논문 2008-45SC-4-2

비결정 모델에 대한 비동기 순차 회로의 교정 제어 II: 제어기 설계

(Corrective Control of Asynchronous Sequential Machines for Nondeterministic Model II: Controller Design)

양 정 민*

(Jung-Min Yang)

요 약

본 논문에서는 비동기 순차 머신의 교정 제어 문제를 다룬다. 교정 제어는 머신의 동작을 주어진 모델의 동작과 일치시키도록 하는 모델 매칭을 실현하는 제어를 말한다. 본 논문의 주요 목적은 비동기 순차 머신이 추종해야 하는 모델의 형태가 비결정적일 때, 즉 여러 개의 결정적 모델의 합으로 주어질 때 교정 제어기를 설계하는 일이다. 본 논문에서는 이전 논문에서 정의된 비결정 모델의 표현 방법 및 비결정 모델에 대한 모델 매칭 문제 정의를 요약한다. 도달가능성 행렬을 이용하여 교정 제어기가 존재할 필요충분조건을 제시하고 제어기가 존재할 경우 그 설계 과정을 기술한다. 또 예제를 통해서 제어기 설계의 적용 가능성을 검증한다.

Abstract

The problem of controlling asynchronous sequential machines is addressed in this paper. Corrective control means to make behavior of an asynchronous sequential machine equal to that of a given model. The main objective is to develope a corrective controller, especially when a model is given as nondeterministic, or a set of reference models. We first review representation of nondeterministic models and model matching problems with nondeterministic models, which are presented in the companion paper. We then propose necessary and sufficient conditions for the existence of corrective controllers and describe their design procedure. To show the applicability, the proposed control scheme is demonstrated in an example.

Keywords : Asynchronous Sequential Machines, Corrective Control, Model Matching, Nondeterministic Model

I. 서 론

본 논문의 목적은 피드백 제어를 통해서 비동기 순차 머신(Asynchronous Sequential Machine)의 동작을 원하는 목적에 맞게 바꾸어주는 교정 제어기를 설계하는 일이다. 전역 클럭(clock) 없이 동작하는 비동기 순차 머신의 설계에 대한 연구는 1950년부터 활발하게 진행되어 왔으며 최근 실용적인 설계 기법들이 개발되면서 다시 주목을 받고 있다[1~2]. 본 논문에서는 비동기 순차 머신을 제어 대상 플랜트(plant)로 인식하고, 출력을 피드백(feedback) 받아서 원하는 동작을 보이는지 관측한 다음 외부 입력과 연계하여 제어 입력을 만들어내는 제어기를 사용한다. Hammer 등이 제안한 이러한 교정 제어(Corrective Control)는 전통적인 자동 제어의 모델 추종 제어(model following control) 원리를 비동기 순차 머신에 적용한 개념으로 크리티컬 레이스(critical race)[3], 무한 순환(infinite cycle)[4] 등 비동기 순차 머신에 내재되어 있는 여러 가지 해저드(hazard)를 없애는 데 성공적으로 적용되었다.

본 논문의 주요 목적은 제어 대상 비동기 순차 머신이 추종해야 하는 모델이 비결정적으로 주어질 때 모델 매칭 문제를 푸는 제어기를 설계하는 일이다. 앞 논문

* 정회원, 대구가톨릭대학교 전자공학과
(Department of Electrical Engineering, Catholic University of Daegu)
접수일자: 2007년10월25일, 수정완료일: 2008년7월8일

[5]에서는 비결정적 모델을 여러 개의 결정적 유한 상태 머신의 합으로 표현하는 방법과 더불어 비결정적 모델과 비동기 순차 회로의 동작이 일치하도록 만드는 제어기의 형태를 제안하였다. 또한 비결정 모델에 대한 모델 매칭 문제를 푸는 데 사용되는 도달가능성 행렬을 정의하고 그 계산 과정을 예시하였다.

본 논문에서는 비결정 모델에 대한 모델 매칭 문제를 해결하는 교정 제어기가 존재할 필요충분조건을 수학적으로 찾는다. 제어기의 존재 여부는 앞 논문에서 기술한 skeleton 행렬의 부등식으로 표현될 수 있다. 또 기존 연구[3~4] 결과를 바탕으로 이러한 교정 제어기가 존재할 경우 제어기의 설계 과정을 제시한다. 사례 연구를 통해서 제안된 교정 제어기를 직접 구현하고 그 적용 가능성을 입증한다.

Ⅱ. 교정 제어 시스템

1. 비동기 순차 머신

본 장에서는 비동기 순차 머신의 동작을 표현하기 위한 수학적 모델을 설명하고 앞 논문[5]에서 제시된 교정 제어기 구조를 요약한다. 논문에서 고려되는 비동기 순차 머신의 논리적 모델은 다음과 같은 유한 상태 머신(Finite State Machine)으로 표현된다.

$$\Sigma=(A,Y,X,x_0,f,h) \tag{1}$$

위 식에서 A는 입력 알파벳 집합, Y는 출력 알파벳 집합, X는 n개의 원소로 이루어진 머신의 상태 집합이다. x_0는 머신의 초기 상태이며 $f:X\times A\rightarrow X$와 $h:X\times A\rightarrow Y$는 각각 머신의 상태 천이 함수(state transition function)와 출력 함수(output function)이다.

$$\begin{aligned} x_{k+1}&=f(x_k,u_k),\\ y_k&=h(x_k,u_k),\ k=0,1,2,\ldots \end{aligned} \tag{2}$$

$u_0,u_1,u_2,\ldots$는 머신의 입력 시퀀스(sequence)이며 $u_0,u_1,u_2,\ldots$와 $y_0,y_1,y_2,\ldots$는 각각 머신의 상태와 출력 시퀀스를 가리킨다. 머신의 스텝(step) k는 입력이나 상태 변수가 변경되었을 때마다 1씩 증가한다.

본 논문에서는 비동기 순차 머신의 현재 상태가 출력값으로 나오는 입력/상태 머신(input/state machine)에서 대해서 다루기로 한다. 즉 출력 알파벳 집합 Y는 X와 일치하며 출력 함수도 아래와 같이 간단하게 표시된다.

$$y_k=x_k,\ k=0,1,2,\ldots \tag{3}$$

비동기 순차 머신은 현재 입력이 바뀌지 않는 한 그 값이 바뀌지 않는 '안정 상태(stable state)'와 순식간에 다음 상태로의 천이가 일어나는 불안정 상태(unstable state) 등 두 종류의 상태를 가진다. 외부 사용자에게는 비동기 순차 머신의 동작 과정 중 안정 상태에서의 모습만이 유효하므로 안정 상태에서만 비동기 순차 머신을 다룰 수 있는 수학적 모델이 필요하다. 비동기 순차 머신 Σ의 stable-state 머신 $\Sigma_{|s}$은 이 목적에 맞게 아래와 같이 정의된다.

$$\Sigma_{|s}=(A,Y,X,x_0,s,h) \tag{4}$$

위 식에서 상태 천이 함수 f 대신 사용되는 'stable recursion 함수' s는 다음과 같이 정의된다.

$$s:X\times A\rightarrow X,\quad s(x,u):=x',x\in X,\ u\in A \tag{5}$$

위 식에서 x'은 어떤 유효 조합(valid pair) (x,u), 즉 $f(x,u)$가 정의되는 상태와 입력 쌍 (x,u)의 다음 안정 상태(next stable state)--상태 x에 있는 머신 Σ에 입력 u가 들어왔을 때 머신이 도달하는 첫 번째 안정 상태를 의미한다. 머신 Σ의 상태 천이는 (x,u)에서 시작하여 $(x_1,u)\rightarrow\cdots\rightarrow(x_q,u)\rightarrow(x',u)$가 되며 (x',u)는 Σ의 안정 조합(안정 조합)이 된다. (x와 x' 사이에 q 개의 불안정 상태가 있다고 가정한다.) 또 입력 알파벳 대신 입력 스트링(string)을 s의 변수로 설정하면 다음과 같이 일반화할 수 있다.

$$s(x_0,u):=(\ldots s(s(s(x_0,u_0),u_1),u_2)\ldots,u_{m-1}),\quad u\in A^+ \tag{6}$$

2. 폐루프 시스템

두 개의 비동기 순차 머신 Σ와 Σ'의 stable-state 머신 동작이 서로 일치하면 두 머신은 stably equivalent하다고 정의된다[5]. stably equivalent한 두 머신은 현재 상태에서 동일한 입력이 들어오면 항상 동일한 출력을 낸다. 고려하는 비동기 순차 머신의 종류가 현재 상태를 출력값으로 가지는 입력/상태 머신이라면 stably equivalent한 두 머신의 상태 집합이 동일해야 하며, 임의의 상태에서 입력에 대한 상태 천이가 서로 일치해야 한다.

비동기 순차 머신에 대한 교정 제어의 기본 목적은

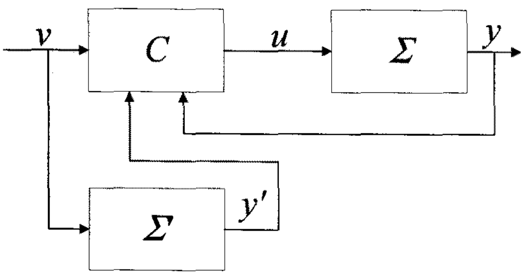


그림 1. 비결정 모델에 대한 교정 제어 시스템
Fig. 1. Corrective control system with nondeterministic model.

제어 대상 머신 Σ의 폐루프 시스템 동작을 어떤 stable-state 머신 Σ′와 stably equivalent되도록 바꾸는 일이다. 만약 어떤 상태에서 머신 Σ의 동작이 Σ′와 똑같다면 교정 제어기는 아무 일을 하지 않고 외부 입력을 그대로 머신에 전달한다. 그러나 머신 Σ의 동작이 Σ′와 차이를 보인다면 제어기는 외부 입력과 상태 피드백을 이용하여 제어 입력을 만들어내 머신 Σ에 전달함으로써 제어를 수행한다.

그림 1은 앞 논문[5]에서 제안된 비동기 순차 머신을 위한 교정 제어 시스템이다. 그림 1은 기존 연구[3-4]에서 사용되었던 폐루프 시스템에 모델 Σ′을 추가한 구조이다. 본 논문의 목적은 모델 Σ′의 동작이 비결정적일 때 적용될 수 있는 교정 제어기를 설계하는 것이므로 제어기 C가 모델 Σ′의 현재 상태(출력 y')를 알아야 할 필요가 생긴다.

교정 제어기 C는 외부 입력 v, Σ의 상태 피드백 $y(=x)$, 모델 Σ′의 현재 상태 $y'(=x')$를 입력으로 가지므로 C의 입력 집합은 $A\times X\times X$이다. C가 만드는 제어 입력 u는 머신 Σ의 입력으로 들어가므로 C의 출력 집합은 A이다. C를 유한 상태 머신으로 표현하면 아래와 같다.

$$C=(A\times X\times X,A,\Xi,\xi_0,\phi,\eta) \tag{7}$$

위 식에서 Ξ은 C의 상태 집합이며 ξ_0는 초기 상태, ϕ는 recursion 함수, η는 출력 함수이다. 그림 1에서 구현되는 폐루프 시스템을 Σ_c라고 표기하면 Σ_c는 C와 Σ로 이루어진 복합 시스템이므로 상태 집합 $X\times\Xi$, 입력 집합 A를 가진다. f_c와 h_c를 각각 Σ_c의 상태 천이 함수, 출력 함수라고 정의하면 폐루프 시스템의 유한 상태 머신 표현식은 아래와 같다.

$$\Sigma_c=(A,X,X\times\Xi,(x_0,\xi_0),f_c,h_c) \tag{8}$$

Σ가 입력/상태 머신이므로 Σ_c의 출력 함수 h_c는 C의 모든 상태 $\xi\in\Xi$와 Σ_c의 모든 유효 조합 (x,ξ,v)에 대해서 $h_c(\xi,x,v):=x$와 같이 되는 항등 함수(identity function)가 된다.

III. 비결정 모델에 대한 모델 매칭 문제

앞 논문[5]에서 제안된 비결정 모델은 비결정적 유한 상태 머신(Nondeterministic Finite State Machine: NFSM)으로 표현되거나 결정적 유한 상태 머신(Deterministic Finite State Machine: DFSM) 여러 개가 합성되어 만들어진다고 설정하였다. NFSM로 표현된 비결정 모델 Σ′을 p개의 DFSM로 분해하는 과정을 아래와 같이 표현한다[5].

$$\Sigma'=\Sigma_1'\parallel\Sigma_2'\parallel\cdots\parallel\Sigma_p' \tag{9}$$

$\Sigma'_i\,(i=1,\ldots,p)$의 stable recursion 함수를 각각 s'_i 라고 표기하면 전체 모델 Σ′의 recursion 함수 $s':A\times X\rightarrow P(X)$는 다음과 같이 얻어진다.

$$s'(x,a):=\bigcup_{i=1}^{p}s'_i(x,a),\ x\in X,\ a\in A \tag{10}$$

위 식과 같이 설정하면 모델 Σ′은 p개의 부-모델 $\Sigma'_i\,(i=1,\ldots,p)$에서 정의되는 모든 상태 천이를 포함한다. Σ′와 $\Sigma'_i\,(i=1,\ldots,p)$는 모두 제어 대상 비동기 순차 머신 Σ의 입력 집합 A와 상태 집합 X와 동일한 입력 및 상태 집합을 가진다.

앞 논문[5]에서는 비결정 모델 Σ′에 대하여 아래와 같은 두 가지 모델 매칭 문제를 정의하였다.

정의 1. 비결정 모델을 가진 입력/상태 머신의 온라인 모델 매칭 문제: 입력/상태 비동기 순차 머신 $\Sigma=(A,X,X,x_0,f,h)$와 비결정 stable state 모델 $\Sigma'=(A,X,X,x_0,s',h')$가 동일한 입력 집합 A와 상태 집합 X를 공유한다고 하자. 그림 1의 폐루프 시스템 Σ_{cls}이 Σ′와 stably equivalent하는 ($\Sigma_{cls}=\Sigma'$) 동작을 보이도록 만드는 교정 제어기 C가 존재할 필요충분조건을 구하고, C가 존재한다면 설계 알고리즘을 찾는다. □

정의 2. 비결정 모델을 가진 입력/상태 머신의 오프라인 모델 매칭 문제: 입력/상태 비동기 순차 머신

$\Sigma = (A, X, X, x_0, f, h)$와 $\Sigma' = (A, X, X, x_0, s', h')$가 동일한 입력 집합 A와 상태 집합 X를 공유한다고 하고 비결정 stable-state 머신 모델 Σ'은 $\Sigma' = \Sigma_1' \parallel \Sigma_2' \parallel \cdots \parallel \Sigma_p'$와 같이 분해 가능하다고 하자. 그림 1의 폐루프 시스템 $\Sigma_{c|s}$이 $\Sigma_i, i = 1, \ldots, p$ 중 적어도 한 개와 stably equivalent하는 동작을 보이도록 만드는 교정 제어기 C가 존재할 필요충분조건을 구하고, C가 존재한다면 설계 알고리즘을 찾는다. □

정의 1의 온라인 문제는 폐루프 시스템 $\Sigma_{c|s}$이 비결정 모델 Σ'에서 일어날 수 있는 가능한 모든 동작을 추종해야 하는 제한적인 문제이며, 정의 2의 오프라인 문제는 $\Sigma_{c|s}$가 비결정 모델 Σ'이 가지는 결정적 동작 중 적어도 하나와 일치되도록 제어기를 꾸미는 일이다.

교정 제어기의 존재 조건 및 설계 과정을 제안하기 전에 비동기 순차 머신의 안정 상태 사이의 도달가능성을 표시해주는 skeleton 행렬 정의를 다음과 같이 요약한다.

정의 3. skeleton 행렬: 입력/상태 비동기 순차 머신 Σ의 stable-state 머신을 $\Sigma_{|s} = (A, X, X, x_0, s, h)$이라고 하고 상태 집합 X의 원소를 $X = \{x_1, \ldots, x_n\}$라고 표시하자. Σ의 skeleton 행렬 $K(\Sigma)$은 아래와 같이 정의되는 $n \times n$차 정방 행렬이다.

$$K_{ij}(\Sigma) = \begin{cases} 1 & \exists u \in A^+ \ s.t. \ x_j = s(x_i, u) \\ 0 & \text{otherwise} \end{cases} \tag{11}$$

□

$K_{ij}(\Sigma) = 1$이면 $\Sigma_{|s}$의 상태를 x_i에서 x_j로 천이시키는 입력 스트링이 적어도 하나 존재한다.

Ⅳ. 교정 제어기 설계

정의 1과 정의 2에서 설정한 모델 매칭 문제를 풀기 전에 폐루프 시스템 $\Sigma_{c|s}$과 모델 Σ' 사이의 stable equivalence 관계를 다른 방법으로 표현한다. 식 (8)의 폐루프 시스템 Σ_c의 stable recursion 함수를 s_c라고 표기하면 $s_c : X \times \Xi \times A \rightarrow X \times \Xi$와 같은 매핑(mapping) 관계를 가진다. 폐루프 시스템 $\Sigma_{c|s}$의 임의 상태 $(x, \xi) \in X \times \Xi$에서 x만을 추출하는 투영 함수 Π_x를 아래와 같이 정의하자.

$$\Pi_x : X \times \Xi \rightarrow X \tag{12}$$

s_c와 Π_x를 이용하여 $\Sigma_{c|s}$와 Σ'의 stable equivalence를 재론하면 다음과 같다. Σ'에서 정의된 모든 유효 조합 $(x, v) \in X \times A$에 대하여 (x, ξ, v)가 $\Sigma_{c|s}$의 유효 조합이 되는 C의 상태 $\xi \in \Xi$가 존재하며 (x, ξ, v)는 아래 식을 만족한다.

$$\Pi_x s_c(x, \xi, v) = s'(x, v) \tag{13}$$

정의 1의 모델 매칭 문제는 식 (13)의 우변, 즉 모델 Σ'의 다음 안정 상태가 고정적이지 않고 실시간에 따라 여러 개의 상태 중 한 개가 선택되는 비결정적인 행동을 보일 때이다.

비동기 머신 Σ와 모델 Σ'가 모두 어떤 안정 조합 (x,a)에 있다고 가정하고 그림 1의 제어 과정을 개념적으로 설명한다. 외부 입력이 a에서 b로 바뀐다면 (그리고 (x,a)가 Σ'의 안정 조합이 아니라면) Σ'의 상태는 $s'(x, b)$ 의 원소 중 하나인 x'로 천이된다. 교정 제어기 C는 모델 Σ'의 출력값을 받아서 (그림 1 참조) 비동기 머신 Σ가 천이되어야 하는 목적 값 상태(=x')를 인지한다. 그런 다음 Σ의 상태를 x에서 x'로 천이시키는 입력 스트링을 Σ에 넣어 모델 매칭을 완성한다. 즉 C는 외부 입력 b를 Σ에 넣는 대신 생성한 제어 입력을 전달한다. C와 Σ가 모두 비동기 순차 머신이므로 이러한 교정 제어에 걸리는 시간은 극히 짧으며 이론적으로 0이다. 따라서 외부 사용자에게는 외부 입력 b가 들어와서 머신의 상태가 x에서 x'로 천이되는 교정 동작이 관측될 것이다.

이러한 동작이 성립되기 위한 기본 조건은 비동기 순차 머신 Σ가 상태 x에서 x'로 도달 가능해야 한다는 사실이다. 또 x'는 $s'(x, b)$의 임의의 원소이므로 Σ는 상태 x에서 $s'(x, b)$에 속하는 모든 상태로 도달 가능해야 한다. 이러한 도달가능성 관계는 skeleton 행렬의 부등식으로 표시할 수 있다.

또 하나 중요한 조건은 그림 1의 폐루프 시스템이 기본 모드(fundamental mode)[6], 즉 시스템 변수가 한 번에 하나씩만 바뀔 수 있다는 조건을 만족하면서 동작해야 한다는 점이다. 기존 논문[3]의 이론을 바탕으로 그림 1의 폐루프 시스템이 기본 모드로 동작하기 위해서 만족시켜야 할 조건을 아래와 같이 규명한다.

정리 1. Σ와 Σ', C는 그림 1과 같이 상호 연결된 비동기 순차 머신이다. 다음 조건이 만족되면 폐루프

시스템은 기본 모드로 동작한다.

(i) Σ가 안정 조합에 있지 않을 때 C의 출력 u는 그 값이 바뀌지 않는다.

(ii) C가 안정 조합에 있지 않을 때 Σ의 출력 y와 외부 입력 v, 그리고 기준 입력 y'은 그 값이 바뀌지 않는다. □

아래 정리는 비결정 모델에 대한 교정 제어기의 기본 모듈을 제공한다. 두 집합 α와 β가 있을 때 α/β를 α 원소에서 β 원소를 뺀 차집합이라고 정의한다. 또한 $s'[S]$를 집합 $S \subset X \times A$가 가지는 stable recursion 함수 s'의 치역이라고 표기한다.

정리 2. 입력/상태 비동기 순차 머신 $\Sigma=(A,X,X,x_0,f,h)$이 있다. $x_i \times U_i$를 어떤 상태 $x_i \in X$가 만드는 유효 조합 집합의 일부라고 하자. (U_i는 입력 A의 부분 집합). x_i가 stably reachable할 수 있는 q개의 상태 $x'_i (i=1,...,q)$가 존재한다. Σ의 폐루프 시스템 Σ_{cls}가 다음과 같은 stable recursion 함수 s'를 가지는 비결정 모델 $\Sigma'=(A,X,X,x_0,s',h')$와 stably equivalent되도록 하는 제어기 C가 존재한다.

(i) $s'[x_i \times U_i] = x'_1, \ldots, x'_q$

(ii) $s'(x,u) = s(x,u)$ for all other pairs, i.e., $\forall (x,u) \in X \times A / x_i \times U_i$

증명: 제어기 C의 목적은 머신의 상태가 x_i에 있을 때 $v \in U_i$인 외부 입력 v가 들어오면 머신을 $x'_i, \ldots x'_q$ 중의 하나로 상태 천이 시키는 일이다. 그림 1에서 머신이 x_i에 있을 때 $v \notin U_i$인 입력이 들어온다면 C는 아무런 일을 하지 않고 외부 입력 v를 그대로 제어 입력 u로 보낸다. v가 집합 U_i에 속한 값으로 바뀌는 순간 C는 먼저 모델 Σ'의 상태를 관측한다. Σ'의 상태가 x'_i라면 C는 머신 Σ의 상태를 x_i에서 x'_i으로 천이시키는 입력 스트링을 내보낸다. 정리 2의 가정에서 x'_i는 x_i로부터 stably reachable하기 때문에 이러한 입력 스트링은 항상 존재한다.

제어기 C의 recursion 함수와 출력 함수를 각각 ϕ와 η이라고 정의하고 $\xi \in \Xi$를 C가 가지는 임의의 상태라고 하자. 그림 1에서 C는 세 개의 입력을 가지므로 ϕ와 η를 각각 $\phi(\xi, x_r, (x,v))$, $\eta(\xi, x_r, (x,v))$라고 표기하자. $x_r (= y')$는 모델 Σ'의 현재 상태를 가리킨다. ξ_0를 C의 초기 상태라고 정의하면 ξ_0에서 제어기가 해야 할 일은 Σ가 상태 x_i의 안정 조합에 머무르는지를 알아서 다음 제어 동작을 하도록 준비하는 것이다. ξ_0에서 C의 동작을 아래와 같이 정의하자.

$$\begin{aligned} &\phi(\xi_0, x_r, (x,v)) := \xi_0, \\ &\qquad \forall (x,v) \in X \times A / x_i \times U(x_i) \\ &\phi(\xi_0, x_r, (x_i,v)) := \xi_0(x), \qquad \forall v \in U(x_i) \end{aligned} \tag{14}$$

위 식에서 $U(x_i) \subset A$는 x_i와 안정 조합을 이루는 모든 입력 알파벳을 가리킨다.

초기 상태에서 제어기 C는 외부 입력을 머신에 그대로 전달하는 역할만을 수행하므로 C의 출력 함수는 아래와 같이 정의된다.

$$\eta(\xi_0, x_r, (x,v)) := v, \ \forall (x,v) \in X \times A \tag{15}$$

위 설정에 따라서 머신 Σ가 상태 x_i와 안정 조합을 이루면, 즉 x_i에서 안정 상태로 있으면 제어기 C는 ξ_0에서 상태 $\xi_0(x)$로 천이된다. $\xi_0(x)$를 C의 transition 상태라고 부른다[5]. 이 상태에서 C는 외부 입력 v와 모델 Σ'의 다음 상태 값을 보고 머신 Σ의 상태 천이 함수를 바꾸는 동작을 시작할 수 있다. 하지만 제어기는 상태 $\xi_0(x)$에서 외부 입력 값이 변하기 전까지는 Σ의 상태를 그대로 유지시켜야 한다. 즉 Σ가 상태 x_i와 안정 조합을 계속 이루도록 해야 하므로 C의 출력 함수 η는 $\xi_0(x)$에서 아래와 같이 설정된다.

$$\eta(\xi_0(x), x_r, (x_i,v)) := t, \ \exists t \in U(x_i), \ \forall v \in A \tag{16}$$

위 식에서 t는 머신의 현재 상태 x_i와 안정 조합을 이루는 입력 알파벳 중의 하나로 선택된다. 이와 같이 제어 입력을 설정하면 머신의 동작은 외부 입력 v에 영향을 받지 않게 되어 차후 제어기 C가 Σ의 동작을 바꿀 수 있다.

Σ가 x_i와 안정 조합을 이루고 있을 때 U_i에 속한 외부 입력 v가 들어와서 모델 Σ'의 상태가 x'_i로 천이 되었다고 하자. 제어기는 머신이 현재 상태 x_i에서 x'_i로 천이할 수 있도록 입력 스트링을 만든다. 가정에 의해서 $s(x_i, \alpha_i) = x'_i$가 되는 다음과 같은 입력 스트링 α_i이 존재한다.

$$\alpha_i := v_i^1 v_i^2 \cdots v_i^{m(i)} \in A^+ \tag{17}$$

위 식에서 $m(i)$는 스트링 α_i의 길이이다. 이 스트링

이 들어오면 Σ는 x_i에서 시작해서 x'_i까지 $m(i)$번의 상태 천이 과정을 거친다. Σ이 거치는 중간 상태들을 아래와 같이 정의하자.

$$\begin{aligned} x_i^1 &:= s(x_i, v_i^1), \\ x_i^2 &:= s(x_i^1, v_i^2), \\ &\vdots \\ x_i^{m(i)-1} &:= s(x_i^{m(i)-2}, v_i^{m(i)-1}), \\ x'_i &:= s(x_i^{m(i)-1}, v_i^{m(i)}) \end{aligned} \tag{18}$$

Σ가 이 상태들을 거치게 하기 위해서 제어기 C에 $m(i)$개의 새로운 상태를 정의하고 아래와 같이 표기하자.

$$\xi^k(x_i, x'_i, U_i), \qquad k = 1, \ldots, m(i) \tag{19}$$

$\xi^k(x_i, x'_i, U_i) \in \Xi$는 외부 입력으로 U_i의 원소가 들어올 때 Σ의 상태를 x_i에서 x'_i으로 천이시키는 데 사용되는 제어기의 과도(transient) 상태라는 뜻이다. C가 $\xi_0(x)$에 있을 때 U_i의 원소가 외부 입력으로 들어와서 모델 Σ'의 상태가 x'_i로 바뀐다면 제어기는 먼저 $\xi^1(x_i, x'_i, U_i)$으로 천이한다. 따라서 ϕ는 다음과 같다.

$$\begin{aligned} &\phi(\xi_0(x), x'_i, (x_i, v)) := \xi^1(x_i, x'_i, U_i), \quad \forall v \in U_i, \\ &\phi(\xi_0(x), x_r, (x_i, v)) := \xi_0(x), \quad \forall v \in U(x_i)/U_i, \\ &\phi(\xi_0(x), x_r, (x_i, v)) := \xi_0, \quad \forall v \in A/(U(x_i) \cup U_i) \end{aligned} \tag{20}$$

위 식의 첫 번째 줄에서 모델의 상태 변수 x_r의 값은 $x_r = x'_i$이 된다. (즉 모델의 다음 안정 상태가 결정된다.) 위 식의 두 번째 줄은 외부 입력이 U_i에 속하지는 않지만 x_i와 안정 조합을 이루는 것이라면 제어기는 $\xi_0(x)$에 계속 머무른다는 의미이며, 세 번째 줄은 외부 입력이 $U_i(x)$와 U_i 어디에도 속하지 않는다면 제어기는 다시 초기 상태로 돌아와야 한다는 의미이다. $\xi^1(x_i, x'_i, U_i)$에서 제어기의 출력 함수 η는 다음과 같이 설정된다.

$$\eta(\xi^1(x_i, x'_i, U_i), x'_i, (x_i, v)) := v_i^1, \quad \forall v \in A \tag{21}$$

상태 $\xi_0(x)$에서 U_i에 속하는 외부 입력을 받고 모델 Σ'의 다음 상태를 x'_i으로 관측한 제어기는 $\xi^1(x_i, x'_i, U_i)$로 천이한 후 $x_i \rightarrow x'_i$ 경로를 만들어주는 첫 번째 입력 v_i^1을 Σ에 준다. Σ는 v_i^1을 받으면 첫 번째 과도 상태 x_i^1으로 천이되며(식 (18) 참조), 이 천이는 다시 제어기 C의 상태를 $\xi^1(x_i, x'_i, U_i)$ 에서 $\xi^2(x_i, x'_i, U_i)$ 로 바꾸어준다. 이러한 C와 Σ의 연쇄적인 반응 형태를 공식화하면 아래와 같다.

$$\begin{aligned} &\phi(\xi^k(x_i, x'_i, U_i), x'_i, (x_i^k, v)) := \xi^{k+1}(x_i, x'_i, U_i), \\ &\quad \forall v \in U_i, \\ &\eta(\xi^k(x_i, x'_i, U_i), x'_i, (x, v)) := v_i^k, \\ &\quad \forall (x, v) \in X \times A,\ k = 1, 2, \ldots, m(i) - 1 \end{aligned} \tag{22}$$

위의 설정은 C와 Σ가 모두 기본 모드를 만족하면서 동작하게끔 만든다.

마지막으로 제어기가 상태 $\xi^{m(i)}(x_i, x'_i, U_i)$에 다다르면 제어 입력으로 $v_i^{m(i)}$를 출력하고 Σ는 목적 상태인 x'_i에 도달하게 된다. 다음과 같이 제어기의 동작을 설정하자.

$$\begin{aligned} &\phi(\xi^{m(i)}(x_i, x'_i, U_i), x'_i, (x'_i, v)) := \xi^{m(i)}(x_i, x'_i, U_i), \\ &\quad \forall v \in U_i, \\ &\phi(\xi^{m(i)}(x_i, x'_i, U_i), x'_i, (x, v)) := \xi_0(x), \\ &\quad \forall (x, v) \in x'_i \times U(x_i), \\ &\phi(\xi^{m(i)}(x_i, x'_i, U_i), x'_i, (x, v)) := \xi_0, \\ &\quad \forall (x, v) \in X \times A / x'_i \times U(x_i), \\ &\eta(\xi^{m(i)}(x_i, x'_i, U_i), x'_i, (x, v)) := v_i^{m(i)}, \\ &\quad \forall (x, v) \in X \times A. \end{aligned} \tag{23}$$

위 설정에 따라서 외부 입력 v가 U_i에 속하는 한 Σ는 안정 조합 $(x'_i, v_i^{m(i)})$에 계속 머무른다. $v \notin U_i$인 외부 입력이 들어오는 순간 제어기는 최종 상태 $\xi^{m(i)}(x_i, x'_i, U_i)$를 벗어나서 ξ_0 또는 $\xi_0(x)$로 천이하여 또 다른 제어 동작을 준비한다.

이와 같은 제어기 설계 과정을 모든 $x'_1, \ldots, x'_q$에 대해서 적용하면 폐루프 시스템은 안정 상태 x_i에서 모델의 다음 상태가 $x'_1, \ldots, x'_q$ 중 임의의 하나로 천이되는 동작을 정확하게 추종할 수 있게 된다. 이러한 과정을 거치는 폐루프 시스템 Σ_{cls}의 recursion 함수 s'는 조건 (i)과 (ii)를 만족시킨다고 말할 수 있다. ((i)에 해당되지 않는 유효 조합을 그대로 두면 조건 (ii)가 자동적으로 만족된다.) 따라서 교정 제어기 C의 존재가 증명되었다.

□

정리 2의 증명 과정은 단일 상태 x_i에서 모델 Σ'가 가질 수 있는 비결정적 동작을 머신 Σ가 추종하도록 하는 제어기를 설계하였다. 정리 2의 가정에도 나와 있듯이 이러한 제어기가 존재할 필요조건은 머신 Σ가 상

태 x_i에서 $s'(x_i \times U_i)$에 속하는 임의의 상태로 도달 가능해야 한다는 사실이다. 이 조건을 정의 3의 skeleton 행렬로 표시하면 아래와 같다.

$$K_{ij(k)}(\Sigma) = 1,\ k = 1,\ldots,q \tag{24}$$

위 식에서 $j(1), j(2), \cdots j(q)$는 $s'(x_i \times U_i)$에 속하는 상태 $x'_1,\ldots,x'_q$가 집합 X에서 차지하는 색인(index)을 말한다.

정리 2를 일반화하여 본 논문의 목적인 임의의 비결정 모델에 대한 교정 제어기의 존재 조건과 설계 과정을 얻는다. 본 논문의 주요 연구 결과는 아래 정리에 나와 있다.

정리 3. 입력/상태 비동기 순차 머신 $\Sigma = (A,X,X,x_0,f,h)$과 stable-state 머신 모델 $\Sigma' = (A,X,X,x_0,s',h')$이 있다. Σ'는 식 (9), (10)과 같이 정의된 비결정 모델이다. Σ의 폐루프 시스템 Σ_{cls}이 Σ'와 stably equivalent하는 동작을 보이도록 만드는 교정 제어기 C가 존재할 필요충분조건은 $K(\Sigma) \ge K(\Sigma')$이다. $K(\Sigma)$와 $K(\Sigma')$는 각각 Σ와 Σ'의 skeleton 행렬이다.

증명: 먼저 $K(\Sigma) \ge K(\Sigma')$라고 가정하자. Σ와 Σ'와의 모델 차이를 $x_1 \times U_1, \ldots x_r \times U_r$로 표현하자. $x_1 \times U_1, \ldots x_r \times U_r$는 서로소이다. 폐루프 시스템 Σ_{cls}이 $x_1,\ldots,x_r$ 중의 한 상태 x_i와 안정 조합을 이룰 때 U_i에 속한 입력 v가 들어오면 비결정 모델 Σ'의 다음 안정 상태 $s'(x_i,v)$와 동일한 상태로 천이해야 한다. $s'(x_i,v) = \{x'_{j(1)},\ldots,x'_{j(q)}\}$라고 하면 가정에서 $K(\Sigma) \ge K(\Sigma')$이므로 다음과 같은 관계가 성립된다.

$$K_{ij(k)}(\Sigma) \ge K_{ij(k)}(\Sigma'),\ k = 1,\ldots,q \tag{25}$$

즉 비결정 모델 Σ'의 상태 x_i에서 $x'_i(i=1,\ldots,q)$로 이동하는 입력 스트링이 존재한다면 비동기 순차 머신 Σ에서도 역시 그러한 스트링이 존재한다. 이것은 정리 2의 가정과 동일하므로 x_i에서 폐루프 시스템 Σ_{cls}의 동작을 Σ'와 일치시키는 교정 제어기가 존재한다. $x_1,\ldots,x_r$에 속하는 다른 상태에 대해서도 비슷하게 제어기를 구성할 수 있으며 이러한 제어기들을 $F(x_i)$, $i=1,\ldots,r$이라고 정의하자. $x_1 \times U_1, \ldots x_r \times U_r$는 서로소이므로 각 제어기들의 최초 상태 ξ_0와 transition 상태 (정리 2 증명 참조) $\xi_0(x)$를 각각 결합하여 하나의 제어기 $C = F(x_1) \vee \cdots \vee F(x_r)$를 만든다. 기존 연구[4]에서 증명된 것처럼 결합 제어기 C는 기본 모드를 만족하면서 동작하므로 모델 매칭을 완성시킨다.

한편 $\Sigma_{cls} = \Sigma'$을 만족시키는 교정 제어기 C가 존재할 때 skeleton 행렬 부등식 $K(\Sigma) \ge K(\Sigma')$이 성립된다는 명제는 기존 연구[3~4]의 결과와 유사한 방법으로 증명된다. 지면 관계상 증명 과정은 생략한다. ([3]의 Theorem 4.1 및 [4]의 Theorem 4 참조) □

한편 정의 2의 모델 매칭 문제는 결정적 모델에 대한 교정 제어기를 설계하는 문제이므로 정리 2와 정리 3의 증명 과정에서 제시한 제어기 설계 과정과 동일하게 풀린다. 또한 skeleton 행렬 부등식도 정리 3과 거의 유사한 형태로 표현된다. 정의 2에 대한 제어기 존재 조건을 증명 없이 다음과 같이 기술한다.

정리 4. 입력/상태 비동기 순차 머신 $\Sigma = (A,X,X,x_0,f,h)$과 stable-state 머신 모델 $\Sigma' = (A,X,X,x_0,s',h')$이 있다. Σ'는 식 (9), (10)과 같이 정의된 비결정 모델이다. 다음 두 명제는 동치(equivalent)이다.

(i) Σ의 폐루프 시스템 Σ_{cls}이 Σ'가 가질 수 있는 DFSM 중 적어도 한 개와 stably equivalent하는 동작을 보이도록 만드는 교정 제어기 C가 존재한다.

(ii) $K(\Sigma) \ge K(\Sigma_i')$인 DFSM Σ_i'가 $\Sigma_1',\ldots,\Sigma_p'$ 중 적어도 한 개 존재한다. $\Sigma_1',\ldots,\Sigma_p'$은 $\Sigma' = \Sigma_1' \parallel \Sigma_2' \parallel \cdots \parallel \Sigma_p'$을 만족시킨다. □

V. 예 제

사례 연구를 통해서 본 논문에서 제안한 교정 제어기의 성능을 검증하기로 한다. 그림 2는 비결정 동작을 가진 모델 Σ'의 상태흐름도(State flow diagram)이다. 앞 논문[5]에서 Σ'와 DFSM 부-모델 Σ'_1, Σ'_2의 skeleton 행렬은 아래와 같이 유도되었다.

$$K(\Sigma'_1) = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 1 & 1 \end{bmatrix} \tag{26}$$

$$K(\Sigma'_2) = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 1 & 1 \end{bmatrix} \tag{27}$$

$$K(\Sigma') = \begin{bmatrix} 1\,1\,1 \\ 0\,1\,1 \\ 0\,1\,1 \end{bmatrix} \tag{28}$$

한편 제어 대상 비동기 순차 머신 Σ이 그림 3과 같다고 하자. Σ은 모델 Σ'과 x_1에서 그 동작이 다르다. 비결정 모델 Σ'은 안정 상태 x_1에서 외부 입력 a가 들어오면 x_2(부 모델 Σ'_1) 또는 x_3(부 모델 Σ'_2)으로 천이되지만, Σ은 안정 조합 (x_1,a)을 유지한다. 따라서 교정 제어기 C의 역할은 x_1에서 폐루프 시스템 Σ_{cls}이 모델 Σ'의 다음 안정 상태와 동일하게 천이되거나 (정의 1의 문제) 부-모델 Σ'_1 또는 Σ'_2의 동작과 일치되게 움직이도록 하는 일이다(정의 2의 문제). Σ의 skeleton 행렬도

$$K(\Sigma) = \begin{bmatrix} 1\,1\,1 \\ 0\,1\,1 \\ 0\,1\,1 \end{bmatrix} \tag{29}$$

와 같이 구해지므로 정리 3과 정리 4에 의해서 그러한 교정 제어기가 존재함을 알 수 있다.

정리 2의 증명 과정에서 제시한 방법으로 C를 설계한다. 먼저 초기 상태 ξ_0에서 C의 동작은 아래와 같이 정의된다.

$$\begin{aligned} &\phi(\xi_0,x_r,(x,v)) := \xi_0,\ \forall\ (x,v)\in X\times A/x_1\times a,b \\ &\phi(\xi_0,x_r,(x_1,v)) := \xi_0(x),\ v\in a,b \end{aligned} \tag{30}$$

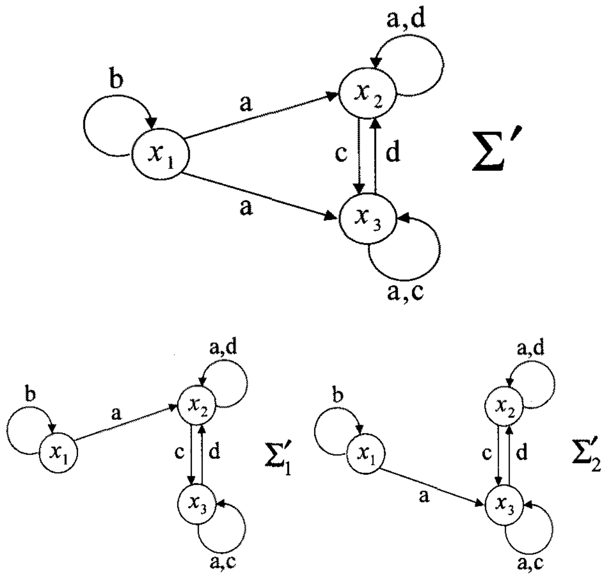


그림 2. 비결정 모델 Σ'

Fig. 2. nondeterministic model Σ'.

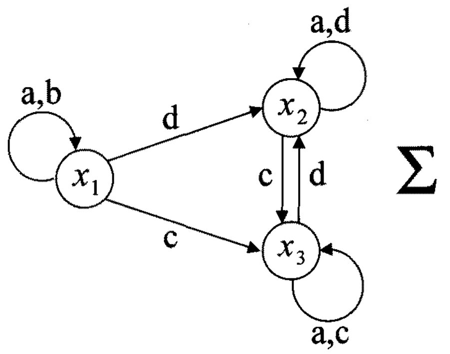


그림 3. 비동기 순차 머신 Σ

Fig. 3. Asynchronous sequential machine Σ.

그림 3에서 $U(x_1)$, 즉 x_1과 안정 조합을 이루는 입력 알파벳은 $\{a,b\}$임을 쉽게 알 수 있다. transition 상태 $\xi_0(x)$에서 C는 외부 입력 a가 들어오면 모델 Σ'의 상태를 관측한 후 입력 스트링을 만들어낸다. 만약 모델 Σ'의 다음 상태가 x_2이면 그림 3에서 볼 수 있듯이 입력 스트링 d(길이가 1인 스트링)를 생성하면 되고 모델 Σ'의 다음 상태가 x_3이면 입력 스트링 c를 생성한다. 식 (11)에서 $m(i)=1$이므로 C의 과도 상태는 $\xi^1(x_1,x_2,a)$와 $\xi^1(x_1,x_3,a)$ 두 개만 만들면 충분하다. 모델 Σ'의 다음 안정 상태가 x_2라고 가정하고 C의 동작을 규정해본다. 식 (20)과 (23)에 따라서 C의 상태 천이 함수와 출력 함수를 설정하면 아래와 같다. 참고로 $U_1=\{a\}$이다.

$$\begin{aligned} &\phi(\xi_0(x),x_2,(x_1,a)) := \xi^1(x_1,x_2,a) \\ &\phi(\xi_0(x),x_r,(x_1,b)) := \xi_0(x), \\ &\phi(\xi_0(x),x_r,(x_1,v)) := \xi_0,\quad v\in c,d \end{aligned} \tag{31}$$

$\eta(\xi_0(x),x_r,(x_1,v)) := b,\ \forall\, v\in A$ (Σ가 x_1과 안정 조합을 유지하도록 $b\in U(x_1)$인 b를 넣어줌)

$$\begin{aligned} &\phi(\xi^1(x_1,x_2,a),x_2,(x_2,a)) := \xi^1(x_1,x_2,a), \\ &\phi(\xi^1(x_1,x_2,a),x_2,(x_2,d)) := \xi_0(x),(d\in U(x_2)) \\ &\phi(\xi^1(x_1,x_2,a),x_2,(x,v)) := \xi_0, \\ &\qquad (x,v)\in X\times A/x_1\times a,d \end{aligned} \tag{32}$$

$$\eta(\xi^1(x_1,x_2,a),x_r,(x,v)) := d,\ \forall\ (x,v)\in X\times A. \tag{33}$$

(제어기 C의 출력을 d로 함으로써 머신 Σ가 x_1에서 x_2로 상태 천이되도록 교정함)

모델 Σ'의 다음 안정 상태가 x_3일 때에도 위와 유

사하게 제어기를 설계하여 두 개의 제어기를 결합하면 최종 제어기가 구성된다. 또 폐루프 시스템 $\Sigma_{c/s}$이 부-모델 Σ'_1 또는 Σ'_2와 stably equivalent되게 하는 문제 (정의 2)를 위한 교정 제어기는 이 두 제어기 중의 하나를 사용하면 된다.

VI. 결 론

Hammer 등이 제안한 비동기 순차 머신을 위한 교정 제어 시스템은 비동기 순차 머신의 동작을 재설계를 거치지 않고 바꿀 수 있다는 점에서 새로운 연구 테마를 제시했다 하겠다. 본 논문에서는 교정 제어 및 모델 매칭의 개념을 소개하였고 특히 제어 대상 비동기 머신이 추종해야 하는 모델의 동작이 비결정적일 경우의 문제를 해결하였다. 기존 연구와 유사하게 제어기의 존재조건은 skeleton 행렬의 부등식으로 표현된다.

제어기의 필요충분조건이 만족될 경우 비결정적 모델과의 매칭을 위한 제어기 설계 과정을 제시하고 예제에서 교정 제어기의 검증을 하였다. 논문에서 제시된 제어기는 기존 연구에서 제안된 제어기를 일반화한 것으로 기본 모드를 만족시키면서 폐루프 시스템이 비결정적 모델의 행동과 일치하도록 교정 동작을 실시한다.

제안된 교정 제어기의 상태 개수를 줄이는 일과 교정 제어 시스템을 현존하는 비동기 순차 머신에 적용하는 연구 등이 차기 과제로 남아 있다.

참 고 문 헌

[1] M. Singh and S. M. Nowick, "Synthesis-for-initializability of asynchronous sequential machines," *in Proceedings of International Test Conference*, pp. 232-241, 1996.

[2] S. W. Moore, G. S. Taylor, P. A. Cunningham, R. D. Mullins and P. Robinson, "Self calibrating clocks for globally asynchronous locally synchronous systems," *in Proceedings of IEEE International Conference on Computer Design*, pp. 73-78, 2000.

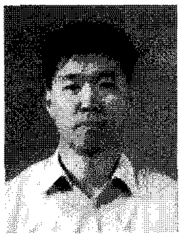
[3] T. E. Murphy, X. Geng and J. Hammer, "On the control of asynchronous machines with races," IEEE Transactions on Automatic Control, vol. 48, no. 6, pp. 1073-1081, 2003.

[4] N. Venkatraman and J. Hammer, "On the control of asynchronous sequential machines with infinite cycles," *International Journal of Control*, vol. 79, no. 7, pp. 764-785, 2006.

[5] 양정민, "비결정 모델에 대한 비동기 순차 회로의 교정 제어 I: 도달가능성 분석," *대한전자공학회 논문지* 제45권 SC 제7호.

[6] Z. Kohavi, *Switching and Finite Automata Theory*, McGraw-Hill, New York, 1970.

저 자 소 개



양 정 민(정회원)
1993년 한국과학기술원 전기 및 전자공학과 학사 졸업
1995년 한국과학기술원 전기 및 전자공학과 석사 졸업
1999년 한국과학기술원 전기 및 전자공학과 박사 졸업
1999년～2001년 한국전자통신연구원 선임연구원
2001년～현재 대구가톨릭대학교 전자공학과 부교수
<주관심분야 : 비동기 머신 제어, 보행 로봇 시스템 등>