

Flexible Automation을 위한 네트워크 관리 시스템 구조에 관한 연구

正會員 姜 文 植* 正會員 李 재 용** 正會員 李 相 培***

A research on the Network Management Architecture for Flexible Automation

Moon-Sik Kang*, Jae-Yong Lee**, Sang-Bae Lee*** *Regular Members*

要 約

본 논문에서는 Flexible Automation을 위한 네트워크 운영과 관리체제를 분석하고 이에따른 요구사항을 고려하여 네트워크 관리 시스템을 설계하였다. 네트워크 관리기능은 자원을 제어하고 관리하는 필수적인 기능으로, 관리 서비스를 제공할 수 있는 방법과 관리구조를 제안하고 해석적 방법과 큐잉모델을 이용하여 전송 데이터의 증가에 따른 지연시간 분포를 구하고 성능을 분석한다. 성능관리를 위한 성능을 측정하고 그 결과를 분석하여 설계된 관리시스템의 운영에 대한 타당성을 확인하였다.

Abstract

In this paper, the network management system is implemented based on the analysis of requirements and network operation and management for Flexible Automation. Network management is necessary, which controls and supervises the network resources in the communication network.

By means of both analytical methods and queueing model, the delay time distributions due to the increasement of transmission data are obtained and analyzed. The operations of this network management system are certified through the test environments with the network adaptor and softwares for each layer.

I. 서 론

최근 네트워크 사용이 점차 증대되면서 네트워크

제공자는 효율적으로 시스템을 관리해야 하는 필요성이 대두되었고 사용자들은 보다 향상된 서비스와 품질을 요구하게 되었다. 또한 네트워크의 크기가 증가되고, 복잡성이 증대되어 네트워크 운영을 위한 제어가 점점 어려워지고, 사용자가 필요로 하는 정보는

*江陵大學校 電子工學科

**浦港工科大學 電子計算學科

***延世大學校 電子工學科

論文番號 : 94-20

다양하게 되어 네트워크의 효율적인 관리의 필요성이 더욱 증대되었으며, 국제 표준화 기구(ISO)에서는 네트워크 관리를 위해 OSI 7 계층 구조를 대상으로 관리 표준모델을 제시하고 있다. 네트워크 관리는 OSI 모델의 응용계층과 각 계층에 존재하는 계층관리로 구성되며, 네트워크 관리 측면에서 보면, 네트워크는 하나 이상의 네트워크 관리자(manager)와 이 관리자로 부터 통제를 받는 여러 피관리자(agent)로 구성된다.^[4, 19, 20]

CMIP(Common Management Information Protocol)는 OSI 프로토콜로 상호 연결된 개방형 시스템간의 기본적인 관리 정보를 교환하기 위한 표준화된 관리 프로토콜로서 ROSE(Remote Operation Service Element)를 통해 MIB를 접근하며, 관리 정보는 ASN.1을 사용하여 노드간에 전달되나, 하나의 관리행위가 몇개의 RPC를 요구하기 때문에 비효율적이다.^[11] OSI 관리기능에는 장애관리(Fault management), 구성관리(Configuration management), 성능관리(Performance management), 회계관리(Accounting management), 그리고 보안관리(Security management) 등으로 분류된다.^[11] 생산자동화에 관한 LAN의 표준 규격으로 제안된 자동화 네트워크 프로토콜인 MAP(Manufacturing Automation Protocol)은 자동화를 위한 표준 통신 서비스들의 집합을 정의하며, ISO(International Organization for Standardization)에 의해 국제 표준으로 채택되었으므로 이러한 네트워크를 대상으로 하여 관리 시스템을 설계한다.

본 논문에서는 Flexible Automation을 위해 네트워크 운영과 관리체제에 대한 분석을 하고 구성관리와 장애관리 그리고 성능관리를 위한 관리 프로세스를 구현하였다. 성능관리를 위한 성능을 측정하고 그 결과를 분석하였다.

II. 네트워크 관리 시스템

2.1 네트워크 관리 모델^[3, 4]

OSI 관리표준은 객체 모델링(Object modelling)에 대한 개념으로 설명가능하며, OSI 관리 모델은 [그림 1]과 같이 하나 혹은 여러개의 관리 프로세스(관리자)와 피관리 프로세스(피관리자) 간의 상호작용으로 이해될 수 있다.^[3] 피관리 프로세스는 자신에게 속해있는 여러 피관리 객체들을 책임지며 피관리

객체들은 관리의 관점에서 본 모든 자원들이 추상적 표현으로 설명될 수 있다. 피관리 객체는 물리적인 개체나 추상적인 개체를 표현할 수도 있으며 이러한 모든 피관리 객체 정보들은 MIB(Management Information Base)라고 하는 개념적인 저장소에 저장된다. 관리 프로세스는 피관리자에 의해서 유지되는 정보들을 CMIS(Common Management Information Service)에서 제공되는 서비스로 조작되고 CMIP운명을 통해 전달하게 된다. MIB에서 정보들은 객체에 관계된 속성(Attribute)로 구성되며 이들 속성들은 자체의 환경에 따라 특정 값을 가지게 된다. CMIS는 속성들의 값을 추출(Get), 변경(Set)하거나 객체를 생성(Create), 소멸(Delete)시키는 서비스를 제공하며 또한 사건보고(Event report)서비스를 이용한 보고기능도 가진다. 따라서 이들 서비스들을 이용하여 피관리 객체 데이터에 대한 조작과 보고 기능을 제공한다. CMIS에서는 피관리 객체들을 제어하기 위한 동작(action) 서비스도 제공한다.

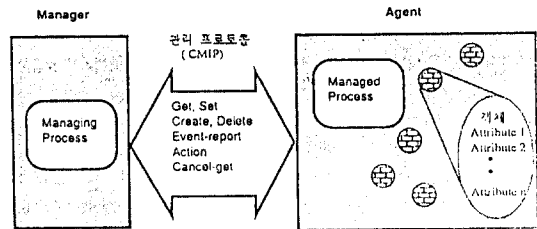


그림 1. OSI 관리의 객체 모델

2.2 CMIS/CMIP을 이용한 관리

ISO에서는 시스템 관리에서 제공하는 기능의 범주를 SMFA(Systems Management Functional Area)라 정의하고 여기에서는 성능, 구성, 고장, 회계, 보안 관리기능으로 구분한다. 이중 고장 관리기능은 ISO 관리 환경에서 비정상적인 동작을 조기발견과 검출한 장애의 원인 파악과 회복을 가능하게 하며 성능 관리기능은 관리자가 시스템의 성능에 대해 감동하거나 평가할 수 있는 능력을 제공하는 OSI 자원. 즉 관리객체의 행동이나 통신 활동에 대해 관리의 효율성을 제공한다. 그러나 관리기능 사이의 관계는 사용자 관점에서 분류되어 각 기능의 범위를 정확히 정의하여 구현하기에는 어려움이 따른다.

본 연구에서는 영국의 UCL 대학에서 개발한 OSI-based 네트워크 관리 시스템인 OSIMIS를 참고로

으며, 이 시스템은 ISODE 7.0 개발환경을 이용하고 있으며 관리 정보로서 ISO 트랜스포트 계층을 이용한다.

2.3 MMS를 이용한 네트워크 관리

MMS(Manufacturing Message Specification)는 ISO의 OSI 참조모델의 응용계층에 위치하는 통신용 표준으로서 MAP 네트워크와 같은 근거리 통신망으로 접속되어 있는 생산환경의 여러 장비들간의 원활한 정보교환을 목적으로 한다. 이는 생산환경의 일반적인 통신환경을 규정하고 있는 핵심 규격이고 공상의 특수한 물리적 장치인 장치제어, 프로그래밍, 컨트롤러, 수치제어 그리고 로봇등의 등장 통신에 대해서는 각각 표준화 기관의 주도로 MMS companion standard가 제정되고 있다. MMS에서는 추상적인 장치인 VMD(Virtual Manufacturing Device)라는 추상적 모델을 정의하여 서비스들이 일관성있게 이해될 수 있도록 하였다. VMD는 외부에서 볼 수 있는 지면을 모델링하고 있어 그 사용자에게 제어, 감시를 위한 기능과 자원들을 제공하게 된다.

2.4 네트워크 관리 구조

네트워크에 연결된 노드들을 관리하기 위하여 노드들간의 정보를 교환하기 위한 네트워크 관리구조가 필요하다. OSI에서는 네트워크 관리구조를 위해 표준화된 CMIP 프로토콜을 제정하였고 이 CMIP 프로토콜은 7계층의 응용계층으로 ROSE를 요구하고 있다. MAP 3.0 네트워크 관리 요구사항은 모든 네트워크 관리 기능을 지원하기 위하여 Initialize, Terminate, Abort, Confirmed Get, Set, Action, Compare, Block, Event Report 과 같은 동작이 필요하다.

이러한 관리 동작은 기존의 통신 프로토콜 구조에 독립적이어서 매우 일반적이라고 할 수 있다. "initialize", "terminate" 그리고 "abort" 동작은 MMS 또는 FTAM 과 같은 응용 서비스 요소보다는 네트워크 관리를 위한 연결을 설정할 때 사용된다. "Get"과 "Set" 동작은 각각 MIB내에 있는 네트워크 관리 변수들을 추출하거나 변경할때 사용된다. "action" 동작은 서비스에서 보통 일단개로 이어지는 어떤 동작의 수행을 요구하게 되는데 예를 들면 지미의 상태값이 될 수 있다. "compare" 동작은 사용자 제공한 값과 지정된 인터페이스의 값을 비교할때 사용된

다. 또한 "block" 동작의 기능은 다수의 동작을 하나의 그룹으로서 수행하도록 클라이언트가 서버에게 요구할 때 사용된다. MAP환경에서의 네트워크 관리구조로 MAP의 응용계층에 위치하는 MMS의 트랜잭션 객체를 사용하였다. MMS에서 트랜잭션 객체는 특성상 동적이고 VMD에 의해 관리된다. 트랜잭션 객체는 서비스 요청을 받을때 만들어지며 서비스에 대한 응답을 클라이언트가 받게될 때 자동적으로 소멸된다. "invoke ID"라고 하는 고유 식별자가 하나의 트랜잭션 객체와 연관된다.

III. 네트워크 관리 프로세스의 구현

3.1 구성관리

관리 시스템에 있어서 구성관리는 직접적으로 피관리 대상의 정보를 얻어내는 역할보다는 그러한 역할을 수행할 수 있도록 한다는 의미가 크므로, CMIS/CMIP 보다는 FTAM이나 하위계층의 ACSE나 ROSE등을 이용해서 다른 피관리 대상의 구성에 대한 정보를 가져오는 역할을 한다. 관리 시스템이 시작할때, 처음에 MO(Management Objects)를 초기화 하거나, 사용자간 알기 쉬운 이름으로 되어 있는 MO들을 시스템에서 사용되는 MO의 set들과 연결하여 속는 등의 작업에는 CMIS/CMIP 등 프로토콜의 이용이 필요하다.

3.2 제안된 관리구조를 이용한 성능관리

성능 관리기능중의 하나는 앞으로의 컴퓨터 네트워크의 동작을 예상하고 성능분석을 위해 통계 자료를 수집하는 일이며, 네트워크를 상기간에 걸쳐 평가할 해야 한다. 많은 통계자료에 대한 정이가 성능관리에 관계되는 부분에서 논의되어 왔으나 본 논문에서는 QOS 파라미터들의 추정과 감시를 위해 평균값, 최대값, 최소값, 표준편차에 대하여 고려한다.

성능 파라미터 추정 방법과 참조 모델을 설명하면 다음과 같다. ISO/OSI 계층구조에서 측정 표본들을 얻는 방법과 위치가 매우 중요하며, 구현시 이들 요소들을 사용하는 방법에 따라 결과치가 다르게 나올 수 있기 때문에 신중히 고려하여야 한다. 시스템에 독립적인 참조 샘플링 포인트(Reference Sampling Point)가 필요하게 되며, 본 논문에서는 참조 샘플링

포인트로서 MAP에서의 응용계층에 위치하는 MMS를 선택하였다. 성능관리를 위한 파라미터를 정의하면 다음과 같다.^[16]

- 연결확립 지연시간

연결확립은 다른 MMS 서비스들이 호출되기 전에 성공적으로 완료되어야 하는데 이 연결확립 지연시간은 MMS 사용자가 M-INITIATE, 요구 프리미티브를 보내서 이에 대응되는 M-INITIATE, 확인 프리미티브를 받을때까지 걸리는 시간을 말하며 ACSE (Association Control Service Element)를 이용한다.

- 데이터 처리량(Data Throughput)

데이터 처리량은 연결확립이 끝난후 데이터 전송 단계에서 단위시간 t동안 전송되는 사용자 데이터의 bit수를 말한다. 이 성능 파라미터는 데이터 전송 동작에 대한 효율의 좋은 척도로서 사용된다.

- 연결해제 지연시간

이는 MMS 사용자가 M-CONCLUDE, 요구와 대응되는 M-CONCLUDE, 확인 프리미티브 사이에 소요되는 시간을 말하며 ACSE를 이용한다.

- 응답시간

많은 응용에서 상대방에게 정보를 요구해서 그에 대한 응답메세지를 받는 경우가 많은데 응답 시간이란 사용자가 요구를 보낸 시점에서 상대방으로부터 응답 메세지가 도착할 때까지의 시간을 말한다. 응답 시간을 측정하기 위해 사용되는 방법은 다음 [그림 2]와 같이 도시될 수 있다.

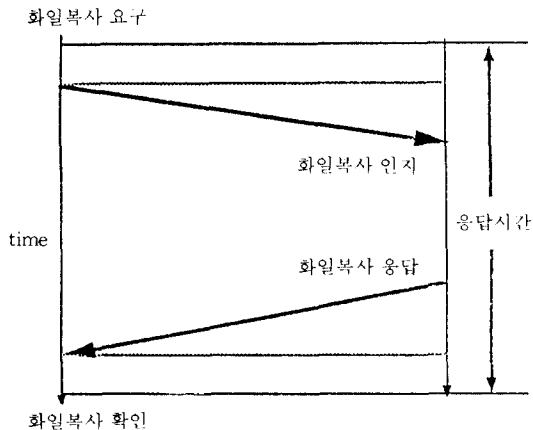


그림 2. 응답시간

MAP시스템의 성능을 측정하기 위하여 크게 세부분으로 분류하였다. 첫째는 성능 측정을 위한 환경을 설정하는 부분으로 실제 성능에 영향을 줄 수 있는 요소들인 메세지 크기, 슬롯 타임등을 변화시킨다. 둘째는 성능 측정을 하는 사용자가 제시한 명세에 맞게 메세지를 생성해서 이들을 전송하는 부분이다. 셋째는 메세지 전송을 통하여 측정된 성능 파라미터를 모아서 이들의 최대값, 최소값, 평균값 그리고 표준편차등의 통계치를 작성한다. 측정된 값들은 개시자와 응답자 양쪽에서 수집되며 한번이상 측정된 것들은 최대, 최소값등 통계자료들을 구한다.

사용자 인터페이스 부분으로는 첫째, 두개의 MAP 시스템간의 응답 시간을 측정하기 위해 사용되는 부분이다. 연결 확립을 요구한 개시자가 보내는 메세지는 상대방 시스템내에 있는 화일명으로서 이 요구 메세지를 보내서 그에 대응되는 화일을 응답자로부터 가져오는때까지 걸리는 시간을 측정한다. 둘째, 응답자에게 개시자로부터 처리율을 측정하기 위한 부분으로 이따 연결확립이 설정된 상태에서 메세지를 전송하는데 필요한 데이터 처리량을 측정한다. 데이터 처리량은 전송된 사용자 데이터 바이트수를 전송동안 경과된 시간으로 나눔으로서 측정하였다. 셋째, 연결을 설정하고 해제하는데 걸리는 시간을 측정하는 부분으로서 어떠한 사용자 메세지도 전송되지 않는다.

IV. 실험 및 결과고찰

4.1 실험 환경

성능 파라미터를 실제로 측정하고 관찰하기 위한 실험환경으로 CONCORD MAP 시스템을 이용하였고 3개의 퍼스널 컴퓨터를 호스트로 사용하였다. 네트워크에 연결된 각 노드들은 Series 1200(PC bus) Controller, Concord 모뎀 그리고 MAP 소프트웨어 스택(MAPware 3.0 software)으로 구성되어 있다. Series 1200 Controller는 IBM PC나 호환기종에서 사용되는 고속의 MAP communication controller로서 주요 구성요소들로는 Intel 80186 마이크로 프로세서와 IEEE 802.4 MAC 계층을 구현한 Motorola 68824 칩등이 있으며 Concord 모뎀은 계층 1(물리적 계층)의 기능을 제공하는 외부모뎀이다. 또한 MAP 소프트웨어 스택에는 계층 2(데이터링크 계층)부터 계층 7(응용 계층)까지를 포함하고 있으며 Series 1200 Controller에 있는 RAM에 다운로드되어서 사용한다.

CMIS/CMIP은 떨어져 있는 시스템에서 원하는 데이터를 어떠한 형태에 맞추어 가져오기 위한 프로토콜이나, 구성관리에서는 직접 ACSE와 ROSE를 이용하여 화일에 접근하도록 하였다. 기본 프로그램은 워크스테이션상에서 C 언어를 이용하여 구현하였다. 기타 이용한 라이브러리들은 XLIB, ISODE-7.0 등이다. 네트워크의 구성을 표현하기 위해서는 RFC 952(DOD INTERNET HOST TABLE SPECIFICATION)에 나타나 있는 형식을 이용하였다. 프로그램은 XLIB을 이용하여 사용자 집면 창(User interface window)을 만들고 호스트 테이블에서 초기 최상위 레벨의 그림을 작도하고, 이때 원격 도메인(remote domain)의 네트워크를 요구하면 isode-7.0을 이용하여 원하는 자료가 있는 원격 시스템과 연결을 설정, 자료를 획득하고 Xlib을 이용하여 화면에 출력하도록 하였다.

4.2 결과고찰

4.2.1 응답 시간(Response time) 측정

데이터 전송단계에서의 성능 파라미터로는 CCITT에서 데이터 처리량과 전달 지연시간을 정의하고 있는데 전달 지연시간을 측정하기 위해서는 서로 다른 시스템의 클럭을 동기화시켜야 하는 문제가 있다. 전달 지연시간 대신 응답시간 성능파라미터를 측정하게 되면 클럭 동기화 문제를 해결할 수 있고 한쪽에서 측정이 가능하기 때문에 유리하다. 특히 생산 환경하에서는 많은 경우에 네트워크에 연결되어 있는 다른 시스템의 정보를 요구해서 그에 대한 응답을 기다리는 작업을 요구하기 때문에 이 파라미터는 MAP 환경에서 적합한 성능 파라미터라고 할 수 있다. 본 실험환경에서는 응답시간을 측정하기 위하여 MMS 서비스 사용자가 mv fcopy(chan, source, source len, dest, dest len) 프리미티브를 요청해서 그에 대한 응답으로 u mv fcopy conf 프리미티브를 받을 때까지 걸리는 시간을 측정하였다. 메시지 크기를 500 바이트에서 128K 바이트까지 다양하게 변화시키면서 측정해 본 결과 메시지 크기가 응답시간에 영향을 줄 수 있는 요소가 됨을 확인하였다. 즉 표 1과 같이 2K바이트를 두번 요청하는 것보다 4K바이트를 한번에 가져오는 것이 상당히 짧은 응답시간을 나타내었다. LAN 환경하에서 데이터를 전송할때 최대 네트워크 패킷 크기보다 더 큰 데이터를 보내는 것이 효율적인 것을 고려하여 대용량의 데이터 전송을 위한 프로토콜을 제안하고 있으며, ¹¹ ISODE(ISO

Development Environment)의 ROSE와 SunRPC를 통해 응답시간을 측정해 본 결과 보내려는 메시지를 한번에 보내는 것이 더 짧은 응답시간을 나타내었다.¹¹⁾

표 1. 메시지 크기, slot time 대 응답 시간

메시지 크기	0.5K	1K	2K	4K	8K	16K	32K	64K
slot time(128)	1.37	2.08	2.86	4.56	7.96	15.38	29.83	58.66
slot time(512)	1.87	2.05	3.18	4.4	7.14	14.05	28.07	56.35

또한, 성능에 영향을 줄 수 있는 연결과 응답시간과의 관계는 측정에 어려움이 있어서 [그림 4]와 같은 MAP 환경하에서의 큐잉 모델을 이용하여 분석하였다.¹⁵

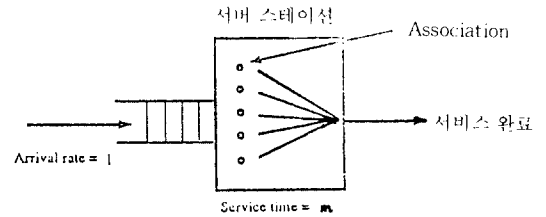


그림 3. 호스트 네트워크 어댑터의 큐잉 모델

[그림 3]은 호스트 네트워크 어댑터에 대한 큐잉 모델로, 다수의 서버를 가진 서버 스테이션으로 모델링될 수 있으며 각 서버는 하나의 연결로 간주될 수 있다. 또한 센서는 서비스를 요구하는 customer로서 모델링될 수 있다. 이때 모든 센서들이 호스트 컴퓨터와 연결하고자 할때 호스트 컴퓨터가 가질수 있는 최대 연결수에 제한을 받게 된다. 그러나 센서들은 보통의 경우 항상 쓰이지 않기 때문에 모든 연결이 유지될 필요가 없다. 센서가 연결을 요구할때 빈도가 심하지 않으면 즉시 서비스가 가능하지만 빈도가 심한 경우는 대기하는 시간이 길어지게 된다. 여기서는 부하가 심하지 않은 경우로 가정을 한다. 센서가 서비스를 받기위해 요구되는 시간은 연결을 확립하기 위한 시간, 호스트 쪽으로의 데이터 전송, 호스트 응용 프로그램에서의 데이터 처리, 호스트에서의 응답 시간 그리고 연결해제 시간으로 구성이 된다. 두개의 연속적인 센서 요구간의 시간간격을 지수(Exponential)

분포로 가정했으며 전체 서비스 시간을 Poisson 분포를 따르는 것으로 가정했다. 따라서 이러한 가정을 두게 되면 앞에서 언급된 센서를 이용한 공장네트워크 구성은 M/M/m queue로서 모델링할 수 있다.¹⁵⁾

연결의 수와 응답시간의 관계가 [그림 4]와 [그림 5]에 나타나 있다. 이 두개의 그림에서 알 수 있듯이 사용가능한 연결의 수가 많을수록 작은 응답시간을 나타내었고, 센서 사용사이의 평균시간이 작은 서비스는 평균시간이 큰것보다 부하가 심하다는 것을 뜻하므로 부하가 심하지 않은 것과 같은 응답시간을 얻기 위해서는 사용가능한 연결의 수가 더 많아야 한다는 것을 나타내고 있다. 아울러 센서 사용사이의 평균시간이 30000일때 사용가능한 연결의 수가 4개 이하이거나 60000일때 1개 이하인 경우에는 허용 응답시간 범위를 넘어서고 있음을 보여준다.

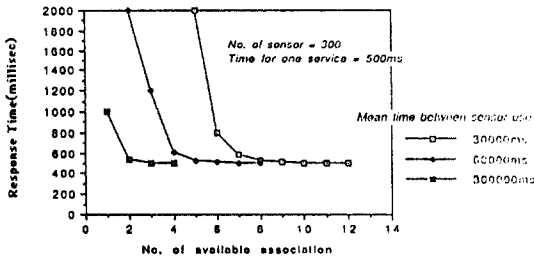


그림 4. 연결과 응답시간과의 관계(Time for one service = 500ms)

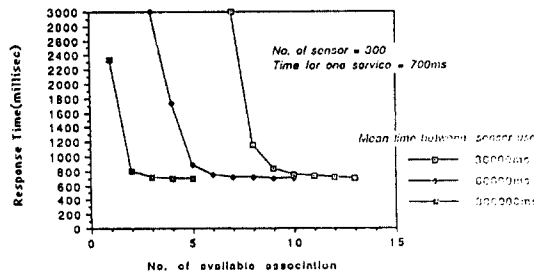


그림 5. 연결과 응답시간과의 관계(Time for one service = 700ms)

4.2.2 데이터 처리량 측정

데이터 처리량은 초당 전송되는 데이터 단위의 수이므로 전송된 사용자 데이터 바이트수쪽 전송동안 경과된 시간을 나눔으로서 측정하였다. 데이터 처리

량은 화일과 같은 큰 데이터 객체를 전달할때 중요하며, 처리량을 다음과 같은 모델로서 분석하였다.

$$\tau = \text{Message Overhead Factor} + \text{Packets Factor} + \text{Bytes Factor} + \text{Association Factor} \quad (4-1)$$

여기서 τ 는 하나의 메시지를 전송하는데 걸리는 시간이며, 데이터 처리량은 메시지 크기를 τ 로 나눈 값과 같다.

Message overhead 요소는 메시지를 전송하는데 걸리는 모든 처리시간을 포함하는데 호스트 어플리케이션의 처리시간, 오퍼레이팅 시스템 오버헤드, MAP 계층의 처리시간을 포함한다. Packets 요소는 데이터 메시지를 전송하기 위해 필요한 패킷의 수를 의미하며 Bytes 요소는 데이터 메시지의 길이를 나타낸다. 아울러 Assoc 요소는 어댑터에서 유지하고 있는 연결의 수를 말한다. 위와 같은 모델로부터 연결의 수가 증가할수록 MAP 네트워크어댑터의 데이터 처리량임 감소하며 메시지를 전송하기 위한 패킷의 수가 데이터 처리량의 큰 요인으로서 나타내 보이고 있다. 본 실험 환경에서 데이터 처리량을 측정하기 위하여 MMS 서비스 응답자가 개시자로부터 fcopy 요구를 받았을 때 그에 대한 응답으로 mv_fread_resp(req_info) 프리미티브를 이용하는데 이 mv_fread_resp 프리미티브를 가지고 데이터 처리량을 측정하였다. [표 2]는 slot time과 메시지 크기를 변화시키면서 측정한 데이터 처리량을 보여주고 있다. 결과로부터 같은 크기의 메시지를 한번에 네트워크에 보내는 것이 나누어서 보내는 것보다 응답시간이 감소함을 알 수 있다.

표 2. 메시지 크기, slot time 대 데이터 처리량

메세지 크기	1K	2K	4K	8K	16K	32K	64K	128K
slot time(128)	0.65	2.27	1.28	1.26	1.18	1.18	1.17	1.11
slot time(256)	1.20	1.14	1.19	1.13	1.11	1.09	1.04	1.01
slot time(512)	1.14	1.10	1.12	1.14	1.06	1.05	1.04	1.01
slot time(1024)	1.14	1.05	1.14	1.14	1.07	1.05	0.98	0.96

[그림 4]의 M/M/m queueing 모델에서 데이터 처리량을 계산할 수 있다. C(t)를 시간 t에 끝나게 되는

job의 수라고 할 때 $C(t)/t$ 는 t 가 ∞ 로 접근할 때 한계에 도달하게 된다. 여기서 $C(t)/t$ 가 데이터 처리량, $E[T]$ 이 된다. 어떤 서버의 데이터 처리량은 서버의 서비스 rate, μ 와 서버의 이용률, ρ 에 의존하게 된다. 따라서,

$$E[T] = \mu \rho \tag{4.2}$$

가 되고

$$\rho = \lambda / (m\mu) \tag{4.3}$$

(4.3)식이 성립하므로,

$$E[T] = \mu(\lambda / (m\mu)) = \lambda / m \tag{4.4}$$

따라서 시스템 전체의 데이터 처리량은 customer의 도착률 λ 에 비례하게 된다. 언급된 서버의 이용률, ρ 는 정상상태에서 1보다 적으므로 $\lambda < m\mu$ 가 성립하게 된다. 이는 m 이 증가할수록 λ 도 증가하게 되므로 결국 시스템 전체의 데이터 처리량은 연결의 수에 비례하게 된다.

4.2.3 연결 설정/해제(Association initiation/conclude) 측정

연결 설정은 쌍방의 응용계층간에 확립되는 것으로 연결확립을 위해 걸리는 시간은 [그림 5]와 같이 어댑터가 사용하는 시간과 호스트의 응용 프로그램이 메시지를 보내거나 받을때 사용하는 시간($\alpha + \beta$)으로 나누어 볼 수 있다. [표 3]은 slot time에 따른 연결 설정/해제 지연시간을 측정할 결과로서 slot time이 512일때 지연시간이 최소가 됨을 보여준다.

표 3. slot time 대 연결 설정/해제 지연시간

slot time	128	256	512	1024
연결설정시간	0.36	0.42	0.34	0.40
연결해제시간	0.32	0.24	0.24	0.32

실험환경에서 연결설정 지연시간을 측정하기 위하여 MMS의 서비스 프리미티브를 이용하였는데 MMS 서비스 사용자가 mv_init(chan, partner) 프리미티브를 요청해서 그에 대한 응답으로 u_mv_init conf

(req ptr) 프리미티브를 받을때까지 걸리는 시간을 측정하였다. 아울러 연결해제 지연시간을 측정하기 위하여 MMS 서비스 사용자가 mp_conclude(chan) 프리미티브를 요청해서 그에 대한 응답으로 u_mv_conclude conf(req ptr)을 받을때까지 걸리는 시간을 측정하였다. 이러한 연결설정과 해제 지연시간은 리포트 데이터 베이스를 요구할 때와 같이 빠른 응답 시간을 필요로할 때 중요하다.

V. 결 론

본 연구는 Flexible Automation을 위해 네트워크 운영과 관리체제에 대한 분석을 통하여 구성관리와 장애관리 그리고 성능관리를 위한 관리 프로세스를 구현하였다. 또한 생산 환경하에서 MMS를 이용하여 관리 서비스를 제공할 수 있는 방법과 네트워크 관리 구조를 제안하였으며 네트워크 관리용 성능을 관리하기 위해 성능을 측정할 수 있는 성능 파라미터를 선정하고 측정하는 방법을 제시하였다. 성능관리를 위해서는 네트워크의 현재상태를 파악하기 위해 관리자는 피관리자의 성능 파라미터를 추출해야 하는데 이를 위하여 관리자측에서 성능 파라미터를 요청해서 결과를 알아보는 round-trip 방법을 이용하였으며, 사용된 성능 파라미터로는 연결 설정/해제 지연 시간, 응답 시간 그리고 데이터 처리량 등이다. 성능 파라미터를 측정하기 위해 시스템에 독립적인 참조 포인트로서 응용계층에 위치하는 MMS를 선택하였다.

성능에 영향을 미치는 요소들로서 조사된 메시지 크기, 연결의 수, 호스트의 속도, 슬롯 타임들중에서 메시지 크기, 연결의 수 그리고 슬롯타임 등의 값을 변화시키면서 성능 파라미터들을 측정하였다. 메시지 크기를 500바이트에서 64K 바이트까지 변경하면서 응답시간을 측정해 본 결과 같은 크기의 메시지를 한꺼번에 네트워크에 보내는 것이 나누어서 보내는 것보다 응답시간의 감소함을 나타내었다. 또한 사용 가능한 연결의 수의 영향을 분석하기 위해 수십개의 센서들이 호스트에 연결되어 있는 MAP 네트워크를 부하가 심하지 않는 경우를 가정하여 M/M/m 큐잉 모델을 이용하였다. 데이터 처리량은 사용되고 있는 연결의 수가 많을수록 서버의 데이터 처리량은 증가하였다. 사용자가 요구하는 서비스의 질을 제공하기 위하여 앞에서 언급한 성능에 영향을 줄 수 있는 요소들은 관리 기능을 위한 자료로 충분히 이용될 수 있다. 성능관리는 네트워크의 효율을 높이기 위해 프

로세스 제어응용과 같이 실시간 제약을 받는 생산환경에서는 필수적이다.

앞으로의 연구 과제는 각 프로토콜의 계층에서 실제 성능에 영향을 주는 성능 파라미터를 정확하게 결정하는 것과, 연구된 성능평가를 기초로 하여 네트워크를 평가하고 조절할 수 있는 실질적인 방법등이다.

참 고 문 헌

1. Lillian N. Cassel, Craig Partridge, "Network Management Architectures And Protocols : Problems And Approaches," IEEE Selected Areas in Communications, VOL.7, NO.7, Sep. 1989.
2. J. R. Pimentel, "Fieldbus Network Management : Requirements and Architectures," Integrated Network Management I, 1989.
3. Richard E. Caruso, "Network Management : A Tutorial Overview," IEEE Communication Magazine, March 1990.
4. ISO 7498/4 : Information Processing Systems-Open Systems Interconnection-Basic Reference Model Part 4-OSI Management Framework.
5. X.SMI/ISO DP 10065-4 : Information Processing Systems-Open Systems Interconnection-Management Information Services-Structure of Management Information part 4 : Guidelines for the Definition of Managed Objects [for CCITT Applications], 1989. (N-4065)
6. X.SMI/ISO DIS 10165-1 : Information Processing Systems-Open Systems Interconnection-Management Information Services-Structure of Management Information part 1 : Management Information Model[for CCITT Applications], 1990.
7. Jock Embry, Peter Manson, Dave Milham, "An Open Network Management Architecture : OSI/NM Forum Architecture and Concepts," IEEE Network Magazine, July 1990.
8. Celia A. Joseph, Kurudi H. Muralidhar, "Integrated Network Management in an Enterprise Environment," IEEE Network Magazine, July 1990.
9. Ina Sugarbrood, "An OSI-Based Interoperability Architecture for Managing Hybrid Networks," IEEE Communication Magazine, March 1990.
10. Morris S. Sloman, J. D. Moffette, "Domain Management for Distributed System," Integrated Network Management I, 1989.
11. Willy Zwaenepoel, "Protocols for Large Data Transfers Over Local Networks," ACM SIGCOMM '85, Apr 1985, pp22-32.
12. K. H. Muralidhar, "Performance Management Measures, Analysis, Control and Optimizaiton," Proceedings of 11th Conference on Local Computing Networks, Oct 1986.
13. K. Mills, M.Wheatley, S. Heatley, "Prediction of Transport Protocol Performance Through Simulation," ACM SIGCOMM '86, Aug 1986.
14. Perter Gunningberg, "Application Protocols and Performance Benchmarks," IEEE Communications Magazine, June 1989.
15. Kishor Shridharbhai Trivedi, "Probability and Statistics with Reliability, Queuing, and Computer Science Applications," Prentice-Hall, 1982.
16. Wojciech Cellary, "Performance Parameters of Computer Network Protocols Referred to Their Formal Model," Protocol Specification, Testing and Verification IV, 1985.
17. Subrata Mazumdar, Aurel A. Lazar, "Monitoring Integrated Networks for Performance Management," IEEE ICC '90, 1990.
18. Lawrence L. Garlick, Raphael Rom, "Reliable Host-to-Host Protocols : Problems and Techniques," Principles of Comm. and Networking Protocols.
19. Lillian N. Cassel, Craing Partridge, "Network Management Architectures And Protocols : Problems And Approaches," IEEE Selected Areas in Communications, VOL.7, NO.7, 1989.
20. Fred Halsall, Nasser Modiri, "An Implementation of an OSI Network Management System," IEEE Network Magazine, July 1990.

—“본 논문은 한국과학재단 목적기초연구과제 지원에 의하여 이루어 졌음”—

姜 文 植(Moon-Sik Kang) 正會員
第18卷 第3號 參照
江陵大學校 電子工學科 教授

李 재 용(Jae-Yong Lee) 正會員
浦港工科大学 電子計算學科 教授

李 相 培(Sang-Bae Lee) 正會員
第18卷 第3號 參照
延世大學校 電子工學科 教授