

# 네트워크 기반 이산 시간 제어 시스템의 최대 허용 지연 한계 및 실시간 스케줄링 기법에 관한 연구

## Maximum Allowable Delay Bounds and Real-time Scheduling Method of Networked Discrete-time Control Systems

김 동 성\*, 최 동 혁  
(Dong-Sung Kim and Dong-Hyuk Choi)

**Abstract** : This paper proposes a new method to obtain a maximum allowable delay bound for a scheduling of networked discrete control systems and event-based scheduling method. The proposed method is formulated in terms of linear matrix inequalities and can give a much less conservative delay bound than the existing methods. A network scheduling method is presented based on the delay obtained through the proposed method, and it can adjust the sampling period to allocate same utilization to each control loop. The presented method can handle three types of data (sporadic emergency data, periodic data and non real-time message) and guarantees real-time transmission of periodic and sporadic emergency data using modified EDF scheduling method.

**Keywords** : networked control systems, maximum allowable delay bound, linear matrix inequalities, scheduling method

### I. 서론

분산 제어 시스템에서는 네트워크에 의해 피드백 제어 루프가 형성되어 있다. 일반적으로 네트워크를 가진 분산 제어 시스템은 네트워크 기반 제어 시스템(NCS: Networked Control Systems)이라 부른다. NCS는 네트워크 매체의 공유로 인해 일정하지 않은 길이를 가진 다양한 지연들이 발생한다. 이러한 지연을 네트워크 유도 지연(network-induced delay)이라 부른다. 네트워크 유도 지연은 메시지의 전송 시간과 오버헤드 시간에 따라 폭 넓게 변화한다. NCS에서 네트워크는 실시간성 주기(periodic) 데이터, 산발성(sporadic) 데이터, 그리고 비실시간 메시지 등 세 종류의 데이터 타입을 포함한다. 매체를 통한 전송 시간은 네트워크 프로토콜에 크게 의존하며, 특히 네트워크의 데이터 연결 계층 프로토콜과 데이터 길이에 큰 영향을 받는다. 그러므로 이러한 네트워크 유도 지연을 네트워크 기반 제어시스템의 성능 저하 없이 작게 만드는 방법이 필요하며, 이러한 방법을 NCS를 위한 네트워크 스케줄링 기법이라 부른다.

네트워크 기반 제어 시스템에서는 샘플링 주기 내에서 샘플링 된 데이터를 전송하는 것과 동시에 제어 시스템의 안정성을 보장하는 것이 중요하다. 대부분의 제어 시스템에서 짧은 샘플링 주기가 바람직하지만, NCS에서는 성능이 나빠짐에도 불구하고 시스템의 안정성을 보장 받을 수 있는 경계선까지는 샘플링 주기가 길어질 수 있다. 이러한 샘플링 주기를 최대 허용 지연 한계(MADB: Maximum Allowable Delay Bound)라 부른다. 그러므로 NCS의 안정성

을 보장하는 MADB를 찾아내고, 네트워크 유도 지연을 MADB보다 작도록 제한하는 적절한 네트워크 스케줄링 기법이 필요하다. 네트워크 스케줄링 기법은 실시간성 주기적인 데이터와 산발성 긴급 데이터의 실시간 전송을 보장하는 동시에 비 실시간성 메시지의 전송을 보장할 수 있어야 한다.

MADB는 제어 시스템의 안정조건으로부터 얻게 된다. NCS의 안정성에 대한 결과들이 발표되었지만, 이 연구들은 주어진 지연 한계에서 NCS의 안정조건을 얻는 것에 대하여만 고려되었다[1-4]. 또한 일반 제어 시스템에서 안정성을 보장하기 위한 MADB에 대한 연구결과도 발표되었다[5-7]. 상기 논문들에서 계산된 MADB는 리카티(riccati) 방정식을 이용한 비현실적인 작은 지연 경계(conservative delay bounds) 값을 산출한다.

필드버스 네트워크를 포함한 NCS에서의 스케줄링 기법에 관한 논문들이 발표 되었지만, NCS의 해석에 있어 중요한 요소인 MADB의 산출과 제어기 지연들을 고려하지 않았다[8-12]. [11]에서는 필드버스 네트워크에서 주기적 데이터와 비주기적 데이터를 위하여 RM(Rate Monotonic) 스케줄링 알고리즘을 변경한 동적 스케줄링 알고리즘이 발표되었다. [12]에서는 주기적인 필드버스 통신을 위한 알고리즘이 발표 되었으나 비주기적 데이터에 대한 고려는 되어 있지 않았다. [13]에서는 네트워크의 대역폭 할당을 고려하고 센서의 샘플링 주기를 결정할 수 있는 스케줄링 알고리즘이 발표 되었다. 그러나 이 논문은 MADB를 고려하지 않은 NCS에 대한 연구이다. 또한 NCS의 동적 스케줄링을 위한 알고리즘들이 제안되었으나, MADB의 산출과 샘플링 주기의 설정과 같은 NCS의 특성을 고려하지 않았다[14,15].

[16]에서 다중 입출력기반으로 세 가지의 데이터 타입을 고려한 네트워크 스케줄링 기법이 발표 되었다. 이 논문에

\* 책임저자(Corresponding Author)

논문접수 : 2006. 3. 1., 채택확정 : 2006. 5. 8.

김동성, 최동혁 : 금오공과대학교 전자공학부

(dskim@kumoh.ac.kr/nemamdero@kumoh.ac.kr)

※ 본 논문은 금오공대 교내 학술연구비에서 지원하여 연구하였음.

서 리카티(riccati) 방정식을 이용하여 구한 MADB는 실제 시스템에 적용하기에는 너무 작고, 제안된 스케줄링 기법은 기본적인 단순한 방법을 사용했다. [17-20]등의 논문들에서는 MADB의 계산 방법과 NCS의 안정성 분석을 발표했다. 하지만, 이러한 결과들은 실제로 사용하기에는 비현실적이고, 개선되어야 할 문제들이 여전히 남아 있었다. 이를 해결하기 위해서는 NCS의 안정성을 위한 현실적인 MADB의 산출과 실시간 및 비실시간 형태의 데이터 타입을 고려한 체계적인 스케줄링 기법에 대한 연구가 필요했다.

LMI (Linear Matrix Inequality)를 이용한 연속시간 기반의 NCS의 MADB 산출방법이 제안되었다[21]. 제안된 기법은 기존의 기법에 비해 보다 현실적인 NCS 지연 경계를 제공한다. 하지만 이산시간 모델을 이용한 안정성 조건과 MADB를 구하는 방법에 대한 연구 및 실시간 주기적인 데이터, 산발성 긴급, 비실시간 메시지 등의 데이터가 혼재된 경우에 더욱 효과적인 네트워크 스케줄링 기법에 대한 연구가 계속 필요하다. 본 논문에서는 LMI 기법을 이용한 이산시간 기반의 MADB 산출기법을 제안하고 이를 이용한 샘플링 주기 결정 방법과 산발성 긴급데이터의 실시간 전송에 관한 스케줄링 기법을 NCS 시스템에 적용 하였다.

본 논문에서 사용된 스케줄링 기법은 산출된 MADB와 전송패킷의 수를 고려하여 각 루프의 채널 사용률을 균등하게 보장할 수 있도록 샘플링 주기를 결정한다. 그리고 산발성 긴급 데이터의 실시간 처리를 위한 변형된 EDF 스케줄링 기법을 사용하여 실시간성 주기적인 데이터와 산발성 긴급 데이터의 실시간 처리를 보장 한다.

본 논문은 다음과 같이 구성되어 있다. 2장에서는 NCS 모델을 설명하고, NCS의 안정성을 위한 MADB를 LMI 형태로 제시한다. 3장에서는 NCS에서 대역폭을 할당하고 샘플링 주기를 결정하는 네트워크 스케줄링 기법을 제안한다. 4장에서는 본 논문에서 제안된 기법을 사용한 시뮬레이션 결과를 보여주고, 마지막 5장에서 결론을 기술하였다.

**II. 네트워크 기반 제어 루프에서 안정성을 위한 MADB**

네트워크 기반 제어 루프는 제어기, 센서, 구동기로 구성된다. 이들 장치들은 일반적으로 필드버스와 같은 네트워크 매체를 공유한다.

센서 노드들이 프로세스의 제어기 명령으로 센서 데이터를 검출하는 순간부터 구동기들이 제어기로 전송된 데이터를 출력하는 순간까지가 최대 허용 지연 기간(MADB)으로 정의된다. 만약 j번째 제어 루프에서 샘플링 주기가 주어진 MADB를 초과하면, 전체 시스템의 안정성은 보장받을 수 없다. 그러므로 주어진 프로세스와 제어기로부터 MADB를 구해내는 방법이 필요하다. 그림 1에서 NCS에서 실시간 및 비실시간 데이터의 처리와 네트워크를 통한 전송 과정에 대한 예를 보인다.

그림 2는 다중 입출력으로 확장한 제어 루프를 나타내고 있다. 제어 루프는 여러 개의 센서의 입력과 여러 개의 구동기의 출력으로 구성된다. 이러한 시스템은 여러 개의 단일 입출력 제어 루프로 나누어 고려할 수 있다. 그림 3의 타이밍 다이어그램은 샘플링 기간 동안에 제어기 노드에서

의 신호, 구동기 노드에서의 신호, 네트워크 유도 지연 등의 처리 순서를 보여준다.

**1. 이산 시간(discrete-time) NCS의 MADB**

다중 입출력 제어 루프는 여러 개의 단일 입출력 제어 루프로 나누어 고려할 수 있다. 네트워크 기반의 단일 제어 루프에서의 프로세스는 다음과 같은 이산시간 상태방정식으로 표현할 수 있다.

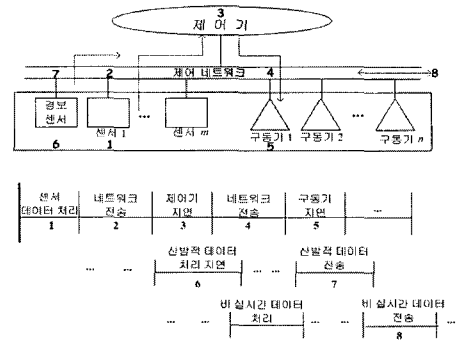


그림 1. 네트워크 기반 제어 루프의 다이어그램.

Fig. 1. Diagram of control loops using a network.

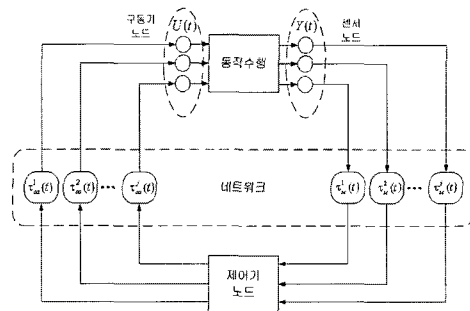


그림 2. 여러 개의 센서와 구동기가 있는 다중 입출력 네트워크 기반 제어 루프 [22].

Fig. 2. MIMO Networked control loops with sensors and actuators [22].

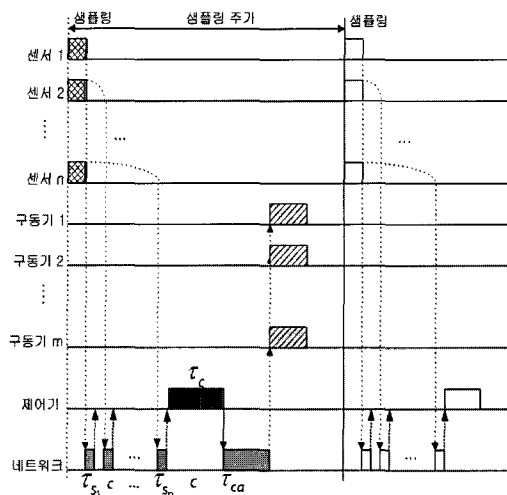


그림 3. NCS에서 신호의 타이밍 선도 [21].

Fig. 3. Timing of signals in the NCS.

$$\begin{aligned} x_p(k+1) &= A_p x(k) + B_p \bar{u}(k), \\ y(k) &= C_p x(k). \end{aligned} \quad (1)$$

( $\bar{u}(k) \in R^{N_A}$ ,  $x_p(k) \in R^{N_P}$ ,  $y(k) \in R^{N_S}$ )

$N_A$ ,  $N_P$ ,  $N_S$  는 각각 고려된 단일 제어 루프의 구동기, 공정, 센서의 수를 나타낸다. 본 논문의 제어 루프 내에 있는 제어기는 다음과 같이 표현할 수 있다.

$$\begin{aligned} x_c(k+1) &= A_c x_c(k) + B_c \bar{y}(k), \\ u(k) &= C_c x_c(k) + D_c \bar{y}(k). \end{aligned} \quad (2)$$

( $u(k) \in R^{N_A}$ ,  $x_c(k) \in R^{N_C}$ ,  $\bar{y}(k) \in R^{N_S}$ )

$N_C$ 는 고려된 단일 제어루프의 제어기의 수를 나타낸다.  $y(k)$ 는 플랜트의 출력이고,  $\bar{y}(k)$ 는 제어기의 입력이다.  $u(k)$ 는 제어기의 출력이고,  $\bar{u}(k)$ 는 플랜트의 입력이다. 센서에서 제어기로 전송하는 과정의 시스템 지연을  $\tau_x$  로 하고, 이산화 스텝 지연시간을  $\tau_{sc}$ 로 나타내면,  $\tau_x$ 는  $0 \leq \tau_x \leq \tau_{sc}$ 를 만족한다. 마찬가지로 제어기에서 플랜트로 전송하는 과정의 이산 시스템 지연을  $\tau_\phi$ 라 하고, 이산화 스텝 지연시간을  $\tau_\phi$ 로 나타내면,  $0 \leq \tau_\phi \leq \tau_{cp}$ 를 만족한다.  $\tau_\phi$ 는 제어기에서 구동기까지의 시간외에도 제어기 지연시간을 포함한다.

이산시간 제어 루프에서 통신 지연시간은 다음과 같이 모델링 된다.

$$\begin{aligned} \bar{y}(k) &= y(k - \tau_{sc}), \\ \bar{u}(k) &= u(k - \tau_{cp}). \end{aligned} \quad (3)$$

$\tau_{cp} = \tau_{ca} + \tau_{other}$

여기서,  $\tau_{ca}$ 는 제어기에서 구동기까지의 지연시간을 나타낸다. (1)~(3)을 이용하여, 이산시간 NCS를 모델링하면 (4)와 같다.

$$\begin{aligned} x_p(k+1) &= A_p x_p(k) + B_p u(k - \tau_{cp}) \\ &= A_p x_p(k) + B_p [C_c x_c(k - \tau_{cp}) + D_c \bar{y}(k - \tau_{cp})] \\ &= A_p x_p(k) + B_p [C_c x_c(k - \tau_{cp}) + D_c y(k - \tau_{cp} - \tau_{sc})] \\ &= A_p x_p(k) + B_p [C_c x_c(k - \tau_{cp}) + D_c C_p x_p(k - \tau_{cp} - \tau_{sc})]. \end{aligned} \quad (4)$$

마찬가지로

$$\begin{aligned} x_c(k+1) &= A_c x_c(k) + B_c y(k - \tau_{sc}) \\ &= A_c x_c(k) + B_c C_p x_p(k - \tau_{sc}). \end{aligned} \quad (5)$$

$x(k) = [x_p(k) \ x_c(k)]^T$  로 하면,

$$\begin{aligned} x(k+1) &= \begin{bmatrix} x_p(k+1) \\ x_c(k+1) \end{bmatrix} \\ &= \begin{bmatrix} A_p & 0 \\ 0 & A_c \end{bmatrix} \begin{bmatrix} x_p(k) \\ x_c(k) \end{bmatrix} \end{aligned} \quad (6)$$

$$\begin{aligned} &+ \begin{bmatrix} 0 & B_p C_c \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_p(k - \tau_{cp}) \\ x_c(k - \tau_{cp}) \end{bmatrix} \\ &+ \begin{bmatrix} 0 & 0 \\ B_p C_p & 0 \end{bmatrix} \begin{bmatrix} x_p(k - \tau_{sc}) \\ x_c(k - \tau_{sc}) \end{bmatrix} \\ &+ \begin{bmatrix} B_p D_c C_p & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_p(k - \tau_{cp} - \tau_{sc}) \\ x_c(k - \tau_{cp} - \tau_{sc}) \end{bmatrix} \\ &= Ax(k) + A_1 x(k - \tau_1) \\ &\quad + A_2 x(k - \tau_2) + A_3 x(k - \tau_3). \end{aligned}$$

와 같이 쓸 수 있다. 여기에서 각 행렬과 지연시간은 다음과 같다.

$$\begin{aligned} A &= \begin{bmatrix} A_p & 0 \\ 0 & A_c \end{bmatrix}, \quad A_1 = \begin{bmatrix} 0 & B_p C_c \\ 0 & 0 \end{bmatrix}, \\ A_2 &= \begin{bmatrix} 0 & 0 \\ B_p C_p & 0 \end{bmatrix}, \quad A_3 = \begin{bmatrix} B_p D_c C_p & 0 \\ 0 & 0 \end{bmatrix}, \\ \tau_1 &= \tau_{cp}, \quad \tau_2 = \tau_{sc}, \quad \tau_3 = \tau_{cp} + \tau_{sc}. \end{aligned}$$

이산시간 NCS에서 각 제어 루프는 세 가지 형태의 네트워크 지연을 사용하여 (6)과 같이 나타낼 수 있다.

다중 상태 지연의 경우에는 (7)과 같은 시스템 식으로 바꿀 수 있다.

$$\begin{aligned} x(k+1) &= Ax(k) + \sum_{i=1}^m A_i x(k - \tau_i) \\ x(k) &= \phi(k), \quad k \in [-\bar{\tau}, 0]. \end{aligned} \quad (7)$$

여기에서  $x(k) \in R^n$  이고, 이산 시간 시스템 상태이다.  $\tau_i > 0$  은 시스템의 지연 단계 이다.  $\phi(k)$ 는 초기 값이고,  $A, A_i$ 는 적당한 차수를 가진 실수 행렬이다.  $\bar{\tau}$ 는  $\tau_i$ 의 상한선이다.

본 논문에서 이산 시간 NCS의 안정성을 보장하는 MADB를 얻는데 있어서, (8)의 두 벡터의 내적에 대한 상한선은 중요한 역할을 한다.

$$-2a^T N b \leq \inf_{X, Y, Z} \begin{bmatrix} X & Y - N \\ Y^T - N & Z \end{bmatrix}. \quad (8)$$

여기에서  $\begin{bmatrix} X & Y \\ Y^T & Z \end{bmatrix} \geq 0$  이고,  $N$ 은 단위행렬이다. 이것을 좀 더 확장하면 아래의 정리를 얻게 된다.

보조정리 1: [23]

기간  $\Omega$ 에  $a(\cdot) \in R^{n_a}$ ,  $b(\cdot) \in R^{n_b}$ ,  $N(\cdot) \in R^{n_a \times n_b}$  과 같이 정의 하면, 어떠한 행렬  $X \in R^{n_a \times n_a}$ ,  $Y \in R^{n_a \times n_b}$ ,  $Z \in R^{n_b \times n_b}$  은 다음과 같은 조건을 만족한다.

$$\begin{aligned} &-2 \sum_j a^T(j) N b(j) \\ &\leq \sum_j \begin{bmatrix} a(j) \\ b(j) \end{bmatrix}^T \begin{bmatrix} X & Y - N \\ Y^T - N & Z \end{bmatrix} \begin{bmatrix} a(j) \\ b(j) \end{bmatrix}, \\ &\quad \left( \begin{bmatrix} X & Y \\ Y^T & Z \end{bmatrix} \geq 0 \right). \end{aligned} \quad (9)$$

보조정리 2:

$D, E, \Delta$ 가 적절한 차수의 실수 행렬이고  $\Delta = \text{diag}\{\Delta_1, \Delta_2, \dots, \Delta_s\}$ ,

$$\Delta_i^T \Delta_i \leq I_{n_i}, \quad i=1, \dots, r$$

와 같으면, 어떠한 실수 행렬

$$\Lambda = \text{diag}\{\lambda_1 I, \dots, \lambda_r I\} > 0$$

에 대하여 다음의 부등식이 성립한다.

$$DAE + E^T \Delta^T D^T \leq DAD^T + E^T \Lambda^{-1} E, \quad (10)$$

$$DAE + E^T \Delta^T D^T \leq D\Lambda^{-1} D^T + E^T \Lambda E. \quad (11)$$

증명 :

$(\Lambda^{-\frac{1}{2}} D^T - \Lambda^{-\frac{1}{2}} \Delta E)^T (\Lambda^{-\frac{1}{2}} D^T - \Lambda^{-\frac{1}{2}} \Delta E) \geq 0$ 과 같은 사실로부터 다음을 따른다.

$$DAE + E^T \Delta^T D^T \leq DAD^T + E^T \Delta^T \Lambda^{-1} \Delta E.$$

$\Delta$ 와  $\Lambda$ 에 대해, 다음의 관계를 가진다.

$$\begin{aligned} \Delta^T \Lambda^{-1} \Delta &= \text{diag}\{\lambda_1^{-1} \Delta_1^T \Delta_1, \dots, \lambda_r^{-1} \Delta_r^T \Delta_r\} \\ &\leq \text{diag}\{\lambda_1^{-1} I_{n_1}, \dots, \lambda_r^{-1} I_{n_r}\} = \Lambda^{-1}. \end{aligned}$$

따라서  $E^T \Delta^T \Lambda^{-1} \Delta E \leq E^T \Lambda^{-1} E$ 은 성립하고 이것은 (10)을 따른다.

(11)의 증명은

$$(\Lambda^{-\frac{1}{2}} D^T - \Lambda^{-\frac{1}{2}} \Delta E)^T (\Lambda^{-\frac{1}{2}} D^T - \Lambda^{-\frac{1}{2}} \Delta E) \geq 0.$$

을 이용한다. 증명 과정은 생략한다. ■

(7)과 같이 지연이 표현된 이산 NCS에 관한 식에서 안정성을 판단하기 위해서는 아래의 정리 1을 사용할 수 있다. 정리 1:

$$\begin{bmatrix} (1, 1) & (1, 2) \\ (1, 2)^T & (2, 2) \end{bmatrix} < 0, \quad (12)$$

$$\begin{bmatrix} X_i & Y_i \\ Y_i^T & Z_i \end{bmatrix} \geq 0. \quad (13)$$

( $i=1, \dots, m$ )

$$(1, 1) \doteq \begin{bmatrix} -P + \sum_{i=1}^m (\bar{\tau} X_i + Y_i + Y_i^T + Q_i) & -Y_M \\ -Y_M^T & -Q_M \end{bmatrix}$$

$$Y_M \doteq [Y_1 \dots Y_m]$$

$$Q_M \doteq \text{diag}\{Q_1, \dots, Q_m\}$$

$$(1, 2) \doteq \begin{bmatrix} A^T P & -\bar{\tau} \sum_{i=1}^m (A-I)^T Z_i \\ A_M^T P & \bar{\tau} A_Z^T \end{bmatrix}$$

$$A_M \doteq [A_1 \dots A_m]$$

$$A_Z \doteq [Z_1 A_1 \dots Z_m A_m]$$

$$(2, 2) \doteq \text{diag}\{-P, -\bar{\tau} Z_1, -\bar{\tau} Z_2, \dots, -\bar{\tau} Z_m\}$$

(12), (13) 과 같은 조건에 맞는  $P_i, Q_i, X_i, Y_i, Z_i$  ( $i=1, \dots, m$ )이 존재한다면, (7)과 같은 시스템은  $0 \leq \tau_i \leq \bar{\tau}_i$  ( $i=1, \dots, m$ )을 만족하는 어떠한 지연  $\tau_i$ 에 대해서도 안정성을 따라간다.

즉, 정리 1을 이용하면, 이산 NCS의 안정성을 유지하는 MADB를 구할 수 있다. 이러한 방법으로 얻어진 MADB는

NCS에서 샘플링 주기의 최대 한계로 사용된다. 3장에서는 산출된 MADB를 이용하여 샘플링 주기를 결정하고, 실시간 전송을 위한 스케줄링을 한다.

### III. 다중 제어 루프에서 스케줄링 알고리즘

3장에서는 MADB를 이용한 네트워크 스케줄링 기법을 제안한다. 제안된 스케줄링 기법은 시스템의 안정성이 유지 되도록 샘플링 주기를 결정하고, 각 노드에 대역폭을 할당한다.

본 논문에서는 기기 알람 정보 같은 산발성 긴급 데이터가 우선순위를 가지는 변형된 EDF (Earliest Deadline First) 스케줄링 방법을 적용했다. 즉, 실시간 주기적인 데이터는 각 제어 루프에 설정된 MADB를 마감시한으로 사용하여 EDF 알고리즘으로 스케줄링 된다. 그리고 실시간 산발 데이터가 발생되면 우선순위를 부여해서 주기적 데이터의 마감시한에 상관없이 먼저 처리된다. 산발성 긴급 데이터가 없다면, 주기적 데이터의 처리 다음에 비 실시간 메시지가 스케줄링 된다.

$T_s$ (Sporadic),  $T_p$ (Periodic),  $T_N$ (Non-real-time message) 으로 구성된 기본 스케줄링 주기는 그림 4와 같이 이루어진다. 이 세 가지의 주기 외에도 동기화 주기  $T_o$ 를 포함할 수 있다.

본 논문에서는 스케줄링 알고리즘을 적용하기 위해서는 다음과 같은 항목들을 가정한다.

- 네트워크에서 통신에러는 없다(재전송이 없다).
- 루프에 있는 센서의 샘플링 타임은 기본 스케줄링 주기의 시작 순간에 맞추어 동기 시킨다.
- 센서에서 제어기 또는 제어기에서 구동기로 전송되는 패킷은 같은 길이를 가진다.
- 각 루프의 샘플링 주기는 루프의 MADB를 초과하지 않는다.
- 제어기 계산 지연은  $\tau_a$ 나  $\tau_s$ 에 포함 된다[20].

첫 번째에서 네 번째까지의 가정은 통신환경을 간소화하기 위해 사용되었다. 다섯 번째 가정은 노드 데이터 전송 시간에 제어기 지연 시간을 합치기 위하여 사용되었다.

각 제어 루프의 샘플링 주기를 결정하고, 스케줄링 가능성을 확인하기 위하여 (14) 와 같은 사용률에 관한 공식을 사용한다.

$$U = \sum_{i=1}^n \frac{C_i}{\min(t_i, D_i)} \leq 1. \quad (14)$$

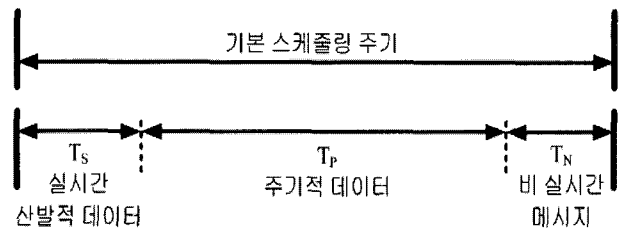


그림 4. 스케줄링 주기 구조.

Fig. 4. Configuration of phase in a scheduling period.

$t_j$ 는 각 작업의 주기를 나타내고,  $D_j$ 는 각 작업의 마감 시한을 나타낸다.  $C_j$ 는 각 작업의 수행시간이다. 따라서 각 제어 루프에서  $t_j$ 는 샘플링 주기,  $D_j$ 는 MADB로  $C_j$ 는 전송시간으로 각각 바꿀 수 있다. (14)를 실시간 주기적인 데이터의 채널 사용률에 관한 식으로 바꾸면 (15)와 같이 바꿀 수 있다.

$$U_p = \sum_{j=1}^n \frac{T_p^j}{\min(T^j, T_D^j)} \leq 1 - \max(U_S, U_N^m). \quad (15)$$

$$T_p^j = \left( \sum_{i=1}^{N_S^j} (T_{S_i}^j + T_{O_p}) + \sum_{i=1}^{N_A^j} (T_{C_i}^j + T_{O_p}) \right)$$

( $U_p$ : 실시간 주기적인 데이터의 채널 사용률,  $U_s$ : 실시간 산발성 긴급 데이터의 채널 사용률,  $U_N^m$ : 메시지의 최소 채널 사용률,  $T^j$ : j 번째 루프의  $\beta$  데이터의 전송 시간( $\beta$ : P(주기), S(산발), N(메시지)),  $N_A^j$ : j 번째 루프의 구동기의 수,  $N_S^j$ : j 번째 루프의 센서의 수,  $T_D^j$ : j 번째 루프의 MADB,  $T^j$ : j 번째 루프의 샘플링 주기,  $T_{O_p}$ :  $\beta$  데이터의 오버헤드 전송시간,  $T_{\alpha_i}^j$ : j 번째 루프의 i 번째  $\alpha$ 노드에서 주기적 데이터 전송 시간( $\alpha$ : C(제어기), A(구동기), S(센서))  $U_p$ 를 계산한 결과가 부등식 (15)를 만족한다면 해당 시스템은 스케줄링이 가능하다고 판단한다. 제한된 스케줄링 방법은 실시간 산발성 긴급 데이터 또는 비 실시간 메시지 중 하나만을 전송하기 때문에 둘 중 더 큰 채널 사용률만 빠진다. 만일 위의 식을 만족하지 않는다면 스케줄링이 불가능하므로 더욱 고속의 필드버스 프로토콜을 사용하여야 한다.

본 논문에서 샘플링 주기 결정시 고려사항은 각 제어루프가 동일한 채널 사용률을 확보 하는 것이다. 각 제어루프에 동일한 채널 사용률이 보장 되도록 작업 주기를 정한 후 이 작업주기를 샘플링 주기로 사용한다. (16)은 j 번째 제어루프에서 동일한 채널 사용률이 보장되는 조건을 나타낸다.

$$\frac{T_p^j}{T^j} = \frac{1 - \max(U_S, U_N^m)}{n}, \quad (T^j \leq T_D^j). \quad (16)$$

(16)을 이용하여 j 번째 샘플링 주기를 구하는 식을 나타내면 (17)과 같다.

$$T^j = \left\lceil \frac{n}{1 - \max(U_S, U_N^m)} \right\rceil \times T_p^j, \quad (T^j \leq T_D^j) \quad (17)$$

각 제어 루프의 스케줄링 우선순위를 확인하는 스케줄링 주기는 샘플링 주기들의 최대 공약수를 이용한다. (17)의 우변  $\lceil n / \{1 - \max(U_S, U_N^m)\} \rceil$ 를 이용하여 최대 공약수를 구할 수 있다. 각 제어루프 클럭의 동기화는 샘플링 주기들의 최소 공배수를 이용한다.

그림 5는 상기결과들을 이용한 샘플링 주기 결정 알고리즘의 흐름선도이다.

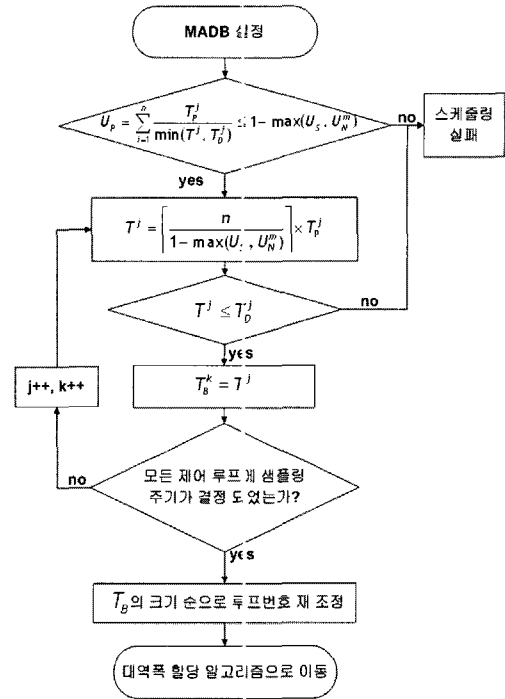


그림 5. 샘플링 주기 결정 알고리즘 흐름선도.  
Fig. 5. Flow chart of sampling period decision algorithm.

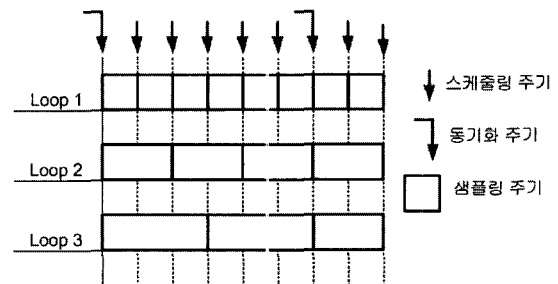


그림 6. 제어루프의 샘플링 주기와 스케줄링 주기.  
Fig. 6. Sampling period and scheduling period.

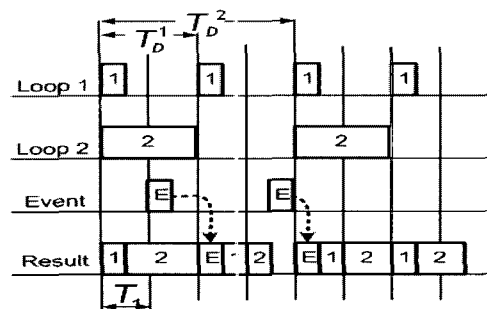


그림 7. 산발성 긴급 데이터 소점형 EDF 스케줄링.  
Fig. 7. Sporadic data preemption EDF scheduling algorithm.

그림 6은 샘플링 주기와 스케줄링 주기의 관계를 나타내고 있다. 제어 루프마다 샘플링 주기가 결정되었으면 샘플링 주기를 기준으로 하여 스케줄링을 실시한다. 본 논문에서는 실시간 산발적인 데이터의 처리를 위한 산발 데이터

선점 EDF 스케줄링을 사용한다. 이 방식에서 산발적인 데이터는 다른 주기적인 작업보다 더 높은 우선순위를 가진다. 즉, 실시간 산발적인 데이터는 다른 주기적인 작업들보다 먼저 스케줄링 된다. 그림 7에서 산발적 데이터 선점 EDF 스케줄링 알고리즘을 표현하고 있다.

실시간 주기적인 데이터들을 스케줄링은 다음과 같은 방법을 사용한다. 루프 1에 있는 센서 데이터 패킷들은 네트워크 매체를 통하여 제어기에 전송된다. 루프 1의 모든 센서 데이터 패킷들이 전송된 후, 루프 1의 제어기에서 제어값의 계산은 시작되고, 루프 1의 제어기 지연이 발생한다. 이 제어기 지연 동안 다음 루프 (루프 2)의 데이터 패킷들이 네트워크 매체를 이용하여 전송된다. 따라서 제어기 지연시간은 루프 2에서의 센서 데이터 전송시간과 겹쳐지게 된다. 다음 루프의 데이터 전송 후에 루프 1의 제어기는 계산된 데이터를 구동기들에게 전송한다.

IV. 시뮬레이션

제안된 방법의 실험을 위해서 세 개의 DC 모터로 이루어진 NCS 시스템을 고려한다[16]. 각 모터는 두 개의 센서와 하나의 구동기로 된 전기자 위치 제어기를 가지고 있고, 필드버스를 통하여 연결되어 있다. 전기자 인덕턴스 ( $L_a$ )와 마찰계수 ( $B_m$ )을 무시한다면, 모터의 동작은 (18)과 같이 모델링 할 수 있다.

$$x_p(k+1) = F_p x_p(k) + G_p u_p(k)$$

$$= \begin{bmatrix} -\frac{K_i K_b}{R_a J} & 0 \\ 1 & 0 \end{bmatrix} x_p(k) + \begin{bmatrix} \frac{K_i}{R_a J} \\ 0 \end{bmatrix} u_p(k), \quad (18)$$

$$y_p(k) = x_p(k).$$

$x_p(k) = [\omega \ \theta]^T$ 이고,  $u_p(k)$ 는 공급 전압(V)이다.  $\omega$ 와  $\theta$ 는 각각 회전자의 각속도(rad/sec)와 변위(rad)이다.  $R_a$ ,  $K_i$ ,  $K_b$ ,  $J$ 는 각각 전기자 저항, 토크상수, 역기전력 상수, 회전자와 부하의 관성을 나타낸다. 만일 이득 ( $K$ )가 상태 피드백 제어기로 사용되었다면, NCS에서 시스템 모델의 단일 제어 루프는 다음과 같이 바뀐다.

$$x_p(k+1) = F_p x_p(k+1) + G_p * K x_p(k - \tau),$$

$$\tau = \tau_c + \tau_{sc} + \tau_{ca}.$$

MADB를 사용한 스케줄링 법칙의 간단한 테스트를 위한 모터의 계수 값은 다음과 같이 한다.

$$R_a = 14(\Omega), \quad K_i = 10(oz - in - A),$$

$$K_b = 0.075(V / rad / sec), \quad J = 0.006(oz - in - sec^2).$$

나머지 모터들은  $R_a$ 를 제외하고 같은 값을 가진다. 나머지 모터의  $R_a$  값은 각각 19( $\Omega$ ), 21( $\Omega$ )이다. 주어진 모터 계수를 바탕으로 계산하였을 때, 기존의 MADB 산출결과와 본 논문의 정리 1산출결과와의 차이를 표 1에서 나타내고 있다. MADB 산출 방식의 비교를 위해 [16]의 Lemma 1을 사용하였다.

표 1에 의하면 정리 1에 의해서 계산된 MADB는 1.780~

표 1. 각 제어루프의 MADB 계산 결과.

Table 1. Simulation results of MADB in the each loop.

제어 루프	$R_a(\Omega)$	기존의 산출방식[16] (ms)	정리 1(ms)
Loop 1	14	3.1	$1.799 \times 10^3$
Loop 2	19	6.1	$1.785 \times 10^3$
Loop 3	21	7.5	$1.780 \times 10^3$

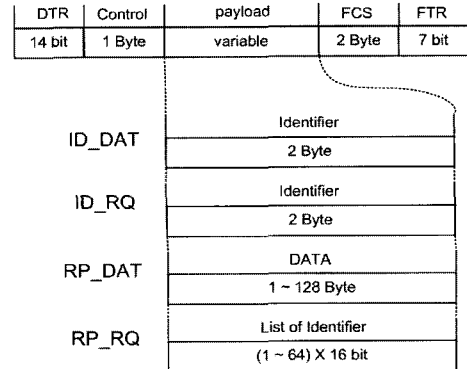


그림 8. FIP의 프레임 형태의 예.

Fig. 8. Frame structure of FIP.

1.799sec 정도의 여유 있는 값이 계산된다. 이 값을 이용하면 제어루프 및 노드수의 증가와 다양한 네트워크 프로토콜의 사용이 가능하다.

계산된 MADB를 사용하여 샘플링 주기를 결정하고 스케줄링하는 과정을 살펴보도록 한다. 계산을 간단하게 하기 위해 MADB는 3, 6, 7ms 로 가정한다. 전송 속도는 1Mbps 이다. 센서와 구동기 데이터는 4Byte 로 가정한다. 시뮬레이션을 위한 네트워크 계수는 다음과 같이 가정한다.

$$T_D^1 = 3ms, \quad T_D^2 = 6ms, \quad T_D^3 = 7ms,$$

$$U_S = 0.17,$$

$$U_N^m = 0.16,$$

$$N_C^j = N_A^j = 1, \quad j = 1, 2, 3,$$

$$N_S^j = 2, \quad j = 1, 2, 3,$$

$$N^j = 3, \quad j = 1, 2, 3,$$

$$N^* = 1, \quad P = 3,$$

$N^*$ 는 NCS 제어루프에서 분리된 산발성 긴급 데이터 발생 노드들의 총 수 이다.  $P$ 는 같은 매체를 사용하는 루프들의 총 개수 이다.

일반적인 동작에서 구동기 노드들은 어떤 데이터도 보내지 않는다고 가정하기 때문에 모든 구동기 노드들의 전송은 고려하지 않는다. 센서의 데이터 길이는 4byte로 가정하고 제어기의 데이터 길이와 산발적으로 발생하는 데이터의 길이는 2byte로 한다. 본 논문에서는 버퍼링 딜레이와 패키징 딜레이는 무시하고, 데이터의 분할 전송은 없다고 가정한다.

폴링 제어 방식을 사용하는 FIP 필드버스 통신을 예로 들어 설명한다. FIP 필드버스 통신에서 사용되는 프레임의 형태는 그림 8과 같다. 주기적인 메시지 전송 시 ID\_DAT

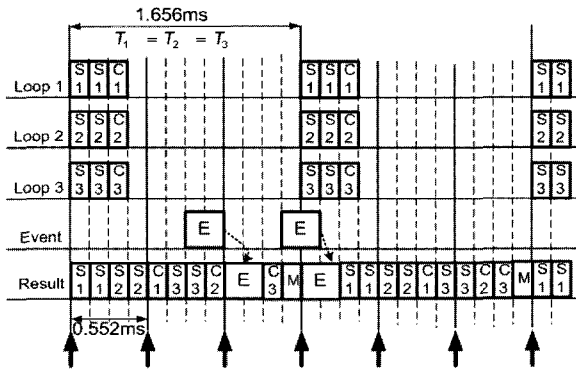


그림 9. MADB가 3, 6, 7ms 이고 각 루프의 노드의 수가 같은 경우의 스케줄링 결과.

Fig. 9. Scheduling result in case of MADB 3, 6, 7ms and same number of nodes.

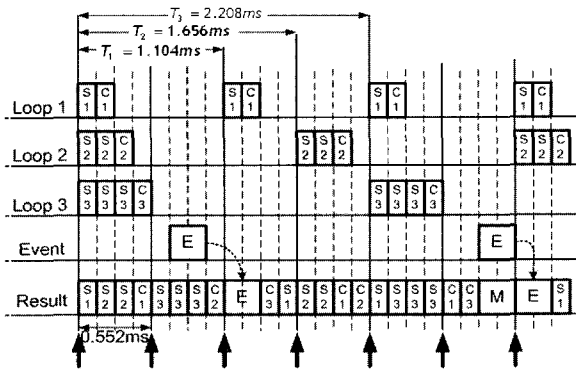


그림 10. MADB가 3, 6, 7ms 이고 각 루프의 노드의 수가 다른 경우의 스케줄링 결과.

Fig. 10. Scheduling result in case of MADB 3, 6, 7ms and various number of nodes.

프레임은 버스 중재자가 보내는 질의 프레임이고 RP\_DAT 는 그에 대한 응답 프레임이다. 비 주기적인 메시지를 보낼 때는 ID\_RQ와 RP\_RQ 프레임이 추가 된다.

$T_{O_p}^M$ 과  $T_{O_s}^M$ 는 각각 주기적인 데이터와 산발적인 데이터의 메시지 오버헤드를 나타낸다.  $T_{O_p}^P$ 과  $T_{O_s}^P$ 는 주기적인 데이터와 산발성 긴급 데이터의 프로토콜 오버헤드를 나타낸다.  $T_s$ 는 산발적인 데이터의 전송 시간이다.

$$T_{S_i}^j = T_{C_i}^j = M = 4(\text{Byte}) \times 8(\mu\text{s}/\text{Byte}) = 32(\mu\text{s}),$$

$$i = 1, 2, \quad j = 1, 2, 3,$$

$$T_{O_p}^M = 45(\text{RP\_DAT}) \times 1(\mu\text{s}/\text{bit}) = 45(\mu\text{s}),$$

$$T_{O_p}^P = 61(\text{ID\_DAT}) \times 1(\mu\text{s}/\text{bit}) = 61(\mu\text{s}),$$

$$T_s^M = 2(\text{Byte}) \times 8(\mu\text{s}/\text{Byte}) = 16(\mu\text{s}),$$

$$T_{O_s}^M = 45(\text{RP\_DAT}) \times 1(\mu\text{s}/\text{bit}) = 45(\mu\text{s}),$$

$$T_{O_s}^P = (61(\text{ID\_RQ}) + (45 + 16)(\text{RP\_RQ}) + 61(\text{ID\_DAT})) \times 1(\mu\text{s}/\text{bit}) = 183(\mu\text{s}),$$

본 시뮬레이션에서 실시간성 주기적 데이터 패킷 하나를 전송하는데 걸리는 시간은 138 $\mu$ s이고, 이 시간을 한 타임

슬롯으로 선정했다. 샘플링주기의 계산이나 산발성 긴급 데이터의 전송 시간은 슬롯 단위로 증가 하거나 감소된다. 만일 산발성 긴급 데이터가 한 샘플링 주기에서 발생하지 않거나 하나만 발생한다면 남은 시간은 비 실시간 메시지 전송으로 채워지게 된다. 시뮬레이션 결과를 그림 9에서 나타내었다.

그림 9에서 세 루프의 주기적인 데이터 전송 노드의 수가 똑같기 때문에 샘플링 주기는 세 루프 모두 1.656 ms로 같다. 스케줄링 주기는 0.552 ms 이다.

각 루프내의 주기적 데이터 전송 노드수가 다른 경우의 스케줄링 결과를 그림 10에 나타내었다. 그림 10은 첫 번째 루프의 노드 수는 2, 두 번째 루프는 3, 세 번째 루프는 4 일 경우에 대하여 스케줄링 한 결과이다. 각 루프의 샘플링 주기는 1.104, 1.656, 2.208 ms 이고, 0.552 ms 마다 스케줄링 한다. 그리고 4.416 ms 마다 각 루프의 동기화가 이루어진다.

### VI. 결론

본 논문에서는 LMI 형식을 이용하여 이산시간 NCS의 안정성을 보장하는 MADB를 산출하는 방법과 산발성 긴급 데이터의 효율적인 처리를 위한 스케줄링 기법을 제안 하였다. 제안된 이산 시간 NCS의 MADB 산출 방식은 여유 있는 MADB의 산출이 가능하므로 많은 수의 노드와 제어 루프에 대한 스케줄링을 가능하게 한다.

계산된 MADB를 이용한 샘플링 주기 결정방법과 대역폭 할당 기법을 소개하였다. 샘플링 주기는 각 제어루프의 채널 사용률이 균등하게 되도록 결정하였다. 대역폭할당은 산발성 긴급 데이터가 채널을 선점하는 방식의 변형된 EDF 스케줄링 기법을 사용하였다. 이를 이용하여 세 가지 형태의 데이터(실시간 주기, 산발성 긴급, 비실시간 메시지)에 대해 효과적으로 대역폭을 할당하였다.

마지막으로 FIP 필드버스에 제안된 스케줄링 법칙을 적용하여서 샘플링 주기선정과 노드들의 대역폭 할당에 유용하게 사용될 수 있음을 보였다.

향후연구로는 샘플링 주기 결정에서 특정 루프가 다른 루프보다 큰 채널 사용률을 필요로 하는 경우에 대한 고려와 대역폭 할당 시에 비실시간 메시지 데이터의 효율적인 처리에 관한 연구가 필요하다.

### Appendix

증명: 정리 1

먼저, (19)과 같은 리아프노프 함수를 선택한다.

$$V(k) \doteq V_1(k) + V_2(k) + V_3(k),$$

$$V_1(k) \doteq x(k)^T P x(k),$$

$$V_2(k) \doteq \sum_{i=1}^m \sum_{\beta=-\tau_i}^{-1} \sum_{j=i+\beta+1}^k \delta x(j)^T Z \delta x(j), \quad (19)$$

$$\delta x(j) \doteq [x(j) - x(j-1)],$$

$$V_3(k) \doteq \sum_{i=1}^m \sum_{j=k-\tau_i}^{k-1} x(j)^T Q x(j).$$

아래의 식이 성립하기 때문에 이산 시간 시스템의 상태

방정식은 (20)과 같이 적을 수 있다.

$$\begin{aligned}
 x(k-h_i) &= x(k) - \sum_{j=k-h_i+1}^k [x(j) - x(j-1)], \\
 x(k+1) &= (A + \sum_{i=1}^m A_i)x(k) \\
 &\quad - \sum_{i=1}^m \left\{ A_i \sum_{j=k-h_i+1}^k [x(j) - x(j-1)] \right\}. \quad (20)
 \end{aligned}$$

그리고  $V_1$ 의 증가량은 아래 (21)을 만족한다.

$$\begin{aligned}
 \Delta V_1(k) &= V_1(k+1) - V_1(k) \\
 &= x(k)^T \left( A + \sum_{i=1}^m A_i \right)^T P \left( A + \sum_{i=1}^m A_i \right) x(k) \\
 &\quad - 2x(k)^T \left( A + \sum_{i=1}^m A_i \right)^T P \Gamma \\
 &\quad + \sum_{i=1}^m \left( \delta x_{h_i}(k)^T A_i^T P A_i \delta x_{h_i}(k) \right) \\
 &\quad - x(k)^T P x(k), \\
 \Gamma &= \sum_{i=1}^m \left\{ A_i \sum_{j=k-h_i+1}^k [x(j) - x(j-1)] \right\} \\
 \delta x_{h_i}(k) &= [x(k) - x(k-h_i)]. \quad (21)
 \end{aligned}$$

모든  $i=1, 2, \dots, m$ 과  $j \in [k-h_i+1, k]$ 에서 본 논문의 (9)의  $a(\cdot)$ ,  $b(\cdot)$ ,  $N$ 을 다음과 같이

$$\begin{aligned}
 a(j) &\doteq x(k), \\
 b(j) &\doteq x(j) - x(j-1), \\
 N &\doteq \left( A + \sum_{i=1}^m A_i \right)^T P A_i.
 \end{aligned}$$

로 정의하고 논문의 정리 1을 적용하면 논문의 (14)을 얻을 수 있다.

$$\begin{aligned}
 \Delta V_1(k) &\leq x(k)^T H_1 x(k) \\
 &\quad + \sum_{i=1}^m 2x(k)^T [-Y_i + A^T P A_i] x(k-h_i) \\
 &\quad + \sum_{i=1}^m x(k-h_i)^T A_i^T P A_i x(k-h_i) \\
 &\quad + \sum_{i=1}^m \sum_{j=k-h_i+1}^k \delta x(j)^T Z_i \delta x(j), \\
 H_i &\doteq A^T P A - P + \sum_{i=1}^m (\bar{h} X_i + Y_i + Y_i^T). \quad (22)
 \end{aligned}$$

따라서  $\Delta V_2(k)$ 와  $\Delta V_3(k)$ 를 얻을 수 있다.

$$\begin{aligned}
 \Delta V_2(k) &= \sum_{i=1}^m h_i H_2^T Z_i H_2 \\
 &\quad - \sum_{i=1}^m \sum_{j=k-h_i+1}^k \delta x(j)^T Z_i \delta x(j), \\
 H_2 &= (A-D)x(k) + A_1 x(k-h),
 \end{aligned}$$

$$\Delta V_3(k) = \sum_{i=1}^m [x(k)^T Q_i x(k) - x(k-h_i)^T Q_i x(k-h_i)].$$

$$\begin{aligned}
 \Delta V(k) &= \Delta V_1(k) + \Delta V_2(k) + \Delta V_3(k) \\
 &\leq \begin{bmatrix} x(k) \\ X_h(k) \end{bmatrix} \begin{bmatrix} (1, 1)_h & (1, 2)_h \\ (1, 2)_h^T & (2, 2)_h \end{bmatrix} \begin{bmatrix} x(k) \\ X_h(k) \end{bmatrix},
 \end{aligned}$$

$$X_h(k) \doteq [x(k-h_1) \cdots x(k-h_m)]^T,$$

$$\begin{aligned}
 (1, 1)_h &\doteq A^T P A - P + \sum_{i=1}^m (\bar{h} X_i + Y_i + Y_i^T) \\
 &\quad + \sum_{i=1}^m (\bar{h}(A-D)^T Z_i (A-D)) + \sum_{i=1}^m Q_i,
 \end{aligned}$$

$$(1, 2)_h \doteq [B_1 \cdots B_m],$$

$$B_i \doteq -Y_i + \bar{h}(A-D)^T Z_i A_i + A^T P A_i,$$

$$(2, 2)_h \doteq \text{diag}(D_1 \cdots D_m),$$

$$D_i \doteq A_i^T P A_i + \bar{h} A_i^T Z_i A_i - Q_i.$$

위와 같이 리아프노프 크라소브스키(Lyapunov-Krasovskii) 안정성 이론과 스킨 상보법(schur complement)을 이용하면 (12)와 (13)을 유지하는 (7)과 같은 이산시간 시스템이 안정성에 다가서는 것을 알 수 있다. ■

참고문헌

- [1] H. Yoram and R. Asok "Integrated communication and control systems: part I- Analysis," *Journal of Dynamic Systems, Measurement, and Control*, 110, pp. 367-373, Dec., 1988.
- [2] R. Asok and H. Yoram "Integrated communication and control systems: part II - Design consideration," *Journal of Dynamic Systems, Measurement, and Control*, 110, pp. 374-381, Dec., 1988.
- [3] R. Krtolica, Ü. Özaüner, H. Chan, H. Göktaş, J. Winkelman, and M. Liubakka, "Stability of linear feedback systems with random communication delays," *International Journal of Control*, 59, pp. 925-953. 1994.
- [4] F.-L. Lian, J. Moyne, and D. Tilbury, "Network design consideration for distributed control systems," *IEEE Transactions on Control Systems Technology*, 10, pp. 297-307, 2002.
- [5] X. Li, and C. E. de Souza, "Criteria for robust stability and stabilization of uncertain linear systems with state delays," *Automatica*, 33, pp. 1657-1662, 1997.
- [6] X. Li, and C. E. de Souza, "Delay-dependent robust stability and stabilization of uncertain linear delay systems: a linear matrix inequality approach," *IEEE Transactions on Automatic Control*, 42, pp. 1144-1148, 1997.
- [7] P. G. Park, "A delay-dependent stability criterion for systems with uncertain time-invariant delays," *IEEE Transactions on Automatica Control*, 44, pp. 876-877. 1999.
- [8] S. Cavalieri, A. Stefano, and O. Mirabella, "Pre-run-time Scheduling to Reduce Schedule Length in the Fieldbus Environment," *IEEE Transactions on Software Engineering*, 21, pp. 865-880. 1995.
- [9] K. Tindell, H. Hansson, and A. Wellings, "Analyzing real-time communications: Controller area network," *IEEE Real-Time Systems Symposium*, pp. 259-263, 1994.
- [10] P. Lorenz, and Z. Mameri, "Real-time software architecture: Application to FIP fieldbus," *Proceedings of AARTC*, pp. 415-423. 1995.
- [11] P. Raja, K. Vijayananda, and J. D. Decotignie, "Polling



- algorithms and their properties for fieldbus networks," *Proceedings of IECON'93*, pp. 530-534, 1993.
- [12] J. P. Beauvais and A.-M. Deplanche, "Heuristics for scheduling periodic complex real-time tasks in a distributed system," *Proceedings of IECON'95*, pp. 55-60, 1995.
- [13] S. H. Hong, "Scheduling algorithm of data sampling times in the integrated communication and control systems," *IEEE Transactions on Control System Technology*, 3, pp. 225-231, 1995.
- [14] K. M. Zuberi and K. G. Shin, "Scheduling messages on controller area network for real-time CIM applications," *IEEE Transactions on Robotics and Automation*, 13, pp. 310-316, 1997.
- [15] H. Ye, G. C. Walsh, and L. G. Bushnell, "Real-time mixed-traffic wireless networks," *IEEE Transactions on Industrial Electronics*, 48, pp. 883-890, 2001.
- [16] H. S. Park, Y. H. Kim, D. S. Kim, and W. H. Kwon, "A scheduling method for network based control systems," *IEEE Transaction on Control System Technology*, 10, pp. 318-330, 2002.
- [17] M. S. Branicky, S. R. Phillips, and Wei Zhang, "Stability of networked control systems: explicit analysis of delay," *Proceedings of the American Control Conference*, 4, pp. 2352-2357, 2000.
- [18] G. C. Walsh, Hong Ye, and L. Bushnell, "Stability analysis of networked control systems," *Proceedings of the American Control Conference*, 4, 2876-2880, 1999.
- [19] W. Zhang, M. S. Branicky, and S. M. Phillips, "Stability of networked control systems," *IEEE Control Systems Magazine*, 21, 84-99, 2001.
- [20] G. C. Walsh, and H. Ye. "Scheduling of networked control systems," *IEEE Control Systems Magazine*, 21, pp. 57-65, 2001.
- [21] D. S. Kim, Y. S. Lee, W. H. Kwon, and H. S. Park, "Maximum allowable delay bounds of networked control systems," *Control Engineering Practice*, vol. 11, pp. 1301-1313, Nov. 2003.
- [22] J. Nilsson, *Real Time Control Systems with Delays*, Ph. D. dissertation, Dept. of Automatic control, Lund Institute of Technology, Lund, Sweden. 1998.
- [23] Y. S. Moon, P. G. Park, W. H. Kwon, and Y. S. Lee, "Delay-dependent robust stabilization of uncertain state-delayed systems," *International Journal of Control*, 74, pp. 1447-1455, 2001.
- [24] Y. S. Kim, S. K. Jeong, and W. H. Kwon, "A pre-run-time scheduling method for distributed real-time systems in an FIP environment," *Control Engineering Practice*, vol. 6, pp. 103-109, 1998.



**김 동 성**

1992년 한양대 전자공학과 졸업. 2003년 서울대학교 전기 및 컴퓨터 공학부 박사. 2003년~2004년 미국 Cornell Univ. 전기 및 컴퓨터 Post Doc. 2004년~현재 금오공과대학교 전자공학부 교수. 관심분야는 실시간 산업용 통신망 및 네

트워크 기반 제어 시스템.



**최 동 혁**

2005년 국립금오공과대학 전자공학부 제어계측전공 졸업. 2005년~현재 동대학원 네트워크 기반 시스템 연구실 석사과정. 관심분야는 무선 제어시스템 및 무선 필드버스.