

## 칼만필터링을 사용한 목표물 추적시스템의 설계

김        종        화\*

### Design of Target Tracking System Using Kalman Filtering

*Kim, Jong-Hwa*

#### Abstract

A new filter algorithm is suggested improving structurally the conventional extended Kalman filter of which the performance is dependent on the selection of the reference axes, by use of line-of-sight axes and gain rotation technique. The implementation method using microcomputer which implements tracking Kalman filter is introduced in terms of hardware and software. Then, through the simulation the performance of suggested filter is compared with that of conventional extended Kalman filter and the possibility of the real time tracking of moving target is investigated.

#### 서        론

해상, 해저, 우주 등에서 기동중인 목표물의 탐색 및 추적에 응용되는 기법중 칼만필터가 가장 널리 사용되고 있다. 적절한 시기에 목표물 추적을 위한 수요가 증가되었음은 물론이고 컴퓨터와 같은 우수한 장비들이 함께 출현되었을 뿐만 아니라 특히 칼만필터는 순환적 선형구조를 가지고 있어 알고리즘이 단순하고 수렴성이 좋으며, 이전의 스펙트럼분석 방식에서 벗어나 시간영역에서 직접적인 처리가 가능한 방법이기 때문이다. 그러나 실제 시스템에 적용할 경우 기동목표물이 가지는 비선형성을 처리하여 선형 시스템에서 보장되는 최적성을 부여하는 문제, 이로 인하여 발생되는 많은 양의 데이터를 온라인으로 처리해야 하는 컴퓨터 능력의 부담으로 인한 비경제성, 디지털 컴퓨터로 신호를 처리할 때 생기는 절삭 오차의 누적에 의한 필터의 발산문제, 목표물의 종류나 형상에 따른 잡음 상호분산 크기의 적절한 선택 등 아직 해결해야 할 많은 문제점들이 남아 있다.<sup>1~6)</sup>

\* 한국해양대학 제어계측공학과 전임강사

본 연구에서는 추적시스템으로 확장 칼만필터를 사용한다는 측면에서, 기준 좌표계의 선택에 의존하는 확장 칼만필터를 새로운 좌표계의 도입과 좌표변환을 이용하여 실시간 추정이 가능하도록 구조적으로 개선할 것이며 현재 광범위하게 사용되고 있는 마이크로 컴퓨터를 이용하여 경제적이며 손쉽게 필터를 실현시키는 방법을 하드웨어 및 소프트웨어의 측면에서 제시할 것이다. 또한 예제를 사용하여 모의실험을 행하므로써 개선된 확장 칼만필터의 추적성능을 기존의 확장 칼만필터와 비교하고, 마이크로 컴퓨터를 이용한 추적시스템의 실시간 추적의 가능성을 제시할 것이다.

## 2. 목표물의 모델링과 확장 칼만필터

기동중인 목표물은 고속으로 비행하거나 책략을 수반함으로써 일반적으로 비선형 동적 시스템으로 표현된다. 이 때 좌표계는 직교좌표계와 극좌표계가 주로 사용되는데, 직교좌표계를 사용할 경우 상태변수는 각 방향의 거리, 속도, 가속도 성분과 확률가속도 모델로 구성되며, 극좌표를 사용할 경우는 사정 거리, 방위각, 고각과 그 미분요소들 및 확률가속도 모델에 근거를 두고 구성된다. 이와같은 좌표계의 선택은 추적 알고리즘의 선택과 측정 시스템의 선택에 크게 좌우된다고 할 수 있으며, 본 연구에서는 비선형 시스템의 추적 알고리즘으로 선형 칼만필터의 우수성을 이용한 확장 칼만필터를 선택한다고 전제하겠다.

만약 극좌표계를 선택한다면 목표물의 상태방정식은 비선형으로 표현되고, 확장 칼만필터로서 필터링을 수행할 경우 선형화가 필요하게 된다. 이럴 경우 선형화로 인한 동적오차는 직교 좌표계를 선택할 경우보다 훨씬 크게 되어 필터 발산의 원인이 되기도 하고 계산량 또한 직교 좌표계를 선택할 경우보다 상대적으로 증가하여 컴퓨터의 능력 부담을 초래한다. 그러나 레이다등 측정 시스템의 측정방정식이 대부분 극좌표계로 표현되므로 필터 알고리즘의 수행시 요구되는 좌표변환 과정이 필요없게 되는 장점이 있다<sup>11)</sup>.

직교 좌표계를 선택할 경우에는 목표물의 동특성을 정확하게 해석할 수 있으므로 비선형으로 표현된 목표물의 운동 방정식을 선형화시킬 때 발생하는 동적오차를 상당히 줄일 수 있으며 계산량도 훨씬 줄어 든다. 그러나 측정 방정식의 좌표와 일치하지 않기 때문에 필터링 과정에서의 좌표변환이 수반 되어야만 한다.

일반적으로 목표물의 상태방정식은 동특성을 정확히 해석한다는 측면에서 직교좌표계를 사용하여 비선형 연속방정식으로 표현된다.

$$\dot{x}(t) = f[x(t), u(t), w(t)] \quad (1)$$

여기서  $x(t)$ 는 상태벡타이고  $u(t)$ 는 제어입력이며  $w(t)$ 는 백색 정규시스템 잡음이다.

제어입력  $u(t)$ 가 최적필터의 설계에 아무런 영향을 미치지 않을때 측정시스템의 측정방정식은

$$m(k) = h[x(k), v(k)] \quad (2)$$

와 같이 이산방정식으로 표현되며,  $h[x(k), v(k)]$ 는 직교좌표계와 극좌표계 사이의 위치성분을

변환시켜 주는 변환벡터를 포함하고 있어야 하며 측정치  $m(k)$ 는  $x(k)$ 의 함수이어야 한다.  $v(k)$ 는 측정잡음이고 잡음들 사이의 관계는

$$\begin{aligned} E[w(t)] &= 0, \quad E[v(k)] = 0, \quad E[w(t)v^T(k)] = 0, \\ E[w(t)w^T(\tau)] &= Q\delta(t-\tau), \quad E[v(k)v^T(j)] = R\delta(k-j) \end{aligned} \quad (3)$$

와 같다고 가정한다. 그러나 대부분의 필터 설계자는 이동목표물의 상태방정식에 대하여 식(1)과 같이 주어지는 비선형 방정식을 원형 그대로 이용하지 않는다. 왜냐하면 실시간 추적의 측면에서 선형화로 주어지는 값들을 계산할 때 온라인 계산시간을 감안하지 않을 수 없을 뿐만아니라 직교좌표계를 사용하여 표현된 상태방정식은 선형시스템으로 표현되더라도 동적오차가 크게 포함되지 않기 때문이다. 그러므로 어느 정도의 모델링 오차를 감수하더라도 상태천이행렬을 구하여 될 수 있으면 선형시스템으로 표현하려고 한다. 따라서 본 연구에서는 목표물에 대한 운동방정식을 선형 방정식으로 가정하기로 한다.<sup>3,12~13)</sup>

이동 목표물에 대한 이산 상태방정식은

$$x(k+1) = \Phi(\Delta t, \alpha)x(k) + w(k) \quad (4)$$

이고, 이때 측정방정식은 목표물의 상태변수의 함수이어야 하며

$$m(k) = h[x(k)] + v(k) \quad (5)$$

로서 극좌표계로 표시된 비선형 방정식으로 표현된다. 여기서  $k$ 는 표준화된 이산시간이고  $\Delta t$ 는 측정시스템의 샘플링 시간,  $x(k)$ 는 직교좌표계에 대한 목표물의 상태변수로서  $x(k) = [x, y, z, \dot{x}, \dot{y}, \dot{z}, \ddot{x}, \ddot{y}, \ddot{z}]$ 이며,  $\Phi$ 는 상태천이행렬로서 샘플링 시간  $\Delta t$ 와 확률가속도 성분의 1차 마르코프 파라미터  $\alpha = [\alpha_x, \alpha_y, \alpha_z]$ 의 함수로서 다음과 같이 결정된다.<sup>9)</sup>

$$\Phi(\Delta t, \alpha) = \begin{pmatrix} I & \Delta t I & (\Delta t)^2 I / 2 \\ \underline{0} & \underline{0} & \Delta t / I \\ \underline{0} & \underline{0} & \alpha I \end{pmatrix} \quad (6)$$

여기서  $I$ 는  $3 \times 3$ 단위행렬이고,  $\underline{0}$ 는  $3 \times 3$  0행렬이다. 그런데  $m(k)$ 는 실제의 측정 시스템이 목표물의 상태를 직접 측정한 측정치로서, 잡음이 포함되어 있으며 사정거리, 방위각, 고각으로 구성되어 있다.

$$m(k) = [r \ \theta \ \phi] \quad (7)$$

$h(k)$ 는 직교좌표계의 위치성분과 극좌표계의 위치성분 사이의 변환벡터로서 잡음이 무시될 경우 다음과 같이 정의된다.

$$r = \sqrt{x^2 + y^2 + z^2}$$

$$\begin{aligned}\theta &= \tan^{-1}(y/x) \\ \phi &= \tan^{-1}(z/\sqrt{x^2+y^2+z^2})\end{aligned}\quad (8)$$

위와 같이 선택된 목표물의 상태방정식과 측정 방정식에 대하여 상태변수  $x(k)$ 를 순환적으로 추정하기 위한 확장 칼만필터 알고리즘은 다음과 같다.

$$\hat{x}(k+1/k) = \Phi \hat{x}(k/k) \quad (9)$$

$$\hat{x}(k/k) = \hat{x}(k/k-1) + K(k)[m(k) - h(x(k/k-1))] \quad (10)$$

$$P(k/k-1) = \Phi P(k-1/k-1) \Phi^T + Q(k) \quad (11)$$

$$P(k/k) = P(k/k-1) - P(k/k-1)H^T[HP(k/k-1)H^T + R(k)]^{-1}HP(k/k-1) \quad (12)$$

$$K(k) = P(k/k)H^T R^{-1}(k) \quad (13)$$

여기서  $H = \partial h(x)/\partial x |_{x=x(k/k-1)}$ 로서 Jacobian행렬이다. 아래 그림 1은 순환적인 확장 칼만필터의 블록선도이며 온라인으로 계산되는 값은 칼만이득  $K(k)$ 와 선형화된 행렬  $H$ 이다.

그런데 측정치는 센서를 통하여 매 샘플링 시간마다 측정되어 필터에 공급되는 값이고 극좌표계로 표현되어 있으므로 식 (10)의 계산을 위해서는 직교좌표계로 표현된 상태변수의 예측된 추정치  $x(k/k-1)$ 를 극좌표계로 변환시켜 주는 변환행렬이 필요하다. 이 변환행렬을  $G(x)$ 라 하면<sup>15)</sup>

$$G(x) = \begin{pmatrix} x/r & y/r & z/r \\ -y/(x^2+y^2) & x/(x^2+y^2) & 0 \\ -xz/r^2\sqrt{x^2+y^2} & -yz/r^2\sqrt{x^2+y^2} & \sqrt{x^2+y^2}/r^2 \end{pmatrix} \quad (14)$$

이고,  $G(x)$ 는 앞에서 언급한 선형화된 측정행렬  $H$ 의 일부가 된다.

$$H = [G(x) \quad 0 \quad 0] \quad (15)$$

역으로 극좌표계를 직교좌표계로 바꾸는 변환행렬을  $F(x)$ 라고 하면

$$F(x) = \begin{pmatrix} x/r & -y & -xz/\sqrt{x^2+y^2} \\ y/r & x & -yz/\sqrt{x^2+y^2} \\ z/r & 0 & \sqrt{x^2+y^2} \end{pmatrix} \quad (16)$$

이고, 이 두 행렬의 관계는  $G(x) = F^{-1}(x)$ 가 된다.

앞에서 언급한 필터 알고리즘은 기준좌표계로서 고정된 직교좌표계를 사용하여 구성되어 있기 때문에 좌표들 사이의 연성된 요소가 많고, 이것들은 필터 알고리즘을 지배하는 오차 상호분산 행렬이나 칼만이득, 변환벡터 등을 계산하는데 소요되는 계산량을 증가시키기 때문에 전체적인 온라인 계산시간이 증가하게 되어 실시간에 문제를 유발시킨다. 따라서 상호연성된 요소들을 제거함으로써 필터를 단순화시킬 수 있는 새로운 좌표계를 도입하여 생각하고자 한다.<sup>20)</sup>

칼만필터링을 사용한 목표물 추적시스템의 설계

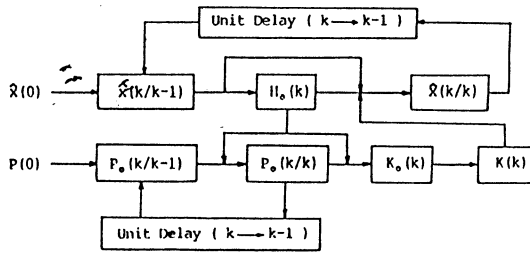


Fig. 1 Block diagram of recursive extended Kalman filter.

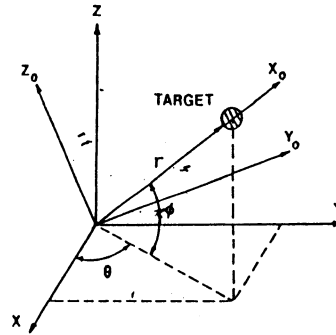


Fig. 2 Line-of-sight coordinate for target model.

여기서 도입하고자 하는 새로운 직교좌표계  $(x_0, y_0, z_0)$ 는 목표물을 따라 움직이는 좌표계로서,  $x_0$ 축은 측정 시스템의 위치와 목표물을 연결하는 관측선(line-of-sight)과 일치하도록 두고,  $y_0$ 축은  $x$ - $y$ 평면상에, 첨자 “0”은 관측선 좌표계를 의미한다고 정의한다. 이 때  $x_0$ 방향의 위치성분은 극좌표계의 사정거리와 같고  $y_0$ 와  $z_0$ 방향의 위치성분은 0이 되며, 극좌표계의 방위각과 고각도 0이 됨을 알 수 있다.

따라서 관측선 좌표계와 이것을 기준으로 사용한 극좌표계 사이의 변환행렬은 식(14), (16)에서 위치성분만이 사용되기 때문에 연성된 요소들이 제거되고 아래와 같이 간단한 형태가 됨을 알 수 있다.

$$G_0(x) = \text{diag}[1 \quad 1/r \quad 1/r] \quad (17)$$

$$F_0(x) = \text{diag}[1 \quad r \quad r] \quad (18)$$

이 때 필터 추정값의 상호분산은 각 하나의 축에 대한 위치, 속도, 가속도 성분들이 연성되어 있는 요소들만을 가지는 대칭행렬이 되며 칼만이득도 3개의 좌표간에는 서로 비연성되어 있고 하나의 좌표에 대하여 대각선 요소만을 가지는 형태로 되어 계산량이 훨씬 줄어들음을 알 수 있다.

관측선 좌표계를 사용하여 구한 칼만이득은 필터에 대한 기준 좌표계로 변환되어야 하는데 식 (13), (15)로부터

$$K(k) = P(k/k) \begin{pmatrix} G^T(x) \\ \underline{0} \\ \underline{0} \end{pmatrix} R^{-1}(k) \quad (19)$$

이 되고, 이득변환의 관계식을 이용하면<sup>20)</sup>

$$K(k) = \begin{pmatrix} F(x) & \underline{0} & \underline{0} \\ \underline{0} & F(x) & \underline{0} \\ \underline{0} & \underline{0} & F(x) \end{pmatrix} \begin{pmatrix} G_0(x) & \underline{0} & \underline{0} \\ \underline{0} & G_0(x) & \underline{0} \\ \underline{0} & \underline{0} & G_0(x) \end{pmatrix} K_0(k) \quad (20)$$

이 된다. 여기서  $T(x)$ 를 다음과 같이 정의하면

$$T(x) = F(x) G_0(x) = \begin{pmatrix} \mathbf{x}/r & -\mathbf{y}/r & -\mathbf{xz}/r \sqrt{\mathbf{x}^2 + \mathbf{y}^2} \\ \mathbf{y}/r & \mathbf{x}/r & -\mathbf{yz}/r \sqrt{\mathbf{x}^2 + \mathbf{y}^2} \\ \mathbf{z}/r & \mathbf{0} & \sqrt{\mathbf{x}^2 + \mathbf{y}^2}/r \end{pmatrix} \quad (21)$$

이고,  $K(k)$ 는 다음과 같이 정리된다.

$$K(k) = \begin{pmatrix} T(x) & \underline{0} & \underline{0} \\ \underline{0} & T(x) & \underline{0} \\ \underline{0} & \underline{0} & T(x) \end{pmatrix} K_0(k) \quad (22)$$

위와 같이 관측선 좌표계를 도입하여 칼만이득을 계산하고 변환벡터를 이용하여 다시 기준좌표계의 칼만이득으로 변환하는 제안된 확장 칼만필터 알고리즘은 다음과 같다.

$$\hat{\mathbf{x}}(k+1/k) = \Phi \hat{\mathbf{x}}(k/k) \quad (23)$$

$$\hat{\mathbf{x}}(k/k) = \hat{\mathbf{x}}(k/k-1) + K(k) [m(k) - H \hat{\mathbf{x}}(k/k-1)] \quad (24)$$

$$P_0(k/k-1) = \Phi P_0(k-1/k-1) \Phi^T + Q(k) \quad (25)$$

$$P_0(k/k) = P_0(k/k-1) - P_0(k/k-1) H_0^T [H_0 P_0(k/k-1) H_0^T + R(k)]^{-1} H_0 P(k/k-1) \quad (26)$$

$$K_0(k) = P_0(k/k) H_0^T R^{-1}(k) \quad (27)$$

$$K(k) = \text{diag}[T(x)] K_0(k) \quad (28)$$

여기서  $H_0$ 는 식(15)로부터 다음과 같이 쓸 수 있다.

$$H_0 = [G_0(x) \quad \underline{0} \quad \underline{0}] \quad (29)$$

이 때 개선된 확장 칼만필터의 순환적 블럭선도는 그림 3과 같다.

### 3. 마이크로 컴퓨터를 이용한 추적용 칼만필터의 개발

일반적으로 필터링을 응용한 대부분의 추적 시스템은 대형의 컴퓨터를 요구한다. 하지만 그런 경우 하드웨어를 구성하는 데 소요되는 경비가 많이 부담되어 상대적으로 효용가치가 없는 작은 규모의 추적시스템은 현실적으로 구성하기가 거의 불가능하다.

따라서 본 절에서는 광범위하게 보급되어 있고 가격이 저렴한 마이크로 컴퓨터를 이용하여 추적용 칼만필터를 구성하는 방법을 제시하겠으며, 마이크로 컴퓨터의 기능을 보완해 줌으로써 실시간 추적의 가능성을 목표물의 시나리오를 사용하여 검토하고자 한다.

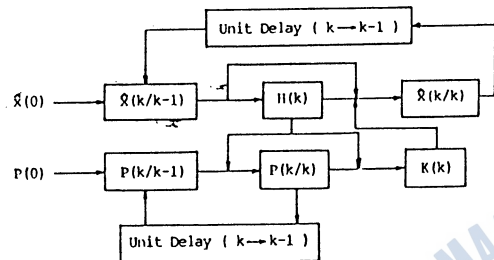


Fig. 3 Block diagram of suggested extended Kalman filter.

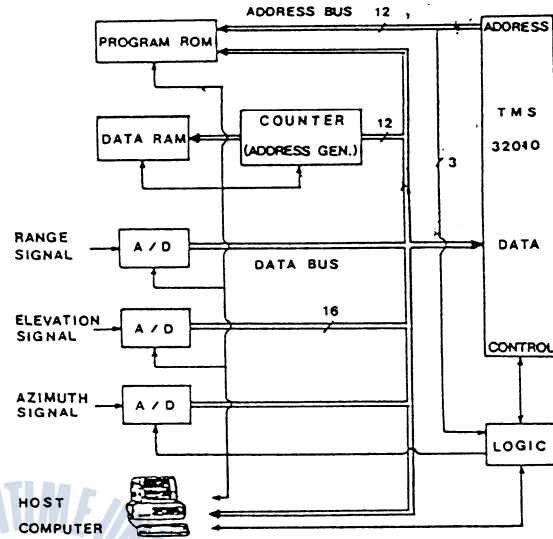


Fig. 4 Block diagram of tracking Kalman filter using TMS32010 DSP.

### 3. 1 하드웨어의 구성

범용 마이크로 컴퓨터를 사용하여 추적용 칼만필터를 구성하고자 할 때 목표물 상태의 실시간 추정이 거의 불가능하다는 것은 쉽게 짐작할 수 있다. 왜냐하면 빠른 속도로 기동하고 있는 목표물에 대한 운동방정식은 비선형이고 상대적으로 시스템의 상태변수가 증가하여 주어진 샘플링 시간내의 온라인 계산량이 많아 질 수밖에 없는데, 마이크로 컴퓨터는 데이터 프로세싱이나 연산과정에서 제약과 한계를 가지고 있기 때문이다. 따라서 먼저 마이크로 컴퓨터가 가지고 있는 제약점을 간단히 검토해 볼 필요가 있다. 일반적으로 마이크로 컴퓨터는 데이터를 내부적으로 처리하기 위하여 저장할 수 있는 주기억 용량이 적고 단어길이가 제한되어 있기 때문에 한꺼번에 많은 데이터를 처리할 수 없다. 또한 클럭이 느리고 연산과정에서 명령어의 수행과정이 직렬형태이며 시프터의 기능이 단위명령 사이클당 배럴시프터가 아니기 때문에 연산속도가 느리다. 그러므로 추적용 칼만필터를 위해서는 이러한 마이크로 컴퓨터의 기능을 개선시켜 주거나 보완하여 많은 데이터를 고속으로 처리할 수 있는 부수적인 장치가 필요하게 된다.

본 연구에서는 이러한 목적을 달성하기 위하여 TMS32010 DSP(Digital Signal Processor)를 사용하였다. TMS32010 DSP는 16×16 비트 곱셈연산을 200 nsec에 처리하며 Harvard구조를 변형하여 사용한 구조이기 때문에 데이터 처리부분과 프로그램 부분이 분리된 파이프라인 방식으로 동작하게 되어 있어 신호 처리능력이 매우 우수하다.<sup>17)</sup>

TMS32010 DSP를 이용하여 구성한 실시간 추적용 칼만필터의 전체적인 구성도는 그림 4에 예시하였으며, 주 컴퓨터를 중심으로 TMS32010 DSP, 프로그램 ROM, 데이터 RAM, A/D 변환기, 논리회로 등으로 구성되어 있다.<sup>19)</sup>



1) 주 컴퓨터

목표물에 대한 정보와 필터에 대한 정보를 시스템 조작자와 주고 받는 창구 역할을 하며 직접 신호를 처리한다거나 필터 알고리즘을 수행하지는 않는다. 전용 신호처리 프로세서로서 TMS32010 DSP를 사용하기 때문에 이 경우 주 컴퓨터로는 IBM PC 유사기종이나 VAX 기종의 마이크로 컴퓨터가 적당하다.

2) TMS32010 DSP

센서로부터 들어 오는 정보를 A/D 변환기를 통하여 받아서 처리하며 필터 알고리즘을 실제로 수행하는 전용 디지털 프로세서이다. 많은 양의 데이터를 고속으로 처리할 수 있으며, 프로그램 메모리와 데이터 메모리가 분리되어 있고, 명령의 인출과 수행을 동시에 할 수 있어 주 컴퓨터의 제약점을 보완해 주기 때문에 추적 알고리즘을 마이크로 컴퓨터로 구성할 경우 매우 적합한 DSP이다.

그림 5는 Harvard구조를 가진 TMS32010 DSP가 명령을 인출하고 수행하는 타이밍선도를 보인 것이다.

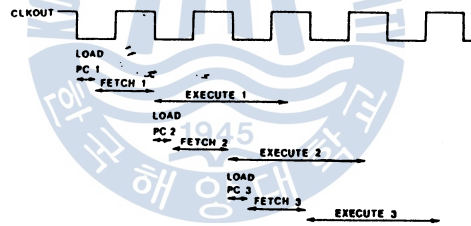


Fig. 5 Timing diagram of TMS32010 DSP.

3) 프로그램 ROM

TMS32010 DSP는 내장된 프로그램 ROM을 보유하고 있지 않은 프로세서이기 때문에 외장된 프로그램 ROM이 필요하다. 어드레스가 12비트이므로 4Kword의 프로그램 ROM이 필요하며 호출시간이 100nsec 이하인 것이 필요하다. 이를 만족시키는 ROM은 쉽게 구할 수 없기 때문에 PROM에 프로그램을 저장시킨 후 프로그램의 수행전에 호출시간 100nsec 이하인 RAM으로 프로그램을 다운로드 시킨후 RAM으로서 프로그램을 수행시킨다.

4) 데이터 RAM

많은 양의 데이터를 처리할 경우에 필요하며, 데이터의 양이 적을 경우에는 TMS32010의 내부에 내장되어 있는 144×16비트의 데이터 메모리를 사용한다. 외부에 데이터 RAM을 확장해야 할 경우 TMS32010의 어드레스는 프로그램 메모리의 어드레스만을 공급하므로 데이터 RAM의 어드레스를 위해서는 RAM 어드레스 발생기가 필요하며 4비트 카운터 4개를 직렬로 사용하면 16비트 까지 확장가능하다. 이 때 사용하는 RAM의 어드레스 시간은 70nsec 이하이어야 한다. 그림 6은 외장된 데이터 RAM을 구성하기 위하여 TMS32010 과의 접합 방법을 예시한 것이다.





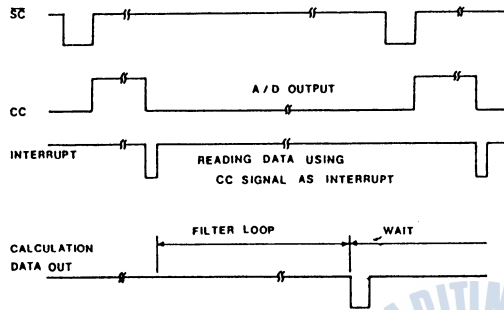


Fig. 7 Timing diagram executing extended Kalman filter.

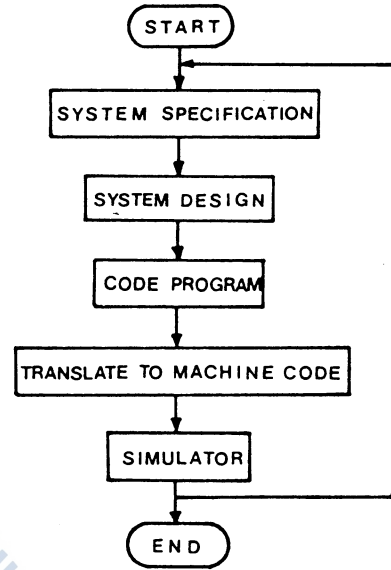


Fig. 8 Flowchart of software development.

### 1) 시스템 내역

실제 시스템에서는 목표물의 사정거리, 고각, 방위각에 대한 측정치는 센서를 통하여 수집되어 데이터 프로세서를 거쳐서 TMS32010의 입력단에 직접 인가된다. 그러나 추적 필터의 구현을 위해서는 알고리즘 자체에 필요한 목표물의 수학적 모델링이 수반되어야 한다.

목표물에 대한 정보는 거의 알려져 있지 않기 때문에 과거의 경험과 공학적 통찰력을 사용하여 시스템 및 측정잡음에 대한 특성 등을 결정하고, 설계자가 의도하는 추적 성능지수를 결정하며 사용되는 측정 시스템에 의하여 샘플링 시간을 결정하여야 한다.

### 2) 시스템의 설계

추적 시스템을 설계하고자 할 때 가장 먼저 선행되어야 하는 것은 사용될 컴퓨터의 용량과 그 구조를 파악하여 설계자가 의도하는 추적 성능 지수를 얻을 수 있는지 검토해야 한다. 일반적으로 필터 설계자는 완벽한 최적의 필터를 구성하려고 노력하지 않는다. 특히 비선형 시스템인 경우에는 계산량이 많으므로 온라인 계산시간을 고려하지 않을 수 없다. 이러한 목적을 달성하기 위하여 시스템의 동작에 기여도가 낮은 요소들을 통찰하여 제거함으로써 필터를 단순화시키고 최적 시스템과 거의 같은 성능을 가지는 준최적 필터를 설계하려고 노력하며, 선형화로 인한 온라인 계산부담을 줄이기 위하여 좌표변환과 역변환 기법 등의 수학적 처리를 통하여 계산시간을 절약하려고 애쓴다. 마지막으로 필터를 실현시킬 때 컴퓨터의 연산방법을 결정해 주므로써 실시간 추적의 가능성을 추구한다.

여기서는 앞에서 제시한 바와 같이 관측선 좌표계와 이득 변환기법을 사용한 확장 칼만필터, 즉 식 (23)~(28)를 하드웨어를 구성하기 위한 필터로 사용하였으며, 이 행렬 알고리즘을 먼저 수 계산한 스칼라 알고리즘의 산술연산을 컴퓨터의 연산방법으로서 선택하였다<sup>20)</sup>.

칼만필터링을 사용한 목표물 추적시스템의 설계

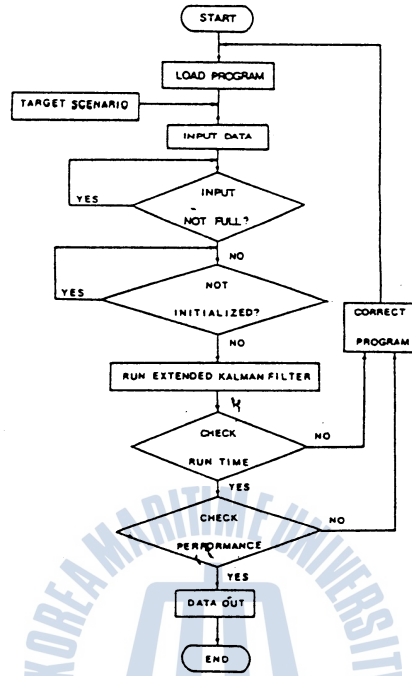


Fig. 9 Flowchart of simulator for the tracking filter.

3) 프로그램의 코우팅 순서

TMS32010 어셈블리어로 필터 알고리즘을 코우딩하기 위해서는 다음과 같은 3단계의 전개과정을 거친다면 손쉽게 해결할 수 있다. 첫째, 설계된 스칼라 필터 알고리즘에 대한 구조적인 흐름도를 그린다. 둘째 흐름도에 따라서 다시 세분화 작업을 행하여 필터 알고리즘을 고급언어로 프로그래밍한다. 어셈블리어로 코우딩하기 전에 고급언어의 프로그래밍을 거치는 이유는 필터 알고리즘의 정확성을 조사하고 쉽게 TMS32010 어셈블리어로 번역할 수 있도록 구조적인 면을 부여하기 위해서이다. 셋째, 최종적인 필터의 실행 프로그램은 고급언어로 된 프로그램을 TMS32010의 매크로 어셈블리어로 코우딩하여 얻고, 사용되는 시스템 상태변수나 샘플링 시간, 잡음들의 상호분산 등의 적당한 스케일은 단어길이가 긴 대용량의 컴퓨터를 사용하여 실행한 고급언어의 프로그램 결과를 이용하여 그 값에 최대한 접근되도록 결정함으로써 얻도록 한다.

4) 모의실험을 위한 필터의 모의실험기

설계된 필터 알고리즘의 성능을 시험하기 위해서는 모의실험기를 구성하여야만 하고, 구성된 모의실험기를 이용하여 반복적으로 필터를 조정함으로써 그 목적을 달성할 수 있다. 아래 그림은 추적용 필터의 조정을 위한 모의실험기의 흐름도를 도시한 것이다.

따라서 추적성과 실시간 추적을 위한 온라인 계산시간을 점검하고 요구되는 값들이 얻어질 때까지 추적 알고리즘을 반복 개선할 수 있게 된다.

#### 4. 모의실험

TMS32010 DSP를 사용하여 구성된 추적용 필터의 실시간 추적 가능성과 성능을 평가해 보기 위하여 다음과 같이 모의실험을 수행하였다.

시나리오로서 고도 150m, 반경 1500m에서 200m/sec로 원운동하는 목표물에 대한 정보를 사용하였으며, 이를 묘사하는 표적의 운동방정식으로 식(4)를 사용하였고 마르코프 파라미터는  $\alpha_x = \alpha_y = \alpha_z = 1.0$ 로 선택하였다. 측정 시스템은 목표물에 대한 기준 좌표계의 원점에 두었으며 샘플링 주파수는 40Hz이고, 목표물의 측정 가능성은 100%로 하였으며, 이를 묘사하는 방정식으로는 식(5)를 사용하였다. 위에서 주어진 목표물에 대한 시나리오는 전략적 책략을 하는 경우에 해당하며 실제의 경우와 유사한 기동 특성을 가지고 있는 경우이다.

모의실험은 본 연구에서 제안한 모의실험기를 사용하여 수행되었으며, 추적성능을 비교하기 위하여 고급언어로 된 프로그램을 CYBER 180-830컴퓨터를 사용하여 결과를 얻었고 이것을 이용하여 추적성능을 개선하였다.

그림 10, 그림 11, 그림 12는 기준좌표계를 사용한 확장 칼만필터와 본 연구에서 제안한 필터의 추적성능을 x, y, z축에 대한 위치 추정오차로써 비교하여 도시한 것이다.

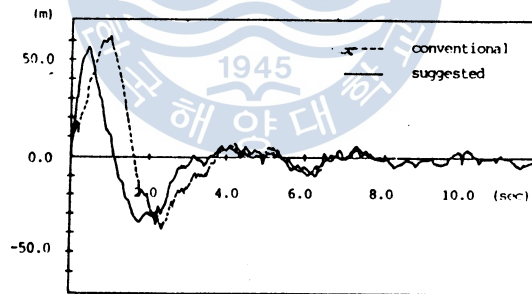


Fig. 10 Tracking errors of filters in the x-position.

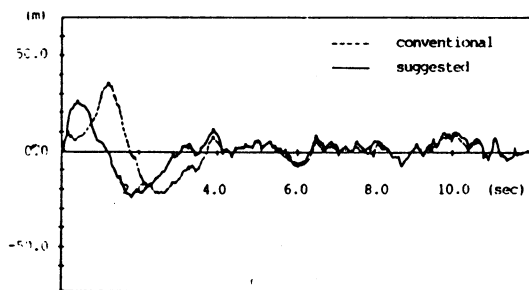


Fig. 11 Tracking errors of filters in the y-position.

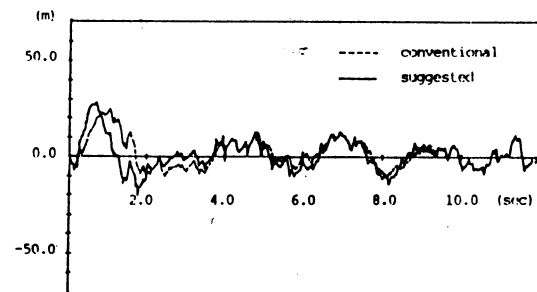


Fig. 12 Tracking errors of filters in the z-position.

위의 그림에서 보는 바와 같이 초기값으로부터 정상상태에 이르는 동안 필터는 과도응답을 나타낸다. 왜냐하면 이 과정에서는 칼만이득이과 이를 계산하기 위한 오차상호분산의 값이 안정되어 있지 못하기 때문이다. 또한 기존의 확장 칼만 필터보다 제안된 확장 칼만필터가 과도응답시간이 짧음을 알 수 있는데, 기존 확장 칼만필터의 경우 오차상호분산 행렬이 연성된 요소들을 많이 포함하고 있다는 것은 설명하였다. 이 요소들은 정상상태에 있어서는 거의 0으로 수렴되어 무시되는 값이지만 과도상태에 있어서는 추적오차가 크기 때문에 무시되지 못하며 이로 인한 오차의 누적이 크게 되어 오차상호분산의 정밀도가 떨어지고 칼만이득에 의한 필터의 보상량이 작아져서 추적오차가 일시적으로 발산현상을 보인다. 그러나 제안된 확장 칼만필터에서는 그러한 연성항이 없으므로 오차가 상대적으로 작게 되어 수렴성이 좋아진다. 그러므로, 연산시간의 측면에서 뿐만 아니라 이러한 측면에서도 오차상호분산의 연성요소들은 제거되어야 함을 알 수 있다.

표 1은 기존 좌표계를 사용한 확장 칼만필터와 제시된 확장 칼만필터를 비교하기 위하여 필터 알고리즘을 수행하는데 필요한 곱셈의 수와 TMS32010명령어의 수를 사용하여 표현한 도표이며, 궁극적으로 필터 한 루우프를 연산하는데 소요되는 연산시간을 제시한 것이다.

**Table 1. Numerical comparison between conventional extended Kalman filter and suggested filter.**

	칼만계인 계산을 중심으로 표현	곱셈의 수	TMS32010 명령어 총수	계산 소요시간
기존 확장 칼만필터	$P(k+1/k) = \Phi P(k/k) \Phi^T + Q$	344	7236	약 3.0msec
	$K(k) = P(k/k-1) H^T [HP(k/k-1) H^T + R]^{-1}$	207		
	$P(k/k) = [I - KH] P(k/k-1)$	324		
	기 타	250		
	합 계	1105		
제시된 필터	$P_0(k+1/k) = \Phi P_0(k/k) \Phi^T + Q$	84	3074	약 1.3msce
	$P_0(k/k) = P_0 - P_0 H^T [H_0 P_0 H_0^T]$	42		
	$K_0(k) = P_0(k/k) H_0^T R(k)^{-1}$	21		
	$K(k) = \text{diag}[T(x)] K_0(k)$	81		
	기 타	250		
	합 계	478		

## 5. 결 론

실시간 추적시스템을 설계할 경우 필터의 구성에 절대적으로 요구되는 것은 컴퓨터의 계산시간 및 기억용량이므로 컴퓨터의 계산부담을 덜어 주기 위해서는 필터에 영향을 미치는 정보의 주된 요소를 통찰하여 시스템의 동작에 기여도가 낮은 요소들은 제거함으로써 최적필터의 수행성능을 감소시키지 않고 최소한의 변수를 사용한 준최적 필터를 구성하는 것이다.

비선형 시스템에 확장 칼만필터를 적용시킬 경우 수반되는 선형화는 관측선 좌표계의 도입과

이득변환기법을 사용함으로써 극복할 수 있었고, 3개의 비연성된 소필터로 구분되어 계산시간과 기억용량의 감소로 컴퓨터의 계산부담이 훨씬 줄어 진다는 것을 알 수 있었다.

일반적으로 실제의 목표물에 대하여 추적시스템을 구성할 경우 대형의 컴퓨터가 요구된다. 그러므로 마이크로 컴퓨터로는 목표물의 정보를 실시간 추정한다는 것이 거의 불가능하다. 본 연구에서는 마이크로 컴퓨터를 사용하여 필터를 실현시키는 방법을 하드웨어 및 소프트웨어의 측면에서 제시하였다. 마이크로 컴퓨터의 제약점을 보완해 주기 위해서 TMS32010 DSP를 사용하였으며, 필터의 조정을 위한 모의실험기를 구성하였고, 연산방법으로 산술연산을 택함으로써 궁극적인 알고리즘은 스칼라 필터 알고리즘을 사용하였다.

제안된 필터 알고리즘과 제시된 추적 시스템의 성능을 시험하기 위해서, 원운동으로 책략비행을 하는 목표물의 시나리오에 대하여 모의실험을 행하였다. 필터의 수행성능을 시험하는 방법으로서 CYBER180-830을 사용하여 얻은 기존의 확장 칼만필터의 결과와 비교하는 방법을 택하였으며, 그 결과 제안된 필터의 수행성능은 기존의 확장 칼만필터와 거의 일치함을 알 수 있었다. 목표물에 대한 수학적 모델과 구성된 필터의 차수는 실제 모델과 거의 가깝게 일치하는 경우이며, 제시된 추적 시스템을 사용하여 제안된 필터를 한 번 연산하는데 소요되는 시간은 약 1.3msec가 소요되었고, 이는 목표물의 실시간 추적이 가능함을 시사해 주었다.

### 참고문헌

- 1) "United States Air Force Computer, Technology Forecast and Weapon System Impact Study," (COM-TEC-2000), HQ AFSCTR 78-03(three volumes), Dec. 1978.
- 2) S.A. Andriole, and G.W. Hopple, "They're Only Human ; Decision-Makers in Command and Control," Signal, pp. 45-49, March 1982.
- 3) "Special Issue on Application of Kalman Filtering," IEEE Trans. Auto. Contrl, vol. AC-28, No 3, March 1983.
- 4) Man Hyung Lee, "Detection and Tracking for Maneuvering Target," 대한 전기 학회 학술 연구 발표회 논문집, 제19회, pp. 35-37, 1984년 1월.
- 5) T.U. Halawani, "An Optimal Approximation for a Certain Class of Nonlinear Filtering Problems," Ph. D Dissertation, Oregon State University, Covallis OR, March 1983.
- 6) M.H.Lee, W.J. Kolodziej, and R.R. Mohler, "On Nonlinear Filtering and Tracking," Oregon State University, Covallis OR, 97330, Report of ONR, Contract No. N00014-1814, Dec. 1982.
- 7) P.S. Maybeck, Stochastic Model, Estimation, and Control, vol. 2, Academic Press, New York, 1982.
- 8) P.Garnell, Guided Weapon Control System, Second Edition, Pergamon Press, Oxford, New York, 1982.
- 9) R.L. Moose, H.F. Vanlandingham, and D.H. McCabe, "Modeling and Estimation for Tracking Maneuvering," IEEE Trans, Aerosp. Electr. Syst. pp. 448-456, 1979.
- 10) J.K. Tungnait, "Continuous-Time System Identification on Compact Parameter Sets," Proceeding of 1983 American Control Conference, San Francisco, CA, pp. 1241-1246, June 1983.
- 11) J.B. Pearson, E.B. Stear, "Kalman Filter Applications in Airborne Radar Tracking," IEEE AES-10, No. 3, May 1974.
- 12) F.R. Castella, F.G. Dannebacke, "Analytical Results for the x, y Kalman Tracking Filter," IEEE AES

-20, No. 6. Nov. 1974.

- 13) K.V. Ramachandra, V.S. Srinivasan, "Steady State Results for the x, y, z, Kalman Tracking Filter, IEEE AES-13, No. 4, Jul. 1977.
- 14) R.A. Singer K.W. Behnke, "Real time Tracking Filter Evaluation and Selection for Tactical Applications. "IEEE AES-7, Jan. 1971.
- 15) H.Goldstein, Classical Mechanics, Cambridge, Mass. Adison-Wesley, 1950.
- 16) Digital Singal Processing Applications with the TMS320 Family, Texas Instruments, 1983.
- 17) TMS32010 User's Guide, Texas Instruments, 1983.
- 18) 이 만형, 김종화, "적응 비선형 필터링과 전략적 책략 이동목표물의 추적에 관하여," 대한전기학회 논문집, 제36권 제12호, pp. 908-917, 1987년 12월.
- 19) 이만형 외6인, "기동목표물의 수학적 모델링과 칼만필터를 이용한 실시간 추적 기법의 개발," 산학협동재단 최종연구보고서, 1986. 7.
- 20) Man Hyung Lee and Jong Hwa Kim, "Target Tracking and Design of Fire Control System Using Kalman Filtering," The 27th SICE Annual Conf., pp. 827-830, Aug. 2-4, 1988.





