

## 패킷자리길리력에 기초한 망오유수정의 한가지 방법

리경심, 전남철

경애하는 김정은동지께서는 다음과 같이 말씀하시였다.

《오늘 세계는 경제의 지식화애로 전환되고있으며 우리앞에는 나라의 경제를 지식의 힘으로 장성하는 경제로 일신시켜야 할 시대적과업이 나서고있습니다.》

론문에서는 자료중심망의 규모가 커지는데 따라 위상구조가 복잡해지는것으로 인한 망오유를 수정하는 한가지 방법을 제안하였다.

### 1. 선행연구 및 문제설정

자료중심망은 수십만개 혹은 수백만개의 봉사기들을 호상연결하여 클라우드봉사를 제공하면서도 늘어나는 사용자들에게 충분한 대역폭을 제공해야 한다.

자료중심망은 크게 세가지 즉 교환기중심의 자료중심망, 봉사기중심의 자료중심망, 빛섬유 및 무선장치중심의 자료중심망으로 분류한다.[1]

론문에서는 교환기중심의 자료중심망에서 발생하는 오유를 수정하기 위해 VL2와 F<sup>2</sup>-Tree위상구조에 대하여 고찰한다.

그림 1에 VL2의 구조를 보여주었다.

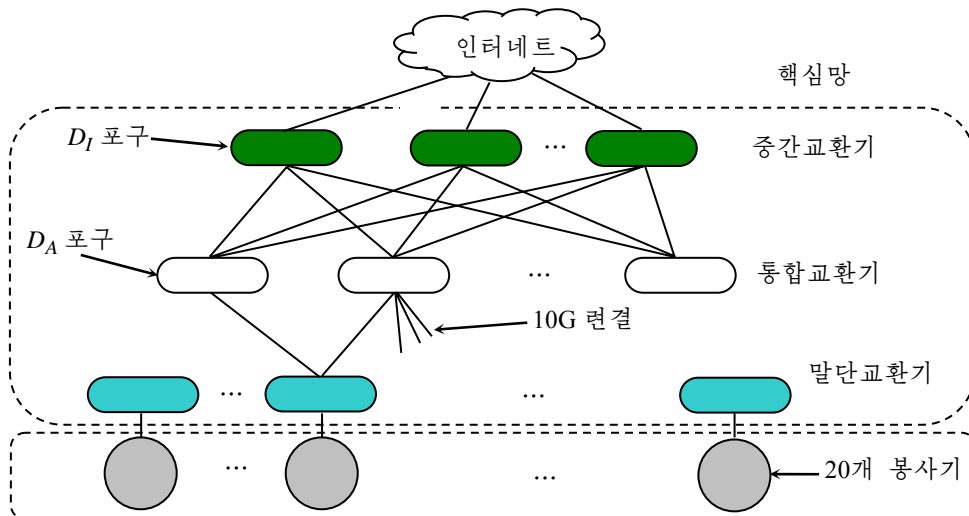


그림 1. VL2구조

VL2는 3준위로 중첩된 클로스망이다.

Fat-Tree와는 달리 VL2는 그림 1에서 보는것처럼  $D_I$  포구 중간교환기들과  $D_A$  포구통합교환기, 봉사기를 지원하는  $20 \times (D_I \times D_A / 4)$  개의 말단교환기들과 호상연결되어있다.

$D_A/2$  개의 중간교환기들은 논리적으로 완전한 쌍방향그래프를 구성하기 위해  $D_I$  개의 통합교환기들과 호상연결된다. 개별적인 말단교환기들은 2개의 통합교환기와 20개의 봉사기와 연결된다.

VL2는 또한 1:1초과가입을 지원하며 케블배치는 고성능연결에 의해 Fat-Tree보다 복잡하지 않다. 또한 고준위교환기의 경로조종비용이 Fat-Tree보다 더 높다.

VL2는 가상적인 2준위망을 실현하기 위해 3준위경로조종구조를 리용한다.

VL2는 VLB를 리용하여 작업부하의 불균형을 없앤다. 또한 교환기들과 봉사기들을 하나의 전체적인 자원덩어리로 보고 봉사기들에 IP주소들과 작업부하를 할당한다.

VL2는 임의의 6개의 도약경로를 추적하기 위해 3개의 연결을 표본화할것을 요구한다. DSCP마당을 추가적으로 리용하는데 마당이 6bit이므로  $k$ 개의 연결이 있는 포트내에서 말단-통합연결을 표본화하기 위해 리용한다.

패킷이 6개의 도약경로상에서 지나간다면 1개의 DSCP값과 2개의 VLAN코리표를 나른다. 이런 방식으로 6개의 도약경로를 지나가는 패킷에 대해서는 자료평면에서 규칙루실을 방지한다.

F<sup>2</sup>-Tree구조는 고장견딜성을 높이기 위해 제안된 방법으로서 여러 뿌리를 가진 자료중심망에서 고장발견시간을 크게 줄일수 있다.

F<sup>2</sup>-Tree구조는 여분경로의 성능을 높이기 위해 일부 연결을 재연결하고 국부적으로 재경로조종하기 위해 일부 교환기구성을 변경한다.

F<sup>2</sup>-Tree구조는 한고리내에서 핵심부층 혹은 통합층의 같은 포트내에 있는 교환기들을 연결한다. 이렇게 하여 고장이 발생했을 때 패킷전송을 정확히 할수 있는 타당한 연결을 위한 즉시적인 사본화연결이 늘어나게 된다.

## 2. 패킷자리길리력에 기초한 경로순환고리검출방법

본문에서는 우와 같은 자료중심망들에서 발생하는 오유들을 해결하는 망오유수정기를 제안하였다.

망오유수정기는 개별적인 교환기와 호스트들이 유일한 ID를 가진다고 가정한다.

linkID = ( $\langle S_i, S_j \rangle$ ): 한쌍의 린접한 switchID

Path = ( $\langle S_i, S_j, \dots \rangle$ ): 목록 switchID

flowID = ( $\langle \text{srcIP}, \text{dstIP}, \text{srcPort}, \text{dstPort}, \text{protocol} \rangle$ ): 5개의 튜플

Flow = ( $\langle \text{flowID}, \text{Path} \rangle$ ): 같은 flowID에서 나온 패킷들이 여러개의 Path들을 통과하는 경우들에 유용하다.

timeRange = timestamp( $\langle t_i, t_j \rangle$ )

망오유수정기는 switchID와 timestamp에 의한 모든 입력점을 해석한다.

실례로 망오유수정기는 ( $\langle *, S_j \rangle$ )를 교환기  $S_j$ 에 들어오는 모든 연결로 해석하며 ( $\langle t_i, * \rangle$ )는 시간  $t_i$ 로부터 해석한다.

망오유수정기는 패킷자리길을 추적하여 망오유를 발견하고 그것에 대한 조종을 진행한다. 망오유수정기가 패킷자리길을 추적하자면 개별적인 교환기들은 패킷머리부에 switchID를 추가해야 한다. 그러나 패킷자리길에 속한 모든 switchID를 추가하면 패킷머리부의 길이가 커진다.

망오유수정기는 상품교환기들을 리용하여 파के트자리길을 추적하기 위해 CherryPick[2]의 연결표본화방법을 리용하여 파케트머리부공간을 최량화하면서 VLAN과 MPLS꼬리표를 리용한다.

그림 2에 자료중심망에서 조종기와 망오유수정기의 호상동작과정을 보여주었다.

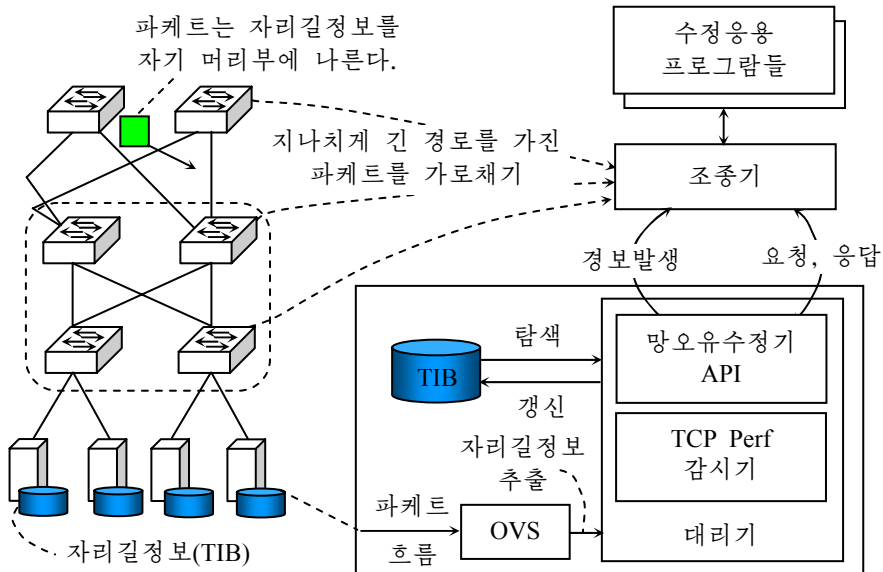


그림 2. 자료중심망에서 조종기와 망오유수정기의 호상동작과정

교환기 ASIC에서 파케트의 IP층을 분석하는것은 규칙루실을 초래하므로 조종기가 파케트를 식별하도록 한다. 조종기는 긴 경로를 가진 파케트를 가로채어 망오유수정기에 정보를 넘기고 망오유수정기는 턱값이상의 긴 경로를 통과하는 파케트를 식별하여 해석한다.

론문에서는 실시간경로조종순환검출응용프로그램을 망오유수정기에 실현하였다.

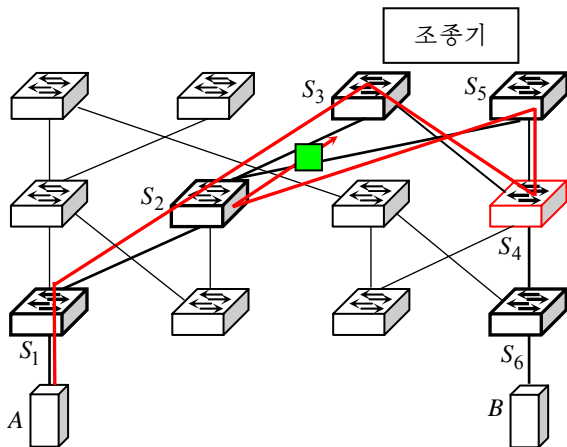


그림 3. 자료중심망에 존재하는 경로순환고리

망오유수정기는 망에서 긴 경로를 가진 파케트를 분석하여 실시간적으로 경로순환고리를 수정한다.

2개이상의 꼬리표를 나르는 하나의 파케트는 자동적으로 조종자에게 직접 전달된다. 조종자는 이상한 파케트들에 대한 검사를 진행하고 경로순환고리들을 검출하여 수정한다.

그림 3에 자료중심망에 존재하는 경로순환고리를 보여주었다.

망오유수정기는 파케트머리부의 길이가 긴 이상파케트들을 분석하여 순환고리를 검출한다.

그림 4에 파케트에 추가되는 연결꼬리표를 보여주었다.

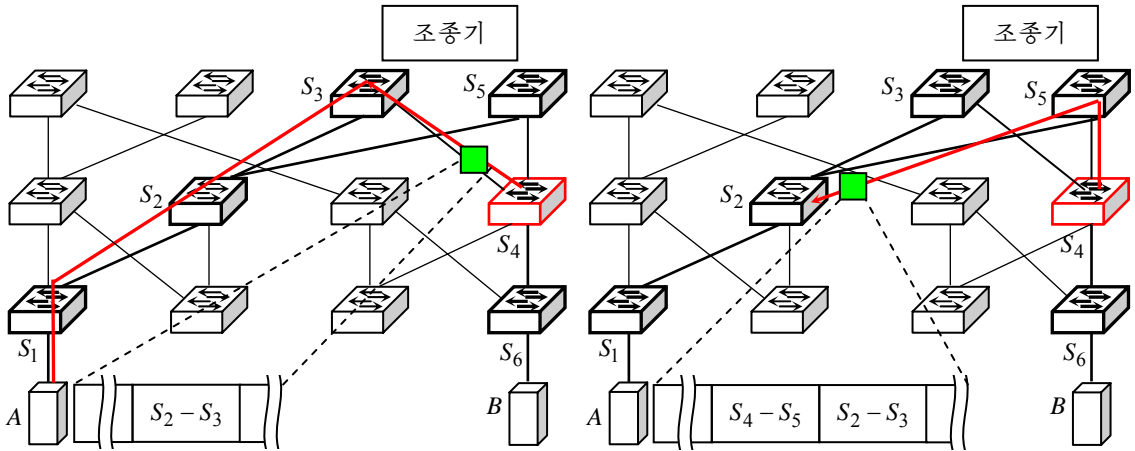


그림 4. 패킷에 추가되는 연결표리표

그림 5에 순환을 검출할수 있는 표리표를 가진 이상패킷을 보여주었다.

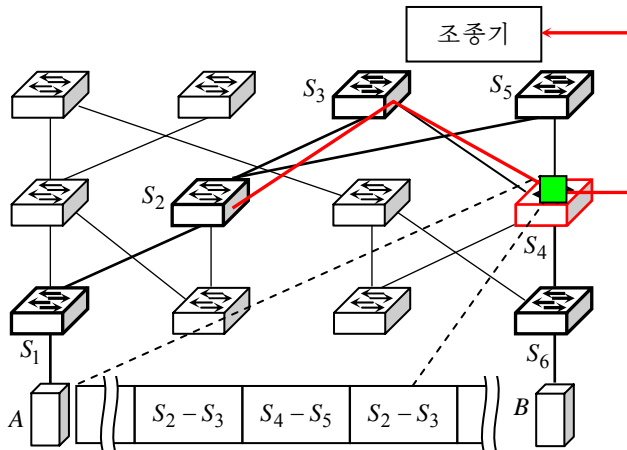


그림 5. 순환을 검출할수 있는 표리표를 가진 이상패킷

핵심부교환기  $S_3$  은 세번째 표리표에 연결 ( $S_2-S_3$ ) 에 해당하는 ID를 첨부한다. 통합층교환기  $S_4$  는 교환기의 ASIC가 2개의 VLAN표리표를 인식한 후 조종자에게 패킷을 자동적으로 전송하며 패킷은 2개의 표리표를 나른다. 이 단계에서 조종자는 즉시 패킷머리부에서 반복된 연결 ( $S_2-S_3$ ) 을 발견함으로써 순환고리를 검출한다.

만일 조종자가 3개의 표리표에서 반복된 linkID를 찾지 못하면 조종자는 그 패킷을 처음 보는 패킷으로 인정하고 3개의 표리표를 국부에 저장한 다음 패킷머리부로부터 그것을 잘라내어 교환기에 패킷을 전송한다.

논문에서는 패킷크기가 64B부터 1 500B사이에서 변하는 통화량을 생성하였다. 개별적인 패킷들은 1, 2개의 VLAN표리표들을 나른다.

자리길기억에 4 000개의 흐름기록(24개의 호스트들과 연결된 하나의 랙크교환기에서 초당 100 000개의 흐름)을 유지하는 동안 망오유수정기는 초당 0.8~3.6Mbps를 탐색하거나 갱신한다.(1 500B에 대하여 0.8Mbps, 64B에 대하여 3.6Mbps)

그림 6에 파के트크기에 따르는 망오유수정기의 처리량을 보여주었다.

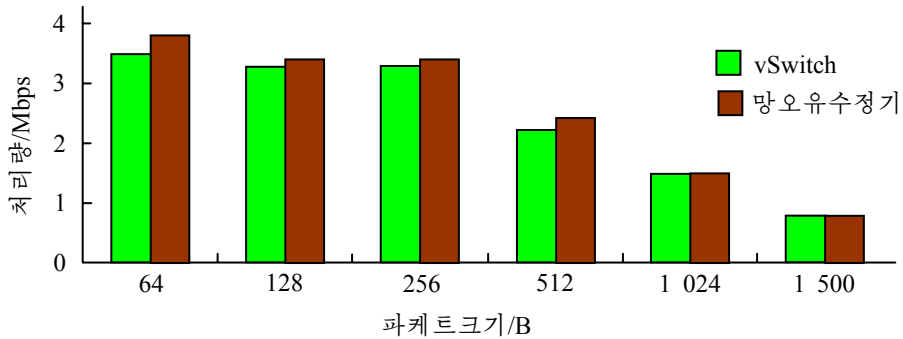


그림 6. 파케트크기에 따르는 망오유수정기의 처리량

그림 6에서 보는바와 같이 망오유수정기의 처리량이 vSwitch의 처리량에 비하여 4% 높다.

## 맺 는 말

파케트자리길리력에 기초하여 경로순환고리를 검출하여 망오유를 수정하는 방법을 제안하고 성능을 평가하였다.

## 참 고 문 헌

- [1] Tao Chen et al.; Journal of Parallel and Distributed Computing, 45, 2016.
- [2] T. Wang et al.; IEEE Globecom, 2252, 2014.

주제110(2021)년 2월 5일 원고접수

## A Method of Network Debugging Based on Tracing Packet Trajectories

*Ri Kyong Sim, Jon Nam Chol*

In this paper we have proposed the network debugger based on tracing packet trajectories, i.e., it debugs routing loop in realtime by trapping a suspiciously long path in data center networks.

Keywords: data center network, packet trajectory