

< 논 문 >

공장자동화용 토큰버스 네트워크의 퍼지 성능관리기 개발

이상호* · 손준우* · 이 석**

(1995년 5월 26일 접수)

Development of Fuzzy Network Performance Manager for Token Bus Networks in Automated Factories

Sang-Ho Lee, Joon-Woo Son and Suk Lee

Key Words : Computer Integrated Manufacturing(컴퓨터 통합생산), Production Management(생산관리), Local Area Network(근거리 통신망), Communication Equipment(통신기구), Manufacturing Automation Protocol(공장자동화용 프로토콜), Priority Mechanism(우선순위 도구), Fuzzy Network Performance Manager(퍼지 네트워크 성능관리기), Data Latency(전송지연), Token Circulation Time(토큰순환시간)

Abstract

This paper presents the development of three types of network performance manager for IEEE 802.4 token bus networks that are a part of Manufacturing Automation Protocol(MAP). The performance managers attempt to keep the average data latency below a certain level specified for each priority class. All of the three performance managers are based on a set of fuzzy rules incorporating the knowledge on the relationship between data latency and parameters of the priority mechanism. These Fuzzy Network Performance Managers(FNPMs) have been evaluated via discrete event simulation to demonstrate their efficacy.

1. 서 론

생산관리(production management) 및 자동화를 위한 컴퓨터의 응용은 50년대 초에 실용화된 수치 제어 공작기계로부터 시작하여 현재 산업용 로봇과 CAD 시스템 등에 이르기까지 매우 필수적인 요소가 되었다. 또한, 하드웨어 가격의 지속적인 하락으로 거의 모든 분야에서 컴퓨터를 사용하고 있으며, 컴퓨터의 숫적 증가와 더불어 자동화 시스템의 보급도 크게 증가되었다. 그러나, 종래와 같은 컴퓨터 시스템의 사용은 여러 기구들을 각기 독립적

으로 운용함으로써, 자동화의 섬(islands of automation)들을 형성하기에 이르렀으며, 생산성의 향상에 있어서도 기대만큼 성취되지 못하였다.

그러나, 이러한 자동화의 섬들을 연결하기 위하여 근거리 통신망(Local Area Network, LAN)⁽¹⁾이 생산자동화 분야에 응용되면서, 제한된 공간 내에 분산되어 있는 여러 컴퓨터와 생산장비들을 공통된 전송매체로 연결하여 상호 간의 정보교환을 가능하게 하였다. 이와 같은 근거리 통신망에 접속된 두 기구는 상위 기구의 도움없이 직접적인 통신을 수행할 수 있으므로, 이를 바탕으로 컴퓨터 통합생산(Computer Integrated Manufacturing, CIM)의 개념을 구현할 수 있게 되었다. 이러한 컴퓨터 통합생산은 한 기업체 안에서 수행되는 다양한 기

*부산대학교 대학원 생산기계공학과, 기계기술연구소

**회원, 부산대학교 생산기계공학과

업활동들을 하나의 폐쇄루프를 이루는 대화형 제어 시스템으로 통합하려는 움직임^(2~4)으로서, 가공과 조립, 검사 및 물류제어를 중심으로 하는 FA, 설계와 해석 및 공정설계로 대변되는 Product/Process design, 그리고 일정계획 및 공정관리 등의 정보관리와 관계되는 Manufacturing Planning & Control이라는 세 가지의 분야가 연동되는 기술이다.⁽⁵⁾ 이를 위해서 매출보고서, 생산주문, 신제품설계도, 공정총괄, 수치제어 동작기계의 프로그램, 여러 생산장비의 상태 등과 같은 기업 내에 발생하는 다양한 정보를 부서간에 공유하기 위해서는 컴퓨터 네트워킹은 없어서는 안될 중추적인 역할을 맡고 있다.

컴퓨터 통합생산을 위한 네트워크는 전송매체와 토폴로지(topology, 네트워크에 접속된 기구들이 연결된 형태), 통신규약(communication protocols) 등을 선정하여 여러 가지 요구조건들을 만족시켜야 하며, 이와 같은 기본 요소들이 선정되면 통신규약을 위한 변수에 적절한 값을 부여하여 시스템을 설계하여야 한다. 하지만, 네트워크의 통신 부하와 통신 자원은 시간에 따라 항상 변화하는 특성을 가지므로 네트워크의 성능도 이들의 상태에 따라 큰 영향을 받는다. 더욱이, 최근의 대규모 통합시스템 추세는 접속기구의 급격한 증가와 다양화로 인해 더욱 효율적인 네트워크의 운영과 유연한 재구성을 요구하게 되었고, 이러한 효율적인 네트워크 운영을 위한 자동화된 성능관리기 중요시되고 있다.^(3,4,6,7)

따라서, 본 연구의 목표는 공장자동화용 표준 통신 프로토콜인 MAP(Manufacturing Automation Protocol)^(9,10)에 이용되는 IEEE 802.4 토큰버스 네트워크⁽¹¹⁾의 성능관리기를 개발하는 것이며, 이는 종래에 수동적으로 조절되어 왔던 토큰버스 통신규약의 우선순위 도구(priority mechanism)의 변수를 상황에 따라 자동적으로 조절하여 성능을 개선시키는데 있다. 이러한 우선순위 도구는 버스의 액세스(access)에 대한 우선권을 부여하기 위한 것으로 문서파일, CAD파일, 센서데이터(sensor data) 등과 같은 다양한 종류의 메시지를 서로 다른 요구조건에 맞추어 전송하려는 것이다.

이와 같이, 우선권에 관한 권리를 조절하는 우선순위 도구의 네가지 변수는 THT(Token Holding Time)와 TRTi(Token Rotation Time, $i=4, 2, 0$)로 구성되며, 이러한 변수들의 값이 전송 지연

(data latency, 큐에서의 메세지 대기시간과 메세지의 전송시간의 합)에 근거해서 성능과 밀접한 관계가 있고, 이 관계가 네트워크의 성능관리를 위한 핵심적인 요소가 된다. 즉, 네트워크 성능과 밀접한 관계가 있는 메세지들의 전송지연은 변수들의 값에 따라 변하므로, 우선순위 도구의 네가지 변수를 상황에 따라 다르게 선점함으로써 성능을 향상시킬 수 있다.

최근에 MAP 네트워크의 성능관리에 관한 구체적인 실현기법으로서 PA(Perturbation Analysis), SO(Stochastic Optimization), LA(Learning Automata) 등의 이론을 복합적으로 이용하여 그 유효성을 검증한 경우가 있으나, 그 구조가 상당히 복잡하며 프로토콜의 일부를 개정해야 할 필요성도 가지고 있었다.^(3,4,6,7) 이러한 점을 고려하여 본 연구에서는 인간의 관리기능을 모방한 퍼지기법^(12~15)을 네트워크의 성능관리에 도입함으로써, 네트워크의 성능과 통신규약 변수의 관계에 대한 지식, 새로운 변수값의 결정에 관한 지식을 퍼지 성능관리가 갖추도록 하였다.

본 연구에서는 IEEE 802.4 토큰버스 프로토콜로 운영되는 네트워크를 대상으로 3가지 형태의 퍼지 네트워크 성능관리기(Fuzzy Network Performance Manager, FNPM)를 개발하였다.^(16~18) 이러한 퍼지 성능관리기는 네트워크의 통신부하에 따라 사용자가 요구하는 전송지연을 만족시키기 위하여 우선순위 변수를 동적으로 조절하는 기능을 가지며, 시뮬레이션을 통해 그 효용성을 입증할 수 있었다.

본 논문은 모두 다섯 절로 이루어져 있으며, 2절에서는 IEEE 802.4 토큰버스 프로토콜의 우선순위 도구와 그 특성을 설명한다. 그리고 3절에서는 개발된 3가지 형태의 퍼지 네트워크 성능관리기의 구성과 특징을 보이며, 4절에서는 이들을 네트워크 시뮬레이션 모델과 결합하여 각각의 성능평가를 수행함으로써, 퍼지 네트워크 성능관리기가 네트워크의 성능관리에 효과적인 수단임을 보인다. 끝으로 5절에서는 4절에서 얻어진 성능평가를 토대로 얻어진 결론을 요약하고 향후 연구 방향을 제시한다.

2. 토큰버스 프로토콜의 특성

토큰버스방식은 통제된 매체 접속제어방식(controlled medium access control method)의 한 종류로서, Fig. 1과 같이 버스(bus)나 트리(tree)

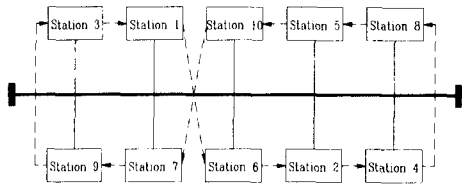


Fig. 1 Logical ring of a token bus network

구조상에 부착된 스테이션들이 논리적 링(logical ring)을 형성하여 상호 통신하는 방식이다. 즉, 각 스테이션들은 논리적 링(logical ring)을 따라 자신의 전/후(predecessor/successor) 스테이션이 어떤 것인가를 인지하여 순차적으로 전송 기회를 획득한다.

토큰이라고 불리는 제어패킷은 특별한 형식의 비트패턴(bit pattern)으로서 액세스 권리를 조정하는데, 어떤 스테이션이 토큰을 받으면 규정된 시간 동안 매체의 제어권이 주어진다. 이 경우 스테이션은 한개 이상의 패킷을 전송할 수 있으며, 다른 스테이션을 폴(poll)하거나 응답을 수신할 수도 있다. 또한, 스테이션이 동작을 완료하거나 주어진 시간이 지나면 논리적 링을 따라 다음 스테이션으로 토큰을 전송함으로써, 토큰을 전달받은 새로운 스테이션이 전송할 수 있는 권리를 받게 된다. 이와 같이 각 스테이션은 순차적으로 토큰을 수신하기 때문에 이 방식은 공정한 액세스를 제공해 준다.

2.1 토큰버스 프로토콜의 우선순위 도구

IEEE 802.4 토큰버스 프로토콜은 메시지 전송을 위한 네 가지의 다른 우선순위를 가지며, 이러한 우선순위 도구는 Fig. 2와 같이 네개의 카운트-다운 타이머, 즉 Token Holding Timer(THT), Token Rotation Timer 4, 2, 0(TRT4, TRT2, TRT0)를 갖는다. 이러한 우선순위 도구의 목적은 한정된 네트워크 용량을 상위 우선순위의 프레임에 할당하고, 하위 우선순위의 프레임은 여분의 용량이 있을 때만 전송하도록 하는 데 있다.

만약, 어떤 스테이션이 토큰을 잡으면 그 스테이션은 Fig. 3과 같은 순서에 따라 각 우선순위들의 데이터를 전송할 수 있다. 즉, 우선순위 6의 메시지는 THT의 시간동안 전송될 수 있는 반면, 우선순위 4, 2, 0과 같은 하위 순위의 메시지는 순차적으로 해당 TRT에서 토큰이 논리적 링을 순환하는

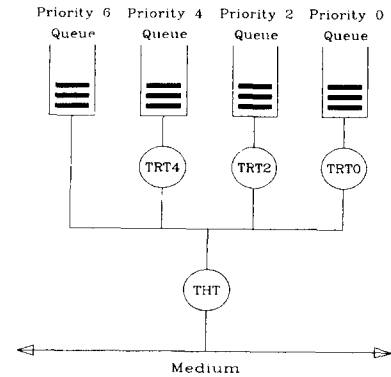


Fig. 2 Schematic structure of a station

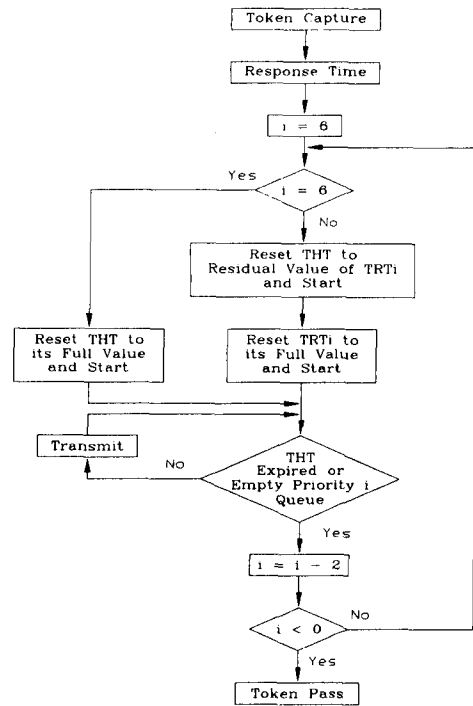


Fig. 3 Flow chart of the IEEE 802.4 priority mechanism

데 걸린 시간인 토큰 순환시간(Token Circulation Time, TCT)을 삭감한 시간동안 전송될 수 있다. 이와 같은 TRT들의 우선순위를 만족시키기 위해 IEEE 802.4 표준에서는 TRT4 ≥ TRT2 ≥ TRT0를 요구하고 있다.

2.2 시뮬레이션 모델

IEEE 802.4 토큰버스 프로토콜의 시뮬레이션을

위해서 SIMAN⁽¹⁹⁾을 이용하여 이산사상 시뮬레이션 모델(discrete event simulation model)^(19,20)을 구성하였으며, 시뮬레이션 모델은 메시지발생 서브모델(Message Generation Submodel, MGS)과 프로토콜 서브모델(Protocol Submodel, PS)로 나뉘어진다.

메세지발생 서브모델은 사용자가 정의한 확률분포에 따라 네트워크 트래픽을 발생시켜서 각 우선순위의 큐(queue)에 메세지를 채우고, 프로토콜 서브모델은 스테이션의 상태에 따라 프로토콜 로직을 실행시켜 전송 큐로부터 이들 메세지를 전송하여 네트워크 시스템의 외부로 제거하는 역할을 수행한다.

이 시뮬레이션은 각 우선순위 레벨을 위한 메세지 발생주기와 메세지 정보길이에 관한 지수적, 상수적 분포를 포함해서 사용자가 정의한 임의의 확률분포를 수용하며, 전송된 메세지 수와 전송지연 등의 데이터를 통계적으로 나타낸다.

2.3 토큰버스 프로토콜의 특성

본 연구에서는 토큰버스 프로토콜의 우선순위에 관한 특성을 파악하기 위하여 여러가지 조건에 대해 시뮬레이션을 수행하였으며, 그 결과 각 우선순위의 타이머에 따른 전송지연과의 관계에 대한 다음과 같은 몇 가지 규칙을 발견하였다.

(1) 일반적으로 어떤 우선순위의 타이머가 길면 길수록, 그 우선순위의 전송지연은 짧아지는 편이다.

(2) 우선순위 6의 전송지연을 감소시키기 위해서는 THT를 증가시키는 것보다 TRT들을 감소시키는 것이 더욱 효과적이다.

(3) 모든 TRT들이 토큰순환시간인 TCT와 비교하여 너무 길면, 큐에서의 대기시간이 우선순위와 무관하게 서로 비슷해 지는 경향이 있다.

(4) 어떤 TRT가 TCT와 비교하여 충분히 길지 않으면, 그 우선순위의 전송이 제대로 이루어지지 않아, 전송지연이 증가하게 된다.

(5) 토큰이 논리적 링을 순환하는데 걸리는 시간(TCT)보다 어떠한 TRT가 짧으면, 그 우선순위에 속한 메세지의 전송은 완전히 중단되며, 이로 인해 다른 우선순위의 전송지연이 감소된다.

(6) 어떤 우선순위의 전송지연이 감소되면 그 영향으로 다른 우선순위의 전송지연이 증가할 수도 있다.

3. 퍼지 네트워크 성능관리기의 개발

시뮬레이션을 통하여 얻은 토큰버스 프로토콜의 특성을 바탕으로 하여 Fig. 4에 나타난 것과 같은 퍼지 네트워크 성능관리기(Fuzzy Network Performance Manager, FNPM)를 구축하였으며, 이를 바탕으로 Basic FNPM,⁽¹⁶⁾ TCT-based FNPM,⁽¹⁷⁾ 그리고 Switch-type FNPM⁽¹⁸⁾을 개발하였다.

3.1 Basic FNPM

Basic FNPM은 전반적으로 통신 부하에 대한 사용자의 요구 전송지연이 쉽게 만족될 수 있는 경우이면서 우선순위 6의 전송지연에 대한 사용자의 정책(policy)이 너그러운 때 적합한 형태로 모든 우선순위의 전송지연을 복합적으로 고려하는 원칙을 바탕으로 하고 있다.

이를 위한 Basic FNPM의 퍼지 입력변수로는 네가지 우선순위의 평균 전송지연(D_i, i=6, 4, 2, 0)을 택하였으며, Fig. 5에서 사용자에게 의해 임의로 선택될 수 있는 퍼지 입력변수의 연속적인 언어변수와 상응하는 소속함수를 보여주고 있다. 즉, 각 우선순위의 평균 전송지연마다 Small, Medium,

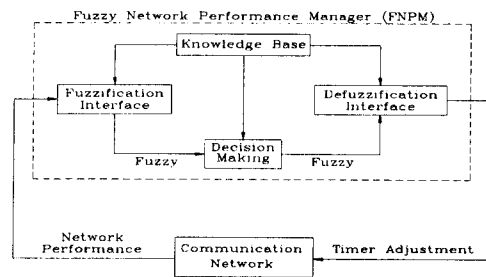


Fig. 4 Configuration of a fuzzy network performance manager

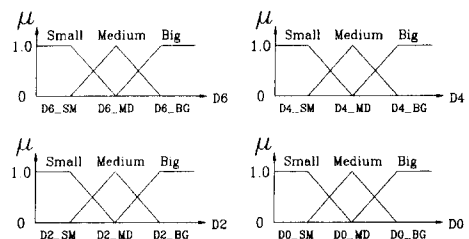


Fig. 5 Membership functions for fuzzy inputs of the basic FNPM

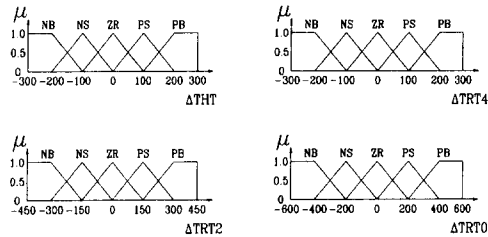


Fig. 6 Membership functions for fuzzy outputs of the basic FNPM

Big의 세가지 언어변수를 설정하였고, Small의 소속함수가 1의 값을 갖기 시작하는 평균 전송지연의 값(Di_SM, i=6, 4, 2, 0)은 사용자가 허용할 수 있는 요구 전송지연을 나타낸다고 할 수 있다.

이러한 퍼지 입력변수는 Basic FNPM의 지식 베이스부에 있는 81개의 제어 룰에 적용되는데, 이러한 Basic FNPM에서의 제어방침은 IEEE 802.4 토큰버스 프로토콜의 우선순위 규정하에서 모든 우선순위의 전송지연을 동시에 고려하여, 복합적으로 변화시키기 위한 것이다.

퍼지법칙의 결론부에서는 네가지 타이머의 길이를 변화시키기 위한 언어변수를 사용하고 있으며, 각 타이머에 대하여 다섯 종류, 즉 NB(Negative Big), NS(Negative Small), ZR(ZeRo), PS(Positive Small), PB(Positive Big)를 사용하였고, 그 소속함수들이 Fig. 6에 나타나 있다. 여기서 결정되는 퍼지한 타이머의 변화는 비퍼지화를 통하여 crisp한 타이머 변화량으로 변환되고 이것이 토큰버스 네트워크의 우선순위 기구를 조절한다.

3.2 TCT-Based FNPM

Basic FNPM에서는 81개의 많은 제어 룰을 필요로 하며, 통신부하에 대한 사용자의 요구 전송지연을 만족시키기 어려운 경우에 있어서는 모든 우선순위의 전송지연을 복합적으로 고려하기 때문에 우선순위간의 차이가 모호해지는 경향이 있었다. 또한 전송지연에 관한 관찰에 의해 Basic FNPM의 룰들은 타이머들을 서로 상대적으로만 조정하는 역할을 하는데, TRT들이 모두 지나치게 긴 경우는 우선순위가 별다른 역할을 못할 수 있고, 또 TRT들이 지나치게 짧은 경우에는 전송이 전혀 허용되지 않는 결과를 얻을 수가 있다. 이러한 결점을 보완하기 위하여 모든 TRT들을 같이 증가시키

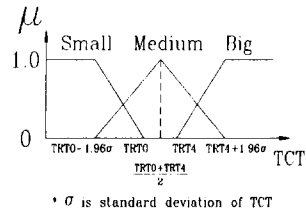


Fig. 7 Membership functions for the TCT of the TCT-based FNPM

거나 감소시키는 룰이 필요하였다.

따라서, TCT-Based FNPM에서는 하위 우선순위들이 다소 희생을 하더라도 최상위 우선순위인 우선순위 6의 지연을 가급적 빠른 시간내에 사용자가 선정한 값 이내로 감소시키도록 하였다. 우선순위 6의 지연이 수용범위안에 들면, 다음 우선순위인 우선순위 4가 하위 우선순위인 우선순위 2와 0에 지나친 지연이 발생하지 않도록 고려하면서, 빠른시간 내에 허용지연의 범위에 근접하도록 하였다. 이후 우선순위 2에 대하여 같은 방법으로 전송지연을 감소시키며, 우선순위 0은 상위 우선순위가 만족할 만한 전송지연을 갖는 상태에서 자신의 지연을 최소로 줄이는 방법을 취하였다.

이를 위한 TCT-Based FNPM에서의 퍼지 입력 변수로서는 Basic FNPM에서의 Fig. 5와 마찬가지로 네트워크의 성능에 직접적인 영향을 미치는 네가지 우선순위의 평균 전송지연 시간과 Fig. 7에 보이는 바와 같이 평균 토큰 순환시간(Token Circulation Time, TCT)을 택하였다. 토큰 순환시간은 토큰이 논리적 링을 완전히 한 바퀴 순환하는데 걸리는 시간으로 여기에는 토큰 송신시간, 각 스테이션에서의 처리시간, 그리고 메세지 패킷의 송신시간이 포함되어 있다. 이러한 TCT는 네트워크 부하와 밀접한 관계에 있으므로 TCT와 비교하여 TRT들을 모두 증가시키거나 감소시키기 위하여 FNPM의 추가적인 입력으로 선택되었다.

TCT-Based FNPM은 네트워크의 성능관리를 위해 TCT를 포함한 5개의 입력변수와 4개의 출력 변수를 사용하였고, Table 1과 같이 12개의 룰로서 제어방침을 나타내었다. 여기서, 첫번째 룰은 TCT가 매우 짧고 TRT들의 타이머 값이 지나치게 큰 경우로서 우선순위 간의 차이가 모호해지기 때문에 모든 TRT들을 감소시키기 위한 것이다. 그리고 두번째 룰은 3가지의 하위 우선순위의 전송

Table 1 Fuzzy rules of TCT-based FNPM

| | TCT | D6 | D4 | D2 | D0 | ΔTHT | ΔTRT4 | ΔTRT2 | ΔTRT0 |
|----|-----|----|----|----|----|------|-------|-------|-------|
| 1 | S | | | | | ZR | NB | NB | NB |
| 2 | L | S | | | | ZR | PB | PB | PB |
| 3 | L | M | | | | PB | ZR | ZR | ZR |
| 4 | M | B | | | | PB | NB | NB | NB |
| 5 | M | M | | | | PS | NS | NS | NS |
| 6 | M | S | B | | | ZR | PB | NS | NS |
| 7 | M | S | M | | | ZR | PS | ZR | ZR |
| 8 | M | S | S | B | | ZR | ZR | PB | NS |
| 9 | M | S | S | M | | ZR | ZR | PS | ZR |
| 10 | M | S | S | S | B | ZR | ZR | ZR | PB |
| 11 | M | S | S | S | M | ZR | ZR | ZR | PS |
| 12 | M | S | S | S | S | ZR | ZR | ZR | ZR |

기회를 증가시키기 위한 것으로, 우선순위 6의 평균 전송지연이 Small이고, TCT가 TRT들과 비교하여 너무 긴 경우로써 하위 우선순위의 전송기회가 그다지 많지 않을때 TRT들을 증가시키기 위한 것이다. 그리고, 세번째 룰은 두번째 룰과 비슷한 상황에서 적용되지만, 우선순위 6의 전송지연이 Medium에 해당되는 경우이다. 나머지 룰들은 TCT가 Medium인 경우로서 우선순위 순으로 사용자의 요구전송 지연시간을 만족시키기 위하여 타이머들이 변화하도록 하였다. TCT-Based FNPM에서도 각 우선순위의 타이머 변화량을 퍼지 출력 변수로 선택하였으며, 타이머들의 변화량에 관한 비퍼지화 영역은 Fig. 6과 같이 Basic FNPM과 동일한 5단계의 연속적인 언어변수 영역을 갖도록 하였다.

3.3 Switch-type FNPM

TCT-Based FNPM은 토큰 순환시간(Token Circulation Time, TCT)을 도입하여 Basic FNPM의 81개에 해당하는 룰을 12개로 대폭 감소시켰으며, 우선순위 도구의 관리에 있어서 상위 우선순위에 보다 비중을 두었다. 그러나 통신 부하에 비하여 우선순위 6에 대한 사용자의 요구 전송지연이 만족되기 어려운 값으로 설정된 경우에 있어서는 TCT-Based FNPM이 우선순위 6의 전송지연

을 감소시키기는 하나, 사용자의 요구 전송지연을 만족시키지는 못한다. 따라서 하위 우선순위의 전송지연에 상관없이 우선순위 6의 전송지연만을 만족시키려는 정책에 있어서는 다른 하위 우선순위의 보다 극단적인 희생을 필요로 한다.

이러한 점에 기초해서, 일정 시간동안 전송된 정보량인 하위 우선순위의 throughput을 고려하여, 퍼지 추론부를 두가지 모듈로 나누었다. 첫번째 모듈(FNPM1)은 최상위 우선순위인 우선순위 6의 평균 요구 전송지연만을 고려한 경우이고, 두번째 모듈(FNPM2)은 우선순위 6의 요구 전송지연이 만족되었거나 하위 우선순위의 throughput이 주어진 트래픽보다 작아졌을때 각 우선순위의 레벨 순으로 사용자의 요구 전송지연을 만족시키도록 하기 위한 것이다.

Fig. 8은 스위치형 퍼지 성능관리기(Switch-type

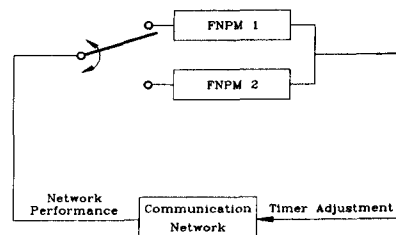


Fig. 8 Configuration of the switch-type FNPM

FNPM)를 보이며 네트워크의 상황에 따라 다른 알고리즘을 갖는다. 여기서는 우선순위 4, 2, 0의 트래픽에 대한 전송된 정보량의 합이라고 할 수 있는 RT(Relative Throughput)를 도입하여, 이 값이 어떤 주어진 한계(RT=2)보다 크고 적음에 따라 스위치가 작동되도록 하였다. FNPM1은 우선순위 6의 전송지연이 사용자의 요구 전송지연을 만족하지 못하면서 하위 우선순위의 전송이 거의 모두 허용(RT>2)될 때 사용되고, FNPM2는 우선순위 6의 전송지연이 사용자의 요구 전송지연을 만족했거나, 하위 우선순위의 상당부분이 전송되지 못한 경우(RT<2)에 수행된다. 이와같은 선택조건을 다음의 Table 2에서 보인다.

네트워크의 성능관리를 위한 입력으로는 기존의 입력에 추가로 RT(Relative Throughput)가 필요할데, 이를 위해 각 우선순위에 있어서 어느 일정 시간 동안 네트워크에 전송되기 위해 받아들여진 비트의 수(offered bit)와 네트워크에서 실제로 전송된 비트의 수(transmitted bit)가 필요하다. 여기서, RT는 우선순위 6를 제외한 하위 우선순위들의 트래픽에 대한 throughput의 합으로서 다음 식과 같다.

$$RT = \frac{T_4}{G_4} + \frac{T_2}{G_2} + \frac{T_0}{G_0}$$

$$= \frac{(\text{transmitted bit})_4}{(\text{offered bit})_4} + \frac{(\text{transmitted bit})_2}{(\text{offered bit})_2} + \frac{(\text{transmitted bit})_0}{(\text{offered bit})_0}$$

퍼지화 영역으로서 사용자가 임의로 신정한 각 우선순위들의 전송지연에 관한 언어변수 영역은 TCT-Based FNPM과 같으며, Small에 해당하는 영역은 사용자가 요구하는 전송지연을 나타낸다.

Table 3는 FNPM1의 제어룰을 나타내는데, 우선순위 6의 전송지연이 사용자의 요구 전송지연을 만족하지 못하면서 하위 우선순위의 throughput의 합인 RT가 2보다 큰 경우, 즉 하위 우선순위의 프레임이 거의 모두 전송되는 경우에 THT를 높이고, 하위 우선순위의 타이머인 TRT들을 대폭 감소시켜 최상위 우선순위의 전송지연을 격감시키는데 목적이 있다. 이같은 노력에 따라 우선순위 6의 전송지연이 사용자의 요구 전송지연을 만족할 경우, 이를 유지하면서 하위 우선순위의 전송지연을 만족시킬 수 있도록 FNPM2로 스위치를 바꾸게

Table 2 Switch operating condition

| D6 | RT | FNPM |
|---------------|----|------|
| Not satisfied | >2 | 1 |
| Not satisfied | ≤2 | 2 |
| Satisfied | ≤2 | 2 |
| Satisfied | >2 | 2 |

Table 3 Fuzzy rules of FNPM1 for the switch-type FNPM

| | D6 | △THT | △TRT4 | △TRT2 | △TRT0 |
|---|----|------|-------|-------|-------|
| 1 | B | PB | NB | NB | NB |
| 2 | M | PS | NS | NS | NS |

된다. FNPM2의 룰은 TCT-Based FNPM에서의 룰인 Table 1과 같다.

FNPM의 출력변수로는 기존의 FNPM과 마찬가지로 전송지연과 직접적인 관계가 있는 각 우선순위의 타이머 변화량을 선택하였으며 비퍼지화를 위한 언어적 영역도 기존의 FNPM과 같게 하였다.

세 종류의 FNPM을 구현하는데 있어서 입력변수들은 fuzzy singleton으로 처리하였고, fuzzy reasoning을 위하여 Mamdani의 max-min 연산을 사용하였다. 그리고, 추론의 결과를 비퍼지화하기 위하여 면적중심법(center of area method)를 사용하였다.⁽¹¹⁾

4. 퍼지 네트워크 성능관리기의 평가

토큰버스 네트워크의 성능관리(fuzzy network performance manager)를 위한 시뮬레이션은 우선순위 별로 각각 다른 메세지 생성간격과 메세지 정보길이에 관한 확률적인 분포를 사용하여 메세지를 발생시키고, 각 우선순위들의 할당된 타이머 값에 따라 전송을 수행한 후, 전송된 메세지의 양과 소요된 전송지연 등의 결과를 산출한다.

2.2절에서 언급된 것과 같이 IEEE 802.4 토큰버스 프로토콜의 성능관리를 위해 통신망 시뮬레이션 모델을 구축하였으며, Table 4에 시뮬레이션에 사용된 여러 조건들을 보인다. 이러한, 통신망 시뮬레이션 모델에서 발생하는 각 우선순위들의 전송지

Table 4 Simulation parameters

| | Priority 6 | Priority 4 | Priority 2 | Priority 0 |
|--|-------------------|------------|------------|------------|
| Number of stations | 10 | | | |
| Simulation time (μsec) | 3.0×10^7 | | | |
| Number of queues per station | 4 | | | |
| Queue capacity of each priority | 10 | 10 | 10 | 10 |
| Assigned traffic (%) | 10 | 20 | 20 | 30 |
| Average message generation interval (μsec) | 50000 | 50000 | 100000 | 200000 |
| Type of probability distribution for message length | Exponential | | | |
| Type of probability distribution for message generation interval | Exponential | | | |

Table 5 Initial conditions for simulation experiments

| | | Priority 6 | Priority 4 | Priority 2 | Priority 0 |
|--|-------|------------|------------|------------|------------|
| Initial timer (μsec) | | 150 | 3500 | 3500 | 3500 |
| Fuzzy input region (μsec) | Di_SM | 2300 | 6000 | 10000 | 20000 |
| | Di_MD | 2800 | 7000 | 11500 | 22000 |
| | Di_BG | 3300 | 8000 | 13000 | 24000 |

연은 3가지 형태의 FNPM에 입력되어 각각의 제어방침에 따른 퍼지룰을 수행하여 각 우선순위들의 타이머 변화량을 출력하며, 이것이 다시 토큰버스 통신망 시뮬레이션 모델의 입력이 된다.

4.1 Basic FNPM의 성능 평가

네트워크의 용량에 대한 사용 부하라고 할 수 있는 트래픽은 Table 4에서와 같이 우선순위 6에 대해서는 10%, 우선순위 4, 2, 0에 대해서는 각각 20, 20, 30%씩을 할당하여 전체적으로 80%의 트래픽에 51가지의 랜덤시드(random seeds)를 사용하여 시뮬레이션을 수행하였다. Table 5는 시뮬레이션의 초기조건에 관한 정보를 나타내고 있으며, Di_SM은 사용자의 요구 전송지연을 의미한다.

Basic FNPM에 의한 각 우선순위의 전송지연에 관한 변화추이를 Fig. 9에서 보이고 있으며, Fig. 10에서는 사용자의 전송지연을 만족시키기 위해 동적으로 변화하는 각 우선순위의 타이머를 보인다. 이 실험에서 설정된 사용자의 요구 전송지연은 대체로 만족되기 어려운 값으로 선정하여, 이 때의

Basic FNPM의 한계를 제시하였다.

초기의 타이머 값에 의한 각 우선순위의 전송지연은 모든 우선순위에 있어서 사용자의 요구 전송지연보다 훨씬 크며, 이에 따라 자신의 전송지연을 줄이기 위해 모든 우선순위의 타이머를 증가시킨다. 하지만, 우선순위 6의 전송지연을 줄이기 위해서는 THT를 늘이는 것보다는 TRT들을 감소시키

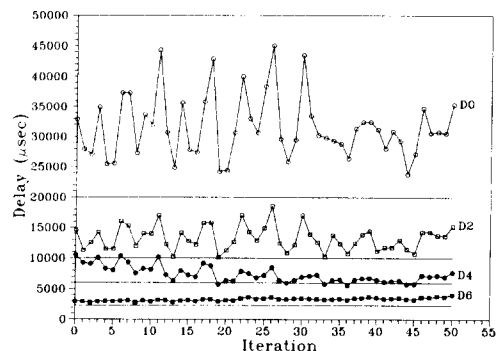


Fig. 9 Average data latency from the simulation experiment with the basic FNPM

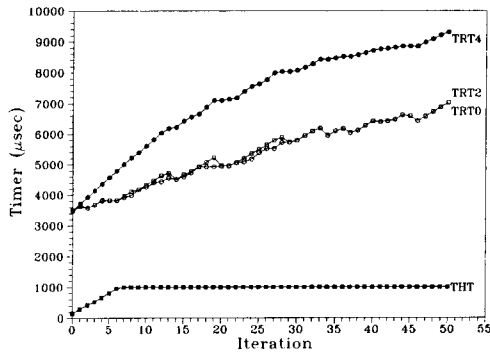


Fig. 10 Timer settings from the simulation experiment with the basic FNPM

는 것이 더욱 효과적인 토큰버스 프로토콜의 특성을 고려하지 못함으로써 우선순위 6의 전송지연은 iteration 7에서 THT의 최대 허용값인 1000 sec에 도달하여도 TRT4의 지나친 증가로 오히려 악화되는 경향을 보이고 있다. 이에 반해 우선순위 4의 전송지연은 TRT4를 계속해서 증가시킴으로써 대체로 자신의 전송지연을 줄이지만, TRT2와 TRT0의 지속적인 증가는 각각의 전송지연을 비슷한 수준으로 유지하는 역할만을 하는 것을 관찰할 수 있다.

Fig. 10에서 보이듯이 우선순위 6를 위한 THT가 7번 iteration 이후부터 일정한 것은 5000 bit의 평균 메시지 길이와 10 Mbps의 전송률을 고려하여 상한선을 설정한 것으로 1000 sec 이상의 타이머를 가지더라도 별다른 차이가 없다는 점에 근거한 것이다. 이상과 같이 THT를 상한선까지 증가시키더라도 TRT들의 길이가 너무 커서 우선순위 6의 요구 전송지연을 만족시키지 못한다. 이것은 Basic FNPM의 룰들이 타이머들을 서로 상대적으로만 조정하는 역할을 하기 때문인 것으로, 이와 같은 난점을 극복하기 위해서는 우선순위 6의 전송지연이 만족스럽지 않으면 모든 TRT들을 감소시키는 룰이 필요하다.

4.2 TCT-Based FNPM의 성능 평가

TCT-Based FNPM의 실험은 Table 5에 의한 Basic FNPM의 실험과 동등한 조건에서 수행하였으며, 이를 바탕으로 우선순위 도구의 목적과 Basic FNPM의 문제점을 어느 정도 극복하였는가를 확인하였다. 여기서도, Basic FNPM의 실험과 마찬가지로 51가지의 랜덤시드를 이용하여 시뮬레

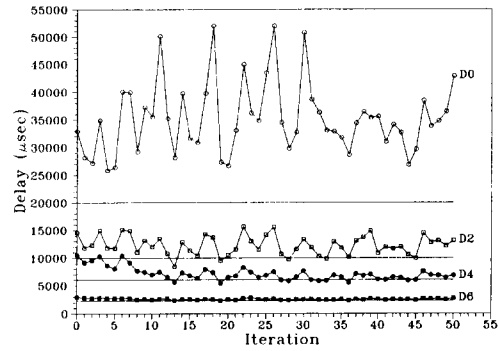


Fig. 11 Average data latency from the simulation experiment with the TCT-based FNPM

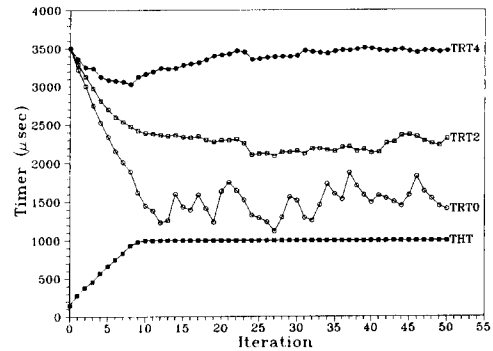


Fig. 12 Timer settings from the simulation experiment with the TCT-based FNPM

이션을 수행하였다.

Fig. 11은 TCT-Based FNPM에 의해 네트워크가 동적으로 작동한 경우의 전송지연을 보이고 있으며, Fig. 12는 이 때의 타이머 변화를 보이고 있는데 TRT4, 2, 0가 각각 3500, 2300, 1500 sec에서 일정해짐으로서 어느 정도 만족스러운 결과에 도달했음을 알 수 있다.

초기에 우선순위 6의 전송지연을 만족시키기 위하여 Table 1의 4, 5번째 룰에 의해 Fig. 12와 같이 THT를 늘이고, TRT들을 대폭적으로 감소시킴으로써 우선순위 6의 전송지연을 사용자의 요구 전송지연에 근접시킨다. 그리고 8번 iteration 이후부터는 TRT4가 증가하고 있는데, 이는 우선순위 4의 전송지연을 고려하기 위한 것으로 iteration이 증가할수록 우선순위 4의 전송지연이 사용자의 요구 전송지연에 근접하고 있음을 알 수 있다.

또한 우선순위 2도 우선순위 6, 4의 전송지연을 돕기 위해 TRT2를 늘이지 않더라도 TRT0의 대

폭적인 감소로 인해 전송지연이 일정하거나 약간씩 감소함을 볼 수 있으며, 13번 iteration 이후 우선순위 0는 상위 우선순위들의 상태에 따라 민감하게 영향을 받으며, 간헐적으로 TRT0를 증가시키거나 감소시키지만 전송지연에 있어서는 대체로 일정한 경향을 보인다.

이상의 결과를 살펴보면 TCT-Based FNPM은 상위 우선순위 순으로 급속히 전송지연을 감소시켜 우선순위 6, 4에 대해서는 사용자의 요구 전송지연에 근접시켰으나, 네트워크 용량에 대한 사용자의 요구 전송지연이 엄격해 우선순위 2, 0에 대한 전송지연에서는 뚜렷한 감소를 볼 수 없었다.

4.3 Switch-Type FNPM의 성능 평가

TCT-Based FNPM을 사용하여 Basic FNPM보다 우선순위 6에 더 많은 비중을 둔 성능관리를 수행하였지만, 우선순위 6의 메시지가 주로 제어시스템의 동작을 위한 신호나 상태를 나타내는 각종 센서 데이터일 때에는 사용자의 요구 전송지연을 반드시 만족시켜야 할 경우도 있다. 이와 같이 최상위 우선순위인 우선순위 6의 전송지연을 무조건 만족시킨후 하위 우선순위의 전송지연을 고려하려는 정책에 있어서는 하위 우선순위의 더욱 적극적인 희생을 필요로 하며, 이를 위해 Switch-type FNPM이 구성되었다.

실험은 앞에서의 조건과 동등하게 수행하였으며 Experiment I에서는 우선순위 6의 요구 전송지연을 만족시키기 위해 FNPM1이 제대로 동작하는지를 확인하였고, Experiment II는 네트워크에 70%의 트래픽이 적용되다가 갑작스럽게 네트워크의 사용량이 80%로 늘어날때 우선순위 6의 타이머 길이가 적절히 조절되는가를 조사하였다.

4.3.1 실험 I

실험의 초기조건은 Table 5와 같으며, 여기서 Di_SM은 사용자의 요구 전송지연을 의미한다.

Fig. 13은 우선순위 6가 FNPM1에 의해 하위 우선순위의 타이머를 줄이는 동시에 THT를 늘임으로서 사용자의 요구 전송지연을 만족시키고 있음을 보이며, 하위 우선순위들은 우선순위 6의 전송지연을 만족시킨후 FNPM2에 의해 우선순위 순으로 사용자의 요구 전송지연에 근접하고 있음을 보인다. 즉, 모든 우선순위의 전송지연이 사용자의 요구 전송지연을 만족시키지 못하는 초기의 상태에서

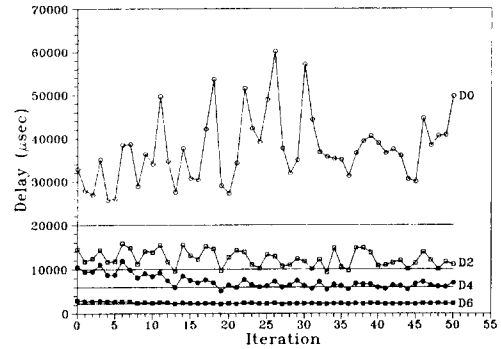


Fig. 13 Average data latency from the simulation experiment I with the switch-type FNPM

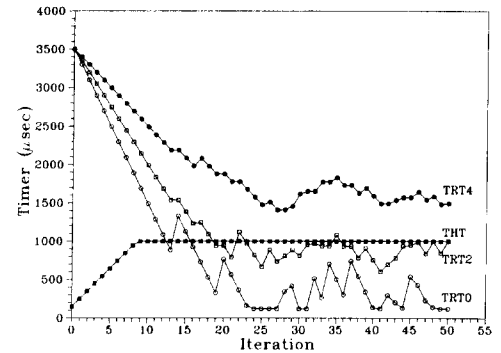


Fig. 14 Timer settings from the simulation experiment I with the switch-type FNPM

우선순위 6가 우선적으로 FNPM1에 의해 만족되어진 후, 약 13번 iteration에서 FNPM2가 수행되기 시작한다. 이후, 우선순위 6의 전송지연을 고려하면서 우선순위 4의 전송지연을 감소시켜 iteration이 증가할수록 사용자의 요구 전송지연에 근접하고 있음을 보인다.

이같은 전송지연의 단축을 위하여 우선순위 2는 자신의 전송지연을 별다른 변화없이 유지하여야 했고, 우선순위 0의 전송지연은 오히려 증가했음을 관찰할 수 있다.

Fig. 14는 Fig. 13에 따른 각 우선순위들의 타이머 추이를 보이고 있는데, FNPM1에 의해 13번 iteration까지 우선순위 6의 타이머를 늘이는 반면, 하위 우선순위의 타이머를 급격히 감소시키고 있다. 이것은 FNPM1의 제어물이 동작하고 있음을 보이며, 이 후 우선순위 6의 전송지연을 만족시키는 한도내에서 하위 우선순위의 전송지연을 줄이기 위해 FNPM2에 의해 하위 우선순위의 타이머를

Table 6 Conditions of the experiments I and II for the switch-type FNPM

| | | Priority 6 | Priority 4 | Priority 2 | Priority 0 |
|--|-------|------------|------------|------------|------------|
| Initial timer (μ sec) | | 150 | 3500 | 3500 | 3500 |
| Traffic for the first 51 iterations (%) | | 10 | 20 | 20 | 20 |
| Traffic for the second 50 iterations (%) | | 10 | 20 | 20 | 30 |
| Fuzzy input region (μ sec) | Di_SM | 2300 | 6000 | 10000 | 20000 |
| | Di_MD | 2800 | 7000 | 11500 | 22000 |
| | Di_BG | 3300 | 8000 | 13000 | 24000 |

조금씩 늘이고 있음을 보인다. 그러나 TRT0는 몇몇 iteration에서 전송을 위한 하한선인 121 sec까지 내려가기도 하였다.

이상에서 나타난 바와 같이, TCT-Based FNPM에서는 TRT4, 2, 0가 각각 3500, 2300, 1500 sec에서 일정해지는데 반해, Switch-type FNPM에서는 TRT4, 2, 0가 각각 1700, 900, 130 sec에서 일정해짐을 알 수 있다. 따라서, 이와 같은 결과는 우선순위 6의 전송지연에 대한 요구를 개선시키기 위해서 TRT들의 감소가 더 효과적이라는 관찰에 잘 부합하고 있다.

4.3.2 실험 2

이 실험은 네트워크의 트래픽이 변화되었을 때 Switch-type FNPM이 어떻게 동적으로 작동하는가를 조사하기 위한 것이다. 즉, Table 6에 나타난 것과 같이 네트워크의 부하를 70%로 유지하여

51 iteration을 수행한 후, 부하를 80%로 변화시켜 다시 50 iteration을 수행하였다.

Fig. 15의 50번 iteration까지는 70%의 트래픽이 적용되어지는 경우이며, 이때 우선순위 6, 4를 제외한 하위 우선순위들의 전송지연이 사용자의 요구 전송지연을 만족하고 있다. 따라서, 4번 iteration까지 FNPM1에 의해 우선순위 6의 전송지연을 만족시키며, 그후에는 우선순위 4도 사용자의 요구 전송지연을 만족하여 모든 우선순위의 전송지연이 사용자의 요구 전송지연을 만족하게 된다.

그러나 51번 iteration에서 갑자기 Table 6과 같이 트래픽이 증가하게 되어 모든 우선순위의 전송지연이 사용자의 요구 전송지연을 훨씬 벗어나게 된다. 따라서 우선순위 6의 전송지연만을 고려하는 FNPM1이 작동되며 목적 완수에 따라 59번 iteration에서 FNPM2로 전환되어 우선순위 4를 사용자의 요구 전송지연에 근접시키고 있다. 하지만, 우선순위 2와 0의 전송지연은 상위 우선순위의 전송지연을 위해 감소되지 못하고 있음을 보인다.

본 실험에서는 70%의 트래픽이 네트워크에 적용되어질 경우에는 사용자의 요구 전송지연을 만족시키기에 충분하였지만, 네트워크의 부하가 갑작스럽게 80%로 증가된 경우에 대해서는 모든 우선순위의 전송지연을 사용자의 요구 전송지연에 만족시키는 것은 불가능하다는 것을 보여주고 있다. 그러나, Switch-type FNPM을 구성하는 FNPM1과 FNPM2의 적절한 작동으로 최상위 우선순위의 요구 전송지연을 만족시킴으로써 우선순위 6에 상당한 비중을 두려는 성능관리 목표와 부합하는 결과를 얻을 수 있었다.

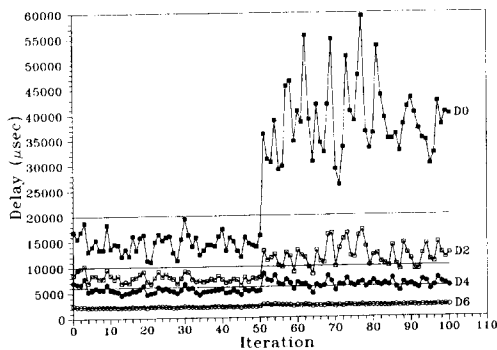


Fig. 15 Average data latency from the simulation experiment II with the switch-type FNPM

5. 결론 및 추진방향

본 논문에서는 공장자동화용 표준통신 프로토콜인 IEEE 802.4 토큰버스 프로토콜의 성능관리를 위해, 네가지 우선순위들의 타이머를 자동적으로 조절하는 3가지 유형의 퍼지 네트워크 성능관리기(Fuzzy Network Performance Manager, FNPM)를 개발하였으며 SIMAN을 이용해 개발한 토큰버스 통신망 시뮬레이션 모델을 통해서 그 성능을 검증하여, 다음과 같은 결론을 얻었다.

(1) Basic FNPM은 네가지 우선순위의 전송지연을 복합적으로 고려한 81개의 퍼지 룰을 이용함으로써 어느 우선순위에도 편중되지 않는 성능관리를 구현하였지만, 퍼지 룰이 지나치게 많을 뿐만 아니라, 우선순위 도구의 목적 자체를 모호하게 하는 경향이 있었다.

(2) 이를 보완하기 위하여, 토큰 순환시간을 도입한 TCT-Based FNPM을 개발하여 12개의 룰로 줄일 수 있었으며 Basic FNPM과 비교해서 상위 우선순위의 전송지연을 더욱 신속하게 요구 전송지연에 근접시켰다.

(3) Switch-Type FNPM에서는 throughput을 고려하여 두가지 모듈의 FNPM을 구성하였는데, 이는 FNPM1에 의해 하위 우선순위의 전송기회를 삭감하여 기존의 FNPM들에서 만족스럽지 못했던 우선순위 6의 전송지연을 만족시켰으며, 이후 FNPM2의 룰에 의해 우선순위 6의 요구 전송지연을 유지하면서, 하위 우선순위의 전송지연을 어느 정도 개선할 수 있었다. 또한, 네트워크 부하의 변동에도 Switch-Type FNPM은 동적으로 잘 작동하여, 상위 우선순위 레벨 순으로 사용자의 요구 전송지연시간에 근접시키고 있음을 보였다.

(4) 본 연구에서 개발된 3가지의 FNPM은 네트워크 사용자의 성능관리 정책에 따라 네트워크의 다양한 응용분야에 선택적으로 사용될 수 있다. 즉, Basic FNPM, TCT-Based FNPM, Switch-Type FNPM의 순서로 우선순위 6에 대한 배려가 더욱 편중되는 경향이 있으므로 실시간 메시지가 없는 사무환경에서는 Basic FNPM이 적합하다. 한편, 생산현장에서는 Switch-Type FNPM이 다양한 실시간 메시지를 위하여 보다 적합하다고 할 수 있다.

(5) 이상의 FNPM들은 네트워크 프로토콜의 어

떠한 개정도 필요하지 않으며, 또한 사용자가 네트워크 트래픽에 관한 통계적인 정보(메세지 간의 생성간격과 메세지의 길이에 대한 확률분포)없이도 그 기능을 수행할 수 있다.

(6) 이미 개발되었던 성능관리기법과 비교하였을 때, FNPM들은 PA(Perturbation Analysis), SO(Stochastic Optimization), LA(Learning Automata)와 같은 복잡한 테크닉없이도 그에 상응하는 효과를 얻을 수 있는 가능성을 제시하였다.

향후의 연구 과제로서는 FNPM에 학습 능력과 소속함수의 자동적인 튜닝(tuning) 기능의 부여, 그리고 사용자의 요구 전송지연에 대한 실현가능성의 평가, 다양한 트래픽에 대한 성능 검증과 보완 등을 들 수 있다.

참고문헌

- (1) Stallings, W., 1993, *Local and Metropolitan Area Networks*, Macmillan.
- (2) O'Rourke, J. T., 1987, *A Case for Computer Integrated Manufacturing*, Keynote Address, NSF Workshop on Computer Networking for Manufacturing Systems.
- (3) 이 석, 1994, "컴퓨터 통합생산을 위한 통신망의 성능관리 : 제1부 성능평가," 한국정밀공학회, 제11권, 제4호, pp. 126~137.
- (4) 이 석, 1994, "컴퓨터 통합생산을 위한 통신망의 성능관리 : 제2부 의사결정," 한국정밀공학회, 제11권, 제4호, pp. 138~147.
- (5) 강무진, 1989, "CIM기술에 의한 생산자동화-사례발표," 대한기계학회, 춘계학술대회 초록집, pp. 533~537.
- (6) Lee, S., 1990, "Real-Time Performance Management of Multiple-Access Networks for Computer Communications," Ph.D. Thesis, The Pennsylvania State University.
- (7) Lee, S. and Ray, A., 1993, "Performance Management of Multiple Access Communication Networks," *IEEE Jour. on Selected Areas in Communication*, Vol. 11, No. 9, pp. 1426~1437.
- (8) Klerer, S. M., 1988, "The OSI Management Architecture: An Overview," *IEEE Network*, Vol. 2, No. 2, pp. 20~29.
- (9) *Manufacturing Automation Protocol (MAP) 3.*

- O Implementation Release*, MAP/TOP Users Group, One SME Drive, P.O.Box 930, Dearborn, MI 48121.
- (10) Jones, V., 1990, *MAP & TOP Networking*, McGraw-Hill.
- (11) IEEE Computer Society, 1990, *Information Processing Systems-Local Area Networks-Part4*, IEEE Inc.
- (12) Lee, C. C., 1990, "Fuzzy Logic in Control Systems: Fuzzy Logic Controller-Part I, II," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 20, No. 2.
- (13) Cox, E., 1992, "Fuzzy Fundamentals," IEEE SPECTRUM, Oct.
- (14) Sugeno, M., 1985, "Industrial Application of Fuzzy Control," North-Holland.
- (15) 三矢直城, 田中一男, 1989, C言語による實用ファジイブック, ラッセル社.
- (16) 이상호, 손준우, 이 석, 1994, "토큰버스 프로토콜의 우선순위 시간할당에 관한 Fuzzy Algorithm의 개발," 한국자동제어학회, '94 한국자동제어 학술회의 논문집(I), pp. 547~552.
- (17) 이상호, 손준우, 이 석, 1994, "퍼지기법을 이용한 IEEE 802.4 토큰버스 네트워크의 성능관리기 개발," 대한기계학회, '94년도 추계학술대회 논문집(I), pp. 467~471.
- (18) 이상호, 손준우, 이 석, 1994, "IEEE 802.4 토큰버스를 위한 퍼지 네트워크 관리기 개발," 한국정밀공학회, '94년도 추계학술대회 논문집, pp. 461~466.
- (19) Pegden, C. D., Shannon, R. and Sadowski, R., 1990, *Introduction to Simulation Using SIMAN*, McGraw-Hill.
- (20) Law, A. M. and Kelton, W. D., 1991, *Simulation Modeling and Analysis*, 2nd ed., McGraw-Hill.