

페트리네트를 기반으로 한 통합 소프트웨어의 개발

Development of a Petri-net based Integrated Software

황상철*, 박찬우*, 이효성**

Sang-Chel Hwang, Chan-Woo Park, Hyo-Seong Lee

* 경희대학교 기계·산업시스템 공학부 대학원생

** 경희대학교 기계·산업시스템 공학부 교수

요약

본 연구에서는 이산사건 동적 시스템을 모델링하고 성능을 분석하기 위한 통합 소프트웨어를 개발하였다. 소프트웨어는 페트리네트를 기반으로 비쥬얼 C++와 비쥬얼 베이직을 이용하여 개발되었으며 윈도우즈 환경 하에서 동작하는 범용 그래픽도구를 내장하고 있다. 본 소프트웨어를 이용하면 사용자는 마우스의 드래그 앤 드롭만으로 페트리네트 모델을 작성할 수가 있으며 모델이 작성되면 수리적 분석과 시뮬레이션을 통하여 시스템의 성능분석을 자동으로 수행하여 준다. 또한 시스템의 논리적인 특성을 분석하기 위하여 coverability tree와 reachability tree를 지원한다. 이외에도 deadlock을 탐지하고 이를 해소하는 기능과 정교한 세부 모델링이 가능하도록 다양한 활성화 함수와 분포를 지원하며 시스템의 성능척도와 관련된 다양한 통계치를 제공해준다.

1. 서론

생산시스템이나 통신시스템과 같은 이산형 사건 동적시스템을 분석하는 방법에는 수리적기법과 시뮬레이션이 있다. 수리적기법은 빠르고 정확한 분석이 가능하며 일단 분석이 이루어지고 나면 시스템의 중요 특성치간에 관계규명이 쉽다는 장점이 있다. 그러나 적용가능한 시스템이 매우 한정되어 있고 분석을 위해서는 전문적인 지식이 필요하다는 단점을 가진다. 반면에 시뮬레이션은 복잡한 시스템도 분석이 가능하지만 시스템에 적합한 모델을 개발하는데 드는 비용과 시간이 많이 소요되며 일단 시스템에 맞는 모델을 작성하고 나면 시스템의 변화를 모델에 반영하기가 어렵다는 단점이

있다.

이와 같은 이유로 수리적분석과 시뮬레이션이 모두 가능한 분석도구의 개발이 필요하며 페트리네트는 이를 가능하게 하는 유용한 대안이다. 페트리네트는 이산형 사건 동적 시스템에서 정보와 제어의 흐름을 모형화하는 데 있어 적합한 네트워크 모델로서 다음과 같은 여러가지 장점을 가지고 있다[6].

- 우선성 (priorities), 동기성(synchronization) 봉쇄(blocking) 등과 같은 시스템의 특수한 행태(behavior)를 정확하고 용이하게 표현할 수 있는 강력한 모델링 능력(modeling

power)을 구비하고 있다.

- Overflow의 발생가능성(boundness), deadlock의 발생가능성(liveness)등 시스템의 논리적 특성 규명을 가능하게 한다.
- 추계적 페트리 네트 등으로 표현 가능한 일부 시스템의 경우에는 시스템의 수리적 성능 분석을 수행할 수 있다.
- 수리적분석이 불가능한 경우에도 페트리 네트 모형을 이용해 모의실험코드를 자동적으로 생성할 수 있으므로 모의실험을 통한 시스템의 성능분석을 가능하게 해준다.
- 시스템의 설계단계에서 이용될 수 있을 뿐 아니라 시스템이 구축된 후에도 시스템의 상태를 감시(monitor)하고 복잡한 활동을 제어하는 제어기로서의 역할을 수행할수 있어 시스템의 전생애(life cycle)를 통해 활용되어질 수 있다.

이와 같은 페트리네트의 장점 때문에 외국에서는 Stroboscope(美), Visual SimNet(獨), Visual Object Net과 같은 여러 소프트웨어가 이미 개발되었다. 그러나 이러한 소프트웨어들은 사용자 편의성이라든가 기능의 제약 때문에 시스템 분석에 적용하기에는 한계가 있다. 국내에서는 저자들에 의해 페트리 네트를 기반으로 한 시뮬레이션 소프트웨어가 개발된 바 있다[1]. 본 연구에서는 기존에 저자들에 의해 개발된 소프트웨어의 몇 가지 문제점을 개선하고, 시뮬레이션 외에 수리적분석이 가능하도록 소프트웨어의 기능을 보완한 새로운 통합 소프트웨어를 개발하였다. 새로이 개발된 소프트웨어의 중요 기능 및 특성은 다음과 같다.

첫째, 그래픽도구를 이용한 시스템 모델링 기능에 자동오류검사(Error Check) 기능, 속성창(Property Window)를 통한 데이터 입력기능 등을 부여하여 사용자의 편의성 및 소프트웨어의 유연성을 극대화한다.

둘째, 추계적 페트리 네트 등으로 표현 가능한 일부 시스템의 경우에는 페트리 네트에 내재되어있는 marking process를 분석함으로써 시

스템의 수리적 성능분석을 자동으로 수행한다.

세째, 페트리 네트 모형으로부터 모의실험을 수행하여 그 결과로 얻어진 시스템의 성능 분석치를 제공한다.

넷째, 시스템의 논리적 특성을 사용자가 쉽게 이해 할수 있도록 애니메이션(Animation) 기능을 추가한다.

다섯째, Marking process의 진행과정을 나타내주는 Reachability Tree 및 Coverability Tree를 자동으로 만들어준다.

본 연구에서 개발된 소프트웨어는 사용자가 이산형사건 동적시스템을 페트리 네트로 모델링하면, 수리적분석 혹은 모의실험을 통해 시스템의 중요한 성능분석치를 자동적으로 제공하여 준다.

2. 페트리 네트을 이용한 모델 작성

2.1 개요

페트리네트는 가용한 자원을 토큰(Token)으로 표시하며, 토큰의 이동조건을 논리적으로 표현하고 그 이동과정을 추적함으로써 시스템의 동적특성을 정확히 모델링할수 있게 해준다. 특히 Concurrent, Synchronization, Fork-Join, Conflict 등 특수한 형태의 동작(activity)을 정확히 표현할수 있는 표현력을 구비하고 있어 이러한 특성을 지닌 동적시스템을 모델링하는 데 이상적인 도구이다. 또한 수리적분석과 시뮬레이션에 의해 시스템의 성능평가를 수행하여 이를 시스템의 설계단계에서 이용할 수 있으며, 시스템이 구축된 후에는 시스템의 상태를 감시(monitor)하고 그 활동을 제어하는 제어기(controller)로서의 역할로 전환 가능해 시스템의 전생애(life cycle)를 통해 활용되어질 수 있다. 이렇게 강력한 기능을 가지고 있으면서도 추계적과정(Stochastic Process)에 대한 전문지식이 없는 비전문가라도 쉽게 배울 수가 있고, 이를 통해서 분석하고자 하는 시스템을 모델링할수 있다는 장점이 있다.

2.2 모델 작성

페트리 네트의 정의는 다음과 같다.

$$PT = (P, T, \in, OUT, M_0)$$

P 는 원으로 표현되는 Place들의 집합, $P=\{p_1, \dots, p^m\}$ 으로 현 시스템에서 가능한 상태를 나타내는 집합이고, T 는 사각형으로 표현되는 Transition들의 집합, $T=\{t_1, \dots, t_n\}$ 으로 상태가 변화되는 시간의 경과를 나타낸다. 점으로 표현되는 Token은 Place와 Place 사이를 이동하면서 상태를 변화시킨다. Token의 이동은 Place와 Transition을 서로 연결하는 Arc에 따라서 이루어지며 IN, OUT의 함수로 표현된다.

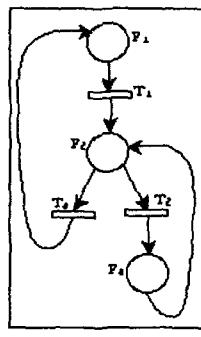


그림 1. 페트리 네트의 예

3. 통합소프트웨어의 구현

3.1 Event Type

분석하고자 하는 시스템이 페트리 네트로 모델링 되면 수리적 분석이 가능한 경우에는 수리적 분석을 할 수 있고, 수리적 분석이 불가능한 경우에는 모의 실험을 통해 시스템의 성능 분석을 수행합니다. 일단 모의 실험이 시작되면 모든 변수들의 값과 통계치가 초기화되며 input database의 data는 프로그램 내부에서 판독된다. 다음 단계에서는 모의 실험 종결사건과 재 초기화 사건을 Event Calendar에 저장한다. 본 연구에서는 모의 실험의 사건을 네 가지 형태(Type)로 구분한다.

Type1: 토큰의 이동 조건이 충족될 경우 발생하는 transition 발사(fire) 사건

Type2: 시스템의 초기 상태의 영향을 감소시키기 위한 모의 실험의 재 초기화 사건

Type3: 교착(deadlock)이 발생할 경우 교착 상태를 해제하기 위한 사건

Type4: 모의 실험 종결 사건

3.2 Event Driven Simulation

Event Calendar란 앞으로 발생할 사건들을 시간의 증가 순(increasing order)으로 정렬한 배열을 의미하며, 소프트웨어는 Event Calendar의 사건의

유형에 따라 각기 다른 방식으로 모의 실험을 진행 한다[3]. 다음 event time이 되어 transition이 발사되면 이에 따라 토큰의 이동이 이루어지고 새로이 enable된 transition을 Event Calendar에 포함시킨다. 이와 더불어 관련된 통계치를 수집하고 다음 event time으로 모의 실험 시간을 진행시킨다.

Event Calendar의 다음 사건이 모의 실험 종결 사건이면 수집된 통계치를 output database에 저장하며 이를 적절한 형태로 가공한 후 리포트 형태로 출력한다.

3.3 Reachability Set

본 소프트웨어에서는 분석 초기 input-database에서 자료를 판독한 후 초기 marking 상태에서 도달 가능한 모든 marking을 찾아, reachability set을 구한다. 이 과정에서 수리적 분석이 힘들 것으로 판단되면 모의 실험 모듈에 의해 시스템의 분석을 수행한다. Reachability set으로부터 페트리 네트가 live한 경우로 판명되면 reduced embedded Markov-chain을 구한 후 이의 상태방정식을 풀음으로써 안정상태 확률을 구한다. 안정상태 확률이 구하여지면 안정상태 확률로부터 시스템의 다양한 성능 평가치를 구할 수 있다.

3.4 수리적 분석

Transition 시간이 0(immediate transition)이거나 지수분포를 따를 경우(exponential transition)의 페트리 네트를 일반형 추계적 페트리 네트(GSPN, Generalized Stochastic Petri Net)라 부른다[4]. GSPN의 경우에는 상태의 수가 지나치게 크지 않을 경우 페트리 네트에 내재되어 있는 연속시간형 마코프-체인을 분석함으로써 시스템의 중요한 성능 평가치를 얻을 수 있게 된다[6]. 이러한 기능을 가진 소프트웨어는 Ciardo 등에 의해 외국에서는 이미 개발된 바 있으나 국내에서는 아직 개발이 이루어지지 않은 것으로 보인다. 본 연구에서는 이러한 수리적 분석 기능을 소프트웨어에 추가함으로서, 모의 실험을 수행하지 않고도 시스템의 성능 분석치를 정확하고 신속하게 구할 수 있도록 하였다.

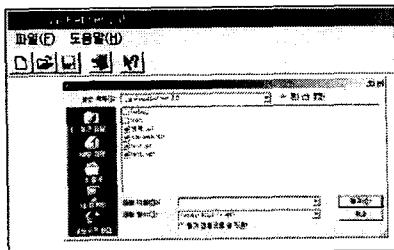


그림 2 프로그램의 구동

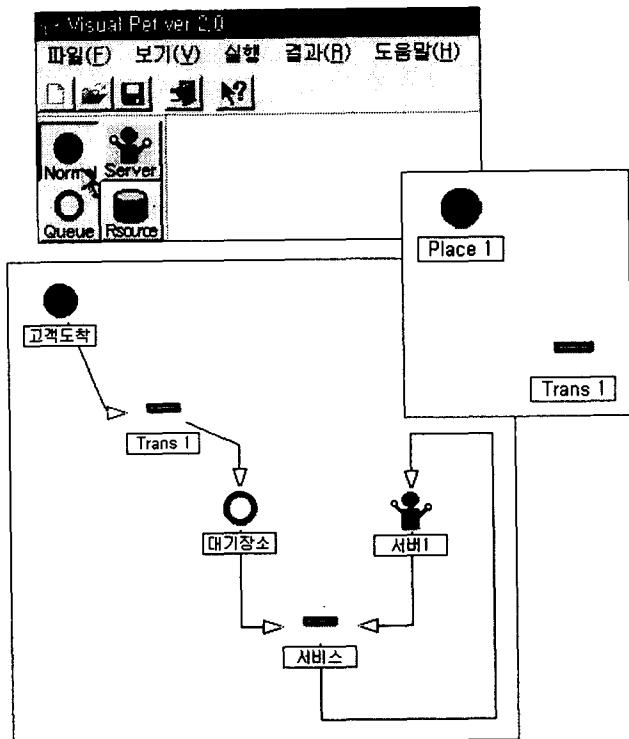


그림 3 개체의 추가 및 연결

3.5 그밖의 특징

본 연구에서는 모의실험 속도를 향상시키고 다양한 형태의 시스템을 모의실험 할 수 있도록 다음과 같은 특성과 기능을 소프트웨어에 구현하였다.

1) Double linked-Allocation Approach

물리적으로 인접한 위치에 각각의 목록(Record)을 저장하는 순차적 할당(sequential-allocation approach)기법과 달리 이중링크 할당(double linked-allocation approach) 기법에서는 각각의 목록(Record)이 상호 논리적 관계(logical relationship)만을 갖게 되므로 이를 이용하면 Event calendar의 관리가 용이해 진다. 따라서

모의실험에 소요되는 시간과 computer memory의 사용량을 감소 시킬수 있으며, 대규모의 사건목록(event record)을 필요로 하는 모의실험의 경우 특히 효과적이 될 것으로 기대한다.

2) 교착(deadlock) 탐지 및 해소방식 도입

서비스를 끝낸 두 개 이상의 unit에 상호 봉쇄(blocking)현상이 발생하여 시스템이 교착상태에 빠질 경우, 이를 즉시 탐지하고 교착을 해소해 주는 기능을 본 연구에서는 모의실험 모듈에 포함시켰다. 본 연구에서는 교착의 원인이 되는 transition의 동시 발사에 의해 토큰을 목적 place에 동시에 이동시킴으로써 교착을 해소하는 방식을 사용하였으며 교착해소에는 일정한 시간이 소요된다고 가정하였다[1].

3) 이외에도 본 연구에서는 dynamic random switch 및 dynamic(marking-dependent) firing rates를 허용하고 지수, 정규, 일양, 감마, 얼랑, 족시안 분포 등 다양한 확률분포를 지원하도록 함으로써 시스템의 모델링 능력을 높이고자 노력하였다.

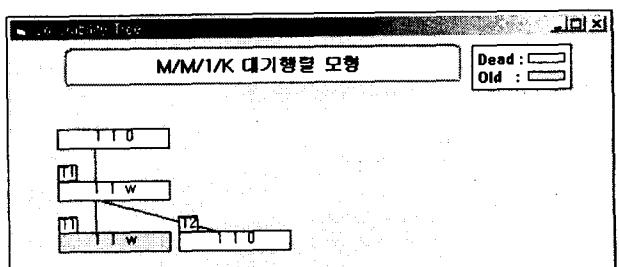


그림 4. Coverability Tree

4. VisualPet의 기능과 특징

4.1 Network Designer

사용자는 대상시스템을 세 가지 구성요소(Place, Transition, Arc)를 이용하여 페트리 네트로 표현한다. 이를 위한 네트워크 디자이너가 개발되었는데, 사용자가 네트워크 모형을 작성하면서 각 구성 요소에 필요한 데이터를 바로 입력할수 있도록 속성창(Property Window)와 아크연결창(Arc Connection Window)을 지원한다.

네트워크 모형과 데이터가 입력되면 Software의

내부에서 이를 Input database에 받아들이게 되는데 이때 만약 입력데이터(Input Data)에 오류가 있으면, Software가 자동으로 오류를 검사하여 적당한 메시지를 출력한다. 따라서 모형의 오류 또는 입력 데이터(Input Data)의 오류는 즉시 발견되고 사용자가 바로 수정 할 수 있다.

이러한 과정을 거쳐 네트워크 모형과 입력 Data base가 완성되면, 사용자는 시스템의 특성에 따라 수리적분석 모듈 혹은 모의실험 모듈을 선택하게 된다.

특정한 분석모듈이 선택되면 소프트웨어는 더 이상의 입력과정 없이 수리적 분석이나 모의실험을 자동으로 수행해 원하는 시스템의 중요한 성능평가 및 Reachability Tree를 제공해 준다. 모의실험 모듈이 선택될 경우에는 모의실험의 Animation 과정이 화면에 출력된다.

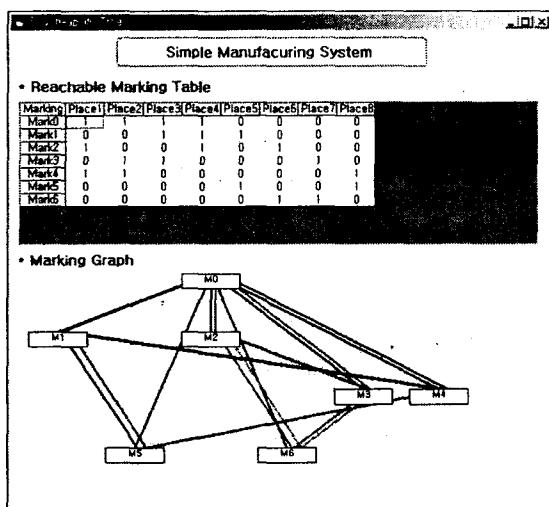


그림 5. Reachability Graph

4.2 수리적분석과 시뮬레이션

상태변환 과정의 시간 분포가 지수분포일 경우에는 페트리 네트의 marking process는 연속시간형 마코프-체인(Continuous Time Markov Chain)과 동일함을 보일 수 있으므로 시스템의 상태의 수가 지나치게 크지 않을 경우 수리적분석이 가능하다. 그러나 시간 분포가 일반 분포를 따르거나 시스템의 상태의 수가 지나치게 클 경우에는 수리적 분

석이 불가능하며 모의실험에 의해 시스템의 성능을 분석하여야만 한다. 모듈 1은 이러한 경우에 선택되는 모의실험 모듈로서 본 연구에서는 시뮬레이션의 성능을 높이기 위하여 기존모듈을 새롭게 개발하였다.

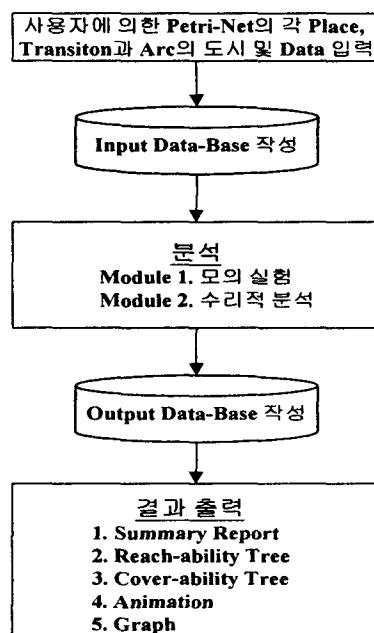


그림 6. 프로그램의 구성

5. 예제

예제로서 2 개의 작업장으로 구성된 간판시스템을 본 소프트웨어를 이용하여 분석하고자 한다. 각

General																									
Project Name	M/M/1/K 대기행렬 모형																								
Author	황승철																								
Date	2000/6/13																								
Input Data																									
Place	Transition																								
<table border="1"> <tr> <th>Name</th> <th>Type</th> <th>InitMark</th> </tr> <tr> <td>장재고객</td> <td>Normal</td> <td>1</td> </tr> <tr> <td>서버</td> <td>Server</td> <td>1</td> </tr> <tr> <td>대기공간</td> <td>Queue</td> <td>0</td> </tr> <tr> <td>마찰고객</td> <td>Normal</td> <td>0</td> </tr> </table>	Name	Type	InitMark	장재고객	Normal	1	서버	Server	1	대기공간	Queue	0	마찰고객	Normal	0	<table border="1"> <tr> <th>Name</th> <th>Type</th> <th>DistType</th> </tr> <tr> <td>고객도착</td> <td>Timed</td> <td>I.Expon</td> </tr> <tr> <td>서비스</td> <td>Timed</td> <td>I.Expon</td> </tr> </table>	Name	Type	DistType	고객도착	Timed	I.Expon	서비스	Timed	I.Expon
Name	Type	InitMark																							
장재고객	Normal	1																							
서버	Server	1																							
대기공간	Queue	0																							
마찰고객	Normal	0																							
Name	Type	DistType																							
고객도착	Timed	I.Expon																							
서비스	Timed	I.Expon																							
Output Data																									
<ul style="list-style-type: none"> • Reinit Event Type : None • End Event Type : FixedTime • Ending Time : 100.00 • Summary Continuous Data of List 																									
<input type="button" value="출력시간 설정"/> <input type="button" value="클립보드로 복사"/> <input type="button" value="인쇄"/> <input type="button" value="취소"/>																									

그림 8. Simulation 결과보고서

작업장마다 간판(kanban)을 위한 게시판(Bulletin Board)과 재고를 위한 버퍼를 가지고 있다. 작업장에 주문이 도착하면 생산지시가 내려진다. 그림7은 이 시스템을 VisualPet으로 모델링 한 것이다.

참고문헌

- [1] 박찬우, 황상철, 이효성, “봉쇄와 교착이 존재하는 시스템의 성능분석을 위한 페트리-네트 기반 모의실험 소프트웨어의 개발”, 시뮬레이션학회 논문지, 제9권, 1호, 2000,3

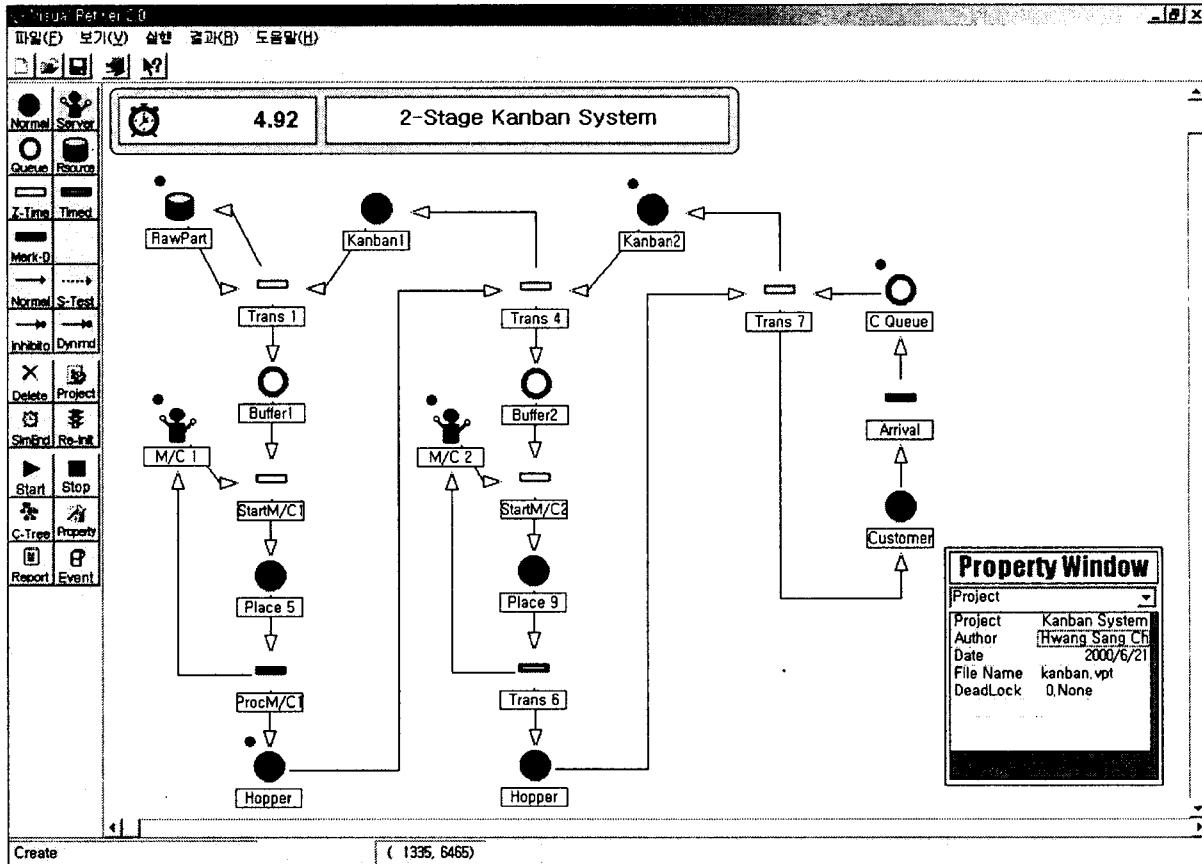


그림 7. VisualPet의 실행화면

6. 결론 및 향후과제

페트리 네트는 통신시스템, 제조시스템 등 다양한 시스템의 모델링 및 제어에 활용될 수 있으므로 본 연구에서 개발된 소프트웨어의 이용범위는 매우 넓으리라 판단된다.

본 연구를 통해 개발된 소프트웨어는 페트리네트의 장점을 그대로 가지고 있기는 하지만 시스템을 표현하는데 있어서 복잡한 모델의 경우에 노드의 수가 너무 많아지는 경향이 있다.(이것은 페트리네트의 단점중 하나이다.) 앞으로 이를 보완하여 실제 시스템을 더욱 충실하게 모델링하고 분석할 수 있는 도구로 발전시켜 나가야 하겠다.

- [2] Desrochers Alan A. and AI-Jaar Robert Y., "Applications of Petri Nets in Manufacturing Systems", IEEE Press
- [3] Law, Avrill M., and W. D. Kelton, "Simulation Modeling and Analysis", 2nd ed. McGraw-Hill, New-York, 1991
- [4] Mayr Ernst W., "An Algorithm for the General Petrinet Reachability Problem"
- [5] Murata Tadao, FELLOW, "Petri Nets: Properties, Analysis and Applications", 1989, IEEE
- [6] N. Viswanadham, and Y. Narahri, "Performance Modeling of Automated Manufacturing Systems", Prentice Hall, Englewood Cliffs, New

Jersey, 1992

[7] Microsoft VisualBasic 5.0 Manual, Microsoft, 1997