Volume 4, Issue 4, April 2014 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Survey Paper of Digital IIR Filter Design

Er.Daljit Singh Bajwa, Er. Karamjeet Singh, Navpreet Kaur Chahal
ECE &BBSBEC,Fatehgarh Sahib
India

Abstract-This paper based on the design of digital filter. Digital filters are basically of two types- FIR filter and IIR filter, digital IIR filter with fixed point representation is to be designed. Optimization techniques are used to design optimal digital IIR filter. Firstly digital low pass IIR filter is to be designed. Then, analyze the fitness landscape properties of optimal digital low pass IIR filter. Based upon the fitness landscape investigation, apply two-state ensemble evolutionary algorithm to the above design optimal digital low pass IIR filter with fixed-point representation. In order to evaluate the performance of TEEA, we experimentally compare it with three state-of-arts of EA's on four types of digital IIR filters with different settings.

Keywords- Digital infinite-impulse response (IIR) filter Order, Continue evolutionary algorithm, Fixed-point representation.

I. INTRODUCTION:

Digital IIR filter is one of the most commonly used computation tools in the digital signal processing systems. Digital IIR filters are used many application such as high-speed and low-power communication transceivers, they are also used in routinely employed as accustom designed digital block [1]**.** IIR Filters are infinite response filter, they have impulse response of infinite duration. IIR filter are recursive filter in which feedback is present from output side to the input side. IIR filter response is depending upon the previous output samples, present and past input samples. Digital IIR filters are one of the most frequently used computational tool used in digital signal processing. It is used in much application such as high-speed and low-power communication transceivers systems. Difference equation of IIR filter is [2]:

A. *Method used to design analog IIR filter*: Butterworth filter approximation, Cheyshev filter , Elliptic filter [2].
B. *Method used to design digital IIR filter*: Bilinear transformation, impulse invariance method, Matched Z-transformation [2].

The bilinear transformation method is used to design the IIR filter. Firstly, design the analog IIR filter using Butterworth filter approximation, cheyshev filter approximation, and elliptic filter approximation. Then, the analog IIR filter is converted into digital IIR filter using bilinear transformation approximation [3].

C. *IIR filter design types*: Low-pass (LP), High-pass (HP), Band-pass (BP), Band-stop (BS) filter with different settings in order, pass-band frequency and stop band frequency, stop-band attenuation while pass-band ripple kept fixed [2,3].

II. Fixed-point Representation:

Fixed-point representation is a decimal representation of the numbers as a string of digits with decimal point. The number right side to the decimal point is known as fractional number and number left side to the decimal point is called integer part of the number. General representation of fixed-point representation is [2,4]:

$$X= (b_{-A},\ldots\ldots\ldots\ldots, b_{-1},b_0,b_1,\ldots\ldots\ldots\ldots,b_B)_r$$

$$X=\sum^{B}_{i=-A} b_i\, r^{-I} \qquad 0\leq b_i \leq(r-1)$$

b_i = digits, r= radix or base , A= integer digits, B=fractional digits

A fixed-point representation allow to cover a range of numbers with the resolution

Where m= 2^b is a number of levels, b= number of bits or digits.

A basic characteristic of the fixed-point resolution is to provide the uniform resolution throughout the given number range. Therefore Δ increases directly with increase in dynamic range.

A. *Classes of fixed-point representation*:

Most commonly used types of fixed-point representation are: binary fixed-point representation and decimal fixed-point representation. Decimal fixed-point representations have scaling factor that is power of 10 whereas binary fixed-point representation have scaling factor that is power of 2. Binary fixed-point representations are commonly used because rescaling operation can be implemented as fast bit shifts.

B. *Operations:*

1) *Addition and Subtraction*: Addition and Subtraction of two same types of fixed-point numbers with common scaling factor will result in exactly same fixed-point type, as long as no overflow occurs. If two different types of fixed-point numbers are added or subtract with different scaling factor, then one of them converted into the other type before the sum.
2) *Multiplication:* In order to multiply fixed-point numbers, this process involves no rounding and resulting scaling factor of two given integers is the product of their scaling factor.
3) *Division:* if the fixed-point numbers are of same type then the quotients of the two integers must be explicitly divided by the common scaling factor.

C. *Comparison between Floating-Point and Fixed-Point Representation*:

1) Fixed-point representation consume less power and less costly for portable application as compare to floating point representation.
2) Fixed-point representation is efficiently save computational resources and is more convenient for direct realization on hardware whereas floating-point is impractical in hardware design and requires higher computational power.
3) Multiplication of two floating-point representation leads to overflow whereas no such overflow occur in fixed-point representation.
4) Fixed-point representation provide uniform resolution throughout the given number range whereas floating-point resolution provide the finer resolution for small number range and provide the coarse resolution for large number range.
5) Fixed-point representation would make the search space miss much useful gradient information and rises new challenges for continuous EAs.

III. **Why evolutionary algorithms are used to design digital IIR filter**?

The design of IIR filter using bilinear transformation need too much prior knowledge and show poor performance in most of the cases. To overcome this problem we use optimization approach which requires less prior knowledge and higher accuracy to design digital IIR filter with fixed-point representation [13, 14].

A. *Three classes of EA's to design digital IIR filters:*

1) Parameter estimation for single-objective digital IIR filter: this design process is expected to provide a final digital IIR filter to meet all requirements. The fitness landscape of digital IIR filter contain many local optima, the motivation of this class algorithms is to develop specific operator to strengthen the exploration ability. The algorithms of this class includes hybrid genetic algorithm, hierarchical genetic algorithm and Taguchi-strategy enhanced GA.
2) System identification for single-objective digital IIR filter: The major difference between parameter estimation and this class is that the fitness value of one individual changes from time to time. This problem belongs to the noise-induced optimization domain. Algorithms used to solve this problem are ACO, Seeker optimization, tabu search, artificial immune system. These algorithms have to work under uncertain condition. Thus accuracy cannot be efficiently guaranteed.
3) Multi-objective digital IIR filter: in the above classes, only one optimization problem is to be minimized i.e., magnitude response error with supplementary conditions but this class optimizes the three objectives: the magnitude response error, the linear phase response and order.

B. *Advantages of evolutionary algorithms over other methods used to design digital IIR filters [8, 19, 20]:*

1) Pre-knowledge of the problems is not necessary for EAs, while the highly nonlinear characteristic must be approximated for transformation methods and other mathematical optimization approaches.
2) EAs are usually work with the population of candidate solutions and can handle constraints adaptively under the strategy set beforehand in a single run

IV. **Design IIR filter with Fixed-point representation**:

To design the digital IIR filter using Fixed- point representation we use the following evolutionary algorithms:

A. Two-state ensemble evolutionary algorithm (TEEA)
B. Self-adapting differential evolution algorithm (SADE)
C. Self-adaptive mixed distribution based uni-variate estimation of distribution algorithm (MUEDA).
D. Self-adaptive control parameters in differential evolution (jDE).

A. *TWO-STATE ENSEMBLE EVOLUTIONARY ALGORITHM (TEEA):*

It is used to cope with the problem of conflict between the exploration and exploitation in digital IIR filter design [8]. To design digital IIR filter digital IIR filter design, a serial two-stage optimization framework is used in our new two-stage ensemble evolutionary algorithm (TEEA).TEEA divides the optimization procedure into two stages to design the digital IIR filter with fixed-point representation are[9]:

1) Global shrink stage: In this stage the objective is to shrink the searching scope to the promising area as quickly as possible with higher convergence speed.
2) Local exploration stage: this is the second stage of TEEA; the objective of this stage is s to explore the limited area extensively to find as good as possible solutions.

TEEA utilizes MUEDA and MDE as its sub-optimizers. A serial structure is chosen to effectively combine the merits of two EAs in TEEA: MUEDA is implemented in the first stage to extract the global information and force the search population to approach the global optimum as close as possible. When the search space reached difficult area then MUEDA fails to generate better solutions after this MDE is triggered to continue the search. The initial population of MDE is a mixture of selected good solutions inherited from MUEDA and new random generated solutions. The MDE will continue the search until the termination condition is met. A trigger is defined based on the criteria that best solution found in current generation is not better than those in the previous generation [8].

1) Sub-optimizer 1: Self-adaptive mixed distribution based uni-variate estimation of distribution algorithm: Probabilistic models are generally employed in estimation of distribution algorithms to describe the most promising area in the solution space and these models are used to guide the generation of the candidate solutions for the next generation. Uni-variate EDA named as MUEDA is adopted in TEEA in order to reduce the complexity of learning of probabilistic model. The exploration ability of the uni-variate EDA is enhanced by combining a levy model with Gaussian model to guide the generation of the candidates in MUEDA. In case of unimodal and many simple multimodal problems, MUEDA performs well in both convergence speed and accuracy but in case of complex multimodal problems its performance is unsatisfying. TEEA utilize the merit of fast convergence of MUEDA to conduct the first stage of the search [8, 9].

2) Sub-optimizer 2: Modified Differential Evolutionary Algorithm: Any evolutionary algorithm which is effectively coping with multimodal problems can be used as the second sub-optimizer. DE algorithm is used for this purpose because of its well-known robustness [10, 11, and 12]. Limitation of MUEDA is overcome by using MDE as a second sub-optimizer to conduct the second stage of search. In case of rugged region of fitness landscape, MUEDA may be failed to provide the better solution than MDE is used to re-boost the search. In MDE, two most effective differential evolution strategies (DE/rand/1 and DE/current to best/1) are probabilistically adopted based on adaptive chaotic control mechanism. The strategy DE/rand/1 is used with a larger probability due to its universality. The parameters of CR and F are self-adaptive and dependent. The general rule is to provide relatively large CR when F is small and relatively small CR when F is large.

B. Self-adaptive differential evolution algorithm (SADE):

In Self-adaptive differential evolution algorithm the choice of learning strategy and the two control parameters are not required to be pre-specified. During the evolution, according to learning experience a suitable learning strategy and parameters settings are gradually self-adapted. SADE is beneficial for adjusting control parameters during evolutionary process, especially when it done without an user interaction [11].

C. Self-adaptive mixed distribution based uni-variate estimation of distribution algorithm (MUEDA):

Probabilistic models are generally employed in estimation of distribution algorithms to describe the most promising area in the solution space and these models are used to guide the generation of the candidate solutions for the next generation. Uni-variate EDA named as MUEDA reduces the complexity of learning of probabilistic model. The exploration ability of the uni-variate EDA is enhanced by combining a levy model with Gaussian model to guide the generation of the candidates in MUEDA. In case of unimodal and many simple multimodal problems, MUEDA performs well in both convergence speed and accuracy but in case of complex multimodal problems its performance is unsatisfying [6].

D. Self-adaptive control parameters in differential evolution (jDE):

The self-adaptive differential evolution (jDE) algorithm is based on the self-adapting control parameter mechanism, was proposed by Brest et al 2006. The self-adaptive control mechanism is used to change control parameters, during the run. The parameters controlled by this algorithm are select weighting factor F and crossover constant CR. Then this control parameters F and CR are produced in a new parent vector [10].

V. RELATED WORK:

Two-stage based ensemble optimization framework for large-scale global optimization: This paper is based on the design of two-stage based ensemble optimization evolutionary algorithm framework for large scale global optimization. This algorithm is implemented by two sub-optimizers. According to different implementation conditions, EOEA framework can easily generated, modified and altered flexibly. To analyze the effect of EOEA's components, compare its performance on diverse kinds of problem with its two sub-optimizers and three varients. In order to show its superiorities over the LSGO algorithms, compare its performance with six classical LSGO algorithms on the CSGO test functions of IEEE Congress of Evolutionary computation (CEC 2008). To evaluate the performance of EOEA by experimental comparison with four state-of-the-arts LSGO algorithms on the test function of CEC 2010 LSGO competition [15].

Evaluation of Two-Stage Ensemble Evolutionary Algorithm for Numerical Optimization: This paper is tried to extend the application area of TSEA from specific engineering problems to numerical optimization problems by altering its sub-optimizers. The experimental studies contain three aspects in this paper ,firstly , the TSEA framework are experimentally compared by comparing TSEA with its sub-optimizers on 26 test functions. Secondly, TSEA is compared with diverse state-of-the-art evolutionary algorithms to show its advantage. Thirdly, the performance of TSEA is further compared with 4 classical memetic algorithm (MA's) on CECO's test functions. The experimental results show the excellent effectiveness, efficiency and reliability of TSEA [16].

Self-adapting control parameters in differential evolution: a comparative study on numerical benchmark problems: This paper presents an efficient technique for adapting control parameter settings associated with differential evolution (DE). The DE algorithm has only the few control parameters, which are kept fixed throughout the evolutionary process. Self-adapting control parameter in differential algorithm is a new version of the DE algorithm which is used for obtaining self-adaptive control parameter settings that show good performance on numerical benchmark problems. The results show that self-adapting control parameter in differential algorithm with self-adaptive control parameter settings is better than, or comparable to, the standard DE algorithm and evolutionary algorithms when considering the quality of the solutions obtained [17].

Two-stage ensemble memetic algorithm: Function optimization and digital IIR filter design: This paper proposed a design of Two-Stage ensemble Memetic Algorithm (TSMA) framework to more appropriately synthesize the strengths of the evolutionary global search and local search techniques. The first optimization stage is local search techniques in which a competition is held among the candidate. The major idea of the first optimization stage is to choose the best local search technique and to obtain good initial state. The second optimization stage is to be implementing effective adaptive MA to pursue high-quality solution. A suit of the higher-order digital IIR filters are designed by surpassing the threshold. To evaluate the performance of TSMA, TSMA is compared with six state-of-the-art EAs, including MDE, SaDE, jDE, CLPSO, PSO-cf and IPOP-CMA-ES. This experiment study provided that TSMA has better effectiveness and efficiency over the other algorithms. Future work can be done on the impact of the choice of global search method, how to apply MAs to very large-scale global optimization tasks and applying TSMA to other real-world engineering problems [18].

VI. CONCLUSION

A two-stage ensemble evolutionary algorithm (TEEA) for fixed-point digital IIR filter design, show the advantages of TEEA over the other state-of-the-art EAs and to evaluate the scalable characteristic of TEEA, apply it to more difficult problems with shorter word length, higher order and lower.

Future Work- To reduce the computational cost on high-quality IIR filter design and to obtain digital IIR filter with a shorter wordlength.

References:

[1] H. Choo, K. Muhammad, K. Roy, Complexity reduction of digital filters using shift inclusive differential coefficients, IEEE Transactions on Signal Processing 52 (June (6)) (2004) 1760–1772.
[2] John G. Proakis , Dimitris G. Manolakis, Digital Signal Processing, Pearson Prentice Hall, 2007.
[3] H.Y.F. Lam, Analog and Digital Filters: Design and Realization, Prentice-Hall, Englewood Cliffs, NJ, 1979.
[4] M. Sarkar, Fixed-Point Implementation of a High-Pass IIR Filter, tech. report, Philips, Leuven, Belgium, July 2001.
[5] T. Weise, Global Optimization Algorithms – Theory and Application, Chemnitz, Germany, 2009, http://www.it-weise.de/
[6] Y. Wang, B. Li, A self-adaptive mixed distribution based uni-variate estimation of distribution algorithm for large scale global optimization, in: R. Chiong (Ed.), Nature-Inspired Algorithms for Optimization, Studies in Computational Intelligence, vol. 193, Springer-Verlag, Hardcover, 2009, ISBN 978-3-642-00266-3, pp. 171–198.
[7] Y. Wang, B. Li, Y.B. Chen, Digital IIR filter design using multi-objective optimization evolutionary algorithm, Applied Soft Computing Journal (2010).
[8] Y. Wang, B. Li, T. Weise, Estimation of distribution and differential evolution cooperation for large scale economic load dispatch optimization of power systems, Information Sciences 180 (2010) 2405–2420.
[9] Y. Wang, B. Li, Two-stage based ensemble optimization for large-scale global optimization, in: Proc. the IEEE Congress on Evolutionary Computation (CEC 2010), July 2010, Barcelona, Spain, 2010, pp. 4488–4495.
[10] J. Brest, S. Greiner, B. Boskovic, M. Mernik, V. Zumer, Self-adapting control parameters in differential evolution: a comparative study on numerical benchmark problems, IEEE Transactions on Evolutionary Computation 10 (December (6)) (2006) 646–657.
[11] A.K. Qin, P.N. Suganthan, Self-adaptive differential evolution algorithm for numerical optimization, Proceedings of the IEEE Congress on Evolutionar Computation 2 (2005) 1785–1791.
[12] R. Storn, K. Price, Differential evolution – a simple and efficient heuristic strategy for global optimization over continuous spaces, Journal of Global Optimization 11 (1997) 341–359.
[13] Y. Yu, Y. Xinjie, Cooperative coevolutionary genetic algorithm for digital IIR filter design, IEEE Transactions on Industrial Electronics 54 (June (3)) (2007) 1811–1819.
[14] K.S. Tang, K.F. Man, S. Kwong, Z.F. Liu, Design and optimization of IIR filter structure using hierarchical genetic algorithms, IEEE Transactions on Industrial Electronics 45 (June (3)) (1998) 481–487.
[15] Y. Wang, B. Li, Two-stage based ensemble optimization for large-scale global optimization, in: Proc. the IEEE Congress on Evolutionary Computation (CEC 2010), July 2010, Barcelona, Spain, 2010, pp. 4488–4495.
[16] Yu Wang, Bin Li, Kaibo Zhang, and Zhen He, Evaluation of Two-Stage Ensemble Evolutionary Algorithm for Numerical Optimization
[17] J. Brest, S. Greiner, B. Boskovic, M. Mernik, V. Zumer, Self-adapting control parameters in differential evolution: a comparative study on numerical benchmark problems, IEEE Transactions on Evolutionary Computation 10 (December(6)) (2006) 646–657.
[18] Yu Wang, Bin Li , Thomas Weise, Two-stage ensemble memetic algorithm: Function optimization and digital IIR filter design, Information Sciences 220 (2013) 408–424.