Resource Management for Edge-Assisted Learning with Deterministic Reliability Constraints

Francesco Binucci[1,2], Osvaldo Simeone[3], and Paolo Banelli[1]
[1]Department of Engineering, University of Perugia, Via G. Duranti 93, 06128, Perugia, Italy
[2]Consorzio Nazionale Interuniversitario per le Telecomunicazioni, Viale G.P. Usberti, 181/A, 43124, Parma, Italy
[3] KCLIP Lab, CIIPS, Department of Engineering, King's College London, London, United Kingdom

***Abstract*—We consider a novel resource allocation framework designed to achieve optimal resource management for edge-assisted inference tasks. This is obtained by a new optimization approach, addressed as conformal Lyapunov optimization, which integrates online conformal risk control (O-CRC) with conventional Lyapunov optimization (LO). Unlike traditional LO, this approach ensures compliance with deterministic long-term reliability constraints. Simulation results, based on an edge-assisted segmentation task, demonstrate the effectiveness of the proposed method in balancing energy consumption and inference performance, while maintaining strict control over deterministic long-term constraints, related to the false negative segmentation rate.**

***Index Terms*—Conformal Risk Control, Lyapunov Optimization, online optimization, resource allocation, mobile edge computing, edge inference**

I. Introduction

Mobile edge computing emerged as a key enabler to deliver cost-effective machine learning (ML) and artificial intelligence (AI) services in new generation mobile networks [1]. In this context, optimal resource allocation of transmission and computational resources is crucial to guarantee the efficient execution of AI services under specific constraints [2] which, depending on the task, may focus on energy consumption, latency, and task reliability.

Lyapunov optimization (LO) is a standard optimization tool employed in developing dynamic resource allocation strategies for networked systems [3]. Relying on queuing theory, LO can effectively tackle long-term average network cost minimization, under long-term average constraints. However, in many use cases, such as autonomous driving, it is necessary to satisfy strict rather than average reliability guarantees, which LO fails to meet. Thus, to overcome this limitation, we propose to employ conformal Lyapunov optimization (CLO), a novel resource allocation framework that pursues average long-term network cost minimization under strict reliability requirements, relying on LO and online-conformal risk control (O-CRC) [4].

The work of Francesco Binucci and Paolo Banelli was supported by the European Union - Next Generation EU under the Italian National Recovery and Resilience Plan (NRRP), Mission 4, Component 2, Investment 1.3, CUP F83C22001690001/E83C22004640001, partnership on "Telecommunications of the Future" (PE00000001 - program "RESTART"). The work of Osvaldo Simeone was partially supported by the European Union's Horizon Europe project CENTRIC (101096379), by the Open Fellowships of the EPSRC (EP/W024101/1) and by the EPSRC project (EP/X011852/1).

Related works. The effectiveness of Lyapunov optimization in optimal resource allocation for edge-assisted inference tasks is confirmed by many works in the recent literature [5]–[7]. In [8] the authors developed various resource allocation strategies for edge-assisted AI/ML tasks focused on the energy, latency and accuracy trade-offs, while [9] investigates LO strategies for optimal computation offloading in mobile cloud computing scenarios. Furthermore, the works [10] and [11] consider LO-based strategies for ultra reliable and low latency communications (URLLC), showing the effectiveness of LO in controlling higher-order statistical moments (e.g., outage probabilities). Other interesting examples can be found in [12]–[14], where LO has been employed in energy-harvesting scenarios, and in [15], [16], where federated learning scenarios are considered.

Differently, online conformal risk control (O-CRC) [4], exploiting the conformal prediction (CP) framework [17], it is designed for inference tasks whose outcome takes the form of a *prediction set*. Typical examples are image segmentation or multi-label classification [18]. More specifically, O-CRC is precisely designed to control long-term reliability metrics in online learning environments and gives theoretical reliability guarantees that the true outcome of the inference task is included in the predicted set with a specified reliability level.

Many works in the recent literature employ CP/O-CRC in networked systems [19]–[21]. [19] proposes a CRC approach to detect occupied subbands in unlicensed spectrum access, while [21] proposes a CP-based approach for dynamical scheduling of URLLC traffic. Focusing on edge-inference scenarios, the approach proposed in [22] leverages on CP arguments to control the uncertainty in federated inference tasks, while work [23] shows the effectiveness of O-CRC in tackling long-term reliability constraints, simultaneously maximizing the inference performance in image segmentation tasks. Except for [23], none of the aforementioned works considers network cost optimization under deterministic long-term guarantees. However, the scope of [23] is not general, as it is tailored only to sensors networks with star topology.

Our contributions. This paper presents CLO, a novel framework for optimal dynamic resource allocation that ensures deterministic reliability constraints in end-to-end decision processes for general wireless networked systems. Simulation results demonstrate the effectiveness of the proposed framework in pursuing network cost optimization under strict

ISBN 978-3-903176-73-7 © 2025 IFIP

reliability constraints, and highlight its advantages over standard resource allocation strategies based on LO.

II. System Model

A. Network Model

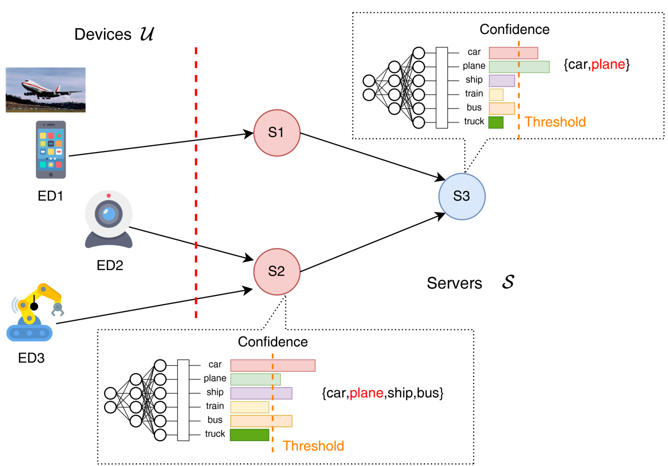

Fig. 1: Hierarchical scenario. Moving toward the cloud for better inference (i.e., smaller prediction sets), with increased latency and energy consumption.

We represent the network as a directed graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$, where $\mathcal{N}$ is the set of nodes and $\mathcal{E} \subseteq \{(n, m) : n, m \in \mathcal{N}, n \neq m\}$ defines the links. The nodes are partitioned as $\mathcal{N} = \mathcal{U} \cup \mathcal{S}$, where $\mathcal{U}$ denotes the set of edge devices (ED) or users, and $\mathcal{S}$ denotes the set of edge or cloud servers. EDs offload data to servers to perform inference tasks under reliability constraints. Each server in $\mathcal{S}$ hosts an inference model for processing data units (DU) from EDs. The network follows a hierarchical structure, depicted in Figure 1, where the computational capacity, and the associated inference accuracy, increase with the distance from EDs [24].

B. Data Acquisition and Processing

We consider a discrete time axis with time-slots of fixed duration δ. For each time-slot, each k-th ED may generate a new inference task. We model the generation of new inference tasks with the random variable $A^k(t) \sim \text{Bern}(\lambda^k)$, where $\lambda^k \in [0, 1]$ is the probability of a generation event. We define the vector $\mathbf{A}(t) = \{A^k(t)\}_{k=1}^K$ to collect the number of tasks generated at the t-th time slot. We denote as $\tau^k(t)$ the specific inference task, encoded with W^k bits. The tasks generated at the t-th slot are collected in the vector $\mathbf{T} = \{\tau^k(t)\}_{k=1}^K$.

The decision on the task is taken at the server s at some later time-slot, denoted by the variable $T^k_{\text{dec}}(t) \geq t$. We also denote as $T^k_s(t)$ the generation time of the task belonging to the k-th ED, and completed by the server s during the t-th slot. Inference performance is affected by the complexity of the model deployed on server s, and by the difficulty of the task $\tau^k(t)$. This behavior is captured by the loss $L_s(\tau, \theta)$ for any decision taken by the server s for a given task τ, where the hyperparameter θ allows tuning the decision strategy acting as a threshold on the confidence level at the output of the inference model (see Figure 1), as detailed in the next section.

C. Timeline

The time slots are partitioned into frames indexed by $f = 0, 1, \ldots$, each composed of S slots. Thus, in a time horizon of T slots, we have $F = T/S$ frames. This time axis organization is meaningful, as it allows for monitoring the average inference performance on the considered frame. Furthermore, it is crucial to provide theoretical guarantees on the convergence of the algorithm to a close-to-optimal solution, as we will detail in Section IV.

D. Reliability and Precision in Set Predictions

Herein we focus on binary image segmentation tasks where, for any input image x, the goal is to provide a binary mask $\mathcal{C}(x, \theta)$ encoding the pixels belonging to a target, as

$$\mathcal{C}(x, \theta) = \{(i, j) : p(i, j|x) \geq \theta\}, \tag{1}$$

where (i, j) are the pixels coordinates, and $p(i, j|x)$ is the estimated probability that pixel (i, j) belongs to the object of interest (e.g., an obstacle to be avoided) [18]. We aim to control the false negative rate (FNR) loss, given by the fraction of pixels belonging to the target not included in the set $\mathcal{C}(x, \theta)$, i.e.,

$$L_s(x, \theta) = \frac{\left|y \cap \overline{\mathcal{C}}(x, \theta)\right|}{|y|}, \tag{2}$$

where y is the set of pixels including the object of interest and $\overline{\mathcal{C}}(x, \theta)$ is the set complement of $\mathcal{C}(x, \theta)$. FNR (2) is a reliability loss that increases for increasing values of the hyperparameter θ. Thus, reducing the hyperparameter θ increases the reliability level of the decision. In general, we are interested in the class of loss functions $L_s(\tau, \theta)$ that satisfy the following assumption.

Assumption 1. The reliability loss $L_s(\tau, \theta)$ is a non-decreasing function in the hyperparameter θ for each server $s \in \mathcal{S}$ and for each task τ. Furthermore, it is bounded in the interval $[0, 1]$, and it satisfies the equality

$$L_s(\tau, 0) = 0, \quad \text{for each } s \in \mathcal{S} \text{ and } \tau. \tag{3}$$

According to Assumption 1, the highest reliability level is achieved by pushing the hyperparameter θ towards 0. However, this leads to less precise (i.e., informative) decisions. To computationally measure this trade-off, we introduce a precision loss $F_s(\tau, \theta)$, adhering to the following assumption.

Assumption 2. The precision loss $F_s(\tau, \theta)$ is a non-increasing function in the hyperparameter θ for each $s \in \mathcal{S}$ and for each task τ. Furthermore, it is bounded in the interval $[0, 1]$, and it satisfies the equality

$$F_s(\tau, 0) = 1, \quad \text{for each } s \in \mathcal{S} \text{ and } \tau. \tag{4}$$

In a segmentation problem a meaningful precision loss is given by the fraction between false positive pixels and the size of the object of interest, i.e.,

$$F_s(x, \theta) = \min\left(\frac{|\overline{y} \cap \mathcal{C}(x, \theta)|}{|y|}, 1\right). \tag{5}$$

E. Transmission and Queuing Model

We focus on the conventional queuing model for multi-hop wireless networks described in [3]. At each time-slot, network nodes can either process a portion of the queued tasks or forward them to another node. During each time-slot, each server has the capacity to process an inference task for each ED. To formalize this, we introduce the following variable

$$I_s^k(t) = \begin{cases} 1, & \text{if server } s \text{ processes a task for the } k\text{-th ED} \\ 0, & \text{otherwise.} \end{cases} \tag{6}$$

Furthermore, we impose a computational constraint on the maximum number of DUs processable by the server s, i.e., $\sum_{k=1}^{K} I_s^k(t) \leq I_s^{\max}$. We collect the decision variables in the set $\mathbf{I}(t) = \{I_s^k(t)\}_{k\in\mathcal{U}, s\in\mathcal{S}}$.

Similarly, we denote by $\mathbf{R}(t) = \{R_{n,m}^k(t)\}_{(n,m)\in\mathcal{E}, k\in\mathcal{U}}$ the indicator variables that specify whether the links are utilized for transmitting a data unit by the k-th edge device (ED) during the t-th time slot. The variable $R_{n,m}^k(t)$ reads as

$$R_{n,m}^k(t) = \begin{cases} 1, & \text{if link } (n,m) \text{ carries a DU of ED } k \\ 0, & \text{otherwise.} \end{cases} \tag{7}$$

We also impose a transmission constraint, enforcing a restriction on the total number of DUs that may traverse any given link (n,m), i.e., $\sum_{k=1}^{K} R_{n,m}^k(t) \leq R_{\mathrm{n,m}}^{\max}$.

EDs inject DUs into the network, which are stored in separate transmission queues. Each node n maintains K queues, one for each ED. Let $Q_n^k(t)$ be the queue state at node n for ED k, measured in DUs. Its evolution is given by

$$\begin{aligned} Q_n^k(t+1) = \max\Bigg(0, Q_n^k(t) - &\sum_{(n,m)\in\mathcal{E}} R_{n,m}^k(t) - \mathbb{1}\{n\in\mathcal{S}\} I_n^k(t)\Bigg) \\ &+ A^n(t)\mathbb{1}\{n\in\mathcal{U}\} + \sum_{(l,n)\in\mathcal{E}} R_{l,n}^k(t). \end{aligned} \tag{8}$$

The queue is drained by the DUs generated by the k-th user and subsequently transmitted or processed by the node n, while it is fed by the incoming traffic from other nodes and, if the node is part of the user set $\mathcal{U}$, by a newly generated task.

The transmission model follows the standard model for multi-hop systems [3], exploiting the links channel states $\mathbf{S}(t) = \{S_{n,m}(t)\}_{(n,m)\in\mathcal{E}}$ and the allocated powers $\mathbf{P}(t) = \{P_{n,m}(t)\}_{(n,m)\in\mathcal{E}}$ to characterize the Shannon channel capacity $\mu_{n,m}(S_{n,m}(t), P_{n,m}(t))$, which determines the transmission latency time $D_{n,m}^k(t) = W^k/\mu_{n,m}(t)$. Thus, the energy consumption for the transmission of a DU generated by the k-th ED at the t-th slot across the link (n,m) is

$$E_{n,m}^k(t) = P_{n,m}(t) D_{n,m}^k(t), \tag{9}$$

and the overall transmission cost is given by

$$E_{\text{tot}}(t) = \sum_{k=1}^{K} \sum_{(n,m)\in\mathcal{E}} R_{n,m}^k(t) E_{n,m}^k(t). \tag{10}$$

F. Performance Metrics

Optimizing the set of reliability hyperparameters $\mathbf{\Theta}(t) = \{\theta^k(t)\}_{k=1}^K$ primarily aims to enforce a deterministic worst-case constraint. Specifically, we want to ensure over time that the average (per frame) reliability loss of the tasks belonging to the k-th ED approaches the target value r^k. Mathematically, this is expressed as

$$\overline{L}^k = \frac{1}{F}\sum_{f=0}^{F-1} \frac{1}{N_f^k} \sum_{t=fS+1}^{(f+1)S} \sum_{s\in\mathcal{S}} I_s^k(t) L_s^k(t) \leq r^k + o(1), \tag{11}$$

where

$$L_s^k(t) = L_s(\tau^k(T_s^k(t)), \theta^k(t)) \tag{12}$$

is the loss accrued by a decision taken at time t by the server s on the task $\tau^k(T_s^k(t))$, generated at the slot $T_s^k(t)$; the quantity $N_f^k = \sum_{t=fS+1}^{(f+1)S} \sum_{s\in\mathcal{S}} I_s^k(t)$ in (12) denotes the number of DUs of the k-th ED whose decisions on have been taken within the f-th frame; and the function $o(1)$ tends to zero as $F \to \infty$. The constraint defined in (11) is deterministic in the sense that it must be satisfied for each run of CLO.

The optimization objective is given by the weighted sum of the transmission energy (10) and of the overall precision loss across all the EDs, i.e.,

$$J(t) = E_{\text{tot}}(t) + \eta F_{\text{tot}}(t), \tag{13}$$

where $\eta \geq 0$ is a multiplier used to explore the energy/precision trade-off. The overall imprecision is given by

$$F_{\text{tot}}(t) = \sum_{k=1}^{K} \sum_{s\in\mathcal{S}} I_s^k(t) F_s(\tau^k(T_s^k(t)), \theta^k(t)), \tag{14}$$

with $F_s(\tau^k(T_s^k(t)), \theta^k(t))$ denoting the precision loss accrued by the decision taken by the server s on the DU $\tau^k(T_s^k(t))$.

G. Problem Formulation

Overall, we aim to addressing the optimization problem

$$\begin{aligned} \min_{\mathbf{\Phi}(t)} \quad & \lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \mathbb{E}\{J(t)\} \\ \text{s.t.} \quad & \text{(a) long-term reliability constraints (11)} \quad \forall k, \\ & \text{(b) } Q_n^k(t) \text{ are mean-rate stable} \quad \forall k, n, \\ & \text{(c) } P_n(t) \leq P_n^{\max}, \ \sum_{k=1}^{K} I_s^k(t) \leq I_s^{\max} \qquad \forall n, s, t, \\ & \text{(e) } \sum_{k=1}^{K} R_{n,m}^k(t) \leq R_{n,m}^{\max} \qquad \forall (n,m)\in\mathcal{E}, t \end{aligned} \tag{15}$$

where $\mathbf{\Phi}(t) = \{\mathbf{I}(t), \mathbf{R}(t), \mathbf{P}(t), \mathbf{\Theta}(t)\}$ is the set of the optimization variables. Problem (15) pursues the average energy/precision trade-off $J(t)$ optimization under: (a) long-term deterministic reliability constraints; (b) mean-rate stability of all the queues; (c) transmission power constraints, and maximum processing capabilities for each server; (e) maximum transmission capacities for each link.

We now discuss the development of a dynamic optimization strategy to effectively tackle problem (15).

III. Conformal Lyapunov Optimization

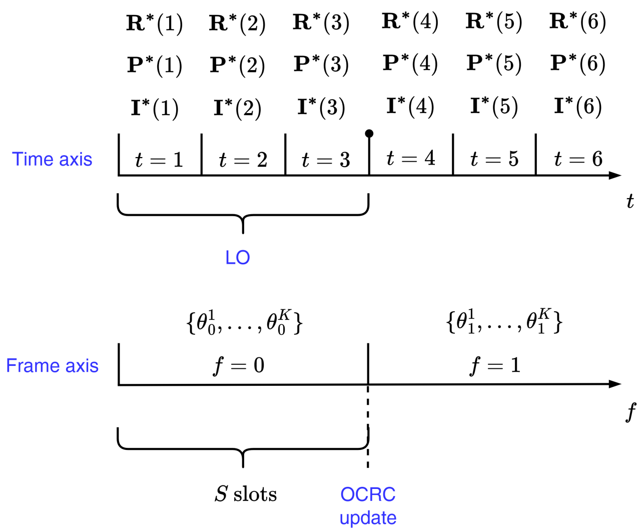

Fig. 2: Time axis organization. LO optimization within a fame and O-CRC hyper-parameters update by frames.

The classical LO framework is unable to address the long-term reliability constraint (15a) since it only accommodates average long-term constraints and does not support deterministic worst-case constraints of this type. Conversely, O-CRC is designed to handle deterministic constraints like (15a), but is not intended for optimization tasks.

We first note that purging problem (15) of the constraint (15a), and fixing the reliability-controlling variables $\mathbf{\Theta}(t)$, the optimization of $\{\mathbf{P}(t), \mathbf{I}(t), \mathbf{R}(t)\}$ falls into the class of problems solvable by LO. Thus, CLO envisages the following steps, depicted in Figure 2:

- We define a frame size S, during which the reliability hyperparameters remain constant. The reliability hyperparameter at the f-th frame is denoted as $\theta_f^k = \theta^k(fS+1)$;
- Within the f-th frame we perform resource allocation according to the standard LO methodology [3], as detailed in Section III-B;
- At the end of each time frame, the reliability hyperparameter is updated on the basis of the average loss accrued within the frame, as detailed in Section III-A.

A. Controlling the Reliability Constraints by CLO

To update the reliability hyperparameters $\mathbf{\Theta}_f = \{\theta_f^k\}_{k=1}^K$, CLO assumes that some *feedback* is available on the average loss associated with the decisions that have been already taken. This feedback is obtained by averaging the losses $L_s^k(t)$, as given in (12), over all the DUs processed within the slots of the f-th frame. Specifically, the feedback is calculated as

$$\overline{L}_f^k = \frac{1}{N_f^k} \sum_{t=fS+1}^{(f+1)S} \sum_{s \in \mathcal{S}} I_s^k(t) L_s^k(t). \tag{16}$$

Once the feedback for all decisions made within the f-th frame is collected, CLO updates the reliability hyperparameters as follows [4]

$$\theta_{f+1}^k = \theta_f^k + \gamma^k \mathbb{1}\{N_f^k > 0\}(r^k - \overline{L}_f^k), \tag{17}$$

where $\gamma^k > 0$ is the learning rate. According to (17), if the reliability constraint r^k is violated within the frame f, i.e., if $\overline{L}_f^k > r^k$, the variable θ_f^k is reduced, resulting in $\theta_{f+1}^k \leq \theta_f^k$. This adjustment will ensure more reliable (and less precise), decisions for ED k in the next frame $f+1$. On the other hand, when the reliability constraint is satisfied within the frame f, i.e., if $\overline{L}_f^k < r^k$, the update (17) increases θ_f^k, leading to more precise decisions at the next time frame.

B. LO Driven Within Frame Resource Allocation

Following LO derivations [3], for each slot $t \in [fS+1, (f+1)S]$, we end up with the following instantaneous problem

$$\begin{aligned} \min_{\{\mathbf{P}(t), \mathbf{I}(t), \mathbf{R}(t)\}} \quad & VJ(t) - \sum_{(n,m)\in\mathcal{E}, k\in\mathcal{U}} U_{n,m}^k(t) R_{n,m}^k(t) \\ & - \sum_{n\in\mathcal{N}, k\in\mathcal{U}} \mathbb{1}\{n \in \mathcal{S}\} Q_n^k(t) I_n^k(t) \\ & \text{s.t.} \quad \text{(15c)-(15e)}, \end{aligned} \tag{18}$$

where $V > 0$ is a hyperparameter; and

$$U_{n,m}^k(t) = Q_n^k(t) - Q_m^k(t) \tag{19}$$

is the differential backlog on link $(n, m) \in \mathcal{E}$ for ED k [3].

Problem 18 does not admit a closed-form solution, given the lack of a closed-form expression for the precision loss (14). A possible approach to face this consists in training a set of neural networks (NNs) to predict the precision loss of the inference models deployed at the servers. Furthermore, the problem is mixed integer since the variables $\mathbf{I}(t)$ and $\mathbf{R}(t)$ are discrete. Evaluation of a near-optimal solution is still possible through numerical solvers, as detailed in Section V.

IV. Theoretical Guarantees

We discuss herein the theoretical guarantees of CLO [1]. For the following analysis, we define the process capturing the environmental stochasticity as $\mathbf{\Omega}(\mathbf{t}) = \{\mathbf{A}(t), \mathbf{S}(t), \mathbf{T}(t)\}$.

Proposition 1. Under Assumptions 1 and 2, as the number of frames $F \to \infty$, the deterministic long-term reliability constraint (15a) is satisfied by CLO for each realization of $\mathbf{\Omega}(t)$. Furthermore, the average reliability loss (11) satisfies the following upper bound for any number of frames F

$$\frac{1}{F} \sum_{f=0}^{F-1} \overline{L_f^k} \leq r^k + \underbrace{\frac{M + \gamma^k - \theta_0^k}{\gamma^k F}}_{U(M)}, \tag{20}$$

where $M = \max_f \{\theta_f^k\} - \gamma^k$.

[1]The Proof of Proposition 1 relies on Theorem 1 in [4], while the proof of Proposition 2 is based on Theorem 4.8 in [3]

If we do not have any prior information on the distribution of the hyperparameter θ_f^k, we can set a worst-case deterministic upper bound imposing $M = 1$ for a segmentation task [4]. Otherwise, the value of M can be estimated by observing the maximum of the hyperparameters θ_f^k after the execution of CLO. Proposition 1 demonstrates that, with respect to the reliability constraint, selecting a small number of slots per frame, S, is advantageous as it increases the number of frames F for a given total number of slots $T = FS$. However, it will be shown shortly that larger values of S are beneficial for reducing the average cost.

Before analyzing the performance of CLO in network cost optimization, we make the following statistical assumption.

Assumption 3. The process $\mathbf{\Omega}(t)$ is i.i.d. over time slots.

Proposition 2. Let

$$G(t) = \sum_{n=1}^{N}\sum_{k=1}^{K} Q_n^k(t)^2 \tag{21}$$

be the Lyapunov function for the system's queues, and assume the condition $\mathbb{E}\{G(fS+1)\} \leq \infty$. Under Assumption 3, denoting by J_f^* the minimum time average cost at the f-th frame achievable by any policy meeting constraint (15b), CLO satisfies the following properties, where μ is a constant term.:

$$\text{(i)}\ \frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\{J(t)\} \leq \frac{1}{F}\sum_{f=0}^{F-1}\left[J_f^* + O\left(\frac{1}{S}\right)\right] + \frac{\mu}{V} \tag{22}$$

(ii) constraint (15b) is satisfied,

This proposition demonstrates that CLO achieves near-optimal long-term performance while meeting all constraints in problem (15). The suboptimality of the solution is limited by a term of order $O(1/S)$. As a result, enhancing network performance necessitates increasing the frame size S.

V. Simulation Results and Conclusion

Setting: we consider a single-hop network with $K = 3$ EDs connected to an edge server (ES). EDs run low-complexity UNet segmentation NNs [25] based on a MobileNetV3 encoder, while the ES employs a ResNet50 encoder [2]. We consider Rayleigh channels with path loss $\mathrm{PL} = 90\,\mathrm{dB}$, a maximum transmit power $P_k^{\max} = 3.5\,\mathrm{W}$ for all k, and a noise power spectral density $N_0 = -174\,\mathrm{dBm/Hz}$. Transmission bandwidth is set to $B_{n,m} = 20$ MHz for all $(n, m) \in \mathcal{E}$, while the time-slot duration is set to $\delta = 50$ ms. The generation of new tasks follows a Bernoulli distribution with parameter $\lambda^k \in \{0.4, 0.8\}$ for all k. We consider a non-stationary setting, where λ^k may change every 100 slots with probability $p = 0.5$. We set $V = 200$, and $\eta \in [0.2, 0.8]$.

Task description: we focus on an edge-assisted binary segmentation task based on the Cityscapes dataset [26], divided into 10,000 training, and 10,000 test images of urban scenarios. Images are composed of $256 \times 256 \times 3$ px encoded with 32-bit, which leads to a task size $W^k = 768$ KB for all k. We converted the original multi-class segmentation task into a binary segmentation task by focusing only on car objects. We impose a reliability constraint on the FNR (2), while we consider (5) as a precision metric. To predict the precision loss (14) in resource allocation, we trained two low-complexity NNs through knowledge distillation to mimic the output of the inference models at the EDs and at ES.

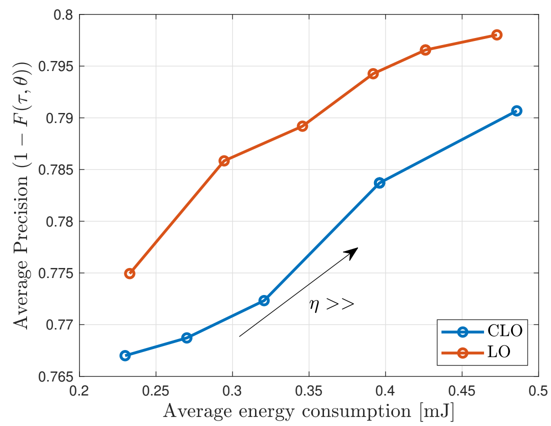

Fig. 3: Energy vs. precision trade-off for LO and CLO.

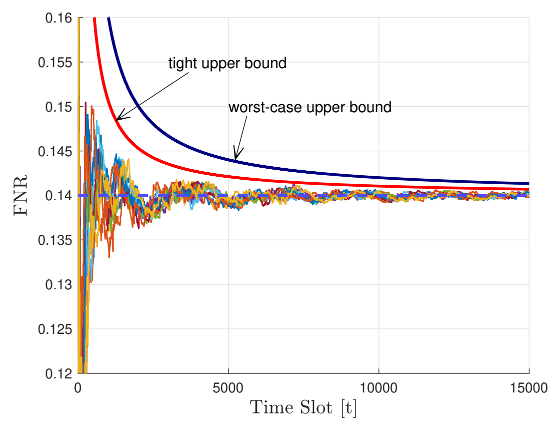

Fig. 4: Long-term FNR (11) for CLO ($\eta = 0.8$, $r^k = 0.14$).

Experimental evaluation: we compare CLO with the standard LO, for a reliability frame size $S = 10$. LO pursues average reliability by introducing a set of virtual queues $\{Z^k\}_{k=1}^K$, fed by instantaneous violations of the reliability constraint r^k [3]. Violations are weighted by a set of step sizes $\{\beta^k\}_{k=1}^K$ which, similarly to the learning rates $\{\gamma^k\}_k^K$ in (17), control the convergence speed [3]. To make fair comparisons, we set $\beta^k = 0.5$ and $\gamma^k = 0.5$ for all k. The per-slot problem (18) is solved using the MOSEK solver's Python API[3].

Unlike CLO, in LO the hyperparameters $\{\theta^k(t)\}_{k=1}^K$ are treated as discrete, belonging to the set $\{0.1, 0.2, \ldots, 0.9\}$, and optimized at each time-slot. To limit the computational complexity of the associated integer optimization problem, we

[2]NN implementations are provided by https://github.com/qubvel-org/segmentation_models.pytorch

[3]https://docs.mosek.com/latest/pythonapi/index.html

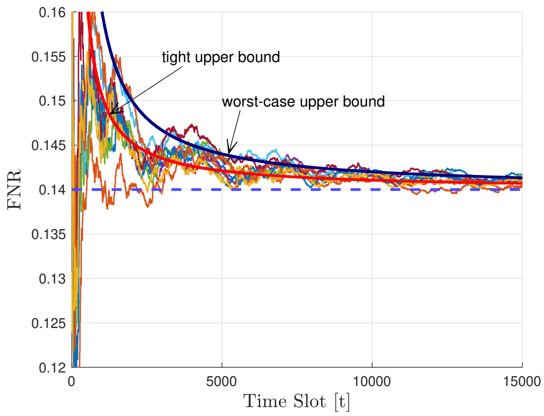

Fig. 5: Long-term FNR (11) for LO ($\eta = 0.8$, $r^k = 0.14$).

impose a common threshold in LO for all users, i.e., $\theta^k(t)$ is set to a unified value $\theta^*(t)$ for all k.

Figure 3 illustrates the trade-off between average energy consumption and precision, given by $1 - F(\tau, \theta)$. These plots have been obtained by running LO and CLO for increasing values of the penalty parameter η in (13), and averaging during the last 1,000 time slots, at the convergence of the reliability constraint. We note that LO achieves higher precision than CLO for the same energy consumption. This represents the price paid by CLO to ensure deterministic reliability guarantees. The reliability constraints are analyzed in Figures 4 and 5, which display the FNR across time for 10 realizations of LO and CLO, under similar energy consumption (corresponding to the rightmost points in Figure 3). Figures also include the worst-case and the a posteriori upper-bound in (20), obtained by estimating the value of M in (20) during the execution of CLO. Simulations confirm that CLO respects the deterministic upper bounds for any realization, while LO often violates them, especially as the time progresses and the bounds tighten.

To conclude, this paper presents conformal lyapunov optimization (CLO) which, by integrating Lyapunov optimization with conformal risk control, guarantees long-term *deterministic* reliability while optimizing the costs and precision of edge-assisted inference tasks in networks with multiple hierarchical servers. Simulations have confirmed the effectiveness of CLO and the expected advantages over Lyapunov-based resource management strategies. Future work may consider more realistic scenarios, including interfering users, transmission outages, computational constraints, and service delays.

References

[1] Z. Zhou, X. Chen, E. Li, L. Zeng, *et al.*, "Edge intelligence: Paving the last mile of artificial intelligence with edge computing," *Proceedings of the IEEE*, vol. 107, no. 8, pp. 1738–1762, 2019.

[2] Q. Zhang, L. Gui, F. Hou, *et al.*, "Dynamic task offloading and resource allocation for mobile-edge computing in dense cloud ran," *IEEE Internet of Things Journal*, vol. 7, no. 4, pp. 3282–3299, 2020.

[3] M. Neely, *Stochastic network optimization with application to communication and queueing systems*. Springer Nature, 2022.

[4] S. Feldman, L. Ringel, S. Bates, and Y. Romano, "Achieving risk control in online learning settings," *Transactions on Machine Learning Research*, 2023.

[5] Y. Jia, C. Zhang, Y. Huang, and W. Zhang, "Lyapunov optimization based mobile edge computing for internet of vehicles systems," *IEEE Transactions on Communications*, vol. 70, no. 11, pp. 7418–7433, 2022.

[6] C. Dong, S. Hu, X. Chen, and W. Wen, "Joint optimization with dnn partitioning and resource allocation in mobile edge computing," *IEEE Transactions on Network and Service Management*, vol. 18, no. 4, pp. 3973–3986, 2021.

[7] Y. Mao, J. Zhang, S. H. Song, and K. B. Letaief, "Stochastic joint radio and computational resource management for multi-user mobile-edge computing systems," *IEEE Transactions on Wireless Communications*, vol. 16, no. 9, pp. 5994–6009, 2017.

[8] M. Merluzzi, P. D. Lorenzo, and S. Barbarossa, "Wireless edge machine learning: Resource allocation and trade-offs," *IEEE Access*, vol. 9, pp. 45377–45398, 2021.

[9] Y. Li, S. Xia, M. Zheng, B. Cao, and Q. Liu, "Lyapunov optimization-based trade-off policy for mobile cloud offloading in heterogeneous wireless networks," *IEEE Transactions on Cloud Computing*, vol. 10, no. 1, pp. 491–505, 2022.

[10] C. Kai, W. Liu, and W. Huang, "Lyapunov optimization-based user scheduling and beamforming design for urllc systems," in *2023 IEEE Wireless Comm. and Networking Conf. (WCNC)*, pp. 1–6, IEEE, 2023.

[11] B. Chang, L. Zhang, L. Li, *et al.*, "Optimizing resource allocation in urllc for real-time wireless control systems," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 9, pp. 8916–8927, 2019.

[12] Y. Mao, J. Zhang, and K. B. Letaief, "A lyapunov optimization approach for green cellular networks with hybrid energy supplies," *IEEE Jour. on Sel. Areas in Comm.*, vol. 33, no. 12, pp. 2463–2477, 2015.

[13] C. Tapparello, O. Simeone, and M. Rossi, "Dynamic compression-transmission for energy-harvesting multihop networks with correlated sources," *IEEE/ACM Transactions on Networking*, vol. 22, no. 6, pp. 1729–1741, 2014.

[14] C. Qiu, Y. Hu, Y. Chen, and B. Zeng, "Lyapunov optimization for energy harvesting wireless sensor communications," *IEEE Internet of Things Journal*, vol. 5, no. 3, pp. 1947–1956, 2018.

[15] Y. Sun, S. Zhou, Z. Niu, and D. Gündüz, "Dynamic scheduling for over-the-air federated edge learning with energy constraints," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 1, pp. 227–242, 2022.

[16] C.-H. Hu, Z. Chen, and E. G. Larsson, "Energy-efficient federated edge learning with streaming data: A lyapunov optimization approach," *IEEE Transactions on Communications*, vol. 73, no. 2, pp. 1142–1156, 2025.

[17] A. N. Angelopoulos and S. Bates, "A gentle introduction to conformal prediction and distribution-free uncertainty quantification," *arXiv preprint arXiv:2107.07511*, 2021.

[18] A. N. Angelopoulos, S. Bates, A. Fisch, L. Lei, and T. Schuster, "Conformal risk control," in *The Twelfth International Conference on Learning Representations*, 2024.

[19] H. Lee, S. Park, *et al.*, "Reliable sub-nyquist spectrum sensing via conformal risk control," *arXiv preprint arXiv:2405.17071*, 2024.

[20] K. M. Cohen, S. Park, O. Simeone, *et al.*, "Guaranteed dynamic scheduling of ultra-reliable low-latency traffic via conformal prediction," *IEEE Signal Processing Letters*, vol. 30, pp. 473–477, 2023.

[21] K. M. Cohen, S. Park, O. Simeone, and S. Shamai Shitz, "Calibrating ai models for wireless communications via conformal prediction," *IEEE Transactions on Machine Learning in Communications and Networking*, vol. 1, pp. 296–312, 2023.

[22] M. Zhu, M. Zecchin, S. Park, C. Guo, C. Feng, and O. Simeone, "Federated inference with reliable uncertainty quantification over wireless channels via conformal prediction," *IEEE Transactions on Signal Processing*, pp. 1–16, 2024.

[23] M. Zhu, M. Zecchin, S. Park, C. Guo, *et al.*, "Conformal distributed remote inference in sensor networks under reliability and communication constraints," *arXiv preprint arXiv:2409.07902*, 2024.

[24] J. Ren, D. Zhang, S. He, Y. Zhang, and T. Li, "A survey on end-edge-cloud orchestrated network computing paradigms: Transparent computing, mobile edge computing, fog computing, and cloudlet," *ACM Comput. Surv.*, vol. 52, oct 2019.

[25] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pp. 234–241, Springer, 2015.

[26] M. Cordts, M. Omran, S. Ramos, *et al.*, "The cityscapes dataset for semantic urban scene understanding," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.