



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2024-0132673
(43) 공개일자 2024년09월04일

(51) 국제특허분류(Int. Cl.)
H04L 43/026 (2022.01) H04L 43/16 (2022.01)
(52) CPC특허분류
H04L 43/026 (2022.05)
H04L 43/16 (2022.05)
(21) 출원번호 10-2023-0025737
(22) 출원일자 2023년02월27일
심사청구일자 2023년02월27일

(71) 출원인
이화여자대학교 산학협력단
서울특별시 서대문구 이화여대길 52 (대현동, 이화여자대학교)
(72) 발명자
양대현
서울특별시 서초구 서초중앙로 200, 17동 901호
장룡호
인천광역시 미추홀구 인하로 100, 1008호
(뒷면에 계속)
(74) 대리인
한상수

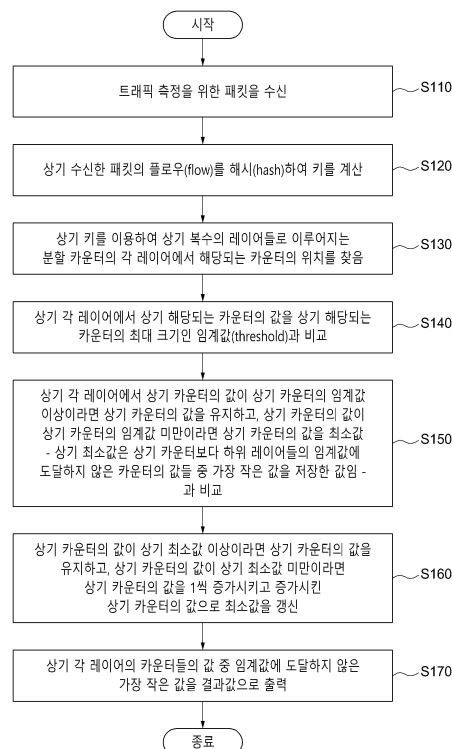
전체 청구항 수 : 총 10 항

(54) 발명의 명칭 스케치 기반의 트래픽 측정이 가능한 고속 네트워크 스위치, 및 트래픽 측정 방법

(57) 요약

본 발명의 일실시예에 따른 복수의 레이어들로 이루어지는 분할 카운터를 구비한 고속 네트워크 스위치에 의해 스케치 기반의 트래픽을 측정하는 방법은, 트래픽 측정을 위한 패킷을 수신하는 단계; 상기 수신한 패킷의 플로우(flow)를 해시(hash)하여 키를 계산하는 단계; 상기 키를 이용하여 상기 복수의 레이어들로 이루어지는 분할 카운터의 각 레이어에서 해당되는 카운터의 위치를 찾는 단계; 상기 각 레이어에서 상기 해당되는 카운터의 값을 상기 해당되는 카운터의 최대 크기인 임계값(threshold)과 비교하는 단계; 상기 각 레이어에서 상기 카운터의 값이 상기 카운터의 임계값 이상이라면 상기 카운터의 값을 유지하고, 상기 카운터의 값이 상기 카운터의 임계값 미만이라면 상기 카운터의 값을 최소값 - 상기 최소값은 상기 카운터보다 하위 레이어들의 임계값에 도달하지 않은 카운터의 값들 중 가장 작은 값을 저장한 값임 - 과 비교하는 단계; 상기 카운터의 값이 상기 최소값 이상이라면 상기 카운터의 값을 유지하고, 상기 카운터의 값이 상기 최소값 미만이라면 상기 카운터의 값을 1씩 증가시키고 증가시킨 상기 카운터의 값으로 최소값을 갱신하는 단계; 상기 각 레이어의 카운터들의 값들 중 임계값에 도달하지 않은 가장 작은 값을 결과값으로 출력하는 단계;를 포함한다.

대표도 - 도1



할 카운터의 각 레이어에서 해당되는 카운터의 위치를 찾는 단계를 포함하고, 상기 각 레이어에서 해당되는 카운터의 위치를 찾는 단계는: 상기 각 레이어에서 상기 해당되는 카운터의 값을 상기 카운터의 최대 크기인 임계값(threshold)과 비교하는 단계; 상기 각 레이어에서 상기 카운터의 값이 상기 카운터의 임계값 이상이라면 상기 카운터의 값을 유지하고, 상기 카운터의 값이 상기 카운터의 임계값 미만이라면 상기 카운터의 값을 최소값 - 상기 최소값은 상기 카운터보다 하위 레이어들의 임계값에 도달하지 않은 카운터의 값들 중 가장 작은 값을 저장한 값임 - 과 비교하는 단계; 상기 카운터의 값이 상기 최소값 이상이라면 상기 카운터의 값을 유지하고, 상기 카운터의 값이 상기 최소값 미만이라면 상기 카운터의 값을 1씩 증가시키고 증가시킨 상기 카운터의 값으로 최소값을 갱신하는 단계; 및 상기 각 레이어의 카운터들의 값 중 임계값에 도달하지 않은 가장 작은 값을 결과값으로 출력하는 단계를 포함할 수 있다.

(72) 발명자

정창훈

인천광역시 서구 비즈니스로 165, 603동 2103호

김시안

서울특별시 양천구 목동중앙북로16길 28, 201호

이 발명을 지원한 국가연구개발사업

과제고유번호	1711166713
과제번호	2016K1A1A2912757
부처명	과학기술정보통신부
과제관리(전문)기관명	한국연구재단
연구사업명	집단연구지원
연구과제명	빅데이터 기반의 DDoS공격의 분석, 모델링, 방어 기술 연구
기 여 율	50/100
과제수행기관명	이화여자대학교
연구기간	2022.03.01 ~ 2022.06.30

이 발명을 지원한 국가연구개발사업

과제고유번호	1711166443
과제번호	2020R1A2C2009372
부처명	과학기술정보통신부
과제관리(전문)기관명	한국연구재단
연구사업명	개인기초연구(과기정통부)
연구과제명	스케치 기반의 실시간 플로우별 네트워크 트래픽 측정 및 모니터링 시스템
기 여 율	45/100
과제수행기관명	이화여자대학교
연구기간	2022.03.01 ~ 2023.02.28

이 발명을 지원한 국가연구개발사업

과제고유번호	1711174090
과제번호	00155966
부처명	과학기술정보통신부
과제관리(전문)기관명	정보통신기획평가원
연구사업명	인공지능융합혁신인재양성
연구과제명	인공지능융합혁신인재양성(이화여자대학교)
기 여 율	5/100
과제수행기관명	이화여자대학교산학협력단
연구기간	2022.07.01 ~ 2022.12.31

명세서

청구범위

청구항 1

복수의 레이어들로 이루어지는 분할 카운터를 구비한 고속 네트워크 스위치에 의해 스케치 기반의 트래픽을 측정하는 방법으로서,

트래픽 측정을 위한 패킷을 수신하는 단계;

상기 수신한 패킷의 플로우(flow)를 해시(hash)하여 키를 계산하는 단계; 및

상기 키를 이용하여 상기 복수의 레이어들로 이루어지는 분할 카운터의 각 레이어에서 해당되는 카운터의 위치를 찾는 단계를 포함하고,

상기 각 레이어에서 해당되는 카운터의 위치를 찾는 단계는:

상기 각 레이어에서 상기 해당되는 카운터의 값을 상기 카운터의 최대 크기인 임계값(threshold)과 비교하는 단계;

상기 각 레이어에서 상기 카운터의 값이 상기 카운터의 임계값 이상이라면 상기 카운터의 값을 유지하고, 상기 카운터의 값이 상기 카운터의 임계값 미만이라면 상기 카운터의 값을 최소값 - 상기 최소값은 상기 카운터보다 하위 레이어들의 임계값에 도달하지 않은 카운터의 값들 중 가장 작은 값을 저장한 값임 - 과 비교하는 단계;

상기 카운터의 값이 상기 최소값 이상이라면 상기 카운터의 값을 유지하고, 상기 카운터의 값이 상기 최소값 미만이라면 상기 카운터의 값을 1씩 증가시키고 증가시킨 상기 카운터의 값으로 최소값을 갱신하는 단계; 및

상기 각 레이어의 카운터들의 값 중 임계값에 도달하지 않은 가장 작은 값을 결과값으로 출력하는 단계를 포함하는 것을 특징으로 하는, 방법.

청구항 2

제 1항에 있어서,

상기 임계값과 비교하는 단계는, 상기 각 레이어들 중 최하위 레이어에서부터 최상위 레이어로 순차적으로 수행되는 것을 특징으로 하는, 방법.

청구항 3

제 1항에 있어서,

상기 패킷의 플로우는 5-튜플(5-tuple)의 구성요소 중 적어도 1개 이상을 포함하는 것을 특징으로 하는, 방법.

청구항 4

제 3항에 있어서,

상기 키를 계산하는 단계에서, 해시 계산시 사용되는 상기 5-튜플의 bit값은 사용자가 임의로 설정이 가능한 것을 특징으로 하는, 방법.

청구항 5

제 1항에 있어서,

상기 분할 카운터의 레이어는 최소 1 bit 크기의 카운터를 가지는 것을 특징으로 하는, 방법.

청구항 6

제 1항에 있어서,

상기 복수의 레이어들은, 상위 레이어에서 하위 레이어로 갈수록 각 카운터의 크기가 일정한 비율로 줄어드는 것을 특징으로 하는, 방법.

청구항 7

제 6항에 있어서,

상기 복수의 레이어들의 개수와 상기 각 카운터의 개수는 사용자가 임의로 설정이 가능한 것을 특징으로 하는, 방법.

청구항 8

트래픽 측정을 하는 고속 네트워크 스위치로서,

패킷을 수신하는 수신부;

상기 수신한 패킷의 플로우(flow)를 해시(hash)하여 키를 계산하는 프로세서; 및

트래픽 측정을 위한 복수의 레이어들로 이루어지는 분할 카운터를 포함하고,

상기 고속 네트워크 스위치는 상기 키를 이용하여 상기 복수의 레이어들로 이루어지는 분할 카운터의 각 레이어에서 해당되는 카운터의 위치를 찾도록 이루어지고,

상기 고속 네트워크 스위치는 각 레이어에서 해당하는 카운터의 위치를 찾는 경우,

상기 각 레이어에서 상기 해당되는 카운터의 값을 상기 해당되는 카운터의 최대 크기인 임계값(threshold)과 비교하고,

상기 각 레이어에서 상기 카운터의 값이 상기 카운터의 임계값 이상이라면 상기 카운터의 값을 유지하고, 상기 카운터의 값이 상기 카운터의 임계값 미만이라면 상기 카운터의 값을 최소값 - 상기 최소값은 상기 카운터보다 하위 레이어들의 임계값에 도달하지 않은 카운터의 값들 중 가장 작은 값을 저장한 값임 - 과 비교하고,

상기 카운터의 값이 상기 최소값 이상이라면 상기 카운터의 값을 유지하고, 상기 카운터의 값이 상기 최소값 미만이라면 상기 카운터의 값을 1씩 증가시키고 증가시킨 상기 카운터의 값으로 최소값을 갱신하고,

상기 각 레이어의 카운터들의 값 중 임계값에 도달하지 않은 가장 작은 값을 결과값으로 출력하도록 이루어지는 것을 특징으로 하는, 고속 네트워크 스위치.

청구항 9

제 8항에 있어서,

상기 해당되는 카운터의 값을 상기 해당되는 카운터의 최대 크기인 임계값과 비교할 때, 상기 각 레이어들 중 최하위 레이어에서부터 최상위 레이어로 순차적으로 수행되는 것을 특징으로 하는, 고속 네트워크 스위치.

청구항 10

제 8항에 있어서,

상기 패킷의 플로우는 5-튜플(5-tuple)의 구성요소 중 적어도 1개 이상을 포함하는 것을 특징으로 하는, 고속 네트워크 스위치.

발명의 설명

기술 분야

[0001] 본 발명은 스케치 기반의 트래픽 측정이 가능한 고속 네트워크 스위치 및 트래픽 측정 방법에 관한 것으로, 더욱 상세하게는 메모리의 효율성을 위한 스케치 기반의 알고리즘으로 고속 네트워크 스위치에서 트래픽을 측정하는 방법 및 트래픽 측정이 가능한 네트워크 스위치에 관한 것이다.

배경 기술

[0002] 네트워크 스위치 레벨에서의 트래픽 측정은 로드 밸런싱(load balancing), 정체 제어(congestion control), 서비스 품질(quality of service), 이상 탐지(anomaly detection)에 중요한 역할을 한다. 네트워크 트래픽 측정의 종류에는 플로우(Source IP, Destination IP, Source Port, Destination Port, Protocol) 별 트래픽 측정, 헤비 히터 탐지(heavy hitter detection), 카디널리티(cardinality), 엔트로피(entropy) 등이 있으며, 정확한 모니터링을 위해선 세분화된 플로우 별 트래픽 측정이 필수적이다.

[0003] 플로우 별 트래픽 측정 방식에는 하드웨어 기반, 샘플링 기반, 스케치 기반 방식 등이 있다. TCAM을 사용하는 하드웨어 기반 측정 방식의 경우, 빠른 처리 속도를 보장하지만 고가이면서 적은 용량의 TCAM 때문에 확장성이 좋지 않다는 단점이 있다. 샘플링 기반 방식의 경우 샘플링 속도와 정확도 사이의 균형이 좋지 않으며, 샘플링 방식에 따라 높은 구현 복잡도가 필요할 수 있으므로 빠른 라우팅을 보장해야 하는 고속 네트워크 스위치에서 구현이 불가능하다는 단점이 있다. 스케치 기반 방식의 경우 고속 네트워크 스위치에서 구현이 가능하나, 고속 네트워크 스위치의 한정된 데이터 플레인 자원 내에서 트래픽을 측정하기 위해서는 스케치의 인코딩, 디코딩 알고리즘의 복잡도가 간단해야함은 물론, 스케치의 정확도 성능에 영향을 주는 네트워크 트래픽의 다양한 분포에 대해서도 정확하게 측정이 가능해야한다는 문제점이 존재했다.

선행기술문헌

특허문헌

[0005] (특허문헌 0001) 대한민국 공개특허공보 제2020-0028477호

발명의 내용

해결하려는 과제

[0006] 상기와 같은 문제점을 해결하기 위한 본 발명의 목적은, 메모리 접근을 최소화 하면서 카운터를 업데이트하는 새로운 인코딩 방식과, 메모리 낭비를 줄이고 네트워크의 트래픽 분포에 따라 공간을 유동적으로 활용할 수 있는 스케치 기반의 트래픽 측정이 가능한 고속 네트워크 스위치 및 트래픽 측정 방법을 제공하는 것이다.

[0007] 본 발명이 이루고자 하는 기술적 과제는 이상에서 언급한 기술적 과제로 제한되지 않으며, 언급되지 않은 또 다른 기술적 과제들은 아래의 기재로부터 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자에게 명확하게 이해될 수 있을 것이다.

과제의 해결 수단

[0009] 상기 기술적 과제를 달성하기 위한 본 발명의 일실시예에 따른 복수의 레이어들로 이루어지는 분할 카운터를 구비한 고속 네트워크 스위치에 의해 스케치 기반의 트래픽을 측정하는 방법은, 트래픽 측정을 위한 패킷을 수신하는 단계; 상기 수신한 패킷의 플로우(flow)를 해시(hash)하여 키를 계산하는 단계; 및 상기 키를 이용하여 상

기 복수의 레이어들로 이루어지는 분할 카운터의 각 레이어에서 해당되는 카운터의 위치를 찾는 단계를 포함하고, 상기 각 레이어에서 해당되는 카운터의 위치를 찾는 단계는: 상기 각 레이어에서 상기 해당되는 카운터의 값을 상기 해당되는 카운터의 최대 크기인 임계값(threshold)과 비교하는 단계; 상기 각 레이어에서 상기 카운터의 값이 상기 카운터의 임계값 이상이라면 상기 카운터의 값을 유지하고, 상기 카운터의 값이 상기 카운터의 임계값 미만이라면 상기 카운터의 값을 최소값 - 상기 최소값은 상기 카운터보다 하위 레이어들의 임계값에 도달하지 않은 카운터의 값들 중 가장 작은 값을 저장한 값임 - 과 비교하는 단계; 상기 카운터의 값이 상기 최소값 이상이라면 상기 카운터의 값을 유지하고, 상기 카운터의 값이 상기 최소값 미만이라면 상기 카운터의 값을 1씩 증가시키고 증가시킨 상기 카운터의 값으로 최소값을 갱신하는 단계; 및 상기 각 레이어의 카운터들의 값 중 임계값에 도달하지 않은 가장 작은 값을 결과값으로 출력하는 단계를 포함할 수 있다.

[0010] 본 발명의 실시예에 있어서, 상기 임계값과 비교하는 단계는, 상기 각 레이어들 중 최하위 레이어에서부터 최상위 레이어로 순차적으로 수행될 수 있다.

[0011] 본 발명의 실시예에 있어서, 상기 패킷의 플로우는 5-튜플(5-tuple)의 구성요소 중 적어도 1개 이상을 포함할 수 있다.

[0012] 본 발명의 실시예에 있어서, 상기 키를 계산하는 단계에서, 해시 계산시 사용되는 상기 5-튜플의 bit값은 사용자가 임의로 설정이 가능할 수 있다.

[0013] 본 발명의 실시예에 있어서, 상기 분할 카운터의 레이어는 최소 1 bit 크기의 카운터를 가질 수 있다.

[0014] 본 발명의 실시예에 있어서, 상기 복수의 레이어들은, 상위 레이어에서 하위 레이어로 갈수록 각 카운터의 크기가 일정한 비율로 줄어들 수 있다.

[0015] 본 발명의 실시예에 있어서, 상기 복수의 레이어들의 개수와 상기 각 카운터의 개수는 사용자가 임의로 설정이 가능하다.

[0016] 본 발명의 다른 실시예에 따른 트래픽 측정을 하는 고속 네트워크 스위치는, 패킷을 수신하는 수신부; 상기 수신한 패킷의 플로우(flow)를 해시(hash)하여 키를 계산하는 프로세서; 및 트래픽 측정을 위한 복수의 레이어들로 이루어지는 분할 카운터를 포함하고, 상기 고속 네트워크 스위치는 상기 키를 이용하여 상기 복수의 레이어들로 이루어지는 분할 카운터의 각 레이어에서 해당되는 카운터의 위치를 찾도록 이루어지고, 상기 고속 네트워크 스위치는 각 레이어에서 해당하는 카운터의 위치를 찾는 경우, 상기 각 레이어에서 상기 해당되는 카운터의 값을 상기 해당되는 카운터의 최대 크기인 임계값(threshold)과 비교하고, 상기 각 레이어에서 상기 카운터의 값이 상기 카운터의 임계값 이상이라면 상기 카운터의 값을 유지하고, 상기 카운터의 값이 상기 카운터의 임계값 미만이라면 상기 카운터의 값을 최소값 - 상기 최소값은 상기 카운터보다 하위 레이어들의 임계값에 도달하지 않은 카운터의 값들 중 가장 작은 값을 저장한 값임 - 과 비교하고, 상기 카운터의 값이 상기 최소값 이상이라면 상기 카운터의 값을 유지하고, 상기 카운터의 값이 상기 최소값 미만이라면 상기 카운터의 값을 1씩 증가시키고 증가시킨 상기 카운터의 값으로 최소값을 갱신하고, 상기 각 레이어의 카운터들의 값 중 임계값에 도달하지 않은 가장 작은 값을 결과값으로 출력하도록 이루어질 수 있다.

[0017] 본 발명의 실시예에 있어서, 상기 해당되는 카운터의 값을 상기 해당되는 카운터의 최대 크기인 임계값과 비교할 때, 상기 각 레이어들 중 최하위 레이어에서부터 최상위 레이어로 순차적으로 수행될 수 있다.

[0018] 본 발명의 실시예에 있어서, 상기 패킷의 플로우는 5-튜플(5-tuple)의 구성요소 중 적어도 1개 이상을 포함할 수 있다.

발명의 효과

[0020] 본 발명의 실시예에 따르면, 메모리 접근을 최소로 하면서 카운터를 업데이트하는 새로운 인코딩 방식과, 메모리 낭비를 줄이고 네트워크의 트래픽 분포에 따라 공간을 유동적으로 활용할 수 있는 분할 카운터 구조를 통해 고속 네트워크 스위치의 한정된 데이터 플레인 자원의 활용에 있어서 이점을 가질 뿐만 아니라, 다양한 네트워크 트래픽 분포에서의 측정 정확성도 향상된 고속 네트워크 스위치 및 이를 이용한 트래픽 측정 방법을 제공할 수 있다.

[0021] 본 발명의 효과는 상기한 효과로 한정되는 것은 아니며, 본 발명의 상세한 설명 또는 특허청구범위에 기재된 발명의 구성으로부터 추론 가능한 모든 효과를 포함하는 것으로 이해되어야 한다.

도면의 간단한 설명

- [0022] 도 1은 본 발명의 일실시예에 따른 복수의 레이어들로 이루어지는 분할 카운터를 구비한 고속 네트워크 스위치에 의해 스케치 기반의 트래픽을 측정하는 방법에 대한 개략적인 구성을 나타내는 흐름도이다.
- 도 2는 본 발명의 다른 실시예에 따른 고속 네트워크 스위치의 개략적인 구조도이다.
- 도 3은 본 발명의 일실시예에 따른 고속 네트워크 스위치에 의해 스케치 기반의 트래픽을 측정하는 방법에서 3개의 레이어를 가지는 분할 카운터에서의 인코딩과 디코딩의 예시를 나타내는 개략도이다.
- 도 4a, 도 4b, 도 4c, 도 4d는 본 발명의 일실시예에 따른 고속 네트워크 스위치에 의해 스케치 기반의 트래픽을 측정하는 방법에서, 카운터를 업데이트하는 과정의 4가지 시나리오를 나타내는 개략도이다.
- 도 5는 본 발명의 일실시예에 따른 고속 네트워크 스위치에 의해 스케치 기반의 트래픽을 측정하는 방법에서, 카운터 분포와 각 카운터에 따른 bit 손실을 기존의 다른 방법과 비교한 그래프이다.
- 도 6a는 본 발명의 일실시예에 따른 고속 네트워크 스위치에 의해 스케치 기반의 트래픽을 측정하는 방법에서의 bit 별 카운터 활용도를 레이어별로 비교한 결과를 나타내고, 도 6b는 FCM 방법의 bit 별 카운터 활용도를 레이어별로 비교한 결과를 나타내는 그래프이다.
- 도 7은 본 발명의 일실시예에 따른 고속 네트워크 스위치에 의해 스케치 기반의 트래픽을 측정하는 방법(CL-MU)과 다른 방법들의 (a)마우스 플로우 및 (b)엘리펀트 플로우 시 각각 발생하는 오류의 정도를 나타내는 그래프이다.
- 도 8a는 마우스 플로우 인코딩에서의 본 발명의 일실시예에 따른 트래픽을 측정하는 방법(CL-MU)과 다른 방법들의 생존율을 비교한 그래프이며, 도 8b는 엘리펀트 플로우 인코딩에서의 본 발명의 일실시예에 따른 트래픽을 측정하는 방법(CL-MU)과 다른 방법들의 생존율을 비교한 그래프이다.

발명을 실시하기 위한 구체적인 내용

- [0023] 이하에서는 첨부한 도면을 참조하여 본 발명을 설명하기로 한다. 그러나 본 발명은 여러 가지 상이한 형태로 구현될 수 있으며, 따라서 여기에서 설명하는 실시예로 한정되는 것은 아니다. 그리고 도면에서 본 발명을 명확하게 설명하기 위해서 설명과 관계없는 부분은 생략하였으며, 명세서 전체를 통하여 유사한 부분에 대해서는 유사한 도면 부호를 붙였다.
- [0024] 명세서 전체에서, 어떤 부분이 다른 부분과 "연결(접속, 접촉, 결합)"되어 있다고 할 때, 이는 "직접적으로 연결"되어 있는 경우뿐 아니라, 그 중간에 다른 부재를 사이에 두고 "간접적으로 연결"되어 있는 경우도 포함한다. 또한 어떤 부분이 어떤 구성요소를 "포함"한다고 할 때, 이는 특별히 반대되는 기재가 없는 한 다른 구성요소를 제외하는 것이 아니라 다른 구성요소를 더 구비할 수 있다는 것을 의미한다.
- [0025] 본 명세서에서 사용한 용어는 단지 특정한 실시예를 설명하기 위해 사용된 것으로, 본 발명을 한정하려는 의도가 아니다. 단수의 표현은 문맥상 명백하게 다르게 뜻하지 않는 한, 복수의 표현을 포함한다. 본 명세서에서, "포함하다" 또는 "가지다" 등의 용어는 명세서상에 기재된 특징, 숫자, 단계, 동작, 구성요소, 부품 또는 이들을 조합한 것이 존재함을 지정하려는 것이지, 하나 또는 그 이상의 다른 특징들이나 숫자, 단계, 동작, 구성요소, 부품 또는 이들을 조합한 것들의 존재 또는 부가 가능성을 미리 배제하지 않는 것으로 이해되어야 한다.
- [0026] 이하에서 본 발명의 트래픽 측정 방법에 따른 스케치 방식은 CL-MU(Count-Less Minimum Update)로 지칭하도록 한다. 이외의 다른 종래의 스케치 방식들은 표준 Count-Min방식의 CM, 피라미드 형태의 데이터 구조를 가진 Count-Min 업데이트 방식의 CL-CM(Count-Less Count-Min), 피라미드 형태의 데이터 구조를 가진 보수적 업데이트 방식의 CL-CU(Count-Less Conservative-Update), 논문 C. H. Song, P. G. Kannan, B. K. H. Low, and M. C. Chan, "Fcm-sketch: Generic network measurements with data plane support," in CoNext 2020. ACM, 2020, pp. 78-92에서 개시된 스케치 방식 기술을 지칭하는 FCM(Feed-forward Count-Min), 및 논문 T. Yang, J. Jiang, P. Liu, Q. Huang, J. Gong, Y. Zhou, R. Miao, X. Li, and S. Uhlig, "Elastic sketch: Adaptive and fast network-wide measurements," in ACM SIGCOMM 2018, 2018, pp. 561-575.에서 개시된 스케치 방식 기술을 지칭하는 Elastic으로 나타낸다.

- [0027] 이하에서는 도면을 참조하여 본 발명의 실시예들에 대하여 설명한다.
- [0029] 도 1은 본 발명의 일실시예에 따른 복수의 레이어들로 이루어지는 분할 카운터를 구비한 고속 네트워크 스위치에 의해 스케치 기반의 트래픽을 측정하는 방법에 대한 개략적인 구성을 나타내는 흐름도이다.
- [0030] 도 1을 참조하면, 복수의 레이어들로 이루어지는 분할 카운터를 구비한 고속 네트워크 스위치에 의해 스케치 기반의 트래픽을 측정하는 방법은 트래픽 측정을 위한 패킷을 수신(S110)하고, 상기 수신한 패킷의 플로우(flow)를 해시(hash)하여 키를 계산(S120)하고, 상기 키를 이용하여 상기 복수의 레이어들로 이루어지는 분할 카운터의 각 레이어에서 해당되는 카운터의 위치를 찾는다(S130). 단계 S130에서는 상기 각 레이어에서 상기 해당되는 카운터의 값을 상기 해당되는 카운터의 최대 크기인 임계값(threshold)과 비교(S140)하고, 상기 각 레이어에서 상기 카운터의 값이 상기 카운터의 임계값 이상이라면 상기 카운터의 값을 유지하고, 상기 카운터의 값이 상기 카운터의 임계값 미만이라면 상기 카운터의 값을 최소값 - 상기 최소값은 상기 카운터보다 하위 레이어들의 임계값에 도달하지 않은 카운터의 값들 중 가장 작은 값을 저장한 값임 - 과 비교(S150)한다. 이어서, 상기 카운터의 값이 상기 최소값 이상이라면 상기 카운터의 값을 유지하고, 상기 카운터의 값이 상기 최소값 미만이라면 상기 카운터의 값을 1씩 증가시키고 증가시킨 상기 카운터의 값으로 최소값을 갱신(S160)시키고, 상기 각 레이어의 카운터들의 값 중 임계값에 도달하지 않은 가장 작은 값을 결과값으로 출력(S170)할 수 있다.
- [0031] 단계 S140은 상기 각 레이어들 중 최하위 레이어에서부터 상위 레이어로 순차적으로 수행될 수 있다. 상기 패킷의 플로우는 5-튜플(5-tuple)의 구성요소 중 적어도 1개 이상을 포함할 수 있고, 상기 5-튜플은 TCP(Transmission Control Protocol), IP(Internet Protocol) 연결을 구성하는 5 개의 다른 값 집합으로, 소스 IP 주소(Source IP), 포트 번호(Source port), 대상 IP 주소(Destination IP), 포트 번호(Destination port), 및 사용중인 프로토콜(protocol)을 포함한다. 상기 패킷의 플로우는 상기 5-튜플의 둘 이상의 구성요소들의 조합으로도 이루어질 수 있다.
- [0032] 단계S120에서, 해시 계산시 레이어 별로 5-튜플의 bit 값 중에서 해시 계산에 사용되는 bit 값은 사용자가 임의로 설정이 가능하다. 또한, 상기 분할 카운터는 최하위 레이어의 카운터 크기로 최소 1 bit를 가질 수 있다.
- [0033] 상기 복수의 레이어들은, 상위 레이어에서 하위 레이어로 갈수록 사용자가 임의로 설정한 일정한 비율에 맞춰 각 카운터의 크기가 감소하며, 그에 따라, 상위 레이어에서 하위 레이어로 갈수록 카운터의 개수가 증가할 수 있다. 또한, 상기 복수의 레이어들의 총 개수도 상기 사용자가 설정할 수 있다.
- [0034] 단계 S170에서 상기 각 레이어의 카운터들의 값 중 임계값에 도달하지 않은 가장 작은 값으로 상기 갱신된 최소값을 출력할 수 있다.
- [0035] 도 2는 본 발명의 다른 실시예에 따른 고속 네트워크 스위치의 개략적인 구조도이다. 상기 트래픽 측정을 하는 고속 네트워크 스위치(200)는 수신부(210), 프로세서(220), 및 분할 카운터(230)를 포함할 수 있다. 본 발명의 일 실시예에 따르면, 바람직하게는 상기 트래픽측정을 하는 고속 네트워크 스위치(200)의 데이터 플레인(data plane)이 수신부(210), 프로세서(220), 및 분할 카운터(230)를 포함할 수 있다.
- [0036] 상기 수신부(210)는 플로우 측정을 위한 패킷을 수신하여 프로세서(220)로 전달한다. 상기 패킷의 플로우는 5-튜플을 구성하는 둘 이상의 구성요소들의 조합으로 이루어질 수 있으나, 이에 한정되는 것은 아니다.
- [0037] 프로세서(220)는 상기 패킷의 플로우를 해싱하기 위한 해시 함수를 포함하고, 상기 해시 함수를 이용하여 상기 패킷의 플로우의 키를 계산할 수 있다. 상기 해시 함수는 상기 복수의 레이어의 카운터의 위치를 찾기 위하여 상기 복수의 레이어 각각에 대응하여 존재할 수 있다. 상기 해시 함수는 필요에 따라 한 개 일 수도 있고, 복수 개 일 수도 있다.
- [0038] 상기 키 계산이 이루어지면, 복수의 레이어들로 이루어지는 분할 카운터(230)에서 상기 키를 통해 각 레이어에서 해당되는 카운터의 위치를 찾도록 하고, 상기 카운터의 위치를 찾고 난 후에는 상기 각 레이어에서 상기 카운터의 값과 상기 카운터의 최대 크기인 임계값과의 비교를 수행한다. 상기 비교를 수행하면서, 상기 각 레이어에서 상기 카운터의 값이 상기 카운터의 임계값 이상이라면 상기 카운터의 값을 유지하고, 상기 카운터의 값이 상기 카운터의 임계값 미만이라면 상기 카운터의 값을 최소값 - 상기 최소값은 상기 카운터보다 하위 레이어들의 임계값에 도달하지 않은 카운터의 값들 중 가장 작은 값을 저장한 값임 - 과 비교를 수행한다. 이 후, 상기 카운터의 값이 상기 최소값 이상이라면 상기 카운터의 값을 유지하고, 상기 카운터의 값이 상기 최소값 미만이라면 상기 카운터의 값을 1씩 증가시키고 증가시킨 상기 카운터의 값으로 최소값을 갱신한다. 상기 임계값 및

최소값과의 비교와 상기 카운터의 업데이트 과정은 상기 복수의 레이어의 최하위 레이어부터 최상위 레이어까지 순차적으로 수행되고, 수행 결과 상기 각 레이어의 카운터들의 값 중 임계값에 도달하지 않은 가장 작은 값인 상기 갱신된 최소값을 결과값으로 출력할 수 있다.

[0040] 도 3은 본 발명의 일실시예에 따른 고속 네트워크 스위치에 의해 스케치 기반의 트래픽을 측정하는 방법에서 3개의 레이어를 가지는 분할 카운터에서의 인코딩과 디코딩의 예시를 나타내는 개략도이다.

[0041] 도 3을 참조하면, 4 bit 카운터로 이루어지고 15의 임계값을 가지는 제1레이어, 8 bit 카운터로 이루어지고 255의 임계값을 가지는 제2레이어, 16 bit 카운터로 이루어지고 65,535의 임계값을 가지는 제3레이어로 이루어지는 분할 카운터의 인코딩 및 디코딩 과정을 확인 할 수 있다.

[0042] 상기 분할 카운터는 패킷의 플로우를 해시 함수로 계산한 키 값을 입력 받고, 상기 키 값을 이용하여 상기 제1레이어, 제2레이어, 제3레이어 각각에서 상기 키 값에 해당되는 카운터의 위치를 찾는다. 이 후, 최하위 레이어인 제1레이어에서부터 상기 플로우의 카운터가 해당 레이어의 카운터가 카운팅할 수 있는 최대 크기인 임계값보다 작은지 확인한다. 도3에서는 제1레이어가 최하위 레이어이기 때문에 만약 임계값보다 제1레이어의 카운터 값이 작다면 상기 제1레이어의 해당되는 카운터의 값을 1씩 증가시키고, 상기 증가한 카운터의 값을 최소값으로 저장한다. 이후 상위 레이어인 제2레이어로 진행하여 제2레이어의 카운터 값과 제2레이어 카운터의 임계값 및 상기 최소값과의 비교를 수행한다. 이 때에는 상기 제2레이어의 카운터 값이 상기 제2레이어 카운터의 임계값 및 상기 최소값 미만일 시에만 상기 제2레이어의 카운터 값의 업데이트를 수행하고 상기 업데이트 된 카운터 값으로 최소값을 갱신한다. 이러한 과정은 최상위 레이어인 제3레이어까지 반복된다.

[0043] 그러나, 상기 제1레이어의 카운터 값이 임계값 이상이라면, 상기 해당 위치의 카운터와 업데이트 하지않고, 최소값도 저장하지 않는다. 이후, 상위 레이어인 제2레이어로 진행하여 이를 다시 반복하고, 최종적으로 최상위 레이어인 제3레이어에 도달하여 이를 반복한다. 상기 동작이 완료되면 상기 분할 카운터는 이미 임계값에 도달한 레이어들을 제외한 나머지 레이어들 중에서 가장 작은 카운터 값을 디코딩 값으로 출력한다. 상기 디코딩 값의 출력은 상기 갱신한 최소값을 출력함으로써 이루어질 수 있다.

[0044] 도 3에서는 키와 해시 함수를 이용하여 얻은 3가지 해시값인 H_1 , H_2 , H_3 로 각 레이어에서 해당하는 카운터의 위치를 찾는 동작을 보인다. 상기 해시 함수의 개수는 임의로 설정할 수 있고, 상기 H_1 , H_2 , H_3 는 각각에 해당되는 3개의 해시 함수를 통하여 도출한 값 또는 하나의 해시 함수를 슬라이싱하여 도출한 값일 수 있다.

[0045] 상기 각 레이어의 해당하는 카운터 각각에서 임계값과의 비교를 수행하고, 제1레이어의 업데이트된 카운터 값인 10을 최소값으로 저장하여, 이를 제2레이어 및 제3레이어와 비교하였다. 도 3에서 제2레이어는 임계값에 도달하였기 때문에 제2레이어의 카운터 값을 업데이트 하지않고 제 3레이어로 진행한다. 제3레이어에서는 카운터의 값이 임계값 미만이지만, 상기의 최소값인 10을 초과하기 때문에 업데이트를 수행하지 않는다. 따라서, 최소값도 갱신되지 않고 10을 유지하게 된다. 이러한 과정을 통해 디코딩 결과 이미 임계값에 도달한 제2레이어를 제외한 제1레이어, 및 제3레이어의 카운터 값 중에서 가장 작은 값인 10을 출력하였다.

[0047] 도 4a, 도 4b, 도4c, 도4d는 본 발명의 일실시예에 따른 고속 네트워크 스위치에 의해 스케치 기반의 트래픽을 측정하는 방법에서, 카운터를 업데이트하는 과정의 4가지 시나리오를 나타내는 개략도이다.

[0048] 도 4a에서는 본 발명의 일실시예에 따른 미니멈 업데이트 방식으로 카운터를 업데이트 하면서 크로스-레이어 업데이트의 경향을 보이는 경우를 도시한다. 도 4a를 살펴보면, 제1레이어에서부터 제3레이어까지 순차적으로 카운터를 업데이트 하는 것을 확인할 수 있다. 제1레이어의 카운터가 1증가하여 16이 되고, 최소값으로 16을 저장하게 된다. 이 후, 제2레이어의 카운터가 16보다 작기 때문에 상기 제2레이어의 카운터도 1증가한다. 이를 제3레이어까지 반복하는 방식으로, 기존의 한 레이어의 카운터 업데이트 성공 시 다른 레이어의 업데이트를 중단하는 업데이트 방식과는 차이점이 있다.

[0049] 도 4b에서는 본 발명의 다른 실시예에 따른 미니멈 업데이트 방식에서 카운터에 오버플로우가 발생했을 시의 업데이트를 도시한다. 도 4b를 살펴보면, 좌측의 그림은 제1레이어의 카운터가 오버플로우된 경우를 나타낸다. 이러한 경우에는 제1레이어에서 카운터의 업데이트를 수행하지않고, 남은 레이어들에서 업데이트를 진행한다. 상기의 경우는 제3레이어보다 제2레이어에서 우연히 충돌이 더 많이 일어나 제3레이어의 카운터의 값보다 제2레이어의 카운터 값이 높아진 경우로, 제2레이어의 카운터 값을 업데이트하면서 최소값으로 813을 저장한다. 이후,

제 3레이어로 진행하여 제3레이어의 카운터 값이 임계값과 상기 최소값인 813보다 작기 때문에 업데이트를 수행하고, 최소값을 699로 갱신하는 모습을 보인다. 우측 그림에서는 제1레이어와 제2레이어가 오버플로우되어 제3레이어의 카운터만 업데이트 되는 경우를 나타낸다. 도 4b의 경우 다른 큰 플로우가 제3레이어에서는 현재 플로우와 충돌하지 않지만, 제2레이어에서는 충돌하여 카운터를 공유하는 경우를 나타낸다. 따라서 상기의 미니멈 업데이트 방식은 하위 레이어에서 오버플로우가 발생하더라도 상위 레이어가 임계값에 도달하지 않았다면 현재 플로우가 디코딩 될 수 있고, 이러한 임계값까지만 업데이트 하는 방식을 통해 엘리펀트 플로우가 마우스 플로우에게 임계값 이상의 노이즈를 줄 수 없도록 할 수 있다.

[0050] 따라서, 상기의 두 그림에서 모두 CL-MU가 작은 값을 가지는 플로우를 상위 레이어의 카운터에서 사용할 수 있도록 함으로써, 작은 플로우의 생존성을 높일 수 있다는 것을 확인할 수 있다.

[0052] 도 4c에서는 본 발명의 다른 실시예에 따른 미니멈 업데이트 방식으로 카운터를 업데이트 하는 방식을 도시한다. 도 4c에서 미니멈 업데이트는 CL-CU 방식과 매우 유사하다. 하지만 CL-CU 방식은 모든 레이어에서 최소값을 먼저 구하는 방식으로 부하가 높아 ASIC를 기반으로 한 네트워크 스위치에서는 사용될 수 없기 때문에, CL-CU 방식과 비슷한 결과를 내지만 스위치에서 사용될 수 있는 방식을 제안한다. 이러한 미니멈 업데이트 방식은 제1레이어의 카운터부터 순차적으로 업데이트 하면서 현재의 최소값을 갱신하는 업데이트 방식을 택한다. 하위 레이어는 카운터의 크기는 작지만 카운터의 수는 더 많기 때문에 하위 레이어에서 상위 레이어 순으로 업데이트가 진행되고, 이는 하위 레이어의 카운터가 최소값을 가질 가능성이 높고, 이러한 최소값의 순차적인 업데이트가 모든 레이어의 카운터 값에 대한 최소값을 제공할 가능성이 높다는 것을 의미한다.

[0053] 도 4c의 좌측 그림에서 제1레이어가 오버플로우되지 않았기 때문에 업데이트를 수행하여, 최소값이 201이 된다. 제2레이어에서는 상기 제2레이어의 카운터 값이 201을 넘기 때문에 업데이트를 수행하지 않는다. 이후 제3레이어에서는 제3레이어의 카운터 값이 201보다 작기 때문에 업데이트가 수행되는 것을 확인할 수 있다. 이와 달리, 도 4c의 우측 그림에서는 제3레이어에서 최소값 151보다 더 큰 카운터 값을 갖기 때문에 업데이트를 수행하지 않는 것을 확인할 수 있다.

[0054] 따라서, 상기의 미니멈 업데이트 방식은 CM 스케치 방식 혹은 FCM 스케치 방식과 달리 엘리펀트 플로우가 마우스 플로우에 의해 영향을 받지 않는다는 것을 알 수 있다.

[0055] 도 4d는 본 발명의 다른 실시예에 따른 미니멈 업데이트 방식에서 상위 레이어의 카운터가 가장 작은 값을 가지는 최악의 경우를 나타낸다. 도 4d를 살펴보면, 최소값으로 제1레이어를 106으로 업데이트 한 후에, 제2레이어 및 제3레이어가 업데이트 된다. 보수적인 업데이트 방식에서는 모든 레이어의 최소값을 확인하는 방식을 통해 제3레이어의 카운터 값만 증가하게 되지만, 미니멈 업데이트 방식에서는 제1레이어 및 제2레이어의 원치 않는 업데이트가 이루어지게 된다.

[0056] 그러나, CL-MU 방식에서는 하위 레이어에서 더 많은 수의 카운터 및 더 작은 크기의 카운터를 가지기 때문에 하위 레이어에서의 충돌과 오류가 적을 가능성이 높다. 그로 인해, 하위 레이어에서 더 작은 값을 가질 확률이 높고, 이러한 최악의 경우가 발생할 확률은 매우 적다.

[0058] 도 5는 본 발명의 일실시예에 따른 고속 네트워크 스위치에 의해 스케치 기반의 트래픽을 측정하는 방법에서, 카운터 분포와 각 카운터에 따른 bit 손실을 기존의 다른 방법과 비교한 그래프이다.

[0059] 도 5의 (a)에서 CL은 본 발명인 CL-MU를 지칭하고, 상기 그래프는 0.2 MB 메모리에서 CM, CL, 및 FCM 스케치 방식의 카운터 분포를 나타낸다. 도 5의 (a)를 살펴보면, CM의 경우 데이터 구조가 3개의 레이어에 균일하게 분포된 모습을 볼 수 있고, CL과 FCM은 모두 피라미드 형태로 카운터 수를 유지하여, 하위 레이어의 작은 크기를 가진 카운터들의 수가 많이 유지되는 것을 확인할 수 있다. CM과 CL, FCM의 데이터 구조의 차이는 결과적으로, 같은 메모리 공간에 있어서 CM보다 CL과 FCM에서 더 많은 카운터의 여유를 가질 수 있게 한다. 게다가, 하위 레이어에서의 카운터 크기의 감소는 CM에서 대부분의 메모리 낭비로 이어지는 MSBs(Most Significant Bit)서의 메모리의 낭비를 줄일 수 있다.

[0060] 도 5의 (b)를 살펴보면, 89퍼센트의 CM 스케치 방식의 카운터들이 전체 메모리의 75%가 넘는 수치인 24 MSBs가량을 낭비하고 있는 모습을 확인할 수 있다. 이러한 점에 있어서, 피라미드 형식의 데이터 구조가 더 나은 메모리 사용률을 보이며, 메모리 사용률의 차이는 플로우 인코딩 방법에 의해서도 발생하는 것을 알 수 있다.

[0061] 상기의 FCM, CL-MU, 및 CL-CM을 비교하면, 본 발명 CL-MU와 CL-CM의 업데이트 방식인 크로스-레이어(cross-layer) 업데이트 방식이 더 나은 메모리 효율성을 보였다. CL-CM이 CL-MU에 비해 조금 더 나은 메모리 사용률을 보였으나, 이는 CL-CM이 발생할 수 있는 오류와 관계없이, 모든 레이어의 카운터를 업데이트하는 방식을 사용했기 때문이다. 이와 달리, CL-MU는 기존의 미니멈 업데이트 방식과는 다른 방식으로 업데이트를 진행하기 때문에 크로스-레이어 업데이트 시에 발생하는 오류를 줄일 수 있다.

[0063] 도 6a는 본 발명의 일실시예에 따른 고속 네트워크 스위치에 의해 스케치 기반의 트래픽을 측정하는 방법에서의 bit 별 카운터 활용도를 레이어별로 비교한 결과를 나타내고, 도 6b는 FCM 방법의 bit 별 카운터 활용도를 레이어별로 비교한 결과를 나타내는 그래프이다.

[0064] 도 6a, 도 6b는 3개의 레이어를 가지는 CL-MU 방식과 FCM 방식을 나타낸다. 도 6a를 살펴보면, CL-MU 방식은 8 bit 이하에서 제2레이어의 83.9%에 해당하는 카운터들을 사용하였고, 제 3레이어의 49.9%에 해당하는 카운터들을 사용하였다. 상기의 그래프에서 CL-MU 방식이 마우스 플로우를 인코딩 시에 제1레이어 뿐만 아니라 제2레이어, 및 제3레이어도 활용하는 크로스-레이어 업데이트 방식인 것을 확인할 수 있으며, 이를 통해 마우스 플로우가 상위 레이어의 사용되지 않는 카운터를 사용하여 생존할 수 있도록 할 수 있다. 이는 상기의 크로스-레이어 업데이트 방식이 복수의 레이어에서 상기 마우스 플로우의 디코딩이 가능하여 해시 충돌을 감소시킬 수 있다는 것을 나타낸다.

[0065] 이와 반대로 도 6b를 살펴보면, FCM 방식은 8 bit 이하에서 제2레이어의 66.5%에 해당하는 카운터들을 사용하였고, 제3레이어의 99%에 해당하는 카운터들을 사용하였다. 따라서, FCM 방식의 플로우 충돌율은 CL-MU와 비교할 때 매우 높아지고, 낮은 평균 오류율(ARE: Average Relative Error)을 가진다.

[0066] 상기의 플로우 충돌율은

[0067]
$$C \approx \frac{n_f(n_f-1)}{2M}$$
 ----- 식 (1)

[0068] 로 나타내어지며, 여기서 n_f 는 플로우 수, M은 데이터 구조에서 가능한 인코딩 위치의 개수를 의미한다.

[0069] 상기의 FCM 방식은 플로우 인코딩 시 한 번에 하나의 레이어에 수행하기 때문에 M_{FCM} 은 언제나 제1레이어의 카운터들의 개수를 의미한다. 이와 달리, CL-MU는 크로스-레이어 업데이트 방식을 통해 플로우를 모든 레이어의 카운터에 인코딩하기 때문에, M_{CL-MU} 가 모든 레이어들의 카운터의 개수를 의미한다. 따라서, 같은 메모리를 가졌다고 가정할 때, M_{CL-MU} 가 M_{FCM} 에 비해 더 큰 값을 가지며, 이는 FCM 방식이 더 높은 플로우 충돌율을 가질 수 있다는 것을 의미한다.

표 1

[0071]

비대칭도	1.0	1.2	1.4	1.6	1.8	2.0	2.2	2.4
CL-MU	0.01	1.48	3.52	15.14	47.97	102.38	151.41	184.69
FCM(k=4)	0.04	1.08	7.38	25.45	83.56	232.92	605.06	851.19
Elastic	0.00	0.13	4.27	19.89	78.17	163.80	198.40	208.86

[0072] 표 1은 CL-MU, FCM, Elastic 스케치 방식의 지프의 법칙에서 비대칭도(skewness)에 따른 평균 오류율(ARE)을 나타내는 표이다.

[0073] 표 1을 살펴보면 CL-MU가 비대칭도 1.0, 1.2일 때를 제외한 대부분의 경우, 가장 높은 정확성을 보여준다. 비대칭도가 낮을수록 더 많은 엘리먼트 플로우가 네트워크 상에 존재한다는 의미이기 때문에 TCAM 기반의 Elastic 방식이 낮은 비대칭도에서 높은 정확성을 보이는 것은 당연하다고 할 수 있다. 하지만 비대칭도가 증가하면서, CL-MU가 다른 방식들에 비해 더 낮은 평균 오류율, 즉 더 높은 정확도를 제공한다.

[0075] 도 7은 본 발명의 일실시예에 따른 고속 네트워크 스위치에 의해 스케치 기반의 트래픽을 측정하는 방법(CL-

MU)과 다른 방법들의 (a)마우스 플로우 및 (b)엘리펀트 플로우 시 각각 발생하는 오류의 정도를 나타내는 그래프이다.

- [0076] 도 7을 참조하면, CL-MU, CM, CL-CM, CL-CU, FCM, 및 Elastic 스케치 방식들에서 255보다 작은 크기를 가지는 마우스 플로우와 255보다 큰 크기를 가지는 엘리펀트 플로우에서의 ARE(Averaged Relative Error)를 비교하여 나타내었다. CL-CM과 CM 방식은 가장 낮은 정확성을 보였으며, 이는 이들의 업데이트 방식이 모든 레이어를 카운터의 값과 상관없이 업데이트 하기 때문인 것으로 보인다. 이러한 업데이트 방식으로 인해 엘리펀트 플로우의 카운터들은 포화되어, 모든 플로우가 같은 레이어의 같은 카운터를 공유하게 된다. 결과적으로, 엘리펀트 플로우의 카운터에서의 오류가 매우 높아지게 된다.
- [0077] FCM 방식은 마우스 플로우를 최하위 레이어에서만 카운트하여 마우스플로우가 상위 레이어를 침범하는 것을 방지하기 때문에 상기CL-CM과 CM보다 더 나은 성능을 보인다. 하지만, 하위 레이어에서 오버플로우가 발생하면, 카운터를 공유하는 모든 플로우가 상위 레이어로 패킷을 인코딩하게 된다는 문제점이 있다. FCM 방식은 도 6b에 개시된 바와 같이 최하위 레이어에서 카운터의 높은 비트별 사용으로 인해 오버플로우가 자주 발생할 수 있고, 이러한 오버플로우는 마우스 플로우의 엘리펀트 카운터로의 침범을 야기할 수도 있다.
- [0078] 도 7의 (b)를 참조하면, Elastic 방식은 TCAM 기반의 플로우 테이블이 메모리에 존재하기 때문에 무거운 부하를 가지게 된다. 이로 인해, 최근에 개시된 스케치 방식인 CL-MU, FCM 중 최악의 엘리펀트 플로우에서의 오류율을 가진다.
- [0079] 이와 달리, CL-MU 방식은 마우스 플로우 및 엘리펀트 플로우에서 최적의 방식인 CL-CU와 비슷한 성능을 보인다. 다만, CL-CU는 모든 레이어에서 플로우의 카운터를 읽어들이고, 그 중에서 가장 작은 카운터 값을 업데이트 하는 방식이기 때문에 같은 메모리에 두 번 접근하는 것이 허용되지 않는 네트워크 스위치의 ASIC 에서는 사용이 불가능하다. 그러나, CL-MU의 경우 네트워크 스위치의 ASIC에도 사용이 가능하다는 장점이 있다.
- [0080] 상기와 같은 그래프를 통해 결과적으로, 미니멈 업데이트 방식이 엘리펀트 플로우에서의 오류를 상당히 감소시키는 것을 확인할 수 있다.
- [0082] 도 8a는 마우스 플로우 인코딩에서의 본 발명의 일실시예에 따른 트래픽을 측정하는 방법(CL-MU)과 다른 방법들의 생존율을 비교한 그래프이며, 도 8b는 엘리펀트 플로우 인코딩에서의 본 발명의 일실시예에 따른 트래픽을 측정하는 방법(CL-MU)과 다른 방법들의 생존율을 비교한 그래프이다.
- [0083] 크로스-레이어 방식과 미니멈 업데이트 방식의 시너지 효과를 확인하기 위해 새로운 메트릭인 플로우 생존율(FSR: Flow Survival Rate)을 제시한다. 상기 플로우 생존율에서는 플로우 추정치가 특정 RE(Relative Error)보다 작다면 생존한다고 정의한다. 따라서, 플로우 생존율이 더 높다는 것은 더 나은 측정 방법임을 의미한다.
- [0084] 도 8a, 도 8b는 CL-MU, FCM, Elastic 스케치 방식의 FSR을 레이어 별로 나타내었다. Elastic 방식은 3개의 레이어를 가지지 않고 2개의 메모리 공간을 가지기 때문에 SRAM을 제1레이어로, TCAM을 제2레이어로 가정하였다.
- [0085] 도 8a에서 CL-MU는 마우스 플로우가 3개의 레이어 모두에서 FSR 0.03% 내지 6.85% 비율로 살아남는 것을 확인할 수 있다. FCM은 FSR 0.00% 내지 0.58%의 낮은 생존율을 보인다. Elastic 방식의 경우 CL-MU와 비교하여 제1레이어에서 거의 두배 가량의 마우스 플로우 생존율을 보였다.
- [0086] 도 8b에서 CL-MU는 엘리펀트 플로우가 제2레이어 및 제3레이어에서 모두 17.15% 내지 87.24%가 생존하는 것을 확인하였고, FCM의 경우 제2레이어 및 제3레이어에서 0.38% 내지 14.21% 만이 생존하였다. Elastic 방식의 경우 CL-MU와 비교할 때 매우 낮은 생존율을 보였다.
- [0087] 결론적으로, CL-MU가 크로스-레이어 방식과 미니멈 업데이트 방식의 이점을 통해 제한된 자원에서 마우스 플로우부터 엘리펀트 플로우까지 균형잡힌 정확도를 보여주는 것을 확인할 수 있다.
- [0088] 상기 도 5 내지 도 8b의 그래프들의 결과, 본 발명인 CL-MU가 여러 지표들에 있어서, FCM, Elastic, CL-CM 보다 진보된 성능을 가지는 것을 확인할 수 있다. 또한, 상기 CL-MU는 CL-CU와 성능적인 지표면에선 유사한 점이 있으나, 메모리 사용 측면에서 이점이 있어, TCAM의 사용 없이 SRAM만으로 구현이 가능한 기술이기 때문에 네트워크 스위치에서도 활용이 가능하다.

[0090] 이상에서 설명된 복수의 레이어들로 이루어지는 분할 카운터를 구비한 고속 네트워크 스위치에 의해 스케치 기반의 트래픽을 측정하는 방법은 하드웨어 구성 요소, 소프트웨어 구성 요소, 및/또는 하드웨어 구성 요소 및 소프트웨어 구성 요소의 조합으로 구현될 수 있다. 예를 들어, 실시예들에서 설명된 시스템, 장치, 방법 및 구성 요소는, 예를 들어, 프로세서, 컨트롤러, 중앙 처리 장치(Central Processing Unit; CPU), 그래픽 프로세싱 유닛(Graphics Processing Unit; GPU), ALU(arithmetic logic unit), 디지털 신호 프로세서(digital signal processor), 마이크로컴퓨터, FPGA(field programmable gate array), PLU(programmable logic unit), 마이크로 프로세서, 주문형 집적 회로(Application Specific Integrated Circuits; ASICS), 서버, 또는 명령(instruction)을 실행하고 응답할 수 있는 기타 다른 어떠한 컴퓨팅 장치와 같이, 하나 이상의 범용 컴퓨터 또는 특수 목적 컴퓨터를 이용하여 구현될 수 있다.

[0091] 전술한 본 발명의 설명은 예시를 위한 것이며, 본 발명이 속하는 기술분야의 통상의 지식을 가진 자는 본 발명의 기술적 사상이나 필수적인 특징을 변경하지 않고서 다른 구체적인 형태로 쉽게 변형이 가능하다는 것을 이해할 수 있을 것이다. 그러므로 이상에서 기술한 실시예들은 모든 면에서 예시적인 것이며 한정적이 아닌 것으로 이해해야만 한다. 예를 들어, 단일형으로 설명되어 있는 각 구성 요소는 분산되어 실시될 수도 있으며, 마찬가지로 분산된 것으로 설명되어 있는 구성 요소들도 결합된 형태로 실시될 수 있다.

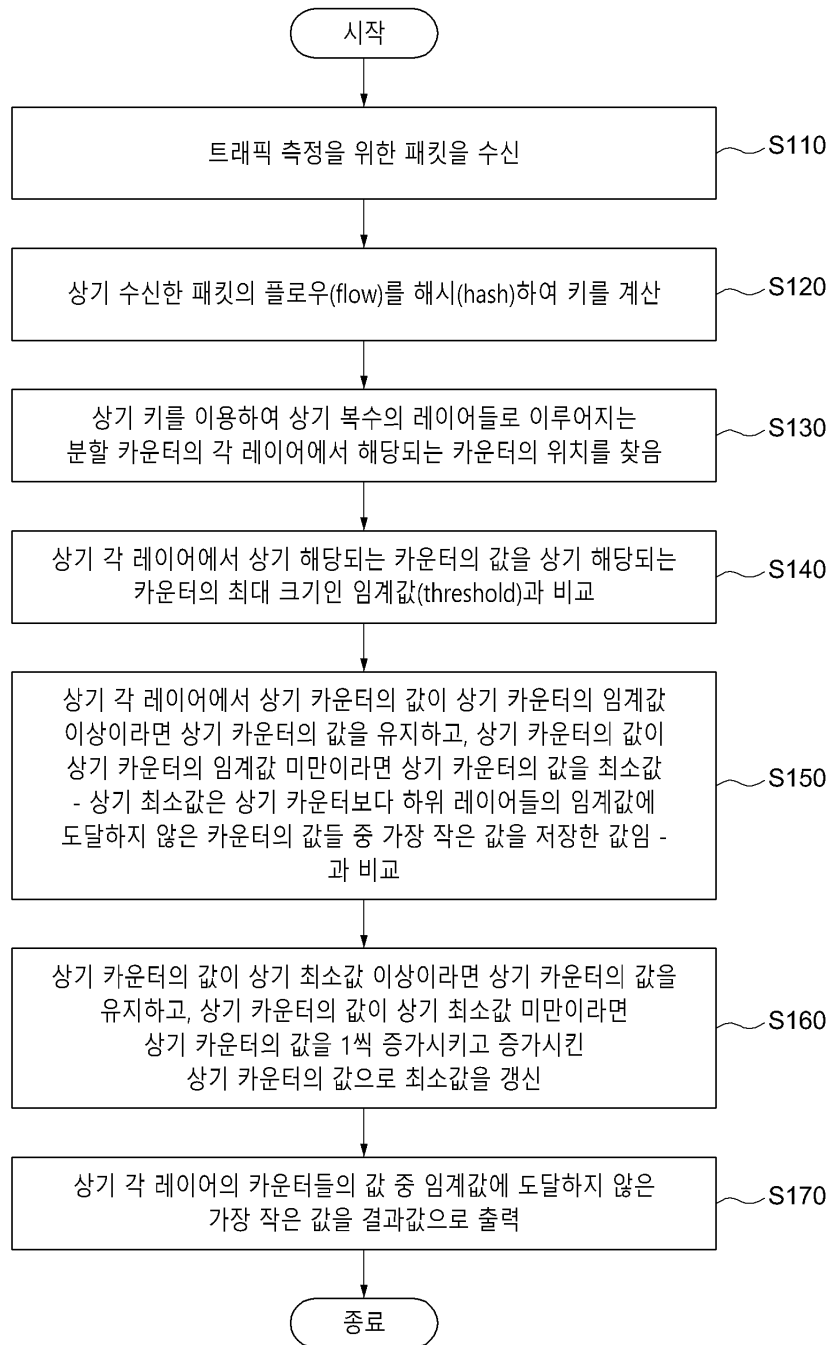
[0092] 본 발명의 범위는 후술하는 특허청구범위에 의하여 나타내어지며, 특허청구범위의 의미 및 범위 그리고 그 균등 개념으로부터 도출되는 모든 변경 또는 변형된 형태가 본 발명의 범위에 포함되는 것으로 해석되어야 한다.

부호의 설명

[0094] 200: 네트워크 스위치
210: 수신부
220: 프로세서
230: 분할 카운터

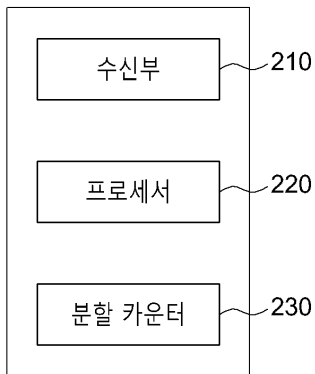
도면

도면1

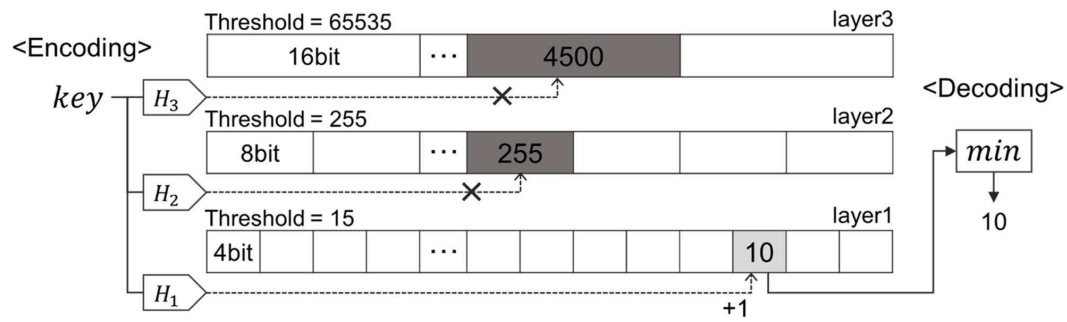


도면2

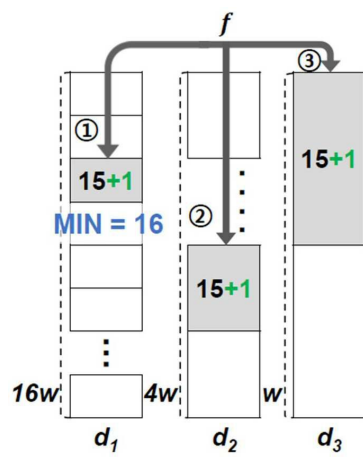
200



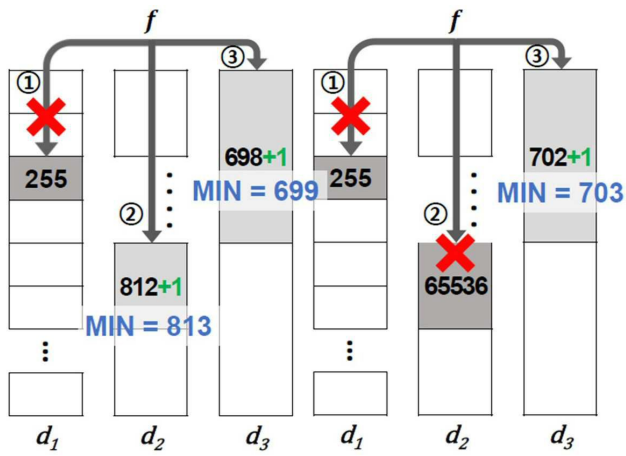
도면3



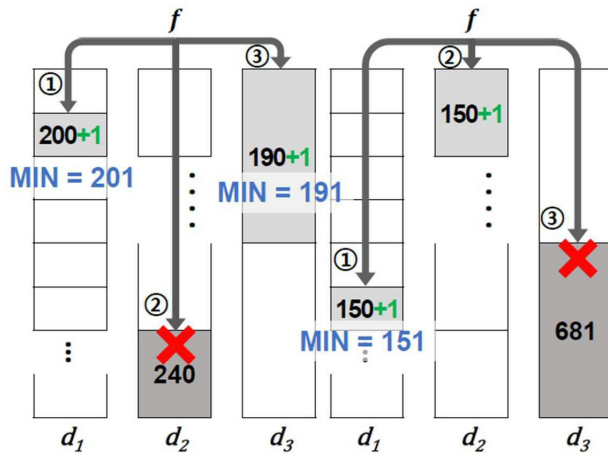
도면4a



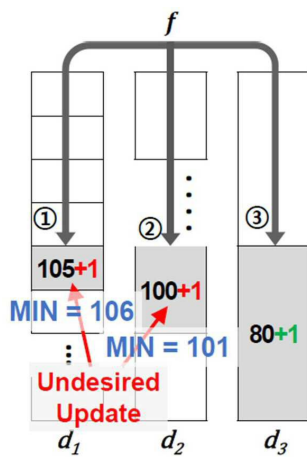
도면4b



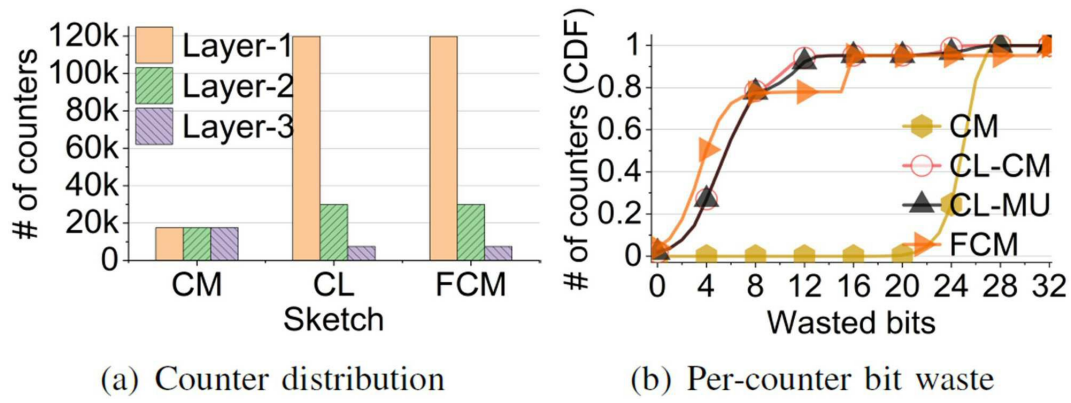
도면4c



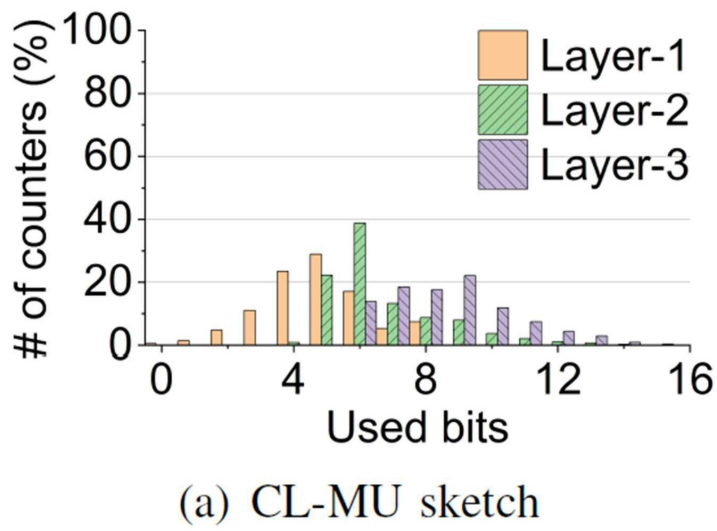
도면4d



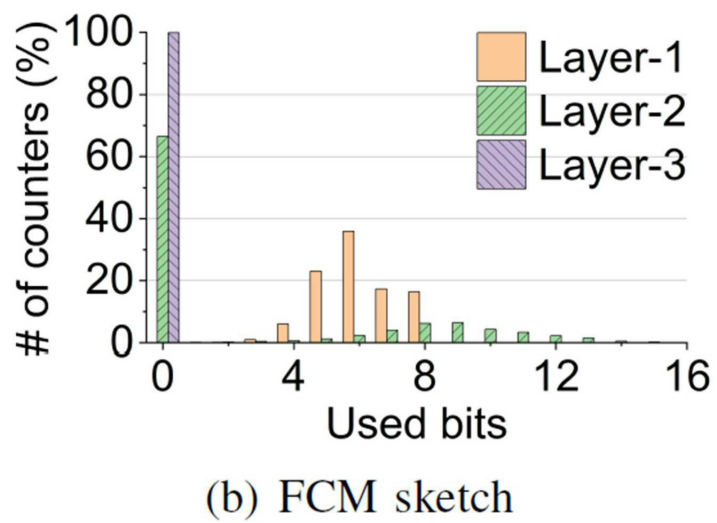
도면5



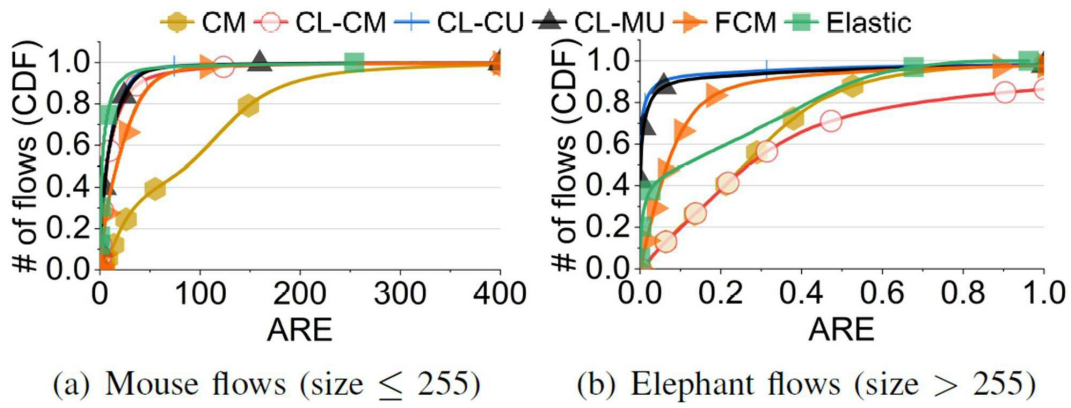
도면6a



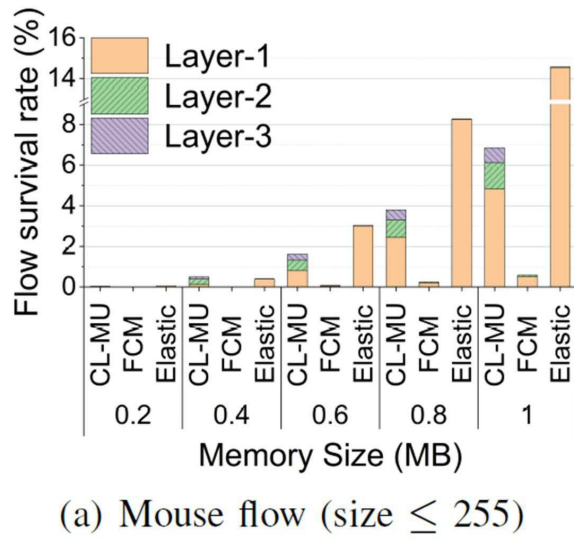
도면6b



도면7



도면8a



도면8b

