AN ADAPTIVE LINE ENHANCER BASED ON THE CONVEX COMBINATION OF TWO IIR FILTERS

Walter J. Kozacky, Member, IEEE, and Tokunbo Ogunfunmi, Senior Member, IEEE

Santa Clara University, Santa Clara, CA, USA

ABSTRACT

An adaptive line enhancer is a self-tuning filter which attempts to retrieve a sinusoid buried in noise. Generally speaking, there is a trade-off between convergence speed and steady-state error. One method to address this is to use a variable step size algorithm, with a large step size for acquisition, and a smaller step size for improved steady-state performance. An alternate topology is based on the convex combination of two adaptive filters: a fast filter handles acquisition, while a slower filter is used to minimize error. In this paper, we present an adaptive line enhancer based on the convex combination of two IIR filters. It achieves both fast tracking and good steady-state performance, albeit at an increase in computational complexity.

***Index Terms*—**Adaptive filtering, adaptive line enhancer, convex combination

1. INTRODUCTION

The adaptive line enhancer (ALE) is a self-tuning filter which attempts to retrieve a sinusoid buried in noise [1-3]. An implementation using a forward prediction filter with a 1-step delay was presented in [2,3], and is shown if Fig. 1, (though other configurations are possible, e.g. [4-6]).

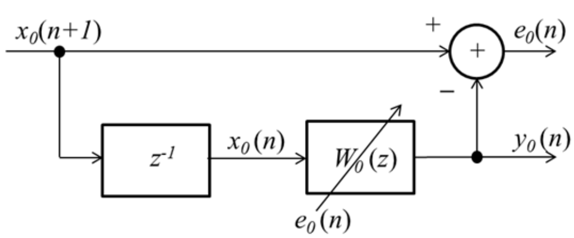

Fig. 1. Adaptive line enhancer based on a 1-step forward predictive adaptive filter.

The input signal $x_0(n)$ in this analysis consists of a single frequency sinusoid, and an uncorrelated zero-mean white Gaussian noise component $v(n)$, written as

$$x_0(n) = A_0 \sin(w_o n) + v(n). \tag{1}$$

Generally speaking, there is a trade-off between adaptive algorithm speed of convergence (i.e., in response to an input step change) and steady-state error. Fast convergence implies a large gradient step size to rapidly respond to input changes, which causes a larger fluctuation of the signal in the steady-state [7-9]. The approach taken in [2] was to use an infinite-impulse response (IIR) filter with a variable step size adjusted to maximize the output energy, using a filter structure for $W_0(z)$ as given in (2). (Details of this approach are given in [2], though other bandwidth modification approaches are possible, e.g. [10-13].) This paper extends the system in [2] to the convex combination topology in [13]. The filter structure used is given by

$$W_0(z) = \frac{[1-b(n)][\cos\theta(n) - z^{-1}]}{1-[1+b(n)]\cos\theta(n)z^{-1} + b(n)z^{-2}}. \tag{2}$$

In (2), parameter b controls the bandwidth, and parameter θ controls the center frequency. After convergence, $W_0(z)$ will have a band-pass characteristic, and the enhanced sinusoid will be present at output $y_0(n)$. To maximize the spectral enhancement of the input sinusoid, b is typically set near 1. Fig. 2 shows a set of plots of the mean-square error (MSE) using a fixed $\theta = \theta_0$, for various values of parameter b.

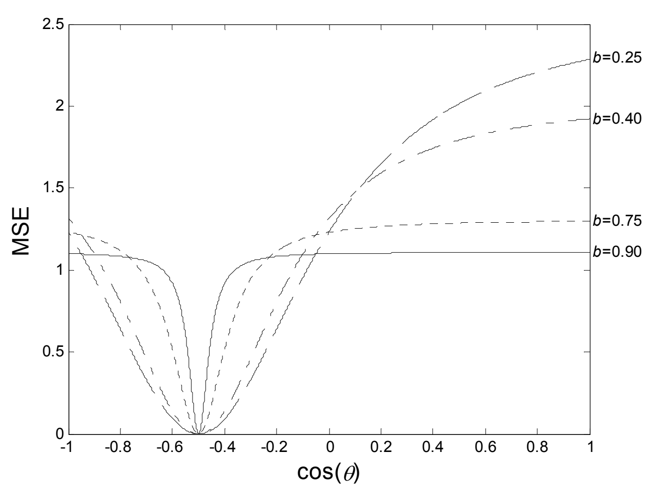

Fig. 2. Adaptive line enhancer MSE error plot for values of bandwidth parameter b.

As the MSE is a unimodal function of frequency, a gradient based technique can be used to tune the filter to the correct frequency. While a large value of b is preferred for sinusoid signal enhancement, when the input frequency differs from the filter center frequency the gradient is substantially reduced. (For example, compare the slope in Fig. 2 at $\cos(\theta_0) = 0.4$ for $b = 0.25$ and $b = 0.9$.) The gradient increases for smaller values of b, (wider bandwidth), which promotes rapid adaptation of θ using gradient based search techniques, while the spectral enhancement improves with larger values of b (narrower bandwidth). A strategy could be to use a large step size and wide bandwidth for acquisition, and a small step size and narrow bandwidth for spectral enhancement. In [2], the step size and bandwidth are dynamically increased to allow acquisition, and then automatically reduced for sinusoid enhancement, based on output power. The configuration in [2] will be referred to as the ALE controlled bandwidth model (ALECB), and will be used as a basis of comparison. Details of this algorithm are listed in Table 1. (Note: Trigonometric functions are treated as one multiply operation the first time they are used, as a look-up table implementation is assumed. Because there is only one filter in the ALECB, we have omitted the subscripts on $x_0(n)$, $y_0(n)$, $b_0(n)$, and $\theta_0(n)$.)

An alternate method uses a topology based on the convex combination of two (or more) adaptive filters [13 - 15]; a structure using two filters is shown in Fig. 3. These two filters will be collectively referred to as $W_0(z)$ in the system of Fig. 1.

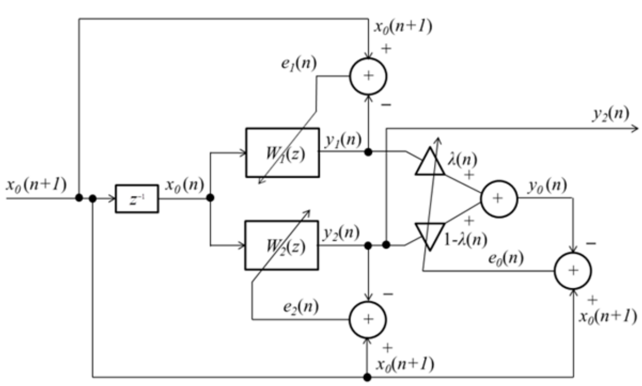

Fig. 3. Adaptive line enhancer based on a 1-step forward predictive adaptive filter implemented as a convex combination of two adaptive filters.

Here, $x_0(n)$ is the input at iteration n, $y_0(n)$ is the system output, $x_0(n+1)$ is the desired response, and $e_0(n)$ is the error term. Filters $W_1(z)$ and $W_2(z)$ operate independently, each using its own error signals for gradient adaptation. The filter outputs, $y_1(n)$ and $y_2(n)$ are then summed with weighting λ and $(1-\lambda)$ respectively. Parameter λ is adapted to minimize the total system error $e_0(n)$. In a convex combination λ is allowed to vary from 0 to 1, (though other topologies are also possible, e.g. [16-18]). $W_1(z)$ is a fast filter meant to quickly acquire a changing input signal, while $W_2(z)$ is a slower filter meant to provide a lower steady-state error. These filters can be of differing lengths or types depending on the application. The two independent outputs are combined based on minimizing the MSE of the system output $y_0(n)$. Combining the two independent filter outputs is usually accomplished by a sigmoid activation function using a secondary variable $\alpha(n)$ to calculate λ. (Details of the approach are given in [13].)

Since the two filters are completely decoupled, the combination will perform like the faster filter until the slower filter $W_2(z)$ achieves a lower MSE, so the overall convergence time will still be dependent on the slower filter. To achieve an overall "speeding up", a step-by-step weight transfer was used in [13], where a portion of the $W_1(z)$ weights were transferred to $W_2(z)$ when filter 1 is significantly outperforming filter 2 (e.g. in response to an input step change). To minimize gradient noise, this transfer is inhibited when λ is close to 1, so filter $W_2(z)$ is used exclusively. A plot of the combining functions is shown in Fig. 4, along with a straight line approximation $\lambda(n) = \mu_\lambda \alpha(n) + 0.5$ (dotted).

ALECB [2] algorithm processing	M=Mult A=Add
$\mathbf{u}_i(n) = [y(n-1) \quad y(n-2) \quad x(n) \quad x(n-1)]^T$ calculate $\cos\theta$, $\sin\theta$, from the current $\theta(n)$	2M
$\mathbf{w}(n) = [[1+b(n)]\cos\theta_i \quad -b \quad [1-b(n)]\cos\theta_i \quad -[1-b(n)]]^T$	3M+2A
$y(n) = \mathbf{w}^T(n)\mathbf{u}(n)$	4M+3A
$e(n) = x(n+1) - y(n)$	1A
$\mathbf{u}_\theta(n) = [y(n-1) \quad \nabla(n-1) \quad \nabla(n-2) \quad x(n)]^T$	
$\mathbf{w}_\theta(n) = [-[1+b(n)]\sin\theta \quad [1+b(n)]\cos\theta \quad -b(n) \quad [b(n)-1]\sin\theta]^T$	3M+2A
$\nabla_\theta(n) = \mathbf{w}_\theta^T(n)\mathbf{u}_\theta(n)$	4M+3A
$\mu_\theta(n) = \mu_{\theta 0}[1-b(n)]^3$	3M
$\theta(n+1) = \theta(n) + \mu_\theta(n)e(n)\nabla_\theta(n)$	2M+1A
$\mathbf{u}_b(n) = [\nabla_b(n-1) \quad \nabla_b(n-2) \quad e(n-1) \quad e(n-2)]^T$	
$\mathbf{w}_b(n) = [[1+b(n)]\cos\theta \quad -b \quad -\cos\theta \quad 1]^T$	1M
$\nabla_b(n) = \mathbf{w}_b^T(n)\mathbf{u}_b(n)$	4M+3A
$b(n+1) = b(n) + \mu_b\left[\frac{1}{[1-b(n)]^2}y^2(n) + \frac{1+b(n)}{1-b(n)}y(n)\nabla_b(n)\right]$	7M+2A
Total:	33M+ 17A

Table 1. ALECB algorithm [2] complexity calculation.

2. ADAPTIVE LINE ENHANCER BASED ON THE CONVEX COMBINATION OF TWO ADAPTIVE IIR FILTERS

Extending the concept of weight transfer and the "speeding up" mechanism from [13] and casting the convex combination system into a predictive filter, we have divided the system into two parts. The first uses the convex combination scheme to calculate potential new values for filter $W_2(z)$ parameter θ_2. The second section uses filter $W_2(z)$ to generate the output, which is taken exclusively from $W_2(z)$ and not from the two filter combination. Both filters use the update in (2) with fixed-bandwidth; parameter b_1 = 0.25 (wide bandwidth for acquisition) and b_2 = 0.9 (narrow bandwidth for filtering).

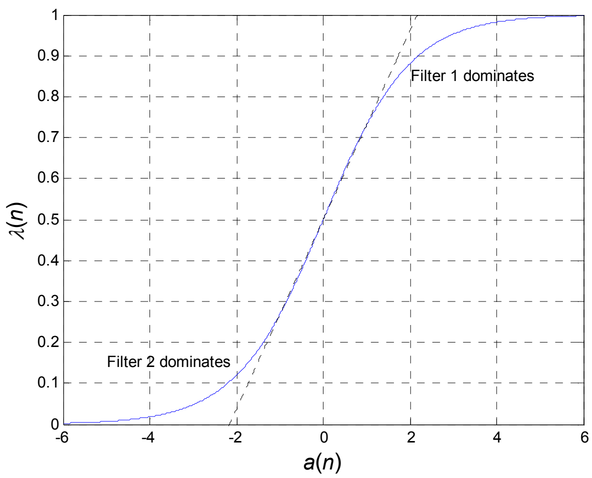

Fig. 4. Convex combination of two adaptive filters weighting transfer function and straight-line approximation (dotted).

The filter update for $W_1(z)$ to recursively adapt the center frequency is given by

$$\theta_1(n) = \theta_1(n-1) + \mu_1 e_1(n) \nabla_{\theta_1}(n). \tag{3}$$

As we are no longer mixing the two filter outputs, a simpler straight line approximation is used (per Fig. 4) and a threshold value (typically equal to or above the halfway point λ = 0.5) serving as a switch to initiate the transfer, resulting in a simpler mechanism given as

$$\begin{aligned} &\text{if } \lambda(n) > 0.5 \\ &\quad \theta_2(n) = \theta_1(n). \end{aligned} \tag{4}$$

Filter $W_2(z)$ now uses its "assisted" value of θ_2 for the gradient update. (Parameter $\alpha(n)$ is an intermediate variable used to develop a gradient update for $\lambda(n)$, as in [13]. An alternate speed-up mechanism is also given in [19].) We have $\mu_1 >> \mu_2$ to achieve both fast acquisition and narrow band operation. While $W_2(z)$ update depends only on $e_1(n)$, the $W_2(z)$ update depends on $e_1(n)$ when $\lambda(n)$ is larger than 0.5, and $e_2(n)$ otherwise. Details of the algorithm are listed in Table 2. (Note: Because parameter b is fixed in the convex combination structure, certain addition operations are not counted. Range checking of variables is not included in the algorithm description or complexity comparison calculations, as it is similar in both the convex combination and ALECB cases).

Convex combination algorithm processing for i=1,2:	M=Mult A=Add
$\mathbf{u}_i(n) = [y_i(n-1) \quad y_i(n-2) \quad x_0(n) \quad x_0(n-1)]^T$ calculate $\cos\theta_i$, $\sin\theta_i$, from the current $\theta_i(n)$	2M
$\mathbf{w}_i(n) = [(1+b_i)\cos\theta_i \quad -b_i \quad (1-b_i)\cos\theta_i \quad -(1-b_i)]^T$	3M
$y_i(n) = \mathbf{w}_i^T(n)\mathbf{u}_i(n)$	4M+3A
$e_i(n) = x_0(n+1) - y_i(n)$	1A
$\mathbf{u}_{\theta_i}(n) = [y_i(n-1) \quad \nabla_{\theta_i}(n-1) \quad \nabla_{\theta_i}(n-2) \quad x_0(n)]^T$	
$\mathbf{w}_{\theta i}(n) = [-(1+b_i)\sin\theta_i \quad (1+b_i)\cos\theta_i \quad -b_i \quad (b_i-1)\sin\theta_i]^T$	4M
$\nabla_i(n) = \mathbf{w}_{\theta_i}^T(n)\mathbf{u}_{\theta_i}(n)$	4M+3A
$\theta_i(n+1) = \theta_i(n) + \mu_i e_i(n) \nabla_{\theta_i}(n)$	2M+1A
end Subtotal for algorithm for each filter:	19M +8A
Output processing:	
$y_0(n) = \lambda(n) y_1(n) + [1-\lambda(n)] y_2(n)$	2M+2A
$e_0(n) = x_0(n+1) - y_0(n)$	1A
$\alpha(n) = \alpha(n-1) + \mu_\alpha e_0(n)[e_2(n) - e_1(n)][\lambda(n)][1-\lambda(n)]$	4M+3A
$\lambda(n) = \mu_\lambda \alpha(n) + 0.5$	1M+1A
if $\lambda(n) > 0.5$ $\theta_2(n) = \theta_1(n)$	2A
Output processing subtotal:	7M+9A
2 x algorithm processing subtotal:	38M+ 16A
Total:	45M+ 25A

Table 2. Convex combination algorithm update and complexity calculation.

3. SIMULATIONS

Both the ALECB and the convex combination parameter values are experimentally selected to yield stable operation for a SNR of -3dB. Reducing the slope of the straight line approximation (Fig. 4) resulted in better performance at low SNR values (i.e., smaller $\alpha(n)$ in Table 3). Fig. 5 illustrates the operation for an input SNR of +10dB with a rapidly changing input and a sampling frequency of 2000 Hz.

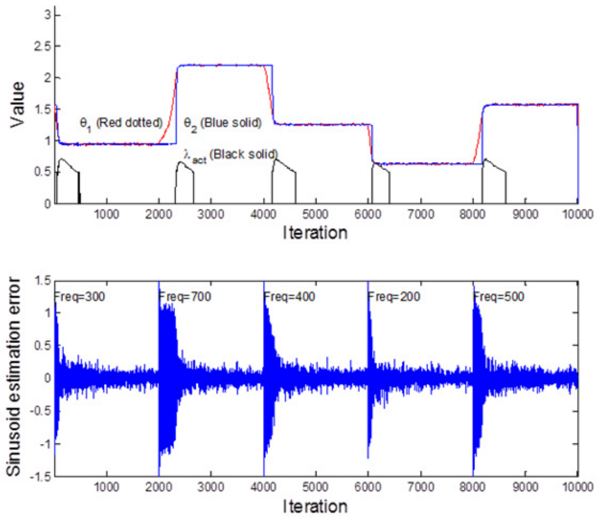

Fig. 5. Convex combination of two adaptive filters algorithm operation.

The top graph of Fig. 5 displays θ_1, θ_2 and $\lambda(n)$. The narrow bandwidth $W_2(z)$ processes the steady-state condition (with a steady-state behavior similar to the ALECB), but is assisted by $W_1(z)$ during input step changes. The bottom trace displays the sinusoid estimation error (the difference between the sinusoid portion of (1) and the $y_2(n)$ output) for input frequency changes every 2000 iterations with a sampling frequency of 2000 Hz. The value of adaptive filter tuned frequency will be used for comparison with the ALECB, illustrated in Fig. 6 for an SNR of +10dB, and in Fig. 7 for an SNR of -3dB. The convex combination algorithm achieves faster convergence for all input step changes.

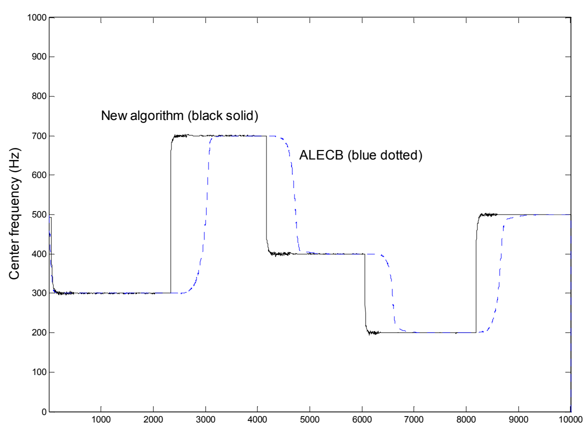

Fig. 6. Convex combination and ALECB algorithm comparison, SNR = +10dB.

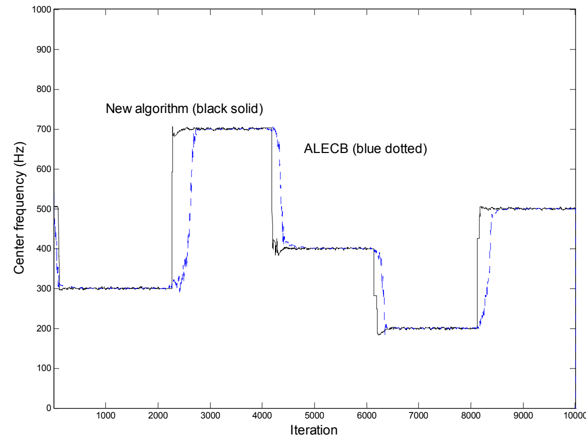

Fig. 7. Convex combination and ALECB algorithm comparison, SNR = -3dB.

4. CONCLUSION

A new structure for adaptive line enhancement, based on the convex combination of two IIR filters, was developed. The computational complexity and sinusoidal acquisition performance of the new structure was compared with the ALECB structure. The new structure provides faster convergence in response to an input frequency change, albeit at an increase in computational complexity. To mitigate the increase in complexity, a simplified mixing mechanism was developed. The system output is taken exclusively from the narrow bandwidth filter, and a straight-line transfer function is used to assist the narrow bandwidth adaptive filter in acquisition. Expansion to multiple input frequencies [20-22] with this structure is an area of further research.

5. REFERENCES

[1] B. Widrow and S. D. Stearns, *Adaptive Signal Processing*, Upper Saddle River, NJ, Prentice-Hall, 1985.

[2] B. Farhang-Boroujeny, "An IIR adaptive line enhancer with controlled bandwidth", *IEEE Transactions on Speech and Audio Processing*, vol. 45, no. 2, pp. 477-481, Feb. 1997.

[3] B. Farhang-Boroujeny, *Adaptive Filters: Theory and Applications*, John Wiley & Sons, West Sussex, England, 1998.

[4] J. E. Cousseau, S. Werner, and P. D. Donate, "Factorized all-pass based IIR adaptive notch filters", *IEEE Transactions on Signal Processing*, vol. 55, no. 11, pp. 5225-5236, Nov. 2007.

[5] A. David, S. D. Stearns, G. R. Elliott, and D. M. Etter, "IIR algorithm for adaptive line enhancement", *Proceedings ICASSP 1984,* pp. 17-20, April 1983.

[6] D. R. Hush, N. Ahmad, R. David, and S. D. Stearns, "An adaptive IIR structure for sinusoidal enhancement, frequency estimation, and detection", *IEEE Transactions on Acoustics, Speech and Signal Processing*, Vol. 34, No. 6, pp. 1380-1390, June 1986.

[7] S. Haykin, "*Adaptive Filter Theory*", Prentice-Hall, Upper Saddle River, NJ, 2002.

[8] J. Okello, K. Ueda, Y. Itoh, and H. Ochi, "An IIR ALE based on constrained FIR filter", *Proceedings ICSP 2002*, pp. 281-284.

[9] T. Aboulnasr and K. Mayyas, "A robust variable step-size LMS-Type algorithm: analysis and simulations", *IEEE Transactions of Signal Processing,* vol. 45, no. 3, pp. 631-639, Mar. 1997.

[10] W. J. Kozacky and T. Ogunfunmi, "An adaptive IIR filter with constraints on the output power level", *Proceedings of the Forty-Fourth Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, Nov. 2010, pp. 984-987.

[11] P. A. Regalia, S.K. Mitra, and P. P. Vaidyanathan, "The digital all-pass filter: A versatile signal processing building block", *Proceedings of the IEEE*, Vol. 76, No. 1, pp. 19-37, January 1988.

[12] S. J. Ban, C. W. Lee, H. Cho, S. W. Kim, "A variable step-size algorithm for direct frequency estimation", S*ignal Processing*, Vol. 90, No. 9, pp. 2800-2805, September 2010.

[13] J. Arenas-Garcia, M. Martinez-Ramon, A. Navia-Vazquez, and A. R. Figueiras-Vidal, "Plant identification via adaptive combination of transversal filters", *Signal Processing*, vol. 86, no. 9, pp. 2430-2438, Sept. 2005.

[14] J. Arenas-Garcia, A. R. Figueiras-Vidal, and A. H. Sayed, "Mean-Square performance of a convex combination of two adaptive filters", I*EEE Transactions on Signal Processing*, vol. 54, no. 3, pp. 1078-1090, Mar. 2006.

[15] Y. Zhang and J. A. Chambers, "Convex combination of adaptive filters for a variable tap-length LMS algorithm", *IEEE Signal Processing Letters*, vol. 13, no. 10, pp. 628-631, Oct. 2006.

[16] N. J. Bershad, J. C. M. Bermudez, and J. Y. Tourneret, "An affine combination of two LMS adaptive filters-transient mean-square analysis", *IEEE Transactions on Signal Processing,* vol. 56, no. 5, pp. 1853-1864, May 2008.

[17] L. A. Azpicueta-Ruiz, A. R. Figueiras-Vidal, J. Arenas-Garcia, "A new least squares adaptation scheme for the affine combination of two adaptive filters", *Proceedings of the IEEE Workshop on Machine Learning and Signal Processing*, pp. 327-332, 2008.

[18] M. Ferrer, A. Gonzalez, M. de Diego, and G. Pinero, "Convex combination filtered-X algorithms for active noise control systems", *IEEE Transactions on Audio, Speech, and Language Processing*, Vol. 21, No. 1, pp. 154-165, January 2013.

[19] V. H. Nascimento and R. C. deLamare, "A low-complexity strategy for speeding up the convergence of convex combinations of adaptive filters", *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 3553-3556, May 2012.

[20] T. Kwan and K. Martin, "Adaptive detection and enhancement of multiple sinusoids using a cascade IIR filter", *IEEE Transactions on Circuits and Systems*, Vol. 36, No. 7, pp. 937-947, July 1989.

[21] G. Kannan, A. A. Milani, and I. Panahi, "Active noise control of noisy periodic signals using signal separation", *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 1617-1620, May 2008.

[22] Y. Xiao, J. Wang, and H. Wei, "A new hybrid active noise control system in the presence of wideband and narrowband noise components", *Proceedings of the 2011 International Conference on Advanced Mechatronic Systems*, pp. 357-361, August 2011.