

Design of Optimal Digital IIR Filters using the Genetic Algorithm

Jung-Doo Jang* and Seong G. Kong**

*Department of Electrical Engineering
Soongsil University
Seoul, 156-743 Korea

**Department of Electrical and Computer Engineering
The University of Tennessee
Knoxville, TN 37996-2100

Abstract

This paper presents an evolutionary design of digital IIR filters using the genetic algorithm (GA) with modified genetic operators and real-valued encoding. Conventional digital IIR filter design methods involve algebraic transformations of the transfer function of an analog low-pass filter (LPF) that satisfies prescribed filter specifications. Other types of frequency-selective digital filters as high-pass (HPF), band-pass (BPF), and band-stop (BSF) filters are obtained by appropriate transformations of a prototype low-pass filter. In the GA-based digital IIR filter design scheme, filter coefficients are represented as a set of real-valued genes in a chromosome. Each chromosome represents the structure and weights of an individual filter. GA directly finds the coefficients of the desired filter transfer function through genetic search for given filter specifications of minimum filter order. Crossover and mutation operators are selected to ensure the stability of resulting IIR filters. Other types of filters can be found independently from the filter specifications, not from algebraic transformations.

1. Introduction

Design of digital infinite impulse response (IIR) filters involves determining the coefficients and the order of a filter transfer function. Classical design schemes find an analog prototype (low-pass) filter that satisfies prescribed filter specifications and then convert the prototype filter to the desired frequency-selective digital filter by appropriate algebraic transformations of complex variables [Oppenheim, 1989]. Bilinear transformation has been a common choice of transformation since it preserves frequency response and stability constraints of the original filter without introducing the aliasing problem. Other types of frequency selective filters as high-pass (HPF), band-pass (BPF), and band-stop (BSF) filters can be obtained from the prototype low-pass (LPF) filter by another algebraic transformations.

The IIR filter design is in fact an optimization problem of finding a rational function, which is the best approximation to the given function (filter specifications). The error surface defined by the difference between the desired and the actual frequency responses is usually multimodal with respect to the filter coefficients. In multimodal optimization problems, popular gradient-based learning algorithms may get stuck in local minima and often do not converge to the global optimum.

Genetic algorithm (GA), as a population-based random

search method, has been successful in searching for global optimum in multimodal optimization problems [Goldberg, 1988], [Michalewicz, 1994]. Key advantages of the GA-based filter design scheme include direct implementation of digital filters without algebraic transformations from analog filters to digital filters [Tang, 1996]. Other types of frequency-selective filters as high-pass, band-pass, and band-stop filters can also be designed independently without transformations from canonical low-pass filter. Several works on the application of the GA to filter design have been reported [Etter, 1982], [Nambiar, 1992], [Flockton, 1993]. Recent efforts using the hierarchical GA were successful in optimizing both IIR filter structure and coefficients [Tang, 1998] by adopting control genes as well as coefficient genes. Control genes are used to represent a filter structure, while coefficient genes correspond to real-valued filter coefficients. Binary control genes indicate which filter coefficient will be involved in genetic operation at every generation. Coefficients with 0 control gene are ignored in genetic search to reduce the filter order.

In this paper, a digital IIR filter design method using the genetic algorithm with modified crossover and mutation operations is presented. The filter transfer function is represented as a cascade form of first- and second-order rational functions. All the coefficients of a specific filter are encoded as a set of real-valued genes of a chromosome. Each chromosome includes real-valued filter coefficients, arranged in a cascade form of first- and second-order terms. The control genes, used in [8] to specify a filter structure, are not included in the chromosome to simplify evolution procedures. Filter coefficients whose absolute values are small enough are

disregarded to reduce the filter order. The IIR filter obtained from the GA must satisfy the stability constraints. In order to guarantee the stability, genetic operators are chosen to produce filter coefficients that remain in the stable region of the filter coefficients. Crossover operator is modified to generate weights according to fitness value difference to ensure faster and reliable convergence to the optimum coefficients. Mutation probability varies in time to enhance the convergence.

2. IIR Filter Design Using The Genetic Algorithm

2.1 Fitness Function for Filter Design

IIR filter design is often based on magnitude response in frequency domain. Typically filter specifications involve several design parameters as cutoff frequencies, pass-band ripple, and stop-band gain in magnitude response. Phase response is not considered in the design procedure in most cases. Figure 1 shows a block diagram that shows a general flow of IIR filter design procedure using the genetic algorithms. A digital IIR filter with a transfer function $H(z)$, modeled as a rational function of the complex variable z^{-1} , has a frequency response $H(e^{j\omega})$. The difference $\varepsilon(\omega)$ of the magnitude frequency response of the filter $H(z)$ to be optimized and the given filter specification $H_d(z)$ is defined as:

$$\varepsilon(\omega) = |H_d(e^{j\omega})| - |H(e^{j\omega})| \quad (1)$$

The magnitude response $|H_d(e^{j\omega})|$ denotes desired filter specification, and $|H(e^{j\omega})|$ indicates the magnitude response of the actual filter. The GA searches for optimal filter coefficients by maximizing the fitness function defined as a reciprocal of the error function.

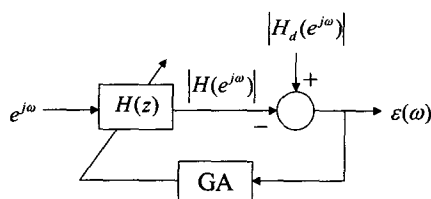


Figure 1: IIR digital filter design using the GA.

Figure 2 graphically shows the error in frequency domain produced by a digital IIR filter to be optimized. Difference between the magnitude of an actual filter response and the desired filter specifications defines the error to be minimized. If the filter response satisfies the specifications, then no error occurs. Frequency range from 0 to the normalized sampling frequency (π) is considered.

Region C indicates the frequency range that filter response does not satisfy desired filter specifications. The error in frequency domain is used to define the fitness function of genetic algorithm, a criterion function to be maximized. The fitness function f is a reciprocal of the sum of squared errors

of the pass-band and stop-band in the region C.

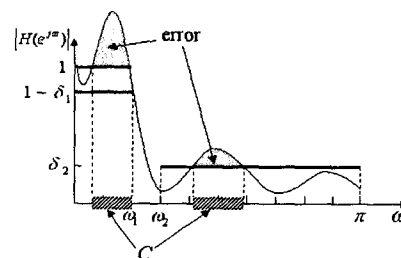


Figure 2: Error in magnitude response between the actual response and desired filter specifications

$$f = \frac{1}{1 + \sum_{\omega \in C} [\varepsilon_p^2(\omega) + \varepsilon_s^2(\omega)]} \quad (2)$$

$$\varepsilon_p(\omega) = \begin{cases} |H(e^{j\omega})| - 1 & \text{if } |H(e^{j\omega})| > 1 \\ (1 - \delta_1) - |H(e^{j\omega})| & \text{if } |H(e^{j\omega})| < (1 - \delta_1) \end{cases} \quad (3)$$

$$\varepsilon_s(\omega) = |H(e^{j\omega})| - \delta_2 \quad \text{if } |H(e^{j\omega})| > \delta_2 \quad (4)$$

where $\varepsilon_p(\omega)$ and $\varepsilon_s(\omega)$ are the errors between the filter specification and actual frequency response in the pass-band and stop-band, respectively. The total error is the sum of pass-band and stop-band errors. The parameter δ_1 defines the pass-band ripple and δ_2 indicates the stop-band gain. A chromosome with the fitness value of 1 indicates the filter that satisfies the specifications.

2.2 Encoding of Filter Coefficients

Filter coefficients are encoded in a chromosome for genetic search. Each filter coefficient refers to a gene of a chromosome. Filter transfer function can be implemented in different forms. In this paper, a cascade realization of first- and second-order factors is considered to represent the digital IIR filter model. Transfer function of a digital IIR filter represented in a cascade first- and second-order terms is shown as

$$H(z) = K \prod_{i=1}^P \left(\frac{1 + b_{i1}z^{-1}}{1 + a_{i1}z^{-1}} \right) \prod_{j=1}^Q \left(\frac{1 + b_{j1}z^{-1} + b_{j2}z^{-2}}{1 + a_{j1}z^{-1} + a_{j2}z^{-2}} \right) \quad (5)$$

The variable K denotes the filter gain. The denominator coefficients $a_i, a_{i1},$ and a_{i2} are associated with poles, while b_i, b_{i1}, b_{i2} are numerator polynomial coefficients associated with zeros. The filter model assumes P first-order terms and Q second-order terms in its transfer function. When $P = 0$, the filter is composed of the second-order terms only, and first-order components if $Q = 0$. The overall filter order becomes $P+2Q$.

A chromosome is composed of a series of genes. Each gene corresponds to a filter coefficient of the first- and second-order terms of the transfer function. The order of a filter determines the number of genes in a chromosome.

Figure 3 shows structure of the chromosome, or an encoding scheme of filter coefficients. The first part of the chromosome contains denominator polynomial coefficients, while the second part are numerator polynomial coefficients. The maximum values for P and Q determine the number of genes in a chromosome. Filter gain can be determined with the condition of the magnitude response at the zero frequency, and is not encoded in a chromosome. All the filter coefficients are encoded as real-valued genes.

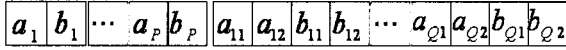


Figure 3: Chromosome representation of the IIR filter coefficients

Given the maximum order of an IIR filter, there are different combinations of the first- and the second-order terms in order to construct a filter. For example, a 5th order filter can have three combinations of the terms: two second-order terms and one first-order term, one second-order term and three first-order terms, and five first-order terms. In this paper, second-order terms are assumed to have priority against the first-order terms.

2.3 Genetic Operations for Optimizing Filter Coefficients

Optimization by the genetic algorithms is carried on using selection, genetic operations, and fitness evaluation. A set of chromosomes generated randomly forms an initial population filters. All the chromosomes in population are chosen to satisfy the stability constraints. Selection procedure finds a pool of individuals for genetic search using genetic operators. Popular choice of selection is the roulette wheel method that selects individuals with probability in proportion to the fitness value. Individuals with larger fitness value have higher chance to be selected for further genetic operations.

Genetic operations must be chosen to produce filter coefficients that ensure stability of the IIR filter. In order to guarantee the designed filter is stable, the poles associated with the denominator polynomial coefficients must be inside the unit circle of the z -plane. Crossover and mutation operations are considered to evolve new filter coefficients. Crossover operator, based on the good building block hypothesis, generates individuals with higher fitness value from individuals in the previous generation. Mutation operation produces random variations of the coefficients for effective genetic search.

Consider two chromosomes and drawn from a population. The two chromosomes contain first-order coefficients $a_i^{(x)}$, $a_i^{(y)}$, and second-order coefficients $a_{j1}^{(x)}$, $a_{j2}^{(x)}$, $a_{j1}^{(y)}$, and $a_{j2}^{(y)}$ of denominator polynomials. To satisfy the stability constraints of IIR filters, the coefficients of the second-order term must be inside the stability triangle determined by $a_{j1} < a_{j2} + 1$, $a_{j1} > -(a_{j2} + 1)$, and $-1 < a_i < 1$. (See Appendix) Genetic operation must ensure that the individuals after genetic operations are still in the

stable region.

The crossover operator used in [Tang, 1998] guarantees stability after the crossover operation. For a pair of filter coefficients, if a random number falls inside the crossover probability, then a new filter coefficient a_i is produced from the two coefficients $a_i^{(x)}$ and $a_i^{(y)}$ in the mating pool as in Equation (6). For the coefficients of the second-order term, Equation (7) applies.

$$a_i = \alpha a_i^{(x)} + (1 - \alpha) a_i^{(y)} \quad (6)$$

$$a_{jk} = \alpha a_{jk}^{(x)} + (1 - \alpha) a_{jk}^{(y)} \quad (7)$$

The value of the parameter α is randomly generated in the interval $[0,1]$. Since the stability triangle is a convex region, the constant α in the range $[0,1]$ guarantees that the new filter coefficients fall in the stable region as well after the crossover operation. In a geometrical point of view, this means that a new filter coefficient a_i generated from the crossover operation is on the line connecting the two filter coefficients $a_i^{(x)}$ and $a_i^{(y)}$ to be crossed over. In this paper, α was chosen depending on the value of difference of the fitness value to put weights for faster convergence.

$$\alpha(\Delta f) = \frac{1}{2} \Delta f + \frac{1}{2} \quad (8)$$

where $\Delta f = f^{(x)} - f^{(y)}$ and $-1 < \Delta f < 1$. For example, if $f^{(x)} = 1$ and $f^{(y)} = 0$, then the value $\alpha = 1$ ensures $a_i = a_i^{(x)}$, the one with higher fitness value. Likewise, if $\Delta f = -1$, then $a_i = a_i^{(y)}$. This improves convergence speed comparing with the case of random constant α .

Mutation probability P_m decreases to improve the search convergence of the genetic algorithm.

$$P_m(k) = \frac{0.2e^{-(k_0 - k)/k}}{1 + e^{-(k_0 - k)/k}} \quad (9)$$

where k denotes generation number and $k_0 = 100$. Figure 4 shows the trend of mutation probability in generations. Early stage of evolution takes mutation with probability close to 0.2, which has a relatively large influence on genetic search. Mutation probability gradually decreases, yet maintains approximately 0.05 for improving search performance.

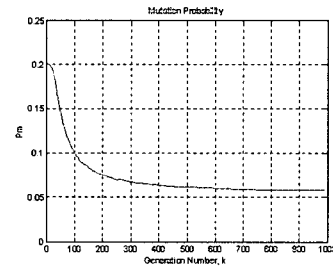


Figure 4: Trend of the mutation probability

In crossover operation, performance improved by assigning weights to individuals with higher fitness value. Mutation

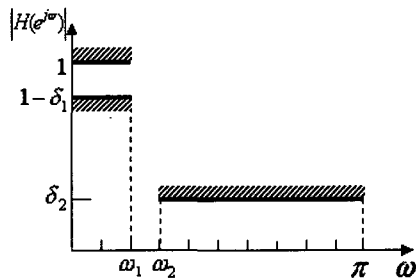
probability changes in generation to improve convergence. Elitist method ensures individual of largest fitness to exist in the following generation to reduce convergence time.

3. Results

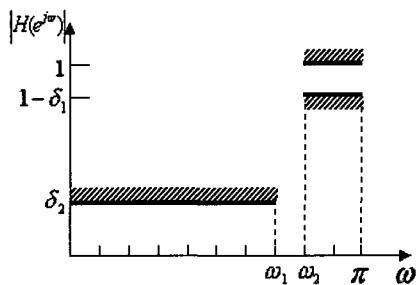
Design of different types of filters LPF, HPF, BPF, and PSF using the genetic algorithm is demonstrated. Figure 5 shows frequency response specifications for each of the frequency-selective filters. Two parameters δ_1 and δ_2 determine the pass-band ripple and the stop-band margin. In the simulation, the pass-band ripple is set to -1.5dB ($\delta_1=0.1586$) and the stop-band gain is -30dB ($\delta_2=0.0361$). For the low-pass filter, pass-band is defined as $[0, \omega_1]$ and stop band is $[\omega_2, \pi]$. Frequency bands $[\omega_1, \omega_2]$ and $[\omega_3, \omega_4]$ are called transition bands. Table 1 shows specifications for different types of frequency-selective filters. Transition bands are all set to 0.1π .

Table 1: Specifications for different types of frequency-selective filters

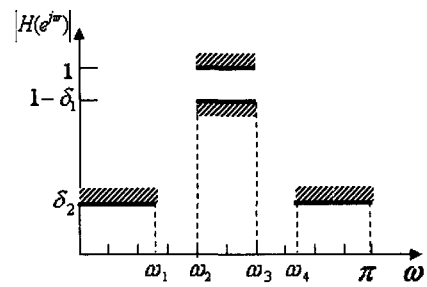
Type	ω_1	ω_2	ω_3	ω_4
LPF	0.2π	0.3π	-	-
HPF	0.7π	0.8π	-	-
BPF	0.2π	0.3π	0.7π	0.8π
BSF	0.2π	0.3π	0.7π	0.8π



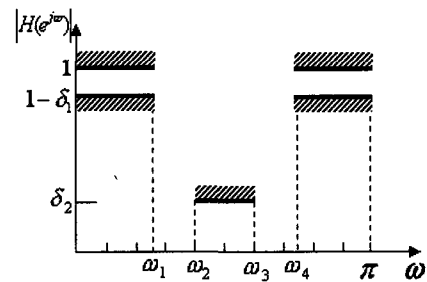
(a) LPF



(b) HPF



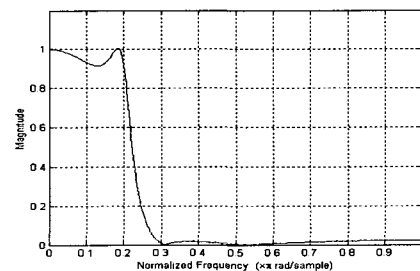
(c) BPF



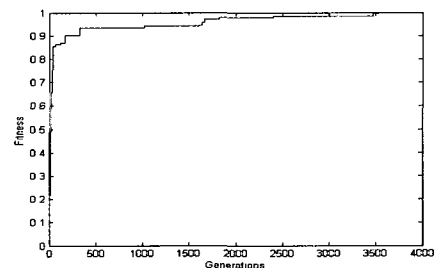
(d) BSF

Figure 5: Frequency-selective filter specifications

The parameters used in the GA-based filter design scheme are: population=100, crossover probability = 0.8, and mutation probability = p_m . The maximum number of generations is 4000 for LPF and 3000 for HPF. For BPF and BSF, the filter converges after approximately 12000 generations. Crossover and mutation probabilities affect convergence of the GA. Gain K is determined to satisfy $|H(e^{j\omega})| = 1$ for LPF, $|H(e^{j\pi})| = 1$ for HPF, $|H(e^{j\pi/2})| = 1$ for BPF, and $|H(e^{j\omega})| = |H(e^{j\pi})| = 1$ for BSF.



(a) Magnitude response

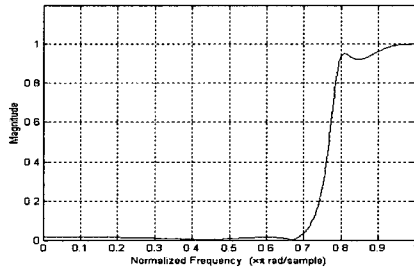


(b) Fitness curve

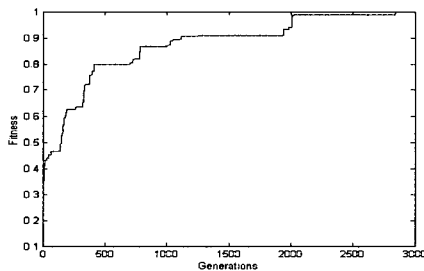
Figure 6: Low-pass digital IIR filter using the GA

Figure 6(a) shows the magnitude frequency response of the low-pass filter obtained using the proposed genetic optimization scheme. Figure 6(b) is a fitness trend during evolution in the design procedure.

From Figure 7 to Figure 9, the proposed IIR filter design method produced HPF, BPF, and BSF directly from the filter specifications. All the magnitude responses of the IIR filter designed by the genetic algorithm satisfy the filter specifications.

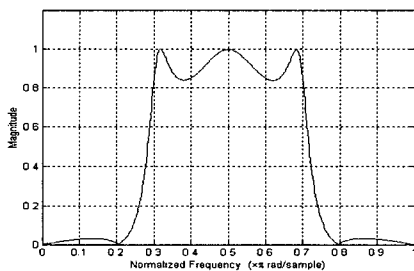


(a) Magnitude response

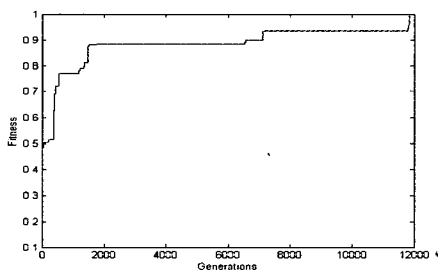


(b) Fitness curve

Figure 7: High-pass digital IIR filter using the GA

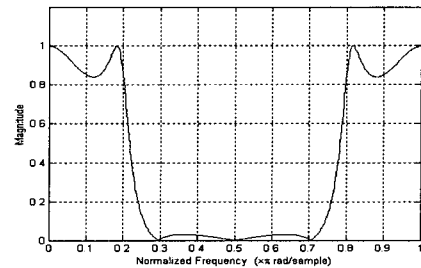


(a) Magnitude response

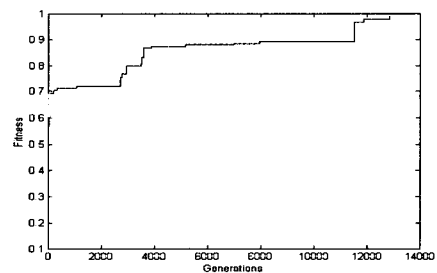


(b) Fitness curve

Figure 8: Band-pass digital IIR filter using the GA



(a) Magnitude response



(b) Fitness curve

Figure 9: Band-stop digital IIR filter using the GA

Equations (10)-(13) show the transfer functions of the corresponding LPF, HPF, BPF, and BSF designed using the GA, respectively.

$$H_{LP}(z) = 0.0469 \frac{(1 + 0.6135z^{-1} + 0.7898z^{-2})}{(1 - 1.3772z^{-1} + 0.7964z^{-2})} \cdot \frac{(1 - 1.0345z^{-1} + 0.9577z^{-2})}{(1 - 1.0474z^{-1} + 0.2957z^{-2})} \quad (10)$$

$$H_{HP}(z) = 0.0369 \frac{(1 - 0.3350z^{-1} + 0.7093z^{-2})}{(1 + 1.2577z^{-1} + 0.4389z^{-2})} \cdot \frac{(1 + 0.9909z^{-1} + 0.9576z^{-2})}{(1 + 1.4197z^{-1} + 0.8223z^{-2})} \quad (11)$$

$$H_{BP}(z) = 0.1167 \frac{(1 + 0.0012z^{-1} - 0.9988z^{-2})}{(1 - 0.0005z^{-1} + 0.4818z^{-2})} \cdot \frac{(1 + 1.6001z^{-1} + 1.0018z^{-2})}{(1 + 1.0625z^{-1} + 0.8458z^{-2})} \cdot \frac{(1 - 1.5995z^{-1} + 1.0011z^{-2})}{(1 - 1.0622z^{-1} + 0.8459z^{-2})} \quad (12)$$

$$H_{BS}(z) = 0.1178 \frac{(1 + 0.2284z^{-1} - 0.8889z^{-2})}{(1 + 1.4344z^{-1} + 0.8079z^{-2})} \cdot \frac{(1 + 0.8345z^{-1} + 0.9601z^{-2})}{(1 + 0.0273z^{-1} - 0.4213z^{-2})} \cdot \frac{(1 - 0.9853z^{-1} + 0.9764z^{-2})}{(1 - 1.4050z^{-1} + 0.7547z^{-2})} \quad (13)$$

For the purpose of comparison, conventional methods of digital IIR filter design were implemented using the MATLAB filter design toolbox [Krauss,1994] for different choices of prototype filters, Butterworth (BW), Chebyshev (CH), and

Elliptic (EL) filters. Table 2 summarizes the filter order obtained for different filter types. The GA-based design method achieved the same order as in the case of the elliptic filters, which is the lowest.

Table 2: Filter order obtained for different filter types

Type	BW	CH I	CH II	EL	GA
LPF	9	5	5	4	4
HPF	9	5	5	4	4
BPF	14	8	8	6	6
BSF	14	8	8	6	6

4. Conclusion

Design of digital IIR digital filters using the genetic algorithm with modified genetic operators is discussed. Filter coefficients of the transfer function in cascade form of first- and second-order terms are encoded as real-valued genes in a chromosome. Chromosomes contain only coefficient genes, not the control genes, for simplicity of genetic operations. Coefficients whose value is sufficiently small are disregarded to reduce the filter order. The fitness function is defined using the sum of squared error in the desired and the actual frequency response. Maximizing the fitness function corresponds to minimizing the error.

Initial population of filter coefficients is chosen to satisfy stability condition for IIR filters. Genetic search produces a population of candidate filters in every generation through genetic operations. Crossover and mutation operators ensure that filter coefficients generated from the genetic operations must satisfy stability constraints as well. Crossover operator is modified to have different weights according to fitness value difference to ensure faster and reliable convergence to the optimum coefficients. Mutation probability changing in time improves the genetic search. The proposed filter design scheme using the genetic algorithm produced LPF and other types of frequency-selective filters directly and independently without algebraic transformations. Comparing with classical design methods of IIR filters, the IIR filter design based on the genetic optimization achieved lowest filter order.

Appendix:

Stability Constraints of Digital IIR Filters

Transfer function of a digital IIR filter can be represented in a cascade form of first- and second-order terms. The cascade realization of the IIR filters has the following form:

$$H(z) = K \prod_{i=1}^p \left(\frac{1 + b_{i1}z^{-1}}{1 + a_{i1}z^{-1}} \right) \prod_{j=1}^q \left(\frac{1 + b_{j1}z^{-1} + b_{j2}z^{-2}}{1 + a_{j1}z^{-1} + a_{j2}z^{-2}} \right) \quad (A1)$$

Filter coefficients associated with poles must satisfy the stability constraints in order to ensure the desired IIR filter will be stable in the bounded-input bounded-output (BIBO) sense. This can be achieved by noting that all the poles must

be located inside the unit circle in z-plane.

Consider the first-order term of the filter whose transfer function is represented as

$$H_1(z) = \frac{1 + b_1z^{-1}}{1 + a_1z^{-1}} \quad (A2)$$

There is a pole at . In order to have the pole inside the unit circle, the filter coefficient must satisfy the condition $|a_1| < 1$.

Second-order terms of the IIR filter have transfer functions of the form

$$H(z) = \frac{1 + b_{11}z^{-1} + b_{12}z^{-2}}{1 + a_{11}z^{-1} + a_{12}z^{-2}} \quad (A3)$$

Let be the poles associated with this second-order term. Then

$$P_1, P_2 = \frac{-a_{11} \pm \sqrt{a_{11}^2 - 4a_{12}}}{2} \quad (A4)$$

The two poles must be inside the unit circle, which gives

$$-2 < -a_{11} \pm \sqrt{a_{11}^2 - 4a_{12}} < 2 \quad (A5)$$

Since the coefficient a_{12} is simply the product of the two poles, the coefficient a_{12} must satisfy the condition $|a_{12}| < 1$ for stability. For real roots, $a_{11} < a_{12} + 1$ for $a_{11} < 2$ and $a_{11} > -(a_{12} + 1)$ for $a_{11} > -2$. In case of double roots, $-2 < a_{11} < 2$. For complex roots, $a_{11}^2 < 2a_{12} + 2$. Figure A-1 shows the common region of coefficients a_{11} and a_{12} of the transfer function $H(z)$ for stability of the second-order terms of IIR filters. The region is called the stability triangle.

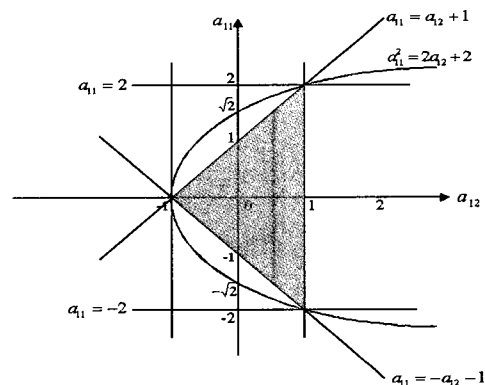


Figure A-1: Region of the filter coefficients that guarantees stability of the second-order IIR filter (the stability triangle)

References

[1] A. V. Oppenheim and R. W. Schaffer, *Discrete-Time Signal Processing*, Prentice Hall, 1989.
 [2] D. E. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison Wesley, 1989.

- [3] Z. Michalewicz, *Genetic Algorithms + Data Structures = Evolutionary Programs*, Springer-Verlag: Berlin, Heidelberg, 1994.
- [4] D. M. Etter, M. J. Hicks, and K. H. Cho, "Recursive Adaptive Filter Design using an Adaptive Genetic Algorithm," *Proc. of IEEE Int. Conf. on ASSP*, pp.635-638, 1982.
- [5] R. Nambiar, C. K. Tang, and P. Mars, "Genetic and learning automata algorithms for adaptive digital filters," *Proc. IEEE Int. Conf. ASSP*, Vol. IV, pp.41-44, 1992.
- [6] S. J. Flockton and M. S. White, "Pole-zero identification using genetic algorithms," *Proc. 5th Int. Conf. on Genetic Algorithms*, pp.531-535, 1993.
- [7] K. S. Tang, K. F. Man, S. Kwong, and Q. He, "Genetic Algorithms and their Applications," *IEEE Signal Processing Magazine*, pp.22-37, 1996.
- [8] K. S. Tang, K. F. Man, S. Kwong, and Z. F. Liu, "Design and Optimization of IIR Filter Structure Using Hierarchical Genetic Algorithms," *IEEE Trans. on Industrial Electronics*, Vol. 45, No. 3, pp.481-487, 1998.
- [9] T. P. Krauss, L. Shure, and J. N. Little, *MATLAB Signal Processing Toolbox*, Mathworks Inc., 1994.
- [10] 10. S. Chen, R. Istepanian, and B. L. Luk, "Digital IIR filter design using adaptive simulated annealing," *Digital Signal Processing*, Vol. 11, No. 3, 241-251, July 2001.

Jung-Doo Jang received the B.S. degree in electrical engineering from Soongsil University in 1998. He completed his M.S. degree in electrical engineering at Soongsil University in 2001. He has been with Intelligent Signal Processing laboratory. His research interests are digital signal processing and evolutionary algorithms. He is a member of KIEE and KFIS.



Seong G. Kong received the B.S. and the M.S. degrees in electrical engineering from Seoul National University, Korea, in 1982 and 1987. He completed the Ph.D. degree in electrical engineering at University of Southern California, Los Angeles, California, in 1991. From 1992 to 2001, he was with Department of Electrical Engineering at Soongsil University, Seoul, Korea as an Associate Professor. He was a visiting researcher at the Technical University of Aachen (RWTH Aachen), Germany, in 1997 and 2000. He served as the editor-in-chief of the Journal of Fuzzy Logic and Intelligent Systems from 1996 to 1999. From 2000 to 2001, he was with the School of Electrical and Computer Engineering at Purdue University, West Lafayette, Indiana, as a visiting scholar. He is currently an Associate Professor of Electrical and Computer Engineering at the University of Tennessee, Knoxville. He is a member of IEEE, SPIE, KIEE, IEEK, and KFIS. Dr. Kong's research interests include image processing and intelligent systems.