

논문 2017-2-5

데이터센터 동적 구성을 위한 스위칭 셀 상태 개선

기장근*

Improvement of Switching Cell States for Dynamic Configuration of Data Centers

Jang-Geun Ki*

요 약

세계 IT 시장을 이끌어 가고 있는 글로벌 리더 기업들의 데이터 인프라 망에서 요구되는 미래 응용 서비스들의 비선형적 확장성을 만족시키기 위해서는 변화하는 작업부하의 필요에 따라 동적이고 자동으로 구성 블록들을 할당하고 조합이 가능한 유연성 있는 시스템이 필요하다.

본 연구에서는 데이터 센터 구성 자원들을 효율적으로 배분하고 이를 상호 연결하기 위해 자원간 최단경로를 계산하는 알고리즘을 이용하여 스위칭 셀들의 상태를 자동으로 제어하는 소프트웨어 프로그램을 개발하고, 1:2 연결을 위한 확장된 스위칭 셀 상태를 지원하여 연결 성공 확률, 연결 평균 길이, 스위치 미사용율 등의 성능 등이 향상됨을 보였다.

Abstract

To meet the nonlinear scalability of future application services required in the data infrastructure networks of global IT leader enterprises, flexible systems are needed to dynamically allocate the components blocks according to the change of workloads.

In this paper, a software program has been developed that automatically controls the state of switching cells using algorithms to calculate the shortest path between resources to efficiently distribute and interconnect configurable resources in the data centers. The extended switching cell states developed in this paper shows the performance improvement of connection success probability, average connection path length, and unused switching cell ratio.

한글키워드 : 데이터센터, 동적구성, 스위칭 셀, 제어 프로그램

keywords : data center, dynamic configuration, switching cell, control program

1. 서 론

구글, 아마존, 이베이, 이트레이드와 같은 기업들의 데이터 인프라 망은 복잡한 소프트웨어 응

* 공주대학교 전기전자제어공학부
(email: kjg@kongju.ac.kr)

접수일자: 2017.11.29. 심사완료: 2017.12.18.

게재확정: 2017.12.20.

용 프로그램들이 수행되는 수만대에서 수십만대에 이르는 컴퓨터들과 저장 장치들을 포함하는 데이터 센터들에 의해 운영되고 있다. IDC의 자료[1]에 의하면 2013년부터 2020년 사이에 모바일, 소셜 네트워크 서비스, 클라우드, 빅데이터 기술 분야에 대한 기업들의 투자가 과거 클라이

언트/서버 기술 분야에 대한 투자보다도 20배나 더 빨리 증가할 것으로 기대되고 있다. 현재의 전통적인 IT 기반구조는 매우 빠르게 발전하고 있는 미래 응용 서비스들의 새롭고 동적인 시스템 요구 조건들을 충족하기에는 융통성이 매우 부족하다. 미래 응용 서비스들의 공통적인 특징은 상대적으로 예측이 힘든 비선형적인 확장성을 수시로 요구한다는 점이며, 이와 같은 동적 변화 요구로 자원들의 효율적인 배치 작업이 매우 어렵게 된다.

물리적인 데이터센터 인프라 구성 및 유지 운영의 비효율성을 극복하기 위해서는 변화하는 작업부하의 필요에 따라 동적이고 자동으로 구성 블록들을 할당하고 조합이 가능한 유연성 있는 시스템이 필요하다. 이와 같은 조립가능 인프라 구조는 자원들을 저장장치들과 컴퓨터들의 풀(pool), 그리고 이들을 연결해주는 네트워크로 구별하고, 제어 소프트웨어가 특정 응용 서비스가 요구하는 처리능력을 만족시키기 위해 필요로 되는 하드웨어를 자원 풀로부터 할당해 논리적으로 구성함으로써 비용이 많이 드는 물리적인 하드웨어 재구성 및 자원 재할당 절차를 필요 없게 만든다. 소프트웨어 개발자들은 정책 프로파일과 서비스 프로파일들을 사용해 각 응용 서비스들의 물리적인 인프라 구조 요구조건을 규정하고, 각 응용 서비스 레벨의 목적을 만족시킬 수 있는 가상 데이터 센터를 구축할 수 있도록 API 호출을 사용해 소프트웨어를 작성한다. 따라서 가상 데이터 센터를 구성하기 위한 자원들의 물리적인 위치는 각 자원들이 서비스처럼 취급되기 때문에 크게 문제되지 않는다. 이와 같은 가상 데이터 센터의 개발은 짧은 프로비저닝 시간과 융통성의 증가로 인해 자원의 낭비를 줄이고 새로운 응용 실행 환경을 구축하는데 걸리는 시간을 획기적으로 줄여줄 것이다.

현재 제한적이긴 하지만 자원들의 동적 구성

을 통한 가상 데이터 인프라를 지원하는 시스템으로 HPE(Hewlett Packard Enterprise)사의 Synergy 시스템[2]과 Cisco사의 UCS(Unified Computing System) M 시리즈 서버[3] 등이 있다. HPE Synergy 시스템에서 CPU는 저장장치 및 메모리와 분리될 수 있지만 지연시간 문제 때문에 물리적으로 구성요소들이 가까이 위치하고 있다. 진정한 의미의 조립가능 시스템은 데이터센터 구성요소들이 Tera bps 정도의 초고속망으로 연결되어 아주 낮은 지연시간을 가져야만 가능할 것이다. Al-Fares[4]의 연구결과에 따르면 다양한 작업부하를 지원하는 가상 데이터센터의 동적인 구성이 가능하기 위해서는 기본적인 구성요소들(프로세서, 저장장치, 메모리, 그래픽 프로세서 및 특정 하드웨어 등)이 적어도 100 Tbps 속도 이상의 네트워크로 연결될 수 있어야 하고, 최대 지연이 10 us 이하의 범위에 있어야 한다.

Nirmal Kumbhare 등은 논문[5]에서 조립가능 인프라 구조로 JITA(Just in Time Architecture) 구조를 제안하고, 구성요소 연결을 위한 광 크로스바 스위치 구조에 대한 아이디어를 제시하였고, 참고문헌 [6]에서는 이 구조에 대한 기본적인 성능분석을 수행한 바 있다. 본 논문에서는 참고문헌 [6]의 확장연구를 수행하여 스위치 연결구조의 확장을 통해 성능 향상을 도모하고, 이들 광 크로스바 스위치 구조의 기능 검증 및 성능 분석을 위한 시뮬레이터 개발에 관해 기술하였다.

2. 광스위치 연결 구조

Nirmal Kumbhare가 제안한 고성능 저지연의 JITA 인프라 구조[5]를 위해 제안된 광스위치 연결구조의 스위칭 셀 동작 원리를 그림 1에 나타내었다[5]. 그림 1에서 각 스위칭 셀은 입출력 광

도파관을 형성하는 수직 형태의 수동 리지 광도파관 2개로 구성되어 있고, 이 위에 2개의 AVCs(Active Vertical Couplers)가 각 교차점위에 놓이는 구조로 되어있다.

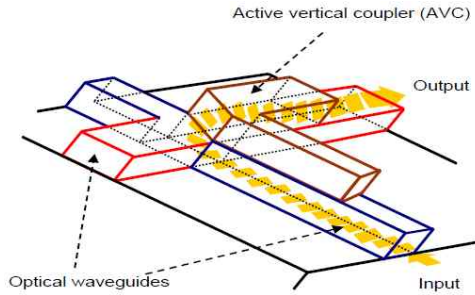


그림 1. Nirmal Kumbhare 스위치셀 구조[5]
Fig. 1. Nirmal Kumbhare switch cell

그림 1의 스위칭 셀은 입출력포트의 N(North), S(South), E(Ease), W(West) 연결 방향에 따라 NS-EW, NE-SW, NW-SE의 3가지 상태를 가질 수 있다. 그러나 이러한 구조는 1:1 연결만을 지원할 수 있고, 1:2 연결을 지원할 수가 없어 참고문헌 [6]에서 1:2 연결 지원을 위한 T 형태의 연결 상태를 지원하도록 확장하여 WNS, ESN, NEW, SWE(그림 2의 스위치 셀 상태 41~44 참고)를 지원하도록 하고 이에 대한 성능 분석을 수행한 바 있다.

본 논문에서는 참고문헌 [6]의 확장 연구 일환으로 T형태의 연결지원 뿐만 아니라 h 형태의 연결을 지원(그림 2의 스위치 셀 상태 51~54, 61~64 참고) 하고 이에 대한 성능을 비교 분석하

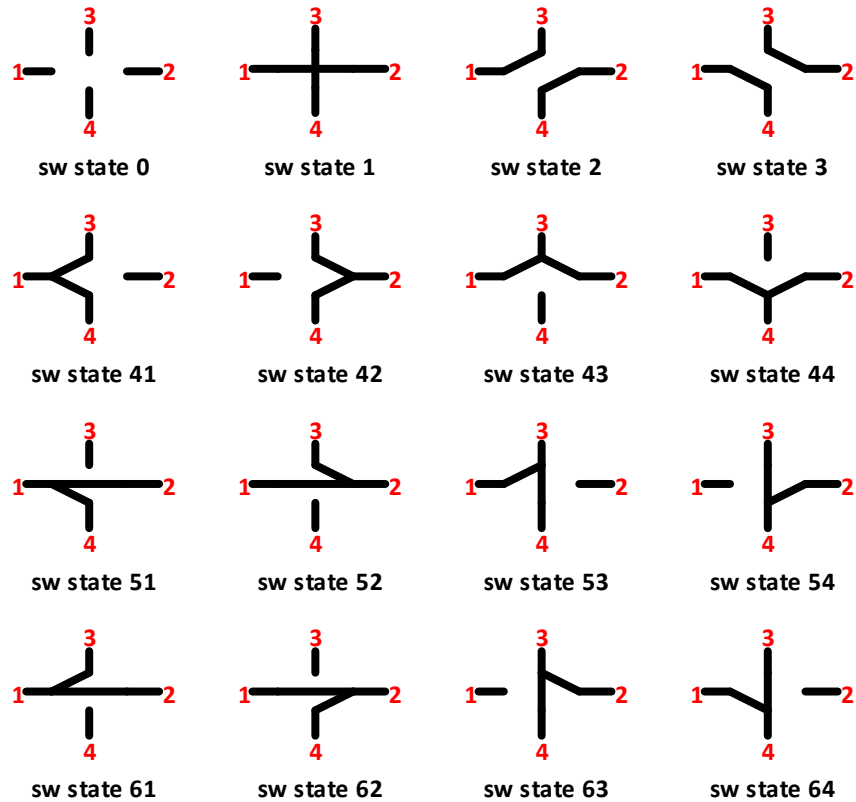


그림 2. 확장된 스위치 셀 연결 구조
Fig. 2. Extended switch cell connection

였다. T 형태의 연결은 입력포트가 좌측과 우측에 있는 포트와만 연결될 수 있으나, h 형태의 연결은 입력포트가 나머지 3개의 어느 포트 2개와도 연결될 수 있음을 의미한다.

3. 스위치 제어 시뮬레이터 개발 및 성능 분석

본 연구의 선행연구로 참고문헌 [6]에서 데이터 센터 구성 자원들을 효율적으로 배분하고 이를 상호 연결하기 위해 자원간 최단경로를 계산하는 알고리즘을 이용하여 스위칭 셀들의 상태를 자동으로 제어하는 소프트웨어 프로그램을 개발한 바 있으며, 1:2 연결 지원을 위해 T자 형태의 스위칭 셀 상태를 지원하도록 개발되었었다.

참고문헌 [6]에서 개발된 스위치 제어 시뮬레이터는 최근 SDN(Software Defined Network) 연구에 많이 사용되고 있는 Mininet을 이용하도록 개발 되었다. Mininet은 가상 호스트들과, 스위치들 및 컨트롤러 그리고 이들을 연결하는 링크들을 제공하는 네트워크 에뮬레이터이다. Mininet의 스위치를 제어하는 컨트롤러 프로그램을 개발하기 위해 컴포넌트 기반의 SDN 프레임워크를 제공하는 Ryu 오픈 플로우 컨트롤러가 사용되었으며, Python 언어를 사용해 스위치 제어 프로그램을 개발하였다.

먼저 스위치 구조를 에뮬레이션하는 스위치 파이썬 프로그램은 Mininet을 이용해 먼저 스위치 행과 열의 수(sw_size_row, sw_size_col)를 입력으로 받아 해당 개수만큼 동서남북 방향에 위치한 적절한 이름의 host들을 생성하고 ip주소와 mac 주소를 할당해 네트워크에 삽입하고, 이 host들을 연결하는 sw_size_row * sw_size_col 개수만큼의 switch들을 생성해 네트워크에 추가한다. 여기서 Mininet에 추가되는 host들은 데이

터센터를 구성하는데 필요한 자원(예를 들어 CPU나 기억장치 및 스토리지 등)들을 의미한다.

다음에 이들 스위치들을 제어하는 컨트롤러로는 RemoteController를 사용할 수 있도록 설정하고, 위에서 추가한 host들과 switch들을 모두 연결하는 link 들을 삽입한 후, 네트워크의 동작을 시작하도록 프로그램 되었다.

Ryu 오픈 플로우 컨트롤러 프로그램은 프로그램 실행 초기에 switch와 host들로 구성된 네트워크를 Graph 클래스로 모델링하고 Dijkstra의 최단경로 알고리즘을 이용하여 지정된 host들을 상호 연결하는 경로를 찾아 해당 스위치들의 상태를 제어하도록 프로그램 되었다.

본 연구에서는 선행연구[6]에 추가적으로 그림 2에 나타난 것과 같이 스위치 셀 상태 41~44의 T자 형태 뿐만 아니라 셀 상태 51~54, 61~64까지의 h자 형태의 셀 상태를 지원하도록 컨트롤러 프로그램을 확장하였고 이에 따른 성능 향상 정도를 분석하였다.

그림 2에서 상태 41은 포트1과 포트3, 포트1과 포트4가 연결됨을 의미하고, 상태 42는 포트2와 포트3, 포트2와 포트4가 연결됨을 의미한다. 상태 51은 포트1과 포트2, 포트1과 포트4가 연결됨을 의미하고, 상태 52는 포트2와 포트1, 포트2와 포트3이 연결됨을 의미한다. 마찬가지로 상태 63은 포트3과 포트4, 포트3과 포트2가 연결되고, 상태 64는 포트4와 포트3, 포트4와 포트1이 연결됨을 의미한다.

그림 3, 4, 5에 확장된 스위칭 셀 연결 구조에 대한 시뮬레이션 결과를 나타내었다. 그림에서 T_only 결과 그래프는 그림 2의 확장된 스위치 셀 상태에서 40번째 상태까지 지원하는 경우에 대한 결과이고, h_any 결과 그래프는 60번째까지의 모든 셀 상태를 지원하는 경우에 대한 결과이다.

그림 3의 1:2 연결 성공 확률을 비교해 보면

기존의 연구에서 40번대 상태의 T 연결만을 지원하는 경우에 비해 60번대 까지의 상태 연결을

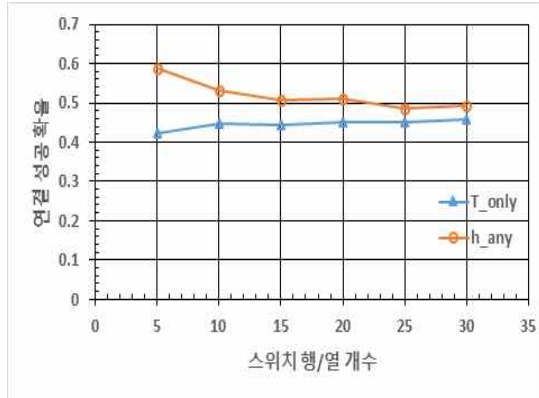


그림 3. 1:2 연결 성공 확률 비교
Fig. 3. 1:2 connection success rate

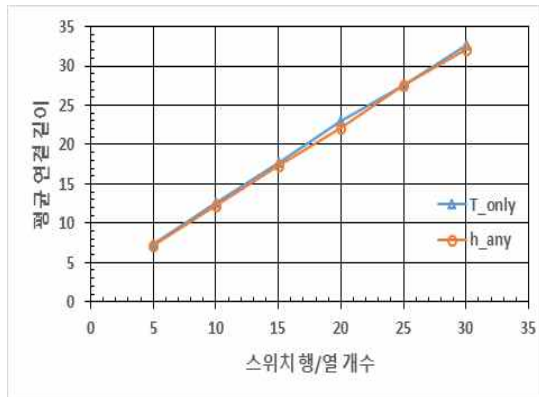


그림 4. 1:2 연결 평균 길이 비교
Fig. 4. 1:2 connection average length

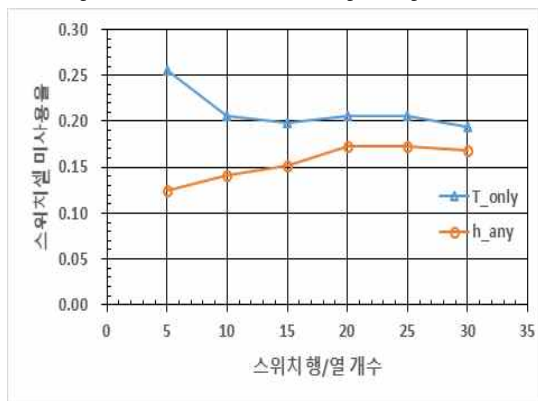


그림 5 1:2 연결 스위치 셀 미사용율 비교
Fig. 5. 1:2 connection switch cell unused

모두 지원하는 경우 연결 성공 확률이 더 큼을 알 수 있으며, 특히 스위치의 행/열의 개수가 작을수록 성공 확률의 차이는 크게 나타났다.

그림 4에서 1:2 연결의 평균 길이는 큰 차이는 없으나 일반적으로 모든 셀 상태를 지원하는 h_any의 경우가 T_only의 경우에 비해 약간 작은 값을 갖는다.

그림 5에서 1:2 연결시 셀 스위치의 미사용율을 비교해 보면 h_any의 경우가 T_only의 경우에 비해 스위치 셀 미사용율이 낮음을 볼 수 있으며, 이는 연결 성공률이 높을수록 보다 많은 연결이 성립되었을 것이고 따라서 보다 많은 스위치 셀들이 사용되었을 것임을 암시한다.

4. 결 론

세계 IT 시장을 이끌어 가고 있는 글로벌 리더 기업들의 데이터 인프라 망은 복잡한 소프트웨어 응용 프로그램들이 수행되는 많은 컴퓨터들과 기억 및 저장 장치들을 포함하는 데이터 센터들에 의해 운영되고 있으며, IDC의 자료에 의하면 앞으로 모바일, 소셜 네트워크 서비스, 클라우드, 빅데이터 기술 분야에 대한 기업들의 투자가 훨씬 더 빨리 증가할 것으로 기대하고 있다.

그런데 비선형적 확장성을 수시로 요구하는 미래 응용 서비스들의 요구를 만족시키기 위해서는 변화하는 작업부하의 필요에 따라 동적이고 자동으로 구성 블록들을 할당하고 조합이 가능한 유연성 있는 시스템이 필요하다.

Nirmal Kumbhare 등은 논문[5]에서 조립가능 인프라 구조로 JITA(Just in Time Architecture) 구조를 제안하고, 구성요소 연결을 위한 광 크로스바 스위치 구조에 대한 아이디어를 제시하였고, 참고문헌 [6]에서는 이 구조에 대한 기본적인 성능분석을 수행한 바 있다. 본 논문에서는 참고

문헌 [6]의 확장연구를 수행하여 스위치 연결구조의 확장을 통해 성능 향상을 도모하고, 이들과 광 크로스바 스위치 구조의 기능 검증 및 성능 분석을 위한 시뮬레이터 개발에 관해 기술하였다.

개발된 스위치 제어 시뮬레이터는 최근 SDN(Software Defined Network) 연구에 많이 사용되고 있는 Mininet을 이용하여 스위치 구조를 모델링하도록 스위치 프로그램이 개발되었다. Mininet의 스위치를 제어하는 컨트롤러 프로그램은 컴포넌트 기반의 SDN 프레임워크를 제공하는 Ryu 오픈 플로우 컨트롤러 프레임워크를 사용하여 스위치 상태제어 프로그램이 개발되었으며, Python 언어를 사용해 Dijkstra 최단경로 알고리즘을 이용해 host들을 연결하는 경로를 찾고 이 경로에 따라 스위치의 상태를 제어하도록 프로그램이 개발되었다.

개발된 프로그램을 이용한 시뮬레이션을 통해 본 연구에서 제안하는 스위치의 확장 상태를 지원하는 경우 연결 성공 확률, 연결 평균 길이, 스위치 미사용율 등의 성능 등이 향상됨을 보였다.

참 고 문 헌

- [1] IDC, "The 3rd Platform: Enabling Digital Transformation", Nov. 2013.
- [2] Hewlett Packard Enterprise, "HPE Synergy", <https://www.hpe.com/us/en/integrated-systems/synergy.html>, 2017.
- [3] Cisco, "Cisco UCS M-Series Modular Servers datasheet", <https://www.cisco.com/c/en/us/products/collateral/servers-unified-computing/ucs-m-series-modular-servers/datasheet-c78-732667.html>, 2017.
- [4] M. Al-Fares, A. Loukissas, A. Vahdat, "A scalable, commodity data center network architecture", ACM Special Interest Group on Data Communication (SIGCOMM) Review, Vol.38, Issue 4, pp.63-74, Aug. 2008.
- [5] Nirmal Kumbhare, Cihan Tunc, Salim Hariri, Ivan Djordjevic, Ali Akoglu, Howard Jay Siegel, "Just In Time Architecture (JITA) for Dynamically Composable Data Centers", IEEE/ACS 13th International Conference of Computer Systems and Applications (AICCSA), Nov.29-Dec.2, 2016.
- [6] 기장근, "데이터센터 구성요소 동적 연결 구조 성능 분석", 한국소프트웨어감정평가학회 논문지, 13권 1호, pp.51-55, Jun. 2017.

저 자 소 개



기장근(Jang-Geun Ki)

1986.2 고려대학교 전자공학과 졸업
 1988.2 고려대학교 전자공학과 석사
 1992.2 고려대학교 전자공학과 박사
 2002.6-2003.6 Univ. of Arizona 방문교수
 2010.6-2011.8 Univ. of Arizona 방문교수
 2016.8-2017.8 Univ. of Arizona 방문교수
 1992.3-현재 : 공주대학교 공과대학 전기
 전자제어공학부 교수
 <주관심분야>통신프로토콜, 이동통신시스템