머신러닝을 이용한 서버 장애 예측 기반 VNF Live Migration

정세연, 유재형, 홍원기
포항공과대학교 컴퓨터공학과

{jsy0906, jhyoo78, jwkhong}@postech.ac.kr

A VNF Live Migration method based on Server Anomaly Prediction using Machine Learning

Seyeon Jeong, Jae-Hyoung Yoo, James Won-Ki Hong
Department of Computer Science and Engineering, POSTECH

요 약

VM (Virtual Machine) live migration 은 동작중인 VM 을 중단시간을 최소화하면서 다른 서버로 이전시키는 기술을 말한다. 클라우드 데이터센터에서는 워크로드(workload) 및 네트워크 트래픽에 대한 로드 밸런싱 목적, 가동되는 서버와 스위치 개수를 최소화하여 전력 소비량을 감소시키는 목적, 서버의 하드웨어 및 소프트웨어 업데이트를 위한 VM 이전과 같은 유지보수 목적 등으로 VM live migration 을 활발히 사용하고 있다. 특히 고장 및 장애 징후를 사전에 탐지해서 예방(prevention) 또는 완화(mitigation)를 위한 수단으로 핵심적이다. 본 논문에서는 NFV (Network Function Virtualization) 환경에서 서버 장애에 따른 제어 복잡도 및 QoS 저하를 낮추기 위한 목적으로, 서버의 장애 예측에 기반한 VNF (Virtual Network Function) live migration 방법을 제안한다. 이는 각 서버에 대한 모니터링 데이터를 머신러닝으로 학습하여 자원 상태 변화에 대한 예측 모델을 생성하고, 예측 결과를 바탕으로 미래에 장애 가능성이 가장 낮은 서버를 최적의 migration 목적지로 선정한다. NFV 테스트베드에서의 실험을 통해 다른 migration 방법 대비 제안하는 방법의 향상된 장애 완화 효과를 검증한다.

Ⅰ. 서론

가상화 기술의 발전으로 다양한 서비스 어플리케이션들이 가상머신 (VM, Virtual Machine) 형태로 동작하게 되면서, 확장성(scalability), 기민성(agility), 신뢰성(reliability) 등 오늘날 클라우드 컴퓨팅 기술의 핵심 목표를 달성할 수 있게 되었다. VM migration 은 동작중인 VM 을 다른 서버로 이전시키는 기술로, 대상 VM 의 특정 시점의 이미지 파일(스냅샷)을 목적지 서버로 복사한 후 하이퍼바이저(hypervisor)를 통해 이미지를 불러들여 VM 을 다시 가동시킨다. VM 을 물리적으로 다른 서버로 이동시킬 수 있지만 이미지 전송 및 VM 이 부팅을 하는 동안에는 서비스를 제공할 수 없다는 문제가 있다.

이러한 문제점을 극복하기 위한 VM live migration [1] 기술은 VM 스냅샷을 목적지 서버로 복사해서 복사본 VM 의 가동 준비가 완료될 때까지 원본 VM 의 동작(서비스)를 유지시킨다. 따라서 두 VM 간의 메모리 동기화를 위한 freezing 시간 및 네트워크 경로 재설정을 포함하는 최소한의 서비스 중단시간(downtime)을 가진다 (그림 1). 이러한 VM live migration 기술은 특히 클라우드 데이터센터에서 워크로드(workload) 및 네트워크 트래픽에 대한 로드 밸런싱 목적, 가동되는 서버와 스위치 개수를 최소화해서 전력 소비량을 감소시키는 목적(VM consolidation), 서버의 하드웨어 및 소프트웨어 업데이트와 같은 정기적인 유지보수를 위해 동작중인 VM 을 다른 서버로 이전시켜서 서비스 제공을 유지하는 목적[2] 등으로 활발히 사용되고 있다. 이에 더하여 앞으로의 VM live migration 기술은 서비스 중단을 야기하는 고장 및 성능 저하를 일으키는 장애 징후를 사전에 탐지하여, 서비스 중단 시간과 사용자가 체감하는 성능 저하를 최소화하는 방향으로 발전되어야 한다. 이러한 고장 예방의 예시로, 장기간 과도한 CPU 사용률을 보이는 과열된 서버의 VM 에 대한 migration 작업을 들 수 있다.

NFV (Network Function Virtualization) 환경 또한 VNF (Virtualized Network Function)에 대한 live migration 을 통해 다양한 관리 효과를 얻을 수 있다. 하지만 NFV 는 SFC (Service Function Chaining)를 통해 여러 VNF 가 연관되어 있고, 다른 NFV 제어 기능(auto-scaling 등)들과 상호작용해야 하므로, 전통적인 VM live migration 방법을 그대로 도입할 수는 없다.

따라서 본 연구에서는 NFV 환경에서 서버 장애에 따른 제어 복잡도 및 QoS 저하를 낮추기 위한 목적으로, 서버 장애에 대한 예측에 기반한 VNF live migration 방법을 제안한다. 본 연구의 contribution 은 아래와 같다.

- 각 서버에 대한 모니터링 데이터를 머신러닝으로 학습하여 서버의 자원 상태 변화에 대한 예측 모델을 생성함.
- VNF live migration 을 통한 서버 장애의 대응 과정에서, 모든 서버의 자원 상태 변화를 예측하여 장애 가능성이 가장 낮은 서버를 migration 목적지로 선정하는 방법을 제안함.
- OpenStack 기반 NFV 환경에서 SFC 를 구성하여 제안하는 VNF live migration 방법을 검증함. 제안하는 방법은 다른 방법 대비 QoS 지표에서 약 15% 수준의 향상을 보임.

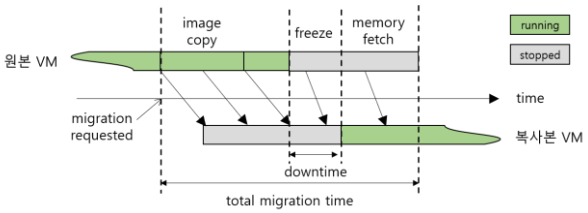

그림 1. VM live migration 동작 과정

Ⅱ. 관련 연구

VM/VNF live migration 에 대한 연구 방향은 아래와 같이 크게 세 가지로 나눌 수 있다. VM/VNF live migration 은 클라우드 데이터센터 관리에 필수적인 기능이지만 일시적인 서비스 중단이 발생하므로, migration 비용을 사용목적에 맞게 적절히 산정하여 적시 적소에만 수행할 필요가 있다.

1. 로드 밸런싱

[3]은 서버 및 네트워크의 자원 사용량에 대한 모니터링을 바탕으로 과부하(overloaded) 상태의 서버 및 링크를 식별하고, 관련된 VNF 들을 부하가 작은 다른 서버로 이전시켜 부하분산을 수행하는 방법을 제안하고 있다. [4]는 SFC reliability 에 대한 정의를 바탕으로 SFC 의 reliability 를 높일 수 있는 대상 VNF 와 목적지 서버를 식별하여 migration 을 수행하며, 결과적으로 migration 을 통해 과부하를 해소하는 방법을 제안하고 있다. [3], [4] 모두 SFC 성능 지표가 향상됨을 보이나, 시뮬레이션 환경에서 수행되어 전체 네트워크에 대한 상태 수집이 가능한 것으로 가정하고 있으며 실제 migration 기능은 시험하지 않았다는 한계가 있다.

2. 전력 소비량 감소

[5]는 데이터센터에서 서비스 성능을 보장하면서 전력 소비를 줄이도록 VM 을 배치하는 방법을 다룬다. 서로 상충되는 지표인 서비스 성능 및 전력 소비량을 강화학습의 보상(reward)에 포함시켜 최대 보상을 가지는 migration 정책(대상 VM 및 목적지 서버)을 학습한다. 강화학습을 통해 트래픽 변화 등 다양한 동적 조건에서 기존 휴리스틱(heuristic) 방식 대비 VM migration 횟수 및 전력 소비량이 감소됨을 보이나, NFV 환경 및 서버 고장/장애 상황은 고려하지 않고 있다.

3. 고장/장애 대응

[6]은 MEC (Mobile Edge Computing) 환경에서 core cloud 에 있는 특정 VNF 의 장애 발생 시, 해당 VNF 를 edge cloud 로 이전하여 운용하는 정책의 비용 적절성 문제를 다룬다. 이를 위해 migration 하지 않을 때의 operator loss 및 migration 할 때의 service path 재설정을 포함한 migration cost 를 모델링하여 전체 비용이 적은 정책을 따른다. 이 과정에서 edge user mobility 예측에 머신러닝을 사용한다. [7]은 운용중인 데이터센터에서 수집된 각 서버의 자원 상태 및 운영 정책을 머신러닝 feature 로 학습하여 미래의 고장 가능성을 예측한 뒤, 고장 점수가 상위권인 서버에서 동작하는 VM 을 하위권 서버로 migration 한다. [6], [7] 모두 migration 정책 결정에 머신러닝을 활용하며, 특히 [7]에서는 고장 예측을 통한 proactive VM live migration 의 필요성을 보이고 있다.

NFV 고장 대응 관련 VNF backup 방법은 master VNF 인스턴스에 대한 slave VNF 인스턴스를 별도로 두어, master VNF 의 고장 시 즉각적으로 slave VNF 로 절체한다 [8]. VNF live migration 방법 대비 중단시간이 작다는 장점이 있으나, master/slave 인스턴스의 실시간 동기화 비용과 over-provisioning 문제가 존재한다.

Ⅲ. 설계 및 구현

본 연구에서 제안하는 서버 장애 예측 기반 VNF live migration 시스템은 크게 Monitoring 모듈, Prediction 모듈, Migration 모듈로 구성되어 NFV MANO (Management and Orchestration) 시스템의 구성요소로 동작한다 (그림 2). NFV 환경은 OpenStack Victoria 릴리즈를 기반으로 VNF 및 SFC 의 제어를 수행하는 별도의 NFVO (NFV Orchestrator) 모듈을 개발하였다. 또한 향후 CNF (Cloud-native Network Function)에 대한 migration 지원을 고려하여 컨테이너(container)를 지원하도록 OpenStack Zun [9] 서비스를 구성하였다.

Monitoring 모듈은 서버 및 VM 의 자원 사용량과 가용 물리/가상 자원에 대한 모니터링 데이터를 주기적으로 수집하여 데이터베이스에 저장한다. 이를 구현하기 위해 각 서버에 collectd 를 모니터링 에이전트로 설치하여 CPU, 메모리, 하드디스크, 네트워크 I/O 사용량 등을 매초마다 수집하고 시계열 데이터베이스인 InfluxDB 에 저장하였다. VM 생성 시 collectd 가 설치된 VNF 이미지를 사용하여 관련 모니터링 프로세스를 자동화하였다. 또한 다른 모듈로부터 다양한 형태의 데이터 쿼리(예, 지난 1 시간 동안 특정 VM 의 CPU 사용량)에 응답하기 위해 swagger 기반 OpenAPI 서버 형태로 Monitoring 모듈을 구축하였다.

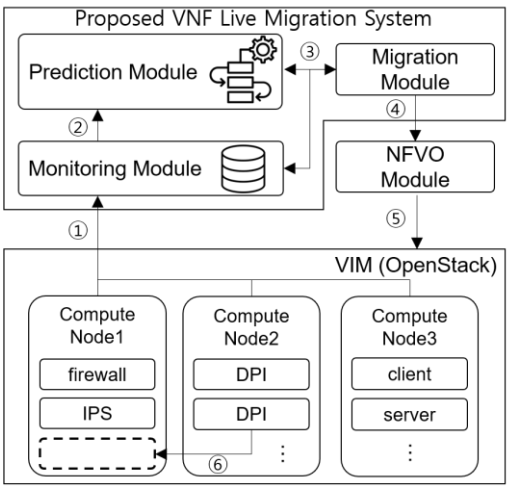

그림 2. 서버 장애 예측 기반 VNF live migration 구조도

Prediction 모듈은 서버 단위로 수집된 모니터링 데이터를 학습하여 서버의 자원 상태를 예측하기 위한 머신러닝 모델을 생성한다. 학습을 끝낸 모델은 Migration 모듈의 요청에 따라 미래 시점의 서버 자원 상태에 대한 예측을 바탕으로 장애 가능성이 가장 낮은 서버를 최적의 migration 목적지로 선정한다. 즉 전체 모델은 일정 기간 모든 서버의 자원 상태를 나타내는 벡터를 입력 받아 최적의 migration 목적지 서버를 출력하는 classification 문제로 정의된다. 이를 위해 서버의 자원 상태에는 일정한 패턴이 있음을 가정한다 (IV. 1 장 실험 시나리오 참고). 본 연구에서는 migration 목적지 선정을 위한 서버 장애 예측에 서버 수준의 모니터링 데이터만을 사용했으나, VM/VNF 수준의 데이터 및 SFC 정보(경로, 길이, SLA 등)를 결합해서 최적의 migration 대상 VNF 를 찾아볼 수 있다 (향후 연구). Prediction 모듈의 전체 모델을 구현하기 위해 Python scikit-learn 라이브러리의 SVM (Support Vector Machine) 알고리즘을 사용하였다.

Migration 모듈은 Monitoring 모듈로부터 VNF live migration 가동 조건(예, 특정 서버의 CPU 사용률 threshold 초과)이 발생하면 최적의 VNF live migration 정책을 결정한다. 본 연구에서는 해당 서버에서 동작하는 VNF 중 하나를 migration 대상으로 선정하며, Prediction 모듈에 질의하여 최적의 migration 목적지 서버를 선정한다. 최종적으로 OpenStack VM live migration API 에 (대상 VNF ID, 목적지 서버 ID)로 구성된 파라미터를 전달한다. 이 과정에서 NFVO 모듈과 연동한다.

IV. 검증

본 연구의 실험에 사용된 서버의 사양은 Intel Xeon 2.67GHz (12 코어), 24GB RAM 이며, OS 로 Ubuntu 18.04 를 사용했다. VNF 가 동작하는 VM 은 1 개의 CPU 코어 및 1GB 메모리를 가지도록 통일하였다. SFC 는 클라이언트의 서비스 요청이 firewall, IPS, DPI 를 순서대로 거쳐 웹 서버에서 응답을 반환하는 서비스를 가정했으며, OpenStack networking-sfc API 를 사용하여 구현하였다.

1. 실험 시나리오

실험에서 VNF live migration 은 특정 서버의 평균 CPU 사용률이 90% 이상일 때 가동된다. 이는 서버 장애 상황의 대표적인 예시로서, 해당 서버의 모든 CPU 코어가 saturated 된 상황을 의미하며 현재 동작중인 VM/VNF 또는 추가적인 서비스 요청에 대해 성능을 보장하지 못하는 상황을 가정한다. 이러한 서버 과부하 상황은 VM 이 예상치 못하게 down 되거나 CPU 과열로 인한 하드웨어 손상 문제로 이어질 수 있다 [10]. 이를 위해 다수의 SFC 에 트래픽을 동시 다발적으로 발생시키는 것에 더해, stress-ng 를 사용하여 모든 서버에 인위적인 부하를 주입하였다. 그림 3 은 시간에 따른 서버 부하 주입을 위한 학습 데이터의 예시를 보이며, 현재 tick 의 과부하 상태 서버에서 동작중인 VNF 를 미래 tick 에서 장애 가능성이 가장 낮은 서버로 migration 하는 정책(화살표)을 보인다. 주입되는 부하의 양과 패턴은 데이터센터에서 일정 주기에 걸쳐 반복되는 트래픽 및 워크로드를 반영하는 것으로 가정한다.

시간	Node1 CPU (%)	Node2 CPU (%)	Node3 CPU (%)	firewall	IPS	DPI
1	90	10	50	N1=>N2	N1	N3
2	100	10	50	N2	N1=>N1	N3
3	10	100	50	N2=>N3	N1	N3
4	50	85	10	N3	N1	N3
5	…			…		
…	확률적으로 일정 random 값 가감			최적 migration 정책 (정답지)		

그림 3. 부하 주입과 migration 정책의 학습 데이터 예시

2. 실험 결과

부하 주입을 통한 서버 자원 상태 예측 모델의 학습 이후, 유사한 부하 패턴을 테스트 데이터로 주입하여 제안하는 서버 장애 예측 기반 VNF live migration 정책을 검증하였다. 또한 migration 을 사용하지 않는 정책(no-mig), 목적지 서버를 임의로 선정하는 정책(random), CPU 사용률이 가장 낮은 서버를 선정하는 정책(least-cpu)과 QoS 성능을 비교하였다.

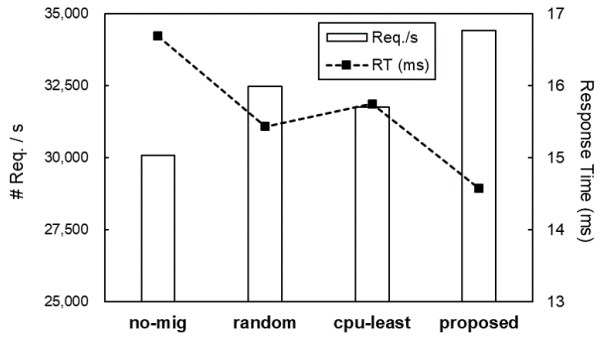

그림 4. VNF live migration 정책 별 SFC 성능

그림 4 는 각 VNF live migration 정책에 따른 SFC 의 성능 지표를 보인다. 초당 평균 request 처리량에 대해 no-mig 대비 random, cpu-least, 제안된 방식(proposed)은 각각 8%, 6%, 15% 증가를 보이며, 평균 response time 에 대해 8%, 6%, 13% 감소를 보였다. 주목할 점은 cpu-least 정책이 random 정책보다 성능이 낮은 것으로, 이는 CPU 사용률이 가장 낮은(least CPU utilization) 서버를 migration 목적지 서버로 선정하는 방식[3][4]이 직관적으로는 적절해 보이지만, 가까운 미래에 해당 서버에 장애(과부화 등)가 발생한다면 오히려 평균 성능이 저하될 수 있음을 의미한다. 제안하는 방식으로 선정된 migration 목적지 서버는 현재 시점에서는 최적이 아닐지라도 미래의 장애 가능성이 가장 낮기 때문에 평균 성능을 향상시킬 수 있다. 또한 본 실험에서는 CPU 과부하와 같은 (일시적인) 서버 장애 상황만을 고려했지만, 서버가 물리적으로 down 되는 고장 상황에서는 QoS 격차가 더욱 커질 것이다.

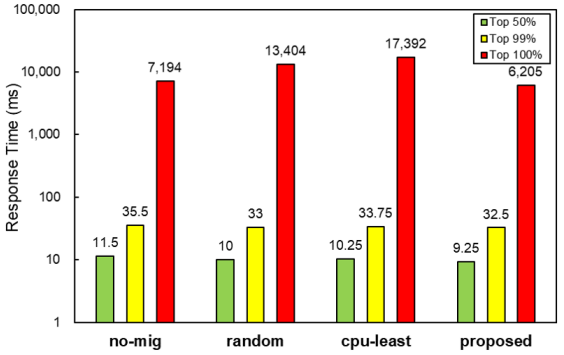

그림 5. VNF live migration 정책 별 response time 비율

그림 5 는 각 VNF live migration 정책에 따른 상위 50%, 99%, 100% response time 을 보인다. 상위 99% 까지의 response time 은 정책에 관계없이 정상적으로 낮은 값을 보이나, 상위 100%에 대해 random 및 cpu-least 정책에서 매우 높은 지연시간을 보이며, 제안하는 방식 또한 no-mig 와 비슷한 수치를 보인다. no-mig 에서 발생하는 tail latency 는 서버 장애에 따른 QoS 저하를 나타내는 반면, random, cpu-least 및 제안하는 방식의 tail latency 는 QoS 저하에 더해 migration 과정(그림 1)에서 발생하는 네트워크 경로 재설정에 따른 일시적인 packet loss 의 영향도 포함한다. 특히 VNF live migration 에서는 SFC 수준의 변경(SFC 경로 등)이 요구되므로 tail latency 에 민감한 서비스에 대해서는 migration 사용에 주의가 요구되며, 이를 줄이기 위한 관련 연구의 필요성을 제기한다.

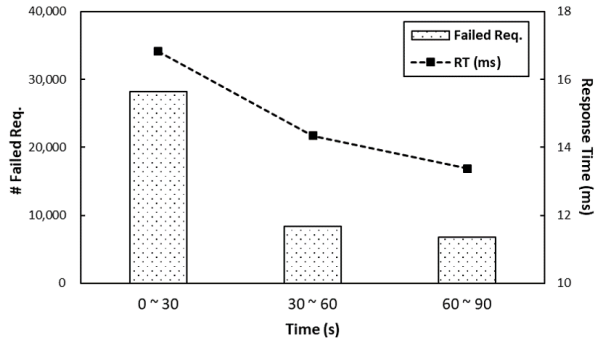

그림 6. VM live migration 요청 이후 시간 별 성능 변화

그림 6 은 동작중인 VM 을 live migration 했을 때 시간에 따른 timeout request 개수 및 평균 response time 의 변화를 보인다. 실험의 OpenStack 테스트베드에서 1 개의 CPU 코어 및 1GB 메모리를 가지는 Ubuntu VM 에 대한 live migration 요청 후 완료까지 평균적으로 45 초가 소요되었다 (OpenStack 로그 기준). 실험 결과에서 migration 요청 직후 30 초 동안은 장애 상태 서버의 원본 VM 에서 서비스를 처리하여 낮은 성능을 보이며, 30~60 초 구간에서 목적지 서버로의 VM migration 이 완료되어 성능이 향상됨을 확인할 수 있다. VM live migration 작업이 완료되는데 걸리는 시간은 VM 의 이미지 크기, 서버의 상태 등에 따라 달라질 수 있다.

V. 결론 및 향후 연구

본 연구는 VM/VNF live migration 기술 및 관련 연구를 소개하고, NFV 환경에서 서버 장애에 대한 완화 수단으로 서버 장애 예측 기반 VNF live migration 방법을 제안한다. 이는 서버의 자원 변화 상태를 머신러닝으로 학습해서 미래의 서버 장애 가능성을 예측하여 가장 적합한 migration 목적지 서버를 선정한다. OpenStack 기반 NFV 테스트베드에서 제안하는 방법의 장애 완화 효과를 QoS 측면에서 비교 검증하였다.

본 연구는 NFV 환경에서의 인공지능 기반 VNF live migration 기술에 대한 사전 연구로서 다양한 향후 연구 방향을 가진다. 첫째, 본 연구는 일종의 reactive VNF live migration 방식으로 임계치(threshold)를 통해 장애 여부를 판단한다. 반면 보다 정교하고 다양한 머신러닝 기술 및 로그(log) 분석 등을 통해 고장/장애 징후를 미리 파악할 수 있다면 proactive VNF live migration 을 통해 고장을 사전에 예방하고 경로 재설정에 따른 packet loss 와 같은 migration 비용을 줄일 수 있을 것이다. 둘째, 본 연구는 특정 SFC 하나를 최적화하기 위해 VNF live migration 을 수행하는 방법을 제안하고 있지만, 다양한 SFC 가 공존할 때 글로벌 최적화를 위한 migration 정책은 다를 수 있다. 이를 위해 SFC 의 end-to-end 지연 시간을 최소화하면서 여러 VNF 를 재배치하는 강화학습 모델을 적용하여 해결할 예정이다.

ACKNOWLEDGMENT

이 논문은 2021년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원(2018-0-00749, 인공지능 기반 가상 네트워크 관리기술 개발)과 2021년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임 (IITP-2021-2017-0-01633, 대학ICT연구센터육성지원사업)

참 고 문 헌

[1] Jo, Changyeon, Youngsu Cho, and Bernhard Egger. "A machine learning approach to live migration modeling." Proceedings of the 2017 Symposium on Cloud Computing. 2017.

[2] Google Cloud Compute Engine. “Setting instance availability policies”. https://cloud.google.com/compute/docs/instances/setting-instance-scheduling-options. Accessed March 25, 2021.

[3] Yi, Bo, et al. "Design and implementation of network-aware VNF migration mechanism." IEEE Access 8 (2020): 44346-44358.

[4] Rui, Lanlan, et al. "Petri Net-Based Reliability Assessment and Migration Optimization Strategy of SFC." IEEE Transactions on Network and Service Management (2020).

[5] Basu, Debabrota, et al. "Learn-as-you-go with Megh: Efficient live migration of virtual machines." IEEE Transactions on Parallel and Distributed Systems 30.8 (2019): 1786-1801.

[6] Ibrahimpasic, Amina Lejla, Bin Han, and Hans D. Schotten. "AI-Empowered VNF Migration as a Cost-Loss-Effective Solution for Network Resilience." arXiv preprint arXiv:2101.09343 (2021).

[7] Lin, Qingwei, et al. "Predicting node failure in cloud service systems." Proceedings of the 2018 26th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering. 2018.

[8] Huang, Huawei, and Song Guo. "Proactive failure recovery for NFV in distributed edge computing." IEEE Communications Magazine 57.5 (2019): 131-137.

[9] OpenStack Foundation. “Welcome to Zun’s documentation!”. https://docs.openstack.org/zun/latest/. Accessed March 25, 2021.

[10] El-Sayed, Nosayba, et al. "Temperature management in data centers: Why some (might) like it hot." Proceedings of the 12th ACM SIGMETRICS/PERFORMANCE joint international conference on Measurement and Modeling of Computer Systems. 2012.