

## 시간제약이 있는 정보처리시스템의 성능분석 모형

허 선<sup>1</sup> · 주국선<sup>1</sup> · 정석윤<sup>2\*</sup> · 윤주덕<sup>3</sup>

<sup>1</sup>한양대학교 산업경영공학과 / <sup>2</sup>한화에스앤씨(주) / <sup>3</sup>(주)미라콤아이앤씨

### A Model for Performance Analysis of the Information Processing System with Time Constraint

Sun Hur<sup>1</sup> · Kook-Sun Joo<sup>1</sup> · Seok-Yun Jeong<sup>2</sup> · Joo-Deok Yun<sup>3</sup>

<sup>1</sup>Department of Industrial management Engineering, Hanyang University, Sa-dong, Ansan, Gyeonggi-do, 426-791, Korea

<sup>2</sup>Hanwha S&C, Janggyo-dong, Jung-gu, Seoul, 100-797, Korea

<sup>3</sup>Miracom Inc., Samsung-dong, Gangnam-gu, Seoul, 135-090, Korea

In this paper, we consider the information processing system, which organizes the collected data to meaningful information when the number of data collected from multiple sources reaches to a predetermined number, and performs any action by processing the collected data, or transmits to other devices or systems.

We derive an analytical model to calculate the time until it takes to process information after starting to collect data. Therefore, in order to complete the processing data within certain time constraints, we develop some design criteria to control various parameters of the information processing system. Also, we analyze the discrete time model for packet switching networks considering data with no particular arrival nor drop pattern.

We analyze the relationship between the number of required packets and average information processing time through numerical examples. By this, we show that the proposed model is able to design the system to be suitable for user's requirements being complementary the quality of information and the information processing time in the system with time constraints.

**Keywords:** Information Processing System, Information Processing Time, Markov Chain, Time Constraint, Packet Switching Networks

#### 1. 서론

유비쿼터스 환경에서 각 기기들이나 시스템들은 주변의 다양한 소스로부터 데이터를 수집하고 이를 처리하여 동작을 수행한다. 이 동작의 근원이 되는 데이터의 발생은 사전 예측이 불가능하다. 유비쿼터스 환경에서의 데이터는 네트워크 내 각 시스템에서 발생 가능한 데이터들을 모두 포함하므로 기존의 네트워크에서 주고받는 데이터보다 훨씬 복잡하고 다양한 형태를 가진다. 다양한 구성원들이 서로 유기적으로 연결되어 동작하는 네트워크에서는 대부분의 경우 단순하고 단일한 데이터를 사용하는 것이 아니라 다수의 다양한 데이터가 모였을

때 비로소 효용가치를 가지게 되는 경우가 많다. 본 논문에서는 특히 의미가 있고 시스템의 동작에 직접적으로 영향을 주는 데이터의 집합을 정보(information)라고 하기로 한다. 하지만 효용성을 가지는 정보라 하더라도 주어진 시간 내에 제공되어야만 사용자의 목적에 맞게 사용될 수 있다.

하나의 동작을 위해 다수의 데이터를 수집하는 경우 기기나 시스템(이를 편의상 정보처리시스템이라고 하자)들은 동작에 필요한 정보를 구성하는 데에 충분한 양의 데이터가 도착하기를 기다리는 대기현상이 발생한다. 이는 시스템의 성능과 직접적으로 연관이 있으며 실제 동작의 수행가능여부를 결정하는 중요한 판단요소이다. 예를 들면, 날씨정보시스템은 다수

본 연구는 지식경제부 우수제조기술연구센터 지원사업의 연구비로 진행되었음(10032910-2009-11).

\* 연락저자 : 정석윤 박사, 100-797 서울시 중구 장교동 한화빌딩 19층, Fax : 729-3416, E-mail : pisces18@hanmail.net

2010년 4월 9일 접수; 2010년 5월 20일 수정본 접수; 2010년 5월 25일 게재 확정.

의 센서를 통해 온도, 압력, 풍향 등 다양한 날씨관련 데이터를 수집한다. 이 때 수집한 데이터는 그 하나하나로서는 의미가 없지만 일정한 조건을 만족하면 새로운 정보로 업데이트된 형태로 처리되어 사용자들에게 제공한다. 하지만 정보를 업데이트할 수 있는 조건(예를 들어 특정지역에서 몇 건 이상의 기압 변화량에 대한 데이터가 있어야 한다는 등)에 필요한 데이터가 도착하여 정보를 처리하는 시간이 너무 길어진다면 적절한 시간 내에 업데이트된 날씨정보를 제공하지 못한다.

450mm 크기의 웨이퍼를 사용하여 칩을 생산하는 첨단 반도체 생산라인에서는 이전 생산방식에 비해 인간의 개입이 전혀 이루어지지 않는 완전자동화가 실현되어야 한다. 따라서 생산 스케줄은 완벽해야 하며 각 공정을 각 장비에 할당하는 *dispatching*도 실시간으로 이루어져야 한다. 그러나 불량률의 발생, 일시적인 자재부족, 긴급주문처리 등과 같은 생산조건상의 변화가 생겨서 이를 국부적인 *dispatching* 조절로 처리할 것인지, 아니면 공정 스케줄링을 다시 해야 할 지 결정을 해야 할 경우가 있다. 스케줄링을 다시 하는 경우에는 많은 시간이 소요되기 때문에 새로운 생산 스케줄이 완성되었을 때에는 이미 그 결과를 반영할 수 없게 되어버리는 수도 있다. 따라서 현장에서 발생하는 생산조건 변화 데이터들을 수집하여 스케줄링을 다시 하게 될 경우 이를 실제로 반영할 수 있는 시간 내에 스케줄링이 완성되도록 스케줄링 개시시점을 정하는 것이 매우 중요하다.

또 다른 예를 들면, 교통정보 네트워크는 교통정보를 수집하는 센서와 수집된 정보를 처리하는 교통정보시스템, 그리고 차량에 설치된 교통정보제공기기로서 구성된다. 교통정보시스템은 시시각각 변화하는 교통상황을 수집하는 센서로부터 데이터를 실시간으로 수신한다. 이들 데이터들은 그 자체로는 유용한 의미를 가지지 못하지만 미리 정해진 일정한 요건(예: 비정상적인 저속차량의 수가 급격히 증가)이 충족되면 의미있는 정보로써 처리되어 이를 주변을 지나는 각 차량에 제공한다. 차량에 설치된 정보수신기기는 차량이 특정 교통구간을 지나기 전에 필요한 정보를 교통정보시스템으로부터 수신하여 운전자에게 제공함으로써 교통의 혼잡함이나 위험을 알릴 수 있다. 즉, 각 센서는 실시간으로 교차로의 교통상황데이터를 수집하며 교통정보시스템은 수집된 정보를 처리하여 자동차가 해당 교차로에 진입하기 전에 자동차의 정보제공기기에 교차로의 교통상황을 전송하여야 한다.

네트워크에서는 여러 데이터가 모여 상황과 환경에 관련된 정보를 생성한다. 그러나 사용자의 환경이나 상황은 시간의 경과 또는 다른 요인으로 인하여 변화할 수 있다. 이런 경우 수집된 데이터는 효용성이 떨어지며 불필요하게 될 수도 있다. 앞선 날씨정보시스템을 예로 들면 어떤 센서가 시스템에 데이터를 전송한 후에 기상상태가 갑자기 변할 경우 새로운 데이터를 다시 전송하게 되면 이전에 시스템에 도착한 데이터는 불필요하고 새로 도착한 데이터를 사용하여야 한다.

본 논문에서는 정보처리시스템이 여러 소스로부터 데이터

를 수집한 후 수집된 데이터의 양이 일정한 개수를 넘게 되면 이들 데이터들을 사용하여 의미를 가지는 정보를 구성하고 처리하여 동작을 수행하거나 다른 기기나 시스템들에게 전송하는 정보처리시스템을 고려한다. 데이터를 수집하기 시작한 후부터 정보를 처리하기까지 소요되는 시간을 수리모형을 통해 구한다. 이로써 일정한 시간제약내에 정보처리를 완료하기 위해 정보처리시스템의 각종 모수들(예 : 데이터 처리를 개시하는 조건, 데이터 처리속도 등)을 제어할 수 있는 설계기준을 마련한다.

이와 관련된 기존연구에서의 접근방식은 크게 두 가지로 구분할 수 있는데, 첫 번째는 새로운 개념과 구조를 제안하는 방법론에 관한 연구이고 두 번째는 수리적인 방법을 이용해서 시스템을 모형화하고 분석하는 연구이다. 첫 번째 방법론에 관한 연구는 다른 분야에서 사용되어 오던 프레임워크나 개념을 정보가 가지는 특성에 맞게 재구성하여 새로운 형태로 제안하고 있다. Sun and Yen(2005)은 ISC(Information Supply Chain)의 개념을 제시하여 기존의 공급사슬의 개념을 정보에 적용하였다. 즉, 기존의 SCM(Supply Chain Management)에서 연구된 관리방법을 ISC에 적용하는 방법론을 제시하였으나 시간의 변화에 대한 고려는 없었다. Hedly *et al.*(2006)에서는 hidden web database에서 수집된 데이터를 이용하여 정보를 생성하는 2단계 프레임워크를 제시하였다. Chakravarthy *et al.*(2002)는 WebVigil이라는 시스템을 제안하였는데, 이는 JIT(Just In Time)의 PUSH 개념을 사용하여 웹 기반의 비구조적인 문서에서 변화를 감시하여 통보하는 체계이다. Choi *et al.*(2005)는 최신 정보를 기반으로 하여 프로세스를 실행하고 목적달성을 위한 경영개념을 항공물류에 접목한 실시간 정보 시스템을 분석하였다. 그러나 이들 방법론에 관한 연구들은 시스템의 분석을 위해 새로운 개념을 제공하기는 하지만 해석적 모형에 대한 접근은 제시하지 않았다.

두 번째 수리적 모형에 관한 연구 가운데 Tolstikov *et al.*(2007)는 사용자의 상황이나 환경에 의존적인 정보를 관리하는 방법론과 시간에 따라 변화하는 정보에 관하여 동적 베이지안 네트워크를 이용하여 해석적인 모형을 제시하였다. Zhang and Ji(2006)는 데이터를 처리하는 information fusion 프레임워크를 제시하고 주기적으로 변하는 정보의 흐름에 대한 수리적인 방법은 동적 베이지안 네트워크를 이용하였다. Yen *et al.*(2004)은 정보의 변화와 예측을 표현하기 위해 information needs의 개념을 이용한 새로운 알고리즘을 제시하였으나 시간의 변화에 따른 정보의 특성 분석은 수행하지 않았다.

기존 연구들에서는 새로운 방법론을 제시하거나 해석적 모형을 제시하였으나 정보를 수집하는 과정에서 나타나는 데이터의 특성을 반영하거나 정보가 가지는 시간제약을 고려한 연구는 미흡하다. 최근에 Hur *et al.*(2007)과 Nam *et al.*(2009), Shin *et al.*(2010)에서는 일정 개수의 데이터가 수집되면 이를 처리하는 정보시스템을 모형화하고 확률과정을 이용하여 성능분석을 하기 위한 수리적인 방법을 제시하였다. 본 연구에서는

이 연구들을 확장하여 데이터의 도착과 소멸이 특정한 패턴이 없는 경우를 고려하며 패킷통신망 환경에 맞는 이산시간 수리 모형도 분석하여 시스템의 중요한 성능적도인 시간제약이 있는 정보의 평균처리시간을 구한다.

제 2장에서는 시간제약을 가지는 실시간 정보처리의 의미를 설명하고 수리모형을 제시한다. 제 3장에서는 연속시간에서의 정보처리시스템 모형의 평균 정보처리시간을 구하고 제 4장에서는 이산시간 모형에서의 평균 정보처리시간을 계산한다. 제 5장에서는 정보처리시스템이 처리해야 하는 정보의 종류가 두 가지 이상인 경우 이들을 모두 처리하는 데에 걸리는 평균 정보처리시간을 구하는 방법을 제시한다. 제 6장에서 수치예제를 보이고, 제 7장에서 결론과 함께 추후 연구방향을 제시한다.

## 2. 시간제약이 있는 정보처리 모형의 설명

다양한 시스템과 기기들이 연결되어 구성되는 유비쿼터스 환경에서 각 정보처리시스템들은 주어진 상황이나 목적에 맞게 필요한 데이터를 수집하고 저장하여 이를 처리한다. 처리된 데이터는 필요한 경우 동일 네트워크 상에 있는 다른 시스템에 전송되거나 시스템 내부에서 목적에 맞는 정보로 다시 처리된다. 이처럼 각 시스템은 다른 시스템으로부터 데이터를 수집하기도 하고 다른 시스템으로 데이터를 방출하기도 한다. <Figure 1>은 정보처리시스템이 데이터를 받아서 정보를 처리하는 과정을 나타내고 있다.

정보처리시스템이 동작하려면 그 동작 수행에 필요한 데이터를 입력받아야 한다. 데이터는 불규칙적으로 발생하며 정보는 한 개 이상의 데이터 소스에서 발생하는 데이터를 이용하여 구성된다. 데이터를 구성하는 기본단위를 패킷(packet)이라고 정의하자. 즉, 데이터는 크기가 일정한 패킷들의 집합으로 구성되며 하나의 데이터에는 한 개 이상의 패킷이 존재한다. 정보처리시스템들이 주고받는 데이터는 패킷들의 집합이며 일정한 개수의 패킷이 수집되면 비로소 유의한 정보구성이 완료되고 시스템은 이 정보를 처리하여 동작을 수행한다. 한편 데이터는 정보처리시스템의 동작이 운용환경이나 시간에 종속적이기 때문에 만약 동작을 수행하기 위한 환경이 사라지거나 다른 요인으로 변화한 후 전달받은 데이터 또는 시스템의 오작동으로 잘못된 데이터는 무의미하며 쓸모없게 되므로 시

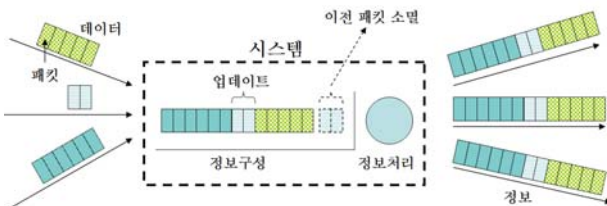


Figure 1. Procedure for processing the data in the information processing systems

스템 내에서 소멸한다.

본 연구에서는 정보단말기들이 패킷을 전송하고 이를 정보처리시스템이 제공받는 일련의 과정을 <Figure 2>와 같은 확률과정으로 모형화한다. 정보처리시스템은 여러 소스로부터 데이터 패킷을 전송받는데 시스템내에 축적되는 패킷의 양이 일정한 개수(즉,  $N$ )를 넘어서면 정보의 구성이 완료되고 이를 처리하기 시작한다. 여기서  $N$ 은 정보처리시스템이 수집한 데이터 패킷의 양이 충분하여 하나의 의미있는 정보로써 처리할 수 있는 수준임을 양적으로 표현하는 수치이며, 이는 이 시스템이 처리하여 생성한 정보의 질(quality)을 나타내는 척도로 볼 수 있다. 정보처리시스템이 정보처리를 완성할 때까지의 시간을 구하고 이를 이 시스템의 성능척도로 활용한다. 즉,  $N$ 이 큰 값이면 이 시스템이 보다 많은 양의 데이터를 수집하여 정보를 만들었으므로 그 정보의 질이 높다고 할 수 있는 반면 정보처리시간은 지연되어 정보의 적시성을 훼손할 수 있고,  $N$ 을 작게 한다면 그 반대의 결과를 얻게 될 것이다. 따라서 정보처리시간의 제약조건(예를 들면 그림에서  $\tau$ )이 있어서 그 시간내에 정보처리가 종료될 수 있도록  $N$ 을 조절함으로써 정보의 질과 적시성 간의 상호보완(trade-off)를 달성할 수 있다.  $N$  이외에 시스템의 CPU 성능과 관련이 있는 정보처리속도 등과 같은 다른 모수들이 평균정보처리시간에 미치는 영향도 파악할 수 있다. 이 과정은 본 연구에서 수치예제를 통해 보이도록 한다.

<Figure 2>의 확률과정은 연속시간 확률과정과 이산시간 확률과정으로 구분한다. 연속시간 모형의 경우에는 패킷의 도착과 소멸이 연속시간 상에서 이루어진다고 가정하지만, 초고속 ATM(Asynchronous Transfer Mode)망과 같이 패킷을 기본 통신단위로 하는 패킷교환망을 사용하는 통신환경에서는 슬롯(slot)이라고 하는 동일한 간격의 작은 시간단위를 기본으로 하고 있다. 따라서 정보의 처리시간, 도착시간 등은 모두 슬롯의 정수 배로 이루어지므로 이산시간 모형을 적용한 성능분석을 실시해야 한다. 본 논문에서는 연속시간 모형과 더불어 이산시간 모형을 분리하여 각각의 정보처리시스템의 성능분석 모형을 제시한다.

## 3. 연속시간모형의 평균 정보처리 시간

연속시간 모형에서는 다음과 같은 가정을 사용한다.

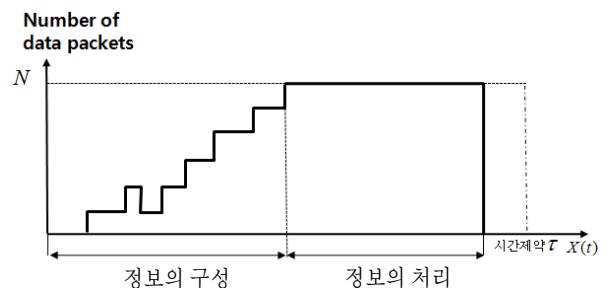


Figure 2. Stochastic process for completing data into the information

- (1) 각 정보단말로부터 정보처리시스템에 도착하는 데이터는 도착률  $\lambda$ 의 포아송과정을 따른다.
- (2) 정보처리시스템에 도착한 패킷의 총 수가  $N$ 이 되면 하나의 정보구성이 완료되며 이를 처리하여 정보처리시스템이 주어진 동작을 개시한다.
- (3) 정보처리시스템이 정보를 처리하는 시간은 평균이  $1/\mu$ 인 지수분포를 따른다.
- (4) 정보구성이 완료되기 전에 시간경과에 따른 퇴화(deterioration) 등의 요인에 의해 패킷이 소멸할 수 있는데, 그 간격은 서로 독립이며 평균이  $1/\alpha$ 인 지수분포를 따른다.
- (5) 정보처리시스템이 이미 정보구성을 완료하여 동작수행에 들어간 후 도착하는 데이터는 정보처리에 영향을 주지 않는다.

이밖에 다음과 같은 기호를 정의하자.

- $\{X(t), t \geq 0\}$  : 상태공간  $E = \{0, 1, \dots, N\}$ 인 확률과정, 단,  $X(t)$ 는 시간  $t$ 에 정보처리시스템에 존재하는 패킷의 개수,
- $g_i$  : 데이터 크기, 즉 한 데이터내의 패킷수가  $i$ 일 확률,  $i = 1, 2, 3, \dots$ ,
- $h_j$  : 소멸되는 패킷의 수가  $j$ 일 확률,  $j = 1, 2, \dots, N-1$ ,
- $T_k$  : 정보처리시스템에 현재  $k$ 개의 패킷이 있을 때, 정보구성에 필요한 나머지  $N-k$ 개 패킷이 추가로 쌓일 때까지 걸리는 시간,  $k = 0, 1, 2, \dots, N-1$ .

확률과정  $\{X(t), t \geq 0\}$ 의 전이율  $q_{jk} (j < k)$ 와  $q_{ml} (m > l)$ 은 다음과 같다.

$$\begin{aligned} q_{jk} &= \lambda g_{k-j}, \quad j < k, \\ q_{jN} &= \lambda \sum_{i=N-j}^{\infty} g_i, \\ j &= 0, 1, \dots, N-2, \quad k = 1, 2, \dots, N-1. \end{aligned} \tag{1}$$

$$\begin{aligned} q_{ml} &= \alpha h_{m-l}, \quad m > l, \\ q_{m0} &= \alpha \sum_{j=m}^{N-1} h_j, \\ m &= 1, 2, \dots, N-1, \quad l = 0, 1, \dots, N-1. \end{aligned} \tag{2}$$

정보가 완성되는 평균시간인  $E(T_0)$ 와 정보 처리시간의 평균인  $1/\mu$ 을 합하면 정보처리시스템에 데이터가 도착하기 시작하여 정보를 처리하기까지의 평균시간을 구할 수 있다.  $N-1$ 개의 일시상태  $0, 1, \dots, N-1$ 과 1개의 흡수상태  $N$ 을 가지는 흡수 CTMC(absorbing continuous time Markov chain)을 고려한다(<Figure 3> 참조).

전체 상태집합을 흡수상태  $A = \{N\}$ 와 일시상태  $A^c = \{0, 1, \dots, N-1\}$ 로 나누고 식 (1)과 식 (2)의 전이율  $q_{jk}$ 와  $q_{ml}$ 을 이용하여 다음과 같은 전이율행렬  $Q$ 를 구성한다.

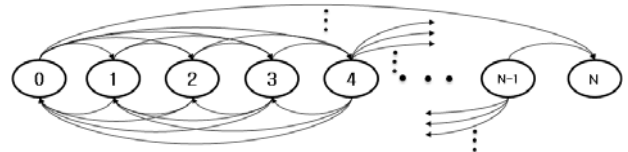


Figure 3. Absorbing continuous time Markov chain

$$Q = \begin{pmatrix} U & U^0 \\ 0 & 0 \end{pmatrix}. \tag{3}$$

여기서  $U$ 는  $A^c \rightarrow A^c$ 의 전이율을 나타내며  $U^0$ 는  $A^c \rightarrow A$ 의 전이율을 나타낸다. 행렬  $U$ 를 이용하여  $E(T_k)$ , 즉  $k$ 개의 패킷이 존재하는 시점부터 정보완성에 걸리는  $N$ 개의 패킷이 도착할 때까지의 평균시간은 다음과 같이 구할 수 있으며, 따라서  $E(T_0) + 1/\mu$ 이 본 연구에서 구하고자 하는 평균 정보처리시간을 구할 수 있다(Lee, 2006).

$$E(T_k) = u_k (-U)^{-1} e, \tag{4}$$

여기서  $U = q_{ij}$ ,  $i, j = 0, 1, \dots, N-1$ ,  $e = (1, 1, \dots, 1)^T$ 이며  $u_k$ 는  $k+1$ 번째 항이 1이고 나머지는 0인  $1 \times N$ 의 벡터이다.

#### 4. 이산시간 모형의 평균 정보처리 시간

본 장에서는 이산시간상에서 패킷의 도착과 소멸을 통하여 완성되는 평균 정보처리 시간을 구하기 위하여 이산시간 마코프 체인으로 모형화한다. 디지털 통신시스템에서 패킷의 도착간격은 주로 연속적으로 이루어진다. 또한 패킷의 도착은 슬롯의 경계에서 이루어지는 것이 아니라 슬롯의 중간에서 발생하지만 시스템에서 측정하는 패킷의 도착은 슬롯 경계에서만 이루어진다(Bruneel and Kim, 1993). 따라서 도착이나 소멸이 슬롯의 중앙에서는 발생하지 않고 슬롯의 경계에서만 발생하는 것으로 가정한다. 특히 패킷의 도착이 <Figure 4(a)>와 같이 슬롯 경계의 뒤쪽에서 이루어진다고 가정하는 지연도착모형(late arrival model)과, <Figure 4(b)>와 같이 슬롯 경계의 앞쪽에서 이루어진다고 보는 조기도착모형(early arrival model) 두 가지를 모두 고려한다. 시스템에 도착한 패킷은 소멸하거나 시스템의 정보 구성에 불필요하게 되어 의미를 상실하게 될 수도 있다. 지연도착모형의 경우에는 패킷의 도착 후 소멸이 발생한다고 가정하며, 따라서 슬롯에서의 패킷 수는 도착 패킷까지 포함한다. 조기도착모형에서는 패킷의 소멸이 발생한 후 패킷의 도착이 이루어진다고 본다.

본 이산시간 모형에서는 다음과 같은 가정을 도입한다.

- (1) 한 시스템에서 슬롯의 길이는 일정하다.
- (2) 데이터의 도착은 한 슬롯에서 최대 한 번만 발생하며 그 확률은  $\lambda$ 이다.

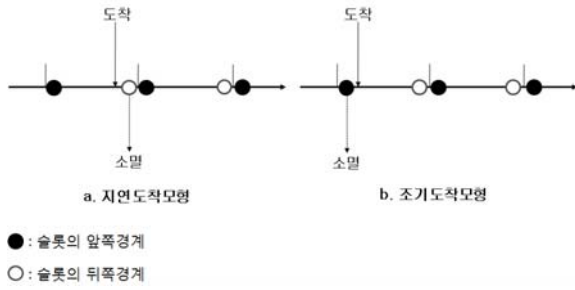


Figure 4. Packet arrival model

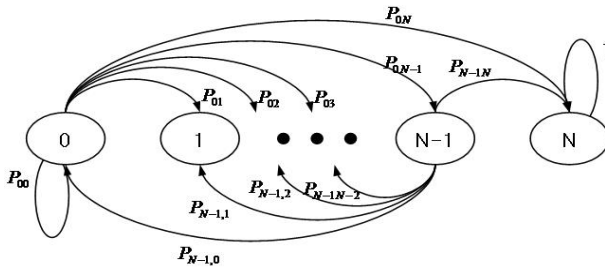


Figure 5. Absorbing discrete time Markov chain

- (3) 시스템의 상황이나 변화된 주변 환경으로 인하여 수집된 데이터가 불필요해지는 상황이 발생하게 되면 패킷은 확률  $\alpha$ 로 소멸한다.
- (4) 패킷이  $N$ 개 모이면 시스템은 이들 패킷을 처리하여 정보를 완성하며 처리시간은 모수  $\mu$ 의 기하분포를 따른다.

다음과 같은 기호를 정의한다.

- $X_n$  :  $n$ 번째 슬롯의 경계에서 존재하는 패킷의 개수,  $n = 1, 2, 3, \dots$ ,
- $T_k$  :  $k$ 개의 패킷이 있을 때 정보구성에 필요한 나머지  $N-k$ 개의 패킷만큼 쌓일 때까지 걸리는 시간(슬롯 수).

이산시간 확률과정  $\{X_n, n = 1, 2, \dots\}$ 는 다음 <Figure 5>와 같은 흡수 마코프체인을 형성한다.

지연도착모형은 슬롯의 끝에서 관찰하는 패킷의 도착과 소멸 중 도착을 우선시하기 때문에 소멸가능한 패킷의 수는 도착한 패킷의 수까지 포함한다. 상태 0에서 상태  $i$  ( $i = 0, 1, \dots, N$ )로 전이되는 과정을 고려하여 지연도착모형의 1단계 전이 확률  $P_{ij}$ 의 일반식을 식 (5)과 같이 계산한다.

$$P_{jk} = \sum_{i=k-j}^{N-j-1} \lambda g_i \times \binom{i+j}{i+j-k} \alpha^{i+j-k} (1-\alpha)^k, \quad (j < k), \quad (5)$$

$$P_{jk} = (1-\lambda) \times \binom{j}{j-k} \alpha^{j-k} (1-\alpha)^k + \sum_{i=1}^{N-j-1} \lambda g_i \times \binom{i+j}{i+j-k} \alpha^{i+j-k} (1-\alpha)^k,$$

$$(j \geq k), \\ (j, k = 0, 1, \dots, N-1),$$

$$P_{jN} = \sum_{i=N-j}^{\infty} \lambda g_i,$$

$$P_{NN} = 1.$$

조기도착모형의 경우에는 슬롯의 끝에서 패킷의 수를 계산할 때 소멸을 우선 고려하기 때문에 소멸이 먼저 발생하고 도착이 발생하는 경우로 패킷의 수를 계산한다. 조기도착모형에서의 1단계 전이확률 일반식을 계산하면 식 (6)와 같다.

$$P_{jk} = \sum_{i=k-j}^k \left[ \lambda g_i \binom{j}{j-k+i} \alpha^{j-k+i} (1-\alpha)^{k-i} \right], \quad (j < k), \quad (6)$$

$$P_{jk} = (1-\lambda) \binom{j}{j-k} \alpha^{j-k} (1-\alpha)^k + \sum_{i=1}^k \left[ \lambda g_i \binom{j}{j-k+1} \alpha^{j-k+1} (1-\alpha)^{k-1} \right], \quad (j \geq k),$$

$$(j, k = 1, 2, \dots, N-1),$$

$$P_{j0} = (1-\lambda) \binom{j}{j} \alpha^j (1-\alpha)^0, \quad (j = 0, 1, \dots, N-1),$$

$$P_{0j} = \lambda g_j, \quad (j = 1, 2, \dots, N-1),$$

$$P_{0N} = \sum_{i=N}^{\infty} \lambda g_i,$$

$$P_{jN} = \sum_{i=N-j}^{\infty} \sum_{h=0}^j \lambda g_{i+h} \binom{j}{h} \alpha^h (1-\alpha)^{j-h}, \quad (j = 1, 2, \dots, N-1),$$

$$P_{NN} =$$

식 (5)과 식 (6)에 의해 1단계 전이확률행렬  $P$ 를 구성하면 이는 다음 식 (7)과 같다.

$$P = \begin{bmatrix} U & U^0 \\ O & 1 \end{bmatrix}. \quad (7)$$

식 (7)의 행렬  $U$ 는 일시상태들, 즉 상태  $0, 1, \dots, N-1$ 간의 전이행렬이고, 열벡터  $U^0$ 는 일시상태로부터 흡수상태, 즉  $N$ 으로 전이하는 확률의 행렬이다. 식 (4)와 유사하게 다음 식 (8)를 통해 본 연구에서 구하고자 하는 평균 정보처리시간  $E(T_0) + 1/\mu$ 를 구할 수 있다(Lee, 2006).

$$E(T_k) = u_k (I - U)^{-1} e, \quad (8)$$

여기서  $U = P_{ij}$ ,  $i, j = 0, 1, \dots, N-1$ ,  $e = (1, 1, \dots, 1)^T$ 이며  $u_k$ 는  $k+1$ 번째 항이 1이고 나머지는 0인  $1 \times N$ 의 벡터이다.

## 5. 정보가 두 가지 이상 존재할 때의 평균 처리시간

정보처리시스템은 하나의 동작만을 하기도 하나 시스템의 특

성에 따라 복수의 동작을 수행하기도 한다. 이에 따라 정보를 구성하기 위해 정보처리시스템에 도착하는 데이터는 서로 다른 여러 종류의 정보를 처리하는 경우도 있다. 따라서 본 절에서는 정보의 종류가 두 가지 이상일 때의 평균정보처리시간을 구한다. 정보처리시스템은 정보구성이 완료된 동작부터 차례로 하나씩 수행한다.

우선 정보의 종류가 두 가지인 경우를 고려한다. 각각의 정보처리시간은 평균이  $1/\mu_1, 1/\mu_2$ 이다. <Figure 6>~<Figure 8>은 두 가지 형태의 동작을 수행하기 위해서 정보를 구성하기 위한 패킷의 개수가 각각  $N_1$ 과  $N_2$ 개일 때의 정보처리과정의 예를 보이고 있다. 첫 번째 동작을 위한 정보의 처리가 종료된 시점에서 두 번째 동작을 위하여 존재하는 패킷의 개수를  $k$ 라고 하면, <Figure 6>은  $k=N_2$ 인 경우를 보이고 있다. 이 때 두 번째 동작은 첫 번째 동작의 처리가 끝날 때까지 대기하고 있다. <Figure 7>는  $k=0$ , <Figure 8>은  $1 \leq k \leq N_2 - 1$ 인 경우를 각각 나타낸다.

패킷의 개수가 0에서 정보를 구성하기 위한 개수를 만족할 때까지 패킷의 개수  $k$ 를 거치는 평균시간은 일시상태 0에서 출발하여 흡수될 때까지 일시상태  $k$ 에 머문 평균시간을 의미

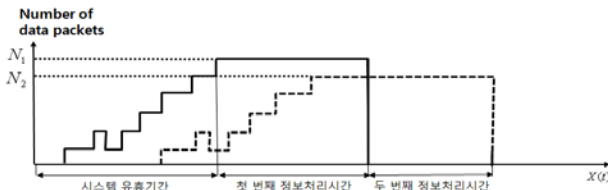


Figure 6. Information processing time in case of two kinds of information (when  $k = N_2$ )

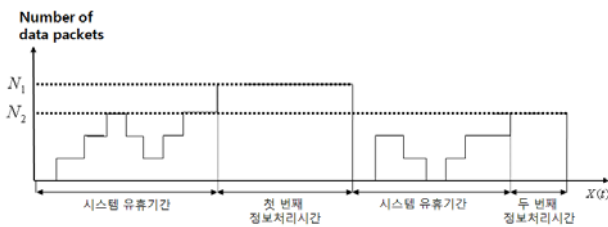


Figure 7. Information processing time in case of two kinds of information (when  $k = 0$ )

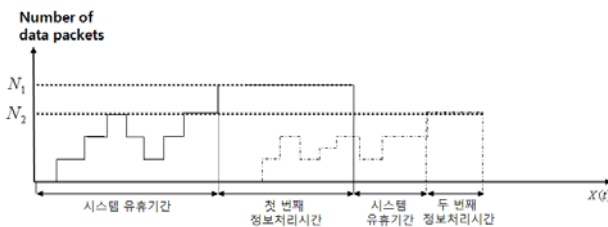


Figure 8. Information processing time in case of two kinds of information (when  $1 \leq k \leq N_2 - 1$ )

하며 이는 연속시간 모형의 경우 행렬  $(-U)^{-1}$ 의  $(0, k)$ 번째 원소, 즉  $(-U)_{0,k}^{-1}$ 이며 이산시간 모형의 경우에는 행렬  $(I-U)^{-1}$ 의  $(0, k)$ 번째 원소, 즉  $(I-U)_{0,k}^{-1}$ 이다. 이제 첫 번째 동작을 위한 정보의 처리가 종료된 시점에서 두 번째 동작을 위하여 존재하는 패킷의 개수가  $k$ 일 확률을  $\nu_k$ 라고 하자. 이 확률은 <Figure 9>와 같은 마코프체인에서 임의시점에서의 안정상태 확률에 해당한다. 이 마코프체인에서 상태 0에 진입하는 시점을 재생성점(regenerative points)로 보고 재생보상정리를 활용하면 확률  $\nu_k$ 는 다음과 같다.

$$\nu_k = \begin{cases} (-U)_{0,k}^{-1} / (E(T_0) + 1/\mu_1), & \text{연속시간모형의 경우,} \\ (I-U)_{0,k}^{-1} / (E(T_0) + 1/\mu_1), & \text{이산시간모형의 경우,} \end{cases} \quad k = 0, 1, \dots, N-1 \quad (9)$$

두 동작이 모두 완료되기까지의 시간을  $E(T^*)$ 라고 하면 이는 다음과 같이 구할 수 있다.  $E(T_0|N=N_1)$ 는 첫 번째 동작을 위한 정보를 구성하기 위한 패킷의 개수  $N_1$ 을 만족하기까지 걸리는 시간을 의미하고  $E(T_k|N=N_2)$ 는 첫 번째 정보처리가 끝난 시점에서 두 번째 동작을 위한 정보 패킷이  $k$ 개가 있을 때 두 번째 동작을 구성하기 위한  $N_2$ 을 만족하기까지 걸리는 시간을 의미하며, 식 (4)와 식 (8)를 이용해 구할 수 있다.

$$E(T^*) = E(T_0|N=N_1) + \sum_{k=0}^{N_2} E(T_k|N=N_2)\nu_k \quad (10)$$

$$+ \frac{1}{\mu_1} + \frac{1}{\mu_2}.$$

필요한 동작의 개수가 세 개 이상인 경우의 평균 정보처리 시간도 식 (10)를 확장하여 다음과 같이 쉽게 구할 수 있다.

$$E(T^*) = E(T_0|N=N_1) + \sum_{k=0}^{N_2} E(T_k|N=N_2)\nu_k \quad (11)$$

$$+ \sum_{k=0}^{N_3} E(T_k|N=N_3)\nu_k + \dots + \frac{1}{\mu_1} + \frac{1}{\mu_2} + \frac{1}{\mu_3} + \dots$$

## 6. 수치 실험

본 장에서는 앞선 제 3, 4장에서 다룬 정보처리모형의 수치에

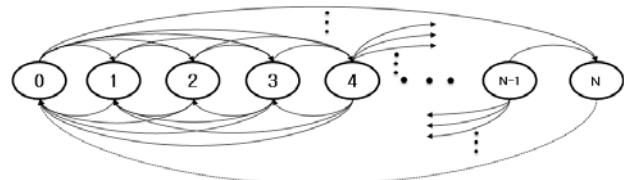


Figure 9. Markov chain for computing steady-state probabilities in arbitrary time

제를 이용하여 시스템의 정보처리 시간에 영향을 주는 모수와 그 강도를 파악하여 시스템 설계 시 어떤 요소를 고려할 것인지 도출한다. 즉, 평균정보처리시간에 결정적인 영향을 미치는 모수를 파악하고 이것이 시스템의 성능에 어떤 영향을 미치는지 분석하고 전체 시스템의 빠른 정보처리시간을 위한 필요 수준으로 성능향상을 이루기 위한 방법을 제시할 필요가 있다. 연속시간 모형이나 이산시간 모형 모두 모수들이 성능에 미치는 영향에 있어서는 근본적으로 차이가 없을 것이므로 본 수치실험은 연속시간 모형의 경우만을 대상으로 한다.

<Figure 10>은 정보를 구성하는데 필요한 패킷의 수  $N=10$  일 때 각 모수의 변화에 따른 평균정보처리시간의 변화를 나타내고 있다.  $1/\mu=0.5$ 로 두고 데이터의 도착률  $\lambda=10, 20, 30, 40, 50$ 으로, 손실률  $\alpha=1, 2, 3, 4, 5$ 로 변화시키면서 평균정보처리시간( $E(T_0)$ )을 계산하였다. <Figure 11>는 <Figure 10>에서의 실험과 동일하지만 다만  $1/\mu=0.25$ 로 감소할 때의 평균정보처리시간을 계산한 결과이다.

두 그림을 비교해 보면, 데이터의 도착률( $\lambda$ )이 작지 않다면 데이터 손실률( $\alpha$ )이 평균정보처리시간에 미치는 영향은 미미함을 알 수 있다. 그리고  $\lambda=10, \alpha=1$ 의 경우 <Figure 10>과 <Figure 11>를 비교하면,  $\mu$ 가 2(<Figure 10>)에서 4(<Figure 11>)로 두 배로 증가함에 따라 평균 정보처리시간이 약 0.05 감소하였다. 그러나  $\lambda$ 가 10에서 20으로 두 배로 늘어나면 평균정보처리시간은 0.2감소, 20에서 40으로 늘면 0.1이 감소하였고 그 폭은  $\alpha$ 가 높아질수록 더 커진다. 이에 따라 구성된 정보를 처리하는 데 걸리는 시간( $1/\mu$ )보다 데이터의 도착률( $\lambda$ )이 정보를 구성하고 처리하는 데 걸리는 평균정보처리시간에 더 많은 영향을 준다는 것을 알 수 있다.

한편 <Figure 12>은 정보를 구성하는 필요 패킷수를  $N=5, 7, 10, 12, 15$ 로 증가함에 따라 변화하는 평균정보처리시간을 나타내고 있다. 단, 여기서  $\lambda=30, \alpha=2, 1/\mu=0.05$ 로 한다. <Figure 12>을 보면 정보를 구성하는데 필요한 패킷수( $N$ )가 증가함에 따라 평균정보처리시간은 크게 증가하는 것을 알 수 있다.

결론적으로, 정보처리시간에 중요한 영향을 미치는 모수는

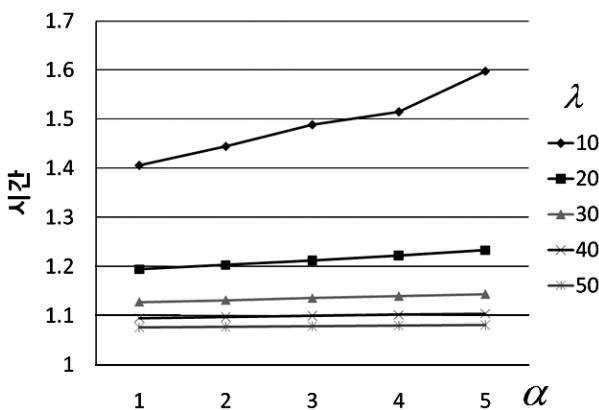


Figure 10. Changes of the average information processing time under various arrival rates and loss ratios( $\mu=2$ )

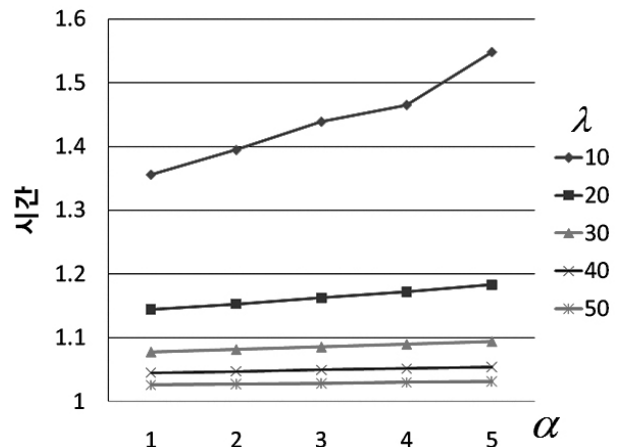


Figure 11. Changes of the average information processing time when reducing the processing time( $\mu=4$ )

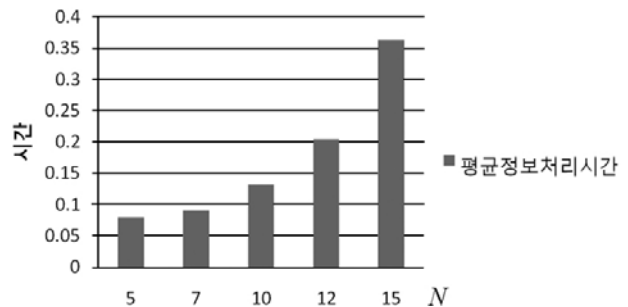


Figure 12. Changes of the average information processing time for various number of required packets

데이터의 도착률( $\lambda$ )과 필요패킷개수( $N$ )이며,  $\mu$ 의 영향은 크지 않았다. 시스템 설계의 측면에서 보면 데이터의 도착률을 높이는 것은 주변으로부터 보다 많은 데이터를 수집하도록 하는 것이며, 필요패킷수( $N$ )는 정보구성을 위한 데이터의 양을 의미하기 때문에 정보의 품질과 관련이 있다. 센서네트워크의 예를 든다면, ①정보처리기의 CPU 성능을 높여서  $\mu$ 를 높이는 것은 정보처리시간에 큰 기여를 하지 않고, 이보다는 오히려 ②센서를 더 많이 살포하거나 ③센싱주기를 짧게 하고, ③정보의 품질은 다소 떨어질 수 있으나 정보처리개시에 필요한 패킷수를 낮추는 방법을 사용하는 것이 정보를 조기에 구성하고 처리하여 정보를 전송하는 데에 좋다는 결론을 얻을 수 있다.

### 7. 결론 및 추후 연구과제

본 연구에서는 유비쿼터스 환경에서 정보처리시스템이 기능을 수행하는 데에 시간제약이 있는 정보처리 모형을 수립하고, 정보를 구성하고 처리하는 평균정보처리시간을 구하였다. 또한 데이터의 도착과 소멸을 나타내는 확률모형에 있어서 기존의 연구결과보다 일반적인 경우로 확장하여 모형을 수립하였으며 특히 연속시간과 이산시간으로 구분하여 모형을 분석

하였다.

평균정보처리시간의 수치예제를 통해 필요패킷의 개수와 평균정보처리시간의 관계를 분석하였으며 분석결과를 이용하여 시간제약을 가지는 시스템에서 정보의 질과 정보처리시간의 상호보완을 하여 사용자의 요구에 적합한 시스템을 설계할 수 있음을 보였다.

본 연구결과를 바탕으로 데이터의 도착과 소멸이 포아송과정이나 지수확률분포를 따르지 않는 일반적인 경우의 모형으로 확장할 수 있다. 또한 유비쿼터스와 같은 복합네트워크에서 전체 시스템이 가지는 시간제약을 만족하기 위한 패킷의 수를 결정하는 문제를 고려할 수 있다. 특히 정보의 품질을 양적인 요소만이 아니고 질적인 요소도 고려하고, 데이터 전송에 지연(delay) 현상이 발생하여 불일치한 정보를 받는 경우 시스템이 시간 제약 이내에 정보를 처리할 수 있도록 모수를 제어하는 방법을 연구하고 있는 중이다.

## 참고문헌

- Bruneel, H. and Kim, B. (1993), *Discrete-Time Models for Communication Systems Including ATM*, Kluwer Academic Publishers, Boston, U.S.A.
- Chakravarthy, S., Jacob, J., Pandrangi, N., and Sanka, A. (2002), WebVigil : An Approach to Just-In-Time Information Propagation in Large Network-Centric Environments, *The 2nd International Workshop on Web Dynamics in conjunction with 11th International World Wide Web Conference*, Honolulu, Hawaii, 301-318.
- Choi, H., Park, N., Lee, H., Seo, Y., and Shin, J. (2005), A Study on Information System Improvement for Air Logistics SCM, *International Journal of Korea Association of Information Systems*, **14**(3), 63-70.
- Hedley, Y., Younas, M., James, A., and Sanderson, M. (2006), Sampling, Information Extraction and Summarization of Hidden Web Databases, *Data and Knowledge Engineering*, **59**(2), 213-230.
- Hur, S., Lee, H., Shin, D., and Lee, W. (2007), A Time-Constrained Information Processing Model in Ubiquitous Environments, *ETRI Journal*, **29**(4), 489-496.
- Lee, H. W. (2006), *Queueing Theory*, Sigma Press, Seoul, Korea.
- Nam, J., Hur, S., Joo, K., and Shin, D. (2009), Performance Analysis of a Composite Service Providing System in a Context-Aware Computing Environment, *Journal of the Korean Institute of Industrial Engineers*, **35**(1), 51-57.
- Shin, D., Hur, S., Lee, H., Nam, J., and Jung, J.-Y. (2010), An Analytical Model for Delivery Evaluation of Multimodal Contents in Pervasive Computing, *Computers in Industry, In press*.
- Sun, S. and Yen, J. (2005), Information Supply Chain : A Unified Framework for Information-Sharing, *Intelligence and Security Informatics*, **59**, 422-428.
- Tolstikov, A., Xiao, W., Biswas, J., Zhang, S., and Tham, C. (2007), Information Quality Management in Sensor Networks based on the Dynamic Bayesian Network Model, *Intelligent Sensors, Sensor Networks and Information, 3rd International Conference on*, 751-756.
- Yen, J., Fan, X., and Volz, R. (2004), Proactive Communications in Agent Teamwork, *Lecture Notes in Computer Science* **2922**, 271-290.
- Zhang, Y. and Ji, Q. (2006), Active and Dynamic Information Fusion for Multisensor Systems with Dynamic Bayesian Networks, *IEEE Transactions on Systems* **36**, 467-472.