

자율 분산 제어 시스템과 공장 자동화

○김 성 수, 최 규 석, 김 영 일, 우 광 방

연세대학교 전기공학과

Autonomous Decentralized Control System and Factory Automation

Sung Soo Kim, Gyo Suck Choi, Yeong Il Kim, Kwang Bang Woo
Dept. of Electrical Eng., Yonsei Univ.

Abstract

This paper describes the concept of ADS (Autonomous Decentralized System) and its application to FA (Factory Automation). The schemes of ADFAS (Autonomous Decentralized Factory Automation System) utilizing these concepts are presented.

1. 서 론

일반적으로 시스템이 복잡해지고 규모가 확장되면 신뢰성, 보수성 및 확장성에 있어서 집중 시스템 (CS: Centralized System)은 분산 시스템 (DS: Decentralized System)보다 불리해진다. 이것은 반도체 기술과 광전송 기술등의 발달로 GROSCH의 법칙이 무너진 때문이다 [1]. 시스템의 기능이 고도화되고 복잡해짐과 아울러 고신뢰도화가 요구되는데에 따라서 자율 분산 시스템 (ADS)이라는 새로운 개념이 생물체에서의 분산시스템으로부터의 유추에 의해 IHARA등에 의하여 제안되었다 [2].

ADS는 일반적으로 늘 비가동상태의 부시스템 (failed subsystem)을 가질 수도 있다는 사실에 근거하고 있고 자율 가제어성 (Autonomous Controllability)과 자율 가협조성 (Autonomous Coordinability)을 가지며 시스템의 평가는 기능의 신뢰성으로써 나타낸다.

그리고 공장자동화 (FA)는 물품의 수주에서 출하까

지의 일체의 생산활동 즉 설계, 가공/처리, 조립, 시험/검사, 반송/보관 및 생산관리/제어등의 제 기능을 효율적, 유기적으로 결합하는 시스템 자동화 기술이다.

본 논문에서는 자율 분산 개념을 도입한 FA 시스템을 구축하는데 이론적 밑바탕이 되는 자율 가제어성과 자율 가협조성 및 자율 가관측성에 대해 기술하고 본 연구실에서 개발을 계획하고 있는 ADFAS (Autonomous Decentralized Factory Automation System)를 소개하기로 한다.

2. 자율 분산 제어 시스템

최근 분자생물학 분야의 연구성과와 더불어 생체 정보 및 제어시스템의 공학적 응용연구가 활발히 전개되고 있다. 이러한 연구중의 하나로 자율 분산 시스템이라는 새로운 시스템이 연구 대상으로서 검토되고 있다. ADS는 다음과 같이 정의한다.

(정의 1) 자율 분산 시스템: 자율 가제어성과 자율 가협조성을 갖는 시스템을 자율 분산 시스템이라고 정의한다.

(정의 2) 자율 가제어성: 전체 시스템중 일부의 부시스템이 down되거나 시스템을 확장, 보수하는 경우에도 다른 부시스템들은 자기 자신을 제어할 수 있을 때 그 시스템은 자율 가제어성을 갖는다고 정의한다.

(정의 3) 자율 가협조성: 전체 시스템중 일부의 부시스템이 down되거나 시스템을 확장, 보수하는 경우

도 다른 부시스템들은 상호 협조를 하여 각각의 목적을 균형화 할 수 있을 때 그 시스템은 자율 가협조성을 갖는다고 정의한다.

자율 분산 시스템은 식 1과 같이 표현할 수 있다.

$$\begin{bmatrix} X_1(k+1) \\ \vdots \\ X_m(k+1) \end{bmatrix} = \begin{bmatrix} A_{11} & \dots & A_{1m} \\ \vdots & \ddots & \vdots \\ A_{m1} & \dots & A_{mm} \end{bmatrix} \begin{bmatrix} X_1(k) \\ \vdots \\ X_m(k) \end{bmatrix} + \sum_{i=1}^m \begin{bmatrix} B_{1i} \\ \vdots \\ B_{mi} \end{bmatrix} U_i(k) \quad (1)$$

여기서 A는 nxn 행렬, B_i는 nxr_i 행렬, X는 nx1 벡터

U_i는 r_ix1 벡터다. X_i(k)는 n_i차원의 상태벡터이고

$$X_i(k) = [X_{i1}(k)^T, \dots, X_{in_i}(k)^T]^T$$

$$X(k) = [X_1(k)^T, \dots, X_m(k)^T]^T \quad (2)$$

$$n = \sum_{i=1}^m n_i \quad \text{이다.}$$

ADS의 제어특성

(정리) 자율 가제어성의 필요충분조건 : 시스템 1 이 자율 가제어이기 위한 필요충분조건은

$$\rho[H(i_1, \dots, i_k)RG(i_1, \dots, i_k)] = n_{i_1} + \dots + n_{i_k} \quad (3)$$

이다. 식 3 은 $1 \leq k \leq m$ 일 때 집합 $(1, 2, \dots, m)$ 의 역집합의 모든 원소 (i_1, i_2, \dots, i_k) 에 대해 성립해야 한다.

여기서 R은 전체 시스템 1 의 가제어성 행렬이고 $H(i_1, \dots, i_k)$ 와 $G(i_1, \dots, i_k)$ 는 식 4 와 5 에 의해 정의된다.

$$X(i_1, \dots, i_k) = H(i_1, \dots, i_k)X = [X_{i_1}^T, \dots, X_{i_k}^T]^T \quad (4)$$

$$R(i_1, \dots, i_k) = RG(i_1, \dots, i_k) = [R_{i_1}, \dots, R_{i_k}] \quad (5)$$

이때 $X(i_1, \dots, i_k)$ 는 부시스템 $S(i_1, \dots, i_k)$ 의 상태벡터다. 여기서 중요한 사실은 시스템 1 은 모든 $i=1, 2, \dots, m$ 에 대해 가제어쌍 (A_{ii}, B_{ii}) 를 갖지 않고서도 자율 가제어일 수 있고, 역으로 모든 $i=1, 2, \dots, m$ 에 대해 가제어쌍을 가져도 자율 가제어가 아닐 수 있다는 것으로 이것은 자율 가제어성이 갖는 중요한 성질이다.

시스템이 자율 가제어성을 갖도록 설계하는 방법에는 두 가지가 있다. 하나는 여러개의 부시스템을 통

합 (integrate)하는 방법이고 다른 하나는 시스템을 여러개의 부시스템들로 분할 (divide)하는 방법이다. 그리고 ADS의 부시스템들의 관측범위는 바로 이웃한 것들 (neighbours)로만 한정된다.

3. 자율 분산과 공장 자동화 시스템

공장 자동화 (FA)란 물품의 수주로부터 물품을 출하하기까지의 일체의 생산활동을 컴퓨터와 산업용 로봇을 써서 시스템의 기본 생산공정을 보다 높은 단계에서 시스템을 자동화하는 기술을 말한다. 이러한 FA는 1950년대부터 부분적으로 시작되었으나 좀 더 근대적인 FA는 1960년대부터 시작되었다고 할 수 있고 1970년대 후반부터는 CIM (Computer Integrated Manufacturing)이 대두되기 시작하였다.

요근래엔 FA 와 FMS (Flexible Manufacturing System) 등과 같이 로봇과 NC 공작기등의 각종의 생산 시스템을 구성하는 설비와 관련기술을 유기적으로 연결시켜서 보다 총합적으로 고도의 생산 시스템을 개발 구축하도록하는 움직임이 급속하게 높아지고 있다[3]. 이들을 가장 크게 총합화하고 지금까지 별도의 영역의 것으로서 취급되어온 제품설계, 생산관리 그리고 생산관리 분야를 컴퓨터에 의한 support의 대상으로서 총합적으로 문제삼도록 하고 있는 것이 CIM (Computer Integrated manufacturing)이므로 이러한 CIM을 목표로 FA를 발전시켜나가야 할 것이다.

FA의 장점은 일체의 생산활동 즉 설계, 가공/처리, 조립, 시험/검사, 반송/보관 및 생산관리/제어등을 효율적 및 유기적으로 결합시켜서 궁극적으로는 무인화 공장을 달성시킬 수 있다고 하는 것이다.

반면에 FA의 단점으로는 상대적으로 많은 부자가 요구되고 위험부담이 크다는 점이다. FA의 문제점을 기술적인 면과 경영상의 두가지 측면에서 살펴볼 수 있겠으나 여기서는 기술적인 면만을 살펴보기로 한다.

첫째, FA기기의 표준화가 제대로 이루어지고 있지 않다. 그래서 FA시스템의 설계나 보수 기술의 발전 및 cost down 등의 저해요인이 된다.

둘째, 아직까지는 FA시스템 전체 및 각 FA 기기의

소프트웨어와 생산관리용 프로그램이, 미속하다.

이러한 문제점의 해결방안은 FA시스템 전체의 시스템공학적 체계화를 이루어나가는 것이다. 그리고 인간을 고려한 FA를 추진함으로써 FA에 수반되는 인간과 기계의 작업분담 문제를 해결할 수 있고 인간의 축적된 경험을 살릴 수도 있다. 그리고 근래엔 자율 분산 개념을 이용한 공장 자동화 시스템이 개발되고있다[4]. 총합적 공장 자동화 시스템 (total factory automation systems)이라고 부르는 이 시스템은 다양한 산업분야에서 사용할 수 있도록, 분산된 workstations에 기초하고 있다. 이 시스템의 주요 구성요소는 flexible manufacturing cell controller와 process local area network이다. 이 시스템의 요구사항은 아래와 같다.

- (1) 전체적 production scheduling으로부터 production line에서 최소 단위로 on-line control로의 통일된 관리 및 제어
- (2) 시스템 하드웨어 및 소프트웨어에서의 고도의 확장성 및 지속성
- (3) high level의 시스템 신뢰성 및 고장허용
- (4) 소프트웨어 시스템 개발 및 고양을 위한 최소의 engineering manpower 및 short-term manpower
- (5) 부시스템으로서의 사용에 적합한 마이크로 컴퓨터의 채택에 의해 만들어질 수 있는 최소가격의 시스템

아래에 이 시스템의 일반적인 구성예를 보인다.

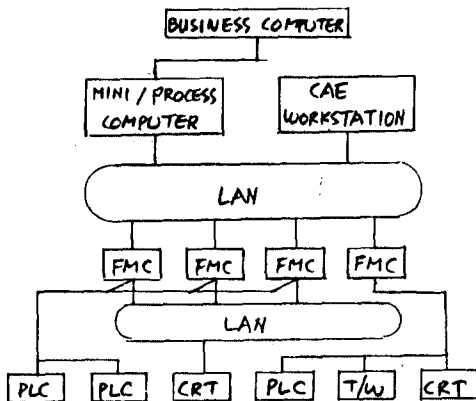


그림 1 자율 분산 개념을 도입한 공장 자동화 시스템의 구성예

4. ADFAS

이상의 관점으로부터 본 연구실에서는 자율 분산 공장 자동화 시스템 (ADFAS : Autonomous Decentralized Factory Automation System) 이라는 새로운 시스템을 연구하고 있다. 자율 분산 개념을 기반으로 하여 연구된 이 제어 방식은 현재의 국내 수준의 기초 연구 및 응용개발의 향상은 물론 생물체에서의 자율 개념을 강조하므로써 새로운 자동화 기술이 이루어질 것이다. 동시에 ADFAS의 확장으로 대규모 시스템 소프트웨어의 기초연구가 수행될 수 있고 실제 공정 제어에 적용시킴으로써 기술개발을 확인할 수가 있다.

ADFAS의 소프트웨어 체계는 LISP를 기반언어로 구성되며 현재 PC 언어로서 쉘 MICRO PASCAL의 컴파일러를 개발중이며 로보트 언어 ALISP도 개발할 계획이다. 그리고 ADFAS를 기반으로한 FA시스템의 연구개발 방향은 다음과 같다. 즉 ADFAS의 하드웨어 구성과 ADFAS 응용은 총합하는 소프트웨어 개발을 기반으로 하여 (1) 지능언어 및 제어방식 연구, (2) 계층 및 대규모 시스템 이론 연구 그리고 (3) 자동화 네트워크 방식 및 구현에 관한 연구로 대별할 수 있는데 본 연구실에서는 지능언어 및 제어방식에 대한 연구를 중심으로 하고 이를 다시 3기(期)로 나누어서 실행해나갈 계획이다.

5. 결론

이상에서 살펴본 바와 같이 ADFAS를 기반으로 하는 FA시스템은 자율 분산개념을 도입하므로 고장허용 (또는 초고신뢰성 : fault tolerance)을 확보할 수 있으므로 신뢰성을 향상시킬 수 있고 부시스템에 자율성을 부여하므로 시스템의 이용률 (utilization)을 높일 수 있다는 결론을 얻었다.

6. 참고문헌

1. L. Kleinrock : " Distributed Systems " IEEE, COMPUTER , 1985, 11, pp. 90-103.
2. 井原康一 外 : "自律分散制御の交通システムへの応用", 日立評論, Vol. 63, No. 11, pp 49-54, 1981.
3. 長谷川 外 : " CIMとは ", 電学学会全国大会シンポジウム, pp. 515-1 ~ 5.15-4, 1986.
4. Norihiro Komoda et al : " An Autonomous, Decentralized Control System for Factory Automation ", IEEE COMPUTER , 1984, 12, pp. 73-83.