전술 임무 단계 이종 AI 모델 세트의 오케스트레이션을 위한 쿠버네티스 커스텀 컨트롤러 설계

최하영, 박범식, 백승호
엘아이지디펜스앤에어로스페이스

hayoung.choi@ligdna.com, bumsik.park@ligdna.com, seungho.baek@ligdna.com

Design of a Kubernetes Custom Controller for Orchestrating Heterogeneous AI Model Sets in Tactical Mission Phase

Choi Ha Young, Park Bum Shik, Baek Seung Ho
LIG Defense&Aerospace

요 약

전술 엣지 환경에서 임무 단계에 따라 요구되는 AI 모델 조합이 달라짐에도, 기존 쿠버네티스 오케스트레이션은 이종 기능 모델의 세트 단위 구성과 이를 통한 복수 Pod의 일괄 전환, 그리고 전술 임무 단계별 자동 트리거를 지원하기 어렵다. 본 논문은 MSDO(Mission State-Driven Orchestrator)라는 쿠버네티스 커스텀 컨트롤러를 설계하여 전술 임무 단계별 서로 다른 복수의 AI 모델을 하나의 세트로 일괄 전환하는 기능을 제안한다. 임무 단계별 배포해야할 Deployment 목록을 묶어 Mission Set으로 정의하고, 단일 Reconcile 루프로 일괄 전환하며, Mission State Machine을 통해 임무 단계 변화를 수신하여 적합한 모델 세트를 자동 배포한다. 제안한 구조의 실험 결과 기존 방식 대비 AI 모델 전환/배포 완료 시간은 83.5% 단축되었고, 부분 전환 노출 시간은 96.0% 감소한 것을 확인하였다.

Ⅰ. 서 론

전술 엣지 환경에서 다수의 AI 모델이 연쇄적으로 협업하는 파이프라인의 구성은 임무 단계에 따라 달라진다. 예를 들어 감시·정찰 단계에서는 이상탐지·패턴분석 모델이 운용되나, 교전 단계로 국면이 전환되는 순간 표적추적·위협평가·타격 스케줄링 모델로 구성된 세트가 즉각적으로 배포되어야 한다.

그러나 기존 쿠버네티스 오케스트레이션은 이러한 요구를 충족하기에 3가지 어려움이 있다. 첫째, HPA/KEDA는 동일한 기능의 Pod를 복제하는 수평 확장만을 지원하기에 서로 다른 기능의 모델로 구성된 세트 단위의 구성이 어렵다[1]. 둘째, 다수의 Pod 교체가 필요한 경우 순차 처리를 하며, 일부 Pod만 교체된 상태인 부분 전환이 발생한다[2]. 셋째, CPU·메모리 등의 자원 또는 사전 정의된 트리거에만 의존하기에 전술 임무 단계의 변화에 따른 자동 전환이 어렵다[3].

본 연구는 이 세가지 어려움을 해소하는 쿠버네티스 커스텀 컨트롤러 MSDO(Mission State-Driven Orchestrator)를 제안한다.

Ⅱ. 관련 연구

KEDA(Kubernetes Event-Driven Autoscaling)는 외부 이벤트를 트리거로 동일 워크로드의 레플리카 증감을 조절하는 프로젝트이다[1]. 따라서 서로 다른 기능을 수행하는 모델을 묶어 배포하는 구성이 불가능하다.

Argo Rollouts는 Blue-Green 및 Canary 배포 전략을 통해 동일 기능 애플리케이션의 안전한 버전 교체가 가능하도록 지원한다[2]. 그러나 결국 단일 Deployment의 순차적 전환을 수행하기에 복수의 서로 다른 모델 Pod를 교체할 때에 부분 전환 상태가 발생한다. 전술 임무 단계는 복수의 모델이 전부 동작할 때에 온전한 하나의 기능이 완성되기에 부분 전환 상태 중에는 파이프라인의 불완전 동작 위험이 있다.

쿠버네티스 Priority Class는 Pod 퇴거 우선순위를 설계 시점에 정적으로 부여한다[4]. 최근 제안된 Workload-Aware Preemption[3] 역시 자원을 지표로 스케줄링을 개선하였다. 전술 엣지 클라우드에서 컨테이너 이미지 및 상태관리 기법[5] 연구는 워크로드의 마이그레이션 지연 시간을 감소시키는 방안을 제안하였으나, 임무 단계의 변화를 통한 모델 세트 자동·일괄 전환은 다뤄지지 않았다.

따라서 서로 다른 기능의 AI 모델 세트 단위의 정의와, 전술 임무 단계를 기반으로 한 모델 세트 단위 자동·일괄 전환을 다루는 연구는 현재 부재하다.

Ⅲ. 제안 아키텍처: MSDO

Ⅲ-1. MSDO 구조

MSDO는 그림 2와 같이 ① Mission State Machine, ② Mission Set, ③ MSDO Custom Controller의 세 핵심 구성요소로 이루어진다.

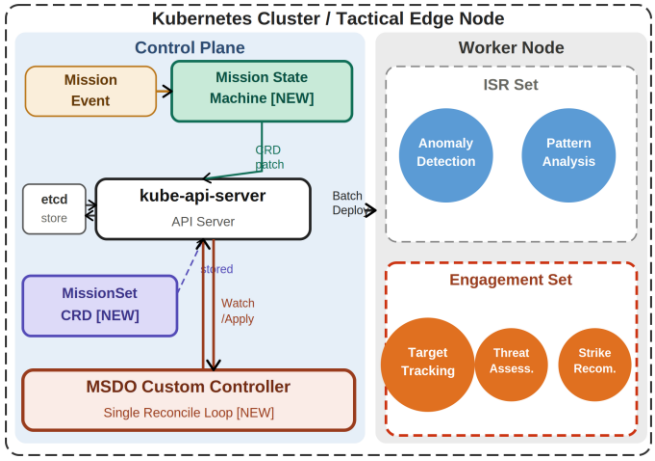


그림 2. MSDO 아키텍처

Ⅲ-2. Mission Set

Mission Set는 임무 단계별로 배포해야할 서로 다른 기능을 갖는 Deployment 목록을 하나의 리소스 단위로 선언적으로 정의하여 세트를 구성한다. 기존 KEDA의 ScaledObject가 단일 Deployment의 복제본 수를 다룬 것과 다르게 Mission Set를 통해 리소스를 커스텀하여 정의함으로써 복수의 서로 다른 기능을 갖는 AI 모델들을 하나의 세트로 묶어 관리한다.

Ⅲ-3. MSDO Custom Controller

MSDO Controller는 Mission Set 의 변경을 감지하여 단일 Reconcile 루프 내에서 신규 모델 세트의 배포와 기존 모델 세트의 제거를 동시에 처리한다. 신규 세트의 모든 Deployment가 Ready 상태에 도달한 이후 기존 세트를 일괄 제거하여 부분 전환 상태가 발생하지 않도록 보장한다. 어느 단계에서든 실패가 발생하면 이전 모델 세트로 자동 롤백하여 안정성을 유지한다.

Ⅲ-4. Mission State Machine

Mission State Machine은 외부로부터 전황 이벤트를 수신하여 현재 임무 단계를 STANDBY(대기) / ISR(감시) / ENGAGEMENT(교전) / WITHDRAWAL(철수) 중 하나로 유지한다. 이 임무 단계의 전환이 감지되면 해당 단계에 대응하는 Mission Set 를 자동으로 업데이트하여 MSDO Controller가 즉각 신규 모델 세트를 배포할 수 있도록 트리거한다. 기존 CPU·메모리 등 하드웨어 자원 지표가 아닌 해당 임무 단계 자체가 트리거가 된다는 점이 특징이다.

Ⅳ. 실험 및 검증

Ⅳ-1. 실험 환경

로컬 쿠버네티스 클러스터에서 실험을 수행하였다. AI 모델은 일정한 컨테이너 기동 시간을 모사하는 nginx:alpine 컨테이너 5종(이상탐지, 패턴분석, 표적추적, 위협평가, 타격 스케줄링)으로 구현하였으며, 임무 단계 전환은 Python 스크립트로 트리거하였다. 비교 대상은 KEDA와 Argo Rollouts를 조합한 순차 처리 방식이며 각 시나리오는 10회 반복 측정하였다.

Ⅳ-2. 실험 시나리오

ISR 단계에서 ENGAGEMENT 단계로의 전환 시, 기존 순차 처리 방식과 MSDO 일괄 전환 방식에 대해 두 가지 지표를 측정하였다. 첫째, 이벤트 발생 시점부터 신규 모델 세트 전체가 정상 운용 상태에 도달하기까지의 전환 완료 시간이다. 둘째, 전환 과정에서 일부 Pod만 배포된 불완전 상태가 지속되는 부분 전환 노출 시간이다.

Ⅳ-3. 실험 결과

기존 평균 6.08초가 소요된 반면, TSDO는 평균 0.99초로 83.8% 단축되었다. 이상치를 제외한 9회 평균은 0.88초이다.

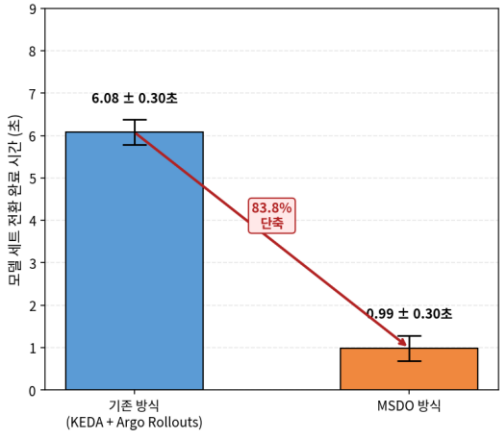


그림 3. 실험 결과 - 전환 완료 시간 비교

기존 방식은 부분 전환 상태 구간에 평균 3.00초 노출된 반면 MSDO는 노출 시간이 평균 0.12초로 96.0% 감소하였다. 10회 실험 전체에서 롤백은 발생하지 않았다.

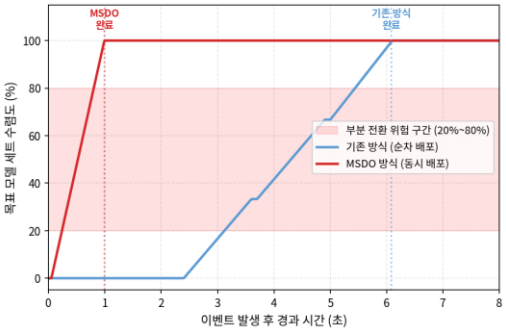


그림 4. 실험 결과 - 수렴도 추이 비교

이상의 결과는 MSDO가 서로 다른 복수의 모델의 세트 단위 배포와 부분 전환의 방지를 실험적으로 검증함을 보여준다. 또한 Mission State Machine 기반의 자동 전환은 아키텍처 수준에서 설계 및 제안하였다.

Ⅴ. 결 론

본 논문은 전술 엣지 환경에서 기존 쿠버네티스 오케스트레이션이 갖는 세 가지 어려움을 해소하는 커스텀 컨트롤러 MSDO를 제안하였다. Mission Set 를 통해 서로 다른 복수의 AI 모델을 세트 단위로 정의하고 배포하였고, MSDO Controller의 일괄 전환 메커니즘으로 부분 전환 상태 발생을 방지하였으며, Mission State Machine을 통해 전술 임무 단계 변화를 직접 오케스트레이션 트리거로 채택하여 기존 자원 중심의 트리거의 한계를 보완하였다.

향후 연구로는 실제 전술 메시지 포맷 연동을 통한 Mission State Machine 검증, 다중 엣지 노드 간 분산 전환 시나리오 확장, 임무 단계 자동 판별을 위한 경량 분류 모델 결합 등이 가능할 것으로 기대된다.

ACKNOWLEDGMENT

이 논문은 2026년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임(RS-2025-02316434, 유·무인 복합체계 대상 신속한 상황인지 및 맞춤형 대응을 위한 복합지능이 가능한 AI 분석 모델 자동 협업 기술 개발)

참 고 문 헌

[1] KEDA Authors, "KEDA: Kubernetes Event-Driven Autoscaling," CNCF Graduated Project, 2024, (https://keda.sh/docs/2.19/)

[2] Argo Project, "Argo Rollouts - Kubernetes Progressive Delivery Controller," 2024, (https://argoproj.github.io/argo-rollouts/)

[3] Kubernetes SIG-Scheduling, "Workload-Aware Preemption," KEP-5710, Kubernetes v1.36 [alpha], 2025, (https://kubernetes.io/docs/concepts/scheduling-eviction/workload-aware-preemption/)

[4] Kubernetes Authors, "Pod Priority and Preemption," Kubernetes Documentation, 2024, (https://kubernetes.io/docs/concepts/scheduling-eviction/pod-priority-preemption/)

[5] 임호근, 김영한, "전술 엣지 클라우드에서의 컨테이너 이미지 상태 관리 기법," 한국통신학회논문지, vol. 47, no. 7, pp. 1010-1016, Jul. 2022.