

사이클이 존재하는 퍼지인식도에서의 계층화 알고리즘에 의한 추론메카니즘에 관한 연구*

A Study on the Inference Mechanism of Cyclic Fuzzy Cognitive Map Using a Levelization Algorithm

이건창, ** 조형래***

Kun Chang Lee, Hyung Rae Cho

Abstract

FCM은 비구조적인 (unstructured) 문제영역에서 주어진 문제에 대한 효과적인 추론시 적용될 수 있는 매우 유용한 추론도구이다. 그러나, FCM에 사이클이 존재하면 추론효과가 크게 감소한다. 본 논문에서는 사이클이 있는 FCM을 이용하여 의사결정의 질을 높일 수 있는 추론방법을 제시한다. 이를 위하여 FCM내에 존재하는 사이클을 확인하고 해소하는 알고리즘을 제시한다. 아울러 사이클이 제거된 FCM의 추론의 질을 저하시키는 문제중의 하나인 동기화 문제 (synchronization problem)를 설명하고, 이를 해결하기 위한 방안으로서 FCM 계층화 (levelization) 알고리즘을 제시한다.

* 본 논문은 한국과학재단 핵심전문연구비 (961-0911-059-1)의 일부지원에 의하여 연구되었음.

** 성균관대학교 경영학부 교수 (leekc@yurim.skku.ac.kr)

*** 경상대학교 산업공학과 교수

1. 서론

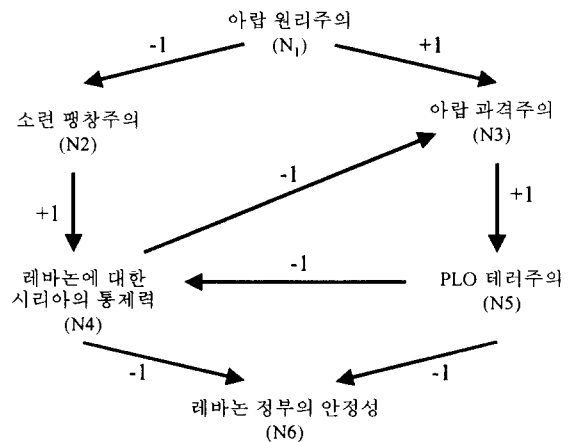
퍼지인식도(Fuzzy Cognitive Map: 이하 FCM이라 약함)는 그 특성상 주어진 문제영역과 관계된 관련요인들간의 인과관계 (causal relationship)를 표현하는데 매우 유용하다 (Lee 1992, 1993; Lee & Kim 1997; Taber 1991). 설명의 편의를 위하여 <그림 1 (a)>에 나타난 예제 FCM을 보자. 이는 Los Angeles Times(1982)에 수록된 Henry Kissinger의 중동평화정책에 관한논문을 바탕으로 작성된 FCM(Kosko, 1986)을 사이클이 존재하는 형태로 변경시킨 것으로서 호의 값이 1 또는 -1인 간략한 형태를 띠고 있다. <그림 1 (a)>을 보면 FCM을 구성하는 노드들은 다음과 같이 세가지 종류로 나눌수 있음을 알 수 있다:

- ①입력노드 : 들어오는 호는 없이 나가는 호만 있는 노드로서 주로 통계가능하거나 측정가능한 사실을 나타낸다.
- ②출력 노드 : 들어오는 호만 있고 나가는 호는 없는 노드로서 주로 최종 관심의 대상이 되며 의사결정에 직접적인 영향을 미치는 사실을 나타낸다.
- ③중간노드 : 들어오는 호 및 나가는 호가 동시에 존재하는 노드로서 입력노드와 출력노드를 연결하는 매개체 역할을 한다.

본 논문에서는 이같이 입력노드 및 출력 노드가 각각 하나 이상 존재하는 경우만을 대상으로 한다. 또한 <그림 1 (b)>는 <그림 1 (a)>에 나타난 FCM을 행렬형태로 나타낸 것으로서, 이를 인과관계 행렬이라 부른다. 이는 FCM에 의한 시뮬레이션이 추론형태에 적합하며, 2장에서 설명할 사이클 확인 및 해소 알고리즘을 개발할 때에 필요한 연관행렬 (adjacency matrix)을 유도하는데에 사용된다.

원래 FCM은 주로 정치 또는 사회과학 분야에서 주어진 환경을 구성하는 요소들간의 인과관계에 관한 지식을 표현하기 위해 개발되었다 (Axelrod, 1976). FCM의 구성요소인 노드 (node)는 주어진 환경을 구성하는 요소(사실, 개념 또는 변수)로 표현되

며 각 노드에 대한 확신도(Certainty factor)는 구간 $[-1, 1]$ 사이의 숫자로 표현된다. 즉 확신도가 1에 가까워 질수록 해당 노드가 나타내는 사실에 대한 강한 긍정을 의미하며, -1에 가까워 질수록 강한 부정을 의미하게 된다. 노드간의 퍼지인과관계는 멤버십이 부여된 호(arc)로 표현된다. 원인노드 N_i 와 결과노드 N_j 를 연결하는 호에 부여된 멤버십 e_{ij} 는 구간 $[-1, 1]$ 사이의 값을 가지며 두 노드 사이의 인과관계의 정도를 나타낸다. $e_{ij} = 0$ 은 두 노드 사이에 아무런 인과관계가 없음을 나타내고, $e_{ij} > 0$ 은 양의 인과관계를 나타낸다. 즉 노드 N_i 에 대한 확신도가 증가(감소)함에 따라 노드 N_j 의 확신도도 같이 증가(감소)하게 된다. 반면에 $e_{ij} < 0$ 은 음의 인과관계를 나타낸다. 즉 노드 N_i 의 확신도가 증가(감소)함에 따라 노드 N_j 의 확신도는 반대로 감소(증가)하게 되는 것이다. 이렇게 각 노드에 대한 확신도 및 퍼지관계에 대한 멤버십을 나타내는 e_{ij} 의 값이 구간 $[-1, 1]$ 사이의 값을 가짐에 따라 기존의 퍼지근사추론 기법(Zadeh, 1979, 1983)에 비해 지식에 대한 표현력 및 추론력이 훨씬 뛰어나다(Lee & Kim 1997).



(a) 중동지역의 국제 역학관계에 관한 FCM

$$E = \begin{matrix} & N_1 & N_2 & N_3 & N_4 & N_5 & N_6 \\ \begin{pmatrix} 0 & -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & -1 & 0 & 0 & -1 \\ 0 & 0 & 0 & -1 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} & N_1 \\ & N_2 \\ & N_3 \\ & N_4 \\ & N_5 \\ & N_6 \end{matrix}$$

(b) 인과관계 행렬

$$A = \begin{matrix} & N_1 & N_2 & N_3 & N_4 & N_5 & N_6 \\ \begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} & N_1 \\ & N_2 \\ & N_3 \\ & N_4 \\ & N_5 \\ & N_6 \end{matrix}$$

(c) 연관행렬

<그림 1> 예제 FCM과 인과관계 행렬, 연관행렬

이러한 특성을 갖는 FCM을 이용하면 해당 문제 영역과 관련된 개념 또는 요인들간의 인과관계 지식을 유도할 수 있고 (Lee & Kim 1997; Lee 1992, 1993), 또한 여러 전문가들의 의견을 서로 상충됨이 없이 결합하여 관련 문제영역에 대한 유용한 인과관계 지식베이스 (causal knowledge base)를 구축할 수가 있다 (Lee 1992, 1993). 이같이 FCM을 의사결정에 활용하고자 하는 사례는 문헌에서 많이 찾아볼 수가 있다 (Axelrod 1976; Eden et al. 1979; Montazemi & Conrath 1986; Styblinski & Meyer 1988; Zhang & Chen 1988; Gotoh 1989; Taber 1991; Lee 1992, 1993; Lee & Kim 1997). 이들 연구 중에서 Kim & Lee (1995)의 연구에서는 FCM내의 퍼지인과관계를 보다 엄밀하게 수학적으로 정의를 내리므로써 FCM을 이용한 시뮬레이션을 위한 새로운 추론원칙을 제시하였다. 그러나, 해당 연구에서도 여전히 FCM내의 사이클 문제와 그에 따른 계층화 문제는 여전히 미해결 과제로 남겨두었다. 따라서,

본 연구에서는

- 1) 사이클이 있는 FCM의 문제점을 해결하는 알고리즘을 제시하고,
- 2) 사이클이 제거된 FCM을 이용한 효과적인 시뮬레이션 추론을 위한 계층화 알고리즘을 제시하므로써 FCM에 의한 시뮬레이션시 추론효과를 향상시키는 방법을 제시한다. 본 논문은 다음과 같이 구성되어 있다. 2절에서는 FCM내의 사이클을 확인하고 이를 해소하는 알고리즘을 제시한다. 3절에서는 동기화 문제를 해결하기 위한 계층화 알고리즘을 제시하고, 4절에서는 이를 이용한 추론과정과 예를 제시한다. 5절에서는 결론 및 향후연구방향을 제시한다.

2. FCM 내의 사이클 확인 및 해소

2.1 FCM 시뮬레이션 추론과 동기화

사이클이 있는 FCM을 논하기 앞서, FCM 시뮬레이션시 발생하는 동기화 문제부터 먼저 언급할 필요가 있다. 왜냐하면 결국 사이클 문제는 동기화와 관련이 있기 때문이다. 우선, FCM에 의한 시뮬레이션을 할 때에 사용되는 추론과정시 입출력값이 계산되는 과정부터 보자. 이는 다음과 같이 일반적으로 두단계로 구성되어 있다.

단계 1: 입력값 계산

$I(N_i)$ 및 $O(N_j)$ 를 각각 노드 N_i 의 입력값 및 출력값이라고 하고, $N_j^i, j=1, \dots, n_i$ 를 노드 N_i 에 영향을 미치는 원인노드라 하자. 그러면 $I(N_i)$ 는 다음과 같이 노드 N_i 에 영향을 미치는 원인노드들의 출력값의 가중합으로 구해진다.

$$I(N_i) = \sum_{j=1}^{n_i} e_{ji} \cdot O(N_j^i), \quad (\text{식 1})$$

여기서 e_{ji} : 퍼지인과관계 $N_j^i \rightarrow N_i$ 의 멤버십 값,

$O(N_j^i)$: 노드 N_j^i 의 출력값(확신도).

단계 2: 출력값의 계산

(식 1)에 의해 노드 N_i 의 입력값 $I(N_i)$ 가 정해지면 노드 N_i 의 확신도를 나타내는 출력값 $O(N_i)$ 는 다음과 같이 계산된다.

$$O(N_i) = I(N_i) .$$

예를 들어 <그림 1>에서 노드 N_2 및 N_5 의 출력값(확신도)이 각각 0.5와 1이라면 노드 N_4 에 대한 입력값은 $1 \times 0.5 + (-1) \times 1 = -0.5$ 가 되고 출력값도 그대로 -0.5가 된다.

본 연구의 경우 단계 2에서의 출력값 계산시 hyperbolic tangent 함수에 의해 계산되는 것으로 가정한다.

$$O(N_i) = \tanh [I(N_i)] . \quad (\text{식 2})$$

본 연구에서 기존의 FCM추론과 관련된 연구에서(Axelrod 1976; Lee 1992,1996; Kosko 1986,1992; Taber 1991) 흔히 채택하듯이 각 노드가 받아들인 입력값을 그대로 출력하지 않고, 이같이 hyperbolic tangent 함수를 이용하고자 하는 이유는 다음과 같다.

- 1) 각 노드의 출력값은 우리가 구하고자 하는 각 노드에 대한 확신도를 의미한다. 앞서 언급하였듯이 확신도는 $[-1, 1]$ 사이의 값을 가지며 hyperbolic tangent 함수는 출력값을 이 범위내로 한정시켜 준다.
- 2) 원인노드로 부터의 입력값이 0인 경우, 즉 원인에 대한 아무런 정보가 없을 때 출력값도 0이 되며, 또한 0을 중심으로 좌우대칭이다.
- 3) FCM에 의해 표현되는 문제영역은 대부분 불확실성을 내포하고 있다.

예를 들어보자. <그림 1 (a)>에서 노드 N_2 및

N_5 의 출력값(확신도)이 각각 0.5와 1이라면 노드 N_4 에 대한 입력값은 $1 \times 0.5 + (-1) \times 1 = -0.5$ 가 되고 출력값은 $\tanh(-0.5) = -0.46$ 이 된다. 즉 '소련 팽창주의' 및 'PLO 테러주의'에 대해 각각 긍정 및 강한 긍정을 나타내면 그 결과 '레바논에 대한 시리아의 통제'에 대해서는 부정이라는 결론을 내릴 수 있다는 의미이다. 특히, 불확실성이 있는 경우에는 보수적인 의사결정이 합리적인데, 이때 FCM추론시 hyperbolic tangent 함수를 이용하면 이러한 보수적인 의사결정을 적절하게 반영할 수가 있다. 예를 들어 극단적인 긍정을 의미하는 1이라는 입력값이 들어와도 출력값은 $\tanh(1) = 0.7616$ 으로 완화된으로써 보수적인 의사결정을 가능케 할 수 있다.

그러나, 이같이 hyperbolic tangent 함수를 사용하여 출력값을 계산할 경우, 기존의 연구에서처럼 단순 행렬연산을 통해 계산하는 것이 불가능하다. 왜냐하면 (식 1) 과 (식 2)에서 보듯이 출력값을 계산하기 위해서는 입력값, 즉 모든 원인노드들의 출력값에 대한 동기화(synchronization)가 필요하기 때문이다. 문제는 기존의 FCM 추론에서처럼 단순한 행렬연산으로는 모든 원인노드들의 출력값이 동기화되었는가를 판단하는 것이 불가능하다는 점이다. 따라서, 입력값의 동기화 여부에 대한 판단을 용이하게 하기 위해서 본 연구에서는 FCM을 계층화, 즉 levelization하는 알고리즘을 제시하고 그에 따른 새로운 FCM 시뮬레이션 추론방식을 제시하고자 한다. 그러나, 이 경우 대전제는 FCM내에 사이클이 존재해서는 안된다는 점이다. 그러므로 본격적인 FCM에 의한 시뮬레이션 추론방식을 거론하기 앞서 우선적으로 해결하여야 할 과제는 FCM내의 사이클을 발견하고 이를 해소하는 알고리즘을 개발하는 것이다.

2.2 사이클이 있는 FCM의 문제점

문제영역에 대한 초기 FCM에는 노드들 간에 사이클이 존재하는 경우가 종종 발생할 수 있다. FCM내에 사이클이 존재하면, 3절에서 설명할 계층화가 되지 않기 때문에 동기화 여부를 효과적으로 판단할 수가 없다. 이를 보다 구체적으로 보기 위하여 <그

림 1 (a)>에 나타나 있는 FCM을 보자. <그림 1 (a)>를 보면 노드 N_3, N_4, N_5 간에 사이클이 존재함을 알 수 있다. 그런데 FCM내에 이같은 사이클이 있는 경우에는 다음과 같은 문제점이 발생한다.

1) 주어진 FCM에 사이클이 존재하는 경우에는 3절에서 제시할 계층화 작업이 불가능하다. 계층에 대한 정의(3절의 [정의 4] 참조)를 보면 결과 노드의 계층은 원인 노드의 계층보다 항상 높아야 함을 알 수 있다. 그런데 사이클이 존재하는 경우에는 원인-결과 관계에 있어 순환이 존재함에 따라 주어진 정의에 의한 계층의 부여가 불가능하다는 것이다. 예를 들어 $L(N_i)$ 를 노드 N_i 에 대한 계층이라고 할 때, <그림 1 (a)>에서 $L(N_3) < L(N_5) < L(N_4) < L(N_3)$ 이라는 모순이 발생하게 된다. 이렇게 계층화가 불가능하다는 것은 노드의 입력값에 대한 동기화가 불가능하여 (식 1)과 (식 2)를 이용한 추론이 불가능하다는 것을 의미한다.

2) 사이클이 존재하는 FCM의 경우에는 추론과정에 있어서 값의 수렴(convergency)이 보장되지 않는다는 문제점이 있다. 예를 들어 <그림 1 (a)>에서 N_3 의 출력값(확신도)이 증가하면 N_5 의 출력값은 증가하고 이에 따라 N_4 의 출력값은 감소하게 된다. 그런데 감소된 N_4 의 출력값은 다시 N_3 의 출력값을 증가시킨다. 이렇게 사이클이 반복됨에 따라 N_5 의 값은 지속적으로 증가하고, 반면에 N_4 의 값은 지속적으로 감소된다. 또한 최종 노드인 N_6 의 값은 N_5 에 의한 감소와 N_4 에 의한 증가를 반복하게 되어 값이 수렴하지 않는다. 이처럼 값이 수렴하지 않을 경우 FCM에 의하여 주어진 문제에 대한 추론을 할 때에 문제가 발생한다. Kosko(1992)는 이러한 수렴문제를 해결하기 위하여 출력값이 0.5보다 작거나 같은 경우에는 출력값을 강제로 0으로 만들고 그렇지 않은 경우에는 1로 만드는 방법을 제시하였으나 이는 오히려 왜곡된 추론 결과를 초래할 가능성이 높다 (Kim & Lee 1995).

이러한 문제점들을 해결하기 위해 본절에서는 주어진 FCM에 내재된 사이클을 발견하고 해소하는 새로운 방법을 제시하고자 한다.

2.3 사이클 확인 알고리즘

FCM과 같은 유향성 그래프에 있어서 사이클을 발견하는 것은 NP-complete한 해밀턴 회로문제(Hamiltonian circuit problem) 영역에 속한다 (Papadimitriou & Steiglitz 1982). 따라서, 기존의 사이클 발견 알고리즘은 지수적인(exponential) 시간을 요한다 (Deo 1974). 따라서 본 연구에서는 FCM의 인과관계 행렬을 단순히 인과관계 여부만을 나타내는 연관행렬로 변환한 다음 이를 기초로 새로운 FCM 사이클 발견 기법을 제시하고자 한다. 연관행렬에 대한 정의는 다음과 같다.

[정의 1] 연관(Adjacency)행렬 : 주어진 FCM에 속한 노드들의 개수를 n 이라 하면 $n \times n$ 연관행렬 A 는 다음과 같이 정의된다.

$$A_{ij} = \begin{cases} 1, & N_i \rightarrow N_j \text{ 인 경우} \\ 0, & \text{여타의 경우} \end{cases} \quad i, j = 1, 2, \dots, n.$$

만일 인과관계행렬 E 가 미리 주어진 경우 연관행렬 A 는 다음과 같이 구할 수도 있다.

$$A_{ij} = \begin{cases} 1, & e_{ij} \neq 0 \text{ 인 경우} \\ 0, & \text{여타의 경우} \end{cases} \quad i, j = 1, 2, \dots, n \blacksquare$$

<그림 1 (a)>에 나타난 FCM에 대한 연관행렬이 <그림 1 (c)>에 나타나 있다.

이제 <그림 1 (2a)>처럼 사이클이 있는 FCM의 경우 이상에서 정의된 연관행렬이 어떤 특성을 갖게 되는지 살펴보자.

연관행렬의 곱 A^k 의 의미

연관행렬 A의 각 요소 A_{ij} 는 노드 N_i 로부터 노드 N_j 로의 인과관계가 있는가의 여부를 나타낸다. 즉 인과관계가 있으면 1이고 없으면 0이다. 그렇다면 연관행렬 A의 제곱인 A^2 은 어떤 의미를 가지는가? A^2_{ij} 를 행렬 A^2 의 i번째 행과 j번째 열에 해당하는 요소라고 하면 이는 노드 N_i 로부터 노드 N_j 로 2단계만에 연결되는 인과관계 경로의 수를 의미한다. 참고로 <그림 1 (a)>의 FCM에 대한 A^2 이 <그림 2>에 나타나 있다. <그림 2>에서 A^2_{23} 의 값이 1 2이라는 것은 노드 N_2 로부터 노드 N_3 으로 2단계만에 도달할 수 있는 경로의 수가 하나 ($N_2 \rightarrow N_4 \rightarrow N_3$)라는 것을 의미한다. 이러한 논의를 일반화하면 행렬 A^k 는 노드간에 k개의 호를 통해서 연결될 2수 있는 인과관계 경로의 수를 표시한다는 것을 알 수 있다. 특히 A^k 의 대각선 요소에는 k단계만에 자기자신으로 되돌아 올 수 있는 경로 즉 사이클이 존재하는 경우에는 0이 아닌 숫자가 나타나게 됨을 유추할 수 있다.

	N_1	N_2	N_3	N_4	N_5	N_6
N_1	0	0	0	1	1	0
N_2	0	0	1	0	0	1
N_3	0	0	0	1	0	1
N_4	0	0	0	0	1	0
N_5	0	0	1	0	0	1
N_6	0	0	0	0	0	0

<그림 2> 그림 1 (a)의 FCM에 대한 A^2

크기가 k인 사이클이 존재할 경우의 A^k 의 형태

우선 크기가 k인 사이클이 하나만 존재하는 경우를 생각해 보자. 여기서 사이클의 크기란 해당 사이클에 포함되는 노드의 수를 의미한다. 크기가 k인

사이클이 존재하는 경우 해당 사이클을 구성하는 각 노드로부터 시작하여 k단계만에 자기자신에게 돌아오는 경로가 존재하게 된다. 이는 N_i 가 크기가 k인 사이클에 속하는 노드일 경우 A^k 의 대각선 요소중 노드 N_i 에 해당하는 A^k_{ii} 의 값이 1이 됨을 의미한다. 따라서 크기가 k인 사이클이 하나만 존재하는 경우 A^k 는 해당 사이클을 구성하는 k개의 노드에 대응하는 대각선 요소에 1이 나타나는 형태를 지니게 된다. <그림 1 (a)>의 FCM에 대한 A^3 가 <그림 3>에 나타나 있다. <그림 3>에서 A^3 의 대각선 요소에 3개의 1이 존재한다는 사실을 통해 크기가 3인 사이클이 하나 존재하며 해당 사이클을 구성하는 노드들은 $N_3, N_4,$ 및 N_5 임을 알 수 있다.

	N_1	N_2	N_3	N_4	N_5	N_6
N_1	0	0	1	1	0	2
N_2	0	0	0	0	1	0
N_3	0	0	①	0	0	1
N_4	0	0	0	①	0	1
N_5	0	0	0	0	①	0
N_6	0	0	0	0	0	0

<그림 3> 그림 1 (a)의 FCM에 대한 A^3

크기가 k인 사이클이 두개 이상 존재하는

경우 A^k 의 형태

이제 크기가 k인 사이클이 $c(>1)$ 개 존재하는 경우를 살펴보자. 이때 A^k 의 형태는 다음과 같이 두 가지 경우로 나누어 생각해 볼 수 있다.

- ① 사이클을 구성하는 노드들 간에 중복이 없는 경우 이 경우에는 c개의 사이클을 구성하는 모든 노드에 대해 k단계만에 자기자신에게 돌아오는 경로가 각각 1개씩 존재하게 되므로 A^k 의 대각선에 $c \times k$ 개의 1이 나타나게 된다.
- ② 사이클을 구성하는 노드들 간에 중복이 있는 경우 이 경우 A^k 의 대각선 요소는 1보다 큰 값을 가

질수 있으며 그값은 해당 노드를 포함하는 사이클의 수를 의미한다. 하지만 이 경우에도 대각선 요소의 총합은 $c \times k$ 가 된다. 예를들어 크기가 4인 사이클이 2개 존재하고 각 사이클을 구성하는 노드 중 2개가 중복된다고 하자. 그러면 A^4 의 대각선에는 4개의 1과 2개의 2가 나타나게 되며, 총합은 $2 \times 4 = 8$ 이 된다.

크기가 k인 사이클이 존재하는 경우 A^{nk} 의 형태

크기가 k인 사이클이 존재하는 경우 사이클을 구성하는 각 노드로 부터 k단계만에 자기 자신에게 돌아오는 경로가 존재한다. 그런데 k단계만에 자기 자신에게 되돌아 올 수 있다면 k의 임의의 배수인 nk단계만에 돌아올 수 있다는 말이된다. 따라서 크기가 k인 사이클이 존재하는 경우 A^k 뿐만아니라 A^{nk} 의 대각선에도 k개의 1이 나타나게 된다. 그런데 크기가 k인 사이클 뿐만아니라 크기가 nk인 사이클이 동시에 존재하는 경우의 A^{nk} 의 형태는 어떻게 되는가? 이 경우에는 A^{nk} 의 대각선 요소 중 0이 아닌 값을 갖는 것의 개수가 nk보다 크게 된다. 따라서 A^{nk} 의 대각선 요소에 0이 아닌 수가 나타날 경우 실제로 크기가 nk인 사이클이 존재하는가 아니면 크기가 k인 사이클이 존재하기 때문인가에 대한 판단은 0이 아닌 대각선 요소의 개수를 세어 봄으로써 알 수 있다.

크기가 각각 k, m인 사이클이 존재하는 경우 A^{km} 의 형태

크기가 각각 k, m인 사이클이 존재하는 경우 A^{km} 의 대각선 요소에는 두 사이클 간에 중복되는 노드의 개수 만큼의 1이 나타나게 된다. 그렇다면 크기가 각각 k, m인 사이클 뿐만아니라 크기가 km인 사이클이 동시에 존재하는 경우의 A^{km} 의 형태는 어떻게 되는가? 앞서와 마찬가지로 이 경우에도 A^{km} 의 대각선 요소중 0이 아닌 값을 갖는 것의 개수가 km보다 크게 된다. 따라서 A^{km} 의 대각선 요

소에 0이 아닌 수가 나타날 경우 실제로 크기가 km인 사이클이 존재하는 가의 여부는 0이 아닌 값을 갖는 요소의 수를 세어 봄으로써 판단할 수 있다.

이상에서 언급한 연관행렬의 곱의 대각선 요소의 형태와 사이클의 존재유무와의 관계를 종합해 보면 [정리 1]과 같다.

[정리 1] $D(A^k)$ 를 A^k 의 대각선 요소중 0이 아닌 것의 개수라 하고, $S(A^k)$ 를 A^k 의 대각선 요소의 총합($\sum_{i=1}^n A_{ii}^k$)이라고 하자. 그러면 다음의 사실이 성립한다.

- ① $D(A^k) < k$: 크기가 k인 사이클이 존재하지 않는다.
- ② $D(A^k) = S(A^k) = k$: 크기가 k인 사이클이 1개 존재하고 이 사이클을 구성하는 노드들은 1의 값을 가지는 대각선 요소에 해당하는 노드들이 된다. (사이클의 경로는 <표 1> 참조)
- ③ $D(A^k) > k, nk \leq S(A^k) < (n+1)k$ (여기서 n은 부동소수가 성립하는 최대의 자연수) : 크기가 k인 사이클이 n개 존재하며 이 경우 각 사이클에 포함되는 노드들 및 이들 간의 경로를 규명하는 알고리즘은 <표 1>과 같다.

<표 1> 사이클의 경로 규명 알고리즘

Step 1 : Preparation

- ① A^k 의 대각선 요소중 0이 아닌 값을 갖는 노드들의 집합 $Z = \{N_i \mid A_{ii}^k \neq 0\}$ 를 구한다.
- ② $n = 1$ 이라 놓는다.

Step 2 : Initialization

- ① 크기가 k인 n번째 사이클을 $C_n = \{\emptyset\}$ 라 한다.
- ② 임의의 $N_j \in Z$ 를 택한다.

Step 3 : Path(N_j)

IF $N_j \in Z$ && $A_{jj} = 1$ 인 N_j 가 존재하면

```

THEN {
    Cn = Cn ∪ { Ni → Nj };
    Aijk = Aijk - 1;
    IF Aijk == 0 THEN Z =
        Z - { Nj };
    Path( Nj ); }
ELSE
    { IF Cn 의 크기 == k
      THEN Report Cn ;
    ELSE Cn 을 버린다.
    IF Z에 남아 있는 노드의 수 < k
      THEN Stop.
    ELSE n = n + 1 and goto Step 2. }
    
```

그런데 [정리 1]의 내용을 바탕으로 주어진 FCM에 내포된 모든 사이클을 발견하기 위하여 연관행렬 A를 몇번이나 곱해야 하는가? 이에 대한 해답은 가능한 최대 경로의 수만큼만 곱하면 된다는 것이다. 그런데 입력 노드간 및 출력 노드간에는 인과관계가 존재하지 않는다는 사실을 고려하면 최대 경로의 수는 다음과 같다. 전체 노드의 수를 n이라 하면

$$\begin{aligned}
 \text{최대 경로의 수} &= n - (\text{입력노드의 수} - 1) \\
 &\quad - (\text{출력노드의 수} - 1) - 1 \\
 &= n - (\text{입력노드의 수} + \\
 &\quad \text{출력노드의 수}) + 1
 \end{aligned}$$

이 된다. 따라서 <그림 1 (a)>에 나타난 FCM의 경우 최대 경로의 수는 5가 되며, 이에 따라 A⁵ 까지만 계산하면 된다. 이 FCM에 대해 A², A³, A⁴ 및 A⁵를 계산해 보면 대각선에 0이 아닌 수가 발생하는 것은 <그림 3>에 나타난 A³뿐이다. 따라서 이를 바탕으로 정리 1 및 <표 1>의 알고리즘을 적용하면 사이클 N₃ → N₅ → N₄ → N₃ 가 도출된다.

2.4 사이클 해소를 위한 휴리스틱

Jung & Burns(1993)는 사이클을 해소하기 위해 사이클을 구성하는 모든 노드들을 하나의 커다란 노드로 통합(merge)하는 방법을 제안하였다. 이 방법은 간편하기는 하나 다음과 같은 문제점이 있다. 첫째, 사이클을 구성하는 노드의 수가 많거나 개념상이질적인 노드로 구성되어 있는 경우 통합에 무리가 따른다. 둘째, 통합했다고 하더라도 통합된 노드와 사이클에 포함되지 않는 여타 노드들간에 일관성있는 인과관계를 부여하기가 어렵다는 점이다. 따라서 본 연구에서는 추론에 있어서 사이클의 효과를 최대한 유지하면서도 사이클을 구성하는 호를 하나 제거하는 것을 통해 사이클을 해소하는 새로운 방법을 제시하고자 한다. 우선 필요한 용어를 정의하면 다음과 같다.

[정의 2] 노드 N_i 의 Fan-in 수, F^I(N_i) : F^I(N_i)로 표기되는 노드 N_i의 Fan-in 수는 노드 N_i에 영향을 미치는 원인노드들의 수를 의미한다. 즉, 노드 N_i로 들어오는 호의 수를 나타낸다. 따라서 n x n 연관행렬 A가 주어진 경우 F^I(N_i)는 다음과 같이 계산된다:

$$F^I(N_i) = \sum_{j=1}^n A_{ji} \quad \blacksquare$$

예를 들어 <그림 1 (a)>의 FCM에서 각 노드들에 대한 Fan-in 수는 다음과 같다.

$$F^I = [F^I(N_1), \dots, F^I(N_6)] = [0, 1, 2, 2, 1, 2].$$

[정의 3] 노드 N_i 의 Fan-out 수, F^O(N_i) : F^O(N_i)로 표기되는 노드 N_i의 Fan-out 수는 노드 N_i에 의해 영향을 받는 결과노드들의 수를 의미한다. 즉, 노드 N_i로부터 나오는 호의 수이다. 따라서 n x n 연관행렬 A가 주어진 경우 F^O(N_i)는 다음과 같이 계산된다:

$$F^O(N_j) = \sum_{j=1}^n A_{ij} \quad \blacksquare$$

예를 들어 <그림 1 (a)>의 FCM에서 각 노드들에 대한 Fan-out 수는 다음과 같다.

$$F^O = [F^O(N_1), \dots, F^O(N_6)] = [2, 1, 1, 2, 2, 0].$$

이상에서 정의된 용어를 바탕으로 본 연구에서 제시하는 사이클 해소방법은 제거 대상 호를 선정하는 단계 및 사이클의 효과를 유지시키기 위해 특정 노드의 출력값을 변화시키는 단계로 나누어진다.

단계1: 제거대상 호를 선정하기 위한 휴리스틱

사이클을 해소하기 위한 제거대상 호를 선정함에 있어서 지켜져야 할 기준은 문제영역을 표현하고 있는 FCM에 미치는 영향을 최소화 해야 한다는 점이다. 이를 위해 본 연구에서는 다음과 같은 휴리스틱을 바탕으로 하였다. 사용된 휴리스틱을 적용 우선순위에 의거하여 나열하면 다음과 같다.

H1 : 호가 제거됨으로 인해 중간노드가 입력 또는 출력노드화 되어서는 안된다.

사이클이 성립되기 위해서는 사이클을 구성하는 각 노드가 들어오는 호 및 나가는 호를 가져야 한다. 따라서 사이클을 구성하는 노드들은 모두 중간노드이다. 그런데 FCM에 있어서 입력노드는 측정 또는 통제가능하다는 의미를 가지고 있고 반면에 출력노드는 최종 관심의 대상이라는 의미를 가지고 있으므로 호가 제거됨으로 인하여 기존의 중간노드가 인위적으로 입력 또는 출력노드화 되어서는 안된다는 것이다. 따라서 노드 N_i 와 노드 N_j 를 연결하는 호를 a_{ij} 라 할 때 $F^O(N_j) = 1$ 이거나 $F^I(N_j) = 1$ 인 호 a_{ij} 는 제거대상에서 우선 제외시킨다. 예를 들어 <그림 1 (a)>의 FCM에서 사이클을 구성하고 있는 호는 a_{35} , a_{54} 및 a_{43} 이다. 이중 $F^O(N_3) = 1$ 이므로 호 a_{35} 는 제거대상에서 제외시킨다.

H2 : $F^O(N_j)$ 가 가장 작은 호 a_{ij} 를 제거 대상으로 선택한다.

사이클이 해소됨으로 인해 사이클을 구성하는 노드들의 출력값은 필연적으로 오차를 갖게 된다. 특히 제거되는 호에 연결된 결과노드의 출력값이 가장 큰 영향을 받는다. 사이클의 효과를 최대한 유지시키기 위해서는 이러한 오차가 사이클에 포함되지 않는 여타 노드에 미치는 영향을 최소화하는 것이 필요하다. 따라서 H1 적용후 제거대상 호가 두 개 이상 남아 있을 경우 이중 노드 N_j 가 영향을 미치는 다른 노드의 수인 $F^O(N_j)$ 가 최소가 되는 a_{ij} 를 제거하는 것이 효과적이다. <그림 1 (a)>의 FCM에 대해 H1을 적용한 결과 남은 호는 a_{54} 및 a_{43} 이다. 그런데 $F^O(N_3) = 1 < F^O(N_4) = 2$ 이므로 a_{43} 을 제거대상 호로 선정하게 된다.

H1 및 H2를 적용한 후에도 제거대상 호가 두 개 이상 남아 있는 경우 최종적으로 다음의 휴리스틱을 적용한다.

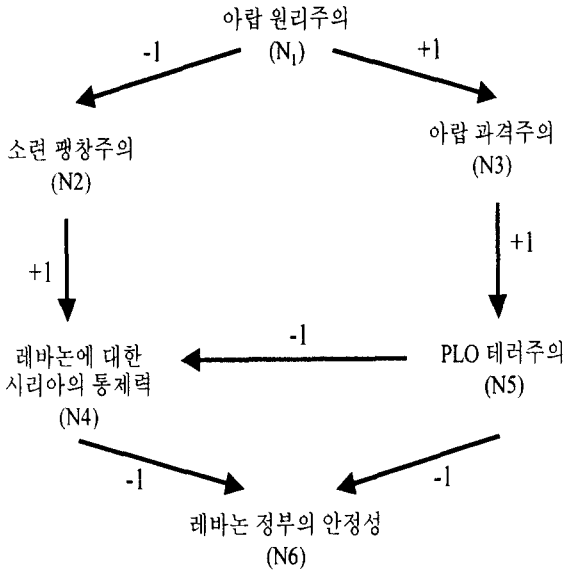
H3 : $\frac{1}{n_j} \sum_{k=1}^{n_j} F^I(N'_k)$ 가 최대가 되는 노드

N_j 에 연결되어 있는 호 a_{ij} 를 제거 대상으로 선택한다.

$N'_k, k = 1, 2, \dots, n_j$ 를 노드 N_j 의 영향을 받는 결과 노드라 하자. 그러면 $F^I(N'_k)$ 값이 크면 클 수록 노드 N'_k 에 대한 노드 N_j 의 영향은 작아진다고 볼 수 있다. 따라서 $F^I(N'_k)$ 의 평균값인

$\frac{1}{n_j} \sum_{k=1}^{n_j} F^I(N'_k)$ 가 최대가 되는 노드 N_j 에 연결되어 있는 호 a_{ij} 를 제거하는 것이 효과적이다.

<그림 1 (a)>의 FCM에 제시된 휴리스틱의 적용을 통해 사이클을 제거한 FCM이 <그림 4>에 나타나 있다.



<그림 4> 사이클이 제거된 FCM

단계2: 사이클의 효과를 최대한 유지시키기 위한 특성노드의 출력값 변경

특성 노드란 앞절에서 제시한 방법에 의해 제거된 호의 결과 부분에 해당하는 노드를 의미한다. 예를 들어 <그림 4>의 FCM에 있어서 제거된 호는 노드 N_4 와 노드 N_3 를 연결하는 a_{43} 이었다. 따라서 이 경우 호 a_{43} 의 결과 부분에 해당하는 N_3 가 특성 노드가 된다. 이러한 특성 노드는 사이클 해소에 의해 출력값의 변화가 야기되는 시발점이 된다. 따라서 본절에서는 사이클 해소에 의한 영향을 최소화하기 위해 변화의 시발점이 되는 특성노드의 출력값을 변경하는 방법을 제시하고자 한다. 이를 위해 사이클을 다음과 같이 두가지 종류로 나누어 생각한다.

(A) 강화(exitatory) 사이클 : 강화 사이클이란 사이클을 구성하는 호중 멤버쉽 값이 음수인 호가 없거나 있더라도 짝수개 존재하는 경우를 말한다. 강화 사이클인 경우 추론시 사이클이 반복됨에 따라 사이클에 속하는 노드들의 출력값이 지속적으로 강화된다. 따라서 강화 사이클을 해소시킨 경우 애초의 사

이클의 영향을 최소화시키기 위해서는 특성 노드에 대한 출력값을 다음과 같이 강화시키는 것이 필요하다. 즉 노드 N_j 가 강화 사이클에 대한 특성 노드일 경우 N_j 의 출력값 $O(N_j)$ 의 계산 방식은 다음과 같이 변경된다 :

$$O(N_j) = (\text{sign})\sqrt{|\tanh[I(N_j)]|} .$$

여기서 (sign)은 애초의 $\tanh[I(N_j)]$ 의 부호를 의미한다.

(B) 약화(inhibitory) 사이클 : 약화 사이클이란 사이클을 구성하는 호중 멤버쉽 값이 음수인 호가 홀수개 존재하는 경우를 말한다. 약화 사이클인 경우 강화 사이클과는 반대로 추론시 사이클이 반복됨에 따라 사이클에 속하는 노드들의 출력값이 지속적으로 약화된다. 따라서 약화 사이클을 해소시킨 경우 애초의 사이클의 영향을 최소화시키기 위해서는 특성 노드에 대한 출력값을 다음과 같이 약화시키는 것이 필요하다. 즉 노드 N_j 가 약화 사이클에 대한 특성 노드인 경우 N_j 의 출력값 $O(N_j)$ 의 계산 방식은 다음과 같이 변경된다 :

$$O(N_j) = \tanh[I(N_j)] \cdot |\tanh[I(N_j)]| .$$

<그림 4>의 FCM에서 제거된 사이클은 강화 사이클이다. 따라서 노드 N_1 에 대한 확신도가 강한 부정을 나타내는 -1 이라면 특성노드 N_3 의 출력값(확신도) $O(N_3)$ 는 다음과 같이 계산된다.

$$\begin{aligned} O(N_3) &= (-) \sqrt{|\tanh(-1)|} = - \sqrt{|-0.76|} \\ &= - 0.87 \end{aligned}$$

이 된다.

이상과 같이 사이클을 해소한 이후에는 3절에서 제시하는 계층화 기법을 적용할 수 있다. 하지만 주의할 점은 특성 노드에 대한 출력값 계산만은 본절에서 제시한 방법이 (식 2)보다 우선한다는 점이다.

3. FCM의 계층화 알고리즘

FCM의 계층화란 주어진 FCM을 계층적인 형태로 재구성하는 것이다. 계층화된 FCM을 사용하면 다음과 같은 장점이 있다.

- 1) 주어진 FCM을 보다 체계적으로 나타내 보임으로써 문제영역에 대한 이해도를 높이고 잘못된 부분이나 누락된 부분의 발견을 용이하게 한다.
- 2) 추론시 각 노드의 출력값을 계산하는 시점여부, 즉 입력값이 동기화되었는가에 대한 판단을 용이하게 한다.
- 3) 계층화된 FCM은 계층 r에 속하는 노드의 출력값 계산은 계층 r-1이하에 속하는 모든 노드의 출력값이 계산(동기화)된 이후에 가능하다는 것을 보여준다.

이같은 장점 때문에 계층화된 FCM을 이용하면, 주어진 입력노드의 값을 바탕으로 낮은 계층에 속하는 노드의 출력값부터 차례로 계산해 나갈수가 있고, 따라서 앞절에서 언급한 동기화 문제를 쉽게 해결할 수 있다. 우선 FCM을 계층화하기 위하여 필요한 정의를 제시하면 다음과 같다.

[정의 4] $L(N_i)$ 를 노드 N_i 의 계층이라 하자. 그러면 $L(N_i)$ 는 다음과 같이 정의된다.

- ① N_i 가 입력노드일 경우 : $L(N_i) = 0$.
- ② N_i 가 입력노드가 아닐 경우 :

$$L(N_i) = \text{MAX}_{j=1}^{n_i} L(N_j) + 1,$$

여기서 $N_j, j=1, \dots, n_i$ 는 노드 N_i 에 영향을 미치는 원인노드들 ■

계층에 대한 정의를 바탕으로 주어진 FCM의 각 노드가 속하는 계층을 구하기 위한 알고리즘을 제시하기 위해서는 다음의 추가적인 정의가 필요하다.

[정의 5] 동기(Synchronization)벡터 : 주어진 FCM에 나타난 노드들의 수가 n인 경우 r차 $1 \times n$ 동기

벡터 $S^r, r = 0, 1, 2, \dots$ 은 다음과 같이 정의된다.

$$S^0: S_i^0 = \begin{cases} 1, & N_i \text{ 가 입력노드인 경우} \\ 0, & N_i \text{ 가 입력노드가 아닌 경우} \end{cases} \quad i = 1, \dots, n.$$

$$S^r = \text{Th}(S^{r-1} \cdot A) + S^{r-1}, \quad r \geq 1.$$

위식에서 연산자 Th()의 기능은 다음과 같다.

$B = S^{r-1} \cdot A$ 라 하고, $C = \text{Th}(B)$ 라 하자. 그러면 $1 \times n$ 벡터 C는 다음과 같이 정의된다 ($F^I(N_i)$ 에 대한 [정의 2] 참조) :

$$C_i = \begin{cases} 1, & B_i = F^I(N_i) \text{ 인 경우} \\ 0, & B_i < F^I(N_i) \text{ 인 경우} \end{cases}, \quad i = 1, \dots, n.$$

$N_j^i, j=1, \dots, n_i$ 를 노드 N_i 에 대한 원인노드라 할 때, $S^r, r = 1, 2, \dots$ 은 다음과 같은 성질을 지닌다.

$$S_i^r = \begin{cases} 1, & \text{Max}_{j=1}^{n_i} [L(N_j^i)] < r \text{ 인 경우} \\ 0, & \text{여타의 경우} \end{cases} \quad i = 1, \dots, n.$$

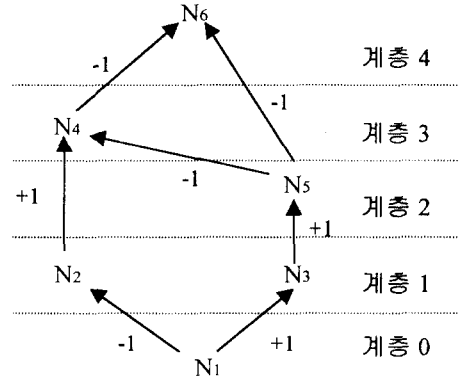
즉 노드 N_i 의 원인노드들의 계층이 r보다 작을 때 S_i^r 의 값은 1이 되고 그렇지 않은 경우 S_i^r 의 값은 0이 된다. 이러한 동기벡터의 특성과 [정의 4]에 나타난 계층에 대한 정의를 동시에 고려하면 다음과 같은 사실이 유도된다.

$$L(N_i) = \text{Min} \{ r \mid S_i^r = 1 \}.$$

이상의 내용을 바탕으로 각 노드들에 대한 계층을 구하는 알고리즘을 제시하면 <표 2>와 같다.

<표 2> FCM 계층화 알고리즘

Step 1 : 0차 동기벡터 S^0 를 구한다.
 Step 2 : $r = 1$ 이라 놓는다.
 모든 중간노드 및 출력노드의 집합을 C 라 한다.
 Step 3 : r 차 동기벡터
 $S^r = Th(S^{r-1} \cdot A) + S^{r-1}$ 를 구한다.
 Step 4 : 모든 $N_i \in C$ 에 대해 다음을 반복한다.
 IF $S_i^r == 1$
 THEN {
 $L(N_i) = r;$
 $C = C - N_i;$
 ELSE $L(N_i) =$ 미정.
 Step 5 : 모든 노드에 대한 계층이 구해졌는가를 판단한다.
 IF $C == \emptyset$ THEN Stop.
 ELSE $r = r + 1$ and goto Step 3.



<그림 5> 계층화된 FCM

<그림 5>를 보면 입력노드가 속하는 계층은 0임을 알 수 있다. 또한 결과 노드가 속하는 계층은 해당 노드에 대한 입력노드가 속하는 계층보다 상위 계층에 속함을 알 수 있다.

<표 2>의 계층화 알고리즘을 <그림 4>의 FCM에 적용해 보면 다음과 같다.

- $S^0 = [\textcircled{1}, 0, 0, 0, 0, 0] \Rightarrow L(N_1) = 0.$
- $S^1 = [1, \textcircled{1}, \textcircled{1}, 0, 0, 0] \Rightarrow L(N_2) = 1, L(N_3) = 1.$
- $S^2 = [1, 1, 1, 1, 0, \textcircled{1}, 0] \Rightarrow L(N_5) = 2.$
- $S^3 = [1, 1, 1, 1, \textcircled{1}, 1, 0] \Rightarrow L(N_4) = 3.$
- $S^4 = [1, 1, 1, 1, 1, 1, \textcircled{1}] \Rightarrow L(N_6) = 4.$

즉, 노드 N_1 은 계층 0, N_2 와 N_3 은 계층 1, N_5 는 계층 2, N_4 는 계층 3, 그리고 N_6 은 계층 4에 속하는 것을 알 수 있다. 이렇게 구한 각 노드에 대한 계층을 바탕으로 FCM을 재작성하면 <그림 5>와 같다.

4. 계층화된 FCM에 대한 추론절차

<그림 5>와 같이 계층화된 FCM이 구해지면 동기화가 필요한 추론이 간편해진다. 즉 (식 1)과 (식 2)를 이용해 낮은 계층에 속하는 노드에 대한 입출력부터 차례로 계산해 나가면 FCM 추론시 발생하는 동기화 문제를 해결할 수가 있다. 이러한 계층별 계산 절차를 벡터 형태로 표기하면 다음과 같다. 우선 계층 r 에 속하는 노드들에 대한 입력값 계산은 다음과 같다.

$$I_r = E_r \cdot O^r \tag{식 3}$$

여기서 I_r : 계층 r 에 속하는 노드들에 대한 입력값을 나타내는 벡터

O^r : I_r 에 속하는 노드에 대한 모든 원인노드들의 출력값을 나타내는 벡터

E_r : O^r 에 속하는 노드와 I_r 에 속하는 노드간의 퍼지인과관계의 멤버십 값으로 구성된 행렬.

그리고 계층 r에 속하는 노드들의 출력값(확신도)을 나타내는 O_r 은 다음과 같이 계산된다.

$$O_r = \tanh(I_r) \quad (\text{식 4})$$

(식 3)에서 E_r 는 인과관계 행렬 E 를 다음과 같이 축약하여 구할 수 있다. 우선 E에서 I_r 에 속하는 노드들에 대한 행만 남기고 나머지 행들을 제거한다. 그리고 나서 0으로만 이루어진 열들을 추가적으로 제거하면 E_r 이 된다. 또한 E_r 의 열에 해당하는 노드들이 O_r 에 속하는 노드가 된다. <그림 5>의 계층화된 FCM에서 측정가능한 입력노드인 아랍 원리주의에 대한 확신도가 강한 긍정을 나타내는 1이라고 가정하고 (식 3) 및 (식 4)에 나타난 계산 절차를 예시하면 다음과 같다.

$$O^1 = [O(N_1)] = [1]$$

$$I_1 = [I(N_2), I(N_3)]' = E_1 \cdot O^1 = [-1, 1]' \cdot [1] = [-1, 1]'$$

$$O_1 = [O(N_2), O(N_3)]' = [\tanh(-1), \sqrt{\tanh(1)}]' = [-0.76, 0.87]'$$

(여기서 N_3 는 강화 사이클의 특성노드이기 때문)

$$I_2 = [I(N_5)] = E_2 \cdot O^2 = [1] \cdot [0.87] = [0.87]$$

$$O_2 = [O(N_5)] = [\tanh(0.87)] = [0.7]$$

$$I_3 = [I(N_4)] = E_3 \cdot O^3 = [1, -1] \cdot [-0.76, 0.7]' = [-1.46]$$

$$O_3 = [O(N_4)] = [\tanh(-1.46)] = [-0.9]$$

$$I_4 = [I(N_6)] = E_4 \cdot O^4 = [-1, -1] \cdot [-0.9, 0.7]' = [0.2]$$

$$O_4 = [O(N_6)] = [\tanh(0.2)] = [0.2]$$

==> 최종 추론결과 :

$$[O(N_1), O(N_2), O(N_3), O(N_4), O(N_5), O(N_6)] = [1, -0.76, 0.87, -0.9, 0.7, 0.2]$$

따라서 입력노드인 아랍 원리주의에 대해 강한 긍정을 나타낼 경우 최종 출력노드인 '레바논 정부의 안정성'에 대해서는 약한 긍정(0.2)이라는 결론을

얻을 수가 있다.

이같은 결과를 기존의 FCM추론방식과 비교하여 보자. 예를들어, <그림 1 (a)>에서 제시된 키신저 박사의 중동평화에 관한 인과관계 지식을 다시 한번 살펴보면 FCM에는 인과관계 피드백 순환루프가 여러 곳에 있을 수 있다. 그러한 피드백 때문에 전문가 는 자유롭게 문제에 대한 인과관계 경로를 그릴 수 있고, 표본 데이터로부터 인과관계 경로를 추론하기 위한 인과관계 적응규칙을 생성할 수도 있다. 본 논문에서는 FCM을 동적시스템(dynamic system)으로 보고, 시간이 경과함에 따라 특정한 균형상태 (stable state)로 수렴한다고 가정한다. 따라서 주어진 외부자극에 대해서 FCM으로 표현된 인과관계 동적시스템은 반응하며 이를 시간의 경과에 따른 "전방향 진행추론" (forward-evolved inference)이라고 한다 (Kosko 1992). <그림 1 (a)>에서의 인과관계 지식을 이용하여 FCM에 의한 전방향 진행추론을 살펴보자. 이러한 추론을 하기에 앞서 우선 개념들로 구성된 '개념노드 벡터' (concept node vector)를 가정하자. <그림 1 (a)>에는 6개의 개념노드가 있다. 따라서 우리는 개념노드 벡터 N를 다음과 같이 정의할 수 있다.

$$N = (N_1, N_2, N_3, N_4, N_5, N_6)$$

이때 각 개념노드는 <그림 1 (a)>의 개념을 의미한다. 예를 들어 아랍 원리주의 (Arab Fundamentalism)가 미치는 영향을 분석하려면 N1을 1로 둔 다음과 같은 개념노드 벡터를 전제하여야 한다.

$$N_1 = (1 \ 0 \ 0 \ 0 \ 0 \ 0)$$

이를 <그림 1 (b)>에서의 인식도(인과관계) 행렬 E와 곱하면

$$N_1 \times E = (0 \ -1 \ 1 \ 0 \ 0 \ 0) \longrightarrow (1 \ 0 \ 1 \ 0 \ 0 \ 0) = N_2$$

와 같이 계산된다. 이때 화살표는 1/2 임계치 기

준을 적용한 것으로서 1/2보다 크면 1, 작으면 0으로 간주한 것이다. N2는 첫번째 요소를 1로 하고 있는데 이는 첫번째 개념노드인 '아랍 원리주의'의 영향을 보고자 하기 때문이다. 같은 방법으로 N2에 인식도 행렬을 곱하면,

$$\underline{N}_2 \times \underline{E} = (0 \ -1 \ 1 \ 0 \ 1 \ 0) \longrightarrow (1 \ 0 \ 1 \ 0 \ 1 \ 0) = \underline{N}_3$$

이 되고, 우리는 여기에서 다섯번째 개념노드가 1로 수정된 것을 볼 수 있다. N3에 인식도 행렬을 다시 곱하면,

$$\underline{N}_3 \times \underline{E} = (0 \ -1 \ 1 \ -1 \ 1 \ -1) \longrightarrow (1 \ 0 \ 1 \ 0 \ 1 \ 0) = \underline{N}_3$$

우리는 N3가 <그림 1 (a)>에서 주어진 인식도의 균형상태임을 알 수 있다. 균형상태에서 보면 결국 아랍 원리주의가 계속 견지되는 한, 아랍 과격주의(세번째 개념노드)가 일어날 것이며 이는 다시 PLO의 테러주의를 부채질할 것임을 쉽게 예측할 수 있다. 아울러 소련의 팽창주의는 오히려 감소되나 레바논 정부의 힘이 그만큼 약해져서 레바논내의 정치적 불안이 초래될 수 있음을 알 수 있다. 결국 우리는 아랍 원리주의가 중동평화에 전체에 미치는 복합적인 영향이 어느정도인지를 이와같은 전방향 진행추론을 통하여 알 수 있다.

그러나, 본 연구에서 제시한 추론결과(1 -0.76 0.87 -0.9 0.7 0.2)는 위에서 설명한 내용을 더욱 명확하게 보여주고 있다. 특히 최종 출력노드인 레바논 정부의 안정성이라는 최종출력노드의 출력값이 0.2로서 '약한 긍정'으로 유도되는 반면에 기존의 추론결과는 단지 '레바논 정부의 안정성'에 해당되는 노드의 출력값이 0으로 되어 정확하게 어느 정도의 출력값을 갖는지가 불분명하다. 또한, 추론과정에서 임의로 1/2이라는 임계치를 사용하므로써 자의적인 추론결과값을 유도한다는 단점이 있다. 그러나, 본 연구에서 제시하는 FCM추론방법은 이러한 기존의 FCM추론방식의 단점을 효과적으로 극복하기 때문에 보다 정교한 추론이 가능하다.

5. 결론 및 향후 연구방향

본 연구에서는 기존의 연구에서와는 달리 FCM을 이용한 추론방식에서 동기화 문제를 지적하였고, 그에 대한 해결 방안으로서 계층화 알고리즘을 제안하였다. 또한 계층화 알고리즘 적용시 문제가 되는 FCM내의 사이클 문제를 발견하고 이를 해소하는 알고리즘을 제시하므로써 FCM 시뮬레이션의 추론결과를 향상시켰다. 앞으로 본 연구와 관련된 흥미 있는 연구과제는 본 연구에서 제시된 알고리즘을 적용한 본격적인 FCM 지식베이스 구현일 것이다. 여기에는 기존의 전문가시스템이 다루지 못했던 새로운 형태의 인과관계 지식을 수학적으로 보다 엄밀하게 제시할 수 있는 이론적인 전개가 선행되어야 할 것이다.

참고문헌

- [1] Axelrod, R. Structure of Decision : The Cognitive Maps of Political Elites, Princeton University Press, Princeton, New Jersey, 1976.
- [2] Burns, J.R. and Winstead, W.H., "M-Labeled Diagraphs: An Aid to the Use of Structural and Simulation Models", *Management Science*, vol. 21, no. 3, 1985, pp. 343-358.
- [3] Burns, J.R., Winstead, W.H., and Haworth, D.A., "Semantic Nets as Paradigms for Both Causal and Judgmental Knowledge Representation", *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 19, no. 1, 1989, pp. 58-67.
- [4] Deo, R., Graph Theory with Applications in Engineering and Computer Science, Prentice Hall, Englewood Cliffs, NJ, 1974.
- [5] Edden, C., Jones, C., and Sims, D., *Thinking in Organizations*. Macmillan Press Ltd., London, England, 1979.
- [6] Gotoh, K., Murakami, J., Yamaguchi, T., and Yamanaka, Y., "Application of Fuzzy Cognitive Maps to Supporting for Plant Control", (in Japanese) *SICE Joint Symposium of 15th Syst.*

- Sym. and 10th Knowledge Engineering Symposium*, 1989, pp. 99-104.
- [7]Jung, D. G. and Burns, J. R., 'Connectionist approaches to inexact reasoning and learning systems for executive and decision support', *Decision Support Systems*, No. 10, 1993, pp. 37-66.
- [8]Kim, H.S. and Lee, K.C., "An Improved Fuzzy Cognitive Map with Fuzzy Causal Relationship and Fuzzy Partial Causal Relationship", *Journal of Expert Systems* 1 (2), 33-55, 1995.
- [9]Kosko, B., 'Fuzzy cognitive maps', *International Journal of Man-Machine Studies*, 24, 1986, pp. 65-75.
- [10]Kosco, B., *Neural Networks and Fuzzy Systems: A Dynamical Systems Approach to Machine Intelligence*, Prentice Hall, Englewood Cliffs, NJ, 1992.
- [11]Lee, K.C., "A Unified Approach to Corporate Planning Simulator Using a Fuzzy Logic-Based Intelligent Model", *Korean Management Science Review*, 9 (2), 1992, 83-108.
- [12]———, "A Cognitive Map Knowledge-Based Strategic Planning Simulation", *Proceedings of 26th Hawaii International Conference on Systems Sciences*, Wailea (Hawaii), Vol. III, January 1993, 249-267.
- [13]———, and Kim, H. S., 'A Fuzzy Cognitive Map-Based Bi-Directional Inference Mechanism: An Application to Stock Investment Analysis', forthcoming in *International Journal of Intelligent Systems in Accounting and Management* 1998.
- [14]———, Chu, S.C., and Kim, H.S., "A Study on the Development of Multiple Experts' Knowledge Combining Algorithm by Using Fuzzy Cognitive Map", *Journal of the Korean Operations Research and Management Science Society*, vol. 19, no. 1, 1994, pp. 17-40.
- [15]Looney, C.G. and Alfize, A.A., "Logical Controls via Boolean Rule Matrix Transformations", *IEEE Transactions on Systems, Man, and Cybernetics*, vol. SMC-17, no. 6, 1987, pp. 1077-1082.
- [16]Montazemi, A.R. and Conrath, D.W., "The Use of Cognitive Mapping for Information Requirements Analysis", *MIS Quarterly*, 1986, pp. 45-56.
- [17]Papadimitriou, C. H. and Steiglitz, K., *Combinatorial Optimization: Algorithms and Complexity*, Prentice Hall, Englewood Cliffs, NJ, 1982.
- [18]Styblinski, M.A. and Meyer, B.D., "Fuzzy Cognitive Maps, Signal Flow Graphs, and Qualitative Circuit Analysis", *Proceedings of the 2nd IEEE International Conference on Neural Networks (ICNN-87)*, vol. II, 1988, pp. 549-556.
- [19]Taber, W.R., "Knowledge Processing with Fuzzy Cognitive Maps", *Expert Systems with Applications*, 2 (1), 1991, pp. 83-87.
- [20]Zhang, W. and Chen, S., "A Logical Architecture for Cognitive Maps", *Proceedings of the 2nd IEEE Conference on Neural Networks (ICNN-88)*, vol. I, 1988, pp. 231-238.

● 저자소개 ●



이건창

1982년 성균관대학교 경영학과 학사

1984년 한국과학기술원(KAIST)경영과학과 석사

1988년 한국과학기술원(KAIST)경영과학과 박사(전공:경영정보시스템)

현 재 성균관대학교 경영학부 정교수

관심분야 인공지능을 이용한 의사결정, 전자상거래, 가상기업, 웹 시뮬레이션

조형래

1981년 서울대학교 계산통계학과 학사

1983년 한국과학기술원(KAIST)경영과학과 석사

1989년 한국과학기술원(KAIST)경영과학과 박사

현 재 경상대학교 산업공학과 교수

관심분야 전자상거래, 시뮬레이션