iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** VIII **Month of publication:** August 2026

DOI: https://doi.org/10.22214/ijraset.2026.84754

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue VIII Aug 2026- Available at www.ijraset.com

Energy-Efficient, Fair, and QoS-Aware RL-Based Power Control for Massive MIMO Networks

Emad Addin Al-Sharafi
Department of Computer Science and Engineering, College of Engineering, Qatar University, Doha, Qatar

Abstract: ***Massive MIMO networks require power-control schemes that jointly account for energy efficiency, inter-cell interference, fairness, and time-varying QoS constraints under mobility and traffic dynamics. This paper proposes a deep reinforcement learning (DRL) framework for downlink power control in a realistic multi-cell massive MIMO setting. We develop Smart-MIMO-Cell-Env, a four-cell downlink environment with 16 antennas and 6 users per cell, correlated Rayleigh channels with 3GPP-inspired large-scale fading, user mobility, time-varying traffic load, channel aging, and optional interference coordination. A centralized agent observes an eight-dimensional state (effective transmit power, aggregate array gain, average SINR and rate, inverse-SINR penalty, mobility factor, traffic load, and channel age) and emits a continuous 88-dimensional action vector of per-user powers and per-antenna gains. A composite reward balances throughput, power consumption, link quality, fairness, and temporal stability. We train and evaluate Proximal Policy Optimization (PPO) and Soft Actor-Critic (SAC) in Ray RLlib under identical environment, state/action spaces, and network architectures, and assess both final policies over 200 independent episodes. PPO attains a mean episode reward of −6.130, mean downlink rate 0.264, mean inverse-SINR penalty $\bar{I}$ = 1.714, power-efficiency indicator 1.973, mean effective power 1.37 × 10^{-1} W, and mean array gain 1.154, whereas SAC settles at −108.724 reward with $\bar{I}$ = 24.032. PPO therefore learns a higher-throughput policy with substantially better link quality at a moderate power level. The results demonstrate the viability of DRL for energy-efficient, fair, and QoS-aware power control in realistic massive MIMO deployments and identify on-policy learning as the better match for the non-stationary dynamics of such networks.***

Keywords: ***Massive MIMO, power control, energy efficiency, deep reinforcement learning, proximal policy optimization, soft actor-critic, interference management, 5G networks.***

I. INTRODUCTION

Massive multiple-input multiple-output (MIMO) technology has become a cornerstone of fifth generation (5G) networks and beyond, offering substantial gains in spectral and energy efficiency through spatial multiplexing with large antenna arrays. By exploiting time-division duplex (TDD) reciprocity and channel hardening, massive MIMO base stations can serve many users simultaneously with linear precoding while keeping signal-processing complexity manageable. Deploying massive MIMO in multi-cell settings, however, introduces challenges that must be addressed for practical operation.

In multi-cell and cell-free deployments, inter-cell interference and pilot contamination remain critical bottlenecks that degrade system performance unless effective coordination and power-control mechanisms are in place. These challenges are exacerbated by dynamic traffic patterns, user mobility, channel aging, and partial or delayed channel-state information (CSI). The 5G New Radio (NR) standard specifies uplink power control via fractional path-loss compensation with open- and closed-loop components and separate rules for channels such as PUSCH, PUCCH, SRS, and PRACH (TS 38.213 [1]). While these standardized mechanisms provide a robust baseline that trades cell-edge coverage against inter-cell interference, they do not explicitly optimize energy efficiency, network-wide fairness, or quality-of-service (QoS) guarantees under realistic non-stationary conditions, nor do they directly address the broader downlink resource-control problem considered here.

Classical optimization-based formulations — weighted sum-rate, max–min, or proportional-fair utilities, for example — deliver strong theoretical guarantees for specific objectives, but they typically assume accurate system models, centralized computation, and slowly varying channels. In practice, model mismatches, mobility, and bursty traffic quickly erode the optimality of such static solutions. These limitations motivate learning-based controllers that adapt online to the environment they experience.

Recent advances in deep reinforcement learning (DRL) offer a promising route to adaptive, near-real-time power allocation without requiring perfect analytic models. DRL-based schemes learn policies that map state representations to continuous control actions while implicitly capturing complex trade-offs among throughput, power consumption, interference, and fairness. Among continuous-control DRL algorithms, Proximal Policy Optimization (PPO) and Soft Actor-Critic (SAC) are the two most widely

©IJRASET: All Rights are Reserved

used, owing to their stability and sample efficiency respectively; which of the two is better suited to non-stationary radio environments is, however, far from obvious a priori.

This work develops a DRL-based framework for downlink power control in multi-cell massive MIMO, instantiates it with PPO and SAC, and evaluates it in a realistic, non-stationary environment. A key practical aspect is that both the *state* and *action* interfaces admit multiple representations: we support compact aggregate feature vectors as well as richer and ablated feature sets used for analysis, and we consider two equivalent action parameterizations of the same underlying control problem (a single concatenated continuous vector, or a factorized representation that treats power and gain control as two coupled sub-actions). The main contributions are as follows.

1) We design Smart-MIMO-Cell-Env, an enhanced four-cell massive MIMO environment with 16 antennas and 6 users per cell, correlated Rayleigh fading with 3GPP-inspired large-scale path loss, user mobility, time-varying traffic load, channel aging, and optional interference coordination. For the primary experiments reported here, the environment exposes a compact eight-dimensional aggregate state capturing effective transmit power, aggregate array gain, average SINR and rate, inverse-SINR penalty, mobility factor, traffic load, and channel age; feature ablations derived from the same underlying signals support the feature-importance analysis. The resulting control problem is high-dimensional and continuous: the agent selects per-user transmit powers and per-antenna gain coefficients, represented either as one concatenated action vector or as two coupled sub-actions.

2) We construct a multi-objective reward function that combines spectral efficiency, power consumption, link-quality robustness via an inverse-SINR penalty proxy, robustness to dynamic conditions (mobility, traffic load, and CSI aging), fairness across cells, and temporal stability of the power policy. The reward explicitly penalizes degraded link quality, excessive effective transmit power, rapid fluctuations in the effective transmit power, and per-cell rate imbalance over a recent window, while rewarding balanced and stable operation.

3) We implement PPO and SAC in Ray RLlib with identical environment, observation and action semantics, network architectures, and hardware budgets, and perform a head-to-head comparison. Under the same training protocol, the final PPO policy evaluated over 200 episodes achieves a mean episode reward of approximately −6.13 with mean downlink rate 0.264 and mean inverse-SINR penalty $\bar{I}$ = 1.714, whereas SAC attains about −108.72 reward, rate 0.047, and $\bar{I}$ = 24.03 at a much lower effective power level. Together with a feature–reward correlation analysis and a per-feature freezing study, the results indicate that PPO is more robust under the non-stationary and partially history-dependent dynamics of Smart-MIMO-Cell-Env and yields more favourable rate–link-quality–power trade-offs than SAC.

The remainder of this paper is organized as follows. Section II reviews related work on power control and reinforcement learning in 5G networks. Section III presents the system model and the Markov decision process formulation. Section IV describes the PPO and SAC configurations used with Smart-MIMO-Cell-Env. Section V reports the experimental results, including training behaviour, policy evaluation metrics, feature-importance analyses, and a discussion of limitations. Section VI concludes the paper and outlines directions for future work.

II. RELATED WORK

A. *From NR Fractional Power Control to Optimization Baselines*

In NR, uplink power control is specified in TS 38.213 through open- and closed-loop *fractional* path-loss compensation, where α denotes the path-loss compensation factor and P_0 the nominal power offset. Tuning (α, P_0) trades cell-edge coverage against inter-cell interference, but still offers no explicit guarantees for energy efficiency, fairness, or delay and outage under non-stationary load conditions [2]. These standardized rules therefore act as a robust but model-driven baseline rather than a network-wide optimizer.

Optimization-based formulations remain the main theoretical reference. Weighted-MMSE and weighted-sum-rate (WMMSE/WSR) problems deliver strong spectral efficiency, while *fairness* is typically enforced through network-wide max–min or proportional-fair (PF) utilities. Recent massive MIMO studies extend these formulations to multi-cell settings with explicit scalability and fairness considerations at the network level [3]. Our work uses these optimization baselines and the NR rules as motivation and reference points for the learning-based controller.

B. *Energy Efficiency: Models and Recent Evidence*

Energy efficiency (EE) is usually modelled as a fractional metric (bits/Joule) that couples transmit power, power-amplifier efficiency and backoff, and circuit power. Recent work quantifies how EE-optimal operating points in massive MIMO can deviate from purely rate-optimal allocations, and how array adaptation and policy update timing affect EE in 5G deployments [4].

In parallel, data-driven controllers have been proposed that learn power-control policies directly from large-scale fading while implicitly capturing EE trade-offs at inference time, although most deep models still optimize rate-oriented proxies by default [5], [6]. The multi-objective reward used in our DRL formulation explicitly encodes both rate and power terms, aligning the learned policies with these EE considerations.

C. Fairness Mechanisms: PF versus Max–Min

Network-wide fairness in multi-cell massive MIMO has moved beyond pure WSR maximization toward scalable utility functions that balance performance across users and cells. Ghazanfari *et al.* propose a geometric-mean fairness surrogate that interpolates between PF and max–min objectives while remaining convex and tractable at large scale [3]. Such formulations motivate RL rewards that incorporate PF-style or worst-user surrogates rather than reporting only average or fifth-percentile rates. In our setting, fairness is captured through explicit rate-balance terms in the reward and through evaluation metrics that monitor per-cell rate dispersion.

D. QoS-Aware Power Control: Delay and Outage

Quality of service has evolved from implicit SINR and rate surrogates toward *explicit* delay and outage constraints. Han *et al.* use effective-capacity analysis with joint power and rate allocation for delay-sensitive links [7]. In parallel, *Lyapunov-guided* methods map delay or stability constraints into queue-based penalties that steer scheduling and offloading decisions; more recent work integrates Lyapunov signals into learning architectures to enforce delay bounds under time-varying traffic [8]. These studies support our choice to include stability and load-awareness components in the reward, even though we do not model explicit queue dynamics.

E. Learning to Optimize versus Deep Reinforcement Learning

Learning-to-optimize approaches approximate optimization mappings — WMMSE-like solutions, for instance — using supervised learning, achieving near-optimal allocations with orders-of-magnitude speedups. In massive MIMO, PowerNet learns pilot and data power from large-scale fading with real-time inference under varying user activity [5]. DRL and multi-agent DRL (MADRL), by contrast, target partially observed and model-mismatched settings. Nasir and Guo demonstrated the feasibility of multi-agent power control with strong rate performance under partial CSI [9]; follow-up work has shown RL-based network tuning that respects NR procedures and configuration rules [10], while recent surveys identify MARL as a key candidate for cell-free and XL-MIMO resource control [11]. In many of these studies, however, fairness, EE, and QoS remain implicit or only weakly enforced in the reward or the constraints. Our framework addresses this gap by embedding all three aspects directly into the scalar reward optimized by PPO and SAC.

F. Percentile and Constraint-Aware Objectives

Beyond PF and max–min utilities, *percentile* objectives have been proposed to unify worst-tail performance — fifth-percentile throughput, for example — with power control in a convex-leaning signal-processing framework [12]. Such objectives are attractive for learned controllers because they map naturally onto robust, tail-aware reward functions. Although our present experiments focus on mean-rate and fairness-aware objectives, the percentile-based formulations of [12] provide a direct path for extending the Smart-MIMO-Cell-Env reward design toward explicit tail-performance guarantees.

III. SYSTEM MODEL AND MDP FORMULATION

This section describes the multi-cell massive MIMO topology, the channel and traffic models, and the Markov decision process (MDP) that underpins the Smart-MIMO-Cell-Env used by the RL agent.

A. Multi-Cell Massive MIMO Topology

The overall multi-cell layout is illustrated in Fig. 1. We consider a downlink multi-cell massive MIMO network with N hexagonal cells. Each cell $l \in \{1, \dots, N\}$ contains a base station (BS) equipped with M antennas that simultaneously serves K single-antenna user equipments (UEs) on the same time–frequency resource. In the experiments we fix $N = 4$, $M = 16$, and $K = 6$, yielding $NK = 24$ users in total.

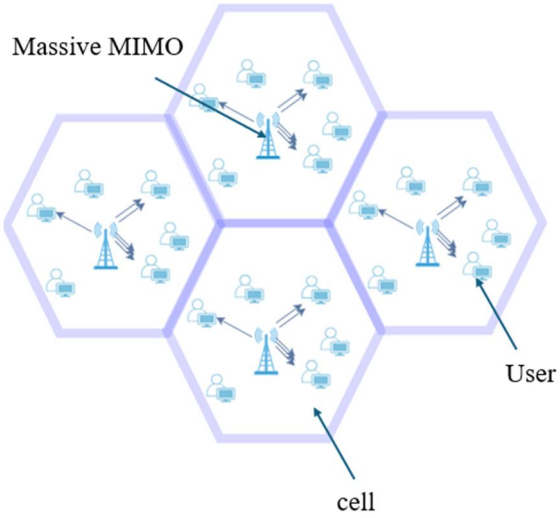


Fig. 1 Illustration of the considered multi-cell massive MIMO topology with N hexagonal cells. Each cell contains one multi-antenna BS serving multiple single-antenna UEs on the same time–frequency resource. The arrows indicate concurrent downlink transmissions and highlight both the intra-cell and the inter-cell interference experienced by the users.

Let $\mathbf{h}_{l,k} \in \mathbb{C}^M$ denote the small-scale fading channel between BS l and user k in that cell, and let $x_{l,k}$ be the data symbol intended for that user. The effective per-UE downlink transmit power, obtained after applying the learned array-gain commands, is denoted $p_{l,k}^{\text{eff}}$. With linear precoding, the received signal at user k in cell l is

$$y_{l,k} = \sqrt{p_{l,k}^{\text{eff}}}\,\mathbf{h}_{l,k}^H \mathbf{w}_{l,k} x_{l,k} + \sum_{j \neq k} \sqrt{p_{l,j}^{\text{eff}}}\,\mathbf{h}_{l,k}^H \mathbf{w}_{l,j} x_{l,j} + \sum_{i \neq l} \sum_{u=1}^{K} \sqrt{p_{i,u}^{\text{eff}}}\,\mathbf{h}_{i,l,k}^H \mathbf{w}_{i,u} x_{i,u} + n_{l,k}, \quad (1)$$

where $\mathbf{w}_{l,k}$ is the precoding vector used by BS l for UE k, $\mathbf{h}_{i,l,k}$ is the channel from BS i to UE k in cell l, and $n_{l,k}$ is circularly symmetric complex Gaussian noise with variance σ^2. The second term in (1) represents intra-cell interference and the third term represents inter-cell interference.

Zero-forcing (ZF) precoding is employed independently at each BS. Denoting the $M \times K$ channel matrix from BS l to its K associated UEs by $\mathbf{H}_l = [\mathbf{h}_{l,1}, \ldots, \mathbf{h}_{l,K}]$, the unnormalized ZF precoder is

$$\mathbf{W}_l = \mathbf{H}_l (\mathbf{H}_l^H \mathbf{H}_l)^{-1}, \quad (2)$$

and a diagonal normalization matrix $\mathbf{D}_l$ is applied to enforce the per-BS or per-antenna power constraint,

$$\mathbf{W}_l = \mathbf{H}_l (\mathbf{H}_l^H \mathbf{H}_l)^{-1} \mathbf{D}_l, \quad (3)$$

with the k-th column of $\mathbf{W}_l$ equal to $\mathbf{w}_{l,k}$.

Using (1), the instantaneous signal-to-interference-plus-noise ratio (SINR) of UE k in cell l is

$$\gamma_{l,k} = \frac{p_{l,k}^{\text{eff}} \left|\mathbf{h}_{l,k}^H \mathbf{w}_{l,k}\right|^2}{\sum_{j \neq k} p_{l,j}^{\text{eff}} \left|\mathbf{h}_{l,k}^H \mathbf{w}_{l,j}\right|^2 + \sum_{i \neq l} \sum_{u=1}^{K} p_{i,u}^{\text{eff}} \left|\mathbf{h}_{i,l,k}^H \mathbf{w}_{i,u}\right|^2 + \sigma^2}, \quad (4)$$

and the corresponding achievable rate is

$$R_{l,k} = \log_2(1 + \gamma_{l,k}) \quad \text{[bits/s/Hz]}. \quad (5)$$

The network-wide mean rate at time step t is

$$\bar{R}_t = \frac{1}{NK} \sum_{l=1}^{N} \sum_{k=1}^{K} R_{l,k}(t). \quad (6)$$

B. Channel, Large-Scale Fading, and Dynamic Features

The physical layer in Smart-MIMO-Cell-Env follows the local-scattering model with 3GPP-inspired large-scale fading. For each BS–UE pair, a mean angle of arrival and an angular standard deviation are randomly drawn, and the spatial correlation matrix $\mathbf{R}_{l,k} \in \mathbb{C}^{M \times M}$ is generated using a one-ring model with half-wavelength antenna spacing. The large-scale channel gain combines distance-dependent path loss with lognormal shadowing, and the resulting gains are normalized by the thermal-noise level of a 10 MHz receiver to obtain realistic SNRs. The small-scale channel vectors are constructed as

$$\mathbf{h}_{l,k} = \mathbf{R}_{l,k}^{1/2}\mathbf{g}_{l,k}, \qquad \mathbf{g}_{l,k} \sim \mathcal{CN}(\mathbf{0}, \mathbf{I}_M), \tag{7}$$

and $N_s = 6$ independent realizations are averaged per time step to compute rates and SINRs.

To emulate realistic time-varying conditions, several dynamic features are updated at each step:

- *Mobility:* UE positions evolve according to a random-walk model inside each cell. The average displacement magnitude across all UEs defines a normalized mobility factor $m_t \in [0,1]$.
- *Time-varying traffic:* Each UE's traffic demand follows a sinusoidal baseline with bounded Gaussian perturbations. The aggregate normalized traffic load is denoted $\ell_t \in [0,1]$, and it scales the achieved rates to reflect demand satisfaction.
- *Channel aging:* Because of mobility and feedback delay, the effective channel is gradually degraded by an aging factor $a_t \in [0,1]$ that scales the useful signal component and is reset when the channel is regenerated.
- *Interference coordination:* An interference-coordination factor reduces inter-cell interference, allowing the environment to emulate schemes such as ICIC and CoMP. This feature is enabled in the main experiments.

These quantities are computed internally by Smart-MIMO-Cell-Env and exposed to the RL agent through the state vector.

C. Markov Decision Process Formulation

The downlink power-control problem is cast as a single-agent MDP in which a centralized controller jointly selects per-UE power levels and per-antenna gain coefficients at each decision step. One instance of Smart-MIMO-Cell-Env represents the N-cell network, and a single RL agent interacts with this environment.

1) State Space

At time step t the environment emits an eight-dimensional state vector

$$\mathbf{s}_t = \left[P_{t,\mathrm{dBm}}^{\mathrm{eff}},\ g_t^{\mathrm{agg}},\ \overline{\mathrm{SINR}}_t,\ \bar{R}_t,\ \mathcal{I}_t,\ m_t,\ \ell_t,\ a_t\right], \tag{8}$$

with the following components:

- $P_{t,\mathrm{dBm}}^{\mathrm{eff}}$: mean effective downlink transmit power across all UEs in dBm, after applying the learned array-gain commands.
- g_t^{agg}: average array-gain factor across BS antennas, summarizing the current per-antenna gain pattern.
- $\overline{\mathrm{SINR}}_t$: average SINR across all UEs and cells.
- $\bar{R}_t$: network-wide mean downlink rate given in (6).
- $\mathcal{I}_t$: *link-quality penalty (inverse-SINR proxy)*, defined as a clipped inverse of the mean SINR, i.e., $\mathcal{I}_t \approx \mathrm{clip}\left(1/\overline{\mathrm{SINR}}_t\right)$.
- $m_t \in [0,1]$: normalized mobility factor summarizing the average UE movement between steps.
- $\ell_t \in [0,1]$: normalized traffic-load indicator capturing the average demand.
- $a_t \in [0,1]$: channel-age variable quantifying how outdated the CSI has become.

This compact representation exposes both fast link-quality indicators (SINR, rate, inverse-SINR penalty) and slower dynamics (mobility, traffic, aging), enabling the agent to learn policies that anticipate how present actions affect future performance.

2) Action Space

The action space is a high-dimensional continuous Box. At each step the agent chooses a single vector that jointly specifies *per-UE transmit powers* and *per-antenna gain commands*:

$$\mathbf{a}_t = [\mathbf{p}_t^{\mathrm{UE}},\ \mathbf{g}_t^{\mathrm{ant}}] \in \mathbb{R}^{NK+NM}, \tag{9}$$

where

$$\mathbf{p}_t^{\mathrm{UE}} = \left[p_{t,1}, \ldots, p_{t,NK}\right]^{\mathrm{T}}, \qquad p_{t,u} \in [P_{\min}, P_{\max}], \tag{10}$$

$$\mathbf{g}_t^{\mathrm{ant}} = \left[g_{t,1}, \ldots, g_{t,NM}\right]^{\mathrm{T}}, \qquad g_{t,a} \in [g_{\min}, g_{\max}], \tag{11}$$

denote the per-UE power commands in dBm and the per-antenna gain commands in linear scale, respectively. In the final configuration we use $P_{\min} = -20$ dBm, $P_{\max} = 23$ dBm, $g_{\min} = 0.5$, and $g_{\max} = 2.0$, yielding an 88-dimensional action space, since $NK + NM = 4 \times 6 + 4 \times 16 = 88$.

Inside the environment, the per-UE power commands are converted from dBm to Watts and reshaped into $p_{l,k}(t)$ for $l \in \{1, \ldots, N\}$ and $k \in \{1, \ldots, K\}$. Likewise, the gain commands are reshaped into $g_{l,m}(t)$ for $m \in \{1, \ldots, M\}$. Here $g_{l,m}(t)$ is a linear *amplitude* gain applied at the transmit chain of antenna m at BS l, so the radiated power contribution scales with $g_{l,m}^2(t)$.

To connect the per-antenna gains to the downlink power used in the SINR and rate evaluation, let $\mathbf{w}_{l,k}(t) \in \mathbb{C}^M$ denote the unit-norm precoding vector for stream (l,k), i.e., $\| \mathbf{w}_{l,k}(t) \|_2^2 = 1$, with entries $w_{l,k,m}(t)$.

The gain vector is applied through the diagonal matrix $\mathbf{G}_l(t) = \mathrm{diag}\left(g_{l,1}(t), \ldots, g_{l,M}(t)\right)$ to obtain the gain-weighted beamformer $\tilde{\mathbf{w}}_{l,k}(t) = \mathbf{G}_l(t)\mathbf{w}_{l,k}(t)$. Accordingly, the *effective radiated power* allocated to stream (l,k) is defined as

$$p_{l,k}^{\mathrm{eff}}(t) = p_{l,k}(t) \parallel \tilde{\mathbf{w}}_{l,k}(t) \parallel_2^2 = p_{l,k}(t) \sum_{m=1}^{M} g_{l,m}^2(t) \left| w_{l,k,m}(t) \right|^2, \quad (12)$$

which reduces to $p_{l,k}^{\mathrm{eff}}(t) = p_{l,k}(t)$ when all antenna gains are unity. The effective power matrix $\{p_{l,k}^{\mathrm{eff}}(t)\}$ is then used consistently in the SINR computation and the rate evaluation.

3) Reward Function

The reward at time t aggregates several objectives: spectral efficiency, energy efficiency, link quality, robustness to dynamic conditions, fairness across cells, and temporal stability of the power profile. Its structure is

$$r_t = \bar{R}_t - \alpha_P P_t^{\mathrm{tot}} - \alpha_I \mathcal{I}_t - \alpha_M m_t + \alpha_L \ell_t - \alpha_A a_t + \beta_F \Phi_{\mathrm{fair}}(t) + \beta_S \Phi_{\mathrm{stab}}(t), \quad (13)$$

where P_t^{tot} is the total effective transmit power computed from the per-stream effective powers $p_{l,k}^{\mathrm{eff}}(t)$ of (12), namely

$$P_t^{\mathrm{tot}} = \sum_{l=1}^{N} \sum_{k=1}^{K} p_{l,k}^{\mathrm{eff}}(t), \quad (14)$$

while $\alpha > 0$ and $\beta \geq 0$ are design weights, and $\Phi_{\mathrm{fair}}(t)$ and $\Phi_{\mathrm{stab}}(t)$ are fairness and stability bonuses. The term $-\alpha_I \mathcal{I}_t$ penalizes degraded link quality through the inverse-SINR proxy $\mathcal{I}_t$ rather than physical interference power, so larger values of $\mathcal{I}_t$ correspond to worse average SINR. The fairness term penalizes the variance of recent per-cell mean rates, encouraging balanced performance across cells, while the stability term rewards smooth power trajectories by penalizing large short-term fluctuations of the effective transmit power. The coefficients are tuned to keep the reward numerically well scaled while reflecting the energy-efficient, link-quality-aware, and QoS-aware design goals of the framework.

4) RL Objective

The agent seeks a stochastic policy $\pi(\mathbf{a} \mid \mathbf{s})$ that maximizes the expected discounted return

$$J(\pi) = \mathbb{E}_{\tau \sim \pi}\left[\sum_{t=0}^{T-1} \gamma^t r_t\right], \quad (15)$$

where $\gamma \in (0,1)$ is the discount factor, T is the episode horizon (100 steps in our experiments), and τ denotes a trajectory of states, actions, and rewards generated by running π in Smart-MIMO-Cell-Env.

The overall closed-loop interaction between the massive MIMO environment, the state representation, the RL agent, and the reward computation is summarized in Fig. 2.

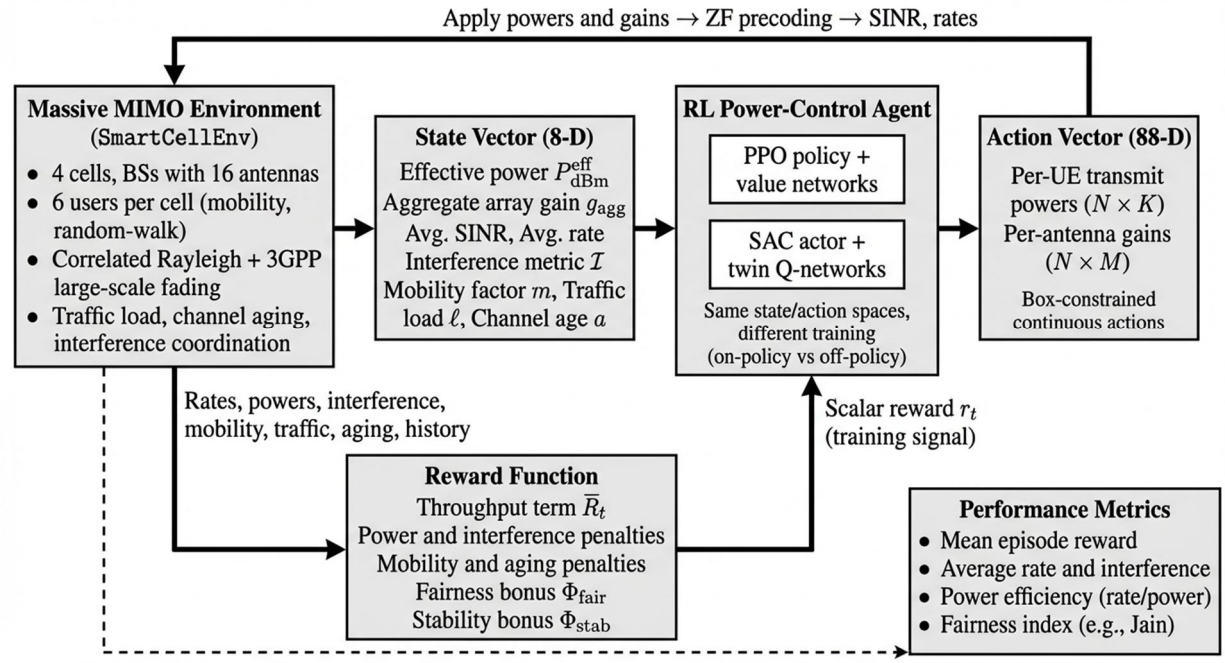


Fig. 2 System-level block diagram of the Smart-MIMO-Cell-Env power-control loop. The centralized RL agent (PPO or SAC) maps the eight-dimensional state to an 88-dimensional action vector of per-UE powers and per-antenna gains; the environment applies ZF precoding, computes SINR and rates, and returns the scalar reward.

IV. REINFORCEMENT LEARNING ALGORITHMS AND CONFIGURATIONS

We employ two state-of-the-art model-free DRL algorithms for continuous control, PPO and SAC, and apply them to the Smart-MIMO-Cell-Env introduced in Section III. Both algorithms are implemented in Ray RLlib with a PyTorch backend and interact with the *same* environment, state representation, and action space, so that the observed differences in performance stem from their update rules and hyperparameters rather than from the problem definition.

A. Common Environment and Hardware Configuration

PPO and SAC share the following configuration:

- **Environment:** both use the registered ID Smart-MIMO-Cell dynamics, which instantiates the enhanced Smart-MIMO-Cell-Env with $N = 4$ cells, $M = 16$ antennas per BS, $K = 6$ users per cell, $N_s = 6$ channel snapshots per step, and all dynamic features (mobility, traffic patterns, channel aging, and interference coordination) enabled.
- **Observation and action spaces:** both agents observe the eight-dimensional state vector of (8) and output actions in the same continuous Box space of dimension $NK + NM = 88$, composed of per-UE power commands and per-antenna gain commands.
- **Hardware-aware layout:** a shared hardware-adaptation routine configures the number of rollout workers, the number of environments per worker, and GPU usage based on the detected device. PPO and SAC are therefore trained under identical computational budgets.
- **Network architecture:** all policy, value, and Q-networks are fully connected multilayer perceptrons with two hidden layers of 256 units and ReLU activations, specified through the RLlib model configuration.

This common setup isolates algorithmic effects: any performance gap between PPO and SAC arises from how the two methods process trajectories and update their parameters, not from differences in the environment, the state and action design, or the hardware configuration.

B. Proximal Policy Optimization

PPO is an on-policy policy-gradient method that stabilizes learning by constraining the deviation between successive policies. At iteration k, PPO collects fresh trajectories using the current policy π_{θ_k}, computes advantage estimates from those rollouts, and then updates θ by maximizing a clipped surrogate objective.

1) Policy and Value Networks

The PPO policy $\pi_\theta(\mathbf{a} \mid \mathbf{s})$ is modelled as a multivariate Gaussian distribution over the continuous 88-dimensional action vector:

$$\pi_\theta(\mathbf{a} \mid \mathbf{s}) = \mathcal{N}\left(\boldsymbol{\mu}_\theta(\mathbf{s}),\ \mathrm{diag}(\boldsymbol{\sigma}_\theta^2)\right),$$

where $\boldsymbol{\mu}_\theta(\mathbf{s}) \in \mathbb{R}^{88}$ and $\boldsymbol{\sigma}_\theta \in \mathbb{R}^{88}$ are produced by the policy network. RLlib uses a state-independent log-standard-deviation head (free_log_std = True), which simplifies optimization in the high-dimensional action space. A separate value network $V_\phi(\mathbf{s})$ with the same backbone architecture estimates the state-value function.

2) PPO Loss and Hyperparameters

Let $r_t(\theta) = \pi_\theta(\mathbf{a}_t \mid \mathbf{s}_t)/\pi_{\theta_{\text{old}}}(\mathbf{a}_t \mid \mathbf{s}_t)$ denote the probability ratio between the new and the old policy, and let $\hat{A}_t$ be the advantage estimate at time t computed with generalized advantage estimation (GAE). The clipped surrogate objective is

$$L_{\text{clip}}(\theta) = \mathbb{E}_t\left[\min\left(r_t(\theta)\hat{A}_t,\ \mathrm{clip}(r_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t\right)\right], \tag{16}$$

with clipping parameter $\epsilon = 0.2$. The total PPO loss combines policy, value-function, and entropy terms in the standard RLlib formulation. The main hyperparameters used in our PPO experiments are: discount factor $\gamma = 0.99$; GAE parameter $\lambda = 0.95$; learning rate 3×10^{-4} with Adam; train_batch_size = 4096; sgd_minibatch_size = 256; num_sgd_iter = 10; entropy_coeff = 0.01; vf_loss_coeff = 0.5; and vf_clip_param = 10.0.

Because PPO is strictly on-policy, each update is based on fresh trajectories generated by the *current* policy under the *current* mobility, traffic, and aging conditions. This aligns well with the slowly drifting statistics of Smart-MIMO-Cell-Env.

C. Soft Actor-Critic

SAC is an off-policy, entropy-regularized actor–critic algorithm designed for sample-efficient learning in continuous control problems. It maintains a stochastic policy, two Q-networks, and the corresponding target networks, and updates them using transitions sampled from a replay buffer.

1) SAC Objective and Critic Updates

The soft Q-functions $Q_{\psi_1}(\mathbf{s},\mathbf{a})$ and $Q_{\psi_2}(\mathbf{s},\mathbf{a})$ are trained to minimize the soft Bellman residual

$$L_{\mathrm{Q}}(\psi_i) = \mathbb{E}_{(\mathbf{s},\mathbf{a},r,\mathbf{s}')}\left[(Q_{\psi_i}(\mathbf{s},\mathbf{a}) - y)^2\right], \tag{17}$$

with target

$$y = r + \gamma\, \mathbb{E}_{\mathbf{a}'\sim\pi_\theta(\cdot|\mathbf{s}')}\left[\min_{j=1,2} Q_{\bar{\psi}_j}(\mathbf{s}',\mathbf{a}') - \alpha \log \pi_\theta(\mathbf{a}' \mid \mathbf{s}')\right],$$

where $\bar{\psi}_j$ denote the parameters of the target networks, updated by soft target updates with coefficient $\tau = 0.005$. Twin Q-networks (twin_q = True) reduce overestimation bias.

2) Policy, Entropy, and Replay Buffer

The SAC policy $\pi_\theta(\mathbf{a} \mid \mathbf{s})$ is also a Gaussian distribution over the 88-dimensional action space, parameterized by a network with two hidden layers of 256 units. Actions are squashed to the valid Box range. The configuration employs automatic entropy tuning with target_entropy = "auto", so that the temperature α adapts to maintain an appropriate degree of stochasticity throughout training. SAC uses a prioritized replay buffer with capacity up to 10^6 transitions, depending on GPU memory. Transitions from all past policies are stored and sampled in mini-batches of size 2048 for the critic and actor updates. The main SAC hyperparameters in our experiments are: discount factor $\gamma = 0.99$; learning rate 3×10^{-4} with Adam; soft target-update rate $\tau = 0.005$; train_batch_size = 4096; a prioritized replay buffer with exponent $\alpha_{\mathrm{PR}} = 0.6$ and importance-sampling exponent $\beta_{\mathrm{PR}} = 0.4$; and n-step returns with $n = 1$, i.e., one-step TD targets.

Unlike PPO, SAC is off-policy: its critic and actor updates are based on a mixture of past and recent experience drawn from the replay buffer. In a non-stationary environment such as Smart-MIMO-Cell-Env, this time-mixing causes SAC to regress against a moving target distribution that blends older mobility, traffic, and aging conditions with the current ones.

D. Qualitative Algorithmic Differences in Smart-MIMO-Cell-Env

From the perspective of Smart-MIMO-Cell-Env, the main differences between PPO and SAC can be summarized as follows:

1) On-policy versus off-policy: PPO uses large, fresh batches drawn from the current environment distribution, whereas SAC trains on a replay buffer that mixes old and new conditions. This makes PPO naturally better aligned with the slowly drifting Smart-MIMO-Cell dynamics.
2) Update stability in an 88-dimensional action space: the clipped surrogate loss of PPO constrains policy updates, preventing catastrophic jumps in the per-UE power and per-antenna gain profiles. SAC has no analogous clipping mechanism and instead relies on learning-rate tuning and critic accuracy for stability.
3) Approximate Markov structure: the GAE-based updates of PPO do not require exact Bellman consistency, whereas the critics of SAC do. Because the reward includes history-based terms (the fairness and stability bonuses), the process is only approximately Markovian, which further favours PPO.
4) Exploration behaviour: the entropy-regularized objective of SAC leads to more aggressive exploration, especially early in training. In the Smart-MIMO-Cell setting, many exploratory actions correspond to very poor SINR and high interference, which increases variance and occasionally degrades the training curve relative to PPO.

These qualitative differences are validated quantitatively by the experimental results in Section V. The corresponding training pipelines are summarized in Fig. 3, where both algorithms interact with the same Smart-MIMO-Cell-Env but use fundamentally different data flows and update mechanisms.

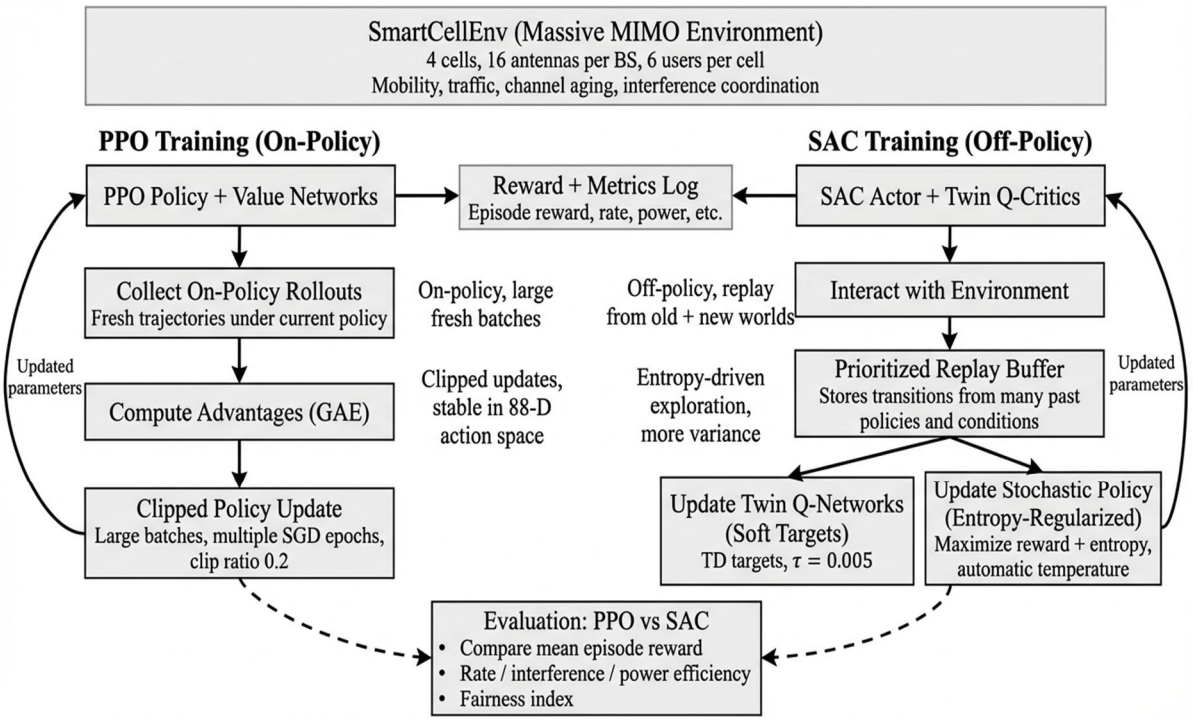


Fig. 3 On-policy PPO versus off-policy SAC training workflows on Smart-MIMO-Cell-Env. PPO uses fresh rollouts with clipped policy and value updates; SAC uses a prioritized replay buffer, twin Q-networks with soft targets, and an entropy-regularized actor. Both interact with the same environment and logged metrics.

V. EXPERIMENTAL RESULTS AND COMPARATIVE ANALYSIS

This section compares PPO and SAC on the multi-cell massive MIMO power-control task implemented in Smart-MIMO-Cell-Env. Both algorithms use the configurations of Section IV, share the same environment parameters ($N = 4$, $M = 16$, $K = 6$, dynamic mobility, traffic and aging, and interference coordination), and are trained under the same software and hardware setup. PPO is trained for 250 iterations and SAC for 275 iterations. The final policies are evaluated over 200 independent test episodes with exploration noise disabled.

A. Training Behaviour

Fig. 4 and Fig. 5 show the evolution of the mean episode reward against the training iteration for PPO and SAC, respectively.

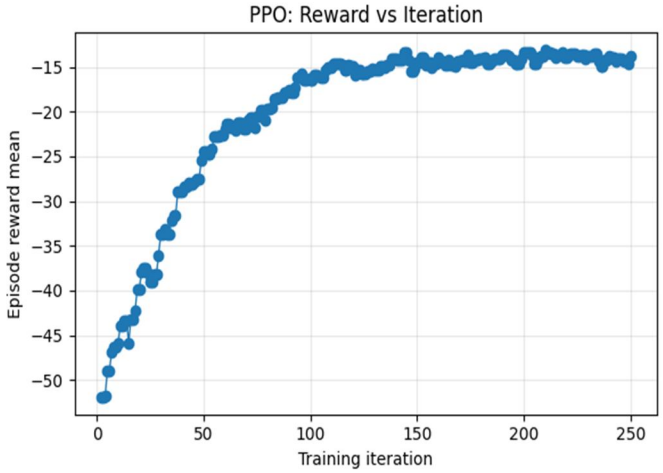


Fig. 4 PPO training on Smart-MIMO-Cell-Env: mean episode reward versus training iteration.

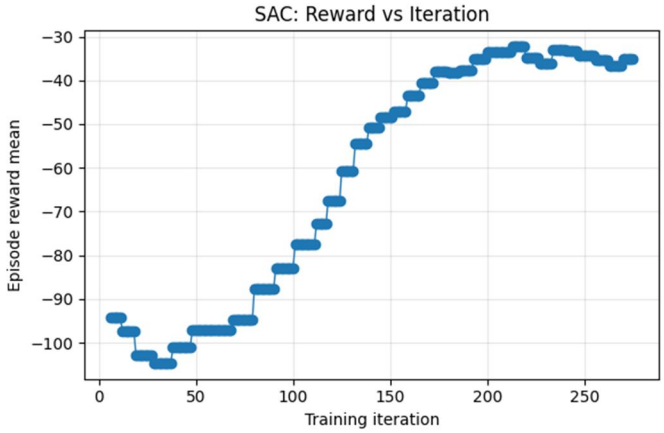


Fig. 5 SAC training on Smart-MIMO-Cell-Env: mean episode reward versus training iteration.

PPO exhibits a smooth and largely monotonic improvement: starting from a highly negative regime that corresponds to uncoordinated per-UE powers and per-antenna gains, the mean episode reward increases steadily and then stabilizes on a relatively flat plateau with only small fluctuations. This indicates that PPO finds a robust policy that consistently balances rate, power, and interference under the slowly drifting mobility, traffic-load, and channel-aging dynamics of Smart-MIMO-Cell-Env.

SAC also improves substantially over its initial policy. Its reward curve rises from a very low starting regime to a much better region, showing that SAC is able to learn a non-trivial power-control strategy despite the non-stationary and partially history-dependent environment. The SAC curve is, however, noticeably noisier and exhibits plateaus and mild regressions in later iterations.

B. Policy Quality and Evaluation Metrics

To assess the policies in a stationary regime, we evaluate the final PPO and SAC policies over 200 independent test episodes. For each episode we record the cumulative reward, the mean downlink rate, the inverse-SINR penalty $\bar{\mathcal{I}}$, the mean effective transmit power $\bar{P}_{\text{eff}}$ (the episode average of P_t^{tot} from (14)), a power-efficiency indicator defined as rate divided by effective power, and the mean array gain. The resulting averages are summarized in Table I.

Two conventions should be kept in mind when reading Table I. First, because the reward of (13) aggregates several penalty terms, its values are negative and *less negative is better*. Second, $\bar{R}$ is the traffic-scaled mean per-user rate reported by the environment: the physical rate of (5) is expressed in bits/s/Hz, but it is multiplied by the normalized traffic-load factor $\ell_t \in [0,1]$ to reflect demand satisfaction, so the reported values are normalized rather than absolute link rates. Power is reported in Watts, and the power-efficiency indicator PE consequently has units of normalized rate per Watt. The metric $\bar{\mathcal{I}}$ is the clipped inverse of the mean SINR and is therefore dimensionless, with larger values indicating worse link quality.

TABLE I

EVALUATION OVER 200 TEST EPISODES WITH EXPLORATION DISABLED

Mean episode return, mean downlink rate $\bar{R}$, inverse-SINR penalty $\bar{\mathcal{I}} \approx \text{clip}\left(1/\overline{\text{SINR}}\right)$, mean effective transmit power $\bar{P}_{\text{eff}}$, power efficiency (PE), and mean array gain for PPO and SAC.

Alg.	$\mathbb{E}[R_{\text{ep}}]$ (reward)	$\bar{R}$ (rate)	$\bar{\mathcal{I}}$ (inv.-SINR pen.)	$\bar{P}_{\text{eff}}$ (W)	PE (rate/W)	$\bar{g}_{\text{array}}$
PPO	−6.130	0.264	1.714	1.37×10^{-1}	1.973	1.154
SAC	−108.724	0.047	24.032	8.99×10^{-3}	5.408	1.312

Several observations follow from Table I.

1) Overall reward. PPO attains a mean episode reward of -6.130, whereas SAC settles around -108.724. The training plateau of PPO therefore corresponds to a policy that is far better aligned with the multi-objective reward than the one found by SAC.

2) Throughput versus link-quality penalty. PPO achieves a mean rate of 0.264 with a small inverse-SINR penalty $\bar{\mathcal{I}} = 1.714$, whereas SAC reaches 0.047 with a much larger penalty $\bar{\mathcal{I}} = 24.032$. Because $\bar{\mathcal{I}}$ is a clipped inverse of the mean SINR, larger values indicate substantially worse average SINR — that is, poorer link conditions — rather than higher measured interference power. PPO therefore delivers higher throughput while maintaining much better link quality.

3) Power and power efficiency. PPO operates at a higher mean effective power, $\bar{P}_{\text{eff}} \approx 1.37 \times 10^{-1}$ W against 8.99×10^{-3} W for SAC, but converts that power into far more useful rate, which yields a much better reward despite a moderate power penalty. SAC achieves a larger rate-per-Watt (PE ≈ 5.408), but only at the cost of low absolute throughput and poor link quality, which is reflected in its far more negative overall reward. This illustrates a well-known pitfall of ratio-based efficiency metrics: a high rate-per-Watt value is not by itself evidence of a useful operating point.

4) Array gain. Both policies operate around a mean array-gain statistic close to unity, with SAC showing a slightly higher value ($\bar{g}_{\text{array}} \approx 1.312$). Most of the performance gap is therefore driven by how the algorithms coordinate power and interference, rather than by extreme array-gain configurations.

While the training curve of SAC improves to a substantially less negative regime, its final policy performs poorly when evaluated with exploration disabled. This points to a large train–test mismatch in our setup rather than to a failure to learn, a point we return to in Section V-D.

C. State Feature Importance: Correlation and Sensitivity

To understand how the agents use the eight state components of (8), we perform two complementary analyses on the trained policies.

1) A *feature–reward correlation* study, in which we log $\mathbf{s}_t$ and r_t over the evaluation episodes and compute the Pearson correlation between each feature and the immediate reward. The absolute correlations are normalized to sum to one, giving an importance score.

2) A *per-feature freezing* study, in which we re-run the evaluation episodes while holding a single state component fixed at its baseline mean value, and compute the change in mean episode reward $\Delta\text{return} = \bar{R}_{\text{ep}}^{\text{baseline}} - \bar{R}_{\text{ep}}^{\text{frozen}}$.

For PPO, the resulting feature–reward correlations and normalized importances are visualized in Fig. 6(a) and Fig. 7(a); the same analysis for SAC is shown in Fig. 6(b) and Fig. 7(b). The corresponding numbers are summarized in Table II, which combines the correlation, importance, and freezing results for both algorithms.

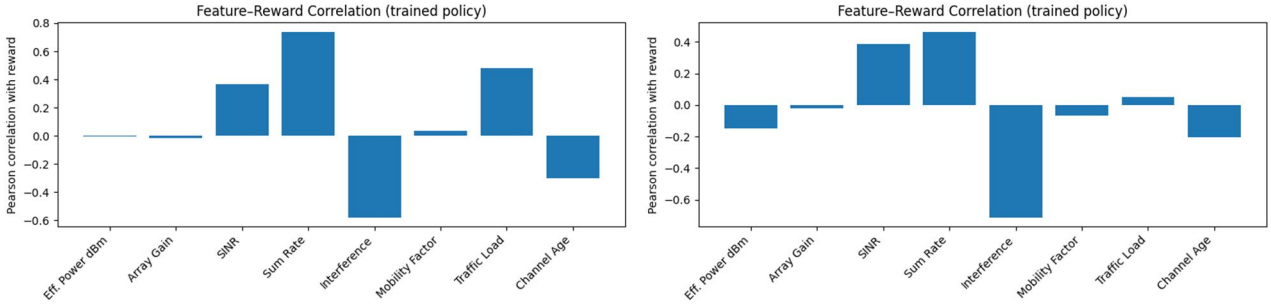


Fig. 6 Pearson correlation between each state feature and the immediate reward for (a) PPO, left, and (b) SAC, right.

TABLE II
FEATURE-LEVEL ANALYSIS FOR PPO AND SAC

For each state component: Pearson correlation ρ with the immediate reward, normalized absolute importance (Imp.), and the drop in mean episode reward when that feature is frozen (Δreturn).

Feature	ρ (PPO)	Imp. (PPO)	Δreturn (PPO)	ρ (SAC)	Imp. (SAC)	Δreturn (SAC)
Eff. power (dBm)	−0.0050	0.0020	0.323	−0.1467	0.0713	7.208
Array gain	−0.0195	0.0077	−0.507	−0.0222	0.0108	1.068
SINR	0.3649	0.1447	0.084	0.3873	0.1882	2.682
Sum rate	0.7378	0.2925	0.660	0.4641	0.2256	2.120
Inverse-SINR penalty	−0.5804	0.2301	−0.165	−0.7122	0.3462	34.959
Mobility factor	0.0320	0.0127	0.332	−0.0689	0.0335	0.557
Traffic load	0.4820	0.1910	0.775	0.0519	0.0252	4.167
Channel age	−0.3010	0.1193	0.301	−0.2040	0.0992	1.765

The correlation and importance columns reveal a consistent picture across both algorithms.

1) Mean rate and SINR are strongly aligned with reward. For PPO, the mean rate has the largest positive correlation ($\rho \approx 0.738$, importance ≈ 0.293), followed by traffic load and SINR. For SAC, mean rate and SINR are also among the top contributors (importance ≈ 0.226 and ≈ 0.188). This directly reflects the $\bar{R}_t$ term of the reward.

2) The inverse-SINR penalty is strongly negatively correlated. In both cases it has one of the largest negative correlations with reward in magnitude (PPO: $\rho \approx -0.580$; SAC: $\rho \approx -0.712$), consistent with the explicit penalty in the reward design.

3) Power, mobility, traffic, and aging provide contextual signals. Effective power and channel age show negative correlations, while mobility and traffic carry weaker but non-negligible effects. Together they encode how dynamic and how challenging the current operating point is.

4) The freezing experiment adds a second perspective.

5) For SAC, freezing the inverse-SINR penalty causes by far the largest performance drop (Δreturn ≈ 34.96), confirming that SAC relies heavily on this feature when selecting actions. Freezing the effective power and the traffic load also produces noticeable drops (Δreturn ≈ 7.21 and ≈ 4.17), while freezing SINR and mean rate yields smaller but still measurable degradations.

6) For PPO, the largest positive Δreturn values correspond to traffic load and mean rate (≈ 0.78 and ≈ 0.66), meaning that freezing these features makes the reward noticeably worse. Freezing the inverse-SINR or array-gain feature yields slightly negative Δreturn values that lie within the observed variability, indicating that freezing them does not consistently degrade performance in this test.

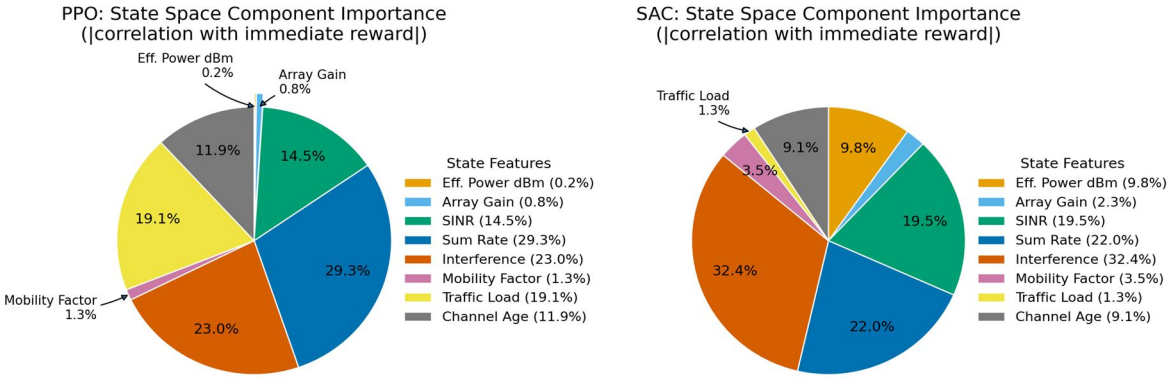


Fig. 7 Normalized absolute feature–reward correlations for (a) PPO, left, and (b) SAC, right, showing the relative importance of each state component.

Overall, the feature analysis answers the question *"when the agent is doing well, which parts of the state tend to look better?"* as follows: high-reward steps are associated with higher mean rate and SINR, lower values of the inverse-SINR penalty and of the effective power, and more benign mobility and channel-aging conditions. SAC leans more heavily on the inverse-SINR penalty, whereas PPO makes somewhat broader use of traffic and rate information. It is worth stressing that these correlations describe how the state and the reward co-vary along the trajectories visited by each policy; they are diagnostic rather than causal, and the freezing study is the more direct test of how much each feature actually drives the control decisions.

D. Why PPO Outperforms SAC in Smart-MIMO-Cell-Env

The empirical findings above are consistent with the algorithmic differences discussed in Section IV.

1) On-policy versus off-policy under drift. The environment distribution drifts because of mobility, time-varying traffic, and channel aging. PPO is on-policy and always trains on trajectories generated by the current policy under the current dynamics, which naturally tracks this drift. SAC trains on a replay buffer that mixes samples from many past policies and conditions, effectively regressing toward a moving average of old and new worlds; this mismatch manifests as noisier learning and, in our evaluation, a much poorer final return.

2) Clipped updates in an 88-dimensional action space. The action vector jointly controls per-UE powers and per-antenna gains. The clipped surrogate loss of PPO constrains how far the policy can move per update, preventing catastrophic jumps that would cause severe SINR degradation — a high inverse-SINR penalty — or rate collapse. SAC has no analogous clipping and must rely on critic accuracy and hyperparameter choices for stability.

3) Complex, partially history-dependent reward. The reward combines instantaneous terms (rate, power, link quality) with fairness and stability bonuses that depend on short histories of rate and power. This breaks exact Markovianity. The GAE-based policy gradient of PPO is comparatively tolerant to approximation in the value baseline, whereas the Q-functions of SAC are more sensitive to critic bias under such model mismatch.

4) Exploration behaviour versus deterministic evaluation. The entropy-regularized objective of SAC promotes persistent stochasticity. When evaluation is performed with exploration disabled, the resulting deterministic action selection can yield a policy that performs poorly under the same state distribution, which is consistent with the large gap between the SAC training curve and its evaluation statistics in Table I. PPO uses a moderate entropy coefficient and large, fresh batches, leading to more controlled exploration and to a policy that converges to a much better trade-off between throughput, link quality, and power.

In summary, while both algorithms learn non-trivial policies in this challenging massive MIMO power-control task, PPO is better matched to the non-stationary, high-dimensional, and history-dependent nature of Smart-MIMO-Cell-Env, and achieves substantially higher final reward and throughput than SAC under the same experimental conditions.

E. Limitations

Several limitations should be kept in mind when interpreting these results. First, the comparison is made under a single shared training protocol and one hyperparameter configuration per algorithm; the reported gap therefore characterizes PPO and SAC *as configured here*, and a dedicated tuning effort — in particular of the SAC entropy target, replay-buffer horizon, and evaluation-time action selection — could narrow it. Second, the evaluation uses a single environment seed family and reports means over 200 episodes without confidence intervals, so small differences, such as the negative Δreturn values observed for PPO, should not be over-interpreted. Third, Smart-MIMO-Cell-Env is a 3GPP-inspired but simplified simulator: it does not model pilot contamination, scheduling, HARQ feedback, or explicit queue dynamics, and QoS is enforced through SINR and stability surrogates rather than through hard delay or outage constraints. Finally, the reward weights are hand-tuned to keep the objective numerically well scaled, and the reported rate is normalized by the traffic-load factor, so the absolute values in Table I are meaningful for comparison across policies within this environment rather than as absolute link-level throughput figures.

VI. CONCLUSION

This paper proposed a DRL framework for energy-efficient, fair, and QoS-aware power control in multi-cell massive MIMO networks. We developed Smart-MIMO-Cell-Env, an enhanced four-cell downlink environment with $M = 16$ antennas and $K = 6$ users per cell, correlated Rayleigh channels with 3GPP-inspired large-scale fading, user mobility, time-varying traffic load, channel aging, and optional interference coordination. On top of this environment, we formulated a centralized MDP in which a single controller jointly selects per-UE transmit powers and per-antenna gain coefficients in an 88-dimensional continuous action space. The reward combines throughput, power consumption, link quality, mobility, traffic load, channel aging, fairness, and temporal power stability, allowing a single learning agent to internalize multiple design objectives.

We implemented PPO and SAC in Ray RLlib under the same environment, state and action spaces, network architectures, and software and hardware setup, so that differences in performance primarily reflect algorithmic rather than implementation choices. Training and evaluation on Smart-MIMO-Cell-Env showed that both algorithms learn non-trivial power-control behaviours during training, but PPO consistently achieves a much higher mean episode reward, higher downlink rates, and substantially better link quality than SAC, while operating at a moderate effective power level and exhibiting smooth, stable training dynamics. SAC converges to a low-power regime with low absolute throughput and degraded link quality ($\bar{R} \approx 0.047$ and $\bar{J} \approx 24.032$ in our 200-episode evaluation), despite a higher rate-per-Watt (PE ≈ 5.408).

The observed performance gap is explained by structural differences between the algorithms and their interaction with the environment. PPO is on-policy and relies on large batches of fresh trajectories together with a clipped surrogate objective and generalized advantage estimation. These properties are well matched to the slowly drifting and only approximately Markovian dynamics induced by mobility, traffic variations, and channel aging, as well as to the high-dimensional action space. SAC, in contrast, is off-policy and trains from a replay buffer that mixes samples from many past policies and operating conditions while exploring aggressively through an entropy-regularized objective. In the presence of non-stationary dynamics and history-based reward components, this time-mixing makes SAC more sensitive to critic bias and hyperparameter choices and, in our experiments, leads to a substantially worse evaluation return.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

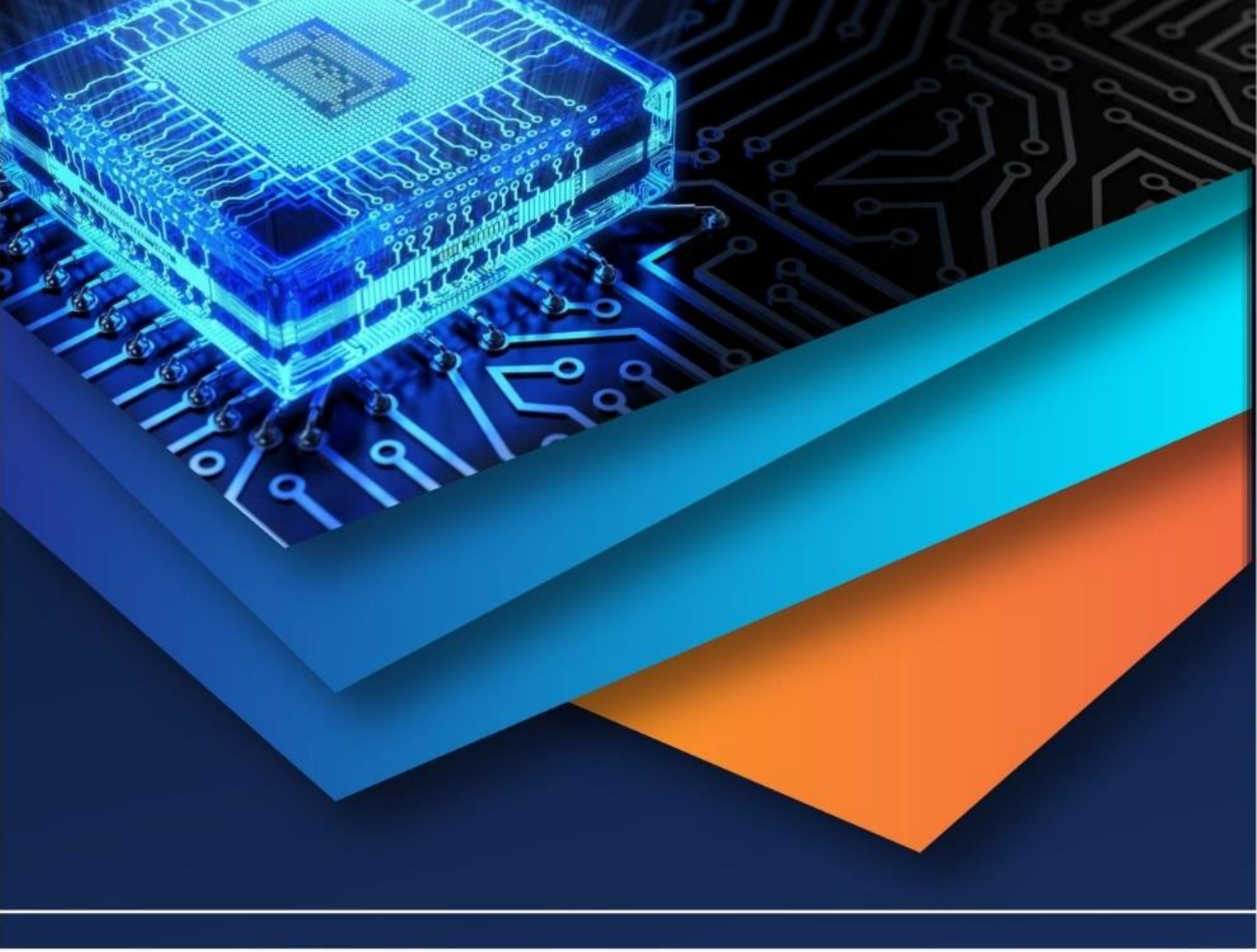
Overall, the proposed framework demonstrates that DRL can serve as a unified mechanism for multi-objective power control in massive MIMO networks, jointly accounting for energy efficiency, interference management, fairness, and robustness to realistic dynamics. Beyond the specific PPO–SAC comparison, the Smart-MIMO-Cell-Env environment, the MDP formulation, and the reward design provide a flexible testbed for future RL-based radio resource management schemes.

Future work includes extending the centralized controller to decentralized and multi-agent architectures, in which per-cell or per-cluster agents coordinate over limited backhaul signalling; incorporating explicit delay and queue-based QoS constraints, for example through Lyapunov or effective-capacity formulations; and integrating more detailed 5G and 6G procedures such as pilot contamination, scheduling, and HARQ feedback. Another important direction is to explore transfer- and meta-learning techniques that adapt trained policies across deployment scenarios, traffic regimes, and hardware constraints, moving closer to practical deployment in next-generation wireless networks.

REFERENCES

[1] NR; Physical Layer Procedures for Control, 3rd Generation Partnership Project (3GPP), Std. TS 38.213, Release 17.
[2] NR; Physical Layer Procedures for Control (3GPP TS 38.213) Release 18, v18.4.0, 3GPP/ETSI Std., 2024. [Online]. Available: https://www.etsi.org/deliver/etsi_ts/138200_138299/138213/18.04.00_60/ts_138213v180400p.pdf
[3] A. Ghazanfari, H. V. Cheng, E. Björnson, and E. G. Larsson, “A fair and scalable power control scheme in multi-cell massive MIMO,” in Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP), 2019, pp. 4499–4503.
[4] Q. Huang et al., “Energy-efficient operation of adaptive massive MIMO for 5G networks,” IEEE Trans. Wireless Commun., 2024.
[5] T. V. Chien, T. C. Nguyen, E. Björnson, and E. G. Larsson, “Power control in cellular massive MIMO with varying user activity: A deep learning solution,” IEEE Trans. Wireless Commun., vol. 19, no. 9, pp. 5732–5748, 2020.
[6] E. Björnson and L. Sanguinetti, “Making cell-free massive MIMO competitive with MMSE processing and centralized implementation,” IEEE Trans. Wireless Commun., vol. 19, no. 1, pp. 77–90, 2020.
[7] R. Han et al., “Effective capacity analysis of delay-sensitive wireless communications with power/rate allocation,” IEEE Trans. Wireless Commun., vol. 23, no. 2, pp. 1534–1549, 2024.
[8] L. Dai, J. Mei, Z. Yang, Z. Tong, C. Zeng, and K. Li, “Lyapunov-guided deep reinforcement learning for delay-aware online task offloading in MEC systems,” J. Syst. Archit., vol. 153, p. 103194, 2024.
[9] Y. S. Nasir and D. Guo, “Multi-agent deep reinforcement learning for dynamic power allocation in wireless networks,” IEEE J. Sel. Areas Commun., vol. 37, no. 10, pp. 2239–2250, 2019.
[10] F. B. Mismar, J. Choi, and B. L. Evans, “A framework for automated cellular network tuning with reinforcement learning,” IEEE Trans. Commun., vol. 67, no. 10, pp. 7152–7167, 2019.
[11] Z. Liu et al., “Cell-free XL-MIMO meets multi-agent reinforcement learning,” IEEE Wireless Commun., vol. 31, no. 4, pp. 64–71, 2024.
[12] A. A. Khan et al., “Percentile optimization in wireless networks—Part I: Problem formulation and power control,” IEEE Trans. Signal Process., vol. 72, pp. 2336–2351, 2024.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)