2013년 5월 전자공학회 논문지 제 50 권 제 5 호
Journal of The Institute of Electronics Engineers of Korea Vol. 50, NO. 5, May 2013 http://dx.doi.org/10.5573/ieek.2013.50.5.253

논문 2012-50-5-29

직렬 결합된 복합 비동기 순차 머신을 위한 모델 정합

(Model Matching for Composite Asynchronous Sequential Machines in Cascade Connection)

양 정 민*

(Jung-Min Yang©)

요 약

본 논문에서는 복합 비동기 순차 머신을 제어하는 방법을 다룬다. 논문에서 고려하는 비동기 머신은 입력/상태 머신 두 개가 직렬 결합된 것으로서 선행 머신의 출력 값이 후행 머신의 입력으로 전달된다. 제어 목적은 폐루프 시스템의 입력과 출력의 정상 상태 특성을 원하는 모델의 동작과 일치시키는 모델 정합 문제를 구현하는 교정 제어기를 꾸미는 일이다. 교정 제어기는 후행 머신의 상태 피드백 정보만 전달 받기 때문에 선행 머신의 상태를 정확하게 알 수 없는 불확실성이 존재한다. 본 논문에서는 그러한 불확실성 하에서 모델 정합 교정 제어기가 존재할 조건을 구하고 설계 알고리듬을 제안한다. 또한 사례 연구를 통해서 제안된 제어기의 설계 과정을 예시한다.

Abstract

In this paper, we study the problem of controlling composite asynchronous sequential machines. The considered asynchronous machine consists of two input/state machines in cascade connection, where the output of the front machine is delivered to the input channel of the rear machine. The objective is to design a corrective controller realizing model matching such that the stable state behavior of the closed-loop system matches that of a reference model. Since the controller receives the state feedback of the rear machine only, there exists uncertainty about the present state of the front machine. We specify the existence condition for a corrective controller given the uncertainty. The design procedure for the proposed controller is described in a case study.

Keywords : Asynchronous Sequential Machines, Corrective Control, Model Matching, Cascade Connection

Ⅰ. 서 론

비동기 순차 머신(asynchronous sequential machine)은 전역 동기화 클럭(global synchronizing clock) 없이 입력의 변화만으로 상태 변수가 바뀌는 시스템을 말한다. 비동기 순차 머신은 주로 디지털 시스템 하드웨어의 중요한 요소로서 지금까지 많이 사용되어 왔다. 특히 1980년대 초반부터 Program Transformation 기법[1], State Transition Graph (STG)[2] 등 새로운 설계 방법들이 제안되면서 비동기 머신에 대한 연구가 다시 활발하게 진행되었다. 비동기 마이크로프로세서 등 시스템 전체를 비동기적으로 운용하거나[3] GALS (globally asynchronous, locally synchronous)[4] 개념을 도입하여 국부적으로 동기화 시스템을 운용하고 전역적으로 비동기를 사용하는 기법도 개발되었다. 하지만 비동기 머신은 동기 머신에 비해서 설계과정이 복잡하고 한 번 설계한 비동기 머신의 동작을 재설계 없이 고치기가 어렵다는 평가를 받는다.

* 정회원, 대구가톨대학교 전자공학과
(Department of Electrical Engineering, Catholic University of Daegu)
※ 이 논문은 2012년도 대구가톨릭대학교 교내연구비 지원에 의한 것임.
© Corresponding Author(E-mail:jmyang@cu.ac.kr)
접수일자: 2012년11월19일, 수정완료일: 2013년4월17일

본 논문에서는 비동기 순차 머신을 제어하는 문제를 다룬다. 자동 제어의 원리를 적용하여 이미 설계된 비동기 머신의 정상 상태(stable state) 동작을 바꾸는 기법을 교정 제어(corrective control)[5,6]라 부른다. 교정 제어는 비동기 머신의 과도 상태 천이(transient state transitions) 속도가 매우 빠르다는 성질을 이용한다. 즉 외부 입력과 출력 피드백을 받은 제어기가 목적에 맞게 일련의 제어 입력을 머신에 인가하면 머신의 고유한 입력/출력 특성을 유지하면서 폐루프 시스템의 정상 상태 동작이 보정된다. 교정 제어는 과도 고장[7], 영구 고장[8], 간헐 고장[9] 등 비동기 머신에서 발생하는 여러 가지 고장에 의한 오동작을 없애는 데 성공적으로 적용되었다.

본 논문의 주요 목적은 복합 비동기 머신에 대한 모델 정합(model matching)을 구현하는 교정 제어기를 설계하는 일이다. 모델 정합이란 폐루프 시스템의 동작을 원하는 모델의 입력/출력 특성과 일치시키도록 머신을 제어하는 것을 말한다[10]. 이번 연구에서는 특히 제어 대상 머신이 두 개의 독립 비동기 머신이 직렬 결합된 복합 시스템인 경우를 다룬다. 선행 머신(front machine)의 출력 값이 후행 머신(rear machine)의 입력으로 전달되는 직렬 결합은 모듈화된 디지털 시스템에서 흔히 관측된다[11]. 따라서 직렬 결합된 복합 비동기 순차 머신을 위한 모델 정합 교정 제어기가 존재할 조건과 설계 알고리듬을 제안하는 일은 의의가 있다고 말할 수 있다.

고장 극복을 제어 목적으로 한 기존 연구들은[7~9] 모두 단일 비동기 머신을 기반으로 한 것들이므로 복합 비동기 머신 제어에 응용되지 못한다. 본 연구는 직렬 결합 비동기 머신을 대상 시스템으로 한 교정 제어 이론에 대한 새로운 접근이다. 한편 저자의 선행 연구[12]에서는 직렬 결합 비동기 머신에서 발생하는 과도 고장을 극복하는 제어기를 설계하였다.

본 논문에서는 먼저 직렬 결합된 비동기 순차 머신을 모델링하고 모델 정합 문제를 설정한다. 그런 다음 선행 머신의 상태 불확실성 문제를 분석한다. 교정 제어기는 후행 머신의 출력 피드백만을 받으므로 선행 머신의 현재 상태를 정확하게 알지 못한 채 모델 정합 제어를 수행해야 한다. 이러한 내재적 상태 불확실성 조건 하에서 교정 제어기가 존재할 필요충분조건을 규명한다. 또한 사례 연구를 통해서 제안된 제어기의 설계 과정을 예시한다.

Ⅱ. 본 론

1. 직렬 결합 비동기 순차 머신

본 논문에서 다루는 비동기 순차 머신은 머신의 현재 상태 값이 출력으로 나오는 입력/상태 머신이다. 유한 상태 머신(finite-state machine)으로 입력/상태 비동기 머신 Σ_1을 표현하면 다음과 같다.

$$\Sigma_1 = (A,\ X,\ x_0,\ f)$$

A는 입력 집합, X는 상태 집합, $x_0 \in X$는 초기 상태(initial state)이며, $f:X\times A \rightarrow X$는 상태 천이 함수(state transition function)이다. 비동기 머신의 상태는 항상 안정 상태(stable state)나 과도 상태(transient state) 중의 하나이다[13]. 임의의 상태-입력 조합 $(x,u)\in X\times A$에서 $f(x,u)=x$라면 x는 안정 상태이며 (x,u)는 안정 조합(stable combination)이다. 반면 $f(x,u)\neq x$이면 x는 과도 상태, (x,u)는 과도 조합(transient combination)이다. Σ가 과도 조합 (x,u)에 진입하면 $f(x,u)=x^1$, $x^2=f(x^1,u),\ldots$ 등으로 과도 상태 $x^1,x^2,\ldots$를 순식간에 거쳐서 $f(x',u)=x'$인 '다음 안정 상태(next stable state)' x'에 도달한다. Σ가 과도 상태에 머무르는 시간이 극히 짧기 때문에(이론적으로는 0) 외부 사용자는 Σ가 안정 상태 x에서 다음 안정 상태 x'로 천이하는 운동만 관측하게 된다. 이러한 Σ의 안정 상태 동작만을 따로 표현하기 위해서 'stable recursion 함수' $s_1:X\times A\rightarrow X$를 다음과 같이 정의하자.

$$s_1(x,u) := x'$$

위 식에서 x'는 (x,u)의 다음 안정 상태이다. 단위 입력 대신 입력 스트링을 s_1의 변수로 설정하면 다음과 같이 일반화할 수 있다.

$$s_1(x,ut) := s_1(s_1(x,u),t),\ x\in X,\ u\in A,\ t\in A^+$$

A^+는 A의 원소로 이루어진 길이 1 이상의 스트링 집합을 의미한다. $s_1(x,t)=x'$인 입력 스트링 $t\in A^+$가 존재하면 상태 x'는 상태 x로부터 '안정적으로 도달가능하다(stably reachable)'[6]라고 부른다.

Σ_1과 동일하게 또 하나의 입력/상태 비동기 머신 Σ_2를 다음과 같이 정의한다.

$$\Sigma_2 = (X,\ Y,\ y_0,\ g)$$

X, Y, y_0은 각각 Σ_2의 입력 집합, 상태 집합, 초기 상태를 가리키며, $g:Y\times X\rightarrow Y$는 상태 천이 함수이다. 또

s_2:Y×X→Y를 Σ_2의 stable recursion 함수로 정의하고 위에서 정의한 s_1과 동일한 특성을 부여하자.

Σ_1과 Σ_2가 직렬 결합된 복합 비동기 머신을 Σ라고 정의한다. 선행 머신 Σ_1의 출력 x가 후행 머신 Σ_2의 입력으로 들어간다(이것이 Σ_2의 입력 집합을 X로 정의한 이유이다). [11] 등에서 나온 결합 공식을 이용하면 Σ는 다음과 같은 입력/출력 비동기 순차 머신으로 유도된다.

$\Sigma = (A, Y, X\times Y, (x_0,y_0), \delta, h)$

A와 Y는 입력 및 출력 집합이고 X×Y는 상태 집합, (x_0,y_0)은 초기 상태이다. 또 δ와 h는 다음과 같은 매핑(mapping)을 가지는 Σ의 상태 천이 및 출력 함수이다.

$$\delta: X\times Y\times A \rightarrow X\times Y$$

$$h: X\times Y\times A \rightarrow Y$$

δ의 특성을 기술하기 전에 본 논문에서는 모든 비동기 순차 머신들이 기본 모드 원리(the principle of fundamental mode operation)[13]를 만족시킨다고 가정한다. 기본 모드 원리는 비동기 머신이 안정 상태에 있을 때에만 입출력 변수가 바뀔 수 있다는 것이다. 기본 모드가 준수되지 않는다면 머신이 과도 상태에 있을 때에도 입력 값이 바뀔 수 있다. 이 경우 클럭에 의한 동기화가 안 되어 있기 때문에 비동기 머신이 가지는 다음 안정 상태는 예측 불가능하게 된다.

Σ_1과 Σ_2가 각각 안정 상태 x_k와 y_k에 있을 때 $f(x_k,u_k)\neq x_k$인 입력 $u_k\in A$가 Σ_1에 들어온다고 가정하자. 앞서 기술한 바대로 Σ_1은 $x^1=f(x_k,u_k)$, $x^2=f(x^1,u_k),\ldots$, $x_{k+1}=f(x_{k+1},u_k)$등으로 일련의 과도 상태 천이를 거친다. $x_{k+1}=s_1(x_k,u_k)$, 즉 x_{k+1}는 다음 안정 상태이다. Σ_1의 출력은 $x^1, x^2,\ldots, x_{k+1}$로 계속 변화하며 이 값이 Σ_2의 입력으로 들어간다. 하지만 기본 모드 원리에 따라서 Σ_2는 다음 안정 상태 x_{k+1}이외의 입력에는 반응하면 안 된다. 만약 Σ_2가 Σ_1의 과도 상태 입력에 반응한다면 Σ_1과 Σ_2가 동시에 과도 조합에 있게 되므로 그 후의 동작은 예측불가능하게 된다. 본 논문에서는 Σ_1과 Σ_2 사이에 이러한 경우가 존재하지 않는다고 설정한다.

복합 비동기 머신 Σ의 stable recursion 함수를 s:X×Y×A → X×Y라고 하면 위에서 기술한 가정으로부터 s는 아래와 같이 유도된다.

$$s((x_k,y_k),u_k) = (s_1(x_k,u_k),\ s_2(y_k,s_1(x_k,u_k)))$$

입력 u_k가 들어오면 선행 머신 Σ_1이 다음 안정 상태 $s_1(x_k,u_k)$로 천이하고 후행 머신 Σ_2는 $s_1(x_k,u_k)$를 입력으로 받아서 다음 안정 상태 $s_2(y_k,s_1(x_k,u_k))$로 천이한다. s_1과 s_2와 마찬가지로 s도 A^+에 속하는 입력 스트링에 대해서 확장 가능하다.

Σ의 출력 함수 h는 Σ가 임의의 상태-입력 조합 $((x_k,y_k),u_k)$에 있을 때 다음과 같이 항상 y_k를 출력 값으로 낸다.

$$h((x_k,y_k),u_k) = y_k \quad \forall((x_k,y_k),u_k)\in X\times Y\times A$$

앞에서 단일 비동기 머신 Σ_1에서 $x'=s_1(x,t)$인 입력 스트링 t∈A+가 존재하면 x′는 x로부터 ‘안정적으로 도달 가능하다’라고 정의하였다. 복합 머신 Σ에서도 Σ를 이차원 상태 (x,y)에서 (x′,y′)로 천이하게 하는 입력 스트링 t가 존재하면 (x′,y′)은 (x,y)로부터 ‘안정적으로 도달 가능하다’라고 부른다. 이 경우 t는 선행 머신 Σ_1를 x에서 x′으로 이동시키며, 그 동작에서 수반되는 Σ_1의 출력 스트링이 후행 머신 Σ_2를 y에서 y′로 옮긴다. $t=u_1u_2\cdots u_k$라 하면 다음과 같은 관계가 성립한다.

$$s_1(x,t)=x',\ s_2(y,z)=y',\ z:=x^1\cdots x^{k-1}x'$$

$$x^1=s_1(x,u_1),\ x^2=s_1(x^1,u_2),\ldots,\ x'=s_1(x^{k-1},u_k)$$

2. 교정 제어 시스템

그림 1은 직렬 결합 비동기 머신 Σ의 모델 정합을 위한 교정 제어 시스템이다. C는 비동기 머신으로 구현되는 교정 제어기이다. v∈A는 외부 입력이며, u∈A는 C가 만드는 제어 입력, 그리고 x∈X, y∈Y는 각각 Σ_1과 Σ_2의 현재 상태 값이다. Σ_c는 Σ와 C로 구성된 폐루프 시스템(closed-loop system)을 가리킨다.

문제의 목적은 폐루프 시스템 Σ_c의 안정 상태(stable state) 동작이 주어진 모델(reference model) Σ′과 일치하도록 하는 교정 제어기 C를 설계하는 일이다. 그림 1로부터 모델 Σ′은 입력 집합이 A, 출력 집합이 Y인 입

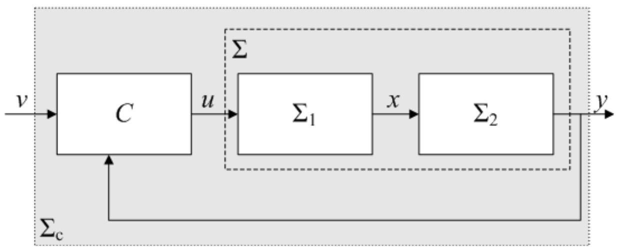

그림 1. 직렬 결합 비동기 순차 머신을 위한 교정 제어 시스템
Fig. 1. Corrective control system for an asynchronous sequential machine in cascade connection.

력/상태 머신이어야 모델 정합 문제가 성립한다는 사실을 알 수 있다. 즉 Σ'를

$$\Sigma' = (A,\ Y,\ y_0,\ s')$$

로 표시할 수 있다. $s':Y\times A\rightarrow Y$는 Σ'의 stable recursion 함수이다. 비동기 순차 머신 사이의 모델 정합 문제는 각 머신의 안정 상태 동작만 고려하므로 위와 같이 모델의 안정 상태 동작 s'만 설정해도 일반성을 잃지 않는다[10].

이번 연구에서는 그림 1에 나타난 바대로 제어기 C가 후행 머신 Σ_2의 상태 변수만 피드백 받는다고 가정한다. 다시 말하면 C는 선행 머신 Σ_1의 현재 상태 x를 알지 못하며, 외부 입력 v와 Σ_2의 상태 피드백 y만으로 x의 범위를 추정해야 한다. 이러한 내재적인 불확실성이 있을 때 주어진 모델 정합 문제를 해결할 수 있는 교정 제어기 C의 존재 조건을 찾는 일이 본 논문의 핵심 주제이다.

Σ가 안정 상태에 있을 때 C로 들어오는 외부 입력이 v, Σ_2의 상태 피드백이 y라고 하자. 기본 모드 원리에 의해서 Σ가 안정 상태에 있다는 것은 Σ_1과 Σ_2가 모두 안정 상태에 있다는 사실을 의미한다. 주어진 현재 입력 (v,y)로 C가 추정할 수 있는 Σ_1의 현재 상태 집합을 $W(v,y)\subset X$라 하면 $W(v,y)$는 아래와 같이 얻어진다.

$$W(v,y) = \{x\in X|\ s_1(x,v)=x,\ s_2(y,x)=y\} \tag{1}$$

$|W(v,y)|>1$이면 C는 Σ_1의 현재 상태 x를 정확하게 알지 못하며 $W(v,y)$의 한 원소라는 사실만 인지한다.

후행 머신 Σ_2의 현재 안정 상태 y만으로도 선행 머신 Σ_1의 상태의 범위를 추정할 수 있다. Σ_2에서 y와 안정 조합을 이룰 수 있는 모든 입력 집합을 $X(y)\subset X$라 정의하면

$$X(y) = \{x\in X|\ s_2(y,x)=y\} \tag{2}$$

이다. Σ_2가 안정 상태 y에 있다면 Σ_1은 $X(y)$에 속한 상태 중 하나에 머무른다. (1)과 (2)로부터 $W(v,y)\subseteq X(y)$라는 관계가 성립한다. 즉 정보를 더 많이 알수록 Σ_1 상태의 불확실성은 줄어든다.

3. 교정 제어기 존재 조건

Σ와 Σ' 사이의 모델 부정합(model mismatch)을 다음과 같은 집합 $D\subset Y\times A\times Y$로 표현한다.

$$D = \{(y_i,a_i,y'_i)\ |\ 1\le i\le r\} \tag{3}$$

$(y_i,a_i,y'_i)\in D$의 뜻은 다음과 같다. 모델 Σ'는 안정 상태 y_i에 있을 때 입력 a_i가 들어오면 다음 안정 상태 y'_i로 천이한다. 즉 $s'(y_i,a)=y'_i$의 상태 천이를 가진다. 하지만 복합 머신 Σ는 (y_i,a_i)에서 Σ'와 다른 특성을 보인다. 식 (3)에서 $1\le i,j\le r$이고 서로 다른 i와 j에 대해서 $y_i=y_j$인 경우도 존재한다. 이때 a_i와 a_j는 서로 달라야 한다. 즉 머신 Σ가 안정 상태 y_i(또는 y_j)에 있을 때 외부 입력 a_i와 a_j가 모두 모델 부정합을 야기한다.

Σ_2의 안정 상태가 y_i일 때 Σ_1이 가질 수 있는 안정 상태 집합을 식 (2)에서 $X(y_i)$라고 정의하였다. 모델 부정합 (y_i,a_i,y'_i)의 의미를 $X(y_i)$를 이용하여 다시 표현하면 다음과 같다.

$$\exists X_d(y_i)\subseteq X(y_i) \text{ and } X_d(y_i)\neq\varnothing \text{ such that}$$

$$\forall x\in X_d(y_i)\ \ s_2(y_i,s_1(x,a_i))\neq y'_i \tag{4}$$

식 (4)는 선행 머신 Σ_1이 $X_d(y_i)$에 속하는 어떤 상태 x에 있을 때 (Σ_2는 y_i에 머물러 있음) 외부 입력 a_i가 들어오면 후행 머신 Σ_2는 y'_i로 천이하지 않는다는 것을 나타낸다($s_2(y_i,s_1(x,a_i))\neq y'_i$). 즉 복합 비동기 머신 Σ가 입력 a_i에 대응하여 가지는 다음 안정 상태에서 Σ_2의 상태 값이 y'_i이 아닌 다른 값이 된다.

(y_i,a_i,y'_i)에 대응하는 Σ_1의 상태 부분 집합 $X_d(y_i)$의 존재는 Σ_1이 $X_d(y_i)$에 속하는 어떤 안정 상태에 머무를 때 외부 입력이 a_i로 바뀌면 모델 정합을 위해서 교정 제어기 C가 작동해야 한다는 것을 시사한다. 즉 C는 입력 a_i 대신 일련의 제어 입력 값 u를 생성해야 한다. 문제는 그림 1에서 알 수 있듯이 일반적으로 C가 Σ_1의 현재 상태를 정확하게 관측하지 못한다는 점이다. 교정 제어기 C가 동작하는 과정은 집합 $X_d(y_i)$과 $X(y_i)$의 크기에 따라서 다음과 같이 나뉜다($X_d(y_i)$과 $X(y_i)$는 C에 미리 알려져 있다고 가정한다).

i) $|X(y_i)|=|X_d(y_i)|=1$

$|X(y_i)|=|X_d(y_i)|=1$은 Σ_2가 안정 상태 y_i에 있을 때 가능한 입력이 한 개, 즉 Σ_1이 가질 수 있는 상태가 한 개밖에 없다는 뜻이다. 이 경우는 Σ_2가 안정 상태 y_i에 진입한 후 외부 입력 a_i가 들어오면 모델 정합을 위해 C가 즉시 작동 가능하다. 또 Σ_1의 현재 상태를 정확히 알 수 있기 때문에 C가 정상적으로 교정 제어 동작을 취할 수 있다. 기존 연구[6,7]에서 모델 부정합 (y_i,a_i,y'_i)을 해결하기 위한 교정 제어기가 존재할 필요충분조건은 y'_i가 y_i로부터 안정적으로 도달 가능하다는 것이었

다. 직렬 결합 비동기 머신 Σ에서는 후행 머신 Σ_2가 y_i에서 y'_i로 안정적으로 도달 가능해야 하므로, 이 경로를 만들어주는 Σ_2의 입력 스트링을 Σ_1이 생성하게 하는 외부 입력 스트링이 존재해야 한다. $X_d(y_i)=\{x_i\}$라고 하고 이 조건을 나타내면 아래와 같다.

$$\exists t \in A^+ \text{ such that } s((x_i,y_i),t)=(x'_i,y'_i),\ x'_i \in X(y'_i) \quad (5)$$

$t:=u_1u_2\cdots u_k$라 하고 위 식을 자세히 설명한다. 먼저 선행 머신 Σ_1이 t를 받으면 x_i에서부터 일련의 안정 상태로 천이한다. Σ_1이 거치는 안정 상태들을 $x^1=s_1(x_i,u_1)$, $x^2=s_1(x^1,u_2)$,..., $x'_i=s_1(x^{k-1},u_k)$라고 표기하자. t가 들어갈 때 Σ_2가 Σ_1로부터 받는 입력 스트링은 앞에서 정의한 변수로부터 $z:=x^1x^{k-1}\cdots x'_i$이 된다. 따라서 식 (5)에 의해서 $s_1(x_i,t)=x'_i$이고 $s_2(y_i,z)=y'_i$가 되며 Σ_1의 다음 안정 상태 x'_i는 모델 정합을 완성하는 Σ_2의 상태 y'_i와 안정 조합을 이루어야 한다($x'_i \in X(y'_i)$).

그림 2는 입력 스트링 t를 이용한 (y_i,a_i,y'_i)에 대한 교정 제어 과정을 도시한다. Σ가 안정 상태 (x_i,y_i)에 진입하면 C는 외부 입력의 변화를 보면서 교정 제어를 준비한다. $v \neq a_i$인 입력이 들어온다면 C는 v를 변경하지 않고 그대로 제어 입력 u=v로 전달하며, Σ는 정상 동작을 계속 진행할 것이다. 외부 입력이 a_i로 바뀐다면 C는 a_i 값 대신 t의 입력 character를 전달함으로써 교정 제어를 시작한다. C가 t의 첫 번째 제어 입력 인자 u_1을 생성하여 Σ에 전달하면 선행 머신 Σ_1은 $s_1(x_i,u_1)=x^1$의 상태 천이를 거치면서 x^1에 도달한다. x^1이 후행 머신 Σ_2에 입력으로 들어가면 Σ_2는 $s_2(y_i,x^1):=y^1$의 상태 천이를 거치면서 y^1로 간다. y^1을 상태 피드백으로 받은 C도 다른 상태로 옮겨간 후 다시 두 번째 제어 입력 인자 u_2를 생성한다. u_2를 받은 Σ는 앞의 동작 연쇄적으로 반복하며, 최종적으로 Σ_2가 원하는 상태인 y'_i에 도달하면 교정 제어 동작이 끝나게 된다.

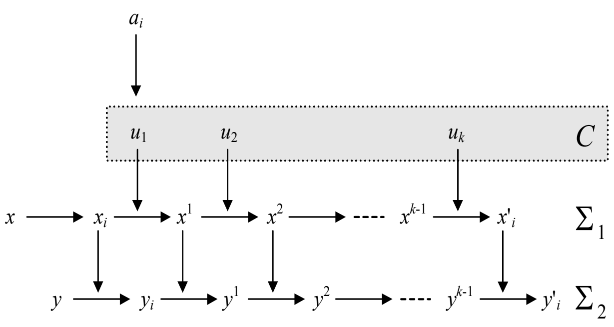

그림 2. (y_i,a_i,y'_i)에 대한 교정 제어 과정
Fig. 2. Corrective control procedure for the model mismatch (y_i,a_i,y'_i).

그림 2의 교정 제어 과정은 전력 클럭 없이 수행되므로 매우 빠르게 진행된다. 따라서 외부 사용자는 머신 Σ가 a_i를 받은 후 즉시 출력 y'_i를 내는 동작만을 관측할 것이다.

ii) $|X(y_i)|>1$, $|X_d(y_i)|\geq 1$

$|X(y_i)|>1$, $|X_d(y_i)|\geq 1$이면 $X(y_i)$의 원소가 둘 이상이며 이 중 모델 부정합 (y_i,a_i,y'_i)을 야기하는 $X_d(y_i)$의 원소가 하나 이상 존재하는 경우이다. 즉 외부 입력 a_i가 들어오면 모델 부정합이 발생할 수도, 발생하지 않을 수도 있으며, 동시에 선행 머신 Σ_1의 상태에 불확실성이 있는 경우이다. 하지만 C가 외부 입력 v를 알고 있으므로 식 (1)에서 정의한 $W(v,y_i)$를 이용하여 Σ_1의 상태 불확실성을 줄일 수 있다. 즉 현재의 외부 입력이 v라면 Σ_1의 현재 안정 상태는 $W(v,y_i)$ 중의 하나이다.

a) $|W(v,y_i)|=1$: 만약 $|W(v,y_i)|=1$이라면 Σ_1의 상태 불확실성은 소멸된다. 또 $|W(v,y_i)|=1$이고 $W(v,y_i)\subset X_d(y_i)$이면 현 상황은 앞의 i)의 경우와 일치한다. 선행 머신 Σ_1이 확실하게 $W(v,y_i)=X_d(y_i):=\{x_i\}$에 있으므로 교정 제어기 C가 존재할 필요충분조건은 식 (5)와 같다. $|W(v,y_i)|=1$이고 $W(v,y_i)\not\subset X_d(y_i)$이면 외부 입력이 a_i로 바뀌어도 모델 부정합이 발생하지 않으므로 C는 a_i가 들어온 후 아무런 제어 동작을 취하지 않는다.

b) $|W(v,y_i)|>1$: $|W(v,y_i)|>1$이면 Σ_1의 상태 불확실성이 여전히 존재한다. 하지만 $W(v,y_i)\cap X_d(y_i)=\varnothing$이면 Σ_1의 현재 상태에서 a_i가 들어와도 모델 부정합이 일어나지 않는다. 반면 $W(v,y_i)\cap X_d(y_i)\neq\varnothing$이면 Σ_1의 실제 상태에 따라서 외부 입력 a_i가 들어올 때 모델 부정합이 생길 수도 있고 생기지 않을 수도 있다. 이때는 제어 동작을 취하지 않으면 모델 부정합이 항상 발생한다는 가정 하에 C가 무조건 교정 제어를 실시해야 상태 불확실성에 상관없이 완전한 모델 정합을 이룰 수 있다. 즉 Σ가 출력 y_i를 낼 때 외부 입력이 a_i로 바뀌면 C는 교정 동작을 즉시 실시한다. 그런데 $|W(v,y_i)|>1$이므로 C가 Σ_1의 정확한 현재 상태를 알지 못한다. 따라서 임의의 $W(v,y_i)$의 원소에도 교정 동작이 성립되도록 하는 조건을 찾아야 한다.

i)과 마찬가지로 제어 입력 스트링을 $t:=u_1u_2\cdots u_k$라고 하자. 외부 입력이 a_i로 바뀌는 즉시 C는 t의 첫 번째 입력 인자 u_1을 생성하여 Σ에 전달한다. 그런데 Σ_1의 상태 불확실성 $W(v,y_i)$가 존재할 때에는($|W(v,y_i)|>1$) u_1이 $W(v,y_i)$의 모든 원소에 대하여 정의되어야 한다. 다시 말하면

$$s_1(x,u_1) \text{ is defined}, \ \forall x \in W(v,y_i) \quad (6)$$

와 같은 조건이 필요하다. 그림 2와 마찬가지로 u_1을 받은 후 Σ_2가 천이한 '다음 안정 상태'를 y^1이라 하자. 상태 피드백 y^1을 받은 순간 C는 Σ_1의 현재 상태에 대한 불확실성을 더 줄일 수 있다.

- 먼저 복합 머신 Σ가 u_1, y^1과 안정 조합을 이루고 있기 때문에 Σ_1의 현재 상태는 $W(u_1,y^1)$ 중 하나이다.
- 또한 Σ_1이 이전 불확실성 $W(v,y_i)$에서 외부 입력 u_1을 받아서 천이하였기 때문에 이 사실로부터 현재 상태에 대한 범위를 추정할 수 있다. 상태 집합 $W(v,y_i)$의 모든 원소가 입력 u_1에 대해서 천이하는 다음 안정 상태 집합을 편의상 $s_1[W(v,y_i),u_1]$로 표시하면 Σ_1의 현재 상태는 $s_1[W(v,y_i),u_1]$ 중 하나이다.

Σ_1의 현재 상태 불확실성을 $\Omega_1 \subset X$이라 하면 Ω_1은 위 두 집합의 교집합으로 표현 가능하다.

$$\Omega_1 = W(u_1,y^1) \cap s_1[W(v,y_i),u_1] \quad (7)$$

(6)과 (7)을 모든 $u_1,u_2,\ldots,u_k$에 대해 적용하면 입력 스트링 $t=u_1u_2\cdots u_k$가 만족시켜야 하는 일반적인 조건이 아래와 같이 유도된다.

$$\begin{aligned} & s_1(x,u_j) \text{ is defined}, \ \forall x \in W(u_{j-1},y^{j-1}) \\ & \Omega_j = W(u_j,y^j) \cap s_1[W(u_{j-1},y^{j-1}),u_j] \quad (8) \\ & u_0 := v, \ y^0 := y_i, \ y^k = y'_i, \ j=1,\ldots,k \end{aligned}$$

식 (8)에서 후행 머신 Σ_2가 마지막으로 천이하는 상태가 $y^k=y'_i$이다. 즉 위 식은 선행 머신 Σ_1의 상태 불확실성을 감내하면서 모델 정합을 이룰 수 있는 입력 스트링 t의 조건을 표현한다. 또 Ω_j는 제어 입력 인자 u_j가 들어온 후 Σ_1이 가지는 상태 불확실성 집합을 말한다. 식 (8)로부터 다음과 같은 관계가 성립한다.

$$|\Omega_1| \geq |\Omega_2| \geq \cdots \quad (9)$$

식 (9)는 머신이 상태 천이를 할수록 불확실성이 줄어든다는 일반적인 원리와 일치한다. 예를 들어 선행 머신 Σ_1의 현재 상태 불확실성이 Ω_j일 때 제어 입력 인자 u_{j+1}가 들어와서 후행 머신 가 y_{j+1}로 천이했다고 하자. 식 (8)에 따라서 Σ_1이 가지는 새로운 상태 불확실성 Ω_{j+1}은 Ω_j의 각 상태들이 u_{j+1}에 의해서 천이하는 상태 집합과 $W(u_{j+1},y^{j+1})$의 교집합으로 이루어지므로 Ω_j보다 더 커질 수는 없다. 그러므로 식 (9)가 유도된다.

새로운 제어 입력 인자를 머신 Σ에 넣을 때마다 Σ_1의 상태 불확실성은 점점 줄어들며, 어떤 index j에 대해서 $|\Omega_j|=1$이 되는 순간 Σ_1의 상태 불확실성은 소멸된다. 교정 제어기 C의 동작은 그림 1과 유사하므로 설명을 생략한다.

4. 제어기 설계

그림 3의 사례 연구 비동기 머신에 대한 모델 정합 문제를 고려한다. Σ_1과 Σ_2는 각각 세 개의 상태를 보유한 입력/상태 비동기 순차 머신이고 Σ는 두 머신의 직렬 결합으로 이루어진 복합 비동기 머신이다. Σ_1과 Σ_2의 입력과 상태 집합은 다음과 같다.

$$\begin{aligned} & A = \{a, b, c\} \\ & X = \{x_1, x_2, x_3\}, \ x_0 := x_1 \\ & Y = \{y_1, y_2, y_3\}, \ y_0 := y_1 \end{aligned}$$

두 머신의 직렬 결합 과정은 II장에서 기술한 내용대로 이루어졌으며, Σ의 모든 상태 천이는 기본 모드 원리를 만족시킨다. 또한 문제를 간단하게 하기 위해서 머신 Σ_1과 Σ_2의 상태 천이 함수는 모두 stable recursion 함수와 동일하도록 설정하였다. 즉

$$\begin{aligned} & f(x,v) = s_1(x,v) \ \forall (x,v) \in X \times A \\ & g(y,x) = s_2(y,x) \ \forall (y,x) \in Y \times X \end{aligned}$$

로 설정하였다.

그림 4는 기준 모델 Σ'의 상태 흐름도이다. 그림 3(c)의 Σ와 Σ'을 비교하여 다음과 같은 모델 부정합을 찾는다.

$$D = \{(y_3,a,y_2)\}$$

즉 Σ'은 안정 상태 y_3에 있을 때 외부 입력이 a로 바뀌면 다음 안정 상태 y_2로 천이하지만 Σ는 그림 3(c)에서 볼 수 있듯이 (x_2,y_3)에서 (x_1,y_1)로 천이하므로 출력이 y_1이 나온다.

교정 제어기의 존재 조건을 규명하기 위해서 $X(y_3)$와 $X_d(y_3)$를 구하면 아래와 같다.

$$X(y_3) = X_d(y_3) = \{x_2\}$$

Σ_2에서 y_3과 안정 조합을 이루는 Σ_1의 유일한 상태가 x_3이며, 외부 입력이 a로 바뀌면 모델 부정합이 일어난

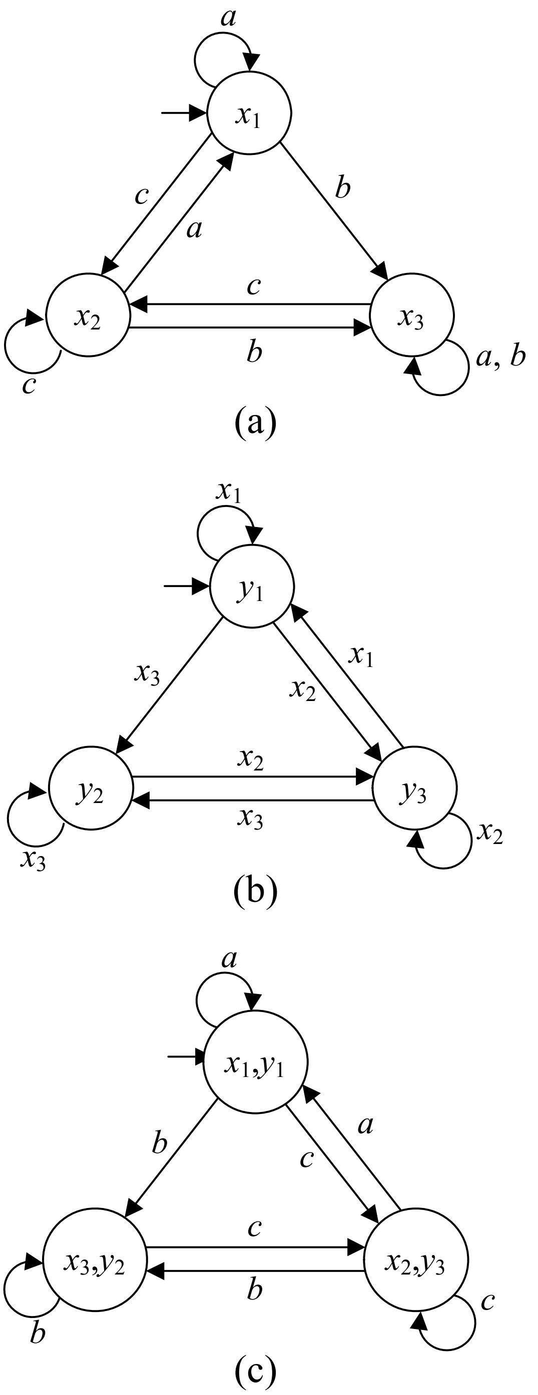

그림 3. 사례 연구 비동기 머신: (a) Σ_1, (b) Σ_2, (c) Σ
Fig. 3. Asynchronous machine Σ for case study: (a) Σ_1, (b) Σ_2, and (c) Σ.

다. $|X(y_3)|=|X_d(y_3)|=1$이므로 Ⅲ장에서 분류한 i)의 경우이다. 그림 3을 보면

$$s((x_2,y_3),ab) = (x_3,y_2)$$

인 입력 스트링 $t:=ab$가 존재한다. 따라서 조건 식 (5)가 만족되므로 교정 제어기 C를 꾸밀 수 있다. 또 Σ_2가

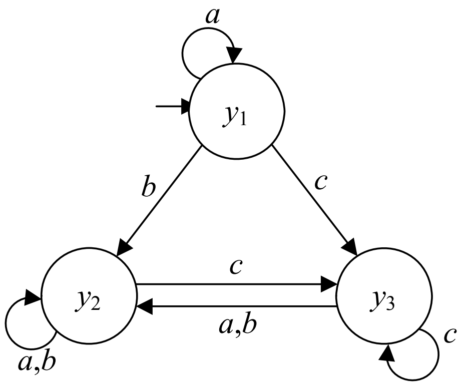

그림 4. 기준 모델 Σ'
Fig. 4. Reference model Σ'.

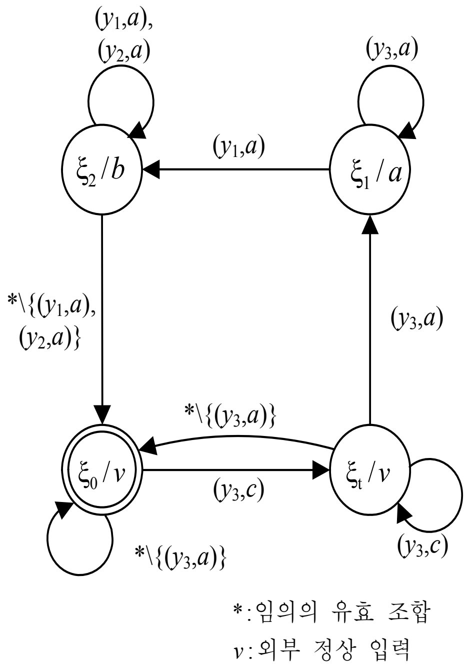

그림 5. 교정 제어기 C
Fig. 5. Corrective controller C.

y_3에 있을 때 Σ_1은 확실히 안정 상태 x_2에 있으므로 Σ_1의 상태 불확실성은 존재하지 않는다.

그림 5는 모델 부정합 (y_3,a,y_2)를 해결하는 교정 제어기 C이다. 기존 연구[6-9]에서 수행한 바대로 C는 $2+|t|=4$개의 상태를 가진다. 각 상태 마디에서 '/' 옆에 표시한 것은 현 상태에서 C가 내는 출력 값이다. 초기 상태 ξ_0에 있던 C는 Σ가 (x_2,y_3)과 안정 조합을 이루는 순간, 즉 Σ의 출력이 y_3으로 바뀌는 순간 transition 상태[6] ξ_t로 천이해야 한다. ξ_t에서 C는 외부 입력 변화를 보고 교정 제어 동작을 시작할지 안 할지를 결정한다. 외부 입력이 a가 아닌 다른 값으로 바뀌면 모델 부정합이 발생하지 않기 때문에 C는 ξ_0으로 복귀한다.

입력 a가 들어오는 순간 C는 교정 동작을 실시하기 위해 다음 상태 ξ_1로 천이한다. ξ_1에서 C는 먼저 t의 첫 번째 입력 인자 a를 Σ에 전달한다. a가 Σ에 들어가면 선행 머신 Σ_1은 $s_1(x_2,a)=x_1$로 천이하고, x_1을 받은 후행 머신 Σ_2는 $s_2(y_3,x_1)=y_1$로 이동한다(그림 3 참조). Σ의 출력 피드백 y_1을 받은 C는 다시 ξ_2 로 이동한 후 t의 두 번째 입력 인자 b를 생성한다. b를 받은 복합 머신 Σ는 결국 (x_3,y_2)로 이동하며 출력 y_2를 냄으로써 모델 정합이 완성된다. 교정 제어기 C는 외부 입력이 바뀌지 않는 한 상태 ξ_2에 머무르며, 새로운 외부 입력이 들어오면 초기 상태 ξ_0로 복귀한다.

그림 5의 제어 동작에서 선행 머신 Σ_1의 상태는 $X(y_3)=X_d(y_3)=\{x_2\}$에서 시작하였기 때문에 식 (9)에 의해서 C에게 주어진 Σ_1의 상태 불확실성은 없다. 상태 불확실성이 존재할 시에는 식 (8)에서 규명한 조건을 만족시키는 제어 입력 스트링을 구해서 그림 5와 같은 교정 제어 동작을 꾸며야 한다.

직렬 결합 비동기 머신에 대한 그림 5의 교정 제어기 설계 과정은 앞서 기술했듯이 선행 머신의 상태 불확실성을 모두 고려하고 있다. 이러한 설계 과정은 단일 머신에 대한 선행 연구에서는 유도될 수 없는 새로운 내용이다.

Ⅲ. 결 론

이번 논문에서는 두 입력/상태 비동기 머신이 직렬 결합된 복합 비동기 순차 머신에 대한 모델 정합 문제를 다루었다. 교정 제어기는 후행 머신의 상태 피드백만을 받기 때문에 선행 머신 상태에 대한 불확실성을 가지고 제어 동작을 해야 한다. 본 연구에서는 이러한 상태 불확실성에도 불구하고 폐루프 시스템의 안정 상태 동작을 주어진 기준 모델의 동작과 일치시키는 교정 제어기가 존재할 조건을 규명하였다. 또한 사례 연구를 통해서 제안된 제어기의 설계 과정을 예시하였다.

두 비동기 모듈이 직렬 결합되어 작동하는 실제 디지털 하드웨어 시스템에 본 논문의 연구 결과를 적용시키는 일이 향후 과제로 남아 있다. 고장 발생을 측정하는 가산기(accumulator)의 카운팅(counting) 상태가 출력으로 전달되어 직렬 결합된 오류 정정기(EDAC)의 입력으로 들어가는 인공위성 회로 등이 사례 연구 대상의 예이다.

참 고 문 헌

[1] T.-A. Chu, "Synthesis of self-timed VLSI circuits from graph-theoretic specifications," in Proceedings of International Conference on Computer Design (ICCD), pp. 220-223, 1987.

[2] A. J. Martin and M. Nyström, "Asynchronous techniques for system-on-chip design," Proceedings of IEEE, vol. 94, no. 6, pp. 1089-1120, 2006.

[2] A. J. Martin, M. Nyström, and C. G. Wong, "Three generations of asynchronous microprocessors," IEEE Design & Test of Computers, vol. 20, no. 6, pp. 9-17, 2003.

[4] J. Muttersbach, T. Villiger, and W. Fichner, "Practical design of globally-asynchronous locally-synchronous systems," in Proceedings of International Symposium on Advanced Research in Asynchronous Circuits and Systems, pp. 52-59, 2000.

[5] J. Hammer, "On corrective control of sequential machines," International Journal of Control, vol. 65, no. 2, pp. 249-276, 1996.

[6] 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.

[7] J.-M. Yang and J. Hammer, "Asynchronous sequential machines with adversarial intervention: the use of bursts," International Journal of Control, vol. 83, no. 5, pp. 956-969, 2010.

[8] 양정민, "교정 제어를 이용한 비동기 순차 머신의 영구 고장 극복", 전자공학회논문지, 제47권 SC 5호, pp. 9-17, 2010.

[9] 양정민, "간헐 고장이 존재하는 비동기 머신의 견실한 상태 피드백 제어", 전자공학회논문지, 제48권 SC 3호, pp. 40-47, 2011.

[10] X. Geng, Model Matching for Asynchronous Sequential Machines, Ph.D. dissertation, University of Florida, 2003.

[11] E. A. Lee and P. Varaiya, Structure and Interpretation of Signals and Systems, 2nd ed., LeeVaraiya.org, 2011.

[12] J.-M. Yang and S. W. Kwak, "Asynchronous correction for cascade composition of finite state machines," in Proceedings of IEEE International Conference on Systems, Man, and Cybernetics, pp. 2401-2406, 2012.

[13] Z. Kohavi, Switching and Finite Automata Theory, 2nd ed. New York: McGraw-Hill, 1978.

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 교수
<주관심분야 : 비동기 머신 제어, 보행 로봇 시스템 등>