

# BBA, PSO ve SPÖ KULLANILARAK GERÇEK ZAMANLI GÜRÜLTÜ GİDERİMİ

## REAL TIME NOISE-CANCELLATION USING ICA, PSO and PE

R. İrem Bor<sup>1,2</sup>, Y. Ziya İder<sup>1</sup>, Orhan Arıkan<sup>1</sup>, Erdem Ertan<sup>2</sup>

<sup>1</sup>Elektrik Elektronik Mühendisliği Bölümü  
Bilkent Üniversitesi

irembor@bilkent.edu.tr

<sup>2</sup>Aselsan Elektronik Sanayi ve Ticaret A.Ş.

ribor@aselsan.com.tr

### ÖZETÇE

Bu çalışmada telsiz haberleşmesinde gürültüsüz konuşma gönderimini sağlamak için gerçek zamanlı çalışabilecek bir gürültü giderimi algoritması geliştirilmiştir. Konuşma ve gürültü kaynakları bilinmemekle birlikte yalnızca karışımları gözlenmektedir. Bu sistem anlık karışım modeli ile modellenmiştir. Konuşma ve gürültü işaretlerini ayırtmak için bağımsız bileşen analizi (BBA) ve parçacık sürü optimizasyonu (PSO) kullanılmıştır. BBA ile ayırılan işaretlerden hangisinin konuşma hangisinin gürültü olduğunu anlamak mümkün değildir (belirsizlik sorunu). Bu durum, haberleşme sırasında konuşma yerine gürültünün gönderilmesine neden olabilir. BBA'nın belirsizlik sorununu aşmak için bir ses perdesi özütleme (SPÖ) algoritması geliştirilmiş ve BBA-PSO ile birleştirilmiştir. BBA-PSO-SPÖ algoritması MATLAB ortamında gerçekleştirilmiştir. Bu çalışmanın katkıları, gerçek zamanlı BBA için varolan amaç fonksiyonlarının değiştirilerek daha gürbüz bir amaç-fonksiyonunun oluşturulması, PSO ile hızlı ve gürbüz çalışma imkanı sağlanması ve SPÖ ile belirsizlik sorununun çözülmesidir.

### ABSTRACT

In order to provide noiseless transmission of speech in wireless communication systems a real-time implementable noise cancellation algorithm is developed. Speech and noise sources are not known but only their mixtures are observed. That system is modeled with instantaneous mixture model. Combination of independent component analysis (ICA) and particle swarm optimization (PSO) algorithms is used to separate speech and noise. However, ICA has an ambiguity such that it is not possible to know which one of the separated signals is speech or noise. As a result, the transmitted signal can be noise, instead of speech. To overcome this ambiguity problem, a pitch extraction (PE) algorithm is developed and combined with ICA-PSO. ICA-PSO-PE algorithm is implemented in MATLAB. Contributions of this work are modifying objective functions of ICA algorithm to make them more robust, combining ICA with PSO to make it work fast and robust, and overcoming the ambiguity problem using PE algorithm.

978-1-4673-0056-8/12/\$26.00 ©2012 IEEE

### 1. GİRİŞ

#### 1.1. Sistem Modeli

Anlık karışım modellerinde  $m$  işaretin karışımı  $m$  alıcı tarafından elde edilir. Bu çalışmada amacımız arka plan gürültüsü ve konuşmacının sesini ayırarak gürültüsüz konuşmayı elde etmektir. Konuşma ve arka plan gürültüsü birer kaynak işareti olarak düşünülmüştür. Kaynak işaretlerinin  $t \in [1, n]$  süresince elde edilmiş örnekleri ( $s_1(t)$  ve  $s_2(t)$ ) satır matrisleriyle ifade edilirse  $\mathbf{s}_1 = [s_1(1), s_1(2), \dots, s_1(n)]$  ve  $\mathbf{s}_2 = [s_2(1), s_2(2), \dots, s_2(n)]$  olur. Bütün sistemi benzer şekilde ifade edersek:

$$\mathbf{s} = \begin{bmatrix} \mathbf{s}_1 \\ \mathbf{s}_2 \end{bmatrix}, \quad \mathbf{x} = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} \quad \text{ve} \quad \mathbf{A} = \begin{bmatrix} \alpha_{11} & \alpha_{21} \\ \alpha_{12} & \alpha_{22} \end{bmatrix} \quad (1)$$

burada  $\mathbf{x}_1$  ve  $\mathbf{x}_2$  karışmış işaretleri (gözlenen işaretleri) ve  $\mathbf{A}$  matrisi, *karışım matrisini* gösterir.  $\mathbf{x} = \mathbf{A}\mathbf{s}$  olur. Amacımız  $\mathbf{A}$  matrisinin tersi olan *ayırma matrisini* ( $\mathbf{W}$ ) bularak gözlenen işaretlerden kaynak işaretlerine ulaşmaktır. Bu problemin zorluğu kaynak işaretlerinin yanı sıra  $\mathbf{A}$  matrisinin de bilinmemesidir.  $\mathbf{y}$  gözlenen işareten ayırılan kaynak işaretlerini gösterirse

$$\mathbf{y} = \mathbf{W}\mathbf{x} \quad (2)$$

olur.  $\mathbf{A}$  matrisinin ters çevrilebilir bir matris olduğu varsayılmaktadır.

Kaynak işaretleriyle ilgili çok az bilgimiz olduğundan kaynak işaretlerini ayırtmak için *kör kaynak ayırma* (blind source separation) yöntemlerinden *bağımsız bileşen analizi* (independent component analysis) (BBA) kullanılmıştır [1, 2]. Bu yöntemde kaynak işaretleriyle ilgili iki şart vardır:

- istatistiksel olarak bağımsız olmalıdırlar
- en fazla bir tanesi normal dağılımlı olabilir

Bu şartlar modellediğimiz sistem için gerçekçidir çünkü arka plan gürültüsü ve konuşma birbirinden bağımsız işaretlerdir. Bunun yanı sıra, konuşma sinyalinin istatistiksel dağılımı Laplace dağılımına benzemektedir. Bu durumda arka plan gürültüsü normal de olsa ayırma yapılabilir.

## 1.2. Bağımsız Bileşen Analizi

BBA, kaynak işaretlerinin istatistiksel olarak bağımsız olmasına ve normal dağılımlı olmamalarına dayanarak ayrıştırma yapar. Ayrıştırma işlemi işaretin dağılımının normal dağılımından farklılaştığı doğrusal dönüşümlerini bulmaktan ibarettir. Bu amaçla *negentropy* temeli amaç fonksiyonları kullanılabilir.

### 1.2.1. Negentropy

Bir rassal değişkenin ne kadar rassal olduğu bilişim kuramı kavramlarından entropi ile ölçülür. Dağılımı  $p_x(x)$  olan bir rassal işaret ( $\mathbf{x}$ ) için türevsel entropi:

$$H(\mathbf{x}) = - \int p_x(\eta) \log p_x(\eta) d\eta \quad (3)$$

Normal dağılımlı değişkenler, aynı ortalama ve varyansa sahip diğer rassal değişkenler içinde en yüksek entropiye sahiptir. Buna ek olarak normal olmayan iki işaretin toplamının entropisi her bir işaretin entropisinden daha yüksektir. Buradan hareketle işaretlerin normal dağılımına benzemezliği (nonnormality), yani negentropisi, ayrışmanın bir ölçüsü olarak kullanılabilir.

Negentropy ( $J$ ), şu şekilde tanımlanmıştır:

$$J(\mathbf{x}) = H(\mathbf{x}_{\text{gauss}}) - H(\mathbf{x}) \quad (4)$$

Burada  $\mathbf{x}_{\text{gauss}}$  ile  $\mathbf{x}$  aynı ilinti matrislerine sahip rassal değişkenlerdir.

Kaynak işaretlerinin dağılımı bilinmediğinden negentropiyi hesaplamak için bazı yakın benzerlikler kullanılır. Polinom olmayan momentler temelli yakın benzerlik, aykırı değerlere karşı gürbüzdür [3]. Eğer tek bir polinom olmayan fonksiyon ( $G$ ) kullanılarak negentropy yakın benzetilirse:

$$J(y) \propto [E\{G(y)\} - E\{G(v)\}]^2 \quad (5)$$

Şu fonksiyonların kullanışlı olduğu gösterilmiştir [3]:

$$G_1(y) = \frac{1}{a_1} \log \cosh a_1 y \quad (6)$$

$$G_2(y) = - \exp\left(-\frac{y^2}{2}\right) \quad (7)$$

## 1.3. Parçacık Sürü Optimizasyonu

BBA ile elde edilen amaç fonksiyonunun optimizasyonu için bulgusal bir problem çözme yöntemi olan PSO kullanılabilir [4]. Her bir parçacık kendi deneyimlerinin en iyisi, yani *pbest* ve sürüdeki en iyi deneyim, yani *gbest* hakkında bilgi sahibidir. Bu çalışmada parçacıklar 2-B uzayda birer noktadır.

$i$  parçacığının pozisyonunu yani başka bir deyişle uzaydaki yerini  $\vec{x}_i$  ile gösteririz. Yer değiştirmesini ise *hız* olarak ifade edip  $\vec{v}_i$  ile gösteririz. Önemli olan  $\vec{v}_i$ 'nin nasıl güncellendiğidir çünkü parçacık, uzayı  $\vec{v}_i$  ile örneklemektedir.

$$\vec{x}_i(t) = \vec{x}_i(t-1) + \vec{v}_i(t) \quad (8)$$

Parçacığın *pbest* ve *gbest* hakkındaki bilgilerini göz önünde bulundurarak PSO'nun hız güncelleme denkleminde ulaşılır:

$$\vec{v}_i(t) = \omega \vec{v}_i(t-1) + c_1 \varphi_1 (\vec{p}_i - \vec{x}_i(t-1)) + c_2 \varphi_2 (\vec{p}_g - \vec{x}_i(t-1)) \quad (9)$$

burada  $\omega$  eylemsizlik katsayısı [5],  $c_1$  ve  $c_2$  ivme faktörleri,  $\varphi$ ler tekdüze dağılımdan seçilmiş rassal pozitif sayılar [6],  $\vec{p}_i$  ve  $\vec{p}_g$  sırasıyla *pbest* ve *gbest* ve  $t$  zaman indeksidir.

Eylemsizlik katsayısı,  $\omega$ , parçacıkların daha çabuk yakınsamalarını sağlar.  $\varphi_1$  ve  $\varphi_2$  bireysel ve toplumsal deneyimler arasında bir denge kurulmasını sağlar. Ayrıca, parçacığın uzayın uygun kısmından ayrılmaması için hız kaskılanır [7]. PSO'dan verim alınabilmesi için parametrelerinin iyi belirlenmesi gerekir[8]. *pbest* ve *gbest* ise şu şekilde hesaplanmaktadır:

$$\vec{p}_i = \begin{cases} \vec{p}_i & \text{eğer } f(\vec{x}_i(t)) > f(\vec{p}_i) \\ \vec{x}_i(t) & \text{eğer } f(\vec{x}_i(t)) < f(\vec{p}_i) \end{cases} \quad (10)$$

$$\vec{p}_g = \max(f(\vec{p}_i)) \forall i$$

Burada  $f$ , parçacığın uyumunu göstermektedir. Bu çalışmada durma şartını global en iyi pozisyonun (*gbest*) belli bir iterasyon sayısı boyunca sabit kalması olarak belirledik.

## 1.4. Ses Perdese Özütleme

Bu çalışmada ayrıştırılan bileşenlerden hangisinin konuşma, hangisinin gürültü olduğunu tespit etmek amacıyla [9]'da sunulan ses perdesi özütleme algoritmasına dayalı bir teknik geliştirilmiştir.

Ses perdesi gecikmesini (pitch lag)  $\tau$  ile gösterirsek normalleştirilmiş bağlantı katsayısı şu şekilde hesaplanır:

$$\rho = \frac{\langle x_0, x_\tau \rangle}{\sqrt{\langle x_0, x_0 \rangle \langle x_\tau, x_\tau \rangle}} \quad (11)$$

Algoritmanın önemli özelliklerinden biri özilintilerin (autocorrelation) hesaplanma biçimidir. Çerçeve uzunluğu  $L$  olan bir çerçevede bir özilinti şu şekilde hesaplanır:

$$\langle x_{\tau_1}, x_{\tau_2} \rangle = \sum_{k=\frac{L}{2}-1}^L x[k - \tau_1] x[k - \tau_2] \quad (12)$$

Kullanılan ses perdesi gecikmeleri 5 ms'den 20 ms'ye kadardır. Bu değerlerin her biri için normalleştirilmiş bağlantı katsayısı hesaplanır ve en büyüğü seçilir. Buna ek olarak, işaretlerin enerjisi de en büyük normalleştirilmiş bağlantı katsayısıyla birlikte değerlendirilebilir:

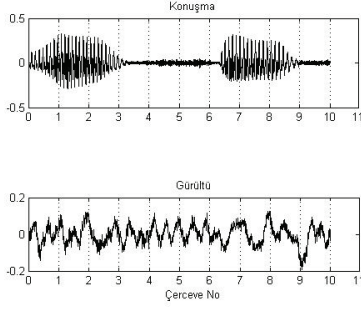
$$P(x) = \sum_{k=\frac{L}{2}-1}^L x[k]^2 \quad (13)$$

$$R = P(x) \rho \quad (14)$$

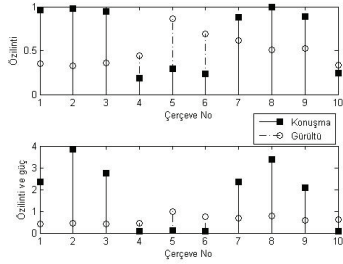
Şekil 1'de konuşma ve gürültü işaretleri, Şekil 2'de özilinti ve  $R$  değerleri görülmektedir. Konuşma ve gürültünün işaretlerinin özilinti ve  $R$  değerleri hem ötümlü hem de ötümsüz çerçevelerde oldukça farklıdır.

## 2. YÖNTEM

Bağımsız bileşenleri gerçek zamanlı ayırabilmek için belirli sayıda örnek içeren çerçeveler kullanılmıştır. Her çerçevede bağımsız bileşenleri ayırtmak için BBA'nın amaç fonksiyonları PSO ile eniyilenmiş ve ayrıştırılan işaretlerden hangisinin ses hangisinin gürültü olduğuna karar verilmesi için SPÖ kullanılmıştır.



Şekil 1: 10 çerçevelik konuşma ve gürültü işareti.



Şekil 2: Şekil 1'deki işaretlerin özilinti ile güç çarpımları ( $R$ ).

### 2.0.1. BBA-PSO Algoritması

İşaretlerin doğrusal dönüşümleri açısal olarak ifade edilebilir.  $\mathbf{W}$  matrisi bu açılar cinsinden ifade edilirse:

$$\Theta = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} \quad ve \quad \mathbf{W} = \begin{bmatrix} a \cos \theta_1 & a \sin \theta_1 \\ b \cos \theta_2 & b \sin \theta_2 \end{bmatrix} \quad (15)$$

Amacımız  $\Theta$  bulmak olduğu için  $a$  ve  $b$  değerleri ihmal edilmiştir. Bu durum BBA'nın belirsizliklerinden biri olup ayrıştırılan işaretlerin ölçekleme oranlı olmasına neden olmaktadır [?].  $\theta_1$  ve  $\theta_2$  ayrı ayrı bulunacağından her bir ayrıştırılmış işaret için:

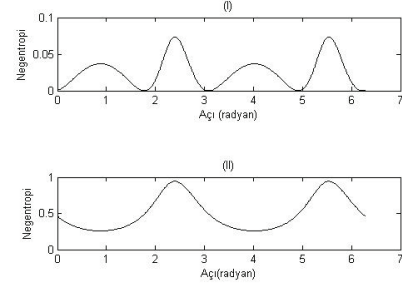
$$\mathbf{y}_1 = \mathbf{w}_1 \mathbf{x} \quad (16)$$

$$\mathbf{y}_2 = \mathbf{w}_2 \mathbf{x} \quad (17)$$

burada  $\mathbf{w}_1 = [\cos \theta_1 \sin \theta_1]$  ve  $\mathbf{w}_2 = [\cos \theta_2 \sin \theta_2]$  olur. Genellikle  $\theta$ lar Şekil 3-(I)'deki gibi en büyük iki değer elde edildiği açılarda bulunur. Bu çalışmada amaç-fonksiyonuna müdahale edilerek  $\theta$ 'lardan birinin en büyük ve diğerinin en küçük değerden elde edilmesi sağlanmıştır.

$$J(y) \propto E\{G(y)\} \quad (18)$$

Şekil 3-(II)'de  $G(y)$  olarak Eşitlik (7) kullanılmıştır. Böylece amaç-fonksiyonu dağılımın merkezine odaklanır ve aykırı değerlere karşı dayanıklı olur. Dikgenleştirme ara işlemine gerek kalmaz. Her iki amaç-fonksiyonuyla da aklaştırma ön işlemi atlanmıştır. Laplacian ve tekdüze dağılımlı rassal değişkenler  $A = [0.8 \ 0.5; 0.6 \ 0.7]$  matrisi kullanılarak karıştırılınca  $\theta_1 =$



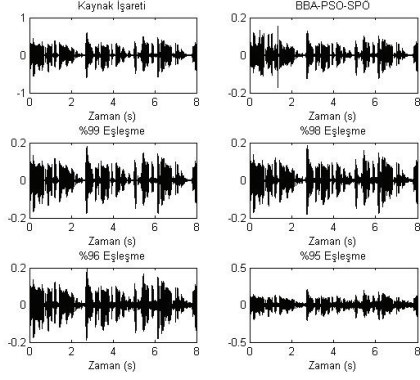
Şekil 3: (I)'de amaç fonksiyonu olarak eşitlik (5) ile (7), (II)'de ise (18) kullanılmıştır.

2.5 ve  $\theta_2 = 0.9$  olmaktadır. Her iki şekilde de doğru açılarda ekstremumlar bulunmaktadır.

### 2.1. BBA-PSO-SPÖ ile Gürültü Çıkarımı

Bu çalışmada karışım matrisinin çok yavaş değiştiği varsayılmıştır. Bu nedenle her çerçeve için sürekli olarak  $\Theta$  hesaplamak yerine bir öğrenme süreciyle  $\Theta$ 'ya karar vermek doğrudur. Karar verdikten sonra bazı kalite ölçütleriyle denetleme yapılarak karışım matrisinin değişip değişmediği kontrol edilebilir.

1. BBA, PSO ve SPÖ ilklenir ve örnek beklentisi (sample mean) işaretin kendisinden çıkarılarak işaret merkezlenir.
2. Parçacıkların yer değerleri ile ayrıştırılmalar yapılır (16).
3. (18) ile her bir parçacığın uyum değeri hesaplanır.
4.  $pbest$  ve  $gbest$  (10)'te olduğu gibi güncellenir.
5. Yer ve hız değerleri her bir parçacık için eşitlik (8) ve (9)'da gösterildiği gibi güncellenir.
6. En az iterasyon sayısına ulaşılan kadar adım 2'ye geri dönlür.
7. En az iterasyon sayısına ulaşıldıktan sonra  $gbest$  değerinin değişimi kontrol edilir. Eğer değişim eşik değerinde veya daha az ise yakınsama tamamlanmıştır. Aksi halde adım 3'e dönlür.
8. Ayrıştırma işlemi (16) ile yapılır ve SPÖ algoritmasına geçilir.
9. Her bir ayrıştırılmış işaret için eşitlik (13)'te gösterilen  $R$  değeri hesaplanır. Bundan yararlanarak  $\tan \theta_1$  ve  $\tan \theta_2$  değerleri doğru kanallarda saklanır.
10. Öğrenme bitmediyse bir sonraki çerçeve için 1. adıma dönlür.
11. Öğrenme bittiyse  $\tan \theta_1$  ve  $\tan \theta_2$  nin saklanmış bütün değerlerinin dağılım eğrisi oluşturulur. En sık rastlanan değer doğru kabul edilir ve ark tanjantları alınarak  $\theta_1$  ve  $\theta_2$  elde edilir.  $R$  kullanılarak  $\mathbf{w}_1$  ve  $\mathbf{w}_2$  elde edilir.
12. Her bir çerçeve  $\mathbf{w}_1$  ve  $\mathbf{w}_2$  kullanılarak bileşenlerine ayrılır ve bir kalite ölçüsüyle  $\mathbf{w}_1$  ve  $\mathbf{w}_2$ 'nin doğruluğu denetlenebilir.



Şekil 4: Gürültü giderimi ile BBA-PSO-SPÖ'nün kıyaslaması.

### 3. SONUÇLAR

Bu çalışmada BBA-PSO-SPÖ algoritmasının  $W$  matrisini başarıyla yaklaştırdığı gösterilmiştir. Başarı ölçütü olarak  $\theta_1$  ve  $\theta_2$ 'nin ne kadar iyi yaklaşıldığı alınmıştır. Kaynak işaretleri SISEC 2010 verilerinden alınmış konuşma ve gürültü kayıtlarıdır. Örnekleme sıklığı 16 kHz ve herbir çerçeve 640 örnektir (40 ms). Tablo 1'de teorik ve bulunan  $\theta$  değerleri, 8 saniyelik verinin sırasıyla 1. ve 2. kanallar için sinyal - gürültü oranı (SGO), hata ve öğrenme süreci boyunca  $\theta$ 'yu bulmak için çerçeve başına yapılan ortalama tekrarlar miktarı gösterilmektedir.  $\theta$  değerlerinin çok yakın olduğu ya da SGO'nun çok düşük olduğu durumlarda dahi sonuçların başarılı olduğu görülmektedir.

Şekil 5'te Tablo 1'de ilk durumda karışık kaynak işaretleri, gözlenen ve ayrıştırılan işaretler gösterilmiştir. Öğrenme sürecinin sonu olan 2. saniyeden sonra işaretlerin kaynak işaretlerine benzerliği dikkat çekmektedir.

Telsizlerde en sık kullanılan gürültü giderimi yöntemlerinden biri ikinci kanaldaki işareti ters çevirerek birinci kanaldakiyle toplamaktır. Mikrofonların kazançları tamamen eşlenmişse ve sesin her iki mikrofona aynı anda geldiği varsayılırsa bu yöntem gürültüyü başarıyla çıkarabilir. Ancak mikrofonların tam eşlenmemesi veya ikinci kanalda gecikme olması bu yöntemin başarısını düşürür. MATLAB'da gecikme ihmal edilerek yapılan simülasyonlar göstermektedir ki BBA-PSO-SPÖ yöntemi mikrofona kazançlarının %100 aynı olduğu durumdaki başarıyı sağlar (Şekil 5).

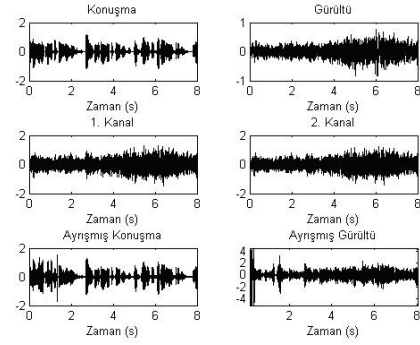
Daha farklı kaynak işaretleriyle ve daha çeşitli amaç-fonksiyonlarıyla yapılacak gözlemler algoritmanın performansı hakkında gerçekçi bilgiler sağlayacaktır. Öğrenme sürecinin sonrasında uygulanacak gürbüz bir kalite ölçme yönteminin geliştirilmesi ve algoritmanın gerçek zamanlı ortamda gerçekleştirilmesi gelecekte yapılacak işlerdendir.

### 4. KAYNAKÇA

- [1] Cardoso, J. F., "Source separation using higher order moments", Proceedings of ICASSP'89, 2109–2112, 1989
- [2] Comon, P., "Independent component analysis: A new concept", Signal Processing, 36, 287–314, 1994

Tablo 1:  $\theta$  değerleri

Teorik $\theta$	SGO(dB)	Bulunan $\theta$	Hata	Tekrarlama
-0.7854	-5.7477	-0.7854	0.0000	16.62
-0.9273	-6.9971	-0.9151	0.0122	16.62
-0.8622	-2.4378	-0.8761	0.0139	16.26
-0.9273	-3.0177	-0.9151	0.0122	16.26
-1.1903	-7.0984	-1.1903	0.0000	16.27
-1.2490	-7.8902	-1.1760	0.0730	16.27



Şekil 5: Ayrıştırılmış işaretler.

- [3] Hyvärinen, A., "New approximations of differential entropy for independent component analysis and projection pursuit", Advances in Neural Information Processing Systems, 10, 273–279, 1998
- [4] Kennedy, J. ve Eberhart, R. C., "Particle swarm optimization", Proceedings of the IEEE International Conference on Neural Networks, 1942–1948, 1995.
- [5] Shi, Y. ve Eberhart, R. C., "A modified particle swarm optimizer", IEEE International Conference on Evolutionary Computation, Anchorage, Alaska, 1998.
- [6] Ozcan, E. ve Mohan, C., "Surfing Waves", Proceedings of the IEEE International Conference on Neural Networks, 1939–1944, 1999.
- [7] Kennedy, J. ve Eberhart, R. C., "A new optimizer using particle swarm theory", Proceedings of the Sixth International Symposium on Micro Machine and Human Science, 39–43, 1995.
- [8] Kennedy, J. ve Eberhart, R. C., Swarm Intelligence, Jorgan Kaufmann Publishers, 2001.
- [9] Ertan, A. E., "Pitch-synchronous processing of speech signal for improving the quality of low bit rate speech coders", Ph.D. dissertation, Dept. Elect. Eng., Georgia Institute of Technology, Atlanta, Georgia, 2004.