


	SAKARYA ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ DERGİSİ <i>SAKARYA UNIVERSITY JOURNAL OF SCIENCE</i>		
	e-ISSN: 2147-835X Dergi sayfası: http://dergipark.gov.tr/saufenbilder		
	<u>Gelis/Received</u> 03-05-2017 <u>Kabul/Accepted</u> 29-08-2017	<u>Doi</u> 10.16984/saufenbilder.310240	

Hata bulma yöntemlerinin yanlış alarm oranları

Yusuf Sevim^{*1}

ÖZ

Bu çalışma bağımsız bileşen analiz (BBA) ve temel bileşen analiz (TBA) algoritmalarının Tennessee Eastman (TE) süreci üzerindeki hata bulma ve yanlış alarm oranları (YAO) üzerine yoğunlaşmaktadır. TBA ve ICA algoritmaları, veri tabanlı hata bulmak için oldukça fazla uygulanmalarına rağmen, algoritmaların YAO üzerine sınırlı çalışma vardır. Bu çalışmada, algoritmaların YAO'ları TE süreci üzerinde incelenecektir. Simülasyon çalışmaları, sunulan algoritmalar hata bulmada oldukça doğruyken, YAO'ları için BBA'nın TBA'dan daha yüksek performansa sahip olduğunu göstermiştir.

Anahtar Kelimeler: Süreç Gözlemeleme, Hata Bulma, Bağımsız Bileşen Analiz, Temel Bileşen Analiz, T^2 istatistik

False alarm rates of fault detection methods

ABSTRACT

This study focuses on the fault detection (FD) and false alarm rates (FAR) of Principal component analysis (PCA) and independent component analysis (ICA) algorithms on the Tennessee Eastman (TE) process. However, PCA and ICA algorithms have been applied widely to systems for data driven fault detection, there are limited work on FARs of the algorithms. In this work, FARs of the algorithms are investigated on TE process. Simulation study indicates that the proposed algorithms are robust for fault detection, and ICA has higher performance than PCA for FARs.

Keywords: Process Monitoring, Fault Detection, Independent Component Analysis, Principal Component Analysis, T^2 Statistics

* Yusuf SEVİM / Corresponding Author

¹ Karadeniz Teknik Üniversitesi, Elektrik Elektronik Mühendisliği Bölümü, ysevim@ktu.edu.tr

1. GİRİŞ (INTRODUCTION)

Günümüzde, fabrika veya kimyasal üretim süreçlerinin güvenliği için süreçlerde meydana gelen hataların tespiti oldukça önem kazanmıştır. Sürecin güvenle sürdürülebilmesi için hatanın veya süreci bozan işlemin hızlı bir şekilde bulunması ve ortadan kaldırılmaları gerekmektedir. Hata tespiti için kullanılan yöntemler model tabanlı [1] ve veri tabanlı [2] olmak üzere iki ana gruba ayrılmaktadırlar. Veri tabanlı yöntemler model tabanlı yöntemlere göre daha basit olmalarına karşın, özellikle çapraz ilintili olan çok boyutlu çok değişkenli veriler uygulamada problem oluşturmaktadır [3]. Çok boyutlu çok değişkenli veri tabanlı süreçlerde hata bulmak için istatistiksel yöntemler kullanan pek çok algoritma geliştirilmiştir. Bu tür algoritmalara temel bileşen analiz (TBA) ve bağımsız bileşen analiz (BBA) algoritmaları örnek verilebilir [4,5]. TBA algoritmasının amacı çok değişkenli veriyi ilintisiz hale getirmek iken [6], BBA algoritmalarının amacı çok değişkenli veriyi istatistiksel olarak bağımsız hale getirmektir [7]. Literatürde en çok bilinen ve kullanılan BBA algoritması HızlıBBA algoritmasıdır. Süreç gözlemeleme uygulamalarında TBA ve BBA algoritmalarının pek çok farklı versiyonları kullanılmıştır [8-10].

TBA tabanlı çok değişkenli istatistiksel süreç gözlemeleme algoritmaları gözlemlenen tüm sinyallerin istatistiksel olarak ilintisiz olduklarını varsayarken, BBA tabanlı algoritmalar tüm verinin istatistiksel olarak birbirinden bağımsız olduğu varsaymaktadır [7,11]. Çoğu süreç için veri üzerinde yapılan bu varsayımlar doğru iken bazı süreçlerde bu varsayımlar süreçteki değişkenlerin farklı olasılık dağılımlara sahip olmalarından dolayı doğru değildir. Özellikle BBA algoritmalarında birden fazla değişken Gauss dağılıma sahip ise BBA algoritmaları bu değişkenleri birbirinden ayıramamaktadır. BBA algoritmalarında sadece bir değişkenin Gauss dağılıma sahip olmasına izin verilir [7]. Bu tür veride hata bulmak için ikinci dereceden istatistikten faydalanan, ikinci dereceden kör kaynak ayrıştırma (İDKKA) algoritmalarının kullanılması gerekir [12]. İkinci dereceden istatistik kullanan bu yöntemde Gauss dağılıma sahip sinyallerin ayrıştırılabilmesi için sinyallerin farklı spektruma sahip olmaları gerekmektedir. Hata bulma uygulamalarının çoğunda bu durum

göz ardı edilir ve süreçten elde edilen verilerin sadece belirli olasılık dağılımlarına sahip oldukları varsayılarak süreç verisine uygulanacak algoritmalar belirlenmektedir. Bu uygulama ile seçilen algoritmanın varsayımına uyan değişkenler için iyi sonuç alınır iken varsayımına uymayan değişkenlerde ise sonuçlar kötü çıkmaktadır. Bu çalışmanın katkısı süreç gözlemeleme tek bir algoritma kullanma yerine farklı algoritmalar kullanarak elde edilen sonuçların değerlendirilmesi ve en iyi sonuç veren algoritmanın sonuçlarının göz önünde bulundurulması süreci hakkında karar verilmesidir.

Bu yayında, ikinci dereceden ve yüksek dereceden istatistik kullanan algoritmaların hata bulma performansları farklı sinyal dağılımları için araştırılmıştır. Bunun için TBA, HızlıBBA ve İDKKA algoritmaları sinyal ayrıştırma için kullanılmış ve elde edilen ayrıştırılmış sinyaller kullanılarak algoritmaların hata bulma performansları karşılaştırılmıştır.

2. VERİ TABANLI SÜREÇ GÖZLEMELEME, HATA BULMA METOTLARI (DATA BASED PROSESS MONITORING, FAULT DETECTION METHODS)

Bu bölümde, temel süreç gözlemeleme ve hata bulma metotları incelenecektir. Aynı zamanda bu yöntemler arasında küçük farklılıklardan bahsedilecektir.

2.1. Temel Bileşen Analizi (TBA) (Principal Component Analysis (PCA))

TBA algoritması, orijinal veriye dönüşüm uygularken önemli olan değişken bilgiyi korumaktadır. TBA algoritmasının pek çok farklı alanda başarılı uygulamaları bulunmaktadır. Süreç gözlemeleme uygulaması olarak da pratikte pek çok kez kullanılmıştır [13-15]. TBA uygulamalarında $x(k) = [x_1(k) \dots x_m(k)]^T$ sıfır ortalamaya sahip, normalize edilmiş ve birim varyansa sahip bir vektör olduğu varsayılırsa, TBA algoritmasının amacı bu vektör için optimum bir lineer dönüşümü bulmak ve veriyi aşağıdaki ifade ile dönüştürmektir.

$$t(k) = P^T x(k) = [t_1(k) \dots t_m(k)]^T \quad (1)$$

bu denklemde ki t_i 'ler temel bileşenler olarak ifade edilirler ve bu temel bileşenler birbirleri ile ilintisizdirler ($PP^T = I$). P matrisinin kolon vektörleri p_i 'ler, λ_i öz değerlerine karşılık gelen öz

vektörleri temsil etmektedir. Bu öz değer ve özvektörler veri matrisi X in ilinti veya kovaryans (Σ) matrisinin öz değer ayrıştırmasına uğratılması ile aşağıdaki denklemdeki gibi elde edilir.

$$\Sigma = P\Lambda P^T \quad (2)$$

Bulunan özdeğer ve özvektör matrisleri iki bölüme ayrıştırılır. İlk bölüm muhafaza edilen süreç değişimine karşılık gelirken ikinci bölüm ihmal edilen süreç değişimine karşılık gelmektedir.

$$\Lambda = \begin{bmatrix} \hat{\Lambda} & 0 \\ 0 & \tilde{\Lambda} \end{bmatrix} \quad P = [\hat{P} \mid \tilde{P}] \quad (3)$$

Aynı zamanda temel bileşen vektörü de iki parçaya bölünür.

$$t(k) = [\hat{t}(k) \mid \tilde{t}(k)]^T \quad (4)$$

Üsteki denklem daha açık bir şekilde yazılabilir.

$$\hat{t}(k) = \hat{P}^T x(k) \quad , \quad \tilde{t}(k) = \tilde{P}^T x(k) \quad (5)$$

Bu denklemlerdeki $\hat{t}(k)$ önemli süreç değişimini temsil ederken, $\tilde{t}(k)$ süreç değişkenleri arasındaki yapay boş kombinasyonu temsil eder. Üsteki ifadeyi temel olarak $x(k)$ veri vektöründe aşağıdaki denklemdeki gibi ayrıştırılabilir.

$$\begin{aligned} x(k) &= \hat{x}(k) + e(k), \quad \hat{x}(k) = \hat{C}x(k), \\ e(k) &= \tilde{C}x(k) \end{aligned} \quad (6)$$

Üstteki denklemdeki $\hat{x}(k)$ TBA modelinden elde edilen veri tahmin vektörünü temsil ederken, $e(k)$ hata vektörünü temsil etmektedir. Burada ki $\hat{C} = \hat{P}\hat{P}^T$ ve $\tilde{C} = I_m - \hat{C}$ şeklinde ifade edilirler. Bu işlem ile veri uzayı birbirine dik iki alt uzaya ayrıştırılır. Temel alt uzay, ilk $m - i$ özvektörler ile şekillendirilir iken en son i özvektörler ile kalan alt uzay şekillendirilir.

2.1.1. TBA ile Hata Tespiti (Fault Detection with PCA)

TBA algoritması ile hata tespiti yapabilmek için ilk önce sistemde arıza yokken sistemden ölçülen veriler toplanır. Toplanan bu verinin Hotelling'in T^2 istatistiği veya karesel tahmini hatalar (KTH) istatistiği çıkarılır ve bu hatasız verilere göre güven bölgesi belirlenir. Belirlenen bu güven bölgesini aşan sinyal olursa hata sinyali olarak değerlendirilir. Hatasız veri için T^2 istatistiğinin hesaplanması aşağıdaki denklem ile yapılmaktadır.

$$T^2(k) = \hat{t}(k)^T \hat{\Lambda}^{-1} \hat{t}(k) \quad (7)$$

Bu istatistik için güven bölgesinin üst sınırı aşağıdaki denklemde F-dağılımı kullanarak elde edilir.

$$T_{a,n,\alpha}^2(k) = \frac{a(n-1)}{n-a} F_{a,n-a,\alpha} \quad (8)$$

Bu denklemdeki n örnek sayısını, a ayrıştırılan ilk bölümdeki temel bileşenlerin sayısını ve α yüzdelik olarak önem seviyesini temsil etmektedir. Benzer şekilde hatasız veri için KTH istatistiği aşağıdaki denklem ile elde edilir.

$$\begin{aligned} KTH(k) &= e(k)^T e(k) \\ KTH(k) &= x(k)^T (I - \tilde{P}\tilde{P}^T)x(k) \end{aligned} \quad (9)$$

Bu denklemdeki I birim matrisi temsil etmektedir. KTH için güven bölgesinin üst sınırı aşağıdaki denklemler ile belirlenir.

$$KTH_\alpha = \theta_1 \left(\frac{h_o c_\alpha \sqrt{2\theta_2}}{\theta_1} + 1 + \frac{\theta_2 h_o (h_o - 1)}{\theta_1^2} \right)^{\frac{1}{h_o}} \quad (10)$$

Üstteki denklemdeki c_α güven aralığıdır ve hata fonksiyonu tablosundan elde edilebilirken, denklemdeki θ_i ve h_o aşağıdaki gibi hesaplanır.

$$\theta_i = \sum_{j=a+1}^m \lambda_j^i \quad (11)$$

$$h_o = 1 - \frac{2\theta_1\theta_3}{3\theta_2^2} \quad (12)$$

2.2. Bağımsız Bileşen Analizi (BBA) (Independent Component Analysis (ICA))

BBA amacı Gaussian olmayan dağılıma sahip kaynak sinyallerinin ayrıştırılarak elde etmektir. m değişkene sahip bir süreç için HızlıBBA algoritması aşağıdaki gibi özetlenebilir.

1.adım: n tane örnek toplanır ve ortalaması sıfır olacak halde merkezileştirilir.

2.adım: Merkezileştirilen veri $Z^T = QX^T$ işlemi ile beyazlaştırılır. Bu işlemde kullanılan $Q = \Lambda^{-1/2}P^T$ ile hesaplanır. Bu denklemdeki Λ , X veri matrisinin kovaryans matrisinin özdeğerleri iken, P bu matrisin özvektörleridir.

3.adım: Aşağıdaki amaç fonksiyonunu m değişken için m defa optimize et

$$b_k = \arg \max_{\forall b_k, E\{y^T\}=1} [E\{G(y)\} - E\{G(v)\}]^2 \quad (13)$$

$$y = b_k^T z \quad (14)$$

Üstteki denklemde v sıfır ortalamalı birim varyansa sahip gauss değişkeni iken, G karesel olmayan nonlinear bir fonksiyondur. Optimizasyon işlemi ile bulunan tüm b_k 'ler bir matrisin içine konularak $B = [b_1 \dots b_m] \in R^{m \times m}$ oluşturulur. Ayırıştırıcı matris ise aşağıdaki gibi oluşturulur.

$$W = B^T Q \quad (15)$$

2.2.1. BBA ile Hata Tespiti (Fault Detection with ICA)

BBA algoritması ile hata tespiti için ayırıştırıcı matris W , W 'nin baskın bölümü W_b ve dışta bırakılan parça W_d olmak üzere iki kısma ayrıştırılır. Bu ayrıştırılan matris kullanılarak $B_b = (W_b Q^{-1})^T$ hesaplanır. Hata tespiti yapmak için TBA algoritmasında olduğu gibi hatasız veri için istatistiklerin hesaplanması gerekir. BBA algoritmaları için önerilen üç tane istatistiksel ölçüm aşağıdaki denklemlerde görülmektedir [5].

$$I^2(k) = x(k)^T W_b^T W_b x(k) \quad (16)$$

$$I_e^2(k) = x(k)^T W_d^T W_d x(k) \quad (17)$$

$$KTH(k) = e(k)^T e(k) \quad (18)$$

$$KTH(k) = (x(k) - \hat{x}(k))^T (x(k) - \hat{x}(k))$$

KTH denklemindeki $\hat{x}(k)$ aşağıdaki gibi hesaplanır.

$$\hat{x}(k) = Q^{-1} B_b W_b x(k) \quad (19)$$

Bu istatistikler için güven bölgelerinin üst sınırları çekirdek yoğunluk tahmini yöntemi ile bulunmasıdır [5]. Sınır değeri yoğunluk dağılımının alanlarının %99'unun seçilmesi ile belirlenir. Tek değişkenli çekirdek tahmin edicisi aşağıdaki denklemdeki gibi tanımlanır.

$$\hat{f}(z) = \frac{1}{nh} \sum_{i=1}^n k \left(\frac{z - z_i}{h} \right) \quad (20)$$

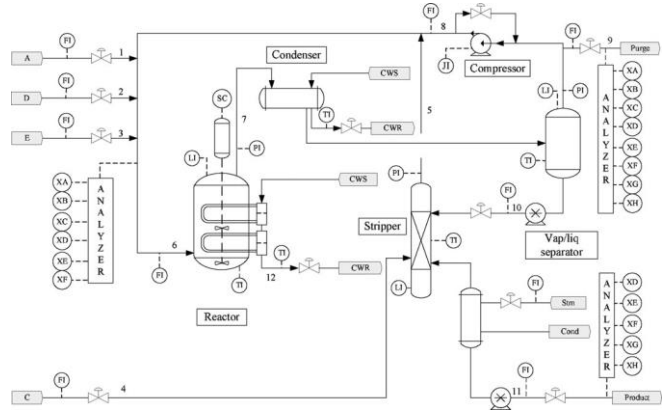
Bu denklemde n örnek sayısı, h çekirdeklerin varyansdır ve $h = 1.06\sigma n^{-1/5}$ denklemi ile

hesaplanmaktadır. Denklemdaki $k(u)$ çekirdek fonksiyonunu temsil etmektedir ve genellikle çekirdek fonksiyon olarak Gauss çekirdek fonksiyonu tercih edilmektedir. Gauss çekirdek fonksiyonu aşağıdaki denklemdeki gibi tanımlanmaktadır.

$$k(x) = \frac{1}{2\pi} \exp\left(-\frac{1}{2}x^2\right) \quad (21)$$

3. BENZETİM ÇALIŞMALARI (SIMULATION STUDIES)

Hata bulma algoritmalarının performanslarını karşılaştırmak için oldukça yaygın kullanılan Tennessee Eastman (TE) süreci kullanılmıştır. TE süreci Şekil 1'de de görüleceği üzere, reaktör, toplayıcı, sıyırıcı, kompresör ve ayırıcı olmak üzere beş ana birimden oluşmaktadır.



Şekil 1. TE süreci (TE process)

TE sürecinde, A, C, D, E ve B etkisiz gazı ile reaktör beslenerek G, H ve F yan ürünü üretilir. Süreçten, 22 tane sürekli ölçüm, 19 tane birleşik ölçüm ve 12 tane müdahale edilebilir ölçüm yapılabilmektedir. Öğrenme ve test verisi için sürecin örnekleme aralığı 3 dakika olarak seçilmiştir. TE sürecinde hataları üretmek için programlanabilir hatalar mevcuttur ve bu programlanabilir hataların listesi Tablo 1'de görülmektedir.

Öğrenme aşamasında kullanılan hatasız verinin uzunluğu 480 örnek iken, test aşamasında kullanılan hatalı veri uzunluğu 960 örnek olarak seçilmiştir. Hatalı verilerdeki tüm hatalar 160.örnekten sonra sisteme dahil edilmiştir. Yapılan çalışmada adil karşılaştırma yapmak için tüm veri hata bulma algoritmaları uygulanmadan önce ölçeklendirilmiştir ve her algoritma için 53 değişkenin tümü kullanılmış olup hiçbir veri boyutu indirgeme yöntemi kullanılmamıştır.

Yapılan çalışmanın sonuçları aşağıdaki tabloda görülmektedir.

Tablo 1. TE süreci için programlanabilir hatalar

Hata Sinyali	Süreç Değişkeni	Hata Türü
1	A/C besleme oranı, B sabit	Birim Basamak
2	B, A/C oranı sabit	Birim Basamak
3	D besleme sıcaklığı	Birim Basamak
4	Reaktör soğutma suyu giriş sıcaklığı	Birim Basamak
5	Toplayıcı soğutma suyu giriş sıcaklığı	Birim Basamak
6	A besleme kaybı	Birim Basamak
7	C bağlantı basınç kaybı	Birim Basamak
8	A, B, C besleme birleşimi	Rastgele Değişim
9	D besleme sıcaklığı	Rastgele Değişim
10	C besleme sıcaklığı	Rastgele Değişim
11	Reaktör soğutma suyu giriş sıcaklığı	Rastgele Değişim
12	Toplayıcı soğutma suyu giriş sıcaklığı	Rastgele Değişim
13	Reaksiyon kinetikleri	Yavaş Kayma
14	Reaktör soğutma suyu vanası	Sıkışma
15	Toplayıcı soğutma suyu vanası	Sıkışma
16	Bilinmiyor	Bilinmiyor
17	Bilinmiyor	Bilinmiyor
18	Bilinmiyor	Bilinmiyor
19	Bilinmiyor	Bilinmiyor
20	Bilinmiyor	Bilinmiyor
21	Vana kalıcı durumda sabitlenmiş	Sabit Pozisyon

Tablo 2'den görüldüğü üzere algoritma performansları 7 hata sinyali (1, 2, 8, 9, 13, 14 ve 17) kullanılarak elde edilmiştir. Bu 7 hata 5 farklı hata türünden elde edilmiştir. 9. ve 17.hata sinyali hariç her iki algortmada 800 hatanın hemen hemen hepsini iyi bir şekilde bulmaktadır. 9.hata sinyali olan D besleme sıcaklığı tüm değişkenler üzerinde etkisini algortmalar tarafından giderilememektedir. Bundan dolayı tüm algortmaların HBO oranları bu hata sinyali için düşük çıkmaktadır. Aynı sorun 17.hata sinyali içinde geçerlidir, fakat algortmaların HBO'ları 9.hata sinyaline göre daha yüksektir. Hata bulmak için kullanılan istatistiksel yöntemlerin bir tanesi bile hata oranını yüksek bulması hata bulma yöntemi için yeterlidir.

Tablo 2'deki algortmaların YAO'ları göz önünde bulundurulduğunda, TBA algortmasının YAO oranları 9. ve 17. hatalar hariç BBA

algortmasından yüksek çıktığı görülmektedir. Bu yüksek çıkmanın anlamı, sistemde hata olmadığı halde algortmanın hata varmış gibi yanlış alarm verdiği anlamına gelmektedir. 9. ve 17.hata sinyalleri sonuçlarında ise BBA algortmasının YAO oranları yüksek, HBO düşük çıktığından dolayı bu hata sinyalleri için TBA algortmasının bu tür sinyaller için daha uygun olduğu anlamına gelmektedir.

4. SONUÇLAR VE DEĞERLENDİRME (RESULTS AND CONCLUSIONS)

Yapılan çalışmada TBA ve BBA algortmalarının hata bulma performansı TE süreci üzerinde incelenmiştir. Yapılan çalışmalar her iki algortmanın HBO'larının yaklaşık olarak eşit olduğu fakat YAO'larının farklı olduğu görülmektedir. Sistemde hata yokken bile TBA algortması hata varmış gibi göstermektedir. Bu onun hata bulma algortması olarak tercih edilmemesine neden olabilir. BBA algortması YAO oranları oldukça düşüktür. 9.hata sinyalinde ise her iki algortmanın hata tespitinde yetersiz kaldığı gözlemlenmektedir. Bu da bu algortmaların bu sinyal için yetersiz olduğu ve sinyalleri oluşturan temel bileşenleri iyi ayırtıramadığı anlamına gelmektedir. Hata teşhisinde algortma seçiminde süreci oluşturan temel bileşenlerin başlangıçta bilinmesi ve buna göre algortma tercihi yapılması oldukça önemlidir. Eğer temel bileşenler bilinmiyor ise hibrit algortmalar ile hata teşhisi yapılması daha doğru sonuçlar verecektir.

Tablo 2. Benzetim çalışmaları (Simulation results)

Hata Sinyali	ALGORİT MALAR	Hata Bulma Oranı (HBO)			Yanlış Alarm Oranı (YAO)			Hata Türü
		KTH	T ² -I ²	I _e ²	KTE	T ² - I ²	I _e ²	
1	TBA	800	796		29	15		Birim Basamak
	BBA	797	796	797	3	0	3	
2	TBA	798	790		33	11		Birim Basamak
	BBA	789	747	789	2	0	3	
8	TBA	787	785		33	11		Rastgele Değişim
	BBA	791	725	795	5	2	10	
9	TBA	218	131		44	32		Rastgele Değişim
	BBA	100	1	139	41	2	53	
13	TBA	772	766		30	7		Yavaş Kayma
	BBA	767	747	765	3	0	3	
14	TBA	798	800		34	15		Sıkışma
	BBA	800	722	800	3	1	2	
17	TBA	631	386		36	47		Bilinmiyor
	BBA	473	262	570	60	1	66	

KAYNAKLAR

- [1] J. Chen ve R. J. Patton, Robust Model-Based Diagnosis for Dynamics Systems, Kluber Academic Publisher, 1999.
- [2] T. Kourti, "Process analysis and abnormal situation detection: from theory to practice," *Control Systems, IEEE*, vol. 22, no. 5, pp. 10-25, 2002.
- [3] S. Yin, ve ark., "A comparison study of basic data-driven fault diagnosis and process monitoring methods on the benchmark Tennessee Eastman process," *Journal of Process Control*, vol. 22, no. 9, pp. 1567-1581, 2012.
- [4] T. Villegas, M. J. Fuente ve M. Rodríguez, "Principal component analysis for fault detection and diagnosis. experience with a pilot plant," in *CIMMACS'10 Proceedings of the 9th WSEAS international conference on computational intelligence, man-machine systems and cybernetics*, 2010.
- [5] J. Lee, C. K. Yoo ve I. Lee, "Statistical process monitoring with independent component analysis," *Journal of Process Control* vol. 14, no. 5, pp. 467-485, 2004.
- [6] H. Abdi ve J. W. Lynne, "Principal component analysis," *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 2, no. 4, pp. 433 -459, 2010.
- [7] A. Hyvärinen, "Fast and robust fixed-point algorithms for independent component analysis," *Neural Networks, IEEE Transactions on* 10.3, 626-634, 1999.
- [8] J. F. MacGregor, T. Kourti ve P. Nomikos, "Analysis, monitoring and fault diagnosis of industrial processes using multivariate statistical projection methods," in *Proceedings of 13th IFAC World Congress, San Francisco, USA*, 1996.
- [9] B. Wise ve N. B. Gallagher, "The process chemometrics approach to process monitoring and fault detection," *Journal of Process Control* 6.6, 329-348, 1996.
- [10] D. Dong ve T. J. McAvoy, "Nonlinear principal component analysis—based on principal curves and neural networks," *Computers & Chemical Engineering* 20.1, 65-78, 1996.
- [11] A. Belouchrani ve ark., "A blind source separation technique using second-order statistics," *Signal Processing, IEEE Transactions on* 45.2, 434-444, 1997.

- [12] A. Yeredor, "Blind separation of Gaussian sources via second-order statistics with asymptotically optimal weighting," *IEEE Signal Processing Letters* 7.7, 197-200, 2000.
- [13] S. Ding ve ark., "On the application of PCA technique to fault diagnosis," *Tsinghua Science & Technology* 15.2, 138-144, 2010.
- [14] J. E. Jackson ve G. S. Mudholkar, "Control procedures for residuals associated with principal component analysis," *Technometrics* 21.3, 341-349, 1979.