OF DIGITAL FILTER

### A. The Description of Digital Filter Matrix

In time domain, digital filter can be described in a set of input sequence and output sequence relation equation or a set of input sequence, output sequence and intermediate variables relation equation.

Figure 1 shows a direct II type IIR digital filter. Except the input and output nodes, the rest are all decelerator's input or output nodes, and state variable method can be used to list the state and output equation of this filter[2][3].

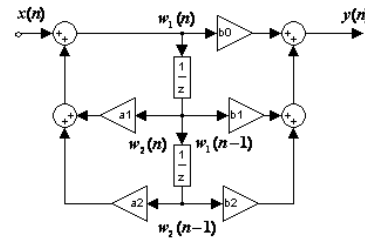


Figure 1. the direct II type IIR digital filter

$$\begin{cases} w_1(n) = a_1 w_1(n-1) + a_2 w_2(n-1) + x(n) \\ w_2(n) = w_1(n-1) \\ y(n) = (b_0 a_1 + b_1) w_1(n-1) + (a_2 b_0 + b_2) w_2(n-1) + b_0 x(n) \end{cases} \quad (1)$$

Before calculating the following variables, the variable in the left of each equation can be calculated directly without the results of equations at the back. Only set one node variable for decelerator input or output nodes, it can make the numbers of equations least, but its computer implementation is not easy. In figure 2, it shows a cascade type digital filter, the label order of its node variables is shown in the figure.

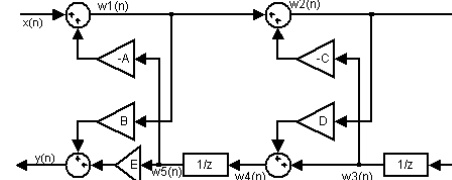


Figure 2. cascade type digital filter

Follows are the time domain equations:

$$\begin{cases} w_1(n) = x(n) - A w_5(n) \\ w_2(n) = w_1(n) - C w_3(n) \\ w_3(n) = w_2(n-1) \\ w_4(n) = w_3(n) + D w_3(n) \\ w_5(n) = w_4(n-1) \\ y(n) = B w_1(n) + E w_5(n) \end{cases} \quad (2)$$

Obviously we cannot solve the equation set by given sequence, so this equation set cannot describe an effective calculation method. We call the equation set of this sequence is incalculable.

Reorder:

$$\begin{cases} w_3(n) = w_2(n-1) \\ w_5(n) = w_4(n-1) \\ w_1(n) = x(n) - Aw_5(n) \\ w_2(n) = w_1(n) - Cw_3(n) \\ w_4(n) = w_3(n) + Dw_5(n) \\ y(n) = Bw_1(n) + Ew_5(n) \end{cases} \quad (3)$$

To solve the equations in this order, the result of variable of every left equation can be directly obtained instead of knowing the result of equations behind.

Equation set (2) can be expressed in matrix form:

$$Y[n] = X[n] + FY[n] + GY[n-1] \quad (4)$$

Among,

$$Y[n] = \begin{bmatrix} w_1(n) \\ w_2(n) \\ w_3(n) \\ w_4(n) \\ w_5(n) \\ y(n) \end{bmatrix} \quad X[n] = \begin{bmatrix} x(n) \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (5)$$

$$F = \begin{bmatrix} 0 & 0 & 0 & 0 & -A & 0 \\ 1 & 0 & -C & 0 & 0 & 0 \\ 0 & D & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ B & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, X[n] = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (6)$$

Equation set (3) can be expressed in matrix form:

$$\begin{bmatrix} w_3(n) \\ w_5(n) \\ w_1(n) \\ w_2(n) \\ w_4(n) \\ y(n) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ x(n) \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -A & 0 & 0 & 0 & 0 \\ -C & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & D & 0 & 0 \\ 0 & E & B & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} w_3(n) \\ w_5(n) \\ w_1(n) \\ w_2(n) \\ w_4(n) \\ y(n) \end{bmatrix} \quad (7)$$

$$+ \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} w_3(n-1) \\ w_5(n-1) \\ w_1(n-1) \\ w_2(n-1) \\ w_4(n-1) \\ y(n-1) \end{bmatrix}$$

After the matrix analyzed, to make the equation set calculated, the diagonal elements and all elements above diagonal of matrix F should be zero.

### B. Prior figure of computability test of digital filter equation

Figure 3 is the signal flow graph of cascade lattice digital filter. For the input signal can be sure, the output signal of the decelerator is the input signal of the moment before; the simplified flow diagram as shown in figure 4 can be obtained after removing the branches from input nodes and delay branches[1].

Divide the simplified signal flow diagram into groups; In the first group  $\{N_1\} = \{w_3(n), w_5(n)\}$ , there are only the node set without output branches; In the second group  $\{N_2\} = \{w_1(n)\}$ , the nodes have input branches which come from  $\{N_1\}$  and they have output branches; In the third group  $\{N_3\} = \{w_2(n)\}$ , the nodes have input branches which come from  $\{N_1\}$  and  $\{N_2\}$  and output branches; In the fourth group  $\{N_4\} = \{w_4(n), y(n)\}$ , the nodes have input branches which come from  $\{N_1\}$ ,  $\{N_2\}$  and  $\{N_3\}$ , but they have not output branches. Calculate the signal variables of every set in order from  $\{N_1\}$ ,  $\{N_2\}$ ,  $\{N_3\}$  to  $\{N_4\}$ , which ensures the computability of the algorithm. If there is no  $\{N_4\}$  that contains only the set of input branches, the signal flow graph is incalculable. The prior figure is as shown in figure 5 after the nodes are divided into groups again.

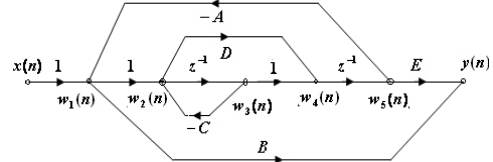


Figure 3. Signal flow diagram

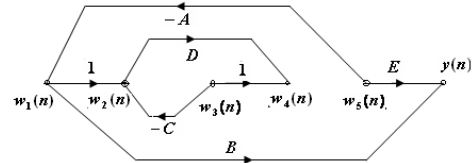


Figure 4. Simplified flow diagram

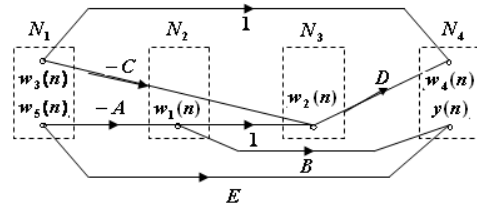


Figure 5. Precedence Diagram

### C. Based on Digital filter can be calculated by computer test and computability equation algorithm

Using professor Ssanjit k.Mitra modification can be tested in MATLAB's M file P\_G\_A with D delay timer and M multipliers of the computability of digital filter, and can get the computability equation algorithm. P\_G\_A (delay, mult) function, input delay time delay vector is a  $2 \times D$  matrix, each column of the two elements, respectively, for

the time delay of input node number and output node number, input vector mult multiplier is a  $3 \times M$  matrix, each column of the three elements respectively for the input node number, output node number of multiplier and coefficient. If the structure of the digital filter can be calculated, the equations are given before and after the reordering of F and G number order of matrix and can calculate equation; If the structure of the digital filter is incalculable, give an error message "STRUCTUR NOT COMPUTABLE".

% figure 2 levels of computability LianGe type digital filter validation and computability equation algorithm

```
A=3;B=2;
C=4;D=5;
E=1;
delay=[2 3;4 5];
mult=[1 6 B;1 2 1;5 1 -A;3 2 -C;2 4 D;3 4 1;5 6 E];
[F,G,NN,F_new,G_new]=P_G_A(delay,mult)
computational results:
```

```
F =
    0    0    0    0   -3    0
    1    0   -4    0    0    0
    0    0    0    0    0    0
    0    5    1    0    0    0
    0    0    0    0    0    0
    2    0    0    0    1    0

G =
    0    0    0    0    0    0
    0    0    0    0    0    0
    0    1    0    0    0    0
    0    0    0    0    0    0
    0    0    0    1    0    0
    0    0    0    0    0    0

NN = 3    5    1    2    4    6
F_new =
    0    0    0    0    0    0
    0    0    0    0    0    0
    0   -3    0    0    0    0
   -4    0    1    0    0    0
    1    0    0    5    0    0
    0    1    2    0    0    0

G_new =
    0    0    0    1    0    0
    0    0    0    0    1    0
    0    0    0    0    0    0
    0    0    0    0    0    0
    0    0    0    0    0    0
    0    0    0    0    0    0
```

### III. THE REALIZATION OF DIGITAL FILTER TRANSFER FUNCTION

#### A. Symbolic analysis algorithm based on MATLAB

When the digital filter structure can be calculated, can choose any order equations expressed in matrix form of  $z$  domain, type(2).

$$T \cdot Y = X \quad (8)$$

Among them, Relationship with the system impulse response samples

$$T = \begin{bmatrix} 1 & 0 & 0 & 0 & A & 0 \\ -1 & 1 & C & 0 & 0 & 0 \\ 0 & -z^{-1} & 1 & 0 & 0 & 0 \\ 0 & -D & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & -z^{-1} & 1 & 0 \\ -B & 0 & 0 & 0 & -E & 1 \end{bmatrix}, \quad Y = \begin{bmatrix} W_1(z) \\ W_2(z) \\ W_3(z) \\ W_4(z) \\ W_5(z) \\ Y(z) \end{bmatrix}, \quad X = \begin{bmatrix} X(z) \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (9)$$

The transfer function for

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\Delta T_{61}}{\Delta T} \quad (10)$$

We can use MATLAB M file sym\_di\_f function transfer function:

%Symbolic analysis method to determine the digital filter transfer function

```
A=3;B=2;C=4;D=5;E=1;
delay=[2 3;4 5];
mult=[1 6 B;1 2 1;5 1 -A;3 2 -C;2 4 D;3 4 1;5 6 E];
in_node=1;out_node=6;
[num,den]=sym_di_f(delay,mult,in_node,out_node)
The results:
Num =
    2.000    13.000    1.000
Den =
    1.000    19.000    3.000
```

Therefore, the transfer function of this filter is

$$H(z) = \frac{2 + 13z^{-1} + z^{-2}}{1 + 19z^{-1} + 3z^{-2}} \quad (11)$$

#### B. Reconstruction algorithm based on MATLAB impulse function

##### 1) Relationship with the system impulse response samples

For N-order causal FIR filter, The first N non-zero values of the unit impulse response are its transfer function coefficients. For N-order causal IIR filter, the unit impulse response is infinitely long, relationship with the transfer function coefficients is much more complicated[6].

Transfer function of describing N-order causal IIR filter structure:

$$H(z) = \frac{B(z)}{A(z)} = \frac{b_0 + b_1 z^{-1} + \dots + b_N z^{-N}}{1 + a_1 z^{-1} + \dots + a_N z^{-N}} \quad (12)$$

$$= \sum_{n=0}^{\infty} h(n) z^{-n}$$

So

$$B(z) = A(z) \cdot H(z) \quad (13)$$

According to the time domain convolution theorem

$$b_n = a_n * h(n) \quad (14)$$

Among

$$b_n = \{b_0 \quad b_1 \quad \dots \quad b_N\}, a_n = \{1 \quad a_1 \quad \dots \quad a_N\} \quad (15)$$

Type (13) describes the relationship between the  $(2N+1)$  coefficients of the numerator and denominator of the transfer function and its impulse response samples.

Application of this relationship, with the former  $(2N+1)$  sample values of  $h(n)$ , the transfer function of any discrete-time structure can be determined[1][2].

2)Reconstruction algorithm of the impulse function of the transfer function and structure verification.

Using the MATLAB simulink to simulate 2-cascaded lattice digital filter shown in Figure 2 , get Figure 6 in the screenshot[4][5].Among,A=3,B=2,C=4,D=5,E=1,N=2.

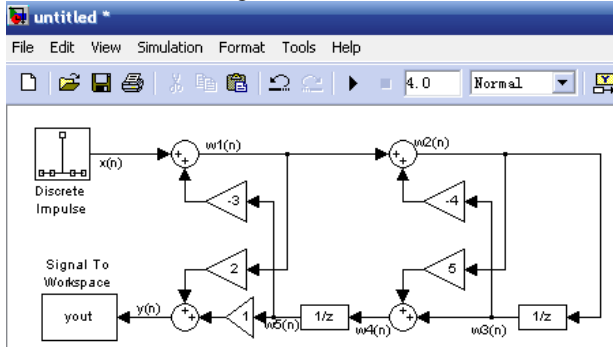



Figure 6. Figure 2 The simulation diagram of cascade type digital filter

In the workspace:

 yout [2;-25;470;-8855;1.6684e+005]

%By using $(2N+1)$  samples in front of  $h(n)$  to determine the simulation filter transfer function

`h=[2 -25 470 -8855 166835]`

`[Num,Den]=strucver(h,2)`

The results:

Num =2 13 1

Den =1 19 3

The transfer function of the filter

$$H(z) = \frac{2 + 13z^{-1} + z^{-2}}{1 + 19z^{-1} + 3z^{-2}} \quad (16)$$

Calculate the first 5 samples of impulse response with the MATLAB simulation digital filter , we can call M-file who is called strucver, to validate the correctness of the software simulation algorithm

```
num=[2 13 1];
den=[1 19 3];
x=impseq(0,4,0);
h=filter(num,den,x)
[num,den]=strucver(h,2)
```

The results:

```
h =
    2   -25   470  -8855  166835
num =
    2   13    1
den =
    1   19    3
```

#### IV. CONCLUSIONS

This article discusses the conditions of the filter algorithm implementation. The filter algorithm is independent of the type of realized digital filter. The research object of this paper is a digital filter which can be used in the MATLAB simulation and can be realized by a set of equations who is used to describe the algorithm in order. In the paper, the condition of equations be calculated is analyzed emphatically, which is used to describe the structure of the filter. The paper also gives the computability of the algorithm validation method and order of Calculable equations. On the basis of these, We Summed up two algorithms that can be used to solve the digital filter transfer function based on a group of describing the structure of the digital filter equation: Semiotic analysis and Impulse function reconstruction method, Finally it put forward a technology ,it can be used to verify system structure from the angle of the input sample to the output sample[1].

#### REFERENCES

- [1] Sanjit K.Mitra.Digital Signal Processing A Computer-Based Approach(3rd edition).Beijing:Publishing House of Electronics Industry,2006,pp.478-531.
- [2] Cheng Peiqing. Digital Signal Processing Tutorial (3rd edition).Beijing: Tsinghua University Press,2007,pp.196-220.
- [3] Wang Fengwen,Shu Dongmei,Zhao Hongcai. Digital Signal Processing(2nd edition).Beijing: Beijing University of Posts and Telecommunications Press,2007,pp.181-189.
- [4] Tang Xianghong,Yue Hengli,Zheng xuefeng.MATLAB and Application in the Electronic Information Class Course(2nd edition).Beijing:Publishing House of Electronics Industry,2009,pp.79-299.
- [5] He Chaoying.MATLAB Application and Experiment Tutorial. Beijing: Publishing House of Electronics Industry,2010,pp.83 -207.
- [6] Chen houjin,Xue jian,Hu jian. Digital Signal Processing.Beijing: Higher Educational Press,2004,pp.305-322.