

# mApp: O-RAN 최적화를 위한 xApp 과 RAN 유닛 간 분산 프레임워크

이욱진, 송영진, 장용훈, 오승관, 이상현\*

고려대학교

{mekdugi, thd4090, disclose, osg0402, sanghyunlee}@korea.ac.kr

## mApp: A Distributed Framework for O-RAN Optimization Across Disaggregated RAN Units via xApps

Lee Wookjin, Song Youngjin, Jang Yong Hun, Oh Seungwan, Lee Sang Hyun\*  
Korea Univ.

### 요 약

본 연구에서는 O-RAN 제어를 위해 네트워크 레벨 전역 조율과 분산 제어의 실시간성을 동시에 달성하는 mApp 프레임워크를 제안한다. 제안 프레임워크는 message-passing 알고리즘을 기반으로 중앙의 xApp 전역 최적화 문제를 분해하고, 분산된 RAN 유닛 간 메시지 교환을 통해 분산 최적화를 수행한다. 또한 O-RAN traffic steering 유즈케이스 시뮬레이션을 통해, 제안 기법이 기존 중앙 집중형 기법 대비 우수한 성능을 달성함을 보인다.

### I. 서 론

차세대 무선 시스템은 다양한 네트워크 요구사항을 수용할 수 있는 유연한 아키텍처를 필요로 한다. Open RAN (O-RAN)은 이에 부응하여 RAN 기능의 맞춤형 구성을 가능하게 하는 개방형 프레임워크로 주목받고 있다. O-RAN 은 RAN 을 여러 계층으로 분리하고, RAN intelligent controller (RIC)이 이들을 중앙에서 제어하는 구조를 채택한다. 기존 O-RAN 에서 non-real-time (non-RT)/near-RT RIC 에서 실행되는 rApp 과 xApp 이 중앙 집중형 제어를 담당해왔다. 그러나 저지연 서비스 등 실시간 제어에 대한 요구가 증가하면서, 최근에는 기존 RAN 구조 내부에서 실행되는 dApp 의 활용이 모색되고 있다 [1]. 이 과정에서 중앙의 xApp 과 로컬의 dApp 간 제어 경계를 명확히 설정해야 하지만, 실제 RAN 제어 대상이 중첩되는 경우가 많아 역할 구분에 어려움이 있다. 특히, 중앙 집중형 제어는 전역 조율에는 효과적이지만 실시간성이 제한되고, 분산 제어는 빠른 응답이 가능하지만 네트워크 레벨의 일관된 조정이 어렵다는 상이한 특성이 있기에 두 방식 중 하나를 선택해야 하는 딜레마에 빠지게 된다.

본 연구에서는 이러한 문제를 해결하기 위해 중앙 집중형과 분산형의 장점을 결합한 중앙 통제형 분산 제어 프레임워크를 제안한다. 제안 프레임워크는 message-passing (MP) 알고리즘을 기반으로, RIC 의 xApp 이 전체 최적화 문제를 로컬 함수들로 분해하여 각 dApp 에 할당한다. 각 dApp 은 할당된 문제를 독립적으로 최적화하고, 인접 유닛과의 메시지 교환을 통해 중앙을 거치지 않고 전역적 합의에 도달한다. 이를 통해 네트워크 레벨의 전역 조율과 분산 실행의 실시간성을 동시에 확보할 수 있다. 제안하는 프레임워크는 선형 할당 최적화 문제로 표현 가능한 다양한 제어 시나리오에 범용적으로 적용할 수 있으며, 특히 O-RAN WG3 문서에서 제시하는 energy saving 유즈케이스를 통해 그 적용 가능성을 검증한다.

### II. 본 론

O-RAN 은 기존 RAN 을 open centralized unit (O-CU), open distributed unit (O-DU), open radio unit (O-

RU)로 분리하고, 이들을 서로 다른 벤더의 장비로 구성할 수 있도록 통신 인터페이스를 표준화하여 호환성을 확보하였다. 아울러 이들 유닛을 통합적으로 제어하는 RIC 를 별도로 두어 non-RT 와 near-RT RIC 로 구분하였으며, 각각 1s 이상의 상위 정책 생성과 10ms~1s 단위의 근실시간 RAN 제어를 담당한다. 각 RIC 에서는 rApp 과 xApp 이 실행되어 서비스 단위의 제어 로직을 수행한다. O-CU 와 O-DU 는 E2 인터페이스를 통해 RIC 에 연결되어 E2 노드로 지칭되며, 중앙형 RIC 에서 수행하기 어려운 실시간 제어를 위한 dApp 의 탑재 및 활용이 활발히 논의되고 있다.

O-RAN WG3 use cases and requirements 표준 문서의 여러 유즈케이스에서는 핵심 제어 대상으로 user equipment (UE)-셀 연결, 통신 자원 할당 등의 선형 할당 문제를 제시한다 [2]. 이러한 제어 작업을 중앙의 xApp 에서 처리하면 전역 최적화가 가능하지만, 네트워크 규모가 증가함에 따라 계산 부하가 급격히 증가하여 실시간 제어가 어려워질 수 있다. 반면, 문제를 로컬 단위로 분할하여 각 E2 노드의 dApp 에서 독립적으로 처리하면 빠른 응답이 가능하지만, 인접 셀 간 간섭이나 부하 불균형 등 전역적 상호작용을 고려하기 어렵다.

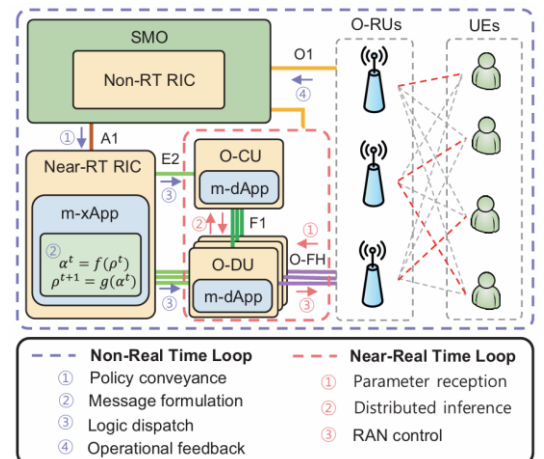


그림 1. mApp 프레임워크와 동작 구조

이러한 한계를 극복하기 위해 본 연구에서는 MP 알고리즘 기반의 중앙 통제형 분산 제어 프레임워크인 mApp 을 제안한다. MP 알고리즘은 전체 최적화 문제를 여러 로컬 함수로 분해하고, 각 함수를 담당하는 노드 간 메시지 교환을 통해 전역적 합의에 도달한다. 이 과정은 기본 산술 연산과 간단한 실수 값 메시지 교환만으로 구성되므로 기존의 O-RAN 프로토콜에 용이하게 통합될 수 있다. 또한 이전 해를 초기값으로 활용하면 소수의 반복만으로 수렴하는 특성이 있어 시간 제약 하에서 효율적인 분산 최적화가 가능하다.

제안하는 mApp 프레임워크의 동작 구조는 그림 1 과 같이 non-RT 루프와 near-RT 루프로 구성된다. Non-RT 루프에서 m-xApp 은 A1 인터페이스를 통해 non-RT RIC 으로부터 상위 네트워크 정책을 수신하고, 이를 기반으로 전역 문제를 구성한 뒤 로컬 함수로 분해하여 각 m-dApp 의 메시지 연산 로직을 정의한다. 정의된 로직은 E2 인터페이스를 통해 각 E2 노드에 전달된다. Near-RT 루프에서 각 E2 노드는 key performance measurement (KPM)를 수신한 후, 내장된 m-dApp 을 통해 해당 측정값을 기반으로 할당된 로컬 팩터 함수를 계산하고, 인접 노드와 메시지를 교환한다. 이 과정을 수차례 반복하여 메시지 값이 수렴하면 그 부호에 따라 제어 상태를 결정하고, open fronthaul (O-FH)을 통해 제어를 실행한다. 이러한 구조를 통해 중앙에서의 전역 조율과 분산 실행의 실시간성을 동시에 확보할 수 있다.

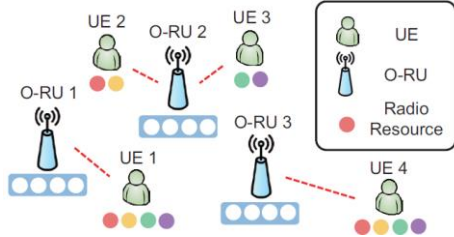


그림 2. O-RAN traffic steering 유즈케이스 시스템 모델

mApp 프레임워크를 O-RAN traffic steering 유즈케이스에 적용하기 위해, 그림 2 와 같이  $N$  개의 UE 집합  $\mathcal{N}$  과  $K$  개의 O-RU 집합  $\mathcal{K}$  로 구성된 네트워크를 고려한다. 각 O-RU  $a$  는 고정 대역폭  $B_a$  를 보유하며, O-RU  $a$  가 UE  $i$  만을 서비스할 때 기대 전송률  $r_{ia}$  는 다음과 같다.

$$r_{ia} = B_a \log_2 \left( 1 + \frac{P_a |h_{ia}|^2}{\sum_{b \in \mathcal{K}} P_b |h_{ib}|^2 + \sigma^2} \right) \quad (1)$$

여기서  $P_a$  는 O-RU  $a$  의 전송 전력,  $|h_{ia}|$  는 UE  $i$  와 O-RU  $a$  간 채널 이득,  $\sigma^2$  는 잡음 전력이다. O-RU  $a$  에  $u_a$  개의 UE 가 연결되면 대역폭이 균등 분배되어 UE  $i$  의 실제 전송률은  $r_{ia}/u_a$  가 된다. UE  $i$  와 O-RU  $a$  간 연결은 이진 변수  $x_{ia}$  로 표현되며, 총 전송률 최대화 문제는 다음과 같이 주어진다.

$$\max_{x_{ia}} \sum_{i \in \mathcal{N}} \sum_{a \in \mathcal{K}} \frac{r_{ia}}{u_a} x_{ia} \quad (2)$$

제약 조건으로 각 UE 는 하나의 O-RU 에만 연결되어야 하며 ( $\sum_{a \in \mathcal{K}} x_{ia} = 1$ ), 각 O-RU 는 최대  $M$  개의 UE 만 수용할 수 있다 ( $0 \leq u_a \leq M$ ).

주어진 최적화 문제를 해결하기 위해 m-xApp 은 각 UE 와 O-RU 에 대해 두 가지 메시지 업데이트 룰  $\rho_{ia}$  와  $\alpha_{ia}$  를 다음과 같이 정의한다.

$$\rho_{ia}^{(t)} = - \max_{b \in \mathcal{K} \setminus a} \alpha_{ib}^{(t-1)} \quad (3)$$

$$\alpha_{ia}^{(t)} = g_{ia}(1) - g_{ia}(0) \quad (4)$$

$$g_{ia}(x) = \max_{1 \leq u \leq M-1} \left( \sum_{l=1}^u \text{rnk}(l)_{j \in \mathcal{N} \setminus i} \left( \frac{r_{ja}}{x+u} + \rho_{ja} \right) \right) \quad (5)$$

이 때,  $\text{rnk}(l)_{j \in \mathcal{N} \setminus i}$  는 UE  $i$  를 제외한 UE 집합에서  $l$  번째로 큰 값을 의미한다. 메시지  $\rho_{ia}$  는 UE  $i$  가 O-RU  $a$  외의 다른 O-RU 에 연결될 때 얻을 수 있는 최대 이득을 나타내며, UE  $i$  가 연결된 O-RU 를 관찰하는 O-DU 의 m-dApp 에서 계산한다.  $g_{ia}(x)$  가 UE  $i$  의 연결 여부  $x$  에 따라 O-RU  $a$  가 얻을 수 있는 최대 전송률을 나타내기, 메시지  $\alpha_{ia}$  는 UE  $i$  가 O-RU  $a$  에 연결될 때의 로컬 이득을 나타내며, O-RU  $a$  를 관찰하는 O-DU 의 m-dApp 에서 계산한다. O-DU 간 수차례 교환 후 메시지 값이 수렴하면, m-dApp 은  $\rho_{ia} + \alpha_{ia} > 0$  인 경우 UE  $i$  를 O-RU  $a$  에 연결하도록 제어 명령을 전달한다.

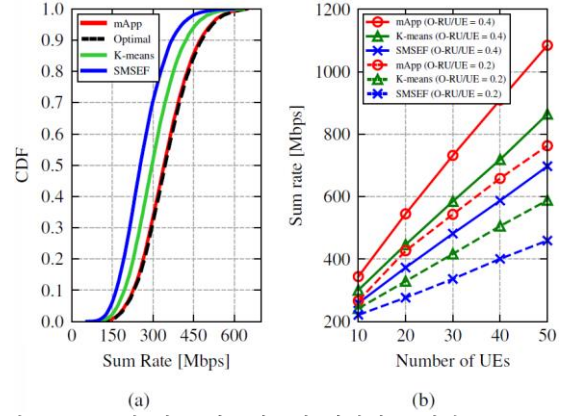


그림 3. mApp 과 비교 알고리즘의 에너지 소비량 (a)  $N=10$  일 때 누적분포함수 (b) UE 수에 따른 에너지 소비량

그림 3 은  $1\text{km} \times 1\text{km}$  영역에  $N=10$  UE 와  $K=4$  O-RU 를 배치하여 10,000 회 무작위 실험을 통해 누적분포함수를 평가한 결과이다. 제안 기법은 분산적 이점을 가지고 있는 동시에, 평균 전송률 344.1Mbps 로 최적해와 거의 일치하여 기존 중앙 집중형의 K-means (300.9Mbps) 대비 14%, SMSEF (259.0Mbps) 대비 33% 향상되었다. UE 수 증가에 따른 확장성 평가에서도 제안 기법은 모든 구간에서 비교 알고리즘을 상회하여 다양한 네트워크 규모에서 안정적인 성능을 보였다.

### III. 결론

본 연구에서는 O-RAN 환경에서의 네트워크 전역 조율과 실시간 분산 RAN 제어를 동시에 달성하는 mApp 프레임워크를 제안하였다. O-RAN Traffic steering 시뮬레이션 결과, 제안 기법은 분산 최적화를 수행하는 동시에 최적해에 근접한 전송률을 달성하여 기존 중앙 집중형 기법 대비 14% 이상의 성능 향상을 확인하였다.

### ACKNOWLEDGMENT

이 논문은 2025 년도 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임 (RS-2025-00563388).

### 참고 문헌

- [1] S. D'Oro *et al.*, "dApps for real-time RAN control: Use cases and requirements," nGRG Research Report, Oct. 2024.
- [2] O-RAN Working Group 3, "Use cases and requirements," O-RAN Alliance, Alfter, Germany, document O-RAN.WG3.TS.UCR-R004-v08.00, Feb. 2025.