

Evrimsel Algoritmalar Kullanarak Daha Düşük Dereceden Sistem Modeli Tasarımı

Design of Lower Order System Model Using Evolutionary Algorithms

¹Yiğit Çağatay Kuyu and *¹Fahri Vatansever

¹Faculty of Engineering, Electrical-Electronics Eng. Dept., Uludag University, Turkey

Özet:

Sistem modelleme, araştırmacılar tarafından kontrol ve sinyal işlemenin çeşitli alanlarında yaygın olarak kullanılmaktadır. Sistemler sayısal sonsuz darbe cevaplı filtreler (IIR) kullanılarak modellenmektedir. IIR filtrenin, sistem modellemede genellikle çokmodlu ve nonlineer hata yüzeyine sahip olması geleneksel yöntemlerden birçoğunun yetersiz kalmasına sebep olmaktadır. Geleneksel yöntemlerin bu dezavantajları, evrimsel algoritmalarla giderilebilmektedir. Bu çalışmada literatürde geçen iki farklı sistem, sıklıkla kullanılan guguk kuşu arama, differansiyel gelişim, genetik ve parçacık sürü optimizasyon algoritmalarıyla daha düşük dereceli adaptif IIR filtreler ile modellenmekte ve kararlılıkları, ilgili modellerin sıfır-kutup diyagramı ile doğrulanmaktadır.

Key words: Sistem modelleme, evrimsel algoritmalar, IIR filtre

Abstract:

System modelling has been widely used by the researchers in various fields of control and signal processing. The systems can be modelled by using digital infinite impulse response (IIR) filters. Most of the conventional methods can be insufficient when modelling the system because of the error surface of IIR filter which is usually nonlinear and multimodal. In order to overcome the conventional methods' drawbacks, evolutionary algorithms (EAs) are developed. In this work, two different systems in the literature have been modelled with lower order adaptive IIR filters by using most recently used EAs which are cuckoo search, differential evolution, genetic and partical swarm optimization algorithms and stability of the lower order IIR models designed by the EAs have been validated with the pole zero diagram of associated model.

Key words: System modelling, evolutionary algorithms, IIR filter

1. Giriş

Sistem modelleme, bir sistemin zaman veya frekans uzayında verilen giriş verisine karşı elde edilen çıkış verisinin matematiksel olarak tanımlanmasıdır[1-2]. Bu matematiksel modellerde derecenin artmasıyla, genellikle doğruluk artarken işlemlerin sayısı ve karmaşıklığı da artmakta ve pratik veya gerçek zamanlı uygulamalar için olumsuz etkiler oluşturabilmektedir. Dolayısıyla aynı özellikleri taşıyan, daha düşük dereceden sistem modelleri oluşturmak için alternatif yöntemler geliştirilmiştir. Bu yöntemlerden bazılarında adaptif filtrelerden faydalanılmaktadır. Adaptif filtreler, istenen bir işarete, çeşitli algoritmalar yardımıyla kendini uydurabilen/uyarlayabilen filtrelerdir. Evrimsel algoritmalar yardımıyla modelleme işlemi,

*Corresponding author: Address: Faculty of Engineering, Department of Electrical-Electronics Engineering, Uludag University, 16059, Bursa, TURKEY. E-mail address: fahriv@uludag.edu.tr, Phone: +902242940905

sayısal adaptif IIR filtre yapısı kullanılarak gerçekleştirilebilmektedir. Sayısal filtre tasarımında kararlılık, sayısal filtrelerin pratikte kullanımı açısından önemli bir kavramdır. Bu nedenle tasarlanan IIR sayısal filtrelerin kararlı olmasına ihtiyaç duyulmaktadır. Evrimsel algoritmalar, matematiksel olarak ifade edilen bir hedefe yaklaşan, doğal olaylardan esinlenerek oluşturdukları yöntemlere dayanan, kendine özgü hesaplama teknikleri içeren algoritmalarlardır. Bu algoritmalar bir çözüm adayı popülasyonu oluşturarak istenen çözüme ulaşmak için bu popülasyonu zamanla evrimleştirmektedir.

Evrimsel algoritmalar kullanılarak farklı özellikteki sistemlerin daha düşük dereceden modellenmesiyle ilgili çalışmalar literatürde mevcuttur. Gaytman ve arkadaşları, parçacık sürü optimizasyonu algoritmasını (PSO) kullanarak, sekizinci dereceden kararlı bir sistemin daha düşük dereceli modellerini oluşturmuşlardır[3]. Yadav ve arkadaşları, ayrık zaman tek giriş tek çıkış dördüncü dereceden sistemin, ikinci dereceden modelini diferansiyel gelişim algoritması (DE) kullanarak elde etmişlerdir[4]. Ma ve Cowan yaptıkları çalışmada genetik algoritma (GA) ile adaptif IIR filtre yapısını kullanarak üç farklı örnek sistem üzerinden, daha düşük dereceli modeller elde etmişlerdir[5]. Adaptif IIR filtrelerin çokmodlu ve lineer olmayan hata yüzeyine sahip olması, yüksek dereceli sistemlerin düşük dereceli modellerinin daha fazla hesaplama karmaşası içermesi, sistem modellemek için yeni algoritmalara ihtiyacı doğurmuştur. Dai ve arkadaşları adaptif IIR filtreleri kullanarak daha düşük dereceli sistemleri modellemek için, yeni bir algoritma olan arayıcı optimizasyon algoritmasını geliştirmişlerdir. Bu çalışmada literatürde geçen yedi farklı sistem modellenmektedir. Bulunan hata değerleri, bu algoritmanın karşılaştırdığı diğer algoritmalarından daha iyi veya birbirine çok yakın düşük dereceli sistemi modelleyebildiğini göstermiştir[6]. Bir diğer yeni optimizasyon algoritması, Panda ve arkadaşları tarafından IIR dinamik sistem modellemede kullanılan kedi sürüsü optimizasyon algoritmasıdır. Bu algoritma kedilerin doğadaki davranışlarından esinlenerek geliştirilmiştir. Bu algoritmayla tasarlanan daha düşük dereceli sistemler, bu algoritmanın karşılaştırıldığı üç farklı optimizasyon algoritmasından daha iyi sonuçlar ortaya koyduğunu göstermiştir[7]. Sistem modelde, algoritmalarının geliştirilmiş versiyonlarına da rastlamak mümkündür. Bu yeni geliştirilmiş versiyondaki algoritmalar, geliştirildikleri algoritmaya kıyasla daha başarılı sonuçlar elde etmektedir[8,9].

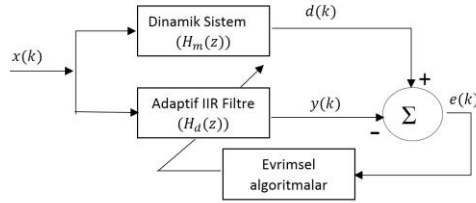
Bu çalışmada, birçok alanda sıklıkla kullanılan GA, PSO, DE ve guguk kuşu arama (CS) algoritmaları kullanılarak iki ve üçüncü dereceden sistemlerin daha düşük dereceden IIR filtreler vasıtasıyla modellenmesi gerçekleştirilmiş ve kararlılıkları incelenmiştir. Makalenin ikinci bölümünde problemin tanımı, üçüncü bölümünde çalışmada kullanılan algoritmalar hakkında bilgiler verilmiş olup, dördüncü bölümde uygulama örnekleri sunulmaktadır.

2. Problemin Tanımı

IIR filtreler, sonsuz darbe cevaplı geri beslemeli filtrelerdir. IIR filtrelerin sabit katsayılı, giriş ve çıkışına bağlı eşitliği Denklem 1'de verilmektedir.

$$y(k) + \sum_{j=1}^N b_j y(k-j) = \sum_{j=0}^M a_j x(k-j) \quad (1)$$

Bu denklemde a ve b , filtre katsayılarını; $N (\geq M)$, filtrenin derecesini belirtmektedir[10]. Adaptif IIR filtreler kullanılarak, dinamik sistem modellemenin blok diyagramı Şekil 1’de gösterilmektedir.



Şekil 1. Sistem modelleme blok diyagramı

Evrimsel algoritmalar, IIR filtre aracılığıyla tanımlanacak daha düşük dereceli transfer fonksiyonu $H_d(z)$ ve tanımlanan dinamik sistem modelinin transfer fonksiyonu $H_m(z)$ arasındaki ilişkiye bağlı olarak, daha düşük dereceli sistem modeli tasarlanmaktadır. $x(k)$; filtrenin girişine uygulanan beyaz gürültü işaretini, $y(k)$; filtrenin çıkış işaretini temsil etmektedir. Hata fonksiyonu, ortalama karesel hata alınarak, Denklem 2’de ifade edilmektedir.

$$J = \frac{1}{L} \sum_{k=1}^L (d(k) - y(k))^2 \quad (2)$$

$d(k)$; filtrenin istenen cevabını, L ; örnek sayısını, J ; minimize edeceğimiz hata fonksiyonunu temsil etmektedir.

3. Evrimsel Algoritmalar

3.1. Diferansiyel gelişim algoritması

Price ve Storn tarafından 1995 yılında geliştirilen DE, populasyon temelli evrimsel bir optimizasyon tekniğidir. Kromozomların oluşturduğu populasyonda, kromozomlara algoritmaya özgü çaprazlama ve mutasyon işlemlerini uygulanarak yeni çözümler üretilir.

$$Y_{j,i,g+1} = X_{j,r_3,g} + F(X_{j,r_1,g} - X_{j,r_2,g}) \quad (3)$$

Denklem 3’te, $Y_{j,i,g+1}$; mutasyon işlemi sonucu oluşan ara kromozomu, $X_{j,r_1,g}$, $X_{j,r_2,g}$ ve $X_{j,r_3,g}$; mutasyon işlemine tabi tutulacak kromozom dışında ve birbirlerinden farklı olan üç kromozomu ve F ; ağırlıklandırılmış ölçekleme faktörünü ifade eder. Daha sonra çaprazlama işlemine geçilir. Bu işlemde, $Y_{j,i,g+1}$ ve $X_{j,i,g}$ kromozomları kullanılarak Denklem 4’te verilen $U_{i,j,g+1}$ deneme kromozomu elde edilir.

$$U_{i,j,g+1} = \begin{cases} Y_{j,i,g+1} & \text{eğer } rand[0,1] \leq CR \\ X_{j,i,g} & \text{diğer durumda} \end{cases} \quad (4)$$

Deneme kromozomuna ait her bir gen CR olasılıkla ara kromozomdan, $1 - CR$ olasılıkla mevcut kromozomdan seçilir. Hedef fonksiyon uygunluk değerlerine bağlı olarak, mevcut kromozom

$X_{j,i,g}$ ile deneme kromozomu $U_{i,j,g+1}$ karşılaştırılarak daha iyi uygunluk değerine sahip kromozom yeni kromozom jenerasyonuna katılır[11].

3.2. Genetik algoritma

GA, çözüme ulaşmak için evrim prensiplerine dayanan ve kendi yapısına özgü işlemleri kullanır. Kromozomlar olarak adlandırılan populasyon bireyleri, problemin olası çözümüne karşılık gelen genetik yapıyı temsil eder. Genler, kromozomları oluşturan vektörlerin her bir boyutuna gelen karşılıklarıdır. GA'da başlangıçta rastgele dizilime sahip kromozomlar oluşturulur. Ardından her bir kromozomun performansının, diğer kromozomlara göre değerlendirilmesi yapılır. Kromozomların çözüm arama sürecinde, daha iyi çözümleri bulmasını sağlamak amacıyla genetik operatörler kullanılır. Kullanımı en çok tercih edilen operatörler; seçme, çaprazlama ve mutasyon operatörleridir. Seçme operatörü, en uygun bireyin hayatta kalması; çaprazlama operatörü, bireylerin genetik bilgilerinin yeniden birleştirilerek bireylerdeki genetik çeşitliliği sağlama; mutasyon operatörü, daha iyi bir çözüme ulaşmak amacıyla yeni bireylerin elde edilmesi prensibine dayanır. Problemin parametrelerinin ikili kodlanmış sayı ya da sayı grubunu kullanan GA; iterasyon veya fonksiyon hesaplama sürecinde, genetik operatörleri vasıtasıyla en uygun çözüme yakınsayarak kendi çözümüne ulaşır[12].

3.3. Guguk kuşu arama algoritması

CS, Xin-She Yang ve Suash Deb tarafından, guguk kuşu türlerinden ilham alınarak geliştirilen optimizasyon algoritmasıdır. Algoritmanın Levy uçuşunu kullanarak, yeni bireyler üretme işlemi Denklem 5'deki gibi elde edilir.

$$X_i^{t+1} = X_i^t + \alpha L(s, \lambda) \quad (5)$$

X_i^t ; i. guguk kuşunun o anki bulduğu çözümü, X_i^{t+1} ; i. guguk kuşunun ürettiği yeni çözümü, $L(s, \lambda)$; guguk kuşunun Levy uçuşunu ve $\alpha (> 0)$ adım büyüklüğü ölçekleme faktörünü ifade etmektedir. Yeni çözümler üretilmesi aşamasında, guguk kuşları $p_a \in (0,1)$ olasılığında en kötü çözümleri terk ederek yerine yenilerini inşa eder. Guguk kuşlarının çözüm arayışı işleminin bitmesinin ardından hedef fonksiyon uygunluk değerlerine bağlı olarak bulduğu çözümler arasındaki en iyi çözüm, optimal çözüm olarak adlandırılır[13].

3.4. Parçacık sürü optimizasyonu algoritması

Kennedy ve Eberhart tarafından geliştirilen PSO tekniğinde her bir çözüm bir parçacıkla temsil edilir ve problem parametrelerine bağlı olarak belirli boyutlara sahiptirler. Her bir parçacık, olası çözümler için çözüm uzayını ararken Denklem 6 ve 7'yi kullanarak konum ve hız bilgilerini günceller.

$$V_i^{t+1} = V_i^t + c_1 r_1^t (p_{best_i}^t - X_i^t) + c_2 r_2^t (g_{best_i}^t - X_i^t) \quad (6)$$

$$X_i^{t+1} = X_i^t + V_i^{t+1} \quad (7)$$

Bu denklemlerde, p_{best} ; parçacığın daha önceden bulunmuş olduğu en iyi konum bilgisini, g_{best} ; popülasyonda bulunan tüm parçacıklar arasındaki en iyi parçacığın konum bilgisi, V ; parçacıkların hızlarını, X ; parçacıkların konumlarını, c_1 ve c_2 ; ivmelendirme faktörlerini, r ve r_2 ; $[0,1]$ aralığından seçilen rastgele sayıları ifade eder. Algoritma, t . iterasyon sonunda i . parçacığın konum ve hız bilgilerini Denklem 6 ve 7'yi kullanarak günceller ve daha iyi çözümlere ulaşmayı amaçlar[14].

4. Evrimsel Algoritmalar ile Sistem Modelleme Uygulamaları

Bu çalışmada, evrimsel algoritmalar ile sistem modelleme uygulaması, CS, DE, GA ve PSO algoritmalarıyla, iki farklı örnek model üzerinden ele alınacaktır. Literatürden seçilen[6-9], iki ve üçüncü dereceden farklı sistemler, bir ve ikinci dereceden sayısal IIR filtrelerle modelleneyecektir. Her bir algoritma 25 bağımsız çalıştırılmaya tabi tutulmuş, algoritmaların en iyi çalıştırılmasına ait yakınsama grafikleri karşılaştırmalı olarak verilmiş, buldukları en iyi hata, ortalama hata, standart sapma değerleri ve katsayıları tablolarda gösterilmiş ve analiz edilmiştir. Algoritmalar tarafından tasarlanan daha düşük dereceli sistemlerin kararlılıkları kutup sıfır diyagramı aracılığıyla doğrulanmıştır.

Uygulama 1: Denklem 8 ile verilen ikinci dereceden sistemin, Denklem 9'da verilen birinci dereceden IIR filtre ile modellenmesi.

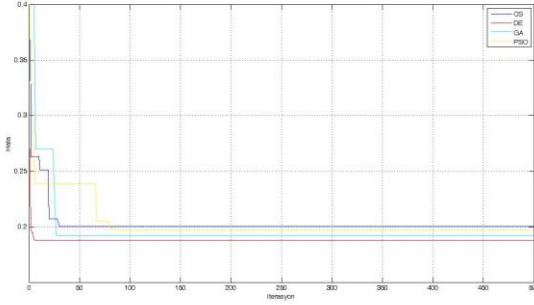
$$H_m(z) = \frac{0.05 - 0.4z^{-1}}{1 - 1.1314z^{-1} + 0.25z^{-2}} \quad (8)$$

$$H_d(z) = \frac{a_0}{1 + b_1z^{-1}} \quad (9)$$

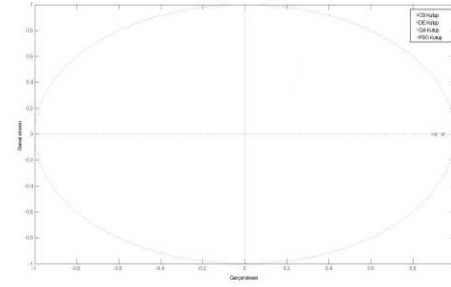
Tablo 1'de, algoritmalar tarafından birinci dereceden IIR filtre vasıtasıyla modellenen sistemin katsayıları ve algoritmaların bulunduğu hata değerleri verilmektedir. Tablo 1'den görüleceği üzere DE, en düşük hataya ulaşarak, sistem üzerinde en iyi modellemeyi yapabilmektedir. Fakat standart sapmalar analiz edildiğinde; DE algoritmasının, CS ve PSO algoritmalarına göre her bir çalıştırma için yüksek hata sapmalarıyla modelleme yapabileceği görülmektedir. Şekil 2'de algoritmaların en iyi hata değerlerine yakınsama grafiği verilmektedir. Bu grafikten, DE algoritmasının hem algoritmalar arasındaki en iyi hata değerine hem de en hızlı yakınsama performansına (yaklaşık 10 iterasyon) sahip olduğu görülmektedir. Şekil 3'te evrimsel algoritmalar tarafından modellenen birinci dereceden sistemin sıfır kutup diyagramı verilmektedir. Modellenen sistemin tüm kutupları birim çember içinde olduğundan dolayı, modellenen sistemin kararlı olduğunu söylenebilmektedir. Diğer yandan, en iyi hata değerlerine sahip DE algoritması ve PSO algoritması arasındaki hata farkı (0.01) göz ardı edilirse, PSO algoritmasının düşük standart sapma değerine sahip olması sebebiyle her farklı çalıştırma için daha stabil bir modelleme yapabileceği söylenebilmektedir. GA, hem hata hem de sapma değerleri bakımından algoritmalar arasında en kötü sonuca ulaştığından, başarılı bir modelleme sağlamamaktadır.

Tablo 1. Modellenen sisteme ait algoritmaların bulunduğu katsayı ve en iyi hata değerleri

Katsayılar	CS	DE	GA	PSO
a_0	-0.299	-0.306	-0.296	-0.343
b_0	1	1	1	1
b_1	-0.944	-0.901	-0.894	-0.908
En iyi hata	0.200	0.187	0.192	0.197
Ortalama hata	0.200	0.202	0.337	0.197
Standart sapma	4.637×10^{-17}	0.074	0.089	5.374×10^{-17}



Şekil 2. Modellenen sisteme ait algoritmaların yakınsama grafiği



Şekil 3. Modellenen sisteme ait sıfır kutup diyagramı

Uygulama 2: Denklem 10 ile verilen üçüncü dereceden sistemin, Denklem 11'de verilen ikinci dereceden IIR filtre ile modellenmesi.

$$H_m(z) = \frac{-0.2 - 0.4z^{-1} + 0.5z^{-2}}{1 - 0.6z^{-1} + 0.25z^{-2} - 0.2z^{-3}} \quad (10)$$

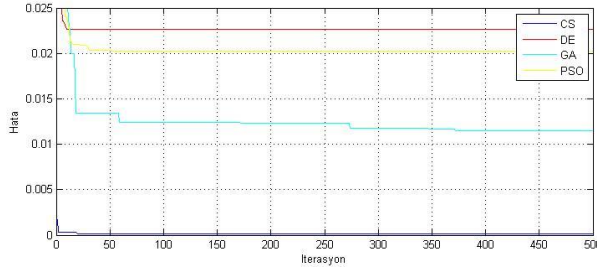
$$H_d(z) = \frac{a_0 + a_1z^{-1}}{1 + b_1z^{-1} + b_2z^{-2}} \quad (11)$$

Algoritmalar tarafından ikinci dereceden IIR filtre vasıtasıyla modellenen sistemin katsayıları ve bulunan hata değerleri Tablo 2'de verilmektedir. Tablo 2'de, CS diğer algoritmalara nazaran en iyi hata ve en düşük standart sapma değerlerine sahip olduğundan, bağımsız çalıştırmalarda benzer başarılı performanslarla modellemeler yapabileceği söylenebilmektedir. PSO algoritması ikinci en iyi hata ve standart sapma değerlerine sahip algoritma olması sebebiyle bu modelleme örneğimiz için, CS'den sonra tercih edilebilir bir algoritma olduğu görülebilmektedir. Şekil 4'te algoritmaların sistemi modellerken en iyi hata değerlerine yakınsama grafiği verilmektedir. Şekil 4'ten CS algoritmasının en iyi hata değerine en az iterasyonla yakınsayarak, bu sistemi modellerken hızlı bir davranış sergilemekte olduğu görülmektedir. Diğer yandan, GA algoritması çözüm uzayını keşfetmede diğer algoritmalara nazaran yavaş bir durum göstermektedir. Hem bulunduğu bağımsız en iyi çözümler hem de en iyi çözümüne yakınsama hızı bakımından iyi bir performans sergilediğinden dolayı, CS algoritmasının bu örnek için en başarılı modellemeyi yapan algoritma olduğu söylenebilmektedir. Şekil 5'te modellenen ikinci dereceden sistemin sıfır kutup diyagramından, tüm algoritmaların sistemi kararlı bir şekilde tasarladıkları görülmektedir.

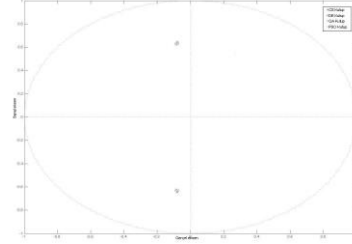
Tablo 2. Modellenen sisteme ait algoritmaların bulunduğu katsayı ve en iyi hata değerleri

Katsayılar	CS	DE	GA	PSO
a_0	-0.224	-0.191	-0.225	-0.211
a_1	-0.582	-0.550	-0.560	-0.570

	b0	1	1	1	1
b1		0.174	0.159	0.149	0.177
b2		0.392	0.420	0.400	0.396
En iyi hata		0.007	0.022	0.011	0.020
Ortalama hata		0.007	0.022	0.052	0.020
Standart sapma		3.393×10^{-19}	4.151×10^{-8}	0.033	3.818×10^{-18}



Şekil 4. Modellenen sisteme ait algoritmaların yakınsama grafiği



Şekil 5. Modellenen sisteme ait sıfır kutup diyagramı

4. Sonuçlar

Bu çalışmada, evrimsel algoritma yaklaşımıyla yüksek dereceden sistemlerin daha düşük dereceden modelleri oluşturulmuştur. Literatürde sıklıkla yer alan CS, DE, GA ve PSO algoritmaları kullanılarak modellenen üzere iki farklı örnek sistem ele alınmış ve bu algoritmaların performans analizleri yapılmıştır. Elde edilen sonuçlara göre, CS ve PSO algoritmaları çok düşük standart sapmalarla her iki örnek için modelleme yapabildiği görülmüştür. Diğer yandan, DE algoritması en hızlı şekilde en iyi çözümüne yakınsadığı fakat bu çözümün daima kullanılan algoritmalar arasında en iyi çözüm olmadığı gözlemlenmiştir. Her iki modelleme örneği için farklı algoritmalar, farklı en iyi hata değerlerine sahip olup, farklı iterasyon sayılarında çözümlerine yakınsamaktadır. Bu modellenecek sistemin derecesine bağlı olarak, algoritmaların farklı davranış sergilediğini göstermektedir. Daha düşük dereceli sistemleri modellemede, kullanılan evrimsel algoritmaların sistemlerin kararlı ve düşük hatalı modellerini oluşturabildiği ve etkin bir şekilde bu alanda kullanılabileceği görülmektedir.

Kaynaklar

- [1] Goodwin GC, Payne RL. Dynamic system identification: experiment design and data analysis. USA: Academic Press; 1977.
- [2] Van NR, Robert E. Identification of system. 2nd ed. New York: Krieger Publ. Co.; 1972.
- [3] Gytman PO, Mannnerfelt CF, Molander P. Contributions to the model reduction problem. IEEE Trans Automat Control 1982; 27: 454-455.
- [4] Yadav JS, Patidar NP, Singhai J, Panda S. Differential Evolution algorithm for model reduction of SISO discrete systems. In: World Congress on Nature & Biologically Inspired Computing(NaBIC) 2009; 1053-1058.
- [5] Ma Q, Cowan CFN. Genetic algorithms applied to the adaptation of IIR filters. Signal Process. 1996; 48(2): 155-163.

- [6] Dai C, Chen W, Zhu Y. Seeker optimization algorithm for digital IIR filter design. *IEEE T Ind Electron* 2010; 57(5): 1710-1718.
- [7] Panda G, Pradhan PM, Majhi B. IIR system identification using cat swarm optimization. *Exp Syst Appl* 2011; 38(10): 12671–83.
- [8] Upadhyay P, Kar R, Mandal D, Ghoshal SP. Crazyness based particle swarm optimization algorithm for IIR system identification problem. *AEU-Int. J. Electron. Commun* 2013; 68: 369-378.
- [9] Upadhyay P, Kar R, Mandal D, Ghoshal SP, Mukherjee V. A novel design method for optimal IIR system identification using opposition based harmony search algorithm. *J Franklin I* 2014; 351: 2454-2488.
- [10] Sheno BA. *Introduction to digital signal processing and filter design*. John Wiley&Sons; 2006.
- [11] Storn R, Price K. *Differential evolution-a simple and efficient adaptive scheme for global optimization over continuous spaces*. ICSI Technical Report TR-95-012 1995.
- [12] Goldberg DE. *Genetic algorithms in search, optimization and machine learning*. 1st ed. USA: Addison-Wesley Longman Publishing; 1989.
- [13] Yang XS, Deb S. Cuckoo search via lévy flights. In: *World Congress on Nature&Biologically Inspired Computing* 2009; 9-11.
- [14] Kennedy J, Eberhart R. Particle swarm optimization. In: *Proceedings of IEEE International Conference on Neural Networks* 1995; 1942-1948.