특별연재 이산사건시스템의 관리제어(Ⅲ)

분산 관리제어

김 성 규*, 조 광 현, 임 종 태***

*한국과학기술원 전기 및 전자공학부, **울산대학교 전기전자 및 자동화공학부

Abstract : 본 논문에서는 관리제어시스템의 대상인 이산사건시스템을 하나의 전역 관리제어기(global supervisor)가 아닌 여러 개의 국소 관리제어기(local supervisor)들의 동시수행(concurrent action)을 통한 분산관리제어에 대해 알아본다. 반적으로 관리제어 대상시스템 G의 사양에 대해 관리제어기가 설계가능하면 하나의 전역 관리제어기(S)를 설계하여 관리제어시스템의 언어 $L(S/G)$를 생성할 수 있다. 이에 반하여, 분산 관리제어이론은 주어진 사양(specification)에 대해 여러 개의 국소 관리제어기(S_i)들의 동시수행($\wedge S_i$)을 통한 언어 $L(\wedge S_i/G)$를 생성함으로써 전역 관리제어기와 동등한 역할 즉, $L(S/G)=L(\wedge S_i/G)$을 이루고자 하는 데 그 목적이 있다. 리제어 대상시스템 G의 발생가능한 사건들에 관해 제어가능성과 관측가능성을 고려하여 모든 사건들이 관측가능할 경우와 관측불가능한 사건들이 존재하는 경우로 구분하여 분산 관리제어시스템의 설계에 대해 알아본다. 또한 관측불가능한 사건들이 존재할 때 사양언어(specificatin language) K가 여러 개의 국소사양(local specification)들의 교집합으로 주어진 경우의 국소 분산 관리제어와 하나의 전역사양(global specification)으로 주어진 경우의 전역 분산 관리제어로 세분하여 각각의 분산 관리제어결과가 전역 관리제어결과와 서로 같아질 수 있는 조건에 대해 알아본다.

1. 서론

일반적으로, 관리제어 대상시스템 G의 사양으로 주어진 언어 K의 제어가능성 및 관측가능성 등에 따라서 대상 제어시스템에 대한 적절한 관리제어시스템 언어 $L(S/G)$를 발생시킬 수 있는 하나의 전역 관리제어기(S)를 설계할 수 있다[8], [9]. 본 논문에서는 분산제어 관점에서 기존 관리제어이론의 개념을 확장한다. 분산제어는 "분할을 통한 해결(divide-and-conquer)" 관점의 접근을 통해 문제의 높은 복잡성을 낮은 복잡성을 가지는 복수의 단위 문제로 전환함으로써 주어진 문제의 계산상의 복잡도를 줄일 수 있다는 이점이 있다. 또한 단위 구성모델을 기초로 전체시스템을 설계하므로 설계의 유연성이 생긴다.

관리제어 대상시스템 G의 발생가능한 사건들의 제어가능성과 관측가능성을 고려하여 모든 사건들이 관측가능할 경우와 관측불가능한 사건들로 구분하여 분산 관리제어시스템의 설계에 대해 알아본다. 또한 관측불가능한 사건들이 존재할 때 주어진 사양언어 K가 여러 개의 국소사양(local specification)들의 교집합으로 주어진 경우와 하나의 전역사양(global specifica- tion)으로 주어진 경우로 세분하여 국소 분산 관리제어와 전역 분산 관리제어의 결과가 같아질 수 있는 조건 즉, $L(S/G)=L(\wedge S_i/G)$을 만족시킬 수 있는 조건에 대해 알아본다.

$L(S/G)$의 부언어(sublanguage) $K\subset L(G)$가 제어목적인 사양언어로 주어졌을 때 K의 제어가능성 관점에서 K의 최고 제어가능 부언어(supremal controllable sublanguage of K) $K^{\uparrow}$를 찾을 수 있고 K가 정규언어(regular language)인 경우 주어진 K의 일반적인 부언어들 중에서 최고 제어가능 부언어인 $K^{\uparrow}$를 찾을 수 있는 알고리즘이 Wonham과 Ramadge에 의해 제안되었다[7]. 이산사건시스템의 분산 관리제어의 기초를 확립한 Wonham과 Ramadge[1]는 모든 사건들이 관측가능하다는 가정하에 관리제어 대상시스템 G의 사양언어 K가 본질적으로 여러 개의 국소사양들의 교집합으로 주어진 문제를 제시하고, 각각의 국소사양을 만족하는 국소 관리제어기를 설계하여 이들의 동시수행을 통해 사양언어 K를 만족시키는 국소 분산 관리제어이론 및 여러 특성들을 연구하였다.

언어 K의 제어가능성과 관측가능성을 동시에 고려한 분산 관리제어이론은 이전의 결과를 더욱 확장하였다. Lin과 Wonham[2]은 시스템 G의 발생가능 사건들의 제어가능성 외에 관측가능성을 고려하여 관측불가능한 사건들이 존재하는 경우 국소적 제어가능성(locally controllability) 조건을 통한 국소 분산 관리제어 측면에서 접근하여 이전 Wonham과 Ramadge[1]의 결과를 확장하고, 그 예로써 생산시스템과 관련된 문제는 국소 분산 관리제어 기법이 적합함을 보였다. 이에 반하여 Rudie와 Wonham[3]는 관측불가능한 사건들이 존재하는 경우는 동일하지만 사양언어 K가 국소사양들의 교집합으로 주어지지 않고 하나의 전역사양으로 주어진 문제를 제시하고, 상호 관측가능성(coobservability) 조건을 통한 전역 분산 관리제어 측면에서 접근하여 통신 네트

워 프로토콜과 관련된 문제는 국소 분산 관리제어 기법보다는 전역 분산 관리제어 기법이 적합함을 보여주었다. 그리고 [11]에서는 모든 국소사양들이 국소 분산관리제어이론의 조건들을 만족시키지 못하는 경우 국소 관리제어와 전역 관리제어의 혼합된 제어구조를 다루었다.

본 논문의 구성은 다음과 같다. 2절에서는 플랜트에서 발생하는 모든 사건들이 관측가능한 경우 분산 관리제어기 설계에 대해 다루고, 3절에서는 발생하는 사건들 중 관리제어기가 관측불가능한 사건들이 존재하는 경우로 확대 해석하여 분산 관리제어기의 설계에 관해 다룬다. 구체적으로는 시스템 G의 사양으로 주어진 언어 K가 근본적으로 국소적인 여러 사양으로 세분된 경우와 하나의 전역사양으로 주어진 경우로 세분하고, 각 경우에 대한 분산 국소 관리제어와 전체 전역 관리제어로 구분하여 관리제어기를 설계한다. 그리고 4절에서는 결론을 맺는다.

2. 분산 관리제어 유형 I : 모든 사건들이 관측가능한 경우

이 절에서는 모든 사건들이 관측가능한 경우 관리제어 대상시스템 G의 사양언어 K가 본질적으로 여러 개의 국소사양들의 교집합으로 주어진 문제에 대한 관리제어기 설계문제를 알아 본다[1].

오토마타 이론(automata theory)에 따라 제어 대상시스템 G를 다음의 5-tuple 오토마타 G={Σ, Q, δ, } $G=(\Sigma,\ Q,\ \delta,\ q_o,\ Q_m)$로 생각하자. 여기서 Q 는 상태집합, Σ 는 유한 사건집합, q_o 는 초기상태, $Q_m \subseteq Q$ 는 표기상태집합, $\delta:\ \Sigma^* \times Q \rightarrow Q$ 은 상태 천이 함수이다. 그리고, $L(G) = \{\ s:\ s \in \Sigma^*$ 이고 $\delta(s,q_o) \in G\}$, $L_m(G) = \{s : s \in \Sigma^*$ 이고 $\delta(s,q_o) \in G_m\}$이며 사건집합 $\Sigma = \Sigma_c \cup \Sigma_{uc}$는 제어가능 사건집합 Σ_c과 제어불가능 사건집합 Σ_{uc}로 구성되어 있다고 하자. 일반적으로 $L(G)$ 는 닫힌언어(closed language)이며, $\overline{L_m(G)} = L(G)$라고 가정한다.

관리제어기 S는 $S=(R,\phi)$로 구성되는데, 여기서 $R=\{\ \Sigma,\ X,\ \zeta,\ x_o, X_m\ \}$은 상태 천이함수 $\zeta: \Sigma \times X \rightarrow X$와 $\sigma \in \Sigma_{uc}$, $x \in X$이면 $\phi(\sigma,x)=1$이고 $\sigma \in \Sigma_c$, $x \in X$이면 $\phi(\sigma,x) \in \{0,1\}$로 정의되는 제어사상(control map) $\phi : \Sigma \times X \rightarrow \{0,1\}$로 구성되는 인식기이다. 또한, 관리제어기가 결합된 폐루프 관리제어시스템 S/G는 $\phi(\sigma,x)=1$이면 $(\zeta(\sigma,x), \delta(\sigma,x))$이고 $\phi(\sigma,x)=0$이면 정의되지 않는 폐루프 상태 천이함수 $(\zeta \times \delta)^{\phi}$: $\Sigma \times X \times Q \rightarrow X \times Q$를 구성요소로 하는 $S/G=(\Sigma, X \times Q,\ (\zeta \times \delta)^{\phi},\ (x_o, q_o),\ X_m \times Q_m)$로 정의되는데 그 역시 오토마톤이다.

분산 관리제어를 위해 다음 두 개의 관리제어기 $S=(A,\Phi)$와 $T=(B,\Psi)$를 생각하자. 여기서 $A=(\Sigma, X, \zeta, x_o)$, $B=(\Sigma, Y, \eta, y_o)$라고 할 때, S와 T의 결합(conjunction) $S \wedge T = (A \times B,\ \Phi * \Psi)$는 다음과 같이 정의된다.

$$
\begin{aligned}
A \times B &= (\Sigma, X \times Y, \zeta \times \eta, (x_o, y_o)),\\
\zeta \times \eta(\sigma,x,y) &= \begin{cases} \zeta(\sigma,x) \text{ 와 } \eta(\sigma,y) \text{ 모두 정의되면} \\ (\zeta(\sigma,x), \eta(\sigma,y)) \text{ 이고} \\ \text{그 외 경우이면 정의되지 않음,} \end{cases}\\
\Phi * \Psi(x,y) &= \Phi(x) \wedge \Psi(y).
\end{aligned}
$$

$S \wedge T$는 제어가능한 사건들 중 Φ와 Ψ에 의해서 발생이 허용된 사건들의 교집합에 의한 제어사상 $\Phi * \Psi$과 함께 병렬합성으로 동작하는 오토마톤 S와 T로 구성된다. 다음 명제는 $S \wedge T$의 G에 대한 완전함과 $S \wedge T/G$의 S/T와 T/G 에 대한 폐루프 언어상의 관계를 나타낸다.

명제 2.1[1] : 위에서 정의된 S, T 는 $S \wedge T$는 다음을 만족한다.

(1) $L(S \wedge T/G) = L(S/G) \cap L(T/G)$,
(2) $L_m(S \wedge T/G) = L_m(S/G) \cap L_m(T/G)$,
(3) $S \wedge T$ 는 G에 대해서 완전한(complete) 제어기이다[7].

유의할 사항은 언어의 논리곱(conjunction)은 비막힘성(nonblocking)에 대해 닫혀 있지 않다는 것이다. 즉, S와 T가 비막힘성의 특성을 갖고 있더라도, 관리제어기 $S \wedge T$ 가 비막힘성을 유지할 수 있다고는 볼 수 없다. 예로써, 만약 비막힘성인 두 관리제어기가 시스템의 어느 상태에서 서로 모순된 제어를 허용하고 있다면 결합에 의한 폐루프 시스템은 막힘성(blocking) 특성을 보일 수도 있다. 분산 관리제어의 비막힘성 특성을 위해 다음 비충돌성(nonconflicting)을 정의한다.

두 언어 $K_1, K_2 \in \Sigma^*$에 대해서 $\overline{K_1 \cap K_2} = \overline{K_1} \cap \overline{K_2}$의 조건을 만족된다면 두 언어는 비충돌적(nonconflicting)이라고 정의한다. 여기서, $\overline{K_1 \cap K_2} \subseteq \overline{K_1} \cap \overline{K_2}$는 자명한 사실이므로, 두 언어 $K_1, K_2 \in \Sigma^*$가 $\overline{K_1 \cap K_2} \supseteq \overline{K_1} \cap \overline{K_2}$를 만족한다면 비충돌성은 만족된다. 예로써, 임의의 닫힌 두 언어는 비충돌성이다. 다음 명제는 $S \wedge T$의 비막힘성은 비충돌성 언어의 관점에서 특징지워질 수 있음을 나타낸다.

명제 2.2[1] : S와 T가 G에 대해 비막힘성 관리제어기라고 하자. 두 언어 $L_m(S/G)$와 $L_m(T/G)$가 비충돌성이면 관리제어기 $S \wedge T$는 비막힘성이며, 그 역도 성립한다.

원하는 제어 목적이 $K = \bigcap_{i \in I} K_i$와 같이 주어진 경우, 다음 명제에서 알 수 있듯이 K의 제어가능성은 K_i언어들로부터 결정될 수 있음을 알 수 있다.

명제 2.3[1] : $K_1, K_2 \in \Sigma^*$라 하자. K_1과 K_2가 비충

돌성이고, 두 언어 모두 제어가능일 때, $K_1 \cap K_2$는 제어가능이다.

위 명제에서 알 수 있듯이 K_1과 K_2의 제어가능성 만족만으로는 $K_1 \cap K_2$의 제어가능성이 만족되지 않는다. 상대적 개념으로써, K_1과 K_2가 충돌성이지만 $K_1 \cap K_2$가 제어가능일 수도 있다. 만약 K_1과 K_2가 닫힌언어라면 위 명제는 항상 성립한다. 이러한 성질을 토대로 다음 관리제어기 설계문제 1을 생각해보자.

관리제어기 설계문제 1 : 공집합이 아닌(nonempty) 닫힌언어 $K_1, K_2 \subseteq L(G)$에 대해서, $L(S/G) = K_1 \cap K_2$를 만족하는 관리제어기 S를 설계하라.

언어 $K_1 \cap K_2$는 공집합이 아닌 닫힌언어이므로 $K_1 \cap K_2$가 제어가능이면 이를 만족하는 관리제어기를 설계할 수 있으며 그 역도 성립한다. K_1, K_2가 닫힌언어이면 비충돌성은 당연히 성립하며 또한, 명제 2.3으로부터 K_1, K_2가 각각 제어가능하면 $K_1 \cap K_2$ 역시 제어가능함으로 주어진 설계문제 1을 만족하는 관리제어기 S를 설계할 수 있다. 즉, K_1, K_2가 각각 제어가능이면 $L(S_1/G) = K_1$, $L(S_2/G) = K_2$를 만족하는 국소 관리제어기 S_1, S_2를 설계할 수 있으며 명제 2.1로부터 $L(S_1 \wedge S_2/G) = K_1 \cap K_2$를 유추할 수 있다.

위 문제에서 분산 관리제어와 전역 관리제어의 계산상의 복잡도(computational complexity)를 비교해 보자. G가 유한 상태(finite state) 시스템이고 언어 K_1, K_2가 정규 언어(regular language)이며 K_1, K_2 각각의 인식기(recognizer)의 오토마타의 상태 수가 n, 시스템 G의 상태 수가 m을 넘지 않는 다고 가정하자. 그러면 n개의 상태 수를 갖는 오토마타로 표현되는 정규 언어 K의 제어가능성 즉, $\overline{K}\Sigma_u \cap L(G) \subseteq \overline{K}$의 확인에 요구되는 계산상의 복잡도는 $O(nm)$이다. 또한, $K_1 \cap K_2$의 제어가능성의 확인에 요구되는 계산상의 복잡도는 $O(n^2)$이다. 따라서, $L(S/G) = K_1 \cap K_2$을 만족하는 관리제어기 설계에 요구되는 계산상의 복잡도는 $O(n^2 m)$이다. 그러나, K_1, K_2의 제어가능성 즉, $L(S_1/G) = K_1$, $L(S_2/G) = K_2$을 만족하는 관리제어기 S_1, S_2의 설계에 요구되는 계산상의 복잡도는 $O(nm)$이며 이 값은 가능한 최대 값이다. 결국, 분산 관리제어 기법에 의한 관리제어기 설계 계산상의 복잡도($O(nm)$)와 전역 관리제어 기법에 의한 관리제어기 설계 계산상의 복잡도($O(n^2 m)$)의 비교에서 알 수 있듯이 분산 관리제어가 가능한 시스템의 경우 관리제어기 설계의 계산상의 복잡도에서 분산 관리제어에 큰 이점이 있음을 알 수 있다.

관리제어기 설계문제 2 : $K_1 \cap K_2 \neq \emptyset$을 만족하는 언어 $K_1, K_2 \subseteq L_m(G)$ 대해서, $L_m(S/G) = K_1 \cap K_2$를 만족하는 비막힘성 관리제어기 S를 설계하라.

위 설계문제를 풀기 위해 다음을 정의한다. 언어 $K \subseteq L_m(G)$가 $\overline{K} \cap L_m(G) = K$을 만족하면 $L_m(G)$-닫힘 이라고 한다. 명제 2.3에서는 $K_1 \cap K_2$의 제어가능성을 다루었지만, 다음 명제에서는 $K_1 \cap K_2$의 $L_m(G)$-닫힘에 대해 나타내고 있다.

명제 2.4[1] : K_1과 K_2가 $L_m(G)$-닫힘이면 $K_1 \cap K_2$도 $L_m(G)$-닫힘이다.

명제 2.3과 명제 2.4로부터 설계문제 2의 해인 다음 명제를 얻을 수 있다.

명제 2.5[1] : $K_1, K_2 \subseteq L_m(G)$ 이며 $K_1 \cap K_2 \neq \emptyset$라고 하자. K_1과 K_2가 비충돌성이고 각각 $L_m(G)$-닫힘이며 제어가능하다면 $L_m(S_1 \wedge S_2/G) = K_1 \cap K_2$를 만족하는 분산 관리제어기가 존재한다.

위 명제 2.5가 의미하는 바는 K_1과 K_2가 각각 $L_m(G)$-닫힘이며 제어가능하다면 $L_m(S_1/G) = K_1$ 와 $L_m(S_2/G) = K_2$를 만족하는 국소 관리제어기 S_1, S_2를 설계할 수 있고, K_1과 K_2가 비충돌성이면 $K_1 \cap K_2$가 제어가능하며 $S_1 \wedge S_2$가 비막힘성임을 나타낸다.

유의할 사항은 $K_1 \cap K_2$의 제어가능성이 언제나 $S_1 \wedge S_2$의 비막힘성을 의미한다는 것은 아니라는 점이다. 그러나 그 역은 언제나 성립한다. 즉, $S_1 \wedge S_2$이 비막힘성임이면 $K_1 \cap K_2$는 언제나 제어가능성이다. 결과적으로 $L_m(S_1/G) = K_1$, $L_m(S_2/G) = K_2$, $L_m(S/G) = K_1 \cap K_2$를 만족하는 언어 K_1, K_2 및 국소 관리제어기 S_1, S_2가 존재하더라도 분산 관리제어기 $S_1 \wedge S_2$는 막힘성이 될 수 있다. 즉, 분산 관리제어기 $S_1 \wedge S_2$의 제어역할이 전역제어기 S의 제어역할과 같지 않을 수 있다는 것이다.

명제 2.5의 분산제어와 전역제어의 계산상의 복잡도를 비교해 보자. 앞에서와 같이 시스템 G가 유한 상태(finite state)이고 언어 K_1, K_2가 정규언어이며 K_1, K_2 각각의 인식기(recognizer)의 상태 수가 n, 시스템 G의 상태 수가 m을 넘지 않는 다고 가정하자. 전역 관리제어 문제에서는 우선 $K_1 \cap K_2$를 계산해야하고, $K_1 \cap K_2$의 제어가능성을 확인해야 되므로 앞에서 알 수 있듯이 요구되는 계산상의 복잡도는 $O(n^2 m)$이다. 반면에 분산 관리제어 문제에서는 K_1, K_2의 제어가능성과 K_1과 K_2의 비충돌성을 각각 확인해야 함으로 전역관리제어기 설계에 요구되는 전체 계산상의 복잡도는 K_1, K_2의 제어가능성 계산상의 복잡도[각각 $O(nm)$]과 K_1과 K_2의 비충돌성 계산상의 복잡도[$K_1 \cap K_2$의 계산에 필요한 $O(n^2)$]을 합한 값인 $O(mn + n^2)$이 된다.

위에서 알 수 있듯이, 전역 및 분산 관리제어기 설계 모두 폐루프 시스템의 비막힘성 확인을 위해 $K_1 \cap K_2$

를 계산해야 하는 단계가 필요한데, $K_1 \cap K_2$의 계산에 요구되는 계산상의 복잡도는 K_1, K_2 각각의 상태 수(n)의 제곱에 비례해 증가한다. 따라서, 시스템 G의 상태 수(m)이 큰 경우 전체 계산상의 복잡도 비교에 있어서 전역 관리제어[$O(n^2m)$]에 대한 분산 관리제어 [$O(mn+n^2)$]의 이점을 알 수 있다.

주어진 언어 K이 제어불가능일때, 일반적인 적절한 방법은 K의 최고 제어가능 부언어(supremal controllable sublanguage)인 $K^{\uparrow}$를 K대신 사양으로 선택하는 것이고 제어가능 언어는 언어의 합집합(union)에 닫혀 있으므로 주어진 K의 부분 언어집합중 $K^{\uparrow}$의 존재는 증명가능하다[7].

따라서, 이런 경우 분산제어시스템과 관련하여 연산자 ↑와 언어 교집합(intersection)과의 상호 연관성에 대해 생각해 볼 수 있다.

정리 2.1[1] : $K_1, K_2 \subseteq \Sigma^*$인 최고 제어가능 언어 $K_1^{\uparrow}$과 $K_2^{\uparrow}$가 각각 비충돌성이면 $K_1^{\uparrow} \cap K_2^{\uparrow} = (K_1 \cap K_2)^{\uparrow}$이다.

정리 2.1은 $(K_1 \cap K_2)^{\uparrow}$의 계산에 있어서 $K_1^{\uparrow}$과 $K_2^{\uparrow}$를 먼저 계산하고 비충돌성을 확인함으로써 $(K_1 \cap K_2)^{\uparrow}$를 얻을 수 있음을 보여준다. 그리고, $K_1^{\uparrow}$과 $K_2^{\uparrow}$를 먼저 계산하는 과정에서 분산 관리제어의 적용 가능성의 용이함을 알 수 있다.

다음 보조정리는 K_1와 K_2가 닫힌 언어인 경우에 대한 위 정리 2.1의 확장이라고 할 수 있다.

보조정리 2.1[1] : $K_1, K_2 \subseteq \Sigma^*$인 사양언어 K_1와 K_2가 각각 닫힌언어이면 $K_1^{\uparrow} \cap K_2^{\uparrow} = (K_1 \cap K_2)^{\uparrow}$이다.

다음 명제는 언어의 닫힘성과 연산자 ↑의 상호 연관성을 나타낸다.

명제 2.6[1] :

(1) $K \subseteq \Sigma^*$가 닫힌언어이면 $K^{\uparrow}$도 닫힌언어이다.

(2) $K \subseteq L_m(G)$인 K이 $L_m(G)$-닫힘이면 $K^{\uparrow}$도 $L_m(G)$-닫힘이다.

관리제어기 설계문제 3 : 공집합이 아닌(nonempty) 닫힌언어 $K_1, K_2 \subseteq L(G)$ 대해서, $L(S/G)$가 $L(S/G) \subseteq K_1 \cap K_2$를 만족하는 최대언어가 되는 관리제어기 S를 설계하라(반드시 비막힘성일 필요는 없음).

위 조건을 만족하는 가장 큰 제어가능언어는 $(K_1 \cap K_2)^{\uparrow}$이다. 만약 $(K_1 \cap K_2)^{\uparrow} \neq \varnothing$이면 보조정리 2.1과 명제 2.6으로부터 $K_1^{\uparrow}, K_2^{\uparrow}$ 모두 공집합이 아닌 닫힌 언어임을 알 수 있다. 따라서, $L(S_1/G) = K_1^{\uparrow}$, $L(S_2/G) = K_2^{\uparrow}$를 만족하는 관리제어기 S_1과 S_2를 설계 할 수 있으며 명제 2.1과 보조정리 2.1로부터 $L(S_1 \wedge S_2/G) = K_1^{\uparrow} \cap K_2^{\uparrow} = (K_1 \cap K_2)^{\uparrow}$임을 알 수 있다.

위 문제에서 분산 관리제어와 전역 관리제어의 계산상의 복잡도를 비교해 보자. 우선 앞에서와 같이 시스템 G가 유한 상태(finite state)이고 언어 K_1, K_2가 정규언어이며 K_1, K_2 각각 인식기(recognizer)의 상태 개수가 n, 시스템 G의 상태 개수가 m을 넘지 않는 다고 가정하자. 전역 관리제어기인 경우 $L(S/G) = (K_1 \cap K_2)^{\uparrow}$를 만족하는 S를 설계할 때 $O(n^2m)$의 계산상의 복잡도가 요구되는 반면, 분산 관리제어인 경우는 $K_1^{\uparrow}$, $K_2^{\uparrow}$를 계산할 때 각각 $O(nm)$의 계산상의 복잡도가 요구될 뿐이므로 분산 관리제어 전체 요구되는 계산상의 복잡도는 $O(nm)$이다.

관리제어기 설계문제 4 : $K_1 \cap K_2 \neq \varnothing$을 만족하는 주어진 $K_1, K_2 \subseteq L_m(G)$ 대해, $L_m(S/G) = K_1 \cap K_2$를 만족하는 비막힘성 관리제어기 S를 설계하라.

위 문제는 $K_1 \cap K_2$에 속한 언어들 중 가장 큰 $L_m(G)$-닫힌 언어이면서 제어가능한 언어가 비지않은 경우이면 필요충분조건으로 해결될 수 있다. 만약 K_1과 K_2가 $L_m(G)$에 닫혀있다면 명제 2.4와 명제 2.6에 의해서 위의 설계문제 4는 $(K_1 \cap K_2)^{\uparrow}$가 비지않은 언어이면 필요충분조건으로 해결될 수 있다. 정리 2.1에 의해서 $K_1^{\uparrow}$과 $K_2^{\uparrow}$가 비충돌성이면 $(K_1 \cap K_2)^{\uparrow} = K_1^{\uparrow} \cap K_2^{\uparrow}$임을 알 수 있다. 이 경우 위 설계문제 4는 $K_1^{\uparrow}$과 $K_2^{\uparrow}$를 기본 구성언어로 하는 설계문제 2로 전환될 수 있으며 다음 명제 2.7을 유추할 수 있다.

명제 2.7[1] : 다음 3가지 조건이 만족되면 관리제어 설계문제 4를 만족하는 관리
제어기 S를 설계할 수 있다.

(1) K_1과 K_2가 $L_m(G)$에 닫혀있다.

(2) $(K_1 \cap K_2)^{\uparrow} \neq \varnothing$이다.

(3) $K_1^{\uparrow}$과 $K_2^{\uparrow}$가 비충돌성이다.

3. 분산 관리제어 유형 II : 관측불가능한 사건들이 존재하는 경우

3-1. 국소 분산 관리제어

사양으로 주어진 언어 K가 여러 개의 국소사양들의 교집합으로 주어진 경우에 대한 분산 관리제어이론을 알아본다.

3-1-1. 투영사상 및 표준 부언어

$\varnothing \neq \Sigma_{loc} \subset \Sigma$ 에 대해서 $\Sigma_{u,loc} = \Sigma_{loc} \cap \Sigma_u$, $\Sigma_{c,loc} = \Sigma_{loc} \cap \Sigma_c$라고 하고 사건집합 Σ_{loc}을 국소 관리제어기에 의해 관측가능한 $\sigma \in \Sigma_{loc}$인 국소 부분 사건집합으로 가정하자. 관측가능성을 구체적으로 정형화하기

위해 [$P\varepsilon=\varepsilon$], [$\sigma\in\Sigma-\Sigma_{loc}$이면 $P\sigma=\varepsilon$], [$\sigma\in\Sigma_{loc}$이면 $P\sigma=\sigma$], [$s\in\Sigma^*$, $\sigma\in\Sigma$이면 $P(s\sigma)=(Ps)(P\sigma)$]인 투영사상 $P:\Sigma^*\to\Sigma^*_{loc}$을 정의하자.

G의 언어 $L(G)$중 관측불가능한 사건들에 의한 부분적인 국소관점(local sense)에서 관측된 동작을 $PL(G)$라고 하고, $PL(G)$의 인식기(recognizer)를 G_{loc} 라고 하자. 즉, G_{loc}은 국소 관리제어기의 관점에서 보이는 시스템 G라 할 수 있다.

일반적으로 G는 부분 모델들인 $G_1,\cdots,G_N$들의 병렬합성(parallel composition)으로 표현되므로 G를 병렬합성 모델 $G=\|_{i=1}^{N}G_i$ 이라고 정의하자. 그러면, $L(G)=\|_{i=1}^{N}L(G_i)$, $PL(G)=\|_{i=1}^{N}PL(G_i)$로 확장할 수 있다.

임의의 $A\subset L(G)$를 $L(G)$의 부언어라고 할 때, A는 국소적으로 $PA\subset\Sigma^*_{loc}$와 같이 관측된다. $P^{-1}(\cdot)$를 P의 역투영사상(inverse projection)이라고 할때 주어진 언어 A가 $A=L(G)\cap P^{-1}(PA)$를 만족할 때 A는 표준언어(normal language)라고 한다. A가 임의의 부언어일 때 $A\subset L(G)\cap P^{-1}(PA)$는 자명하다. 따라서, A가 국소 제어가능이 되기 위해서는 $A\supset L(G)\cap P^{-1}(PA)$이 만족되어야 한다. 즉, 임의의 어떤 $s\in A$에 대해서 $t\in L(G)$ 그리고 $Pt=Ps$ 이면 반드시 $t\in A$도 성립함을 의미한다.

3-1-2. 국소/전역 최고 제어가능 부언어

공집합이 아닌 닫힌 부언어 $E_{loc}\subset\Sigma^*_{loc}$를 국소적 관점에서의 제어사양이라고 하면, 전역적 관점에서는 $L(G)\cap P^{-1}E_{loc}$와 같이 관측된다. 따라서, 전역 관리제어기(global supervisor) S는 $K:=(L(G)\cap P^{-1}E_{loc})^{\uparrow}\subset\Sigma^*$가 만족되도록 설계될 것이다. 그러나, 국소 관리제어기(local supervisor) S_{loc}은 $E_{loc}\subset\Sigma^*_{loc}$만을 관측하므로 $K_{loc}:=(PL(G)\cap E_{loc})^{\uparrow}_{loc}\subset\Sigma^*_{loc}$가 만족되도록 설계될 것이다. 그리고, 국소 관리제어기 S_{loc}은 $\sigma\in\Sigma_c-\Sigma_{c,loc}$인 사건들에 대해 언제나 발생을 허용(enable)한다는 기본 제어법칙(default control rule)에 따라 동작한다고 가정한다면 국소 관리제어기 S_{loc}은 $\tilde{K}:=L(G)\cap P^{-1}K_{loc}\subset\Sigma^*$를 만족하는 전역 관리제어기인 $\tilde{S}$로 확장될 수 있다.

결국, 분산 관리제어의 문제는 $K=\tilde{K}$을 만족하는가에 관한 문제가 된다. 즉, 투영사상에 의해서만 사건들을 관측할 수 있는 국소 관리제어기가 $\tilde{S}$가 기본 제어 법칙(default control rule)에 따라 동작한다고 가정할 때 모든 사건들을 관측할 수 있는 전역 관리제어기 S와 같은 역할을 할 수 있는가 하는 문제로 귀착된다.

명제 3.1[2] : $K\supseteq\tilde{K}$.

명제 3.1은 국소 관리제어기($\tilde{S}$)가 전역 관리제어기(S)보다 더 효과적인 제어를 할 수 없다는 것을 나타낸다. 즉, 국소 관리제어의 전역 관리제어에 대한 한계를 나타내고 있다.

명제 3.2[2] : $PL(G)\cap E_{loc}$이 국소 제어가능이면 $K=\tilde{K}$ 이다.

위 명제는 주어진 국소사양 E_{loc}이 전체 시스템 관점이 아닌 투영사상에 의한 관측가능한 언어들만으로 구성된 투영된 관점에서의 시스템 $PL(G)$에 대해서 제어가능성이 만족되는 경우 $K=\tilde{K}$인 결과를 나타내고 있다.

정리 3.1[2] : K가 표준(normal)이면 $K=\tilde{K}$ 이며 그 역도 성립한다.

주어진 사양언어 K가 표준조건을 만족할 때 $PL(G)$상에서의 국소제어에 의한 결과가 기본 제어법칙에 따른 $L(G)$상에서의 전역제어에 의한 결과와 언제나 같음을 그 정의로부터 유추할 수 있다. 따라서, 사양언어 K가 표준(normal)이면 위 정리 3.1의 결과 $K=\tilde{K}$를 유추할 수 있다.

보조정리 3.1[2] : $K=\tilde{K}$ 이면 $K(\Sigma-\Sigma_{c,loc})\cap L(G)\subset K$이다.

보조정리 3.1은 K를 구현함에 있어서, $\sigma\in\Sigma_c-\Sigma_{c,loc}$인 사건들은 발생억제(disable)될 필요성이 없음을 나타낸다.

정의 3.1 [주요조건 (MC : Main Condition)] : (모든 w,σ에 대해서) $w\in K_{loc}$, $\sigma\in\Sigma_{c,loc}$, $w\sigma\in PL(G)$, $w\sigma\notin K_{loc}$이면 ($s\in K$, $Ps=w$, $s\sigma\in L(G)$를 만족하는 모든 s에 대해서) $s\sigma\notin K$이다.

MC의 의미는 국소 관리제어기 S_{loc}에 의해서 발생억제(disable)된 $\sigma\in\Sigma_{c,loc}$ 사건은 전역 관리제어기 S에 의해서도 반드시 발생억제가 되어야 한다는 것이다. 즉, 전역 관리제어기 S에 이용가능한 전역적 정보(global information)는 국소 관리제어기 $\tilde{S}$에 의한 제어결과보다 더 큰 결과가 나오지 않는 범위 내에서 이용 가능해야 함을 의미한다. 따라서 다음 정리 3.2를 유추할 수 있다.

정리 3.2[2] : $K_{loc}\neq\emptyset$라고 하자. MC가 만족되면 $K=\tilde{K}$ 이며 그 역도 성립한다.

보조정리 3.1[2] : $M\subset L(G)$이 닫힌 언어이고 $K:=M^{\uparrow}$, $s\in K$, $s\sigma\in L(G)$ 라고 하자. $s\sigma\notin K$이면 $s\sigma l\in L(G)-M$인 $l\in\Sigma^*_u$가 존재하며, 그 역도 성립한다.

보조정리 3.1은 $s\in K$ 다음에 발생가능한 사건 σ가 발생억제 될 때는 그 사건의 발생을 허용함으로 $s\sigma$다음에 발생가능한 문자열들이 사양 M을 벗어나는 제어불가능한 문자열일 때만 가능함을 나타낸다.

3-1-3. 분산 관리제어

공집합이 아닌 국소 사건집합 $\Sigma_i \subset \Sigma$ $(i \in I)$이 주어졌다고 가정하고, 그 때의 해당 투영사상을 $P_i : \Sigma^* \rightarrow \Sigma_i^*$라고 하자. 또한, 시스템 G의 전역사양 E가 다음과 같이 국소사양($E_i \subset \Sigma_i^*$)들의 역투영사상을 통한 교집합으로 주어졌다고 가정하자.

$$E = \bigcap_i P_i^{-1} E_i \subset \Sigma^*.$$

다음 명제는 $i \in I$에 대한 각각의 전역 관리제어기 S_i 하나에 의한 제어결과가 $(L(G) \cap P_i^{-1} E_i)^{\uparrow}$일 때, 모든 S_i들의 동시 수행(concurrent action)에 의한 결과인 $L((\bigwedge_i S_i)/G)$는 $(L(G) \cap E)^{\uparrow}$와 같음을 나타낸다.

명제 3.3[2] : S_i를 $L(S_i/G) := (L(G) \cap P_i^{-1} E_i)^{\uparrow}$, $i \in I$가 만족되는 국소관리제어기라고 하자. 그러면 $L((\bigwedge_i S_i)/G) := (L(G) \cap E)^{\uparrow}$이 성립한다.

$i \in I$에 대해서 다음을 정의하자.

$$\begin{aligned} K_i &:= (L(G) \cap P_i^{-1} E_i)^{\uparrow}, \\ K_{i,\mathrm{loc}} &:= (P_i L(G) \cap E_i)^{\uparrow}_{i,\mathrm{loc}}, \\ \widetilde{K}_i &:= L(G) \cap P_i^{-1} K_{i,\mathrm{loc}}. \end{aligned}$$

또한, $S_{i,\mathrm{loc}}$을 $L(S_{i,\mathrm{loc}}/G_{i,\mathrm{loc}}) := K_{i,\mathrm{loc}}$, $i \in I$를 만족하는 $G_{i,\mathrm{loc}}$에 대한 국소 관리제어기라고 하고 (단, 여기서 $L(G_{i,\mathrm{loc}}) = P_i L(G)$이다), $\widetilde{S}_i$를 MC에 따른 G에 대한 $S_{i,\mathrm{loc}}$의 전역 관리제어기라고 하자. 즉, $\widetilde{S}_i$는 $\Sigma - \Sigma_i$에 속하는 모든 사건들의 발생에 관해서는 어떤 제어도 하지 않으며 또한 $\Sigma_c - \Sigma_{i,c}$에 속하는 모든 사건들의 발생은 언제나 허용한다. 그러면, $L(\widetilde{S}_i/G) = \widetilde{K}_i$, $(i \in I)$가 성립한다.

다음 정리 3.2는 모든 $i \in I$에 대해서, 국소 관리제어기 $S_{i,\mathrm{loc}}$에 의한 $P_i L(G)$상에서의 제어결과에 대한 MC에 따른 $L(G)$상에서의($S_{i,\mathrm{loc}}$에 대한 전역 관리제어기인 $\widetilde{S}_i$에 의한) 제어결과인 $\widetilde{K}_i$가 $K_i = \widetilde{K}_i$의 조건을 만족할 때 모든 $\widetilde{S}_i$들의 동시 수행에 의한 결과인 $L((\bigwedge_i \widetilde{S}_i)/G)$는 $(L(G) \cap E)^{\uparrow}$와 같음을 나타낸다.

정리 3.2[2] : 모든 $i \in I$에 대해서 $K_i = \widetilde{K}_i$라고 하자. 그러면

$$L((\bigwedge_i \widetilde{S}_i)/G) := (L(G) \cap E)^{\uparrow}$$

이 성립한다.

3-2. 전역 분산 관리제어

사양으로 주어진 언어 K가 세분 되지 않은 하나의 전역사양으로 주어진 경우에 대한 분산 관리제어를 알아보자. 각 국소 관리제어기에 대해 $i=2$인 경우만 고려한다. $i \geq 3$인 경우는 $i=2$를 쉽게 확장할 수 있기 때문이다.

시스템 G에서 사건집합 $\Sigma_{1,c}, \Sigma_{2,c}, \Sigma_{1,o}, \Sigma_{2,o} \in \Sigma$들과, 각 해당 투영사상(projection) $P_1 : \Sigma^* \rightarrow \Sigma_{1,o}^*$, $P_2 : \Sigma^* \rightarrow \Sigma_{2,o}^*$들이 주어졌다고 하자. 단, 여기서 $\Sigma_{i,c} = \Sigma_c \cap \Sigma_{i,o}$이다.

관리제어기 설계문제 5 : 시스템 G의 제어 목적인 사양 K가 공집합이 아닌 하나의 전반적인 사양으로 $K \subset L_m(G)$로 주어졌을 때 $L_m(\widetilde{S}_1 \wedge \widetilde{S}_2/G) = K$을 만족하는 국소 관리제어기 S_1과 S_2를 설계하라.

위 문제에 대한 관리제어기를 설계하기 위하여 다음과 같이 $\mathrm{nextact}_K$, $\mathrm{markact}_K$, 상호 관측가능성(coobservability) 등을 정의한다.

정의 3.2[3]($\mathrm{nextact}_K$) : (모든 $\sigma \in \Sigma$, $s, s' \in \Sigma^*$에 대해서) $s'\sigma \in \overline{K}$, $s \in \overline{K}$, $s\sigma \in L(G)$일 때 $s\sigma \in \overline{K}$가 만족되면 $(s, \sigma, s') \in \mathrm{nextact}_K$라고 정의한다.

사건 σ가 각 문자열 s와 s' 이후에 모두 발생가능하다고 하자. 그러면 정의 3.2의 $(s, \sigma, s') \in \mathrm{nextact}_K$의 의미는 관리제어기가 s' 이후에 발생가능한 사건 σ에 대한 허용(enable) 혹은 억제(disable)의 제어 판단이 s 이후 발생가능한 같은 사건 σ에 대한 제어 판단과 같다는 의미이다.

정의 3.3[3]($\mathrm{markact}_K$) : ($s, s' \in \Sigma^*$에 대해서) $s' \in K$, $s \in \overline{K} \cap L_m(G)$일 때 $s \in K$가 만족되면 $(s, s') \in \mathrm{markact}_K$라고 정의한다.

정의 3.3의 $(s, s') \in \mathrm{markact}_K$의 의미는 관리제어기가 문자열 s와 s'에 대해 같은 표기상태를 허용해야 한다는 의미이다.

다음 정의 3.4를 위해서 $s, s', s'' \in \Sigma^*$라고 가정하자.

정의 3.4[3] [상호 관측가능(coobservable)] :

(1) (모든 $\sigma \in \Sigma_{1,c} \cap \Sigma_{2,c}$에 대해서)
[$(s, \sigma, s') \in \mathrm{nextact}_K$ 혹은 $(s, \sigma, s'') \in \mathrm{nextact}_K$],

(2) (모든 $\sigma \in \Sigma_{1,c} \cap \Sigma_{2,c}$에 대해서)[$(s, \sigma, s') \in \mathrm{nextact}_K$],

(3) (모든 $\sigma \in \Sigma_{2,c} \cap \Sigma_{1,c}$에 대해서)[$(s, \sigma, s'') \in \mathrm{nextact}_K$],

(4) $(s, s' \in \mathrm{markact}_K)$ 혹은 $(s, s'' \in \mathrm{markact}_K)$

이 만족되면 $K \subset L_m(G)$은 G, P_1, P_2에 대해서 상호관측가능(coobservable)하다라고 정의한다.

정의 3.4의 의미는 $s, s' \in K$인 문자열에 대해 사건 σ가 각 문자열 s와 s'다음에 모두 발생가능하다고 할 때 관측불가능한 사건들에 의한 불명확한 투영사상의 결과 $s \in K$의 관측 후 s 이후 발생가능한 사건 σ의 허용 혹은 억제의 제어결과는 해당 투영사상에 의해 s와 구별이 불가능한 s' (즉, $P_i(s) = P_i(s')$를 만족하는 s') 이후 발생가능한 같은 사건 σ의 허용 혹은 억제의 제

어와 같아야 하며[(1)-(4)], 표기상태도 같아야 함[(4)]을 나타내고 있다.

정리 3.3[3] : $K \subset L_m(G)$가 G에 대해서 제어가능하고, G, P_1, P_2에 대해서 상호 관측가능하면 관리제어기 설계문제 5를 만족하는 국소 관리제어기 S_1과 S_2가 존재하며 그 역도 성립한다.

모든 국소 분산 관리제어기 (S_i)가 전역관측(global observation)이 아닌 각각의 관측가능 사건집합에 따른 투영사상, P_i ($i \in I$)에 따른 부분관측(partial observation)의 범위 내에서 주어진 전역 사양언어 K에 대한 제어를 한다고 가정하자. 이와 같은 상황에서 위 정리 3.3은 모든 국소 분산 관리제어기 (S_i)는 스스로의 부분관측가능 범위 내에서 관측한 확신할 수 있는 정보에 의해서만 해당 사건들에 대한 제어결정을 내려야 하고, 그 결정에 있어서 불확정성이 있는 사건들에 대한 제어결정은 다른 국소 분산 관리제어기 (S_i)들의 결정에 위임하는 형태로 동시 수행 $L((\bigwedge_i \tilde{S}_i)/G)$을 해야 하며, 그 결과가 K와 같아질 수 있는 즉, $L((\bigwedge_i \tilde{S}_i)/G) = K$를 만족하는 시스템 G의 요건을 전역 사양언어 K와 각각의 투영사상 P_i ($i \in I$)의 관계에서 요구하고 있다.

4. 분산 관리제어 예제 : 모든 사건들이 관측가능한 경우

[예] 다음 그림 1과 같이 직렬로 연결된 2대의 기계 (M_1, M_2)와 사이에 놓여진 1개의 버퍼(B)로 이루어진 간략화된 생산시스템을 생각해 보자[1], [8].

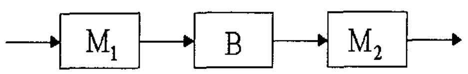

그림 2. 생산시스템의 예 : 두 대의 기계와 한 대의 버퍼.

이 경우 각각의 기계 M_1, M_2는 '대기(I: idle)', '동작(W: working)', '고장(D: down)'의 세가지 상태를 가지며 버퍼 B는 '비어있음(E: empty)' 또는 '차있음(F: full)' 상태를 가지는 1개의 슬롯(slot)으로 구성되어 있다고 가정하자. 생산시스템의 기계 부분 이산사건모델은 사건집합 $\{\alpha_i, \beta_i, \lambda_i, \mu_i\}$를 바탕으로 그림 2와 같다.

가공물품(workpiece)이 들어 왔을 때 대기상태(I)에서 동작상태(W)로의 천이를 나타내는 사건(α_i)와 고장의 수리가 완료되었을 때의 고장상태(D)로부터 대기상태(I)로의 천이(μ_i)는 제어가능한 사건들이다. 기계의 동작은 다음과 같다. 기계 M_1 이 물품을 받고(α_1) 가공이 성공적으로 완료되면 물품을 버퍼로 이동시키거나(β_i)

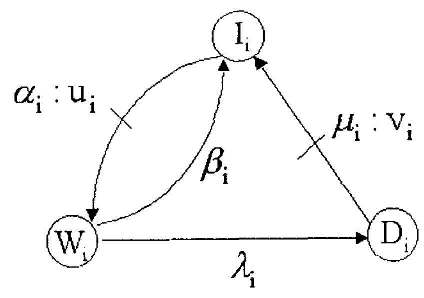

그림 3. 부분 이산사건모델: 기계 M_i.

혹은 고장이 나서 그 물품을 버린다(λ_i). 그러나 그런 경우는 다음에 곧 고장 수리가 이루어져야 한다(μ_1). 기계 M_2의 동작은 버퍼에 하나의 물품이 존재할 때만 버퍼로부터 물품을 받는 다는 점을 제외하면 기계 M_1의 동작과 동일하다. 단, 버퍼는 초기에 비어있다고 가정한다. 관리제어기의 설계문제는 다음 두 가지의 정성적(qualitative)인 사양을 만족하는 관리제어기를 설계하는 것이라고 하자 :

(1) 버퍼의 과잉(overflow) 혹은 결핍(underflow)이 발생하면 안 된다.

(2) 기계의 고장수리는 기계의 고장이 발생한 경우만 가능하며, 두 기계가 모두 고장이 난 경우는 기계 M_2가 기계 M_1보다 먼저 수리되어야 한다.

위에 G_i서의 정성적 사양을 정형화 해보자. $i = 1, 2$에 대해서 가 기계 M_i의 이산사건모델이며, 또한 $G = G_1 \| G_2$를 G_1과 G_2의 병렬합성 모델이라고 하자. 즉, $\Sigma_1 \cap \Sigma_2 = \emptyset$인 $G_i = (Q_i, \Sigma_i, \delta_i, q_{oi}, Q_{mi})$, ($i$=1, 2)라고 하면 $G = G_1 \| G_2$는 $G = (Q, \Sigma, \delta, q_o, Q_m)$인 오토마톤이 된다. 단 여기서 $Q = Q_1 \times Q_2$, $\Sigma = \Sigma_1 \cup \Sigma_2$, $q_0 = (q_{01}, q_{02})$, $Q_m = Q_{m1} \times Q_{m2}$이며 δ는 $[q_2' = q_2$ 그리고 $q_1' = \delta_1(\sigma, q_1)]$ 혹은 $[q_1' = q_1$ 그리고 $q_2' = \delta_2(\sigma, q_2)]$일 때만 $\delta(\sigma, (q_1, q_2)) = (q_1', q_2')$로 정의된다. 위 정의는 동시적(concurrently), 독립적(independently), 비동기적(asynchronously)으로 동작하는 G_1과 G_2 쌍을 나타내기 위함인데, 그렇게 함으로써 G_1과 G_2의 사건들의 순간적인 동시발생을 배제한다. $K_1 \subseteq L_m(G)$을 사양 (i)을 만족하는 문자열들의 집합이라고 하자. 그리고, 각 $\sigma \in \Sigma$ 와 $w \in \Sigma^*$에 대해서 $|w|_\sigma$를 w내에서 σ의 발생빈도 수라고 한다면 $K_1 = \{w : w \in L_m(G)$ 그리고 $0 \le |w|_{\beta_1} - |w|_{\alpha_2} \le 1\}$로 표현된다. 이와 유사하게 $K_2 \subseteq L_m(G)$을 사양 (2)를 만족하는 문자열들의 집합이라고 하고, 각 $w \in \Sigma^*$ 와 $i = 1, 2$에 대해서 $l_i(w)$를 w내에서의 마지막 사건이라고 한다면 $K_2 = \{w : w \in L_m(G)$에 대해 $l_1(w) = \lambda_1$

이고 $l_2(w)\neq\lambda_1$이면 $w\mu_1\in\overline{K_2}$이고, $l_2(w)=\lambda_2$이면 $w\mu_2\in\overline{K_2}$ 이고 $w\mu_1\notin\overline{K_2}$이다 }로 표현된다.

관리제어 문제는 설계문제 4의 형태인 $L(S/G)\subseteq K_1\cap K_2$를 만족하는 관리제어기 S를 설계함에 있어서 분산 관리제어 접근 방식을 통해 사양 (1)과 (2)를 모두 만족하는 분산 관리제어기 S_1, S_2를 설계하는 문제이다. 우선 사양 (1)에 대해서는 K_1이 제어가능한 사양언어임을 쉽게 알 수 있다. 그리고, u_1, u_2가 각각 사건 α_1, α_2의 제어변수이고, x_B가 버퍼 B의 상태를 나타낸다고 할 때, 사양 (1)은 $x_B=E$이면 $u_2=0$이고, $x_B=F$이면 $u_1=0$일 조건을 요구하고, 이것은 제어사상 $f:\{E,F\}\rightarrow\{(u_1,u_2):u_1,u_2\in\{0,1\}\}$의 부분적인 제어사상($\overline{f}$)을 나타낸다.

표 1. 제어사상 $\overline{f}$.

$\overline{f}$	u_1 u_2
E	1 0
F	0 1

또한, 이를 버퍼의 상태를 관측할 수 있는 관측기를 이용하여 그림 3과 같이 구현할 수 있다.

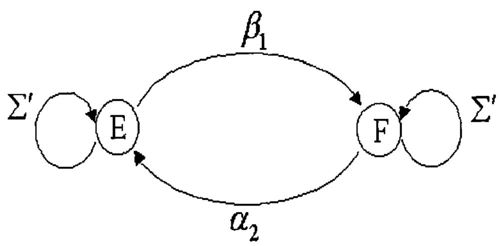

그림 4. 오토마톤 A.

여기서 $\Sigma'=\Sigma-\{\beta_1,\alpha_2\}$이다. $\overline{f}$와 A를 결합하고 S의 모든 상태에서 $v_1, v_2=1$로 설정함으로써 사양 (1)을 만족하는 관리제어기 $S_1=(A,\overline{f})$을 설계할 수 있다.

다음 사양(2)를 생각해 보자. K_2가 제어가능 언어이므로 v_1, v_2를 각각 사건 μ_1, μ_2의 제어변수, (x_1, x_2)를 M_1, M_2의 상태라고 할 때, 사양 (2)는 $v_1=1$이면 $x_1=D_1$이고, $v_2=1$이면 $x_2=D_2$이며, $x_1=D_1$ 이고 $x_2=D_2$이면 $v_1=0$ 이고 $v_2=1$일 조건을 요구한다. 이는 제어사상 $g:X_1\times X_2\rightarrow\{(v_1,v_2):v_1,v_2\in\{0,1\}\}$의 부분적인 제어사상이며, 마찬가지로 전체사상은 다음 표 2 와 같이 정리 할 수 있다.

표 2. 제어사상 g.

g	v_1 v_2
(I_1, I_2)	0 0
$(I_1,, W_2)$	0 0
(I_1, D_2)	0 1
(W_2, I_2)	0 0
(W_1, W_2)	0 0
(W_1, D_2)	0 1
(D_1, I_2)	1 0
(D_1, W_2)	1 0
(D_1, D_2)	0 1

$G_1\|G_2$의 상태를 관측할 수 있는 관측기를 이용하여 구현할 수 있는 데 S_2의 경우 기계가 다른 상태로부터 고장 상태로의 천이만 의미가 있으므로 그림 4의 간략화된 오토마톤 B와 표 3의 제어사상 $\overline{g}$만으로도 충분하다.

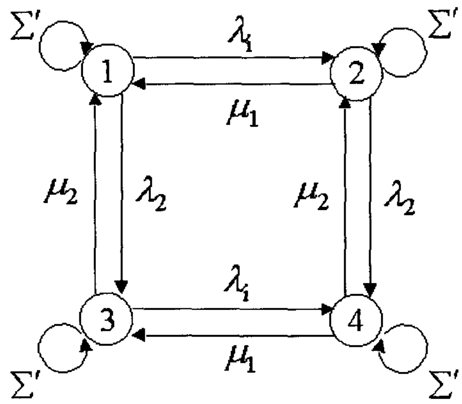

그림 5. 오토마톤 B.

표 3. 간략화된 제어사상 $\overline{g}$.

$\overline{g}$	v_1 v_2
1	0 0
2	1 0
3	0 1
4	0 1

여기서 $\Sigma'=\{\alpha_1,\alpha_2,\beta_1,\beta_2\}$ 이며 $G_1\|G_2$의 모든 가능한 상태는 다음 부분 집합 중 하나에 속한다:

$$\begin{aligned} 1 &= \{(x_1,x_2):x_1\neq D_1 \text{ 그리고 } x_2\neq D_2\}, \\ 2 &= \{(D_1,I_2),(D_1,W_2)\}, \\ 3 &= \{(I_1,D_2),(W_1,D_2)\}, \\ 4 &= \{(D_1,D_2)\}. \end{aligned}$$

$\overline{g}$과 B를 결합하고 S의 모든 상태에서 $u_1, u_2 = 1$로 설정함으로써 사양 (2)를 만족하는 관리제어기 $S_2 = (B, \overline{g})$를 설계할 수 있다. 최종적으로, 관리제어기 S를 $S = S_1 \wedge S_2$로 설정함으로써 주어진 사양을 만족하는 관리제어기를 설계할 수 있다.

5. 결론

본 논문에서는 이산사건시스템의 분산 관리제어에 관해 다루었다. 관리제어 대상시스템 G의 사양에 대해 관리제어기가 설계가능하다면 하나의 전역 관리제어기 (S)를 설계하여 사양을 만족시키는 관리제어시스템 언어 $L(S/G)$를 생성할 수 있다. 이에 반하여, 분산 관리제어에서는 주어진 사양에 대한 복수단위 관리제어기 (S_i)들의 동시수행($\wedge S_i$)을 통해 전역 관리제어기와 동등한 제어목적을 이룰 수 있음을 알 수 있었다.

일반적으로 분산 관리제어의 가능조건은 분산이 아닌 전역 관리제어의 가능조건보다는 성립요건이 까다롭다. 그러나, 만약 분산 관리제어가 적용 가능하다면 계산상의 복잡도의 감소 및 단위 구성모델의 설계변경에 따른 설계의 유연성 등의 이점이 있다. 본 논문에서는 사건들의 관측가능성과 관측불가능성을 고려하여 각각에 따른 분산 관리제어이론을 살펴보았으며, 또한 관측불가능한 사건들이 존재할 때 사양으로 주어진 언어 K가 여러 국소사양들의 교집합으로 주어진 경우의 국소 분산 관리제어와 하나의 전역사양으로 주어진 경우의 전역 분산 관리제어로 세분화하여 분산 관리제어 이론 원래의 목적인 국소 관리제어와 전역 관리제어의 결과가 같아질 수 있는 조건인 $L(S/G) = L(\wedge S_i / G)$의 만족 조건에 대해서도 알아보았다.

분산 관리제어 이론에서 고려해야 될 중요한 문제점들 중 하나는 설계된 각각의 국소 관리제어기들 상호간의 비막힘성에 대한 확인이다. 분산 관리제어에서 반드시 필요한 과정인 이러한 비막힘성의 검증과정은 분산 관리제어 스스로의 단점이 될 수도 있지만, 전역제어와 비교할 때 전체 문제에 대한 계산상의 복잡도를 감소시킬 수 있다는 분산 제어방식 고유의 이점을 희석시키지는 못한다고 할 수 있다. 그러나, 아직까지는 언어상호간의 비충돌성 관계로부터의 접근이외에는 다른 접근방법이 제시되지 않고 있는데 이런 비막힘성 특성은 문제설정(problem specification) 단계에서부터 고려하거나 혹은 시스템 구조적인 측면보다는 좀 더 상위 관점으로부터의 접근을 통해서 해결될 수도 있을 것이다.

현재까지의 분산 관리제어 구조는 제어목적인 하나의 전역적 사양언어가 세분된 여러 사양언어들의 교집합을 통해서 주어진 문제들을 가정하고 있는데 이런 구조는 하나의 시스템에서 여러개의 목적을 가지고 있는 문제인 경우에 적당한 구조이다. 그리고, 이와 반대적 개념인 전역적으로 주어진 하나의 사양언어 문제인 경우는 상호관측가능성 측면으로 해결될 수 있다. 그러나, 이 방법은 그 계산상의 복잡도가 상대적으로 크다는 단점이 있으므로, 주어진 하나의 전역 사양언어를 여러 개의 세분화된 사양언어들의 결합으로 분해하여 해석하는 접근방법도 생각해 볼 문제이다[5]. 또한, 전역 제어에 비해서 상대적으로 한계성을 가지고 있는 정보처리 능력에 대한 분산 관리제어 자체의 단점을 인정하고 전역과 분산 관리제어의 적당한 혼합제어에 대한 연구도 진행되고 있다[4], [11].

제어목적을 이루기 위해 주어진 사양언어와 해당 시스템과의 상호간 특성에 대해서 많은 연구가 행해지고 있는 반면에, 시스템 자체를 해석하는 측면으로도 많은 연구가 되고 있다. 특히, 모델 불확실성을 근거로 한 다중모델(multiple model) 접근방법이 소개되었고, 이를 기반으로 한 많은 이산사건시스템의 강인관리제어 및 최적 관리제어이론이 연구되고 있으나 대체적으로 전역적 제어 측면에서만 연구되고 있다. 따라서, 분산 관리제어 이론에서도 이런 다중모델 구조하에서의 연구가 필요할 것이며, 이에 관하여 많은 해결해야할 문제들이 있을 것으로 생각된다. 이외에도, 이산사건 시스템의 고장진단 및 검출에 관한 많은 연구가 행해지고 있으며, 이런 결과에 분산 관리제어이론 자체의 이점을 고려한 응용이 필요할 것이다.

참고문헌

[1] W. M. Wonham and P. J. Ramadge, “Modular supervisory control of discrete event systems”, *Math. of Control, Signals, and Systems*, vol. 1, pp. 13-30, 1998.

[2] F. Lin and W. M. Wonham, “Decentralized Supervisory Control of Discrete Event Systems”, Information Sciences., vol. 44. pp. 199-224, 1988.

[3] K. Rudie and W. M. Wonham, “Think globally, act locally: Decentralized supervisory control”, *IEEE Trans. Automat. Contr.*, vol. 37, pp. 1692-1708, 1992.

[4] F. Lin and W. M. Wonham, “Decentralized control and coordination of discrete event systems with partial observation”, *IEEE Trans. Automat. Contr.*, vol. 35, pp. 1330-1337, 1990.

[5] F. Lin and H. Mortazavian, “A Normality Theorem for Decentralized Control of Discrete-Event Systems”, *IEEE Trans. Automat. Contr.*, vol. 35, pp. 1330-1337, 1990.

[6] F. Lin, “Control of large scale discrete event systems: Task allocation and coordination”, *Systems and Control Letters*, vol. 17, pp. 169-175, 1991.

[7] P. J. Ramadge and W. M. Wonham, “Supervisory

control of a class of discrete event processes", *SIAM J. of Control and Optimization*, vol. 25, pp. 206-230, 1987.

[8] W. M. Wonham and P. J. Ramadge, "On the supremal controllable sublanguage of a given language", *SIAM J. of Control and Optimization*, vol. 25. pp. 637-659, 1987.

[9] 조 광현, 임 종태, "이산사건시스템의 관리제어(I): 이산사건모델링 및 관리제어이론", 제어 · 자동화 · 시스템공학회지, 제6권, 제3호, 2000

[10] 조 광현, 임 종태, "이산사건시스템의 관리제어(II): 관리제어기 설계", 제어 · 자동화 · 시스템공학회지, 제6권, 제3호, 2000

[11] K. -H. Cho and J. -T. Lim, "Mixed centralized/decentralized supervisory control of discrete event dynamic systems", Automatica, vol. 35, pp. 121-128, 1999.

김 성 규

1995년 경북대학교 전자공학과 졸업. 한국과학기술원 전기및전자공학과 석사 (1998), 현재 동대학원 박사과정. 관심분야는 supervisory control of discrete event systems and applications, network congestion control 등.

조 광 현

제어 · 자동화 · 시스템공학회지 제6권제3호 참조. 현재 울산대학교 전기전자 및 자동화공학부 전임강사.

임 종 태

제어 · 자동화 · 시스템 공학회지 제4권 제4호 참조. 현재 한국과학기술원 전기및전자공학과 교수.