

## **EFFICIENT DIGITAL FILTERS WITH VARIABLE CUT-OFF BASED ON ALL-PASS AND SHARPENING DIGITAL FILTERS**

**Maja M. Lutovac<sup>1</sup>, Vlastimir D. Pavlović<sup>2</sup>, Miroslav D. Lutovac<sup>3</sup>**

<sup>1</sup>Lola Institute, Belgrade, Serbia

<sup>2</sup>University of Niš, Faculty of Electronic Engineering, Niš, Serbia

<sup>3</sup>University Singidunum, Belgrade, Serbia

**Abstract.** All-pass digital filters are known as excellent prototype filters in many applications, such as control systems, signal processing and communications. Efficient filters can be designed because transfer function is very robust to small element changes. The drawback of digital filters based on all-pass filter sections is that they are not suitable for tuning and the design of filters with variable cut-off frequency. In this paper there is proposed a new design method suitable for the design of filters that may satisfy different attenuation specifications and variable cut-off based on low-order wave digital filters. The method is derived using computer algebra system.

**Key words:** approximation, variable cut-off frequency, all-pass digital filters

### 1. INTRODUCTION

All-pass and wave digital filters (WDFs) are derived from real loss-less reference analog filters and, if properly designed, behave completely like passive circuits, [1-4]. The efficiency of wave digital filters is presented in a number of papers [5], [6], and more details on the design procedure can be found in [7]. Some modern communication devices require changing the filter specification in real time applications, such as to change the maximum pass-band and minimum stop-band attenuation, or the pass-band and stop-band edge frequencies [8], [9].

There is a variety of design approaches to solve the design problem of efficient digital filters that have variable attenuations and edge frequencies. Some solutions are based on connection of several low-order filter sections such as in [9]. The cascade connection of low-order filters can be used to obtain changes of stop-band attenuation with vary small pass-band ripple [10]. The edge frequencies can be changed using Constantinides frequency transformations [11]. The direct usage of frequency transformations is not applicable in practical implementations because the loop without delay elements appears. After

---

Received November 14, 2012

**Corresponding author:** Maja M. Lutovac

Lola Institute, Belgrade, Serbia • E-mail: mlutovac@singidunum.ac.rs

\* **Acknowledgement:** The work presented here was supported by the Serbian Ministry of Education and Science (Projects TR32032).

rearranging the transfer function, a new transfer function can be derived, but the transformed transfer function is not appropriate for efficient implementations. The digital filters with algebraic loops can be implemented using iterated algorithm presented in [12].

The basic idea of this paper is to combine the design of digital filters with programmable stop-band attenuation using low-order wave digital filters. The method is based on [10-14] but with wave digital filters instead of transposed direct form second-order sections. Next, each wave digital filter section of lower order is implemented using iterated algorithm in order to provide variable cut-off suitable for efficient implementation. Both methods are based on efficient start-up wave digital filters that are optimized for multiplier less implementation, and the appropriate stop-band attenuation or variable edge frequencies can be obtained without redesign.

## 2. SHARPENING METHOD

The design of filters based on sharpening method is based on wave digital filters whose approximation is a type of the elliptic filters with minimal Q factors (EMQF) and one compensating section that is based on the design of Jacobi elliptic approximation with two different pass-band attenuations

$$H(z) = (H_{EMQF}(z))^{m-1} H_C(z). \quad (1)$$

The low-pass EMQF filter can be implemented using two all-pass wave digital filters  $A_0(z)$  and  $A_1(z)$ ,

$$H_{EMQF}(z) = \frac{1}{2} A_0(z) + \frac{1}{2} A_1(z). \quad (2)$$

where for the 5th-order all-pass functions  $A_0(z)$  and  $A_1(z)$  can be presented in the following forms

$$A_0(z) = \frac{\beta_{s1} + \alpha(1+\beta_{s1})z^{-1} + z^{-2}}{1 + \alpha(1+\beta_{s1})z^{-1} + \beta_{s1}z^{-2}}. \quad (3)$$

$$A_1(z) = \frac{\alpha_1 + z^{-1}}{1 + \alpha_1 z^{-1}} \frac{\beta_{s2} + \alpha(1+\beta_{s2})z^{-1} + z^{-2}}{1 + \alpha(1+\beta_{s2})z^{-1} + \beta_{s2}z^{-2}}. \quad (4)$$

The key feature is that the filter order should be odd.

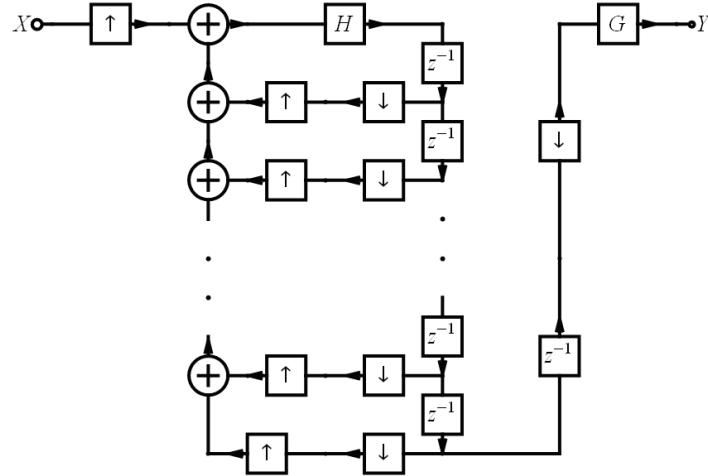
The efficiency of the all-pass sections is in that only two coefficients are required for each second-order section, and that one coefficient ( $\alpha$ ) has the same value in all second-order sections. This property is important for filters with variable cut-off frequency, because small number of coefficients should be computed in all second-order sections when the cut-off frequency is changed.

Figure 1 presents the classic realization of  $(m-1)$  EMQF filters and one compensating filter



**Fig. 1** Cascaded realization of filter based on sharpening method

Hardware implementation of filter using programmable chips can be very efficient when processing speed is several times lower than the operating speed (sample period is several times higher than the period of the clock frequency at which digital components work). The same hardware can be used for processing as the EMQF filter instead of  $(m-1)$  EMQF filters. The basic scheme is shown in Figure 2.



**Fig. 2** Realization of the filter based on sharpening method using multi-rate technique

Instead of  $(m-1)$  EMQF filters only one EMQF filter is used, but each delay element is replaced by  $(m-1)$  delay elements. Input signal is up-sampled  $m$  times, and is filtered using the filter  $H$ . Next, the filtered signal is delayed, down-sampled and up-sampled in such a way that appears at the input of the filter  $H$  after each input sample. The samples of twice filtered signal appear at the input of the filter  $H$  after one-time filtered samples. After  $(m-1)$  filtering the input signal, only those samples that are selected using down-sample element are the signal of sharpening filter at the output of the filter  $G$ .

It should be noted that up-sampling and down-sampling do not require additional hardware. Also, additional adders are actually multiplexers that select which sample of the input channel will appear at the filter  $H$  input.

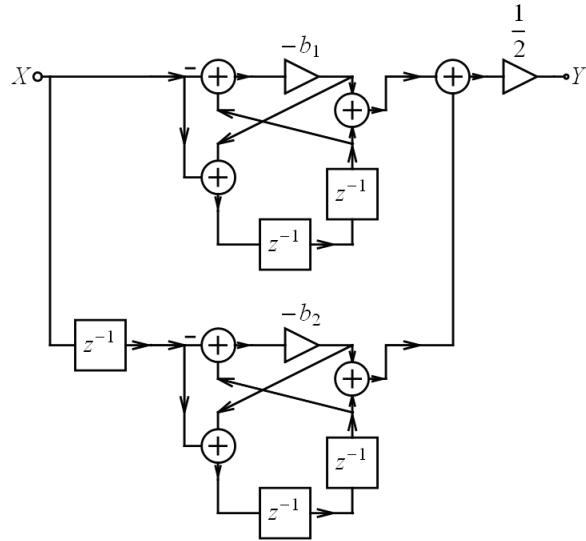
### 3. ALL-PASS DIGITAL FILTER SECTIONS

Figure 3 shows the all-pass realization based on Ansari-Liu all-pass first-order sections [7].

This type of realization is used for variable cut-off filters in [12]. One of the possible problems in fixed-point implementation is coefficient quantization and overflow effects.

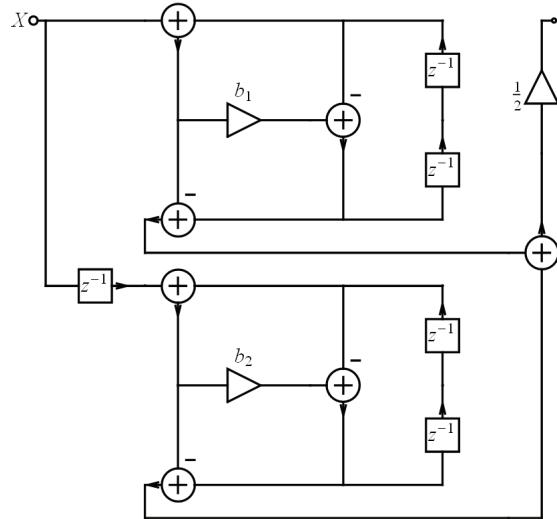
Wave digital filters may be a good solution for solving such problems [7]. The design is based on wave analog filters, and the digital wave filters have some good properties that are described in [7]. Figure 4 shows the same filter realization of EMQF filter [13] as that based on all-pass sections, except the wave digital filter realization is used.

The main difference is that all-pass sections have coefficients  $|b_1| \geq 0.5$  and  $|b_2| \geq 0.5$ , while WDF has  $|b_1| \leq 0.5$  and  $|b_2| \leq 0.5$ .



**Fig. 3** Ansari-Liu all-pass first-order filter with doubled delay used for the realization of the second order half-band filter section

Both filter realizations have the same transfer functions in spite of the fact that coefficients are different. The same transfer function as all-pass filter can be obtained using a different filter structure, and we are using a computer algebra system to prove that.



**Fig. 4** First-order all-pass digital filters with doubled delay used for the realization of the second order half-band filter section

#### 4. ANALYSIS USING COMPUTER ALGEBRA SYSTEM

Computer algebra systems (CAS) that we use are Mathematica and SchematicSolver, in a same way as in [12]. Firstly, we load knowledge of SchematicSolver [15] into Mathematica [16]. Next, we draw both schematic shown in Figures 3 and 4. Both schematic are SISO (single-input-single-output) systems. CAS has a function for computing transfer functions. For example, for all-pole realization we can obtain the transfer function of the overall filter, as shown in Figure 5.

```
H1 =
DiscreteSystemTransferFunction[schematic][[1]][[1]][[1]] // Factor

( (1 + z) (z2 + b1 - z b1 + z2 b1 - z3 b1 + z4 b1 + z b2 -
z2 b2 + z3 b2 + z2 b1 b2) ) / (2 z (z2 + b1) (z2 + b2) )
```

**Fig. 5** Automatic derivation of transfer function

The numeric values can be computed using Jacobi sine function and the elliptic integral [7].

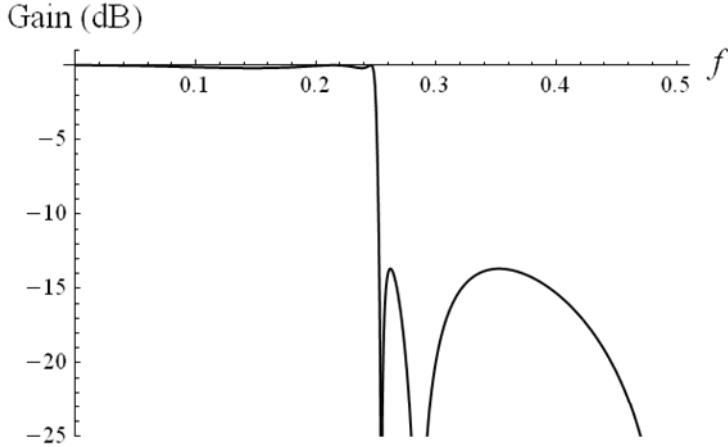
CAS computes numeric values as replacement rules that are used for computing the transfer function of EMQF filter, as shown in Figure 6.

```
Hn = H1 /. {b1 → bHB51, b2 → bHB52} // Together
(0.5 (1. + z) (0.640576 + 0.316526 z +
1.29657 z2 + 0.316526 z3 + 0.640576 z4) ) /
(z (0.640576 + z2) (0.957102 + z2) )

DiscreteSystemDisplayForm[Hn]
(0.320288 + 0.478551 z-1 + 0.806548 z-2 +
0.806548 z-3 + 0.478551 z-4 + 0.320288 z-5) /
(1. + 1.59768 z-2 + 0.613097 z-4)
```

**Fig. 6** Transfer function after replacement filter coefficients with numeric values

The previous function is used to present transfer function as rational function in  $z^{-1}$ . Finally, the amplitude response can be drawn in dB, see Figure 7. The pass-band ripple is small, but the minimum stop-band attenuation is not sufficient to be used for any practical application.



**Fig. 7** Amplitude response of EMQF filter (gain in dB)

The same procedure is repeated for WDF realization. The schematic is drawn using appropriate GUI, next the transfer function is automatically derived from the schematic. It is obvious that this symbolically derived transfer function (H2) is not the same as H1 for all-pass realization, see Figure 8.

```

H2 = DiscreteSystemTransferFunction[schematicWDF][[1]][[[1]][[1]]]
(-1 - z - 2 z2 - 2 z3 - z4 - z5 + b1 + z2 b1 + z3 b1 +
z5 b1 + z b2 + z2 b2 + z3 b2 + z4 b2 - z2 b1 b2 - z3 b1 b2) /
(2 z (1 + z2 - b1) (1 + z2 - b2))

Hn2 = H2 /. {b1 → 1 - bHB51, b2 → 1 - bHB52} // Together
(0.5 (-0.640576 - 0.957102 z - 1.6131 z2 -
1.6131 z3 - 0.957102 z4 - 0.640576 z5) ) /
(z (0.640576 + z2) (0.957102 + z2))

DiscreteSystemDisplayForm[Hn2]
(-0.320288 - 0.478551 z-1 - 0.806548 z-2 -
0.806548 z-3 - 0.478551 z-4 - 0.320288 z-5) /
(1. + 1.59768 z-2 + 0.613097 z-4)

```

**Fig. 8** Transfer function after replacement filter coefficients of the second filter structure

The numeric values of both transfer functions are the same, except that WDF is multiplied by -1 with respect to all-pass realization. By inspection, it is obvious that coefficients of WDF are computed as  $(1-b)$ , where  $b$ -coefficients are coefficients of all-pass filter.

### 5. FILTERS WITH ALGEBRAIC LOOP

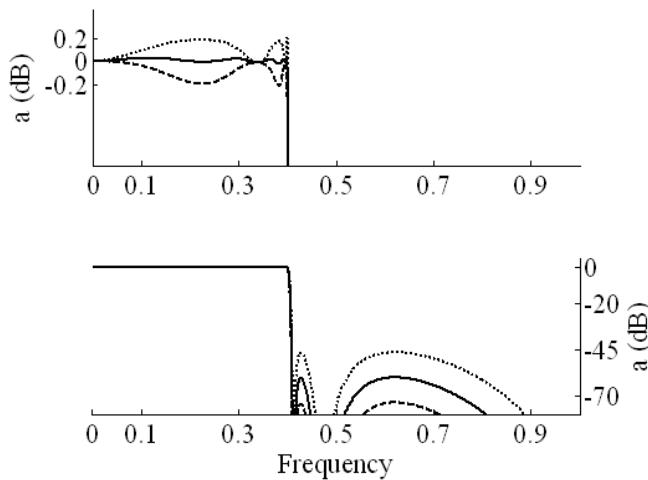
The simplest way to change the cut-off frequency of digital filters is to transform one type of filter into another by Constantindes frequency transformations [11]

$$z^{-1} \rightarrow \frac{a + z^{-1}}{1 + a z^{-1}}, \quad a \neq 0. \quad (5)$$

Suppose that we transform one low-pass filter into another low-pass filter with different pass-band edge frequency. This transformation implies that each delay element is replaced with all-pass filter section. The digital filter can be implemented only if there is no direct loop without delay elements. After replacement of each delay element with first or second order filter section, we can identify the algebraic loop. That means that it is impossible to compute output signal because some inputs of digital filter elements are unknown. In practice, a new transfer function is derived, and new filter coefficients are calculated. After redesign, the digital filter is no more efficient. The problem can be solved using iterated calculation with multiplier-less filters [1].

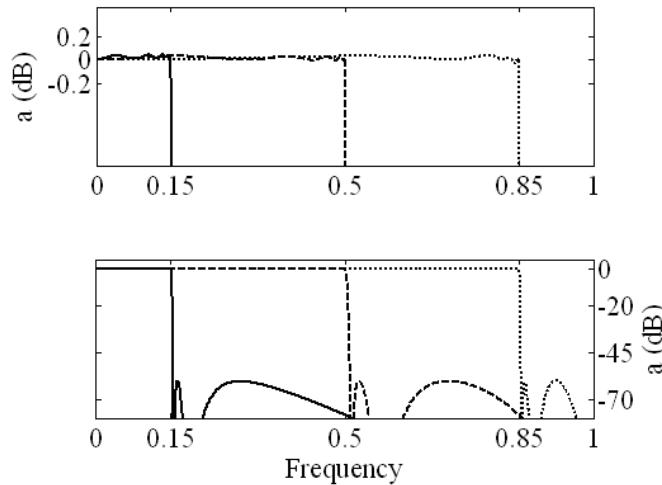
In order to implement a digital filter with the algebraic loop, we propose new computation procedure: we cut the loop and insert the unit delay element. Consequently, during the time between two input samples, we use a higher sampling rate and compute an approximate value at the filter output, similarly to the multi-rate filtering operation. Even more, there are some filter implementations in which the output of the multiplier is computed in two or more steps in order to implement multiplier coefficients with higher accuracy. More detail on the numeric error can be found in [12].

Figure 9 shows how stop-band attenuation can be changed using sharpening method, with the pass-band attenuation less than 0.2 dB. The usage of frequency transformation shows how the edge frequencies can be changed in wider range, with unchanged shape of the attenuation characteristic, as illustrated in Figure 10. The basic second-order section based on wave digital filters is shown in Figure 3.



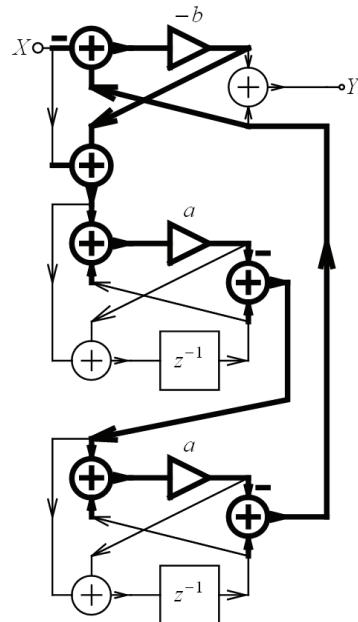
**Fig. 9** Pass-band and stop-band attenuation of filters designed using sharpening method

The implementation code derived in CAS can be used in other software tools, such as Matlab.



**Fig. 10** Filters with variable cut-off frequency

Figure 11 shows the algebraic loop of the second-order section, where delay elements are replaced by first order filter section.



**Fig. 11** Schematics of filter with algebraic loop

## 5. CONCLUSION

Computer algebra systems can be used to derive properties of new systems, to perform thorough analysis, and automate design procedure. This paper presents how several different methods are combined to design a new efficient and robust digital filter. The method combines multi-rate technique, sharpening technique, and the usage of the most efficient elliptic low-order elliptic filters (EMQF).

In this paper there is proposed a new design method suitable for the design of filters that may satisfy different attenuation specifications and variable cut-off frequency based on low-order wave digital filters.

## REFERENCES

- [1] A. Fettweis, "Digital filter structures related to classical networks", *Arch. Elek. übertragung.*, vol. 25, pp.79-89, 1971.
- [2] A. Fettweis, "Digital circuits and systems", *IEEE Trans. on Circuits and Systems*, Vol. CAS-31, pp.321-348, 1974.
- [3] A. Fettweis, H. Levin. and A. Sedlemeyer, "Wave digital lattice filters", *Int. J. Circuit Theory Appl.*, pp. 203-211, 1974.
- [4] L. Gazsi, "Explicit formulas for lattice wave digital filters", *IEEE Trans. Circuits Syst.*, vol. CAS-32, pp. 68-8, 1985.
- [5] M. Lutovac and L. Milic, "Design of computationally efficient elliptic IIR filters with reduced number of shift-and-add operations in multipliers", *IEEE Transactions on Signal Processing*, vol. 45, pp. 2422-2430, 1997.
- [6] M. Lutovac and Lj. Milić, "Lattice wave digital filters with a reduced number of multipliers", *Yugoslav IEEE MTT Chapt. Inf.*, vol. 2, pp. 29-39, 1996.
- [7] M. D. Lutovac, D. V. Tošić, and B. L. Evans, *Filter Design for Signal Processing Using MATLAB and Mathematica*, New Jersey: Prentice Hall, 2000.
- [8] L. Milić and T. Saramäki, "Three classes of IIR complementary filter pairs with an adjustable crossover frequency", Proc. of the 2003 *Int. Symp. on Circuits and Systems*, Bangkok, Thailand, vol. 4, pp. IV-145-IV-148, 2003.
- [9] G. Stoyanov, I. Uzunov, and M. Kawamata, "High tuning accuracy design of variable IIR filters as cascade of identical sub-filters", Proc. of the 2005 *Int. Symp. on Circuits and Systems*, Kobe, Japan, pp. 3729-3732, 2005.
- [10] V. M. Poučki, A. Žemva, M. D. Lutovac, and T. Karčnik, "Elliptic IIR filter sharpening implemented on FPGA", *Digital Signal Processing*, vol. 20, pp. 13–22, 2010.
- [11] G. Constantinides, "Spectral transformations of digital filters", *Proc. Inst. Elec. Eng.*, vol. 117, pp. 270–327, 1970.
- [12] M. Lutovac, J. Čertić, Lj. Milić, "Digital Filter Design Using Computer Algebra Systems", *Circuits Syst. Signal Process.*, vol. 29, pp. 51-64, 2010.
- [13] M. Lutovac and V. D. Pavlović, "Efficient Multirate EMQF Digital Filters", *Electronics and Electrical Engineering*, Kaunas: Technological, vol. 119, no. 3, pp. 39–44, 2012.
- [14] M. Lutovac, *Design of Analog and Digital Filters Using Computer Algebra Systems*, Belgrade: Academic Mind, in Serbian, 2011.
- [15] M. D. Lutovac and D. V. Tošić, *SchematicSolver - Symbolic Signal Processing*, Version 2.22, Belgrade, Serbia, Miroslav Lutovac, 2011.
- [16] S. Wolfram, *The Mathematica Book*, Champaign: Wolfram Media, 2003.