

Reliable Attitude Estimation by Extended Kalman Filter Using Norm Analysis of Sensors

조선대학교 항공우주공학과
항법제어 및 응용연구실
20092984
김종명

Contents

I. Introduction

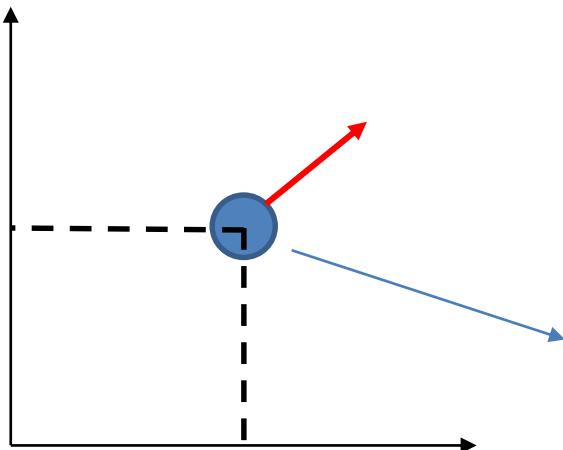
II. Algorithm

III. Simulation & Result

IV. Future Plan

I Introduction

Research purpose



Ideal Value
$p = \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 3 \\ 3 \end{bmatrix}$
$v = \begin{bmatrix} v_x \\ v_y \end{bmatrix} = \begin{bmatrix} 2 \\ 1 \end{bmatrix}$
$a = \begin{bmatrix} a_x \\ a_y \end{bmatrix} = \begin{bmatrix} 0.1 \\ 1 \end{bmatrix}$
$\omega = \begin{bmatrix} \omega_x \\ \omega_y \end{bmatrix} = \begin{bmatrix} 0.02 \\ 0.01 \end{bmatrix}$

Sensor output
$p = \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 3.1 \\ 2.9 \end{bmatrix}$
$v = \begin{bmatrix} v_x \\ v_y \end{bmatrix} = \begin{bmatrix} 1.5 \\ 1.2 \end{bmatrix}$
$a = \begin{bmatrix} a_x \\ a_y \end{bmatrix} = \begin{bmatrix} 0.05 \\ 1.3 \end{bmatrix}$
$\omega = \begin{bmatrix} \omega_x \\ \omega_y \end{bmatrix} = \begin{bmatrix} 0.011 \\ 0.021 \end{bmatrix}$

I Introduction

Research purpose

$$Output = Ideal + \alpha$$

α : Bias, Noise, ...



I Introduction

Research purpose

Sensor

Magnetometer
Accelerometer
Gyro

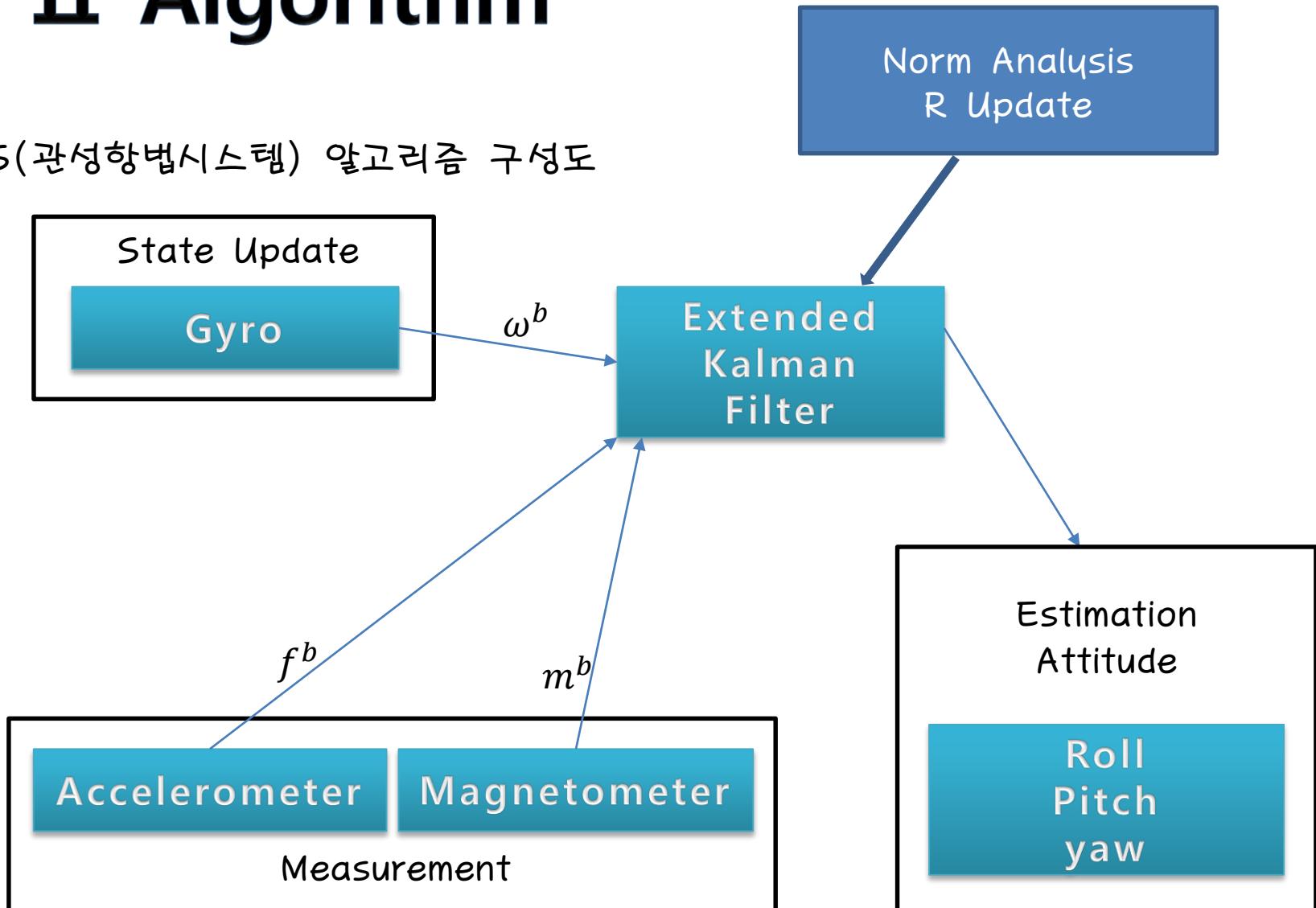
Filter

Extended
Kalman Filter

Norm Analysis + R Update

II Algorithm

INS(관성항법시스템) 알고리즘 구성도



II Algorithm

1. System Model

$$\dot{q} = \frac{1}{2} \Omega(\omega) q$$

$$\Omega(\omega) = \begin{bmatrix} -[\omega \times] & \omega \\ -\omega^T & 0 \end{bmatrix} = \begin{bmatrix} 0 & \omega_z & -\omega_y & \omega_x \\ -\omega_z & 0 & \omega_x & \omega_y \\ \omega_y & -\omega_x & 0 & \omega_z \\ -\omega_x & -\omega_y & -\omega_z & 0 \end{bmatrix}$$

$$\begin{aligned}\dot{\tilde{x}} &= F(t)\Delta\tilde{x} + G(t)w(t) \\ F(t) &= \begin{bmatrix} -[\hat{\omega}(t)\times] & -I_{3\times 3} \\ 0_{3\times 3} & 0_{3\times 3} \end{bmatrix} \\ G(t) &= \begin{bmatrix} -I_{3\times 3} & 0_{3\times 3} \\ 0_{3\times 3} & I_{3\times 3} \end{bmatrix} \\ Q(t) &= \begin{bmatrix} \sigma_v^2 I_{3\times 3} & 0_{3\times 3} \\ 0_{3\times 3} & \sigma_u^2 I_{3\times 3} \end{bmatrix}\end{aligned}$$

II Algorithm

2. Extended Kalman Filter

Table 7.1: Extended Kalman Filter for Attitude Estimation

Initialize	$\hat{\mathbf{q}}(t_0) = \hat{\mathbf{q}}_0, \quad \hat{\beta}(t_0) = \hat{\beta}_0$ $P(t_0) = P_0$
Gain	$K_k = P_k^- H_k^T(\hat{\mathbf{x}}_k^-) [H_k(\hat{\mathbf{x}}_k^-) P_k^- H_k^T(\hat{\mathbf{x}}_k^-) + R]^{-1}$ $H_k(\hat{\mathbf{x}}_k^-) = \begin{bmatrix} [A(\hat{\mathbf{q}}^-)\mathbf{r}_1 \times] & 0_{3 \times 3} \\ \vdots & \vdots \\ [A(\hat{\mathbf{q}}^-)\mathbf{r}_n \times] & 0_{3 \times 3} \end{bmatrix}_{t_k}$
Update	$P_k^+ = [I - K_k H_k(\hat{\mathbf{x}}_k^-)] P_k^-$ $\Delta\hat{\mathbf{x}}_k^+ = K_k [\tilde{\mathbf{y}}_k - \mathbf{h}_k(\hat{\mathbf{x}}_k^-)]$ $\Delta\hat{\mathbf{x}}_k^+ \equiv [\delta\hat{\alpha}_k^{+T} \quad \Delta\hat{\beta}_k^{+T}]^T$ $\mathbf{h}_k(\hat{\mathbf{x}}_k^-) = \begin{bmatrix} A(\hat{\mathbf{q}}^-)\mathbf{r}_1 \\ A(\hat{\mathbf{q}}^-)\mathbf{r}_2 \\ \vdots \\ A(\hat{\mathbf{q}}^-)\mathbf{r}_n \end{bmatrix}_{t_k}$ $\hat{\mathbf{q}}_k^+ = \hat{\mathbf{q}}_k^- + \frac{1}{2} \Xi(\hat{\mathbf{q}}_k^-) \delta\hat{\alpha}_k^+, \quad \text{re-normalize quaternion}$ $\hat{\beta}_k^+ = \hat{\beta}_k^- + \Delta\hat{\beta}_k^+$
Propagation	$\dot{\omega}(t) = \tilde{\omega}(t) - \hat{\beta}(t)$ $\dot{\hat{\mathbf{q}}}(t) = \frac{1}{2} \Xi(\hat{\mathbf{q}}(t)) \dot{\omega}(t)$ $\dot{P}(t) = F(t)P(t) + P(t)F^T(t) + G(t)Q(t)G^T(t)$ $F(t) = \begin{bmatrix} -[\hat{\omega}(t) \times] & -I_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} \end{bmatrix}, \quad G(t) = \begin{bmatrix} -I_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & I_{3 \times 3} \end{bmatrix}$

$$\dot{\mathbf{q}} = \frac{1}{2} \Omega(\omega) \mathbf{q}$$

II Algorithm

2. Extended Kalman Filter

Propagation

$$\hat{q}_{k+1}^- = \bar{\Omega}(\hat{\omega}_k^+) \hat{q}_k^-$$

$$\bar{\Omega}(\hat{\omega}_k^+) = \begin{bmatrix} \cos\left(\frac{1}{2}\|\hat{\omega}_k^+\|\Delta t\right)I_{3\times 3} - [\hat{\psi}_k^+ \times] & \hat{\psi}_k^+ \\ -\hat{\psi}_k^{+T} & \cos\left(\frac{1}{2}\|\hat{\omega}_k^+\|\Delta t\right) \end{bmatrix}$$

$$\hat{\psi}_k^+ = \frac{\sin\left(\frac{1}{2}\|\hat{\omega}_k^+\|\Delta t\right)\hat{\omega}_k^+}{\|\hat{\omega}_k^+\|}$$

$$P_{k+1}^- = \Phi_k P_k^+ \Phi_k^T + \gamma_k Q_k \gamma_k^T$$

Φ_k, γ_k (Transform matrix)

II Algorithm

2. Extended Kalman Filter

Gain

$$K_k = P_k^- H_k^T(\hat{x}_k^-) \left[H_k(\hat{x}_k^-) P_k^- H_k^T(\hat{x}_k^-) + R \right]^{-1}$$

$$H_k(\hat{x}_k^-) = \begin{bmatrix} \left[A(\hat{q}^-) r_1 \times \right] 0_{3 \times 3} \\ \left[A(\hat{q}^-) r_2 \times \right] 0_{3 \times 3} \end{bmatrix}_{t_k} \quad \begin{aligned} r_1 &= m_n \\ r_2 &= a_n \end{aligned}$$

II Algorithm

2. Extended Kalman Filter

Update

$$P_{k+1}^+ = \left[I - K_k H_K^T(\hat{x}_k^-) \right] P_{k+1}^-$$

$$\Delta \hat{x}_k^+ = K_k \left[y_k - h_k(\hat{x}_k^-) \right]$$

$$\Delta \hat{x}_k^+ = \begin{bmatrix} \delta \alpha_k^{+T} & \Delta \beta_k^{+T} \end{bmatrix}^T$$

$$h_k(\hat{x}_k^-) = \begin{bmatrix} A(\hat{q}^-)r_1 \\ A(\hat{q}^-)r_2 \end{bmatrix}_{t_k}$$

II Algorithm

2. Extended Kalman Filter

Update

$$\hat{q}_k^+ = \hat{q}_k^- + \frac{1}{2} \Xi(\hat{q}_k^-) \delta \alpha_k^+$$

re-normalize quaternion

$$\Xi(\hat{q}_k^-) = \begin{bmatrix} q_4 I_{3 \times 3} + [q_{1-3} \times] \\ -q_{1-3}^T \end{bmatrix}$$

$$\hat{\beta}_k^+ = \hat{\beta}_k^- + \Delta \hat{\beta}_k^+$$

II Algorithm

3. Norm Analysis

$$\begin{aligned} & \begin{bmatrix} \delta f_x & \delta f_y & \delta f_z \end{bmatrix} \begin{bmatrix} \delta f_x & \delta f_y & \delta f_z \end{bmatrix}^T = \delta f_x^2 + \delta f_y^2 + \delta f_z^2 \\ &= \left[H_k \delta \hat{x}_k^- + v \right]^T \left[H_k \delta \hat{x}_k^- + v \right] \\ &= \text{trace} \left(\left[H_k \delta \hat{x}_k^- + v \right] \left[H_k \delta \hat{x}_k^- + v \right]^T \right) \\ &= \text{trace} \left(H P H^T + R \right) \end{aligned}$$

if $\text{abs}(g - \text{norm}(f)) > \text{thr}$ then, Measurement update

$$\text{norm}(f) = \sqrt{f_x^2 + f_y^2 + f_z^2}$$

II Algorithm

3. Norm Analysis

$$H_1 = \begin{bmatrix} A(\hat{q}^-)r_2 \times & 0_{3 \times 3} \end{bmatrix}$$

$$\delta f_x^2 + \delta f_y^2 + \delta f_z^2 = \text{trace}\left(H_1 P H_1^T + R_a\right)$$

$$\sqrt{\delta f_x^2 + \delta f_y^2 + \delta f_z^2} = \sqrt{\text{trace}\left(H_1 P H_1^T + R_a\right)} = 1\sigma$$

III Simulation & Result

Simulation A

State : Quaternion

State Update : Gyro

Measurement : Accelerometer & Magnetometer

Filter : EKF

```
if i*dt>100 && i*dt<=150  
  
    b1 = Aq1*r1 + v_m;  
    b2 = Aq1*r2 + v_g + 3*ones(3,1);  
else  
    b1 = Aq1*r1 + v_m;  
    b2 = Aq1*r2 + v_g;  
  
end
```

III Simulation & Result

Simulation B

State : Quaternion

State Update : Gyro

Measurement : Accelerometer & Magnetometer

Filter : EKF + Norm Analysis

```
S = trace(H2*P*H2'+Ra);
thr = sqrt(S);

if abs(9.81 - norm(b2)) > thr
    % Gain
    Ra1 = Ra;
    R1 = [Rm zeros(3); zeros(3) Ra1];
    K = P*H1'*(H1*P*H1'+Rm)^-1;

    % Update
    P = (eye(length(P)) - K*H1)*P;
    dx = K*(b1-[Aq+r1]);
    qh = qh + 1/2*[qh(4)*eye(3)+skwsym(qh(1:3)); -qh(1:3)']*dx(1:3);
    qh = qh/norm(qh); %normalize quaternion
end
```

```
bh = bh + dx(4:6);
else
    b = b+1;
    % Gain
    K = P*H'*(H*P*H'+R)^-1;

    %Update
    P = (eye(length(P)) - K*H)*P;
    dx = K*(y-[Aq+r1;Aq+r2]);

    % Update
    qh = qh + 1/2*[qh(4)*eye(3)+skwsym(qh(1:3)); -qh(1:3)']*dx(1:3);
    qh = qh/norm(qh); %re-normalize quaternion
    bh = bh + dx(4:6);
end
```

III Simulation & Result

Simulation C

State : Quaternion

State Update : Gyro

Measurement : Accelerometer & Magnetometer

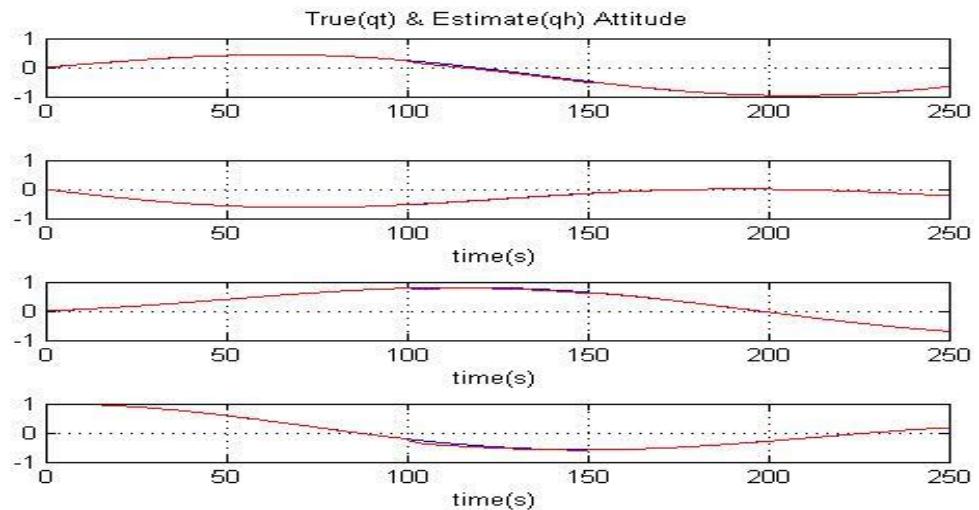
Filter : EKF + Norm Analysis + R Update

```
if abs(9.81 - norm(b2)) > thr  
  
    Ra1 = Ra + i*thr*ones(3);  
  
    R = [Rm zeros(3);zeros(3) Ra1];  
  
else  
  
    R = [Rm zeros(3); zeros(3) Ra];  
  
end
```

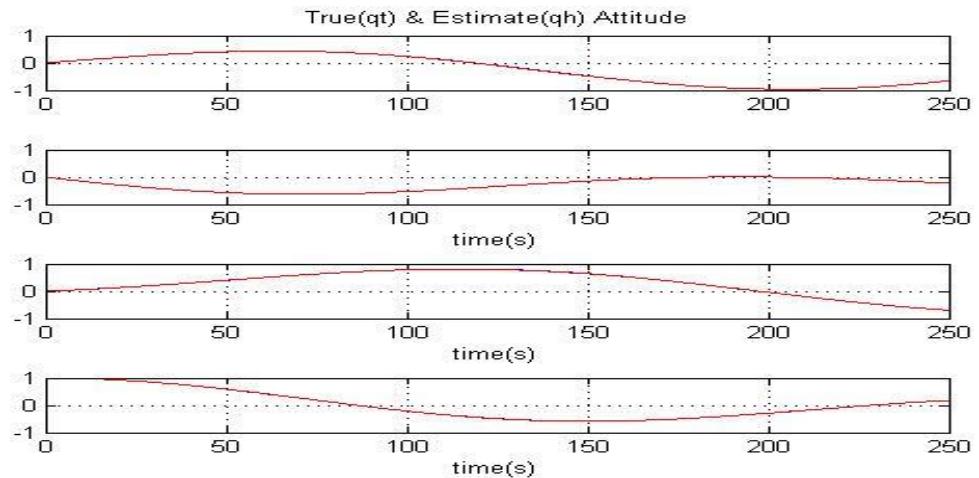
III Simulation & Result

Estimate & true
Attitude

Simulation A



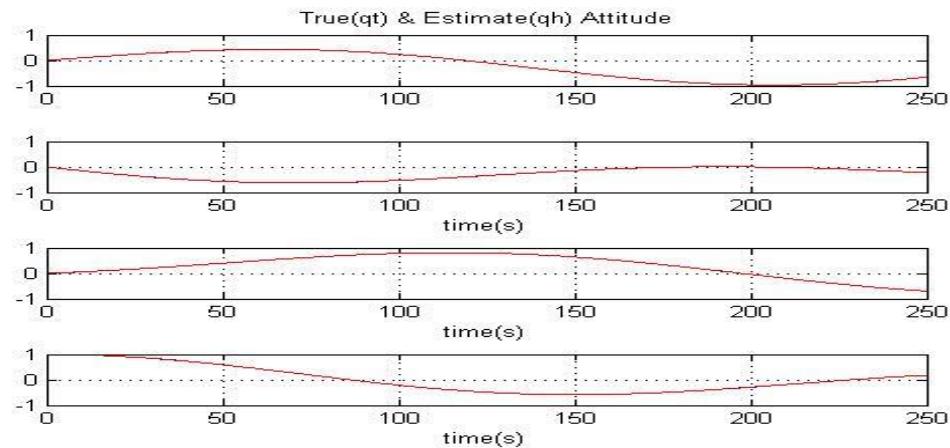
Simulation B



III Simulation & Result

Estimate & true
Attitude

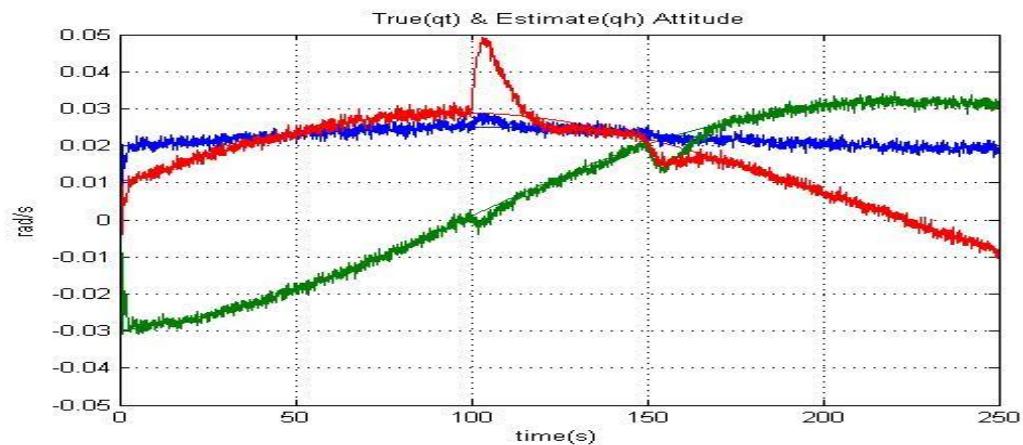
Simulation C



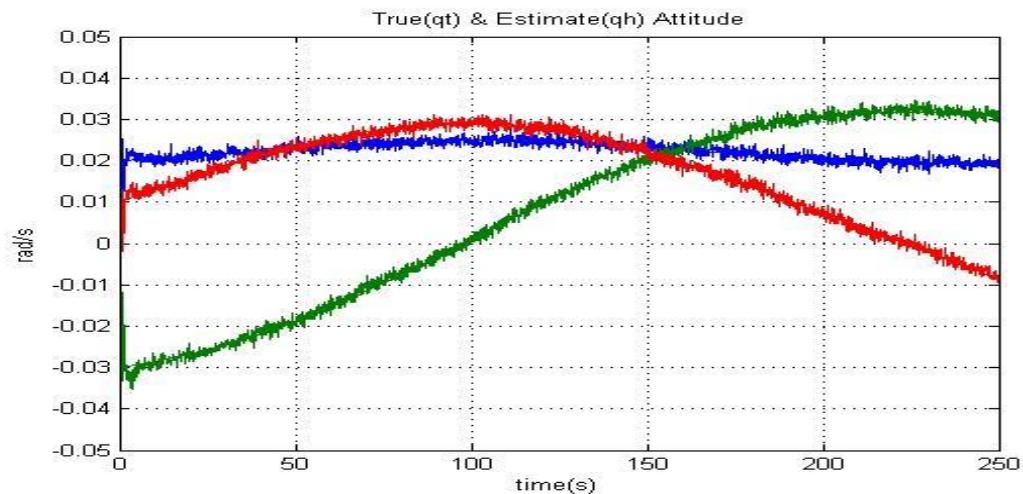
III Simulation & Result

Angular rate

Simulation A



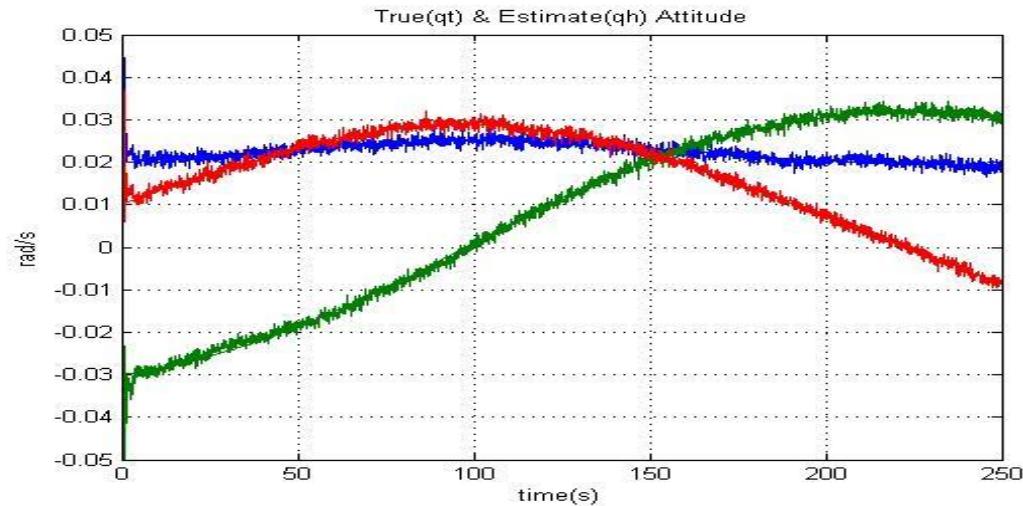
Simulation B



III Simulation & Result

Angular rate

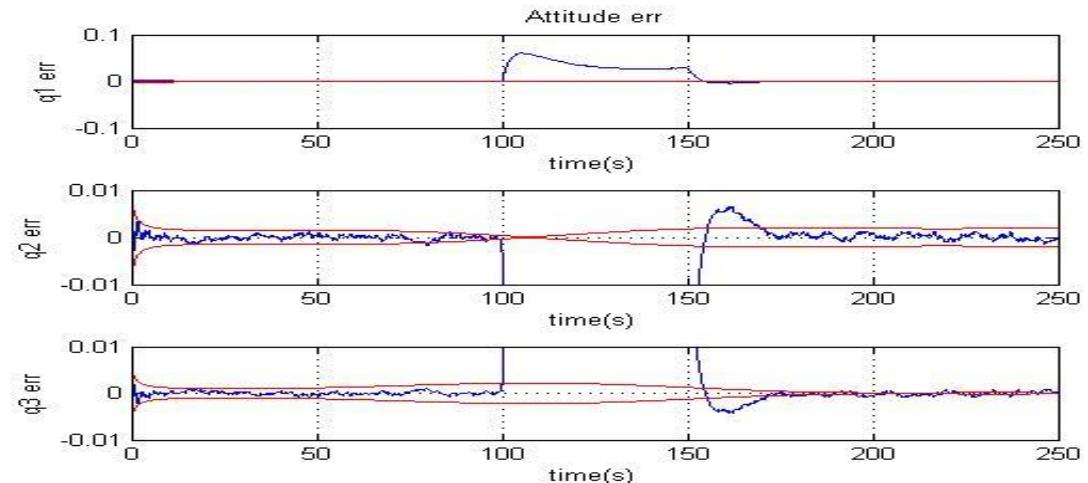
Simulation C



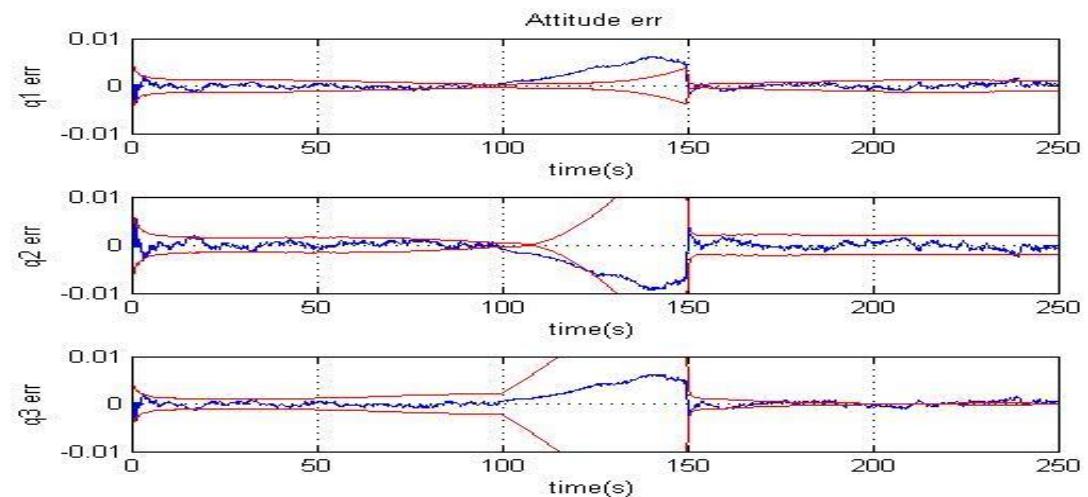
III Simulation & Result

Attitude Error

Simulation A

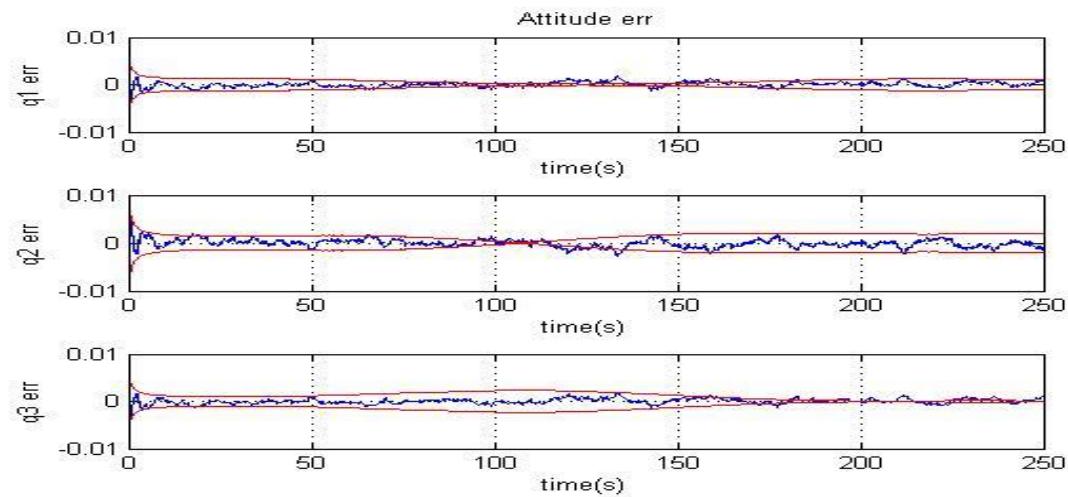


Simulation B



III Simulation & Result

Attitude Error

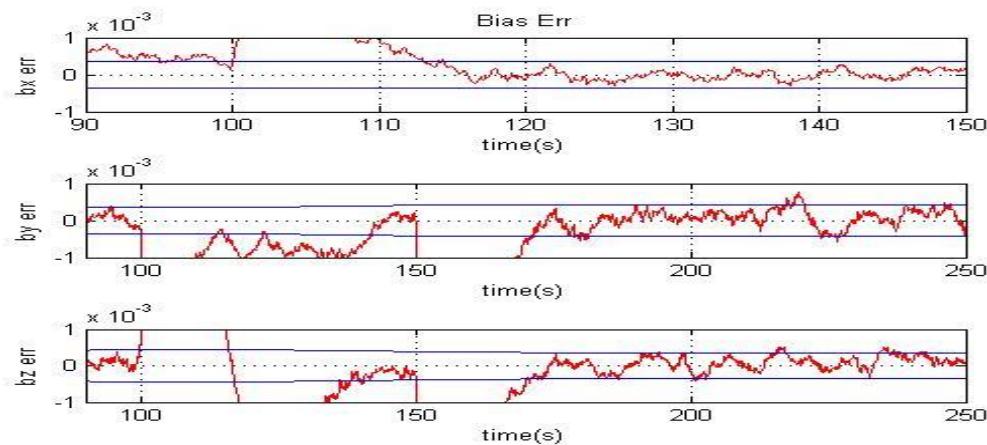


Simulation C

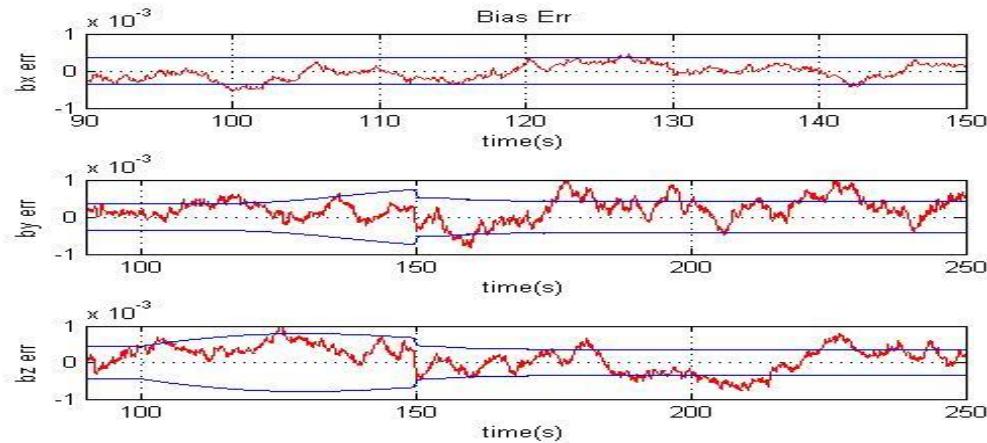
III Simulation & Result

Bias Error

Simulation A



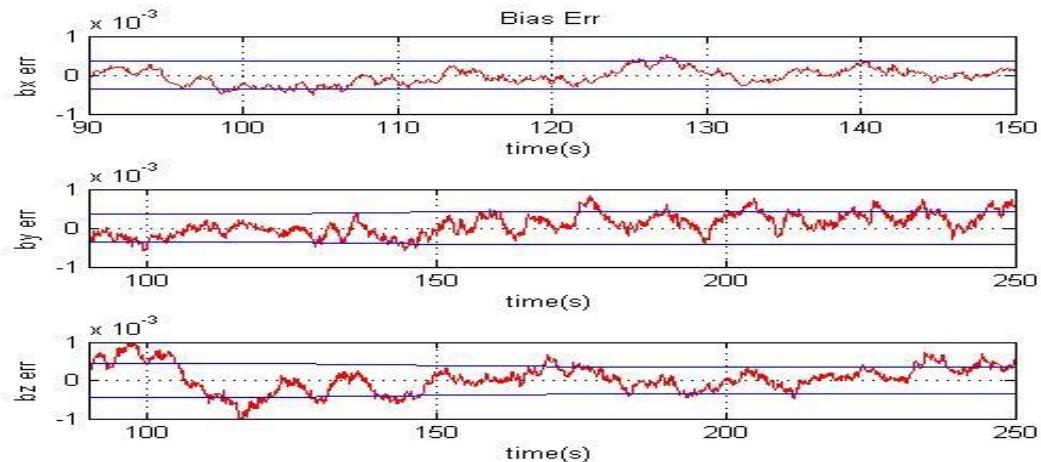
Simulation B



III Simulation & Result

Bias Error

Simulation C



III Simulation & Result

Result

1. 3가지 방법에 대한 시뮬레이션과 그에 따른 결과 도출을 성공함.
2. 기존의 방법보다 오차가 작음을 확인함

IV Future Plan

1. Unscented Kalman Filter
2. Particle Kalman Filter
3. Estimate Attitude

The background image shows a panoramic view of the St. Peter's Basilica in Rome, Italy, during sunset. The dome of the basilica is illuminated, casting a warm glow over the surrounding area. The Tiber River in the foreground reflects the golden light of the setting sun and the lights from the bridge and buildings along the riverbank. The sky is a mix of orange, yellow, and blue, with some clouds visible.

Q & A