

씩 차례로 가져다가 수행한다. DBMS는 갱신 작업을 하기 위하여 반드시 보조기억장치를 접근해야 하기 때문에, 갱신 작업은 비교적 긴 시간을 필요로 한다.

[1]에 소개된 이동물체 데이터베이스의 페트리 넷 모형은 Design/CPN[2]으로 구축되었다. 그림 1은 CPN-Tools[3]로 구축된 이동물체 데이터베이스의 페트리 넷 모형이다. 본 논문은 그림 1에 보이는 페트리 넷의 구조적 성질을 분석함으로써 이동물체 데이터베이스의 구조적 성질을 알아본다.

2. 관련 연구

관심 있는 시스템의 페트리 넷 모형을 구축하고 페트리 넷을 분석하여 실제 시스템의 성질을 분석하는 방법에 대한 연구 결과가 매우 많이 발표되었다. 특히, 실제 시스템을 구축하기 전에 페트리 넷을 이용하여 설계 중인 시스템의 성질을 분석함으로써 구현 이전 단계에 설계상의 오류를 발견할 수 있다.

[4]는 컬러 페트리 넷(colored Petri net)이라는 페트리 넷의 변형을 소개하고, 컬러 페트리 넷을 이용하여 소프트웨어 시스템의 모형을 구축하고 분석하는 방법을 소개한다. 또한, 분산 실시간 시스템의 명세를 페트리 넷으로 표현하고 검증하는 방법이 [5, 6]에 소개된다. Business 처리 시스템의 모델 방법으로 event-driven process chain (EPCs)이 널리 사용된다. [7]에는 EPCs를 컬러 페트리 넷으로 변형하는 방법이 소개된다. [8, 9, 10]은 Time Petri net을 이용하여 실시간 시스템의 명세를 표현하고 검증하는 방법을 소개한다.

3. 페트리 넷 모형 분석

그림 1의 t1은 p2와 p6에 n 개의 토큰을 놓는, 초기화를 위한 트랜지션이다. 그러므로 p2와 p6에 각각 n 개의 토큰을 놓음으로써 트랜지션 t1과 플레이스 p1, p3를 제거할 수 있다. p4는 T2의 입력 플레이스이자 출력 플레이스이며 입력 간선의 inscription과 출력 간선의 inscription이 동일하고, p4에는 T2를

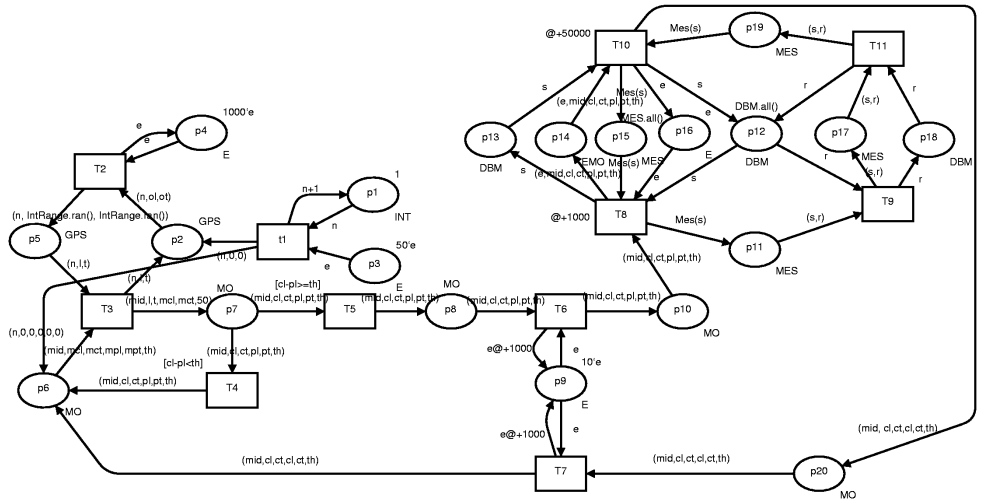


그림 2 CPN-Tools로 구축한 이동물체 데이터베이스의 페트리 넷 모델

enable 시키기에 충분한 만큼의 토큰이 존재한다. 이와 같은 플레이스 p4를 ‘토큰을 가지고 있는 self-loop’이라 하고, 이러한 self-loop를 제거하여도 liveness와 boundedness, 등의 성질에 변함이 없다. p9도 역시 같은 이유로 제거가 가능하다.

그림 1의 p13과 p14는 T8이 격발할 때 동시에 새로운 토큰을 받고 이 토큰은 T10이 격발할 때 동시에 소멸된다. 또한, p16에 토큰이 하나뿐임으로 T8이 한번 격발하면 T10이 격발하여 하나의 토큰을 복원하지 않으면 T8을 연속으로 격발하기가 불가능하다. 그러므로 p13과 p14를 하나로 합병하여도 그림의 liveness와 boundedness의 성질이 불변이다. 초기화 트랜지션과 self-loop를 제거하고 p13과 p14를 합병하여 얻은 페트리 모형이 그림 2에 보이며, 그림 1의 페트리 넷이 live하고 bounded할 필요-충분조건은 그림 2의 페트리 넷이 live하고 bounded한 것이다.

그림 2의 T2의 입력 간선이 하나뿐이고 출력 간선 역시 하나뿐이다. 따라서 p2와 p5를 합병하고 T2를 삭제하여도 liveness와 boundedness 성질에는 변함이 없다. 마찬가지로 방법으로 p7, p8, p10을 합병하고 T5와 T6를 삭제하여도 liveness와 boundedness 성질에 변함이 없다. 따라서 그림 2가 live하고 bounded할 필요충분조건은 그림 3이 live하고 bounded한 것이다.

그림 3의 p2는 self-loop임으로 삭제하고, T11과 T9를 합병하고, p20과 p6을 합병하여도 liveness와 boundedness 성질에 변함이 없다. 이렇게 얻은 페트리 넷이 그림 4에 보인다.

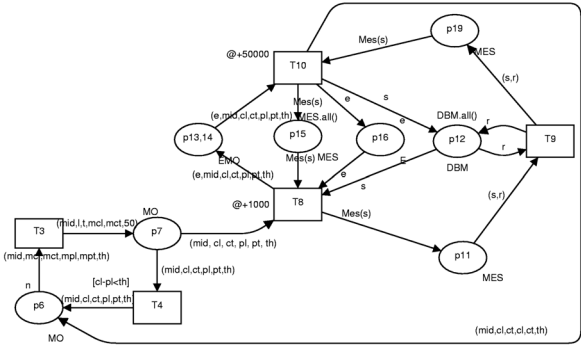


그림 3 그림 3에서 self-loop 삭제, T11과 T9 합병, p20과 p6 합병 결과.

그림 4에서 database 부분만 분리한 것이 그림 5에 보인다. 그림 5의 페트리 넷을 행렬로 표현하면 표 1과 같다. 표 1에서 $\Sigma = \Sigma s \neq r <s, r>$, 이며 그림 5의 Mes(s)는 Σ 를 생성하는 CPN-tool의 함수다. 그림 5의 DBMS.all()은 분산데이터베이스를 구성하는 모든 지역 데이터베이스를 나타내는 토큰을 생성하는 CPN-tool 함수이며, MES.all()은 가능한 모든 메시지를 생성하는 함수이다.

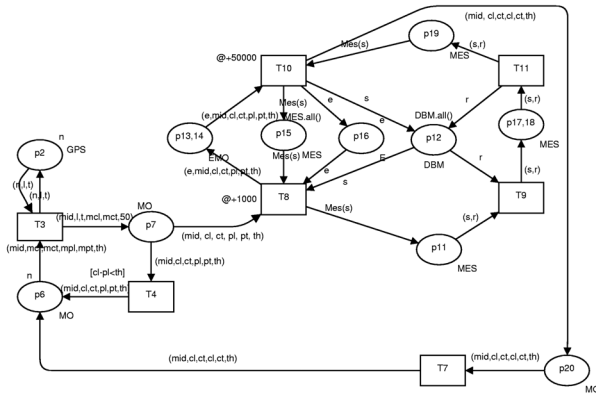


그림 4 그림 2에서 p2와 p5를 합병하고, p7, p8, p10을 합병하여 얻은 페트리 넷.

그림 5는 다음과 같은 S-invariant를 가지고 있음을 알 수 있다.

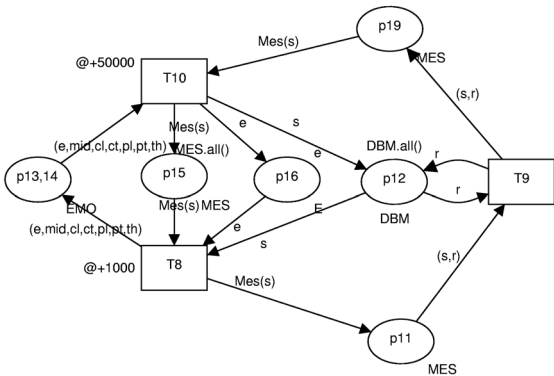


그림 5 그림 4의 Database 부분

$$\begin{aligned}
 y_1 &= (1 \ 0 \ 1 \ 0 \ 0 \ 1)^T \\
 y_2 &= (0 \ \Sigma \ <e, \text{mid}, \dots> \ 0 \ 0 \ 0)^T \\
 y_3 &= (0 \ e \ 0 \ <e, \text{mid}, \dots> \ 0 \ 0)^T \\
 y_4 &= (0 \ s \ 0 \ 0 \ <e, \text{mid}, \dots> \ 0)^T \\
 y_5 &= (e \ 0 \ 0 \ \Sigma \ 0 \ e)^T \\
 y_6 &= (s \ 0 \ 0 \ 0 \ \Sigma \ s)^T
 \end{aligned}$$

y_1 의 의미는 p19, p15, p11의 토큰의 수가 불변임을 의미하고, y_2 의 의미는 p13의 토큰에 Σ 를 곱하고 p15의 토큰에 $<e, \text{mid}, \dots>$ 를 곱하면 p13, p15의 토큰의 수가 불변임을 의미한다. 마찬가지로 나머지의 S-invariant들도 각각 해당 플레이스의 토큰의 가중치 합이 불변임을 의미한다.

y_1, y_3, y_6 의 합은 $y = (s+1 \ e \ 1 \ <e, \text{mid}, \dots> \ \Sigma \ s+1)^T$ 이며, y 의 모든 원소가 양수임으로 모든 플레이스의 토큰의 가중치의 합이 불변임을 알 수 있다. 즉, 그림 5의 페트리 넷은 bounded 넷임을 알 수 있다. 즉, 데이터베이스를 구성하는 메시지의 총 수와 지역 데이터베이스의 총 수가 불변이라는 결론을 얻을 수 있다.

표 1의 행렬로부터 T-invariant $(1 \ 1 \ n-1)$ 을 구할 수 있다. 즉, 트랜지션 t8을 격발하고, t9을 n-1회 격발한 뒤, t10을 1회 격발하면 원래 마킹으로 되돌아온다. 또한 p16에 오직 한 개의 초기 토큰이 존재하고 t8을 격발하면 이 토큰이 삭제됨으로 t8 격발 후에는 t9 격발만 가능하다. t10 격발은 p19에 n-1 개의 토큰이 있어야 가능하므로 결국 t9을 n-1번 격발한 다음 t10을 한번 격발하여 초기 마킹으로 돌아오는 격발 순서가 유일한 격발 순서로 본 페트리 넷은 live함을 알 수 있다.

	p19	p13	p15	p16	p12	p11
t10	$-\Sigma$	$-\langle e, \text{mid}, \dots \rangle$	Σ	e	s	0
t8	0	$\langle e, \text{mid}, \dots \rangle$	$-\Sigma$	-e	-s	Σ
t9	(s,r)	0	0	0	0	-(s,r)

표 1 그림 5의 행렬 표현

그림 4에서 그림 5와 나머지 부분과의 관계를 살펴보면 그림 5의 입구는 p10뿐이고 출구는 p20뿐임을 알 수 있다. 따라서, 그림 5를 하나의 트랜지션으로 표현하면 그림 6을 얻으며, 그림 6이 live하고 bounded할 필요충분조건은 그림 4가 live하고 bounded한 것임을 알 수 있다.

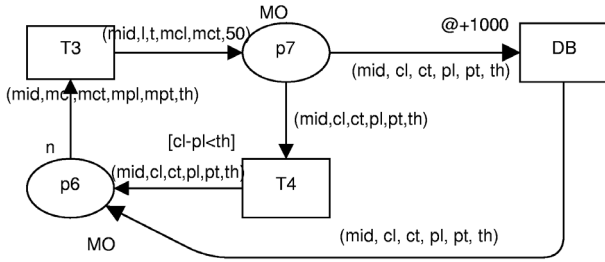


그림 6 데이터베이스 부분을 하나의 트랜지션으로 나타냄.

그림 6은 live하고 bounded함을 쉽게 알 수 있다. p6와 p7의 토큰의 합은 항상 n이고, T3 격발 후 p7에 도착한 토큰은 T4 격발이나 DB 격발 후 다시 p6로 돌아온다. 따라서 이동물체 데이터베이스 시스템은 live하고 bounded함을 알 수 있다.

4. 결론

시스템 설계 단계에서 구축하고자 하는 시스템의 모형을 분석하여 설계 중인 시스템의 요구 사항을 검증하는 것이 매우 중요하다. 이동물체 데이터베이스 시스템의 요구 사항 중에는, 시스템이 deadlock에 빠지지 않고 항상 서비스를 제공해야 한다는 것과, 시스템에 관련된 자원이 일정 수준을 유지해야 한다는 것이 있다. 본 논문은 이동물체 데이터베이스의 페트리 넷 모형이 live하고 bounded함을 보였다. 즉, 이동물체 데이터베이스 시스템이 deadlock에 빠지지 않고 항상 서비스를 제공하며, 사용하는 자원의 양이 일정 수준을 유지한다는 것을 검증한 것이다.

참고문헌

- [1] 임재걸, 이계영, "이동물체 데이터베이스의 페트리 넷 모형," 대한전자공학회 논문지 2004년 5월호 게재 예정.
- [2] K. Jensen, Design/CPN [Online]. Dept. Computer Science, Univ. Aarhus, Denmark. Available: <http://www.daimi.au.dk/designCPN/>
- [3] K. Jensen, Design/CPN [Online]. Dept. Computer Science, Univ. Aarhus, Denmark. Available: <http://wiki.daimi.au.dk:8000/cpntools/cpntools.wiki>
- [4] Jensen, Kurt, "Coloured Petri Nets: A High Level Language for System Design and Analysis," *Lecture Notes in Computer Science, Vol. 483; Advances in Petri Nets 1990*, Berlin, Germany: Springer-Verlag, pages 342-416, 199

- [5] M. Bax and N. Giambiasi, "Modeling and simulating temporal constraints in OpenRT," *Proc. 1996 Simulation Multiconference - Simulators International XIII, 8-11 April 1996, New Orleans, LA*, pages 128-133. 1996.
- [6] A. Alekseev, A. Bystrov, S. Kurtov, S. Mylnikov, V. Nepomnyashchy, E. Okunishnikova, P. Chubarev, and T. Churina, "Application of Petri nets in verification of distributed systems represented in the Estelle language," *Journal of Computer and Systems Sciences, Vol. 38, No. 5*, pages 771-781. 199
- [7] P. Langner, C. Schneider and J. Wehler, "Petri Net Based Certification of Event-driven Process Chains," *Lecture Notes in Computer Science, Vol. 1420*, Berlin: Springer-Verlag, 19th Int. Conf. on Application and Theory of Petri Nets, ICATPN'98, Lisbon, Portugal, June 1998, pages 286-305, 199
- [8] D. Gurovic, W. Fengler and J. Ntzel, "Development of real-time system specifications through the refinement of duration interval Petri nets," *Proc. IEEE Int. Conf. on Systems, Man, and Cybernetics (SMC'2000), 8-11 October 2000, Nashville, TN, Vol. 4*, pages 3093-3098. 2000.
- [9] O. Roux, D. Delfieu, and P. Molinaro, "Discrete Time Approach of Time Petri Nets for Real-Time Systems Analysis," *ETFA2001 proceedings of the 2001 8th IEEE International Conference on Emerging Technologies and Factory Automation, Volume 2, IEEE Catalog number: 01TH8597*, pages 197-204. October 2001.
- [10] J. Wang, Y. Deng, and G. Xu, "Reachability analysis of real-time systems using time Petri nets," *IEEE Trans. on Systems, Man, and Cybernetics; B: Cybernetics, Vol. 30, No. 5*, pages 725-736. 200