Piyanuch TANGKITTIPON[1], Yotaka TUNGTRAGUL[1], Arintorn JESADAMETHAKAJORN[1], Atirarj SUKSAWAD[1]

Faculty of Engineering, Rajamangala University of Technology Isan Khonkaen Campus, Khonkaen 40000, Thailand (1)
ORCID: 1. 0000-0002-0856-5941; 2. 0000-0002-3020-8019; 3. 0009-0006-3160-8299; 4. 0000-0002-1375-9564;

doi:10.15199/48.2024.06.14

Designing a frequency filter with optimal parameters for the Kaiser window using an evolutionary heuristic with a bee colony algorithm

***Abstract**. Digital filters play a crucial role in signal processing. In this paper, the design of frequency filters with optimal parameters using an evolutionary heuristic with a bee colony algorithm will be mentioned. The objective of this study is to present a method for designing frequency filters using an evolutionary heuristic search method called the Bee Colony Algorithm, utilizing the Kaiser window. To find the optimal filter order and compare it with the specified filter sequence. According to simulation data, the proposed Bee Colony Algorithm outperforms the original in terms of main lobe width, side lobe attenuation, and leakage factor. The simulation findings also show a clear reduction in the side lobe and leakage factors, while the main lobe stays unchanged.*

Streszczenie.** Filtry cyfrowe odgrywają kluczową rolę w przetwarzaniu sygnału. W artykule omówione zostanie projektowanie filtrów częstotliwości o optymalnych parametrach z wykorzystaniem heurystyki ewolucyjnej z algorytmem kolonii pszczół. Celem pracy jest przedstawienie metody projektowania filtrów częstotliwości z wykorzystaniem ewolucyjnej metody wyszukiwania heurystycznego zwanej Algorytmem Kolonii Pszczół z wykorzystaniem okna Kaisera. Aby znaleźć optymalną kolejność filtrów i porównać ją z określoną sekwencją filtrów. Według danych symulacyjnych proponowany algorytm kolonii pszczół przewyższa oryginał pod względem szerokości listka głównego, tłumienia listka bocznego i współczynnika wycieku. Wyniki symulacji pokazują również wyraźną redukcję współczynników listka bocznego i wycieku, podczas gdy płat główny pozostaje niezmieniony.* (Projekt filtra częstotliwości o optymalnych parametrach dla okna Kaisera z wykorzystaniem heurystyki ewolucyjnej z algorytmem kolonii pszczół***)

Keywords: evolutionary heuristic, kaiser window, bee colony algorithm
Słowa kluczowe: Heurystyka ewolucyjna, okno Kaisera, algorytm kolonii pszczół

Introduction

The digital filter is the core element of signal processing. The digital filter is employed to benefit signal separation and signal recovery [1-36]. The design process has been presented using window techniques with MATLAB software. A window function is a mathematical function that is used to taper the data, reducing the leakage of energy from the main lobe of the spectrum into the side lobes. [1-18].

Window functions come in various types to choose from, and each type is used to meet the specific requirements of different filters. Popular types of windows used include Gaussian, Tukey [9]., rectangular [11-12]., Hanning [10-12]., Hamming [11-15]., Bartlett [10]., Butterworth., Blackman [13-14]., and Kaiser. The Kaiser window is a type of window function used in digital signal processing, filter design, and Fourier analysis. The Kaiser window is an approximation to the prolate spheroidal window, for which the ratio of the main lobe energy to the sidelobe energy is maximized. The Kaiser window is utilized for processing digital signals, designing filters, and analyzing Fourier transforms. [9-11, 13-18]

The design and development of frequency filter circuits continue to progress continuously. The goal is to improve operational efficiency. The design has been fine-tuned using metaheuristic algorithmic techniques. The most commonly used metaheuristic algorithmic techniques are: genetic algorithm [19-20, 24]., differential evolution [21-23]., particle swarm optimization [23-30, 33]., ant colony algorithm [31-32]., bat algorithm [33-34]., and bee algorithm [35–37].

The experiments show that increasing the beta parameters for each filter order results in an increase in the main lobe, while the side lobes and leakage factors decrease. This leads to unclear signals and interference signals. Therefore, it recommends reducing the width of the main lobe and sidelobes, as well as minimizing leakage factors, by employing metaheuristic techniques to achieve a more efficient filter.

Materials and methods

Kaiser Window

The Kaiser window is a type of window function that is commonly used in signal processing and digital signal processing (DSP). Kaiser windows are often used in applications where a well-controlled frequency response is needed, such as in filter design or when analyzing signals in the frequency domain. The flexibility obtained from the parameter β makes the Kaiser window a versatile option in various signal processing applications. The following is the formula for the Kaiser window:

(1) $$w[n] = I_0\left(\beta\sqrt{1-\left(\frac{n-M}{M}\right)^2}\right)/I_0(\beta)$$

Finding the value of β in the Kaiser window is a tuning process that involves adjustments to achieve the desired characteristics of the window for signal processing applications. This is often done using practical or theoretical methods to select a suitable β value based on the requirements of the task or signal analysis.

(2) $$\beta = \begin{cases} 0.1102(A_{sb}-8.7) & for\ A_s > 50 \\ 0.5842(A_{sb}-21)^{0.4} + 0.07886(A_{sb}-21) & for\ 21 \le A_s \le 50 \\ 0 & for\ A_s < 21. \end{cases}$$

Determining the filter order in the Kaiser window is a crucial step in designing a digital filter or other filters. The calculation of the filter order often involves specifying the frequency response characteristics desired for the filter.

(3) $$N = \frac{2.056A_{sb}-16.4}{2.285(\Delta\omega)}$$

Bee colony Algorithm

The Bee Colony Algorithm (BCA) is a computational algorithm that simulates the foraging behavior of bees in nature and is applied to solve practical problems in mathematical sciences and engineering. BCA Inspired by the natural behavior of bees searching for food, the algorithm involves sending scout bees to various locations and tracking the best results. This process helps discover potential positions with the lowest or highest values according to specified conditions. The algorithm is particularly effective in solving complex optimization problems by mimicking the foraging behavior of bees. The algorithm for bee colonies includes the subsequent steps:

Step 1: As stated in Table 1, define the variables in the Bee Algorithm.

Step 2: Randomly sample a population of scout bees (n) to search for the initial answer and set the number of iterations to NC = 0.

Step 3: Evaluate the results obtained from the scout bee search, arranging them in ascending order. This will make it easier to choose a good answer.

Step 4: Select the answer with the best performance (m).

Step 5: Separate the answer (m) into two groups, where the first group contains the best solution (e), and the second group contains the solutions ranked below the best (m-e).

Step 6: Define the scope for searching neighboring answers around the region of the best answer (m) by determining the scope of the initial search for sources of nectar at each source (n_{gh}).

Step 7: Have scout bees (n_{ep}) search around for answers (e), and worker bees (n_{sp}) search around for answers (m-e). Evaluate the results obtained from the search for worker bees at each source and select the best answers from each source.

Step 8: Check the stopping conditions. If the conditions are met, stop searching. If the condition is not met, increase the number of iterations by setting NC=NC+1. Subject to the condition of the specified maximum number of iterations.

Step 9: Designate scout bees (n-m). Find new answers, then return to step 2 and repeat the process until the best solution is obtained.

Table 1. The parameters of the bee colony algorithm

variable	Description	Parameter value
n	The number of scout bees.	50
m	The number of nectar sources that have quantities of nectar from the search of scout bees (n).	25
e	The number of nectar sources with the highest quantities of nectar from the selected (m).	10
n_{ep}	The number of worker bees assigned to search at the nectar source (e).	20
n_{sp}	The number of worker bees assigned to search at the nectar (m-e).	30
n_{gh}	The size of the search space for each nectar source.	0.5
NC	The number of iterations.	15

simulation result

From gathering information and conducting experiments on the design of a digital filter using Kaiser windows, it is evident that increasing the beta parameter results in an increase in the main lobe while decreasing the side lobe and leakage factors. However, the increased values of the filter sequence result in the main lobe, side lobe, and leakage factors decreasing. This indicates that increasing beta parameters and adding filter order values affect the width of the main lobe and the attenuation of the side lobes.

The frequency filter design proposes an evolutionary heuristic search method called the Bee Colony Algorithm using the Kaiser window. This aims to reduce the main lobe width, suppress side lobe levels, and mitigate leakage factors.

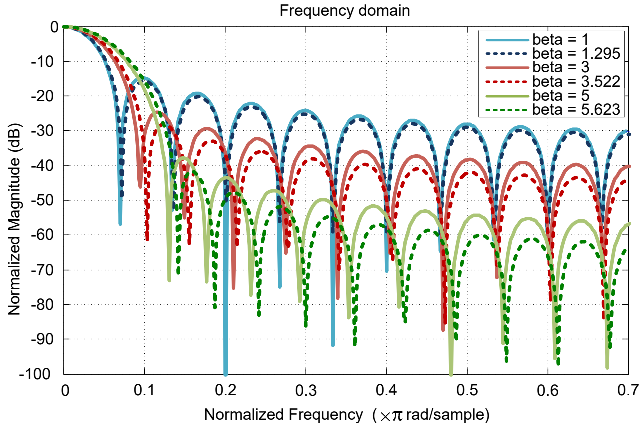

Fig. 1. Frequency domain with a filter order of 30.

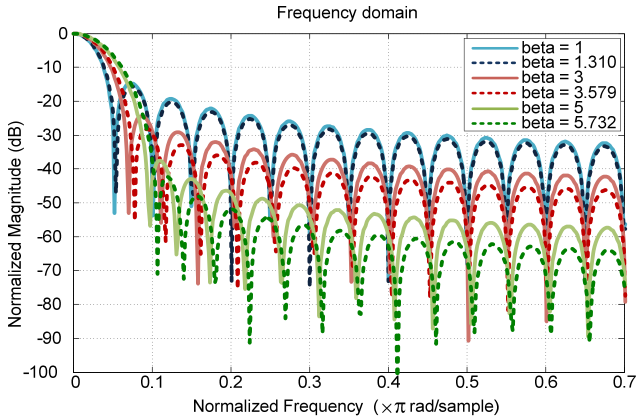

Fig. 2. Frequency domain with a filter order of 40.

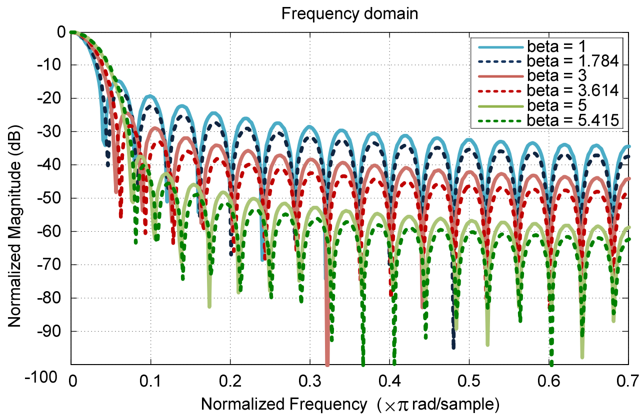

Fig.3. Frequency domain with a filter order of 50.

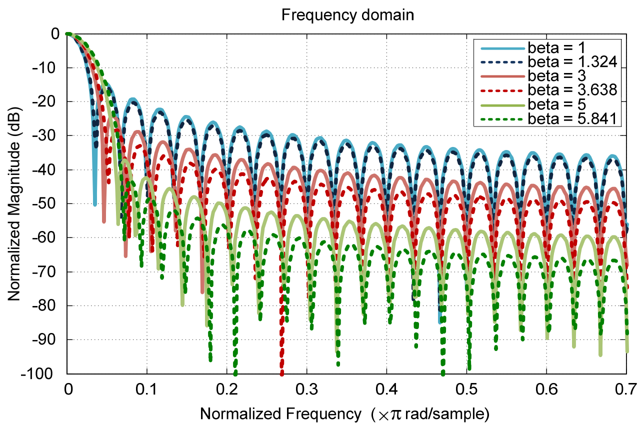

Fig.4. Frequency domain with a filter order of 60.

Table 2 shows the Kaiser window functions with specified beta parameters and the optimal beta parameter using the bee colony algorithm.

Length of Window function	Variable parameter beta	Time Domain		Frequency Domain		
		Max. Amp	Min. Amp	Mainlobe width (-3dB)	Relative side lobe Attenuation	Leakage Factor
30	1	1	0.78	0.054688	-14.7 dB	6.33 %
	1.295		0.68		-15.8 dB	4.96 %
	3		0.20	0.070313	-24.7 dB	0.49 %
	3.522		0.13		-28.1 dB	0.20 %
	5		0.03	0.085938	-37.8 dB	0.01 %
	5.623		0.02		-41.7 dB	0 %
40	1	1	0.78	0.042969	-14.7 dB	6.51 %
	1.310		0.67		-15.7 dB	5.06 %
	3		0.20	0.054688	-24.5 dB	0.54 %
	3.579		0.12		-28.2 dB	0.20 %
	5		0.03	0.066406	-37.6 dB	0.02 %
	5.732		0.01		-42.3 dB	0 %
50	1	1	0.78	0.035156	-14.7 dB	6.47 %
	1.784		0.50		-17.7 dB	3.10 %
	3		0.20	0.042969	-24.3 dB	0.56 %
	3.614		0.12		-28.2 dB	0.20 %
	5		0.03	0.050781	-37.4 dB	0.02 %
	5.415		0.02		-40.1 dB	0.01 %
60	1	1	0.78	0.027344	-14.7 dB	6.44 %
	1.324		0.67		-15.8 dB	4.97 %
	3		0.20	0.035156	-24.2 dB	0.57 %
	3.638		0.12		-28.3 dB	0.20 %
	5		0.03	0.042969	-37.3 dB	0.02 %
	5.841		0.01		-42.9 dB	0 %

Figs.1-4. show the frequency domain using Kaiser window functions, and Table 2 illustrates a comparison between the specified beta parameters and the optimal beta parameters using the bee colony algorithm. Including the width of the main lobe, reduction of the side lobe, and leakage factors. It is evident that the optimal beta parameters using the bee colony algorithm are the main lobe being constant while the side lobe and leakage factors decreasing ostensibly.

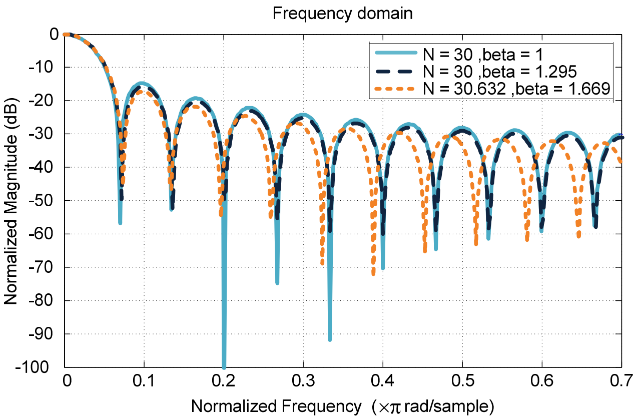

Fig.5. Frequency domain with a filter order of 30, beta is 1, 1.295, and the filter order of 30.632, beta is 1.669.

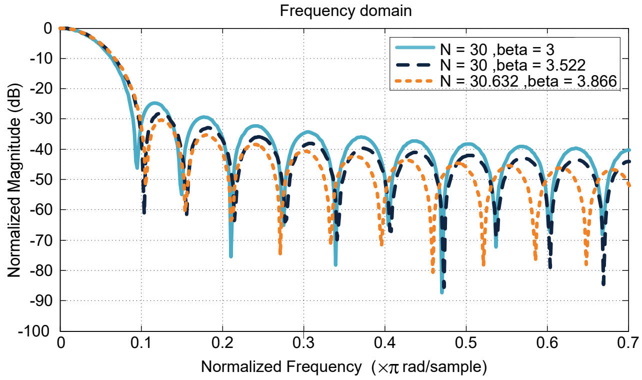

Fig.6. Frequency domain with a filter order of 30, beta is 3, 3.522, and the filter order of 30.632, beta is 3.866.

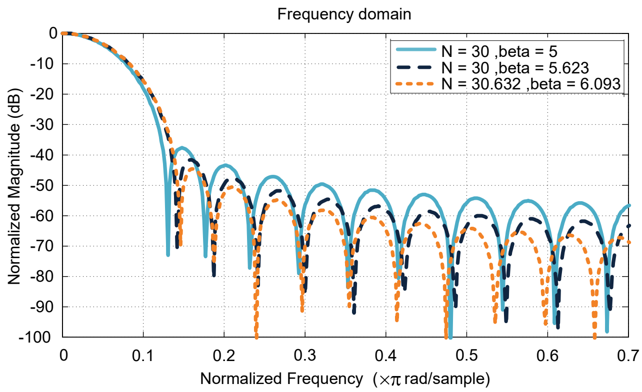

Fig.7. Frequency domain with a filter order of 30, beta is 5, 5.623, and the filter order of 30.632, beta is 6.093.

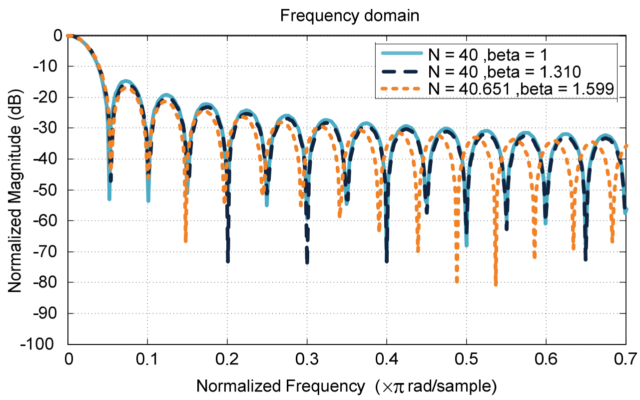

Fig.8. Frequency domain with a filter order of 40, beta is 1, 1.310, and the filter order of 30.632, beta is 1.599.

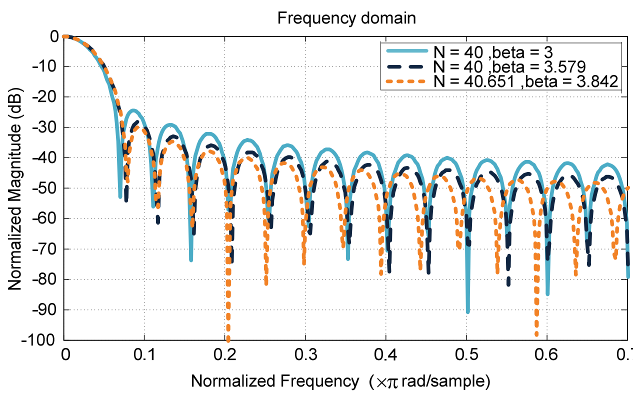

Fig.9. Frequency domain with a filter order of 40, beta is 3, 3.579, and the filter order of 30.632, beta is 3.842.

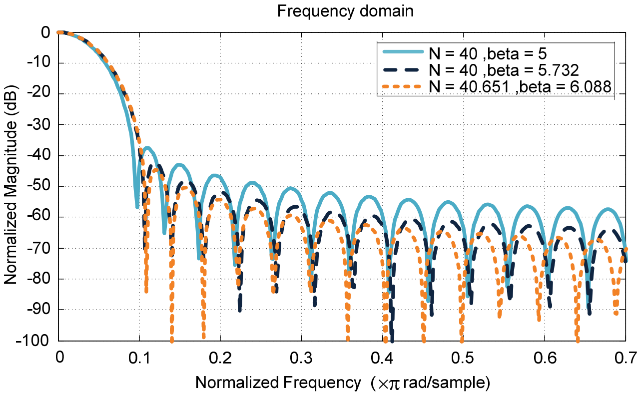

Fig.10. Frequency domain with a filter order of 40, beta is 5, 5.732, and the filter order of 30.632, beta is 6.088.

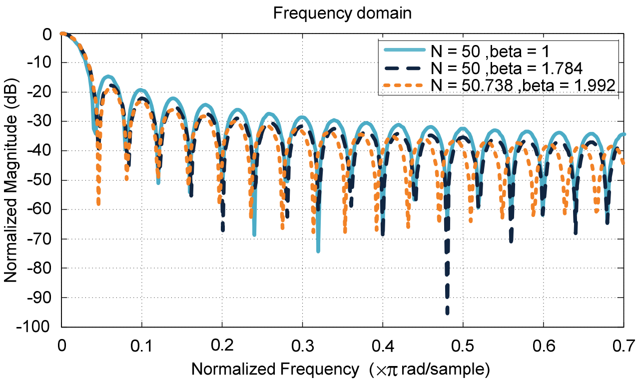

Fig.11. Frequency domain with a filter order of 50, beta is 1, 1.784, and the filter order of 50.738, beta is 1.992.

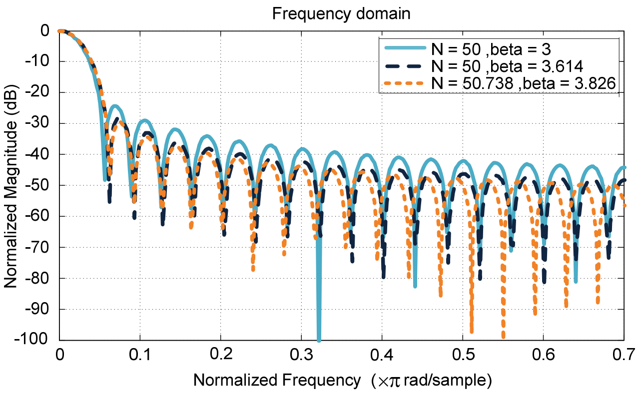

Fig.12. Frequency domain with a filter order of 50, beta is 3, 3.614, and the filter order of 50.738, beta is 3.826.

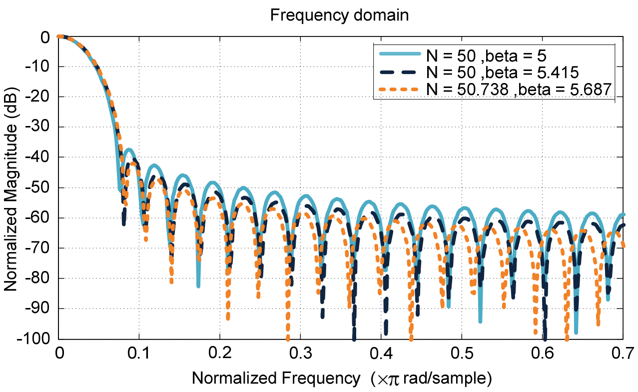

Fig.13. Frequency domain with a filter order of 50, beta is 5, 5.415, and the filter order of 50.738, beta is 5.687.

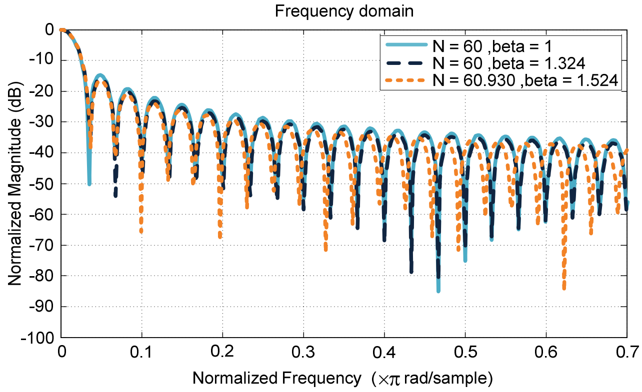

Fig.14. Frequency domain with a filter order of 60, beta is 1, 1.324, and the filter order of 60.930, beta is 5.687.

Table 3 shows the Kaiser window functions along with specified beta parameters, optimal beta parameters, and the optimal filter order using the bee colony algorithm.

Length of Window function	Variable parameter beta	Time Domain		Frequency Domain		
		Max. Amp	Min. Amp	Mainlobe width (-3dB)	Relative side lobe Attenuation	Leakage Factor
30	1	1	0.78	0.054688	-14.7 dB	6.33 %
	1.295		0.68		-15.8 dB	4.96 %
30.632	1.669		0.54		-17.3 dB	3.37 %
30	3		0.20	0.070313	-24.7 dB	0.49 %
	3.522		0.13		-28.1 dB	0.20 %
30.632	3.866		0.09		-30.4 dB	0.11 %
30	5		0.03	0.085938	-37.8 dB	0.01 %
	5.623		0.02		-41.7 dB	0 %
30.632	6.093		0.01		-44.7 dB	0 %
40	1	1	0.78	0.042969	-14.7 dB	6.51 %
	1.310		0.67		-15.7 dB	5.06 %
40.651	1.599		0.57		-16.9 dB	3.79 %
40	3		0.20	0.054688	-24.5 dB	0.54 %
	3.579		0.12		-28.2 dB	0.20 %
40.651	3.842		0.10		-29.9 dB	0.13 %
40	5		0.03	0.066406	-37.6 dB	0.02 %
	5.732		0.01		-42.3 dB	0 %
40.651	6.088		0.01		-44.6 dB	0 %
50	1	1	0.78	0.035156	-14.7 dB	6.47 %
	1.784		0.50		-17.7 dB	3.10 %
50.738	1.992		0.44		-18.7 dB	2.41 %
50	3		0.20	0.042969	-24.3 dB	0.56 %
	3.614		0.12		-28.2 dB	0.20 %
50.738	3.826		0.10		-29.6 dB	0.14 %
50	5		0.03	0.050781	-37.4 dB	0.02 %
	5.415		0.02		-40.1 dB	0.01 %
50.738	5.687		0.01		-42.0 dB	0.01 %
60	1	1	0.78	0.027344	-14.7 dB	6.44 %
	1.324		0.67		-15.8 dB	4.97 %
60.930	1.524		0.59		-16.5 dB	4.10 %
60	3		0.20	0.035156	-24.2 dB	0.57 %
	3.638		0.12		-28.3 dB	0.20 %
60.930	3.816		0.10		-29.4 dB	0.15 %
60	5		0.03	0.042969	-37.3 dB	0.02 %
	5.841		0.01		-42.9 dB	0 %
60.930	6.081		0.01		-44.5 dB	0 %

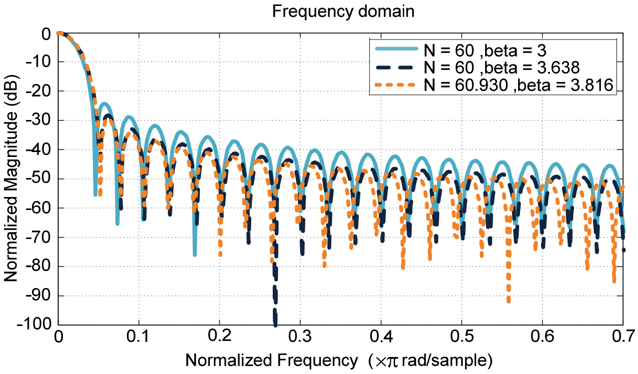

Fig.15. Frequency domain with a filter order of 60, beta is 3, 3.638, and the filter order of 60.930, beta is 3.816.

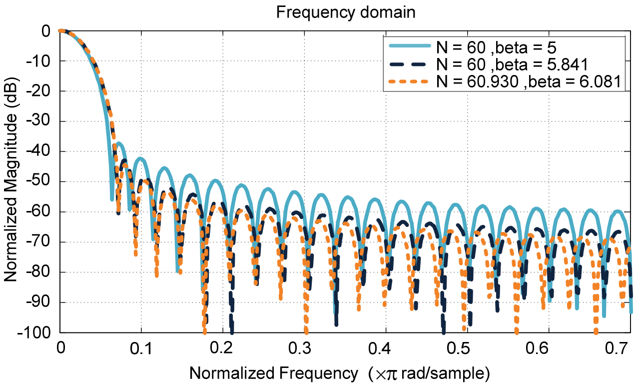

Fig.16. Frequency domain with a filter order of 60, beta is 5, 5.841, and the filter order of 60.930, beta is 6.081.

Figs. 5-16 depict the frequency domain using the Kaiser window function, and Table 3 illustrates the comparison between the specified filter order of 30, 40, 50, and 60 and the optimal filter order using the bee colony algorithm of 30.632, 40.651, 50.738, and 60.930. Using optimal filter order using the bee colony algorithm yields a newly improved set of beta parameters. It is clear that the side lobe and leakage factors decrease clearly when compared to the specified filter sequence and the beta values designed in Table 3. Meanwhile, the main lobe remains unchanged.

Conclusions

The article discusses enhancing efficiency using the utilization of bee colony algorithms to find beta parameters and filter order sequences using the Kaiser window function. Experiments 1 show that increasing the beta parameter results in an increase in the main lobe, while the side lobe and leakage factors decrease. At the same time, increasing the filter sequence indicates a decrease in both the main lobe and the side lobe. Experiment 2 illustrates a comparison between the specified beta parameters and the optimal beta parameters using the bee colony algorithm. It is evident that the optimal beta parameters using the bee colony algorithm are the main lobe being constant while the side lobe and leakage factors decreasing ostensibly. Finally, it compares the supplied filter order to the best filter order using the bee colony method. Using the ideal filter order results in a freshly enhanced set of beta parameters. It is clear that the main lobe is constant, while the side lobe and leakage factors are significantly lower when compared to the stated filter sequence.

Financing

This research was supported by the Faculty of Engineering, Rajamangala University of Technology Isan, Khon Kaen Campus, Khon Kaen, Thailand, Contract No. ENG 2/67.

Acknowledgements

This research was supported and funded by faculty of engineering, Rajamangala university of technology Isan, Khon Kaen campus, Khon Kaen, Thailand.

Authors:
Dr. Piyanuch TANGKITTIPON, Faculty of Engineering, Rajamangala University of Technology Isan, Khon Kaen Campus, Khon Kaen, Thailand, Email: piyanuch.ch@rmuti.ac.th
Yotaka TUNGTRAGUL, Faculty of Engineering, Rajamangala University of Technology Isan, Khon Kaen Campus, Khon Kaen, Thailand, Email: yotaka.tu@rmuti.ac.th
Arintorn JESADAMETHAKAJORN, Faculty of Engineering, Rajamangala University of Technology Isan, Khon Kaen Campus, Khon Kaen, Thailand, Email: arintorn.je@rmuti.ac.th
Dr. Atirarj SUKSAWAD, Faculty of Engineering, Rajamangala University of Technology Isan, Khon Kaen Campus, Khon Kaen, Thailand, Email: atirarj.su@rmuti.ac.th

Corresponding Authors: Dr. Atirarj SUKSAWAD
Email: atirarj.su@rmuti.ac.th

REFERENCES

[1] SHAYESTEH, Mahrokh G., MOTTAGHI-KASHTIBAN, Mahdi., FIR filter design using a new window function. 2009 16th International Conference on Digital Signal Processing. IEEE, (2009), p. 1-6.
[2] Patel, R., Kumar, M., Jaiswal, A. K., Saxena, R., Design technique of bandpass FIR filter using various window function, IOSR-JECE, (2013), 6: 52-7.
[3] ELETI, Alia Ahmed., ZEREK, Amer R., FIR digital filter design by using windows method with MATLAB, 14th International Conference on Sciences and Techniques of Automatic Control & Computer Engineering-STA'2013 .IEEE, (2013), p. 282-287.
[4] RAKSHIT, Hrishi., ULLAH, Muhammad Ahsan., A comparative study on window functions for designing efficient FIR filter, 2014 9th international forum on strategic technology (IFOST). IEEE, (2014), p. 91-96.
[5] SHIL, Mitun., RAKSHIT, Hrishi., ULLAH, Hadaate., An adjustable window function to design an FIR filter, 2017 IEEE International Conference on Imaging, Vision & Pattern Recognition (icIVPR). IEEE, (2017), p. 1-5.
[6] MOTTAGHI-KASHTIBAN, Mahdi., SHAYESTEH, Mahrokh G., A new window function for signal spectrum analysis and FIR filter design, 2010 18th Iranian Conference on Electrical Engineering. IEEE, (2010), p. 215-219.
[7] XU, Yanjie., Design of FIR filter with several window functions, 2021 IEEE 3rd International Conference on Frontiers Technology of Information and Computer (ICFTIC). IEEE, (2021), p. 638-642.
[8] KAYA, Turgay., İNCE, Melih Cevdet., The FIR filter design by using window parameters calculated with GA, 2009 Fifth International Conference on Soft Computing, Computing with Words and Perceptions in System Analysis, Decision and Control. IEEE, (2009), p. 1-4.
[9] ROY, Tushar Kanti., MORSHED, Monir., Performance analysis of low pass FIR filters design using Kaiser, Gaussian and Tukey window function methods, 2013 2nd international conference on advances in electrical engineering (icaee). IEEE, (2013), p. 1-6.
[10] AHMED, Suhaib., Design Analysis of High Pass FIR Filters Using Hanning, Bartlett and Kaiser Windows, International Journal of Advanced Research in Computer Science and Software Engineering, (2012), 2.11.
[11] GAUTAM, Ganesh., SHRESTHA, Surendra., CHO, Seongsoo., Spectral Analysis of Rectangular, Hanning, Hamming and Kaiser Window for Digital Fir Filter, Int. J. Adv. Smart Converg, (2015), 4: 138-144.
[12] GUPTA, Sonika., PANGHAL, Aman., Performance analysis of fir filter design by using rectangular, hanning and hamming

windows methods, International Journal of Advanced Research in Computer Science and Software Engineering, (2012), 2.6.
[13] AHMED, Suhaib., BASHIR, Mudasir., SURI, Ashish., Low Pass FIR Filter Design and Analysis Using Hamming, Blackman and Kaiser Windows, International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, (2014), 3.4.
[14] RAZA, Ghulam Ahmad., ALAM, Md Jafir., ANSARI, Md Nasim., Design and performance analysis of band pass filter using Blackman, Hamming and Kaiser windows, IJRET, (2014), 3: 211-214.
[15] RAZA, Ghulam Ahmad., ALAM, Md Jafir., Design and Performance Analysis of Low Pass Fir Filter Using Hamming and Kaiser Windows, Int. J. Adv. Res. Sci. Eng, (2014), 3: 394-400.
[16] Arya, R., Jaiswal, S., Design of Low pass FIR Filters using Kaiser Window Function with variable parameter Beta (β), International Journal of Multidisciplinary and Current Research, (2015), 3, 220-224.
[17] MURALIDHAR, P., V, et al., Implementation of different FIR high pass filters using fractional Kaiser window, 2010 2nd International Conference on Signal Processing Systems. IEEE, (2010), p. V2-651-V2-655.
[18] AVCI, Kemal., Performance analysis of Kaiser-Hamming Window for nonrecursive digital filter design, 2013 21st Signal Processing and Communications Applications Conference (SIU). IEEE, (2013), p. 1-4.
[19] SUCKLEY, D., Genetic algorithm in the design of FIR filters, IEE Proceedings G (Circuits, Devices and Systems), (1991), 138.2: 234-238.
[20] CHAUHAN, Ranjit Singh., ARYA, Sandeep K., An optimal design of FIR digital filter using genetic algorithm Contemporary Computing: 4th International Conference, IC3 2011, Noida, India, August 8-10, 2011. Proceedings 4. Springer Berlin Heidelberg, (2011), p. 51-56.
[21] KARABOGA, Nurhan., CETINKAYA, Bahadir., Design of digital FIR filters using differential evolution algorithm, Circuits, Systems and Signal Processing, (2006), 25: 649-660.
[22] CHATTOPADHYAY, S., SANYAL, S. K., CHANDRA, A., Optimization of control parameter of differential evolution algorithm for efficient design of FIR filter, 2010 13th International Conference on Computer and Information Technology (ICCIT). IEEE, (2010), p. 267-272.
[23] LUITEL, Bipul., VENAYAGAMOORTHY, Ganesh K., Differential evolution particle swarm optimization for digital filter design, 2008 IEEE Congress on Evolutionary Computation (IEEE World Congress on Computational Intelligence). IEEE, (2008), p. 3954-3961.
[24] Ababneh, J. I., Bataineh, M. H., Linear phase FIR filter design using particle swarm optimization and genetic algorithms, Digital Signal Processing, (2008), 18.4: 657-668.
[25] Mandal, S., Mallick, P., Mandal, D., Kar, R., Ghoshal, S. P., Optimal FIR band pass filter design using novel particle swarm optimization algorithm, 2012 IEEE Symposium on Humanities, Science and Engineering Research. IEEE, (2012), p. 141-146.
[26] Mandal, S., Ghoshal, S. P., Kar, R., Mandal, D., Kishore, N. V. R., FIR band stop filter optimization by improved particle swarm optimization, 2011 World Congress on Information and Communication Technologies. IEEE, (2011), p. 699-704.
[27] Mondal, S., Chakraborty, D., Kar, R., Mandal, D., Ghoshal, S. P., Novel particle swarm optimization for high pass FIR filter design, 2012 IEEE Symposium on Humanities, Science and Engineering Research. IEEE, (2012), p. 413-418.
[28] Kar, R., Mandal, D., Mondal, S., Ghoshal, S. P., Craziness based particle swarm optimization algorithm for FIR band stop filter design, Swarm and evolutionary computation, (2012), 7: 58-64.
[29] Li, H., Zhang, A., Zhao, M., XU, Q., Particle swarm optimization algorithm for FIR digital filters design, ACTA ELECTONICA SINICA, (2005), 33.7: 1338.
[30] SINGH, Ajay Pal., Design of linear phase low pass fir filter using particle swarm optimization algorithm, international journal of computer applications, (2014), 98.3.
[31] KARABOGA, Nurhan., KALINLI, Adem., KARABOGA, Dervis., Designing digital IIR filters using ant colony optimisation algorithm, Engineering applications of artificial intelligence, (2004), 17.3: 301-309.
[32] LOUBNA, Kritele., BACHIR, Benhala., IZEDDINE, Zorkani., Optimal digital IIR filter design using ant colony optimization, 2018 4th International Conference on Optimization and Applications (ICOA). IEEE, (2018), p. 1-5.
[33] SEVERINO, Alcemy GV., LINHARES, Leandro LS., DE ARAÚJO, Fábio MU., Optimal design of digital low pass finite impulse response filter using particle swarm optimization and bat algorithm, 2015 12th International Conference on Informatics in Control, Automation and Robotics (ICINCO). IEEE, (2015), p. 207-214.
[34] KAUR, Harmandeep., SAINI, Satish., SEHGAL, Amit., Optimal Designing of FIR Filter with Hybrid Bat Optimization Algorithm, Innovations in Electronics and Communication Engineering: Proceedings of the 9th ICIECE 2021. Singapore: Springer Singapore, (2022), p. 523-541.
[35] TUNGTRAGUL, Yotaka., SUKSAWAD, Atirarj., SA-NGIAMVIBOOL, Worawat., The Optimal Design of Finite Impulse Response High Pass Filter Using Bee Colony Algorithm, Przeglad Elektrotechniczny, (2023), 12.
[36] Bose, D., Biswas, S., Vasilakos, A. V., Laha, S., Optimal filter design using an improved artificial bee colony algorithm, Information Sciences, (2014), 281: 443-461.
[37] KARABOĞA, NURHAN., CETINKAYA, Mehmet Bahadir., A novel and efficient algorithm for adaptive filtering: artificial bee colony algorithm, Turkish Journal of Electrical Engineering and Computer Sciences, (2011), 19.1: 175-190.