**연구계획서(프로포절)**

제목(가제): 클라우드–엣지–IoT 통합 구조에서의 지능형 자원 관리 시스템 연구: 데이터 수집부터 처리까지 전 과정을 자동화하는 프레임워크

소속/성명/이메일: (작성)

제출일: (작성)

**Summary**

본 연구는 클라우드–엣지–IoT(CEI) 통합 구조에서 데이터 수집→전처리→추론/분석→저장/옵스까지 전 과정을 자동화하는 지능형 자원 관리 프레임워크를 제안한다. SLO 인지형 오케스트레이션, 적응형 오프로딩·압축·캐싱, 강화학습 기반 스케줄러, 데이터/모델 품질 자동화를 통합하여 비용–지연–정확도 트레이드오프를 최적화한다. 테스트베드에서 KPI(지연, 처리량, SLO 달성률, 비용/전력, 데이터 품질, 복구시간)를 평가하고 가이드라인·오픈 아티팩트를 제공한다.

**Research Goals (No more than 500 words)**

* 모듈러 CEI 자동화 프레임워크 설계/구현(디바이스, 데이터 파이프라인, 오케스트레이션, 옵스).
* SLO 인지형 멀티티어 스케줄링과 적응형 오프로딩·압축·캐싱 정책 고안.
* 스트림+배치 혼합 DAG 실행기 구현 및 정책 엔진 연계.
* 텔레메트리 기반 지능형 자원 관리(강화학습/밴딧) 적용.
* 재현 가능한 테스트베드 및 CI/CD 파이프라인 공개.

**Research Description (No more than 1000 words)**

아키텍처: 디바이스–엣지–클라우드 3계층, 컨트롤/데이터 플레인 분리. 컨트롤은 정책/스케줄러/카탈로그, 데이터는 수집–처리–저장–서빙.

핵심 모듈: 디바이스&IoT 관리(Device Twin/OTA/보안), 데이터 파이프라인(수집/정화/품질), 오케스트레이션(SLO 스케줄링/오프로딩/압축/캐싱), 추론/분석(엣지 분할추론/연합학습), 옵스(MLOps/DataOps/모델 카탈로그/모니터링).

정책/스케줄링: 지연/가용성 SLO를 목적함수로 하고, 상태(RTT/무선품질/부하/자원/데이터 품질)에 기반해 RL+휴리스틱으로 의사결정. 밴딧으로 온라인 적응성 강화.

데이터/모델 라이프사이클: 수집→정화→특징화→학습→배포→모니터링→피드백 자동화. 품질 규칙과 성능 모니터링을 정책 엔진과 연계.

보안/프라이버시: 디바이스 인증, 경량 암호화, 온디바이스 마스킹/요약, 연합학습+차등프라이버시 옵션.

테스트베드: KubeEdge/EdgeX + Kubernetes, Wi‑Fi 6/6E, 5G SA, 장애/혼잡/이동성 주입. 스트림 처리(Flink/Spark), 저장(TSDB/객체저장소), 서빙(Ray Serve/TF Serving/ONNX Runtime).

평가: E2E 지연, SLO 위반률, 처리량, 비용/전력, 데이터 품질/드리프트, 복구시간, 오토스케일 수렴.

**Expected Research Outcomes (No more than 500 words)**

* CEI 자동화 프레임워크 및 레퍼런스 구현 공개.
* SLO 인지형 자원 관리 정책과 설계 가이드 제공.
* 데이터/모델 라이프사이클 자동화 템플릿과 품질 규칙 제공.
* 테스트베드/데이터셋/CI 공개로 재현성 확보.

**Key Words**

Cloud–Edge–IoT, Intelligent Resource Management, SLO-aware Scheduling, Offloading, MLOps, DataOps, Federated Learning, Stream Processing, KubeEdge, Kubernetes.

# 1) 연구 제목(가제)

클라우드–엣지–IoT 통합 구조에서의 지능형 자원 관리 시스템 연구: 데이터 수집부터 처리까지 전 과정을 자동화하는 프레임워크

# 2) 배경 및 필요성

CEI 환경은 다양한 이기종 자원과 급변하는 네트워크 상태로 인해 정적 정책으로는 일관된 SLO 보장이 어렵다.

현존 오케스트레이션은 컨테이너/함수 단위의 스케일은 우수하나, 계층간 오프로딩/압축/캐싱의 동적 결정을 내재화하지 못한다.

# 3) 연구 목표

* 전과정 자동화 프레임워크 설계 및 오픈소스 구현.
* SLO 인지형 멀티티어 스케줄러와 적응형 오프로딩·압축·캐싱.
* 혼합 DAG 실행/컴파일러 및 정책 엔진 연계.
* 재현 가능한 테스트베드/KPI로 정량 평가 및 가이드 제시.

# 4) 통합 아키텍처

* 컨트롤 플레인: 정책 엔진, SLO 스케줄러, 카탈로그, 텔레메트리.
* 데이터 플레인: 수집(MQTT/HTTP/Kafka)–처리(Flink/Spark)–저장(TSDB/객체저장소)–서빙(Ray/TF/ONNX).
* 보안/프라이버시: 인증·암호화·프라이버시 보존 처리(연합학습/차등프라이버시).

# 5) 핵심 모듈

* 지능형 스케줄러(RL+휴리스틱): 지연/비용/정확도 가중 최적화.
* 적응형 오프로딩/압축/캐싱: 네트워크/부하/품질 신호 기반.
* DAG 컴파일러: 워크로드를 계층별로 분할 배치, 데이터 이동 비용 포함.
* 품질/옵스: 데이터/모델 품질 규칙, 자동 재학습·롤백·카나리아.

# 6) 연구 질문(RQ) & 가설(H)

* RQ1/H1: SLO 위반률 30%+ 감소 가능 여부.
* RQ2/H2: 동일 SLO에서 비용 15–25% 절감 또는 지연 20%+ 개선.
* RQ3/H3: 품질 자동화로 드리프트/회귀 사고 40%+ 저감.
* RQ4/H4: 혼합 DAG로 처리량 1.3×, 평균 지연 25%↓.

# 7) 방법론

테스트베드/데이터/지표/분석 방법은 Summary에 기재된 구성과 동일하며, 다음을 보완한다:

* SLO 위반 비용 모델을 정의하고, 정책 엔진의 보상 함수로 사용.
* 실시간/마이크로배치 혼합 워크로드의 DAG 분할 기준(연산/데이터 의존성·지연 민감도)을 수식화.
* 엣지 선택/집합 최적화(지연/부하/연결성)를 근사 알고리즘으로 해결.

# 8) 성능 평가 계획

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **벤치마크/시나리오** | **워크로드** | **KPI** | **베이스라인** | **제안 기법** |
| 영상 분석(프라이버시) | RTSP 다중 스트림 + 객체탐지/마스킹 | E2E 지연, SLO 위반률, 백홀 절감 | K8s 기본 + 고정 오프로딩 | SLO 스케줄러 + 적응 오프로딩/압축/캐싱 |
| 예지 보전(시계열) | 센서 시계열 + 이상탐지 | 지연, 처리량, 전력 | KEDA/Knative 기본 오토스케일 | SLO 스케줄러 + DAG 분할/캐싱 |
| 모바일 AR | 분할추론 + 인코딩 파이프라인 | 95%tile 지연, QoE(VMAF), 배터리 | 고정 배치/프리셋 | 밴딧/RL 기반 프리셋/마이그레이션 |
| 버스티/장애 복구 | 부하 급증·노드 장애 주입 | 복구시간, 재시도, 손실율 | 기본 HPA/재시작 정책 | 정책 가드레일 + 선제 스케일/마이그레이션 |

* 베이스라인: K8s 기본 스케줄링 + 고정 오프로딩/스케일 + KEDA/Knative.
* 지표: E2E 지연/지터, SLO 위반률, 처리량, 비용/전력, 품질 지표, 복구시간, 수렴속도.
* 분석: 혼합효과/ANOVA/생존분석, 파레토 프런티어, 큐잉/코스트 모델, 민감도.

# 9) 기대 결과 및 기여

* CEI 자동화 프레임워크·코드·데이터 공개.
* 운영 가이드와 실무 적용 가능 레시피 제공.
* 학술적 성과: 스케줄링/오프로딩/데이터 품질 통합 모델.

# 10) 참고 문헌 (Related Work) 요약

1. [RW1] “The Case for VM-Based Cloudlets in Mobile Computing,” IEEE Pervasive Computing, 2009.

* 장점: 엣지 인프라의 '클라우드렛' 개념을 정립하고 근접 연산으로 지연 감소의 이점을 선도적으로 제시.
* 단점: 비전 중심으로 대규모 정량 실험과 다양한 워크로드 비교는 제한적.

1. [RW2] “Edge Computing: Vision and Challenges,” IEEE Internet of Things Journal, 2016.

* 장점: 엣지 컴퓨팅의 개념·아키텍처·과제를 종합적으로 정리한 대표 비전/서베이 논문.
* 단점: 정량적 KPI 비교나 구현 세부(오케스트레이션 지연·마이그레이션) 사례는 제한.

1. [RW3] “A Survey on Mobile Edge Computing: The Communication Perspective,” IEEE Communications Surveys & Tutorials, 2017.

* 장점: 통신 관점에서 오프로딩/자원관리 문제를 체계적으로 분류·정리한 고피인용 서베이.
* 단점: 아키텍처별(Cloudlet/Fog/Federation) 교차 실험 비교와 전구간 오케스트레이션 지표는 상대적으로 부족.

1. [RW4] “Fog and IoT: An Overview of Research Opportunities,” IEEE Network, 2016.

* 장점: Fog-IoT 융합의 기회·과제를 네트워킹 중심으로 체계화, 근원지 전처리·대역폭 절감의 가치 명시.
* 단점: 응용별 KPI(지연·지터·연속성)와 운영 자동화(스케줄링·정책)의 정량 평가가 제한.

1. [RW5] “Computation Offloading and Resource Allocation for Mobile Edge Computing,” IET Communications, 2019.

* 장점: 오프로딩과 자원할당의 결합 최적화를 수학적으로 정식화하여 통신-계산 자원 결합 전략 제시.
* 단점: 엔드-투-엔드 시스템 수준의 오케스트레이션 지연·이동성 시나리오 실증은 제한.

1. [RW6] “AccuMO: Accuracy-Centric Multitask Offloading in Edge-Assisted Mobile AR,” ACM MobiCom, 2023.

* 장점: 정확도 중심의 멀티태스크 오프로딩 프레임워크로 AR의 정확도·지연 트레이드오프 개선을 실험적으로 제시.
* 단점: 특정 AR 태스크/디바이스에 초점이라 범용 워크로드·다양한 엣지 인프라로 일반화하려면 추가 검증 필요.

# 11) 제안 기법 (성능, 방식, 알고리즘)

SLO 인지형 멀티티어 스케줄러 + 적응형 오프로딩/압축/캐싱 + DAG 컴파일러 + 정책 엔진의 결합.

* 상태·행동·보상 정의 및 오프폴리시 RL + 온라인 밴딧 혼합.
* 제약최적화(지연 SLA) 하 비용 최소화/지연 최소화의 이중 모드 지원.
* 안전장치: 휴리스틱 가드레일/롤백·카나리아·실험 게이트.

# 12) Target Conference/Journal 및 일정

|  |  |  |
| --- | --- | --- |
| **Venue** | **Type** | **마감(현행 공지)** |
| ACM SoCC 2025 | Conference | Event: 2025-11-19~21 (2026 CFP 추후 공지) |
| IEEE TMC / IEEE IoT-J | Journal | 상시 투고) |

* M1: 요구/지표/품질 규칙 확정, 테스트베드 구축
* M2–M3: 모듈 구현 및 예비실험
* M4: 본 실험(버스티/장애/이동성)
* M5: 분석·그림/표·가이드 정리
* M6: 논문화/제출·코드 공개