Meta-Learning-Based Handover Management in NextG O-RAN

Michail Kalntis, *Member, IEEE*, George Iosifidis, José Suárez-Varela, Andra Lutu, and Fernando A. Kuipers, *Senior Member, IEEE*

***Abstract*—While traditional handovers (THOs) have served as a backbone for mobile connectivity, they increasingly suffer from failures and delays, especially in dense deployments and high-frequency bands. To address these limitations, 3GPP introduced Conditional Handovers (CHOs) that enable proactive cell reservations and user-driven execution. However, both handover (HO) types present intricate trade-offs in signaling, resource usage, and reliability. This paper presents unique, countrywide mobility management datasets from a top-tier mobile network operator (MNO) that offer fresh insights into these issues and call for adaptive and robust HO control in next-generation networks. Motivated by these findings, we propose CONTRA, a framework that, for the first time, jointly optimizes THOs and CHOs within the O-RAN architecture. We study two variants of CONTRA: one where users are a priori assigned to one of the HO types, reflecting distinct service or user-specific requirements, as well as a more dynamic formulation where the controller decides on-the-fly the HO type, based on system conditions and needs. To this end, it relies on a practical meta-learning algorithm that adapts to runtime observations and guarantees performance comparable to an oracle with perfect future information (universal no-regret). CONTRAis specifically designed for near-real-time deployment as an O-RAN xApp and aligns with the 6G goals of flexible and intelligent control. Extensive evaluations leveraging crowd-sourced datasets show that CONTRAimproves user throughput and reduces both THO and CHO switching costs, outperforming 3GPP-compliant and Reinforcement Learning (RL) baselines in dynamic and real-world scenarios.**

***Index Terms*—Mobility management, O-RAN, conditional handovers, meta-learning, network datasets, data analytics.**

I. Introduction

A. Motivation

MOBILITY management is a cornerstone of mobile communication systems, ensuring service continuity as users move through diverse coverage areas of deployed *cells*. At the heart of this capability lies the *handover* (HO) mechanism, which transfers active sessions from one cell to another. Despite decades of optimization and standardization [1], [2], HOs continue to face significant challenges in today's networks. In particular, dense deployments, heterogeneous radio access technologies (RATs), and the shift to higher frequencies increase the likelihood of HO delays and failures [3], [4]. These issues are compounded by the increasingly dynamic nature of user mobility and volatile signal conditions, which are suboptimally handled by existing *reactive* traditional HO (THO) procedures, as evidenced in our analysis with real-world data in Sec. IV.

While the 5G architecture introduced flexibility [5], [6], the fundamental approach to HO has remained largely unchanged. However, the 6G vision, expressed in various 3GPP workshops and white papers [7], [8], [9], [10], [11], paints a different picture, where mobility should be *intelligent, proactive, and relying on native Artificial Intelligence (AI)*. To support highly demanding 6G use cases, the network must adapt in real time to user requirements and the current network conditions [12]. Specifically, seamless mobility across different service domains, ultra-reliable, low-latency HO, and resource efficiency are identified as critical enablers for 6G [9], [10]. We thus call for a *paradigm shift* in mobility management, underpinned by (near) real-time data-driven control.

To mitigate these limitations, 3GPP introduced Conditional Handovers (CHOs) [1], [13]. The main idea is to *proactively* reserve resources in multiple *candidate* cells while signal conditions are favorable, and delegate the final HO decision to the user. This novel approach of *network-configured, user-decided HOs*, instead of the traditional *network-configured, user-assisted* HOs, has shown to reduce HO delays and failures [14], [15]. At the same time, CHOs introduce new challenges: selecting the optimal set of cells to prepare is highly context-sensitive; preparing too many cells leads to resource overutilization, while preparing too few risks service interruption due to fallback to traditional HOs [4], [16], [17], [18]. As CHO policies remain non-adaptive in practice, the trade-offs they involve are often suboptimally managed.

In this paper, we argue that next-generation (NextG) networks will benefit from AI-based solutions that jointly optimize traditional and conditional HOs and adapt the HO strategy to the current user requirements. In fact, 6G early releases are expected to include both network-triggered and user-triggered HO solutions [8]. At the same time, the rise of *O-RAN* provides a practical vehicle for deploying such intelligence since HO logic can now be implemented as near-real-time xApps, enabling a flexible and programmable AI-native radio access network intelligent controller (near-RT RIC); see Figure 1. This controller communicates with the O-RAN-compliant central, distributed, and radio units (O-CU,

Received 20 June 2025; revised 26 December 2025; accepted 30 December 2025. Date of publication 12 January 2026; date of current version 24 February 2026. This work was supported in part by the National Growth Fund through the Dutch 6G Flagship Project "Future Network Services" and in part by European Commission under Grant 101139270 (ORIGAMI), Grant 101192462 (FLECON-6G), and Grant 101167288 (ANTENNAE). *(Corresponding author: Michail Kalntis.)*

Michail Kalntis, George Iosifidis, and Fernando A. Kuipers are with Delft University of Technology, 2628 Delft, The Netherlands (e-mail: m.kalntis@tudelft.nl; g.iosifidis@tudelft.nl; f.a.kuipers@tudelft.nl).

José Suárez-Varela and Andra Lutu are with Telefónica Research, 28050 Madrid, Spain (e-mail: jose.suarez-varela@telefonica.com; andra.lutu@telefonica.com).

Digital Object Identifier 10.1109/JSAC.2026.3652037

1558-0008 © 2026 IEEE. All rights reserved, including rights for text and data mining, and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: University of Cyprus. Downloaded on September 28,2026 at 10:36:41 UTC from IEEE Xplore. Restrictions apply.

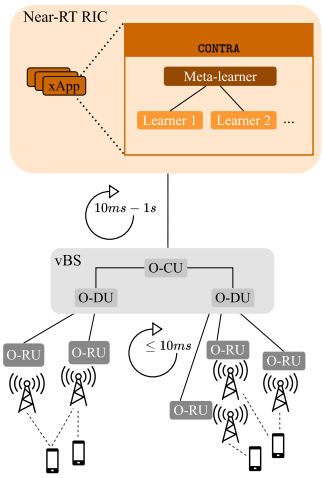


Fig. 1. CONTRA deployed as an xApp in near-RT O-RAN RIC.

O-DU, and O-RU, respectively), delegating the decisions to multiple cells and users. This architectural openness and AI-based control creates a timely opportunity to rethink mobility management for NextG networks.

In summary, in this paper, we answer the question: *How can we design a robust and principled AI-driven framework for jointly optimizing traditional and conditional handovers in NextG O-RAN architectures?* This question is motivated by the new mobility needs in 6G, technological developments in HO mechanisms, new AI-based algorithms that can potentially enable HO optimization under realistic conditions, and the lack of unified THO and CHO mechanisms.

B. Methods & Contributions

To answer this question, we first analyze new countrywide datasets of mobility events from a top-tier mobile network operator (MNO) in a European country. Unlike prior studies that focus on limited end-user measurement campaigns or confined regions (see Sec. II), our dataset spans 13.5K cells and 40M users, and provides comprehensive visibility into handover failures (HOFs), HO delays, signal fluctuations, and cell heterogeneity. Employing a statistical analysis, we quantify the conditional effect on HOFs and HO delays as a function of cell's frequency band, vendor, and location/time. We observe a Pareto-like distribution in signal fluctuation: for 20% of movements, the Signal-to-Interference-plus-Noise Ratio (SINR) varies by 100% within 1 s, which makes them ideal candidates for CHOs. These insights confirm the need for adaptive and robust HO control at user-level granularity and motivate the solution of this work.

We next introduce a first-of-its-kind *unified modeling framework* that jointly captures both traditional (explicit) and conditional (implicit) HO processes. Our model represents the decisions of a controller [11] in an O-RAN setting and accounts for user throughput, cell load, signaling overhead, and service delay due to HOs (switching costs). We study two variants of CONTRA: one with a priori HO type assignment per user, reflecting distinct service or user-specific requirements, and another where the controller determines on-the-fly the HO type based on system dynamics. The key novelty lies in treating HO type selection, candidate cell preparation, and cell association as a single online learning problem without relying on knowledge from underlying stochastic processes (e.g., SINR fluctuations, cost of signaling). We model the resource trade-offs and performance dependencies between HO types and across users, enabling per-user decision-making at the granularity of O-RAN near-real-time control loops.

To solve this problem, we propose CONTRA, a meta-learning-based algorithm (see Figure 1) that maintains a pool of learners (i.e., experts) tuned for different signal conditions and mobility patterns, and employs a meta-learner to track their performance. This design enables *adaptation* to changing network environments, types of user equipment (UE) and cells, without prior statistics, and ensures performance commensurate with that of an oracle with knowledge of the future. Our approach is based on the theory of online convex optimization (OCO) [19] and our formulation builds on recent advances in *smoothed online learning* (SOL) [20]. The proposed algorithm provides theoretical guarantees in terms of *expected dynamic regret*, a rigorous measure of how well the algorithm performs compared to the oracle, under a vast range of scenarios for the changing parameters. Finally, we evaluate CONTRA using the datasets from the top-tier MNO and actual users. Benchmarking it against 3GPP-compliant and Reinforcement Learning (RL) based THO and CHO mechanisms, we find that it provides higher user throughput, lower signaling cost, and improved robustness, particularly in volatile and real SINR environments. In summary, our contributions are:

- We present and analyze countrywide mobility datasets to shed light on parameters, such as frequency bands, vendors, and location/time, that affect HO failures and delays.
- Motivated by the 6G vision and the analyzed datasets, we model HO control as a learning problem and propose CONTRA, the first unified THO/CHO orchestration framework for real-time and robust HO control in NextG O-RAN. We study two variants of CONTRA: a static one with predefined HO types per user, and a dynamic one, where the controller decides the HO type on-the-fly. Our solution relies on a meta-learner that is oblivious to UE types, mobility, and network conditions, offering performance guarantees (expected dynamic regret).
- We release the source code[1] of our implementation online, under a permissive free software license, along with detailed documentation.
- We evaluate CONTRA using crowdsourced data and multiple scenarios against 3GPP-compliant and RL benchmarks. The experiments highlight CONTRA's *efficacy, deployability, and alignment with 6G goals* of intelligent Radio Access Network (RAN) functions.

Organization: Sec. II discusses the related work and Sec. III provides background on THOs and CHOs. Sec. IV presents our datasets and the findings from their analysis. Sec. V introduces the main system model and problem formulation, and in Sec. VI, we present the proposed solution

[1]https://github.com/MikeKalnt/conditional_traditional_handover_management

Authorized licensed use limited to: University of Cyprus. Downloaded on September 28,2026 at 10:36:41 UTC from IEEE Xplore. Restrictions apply.

for this first problem. Sec. VII extends our study to the model with dynamic handover type selection and we evaluate our solutions in Sec. VIII. Our study concludes in Sec. IX and the Appendix includes lemma proofs omitted from the main text.

II. Related Work

A. Measurements & Traditional Handovers

HOs are mainly studied with traces from UEs, which are inevitably limited to certain manufacturers, areas and devices [21]. Recent large-scale, network-side studies [3] provide more visibility, but do not propose models or solutions. Prior works have addressed the joint optimization of throughput and HOs [22], [23], [24], [25], [26], offering important insights into association strategies under delay constraints. Our objective aligns with these studies, and we extend their models to capture the delay characteristics and costs associated with THOs and CHOs, and heterogeneity across cells and UEs, by analyzing countrywide datasets. Separately, recent work has addressed HOs under minimal assumptions via online learning [27], but without considering CHOs. Our approach builds on these works by incorporating, for the first time, THO and CHO characteristics into a unified, adaptive learning framework.

B. Conditional Handovers

CHOs have been proposed as a solution for non-terrestrial networks [15], [28], [29], 5G NR-unlicensed systems [4], fast-moving users [4], as well as in beamforming and contention-free random access [14], [30]. Proposals for improving the CHO mechanism include [18], [31] which tweak the CHO thresholds to decrease HOFs; [32], [33] which use historical handover data to decide the cell preparations; and [34] which employs UE trajectory prediction to optimize the CHO decisions. Machine Learning-assisted approaches include [35], [36] which predict ideal association strategies based on SINR data or predict the SINRs values. These important works, however, propose heuristic solutions (no optimality guarantees) and/or rely on historical data for offline training of models, which in practice may be unavailable or non-representative of encountered conditions.

In contrast, motivated by our measurements that show this problem is dynamic across UEs, cells, and time, we leverage an adaptive optimization framework. Unlike other works that explicitly avoid decision changes in cell preparations [17], [37] or propose static optimization formulations [16] which, unavoidably, rely on heavy assumptions (static and known parameters), we propose a tunable approach to the network's preference switching model that can cope with time-varying and unknown cost values. In [38], a meta-learning approach with minimal assumptions is used but focuses only on CHOs and omits a fair scheduler for allocation, as shown in eq. (5).

C. Meta-Learning

Meta-learning is finding increasing applications in communication systems due to its robustness to distribution shifts and fast-adaptation [39]. We refer to [40] for merging proposed actions for management and orchestration (MANO) operations; [41] for beamforming adaptation and MIMO systems; [42] for user-level traffic prediction over a short time horizon; [43] for wireless drone networks; [44] for IoT devices learning together; [45] for load balancing; and [46] for handovers in vehicle-to-network communications. Here, we utilize the dynamic version of this tool, combined with online learners, so that optimal CHO and THO decisions can be learned on-the-fly, achieving rigorous theoretical and practical performance guarantees.

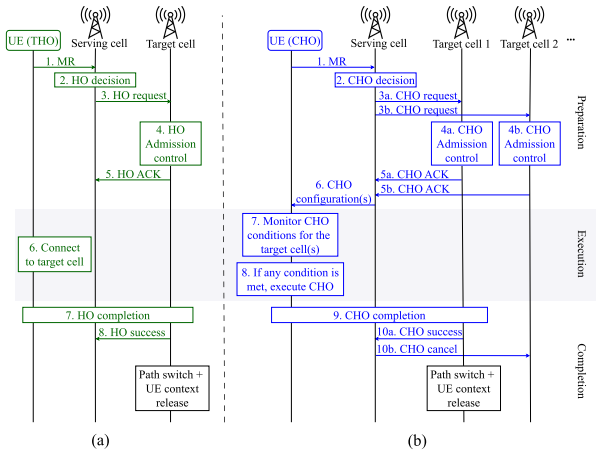


Fig. 2. Basic steps/procedures of UEs performing (a) traditional (green) and (b) conditional (blue) HOs, requiring 8 and 10 main steps, respectively.

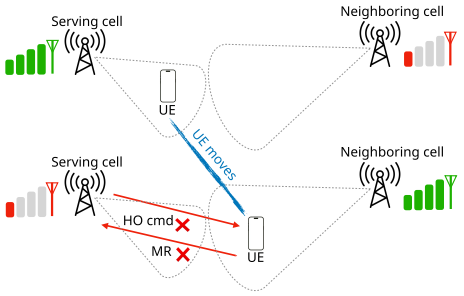


Fig. 3. A failed traditional HO where the MR (HO) command does not reach the serving cell (UE) as the signal drops.

III. Background

A. Traditional HO Basics

THOs are triggered when the signal quality of the serving cell degrades (e.g., A2 event) and/or another neighboring cell's signal becomes higher (e.g., A3 event) [47], [48]. Specifically, the serving cell configures the UE monitoring conditions, such as hysteresis, offset, and time-to-trigger (TTT) parameters, and the UE sends a measurement report (MR) according to these (step 1), as illustrated in Figure 2(a). The *preparation phase* continues with the HO decision and admission (steps 2, 3, 4, 5), which dictate the *one* target cell that the UE should connect to (step 6). This transition (i.e., *execution phase*) is executed *immediately* after the UE receives the command and finishes in the *completion phase*.

This *reactive* approach, where THOs are initiated *after* signal conditions deteriorate, often leads to increased HO delay and HOF rates, particularly due to the small cell density and rapid signal fading of high frequency bands (>24 GHz, FR2)

Authorized licensed use limited to: University of Cyprus. Downloaded on September 28,2026 at 10:36:41 UTC from IEEE Xplore. Restrictions apply.

[3], [49]. As illustrated in Figure 3, it is common for HOFs to occur when a user attempts to send an MR under degrading signal conditions; or even if the MR is successfully sent, the worsening signal conditions may prevent the user from receiving the HO command from the serving cell.

B. Conditional HO Basics

CHOs, designed first as part of 3GPP Rel. 16, address these limitations by *(i)* separating the preparation and execution phases, with their gap potentially being as large as 9–10 s [17], and *(ii)* offloading part of the HO decision-making to the user before signal conditions deteriorate [1]. As can be seen in Figure 2(b), a CHO decision is taken while signal quality is still adequate (step 2), and the source cell can pre-configure (i.e., prepare) multiple *candidate* target cells (steps 3, 4, 5) based on the MR of the UE (step 1). To conclude the preparation phase, the source cell provides the monitoring conditions (step 6), and the execution phase starts: the user applies these conditions continuously (step 7) to evaluate the signal of the source and candidate target cells.

If any of the predefined conditions are met (step 8), the UE executes the stored HO command (as if it had just been received) without an MR or reply from the serving cell. If more than one cell meets the execution condition, the UE decides which to access; typically, the one with the highest SINR. CHO is finalized in the completion phase, where resources are released from the serving cell and a new path to the target cell is established (steps 9, 10, and beyond). Once resources are reserved for a UE through admission control (step 4), they are not released until a cancellation is sent (step 10b).

C. Conditional HO Key Trade-Offs

Determining which cells to prepare is critical in CHOs for balancing resource efficiency and mobility robustness. Ideally, the MNO, whose goal is to economize cell resources, would allow the preparation of the single cell to which the UE will connect in the next slot (i.e., the "correct" target cell) and whose signal strength is high, hence maximizing the UE's throughput. However, this cannot always be predicted and multiple candidate cells often need to be prepared, e.g., because the user is located at the edge of a cell and its trajectory is unknown. At the same time, a long list of prepared cells does not *necessarily* increase the likelihood of including the "correct" target cell, especially if all prepared cells exhibit low signal quality. Conversely, while preparing fewer cells may save resources, it increases the risk of falling back on traditional HOs; thus, larger HO delays and HOF probability (see Sec. IV).

Another aspect to consider when optimizing CHOs is the *signaling cost* of preparing cells. For instance, in environments such as FR2, the small cell density and rapid signal fluctuations already result in more frequent HOs, leading to significant signaling overhead. The additional burden of continuously preparing and releasing cells due to constant signal variations further exacerbates this overhead [17], [37]. For that reason, although initial 3GPP releases specify that UEs should release CHO candidate cells after any successful HO completion to conserve resources [1], subsequent studies suggest that this approach is not always optimal for minimizing HOFs and signaling overhead [49].

These trade-offs emphasize that the selection process for cell preparations should focus on finding a balance between identifying the "best" (in terms of signal quality) potential target cells and keeping the number of prepared cells small (i.e., limited reserved resources).

D. Granularity

An essential aspect of any intelligent mechanism is the granularity at which THO/CHO decisions are made (step 2 of Figure 2). Typically, these decisions are taken in hundreds of milliseconds (e.g., 100–640 ms), driven by the TTT parameter; namely, the duration for which the signal of a neighboring cell should be constantly better than that of the serving cell [18], [25], [47], [48]. This granularity aligns with the decisions of a central controller in the near-RT of O-RAN, which could even handle multiple cells simultaneously [50].

IV. Dataset & Preliminary Analysis

To motivate the benefits of CHOs, we showcase the problems with traditional HOs using data collected from an MNO in a European country. We analyze measurements for one week, from 24-June-2024 to 30-June-2024, and focus on HO events in the most recent radio access generations: 4G and 5G. At the time this study was conducted, CHOs were unavailable in the analyzed network and the majority of 5G traffic was served through 5G-Non-Standalone (NSA), i.e., relying on the 4G Evolved Packet Core (EPC) instead of the newly deployed 5G core. Our goal is to identify the features (e.g., vendor, location) in the source and target cells, as well as in the UEs, that contribute to increased HOFs and delays. To complement our analysis, we additionally leverage: (i) crowdsourced datasets of signal quality metrics per second (e.g., SINR and RSRP) of 13,590 cells (3,496 cell sites), with precise measurement locations to characterize signal fluctuations; and (ii) census data to examine the correlation of population density with HOs and cell deployments.

We analyze mobility management from the perspective of *both source and target* cells at the country-level. This overcomes limitations of prior works that relied only on UE-side measurement datasets, such as [21], [51], as well as other large network-side works that focus either only on measurements [3], or only on the source cell characteristics, omitting the impact of target cell features apart from the change in radio access technology [27].

A. Signal Quality Metrics and Mobility

In Figure 4, we examine temporal signal variations, namely SINR and RSRP, across various speeds using our crowdsourced datasets, which offer measurements at 1-second intervals. We classify samples into three mobility categories: *static*, *medium* speed (3–10 km/h, likely walking/running), and *high* speed movements (>80 km/h, likely vehicular). In general, SINR is sensitive to short-term effects such as fast fading, interference, and noise floor variations, while RSRP reflects a more stable signal strength from the serving cell.

Authorized licensed use limited to: University of Cyprus. Downloaded on September 28,2026 at 10:36:41 UTC from IEEE Xplore. Restrictions apply.

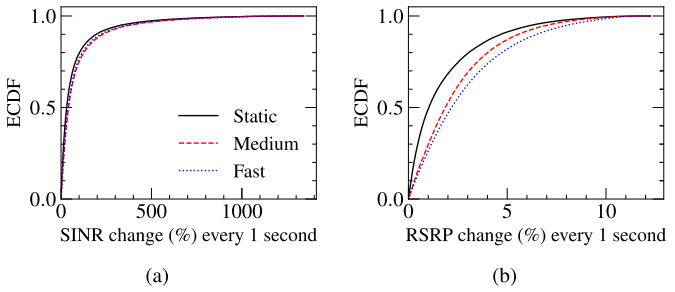


Fig. 4. (a) SINR and (b) RSRP changes per second.

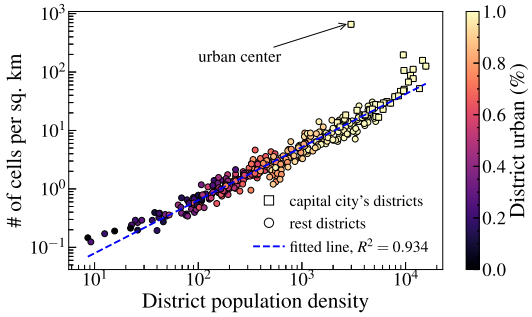


Fig. 5. Number of cells per sq. km in the country (district level). Each point is colored according to the ratio of postcodes classified as urban (the others are rural).

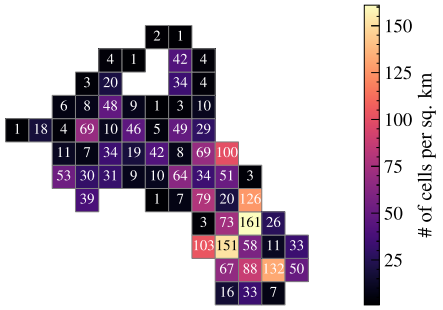


Fig. 6. Number of cells/sq.km in an urban area with 2.2k cells.

Our findings reveal that SINR exhibits considerable fluctuations, even at the second-level granularity, with more than 20% of movements experiencing a 100% or greater change in SINR in one second, regardless of speed. Given that our passive measurement dataset covers cell deployments in the frequency bands below 3.5 GHz, we highlight that SINR variability does not increase with mobility due to, for instance, greater Doppler spread and more dynamic interference conditions. Doppler effects typically become relevant at higher frequencies, such as mmWave and above [52], and become more noticeable in very high-speed scenarios, such as trains traveling at speeds of up to 500 km/h and Vehicle-to-Vehicle (V2V) communications [52], [53]. In contrast, our high-mobility tier mainly captures vehicular UEs moving at speeds above 80 km/h (and rarely exceeding 130 km/h) in an urban environment. On the other hand, RSRP is more stable, and speed plays a bigger role; e.g., 20% medium-speed movements have their RSRP changing more than 4%.

It is worth recalling that in HO procedures, a typical threshold for initiating the HO based on the A3 event is commonly set around 3 dB [25]. This threshold provides a useful reference when interpreting Figure 4, as it highlights how frequently such conditions can be met.

Key Takeaways: The SINR fluctuates by more than 100% for over 20% of UEs within short time intervals of 1 s. Such rapid signal variations increase the likelihood of suboptimal HO triggering, and thus, of HO delays and HOFs. The proactive cell preparation approach of CHOs can mitigate the impact of such fluctuations and improve user performance.

B. Spatial Heterogeneity

From the official census data of the analyzed country, we dissect the 300 + defined districts and 2.5M+ postcodes, and combine them with our collected cell-level datasets from the MNO to understand how cells are placed across different environments, thus affecting the number of cells that can be prepared in CHOs. Postcodes have been classified as *urban* or *rural*, based on whether they are allocated to an area with a population of more or less than 10k residents, respectively.

Figure 5 displays the district-level cell density (number of cells per sq. km) compared to the population density. Each point corresponds to one of the 300 + districts in the studied country and is color-coded based on the percentage of postcodes classified as urban within the district. As expected, there is a high correlation between these two factors (Pearson correlation of 0.967). In districts where more than 90% of postcodes are urban, shown mainly in yellow, we observe 10 to 100 cells per sq. km, with areas near the capital city exceeding this range. It is interesting to note that the urban center of the capital city (highest yellow rectangle) attracts a substantial influx of non-resident users due to being a major administrative and economic hub; consequently, operators deploy more than 650 cells per sq. km to accommodate the increased demand. On the other hand, areas with less than 20% urban postcodes exhibit as few as 0.12 cells per sq. km, which is sufficient to meet the reduced user demand there.

In Figure 6, we further delve into a dense urban district to illustrate the spatial heterogeneity of the cell deployment. We partition the district into one sq. km tiles, revealing 2.2k+ total deployed cells. The per-tile cell count spans over two orders of magnitude, ranging from a single cell in sparsely covered areas to more than 160 per sq. km in highly saturated hotspots. Notably, some adjacent tiles exhibit substantial disparities, with cell counts increasing from as few as 3 cells in one tile to as many as 103 in a neighboring one. In CHOs, this means that a UE traversing these areas must dynamically account for a highly variable and often large set of candidate cells.

Key Takeaways: Cellular network deployments are highly heterogeneous in terms of density, which is correlated with population. In our dataset, there are 650 + cells per sq. km in the urban center of the capital city, while in remote areas, there are as few as 0.12 per sq. km. In urban areas, the number of candidate cells for CHO can be very large and depends on cell density. There are even significant differences in cell counts between adjacent tiles (1 sq. km). This means that the signaling cost for preparing target cells is not uniform across the network, as it depends on the cell deployment density

Authorized licensed use limited to: University of Cyprus. Downloaded on September 28,2026 at 10:36:41 UTC from IEEE Xplore. Restrictions apply.

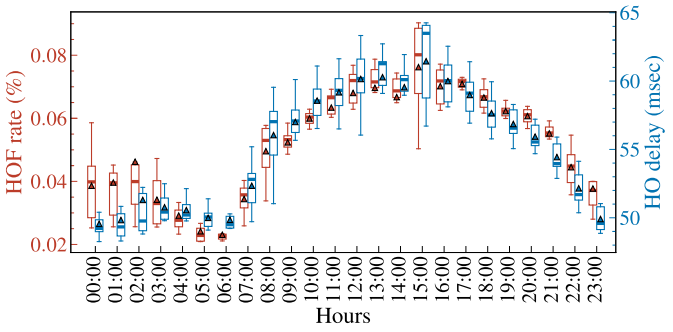


Fig. 7. HOF rate (left, red y-axis) and delay of successful HOs (right, blue y-axis) per hour for a 1-week period. Triangles in boxplots depict the mean.

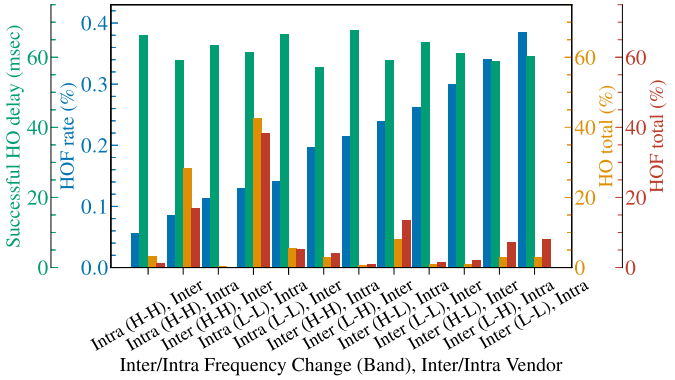


Fig. 8. (left to right) Delay of successful HOs (ms), HOF rate (%), percentage of total HOs, and HOFs based on the source-target frequency bands and vendors.

and properties (distance, frequency, etc). Therefore, the system model in Sec. V considers cell-dependent signaling costs.

C. HO Failures & Delays

We analyze factors that increase HOFs and delays, thereby highlighting the scenarios in which CHOs offer clear benefits.

Figure 7 presents the hourly evolution of successful HO delays (in ms) and HOF rate (in %) for 1 week in the entire country. Boxplots aggregate cell-level data from the same hour. The two metrics exhibit a diurnal pattern, reaching a max of 0.08% HOF rate and 62 ms latencies (median values) at 15:00, which is the hour when most HOs occur and the network is most congested [3]. Notably, from the pre-dawn median minimum (06:00) to the max peak at 15:00, there is an increase of 252% in the HOF rate; at the same time, HO delay increases by 28%. Also, during the night hours 00:00–06:00, even though the HOF rate decreases significantly, the delay of successful HOs increases by approximately 3 ms. This is partially due to the MNO applying dynamic energy-saving policies that switch off cells acting as capacity boosters when they are not needed to meet the demand [54]. Clearly, existing HO mechanisms offer room for improvement, especially during peak hours, to reduce HOF rates and delays, which can be achieved by leveraging a dynamic and robust THO/CHO algorithm.

We further analyze which features of the source and target cells mainly affect the HOF rate. For that, we focus on source-target cell pairs with at least one HOF registered per day (excluding approximately 3.4% that are outliers), and study the effect of: (i) vendor, (ii) frequency band, (iii) model name, (iv) transmit power, (v) cell type (macro, micro, etc.), (vi) region (north, south, etc.) and (vii) area type (urban/rural).

We start this analysis with a non-parametric Kruskal-Wallis test, which confirms that all examined factors affect the HOF rate ($p < 0.05$).[2] To determine the importance of each feature, we run multiple Machine Learning (ML) models. We discard features with high multicollinearity (VIF >3) and Pearson correlation (>0.5), and include the number of HOs as an input feature to account for spatio-temporal variations (Figures 6 and 7). The HOF rate is log-transformed, given its heavy-tailed distribution. Among the evaluated models (linear/lasso/ridge regression, k-nearest neighbours, and random forest), the random-forest regressor achieved the best performance, reaching an R^2 of 0.8 (compared to at most 0.58 for the other models) and RMSE of 0.66 (compared to at most 0.97 for the other models). As a reference, a naive model that always predicts, in the test set, the mean HOF rate from the training set, achieves an RMSE of 1.47. Permutation-based importance reveals that the number of HOs is by far the strongest predictor, followed by the cell frequency band and the vendor; excluding these last two features drops the performance to $R^2 = 0.6$, confirming that frequency and vendor still contribute meaningfully.

Figure 8 shows the effect of these latter features on HOs and HOFs. The x-axis indicates whether the HO occurred in the same (intra) or different (inter) frequency and antenna vendor, as well as if the frequencies of the source and target cells lie in the low ($\leq$ 2GHz) or mid-high (>2GHz) spectrum. The y-axis shows the HOF rate in ascending order, the delay of successful HOs and the percentage of total HO and HOF count within each category. We observe that the mean HOF rate in inter-frequency HOs, mainly when the source and/or target cells operate in low frequencies, is $\approx 0.26\%$, i.e., $\times 2.4$ more than the 0.11% of the intra-frequency ones. However, the HO delay for the former is no more than 1.5 ms higher. Intra-frequency HOs in the same low-spectrum frequency handle the majority of HOs (i.e., 42%), but their contribution to failures closely matches this number (38.4% of all HOFs). Lastly, note that HO delay is mainly affected by the change in vendor: even though 88% of HOs are intra-vendor with a delay of 59–60 ms, inter-vendor HOs exhibit a delay that is 5–6 ms higher. These inter-vendor HOs are prevalent at regional borders (e.g., west-east border), given that a single vendor predominantly serves each region in the studied country. This analysis reveals that accounting for cell features is crucial for HOs.

Key Takeaways: HO failures and delays are strongly influenced by: (i) time of day, exhibiting diurnal patterns that peak at 15:00 with a 0.08% HOF rate and a 62 ms HO delay (median values); (ii) frequency changes between the source and target cells, with inter-frequency HOs having $\times 2.4$ higher HOFs than intra-frequency HOs, although with a slight difference in HO delay, and (iii) vendor transitions, where inter-vendor HOs exhibit 5–6 ms higher delay than intra-vendor. This evidences the potential benefit of accounting for

[2]Normality and homoscedasticity assumptions are violated; hence, analysis of variance (ANOVA) [55] was not tested.

Authorized licensed use limited to: University of Cyprus. Downloaded on September 28,2026 at 10:36:41 UTC from IEEE Xplore. Restrictions apply.

the previous cell-level characteristics for HO optimization, as we consider in the design of our algorithm.

V. System Model & Problem Formulation

A. Modeling Components

We consider a heterogeneous cellular network comprising a set $\mathcal{J}$ of J cells that serve a set $\mathcal{I}$ of I UEs. We assume that a central controller takes decisions for multiple UEs/cells [11] in a time-slotted manner for a set $\mathcal{T}$ of T slots [22], [56] with each slot lasting hundreds of milliseconds (see Sec. III), as enabled by near-RT RIC in O-RAN [50].[3]

These decisions concern both THOs and CHOs, and are taken on the *same time-scale*. The set $\mathcal{I}$ is partitioned into the set $\mathcal{I}_{\text{THO}}$ of I_{THO} UEs and $\mathcal{I}_{\text{CHO}}$ of I_{CHO} UEs, who follow THOs (*THO-enabled*) and CHOs (*CHO-enabled*), respectively, where $\mathcal{I}_{\text{THO}} \cap \mathcal{I}_{\text{CHO}} = \emptyset$ and $\mathcal{I}_{\text{THO}} \cup \mathcal{I}_{\text{CHO}} = \mathcal{I}$. This distinction can arise due to service-level agreements (SLAs), network slicing policies [57], or vertical-specific requirements, in which different classes of UEs are managed separately; e.g., enhanced-reliability or low-latency UEs may be assigned to CHO. Alternatively, an MNO may apply a rule-based classification, informed by prior data (e.g., historical patterns), to assign UEs to either *HO type*.

For the THO-enabled UEs, we introduce the decisions:

$$\boldsymbol{x}_t = \big(x_{ij}(t) \in \{0,1\}, i \in \mathcal{I}_{\text{THO}},\ j \in \mathcal{J}\big),$$

where $x_{ij}(t) \in \{0,1\}$ defines the *explicit one-cell association decision* with $x_{ij}(t) = 1$ if user i is assigned from the controller to cell j in slot t. For the CHO-enabled UEs, we introduce:

$$\boldsymbol{y}_t = \big(y_{ij}(t) \in \{0,1\}, i \in \mathcal{I}_{\text{CHO}},\ j \in \mathcal{J}\big),$$

where $y_{ij}(t) \in \{0,1\}$ is the *preparation decision* (*implicit multi-cell association decision*): $y_{ij}(t) = 1$ means that cell j is prepared from the controller in slot t for user i and the user can decide whether to connect to this cell (as explained later). These decisions are drawn from the sets:

$$\mathcal{X} = \left\{ \boldsymbol{x} \in \{0,1\}^{I_{\text{THO}} \cdot J} \,\middle|\, \sum_{j \in \mathcal{J}} x_{ij} = 1, i \in \mathcal{I}_{\text{THO}} \right\},$$

where naturally each $i \in \mathcal{I}_{\text{THO}}$ is assigned to one cell, and:

$$\mathcal{Y} = \left\{ \boldsymbol{y} \in \{0,1\}^{I_{\text{CHO}} \cdot J} \,\middle|\, 1 \le \sum_{j \in \mathcal{J}} y_{ij} \le b_i, i \in \mathcal{I}_{\text{CHO}} \right\},$$

where no more than b_i cells can be prepared for $i \in \mathcal{I}_{\text{CHO}}$.

The key metric is the SINR for the signal delivered by cell $j \in \mathcal{J}$ to any user $i \in \mathcal{I}$ in slot $t \in \mathcal{T}$:

$$s_{ij}(t) = \frac{q_j \phi_{ij}(t)}{W_j \sigma^2 + \sum_{k \in \mathcal{B}_j} q_k \phi_{ik}(t)}, \tag{1}$$

where q_j is the transmit power of cell j, $\mathcal{B}_j$ the set of cells that operate in the same frequency as j, $\phi_{ij}(t)$ the channel gain (including pathloss, shadowing, and antenna gains), W_j is the bandwidth, and σ^2 the power spectral density. In line with previous works [22], [23], [24], [25], $s_{ij}(t)$ is the average SINR in t, since UEs report multiple values during each slot.

We distinguish two cases for the *maximum throughput* a UE can receive from a cell, based on whether it is THO- or CHO-enabled: For the former [22], [23], [27], this throughput is expressed as:

$$c_{ij}(t) = W_j \log\big(1 + s_{ij}(t)\big), i \in \mathcal{I}_{\text{THO}}, \tag{2}$$

while for the latter, we introduce:

$$c'_{ij}(t) = \begin{cases} W_j \log\big(1 + s_{ij}(t)\big), & \text{if } j = \underset{k:\, y_{ik}(t)=1}{\arg\max}\ s_{ik}(t) \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

where $i \in \mathcal{I}_{\text{CHO}}$. The throughput of the CHO-enabled UEs in eq. (3) depends on their own decision, which, in turn, depends on the measured signals and the list of prepared cells it receives from the network. We consider here the standard cell-selection rule, in which the CHO-enabled UE connects to the best-SINR cell among those prepared for it.[4]

Finally, we define the load of each cell $j \in \mathcal{J}$:

$$\ell_j(t) = \sum_{i \in \mathcal{I}_{\text{THO}}} x_{ij}(t) + \sum_{i \in \mathcal{I}_{\text{CHO}}} y_{ij}(t) \le C_j, \tag{4}$$

as the total number of associated users (both THO- and CHO-enabled), which cannot exceed the upper bound C_j, in lieu of the actual spectrum budget capacity. Note that for the CHO-enabled users, the cell resources are reserved independently of whether the UEs will actually select the cell, and hence CHOs are likely to induce unnecessary resource waste. This is a key CHO issue that our model tackles.

To streamline the presentation, we define $\boldsymbol{z}_t = (\boldsymbol{x}_t, \boldsymbol{y}_t)$ and the associated set:

$$\mathcal{Z} = \Big\{ \boldsymbol{z} = (\boldsymbol{x}, \boldsymbol{y}) \mid \ell_j \le C_j,\ \ \forall j \in \mathcal{J},\ \boldsymbol{x} \in \mathcal{X},\ \boldsymbol{y} \in \mathcal{Y} \Big\}.$$

B. Problem Statement

In our framework, the network controller makes *joint* decisions for the THO- and CHO-enabled users, whose performance is coupled through the load of each cell, as can be seen in eq. (4). With this in mind, we introduce the *utility* function the controller needs to maximize:

$$g_t(\boldsymbol{z}_t) = \sum_{j \in \mathcal{J}} \Bigg(\underbrace{\sum_{i \in \mathcal{I}_{\text{THO}}} x_{ij}(t) \log \frac{c_{ij}(t)}{\ell_j(t)}}_{\substack{\text{explicitly assigned UEs} \\ \text{(can do THO)}}} + \underbrace{\sum_{i \in \mathcal{I}_{\text{CHO}}} y_{ij}(t) \log \frac{c'_{ij}(t)}{\ell_j(t)}}_{\substack{\text{implicitly assigned UEs} \\ \text{(can do CHO)}}} \Bigg) \tag{5}$$

where the two terms define the throughput of the THO-enabled and CHO-enabled UEs.[5] We assume the cell resources are

[3]The modeling and optimization framework of this section can be extended and implemented by other RAN systems, not solely O-RAN, as long as there is support for centralized decision-making that is fed by signal measurements.

[4]To avoid often CHOs and ping-pong effects [58] when finding the highest-SINR prepared cell, it is possible to subtract a cell-specific offset, as happens in the A3 event [18], [47], [48].

[5]We consider that there is no idle cell, and in case of zero throughput, $c_{ij}(t) + 1$ or $c'_{ij}(t) + 1$ can be used inside the logarithm.

Authorized licensed use limited to: University of Cyprus. Downloaded on September 28,2026 at 10:36:41 UTC from IEEE Xplore. Restrictions apply.

allocated fairly across the users via, e.g., a round robin or a proportional-fair scheduler [59]. The logarithmic transformation balances throughput across users to achieve fairness [22]; however, other mappings or schedulers can be used as easily.

Using the association decision $x_{ij}(t)$ for user $i \in \mathcal{I}_{\text{THO}}$, the THO is modeled with the change $x_{ij}(t) \neq x_{ij}(t-1)$. This way, the total number of THOs can be captured with the norm $\|\boldsymbol{x}_t - \boldsymbol{x}_{t-1}\|$, as in [22]. Nevertheless, we are interested in the *THO delay (switching) cost* and not merely in the number of THOs. Following our findings in Sec. IV, we propose using the weighted norm $\|\boldsymbol{x}_t - \boldsymbol{x}_{t-1}\|_{A_t}$, where $\boldsymbol{A}_t = \text{diag}(a_n(t) > 0)$ is a positive definite matrix with its diagonal weights $a_n(t) \in [0,1]$, $n = 1, \ldots, I_{\text{THO}} \cdot J$, penalizing differently the THOs for each UE-cell pair; a penalty that may even change over time.[6]

On the other hand, a CHO is executed from the UE based on the prepared cells by the controller captured through $\boldsymbol{y}_t$: the user's decision in the CHO case is to be assigned to the highest-SINR prepared cell. Clearly, frequent cell preparations and releases may lead to increased signaling [4], [18], especially in dense cell deployments (can be up to 161 cells, see Sec. IV). Hence, we introduce the *CHO signaling (switching) cost* $\|\boldsymbol{y}_t - \boldsymbol{y}_{t-1}\|_{B_t}$. In this case, $\boldsymbol{B}_t = \text{diag}(b_{n'}(t) > 0, n' = 1, \ldots, I_{\text{CHO}} \cdot J)$ is a diagonal positive definite matrix with weights $b_{n'}(t) \in [0,1]$, to penalize differently the preparations and releases of cells for each UE-cell pair[7] in slot t. Given that THOs are more prone to failures and delays than CHOs, it holds that $a_n(t) > b_{n'}(t),\ \forall t \in \mathcal{T}$.

These THO and CHO switching costs can be combined as:

$$\|\boldsymbol{z}_t - \boldsymbol{z}_{t-1}\|^2_{C_t} = \|\boldsymbol{x}_t - \boldsymbol{x}_{t-1}\|^2_{A_t} + \|\boldsymbol{y}_t - \boldsymbol{y}_{t-1}\|^2_{B_t}, \quad (6)$$

where $\boldsymbol{C}_t$ is the full block diagonal matrix:

$$\boldsymbol{C}_t = \begin{bmatrix} \boldsymbol{A}_t & 0 \\ 0 & \boldsymbol{B}_t \end{bmatrix}, \quad (7)$$

and the induced norm and its dual [60], are:

$$\|\boldsymbol{x}_t\|^2_{A_t} = \sum_{n=1}^{I_{\text{THO}} \cdot J} a_n(t) x_n(t)^2, \|\boldsymbol{x}_t\|^2_{A_t *} = \sum_{n=1}^{I_{\text{THO}} \cdot J} x_n(t)^2 / a_n(t), \quad (8)$$

and similarly for $\boldsymbol{y}_t$.

Putting the above together, the problem that the network controller wishes to address is the following:

$$\mathbb{P}_1 : \max_{\{\boldsymbol{z}_t\}_t} \sum_{t=1}^{T} \left(g_t(\boldsymbol{z}_t) - \|\boldsymbol{z}_t - \boldsymbol{z}_{t-1}\|_{C_t} \right)$$
$$\text{s.t.} \quad \boldsymbol{z}_t \in \mathcal{Z}, \quad \forall t \in \mathcal{T},$$

where $g_t(\boldsymbol{z}_t) - \|\boldsymbol{z}_t - \boldsymbol{z}_{t-1}\|_{C_t}$ is the *objective* function. Next, we explain in detail why $\mathbb{P}_1$ cannot be solved offline.

C. Optimization Challenges

Solving $\mathbb{P}_1$ offline is impossible since, at $t = 1$, the controller lacks knowledge of the future SINRs of UEs as it has no access to, or influence over, their mobility. Thus, $\mathbb{P}_1$ must be tackled in an online fashion. In fact, in increasingly-many scenarios, the problem parameters (SINRs, loads, switching costs, etc.) are unknown even at the beginning of each slot; they are only revealed *after* the association and preparation decisions are made. For example, while it may seem the controller could observe the current SINRs before making a decision, in practice, there is a non-negligible delay between the time a UE measures and reports these values and the time the network processes this information. And for fast-moving users or highly-volatile environments, this delay will yield outdated information. This means that we need an online *learning* approach, where both the explicit (for THO) and implicit (for CHO) association decisions are made based on historical UE, network, and environment data, without presuming knowledge of their future values.

What is more, the signaling and switching overheads depend on successive decision changes: a user that remains in acell or a cell that remains prepared does not incur additional costs. This introduces a memory effect in the optimization, as past decisions influence current decisions. We also stress that the A_t-norm and B_t-norm imply the costs change over time; however, this cost does not depend on previous decisions. Finally, $\mathbb{P}_1$ is further compounded by the discreteness of the variables $\boldsymbol{x}_t$ and $\boldsymbol{y}_t$ which prevents the application of off-the-shelf online convex optimization techniques.

Despite these challenges, the goal is to design an online learning algorithm that determines UE-cell associations and preparations and is oblivious to all time-varying and unknown *system conditions*, including future SINR, load, switching costs, and HO delays for each UE-cell pair.

VI. Algorithm Design

To leverage the OCO/SOL toolbox for tackling the problem at hand, the following two main issues need to be addressed: *(i)* the discreteness of decision variables, and *(ii)* the (convex) max operator in the definition of the throughput concerning the CHO-enabled UEs, as can be seen in eqs. (3) and (5). In a nutshell, our solution strategy involves relaxing the discreteness of the decision variables, transforming the resulting continuous problem into a concave one, solving the new concave problem, and then mapping the obtained continuous solution back to the discrete domain via careful rounding.

A. Relaxation & Transformation

First, we define the convex hulls $\mathcal{X}^{\text{c}} = \text{co}(\mathcal{X})$, $\mathcal{Y}^{\text{c}} = \text{co}(\mathcal{Y})$, $\mathcal{Z}^{\text{c}} = \text{co}(\mathcal{Z})$ that relax the integrality of the decision variables:

$$\mathcal{X}^{\text{c}} = \left\{ \boldsymbol{x} \in [0,1]^{I_{\text{THO}} \cdot J} \;\middle|\; \sum_{j \in \mathcal{J}} x_{ij} = 1, i \in \mathcal{I}_{\text{THO}} \right\},$$

$$\mathcal{Y}^{\text{c}} = \left\{ \boldsymbol{y} \in [0,1]^{I_{\text{CHO}} \cdot J} \;\middle|\; 1 \leq \sum_{j \in \mathcal{J}} y_{ij} \leq b_i, i \in \mathcal{I}_{\text{CHO}} \right\},$$

$$\mathcal{Z}^{\text{c}} = \left\{ \boldsymbol{z} = (\boldsymbol{x}, \boldsymbol{y}) \mid \ell_j \leq C_j, \quad \forall j \in \mathcal{J},\ \boldsymbol{x} \in \mathcal{X}^{\text{c}},\ \boldsymbol{y} \in \mathcal{Y}^{\text{c}} \right\}.$$

[6] E.g., 3G UEs have higher cost for changing cells as they are more prone to HO failures [27]. In this way, the model accounts for the UE's HO capabilities.

[7] E.g., the signaling cost might be more detrimental during peak-hours.

Authorized licensed use limited to: University of Cyprus. Downloaded on September 28,2026 at 10:36:41 UTC from IEEE Xplore. Restrictions apply.

Then, using the properties of logarithm, eq. (5) becomes:

$$g_t(\boldsymbol{z}_t) = \underbrace{\sum_{j\in\mathcal{J}}\sum_{i\in\mathcal{I}_{\text{THO}}} x_{ij}(t)\log c_{ij}(t)}_{g_t^{\text{THO}}(\boldsymbol{x}_t)} + \underbrace{\sum_{j\in\mathcal{J}}\sum_{i\in\mathcal{I}_{\text{CHO}}} y_{ij}(t)\log c'_{ij}(t)}_{g_t^{\text{CHO}}(\boldsymbol{y}_t)} - \underbrace{\sum_{j\in\mathcal{J}} \ell_j(t)\log \ell_j(t)}_{g_t^{\text{load}}(\boldsymbol{x}_t,\ \boldsymbol{y}_t)}, \tag{9}$$

where the throughput for the THO-enabled UEs, $g_t^{\text{THO}}(.)$, is linear in $\boldsymbol{x}_t \in \mathcal{X}^{\text{c}}$, and the load penalty, $-g_t^{\text{load}}(.)$, describes the entropy; thus, both are *concave*. On the other hand, the throughput for the CHO-enabled UEs, $g_t^{\text{CHO}}(.)$, is not concave even when $\boldsymbol{y}_t \in \mathcal{Y}^{\text{c}}$, due to the influence of preparation decisions $\boldsymbol{y}_t$ on $c'_{ij}(t)$ through the max operator, see eq. (3). To illustrate this more clearly, we can equivalently write this part as follows:

$$g_t^{\text{CHO}}(\boldsymbol{y}_t) \triangleq \sum_{i\in\mathcal{I}_{\text{CHO}}} \max_{j:\, y_{ij}(t)=1} \log c_{ij}(t), \tag{10}$$

where this transformation achieves the same/desired behavior: the CHO-enabled UE obtains throughput from the cell with the best SINR *among the prepared ones*. Given that the max operator of eq. (10) is nonsmooth and convex, we introduce a linear masking mechanism that leads to a *concave* surrogate, inspired by the standard LogSumExp function [61] used in deep neural networks (DNNs):

$$\tilde{g}_t^{\text{CHO}}(\boldsymbol{y}_t) \triangleq \sum_{i\in\mathcal{I}_{\text{CHO}}} \frac{1}{\alpha}\log\left(\sum_{j\in\mathcal{J}} y_{ij}(t)\, c_{ij}(t)^{\alpha}\right), \tag{11}$$

where $\alpha > 0$ controls the tightness of the approximation.

Lemma 1 (Approximation of CHO Throughput): *For $\boldsymbol{y}_t \in \mathcal{Y}^c$ and $t \in \mathcal{T}$,*

$$\lim_{\alpha\to\infty} \tilde{g}_t^{CHO}(\boldsymbol{y}_t) = g_t^{CHO}(\boldsymbol{y}_t).$$

Proof: Let $d_i^*(t) = \max_{j:\, y_{ij}(t)=1} \log c_{ij}(t)$ and $d_{ij}(t) = \log c_{ij}(t)$. Then, eq. (11) becomes:

$$\tilde{g}_t^{\text{CHO}}(\boldsymbol{y}_t) = \sum_{i\in\mathcal{I}_{\text{CHO}}} \frac{1}{\alpha}\log\left(\sum_{j\in\mathcal{J}} y_{ij}(t) e^{\alpha d_{ij}(t)}\right)$$

$$= \sum_{i\in\mathcal{I}_{\text{CHO}}} \frac{1}{\alpha}\log\left(e^{\alpha d_i^*(t)}\left(1 + \sum_{j:\, d_{ij}(t)\neq d_i^*(t)} y_{ij}(t) e^{\alpha\left(d_{ij}(t)-d_i^*(t)\right)}\right)\right)$$

$$= \sum_{i\in\mathcal{I}_{\text{CHO}}}\left(d_i^*(t) + \frac{1}{\alpha}\log \left(1 + \sum_{j:\, d_{ij}(t)\neq d_i^*(t)} y_{ij}(t) e^{\alpha\left(d_{ij}(t)-d_i^*(t)\right)}\right)\right),$$

which converges to the max, namely, $d_i^*(t)$, since $d_{ij}(t) < d_i^*(t)$ by definition; and then $e^{\alpha(d_{ij}(t)-d_i^*(t))} \to 0$ as $\alpha \to \infty$.□

Given that from Lemma 1, eq. (11) approximates eq. (10), the transformed utility:

$$\tilde{g}_t(\boldsymbol{z}_t) = g_t^{\text{THO}}(\boldsymbol{x}_t) + \tilde{g}_t^{\text{CHO}}(\boldsymbol{y}_t) - g_t^{\text{load}}(\boldsymbol{x}_t, \boldsymbol{y}_t) \tag{12}$$

is concave, as desired.

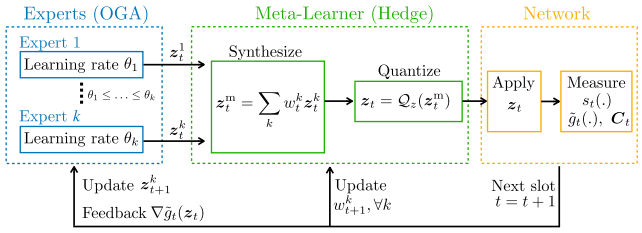


Fig. 9. Learning mechanism. *(i)* Each expert uses a different learning rate and proposes a HO policy to the meta-learner (blue box). *(ii)* The meta-learner combines the proposals, using the experts' weights, and implements the result after discretizing it (green box). *(iii)* The mechanism observes the performance and assesses how good each expert's proposal was (orange box). *(iv)* The experts adapt their decisions and the meta-learner adapts their weights.

B. The Meta-Learning Approach

We approach $\mathbb{P}_1$ as a *smoothed online learning* problem and solve it using a *meta-learning* approach based on the *experts* framework [62], [63], as can be seen Figure 9. This is particularly well-suited to our setting because switching costs are incorporated into the formulation, and system conditions can vary significantly across time slots due to, e.g., unknown user mobility. As a result, deploying a single learner with a fixed learning rate may perform well in some regimes but poorly in others.

We deploy a set $\mathcal{K}$ of K learning agents, called *experts*, or simply *learners*, each with a different learning step/rate $\theta = (\theta_k, k \in \mathcal{K})$ that is applied to online gradient ascent (OGA) [64]. An expert with a larger learning rate puts more emphasis on the latest gradient and hence adapts more quickly to rapid changes in SINRs and signaling costs, making it more suitable for highly volatile scenarios. In contrast, experts with a smaller learning rate update their decisions more carefully (do not move fast from previous decisions) and hence are less affected by parameter fluctuations. This makes them better suited for relatively stationary scenarios as they induce less handovers (decision changes).

Given that the volatility of the environment and UEs' mobility patterns are unknown in advance, we use a large enough set $\mathcal{K}$ of experts, each tuned with a different learning rate. The rationale is that, with careful selection of the rates (number of experts and steps), at least one expert is guaranteed to perform well for the encountered scenario. We identify the best expert for each scenario at runtime by using a meta-learning algorithm, which implements Hedge [65] (and not OGA as the experts) and learns how much weight to put on each expert's proposal. These weights are dynamically updated based on observed expert performance.

Formally, at the beginning of each slot $t = 1, \ldots, T$, each expert $k \in \mathcal{K}$ shares its suggestion $\boldsymbol{z}_t^k = (\boldsymbol{x}_t^k, \boldsymbol{y}_t^k)$, where we initialize $z_0^k = 0$ for all k. The meta-learner combines these suggestions into a single decision $\boldsymbol{z}_t^{\text{m}} = (\boldsymbol{x}_t^{\text{m}}, \boldsymbol{y}_t^{\text{m}})$ as follows:

$$\boldsymbol{z}_t^{\text{m}} = \sum_{k\in\mathcal{K}} w_t^k \boldsymbol{z}_t^k, \tag{13}$$

where $\boldsymbol{w}_t = (w_t^k, k \in \mathcal{K})$, with $\boldsymbol{w}_t^\top \mathbf{1}_K = 1$, are the meta-learner's adaptive weights for each expert. The goal is to assign higher weights to experts that perform better.

Authorized licensed use limited to: University of Cyprus. Downloaded on September 28,2026 at 10:36:41 UTC from IEEE Xplore. Restrictions apply.

Since the decision $\boldsymbol{z}_t^{\mathrm{m}}$ of the meta-learner takes continuous values due to $\boldsymbol{w}_t$, we apply a quantization function $Q_{\mathcal{Z}}$ to project it back to a valid (i.e., implementable) discrete decision $\boldsymbol{z}_t$. We require that the quantization is unbiased, i.e., $\mathbb{E}[\boldsymbol{z}_t] = \boldsymbol{z}_t^{\mathrm{m}}$, which holds by: (i) picking a cell j to assign a THO-enabled UE $i \in \mathcal{I}_{\text{THO}}$ with probabilities $\boldsymbol{x}_i^{\mathrm{m}}(t)$, creating $\boldsymbol{x}_t \in \mathcal{X}$, and (ii) deciding whether to prepare a cell j for a CHO-enabled UE $i \in \mathcal{I}_{\text{CHO}}$ through sampling from a Bernoulli distribution with probabilities $\boldsymbol{y}_{ij}^{\mathrm{m}}(t)$, creating $\boldsymbol{y}_t \in \mathcal{Y}$.

Once the decision $\boldsymbol{z}_t$ is implemented, the controller observes the SINR values $s_{ij}(t)$ for all UEs and cells, and calculates $\tilde{g}_t(.)$ through eq. (12), as well as the switching costs (by observing C_t). We pad with zeros the entries $s_{ij}(t)$ for which no SINR is received, because these cells are unreachable. As a next step, the meta-learner evaluates the decision $\boldsymbol{z}_t^k$ of each expert using the surrogate (i.e., partially linearized) loss:

$$l_t(\boldsymbol{z}_t^k) = -\langle \nabla \tilde{g}_t(\boldsymbol{z}_t), \boldsymbol{z}_t^k - \boldsymbol{z}_t \rangle + \|\boldsymbol{z}_t - \boldsymbol{z}_{t-1}\|_{C_t} \tag{14}$$

and updates its weights using the step η (defined later) as:

$$w_{t+1}^k = \frac{w_t^k e^{-\eta l_t(\boldsymbol{z}_t^k)}}{\sum_{k\in\mathcal{K}} w_t^k e^{-\eta l_t(\boldsymbol{z}_t^k)}}. \tag{15}$$

Lastly, it sends the gradient $\nabla g_t(\boldsymbol{z}_t)$ in the implementable decision $\boldsymbol{z}_t$ to all experts, and they update their choices via OGA:

$$\boldsymbol{z}_{t+1}^k = \Pi_{\mathcal{Z}}\Big(\boldsymbol{z}_t^k + \theta_k \nabla \tilde{g}_t(\boldsymbol{z}_t)\Big). \tag{16}$$

The projection $\Pi_{\mathcal{Z}}(.)$ ensures that the proposed decisions $\boldsymbol{z}_{t+1}^k$ lie in the feasible space, namely, $\boldsymbol{z}_{t+1}^k \in \mathcal{Z}$. A summary of the steps can be seen in Algorithm 1 (CONTRA).

Algorithm 1 CONditional-TRAditional HOs (CONTRA)

Required: Step η for meta-learner and $\{\theta_k\}_{k\in\mathcal{K}}$ for experts
Initialize: Sort $\theta_1 \le \theta_2 \le \ldots \le \theta_K$ and set $w_1^k = \frac{1+1/K}{k(k+1)}$
for $t = 1, 2, \ldots, T$ **do**
  Each expert $k \in \mathcal{K}$ shares its decision $\boldsymbol{z}_t^k$
  Combine all decisions $\boldsymbol{z}_t^k$ into $\boldsymbol{z}_t^{\mathrm{m}}$ using (13)
  Create an implementable discrete decision $\boldsymbol{z}_t = Q_{\mathcal{Z}}(\boldsymbol{z}_t^{\mathrm{m}})$
  Observe SINRs, $\tilde{g}_t(.)$ and switching costs C_t
  Update the weights of experts using (14) and (15)
  Send $\nabla g_t(\boldsymbol{z}_t)$ to each expert
  Each expert updates its decision using (16)
end

C. Performance Guarantees & Complexity

To assess the performance of our algorithm, we compare it against a powerful benchmark (i.e., *oracle*) that has full a priori knowledge of the system conditions (i.e., SINR, load, costs etc) and can choose the best possible sequence of decisions over the entire horizon to maximize the cumulative objective of $\mathbb{P}_1$. This serves as a highly competitive benchmark, exceeding both static (single best solution for all time slots) and dynamic ones (best solution for each time slot independently) [19], [66].

For that, we leverage the *Expected Dynamic Regret* [20]:

$$\mathbb{E}\left[\mathcal{R}_T\right] \triangleq \sum_{t=1}^{T}\Big(\tilde{g}_t(\boldsymbol{z}_t^\star) - \|\boldsymbol{z}_t^\star - \boldsymbol{z}_{t-1}^\star\|_{C_t}\Big) - \sum_{t=1}^{T}\mathbb{E}\Big[\tilde{g}_t(\boldsymbol{z}_t) - \|\boldsymbol{z}_t - \boldsymbol{z}_{t-1}\|_{C_t}\Big], \tag{17}$$

where $\{\boldsymbol{z}_t\}_t$ and $\{\boldsymbol{z}_t^\star\}_t$ are the algorithm's and powerful-oracle's decisions, respectively. The latter is the solution of $\mathbb{P}_1$, and the expectation captures the randomization in CONTRA's decisions due to quantization $Q_{\mathcal{Z}}$. Our goal is to design an algorithm that ensures this gap diminishes with time, $\lim_{T\to\infty}\mathbb{E}[\mathcal{R}_T]/T = 0$ for any possible benchmark sequence $\{\boldsymbol{z}_t^\star\}_t$. It is important to highlight that achieving close-to-zero expected dynamic regret is particularly challenging.

Ideally, designing an effective algorithm in such settings requires prior knowledge of how much optimal decisions can vary over time, commonly characterized by the *path length*:

$$P_T = \sum_{t=1}^{T}\|\boldsymbol{z}_t^\star - \boldsymbol{z}_{t-1}^\star\|_{C_t}. \tag{18}$$

However, this information is not available in practice. To overcome this limitation, we have relied on a meta-learning algorithm that adaptively learns from a pool of experts with different learning rates, each tailored to perform well under different system conditions. To bound the expected dynamic regret, we first prove the following lemma.

Lemma 2 (Bound of Domain and Gradients): *By considering:* $a_n(t) \le a_{\max}$, $b_{n'}(t) \le b_{\max}$, $c_n'(t) \le c_n(t) \le c_{\max}$ *and* $M = \max\{\log c_{\max} - 1,\ \log I + 1\}$, *with* $t \in \mathcal{T}$, *it holds that:*

- $\|\boldsymbol{z} - \boldsymbol{z}'\|_2 \le \sqrt{2\,I_{THO}} + \sqrt{I_{CHO}\,(J-1)} \triangleq D$,
- $\|\boldsymbol{z} - \boldsymbol{z}'\|_{C_t} \le \sqrt{2\,I_{THO}\ a_{\max}} + \sqrt{I_{CHO}\,(J-1)\ b_{\max}} \triangleq D_C$,
- $\|\boldsymbol{z} - \boldsymbol{z}'\|_{C_t*} \le \sqrt{2\,I_{THO}/a_{\max}} + \sqrt{I_{CHO}\,(J-1)/b_{\max}} \triangleq D_{C^*}$,
- $\|\nabla\tilde{g}_t(\boldsymbol{z})\|_2 \le M\sqrt{I\,J} \triangleq G$,
- $\|\nabla\tilde{g}_t(\boldsymbol{z})\|_{C_t} \le M\sqrt{(I_{THO}\ a_{\max} + I_{CHO}\ b_{\max})\,J} \triangleq G_C$.

Proof: We provide the full proof in the Appendix. □

The main result for the performance guarantee of CONTRA is presented in the sequel.

Lemma 3 (Performance Analysis/Optimality Guarantee): *Using the parameters:*

- $K = \left\lceil \log_2\sqrt{1+2T} \right\rceil + 1$, (19)
- $\theta_k = 2^{k-1}\sqrt{\dfrac{D_C^2}{T(G^2+2G_C)}},\ k = 1, \ldots, K$, (20)
- $\eta = 1/\sqrt{T\nu}$ *with* (21)
  $\nu \triangleq (D_C + 1/8)(GD + 2D_C)^2$, (22)

the discrete decisions $\{\boldsymbol{z}_t\}_t$ *of our algorithm* CONTRA *ensure:*

$$\mathbb{E}\left[\mathcal{R}_T\right] \le \sqrt{T}\Big(\sqrt{\nu}\left(1 + \ln(1/w_1^k)\right) + \sqrt{(G^2+2G_C)(D_C^2 + 2D_{C*}P_T)}\ \Big)$$

Authorized licensed use limited to: University of Cyprus. Downloaded on September 28,2026 at 10:36:41 UTC from IEEE Xplore. Restrictions apply.

$$+T\left(G+\sqrt{a_{\max}+b_{\max}}\right)\sqrt{I_{CHO}J/4+I_{THO}(1-1/J)}, \tag{23}$$

Proof: We provide the full proof in the Appendix. □

With Lemma 3, we bound the expected regret of the implementable decisions, in contrast to previous works [22], [25]. Even though the continuous preparation variables achieve sublinear dynamic regret, the discretization introduces an unavoidable non-diminishing error. As shown in Sec. VIII, this error is small in practical scenarios, and the algorithm converges towards the optimal solutions.

Computational Complexity: The primary source of complexity in our algorithm arises from the projection step performed by each learner $k \in \mathcal{K}$ during OGA, as shown in eq. (16); all other computations are executed in constant time $\mathcal{O}(1)$. The projection difficulty depends on *(i)* the structure of the feasible set, which essentially depends on the constraints of our problem, and *(ii)* the size of the problem, which in practice corresponds to the area monitored by each controller (i.e., the number of UEs and cells). We note that for convex sets in general, there might be no closed-form solution, with the projection requiring $\mathcal{O}(d^3)$, where d is the set's dimension.

Our current implementation indeed computes the projections directly by solving the concave subproblems, which have been found sufficient for the studied problems. Specifically, and despite relying on a general-purpose solver rather than specialized projection routines and a modest hardware (default) setup of a MacBook Pro equipped with an Apple M1 chip (8-core CPU), the per-slot inference (execution) time remains below the near-RT threshold required in O-RAN operation when up to 150 UEs and 25 cells are considered.

More precisely, the inference time increases from 18 ms ($I = 1$ user) to 62 ms ($I = 150$ users) when $J = 5$ cells. For $J = 15$ ($J = 25$), it begins near 18 ms (18 ms) for $I = 1$ user and reaches up to 384 ms (1 s) for $I = 150$ users. It is important to note that the exact runtime may vary depending on the specifications of different machines. For larger search spaces, one can employ tailored projection algorithms, such as [67] and [68], and/or execute the algorithm on GPUs or high-performance computing servers.

VII. Dynamic Handover Type Selection

A. Problem Formulation

In contrast to the framework of Sec. V, we now consider a more flexible and dynamic formulation in which users are not a priori assigned to a specific HO type. Instead, the network controller can decide on-the-fly (i.e., at each time slot t), whether a UE should be THO- or CHO-enabled, based on current system conditions and performance trade-offs, enabling a more responsive and efficient HO strategy, as envisioned by 6G and NextG [9]. Specifically, we allow each UE to be in either HO mode and decide this dynamically: when a UE moves fast, it may benefit from CHO due to the rapid signal fluctuations that lead to higher THO failures and delays, while it might find THO suitable when it moves slower.

In this case, the controller problem becomes:

$$\mathbb{P}_2: \max_{\{\boldsymbol{z}_t\}_t} \sum_{t=1}^{T} \left(g_t(\boldsymbol{z}_t) - \|\boldsymbol{z}_t - \boldsymbol{z}_{t-1}\|_{C_t}\right)$$

$$\text{s.t.} \quad \sum_{j\in\mathcal{J}} x_{ij}(t) \le 1, \quad \forall i \in \mathcal{I}, t \in \mathcal{T} \tag{24}$$

$$y_{ik}(t) \le 1 - \sum_{j\in\mathcal{J}} x_{ij}(t), \ \forall i \in \mathcal{I}, k \in \mathcal{J}, t \in \mathcal{T} \tag{25}$$

$$\sum_{j\in\mathcal{J}} y_{ij}(t) \le b_i, \ \forall i \in \mathcal{I}, t \in \mathcal{T} \tag{26}$$

$$\sum_{j\in\mathcal{J}} x_{ij}(t) + \sum_{j\in\mathcal{J}} y_{ij}(t) \ge 1, \quad \forall i \in \mathcal{I}, t \in \mathcal{T} \tag{27}$$

$$\sum_{i\in\mathcal{I}} x_{ij}(t) + \sum_{i\in\mathcal{I}} y_{ij}(t) \le C_j, \ \forall j \in \mathcal{J}, \forall t \in \mathcal{T} \tag{28}$$

$$\boldsymbol{z}_t = (\boldsymbol{x}_t, \boldsymbol{y}_t) \in \{0,1\}^{I\cdot J} \times \{0,1\}^{I\cdot J}, \ \forall t \in \mathcal{T}. \tag{29}$$

Eq. (24) allows a UE $i \in \mathcal{I}$ to be THO-enabled (i.e., explicitly assigned to a cell if $x_{ij}(t) = 1$), or CHO-enabled (i.e., implicitly assigned through CHOs to a cell, otherwise). Eq. (25) complements the previous one, ensuring that a UE can be explicitly or implicitly assigned to a cell, but not both; eq. (26) prevents each CHO-enabled UE from having more than b_i prepared cells, similarly to Sec. V. Eq. (27) ensures that a UE will be explicitly or implicitly assigned, preventing blocking completely some UEs, and eq. (28) enforces each cell's capacity constraints using the total number of users. Lastly, eq. (29) defines the binary decisions.

B. Performance Analysis

The analysis and proofs follow nearly verbatim those of Sec. V, with the modification of the sampling technique $Q_{\mathcal{Z}}$. We highlight that the analysis of Sec. V relies on fixed sets $\mathcal{I}_{\text{THO}}$ and $\mathcal{I}_{\text{CHO}}$, and therefore, the unbiased estimator defined earlier cannot be used in the case of dynamic HO type selection, where $\boldsymbol{x}_t$ and $\boldsymbol{y}_t$ are coupled with the controller deciding the HO type of each UE in every slot. For that reason, we create a new unbiased estimator.

Let $\boldsymbol{z}_t^{\text{m}} = (\boldsymbol{x}_t^{\text{m}}, \boldsymbol{y}_t^{\text{m}})$ denote the *continuous* meta-learner output at slot t, where $\boldsymbol{x}_t^{\text{m}} \in [0,1]^{I\cdot J}$ and $\boldsymbol{y}_t^{\text{m}} \in [0,1]^{I\cdot J}$. The definition of $\boldsymbol{z}_t^{\text{m}}$ resembles eq. (13), with the distinction that in that earlier formulation, the users were partitioned into two disjoint sets $\mathcal{I}_{\text{THO}}$ and $\mathcal{I}_{\text{CHO}}$.

To construct the implementable binary decision $\boldsymbol{z}_t = (\boldsymbol{x}_t, \boldsymbol{y}_t)$, we start by defining for each $i \in \mathcal{I}$ the probability to be explicitly assigned (THO-enabled) at slot t:

$$\pi_i(t) \triangleq \sum_{j\in\mathcal{J}} x_{ij}^{\text{m}}(t), \tag{30}$$

where eq. (24) ensures that $\pi_i(t) \in [0,1]$. To obtain an unbiased estimator, i.e., $\mathbb{E}[\boldsymbol{z}_t] = \boldsymbol{z}_t^{\text{m}}$, we sample:

$$z_i(t) \sim \text{Bernoulli}\big(\pi_i(t)\big), \tag{31}$$

Authorized licensed use limited to: University of Cyprus. Downloaded on September 28,2026 at 10:36:41 UTC from IEEE Xplore. Restrictions apply.

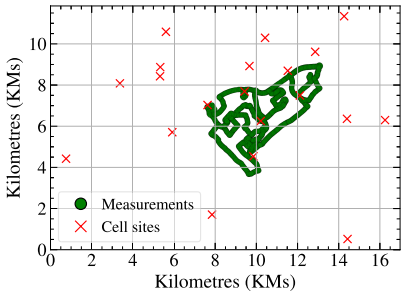


Fig. 10. Cell sites (red) and SINR measurement locations (green) used for evaluation.

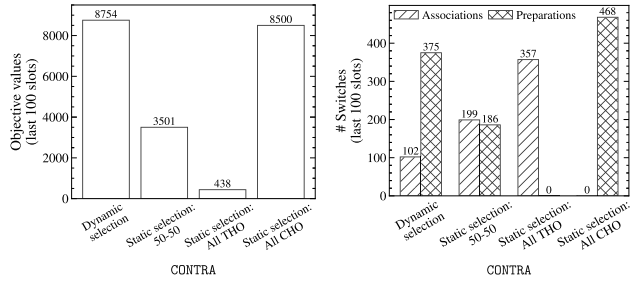


Fig. 11. Real SINR scenario with $I = 100$ UEs. Over the last 100 slots of the $T = 1$k simulation, (a) shows the objective function values (i.e., throughput and switching costs), and (b) the number of switches in associations (i.e., $\boldsymbol{x}_t$) and preparations (i.e., $\boldsymbol{y}_t$), for the dynamic and three variants of static HO type selection of CONTRA.

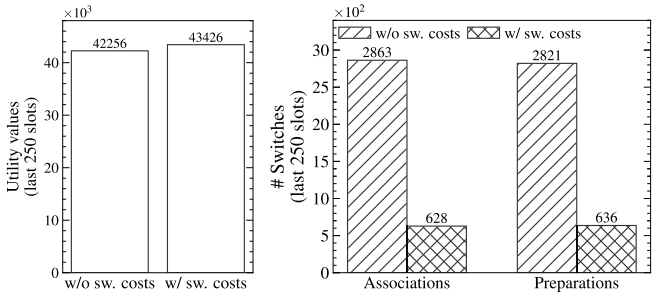


Fig. 12. Real SINR scenario with $I = 100$ UEs. Over the last 250 slots of the $T = 1$k simulation, (a) shows the utility function values (i.e., $\tilde{g}_t$), and (b) the number of switches in associations (i.e., $\boldsymbol{x}_t$) and preparations (i.e., $\boldsymbol{y}_t$), for ignoring and taking into account the switching costs.

and if $z_i(t) = 1$, the UE $i \in \mathcal{I}$ is THO-enabled at slot t, else CHO-enabled. Lastly, the associations/preparations are decided with the normalized quantities: $x_{ij}(t) = x^{\mathrm{m}}_{ij}(t)/\pi_i(t)$ and $y_{ij}(t) = y^{\mathrm{m}}_{ij}(t)/(1-\pi_i(t))$.

From the law of total expectation, and as $x_{ij}(t) = 0$ when $z_i(t) = 0$, we get $\mathbb{E}[x_{ij}(t)] = x^{\mathrm{m}}_{ij}(t)$ and similarly $\mathbb{E}[y_{ij}(t)] = y^{\mathrm{m}}_{ij}(t)$. This concludes that indeed, $\mathbb{E}[\boldsymbol{z}_t] = \boldsymbol{z}^{\mathrm{m}}_t$.

VIII. Performance Evaluation

We compare CONTRA with various *(i)* 3GPP-compliant, threshold-based THO and CHO algorithms, as well as *(ii)* RL benchmarks. The former are the algorithms currently used by MNOs and antenna vendors [1], [48], solving either solely the THO or CHO problem (i.e., not jointly). Given that these benchmarks treat THO and CHO as independent procedures, we adapt and extend *three* existing foundational *RL* algorithms from the literature that were originally designed for other tasks [25], [26], [69], such as only THO optimization. In this way, it becomes possible to compare the proposed algorithm with more advanced (i.e., not relying on pre-defined thresholds) benchmarks for precisely the problem at hand, namely, the joint optimization of traditional and conditional handovers.

The algorithms are evaluated in two main scenarios: *(i) volatile* SINR scenario, in which $s_{ij}(t)$ fluctuates every 10 slots within the range of [0, 30]dB [70], reflecting a wide spectrum of signal conditions (i.e., from poor to excellent values), and *(ii)* real SINR scenario from crowdsourced countrywide measurements. For the latter, we focus on an area of approximately 160 sq. kms, where 73,303 signal measurements in a second granularity are taken for 100 cells (20 cell sites). These cell sites are shown in Figure 10, marked in red, and the measurement areas are indicated in green. This dataset presents a highly competitive scenario as SINRs can vary arbitrarily and occasionally exhibit adversarial behavior (see Sec. IV). Given that these measurements are anonymized, we consider 100 fast-moving UEs, with random velocities ranging from 20 to 28 m/s and variances in [0, 5], placed at $t = 1$ in a random location with measurement (green). We adopt the Gauss-Markov mobility model [71], in line with prior works [22], with 0.9 randomness parameter to find the location of each UE in each subsequent slot $t = 2, \ldots, T$, for $T = 1$k. To incorporate the signal measurements, we map each new location to the nearest location with available measurements.

A. Understanding the Proposed Algorithm

First and foremost, it is imperative to showcase the differences between the static (Sec. V) and dynamic (Sec. VII) HO type selection. For that reason, we compare four variants of CONTRA: *(i)* dynamic HO type selection, where the controller decides if and when each user should operate in THO or CHO mode according to our framework, and three variants of static HO type selection, namely, *(i)* a random split with half of the users predefined as THO-enabled and the other half as CHO-enabled, as well as a configuration in which all users are *(ii)* THO-enabled, and *(iii)* CHO-enabled.

In detail, Figure 11 compares the performance of these four CONTRA variants over the last 100 slots of the real-SINR scenario. When all users are in THO mode, the objective function values are the lowest (i.e., 438), primarily due to the large number (357) and high cost of THOs. The 50–50 split between THO- and CHO-enabled users improves performance by nearly $\times 8$ compared to the all-THO configuration; yet it remains roughly $\times 2.5$ lower than both the automatic mode-selection and all-CHO cases. The automatic mode achieves higher performance ($\approx 3\%$) compared to the all-CHO configuration, as it executes fewer CHO preparations, even though it requires 102 THOs. From here onward, CONTRA is assessed in the dynamic HO formulation.

Secondly, we focus on the trade-off between throughput and the two switching cost terms to understand the importance

Authorized licensed use limited to: University of Cyprus. Downloaded on September 28,2026 at 10:36:41 UTC from IEEE Xplore. Restrictions apply.

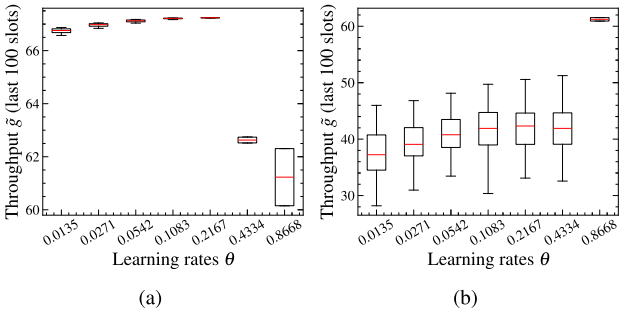


Fig. 13. Cumulative throughput $\tilde{g}$ for the last 100 of 1k slots for different learners in (a) stationary and (b) real SINR case.

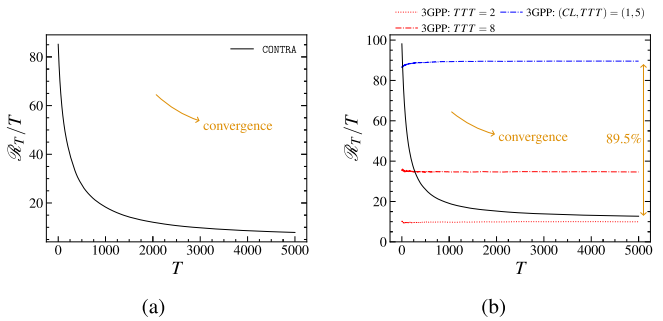


Fig. 14. Volatile scenario (SINR changes every 10 slots), for $T = 5$k slots, for the (a) continuous and (b) discrete decisions.

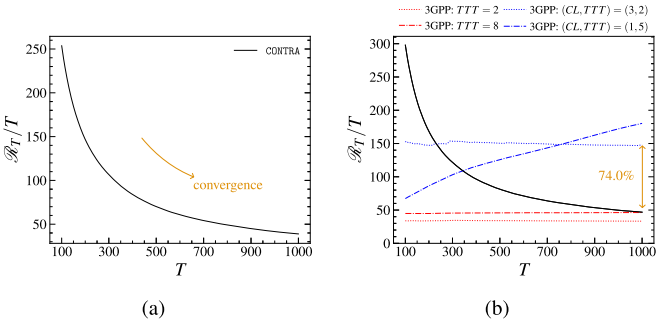


Fig. 15. Real-data SINR scenario for $T = 1$k, for the (a) continuous and (b) discrete decisions.

of the latter for the studied problem. Figure 12 shows the utility function and the number of association and preparation changes when switching costs are *(i)* excluded, and *(ii)* taken into account, for the real-SINR scenario. Focusing on the last 250 slots of $T = 1$k, we note that taking into account the switching cost reduces the number of associations (and thus, THOs) and preparation changes by approximately 77%, while achieving a 2.7% higher throughput. Hence, it becomes evident that solely maximizing the utility/throughput (i.e., without considering the switches) may not be the optimal strategy, given that comparable, or even better, performance can be achieved with fewer switches, thereby reducing signaling overhead, resource utilization, and HO delays.

Thirdly, to show the importance of combining learners with different rates, Figure 13 shows the attained throughput $\tilde{g}(.)$ in a case where the SINRs are *stationary* (changing only every 200 slots) and real (changing significantly between slots). Given that $T = 1$k, eq. (19) determines that 7 experts will be used, with learning rates 0.0135, 0.0271, 0.0542, 0.1083, 0.2167, 0.4334, and 0.8668 (combining eq. (20) and Lemma 2). In the stationary case, the OGA experts with small learning rates obtain up to 12% better throughput than the highest-learning-rate one. On the other hand, in the real case, the highest-learning-rate expert achieves 65% more throughput. These findings support the claims that in more stationary scenarios, smaller-learning-rate experts perform better as they update their decisions more carefully, while in volatile/real cases, it is the opposite, as higher-learning-rate experts adapt more quickly to the rapid changes of the environment.

B. Regret Analysis

Sec. VI shows that the regret guarantees hold for any benchmark; however, computing the benchmark that has full knowledge of the future is computationally intensive for mixed-integer programs [25], [72]. To facilitate, therefore, the average dynamic regret calculation, the optimal `Oracle` considered solves the optimization problem *in every step* using *CVXPY* [73], and a smaller number of $I = 20$ users. Apart from the `CONTRA`, we show the average dynamic regret of 3GPP threshold-based benchmarks. In the sequel, we refer to the 3GPP CHO benchmarks as (CL, TTT), where the first element denotes the number of cells that are prepared in each slot, provided they are the highest-SINR cells for more than TTT consecutive slots (similar to the A3 event); while for the 3GPP THO benchmarks, only TTT is used. For instance, the benchmark $(1, 2)$ prepares one cell for each user if it remains the highest-SINR for more than two slots.

Figure 14 shows the average dynamic regret of our proposed algorithm and the 3GPP-compliant competitors in the volatile case. In Figures 14(a) and 14(b), we show the continuous, $\boldsymbol{z} \in \mathcal{Z}^{\mathrm{c}}$, and discrete, $\boldsymbol{z} \in \mathcal{Z}$, decisions, respectively. In both cases, we observe that the average dynamic regret converges towards zero for $T = 5$k slots; and in the latter case, where the decisions are actually implementable in practice, `CONTRA` outperforms the best-performing CHO-only 3GPP benchmark by 89.5%. The average dynamic regret of the THO-only benchmarks stays almost constant for all slots ("stuck" in sub-optimal decisions). Lastly, we observe a relatively small discretization error of 16.2% measured between the continuous and discrete decision plots at $t = 5$k.

From Figure 15, the average dynamic regret converges again towards zero for the real SINR cases, showing that `CONTRA` is adaptable in all scenarios, with the gap between the continuous and discrete decisions being solely 1.83%. On the other hand, the best CHO-only (THO-only) 3GPP-compliant benchmark attains 74% (28.2%) higher (lower) dynamic regret at $t = 5$k. Yet, the THO-only benchmark is stuck in suboptimal (non-diminishing regret) decisions.

C. Comparison With Reinforcement Learning

To compare `CONTRA` with RL approaches, we need to adjust the definitions of *RL state* and *RL action*. We define the *RL state* with the SINRs of the previous time slots, see e.g., [25] and [26]. We intentionally avoid incorporating other UE

Authorized licensed use limited to: University of Cyprus. Downloaded on September 28,2026 at 10:36:41 UTC from IEEE Xplore. Restrictions apply.

features, such as velocity or ping-pong counters, to remain as aligned as possible with our problem formulation. We also define an *RL action* that determines whether each user $i \in \mathcal{I}$ is THO- or CHO-enabled and how many cells to prepare in the latter case. In principle, the CHO decisiona should be modeled as the selection of any subset of candidate cells. This formulation captures the "true" decision space of CHO, but in the case of RL, it leads to an exponential growth in the number of possible actions; specifically, $J+(2^J-1)$ per user. The first term, J, corresponds to actions regarding THOs: a user can be assigned to any of the available J cells. The second term (exponential), 2^J-1, corresponds to any non-empty subset of the J cells that can get prepared (minus the empty set).

Even when viewed purely from the search-space perspective, this exponential growth makes RL infeasible for moderate J, as exploration becomes prohibitively expensive. In contrast, CONTRA maintains a tractable search-space, avoiding the combinatorial explosion inherent to this RL formulation. To avoid the exponential explosion, we adopt a more compact parameterization for the RL approaches with only $2J$ actions per user. Here, the first J terms still correspond to THOs, while the rest J to CHOs, where the user prepares up to the top-J strongest cells in terms of SINR in the previous slots. The reward/objective remains as in CONTRA (see Sec. V), namely, throughput minus switching costs.

The three RL benchmarks are: *(i)* policy-gradient methods without a critic, through REINFORCE algorithm [69], *(ii)* value-based methods through Deep Q-Network (DQN) [25], and *(iii)* actor-critic methods with proximal updates through Proximal Policy Optimization (PPO) [26].

REINFORCE is a simpler method, that tries actions, observes total rewards, and adjusts the policy toward those that performed well. It does not estimate how good a state is (no critic), which makes it simple but noisy and data-hungry. More precisely, it implements a Monte Carlo policy-gradient method without a critic. Its policy network is a two-layer multilayer perceptron (MLP) with 128 neurons per layer and Tanh activations, followed by a softmax output layer producing a categorical distribution over all possible actions. After each episode, the algorithm computes discounted returns and updates the policy parameters using gradients of the log-probabilities weighted by returns. The optimizer is Adam with a learning rate of 3×10^{-4}.

DQN does not directly learn a policy; instead, it learns an action–value function $Q(state, action)$ that shows how good each action is. The agent then picks the action with the highest Q, allowing though a random exploration. More specifically, DQN consists of two hidden layers with 128 neurons and ReLU activations. Exploration follows an ε-greedy strategy with $\varepsilon_{\text{start}} = 1.0$, $\varepsilon_{\text{end}} = 0.05$, and exponential decay constant 200. A replay buffer of size 50k and minibatch size 64 are used, with soft target-network updates ($\tau = 0.01$) to improve stability. The loss is the mean-squared temporal-difference error, optimized with Adam at a learning rate of 10^{-3}.

PPO combines both ideas. It learns an *actor* (policy) and a *critic* (value estimate), and ensures stable updates by clipping how much the policy can change each step. Both the actor and critic share a two-layer MLP backbone (128 neurons per layer, Tanh activations). The actor outputs a categorical distribution over all actions, while the critic outputs a scalar state-value estimate. Training uses generalized advantage estimation with $\lambda = 0.95$ and the clipped objective parameter $\epsilon = 0.2$; also, minibatches of 64 samples for 3 epochs per cycle are used. Both networks are optimized using Adam with rate 3×10^{-4}.

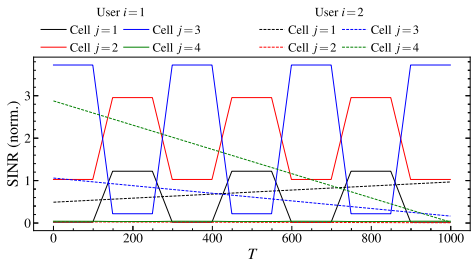
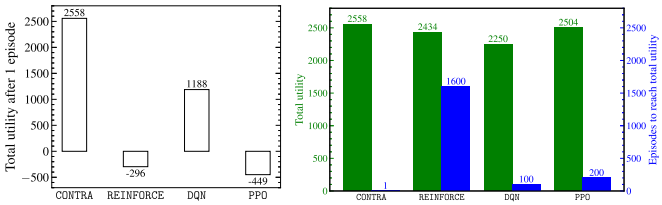


Fig. 16. Simple deep-dive example for $I = 2$ UEs, $J = 4$ cells, and $T = 1$k slots, with two SINR modes: one with more dynamic/abrupt changes (continuous lines) and one with more gradual changes (dotted lines).



(a) (b)

Fig. 17. Total utility for (a) one episode and (b) multiple episodes, of CONTRA vs three RL benchmarks for a simple deep-dive example with $I = 2$ UEs, $J = 4$ cells, and $T = 1$k slots, with the SINRs of Figure 16.

Unlike CONTRA, which learns and adapts online with a single exposure to the environment, these RL methods are trained over hundreds of simulated episodes, effectively granting them repeated access to the same system dynamics. This allows RL algorithms to asymptotically approximate the optimal policy under repeated trials, whereas CONTRA must adapt on-the-fly.

To showcase the disadvantages of RL even in a simple setting with four cells, we consider only two users with different types of mobility/SINR dynamics as can be seen in Figure 16: User 1 frequently switching between good and bad cells (high variability), and User 2 steadily moving toward a better SINR environment (gradual change). More precisely, the SINR of User 1 remains constant for 100 slots, then changes during a 50-slot transition period, and subsequently stabilizes again for another 100 slots, and so on.

In Figure 17(a), we run each algorithm once (i.e., one episode), since our proposed method requires only a single pass to produce its decisions and learn. As expected, RL algorithms, and specifically REINFORCE and PPO, perform poorly (negative utility) as they require many interactions over the same conditions to learn. On the other hand, DQN can improve its decision in the same episode, achieving, however, 53.6% lower utility than CONTRA. In Figure 17(b), we allow the RL benchmarks to learn this simple scenario by running them for multiple episodes and reporting their final utilities. Compared to our algorithm that requires a single episode/pass to reach $\approx$ 2.5k total utility, REINFORCE, DQN, and PPO

Authorized licensed use limited to: University of Cyprus. Downloaded on September 28,2026 at 10:36:41 UTC from IEEE Xplore. Restrictions apply.

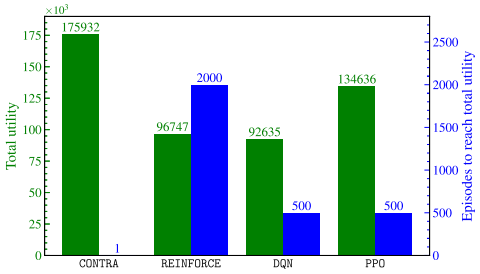


Fig. 18. Total utilities (i.e., $\tilde{g}_t$) and number of episodes required to reach these values, for the real SINR scenario with $I = 100$ UEs and $T = 1$k slots.

reach until 95.2%, 88.0%, and 97.9% of its performance, by running for 1.6k, 100, and 200 episodes, respectively.

Finally, in the real SINR scenario, the performance gap becomes even more pronounced. From Figure 18, and compared to our algorithm that requires a single pass to reach $\approx$ 175k total utility, REINFORCE, DQN, and PPO reach until 54.9%, 52.6%, and 76.6% of its performance, by running for 2k, 500, and 500 episodes, respectively. The gap thus widens significantly in realistic network conditions, showing that CONTRA attains substantially higher performance without the extensive training required by RL-based approaches. Even with this simplified assumption, our approach outperforms the RL in the experiments executed with real SINR/UEs. This poor performance of RL approaches is expected, as they do not adapt to volatile or adversarial environments (known to converge under stationarity only), and do not offer performance guarantees as those we provide in Lemma 3.

IX. Conclusion

This study is motivated by the demanding mobility support requirements in 6G, technological advances in handover (HO) mechanisms, and the availability of AI-based management solutions in O-RAN. Its goal is to address a challenge that is critical as 5G deployments mature and 6G systems emerge: executing HOs rapidly and reliably, especially in dense deployments and high-frequency bands, where (i) traditional HO (THO) mechanisms exhibit high failure rates and thus, increased delays, and (ii) the newly introduced by 3GPP, Conditional HOs (CHOs), which tackle these issues by enabling proactive cell reservations, raise intricate trade-offs in signaling and resource utilization. To better understand these issues, we analyze countrywide datasets from a major operator in Europe; and the findings underscore the need for more adaptive and robust HO control. Motivated by this, we propose CONTRA, a meta-learning, provably-optimal, minimal-assumption, O-RAN-compatible algorithm that jointly optimizes THO and CHO and adapts to run-time observations, demonstrating significant improvements in real-world and synthetic scenarios. While our work focuses on O-RAN, the core ideas can be applied to other 3GPP-compliant RAN architectures that support real-time measurements and centralized decision making.

Appendix

A. Proof of Lemma 2

First, we observe that the diameter of the domain is bounded, i.e., $\|\boldsymbol{z}-\boldsymbol{z}'\|_2 \leq \|\boldsymbol{x}-\boldsymbol{x}'\|_2+\|\boldsymbol{y}'-\boldsymbol{y}'\|_2 \leq \sqrt{2I_{\text{THO}}} + \sqrt{I_{\text{CHO}}(J-1)} = D$, since the maximum distance for the former occurs when assignments $\boldsymbol{x}$ and $\boldsymbol{x}'$ are all different, and for the latter, when all cells are prepared in $\boldsymbol{y}$ and one cell is prepared per UE in $\boldsymbol{y}'$ (i.e., minimum preparation). Similarly, we prove the bound for the C_t-norm and its dual.

Finally, we compute the gradients component-wise and, for simplicity, we omit the time t index. For $i \in \mathcal{I}_{\text{THO}}, j \in \mathcal{J}$ (similarly for $i \in \mathcal{I}_{\text{CHO}}$ and the y_{ij}), and since $\partial l_j/\partial x_{ij} = 1$, we have $\partial \tilde{g}_t/\partial x_{ij} = \log c_{ij} - \log l_j - 1$. Using $c_n \leq c_{\max}$ and $l_j \leq I_{\text{THO}} + I_{\text{CHO}} = I$, we can write: $|\partial g_t/\partial x_{ij}| \leq \max\{\log c_{\max} - 1,\ \log I + 1\} = M$. The proof is finalized, as there are $I\,J$ total components in the gradient vector; and likewise for the C_t-norm.

B. Proof of Lemma 3

The proof tailors the result of [20], [27], [74], which, however, solve the UE-cell association problem (THO only). Eq. (17) can be rewritten: $\mathbb{E}[\mathcal{R}_T] =$

$$\sum_{t=1}^{T}\Big(\big(\tilde{g}_t(\boldsymbol{z}_t^\star) - \|\boldsymbol{z}_t^\star - \boldsymbol{z}_{t-1}^\star\|_{C_t}\big) - \big(\tilde{g}_t(\boldsymbol{z}_t^{\text{m}}) - \|\boldsymbol{z}_t^{\text{m}} - \boldsymbol{z}_{t-1}^{\text{m}}\|_{C_t}\big)\Big)$$
$$+ \mathbb{E}\Bigg[\sum_{t=1}^{T}\Big(\big(\tilde{g}_t(\boldsymbol{z}_t^{\text{m}}) - \|\boldsymbol{z}_t^{\text{m}} - \boldsymbol{z}_{t-1}^{\text{m}}\|_{C_t}\big)$$
$$- \big(\tilde{g}_t(\boldsymbol{z}_t) - \|\boldsymbol{z}_t - \boldsymbol{z}_{t-1}\|_{C_t}\big)\Big)\Bigg],$$

where we bound the expected dynamic regret of the relaxed preparation decisions $\{\boldsymbol{z}_t^{\text{m}}\}_t$ and, afterwards, the error of the discrete decisions $\{\boldsymbol{z}_t\}_t$ as can be seen from the first and second terms, respectively. From the first term in eq. (23), sublinear dynamic regret can be achieved for the relaxed decisions. It follows by bounding the regret of each expert w.r.t. the benchmark and then the regret of the meta-learner w.r.t. any expert. For the extra cost/error introduced due to the quantization routine $Q_{\mathcal{Z}}$, we bound the term as follows:

$$\mathbb{E}\left[\sum_{t=1}^{T}\Big(\tilde{g}_t\left(\boldsymbol{z}_t^{\text{m}}\right) - \left\|\boldsymbol{z}_t^{\text{m}} - \boldsymbol{z}_{t-1}^{\text{m}}\right\|_{C_t} - \tilde{g}_t\left(\boldsymbol{z}_t\right) + \left\|\boldsymbol{z}_t - \boldsymbol{z}_{t-1}\right\|_{C_t}\Big)\right](*)$$
$$\leq \left(G + \sqrt{a_{\max} + b_{\max}}\right)\sum_{t=1}^{T}$$
$$\sqrt{\mathbb{E}\left[\sum_{i=1}^{I}\sum_{j=1}^{J}\left(z_{ij}(t) - \mathbb{E}\left[z_{ij}(t)\right]\right)^2\right]}(**)$$
$$\leq T\left(G + \sqrt{a_{\max} + b_{\max}}\right)\sqrt{I_{\text{CHO}}J/4 + I_{\text{THO}}(1 - 1/J)}$$

leading to eq. (23). Specifically, for $(*)$, we use Jensen's inequality, the linearity of expectation, and Lipschitz continuity. For the latter, consider $\boldsymbol{z}_1, \boldsymbol{z}_2 \in \mathcal{Z}^{\text{c}}$, use the triangle inequality and Lipschitz constant G of g_t and the Euclidean norm, and define $\boldsymbol{z}_{t-1} = C$, to get:

$$\Big|\big(\tilde{g}_t(\boldsymbol{z}_1) - \|\boldsymbol{z}_1 - C\|_{C_t}\big) - \big(\tilde{g}_t(\boldsymbol{z}_2) - \|\boldsymbol{z}_2 - C\|_{C_t}\big)\Big|$$
$$\leq \big|\tilde{g}_t(\boldsymbol{z}_1) - \tilde{g}_t(\boldsymbol{z}_2)\big| + \big|\|\boldsymbol{z}_1 - C\|_{C_t} - \|\boldsymbol{z}_2 - C\|_{C_t}\big|$$

Authorized licensed use limited to: University of Cyprus. Downloaded on September 28,2026 at 10:36:41 UTC from IEEE Xplore. Restrictions apply.

$$\leq \left(G + \sqrt{a_{\max} + b_{\max}}\right) \|\boldsymbol{z}_1 - \boldsymbol{z}_2\|.$$

Lastly, note that $(**)$ calculates the variance of $\boldsymbol{z}_t$. For $\boldsymbol{y}_t \in \{0,1\}^{I_{\text{CHO}} \cdot J}$, the maximum variance each component can obtain is $1/4$ (variance per element is $q(1-q)$ due to Bernoulli trial with prob q, and maximizes for $q = 0.5$); hence, we can upper bound the part for CHO-enabled UEs expression with $I_{\text{CHO}} J/4$. For $\boldsymbol{x}_t \in \{0,1\}^{I_{\text{THO}} \cdot J}$, the binary vector is subject to a simplex per user; thus, each sum of THO-enabled users w.r.t. j is upper bounded by variance of $1 - 1/J$, and the overall part of the term in the square root has bound $I_{\text{THO}}(1 - 1/J)$.

REFERENCES

[1] *5G; NR; NR and NG-RAN Overall Description*, document TS 38.300, 3GPP, 2024.

[2] *LTE; E-UTRA and E-UTRAN; Overall Description*, document TS 36.300, 3GPP, 2024.

[3] M. Kalntis et al., "Through the Telco lens: A countrywide empirical study of cellular handovers," in *Proc. ACM Internet Meas. Conf.*, Nov. 2024, pp. 51–67.

[4] J. Stanczak, U. Karabulut, and A. Awada, "Conditional handover in 5G–principles, future use cases and FR2 performance," in *Proc. Int. Wireless Commun. Mobile Comput. (IWCMC)*, May 2022, pp. 660–665.

[5] W. Chen, P. Gaal, J. Montojo, and H. Zisimopoulos, *Fundamentals of 5G Communications: Connectivity for Enhanced Mobile Broadband and Beyond*, 1st ed., New York, NY, USA: McGraw-Hill.

[6] V. W. S. Wong, R. Schober, D. W. K. Ng, and L.-C. Wang, *Key Technologies for 5G Wireless Systems*. Cambridge, U.K.: Cambridge Univ. Press.

[7] P. Jain, R. Borsato, *3GPP Workshop on 6G; Chairman's Summary*. [Online]. Available: https://www.3gpp.org/technologies/6gworkshop-2025

[8] H. Bergström et al., "Dependable networks: From best-effort to guaranteed performance," Ericsson, Sweden, Tech. Rep. #5, May 2025.

[9] X. Chen, D. W. K. Ng, W. Yu, E. G. Larsson, N. Al-Dhahir, and R. Schober, "Massive access for 5G and beyond," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 3, pp. 615–637, Mar. 2021.

[10] W. Chen et al., "5G-advanced toward 6G: Past, present, and future," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 6, pp. 1592–1619, Jun. 2023.

[11] *6GWS-250097: A Sensing-Mode Handoff Mechanism for High-Resolution and Accurate Sensing in 6G Networks*, Vestel Electron., Turkey, Mar. 2025.

[12] J. Vikberg, G. Hall, T. Cagenius, R. Wang, and J. Schultz, "Robustness evolution: Building robust critical networks with the 5G system," *Ericsson Technol. Rev.*, vol. 2021, no. 11, pp. 2–12, Nov. 2021.

[13] Ericsson.*This is the Key to Mobility Robustness in 5G Networks*. Accessed: Jun. 1, 2025. [Online]. Available: https://www.ericsson.com/en/blog/2020/5/the-key-to-mobility-robustness-5g-networks

[14] S. B. Iqbal, U. Karabulut, A. Awada, A. N. Barreto, P. Schulz, and G. Fettweis, "Mobility performance analysis of RACH optimization based on decision tree supervised learning for conditional handover in 5G beamformed networks," in *Proc. EW*, 2023, pp. 68–75.

[15] E. Juan, M. Lauridsen, J. Wigard, and P. Mogensen, "Performance evaluation of the 5G NR conditional handover in LEO-based non-terrestrial networks," in *Proc. WCNC*, 2022, pp. 2488–2493.

[16] A. Prado, F. Mehmeti, and W. Kellerer, "Cost-efficient mobility management in 5G," in *Proc. IEEE 24th Int. Symp. World Wireless, Mobile Multimedia Netw. (WoWMoM)*, Jun. 2023, pp. 215–224.

[17] S. Bin Iqbal, A. Awada, U. Karabulut, I. Viering, P. Schulz, and G. P. Fettweis, "On the modeling and analysis of fast conditional handover for 5G-advanced," in *Proc. IEEE 33rd Annu. Int. Symp. Pers., Indoor Mobile Radio Commun. (PIMRC)*, Sep. 2022, pp. 595–601.

[18] H. Martikainen, I. Viering, A. Lobinger, and T. Jokela, "On the basics of conditional handover for 5G mobility," in *Proc. IEEE 29th Annu. Int. Symp. Pers., Indoor Mobile Radio Commun. (PIMRC)*, Sep. 2018, pp. 1–7.

[19] E. Hazan, "Introduction to online convex optimization," *Found. Trends Optim.*, vol. 2, nos. 3–4, pp. 157–325, Aug. 2016.

[20] L. Zhang, W. Jiang, S. Lu, and T. Yang, "Revisiting smoothed online learning," in *Proc. NeurIPS*, 2021, pp. 13599–13612.

[21] A. Hassan et al., "Vivisecting mobility management in 5G cellular networks," in *Proc. ACM SIGCOMM Conf.*, Aug. 2022, pp. 86–100.

[22] M. Gupta, S. Chinchali, P. P. Varkey, and J. G. Andrews, "Forecaster-aided user association and load balancing in multi-band mobile networks," *IEEE Trans. Wireless Commun.*, vol. 23, no. 5, pp. 5157–5171, May 2024.

[23] M. Gupta, R. M. Dreifuerst, A. Yazdan, P.-H. Huang, S. Kasturia, and J. G. Andrews, "Load balancing and handover optimization in multi-band networks using deep reinforcement learning," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2021, pp. 1–6.

[24] J. Choi, W.-H. Lee, Y.-H. Kim, J.-H. Lee, and S.-C. Kim, "Throughput estimation based distributed base station selection in heterogeneous networks," *IEEE Trans. Wireless Commun.*, vol. 14, no. 11, pp. 6137–6149, Nov. 2015.

[25] A. Prado, F. Stöckeler, F. Mehmeti, P. Krämer, and W. Kellerer, "Enabling proportionally-fair mobility management with reinforcement learning in 5G networks," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 6, pp. 1845–1858, Jun. 2023.

[26] J. Voigt, P. J. Gu, and R. M. Rost, "A deep RL-based approach for adaptive handover protocols," in *Proc. SCC*, 2025, pp. 1–6.

[27] M. Kalntis, A. Lutu, J. O. Iglesias, F. A. Kuipers, and G. Iosifidis, "Smooth handovers via smoothed online learning," in *Proc. IEEE Conf. Comput. Commun. (IEEE INFOCOM)*, May 2025, pp. 1–10.

[28] M. I. Saglam, "Conditional handover for non-terrestrial networks," in *Proc. 10th Int. Conf. Wireless Netw. Mobile Commun. (WINCOM)*, Oct. 2023, pp. 1–5.

[29] L. Yang, X. Yang, and Z. Bu, "A conditional handover strategy based on trajectory prediction for high-speed terminals in LEO satellite networks," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, Jun. 2024, pp. 1697–1701.

[30] J. Stańczak, U. Karabulut, and A. Awada, "Conditional handover modelling for increased contention free resource use in 5G-advanced," in *Proc. IEEE 34th Annu. Int. Symp. Pers., Indoor Mobile Radio Commun. (PIMRC)*, Sep. 2023, pp. 1–6.

[31] S. C. Sundararaju, S. Ramamoorthy, D. P. Basavaraj, and V. Phanindhar, "Advanced conditional handover in 5G and beyond using Q-learning," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Apr. 2024, pp. 1–6.

[32] C. Fiandrino, D. J. Martínez-Villanueva, and J. Widmer, "A study on 5G performance and fast conditional handover for public transit systems," *Comput. Commun.*, vol. 209, pp. 499–512, Sep. 2023.

[33] J. Stanczak, U. Karabulut, A. Awada, and P. Spapis, "Signalling-efficient CFRA resource updating for conditional handover in 5G-advanced," in *Proc. IEEE Conf. Standards Commun. Netw. (CSCN)*, Nov. 2023, pp. 34–39.

[34] A. Prado, H. Vijayaraghavan, and W. Kellerer, "ECHO: Enhanced conditional handover boosted by trajectory prediction," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2021, pp. 1–6.

[35] C. Lee, H. Cho, S. Song, and J.-M. Chung, "Prediction-based conditional handover for 5G mm-Wave networks: A deep-learning approach," *IEEE Veh. Technol. Mag.*, vol. 15, no. 1, pp. 54–62, Mar. 2020.

[36] H.-S. Park, H. Kim, C. Lee, and H. Lee, "Mobility management paradigm shift: From reactive to proactive handover using AI/ML," *IEEE Netw.*, vol. 38, no. 2, pp. 18–25, Mar. 2024.

[37] S. B. Iqbal, S. Nadaf, A. Awada, U. Karabulut, P. Schulz, and G. P. Fettweis, "On the analysis and optimization of fast conditional handover with hand blockage for mobility," *IEEE Access*, vol. 11, pp. 30040–30056, 2023.

[38] M. Kalntis, F. A. Kuipers, and G. Iosifidis, "CHOMET: Conditional handovers via meta-learning," in *Proc. 23rd Int. Symp. Model. Optim. Mobile, Ad Hoc, Wireless Netw. (WiOpt)*, May 2025, pp. 1–8.

[39] L. Chen, S. T. Jose, I. Nikoloska, S. Park, T. Chen, and O. Simeone, "Learning with limited samples: Meta-learning and applications to communication systems," *Found. Trends Signal Process.*, vol. 17, no. 2, pp. 79–208, Jan. 2023.

[40] A. Collet, A. Bazco-Nogueras, A. Banchs, and M. Fiore, "AutoManager: A meta-learning model for network management from intertwined forecasts," in *Proc. IEEE Conf. Comput. Commun. (IEEE INFOCOM)*, May 2023, pp. 1–10.

[41] J. Zhang, Y. Yuan, G. Zheng, I. Krikidis, and K.-K. Wong, "Embedding model-based fast meta learning for downlink beamforming adaptation," *IEEE Trans. Wireless Commun.*, vol. 21, no. 1, pp. 149–162, Jan. 2022.

[42] Q. He, A. Moayyedi, G. Dán, G. P. Koudouridis, and P. Tengkvist, "A meta-learning scheme for adaptive short-term network traffic prediction," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 10, pp. 2271–2283, Oct. 2020.

[43] Y. Hu, M. Chen, W. Saad, H. V. Poor, and S. Cui, "Distributed multi-agent meta learning for trajectory design in wireless drone networks," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 10, pp. 3177–3192, Oct. 2021.

Authorized licensed use limited to: University of Cyprus. Downloaded on September 28,2026 at 10:36:41 UTC from IEEE Xplore. Restrictions apply.

[44] S. Yue, J. Ren, J. Xin, D. Zhang, Y. Zhang, and W. Zhuang, "Efficient federated meta-learning over multi-access wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 5, pp. 1556–1570, May 2022.
[45] A. Feriani et al., "Multiobjective load balancing for multiband downlink cellular networks: A Meta{-} reinforcement learning approach," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 9, pp. 2614–2629, Sep. 2022.
[46] R. M. Sohaib, O. Onireti, K. Tan, Y. Sambo, R. Swash, and M. A. Imran, "Meta-transfer learning-based handover optimization for V2N communication," *IEEE Trans. Veh. Technol.*, vol. 73, no. 11, pp. 17331–17346, Nov. 2024.
[47] *LTE; E-UTRA; Radio Resource Control (RRC); Protocol Specification*, document TS 36.331, {3GPP}, 2024.
[48] *5G; NR; Radio Resource Control (RRC); Protocol Specification*, document TS 38.331, 3GPP, 2024.
[49] *Consecutive Conditional HO*, document R2-1909862, 3GPP, 2019.
[50] A. Garcia-Saavedra and X. Costa-Pérez, "O-RAN: Disrupting the virtualized RAN ecosystem," *IEEE Commun. Standards Mag.*, vol. 5, no. 4, pp. 96–103, Dec. 2021.
[51] M. Ghoshal et al., "Performance of cellular networks on the wheels," in *Proc. ACM Internet Meas. Conf.*, Oct. 2023, pp. 678–695.
[52] C. Ballesteros, A. Pfadler, L. Montero, J. Romeu, and L. Jofre, "Adaptive beamwidth optimization under Doppler ICI and positioning errors at mmWave bands," *Proc. Veh. Commun.*, vol. 34, Mar. 2022, Art. no. 100456.
[53] H. Elgendi, T. Levanen, A. Lo, E. Peralta, S. Nielsen, and M. Valkama, "Uplink performance of LTE and NR with high-speed trains," in *Proc. IEEE 93rd Veh. Technol. Conf. (VTC-Spring)*, Apr. 2021, pp. 1–5.
[54] Y. S. Soh, T. Q. S. Quek, M. Kountouris, and H. Shin, "Energy efficient heterogeneous cellular networks," in *Proc. IEEE JSAC*, Mar. 2013, vol. 31, no. 5, pp. 840–850.
[55] R. A. Fisher, *Statistical Methods for Research Workers*. Edinburgh, Scotland: Oliver and Boyd, 1970.
[56] M. Kalntis, G. Iosifidis, and F. A. Kuipers, "Adaptive resource allocation for virtualized base stations in O-RAN with online learning," *IEEE Trans. Commun.*, vol. 73, no. 3, pp. 1787–1800, Mar. 2025.
[57] J. Dai et al., "O-RAN-enabled intelligent network slicing to meet service-level agreement (SLA)," *IEEE Trans. Mobile Comput.*, vol. 24, no. 2, pp. 890–906, Feb. 2025.
[58] D. Zidic, T. Mastelić, I. N. Kosović, M. Ćagalj, and J. Lörincz, "Analyses of ping-pong handovers in real 4G telecommunication networks," *Comput. Netw.*, vol. 227, Aug. 2023, Art. no. 109699.
[59] P. Viswanath, D. N. C. Tse, and R. Laroia, "Opportunistic beamforming using dumb antennas," *IEEE Trans. Inf. Theory*, vol. 48, no. 6, pp. 1277–1294, Jun. 2002.
[60] A. Beck, *First-order Methods in Optimization*(MOS-SIAM Series on Optimization). Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 2017.
[61] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.
[62] N. Littlestone and M. K. Warmuth, "The weighted majority algorithm," in *Proc. 30th Annu. Symp. Found. Comput. Sci.*, Apr. 1989, pp. 256–261.
[63] S. Arora, E. Hazan, and S. Kale, "The multiplicative weights update method: A meta-algorithm and applications," *Theory Comput.*, vol. 8, no. 6, pp. 121–164, Mar. 2012.
[64] M. Zinkevich, "Online convex programming and generalized infinitesimal gradient ascent," in *Proc. ICML*, 2003, pp. 928–935.
[65] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *J. Comput. Syst. Sci.*, vol. 55, no. 1, pp. 119–139, Aug. 1997.
[66] T. Lattimore and C. Szepesvári, *Bandit Algorithms*. Cambridge, U.K.: Cambridge Univ. Press, 2020.
[67] J. Duchi, S. Shalev-Shwartz, Y. Singer, and T. Chandra, "Efficient projections onto thel1-ball for learning in high dimensions," in *Proc. 25th Int. Conf. Mach. Learn.*, 2008, pp. 272–279.
[68] G. S. Paschos, A. Destounis, and G. Iosifidis, "Online convex optimization for caching networks," *IEEE/ACM Trans. Netw.*, vol. 28, no. 2, pp. 625–638, Apr. 2020.
[69] R. J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," *Mach. Learn.*, vol. 8, nos. 3–4, pp. 229–256, May 1992.
[70] *LTE; E-UTRA; Requirements for Support of Radio Resource Management*, document TS 36.133, {3GPP}, 2024.
[71] T. Camp, J. Boleng, and V. Davies, "A survey of mobility models for ad hoc network research," *Wireless Commun. Mobile Comput.*, vol. 2, no. 5, pp. 483–502, Aug. 2002.
[72] M. A. Bragin and E. L. Tucker, "Surogate ' level-based' Lagrangian relaxation for mixed-integer linear programming," *Sci. Rep.*, vol. 12, no. 1, p. 22417, Dec. 2022.
[73] S. Diamond and S. Boyd, "CVXPY: A Python-embedded modeling language for convex optimization," *J. Mach. Learn. Res.*, vol. 17, no. 83, pp. 1–5, 2016.
[74] L. Zhang, S. Lu, and Z.-H. Zhou, "Adaptive online learning in dynamic environments," in *Proc. NeurIPS*, 2018, pp. 1323–1333.

Authorized licensed use limited to: University of Cyprus. Downloaded on September 28,2026 at 10:36:41 UTC from IEEE Xplore. Restrictions apply.