

論文96-33B-12-10

다단계 퍼지추론 시스템의 퍼지 페트리네트 모델링과 근사추론

(Multistage Fuzzy Production Systems Modeling and Approximate Reasoning Based on Fuzzy Petri Nets)

全命根 *

(Chun Myung Geun)

요 약

본 논문에서는 연쇄적인 구조를 갖는 규칙과 이로 인하여 다단계 근사추론 과정을 필요로 하는 일반적인 형태의 퍼지 추론시스템에 대한 퍼지 페트리네트 모델링과 근사추론에 대하여 다루고자 한다. 퍼지 페트리네트로 모델링된 퍼지 추론시스템을 몇개의 행렬로 표현하고 이를 이용하여 퍼지페트리네트의 상태방정식을 유도한다. 퍼지 추론시스템의 근사추론은 모델링된 퍼지 페트리 네트의 동적 특성과 대응관계가 있으므로 구해진 상태방정식으로 부터 다단계 근사추론을 쉽게 구현하는 방법을 제시한다.

Abstract

In this work, a fuzzy Petri net model for modeling a general form of fuzzy production system which consists of chaining fuzzy production rules and so requires multistage reasoning process is presented. For the obtained fuzzy Petri net model, the net will be transformed into some matrices, and also be systematically led to an algebraic form of a state equation. Since it is found that the approximate reasoning process in fuzzy systems corresponds to the dynamic behavior of the fuzzy Petri net, it is further shown that the multistage reasoning process can be carried out by executing the state equation.

I. 서 론

페트리네트는 Carl Adam Petri에 의해 1962에 통신프로토콜의 오토마타를 연구하기 위해 고안되었는데, 모델링된 시스템의 동적상태 및 구조를 그림으로 쉽게 기술할 수 있고, 비동기적이고 분산적이고 비결정적인 시스템을 기술하는데 유용한 관계로 생산시스템, 순차 제어, 이산 사건 시스템 및 통신 프로토콜을 분석하고

설계하는데 널리 쓰이고 있다^[1].

한편 논리시스템을 기술하는데 있어서, 페트리 네트는 지식베이스를 이루는 규칙들간의 상호 관계를 쉽게 기술할 수 있을 뿐만 아니라, 체계적인 추론과정을 보여 줄 수 있으므로 지식 표현의 모델로도 응용되고 있다. 그러나 페트리네트는 사람의 지식이나 실제적인 시스템을 기술하는데 필요한 애매함이나 불확실함을 표현하는데 부족한 구조적 결함이 있었다.

* 正會員, 忠北大學校 制御計測工學科
(Chungbuk National University Department of Control and Instrumentation Engineering)

※ 본 연구는 충북대학교 국책 자유공모과제(과제번호: 96F-13)의 지원으로 수행된 것임.

接受日字:1996年7月12日, 수정완료일:1996年11月21日

Lipp^[2]은 이러한 구조적 결함을 보강하고 복잡한 산업용 시스템의 모델링에 응용하고자 페트리 네트에 퍼지논리의 개념을 도입하여 퍼지 페트리네트를 제안하였다. Cao와 Sanderson^[3]은 센서의 고장에 대응하여 고장을 복구하는 데 이를 응용하였으며, Scarpelli와 Gomide^[4]는 근사추론에 응용하여 생산공정의 모

델링에 유용하게 사용될 수 있음을 보였다.

한편 Looney¹⁵⁾는 논리 시스템에 대한 퍼지 페트리네트의 동적특성을 연구하였으며, Chen¹⁶⁾ 등은 퍼지추론시스템의 퍼지페트리네트 모델링 방법과 모델링된 네트로 부터 규칙의 일관성을 점검하고 도달 가능 집합(reachability sets) 기법을 이용한 추론 방법을 제시하였다. 그러나, 위의 방법들은 reachability tree에 바탕을 둔 알고리즘적인 탐색 기법을 사용한 관계로 여러개의 입력조건 들이 동시에 주어졌을 경우에 이를 병렬로 처리할 수 없는 단점이 있다.

반면에, Chun^{17), 18)}은 퍼지페트리네트의 행렬에 의한 표현 방법과 이를 이용한 지식표현 방법 및 추론알고리즘을 제안함으로써 병렬로 여러개의 규칙을 동시에 처리하는 병렬추론알고리즘¹⁹⁾을 제안하였다. 그렇지만 위의 논문에서는 0과 1사이의 값을 갖는 퍼지 진리값에 대해 다루었고, 전문가 지식표현에 있어서 필요한 연쇄구조를 갖는 규칙의 모델링이나 다단계 추론방법에 대해서는 다루지 않았으므로 실제 적용에는 한계가 있다.

기존의 단순구조 퍼지 추론시스템의 추론규칙은 전진부에 오는 입력변수와 후진부에 오는 출력변수를 구분하여 이들이 서로 겹치지 않게 되어 있다. 하나 인간의 사고에 있어서는 "If A이면 B이다"라는 규칙과 "If B이면 C이다"라는 규칙이 있을 경우 A라는 사실이 입력되면 B라는 사실이 유추되고 이것이 다시 두번째 규칙에 입력되어 C라는 사실을 유추해 내는 삼단논법에 의한 추론이 일반적이다. 위와 같이 어떤 추론 규칙의 출력변수가 다른 추론 규칙의 입력변수로 사용될때 두개의 규칙이 연쇄구조를 갖는 규칙(chaining rule)이라 한다^{110), 111)}.

추론의 입장에서는 위와 같이 연쇄구조를 갖는 경우에 하나의 사실이 관측되었을때, 중간 추론 결과들이 다시 다른 규칙의 입력이 되므로 한번의 추론만으로 최종추론 결과를 얻을 수 없고 여러단계의 추론 과정을 거치는 다단계 추론 과정을 요구하게 된다. Bugarin 등¹¹⁰⁾은 이러한 연쇄구조를 갖는 퍼지 추론시스템의 퍼지 페트리네트 모델링 및 추론에 대해 다루었다. 그러나 연쇄 구조를 갖는 퍼지 변수를 추출해 내는 체계적인 방법등이 제시되지 않았으며 연쇄 구조를 갖는 규칙에 부가적으로 트랜지션을 새로히 할당하는 관계로 모델링의 효율이 저하되며, 추론에 있어서는 페트리네트의 reachability tree에 바탕을 둔 알고리즘적

인 탐색 기법을 사용한 관계로 여러개의 입력조건 들이 동시에 주어졌을 경우에 이를 병렬로 처리할 수 없는 단점이 있다.

따라서, 본 논문은 연쇄구조를 갖는 퍼지 추론시스템의 퍼지 페트리네트 모델링 및 추론방법을 행렬에 기초한 방법을 사용함으로써 행렬의 성질을 이용하여 연쇄 구조를 갖는 퍼지 변수를 추출해 내는 체계적 방법을 제시하여 인위적으로 트랜지션을 추가하는 것을 피하고 간단한 형태의 대수식으로 표현되는 다단계 병렬 추론 알고리즘을 제시하고자 한다.

본 논문의 구성은 다음과 같다. 2장에서는 퍼지 페트리네트의 정의 및 뒤에서 사용 될 여러가지 중요한 정의들을 기술하며, 3장에서는 연쇄구조를 갖는 퍼지 추론시스템을 정의하고 이를 퍼지페트리네트에 의해 모델링 하는 방법과 상태방정식으로 표현되는 다단계 근사추론 알고리즘을 제시한다. 이어서 4장에서 결론을 기술한다.

II. 퍼지 페트리네트와 기본성질

1. 페트리네트 용어 및 주요성질

페트리네트, C 는 다섯개의 구성요소로 이루어져 있다. $C = (P, T, a, \beta, v)$, $P = \{p_1, p_2, \dots, p_n\}$ ($n \geq 1$)는 플레이스의 집합이며, $T = \{t_1, t_2, \dots, t_m\}$ ($m \geq 1$)는 트랜지션의 집합, 그리고 $a \subset (P \times T)$ 와 $\beta \subset (T \times P)$ 는 방향성아크(directed arcs)의 집합이다. v 는 마킹(marking)으로 ($v: P \rightarrow Q, Q = \{0, 1, 2, \dots\}$) 각각의 플레이스에 토큰(token)을 할당한다¹¹¹⁾. $v(p_i) = v_i$ 를 플레이스 p_i 에 있는 토큰의 갯수를 나타낸다고 하면, $v = (v_1, v_2, \dots, v_n)^T$ 로 표현된다.

페트리네트는 두 종류의 마디를 갖는 방향성 그래프(bipartite directed graph)로 나타낼 수 있다. 그림 1은 페트리네트 그래프의 한 예로 원(○)은 플레이스를 막대기(□)는 트랜지션을 나타낸다. 그리고 플레이스 내의 점(·)은 토큰을 나타낸다. 방향성 아크는 플레이스와 트랜지션의 관계를 나타낸다.

트랜지션 t_i 의 입력 플레이스 $I(t_i)$ 와 출력 플레이스 $O(t_i)$ 는 다음과 같이 정의 된다.

$$I(t_i) = \{p | (p, t_i) \in a\} \quad (1)$$

$$O(t_i) = \{p | (t_i, p) \in \beta\}$$

페트리네트 C 는 (P, T, D^+, D^-, v) 와 같이 대수적인 방법으로도 표현된다¹¹¹⁾. $D^+ = (d_{ij}^+)$ 와 $D^- = (d_{ij}^-)$ 는 출력 연결행렬(output incidence matrix)와 입력 연결행렬 (input incidence matrix)을 나타내는데, d_{ij}^+ 은 트랜지션 t_j 로 부터 그것의 출력 플레이스 중 하나인 p_i 로의 아크의 갯수를 나타내며, d_{ij}^- 는 플레이스 p_i 로 부터 트랜지션 t_j 로의 아크의 갯수를 나타낸다(그림 1의 (c) 참조).

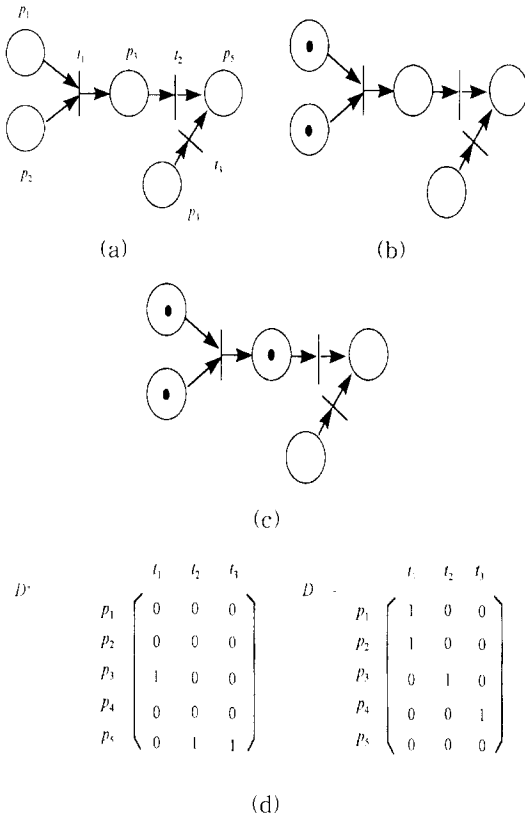


그림 1. 페트리 네트의 예
(a) 추론시스템의 페트리네트 모델 (b) C_1 과 C_2 가 만족 되었을 때 (c) t_1 이 점화 되었을 때 (d) 연결행렬

Fig. 1. Example of a Petri Net.
(a) A Petri net model of a production system, (b) Case when C_1 and C_2 are satisfied, (c) The state after t_1 fires, (d) Incidence matrix model

페트리네트로 규칙을 표현할 때, 보통 플레이스로 조건을 트랜지션으로 규칙을 나타내며, 플레이스에 토큰이 있을때, 그 플레이스가 나타내는 조건이 만족되었음을 나타낸다. 따라서 어떤 트랜지션 t_i 의 입력 플레이

스가 모두 토큰을 가지고 있을때, t_i 가 인에이블(enable)하다 라고 하며, 인에이블한 트랜지션은 점화(firing) 될 수 있다. 인에이블은 트랜지션이 나타내는 규칙의 전제조건이 모두 만족된 상황을 나타낸다.

앞으로, $I^{n \times m}$ 을 $(n \times m)$ 차원의 이진행렬을 나타내고 I_n 을 $(n \times n)$ 차원의 항등행렬, 그리고 $R_1^{n \times m}$ 을 각각의 요소들이 0과 1사이의 값을 갖는 $(n \times m)$ 차원의 행렬을 표기하기로 하자.

위에서 설명된 개념들을 정리하고자 다음의 예를 제시한다.

예제 1 : 다음과 같이 다섯개의 조건 $C_i(i=1,2,\dots,5)$ 와 세개의 규칙 $R_i(i=1,2,3)$ 로 주어지는 추론 시스템을 생각해 보자.

R_1 : If C_1 and C_2 are satisfied, then C_3 will be satisfied.

R_2 : If C_3 is satisfied, then C_5 will be satisfied.

R_3 : If C_4 is satisfied, then C_5 will be satisfied.

위와 같이 주어진 추론 시스템에 대한 페트리 네트 그래프 모델이 그림 1에 주어져 있다. 그림에서 p_i 과 t_i 는 각각 C_i 와 R_i 에 해당되며, 구해진 페트리 네트에 대한 연결 행렬은 그림 1.(d)에 나타내었다. 그림 1.(b)는 C_1 과 C_2 가 모두 만족 되었을 때를 나타낸다. 그러면, 규칙 R_1 이 만족되고 C_3 가 참이 된다. 이때의 상황이 그림 1.(c)에 표시되어 있다. 이와같이, 토큰의 흐름으로 이러한 상태 변화를 쉽게 나타낼 수 있으며, 연결행렬을 이용하면 페트리 네트가 쉽게 표현할 수 있음을 그림 1.(d)를 통해서 알 수 있다.

2. 퍼지 페트리 네트

정의 2.1 : 퍼지 페트리 네트 FPN는 여섯개의 구성 요소로 이루어 진다. $FPN = (P, T, \alpha, \beta, W, \Omega)$ 여기서

$P = \{p_1, \dots, p_n\} (n \geq 1)$ 는 플레이스의 집합이며,
 $T = \{t_1, \dots, t_m\} (m \geq 1)$ 는 트랜지션의 집합,
 $\alpha \subset \{P \times T\}$ 와 $\beta \subset \{T \times P\}$ 는 방향성 아크의 집합이며,

$W: T \rightarrow [0, 1]^m$ 는 가중치 함수,
 $\Omega: P \rightarrow [0, 1]^n$ 는 $[0, 1]$ 사이의 값을 갖는 토큰을 할당하는 마킹이다.

$W(t_i) = w_i$ 를 트랜지션 t_i 에 관계되는 가중치라 하면, $W = (w_1, w_2, \dots, w_m)^T$ 가 되며, $\Omega(p_i) = \omega_i$ 를 p_i 의 토큰이 갖는 값으로 하면, $\Omega = (\omega_1, \omega_2, \dots, \omega_n)^T$ 로 표현할 수 있다.

위의 퍼지페트리 네트는 다음과 같은 점에서 보통의 페트리 네트와 구별된다.

(1) 트랜지션이 $W = (w_1, w_2, \dots, w_m)^T$ 에 의해 0과 1사이의 가중치를 갖는다.

(2) 토큰이 0과 1사이의 값을 갖는다. 만약 $W = [1, 1, \dots, 1]^T$ 이고 $\Omega : P \rightarrow (0, 1)^n$ 이면, 퍼지 페트리네트는 보통의 페트리 네트가 된다.

이제 퍼지페트리 네트의 행렬에 의한 표현 방법을 생각해 보자. 입력 연결 행렬과 출력 연결행렬은 앞서의 페트리 네트와 같이 정의 된다.

정의 2.2 : 퍼지 페트리 네트 $FPN = (P, T, \alpha, \beta, W, \Omega)$ 에 대하여 D_w^+ 는 다음과 같이 정의하자.

$$D_w^+ \equiv D^+ \cdot \text{Diag}(W) \quad (2)$$

여기서, $\text{Diag}(W)$ 는 벡터 W 로 이루어진 $(m \times m)$ 차원의 대각행렬(diagonal matrix)이다.

3. 퍼지페트리 네트와 퍼지수

퍼지 논리를 이용하여 실제의 문제를 풀고자 할 때 사용되는 퍼지수(fuzzy number)를 어떻게 선택하느냐에 따라 계산 효율 및 기술의 간결성이 좌우 된다. 여기서는 LR형식의 평평한 퍼지수(LR-type flat fuzzy number, ^[12])를 이용하여 퍼지수를 표현하고자 한다. 주어진 LR형식의 평평한 퍼지수 $F = (m_1, m_2, \alpha, \beta)_{LR}$ 에 대하여 ω -수준 퍼지 구간(ω level fuzzy interval) ωF 는 다음과 같이 정의 된다.

정의 2.3 : 주어진 LR형식의 평평한 퍼지수 $F = (m_1, m_2, \alpha, \beta)_{LR}$ 와 0과 1사이의 값 ω 에 대하여, ω 수준 퍼지 구간 ωF 는 두개의 파라미터 $a = m_1 - \alpha L^{-1}(\omega)$ 와 $b = m_2 + \beta R^{-1}(\omega)$ 그리고 다음의 소속 함수에 의해 정의된다.

$$\mu_F^\omega(x) = \begin{cases} L(\frac{m_1 - x}{\alpha}) & \text{for } x \leq a, \alpha > 0, \\ \omega & \text{for } a \leq x \leq b, \\ R(\frac{x - m_2}{\beta}) & \text{for } x \geq b, \beta > 0. \end{cases} \quad (3)$$

기호로는 $\omega F = (m_1, m_2, \alpha, \beta)_{LR}^\omega$ 로 나타내기로 하자.

$\omega = 1$ 이면, ωF 는 LR형식의 평평한 퍼지수 F 가 된다. 그림 2는 평평한 퍼지수 F 와 그에 대한 ωF 를 나타내었다.

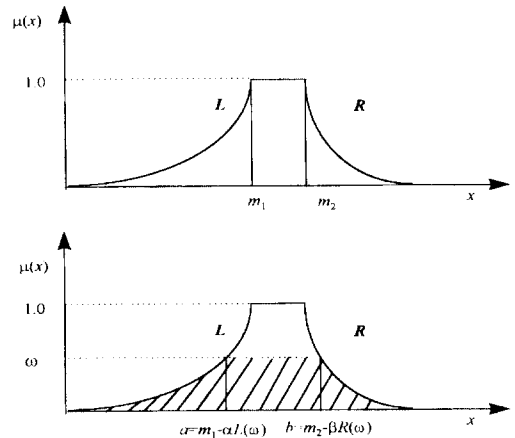


그림 2. LR-퍼지수

(a) LR-형태 평평한 퍼지수 (b) ω -수준 퍼지 구간

Fig. 2. LR-fuzzy number.

(a) LR-type flat fuzzy number, (b) ω -level fuzzy interval

정의 2.4 : x 를 퍼지 언어변수(linguistic fuzzy variable)라 하고 L 과 \tilde{L} 를 전체집합 U 에서 μ_L 와 $\mu_{\tilde{L}}$ 의 소속함수를 갖는 두개의 퍼지 부분집합이라 할 때, 퍼지 명제 " x is L "에 대하여 " x is \tilde{L} "의 사실이 관찰 되었다면 주어진 명제의 퍼지 진리값 $\tau \in [0, 1]$ 는 possibility measure ^[12]에 기초하여 다음과 같이 정의한다.

$$\tau = \sup_u \min\{\mu_{L(u)}, \mu_{\tilde{L}(u)}\} \quad (4)$$

정의 2.5 : 각각의 소속함수가 $\mu_{L_i} = ((m_{1i}), (m_{2i}), \alpha_i, \beta_{\partial_i})$ ($i=1, 2, \dots, l$)로 표시되는 l 개의 퍼지수를 갖는 퍼지변수 x 에 대해 퍼지관계값 r_{ij} ($1 \leq i \leq l, 1 \leq j \leq l$)로 이루어진 퍼지값 관계행렬(relational fuzzy value matrix) $R = (r_{ij}) \in R_1^{l \times l}$ 을 다음과 같이 정의한다.

$$r_{ij} = \sup_u \min\{\mu_{L_i}(u), \mu_{L_j}(u)\} \quad (5)$$

그림 5와 같이 주어진 퍼지변수에 대한 퍼지값 관계행렬은 위의 정의로부터 다음과 같이 구해진다.

$$R = \begin{matrix} & L_1^i & L_2^i & L_3^i & L_4^i & L_5^i & L_6^i & L_7^i \\ L_1^i & 1 & 0.5 & 0 & 0 & 0 & 0 & 0 \\ L_2^i & 0.5 & 1 & 0.5 & 0 & 0 & 0 & 0 \\ L_3^i & 0 & 0.5 & 1 & 0.5 & 0 & 0 & 0 \\ L_4^i & 0 & 0 & 0.5 & 1 & 0.5 & 0 & 0 \\ L_5^i & 0 & 0 & 0 & 0.5 & 1 & 0.5 & 0 \\ L_6^i & 0 & 0 & 0 & 0 & 0.5 & 1 & 0.5 \\ L_7^i & 0 & 0 & 0 & 0 & 0 & 0.5 & 1 \end{matrix}$$

III. 퍼지 페트리네트와 근사추론

1. 퍼지 페트리네트 모델

본 논문에서 다루고자 하는 다단계 퍼지 추론시스템 (multistage fuzzy production system)의 형태는 다음과 같은 것이다.

정의 3.1 : 언어변수 x_i 는 전체집합 U_i ($1 \leq i \leq n$, $1 \leq j \leq l_i$)에서 l_i 개의 소속함수 $\mu_{L_i} = ((m_1)^j, (m_2)^j, \dots, a_j^j, \beta_j^j)_{L_i, K}$ 를 갖는 LR형태의 평평한 퍼지수 L_i^j 로 이루어졌고 X 는 그것의 집합이라 하자.

D 를 단순 퍼지 명제 $d_j^{(i)}$ ($1 \leq i \leq n, 1 \leq j \leq l_i$) 즉, " x_i is L_i^j "로 이루어진 집합이라고 하면, 퍼지 추론 시스템 Ψ 는 다음과 같이 4개의 구성요소로 이루어져 있다.

$$\Psi = (X, \mathcal{A}, C, \Phi)$$

여기서 $\mathcal{A} = \{\delta_i | 1 \leq i \leq m\}$ 는 퍼지 추론 규칙 δ_i 의 집합인데, (i_1, i_2, \dots, i_r) 와 (k_1, k_2, \dots, k_r) 를 $(1, 2, \dots, n)$ 의 subsequence라 하면

$$\delta_i = \text{If } d_{i_1}^{j_1} \text{ and } d_{i_2}^{j_2} \dots \text{ and } d_{i_r}^{j_r} \text{ then } d_{k_1}^{k_1} \text{ and } d_{k_2}^{k_2} \dots \text{ and } d_{k_r}^{k_r}; \tag{6}$$

또는

$$\delta_i = \text{If } d_{i_1}^{j_1} \text{ or } d_{i_2}^{j_2} \dots \text{ or } d_{i_r}^{j_r} \text{ then } d_{k_1}^{k_1} \text{ and } d_{k_2}^{k_2} \dots \text{ and } d_{k_r}^{k_r}; \tag{7}$$

의 형태를 갖는다. $\Phi : D \rightarrow [0, 1]$ 이고 $C : \mathcal{A} \rightarrow [0, 1]^m$. $\Phi(d_j^i) = \tau_j^i$ 를 d_j^i 의 퍼지 진리값으로 표기하면, $\Phi = [\tau_1^1, \dots, \tau_n^1]^T$. 또한 $C(\delta_i) = c_i$ 를 δ_i 에 대한 확신도^[16]라 하면 $C = [c_1, c_2, \dots, c_m]^T$.

여기서, 결론부에 논리 OR로 연결된 퍼지 추론규칙에 대해서는 고려하지 않았다. 왜냐하면, 이는 특정 추론결과를 유추해 내지 않을 뿐 아니라 유추에도 적절

하지 않기 때문이다^[6]. k_i 를 δ_i 의 전건부에 있는 퍼지 명제의 갯수라 하면, 퍼지 추론 규칙이 (7)과 같이 주어진 δ_i 는 $\delta_i = \text{If } d_{i_1}^{j_1} \text{ then } d_{k_1}^{k_1} \text{ and } d_{k_2}^{k_2} \dots \text{ and } d_{k_r}^{k_r}$ 와 같은 k_i 개의 퍼지 추론규칙 $\delta_i^1 \dots \delta_i^{k_i}$ 로 나타낼 수 있다^[6].

위에서 정의된 퍼지 추론시스템을 다단계 퍼지 추론 시스템이라고 한 이유는 기존의 퍼지 추론시스템에서는 각각의 퍼지 추론규칙의 전건부에 오는 입력변수와 후건부에 오는 출력변수를 구분하여 이들이 서로 겹치지 않게 되어 있는 것과 달리 어떤 추론 규칙의 후건부에 있는 출력변수가 다른 추론 규칙의 전건부의 입력변수로 사용될 수 있는 연쇄구조를 갖는 규칙을 사용하고 있기 때문이다.

추론의 입장에서 위와 같이 연쇄구조를 갖는 경우에 하나의 사실이 관측되었을때, 중간 추론 결과들이 다시 다른 규칙의 입력이 되므로 한번의 추론만으로 최종추론 결과를 얻을 수 없고 여러단계의 추론 과정을 거치는 다단계 추론 과정을 거치야만 한다.

위에서 정의된 다단계 퍼지 추론 시스템을 퍼지 페트리네트로 모델링 하는 방법은 다음과 같다.(그림 3 참조).

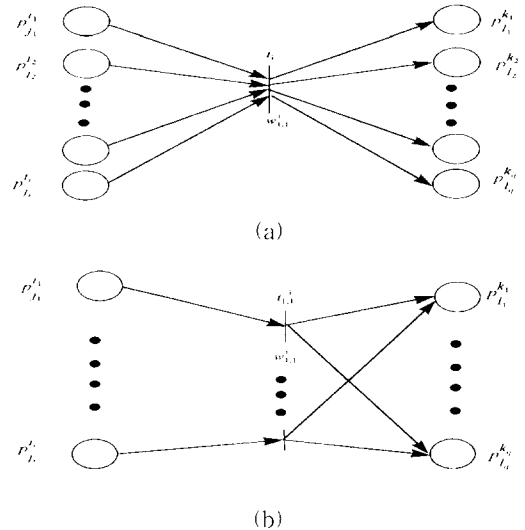


그림 3. 퍼지 페트리네트 모델의 예
 (a) 식 (6)과 같은 형태의 추론규칙 (b) 식 (7)과 같은 형태의 추론규칙
 Fig. 3. Example of a fuzzy Petri net model.
 (a) Case of fuzzy Production rule such as (6) (b) Case of fuzzy Production rule such as (7)

퍼지 페트리네트 모델링 알고리즘

입력 : 퍼지 추론시스템 $\Psi = (X, \Delta, C, \Phi)$

출력 : 퍼지 페트리네트 모델 $FPN = (P, T, \alpha, \beta, W, \Omega)$.

[단계 1.] { Define a set of places P }

for $i \leftarrow 1$ to n do

for $j \leftarrow 1$ to l_i do ;

$p_j^i \leftarrow d_j^i$;

od

[단계 2.] { Define a set of transitions T_i and their weightings to obtain T and W . Let the number of the fuzzy propositions in the left-hand-side of δ_i be k_i }.

for $i \leftarrow 1$ to m do

If δ_i is given as (6)

then $t_i \leftarrow \delta_i, T_i = \{t_i\}$,

and $w_{1,1}^i \leftarrow c_i$;

else

for $j \leftarrow 1$ to k_i do

$t_j^i \leftarrow \delta_j^i, w_j^i \leftarrow c_i$;

od

$T_i \leftarrow \{t_1^i \dots t_{k_i}^i\}$

fi

od

$T = \bigcup_{i=1}^m T_i$ and $W = [w_1^1 \dots w_{k_m}^m]^T$;

[단계 3.] { Define α }

If δ_i is given as (6)

then make an arc from each place associated with the

fuzzy proposition in the left-hand-side of δ_i to the t_i ;

else make an arc from the j th ($1 \leq j \leq k_i$) place associated

with the fuzzy proposition in the left-hand-side of δ_i to t_i ;

fi

[단계 4.] { Define β }

If δ_i is given as (6)

then make an arc from t_i to each place associated with the fuzzy proposition in the right-hand-side of δ_i to the t_i ;

else make an arc from each t_j^i ($1 \leq j \leq k_i$) to the place associated with the fuzzy proposition in the right-hand-side of δ_i to t_i ;

fi

[단계 5.] { Define the marking of the fuzzy Petri net Ω }

$\Omega = \Phi$, that is,

$$\Omega = [\omega_1^1, \omega_2^2, \dots, \omega_{l_1}^1; \dots; \omega_1^m, \omega_2^m, \dots, \omega_{l_m}^m]^T$$

$$\left[\tau_1^1, \tau_2^2, \dots, \tau_{l_1}^1; \dots; \tau_1^m, \tau_2^m, \dots, \tau_{l_m}^m \right]^T$$

where $\omega_j^i \in [0,1]$ is an attribute of a token at p_j^i .

2. 퍼지 페트리네트에서의 추론과정

지금까지의 논의에서는 퍼지 페트리네트 모델에서 사용되는 논리연산자들에 대하여 전혀 제한을 두지 않았다. 이는 퍼지 페트리네트 모델에 일반적인 t-norm과 t-conorm이 사용될 수 있음을 나타낸다^[11]. 그러나 추론을 위해서는 특정한 t-norm과 t-conorm 그리고 추론자를 정해야 한다. 본 논문에서는 'min' 연산자와 'max' 연산자를 t-norm과 t-conorm 대신에 사용하고 여러가지 추론방법 중에서 확신도(certainty factor:CF)를 고려한 Mamdani의 추론 방법을 사용하고자 한다^[13]. 확신도는 각 규칙에 대하여 각각의 믿음도를 0과 1의 숫자로 나타낸 것으로 이 값이 1인 경우에 그 규칙에 대한 믿음값이 참임을 나타낸다.

예제 2: 다음과 같이 각각이 $\mu_{L_1} = ((m_{11}), (m_{21}), \alpha^1,$

$\beta_1^{0.15}$ ($i = 1, 2, \dots, 6$)와 같은 퍼지수를 갖는 퍼지 변수로 구성된 세개의 퍼지 추론규칙으로 이루어진 퍼지 추론시스템에 " x_j 는 \hat{L}_j ($j = 1, 2, 3, 4$)"라는 사실이 주어졌을때 다단계추론과정을 살펴보자.

$$R_1: \text{ If } x_1 \text{ is } L_1^1 \text{ and } x_2 \text{ is } L_1^2 \text{ then } x_5 \text{ is } L_1^5 \\ (CF = c_1)$$

R_2 : If x_3 is L_3^3 and x_4 is L_4^4 then x_5 is L_5^5
($CF = c_2$)

R_3 : If x_5 is L_5^5 then x_6 is L_6^6 ($CF = c_3$)

먼저, R_1 으로 부터 x_5 에 대한 퍼지진리값을 구해보면 $\mu_{L_5^5} = ((m_1)_1^5, (m_2)_1^5, \alpha_1^5, \beta_1^5)_{L_5^5 R_1}^{\tau_1^5}$ 와 같이 표현된다. 여기서 퍼지 진리값은 식 (5)에 따라 $\tau_1^5 = c_1 \times \min\{\tau_1^3, \tau_1^4\}$ 이다.

R_2 으로 부터 x_5 에 대한 퍼지진리값을 구해보면 $\tau_2^5 = c_2 \times \min\{\tau_1^3, \tau_1^4\}$ 이 된다. 따라서 $\mu_{L_5^5} = ((m_1)_3^5, (m_2)_3^5, \alpha_3^5, \beta_3^5)_{L_5^5 R_2}^{\tau_2^5}$ 와 같이 표현된다. 위의 추론 결과로 얻어진 $\mu_{L_5^5}$ 와 $\mu_{L_5^5}$ 이 다시 추론규칙 R_3 의 입력으로 들어가게 된다. 채택된 추론방법에 따르면 x_6 에 대한 최종 추론결론은 $\mu_{L_6^6} = ((m_1)_1^6, (m_2)_1^6, \alpha_1^6, \beta_1^6)_{L_6^6 R_3}^{\tau_1^6}$ 와 같이 기술된다. 여기서 $\tau_1^6 = c_3 \times \tau_2^5$ 는 퍼지 진리값의 정의에 따라 다음과 같이 구해진다.

$$\begin{aligned} \tau_2^5 &= \sup_u [\min(\mu_{L_5^5}, \min(\mu_{L_5^5}, \mu_{L_5^5})] \\ &= \sup [\sup_u \min(\mu_{L_5^5}, \mu_{L_5^5}), \sup_u \min(\mu_{L_5^5}, \mu_{L_5^5})] \\ &= \sup [\min(\tau_1^3, \tau_1^4), \min(\tau_2^3, \tau_2^4)] \end{aligned}$$

윗식에서 알 수 있듯이 연쇄구조를 갖는 퍼지변수에 대한 퍼지 진리값은 해당 퍼지 변수를 이루는 퍼지수에 대한 퍼지 진리값과 (5)식으로 주어지는 퍼지관계값들 중에서 작은 값을 택하여 이들 가운데 최대값을 취하면 된다. 그림 4는 위의 예에 대한 추론과정을 나타낸다. 추론과정이 토큰의 흐름으로 잘 표현 됨을 알 수 있다.

이제, 퍼지페트리네트의 토큰의 흐름 및 그것이 갖는 특성값의 변화를 기술하는 상태 방정식에 대하여 알아 보자. Looney^[15]에 따르면 "페트리네트가 논리시스템을 모델링 할 경우 보통의 발화 규칙과는 달리, 발화되는 트랜지션은 그에 해당되는 모든 출력플레이스에 토큰을 전달하여야 하며, 마찬가지로 플레이스에서는 그것에서 출발하는 모든 아크에에 토큰을 보급해야한다. 따라서 트랜지션의 발화 후에도 그것의 입력 플레이스에 있는 토큰은 소멸하지 않는다" 라고 지적하고 있다.

이제 Looney의 기술을 만족하면서 채용된 논리연산 및 추론을 행하는 상태 방정식을 유도하여 보자.

앞에서 제시된 퍼지 페트리네트 모델링 방법에 따라 주어질 다단계 퍼지 추론 시스템에 대응되는 퍼지 페

트리네트 $FPN = (P, T, \alpha, \beta, W, \Omega)$ 을 얻는다. 먼저, 부록에 제시된 방법에 따라 연쇄 구조를 갖는 퍼지변수에 대응되는 플레이스를 추출해 낸다. 이렇게 하여 입출력 변수에 해당되는 플레이스의 마킹을 Ω^c 라 하고 연쇄구조를 갖는 변수에 대한 것을 Ω^e 라 하여 전체 마킹을 $\Omega = \begin{bmatrix} \Omega^e \\ \Omega^c \end{bmatrix}$ 와 같이 표시한다. 위에서 구해진 연쇄구조를 갖는 퍼지 변수에 대하여 정의 2.5에 나와 있는 퍼지값 관계행렬을 구하고 R 이라 둔 후 식 (2)로 부터 $D_w^+ \in R_1^{n \times m}$ 를 얻는다.

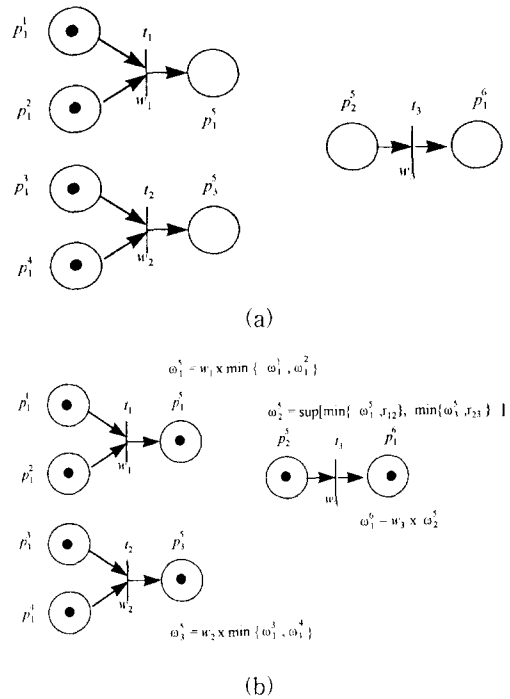


그림 4. 퍼지페트리네트에서의 추론과정

(a) 초기 마킹 (b) 최종 마킹

Fig. 4. Reasoning Process in a fuzzy Petri net.

(a) Initial marking (b) Final marking

추론단계를 k 로 나타내기로 하자. 먼저 $k=0$ 에서의 퍼지 진리값 $\Omega_0 = [\omega_1^{nc}(0), \omega_2^{nc}(0) \dots \omega_n^c(0)]^T$ 를 구한다. 그러면 퍼지 페트리네트의 상태 방정식은 다음과 같이 구해진다.

$$\overline{\Omega}_{k+1} = \begin{bmatrix} \overline{\Omega}_{k+1}^{nc} \\ \dots \\ \overline{\Omega}_{k+1}^c \end{bmatrix} = \Omega_k \vee (D_w^+ \odot ((D^-)^T \otimes \Omega_k)) \quad (8)$$

$$\Omega_{k+1} = \begin{bmatrix} \Omega_{k+1}^{nc} \\ \dots \\ \Omega_{k+1}^c \end{bmatrix} = \begin{bmatrix} \overline{\Omega}_{k+1}^{nc} \\ \dots \\ R \oplus \overline{\Omega}_{k+1}^c \end{bmatrix} \quad (9)$$

위의 (8)식과 (9)식에서, \oplus 와 \odot 은 각각 max-min 합성과 max-product 합성을 나타내고, \otimes 는 다음과 같이 정의된다. 먼저, σ 를 $x \in [0, 1]$ 에 대하여 다음과 같은 성질을 갖는 기호라 하자. $\forall(\sigma, x) = \wedge(\sigma, x) = x$, 그러면 $x_1, x_2 \in [0, 1]$ 에 대한 이진 연산자 \cdot 는 다음과 같이 정의 된다.

$$x_1 \cdot x_2 = \begin{cases} \sigma & \text{if } x_1 = 0 \\ x_1 \cdot x_2 & \text{otherwise.} \end{cases}$$

여기서, $(D^{-1})^T = (((d^-)^T)_{ij})$ 로 두면, \otimes 는 다음과 같이 정의된다.

$$(D^{-1})^T \otimes \Omega_k = (\bigwedge_{1 \leq j \leq n} ((d^-)^T)_{ij} \cdot \omega_j(k))$$

위에서 제안된 식에 따른 추론과정을 다음과 같이 정리 할 수 있다. (8)식은 추론 단계 $k+1$ 에서 앞 단계 k 에서 구한 추론값을 이용하여 연쇄 구조를 고려하지 않은 다음 단계의 임시 추론값 $\bar{\Omega}_{k+1}$ 을 확신도를 고려한 Mamdani의 추론법에 따라 구하는 것을 나타내고 있다. (9)식에서는 (8)식에서 구한 $\bar{\Omega}_{k+1}$ 를 이용하여 연쇄구조를 갖는 퍼지 변수에 대한 진리값을 포함한 최종적인 다음단계의 추론값 Ω_{k+1} 를 구한다. 식에서 알 수 있듯이 Ω_{k+1}^{nc} 에 대해서는 (8)식에서 구한 결과를 그대로 대입하며, Ω_{k+1}^c 에 대해서는 $R \oplus \bar{\Omega}_{k+1}$ 를 통하여 중간 추론결과가 다른 퍼지 추론 규칙의 입력이 되는 경우에 대하여 최종 추론값을 구하게 된다. 위와 같이 각 단계 $k=1, 2, \dots$ 에서 식 (8)과 (9)에 따라 점화 될 수 있는 모든 트랜지션들을 Ω_{k+1} 이 Ω_k 와 같을 때까지 점화 함으로서 최종 추론 결과를 얻게 된다.

예제 3: 다음과 같은 다단계 퍼지 추론시스템에 대하여 생각해 보자

R_1 : If x_1 is L_2^1 then x_3 is L_3^1 ($c_1 = 0.97$)

R_2 : If x_3 is L_1^3 then x_4 is L_3^4 and x_5 is L_3^5 ($c_1 = 0.95$)

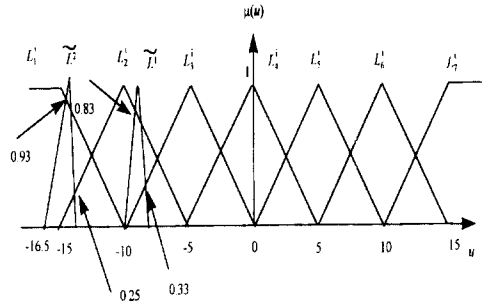
R_3 : If x_1 is L_2^1 then x_3 is L_3^3 ($c_3 = 0.85$)

R_4 : If x_1 is L_3^1 and x_2 is L_1^2 then x_3 is L_3^3 ($c_4 = 0.83$)

R_5 : If x_3 is L_3^3 then x_3 is L_1^1 and x_4 is L_2^1 ($c_5 = 0.96$)

R_6 : If x_3 is L_2^3 then x_5 is L_2^5 ($c_6 = 0.94$)

R_7 : If x_2 is L_2^2 then x_5 is L_2^5 ($c_6 = 0.70$)
여기서 언어변수 $x_i (i=1, 2, \dots, 5)$ 은 그림 5와 같은 LR형태의 평평한 퍼지수를 갖는다. 위와 같은 퍼지 추론시스템에서 x_1 과 x_2 에 그림 5와 같이 L^1, L^2 와 같은 사실이 입력 되었다.



$$\begin{aligned} L_1^i &= (-\infty, -15, 5, 5)_{LR} & L_2^i &= (-10, -10, 5, 5)_{LR} \\ L_3^i &= (-5, -5, 5, 5)_{LR} & L_4^i &= (0, 0, 5, 5)_{LR} \\ L-5^i &= (5, 5, 5, 5)_{LR} & L_6^i &= (10, 10, 5, 5)_{LR} \\ L_7^i &= (15, \infty, 5, 5)_{LR} & & (i = 1, 2, 3, 4, 5) \end{aligned}$$

$$\begin{aligned} \tilde{L}^1 &= (-9, -9, 1, 1)_{LR} \\ \tilde{L}^2 &= (-14.5, -14.5, 2, 1)_{LR} \\ L(u) = R(u) &= -|u| + 1 \end{aligned}$$

그림 5. 사용된 평평한 LR-형식의 퍼지수
Fig. 5. LR-type flat fuzzy numbers.

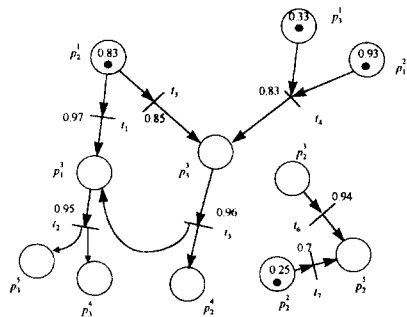


그림 6. 퍼지 페트리네트 모델
Fig. 6. A fuzzy Petri net model.

$$D = \begin{matrix} & t_1 & t_2 & t_3 & t_4 & t_5 & t_6 & t_7 \\ \begin{matrix} p_1^1 \\ p_1^2 \\ p_1^3 \\ p_1^4 \\ p_1^5 \\ p_1^6 \\ p_1^7 \\ p_1^8 \\ p_1^9 \\ p_1^{10} \end{matrix} & \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 \end{pmatrix} & \begin{matrix} t_1 & t_2 & t_3 & t_4 & t_5 & t_6 & t_7 \\ \begin{matrix} p_2^1 \\ p_2^2 \\ p_2^3 \\ p_2^4 \\ p_2^5 \\ p_2^6 \\ p_2^7 \\ p_2^8 \\ p_2^9 \\ p_2^{10} \end{matrix} & \begin{pmatrix} 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} \end{matrix} \quad (a)$$

$$H = \begin{matrix} & p_1^3 & p_2^3 & p_3^3 \\ \begin{matrix} p_1^3 \\ p_2^3 \\ p_3^3 \end{matrix} & \begin{pmatrix} 1 & 0.5 & 0 \\ 0.5 & 1 & 0.5 \\ 0 & 0.5 & 1 \end{pmatrix} \end{matrix}$$

(b)

그림 7. 사용된 행렬

(a) 연결행렬 (b) 퍼지값 관계행렬

Fig. 7. Adopted Matrices.

(a) Incidence Matrices (b) Relational Fuzzy Value Matrix

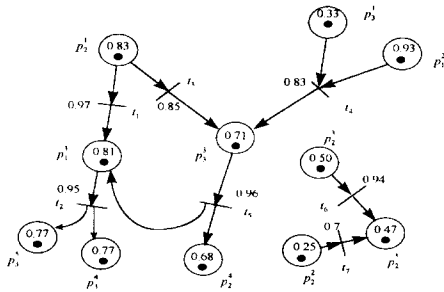


그림 8. 최종 마킹된 퍼지 페트리 네트

Fig. 8. Fuzzy Petri net with a final marking.

	Ω_0	Ω_1	Ω_2
ω_1^1	0.83	0.83	0.83
ω_2^1	0.33	0.33	0.33
ω_3^1	0.93	0.93	0.93
ω_1^2	0.25	0.25	0.25
ω_2^2	0.00	0.00	0.68
ω_3^2	0.00	0.00	0.77
ω_1^3	0.00	0.17	0.47
ω_2^3	0.00	0.00	0.77
ω_3^3	0.00	0.81	0.81
ω_1^4	0.00	0.50	0.50
ω_2^4	0.00	0.71	0.71

그림 9. 각 초론단계에서의 추론값

Fig. 9. Reasoning process at each step k .

제안된 알고리즘에 따라 완성된 퍼지 페트리 네트 모델은 그림 6과 같다. 각 트랜지션의 가중치는 다음과 같으며,

$$W = C = \begin{pmatrix} t_1 & t_2 & t_3 & t_4 & t_5 & t_6 & t_7 \\ 0.97 & 0.95 & 0.85 & 0.83 & 0.96 & 0.94 & 0.70 \end{pmatrix}$$

초기마킹은 다음과 같다.

$$\Omega_0 = \varnothing_0 = \begin{pmatrix} \omega_1^1 & \omega_2^1 & \omega_3^1 & \omega_1^2 & \omega_2^2 & \omega_3^2 & \omega_1^3 & \omega_2^3 & \omega_3^3 \\ 0.83 & 0.33 & 0.93 & 0.25 & 0 & 0 & 0 & 0 & 0 \\ \vdots & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tau$$

그림 7에 위 예제에 대한 연결행렬과 퍼지값 관계행렬을 나타 내었으며, 그림 8 은 최종적인 마킹이 된 퍼지

페트리네트를 나타내고 있다. 각 추론단계 k 에서의 플레이스의 마킹은 그림 9에 나타내었다.

만약 위 퍼지 추론시스템의 목적이 언어변수 x_5 에 대한 추론 결과를 얻는 것이라면, x_5 에 대한 대표값 u_5^0 는 비퍼지화 방법 중 하나인 무게중심법을 이용하여 구할 수 있다^[13].

$$u_5^0 = \frac{\int_{-\infty}^{+\infty} \mu_{L^5} \cdot u_5 \cdot du_5}{\int_{-\infty}^{+\infty} \mu_{L^5} \cdot du_5} = -8.11$$

여기서,

$$\mu_{L^5} = \max_{u_5} \{ \mu_{L_2^5}(u_5), \mu_{L_3^5}(u_5) \};$$

$$\mu_{L_2^5} = (-10, -10, 5, 5), \mu_{L_3^5} = (-5, -5, 5, 5);$$

$$L(u_5) = R(u_5) = -|u_5| + 1.$$

위에서 알 수 있듯이 각 추론 단계의 흐름이 플레이스에 있는 토큰의 흐름으로 알 수 있으며, 그 값의 변화 또한 상태 방정식을 통하여 체계적으로 구할 수 있다.

IV. 결 론

본 논문에서는 퍼지페트리 네트를 이용한 다단계 퍼지 추론 시스템의 모델링 방법과 추론 방법에 대하여 다루었다. 퍼지 페트리 네트는 그래프적인 방법에 기초하므로 규칙과 그것을 이루는 조건들과의 상호 관계가 이해하기 쉽고 토큰의 흐름을 관찰함으로써 추론의 결과를 쉽게 예측할 수도 있다.

기존의 퍼지 추론시스템은 입력, 출력 변수만으로 이루어진 단순한 구조인 반면 본 논문에서 기술된 퍼지 추론 시스템은 인간의 사고과정 중에서 삼단논법 등의 다단계 추론 규칙을 포함하므로 전문가의 지식에 기반을 둔 전문가 제어 시스템이나 의사 결정 시스템에 유용하게 적용 될 수 있으리라 생각된다. 아울러 연쇄 구조를 갖지 않는 단순한 구조의 추론 시스템은 본 논문에서 제시된 방법 중 연쇄구조를 갖는 퍼지 변수가 없는 특수한 경우로 취급하여 동일한 방법으로 문제를 다룰 수 있다.

본 논문에서 제시된 퍼지 페트리네트의 표현방법은 행렬에 기초한 방식으로, 알고리즘 구현이 용이하고 다

단계 추론식도 간단한 산술연산만을 요구하는 병렬 처리가 가능한 구조를 가지고 있으므로 실시간 고장진단 시스템이나 전문가 제어시스템 등에 응용 될 수 있리라 생각된다.

참 고 문 헌

- [1] Murata T, "Petri Nets: Properties, Analysis and Applications" *Proceeding of the IEEE*, vol. 77, p. 541-580, 1989.
- [2] Lipp H P, "The Application of a Fuzzy Petri Net for Controlling Complex Industrial Process" *Proceedings of IFAC Symposium on Fuzzy Information, Knowledge, Representation and Decision Analysis*, p. 471-477, 1983.
- [3] Cao T and Sanderson A C, "Representation and Analysis of Uncertainty Using Fuzzy Petri Nets" *Journal of Intelligent and Fuzzy Systems*, vol. 3, p. 3-19, 1995.
- [4] Scarpelli H and Gomide F, "Fuzzy Reasoning and Fuzzy Petri Nets in Manufacturing Systems Modeling" *Journal of Intelligent and Fuzzy Systems*, vol. 1, p. 225-242, 1993.
- [5] Looney C G, "Fuzzy Petri Nets for Rule-Based Decisionmaking", *IEEE Transactions on System, Man, and Cybernetics* vol. 15, p. 178-183, 1988.
- [6] Chen S M, Ke J S, and Chang J F, "Knowledge Representation Using Fuzzy Petri Nets" *IEEE Transactions on Knowledge and Data Engineering*, Vol. 1, p. 311-319, 1990.
- [7] Chun M G and Bien Z, "Fuzzy Petri Net representation and reasoning methods for rule-based decisionmaking systems" *IEEE Transactions on Fundamentals of Electronics and Computer Science*, E76-A, p. 974-983, 1993.
- [8] Myung-Geun Chun and Zeungnam Bien, "A Knowledge network and inference engine for fuzzy expert systems based on fuzzy Petri net", *Proceedings of the Fifth IFSA World Congress*, p. 203-206, 1993.
- [9] Siler W, Tucker D, and Buckley J, "A Parallel Rule Firing Fuzzy Production Systems with Resolution of Memory Conflicts by Weak Fuzzy Monotonicity, Applied to the Classification of Multiple Objects Characterized by Multiple Uncertain Features", *International Journal of Man-Machines Studies*, vol. 26 p. 321-332, 1987.
- [10] Bugarin A and Barro S, "Fuzzy Reasoning Supported by Petri Nets", *IEEE Transactions on Fuzzy Systems*, vol. 2, p. 135-150, 1994.
- [11] Driankov D and Hellendoorn H, "Chaining of Fuzzy IF-THEN rules in Mamdani-Controllers", *International Conference on Fuzzy Systems*, vol. 1, p. 103-108, 1995.
- [12] Dubois D and Prade H, *Fuzzy Sets and Systems-Theory and Applications* Academic Press, New York, 1980.
- [13] Mizumoto M, "Fuzzy Controls Under Various Fuzzy Reasoning Methods", *Information Science*, vol. 45 p. 129-151, 1988.
- [14] Pedrycz W and Gomide F, "A Generalized Fuzzy Petri Net Model" *IEEE Transactions on Fuzzy Systems*, vol. 2, p. 295-301, 1994.

부 록

모델링된 퍼지 페트리네트 $FPN = (P, T, \alpha, \beta, W, \Omega)$ 에서 각각의 퍼지 변수 x_i 에 대해 모델링된 플레이스의 갯수를 k_i 개라 하자. 즉, $p_i^j (1 \leq i \leq n, 1 \leq j \leq k_i)$. 입력 연결행렬 D^- 와 출력 연결행렬 D^+ 를 다음과 같이 부분행렬로 표시하자.

$$D^- = \begin{pmatrix} d_1^- \\ d_2^- \\ d_3^- \\ \vdots \\ d_n^- \end{pmatrix} \quad D^+ = \begin{pmatrix} d_1^+ \\ d_2^+ \\ d_3^+ \\ \vdots \\ d_n^+ \end{pmatrix} \quad \text{여기서, } d_i^-, d_i^+ \in I^{k_i \times m}$$

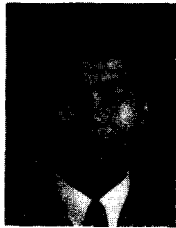
위의 정의된 부분행렬을 이용, 임의의 퍼지 변수 x_k 에 대하여 그것이 연쇄 구조를 갖는 퍼지변수인가를 다음과 같이 판별할 수 있다.

입의 퍼지 변수 x_k 에 대하여 d_k^+ 가 영행렬(null matrix)이고 d_k^- 가 영행렬이 아니면 x_k 는 입력 퍼지 변수이다. 왜냐하면, d_k^+ 가 영행렬이란 의미는 x_k 를 나타내는 플레이스들이 추론 규칙의 후건부에 오는 경우가 없다는 것을 뜻하며 d_k^- 가 영행렬이 아니라는 사실은 플레이스들이 추론규칙의 전건부에 있다는 사실을 나타낸다. 위의 두 사실로부터 x_k 는 입력 퍼지 변수임을 알 수 있다. 위와 유사하게 d_k^- 가 영행렬이고

d_k^+ 가 영행렬이 아니면 x_k 는 출력 퍼지 변수임을 알 수 있다.

따라서 위의 결과로부터 d_k^- 와 d_k^+ 가 모두 영행렬이 아니면 이는 퍼지 변수가 입력 변수일 때도 있고 출력 변수일 때도 있음을 나타내며, 이는 퍼지 변수 x_k 가 퍼지 추론 규칙의 전건부와 후건부 모두에 존재하는 연쇄구조를 가지고 있는 변수임을 나타낸다.

— 저 자 소 개 —



全命根(正會員)

1964년 7월 17일생. 1987년 2월 부산대학교 전자공학과 졸업(공학사). 1989년 2월 KAIST 전기 및 전자공학과 졸업(공학석사). 1993년 8월 KAIST 전기 및 전자공학과 졸업(공학박사). 1993년 8월 ~ 1996년 2월 삼성전자 자동화연구소 선임 연구원. 1996년 3월 ~ 현재 충북대학교 공과대학 제어계측공학과 전임강사. 주관심분야는 실시간 제어 시스템, 지능제어, 페트리 네트 등임