https://www.sworldjournal.com/index.php/swj/article/view/swj19-01-036
DOI: 10.30888/2663-5712.2023-19-01-036

UDC 004.02

ALGORITHM FOR ORDERING CASCADES OF NONRECURSIVE DIGITAL FILTERS BASED ON A GENETIC ALGORITHM

АЛГОРИТМ ВПОРЯДКУВАННЯ КАСКАДІВ НЕРЕКУРСИВНИХ ЦИФРОВИХ ФІЛЬТРІВ НА БАЗІ ГЕНЕТИЧНОГО АЛГОРИТМУ

Petrosian R.V. / Петросян Р.В.
ORCID: 0000-0002-0388-8821
Petrosian A.R. / Петросян А.Р.
ORCID: 0000-0003-0960-8461
Postgraduate student / аспірант
Loktikova T.M. / Локтікова Т.М.
ORCID: 0000-0002-3525-0179
Kuzmenko O.V. / Кузьменко О.В.
ORCID: 0000-0002-4937-3284
Zhytomyr Polytechnic State University, Zhytomyr, Ukraine
Державний університет «Житомирська політехніка», Житомир, Україна

Abstract. *The paper proposes an algorithm for ordering FIR filter cascades based on a genetic algorithm to minimize the error arising from the quantization of multiplication results. Choosing the optimal order of arrangement of cascades is a difficult problem, as it belongs to the class of NP-complete problems.*

A noise model of the sequential form of FIR filter implementation was chosen. According to the noise model, an algorithm for ordering filter cascades based on a genetic algorithm has been developed. A genetic algorithm was used to solve the problem, as it allows obtaining a quasi-optimal solution in less time compared to algorithms that provide exact solutions.

A fitness function is proposed for minimizing the error when ordering the cascades of the FIR filter. A detailed analysis of the methods of selection, crossing and mutation was carried out, as well as the selection of the methods most suitable for solving the given task. The hyperparameters of the genetic algorithm have been tuned for greater efficiency.

A number of experiments were carried out to check the algorithm work. Several FIR filters of different types and orders were synthesized. As a result of experimental studies, it was possible to find out that the developed algorithm really has a high speed compared to algorithms that allow obtaining exact solutions. The proposed algorithm is more effective when ordering a large number of cascades, because it significantly reduces the time spent on finding a solution to the problem.

Key words: *ordering of FIR filter cascades, sequential form of FIR filter implementation, design of FIR filters, genetic algorithm.*

Introduction

Digital filters (DF) are an important signal processing tool and are widely used in various fields. A common form of implementation of digital filters is sequential. In this case, the transfer function of the DF is represented as a product of the transfer functions of several DF (cascades) of lower orders. The presence of the quantization effect leads to the fact that the error depends on the mutual arrangement of the cascades, so the problem arises of finding such an order of arrangement of the cascades to minimize the output error of the DF.

Analysis of recent research and publications on which the authors rely

FIR filters are one of the most common filter types in digital signal processing due to their simple implementation, linearity, scalability, and high performance in signal processing. They are used to extract symmetrical components in three-phase circuits, filter mechanical vibrations in unmanned aerial vehicles, remove noise in acoustic systems, etc., so interest in them is constantly growing.

However, during digital signal processing, there are always errors in the output signal, including when using DF. The reasons for such errors are:

- an error due to non-ideal frequency characteristics of the digital filter;
- errors of digital filter coefficients;
- quantization errors of multiplication results in DF.

The error due to the non-ideal frequency characteristics of DF depends on the design method used and the order of the filter. Obviously, to reduce this error, it is necessary to increase the order of DF, which is designed. There are many design methods, among which classical [1-3] and heuristic methods of designing FIR filters [4-7] can be distinguished. Among the heuristic algorithms, it is possible to pay attention to the genetic algorithm (GA) [8, 9], which has a number of advantages over other optimization algorithms: it belongs to global optimization methods and is parallel scalable. Also it can be used for multi-criteria optimization, etc. Works [2, 3, 10] provide a description and comparative analysis of some FIR filter design methods.

Errors in DF coefficients arise during the design of FIR filters, and also due to the limited bit capacity of microprocessor or microcontroller registers [2, 3]. The sensitivity of the frequency response to the errors of the DF coefficients depends on the structure of the FIR filter that was chosen during the design. The most common structure of given filters, which has a low frequency response sensitivity to the errors of the DF coefficients, is a sequential form of FIR filter implementation.

Quantization errors of multiplication results in DF arise during filtering of the input signal due to the limited bit capacity of the microprocessor or microcontroller registers [2, 3]. The size of the output error of the given filter, which is due to quantization of the results of multiplication operations, depends on the structure of the FIR filter chosen during the design. However, when using a sequential form of DF implementation, this error also depends on the order of arrangement of cascades in the filter.

The difficulty of choosing the optimal order of arrangement of FIR filter cascades is that as the number of filter cascades increases, the number of options for its implementation increases significantly, which will be determined according to expression (1):

$$N = S!, \tag{1}$$

where S is the number of cascades of the FIR filter.

Therefore, it can be noted that the problem of arranging cascades in the sequential form of FIR filter implementation belongs to the class of NP-complete problems.

In [3], general recommendations for the ordering of cascades in the FIR filter are offered, which allow to reduction this error.

In [11], the Bellman dynamic programming method is used to obtain the optimal arrangement of DF cascades in order to reduce the output error. This method allows you to get the optimal arrangement of the filter cascades, but the complexity of this algorithm is exponential.

It can be noted that the algorithms that allow us to obtain an optimal solution to our problem require a long time. To solve this class of problems, heuristic algorithms are often used, because they allow obtaining a quasi-optimal solution, but the time to search for this solution is significantly reduced. Among such works is [12], where it is proposed to perform ordering and distribution of coefficients in the implementation of FIR-filter structures on the basis of heuristic algorithms in order to minimize the energy consumption of VLSI (very large scale integration).

The purpose of the paper is to develop an algorithm for an approximate solution to the problem of ordering cascades in a sequential form of FIR filter implementation to minimize the output error.

The research method

The generalized diagram of the sequential form of DF implementation is presented in fig. 1. Let's assume that the DF is synthesized, decomposed into cascades and the DF coefficients are limited by the required bit depth, so it remains to determine such an arrangement of the cascades that will ensure the minimum output error of DF.

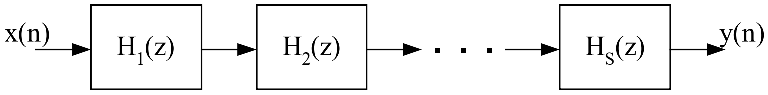

Figure 1 – Sequential form of DF implementation

where x(n), y(n) are input and output data sequences, respectively.

Every cascade is a source of quantization errors ε_s. In this case, the influence of all component errors can be shown using the following diagram (Figure 2).

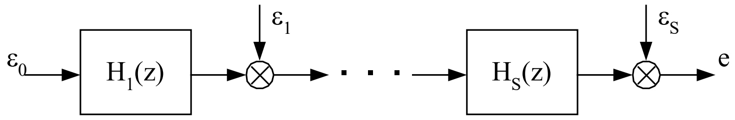

Figure 2 – Noise model of DF in sequential implementation

According to the Figure 2, the quantization noise at the filter output will be determined according to expression (2):

$$e(z) = \sum_{s=1}^{S} \varepsilon_s(z) H'_{s+1}(z), \tag{2}$$

where $\varepsilon_s(z)$ is the quantization noise of each s-th cascade; $H'_s(z)$ is the transfer function of the last $S-s+1$ cascades, i.e. $H'_s(z) = \prod_{u=s}^{S} H_u(z)$ (consider $H'_{S+1}(z)=1$).

Each noise source can be considered as an independent source of "white" noise with spectral density $S\varepsilon_s$ and dispersion $D\varepsilon_s$. The spectral density of each noise source at the DF output is determined by (3):

$$S\varepsilon_{s\,df}(z) = S\varepsilon_s(z) H'^2_{s+1}(z), \tag{3}$$

and the dispersion of each noise source at the output of DF will be determined according to expression (4):

$$D\varepsilon_{s\,df} = D\varepsilon_s \sum_n h^2_{s+1}(n), \tag{4}$$

where $h_{s+1}(n)$ is the impulse characteristic of DF $H'_{s+1}(z)$.

Therefore, the dispersion of the resulting quantization noise at the output of the DF is determined as the sum of individual components (5):

$$D\varepsilon_{df} = \sum_{s} D\varepsilon_{s\,df} \,. \tag{5}$$

Expression (5) can be considered as a function (fitness function) that allows minimizing the output error of the FIR filter caused by quantization of the result of multiplication operations. Having used the fitness function, it is possible to perform its minimization with the help of GA.

The key concept of GA is an individual who encodes a possible solution to the problem. An individual is characterized by a chromosome or a set of chromosomes. During the solution of the task, a population of individuals is created. Each individual is evaluated by the degree of fitness determined by the fitness function. This way, individuals who have better adapted to the "environment" (have the best solution) are identified. GA is iterative, so at each iteration a new population of individuals with better fitness than the previous one is generated.

The creation of a new population occurs by applying genetic operators (selection, crossover, mutation) to current individuals.

There are quite a large number of different selection methods, but the following methods are more common: roulette method, ranking method, tournament selection. Each of the selection methods has advantages and disadvantages. A disadvantage of the roulette method is that individuals with a very small value of the fitness function are removed from the population too quickly, which can lead to premature convergence of the GA. To prevent this effect, the scaling of the fitness function is used. A disadvantage of the ranking method is that individuals may receive the same rank, which means that they will have the same opportunity to choose, despite their differences in fitness. The difficulty of choosing the size of the tournament should be attributed to the disadvantages of the tournament method. When solving the problem, the last method of selection was used - the tournament method with 3 people as more effective.

There are various methods of crossover, but when choosing them, it is necessary to take into account the peculiarity of the phenotype of the task. In our problem, we need to get the order of FIR filter cascades, so the chromosome will represent a list of cascade numbers. Obviously, most crossover methods are not suitable for our problem, because the cascade number cannot be repeated in the list. Among the crossover methods that can be used to solve our problem, we can highlight: order crossover; partially mapped crossover; cyclic crossover. The analysis showed that there are no significant differences in the solutions, so any of the methods can be used. In our case, we settled on an order crossover.

Taking into account the phenotype of the problem, the following mutation methods can be used: mutation by shuffling and mutation by exchange. When solving the problem, the first method was used, which is more effective, because it makes more significant changes in the adaptability of individuals.

The algorithm for ordering the cascades of the FIR filter using GA will look like this:

1. The initial population of GA is formed, where each individual represents a list of the location of the FIR filter cascades;
2. The suitability of each individual is calculated (the output quantization error at the output of the FIR filter is calculated);
3. Selection of individuals with the best suitability for creating a new population is carried out (the most successful solutions are selected);
4. If the stopping condition is met, an individual with the best fitness is returned;
5. The operation of crossover selected individuals to form a new population is performed;
6. The mutation operation is performed;
7. Go to point 2 to process the new population.

The condition for stopping the algorithm can be both the formation of a certain number of generations and insignificant changes in the values of the fitness function.

A number of experiments were carried out to check the operation of the algorithm. Several FIR filters of different types and orders were synthesized. The results of the algorithm will be compared with the method of complete search (brute force search).

Modeling was carried out using the Python programming language. Hyperparameters must be configured to implement GA. In our case, they will look like this:

```
POPULATION_SIZE = 10      # the number of individuals in the population
P_CROSSOVER = 0.9         # the probability of crossover of an individuals
P_MUTATION = 0.1          # the probability of mutation of an individuals
MAX_GENERATIONS = 50      # the maximum number of generations
```

Figure 3 shows the results of ordering the cascades of the 8th-order FIR filter using GA.

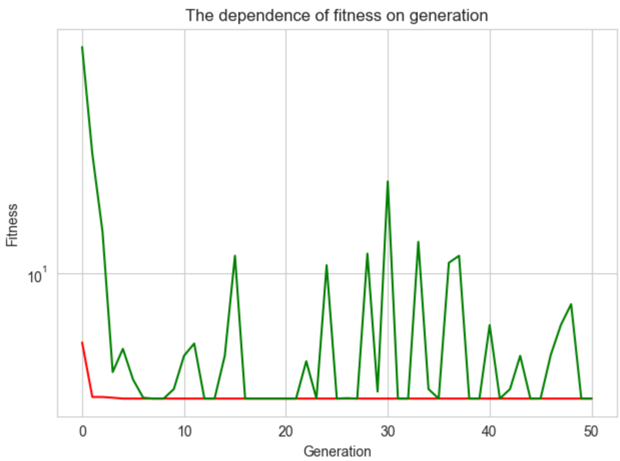

Figure 3 – Change in fitness by generation (8th-order DF)

This filter consists of 4 cascades with the following coefficients: 1. [0.02827533, -0.12520231, 0.21901412]; 2. [1, 1.73074291, 1]; 3. [1, 1.96213129, 1]; 4. [1, -0.57166319, 0.12910277].

As can be seen, the value of the errors can differ by more than an order of magnitude. The minimum value corresponds to the arrangement of the cascades (2, 1, 3, 0) and is equal to the value $3.339D_m$, where D_m is the dispersion of quantization of one multiplication operation. The result was found after 4 generations. And although the average fitness of the generations changed to find a better solution, the minimum fitness remained unchanged. Obviously, this is the best solution, which coincides with the method of complete search.

For a better approximation of the output filter, let's increase the order of the FIR filter to 20. This filter will consist of 10 cascades with the following coefficients: 1. [6.32109790e-04, -3.17828355e-03, 4.23190299e-03]; 2. [1, -3.47495702, 5.62408813]; 3. [1, -1.06171835, 4.05576043]; 4. [1, 1.77252184, 1]; 5. [1, 1.98892234, 1]; 6. [1, 1.6528648, 1]; 7. [1, 1.90612165, 1], 8. [1, -2.61780342e-01, 2.46562887e-01]; 9. [1, -6.17870300e-01, 1.77806602e-01]; 10. [1, -7.51029398e-01, 1.49367741e-01].

After the complete search a result was obtained, but the search time was 67 minutes! The minimum value corresponds to the value of $3.083D_m$. The result of finding the optimal arrangement of the stages of the 20th order FIR filter using GA can be seen in Figure 4. A fairly good solution was found after 10 generations, an even better one after 14 generations ($3.097D_m$), and the optimal solution was found after 30 generations ($3.096D_m$). This is not the best solution, because it does not coincide with the method of complete search and corresponds to the arrangement of cascades (1, 4, 7, 3, 6, 2, 9, 8, 5, 0). The difference in error between the variants is very small, but the time to find the solution is significantly different (the GA search time was less than 2 seconds).

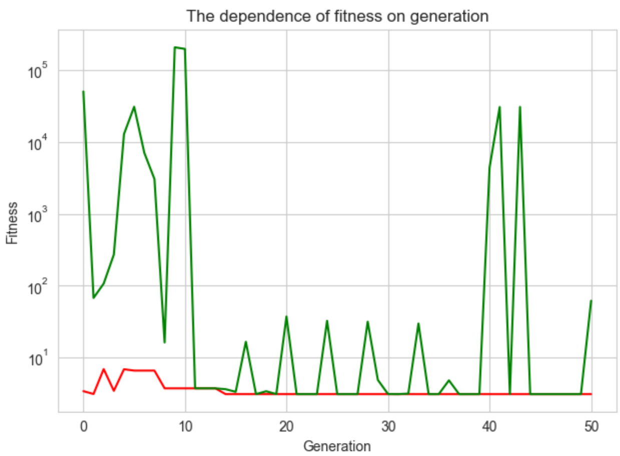

Figure 4 – Change in fitness by generation (20th-order DF)

Conclusions and prospects for further research

During the analysis, the noise model of the sequential implementation of the FIR filter was chosen. According to the noise model, an algorithm for arranging filter cascades based on GA was developed, which minimizes the total quantization error at the filter output. To solve this optimization problem, the following were chosen: tournament method of selection with three people; method of orderly crossing; method of mutation based on shuffling.

As a result of experimental studies, it was possible to find out that the developed algorithm has a high performance compared to algorithms that allow obtaining exact solutions. The proposed algorithm is more effective when ordering a large number of cascades (6 or more), because it significantly reduces the time spent on finding a solution to the problem.

References

1. Gazi O. Understanding Digital Signal Processing. Singapore : Springer Singapore, 2018. DOI: 10.1007/978-981-10-4962-0.

2. D. G. Manolakis, J. G. Proakis. Digital signal processing. New International Edition. Pearson Higher Ed, 2013.

3. L. R. Rabiner, B. Gold. Theory and application of digital signal processing. Prentice-Hall, Englewood Cliffs, N.J, 1975.

4. Ruslan Petrosian, Oleksandr Kuzmenko, Arsen Petrosian. Method for calculating the FIR filter based on genetic algorithm. Computer Systems and Information Technologies. 2021. Vol. 1. P. 19–24. URL: https://csitjournal.khmnu.edu.ua/index.php/csit/article/view/45/33.

5. Thakare V. V., SahuR K .A. Genetic Algorithm for Optimizationof MSE & Ripples in Linear Phase Low Pass FIR Filter & Also Compare with Cosine Window Techniques. International Journals Digital Communication and Analog Signals. 2015. Vol 1, No. 1. P. 9–15. DOI: 10.37628/jdcas.v1i1.42.

6. Optimal design of digital FIR filters based on back propagation neural network / Yang J. et al. IEICE Electronics Express. 2023. Vol. 20, No 1. P. 20220491-20220491. DOI: 10.1587/elex.19.20220491.

7. Petrosian R., Chukhov V., Petrosian A. Development of a method for synthesis the FIR filters with a cascade structure based on genetic algorithm. Technology audit and production reserves. 2021. Vol. 4, No. 2(60). P. 6–11. DOI: 10.15587/2706-5448.2021.237271.

8. Mutingi M., Mbohwa C. Grouping genetic algorithms: Advances and Applications. Switzerland: Springer International Publishing. 2017. 243 p.

9. Runwei Cheng J., Gen M. Parallel Genetic Algorithms with GPU Computing. Industry 4.0 - Impact on Intelligent Logistics and Manufacturing. 2020. DOI: 10.5772/intechopen.89152.

10. Kaur S., Singh B., Singh M. Different Design Approaches for the Optimization of FIR Filter Coefficients. Research Cell: An International Journal Of Engineering Sciences. 2016. Vol. 17. P. 229-236. URL: http://ijoes.vidyapublications.com/paper/Vol17/32-Vol17.pdf.

11. Petrosian R. V. Vporiadkovuvannia kaskadiv tsyfrovykh filtriv pry kaskadnii realizatsii. Visnyk ZhDTU. Seriia "Tekhnichni nauky". 2008. №2(45). S. 90–92. URL: http://vtn.ztu.edu.ua/article/view/81607/79196.

12. da Luz A. G., da Costa E. A. C., de Aguiar M. S. Ordering and partitioning of coefficients based on heuristic algorithms for low power FIR filter realization. Proceedings of the 23rd symposium on Integrated circuits and system design. 2010. P. 180-185. DOI: 10.1145/1854153.1854198.

Анотація. *У статті пропонується алгоритм впорядкування каскадів КІХ-фільтра на основі генетичного алгоритму для мінімізації похибки, що виникає внаслідок квантування результатів множення. Вибір оптимального порядку розташування каскадів є складною задачею, оскільки вона належить до класу NP-повних задач.*

Було обрано шумову модель послідовної форми реалізації КІХ-фільтра. Відповідно до шумової моделі розроблено алгоритм впорядкування каскадів фільтра на основі генетичного алгоритму. Для розв'язання задачі використано генетичний алгоритм, тому що він дозволяє *отримати квазіоптимальний розв'язок за менший час у порівнянні з алгоритмами, які дають точні розв'язки.*

У роботі запропоновано фітнес-функцію для мінімізації похибки при впорядкуванні каскадів КІХ-фільтра. Проведено детальний аналіз методів селекції, схрещування та мутації, а також здійснено вибір методів, які найбільш підходять для вирішення поставленої задачі. Виконано налаштування гіперпараметрів генетичного алгоритму для отримання більшої ефективності.

Для перевірки роботи алгоритму проведено низку експериментів. Було синтезовано декілька КІХ-фільтрів різного типу та порядку. В результаті експериментальних досліджень вдалося з'ясувати, що розроблений алгоритм дійсно має високу швидкодію порівняно з алгоритмами, які дозволяють отримати точні розв'язки. Запропонований алгоритм більш ефективний при впорядкуванні великої кількості каскадів, тому що суттєво зменшує витрати часу на пошук розв'язку задачі.

Ключові слова: *впорядкування каскадів КІХ-фільтра, послідовна форма реалізації КІХ-фільтра, проектування КІХ-фільтрів, генетичний алгоритм.*

Стаття відправлена: 17.05.2023 г.