성균관대학교
SUNG KYUN KWAN UNIVERSITY

Self-Organization: Adaptation, Healing and Configuration

성균관대학교
Syed Muhammad Raza

2016.11.25

성균관대학교 SUNG KYUN KWAN UNIVERSITY

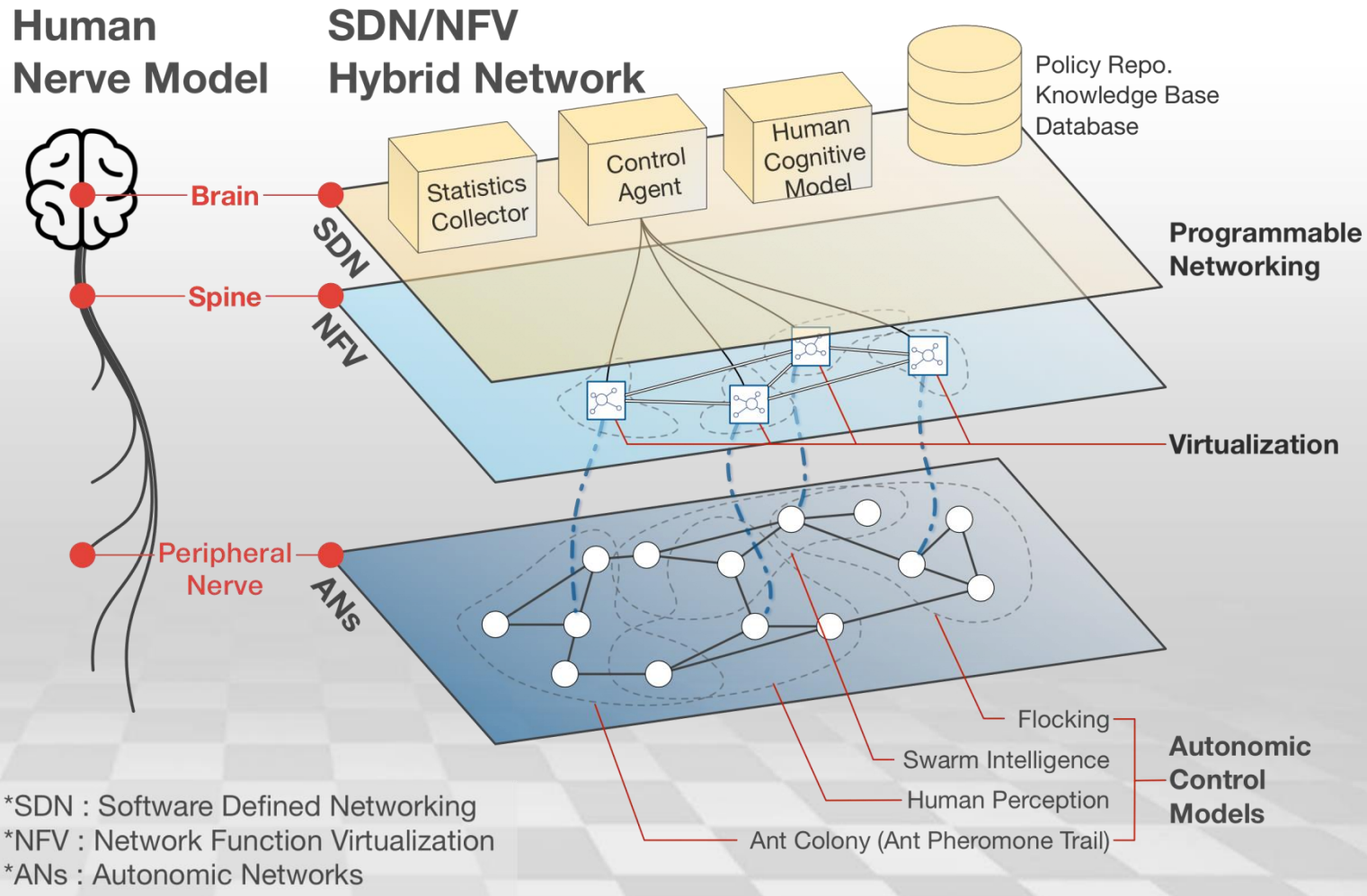
발표 순서

1. 개요
2. 핵심 연구 기술
   - Bio-inspired Algorithm and NMS Application Library
   - Policy Repository and Network Management Visualizer
   - SDN/NFV/Controller/Autonomic Networks
3. Topics for today

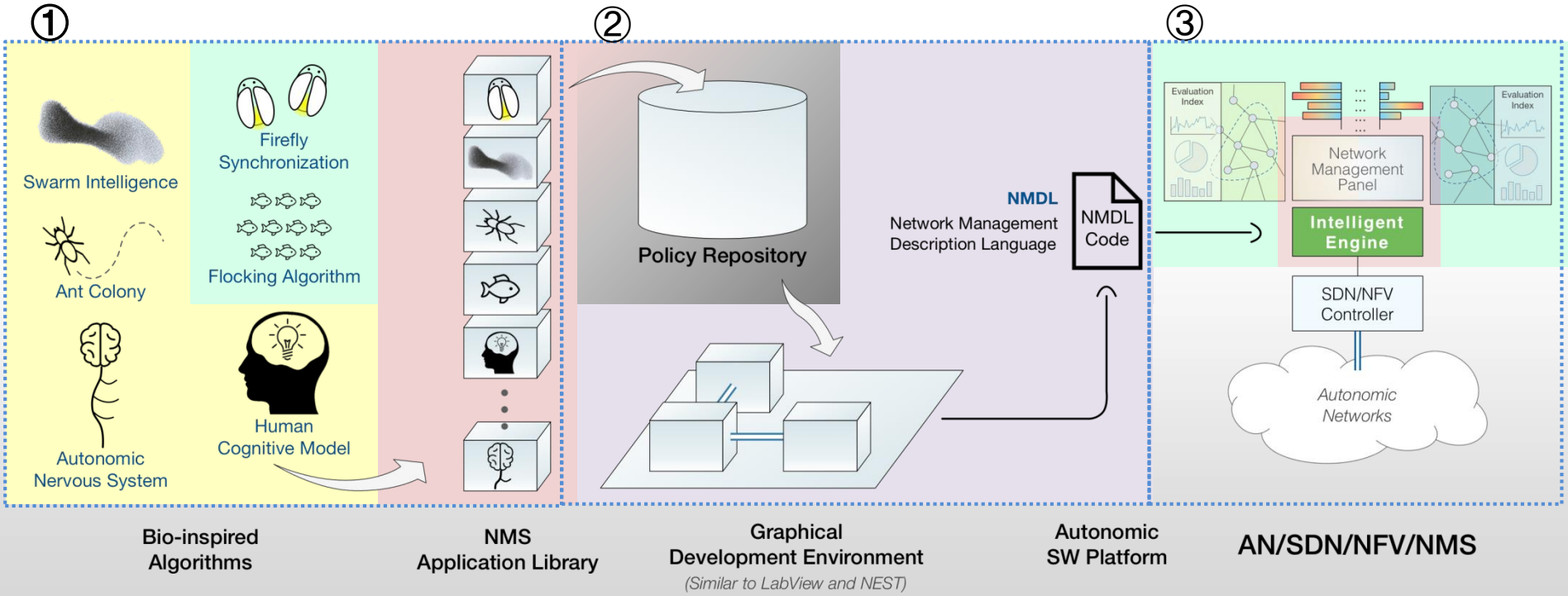
   Self-Organization: Configuration, Adaption and Healing
4. Testbeds and Open Sources
5. Q & A

개념도

"생체의 생존 및 사회활동 알고리즘을 이용한
자율 제어 네트워킹 및 자율 관리 핵심기술 개발"



기술 연계도



1. Bio-inspired Algorithm and NMS Application Library
2. Policy Repository and Network Management Visualizer
3. SDN/NFV/Controller/Autonomic Networks

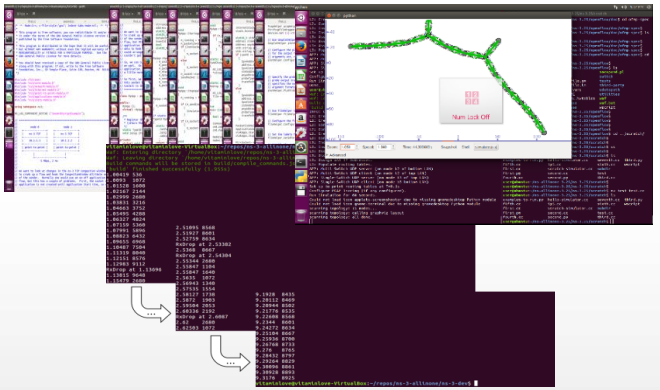
2.1 Bio-inspired Algorithm and NMS application Library

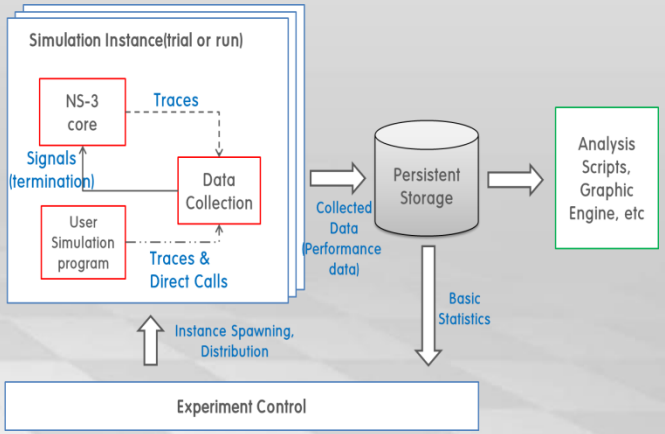
성균관대학교 SUNG KYUN KWAN UNIVERSITY

- 개미 군집 최적화 기반 라우팅 및 자가 치유
  - 개미의 경로 관리 및 채집 행동 양식 기반의 자가 조직화 및 치유 가능 라우팅 및 네트워크
- Firefly Synchronization Algorithm
  - 다중 컨트롤러 간 동기화
  - 알고리즘의 시간 동기화를 위한 분산 합의
  - Multi-hop 유지를 통한 데이터 전송 문제 해결
- Self-configuration/-organization
  - 스위치 장애에 대응하는 Self-configuration
  - 링크 장애 영향을 최소화하는 Self-organization
  - 링크 장애 발생 시 링크의 백업 경로로 Flow 복구
- MBF(Multiple Bloom Filter)기반 자원 관리
  - 스위치 자원의 효율적 관리를 위한 MBF 모델 최적화
  - MBF 기반의 Flow Eviction 메커니즘
  - ASIC, SRAM기반의 OpenFlow 스위치 향 MBF 모델
- 생체 모방 자율 제어 알고리즘 성능 비교 모델
  - 알고리즘 기본 기능 도출을 통한 성능 비교 지표 추출로 측정 방안 검토 및 모델 설계
- SDN 기반 도메인 간 이동성 관리
  - SDN Controller의 Mobility Management
- SDN기반의 효율적인 쿼리 모니터링 기법
  - 기존의 SDN 기반 네트워크 모니터링 기법 분석
  - 쿼리 비용을 최소화하는 모니터링 기법
- 네트워크 트래픽 예측 및 최적화 연구
  - 트래픽 분산을 통한 네트워크 자원 관리 모델 도출
  - 네트워크상의 bottleneck 예상지점 및 현재 트래픽 분산 상태에 기반한 라우팅 기법 연구
  - 트래픽 예측 모델 및 데이터 트레이닝 세트의 정규화
- 네트워크 제어정보 동기화 기술
  - Raft 알고리즘 기반 동기화 기술
  - Hazelcast 활용한 Consensus 변경 인식 및 ICN 기반 네트워크 제어정보 동기화 기술
  - ICN 기반 제어정보 동기화 및 효율 테스트
- 네트워크 적용 시 성능 비교 요소 선별
  - 성능 비교 요소(네트워크 초기 구성, 지역 내 노드 개수, 경로 설정/복구 시간 등) 도출
  - 주요 비교 요소 항목을 통한 가중치 기반의 평가 지수

2.2 Policy Repository and Network Management Visualizer

- 오픈 소스 기반 Policy Repository
  - 비정형 DB를 지원하는 MongoDB 기반 Policy Repository 개발
  - 네트워크 기능배포/관리를 위한 Document 기반 비정형 데이터 모델 설계
  - 컨트롤러/Policy Repository 간 연동SW 및 네트워크 기능 배포 기술 개발
- Network Function 배포 및 관리 기술
  - NFV환경/자율 네트워킹을 위한 네트워크 기능 관리 방안 연구
  - 비정형 DB를 지원하는 NoSQL 기반 네트워크 기능 관리 아키텍처 연구
  - 비정형 DB 기반 Policy Repository 및 NoSQL API 기반 네트워크 기능 배포 기능 설계
- 인지기반 생체모방 알고리즘 연구 및 개발
  - 알고리즘의 기본 기능 도출 및 성능 비교 지표 도출
  - 알고리즘 개선을 위한 가중치 중심의 성능 평가 알고리즘 개발
  - NS-3 환경에서의 알고리즘 동작 시 시스템처리량과 패킷 전달 비율 측정
    - 알고리즘의 성능평가 수행
    - NS-3 Visualizer 개발

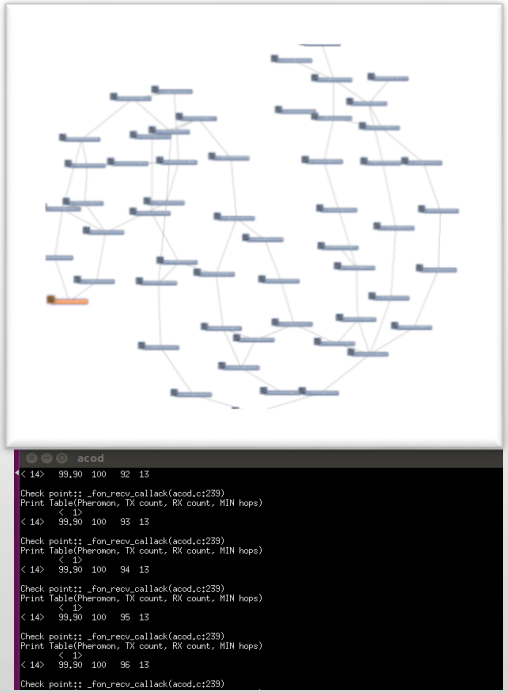




2.3 SDN/NFV/Controller/Autonomic Networks

성균관대학교 SUNG KYUN KWAN UNIVERSITY

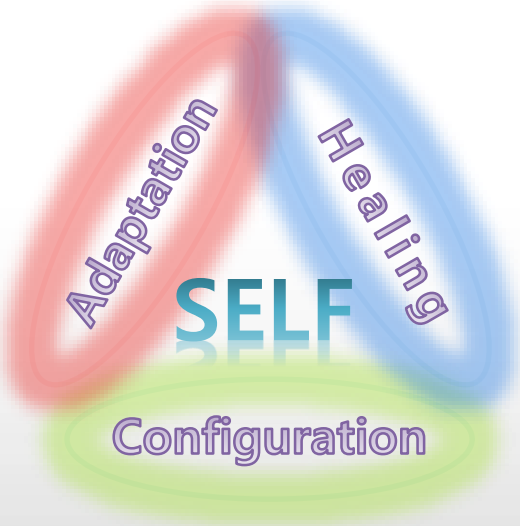
- Function Oriented Network 아키텍처 설계 및 개발
  - 네트워크 기능 호출을 위한 SmartPacket 구조 설계
  - 가변길이 메시지를 고려한 SmartPacket 생성 및 송수신 기술
  - 네트워크 기능 배포를 위한 OpenFunction 인터페이스
- Function Oriented Network Functions 설계 및 개발
  - TCP/Raw Socket 기반 Hop-by-Hop 전송 제어 기능
  - 전송경로 및 트래픽 정보 수집이 가능한 네트워크 모니터링 기능
  - Ant-Colony 기반 패킷 전송 및 라우팅
- Multi-RAT를 지원하는 Local ID 기반 메시지 전송 기술 개발
  - 통신링크에 독립적인 메시지 전송을 위한 Abstract Layer 설계 및 개발
  - L2 Abstract Layer와 호환 가능한 WiFi/Bluetooth/Ethernet 모듈
- 유/무선 네트워크 환경을 위한 테스트베드 구축
  - 요구사항 도출, 테스트베드 설계 및 구축
  - OpenFlow-Mesh (OF-mesh) 네트워크 아키텍처 설계
  - OF-mesh 네트워크에서 In-band communication향 프로토콜 개발
  - 요구 사항에 적합한 제어기 선정 (NOX, OpenDaylight, POX, ONOS, etc)
  - 선정된 컨트롤러를 통해서 연구 결과물 테스트 및 성능 향상을 도모함

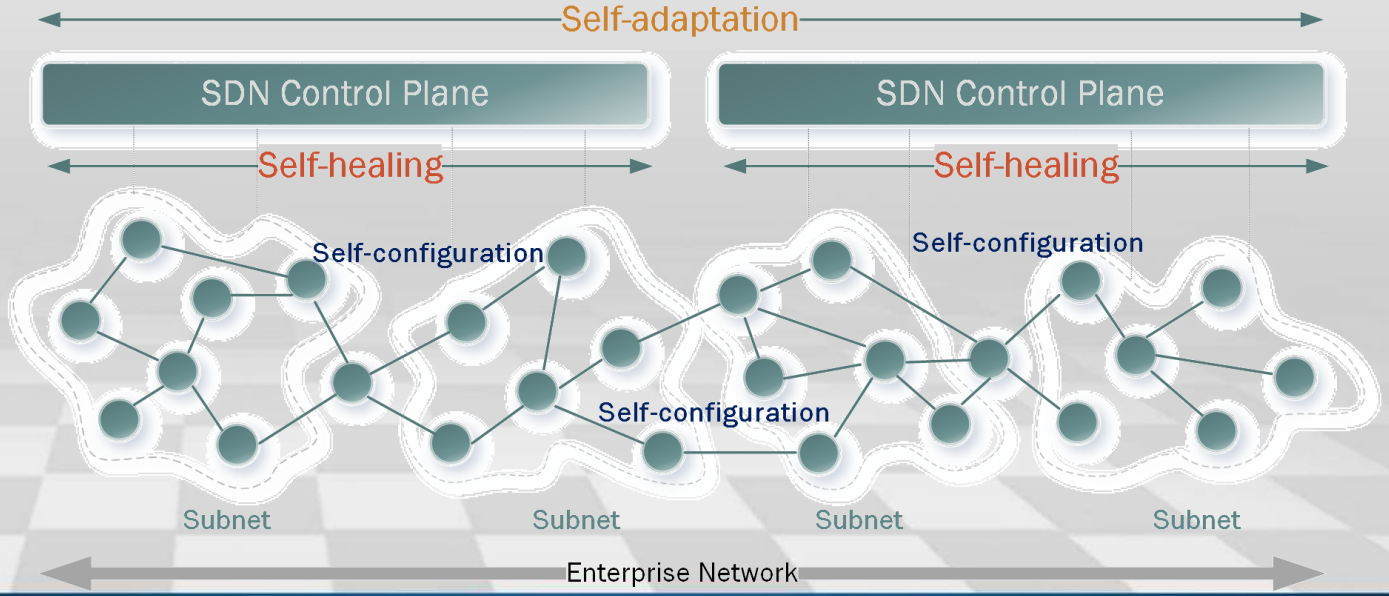


Fast Path Recovery Algorithm for Ant Colony Optimization (ACO) Routing

3. Self-Organization

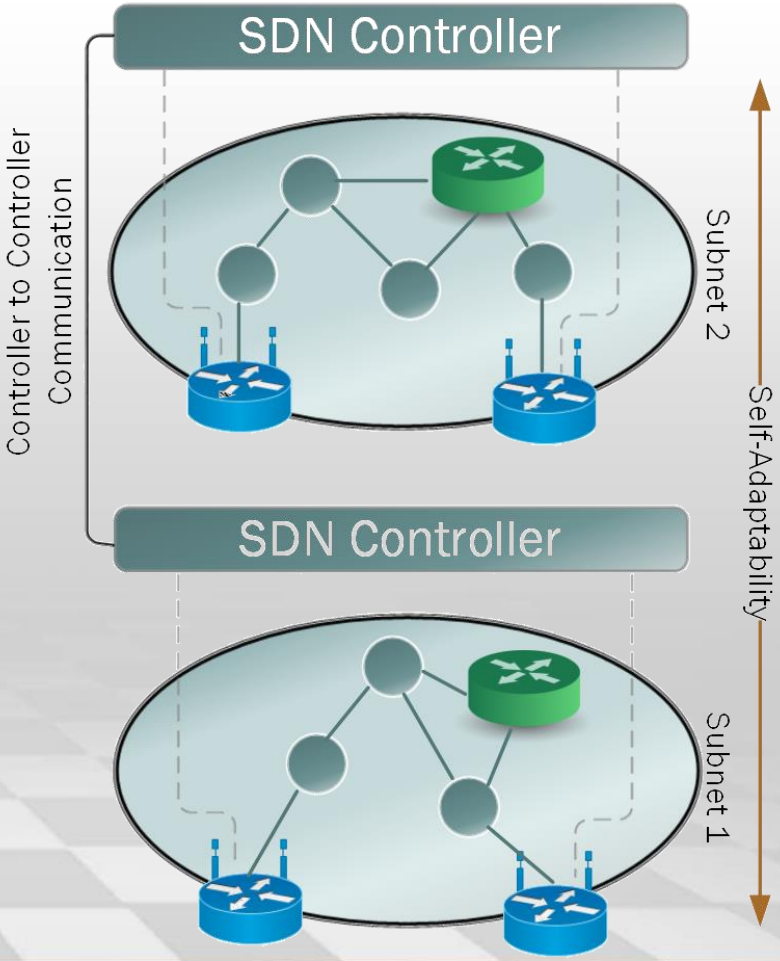
- **Autonomic properties of a network considered:**
  - Self-Adaptation
    - Anticipate and detect the changes in network state and manage accordingly
  - Self-Healing
    - Anticipate and detect disruptions in the network and manage accordingly
  - Self-Configuration
    - Anticipate and detect configuration changes in the network and manage accordingly





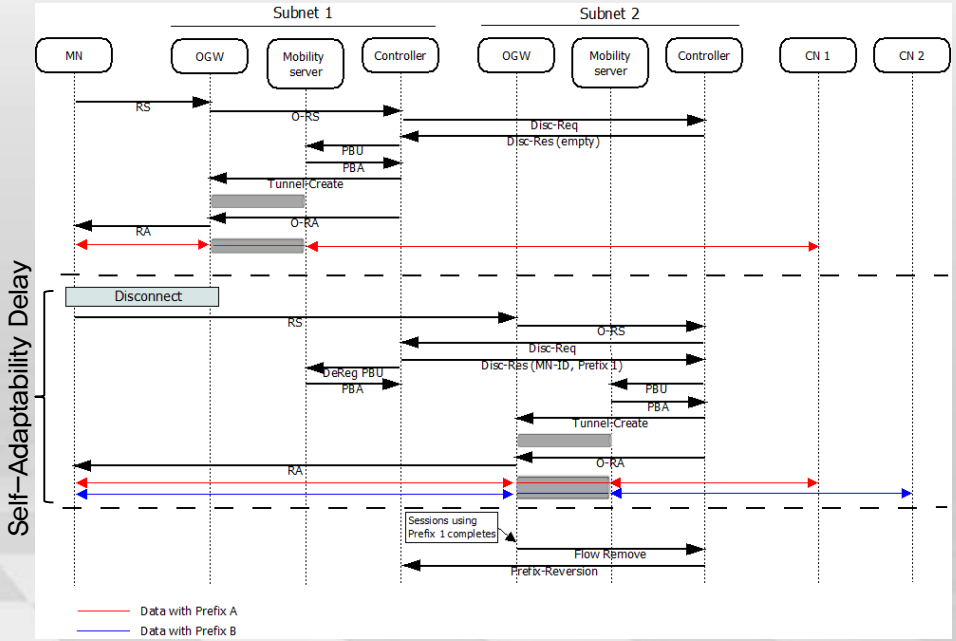
3.1 Self-Adaptation

- **Motivation:**
  - Self-Adaptation across different subnets in an enterprise network
- **Goal:**
  - Service adaptability by the networks as the flows moves across different subnets
- **Environment:**
  - Enterprise network where different subnets are controlled by different SDN controllers
- **Problem:**
  - Distributed communication between SDN controllers to establish self-adaptability across SDN domains



SDN based inter subnet self-adaptability of service

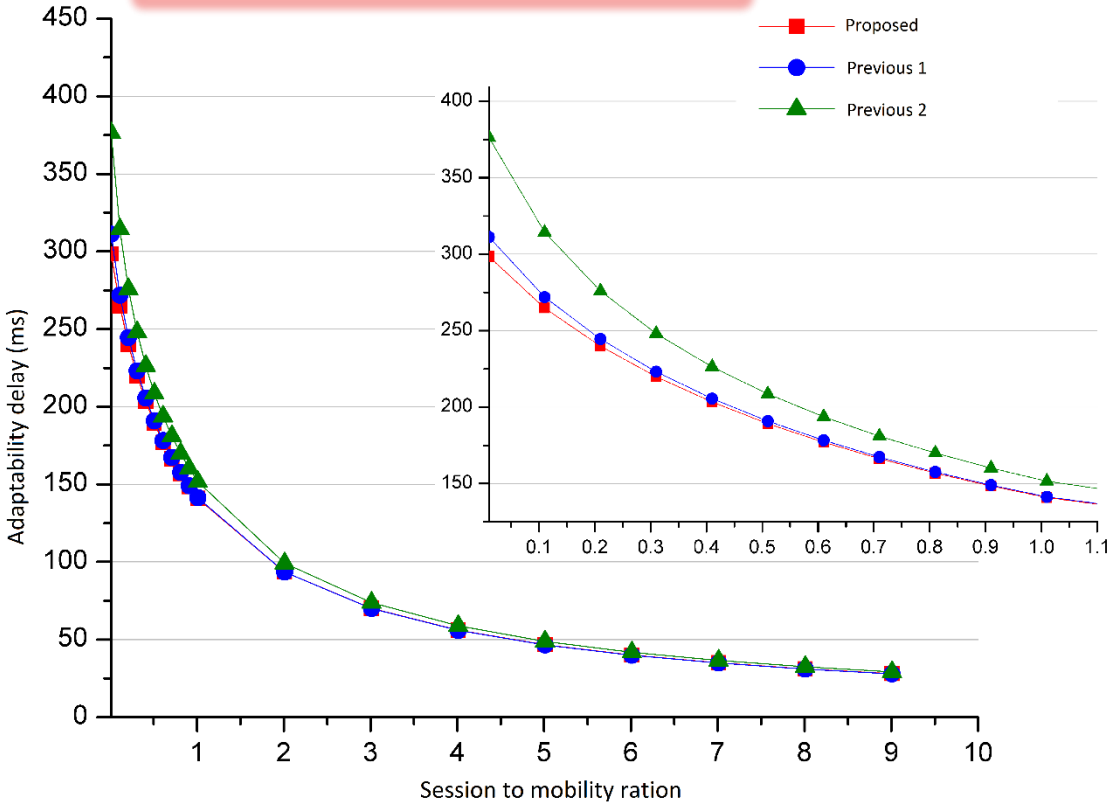
- **What**
  - Service adaptability across different SDN controlled subnets using on demand concepts
- **How**
  - Distributed communication between different domain controllers
  - Novel Controller to Controller Communication protocol (C3)
  - Service continuity across different domains through sharing of home network prefixes
- **Results**
  - Service continuity with minimum disruption of the service



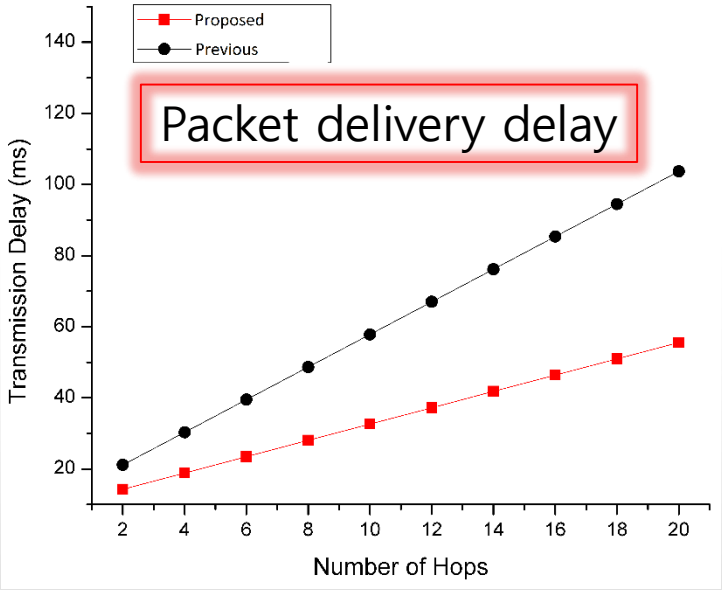
S. M. Raza, P. Thorat, R. Challa, H. Choo, D. S. Kim, "SDN based inter-domain Mobility for PMIPv6 with route optimization," *NetSoft Conference and Workshops (NetSoft), 2016 IEEE*

Performance Evaluation

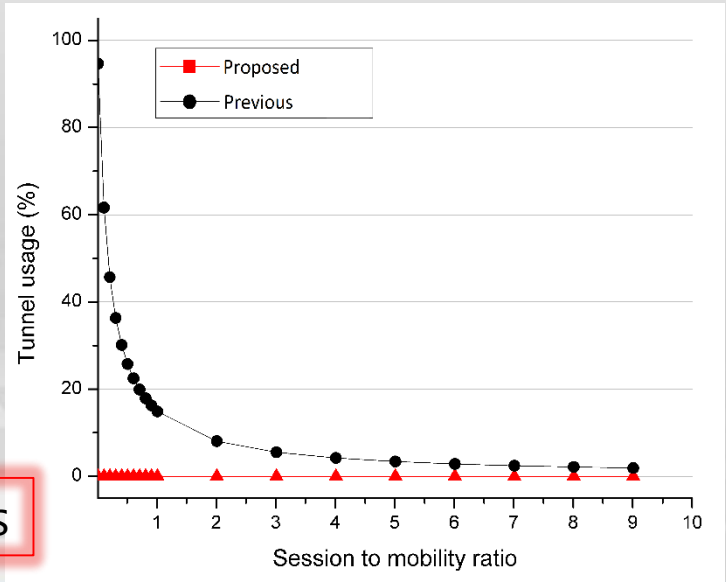
Adaptability performance



Packet delivery delay

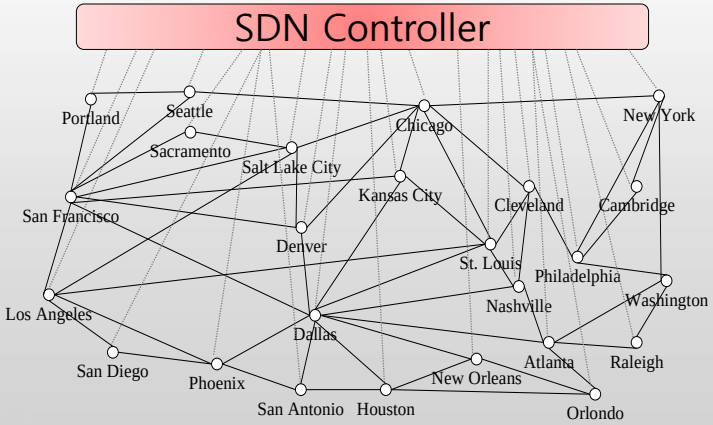


Tunneling overheads



3.2 Self-Healing

- **Motivation**
  - Rapid failure detection and recovery from failures in SDN
- **Goal**
  - Recovery from failure within delay under carrier grade requirement of 50 ms
- **Environment**
  - Emulated enterprise network with 25 switches and 52 links in Mininet, managed by a centralized NOX SDN controller



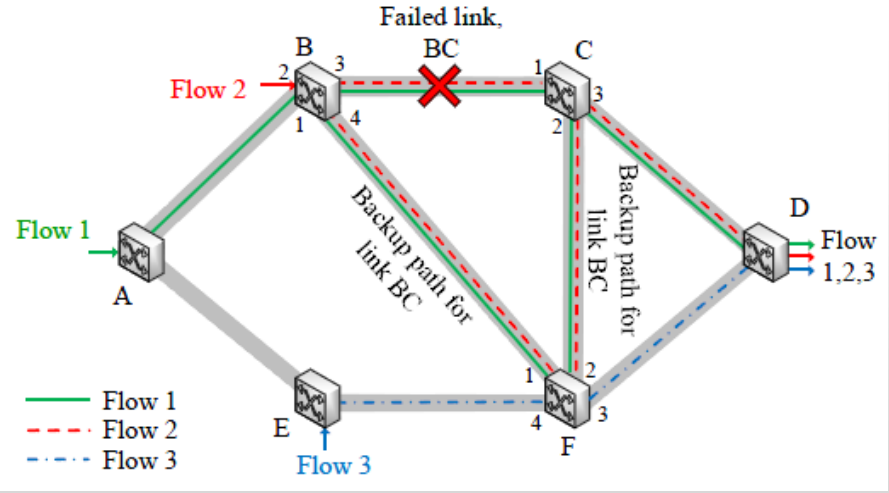
- **Problem**
  - Currently, the convergence time for distributed approaches is higher than the carrier grade requirement
  - Even using SDN, the controller assisted recovery time depends on the number of flows to be recovered
  - Flow table capacity stretches to the limits with flow rules for every disrupted flow

Link Failure Handling in SDN

- **What**
  - Rapid recovery from single link failures in the OpenFlow-based networks

- **How**
  - Our proposed* network design
    - achieves recovery within 50 ms without overwhelming the network with recovery specific control messages
    - Reduction in the memory requirement of the switch by aggregating the disrupted flows using VLAN tagging

- **Results**
  - Link recovery within 2 ms approx.
  - 99% reduction in flow table capacity requirement



Flow table

IP src	IP dst	InPort	VLAN ID	Action
192.168.1.1	192.168.1.8	*	*	Forward packet to group 3
192.168.1.2	192.168.1.9	*	*	Forward packet to group 3

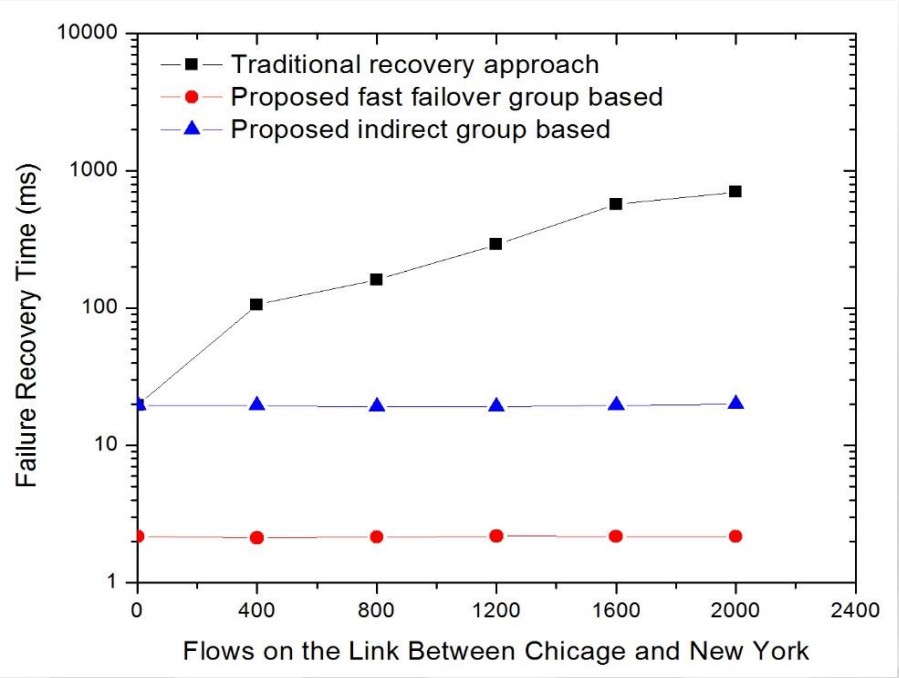
Group table

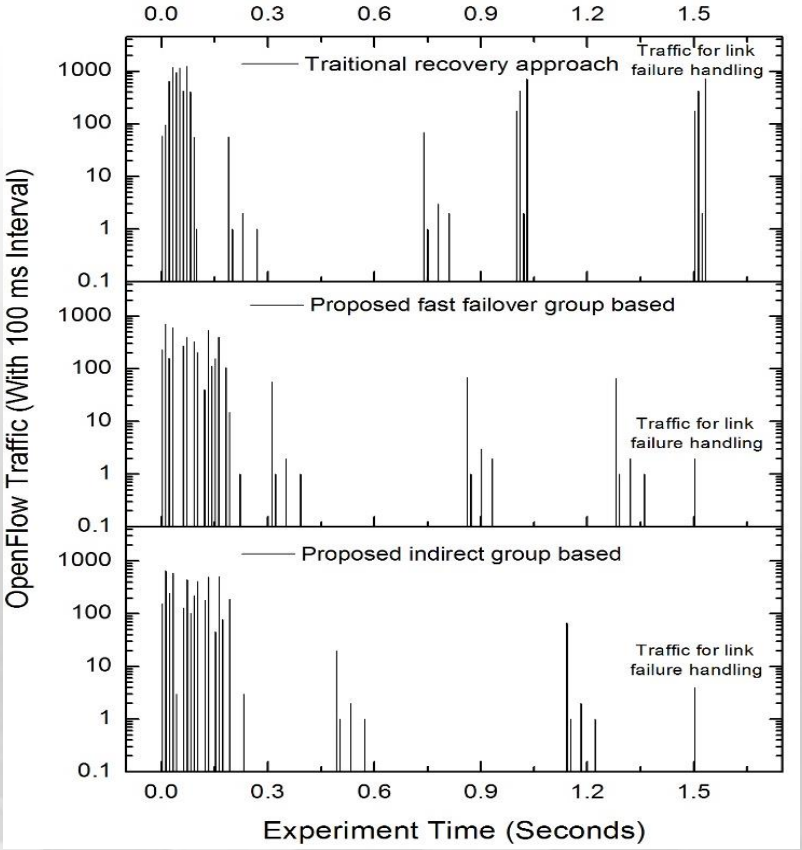
Group ID	Group type	Action buckets	
3	Fast failover: Execute bucket 2 on port 3 failure	1: Output to port 3	} Primary bucket
		2: Output to port 4	} Alternate bucket

P. Thorat, S. M. Raza, H. Choo and D. S. Kim, "Rapid Recovery from Link Failures in Software Defined Networks," under review in the Journal of Communication Network

Performance Evaluation

- Our proposed approaches achieves rapid recovery with minimal intervention from the controller



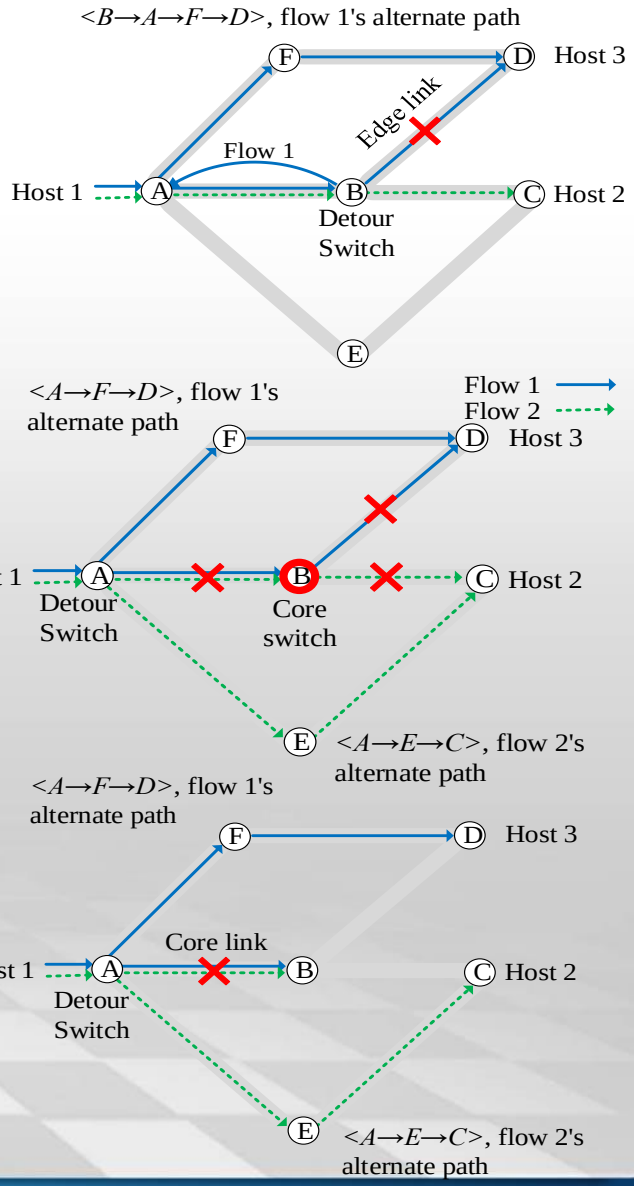


Failure recovery time

Total traffic flow at the controller

Joint Link and Switch Failure Handling in SDN

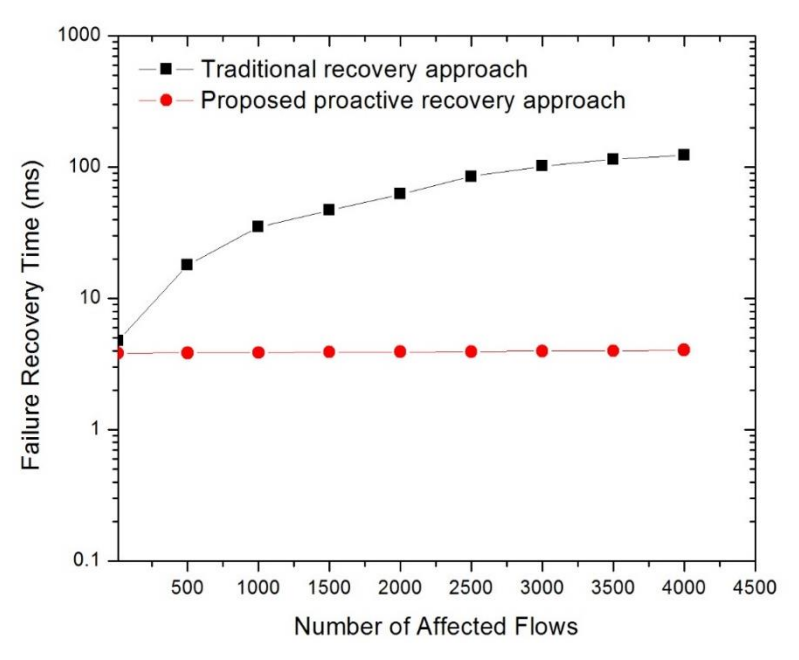
- **What**
  - Rapid joint link and switch failure handling mechanism by using the indirect group type in the OpenFlow switches
- **How**
  - Enhanced failure recovery mechanisms with flow grouping and forwarding rules aggregation methods for rapid and lightweight failure handling
  - Proposed# Proactive operation mode at the controller (autonomous)
- **Results**
  - link recovery within 4 ms approx.
  - 99% reduction in flow table capacity requirement

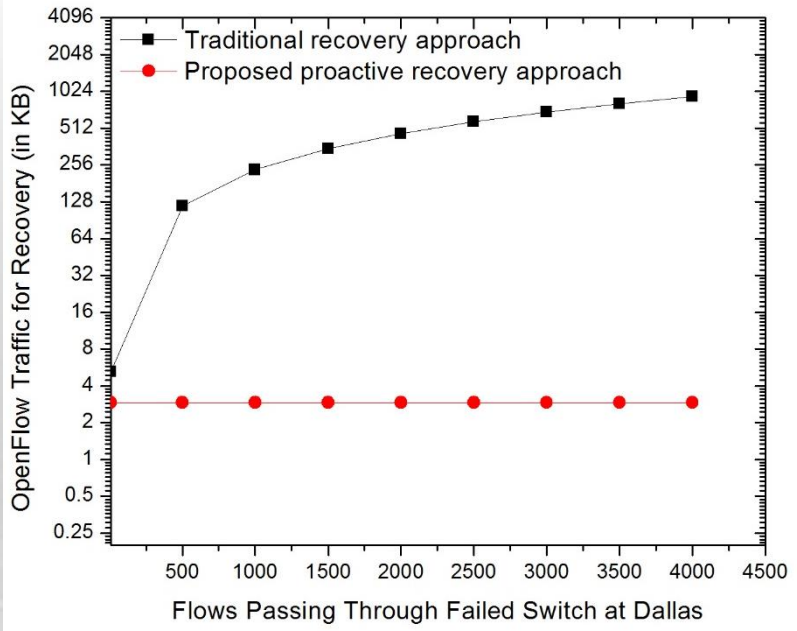


P. Thorat, R. Challa, S. M. Raza, D. S. Kim, and H. Choo, "Proactive Failure Recovery Scheme for Data Traffic in Software Defined Networks," *NetSoft Conference and Workshops (NetSoft), 2016 IEEE*

Performance Evaluation

- **Our proposed approaches**
  - Performs recovery within 50 ms time interval
  - Requires minimal intervention from the controller



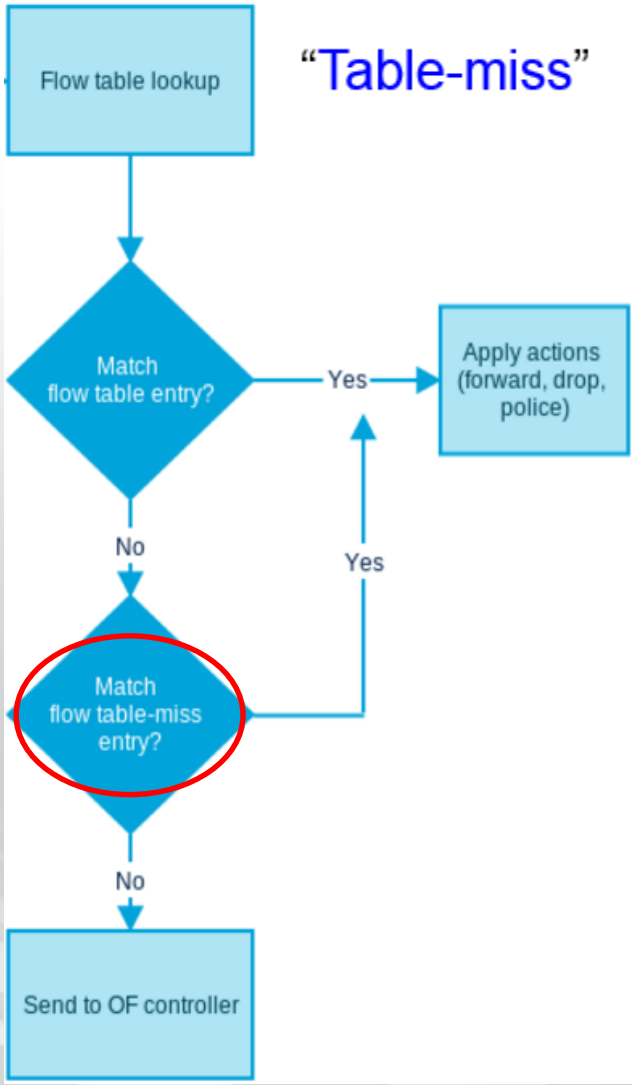


Failure recovery time

Control traffic for recovery from controller

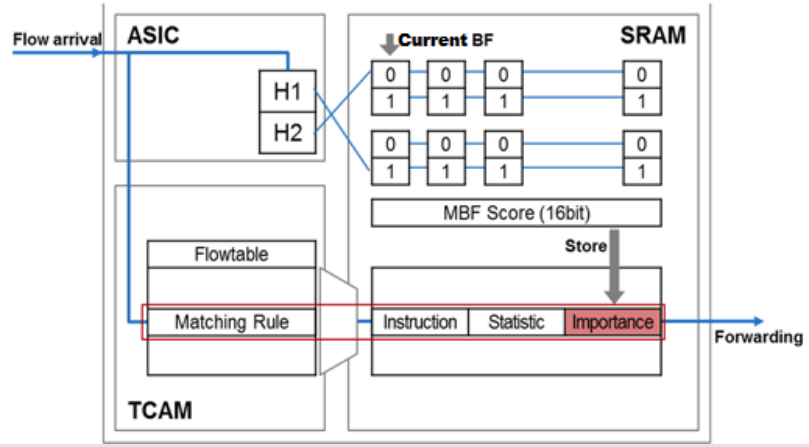
3.3 Self-Configuration

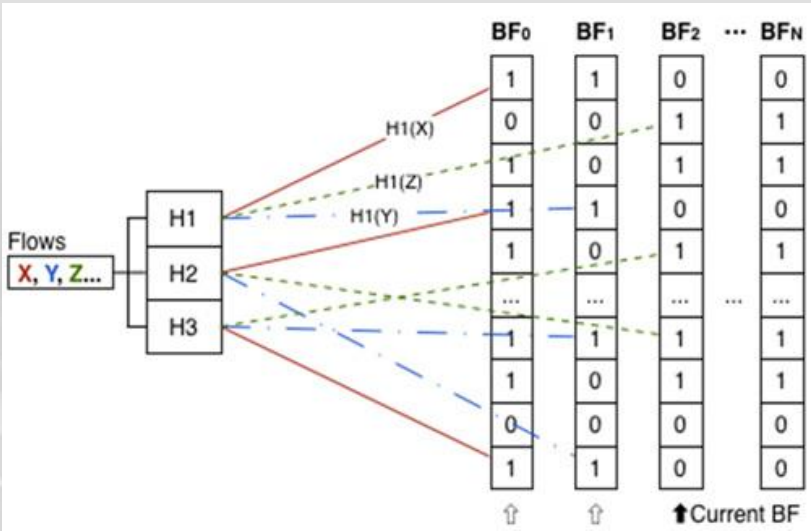
- **Motivation**
  - Intelligent self-configuration and software defined management of resources in Network Function Virtualization Infra. (NFVI) layer
- **Goal:**
  - To reduce configuration related signaling overhead
  - Enhanced resource utilization by identifying and replacing stale configurations with important ones
- **Environment:**
  - Virtualized OpenFlow switches with limited flow – table memory (resource)
- **Problem:**
  - To address "Table-Miss" problem by retaining important flow-entries, resulting in lower control signaling overhead and enhanced utilization of switch resources (TCAM memory)



Efficient Flow Table Management

- **What**
  - Smart self–configuration strategy for efficient hardware resource management in virtualized switches
- **How**
  - A highly efficient data structure – Column Major form of Multi–Bloom Filter (MBF) to identify "important" flow entries
  - MD5 based effective hash functions
  - Efficient resource (flow–table) utilization using OpenFlow v1.4 "Eviction" feature
- **Results**
  - Achieved 37% higher Table Hit–Ratio compared to LRU scheme (2K table size)





Challa, Rajesh, Yongseung Lee, and Hyunseung Choo. "Intelligent eviction strategy for efficient flow table management in OpenFlow Switches." *NetSoft Conference and Workshops (NetSoft), 2016 IEEE*

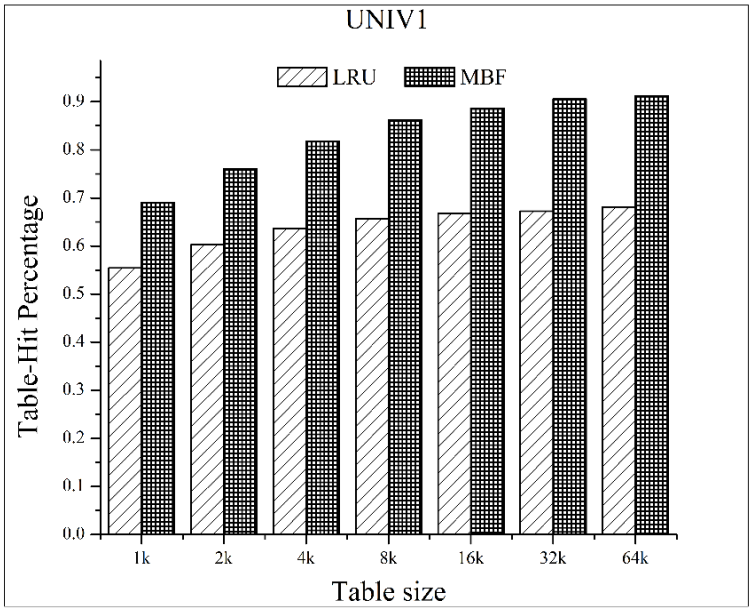
Performance Evaluation

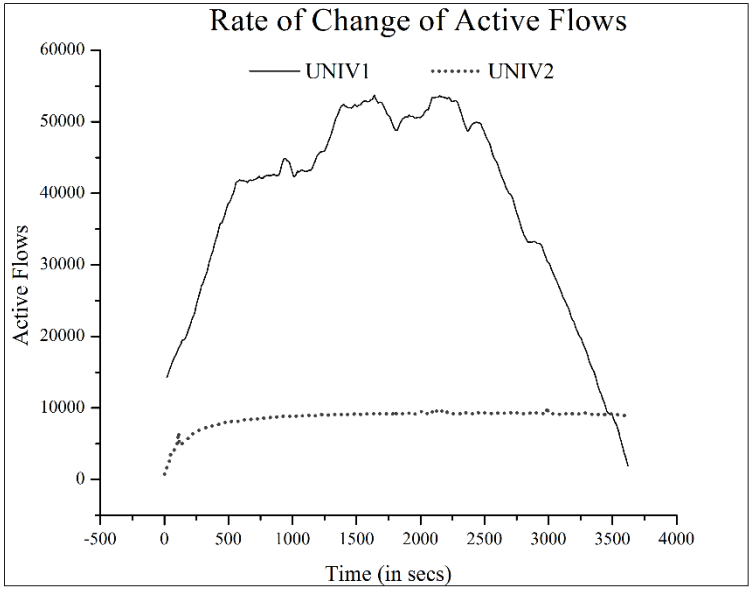
Table I: Simulation parameters and values used in MBF

System Model Parameters	Notation	Values
Number of Hash Functions	k	2
Bloom Filter Size	m	1024
Number of flows (incoming)	n	1600
Unit Time Period	ΔT	1 *Sec*
Number of Bloom Filters	N	16
Logging Duration	D	16 *Secs*

Table II: Data center traffic trace details used as an Input for our simulation.

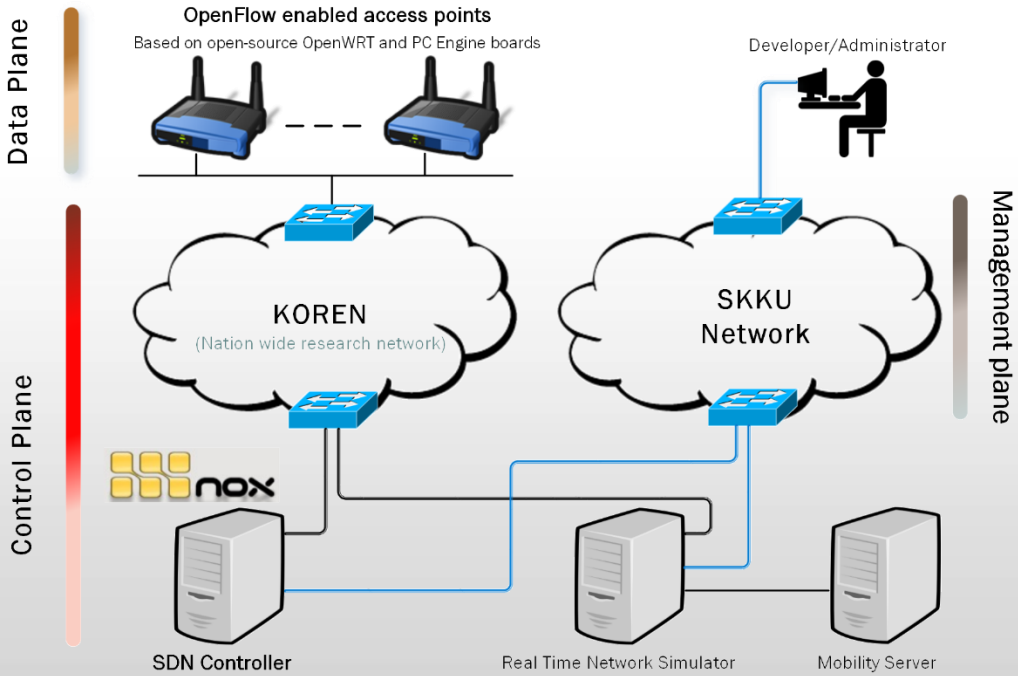
Parameters	UNIV1	UNIV2
Duration	65 *Mins*	77 *Mins*
Unique Flows	283,973	56,776
Number of Packets	19,855,388	47,668,207
Size of Packets (GB)	13.0	35.9
Active Flows	$40 \sim 50K$	$8 \sim 12K$



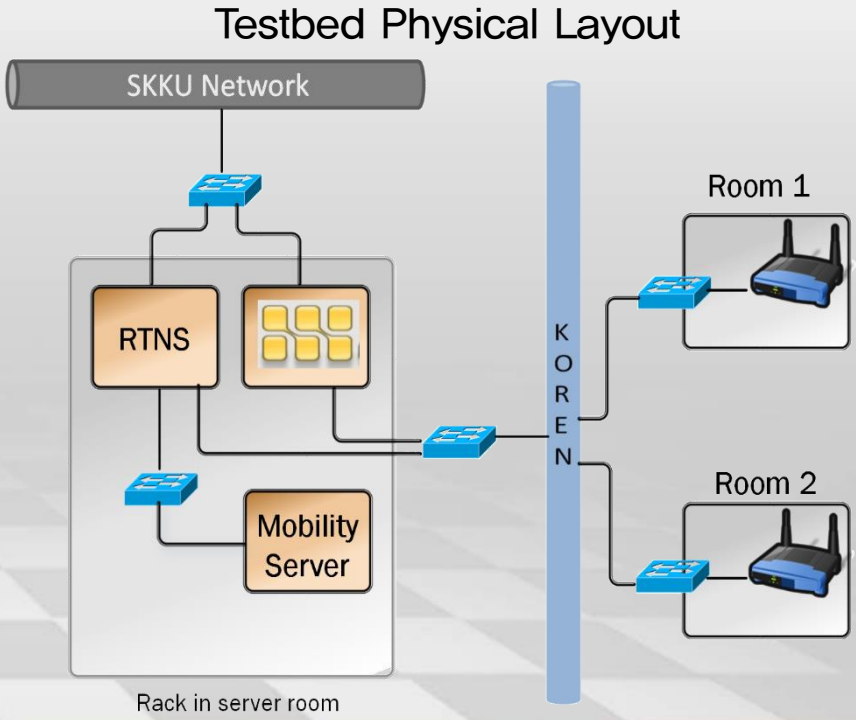


4. Testbeds and Open Sources

SDN Testbed with support for mobility



Testbed Architecture



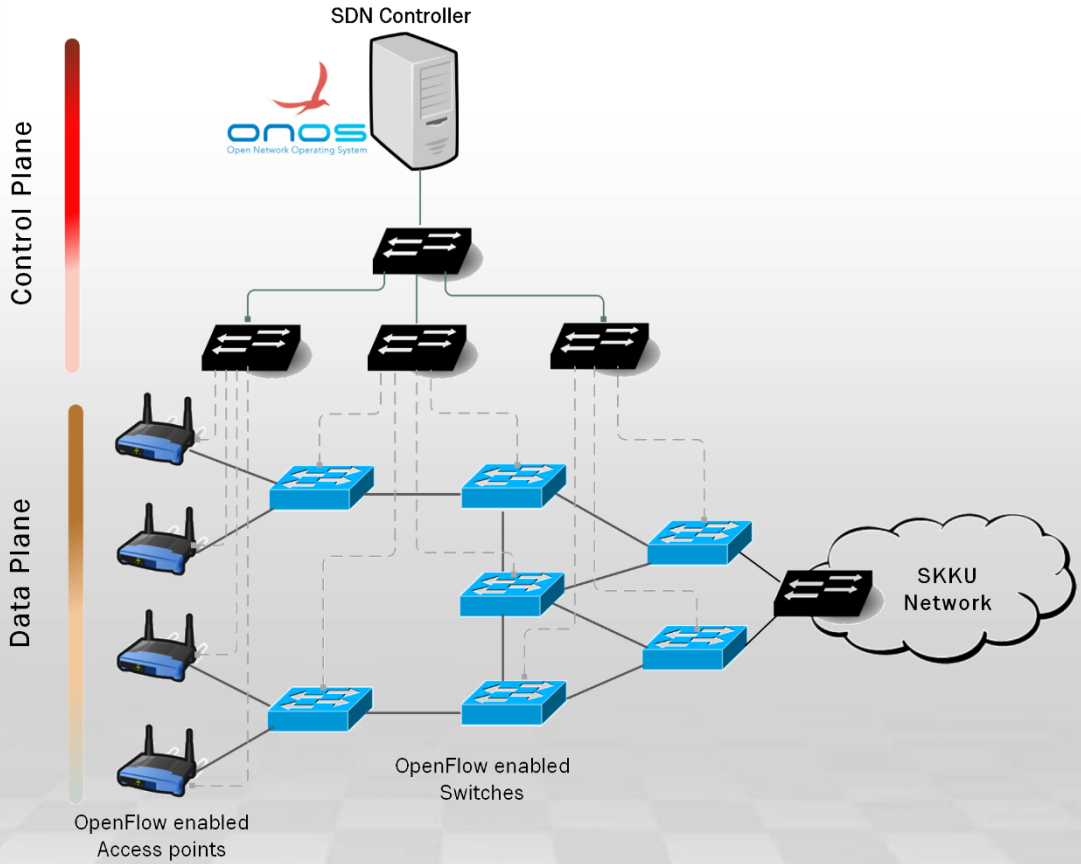
Open sources used:

- NoX by Nicira (SDN controller)
- OpenWRT (switch OS)
- OpenSoftSwitch (OpenFlow)

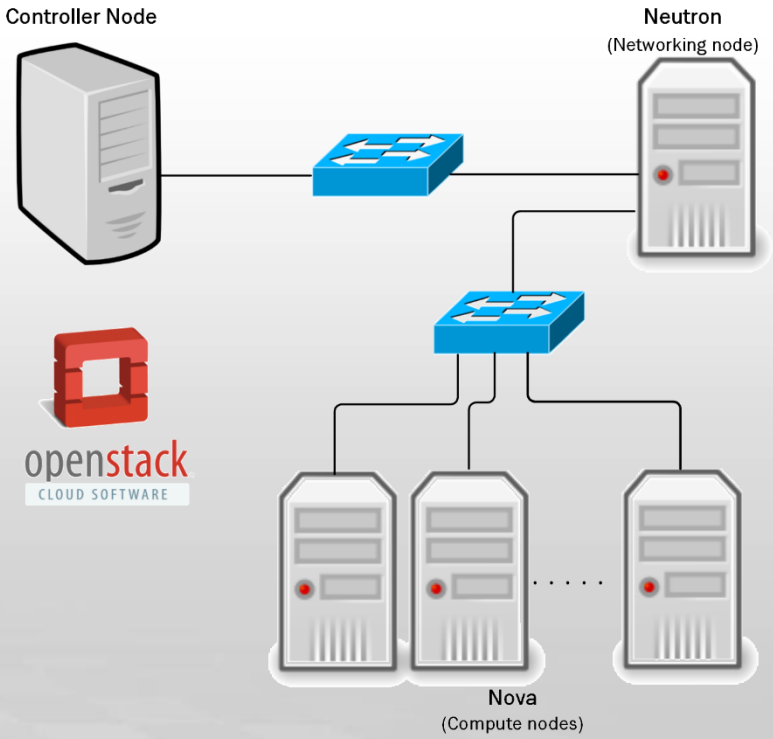
4. Testbeds and Open Sources

Ongoing Testbed developments

1. Extension of SDN Testbed



2. NFV Testbed



Open sources used:

- ONOS (SDN controller)
- Raspberry pi (switch hard and OS)
- Open vSwitch (OpenFlow)

Concluding Remarks

- Research on Self–* attributes of autonomic networks
- Novel solutions and performance improvements under each Self–* attribute of autonomic networks
- Preliminary Hybrid SDN testbed development
- Enhancement to full SDN testbed
- Development of preliminary NFV testbed with OpenStack
- Continuous enhancements in NFV testbed by adding more plugins and compute nodes
- Integration of SDN and NFV testbeds

Q & A