

시간논리구조에서 이산사건시스템의 최적화 및 추론

Optimization and reasoning for Discrete Event System in a Temporal Logic Frameworks

황 형 수*, 정 용 만*

Hyung-Soo Hwang*, Yong-Man Jeong*

※이 논문은 '96학년도 원광대학교 교비지원에 의해서 연구됨

ABSTRACT

A DEDS is a system whose states change in response to the occurrence of events from a predefined event set. In this paper, we consider the optimal control and reasoning problem for *Discrete Event Systems(DES)* in the *Temporal Logic framework(TLF)* which have been recently defined. The *TLF* is enhanced with objective functions(event cost indices) and a measurement space is also defined. A sequence of event which drive the system from a given initial state to a given final state is generated by minimizing a cost function index. Our research goal is the reasoning of optimal trajectory and the design of the optimal controller for DESs. This procedure could be guided by the heuristic search methods. For the heuristic search, we suggested the Stochastic Ruler algorithm, instead of the A* algorithm with difficulties as following; the uniqueness of solutions, the computational complexity and how to select a heuristic function. This SR algorithm is used for solving the optimal problem. An example is shown to illustrate our results.

I. 서 론

이산사건시스템은 최근 많은 연구가 이루어지고 있으며 많은 대규모 동적 시스템들은 이산사건구조로 이루어져 있다. 이산사건시스템은 미리 정의된 사건 집합으로부터 사건의 발생에 따라 상태가 변화하는 시스템으로 제조공정의 조립라인, 통신시스템, 전문가시스템 등이 있다. 이산사건시스템을 제어하고 모델링 하는 방법은 Automata and formal Language

[1], Finitely Recursive Process, Mini-Max Algebra, Petri-Net 그리고 시간논리구조를 이용한 방법이 제안되었다. 가제어성, 가관측성, 감시제어 그리고 최적제어 등과 같은 많은 제어이론들이 이산사건 시스템에 대하여도 연구되었다. 특히 최적제어에 관한 여러 가지 개념들이 이산사건 시스템에 대하여도 연구되었다. Ramadge and Wonham[1]은 주어진 언어에 대한 제어 가능한 부언어의 최적성을 고려하였고, Sengupta and Lafontaine[5]는 최적제어 문제의 그래프-이론 형

* 원광대학교 제어계측공학과

식을 제안하였으며, Passino and Antsaklis[6]는 이산 사건시스템의 모든 상태천이의 비용을 고려한 최적화 방법에 대하여 연구하였다. 또한 Lin and Ionescu [7]은 최근 시간논리구조에서 이산사건시스템을 제어하기 위한 방법을 연구하였으며, A* 알고리즘을 이용하여 이산사건시스템의 최적화방법을 제시하였다. 아울러 이 논문에서 저자들은 해의 유일성, 계산의 복잡성 그리고 경험 함수의 선택의 문제 등 몇 가지 문제점을 제시하였다.

Lin & Ionescu에 의해서 제시된 최적화 방법에서 제어기는 주어진 초기상태로부터 최종상태까지 시스템이 경로에 따라 이동하도록 설계되어 졌으며, 초기 상태로부터 최종상태로 이동할 때 최적경로는 A* 알고리즘을 이용하여 주어진 비용함수(cost function)의 지수가 최소값을 갖도록 설계되었다.

본 논문은 지금까지 제시된 방법들 중 가장 우수한 방법으로 평가되는 Lin & Ionescu가 제시한 시간논리구조에서 이산사건시스템의 최적화 방법에서 A* 알고리즘을 이용함으로써 발생한 문제들을 해결하기 위하여 확률 분포에 기초한 Stochastic Ruler 알고리즘의 최적 Ruler값을 이용하여 최적 경로를 추론함으로써 해의 유일성, 계산의 복잡성 등의 문제점을 해결하고 제어기를 설계하기 위한 최적화 방법에 대하여 연구하였다.

논문의 2장은 시간논리 구조가 이산사건시스템의 제어기 설계 및 최적화를 위하여 소개되며, 3장에서는 최적 경로 추론에 의한 최적화 문제를 해결하기 위한 Stochastic Ruler 알고리즘, 4장에서는 예를 들었으며, 5장은 결론으로 구성하였다.

II. 시간 논리 구조

시간 논리는 시간 순차열에 따른 추론 지향적인 논리이며, 논리를 이용하고 시제 표현식으로 표현될 수 있는 두 가지 장점을 갖는 시간논리는 Allen's Properties-Event-Process[8]와 McDermott's Fact-Event[9]의 시제 추론을 이용하였다.

이 시간논리는 주로 Software 증명에 이용되었으며 최근에는 이산사건시스템의 제어문제에 적용되었다[3]. 시간논리는 시간의 개념을 포함시킨 일반 논리의 확장

이며 또한 시간논리는 종래의 부울 연결자 \neg (not), \wedge (and), \vee (or), \rightarrow (implies), \leftrightarrow (if and only if))와 시간의 변화와 시간의 양을 표현하기 위한 시제 연산자 \square (henceforth), \diamond (eventually), \circ (next), U (until), P (proceed) 와 같은 연결자와 연산자를 이용하여 시간과 시제관계에 관한 추론이 가능한 구조를 갖고 있다.

2.1 측정 공간에서 시간논리구조식

공정의 시간논리구조는 6-변수, 즉 $M = (S, E, F^*, f, s_0, l)$ 로 정의된다. 여기서 S 는 유한한 상태들의 집합이며, 그리고 E 는 유한한 사건들의 집합을 나타낸다. 각 사건들은 한 상태를 다른 상태로 천이 시킨다. s_0 는 초기상태, f 는 발화(firing) 함수이며 상태들의 집합 S 로부터 E^* (E^* 는 사건들의 집합 E 의 부분집합으로 각 원소들은 실행 가능한 사건들을 나타낸다.)에 대응시킨다. 즉 $\forall s \in S$ 그리고 $\forall e \in f(s)$ 일 때 $s' = f(s, e)$ 로 정의되며 $f(s)$ 는 상태 s 에서 발생되는 사건들의 집합을 나타낸다. l 은 사건들의 집합 E 로부터 F^* (F^* 는 논리식의 집합)로의 labeling 함수를 나타낸다.

발화함수 f 에 의하여 $e(s) = f(s, e) = s'$ 는 상태 s 로부터 다음상태 s' 의 천이를 나타낸다. 따라서 발화함수를 다음상태함수(next state function)라 부른다. 시간논리구조에서 한 상태는 정확하게 한 사건의 실행에 대응한다. 이러한 사실을 반영하기 위하여 한 사건의 발생의 결과를 나타내기 위한 식 $l(e_i, s_i)$ 는 다음과 같다.

$$\square [s_i(x) \wedge e_{i+1}(x) \rightarrow s_{i+1}(\circ x)]$$

여기서 $s_i(x)$ 는 원소 x 의 현재상태를 나타내며, $e_{i+1}(x)$ 는 x 에 대응하는 사건의 발생을 나타내며, $s_{i+1}(\circ x)$ 는 x 의 다음상태를 표현한다.

2.2 측정 공간에서 도달성

이 장에서는 시간논리구조에서 도달성을 소개한다. 그리고 상태들의 도달성과 시간논리구조식 사이의 관계를 규명하고 끝으로 도달성 그래프 작성과정을 소개한다.

임의의 한 상태 s 에서 한 사건 발생의 결과는 다음 상태 s' 로 천이를 의미한다. 즉 다음 상태 s' 은 이전 상태 s 로부터 즉각적인 도달을 의미한다.

시간논리구조 $M = (S, E, F^*, f, s_0, l)$ 에서 다음상태 s' 은 식 $f(e, s_0) = s'$ 를 만족하는 임의의 사건 $e \in Es_0$ 가 존재한다면 초기상태 s_0 로부터 바로 도달된다. 여기서 $e \in Es_0$ 는 초기상태 s_0 로부터 발생할 수 있는 사건의 집합을 나타낸다. 이 개념은 주어진 시간논리구조 M 에 대하여 도달 가능한 상태들의 집합을 정의하도록 확장할 수 있다. 만일 상태 s' 이 초기 상태 s_0 로부터 즉각적으로 도달되고 상태 s'' 은 상태 s' 으로부터 즉각 도달되어 진다면 그때 상태 s'' 은 초기 상태 s_0 로부터 도달 가능하다고 할 수 있다.

우리는 초기상태 s_0 로부터 도달될 수 있는 모든 상태들로 이루어진 임의의 시간논리구조 M 의 도달성집합 $R(M, s_0)$ 을 정의한다. 즉 $s_0 \in R(M, s_0)$ 이며, 만일 $s \in R(M, s_0)$ 이고 $e \in Es$ 에 대하여 $s' = f(e, s)$ 이 성립하면 그때 역시 $s' \in R(M, s_0)$ 이 성립하며 결국 $R(M, s_0)$ 은 이들에 의해서 정의된 가장 적은 상태들의 집합이다. 한 상태와 사건들의 열을 새로운 상태로 대응시키는 발화함수(다음상태함수)는 편리하게 확장될 수 있다. 다음상태함수가 임의의 상태 s 로부터 다음 상태 s' 로 확장되어 졌을 때 사건들의 열 $E_s \in \sigma$ 와 $\forall e \in E$ 은, 즉, $\sigma e \in E_s$ 에 의하여 $f(\sigma e, s) = f(e, f(\sigma, s)) = s$, $f(\epsilon, s) = s$ 가 된다. 여기서 σ 는 모든 사건들을 의미하고 ϵ 은 공사건을 의미하며 E_s 는 상태 s 에서 사건이 발생할 수 있는 사건들의 집합을 의미한다.

도달성은 일반적으로 확장된 다음상태함수를 사용한다. 사건들의 열 e_1, e_2, \dots, e_k 와 상태 s 에 대하여 다음상태 $s' = f(e_1, e_2, \dots, e_k, s)$ 은 먼저 e_1, e_2 등 e_k 가 발생할 때 까지의 결과이다[3]. 따라서 시간논리구조 $M = (S, E, F^*, f, s_0, l)$ 이 주어졌을 때 만일 상태 $s' = f(e, s)$ 가 $e \in E_s$ 에 대하여 상태 s 로부터 도달 가능하며, 상태 s_0 에서 상태 s 로 변화시킬 수 있는 사건 발생이 존재하고, 동적 방정식 $l(e, s) \in F$ 가 참이면, 상태 s 는 집합 $R(M, s_0)$ 에 존재한다.

이것은 천이 대응 f 와 Labeling 함수 l 의 관계가 성립함을 나타내며, 시간논리구조에서 상태들의 도달성과 식의 유효성 사이의 관계를 입증한다. 이것은 상태들의 도달성이 대응하는 동적 방정식의 유효성과 동가라는 것을 의미한다. 그리고 시간논리구조에 대한 도달성 그래프는 이러한 관계를 입증하는데 도움을 줄 것이며, 도달성 그래프는 도달 가능한 상태

를 나타내기 위한 상태 값의 절점과 천이 상태를 나타내기 위한 사건을 나타내는 모서리로 구성된다. 사건의 비용을 측정하기 위하여 우리는 시간논리구조와 관련이 있는 측정공간을 정의하기 위한 측정함수를 소개하여야 한다. 사건비용함수 $\theta: E \rightarrow R(\theta(\bar{e}_{ij}) = \theta(s_i, s_j))$ 이다. 여기서 $\bar{e}_{ij} \in Es_i$ 이며, $s_i = f(\bar{e}_{ij}, s_j)$ 이다. 여기서 R 은 도달성 그래프를 말하며, \bar{e}_{ij} 는 상태 s_i 와 s_j 사이 호를 나타낸다. 사건비용함수는 사건 $\bar{e}_{ij} \in Es_i$ 가 성립해야 정의된다. 분명히 $\theta(\epsilon) = 0$ 이며, 여기서 ϵ 은 영 사건을 의미한다. 만일 $e_{1k} = e_1 e_2 \dots e_k$ 면 $\theta(\bar{e}_{1k}) = \sum_{i=1}^n \theta(e_i)$ 이다. 따라서 우리는 시간 논리구조 M 에 비용함수 θ 를 추가함으로써 개선된 시간논리구조 $M' = (S, E, F^*, f, s_0, l, \theta)$ 을 얻는다.

사건의 발생은 수행되어진 행동을 나타내며, $\theta_{ij} = \theta(\bar{e}_{ij})$ 은 사건 e_{ij} 의 발생에 의하여 상태 s_i 로부터 상태 s_j 로 진행된 비용 측정을 나타낸다. 개선된 시간논리구조의 실행은 상태들의 열, 사건들의 열 그리고 사건들의 비용의 열 등의 세 순열을 얻게 된다. 이것은 도달성 그래프에 의하여 분명하게 표현된다. 이 그래프는 시간논리구조의 그래프적인 표현의 확장이며 사건 비용함수의 사양에 따른 사건 발생에 대한 비용의 사양을 포함한다. 그러한 비용은 사건의 발생에 관련된 행동의 실행에서 소모된 자원의 측정을 나타낸다. 다음 장에서 Stochastic Ruler 알고리즘을 이용하여 이산사건시스템의 최적제어 문제를 다루기 위하여 앞에서 다른 세 순열을 정의한다.

◆ $S = \{s_0, s_1, \dots\}$ 은 상태들의 집합이다.

◆ $E = \{\bar{e}_{ij} | s_j = f(\bar{e}_{ij}) | s_i \in S\}$ 는 상태 s_i 로부터 s_j 를 향하는 사건 e_{ij} 에 대한 유향호(Directed arcs)의 집합이다.

◆ $R = \{\{\theta_{ij} | \theta_{ij} = \theta(\bar{e}_{ij}), \bar{e}_{ij} \in E\} \cup \{\theta_j | \theta_j = \theta(s_j), s_j \in S\}\}$.

여기서 θ_{ij} 는 상태 s_i 로부터 상태 s_j 사이의 각 호에 관련된 사건 비용함수의 집합이며, $\forall \theta_{ij} \in R$ 에 대하여 $\theta_{ij} \geq 0$ 이다. 여기서 θ_j 는 균일하게 분포된 임의 변수로서 Stochastic Ruler를 나타낸다. $\theta_j, \theta_j \geq 0$ 는, 현재상태 s_i 로부터 존재할 수 있는 다음 상태들 $s_j (j = 1, 2, \dots, i = j)$ 에 관련된 사건비용함수의 집합이며 이 함수는 SR알고리즘을 이용한 모의실험에 의해 얻어진다.

III. 이산사건시스템의 최적제어 문제

제어이론에서 최적화는 제어기 합성 및 최적제어에서 중요한 부분이다. 이 장에서는 시간논리구조에서 이산사건시스템의 최적경로에 관한 최적제어이론을 정립한다. 이 과정은 경험적 탐색 방법에 의하여 얻어진다. 최적경로를 추적하기 위하여 A^* 알고리즘을 이용한 연구가 이루어졌다. 그러나 A^* 알고리즘을 사용했을 때 발생하는 해의 유일성, 계산의 복잡성 그리고 경험함수의 선택 등의 문제점들이 [7] 발생했으며, 이 문제점들을 해결하기 위하여 본 논문에서는 A^* 알고리즘 대신에 Stochastic Ruler 알고리즘[11]을 사용한다.

3.1 플랜트와 제어기의 합성

우리가 앞에서 소개한 개선된 시간논리구조 $M' = (S, E, F^*, f, s_0, l, \theta)$ 은 발생되어진 사건들에 대하여 선택되어 질 수 있는 다음상태에 대한 비용함수를 정의하였다.

플랜트에서 주어진 초기상태 s_0 와 최종상태 s_f , 제어기의 초기상태 c_0 에 대하여 개선된 시간논리구조를 이용하여 플랜트와 제어기는 $M'_p = (S_p, E_p, F_p^*, f_p, s_0, l_p, \theta_p, s_f)$ 과 $M'_c = (S_c, E_c, F_c^*, f_c, c_0, l_c)$ 에 의하여 각각 모델링 되었다. 우리는 플랜트 M_p 에서 사건과 제어기 M_c 에서 사건을 동기화 시킴으로써 구성한 폐루프 이산사건시스템을 정의하였다. 폐루프 시스템 M_{cls} 는 $M'_p \parallel M'_c = (S_p \times S_c, E_p, F_p^* \cup F_c^*, f_{cls}, (s_0, c_0), (l_p, l_c), \theta_{cls})$ 처럼 정의될 수 있다.

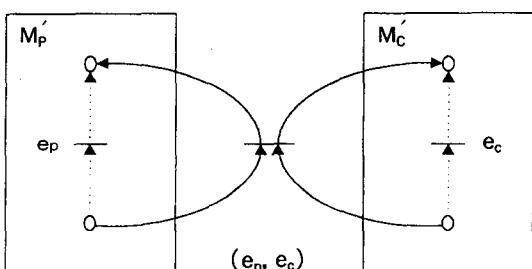


그림 1. M'_p 과 M'_c 에서 두 사건의 동기화

Fig. 1 Synchronization of two event in M'_p and M'_c

일반적으로, 폐루프 시스템 M_{cls} 는 시스템의 요구

되는 동작을 지정하기 위하여 주어졌다. 그림 1은 M'_p 과 M'_c 으로부터 두 사건의 동기화를 나타낸다. 그러므로 같은 방법으로 구성된 제어기는 초기상태 s_0 로부터 최종상태 s_f 까지 요구되는 경로 혹은 케도를 따라 플랜트가 구동되도록 일련의 사건들의 열이 발생한다.

이 경우에 많은 경로가 존재하게 되는데, 최적화란 제어기가 초기상태 s_0 로부터 최종상태 s_f 까지 최적의 경로 혹은 총 사건 비용지수가 최소 값이 되도록 플랜트를 구동하도록 일련의 사건들의 열이 발생할 수 있도록 하는 것이 요구된다. 최소의 사건비용지수를 갖는 최적경로를 찾기 위하여 다음에 소개될 Stochastic Ruler 알고리즘이 사용된다.

3.2 최적 목적함수

플랜트의 시간논리구조 M 에서 최적경로($\sigma \in S^*$)를 결정하는 것은 어려운 최적화 문제이다. 즉 $g(s) \leq g(s')$, $\forall s \in S^* \subset S$ 를 만족하는 최적 상태 집합 S^* 를 구하는 것이다. 여기서 $S = \{s_1, \dots, s_k\}$ 는 σ 에서 실행 가능한 상태들의 집합이며 $g(s)$ 는 상태 s 에 대응하는 목적함수이다. 목적함수 $g(s)$ 가 분리 가능할 때, $g(s)$ 가 $g(s) = \sum_{i=1}^n g_i(s^{(i)})$ 로서 쓰여질 수 있다. 여기서 $s = (s^1, \dots, s^n)$ 이다. 그러나 본 논문에서의 이산사건 시스템의 최적제어 문제는 분리될 수 있는 목적함수가 아니며 또한 탐색 알고리즘은 측정공간이라 불리는 상태공간에서 어떻게 다음 상태를 탐색할 것인가에 의하여 특성 지어진다. 탐색 방법의 대부분은 branch-and-bound 알고리즘과 simulated annealing 알고리즘이다. [10] 이 두 알고리즘은 목적함수 $g(s)$ 의 해석적 표현이 요구된다. 우리가 지적한 것처럼 고려되어진 목적함수는 해석적인 형태로도 표현될 수 없으며, 단지 목적함수 $g(s)$ 는 경험적인 데이터나 혹은 이산사건 모의실험[11]에 근거를 두고 평가되어 질 수 있다.

특히 우리는 상태 s_0 로부터 시작해서 상태 s_k 에서 끝나는 경로에 대하여 먼저 생각한다. $M(s_0, s_k)$ 는 초기상태 s_0 에서 시작해서 s_k 에서 끝나는 시간논리구조 M 의 모든 유한 경로($\sigma = s_0, s_1, \dots, s_k$)의 집합을 의미한다. 우리의 목적은 목적함수를 최소화하도록 최적 경로 σ^* 를 구할 수 있는 열을 구하는 것이다. 즉 $g(s)$

$= \min\{g(s) | s \in S\}$, 여기서 실행 가능한 집합 $S = \{s_1, s_2, \dots, s_k\} = \{s_i | i \in I\}$ 와 지수 집합 $I = \{1, 2, \dots, k\}$ 는 유한하며 그리고 목적함수 g 는 $g: S \rightarrow IR$ 이다. 여기서 문제의 중요한 특징은 실행 가능한 해의 집합이 이산특성을 갖고 있는 것이다. 그러나 s 의 크기가 임의의 크기로 만들어 질 수 있고 그리고 이론적 해석이 편리하다고 할지라도 s 의 유한성은 실제 용용에서의 제한 때문에 고려 될 수 없다.

최적제어기 합성 문제는 경로 $\sigma^* \in M(s_0, s_k)$ 를 따라 시스템 M 을 구동하는 상태들의 열을 얻을 수 있는 제어기를 찾는 것이다. 그러한 상태들의 열을 최적경로(σ^*)라 부른다. 그러한 상태들의 열 σ^* 이 결정되면 상태열은 경로 σ^* 을 따라 플랜트 M 을 구동시킬 수 있다.

3.3 Stochastic Ruler알고리즘을 이용한 제어기합성의 최적화

시간논리구조에서 제어기의 설계는 일반적으로 목적함수 값을 최소화 할 수 있는 상태 열에 대응 하는 경로를 찾기 위한 상태 탐색공간을 포함한다. 여기서 본 논문은 Stochastic Ruler 알고리즘을 이용한 경험적 탐색 이론을 이용한다. 따라서 우리는 위의 문제에 대한 부분적인 해 보다는 전체적인 해를 찾는다. 본래 이산사건 시스템의 최적화문제는 다음형태의 전체적인 최적 집합을 찾는 것이다[11]

$$\sigma \in S^* = \{s \in S | g(s) \leq g(s') \in S\}$$

여기서 $g(s)$ 는 상태 s 에서 시스템의 목적함수이다.

이산사건시스템에서는 $g(s)$ 는 분석적 수식이 불가능 하므로, 우리의 연구에서는 $g(s)$ 를 $H(s)$ 라는 추정값으로 얻는다. 즉, $g(s) = E[H(s)]$. 여기에서 $H(s)$ 는 주어진 상태 s 에 대한 목적함수값들의 평균치로 정한다. $\theta(a, b)$ 는 a 와 b ($a < b$) 사이에 일양분포(uniform distribution) 내의 임의 값을 나타내며, 이 값을 Stochastic Ruler라고 한다. 이 값의 범위는 $g(s)$ 의 상한과 하한 경계값으로 각각 $a(s)$ 와 $b(s)$ 이다. 따라서 최적화 문제를 해결하기 위해서, 위에서 정의한 $H(s)$ 과 $\theta(a, b)$ 를 이용하여 다음과 같은 최적화 조건식을 유도한다. 즉, $\Sigma^* = \{s \in S | P(s, a, b) \geq P(s', a, b), \forall s' \in S\}$, 여기서 $P(s, a, b) = P[H(s) \leq \theta(a, b)]$ 이다. 이 조건을 적용한 SR 알고리즘이 구성된 상태들의 집합 Σ^* 는 $\sigma \in S^*$ 에 수렴함을 보인는데는 어렵지 않다. 이를 세부적으로 살펴보면, SR 알고리즘은 이미 지정된 초기상태 $s_0, s \in S$ 에서 출발한다. 그리고 상태 s 의 이웃이라 불리는 이웃 상태 집합 $N(s) (N(s) \in S)$ 를 정의한다. 우리는 $N(s)$ 에서 상태중의 임의로 상태 하나 s' 을 선택한다. 그리고 이웃 상태중의 한 상태인 s' 에서 목적함수 $g(s')$ 의 여러 개의 표본값 x_1, x_2, \dots, x_n 을 끌어낸다. 이들 값들은 구간 $[a, b]$ 에서 균일하게 분포된 상태들로부터 얻어진 표본치 y_1, \dots, y_n 과 비교한다. 만일 $x_i \leq y_i, \forall i=1, \dots, n$ 이면 탐색은 새로운 상태 s' 으로 이동하고 그려지 않으면 현재상태 s 에 머물러 있게 된다. 이렇게 하면 목적함수의 표본치가 Stochastic Ruler의 표본치 보다 적을 확률이 최대치가 될 것임을 직관적으로 알 수 있다. 이 Ruler는 모든 상태에 대하여 같은 임의변수이므로 이 과정은 목적함수가 최소 값이 될 때까지 계속 진행되어야 한

- Data : $N, R, \{M_k\}, a, b, s_0 \in S$.
Step 0 : Set $X_0 = s_0$ and $k = 0$.
Step 1 : Given $X_k = s$, choose a candidate Z_k from $N(s)$ with distribution $P[Z_k = s' | X_k = s] = R(s, s')$, $s' \in N(s)$.
Step 2 : Given $Z_k = s'$, set

$$X_{k+1} = \begin{cases} Z_k & \text{with probability } P_{SR}(k) \\ X_k & \text{with probability } (1 - P_{SR}(k)) \end{cases}$$
where $P_{SR}(k) = \{P[H(s') \leq \theta(a, b)]\}^{M_k}$.
Step 3 : Set $k = k + 1$, and go to Step 1.

그림 2. Stochastic Ruler 알고리즘

Fig. 2 Stochastic Ruler Algorithm

다. 이 알고리즘을 정형화하기 위하여 몇몇 정의가 필요하다. 만일 현재 상태가 s 이고 다음상태가 s' 이라면 $R(s, s')$, ($s, s' \in S$)는 미리 지정한 확률이며, 그리고 R 은 대칭이어야 한다. 즉, $R(s, s') = R(s', s)$ 이다. 그리고 M_k 는 $M_{k+1} > M_k$ 이며 $M_k \rightarrow \infty$, ($k \rightarrow \infty$)을 만족하는 무한 비감쇠 순열이다.

SR알고리즘은 그림 2와 같다.

위의 알고리즘의 step 2의 구현은 다음과 같다. $Z_k = s'$ 이 주어졌을 때 $H(s')$ 로부터 표본치 $\{h_i(s')\}_{i=1, M_k}$ 가 되도록 M_k 를 끌어내고, 다음은 $\theta(a, b)$ 로부터 표본 $\{\theta_i\}_{i=1, M_k}$ 가 되도록 M_k 를 유도한다. 만일 $h_i(s') \leq \theta_i$ ($i = 1, 2, \dots, M_k$)이면 $X_{k+1} = Z_k = s'$ 으로 하고, 그러치 않으면 $X_{k+1} = X_k$ 로 한다. 여기서는 영이 아닌 가장 적은 $R(s, s')$ 을 구한다; $\rho = \min_{s \in S} \min_{s' \in N(s)} R(s, s')$ ($\rho > 0$). 그리고 가장 적은 $P(s, a, b)$ 를 구한다; $\mu(a, b) = \min_{s \in S} P(s, a, b)$ ($0 < \mu(a, b) < 1$).

그리고 실수 c 와 σ 를 정의한다. c 와 σ 각각 $1/r \leq c$ ($c > 0$) 그리고 $\sigma \geq 1/(\mu(a, b))$, 여기서 r 은 상태 s 로부터 s' 까지의 최소 길이의 경로를 나타낸다. 본 논문에서 상태간의 거리가 균등하다고 가정하여 r 의 값은 1을 갖는다. 정수 k_0 는 $1 \leq \text{clog}_\beta(k_0 + 1)$ 를 만족하도록 한다. 만일 $M_k = [\text{clog}_\beta(k + k_0 + 1)]$ 가 몇몇 정의 실수 c, β, k_0 에 대하여 성립한다면 알고리즘이 구한 최적 제어 확률은 점진적으로 증가한다[11].

SR 알고리즘은 비교될 수 있는 두 상태 s, s' 을 고려한다. 그리고 상태 s 와 s' 의 목적함수의 실질적인 의미는 각각 $P(s)$ 와 $P(s')$ 으로 나타내어진다. 그 때 만일 $|P(s) - P(s')|$ 가 크다면 추정값의 차로 미루어 더 나은 성능을 제공하는 상태를 추정하기가 쉬워진다.

반대로 $|P(s) - P(s')|$ 가 적다면 정확한 평가를 하는 것은 불가능하다. 그러나 두 상태들 사이에서 어느 구성상태를 선택하더라도 성능면에서 크게 차이는 없으나 부정확한 추정을 할 가능성이 있으며, 부정확한 추정은 반복계산에 의해서 정확한 상태를 추정한다.

IV. 예 제

제시된 방법의 예를 보이기 위하여 8-퍼즐이라 불리는 인공지능의 전형적인 실례를 들어 고찰해 본다. 8-퍼즐은 9 셀(cell)의 판이 있고, 셀에는 8개의 타일과 한 개의 빈 셀이 있다. (그림3) 타일은 1에서 8까지 번호가 붙여져 있고, 셀(cell)에는 1에서 9까지의 번호가 붙어 있다. 어떤 인접한 셀이 비어있다면, 타일은 한 셀에서 비어있는 다른 셀로 이동할 수 있다. 예를 들어, 그림 3에서 보인 바와 같이 타일 6은 셀 9를 비우고 셀 6으로 이동할 수 있다. 게임은 그림 3-a)에서 보인 것처럼 임의의 초기상태에서 시작한다. 타일 이동의 적절한 순서는 최소한의 이동으로 그림 3-d)에 보인 최종상태에 도달하도록 제어기에 의해 이루어 져야 한다.

함수 f 의 맵핑(mapping) 예는 그림 3-b), c)에서 다음과 같이 보여진다. s_i 는 그림 3-c)에서 구성의 상태라 하고, e_{ij} 는 셀 i 에 있는 타일이 셀 j 로 이동하는 사건이라 하자. 그러면 $s_j = f(e_j, s_i)$ 는 구성의 상태이다. 여기서, 어떤 타일이 셀 y 에서 z 로 이동 할 때 y 와 z 가 셀 이동 수라면 $e_j = \bar{e}_{yz}$ 이다. 우리는 $s_j = \{s_j \in S | P(s_j, a, b) \geq P(s_i, a, b) \forall s_i \in S\}$ 의 최적 해에 대해 SR 알고리즘을 설명한다. 여기서, $P(s_i, a, b) = P[H(s_i)$

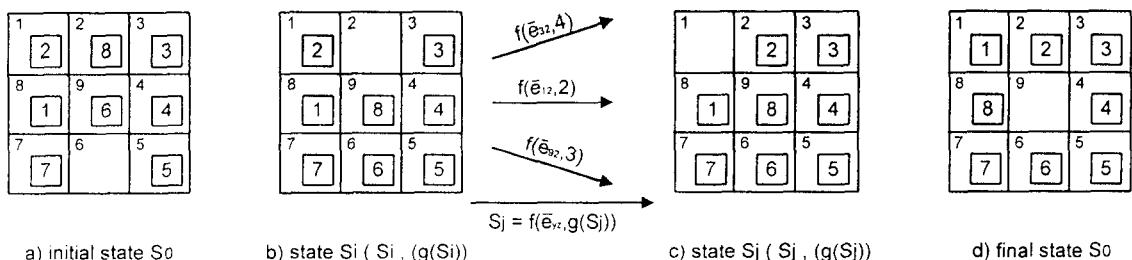


그림 3. 8-퍼즐과 f 함수의 맵핑

Fig. 3 The 8-puzzle and the mapping of function f

$\leq \theta(a, b)$]이다.

x 를 타일에 붙여진 번호를 표시하는 지역변수라 하고, y 와 z 를 셀에 붙여진 번호를 표시하는 지역변수라 하자. 그러면 다음과 같은 속성을 수반한다.

- IN(x, y) – 타일 x 는 셀 y 에 있다.
- EM(z) – 셀 z 는 비어있다.
- EQ(x, y) – x 에 붙여진 숫자는 y 와 같다.
- ADD1(n) – 수 n 은 1이 증가된다.
- MIN1(n) – 수 n 은 1이 감소된다.

어떤 다른 속성은 이 예제에서 다루어지지 않는다. 그리고 가능한 사건은 다음과 같다.

- move(x, y, z) : 타일 x 가 셀 y 에서 y 에 인접한 셀 z 로 이동.

플랜트에서 발화규칙은 다음에 나오는 시간논리구조에 의해 주어진다.

$$\square [IN(x, y) \wedge EM(z) \wedge move(x, y, z) \Rightarrow \bigcirc (IN(x, z) \wedge EM(y))].$$

위의 서술을 기초로 제어기 설계를 다음과 같이 주어진다.

$$\square [move(x, y, z) \rightarrow \bigcirc [(EQ(x, z) \wedge MIN1(g(s_i))) \vee (\neg EQ(x, z) \wedge ADD1(g(s_i)))]].$$

여기서, $g(s)$ 는 그들의 위치나 각 상태의 목적함수 밖에 있는 타일의 수를 나타내는 지역변수이고 s_i 와 s_j 에 대하여 $g(s_i) = 3$ 과 $g(s_j) = 2$ 이다. $g(s_i)$ 에 대한 상태 s_i 와 $g(s_j)$ 에 대한 상태 s_j 그리고 페루프 시스템에서 맵핑(mapping) f 는 위의 그림 3-b), c)에서 보였다.

본 연구에서는 8-퍼즐 환경에 대해 SR 알고리즘을 이용하여 최종상태 및 최종상태에 도달하기 위한 이동의 최적 경로를 그림 4와 같이 보일 수 있다. 본 모의실험에서 SR 알고리즘의 환경변수와 초기값은 다음과 같다: $\theta = [1, 10]$ 간의 일양분포(uniform distribution)의 값으로 $a=1$, $b=2$ 이며, 최적화를 위한 가장작은 $P(s, a, b)$ 을 찾기 위한 무한증가변수인 $M_k = k/20$ 으로

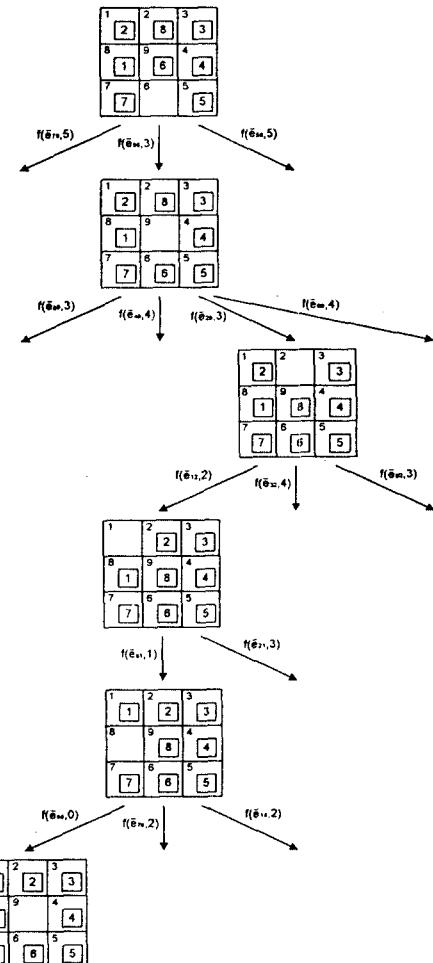


그림 4. 8-퍼즐의 탐색 구조
Fig. 4 A search tree of the 8-puzzle

두고, 초기상태 s_0 의 목적함수인 $g(s_0)$ 는 4(이값은 초기퍼즐 상태에서 셀과 타일의 번호가 다른것들의 수를 나타낸다.) 인접상태의 수를 나타내는 N 은 최대 4로 두었다. 그림 5에서 보여진 모의실험의 결과로부터 사건의 최적 제어순서는 $move(6, 9, 6)$, $move(8, 2, 9)$, $move(2, 1, 2)$, $move(1, 8, 1)$, $move(8, 9, 8)$ 이다. 이 그림으로부터 우리는 최종단계가 두 상태($move(1, 8, 1)$ 과 $move(8, 9, 8)$, 또는 $move(8, 9, 8)$ 과 최종상태)사이에서 진동하는 것을 볼 수 있다. 이런 진동은 더 좋은 목적함수 $g(s_f)$ 로 최종상태 s_f 를 찾기 위한 반복적인 절차를 나타내고 있다.

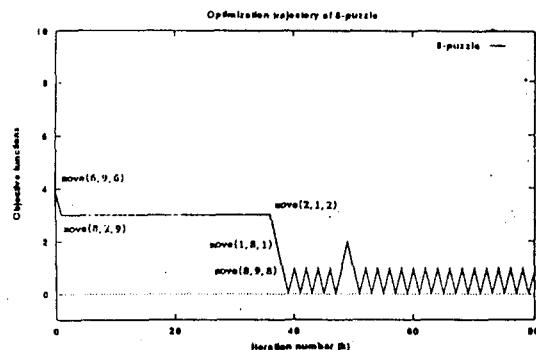


그림 5. 8-퍼즐의 최적화 궤도

Fig. 5 The optimization trajectory of 8-puzzle

이것은 최적제어 문제인 DES의 최소 사전비용 제어문제의 실례이며, 우리가 제안한 SR 알고리즘은 이산사건동적시스템의 최적경로의 추론과 최적화에 적합한 방법임을 나타낸다.

V. 결 론

본 논문은 시간논리구조에서 이산사건 시스템의 최적경로의 추론 및 최적화 방법에 대하여 연구하였다. 지금까지 연구되어진 이산사건 시스템의 최적경로의 추론 및 최적화 방법들은 주어진 초기 상태로부터 최종상태까지 시스템의 상태 변화에 대하여 주어진 비용함수가 최소값을 갖도록 설계되었다. 제시된 방법들중에서 시간논리구조에서 A* 알고리즘을 이용한 방법이 가장 우수한 방법으로 평가되나, 이 방법은 해의 유일성, 경험함수의 선택, 계산의 복잡성 등의 몇 가지 문제점들이 노출되었다. 따라서 본 논문에서는 개선된 시간논리 구조를 제안하여 이산사건 시스템에 대한 최적화 문제를 수식화 했다. 이를 이용하여 A* 알고리즘을 이용한 방법에서 문제가 되는 해의 유일성, 경험함수의 선택, 그리고 계산의 복잡성 등의 문제를 Stochastic Ruler 알고리즘을 이용하여 해결하였다. Stochastic Ruler 알고리즘을 이용한 이 방법은 일양분포를 이용하는 통계적인 방법이기 때문에 경험함수를 선택할 필요가 없으며, 또한 유일한 해가 존재하는 우수한 방법이다. 이 결과를 보이기 위해 8-퍼즐이라 불리는 인공지능의 전형적인 예를 들어 모의 실험을 하였으며, 실험결과 초기상태로부터

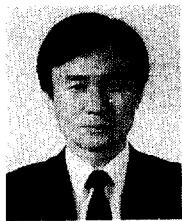
터 최종상태까지의 유일한 최적경로가 얻어짐을 보였다.

참 고 문 현

- P. J. Ramadge and W. M. Wonham, "Supervisory control of a class of discrete event process" SIAM J. Control. Optim., Vol.25, no.1, pp.206-230, 1987.
- Cuneyt M. Ozveren and Alan S. Willsky, "Observability of discrete event by systems" IEEE Trans. on Automatic Control. Vol.35.NO.7, July 1990.
- D. Ionescu, J. -Y. Lin, and H. S. Hwang, "Specification of Intelligent controllers for discrete event system in a temporal logic framework" IFAC workshop on Algorithms and architecture for real time control, Seoul, Korea Aug. 1992. pp. 237-242.
- H.S. Hwang, S.C. Joo and D. Ionescu, "The Controller modeling using the temporal logic model in Discrete Event Dynamic Systems" Journal of the KISS, Vol. 21, NO.9 1994.9, pp 1665-1674.
- R. Sengupta and S. Lafontaine, "Optimal control of a class of discrete event systems", IFAC Symposium on Distributed Intelligence Systems, Arling, Virginia, August 13-15, 1991, pp.25-30.
- K. M. Passino and P. J. Antsaklis, "On the optimal control of discrete event systems" Proc. 28th IEEE Conf. Decision and Control, Tampa, Florida, Dec. 1989, pp.2713-2718.
- J. -Y. Lin and D. Ionescu, "Optimization of controller design for discrete event systems in a temporal logic framework" 1992 American Control conference, pp.2819-2823, 1992.
- J. F. Allen, "Toward a general theory of action and time" Artificial Intelligence, Vol. 23, pp. 123-154, 1984.
- D. McDermott, "A temporal logic for reasoning about processes and plans" Cognitive Science, 6, pp.101-155, 1982.
- N. J. Nilsson, Principles of Artificial Intelligence.

Palo Alto, California, 1980.

11. D. Yan and H. Mukai, "Stochastic Discrete Optimization," *SIAM Journal on Control Optimization*, 1992.



황 형 수(Hyung-Soo Hwang)정회원

1955년 8월 2일

1974년 3월~1981년 2월: 전북대학교 전기공학과 졸업

1981년 3월~1983년 8월: 전북대학교 대학원 전기공학과 졸업(석사)

1984년 3월~1987년 8월: 전북대학교 대학원 전기공학과 졸업(박사)

1988년 3월~1997년 현재: 원광대학교 제어계측공학과 전임강사, 조교수, 부교수

1991년 8월~1992년 7월: Dept. of Electrical Eng., University of Ottawa, Canada(Post-Doc.)

※ 연구관심분야: 퍼지 및 지능제어, 이산사건시스템 제어, 실시간 제어시스템, 제어이론 등임



정 용 만(Yong-Man Jeong) 정회원

1973년 4월 27일

1992년 3월~1996년 2월: 원광대학교 제어계측공학과 졸업

1996년 3월~1997년 3월: 원광대학교 대학원 제어계측공학과 재학 중
연구관심분야: 퍼지 및 지능제어, 이산사건시스템 제어, 제어이론 등임