

Controller Area Network 의 실시간 서비스 품질 향상을 위한 동적 ID 할당 알고리즘 개발

Development of Dynamic ID Allocation Algorithm for Real-time Quality-of-Service of Controller Area Network

이석¹, 하경남¹, 이경창^{2,✉}Suk Lee¹, Kyoung Nam Ha¹ and Kyung Chang Lee^{2,✉}¹ 부산대학교 기계공학부 (School of Mechanical Engineering, Pusan National Univ.)² 부경대학교 제어자동화공학과 (Department of Control and Automation Engineering, Pukyong National Univ.)

✉ Corresponding author: gclee@pknu.ac.kr, Tel: 051-629-6332

Manuscript received: 2009. 4. 3 / Accepted: 2009. 6. 22

Recently, CAN (Controller Area Network) is widely used as an in-vehicle networking protocol for intelligent vehicle. The identifier field (ID) of CAN is used not only to differentiate the messages but also to give different priorities to access the bus. This paper presents a dynamic ID allocation algorithm in order to enhance the real-time quality-of-service (QoS) performance. When the network traffic is increased, this algorithm can allocate a network resource to lower priority message without degradation of the real-time QoS performance of higher priority message. In order to demonstrate the algorithm's feasibility, message transmission delays have been measured with and without the algorithm on an experimental network test bed

Key Words: Intelligent Vehicle (지능형 차량), In-vehicle Network (차량 내부 네트워크), Controller Area Network (CAN, 캔), Dynamic Identifier Allocation (동적 ID 할당), Real-time Quality-of-Service (실시간 서비스 품질)

1. 서론

최근 들어, 승용차나 트럭, 산업용 자동차, 군사용 자동차 분야에서 지능형 자동차(intelligent vehicle)에 대한 관심이 급속히 증대되고 있다. 일반적으로, 지능형 자동차가 구현되기 위해서는 차량의 위치나 속도와 같은 차량의 상태와 차량 외부의 환경이 실시간으로 인식될 수 있어야 한다. 또한, 자동차가 내부 및 외부의 정보를 바탕으로 반자동 또는 자동으로 제어될 수 있어야 한다. 이러한 이유로, 지능형 자동차는 운전자의 편의를 위하여 차량 일부를 제어하는 운전자 보조 시스템(driver assistance system)이나 운전자에게 정보를 제

공하고 위험 상황을 경고하는 충돌 경고 시스템(collision warning system) 등과 같은 다양한 지능형 센싱 및 제어 알고리즘이 요구되고 있다.^{1,2}

지능형 자동차 기술이 더욱 높은 수준으로 발전됨에 따라, 자동차에서 사용되는 가속도 센서, 온도 센서, 레이더 센서 및 각종 제어용 모터 등과 같은 전자 부품의 수가 급속도로 증가하고 있다. 특히, 자동차를 제작하는데 드는 비용 중에서 전자 부품이 차지하는 비중이 23% 이상으로 증가되고 있으며, 앞으로 그 비중은 더 증가할 것으로 예상되고 있다.³ 그러나, 센서나 액츄에이터, 스위치를 전선을 이용하여 전자 제어기(Electronic Control Unit, ECU)와 점대점(point-to-point) 방식으로

연결하는 전통적인 배선 시스템(harness system)은 전선이 기하급수적으로 증대되는 결과를 초래한다. 이러한 전선의 증가는 배선 시스템을 더욱 복잡하게 함으로써 차량의 정비와 새로운 기능의 추가를 어렵게 할 뿐만 아니라, 차량 중량을 증가시켜 차량 성능을 악화시키는 결과를 야기한다.⁴

이러한 문제를 해결하기 위하여, 자동차 회사가 중심이 되어 ECU 와 센서, 액추에이터, 스위치를 한 가닥의 공유된 전선으로 연결하려는 차량 내부 네트워크 시스템(in-vehicle networking system, IVN)이 개발되었다. 현재, 윈도우 모터(window motor)나 도어 락 스위치(door lock switch) 등과 같은 바디 네트워크(body network)용 프로토콜로 Controller Area Network(CAN)이나 J1850, Local Interconnect Network(LIN) 등이 차량에 적용되고 있다. 또한, 브레이크나 조향 시스템과 같은 실시간을 요구하는 샤시 네트워크(chassis network)용 프로토콜로 Time Triggered Protocol/ Class C(TTP/C), FlexRay, Time Triggered CAN(TTCAN) 등이 개발되고 있다. 뿐만 아니라, 운전자에게 서비스 및 오락 정보를 제공하기 위해 대용량의 정보의 공유할 있는 Media Oriented Systems Transport(MOST)나 IDB 1394 와 같은 멀티미디어 네트워크용 프로토콜도 개발되고 있는 추세이다.⁴

특히, CAN 프로토콜은 다양한 에러 감지 메커니즘과 재전송 능력을 가지고 있기 때문에, 전송되는 데이터의 높은 안정성과 사용 가능성이 보장된다.⁵ 또한, CAN 은 차량의 바디 네트워크 시스템이나 샤시 네트워크 시스템에서, 여러 개의 노드가 주기적으로 메시지를 보내더라도 버스 상에서 메시지 충돌이 효과적으로 중재될 수가 있다고 알려져 있다. 이러한 CAN 의 특징은 IVN 에서 요구하는 데이터 특성을 비교적 잘 만족하기 때문에, 현재 전세계적으로 바디 네트워크 시스템뿐만 아니라 샤시 네트워크 시스템에서도 많은 적용이 이루어지고 있다.

그러나, CAN 네트워크 시스템에서 버스 트래픽이 급격하게 증가하게 되면, CAN 의 특성에 의하여 우선 순위가 높은 식별자(identifier, ID)를 가지는 메시지들이 전송 권한을 모두 점유하게 된다. 따라서, 상대적으로 우선 순위가 낮은 메시지들은 전송 기회를 잃게 됨으로써 원하는 시간 내에 데이터를 전송할 수 없게 될 뿐만 아니라, 시스템에 큰 위험을 초래할 수도 있다.⁶

이러한 문제를 해결하기 위하여, 본 논문에서

는 주기성 메시지가 발생하는 CAN 네트워크 시스템에서 버스 트래픽이 증가할 때 우선 순위가 높은 메시지의 실시간 QoS 성능(real-time quality-of-service performance)을 해치지 않는 범위 내에서 우선 순위가 상대적으로 낮은 메시지에게도 최소한의 전송 기회를 부여할 수 있는 동적 ID 할당 알고리즘(dynamic ID allocation algorithm)을 제안한다.⁷ 또한, 본 논문에서는 제안하는 알고리즘의 성능을 평가하기 위하여, 임베디드 시스템 기반의 테스트베드를 구성하고, 그 성능을 평가한다.

본 논문은 총 4 장으로 구성되어 있다. 2 장에서는 CAN 을 위한 동적 ID 할당 알고리즘에 대해서 설명하고, 3 장에서는 동적 ID 할당 알고리즘의 성능을 평가하기 위한 임베디드 시스템 기반의 테스트베드를 설명하고, 버스 트래픽의 변화에 따른 동적 ID 할당 알고리즘의 성능 평가 결과를 보여준다. 마지막으로 4 장에서 요약과 결론을 제시한다.

2. 동적 ID 할당 알고리즘의 구조

CAN 은 메시지의 내용에 따라 11 비트 또는 29 비트로 이루어진 ID 를 부여하여 모든 메시지를 구별하고, 메시지의 전송 우선권을 결정하는 내용 기반 주소 지정 방식(content-based addressing)을 사용하고 있다. 즉, 만약 어떤 스테이션이 메시지를 전송하기 시작하면, 나머지 스테이션들은 수신된 데이터가 자신이 가진 ID 와 관련이 있는 지를 비교한다. 만약, 자신이 가진 ID 와 관련이 있으면 받아들이고, 관련이 없으면 무시한다. 특히, 두 개 이상의 스테이션이 거의 동시에 전송을 시도했을 경우에는 충돌이 발생할 수 있다. 이러한 상황에서, CAN 은 충돌된 메시지들의 ID 를 한 비트씩 비교하여 가장 낮은 ID 값을 가진 메시지(우선 순위가 가장 높음)만 전송을 계속하고, 나머지 메시지는 즉시 전송을 중단하는 CSMA/NBA(carrier Sense Multiple Access with Non-destructive Bitwise Arbitration)을 사용한다. 이로 인하여, CAN 네트워크에서 우선 순위가 낮은 ID 를 가지는 메시지들은 전송 기회를 잃게 되고 원하는 시간 내에 전송되지 못함에 따라, 메시지의 실시간 QoS 성능을 저하시키게 된다. 여기에서, 실시간 QoS 성능은 메시지가 사용자가 정의한 허용 전송 지연(allowable transmission delay)내에 전송되면 만족되는 것으로 정의한다.

본 논문에서는 우선 순위가 높은 메시지의 실시간 QoS 성능을 해치지 않는 범위 내에서, 우선 순위가 상대적으로 낮은 메시지에게도 전송 기회를 부여함으로써 실시간 QoS 성능을 향상시킬 수 있는 동적 ID 할당 기법을 제시한다. 즉, 상대적으로 우선 순위가 높은 메시지가 허용 전송 지연 이내에서 전송이 이루어지고 있을 때, 우선 순위가 낮은 메시지의 ID 를 실시간으로 조정함으로써 우선 순위가 낮은 메시지도 전송될 수 있는 기회를 부여하여 QoS 성능이 향상될 수 있도록 할 것이다.

그림 1 은 동적 ID 할당 알고리즘을 가진 CAN 노드의 구조를 나타내고 있다. 그림에서, CAN 노드는 트랜시버(transceiver)와 메시지 버퍼(message buffer), 지연 측정기(delay measurer), ID 재할당기(ID reassignor), 응용 프로그램(application program)로 구성되어 있다. 먼저, 지연 측정기는 응용 프로그램에서 만들어진 메시지의 전송 지연을 측정한다. 여기에서, 전송 지연은 응용 프로그램에서 메시지가 생성되어 전송을 위하여 메시지 버퍼에 저장하라는 기능을 가진 송신 원시함수(Tx primitive)를 호출하고 난 후부터, 해당 메시지가 충돌없이 정상적으로 전송이 성공적으로 이루어졌음을 응용 프로그램에 알리는 기능을 가진 수신 원시함수(Rx primitive)를 받았을 때까지의 시간으로 정의한다. 그리고 나서, 지연 측정기는 사용자에게 의하여 설정된 시간 간격마다 측정된 각 메시지들의 전송 지연의 평균 값을 계산하여 ID 재할당기에 제공한다. 특히, 지연 측정기는 ID 별로 평균 전송 지연을 계산하여 ID 재할당기로 제공한다. ID 재할당기

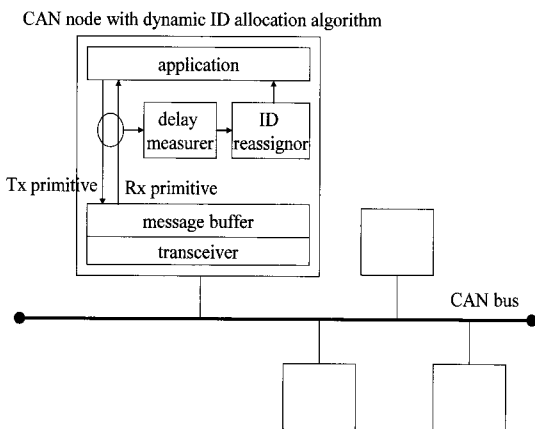


Fig. 1 Structure of dynamic ID allocation algorithm

는 계산된 각 메시지들의 평균 전송 지연을 이용하여 동적 ID 할당 알고리즘에 따라 해당 메시지의 ID 를 재할당한다.

그림 2 는 동적 ID 할당 알고리즘을 위한 CAN 의 데이터 프레임 구조를 나타내고 있다. 본 논문에서는 그림과 같이 29 비트의 ID 필드(16비트)를 3 개의 서브 필드(sub field)로 나누었다. 먼저, 메시지 타입 필드(message type field)는 해당 메시지가 하드 리얼타임(hard real-time) 메시지인지, 소프트 리얼타임(soft real-time) 메시지인지를 구분한다. 일반적으로, 허용 전송 지연 내에 반드시 전송되어야 하는 메시지들을 하드 리얼타임 메시지라 정의하며, 허용 전송 지연 범위 내에서 약간의 전송 지연을 허용하는 메시지들을 소프트 리얼타임 메시지라고 정의한다.⁸ 본 논문에서는 하드 리얼타임 메시지는 소프트 리얼타임 메시지보다 높은 메시지 타입 ID 를 할당함으로써, 소프트 리얼타임에 의하여 하드 리얼타임 메시지의 실시간 QoS 성능이 저하되지 않도록 하였다. 특히, 본 논문에서는 하드 리얼타임 메시지의 ID 는 조정하지 않음으로써, 하드 리얼타임의 QoS 성능이 악화되지 않도록 하였다.

다음으로, ID 변경 필드(ID change field)는 동적 ID 할당 알고리즘에 따라 변경되는 ID 필드이다. 동일한 메시지 타입 필드를 가진 메시지들의 우선 순위는 동일한 메시지 타입 그룹 내에서 ID 변경 필드의 값 변화에 따라 우선 순위가 조정된다. 마지막으로, 고유 필드(unique field)는 각 메시지가 가지는 고유의 ID 이다. 고유 필드는 동적 ID 할당 알고리즘의 동작 중에 ID 변경 필드의 값이 같아 지더라도 구분이 가능하도록 중복되지 않는 ID 를 가져야 한다. 또한, 스테이션들은 메시지의 고유 필드를 판별하고, 자신과 관련있는 메시지들을 수신할 수 있어야 한다.

그림 3 은 동적 ID 할당 알고리즘의 의사 코드를 나타내고 있다. 그림에서, T_{QoS} 는 실시간 QoS 성능을 만족시키도록 네트워크 설계자에 의하여

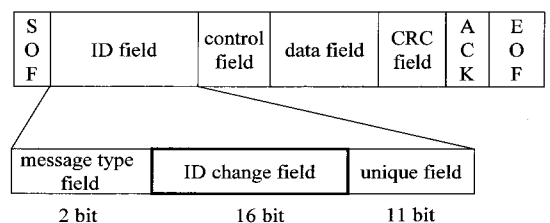


Fig. 2 Data frame for dynamic ID allocation algorithm

설정된 시간이다. 즉, 메시지가 T_{QoS} 이내에 전송되면 실시간 QoS 성능이 만족되었다고 본다. 일반적으로, IVN 시스템에서 메시지는 생성 주기의 $\frac{1}{2}$ 이내에 전송되는 경우 실시간 QoS가 만족되었다고 본다. 이에 따라, 본 논문에서는 T_{QoS} 를 메시지 생성 주기 T_{period} 의 $\frac{1}{2}$ 로 설정하였다.

본 논문에서 제안하는 동적 ID 할당 알고리즘에서는 평균 전송 지연 T 가 $T_{QoS} \pm C_j$ 이내에 있는 경우 실시간 QoS 성능을 만족한다고 간주한다. 이에 따라, ID 변경 필드의 값을 변경하지 않고 그대로 유지하도록 설정한다. 여기에서, C_j 는 이론적인 최대 전송지연 값으로서, 다음과 같이 정의한다.⁹

$$C_j = (67 + 8d_j + \frac{55 + 8d_j}{5})\tau_{bit} \quad (1)$$

식에서, d_j 는 데이터 필드의 바이트 수이며, τ_{bit} 는 1 비트를 전송하는데 걸리는 시간으로 정의한다.

다음으로, 평균 전송 지연 T 가 $T_{QoS} - C_j$ 보다 작거나 $T_{QoS} + C_j$ 보다 큰 경우, 해당 메시지의 ID 변경 필드를 1 증가시키거나 1 감소시킨다. 즉, 해당 메시지의 평균 전송 지연 T 가 T_{QoS} 보다 작은 경우에는 메시지의 우선 순위를 낮추어 줌으로써(즉, ID의 값은 증가시켜) 상대적으로 전송 기회를 줄여 준다. 이러한 결과에 의하여, 평균 전송 지연이 낮은 메시지들의 전송을 약간 늦추고, 평균 전송 지연이 큰 메시지들의 전송이 먼저 이루어지도록

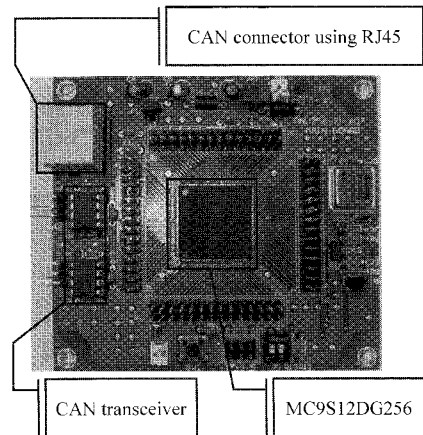
한다.

만약, 평균 전송 지연 T 가 T_{lowest} 보다 작거나 T_{upper} 보다 큰 경우에는 해당 메시지의 ID 변경 필드의 값을 급속하게 변경시켜 평균 전송 지연이 T_{QoS} 근처로 이동하도록 만들 필요가 있다. 즉, 평균 전송 지연이 매우 높은 경우에는 메시지의 우선 순위를 급하게 낮추어 평균 전송 지연이 낮아지도록 유도한다. 또한, 평균 전송 지연이 매우 낮은 경우에는 메시지의 우선 순위를 급하게 높여 다른 메시지들이 전송될 수 있도록 도와주도록 한다. 여기에서, T_{lowest} 는 전송 지연의 하한(the lowest limit), T_{upper} 는 전송 지연의 상한(the upper limit)으로서, 다음과 같이 정의하였다.

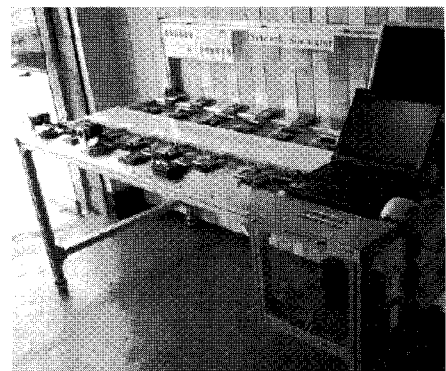
$$\begin{aligned} T_{lowest} &= T_{QoS} - 0.1 \cdot T_{Period} \\ T_{upper} &= T_{QoS} + 0.1 \cdot T_{Period} \end{aligned} \quad (2)$$

$$\begin{aligned} & \text{If } T < T_{lowest} \\ & \quad \text{Then } \Delta ID = + \left\lceil \frac{T_{QoS} - T_{lowest}}{C_j} \right\rceil \\ & \text{If } T_{lowest} \leq T < T_{QoS} - C_j \\ & \quad \text{Then } \Delta ID = +1 \\ & \text{If } T_{QoS} - C_j \leq T \leq T_{QoS} + C_j \\ & \quad \text{Then } \Delta ID = 0 \\ & \text{If } T_{QoS} + C_j < T \leq T_{upper} \\ & \quad \text{Then } \Delta ID = -1 \\ & \text{If } T_{upper} < T \\ & \quad \text{Then } \Delta ID = - \left\lceil \frac{T_{upper} - T_{QoS}}{C_j} \right\rceil \end{aligned}$$

Fig. 3 Pseudo code for dynamic ID allocation algorithm



(a) CAN communication module



(b) experimental testbed for dynamic ID allocation algorithm

3. 동적 ID 할당 알고리즘의 성능 평가

본 절에서는 동적 ID 할당 알고리즘의 성능을 평가하기 위하여 그림 4 와 같은 실험 모델을 구성하였다. 그림 4(a)는 동적 ID 할당 알고리즘을 위한 CAN 통신 노드를 나타내고 있다. 통신 노드의 마이크로컨트롤러로는 CAN 프로토콜 컨트롤러를 내장한 MC9S12DG256 을 사용하였다.¹⁰ MC9S12DG256 는 프리스케일(Freescale)사에서 생산된 마이크로컨트롤러로서, 매우 많은 자동차의 ECU 용 프로세서로 사용되고 있다. 또한, CAN 트랜시버로는 필립스의 PCA82C250 을 사용하였다. 그림 4(b)는 동적 ID 할당 알고리즘의 성능 평가를 위한 실험 모델로서, 11 개의 CAN 노드가 설치되어 있다. 또한, CAN 노드는 마이크로컨트롤러에서 제공하는 CAN Tx 인터럽트(메시지가 생성되어 CAN 프로토콜 컨트롤러에게 송신 명령을 내릴 때 발생하는 인터럽트)와 Rx 인터럽트(전송 버퍼에서 충돌없이 메시지 전송이 종료될 때 발생하는 인터럽트) 간의 시간을 측정함으로써 전송 지연을 측정하였다. 또한, CAN 네트워크의 통신 속도는 500Kbps 로 설정하였다.

표 1 은 각 CAN 노드에서 생성되는 메시지의 ID 를 나타내고 있다. 본 실험 모델에서는 각 CAN 노드에서 3 가지 그룹의 메시지가 10msec 주기로 생성되도록 설정하였다. 첫번째 ID 그룹은 하드 리얼타임 메시지로 간주하였으며, 나머지 두 개의 ID 그룹은 소프트 리얼타임 메시지로 간주하였다. 특히, 본 논문에서는 하드 리얼타임 메시지의 ID 는 고정하고, 소프트 리얼타임 메시지의 ID 만 조

정할 수 있도록 동적 ID 할당 알고리즘을 프로그램하였다. 이를 위하여 하드 리얼타임 메시지의 메시지 타입 필드는 01 로, 소프트 리얼타임 메시지의 메시지 타입 필드는 10 으로 설정하여, 하드 리얼타임 메시지의 우선 순위가 소프트 리얼타임 메시지보다 높도록 설정하였다. 이러한 방법에 따라, 소프트 리얼타임 메시지들이 하드 리얼타임 메시지의 전송에 영향을 미치지 못하도록 하여, 하드 리얼타임 메시지가 우선적으로 전송될 수 있도록 하였다.

마지막으로, 동적 ID 할당 알고리즘을 위하여, T_{QoS} 는 메시지의 생성 주기 10msec 의 1/2인 5msec 로 설정하였으며, T_{lowest} 는 식 (2)에 따라 4msec 로, T_{upper} 는 6msec 로 설정하였다. 그리고, C_j 는 500Kbps 에서 8 바이트의 데이터가 전송된다고 가정하고, 식 (1)을 계산하여 300 μ sec 로 근사화시켜 사용하였다. 또한, ID 를 변경하는 주기를 300msec 로 설정하고, 각 노드에서 300msec 주기로 평균 전송 지연이 계산되도록 설정하였다. 따라서, 동적 ID 할당 알고리즘을 탑재한 CAN 노드들은 300msec 주기로 ID 변경 필드를 평균 전송 지연 (average transmission delay) 값에 따라 3, 1, 0, -1, -3 씩 변화되도록 하였다.

그림 5 는 표 1 에 정의되어 있는 ID 를 가지는 메시지들을 이용한 동적 ID 할당 알고리즘의 성능 평가 결과를 나타내고 있다. 그림에서, 노드가 2 개인 경우에는 1 번과 11 번 노드만을 CAN 네트워크에 참여시켜 통신이 이루어지도록 하였으며, 300msec 주기로 평균 전송 지연을 계산하고 ID 할당을 수행하였다. 또한, 본 논문에서는 트래픽의 변화에 따른 동적 ID 할당 알고리즘의 성능을 평가하기 위하여, 2 번 노드부터 차례대로 네트워크에 추가시켜 가며 평균 전송 지연을 측정하였다.

그림 5(a)는 메시지 그룹 1 인 하드 리얼타임 메시지의 평균 전송 지연을 나타내고 있다. 본 실험에서는 하드 리얼타임 메시지의 ID 는 변경하지 않았다. 따라서, 가장 우선순위가 높은 ID 60 의 메시지는 거의 일정한 값을 가지며, ID 660 의 메시지는 트래픽이 증가함에 따라 거의 선형으로 평균 전송 지연이 증가됨을 확인하였다.

그림 5(b)와 5(c)는 메시지 그룹 2 와 3 인 소프트 리얼타임메시지의 평균 전송 지연을 나타내고 있다. 그림에서, 고정 ID 를 사용하는 경우에는 트래픽이 증가함에 따라 각 ID 의 평균 전송 지연은 거의 선형으로 증가함을 관찰할 수 있었다. 특히,

Table 1 Message ID for performance evaluation

| node | 1 st ID group | 2 nd ID group | 3 rd ID group |
|------|--------------------------|--------------------------|--------------------------|
| 1 | 60 | 720 | 1380 |
| 2 | 120 | 780 | 1440 |
| 3 | 180 | 840 | 1500 |
| 4 | 240 | 900 | 1560 |
| 5 | 300 | 960 | 1620 |
| 6 | 360 | 1020 | 1680 |
| 7 | 420 | 1080 | 1740 |
| 8 | 480 | 1140 | 1800 |
| 9 | 540 | 1200 | 1860 |
| 10 | 600 | 1260 | 1920 |
| 11 | 660 | 1320 | 1980 |

동일 그룹 내에서 우선 순위가 높은 메시지의 기율기가 우선 순위가 낮은 메시지의 기율기보다 낮게 나타나며, 트래픽이 증가됨에 따라 평균 전송 지연의 차이가 커짐을 확인할 수 있었다.

반면, 동적 ID 할당 알고리즘을 사용하는 경우, 트래픽이 증가됨에 따라 평균 전송 지연이 증가되는 점은 비슷하지만, 두 개 ID 의 평균 전송 지연의 기율기가 비슷해지려는 경향을 보인다. 특히, 동적 ID 할당 알고리즘을 적용하는 경우, 고유 필드의 ID 가 상대적으로 작은 메시지(즉, 우선 순위가 높은 메시지)도 고유 필드의 ID 가 상대적으로

큰 메시지(즉, 우선 순위가 높은 메시지)보다 평균 전송 지연이 크게 나타나는 결과도 확인할 수 있었다. 이는 ID 변경 필드의 값이 변화됨에 따라, 우선 순위가 바뀌기 때문인 것으로 판단된다.

4. 요약 및 결론

본 논문에서는 차량용 네트워크 프로토콜로 널리 사용되는 CAN 네트워크에서, 전송 방식에 의하여 우선 순위가 낮은 메시지들의 전송 지연이 증가되는 문제를 개선하기 위하여 전송 지연의 변화에 따라 ID 를 동적으로 변화시킬 수 있는 동적 ID 할당 알고리즘을 제안하였다. 특히, 본 논문에서는 하드 리얼타임 메시지의 실시간 QoS 성능을 해치지 않는 범위 내에서, 우선 순위가 상대적으로 낮은 메시지에게도 최소한의 전송 기회를 부여할 수 있도록 알고리즘을 개발하였다. 그리고, 본 논문에서는 동적 ID 할당 알고리즘을 탑재한 CAN 노드들을 제작하고, 다수의 CAN 노드들로 구성된 실험 모델을 제작하였다. 이러한 실험 모델을 이용하여 제안된 동적 ID 할당 알고리즘의 성능을 평가하였다.

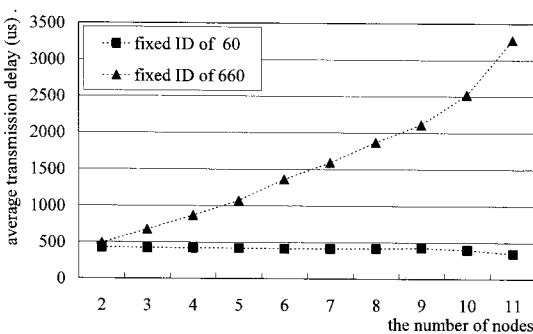
본 논문에서 제안된 동적 ID 할당 알고리즘을 이용하여 실시간으로 ID 를 변경함으로써, 상대적으로 전송 지연에 여유가 있는 메시지들의 평균 전송 지연을 상승시켜 상대적으로 전송 지연이 높은 메시지들의 전송 지연을 감소시킬 수 있게 되었다.

그러나, 본 논문에서는 적용 대상을 다수의 주기 메시지가 발생하는 시스템으로 한정하여 평가를 수행하였으며, ID 의 증감을 위한 값으로 네트워크의 트래픽에 따른 경험적인 값을 이용하였다. 따라서, 동적 ID 할당 알고리즘의 보다 실제적인 응용을 위해서는 주기 메시지뿐만 아니라, 비주기 메시지에 대한 고려도 필요하며, 네트워크 상태에 따라 최적의 ID 증감량을 결정하기 위한 추가적인 연구가 이루어져야 한다.

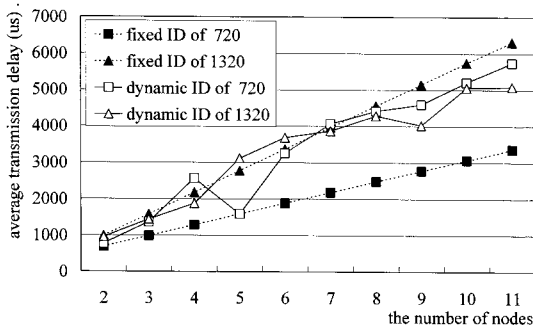
후 기

이 논문은 2007 년도 정부재원(교육인적자원부 학술연구조성사업비)으로 한국학술진흥재단의 지원을 받아 연구되었음(KRF-2007-314-D00022).

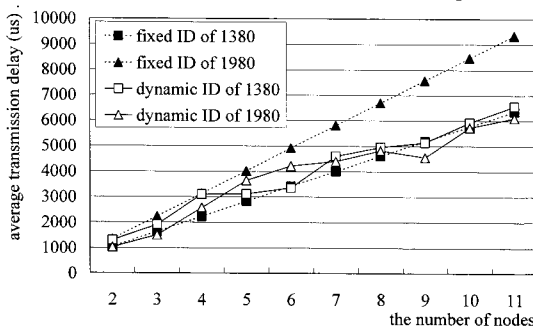
참고문헌



(a) average transmission delay of 1st ID group



(b) average transmission delay of 2nd ID group



(c) average transmission delay of 3rd ID group

Fig. 5 Average transmission delay of 3rd ID group

1. Leen, G. and Hefferman, D., "Digital Networks in the Automotive Vehicle," IEEE Computer and Control Engineering Journal, Vol. 10, No. 6, pp. 257-266, 1999.
2. Kim, B. K., Han, S. C. and Lee, H. J., "A Study on Rear Lateral Collision Warning System of Vehicle using Fuzzy Control Algorithms," Journal of the KSPE, Vol. 24, No. 2, pp. 73-85, 2007.
3. Leen, G. and Hefferman, D., "Expanding Automotive Electronic Systems," IEEE Computer, Vol. 35, No. 1, pp. 88-93, 2002.
4. Lee, S., Kim, M. H. and Lee, K. C., "Survey on in-vehicle network system researches," Journal of the KSPE, Vol. 23, No. 9, pp. 7-14, 2006.
5. Bosch, "CAN Specification Version 2.0," Bosch, 1991.
6. Kopetz, H., "Why Time-Triggered Architectures will Succeed in Large Hard Real-Time Systems," Proceedings of the 5th IEEE Workshop on Future Trends of Distributed Computing System, pp. 2-9, 1995.
7. Grenier, M. and Navet, N., "Fine-Tuning MAC-Level Protocols for Optimized Real-Time QoS," IEEE Transactions on Industrial Informatics, Vol. 4, No. 1, pp. 6-15, 2008.
8. Livani, M. A., Kaiser, J. and Jia, W., "Scheduling Hard and Soft Real-time Communication in a Controller Area Network," Control Engineering Practice, Vol. 7, No. 12, pp. 1515-1523, 1999.
9. Tindell, K. and Burns, A., "Guaranteed Message Latencies for Distributed Safety-Critical Hard Real Time Control Networks," Technical Report YCS 94-229, pp. 1-17, 2000.
10. Kim, D. S. and Choi, D. H., "Maximum Allowable Delay Bounds and Real-time Scheduling Method of Networked Discrete-time Control Systems," Journal of Control, Automation and Systems Engineering, Vol. 12, No. 7, pp. 719-727, 2006.