Multi-Objective Optimisation of URLLC-Based Metaverse Services

Xinyu Gao, *Member, IEEE,* Wenqiang Yi, *Member, IEEE*
Yuanwei Liu, *Senior Member, IEEE*, and Lajos Hanzo, *Life Fellow, IEEE*

Abstract—Metaverse aims for building a fully immersive virtual shared space, where the users are able to engage in various activities. To successfully deploy the service for each user, the Metaverse service provider and network service provider generally localise the user first and then support the communication between the base station (BS) and the user. A reconfigurable intelligent surface (RIS) is capable of creating a reflected link between the BS and the user to enhance line-of-sight. Furthermore, the new key performance indicators (KPIs) in Metaverse, such as its energy-consumption-dependent total service cost and transmission latency, are often overlooked in ultra-reliable low latency communication (URLLC) designs, which have to be carefully considered in next-generation URLLC (xURLLC) regimes. In this paper, our design objective is to jointly optimise the transmit power, the RIS phase shifts, and the decoding error probability to simultaneously minimise the total service cost and transmission latency and approach the Pareto Front (PF). We conceive a twin-stage central controller, which aims for localising the users first and then supports the communication between the BS and users. In the first stage, we localise the Metaverse users, where the stochastic gradient descent (SGD) algorithm is invoked for accurate user localisation. In the second stage, a meta-learning-based position-dependent multi-objective soft actor and critic (MO-SAC) algorithm is proposed to approach the PF between the total service cost and transmission latency and to further optimise the latency-dependent reliability. Our numerical results demonstrate that 1) The proposed solution strikes a tradeoff between the total service cost and transmission latency, which provides a candidate group of optimal solutions for diverse practical scenarios. 2) The proposed meta-learning-based MO-SAC algorithm is capable of adaption to new wireless environments, compared to the benchmarkers. 3) The approximate PF depicted discovered the relationships among the KPIs for the Metaverse, which provides guidelines for its deployment.

Index Terms—Metaverse, xURLLC, Multi-objective optimisation, Approximate-Pareto front

I. INTRODUCTION

The Metaverse is constituted by a network of three-dimensional (3D) virtual worlds relying on the social connections of computer-generated applications and the physical world. It hinges on technologies that empower multisensory connections with virtual environments, digital objects, and individuals [2]. Although in each virtual application, the localisation and communication can be processed separately. Metaverse has different applications in the same virtual world, so the localisation and communication for different users are overlapping at the same time. Hence, to avoid interference between localisation and communication stages, different operating frequencies-based physical-layer solutions are sought to support stringent localisation and data rate requirements for each user. Compared to visible light communication or vision-based technologies, millimetre wave (mmWave) technology is less affected by natural light and atmospheric media [3]. Furthermore, the locations of outer mmWave transmitters are known and they are regarded as accurate, while the localisation trackers of built-in are usually inaccurate since the errors accumulate [4]. To this end, mmWave techniques can be used for localising users, and the non-overlapping TeraHertz (THz) band may be harnessed for providing a Gbps-level data rate without imposing self-interference from sensing echoes. Additionally, the wireless sensing and communications services in Metaverse tend to rely on direct line-of-sight (LOS) propagation between the base station (BS) and users, but this may be blocked by obstacles [5]. The popular reconfigurable intelligent surfaces (RISs) [6]–[8], are capable of creating a reflected path in a cost-efficient manner for improving both the sensing accuracy and the spectral efficiency.

The 5G-style ultra-reliable low latency communication (URLLC) may be harnessed for early Metaverse applications on a small scale [9]–[11], as demonstrated in extended reality (XR) scenarios [12]–[14]. However, the evolved large-scale Metaverse imposes even more stringent localisation, data rate, and energy efficiency requirements on next-generation URLLC (xURLLC) [15], [16]. In addition to the aforementioned physical-layer challenges of the xURLLC-enabled Metaverse, we have to comprehensively consider the quality of experience (QoE) versus energy cost trade-off.

A. State-of-the-art

1) Evolution of Wireless Communication Aided Metaverse:
To support the ubiquitous coverage of Metaverse, flawless wireless transmission is necessary [17]. With the global research momentum ramping up surrounding the Metaverse, XR

is one of the most tangible manifestations of the Metaverse at the current state-of-the-art. Briefly, XR is an umbrella term that covers virtual reality (VR), augmented reality (AR), and mixed reality. A novel VR model based on multi-attribute utility theory was proposed by Chen *et al.* [18] for capturing the VR users' quality of service. The echo state network-based learning algorithm [19] is proposed for collecting the tracking information gleaned from VR users, and for trans-mitting video and the accompanying audio to the VR users via wireless channels. As a further advance, Taha *et al.* [20] developed a comprehensive red, green, and blue (RGB)-based framework for constructing accurate localisation and high-resolution depth maps using mmWave systems. The simula-tions showed the promising gains of mmWave-based depth perception compared to the conventional RGB approaches [21] in VR scenarios. Then Batalla *et al.* [22] analysed the quality of VR/AR video streamed over WiFi and 5G networks in harsh industrial environments. Indeed, in order to maintain low latency, while outsourcing computation in the face of mobility, the mobile edge cloud will play a key role in next-generation networks. In this context, a novel performance metric namely the XR quality index (XQI) was proposed by Dou *et al.* [23] to reflect the impact of networking-induced imperfections on XR services. Specifically, both fine-grained and coarse-grained XQI models are provided for producing a final score that can reflect the impact of realistic imperfect networking on the QoE. The Metaverse relies on the elements of XR and combines them with the conventional Internet. To elaborate, there are several emerging considerations in the Metaverse. For example, to achieve customised high-quality Metaverse services, a user-attention-aware network resource alposition was designed by Du *et al.* [24]. To minimise the commu-nication load, a sampling, communication, and prediction co-design framework was proposed by Meng *et al.* [25] subject to a constraint on tracking the mean squared error between a real-world device and its digital model in the Metaverse. To minimise the cost to the service provider, Ng *et al.* [26] conceived a virtual education case study in the Metaverse and solved the associated unified resource alposition problem in the face of stochastic user demand. With the development of the Metaverse, the range of key performance indicators (KPIs) has also been extended to the associated wireless network services. The roll-out of large-scale applications imposes high costs on the Metaverse service providers (MSP) and network service providers (NSP). Therefore, the energy-related total operational expenses, namely total service cost, should also be considered as a KPI in Metaverse.

2) Evolution from URLLC to xURLLC: In this context, Xie *et al.* [27] studied the URLLC downlink, where a RIS assisted a BS sends individual short-packet messages to multiple users. They solved the associated latency minimisation problem via alternating optimisation. Then the resource alposition of RIS-assisted multiple-input single-output orthogonal frequency di-vision multiple access (OFDMA)-aided multicell networks was investigated by Ghanem *et al.* [28], where a set of cooperating BSs served a set of URLLC users. As a further development, Hashemi *et al.* [29] proposed a multi-objective (MO) optimisation problem for maximising the achievable

finite blocklength rate, while minimising the blocklengths of a RIS-assisted system. In addition to the latency and reliability requirements considered in traditional URLLC, Haber *et al.* [30] optimised the use of the RIS along with the design of the offloading and resource alposition parameters for maximising the sum rate of the users while satisfying stringent reliability specifications. In [31], the self-adaptive flexible transmission time interval scheduling strategy of the enhanced mobile broadband and URLLC coexistence scenario was proposed for improving the reliability of the system. Then Chaccour *et al.* [32] investigated whether the data-rate QoE requirements can be met in case of wireless connectivity in the THz frequency bands. Ren *et al.* [33] proposed to deploy RISs for enhancing the transmission reliability under specific data rate constraints. A RIS-assisted wireless communication system relying on non-linear energy harvesting and ultra-reliable low-latency constraints was designed by Dohk *et al.* [34] for applications in industrial automation. Furthermore, a URLLC-aided relaying system was considered in [35], where the design objective was to minimise the total energy subject to optimal resource alposition.

B. Motivations and Contributions

The xURLLC enhances real-time interactions, immer-sive experiences, and collaboration within the Metaverse by providing ultra-reliable and low-latency communication. Although the existing literature outlined the benefits of the communication-aided Metaverse and the foundations of xURLLC, there are still numerous crucial research challenges. Firstly, multiple KPIs, such as the total service cost and transmission latency, have to be optimised simultaneously in the emerging xURLLC-enabled Metaverse. The conflicting relationships among these KPIs typically result in a MO optimisation problem. Secondly, accurate interference-free lo-calisation and communication are of salient significance in this context. Thirdly, improving the channel conditions by RISs based on the specific positions of the users is still a challenge. We are inspired to solve these problems, hence the main contributions of this work can be summarised as follows:

- We propose a self-interference-free twin-stage central controller for an indoor RIS-assisted Metaverse sce-nario, which intrinsically amalgamates mmWave and non-overlapping THZ for localising the users and supports communication between the BS and users. To analyse the performance of the proposed network in the Metaverse system, new KPIs, such as the total service cost as well as transmission latency are defined and considered. Since the bit-level reliability attained and the transmission latency in the physical layer are correlated, our goal is to strike a balance between the total service cost and the transmission latency of the Metaverse users.
- We design a stochastic gradient descent (SGD)-based algorithm for approaching the best-case localisation per-formance characterised by the Cramér-Rao method. We adopt a localisation-dependent meta-learning-based MO-soft-actor-and-critic (SAC) algorithm to train a policy, which aims for adapting to new tasks, such as new wireless environments. The core point of this strategy
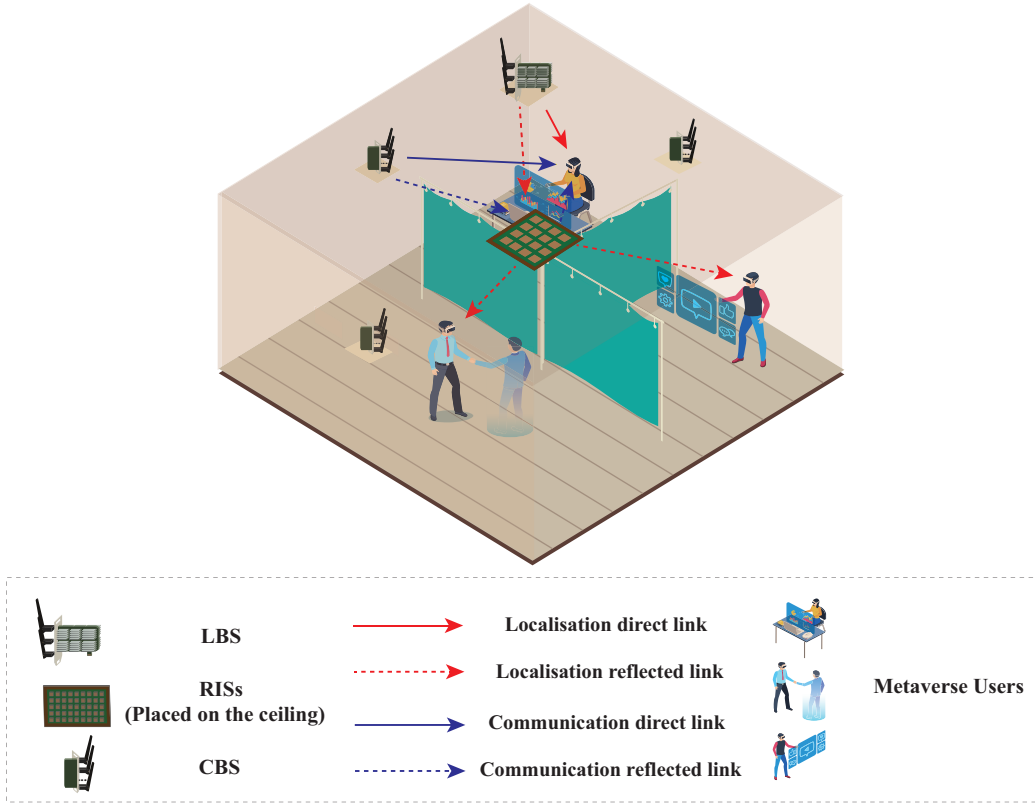
Fig. 1: Illustration of an Metaverse scenario considered.

is to integrate the action values acquired by random sampling preferences and assign dynamic weights for both the loss functions for further updating the policy by homotopy optimisation. We demonstrate that this algorithm is capable of adapting to the next tasks in the face of uncertainty.

- Our proposed solution strikes a tradeoff between the total service cost and transmission latency, which provides a candidate group of near-optimal solutions for practical scenarios. The approximate Pareto Front (PF) portrays the relationships among the KPIs for the Metaverse, which provides guidelines for the application's deployment for MSP and NSP. Compared to conventional MO-SAC and single-objective algorithms, our meta-learning-based MO-SAC algorithm adapts more promptly to new tasks upon increasing the number of training samples and approaches the PF.

C. Orgainisations

The rest of this paper is organised as follows. Section II presents our system model followed by the formulation of our simultaneous total service cost and latency optimisation problems. In Section III, we propose a bespoke meta-learning-based MO-SAC algorithm. Section IV presents our numerical results for quantifying the meta-learning-based MO-SAC algorithm's performance. Finally, Section V concludes this paper.

II. SYSTEM MODEL AND PROBLEM FORMULATION

In this paper, we focus our attention on an indoor Metaverse scenario consisting of U single-antenna Metaverse users. To support Metaverse services for the users, we consider a twin-stage central controller to assist their actions. The *first stage* determines the position of Metaverse users by a mmWave localisation base station (LBS), namely a mmWave Wi-Fi router, having N antennas, where the RIS is able to create a reliable reflected link between the LBS and the blocked Metaverse users. The *second stage* supports information communication between the B THz communication base stations (CBSs) and the Metaverse users, where the RIS enhances the link quality by mitigating the attenuation between the CBS and Metaverse users. The RIS equipped with $K = BK_b$ reconfigurable elements is employed for assisting the hybrid wireless network, where B and K_b denote the number of sub-surfaces and the number of elements in each sub-surface, respectively [1]. The CBSs and LBS are distributed on the wall, while the RIS is located on the ceiling [2]. Due to the narrow pencil-beams of THz systems [5], each CBS only serves a single user at each time slot t, i.e. $U \leq B$. In this model, we focus on the communication performance within a time period $\mathcal{T}$, which consists of T time slots. Then, the aforementioned two stages will be introduced in more depth in the following subsections.

A. Localisation Stage

In this stage, the LBS and RIS cooperate to locate the Metaverse users. Let us assume having a 3D Cartesian coordinate system, where the origin is at the centre of the floor, as shown in Fig. 1. The positions of the LBS, the u-th Metaverse

---

[1]Note that to take full advantage of the RIS resources, the number of RIS sub-surfaces is consistent with the number of CBSs, while each CBS is able to build a link with the sub-surface.

[2]We assume that the vertical height ratio of the ceiling and Metaverse users is high enough so that the RIS is able to locate all the Metaverse users without blockage in its fields of view.

user, and the RIS are given by $\mathbf{p}_L = (x_L, y_L, h_L)$, $\mathbf{p}_{u,t} = (x_{u,t}, y_{u,t}, h_{u,t})$, and $\mathbf{p}_r = (x_r, y_r, h_r)$, respectively. The LBS transmits a continuous mmWave wave OFDM signal in a resource block containing M adjacent subcarriers. Since the frequencies reach the mmWave and Terahertz band, the NLOS components of the channel models can be ignored. Hence, the path loss, and AOA and AOD of the LOS component can be determined, once the BS and RIS are deployed, which can be further regarded as deterministic. The channel gain between the LBS and u-th Metaverse user at the m-th subcarrier at time slot t, $t \in [0, T]$ can be expressed as:

$$(\mathbf{h}_{u,t}^m)^H = (\mathbf{h}_{L,u,t}^m)^H + (\mathbf{h}_{R,u,t}^m)^H \mathbf{\Theta}(t) \mathbf{H}_{L,R,t}^m, \quad (1)$$

where $\mathbf{\Theta}(t) = \text{diag}(e^{j\theta_1,t}, e^{j\theta_2,t}, \cdots, e^{j\theta_K,t})$ is the reflection coefficient matrix of RIS. Accordingly, the receive signal can be expresses as:

$$Y_{u,t}^m = (\mathbf{h}_{u,t}^m)^H X_{u,t}^m + n_0, \quad (2)$$

where variable n_0 is the circularly symmetric complex Gaussian noise with zero mean and variance ν^2. The channels spanning from the LBS to the u-th Metaverse user $(\mathbf{h}_{L,u,t}^m)^H \in \mathbb{C}^{1 \times N}$, from the RIS to u-th Metaverse user $(\mathbf{h}_{R,u,t}^m)^H \in \mathbb{C}^{1 \times K}$, and from LBS to RIS $\mathbf{H}_{L,R,t}^m \in \mathbb{C}^{K \times N}$ are defined as [3]:

$$(\mathbf{h}_{L,u,t}^m)^H = \alpha_{L,u,t} e^{-j2\pi m \frac{\tau_{L,u,t}}{M\mathcal{B}}} \mathbf{a}_L^H(\omega_{L,u,t}), \quad (3)$$

$$(\mathbf{h}_{R,u,t}^m)^H = \alpha_{R,u,t} e^{-j2\pi m \frac{\tau_{R,u,t}}{M\mathcal{B}}} \mathbf{a}_R^H(\omega_{R,u,t}), \quad (4)$$

$$\mathbf{H}_{L,R,t}^m = \alpha_{L,R,t} e^{-j2\pi m \frac{\tau_{L,R,t}}{M\mathcal{B}}} \mathbf{a}_R(\varphi_{L,R,t}) \mathbf{a}_L^H(\omega_{L,R,t}), \quad (5)$$

where $\alpha_{c_0,c_1,t} = \rho_{c_0,c_1,t} e^{j\phi_{c_0,c_1,t}}$ has the modulus $\rho_{c_0,c_1,t}$ and phase $\phi_{c_0,c_1,t}$, $c_0 \in \{L, R\}$, $c_1 \in \{R, u\}$. The variables $\varphi_{c_0,c_1,t}$, $\omega_{c_0,c_1,t}$, and $\tau_{c_0,c_1,t} = ||\hat{p}_{c_0,c_1,t}||/c + \eta_0$ are the angle-of-arrival (AOA), the angle-of-departure (AOD), and the delay. The variables $\hat{p}_{c_0,c_1,t}$, c, and η_0 are the distance between c_0 and c_1, speed of light, and clock offset, respectively. The variable $\mathcal{B} = 1/B_0$ represents the sampling time with B_0 being the bandwidth. The transmit steering vector (TSV) in the localisation stage is given by:

$$\mathbf{a}_{c_0}^H(\omega_{c_0,c_1,t}) =$$
$$[1, e^{j\frac{2\pi}{\lambda_c} z \sin \omega_{c_0,c_1,t}}, \cdots, e^{j(N-1)\frac{2\pi}{\lambda_c} z \sin \omega_{c_0,c_1,t}}]^H, \quad (6)$$

where λ_c and z are the mmWave wavelength and the antenna element spacing, respectively. Let us denote the estimated position error and the accurate position area by $\mathbf{p}_{u,t}^e = (x_{u,t}^e, y_{u,t}^e, h_{u,t}^e)$, and $\mathbf{p}_{u,t}^* = (x_{u,t}^*, y_{u,t}^*, h_{u,t}^*)$, respectively. The estimated positions of u-th user at time slot t can be expressed as:

$$\mathbf{p}_{u,t} \in [\mathbf{p}_{u,t}^* - \mathbf{p}_{u,t}^e, \mathbf{p}_{u,t}^* + \mathbf{p}_{u,t}^e], \quad \eta_0 < \overline{\eta}, \quad (7)$$

where $\overline{\eta}$ is the maximum tolerable clock offset determining the maximum synchronisation error for localisation.

B. Communication Stage

In the second stage, the central controller pairs each CBS with one of the Metaverse users based on the localisation results. Each CBS orientates its main lobe direction towards

---

[3]The elements in RIS are deployed edge-to-edge in the isotropic scattering environment considered, and an omnidirectional antenna LBS is deployed [36], [37].

---

the associated user. Let $s_{b,u,t} \in \{0, 1\}$ denote the index of the link status between the b-th CBS and the u-th user at time slot t. If $s_{b,u,t} = 1$, the link is established; otherwise, $s_{b,u,t} = 0$. The transmit gain of each CBS and the receive gain of each user is given by [38]:

$$G_{\text{Tr}} = \frac{4\pi}{\delta_{\text{Tr}} + 1} \Omega_{\text{Tr}}, \quad (8)$$

$$G_{\text{Re}} = \frac{4\pi}{\delta_{\text{Re}} + 1} \Omega_{\text{Re}}, \quad (9)$$

where $\delta_{(\beta)}, \beta \in \{\text{Tr}, \text{Re}\}$ is the power ratio between the side lobes and the main lobe of the transmit antenna (TA) and receive antenna (RA). We have $\Omega_\beta = 4 \arcsin(\tan(\psi_H^\beta/2) \tan(\psi_V^\beta/2))$, where ψ_H^β and ψ_V^β denote the horizontal and vertical beam widths of the TA and RA, respectively. Let $\hat{g}_{b,b,u,t} \in \mathbb{C}^{1 \times K_b}$ represent the cascaded channel gain from the b-th CBS to the u-th Metaverse user via the b-th sub-surface. Thus, the cascaded channel from the b-th CBS to the u-th link via the b-th sub-surface in the RIS is given by [39]:

$$g_{b,b,u,t}(s_{b,u,t}) = s_{b,u,t}\hat{g}_{b,b,u,t}\mathbf{a}_R(\omega_{b,u,f,t})\mathbf{\Theta}(t)\mathbf{a}_S^H(\varphi_{b,b,f,t}), \quad (10)$$

where $\omega_{b,u,f,t}$, $\varphi_{b,b,f,t}$ are the AOA from b-th sub-surface to u-th user and AOD from b-th CBS to the b-th sub-surface, respectively. The TSVs from the b-th sub-surface of the RIS to the u-th user and from the b-th CBS to the b-th sub-surface of the RIS can be rewritten as:

$$\mathbf{a}_R(\omega_{b,u,f,t})$$
$$= [e^{-j\omega_{b_1,u,f,t}}, \cdots, e^{-j\omega_{b_k,u,f,t}}, \cdots, e^{-j\omega_{b_{K_b},u,f,t}}], \quad (11)$$

$$\mathbf{a}_S^H(\varphi_{b,b,f,t})$$
$$= [e^{-j\varphi_{b,b_1,f,t}}, \cdots, e^{-j\varphi_{b,b_k,f,t}}, \cdots, e^{-j\varphi_{b,b_{K_b},f,t}}], \quad (12)$$

where the variables $\omega_{b_k,u,f,t}$ and $\varphi_{b,b_k,f,t}$ are the phase differences of the incoming signal to the k-th reflecting element relative to the first element b_1 of the b-th sub-surface, and the signal to the u-th user reflected from the k-th reflecting element relative to the first element b_1 of the b-th sub-surface, respectively. According to [40], the variable $\hat{g}_{b,b,u,t}$ in (10) can be expressed as:

$$\hat{g}_{b,b,u,t} = \frac{\sqrt{G_{\text{Tr}}G_{\text{Re}}}c}{8\sqrt{\pi^3}f\sigma_{u,b}} e^{-j\frac{2\pi f}{c}d_{b,b,u,t}} e^{-\frac{1}{2}V(f)d_{b,b,u,t}}, \quad (13)$$

where $\sigma_{u,b}$, and f are the reference distance of the b-th sub-surface, and the THz operating frequency, respectively. Furthermore, $d_{b,b,u,t} = |[x_{b_1} - x_{u,t}, y_{b_1} - y_{u,t}, z_{b_1} - z_{u,t}]| + |[x_b - x_{b_1}, y_b - y_{b_1}, z_b - z_{b_1}]|$, where $(x_{b_1}, y_{b_1}, z_{b,1})$ is the coordinate of the first element of the b-th sub-surface of the RIS. Finally, $e^{-V(f)}$ denotes the medium transmittance obeying Beer-Lambert law with $V(f)$, which is regarded as the overall absorption medium coefficient at frequency f [41]. The path loss of the direct link between b-th SBS and u-th user can be given by [32]:

$$\mathcal{L}_{b,u,t}(s_{b,u,t}) =$$
$$\begin{cases} \frac{\sqrt{G_{\text{Tr}}G_{\text{Re}}}c}{8\sqrt{\pi^3}f\sigma_{u,b}} e^{-\frac{1}{2}V(f)E_{b,u,t}(s_{b,u,t})}, & s_{b,u,t} = 1, \\ 0, & s_{b,u,t} = 0, \end{cases} \quad (14)$$

where $E_{b,u,t}(s_{b,u,t})$ denotes the distance between b-th SBS and u-th user.

1) Data Rate: In the system studied, the achievable data rate is defined as the ratio of the number of information bits to the number of symbols transmitted. The signal-to-noise ratio (SNR) ϱ of the b-th CBS to the u-th user via the b-th sub-surface of the RIS at time slot t can be expressed as:

$$\varrho_{b,b,u,t}(s_{b,u,t}) = \frac{P_t|g_{b,b,u,t}(s_{b,u,t}) + \mathcal{L}_{b,u,t}(s_{b,u,t})|^2}{n_{u,t}^2(s_{b,u,t})}, \quad (15)$$

where $n_{u,t}^2(s_{b,u,t})$ denotes the noise power. Hence the associated achievable data rate can be expressed as follows [42]:

$$D_{b,b,u,t}(s_{b,u,t}) = $$
$$s_{b,u,t}\left(\log_2\left(1 + \varrho_{b,b,u,t}(s_{b,u,t})\right) - \sqrt{\frac{\hat{D}}{m}}\frac{Q^{-1}(\varepsilon)}{\ln 2}\right), \quad (16)$$

where m and ε are the transmission block length and the decoding error probability (DEP). $Q^{-1}(x)$ is the inverse of the function $Q(x) = 1/\sqrt{2\pi}\int_x^\infty e^{-t^2/2}dt$, while $\hat{D} = 1 - \left(1 + \varrho_{b,b,u,t}(s_{b,u,t})\right)^2$.

2) Transmission Latency: Assuming that the transmitted data size assigned to the u-th user at time slot t is $S_{u,t}$, the resultant transmission delay can be formulated as:

$$J_{u,t}(s_{u,t}) = \frac{S_{u,t}}{\sum_{b=1}^{B} D_{b,b,u,t}(s_{b,u,t})}. \quad (17)$$

Then, the transmission state of the u-th user at time slot t can be expressed as:

$$\mathcal{K}_{u,t}(\mathbf{s}_{u,t}) = \begin{cases} 1, & J_{u,t}(\mathbf{s}_{u,t}) \leq \Delta t, \\ 0, & \text{Otherwise}, \end{cases} \quad (18)$$

where Δt is the maximum tolerable transmission delay, and $\mathbf{s}_{u,t} = [s_{1,u,t}, s_{2,u,t}, \cdots, s_{B,u,t}]$. Then we define a index $\overline{J}_{u,t}(s_{u,t})$ to represent the maximum transmission latency among Metaverse users at each time slot, which can be expressed as:

$$\overline{J}_{u,t}(s_{u,t}) = \max\{J_{1,t}(s_{1,t}), \cdots, J_{u,t}(s_{u,t}), \cdots, J_{U,t}(s_{U,t})\} \quad (19)$$

The latency performance of the networks considered is evaluated by transmission latency index in the networks considered.

3) Total Service Cost: Given its demanding specifications, the total service cost is a pivotal KPI for the Metaverse, which depends on the energy and DEP. The total service cost of the Metaverse services is determined by both the MSP and NSP, and it can be defined as follows [26]:

$$\mathcal{E}(P_t, \varepsilon) = -C_{\text{meta}} + C_{\text{net}}(P_t, \varepsilon),$$
$$= -(F_{\text{meta}}^s + F_{\text{QoE}}^s) + (f_P^s \sum_{t=1}^{T} P_t + f_\varepsilon^s \varepsilon) \quad (20)$$

where C_{meta} and $C_{\text{net}}(P_t, \varepsilon)$ are the cost and revenue of the entire Metaverse system, respectively. Explicitly, C_{meta} is determined by the fixed cost F_{meta}^s of hardware and the cost F_{QoE}^s of satisfying the QoE required. Furthermore, the $C_{\text{net}}(P_t, \varepsilon)$ is determined by the cost of transmit power $f_P^s \sum_{t=1}^{T} P_t$ and the DEP $f_\varepsilon^s \varepsilon$, which has to be optimised, where f_P^s and f_ε^s are the fee for unit P_t value and ε value, respectively.

C. Problem Formulation

Both the latency and the cost are important for Metaverse applications. For example, low latency ensures a smooth user experience and real-time interactions, while optimising the costs allows for scalability and accessibility, fostering wider adoption of the Metaverse. Naturally, our preference is to achieve high reliability at a low total service cost. In the model studied, the reliability and transmission latency are correlated, where a low transmission rate indicates high reliability. Therefore, our goal is to strike a balance between the total service cost and the minimal transmission latency of the Metaverse users. The minimisation problem can be formulated as follows:

$$\min_{s_{b,u,t},\varepsilon,\eta_0,P_t,\mathbf{\Theta}(t)} \left[\sum_{t=1}^{T}\mathcal{E}(P_t,\varepsilon), \overline{J}_{u,t}(s_{u,t})\right] \quad (21)$$

$$\text{s.t.} \quad 0 < P_t \leq P_{\max}, \quad (21a)$$
$$0 \leq \theta_k \leq 2\pi, k \in \{1, 2, \cdots, K\}, \quad (21b)$$
$$\sum_{b=1}^{B} s_{b,u,t} = 1, \forall u \in 1, 2, \cdots, U, \quad (21c)$$
$$\sum_{u=1}^{U} s_{b,u,t} = 1, \forall b \in 1, 2, \cdots, B, \quad (21d)$$
$$s_{b,u,t} \in \{0, 1\}, \quad (21e)$$
$$\varepsilon \leq \varepsilon_m, \quad (21f)$$
$$\eta_0 < \overline{\eta}, \quad (21g)$$

where $P_{\max}$ and ε_m are the maximum transmit power of the Metaverse users and the maximum tolerable DEP, respectively. Constraint (21a) limits the transmit power, while constraint (21b) limits the phase of the RIS reflection coefficients. Furthermore, constraints (21c), (21d), and (21e) represent to the one-to-one mapping of the CBSs and Metaverse users at time slot t. Constraints (21f) and (21g) represent the limit of ε and of the maximum synchronisation error η_0 for localisation, respectively. However, problem (21) is difficult to solve for the following four reasons. Firstly, the Metaverse users' positions change as time elapses, making the channel hard to model. Secondly, the problem formulated is considered a Markov Decision Process (MDP) problem. Thus, the requirements of any immediately consecutive time slots are governed by MDP, which does not lend itself to employment in our infinite variable optimisation. Thirdly, the total service cost of the MSP and the transmission latency of the NSP are subject to a trade-off, when optimising the DEP and transmit power. Explicitly, upon increasing the DEP and the transmit power, the total service cost increases, while the latency decreases. Hence, jointly optimising the total service cost and latency for obtaining a group of Pareto-optimal (PO) results provides us with a range of different system settings for NSP according to the prevalent practical requirements, compared to conventional methods. Fourthly, the model trained is required to achieve prompt convergence to the new tasks, but this issue is beyond the scope of conventional MO optimisation algorithms. Therefore, the traditional optimisation and reinforcement learning (RL) algorithms are unsuitable for solving the problem (21).

As a remedy, we propose a novel meta-learning-based MO optimisation algorithm.

III. META LEARNING-BASED POSITION-DEPENDENT MO-RL ALGORITHM

In this section, we introduce a position-dependent MO-RL algorithm based on the meta-learning framework [43]. There are two stages of the proposed algorithm: the position-dependent MO-RL learning stage, and the meta-learning stage. The position-dependent MO-RL stage is able to simultaneously optimise multiple objectives based on the estimated positions of users, while the meta-learning aims for training a model for prompt adaptation to new tasks. The timeline of proposed algorithms is shown in Fig. 2.
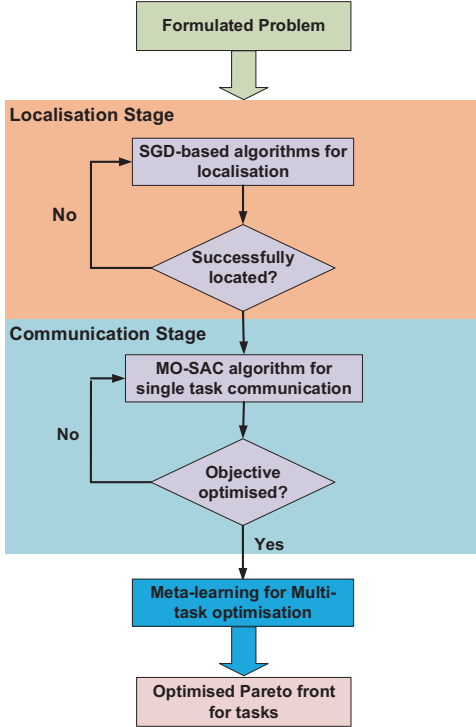

Fig. 2: The sequence of the algorithms proposed for solving the problem formulated.

A. Position-dependent MO-RL Learning Stage

The localisation-dependent MO-RL Learning stage aims for finding a policy to solve problem (21). For localisation, the Cramér-Rao method [44] is employed for calculating the lower bounds of the Metaverse users' localisation, and a policy gradient-based algorithm is invoked for estimating the positions. Then, (21) can be optimised according to the estimated positions.

1) Cramér-Rao Lower Bounds: The steps are briefly presented for highlighting the lower bounds of the localisation estimation of the user position $\mathbf{p}_{u,t}$ and clock offset η_0. According to the channel model of the localisation stage, the channel parameter vector of u-th user at time slot t can be expressed as follows:

$$\overline{\mathbf{q}}_{u,t} = [\tau_{\mathrm{L},u,t}, \omega_{\mathrm{L},u,t}, \rho_{\mathrm{L},u,t}, \phi_{\mathrm{L},u,t}, \tau_{\mathrm{L},u,t}, \omega_{\mathrm{L},u,t},$$
$$(\rho_{\mathrm{L},\mathrm{R},t}\rho_{\mathrm{R},u,t}), (\phi_{\mathrm{L},\mathrm{R},t} + \phi_{\mathrm{R},u,t})]. \quad (22)$$

Upon denoting the unbiased estimator $\overline{\mathbf{q}}_{u,t}^e$ of $\overline{\mathbf{q}}_{u,t}$, the corresponding Fisher information matrix (FIM) $\mathcal{J}_{\overline{\mathbf{q}}_{u,t}} \in \mathbb{C}^{8\times8}$ should satisfing [44]:

$$\mathbb{E}\{(\overline{\mathbf{q}}_{u,t}^e - \overline{\mathbf{q}}_{u,t})(\overline{\mathbf{q}}_{u,t}^e - \overline{\mathbf{q}}_{u,t})^H\} \succeq \mathcal{J}_{\overline{\mathbf{q}}_{u,t}}^{-1}, \quad (23)$$

where $\mathbf{E}_0 \succeq \mathbf{F}_0$ means that $\mathbf{E}_0 - \mathbf{F}_0$ is positive semi-definite. Accordingly, the (a,b)-th FIM entry $[\mathcal{J}_{\overline{\mathbf{q}}_{u,t}}]_{(a,b)}$ can be expressed according to **Lemma 1**:

$$[\mathcal{J}_{\overline{\mathbf{q}}_{u,t}}]_{(a,b)} = \frac{2}{\epsilon^2} \sum_{m=0}^{M-1} \mathcal{R}\left\{\left(\frac{\partial \tilde{h}_{u,t}^m}{\overline{q}_{u,t}^a}\right)^H \frac{\partial \tilde{h}_{u,t}^m}{\overline{q}_{u,t}^b}\right\}, \quad (24)$$

where we have $\tilde{h}_{u,t}^m = [\sqrt{P_0}(\mathbf{h}_{\mathrm{L},u,t}^m + (\mathbf{h}_{\mathrm{R},u,t}^m)^H \mathbf{\Theta}(t) \mathbf{H}_{\mathrm{LBS},\mathrm{R},t}^m)^H] X_{u,t}^m$. The variables P_0 and $X_{u,t}^m$ represent the constant transmit power of mmWave communication and transmitted symbol of the m-th subcarrier at time slot t.

Lemma 1. *For a vector $\mathbf{Z}$ which follows a symmetric complex Gaussian distribution (Z_0, Z_1), the (a,b)-th entry of its Fisher information matrix is given by the Slepian-Bangs formula [44] of:*

$$[\mathcal{J}_{\mathbf{Z}}]_{(a,b)} = 2\mathcal{R}\left\{\left(\frac{\partial Z_0}{\mathbf{Z}^a}\right)^H Z_1^{-1} \frac{\partial Z_0}{\mathbf{Z}^b}\right\} + \mathcal{T}\left\{Z_1^{-1} \frac{Z_1}{\mathbf{Z}^a} Z_1^{-1} \frac{Z_1}{\mathbf{Z}^b}\right\}. \quad (25)$$

Therefore, the FIM can be expressed in the channel domain as follows:

$$\mathcal{J}_{\overline{\mathbf{q}}_{u,t}} = \begin{bmatrix} \mathcal{J}_{\overline{\mathbf{q}}_{\mathrm{L},u,t}} & \mathcal{J}_c \\ \mathcal{J}_c & \mathcal{J}_{\overline{\mathbf{q}}_{\mathrm{L},\mathrm{R},u,t}} \end{bmatrix}, \quad (26)$$

where $\mathcal{J}_{\overline{\mathbf{q}}_{\mathrm{L},u,t}} \in \mathbb{C}^{4\times4}$ and $\mathcal{J}_{\overline{\mathbf{q}}_{\mathrm{L},\mathrm{R},u,t}} \in \mathbb{C}^{4\times4}$ are the FIM submatrices corresponding to the LOS path and the non-line-of-sight (NLOS) path, respectively. The variable $\mathcal{J}_c$ is the LOS-NLoS path cross-correlation. To obtain the FIM in the position domain, the position parameter vector $\hat{\mathbf{q}}_{u,t}$ can be transformed from the unknown channel parameters vector $\overline{\mathbf{q}}_{u,t}$ as follows:

$$\hat{\mathbf{q}}_{u,t} = [x_{u,t}, y_{u,t}, h_{u,t}, \rho_{\mathrm{L},u,t}, \phi_{\mathrm{L},u,t},$$
$$(\rho_{\mathrm{L},\mathrm{R},t}\rho_{\mathrm{R},u,t}), (\phi_{\mathrm{L},\mathrm{R},t} + \phi_{\mathrm{R},u,t}), \eta_0], \quad (27)$$

and the corresponding FIM in the position domain is given by:

$$\mathcal{J}_{\hat{\mathbf{q}}_{u,t}} = \frac{\partial \overline{\mathbf{q}}_{u,t}^H}{\hat{\mathbf{q}}_{u,t}} \mathcal{J}_{\overline{\mathbf{q}}_{u,t}} \left(\frac{\partial \overline{\mathbf{q}}_{u,t}^H}{\hat{\mathbf{q}}_{u,t}}\right)^H. \quad (28)$$

Therefore, the lower bounds of the u-th user's position estimate and the clock offset estimate can be derived from the diagonal elements of the corresponding inverted FIM $\mathcal{J}_{\hat{\mathbf{q}}_{u,t}}$. The position error bound (PEB) of the u-th user at time slot t is formulated as:

$$\mathrm{PEB}_{u,t} = \sqrt{[\mathcal{J}_{\hat{\mathbf{q}}_{u,t}}]_{(1,1)}^{-1} + [\mathcal{J}_{\hat{\mathbf{q}}_{u,t}}]_{(2,2)}^{-1} + [\mathcal{J}_{\hat{\mathbf{q}}_{u,t}}]_{(3,3)}^{-1}}. \quad (29)$$

Then equation (7) can be written as:

$$\mathbf{p}_{u,t} \in [(\mathbf{p}_{u,t}^* - \mathrm{PEB}_{u,t}), (\mathbf{p}_{u,t}^* + \mathrm{PEB}_{u,t})], \quad \eta_0 = \eta_0^e, \quad (30)$$

where η_0^e is the estimated clock offset.

2) SGD-based Algorithm for Position Estimation: The Cramér-Rao bound is important for the Metaverse as it provides a benchmark for assessing the achievable accuracy in

parameter estimation. This is relevant for tracking user movements, locations, or gestures, helping optimise the reliability of sensor-based applications within the Metaverse. However, this method is the general solution to evaluate the positions, our goal is to minimise the lower bound to approach the accurate estimations. Based on the lower bounds, we design the direct estimator $\overline{\mathbf{p}}_{u,t} = [x_{u,t}, y_{u,t}, h_{u,t}, \eta]$ of u-th Metaverse user at time slot t. The signal $\mathbf{Y}_{u,t}$ received over M subcarriers can be expressed as:

$$\mathbf{Y}_{u,t} = \sqrt{P_0}\boldsymbol{\iota}\boldsymbol{\alpha} + n_0, \tag{31}$$

where we have $\boldsymbol{\alpha} = [\alpha_{\mathrm{L},u,t}, (\alpha_{\mathrm{L,R},t}\alpha_{\mathrm{R},u,t})]$, and $\boldsymbol{\iota} = [\mathbf{X}_{\mathrm{L},u,t}^H \mathbf{a}_{\mathrm{L}}(\omega_{\mathrm{L},u,t}), \mathbf{X}_{\mathrm{L,R},u,t}^H (\mathbf{a}_{\mathrm{R}}(\varphi_{\mathrm{L,R},t})\mathbf{a}_{\mathrm{L}}^H(\omega_{\mathrm{L,R},t}))^H \boldsymbol{\Theta}(t) \mathbf{a}_{\mathrm{R}}^H(\omega_{\mathrm{R},u,t})]$. The $\mathbf{X}_{\mathrm{L},u,t} = [X_{\mathrm{L},u,t}^0, e^{-j2\pi/M\mathcal{B}}X_{\mathrm{L},u,t}^1, \cdots, e^{-j2\pi(M-1)/M\mathcal{B}}X_{\mathrm{L},u,t}^{M-1}]$, while $\boldsymbol{\alpha}$ is the unknown parameter vector, then the estimation problem can be formulated as

$$\overline{\mathbf{p}}_{u,t}^* = \arg\min_{\overline{\mathbf{p}}_{u,t}}[\min_{\boldsymbol{\alpha}} \mathcal{L}(\overline{\mathbf{p}}_{u,t}, \boldsymbol{\alpha})], \tag{32}$$

where $\mathcal{L}(\overline{\mathbf{p}}_{u,t}, \boldsymbol{\alpha}) = ||\mathbf{Y}_{u,t} - \sqrt{P_0}\boldsymbol{\iota}\boldsymbol{\alpha}||^2$. This likelihood function can be minimised with respect to $\boldsymbol{\alpha}$, which can be rewritten as

$$\overline{\mathcal{L}}(\overline{\mathbf{p}}_{u,t}) = ||\mathbf{Y}_{u,t} - \sqrt{P_0}\boldsymbol{\iota}(\overline{\mathbf{p}}_{u,t})\hat{\boldsymbol{\alpha}}(\overline{\mathbf{p}}_{u,t})||^2, \tag{33}$$

where $\hat{\boldsymbol{\alpha}} = \frac{1}{\sqrt{P_0}}(\boldsymbol{\iota}^H\boldsymbol{\iota})^{-1}(\boldsymbol{\iota}^H)\mathbf{Y}_{u,t}$. The policy gradient can be written as:

$$\overline{\gamma} = \overline{\gamma} + \frac{1}{\hat{\mathcal{B}}}\nabla_{\overline{\mathbf{p}}_{u,t}}\overline{\mathcal{L}}^{\hat{b}}(\overline{\mathbf{p}}_{u,t}),$$
$$= \overline{\gamma} + \frac{1}{\hat{\mathcal{B}}}\nabla_{\overline{\mathbf{p}}_{u,t}}||\mathbf{Y}_{u,t}^{\hat{b}} - \sqrt{P_0}\boldsymbol{\iota}^{\hat{b}}(\overline{\mathbf{p}}_{u,t})\hat{\boldsymbol{\alpha}}^{\hat{b}}(\overline{\mathbf{p}}_{u,t})||^2, \tag{34}$$

where $\overline{\gamma}$, $\hat{\mathcal{B}}$, and $\hat{b}$ are the parameters of the SGD algorithm, the minibatch of training data, and the $\hat{b}$-th training sample, respectively. Therefore, the final optimised estimator $\overline{\mathbf{p}}_{u,t}$ for the u-th user at time slot t can be obtained according to the SGD algorithm. The pseudo-code of the SGD algorithm is provided in **Algorithm 1**.

3) MO-RL Learning Stage: Based on the positions, a MO-based SAC algorithm is invoked [46]. In SAC, the MDP can be written as $\langle \overline{\mathbf{S}}, \overline{\mathbf{A}}, \mathbf{p}, \overline{\mathbf{R}}, \gamma_0 \rangle$, where $\overline{\mathbf{S}}$, $\overline{\mathbf{A}}$ and γ_0 denote the state space, action space, and the discount factor, respectively. The unknown transition probability $\mathbf{p} : \overline{\mathbf{S}} \times \overline{\mathbf{S}} \times \overline{\mathbf{A}} \to [0, \infty)$ represents the probability of a transition from the current state $\overline{\mathbf{s}}_t \in \overline{\mathbf{S}}$ to the next state $\overline{\mathbf{s}}_{t+1} \in \overline{\mathbf{S}}$, while the $\overline{\mathbf{r}}$ is the reward vector bounded $\overline{\mathbf{r}} : \overline{\mathbf{S}} \times \overline{\mathbf{A}} \to [\overline{r}_{\min}, \overline{r}_{\max}]$. In the system studied, the controller acts as the agent, while the variables $\overline{\mathbf{s}}_t$, $\overline{\mathbf{a}}_t$, and $\overline{\mathbf{r}}$ can be expressed as follows: $\overline{\mathbf{s}}_t = [\theta_t, P_t]$, $\overline{\mathbf{a}}_t = [\Delta\theta, \Delta P]$, $\overline{\mathbf{r}} = [\Delta(\sum_{b=1}^B \mathcal{E}(P_{t+1}, \varepsilon)^b - \sum_{b=1}^B \mathcal{E}(P_t, \varepsilon)^b), \Delta(J_{u,t+1}(s_{u,t+1}) - J_{u,t}(s_{u,t}))]$. The variables $\zeta_\pi(\overline{\mathbf{s}}_t)$ and $\zeta_\pi(\overline{\mathbf{s}}_t, \overline{\mathbf{a}}_t)$ denote the trajectories of the state and the state-action marginals under the policy $\pi(\overline{\mathbf{a}}_t|\overline{\mathbf{s}}_t)$. A general maximum-entropy objective having the expected entropy over $\zeta_{\pi(\overline{\mathbf{s}}_t)}$ can be expressed as [46]:

$$\varpi(\pi) = \sum_{t=0}^T \mathbb{E}_{(\overline{\mathbf{s}}_t, \overline{\mathbf{a}}_t)\sim\zeta_\pi}[\overline{\mathbf{r}}(\overline{\mathbf{s}}_t, \overline{\mathbf{a}}_t) + \epsilon\mathcal{H}(\pi(\cdot|\overline{\mathbf{s}}_t))], \tag{35}$$

where ϵ is the temperature parameter. To extend (35) to the MO scenario, a linear preference vector $\mathbf{w} \subseteq \mathbf{W}$ is invoked for estimating the expected total rewards. The goal of preferences

---

Algorithm 1 SGD Algorithm with minibatch on Localisation

Input:
  Number of epochs for SGD algorithm $\mathcal{V}$, mini-batch size for SGD algorithm $\hat{\mathcal{B}}$, learning rate $\hat{\eta}$, an initial parameters of the SGD algorithm $\overline{\gamma}$.
Output: Optimised estimator $\overline{\mathbf{p}}_{u,t}$.
1: Initial positions error $\mathbf{p}_{u,t}^{\mathrm{ini}}$ for u-th user at time slot t according to the PEB.
2: **for** epoch = 1, 2, $\cdots$, $\mathcal{V}$ **do**
3:   **if** stopping criterion not met **then**
4:     **while** stopping criterion not met **do**
5:       Random sample $\hat{\mathcal{B}}$ examples from the normalised training set.
6:       Initialise $\overline{\gamma}$.
7:       **for** iteration = 1, 2, $\cdots$, $\hat{\mathcal{B}}$ **do**
8:         Compute policy gradient:
           $\overline{\gamma} = \overline{\gamma} + \frac{1}{\hat{\mathcal{B}}}\nabla_{\overline{\mathbf{p}}_{u,t}}||\mathbf{Y}_{u,t}^{\hat{b}} - \sqrt{P_0}\boldsymbol{\iota}^{\hat{b}}(\overline{\mathbf{p}}_{u,t})\hat{\boldsymbol{\alpha}}^{\hat{b}}(\overline{\mathbf{p}}_{u,t})||^2$.
9:       **end for**
10:      Update $\overline{\gamma} \leftarrow \overline{\gamma} - \hat{\eta}\overline{\gamma}$.
11:     **end while**
12:   **end if**
13: **end for**
14: Obtain the optimised positions error $\mathbf{p}_{u,t}^e$ from $\mathcal{V}$ iterations.
15: Calculate the estimated position $\mathbf{p}_{u,t} \in [\mathbf{p}_{u,t}^* - \mathbf{p}_{u,t}^e, \mathbf{p}_{u,t}^* + \mathbf{p}_{u,t}^e]$.

---

is to assign specific weights for the objectives, where the importance of each objective remains the same under each preference. The (35) can be rewritten as:

$$\varpi(\pi, \mathbf{w}) = \sum_{t=0}^T \mathbb{E}_{(\overline{\mathbf{s}}_t, \overline{\mathbf{a}}_t, \mathbf{w})\sim\zeta_\pi}[\overline{\mathbf{r}}(\overline{\mathbf{s}}_t, \overline{\mathbf{a}}_t, \mathbf{w}) + \epsilon\mathcal{H}(\pi(\cdot|\overline{\mathbf{s}}_t), \mathbf{w})]. \tag{36}$$

The parameter ϵ controls the grade of stochasticity of the optimal policy by determining the relative importance of the entropy $\mathcal{H}[\pi(\cdot|\overline{\mathbf{s}}_t, \mathbf{w})]$. However, since ϵ is subsumed into the reward by scaling it according to ϵ^{-1}, it can be omitted in the rest of the paper. As for the fixed policy, the soft Q-value can be iteratively computed and repeated by applying the Bellman backup operator $\mathcal{D}^\pi$, which can be formulated as

$$\mathcal{D}^\pi \mathbf{Q}(\overline{\mathbf{s}}_t, \overline{\mathbf{a}}_t, \mathbf{w}) = \overline{\mathbf{r}}(\overline{\mathbf{s}}_t, \overline{\mathbf{a}}_t, \mathbf{w}) + \gamma_0\mathbb{E}_{\overline{\mathbf{s}}_{t+1}\sim\zeta_\pi}[V(\overline{\mathbf{s}}_{t+1}, \mathbf{w})], \tag{37}$$

where $V(\overline{\mathbf{s}}_{t+1}, \mathbf{w}) = \mathbb{E}_{\overline{\mathbf{a}}_{t+1}\sim\pi}[\mathbf{Q}(\overline{\mathbf{s}}_t, \overline{\mathbf{a}}_t, \mathbf{w}) - \log\pi(\overline{\mathbf{a}}_t|\overline{\mathbf{s}}_t)]$ is the soft state value function. Then, we define an optimality filter $\mathcal{G}$ for (37) as follows:

$$\mathcal{G}\mathbf{Q}(\overline{\mathbf{s}}_t, \mathbf{w}) = \arg_\mathbf{Q}\sup_{\overline{\mathbf{a}}_t\in\overline{\mathbf{A}}, \mathbf{w}'\subseteq\mathbf{W}}\mathbf{w}^H\mathbf{Q}(\overline{\mathbf{s}}_t, \overline{\mathbf{a}}_t, \mathbf{w}'), \tag{38}$$

where $\arg_Q$ represents the maximum Q value operation and $\sup(\cdot)$ is the supremum operation. The mo optimality operator $\mathcal{D}$ may then defined as:

$$\mathcal{D}\mathbf{Q}(\overline{\mathbf{s}}_t, \overline{\mathbf{a}}_t, \mathbf{w}) = \overline{\mathbf{r}}(\overline{\mathbf{s}}_t, \overline{\mathbf{a}}_t, \mathbf{w}) + \gamma_0\mathbb{E}_{\overline{\mathbf{s}}_{t+1}\sim\zeta_\pi}\{\mathbb{E}_{\overline{\mathbf{a}}_{t+1}\sim\pi}[\mathcal{G}\mathbf{Q}(\overline{\mathbf{s}}_t, \overline{\mathbf{a}}_t, \mathbf{w}) - \log\pi(\overline{\mathbf{a}}_t|\overline{\mathbf{s}}_t)]\}. \tag{39}$$

Theorems 1 - 3 demonstrate the feasibility of the mo optimality operator $\mathcal{D}$ of our MO-RL.

Theorem 1. *Let $\mathbf{Q}'$ be the preferred optimal value function as follows:*

$$\mathbf{Q}'(\bar{\mathbf{s}}_t, \bar{\mathbf{a}}_t, \mathbf{w}) = \Big\{ \arg_{\mathbf{Q}} \sup_{\pi \in \Pi} \mathbf{w}^H \cdot$$

$$\mathbb{E}_{\bar{\mathbf{s}}_{t+1} \sim \zeta_\pi} \Big[\sum_{t=0}^{T} \gamma_0^t \bar{\mathbf{r}}(\bar{\mathbf{s}}_t, \bar{\mathbf{a}}_t, \mathbf{w}) \Big] - \mathbf{w}'^H \log \pi(\bar{\mathbf{a}}_t | \bar{\mathbf{s}}_t, \mathbf{w}) \Big\}. \quad (40)$$

Then, it can be shown that: $\mathbf{V}'(\bar{\mathbf{s}}_t, \bar{\mathbf{a}}_t, \mathbf{w}) = \mathcal{G} \mathbf{Q}'(\bar{\mathbf{s}}_t, \bar{\mathbf{a}}_t, \mathbf{w})$.

Proof. See Appendix A. $\square$

Theorem 2. *Let us denote $\mathbf{Q}$ and $\hat{\mathbf{Q}}$ as any two MO Q value functions. The Lipschitz condition $\bar{d}(\mathcal{G}\mathbf{Q}, \mathcal{G}\hat{\mathbf{Q}}) \leq \gamma_0 \bar{d}(\mathbf{Q}, \hat{\mathbf{Q}})$ can be obtained, where γ_0 is the discount factor.*

Proof. See Appendix B. $\square$

Theorem 3. *If $\mathcal{G}$ can be contracted by the discount factor γ_0 across the complete pseudo-metric space $\langle \mathbf{Q}, \bar{d} \rangle$, it can be shown that $\lim_{t \to \infty} \bar{d}(\mathcal{G}^t, \mathbf{Q}, \hat{\mathbf{Q}}) = 0$, where $\mathbf{Q}$ is the value space.*

Proof. See Appendix C. $\square$

As mentioned above, the state value is capable of approximating the soft value. The soft value function is trained to minimise the squared residual error, which can be expressed as:

$$\varpi_V(\mu_0) = \mathbb{E}_{\bar{\mathbf{s}}_t \sim \mathcal{F}} \Big[\frac{1}{2} (V(\bar{\mathbf{s}}_t, \mathbf{w}) -$$

$$\mathbb{E}_{\bar{\mathbf{a}}_t \sim \pi_{\mu_1}} [\mathbf{Q}_{\mu_2}(\bar{\mathbf{s}}_t, \bar{\mathbf{a}}_t, \mathbf{w}) - \log \pi_{\mu_1}(\bar{\mathbf{a}}_t | \bar{\mathbf{s}}_t, \mathbf{w})])^2 \Big], \quad (41)$$

where $\mathcal{F}$ represents the previous sample states and the actions distribution or the replay buffer. Therefore, the gradient can be estimated as:

$$\hat{\nabla}_{\mu_0} \varpi_V(\mu_0) = \nabla_{\mu_0} V_{\mu_0}(\bar{\mathbf{s}}_t) [V_{\mu_0}(\bar{\mathbf{s}}_t) -$$

$$\mathbf{Q}_{\mu_2}(\bar{\mathbf{s}}_t, \bar{\mathbf{a}}_t, \mathbf{w}) + \log \pi_{\mu_1}(\bar{\mathbf{a}}_t | \bar{\mathbf{s}}_t, \mathbf{w})], \quad (42)$$

where the actions are sampled according to the current policy. The soft Bellman residual can be calculated by:

$$\varpi_{\mathbf{Q}}(\mu_2) = \mathbb{E}_{(\bar{\mathbf{s}}_t, \bar{\mathbf{a}}_t) \sim \mathcal{F}} \Big[\frac{1}{2} (\mathbf{Q}_{\mu_2}(\bar{\mathbf{s}}_t, \bar{\mathbf{a}}_t, \mathbf{w}) -$$

$$\bar{\mathbf{R}}(\bar{\mathbf{s}}_t, \bar{\mathbf{a}}_t, \mathbf{w}) - \gamma_0 \mathbb{E}_{\bar{\mathbf{s}}_{t+1} \sim \zeta_\pi} V_{\hat{\mu}_0}(\bar{\mathbf{s}}_{t+1}, \mathbf{w}))^2 \Big], \quad (43)$$

and the stochastic gradients can be expressed as:

$$\hat{\nabla}_{\mu_2} \varpi_{\mathbf{Q}}(\mu_2) = \nabla_{\mu_2} \mathbf{Q}_{\mu_2}(\bar{\mathbf{s}}_t, \bar{\mathbf{a}}_t, \mathbf{w}) [(\mathbf{Q}_{\mu_2}(\bar{\mathbf{s}}_t, \bar{\mathbf{a}}_t, \mathbf{w}) -$$

$$\bar{\mathbf{R}}(\bar{\mathbf{s}}_t, \bar{\mathbf{a}}_t, \mathbf{w}) - \gamma_0 V_{\mu_0}(\bar{\mathbf{s}}_{t+1}, \mathbf{w}))]. \quad (44)$$

The update policy method can be expressed as:

$$\varpi_\pi(\mu_1) = \mathbb{E}_{\bar{\mathbf{s}}_t \sim \mathcal{F}, \epsilon_t \sim \mathcal{N}} [\log \pi_{\mu_1}(\bar{f}_{\mu_1}(\epsilon_t; \bar{\mathbf{s}}_t) | \bar{\mathbf{s}}_t, \mathbf{w}) -$$

$$\mathbf{Q}_{\mu_2}(\bar{\mathbf{s}}_t, \bar{f}_{\mu_1}(\epsilon_t; \bar{\mathbf{s}}_t), \mathbf{w})], \quad (45)$$

and the approximate gradient can be formulated as:

$$\hat{\nabla}_{\mu_1} \varpi_\pi(\mu_1) = \nabla_{\mu_1} \log \pi(\bar{\mathbf{a}}_t | \bar{\mathbf{s}}_t, \mathbf{w}) + [\nabla_{\bar{\mathbf{a}}_t} \log \pi(\bar{\mathbf{a}}_t | \bar{\mathbf{s}}_t, \mathbf{w}) -$$

$$\nabla_{\bar{\mathbf{a}}_t} \mathbf{Q}(\bar{\mathbf{s}}_t, \bar{\mathbf{a}}_t, \mathbf{w})] \nabla_{\mu_1} \bar{f}_{\mu_1}(\epsilon_t; \bar{\mathbf{s}}_t), \quad (46)$$

where ϵ_t and $\bar{f}_{\mu_1}(\epsilon_t; \bar{\mathbf{s}}_t)$ denote an input noise vector and a neural network transformation, respectively. By optimising these gradients, we aim for approaching the PF. Therefore, the associated gradient optimisation can be regarded as multiple gradient optimisation constructed for all objectives. According to the Karush-Kuhn-Tucker (KKT) conditions, there exists a set of $\nu_1, \nu_2, \cdots, \nu_M$ for M objectives so that:

- $\nu_1, \nu_2, \cdots, \nu_M \geq 0$.
- $\sum_{m=1}^{M} \nu_m = 1$.
- $\sum_{m=1}^{M} \nu_m \hat{\nabla}_{\mu_0} \varpi_V^m(\mu_0) = 0$, $\sum_{m=1}^{M} \nu_m \hat{\nabla}_{\mu_2} \varpi_{\mathbf{Q}}^m(\mu_2) = 0$, $\sum_{m=1}^{M} \nu_m \hat{\nabla}_{\mu_1} \varpi_\pi^m(\mu_1) = 0$.

Since the objectives may have values of the different scales and multiple gradient optimisation is sensitive to the different ranges, the range of gradient function has been limited to [0, 1].

Definition 1. *A solution $\mu_{(\bar{u})}^1$ dominates a solution $\mu_{(\bar{u})}^2$, $\bar{u} = \{0, 1, 2\}$ if for all objectives they satisfying: $\hat{\nabla}_{\mu_0^1} \varpi_V^m(\mu_0^1) \leq \hat{\nabla}_{\mu_0^2} \varpi_V^m(\mu_0^2)$, $\hat{\nabla}_{\mu_2^1} \varpi_{\mathbf{Q}}^m(\mu_2^1) \leq \hat{\nabla}_{\mu_2^2} \varpi_{\mathbf{Q}}^m(\mu_2^2)$, $\hat{\nabla}_{\mu_1^1} \varpi_\pi^m(\mu_1^1) \leq \hat{\nabla}_{\mu_1^2} \varpi_\pi^m(\mu_1^2)$, while there is at least one objective satisfying $\hat{\nabla}_{\mu_0^1} \varpi_V^n(\mu_0^1) \leq \hat{\nabla}_{\mu_0^2} \varpi_V^n(\mu_0^2)$, $\hat{\nabla}_{\mu_1^1} \varpi_{\mathbf{Q}}^n(\mu_1^1) \leq \hat{\nabla}_{\mu_2^2} \varpi_{\mathbf{Q}}^n(\mu_2^2)$, $\hat{\nabla}_{\mu_1^1} \varpi_\pi^n(\mu_1^1) \leq \hat{\nabla}_{\mu_1^2} \varpi_\pi^n(\mu_1^2)$, $\forall m, n \in \{1, 2, \cdots, M\}$.*

Definition 2. *A solution $\mu_{(\bar{u})}^1$ constitutes a PO solution, if no other solution $\mu_{(\bar{u})}^2$ dominates $\mu_{(\bar{u})}^2$, $\bar{u} = \{0, 1, 2\}$.*

Definition 3. *The full set of non-dominated solutions constitutes a PO set.*

The solution that satisfies the **definitions 1 - 3** is defined as a PO solution. Thus, the optimisation problem can be formulated as follows:

$$\min_{\nu_1^0, \cdots, \nu_M^0} \Big\{ \Big| \Big| \sum_{m=1}^{M} \nu_m^0 \hat{\nabla}_{\mu_0} \varpi_V^m(\mu_0) \Big| \Big|_2^2, \Big| \sum_{m=1}^{M} \nu_m^0 = 1, \nu_m^0 \geq 0 \Big\},$$

$$\min_{\nu_1^1, \cdots, \nu_M^1} \Big\{ \Big| \Big| \sum_{m=1}^{M} \nu_m^1 \hat{\nabla}_{\mu^1} \varpi_\pi^m(\mu_1) \Big| \Big|_2^2 \Big| \sum_{m=1}^{M} \nu_m^1 = 1, \nu_m^1 \geq 0 \Big\},$$

$$\min_{\nu_1^2, \cdots, \nu_M^2} \Big\{ \Big| \Big| \sum_{m=1}^{M} \nu_m^2 \hat{\nabla}_{\mu_2} \varpi_{\mathbf{Q}}^m(\mu_2) \Big| \Big|_2^2 \Big| \sum_{m=1}^{M} \nu_m^2 = 1, \nu_m^2 \geq 0 \Big\}, \quad (47)$$

where $||\cdot||_2^2$ and $\nabla_{(\cdot)}$ denote the L2 norm and gradient descent (GD) operator. Since it has two objectives in problem (21), Equation (47) can be simplified as:

$$\min_{\nu^0 \in [0,1]} ||\nu^0 \hat{\nabla}_{\mu_0} \varpi_V^1(\mu_0) + (1 - \nu^0) \hat{\nabla}_{\mu_0} \varpi_V^2(\mu_0)||_2^2,$$

$$\min_{\nu^1 \in [0,1]} ||\nu^1 \hat{\nabla}_{\mu_1} \varpi_\pi^1(\mu_1) + (1 - \nu^1) \hat{\nabla}_{\mu_1} \varpi_\pi^2(\mu_1)||_2^2,$$

$$\min_{\nu^2 \in [0,1]} ||\nu^2 \hat{\nabla}_{\mu_2} \varpi_{\mathbf{Q}}^1(\mu_2) + (1 - \nu^2) \hat{\nabla}_{\mu_2} \varpi_{\mathbf{Q}}^2(\mu_2)||_2^2. \quad (48)$$

The optimisation problem (48) is equivalent to finding a minimum-norm point in the convex hull, which is a convex quadratic problem subject to linear constraints. Thus, an analytical solution of Equation (48) can be formulated as:

$$\nu^0 = \Big\{ \frac{[\hat{\nabla}_{\mu_0} \varpi_V^2(\mu_0) - \hat{\nabla}_{\mu_0} \varpi_V^1(\mu_0)]^T \hat{\nabla}_{\mu_0} \varpi_V^2(\mu_0)}{||\hat{\nabla}_{\mu_0} \varpi_V^1(\mu_0) - \hat{\nabla}_{\mu_0} \varpi_V^2(\mu_0)||_2^2} \Big\}_{[0,1]},$$

$$\nu^1 = \left\{ \frac{[\hat{\nabla}_{\mu_1}\varpi_\pi^2(\mu_1) - \hat{\nabla}_{\mu_1}\varpi_\pi^1(\mu_1)]^T\hat{\nabla}_{\mu_1}\varpi_\pi^2(\mu_1)}{||\hat{\nabla}_{\mu_1}\varpi_\pi^1(\mu_1) - \hat{\nabla}_{\mu_1}\varpi_\pi^2(\mu_1)||_2^2} \right\}_{[0,1]},$$

$$\nu^2 = \left\{ \frac{[\hat{\nabla}_{\mu_2}\varpi_Q^2(\mu_2) - \hat{\nabla}_{\mu_2}\varpi_Q^1(\mu_2)]^T\hat{\nabla}_{\mu_2}\varpi_Q^2(\mu_2)}{||\hat{\nabla}_{\mu_2}\varpi_Q^1(\mu_2) - \hat{\nabla}_{\mu_2}\varpi_Q^2(\mu_2)||_2^2} \right\}_{[0,1]},$$

$$(49)$$

where $\{\}_{[0,1]}$ represents clipping ν to $[0,1]$. The pseudo-code of the MO-SAC algorithm is provided in **Algorithm 2**[4].

Algorithm 2 MO-SAC Algorithm

Input:

The number of iterations for MO-RL learning phase $\mathcal{U}$, mini-batch size for MO-RL learning phase $\overline{\mathcal{B}}$, replay memory for MO-RL learning phase $\mathcal{D}$, preference space $\overline{\Omega}$, an exponentially moving average of the value network weights $\hat{\mu}_0$.

Output: Tuple $\{(\bar{\mathbf{s}}_t, \bar{\mathbf{a}}_t, \bar{\mathbf{r}}(\bar{\mathbf{s}}_t, \bar{\mathbf{a}}_t, \mathbf{w}), \bar{\mathbf{s}}_{t+1}, \mathbf{w})\}$, $t = \{1, 2, \cdots, T\}$.

1: **for** size = $1, 2, \cdots, \overline{\mathcal{B}}$ **do**
2:    Copy the parameters χ_n to the SAC networks.
3:    **for** Iteration = $1, 2, \cdots, \mathcal{U}$ **do**
4:       Sample $\mathbf{w}$ from preference space $\overline{\Omega}$.
5:       Sample $\bar{\mathbf{a}}_t$ from $\pi_{\mu_2}(\bar{\mathbf{a}}_t|\bar{\mathbf{s}}_t, \mathbf{w})$.
6:       Sample normalised $\bar{\mathbf{s}}_{t+1}$ from $\zeta_\pi(\bar{\mathbf{s}}_{t+1}|\bar{\mathbf{s}}_t, \bar{\mathbf{a}}_t, \mathbf{w})$.
7:       Sample from $\{(\bar{\mathbf{s}}_t, \bar{\mathbf{a}}_t, \bar{\mathbf{r}}(\bar{\mathbf{s}}_t, \bar{\mathbf{a}}_t, \mathbf{w}), \bar{\mathbf{s}}_{t+1}, \mathbf{w})\}$.
8:    **end for**
9:    Update $\mu_0 \leftarrow \mu_0 - z_V\hat{\nabla}_{\mu_0}\varpi_V(\mu_0)$.
10:   Update $\mu_1 \leftarrow \mu_1 - z_\pi\hat{\nabla}_{\mu_1}\varpi_\pi(\mu_1)$.
11:   Update $\mu_2 \leftarrow \mu_2 - z_\mathbf{Q}\hat{\nabla}_{\mu_2}\varpi_\mathbf{Q}(\mu_2)$.
12:   Update $\hat{\mu}_0 \leftarrow \tau_0\mu_0 + (1 - \tau_0)\hat{\mu}_0$.
13: **end for**

B. Meta-learning Stage

In the meta-policy learning stage, the goal is to optimise the meta-policy parameters $\hat{\chi} = \{\hat{\mu}_0, \hat{\mu}_1, \hat{\mu}_2\}$. In the model studied, the different initial positions of Metaverse users can be defined as different tasks. There are two sub-stages for the meta-learning stage: the training and the adaptation sub-stages.

1) Training Sub-stage: Assume that there are N tasks (MO-RL models) for training, and the support set as well as the query set of each task are defined as $\mathbb{A}_n$ and $\mathbb{B}_n$, $n \in 1, 2, \cdots, N$. In each training iteration, a weight vector $\mathbf{W} = W_1, W_2, \cdots, W_N$ is randomly sampled to create N MO-RL models. In the inner-loop[5] update, the optimised gradient can be specified for the n-th task as:

$$\chi_n = \begin{cases} \mu_{0,n} = \arg\min_{\varpi_V}\hat{\nabla}_{\mu_{0,n}}\varpi_V(\mu_{0,n}, \mathbb{A}_n), \\ \mu_{1,n} = \arg\min_{\varpi_\pi}\hat{\nabla}_{\mu_{1,n}}\varpi_\pi(\mu_{1,n}, \mathbb{A}_n), \\ \mu_{2,n} = \arg\min_{\varpi_Q}\hat{\nabla}_{\mu_{2,n}}\varpi_Q(\mu_{2,n}, \mathbb{A}_n), \end{cases} \quad (50)$$

---

[4]The action does not need normalisation, since the action range is usually defined as [-1, 1] so its variance and mean can approach 0 and 1 to follow the normal distribution. The reward function can not be normalised, since the Bellman equation-based reinforcement learning does not allow reward subtract a non-zero constant, since that would destroy the reward function of the environment itself.

[5]In the inner loop, the agent learns task-specific network parameters by performing one gradient step on a task-specific loss.

---

where $\mu_{0,n}$, $\mu_{0,n}$, $\mu_{0,n}$ are the network weights of squared residual error, soft Bellman residual, and update policy method for the n-th task. Then, the update process of meta-policy parameters χ_n of the n-th MO-RL model can be expressed as follows:

$$\chi_n^{\hat{T}+1} = \begin{cases} \mu_{0,n}^{\hat{T}+1} \leftarrow \mu_{0,n}^{\hat{T}} - z_V\hat{\nabla}_{\mu_{0,n}}\arg\min_{\varpi_V}\varpi_V(\mu_{0,n}, \mathbb{A}_n), \\ \mu_{1,n}^{\hat{T}+1} \leftarrow \mu_{1,n}^{\hat{T}} - z_\pi\hat{\nabla}_{\mu_{1,n}}\arg\min_{\varpi_\pi}\varpi_\pi(\mu_{1,n}, \mathbb{A}_n), \\ \mu_{2,n}^{\hat{T}+1} \leftarrow \mu_{2,n}^{\hat{T}} - z_\mathbf{Q}\hat{\nabla}_{\mu_{2,n}}\arg\min_{\varpi_\mathbf{Q}}\varpi_\mathbf{Q}(\mu_{2,n}, \mathbb{A}_n), \end{cases}$$

$$(51)$$

where z_V, z_π, and $z_\mathbf{Q}$ are the learning rates for the inner-loop update. Then, for the outer-loop[6] update, the optimised parameters can be expressed as follows:

$$\hat{\chi} = \begin{cases} \hat{\mu}_0 = \arg\min_{\varpi_V}\hat{\nabla}_{\hat{\mu}_0}\varpi_V(\hat{\mu}_0, \mathbb{A}_n), \\ \hat{\mu}_1 = \arg\min_{\varpi_\pi}\hat{\nabla}_{\hat{\mu}_1}\varpi_\pi(\hat{\mu}_1, \mathbb{A}_n), \\ \hat{\mu}_2 = \arg\min_{\varpi_Q}\hat{\nabla}_{\hat{\mu}_2}\varpi_Q(\hat{\mu}_2, \mathbb{A}_n), \end{cases} \quad (52)$$

and the update process of meta-policy parameters $\hat{\chi}$ can be expressed as follows:

$$\hat{\chi}^{\hat{T}+1} =$$
$$\begin{cases} \hat{\mu}_0^{\hat{T}+1} \leftarrow \hat{\mu}_0^{\hat{T}} - \sum_{n=1}^N \hat{z}_V\hat{\nabla}_{\mu_0}\arg\min_{\varpi_V}\varpi_V(\mu_{0,n}, \mathbb{B}_n), \\ \hat{\mu}_1^{\hat{T}+1} \leftarrow \hat{\mu}_1^{\hat{T}} - \sum_{n=1}^N \hat{z}_\pi\hat{\nabla}_{\mu_1}\arg\min_{\varpi_\pi}\varpi_\pi(\mu_{1,n}, \mathbb{B}_n), \\ \hat{\mu}_2^{\hat{T}+1} \leftarrow \hat{\mu}_2^{\hat{T}} - \sum_{n=1}^N \hat{z}_\mathbf{Q}\hat{\nabla}_{\mu_2}\arg\min_{\varpi_\mathbf{Q}}\varpi_\mathbf{Q}(\mu_{2,n}, \mathbb{B}_n), \end{cases}$$

$$(53)$$

where $\hat{z}_V$, $\hat{z}_\pi$, and $\hat{z}_\mathbf{Q}$ are the learning rates for the inner-loop update.

2) Adaptation Sub-stage: When the training is finished, the trained model aims for adapting to the new task based on the optimised parameters. The well-trained meta-model is able to achieve fast adaptation to the new task, where the parameters χ of the new task can be updated by:

$$\chi = \begin{cases} \mu_0 \leftarrow \mu_0 - z_V\hat{\nabla}_{\mu_0}\arg\min_{\varpi_V}\varpi_V(\mu_0), \\ \mu_1 \leftarrow \mu_1 - z_\pi\hat{\nabla}_{\mu_1}\arg\min_{\varpi_\pi}\varpi_\pi(\mu_1), \\ \mu_2 \leftarrow \mu_2 - z_\mathbf{Q}\hat{\nabla}_{\mu_2}\arg\min_{\varpi_\mathbf{Q}}\varpi_\mathbf{Q}(\mu_2). \end{cases} \quad (54)$$

When prompt adaptation of the meta-model achieves convergence, the resultant approximate PF may be depicted. The pseudo-code of the meta-learning-based MO-SAC algorithm is provided in **Algorithm 3**.

C. Optimality Analysis for Meta Learning-based MO-RL Algorithm

To evaluate the optimality, the algorithms will be analysed into two aspects: 1) The structure of the algorithms and 2) The complexity of the algorithms.

1) Algorithm Structures: For the structure, we can observe that the proposed SGD-based algorithm depends on the learning rate. If the learning rate is too small, the convergence

---

[6]In the outer loop, the model parameters from before the inner loop update are updated to reduce the loss after the inner loop update on the individual tasks.

Algorithm 3 Meta Learning-based MO-SAC Algorithm

Input:

    Mini-batch size for meta-training phase $\tilde{\mathcal{B}}$, number of iterations for meta-training stage $\mathcal{U}_{it}$, replay memory for meta-training stage $\mathcal{D}_t$, number of iterations for meta-adaptation stage $\mathcal{U}_a$, replay memory for meta-adaptation stage $\mathcal{D}_a$, inner-loop learning rates $(z_V, z_\pi, z_{\mathbf{Q}})$, outer-loop learning rates $(\hat{z}_V, \hat{z}_\pi, \hat{z}_{\mathbf{Q}})$, number of training tasks N, number of adaptation tasks N_a.

Output: The approximate PFs for N_a tasks.

Meta Training Stage:

1: **Initialise:** meta-policy parameters $\hat{\chi}$, MO-RL parameters χ_n for the n-th training task, support set $\mathbb{A}_n$ and the query set $\mathbb{B}_n$ of each task, task set $\{\mathcal{N}\}$.

2: **for** mini-batch size $\tilde{\mathcal{B}}$ **do**

3:     Sample N tasks from task set $\{\mathcal{N}\}$.

4:     **for** iterations $= 1, 2, \cdots, \mathcal{U}_t$ **do**

5:         **for** training tasks $= 1, 2, \cdots, N$ **do**

6:             Obtain the positions of users by **Algorithm 1**.

7:             Determine the pairing index $s_{b,u,t}$ according to the distance according to the positions of CBSs and users.

8:             Generate tuple $\{(\bar{\mathbf{s}}_t, \bar{\mathbf{a}}_t, \bar{\mathbf{r}}(\bar{\mathbf{s}}_t, \bar{\mathbf{a}}_t, \mathbf{w}), \bar{\mathbf{s}}_{t+1}, \mathbf{w})\}$, $t = \{1, 2, \cdots, T\}$ from **Algorithm 2**.

9:             Store tuple $\{(\bar{\mathbf{s}}_t, \bar{\mathbf{a}}_t, \bar{\mathbf{r}}(\bar{\mathbf{s}}_t, \bar{\mathbf{a}}_t, \mathbf{w}), \bar{\mathbf{s}}_{t+1}, \mathbf{w})\}$, $t = \{1, 2, \cdots, T\}$ into memory $\mathcal{D}_t$.

10:            Sample $(\bar{\mathbf{s}}_n, \bar{\mathbf{a}}_n, \bar{\mathbf{r}}_n, \bar{\mathbf{s}}'_n)$ from $\mathcal{D}_t$ as $\mathbb{A}_n$.

11:            **for** iterations $= 1, 2, \cdots, \mathcal{U}_{it}$ **do**

12:                Update χ_n by invoking (51).

13:            **end for**

14:            Calculate loss function of the n-th task by $\mathbb{B}_n$.

15:         **end for**

16:         Update χ by invoking (53).

17:     **end for**

18: **end for**

Meta Adaptation Stage:

19: **for** iterations $= 1, 2, \cdots, \mathcal{U}_a$ **do**

20:     **for** adaptation tasks $= 1, 2, \cdots, N_a$ **do**

21:         Copy χ to the n_a-th task.

22:         Generate experience tuple according to **line 8**.

23:         Store $\mathcal{D}_a$ into replay memory $\mathcal{D}_a$.

24:         Update χ by invoking (54).

25:         Obtain approximate PF for the n_a-th task.

26:     **end for**

27: **end for**

---

is slow, and if it is too large, it will oscillate near the minimum value. Hence, the proposed SGD-based algorithm is near-optimal. Moreover, the MO-SAC algorithm introduces the neural networks, where the optimised weights and bias are both approaching the global optimal results but they are difficult to achieve. Thus, we conclude that this algorithm is near-optimal.

2) Algorithm Complexity: The complexity of the proposed meta-learning-based MO-RL algorithm is dominated by the MO-RL stage and meta-learning stage. The MO-RL stage has four parameters from the SGD algorithm and five parameters

from the MO-SAC algorithm, which are the size of the input data $\tilde{D}$, the number of epochs in the SGD algorithm $\mathcal{V}$, the minibatch size $\hat{\mathcal{B}}$ of the SGD algorithm, the number of objectives $\tilde{N}$, the size of perference space $\tilde{\Omega}$, the size of state space $\tilde{S}_{\tilde{n}}$ of the objective $\tilde{n}$, the size of action space $\tilde{A}_{\tilde{n}}$ of the objective $\tilde{n}$, the size of reward function $\tilde{R}$, the mini-batch size of the MO-SAC algorithm $\overline{\mathcal{B}}$, and the number of iteration $\mathcal{U}$. So the complexity of the MO-RL stage is on the order of $\mathrm{O}[\tilde{D}\mathcal{V}\hat{\mathcal{B}}\tilde{R}\overline{\mathcal{B}}\mathcal{U}\tilde{\Omega}(\tilde{S}_{\tilde{n}}\tilde{A}_{\tilde{n}})^{\tilde{N}}]$. The compelxity of the Meta-learning stage is dominated by the MO-RL stage, the mini-batch size for meta-training stage $\tilde{\mathcal{B}}$, the number of iterations of the meta-training stage $\mathcal{U}_{it}$, the replay memory of the meta-training stage $\mathcal{D}_t$, the number of iterations for meta-adaptation stage $\mathcal{U}_a$, the replay memory for meta-adaptation stage $\mathcal{D}_a$, and the number of training tasks N. So the complexity of the meta-learning-based MO-RL can be expressed as $\mathrm{O}[\tilde{D}\mathcal{V}\hat{\mathcal{B}}\tilde{R}\overline{\mathcal{B}}\mathcal{U}\tilde{\Omega}(\tilde{S}_{\tilde{n}}\tilde{A}_{\tilde{n}})^{\tilde{N}}(\tilde{\mathcal{B}}\mathcal{U}_{it}\mathcal{U}_a\mathcal{D}_t\mathcal{D}_a)^N]$. According to the system studied, the complexity of the meta Learning-based MO-RL algorithm is $\mathrm{O}[3\mathcal{V}\hat{\mathcal{B}}\overline{\mathcal{B}}\mathcal{U}(|\bar{\mathbf{S}}_{\tilde{n}}||\bar{\mathbf{A}}_{\tilde{n}}|)^2(\tilde{\mathcal{B}}\mathcal{U}_{it}\mathcal{U}_a\mathcal{D}_t\mathcal{D}_a)^N]$.

IV. NUMERICAL RESULTS

In this section, we investigate the performance of RIS-assisted wireless Metaverse networks. The users are assumed to be randomly and uniformly placed in a 10m $\times$ 10m rectangular area, as shown in Fig. 1. Since we assume that the vertical height ratio of the ceiling and the users is huge enough to avoid blockage between the RIS and the users, the heights of the ceiling and the users are set to 6m and 1.7m. Additionally, the total number of CBSs and users are chosen to be 4 and 4, while $P_{\max}$ and ε_m are 100mW and 10^{-5}. The noise power of each user is fixed to -110dBm. In this model, the definition of reliability in the THz regime refers to the average number of successfully served Metaverse users. At each time slot t, if $\mathcal{K}_{u,t}(s_{u,t}) = 1$, the u-th will be defined as a reliable user. Thus, the reliability of the u-th user at a time slot t can be expressed as follows:

$$\upsilon_{u,t}(\mathbf{s}_{u,t}) = \mathcal{K}_{u,t}(s_{u,t}) \vee \mathcal{K}_{u,t-1}(s_{u,t-1}), \qquad (55)$$

where $\vee$ denotes the logical "or" operators. The newly served users at time slot t will be

$$\overline{\upsilon}_t(\mathbf{s}_{u,t}) = \{u | \upsilon_{u,t}(\mathbf{s}_{u,t}) = 1, \upsilon_{u,t-1}(\mathbf{s}_{u,t-1}) = 0\}, \quad (56)$$

then, the number of successfully served users in T time slots can be expressed by:

$$\Xi_{1:T}(\mathbf{s}_{1:T}) = \sum_{t=1}^{T} |\overline{\upsilon}_t(\mathbf{s}_{u,t})|. \qquad (57)$$

The seamless experience of this user can be guaranteed, when the user is served successfully at each time slot. Therefore, the range of maximum tolerable transmission delay Δt can be set between 0 to t and it is further determined according to different requirements of practical scenarios. In the model studied, we strike a trade-off between the total service cost and maximum transmission latency among all users of the communication system upon one time slot constraint and then determine Δt according to the different requirements. The

TABLE I: Simulation parameters for our meta learning-based MO-SAC algorithm and communication networks

| Parameter	Description	Value
$\hat{\mathcal{B}}$	Mini-batch size for meta-training phase	64
$\mathcal{U}_t$	Number of iterations for meta-training stage	10000
$\mathcal{D}_t$	Replay memory for meta-training stage	1000000
$\mathcal{U}_{it}$	Number of iterations for meta-adaptation stage	10000
$\mathcal{D}_a$	Replay memory for meta-adaptation stage	1000000
$(z_V, z_\pi, z_\mathbf{Q})$	Inner-loop learning rates	$(3, 3, 3) \times 10^{-4}$
$(\hat{z}_V, \hat{z}_\pi, \hat{z}_\mathbf{Q})$	Outer-loop learning rates	$(3, 3, 3) \times 10^{-3}$
$\mathcal{U}$	Number of iterations for MO-RL learning phase	1000
$\overline{\mathcal{B}}$	Mini-batch size for MO-RL learning phase	32
$\mathcal{D}$	Replay memory for MO-RL learning phase	10000
λ_c	MmWave carrier wavelength	5mm [17]
f	THz carrier frequency	0.2THz [40]
$P_{\max}$	Maximum transmit power	200mW = 23.01dBm
h_L	Height of LBS	2m
h_S	Height of CBS	2.5m
R_g	Length of each grid	0.05m
C_{meta}	Cost of MSP	150
t	Time slot	2ms

variable Δt guides choosing the permitted transmission delay of the entire system and associated acceptable total services cost, and further determine the reliability of each user in a time period $\mathcal{T}$. Then, let us now discuss the performance of the proposed Meta MO-SAC algorithm. As for the performance of the localisation stage, the **Adam algorithm** [47], **Batch Gradient descent (BGD) algorithm** [48], and **SGD algorithm** [49] is used as the benchmark. As for the performance of the communication phase, the fixed weights of the SAC and MO-SAC algorithm dispensing with the meta-framework are invoked as benchmarkers. The parameters of the meta-learning-based MO-SAC network and communication network are summarised at a glance in Table. I. The channel coefficients remain the same at each step, but they are different in each episode.

A. Performance of Localisation Phase

Fig. 3 characterises the PEB by the Cramér-Rao lower bounds and the SGD algorithm relying on a minibatch, where we have SNR = 5dB, $K = 128$. It is observed that a user position close to the RIS and LBS has a high localisation accuracy. The SGD algorithm relying on a minibatch achieves an excellent accuracy below 10^{-3}m, which is more accurate than the PEB based on the Cramér-Rao lower bounds. This is because as the distance increases, the signal will also be severely attenuated. Hence the users who are farther away receive poorer signal quality than users who are closer. Additionally, localisation relies on multi-path transmission, and the increase in distance will inevitably increase the noise, which will also affect the localisation accuracy. To further investigate the performance of the proposed SGD algorithm with a minibatch, we compare the root-mean-square error (RMSE) of the estimated positions of different received SNRs and at the different clock offset thresholds shown in Fig. 4. Observe in Fig. 4 that the RMSE is reduced upon increasing the SNR and
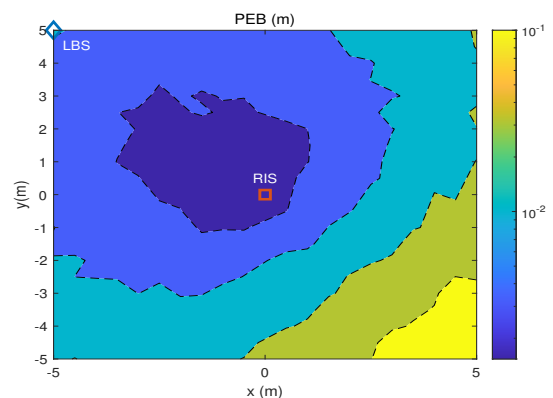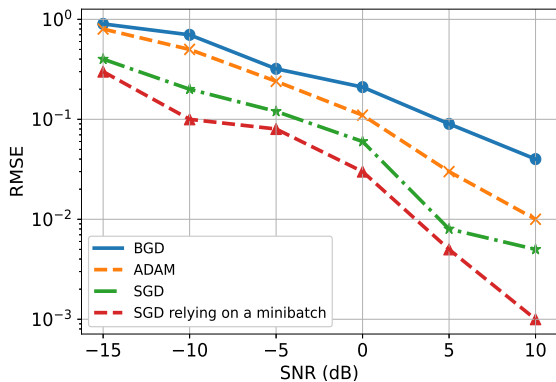

(a) PEB by Cramér-Rao lower bounds.
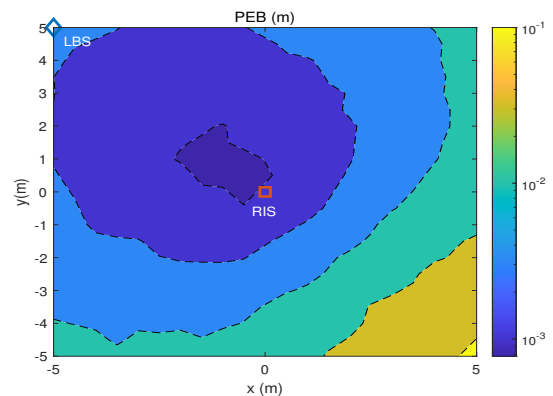
(b) PEB by SGD algorithm relying on a minibatch.

Fig. 3: PEB by the different algorithms, SNR = 5dB, $K = 128$.

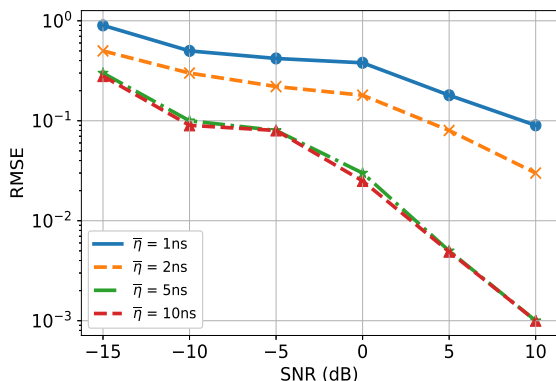(a) RMSE of the estimated positions vs the SNR for different algorithms.

(b) RMSE of the estimated positions vs the SNR parameterised by different thresholds $\overline{\eta}$.

Fig. 4: RMSE of estimated positions vs the received SNR parameters of Table. I.

(a) Training curves under different algorithms, $K = 128$.

(b) Adaptation curves under different algorithms, $K = 128$, $N_a = 1$.

Fig. 5: The reward vs episode-index of the proposed algorithm compared to the benchamarkers.

that compared to the BGD, ADAM, and SGD algorithms, the proposed SGD algorithm relying on a minibatch exhibits better performance. Furthermore, when the SNR achieves 10dB, the RMSE is able to arrive at 10^{-3}, and when the maximum tolerable clock offset increases, the RMSE is decreased. This is because the minibatch setting provides the learning experience required, which increases the accuracy of the localisation, and having an increased threshold is capable of increasing the fault tolerance for localisation.

B. Performance of the Meta-Learning-Based MO-SAC Algorithm

Fig. 5 characterises the performance of the proposed meta-learning-based MO-SAC algorithm compared to the benchmarkers. Fig. 5(a) illustrates the convergence of the proposed meta-MO-SAC algorithms. The moving reward is the smooth operation for the reward to clearly depict the learning trend of the algorithms. Compared to the benchmarkers, it is observed that the proposed meta-MO-SAC exhibits a slower convergence. This is because of having large quantities of training samples inevitable increases the computational complexity. For the case of fixed weights, although it converges the fastest, it gleans the least reward, so it may not perform as well as the MO algorithm. In the case of the MO-SAC operating without a meta-framework, it converges faster than the meta-
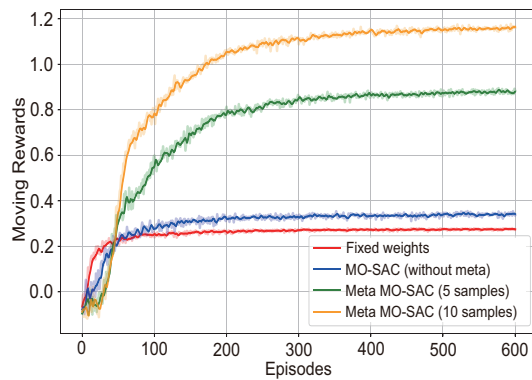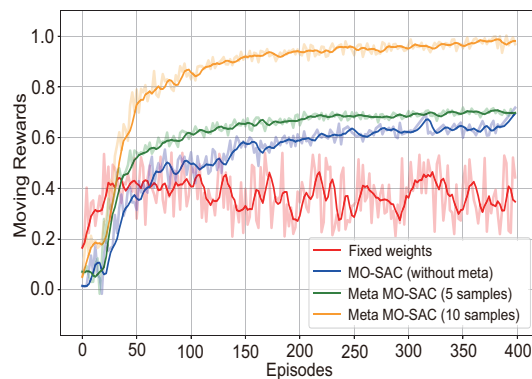
learning-based MO-SAC, but its performance has to be further proved according to the adaptation stage. Fig. 5(b) verifies the performance of the trained model when the number of adaptation tasks N_a is set to 1. It is observed that the meta-learning-based MO-SAC algorithm converges promptly to the new task in fewer episodes than the benchmarkers, while the model trained with more training tasks N is capable of exhibiting faster adaption speed to the new tasks. Additionally, the fixed-weights scenario diverges and fails to adapt to new tasks. The MO-SAC algorithm exhibits a slow adaption speed for new tasks, and it is outperformed by the meta-learning-based MO-SAC. Therefore, the adaptation stage illustrates the performance of our proposed meta-algorithm-based MO-SAC algorithms, where the adaption speed can be readily improved upon harnessing longer training.

C. Approximate PF

Fig. 6 depicts the approximate PFs based on the adaptation stage, since it has constrained options. As mentioned in the formulated problem, our goal is to strike a transmission latency versus total service cost tradeoff. As shown in Fig. 6(a), the approximate PF found by the proposed meta-MO-SAC algorithm approaches the coordinates, which covers more feasible solutions than the benchmarkers. Additionally, the model trained based on numerous training samples outperforms that based on a few training samples. As shown in Fig. 6(b),
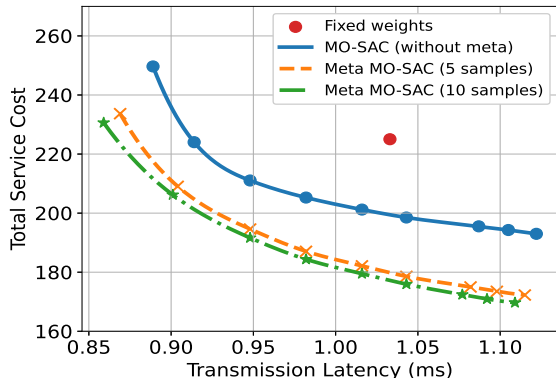
(a) Approximate PF of total service cost vs transmission latency by proposed algorithm compared to the benchmarkers, $K = 128$.
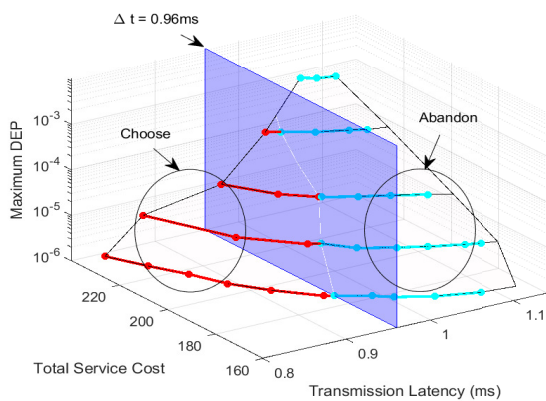
(b) Approximate PF of Total Service Cost vs Transmission Latency by proposed Meta-MO-SAC algorithm constraint by the different maximum DEP values, $K = 128$.

Fig. 6: The Approximate PF of Total Service Cost vs Transmission Latency by the proposed algorithm compared to the benchamarkers.

the approximate PF changes with maximum DEP constraint. With the increase of the maximum DEP, the coverage of PF shrinks, which indicates that the DEP plays an important role in determining the coverage of PF. This is because when the maximum DEP decreases, the achievable data rate range becomes wider and further illustrates the low latency achieved for the entire system. Then, according to Equations (55) - (57), we have set a maximum tolerable transmission delay for the entire system to characterise the requirements of a practical scenario. For example, when the maximum tolerable transmission delay Δt is set to 0.96ms, the right-hand side of the approximate PF in Fig. 6(b) will be abandoned and the transmission latency below 0.96ms is chosen. Furthermore, the trend is that the PF becomes shorter with the increase of the maximum DEP and there is no obvious tradeoff between the total service cost and the transmission latency. Hence, the associated MO optimization can be formulated as a single objective optimization, when the maximum DEP is high enough. This insight provides a guideline for Metaverse deployments by MSP and NSP.

D. Different Number of Elements in the RIS

Fig. 7 quantifies the impact of the different number of RIS elements on the system. According to the settings of Δt in Fig. 6(b), when the maximum tolerable transmission latency increases, the range of PF increases. Having a high latency tolerable increases the probability of users being served successfully. Therefore, the number of users served successfully increases, as the maximum tolerable transmission latency increases. In this case, when the number of elements in the RIS increases, the reliability is improved, because the channel gains of a communication system are improved. Therefore, these results demonstrate that RISs have a positive impact on a communication system.
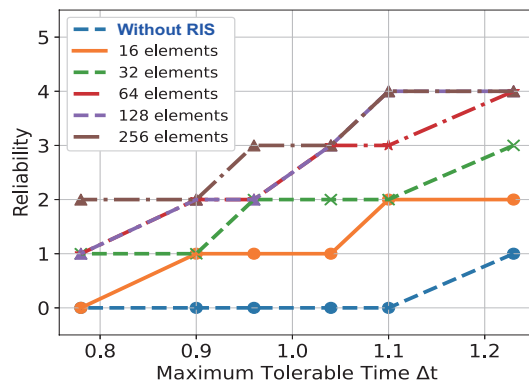

Fig. 7: Reliability vesus Maximum Latency Δt, DEP = 10^{-5}, $N_a = 1$.

V. CONCLUSIONS

Two KPIs with conflicting relationships, i.e., total service cost and transmission latency, were considered in URLLC-enabled Metaverse. Our designed objective was to simultaneously minimise the total service cost and transmission latency in URLLC-based Metaverse networks approaching PF, while jointly optimising the transmit power, the RIS phase shifts, and the decoding error probability. To solve this problem, a meta-learning-based position-dependent MO-SAC algorithm was proposed. The core idea of the proposed algorithm was to dynamically assign the weights each time, when a network update process takes place. Additionally, there were two key points to assist the algorithm: the SGD algorithm with a mini-batch was harnessed for determining the positions of the users, while the MO-SAC algorithm was invoked for learning a policy for each task. The numerical results showed that: 1) The proposed solution struck a compelling tradeoff between the total service cost and transmission latency, which provided a candidate group of optimal solutions for practical scenarios. 2) The proposed meta-learning-based MO-SAC algorithm was capable of achieving a faster adaptation to new wireless environments than the benchmarkers. 3) The approximate PF discovered the relationships among the KPIs for the Metaverse, which provided guidelines for its deployment. Furthermore, extending our ML algorithm to the general system studied is the main focus in the next step of research.

APPENDIX A: PROOF OF THEOREM 1

Theorem 1. *Let* $\mathbf{Q}$'' *be the preferred optimal value function formulated as follows:*

$$\mathbf{Q}'(\overline{\mathbf{s}}_t,\overline{\mathbf{a}}_t,\mathbf{w}) = \Big\{\arg_{\mathbf{Q}}\sup_{\pi\in\Pi}\mathbf{w}^H\mathbb{E}_{\overline{\mathbf{s}}_{t+1}\sim\zeta_\pi}\Big[\sum_{t=0}^{T}\gamma_0^t\overline{\mathbf{r}}(\overline{\mathbf{s}}_t,\overline{\mathbf{a}}_t,\mathbf{w})\Big] - \mathbf{w}^{'H}\log\pi(\overline{\mathbf{a}}_t|\overline{\mathbf{s}}_t,\mathbf{w})\Big\}. \tag{A.1}$$

Then, it can be obtained that: $\mathbf{V}'(\overline{\mathbf{s}}_t,\overline{\mathbf{a}}_t,\mathbf{w}) = \mathcal{G}\mathbf{Q}'(\overline{\mathbf{s}}_t,\overline{\mathbf{a}}_t,\mathbf{w})$.

Proof. Firstly, upon taking into account that $\overline{d}(\mathbf{Q}',\mathcal{G}\mathbf{V}') = \sup_{\overline{\mathbf{a}}_t\in\overline{\mathbf{A}},\mathbf{w}\subseteq\mathbf{W}}|\mathbf{w}^H\mathbf{Q}(\overline{\mathbf{s}}_t,\overline{\mathbf{a}}_t,\mathbf{w}) - \mathcal{G}\mathbf{V}(\overline{\mathbf{s}}_t,\overline{\mathbf{a}}_t,\mathbf{w})| = 0$, we are able to observe that $\mathbf{w}^H\mathcal{G}\mathbf{Q}(\overline{\mathbf{s}}_t,\overline{\mathbf{a}}_t,\mathbf{w}) = \mathbf{w}^H\mathbf{V}(\overline{\mathbf{s}}_t,\overline{\mathbf{a}}_t,\mathbf{w})$. The proof is provided as follows:

$$\begin{aligned}
\mathbf{w}^H\mathcal{G}\mathbf{Q}(\overline{\mathbf{s}}_t,\overline{\mathbf{a}}_t,\mathbf{w}) &= \mathbf{w}^H\overline{\mathbf{r}}(\overline{\mathbf{s}}_t,\overline{\mathbf{a}}_t,\mathbf{w}) + \mathbf{w}^H\gamma_0\mathbb{E}_{\overline{\mathbf{s}}_{t+1}\sim\zeta_\pi}[V(\overline{\mathbf{s}}_{t+1},\mathbf{w})],\\
&= \mathbf{w}^H\overline{\mathbf{r}}(\overline{\mathbf{s}}_t,\overline{\mathbf{a}}_t,\mathbf{w}) + \mathbf{w}^H\gamma_0\mathbb{E}_{\overline{\mathbf{s}}_{t+1}\sim\zeta_\pi}\arg_Q\sup_{\overline{\mathbf{a}}_t\in\overline{\mathbf{A}},\mathbf{w}'\subseteq\mathbf{W}}[\mathbf{w}^H\mathbf{Q}(\overline{\mathbf{s}}_t,\overline{\mathbf{a}}_t,\mathbf{w}') - \mathbf{w}^H\log\pi(\overline{\mathbf{a}}_t|\overline{\mathbf{s}}_t,\mathbf{w}')],\\
&= \mathbf{w}^H\overline{\mathbf{r}}(\overline{\mathbf{s}}_t,\overline{\mathbf{a}}_t,\mathbf{w}) + \gamma_0\mathbb{E}_{\overline{\mathbf{s}}_{t+1}\sim\zeta_\pi}\sup_{\overline{\mathbf{a}}_t\in\overline{\mathbf{A}},\mathbf{w}'\subseteq\mathbf{W}}[\mathbf{w}^H\mathbf{Q}(\overline{\mathbf{s}}_t,\overline{\mathbf{a}}_t,\mathbf{w}') - \mathbf{w}^H\log\pi(\overline{\mathbf{a}}_t|\overline{\mathbf{s}}_t,\mathbf{w}')],\\
&= \mathbf{w}^H\overline{\mathbf{r}}(\overline{\mathbf{s}}_t,\overline{\mathbf{a}}_t,\mathbf{w}) + \gamma_0\mathbb{E}_{\overline{\mathbf{s}}_{t+1}\sim\zeta_\pi}\sup_{\overline{\mathbf{a}}_t\in\overline{\mathbf{A}},\mathbf{w}'\subseteq\mathbf{W}}\Big\{\mathbf{w}^H\Big\{\arg_{\mathbf{Q}}\sup_{\pi\in\Pi}\mathbf{w}^{'H}\mathbb{E}_{\overline{\mathbf{s}}_{t+1}\sim\zeta_\pi}\Big[\sum_{t=0}^{T}\gamma_0^t\overline{\mathbf{r}}(\overline{\mathbf{s}}_t,\overline{\mathbf{a}}_t,\mathbf{w})\Big]\\
&\quad - \mathbf{w}^{'H}\log\pi(\overline{\mathbf{a}}_t|\overline{\mathbf{s}}_t,\mathbf{w})\Big\}\Big\},\\
&= \mathbf{w}^H\overline{\mathbf{r}}(\overline{\mathbf{s}}_t,\overline{\mathbf{a}}_t,\mathbf{w}) + \gamma_0\mathbb{E}_{\overline{\mathbf{s}}_{t+1}\sim\zeta_\pi}\sup_{\overline{\mathbf{a}}_t\in\overline{\mathbf{A}}}\Big\{\mathbf{w}^H\Big\{\arg_{\mathbf{Q}}\sup_{\pi\in\Pi}\mathbb{E}_{\overline{\mathbf{s}}_{t+1}\sim\zeta_\pi}\Big[\sum_{t=0}^{T}\gamma_0^t\overline{\mathbf{r}}(\overline{\mathbf{s}}_t,\overline{\mathbf{a}}_t,\mathbf{w})\Big]\Big\}\Big\},\\
&= \mathbf{w}^H\Big\{\arg_{\mathbf{Q}}\sup_{\pi\in\Pi}\mathbf{w}^H\Big\{\mathbb{E}_{\overline{\mathbf{s}}_{t+1}\sim\zeta_\pi}\Big[\sum_{t=0}^{T}\gamma_0^t\overline{\mathbf{r}}(\overline{\mathbf{s}}_t,\overline{\mathbf{a}}_t,\mathbf{w})\Big] - \log\pi(\overline{\mathbf{a}}_t|\overline{\mathbf{s}}_t,\mathbf{w})\Big\}\Big\},\\
&= \mathbf{w}^H\mathbf{V}(\overline{\mathbf{s}}_t,\overline{\mathbf{a}}_t,\mathbf{w}). \tag{A.2}
\end{aligned}$$

Since we have the $\overline{d}(\mathbf{Q}',\mathcal{G}\mathbf{V}') = 0$, the preferred optimal value function is a fixed point of the proposed optimality operator. This concludes the proof of **Theorem 1**. $\qquad\square$

APPENDIX B: PROOF OF THEOREM 2

Theorem 2. *Let us define a pair of MO Q value functions by* $\mathbf{Q}$ *and* $\hat{\mathbf{Q}}$. *Then the Lipschitz condition of* $\overline{d}(\mathcal{G}\mathbf{Q},\mathcal{G}\hat{\mathbf{Q}}) \le \gamma_0\overline{d}(\mathbf{Q},\hat{\mathbf{Q}})$ *can be satisfied, where* γ_0 *is the discount factor.*

Proof. Upon assuming $\sup_{\overline{\mathbf{a}}_t\in\overline{\mathbf{A}},\mathbf{w}'\subseteq\mathbf{W}}\mathbf{w}^H\mathbf{Q}(\overline{\mathbf{s}}_t,\overline{\mathbf{a}}_t,\mathbf{w}') \ge \sup_{\overline{\mathbf{a}}_t\in\overline{\mathbf{A}},\mathbf{w}'\subseteq\mathbf{W}}\mathbf{w}^H\hat{\mathbf{Q}}(\overline{\mathbf{s}}_t,\overline{\mathbf{a}}_t,\mathbf{w}')$, we have

$$\begin{aligned}
\overline{d}(\mathcal{G}\mathbf{Q},\mathcal{G}\hat{\mathbf{Q}}) &= \sup_{\overline{\mathbf{s}}_t\in\overline{\mathbf{S}},\overline{\mathbf{a}}_t\in\overline{\mathbf{A}},\mathbf{w}'\subseteq\mathbf{W}}|\mathbf{w}^H(\mathbf{Q}(\overline{\mathbf{s}}_t,\overline{\mathbf{a}}_t,\mathbf{w}') - \hat{\mathbf{Q}}(\overline{\mathbf{s}}_t,\overline{\mathbf{a}}_t,\mathbf{w}'))|\\
&= \sup_{\overline{\mathbf{s}}_t\in\overline{\mathbf{S}},\overline{\mathbf{a}}_t\in\overline{\mathbf{A}},\mathbf{w}'\subseteq\mathbf{W}}|\gamma_0\mathbf{w}^H\mathbb{E}_{\overline{\mathbf{s}}_{t+1}\sim\zeta_\pi}[\mathbf{Q}(\overline{\mathbf{s}}_t,\overline{\mathbf{a}}_t,\mathbf{w})] - \gamma_0\mathbf{w}^H\mathbb{E}_{\overline{\mathbf{s}}_{t+1}\sim\zeta_\pi}[\hat{\mathbf{Q}}(\overline{\mathbf{s}}_t,\overline{\mathbf{a}}_t,\mathbf{w})]|\\
&\le \gamma_0\sup_{\overline{\mathbf{s}}_t\in\overline{\mathbf{S}},\mathbf{w}'\subseteq\mathbf{W}}|\mathbf{w}^H[\arg_Q\sup_{\overline{\mathbf{a}}_t\in\overline{\mathbf{A}},\mathbf{w}'\subseteq\mathbf{W}}\mathbf{Q}(\overline{\mathbf{s}}_t,\overline{\mathbf{a}}_t,\mathbf{w}') - \arg_Q\sup_{\overline{\mathbf{a}}'_t\in\overline{\mathbf{A}},\mathbf{w}''\subseteq\mathbf{W}}\hat{\mathbf{Q}}(\overline{\mathbf{s}}_t,\overline{\mathbf{a}}_t,\mathbf{w}'')]|,\\
&\le \gamma_0\sup_{\overline{\mathbf{s}}_t\in\overline{\mathbf{S}},\mathbf{w}'\subseteq\mathbf{W}}|\sup_{\overline{\mathbf{a}}_t\in\overline{\mathbf{A}},\mathbf{w}'\subseteq\mathbf{W}}\mathbf{w}^H\mathbf{Q}(\overline{\mathbf{s}}_t,\overline{\mathbf{a}}_t,\mathbf{w}') - \sup_{\overline{\mathbf{a}}'_t\in\overline{\mathbf{A}},\mathbf{w}''\subseteq\mathbf{W}}\mathbf{w}^H\hat{\mathbf{Q}}(\overline{\mathbf{s}}_t,\overline{\mathbf{a}}_t,\mathbf{w}'')|,\\
&= \gamma_0\sup_{\overline{\mathbf{s}}_t\in\overline{\mathbf{S}},\mathbf{w}'\subseteq\mathbf{W}}|\mathbf{w}^H\mathbf{Q}(\overline{\mathbf{s}}_t,\overline{\mathbf{a}}_t,\mathbf{w}') - \mathbf{w}^H\hat{\mathbf{Q}}(\overline{\mathbf{s}}_t,\overline{\mathbf{a}}_t,\mathbf{w}') + \mathbf{w}^H\hat{\mathbf{Q}}(\overline{\mathbf{s}}_t,\overline{\mathbf{a}}_t,\mathbf{w}') - \sup_{\overline{\mathbf{a}}'_t\in\overline{\mathbf{A}},\mathbf{w}''\subseteq\mathbf{W}}\mathbf{w}^H\hat{\mathbf{Q}}(\overline{\mathbf{s}}_t,\overline{\mathbf{a}}_t,\mathbf{w}'')|,\\
&\le \gamma_0\sup_{\overline{\mathbf{s}}_t\in\overline{\mathbf{S}},\mathbf{w}'\subseteq\mathbf{W}}|\mathbf{w}^H\mathbf{Q}(\overline{\mathbf{s}}_t,\overline{\mathbf{a}}_t,\mathbf{w}') - \mathbf{w}^H\hat{\mathbf{Q}}(\overline{\mathbf{s}}_t,\overline{\mathbf{a}}_t,\mathbf{w}')|,\\
&\le \gamma_0\sup_{\overline{\mathbf{s}}_t\in\overline{\mathbf{S}},\mathbf{a}_t\in\overline{\mathbf{A}},\mathbf{w}'\subseteq\mathbf{W}}|\mathbf{w}^H\mathbf{Q}(\overline{\mathbf{s}}_t,\overline{\mathbf{a}}_t,\mathbf{w}') - \mathbf{w}^H\hat{\mathbf{Q}}(\overline{\mathbf{s}}_t,\overline{\mathbf{a}}_t,\mathbf{w}')|,\\
&= \gamma_0\overline{d}(\mathbf{Q},\hat{\mathbf{Q}}). \tag{B.1}
\end{aligned}$$

Then, we proved **Theorem 2**. $\qquad\square$

APPENDIX C: PROOF OF THEOREM 3

Theorem 3. *If* $\mathcal{G}$ *can be contracted with the aid of the discount factor* γ_0 *on the complete pseudo-metric space* $\langle\mathcal{Q},\overline{d}\rangle$, *it can be shown that* $\lim_{t\to\infty}\overline{d}(\mathcal{G}^t,\mathbf{Q},\hat{\mathbf{Q}}) = 0$, *where* $\mathcal{Q}$ *is the value space.*

Proof. Similarly to **Theorem 2**, we define a pair of mo Q value functions by $\mathbf{Q}$ and $\hat{\mathbf{Q}}$. we have:

$$\overline{d}(\mathbf{Q},\hat{\mathbf{Q}}) \le \overline{d}(\mathbf{Q},\mathcal{G}\mathbf{Q}) + \overline{d}(\mathcal{G}\mathbf{Q},\mathcal{G}\hat{\mathbf{Q}}) + \overline{d}(\mathcal{G}\hat{\mathbf{Q}},\hat{\mathbf{Q}}) \le \overline{d}(\mathbf{Q},\mathcal{G}\mathbf{Q}) + \gamma_0\overline{d}(\mathbf{Q},\hat{\mathbf{Q}}) + \overline{d}(\mathcal{G}\hat{\mathbf{Q}},\hat{\mathbf{Q}}), \tag{C.1}$$

and it can be shown that:

$$\overline{d}(\mathbf{Q},\hat{\mathbf{Q}}) \le \frac{\overline{d}(\mathbf{Q},\mathcal{G}\mathbf{Q}) + \overline{d}(\hat{\mathbf{Q}},\mathcal{G}\hat{\mathbf{Q}})}{1-\gamma_0}. \tag{C.2}$$

Upon considering the pair of values $\mathcal{G}^{t_1}\mathbf{Q}$ and $\mathcal{G}^{t_2}\mathbf{Q}$ in $\{\mathcal{G}^t\mathbf{Q}\}$, the following upper bound can be formulated:

$$\overline{d}(\mathcal{G}^{t_1}\mathbf{Q},\mathcal{G}^{t_2}\mathbf{Q}) \le \frac{\overline{d}(\mathcal{G}^{t_1}\mathbf{Q},\mathcal{G}^{t_1+1}\mathbf{Q}) + \overline{d}(\mathcal{G}^{t_2}\hat{\mathbf{Q}},\mathcal{G}^{t_2+1}\hat{\mathbf{Q}})}{(1-\gamma_0)} \le \frac{\gamma_{t_1}\overline{d}(\mathbf{Q},\mathcal{G}\mathbf{Q}) + \gamma_{t_2}\overline{d}(\mathbf{Q},\mathcal{G}\mathbf{Q})}{1-\gamma_0} \le \frac{\gamma_{t_1}+\gamma_{t_2}}{1-\gamma_0}\overline{d}(\mathbf{Q},\mathcal{G}\mathbf{Q}). \tag{C.3}$$

Since $\gamma_0\in[0,1)$, for any two values of $\mathcal{G}^{t_1}\mathbf{Q}$, $\mathcal{G}^{t_2}\mathbf{Q}$, we can have:

$$\begin{aligned}
\overline{d}(\mathcal{G}^{t_1}\mathbf{Q},\mathbf{Q}) &= \lim_{t_1\leftarrow\infty}\overline{d}(\mathcal{G}^{t_1+1}\mathbf{Q},\mathbf{Q}) = \lim_{t_1\leftarrow\infty}\overline{d}(\mathcal{G}^{t_1}\mathbf{Q},\mathbf{Q}) = 0,\\
\overline{d}(\mathcal{G}^{t_2}\mathbf{Q},\mathbf{Q}) &= \lim_{t_2\leftarrow\infty}\overline{d}(\mathcal{G}^{t_2+1}\mathbf{Q},\mathbf{Q}) = \lim_{t_2\leftarrow\infty}\overline{d}(\mathcal{G}^{t_2}\mathbf{Q},\mathbf{Q}) = 0. \tag{C.4}
\end{aligned}$$

Hence, we proved **Theorem 3**. $\qquad\square$

REFERENCES

[1] X. Gao, Y. Zou, W. Yi, J. X, R. Liu, and Y. Liu, "Multi-objective Optimization of Energy and Latency in URLLC-enabled Wireless VR Networks", *Proc. 18th Int. Symp. Wirel. Communi. Syst. (ISWCS)*, 2022, accept to appear. (Invited paper)

[2] J. Joshua, "Information bodies: Computational anxiety in Neal Stephenson's snow crash," *Interdiscip. Lit. Stud.*, vol. 19, no. 1, pp. 17–47, Jan. 2017.

[3] K. Yoneda, N. Hashimoto, R. Yanase, "Vehicle Localization using 76GHz Omnidirectional Millimeter-Wave Radar for Winter Automated Driving," *Proc. IEEE Intell. Veh. Symp.* (IV), Changshu, China, 26–30 June 2018, pp. 971–977.

[4] X. Gao et al., "Review of Wheeled Mobile Robots' Navigation Problems and Application Prospects in Agriculture," *IEEE Access*, vol. 6, pp. 49248-49268, 2018.

[5] H. Zhang, H. Zhang, W. Liu, K. Long, J. Dong, and V. C. M. Leung, "Energy efficient user clustering, hybrid precoding and power optimization in TeraHertz MIMO-NOMA systems," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 9, pp. 2074–2085, Sept. 2020.

[6] C. Pan *et al.*, "Reconfigurable Intelligent Surfaces for 6G Systems: Principles, Applications, and Research Directions," *IEEE Communi. Mag.*, vol. 59, no. 6, pp. 14-20, June 2021.

[7] C. Zhang, W. Yi, Y. Liu, K. Yang and Z. Ding, "Reconfigurable Intelligent Surfaces Aided Multi-Cell NOMA Networks: A Stochastic Geometry Model," *IEEE Trans. Communi.*, vol. 70, no. 2, pp. 951-966, Feb. 2022.

[8] Y. Liu *et al.*, "Reconfigurable Intelligent Surfaces: Principles and Opportunities," *IEEE Commun. Surv. Tutor.*, vol. 23, no. 3, pp. 1546-1577, thirdquarter 2021.

[9] C. She, C. Yang and T. Q. S. Quek, "Cross-Layer Optimization for Ultra-Reliable and Low-Latency Radio Access Networks," *IEEE Trans. Wirel. Communi.*, vol. 17, no. 1, pp. 127-141, Jan. 2018.

[10] C. She, C. Yang and T. Q. S. Quek, "Radio Resource Management for Ultra-Reliable and Low-Latency Communications," *IEEE Communi. Mag.*, vol. 55, no. 6, pp. 72-78, June 2017.

[11] D. Van Huynh, S. R. Khosravirad, A. Masaracchia, O. A. Dobre and T. Q. Duong, "Edge Intelligence-Based Ultra-Reliable and Low-Latency Communications for Digital Twin-Enabled Metaverse," *IEEE Wirel. Communi. Lett.*, vol. 11, no. 8, pp. 1733-1737, Aug. 2022.

[12] G. Minopoulos and K. E. Psannis, "Opportunities and Challenges of Tangible XR Applications for 5G Networks and Beyond," *IEEE Consum. Electron. Mag.*, doi: 10.1109/MCE.2022.3156305.

[13] B. Kenwright, "The Future of Extended Reality (XR)," *Communi. Article*, January, 2020.

[14] Z. Bojkovic, D. Milovanovic, T.P. Fowdur, and M. Indoonundon, "6G ultra-low latency communication in future mobile XR applications," *Proc. Int. Symp. Signal Process. Intell. Recognit. Sys.*, Springer, Singapore, pp. 302-312, October, 2020.

[15] Y. Lu, H. Zheng, S. Chand, W. Xia, Z. Liu, X. Xu, L. Wang, Z. Qin, and J. Bao, "Outlook on human-centric manufacturing towards industry 5.0," *J. Manuf. Syst.*, vol. 62, pp. 612–627, Jan. 2022.

[16] H. Du et al., "Attention-aware Resource Allocation and QoE Analysis for Metaverse xURLLC Services," *IEEE J. Sel. Areas Commun.*, doi: 10.1109/JSAC.2023.3280978.

[17] F. Tang, X. Chen, M. Zhao and N. Kato, "The Roadmap of Communication and Networking in 6G for the Metaverse," *IEEE Wirel. Communi.*, doi: 10.1109/MWC.019.2100721.

[18] M. Chen, W. Saad and C. Yin, "Virtual Reality Over Wireless Networks: Quality-of-Service Model and Learning-Based Resource Management," *IEEE Trans. Communi.*, vol. 66, no. 11, pp. 5621-5635, Nov. 2018.

[19] H. Wang, Z. Lei, Y. Liu, J. Peng, and J. Liu, "Echo state network based ensemble approach for wind power forecasting," *Energy convers. manag.*, 201, p.112188, 2019.

[20] A. Taha, Q. Qu, S. Alex, P. Wang, W. L. Abbott and A. Alkhateeb, "Millimeter Wave MIMO-Based Depth Maps for Wireless Virtual and Augmented Reality," *IEEE Access*, vol. 9, pp. 48341-48363, 2021.

[21] F. Theilmann, and S. Grusche, "An RGB approach to prismatic colours," *Phys. Educ.*, vol. 48, no. 6, p. 750, 2013.

[22] J. M. Batalla, "On Analyzing Video Transmission Over Wireless WiFi and 5G C-Band in Harsh IIoT Environments," *IEEE Access*, vol. 8, pp. 118534-118541, 2020.

[23] S. Dou *et al.*, "XR Quality Index: Evaluating RAN Transmission Quality for XR Services over 5G and Beyond," *Proc. of IEEE 32nd Annu. Int. Symp. Pers., Indoor and Mob. Radio Communi.* (PIMRC), 2021, pp. 1-6.

[24] H. Du, J. Wang, D. Niyato, J. Kang, Z. Xiong, and D. I. Kim, "Exploring attention-aware network resource alposition for customized Metaverse services," *arXiv preprint arXiv:2208.00369*, 2022.

[25] Z. Meng, C. She, G. Zhao, and D. De Martini, "Sampling, Communication, and Prediction Co-Design for Synchronizing the Real-World Device and Digital Model in Metaverse," *arXiv preprint arXiv:2208.04233*, 2022.

[26] W. C. Ng, W. Yang Bryan Lim, J. S. Ng, Z. Xiong, D. Niyato and C. Miao, "Unified Resource Alposition Framework for the Edge Intelligence-Enabled Metaverse," *Proc. of IEEE Int. Conf. Communi.* (ICC), 2022, pp. 5214-5219.

[27] H. Xie, J. Xu, Y. -F. Liu, L. Liu and D. W. K. Ng, "User Grouping and Reflective Beamforming for IRS-Aided URLLC," *IEEE Wirel. Communi. Lett.*, vol. 10, no. 11, pp. 2533-2537, Nov. 2021.

[28] W. R. Ghanem, V. Jamali and R. Schober, "Joint Beamforming and Phase Shift Optimization for Multicell IRS-aided OFDMA-URLLC Systems," *Proc. IEEE Wirel. Communi. Netw. Conf.* (WCNC), 2021, pp. 1-7.

[29] R. Hashemi, S. Ali, N. H. Mahmood and M. Latva-Aho, "Joint Sum Rate and Blocklength Optimization in RIS-Aided Short Packet URLLC Systems," *IEEE Communi. Lett.*, vol. 26, no. 8, pp. 1838-1842, Aug. 2022.

[30] E. E. Haber, M. Elhattab, C. Assi, S. Sharafeddine and K. K. Nguyen, "Latency and Reliability Aware Edge Computation Offloading in IRS-aided Networks," *Proc. of IEEE Int. Conf. Communi.* (ICC), 2022, pp. 5035-5040.

[31] J. Zhang, X. Xu, K. Zhang, B. Zhang, X. Tao and P. Zhang, "Machine Learning Based Flexible Transmission Time Interval Scheduling for eMBB and uRLLC Coexistence Scenario," *IEEE Access*, vol. 7, pp. 65811-65820, 2019.

[32] C. Chaccour, M. N. Soorki, W. Saad, M. Bennis and P. Popovski, "Can TeraHertz Provide High-Rate Reliable Low-Latency Communications for Wireless VR?," *IEEE Internet Things J.*, vol. 9, no. 12, pp. 9712-9729, 15 June15, 2022.

[33] H. Ren, K. Wang and C. Pan, "Intelligent Reflecting Surface-Aided URLLC in a Factory Automation Scenario," *IEEE Trans. Communi.*, vol. 70, no. 1, pp. 707-723, Jan. 2022.

[34] S. Dhok, P. Raut, P. K. Sharma, K. Singh and C. -P. Li, "Non-Linear Energy Harvesting in RIS-Assisted URLLC Networks for Industry Automation," *IEEE Trans. on Commun.*, vol. 69, no. 11, pp. 7761-7774, Nov. 2021.

[35] A. Ranjha and G. Kaddoum, "URLLC-Enabled by Laser Powered UAV Relay: A Quasi-Optimal Design of Resource Alposition, Trajectory Planning and Energy Harvesting," *IEEE Trans. Veh. Technol.*, vol. 71, no. 1, pp. 753-765, Jan. 2022.

[36] E. Bjornson and L. Sanguinetti, "Rayleigh fading modeling and channel hardening for reconfigurable intelligent surfaces," *IEEE Wirel. Commun. Lett.*, vol. 10, no. 4, pp. 830-834, Apr. 2021.

[37] J. He, F. Jiang, K. Keykhosravi, J. Kokkoniemi, H. Wymeersch and M. Juntti, "Beyond 5G RIS mmWave Systems: Where Communication and Localization Meet," *IEEE Access*, vol. 10, pp. 68075-68084, 2022.

[38] C. Lin and G. Y. Li, "Adaptive Beamforming with Resource Alposition for Distance-Aware Multi-user Indoor TeraHertz Communications," *IEEE Trans. on Communi.*, vol. 47, no. 1, pp. 2985–2995, Aug. 2015.

[39] A. -A. A. Boulogeorgos and A. Alexiou, "Pathloss Modeling of Reconfigurable Intelligent Surface Assisted THz Wireless Systems," *Proc. of IEEE Int. Conf. Communi.* (ICC), 2021, pp. 1-6.

[40] Y. Pan, K. Wang, C. Pan, H. Zhu and J. Wang, "Sum-Rate Maximization for Intelligent Reflecting Surface Assisted TeraHertz Communications," *IEEE Trans. Veh. Technol.*, vol. 71, no. 3, pp. 3320-3325, March 2022.

[41] V. Petrov, D. Moltchanov, and Y. Koucheryavy, "Interference and SINR in Dense TeraHertz Networks," *Proc. of IEEE Veh. Tech. Conf.* (VTC), Boston, MA, Sept. 2015, pp. 1–5.

[42] W. Liu, G. Nair, Y. Li, D. Nesic, B. Vucetic and H. V. Poor, "On the Latency, Rate, and Reliability Tradeoff in Wireless Networked Control Systems for IIoT," *IEEE Internet Things J.*, vol. 8, no. 2, pp. 723-733, 15 Jan.15, 2021.

[43] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic Meta-learning for Fast Adaptation of Deep Networks," *arXiv Preprint arXiv:1703.03400*, 2017.

[44] P. Stoica and A. Nehorai, "MUSIC, maximum likelihood, and Cramer-Rao bound," *IEEE Trans. Signal Process.*, vol. 37, no. 5, pp. 720-741, May 1989.

[45] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Prentice-Hall, Inc., 1993.

[46] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor," *Proc. of Int. conf. mach. learn.* (ICML), vol. 80, pp. 1861–1870, 2018.

[47] I. Jais, A. Ismail, and S. Nisa, "Adam Optimization Algorithm for Wide and Deep Neural Network," *Knowl. Eng. Data Sci.*, vol. 2, no. 1, pp. 41-46, 2019.

[48] S. Ruder, "An overview of gradient descent optimization algorithms," *arXiv preprint arXiv:1609.04747*, 2016.

[49] P. Goyal *et al.* "Accurate, Large Minibatch SGD: Training ImageNet in 1 Hour," *arXiv preprint arXiv:1706.02677*, 2017.