Proceedings of the 7th International Conference on Signal Processing and Machine Learning
DOI: 10.54254/2755-2721/2027.38552

Digital IIR Filter Design Methods and Filter Selection Trade-Offs in Hardware Implementation

Yiyao Hu

Ulink College of Shanghai, Shanghai, China
hushimei123@126.com

Abstract. Digital IIR filters are widely used for signal denoising in signal processing systems. IIR filter implementation requires a lower order with the same design specifications when compared with FIR filters. Among typical IIR categories, Butterworth and Chebyshev filters are two commonly used filters. This paper introduces the fundamental theory of digital IIR filters, reviews two major indirect design approaches including, bilinear transformation and impulse invariance, and explains the different properties of Butterworth, Chebyshev type I and Chebyshev type II filters. Further qualitative comparison of different types of filters is conducted in hardware implementation scenarios. The analysis concludes that Butterworth filters have no ripple in the passband response and superior signal integrity at the cost of higher hardware complexity, whereas Chebyshev filters achieve sharper transition bands with reduced order at the expense of ripples. This work clarifies practical selection principles in scenarios with limited resources and points out that unified quantitative selection metrics for IIR filter hardware deployment remain insufficient for low-cost scenarios.

Keywords: Digital filter, IIR, Filter design, Hardware implementation

1. Introduction

Signal processing is an indispensable technology in modern electronic and communication systems. In real-world engineering processes, raw signal collections are accompanied by random noise, which introduces inaccuracies in subsequent signal analysis. Filtering thus plays an important role in eliminating unwanted noise and interference, improving signal quality. Digital filters offer more flexibility and better control of accuracy, which are more favored than analog filters [1]. IIR (Infinite Impulse Response) filters and FIR (Finite Impulse Response) filters are the two most commonly used digital filters, which are widely used for signal denoising. Compared with FIR, IIR requires a lower order with the same design specifications [2]. This characteristic requires fewer computational resources and higher efficiency, making IIR more suitable for scenarios with limitations on power consumption, operation speed and computational resources. However, the phase response of the IIR filters is nonlinear, so FIR is more suitable if the application requires phase information [3].

Butterworth and Chebyshev filters are two main types of IIR filters because of their reliable filtering performance and adjustable parameters. With their distinct characteristics, they are applicable to different demands. In hardware implementations, the selection and design of IIR filters

© 2026 The Authors. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/).

DOI: 10.54254/2755-2721/2027.38552

face multiple trade-offs. When designing filters, we want to use the lowest amount of hardware resources with acceptable resolutions. Chebyshev filters with lower orders reduce computational resource usage and latency, yet introduce ripples that distort raw signals. By comparison, Butterworth filters usually have higher orders than Chebyshev filters, consuming more hardware resources while improving signal integrity.

This paper systematically explains the basic principles and frequency characteristics of IIR filters, compares the differences between Butterworth and Chebyshev filters, and analyzes the key trade-offs between filtering performance, resource consumption and signal integrity in hardware implementations, aiming to provide a practical design reference for IIR filter deployment.

2. Basics of IIR

2.1. Basic theory of IIR

IIR filters yield an indefinite impulse response. Its output incorporates feedback elements [4]. It has high computational efficiency, which requires far fewer orders than FIR (Finite Impulse Response) to achieve equivalent effects [5]. Types of IIR include Butterworth, Chebyshev type I, Chebyshev type II, and Elliptical. The function of IIR filters is:

$$H(z) = \frac{\sum_{k=0}^{M} b_k z^{-k}}{1+\sum_{k=1}^{N} a_k z^{-k}} \tag{1}$$

The coefficients of b_k and a_k correspond to the forward path and feedback path, respectively.

2.2. Design methods

IIR filter design methods can be divided into two categories: direct and indirect. Direct design methods involve optimizing the coefficients bk and ak of IIR filters directly in the z-domain. Computers then adjust the coefficients iteratively to approximate the desired frequency response. However, this method consumes substantial computational resources and requires long computation time, making it unsuitable for engineering applications [6].

Indirect design methods include the bilinear transformation and impulse-invariance. Table 1 shows a comparison of the two methods, including their advantages, applicable filter types and aliasing problems.

Table 1. Comparison of the two methods

	Bilinear transformation	Impulsive invariance
Advantages	High universality, stable performance	Simple mapping
Applicable filter type	All filter types	Low-pass, band-pass
Aliasing problem	Exists	None

2.2.1. Bilinear transformation

The bilinear transformation is the most common method used for digital filter design. Compared with other design techniques, it has no frequency aliasing. It is defined as:

DOI: 10.54254/2755-2721/2027.38552

$$s = \frac{2}{T} \cdot \frac{1-z^{-1}}{1+z^{-1}} \tag{2}$$

where T is the sampling interval.

It converts an s -domain continuous-time transfer function Ha(s) to a discrete-time z-domain transfer function H(z) [7]. The first step in using the bilinear transformation to design a filter is to give the digital filter specifications: pass-band cut-off frequency ωp, stop-band cut-off frequency ωs, pass-band ripple Rp and stop-band attenuation As. Next, pre-wrap the digital specifications to analog specifications: transform cut-off frequencies ωp and ωs to Ωp and Ωs. Finally, design the prototype analog filter Ha(s), and apply the bilinear transformation to obtain the digital filter H(z) [8].

2.2.2. Impulse invariance

The impulse invariance method is also a popular method to transform analog filters to digital filters. Given an analog filter with impulse response ha(t), its impulse response is denoted by

$$h[n] = T \cdot h_a(nT) \tag{3}$$

where T is the sampling interval.

The procedures for using impulse invariance to design filters are:

1. Design an analog filter transfer function ha(t) with given filter specifications.
2. Decompose the analog transfer function into partial fraction to acquire individual poles Sk and corresponding residues.
3. Map each analog pole to the z-plane by $z_k = e^{s_k T}$, and reconstruct the digital transfer function H(z) from the mapped pole-residue pairs.

Impulse invariance is only suitable for band-limited low-pass and band-pass filters. For high-pass filters, due to the spectral aliasing of impulse invariance, other methods like bilinear transformation are preferred.

3. Butterworth filters and Chebyshev filters

3.1. Butterworth filters

The Butterworth filter is close to an ideal filter because it has a maximally-flat magnitude response in the passband. Its magnitude response is monotonic in both the passband and the stopband. Its squared-magnitude frequency response is:

$$|H(j\omega)|^2 = \frac{1}{1+(\frac{\omega}{\omega_c})^{2n}} \tag{4}$$

where n is the filter order and ωc is the cutoff frequency.

When designing a Butterworth filter, the cut-off frequency ωc and the filter order N determine the filter's performance. We can determine the pass-band cut-off frequency ωp, stop-band cut-off frequency ωs, pass-band ripple Rp and stop-band attenuation As based on the signal's spectrum characteristics. Then we can calculate N and ωc [9].

DOI: 10.54254/2755-2721/2027.38552

3.2. Chebyshev type I and type II filters

Chebyshev type I filters are characterized by a frequency-response magnitude that is equiripple in the passband and monotonic in the stopband. Its squared-magnitude frequency response is:

$$|H(j\omega)|^2 = \frac{1}{1+\varepsilon^2 C_n^2(\frac{\omega}{\omega_p})} \tag{5}$$

where Cn is the n-th order Chebyshev polynomial.

Chebyshev type II filters are characterized by a frequency-response magnitude that is monotonic in the passband and equiripple in the stopband. Its squared-magnitude frequency response is:

$$|H(j\omega)|^2 = \frac{1}{1+\varepsilon^2 C_n^2(\frac{\omega_s}{\omega})} \tag{6}$$

4. Comparison between Butterworth and Chebyshev for hardware applications

Butterworth and Chebyshev filters are widely used for practical signal processing in hardware applications. The inherent differences between the Butterworth and Chebyshev filters lead to distinct applications and trade-offs. The Butterworth filter has a maximally flat magnitude response in the passband with better phase linearity and negligible ripple. These characteristics minimize signal distortion and make the Butterworth filter applicable for scenarios that require high signal integrity. However, it has a slower roll-off, which requires a higher filter order to achieve a satisfactory stopband specification. Increasing filter orders requires higher power, resulting in strict hardware dependency. Thus, Butterworth filters are more applicable to high-performance embedded hardware, where signal integrity is prioritized over resource efficiency.

The Chebyshev filter has a steeper roll-off at a lower order, but it achieves this at the expense of introducing ripples within the passband [10]. This filter greatly reduces overall computational complexity and signal processing latency, making the filter suitable for resource-limited embedded devices.

Overall, the hardware selection criteria for Butterworth and Chebyshev filters cover three aspects: computational capability, hardware costs, and signal distortion. In scenarios with abundant computing resources and a high budget, Butterworth filters are preferred because of their minimal signal distortion and filtering stability. By comparison, Chebyshev filters are suitable for low-budget, lightweight embedded hardware scenarios with limited computational power, where minor signal distortion can be acceptable. Table 2 concludes the comparison of Butterworth and Chebyshev filters.

Table 2. Comparison of Butterworth and Chebyshev Filter

	Butterworth Filter	Chebyshev Filter
Passband Magnitude Response	Maximally-flat magnitude response with negligible ripple	obvious passband ripples
Signal integrity	Higher	Relatively lower, Involves slight signal distortion caused by passband ripples
Roll-off rate	gradual	steep

DOI: 10.54254/2755-2721/2027.38552

Table 2. (continued)

Required filter order	Higher order required to achieve satisfactory stopband specifications	Lower order can achieve ideal stopband filtering performance
Computational complexity	Higher due to higher order required	Lower due to lower order required

Although existing studies have clarified the qualitative trade-off between Butterworth and Chebyshev filters under fixed filter specifications, systematic analysis of the trade-off between ripple tolerance, computational capability, and cost is still inadequate. There is currently a lack of unified quantitative selection criteria of different filters in low-budget embedded hardware scenarios.

5. Conclusion

The paper first summarized the fundamental theories of IIR filters, design methods of digital IIR filters, Butterworth filters, and Chebyshev filters. Compared with direct design methods of IIR filters, indirect methods, including bilinear transformation and impulse invariance, are more applicable in engineering implementation. Bilinear transformation effectively eliminates frequency aliasing, while impulse invariance is only reliable for band-limited low-pass and band-pass filter design.

On this basis, the paper discussed the trade-offs of the IIR filter hardware implementation based on a comparative analysis of the characteristics of the Butterworth and Chebyshev filters qualitatively. By discussing several characteristics of the two filters, including passband magnitude response, signal integrity, roll-off rate, filter order, and computational complexity, the author concludes that three main factors are involved in the selection of the filters. With three main factors: hardware cost, computational capability and signal integrity, the author concludes that Butterworth filters are suitable for high-performance hardware prioritizing signal integrity, and Chebyshev filters are suitable for lightweight hardware with limited computational capabilities where minor signal distortion is acceptable.

However, this study mainly focuses on the qualitative trade-offs of the filters in hardware implementation. Thus, future studies will focus on the quantitative analysis of the trade-offs for a more practical selection.

References

[1] Getu, B. N. (2020). Digital IIR filter design using bilinear transformation in MATLAB. *2020 International Conference on Communications, Computing, Cybersecurity, and Informatics (CCCI)*, 1–6. doi: 10.1109/CCCI49893.2020.9256625.

[2] Sutradhar, S. R., Sayadat, N., Rahman, A., Munira, S., Haque, A. K. M. F., & Sakib, S. N. (2017). IIR based digital filter design and performance analysis. *2017 2nd International Conference on Telecommunication and Networks (TEL-NET)*, 1–6. Noida, India. doi: 10.1109/TEL-NET.2017.8343596.

[3] Lažeta, L., Marković, I., & Šimović, V. (2021). IIR filters designed for comparison and minimum-order design exploration using Matlab. *2021 44th International Convention on Information, Communication and Electronic Technology (MIPRO)*, 875–879. Opatija, Coratia. doi: 10.23919/MIPRO52101.2021.9596760.

[4] Kumar, V., Arya, M., Kumar, A., & Jhariya, D. K. (2024). Design and comparison between IIR Butterworth and Chebyshev digital filters using MATLAB. *2024 Fourth International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT)*, 1–7. Bhilai, India. doi: 10.1109/ICAECT60202.2024.10469026.

DOI: 10.54254/2755-2721/2027.38552

[5] Cao, X. J. (2025). Design and simulation of IIR digital filters. *Mechanical Engineering & Automation, 54*(6), 73–74+77.
[6] Chen, Z. X., & Xiong, Y. X. (2022). Design and implementation of FIR and IIR digital filters based on MATLAB/GUI. *Intelligent Computer and Applications, 12*(1), 80-83.
[7] Al-Alaoui, M. A. (2007). Novel approach to analog-to-digital transforms. *IEEE Transactions on Circuits and Systems I: Regular Papers, 54*(2), 338-350. https://doi.org/10.1109/TCSI.2006.885982
[8] Getu, B. N. (2020). Digital IIR filter design using bilinear transformation in MATLAB. *2020 International Conference on Communications, Computing, Cybersecurity, and Informatics (CCCI)*, 1-6. Sharjah, United Arab Emirates. https://doi.org/10.1109/CCCI49893.2020.9256625
[9] Zhang, X., & Jiang, S. (2021). Application of Fourier transform and Butterworth filter in signal denoising. *2021 6th International Conference on Intelligent Computing and Signal Processing (ICSP)*, 1277-1281. Xi'an, China. https://doi.org/10.1109/ICSP51882.2021.9408933
[10] Wu, Y. (2024). Optimizing IIR filter design using multi-objective genetic algorithm: A focus on passband ripple and stopband attenuation. *2024 8th International Workshop on Control Engineering and Advanced Algorithms (IWCEAA)*, 64-69. Nanjing, China. https://doi.org/10.1109/IWCEAA63616.2024.10823855