

# IMS용 로봇의 고장진단기법에 관한 연구

## Fault Detection and Identification for a Robot used in Intelligent Manufacturing

이 상 길, 송 택 렬  
(Sang Gil Lee and Taek Lyul Song)

**Abstract** : To increase reliability and performance of an IMS(Intelligent Manufacturing System), fault tolerant control based on an accurate fault diagnosis is needed. In this paper, robot FDI(fault detection and identification) is proposed for IMS where the robot is controlled with state estimates of a nonlinear filter using a mathematical robot model. The Chi-square test and GLR(General likelihood ratio) test are applied for fault detection and fault size is estimated by a proposed bias filter. Performance of the proposed algorithm is tested by simulation for studies.

**Keywords** : fault, detection, identification, robot, EKF

### 1. 서론

항공기, 로봇매니퓰레이터, CNC(Computer Numerical Controller)등과 같은 많은 동적시스템에서는 시스템 파라미터 등과 같은 갑작스런 변화로 인한 고장을 감지, 대처하는 것이 필요하다. 고장이란 구동기, 센서 등과 같은 시스템 구성요소의 오동작이나 갑작스런 외란으로 인해 정상동작을 벗어나 원하는 작업을 수행하지 못하게 되는 허용될 수 없는 편차인 시스템의 오동작이라고 할 수 있다. 이러한 오동작은 플랜트전체의 예기치 못한 성능변화의 요인이 될 수 있다. 로봇, CNC공작기기 등으로 이루어진 지능생산시스템(Intelligent Manufacturing System)에서 로봇은 아주 중요한 구성요소이다. IMS에 필요한 각종의 요소기기들을 움직이기 위한 제어 시스템은 보다 고도화 되고 지능화 되어서 외부 환경의 변화와 외란에 대처할 수 있어야 한다. 이와 같이 외부 환경에 적용할 수 있는 제어 시스템의 필요성과 함께 IMS에서 대두되는 점은 제어 시스템의 오동작에 의해서 발생할 수 있는 문제점의 사전 감지 및 대처 능력이다. 갑작스런 고장으로 인한 제조시스템 전체의 중단은 제품생산과 운영시스템에 막대한 손실을 가져다주기 때문이다. 이에 허용되는 시간내에 시스템의 고장을 감지하여 시스템의 제어신호가 고장에 의해 과도한 반응을 나타내지 않도록 하는 고장의 조기감지, 판별과 회복을 함으로써 신뢰성, 생산성등을 높일 수 있다. 고장감지(fault detection)기능은 일정한 오차내에 어떤 측정치나 추정치가 있는지를 체크하는 것으로써 고장의 초기에 시스템에 경고신호를 보낼수 있고 고장판별(fault identification)기능은 고장부위의 검출, 고장의 정도를 알 수 있게 함으로써 시스템의 유지보수에 큰 도움을 줄 수 있으며 요소기기의 여분이 있을 경우 시스템을 재구성하고, 경로를 재설계하고

필요한 제어알고리즘을 구현하는데 필수적으로 소요된다. 고장의 유형은 센서, 구동기, 시스템의 세종류로 구분할 수 있으나[1], 이 논문에서는 로봇에 치명적일 수 있는 구동기의 고장에 한하여 논의하겠다.

기존의 고장진단기법(Fault Detection and Identification)에는 시스템 동력학의 수학적 모델에 근거한 방법과 하드웨어적 여분, 특수센서 등을 이용한 모델에 근거하지 않은 방법이 있다. 후자의 방법은 여분의 센서와 같은 장치를 설치하여 출력값을 비교함으로써 고장을 진단하는 것으로써 비용의 증가나 시스템 전체의 효율감소의 우려가 있다. 해석적 여분을 이용한 전자는 수학적 모델에 근거한 이론적 추정치에 대한 센서의 측정치가 비교되어진다. 이 방법은 대체적으로 정확하게 모델되어지고 공칭행위가 쉽게 특성지어지는 전자시스템이나 정보시스템에는 유용하지만 복잡한 시스템에서는 정확한 모델링이 곤란한 어려움이 있다. 이에 모델오차의 민감도에 대한 문제해결의 접근으로써 강인성의 입장에서 접근한 이론도 있다[2,3].

이 논문에서의 상태추정을 통한 고장진단은 로봇제어에 필요한 센서들에서의 측정치만을 이용하므로 추가적으로 요구되는 고가의 계측시스템 없이도 필요한 정보들을 이용할 수 있으므로 경비절감을 할 수 있는 잇점이 있다. 이 논문과 같이 수학적 모델에 근거한 FDI기법은 다양한 접근방법이 연구되어왔으며 크게 나누어 상태추정 접근법과 파라미터추정 접근법으로 생각할 수 있다. 상태추정 접근법에는 감지필터(detection filter), 칼만필터 또는 다중칼만필터를 이용한 이노베이션검사(innovation test)[4], 패리티공간방법(parity space approach), 전문가 시스템등이 있다.

이 논문에서 사용된 FDI 기법은 Extended Kalman Filter(EKF)를 이용한 이노베이션검사의 한 방법이며, 이 EKF 알고리즘은 로봇제어를 위해 feedback linearization에 사용되는 상태변수를 추정하는 동시에 바이어스형태의 고장이 발생했을 경우 이의 크기를 추정할 수 있는

접수일자 : 1998. 3. 16., 수정완료 : 1998. 7. 28.

이상길, 송택렬 : 한양대학교 제어계측공학과

※ 이 논문은 1995년도 한국학술진흥재단의 대학부설연구소 연구과제 연구비에 의하여 연구되었습니다.

residual을 제공한다. 제안된 바이어스추정용 필터에 의해 고장의 크기가 추정되며, 고장발생시간의 추정은 기존의 Chi-square 테스트와 GLR 테스트를 적용하여 이들의 성능을 비교분석 했다. 제시한 방법은 기존의 입력추정 알고리즘[5]이 갖는 계산의 복잡성과 과도한 정보교환량을 개선시킬 수 있다.

II. 시스템 모델링

IMS용 로봇은 회전관절이나 미끄럼관절로 연결된 링크들로 구성되어 있으며 각 링크는 구동기로 동작한다. 자유도보다 많은 구동기가 존재하는 여분의 경우 각 링크들은 다양한 운동을 할 수 있다. 그림 1은 2자유도 매니퓰레이터를 나타내는 것으로서 이론적 모델은 간단화될 수 있다. 여기에서 각 링크는 강체로 질량은 각 링크의 끝단에 존재하는 것으로 가정했다. 고장시 경로를 재설계 하기 위해 여유구동이 존재하도록 모델링하여야 하나 제시하는 고장진단 알고리즘의 성능을 분석하기 위해 이와 같은 간단한 매니퓰레이터의 모델을 고려했다.

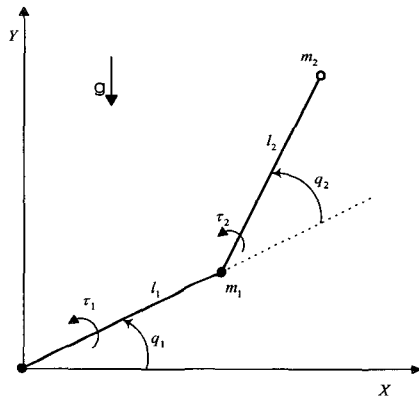


그림 1. 2 자유도 매니퓰레이터.  
Fig. 1. 2 DOF manipulator.

그림 1은 Young[6]과 Pei[7]에 의해 제안된 2자유도 매니퓰레이터를 나타낸다. 매니퓰레이터의 관절각의 각도, 각속도, 각가속도벡터를 각각  $q, \dot{q}, \ddot{q}$ 라하고 제어입력인 토크를  $\tau$ 라 하면 운동방정식은 아래와 같고 여기에 사용된 파라미터의 정의와 값들을 4장에 수록하였다.

$$M(q)\ddot{q} + D(q, \dot{q}) + G(q) = \tau \quad (1)$$

여기서,

$$q = (q_1, q_2)^T$$

$$M_{11} = (m_1 + m_2)l_1^2 + m_2 l_2^2 + 2m_2 l_1 l_2 \cos q_2 + I_1 + I_2$$

$$M_{12} = M_{21} = m_2 l_2^2 + m_2 l_1 l_2 \cos q_2 + I_2$$

$$M_{22} = m_2 l_2^2 + I_2$$

$$D_1 = -m_2 l_1 l_2 \sin q_2 \dot{q}_2^2 - 2m_2 l_1 l_2 \sin q_2 \dot{q}_1 \dot{q}_2$$

$$D_2 = m_2 l_1 l_2 \sin q_2 \dot{q}_1^2$$

$$G_1 = (m_1 + m_2)l_1 \cos q_1 g + m_2 l_2 \cos(q_1 + q_2)g + d_1$$

$$G_2 = m_2 l_2 \cos(q_1 + q_2)g + d_2$$

$M(q)$ 는 관성행렬로써 positive definite한 대칭행렬이다.

$D(q, \dot{q})$ 는 코리올리스와 원심력을 나타내는 벡터이고  $G(q)$ 는 중력을 나타내는 벡터이며  $d_i$ 는  $i$ 번째 링크의 외란이다.  $l_i, I_i, m_i$ 는 각각  $i$ 번째 링크의 길이, 관성모멘트, 질량을 나타내며  $g$ 는 중력가속도이다. 벡터  $q$ 는 첫 번째 관절각과 두 번째 관절각의 각도정보를 나타낸다.

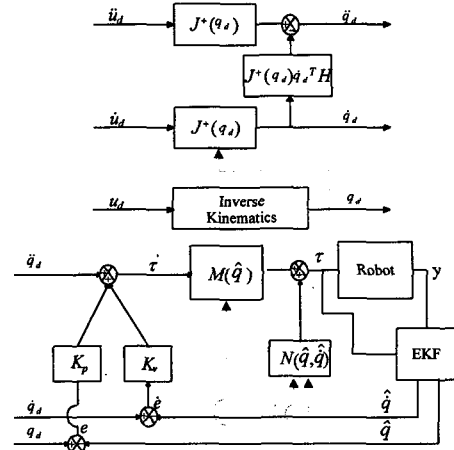


그림 2. 로봇 제어시스템.  
Fig. 2. Robot control system.

(1)은 MIMO 비선형시스템 방정식으로써 간략히  $M(q)\ddot{q} + N(q, \dot{q}) = \tau$ 으로 표현하면 로봇제어시스템은그림 2와 같이 나타낼수 있다. 여기서는 Cartesian 좌표계의 위치, 속도, 가속도인  $u, \dot{u}, \ddot{u}$ 를 Joint 좌표계인  $q, \dot{q}, \ddot{q}$ 로 변환시키는데 역기구학(Inverse Kinematics) [8]의 해석방법을 이용하였다. 매니퓰레이터의 kinematics에서 원하는 Cartesian 좌표계에서의 위치벡터  $u$ 와  $q$ 의 관계식  $u=f(q)$ 로부터 역기구학을 이용하면 위치벡터  $q$ 를 얻을 수 있고, 자코비안  $J$ 를  $\partial f/\partial q$ 로 정의하면  $\dot{u} = J\dot{q}$ 을 얻고 이의 역기구학을 취하면  $\dot{q} = J^+ \dot{u}$ 을 얻는다. 여기서  $J^+$ 는  $J$ 의 pseudoinverse를 나타낸다. 또한 헤시안행렬  $H$ 를  $\partial^2 J/\partial q^2$ 로 정의하면  $\ddot{u} = J\ddot{q} + \dot{q}^T H \dot{q}$ 을 얻고 이에 역기구학을 적용하면  $\ddot{q} = J^+ (\ddot{u} - \dot{q}^T H \dot{q})$ 을 얻는다.

로봇의 궤환제어에 사용되는 상태변수는 잡음이 실린 측정치로 얻어지는 관절각의 위치정보를 이용하여 필터로써 추정할 수 있으며, 그림2의 로봇제어시스템은 비선형이므로 비선형시스템에 적용 가능한 확장칼만필터(Extended Kalman Filter : EKF)[9]를 사용한다. 또한 로봇제어를 용이하게 하기 위해 비선형시스템을 선형시스템으로 변환하기위해 궤환선형화(feedback linearization) [10] 방법을 이용한다. 여기서는 궤환선형화의 하나인 계산토크제어를 사용한다. 로봇 동력학에서 비선형 항인  $M(q), N(q, \dot{q})$ 가 제거되도록 입력  $\tau$ 를  $\tau = \alpha\tau' + \beta$ 로 잡고  $\alpha = M(q), \beta = N(q, \dot{q})$ 가 되도록 하면 선형이 된다.

$$M(q)\ddot{q} + N(q, \dot{q}) = \alpha\tau' + \beta \quad (2)$$

$$\ddot{q} = \tau'$$

여기서 주어진 명령을 따르도록 하는 새로운 input  $\tau' = \ddot{q}_d + K_v(\dot{q}_d - \dot{q}) + K_p(q_d - q)$ 를 도입하여 상태공간 방정식으로 표현하면 아래와 같다.

$$\begin{aligned} \dot{x} &= Ax + \mu \\ y &= Hx \end{aligned} \quad (3)$$

여기서,

$$x = (q_1, q_2, \dot{q}_1, \dot{q}_2)^T$$

$$A = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -K_p & 0 & K_v & 0 \\ 0 & -K_p & 0 & K_v \end{pmatrix}$$

$$\mu = \begin{pmatrix} 0 \\ 0 \\ \ddot{q}_{1d} + K_v \dot{q}_{1d} + K_p q_{1d} \\ \ddot{q}_{2d} + K_v \dot{q}_{2d} + K_p q_{2d} \end{pmatrix}$$

$$y = (q_1, q_2)^T$$

$$H = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

$q_1 = x_1$  (첫번째 관절각의 위치)

$q_2 = x_2$  (두번째 관절각의 위치)

$\dot{q}_1 = x_3$  (첫번째 관절각의 속도)

$\dot{q}_2 = x_4$  (두번째 관절각의 속도)

(3)은 (1)의 비선형성이 완전히 제거된 상태공간방정식이다. 그러나 실제시스템에서는 센서잡음과 가용한 상태변수의 제한성 때문에 (2)의  $\alpha = M(q)$ ,  $\beta = N(q, \dot{q})$ 을 정확히 구현할 수가 없고 EKF로부터 얻어지는  $\hat{q}, \hat{\dot{q}}$ 를 사용하여  $\alpha, \beta$ 의 값을 정하여야 한다. 추정치가 잡음이 실린 위치정보  $q_1, q_2$  뿐이므로 필터를 사용하여 위치와 속도정보  $\hat{q}, \hat{\dot{q}}$ 를 추정할 수가 있다. 정상상태가 되었을 때 EKF부터 추정되는  $\hat{q}_1, \hat{q}_2, \hat{\dot{q}}_1, \hat{\dot{q}}_2$ 가 실제추정치와 근사한 것으로 생각하여 제어입력을  $\tau = \alpha\tau' + \beta$ 로 만들어 주며 여기서  $\alpha = M(\hat{q})$ ,  $\beta = N(\hat{q}, \hat{\dot{q}})$ 를 사용한다. 따라서 EKF의 상태변수 추정치를 활용한 폐환선형화된 로봇제어시스템의 동력학을 정리하면 아래와 같다.

$$\dot{\hat{q}} = M^{-1}(q)[M(\hat{q})\tau' + N(\hat{q}, \hat{\dot{q}}) - N(q, \dot{q})]$$

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{pmatrix} = \begin{pmatrix} x_3 \\ x_4 \\ (-g_1(x) + g_2(x)) \end{pmatrix} \quad (4)$$

$$\cong f(x, \tau)$$

여기서

$$g_1(x) = M^{-1}(x_2)N(x_1, x_2, x_3, x_4)$$

$$g_2(x) = M^{-1}(x_2)[M(\hat{x}_2)\tau' + N(\hat{x}_1, \hat{x}_2, \hat{x}_3, \hat{x}_4)]$$

$$\tau' = \begin{pmatrix} \ddot{q}_{1d} + K_v(\dot{q}_{1d} - \dot{\hat{x}}_3) + K_p(q_{1d} - \hat{x}_1) \\ \ddot{q}_{2d} + K_v(\dot{q}_{2d} - \dot{\hat{x}}_4) + K_p(q_{2d} - \hat{x}_2) \end{pmatrix}$$

$$\begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix}$$

### III. 제안된 FDI

#### 1. FDI를 위한 시스템 모델

상태변수의 추정치가 사용된 제어입력  $\tau$ 가 부가된 매니플레이터의 시스템 방정식인 (4)에 바이어스 형태의 구동기 고장과 공정 및 측정 잡음을 추가하여 FDI를 위한 시스템 동력학 모델을 설정하면 다음과 같이 표현할 수 있다.

$$\begin{aligned} \dot{x} &= f(x, \tau) + Bb\delta(t) + Gw(t) \\ y_k &= H_k x_k + v_k \end{aligned} \quad (5)$$

여기서

$x$  : 상태변수 벡터

$f(x, \tau)$  : 비선형함수

$B$  : 바이어스 방향벡터

$b$  : 바이어스 크기벡터

$G$  : 상태잡음행렬

$H_k$  : 출력 행렬

$w$  : 평균이 0이고 Power Spectral Density(PSD)가  $Q$ 인 공정 잡음을 나타내는 랜덤 프로세스

$v_k$  : 평균이 0이고 공분산이  $R$ 인 측정 잡음

그림 3은 로봇제어 시스템에 대해 이 논문이 제시하는 고장진단 방법의 흐름도를 도시한 것이다. 플랜트에 크기가 알려지지 않은 바이어스 형태로 고장을 표현했으며 시스템의 불확실한 동력학을 공정잡음으로 모델링하여 시스템 동력학 모델에 추가할 수 있다. EKF는 플랜트의 제어입력을 형성하기 위해 시스템 상태변수를 추정하는 동시에 고장진단을 위한 잔차를 생성한다. 제시한 FDI에서는 잔차의 확률론적 특성을 고려하여 고장발생 여부를 판단하는 고장결정 과정과 고장으로 판명시 고장의 정도를 추정하는 고장판별 과정을 거치게 된다.

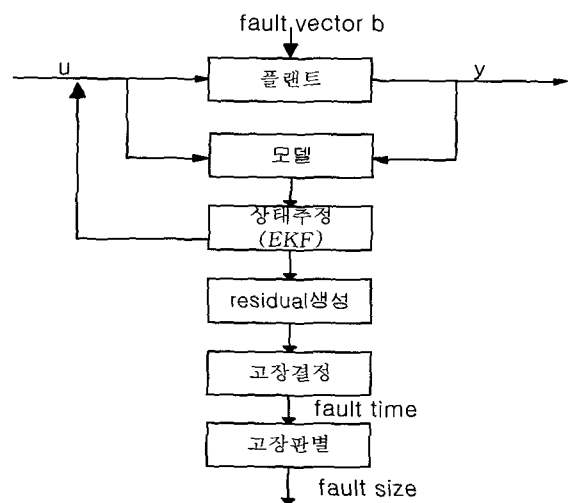


그림 3. FDI를 위한 흐름도.  
Fig. 3. Flow diagram for FDI.

#### 2. 상태추정

매니플레이터의 동력학 방정식 (1)은 폐환선형화 기법을 적용하여 선형화 시킬 수 있으며 비선형필터인 EKF

(Extended Kalman Filter)[9]를 사용하여 제어입력  $\tau$ 를 형성하는 상태변수의 추정치를 구한다. 이 논문에서의 EKF는 고장진단을 위한 잔차를 생성하며 동력학 모델에 고장을 고려하지 않은 공칭필터(nominal filter)의 역할을 수행한다. EKF의 알고리즘을 요약하면 아래와 같다.

전과 방정식 ( $t_{k-1} \leq t < t_k$ )

$$\text{상태변수 } \frac{d\bar{x}}{dt} = f(\bar{x}, \tau), \bar{x}(t_{k-1}) = \hat{x}_{k-1} \quad (6)$$

$$\text{공분산 } \frac{d\bar{P}}{dt} = F\bar{P} + \bar{P}F^T + GQG^T, \quad \bar{P}(t_{k-1}) = \hat{P}_{k-1} \quad (7)$$

$$F = \left. \frac{\partial f}{\partial x} \right|_{x=\bar{x}}$$

측정정보 갱신 방정식 ( $t = t_k$ )

$$\text{상태변수 } \hat{x}_k = \bar{x}_k + K_k(y_k - H_k \bar{x}_k) \quad (8)$$

$$\text{공분산 } \hat{P}_k = (I - K_k H_k) \bar{P}_k \quad (9)$$

$$K_k = \bar{P}_k H_k^T (H_k \bar{P}_k H_k^T + R_k)^{-1} \quad (10)$$

여기서

$\hat{\quad}$  : 시간  $t_k$ 에서의 추정치

$\bar{\quad}$  : 시간  $t_k$ 에서의 예측치.

### 3. 고장감지(fault detection)

고장감지는 크게 IFD(Instrument Fault Detection), AFD(Actuator Fault Detection), CFD(Component Fault Detection)으로 분류할 수 있으나, 이 논문에서는 구동기에 고장이 바이어스 형태로 나타날때의 고장감지 기법에 대하여 알아보기로 한다. 고장감지를 위해서는 우선 고장의 징조를 알기 위하여 잔차를 생성하고 실제로 고장이 발생했는지의 여부를 결정하기 위하여 생성된 잔차값을 검사하는 과정이 필요하며 잡음환경 하의 잔차의 경우에는 고장결정의 방법에 잔차의 통계학적 특성을 고려한 테스트를 한다.

#### 잔차생성(residual generation)

잔차생성은 다양한 프로세서에 따라 측정가능한 입·출력신호를 이용한 패러티기법, 상태추정치를 이용한 상태추정기법, 측정불가능한 파라미터의 경우 parameter identification[11]등의 기법이 있지만 이 논문에서는 EKF를 이용한 상태추정으로 잔차를 계산한다. 공정잡음과 측정잡음이 포함된 실제 출력값  $y_k$ 와  $y_k$ 의 추정치  $H_k \bar{x}_k$ 의 차이가 잔차이며 (8)의 상태변수의 갱신에 사용되는 항으로 이 값의 통계학적 테스트에 의해 고장결정을 한다.

#### 고장결정(fault decision)

이 논문에서는 EKF가 제어입력을 형성하는 상태추정에 이용되었으므로 EKF가 생성하는 잔차를 고장의 결정에 적용하였다. 고장결정을 위한 잔차를 테스트하는 기존의 방법으로는 Direct Parallel Testing, Multivariate Testing[12], Sequential Probability Ratio Test[13], Generalized Likelihood Ratio Test[14][15], Bayesian Methods[16] 등을 들 수 있으며 이 논문에서는 잔차의 Chi-square 분포[17]를 이용한 방법과 GLR 테스트를 고

장결정에 사용했다. 사용된 두가지 방법에 대해 다음 절에 설명하고 장·단점을 비교하며 시뮬레이션을 통한 성능분석결과를 4장에 수록하였다.

### 4. 고장판별(fault identification)

이 논문에서의 로봇은 각 조인트에서 잡음이 실린  $q_1, q_2$ 만 측정되고  $\dot{q}_1, \dot{q}_2$ 의 값은 측정되지 않는다고 가정했으므로 EKF를 동작시켜서 상태변수를 추정을 해야 한다. 추정결과로  $\hat{q}_1, \hat{q}_2, \hat{\dot{q}}_1, \hat{\dot{q}}_2$  값을 얻을 수 있으며 이 값들이 로봇의 제어시스템 구성에 사용된다. 로봇시스템은 (5)와 같이 비선형시스템 동력학을 만족하나 정상상태에서는 시스템의 상태변수와 EKF를 통한 이의 추정치가 같아진다고 가정하여 비선형시스템이 (3)과 같이 선형화된 것으로 간주한다. 여기에서 추가로 동력학의 불확실성을 공정잡음으로 모델링하고, 알려지지 않은 바이어스 형태로 고장을 모델링하여서 고장판별을 위한 시스템 동력학 모델로 사용하고자 하는 것이 아래의 (11)이다. 이 식은 EKF로부터의 잔차를 Chi-square분포와 GLR테스트에 적용하여 고장을 감지한 후, 고장의 크기를 판별할 수 있는 바이어스 추정 알고리즘의 개발에 사용된다.

아래와 같은 시스템의 크기가 알려지지 않은 입력  $b$ 의 크기를 추정하는 문제를 고려해 보자.

$$\begin{aligned} \dot{x}(t) &= A(t)x(t) + \mu(t) + Bb\delta(t-t_b) + Gw(t) \\ y_k &= H_k x_k + v_k \end{aligned} \quad (11)$$

여기서  $w(t)$ 와  $v_k$ 는 백색 가우시안 잡음으로서 각각 평균이 0이며 PSD가  $Q(t)$ , 공분산이  $R_k$ 이다. 시간  $t_b$ 에서 고장이 생기고 이후의 임의의 시간  $t_k$ 에서 필터의 잔차는

$$y_k - H_k \bar{x}_k = H_k e_k^b + (H_k e_k^f + v_k) \quad (12)$$

으로 표현될 수 있으며 추정오차  $e_k = x_k - \bar{x}_k$ 는  $b$ 가 없다고 고려한 필터의 추정오차  $e_k^f$ 와  $b$ 에 의한 오차  $e_k^b$ 로 구성되어 있다. 그러면  $e_k^b$ 는 필터의 이득을  $K_k$ 라 하면

$$e_k^b(t_k) = V(t_k)b \quad (13)$$

로 유도되고  $V(t_k)$ 는  $t = t_{k-1}$ 과  $t = t_k$  사이에서

$$\begin{aligned} \dot{V}(t) &= (A(t_{k-1}) - K_{k-1}H_{k-1})V(t) \\ V(t_{k-1}) &= V_{k-1} \end{aligned} \quad (14)$$

을 이용하여 구한다. 참고로  $V(t_b) = B$ 를 초기조건으로 한다. 위의 결과로부터 (12)를

$$y_k - H_k \bar{x}_k = H_k V_k b + v_k' \quad (15)$$

으로 표시할 수 있으며,  $v_k' \sim N(0, H_k \bar{P}_k H_k^T + R_k)$ 는 바이어스가 없다고 가정한 필터의 잔차를 나타내며 백색 잡음이다. 이 경우  $b$ 의 추정문제는

$$b_{k+1} = b_k \quad (16)$$

의 동력학 방정식과 식 (15)의 의사측정치(pseudomeasurement)로 구성된 선형시스템에 칼만필터의 일종인 바이어스 필터를 설계하는 문제로 귀속된다. 시간에 따라 천천히 변화하는 형태의 바이어스를 측정하기 위해 이

논문의 바이어스 추정필터에서  $b=0$  으로 표현한 바이어스의 동력학 모델을 Brownian motion process 바꾸어 사용하거나 band limited colored noise로 모델링하는 방법을 사용할 수도 있다.  $b$ 의 추정 알고리즘은

$$\begin{aligned} \hat{\delta}_{k+1} &= \hat{\delta}_k + P_{k+1}^b (H_{k+1} V_{k+1})^T \\ &\cdot [H_{k+1} \overline{P_{k+1}} H_{k+1}^T + R_{k+1}]^{-1} \\ &\cdot [y_{k+1} - H_{k+1} \overline{x_{k+1}} - H_{k+1} V_{k+1} \hat{\delta}_k] \\ (P_{k+1}^b)^{-1} &= (P_k^b)^{-1} + (H_{k+1} V_{k+1})^T \\ &\cdot [H_{k+1} \overline{P_{k+1}} H_{k+1}^T + R_{k+1}]^{-1} \cdot H_{k+1} V_{k+1} \end{aligned} \quad (17)$$

으로 요약된다. 또한 이 추정치를 [5,18]의 결과와 연계시켜 시스템상태변수  $x_k$ 의 추정치  $\hat{x}_k$ 를  $b$ 의 영향을 고려하여 수정하면

$$\hat{x}_k^c = \hat{x}_k + V_k \hat{\delta}_k \quad (18)$$

가 되고 수정된 공분산 행렬은

$$\hat{P}_k^c = \hat{P}_k + V_k \hat{P}_k^b V_k^T \quad (19)$$

가 되어야 한다. 다음에 고장결정을 위해서 이 논문에서 사용된 Chi-square분포와 GLR 테스트 방법을 설명하고자 한다.

Chi-square 분포를 이용한 방법

일반적으로  $n$ 개의 측정치  $y_i$ 가 평균이  $\bar{m}$ 이고 분산이  $\sigma^2$ 인 정규분포로부터 얻어졌다면 자유도  $n$ 인 Chi-square분포는 (20)과 같이 표현될 수 있으며 자유도가 감소할수록 분포도의 비대칭도가 감소한다.

$$\chi_n = \sum_{i=1}^n \frac{(y_i - \bar{m})^2}{\sigma^2} \quad (20)$$

잔차  $\nu_k$ 는  $\nu_k = y_k - H_k \overline{x_k}$ 으로 공분산이  $H_k \overline{P_k} H_k^T + R_k$ 인 백색 가우시안 잡음이고 고장이 일어나지 않을 때 평균은 0이다. 이 잔차의 2차함수 형태인 결정함수  $J_k$ 는 Chi-square분포를 나타내는 (20)을 이용하여 아래식으로 정할 수가 있으며 고장이 발생할 때  $J_k$ 값이 큰 영향을 받게 된다.

$$J_k = \nu_k^T S_k^{-1} \nu_k \quad (21)$$

여기서

$\nu$  : 잔차 벡터

$S_k$  :  $\nu$ 의 공분산 행렬

Chi-square 분포를 이용한 경우에는 결정함수  $J_k$ 가 문턱값  $\lambda$ 보다 클 경우 고장으로 판명하게 된다. 따라서 고장결정을 위한 수식은 아래와 같다.

$J_k \geq \lambda$  이면 고장

$J_k < \lambda$  이면 정상

여기서 고장결정을 위한 문턱값(threshold)  $\lambda$ 는 Chi-square 분포에서의 PFA(Probability False Alarm)  $\alpha$ 와 자유도(degree of freedom)  $n$ 으로부터 정해진다. 자유도  $n=2$ 이고  $\alpha$ 는 시스템의 성질에 따른 경험치를 주로 많이 사용하나 여기서는 고장일 확률  $\alpha$ 를 0.005로 정하여 시뮬레이션에 사용하였다.

GLR 테스트

이 방법은 고장의 크기가 알려져 있지 않은 경우에 사용한다. 분산이  $H_k \overline{P_k} H_k^T + R_k$ 인 잔차벡터  $\nu$ 에 대해 결정함수  $L$ 은 아래와 같이 유도된다.

$$(P_k^b)^{-1} = \sum_{j=\theta}^k (H_j V_j)^T (H_j \overline{P_j} H_j^T + R_j)^{-1} \cdot (H_j V_j) \quad (22)$$

$$d(k, \theta) = \sum_{j=\theta}^k (H_j V_j)^T (H_j \overline{P_j} H_j^T + R_j)^{-1} \cdot (y_j - H_j \overline{x_j}) \quad (23)$$

$$L(k, \theta) = d_k^T P_k^b d_k \quad (24)$$

여기서

$P_k^b$  : 바이어스 추정치의 error 공분산

$L(k, \theta)$  : 결정함수

$k$  : 현재시간

$\theta$  : 고장이 발생하는 시간

$k - \alpha < \theta \leq k - \beta$

$\alpha, \beta$  : 데이터 창 의 범위를 나타내는 상수

GLR 테스트에서 고장의 크기는 maximum likelihood estimate(MLE)을 사용하게 된다. 각각의 시간에서의 잔차  $\nu_1, \nu_2, \dots, \nu_k$ 와  $\theta = k - i$ 에서 고장이 발생했다는 가설  $H_i$ 에 기초하여 MLE  $\hat{\theta}$ 와  $\hat{\delta}$ 이 계산된다. MLE  $\hat{\theta}$ 는 결정함수  $L$ 을 최대화하는 시간  $\theta$ 의 값이다. 또한 계산된 결정함수  $L$ 이 문턱값  $\lambda$ 를 넘을 경우에만 시간  $\hat{\theta}$ 에서 고장이 생긴 것으로 간주하고 그렇지 않을 경우 정상으로 간주한다. 따라서 고장결정을 위한 수식은 아래와 같다.

$L(k, \hat{\theta}) \geq \lambda$  이면 고장

$L(k, \hat{\theta}) < \lambda$  이면 정상

GLR 테스트의 완전한 실행을 위해서는  $\theta=1, \dots, k$ 를 위한  $d(i, \theta)$ 와  $L(k, \theta)$ 를 계속적으로 계산해야 된다. 시간이 지날수록 점점 많아지는 다중필터를 구성해야 하는 문제점이 발생한다. 이런 문제점을 없애기 위해서 일정한 간격으로 시간을 구분하고 그 구간내에서 일정한 개수의 필터를 사용해야 할 필요가 있으며, (24)에 적용되는 역행렬을 구하는 문제와 직결된 시스템의 가관측성을 고려할 때 임의의 시간  $k$ 에서  $\theta$ 를  $k - \alpha < \theta \leq k - \beta$ 인 범위로 한정시켜야 한다. 이 논문에서는  $\alpha=10$ ,  $\beta=2$ 로 설정하였다. 창이 넓으면 고장감지와 판별에는 유용하지만 많은 양의 계산이 필요하기 때문에 적정선에서 선정하여야 한다.

Chi-square분포를 이용한 고장감지 알고리즘의 경우 시간  $t_k$ 에서 고장이 난 것으로 판단되면 (14)에서 행렬  $V_{k+1}$ 를 계산할 수 있다. (17)에서 (19)까지의 알고리즘을 이용하여 바이어스의 크기와 수정된 상태변수를 계산할 수 있다. GLR테스트를 이용한 알고리즘의 경우 결정함수가 문턱값을 넘었을 때의 MLE  $\hat{\theta}$ 가 고장시간이 되고  $\hat{\theta}$ 을 맞춘 가설하에서 계산된  $V_{k+1}$ 와 (17)에서 (19)까지의 알고리즘을 이용하여  $\hat{\delta}$ 와  $\hat{x}^c$ 를 추정한다.

IV. 시뮬레이션 및 결과

시뮬레이션에 사용된 그림 1의 매니플레이터의 파라미터 값들은

- 링크길이  $l_1 = 1m, l_2 = 0.8m$
- load inertia  $I_1 = 5kg\ m^2, I_2 = 5kg\ m^2$
- 질량  $m_1 = 0.5kg, m_2 = 3kg$
- 링크의 외란  $d_1 = 10N \cdot m, d_2 = 10N \cdot m$

초기 말단위치(end position)  $X = 1.3m, Y = 1.0m$ 이고, 각 링크에서의 조인트각은 역기구학에서 계산된다. 또한 매니플레이터가 원운동을 하도록 제어입력을 구성하였다.

고장진단모델 (5)에서 사용된 파라미터 값들은 바이어스벡터

$$b = (0rad/s, 0.5rad/s, 0rad/s^2, 0rad/s^2)^T$$

바이어스방향벡터  $B = I_4$

상태잡음행렬  $G = (0, 0, 1, 1)^T$

프로세스 노이즈벡터  $w$ 의 평균 0

$$PSD \text{는 } 0.25 (rad/s)^2 / s$$

측정치 노이즈벡터  $v$ 의 평균 0

$$\text{분산은 } 0.00005 rad^2$$

이다. 케환선형화에 사용되는 상태변수 추정치를 제공하는 EKF의 초기값들은

$$\text{상태의 초기값 } \hat{x}(0) = (0rad, 0rad, 0rad/s, 0rad/s)^T$$

$$\text{공분산의 초기값 } \hat{P}(0) = 5I_4$$

로 가정했으며 바이어스를 추정하는 바이어스 필터의 초기값으로

바이어스 초기값

$$\hat{b}(0) = (0rad, 0rad, 0rad/s, 0rad/s)^T$$

바이어스 공분산 초기값  $\hat{P}^b(0) = 0.0005I_4$ 를 사용했다.

이 논문에서는 RK-4방법을 사용하여 적분하였으며 적분간격은 0.001초로 했으며 문턱값  $\lambda = 10.6$ 으로 하였다. 고장감지 알고리즘은 EKF로 상태추정을 하여서 시스템이 정상상태일 때 작동된다. 잔차를 이용한 결정함수로 고장을 감지하게 되면 고장 판별 알고리즘이 작동된다. 이 알고리즘에서 잔차의 값은 EKF로 추정된 값이 사용된다. 그림 4와 그림 5는 Chi-square 분포를 이용하여 고장감지를 한 결과 추정된 바이어스의 발생시간과 크기를 나타낸 것이다.

이 그림들은 4초에  $q_2$ 에 바이어스가 생겼을 때 매니플레이터의 관절각  $q_1, q_2$ 가 간섭현상이 존재하는 시스템 동력학 방정식(4)에 의해 변화하는 실제값과 케환선형화에 필요한 상태변수를 추정하는 EKF에서 제공하는 잔차를 사용하여 Chi-square 테스트로서 고장을 감지한 후 바이어스 필터를 작동시켜 바이어스의 크기  $\hat{b}_k$ 을 추정하고 이를 (18)에 대입하여 수정된 상태변수의 추정치를 구한 결과와 비교한 것으로 바이어스 필터의 추정성능이 우수함을 알 수 있다. 그림 6과 그림 7은 EKF의 잔차값

의 변화를 나타낸 것이다. 시뮬레이션에서 바이어스는 두 번째 링크의 속도성분에 delta 함수로 주어져서 위치가 갑작스럽게 변화하는 것으로 가정했으므로 그림 6에서와 같이  $q_1$ 에서의 잔차값은 갑작스런 변화가 나타나지 않는다. 그러나, 그림 7에서  $q_2$ 의 잔차값은 갑작스런 변화가 있다. 그림 8은 GLR 테스트에 의하여 고장진단을 한 결과이다. 바이어스 크기 추정성능은 그림 5와 비슷하나 고장발생시간이 Chi-square는 4.09초에서 GLR은 4.04초에서 추정되어 GLR이 보다 정확함을 알 수 있었다.

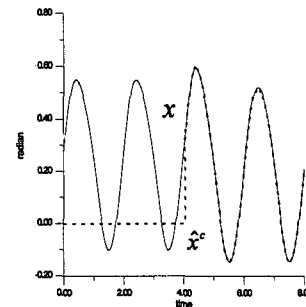


그림 4.  $q_1$ 에 작용한 고장 시간과 크기. (Chi-square분포를 이용).

Fig. 4. Performance of the FDI for  $q_1$  with Chi-square distribution.

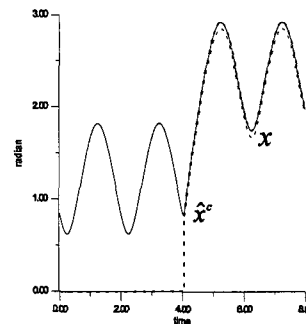


그림 5.  $q_2$ 에 작용한 고장 시간과 크기. (Chi-square분포를 이용).

Fig. 5. Performance of the FDI for  $q_2$  with Chi-square distribution.

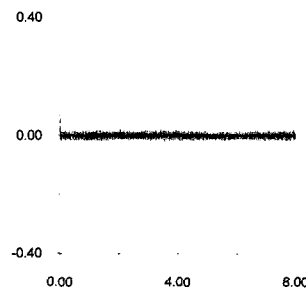


그림 6.  $q_1$ 의 Residual 값

Fig. 6. Residual for  $q_1$

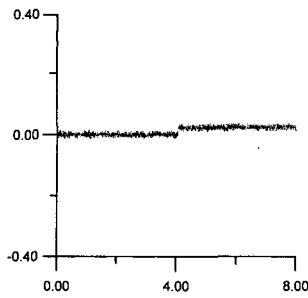


그림 7.  $q_2$  Residual 값.

Fig. 7. Residual for  $q_2$ .

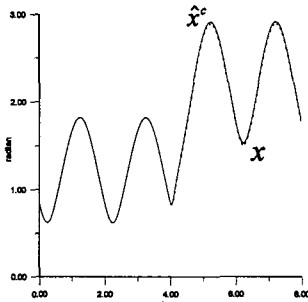


그림 8.  $q_2$ 에 작용한 고장 시간과 크기.  
(GLR 테스트를 이용)

Fig. 8. Performance of the FDI for  $q_2$  with GLR test.

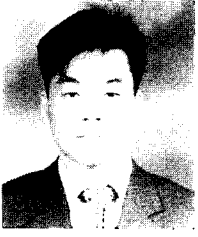
V. 결론

이 논문에서는 비선형시스템인 로봇을 궤환선형화 방법의 하나인 계산토크제어를 이용하여 선형화시켰다. 궤환선형화에 사용되는 상태들은 EKF를 이용하여 추정하였다. 바이어스 형태의 고장발생시 EKF의 잔차를 이용하고 Chi-square테스트와 GLR테스트를 이용하여 고장발생시간을 추정하며, 바이어스형태의 고장의 크기를 추정하기 위해 새로운 바이어스 추정필터 알고리즘을 제시하였다. 고장발생시간의 추정을 위한 Chi-square분포와 GLR테스트는 본 논문에서 제시한 추정필터와 결합하여 좋은 추정성능을 보였으며 GLR테스트는 Chi-square 방법에 비해 실행시간이 더 많이 소요되나 보다 정확한 고장발생시간을 추정할 수 있었다. 제시한 바이어스 추정필터를 사용한 고장진단은 기존의 입력추정 알고리즘이 갖는 계산의 복잡성과 과도한 정보교환의 단점은 개선된 것이다.

참고문헌

[1] P. M. Frank, "Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy-A survey and some new results," *Automatica*, vol. 26, no. 3, pp. 459-474, 1990.  
 [2] E. Y. Chow and A. S. Willsky, "Analytical redundancy and the design of robust failure detection systems," *IEEE Trans. on Automatic Control*, vol. AC-29, no. 7, pp. 603-614, Jul., 1984.

[3] E. Wahnon, A. Benveniste, L. El Ghaoui and R. Nikoukhah, "An optimum robust approach to statistical failure detection and identification," *Proc. of the 30th Conference on Decision and Control*, Dec., 1991.  
 [4] R. K. Mehra and I. Peshon, "An innovations approach to fault detection and diagnosis in dynamic system," *Automatica*, vol. 7, no. 635-640, 1971.  
 [5] P. L. Bogler, "Tracking a maneuvering target using input estimation," *IEEE Trans. on Aerospace and Electronic Systems*, vol. AES-23, no. 3, pp. 298-310, 1987.  
 [6] K. D. Young, "Controller design for a manipulator using theory of variable structure system," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-8, pp. 101-109, Feb., 1987.  
 [7] H. L. Pei and Q. J. Zhou, "Variable structure control of linearizable systems with applications to robot manipulators," *Proc. of the 1991 IEEE International Conference on Robotics and Automation*, Apr., 1991.  
 [8] M. Ziegler, "Geometric approach in solving inverse kinematics of PUMA robots," *IEEE Trans. on Aero. and Elec. Sys.*, vol. AES-20, no. 6, Nov., 1984.  
 [9] M. S. Grewal and A. P. Andrews, *Kalman Filtering Theory and Practice*, Prentice Hall.  
 [10] J. E. Slotine and W. Li, *Applied Nonlinear Control*, Prentice Hall, 1991.  
 [11] R. Isermann, "Process fault detection based on modelling and estimation methods-A survey," *Automatica*, vol. 20, no. 4, pp. 387-404, Pergamon Press, 1984.  
 [12] J. C. Hancock and P. A. Wintz, *Signal Detection Theory*, McGraw Hill, New York, 1966.  
 [13] G. B. Wetherill, 1975, *Sequential Methods in Statics*, Chapman and Hall, London, 1975  
 [14] A. S. Willsky, "A survey of design methods for failure detection in dynamic systems," *Automatica*, vol. 12, pp. 601-611, 1976.  
 [15] A. W. Willsky and H. L. Jones, "A generalized likelihood ratio approach to the detection and estimation of jumps in linear Systems," *IEEE Trans. on Automatic Control*, vol. AC-21, no. 1, pp. 108-112, Feb., 1976.  
 [16] I. Miller and J. E. Freund, 1985, *Probability and Statistics for Engineering* Englewood Cliffs, N. J., 1983.  
 [17] B. E. Wampold and C. J. Drew, *Theory and Application of Statistics*, McGraw-Hill International Editions.  
 [18] B. Friedland, "Treatment of bias in recursive filtering," *IEEE Trans. on Automatic Control*, vol. AC-14, no. 4, pp. 359-367, Jan., 1969.



**이 상 길**

1996년 숭실대학교 기계과 졸업.  
1998년 한양대학교 제어계측공학과  
석사. 관심분야는 FDI, Estimation,  
Control Theory 등.

**송 택 렬**

제어·자동화·시스템공학회 논문지 제4권, 제1호, 참조.