최적 관리제어

이 문 상*, 조 광 현**, 임 종 태*
*한국과학기술원 전기 및 전자공학부, **울산대학교 전기전자 및 자동화공학부

Abstract : 본 논문에서는 관리제어시스템의 동적특성을 허용언어(admissible language) 범위 이내에서 최적화시키는 최적 관리제어기법들을 소개한다. 본 논문에서 주로 다루고자 하는 최적 관리제어기법은 Kumar와 Garg에 의해 제안된 기법과 Cho와 Lim에 의해 제안된 계층적 최적 관리제어기법, 그리고, Sengupta와 Lafortune이 제안한 최적 관리제어기법 등이다. 첫 번째 기법에서는 우선 시스템의 최적화를 위해 고려되고 있는 비용함수(cost function)를 소개한 후, 최대흐름 최소분할생성 정리(max-flow min-cut theorem)를 이용한 최적 관리제어기 설계기법을 제시하고, 이를 부분관측 하에서도 최적 관리제어기를 설계할 수 있도록 확장한다. 그런 후 제시된 설계기법에 의해 설계된 관리제어시스템에서 발생 할 수 있는 문제점들을 지적하고, Cho와 Lim에 의해 제안된 완전 최소분할생성(complete min-cut)이라는 개념을 도입하여 지적된 문제점들을 해결할 수 있는 방법을 제시한다. 또한 시스템의 고장을 고려한 계층적 최적 관리제어(layered optimal supervisory control)기법을 소개한다. 그리고 마지막으로 Sengupta와 Lafortune이 제안한 최적 관리제어기법에 대해서 살펴본다.

1. 서론

관리제어이론에서 관리제어기의 역할은 플랜트에서 특정 사건이 발생하도록 하는 능동적인 제어기의 역할이 아니라 플랜트가 생성하는 사건열을 관측해가며 각각의 시점에서 발생될 수 있는 사건들 가운데 제어가능한 사건들을 선택적으로 억제(disable)함으로써 플랜트의 동적특성을 우리가 원하는 범위 내에 있도록 제한하는 것이다. 이러한 관리제어이론 하에서 초기의 최적 관리제어의 개념은 주어진 사양 언어 K에 대해 $L(S/G)$가 K의 최고 제어가능 부언어(supremal controllable sublanguage)가 되도록 하는 최소 제한적(minimally restrictive) 관리제어기를 설계하는 것이었다. 하지만 최근에는 연속변수시스템에서 생각되어지는 것처럼 플랜트의 운영을 고려한 비용함수(cost function)를 정의하고, 이 비용함수가 최소화되도록 관리제어기를 설계하는 최적 관리제어기법에 관한 연구가 활발히 진행되고 있다 [1]-[6]. 이러한 연구들에서는 시스템 운영시 고려되어야 할 사항들을 반영한 적절한 비용함수를 정의하고, 정의된 비용함수를 최소화시킬 수 있는 관리제어기 설계를 위한 이론적 배경을 구축하여 구체적인 관리제어기 설계기법을 제시하고 있다.
우선 [1]과 [3]에서는 비용함수를 플랜트의 상태천이집합(state transition set)에 대해 정의하였다. 이는 플랜트가 상태천이를 하기 위해, 즉 사건을 발생시키기 위해 필요한 비용을 반영한 것이다. 제어목적은 초기상태에서 표기상태(marker states)에 이르기까지 비용함수가 최소가 되는 최적의 상태궤적을 따라 관리제어시스템이 동작되도록 플랜트의 동적특성을 제한하는 것이다. 결국 이러한 역할을 함으로써 비용함수를 최소화시키는 관리제어기를 설계하기 위해서는 최적의 상태궤적을 구해내야 하고, 결과적인 최적관리제어 문제는 비중그래프(weighted graph)에서 최단경로를 찾는 문제로 귀결된다. 그러나 플랜트의 동적특성이 구해진 상태궤적상에서 이루어지도록 하기 위해서는 일반적인 관리제어이론에서처럼 사건의 발생을 억제하는 것이 아니라 특정한 사건의 발생을 유발(forcing)해야하는 관리제어기의 역할이 필요하게 된다. 그러므로 이러한 접근방식은 일반적인 관리제어 이론체계와는 약간의 차이가 있는 접근방식이라고 할 수 있겠다. 또한 플랜트의 제어를 위한 관리제어기의 동작에 필요한 비용은 전혀 고려가 되고 있지 않은 점도 개선되어야 할 문제라고 할 수 있다. [2], [4], [5], [6]에서는 관리제어기의 제어에 대한 비용까지 고려한 비용함수를 제안하였다. 따라서 앞서 언급했던 연구들에 비해 좀 더 일반적인 최적 관리제어 기법을 제안했다고 할 수 있다. 하지만 [4], [6]에서 제안된 비용함수와 [2], [5]에서 제안된 비용함수는 그 의미에서 약간의 차이가 있으며 각각 전혀 다른 방법을 통해 비용함수를 최적화 할 수 있는 최적 관리제어기를 설계하고 있다. 본 논문에서는 현재까지 연구개발되어져온 이러한 각각의 최적 관리제어기법들에 대해 살펴보고자 한다.

본 논문의 구성은 다음과 같다. 2절에서는 [4]에서 제안된 최적 관리제어기법에 대해 살펴보고, 3절에서는 [4]에서 제안된 기법이 지니고 있는 문제점의 해결방법과 계층적 최적 관리제어기법[6]에 관해 살펴본다. 그

리고 4절에서는 [2], [5]에서 제안된 최적 관리제어기법을 다루고 5절에서 결론을 맺는다.

2. 최적 관리제어기법(유형 I)

이 절에서는 Kumar와 Garg가 제안한 최적 관리제어기법[4]에 대해 알아본다.

2-1. 비용함수

[4]에서는 최적 관리제어기를 설계하기 위해 두 가지의 비용함수를 고려하고 있다. 그 하나는 플랜트의 상태천이, 즉 사건집합에 대해 정의된 '제어 소모비용(cost of control)' 이고, 다른 하나는 상태집합에 정의된 '제어 벌칙비용(penalty of control)'이다. 전자의 경우 한 상태천이에 대해 항상 양의 실수가 정의되는데 이는 상태 천이, 즉 사건의 발생을 억제하기 위해 필요한 비용을 고려한 것으로서 관리제어기가 하나의 상태천이를 억제시킬 경우 그 천이에 정의된 비용이 부가되고 그렇지 않을 경우는 아무 비용도 부가되지 않는다. 제어불가능한 사건에 대해서는 무한대의 비용을 부가함으로써 사건의 제어불가능성을 반영할 수 있다. 후자의 경우는 상태의 도달 가능성에 따라 벌칙비용이 더해진다. 상태집합은 의도상태(desired state)집합과 비의도상태(undesired state) 집합으로 나뉘어지며 의도상태에는 음의 실수가, 비의도상태에는 양의 실수가 벌칙비용으로 정의된다. 관리제어기에 의해 비의도상태가 도달가능하게 되면 그 상태에 정의된 양의 벌칙비용만큼이 벌칙비용으로 부가되고, 의도상태가 도달불가능하게 되는 경우는 그 상태에 정의된 음의 벌칙비용의 절대값이 벌칙비용으로 부가된다. 물론 의도상태가 도달 가능하거나 비의도상태가 도달불가능한 경우는 아무 벌칙비용도 부가되지 않는다. 이러한 내용을 체계적으로 정리하면 다음과 같다.

플랜트를 $G=\{Q, \Sigma, \delta, q_o\}$, 관리제어기 S를 플랜트의 각 상태에서 억제해야할 사건의 부분합으로의 사상(map) $S: Q \rightarrow 2^{\Sigma}$ 으로 생각하자(여기서는 표기상태를 따로 고려하지 않는다). 또한, G_S 를 관리제어기 S에 의해 제어되는 플랜트라고 하고, $Re(G_s)$를 제어되는 플랜트의 초기상태에서 도달 가능한 상태집합이라고 하자. 그리고 제어 소모비용 c를 상태 천이의 집합에서 양의 실수로의 사상 $c: Q \times \Sigma \rightarrow R^+$, 제어 벌칙비용 p를 상태 집합에서 실수로의 사상 $p: Q \rightarrow R$ 이라고 생각하자. 물론 $p(q) < 0$이면 q는 의도상태이고 $p(q) > 0$이면 q는 비의도상태가 된다. 그러면 관리제어기에 대한 순수비용(net cost)은 다음과 같이 정의될 수 있으며 최적 관리제어기는 최소의 순수비용을 갖는 관리제어기가 된다.

정의 2.1[4](관리제어기의 순수비용) : 어떤 관리제어기 S에 대한 순수비용 $C(S)$는 $C(S)=\sum_{q\in Re(G_s)}[\sum_{\sigma\in S(q)} c(q,\sigma)]+\sum_{q\in Re(G_s), p(q)>0} p(q)+\sum_{q\notin Re(G_s), p(q)<0} -p(q)$이다.

정의 2.1의 관리제어기 순수비용을 살펴보면 제어의 비용이나 벌칙비용은 모두 하나의 상태천이나 상태에 대해 한번만 고려가 되고 있다. 그러나 G_S에서 어떤 상태는 한번 이상 경유할 수 있는 것들도 있을 수 있다. 이런 경우에는 그 상태를 경유할 때 마다 관리제어기가 억제되어야 할 상태천이, 즉 사건의 발생을 억제해 주어야 한다. 그렇다면 왜 제어 소모비용을 한번만 고려한 것인가? 간단한 예를 들어 설명하면, 실제 시스템에서 사건의 발생을 억제하기 위해서는 스위치나 그 밖의 제어장치를 설치해 주어야만 하고, 필요할 때 설치된 제어장치를 통해 사건의 발생을 억제하게 된다. 그러므로 스위치를 여닫는데 드는 비용은 처음 그 스위치를 설치하는데 드는 비용에 비해 무시할 수 있다고 생각한 것이다. 상태에 대한 벌칙비용을 한 번씩만 고려한 것도 비슷한 맥락에서 이해하면 될 것이다.

2-2. 최대흐름 최소분할생성 정리

최적 관리제어기의 설계문제는 각 상태천이에 대한 억제, 허용여부를 결정하여 정의 2.1의 순수비용을 최소화하는 문제이다. 결국 이는 플랜트의 상태집합을 도달가능한 상태집합과 도달불가능한 상태집합으로 분할하는 최적분할 문제로 귀결되고, 이러한 최적분할 문제는 방향그래프(directed graph)의 최적분할에 사용되는 최대흐름 최소분할생성 정리를 써서 해결할 수 있다. 이제부터 최대흐름 최소분할생성 정리를 이해하기 위해 흐름망(flow network)과 흐름(flow), 그리고 분할생성(cut) 등의 정의를 먼저 살펴보고 최대흐름 최소분할생성 정리에 관해 알아보도록 한다.

정의 2.2[4] (흐름망) : 흐름망 N은 $N=(V, E, u)$로 나타낼 수 있는 가중 방향그래프(weighted directed graph)이다. 여기서 V는 그래프의 절점(node)의 집합, E $(\subseteq V^2)$는 연결선(edge)의 집합을 나타내며 $u: E \rightarrow R^+$ 는 각 연결선의 최대 흐름용량(capacity)를 나타낸다. V는 근원절점(source node) s와 종단절점(terminal node) t라는 특별한 두 절점을 포함한다.

정의 2.3[4] (흐름) : 흐름망 N에서의 흐름은 $f: E \rightarrow R^+$ 인 하나의 사상이며 다음과 같은 세가지 성질을 만족한다.

(1) $\forall e \in E: f(e) \leq u(e)$,

(2) $\sum_{v\in V:(s,v)\in E} f((s,v)) = \sum_{v'\in V:(v',t)\in E} f((v',t))$,

(3) $\forall v \in V, v\neq s,\ v\neq t: \sum_{v'\in V:(v,v')\in E} f((v,v')) = \sum_{v''\in V:(v'',v)\in E} f((v'',v))$.

흐름의 첫 번째 성질은 흐름은 결코 연결선의 용량을 넘을 수 없다는 것이고, 두 번째 성질은 근원절점에서

나온 순수 흐름(net flow)은 종단절점으로 흘러들어가는 순수 흐름과 같다는 것이며, 세 번째 성질은 근원절점이나 종단절점 이외의 절점을 통과하는 흐름의 양은 보존된다는 것이다.

정의 2.4[4] (분할생성) : 흐름망 N에 대한 분할생성은 절점집합 V를 근원절점 s와 종단절점 t가 서로 다른 쪽에 속하도록 양분하는 것을 의미한다. 분할생성에 의해 V가 s를 포함하는 $V_s(\subset V)$와 t를 포함하는 $V_t(=V-V_s)$로 분할되었을 때 그 분할생성의 용량(capacity of cut)은 $\sum_{(i,j)\in(V_s\times V_t)\cap E} u((i,j))$가 된다. 또 최소분할생성(min-cut)은 최소의 용량을 가지는 분할생성이 된다.

정의 2.4에 따르면 흐름망 N의 절점집합을 근원절점 s와 종단절점 t가 서로 다른 부분에 속하도록 둘로 분할해 주는 것은 모두 분할생성이 되며 그 분할생성의 용량은 s를 포함한 V_s에 속한 절점과 V_t에 속한 절점을 연결하는 모든 연결선들의 용량의 합이 되는 것이다. 이제 최대흐름 최소분할생성 정리를 알아보도록 하자.

정리 2.1[4] (최대흐름 최소분할생성 정리) : 흐름망의 흐름 최대값은 그 흐름망의 최소분할생성의 용량과 같다.

정리 2.1의 의미는 다음과 같이 이해할 수 있다. 관을 통해 물을 통과시킬 때 관을 통해 최대로 통과될 수 있는 단위시간당 물의 양은 그 관의 단면적에 비례하게 된다. 그리고 만일 관의 단면적이 일정하지 않다면 최대로 통과될 수 있는 단위시간당 물의 양은 그 관의 최소 단면적에 비례할 것이다. 흐름도의 각각의 연결선을 단면적이 일정한 관들에 비유해서 생각한다면 연결선의 용량은 단면적으로 생각될 수 있으며 흐름은 관을 통과하는 단위시간당 물의 양으로 생각될 수 있다. 또, 최소분할생성의 용량은 근원절점과 종단절점 사이에 연결된 여러 관들에 대한 단면적의 합의 최소값이 된다. 그러므로 흐름망의 최소분할생성의 용량을 구하면 결국 흐름망의 흐름의 최대값을 알 수 있게 되는 것이다.

우리의 목표는 최소의 순수비용을 갖는 최적 관리제어기를 설계하는 것이다. 지금부터 우리는 이 최적 관리제어기 설계 문제를 플랜트를 적절히 변환시킨 흐름망에서 최소분할생성을 구하는 문제로 바꾸어 해결할 것이다. 이때 최소분할생성은 정리 2.1에 의해 흐름망의 최대흐름을 구함으로써 구할 수 있는데 이는 최대흐름을 구하는 알고리즘이 최소분할생성에 속하게 되는 연결선들을 구함으로써 최대흐름을 구하기 때문이다. 흐름망에 대한 최대흐름을 구하는 알고리즘은 조합알고리즘(combinatorial algorithm)을 다루고 있는 많은 책들에서 다루고 있으며 대표적인 알고리즘으로 Ford-Fulkerson 알고리즘, Dinic 알고리즘, Karzanov 알고리즘 등을 들 수 있다[8]. 그러나 최대흐름을 구하는 알고리즘들에 대해 자세히 다루지 않고 그 중 하나의 방법을 차용하여 최대흐름과 최소분할생성을 구할 수 있는 것으로 간주한다.

2-3. 최적 관리제어기의 설계

이 절에서는 앞서 소개한 최대흐름 최소분할생성 정리를 써서 최적 관리제어기 설계기법을 제시한다. 관리제어기는 플랜트의 상태집합 Q를 $Re(G_s)$와 $Q-Re(G_s)$로 분할한다. 이제 우리는 경제적 관리제어기(parsimonious supervisor)를 정의하고 관리제어기의 경제성(parsimonicity)이 최적 관리제어기의 필요조건임을 보일 것이다.

정의 2.5[4] (경제적 관리제어기) : 관리제어기 $S: Q\to 2^{\Sigma}$ S가 경제적 관리제어기이기 위한 필요충분조건은 각 상태 $q\in Q$ 와 각 사건 $\sigma\in\Sigma$에 대해 $\sigma\in S(q)\Leftrightarrow[q\in Re(G_s),\ \delta(q,\sigma)\notin Re(G_s)]$가 된다.

경제적 관리제어기는 $Re(G_s)$에 속하는 상태들 사이에 정의된 상태천이들, 즉 사건들이나 $Re(G_s)$에 속하지 않는 상태들 사이에 정의된 사건들을 억제하지 않는 관리제어기이다. 이러한 경제적 관리제어기를 정의하는 이유는 관리제어기의 경제성이 최적 관리제어기이기 위한 필요조건이 되기 때문이다. 왜냐하면 일단 상태집합 Q가 관리제어기에 의해 $Re(G_s)$와 $Q-Re(G_s)$로 분할이 되면 정의 2.1의 관리제어기 순수비용 식에서 두 번째와 세 번째 항이 되는 제어에 대한 벌칙비용 항들의 값은 정해지게 되어 $Re(G_s)$와 $Q-Re(G_s)$로 분할하기 위해 억제했던 사건들 이외에 $Re(G_s)$에 속하는 상태들 사이에 정의된 사건들이나 $Q-Re(G_s)$에 속하는 상태들 사이에 정의된 사건들을 억제하는 것은 부가적인 비용의 증가를 가져오기 때문이다. 이러한 사실을 통해 우리는 다음과 같은 보조정리와 따름정리를 생각할 수 있다.

보조정리 2.1 [4] : 만일 관리제어기 $S: Q\to 2^{\Sigma}$ 가 최적 관리제어기이면 반드시 S는 경제적 관리제어기이다.

보조정리 2.1은 경제적 관리제어기가 아닌 최적 관리제어기는 존재할 수 없다는 의미를 가진다. 경제적 관리제어기가 아닌 관리제어기에 대해서는 항상 억제된 도달가능한 상태 사이의 상태천이들, 혹은 도달불가능한 상태 사이의 상태천이들이 존재하므로 이런 상태천이들을 허용하는 경제적 관리제어기는 같은 도달가능 상태집합을 가지면서도 항상 더 작은 순수비용을 가지게 되는 것이다. 이로부터 다음 따름정리 2.1을 도출해낼 수 있다.

따름정리 2.1 [4] : 최적 관리제어기를 설계하는 것은 경제적 관리제어기 중에서 최소의 순수비용을 갖는 관리제어기를 설계하는 것이다.

이제 최대흐름 최소분할생성 정리를 이용하여 최적 관리제어기 설계기법을 제시하고자 한다. 이에 앞서 우선 플랜트와 플랜트 상태집합, 사건집합에 대해 제어 소모비용과 제어 벌칙비용이 주어진 경우 플랜트를 흐름망으로 전환하는 방법을 살펴보도록 한다.

정의 2.6 [4] : 플랜트 $G=(Q,\ \Sigma,\ \delta,\ q_0)$가 주어졌을 때 흐름망 N_G는 $N_G=(V_G,\ E_G,\ u_G)$와 같이 정의되며, 각각의

V_G, E_G , u_G는 다음과 같다.

(1) $V_G = Q \cup \{ s, t \}$이고, $s, t \not\in Q$,

(2) $E_G = \{(q_1, q_2) \in Q \times Q \mid \delta(q_1, \sigma) = q_2$를 만족하는

$\sigma \in \Sigma$ 존재$\} \cup \{(q, t) \in Q \times \{t\} \mid p(q) > 0\}$

$\cup \{(s, q) \in \{s\} \times Q \mid p(q) < 0\}$,

(3) $\forall (q_1, q_2) \in E_G \cup (Q \times Q)$:

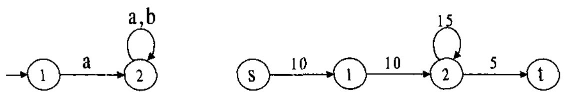
$u_G((q_{1,} q_2)) := \sum_{\sigma \in \Sigma\, \delta(q_1, \sigma) = q_2} c(q_1, \sigma)$,

$p(q) > 0$인 모든 $q \in Q$에 대해 : $u_G((q, t)) := p(q)$,

$p(q) < 0$인 모든 $q \in Q$에 대해: $u_G((q, t)) := -p(q)$.

정의 2.6에서 N_G는 플랜트의 상태집합에 근원절점 s, 종단절점 t를 추가한 후 원래 플랜트의 상태집합 사이에서 정의되어있는 상태천이는 그대로 두고 그 천이가 나타내는 연결선의 용량을 천이에 정의된 제어의 소모비용값으로 하고, 근원절점 s에서 의도상태들로, 또는 비의도상태들에서 종단절점 t로 연결되는 추가적인 연결선을 만들어 그 연결선의 용량을 상태들의 제어 벌칙비용의 절대값으로 생각하는 것이다. 이렇게 만든 N_G는 주어진 플랜트와 제어 소모비용, 제어 벌칙비용 등이 주어졌을 때 최적 관리제어기를 설계하기 위해 사용될 것이다. 간단한 예를 통해 플랜트 G를 N_G로 바꾸는 방법을 알아보자.

[예] 그림 1의 좌측에 주어진 플랜트 G를 생각하자. 상태집합 Q는 $\{1, 2\}$이고 사건집합 Σ는 $\{a, b\}$, 초기상태 $q_0 = 1$, 상태천이함수 δ는 $\delta(1, a) = 2$, $\delta(2, a) = 2$, $\delta(2, b) = 2$로 주어져 있다. 플랜트 G의 생성언어는 $L(G) = \overline{a(a+b)^*}$가 된다는 것을 알 수 있으며 제어 소모비용과 벌칙비용이 $c(a) = 10$, $c(b) = 5$, $p(1) = -10$, $p(2) = 5$로 주어져 있을 때 N_G는 정의 2.6에 의해 그림 1의 우측에 보이는 것과 같이 유도될 수 있다.



(G) (N_G)

그림 2. 플랜트와 유도된 흐름망의 예.

정의 2.7 [4] : 관리제어기 $S : Q \rightarrow 2^{\Sigma}$ 가 주어졌을 때 S에 의해 $[Re(G_s) \cup \{s\}] \cup [(Q - Re(G_s)) \cup \{t\}]$와 같이 정의되는 N_G에 대한 분할생성을 유도해 낼 수 있고, N_G에 대한 분할생성 $V_s \cup (V_G - V_s)$, (단 $V_s \subseteq V_G$, $s \in V_s$, $t \not\in V_s$)와 같이 주어졌을 때는 이 분할생성으로부터 $q \in Q$, $\sigma \in \Sigma$, $\sigma \in S(q) \Leftrightarrow q \in V_s$, $\delta(q, \sigma) \not\in V_s$인 조건을 만족하는 경제적 관리제어기 S를 유도해 낼 수 있다.

정의 2.7에서 우리가 알 수 있는 사실은 N_G의 분할생성 집합과 G의 경제적 관리제어기 집합이 일대일 대응관계가 있다는 것이다. 이제 우리는 정의 2.7 과 함께 다음의 보조정리 2.2와 정리 2.2를 이용해 최소의 순수비용을 갖는 최적 관리제어기를 설계할 수 있다.

보조정리 2.2 [4] : 만일 관리제어기 $S : Q \rightarrow 2^{\Sigma}$ 가 경제적 관리제어기이면 관리제어기 S의 순수비용 $C(S)$는 S에 의해 유도되는 N_G에 대한 분할생성의 용량과 같다.

정리 2.2 [4] : N_G의 최소분할생성에 의해 유도되는 경제적 관리제어기가 최적 관리제어기이다.

보조정리 2.2는 플랜트 G로부터 N_G를 유도하는 과정으로부터 도출되는 결과이다. S에 의해 유도되는 N_G에 대한 분할생성은 $[Re(G_s) \cup \{s\}]$에서 $[(Q - Re(G_s)) \cup \{t\}]$로 연결된 연결선들만을 포함한다. 이때 $Re(G_s)$에서 $(Q - Re(G_s))$로 연결된 연결선의 용량들이 경제적 관리제어기가 억제해야만 하는 상태천이들의 제어 소모비용값들이고 근원절점 s에서 $(Q - Re(G_s))$로 연결된 연결선들의 용량이 $Re(G_s)$에 속하지 못하게되는 의도상태들의 제어 벌칙비용의 절대값, 그리고 $Re(G_s)$에서 종단절점 t로 연결된 연결선들의 용량이 $Re(G_s)$에 속하게되는 비의도상태들의 벌칙비용이 된다. 그러므로 S에 의해 유도되는 N_G에 대한 분할생성의 용량이 S와 같은 도달가능상태집합을 갖는 경제적 관리제어기의 순수비용이 되는 것이다. 그러므로 N_G의 최소분할생성을 구하고 이에 대한 경제적 관리제어기를 유도하면 이것이 최적 관리제어기가 되는 것이다.

지금까지 제시된 최적 관리제어기 설계 방법을 요약, 정리하면 다음과 같다. 플랜트 G와 제어 소모비용, 제어 벌칙비용이 주어졌을 때 정의 2.6을 이용하면 플랜트를 흐름망 N_G로 표현할 수 있으며 이 흐름망으로부터 정의 2.7을 이용하여 구해지는 분할생성은 다시 이 플랜트 G의 경제적 관리제어기로 변환될 수 있다. 이때 분할생성의 용량은 보조정리 2.2에 의한 분할생성에 대응하는 경제적 관리제어기의 순수비용이 된다. 그러므로 정리 2.1에 주어진 최대흐름 최소분할생성 정리와 최대흐름을 구하는 알고리즘을 이용하여 N_G의 최소분할생성을 찾게 되면 이에 대응하는 경제적 관리제어기는 관리제어기들 중 최소의 순수비용을 갖는 관리제어기가 되고, 결국 따름정리 2.1에 의해 최적 관리제어기를 구하게 되는 것이다.

2-4. 부분관측하에서의 최적 관리제어기의 설계

지금까지는 모든 사건이 관측가능하다는 가정 하에서 최적 관리제어를 생각했다. 그렇다면 관측불가능한

사건이 존재할 때는 어떻게 최적 관리제어기를 설계할까? 이 절에서는 부분관측하에서도 최적 관리제어기를 설계할 수 있도록 앞 절에서 제시한 설계기법을 확장하도록 한다.

관측불가능한 사건이 있다는 것은 플랜트에서 관측불가능한 사건의 발생으로 인한 상태변화를 관리제어기에서 감지할 수가 없기 때문에 관리제어기가 플랜트의 상태변화에 대한 정보를 완벽히 알고 있지 못하다는 의미가 된다. 그러므로 관리제어기는 플랜트의 상태공간 Q에서 관측공간 Y로의 사상 $\Psi : Q \rightarrow Y$를 통해 플랜트의 상태정보를 받아들이게 되고, 이러한 상황하에서 관리제어기는 $S' : Y \rightarrow 2^{\Sigma}$ 와 같이 나타낼 수 있다. 관리제어기 S'은 관측상태 $y \in Y$에 따라 제어를 해나가야 하기 때문에 $\Psi^{-1}(y) = \{q \in Q \mid \Psi(q) = y\}$인 상태집합에서 같은 제어행동을 취해야 한다. 즉, 이러한 부분관측하에서의 관리제어기 S'은 제약조건 $C_1 : \forall q_1, q_2 \in Q : \Psi(q_1) = \Psi(q_2) \Rightarrow S(q_1) = S(q_2)$을 만족시키는 완전관측하에서의 관리제어기 $S : Q \rightarrow 2^{\Sigma}$로 나타낼 수 있다. 결국 부분관측하에서의 최적 관리제어문제는 제약조건 C_1을 만족시키면서 최소의 순수비용을 갖는 관리제어기를 설계하는 문제가 된다.
플랜트 G와 소모비용, 벌칙비용, 그리고 Ψ가 주어졌을 경우 최적 관리제어기 설계를 위해서는 다음의 정의 2.8에 따라 G를 G'으로 변환하여야 한다.

정의 2.8 [4] : 플랜트 $G=\{Q,\ \Sigma,\ \delta,\ q_o\}$가 주어졌을 때 제한조건 C_1을 고려한 변형 플랜트는 $G' = \{Q,\ \Sigma',\ \delta',\ q_o\}$로 나타나고 Σ', δ'은 다음의 조건을 만족한다.

(1) $\Sigma' = \Sigma \cup \{\theta\}$, 단 $\theta \notin \Sigma$,

(2) $\sigma' \neq \theta$인 경우, $\forall q \in Q,\ \forall \sigma' \in \Sigma'$:
$\delta'(q, \sigma') = \delta(q, \sigma')$,

(3) $\Psi(q_1) = \Psi(q_2)$, $\delta(q_1, \sigma) \neq \delta(q_2, \sigma)$를 만족하는 모든 $\sigma \in \Sigma$,
$q_1, q_2 \in Q$에 대해 $\delta'(\delta(q_1, \sigma), \theta) = \delta(q_2, \sigma)$,
$\delta'(\delta(q_2, \sigma), \theta) = \delta(q_1, \sigma)$.

정의 2.8에 의해 변환된 플랜트 G'은 만일 플랜트 G에서 관리제어기에 의해 구분되지 못하는 다른 두 상태 q_1, q_2에서 같은 사건의 발생에 각각 다른 상태 q'_1, q'_2로 천이되는 경우 q'_1, q'_2 사이에 부가적인 사건 θ를 첨가해 준 것이다. 그리고 제어의 비용 $c : Q \times \Sigma \rightarrow R^+$를 다음과 같은 성질을 만족하는 $c' : Q \times \Sigma \rightarrow R^+$ 로 확장한다.

즉, $\forall q \in Q,\ \sigma' \in \Sigma'$ 에 대해

$$c'(q, \sigma') = \begin{cases} \sigma' \in \Sigma \text{ 인 경우 } c(q, \sigma')\text{이고,} \\ \sigma' = \theta \text{ 인 경우 } \quad \infty \quad \text{이다 .} \end{cases}$$

이렇게 변형된 플랜트 G'와 확장된 비용 c'을 구하고 나면 우리는 다음의 정리 2.3에 의해 부분관측 하에서도 최적 관리제어기를 설계할 수 있다.

정리 2.3 [4] : 플랜트 G와 소모비용 c, 벌칙비용 p에 대해 부분관측 하에서 최적관리제어기를 설계하는 것은 변형 플랜트 G', 확장 소모비용 c', 벌칙비용 p에 대해 완전관측 하에서 최적 관리제어기를 설계하는 것과 동일하다.

변형된 플랜트에서의 사건 θ는 ∞의 비용을 가지기 때문에 결코 최적 관리제어기에 의해 억제될 수가 없다. 그러므로 관리제어기에 의해 구분되지 못하는 다른 두 상태 q_1, q_2에서 같은 사건 σ의 발생에 각각 다른 상태 q'_1, q'_2로 천이되는 경우 q'_1, q'_2가 하나는 $Re(G'_S)$에 속하고 다른 하나는 그렇지 못하게 되는 경우는 없다. 또, 최적 관리제어기는 반드시 경제적 관리제어기이기 때문에 q_1, q_2가 둘 다 $Re(G'_S)$에 속하는 경우에 q_1, q_2 둘 중 하나에서는 σ의 발생을 허용하고 다른 하나에서는 억제하는 일이 발생하지 않기 때문에 G'에 대해 설계한 관리제어기는 항상 C_1을 만족하게 되고 이런 관리제어기 중 최소의 순수비용을 갖기 때문에 결국 부분관측 하의 최적 관리제어기가 되는 것이다.

3. 최적 관리제어기법(유형 I) 의 문제점 해결과 계층적 관리제어기법

이 절에서는 2절에서 소개한 최적 관리제어기법이 안고 있는 문제점을 지적하고, 이를 해결하는 방법을 제시한다. 또 시스템의 고장까지 고려한 계층적 관리제어기법에 대해서도 살펴본다.

3-1. 완전 최소분할생성

[4]에서 제안된 관리제어기법은 시스템의 구축과 운영을 고려한 적절한 비용함수의 채택하고 용이한 관리제어기 설계기법을 제시했으나 약간의 문제점을 가지고 있다. 그것은 바로 플랜트를 고려할 때 일반적인 관리제어이론에서 다루고 있는 표기상태를 전혀 고려하지 않았다는 점이다. 이러한 문제점은 그림 2의 간단한 플랜트의 경우로도 쉽게 지적될 수 있다.

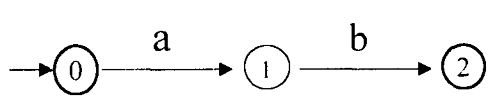


그림 3. 표기언어가 보장되지 않는 예.

만일 상태 0, 2가 의도상태이고 상태 1이 비의도상태이며 사건집합에 정의된 비용 값이 상태집합에 정의된 벌칙비용에 비해 무시할 만큼 작다고 하면 최적 관리제어기의 설계 결과는 $S(0) = a$ 와 같다. 그러나 만일 상태 2가 표기상태라고 하면 이렇게 설계된 관리제어기를 가지고는 표기언어 ab를 얻을 수가 없다. 이러한

관리제어기가 설계되는 이유는 바로 관리제어기 설계 시 애초에 표기상태를 염두에 두지 않았기 때문이다. 간단한 예를 들어 설명하자면, "연료의 소비를 최소화하며 자동차를 운행하라" 라는 문제에 대해 "자동차를 움직이지 않으면 된다"와 같은 사소한 해를 구한 것과 같다. 이는 문제에 자동차 운행의 목적이 정의되어 있지 않기 때문이다. 하지만 예를 들어 "A지점에서 B지점까지" 라는 자동차 운행의 목적이 문제 내에 정의가 되어 있다면 이러한 해를 얻지는 않았을 것이다. 그러므로 최적 관리제어기 설계시에도 최소분할생성을 구할 때 플랜트 운영의 목적이 되는 표기언어를 염두에 둔 제약조건이 추가되어야 한다. [6]에서는 완전 최소분할생성(complete min-cut)이라는 개념을 도입하여 이러한 문제점을 해결하고 있다.

정의 3.1[6] (완전 최소분할생성) : 플랜트 G로부터 유도된 흐름망의 최소분할생성은 그 최소분할생성에 의해 남게되는 플랜트 G의 부분그래프(subgraph)가 접근가능하지 못하거나 상호접근가능하지 못한 상태[7]를 제거했을 때 트림(trim)이 되면 완전 최소분할생성이다.

완전 최소분할생성을 구하는 법을 제시하기 위해 다음의 보조정리 3.1을 통해 비교차 부최소분할생성(non-crossing submin-cut)을 소개한다.

보조정리 3.1[6] (비교차 부최소분할생성) : (Q_s, Q_t)를 각각 $s \in Q$ 와 $t \in Q$를 포함하는 하나의 분할생성에 의한 분할이라고 생각하고, s', t'을 Q_s에 포함된 두 절점이라고 생각하자. 그러면 Q_s를 $(Q_{s'}, Q_{t'})$으로 분할하면서 Q를 (Q_s, Q_t)로 분할하는 분할생성과 서로 교차하지 않는 부최소분할생성이 존재한다.

플랜트 G로부터 유도된 흐름망이 주어졌을 때 우리는 다음의 명제 3.1을 통해 완전 최소분할생성을 구할 수 있다.

명제 3.1[6] : 플랜트 G로부터 유도된 흐름망에서 (Q_s, Q_t)로 분할하는 최소분할생성이 완전 최소분할생성이면 Q_s를 (Q_{q_0}, Q_{q_m})으로 분할하는 부최소분할생성의 용량이 0보다 크고 그 역도 성립한다. 여기서 q_0는 G의 초기상태이고 q_m은 표기상태 중 하나이다.

이제 최적 관리제어기를 설계할 때 G로부터 유도된 흐름망의 단순한 최소분할생성을 구하지 말고 완전 최소분할생성을 구하면 관리제어기의 순수비용도 최소화하면서 플랜트의 표기언어 생성도 보장할 수 있게 된다.

3-2. 계층적 최적 관리제어기법

3-2-1. 생성보장 계층적 최적 정상 부언어

3.1절에서 제시된 완전 최소분할생성을 이용하면 우리는 표기언어의 생성을 보장받으면서 최적 관리제어기를 설계할 수 있다. 이를 통해 정상언어 K가 주어지고 생성보장 최적 정상 부언어(achievable optimal legal sublanguage)를 구할 수 있다고 가정했을 때(이때 생성보장의 의미는 제어가능하고 관측가능하다는 의미) 우리는 다음과 같은 문제를 생각해 볼 수 있다. 만일 시스템의 일부에 고장이 발생하여 더 이상 최적 정상 부언어를 생성하도록 제어할 수가 없다면 어떻게 대처해야 겠는가? 계층적 최적제어 기법이란 이런 상황까지 고려하여 사전에 시스템의 고장까지를 고려하여 최적 정상 부언어 뿐만 아니라 부최적 정상 부언어(suboptimal legal sublanguage)들을 구한 후 자료화하여 저장해 놓았다가 시스템에 고장이 발생하면 관리제어기를 다시 재조정하여 부최적 정상 부언어를 생성하도록 하는 제어기법의 개념이다. 이러한 상황은 그림 3을 통해 살펴볼 수 있다.

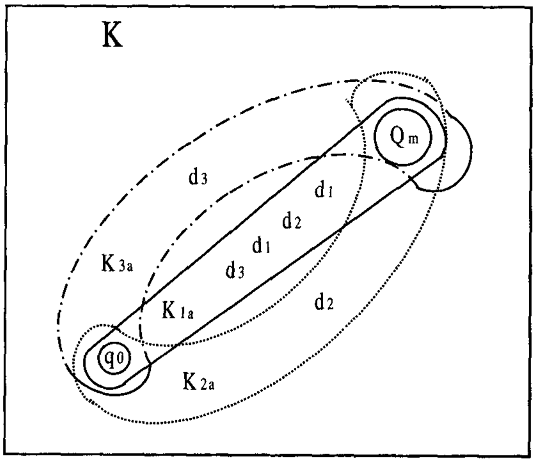


그림 4. 생성보장 계층적 최적 정상 부언어들.

그림 3에서 K_{1a}가 생성보장 최적 정상부언어이고 첫 번째 부최적 정상부언어 K_{2a}, 두 번째 부최적 정상 부언어 K_{3a}까지 얻어놓았다고 가정을 하자. 시스템의 고장이 없을 경우는 시스템을 모든 K_{ia} $(i=1,2,3)$를 따라 동작시키는 것이 가능하고, 이때는 당연히 최적 정상 부언어인 K_{1a}가 선택될 것이다. 그러나 시스템에 고장 1 ($d1$)이 발생한 경우는 K_{1a} 는 불가능해지므로 가능한 부최적 정상 부언어는 K_{2a}, K_{3a}가 되며 이때는 K_{2a}를 따라 시스템이 동작되도록 관리제어기는 재조정된다. 고장 2 ($d2$)가 발생한 경우는 가능한 부최적 정상 부언어가 K_{3a}밖에 없으므로 관리제어기는 K_{3a}에 맞춰서 재조정된다. 다음의 정리 3.1은 최적 정상 부언어와 부최적 정상 부언어를 구하는 체계적인 방법을 제시하고 있다.

정리 3.1[6] : 정상언어 K의 생성보장 최적 정상 부언어 K_{1a}는 K의 인식기 R_K로부터 유도되는 흐름망 N_{R_K}에서 초기상태와 표기상태들을 의도상태라고 본 새로운 벌칙비용을 적용했을 때 완전 최소분할생성에

의해 설계된 관리제어기에 의해 제어되는 플랜트가 생성하는 언어가 된다. 즉,

$$K_{1a}=L(N_{R_K}\text{의 완전 최소분할생성의 해})$$

라고 할 수 있다. 만일 K_{na}가 K의 n번째 부최적 정상 부언어라고 하면 $K_{(n+1)a}$는 다음과 같이 나타난다.

$$K_{(n+1)a}=K'_{(n+1)a}-\sum_{i=1}^{n}K_{ia}.$$

여기서 N_{R_K}에 대한 제어 비용이 $c(\cdot, e^{j_1}_{K'_{1a}})=c(\cdot, e^{j_2}_{K'_{2a}})=\cdots=c(\cdot, e^{j_n}_{K'_{na}})$로 주어졌을 때 $K'_{(n+1)a}=L(\min_{j_i\in[1,m_K]}(N_{R_K}\text{의 완전 최소분할생성 해}))\ i\in[1,n]$가 되고, $K'_{1a}=K_{1a}$, $e^{j_i}_{K'_{ia}}\in\{K'_{ia}$에 해당하는 분할생성에 속하는 연결선들$\}$, 그리고 $M_{K'_{ia}}=|E_{K'_{ia}}|$이다. 만일 $K'_{(N+1)a}=K$이면 N개의 K_{ia}가 존재하는 것이다.

정리 3.1의 내용을 간단히 다시 설명하면 우선 최적 정상 부언어 K_{1a}를 구하고 나서 K_{1a}에 해당하는 완전 최소분할생성에 속하게 되는 연결선에 대한 비용을 하나씩 차례로 ∞로 바꿔주어 분할생성이 그 연결선을 포함하지 못하게 하면서 다시 완전 최소분할생성을 구하여 그중 최소의 용량을 가지는 완전 최소분할생성에 해당하는 부언어에서 K_{1a}를 빼준 언어를 K_{2a}로 잡는 것이다. K_{3a}는 다시 K_{1a}와 K_{2a}에 해당하는 분할생성에 해당하는 연결선들에 각각 하나씩 ∞의 비용을 부가하여 K_{2a}를 구할 때와 같은 방식으로 구하게 된다. 이렇게 반복하면 다음 차례의 부최적 정상 부언어의 크기는 점점 커지게 되며 결국 주어진 정상 언어 K와 같아지게 된다. 만일 $K'_{(N+1)a}=K$가 되면 K_{Na}가 K내에서 구할 수 있는 부최적 정상 부언어가 된다.

[예] 그림 4와 같은 K의 인식기를 생각해 보자. 초기상태는 상태 0이고 표기상태는 상태 8이고 이 두 상태가 의도상태이다. 각 상태의 벌칙비용은 표 1에 주어져 있으며 $\beta,\gamma>0$이다. 사건집합 Σ는 제어가능한 사건집합 $\Sigma_c=\{a,b,c\}$와 $\Sigma_{uc}=\{d,e\}$로 이루어져 있다. 제어 가능한 사건의 억제를 위한 비용은 α로서 $\alpha\ll\beta$이고 제어불가능한 사건의 억제를 위한 비용은 당연히 ∞가 된다.

이런 상황하에서 생성보장 계층 최적 정상 부언어들인 K_{ia}들은 정리 3.1을 통해 구할 수 있으며 표 2와 같은 결과를 얻게 된다.

표 1. 그림 4의 상태들에 대한 벌칙비용.

상태	벌칙비용	상태	벌칙비용	상태	벌칙비용
0	$-\gamma$	3	2β	6	β
1	β	4	β	7	2β
2	β	5	2β	8	$-\gamma$

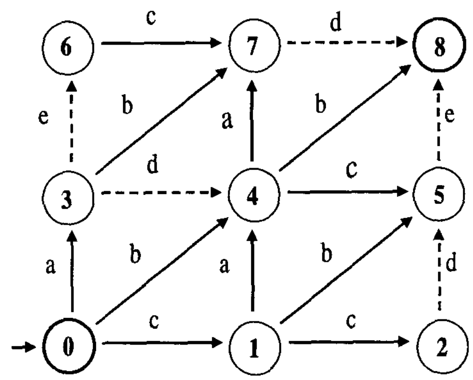


그림 5. K의 인식기.

표 2. 생성보장 계층적 최적 정상 부언어들.

계층	부언어	순수비용
K_{1a}	b^2	$\beta+4\alpha$
K_{2a}	cab	$2\beta+6\alpha$
K_{3a}	cbe	$3\beta+4\alpha$
K_{4a}	c^2de	$4\beta+4\alpha$
K_{5a}	$cace+ca^2d$	$4\beta+6\alpha$
K_{6a}	$a(d(ce+b+ad)+bd+ecd)$	$5\beta+2\alpha$

3-2-2. 비생성보장 계층적 최적 정상 부언어

만일 3.2.1절에서와 같이 주어진 K내에서 생성보장 최적 정상 부언어가 존재하지 않는 경우나 시스템의 고장에 의해 더 이상 생성보장 최적 정상 부언어로 플랜트 동작을 제한하는 것이 불가능한 경우는 어떻게 할 것인가? 이러한 경우는 우선 비생성보장 최적 정상 부언어(non-achievable optimal legal sublanguage)에 따라 관리제어기를 조정하여 시스템을 최적화 시킨다. 그러나 이런 경우는 제어 불가능한 사건의 발생에 의해 플랜트의 동작을 최적 정상 부언어로 제한 할 수 없으므로 플랜트 동작이 최적 정상 부언어를 벗어나는 경우에는 주어진 상황에서 다시 다른 비생성보장 부최적 정상 부언어에 맞춰 관리제어기를 재조정하여 시스템의 최적화를 도모한다. 이렇게 변화하는 상황에 맞게 관리제어기를 재조정해 주기 위해서는 플랜트의 동작이 최적 정상 부언어를 벗어나는 모든 경우에 대한 부최적 정상 부언어가 미리 오프라인(off-line)으로 계산이 되어 있어야 한다.

언어 K의 인식기 상태들은 제어불가능한 사건의 발생에 의해 K를 벗어날 가능성 여부에 따라 유도가능상태(tractable state)와 유도불능상태(intractable state)로 나뉠 수 있다.

정의 3.1 [6](유도불능상태) : 어떤 상태 q가 정상언어 L의 인식기 R_L의 도달가능한 상태집합 $Re(R_L)$에 속

하지만 $\delta(q,\sigma) \notin Re(R_L)$, $\sigma \in \Sigma_{uc}$이면 상태 q는 언어 L에 대해 유도불능상태가 된다.

정의 3.2 [6](유도가능상태) : 어떤 상태 q가 언어 L에 대해 유도불능상태가 아니면 유도가능상태이다.

다음의 그림 5는 비생성보장 계층적 최적 정상 부언어들을 이용하여 시스템을 최적화시키는 예를 보여준다. 생성보장 최적 정상 부언어가 존재하지 않는 경우 그림 5와 같은 비생성보장 계층적 최적 정상 부언어들 K_i를 구할 수 있으면 시스템을 구동을 시작할 때는 K_1에 맞춰서 관리제어기를 조정해 주고 K_1에 대한 유도불능상태에서 제어 불가능한 사건의 발생에 의해 플랜트의 동작이 K_1을 벗어나는 경우는 다시 관리제어기를 K_2에 맞춰 재조정하게 된다. 물론 플랜트 동작이 K_1을 벗어나지 않는 경우는 관리제어기의 재조정은 필요가 없다. 또 K_2에 대한 유도불능 상태에서 제어 불가능한 사건의 발생에 의해 플랜트의 동작이 K_2를 벗어나는 경우에는 다시 관리제어기를 K_3에 맞춰서 재조정 해주게 된다. 이러한 비생성보장 계층적 최적 정상 부언어들은 다음의 정리 3.2에 의해 구할 수 있다.

정리 3.2 [6] : 초기상태(혹은 $i>1$인 경우 유도불능상태로부터)의 비생성보장 최적 정상 부언어 K_i는 K의 인식기 R_K로부터 유도되는 흐름망 N_{R_K}에서 초기상태(혹은 유도불능 상태에서 벗어나게 되는 상태)와 표기상태들을 의도상태라고 본 새로운 벌칙비용을 적용하고, 제어 불가능한 사건에 대해서 제어 가능한 사건과 똑같은 비용을 적용했을 때 완전 최소분할생성에 의해설계된 관리제어기에 의해 제어되는 플랜트가 생성하는 언어가 된다. 즉,

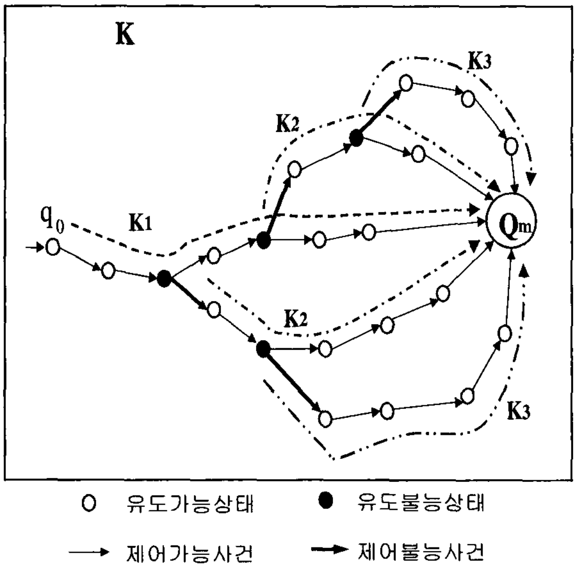


그림 6. 비생성보장 계층적 최적 정상 부언어들.

$K_i = L(c(\cdot,\sigma_{uc}) = c(\cdot,\sigma_c) \neq \infty$ 일대의 N_{R_K}의 완전 최소분할생성의 해)

라고 할 수 있다. 이때 $\sigma_c \in \Sigma_c, \sigma_{uc} \in \Sigma_{uc}$이다.

[예] 그림 5와 같은 K의 인식기이다. 초기상태는 상태 0이고 표기상태는 상태 16이다. 각 상태의 벌칙비용은 표 3에 주어져 있으며 $\beta, \gamma > 0$이다. 사건집합 Σ는 제어가능한 사건집합 $\Sigma_c = \{a, b, c\}$와 $\Sigma_{uc} = \{d\}$로 이루어져 있다. 제어 가능한 사건과 제어 불가능한 사건에 대한 비용은 모두 α ($\alpha \ll \beta$)로 본다. 이 경우는 생성보장 계층적 최적 정상 부언어가 존재하지 않기 때문에 정리 3.2를 통해 비생성보장 계층적 최적 정상 부언어를 구한 결과가 표 4에 주어져 있다.

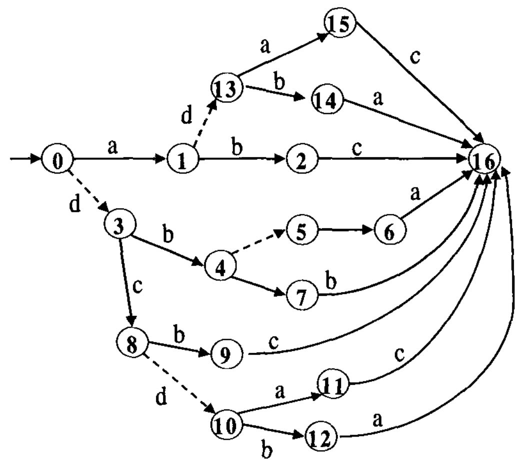


그림 7. K의 인식기.

표 3. 그림 6의 상태들에 대한 벌칙비용.

상태	벌칙비용	상태	벌칙비용	상태	벌칙비용
0	$-\gamma$	6	3β	12	2β
1	β	7	3β	13	β
2	β	8	2β	14	2β
3	β	9	β	15	4β
4	3β	10	β	16	$-\gamma$
5	3β	11	β		

표 4. 비생성보정 계층적 최적 정상 부언어들.

계층	부언어	순수비용
K_1	abc	2β
K_2 (*at* 3)	cbc	2β
K_3 (*at* 10)	ac	4β
K_4 (*at* 13)	ba	3β

앞서 설명한 계층적 관리제어 기법의 전체적인 구조는 그림 7과 같이 도식화할 수 있다. 계층적 관리제어 기법

에서는 우선 오프라인으로 정상언어 K에 대해 생성보장 계층적 최적 정상 부언어 K_{ia}와 비생성보장 계층적 최적 정상 부언어 K_i를 계산하여 이를 자료화시켜 놓은 뒤 이 자료와 플랜트에서 발생하는 사건열을 관측하면서 적절히 관리제어기를 재조정하면서 시스템을 최적화한다.

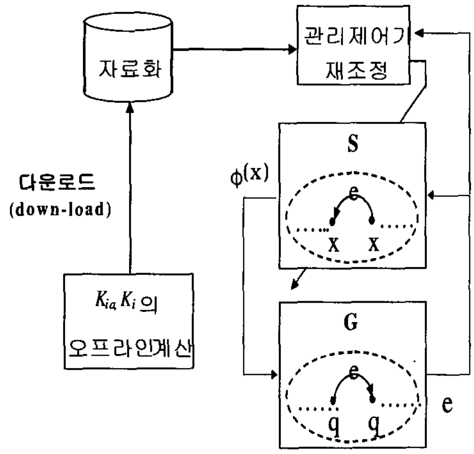


그림 7. 계층적 최적 관리제어의 체계.

4. 최적 관리제어기법(유형 II)

이 절에서는 Sengupta와 Lafortune이 제안한 최적 관리제어기법[2], [5]에 대해 살펴본다.

4.1 비용함수

[2], [5]에서는 [4]에서와 마찬가지로 두 가지의 비용을 고려하고 있다. 양자 모두 사건의 집합에 정의되는데, 한 경우는 [1], [3]에서 정의되었던 것과 같은 의미의 비용으로 플랜트가 사건을 발생시키기 위해 필요한 비용을 고려한 것이고, 다른 경우는 [4]에서와 비슷하게 관리제어기가 사건의 발생을 억제하기 위해 필요한 비용을 고려한 것이다.

정의 4.1 [2], [5] : 다음의 c_e, c_c는 각각 사건의 비용과 제어의 비용이다.

$$c_e : \Sigma \rightarrow R^+ \cup \{0\},$$

$$c_c : \Sigma \rightarrow R^+ \cup \{0,\ \infty\}.$$

만일 $\sigma \in \Sigma_{uc}$이면 $c_c(\sigma) = \infty$가 된다. 즉, 제어불가능한 사건은 억제될 수 없음을 의미한다.

이제 하나의 사건열에 대한 비용함수를 정의하기 위해 수학적인 표기법 몇가지를 소개한다. $|\cdot|$은 집합의 크기나 사건열의 길이를 나타낸다. 관리제어 이론에서 생각되는 제어법칙(관리제어기) $\Psi = (\psi_1, \psi_2, \cdots \psi_{|\Sigma|})$, $\psi_j : \overline{L_m} \rightarrow \{0, 1, \tau\}$는 어떤 사건열이 발생한 이후 각각의 사건의 허용, 억제 여부를 나타낸다. 만일 $\psi_j(s) = \tau$이면 σ_j는 사건열 s가 발생한 후의 상태에서 정의가 되어 있지 않아 억제, 허용여부를 고려하지 않아도 된다는 의미이고, $\psi_j = 0$ (혹은 1)이면 사건열 s가 발생한 후 σ_j를 억제(혹은 허용)한다는 의미이다. Ψ_L은 플랜트의 동작을 언어 L내로 제한해 주는 제어법칙을 의미한다. 함수 $p_j(\cdot)$은 사건열 가운데 길이 j인 접두어(prefix)를 나타내고, $p_0(\cdot) = \varepsilon$이 된다. 투영함수(projection function) Π_0는 다음과 같이 정의되는데, $|\Sigma|$-쌍($|\Sigma|$-tuple)으로 이루어진 제어법칙으로부터 억제되는 사건들의 집합을 나타내주는 함수이다.

$$\Pi_0 : \{0, 1, DC\}^{|\Sigma|} \rightarrow 2^{\Sigma},$$

$$\Pi_0((x_1, \cdots, x_{|\Sigma|})) = \{\sigma_j \in \Sigma : x_j = 0,\ 1 \leq j \leq |\Sigma|\}.$$

이러한 수학적 표기법을 이용하여 다음과 같이 사건열에 대한 비용함수를 정의한다.

정의 4.2 [2], [5] : $s = \sigma_1, \cdots, \sigma_{|s|}$를 $A_m(\subseteq L_m)$에 속하는 사건열이라 하자.

(1) 부언어 A_m에 대한 사건열 s의 비용 $c(s, A_m)$은

$$c(s, A_m) = \sum_{j=1}^{j=|s|} c_e(\sigma_j) + \sum_{j=0}^{j=|s|} \sum_{\sigma \in \Pi_0(\Psi_{A_m}(p_j(s)))} c_c(\sigma) \text{이고}$$

(2) A_m에 대한 목적함수 $c_{sup}(A_m)$는

$$c_{sup}(A_m) = sup_{s \in A_m} c(s, A_m) \text{ 이다.}$$

정의 4.2 (1)의 의미를 간단히 살펴보면, 플랜트의 동작을 A_m내로 제한하는 관리제어기에 의해 제어되는 플랜트에서 발생하는 사건열 s에 대한 비용은 사건열 내의 사건들이 발생하는데 필요한 비용들의 합과 사건열 내의 초기상태에서부터 각 사건들이 발생한 후에 플랜트의 동작이 A_m을 벗어나지 못하게 하기위해 억제해 주어야 하는 사건들에 대한 억제 비용의 합을 더한 것이 된다. 그러므로 같은 사건열이라 해도 어떤 관리제어기의 제어되고 있느냐에 따라 비용은 달라지는 것이다. 그리고 (2)에서는 A_m 내의 사건열들의 비용들 중 최대값을 A_m의 목적함수로 잡는다는 의미이다.

4.2 최적 관리제어기의 설계

[2], [5]에서 고려한 최적 관리제어문제는 최소의 목적함수를 갖는 제어가능하고 $L_m(G)$-닫힘성[7]을 갖는 최적 부언어 A_{om}을 찾아 그에 해당하는 관리제어기를 설계하는 문제이다. 그러나 불행히도 이러한 성질을 만족하는 부언어 A_{om}은 유일하게 결정되지 않는다. 하지만 만일 우리가 최소의 목적함수를 갖는 부언어 A_{om1}과 A_{om2}를 찾았다고 하면 $A_{om1} + A_{om2}$ 역시 최소의 목적함수를 갖는 부언어가 된다. 이런 관계가 성립하는 이유는 다음과 같다. 관리제어시스템에서 어떤 하나의 사건열이 발생했다고 가정했을 때 그 사건열이 발생하는

데 필요한 비용은 항상 같지만 관리제어기가 관리제어 시스템의 동적특성을 주어진 부언어 범위 이내로 제한하기 위해 사건의 발생을 관측하면서 각 사건발생 관측 후에 억제해주어야 할 사건의 수는 주어진 부언어에 따라 달라지며 부언어가 커질수록 억제해주어야 하는 사건의 수는 같거나 줄어들기 때문에 목적함수 역시 같거나 줄어들 수 밖에 없다. 따라서 A_{om1}과 A_{om2}가 최소의 목적함수를 갖는 부언어라고 할 때 $A_{om1}+A_{om2}$에 대한 목적함수는 A_{om1}과 A_{om2}에 대한 목적함수와 비교했을 때 작거나 같을 수 밖에 없지만 A_{om1}과 A_{om2}이 최소의 목적함수를 갖는다고 가정했으므로 $A_{om1}+A_{om2}$이 더 작은 목적함수를 가질 수는 없다. 결국 $A_{om1}+A_{om2}$ 역시 A_{om1}나 A_{om2}와 같은 목적함수를 갖게되는 것이다. 그러므로 우리는 다음과 같은 정리 4.1을 생각 할 수 있다.

정리 4.1 [2], [5] : 만일 최적 부언어가 존재하면 유일한 최고 최적 부언어가 존재한다.

서론에서 언급했던 바와 같이 관리제어이론이 제안된 초기에는 최소 제한적 관리제어기가 최적 관리제어기로 생각되었다. 그러므로 최소의 목적함수를 갖는 부언어가 여러 개가 존재한다면 최고 부언어에 맞추어 관리제어기를 설계하는 것은 당연하다고 할 수 있다. 그러나 목적함수의 최소화와 함께 최소제한성을 모두 만족시키는다는 것은 쉽지 않은 문제이다. 이는 L/s를 언어 L의 사건열 s에 대한 후언어[7]라고 할 때 최적 부언어 A_{om}이 $s\in\overline{A_{om}}$인 모든 s에 대해 A_{om}/s가 L_m/s의 최적 부언어가 된다는 동적계획법의 원리(principle of dynamic programming)을 항상 만족시키는 것은 아니어서 최적 부언어를 효율적으로 계산하기 위해 동적계획법(dynamic programming)을 이용할 수가 없기 때문이다. 따라서 [2], [5]에서는 다음의 정의 4.3에서 동적계획법의 원리를 만족시키기 때문에 동적계획법을 통해 계산할 수 있는 DP-최적 부언어를 정의하고 최고 DP-최적 부언어를 구하는 방법을 제시했다(DP는 동적계획법을 이용하여 효율적으로 계산할 수 있다는 의미).

정의 4.3 [2], [5] : $A_{DO}\subseteq L_m$이 DP-최적 부언어이기 위한 필요충분조건은 A_{DO}가 최적이고 $s\in\overline{A_{DO}}$인 모든 s에 대해 A_{DO}/s가 L_m/s의 최적 부언어가 된다는 것이다.

물론 모든 경우에 DP-최적 부언어가 존재하는 것은 아니다. 그러나 $|\Sigma|<\infty$이면 DP-최적 부언어가 존재하고 또한 유일한 최고 DP-최적 부언어가 존재한다는 것이 다음의 정리 4.2에 소개되어있다.

정리 4.2 [2], [5] : $|\Sigma|<\infty$인 경우, 만일 최적 부언어가 존재하면 DP-최적 부언어가 존재하며, 또한 유일한 최고 DP-최적 부언어 $L_{DO}^{\uparrow}$가 존재한다.

결국 $L_{DO}^{\uparrow}$에 따라 관리제어기를 설계하게 되면 효율적으로 계산할 수 있는 범위 내에서의 최소제한적 최적 관리제어기를 설계하게 되는 것이다. $L_{DO}^{\uparrow}$를 구하는 알고리즘은 [5]에 소개되어있으며, $L_{DO}^{\uparrow}$를 구하는 알고리즘의 계산복잡도는 $||L_m||=n$일 때, $O(n^2|\Sigma|\log(|\Sigma|)+n^3\Sigma)$로 주어진다는 것이 증명되어 있다.

5. 결론

본 논문에서는 여러가지 최적 관리제어기법에 대해 살펴보았다. 최적 관리제어기법에서는 비용함수를 어떻게 정의하느냐에 따라 그 접근방법이 모두 달라 일관된 하나의 프레임내에서 소개할 수 없었다. 본 논문에서 살펴본 두 가지 유형의 최적 관리제어기법은 플랜트 뿐 아니라 관리제어기의 동작까지도 고려한 가장 일반적인 기법이라고 말할 수 있다. 이러한 최적 제어관리기법의 개발은 시스템의 최적화에 주된 목적이 있겠으나 개발된 기법과 적절한 비용의 할당을 통해 기존의 관리제어이론에서 생각되었던 문제들도 통일된 프레임 안에서 해결하려는 부가적인 목적도 포함되어 있다. 이러한 사실은 본 논문에서 소개하지는 않았지만 [4]에서는 정상언어가 주어졌을 때 최적 관리제어기 설계기법을 이용하여 최고 제어가능 표준 부언어 등을 계산할 수 있는 적절한 플랜트의 변형방법과 제어 소모비용, 제어 벌칙비용의 정의방법을 제시하고 있다는 점에서 살펴 볼 수 있다.

연속변수시스템에 대한 제어이론의 역사적 발전과정에서 여러 기초적인 개념들을 토대로 최적제어의 이론적 발전이 이어져 온 것처럼 이산사건시스템에 대한 관리제어이론에서도 논리적인 사양을 만족하는 관리제어기 설계를 위한 기본적인 틀이 마련된 상태라고 봤을 때, 주어진 사양을 만족시키는 것에서 더 나아가 정량적인 시스템 성능의 최적화를 꾀하는 최적 관리제어 기법의 연구는 필수 불가결한 것이며 앞으로도 많은 연구가 뒤따를 것으로 기대된다.

참고문헌

[1] K. M. Passino and P. J. Antsaklis, "On the optimal control of discrete event systems", *Proc. IEEE Conf. on Decision and Control*, Tampa, FL, pp. 2713-2718, 1989.

[2] R. Sengupta and S. Lafortune, "A deterministic optimal control theory for discrete event systems," *Proc. 32nd IEEE Conf. On Decision and Control,* San Antonio, Texas, pp. 1182-1187, 1993.

[3] D. Ionescu and J.-Y. Lin, "Optimal supervision of discrete event systems in a temporal logic framework," *IEEE Trans. on Systems, Man and*

Cybernetics, vol. 25, no. 12, pp. 1595-1605, 1995.

[4] R. Kumar and V. K. Garg, "Optimal supervisory control of discrete event dynamical systems", *SIAM J. of Control and Optimization*, vol. 33, pp. 419-439, 1995.

[5] R. Sengupta and S. Lafortune, "An optimal control theory for discrete event systems", *SIAM J. of Control and Optimization*, vol. 36, no. 2, pp. 488-541, 1998.

[6] K.-H. Cho and J.-T. Lim, "Layered optimal supervisory control of discrete event dynamic systems", *Int. J. of Systems Science*, vol. 30, no. 4, pp. 395-405, 1999.

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

[8] T. C. Hu, Combinatorial Algorithms, Addison-Wesley, Menlo Park, CA, 1982.

이 문 상

1996년 한국과학기술원 전기및전자공학과 졸업. 동대학원 석사(1995), 현재 동대학원 박사과정. 관심분야는 supervisory control of discrete event systems, control and automation of manufacturing systems, circuit test, supervisory control of semiconductor manufacturing systems, traffic management in IVHS 등.

조 광 현

제어 · 자동화 · 시스템 공학회지 제6권 제3호 참조. 현재 울산대학교 전기전자 및 자동화공학부 전임강사.

임 종 태

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