Cloud–Native driven Stochastic Policy for Scalable Analytics Engine

Luis Blanco, Hatim Chergui, Swastika Roy, Engin Zeydan

Presentation at ITU-T FG-AN meeting in 13 October 2022,

This Project has received funding from the European Union´s Horizon 2020 research and innovation programme under Grant Agreement No. 871780

MonB5G

Project Overview
October 2022
ITU-T FG-AN Meeting

Dr. Engin Zeydan
MonB5G Project Coordinator
Services as Networks Research Unit CTTC

This project has received funding from the European Union's Horizon 2020 research and innovation programme under Grant Agreement No. 856691

About Centre Tecnològic de Telecomunicacions de Catalunya (CTTC)

No profit research center funded in 2001, after a public initiative

It receives financial support from the Generalitat de Catalunya and from research projects (both industrial and competitive funds)

It is approximately self-funded at 60%

Research is both applied and fundamental

It contributes about 83 journals and 127 int'l conferences every year.

CTTC in numbers

STAFF 133

PUBLICATIONS

- **825** Journals (ISI Indexed)
- **2.061** International conferences
- **122** Books & Chapters

BUDGET

8,1M€

PROJECTS CURRENTLY ACTIVE

91

FLAGSHIP EVENTS

187

PROJECT RANKINGS

In terms of EC funding

22nd	25th	2nd
in H2020 in Spain	in LEIT-ICT (4th in Spain)	in 5G-PPP (1st in Spain)

In terms of project participation

27th	26th	2nd
in H2020 in Spain	in LEIT-ICT (6th in Spain)	in 5G-PPP (1st in Spain)

R&D Centers

- **1** Spin-off
- **60** Awards
- **26** Granted patents in force

LABORATORIES

8 facilities
280 m²

Content

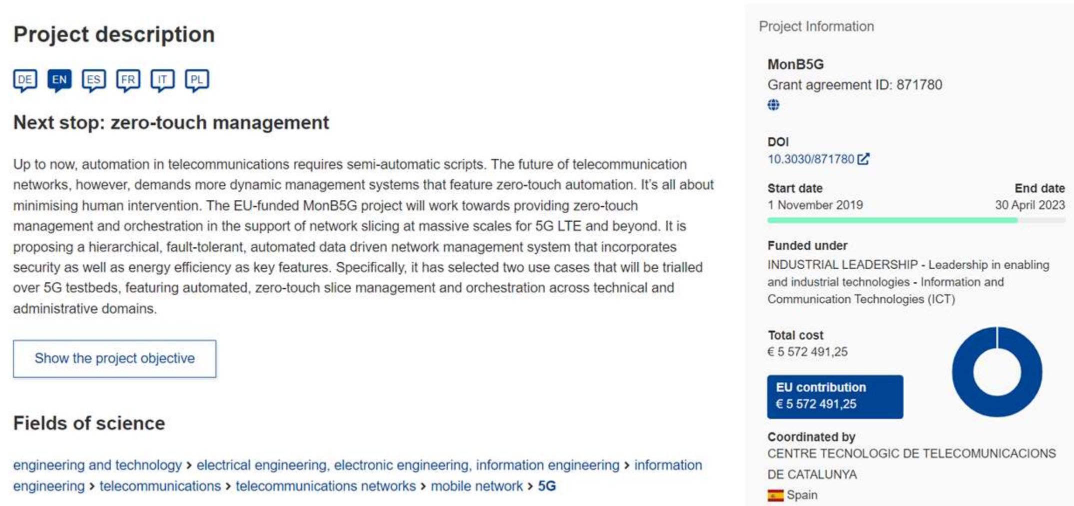
H2020 MonB5G
(Distributed management of Network Slices
in beyond 5G)

- MonB5G overview
  - Vision
  - Building Blocks
  - Timeline

https://www.monb5g.eu/

General Information

- **Grant Agreement:** 871780
- **Duration:** 42 months
- **Starting date:** 01/11/2019
- **Total budget:** 5,572,491.25 Euros
- **EC funding:** 5,572,491.25 Euros
- **Total PMS:** 662
- **Contact people:**
  - Dr. Engin Zeydan (Project Coordinator, CTTC), Selva Via (Project Manager , CTTC)
- **URL:** www.monb5g.eu



Project description

DE EN ES FR IT PL

Next stop: zero-touch management

Up to now, automation in telecommunications requires semi-automatic scripts. The future of telecommunication networks, however, demands more dynamic management systems that feature zero-touch automation. It's all about minimising human intervention. The EU-funded MonB5G project will work towards providing zero-touch management and orchestration in the support of network slicing at massive scales for 5G LTE and beyond. It is proposing a hierarchical, fault-tolerant, automated data driven network management system that incorporates security as well as energy efficiency as key features. Specifically, it has selected two use cases that will be trialled over 5G testbeds, featuring automated, zero-touch slice management and orchestration across technical and administrative domains.

Show the project objective

Fields of science

engineering and technology › electrical engineering, electronic engineering, information engineering › information engineering › telecommunications › telecommunications networks › mobile network › 5G

Project Information

MonB5G
Grant agreement ID: 871780

DOI
10.3030/871780

Start date: 1 November 2019
End date: 30 April 2023

Funded under
INDUSTRIAL LEADERSHIP - Leadership in enabling and industrial technologies - Information and Communication Technologies (ICT)

Total cost
€ 5 572 491,25

EU contribution
€ 5 572 491,25

Coordinated by
CENTRE TECNOLOGIC DE TELECOMUNICACIONS DE CATALUNYA
Spain

https://cordis.europa.eu/project/id/871780

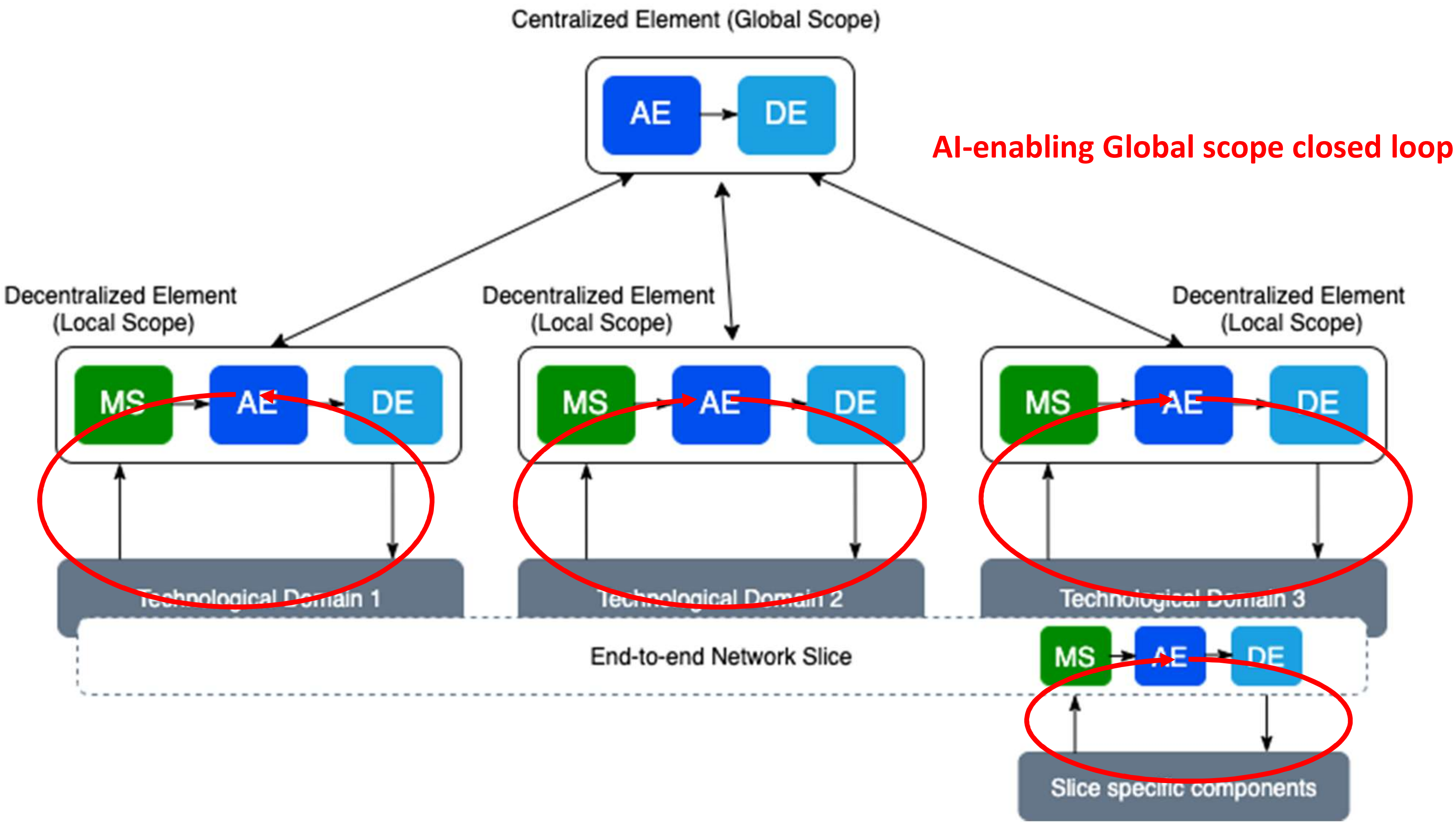
Consortium

MonB5G proposes a novel autonomic management and orchestration framework, heavily leveraging distribution of operations together with state-of-the-art data-driven AI-based mechanisms.

MonB5G Overview

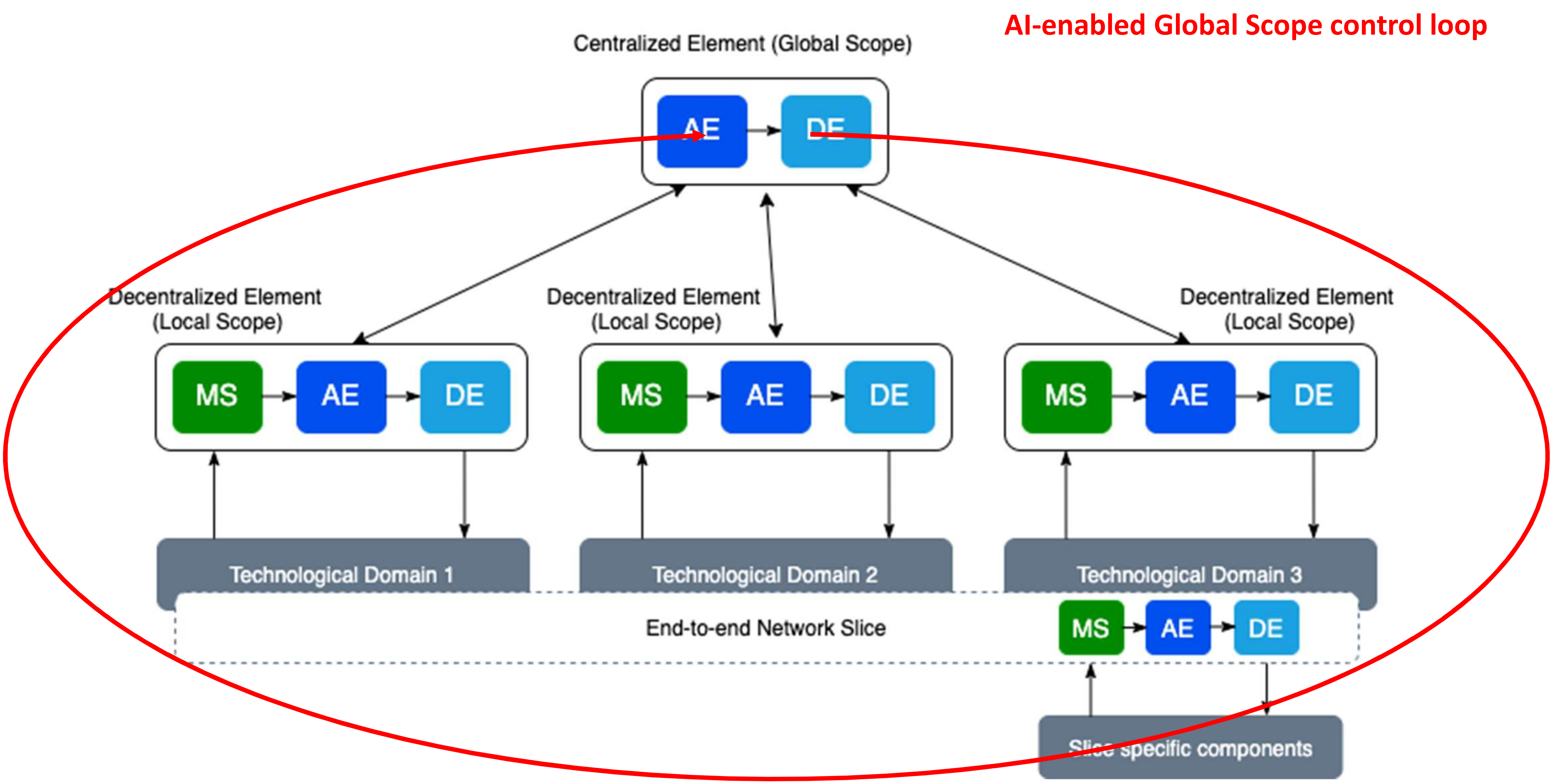
- **Vision:** Hierarchical, distributed, scalable, and AI-based management of a massive number of network slices across domains, towards zero-touch management.
- **Technical approach:**
  - Distribute the management functions over all entities in charge of the Life Cycle Management (LCM) of network slices
    - Delegate service-level management functions to be on-boarded within the network slice
  - Distributed closed control loops that assist the LCM entities with state-of- the-art AI-based and data-driven mechanisms
    - MS: Monitoring Systems; AE: Analytical Engine; DE: Decision Engine
- **Two use-cases will be demonstrated**
  - Zero-Touch Network and service management with end-to-end SLAs
  - AI-assisted policy-driven security monitoring & enforcement

Project High-level vision



H. Chergui, A. Ksentini, L. Blanco and C. Verikoukis, "**Toward Zero-Touch Management and Orchestration of Massive Deployment of Network Slices in 6G**," in IEEE Wireless Communications, vol. 29, no. 1, pp. 86-93, Feb. 2022

Project High-level vision



MonB5G: Main functional blocks

Distributed Management & Orchestration Architecture

Defining a scalable & Zero-touch service management architecture for massive network slice deployment
(1) Use-case refinement
(2) AE/DE requirement
(3) KPI

Closed control loop (Data driven algorithms)

Monitoring Systems (MS)

(1) Distributed monitoring
(2) Graph-based data representation

Analysis Engine (AE)

(1) Distributed ML (Federated Learning) Distributed Inference
(2) AI-driven network fault management
(3) Slice-level KPI prediction

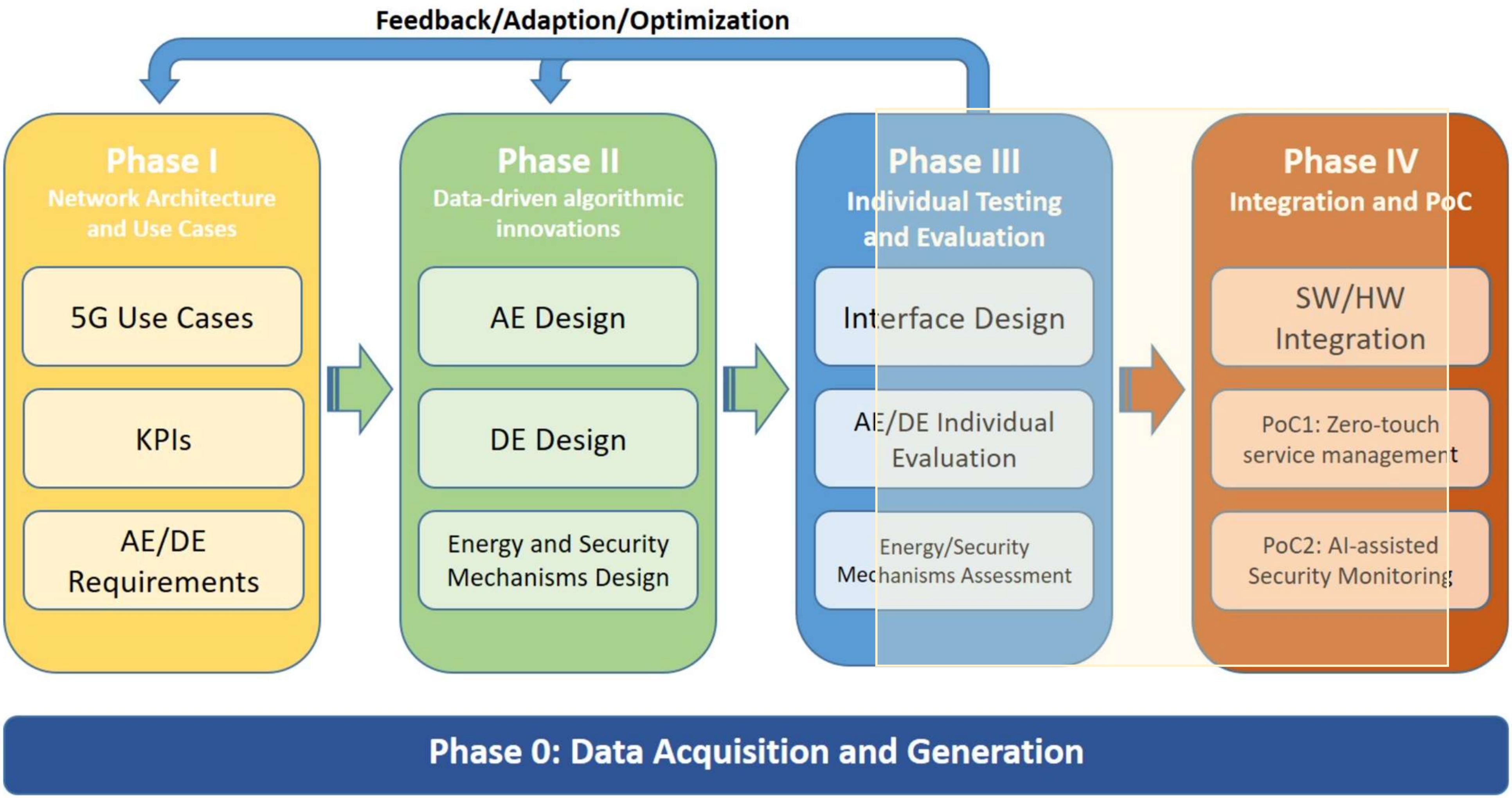
Decision Engine (DE)

(1) Distributed Reinforcement Learning for slice orchestration
(2) Data-driven inter-slice management

Security and Energy Management mechanisms

(1) Zero touch security management architecture
(2) Trust management
(3) AI-based Security management mechanisms
(4) AI-based Energy management mechanisms

Timeline


Feedback/Adaption/Optimization
Phase I
Network Architecture and Use Cases
5G Use Cases
KPIs
AE/DE Requirements
Phase II
Data-driven algorithmic innovations
AE Design
DE Design
Energy and Security Mechanisms Design
Phase III
Individual Testing and Evaluation
Interface Design
AE/DE Individual Evaluation
Energy/Security Mechanisms Assessment
Phase IV
Integration and PoC
SW/HW Integration
PoC1: Zero-touch service management
PoC2: AI-assisted Security Monitoring
Phase 0: Data Acquisition and Generation

ITU-T SG13 Contributions

Orange Poland, in the name of the consortium has made contributions and proposals to ITU-T

ITU-T: Contribution about MonB5G scalable architecture

ITU-T Study Group 13 Future Networks and emerging technologies	
Questions: Q20,21/SG13	
Question 20/13 Networks beyond IMT-2020 and machine learning: Requirements and architecture	Question 21/13 Networks beyond IMT-2020: Network softwarization Including software-defined networking, network slicing and orchestration

ITU-T journal paper recently accepted!

ITUJournal
Future and evolving technologies

FREE I FAST I FOR ALL

Special issue

Integrated and autonomous network management and control for 6G time-critical applications

AI-DRIVEN PREDICTIVE AND SCALABLE MANAGEMENT AND ORCHESTRATION OF NETWORK SLICES

Sławomir Kukliński[1,2], Lechosław Tomaszewski[1], Robert Kołakowski[1,2], Anne-Marie Bosneag[3], Ashima Chawla[3], Adlen Ksentini[4], Sabra Ben Saad[4], Xu Zhao[5], Luis A. Garrido[6], Anestis Dalgkitsis[6], Bahador Bakhshi[7], Engin Zeydan[7]

[1]Orange Polska, Orange Innovation Poland, ul. Obrzeżna 7, 02-691 Warszawa, Poland, [2]Warsaw University of Technology, Faculty of Electronics and Information Technology, ul. Nowowiejska 15/19, 00-665 Warszawa, Poland, [3]Ericsson Ireland, Network Management Lab, Athlone, Co. Westmeath, N37PV44, Ireland, [4]Eurecom, Campus SophiaTech, 450 Route des Chappes, 06410 Biot, France, [5]NEC Laboratories Europe, Kurfürsten-Anlage 36, 69115 Heidelberg, Germany, [6]Iquadrat Informatica, S.L, Carrer Doctor Rizal, 10, 08006, Barcelona, Spain, [7]Centre Tecnològic de Telecomunicacions de Catalunya, Carrer Doctor Rizal, 10, 08006, Barcelona, Spain

NOTE: Corresponding author: Sławomir Kukliński, slawomir.kuklinski@orange.com

Abstract – *The future network slicing enabled mobile ecosystem is expected to support a wide set of heterogenous vertical services over a common infrastructure. The service robustness and their intrinsic requirements together with the heterogeneity of mobile infrastructure and resources in both technological and spatial domain significantly increase the complexity and create new challenges regarding network management and orchestration. High degree of automation, flexibility and programmability are becoming the fundamental architectural features to enable seamless support for the modern telco-based services. In this paper, we present a novel management and orchestration platform for network slices, which has been devised by the Horizon 2020 MonB5G project. The proposed framework is a highly scalable solution for network slicing management and orchestration that implements a distributed and programmable AI-driven management architecture. The cognitive capabilities are provided at different levels of management hierarchy by adopting necessary data abstractions. Moreover, the framework leverages intent-based operations to improve its modularity and genericity. The mentioned features enhance the management automation, making the architecture a significant step towards self-managed network slices.*

Keywords – 5G, 6G, AI, management, ML, network slicing, orchestration, ZSM

Cloud–native Driven Stochastic Policy for Zero-Touch Scalable Analytics Engine

This Project has received funding from the European Union´s Horizon 2020 research and innovation programme under Grant Agreement No. 871780

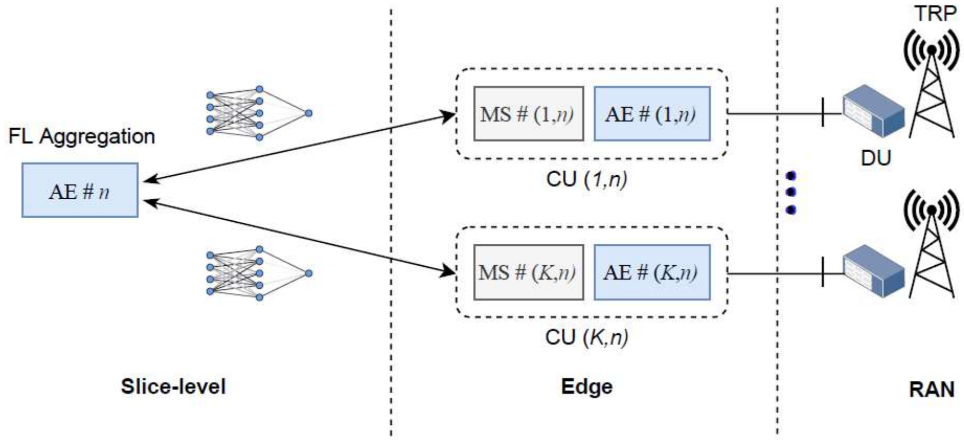
Contribution

- To deal with the **FL resource provisioning** task at the local analytic engines (AEs), we formulate the corresponding SLA-constrained optimization problem under the proxy-Lagrangian framework and solve it via a non-zero sum two-player game strategy.

- To **ensure scalability under massive slicing**, a novel **SLA-driven stochastic FL policy** is designed. A subset of active AEs is selected in each FL round, based on their violation rate (convergence time & communications overhead improvement, energy efficiency).

- Deploy the proposed solution in a containerized in a **cloud-native environment**.

Network model and dataset



Feature	Description
OTT Traffics	Apple, Facebook, Facebook Messages, Facebook Video, Instagram, NetFlix, HTTPS, QUIC, Whatsapp, and Youtube
CQI	Channel quality indicator
MIMO Full-Rank	MIMO full-rank usage (%)
# Users	Downlink Average active users

Output	Description
CPU Load	CPU resource consumption (%)

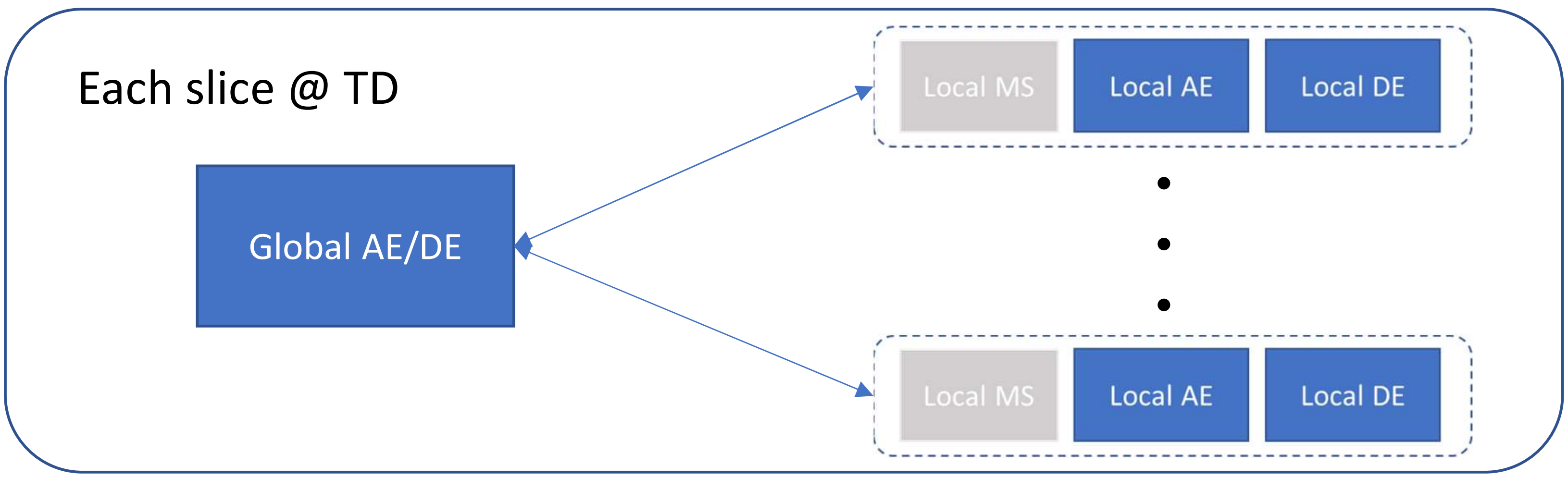
- **6G RAN-Edge topology** under per-slice CU/DU functional split. Each TRP co-located with its DU.
- Each CU k (k=1,…,n) has a MS and an AI-enabled AE.
- Each CU performs data collection to build a local dataset $D_k = \{x_k^{(i)}, y_k^{(i)}\}_{i=1}^{D_k}$ of size D_k.
- An OSS server (at the cloud) plays the role of the FL model aggregator.
- SLA is established between slice *n* tenant and infrastructure provider so that the CPU resources not exceed [α_n, β_n] with a prob. higher than a threshold γ_n.

Resource Prediction under SLA

- Predict slice-level resource usage under SLA constraints,
- For each slice: Multiple decentralized AEs as per the architecture,
- Challenges:
  - Extend federated learning framework to include SLA constraints
  - Ensure local SLA per slice while using small decentralized local datasets

Global Federated Averaging algorithm (Server)
- Average local models having the same constraints (same slice)
- Send updated global model to clients

Local Data-constrained Models (Clients)
- Local datasets
- Local constraints (e.g., congestion rate)



Mini-Datasets and Settings

- **3 slices:**
  - **eMBB:** NetFlix, Youtube and Facebook Video,
  - **Social Media:** Facebook, Facebook Messages, Whatsapp and Instagram,
  - **Browsing:** Apple, HTTP and QUIC.
- **For each slice: 200 MS/AE instances (clients)**
- **Local mini-datasets of size = 1000 samples NIID**

	Feature	Description
Features	**OTT Traffics per TRP**	Includes the hourly traffic for the top OTTs: Apple, Facebook, Facebook Messages, Facebook Video, Instagram, NetFlix, HTTPS, QUIC, Whatsapp, and Youtube
	CQI	Channel quality indicator reflecting the average quality of the radio link of the TRP
	MIMO Full-Rank	Usage of MIMO full-rank spatial multiplexing in %
Output	**DLPRB**	Number of occupied downlink physical resource blocks
	CPU Load	CPU resource consumption in %
	RRC Connected Users	Number of RRC users licenses consumed per eNB

Statistical Federated Learning

- SLA : any assigned resource to the tenant should not exceed a range $[\alpha_n, \beta_n]$ with a probability higher than an agreed threshold γ_n.
- This translates into learning the CPU resource allocation model under **empirical cumulative density function (CDF)** constraints
- Amounts to solving the following local optimization task at FL round t

$$\min_{\mathbf{W}_{k,n}^{(t)}} \frac{1}{D_{k,n}} \sum_{i=1}^{D_{k,n}} \ell\left(y_{k,n}^{(i)}, \hat{y}_{k,n}^{(i)}\left(\mathbf{W}_{k,n}^{(t)}, \mathbf{x}_{k,n}\right)\right),$$

$$\text{s.t. } F_{\mathbf{x}_{k,n} \sim \mathcal{D}_{k,n}}(\alpha_n) = \frac{1}{D_{k,n}} \sum_{i=1}^{D_{k,n}} \mathbb{1}\left(\hat{y}_{k,n}^{(i)} < \alpha_n\right) \leq \gamma_n,$$

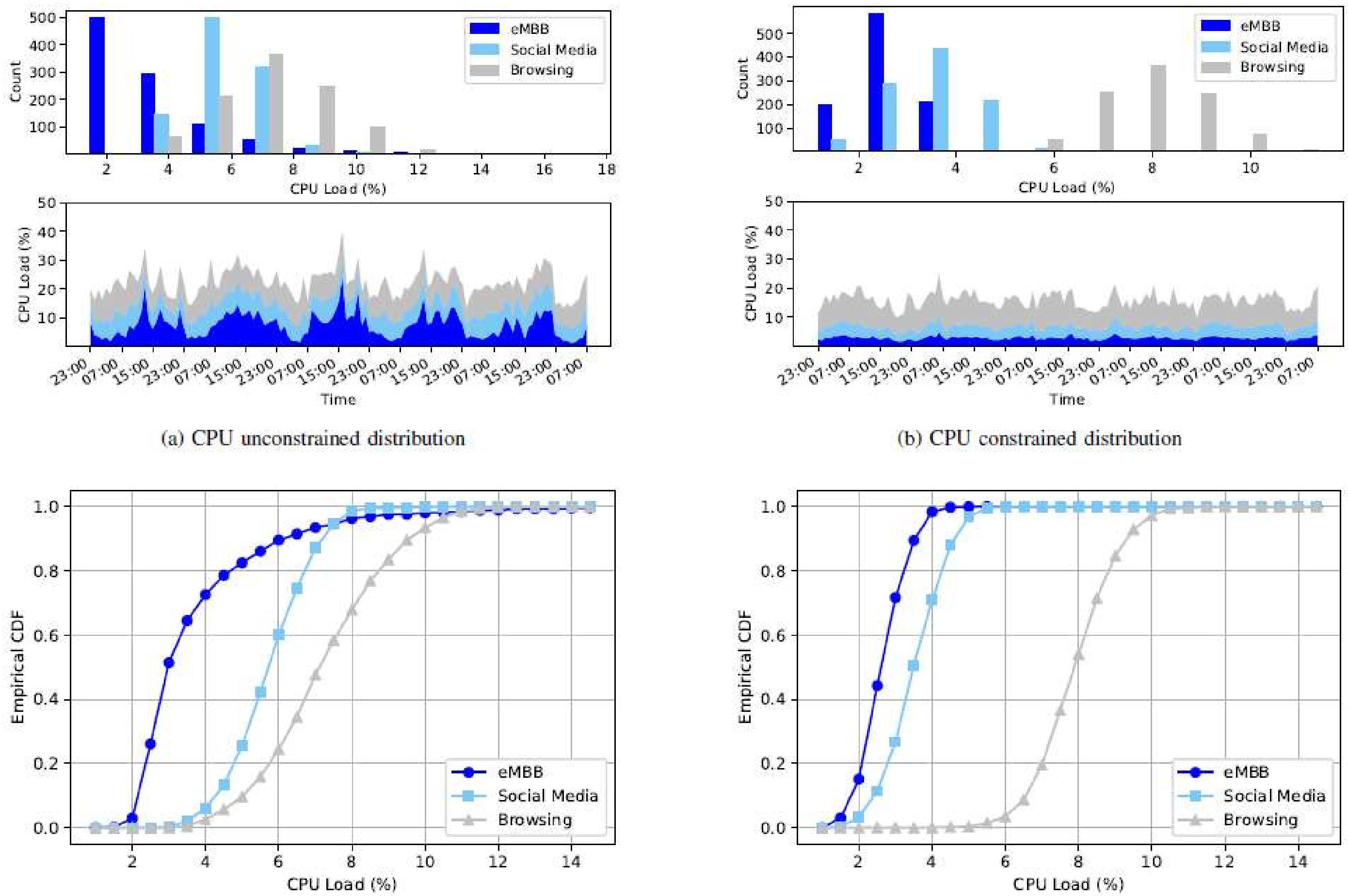
Empirical CDF

$$\tilde{F}_{\mathbf{x}_{k,n} \sim \mathcal{D}_{k,n}}(\beta_n) = \frac{1}{D_{k,n}} \sum_{i=1}^{D_{k,n}} \mathbb{1}\left(\hat{y}_{k,n}^{(i)} > \beta_n\right) \leq \gamma_n,$$

H. Chergui, L. Blanco and C. Verikoukis, "**Statistical Federated Learning for Beyond 5G SLA-Constrained RAN Slicing**," in IEEE Transactions on Wireless Communications, March 2022.

Statistical FL. Results (1/2)

- CPU load distributions



(a) CPU unconstrained distribution

(b) CPU constrained distribution

(c) CPU unconstrained CDF

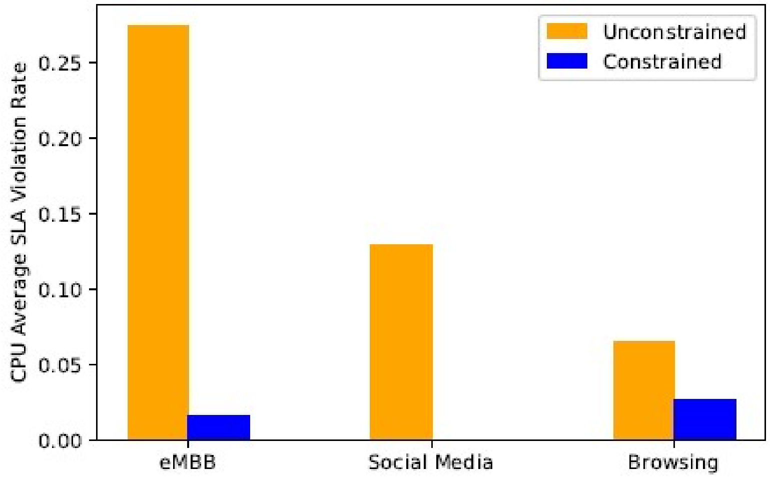
(d) CPU constrained CDF

Figure 4: CPU load distributions, with $\alpha = [0, 0, 0]$, $\beta = [4, 7, 10]$ % and $\gamma = [0.01, 0.01, 0.01]$.

H. Chergui, L. Blanco and C. Verikoukis, "**Statistical Federated Learning for Beyond 5G SLA-Constrained RAN Slicing**," in IEEE Trans. on Wireless Comm., March 2022.

Statistical FL. Results (1/2)

- CPU average SLA violation rate



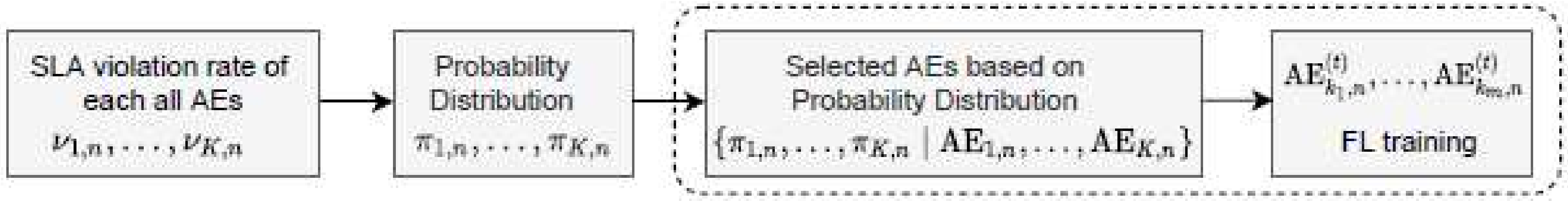
- Dramatic overhead reduction at convergence

Table I: Overhead and energy comparison

Rounds	50	60	70	80
Overhead CCL (KB)		18750		
Overhead StFL (KB)	1055	1266	1477	1688
Energy CCL (mJ)		118.3		
Energy StFL (mJ)	6.7	8	9.3	10.7
Energy Gain	**×17.8**	**×14.8**	**×12.7**	**×11.1**

H. Chergui, L. Blanco , L. A. Garrido, K. Ramantas, S. Kuklinski, A. Kasentini, S. Kuklinksli, "**Zero-Touch AI-Driven Distributed Management for Energy-Efficient 6G Massive Network Slicing**," in IEEE Dec. 2021.

SLA-driven stochastic AE Selection Policy



- **GOAL**: To **ensure scalability under massive slicing**, a **novel SLA-driven stochastic FL policy** is designed.

- Based on the SLA violation rate, a subset of the *m* out of *K* AEs participate in the training (each FL round)

- SLA violation evaluation in test mode:

$$\nu_{k,n} = \frac{1}{\tilde{D}_n} \sum_{i=1}^{\tilde{D}_n} \mathbb{1}\left[\left(\hat{y}^{(i)}_{k,n} < \alpha_n\right) \bigcup \left(\hat{y}^{(i)}_{k,n} > \beta_n\right)\right]$$

- AEs with low SLA violation have higher probability to participate in the FL round (**softmin-based policy**)

- The trained model is broadcast to all AEs

SLA-driven stochastic AE Selection Policy

- SLA violation evaluation in test mode:

$$\nu_{k,n} = \frac{1}{\tilde{D}_n} \sum_{i=1}^{\tilde{D}_n} \mathbb{1}\left[\left(\hat{y}^{(i)}_{k,n} < \alpha_n\right) \bigcup \left(\hat{y}^{(i)}_{k,n} > \beta_n\right)\right]$$

- A subset of the *m* out of *K* AEs participate in each FL round.

- AEs with a low SLA violation have a higher probability to participate in the FL round (**softmin function)**

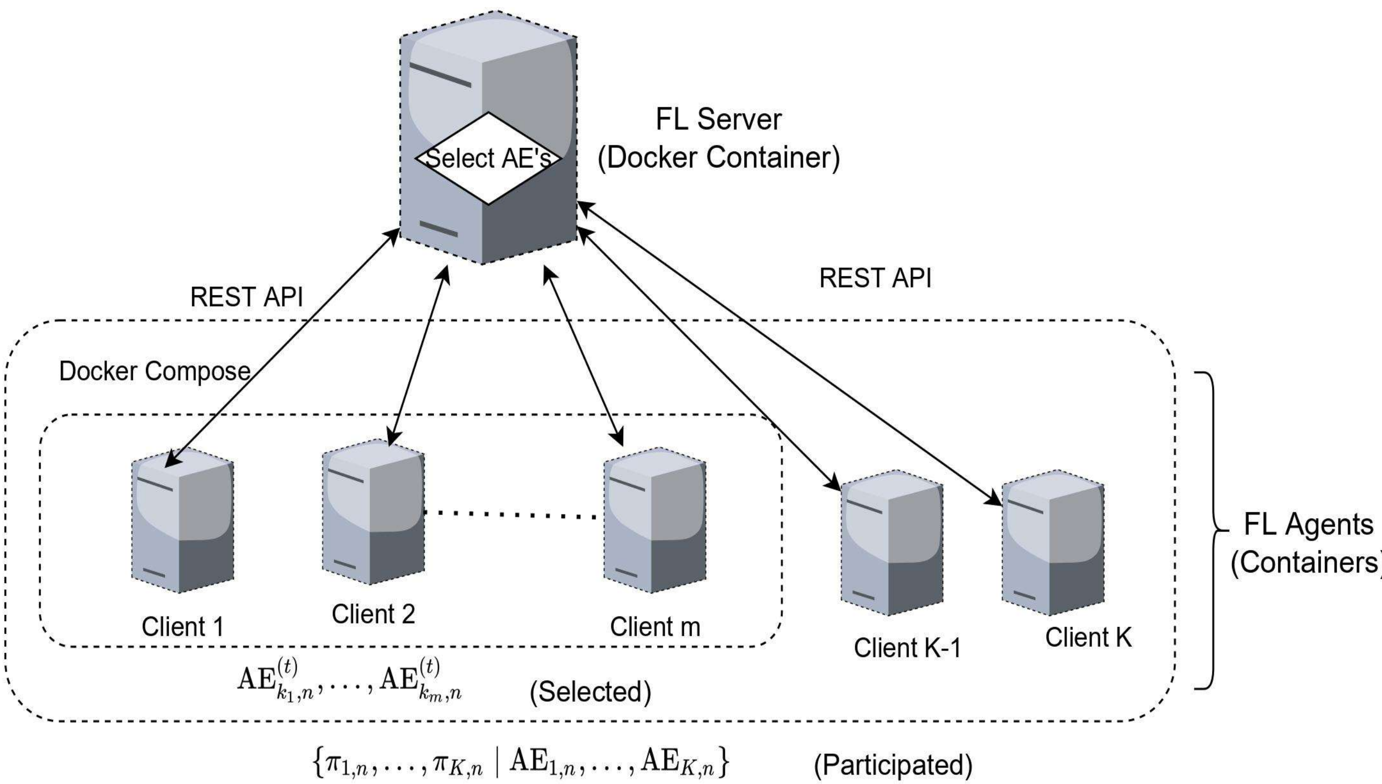
Algorithm 1: SLA-Driven Stochastic Federated Learning Policy.

Input: K, m, η_λ, T, L. `# See Table II`
parallel for $k = 1, \ldots, K$ **do**
`# Calculate SLA based violation rate`
AE (k, n) calculates $\nu_{k,n}$ according to 4 and reports it to the aggregation server
end parallel for
`# Federated Learning`
`# Server generates probability distribution using Softmin function`
for $k = 1, \ldots, K$ **do**
$\boxed{\pi_{k,n} = \frac{\exp\{-\nu_{k,n}\}}{\sum_{l=1}^{K} \exp\{-\nu_{l,n}\}}, \; k = 1, \ldots, K}$
end
Server initializes $\mathbf{W}^{(0)}_n$ with initial training parameter
for $t = 0, \ldots, T-1$ **do**
`# Server selects the` m `AEs ID using np.random.choice`
$\mathrm{AE}^{(t)}_{k_1,n}, \ldots, \mathrm{AE}^{(t)}_{k_m,n} \sim \{\pi_{1,n}, \ldots, \pi_{K,n} \mid \mathrm{AE}_{1,n}, \ldots, \mathrm{AE}_{K,n}\}$
Server broadcasts $\mathbf{W}^{(0)}$ to the m selected AEs
parallel for $k \in \{k_1, \ldots, k_m\}$ **do**
`# Local epochs`
for $l = 0, \ldots, L-1$ **do**
Solve the proxy-Lagrangian game between $\mathcal{L}_{\mathbf{W}^{(t)}_{k,n}}$ and $\mathcal{L}_\lambda$ and get $\mathbf{W}_{k,l}$
end
return $\mathbf{W}^{(t)}_{k,n} = \mathbf{W}_{k,L-1}$
Each local AE k sends $\mathbf{W}^{(t)}_{k,n}$ to the aggregation server.
end parallel for
`# FL Server Aggregation`
return $\mathbf{W}^{(t+1)}_n = \sum_{k \in \{k_1, \ldots, k_m\}} \frac{D_{k,n}}{D_n} \mathbf{W}^{(t)}_{k,n}$
Broadcasts $\mathbf{W}^{(t+1)}_n$ to all K AEs.
end

S. Roy, H. Chergui, L. Sanabria-Russo and C. Verikoukis, "**A Cloud Native SLA-Driven Stochastic Federated Learning Policy for 6G Zero-Touch Network Slicing**," IEEE ICC 2022.

Docker Implementation. Architecture

- AEs simultaneously run by using **Docker compose** tool.
- Through **REST API**, the FL Server and AEs (clients) can communicate with each other.
- **FastAPI** as a REST API is used in our implementation because it is a modern, open-source, fast, and highly performant Python web framework used for building Web APIs.



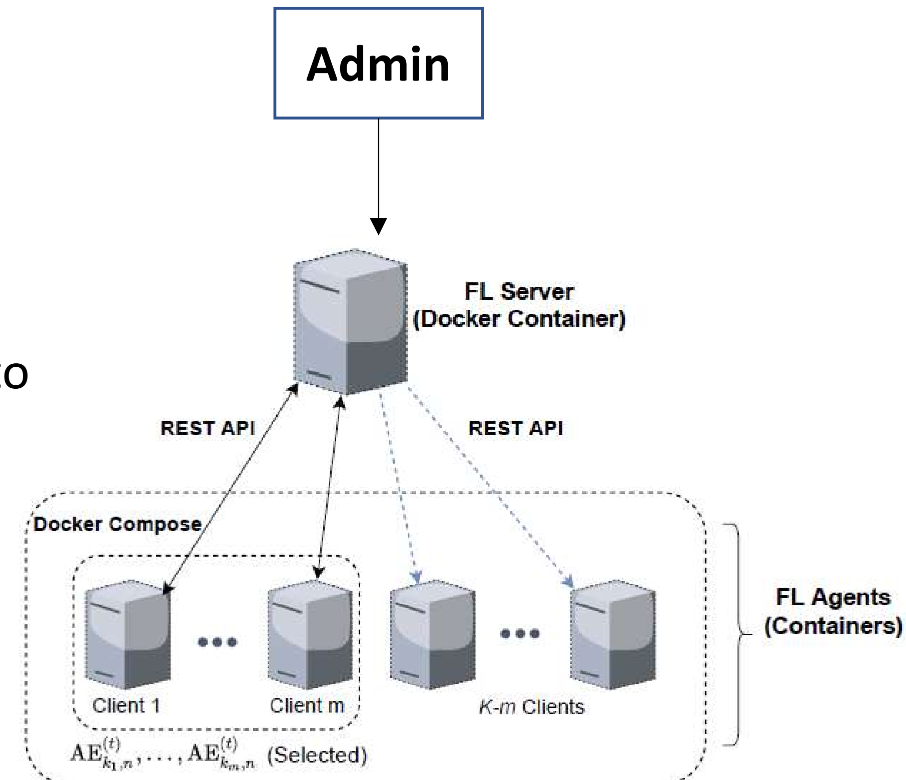
Docker Implementation: Communication process

Server (4 APIs):

- **POST/client:** Registering clients with the Server. **(from Client to Server)**
- **GET/select client:** Initiate policy for selecting clients and corresponding FL training. **(from Admin to Server)**
- **POST/SLA:** Clients send their SLA violation rate to the Server node. **(from Client to Server)**
- **PUT/model-weights:** Clients send calculated model parameters to the Server node. **(from Client to Server)**

Client (3 APIs):

- **PUT/SLA:** Server requests each of the clients to calculate their SLA violation rate. **(from Server to Client)**
- **POST/training:** Server requests the selected clients to start FL training with new model weights. **(from Server to Client)**
- **PUT/worker_model:** Update client initial model parameters. **(from Server to Client)**



Docker Implementation. Workflow

STEP 1: REGISTRATION: All clients register with their IP address in the server node using **POST/client** request.

STEP 2: START: After registration, server sends a request to all the registered clients to start the client selection process through **POST/select-client** request.

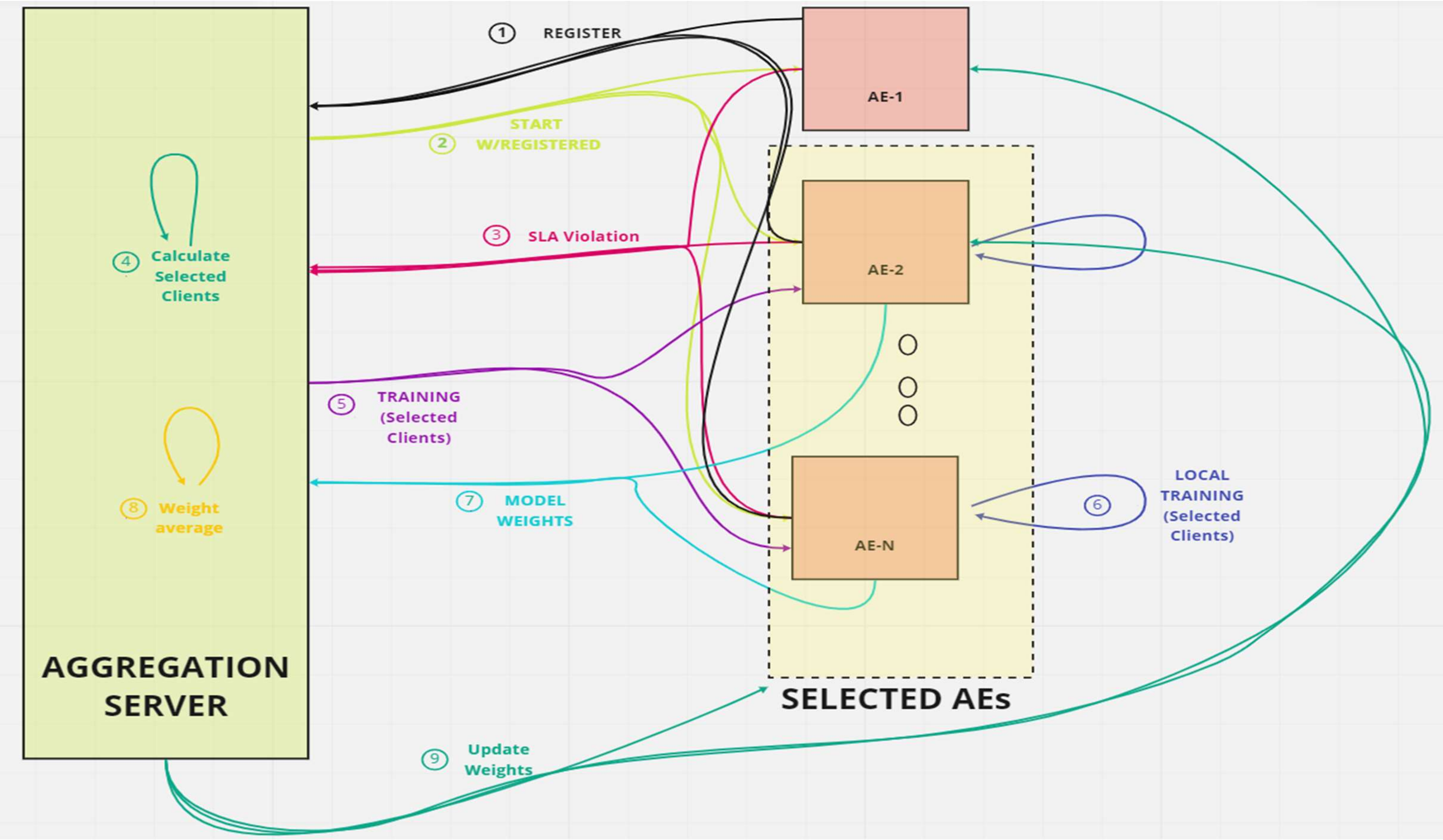
STEP 3: COMPUTE SLAs: All clients compute and send their SLA violation rate to the server through **PUT/SLA** & **POST/SLA**.

STEP 4: COMPUTE PROB. DISTRIBUTION: Server generates the probability distribution of the clients using the **softmin** function and selected clients using np.random.choice.

STEPS 5-6: TRAINING: Server sends **POST/training** requests to the selected clients and start FL training.

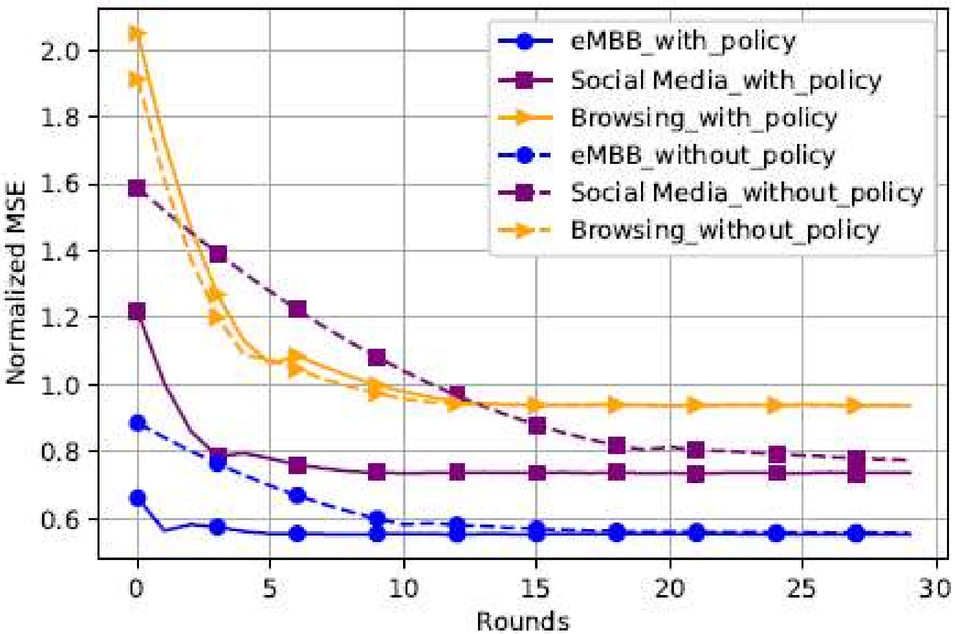
STEPS 7-8-9: COMPUTE & UPDATE WEIGHTS: Model weights of each selected client are sent to the server through **PUT/model-weights**, and then Server averages the weighs and update the weights of the clients using **PUT/worker-model** & repeat same procedure next FL rounds (GO TO STEP 3).

Docker Implementation Workflow

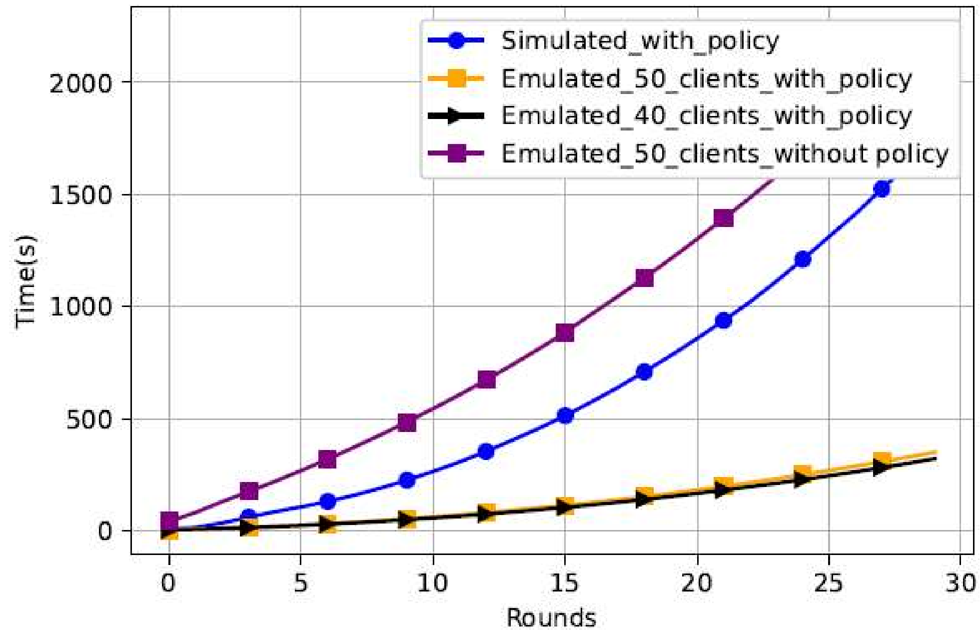


Scalability

- Select 25 AEs out of (40, 50)



Faster convergence

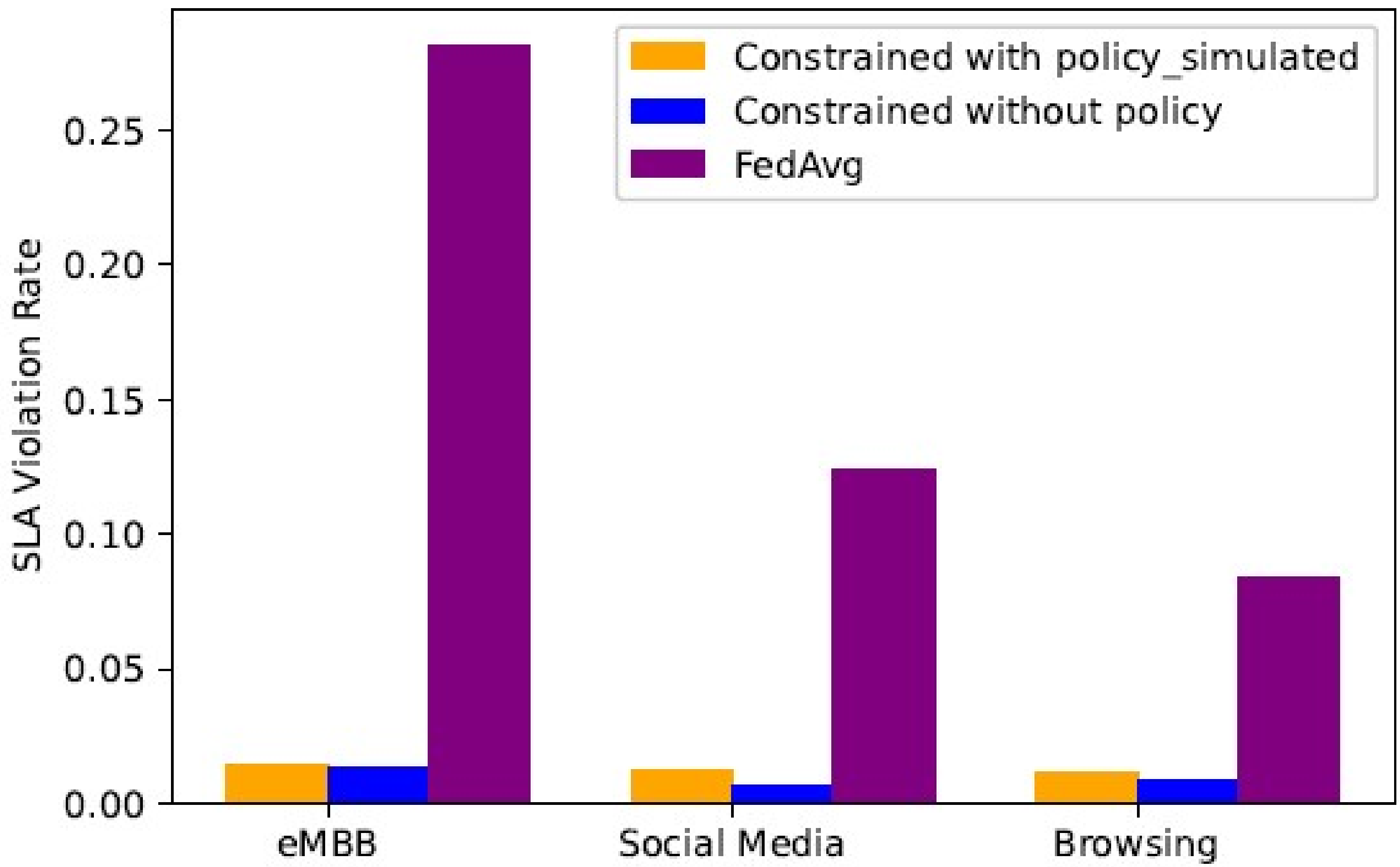


Scalability

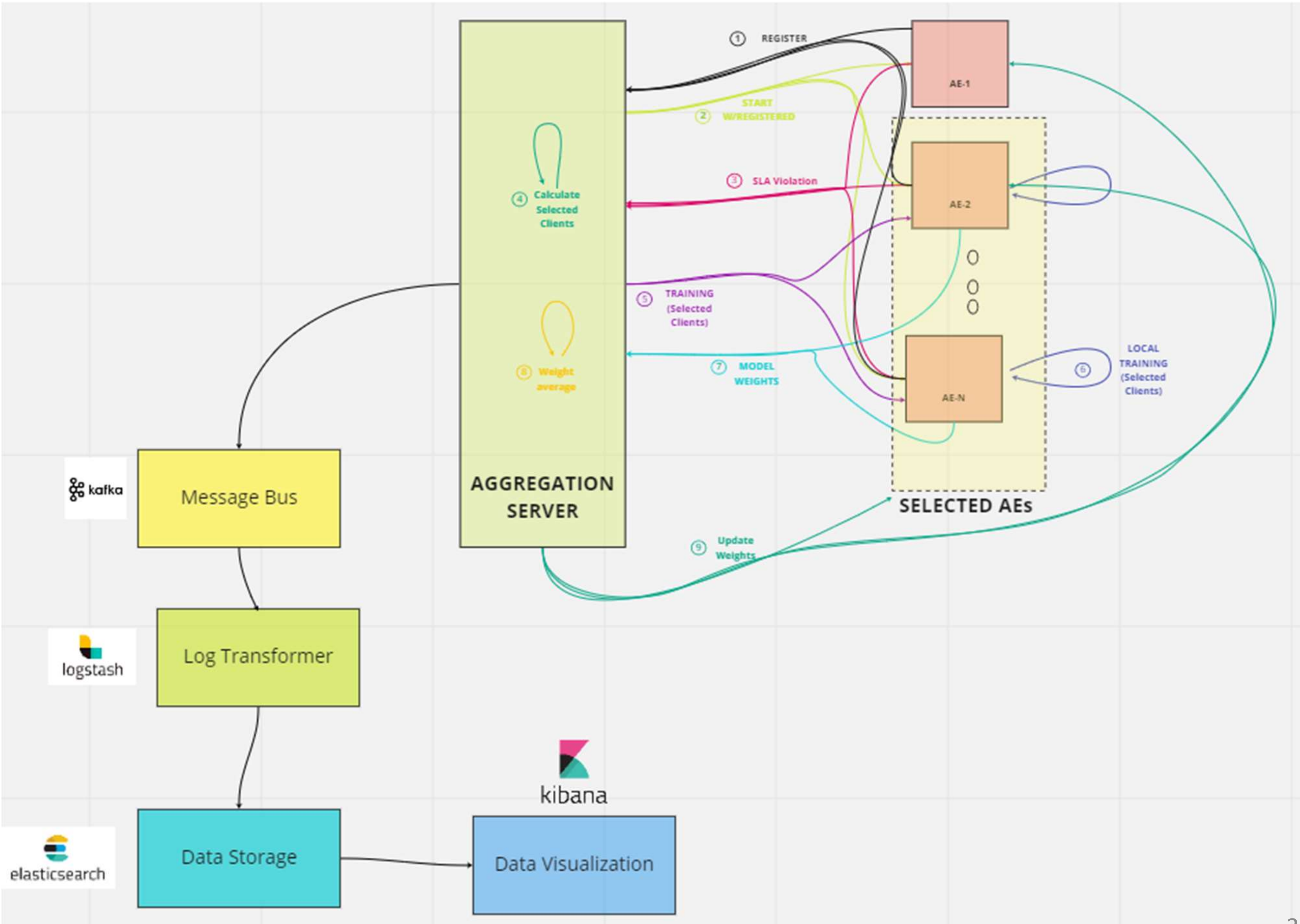
S. Roy, H. Chergui, L. Sanabria-Russo and C. Verikoukis, "**A Cloud Native SLA-Driven Stochastic Federated Learning Policy for 6G Zero-Touch Network Slicing**," IEEE ICC 2022.

SLA

- Select 50 AEs out of 100 (Simulated)

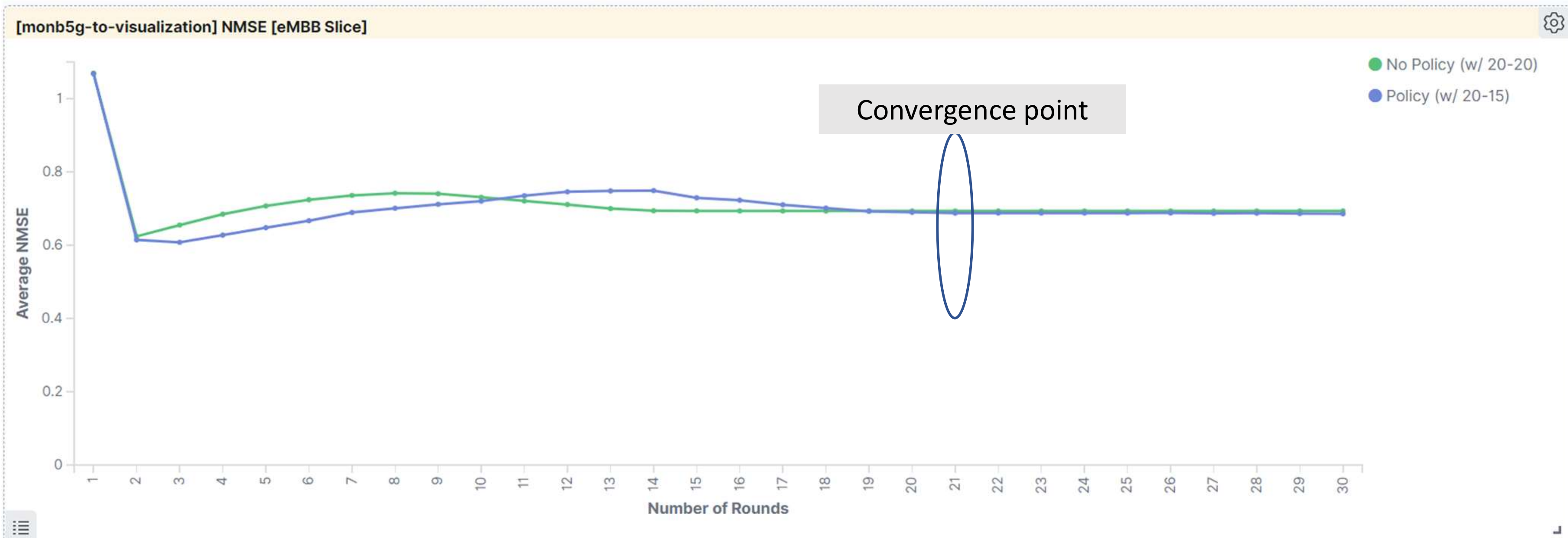


Visualization Setup

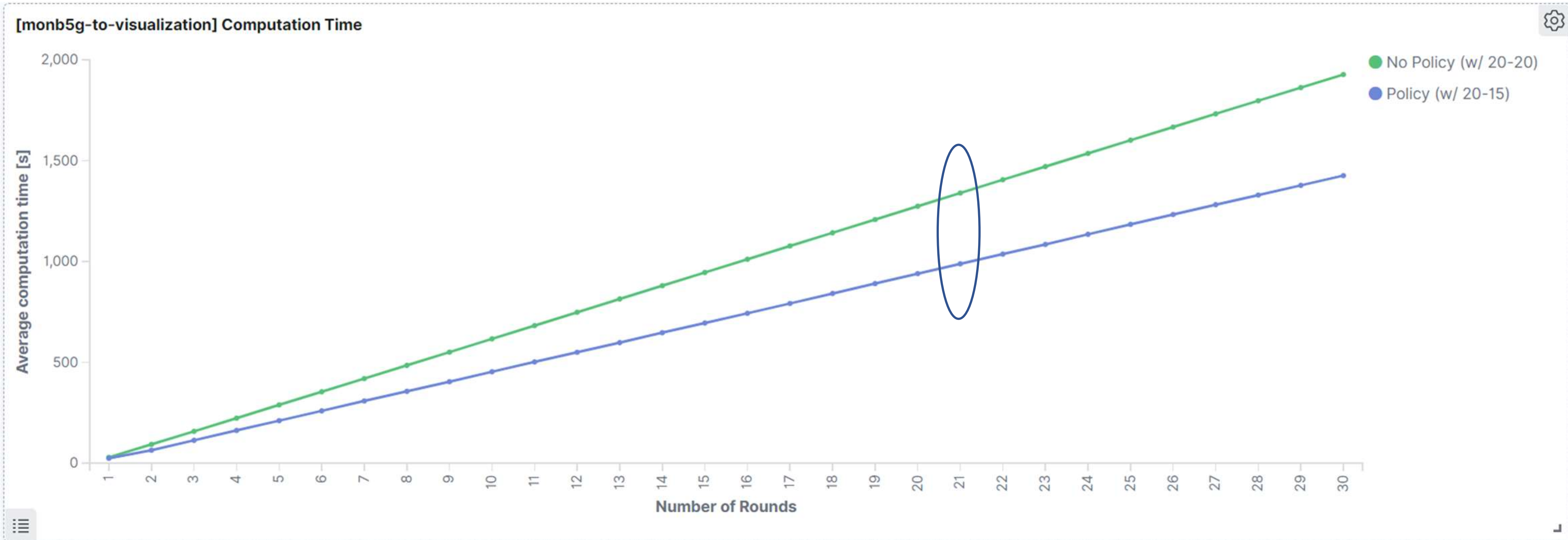


Demo and Kibana Dashboard

Kibana Dashboard



Kibana Dashboard



Kibana Dashboard & Overhead

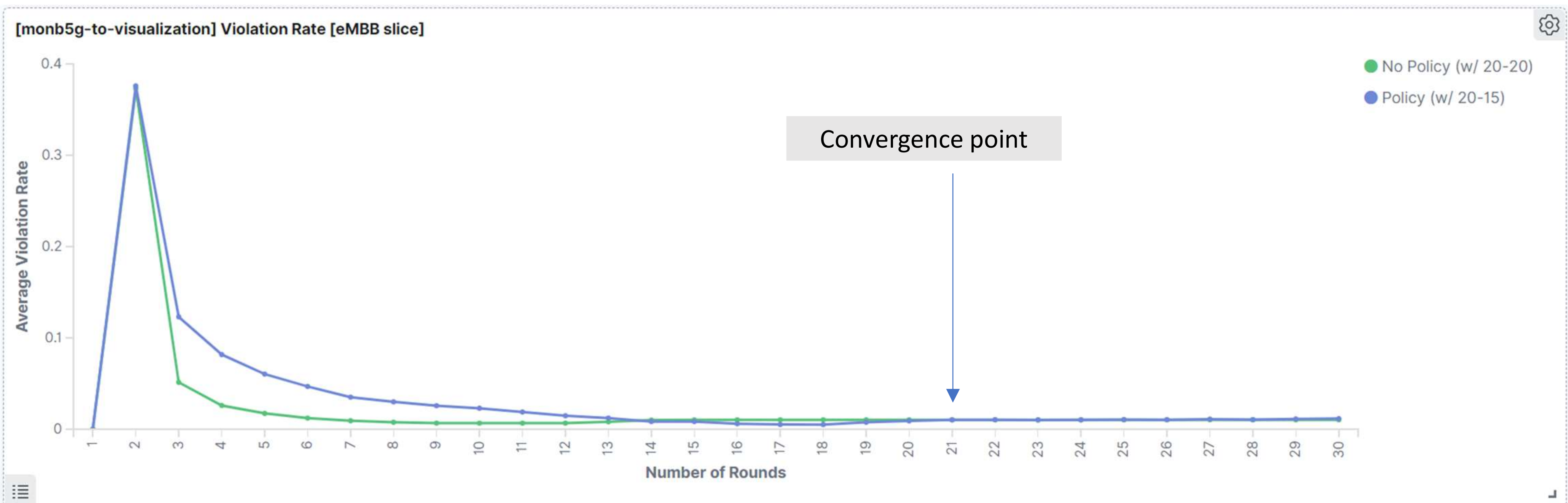
- **Uplink Overhead** ≈ # FL Rounds x # Selected Clients x # Weights x # 32 bits

Convergence at round 21 (SLA violation reaches 0.01 and Loss variation is low)

➔

  - Policy uplink overhead is ≈ **30.2 KB**
  - No Policy FL uplink overhead is ≈ **40.3 KB**

- **Reduction in overhead** compared to non-policy FL (or vanilla/traditional FL) **=> %25**



References

- H. Chergui, L. Blanco and C. Verikoukis, "**Statistical Federated Learning for Beyond 5G SLA-Constrained RAN Slicing**," in IEEE Transactions on Wireless Communications, vol. 21, no. 3, pp. 2066-2076, March 2022.
- S. Roy, H. Chergui, L. Sanabria-Russo and C. Verikoukis, "**A Cloud Native SLA-Driven Stochastic Federated Learning Policy for 6G Zero-Touch Network Slicing**," IEEE ICC 2022.
- H. Chergui, L. Blanco , L. A. Garrido, K. Ramantas, S. Kuklinski, A. Kasentini, S. Kuklinksli, "**Zero-Touch AI-Driven Distributed Management for Energy-Efficient 6G Massive Network Slicing**," in IEEE Network, vol. 35, no. 6, pp. 43-49, November/December 2021.
- H. Chergui, A. Ksentini, L. Blanco and C. Verikoukis, "**Toward Zero-Touch Management and Orchestration of Massive Deployment of Network Slices in 6G**," in IEEE Wireless Communications, vol. 29, no. 1, pp. 86-93, Feb. 2022

Thank You!!!

CTTC

This Project has received funding from the European Union´s Horizon 2020 research and innovation programme under Grant Agreement No. 871780