

## 論文

J. of The Korean Society for Aeronautical and Space Sciences 45(10), 863-871(2017)

DOI:https://doi.org/10.5139/JKSAS.2017.45.10.863

ISSN 1225-1348(print), 2287-6871(online)

## 헬리콥터 능동진동제어시스템 가속도 신호 처리

김도형\*

Accelerometer Signal Processing  
for a Helicopter Active Vibration Control System

Do-Hyung Kim\*

Korea Aerospace Research Institute\*

## ABSTRACT

LMS (least mean square) algorithm widely used in the AVCS (active vibration control system) of helicopters calculates control input using the forward path transfer function and error signal. If the error signal is sinusoidal, it can be represented as the combination of cosine and sine functions with frequency and phase synchronized with the reference signal. The control input also has the same frequency, therefore control algorithm can be simply implemented if the cosine and the sine amplitudes of the control input are calculated and the frequency and phase of the reference signal are used. Calculation of the control input is implemented as simple matrix operation and the change of the control command is slower than the frequency of the error signal, consequently control algorithm can be operated at lower frequency. The signal processing algorithm extracting cosine and sine components of the error signals are modeled using Simulink and PIL (processor-in-the-loop) mode simulation was executed for real-time performance evaluation.

## 초 록

헬리콥터 능동진동제어시스템에 널리 이용되는 LMS (least mean square) 알고리즘은 전방경로 (forward path) 전달함수와 에러 신호의 연산을 통해 제어 입력을 계산한다. 에러 신호가 정현파 형태일 경우, 기준 신호에 동기화된 진동수, 위상을 가지는 코사인과 사인 함수의 조합으로 표현될 수 있다. 제어 신호 또한 동일한 진동수를 가지게 되므로 제어 입력의 코사인, 사인 성분의 크기만 계산하고, 기준 신호의 진동수, 위상 정보를 활용하면 제어알고리즘은 단순하게 구현될 수 있다. 제어 입력 신호의 계산은 단순한 행렬 연산으로 구현되고, 제어 명령의 변화는 에러 센서의 주파수에 비해 느리기 때문에 제어알고리즘은 낮은 주파수에서 운용 가능하다. 에러 센서의 코사인, 사인 성분을 추출하는 신호처리 알고리즘을 시뮬링크 모델로 구현하고, PIL (processor in the loop) 모드 시뮬레이션을 통해 실시간 작동 성능을 평가하였다.

**Key Words :** Active Vibration Control System(능동진동제어시스템), Least-Mean-Square, Signal Processing (신호처리), Processor In the Loop, Execution Timing Analysis (실행 시간 분석)

† Received : June 5, 2017    Revised : September 11, 2017    Accepted : September 13, 2017

\* Corresponding author, E-mail : dhkim@kari.re.kr

## I. 서론

헬리콥터 동체 진동의 주성분은 주로터에서 발생되어 동체로 전달되는  $N/\text{rev}$ 의 ( $N$ 은 블레이드 개수) 진동수를 가지며 [1,2], 기본적으로 기체구조 및 주요 구성품의 동특성이  $N/\text{rev}$  진동수와 공진이 발생되지 않도록 설계 시 고려한다 [3]. 항공기 주요 구조물의 공진회피 설계 외에 진동을 해결하기 위해 수동형 및 능동형 진동저감 기법을 적용할 수 있다. 수동형 진동저감 기법은 외부의 에너지 유입 없이 진동에 반대되는 힘(force)이나 움직임(motion)을 발생시켜 진동을 흡수(absorption) 또는 절연(isolation)시키는 것을 말한다. 능동형 진동저감 기법은 외부의 에너지를 이용하여 진동에 반대되는 제어 가능한 힘을 발생시켜 진동을 상쇄(cancellation)시키는 방법이다. 종래의 헬리콥터에서는 수동형 진동저감 장치가 주로 사용되어 왔으나, 1990년대 이후로 가속도계와 하중발생기를 이용한 능동진동제어시스템(active vibration control system, AVCS)[4]이 상용화 되어 S-92, V-22, UH-60M, S-76D, AW139, EC225/725 등의 헬리콥터에 적용되고 있다. AVCS는 수동형 진동저감 장치에 비해 작은 중량으로 보다 넓은 주파수 영역에서 우수한 진동저감 성능을 발휘할 수 있는 장점이 있다[5].

국내에서도 최근 AVCS를 적용하기 위한 연구개발을 활발히 수행하고 있다. 2013년에는 수리온을 대상으로 AVCS 성능 시연 프로그램을 통해 AVCS 배치 최적화가 수행된 바 있으며[5], 실험실 수준에서 진동제어 성능에 대한 연구도 수행된 바 있다[6]. 현재 소형민수헬기(light civil helicopter, LCH) 핵심기술 개발과제에서는 AVCS 제어알고리즘 소프트웨어 개발, 시스템 통합 및 최적화 관련 연구개발을 수행하고 있다.

AVCS의 주요 구성품은 진동 측정을 위한 가

속도계, 제어 입력을 계산하는 제어기, 하중을 발생시키는 작동기, 기준신호 측정을 위한 타코미터이며, 전체 시스템 구성은 Fig. 1과 같이 도식화 할 수 있다. AVCS에 적용되고 있는 제어 기법은 Fx-LMS (filtered-x least mean square)[7] 알고리즘에 기반을 두고 있으며, 기본적인 형태는 Fig. 1의 아래쪽 점선 부분과 같다. 타코미터로 측정하는 기준 신호  $x_k$ 와 가속도계로 측정하는 에러 신호  $e_k$ 를 이용하여 업데이트된 제어 입력  $u_k$ 를 작동기로 보낸다. 제어 입력  $u_k$ 로 인해 생성된 진동  $\hat{y}_k$ 가 제어하지 않을 때 발생하는 진동  $y_k$ 를 상쇄시키도록 만드는 것이 기본 원리이다. 여기서  $C^*$ 는 제어 입력  $u_k$ 와 작동기로 발생시킨 진동  $\hat{y}_k$  간의 관계를 나타내는 전방 경로(forward path) 전달함수로 작동기의 동특성을 표시하며,  $C$ 는 지상시험을 통해 획득한 시스템식별 결과로써  $C^*$ 에 대한 수학적 모델이다.  $H^*$ 는 주로터에 의해 유발되는 동체진동을 나타내는 교란 경로(disturbance path) 전달함수로써 동체의 동특성을 나타낸다. 하첨자  $k$ 는  $k$ 번째 시간 스텝을 의미한다.

진동제어를 위해서 기준 신호와 에러 신호는 제어 대상 진동수 보다 충분히 높은 주파수로 샘플링 되어야 한다. 제어알고리즘을 센서 샘플링과 동일한 빠르기로 구동시키는 것은 가능하지만 임베디스 시스템으로 구현 시 많은 리소스를 필요로 한다. 헬리콥터 동체 진동의 주성분인  $N/\text{rev}$  진동수의 변화는 크지 않기 때문에 제어 알고리즘에서 계산되는 제어 입력의 주파수 또한 변동이 크지 않다. 이러한 특성을 이용하여 센서 신호처리는 높은 주파수로 작동하고, 진동제어 입력은 낮은 주파수로 작동하도록 설계할 수 있다. 또한 센서 신호처리와 제어 알고리즘을 분리함으로써 제어기를 작동하지 않을 경우에도 진동 정보를 개별적으로 획득할 수 있다. 본 연구에서는 낮은 주파수로 구동할 수 있는 진동제어 알고리즘과 제어 입력 계산에 필요한 가속도 신호처리 알고리즘을 구성하였다. 가속도 신호처리 알고리즘은 시뮬링크를 이용하여 모델링하고 PIL(processor in the loop) 모드 시뮬레이션을 통해 실시간 작동 성능을 평가하였다.

## II. 알고리즘

### 2.1 진동제어 알고리즘

Figure 1에 도식한 에러 신호  $e_k$ 는 가속도계를 통해 측정되는 가속도 신호이다. 측정된 가속

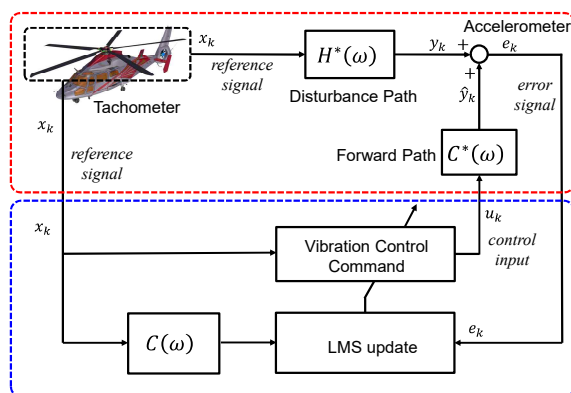


Fig. 1. System configuration

도 신호는 대역통과 필터를 이용하여 N/rev 성분 이외의 잡음을 제거하게 되므로 필터된 신호는 N/rev 주파수의 정현파 형태가 되고 다음과 같이 표현할 수 있다.

$$e_k = E_{C_k} \cos(2\pi f k T_s) + E_{S_k} \sin(2\pi f k T_s) \quad (1)$$

$$= E_{C_k} \cos(2\pi (\frac{f}{f_s}) k) + E_{S_k} \sin(2\pi (\frac{f}{f_s}) k)$$

여기서,  $T_s$ 는 샘플링 주기,  $f_s$ 는 샘플링 주파수,  $f$ 는 제어 대상인 N/rev 진동수를 나타내며,  $2\pi(f/f_s)k$ 는 기준 신호  $x_k$ 로 타코미터로 측정된 방위각(azimuth angle)에 해당한다. 에러 신호를 기준 신호를 이용하여 식 (1)과 같이 수학적 으로 표현하기 위해서는 코사인, 사인 성분의 크기를 나타내는  $E_{C_k}$ 와  $E_{S_k}$ 의 식별이 필요하다.

P개의 가속도 센서와 M개의 작동기로 구성된 진동제어 시스템을 고려하면, k번째 시간 스텝에서의 에러 신호는 다음과 같이 나타낼 수 있다.

$$E_k = C^*(\omega) U_k + Y_k \quad (2)$$

$$\approx C(\omega) U_k + Y_k$$

$[P \times 1] \quad [P \times M][M \times 1] \quad [P \times 1]$

$E_k$ 는 P개의 에러 신호 벡터로써  $E_{C_k} + iE_{S_k}$ 와 같이 복소수로 표현할 수 있다.  $U_k$ 는 M개의 제어 입력 벡터로  $U_{C_k} + iU_{S_k}$ 로 표현하고,  $Y_k$ 는 교란 경로를 통해 P 개의 가속도계로 측정되는 진동 신호 벡터로 동일하게  $Y_{C_k} + iY_{S_k}$ 로 표현할 수 있다. 전방 경로 전달함수는 가진 주파수에 따라 달라지므로  $C(\omega)$ 로 표현하였다. 여기서  $\omega$ 는 진동수를 나타낸다. 로터회전수 변화에 따른 항공기 진동 특성 변화에 대처하기 위해서는 주로터의 운용 주파수를 포함하는 주파수 대역에서 전방 경로 전달함수를 획득하는 것이 필요하다.

최적의 제어 입력을 계산하기 위해 평균제곱 오차(mean square error, MSE)를 목적함수로 설정하였다.

$$J_k = E_k^H E_k \quad (3)$$

$[1 \times 1] \quad [1 \times P][P \times 1]$

식 (3)의 목적함수를 최소화시키는 조건은 다음과 같이 계산할 수 있다.

$$\frac{\partial J_k}{\partial U_k} = 0 \quad (4)$$

$$C^H(CU_k + Y_k) = 0$$

$$C^H C U_k + C_k^H Y_k = 0$$

식 (4)의 해가 최적 제어 입력이다.

$$U_{opt} = -(C^H C)^{-1} C^H Y_k \quad (5)$$

$[M \times 1] \quad [M \times M] \quad [M \times P][P \times 1]$

식 (5)를 살펴보면 최적 제어 입력은 전방 경로 전달함수와 교란 경로 벡터의 연산으로 구성되어 있다. 즉, 특정 비행 상태에서 발생하는 진동(교란 경로)과 작동기 동특성(전방 경로)에 대한 전달함수를 획득하면, 해당 비행 상태에서의 최적 제어 입력을 계산할 수 있다.

그러나 가속도계로 측정하는 에러 신호는 교란 경로 신호 Y와 전방 경로에 의해 유발되는 진동 CU가 혼합된 신호이므로 식 (5)의 최적 제어 입력을 실시간 제어시스템에 바로 적용할 수 없다. 실시간 제어 시스템으로 구현하기 위해서 매 시간 스텝마다 제어 입력이 최적 해를 지향하도록 경사 하강법(gradient descent)을 이용한 제어 입력 업데이트 방법을 적용할 수 있다.

$$U_{k+1} = U_k - \frac{\mu}{2} \left( \frac{\partial J_k}{\partial U_k} \right) \quad (6)$$

$$= U_k - \mu C(\omega)^H E_k$$

$[M \times 1] \quad [M \times 1] \quad [1 \times 1][M \times P][P \times 1]$

여기서,  $\mu$ 는 스텝 사이즈를 조절하는 파라미터이다.  $C(\omega)$ 는 복소수 행렬이고,  $E_k$ 는 복소수 벡터이므로 계산되는  $U_k$  또한 복소수 벡터이다. 계산된 제어 입력의 실수부는 코사인성분, 허수부는 사인성분의 크기를 나타낸다. 식 (6)의 연산은 복소수 행렬 연산이지만 실수부와 허수부를 분리하여 제어 입력의 코사인 성분과 사인 성분을 각각 계산할 수도 있다. 계산된  $U_k$ 는 기준 신호를 이용하여 식 (1)과 같은 형태로 시간영역에서 재구성 되어 사용된다.

$$u_k = U_{C_k} \cos(2\pi (\frac{f}{f_s}) k) + U_{S_k} \sin(2\pi (\frac{f}{f_s}) k) \quad (7)$$

정현파 형태의 에러 신호에 대해서 계산되는 제어 입력은 정현파 형태로 나타나고, 진동수 변화가 빠르지 않을 경우 코사인, 사인 성분의 크기 변화는 에러 신호 주파수에 비해 매우 느리기 때문에 제어알고리즘의 업데이트는 낮은 주파수에서 구현 가능하다.  $C(\omega)$ 는 지상시험으로 획득 가능하므로 식 (6)의 제어 입력을 실시간으로 계산하기 위해서는 에러 신호 식별, 즉  $E_{C_k}$ 와  $E_{S_k}$ 를 실시간으로 추출하는 신호처리 알고리즘이 필요하다.

## 2.2 가속도 신호처리 알고리즘

식 (1)로 표현된 에러 신호의  $E_{C_k}$ 와  $E_{S_k}$ 는 이산 푸리에 변환으로 계산할 수 있다.

$$\hat{E}_{C_k} = \left( \frac{2T_s}{T} \right) \sum_1^N e_k \cos(2\pi \left( \frac{f}{f_s} \right) k) \quad (8)$$

$$\hat{E}_{S_k} = \left( \frac{2T_s}{T} \right) \sum_1^N e_k \sin(2\pi \left( \frac{f}{f_s} \right) k) \quad (9)$$

여기서  $\hat{E}_{C_k}$ ,  $\hat{E}_{S_k}$ 는 이산 푸리에 변환으로 추정된  $E_{C_k}$ ,  $E_{S_k}$ 를 의미하고,  $N$ 은 샘플 수,  $T$ 는 신호처리에 사용되는 데이터 전체의 시간을 나타낸다. 식 (8), (9)의 합산식 내부의 항에 각각 식 (1)을 대입하면 다음과 같이 전개된다.

$$[e_k \cos(2\pi \left( \frac{f}{f_s} \right) k)] \quad (10)$$

$$\begin{aligned} &= E_{C_k} \cos^2(2\pi \left( \frac{f}{f_s} \right) k) \\ &\quad + E_{S_k} \sin(2\pi \left( \frac{f}{f_s} \right) k) \cos(2\pi \left( \frac{f}{f_s} \right) k) \\ &= \frac{1}{2} [E_{C_k} [1 + \cos(4\pi \left( \frac{f}{f_s} \right) k)] + E_{S_k} \sin(4\pi \left( \frac{f}{f_s} \right) k)] \\ &= \frac{1}{2} [E_{C_k} + E_{C_k} \cos(4\pi \left( \frac{f}{f_s} \right) k)] + E_{S_k} \sin(4\pi \left( \frac{f}{f_s} \right) k) \end{aligned}$$

$$[e_k \sin(2\pi \left( \frac{f}{f_s} \right) k)] \quad (11)$$

$$\begin{aligned} &= E_{C_k} \cos(2\pi \left( \frac{f}{f_s} \right) k) \sin(2\pi \left( \frac{f}{f_s} \right) k) \\ &\quad + E_{S_k} \sin^2(2\pi \left( \frac{f}{f_s} \right) k) \\ &= \frac{1}{2} [E_{C_k} \sin(4\pi \left( \frac{f}{f_s} \right) k)] + E_{S_k} [1 - \cos(4\pi \left( \frac{f}{f_s} \right) k)] \\ &= \frac{1}{2} [E_{S_k} + E_{C_k} \sin(4\pi \left( \frac{f}{f_s} \right) k)] - E_{S_k} \cos(4\pi \left( \frac{f}{f_s} \right) k) \end{aligned}$$

식 (10), (11)을 각각 식 (8), (9)에 대입하면 다음과 같이 나타낼 수 있다.

$$\begin{aligned} \hat{E}_{C_k} &= \left( \frac{NT_s}{T} \right) E_{C_k} \\ &\quad + \left( \frac{T_s}{T} \right) [E_{C_k} \sum_1^N \cos(4\pi \left( \frac{f}{f_s} \right) k) \\ &\quad + E_{S_k} \sum_1^N \sin(4\pi \left( \frac{f}{f_s} \right) k)] \end{aligned} \quad (12)$$

$$\begin{aligned} \hat{E}_{S_k} &= \left( \frac{NT_s}{T} \right) E_{S_k} \\ &\quad + \left( \frac{T_s}{T} \right) [E_{C_k} \sum_1^N \sin(4\pi \left( \frac{f}{f_s} \right) k) \\ &\quad - E_{S_k} \sum_1^N \cos(4\pi \left( \frac{f}{f_s} \right) k)] \end{aligned} \quad (13)$$

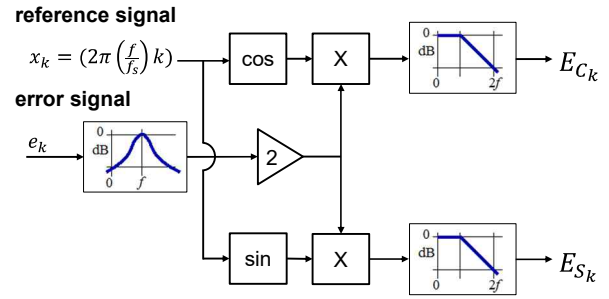


Fig. 2. Accelerometer signal processing algorithm

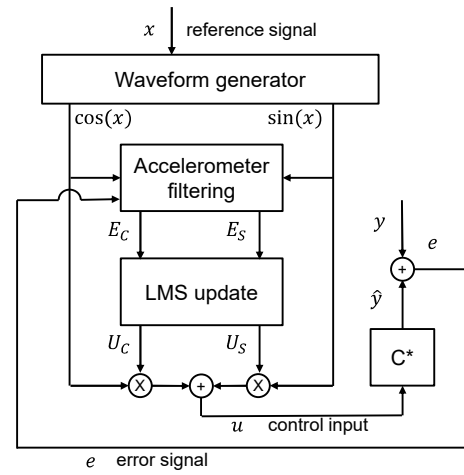


Fig. 3. Vibration control algorithm

이 때, 샘플링 주기가 일정할 경우  $T = NT_s$  이므로 식 (12)와 (13)은  $E_{C_k}$ ,  $E_{S_k}$  및  $2f$  주파수 성분의 합으로 구성될 수 있다. 신호처리에 사용되는 데이터 개수가 매우 많으면 ( $T \gg T_s$ )  $2f$  주파수 성분의 영향은 작아지고  $\hat{E}_{C_k} \approx E_{C_k}$ ,  $\hat{E}_{S_k} \approx E_{S_k}$ 인 결과를 얻을 수 있다. 그러나 실시간 신호처리에 있어서 많은 양의 데이터를 사용하는 것은 효과적이지 않다.

식 (10), (11)을 다시 살펴보면, 식 (12), (13)과 마찬가지로  $E_{C_k}$ ,  $E_{S_k}$  정보를 포함하는 DC 성분과  $2f$  주파수 성분으로 구성되어 있음을 알 수 있다. 여기서 주목할 사항은, 추출하려는 정보는 DC 성분, 불필요한 정보는  $2f$  주파수 성분으로 주파수 영역에서 이격되어 있고, 식 (12), (13)과 달리 합계 식이 없어 동일한 시간 스텝의 정보로 계산 가능하다는 점이다. 이 신호에 저주파 통과 필터를 적용하면 DC 성분에 있는 정보, 즉  $E_{C_k}$ ,  $E_{S_k}$ 를 쉽게 추출할 수 있다. 이 때, 식 (10), (11)의 신호를 저주파 통과 필터를 이용할 경우

$E_{C_k}, E_{S_k}$  신호가 1/2 크기로 추출되므로 곱하기 2의 연산이 필요하다. 그리고 가속도계 신호는 전기적 잡음과 불필요한 주파수 성분을 포함하고 있으므로 대역통과 필터를 이용한 잡음 제거 과정은 선처리 과정으로 필요하다. 이상의 가속도 신호처리 알고리즘을 도식화 하면 Fig. 2와 같다.

본 연구의 신호처리 및 진동제어 알고리즘을 전체적으로 도식화하면 Fig. 3과 같다. 여기서 신호처리 알고리즘은 코사인, 사인 파형 발생기 (waveform generator)와 가속도 필터링 (accelerometer filtering) 블록으로 표현된다.

### III. 신호처리 모델 설계

#### 3.1 대역통과 필터 설계

대역통과 필터는 다양한 형태로 설계될 수 있으나, 본 연구에서는 실시간 제어 시스템에 적용하기 위해 2차의 저주파 통과 필터와 2차의 고주파 통과 필터의 조합으로 구현하여 통과대역의 컷오프 주파수를 개별적으로 조절할 수 있도록 구현하였다. 설계의 편의를 위해 아날로그 필터를 설계한 후에 디지털 IIR(Infinite impulse response) 필터로 변환하여 사용하였다. 저주파 통과 필터와 고주파 통과 필터의 전달함수는 식 (14), (15)와 같다.

$$H(s) = \frac{Gain}{(1/\omega_0^2)s^2 + (1/\omega_0 Q)s + 1} \quad (14)$$

$$H(s) = \frac{Gain \times s^2}{s^2 + (\omega_0/Q)s + \omega_0^2} \quad (15)$$

여기서,  $\omega_0$ 는 컷오프 주파수,  $Q$ 는 필터 감쇠특성을 조절하는 Q-factor이다. 통과 대역의 중심 주파수에서 0 dB의 특성을 가지도록 Gain을 조절하였다. 필터 설계에 적용한 파라미터는 Table 1과 같다. 통과 대역을 결정하는 컷오프 주파수는 N/rev을 기준으로  $\pm 0.6/\text{rev}$  수준의 통과 대역을 가지도록 설정하여 전기적 잡음뿐만 아니라 로터 회전수의 하모닉 성분도 필터링 되도록 하였다. 본 연구의 적용 대상인 LCH의 경우 N=5,

Table 1. Filter design parameters

Parameters	$\omega_0$ (rad/sec)	$Q$	Gain
low pass	205.25	4	0.15135
high pass	157.58	4	0.6625

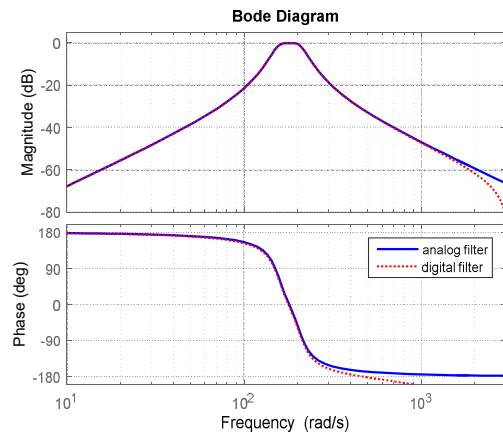


Fig. 4. Frequency response of the designed band-pass filter

로터회전수(1/rev)는 342~350 rpm 이다. 설계된 아날로그 필터를 1 kHz로 작동하는 디지털 필터로 변환하여 사용하였고, 주파수 응답 특성은 Fig. 4와 같다.

#### 3.2 저주파 통과 필터 설계

가속도 신호처리 알고리즘에서  $E_{C_k}, E_{S_k}$  정보를 추출하기 위한 저주파 통과 필터는 식 (14)에서  $Q=0.707$ , 컷오프 주파수  $\omega_0$ 는 1/rev 수준의 주파수 값인 35.81 rad/sec (= 5.7 Hz), Gain=1을 적용하여 설계하였다.  $f=29.17$  Hz,  $f_s=1$  kHz 일 때, DC 오프셋 0.05를 가지고  $E_C=0.3, E_S=-0.2$  인 예러 신호에 설계된 저주파 통과필터를 적용하여  $E_{C_k}, E_{S_k}$  정보를 추출하는 시뮬레이션을 수행한 결과를 Fig. 5에 나타내었다. 신호처리 알고리즘은 0.2초 후에 수렴된  $E_C, E_S$  값을 출력해 주고 있다. 또한 예러 신호에 포함된 DC 성분 잡음은  $E_{C_k}, E_{S_k}$ 에 나타나지 않았다. DC

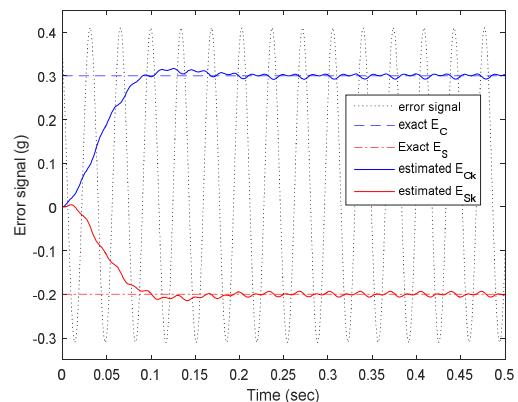


Fig. 5.  $E_c, E_s$  components extraction

성분의 잡음은 식 (10), (11)의 계산 과정에서 29.17 Hz 주파수 성분으로 변환되므로 저주파 통과필터의 컷오프 주파수 5.7 Hz 보다 높아서 차단되기 때문이다.

### 3.3 시뮬링크 모델링

본 연구의 알고리즘을 적용할 대상인 LCH의 로터 회전수 350 rpm 기준, N/rev 주파수는 29.17 Hz 이다. 이에 대해 40 Hz의 속도로 작동하는 제어알고리즘을 설계하였으며, 시뮬레이션을 통해 진동저감 성능을 확인한 바 있다[8]. 그러나 제어기 작동 주파수와 동일한 40 Hz로 샘플된 가속도 데이터로는 29.17 Hz의 진동을 식별할 수 없다. LCH AVCS는 센서 샘플링 주파수로 1 kHz를 사용한다. 여기서 실시간 제어 시스템 구현을 위한 두 가지 방법을 생각할 수 있다. 첫 번째는 가속도 신호처리 알고리즘은 빠른 속도로 작동시키고, 제어 알고리즘은 40 Hz로 천천히 작동시키는 방법이다. 두 번째는 신호처리 알고리즘을 제어기와 동일하게 40 Hz로 작동시키지만 1 kHz로 샘플된 데이터를 모두 이용하는 것이다. 본 연구에서는 두 번째 방법을 적용하여 샘플된 데이터를 25개씩 모아서 신호처리 함수로 보내어 정보의 손실 없이 40 Hz 속도로 작동하는 모델을 구성하였다. LCH AVCS의 경우 14개의 가속도계를 사용하므로 입력되는 가속도 신호는 매 스텝마다 14×25=350개이다. 시뮬링크에서 매 스텝마다 14×25개의 가속도 데이터를 연속적으로 처리하기 위해서 For Iterator Subsystem 블록을 사용하였다. 25 msec (40 Hz) 마다 14×1 개의 가속도 데이터를 처리하는 작업을 25회 수행하므로, 1 msec (1 kHz)의 속도로 14×1 개의 가속도 데이터를 처리하는 것과 동일

한 기능을 발휘한다.

신호처리 알고리즘에 사용되는 4 개의 2차 IIR 필터는 동일한 구조를 가진다. 시뮬링크에서는 Fig. 6과 같이 필터링 할 신호, 필터의 분자와 분모 계수를 입력으로 받아 연산을 수행하고 필터링 된 신호를 출력하는 형태로 모델링하였다. 모델에 사용된 블록은 Product, Add 및 UnitDelay 세 가지이다.

## IV. PIL 모드 시뮬레이션

### 4.1 시뮬레이션 환경

임베디드 소프트웨어 개발을 위해 사용한 모델 기반 개발(model based development, MBD) 도구인 시뮬링크에서 시뮬레이션 모드는 크게 세 가지로 구분할 수 있다. Normal, SIL(software in the loop), 그리고 PIL 모드이다. Normal 모드는 모델을 개발하는 과정에서 지속적인 시뮬레이션 수행 및 결과 확인을 통한 모델의 수정, 보완을 수행하는 일반적인 개발과정에서 사용된다. SIL 모드에서는 모델로부터 코드를 생성하지만, 개발용 PC에서 컴파일 하여 시뮬레이션을 수행한다. PIL 모드에서는 코드 생성, 컴파일, 링크 과정을 통해 object 파일을 만들고 타겟에 탑재하여 실행하기 때문에 MCU(micro controller unit)에서의 실시간 작동 성능 분석이 가능하다[9].

PIL 모드 시뮬레이션을 수행하기 위해 AVCS 제어기와 동일한 MCU인 Texas Instrument 사의 TMS320F28335를 장착한 평가 보드인 eZdsp F28335를 활용하였다. 타겟 하드웨어를 사용할 수 있도록 Code Composer Studio V6, C2000 컴파일러, MATLAB Support Package 등을 설치하

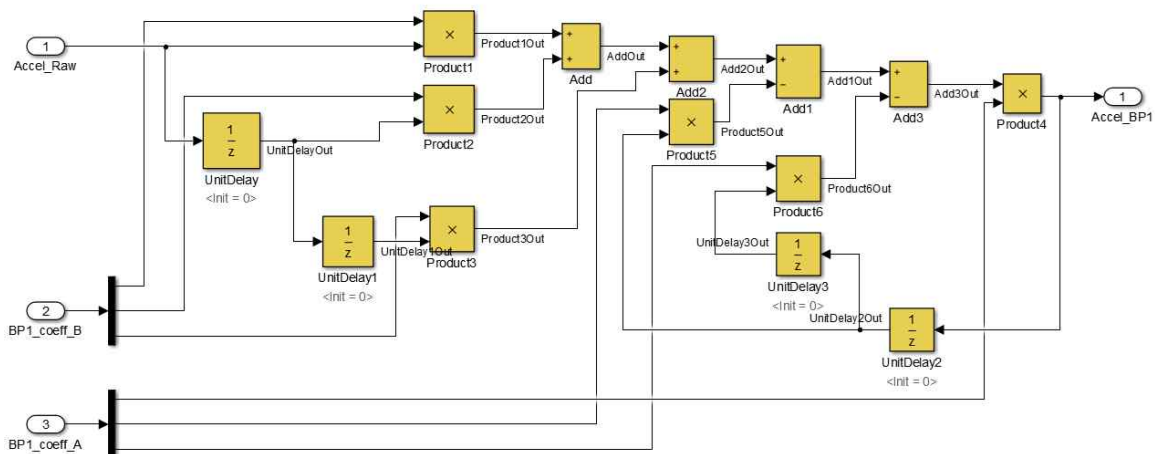


Fig. 6. Second order IIR filter implementation



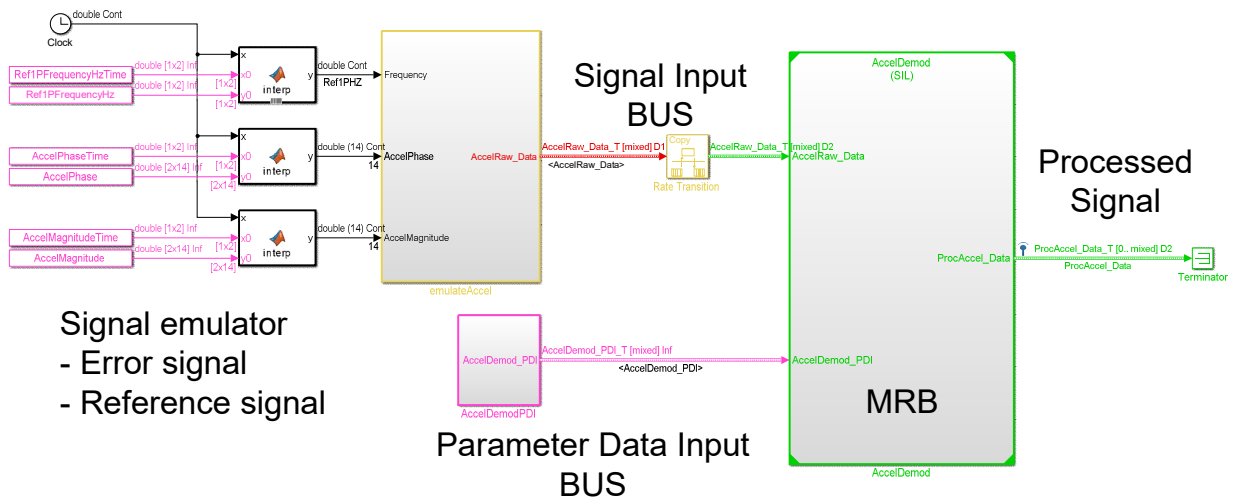


Fig. 7. Test Harness model

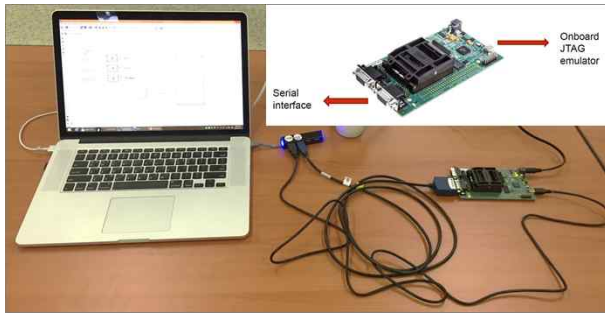


Fig. 8. PIL mode simulation

고, Simulink 환경설정을 구성하였다. 시뮬레이션 대상인 가속도 신호처리 모델을 Model Reference Block (MRB)으로 구현하고 가상의 여러 신호, 기준 신호 및 파라미터를 생성해 주는 블록을 만들어서 신호처리 MRB가 PIL 모드에서 구동되도록 시뮬레이션을 수행하였다. 테스트 하네스 구성은 Fig. 7과 같다.

개발용 PC와 eZdsp의 연결은 Fig. 8과 같다. PC의 USB 포트에 연결된 eZdsp의 JTAG 포트를 통해 컴파일된 object 파일을 로딩하여 실시간으로 실행하였다. 매 시간 스텝마다 모델의 입력과 출력 데이터를 기록하기 위하여 시리얼 통신을 사용하였다.

#### 4.2 시뮬레이션 결과

타코미터로부터 1/rev의 주파수와 방위각 정보를 추출하여 기준 신호로 사용하고, 14개의 가속도계가 0.1g부터 1.0g의 크기를 갖도록 입력 신호를 만들어 주었다. 가속도 신호처리 함수를 통해 획득한  $E_C$ ,  $E_S$  값으로 가속도 신호의 크기를 추정해 보면 Fig. 9와 같이 약 0.25초 만에

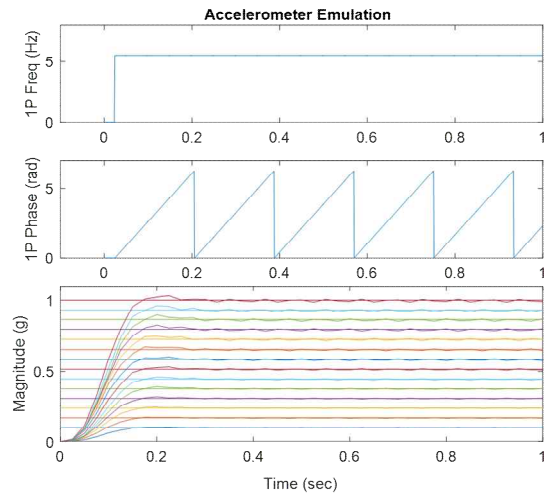


Fig. 9. Signal processing performance in time domain

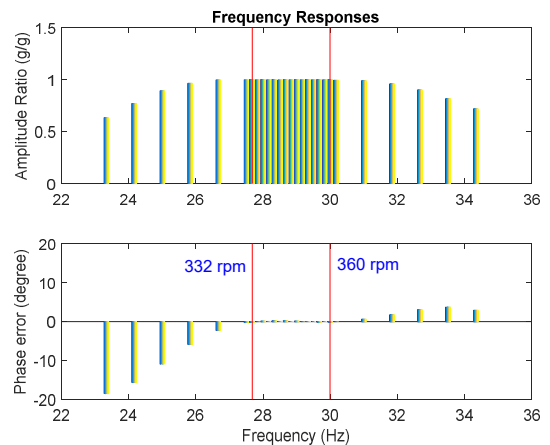


Fig. 10. Frequency response of the signal processing function

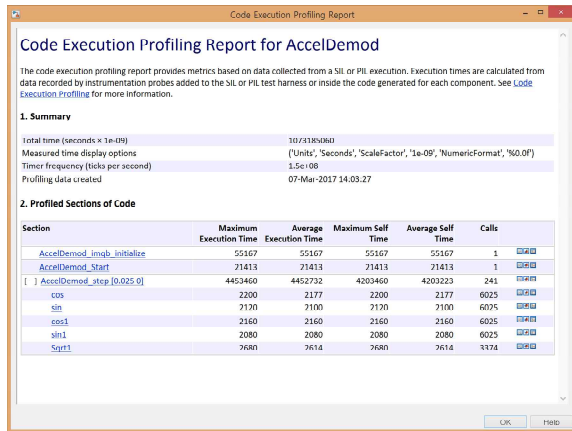


Fig. 11. Code execution profiling report

가속도 신호 정보를 추정하고 있음을 살펴볼 수 있다.

잡음 제거를 위해 적용한 대역 통과 필터의 성능을 확인하기 위하여 기준 신호의 주파수를 4.67 Hz ~ 6.87 Hz로 변화를 주면서 시뮬레이션을 수행하였다. 이는 로터 회전수 280 rpm ~ 412 rpm, N/rev 주파수 23.33 ~ 34.33 Hz에 해당된다. 신호처리된 가속도 신호와 가공하지 않은 가속도 신호의 관계를 살펴보면 Fig. 10과 같이 N/rev 주파수 주변에서는 크기 비가 1이고, 위상변화는 1도 이내가 되는 우수한 성능을 보여주고 있다.

대역통과 필터는 위상변화를 수반하게 된다. 본 연구에서는 보다 넓은 영역에서 위상변화를 최소화하기 위해 Fig. 4의 Bode plot의 중심 주파수 주변 구간의 위상변화를 선형 방정식으로 추정하여 보상하는 방법을 적용하였다. 그 결과 Fig. 10 하단과 같이 일반적인 필터와는 달리 넓은 영역에서 위상 변화가 없는 결과를 얻을 수 있었다. Fig. 10에 표시된 두 개의 실선은 332 rpm, 360 rpm에 상응하는 N/rev 진동수로서, LCH 주로터 회전수인 342 rpm ~ 350 rpm을 기준으로  $\pm 10$  rpm 범위에 해당하는 경계값이다. 해당 범위를 벗어난 영역의 신호가 입력될 경우, 신호처리 되어 출력되는 신호의 크기는 작아지고 위상변화가 발생하므로 진동제어 성능을 보장할 수 없게 되고, 오히려 더 큰 진동을 유발할 수도 있다. 이러한 상태를 방지하기 위한 연속 고장진단 기능 또한 설계되어 소프트웨어에 반영되어 있으나, 본 논문의 주제를 벗어난 영역으로 상세히 기술하지 않겠다.

임베디드 시스템에서 구동되는 실시간 소프트웨어는 요구되는 실행시간 제한 값 이내에 완료되어야 한다. 40 Hz로 작동되는 LCH AVCS 제

어알고리즘 소프트웨어는 호출 주기인 25 msec의 50%인 12.5 msec 가 실행시간 요구조건이다. 전체 알고리즘 소프트웨어 중 본 연구의 신호처리 알고리즘이 가장 많은 시간을 소비하고 있다. PIL 모드 시뮬레이션은 타겟 MCU 상에서 소프트웨어가 작동하므로 양산용 소프트웨어가 실행되는 것과 동일한 환경에서의 실행시간 분석이 가능하다. 시뮬링의 Execution Profiling 기능을 활용하여 PIL 모드 시뮬레이션을 수행하면 Fig. 11과 같은 Code Execution Profiling Report를 생성할 수 있다.

프로그램을 Flash 메모리에 탑재해서 동작할 경우 설계된 신호처리 알고리즘 함수의 최대 실행시간은 4.45346 msec로 실행시간 제한 값의 35.6%를 차지하였다. 실행시간 요구사항 충족여부에 대한 판단은 LCH AVCS 제어 알고리즘 소프트웨어 전체에 대한 실행시간 분석을 통해 입증할 예정이다.

## V. 결 론

헬리콥터 동체 진동의 주성분인 N/rev 진동수의 변화가 크지 않은 특성을 고려하여 상대적으로 낮은 주파수로 작동 가능한 진동제어 알고리즘과 제어 입력 계산에 필요한 가속도 신호처리 알고리즘을 도출하였다. 수립된 가속도 신호처리 알고리즘은 시뮬링을 이용하여 모델링하였고, 40 Hz로 동작하면서 1 kHz의 데이터를 처리하도록 설계하였다. 설계된 알고리즘 모델은 PIL 모드 시뮬레이션을 통해 타겟 MCU에 탑재 및 실행되는 상태에서 실시간 동작 성능을 분석하였다. 가속도 추출 기능, 대역통과 필터 기능이 원활하게 동작함을 확인하고, 실행시간은 제한 값의 35.6% 이내에서 동작 가능함을 확인하였다.

## 후 기

본 논문은 산업통상자원부의 지원으로 수행중인 소형무장헬기 연계 민수헬기 핵심기술개발사업(과제번호: 10053157) 연구결과 중 일부임.

## References

- 1) Loewy, Robert G., "Helicopter Vibrations: A Technological Perspective," *Journal of the American Helicopter Society*, Vol. 29, No. 4, 1984, pp. 4-30.



- 2) Yun, C. Y., Kim, D.-H., and Kang, H. J., "Vibration Prediction of Helicopter Airframe," *KSNVE Annual Spring Conference*, Yeosu, April 2013, pp. 25-26.
- 3) Jung, S.-U., Kwak, D.-I., Kim, S.-H., Choi, J.-H., and Shim, D.-S., "Vibration Reduction Devices for Korean Utility Helicopter," *Journal of The Korean Society for Aeronautical and Space Sciences*, Vol. 41, No. 12, 2013, pp. 987-993.
- 4) Swanson, D., Black, P., Girondin, V., Bachmeyer, P., and Jolly, M., "Active Vibration Control Using Circular Force Generators," 41st European Rotorcraft Forum, Munich, Germany, September 2015, pp. 1-4.
- 5) Kim, D.-H., Kim, T.-J., Paek, S.-K., Kwak, D.-I., and Jung, S.-U., "Application and Performance Evaluation of Helicopter Active Vibration Control System for Surion," *Journal of The Korean Society for Aeronautical and Space Sciences*, Vol. 43, No. 6, 2015, pp. 156-165.
- 6) Kim, D.-H., Kim, T.-J., Jung, S.-U., and Kwak, D.-I., "Test and Simulation of an Active Vibration Control System for Helicopter Application," *International Journal of Aeronautical and Space Sciences*, Vol. 17, No. 3, 2016, pp. 442-453.
- 7) S.J. Elliott and P.A. Nelson, "Active noise control," *IEEE signal processing magazine*, October 1993, pp. 12~35.
- 8) Lim, H.-C., Jung, S. U., Kwak, D.-I., and Kim, D.-H., "Design of Performance Parameter in accordance with Helicopter vibration change," *Proceeding of The Korean Society for Aeronautical and Space Sciences Fall Conference*, Jeju, November 2016, pp. 16-18
- 9) *Simulink User's Guide*, The Mathworks Inc., Natick, MA, 2015.