

오프라인 항법을 위한 비선형 고정구간 스무더 설계

Design of Nonlinear Fixed-interval Smoother for Off-line Navigation

유재종, 이장규, 박찬국, 한형석

(Jaejong Yu, Jang Gyu Lee, Chan Gook Park and Hyung Seok Han)

Abstract : We propose a new type of nonlinear fixed interval smoother to which an existing nonlinear smoother is modified. The nonlinear smoother is derived from two-filter formulas. For the backward filter, the propagation and the update equation of error states are derived. In particular, the modified update equation of the backward filter uses the estimated error terms from the forward filter. Data fusion algorithm, which combines the forward filter result and the backward filter result, is altered into the compatible form with the new type of the backward filter. The proposed algorithm is more efficient than the existing one because propagation in backward filter is very simple from the implementation point of view. We apply the proposed nonlinear smoothing algorithm to off-line navigation system and show the proposed algorithm estimates position, and attitude fairly well through the computer simulation.

Keywords : extended kalman filter, linearized filter, nonlinear smoother, off-line navigation

I. 서론

스무딩(smoothing) 기법은 임의의 시간에서 시스템 상태 변수 추정을 그 시간 이후의 측정치까지 이용하는 방법으로 필터링(filtering) 기법과 비교하면 추정오차가 작은 결과를 얻을 수 있다. 그러나, 실시간으로 구현하는 것이 불가능하기 때문에 시험에서 얻은 데이터를 이용하여 궤적을 추적하는 것과 같이 오프라인(off-line)으로 데이터를 처리할 수 있는 경우에 많이 적용되고 있다. 스무딩은 추정하려는 시점과 측정치 이용 방법에 따라 고정시점(fixed-point), 고정지연(fixed-lag), 고정구간(fixed-interval) 등 세 가지로 분류되며, 본 논문에서 다루고 있는 고정구간 스무딩은 고정구간 내에 있는 모든 측정치를 이용하여 상태변수를 추정하는 방법이다[1].

고정구간 스무더에 대한 기본적인 구조는 정방향 필터(forward filter), 역방향 필터(backward filter), 그리고, 이들 추정 결과를 융합하는 부분으로 구성되는 이중필터(two-filter) 구조를 갖고 있으며[2], 스무딩 기법에 대한 연구 결과는 대부분 선형 시스템에 대한 것이다. 비선형 시스템에 대한 스무딩 알고리즘은 Leondes[3] 등에 의하여 제안되었으며, 기본적인 구조는 선형 시스템인 경우와 마찬가지로 이중필터 구조를 사용한다[4].

이러한 이중필터 구조를 갖는 비선형 시스템의 스무더는 정방향 필터로 확장 칼만필터(extended Kalman filter)를 사용하고, 역방향 필터로 정방향 필터의 추정 결과를 공칭점(nominal point)으로 사용하는 비선형 시스템에 대한 선형화

본 논문에서는 기존의 이중필터 구조를 갖는 비선형 스무더에서 정방향 필터는 기존의 방식과 동일하게 확장 칼만필터를 사용하고, 역방향 필터는 새로운 형태의 알고리즘으로 수정하여 제안하였다. 또한, 제안된 알고리즘의 역방향 필터 결과를 정방향 필터와 융합(fusion)하는 알고리즘도 수정하였다.

제안된 비선형 스무딩 기법에 대한 성능을 확인하기 위하여 스트랩다운 방식을 사용하는 관성항법 시스템을 대상으로 시뮬레이션을 수행하였다. 본 논문에서 대상으로 하는 항법시스템은 시스템을 탑재한 기구를 일정시간 주행시키면서 저장한 데이터를 이용하여 주행 궤적을 보다 정확하게 추정하는 시스템이며, 오프라인 방식으로 항법 해를 구하기 때문에 제안된 스무딩 기법을 적용하기에 적합한 시스템이다.

본 논문의 구성은 II장에서 기존의 비선형 스무더를, III장에서는 새로이 제안하는 스무더를 각각 기술하고, IV장에서 제안된 알고리즘을 오프라인 항법 시스템에 적용한 시뮬레이션 결과를 제시하고, V장에서 결론을 맺는다.

II. 기존의 비선형 스무더

비선형 시스템의 고정구간 스무더는 기본적으로 정방향 및 역방향 필터에서 시스템의 상태변수를 추정한 후 그 결과를 융합(fusion) 알고리즘을 사용하여 최종적으로 스무딩된 상태변수를 구하는 이중필터 구조를 사용한다. 이러한 비선형 고정구간 스무더는 정방향 필터로 확장 칼만필터를, 역방향 필터로 선형화 칼만필터를 각각 이용한다[4]. 비선형 스무더를 유도하기 위하여 다음과 같은 비선형 연속 시스템 모델 및 이산 측정치 모델을 가정한다.

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

$$z(t_i) = h[x(t_i), t_i] + v(t_i) \quad (2)$$

논문접수 : 2002. 4. 1., 채택확정 : 2002. 9. 27.

유재종 : 서울대학교 전기컴퓨터공학부(jaejong@asrignc3.snu.ac.kr)

이장규 : 서울대학교 전기컴퓨터공학부(jgl@snu.ac.kr)

박찬국 : 광운대학교 정보제어공학과(cgpark@daisy.kwangwoon.ac.kr)

한형석 : 경원대학교 전기전자공학부(hshan@mail.kyungwon.ac.kr)

※ 본 논문은 서울대학교 자동제어특화연구센터와 광운대학교

2001년도 교내학술 연구비 지원에 의한 연구 결과입니다.

필터(linearized filter)를 사용한다.

여기서, 공정잡음 $w(t)$ 및 측정잡음 $v(t_i)$ 는 평균이 영(zero)인 가우시안 분포를 갖는 백색잡음으로 각각 $E\{w(t)w^T(t+\tau)\} = Q(t)\delta(\tau)$, $E\{v(t_i)v^T(t_j)\} = R(t_i)\delta_{ij}$ 를 만족하며, 공정잡음 $w(t)$ 와 측정잡음 $v(t_k)$ 는 상관관계가 없는 $E\{w(t)v^T(t_k)\} = 0 \forall t, t_k$ 의 관계를 만족한다고 가정한다.

1. 비선형 정방향 필터(forward filter)

비선형 시스템의 스무더에서 정방향 필터로 사용되는 확장 칼만 필터는 실시간으로 상태변수를 추정하는데 많이 사용되고 있으며, 본 절에서는 (1) 및 (2)로 주어진 모델에 대한 확장 칼만 필터 알고리즘을 정리하였다. 여기서 언급된 필터 알고리즘은 III장에서 언급된 새로운 스무더 필터 알고리즘에서 사용한다.

상태변수의 초기 조건이 $x(0) \sim N[\hat{x}_0, P_0]$ 와 같은 정규분포를 갖는다고 가정하면, 시스템 상태변수 추정치 $\hat{x}(t)$ 및 공분산 행렬 $P(t)$ 에 대한 시간 전달(time propagation) 식은 다음과 같다.

$$\hat{x}(t | t_{k-1}) = f[x(t | t_{k-1}), t], \hat{x}(t_{k-1} | t_{k-1}) = \hat{x}(t_{k-1}^+) \quad (3)$$

$$P(t | t_{k-1}) = F[t; \hat{x}(t | t_{k-1})]P(t) + P(t)F^T[t; \hat{x}(t | t_{k-1})] + G(t)Q(t)G^T(t), \quad P(t_{k-1} | t_{k-1}) = P(t_{k-1}^+) \quad (4)$$

이때, 시간은 $t_{k-1}^+ \leq t \leq t_k$ 를 만족하며, 뺄셈자 +는 측정치 갱신 직후를, -는 측정치 갱신 직전을 각각 나타낸다.

측정치가 있을 경우($t = t_k^+$)에 갱신(measurement update) 식은 다음과 같이 주어진다.

$$\hat{x}(t_k^+) = \hat{x}(t_k) + K_k \{z(t_k) - h[\hat{x}(t_k), t_k]\} \quad (5)$$

$$K_k = \frac{P(t_k)H^T[t_k; \hat{x}(t_k)] \{H[t_k; \hat{x}(t_k)] P(t_k)H^T[t_k; \hat{x}(t_k)] + R(t_k)\}^{-1}}{\quad} \quad (6)$$

$$P(t_k^+) = \{I - K_k H[t_k; \hat{x}(t_k)]\} P(t_k^-) \{I - K_k H[t_k; \hat{x}(t_k)]\}^T + K_k R(t_k) K_k^T \quad (7)$$

여기서,

$$F[t; \hat{x}(t | t_k)] = \left. \frac{\partial f[\hat{x}(t), t]}{\partial x} \right|_{x = \hat{x}(t | t_k)}, \quad \forall t \in [t_k, t_{k+1})$$

$$H[t_k; \hat{x}(t_k)] = \left. \frac{\partial h[\hat{x}(t_k), t_k]}{\partial x} \right|_{x = \hat{x}(t_k)}$$

K_k 는 칼만필터 이득 행렬을 각각 나타낸다.

확장 칼만 필터는 실시간으로 항법 해를 구하는 시스템에서 항법 오차 보정을 위하여 많이 사용하고 있는데, 본 논문에서는 추정된 상태변수 값이 선형화 필터로 구현되는 역방향 필터의 기준궤적(nominal trajectory)으로도 사용된다.

2. 비선형 역방향 필터(backward filter)

비선형 시스템에서는 선형화를 위한 기준 궤적이 주어진 경우에 선형화 필터를 사용하여 주어진 시스템의 상태변수를 추정하는데, 비선형 스무더는 이러한 선형화 필터를 역방향 필터의 알고리즘으로 사용한다. 본 절에서는 선형화 필터에 대한 식을 소개한 후 이를 이용하여 역방향 필터를 유도한다.

2.1. 선형화 필터

비선형 시스템을 선형화하기 위한 기준 궤적 $x_n(t)$ 및 이와 관련된 측정치 $z_n(t_k)$ 가 다음과 같이 주어졌다고 가정한다.

$$\dot{x}_n(t) = f[x_n(t), t], \quad x_n(t_0) = x_0 \quad (8)$$

$$z_n(t_k) = h[x_n(t_k), t_k] \quad (9)$$

상태변수의 초기 조건이 $x(0) \sim N[\hat{x}_0, P_0]$ 와 같은 정규분포를 갖는다고 가정하면, 시스템의 상태변수 추정치 $\hat{x}(t)$ 및 공분산 행렬 $P(t)$ 에 대한 시간 전달($t_{k-1}^+ \leq t \leq t_k$) 식은 다음과 같다.

$$\hat{x}(t) = f[x_n(t), t] + F[t; x_n(t)] \{\hat{x}(t) - x_n(t)\} \quad (10)$$

$$P(t) = F[t; x_n(t)]P(t) + P(t)F^T[t; x_n(t)] + G(t)Q(t)G^T(t) \quad (11)$$

측정치가 있을 경우($t = t_k^+$)에 갱신 식은 다음과 같이 주어진다.

$$\hat{x}(t_k^+) = \hat{x}(t_k) + K_k \{z(t_k) - h[x_n(t_k), t_k] - H[t_k; x_n(t_k)] \{\hat{x}(t_k) - x_n(t_k)\}\} \quad (12)$$

$$K_k = \frac{P(t_k)H^T[t_k; x_n(t_k)] \{H[t_k; x_n(t_k)] P(t_k)H^T[t_k; x_n(t_k)] + R(t_k)\}^{-1}}{\quad} \quad (13)$$

$$P(t_k^+) = \{I - K_k H[t_k; x_n(t_k)]\} P(t_k^-) \{I - K_k H[t_k; x_n(t_k)]\}^T + K_k R(t_k) K_k^T \quad (14)$$

여기서, $F[t; x_n(t)] = \left. \frac{\partial f[x(t), t]}{\partial x} \right|_{x = x_n(t)}$,

$H[t_k; x_n(t_k)] = \left. \frac{\partial h[x(t_k), t_k]}{\partial x} \right|_{x = x_n(t_k)}$ 를 나타낸다.

2.2. 역방향 필터

역방향 필터는 시간 전달을 정방향 필터와 반대로 하기 때문에 필터 식을 새로운 변수 $\tau = (t_j - t)$ 로 정의하여 유도한다. 정방향 필터의 결과인 (3)을 선형화를 위한 기준궤적으로 이용하여 이를 (10) 및 (11)에 대입하면 다음과 같이 정리할 수 있다.

$$\begin{aligned} \frac{d}{dt} \hat{x}_b(\tau) = & -f[\hat{x}_f(t_f - \tau), t_f - \tau] \\ & - F[t_f - \tau; \hat{x}_f(t_f - \tau)]\{\hat{x}_b(\tau) - \hat{x}_f(t_f - \tau)\} \end{aligned} \quad (15)$$

$$\begin{aligned} \frac{d}{dt} P_b(\tau) = & -F[t_f - \tau; \hat{x}_f(t_f - \tau)]P_b(\tau) \\ & - P_b(\tau)F^T[t_f - \tau; \hat{x}_f(t_f - \tau)] \\ & + G(t_f - \tau)Q(t_f - \tau)G^T(t_f - \tau) \end{aligned} \quad (16)$$

여기서, $d\tau = -dt$, $\hat{x}_f(t_f - \tau)$ 는 정방향 필터의 추정 결과이다.

일반적으로 역방향 필터에서는 상태변수 초기 값에 대한 정보가 없기 때문에 초기 공분산은 $P_b(t_f) = \infty$, 또는 $P_b^{-1} = 0$ 으로 주어지며, 구현의 편리성 등을 고려하여 정보 필터(information filter) 형태로 구현한다[1]. 역방향 필터에 대한 초기값 $\hat{x}_b(t_f)$ 은 모르지만 유한한 값이므로 다음과 같이 초기값이 영(zero)인 새로운 변수를 정의하여 역방향 필터를 정보필터 형태로 구현한다.

$$\begin{cases} \hat{y}_b(t_f) = P_b^{-1}(t_f) \hat{x}_b(t_f) = 0 \\ \hat{y}_b(t_k^-) = P_b^{-1}(t_k) \hat{x}_b(t_k^-) \\ \hat{y}_b(t_k^+) = P_b^{-1}(t_k^+) \hat{x}_b(t_k^+) \end{cases} \quad (17)$$

(17)의 관계를 이용하여 (15) 및 (16)을 공분산 역행렬 $P_b^{-1}(\tau)$ 및 상태변수 $\hat{y}_b(\tau)$ 에 대한 시간 전달 식으로 나타내면 다음과 같다.

$$\begin{aligned} \frac{d}{dt} P_b^{-1}(\tau) = & -P_b^{-1}(\tau)\{\frac{d}{dt} P_b(\tau)\}P_b^{-1}(\tau) \\ = & P_b^{-1}(\tau)\overline{F}(\tau) + \overline{F}^T(\tau)P_b^{-1}(\tau) \\ & - P_b^{-1}(\tau)\overline{Q}(\tau)P_b^{-1}(\tau) \end{aligned} \quad (18)$$

$$\begin{aligned} \frac{d}{dt} \hat{y}_b(\tau) = & \frac{d}{dt} \{P_b^{-1}(\tau) \hat{x}_b(\tau)\} \\ = & \{\frac{d}{dt} P_b^{-1}(\tau)\} \hat{x}_b(\tau) + P_b^{-1}(\tau)\{\frac{d}{dt} \hat{x}_b(\tau)\} \\ = & \{\overline{F}^T(\tau) - P_b^{-1}(\tau)\overline{Q}(\tau)\} \hat{y}_b(\tau) \\ & - P_b^{-1}(\tau)\{\overline{A}[\hat{x}_f(t_f - \tau), t_f - \tau] \\ & - \overline{F}(\tau) \hat{x}_f(t_f - \tau)\} \end{aligned} \quad (19)$$

여기서, $\overline{F}(\tau) = F[t_f - \tau; \hat{x}_f(t_f - \tau)]$, $\overline{Q}(\tau) = G(t_f - \tau)Q(t_f - \tau)G^T(t_f - \tau)$, $\hat{y}_b(\tau) = P_b^{-1}(\tau) \hat{x}_b(\tau)$ 를 각각 나타내며, 초기 조건은 $\hat{y}_b(0) = 0, P_b^{-1}(0) = 0$ 와 같다.

측정치 갱신 식은 (12)를 이용하면 다음과 같이 나타낼 수 있다.

$$\begin{aligned} \hat{y}_b(\tau_k^+) = & P_b^{-1}(\tau_k^+) \hat{x}_b(\tau_k^+) \\ = & P_b^{-1}(\tau_k^+) \hat{x}_b(\tau_k^+) + P_b^{-1}(\tau_k^+)K_b(\tau_k)\{z(\mu_k) \\ & - h[\hat{x}_f(\mu_k^+), \mu_k] - H(\mu_k)[\hat{x}_b(\tau_k) - \hat{x}_f(\mu_k^+)]\} \end{aligned}$$

$$\begin{aligned} = & \hat{y}_b(\tau_k) + H^T(\mu_k)R^{-1}(\mu_k)\{z(\mu_k) \\ & - h[\hat{x}_f(\mu_k^+), \mu_k] + H(\mu_k) \hat{x}_b(\tau_k)\} \end{aligned} \quad (20)$$

$$P_b^{-1}(\tau_k^+) = P_b^{-1}(\tau_k) + H^T(\mu_k)R^{-1}(\mu_k)H(\mu_k) \quad (21)$$

여기서, $\mu_k = t_f - \tau_k$ 를 나타내며, 공분산 역행렬에 대한 갱신을 나타내는 (21)과 필터 이득 행렬 (13)에 대한 관계식을 이용하면 $P_b^{-1}(\tau_k^+)K_b(\tau_k) = H^T(\mu_k)R^{-1}(\mu_k)$ 임을 알 수 있다.

3. 데이터 융합(data fusion) 알고리즘

정방향 필터 결과인 $\hat{x}_f(t_k^+), P_f(t_k^+)$ 와 역방향 필터 결과인 $\hat{y}_b(\tau_k), P_b(\tau_k)$ 로부터 스무딩된 상태변수는 다음 식을 이용하여 계산한다.

$$\hat{x}_s(t_k) = P_s(t_k)[P_f^{-1}(t_k^+) \hat{x}_f(t_k^+) + \hat{y}_b(\tau_k)] \quad (22)$$

$$P_s^{-1}(t_k) = P_f^{-1}(t_k^+) + P_b^{-1}(\tau_k^-) \quad (23)$$

여기서, $\tau_k = t_f - t_k$ 이며, 역방향 필터의 시간전달 및 측정치 갱신은 최종시간 t_f 에서 시작한다.

이러한 데이터 융합은 측정치가 있는 시점에서만 적용할 수 있기 때문에 측정치와 측정치 사이에서 상태변수는 시스템 모델로 주어진 다음의 비선형 방정식을 수치적으로 적분하여 계산한다.

$$\dot{\hat{x}}_s(t) = f[\hat{x}_s(t), t] \quad (24)$$

III. 새로운 비선형 스무더

기존의 이중필터 구조를 갖는 비선형 스무더로부터 유도되는 새로운 스무더는 II절에서 언급한 역방향 필터 및 데이터 융합 알고리즘을 수정한 것으로, 기본적인 필터의 특성은 기존의 결과와 동일하다. 그러나, 기존의 알고리즘과 비교하여 제안된 알고리즘은 구현이 쉽고, 물리적인 의미를 쉽게 파악할 수 있다. 비선형 스무더의 정방향 필터는 II.1절에서 언급했던 확장 칼만필터를 사용하며, 추정된 상태변수는 역방향 필터에 대한 기준계적으로 사용된다.

역방향 필터를 유도하기 위하여 II.2.절에서 기술했던 알고리즘을 이용하여 상태변수의 오차에 대한 식으로 수정하면 다음과 같다.

$$\begin{aligned} \frac{d}{dt} \hat{y}_b(\tau) = & \frac{d}{dt} \{P_b^{-1}(\tau) \hat{x}_b(\tau)\} \\ = & \frac{d}{dt} \{P_b^{-1}(\tau)(x_n(\tau) - \delta\hat{x}_b(\tau))\} \\ = & \frac{d}{dt} \{P_b^{-1}(\tau)x_n(\tau)\} - \frac{d}{dt} \{P_b^{-1}(\tau) \delta\hat{x}_b(\tau)\} \\ = & \{P_b^{-1}(\tau)\overline{F}(\tau) + \overline{F}^T(\tau)P_b^{-1}(\tau) \\ & - P_b^{-1}(\tau)\overline{Q}(\tau)P_b^{-1}(\tau)\}x_n(\tau) \\ & + P_b^{-1}(\tau)\frac{d}{dt} x_n(\tau) - \frac{d}{dt} \{P_b^{-1}(\tau) \delta\hat{x}_b(\tau)\} \end{aligned} \quad (25)$$

(25)는 상태변수 $\hat{x}_b(\tau)$ 를 기준케적에 대한 상태변수 $x_n(\tau)$ 와 섭동량 $\delta\hat{x}_b(\tau)$ 로 분리하여 나타낸 것이며, $\delta\hat{y}_b(\tau)$ 의 시간 전달은 $x_n(\tau) = \hat{x}_f(t_f - \tau)$ 관계 및 (19)를 이용하여 다음과 같이 표시할 수 있다.

$$\begin{aligned} \frac{d}{dt} \delta\hat{y}_b(\tau) &= \frac{d}{dt} \{P_b^{-1}(\tau) \delta\hat{x}_b(\tau)\} \\ &= \{P_b^{-1}(\tau)\overline{F}(\tau) + \overline{F}^T(\tau)P_b^{-1}(\tau) \\ &\quad - P_b^{-1}(\tau)\overline{Q}(\tau)P_b^{-1}(\tau)\} \hat{x}_f(t_f - \tau) \\ &\quad - P_b^{-1}(\tau)f[\hat{x}_f(t_f - \tau), t_f - \tau] \\ &\quad - \{\overline{F}^T(\tau) - P_b^{-1}(\tau)\overline{Q}(\tau)\} \hat{y}_b(\tau) \\ &\quad + P_b^{-1}(\tau)\{f[\hat{x}_f(t_f - \tau), t_f - \tau] - \overline{F}(\tau) \hat{x}_f(t_f - \tau)\} \\ &= \{\overline{F}^T(\tau) - P_b^{-1}(\tau)\overline{Q}(\tau)\} \delta\hat{y}_b(\tau) \end{aligned} \quad (26)$$

측정치 갱신은 그림 1에 나타난 것처럼 측정치가 있는 시점에서 정방향 필터 및 역방향 필터 관계를 이용하여 얻을 수 있다. 그림에서 정방향 필터 추정치 $\hat{x}_f(t_f - \tau_k)$ 를 선형화를 위한 기준케적으로 사용하면 역방향 필터의 추정치 $\hat{x}_b(\tau_k)$ 는 다음과 같은 관계를 얻을 수 있다.

$$\begin{cases} \hat{x}_b(\tau_k^+) = \hat{x}_f(\mu_k) - \delta\hat{x}_b(\tau_k^+) \\ \hat{x}_b(\tau_k^-) = \hat{x}_f(\mu_k^+) - \delta\hat{x}_b(\tau_k^-) \\ \hat{x}_f(\mu_k^+) = \hat{x}_f(\mu_k) - \delta\hat{x}_f(\mu_k) \end{cases} \quad (27)$$

여기서, $\mu_k = t_f - \tau_k$ 이며, $\delta\hat{x}_b(\tau_k)$ 는 상태변수 $x_f(\mu_k)$ 로부터 섭동량으로 본 논문에서 제안하는 역방향 필터의 상태변수를 나타낸다.

(27)을 (20)에 대입하여 정리하면 다음과 같은 관계식을 얻을 수 있다.

$$\begin{aligned} P_b^{-1}(\tau_k^+) \hat{x}_b(\tau_k^+) &= P_b^{-1}(\tau_k^+) \{ \hat{x}_f(\mu_k) - \delta\hat{x}_b(\tau_k^+) \} \\ &= P_b^{-1}(\tau_k) \{ \hat{x}_f(\mu_k^+) - \delta\hat{x}_b(\tau_k^-) \} \\ &\quad + H^T(\mu_k)R^{-1}(\mu_k) \{ \delta z(\mu_k) + H(\mu_k) \hat{x}_f(\mu_k) \} \end{aligned} \quad (28)$$

여기서, $\delta z(\mu_k) = z(\mu_k) - h[\hat{x}_f(\mu_k), \mu_k]$ 를 나타낸다.

(28)에서 $\delta\hat{y}_b(\tau_k^+)$ 에 대한 관계식을 유도하면 다음과 같이 나타낼 수 있으며, 공분산 역행렬에 대한 갱신은 앞서 언급했던 (21)을 이용한다.

$$\begin{aligned} \delta\hat{y}_b(\tau_k^+) &= P_b^{-1}(\tau_k^+) \delta\hat{x}_b(\tau_k^+) \\ &= P_b^{-1}(\tau_k^+) \hat{x}_f(\mu_k) \\ &\quad - P_b^{-1}(\tau_k) \{ \hat{x}_f(\mu_k^+) - \delta\hat{x}_b(\tau_k^-) \} \\ &\quad - H^T(\mu_k)R^{-1}(\mu_k) \{ \delta z(\mu_k) + H(\mu_k) \hat{x}_f(\mu_k) \} \\ &= P_b^{-1}(\tau_k) \delta\hat{x}_f(\mu_k) \\ &\quad + P_b^{-1}(\tau_k) \delta\hat{x}_b(\tau_k) - H^T(\mu_k)R^{-1}(\mu_k) \delta z(\mu_k) \\ &= \frac{P_b^{-1}(\tau_k) \delta\hat{x}_f(\mu_k)}{\delta\hat{y}_b(\tau_k)} - H^T(\mu_k)R^{-1}(\mu_k) \delta z(\mu_k) \end{aligned} \quad (29)$$

스무딩 상태변수 $\hat{x}_s(t_k)$ 는 제안된 방법으로 $\delta\hat{y}_b(\tau_k)$ 를 추정된 후 정방향 필터 추정치인 $\hat{x}_f(t_k)$ 와 다음과 같은 방법으로 융합하여 얻을 수 있다.

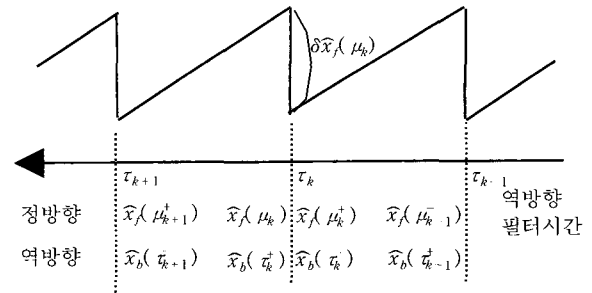


그림 1. 정방향 필터/역방향 필터 상태변수 관계.
Fig. 1. Relation between forward filter and backward filter.

$$\begin{aligned} \hat{x}_s(t_k) &= P_s(t_k)[P_f^{-1}(t_k^+) \hat{x}_f(t_k^+) + \hat{y}_b(\tau_k^+)] \\ &= P_s(t_k) \{ P_f^{-1}(t_k^+) \hat{x}_f(t_k^+) \\ &\quad + P_b^{-1}(\tau_k^-) [\hat{x}_f(t_k^+) - \delta\hat{x}_b(\tau_k^-)] \} \\ &= \hat{x}_f(t_k^+) - P_s(t_k) \delta\hat{y}_b(\tau_k) \end{aligned} \quad (30)$$

여기서, $P_s^{-1}(t_k) = P_f^{-1}(t_k^+) + P_b^{-1}(\tau_k)$ 를 나타내며, 역방향 필터에서 $\tau_k = t_f - t_k$ 이다.

본 논문에서 제안한 비선형 스무딩 필터 알고리즘은 역방향 필터 결과 $\delta\hat{y}_b(t)$ 를 사용하기 때문에 (19), (20), (22)와 비교하면 보다 간단하게 알고리즘을 구성할 수 있음을 알 수 있다.

IV. 오프라인 항법 시스템에 적용

제안한 비선형 스무더에 대한 성능을 검토하기 위하여 시뮬레이션을 수행하였다. 시뮬레이션에서 사용한 대상 시스템은 자이로 3개와 가속도계 3개로 구성된 관성센서 유닛(IMU)을 항체에 직접 장착하는 스트랩다운 관성항법 시스템이다. 항법 시스템이 장착된 기구를 일정 시간 동안 수행시킨 후에 수행 중에 저장된 각종 센서 데이터를 이용하여 오프라인 방식으로 항법 해를 구하는 시스템으로 항법 해의 오차를 줄이기 위하여 비선형 스무더를 적용하기에 매우 적합한 시스템이다. 본 절에서는 비선형 필터 구성을 위한 항법 오차 모델 및 센서 오차 모델에 대하여 기술하고, 제안된 스무딩 기법을 주어진 모델에 적용한 시뮬레이션 결과를 제시한다.

1. 항법 오차 모델

스트랩다운 관성항법 오차 모델은 지구반경 모델의 위도에 대한 변화율 및 중력 성분에 대한 오차 δg^n 는 무시하고 유도하였으며, 그 결과를 정리하면 다음과 같다[5].

$$\delta L = \frac{\rho_E}{R_m + h} \delta h + \frac{\delta v_N}{R_m + h} \quad (31)$$

$$\delta l = \rho_N \sec L \tan L \delta L - \frac{\rho_N \sec L}{R_t + h} \delta h + \frac{\sec L}{R_t + h} \delta v_E \quad (32)$$

$$\delta h = -\delta v_D \quad (33)$$

$$\begin{aligned} \delta \dot{v}^n &= [C_b^n f^b] \times \phi - [2\omega_{ie}^n + \omega_{en}^n] \times \delta v^n \\ &\quad + C_b^n \delta f^b + v^n \times (2\delta\omega_{ie}^n + \delta\omega_{en}^n) \end{aligned} \quad (34)$$

$$\phi = -\omega_{in}^n \times \phi - C_b^n \delta \omega_{ib}^b + \delta \omega_{in}^n \quad (35)$$

여기서, δL 은 위도(L) 오차, δl 은 경도(l) 오차, δh 는 고도(h) 오차를, $\delta v^n = [\delta v_N \ \delta v_E \ \delta v_D]^T$ 는 항법 좌표계에서 속도(v^n) 오차를 각각 나타낸 것이다. 자세오차는 항법 컴퓨터에서 계산된 해석적 플랫폼(analytic platform) 좌표계와 실제 항법 좌표계 사이의 미소각(tilt angle)으로 정의할 수 있으며, $\phi = [\phi_N \ \phi_E \ \phi_D]^T$ 는 이러한 자세오차를 나타낸 것이다. 또한, R_m 은 meridian radius, R_t 는 transverse radius, C_b^n 은 동체좌표계에서 항법좌표계로의 변환행렬, Ω 는 지구 자전 각속도를 각각 나타내며, $\rho_{N,E,D}$, δw_{ic}^n 및 δw_{en}^n 은 각각 다음과 같이 정의된다.

$$[\rho_N \ \rho_E \ \rho_D]^T = [l \cos L \ -L \ -l \sin L]^T \quad (36)$$

$$\delta \omega_{ic}^n = [-\Omega \sin L \ \delta L \ 0 \ -\Omega \cos L \ \delta L]^T \quad (37)$$

$$\delta \omega_{en}^n \approx \begin{bmatrix} -\frac{\rho_N}{R_t+h} \delta h + \frac{1}{R_t+h} \delta v_E \\ -\frac{\rho_E}{R_m+h} \delta h - \frac{1}{R_m+h} \delta v_N \\ \rho_N \sec^2 L \delta L - \frac{\rho_D}{R_t+h} \delta h + \frac{\rho_D}{v_E} \delta v_E \end{bmatrix} \quad (38)$$

자세오차를 나타내는 (35)에서, $\omega_{in}^n = \omega_{ic}^n + \omega_{en}^n$ 으로 표현할 수 있기 때문에 (37)과 (38)의 합으로 $\delta \omega_{in}^n$ 을 나타낼 수 있다. 또한, (34)에서 δf^b 는 가속도계 오차를, (35)에서 $\delta \omega_{ib}^b$ 는 자이로 오차를 각각 나타내며, 이들 관성센서에 대한 오차는 다음과 같다.

관성센서인 가속도계 및 자이로에 대한 오차 요인으로는 환산계수(scale factor) 오차 등을 비롯하여 여러 가지가 있으나[5] 시뮬레이션에서는 가속도계 오차 및 자이로 오차를 랜덤상수(random constant) 및 백색잡음의 합으로 다음과 같이 모델링하였다.

$$\delta f^b = \nabla_a + w_a \quad (39)$$

$$\delta \omega_{ib}^b = \varepsilon_g + w_g \quad (40)$$

$$\nabla_a = 0 \quad (41)$$

$$\varepsilon_g = 0 \quad (42)$$

여기서, $\nabla_a = [\nabla_x \ \nabla_y \ \nabla_z]^T$ 는 가속도계 랜덤상수 오차를, $\varepsilon_g = [\varepsilon_x \ \varepsilon_y \ \varepsilon_z]^T$ 는 자이로 랜덤상수 오차를, w_a 는 가속도계 백색잡음을, w_g 는 자이로 백색잡음을 각각 나타낸다.

2. 측정치 오차 모델

항법 해를 보정하기 위하여 사용하는 보조센서인 주행거리계(odometer) 출력 및 위치 기준점(position reference point)에 대한 측정치 오차 모델은 다음과 같다.

주행거리계 출력으로부터 속도를 얻는데 발생할 수 있는 오차 요인으로는 환산계수 오차만을 고려하였으며, 수식으로 나타내면 다음과 같다.

$$V_m^b \approx (1 - K_{xo}) V_x^b \quad (43)$$

여기서, V_m^b 는 주행거리계 출력속, K_{xo} 는 환산계수 오차를, V_x^b 는 동체의 진행방향 속도를 각각 나타낸다.

(43)를 이용하여 항법 좌표계에서 속도 측정치에 대한 오차 모델을 유도하면 다음과 같다[6].

$$\delta V_m = \delta v^n - \mathcal{T}^n \times \begin{bmatrix} \phi_N \\ \phi_E \\ \phi_D \end{bmatrix} - \mathcal{C}_b^n V_x^b \Delta K + \nu_v(t) \quad (44)$$

여기서, \mathcal{T}^n 는 항법 좌표계로 표현된 속도 추정치를, \mathcal{C}_b^n 는 추정된 값을 이용하여 계산된 좌표변환행렬을, ΔK 는 추정된 환산계수 오차를, $\nu_v(t)$ 는 속도 측정치에 대한 잡음을 각각 나타내며, 속도 측정치에 대한 잡음은 평균이 영(zero)인 백색잡음으로 확률분포는 가우시안 정규분포를 갖는다고 가정하였다.

위치 기준점은 위도, 경도 및 고도에 대한 정보를 제공하기 때문에 위치 정보에 대한 측정치 오차 모델은 다음과 같이 간단하게 모델링할 수 있다.

$$\begin{aligned} \delta P_m &= P - P_m = (P_{true} + \delta P) - P_{true} + \nu_p(t) \\ &= \delta P + \nu_p(t) \end{aligned} \quad (45)$$

여기서, $\nu_p(t)$ 는 위치 기준점에 대한 오차를 나타내며, 평균이 영(zero)인 정규분포를 갖는다고 가정한다.

3. 시뮬레이션 조건

시뮬레이션에서는 필터 모델로 앞에서 기술한 항법 오차 모델과 주행거리계의 환산계수 오차를 포함한 16차 모델을 사용하였다. 또한, 측정치 모델은 (44) 및 (45)로 주어진 속도 측정치 및 위치 기준점에 대한 오차 모델을 사용하였다.

속도 측정치는 항체의 진행 방향 속도(V_x^b)를 1초 주기로 측정하고, 위치 기준점은 2km 간격으로 설치되어 있다고 가정하였다. 시뮬레이션 수행에 사용된 궤적은 그림 2와 같으며, 초기 위치는 위도 37도, 경도 127도, 고도 0m이다. 속도는 출발 및 도착 시점(0.8m/s²)으로 5초 동안 가속 및 감속 운동을 제외하고는 4m/s의 일정한 속도로 운동한다고 가정하였다. 또한, 그림에서 자세변화 있는 곳에서 피치 레이트(q)는 30deg/s, 요레이트(r)는 45deg/s 크기로 각속도 운동을 하고, 롤 운동은 없다고 가정하였다. 또한, 초기 자세 각 오차에 대한 표준편차(1σ) 값으로 롤 및 피치는 1deg, 요는 2deg를 가정하였다.

센서 오차 모델에 사용된 백색잡음과 랜덤 상수에 대한 표준편차(1σ)는 표 1과 같으며, 항법 성능은 몬테칼로 시뮬레이션을 100회 수행한 후 각 항법 변수에 대한 오차의 표준편차를 구하여 검토하였다.

표 1. 센서에 대한 표준편차(1σ).

Table 1. Standard deviation of sensor error.

자이로	바이어스	3deg/h
	잡음	0.35deg/h
가속도계	바이어스	1mg
	잡음	50ug
위치 기준점	잡음	0.2m
속도 측정치	환산계수 오차	0.1%
	잡음	20mm/s

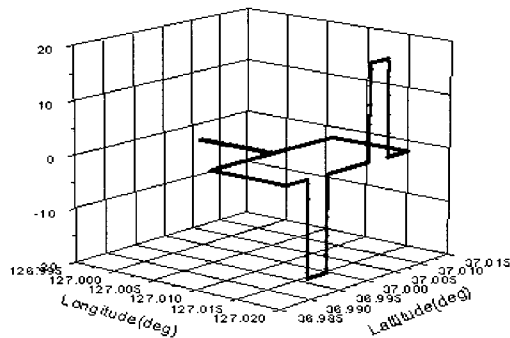
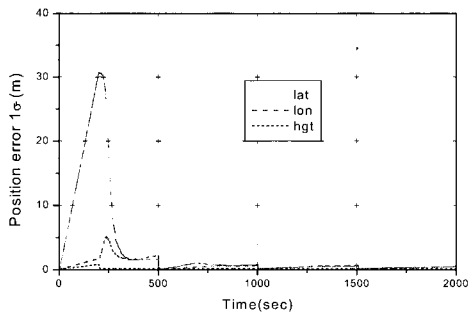
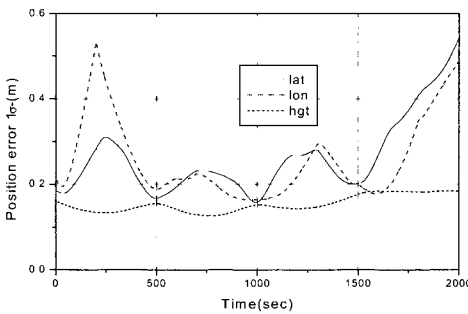


그림 2. 시뮬레이션 궤적.
Fig. 2. Simulation trajectory.

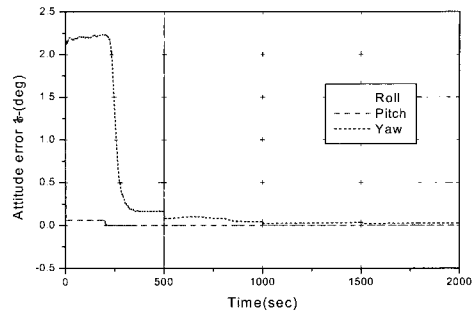


(a) 정방향 필터(forward filter)

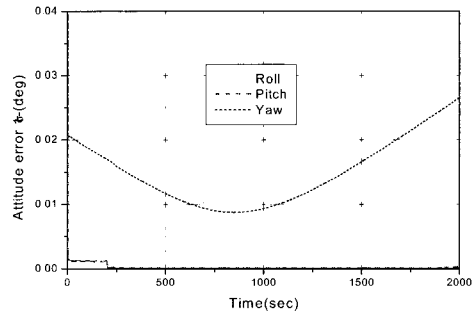


(b) 스무더(smoother)

그림 3. 위치오차 비교.
Fig. 3. Comparison of position errors.



(a) 정방향 필터(forward filter)



(b) 스무더(smoother)

그림 4. 자세오차 비교.
Fig. 4. Comparison of attitude errors.

4. 시뮬레이션 결과

시뮬레이션 결과는 항법 성능에 대한 중요한 지표가 되는 위치 및 자세 오차를 정방향 필터 결과와 제안한 스무더 결과를 비교하여 그림 3~4에 나타낸다. 그림 3은 위치 오차를 비교한 것으로 (a)의 정방향 필터 결과는 위치 기준 점을 이용하는 지점(예, 500초 부근)에서는 오차가 급격하게 감소하지만 위치오차가 약 30m까지 되는 것을 알 수 있다. 그러나, 스무더를 사용한 (b)에서는 위치오차가 0.5m 이내로 제한되는 것을 알 수 있다. 그림 3에서, 최종시간인 2000초 부근에서는 정방향 필터 결과와 스무더 결과가 거의 같은 크기를 갖는데, 이것은 역방향 필터 초기 조건이 $\delta\hat{y}_b(0)=0, P_b^{-1}(0)=0$ 으로 주어지기 때문이다. 자세오차를 비교한 그림 4에서도 스무더를 사용한 결과가 훨씬 좋은 것을 알 수 있다. 그림에서, 요각에 대한 오차 수렴특성이 상대적으로 좋지 않은 것은 스트랩다운 관성항법 시스템에서 일반적인 특징이다.

V. 결론

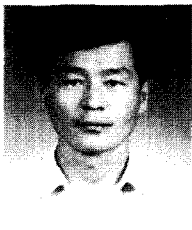
본 논문에서는 기존의 이중 필터 구조를 갖는 비선형 고정구간 스무딩 알고리즘을 수정하여 새로운 형태의 비선형 스무더를 제안하였다. 제안된 필터는 기존의 필터와 특성은 같지만 역방향 필터의 측정치 갱신에서 정방향 필터의 오차 추정 결과를 이용하는 방식으로 수정하여 역방향 필터의 구현을 간단하게 하였다. 또한, 정방향 및 역방향 필터 결과를 융합하는 스무딩 알고리즘도 쉽게 구현할 수 있다. 비선형

특성이 강한 스트랩다운 관성항법 시스템을 대상으로 몬테칼로 시뮬레이션을 수행하여 제안된 스무더에 대한 성능을 확인하였다.

참고문헌

[1] P. S. Maybeck, *Stochastic Models, Estimation, and Control*, Vol. 2, Academic Press, 1982.
 [2] D. C. Fraser and J. E. Potter, "The optimum linear smoother as a combination of two optimum linear filters," *IEEE Trans. on Automatic Control*, Vol. 7, No. 8, pp. 87-390, August 1969.

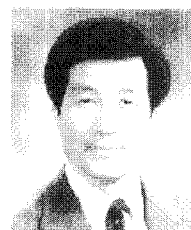
[3] C. T. Leondes, J. B. Peller and E. B. Stear, "Nonlinear smoothing theory," *IEEE Trans. on Systems Science and Cybernetics*, Vol. SSC-6, No. 1, pp. 63-71, January 1970.
 [4] Arthur Gelb, *Applied optimal estimation*, MIT Press, 1974.
 [5] D. H. Titterton, J. L. Weston, *Strapdown Inertial Navigation Technology*, Peter Peregrinus Ltd, 1997.
 [6] Myeong-Jong Yu, Jang Gyu Lee, Heung-Won Park, "Comparison of SDINS In-Flight alignment using equivalent error models," *IEEE Transactions on Aerospace and Electronic Systems*, AES-35, No. 3, pp. 1046-1054, July 1999.



유재중

1965년 2월 12일생. 1987년 서울대학교 공대 제어계측공학과 졸업. 1989년 동대학원 제어계측공학과 졸업(석사). 1989년~현재 국방과학연구소 선임연구원. 2000년~현재 서울대학교 공대 전기컴퓨터공학부 박사과정 재학중. 관

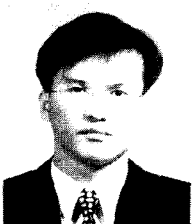
심분야는 유도조종기법, 관성항법 시스템, 고장검출기법.



이장규

1946년 3월 28일생. 1971년 서울대 공대 전기공학과 졸업. 1974년 Univ. of Pittsburgh 전기공학과 졸업(석사). 1977년 Univ. of Pittsburgh 전기공학과 졸업(공학박사) 1977년~1981년 The Analytic Sciences Corporation 연구원(Technical

Staff). 1981년~1982년 Charles Stark Draper Laboratory 연구원(Technical Staff). 1982년~현재 서울대학교 전기·컴퓨터공학부 교수. 1994년~현재 서울대 자동제어특화연구센터 소장. 연구관심분야는 추정이론, 관성항법 센서 및 시스템, GPS, 무선측위 시스템.



박찬국

1961년9월13일생. 1985년 서울대학교 공과대학 제어계측공학과 졸업. 1987년 동대학원 졸업(석사). 1993년 동대학원 졸업(공학박사). 1993년 서울대학교 제어계측신기술연구센터 박사후 연구원. 1994년~현재 광운대학교 정보제어

공학과 부교수. 1998년 미국 UCLA 기계항공공학과 방문연구원. 1994~현재 서울대학교 자동제어특화연구센터 책임연구원. 연구관심분야는 관성항법시스템 개발, GPS/INS 결합기법, MEMS IMU 응용, 칼만필터링, 센서고장검출.



한형석

1963년 6월 30일생. 1986년 서울대학교 공대 제어계측공학과 졸업. 1988년 동대학원 제어계측공학과 졸업(석사). 1993년 동대학원 제어계측공학과 졸업(공학박사). 1993년~1997년 순천향대학교 공대 제어계측공학과 조교수.

1997년~현재 경원대학교 공대 전기전자공학부(전자공학전공) 부교수. 관심분야는 건설제어, 유도 및 제어.