

대규모 분산 시뮬레이션 환경을 위한 실시간성 지원 DDS/SDN 통합 아키텍처

DDS/SDN integration architecture with real-time support for large-scale distributed simulation environments

김 다 올 *, 조 인 휘 *, 김 원 태**★

Daol Kim*, Inwhoo Joe*, Wontae Kim**★

Abstract

Recently, as the development system has become larger, sequential simulation methods have become impossible to verify systems that take a long time or require real time results. Therefore, a study of a distributed simulation system that simulates several processes has been conducted. In order to simulate real-time systems, efficient data exchange between distributed systems is required. Data Distribution Service is a data-oriented communication middleware proposed by Object Management Group and provides efficient data exchange and various QoS. However, in a large-scale distributed simulation environment distributed over a wide area, there is a problem of Participant Discovery and QoS guarantee due to domain separation in data exchange. Therefore, in this paper, we propose a DDS/SDN architecture that can guaranteed QoS and effective Participant Discovery in an SDN-based network.

요 약

최근 개발 시스템의 대형화로 순차적 시뮬레이션 방법으로는 오랜 시간이 소요되거나 실시간으로 결과가 요구되는 시스템 검증이 불가능해 졌다. 때문에 여러 프로세스로 나누어 시뮬레이션하는 분산 시뮬레이션 시스템의 연구가 진행되었다. 그리고 실시간 시스템의 시뮬레이션을 위해 분산된 시스템들 간의 효율적인 데이터 교환이 필요하다. Data Distribution Service는 Object Management Group에서 제안한 데이터 중심의 통신 미들웨어로 효율적인 데이터 교환과 다양한 QoS를 제공한다. 그러나 광역에 분포되어 있는 대규모 분산 시뮬레이션 환경에서는 데이터 교환 시 도메인 분리로 참여자 탐색 문제와 QoS 보장의 어려움을 갖는다. 따라서, 본 논문에서는 SDN 기반의 네트워크에서 QoS보장과 효과적인 참여자 탐색이 가능한 DDS/SDN 아키텍처를 제안한다.

Key words : Real-time simulation, Distribution simulation, DDS, SDN, QoS

* Dept. of Computer Software, Hanyang University

** Dept. of Computer Science and Engineering, Koreatech University

★ Corresponding author

E-mail : wtkim@koreatech.ac.kr, Tel:+82-41-560-1485

※ Acknowledgment

This paper was supported by the National Research Foundation of Korea(No. 2017R1A2B4010875).

Manuscript received Mar. 10, 2018; revised Mar. 27, 2018 ; accepted Mar. 28, 2018

This is an Open-Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

I. 서론

분산 시뮬레이션이란 통합된 시뮬레이션을 여러 프로세스로 나누어 시뮬레이션 하는 것을 말한다 [1]. 최근 개발되는 다양한 시스템들은 기존 시스템들의 비해 대형화되어 순차적 시뮬레이션 방법으로는 검증이 불가능하거나 결과 도출까지 오랜 시간이 소요되는 경우가 증가하고 있다. 최신 개발 시스템 검증을 위해 여러 프로세스에 시뮬레이션을 나누어 실행하는 분산 시뮬레이션 시스템의 적용 및 통합 시뮬레이션 환경을 만드는 연구가 진행되었다 [2][3]. 특히, 분산 시뮬레이션에서 실시간 시스템 검증 시 실시간성, 신뢰성 등을 위해 시뮬레이터간의 효율적 데이터 교환이 필요하다. OMG(Object Management Group)에서 제안한 DDS(Data Distribution Service)는 통신 미들웨어로 효율적 데이터 교환을 위한 메커니즘과 다양한 QoS를 제공 한다 [4][5].

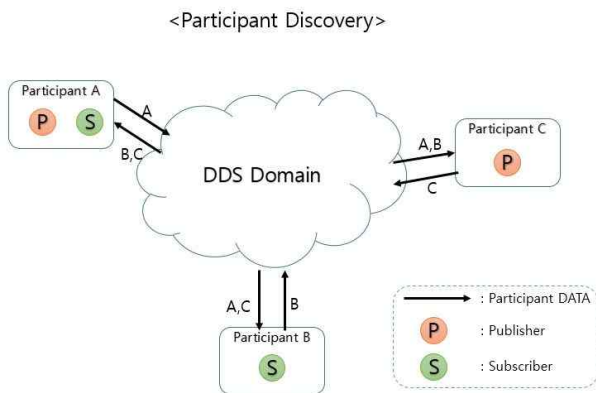


Fig 1. Participant Discovery Process
그림 1. 참여자 탐색 과정

DDS는 데이터 전송 이전에 탐색(Discovery) 과정을 거친다. 탐색 과정은 도메인(Domain)에 포함된 참여자(Participant)와 종단점(Endpoint) 정보를 파악하기 위한 참여자 탐색(Participant Discovery) 과정과 종단점 탐색(Endpoint Discovery) 과정으로 구성된다. 이때 참여자 탐색 과정은 같은 도메인 내에 포함된 참여자만 발견 가능하며 도메인이 분리되는 경우 참여자 발견이 불가능한 문제점이 있다. 데이터 전송 이전에 발생하는 이러한 문제를 해결하기 위해 OMG는 DDS 도메인간의 연결을 위해 Bridging Mechanism을 제공 한다 [6]. 하지만 이 방식은

동일한 네트워크 스위치에 연결된 도메인에서만 효과를 발휘한다. 멀리 떨어진 도메인 사이의 네트워크는 고려되지 않는다.

DDS의 탐색과정 완료 후에는 데이터(Data) 전송이 이루어진다. DDS에서 제공하는 QoS는 미들웨어의 특성상 응용 수준(Application level)에서만 고려되기 때문에 다양한 어플리케이션이 요구하는 QoS를 보장하지 못한다. L Bertaux[7]는 분산된 실시간 어플리케이션과 가변적인 요구 사항 충족을 위해 DDS와 SDN의 결합을 제안하였다. DDS QoS를 캡처하고 네트워크 최적화를 위해 SDN을 활용하여 네트워크 2계층과 3계층에 적용하는 아키텍처를 제안하였다. 그러나 시스템에 대한 세부 사항, 성능평가는 제공되지 않는다. 본 논문에서는 실시간 분산 시뮬레이션 환경에서 DDS를 통해 데이터 교환 시 발생하는 원격 탐색 문제와 QoS 보장을 위한 SDN기반의 아키텍처(Architecture)를 제안한다.

본 논문의 나머지 구성은 다음과 같다. 2장에서 제안하는 아키텍처에 대해 설명하고 3장에서 아키텍처의 구현 및 성능을 분석한다. 마지막 장에서 결론을 제시한다.

II. Proposed Architecture

이 장에서는 DDS의 원격 탐색 문제 및 Deadline QoS 보장을 고려한 SDN 기반 제안 구조를 설명한다.

1. 대규모 분산 시뮬레이션 환경을 위한 실시간성 지원 DDS/SDN 아키텍처

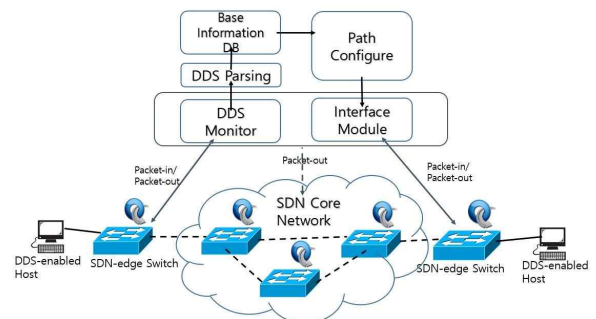


Fig 2. A SDN-based DDS Architecture
그림 2. SDN기반 DDS 아키텍처

분산 시뮬레이션 시스템에서 효율적 데이터 교환을 위해 통신 미들웨어인 DDS를 사용하는 경우, 탐색 과정에서 원격 도메인 간에 탐색 문제가 발생한다. 또한 DDS에서 제공하는 다양한 QoS 정책은 응용 계층에서만 고려되고 네트워크 수준에서는 고려되지 않는다. 실시간 분산 시뮬레이션 시스템에선 전송 데이터의 실시간성 및 신뢰성이 중요하다. 그러나, DDS QoS가 응용 계층에서만 고려 될 경우 네트워크의 변동 사항에 의해 쉽게 QoS 위반이 일어날 수 있다. DDS에서 제공하는 QoS 중 Deadline QoS는 실시간 분산 시뮬레이션 시스템에서 실시간성과 연관된 QoS에 해당한다. 이러한 사항들을 고려하여 그림 2와 같이 SDN 기반 DDS 아키텍처를 구성하고, 실시간성을 보장할 수 있도록 DDS의 Deadline QoS 보장을 위한 컴포넌트를 SDN 컨트롤러에 추가한다.

2. SDN의 제어 계층과 응용계층 아키텍처

기존 SDN에서 제공하는 컨트롤러와 어플리케이션의 기능만으론 DDS의 원격 탐색 문제 및 Deadline QoS 보장이 불가능하기 때문에 몇 가지 모듈이 추가적으로 필요하다. 각 모듈은 그림3과 같이 SDN의 제어 계층(Control layer)와 응용 계층(Application layer)에 위치한다.

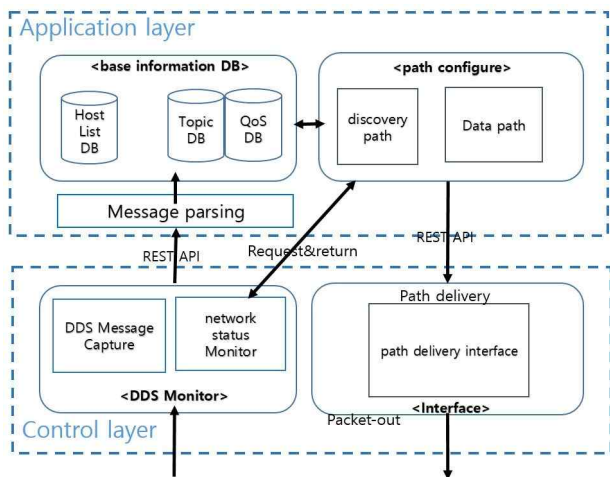


Fig 3. The DDS/SDN integration Architecture
 그림 3. DDS/SDN 통합 아키텍처

가. 제어 계층(Control layer)

SDN의 제어 계층에는 DDS Monitor 모듈과 Interface 모듈이 있다.

DDS Monitor 모듈은 컨트롤러로 전달 된 Packet-in 메시지 중 DDS 메시지를 확인하고

네트워크 상태를 파악하는 모듈이다. DDS 메시지는 SDN 스위치에서 SDN 컨트롤러로 전달 된 Packet-in 메시지 중 RTPS 프로토콜 사용 여부를 통해 확인한다. DDS 메시지에는 참여자 탐색 메시지(Participant Discovery Message), 종단점 탐색 메시지(Endpoint Discovery Message), 데이터 메시지(Data Message)가 있다. RTPS 프로토콜 사용 확인을 통하여 확인 된 DDS 메시지는 REST API를 사용해 응용 계층(Application layer)의 Message Parsing 모듈로 전달된다. DDS Monitor의 Network Status Monitor는 응용 계층의 Path Configure 모듈의 요청이 있을 때 네트워크 상태를 전달한다.

Interface 모듈은 path configure 모듈에서 전달 받은 네트워크 설정 정보를 SDN 스위치에 전달 하는 모듈이다. SDN 스위치는 전달 받은 정보에 따라 상태가 변경 된다.

나. 응용 계층(Application Layer)

응용 계층에는 Message Parsing 모듈, Base Information DB 모듈, Path Configure 모듈이 있다. 그림 4는 응용 계층 모듈들의 동작을 나타낸다.

Message Parsing 모듈은 DDS monitor로부터 전달 받은 메시지에서 필요한 정보를 획득한다.

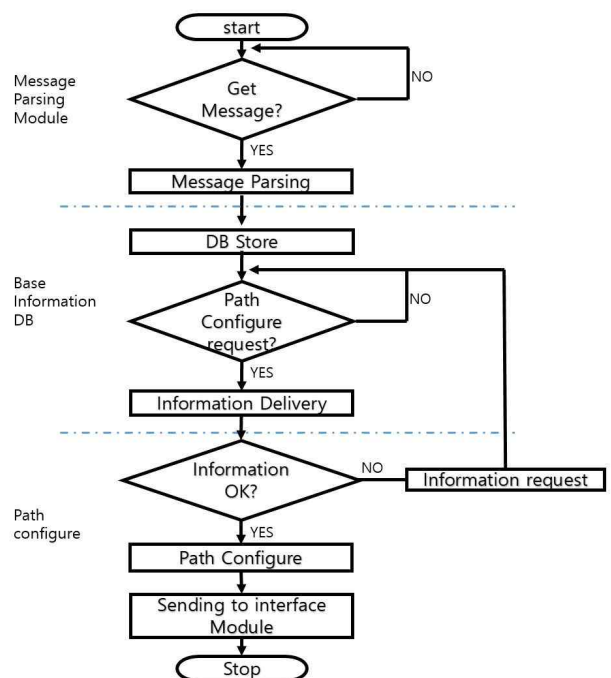


Fig 4. The application layer flowchart
 그림 4. 응용 계층 순서도

필요 정보의 종류로는 메시지가 발생한 호스트(Host)의 정보, 토픽(Topic) 정보, QoS 정보 등이 있다. 획득 된 정보는 Base Information DB 모듈로 전달된다.

Base Information DB 모듈은 Message Parsing 모듈에서 전달 받은 정보를 분류하여 저장한다. 각 DB에 저장된 정보와 비교하여 중복되는 경우 저장하지 않는다. 변경되거나 새로운 정보가 전달되었을 때, DB에 저장한다. 또한, Path Configure 모듈에서 정보를 요청할 경우 DB에서 해당 정보를 전달한다.

Path Configure 모듈은 QoS 보장을 위한 데이터 경로와 원격 탐색 문제 해결을 위한 경로를 계산한다. 경로 계산 이전에 Base Information DB와 DDS Monitor 모듈에 필요한 정보를 요청하고 SDN 컨트롤러에 포함된 컴포넌트에도 필요 정보를 요청한다. 원격 탐색 문제의 네트워크 설정을 변경하는 경우, Base Information DB에 호스트 목록(Host List) 정보를 요청하고, SDN 컨트롤러의 컴포넌트에 토폴로지 정보를 요청한다. 네트워크 토폴로지에서 참여자 탐색 메시지가 발생한 네트워크 목록과 비교하여 네트워크간의 최단 경로를 도출한다. 도출된 경로는 동일한 Vlan ID를 가지도록 네트워크 설정을 변경해 준다. 이러한 네트워크 설정을 통해 도메인이 분리되지 않도록 하여 원격 탐색 문제가 발생하는 것을 방지한다. 또한, SDN에 연결된 전체 네트워크를 하나의 논리적 네트워크로 변경하여 원격 탐색 문제를 해결하는 방법보다 전체 네트워크의 평균 부하를 낮출 수 있다. DDS QoS 보장을 위한 Data 전달 경로를 설정하는 경우에는 Base Information DB에서 토픽, QoS 정보를 요청하여 전달 받고, DDS Monitor에서 네트워크 상태 정보를 전달 받는다. 또한, SDN 컨트롤러에 포함된 컴포넌트로부터 토폴로지 정보를 요청하여 전달받아 데이터 전송 경로를 도출한다. 이렇게 도출된 네트워크 설정 정보 및 데이터 경로(Data Path) 정보는 REST API를 통해 제어 계층(Control layer)의 Interface 모듈로 전달한다.

다. 데이터 경로 설정

제안된 구조에서 모든 네트워크 설정관련

사항들은 Path Configure 모듈에서 계산한다. 참여자 탐색 과정을 위해 네트워크 설정 변경 사항을 도출하는 경우 Base Information DB 모듈에서 전달 받은 호스트 목록 간에 참여자 탐색 메시지가 전달될 수 있도록 변경한다. 참여자가 생성된 호스트와 관련된 최소의 스위치만 포함 되도록 토폴로지에서 해당 호스트가 연결된 스위치 사이의 경로를 다익스트라(Dijkstra) 알고리즘을 이용하여 계산한다.

데이터 전송 경로를 도출하는 경우 분산 시뮬레이션 요구 사항 중 종단간(End to End)의 전송 시간 제한이 중요 사항으로 간주 된다. 때문에 DDS에서 제공하는 QoS 중 Deadline QoS는 종단간의 전송 시간 제약을 규정하진 않지만 DataReader에서 Deadline QoS 설정된 경우 데이터를 받지 못하면 위반 통보를 받는다. 때문에 DDS의 QoS 중 경로설정 시 Deadline QoS를 고려한다. 종단점 탐색 이후 발간자(Publisher)와 구독자(Subscriber) 간의 데이터 전달 경로는 DDS QoS와 상관없이 네트워크 레벨에서 설정되는데 이는 네트워크 환경에 따라 다익스트라(Dijkstra) 알고리즘에 의하여 결정되거나, 임의의 알고리즘에 따라서 경로가 설정될 수 있다. 이러한 경우, 같은 호스트에서 다수의 참여자가 생성되어 데이터를 전송하면 동일한 경로에 전송 데이터의 부하가 가중될 수 있다. 전송 데이터의 부하가중은 네트워크에 큰 부담을 주며, DDS QoS를 보장하지 못하는 원인이 될 수 있기 때문에 Deadline QoS를 고려하여 데이터전달 경로를 분산시켜 준다. 본 논문에서는 Deadline QoS의 존재 여부에 따라서 경로를 분산하였다. 분산을 위한 데이터 경로 설정 알고리즘은 아래 그림5와 같다. 또한 설정된 경로는 이후 예약 방식을 통해 네트워크 상태가 변동되어도 QoS를 보장할 수 있도록 한다.

```

Select Case State
Case State == Null:
  Nodediscovery ← Host List in DB Module
  Discovery Path ← Dijkstra(Nodediscovery)
  State = 1
  Return Discovery Path
Case State != Null:
  Nodeexchange ← Nodediscovery
  Topici ← Topic List in DB Module
  QoSi ← QoS of Topici in DB Module
  While i ≤ Number of Total Topic x
    If QoSi == Null
      DataPathi ← select largest flow capacity
    Else If QoSi != Null
      Weight ← according to flow capacity, length, etc
      DataPathi ← select largest Weight
    End If
    i++
  End
End Case
    
```

Fig 5. The path configure algorithm
 그림 5. 경로 결정 알고리즘

III. 실험 및 분석

이 장에서는 제안 구조에서 발생하는 참여자 탐색 메시지의 수와 평균 딜레이 성능을 비교한다. 시뮬레이션은 우분투(Ubuntu) 12.04에서 가상 네트워크 환경을 구성한다. 가상 네트워크 환경은 미니넷(Mininet) 2.2.1 및 Open vSwitch 2.5.3을 통해 구성되었으며, SDN 컨트롤러는 RYU 컨트롤러를 사용하고 데이터 전송에는 OpenDDS 3.11를 사용한다. 구성된 가상 네트워크 환경의 화면은 아래 그림 6과 같다.

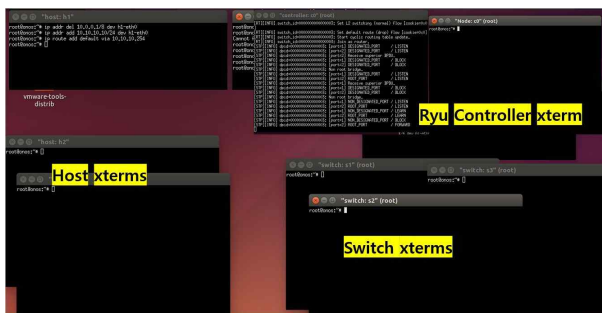


Fig 6. The simulation environment
 그림 6. 시뮬레이션 환경

분산 네트워크 환경에서 탐색 불가능한 문제 해결을 위해 SDN을 적용하고 분산 시뮬레이션 네트워크 환경을 논리적으로 통합하는 방식으로 원격 탐색 문제를 해결 할 수도 있다. 기존 DDS에서 참여자 탐색 메시지는 도메인 내에 존재하는 참여자의 수만큼 탐색 메시지가 발생한다. SDN을 적용하여 분산 시뮬레이션 네트워크 환경을 통합하면 참여자 메시지는 원격 도메인의 참여자에게 전달된다. 이때 발생하는 참여자 탐색 메시지의 수는 분산 시뮬레이션에 참여하는 스위치의 수만큼 증가하게 된다. 분산 시뮬레이션 환경에 x 개의 도메인이 존재할 때, i 번째 도메인에 속한 참여자의 수를 P_i 라 하고 스위치가 S 개 존재한다 할 때, 총 발생 메시지 수는 아래 수식 (1)과 같다.

$$\left(\sum_{i=1}^x P_i\right)(1+S) \tag{1}$$

참여자 탐색은 가능하지만 많은 탐색 메시지가 전체 네트워크에 전송되어 영향을 미칠 수 있기 때문에 제안구조에선 논리적 네트워크 설정 시 시뮬레이션의 참여하지 않는 경로는 통합 네트워크에서 제외 될 수 있도록 Vlan ID를 활용한다. Vlan ID를 할당 받지 못한 스위치의 수를 V 개라 할 때, 제안 구조에서 발생하는 메시지 수는 아래 수식 (2)와 같다.

$$\left(\sum_{i=1}^x P_i\right)(1+S-V) \tag{2}$$

분산 시뮬레이션에 영향을 미치지 않는 스위치들을 배제함으로써 전체 네트워크에 전송되는 참여자 탐색 메시지를 감소시킬 수 있다.

가상 시뮬레이션 환경에서도 참여자 탐색 메시지 수, 중단간 평균 딜레이 성능 비교를 위해 표 1과 같은 시뮬레이션 파라미터를 사용한다.

Table 1. The simulation parameters
 표 1. 시뮬레이션 파라미터

Parameters	Value
Number of Hosts	1-5
Number of Switches	1-9
Deadline QoS	500µs
Data Size	0.5-2Kbyte

아래 그림 7은 일정한 호스트 사이에서 스위치가 증가하는 경우 전체 네트워크에 전송되는 참여자 탐색 메시지 수를 비교 한다.

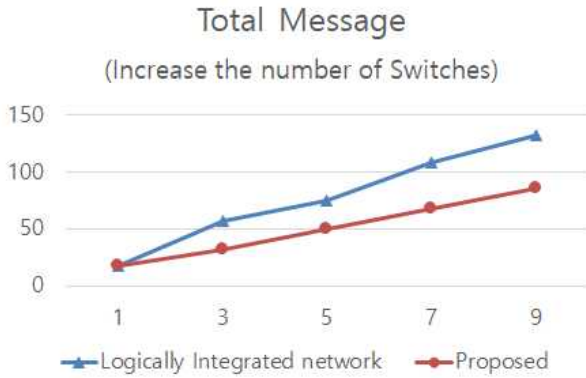


Fig 7. The number of participant discovery message when the number of switches increases

그림 7. 스위치 수가 증가할 때 참여자 탐색 메시지 수

그림 8은 참여 호스트의 수가 증가하고 스위치 수는 동일할 때 전체 네트워크에 전송되는 참여자 탐색 메시지 수를 비교한다.

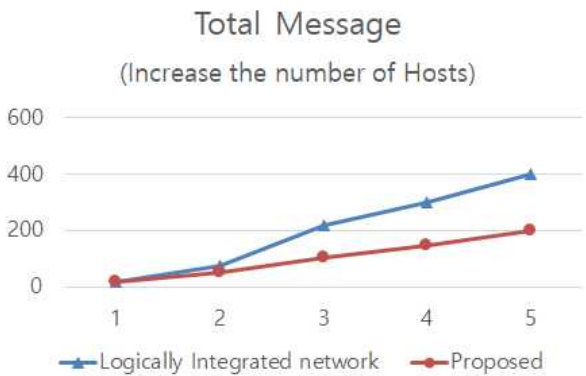


Fig 8. The number of participant discovery message when the number of hosts increases

그림 8. 호스트 수가 증가할 때 참여자 탐색 메시지 수

SDN을 통해 논리적으로 통합 된 네트워크는 제안 구조에서 시뮬레이션에 참여하는 호스트가 있는 경로만 설정해준 경로보다 많은 경로가 포함되지 때문에 더 많은 참여자 탐색 메시지가 발생하게 된다.

그림 9은 네트워크 혼잡 상황에서 전송 데이터 크기가 증가할 때 종단간의 평균 지연시간을 비

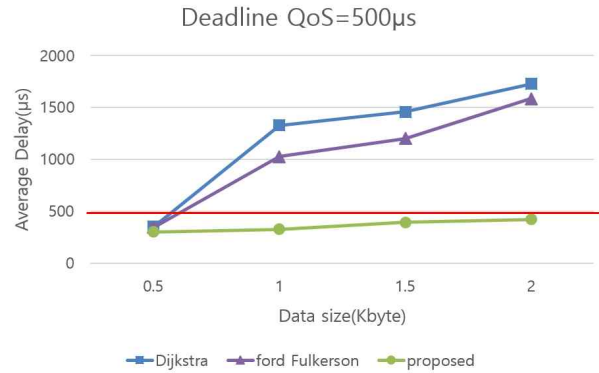


Fig 9. The average end-to-end delay

그림 9. 평균 종단간 딜레이

교한다. SDN 기반 분산 시뮬레이션 환경에서 데이터 전송 경로는 각 SDN 컨트롤러의 규칙에 따라 설정된다. 시뮬레이션에 사용 된 RYU 컨트롤러는 라우팅을 할 때 다익스트라(Dijkstra) 알고리즘을 사용하여 경로를 설정하기 때문에 제안 구조와 Dijkstra 알고리즘을 비교한다. 또한 네트워크 혼잡 상황을 고려할 땐 대역폭을 가중치로 사용하는 포드 풀커슨(Ford-Fulkerson) 알고리즘과도 비교한다. 두 알고리즘과 제안 된 구조의 종단간 평균 지연시간을 비교하였을 때 제안 구조의 종단 간 평균 지연 시간이 다익스트라 알고리즘보다 약 15%-74%, 포드 풀커슨 알고리즘 보다 약 13%-73% 낮게 측정되는 것을 볼 수 있다.

IV. 결론

본 논문에서는 분산 시뮬레이션 환경에서 DDS를 사용하여 데이터 교환이 이루어질 때, 원격 탐색 문제와 실시간성과 연관 된 Deadline QoS 지원을 위한 SDN 기반 DDS 아키텍처를 제안하였다. 제안 된 아키텍처는 원격 도메인간의 탐색 문제를 시뮬레이션에 참여하는 네트워크가 포함 된 경로만 논리적 통합하는 형태로 해결하고 전체 발생 메시지 수를 감소시킨다. 또한 DDS의 Deadline QoS 보장을 위해 SDN 컨트롤러에서 QoS값과 네트워크 상황을 고려한 Data 전달 경로를 설정한다. QoS를 고려한 분할 경로를 통해 Data가 전달되기 때문에 특정 링크의 과부하로 인한

Deadline QoS 위반율을 낮추고 자원 예약을 통해 가변적인 네트워크 상황에서도 실시간성을 지원이 가능하다.

현재 실시간성 지원을 위해 DDS에서 제공하는 QoS 중 Deadline QoS만을 고려하였으나 향후 DDS에서 제공하는 다른 QoS 고려한 알고리즘의 추가 및 실시간성을 보장할 수 있도록 하는 연구가 필요하다.

References

- [1] R.M. Fujimoto, "Distributed Simulation Systems," *Simulation conference*, 2003, pp. 124-134.
- [2] R.M. Fujimoto, *Parallel and Distributed Simulation Systems*, Wiley, 2000.
- [3] Y.J. Kim, J.H. Kim and T.G. Kim, "Heterogeneous Simulation Framework Using DEVS BUS," *SIMULATION: Transactions of The Society for Modeling and Simulation International*, vol. 79, no. 1, pp. 3-18, 2003. DOI: 10.1177/0037549703253543
- [4] Object Management Group(OMG), "Object Management Group(OMG) Data distribution Service," <http://www.omg.org/omg-dds-portal/>
- [5] Object Management Group(OMG), *Data distribution Service for Real-time Systems Specification*, Object Management Group(OMG), 2004.
- [6] J.M. Lopez-Vega, J. Povedano-Molina, G. Pardo-Castellote and J.M. Lopez-Soler, "A content-aware bridging service for publish/subscribe environments," *The Journal of System and Software*, Vol. 86, pp.108-124, 2012. DOI:10.1016/j.jss.2012.07.033
- [7] L. Bertaux, A. Hakiri, S. Medjiah, P. Berthou and S. Abdellatif, "A DDS/SDN Based communication System for Efficient Support of Dynamic Distributed Real-Time Applications," *IEEE/ACM 18th International Symposium*, 2014, pp. 77-84.

BIOGRAPHY

Daol Kim(Member)



2016.2: BS degree in e-business, Bucheon University
2016.3~:MS degree course in Computer Software, Hanyang University

Inwhee Joe(Member)



1983.2: BS degree in Electronics Engineering, Hanyang University
1995.2: MS degree in Information and Communication, Arizona University
1998.2: PhD degree in Information and Communication, Georgia Tech University
1998~2000 : Oak Ridge National Institute Researcher
2000~2002 : Bellcore Lab (Telcordia) Scientist
2002~ : Professor, Hanyang university

Won-Tae Kim (Member)



1994.2: BS degree in Electronics Engineering, Hanyang University
1996.2: MS degree in Electronics Engineering, Hanyang University
2000.8: PhD degree in Electronics Engineering, Hanyang University
2001.1~2005.2 : CTO, Rostic Technologies Inc.
2005.3~2015.8 : CPS Team Manager, ETRI
2015.9~ : Assistant professor, Koreatech university