



스위칭 비동기 순차 머신을 위한 모델 정합 교정 제어기 설계

Design of Corrective Controllers for Model Matching of Switched Asynchronous Sequential Machines

양정민[†]
Jung-Min Yang[†]

경북대학교 전자공학부
School of Electronics Engineering, Kyungpook National University

요약

본 논문에서는 교정 제어에 의한 스위칭 비동기 순차 머신의 모델 정합 문제를 다룬다. 스위칭 비동기 순차 머신은 스위칭 신호에 따라서 여러 개의 비동기 순차 머신 특성을 번갈아 가면서 가지는 시스템이라고 정의한다. 이번 연구에서 스위칭 시스템이 가질 수 있는 스위칭 시퀀스(sequence)는 일정하게 고정되어 있다고 가정한다. 제어 목적은 페루프 시스템의 안정 상태 동작을 주어진 기준 모델과 일치시키는 교정 제어기의 존재조건을 규명하고 제어기를 설계하는 일이다. 이를 위해서 스위칭 비동기 머신이 가지는 도달가능성을 표현하는 새로운 skeleton 행렬을 도입하고 모델 정합 교정 제어기의 존재 조건을 기술한다. 또한 사례 연구를 통해 스위칭 신호를 생성하면서 동시에 교정 제어 입력을 변화시키는 새로운 교정 제어 알고리즘을 제시한다.

키워드 : 스위칭 비동기 순차 머신, 모델 정합, 교정 제어, 스위칭 시퀀스

Abstract

This paper presents the solution to model matching of switched asynchronous sequential machines by corrective control. We propose a model of switched asynchronous sequential machines, in which the system can have different dynamics of asynchronous machines governed by a pre-determined sequence of switching. The control objective is to derive a corrective control law so that the stable state behavior of the closed-loop system can match that of a prescribed model. A new skeleton matrix is defined to represent the reachability of the switched asynchronous machine, and a novel control scheme is presented that interweaves the switching signal and the corrective control procedure. A design algorithm for the proposed controller is illustrated in a case study.

Key Words : Switched Asynchronous Sequential Machines, Model Matching, Corrective Control, Switching Sequence.

Received: Aug. 1, 2014
Revised : Feb. 11, 2015
Accepted: Feb. 17, 2015
[†]Corresponding author(jmyang@ee.knu.ac.kr)

1. 서론

교정 제어(corrective control)는 재설계를 거치지 않고도 비동기 순차 머신(asynchronous sequential machine)의 안정 상태(stable state) 동작을 바꿀 수 있다는 점에서 이산 시스템에 대한 새로운 자동 제어 기법으로 주목받고 있다[1-3]. 교정 제어의 핵심은 비동기 시스템이 전역 클럭(clock)에 지배 받지 않는다는 성질을 이용하여 피드백(feedback) 제어 과정을 매우 빨리 실행한다는 점이다. 이 과정에서 이미 설계된 비동기 머신의 천이 특성을 재설계를 하지 않고도 원하는 목적에 맞게 바꾸는 일이 가능하다. 교정 제어에 대한 이론적 연구[1-3] 이외에도 교정 제어를 이용하여 디지털 시스템에서 발생하는 고장을 탐지하고 극복하는 모듈을 개발하는 응용 연구도 소개되었다[4, 5].

이번 연구의 목적은 스위칭 비동기 순차 머신(switched asynchronous sequential machine)을 도입하고 스위칭 비동기 순차 머신에 대한 모델 정합(整合)(model matching) 문제를 해결하는 교정 제어기를 제안하는 일이다. 모델 정합이란 페루프 시스템의 안정 상태 동작을 주어진 기준 모델(reference model)의 동작과 일치시키는 문제이다. 스위칭 비동기 순차 머신은 스위칭 신호에 따라서 여러 개의 비동기 머신의 특성을 번갈아 가면서 가지는 시스템을 말한다. 스위칭 시스템(switched system)의 제어에 대한 연구는 연속/이산 시간 선형 시스템에 대해서 주로 이루어져 왔으나[6], 비동기 머신을 포함한 이산 사건 시스템(discrete event system)에 스위칭 시스템을 도입한 연구는 거의 없다. 본 논문에서 스위칭 비동기 머신을 연구하는 주요 동기는 다음과 같다.

이 논문은 2012년도 정부(미래창조과학부)의 재원으로 한국연구재단의 기초연구사업 지원을 받아 수행된 것임 (No. NRF-2012R1A2A2A01003419). 이 논문은 2010년도 정부(교육부)의 재원으로 한국연구재단의 기초연구사업 지원을 받아 수행된 것임 (No. NRF-2010-0007271).
This is an Open-Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

- 1) [2]의 결과를 비롯하여 최근 교정 제어 이론을 유전자 조절 네트워크(gene regulatory network)를 표현하는 주요 모델인 불리언 네트워크(boolean network)[7]의 제어에 적용하는 연구가 발표되었다. 또한 불리언 네트워크가 묘사하는 진핵세포(eukaryotic cell) 등의 운동이 실제로 여러 개의 스위칭 모델로 표현되기 때문에 스위칭 불리언 네트워크(switched boolean network)에 대한 연구도 활발히 진행 중이다[8]. 그런데 세포의 운동은 본질적으로 비동기적 특성을 가지지만 기존 연구에서는 비동기 상태 천이 등을 다루지 않았다. 본 연구는 스위칭 모델에 비동기적 운동 개념을 도입하여 이러한 간극을 메꾸고자 한다.
- 2) 스위칭 모델은 하드웨어 여유도(redundancy)를 가진 내고장성 시스템(fault-tolerant system)[9]도 표현 가능하다. 비슷한 동작을 하는 여러 개의 부(副)-시스템으로 구성된 복합 시스템에서 하나의 부-시스템이 고장 나면 다른 부-시스템으로 현재 상태를 천이시킨 다음 고장 난 부-시스템이 정상 동작으로 복구할 때까지 시스템 운용을 계속 진행할 수 있다. 이 개념을 비동기 머신에서 생각하면 스위칭 비동기 머신에 대한 교정 제어 문제로 일반화시킬 수 있다.

본 논문에서는 스위칭 비동기 머신 모델링을 제안하고 제어 대상 스위칭 비동기 머신의 안정 상태 페루프 동작을 주어진 기준 모델과 일치시키도록 하는 모델 정합 교정 제어기의 필요충분조건을 규명한다. 이를 위해서 스위칭 비동기 머신의 도달가능성을 표현하는 새로운 skeleton 행렬을 정의한다. 스위칭 비동기 머신에 대한 교정 제어 과정은 제어 입력과 스위칭 신호를 함께 변화시켜야 하기 때문에 기존 연구[1-3]와는 다른 접근 방법이 필요하다. 이번 연구에서는 대상 시스템이 스위칭 신호가 가지는 여러 가지 특성 중 결정적 시퀀스(deterministic sequence)[10]를 가진다고 설정한다. 즉 스위칭 신호의 시퀀스는 정해져 있으며 시스템은 이 순서로 스위칭을 해야 한다. 또한 사례 연구를 통해 제안된 교정 제어기를 설계하는 과정을 예시한다.

2. 스위칭 비동기 순차 머신

m개의 부-머신으로 구성된 스위칭 비동기 순차 머신 Σ 를 다음과 같이 정의하자.

$$\begin{aligned} \Sigma &= \{(\Sigma^i, i \in M), \sigma, (1, x_0)\} \\ \Sigma^i &= (A, X, f_i) \end{aligned} \tag{1}$$

$M = \{1, 2, \dots, m\}$ 이며 Σ^i 는 i번째 부-머신이고 A는 입력 집합, $X = \{x_1, \dots, x_n\}$ 는 상태 집합, $f_i: X \times A \rightarrow X$ 는 Σ^i 의 상태 천이 함수이다. 모든 부-머신은 동일한 입력 집합과 상태 집합을 가진다. 또한 Σ^i 는 현재 상태가 출력으로 나오는 입력/상태 머신이라고 설정한다. $\sigma: N \rightarrow X$ 는 스위칭 신호이며 앞서 설정했듯이 일정한 시퀀스를 가진다. 기술의 편의상 Σ 가 가지는 스위칭 시퀀스를 $1 \rightarrow 2 \rightarrow \dots \rightarrow m$ 이라 하자(index를 재조절하면 M에서 생길 수 있는 임의의 시퀀스를 이와 같이 표현할 수 있다). 스텝(step) $k \in N$ 에서 $\sigma(k)$ 의 값은 아래와 같이 나온다.

$$\begin{aligned} \sigma(k) &= \sigma(k-1)+1 && \text{if } \sigma(k-1) < m \\ &= 1 && \text{if } \sigma(k-1) = m \end{aligned} \tag{2}$$

$(1, x_0)$ 에서 1은 Σ 가 가지는 초기 부-머신의 index이며 $x_0 \in X$ 는 초기 상태이다. 다시 말하면 Σ 는 부-머신 Σ^i 의 상태 x_0 에서 동작을 시작한다.

비동기 머신 Σ^i 의 상태 x는 현재 입력 a에 따라서 안정 상태(stable state)와 과도 상태(transient state) 중 하나에 속한다. (x, a) 가 안정 조합이라면 $f_i(x, a) = x$ 이며 입력이 a로 유지되는 한 Σ^i 는 x에 계속 머무른다. 입력이 $f_i(x, b) \neq x$ 인 b로 바뀐다면 x는 과도 상태가 되고 Σ^i 는 상태 천이를 시작한다. Σ^i 는 일반적으로 여러 개의 과도 상태를 거친 후 $f_i(x', b) = x'$ 인 다음 안정 상태(next stable state) x' 에 도달한다. 모든 상태 천이는 전역 클럭 없이 실행되므로 Σ^i 가 x에서 x'에 도달하는 시간은 극히 짧으며 중간에 거치는 과도 상태는 외부 사용자에게 관측되지 않는다. 따라서 과도 상태를 생략하고 현재 안정 상태와 다음 안정 상태만으로 임의의 상태 천이를 표현하는 것이 편리하다. 이를 위해서 stable recursion 함수 $s_i: X \times A \rightarrow X$ 를 아래와 같이 정의한다.

$$s_i(x, a) := x' \quad (x' \text{는 } (x, a) \text{의 다음 안정 상태}) \tag{3}$$

s_i 의 정의구역을 단위 입력에서 입력 스트링(string)까지 확장하면 다음과 같다.

$$s_i(x, ut) := s_i(s_i(x, u), t), \quad x \in X, u \in A, t \in A^+ \tag{4}$$

위 식에서 A^+ 는 A에 속한 원소로 이루어지는 길이 1 이상의 스트링 집합을 말한다. 본 연구에서는 f_i 와 s_i 가 정의구역 $X \times A$ 에 대한 전역 함수라고 설정한다. 즉 A의 임의의 원소는 상태 x와 과도 조합 또는 안정 조합을 이룬다. 비동기 디지털 시스템 등 현존하는 비동기 머신에서도 어떤 상태에서 과도 상태 천이를 일으키지 않는 입력이 들어오면 머신은 현재 상태를 그대로 유지하므로[4, 5], 즉 그 입력과 안정 상태를 이루기 때문에 이 가정은 일반성을 잃지 않는다.

$U_i(x) \subset A$ 와 $T_i(x) \subset A$ 를 Σ^i 에서 x와 안정 조합 및 과도 조합을 이루는 입력 집합이라고 각각 정의하자. 즉 $\forall a \in U_i(x)$ 에서 $s_i(x, a) = x$ 이며, $\forall b \in T_i(x)$ 에서 $s_i(x, b) \neq x$ 이다. s_i 가 전역 함수이므로

$$U_i(x) \cap T_i(x) = \emptyset, \quad U_i(x) \cup T_i(x) = A, \quad \forall x \in X \tag{5}$$

이다.

그림 1은 본 논문에서 제안하는 스위칭 비동기 머신에 대한 교정 제어 시스템이다. C는 교정 제어기이며 역시 비동기 머신 형태로 설계된다. $v \in A$ 는 외부 입력이며, $u \in A$ 와 $\sigma \in M$ 는 C가 생성하는 제어 입력과 스위칭 신호를 각각 가리킨다. u는 디멀티플렉서(demultiplexor)를 거쳐서 현재 선택된 부-머신으로 들어간다. C가 σ 를 생성하면 그림에 나와 있듯이 정해진 스위칭 시퀀스에 기준하여 Σ 가 가지는 부-머신이 바뀌고, 이 정보는 디멀티플렉서에 전달되어 선택된 부-머신이 제어 입력 u를 받는다. m개의 부-머신으로부터 나오는 상태 피드백이 멀티플렉서(multiplexor)를 거치며, 이 중 Σ 가 현재 가지는 부-머신의 상태 값 x가 선택되어 C에 입력된다.

제어 목적은 Σ 와 C가 결합된 페루프 시스템의 안정 상태 동작을 주어진 기준 모델 $\Sigma_f = (A, X, s_f)$ 과 일치시키도록 하는 C를 설계하는 일이다. '안정 상태 동작'이란 과도 상태를 제외한 비동기 머신의 상태 천이 특성을 말한다. 즉 임의의 상태-입력 조합에서 페루프 시스템의 stable recursion 함수가 s_f 과 동일한 동작을 보이도록 하는 것이 제어 목적이다.

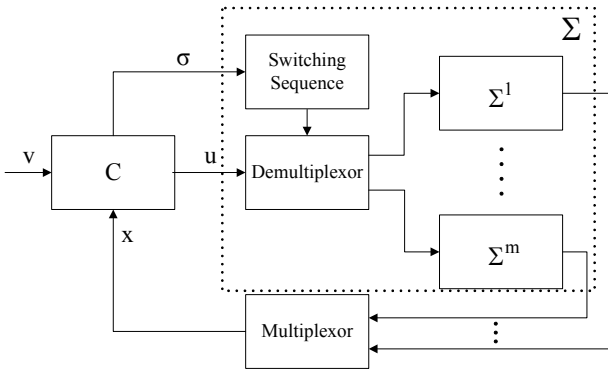


그림 1. 스위칭 비동기 머신에 대한 교정 제어 시스템
Fig. 1. Corrective control system of the switched asynchronous machine

비동기 머신이 예측 가능한 천이 특성을 보이기 위해서는 두 개 이상의 변수가 동시에 변하지 말아야 하는 '기본 모드 (fundamental mode operation)'로 항상 동작해야 한다[11]. 그림 1의 페루프 시스템이 기본 모드를 준수하기 위해서는 다음 조건이 만족되어야 한다([11]의 Proposition 4로부터 유도하였다).

조건 1: 기본 모드 준수 조건

- 1) Σ 가 과도 상태에 있을 때 C는 안정 상태에 있어야 하며, C가 과도 상태에 있을 때 Σ 는 안정 상태에 있어야 한다.
- 2) u 와 σ 값은 동시에 변하지 말아야 한다.

조건 1,2)의 반례를 들어 Σ 가 부머신 Σ^i 의 안정 상태 x 에 있을 때 u 와 σ 값이 동시에 변한다고 가정하자. 이전 부머신의 index가 i 이므로 $\sigma=i+1$ 이다($i < m$ 이라 가정). 그런데 u 와 σ 가 동시에 변하기 때문에 다음 두 가지 가능성이 공존한다.

- i) Σ 는 부머신 Σ^{i+1} 의 상태 x 로 스위칭한 후 u 를 입력 받아 $s_{i+1}(x,u)$ 로 천이한다(σ 가 u 보다 먼저 변할 때).
 - ii) Σ 는 Σ^i 에서 u 에 의해 $s_i(x,u)$ 로 천이한 후 $\sigma=i+1$ 을 입력 받고 Σ^{i+1} 의 상태 $s_{i+1}(x,u)$ 로 스위칭 한다(u 가 σ 보다 먼저 변할 때).
- i)과 ii)를 비교해서 알 수 있듯이 조건 1이 만족되지 않으면 Σ 의 다음 동작을 정확하게 예측하지 못한다.

3. 제어기 존재 조건: 도달가능성

3.1 단일 비동기 머신에 대한 도달가능성

본 장에서 스위칭 비동기 머신에 대한 모델 정합 교정 제어기 존재 조건을 규명한다. 제어기 설계 과정은 다음 장 사례 연구에서 예시한다.

단일 비동기 머신의 경우 교정 제어기의 존재 여부는 머신의 도달가능성으로 표현되며, 머신의 도달가능성은 다시 'skeleton 행렬'로 간략하게 기술 가능하다[1, 3]. 본 논문에서는 이전 연구에서 도입한 skeleton 행렬을 소개하고 스위칭 머신의 도달가능성을 표현하는 새로운 skeleton 행렬과 행렬 연산을 정의한다.

정의 1: skeleton 행렬[1]

stable recursion 함수 s_i 를 가지는 단일 비동기 순차 머신 $\Sigma^i=(A, X, f_i)$ 의 skeleton 행렬 $K(\Sigma^i)$ 는 $n \times n$ 차 행렬이며 (p,q) 번째 원소 $K_{p,q}(\Sigma^i)$ 는 다음과 같다($p,q \in \{1, \dots, n\}$).

$$K_{p,q}(\Sigma^i) = \begin{cases} 1 & \exists t \in A^+ \text{ s.t. } s_i(x_p, t) = x_q \wedge |t| \leq n-1 \\ 0 & \text{otherwise} \end{cases}$$

다시 말하면 $s_i(x_p, t) = x_q$ 인 입력 스트링 t 가 존재하면 x_q 는 x_p 로부터 도달가능하며 $K_{p,q}(\Sigma^i) = 1$ 로 표현한다. Σ^i 의 페루프 동작을 주어진 기준 모델 Σ_r 과 일치시키는 교정 제어기가 존재하기 위해서는 임의의 상태에서 Σ_r 의 도달가능성이 Σ_r 보다 크거나 같아야 한다. 이 조건을 skeleton 행렬로 나타내면

$$K(\Sigma^i) \geq K(\Sigma_r) \tag{6}$$

이다[1-3]. 위 식에서 부등식은 두 행렬의 모든 원소 사이에서 성립되어야 한다.

3.2 스위칭 비동기 머신에 대한 도달가능성

스위칭 비동기 머신 Σ 는 m 개의 부머신 사이를 스위칭하는 동특성을 보이기 때문에 단일 비동기 머신보다 더 큰 도달가능성을 가진다는 사실은 자명하다. 하지만 Σ 의 정확한 도달가능성을 유도하기 위해서는 다음 두 가지 제한 사항을 고려해야 한다. 먼저 Σ 는 시퀀스 $1 \rightarrow \dots \rightarrow m \rightarrow 1 \rightarrow \dots$ 에 맞추어서 스위칭을 해야 한다. 예를 들어 최초 구동 시 Σ 는 부머신 Σ^1 에서 시작하는데 스위칭을 통해 도달하는 다음 부머신은 반드시 Σ^2 이어야 한다. 또한 Σ 가 스위칭을 할 때 현재의 안정 상태 특성이 유지되지 못하는 경우가 생긴다. 예를 들어 Σ 가 현재 부머신 Σ^i 의 상태 x 에 있다고 하자. 조건 1에 의해서 Σ 가 스위칭을 하려면 x 는 안정 상태이어야 하므로 Σ 는 $a \in U_i(x)$ 인 어떤 안정 조합 (x,a) 에 있다. 이때 C가 스위칭 신호를 생성하여 Σ 가 Σ^{i+1} 로 옮겨간다고 하자. 기본 모드 원칙에 따라서 입력 값은 변하지 않기 때문에 스위칭 직후 외부 입력은 여전히 a 이다. 그런데 부머신 Σ^{i+1} 의 상태 x 가 a 와 안정 조합을 이루지 않는다면, 즉 $a \notin U_{i+1}(x)$ 라면 스위칭 직후 (x,a) 가 과도 조합이 되기 때문에 과도 상태 천이가 일어난다. 물론 스위칭을 하기 전 C가 제어 입력을 $b \in U_i(x) \cap U_{i+1}(x)$ 인 b 로 바꾸면 스위칭 후에도 안정 상태 x 는 유지된다. 하지만

$$U_i(x) \cap U_{i+1}(x) = \emptyset \tag{7}$$

라면 두 머신 Σ^i 와 Σ^{i+1} 에서 모두 상태 x 와 안정 조합을 이루는 입력 b 가 존재하지 않기 때문에 스위칭을 한 후 안정 상태 x 가 절대 유지되지 못한다. Σ^i 와 Σ^{i+1} 사이에 위의 관계가 성립된다면 $U_i(x)$ 와 $T_{i+1}(x)$ 의 포함 관계도 살펴봐야 한다. 만약

$$U_i(x) = T_{i+1}(x) \tag{8}$$

라면 스위칭 이후 안정 상태 x 를 유지하지는 못하지만 x 의 도달가능성은 잃지 않는다. 예를 들어 Σ 가 Σ^i 의 안정 상태 x 에 있을 때 교정 제어와 스위칭을 통해 Σ 를 x 에서 x' 로 이동시켜야 한다고 가정하자. 또 Σ^i 는 x 에서 x' 까지 가는 도달가능성을 가지고 있지 않지만 Σ^{i+1} 은 가지고 있다고 하자. 즉 $s_{i+1}(x, u_1 \dots u_k) = x'$ 인 입력 스트링 $u_1 \dots u_k$ 가 존재한다(이때 (x, u_1) 은 과도 조합임이 자명하다). $U_i(x) = T_{i+1}(x)$ 라면 $u_1 \in T_{i+1}(x)$ 이고 $u_1 \in U_i(x)$ 이다. 따라서 스위칭하기 직전 C가 제어 입력을 u_1 로 바꾸면 Σ^i 에서 안정 상태 x 가 유지되고, 스위칭 직후 Σ^{i+1} 이 과도 조합 (x, u_1) 에 진입하면서 그 다음 제어 입력 스트링에 의해서 x' 까지 도달하는 교정 제어 과정이 이루어진다.

어진다.

$U_i(x) \cap U_{i+1}(x) = \emptyset$ 이지만 $U_i(x) \neq T_{i+1}(x)$ 라면 $U_{i+1}(x) \cup T_{i+1}(x) = A$ 라는 관계로부터

$$U_i(x) \supseteq T_{i+1}(x) \tag{9}$$

가 유도된다. 즉 $U_i(x)$ 는 $T_{i+1}(x)$ 의 진부분집합(proper subset)이다. 이것은 Σ^{i+1} 의 상태 x 와 과도 조합을 이루는 입력이 Σ^i 의 상태 x 와 안정 조합을 이루는 입력 집합 $U_i(x)$ 이외에도 존재할 수 있다는 의미이다. 따라서 Σ^i 의 상태 x 에서 스위칭이 실행되어 Σ^{i+1} 로 이동하면 x 에서 출발하는 Σ^{i+1} 의 도달가능성 일부를 잃을 수 있다.

이상의 논의를 일반화하여 스위칭 비동기 머신 Σ 의 도달가능성을 표현해보자. 먼저 m 개의 부머신에 대한 skeleton 행렬 $K(\Sigma^i)$ 를 정의 1을 이용하여 구한다. 그런 다음 각 skeleton 행렬로부터 '스위칭을 고려한 skeleton 행렬' $W(\Sigma^i) \in \{0,1\}^{n \times n}$ 을 다음과 같이 정의한다.

정의 2: 스위칭을 고려한 skeleton 행렬 $W(\Sigma^i)$

i=1: $W(\Sigma^1) := K(\Sigma^1)$

2 ≤ i ≤ m: $K(\Sigma^i)$ 와 $W(\Sigma^i)$ 의 p 번째 행(row) 벡터를 $\kappa_p, \omega_p \in \{0,1\}^{1 \times n}$ 라 하고, κ_p 와 ω_p 의 q 번째 원소를 $\kappa_p(q), \omega_p(q)$ 라 표기하자($p, q \in \{1, \dots, n\}$). 즉 $\kappa_p(q) = K_{p,q}(\Sigma^i), \omega_p(q) = W_{p,q}(\Sigma^i)$ 이다.

- 1) $U_{i-1}(x_p) \cap U_i(x_p) \neq \emptyset$ 이면 $\omega_p := \kappa_p$ 로 설정한다.
- 2) $U_{i-1}(x_p) \cap U_i(x_p) = \emptyset$ 이고
 - i) $U_{i-1}(x_p) = T_i(x_p)$ 이면 $\omega_p := \kappa_p$ 로 설정한다.
 - ii) $U_{i-1}(x_p) \supseteq T_i(x_p)$ 이면 Σ^i 의 x_p 에서 $T_i(x_p)/U_{i-1}(x_p)$ 에 속한 입력에 의해서만 도달하는 상태의 index를 q_1, \dots, q_r 이라 하자. $\omega_p(p) = \omega_p(q_1) = \dots = \omega_p(q_r) = 0$ 으로 설정하고 나머지 ω_p 의 원소는 κ_p 와 일치시킨다.

정의 2의 의미를 부연 설명하면 다음과 같다. 1)은 Σ^{i-1} 과 Σ^i 에서 각각 상태 x_p 와 안정 조합을 이루는 동일한 입력이 존재하는 경우이다. Σ^{i-1} 에서 Σ^i 로 스위칭한 후 머신은 계속 안정 상태 x_p 에 머무므로 x_p 에서 시작하는 Σ^i 의 도달가능성은 변하지 않는다($\omega_p := \kappa_p$). 2.i)은 1)과 유사한 경우이다. Σ^{i-1} 에서 Σ^i 로 스위칭한 후 안정 상태 x_p 를 유지할 수는 없지만 $U_{i-1}(x_p) \cap U_i(x_p) = \emptyset$, x_p 에서 시작하는 도달가능성을 구현하는 임의의 입력을 스위칭하기 전에 제어 입력으로서 미리 생성할 수 있다($U_{i-1}(x_p) = T_i(x_p)$). 따라서 p 번째 행에서의 skeleton 행렬 원소 값은 그대로 유지된다. 마지막으로 2.ii)는 스위칭한 후 안정 상태 x_p 를 유지할 수 없는 동시에 부머신 Σ^i 의 x_p 가 가지는 도달가능성도 스위칭으로 인해서 일부 상실되는 경우이다. 가정에 의해 $U_{i-1}(x_p) \supseteq T_i(x_p)$ 이므로 $T_i(x_p)/U_{i-1}(x_p)$ 는 공집합이 아니다. 여기서 '상실되는 도달가능성'이란 Σ^i 가 x_p 에서 $T_i(x_p)/U_{i-1}(x_p)$ 에 속한 입력에 의해서만 도달 가능한 상태들을 말한다. 따라서 그러한 상태의 index에 해당하는 ω_p 원소의 값은 0이어야 한다. 또한 스위칭을 거친 후 안정 상태 x_p 에 머무를 수 없기 때문에 $\omega_p(p)$ 도 0이다.

skeleton 행렬의 원소가 가지는 값은 0 또는 1밖에 없기 때문에 두 skeleton 행렬의 곱셈(product) 연산을 해당 행과 열(column) 벡터의 논리 곱의 합(logic sum of product)으로 정의하면 곱셈의 결과도 skeleton 행렬이 된다. 이를 위해 임의의 두 skeleton 행렬 $D, E \in \{0,1\}^{n \times n}$ 의 곱 DE 를 아래와 같이 정의하자.

$$DE_{p,q} := \sum_{i=1}^n (D_{p,i} \cdot E_{i,q}), \quad p, q \in \{1, \dots, n\} \tag{10}$$

위 식에서 덧셈 '+'은 논리 합(logic OR), 곱셈 '.'은 논리 곱(logic AND)을 각각 의미한다. 행렬 곱과 유사하게 D 와 E 의 행렬 합도 아래와 같이 정의하자.

$$(D+E)_{p,q} := D_{p,q} + E_{p,q} \quad ('+'는 논리 합) \tag{11}$$

정의 2의 스위칭을 고려한 skeleton 행렬 $W(\Sigma^i)$ 와 위에서 도입한 행렬 곱과 행렬 합을 이용하여 시퀀스 $(1, 2, \dots, m)$ 을 가지는 스위칭 비동기 머신 Σ 의 skeleton 행렬 $K(\Sigma)$ 를 아래와 같이 정의한다.

정의 3: 시퀀스 $(1, 2, \dots, m)$ 을 가지는 스위칭 비동기 머신 Σ 의 skeleton 행렬 $K(\Sigma)$ 은 아래와 같이 정의된다.

$$K(\Sigma) := W(\Sigma^1) + W(\Sigma^1)W(\Sigma^2) + \dots + W(\Sigma^1)W(\Sigma^2) \dots W(\Sigma^m) \tag{12}$$

$K(\Sigma)$ 의 의미를 이해하기 위해서는 스위칭 머신 Σ 가 가지는 도달가능성은 개별 부머신이 가지는 도달가능성의 총합에서 스위칭을 거치면서 손상되는 도달가능성을 빼 것이라는 사실을 알아야 한다. Σ 는 초기에 부머신 Σ^1 에서 동작을 시작하기 때문에 스위칭을 하기 전 Σ^1 의 도달가능성을 그대로 물려받는다. 따라서 정의 3의 식에서 볼 수 있듯이 $K(\Sigma)$ 는 $K(\Sigma^1) (= W(\Sigma^1))$ 를 포함한다. 또한 스위칭을 거쳐 Σ^1 에서 Σ^2 로 이동한다면 정의 2로부터 $W(\Sigma^1)W(\Sigma^2)$ 에 해당하는 도달가능성을 획득한다. 스위칭 시퀀스 $(1, \dots, m)$ 에 따라 Σ 는 Σ^1 에서 Σ^m 까지 이동할 수 있기 때문에 각 부머신에서 Σ 가 가지는 도달가능성을 모두 더하면 정의 3의 식과 같이 $K(\Sigma)$ 가 표현된다.

정의 3의 결과는 비동기 머신의 특성을 고려하지 않은 불리안 네트워킹의 결과(8)와 대비된다. 스위칭을 했을 때 현재의 안정 상태가 그대로 유지된다고 한다면 Σ 의 도달가능성은 모든 부머신이 지나는 원래의 도달가능성을 합친 값이므로 $K(\Sigma) = K(\Sigma^1) + \dots + K(\Sigma^m)$ 이 될 것이다. 이는 [8]에서 소개된 스위칭 불리안 네트워크의 도달가능성 유도 과정과 동일한 결과이다. 하지만 기본 모드를 준수하는 그림 1의 스위칭 비동기 머신 Σ 에서는 현재 입력과 현재 상태 사이의 관계를 반드시 고려해야 하기 때문에 정의 2와 같은 스위칭을 고려한 skeleton 행렬 $W(\Sigma^i)$ 를 사용해야 한다.

3.3 모델 정합 교정 제어기 존재조건

이제 $K(\Sigma)$ 를 이용하여 모델 정합 교정 제어기가 존재할 조건을 기술한다. 아래 정리는 본 연구의 주요 결과이다.

정리 1: m 개의 부머신 $\Sigma^i = (A, X, f_i)$ 과 시퀀스 $(1, \dots, m)$ 을 가지는 스위칭 비동기 머신 $\Sigma = (\Sigma^i, i \in M, \sigma, (1, x_0))$ 에 대해서 기준 모델 $\Sigma_r = (A, X, s_r)$ 이 주어질 때 모델 정합을 구현하는 그림 1의 교정 제어기 C 가 존재할 필요충분조건은 아래와 같다.

$$K(\Sigma) \geq K(\Sigma_r) \tag{13}$$

$K(\Sigma)$ 은 정의 3에서 정의한 스위칭 머신 Σ 의 skeleton 행렬이고 $K(\Sigma_r)$ 은 기준 모델 Σ_r 의 skeleton 행렬이다.

$$W_{p,q}(\Sigma^1) + [W(\Sigma^1)W(\Sigma^2)]_{p,q} + \dots + [W(\Sigma^1) \dots W(\Sigma^m)]_{p,q} = 1 \quad (14)$$

이다. 위 식은 (p,q)번째 원소 값이 1인 skeleton 행렬 곱이 적어도 하나 존재한다는 뜻이다. 이 중 곱해진 skeleton 행렬 개수가 최소인 행렬을 선택한 후 그 행렬 안에서 $x_p \rightarrow x_q$ 를 구현하는 입력 스트링 중 최소 길이의 스트링을 찾아서 제어 입력 스트링으로 선택하면 된다. 그림 2는 정리 1과 정리 2에서 규명한 모델 정합 교정 제어기의 존재 조건과 제어 입력 스트링 선택 기준을 포함하는 제어기 설계 알고리즘을 나타낸 흐름도이다. 교정 제어기의 상세한 설계 과정은 다음 장 사례 연구에서 기술한다.

4. 사례 연구

제한한 교정 제어기의 존재조건 규명 및 설계 과정을 예시하기 위하여 그림 3에 나타난 부머신 $\Sigma^1, \Sigma^2, \Sigma^3$ 으로 구성된 스위칭 비동기 머신 Σ 와 그림 4에 주어진 기준 모델 Σ_r 을 생각하자. Σ 와 Σ_r 의 입력 및 상태 집합은 $A=\{a, b, c, d\}, X=\{x_1, x_2, x_3, x_4\}$ 이다. $\Sigma^1 \sim \Sigma^3$ 과 Σ_r 은 모두 비동기 누산기(asynchronous accumulator)[12]의 한 종류이다. 누산기는 인공위성 등에 장착되어 메모리 비트(bit) 오류를 카운팅(counting)하는 데 주로 사용되는 디지털 회로이다. X에 속한 각 상태의 의미는 아래 표 1과 같다.

표 1. $x_1 \sim x_4$ 의 의미
Table 1. Meaning of $x_1 \sim x_4$

X	Meaning
x_1	Initial state
x_2	Accumulation of 1 bit
x_3	Accumulation of even bits
x_4	Accumulation of odd bits

x_1 은 누산기의 초기 상태이며, $x_2 \sim x_4$ 는 누산기가 1 비트, 짝수 비트, 그리고 홀수 비트를 카운팅하여 그 결과를 저장 소자(storage element)에 보관하는 작업을 의미한다. 여기서 “1 비트”, “짝수 비트”, 그리고 “홀수 비트”는 누적된 비트 수를 의미한다. 즉 누산기가 x_2 에 있으면 현재 1 비트가 누적되었다는 것을 뜻하며, x_3 또는 x_4 에 머무르면 지금까지 누적된 비트 수가 짝수 또는 홀수라는 것을 뜻한다. 그림 4의 기준 모델 Σ_r 은 다음과 같은 동작 특성을 보인다. 먼저 초기 상태 x_1 에서 외부 입력에 따라 x_2 또는 x_4 로 천이할 수 있다. Σ_r 가 x_2 로 천이했다고 하자. 그림 4에서 Σ_r 은 x_2 에서 다시 x_1 로 복귀하거나 x_3 으로 천이 가능하다. 이것은 누산기가 1 비트가 누적되었다는 정보를 출력한 후 다시 초기 상태로 되돌아가거나($s_c(x_2, a) = x_1$), 다음 입력 비트를 받고 짝수 비트가 누적되었다는 정보를 알려주는 상태 x_3 으로 이동한다는 뜻이다. Σ_r 가 x_3 으로 천이한 후에는 후속 입력 비트에 따라 현재 누적된 비트 수의 패리티(parity)를 출력하는 동작을 계속 수행한다(즉 x_3 와 x_4 사이를 이동). 또한 Σ_r 은 x_1 에서 입력 비트에 반응하여 x_4 로 즉시 이동하여 누적 비트 수의 패리티 출력 모드를 시작할 수도 있다($s_c(x_1, c) = x_4$).

Σ 은 $m=3$, 즉 세 개의 부머신을 가지며, 스위칭 시퀀스는 $1 \rightarrow 2 \rightarrow 3 \rightarrow 1 \rightarrow \dots$ 이다. 주어진 문제는 Σ 와 교정 제어기 C로

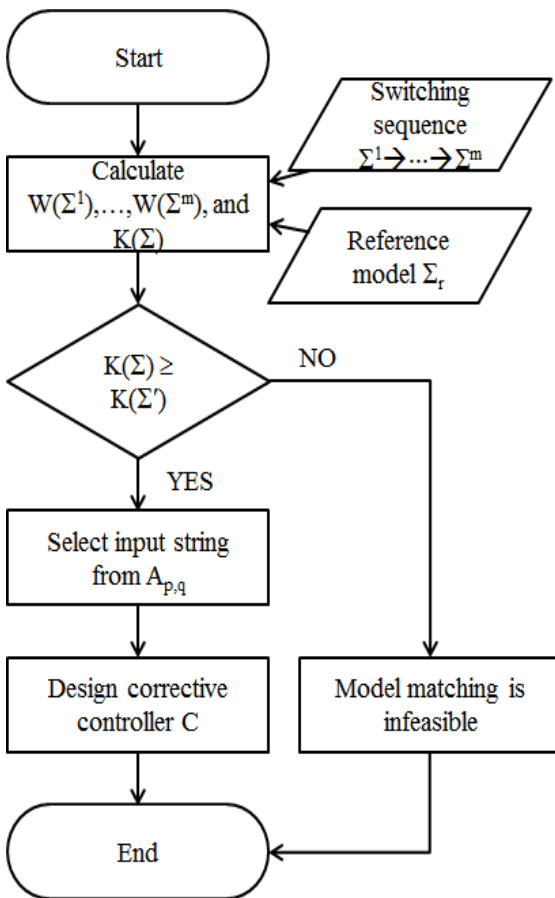


그림 2. 교정 제어기 설계 알고리즘

Fig. 2. Algorithm for designing the corrective controller

교정 제어기 C를 설계할 때에는 두 상태 간의 도달가능성을 실현하는 입력 스트링을 사용한다. 입력 스트링의 길이가 길수록 C의 상태 개수도 증가하므로[1] C의 크기를 줄이기 위해서는 되도록 길이가 짧은 스트링을 제어 입력으로 사용해야 한다. 그런데 이번 연구에서는 C가 제어 입력뿐 아니라 스위칭 신호 σ 도 생성한다. 그림 1에 나타냈듯이 스위칭 신호가 생성되면 디멀티플렉서가 작동되어 부머신에 들어가는 입력 채널을 바꾸어야 하며, 피드백 과정에서도 멀티플렉서를 작동시켜야 한다. 하드웨어 구현 관점에서 보면 스위칭 과정에서 소요되는 부하(負荷)가 제어기 상태 개수를 늘리는 일보다 더 크다. 따라서 C의 구현 시 C의 부하를 최소화시키는 제어 입력 스트링을 구하는 원칙은 아래와 같이 나온다.

정리 2: (1)의 스위칭 비동기 순차 머신 Σ 와 기준 모델 Σ_r 이 정리 1의 조건 (13)을 만족시킨다고 하자. $K_{p,q}(\Sigma_r)=1$ 인 임의의 상태 쌍 x_p 와 x_q 사이의 상태 천이에 대한 모델 정합을 구현하는 제어 입력 스트링 집합을 $A_{p,q}C(AU(\sigma))^*$ 라 하자. 제어기 C의 부하를 최소화시키는 제어 입력 스트링은 $A_{p,q}$ 원소 중 (i) 스위칭 신호를 가장 적게 포함하면서 (ii) 길이가 제일 짧은 것이다.

예를 들어 펌프프 시스템을 x_p 에서 x_q 로 이동시키는 교정 제어기를 설계해야 한다고 하자. 즉 $K_{p,q}(\Sigma_r)=1$ 이다. 정리 1의 조건식이 만족된다면 $K_{p,q}(\Sigma)=1$ 이며 정의 3에 의해

이루어진 그림 1의 페루프 시스템의 안정 상태 동작을 Σ^1 의 동작과 일치시키는 C를 설계하는 것이다. 그림 3에서 알 수 있듯이 개별 부-머신이 묘사하는 누산기의 동작이 기준 모델 Σ^1 의 동작을 구현할 수 없음을 자명하다. 기술의 편의상 그림 3에서 모든 부-머신의 상태 천이 함수 f_i 가 stable recursion 함수 s_i 와 동일하다고 설정하였다. 즉 본 문제는 부-머신 세 개의 안정 상태 동작이 그림 3

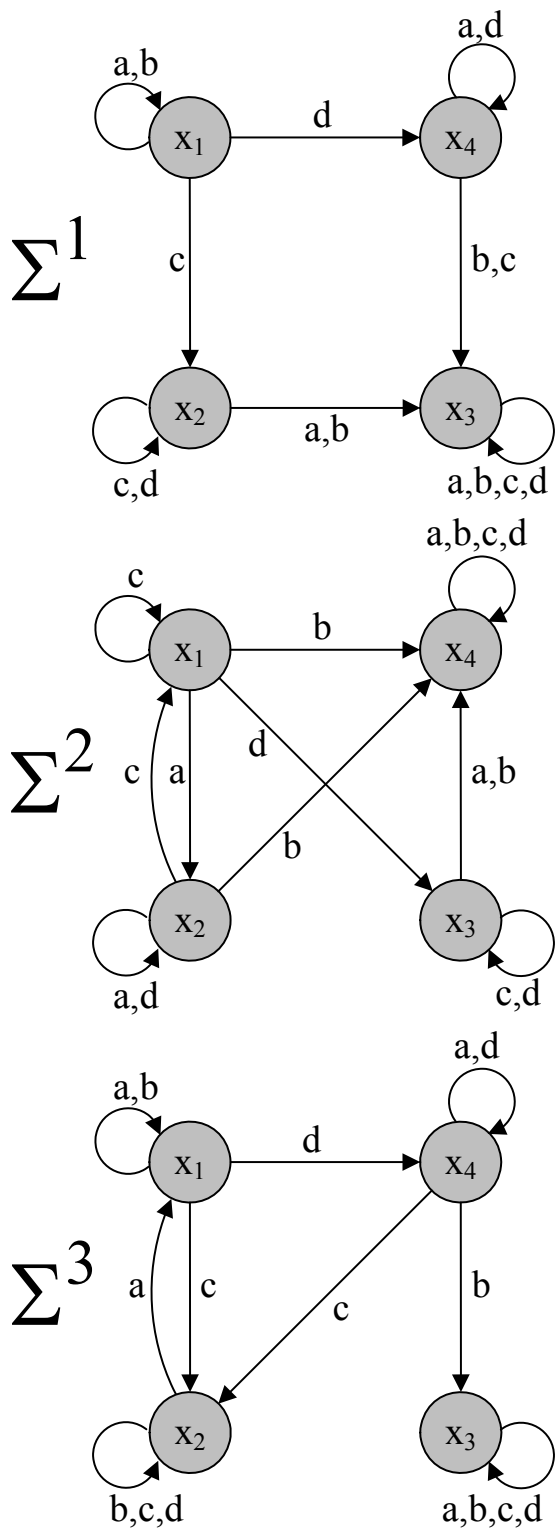


그림 3. $\Sigma^1, \Sigma^2, \Sigma^3$ 을 가진 스위칭 비동기 머신 Σ
 Fig. 3. Switched asynchronous machine Σ with $\Sigma^1, \Sigma^2,$ and Σ^3

과 같고 시퀀스 (1,2,3)을 가지는 모든 스위칭 머신에 적용될 수 있다.

C의 존재 조건을 규명하기 위해서 먼저 각 부-머신의 skeleton 행렬을 계산하면 아래와 같다(skeleton 행렬의 자세한 계산 과정은 [1, 2] 참조).

$$\begin{aligned}
 K(\Sigma^1) &= \begin{bmatrix} 1111 \\ 0110 \\ 0010 \\ 0011 \end{bmatrix} & K(\Sigma^2) &= \begin{bmatrix} 1111 \\ 1111 \\ 0011 \\ 0001 \end{bmatrix} \\
 K(\Sigma^3) &= \begin{bmatrix} 1111 \\ 1111 \\ 0010 \\ 1111 \end{bmatrix}
 \end{aligned} \tag{15}$$

다음으로 정의 2에서 도입한 스위칭을 고려한 skeleton 행렬 $W(\Sigma^2), W(\Sigma^3)$ 를 구한다. 그림 3의 Σ^1 과 Σ^2 에서 각 상태와 안정 조합을 이루는 입력 character들을 관찰하자. 상태 x_2, x_3, x_4 에서는 두 부-머신에서 모두 해당 상태와 안정 조합을 이루는 입력이 존재하지만, 상태 x_1 에서는 그러한 입력이 없다. 즉 $U_1(x_1) \cap U_2(x_1) = \emptyset$ 이다. 그렇다면 정의 2의 2)에 따라서 $U_1(x_1)$ 과 $T_2(x_1)$ 사이의 포함 관계를 살펴야 한다. 그림 3에서 $U_1(x_1) = \{a, b\}$ 이고 $T_2(x_1) = \{a, b, d\}$ 이다. $U_1(x_1) \not\subseteq T_2(x_1)$ 이고 $T_2(x_1)/U_1(x_1) = \{d\}$ 이므로 Σ^2 의 x_1 에서 d에 의해서만 도달 가능한 상태가 있는지 검사한다. 그림 3에서 볼 수 있듯이 x_3 이 그러한 상태이므로 $W(\Sigma^2)$ 의 첫번째 행 벡터는 [0 1 0 1]으로 유도된다. $K(\Sigma^2)$ 과 비교하여 (1,1)과 (1,3)번째 원소 값이 1에서 0으로 바뀌었다.

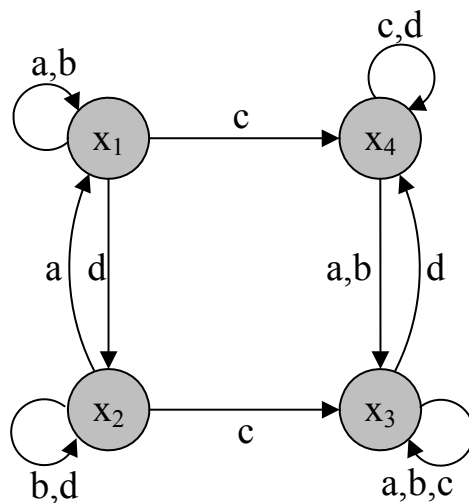


그림 4. 기준 모델 Σ
 Fig. 4. Reference model Σ .

이상과 같은 방법으로 $W(\Sigma^2)$ 와 $W(\Sigma^3)$ 을 구하면

$$\begin{aligned}
 W(\Sigma^2) &= \begin{bmatrix} 0101 \\ 1111 \\ 0011 \\ 0001 \end{bmatrix} & W(\Sigma^3) &= \begin{bmatrix} 0110 \\ 1111 \\ 0010 \\ 1111 \end{bmatrix}
 \end{aligned} \tag{16}$$

이다. 정의 3을 적용하여 $K(\Sigma)$ 를 계산하면 다음과 같다.

$$\begin{aligned}
 K(\Sigma) &= W(\Sigma^1) + W(\Sigma^1)W(\Sigma^2) \\
 &\quad + W(\Sigma^1)W(\Sigma^2)W(\Sigma^3) \\
 &= \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}
 \end{aligned}
 \tag{17}$$

또 그림 4로부터 모델 Σ_r 의 $K(\Sigma_r)$ 을 유도하면

$$K(\Sigma_r) = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}
 \tag{18}$$

이다. $K(\Sigma) \geq K(\Sigma_r)$ 이므로 정리 1에 의해서 모델 정합 교정 제어기 C가 존재한다.

C의 설계는 기존 연구 [1-3]의 틀에서 스위칭 신호를 제어 입력 중간에 생성하는 함수를 추가하여 완성한다. Σ 가 현재 부머신 Σ^1 의 상태 x_2 에 있을 때 Σ 를 x_1 로 보내는 모델 정합 문제를 해결하는 제어기 설계를 예를 들어 설명한다. 그림 4를 보면 $s_1(x_2, a) = x_1$ 이므로 Σ_r 은 안정 상태 x_2 에서 입력 a 가 들어오면 x_1 로 천이한다. 하지만 그림 3에서 Σ^1 은 x_2 에서 a 에 반응하여 x_3 로 천이하는 모델 부정합을 가진다. 이를 해결하는 제어기 C를 설계하기 위해 먼저 $W(\Sigma^1) (=K(\Sigma^1))$ 의 (2,1)번째 원소 값을 확인한다. $W_{2,1}(\Sigma^1) = 0$ 이므로 Σ 가 스위칭을 하지 않고 Σ^1 에 머물러 있는 한 모델 정합을 이루지 못한다. 다음으로 두번째 행렬 곱의 (2,1)번째 원소 값을 체크하면 $[W(\Sigma^1)W(\Sigma^2)]_{2,1} = 1$ 이다. 따라서 Σ^1 에서 Σ^2 로 스위칭을 한 후에 모델 정합을 이룰 수 있다. 또 그림 3에서 Σ^2 가 $s_2(x_2, c) = x_1$ 의 상태 천이를 가지기 때문에 모델 정합을 위한 도달가능성이 존재함을 알 수 있다.

그림 5는 Σ 를 x_2 에서 x_1 까지 이동시키는 C의 제어 모듈이다. $\{z_0, \dots, z_3\}$ 은 C가 가지는 상태 집합이며, C의 입력 집합은 $XU(m) \times A$ 이다. 여기서 X는 현재 작동 중인 부머신에서 나오는 상태 피드백을 말하며 m은 Σ 가 스위칭을 한 후 그림 1의 디멀티플렉서가 내보내는 스위칭 확인 신호이다. 또 A는 그림 1의 외부 입력 v의 집합을 뜻한다. C의 출력 집합은 $AU(m)$ 이다. A는 C가 생성하는 제어 입력이고(그림 1의 u), 'σ'는 앞에서 정의했듯이 스위칭을 명령하는 신호이다. σ가 생성되면 그림 1의 스위칭 시퀀스에서 저장된 순열에 따라 Σ 가 다음 부머신으로 스위칭한다.

그림 5에서 C는 초기 상태 z_0 에 있을 때 Σ 가 Σ^1 의 x_2 와 안정 조합을 이루면 다음 상태 z_1 로 천이한다. 그림 3의 Σ^1 에서 x_2 의 안정 조합은 $(x_2, c), (x_2, d)$ 이므로 C는 z_0, z_1 에서 그림 5와 같은 상태 천이를 가진다. z_1 에서 C는 외부 입력의 변화를 관측하면서 모델 정합 교정 제어를 실행할지 여부를 결정한다. z_0 과 z_1 에서 C는 입력 변화만 볼 뿐 실제적인 교정 제어를 실행하지는 않기 때문에 외부 입력 v를 그대로 제어 입력 u로 내보낸다.

C가 z_1 에 있을 때 외부 입력이 a로 바뀌는 순간 교정 동작이 실행된다. (제어 과정 없이 a를 제어 입력으로 전달하면 모델 부정합이 발생한다.) C는 먼저 z_1 에서 z_2 로 상태 천이한 다음 a 대신 스위칭 신호 σ를 생성한다. 상태 피드백에서 m, 즉 스위칭이 완료되었다는 신호가 나오면 C는 다시 다음 상태 z_3 로 이동한다. 이때 스위칭 머신은 Σ^2 의 x_2 에 있다. 그림 3에서 볼 수 있듯이 $s_2(x_2, c) = x_1$ 이다. 따라서 C는 제어 입력으로 c를 생성하며, 디멀티플렉서를 거쳐 이 입력은 Σ^2 에 전달된다. 마지막으로 Σ^2 가 목적 상태 x_1 로 천이하고 상태 피드백도 x_1 로 변하면 C는 초기 상태 z_0 로 복귀한다.

이상의 동작은 클럭 없이 매우 빠른 속도로 수행되므로 외부 사용자가 보면 페루프 시스템은 안정 상태 x_2 에서 입력 a를 받아 다음 상태 x_1 로 즉시 천이하는 것처럼 관측된다. 다른 모델 부정합에 대해서도 상기한 방법과 유사하게 교정 제어기를 설계할 수 있다.

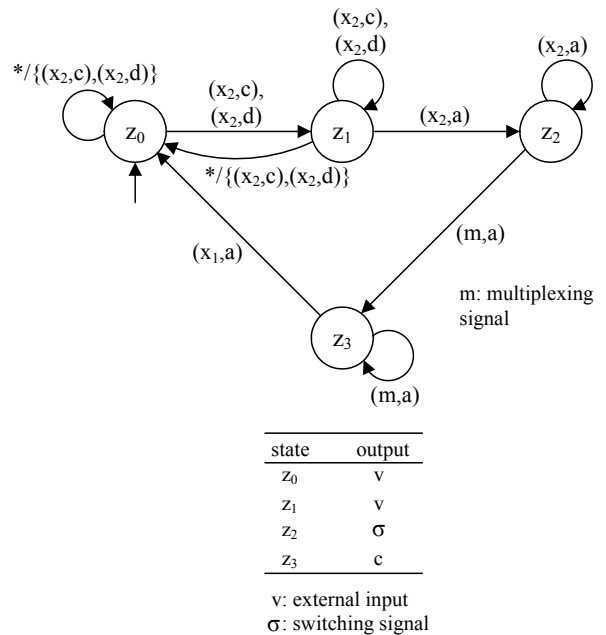


그림 5. $x_2 \rightarrow x_1$ 을 구현하는 C의 제어 모듈
Fig. 5. Control module of C that implements $x_2 \rightarrow x_1$

5. 결론 및 향후 연구

본 논문에서는 스위칭 비동기 순차 머신을 정의하고 스위칭 비동기 머신에 대한 모델 정합을 이루는 교정 제어기의 존재조건을 규명하였다. 또한 사례 연구를 통해 제어 입력과 스위칭 신호를 생성하는 교정 제어기의 설계 과정을 예시하였다. 본 논문에서 도입한 스위칭 비동기 순차 머신은 향후 유사한 작업을 수행하는 여러 개의 모듈로 구성된 하드웨어 여유도를 지닌 시스템의 고장 극복 제어에 적용될 수 있다. 고장이 일어나지 않는 정상 시스템을 기준 모델로 하면 이번 연구에서 해결한 모델 정합 문제의 틀에서 고장 극복 제어기를 꾸밀 수 있다. 스위칭 시퀀스에 대한 제한 조건을 완화하는 문제 등도 추후 연구에서 다룰 예정이다.

References

- [1] J. Peng and J. Hammer, "Bursts and output feedback control of non-deterministic asynchronous sequential machines," *European Journal of Control*, vol. 18, no. 3, pp. 286-300, 2012.
- [2] X. Xu and Y. Hong, "Matrix approach to model matching of asynchronous sequential machines," *IEEE Transactions on Automatic Control*, vol. 58, no. 11, pp. 2974-2979, 2013.
- [3] J.-M. Yang, "Fault-tolerant control of a class of asynchronous sequential machines with permanent faults,"

Automatica, vol. 50, no. 3, pp. 989-993, 2014.

[4] S. W. Kwak and J.-M. Yang, "Fault diagnosis and tolerance for asynchronous counters with critical races caused by total ionizing dose in space," *Journal of Korean Institute of Intelligent Systems*, vol. 22, no. 1, pp. 49-55, 2012.

[5] J.-M. Yang and S. W. Kwak, "Output feedback control of asynchronous sequential machines with disturbance inputs," *Information Sciences*, vol. 259, pp. 87-99, 2014.

[6] Z. Sun and S. S. Ge, *Switched Linear Systems: Control and Design*, London: Springer-Verlag, 2006.

[7] B. J. Choi and P. P. Wang, "A Study on the design of a biologizing control system," *Journal of Korean Institute of Intelligent Systems*, vol. 14, no. 5, pp. 630-634, 2004.

[8] H. Li and Y. Wang, "Consistent stabilizability of switched boolean networks," *Neural Networks*, vol. 46, pp. 183-189, 2013.

[9] L. Sterpone and M. Violante, "Analysis of the robustness of the TMR-architecture in SRAM-based FPGAs," *IEEE Transactions on Nuclear Science*, vol. 53, no. 5, pp. 1545-1549, 2005.

[10] A. Saadatpour, I. Albert, and R. Albert, "Attractor analysis of asynchronous Boolean models of signal transduction networks," *Journal of Theoretical Biology*, vol. 266, no. 4, pp. 641-656, 2010.

[11] Z. Kohavi and N. K. Jha, *Switching and Finite Automata*

Theory, 3rd ed., Cambridge UK: Cambridge University Press, 2010.

[12] K.-S. Chong, B.-H. Gwee and J. S. Chang, "Design of several asynchronous-logic macrocells for a low-voltage micropower cell library," *IET Circuits, Devices and Systems*, vol. 1, no. 2, pp. 161-169, 2007.

저 자 소 개



양정민(Jung-Min Yang)

1993년 : 한국과학기술원 전기및전자공학과(공학사)

1995년 : 한국과학기술원 전기및전자공학과(공학석사)

1999년 : 한국과학기술원 전기및전자공학과(공학박사)

2013년~현재 : 경북대학교 전자공학부 부교수

관심분야 : 비동기 머신 교정 제어, 실시간 시스템 고장 진단 및 극복, 불리언 제어 네트워크

E-mail : jmyang@ee.knu.ac.kr