

분산 감독제어기와 부분관측 감독제어기의 비교: CIM Testbed 응용

A Comparison of Decentralized and Partially Observed Supervisors: Application to a CIM Testbed

손형일, 이석*
(Hyoung Il Son and Suk Lee)

Abstract : Supervisory control theory, which was first proposed by Ramadge and Wonahm, is a well-suited control theory for the control of complex systems such as semiconductor manufacturing systems, automobile manufacturing systems, and chemical processes because these are better modeled by discrete event models than by differential or difference equation models at higher levels of abstraction. Moreover, decentralized supervisory control is an efficient method for large complex systems according to the divide-and-conquer principle. Decentralized supervisors cannot observe the events those of which occur only within the other supervisors. Therefore decentralized supervisors can be designed according to supervisory control theory under partial observation. This paper presents a solution and a design procedure of supervisory control problem (SCP) for the case of decentralized control and SCP under partial observation (SCPPO). We apply the proposed design procedure to an experimental CIM Testbed. And we compare and analyze the designed decentralized supervisors and partially observed supervisors.

Keywords : decentralized control, discrete event systems, manufacturing automation, observability, supervisory control theory

I. 서론

반도체 생산시스템, 자동차 생산시스템, 화학공정 시스템, HVAC(Heating, Ventilation and Air Conditioning) 시스템, 발전소 등과 같이 복잡한 시스템의 모델링 및 제어에는 이산이벤트 시스템(DES: Discrete Event Systems) 이론에 기초하여 Ramadge 와 Wonham이 제안한 감독제어 이론(supervisory control theory)[1][2]이 적합하다. 감독제어 이론은 플랜트의 제어사양을 만족시키고 실제 발생 가능한 이벤트들(eligible events)을 최대한 허용시키는(maximally permissive) 상위 제어를 설계할 수 있기 때문에 많은 관심을 받고 있으며 특히, 복잡한 제어 시스템에 대해서 효과적임이 증명되고 있다[3][4]. 그리고 이런 제어를 설계하는 문제를 SCP(Supervisory Control Problem)로 명칭한다[5].

일반적으로 SCP는 제어대상 시스템의 구속조건(legal language 또는 behavior specification)에 대한 감독제어기 즉, 중앙(centralized) 감독제어기를 구하는 것이다[5]. 그렇지만 시스템이 커지고, 더욱 복잡하게 된다면 SCP를 풀 때 발생 가능한 이벤트 수에 대해 계산복잡도가 지수적으로 크게 증가하게 된다. 이 문제를 해결하기 위해 분할 정복(divide-and-conquer) 원리를 이용하여 시스템을 몇 개의 하부 시스템으로 나누어 감독제어기를 구하면 계산복잡도를 크게 줄일 수 있다. 이렇게 구해진 감독제어기를 수평적으로는 모듈(modular) 감독제어기, 수직적으로는 상위레벨(high-level) 감독제어기라 한다. 그리고 모듈 및 상위레벨 감독제어기로 구성된 감독제어 시스템을 분산 감독제어 시스템(decentralized supervisory control system)으로 정의한다[6-8].

SCP는 간단하게 시스템에서 발생하는 이벤트가 모두 관측 가능(observable) 하다고 가정하고 문제를 풀지만, 실제 시스템에서는 비용 등의 문제 때문에 이벤트 관측용 센서를 설치할 수 없거나 센서 노이즈 등의 이유로 센서 신호를 충분히 신뢰하지 못해 관측 불가능(unobservable) 한 이벤트들이 존재하게 된다. 이와 같이 시스템에서 발생하는 이벤트들의 일부만 관측 가능한 경우의 SCP를 부분관측하의 SCP(SCPPO: SCP under Partial Observation)로 정의한다.

본 연구는 “분산 감독제어 시스템과 부분관측하의 감독제어 시스템은 어떤 차이가 있으며 시스템 거동을 어떻게 변화시키는가?”는 의문에서 시작되었다. 즉, 분산 감독제어 시스템에서 모듈 감독제어기들은 서로의 하부 시스템에서 발생하는 이벤트들을 관측하지 못하고, 상위레벨 감독제어기는 시스템의 일부 이벤트만을 관측하여 상위레벨 제어를 하기 때문에 분산 감독제어 시스템은 부분관측하의 감독제어 시스템과 설계된 감독제어기가 같고 결국, 같은 시스템 거동을 보일 것으로 예상된다. 본 논문에서는 먼저 SCP를 풀 때의 가제어성(controllability)에 대한 계산복잡도를 줄이기 위해서 서브플랜트(sub-plant)의 개념을 도입하였다. 그리고 서브플랜트의 개념과 Wonham 등의 방법론들을 이용하여 상위 플랜트(high-level plant)와 상위 구속조건에 대한 SCP를 푸는 방법을 제안하였다. 마지막으로 하위 플랜트(low-level plant)의 이벤트들이 부분 관측일 경우에 SCP를 푸는 방법을 제안하였고, 제안한 방법들이 이론적으로 합당함을 증명하였다. 상에서 기술한 의문은 제시한 두 종류의 SCP에 대한 해를 비교함으로써 분석할 수 있다.

3대의 로봇, 2대의 AGV(Automatic Guided-Vehicle), NC(Numerical Controlled) 공작기계, 컨베이어 벨트 등으로 구성된 CIM(Computer Integrated Manufacturing) 시스템을 모사하는 CIM Testbed[9]를 제어대상 플랜트로 두고 제안한 두 가지 방

* 책임저자(Corresponding Author)

논문접수 : 2008. 4. 7., 채택확정 : 2008. 7. 31.

손형일 : KAIST 기계공학전공 박사과정(hyoungil.son@kaist.ac.kr)

이석 : 부산대학교 기계공학부(slee@pnu.edu)

법으로 상위레벨 감독제어기와 부분관측 감독제어기를 설계하여 이를 비교하였다. 본 논문에서는 상위레벨 감독제어기를 설계하기 위해 이전 연구결과 [9]에서 설계, 구현 및 검증한 모듈 감독제어기를 이용하였다. 따라서, [9]에서 설계한 모듈 감독제어기와 본 논문에서 설계하는 상위레벨 감독제어기로 CIM Testbed에 대한 분산 감독제어 시스템을 구성하였다. 그리고 상위레벨 플랜트에서 발생하는 이벤트들만이 관측 가능하다고 가정하고 SCP를 풀어 CIM Testbed에 대한 부분관측하의 감독제어기를 구하고 이를 상위레벨 감독제어기와 비교하였다. 또한, 이들 감독제어기 아래에서의 플랜트 거동도 비교하여 그 특성을 분석하였다.

II. 감독제어 이론

1. 시스템 모델링

DES는 $G = \{Q, \Sigma, \delta, q_0, Q_m\}$ 과 같이 5개의 구성요소를 가진 오토마톤(automaton)으로 모델링 된다. 여기서 Q 는 상태(state)들의 집합, Σ 는 이벤트들의 집합, δ 는 $\delta: Q \times \Sigma^* \rightarrow Q$ 로 정의되는 천이함수(transition function), q_0 는 초기 상태, Q_m 은 작업의 완료를 나타내는 목표 상태 집합(marked state set)이다. 천이함수 δ 에서 Σ^* 는 ε 으로 나타내는 빈 이벤트(null event)와 $\sigma_1, \sigma_2, \sigma_3, \dots, \sigma_k, k \geq 1$ 으로 나타내는 이벤트 연속(문자열: sequence 또는 string)의 집합을 나타낸다. 그리고 제어가능(controllable) 이벤트 집합을 Σ_c , 제어불가능(uncontrollable) 이벤트 집합을 Σ_{uc} , 관측가능(observable) 이벤트 집합을 Σ_o , 그리고 관측불가능(unobservable) 이벤트 집합을 Σ_{uo} 로 나타낸다. 오토마톤 G 가 생성하는 언어(language)를 (1)로 정의한다.

$$L(G) = \{s \mid s \in \Sigma^*, \delta(q_0, s)!\} \quad (1)$$

여기서 $\delta(q_0, s)!$ 는 q_0 에서 문자열 s 가 일어난 다음의 상태가 정의됨을 나타낸다. $L(G)$ 의 전치 닫힘 언어(prefix closure)는

$$\overline{L(G)} = \{t \in \Sigma^* \mid t \leq s \text{ for some } s \in L(G)\} \quad (2)$$

로 나타낸다. 그리고 오토마톤 G 의 목표 언어를

$$L_m(G) = \{s \mid \delta(q_0, s) \in Q_m, L_m(G) \subseteq K(G)\} \quad (3)$$

로 정의한다. 이때 G 가 $\overline{L_m(G)} = L(G)$ 를 만족하면 $L(G)$ 를 비막힘(nonblocking)이라고 말한다. 그리고 비막힘은 감독제어 이론에서 적합한 감독제어기(proper supervisor)가 되기 위한 필요조건이 된다[5][9]. 마지막으로 $\sigma \in \Sigma_{uo}, s \in L(G)$ 에 대해서 $P(\sigma) = \varepsilon, P(s\sigma) = P(s)$ 가 되는 투영함수(projection map) P 가 정의된다[5,10].

2. 감독제어기

감독제어기도 시스템 모델링과 비슷하게 5개의 구성요소를 가진 오토마톤 $S = \{X, \Sigma, \xi, x_0, X_m\}$ 으로 표현된다. G 를 주어진 플랜트라고 하면 감독제어기 S 의 감독제어 아래에서의 플랜트 G 의 거동은 (4)로 나타낸다.

$$S/G = \{X \times Q, \Sigma, \xi \times \delta, (x_0, q_0), X_m \times Q_m\} \quad (4)$$

그리고 $L(G)$ 에 대한 $L(S)$ 의 가제어성(controllability)과 가관측성(observability)이 정의 1과 2에 각각 정의되어 있다.

정의 1: $S \subseteq G$ 에 대해서 다음을 만족하면 S 는 (G, Σ_{uc}) 에 대해서 제어가능 하다고 한다.

$$(\forall s, \sigma) s \in \overline{L(S)}, \sigma \in \Sigma_{uc}, s\sigma \in L(G) \Rightarrow s\sigma \in \overline{L(S)} \quad (5)$$

가제어성은 감독제어기 S 에서 허용되는 임의의 문자열 s 가 있고 플랜트 G 에서 발생할 수 있는 어떤 제어불가능 이벤트 σ 를 정의할 때, 문자열 $s\sigma$ 가 플랜트 G 에서 발생하고 감독제어기 S 도 문자열 $s\sigma$ 가 일어날 수 있도록 허용한다면 감독제어기 S 는 플랜트 G 에 대해서 제어가능 하다는 것을 의미한다.

정의 2: $S \subseteq G$ 에 대해서 다음을 만족하면 S 는 (G, P, Σ_{uo}) 에 대해서 관측가능 하다고 한다.

$$\forall s, s', \sigma \in \overline{L(S)}, P(s) = P(s'), \sigma \in \Sigma_{uo}, s\sigma \in \overline{L(S)}, s'\sigma \in L(G) \Rightarrow s'\sigma \in \overline{L(S)} \quad (6)$$

정의 2는 투영함수에 의해서 같은 문자열을 나타내는 s, s' 가 감독제어기 S 에 의해서 허용되고, 관측불가능 이벤트 σ 가 정의될 때, 감독제어기가 $s\sigma$ 를 허용하고 $s'\sigma$ 가 플랜트에서 일어날 수 있다면 감독제어기는 $s'\sigma$ 도 허용해야 된다는 것을 의미한다.

정의 1을 바탕으로 SCP는 정의 3으로 정의된다.

정의 3: $K \subseteq G$ 인 K 가 주어졌을 때, $L(S/G) = K$ 이고 $L(S/G) = \overline{L_m(S/G)}$ 를 만족하며, (G, Σ_{uc}) 에 대해서 제어가능하고, 최대 언어(supremal language)인 S 를 구하여야.

따라서 K 를 플랜트 G 에 대한 구속조건(legal language)이라 할 때, $L(S/G) = K$ 가 되고, G 에 대해서 비막힘이고 제어 가능한 감독제어기 S 를 찾는게 SCP이다. 그리고 구속조건을 만족하고 제어 가능한 감독제어기는 여러 개가 존재할 수 있는데, 이런 감독제어기들 중에서 K 의 최대 제어가능 부분언어(supremal controllable sublanguage)가 최종적인 SCP의 해가 되는 것이다. 따라서 이렇게 구한 감독제어기 S 가 플랜트 G 에 발생하는 언어를 최대한 허용하게 된다.

마지막으로 감독제어기 S 를 구속조건 $K_j, j=1, 2, \dots, n$ 에 대해서 $S_i, i=1, 2, \dots, m$ 로 설계할 때 이런 S_i 들을 모듈 감독제어기로 정의한다. 그리고 모듈 감독제어기들은 비충돌성(nonconflict) 조건 즉, $\overline{S} = \overline{S_1} \wedge \overline{S_2} \wedge \dots \wedge \overline{S_m}$ 을 만족해야 한다.

3. 상위레벨 감독제어기

플랜트 G 를 하위레벨 플랜트 $G_{lo} = \{Q_{lo}, \Sigma_{lo}, \delta_{lo}, q_{lo,0}, Q_{lo,m}\}$ 로 나타내고 $\Theta: L(G_{lo}) \rightarrow T^*$ 인 정보함수(information map) Θ 를 정의하여 $L(G_{hi}) = \Theta\{L(G_{lo})\}$ 를 만족하는 상위레벨 플랜트 G_{hi} 를 생각하자. 여기서 $T = \{\tau_0, \tau_1, \tau_3, \dots\}$ 는 하위레벨 이벤트들 중에서 상위레벨 플랜트에서 의미를 가지는 이벤트들의 집합을 의미한다. 그러므로 정보함수 Θ 는 임의의

투영함수이다. 이렇게 만들어진 상위레벨 플랜트 G_{hi} 역시 G_{lo} 와 마찬가지로 오토마톤 $G_{hi} = \{Q_{hi}, \Sigma_{hi}, \delta_{hi}, q_{hi,0}, Q_{hi,m}\}$ 로 표현된다. 따라서 상위레벨 플랜트 G_{hi} 에 대해서 상위레벨 구속조건 K_{hi} 를 만들어 2.2절에서 정의한 SCP를 풀면 그 답이 상위레벨 감독제어기가 되는 것이다.

정보함수 Θ 는 G_{lo} 의 각 상태에 상위레벨 이벤트 τ 를 상태 출력(state output)으로 뚫으로써 정의할 수 있다. G_{lo} 에서 Θ 에 대한 상태 출력을 가지는 상태를 보컬 상태(vocal state)라 한다. 그리고 Θ 에 의해서 G_{hi} 를 구하기 전에 G_{hi} 가 G_{lo} 의 제어구조를 보존하게 하기 위해 다음의 두 가지 조건을 만족하도록 G_{lo} 를 재구성해야 된다.

조건 1: OCC (Output Control Consistency)

$$\Theta^{-1}\{L(G_{hi})\}^\uparrow = L(G_{lo}) \quad (7)$$

조건 2: SOCC (Strictly Output Control Consistency)

$$\Theta\left[\Theta^{-1}\{L(G_{hi})\}^\uparrow\right] = L(G_{hi}) \quad (8)$$

조건 1은 임의의 하위레벨 플랜트의 상태로 상태 출력을 가지는 하위레벨 이벤트에 의한 상태 천이가 일어나고 상태 출력을 가지지 않는 하위레벨 이벤트에 의한 상태 천이가 일어날 경우, 이와 같은 두 개의 상태 천이에 대해서 임의의 하위레벨 플랜트의 상태를 서로 구분해야 됨을 의미한다. 그리고 조건 2는 조건 1에 의해서 재구성된 하위레벨 플랜트의 상태들에 대해서 상태 천이가 일어나는 이벤트가 제어가능한지 제어불가능한지에 대해서 상태 출력 값을 새로 정의해줘야 한다는 것을 나타내고 조건 1, 조건 2를 합쳐서 계층적 일관성(hierarchical consistency)이라 한다. 마지막으로 그림 1에 분산 감독제어 시스템의 구조를 나타내었다.

4. 부분관측 하의 감독제어기

만약 플랜트 내에서 발생하는 이벤트들이 모두 관측가능하지 않다면 SCP를 풀 때 슈퍼바이저의 가관측성을 고려해야 된다. 따라서 이 경우 정의 1, 정의 2를 바탕으로 SCP를 정의 4로 새롭게 정의해야 되며 이를 SCPPO(SCP under Partial Observation)라 한다.

정의 4: $K \subseteq G$ 인 K 와 투영함수 P 가 주어졌을 때, $L(S/G) = K$ 이고 $L(S/G) = \overline{L_m(S/G)}$ 를 만족하며, (G, Σ_{uc}) 에 대해서 제어가능하고, (G, P, Σ_{mo}) 에 대해서 관측가능하고,

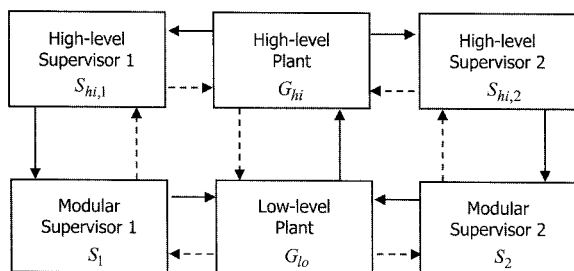


그림 1. 분산 감독제어 시스템.
Fig. 1. Decentralized supervisory control system.

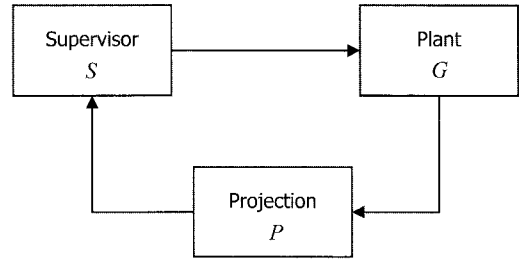


그림 2. 부분관측 하의 감독제어 시스템.
Fig. 2. Supervisory control system under partial observation.

최대 언어(supremal language)인 S 를 구하여야.

즉, G 에 대해서 K 의 최대 제어가능 및 관측가능 부분언어를 구하는 것이 SCPPO이다. 그리고 그림 2에 부분관측 하의 감독제어 시스템을 나타내었다.

III. 감독제어기 설계

1. SCP

본 절에서는 II 장에서 정의한 SCP를 풀기 전에 먼저 다음의 두 가지 문제를 고려하자. 첫 번째, 계산복잡도 문제를 생각해 보자. 전체 플랜트에 대해서 SCP를 풀다는 것은 너무나 큰 계산복잡도를 필요로 하므로 서브플랜트 $G_{sub,i}$ 를 만들어 SCP를 풀면 계산복잡도를 크게 줄일 수 있다. 이를 정리 1에 나타내었다.

정리 1: $G = G^1 \times G^2 \times \dots \times G^n$ 으로 나타낼 수 있는 플랜트가 주어졌을 때, 구속조건 K_i 에 대한 서브플랜트를 $G_{sub,i} \in G$ 로 정의할 때, S_i 가 만약 $(K_i, G_{sub,i})$ 에 대한 SCP의 해가 되고 $G_{sub,j}, i \neq j$ 와 비충돌적(nonconflicting)이면 S_i 는 (K_i, G) 에 대한 SCP의 해가 된다.

증명: SCP에 의해서 S_i 가 (K_i, G) 에 대해서 제어가능하고 비막힘이며 최대 언어임을 증명하면 된다. 먼저, S_i 의 가제어성을 생각해 보자. G 의 이벤트 집합을 Σ , $G_{sub,i}$ 의 이벤트 집합을 $\Sigma_{sub,i}$ 라 정의하면 새로운 이벤트 집합 $\Sigma_{sub,i}^c = \Sigma - \Sigma_{sub,i}$ 를 정의할 수가 있다. 이때 $\Sigma_{sub,i}^c$ 는 S_i 에서 자기순환(selfloop) 이벤트들의 집합이므로 $(\Sigma_{sub,i}^c)_{uc} \subseteq \Sigma_{sub,i}^c$ 에 속하는 모든 제어불가능 이벤트들이 S_i 에서 허용된다. 따라서 S_i 는 G 에 대해서 제어가능하다. 그리고 S_i 가 $G_{sub,j}, i \neq j$ 와 비충돌적이므로 G 에 대해서 비막힘이다. 마지막으로 $\Sigma_{sub,i}^c$ 가 S_i 에서 모두 허용되므로 K_i 에 대해서 최대 언어가 된다. □

두 번째 문제로 SCP의 해를 찾음에 있어서 같은 언어를 발생시키지만 상태의 수가 적고 상태 천이가 적게 발생하여 구현하기가 쉬운 오토마톤을 찾는 것을 고려해 보자. 일반적으로 감독제어기 S 는 다음을 만족시킨다.

$$L(S/G) = L(S) \quad (9)$$

그렇지만 이렇게 설계된 감독제어기는 플랜트에서 발생하는 이벤트들에 대해서 구속조건보다 훨씬 많은 상태가 있기 때

문에 매우 복잡하게 된다. 그래서 실제 감독제어기 구현의 어려움이 발생하기 때문에 다음을 만족하는 보다 간단한 감독제어기 S' 를 구하는 게 감독제어기의 구현 시 간편하다.

$$L(S'/G) = L(S) \tag{10}$$

즉, 감독제어기 S' 의 감독제어 아래에서의 플랜트 거동 언어는 감독제어기 S 감독제어 아래에서의 플랜트 거동 언어와 같지만 (10)을 만족하면서 오토마톤은 S 보다 간단한 S' 을 설계할 수 있다. 이때 구속조건을 K 로 정의하면, S 는 최대 $K \wedge G$ 와 같은 상태 수를 가지지만 S' 은 최대 K 와 같은 상태 수를 가진다. 그렇지만 감독제어기의 감독제어 아래에서의 플랜트 거동을 관측하기에는 S' 가 S 보다 간단하다. 정리 2에 지금까지의 내용을 정리하였다.

정리 2: 다음의 조건을 모두 만족시키는 감독제어기 S' 가 플랜트 G 에 대한 최적 적합 감독제어기(optimal proper supervisor)이다.

- 1) 감독제어기 S' 가 플랜트 G 에 대해서 제어가능 해야 된다.
- 2) $\overline{L_m(S')} = L(S')$
- 3) $\overline{L_m(G) \wedge L_m(S')} = L(G) \wedge L(S')$
- 4) S 를 K 에 대한 최대 제어가능 부분언어라 할 때, $L(S'/G) = L(S)$ 를 만족해야 된다.

증명: 1)은 설계된 감독제어기가 플랜트에 대해서 가제어성을 만족해야 됨을 의미하고, 2)는 감독제어기가 비막힘이어야 됨을 나타낸다. 그리고 3)은 감독제어기가 플랜트와 비충돌적이어야 됨을 나타낸다. 즉, 감독제어기가 플랜트에 대해서 비막힘이어야 됨을 말하고 있다. 따라서 1), 2), 3)을 설계된 감독제어기 S' 가 모두 만족하면 S' 는 적합한 감독제어기가 된다. 마지막으로 4)는 설계된 감독제어기가 플랜트를 감독 제어할 때 최대 제어가능 부분언어를 발생해야 됨을 의미한다. 따라서 S' 가 최적 (optimal 또는 minimally restrictive)한 감독제어기가 된다.

정리 1, 정리 2를 이용하여 SCP의 해를 찾는 방법을 정리 3에 나타내었다.

정리 3: 다음의 방법에 의해서 플랜트 G 와 구속조건 $K_i, i=1,2,\dots,m$ 에 대해 SCP의 해가 되는 S_i 또는 S'_i 을 구할 수 있다.

<SCP 해결 절차(solution procedure)>

단계 0. 제어대상 플랜트의 오토마톤 G 와 구속조건 오토마톤 K_i 를 정의한다.

단계 1. $G_{sub,i}$ 를 설계한다.

단계 2. K_i 의 $G_{sub,i}$ 에 대한 제어가능 이벤트들을 조사하여 가제어성을 검사한다. K_i 가 제어가능하면 단계 5로 제어 가능하지 않다면 단계 3으로 가라.

단계 3. K_i 의 제어가능 이벤트들에서 가제어성을 만족시키지 못하는 이벤트들을 고려하여 K_i 를 재구성한다. 단, K_i 를 재구성 할 때는 해법이 존재하지 않는다.

단계 4. 단계 2로 돌아가라. 만약 가제어성을 만족하는 K_i 를 찾지 못한다면 단계 7로 가라.

단계 5. K_i 가 비막힘인지 검사한다. 만약 K_i 가 막힘이 되는 상태가 존재한다면 그 상태를 제거하고 단계 2로 돌아가라.

단계 6. K_i 가 $G_{sub,i}$ 에 대해서 비충돌적인지 검사한다. K_i 가 비충돌적이라면 $S'_i = K_i$ 가 된다. 그러나 K_i 가 $G_{sub,i}$ 와 충돌적이라면 충돌적이 되는 K_i 의 이벤트 연속들을 조사하여 K_i 를 재구성하고 단계 2로 돌아가라.

단계 7. K_i 의 $G_{sub,i}$ 에 대한 최대 제어가능 부분언어 S_i 를 찾는다. 이렇게 구한 S_i 가 $(K_i, G_{sub,i})$ 에 대한 SCP의 해가 된다.

단계 8. 만약 $L(S_i) = L(S'_i/G_{sub,i})$ 이면 S'_i 는 $(K_i, G_{sub,i})$ 에 대한 SCP의 해가 된다.

증명: 정리 1, 정리 2의 증명으로부터 쉽게 증명할 수 있으므로 증명은 생략한다. □

2. 상위레벨 SCP

II 장 4절에서 정의한 조건 1, 조건 2와 정리 3을 이용하여 상위레벨 감독제어기를 설계하는 과정을 정리 4에 나타내었다.

정리 4: 다음의 방법에 의해서 상위레벨 플랜트 G_{hi} 와 상위레벨 구속조건 $K_{hi,i}, i=1,2,\dots,m$ 에 대해 SCP의 해가 되는 $S_{hi,i}$ 또는 $S'_{hi,i}$ 을 구할 수 있다.

<상위레벨 SCP 해결 절차>

단계 0. 제어대상 하위레벨 플랜트의 오토마톤 G_{lo} 와 상위레벨 구속조건 $K_{hi,i}$ 를 정의한다. 그리고 정보함수 Θ 도 정의한다.

단계 1. $(G_{lo})_{sub,i}$ 를 설계한다.

단계 2. Θ 에 대한 보컬 상태를 정의한 $(G_{lo})_{sub,i}^{vocal}$ 을 구성한다.

단계 3. $(G_{lo})_{sub,i}^{vocal}$ 에 대해서 조건 1을 만족시키는 $\{(G_{lo})_{sub,i}^{vocal}\}_{occ}$ 를 만든다.

단계 4. $\{(G_{lo})_{sub,i}^{vocal}\}_{occ}$ 에 대해서 조건 2를 만족시키는 $\{(G_{lo})_{sub,i}^{vocal}\}_{socc}$ 를 만든다.

단계 5. $G_{hi} = \Theta[\{(G_{lo})_{sub,i}^{vocal}\}_{socc}]$ 를 만든다. 그리고 G_{hi} 를 상위레벨 플랜트로 정의한다.

단계 6. $(K_{hi,i}, G_{hi})$ 에 대해서 정리 3의 단계 2부터 단계 8까지를 진행한다.

증명: 조건 1, 조건 2와 정리 1, 정리 2의 증명으로부터 쉽게 증명할 수 있으므로 증명은 생략한다. □

3. 부분관측 SCP

SCPPO는 (P, Σ_{uo}) 에 대해서 K 의 최대 관측가능 부분언어를 구한 다음 그에 대한 SCP를 풀면 된다. 이를 정리 5에 나타내었다.

정리 5: 다음의 방법에 의해서 플랜트 G 와 구속조건 $K_i, i=1,2,\dots,m$ 그리고 투영함수 P 에 대해서 SCPPO의 해가 되는 S_i 또는 S'_i 을 구할 수 있다.

<SCPPO 해결 절차>

단계 0. 제어대상 플랜트의 오토마톤 G 와 구속조건의 오토마톤 K_i 를 정의한다. 그리고 투영함수 P 와 Σ_{uo} 를 정의한다.

단계 1. $G_{sub,i}$ 를 설계한다.

단계 2. (P, Σ_{uo}) 에 대한 K_i 의 최대 관측가능 부분언어를 만든다. 그리고 이를 K_i^0 로 둔다.

단계 3. $P(K_i^0)$ 를 구한다.

단계 4. $P(G_{sub,i})$ 를 구한다.

단계 5. $P(K_i^0), P(G_{sub,i})$ 에 대해서 정리 3의 단계 2에서 단계 8까지를 진행한다.

단계 6. $P^{-1}\{P(S_i)\}$ 를 구한다.

단계 7. $P^{-1}\{P(S_i)\}$ 가 $G_{sub,i}$ 와 비충돌적인지 검사한다. 만약 충돌적이라면 K_i 를 재구성하여 단계 2부터 다시 시작한다.

단계 8. 마지막으로 $G_{sub,i} \wedge P^{-1}\{P(S_i)\}$ 를 구한다. 이렇게 구한 오토마톤이 감동제어기 S_i 감동제어 아래에서의 플랜트 $G_{sub,i}$ 의 거동이 된다. 그리고 $(K, P, \Sigma_{uc}, \Sigma_{uo})$ 에 대한 $G_{sub,i}$ 의 감동제어기는 S_i 가 된다.

증명: 단계 2에서 (P, Σ_{uo}) 에 대한 K_i 의 최대 관측가능 부분언어를 만들었기 때문에 $P^{-1}\{P(K_i^0)\} = K_i^0$ 이 된다. 따라서 $P(K_i^0), P(G_{sub,i})$ 에 대해서 SCP를 푸는 것은 $(P, \Sigma_{uc}, \Sigma_{uo})$ 에 대해서 K_i 의 최대 제어가능 및 관측가능 부분언어를 구한 것이 된다. □

정보함수 θ 와 투영함수 P 사이에는 유사성이 있다. 즉, θ 에 대한 상위레벨 이벤트와 P 에 대한 관측가능 이벤트의 유사성 때문에 정리 4의 하위레벨 이벤트를 관측불가능 이벤트로 두고 SCPPO를 정리 4로 구한 상위레벨 슈퍼바이저와 같은 결과가 나올 것이라 예상된다. 그렇지만 일반적으로 감동제어기의 감동제어 아래에서의 플랜트 거동은 다르게 나타날 것이다. 왜냐하면 정리 4에서도 알 수 있듯이 상위레벨 감동제어기는 상위레벨 이벤트만을 고려해서 SCP를 푸는 것이 아니기 때문이다. 즉, 계층적 일관성을 만족시키기 위해 하위레벨 이벤트도 고려를 해야 한다. 그렇지만 SCPPO에는 단지 관측가능 이벤트만을 가지고 SCP를 풀기 때문에 두 경우 SCP의 해는 같더라도 플랜트에서 일어나는 언어는 서로 다를 수가 있다.

IV. CIM Testbed 응용

1. 시스템 구성

본 논문에서 제시한 감동제어기 설계방법을 비교, 분석하기 위해 [9]에서 설계된 CIM Testbed를 대상 시스템으로 사용하였다. CIM Testbed는 3대의 로봇, 2대의 AGV, NC 공작기계, 컨베이어 벨트 등으로 구성되어 CIM 시스템을 모사하며 그림 3에 전체 시스템 구성을 나타내었다.

2. 플랜트 모델링

전체 플랜트 모델은 플랜트 각각 구성요소들의 오토마톤 모델들의 동기 곱 연산(synchronous product)으로 구해진다. 설계된 구성요소들의 오토마톤은 [9]를 참고하고 전체 플랜트

모델의 오토마톤은 4,294,967,296개의 상태를 가지고 있다. 그리고 표 1에 플랜트에서 발생하는 이벤트 목록을 정리하였다.

3. 모듈 감동제어기

본 논문에서는 분산 감동제어 시스템을 설계하기 위해 [9]에서 설계, 구현 및 검증한 모듈 감동제어기를 사용하였다. [9]에서는 아래의 구속조건을 주어 비막힘성과 비충돌성을 만족하는 모듈 감동제어기를 설계하였다.

구속조건:

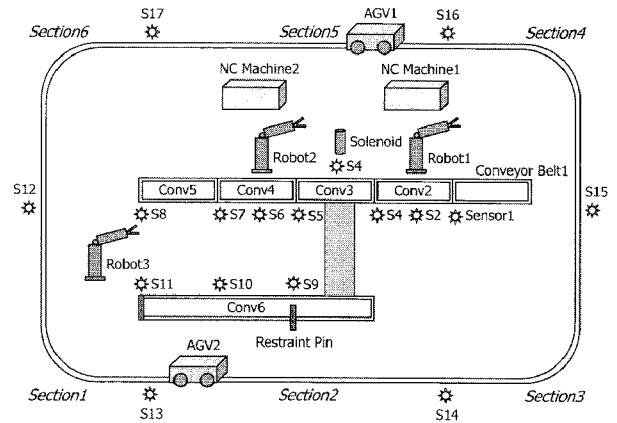


그림 3. CIM Testbed.

Fig. 3. CIM Testbed.

표 1. 이벤트 목록.

Table 1. Event list.

Plant	Event	Controllability
AGV	$mv_AGV\ i$	Controllable
	$umv_AGV\ i$	Uncontrollable
Robot	$op1$	Controllable
	$op2$	Controllable
	$op3$	Controllable
	$op4$	Controllable
	$op5$	Controllable
	$op6$	Controllable
	end_op5	Uncontrollable
NC Machine	opi	Controllable
	end_opi	Uncontrollable
Conveyor Belt	mv_Convi	Controllable
	umv_Convi	Uncontrollable
Sensor	WP_ati	Uncontrollable
	$noWP_ati$	Uncontrollable
	$AGV\ i_atk$	Uncontrollable
	$noAGV\ i_atk$	Uncontrollable
Restraint Pin	pin_down	Controllable
	pin_up	Controllable
Solenoid	mv_Sol	Controllable
	umv_Sol	Uncontrollable

- 1) 컨베이어 벨트 2, 4, 6의 버퍼 크기를 2로 한다.
- 2) 로봇 1이 작업물을 집어 NC 공작기계 1에 옮긴 다음 NC 공작기계 1이 가공을 완료하면 로봇 1이 작업물을 집어 컨베이어 벨트 위에 올려놓는다.
- 3) 로봇 2가 작업물을 집어 NC 공작기계 2에 옮긴 다음 NC 공작기계 2가 가공을 완료하면 로봇 2가 작업물을 집어 컨베이어 벨트 위에 올려놓는다.
- 4) 두 생산라인에서 가공 완료된 제품을 로봇 3이 두 대의 AGV에 구분하여 실는다.
- 5) 슬레노이드가 작업물을 분류시킨다.
- 6) 두 대의 AGV는 각각의 위치에 제품을 하역한다.
- 7) AGV는 서로 충돌하지 말아야 한다.

설계된 감독제어기는 구속조건 1) ~ 5)에 대해서 8개, 6), 7)에 대해서 6개이다. 구속조건 1)에 대한 모듈 감독제어기를 그림 4에 나타내었고, 나머지 감독제어기는 [9]를 참고하여야.

4. 상위레벨 감독제어기

상위레벨 구속조건은 다음과 같다.

상위레벨 구속조건:

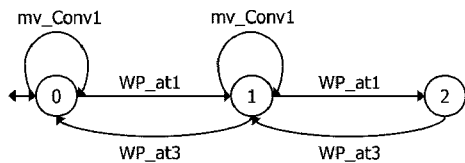
- 1) 컨베이어 벨트 2~4의 전체 버퍼 크기를 3으로 한다.
- 이 때 슬레노이드에 의해서 컨베이어 벨트 6으로 이동하는 가공물은 없다고 가정한다. 상위레벨 구속조건에 대해 정리 4를 이용하여 설계된 상위레벨 감독제어기를 그림 5에 나타내었다. 그리고 그림 5에 표시되지 않은 모든 상위레벨 이벤트들은 각 상태에서 자기순환 이벤트들이다.

아래에 정리 4에 의해서 상위레벨 감독제어기를 설계하는 과정을 상세히 설명하였다. 아래 과정에 나와있는 모든 오토

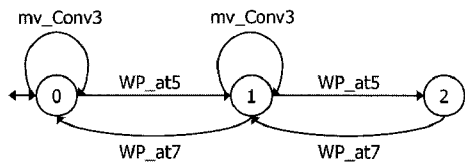
표 2. 정보함수 Θ .

Table 2. Information map Θ .

Low-level event	$\tau = \Theta(\sigma)$	High-level event
<i>WP_at1</i>	τ_1	<i>WP_at1</i>
<i>noWP_at1</i>	τ_0	<i>null</i>
<i>WP_at2</i>	τ_0	<i>null</i>
<i>noWP_at2</i>	τ_0	<i>null</i>
<i>WP_at3</i>	τ_0	<i>null</i>
<i>noWP_at3</i>	τ_0	<i>null</i>
<i>WP_at5</i>	τ_0	<i>null</i>
<i>noWP_at5</i>	τ_0	<i>null</i>
<i>WP_at6</i>	τ_0	<i>null</i>
<i>noWP_at6</i>	τ_0	<i>null</i>
<i>WP_at7</i>	τ_2	<i>WP_at7</i>
<i>noWP_at7</i>	τ_0	<i>null</i>
<i>mv_Conv1</i>	τ_3	<i>mv_Conv1</i>
<i>umv_Conv1</i>	τ_0	<i>null</i>
<i>mv_Conv2</i>	τ_0	<i>null</i>
<i>umv_Conv2</i>	τ_0	<i>null</i>
<i>mv_Conv3</i>	τ_0	<i>null</i>
<i>umv_Conv3</i>	τ_0	<i>null</i>
<i>mv_Conv4</i>	τ_0	<i>null</i>
<i>umv_Conv4</i>	τ_0	<i>null</i>



(a) 컨베이어 벨트 2에 대한 버퍼크기 감독제어기.
(a) Supervisor for buffer size of conveyor belt 2.



(b) 컨베이어 벨트 4에 대한 버퍼크기 감독제어기.
(b) Supervisor for buffer size of conveyor belt 4.

그림 4. 버퍼크기 감독제어기.
Fig. 4. Supervisor for buffer size.

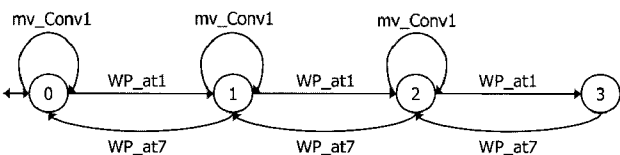


그림 5. 상위레벨 감독제어기.
Fig. 5. High-level supervisor.

마타들은 상태 수가 너무 많기 때문에 본 논문에서는 오토마톤으로 나타내지 않고 상태와 천이 수로 나타내었다.

<상위레벨 관리제어기 설계과정>

단계 0. $G_{lo} = \text{SENSOR1} \times \text{SENSOR2} \times \text{SENSOR3} \times \text{SENSOR5} \times \text{SENSOR6} \times \text{SENSOR7} \times \text{CONV1} \times \text{CONV2} \times \text{CONV3} \times \text{CONV4}$ 로 구해진다. 구해진 G_{lo} 는 (상태: 1024, 천이: 13312)이다. 그리고 상위레벨 구속조건 $K_{hi,1}^0$ 은 자기순환 이벤트들을 제외한 그림 5의 오토마톤으로 정의한다. 따라서 $K_{hi,1}^0$ 은 (상태: 4, 천이: 9)가 된다. 마지막으로 정보함수 Θ 를 표 3으로 정의한다. 단, 상위레벨 이벤트들의 가제어성은 하위레벨 이벤트들과 같다.

단계 1. $(G_{lo})_{sub,1}$ 은 그림 4의 컨베이어 벨트 2에 대한 버퍼크기 관리제어기와 컨베이어 벨트 4에 대한 버퍼크기 관리제어기의 동기 곱 연산으로 설계한다. 설계된 $(G_{lo})_{sub,1}$ 는 (상태: 9, 천이: 102)다.

단계 2. $(G_{lo})_{sub,1}$ 로 부터 $(G_{lo})_{sub,1}^{vocal}$ 을 만든다. 만들어진 $(G_{lo})_{sub,1}^{vocal}$ 는 (상태: 27, 천이: 309)이다. 그림 6, 7에 이 과정의 일부분을 나타내었다. 그림 7을 보면 $(G_{lo})_{sub,1}^{vocal}$ 의 상태 0에서 이벤트 WP_at1 이 발생하면 상태 11로 천이가 되고 상태 출력은 τ_1 이 된다. 그리고 이벤트 WP_at7 , mv_Conv1 이 발생하면 상태 9, 10으로 각각 천이되고 상태 출력은 각각 τ_2 , τ_3 이 된다. 나머지 출력에 대해서는 상태 출력이 모두 τ_0 이다.

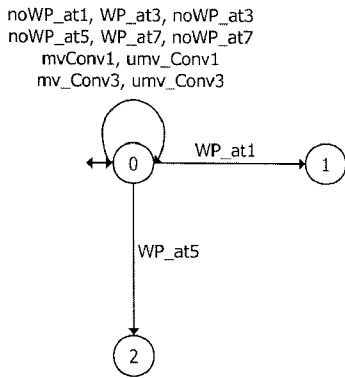


그림 6. $(G_{lo})_{sub,1}$ 의 일부분.

Fig. 6. Part of $(G_{lo})_{sub,1}$

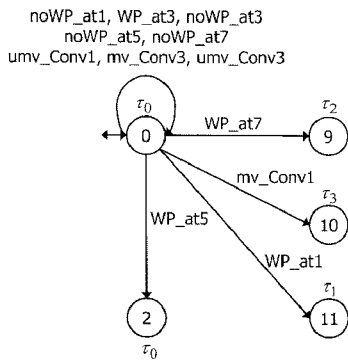


그림 7. 그림 6의 $(G_{lo})_{sub,1}$ 에 대한 $(G_{lo})_{sub,1}^{vocal}$.

Fig. 7. $(G_{lo})_{sub,1}^{vocal}$ for $(G_{lo})_{sub,1}$ shown in Fig. 6.

단계 3. $(G_{lo})_{sub,1}^{vocal}$ 에서 $\{(G_{lo})_{sub,1}^{vocal}\}_{OCC}$ 를 만든다. 만들어진 $\{(G_{lo})_{sub,1}^{vocal}\}_{OCC}$ 는 (상태: 48, 천이: 548)이다. 그림 8, 9를 이용하여 이 단계를 설명하면 다음과 같다. 그림 8의 상태 0에서 이벤트 mv_Conv3 이 발생하지 않고 WP_at7 이 발생한 경우와, mv_Conv3 이 발생하고 WP_at7 이 발생한 경우에 대해서 각각 다르게 상태 출력을 정의해야 된다. 왜냐하면 mv_Conv3 이 제어가능 이벤트이기 때문에 전자의 경우에는 WP_at7 의 발생을 차단(disable)할 수 없지만, 후자의 경우에는 mv_Conv3 을 차단해 결과적으로 WP_at7 을 차단시킬 수 있다. 따라서 후자의 경우에는 상태 출력 값을 제어가능 이벤트로 정의해야 된다. 따라서 이런 문제들을 해결하기 위해 표 2에 정의한 정보함수 Θ 에 추가해야 되는 경우들을 표 3에 나타내었다. 그림 9에 나타낸 최종적으로 설계된 $\{(G_{lo})_{sub,1}^{vocal}\}_{OCC}$ 는 상태 0에서 WP_at7 이 발생하면 상태 9로 천이되고 상태 출력은 τ_2 가 된다. 이어서 mv_Conv3 이 발생하면 상태 27로 천이되고, 상태 27에서 WP_at7 이 발생하면 상태 출력은 τ_2 가 아닌 τ_5 가 되는 것이다.

단계 4. 만들어진 $\{(G_{lo})_{sub,1}^{vocal}\}_{SOCC}$ 는 (상태: 39, 천이, 449)이다. 이 단계에서는 표 3에서 새롭게 정의한 τ_4, τ_5 는 실제 가능한 이벤트가 아니므로 τ_2 와 실제 가능한 상위레벨 이벤트로 새롭게 정의해야 된다. 즉, 정보함수 Θ 는 앞의 이벤트 연속이 무엇이든지 이벤트 WP_at7 이 발생하면 상태 출력

표 3. OCC 조건을 만족시키기 위해 추가되는 정보함수.

Table 3. Additional information map to satisfy the condition for OCC.

Low-level sequence	$\tau = \Theta(\sigma)$	High-level event
$mv_Conv3 \cdot WP_at1$	τ_4	Controllable WP_at1
$mv_Conv3 \cdot WP_at7$	τ_5	Controllable WP_at7

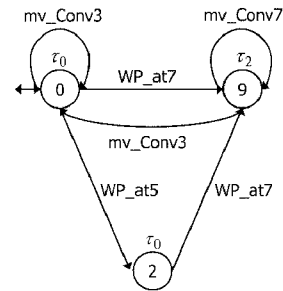


그림 8. $(G_{lo})_{sub,1}^{vocal}$ 의 일부분.

Fig. 8. Part of $(G_{lo})_{sub,1}^{vocal}$.

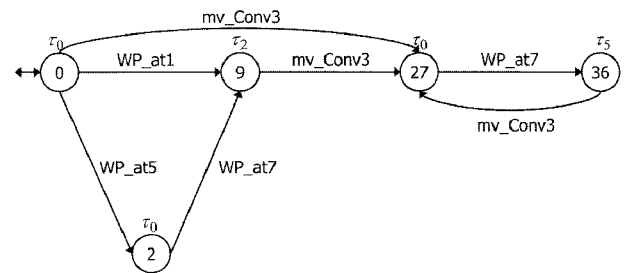


그림 9. 그림 8의 $(G_{lo})_{sub,1}^{vocal}$ 에 대한 $\{(G_{lo})_{sub,1}^{vocal}\}_{OCC}$.

Fig. 9. $\{(G_{lo})_{sub,1}^{vocal}\}_{OCC}$ for $(G_{lo})_{sub,1}^{vocal}$ shown in Fig. 8.

은 τ_2 가 되게 하고 그 이후에 발생하는 하위레벨 이벤트에 대해서는 새로운 상태 출력 (즉, 상위레벨 이벤트)을 정의해야 된다. 정보함수 Θ 를 새롭게 정의하기 이전에 하위레벨 이벤트를 다음과 같이 분할(partition)하자.

$$\Pi\{(\Sigma_{lo})_{sub,1}\} = \begin{cases} \Sigma_1 = \{WP_at1, WP_at7, mv_Conv1\} \\ \Sigma_2 = \{mv_Conv3\} \\ \Sigma_3 = (\Sigma_{lo})_{sub,1} - \Sigma_1 - \Sigma_2 \end{cases} \quad (11)$$

분할 $\Pi\{(\Sigma_{lo})_{sub,1}\}$ 의 의미는 다음과 같다. 표 2에서 정의한 상위레벨 이벤트는 Σ_1 으로 나누고, 나머지 하위레벨 이벤트들 중 제어가능 이벤트들은 Σ_2 로 그리고 제어불가능 이벤트들은 Σ_3 로 나누는 것이다. 이에 따라 최종적으로 정의된 정보함수 Θ 를 표 4에 나타내었다. 여기서 $\tau_6 \sim \tau_{17}$ 는 단지 하위레벨 플랜트에서 제어가능 이벤트가 발생했는지만을 상위레벨 플랜트에서 표시하며, 하위레벨 문자열 발생 후 $\{(G_{lo})_{sub,1}^{vocal}\}_{OCC}$ 의 상태에 따라서 나누어 진다.

그림 10을 보면 어떤 상태에서도 이벤트 WP_at7 이 발생하면 상태 출력은 τ_2 가 된다. 그리고 mv_Conv3 이 발생하면 τ_5 을 상태 출력한다.

단계 5. 상위레벨 플랜트 $G_{hi} = \Theta\{(\{(G_{lo})_{sub,1}^{vocal}\}_{SOCC})\}$ 를 만든다. G_{hi} 는 (상태: 14, 천이: 78)이다.

표 4. SOCC 조건을 만족시키기 위해 추가되는 정보함수.
Table 4. Additional information map to satisfy the condition for SOCC.

Low-level sequence	$\tau = \Theta(\sigma)$	High-level event
$\sigma_1\sigma_2$	$\tau_7, \tau_9, \tau_{11}, \tau_{13}, \tau_{15}, \tau_{17}$	Controllable event occur in low-level plant
$\sigma_1\sigma_2\sigma_3$	$\tau_6, \tau_8, \tau_{10}, \tau_{12}, \tau_{14}, \tau_{16}$	Uncontrollable event occur in low-level plant

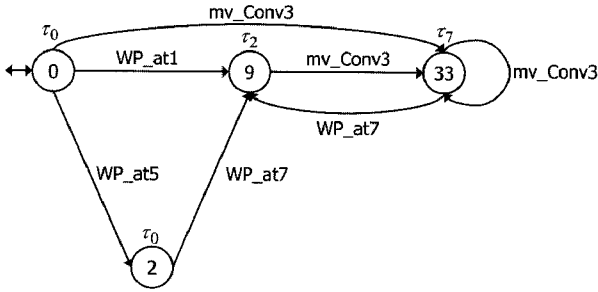


그림 10. 그림 9에 대한 $\{(G_{io})_{sub,1}^{vocal}\}_{SOCC}$.
Fig. 10. $\{(G_{io})_{sub,1}^{vocal}\}_{SOCC}$ for $\{(G_{io})_{sub,1}^{vocal}\}_{OCC}$ shown in Fig. 9.

단계 6-0. $K_{hi,1}^0$ 의 G_{hi} 에 대한 제어 데이터를 보면 $K_{hi,1}^0$ 은 각 상태에서 상태 천이가 정의되지 않은 제어불가능 이벤트 $\tau_{uc,1} \in T_{uc,1}$ 을 차단한다. 즉, $K_{hi,1}^0$ 의 상태 0에서는 τ_2 를 상태 3에서는 τ_1 을 각각 차단하므로 가제어성을 만족하지 않는다. 따라서 이런 이벤트들을 각 상태에서 자기순환 시키고 수정된 상위레벨 구축조건 $K_{hi,1}^1$ 을 만든다.

단계 6-1. $K_{hi,1}^1$ 의 제어 데이터는 $K_{hi,1}^1$ 의 상태 3에서 제어 가능 이벤트 τ_3 를 차단하는 것이므로 $K_{hi,1}^1$ 은 G_{hi} 에 대해서 가제어성을 만족한다.

단계 7. $K_{hi,1}^1$ 은 그림 10에서 알 수 있듯이 비막힘이다.

단계 8. $K_{hi,1}^1$ 은 $L_m(G_{hi}) \wedge L_m(K_{hi,1}^1) = L(G_{hi}) \wedge L(K_{hi,1}^1)$ 을 만족하므로 $K_{hi,1}^1$ 은 G_{hi} 과 비충돌이므로 $S'_{hi,1} = K_{hi,1}^1$ 으로 둘 수 있다. 여기서 $G_{hi} \wedge K_{hi,1}^1$ 는 (상태: 52, 천이: 287)이고 $S'_{hi,1}$ 는 (상태: 4, 천이: 59)이다.

단계 9. $K_{hi,1}^1$ 의 G_{hi} 에 대한 최대 제어가능 부분언어를 구하면 (상태: 52, 천이: 287)인 오토마톤이다. 이것이 주어진 SCP의 해 $S_{hi,1}$ 이다.

단계 10. 최종적으로 $L(S'_{hi,1} \times G_{hi}) = L(S_{hi,1})$ 이므로 $S'_{hi,1}$ 는 $S_{hi,1}$ 보다 상태 수와 천이가 작은 $(K_{hi,1}^1, G_{hi})$ 에 대한 SCP의 해가 된다.

구해진 상위레벨 감독제어기 $S'_{hi,1}$ 는 상위레벨 이벤트 τ_1, τ_2 에 대해서만 상태 천이를 한다. 그리고 상태 0, 1, 2에서는 mv_Conv1 을 허용(enable)하다가 상태 3으로 가면 mv_Conv1 을 차단한다. 따라서 표 2에서 정의한 상위레벨 이벤트들만이 의미를 가지게 된다.

5. 부분관측 하의 감독제어기

정리 5를 이용하여 앞 절에서 설계한 상위레벨 감독제어기와 같은 조건의 부분관측 감독제어기를 설계하겠다. 이것은 3절의 상위레벨 이벤트들을 관측가능 이벤트, 하위레벨 이벤트들을 관측불가능 이벤트로 정의하여 정리 5에 따라 SCPPO를 풀면 된다. 아래에 감독제어기 설계 과정을 나타내었으며 역시, 상태전이가 많은 오토마톤은 상태와 천이수로 나타내었다.

<부분관측 하의 감독제어기 설계과정>

단계 0. G 는 앞 절의 단계 0의 G_{io} 와 같다. 그리고 구축조건 K_1 은 그림 5와 같이 정의한다. 여기서 그림 5에 표시되지 않은 이벤트들은 각 상태에서 자기순환이다. 마지막으로 투영함수 P 를 표 5에 정의하였다.

단계 1. $G_{sub,1}$ 은 컨베이어 벨트 2에 대한 버퍼크기 감독제어기와 컨베이어 벨트 4에 대한 버퍼크기 감독제어기의 감독제어 아래에서의 플랜트 거동으로 정의하자. 만들어진 $G_{sub,1}$ 은 (상태: 576, 천이 4416)이다.

단계 2. (P, Σ_{uc}) 에 대해서 K_1 의 최대 관측가능 부분언어를 구하여 K_1^0 으로 정의하자. 구해진 K_1^0 은 (상태: 2304; 천이: 17472)다.

단계 3. $P(K_1^0)$ 을 구한다. $P(K_1^0)$ 은 그림 11에 나타내었다.

단계 4. $P(G_{sub,1})$ 을 구한다. 그림 12에 $P(G_{sub,1})$ 을 나타내었다.

단계 5. $P(K_1^0)$ 은 $P(G_{sub,1})$ 에 대해서 상태 3에서 이벤트 mv_Conv1 을 차단하는 제어 데이터를 가지고 있으므로 가

표 5. 투영함수 P .

Table 5. Projection map P .

Event	$P(\sigma)$	Observability
WP_at1	WP_at1	Observable
$noWP_at1$	ϵ	Unobservable
WP_at2	ϵ	Unobservable
$noWP_at2$	ϵ	Unobservable
WP_at3	ϵ	Unobservable
$noWP_at3$	ϵ	Unobservable
WP_at5	ϵ	Unobservable
$noWP_at5$	ϵ	Unobservable
WP_at6	ϵ	Unobservable
$noWP_at6$	ϵ	Unobservable
WP_at7	WP_at7	Observable
$noWP_at7$	ϵ	Unobservable
mv_Conv1	mv_Conv1	Observable
umv_Conv1	ϵ	Unobservable
mv_Conv2	ϵ	Unobservable
umv_Conv2	ϵ	Unobservable
mv_Conv3	ϵ	Unobservable
umv_Conv3	ϵ	Unobservable
mv_Conv4	ϵ	Unobservable
umv_Conv4	ϵ	Unobservable

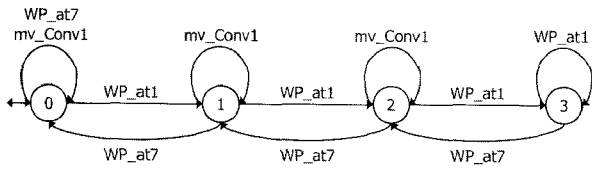


그림 11. 오토마톤 $P(K_1^0)$.

Fig. 11. Automaton of $P(K_1^0)$.



그림 12. 오토마톤 $P(G_{sub,1})$

Fig. 12. Automaton of $P(G_{sub,1})$

제어성을 만족한다. 그리고 $P(K_1^0)$ 은 비막힘이고 $P(G_{sub,1})$ 과 비충돌이다. 또 $P(K_1^0)$ 의 $P(G_{sub,1})$ 에 대한 최대 제어가능 부분언어는 $P(K_1^0)$ 자신이 된다. 따라서 $S_1 = P(K_1^0)$ 이다.

단계 6. $P^{-1}\{P(K_1^0)\}$ 을 구하면 (상태:4, 천이:17)이다.

단계 7.

$L_m(G_{sub,1}) \wedge L_m[P^{-1}\{P(S_1)\}] = L(G_{sub,1}) \wedge L[P^{-1}\{P(S_1)\}]$ 이므로 $[P^{-1}\{P(S_1)\}]$ 은 $G_{sub,1}$ 와 비충돌이다.

단계 8. 따라서 S_1 이 $\{K, P, \Sigma_{uc}, \Sigma_{nc}\}$ 에 대한 $G_{sub,1}$ 의 감독제어기가 되고 $G_{sub,1} \wedge P^{-1}\{P(S_1)\}$ 이 감독제어기 S_1 아래에서의 플랜트 거동이 된다. 오토마톤 $G_{sub,1} \wedge P^{-1}\{P(S_1)\}$ 은 (상태:2304, 천이:174722)다.

6. 감독제어기 비교

그림 5와 그림 11에서 보듯이 상위레벨 감독제어기와 부분관측 하의 감독제어기는 상태와 자기순환 이벤트를 제외한 상태 천이가 같다. 그리고 두 감독제어기 모두 상태에서 이벤트 mv_Conv1 을 차단하므로 제어 데이터도 같다. 또 자기순환 이벤트도 상위레벨 이벤트의 의미를 고려하면 똑같이 일어난다고 할 수 있다. 즉, 상태가 직접 바뀌는 이벤트를 제외하고는 모두 자기순환 이벤트이므로 실제적으로 두 감독제어기는 같다고 할 수 있다. 하지만 그림 13과 14에서 보듯이 두 감독제어기 아래에서의 플랜트 거동은 서로 다르

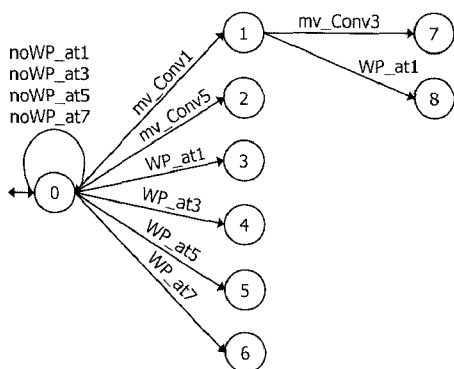


그림 13. $G_{sub,1} \wedge P^{-1}\{P(S_1)\}$ 의 일부분.

Fig. 13. Part of $G_{sub,1} \wedge P^{-1}\{P(S_1)\}$.

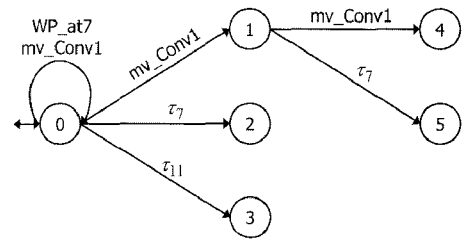


그림 14. $G_{hi} \wedge K_{hi,1}^1$ 의 일부분.

Fig. 14. Part of $G_{hi} \wedge K_{hi,1}^1$.

게 나타난다.

그림 13에서 알 수 있듯이 $G_{sub,1} \wedge P^{-1}\{P(S_1)\}$ 에서는 가관측성을 만족하기 위해 모든 관측불가능 이벤트에 대해서 상태가 바뀌게 된다. 그렇지만 $G_{hi} \wedge K_{hi,1}^1$ 에서는 τ_7, τ_{11} 과 같이 SOCC에 영향을 주는 이벤트에 대해서만 상태가 바뀌게 된다. 즉, 플랜트에 대한 감독제어기의 작용은 같지만 감독제어기 감독제어 아래에서의 플랜트 거동은 부분관측 감독제어기에 대한 플랜트 거동 $G_{sub,1} \wedge P^{-1}\{P(S_1)\}$ 이 더 복잡하게 일어난다. 또, 계층적 일관성의 계산복잡도보다 가관측성의 계산복잡도가 훨씬 복잡하다. 왜냐하면 본 예제에서는 제어가능 이벤트보다 관측불가능 이벤트의 개수가 더욱 많기 때문이다.

V. 결론

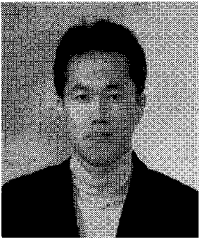
본 논문에서는 가제어성 검사의 계산복잡도를 줄이기 위해 서브플랜트의 개념을 정의하였다. 그리고 분산 감독제어 시스템의 구성을 위해 서브플랜트의 개념으로 상위레벨 감독제어기 설계 방법과 부분관측 하의 감독제어기 설계 방법을 제안하였다. 또한 CIM 시스템을 모사하는 CIM testbed에 대해서 본문에서 제시한 방법에 따라 상위레벨 감독제어기를 설계하였다. 그리고 상위레벨 이벤트에 속하지 않는 하위레벨 이벤트들을 관측불가능 이벤트로 두어 부분관측 하의 감독제어기도 설계하여, 두 감독제어기의 각 상태에서 자기순환 이벤트를 제외한 상태 천이와 제어 데이터가 같음을 확인하였다. 하지만 상위레벨 감독제어기와 부분관측 하의 감독제어기에 대한 플랜트 거동은 각각 다르게 나타남을 확인하고 이유를 분석하였다.

제어가능 이벤트보다 관측불가능 이벤트의 개수가 많은 경우 가관측성의 계산복잡도가 계층적 일관성의 계산복잡도보다 크기 때문에 상위레벨 감독제어기를 설계하는 것이 간단하다. 그리고 반대의 경우에는 부분관측 감독제어기를 설계하는 것이 상대적으로 간단하고 상위레벨 감독제어기보다 감독제어 아래에서의 플랜트 거동을 상세하게 나타낸다.

추후 연구과제는 임의의 이산사건 시스템에 대해서 본 논문의 결과를 설명할 수 있는 정리를 만드는 것이다. 즉, 상위레벨 감독제어기와 상위레벨 이벤트에 속하지 않는 하위레벨 이벤트를 관측불가능 이벤트로 정의한 경우 설계한 부분관측 감독제어기의 제어특성과 이들의 감독제어 아래에서의 플랜트 거동 차이를 일반적으로 설명하는 정리를 만들어 내야 할 것이다.

참고문헌

- [1] P. J. Ramadge and W. M. Wonham, "Supervisory control of a class of discrete event process," *SIAM Journal of Control and Optimization*, vol. 25, no. 1, pp. 206-230, 1987.
- [2] W. M. Wonham and P. J. Ramadge, "On the supremal controllable sublanguage of a given language," *SIAM Journal of Control and Optimization*, vol. 25, no. 3, pp. 637-659, 1987.
- [3] C. G. Cassandara and S. Lafortune, *Introduction to Discrete Event Systems*, Kluwer Academic Publishers, 1999.
- [4] R. J. Ramadge and W. M. Wonham, "The control of discrete event systems," *Proceedings of IEEE*, vol. 7, pp. 81-98, January 1989.
- [5] W. M. Wonham, "Notes on control of discrete event systems," Department of Electrical and Computer Engineering, University of Toronto, 1998.
- [6] W. M. Wonham and P. J. Ramadge, "Modular supervisory control of discrete event systems," *Mathematics of Control Signals and Systems*, vol. 1, no. 1, pp. 13-30, February 1988.
- [7] F. Lin and W. M. Wonham, "Decentralized supervisory control of discrete event systems," *International Journal of Information Sciences*, vol. 44, pp. 199-224, April 1988.
- [8] K. Rudie and W. M. Wonham, "Think globally, act locally: decentralized supervisory control," *IEEE Transactions on Automatic Control*, vol. 37, no. 11, pp. 1692-1708, November 1992.
- [9] 손형일, 이석, "CIM Testbed의 제어를 위한 Supervisor의 설계와 구현," 제어·자동화·시스템공학 논문지, 제 6 권, 제 6 호, pp. 478-485, June 2000.
- [10] 손형일, 김기웅, 이석, "이산이벤트시스템의 고장진단," 제어·자동화·시스템공학 논문지, 제 7 권, 제 5 호, pp. 375-383, May 2001.



손형일

1998년 부산대학교 생산기계공학과(공학사). 2000년 부산대학교 지능기계공학과 (공학석사). 2000년 ~ 현재 KAIST 기계공학과 박사과정 재학 중. 2003년 ~ 2005년 LG전자 생산기술원 선임연구원. 2005년 ~ 현재 삼성전자 LCD총괄 책임연구원. 관심분야는 햅틱, 로봇 원격조작, 수술로봇, 이산 이벤트/하이브리드 시스템.



이 석

1961년 12월 11일 생. 1984년 서울대학교 기계공학과 졸업. 펜실바니아 주립대학교 석사(1985년), 동대학원 박사(1990년). 1990년~1993년 신시내티 대학교 기계공학과 조교수. 1993년~현재 부산대학교 기계공학부 교수. 관심분야는 산업용 네트워크, 차량용 네트워크, 홈 네트워크, 센서 네트워크, 이산 이벤트 시스템.