

LonWorks를 이용한 공장자동화용 네트워크의 성능향상을 위한 전송률기반 트래픽제어기의 설계와 구현

Design and Implementation of Rate-Based Traffic Controller for Performance Improvement of FA-Networks Employing LonWorks

김병희, 조광현, 박경섭

(Byoung-Hee Kim, Kwang-Hyun Cho, and Kyoung-Sup Park)

Abstract : As the interest of flexible manufacturing systems and computer integrated manufacturing systems increase, the distribution of centralized control systems using industrial control networks is getting more attention. In this paper, we investigate the rate-based traffic control of industrial control networks to improve the performance regarding the throughput, fairness, and error rates. Especially, we consider the protocol of LonWorks™ which consists of all OSI 7-layers and supports various communication media at a low cost. Basically, the proposed rate-based traffic control system is closed loop by utilizing the feedback channel errors, which shows improved performance when compared with other industrial control networks commonly operated in open loop. To this end, an additional network node called monitoring node is introduced to check the channel status without increasing the channel load. The proposed control loop is in effect whenever the feedback channel error becomes greater than an admissible value. We demonstrate the improved performance of the controlled network system in view of throughput and fairness measures by implementing the lab-scale network system based on LonWorks and through the experimentation upon it.

Keywords : industrial control networks, traffic control, fieldbus, local operating networks, LonWorks

I. 서론

최근 수년간 정보시스템은 1960년대의 주류였던 메인프레임에 의한 집중형 시스템에서 오늘날의 퍼스널 컴퓨터라든가 워크스테이션에 의한 분산형, 네트워크형으로 전환되었다. 컴퓨터 및 통신기술이 급속하게 발전하면서 통신망을 이용한 자동화시스템의 보급이 크게 증가하고 있다. 1980년대 후반부터 자동화 현장에 설치된 각종 제어 및 자동화관련 장비들에서 생성되는 데이터들의 실시간 통신을 지원하는 동시에 가격이 저렴한 네트워크 시스템의 필요성이 제기되어 왔으며, 이러한 요구사항을 만족시키기 위해서 개발된 네트워크 산업용 제어네트워크이다[8], [9]. 산업용 네트워크가 출현하기 이전에 필드 장비간의 통신은 주로 4~20mA 등의 아날로그 신호들이 사용되었다. 이러한 아날로그 신호들은 배선이 복잡하고 신호전송 시 외란에 민감하게 반응한다는 단점을 가지고 있다. 이런 기존의 아날로그 방식을 산업용 네트워크를 이용해 대체함으로써 배선이 간단해져 비용의 절감을 가져왔다. 또한 전송신호를 디지털화함으로써 기존의 아날로그 신호보다 외란에 강해진다는 장점도 가지게 되었다. 더욱이 필드버스를 이용하여 동일한 배선으로 여러 가지 센서 및 기기들을 연결함으로써 네트워크상에서 각종 기기들을 모니터링 할 수 있을 뿐 아니라 문제발생시 네트워크상에서 간단한 조치를 취할 수 있어 시스템의 유지 보수에 사용되는 비용도 절감할 수 있게 되었다.

다. 산업용 제어네트워크가 가져야 할 중요한 요소를 보면 다음과 같다 : ①응답 시간을 예측할 수 있어야 한다, ②노드 구성 시 가격이 비싸지 않아야 한다, ③결함에 강해야 한다, ④이미 존재하는 시스템과 호환성을 가져야 한다. 이런 제어용 네트워크를 엄밀히 구분하면 센서버스, 디바이스버스, 그리고 필드버스로 구분할 수 있다[10]. 필드버스는 현재 표준의 개발활동이 진행 중에 있고 이 산운용을 목적으로 하고 있는 디바이스버스와 센서버스 역시 모든 자동화산업에 걸쳐 주목을 받으며 새로운 이슈로 도약하고 있다. 센서버스는 센서와 액츄에이터 사이에서 비트레벨 메시지를 전송한다. 이런 센서버스가 비트레벨 센서, 스위치, 액츄에이터에서 고속 통신을 제공하는 반면 디바이스버스는 컨트롤러 이하의 바이트레벨 디바이스의 통신요구 사항을 제공하도록 되어있다. 본 논문에서는 이런 제어용 네트워크의 일종인 에셀론사의 LonWorks[2]-[4], [16]를 이용하여 구성된 네트워크 시스템을 대상으로 한다. LonWorks는 앞서의 센서버스, 디바이스버스, 필드버스가 가지는 다양한 장점을 모두 지니며 최근 산업계의 자동화용 네트워크를 위한 새로운 네트워크로 부상하고 있다. 그러나 한편으로 LonWorks에서 채택하고 있는 프로토콜(LonTalk™)이 CSMA (Carrier Sense Multiple Access)[18], [19]방식의 비결정적 (nondeterministic) 특성을 지니기 때문에 실제 산업 자동화 현장에서의 LonWorks이용이 지연되고 있다. 본 논문에서는 이러한 비결정적 프로토콜이 가지는 단점을 해결하기 위해 전송률기반 트래픽제어기를 설계하고자 한다. 일반적으로 전송률기반 트래픽제어기는 ATM (Asynchronous Transfer Mode) 네트워크에서 많이 연구

접수일자: 1999. 10. 4. 수정일자: 2000. 2. 8

김병희, 조광현, 박경섭: 울산대학교 전기전자및자동화공학부.

※본 연구는 정보통신부의 정보통신 우수시범학교 지원사업에 의하여 수행된 것입니다.

발전되어 왔다[6], [7]. ATM 네트워크의 ABR (Available Bit Rate) 서비스에서는 RM(Resource Management) 셀을 이용해 송신측에 제한을 주는 방식으로 트래픽을 관리한다. 그 외 여러 가지 스위치 메커니즘을 사용하고 있는데 대표적인 것이 EFCI(Explicit Forward Congestion Indication) 스위치와 ER(Explicit Rate) 스위치이다. 하지만 대부분의 이런 트래픽제어는 ATM이나 TCP/IP와 같은 네트워크에 이용되고 있다. 따라서 본 논문에서는 주로 개루프(open loop)형태를 이용하는 산업용 네트워크에 전송률기반의 트래픽제어기를 설계함으로써 제한을 인가하여 폐루프(closed loop)를 형성해 네트워크 성능향상을 보이게 한다. 실험을 위해서 노드를 4개 제작하고 LonBuilder[15]에서 제공하는 에뮬레이터2개를 트래픽 발생기로 사용한다. 노드 3개, 4개가 각각 서로 통신할 때 각 노드의 전송률, 오류률, 공정성, 평균지연등을 검증한다. 2절에서는 LonWorks의 일반적인 구조와 특성에 대해 기술한다. 3절에서는 실험을 위해 직접 제작한 Lon노드에 대해 기술한다. 4절에서는 전송률기반 트래픽제어기의 설계에 대해서 기술한다. 5절에서는 실험결과를 통해 제안된 제어알고리즘의 효율성을 검증하고 6절에서 결론을 맺는다.

II. LonWorks의 구조 및 특성

LonWorks는 컴퓨터 산업에서 일어난 분산화, 네트워크화, 개방화를 자동화 세계에서 실현할 것을 목적으로 에셀론사에 의해 고안 되었다. LonWorks는 뉴런 칩이라고 명명된 저 비용의 LSI를 각종 센서, 스위치라든가 액츄에이터에 분산 배치하고 그들을 접속하여 제어네트워크 시스템을 구축한다[11].

1. LON (Local Operating Network)

LonWorks의 LON은 Local Operating Network[5]를 의미한다. 네트워크는 크게 데이터 네트워크와 제어용 네트워크로 구분된다. 데이터 네트워크는 정보의 양이 많고 전송의 빈도가 많지 않으며 불규칙적인 특징을 갖는다. 반면 제어용 네트워크는 전송할 양이 적고 신뢰성이 있어야 하며 실시간적으로 수행되어야 한다[11]. LON과 LAN은 모두 네트워크이지만 그 목적은 전혀 다르다. LAN은 Local Area Network의 약자이며 컴퓨터, 프린터, 디스크 서버 등의 사이에서 문서나 이미지 등 대량의 데이터를 전송하여 데이터 처리를 목적으로 한다. 그에 비해 LON은 각 노드간에서 명령이나 상태 등 소량의 데이터를 송.수신하여 네트워크 전체 또는 그룹 단위로 고도로 조직화된 일을 해서 제어하는 것을 목적으로 한다[12]. LON은 최하위 센서, 액츄에이터 같은 디바이스와 통신, 모니터/컨트롤 기능을 담당하는 디바이스를 마스터, 데이터 네트워크가 하나의 동일한 프로토콜로 묶여 하나의 라인으로 통신하고자 하는 개념이다. 이 개념은 컴퓨터 산업이 기존의 메인 프레임/터미널 시절에서 워크스테이션, 이제는 인텔리전트 PC, 클라이언트/서버 아키텍처로 발전해 왔다는 점을 착안하여 컨트롤 산업도 기존의 마스터/슬레이브, DDC (또는 DCS, PLC) 방식에서

점 대 점, 평면 구조 (flat architecture)로 가야 한다는 생각에서 비롯된 것이다.

2. LonWorks와 다른 산업용 네트워크 시스템과의 차이점

LonWorks는 종종 다른 산업용 버스시스템과 비교된다. 앞서 설명한바와 같이 산업용 버스 시스템을 센서 버스, 디바이스 버스, 필드버스 3종류로 분류될 수 있는데 그 중 LonWorks만이 모든 버스로서의 기능을 갖추고 있다. 그 차이점은 LonWorks는 센서, 액츄에이터 등의 디바이스를 모두 인텔리전트화 한 다음 그것들을 점 대 점 (peer to peer)으로 연결하여 분산적으로 제어 시스템을 구축하는 반면 다른 산업용 버스의 대부분은 PLC, DCS와 같은 디바이스간의 통신망으로 위치가 부여되어 있기 때문에 집중형 제어에서 분산형제어로 아키텍처를 근본적으로 전환하지 못하기 때문이다. 또한 LonWorks는 OSI-7 계층을 모두 지원하므로 확장성 및 상호 운용성이 높고 여러 가지 통신매체 (전력선, RF, 쌍 꼬임선, 광 파이버등)를 다양하게 사용할 수 있고 자유 구조 (free topology)를 지원하며 점 대 점 통신을 하기 때문에 상위의 디바이스가 제 기능을 상실해도 상호 통신이 가능하다는 점들을 생각할 수 있다[13].

3. LON의 구조

LonWorks는 다음의 4가지 요소로 구성된다.

- 뉴런 칩(neuron chip)
- 트랜시버(transceiver)
- 통신 프로토콜(LonTalk)
- 통신망운용시스템(LNS : LonWorks Network Service)

3.1 뉴런 칩 (neuron chip)

뉴런 칩에는 통신 및 연산을 위한 3개의 8비트 프로세서와 모델에 따라서 내부 메모리 또는 외부 메모리를 사용한다. 뉴런 칩이 가지는 3개의 프로세서는 MAC 프로세서, 네트워크 프로세서, 응용 프로세서이다. 프로세서 자체가 프로토콜 스택을 포함하고 있으므로 디바이스 개발자는 디바이스의 응용 프로그램만 작성하고 통신과 관련한 프로토콜을 모두 구현할 필요가 없다. 뉴런 칩은 내부 메모리만 사용하느냐 외부메모리를 사용하느냐에 따라 각각 3120, 3150칩으로 분류된다. 뉴런 칩을 가지고 디바이스 개발자가 해당 디바이스의 어플리케이션 코드를 작성할 때 뉴런 C라는 언어를 사용한다. 뉴런 C는 ANSI C의 일종으로 분류되는데, 특이한 점은 When절에 의한 이벤트 방식(event-driven)으로 구현한다는 점이다.

3.2 트랜시버 (transceiver)

LonWorks 트랜시버는 매체와의 통신을 담당한다. LonWorks의 장점 중 하나는 다양한 전송 매체를 지원한다는 것이다. 쌍 꼬임선(twist Pair : TP), 전력선 (과워라인에 데이터를 실어 보냄), 링크파워(link power, 약전, 24V선에 데이터를 실어 보냄), 무선(wireless, RF), 동축케이블(coaxial), 광 (fiber optic), 적외선(infrared) 등 여러 가지 통신 매체를 지원한다.

3.3 통신 프로토콜 (LonTalk)

LonTalk프로토콜은 국제표준화 기구인 OSI 참조모델(7 계층)에 준거하여 기술되어있다. 다른 필드버스와는 다

르게 OSI 7계층을 모두 지원하기 때문에 상호 운용성 측면에서 유리하다. 또한 LonTalk프로토콜은 CSMA/CA(Carrier Sense Multiple Access/Collision Avoidance) 방식을 사용한다. 따라서 주기적인 데이터의 전송요구사항은 비주기적인 데이터의 전송요구사항에 포함된다. LonTalk프로토콜은 특별히 Predictive P-persistence CSMA방식을 사용한다. 이 방식은 예상 채널 부하의 지식을 이용해서 채널 접근을 난수화 하는 충돌 회피 기술이다. MAC 알고리즘의 예측 가능성은 backlog 예측에 기초한다. 각 노드는 현재 채널 backlog(BL)의 어림잡은 값(estimated value)을 유지하고 있는데 이 BL값은 MPDU(MAC Protocol Data Unit)를 보내거나 받으면 증가하고 각 패킷 주기마다 주기적으로 감소한다[14].

3.4 운용시스템 (LNS : LonWorks Network Service)

LonWorks가 여타 필드버스와 차별화 될 수 있는 점은 일관된 네트워크 관리(network management) 솔루션까지 갖춘 컨트롤 네트워크라는 것이다. LNS는 LonWorks네트워크의 OS(operating system) 또는 NMS(network management system) 이라 할 수 있다. LNS는 시스템 설치자 및 통합자가 서로 다른 벤더의 제품을 묶어서 전체 네트워크를 구성하는데 편리한 이점을 제공한다. 즉, LNS는 개인용 컴퓨터를 비롯하여 다른 마이크로 프로세서나 마이크로 컨트롤러에 구현되어 LonWorks네트워크를 구성,설치,모니터링/컨트롤 할 수 있는 어플리케이션을 작성할 수 있게 해준다. 특히 LNS는 클라이언트/서버 아키텍처를 지원하므로 네트워크 관리를 관장하는 네트워크 서비스 서버(network service server : NSS)에 127개까지의 네트워크 서비스 인터페이스(network service interface) 도구들을 접속할 수 있다. 즉, 일반 터미널, 모뎀을 통한 원격 접속, 노트북, 휴대 가능한 작은 계측기, 심지어는 IP상의 개인용 컴퓨터 등이 LNS의 클라이언트로 LonWorks네트워크에 접속하여 네트워크를 모니터링/컨트롤할 수 있다.

III. 실험을 위한 Lon노드 제작

실험을 위해서 자체적으로 Lon노드를 구성 제작하였다. 노드는 확장성을 위해서 3150칩을 이용하고 입/출력 단자를 따로 분리하여 그 위에 다른 응용 보드를 추가할 수 있도록 하였다. 또한 모니터링을 위한 컴퓨터와 인터페이스를 위해서 RS232C 포트를 추가하였다. 실험을

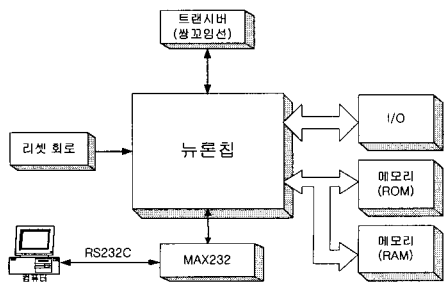


그림 1. 제작한 Lon노드의 구조도.
Fig. 1. Block diagram of Lon-node.

위해 여러 가지 통신 매체 중 쌍꼬임선을 사용하였고 트랜시버로는 1.25Mbps를 지원하는 트랜시버를 사용하였다. 아래 그림은 노드의 구조도 이다.

IV. 전송률기반 트래픽제어기 설계

대부분의 제어용 네트워크는 개루프 특성을 가진다. 하지만 이러한 개루프 시스템은 노드가 많아질수록 오류도 높아질 뿐 아니라 여러 노드가 동시에 다발적으로 전송을 시도할수록 전송률도 떨어지는 단점을 가지고 있다. 따라서 본 논문에서는 채널의 오류를 케환 받고 전 노드로부터 들어오는 데이터의 주기를 측정해서 다음 데이터가 들어오는 시점을 예상함으로써 이런 개루프 시스템을 폐루프 시스템으로 만들어 네트워크 성능개선을 하고자 한다. 앞서 설명한 바와 같이 LonWorks는 데이터링크 계층이나 네트워크 계층이 모두 ASIC화 되어 있기 때문에 본 논문에서는 이러한 계층을 포함한 전체 네트워크를 제어대상 시스템으로 간주 하고 채널의 오류 신호를 케환 받았다. 실험을 위해서 원형 대기열을 설계하고 데이터는 항상 대기열을 거쳐서 네트워크 채널을 통해 전송되는 형태를 취했다. 일반적으로 데이터는 임의의 시간에 임의의 데이터가 입력된다고 가정한다. 그러나 입력데이터 생성주기의 변화는 작은 시간범위 내에서 이루어지며, 따라서 각 노드에서 연속적으로 수신되는 데이터간격은 작은 시간범위 (수신데이터간격의 2-3배) 내에서는 일정하다고 가정한다. 그림 2는 그 개념도를 보여준다.

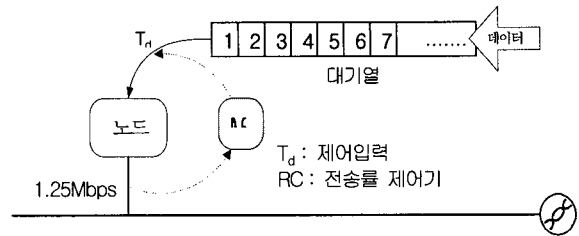


그림 2. 각 노드에서의 전송률기반 제어를 도입한 폐루프의 개념도.

Fig. 2. Schematic diagram of the closed loop network configuration at each node.

실험을 위해 다음과 같이 시스템을 구성한다. LonWorks는 다른 산업용 네트워크와는 다르게 비결정론적 통신 프로토콜을 사용한다. 따라서 어떤 특정한 노드로 정체 현상이 일어나면 다른 노드들은 통신이 두절되는 경우가 발생한다. LonWorks는 커넥션 기반 네트워크가 아니라 정보 기반 네트워크이다. 따라서 네트워크가 일단 구성 되고 나면 네트워크 변수 바인딩 정보를 기반으로 전송을 하므로 특정한 노드로 전송이 집중되는 정체 현상은 거의 발생하지 않는다. 즉 일단 노드1에서 노드2로 네트워크 변수가 바인딩 되고 나면 노드1의 출력 네트워크 변수의 변화에 따라 노드2가 반응한다. 따라서 네트워크 구성이 끝나면 노드1에 의해 노드3이 반응하지는 않는다. 따라서 본 연구에서는 어떤 특정한 노드로 데이터가

집중되는 경우는 고려하지 않는다. 네트워크 채널 상태를 좀더 현상과 비슷하게 만들기 위해서 트래픽 발생기를 설치하였다. 이 트래픽 발생기는 항상 서로 양방향 통신을 하면서 다른 노드들에게는 영향을 미치지 않고 다만 채널 상태에만 영향을 미친다.

1. 전송률

전송률은 일반적으로 전송한 데이터와 성공적으로 전송된 데이터의 비율로 나타낸다. 본 논문에서는 전송률을 나타내기 위해 32byte 단위의 데이터를 1000회 전송하고 그 중 성공적으로 전송된 데이터의 횟수를 측정하였다. 그림 5와 그림 7에 각 실험에서의 전송률을 그래프로 나타내었으며 초당 전송된 평균 패킷을 나타내었다.

2. 공정성

네트워크 상호간 통신 수준에서 공정성은 일반적으로 사용자 지연이나 망의 처리량에 의해 제한된다. Mark는 공정성은 모든 사용자에게 똑 같은 만족을 제공하는 능력이라고 정의 하였다[1]. 사용자의 만족이란 모든 사용자가 거의 비슷한 평균 지연시간을 가지고 채널을 접근하고 처리량 또한 일정하다는 것을 의미한다. 성능 향상과 함께 공정성을 보장하기 위해서 공정성 지수를 설정한다. 각각의 노드에 입력되는 데이터와 최대 오류와의 비율과 최소 오류와의 비율의 차가 0에 접근할수록 각 노드는 공정한 채널 접근 기회를 가진다고 정의 하였다. 따라서 본 논문에서는 다음과 같이 공정성 지수를 설정한다.

$$J_f = \frac{\max(I_i - s_i)}{I_i} - \frac{\min(I_i - s_i)}{I_i} \quad (1)$$

여기서 I_i 는 i번째 노드의 입력 데이터 개수, S_i 는 i번째 노드가 성공적으로 보낸 데이터 개수를 나타낸다.

본 논문에서는 네트워크의 케환 정보를 이용하여 앞의 전송률을 최대화 하며 위 공정성 지수를 최소화 시키는 제어입력을 결정 한다.

3. 제어알고리즘

산업용 제어네트워크는 어떤 큰 데이터를 꾸준히 보내는 것이 아니라 그때 그때의 불규칙적인 데이터를 처리해야 하기 때문에 공정성 측면에서 뿐만 아니라 제어네트워크 특성상 어떤 데이터가 들어오고 그 다음 데이터가 들어오는 시간의 간격은 제어입력을 결정할 때 상당히 중요한 의미를 갖는다. 또한 전송할 데이터가 많이 있는 노드에게 좀더 많은 기회를 주기 위해서 대기열에 들어 있는 데이터의 양도 고려한다. 본 논문에서는 모든 데이터의 우선순위가 동일하다고 가정한다. 예측된 시점 내에 오류가 많고, 데이터가 대기열에 입력 되는 시간이 길수록 채널 접근을 적게 하고 대기열에 있는 데이터의 수가 많을수록 채널 접근을 많이 시도해야 한다. 그러면 T_d^n 을 각 노드에서 n번째 데이터전송시 대기열로부터 네트워크 채널로 데이터 전송을 시도하기까지의 인위적인 지연시간이라고 하자. 본 논문에서 제안한 제어알고리즘의 n번째 제어입력 T_d^n 는 다음과 같이 결정된다($n \in \mathbb{N}$).

$$T_d^n = k_1 \cdot E^{(n-1)} + k_2 \cdot [(n-1)\text{번째 데이터 전송시 데}$$

이터가 대기열에 입력되는 시간의 총합] + $k_3 \cdot$

$$\left[\frac{1}{(n-1)\text{번째 데이터전송시 대기열에 들어있는 데이터의 수}} \right] \quad (2)$$

여기서 $E^{(n-1)}$ 은 (n-1)번째 데이터 전송오류를 k_1, k_2, k_3 는 가중치를 각각 의미한다. 이때 가중치 k_1, k_2, k_3 는 실험을 통해서 결정한다. 제안한 알고리즘의 케환변수 중 송신데이터가 대기열에 입력되는 순차적인 시간간격은 중요한 의미를 가질 뿐 아니라 제안한 알고리즘의 수렴성질로부터 '데이터가 대기열에 입력되는 시간의 총합'과 알고리즘의 수렴성이 원형대기열의 영향을 받으므로 '대기열에 들어있는 데이터의 수'가 주요매개변수이다. 따라서 k_1 값은 고정시킨 뒤 k_2 와 k_3 값을 실험적으로 최적화 시킨 후 k_1 값을 다시 실험적으로 최적화 시킨다. 이러한 방법을 통해 항상 최적의 가중치를 얻을 수는 없으나 주어진 네트워크의 성능을 만족시키는 범위 이내로 준최적화 시킬 때까지 실험을 반복하여 가중치를 결정한다. 위의 개념을 바탕으로 제어알고리즘을 다음과 같은 단계로 나누어 기술할 수 있다.

3.1 단계 1

먼저 다음 데이터를 보낼 시기를 예측하기 위해서 다른 노드로부터 들어오는 데이터의 주기를 계산한다. 주기를 계산하기 위해서 어떤 시간동안 들어오는 데이터를 저장해서 그 간격을 계산한다. 예를 들어 만약 첫 번째 데이터가 3ms에 들어오고 두 번째 데이터가 6ms, 세 번째 데이터가 10ms에 들어왔다면 그 다음 데이터는 13~14ms에 들어올 것을 예측할 수 있다.

3.2 단계 2

단계1에서 다음에 들어올 데이터를 예측하고 나면 다음 데이터를 보낼 전송률을 결정한다. 만약 제어입력 T_d^n 가 다음 데이터가 들어올 예측 시간 보다 작다면 T_d^n 의 배수 만큼 데이터를 전송한다. 예를 들어 만약 제어입력이 3ms이고 예측시간이 8ms이면 2번 전송을 할 수 있다.

3.3 단계 3

만약 제어입력이 예측시간보다 길다면 이것은 이 노드가 다른 노드에 비해서 전송할 데이터가 적다는 것을 의미한다. 그러므로 제어입력 시점에 한번 전송을 한다.

3.4 단계 4

만약 다음 데이터가 예측한 시점 보다 빨리 들어온다면 이전의 주기는 무시하고 이번 시점을 포함해서 주기 예측을 다시 한다. 그 후 주기와 데이터 전송률을 다시 결정해서 전송을 시도한다. 단계 2를 반복한다. 다음은 제어알고리즘의 흐름도이다.

위 제어알고리즘은 허용된 채널 오류률을 초과할 때 작동한다. 채널 오류률의 허용치는 네트워크 구성시 결정된다. 전송을 하면서 다른 노드로부터 들어오는 수신데이터의 시간간격(cycle)을 측정을 해서 다음 수신데이터의 도달시간을 예측한다. 이 예측된 시점과 제어입력(T_d)을 비교하여 데이터를 송신할 시점을 결정하여 전송을 시도한다. 만약 데이터를 송신하기 전에 또 다른 수신데

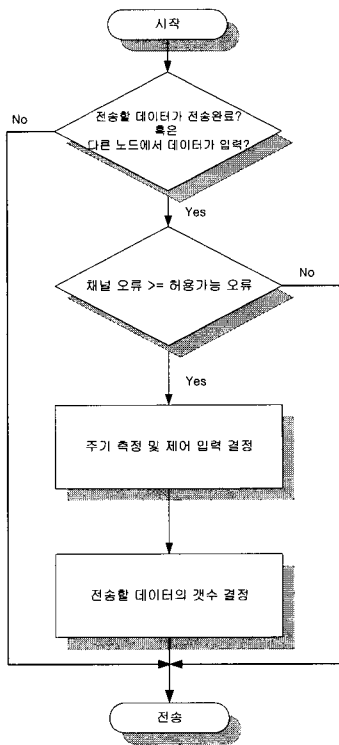


그림 3. 제어알고리즘의 흐름도.
Fig. 3. Flow chart of control algorithm.

이터가 들어오면 이전의 값들은 무시하고 수신데이터의 주기를 재 측정하여 다음 제어알고리즘을 적용한다. 위와 같은 제어알고리즘을 기반으로 트래픽제어기를 설계한다. 제안한 제어알고리즘의 안정성(stability)을 보이기 위해 수렴성(convergency)에 대해 기술한다. 만약 $(n+1)$ 번째 수신데이터가 계속해서 제어입력(T_d)보다 빨리 도달하게 되면 제어알고리즘은 수렴하지 못하고 대기열에 송신데이터가 가득 차게 된다. 만약 대기열이 무한히 크다면 언젠가는 안정되겠지만 실제 시스템에서는 불가능하다. 따라서 다음과 같이 그 수렴성의 한계가 정해진다. 즉, l_q 를 대기열의 크기라고 할 때 제어알고리즘의 수렴성 여부는 모든 정수 m 에 대해 $d^{(n+m)} > T_d^{(n+m-1)}$ 이면 항상 수렴하고 $d^{(n+m)} < T_d^{(n+m-1)}$ 일 때는 $n_q \leq l_q$ 의 범위 내에서 수렴하게 됨을 알 수 있다.

V. 실험

제안한 제어알고리즘을 검증하기 위해 자체 제작한 Lon노드를 이용하여 실험실 수준의 제어 네트워크를 구성하였다. 트래픽 발생기는 LonBuilder에서 제공하는 에뮬레이터를 이용하였다. 실험은 노드가 3개일 때와 4개일 때, 각각 제어알고리즘이 있을 때와 없을 때 전송률과 채널의 오류를 측정, 비교하였다. 실험은 각각 5번씩 수행하여 그 평균 값을 계산하였다.

1. 주요매개변수의 가중치 결정

주요매개변수의 가중치를 결정하기 위해 k_1 값은 1로 고정시키고 나머지 두 값을 변화 시켜가며 실험을 하였다. 그 결과 k_2 가 7이고 k_3 가 3일 때 전송률이 가장

높게 나타났고 공정성도 평균이하의 값을 나타내었다. 제어입력 T_d 는 앞서 기술한 것과 같이 에러에 대한 요소와 대기열에 들어있는 송신데이터의 요소들로 구성된다. 그 중 노드의 개수와 관련이 있는 에러에 대한 요소는 주요 매개변수가 아니기 때문에 결정된 가중치는 노드의 개수에 거의 영향을 받지 않음을 실험을 통해 확인하였다.

2. 실험 1 (노드가 3개일 때)

실험을 위해서 1번 노드는 2번 노드에, 2번 노드는 3번 노드에 3번 노드는 1번 노드에 데이터를 전송 하게 구성하였다. 트래픽발생기(TG)는 다른 노드에 영향을 주지 않고 채널 상태에만 영향을 준다. 앞서 설명 한 바와 같이 LonWorks는 정보기반 네트워크이기 때문에 실제 현장에서도 네트워크 구성시 네트워크 변수가 바인딩되고 나면 그 바인딩된 출력변수의 변화에 의해서만 입력변수가 변화하기 때문에 소스노드(source node)가 데이터를 보낼 목적노드(destination node)가 명백하게 정의된다. 다음은 그 구성과 실험 결과이다.

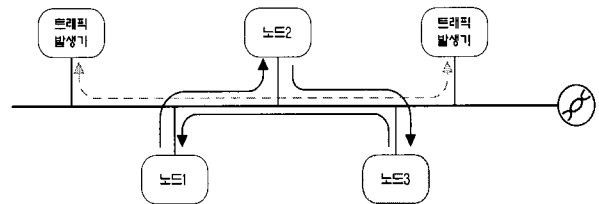


그림 4. 실험1의 구성도.
Fig. 4. Configuration of experiment 1.

2.1 실험 1 결과

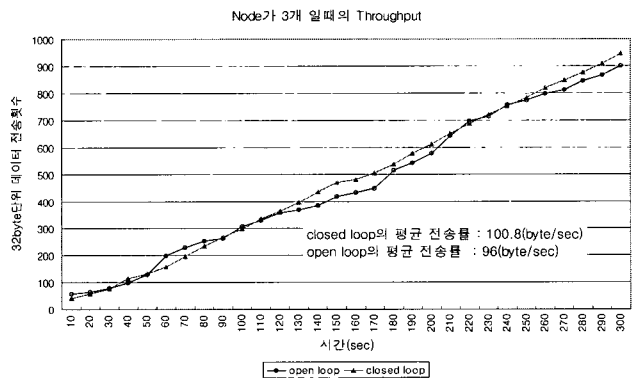


그림 5. 실험1의 데이터 전송량.
Fig. 5. The throughput of experiment 1.

다음의 표는 노드가 3개일 때 개루프일 때와 폐루프일 때 공정성 지수와 오류률, 평균지연의 비교이다. 공정성 지수는 본문에서 제시한 (2)을 따랐다. 이때 가중치 k_1 은 1이며 k_2 는 7, k_3 는 3이다. 위 그림 5와 표1에서 개루프일 때에 비해 폐루프일 때 평균지연의 큰 증가 없이(8.9%) 전송률과 공정성, 오류률이 향상 (각각 87.2%, 45%)되었음을 보여 준다.

표 1. 실험1의 공정성 지수 및 오류률, 평균지연.
Table 1. Fairness, error rate, and average delay of experiment 1.

	개루프	폐루프
J_f	0.094	0.021
오류률 (%)	10	3.68
평균지연 (ms)	33.9	37.2

3. 실험 2 (노드가 4개일 때)

실험을 위해서 1번 노드는 2번 노드에, 2번 노드는 3번 노드에 3번 노드는 4번 노드에 4번 노드는 1번 노드에 데이터를 전송 하게 구성하였다. 다음은 그 구성과 실험 결과이다.

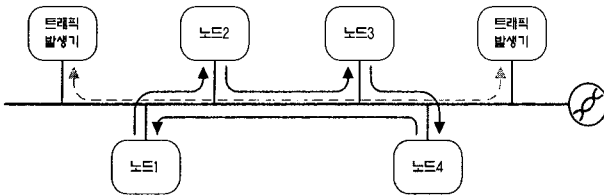


그림 6. 실험2의 구성도.
Fig. 6. Configuration of experiment 2.

3.1 실험 2 결과

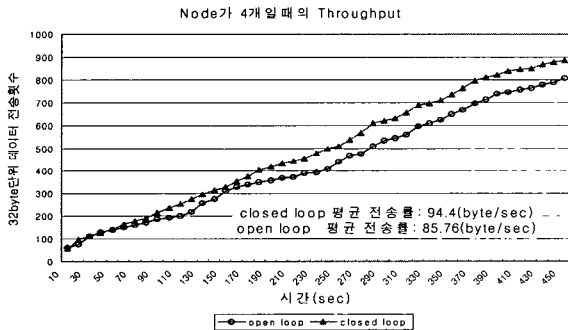


그림 7. 실험2의 데이터 전송량.
Fig. 7. The throughput of experiment 2.

다음의 표 2는 노드가 4개일 때 개루프와 폐루프의 공정성 지수, 오류률과 평균지연의 비교이다. 이때 가중치 k_1 은 1이며 k_2 는 7, k_3 는 3이다.

표 2. 실험 2의 공정성 지수와 오류률, 평균지연.
Table 2. Fairness, error rate, and average delay of experiment 2.

	개루프	폐루프
J_f	0.16	0.034
오류률 (%)	19.4	9.8
평균지연 (ms)	113.2	121.9

노드가 4개일 때 역시 개루프에 비하여 폐루프일때 평균지연의 큰 증가 없이(7.6%) 전송률(12%)과 공정성,

오류률이 향상 (각각 78%, 49%)되었음을 보여준다.

각각의 경우에 데이터전송의 평균지연이 약간 증가하였으나 실시간 데이터의 일반적인 허용지연이 데이터의 크기가 10kbit 이하일 때 20 ~ 100ms인 점을 감안하면 충분히 데이터의 실시간성을 보장함을 알 수 있다. 실험 결과에서 알 수 있듯이 노드가 많아질수록 제안한 제어 알고리즘을 통해 성능이 향상되는 것을 알 수 있다[10].

VI. 결론

본 논문에서는 제어용 네트워크의 일종인 LonWorks를 이용하여 그 특성을 파악하고 통신노드를 자체 제작하여 그 성능을 개선하기 위해서 전송률기반 트래픽제어기를 설계하였다. 데이터의 주기를 예측하고 채널의 오류신호를 캐환 받아 각 노드의 데이터 전송률을 결정함으로써 성능을 개선하였다. 성능분석을 위해 전송률과 공정성이라는 2가지 효율 지수를 선정하고 이를 토대로 성능향상을 검증하였다. 특히 채널에 부하를 주지않고 채널의 상태를 감시하며 전송률을 측정하기 위해서 모니터링 노드를 도입하였다. 실험을 통해 향상된 네트워크 성능을 검증하였으며 그 결과 노드의 수가 증가할수록 제안한 트래픽제어기에 의해 네트워크의 성능이 더욱 향상됨을 확인 할 수 있었다.

참고문헌

- [1] C. Douligeris and L. N. Kumar, "Access to a network channel: A survey into the unfairness problem", *Proc. ICC'92*, pp. 1184-1185, 1992.
- [2] J. M. Alonso, J. Ribas, J. J. Del Coz, M. Rico, "Intelligent control system for fluorescent lighting based on LonWorks technology", *Proc. of 24th Annual Conf. of IEEE Industrial Electronics Society*, vol. 1, pp. 92-97, 1998.
- [3] R. A. Ziemerink and C. P. Bodenstein, "Utilising a LonWorks control network for factory communication to improve overall equipment effectiveness", *Proc. of IEEE Int. Symp. on Industrial Electronics '98* vol. 2, pp. 684-689, 1998.
- [4] P. Palensky, R. Poster, and H. Reiter, "Demand side management in private homes by using LonWorks", *Proc. of IEEE Int. Workshop on Factory Communication System*, pp. 342-347, 1997.
- [5] P. W. M. Tsang and R. W. C. Wang, "Development of a distributive lighting control system using local operating network", *IEEE Trans. on Consumer Electronics*, vol. 4, pp. 879-889, 1994.
- [6] K.-H. Cho and J.-T. Lim, "Supervisory rate-based flow control of ATM networks for ABR services", *IEICE Trans. on Communications*, vol. E81-B, no. 6, pp. 1269-1271, 1998.
- [7] S. Chong, R. Nagarajan and Y. T. Wang, "First-order rate-based flow control with dynamic queue threshold for high-speed wide-area ATM

network”, *Computer Network & ISDN Systems*, vol. 29, no. 17-18, pp. 2201-2212, 1998.

[8] 홍승호, 김기암, 김지용, 고성준, “분산제어 및 자동화시스템과 필드버스”, 제어·자동화·시스템공학회지, 제2권, 제4호, pp. 19-20, 1996.

[9] 김기암, 홍승호, “자동화시스템에 Profibus 네트워크 인터페이스 구현 및 성능평가”, 제어·자동화·시스템공학 논문지, 제4권, 제1호, pp. 113-114, 1998.

[10] 홍승호, “필드버스 기술동향”, 제어·자동화·시스템공학회지, 제4권, 제6호, pp. 13-14, 11월호, 1998.

[11] 편집부, “Device Network이란 무엇인가”, 월간 CONTROL, pp. 48-54, 1월호, 1998.

[12] 백우진, 박경섭, “지능 분산제어를 위한 LON 시스템 구축”, 석사학위논문(울산대학교), pp. 12-13, 1998.

[13] 枝隆可, “CIM/FA에 혁명이 기대되는 분산형 지적

제어네트워크”, 자동화기술, pp. 39-47, 8월호, 1997.

[14] 高井 正美, “계측기술과 LonWorks”, 월간 자동제어계측, pp. 72-73, 11월호, 1998.

[15] *LonTalk Protocol Specification ver3.0*, pp. 20-23, Echelon, Co., 1994.

[16] *LonBuilder User's Guide*, Echelon, Co., 1995.

[17] *LonWorks Technology Device Data*, Motorola Inc., USA, 1995.

[18] Lai R, “Performance results for the CSMA/CD protocol using GreatSPN”, *Journal of Systems & Software*, vol. 37, no. 1, pp. 75-90, 1997.

[19] Liu Y. S, Wu S. W, “Adaptive coding scheme for wireless LANs with CSMA/CA MAC protocol”, *Electronics Letters*, vol. 34, no. 15, pp. 1450-1452, 1998.



김 병 희

1998년 울산대학교 전기전자및자동화공학부 제어계측공학과 졸업. 1998~현재 동 대학원 석사과정. 관심분야는 분산제어, 실시간 제어, IBS 시스템 및 산업용네트워크.



박 경 섭

(生)1951년 (專) 광섬유 센서 (學) 74. 서울대 공학사, 76.서울대 공학석사, 89. (美) The University of Texas at Austin 공학박사. 현재 울산대 전기전자 및 자동화공학부 교수. 관심분야는 광센서, 초고속산업용 네트워크, 자율이동로봇.



조 광 현

1993.2. KAIST, 전기및전자공학, B.S. 1995.2. KAIST, 전기및전자공학, M.S. 61998.8. KAIST, 전기및전자공학, Ph.D. 1999.2. KAIST, 전기및전자공학과, 위촉연구원 및 연수연구원. 1999.3.~현재. 울산대학교, 전기전자및자동화 공학부, 전임강사 관심분야는 Supervisory control of discrete event systems, nonlinear control, analysis and synthesis of hybrid systems; Applications including: control and automation of manufacturing systems, synthesis of industrial control networks, supervisory control of semiconductor manufacturing systems, congestion control of communication networks, traffic management in IVHS, etc.