



**(19) 대한민국특허청(KR)**  
**(12) 등록특허공보(B1)**

(45) 공고일자 2022년04월11일  
(11) 등록번호 10-2385762  
(24) 등록일자 2022년04월07일

(51) 국제특허분류(Int. Cl.)  
H04L 43/0852 (2022.01) H04B 7/26 (2006.01)  
H04L 43/16 (2022.01) H04L 47/11 (2022.01)  
H04L 47/127 (2022.01) H04L 47/25 (2022.01)  
H04L 47/283 (2022.01) H04W 28/22 (2009.01)  
H04W 52/26 (2009.01)

(52) CPC특허분류  
H04L 43/0852 (2013.01)  
H04B 7/2659 (2013.01)

(21) 출원번호 10-2021-7041007  
(22) 출원일자(국제) 2020년05월15일  
심사청구일자 2021년12월15일

(85) 번역문제출일자 2021년12월14일  
(65) 공개번호 10-2022-0007698  
(43) 공개일자 2022년01월18일  
(86) 국제출원번호 PCT/US2020/033154  
(87) 국제공개번호 WO 2020/236599  
국제공개일자 2020년11월26일

(30) 우선권주장  
16/415,609 2019년05월17일 미국(US)

(56) 선행기술조사문헌  
US20160374094 A1  
(뒷면에 계속)

(73) 특허권자  
펜산도 시스템즈 인크.  
미국 95035 캘리포니아 밀피타스 엘더 드라이브 570

(72) 발명자  
판 룡  
미국 95035 캘리포니아 밀피타스 엘더 드라이브 570  
뉴던 피터  
미국 95035 캘리포니아 밀피타스 엘더 드라이브 570

(74) 대리인  
김태홍, 김진희

전체 청구항 수 : 총 31 항

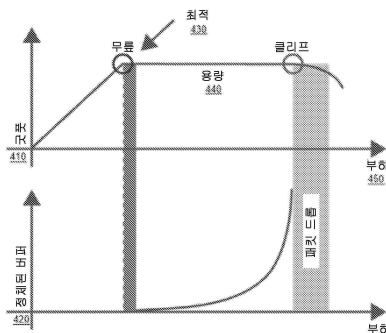
심사관 : 정남호

(54) 발명의 명칭 **속도 최적화된 정체 관리**

**(57) 요약**

RoCEv2 네트워크에서 정체를 감소시키기 위한 방법 및 시스템이 제공된다. 이 방법은 송신기 노드에서 수신기 노드로 흐르는 트래픽에 대해 데이터 센터에서 대규모로 동작하도록 구성된다. 설명된 방법은 빠른 시작 단계, 전환 단계 및 조절 단계로 이루어진 3개의 단계를 갖는다. 빠른 시작 단계에서, 송신기는 빠른 초기 속도로 데이터를 수신기에게 송신한다. 이 송신은 수신기가 정체 이벤트를 관찰할 때까지 계속될 수 있다. 수신기가 정체 이벤트를 관찰한 경우, 송신기는 이 방법이 전환 단계에 진입할 때 데이터 송신 속도를 감소시킨다. 감소된 속도로부터, 이 방법은 피드백 제어 루프와 가산적 증가 승산적 감소(additive increase multiplicative decrease; AIMD) 알고리즘의 조합을 사용하여 속도가 증가되는 조절 단계에 진입한다.

**대표도** - 도4



(52) CPC특허분류

*H04L 43/16* (2013.01)  
*H04L 47/11* (2013.01)  
*H04L 47/127* (2013.01)  
*H04L 47/25* (2013.01)  
*H04L 47/283* (2013.01)  
*H04W 28/22* (2013.01)  
*H04W 52/26* (2013.01)

(56) 선행기술조사문헌

US20170019803 A1  
US20150117218 A1  
KR1020070093077 A  
KR1020140143355 A  
KR1020160144257 A  
US20080037420 A1  
US20130343191 A1  
T. Kurp, Reconfigurable Sampling for Enhanced Energy Efficiency in Power-Constrained Wireless Systems(2011.)

---

## 명세서

### 청구범위

#### 청구항 1

네트워크 프로토콜에서 정체를 관리하는 컴퓨터 구현 방법(computer-implemented method)에 있어서,

- a) 네트워크에서 정체 이벤트를 검출하는 단계;
  - b) 데이터 송신 속도를 초기 데이터 송신 속도로부터 최소 데이터 송신 속도로 늦추는 단계;
  - c) 상기 데이터 송신 속도를 상기 초기 데이터 송신 속도의 절반으로 증가시키는 단계;
  - d) 상기 네트워크의 단방향 지연(one-way latency)을 주기적으로 모니터링하는 단계; 및
  - e) 피드백 제어 루프와 가산적 증가 승산적 감소(additive increase multiplicative decrease; AIMD) 알고리즘의 조합을 사용하여 상기 데이터 송신 속도를 조정하는 단계 - 상기 피드백 제어 루프는 제1 주파수에서 갱신된 데이터 송신 속도를 주기적으로 계산하는 데 사용되고, 상기 AIMD 알고리즘은 제2 주파수에서 상기 데이터 송신 속도를 조정하는 데 사용되며, 상기 제2 주파수는 상기 제1 주파수보다 낮음 -
- 를 포함하는, 정체를 관리하는 컴퓨터 구현 방법.

#### 청구항 2

제1항에 있어서,

상기 초기 데이터 송신 속도는 대역폭 지연 곱(bandwidth delay product)과 동일한 초기 윈도우를 설정하고, 상기 초기 윈도우를 설정한 후에 정체가 검출되지 않으면 윈도우 크기를 증가(ramp up)시킴으로써 적어도 부분적으로 결정되는 것인, 정체를 관리하는 컴퓨터 구현 방법.

#### 청구항 3

제1항에 있어서,

단방향 지연을 모니터링하는 단계는 상기 단방향 지연과 문턱 지연(threshold latency)의 비교를 수행하는 단계와, 상기 단방향 지연의 추세를 결정하는 단계를 포함하고, 상기 문턱 지연은 데이터 패킷의 송신을 위한 왕복 시간에 기초하는 것인, 정체를 관리하는 컴퓨터 구현 방법.

#### 청구항 4

제1항에 있어서,

상기 최소 데이터 송신 속도는 상기 초기 데이터 송신 속도의 적어도 1% 미만인 것인, 정체를 관리하는 컴퓨터 구현 방법.

#### 청구항 5

제1항에 있어서,

데이터 송신 속도를 초기 데이터 송신 속도로부터 최소 데이터 송신 속도로 늦추는 단계는 수신기로부터 송신기로 정체 통지 패킷을 송신하는 단계와, 송신기가 상기 정체 통지 패킷을 수신할 때 속도를 갱신하는 단계를 포함하는 것인, 정체를 관리하는 컴퓨터 구현 방법.

#### 청구항 6

제1항에 있어서,

단방향 지연을 모니터링하는 단계는 수신된 패킷의 총 바이트 수를 추적하는 단계를 포함하는 것인, 정체를 관리하는 컴퓨터 구현 방법.

**청구항 7**

제1항에 있어서,

- a) 문턱 수의 기간 동안 상기 데이터 송신 속도를 조정함으로써 적어도 상기 초기 데이터 송신 속도를 획득하는 단계; 및
- b) 정체 이벤트에 대해 상기 네트워크를 모니터링하는 단계를 더 포함하는, 정체를 관리하는 컴퓨터 구현 방법.

**청구항 8**

네트워크에서 네트워크 정체를 관리하기 위한 시스템에 있어서, 상기 시스템은 복수의 컴퓨팅 노드를 포함하고, 상기 컴퓨팅 노드는,

- a) 하나 이상의 이전에 획득된 지연 값 및 명령어의 세트를 저장하기 위한 메모리;
  - b) 동작들을 수행하기 위해 상기 명령어의 세트를 실행하도록 구성된 하나 이상의 프로세서
- 를 포함하고, 상기 동작들은,
- i) 상기 네트워크에서 정체 이벤트를 검출하는 동작;
  - ii) 데이터 송신 속도를 초기 데이터 송신 속도로부터 최소 데이터 송신 속도로 늦추는 동작;
  - iii) 상기 데이터 송신 속도를 상기 초기 데이터 송신 속도의 절반으로 증가시키는 동작;
  - iv) 상기 네트워크의 단방향 지연을 주기적으로 모니터링하는 동작;

v) 피드백 제어 루프와 가산적 증가 승산적 감소(additive increase multiplicative decrease; AIMD) 알고리즘의 조합을 사용하여 상기 데이터 송신 속도를 조정하는 동작 - 상기 피드백 제어 루프는 제1 주파수에서 갱신된 데이터 송신 속도를 주기적으로 계산하는 데 사용되고, 상기 AIMD 알고리즘은 제2 주파수에서 상기 데이터 송신 속도를 조정하는 데 사용되며, 상기 제2 주파수는 상기 제1 주파수보다 낮음 - 을 포함하는 것인, 네트워크 정체를 관리하기 위한 시스템.

**청구항 9**

제8항에 있어서,

상기 네트워크는 적어도 하나의 링크를 갖는 것인, 네트워크 정체를 관리하기 위한 시스템.

**청구항 10**

제8항에 있어서,

상기 복수의 컴퓨팅 노드는 송신기 노드 및 수신기 노드를 포함하는 것인, 네트워크 정체를 관리하기 위한 시스템.

**청구항 11**

제8항에 있어서,

상기 초기 데이터 송신 속도는 대역폭 지연 곱과 동일한 초기 윈도우를 설정하고, 상기 초기 윈도우를 설정한 후에 정체가 검출되지 않으면 윈도우 크기를 증가시킴으로써 적어도 부분적으로 결정되는 것인, 네트워크 정체를 관리하기 위한 시스템.

**청구항 12**

제8항에 있어서,

단방향 지연을 모니터링하는 동작은 상기 단방향 지연과 문턱 지연의 비교를 수행하는 동작과, 상기 단방향 지연의 추세를 결정하는 동작을 포함하고, 상기 문턱 지연은 데이터 패킷의 송신을 위한 왕복 시간에 기초하는 것

인, 네트워크 정체를 관리하기 위한 시스템.

**청구항 13**

제8항에 있어서,

상기 최소 데이터 송신 속도는 상기 초기 데이터 송신 속도의 적어도 1% 미만인 것인, 네트워크 정체를 관리하기 위한 시스템.

**청구항 14**

제8항에 있어서,

데이터 송신 속도를 초기 데이터 송신 속도로부터 최소 데이터 송신 속도로 늦추는 동작은 수신기로부터 송신기로 정체 통지 패킷을 송신하는 동작과, 송신기가 상기 정체 통지 패킷을 수신할 때 속도를 갱신하는 동작을 포함하는 것인, 네트워크 정체를 관리하기 위한 시스템.

**청구항 15**

제8항에 있어서,

단방향 지연을 모니터링하는 동작은 수신된 패킷의 총 바이트 수를 추적하는 동작을 포함하는 것인, 네트워크 정체를 관리하기 위한 시스템.

**청구항 16**

네트워크에서 네트워크 정체를 관리하기 위한 컴퓨터 구현 방법에 있어서,

- a) 데이터 속도를 초기 데이터 속도로 설정하는 단계와, 수신기가 정체 이벤트를 검출할 때까지 송신기와 수신기 간에 데이터 송신을 수행하는 단계를 포함하는 빠른 시작 단계(fast start stage);
- b) 상기 데이터 속도를 제1 기간 동안 최소 데이터 속도로 감소시킨 다음, 상기 데이터 속도를 제2 기간 동안 상기 초기 데이터 속도의 절반으로 증가시키는 단계를 포함하는 전환 단계; 및
- c) 피드백 제어 루프와 가산적 증가 승산적 감소(additive increase multiplicative decrease; AIMD) 알고리즘의 조합을 수행하여 상기 데이터 속도를 수정하는 단계를 포함하는 조절 단계 - 상기 피드백 제어 루프는 제1 주파수에서 갱신된 데이터 송신 속도를 주기적으로 계산하는 데 사용되고, 상기 AIMD 알고리즘은 제2 주파수에서 상기 데이터 송신 속도를 조정하는 데 사용되며, 상기 제2 주파수는 상기 제1 주파수보다 낮음 -

를 포함하는, 네트워크 정체를 관리하기 위한 컴퓨터 구현 방법.

**청구항 17**

제16항에 있어서,

상기 빠른 시작 단계 이전에, 상기 송신기와 상기 수신기 간에 클록 속도 및 라인 속도를 공유하는 단계를 포함하는 초기 설정 단계를 더 포함하는, 네트워크 정체를 관리하기 위한 컴퓨터 구현 방법.

**청구항 18**

제16항에 있어서,

상기 빠른 시작 단계 이전에, 데이터 패킷을 송신하기 위해 윈도우 크기를 초기 윈도우 크기로 설정하는 단계를 더 포함하는, 네트워크 정체를 관리하기 위한 컴퓨터 구현 방법.

**청구항 19**

제18항에 있어서,

상기 네트워크 내에서 정체 이벤트가 검출되지 않는 동안, 상기 윈도우 크기를 증가시키는 단계를 더 포함하는, 네트워크 정체를 관리하기 위한 컴퓨터 구현 방법.

**청구항 20**

제19항에 있어서,

상기 윈도우 크기는 기하급수적으로 증가되는 것인, 네트워크 정체를 관리하기 위한 컴퓨터 구현 방법.

**청구항 21**

제16항에 있어서,

상기 전환 단계는 상기 정체 이벤트의 검출에 이어서 시작되는 것인, 네트워크 정체를 관리하기 위한 컴퓨터 구현 방법.

**청구항 22**

제16항에 있어서,

상기 승산적 감소는 확률적 이벤트에 응답하여 수행되는 것인, 네트워크 정체를 관리하기 위한 컴퓨터 구현 방법.

**청구항 23**

제16항에 있어서,

승산적 감소는 승산적 감소 확률에 기초하여 상기 제2 주파수에서 수행되는 것인, 네트워크 정체를 관리하기 위한 컴퓨터 구현 방법.

**청구항 24**

네트워크에서 네트워크 정체를 관리하기 위한 시스템에 있어서, 상기 시스템은 복수의 컴퓨팅 노드를 포함하고, 상기 컴퓨팅 노드는,

- a) 하나 이상의 이전에 획득된 지연 값 및 명령어의 세트를 저장하기 위한 메모리; 및
- b) 상기 명령어의 세트를 실행하도록 구성된 하나 이상의 프로세서

를 포함하고, 상기 명령어의 세트는 적어도,

- i) 데이터 속도를 초기 데이터 속도로 설정하는 단계와, 수신기가 정체 이벤트를 검출할 때까지 송신기와 수신기 간에 데이터 송신을 수행하는 단계를 포함하는 빠른 시작 단계;
- ii) 상기 데이터 속도를 제1 기간 동안 최소 데이터 속도로 감소시킨 다음, 상기 데이터 속도를 제2 기간 동안 상기 초기 데이터 속도의 절반으로 증가시키는 단계를 포함하는 전환 단계; 및
- iii) 피드백 제어 루프와 가산적 증가 승산적 감소(additive increase multiplicative decrease; AIMD) 알고리즘의 조합을 수행하여 상기 데이터 속도를 수정하는 단계를 포함하는 조절 단계 - 상기 피드백 제어 루프는 제1 주파수에서 갱신된 데이터 송신 속도를 주기적으로 계산하는 데 사용되고, 상기 AIMD 알고리즘은 제2 주파수에서 상기 데이터 송신 속도를 조정하는 데 사용되며, 상기 제2 주파수는 상기 제1 주파수보다 낮음 -

를 구현하기 위한 것인, 네트워크 정체를 관리하기 위한 시스템.

**청구항 25**

제24항에 있어서,

상기 빠른 시작 단계 이전에, 상기 송신기와 상기 수신기 간에 클록 속도 및 라인 속도를 공유하는 단계를 포함하는 초기 설정 단계를 더 포함하는, 네트워크 정체를 관리하기 위한 시스템.

**청구항 26**

제24항에 있어서,

상기 빠른 시작 단계 이전에, 데이터 패킷을 송신하기 위해 윈도우 크기를 초기 윈도우 크기로 설정하는 것을 더 포함하는, 네트워크 정체를 관리하기 위한 시스템.

**청구항 27**

제26항에 있어서,

상기 네트워크 내에서 정체 이벤트가 검출되지 않는 동안, 상기 윈도우 크기를 증가시키는 것을 더 포함하는, 네트워크 정체를 관리하기 위한 시스템.

**청구항 28**

제27항에 있어서,

상기 윈도우 크기는 기하급수적으로 증가되는 것인, 네트워크 정체를 관리하기 위한 시스템.

**청구항 29**

제24항에 있어서,

상기 전환 단계는 상기 정체 이벤트의 검출에 이어서 시작되는 것인, 네트워크 정체를 관리하기 위한 시스템.

**청구항 30**

제24항에 있어서,

상기 승산적 감소는 확률적 이벤트에 응답하여 수행되는 것인, 네트워크 정체를 관리하기 위한 시스템.

**청구항 31**

제30항에 있어서,

상기 승산적 감소는 상기 확률적 이벤트에 기초하여 상기 제2 주파수에서 수행되는 것인, 네트워크 정체를 관리하기 위한 시스템.

**청구항 32**

삭제

**청구항 33**

삭제

**청구항 34**

삭제

**청구항 35**

삭제

**청구항 36**

삭제

**청구항 37**

삭제

**청구항 38**

삭제

**청구항 39**

삭제

- 청구항 40  
삭제
- 청구항 41  
삭제
- 청구항 42  
삭제
- 청구항 43  
삭제
- 청구항 44  
삭제
- 청구항 45  
삭제
- 청구항 46  
삭제
- 청구항 47  
삭제
- 청구항 48  
삭제
- 청구항 49  
삭제
- 청구항 50  
삭제
- 청구항 51  
삭제
- 청구항 52  
삭제
- 청구항 53  
삭제
- 청구항 54  
삭제
- 청구항 55  
삭제



**발명의 설명**

**기술 분야**

[0001] 관련 출원들에 대한 상호참조

[0002] 본 출원은 2019년 5월 17일에 출원된 "속도 최적화된 정체 관리(RATE-OPTIMIZED CONGESTION MANAGEMENT)"라는 발명의 명칭의 미국 출원 제16/415,609호의 이익을 주장하며, 그 내용은 모든 목적을 위해 참조로 여기에 포함된다.

**배경 기술**

[0003] 데이터 센터 애플리케이션은 호스트의 네트워크 스택을 포함하여 네트워크로부터 높은 처리량과 매우 낮은 지연(latency)을 요구한다. 표준 TCP/IP의 높은 CPU 오버헤드로 인해, 대체 기술로서 이더넷을 통한 원격 직접 메모리 액세스(Remote Direct Memory Access; RDMA)(RDMA over Ethernet; RoCEv2)이 현대 데이터 센터에 배포되어 낮은 지연과 고성능을 달성한다. 부하가 많으면 데이터 센터 네트워크에 정체가 발생하고 패킷 드롭(drop)이 발생할 수 있다. 결과적으로 RDMA의 성능이 크게 저하되고 RoCEv2는 드롭 없는 네트워크를 달성하기 위해 PFC(Priority-based Flow Control)에 의존한다. PFC는 네트워크의 한 영역에서 트래픽이 느려지면 네트워크의 다른 영역에서 트래픽을 느려지게 하는 정체 확산을 일으킬 수 있다. 따라서 정체 확산을 회피하도록 정체 관리가 필요하다. 예를 들어, DCQCN(Data Center Quantized Congestion Notification)과 같은 RoCEv2 네트워크의 기존 정체 관리 방법은 대규모로 제대로 동작하지 않아 수천 개의 일시 중지(pause) 프레임을 생성하고 네트워크 활용도와 평균 처리량이 낮고 정체된 흐름이 매우 느리게 종료되게 할 수 있다.

**발명의 내용**

[0004] 대규모로 효과적인 데이터 센터 네트워크의 정체 관리가 필요하다. 이러한 방법은 높은 링크 활용도, 매우 낮은 지연 및 네트워크 공정성(network fairness)을 달성할 수 있다. 개시된 방법은 정체 이벤트에 응답하여 데이터 속도를 동적으로 변경하고 피드백 제어 알고리즘을 가산적 증가 승산적 감소(additive increase multiplicative decrease; AIMD) 알고리즘과 결합함으로써 이러한 목표를 달성한다.

[0005] 일 양상에서, 네트워크 프로토콜에서 정체를 관리하는 컴퓨터 구현 방법(computer-implemented method)이 개시된다. 방법은 네트워크에서 정체 이벤트를 검출하는 단계를 포함한다. 그런 다음, 방법은 초기 속도로부터 최소 속도로 데이터 송신 속도를 늦추는 단계를 포함한다. 그런 다음, 방법은 데이터 송신 속도를 초기 속도의 절반으로 증가시키는 단계를 포함한다. 그런 다음, 방법은 네트워크의 단방향 지연(one-way latency)을 주기적으로 모니터링하는 단계를 포함한다. 마지막으로, 네트워크는 피드백 제어 알고리즘과 가산적 증가 승산적 감소(AIMD) 알고리즘을 사용하여 데이터 송신 속도를 조정하는 단계를 포함한다. 일부 실시예에서, 초기 데이터 송신 속도를 설정하는 단계는 대역폭 지연 곱(bandwidth delay product)과 동일한 초기 윈도우를 설정하는 단계와 초기 윈도우를 설정한 후에 정체가 검출되지 않으면 윈도우 크기를 증가(ramp up)시키는 단계를 더 포함한다. 일부 실시예에서, 초기 데이터 송신 속도는 네트워크의 회선 속도(line rate)와 동일하다. 일부 실시예에서, 문턱 지연(threshold latency)은 데이터 패킷의 송신을 위한 왕복 시간에 기초한다. 일부 실시예에서, 최소 데이터 송신 속도는 초기 데이터 송신 속도의 적어도 1퍼센트 미만이다. 일부 실시예에서, 피드백 제어 루프는 비례적분 제어를 포함한다. 일부 실시예에서, 최소 데이터 송신 속도로 송신기로부터 수신기로 데이터를 송신하는 단계는 수신기로부터 송신기로 정체 통지 패킷을 송신하는 단계, 및 송신기가 정체 통지 패킷을 수신할 때 속도를 갱신하는 단계를 포함한다. 일부 실시예에서, 단방향 지연을 지속적으로 모니터링하는 단계는 수신된 패킷의 총 바이트 수를 추적하는 단계를 포함한다. 일부 실시예에서, 관리되는 송신 속도는 최소 송신 속도에서의 한 기간 및 초기 속도의 절반에서의 한 기간 후에 계산된다. 일부 실시예에서, 기간은 송신된 패킷의 왕복 시간에 기초한다. 일부 실시예에서, 초기 송신 속도는 송신기와 수신기 간의 회선 속도에 기초한다. 일부 실시예에서, 초기 송신 속도는 송신기의 링크 속도와 수신기의 링크 속도 간의 최소 속도이다. 일부 실시예에서, AIMD 알고리즘은 관리되는 속도가 문턱값 미만인 경우 관리 속도에 대해 가산(addition)을 수행하고 관리되는 속도가 문턱값을 초과하는 경우 관리되는 속도에 대해 승산(multiplication)을 수행하는 단계를 포함한다. 일부 실시예에서, 가산은 양수에 의해 되어진다. 일부 실시예에서, 곱셈은 0과 1 사이의 수에 의해 되어진다. 일부 실시예에서, 컴퓨터 구현 방법은 문턱 수의 기간 동안 데이터 송신 속도를 조정하는 단계와 정체 이벤트에 대해 네트워크를 모니터링하는 단계로부터 적어도 초기 속도를 획득하는 단계를 더 포함한다.

[0006]

또 다른 양상에서, 네트워크에서 네트워크 정체를 관리하기 위한 시스템이 개시된다. 시스템은 복수의 컴퓨팅 노드를 포함한다. 컴퓨팅 노드는 하나 이상의 프로세서에 추가하여 하나 이상의 이전에 획득된 지연 값 및 명령어 세트를 저장하기 위한 메모리를 포함한다. 프로세서는 일련의 명령어를 실행하여 물리 계층 링크에서 정체 이벤트를 검출하고, 초기 속도로부터 최소 속도로 데이터 송신 속도를 늦추고, 데이터 송신 속도를 초기 속도의 절반으로 증가시키고, 주기적으로 네트워크의 단방향 지연을 모니터링하며, 피드백 제어 알고리즘 및 가산적 증가 승산적 감소(AIMD) 알고리즘을 사용하여 데이터 송신 속도를 조정하도록 구성된다. 일부 실시예에서, 네트워크는 적어도 하나의 이더넷 링크를 갖는다. 일부 실시예에서, 복수의 컴퓨팅 노드는 송신기 노드 및 수신기 노드를 포함한다. 일부 실시예에서, 초기 데이터 송신 속도를 설정하는 것은 대역폭 지연 곱과 동일한 초기 윈도우를 설정하는 것과 초기 윈도우를 설정한 후에 정체가 검출되지 않으면 윈도우 크기를 증가시키는 것을 더 포함한다. 일부 실시예에서, 초기 데이터 송신 속도는 이더넷 링크의 회선 속도와 동일하다. 일부 실시예에서, 문턱 지연은 데이터 패킷의 송신을 위한 왕복 시간에 기초한다. 일부 실시예에서, 최소 데이터 송신 속도는 초기 데이터 송신 속도의 적어도 1퍼센트 미만이다. 일부 실시예에서, 피드백 제어 루프는 비례 적분 제어기를 포함한다. 일부 실시예에서, 최소 데이터 송신 속도로 송신기로부터 수신기로 데이터를 송신하는 것은 수신기로부터 송신기로 정체 통지 패킷을 송신하는 것, 및 송신기가 정체 통지 패킷을 수신할 때 속도를 갱신하는 것을 포함한다. 일부 실시예에서, 단방향 지연을 지속적으로 모니터링하는 것은 수신된 패킷의 총 바이트 수를 추적하는 것을 포함한다. 일부 실시예에서, 관리되는 송신 속도는 최소 송신 속도에서의 한 기간 및 초기 속도의 절반에서의 한 기간 후에 계산된다. 일부 실시예에서, 기간은 송신된 패킷의 왕복 시간에 기초한다. 일부 실시예에서, 초기 송신 속도는 송신기와 수신기 간의 회선 속도에 기초한다. 일부 실시예에서, 초기 송신 속도는 송신기와 수신기 간의 최소 회선 속도이다. 일부 실시예에서, AIMD 알고리즘은 관리되는 속도가 문턱값 미만인 경우 관리되는 속도에 대해 가산을 수행하고 관리되는 속도가 문턱값을 초과하는 경우 관리되는 속도에 대해 승산을 수행하는 것을 포함한다. 일부 실시예에서, 가산은 양수에 의해 되어진다. 일부 실시예에서, 승산은 0과 1 사이의 수에 의해 되어진다.

[0007]

또 다른 양상에서, 네트워크 프로토콜에서 정체를 관리하는 컴퓨터 구현 방법이 개시된다. 방법은 데이터 속도를 초기 데이터 속도로 설정하는 단계와, 수신기가 정체 이벤트를 검출할 때까지 송신기와 수신기 간에 데이터 송신을 수행하는 단계를 포함하는 빠른 시작 단계(fast start stage)를 포함한다. 방법은 또한 데이터 속도를 한 기간 동안 최소 데이터 속도로 감소시킨 다음 데이터 속도를 한 기간 동안 초기 속도의 절반으로 증가시키는 단계를 포함하는 전환 단계를 포함한다. 방법은 또한 데이터 속도를 수정하기 위해 피드백 제어 루프와 가산적 감소 승산적 증가 알고리즘의 조합을 사용하는 단계를 포함하는 조절 단계를 포함한다. 일부 실시예에서, 컴퓨터 구현 방법은 빠른 시작 단계 이전에, 송신기와 수신기 간에 클록 속도 및 회선 속도를 공유하는 단계를 포함하는 초기 설정 단계를 더 포함한다. 일부 실시예에서, 컴퓨터 구현 방법은 빠른 시작 단계 이전에, 데이터 패킷을 송신하기 위해 윈도우 크기를 초기 윈도우 크기로 설정하는 단계를 더 포함한다. 일부 실시예에서, 컴퓨터 구현 방법은 네트워크 내에서 정체 이벤트가 검출되지 않는 동안 윈도우 크기를 증가시키는 단계를 더 포함한다. 일부 실시예에서, 윈도우 크기는 기하급수적으로 증가된다. 일부 실시예에서, 전환 단계는 정체 이벤트의 검출 이후에 시작된다. 일부 실시예에서, 피드백 제어 알고리즘은 비례 적분 알고리즘이다. 일부 실시예에서, 피드백 제어 루프는 현재 속도, 비례항(proportional term), 및 적분항(integral term) 간의 차이로 설명될 수 있다. 일부 실시예에서, 비례항은 현재 지연과 문턱 지연 간의 차이를 설명하는 반면, 적분항은 현재 지연과 이전 지연 간의 차이를 설명한다. 일부 실시예에서, 승산적 증가는 확률적 이벤트에 응답하여 수행된다. 일부 실시예에서, 승산적 증가는 피드백 제어 루프보다 낮은 주파수에서 수행된다.

[0008]

또 다른 양상에서, 네트워크에서 네트워크 정체를 관리하기 위한 시스템이 개시된다. 시스템은 복수의 컴퓨팅 노드를 포함한다. 컴퓨팅 노드는 하나 이상의 이전에 획득된 지연 값 및 명령어 세트를 저장하기 위한 메모리를 포함한다. 컴퓨팅 노드는 또한, 데이터 속도를 초기 데이터 속도로 설정하는 단계, 수신기가 정체 이벤트를 검출할 때까지 송신기와 수신기 간에 데이터 송신을 수행하는 단계를 포함하는 빠른 시작 단계, 한 기간 동안 데이터 속도를 최소 데이터 속도로 감소시킨 다음 한 기간 동안 초기 속도의 절반으로 데이터 속도를 증가시키는 단계를 포함하는 전환 단계, 및 데이터 속도를 수정하기 위해 피드백 제어 루프와 가산적 감소 승산적 증가 알고리즘의 조합을 사용하는 단계를 포함하는 조절 단계를 구현하기 위한 명령어들의 세트를 실행하도록 구성된 하나 이상의 프로세서를 포함한다. 일부 실시예에서, 시스템은 빠른 시작 단계 이전에, 송신기와 수신기 간에 클록 속도 및 회선 속도 속도를 공유하는 단계를 포함하는 초기 설정 단계를 더 포함한다. 일부 실시예에서, 시스템은 빠른 시작 단계 이전에 데이터 패킷을 송신하기 위해 윈도우 크기를 초기 윈도우 크기로 설정하는 것을 더 포함한다. 일부 실시예에서, 시스템은 네트워크 내에서 정체 이벤트가 검출되지 않는 동안 윈도우 크기를 증가시키는 것을 더 포함한다. 일부 실시예에서, 윈도우 크기는 기하급수적으로 증가된다. 일부 실시예에서, 전환

단계는 정체 이벤트의 검출 이후에 시작된다. 일부 실시예에서, 피드백 제어 알고리즘은 비례 적분 알고리즘이다. 일부 실시예에서, 피드백 제어 루프는 현재 속도, 비례항, 및 적분항 간의 차이로 설명될 수 있다. 일부 실시예에서, 비례항은 현재 지연과 문턱 지연 간의 차이를 설명하는 반면, 적분항은 현재 지연과 이전 지연 간의 차이를 설명한다. 일부 실시예에서, 승산적 증가는 확률적 이벤트에 응답하여 수행된다. 일부 실시예에서, 승산적 증가는 피드백 제어 루프보다 낮은 주파수에서 수행된다.

**도면의 간단한 설명**

- [0009] 본 요지(subject matter)의 진보적 특징들은 첨부된 청구항들에 자세히 기재되어 있다. 본 요지의 특징 및 장점은 본 요지의 원리가 이용되는 예시적인 실시예를 기재하는 다음의 상세한 설명과, 그 첨부된 도안(본 명세서에서 "도면(figure)" 및 "도(FIG)"로 또한 지칭됨)을 참조함으로써 더 잘 이해가 될 것이다.
- 도 1은 링크를 통해 통신을 나타내는 예시적인 블록도이다.
- 도 2는 본 명세서에 설명된 실시예들에 따라 정체 관리 시스템에서 측정된 복수의 변수들을 개략적으로 도시한다.
- 도 3은 물리적 계층 링크에서의 정체 관리 방법에 대한 흐름도를 도시한다.
- 도 4는 통신을 위한 최적의 동작 포인트를 도시한다.
- 도 5는 두 방법으로 수행된 실험으로부터의 데이터를 사용하여 DCQCN과 개시된 방법의 지연 비교를 도시한다.
- 도 6a 및 6b는 도 5의 실험으로부터 네트워크에서의 데이터 흐름의 예시를 도시한다.
- 도 7a 및 도 7b는 개시된 방법과 DCQCN 간의 큐 점유(queue occupancy)의 비교를 도시한다.

**발명을 실시하기 위한 구체적인 내용**

- [0010] 개시된 방법은 링크 계층, 예를 들어, 이더넷 네트워크에서 RoCEv2 네트워크의 정체를 효과적으로 관리한다. 본 방법은 하나 이상의 네트워크 링크에 의해 접속될 수 있는 송신기 노드와 수신기 노드 간의 통신에서 구현될 수 있다. 송신기와 수신기는 다수의 사용자로부터의 데이터 흐름을 관리할 수 있으며 링크를 통해 데이터를 송신하는 다수의 사용자 간에 송신된 데이터에 대한 효율성과 공정성을 유지해야 할 수 있다.
- [0011] 개시된 방법은 빠른 시작 단계, 전환 단계 및 조절 단계로 이루어진 3개의 주요 단계를 가지고 있다. 빠른 시작 단계에서, 송신기와 수신기는 수신기가 정체 이벤트를 기록할 때까지 초기 속도로 서로 데이터를 송신할 수 있다. 정체 이벤트는 수신기의 관점에서 측정된 지연 시간의 급증(spike)으로서 검출될 수 있다. 정체 이벤트 이후에, 전환 단계가 시작된다. 전환 단계에서, 송신기는 정체를 관리하기 위해, 초기에 데이터 속도를 한 기간 동안 최소 속도로 감소시킨 다음, 데이터 속도를 초기 속도의 절반으로 증가시킨다. 전환 단계에 이어, 조절 단계는 피드백 제어와 가산적 증가 승산적 감소(additive increase multiplicative decrease; AIMD) 알고리즘의 조합을 사용하여 낮은 지연, 효율성 및 공정성을 유지하면서 속도를 증가시킨다.
- [0012] 피드백 제어 루프는 속도 증가 또는 감소가 예를 들어, 비례 적분 알고리즘을 사용하여 정밀하게 조정되는 조절 단계에서 주요 속도 계산을 안내한다. AIMD 알고리즘은 흐름이 장기간에 걸쳐 균형을 다시 맞추도록 하는 작은 섭동(perturbation)을 추가한다. 이것은 데이터 패킷의 정체된 큐로의 총 도착 속도에 대해 너무 많은 중단을 야기하지 않고 수행된다.
- [0013] 조절 단계에 이어서, 송신기와 수신기는 결국 초기 속도로 데이터 송신을 다시 시작할 수 있다. 또 다른 정체 이벤트가 검출되면 본 방법이 다시 시작될 수 있다.
- [0014] 일 양상에서, 지연의 변화의 측정에 기초하여 네트워크 정체 상태를 결정하기 위한 본 개시의 방법 및 시스템이 제공된다. 지연 측정은 순방향 경로 지연 또는 예를 들어, 왕복 시간(round-trip time; RTT) 지연과 같은 왕복 지연일 수 있다. 지연의 변화 또는 변경은 정체 신호로 사용될 수 있으므로 송신기는 정체 정도에 대해 적시에 통지될 수 있다.
- [0015] 도 1은 여기에 설명된 요지의 실시예에 따른 링크(103)를 통한 통신을 도시하는 예시적인 블록도이다. 송신기(101) 및 수신기(105)는 링크(103)를 통해 통신하고 있다. 여기에 설명된 예에서, 시스템은 특정 송신 및 수신 작업을 송신기 및 수신기에게 위임한다. 그러나 패킷은 송신기에서 수신기로 또는 수신기에서 송신기로 송신될 수 있다. 통신은 정체를 제어하기 위해 정체 관리 메커니즘을 사용할 수 있다. 일부 경우에, 링크를 따라 네트

워크 정체 상태가 결정될 수 있다. 송신기와 수신기는 각각 소스 및 목적지라고 지칭될 수도 있다. 다른 경우에는, 왕복(송신기에서 수신기로 및 그 반대로)에 대한 네트워크 정체가 결정될 수 있다.

[0016] 송신기(101) 및 수신기(105)는 예를 들어, 개인용 컴퓨터, 태블릿 컴퓨터, 스마트폰, 셋톱 박스, 데스크톱 컴퓨터, 랩톱, 게임 시스템, 서버, 데이터 센터 및 다양한 다른 디바이스 또는 시스템과 같은 다양한 유형의 컴퓨팅 디바이스일 수 있다. 복수의 다른 디바이스 또는 컴포넌트가 네트워크를 통해 접속을 확립할 수 있다. 하나 이상의 송신기는 하나 이상의 수신기와 접속할 수 있다. 송신기와 수신기는 유선 및/또는 무선 접속을 통해 네트워크에 접속할 수 있다. 송신기 또는 수신기는 네트워크에서 소스 노드 및 목적지 노드라고 지칭될 수 있다. 노드는 통신 능력을 갖춘 임의의 디바이스일 수 있다. 통신은 유선 통신 또는 무선 통신일 수 있다.

[0017] 송신기와 수신기는 각각 메모리(120)와 CPU(130)를 갖는다. 수신기는 RDMA를 사용하여 송신기의 운영 체제를 우회하면서 송신기의 메모리에 저장된 항목에 액세스할 수 있다.

[0018] 링크(103)는 무선 네트워크, 유선 네트워크 또는 이 둘의 조합으로부터 유래할 수 있다. 예를 들어, 네트워크는 ISP, 셀룰러 또는 광대역 케이블 제공자를 통한 인터넷, 인트라넷, 셀룰러 네트워크, 홈 네트워크, 개인 영역 네트워크 등 중 하나 이상을 포함할 수 있다. 링크(103)는 예를 들어, 하나 이상의 네트워크 컴포넌트, 데이터 서버, 접속 노드, 스위치 등과 같은 인터넷 프로토콜 인터페이스와 관련될 수 있다. 또 다른 양상에서, 송신기(101) 및 수신기(105)는 링크(103)의 일부로서 고려될 수 있다.

[0019] 송신기(101) 및 수신기(105) 모두는 네트워크 인터페이스 카드(network interface card; NIC)일 수 있는 네트워크 인터페이스(140)를 포함한다. 네트워크 인터페이스를 통해 송신기 노드 및 수신기 노드가 링크(103)와 인터페이싱할 수 있다. 네트워크 인터페이스(140)는 스마트 능력(smart capabilities)을 가질 수 있으며, 이를 통해 네트워크의 노드에 대해 데이터 송신 속도를 제어하는 알고리즘을 구현할 수 있다.

[0020] 소스 노드(예컨대, 송신기)로부터 목적지 노드(예컨대, 수신기)로 송신된 데이터 패킷(111)은 링크를 따라 지연을 경험할 수도 있고 경험하지 않을 수도 있다. 일부 실시예에서, 지연의 변동 또는 변화의 측정은 정체 정도의 즉각적인 신호를 제공할 수 있다. 지연은 데이터 패킷(111)의 도착 시간에서 데이터 패킷(111)의 송신 시간을 뺀 값으로 계산될 수 있다. 일부 실시예에서, 데이터 패킷의 송신 시간은 송신기의 클럭(107)에 의해 제공되고 데이터 패킷의 도착/수신 시간은 수신기의 클럭(109)에 의해 제공된다.

[0021] 송신기 클럭(107)과 수신기 클럭(109)은 동기화될 필요가 없다. 일부 경우에, 송신기 클럭(107) 및/또는 수신기 클럭(109)은 동기화되지 않은 자유 실행 클럭(free running clocks)일 수 있다. 일부 경우에, 송신기 클럭(107) 및 수신기 클럭(109)은 동일한 속도들 또는 주파수들에서 동작하고 있다. 대안적으로, 송신기 클럭(107) 및 수신기 클럭(109)은 상이한 속도 또는 주파수에서 실행될 수 있다. 일부 경우에, 접속 확립 시간에 공통 속도 또는 주파수가 설정될 수 있다. 예를 들어, 공통 속도 또는 주파수는 데이터 패킷을 송신하기 전에 접속 프로토콜에 의해 미리 결정된 주파수이다. 일부 경우에, 송신기 및 수신기 각각과 연관된 속도 또는 주파수가 접속 확립에서 교환될 수 있다. 일부 경우에, 수신기는 송신기의 속도나 주파수에 의해 통지될 수 있다. 대안적 경우에, 송신기와 수신기는 서로 승인되지 않은(unacknowledged) 상이한 주파수들에서 실행될 수 있다. 예를 들어, 송신기와 수신기는 모두 접속 설정에서 핸드셰이크 요청(예컨대, 핸드셰이크 유형을 갖는 제어 패킷)을 송신하며, 여기서 핸드셰이크 패킷은 송신기 및 수신기 모두가 각각 송신기 및 수신기와 연관된 주파수를 사용하여 통신하기로 동의하는 공통 주파수를 포함할 수 있다. 클럭 주파수 또는 속도는 접속 확립 시간뿐만 아니라 언제든지 송신될 수 있다는 점에 유의해야 한다. 예를 들어, 클럭의 주파수를 포함하는 핸드셰이크 패킷은, 통지 메시지가 생성되거나 타임아웃 타이머가 만료될 때 간격으로 한 번 송신될 수 있다. 클럭은 예를 들어, NIC 하드웨어 시간 또는 호스트 디바이스(예컨대, CPU) 클럭과 같은 임의의 유형의 로컬 하드웨어 또는 디바이스 클럭을 포함할 수 있다. 송신기 클럭과 수신기 클럭은 동일한 유형일 수도 있고 아닐 수도 있다. 일부 상황에서, NIC 타임스탬프를 호스트 타임스탬프에 매핑하기 위해 교정 또는 매핑 체계가 사용될 수 있다. 일부 경우에, 송신기 클럭과 수신기 클럭 간에 클럭 주파수 드리프트가 발생하여 지연이 변경되는 경향이 나타날 수 있다. 주파수 드리프트의 검출 및 정정은 본 명세서의 뒷부분에서 논의된다.

[0022] 데이터 패킷(111)이 송신되는 현재 시간을 나타내는 타임스탬프가 데이터 패킷(111)에 포함될 수 있다. 타임스탬프는 송신기 클럭에 의해 생성될 수 있다. 타임스탬프는 데이터 패킷의 임의의 필드에 임베드(embed)될 수 있다. 타임스탬프는 다중 비트일 수 있다. 타임스탬프는 예를 들어, 32비트 또는 64비트와 같은 임의의 수의 비트를 포함할 수 있다. 타임스탬프는 피코초, 밀리초 또는 마이크로초 규모의 시간 해상도를 제공할 수 있다.

[0023] 일부 실시예에서, 각각의 데이터 패킷(111)은 타임스탬핑된다. 대안적으로, 일부 데이터 패킷은

타임스탬핑된다. 예를 들어, 송신 시간을 나타내는 타임스탬프는 X개의 패킷마다 포함될 수 있으며, 여기서 X는 임의의 정수일 수 있다.

[0024] 타임스탬프를 포함하는 데이터 패킷을 수신하면, 수신기에 의해 지연이 계산될 수 있다. 지연은 수신기에서 로컬 클럭의 현재 값에서 데이터 패킷의 타임스탬프 값을 빼거나 그 반대로 해서 결정될 수 있다. 수신기에서 로컬 클럭의 현재 값이 송신기 클럭과 동기화되지 않을 수 있으므로, 그 차이(differential)가 절대 지연을 나타내지 않을 수 있다. 따라서 시간 차이는 의사 지연(pseudo-latency)이라고 할 수 있다. 본 개시에서, "의사-지연" 및 "지연"이라는 용어들은 동가이며, "지연"이 절대 지연을 나타내는 것으로 해석되어서는 안 된다.

[0025] 지연의 차이 또는 변화는 정체의 정도를 예측하는 데 사용될 수 있다. 지연 값들의 변화의 차이는 큐잉 지연(queuing delay)의 변화를 나타낼 수 있다. 예를 들어, 지연 값의 증가로 인한 지연의 양의 변화(positive change)는 상승하는 큐를 나타낼 수 있고, 지연의 음의 변화(negative change)는 하강하는 큐를 나타낼 수 있다. 양의 변화가 검출되는 상황에서, 대기 큐(standing queue)가 형성될 때까지 기다리지 않고 지연(delay)에 대응함으로써 저지연 방법을 달성하는 것이 유리하다. 음의 변화가 검출되는 상황에서, 대역폭을 최대한 활용하기 위해 송신 속도를 조정하는 것이 유리하다. 일부 경우에, 예를 들면, 지연 변화를 지연 차이 문턱값과 비교하는 것에 의해 지연 변화가 특정 범위를 초과하는 것으로 결정되면, 통지 메시지가 생성될 수 있다. 증가 변화 또는 감소 변화가 미리 결정된 지연 차이 문턱값을 초과할 때 통지 메시지가 생성될 수 있다. 정체 상태는 문턱값을 초과하는 지연의 변화를 검출하는 것을 포함할 수 있다. 정체 상태는 측정된 지연 변화를 포함할 수 있다. 정체 상태는 본 명세서에서 나중에 논의되는 바와 같이 지연과 관련된 임의의 다른 변수를 더 포함할 수 있다. 일부 실시예들에서, 정체 상태의 결정 시에, 통지 메시지가 수신기에 의해 생성될 수 있다. 통지 메시지는 지연과 관련된 복수의 변수를 포함할 수 있다.

[0026] 도 2는 본 명세서에 설명된 요지의 실시예들에 따라 정체 관리 시스템에서 측정된 복수의 변수들을 개략적으로 도시한다. 송신기(201) 및 수신기(203)는 도 1에 설명된 바와 같이 각각 로컬 클럭(205, 207)과 연관될 수 있다. 송신기 및 수신기는 예를 들어, 데이터 패킷, 제어 패킷, 통지 패킷 등과 같은 메시지를 송신하고 수신하도록 구성될 수 있다. 도 2에 도시된 바와 같이, 하나 이상의 데이터 패킷(211, 217, 219)이 송신기(201)로부터 수신기(203)로 송신될 수 있다. 일부 경우에, 하나 이상의 데이터 패킷은 송신기의 로컬 클럭(205)에 의해 타임스탬핑될 수 있다. 송신기와 수신기는 통신 시스템에 묘사되지만 임의의 다른 컴포넌트도 시스템에 포함될 수 있다.

[0027] 데이터 패킷은 제1 시간에 송신기 또는 수신기로부터 송신되고 제2 시간에 상대방에 의해 수신될 수 있다. 제1 시간(예컨대, T1) 및 제2 시간(예컨대, T2)은 각각 로컬 클럭(205, 207)에 의해 결정될 수 있다. 예를 들어, T4 및 T5, 또는 T6 및 T7과 같은 다양한 시점은 동시에 그리고/또는 다양한 순서로 발생할 수 있다. 각 통신은 하나 이상의 패킷을 포함할 수 있다는 점에 유의해야 한다; 그러나 통신은 가독성을 위해 한 컴포넌트에서 다른 컴포넌트로 송신되는 단일 패킷으로서 설명된다. 또한, 통신은 가독성을 위해 겹치지 않고 다양한 시간에 묘사된다는 점에 또한 유의해야 한다. 또한, 문맥이 달리 제시하지 않는 한, 임의의 통신은 중첩될 수 있고, 다른 순서로 있을 수 있으며, 다양한 네트워크 컴포넌트(예컨대, 중간 노드, 스위치 등) 간에 전달될 수 있다. 송신기와 수신기 간에 임의의 다른 양의 메시지가 송신되고 수신될 수 있다는 점에 유의해야 한다; 마찬가지로, 송신기와 수신기는 다이어그램에 도시되지 않은 다양한 다른 컴포넌트에 메시지를 송신하고 그리고/또는 이 컴포넌트로부터 메시지를 수신할 수 있다.

[0028] 전술한 바와 같이, 지연의 변화에 기초해 지연의 정도가 결정될 수 있다. 일부 실시예에서, 지연의 변화가 지연 차이 문턱값을 초과하는 것으로 결정될 때, 통지 메시지(215, 216)가 생성될 수 있다. 일부 경우에, 새로운 간격(227)은 통지 메시지의 생성에서 시작될 수 있다. 일부 경우에, 새로운 간격(227)은 정체 상태의 검출에서 시작될 수 있다. 일부 경우에, 새로운 간격은 이전 간격으로부터 마지막 데이터 패킷(211)의 수신 시간 또는 도착 시간(예컨대, T2)에서 시작될 수 있다. 간격은 예를 들어, 마이크로초, 밀리초, 초 등과 같은 임의의 시간 길이를 가질 수 있다.

[0029] 이 도면에 도시된 바와 같이, 간격 내 제1 데이터 패킷의 지연(221)이 계산될 수 있다. 수신기(203)에 의해 제1 데이터 패킷을 수신하면, 수신기는 수신기에서의 로컬 클럭의 현재 값 T4에서 데이터 패킷의 타임스탬프 값 T3을 뺄 수 있다. 지연 값은 송신기로부터 수신기로 송신된 하나 이상의 메시지 또는 데이터 패킷(217)에 대해 계산될 수 있다. 두 개 이상의 데이터 패킷 간에 지연의 차이 또는 변화가 계산될 수 있다.

[0030] 일부 경우에, 지연의 변화는 현재 지연 값과 이전 지연 값의 차이로서 계산될 수 있다. 이전 지연 값은 단일 패킷의 지연 값 또는 이전에 획득된 다수의 패킷의 평균일 수 있다. 일부 경우에, 이전 지연 값은 간격(227)에서

제1 패킷(213)의 지연 값(221)일 수 있다. 일부 경우에, 이전 지연 값은 최소 지연 값일 수 있다. 일부 경우에, 이전 지연 값은 예를 들어, 제1 다수의 패킷과 같은 다수의 이전에 획득된 지연 값의 평균 또는 다수의 패킷의 이동 평균일 수 있다. 일부 경우에, 이전 간격으로부터의 데이터에 기초해 이전 지연 값이 생성될 수 있다. 이전 지연 값은 간격(227) 동안 고정되거나 변경될 수 있다. 예를 들어, 수신기(203)에 의해 추적된 최소 지연 값이 이전 지연 값으로서 취해지면, 최솟값이 갱신될 때 이전 지연 값이 변경될 수 있다. 예를 들어, 데이터 패킷(217)은 간격에서 최소 지연 값(223)을 가질 수 있고, 그러면, 후속 데이터 패킷에 대해, 지연의 변화는 각각의 지연 값과 최소 지연 값 간의 차이로서 계산될 수 있다. 유사하게, 제1 패킷의 지연 값(221)이 이전 지연 값으로 취해질 때, 간격 내의 모든 후속 데이터 패킷에 대해, 지연의 변화는 각각의 지연 값과 제1 패킷의 지연 값 간의 차이로서 계산될 수 있다.

[0031] 일부 경우에, 지연의 변화가 현재 지연 값과 이전 지연 값 간의 차이로서 결정된다. 현재 지연 값은 단일 패킷의 지연 값일 수 있다. 현재 지연 값은 다수의 패킷의 이동 평균일 수 있다. 이는 지연 측정에서 순간적인 급증을 방지하는 이점을 제공할 수 있다. 다양한 다른 방법 또는 알고리즘을 사용하여 급증을 필터링할 수 있다. 일부 경우에, 지연의 변화는 현재 지연 값과 이전 지연 값으로부터 유도된 값일 수 있다. 예를 들어, 현재 지연 값과 이전 지연 값 간의 차이를 이전 지연 값(또는 현재 지연 값)으로 나눈 값인 지연의 변화는 무차원(dimensionless)일 수 있다.

[0032] 일부 경우에, 간격의 마지막 패킷(219)은 지연 차이 문턱값을 초과하는 지연 변화를 갖는 패킷이다. 마지막 패킷의 현재 지연은 이동 평균일 수도 있고 아닐 수도 있다. 이 변화는 현재 지연 값과 이전 지연 값 간의 차이의 절댓값일 수 있다. 이 차이는 양수 또는 음수일 수 있다. 지연 차이 문턱값이 충족될 때, 통지 메시지(216)가 생성될 수 있다.

[0033] 일부 실시예에서, 지연의 변화는 수신기에 의해 결정될 수 있다. 대안적으로, 지연의 변화는 송신기에 의해 결정될 수 있다. 일부 실시예에서, 네트워크 정체의 상태는 수신기에 의해 결정될 수 있다. 대안적으로, 네트워크 정체의 상태는 송신기에 의해 결정될 수 있다.

[0034] 일부 실시예에서, 통지 메시지는 수신기에 의해 생성될 수 있다. 통지 메시지는 수신기로부터 송신기로 송신될 수 있다. 통지 메시지는 지연의 변화가 지연 차이 문턱값을 초과하는 것으로 결정될 때 즉시 송신될 수 있다(예컨대, T2에서 송신된 메시지(215)). 통지 메시지는 간격의 끝보다 늦은 시점에 송신될 수 있다(예컨대, T9에서 송신된 메시지(216)). 통지 메시지는 정체 상태 또는 큐잉 지연의 증가 또는 감소를 나타내는 복수의 변수를 포함할 수 있다. 일부 실시예에서, 통지 메시지에 포함된 정보의 적어도 일부는 추가 정체 관리를 위해 송신기에 의해 사용된다.

[0035] 도 3은 예를 들어, 이더넷 링크와 같은 데이터 링크에서의 정체 관리 방법에 대한 흐름도를 도시한다. 이 흐름도는 초기 설정 단계, 빠른 시작 단계, 전환 단계, 및 조절 단계를 포함한다. 이 방법은 송신기(101)와 수신기(105) 간의 데이터 송신과 함께 도 1의 환경에서 구현될 수 있다.

[0036] 초기 설정 단계(310)에서, 송신기와 수신기는 서로의 클록 속도 및 회선 속도에 관한 정보를 공유할 수 있다. 이 정보를 통해 송신기와 수신기는 허용 가능한 지연을 결정할 수 있으므로 비정상적으로 큰 지연 값(예컨대, 특정 송신기와 수신기 간의 데이터 흐름에 대해 특정 문턱값을 초과하는 값)을 등록하여 정체 이벤트를 검출할 수 있다.

[0037] 빠른 시작 단계(320)에서, 송신기는 초기 속도로 데이터를 수신기에게 송신한다. 빠른 시작 단계(320)는 윈도우 성장 단계(window growth phase)가 선행될 수 있으며, 여기서 송신기는 데이터 패킷을 송신하기 위한 초기 윈도우 크기를 설정하고 네트워크 내에서 정체가 검출되지 않는 한 윈도우의 크기를 증가시킬 수 있다. 초기 윈도우 크기는 예를 들어, 회선 속도에 네트워크 내에서의 최소 전파 지연 시간을 곱한 대역폭 지연 곱으로 설정될 수 있다. 윈도우 크기가 증가함에 따라, 수신기는 예를 들어, 수신된 50 킬로바이트의 데이터마다 패킷을 송신기에게 송신함으로써 정체를 검출하지 않았음을 송신기에게 주기적으로 통지할 수 있다. 윈도우가 성장할 수 있는 최대 크기는 링크 속도에, 네트워크 내 전파 지연의 최댓값을 곱한 값에 의해 결정될 수 있다. 수신기는 송신기로부터 수신하는 데이터 패킷과 관련하여 단방향 지연의 급증을 관찰하여 정체 이벤트를 검출할 수 있다. 단방향 지연의 급증은 패킷의 왕복 시간보다 큰 지연에 대응할 수 있다. 예를 들어, 15 마이크로초 내지 25 마이크로초의 지연으로 인해 수신기가 정체 이벤트를 등록하게 할 수 있다. 일부 실시예에서, 수신기는 정체 이벤트를 등록하기 위해 최소 지연과 관련된 지연을 측정할 수 있다. 이들 실시예에서, 수신기는 문턱 지연을 구성할 수 있고, 문턱 지연을 초과하면 수신기는 검출된 지연을 정체 이벤트로서 등록할 수 있다.

- [0038] 송신기는 윈도우 크기를 기하급수적으로 증가시킬 수 있다. 송신기는 또한 선형적으로, 대수적으로, 또는 단계적으로 크기를 증가시킬 수 있다. 윈도우 크기는 문턱값 크기가 달성될 때까지 증가될 수 있다. 패킷은 비트를 사용하여 윈도우 크기를 증가시키도록 송신기에게 시그널링할 수 있다.
- [0039] 빠른 시작 단계(320) 내에서 윈도우 성장 후, 송신기는 초기 속도를 설정한다. 초기 속도는 예를 들어, 25 GBit/s 또는 100 GBit/s와 같은 회선 속도일 수 있다. 초기 속도는 예를 들어, 송신기와 수신기를 접속하는 물리적 네트워크의 유형, 송신기와 수신기 간의 물리적 거리, 그리고 송신기와 수신기의 동작 구성 등의 요인에 의해 영향을 받을 수 있다. 윈도우에 과부하가 걸리지 않는 한 송신기는 트래픽을 최고 속도로 송신할 수 있다. 수신기는 CNP 패킷을 사용하여 송신기와 통신할 수 있다. 예를 들어, 수신기는 수신기에게 패킷을 회선 속도로 계속 송신하게끔 알리도록 CNP-W 패킷을, 또는 정체 이벤트가 검출되면 CNP-R 패킷을 송신할 수 있다.
- [0040] 수신기가 정체 이벤트를 검출하면 시스템은 전환 단계(330)에 진입한다. 수신기는, 송신기가 데이터 송신 속도를 최소 속도로 낮추어야 함을 나타내는 비트를 가진 패킷을 송신기에게 송신할 수 있다. 최소 속도는 초기 속도보다 훨씬 작을 수 있다. 예를 들어, 최소 속도는 초기 속도보다 2 내지 4배(two to four orders) 더 작을 수 있다. 시스템은 하나 이상의 기간 동안 전환 상태를 유지할 수 있다. 기간(period)은 패킷의 왕복 시간(round trip time; RTT)에 기초할 수 있다. 예를 들어, 기간은 25 마이크로초일 수 있다. 이것은 시스템의 정체를 줄이기 위해 수행된다. 특정 수의 기간들 동안 최소 속도를 유지한 후, 시스템은 데이터 송신 속도를 초기 송신 속도의 절반으로 증가시킬 수 있다. 수신기는 또한 송신기와 수신기 간의 처리량을 측정할 수 있다. 처리량이 초기 속도의 절반의 일부(a portion of one-half) 미만인 경우, 수신기는 측정된 처리량의 절반으로 송신하도록 송신기에게 알리는 패킷을 송신기에게 송신할 수 있다. 송신기는 패킷을 수신할 때 송신 속도를 떨어뜨릴(drop) 수 있다.
- [0041] 시스템은 2개의 갱신 기간 동안 전환 단계에 머문 후 조절 단계(340)로 이동한다. 조절 단계는 피드백 제어 알고리즘과 가산적 증가 승산적 감소(AIMD)를 사용하여 데이터 송신 속도를 수정할 수 있다.
- [0042] 피드백 제어 알고리즘은 비례 적분(proportional integral; PI) 알고리즘일 수 있다. 피드백 제어 알고리즘은 또한 비례 알고리즘 또는 비례 적분 미분(proportional-integral-derivative; PID) 알고리즘을 사용하여 구현될 수 있다. PI 알고리즘은 효율성과 낮은 지연을 보장할 수 있다. 데이터 송신 속도의 계산은 현재 지연이 문턱값(예컨대, 최소 지연 값)으로부터 얼마나 멀리 떨어져 있는지 그리고 큐잉 지연이 현재 증가 또는 감소 추세인지에 기초하여 주기적으로 갱신될 수 있다. 알고리즘의 "비례적인" 양상은 지연이 길(high) 경우 알고리즘이 데이터 송신 속도를 감소시킬 가능성이 더 높은 반면, 알고리즘의 "적분" 양상은 큐잉 지연이 증가하는 추세인 경우 데이터 송신 속도를 감소시키도록 작용한다.
- [0043] 피드백 제어 루프는 다음 수학적식을 사용하여 설명될 수 있다.
- [0044] 
$$R(n+1) = R(n) - \alpha(\Delta(n) - \theta) - \beta(\Delta(n) - \Delta(n-1))$$
- [0045] 이 수학적식에서, 갱신된 속도  $R(n+1)$ 은 현재 속도  $R(n)$ 에서 비례항  $\alpha(\Delta(n) - \theta)$ 과 적분항  $\beta(\Delta(n) - \Delta(n-1))$ 을 뺀으로써 계산된다. 비례항은 상수에 의해 곱해진 현재 지연  $\Delta(n)$ 과 문턱 지연  $\theta$  간의 차이를 설명한다. 적분항은 증가 또는 감소 지연 추세를 이 수학적식에 통합하기 위해, 다른 상수에 의해 곱해진 현재 지연  $\Delta(n)$ 과 이전 지연  $\Delta(n-1)$  간의 차이를 설명한다.
- [0046] AIMD 알고리즘은 피드백 제어 알고리즘과 함께 동작하여 시스템의 공정성을 보장한다. AIMD 알고리즘은, 지연이 감소하면 데이터 송신율에 대해 가산적 증가를 수행할 수 있고, 지연이 증가하면 승산적 감소를 수행할 수 있다. 이 프로세스는 데이터 흐름들 간의 링크에 공정성을 추가하고 지연을 제어하는 데 필요한 시간 규모(time scale)보다 더 긴 시간 규모에 걸쳐 수행될 수 있다. 예를 들어, 시스템은 정체 이벤트가 수신기에 의해 검출될 때까지 상이한 데이터 흐름들에 대한 속도들에 대해 가산적 증가를 수행할 수 있다. 이 시점에서, 이 시스템은 정체를 중지하기 위해 데이터 속도에 대해 승산적 감소를 수행할 수 있다. 이 프로세스는 흐름들이 동일한 데이터 속도를 가질 때까지 반복될 수 있다.
- [0047] 일부 실시예에서, 시스템이 조절 단계에 있는 동안 가산적 증가가 지속적으로 수행될 수 있다. 이것이 발생하는 동안 수신기는 예를 들어, CNP-R 패킷과 같은 패킷을 송신기에게 송신하여 속도를 가산적으로 증가시킬 수 있다. 승산적 감소는 주기적으로 수행될 수 있다. 일부 실시예에서, 시스템은 특정 수의 조절 주기 후에 승산적 감소를 수행할 수 있다. 다른 실시예에서, 승산적 증가는 예를 들어, "동전 던지기(coin toss)"와 같은 확률적

이벤트에 응답하여 수행될 수 있다. 시스템은 "동전"에 승산적 감소 확률을 할당하고 패킷이 도착할 때마다 "동전 던지기(flip the coin)"를 하여 해당 주기 동안 속도에 대해 승산적 감소를 수행할지 여부를 결정할 수 있다. 예를 들어, 승산적 감소 확률이 30%인 경우, 10개의 패킷 송신 중 3번의 승산적 감소가 수행될 것으로 예상될 수 있다.

[0048] 시스템은 피드백 제어 루프와 AIMD 루프를 공동으로 수행할 수 있다. AIMD 프로세스는 장기간에 걸쳐 공정성이 달성될 수 있으므로 피드백 제어 루프보다 낮은 주파수에서 수행될 수 있다. 제어 루프를 수행하는 것보다 적은 빈도로 AIMD를 주기적으로 수행하면 큐에 도착하는 패킷에 대해 중단을 줄이면서 데이터 흐름의 균형이 다시 맞추어지게 한다.

[0049] 조절 단계(340)는 데이터 송신이 초기 데이터 속도로 복귀될 때까지 계속될 수 있다. 시스템이 여러 기간 동안 초기 속도로 복귀하면, 시스템은 빠른 시작 프로세스(320)를 다시 시작할 수 있다.

[0050] 도 4는 통신을 위한 효율 그래프(400)를 도시한다. 이 도면은 링크 계층에 대한 패킷 부하가 증가함에 따라 굿풋(goodput)(단위 시간당 전달되는 유용한 정보 비트 수)과 정체된 버퍼 크기(420)가 어떻게 증가하는지 보여준다. 이 도면은 데이터 부하가 0으로부터 증가함에 따라 굿풋(410)도 선형적으로 증가함을 보여준다. 패킷 부하(450)가 용량(440)을 초과하여 증가할 때, 곡선은 선형에서 평평하게 천이하여 그래프에서 무릎(knee)을 형성한다. 이것은 최대 굿풋과 무시할 수 있는(negligible) 큐를 갖는 시스템의 최적 동작 지점(430)이다. 하단 그래프는 이 지점 이후에 시스템에 의해 모든 패킷이 송신될 수는 없기 때문에 패킷 부하가 무릎을 지나 증가함에 따라 큐가 구축되는 것을 보여준다. 이 그래프는 큐가 구축됨에 따라 굿풋(410)이 큰 부하에서도 높게 유지됨을 보여준다. 이 시스템은 작은 대기 큐를 사용해 최적으로 동작될 수 있다. 새 패킷이 큐에 추가되기 전에, 큐잉되어 있는(enqueued) 패킷이 송신되지 않을 수 있으므로 큐는 부하가 증가함에 따라 크기가 기하급수적으로 증가할 수 있다. 특정 부하에서, 이 시스템이 이러한 큰 큐를 유지 관리할 수 없기 때문에 패킷이 드롭될 수 있다. 이 그래프는 개시된 방법에서, 이 시스템이 무릎 가까이에서 동작함을 보여준다.

[0051] 도 5는 두 방법 모두를 사용해 수행된 실험의 데이터를 사용하여 DCQCN과 개시된 방법의 지연 비교(500)를 도시한다. 도 5의 실험에서, DCQCN 및 개시된 방법 모두에 대해, 20,000개의 데이터 흐름이 포아송 프로세스로서 수신기에 도달한다. 흐름들은 다양한 왕복 시간(RTT)을 가진 16개의 상이한 포트들로부터 오며 모두 수신기의 목적지를 갖는다. 패킷이 수신기에 도달하는 속도를 제어하여 송신기와 수신기 간의 링크가 20%, 40%, 60%, 80% 만큼 부하되는 실험을 수행한다. DCQCN과 이 방법 모두에서 이러한 16개 포트에 대해, 이더넷 프레임의 최소 일시 중지 기간, 평균 일시 중지 기간 및 최대 일시 중지 기간이 측정되었다. 이 그래프에서 볼 수 있는 바와 같이, 높은 부하에서 DCQCN은 개시된 방법에 비해 성능이 좋지 않다. 실제로 DCQCN의 일시 중지 기간은 고부하에서 기하급수적으로 더 나빠진다. 이에 비해, 고부하에서, 개시된 방법은 고부하에서도 일시 정지 프레임이 생성되지 않는다.

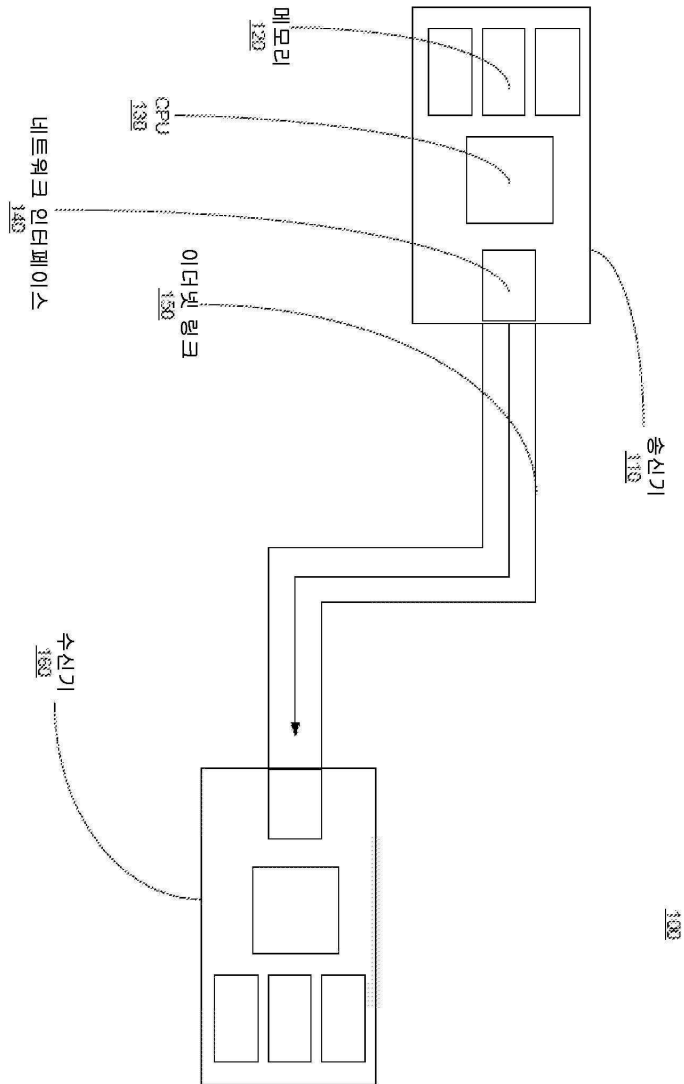
[0052] 도 6a 및 6b는 도 5의 실험으로부터 네트워크에서 데이터 흐름의 예시를 도시한다. 도 6a의 예시(600)는 개시된 방법으로부터의 처리량을 보여주는 반면, 도 6b의 예시(650)는 DCQCN으로부터의 처리량을 보여준다. 예시(650)는 상당한 희생 흐름(victim flows)을 보여준다. 희생 흐름 즉, 감소된 처리량 흐름은 데이터 센터 네트워크에 걸쳐 확산된 정체를 나타낸다. 따라서, 도 6a 및 6b는 DCQCN이 사용될 때 상당한 정체 확산을 나타내지만 개시된 방법이 사용될 때는 무시할 수 있는 정체를 나타낸다.

[0053] 도 7a 및 도 7b는 개시된 방법(700)과 DCQCN(750) 간의 큐 점유의 비교를 도시한다. 이 그래프는 DCQCN보다, 시스템을 위한 큐에 더 적은 데이터가 있음을 보여준다. 도 7a 및 도 7b의 차트는 실험 결과를 반영한다. 이 실험에서, 3-계층 데이터 센터 네트워크에 640개의 노드가 있다. 각 노드는 한 쌍의 랙 상단형(top-of-rack; ToR) 스위치에 접속된 2개의 25Gbps 링크를 갖는다. 노드들 간의 왕복 전파 지연은 12 마이크로초이다. PFC가 활성화된다. 640개의 노드는 한 쌍의 ToR 스위치 아래에 있는 32개 노드의 한 그룹에 데이터 패킷을 지속적으로 송신한다. 32개의 노드 각각은 다른 639개의 노드에도 데이터 패킷을 송신한다. 이 실험은 시뮬레이션 중간에 32개 노드 중 하나쪽으로는 하나의 링크를 제거하고 링크 제거가 다른 서버에 미치는 영향을 관찰한다. 도 7a 및 7b는 개시된 방법과 DCQCN 하에서 이 실험으로부터 정체된 링크의 큐 점유를 비교하고, 개시된 방법이 큐 점유를 DCQCN 하에서의 점유의 단지 10분의 1이 되게 감소시키는 것을 보여준다.

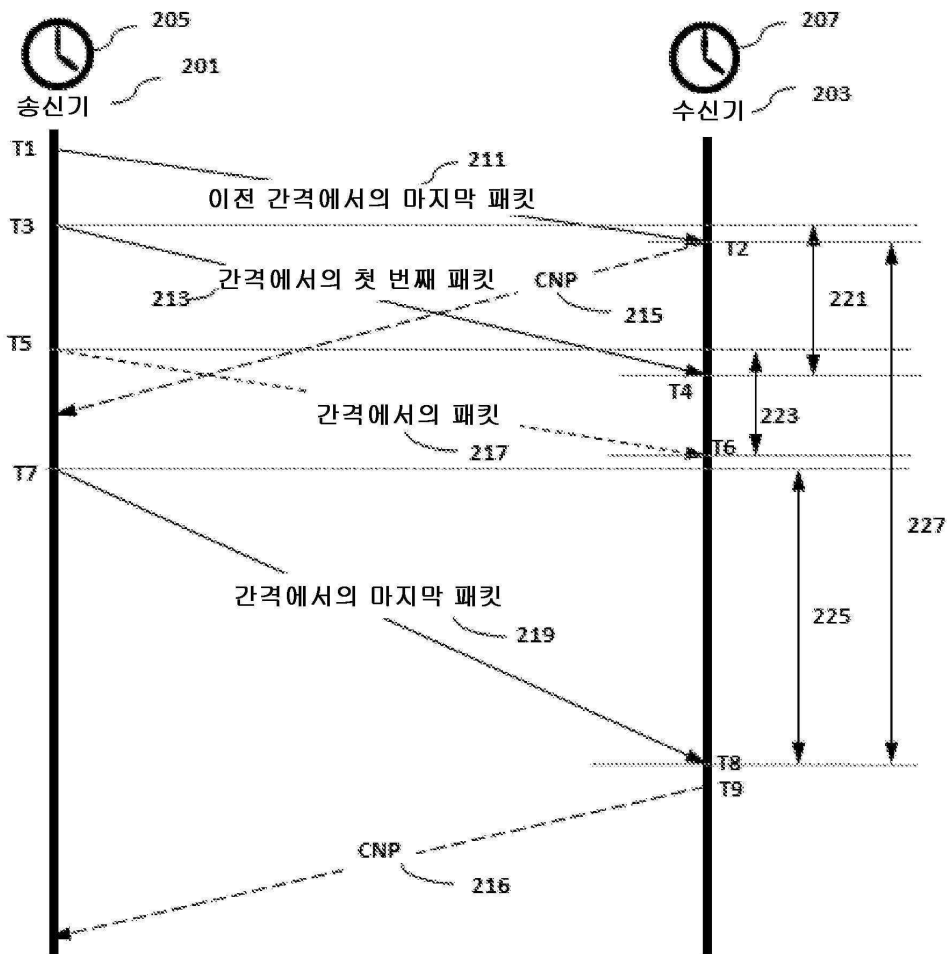


도면

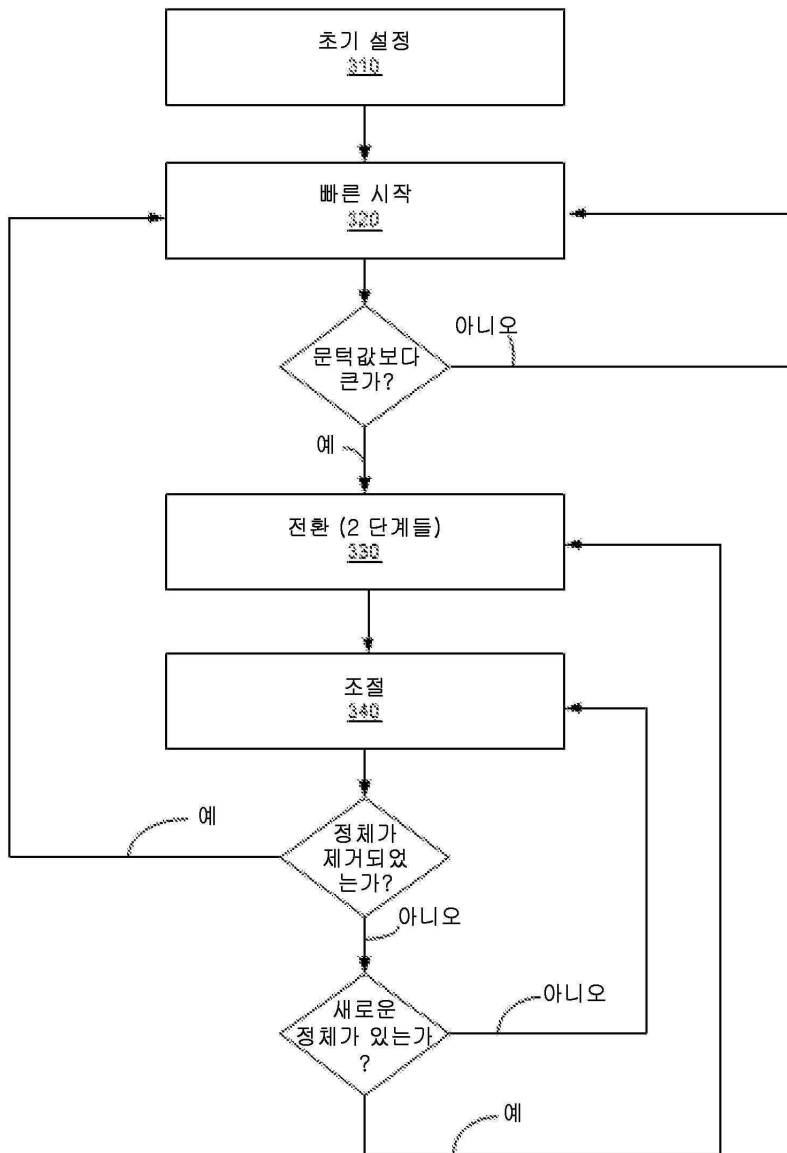
도면1



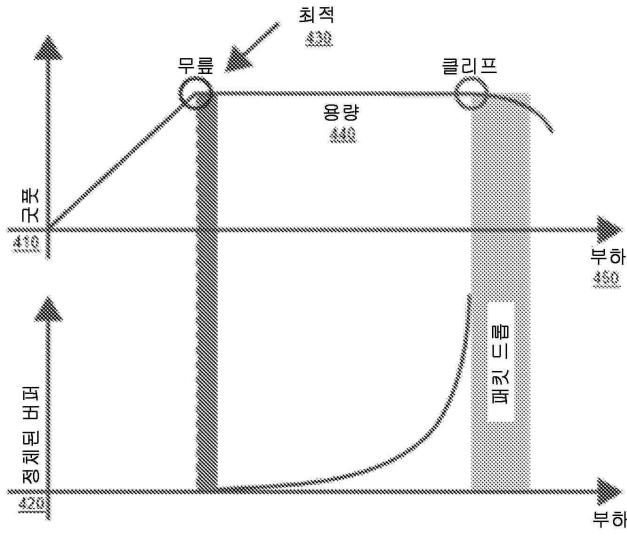
도면2



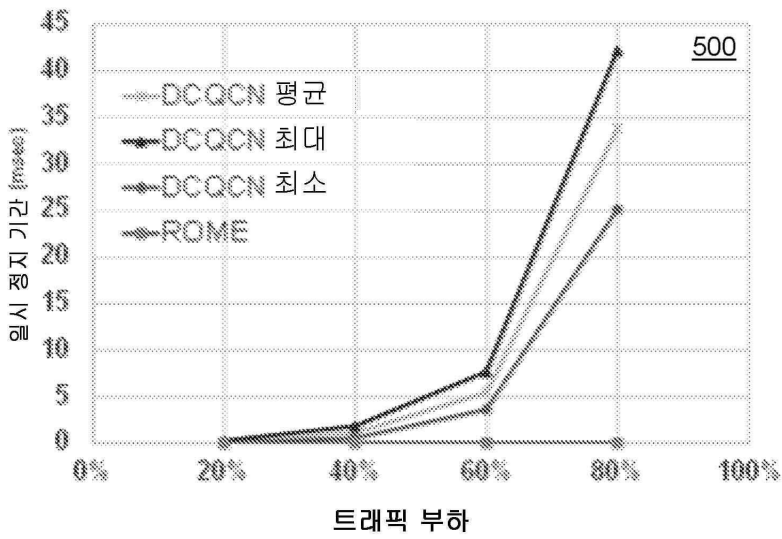
도면3



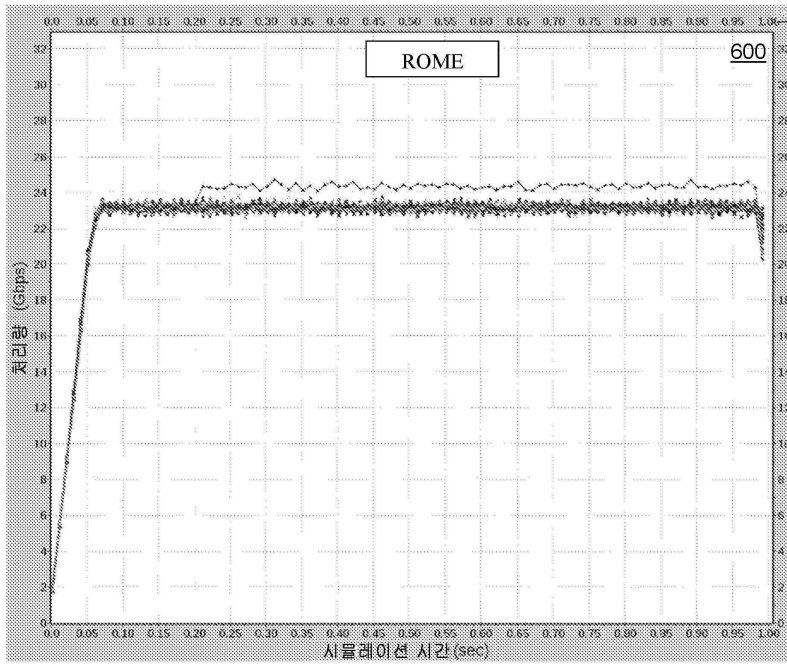
도면4



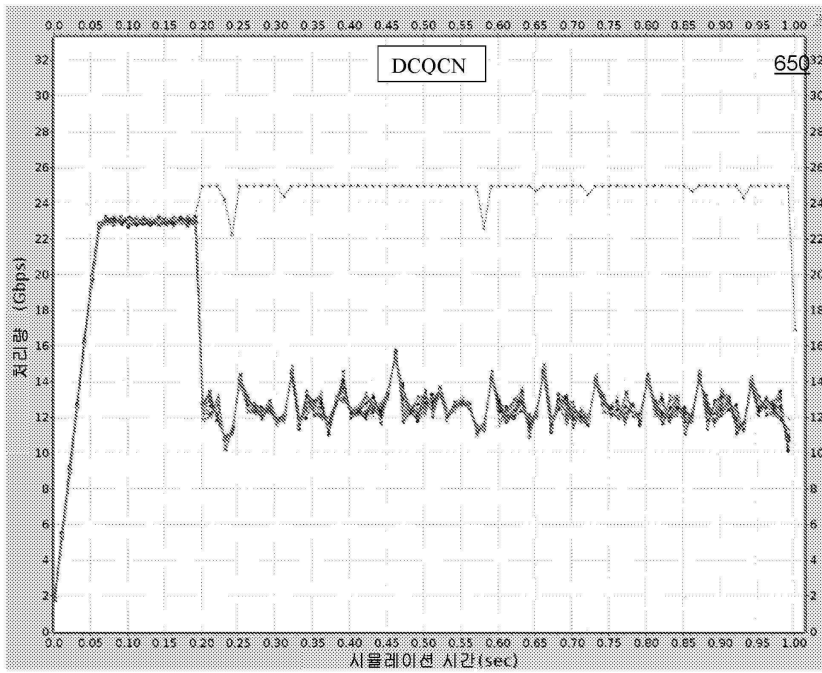
도면5



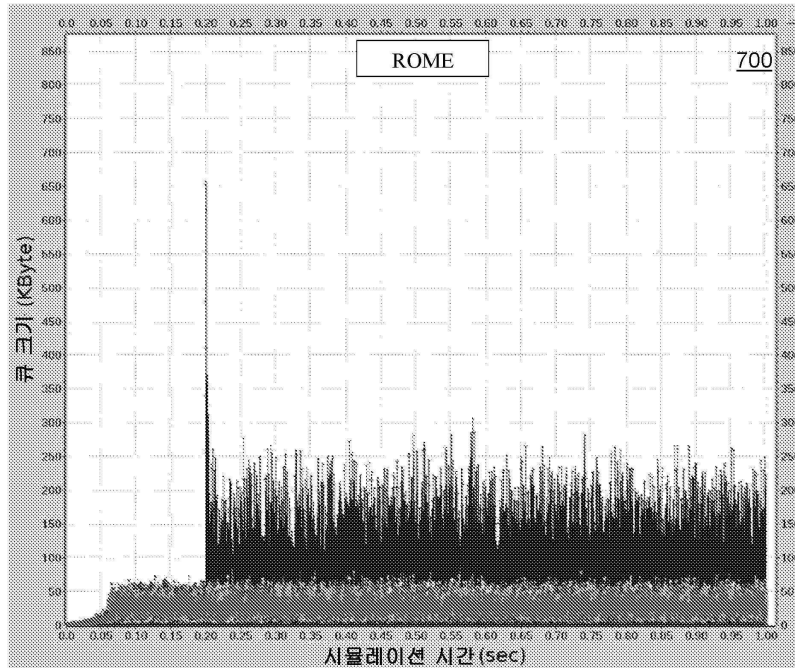
도면6a



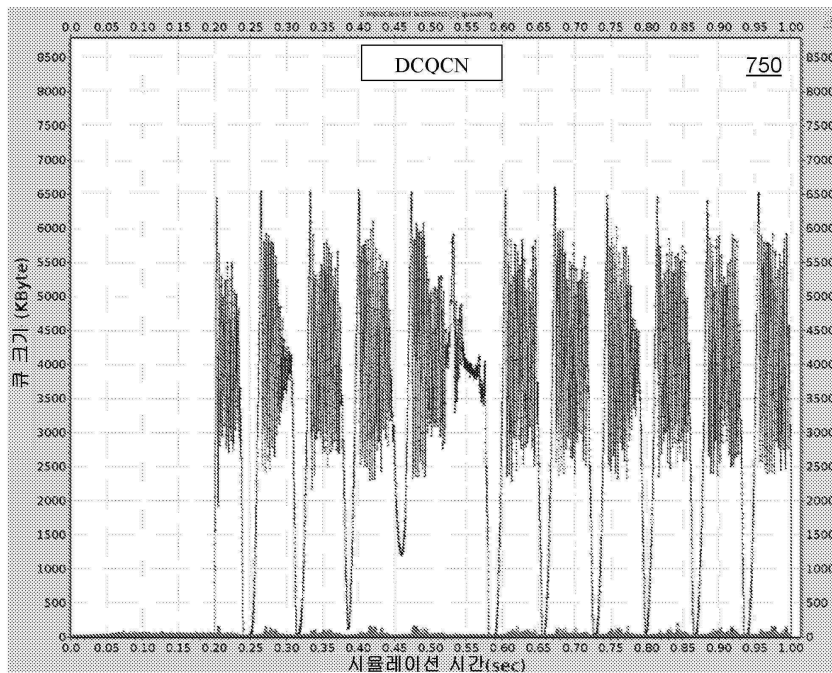
도면6b



도면7a



도면7b



【심사관 직권보정사항】

【직권보정 1】

【보정항목】 청구범위

【보정세부항목】 청구항 7

【변경전】

제1항에 있어서,

a) 문턱 수의 기간 동안 상기 데이터 송신 속도를 조정함으로써 적어도 상기 데이터 송신 초기 속도를 획득하는 단계; 및

b) 정체 이벤트에 대해 상기 네트워크를 모니터링하는 단계

를 더 포함하는, 정체를 관리하는 컴퓨터 구현 방법.

**【변경후】**

제1항에 있어서,

a) 문턱 수의 기간 동안 상기 데이터 송신 속도를 조정함으로써 적어도 상기 초기 데이터 송신 속도를 획득하는 단계; 및

b) 정체 이벤트에 대해 상기 네트워크를 모니터링하는 단계

를 더 포함하는, 정체를 관리하는 컴퓨터 구현 방법.

**【직권보정 2】**

**【보정항목】** 청구범위

**【보정세부항목】** 청구항 16

**【변경전】**

네트워크에서 네트워크 정체를 관리하기 위한 컴퓨터 구현 방법에 있어서,

a) 데이터 속도를 초기 데이터 속도로 설정하는 단계와, 수신기가 정체 이벤트를 검출할 때까지 송신기와 수신기 간에 데이터 송신을 수행하는 단계를 포함하는 빠른 시작 단계(fast start stage);

b) 상기 데이터 속도를 제1 기간 동안 최소 데이터 속도로 감소시킨 다음, 상기 데이터 속도를 제2 기간 동안 상기 초기 속도의 절반으로 증가시키는 단계를 포함하는 전환 단계; 및

c) 피드백 제어 루프와 가산적 증가 승산적 감소(additive increase multiplicative decrease; AIMD) 알고리즘의 조합을 수행하여 상기 데이터 속도를 수정하는 단계를 포함하는 조절 단계 - 상기 피드백 제어 루프는 제1 주파수에서 갱신된 데이터 송신 속도를 주기적으로 계산하는 데 사용되고, 상기 AIMD 알고리즘은 제2 주파수에서 상기 데이터 송신 속도를 조정하는 데 사용되며, 상기 제2 주파수는 상기 제1 주파수보다 낮음 -

를 포함하는, 네트워크 정체를 관리하기 위한 컴퓨터 구현 방법.

**【변경후】**

네트워크에서 네트워크 정체를 관리하기 위한 컴퓨터 구현 방법에 있어서,

a) 데이터 속도를 초기 데이터 속도로 설정하는 단계와, 수신기가 정체 이벤트를 검출할 때까지 송신기와 수신기 간에 데이터 송신을 수행하는 단계를 포함하는 빠른 시작 단계(fast start stage);

b) 상기 데이터 속도를 제1 기간 동안 최소 데이터 속도로 감소시킨 다음, 상기 데이터 속도를 제2 기간 동안 상기 초기 데이터 속도의 절반으로 증가시키는 단계를 포함하는 전환 단계; 및

c) 피드백 제어 루프와 가산적 증가 승산적 감소(additive increase multiplicative decrease; AIMD) 알고리즘의 조합을 수행하여 상기 데이터 속도를 수정하는 단계를 포함하는 조절 단계 - 상기 피드백 제어 루프는 제1 주파수에서 갱신된 데이터 송신 속도를 주기적으로 계산하는 데 사용되고, 상기 AIMD 알고리즘은 제2 주파수에서 상기 데이터 송신 속도를 조정하는 데 사용되며, 상기 제2 주파수는 상기 제1 주파수보다 낮음 -

를 포함하는, 네트워크 정체를 관리하기 위한 컴퓨터 구현 방법.

**【직권보정 3】**

**【보정항목】** 청구범위

**【보정세부항목】** 청구항 24

**【변경전】**

네트워크에서 네트워크 정체를 관리하기 위한 시스템에 있어서, 상기 시스템은 복수의 컴퓨팅 노드를 포함하고, 상기 컴퓨팅 노드는,

a) 하나 이상의 이전에 획득된 지연 값 및 명령어의 세트를 저장하기 위한 메모리; 및

b) 상기 명령어의 세트를 실행하도록 구성된 하나 이상의 프로세서

를 포함하고, 상기 명령어의 세트는 적어도,

- i) 데이터 속도를 초기 데이터 속도로 설정하는 단계와, 수신기가 정체 이벤트를 검출할 때까지 송신기와 수신기 간에 데이터 송신을 수행하는 단계를 포함하는 빠른 시작 단계;
- ii) 상기 데이터 속도를 제1 기간 동안 최소 데이터 속도로 감소시킨 다음, 상기 데이터 속도를 제2 기간 동안 상기 초기 속도의 절반으로 증가시키는 단계를 포함하는 전환 단계; 및
- iii) 피드백 제어 루프와 가산적 증가 승산적 감소(additive increase multiplicative decrease; AIMD) 알고리즘의 조합을 수행하여 상기 데이터 속도를 수정하는 단계를 포함하는 조절 단계 - 상기 피드백 제어 루프는 제1 주파수에서 갱신된 데이터 송신 속도를 주기적으로 계산하는 데 사용되고, 상기 AIMD 알고리즘은 제2 주파수에서 상기 데이터 송신 속도를 조정하는 데 사용되며, 상기 제2 주파수는 상기 제1 주파수보다 낮음 -

를 구현하기 위한 것인, 네트워크 정체를 관리하기 위한 시스템.

**【변경후】**

네트워크에서 네트워크 정체를 관리하기 위한 시스템에 있어서, 상기 시스템은 복수의 컴퓨팅 노드를 포함하고, 상기 컴퓨팅 노드는,

- a) 하나 이상의 이전에 획득된 지연 값 및 명령어의 세트를 저장하기 위한 메모리; 및
- b) 상기 명령어의 세트를 실행하도록 구성된 하나 이상의 프로세서

를 포함하고, 상기 명령어의 세트는 적어도,

- i) 데이터 속도를 초기 데이터 속도로 설정하는 단계와, 수신기가 정체 이벤트를 검출할 때까지 송신기와 수신기 간에 데이터 송신을 수행하는 단계를 포함하는 빠른 시작 단계;
- ii) 상기 데이터 속도를 제1 기간 동안 최소 데이터 속도로 감소시킨 다음, 상기 데이터 속도를 제2 기간 동안 상기 초기 데이터 속도의 절반으로 증가시키는 단계를 포함하는 전환 단계; 및
- iii) 피드백 제어 루프와 가산적 증가 승산적 감소(additive increase multiplicative decrease; AIMD) 알고리즘의 조합을 수행하여 상기 데이터 속도를 수정하는 단계를 포함하는 조절 단계 - 상기 피드백 제어 루프는 제1 주파수에서 갱신된 데이터 송신 속도를 주기적으로 계산하는 데 사용되고, 상기 AIMD 알고리즘은 제2 주파수에서 상기 데이터 송신 속도를 조정하는 데 사용되며, 상기 제2 주파수는 상기 제1 주파수보다 낮음 -

를 구현하기 위한 것인, 네트워크 정체를 관리하기 위한 시스템.