Movable Antenna-Enhanced Wireless Powered Mobile Edge Computing Systems

Pengcheng Chen, *Student Member, IEEE*, Yuxuan Yang, *Student Member, IEEE*, Bin Lyu, *Member, IEEE*,
Zhen Yang, *Senior Member, IEEE*, and Abbas Jamalipour, *Fellow, IEEE*

Abstract—In this paper, we propose a movable antenna (MA) enhanced scheme for wireless powered mobile edge computing (WP-MEC) system, where the hybrid access point (HAP) equipped with multiple MAs first emits wireless energy to charge wireless devices (WDs), and then receives the offloaded tasks from the WDs for edge computing. The MAs deployed at the HAP enhance the spatial degrees of freedom (DoFs) by flexibly adjusting the positions of MAs within an available region, thereby improving the efficiency of both downlink wireless energy transfer (WPT) and uplink task offloading. To balance the performance enhancement against the implementation intricacy, we further propose three types of MA positioning configurations, i.e., dynamic MA positioning, semi-dynamic MA positioning, and static MA positioning. In addition, the non-linear power conversion of energy harvesting (EH) circuits at the WDs and the finite computing capability at the edge server are taken into account. Our objective is to maximize the sum computational rate (SCR) by jointly optimizing the time allocation, positions of MAs, energy beamforming matrix, receive combing vectors, and offloading strategies of WDs. To solve the non-convex problems, efficient alternating optimization (AO) frameworks are proposed. Moreover, we propose a hybrid algorithm of particle swarm optimization with variable local search (PSO-VLS) to solve the sub-problem of MA positioning. Numerical results validate the superiority of exploiting MAs over the fixed-position antennas (FPAs) for enhancing the SCR performance of WP-MEC systems.

Index Terms—Movable antennas (MAs), wireless powered mobile edge computing (WP-MEC), MA positioning configurations, particle swarm optimization with variable local search (PSO-VLS).

I. INTRODUCTION

With the rapid development of Internet of Things (IoT), the scale of data produced by heterogeneous wireless devices (WDs) has grown exponentially in recent years [1]. However, constrained by the limitations of physical dimensions, the batteries of WDs are normally capacity-constrained, resulting in insufferable replacement cost. In addition, for cost saving, the WDs are typically equipped with processors with poor computing capability. As such, these WDs can not sustainably power the emerging resource-intensive applications with strict delay demands [2]. Cloud computing, allowing task offloading from WDs to the powerful cloud servers, can significantly

P. Chen, B. Lyu, and Z. Yang are with the Key Laboratory of Ministry of Education in Broadband Wireless Communication and Sensor Network Technology, Nanjing University of Posts and Telecommunications, Nanjing 210003, China (email: 2020010104@njupt.edu.cn, blyu@njupt.edu.cn, yangz@njupt.edu.cn).

Y. Yang and A. Jamalipour are with the School of Electrical and Information Engineering, University of Sydney, Sydney, NSW 2006, Australia (email:yyan5256@uni.sydney.edu.au, a.jamalipour@ieee.org).

amplify the computing capability of WDs. However, these servers are always deployed in internet center and far away from the WDs, which causes severe energy loss and unbearable response delay.

Mobile edge computing (MEC) has emerged as an effective solution for enhancing the capabilities of WDs by placing servers in close proximity to the WDs [3]. Compared to the cloud computing, MEC facilitates more efficient access for WDs to the computation resources of the edge servers over the centralized cloud in the distance [4]. Wireless power transfer (WPT) technique has gained increasing attention for its ability of handling the critical issue of energy shortage at the WDs [5]. Specifically, the WDs featuring energy harvesting (EH) circuits can harvest the energy of radio frequency (RF) signals from a controllable energy station and then store it in rechargeable batteries [6]. In this way, the batteries of the WDs keep on recharging, which makes the self-sustainability of WDs realistic. The combination of WPT and MEC, i.e., wireless powered MEC (WP-MEC), has pioneered an appealing paradigm for ubiquitous computing in a self-sustainable manner [7]. In WP-MEC systems, WDs leverage the harvested energy for local computing and/or task offloading. The proliferation of resource-intensive applications has necessitated the utilization of multiple antennas at the energy station to achieve efficient WPT systems by virtue of favorable energy beams. However, the spatial degrees of freedom (DoFs) are not fully exploited since the positions of the conventional antennas remain stationary.

Recently, an emerging antenna paradigm called movable antenna (MA) has originated from the above dilemma, which is able to fully exploit such DoFs in the continuous spatial region [8]. Specifically, in MA-enhanced systems, each antenna is linked with an RF chain through a flexible cable, whose length can be flexibly adjusted in a certain range. In addition, each antenna is equipped with a driver component, e.g., stepper motor or servo, and thus it can move within an available spatial region [9]–[11]. Compared to the fixed-position antennas (FPAs), the flexible mobility of MAs facilitates the antennas at favorable positions for better channel conditions so as to realize efficient WPT and communications. Generally, fewer antennas are acquired by the MA-enhanced systems to utilize such spatial DoFs [12], which substantially reduces the computational complexity of signal processing in comparison with the FPA-enabled systems.

A. Related Works

We review the existing works from next three perspectives, i.e., resource allocation in MEC systems, designs of WP-MEC systems, and MA-enhanced communications, respectively.

1) Resource allocation in MEC systems: The objective of MEC is to merge wireless communications and mobile computing seamlessly, yielding diverse designs of offloading strategies and network architectures. Considering the load imbalance at the edge sides caused by the differences in tasks and network capability, the authors in [13] designed an efficient task offloading architecture to realize flexible collaborations between multiple edge servers. The authors in [14] developed an efficient pricing mechanisms for resource allocation based on auction and linear programming. The authors in [15] studied the impacts of task priority of heterogeneous WDs in the unmanned aerial vehicle (UAV) mounted MEC systems, where the offloading strategies and UAV trajectory are jointly optimized according to the time-varying priorities of tasks. The authors in [16] studied the placements of services in MEC systems to reduce the response latency. An edge platform was constructed in [17], which allowed flexible resources allocation for WDs with diverse requirements.

2) Designs of WP-MEC systems: To facilitate ubiquitous computing sustainability, WPT-enabled MEC, has attracted increasing attention. The authors in [18] studied a weighted SCR maximization problem in the WP-MEC system, where the WDs follow the binary offloading mode. The authors in [19] studied the resource management for IRS-assisted WP-MEC system, where the energy consumption of the IRS and WDs was considered. To mitigate the doubly near-far effects, the authors in [20] proposed a cooperative scheme, in which the far-WD with less harvested energy can offload its tasks with the aid of the near-WD with better channel condition. The authors in [21] utilized an UAV to provide wireless energy and receive offloaded tasks in the vicinity of WDs. The author in [22] focused on the security of task offloading and leveraged the power station to emit jamming signals to interfere with the eavesdroppers. The authors in [23] and [24] studied backscatter-aided WP-MEC systems, in which the WDs can simultaneously obtain computing services from the edge server and harvest energy. The authors in [25] also studied a backsctter-aided WP-MEC system and maximized the weighted SCR of all the WDs. The authors in [26] studied the problem of max-min energy fairness among the multiple WDs. In addition, the authors in and [27]–[29] maximized the computational rate in the WP-MEC systems, where the time spent on edge computing is considered.

3) MA-enhanced communications: MA has emerged as a promising approach to achieving better communication performance by exploiting new spatial DoFs via flexibly adjusting the positions of MAs. The authors in [9] developed a field-response based channel model under the far-field conditions and analyzed the maximum channel gain of the MA-aided system, in which both the receiver and transmitter are equipped with a single MA. For further capacity improvement, the authors in [10] studied an MA-aided point-to-point multiple-input multiple-output (MIMO) system, and jointly optimized the positions of MAs at both receiver and transmitter. The authors in [11] and [12] studied the MA-enabled multiuser communication networks and maximized the sum-rate of all the WDs. To reduce the pilot overhead, the authors in [30] and [31] proposed a compressed sensing based channel estimation framework, by which the complete channel state information (CSI) can be reconstructed via limited number of channel measurements. The authors in [32] maximized the achievable rate by jointly optimizing the MA positions and the transmit covariance matrix based on statistical CSI. The authors in [33] maximized the minimum beamforming gain over the desired directions by joint optimizing the positions and weight vector of the MA array. The effectiveness of the directional gain of the linear MA array was confirmed in [34], i.e., it enhances the gain in the target direction and suppresses interference in non-target directions. Moreover, the authors in [35] maximized the minimum capability among all the WDs in an MA-aided multi-user system, and further validated the superiority of MAs in channel gain enhancement and interference suppression over the FPAs. The authors in [36]–[38] demonstrated that MAs have significant power saving properties. Moreover, the benefits of MAs in physical security are exploited and conformed by the authors in [39].

B. Motivations and Contributions

In the previous studies (i.e., [18]–[29]), the transmitters and receivers in WP-MEC systems were equipped with the conventional FPAs, which cannot fully exploit spatial diversity gains. MA technology, flexibly deploying the antennas at the positions with better channel conditions, has been regarded as a novel approach to acquiring the spatial DoFs. Therefore, it is necessary to exploit the advantages of MA technology in the WP-MEC systems for further performance improvement.

Recently, several works have validated the advantages of MA in capacity improvement [9]–[12], interference suppression [34], [35], power conservation [36]–[38], and physical security [39] in communications. However, the methods proposed in [9]–[12] and [34]–[39] are not suitable for the WP-MEC systems. It is due to the fact that the MA technology has not been used for the WPT improvement to the best of our knowledge. Hence, it is necessary to work out a new method with high accuracy to optimizing the positions of MAs in the WP-MEC systems.

In addition, the authors in [18]–[28] assumed that the edge server executes the offloaded tasks after the task offloading stage. As a result, the edge server keeps idle for a long time during each time block. Moreover, the previous works (i.e., [18]–[23], [25] and [26]) ideally assumed that the computing capability of edge server is infinite and ignore the time spent on executing the offloaded tasks.

Based on above observations, we propose an MA-enhanced WP-MEC system, where the HAP is equipped with MA array instead of the conventional FPAs. The MA-enabled HAP emits wireless energy to charge the WDs firstly and then receives the offloaded tasks from the WDs. The WPT efficiency can be improved and the communication links for task offloading can be strengthened by properly adjusting both the beamforming designs and the positions of MA array at the HAP.Due to the

non-linear conversion efficiency of the built-in rectifier, the EH circuit typically causes a non-linear power conversion [40]. To capture the non-linear power conversion of EH circuits at the WDs, a practical EH model is applied. In addition, the finite computing capability of edge server is considered. According to the characteristics of the partial offloading mode (i.e., arbitrary segmentation and parallel computation [4]), we consider the immediate response at the edge server, i.e., the edge server immediately executes the offloaded tasks as soon as they are received. In this way, the finite edge computing capability can be fully exploited. Our objective is to maximize the sum computational rate (SCR) of all WDs by jointly optimizing the time allocation, the positions of MAs, the energy beamforming matrix and receive combing vectors at the HAP, as well as the offloading strategies of the WDs. The main contributions are summarized as follows.

Fig. 1. The MA-enhanced WP-MEC system.

Fig. 2. The time scheduling.

- To the best of our knowledge, this is the first work that exploits MAs in the WP-MEC systems. To balance the enhancement of performance against the augmentation in implementation intricacy due to the MA positioning optimization, three types of MA positioning configurations, i.e., dynamic MA positioning, semi-dynamic MA positioning, and static MA positioning, are proposed based on how the MAs are allowed to adjust their positions across time. Specifically, for the dynamic MA positioning scheme, the positions of MAs can be adjusted flexibly not only for WPT but also for each WD's task offloading, thereby improving the WPT efficiency and strengthening the communication links significantly. To achieve a better trade-off between the performance enhancement and the implementation intricacy, the semi-dynamic MA positioning scheme allows all WDs to share the same positions of MAs during the entire task offloading stage. In addition, the implementation intricacy can be further reduced by the static MA positioning scheme, in which the positions of MAs remain stationary throughout the entire time block.
- For the case with dynamic MA positioning, we propose an efficient alternating optimization (AO) framework to solve the formulated problem. Specifically, the feasibility-checking sub-problem of energy beamforming matrix and MA positioning optimization during the WPT stage is firstly converted into an explicit goal-based problem. Then, the successive convex approximation (SCA) is applied to tackle the non-convexity caused by the non-linear EH model. In addition, according to the maximum-ratio combining (MRC) criterion, the receive combing vectors are calculated in the closed-form with respect to the positions of MAs. The hybrid algorithm of particle swarm optimization with variable local search (PSO-VLS) is proposed to optimize the positions of MAs. Compared to the standard PSO algorithm in [35], the PSO-VLS algorithm can achieve a better SCR performance.
- For the case with semi-dynamic MA positioning and the case with static MA positioning, we extend the AO framework with the PSO-VLS algorithm to solve the formulated problems. Specifically, the MA positioning
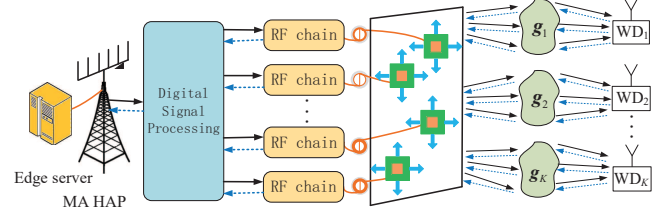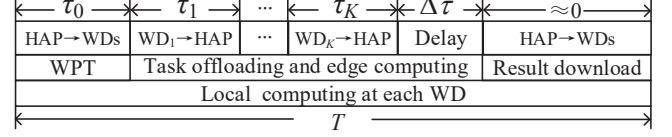
optimizations for both cases are solved by the PSO-VLS algorithm by properly redesigning the fitness functions.
- Numerical results validate the high efficiency of the PSO-VLS algorithm and the AO frameworks. Compared to the standard PSO algorithm, the proposed PSO-VLS algorithm provides solutions for MA positioning with higher accuracy, especially for the scenario with sufficient channel paths. Moreover, extensive results demonstrate the superiority of the MA-enhanced scheme over the FPA-enabled scheme in WP-MEC systems.

The rest of this paper is organized as follows. Section II introduces the system model and problem formulations for the MA-enhanced WP-MEC system with three types of MA positioning configurations. Section III proposes an efficient AO framework to solve the formulated problem for the case with dynamic MA positioning. In Section IV, the AO framework is extended to solve the formulated problems for the cases with semi-dynamic MA positioning and static MA positioning, respectively. Section V provides the numerical results. Section VI concludes this paper.

II. SYSTEM MODEL AND PROBLEM FORMULATION

As depicted in Fig. 1, we consider an MA-enhanced WP-MEC system, which comprises an HAP with M MAs, an edge server with finite computing capability, and K single-FPA WDs. Each of the MAs at HAP is connected to an RF chain via a flexible cable and is driven by a stepper motor [9]. Thus, the positions of MAs can be mechanically adjusted within a given region at the HAP, i.e., $\mathcal{C}$. The HAP firstly emits wireless energy to charge the WDs and then receives the offloaded tasks from the WDs. The WDs are assumed to operate in the partial offloading mode, i.e., offloading tasks while processing computational tasks locally. To guarantee completely execution of the offloaded tasks at the edge server, the processing delay caused by the finite computing capability of edge server needs to be taken into account.

Fig. 2 illustrates the scheduling of the considered time block with duration of T in seconds, which consists of the WPT stage, the task offloading and edge computing stage, as well as the result downloading stage. The WPT stage with

duration of τ_0 is used for charging the WDs. Then, during the task offloading and edge computing stage, the WDs may opt for employing part of the harvested energy to offload some tasks to the edge server in rotation. Specifically, the slot with duration of τ_k is exclusively used for WD k to offload its tasks, where $k \in \mathcal{K} \triangleq \{1, \cdots, K\}$. To fully exploit the finite edge computing capability, we consider the immediate response at the MEC server, i.e., the MEC server immediately processes any offloaded tasks from WDs once they are received, instead of implementing the edge computing after completing the offloading of all tasks [24]. In addition, a dedicated slot with duration of $\Delta\tau$ is required to guarantee the complete execution of residual tasks. Similar to [41], the duration for result downloading can be negligible due to the minuscule volume of results typically involved. Advanced power splitting techniques empower the on-chip processors of WDs during the WPT stage by utilizing portion of the harvested energy, facilitating continuous local computing throughout the whole time block [42].

In this paper, three types of MA positioning configurations, i.e., dynamic MA positioning, semi-dynamic MA positioning, and static MA positioning, are considered. Without loss of generality, the case with dynamic MA positioning is modeled in details. As mentioned above, the positions of MAs can be flexibly adjusted during each time slot in this case.

A. Channel Model

Similar to [9], [10], [34], [35], The field-response channel model is employed, i.e, the channel response is the super-position of coefficients of multiple channel paths between the HAP and WDs. In addition, for this MA-enhanced WP-MEC system, the far-field condition holds due to the smaller moving area for MAs compared with signal transmission range between the WDs and the HAP. Thus, the plane-wave model can be utilized to form the field response channel [9]. Specifically, the angle of departure (AoD), the angle of arrival (AoA), and the amplitude of complex coefficient for each channel path remains constant despite various positions of MAs, while only the phases of the channel paths various with respect to the positions of MAs.

Let Cartesian coordinates $\boldsymbol{r}_{i,m} = [x_{i,m}, y_{i,m}]$ denote the position of the m-th MA during τ_i, where $i \in \mathcal{I} \triangleq \{0, 1, \cdots, K\}$ and $m \in \mathcal{M} \triangleq \{1, \cdots, M\}$. Let L_k denote the number of channel paths between the HAP and WD k. The elevation and azimuth AoDs of the l-th channel path at the HAP for WD k are respectively denoted by $\theta_{k,l} \in [0, \pi]$ and $\phi_{k,l} \in [0, \pi]$. Then, for this channel path, the deference of signal propagation between the position $\boldsymbol{r}_{i,m}$ and the reference point $\boldsymbol{r}_0 = [0, 0]$ is

$$\rho_{k,l}(\boldsymbol{r}_{i,m}) = x_{i,m}\sin(\theta_{k,l})\cos(\phi_{k,l}) + y_{i,m}\cos(\theta_{k,l}). \quad (1)$$

Accordingly, during τ_i, the field response vector (FRV) of between the m-th MA and WD k is obtained as

$$\boldsymbol{f}_k(\boldsymbol{r}_{i,m}) = \left[e^{j\frac{2\pi}{\lambda}\rho_{k,1}(\boldsymbol{r}_{i,m})}, \cdots, e^{j\frac{2\pi}{\lambda}\rho_{k,L_k}(\boldsymbol{r}_{i,m})}\right]^T \in \mathbb{C}^{L_k \times 1},$$

where λ is the carrier wavelength. By stacking $\boldsymbol{f}_k(\boldsymbol{r}_{i,m})$ of all MAs, the field response matrix (FRM) from WD k and the m-th MA during τ_i is

$$\mathbf{F}_{i,k}(\widetilde{\boldsymbol{r}}_i) \triangleq [\boldsymbol{f}_k(\boldsymbol{r}_{i,1}), \cdots, \boldsymbol{f}_k(\boldsymbol{r}_{i,M})] \in \mathbb{C}^{L_k \times M}, \quad (2)$$

where $\widetilde{\boldsymbol{r}}_i \triangleq [\boldsymbol{r}_{i,1}, \cdots, \boldsymbol{r}_{i,M}]$ denote the antenna position vector (APV) of all MAs during τ_i. As such, the channel response vector (CRV) from WD k to the HAP is

$$\boldsymbol{h}_k(\widetilde{\boldsymbol{r}}_i) = \mathbf{F}_{i,k}^H(\widetilde{\boldsymbol{r}}_i)\boldsymbol{g}_k \in \mathbb{C}^{M \times 1}, \quad (3)$$

where $\boldsymbol{g}_k = [g_{k,1}, \cdots, g_{k,l}, \cdots, g_{k,L_k}]^T \in \mathbb{C}^{L_k \times 1}$ is the path response vector (PRV). Herein, $g_{k,l}$ is assumed to be block-fading and perfectly estimated by the methods proposed in [30] and [31].

The channel reciprocity between the downlink channels and the counterpart uplink channels is assumed and thus the CRV from the HAP and WD k during τ_i is $\boldsymbol{h}_k^T(\widetilde{\boldsymbol{r}}_i)$ [43].

B. Wireless Power Transfer

Let $\boldsymbol{x}_0 = \sum_{j=1}^{M_t} \boldsymbol{w}_{0,j}x_{0,j}$ denote the combined effect of multiple independent energy beams emitted by the HAP, where $M_t \leq M$ is the number of energy beams, and $x_{0,j}$ is the carried signal satisfying $\mathbb{E}[\|x_{0,j}\|^2] = 1$. Accordingly, during τ_0, the signal received by WD k is

$$y_{0,k} = \boldsymbol{h}_k^T(\widetilde{\boldsymbol{r}}_0)\boldsymbol{x}_0 + n_k, \quad (4)$$

where $n_k \sim \mathcal{CN}(0, \sigma_0^2)$ denotes the additive white Gaussian noise (AWGN). For simplicity, let $\mathbf{Q} \triangleq \mathbb{E}\left[\|\boldsymbol{x}_0\|^2\right] = \sum_{j=1}^{M_t} \boldsymbol{w}_{0,j}\boldsymbol{w}_{0,j}^H \succeq 0$ denote the energy beamforming matrix. Mathematically, $\boldsymbol{w}_{0,j}$ can be recovered by the eigenvalue decomposition (EVD) of $\mathbf{Q}$.

To capture the non-linear power conversion of the EH circuits, a practical non-linear EH model is applied [44]. Accordingly, the amount of power harvested by WD k is

$$\Xi_k = \frac{X_k}{1 + \exp\left(-a_k\left(\boldsymbol{h}_k^T(\widetilde{\boldsymbol{r}}_0)\mathbf{Q}(\boldsymbol{h}_k^T(\widetilde{\boldsymbol{r}}_0))^H - b_k\right)\right)} - Y_k.$$

Herein, a_k, b_k, M_k, $X_k = M_k(1 + \exp(a_kb_k))/\exp(a_kb_k)$, and $Y_k = M_k/\exp(a_kb_k)$ are constants determined by the configurations of the EH circuit.

C. Computing Model

1) Local Computing: As mentioned above, the local computing rate of WD k in bits/second (bps) is

$$R_{L,k} = \frac{f_k T}{\varphi_k T} = \frac{f_k}{\varphi_k}, \quad (5)$$

where f_k (Hz) denote the frequency of WD k, and φ_k (cycles/bit) denotes the complexity of the tasks at WD k. The energy consumption for local computing is

$$E_{L,k} = \kappa f_k^3 T, \quad (6)$$

where κ (Watt/Hz^3) is the capacitance coefficient of the on-chip processor at WD k [45].

2) Task Offloading: The WDs may opt for performing task offloading in rotation by employing part of the harvested energy. Let x_k denote the transmit signal of WD k satisfying $\mathbb{E}\left[|x_k|^2\right] = 1$. The estimated signal at the HAP is

$$y_k = \boldsymbol{w}_k^H \boldsymbol{h}_k(\widetilde{\boldsymbol{r}}_k)\sqrt{p_k}s_k + \boldsymbol{w}_k^H \boldsymbol{n}_0, \qquad (7)$$

where $\boldsymbol{w}_k \in \mathbb{C}^{M\times 1}$ is a receive combing vector to combine the received signal at the HAP for WD k, p_k is the transmit power of WD k, and $\boldsymbol{n}_0 \sim \mathcal{CN}(\boldsymbol{0}, \sigma_0^2 \boldsymbol{I}_M)$ is the AWGN. The offloading rate of WD k is

$$R_{O,k} = \frac{B\tau_k}{T}\log_2\left(1 + \frac{|\boldsymbol{w}_k^H\boldsymbol{h}_k(\widetilde{\boldsymbol{r}}_k)|^2 p_k}{\|\boldsymbol{w}_k^H\|^2\sigma_0^2}\right), \qquad (8)$$

where B (Hz) is the frequency bandwidth.

3) Edge Computing: Similar to [29], to guarantee all offloaded tasks are impeccably executed by edge servers, we have

$$\sum_{i=k}^{K}\varphi_i R_{O,i}T \le f_E \cdot \left(\Delta\tau + \sum_{i=k}^{K}\tau_i\right), \qquad (9)$$

where f_E (Hz) denotes the edge computing frequency.

D. Problem Formulation

Denote by sets $\boldsymbol{\tau} = \{\Delta\tau, \tau_i, i \in \mathcal{I}\}$, $\boldsymbol{p} = \{p_k, k \in \mathcal{K}\}$, $\boldsymbol{f} = \{f_k, k \in \mathcal{K}\}$, $\mathbf{w} = \{\boldsymbol{w}_k, k \in \mathcal{K}\}$, and $\widetilde{\mathbf{r}} = \{\widetilde{\boldsymbol{r}}_i, i \in \mathcal{I}\}$. Our objective is to maximize the SCR of all WDs in the MA-enhanced WP-MEC system by jointly optimizing the time allocation, positions of MAs, energy beamforming matrix, receive combing vectors, and offloading strategies of WDs.

1) Dynamic MA positioning: In this case, the optimization problem is formulated as

$$(\text{PA}): \max_{\boldsymbol{\tau},\boldsymbol{p},\boldsymbol{f},\mathbf{w},\widetilde{\mathbf{r}},\mathbf{Q}\succeq 0}\sum_{k\in\mathcal{K}}(R_{L,k} + R_{O,k}) \qquad (10a)$$

$$s.t. \ \kappa f_k^3 T + p_k\tau_k \le \tau_0\Xi_k, \ k \in \mathcal{K}, \qquad (10b)$$

$$\sum_{i\in\mathcal{I}}\tau_i + \Delta\tau \le T, \ 0 \le \tau_i, \Delta\tau \le T, \qquad (10c)$$

$$\sum_{i=k}^{K}\varphi_i R_{O,i}T \le f_E \cdot \left(\Delta\tau + \sum_{i=k}^{K}\tau_i\right), \ k \in \mathcal{K}, \qquad (10d)$$

$$\text{Tr}(\mathbf{Q}) \le P_{\max}, \qquad (10e)$$

$$\boldsymbol{r}_{i,m} \in \mathcal{C}, \ m \in \mathcal{M}, i \in \mathcal{I}, \qquad (10f)$$

$$\|\boldsymbol{r}_{i,m} - \boldsymbol{r}_{i,n}\|_2 \ge D, \ 1 \le m \ne n \le M, i \in \mathcal{I}, \quad (10g)$$

where $P_{\max}$ is the maximum transmit power of the HAP. (10b) is the energy-causality constraint, which indicates that the energy consumed by each WD cannot exceed the harvested energy. (10d) is caused by the finite edge computing capability. (10g) ensures that the inter-MA distance should be not less than D in actual practice.

2) Semi-dynamic MA positioning: In this case, two APVs are allocated for WPT and task offloading, respectively. Let $\widetilde{\boldsymbol{r}}_u$ denote the APV during entire task offloading stage, i.e., $\widetilde{\boldsymbol{r}}_u = \widetilde{\boldsymbol{r}}_k, \forall k \in \mathcal{K}$. Then, the problem is formulated as

$$(\text{PB}): \max_{\boldsymbol{\tau},\boldsymbol{p},\boldsymbol{f},\mathbf{w},\widetilde{\boldsymbol{r}}_0,\widetilde{\boldsymbol{r}}_u,\mathbf{Q}\succeq 0}\sum_{k\in\mathcal{K}}(R_{L,k} + R_{O,k}) \qquad (11a)$$

$$s.t. \ (10b), (10c), (10d), (10e),$$

$$\boldsymbol{r}_{i,m} \in \mathcal{C}, \ m \in \mathcal{M}, i \in \{0, u\}, \qquad (11b)$$

$$\|\boldsymbol{r}_{i,m} - \boldsymbol{r}_{i,n}\|_2 \ge D, \ 1 \le m \ne n \le M, i \in \{0, u\}. \quad (11c)$$

3) Static MA positioning: In this case, the same APV, denoted by $\widetilde{\boldsymbol{r}}_s$, is adopted throughout the entire time block, i.e., $\widetilde{\boldsymbol{r}}_s = \widetilde{\boldsymbol{r}}_i, \forall i \in \mathcal{I}$. Then, the problem is formulated as

$$(\text{PC}): \max_{\boldsymbol{\tau},\boldsymbol{p},\boldsymbol{f},\mathbf{w},\widetilde{\boldsymbol{r}}_s,\mathbf{Q}\succeq 0}\sum_{k\in\mathcal{K}}(R_{L,k} + R_{O,k}) \qquad (12a)$$

$$s.t. \ (10b), (10c), (10d), (10e),$$

$$\boldsymbol{r}_{s,m} \in \mathcal{C}, \ m \in \mathcal{M}, \qquad (12b)$$

$$\|\boldsymbol{r}_{s,m} - \boldsymbol{r}_{s,n}\|_2 \ge D, \ 1 \le m \ne n \le M. \qquad (12c)$$

Obviously, the above three problems are all highly non-convex due to the non-convexity of objective functions with APVs, as well as the multiplicative terms in (10b) and (10d). The globally optimal solutions can not be obtained by the existing convex optimization tools.

III. SOLUTION FOR DYNAMIC MA POSITIONING

In this section, we propose an efficient AO framework to obtain a sub-optimal solution for (PA). Specifically, the variables are first partitioned as $\{\boldsymbol{\tau}, \boldsymbol{p}, \boldsymbol{f}\}$, $\{\widetilde{\boldsymbol{r}}_0, \mathbf{Q}\}$, and $\{\mathbf{w}, \widetilde{\boldsymbol{r}}_k, k \in \mathcal{K}\}$. Then, we optimize each block of variables alternately, until convergence is achieved.

A. Optimization of $\{\boldsymbol{\tau}, \boldsymbol{p}, \boldsymbol{f}\}$

Given $\{\widetilde{\boldsymbol{r}}_0, \mathbf{Q}\}$ and $\{\mathbf{w}, \widetilde{\boldsymbol{r}}_k, k \in \mathcal{K}\}$, (PA) is reduced to

$$(\text{PA1}): \max_{\boldsymbol{\tau},\boldsymbol{p},\boldsymbol{f}}\sum_{k\in\mathcal{K}}(R_{L,k} + R_{O,k})$$

$$s.t. \ (10b), (10c), (10d).$$

According to [27], the maximum SCR can be obtained if each WD exhausts its harvested energy, i.e.,

$$\kappa f_k^3 T + p_k\tau_k = \tau_0\Xi_k. \qquad (14)$$

Then, $R_{L,k}$ in (5) can be rewritten as

$$R_{L,k}(\tau_0, p_k, \tau_k) = \frac{1}{\varphi_k}\left(\frac{\tau_0\Xi_k - p_k\tau_k}{\kappa T}\right)^{\frac{1}{3}}, \qquad (15)$$

which is non-convex with respect of τ_k and p_k. Then, the auxiliary variable $e_k = p_k\tau_k$ is introduced. Accordingly, $R_{L,k}(\tau_0, p_k, \tau_k)$ in (15) and $R_{O,k}$ in (8) can be rewritten as

$$R_{L,k}(\tau_0, e_k) = \frac{1}{\varphi_k}\left(\frac{\tau_0\Xi_k - e_k}{\kappa T}\right)^{\frac{1}{3}},$$

and

$$R_{O,k}(\tau_k, e_k) = \frac{B\tau_k}{T}\log_2\left(1 + \frac{e_k|\boldsymbol{w}_k^H\boldsymbol{h}_k(\widetilde{\boldsymbol{r}}_k)|^2}{\tau_k\|\boldsymbol{w}_k^H\|^2\sigma_0^2}\right),$$

respectively. Then, (PA1) can be converted to a convex problem as follows:

$$(\text{PA1.1}) : \max_{\boldsymbol{\tau}, \boldsymbol{e}} \sum_{k \in \mathcal{K}} \left(R_{L,k}(\tau_0, e_k) + R_{O,k}(\tau_k, e_k) \right)$$
$$s.t. \ (10c), (10d), 0 \leq e_k \leq \tau_0 \Xi_k, \ k \in \mathcal{K},$$

where $\boldsymbol{e} = \{e_k, k \in \mathcal{K}\}$.

B. Optimization of $\{\widetilde{r}_0, \mathbf{Q}\}$

Given $\{\boldsymbol{\tau}, \boldsymbol{p}, \boldsymbol{f}\}$ and $\{\mathbf{w}, \widetilde{r}_k, k \in \mathcal{K}\}$, (PA) is reduced to a feasibility-checking problem as follows:

$$(\text{PA2}) : \text{Find } \{\widetilde{r}_0, \mathbf{Q} \succeq 0\} \tag{16a}$$
$$s.t. \ (10b), (10e),$$
$$\boldsymbol{r}_{0,m} \in \mathcal{C}, \ m \in \mathcal{M}, \tag{16b}$$
$$\|\boldsymbol{r}_{0,m} - \boldsymbol{r}_{0,n}\|_2 \geq D, \ 1 \leq m \neq n \leq M. \tag{16c}$$

To solve (PA2) efficiently, it is better to convert (PA2) into an explicit goal-based problem. According to (14), auxiliary variable β_k is introduced as energy partition factor. Specifically, part of the harvested energy, i.e., $\beta_k \tau_0 \Xi_k$, is allocated for task offloading, and the residual of harvested energy, i.e., $(1 - \beta_k)\tau_0 \Xi_k$, is allocated for local computing at WD k. Then, $R_{L,k}$ in (5) and $R_{O,k}$ in (8) can be rewritten as

$$R_{L,k}(\widetilde{r}_0, \mathbf{Q}) = \frac{1}{\varphi_k} \left(\frac{(1 - \beta_k)\tau_0 \Xi_k}{\kappa T} \right)^{\frac{1}{3}},$$

and

$$R_{O,k}(\widetilde{r}_0, \mathbf{Q}) = \frac{B \tau_k}{T} \log_2 \left(1 + \frac{\beta_k \tau_0 \Xi_k |\boldsymbol{w}_k^H \boldsymbol{h}_k(\widetilde{r}_k)|^2}{\tau_k \|\boldsymbol{w}_k^H\|^2 \sigma_0^2} \right),$$

respectively. Then, (PA2) is converted into

$$(\text{PA2.1}) : \max_{\widetilde{r}_0, \mathbf{Q} \succeq 0} \sum_{k \in \mathcal{K}} \left(R_{L,k}(\widetilde{r}_0, \mathbf{Q}) + R_{O,k}(\widetilde{r}, \mathbf{Q}) \right)$$
$$s.t. \ (10d), (10e), (16b), (16c),$$

Obviously, (PA2.1) is non-convex since $\widetilde{r}_0$ and $\mathbf{Q}$ are highly coupled. Then, (PA2.1) is further split into two sub-problems with respect to $\widetilde{r}_0$ and $\mathbf{Q}$, respectively.

1) Optimizing $\widetilde{r}_0$: Given $\mathbf{Q}$, (PA2.1) is reduced to

$$(\text{PA2.2}) : \max_{\widetilde{r}_0} \sum_{k \in \mathcal{K}} \left(R_{L,k}(\widetilde{r}_0) + R_{O,k}(\widetilde{r}_0) \right)$$
$$s.t. \ (10d), (16b), (16c).$$

It is computationally prohibitive to search the MA positioning directly, especially when the dimension of $\widetilde{r}_0$ is large. Thank to the fast convergence, easy implementation, and robust searching ability, the PSO algorithm is applied to search a sub-optimal of MA positioning [35].

We firstly randomly initialize N particles with positions, i.e., $\mathcal{P}_0^{(0)} = \{\widetilde{r}_0^{(0),1}, \cdots, \widetilde{r}_0^{(0),n}, \cdots, \widetilde{r}_0^{(0),N}\}$, where the position of each particle represents a candidate solution for the APV, i.e.,

$$\widetilde{r}_0^{(0),n} = \left[\boldsymbol{r}_{0,1}^{(0),n}, \cdots, \boldsymbol{r}_{0,m}^{(0),n}, \cdots, \boldsymbol{r}_{0,M}^{(0),n} \right]$$
$$= \left[\underbrace{x_{0,1}^{(0),n}, y_{0,1}^{(0),n}}_{\text{MA 1}}, \cdots, \underbrace{x_{0,m}^{(0),n}, y_{0,m}^{(0),n}}_{\text{MA m}}, \cdots, \underbrace{x_{0,M}^{(0),n}, y_{0,M}^{(0),n}}_{\text{MA M}} \right].$$

To characterize the movement of each particle, the velocities are randomly initialized as $\mathcal{V}_0^{(0)} = \{\boldsymbol{v}_0^{(0),1}, \cdots, \boldsymbol{v}_0^{(0),n}, \cdots, \boldsymbol{v}_0^{(0),N}\}$, where

$$\widetilde{\boldsymbol{v}}_0^{(0),n} = \left[\boldsymbol{v}_{0,1}^{(0),n}, \cdots, \boldsymbol{v}_{0,m}^{(0),n}, \cdots, \boldsymbol{v}_{0,M}^{(0),n} \right]$$
$$= \left[\underbrace{v_{x,0,1}^{(0),n}, v_{y,0,1}^{(0),n}}_{\text{MA 1}}, \cdots, \underbrace{v_{x,0,m}^{(0),n}, v_{y,0,m}^{(0),n}}_{\text{MA m}}, \cdots, \underbrace{v_{x,0,M}^{(0),n}, v_{y,0,M}^{(0),n}}_{\text{MA M}} \right].$$

The key idea of the standard PSO algorithm is that each particle successively updates its position according to the personal best position found so far by itself, denoted by $\widetilde{r}_{0,pbest}^n$, and the global best position found so far by all particles, denoted by $\widetilde{r}_{0,gbest}$. However, the particles in the standard PSO algorithm are unable to explore the local neighborhoods to find a better solutions, which indicates that the standard PSO algorithm tends to be trapped in undesired sub-optima [46].

To cope with this problem, the variable local search is introduced to the standard PSO algorithm, shortened to PSO-VLS, to inspect all the personal best positions within the variable neighborhood. Specifically, $\widetilde{r}_{0,gbest}$ is replaced by the best position found so far by particle n and its neighbors, denoted by $\widetilde{r}_{0,nbest}^n$. The neighborhood is determined by the distances between particles, calculated as $\|\widetilde{r}_0^{(t),n} - \widetilde{r}_0^{(t),\bar{n}}\|_2$, where t denotes the iteration index. In addition, to improve the quality of $\widetilde{r}_{0,nbest}^n$ with the increasing of iteration times, the number of particles within the neighborhood, denoted by $N^{(t),n}$, also increases. At the beginning, the neighborhood of each particle is set as the particle itself, i.e., $N^{(0),n} = 1$. Then, the size of neighborhood grows in scale as iteration increases as follows:

$$N^{(t+1),n} = \begin{cases} 1, & \text{if } t = 0, \\ N^{(t),n} + \beta, & \text{if } N^{(t),n} + \beta < N, \\ N, & \text{if } N^{(t),n} + \beta \geq N, \end{cases} \tag{19}$$

which indicates that the neighborhood will be gradually expanded to include all particles.

For each iteration, the velocity and position of particle n are updated as

$$\widetilde{\boldsymbol{v}}_0^{(t+1),n} = \omega \widetilde{\boldsymbol{v}}_0^{(t),n} + c_1 \phi_1 \left(\widetilde{r}_{0,pbest}^n - \widetilde{r}_0^{(t),n} \right)$$
$$+ c_2 \phi_2 \left(\widetilde{r}_{0,nbest}^n - \widetilde{r}_0^{(t),n} \right), \tag{20}$$

and

$$\widetilde{r}_0^{(t+1),n} = \widetilde{r}_0^{(t),n} + \widetilde{\boldsymbol{v}}_0^{(t+1),n}, \tag{21}$$

respectively. Herein, $\omega \in [\omega_{\min}, \omega_{\max}]$ is inertia weight of velocity, which linearly decreases with t so as to improve the convergence speed, i.e.,

$$\omega = w_{\max} - (\omega_{\max} - \omega_{\min}) \cdot (t/\mathcal{T}), \tag{22}$$

where $\mathcal{T}$ is the default total number of iterations. c_1 and c_2 are the learning factors, indicating the step size of each particle moving towards the best position of itself and the best position of all particles, respectively. ϕ_1 and ϕ_2 are assigned randomly within $[0, 1]$, whose aim is to enhance the randomness of the search for escaping from local optima.

To avoid excessive roaming of particles, a velocity range, i.e., $[v_{\min}, v_{\max}]$, is imposed to each entry of $\widetilde{\boldsymbol{v}}_0^{(t),n}$, i.e.,

$$v_{x,0,m}^{(t),n} = \begin{cases} v_{\min}, & \text{if } v_{x,0,m}^{(t),n} \leq v_{\min}, \\ v_{\max}, & \text{if } v_{x,0,m}^{(t),n} \geq v_{\max}, \\ v_{x,0,m}^{(t),n}, & \text{otherwise,} \end{cases} \tag{23}$$

and

$$v_{y,0,m}^{(t),n} = \begin{cases} v_{\max}, & \text{if } v_{y,0,m}^{(t),n} \leq v_{\min}, \\ v_{\max}, & \text{if } v_{y,0,m}^{(t),n} \geq v_{\max}, \\ v_{y,0,m}^{(t),n}, & \text{otherwise.} \end{cases} \tag{24}$$

In addition, each entry of $\widetilde{\boldsymbol{r}}_0^{(t+1),n}$ should be checked to ensure that each particle stays inside the available region. As such, we have

$$x_{0,m}^{(t),n} = \begin{cases} -A/2, & \text{if } x_{0,m}^{(t),n} \leq -A/2, \\ A/2, & \text{if } x_{0,m}^{(t),n} \geq A/2, \\ x_{0,m}^{(t),n}, & \text{otherwise,} \end{cases} \tag{25}$$

and

$$y_{0,m}^{(t),n} = \begin{cases} -A/2, & \text{if } y_{0,m}^{(t),n} \leq -A/2, \\ A/2, & \text{if } y_{0,m}^{(t),n} \geq A/2, \\ y_{0,m}^{(t),n}, & \text{otherwise.} \end{cases} \tag{26}$$

To evaluate the fitness of each particle, the fitness function is formulated as

$$\mathcal{F}_0(\widetilde{\boldsymbol{r}}_0^{(t),n}) = \begin{cases} R(\widetilde{\boldsymbol{r}}_0^{(t),n}), & \text{if (10d) and (16c) hold,} \\ -1, & \text{otherwise,} \end{cases} \tag{27}$$

where $R(\widetilde{\boldsymbol{r}}_0^{(t),n}) = \sum_{k \in \mathcal{K}} \left(R_{L,k}(\widetilde{\boldsymbol{r}}_0^{(t),n}) + R_{O,k}(\widetilde{\boldsymbol{r}}_0^{(t),n}) \right)$. With the fitness evaluation conducted on the particles during each iteration, their personal and neighborhood best positions are updated until convergence or the total number of iterations is reached. The best position among the particles is selected to be a sub-optimal solution for $\widetilde{\boldsymbol{r}}_0$.

The details of the PSO-VLS algorithm for solving (PA2.2) are summarized in Algorithm 1, whose convergence is guaranteed by the following inequalities:

$$\mathcal{F}_0(\widetilde{\boldsymbol{r}}_{0,pbest}^{(t+1),n}) \geq \mathcal{F}_0(\widetilde{\boldsymbol{r}}_{0,pbest}^{(t),n}),$$

and

$$\mathcal{F}_0(\widetilde{\boldsymbol{r}}_{0,nbest}^{(t+1),n}) \geq \mathcal{F}_0(\widetilde{\boldsymbol{r}}_{0,nbest}^{(t),n}),$$

respectively. Accordingly, we have

$$\mathcal{F}_0(\widetilde{\boldsymbol{r}}_0^{(t+1)}) \geq \mathcal{F}_0(\widetilde{\boldsymbol{r}}_0^{(t)}),$$

which indicates that fitness value of $\widetilde{\boldsymbol{r}}$ is non-decreasing during the iterations. In addition, the objective value of (PA2.2) is bounded due to (10e). Therefore, the convergence of Algorithm 1 is guaranteed. Compared to the standard PSO, each particle in PSO-VLS needs to calculate the distances to other particles to conform its neighbors. The computational complexity of Algorithm 1 is $\mathcal{O}\left(\mathcal{T}N^2\right)$.

Algorithm 1 PSO-VLS algorithm for solving (PA2.2).

1: Initialize N particles with positions $\mathcal{P}_0^{(0)}$ and velocities $\mathcal{V}_0^{(0)}$, $\widetilde{\boldsymbol{r}}_{0,pbest}^n = \widetilde{\boldsymbol{r}}_0^{(0),n}$, and $\widetilde{\boldsymbol{r}}_{0,nbest}^n = \widetilde{\boldsymbol{r}}_0^{(0),n}$.
2: Evaluate the fitness of each particle.
3: **for** $t = 1$ to $\mathcal{T}$ **do**
4:     Update the inertia weight ω according to (22).
5:     **for** $n = 1$ to N **do**
6:         Update the velocity of particle n according to (20), (23) and (24).
7:         Update the position of particle n according to (21), (25) and (26).
8:         Evaluate the fitness of particle n according to (27).
9:         **if** $\mathcal{F}_0(\widetilde{\boldsymbol{r}}_0^{(t),n}) > \mathcal{F}_0(\widetilde{\boldsymbol{r}}_{0,pbest}^n)$ **then**
10:           $\widetilde{\boldsymbol{r}}_{0,pbest}^n \leftarrow \widetilde{\boldsymbol{r}}_0^{(t),n}$.
11:         **end if**
12:         **if** $\max\{$neighborhood fitness values$\} > \mathcal{F}_0(\widetilde{\boldsymbol{r}}_{0,nbest}^n)$ **then**
13:           $\widetilde{\boldsymbol{r}}_{0,nbest}^n = \arg\max\{$neighborhood fitness values$\}$.
14:         **end if**
15:     **end for**
16: **end for**
17: Obtain $\widetilde{\boldsymbol{r}}_0 = \arg\max\left\{\mathcal{F}_0(\widetilde{\boldsymbol{r}}_{0,nbest}^n), \; n = 1, \cdots, N\right\}$.

---

2) Optimizing $\mathbf{Q}$: Given $\widetilde{\boldsymbol{r}}_0$, (PA2.1) is reduced to

$$(\text{PA2.3}): \max_{\mathbf{Q} \succeq 0, \boldsymbol{z}} \sum_{k \in \mathcal{K}} \left(R_{L,k}(\mathbf{Q}) + R_{O,k}(\mathbf{Q}) \right)$$

$$s.t. \ (10d), (10e),$$

$$z_k = \exp\left(-a_k \left(\boldsymbol{h}_k^T(\widetilde{\boldsymbol{r}}_0) \mathbf{Q} \left(\boldsymbol{h}_k^T(\widetilde{\boldsymbol{r}}_0) \right)^H - b_k \right) \right), \tag{28a}$$

where z_k is the auxiliary variable, and $\boldsymbol{z} = \{z_k, k \in \mathcal{K}\}$. Then, Ξ_k can be rewritten as $\Xi_k = \dfrac{X_k}{1+z_k} - Y_k$. Since Ξ_k is non-convex of z_k, the SCA method is applied to approximate Ξ_k as

$$\Xi_k = X_k \left(\frac{1}{1 + z_k^{(r)}} - \frac{z_k - z_k^{(r)}}{\left(1 + z_k^{(r)}\right)^2} \right) - Y_k,$$

where $z_k^{(r)}$ is a feasible solution in the r-th iteration. Accordingly, (PA2.3) is simplified to a convex problem.

C. Optimization of $\{\mathbf{w}, \widetilde{\boldsymbol{r}}_k, k \in \mathcal{K}\}$

Given $\{\boldsymbol{\tau}, \boldsymbol{p}, \boldsymbol{f}\}$ and $\{\widetilde{\boldsymbol{r}}_0, \mathbf{Q}\}$, (PA) is reduced to

$$(\text{PA3}): \max_{\mathbf{w}, \widetilde{\boldsymbol{r}}_k, k \in \mathcal{K}} \sum_{k \in \mathcal{K}} (R_{L,k} + R_{O,k}) \tag{29a}$$

$$s.t. \ (10d),$$

$$\boldsymbol{r}_{k,m} \in \mathcal{C}, \ m \in \mathcal{M}, k \in \mathcal{K}, \tag{29b}$$

$$\|\boldsymbol{r}_{k,m} - \boldsymbol{r}_{k,n}\|_2 \geq D, \ 1 \leq m \neq n \leq M, k \in \mathcal{K}. \tag{29c}$$

Before solving (PA3), it is commonly believed that (10d) can be omitted safely, due to the fact that a better SCR performance can be obtained by maximizing the capacity. Ac-

---

Algorithm 2 AO framework for solving (PA).

1: Initialize $\mathbf{w}$, $\widetilde{r}_0$, $\mathbf{Q}$ and $\boldsymbol{z}$.
2: **for** $k = 1$ to K **do**
3:     Recover $\widetilde{r}_k$ by solving (PA3.2) with the PSO-VLS algorithm.
4: **end for**
5: **repeat**
6:     Update $\boldsymbol{\tau}$ and $\boldsymbol{e}$ by solving (PA1.1) with CVX [47], and calculate $p_k = e_k/\tau_k$, $\forall k \in \mathcal{K}$.
7:     Calculate $\beta_k = p_k\tau_k/\tau_0\Xi_k$, $\forall k \in \mathcal{K}$.
8:     Update $\widetilde{r}_0$ by solving (PA2.2) with the PSO-VLS algorithm shown in Algorithm 1.
9:     **repeat**
10:        Update $\mathbf{Q}$ and $\boldsymbol{z}$ by CVX [47].
11:     **until** Convergence.
12: **until** Convergence.
13: Calculate $\boldsymbol{w}_k$ according to (31), $\forall k \in \mathcal{K}$.
14: **Return:** $\boldsymbol{\tau}$, $\boldsymbol{p}$, $\boldsymbol{f}$, $\mathbf{w}$, $\widetilde{r}$, $\mathbf{Q}$.

---

cordingly, removing the terms irrelevant to $\{\widetilde{r}_k, \boldsymbol{w}_k, k \in \mathcal{K}\}$, (PA3) can be split into K independent sub-problems, i.e.,

$$(\text{PA3.1}): \max_{\boldsymbol{w}_k,\widetilde{r}_k} R_{O,k} \qquad (30a)$$
$$s.t. \ (29b),(29c).$$

Given $\widetilde{r}_k$, the optimal $\boldsymbol{w}_k$ can be designed by the MRC criterion [28], i.e.,

$$\boldsymbol{w}_k = \sqrt{p_k}\boldsymbol{h}_k(\widetilde{r}_k)/\sigma_0^2, \ k \in \mathcal{K}. \qquad (31)$$

Then, (PA3.1) can be equivalently to the channel gain maximization problem as follows

$$(\text{PA3.2}): \max_{\widetilde{r}_k}, \|\boldsymbol{h}_k(\widetilde{r}_k)\|^2$$
$$s.t. \ (29b),(29c).$$

The PSO-VLS algorithm is also applied to solve (PA3.2), and the fitness of particle n during the t-th iteration is evaluated by

$$\mathcal{F}_k(\widetilde{r}_k^{(t),n}) = \begin{cases} \|\boldsymbol{h}_k(\widetilde{r}_k^{(t),n})\|^2, & \text{if (29c) holds,} \\ -1, & \text{otherwise.} \end{cases}$$

D. Convergence and Complexity Analysis

The procedure for solving (PA) is summarized in Algorithm 2. As mentioned above, the PSO-VLS algorithm for solving (PA2.2) is guaranteed to converge. Similarly, the channel gain of each WD for task offloading increases by optimizing $\widetilde{r}_k$ with PSO-VLS algorithm, i.e., $\|\boldsymbol{h}_k(\widetilde{r}_k^{(t+1),n})\|^2 \geq \|\boldsymbol{h}_k(\widetilde{r}_k^{(t),n})\|^2$. More tasks can be executed by setting $\{\boldsymbol{\tau}, \boldsymbol{p}, \boldsymbol{f}\}$ more properly with the guidance of other variables. In addition, the SCR performance increases by optimizing $\mathbf{Q}$ with SCA method, because more energy can be harvested for task execution. Meanwhile, the performance of the MA-enhanced WP-MEC system is bounded by the communication resource and computing capability. Therefore, the convergence of Algorithm 2 is guaranteed.

The complexity is then analyzed as follows. (PA1.1) can be solved with complexity of $\mathcal{O}\big((2K+2)^{3.5}\big)$, where $(2K+2)$ is the number of variables. In addition, $\mathbf{Q}$ and $\boldsymbol{z}$ can be obtained by solving (PA2.3) with worst-case complexity of $\mathcal{O}(L_{\text{SCA}}(M^2 + K)^{3.5})$ [48], where M^2 is the number of variables in matrix $\mathbf{Q}$, and L_{SCA} is the number of iterations required to converge. Therefore, the total complexity of Algorithm 2 is $\mathcal{O}\Big(K\mathcal{T}N^2 + L_{\text{AO}}\big((2K+2)^{3.5} + L_{\text{SCA}}(M^2 + K)^{3.5} + \mathcal{T}N^2\big)\Big)$, where L_{AO} is the number of iterations required till convergence.

IV. EXTENSION TO SEMI-DYNAMIC AND STATIC MA POSITIONING

In this section, the AO framework with the PSO-VLS algorithm is extended to solve the formulated problems for the other two types of MA positioning configurations, i.e., semi-dynamic MA positioning and static MA positioning. It is observed that the variables, i.e., $\{\boldsymbol{\tau}, \mathbf{Q}, \mathbf{w}, \boldsymbol{p}, \boldsymbol{f}\}$, in (PB) and (PC) can be optimized using the similar methods as (PA). As such, we only focus on the optimization of APVs.

A. Optimization of $\widetilde{r}_u$ for Semi-dynamic MA Positioning

In the case, the APV optimization sub-problem for (PB) is

$$(\text{PB1}): \max_{\widetilde{r}_u} \sum_{k \in \mathcal{K}} R_{O,k}(\widetilde{r}_u)$$
$$s.t. \ (10d),(11b),(11c),$$

where

$$R_{O,k}(\widetilde{r}_u) = \frac{B\tau_k}{T} \log_2 \Big(1 + \frac{\beta_k\tau_0\Xi_k|\boldsymbol{w}_k^H\boldsymbol{h}_k(\widetilde{r}_u)|^2}{\tau_k\|\boldsymbol{w}_k^H\|^2\sigma_0^2}\Big).$$

The PSO-VLS algorithm is applied to solve (PB1), and the fitness of particle n in the t-th iteration can be evaluated as

$$\mathcal{F}_u(\widetilde{r}_u^{(t),n}) = \begin{cases} R_{O,k}(\widetilde{r}_u^{(t),n}), & \text{if (10d) and (11c) hold,} \\ -1, & \text{otherwise.} \end{cases}$$

The procedure for solving (PB) is summarized in Algorithm 3, whose convergence can be demonstrated analogous to that of Algorithm 2. Based on the analyses above, the total complexity of Algorithm 3 is $\mathcal{O}\Big(L_{\text{AO}}\big((2K+2)^{3.5} + L_{\text{SCA}}(M^2 + K)^{3.5} + 2\mathcal{T}N^2\big)\Big)$.

B. Optimization of $\widetilde{r}_s$ for Static MA Positioning

In the case, the APV optimization sub-problem for (PC) is

$$(\text{PC1}): \max_{\widetilde{r}_s} \sum_{k \in \mathcal{K}} \big(R_{L,k}(\widetilde{r}_s) + R_{O,k}(\widetilde{r}_s)\big)$$
$$s.t. \ (10d),(12b),(12c).$$

where

$$R_{L,k}(\widetilde{r}_s) = \frac{1}{\varphi_k} \Big(\frac{(1 - \beta_k)\tau_0\Xi_k(\widetilde{r}_s)}{\kappa T}\Big)^{\frac{1}{3}},$$

and

$$R_{O,k}(\widetilde{r}_s) = \frac{B\tau_k}{T} \log_2 \Big(1 + \frac{\beta_k\tau_0\Xi_k(\widetilde{r}_s)|\boldsymbol{w}_k^H\boldsymbol{h}_k(\widetilde{r}_s)|^2}{\tau_k\|\boldsymbol{w}_k^H\|^2\sigma_0^2}\Big).$$

Algorithm 3 AO framework for solving (PB).

1: Initialize $\mathbf{w}$, $\widetilde{r}_0$, $\widetilde{r}_u$, $\mathbf{Q}$ and $\mathbf{z}$.
2: **repeat**
3: Update time allocation $\boldsymbol{\tau}$ and $\mathbf{e}$ by solving (PA1.1) with CVX [47], and calculate $p_k = e_k/\tau_k$, $k \in \mathcal{K}$.
4: Calculate $\beta_k = p_k\tau_k/\tau_0\Xi_k$, $k \in \mathcal{K}$.
5: Update $\widetilde{r}_0$ by using the PSO-VLS algorithm.
6: **repeat**
7:    Update $\mathbf{Q}$ and $\mathbf{z}$ by CVX [47].
8: **until** Convergence.
9: Recover $\widetilde{r}_u$ by solving (PB1) with the PSO-VLS algorithm.
10: **until** Convergence.
11: Calculate $\mathbf{w}_k$ according to (31), $k \in \mathcal{K}$.
12: **Return:** $\boldsymbol{\tau}$, $\mathbf{p}$, $\mathbf{f}$, $\mathbf{w}$, $\widetilde{r}_0$, $\widetilde{r}_u$, $\mathbf{Q}$.

We solve (PC1) by the PSO-VLS algorithm and calculated the fitness of particle n in the t-th iteration as

$$\mathcal{F}_s(\widetilde{\boldsymbol{r}}_s^{(t),n}) = \begin{cases} R(\widetilde{\boldsymbol{r}}_s^{(t),n}), & \text{if (10d) and (12c) hold}, \\ -1, & \text{otherwise}, \end{cases}$$

where $R(\widetilde{\boldsymbol{r}}_s^{(t),n}) = \sum_{k\in\mathcal{K}} \left(R_{L,k}(\widetilde{\boldsymbol{r}}_s^{(t),n}) + R_{O,k}(\widetilde{\boldsymbol{r}}_s^{(t),n}) \right)$.

The procedure for solving (PC) is summarized in Algorithm 4. The convergence of Algorithm 4 can be verified in a similar way as Algorithm 2. The complexity of Algorithm 4 is $\mathcal{O}\left(L_{\text{AO}} \left((2K+2)^{3.5} + L_{\text{SCA}}(M^2 + K)^{3.5} + \mathcal{T}N^2 \right) \right)$.

Algorithm 4 AO framework for solving (PC).

1: Initialize $\mathbf{w}$, $\widetilde{r}_s$, $\mathbf{Q}$ and $\mathbf{z}$.
2: **repeat**
3: Update time allocation $\boldsymbol{\tau}$ and $\mathbf{e}$ by solving (PA1.1) with CVX [47], and calculate $p_k = e_k/\tau_k$, $k \in \mathcal{K}$.
4: Calculate $\beta_k = p_k\tau_k/\tau_0\Xi_k$, $k \in \mathcal{K}$.
5: Update $\widetilde{r}_s$ by solving (PC1) with the PSO-VLS algorithm.
6: **repeat**
7:    Update $\mathbf{Q}$ and $\mathbf{z}$ by CVX [47].
8: **until** Convergence.
9: **until** Convergence.
10: Calculate $\mathbf{w}_k$ according to (31), $k \in \mathcal{K}$.
11: **Return:** $\boldsymbol{\tau}$, $\mathbf{p}$, $\mathbf{f}$, $\mathbf{w}$, $\widetilde{r}_s$, $\mathbf{Q}$.

V. NUMERICAL RESULTS

Numerical results are provided to evaluate the SCR performance of the MA-enhanced WP-MEC systems, where the available region is set as a square area centered at (0.0) meter (m), i.e., $\mathcal{C} = [-A/2, A/2] \times [-A/2, A/2]$. The WDs are randomly located around the HAP with distances following uniform distribution from 7 m to 8 m [29]. It is assumed that each entry in PRV $\boldsymbol{g}_k$ follows $g_{k,l} \sim \mathcal{CN}(0, C_0 d_k^{-\alpha})$, where C_0 denotes the large-scale fading at reference distance 1 m, and α is the path-loss exponent. The other parameters are listed in TABLE I.

TABLE I
PARAMETER SETTINGS

Description	Parameter and Value
Communication model	$T = 1$ second, $M = 8$, $K = 6$
	$B = 50$ KHz
	$\lambda = 0.1$ m, $L_k = 10$, [35]
	$A = 3\lambda$, $D = 0.5\lambda$ [35]
	$C_0 = (\lambda/4\pi)^2$, $\alpha = 2.2$
	$P_{\max} = 40$ dBm, $\sigma_0^2 = -80$ dBm
EH model [49]	$M_k = 0.024$ W, $a_k = 150$, $b_k = 0.014$
Computation model [29]	$f_E = 0.4$ GHz
	$\varphi_k = 1000$ cycles/bit
	$\kappa = 10^{-26}$ Watt/Hz^3
PSO-VLS algorithm [35]	$\mathcal{T} = 200$, $\beta = 1$
	$\omega_{\min} = 0.4$, $\omega_{\max} = 0.9$
	$c_1 = 1.4$, $c_2 = 1.4$
	$v_{\min} = -0.5\lambda$, $v_{\max} = 0.5\lambda$

A. Convergence Behaviors

Fig. 3 depicts the convergence behaviors of the PSO-VLS algorithm (i.e., Algorithm 1) and the AO frameworks for the three cases with different types of MA positioning configurations (i.e., Algorithms 2~4). From Fig. 3(a), it is obvious that the PSO-VLS algorithm achieves a better SCR performance than the standard PSO algorithm after a slightly more iterations. The reason is that each particle in the PSO-VLS algorithm is reasonable for the search of its variable neighborhoods instead of all the particles. The diversity of particles in the PSO-VLS algorithm keep longer, and slightly more iterations are required. Fig. 3(b) shows that the SCRs obtained by Algorithms 2~4 increase with iteration index and converge within almost the same number of iterations, which demonstrates the fast convergence and good robustness of the proposed AO frameworks. Specifically, the SCR obtained by Algorithm 2 increases from 2.0×10^5 bps to 3.144×10^5 bps, which yields 57.2% performance improvement. In addition, there are 48.7% and 45.1% performance improvements for Algorithm 3 and Algorithm 4, respectively. Algorithm 2 achieves the highest performance improvement over Algorithms 3 and 4, because more spatial DoFs can be obtained by flexibly adjusting the positions of MAs during each time slot, inevitably incurring a certain magnify in computational complexity.

B. Effectiveness of the AO Frameworks

To evaluate the effectiveness of the proposed AO frameworks, the performance comparisons between the obtained solutions and the optimal solutions achieved by the exhaustive method are carried out. Before solving the formulated problems (i.e., PA, PB, and PC) by the exhaustive method, some conversions are made. Firstly, the continuous spatial region $\mathcal{C}$ is equally divided into 16 parts, whose centers can be used for antenna deployments. Then, to further reduce the computational complexity of the exhaustive method, the number of antennas is set to 2 (i.e., $M = 2$). At last, the continuous value
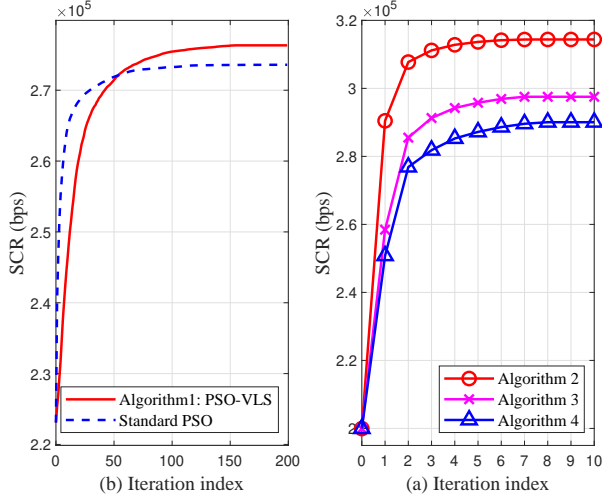
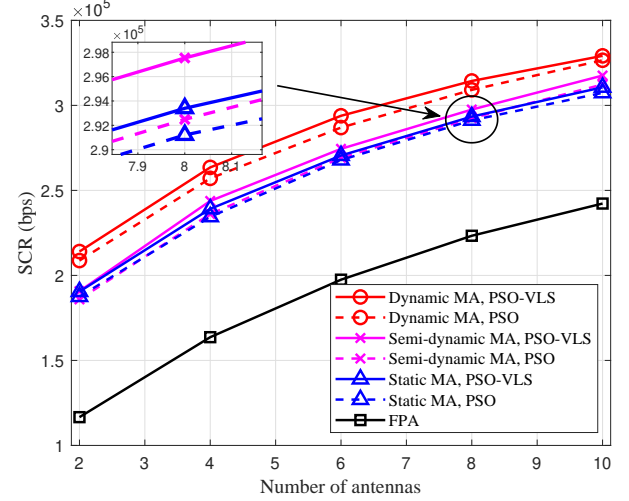Fig. 3. Convergence behaviors of the proposed algorithms.

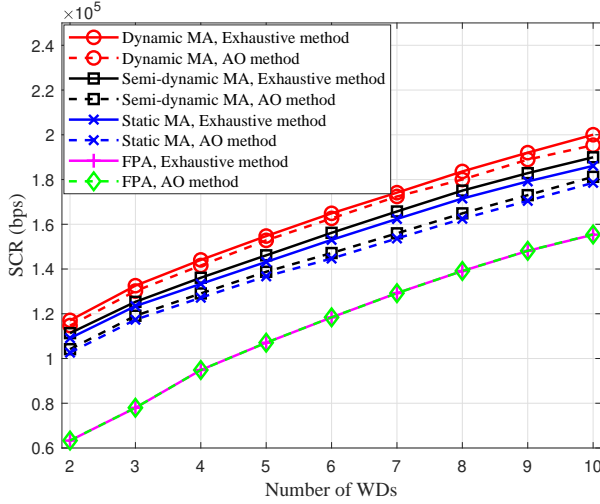Fig. 5. SCR versus the number of antennas at the HAP.

Fig. 4. The performance comparisons between the proposed AO frameworks and the exhaustive method with $M = 2$.

of τ_0 is assumed to take a finite number of values, i.e., obtained by equally quantizing the interval $[0, T]$ into 100 levels. Thus, we have $\tau_0 \in \{0, \Delta T, 2\Delta T, \cdots, T\}$, where $\Delta T = T/100$. Based on these conversions, the formulated problems are resolved by the AO frameworks and the exhaustive method.

Fig. 4 depicts the performance comparisons between the proposed AO frameworks and the exhaustive method. Compared to the exhaustive method, the AO frameworks achieve near-to-optimal performances with only 1.68 %, 5.42 % and 4.96 % degradations in average for the schemes with dynamic MA positioning, semi-dynamic MA positioning, and static MA positioning, respectively. It demonstrates the effectiveness of the AO frameworks. It is worth noting that the AO frameworks achieve almost the same performance with the exhaustive method for the FPA scheme, which further verifies the effectiveness of the AO frameworks.

C. Performance Comparison

The following benchmark schemes are defined for performance comparison:

- **FPA**: The HAP is equipped with FPA-based uniform planar array (UPA) with M antennas, spaced by D [35]. Specifically, each antenna at the HAP has a fixed position. The corresponding energy beamforming matrix (i.e., $\mathbf{Q}$) can be solved via the SCA method shown in Section III B 2). In addition, the receive combing vectors (i.e., $\mathbf{w}$) can be calculated by using (31).
- **PSO**: The APV of all MAs are optimized by the standard PSO algorithm [35]. Specifically, compared to the PSO-VLS algorithm, the best position found so far by the n-th particle and its neighbors (i.e., $\widetilde{\boldsymbol{r}}_{0,nbest}^n$) is replaced by the global best position found so far by all particles (i.e., $\widetilde{\boldsymbol{r}}_{0,gbest}$). For each iteration, the velocity and position of the n-th particle are updated as $\widetilde{\boldsymbol{v}}_0^{(t+1),n} = \omega \widetilde{\boldsymbol{v}}_0^{(t),n} + c_1\phi_1\big(\widetilde{\boldsymbol{r}}_{0,pbest}^n - \widetilde{\boldsymbol{r}}_0^{(t),n}\big) + c_2\phi_2\big(\widetilde{\boldsymbol{r}}_{0,gbest} - \widetilde{\boldsymbol{r}}_0^{(t),n}\big)$ and (21), respectively.
- **Offloading-only**: The harvested energy of each WD is exhausted to perform task offloading, instead of implementing the local computing. That is to say, the local computing is in unavailable for this benchmark scheme.

Fig. 5 depicts the SCR versus the number of MAs, i.e., M. It can be seen that the SCRs of all schemes increase with K, which can be explained as follows. On the one hand, the HAP can perform a more fine-grained energy emission during the WPT stage by taking into account the spatial diversities of all WDs. On the other hand, the stronger beamforming gain can be obtained for task offloading. The MA schemes significantly outperform the FPA scheme, because the MA schemes obtain more spatial DoFs by properly optimizing the positions of MAs. Obviously, the PSO-VLS schemes obtain better performance than that of the counterpart PSO schemes, which further validates the benefit of PSO-VLS algorithm in escape from the local optima.

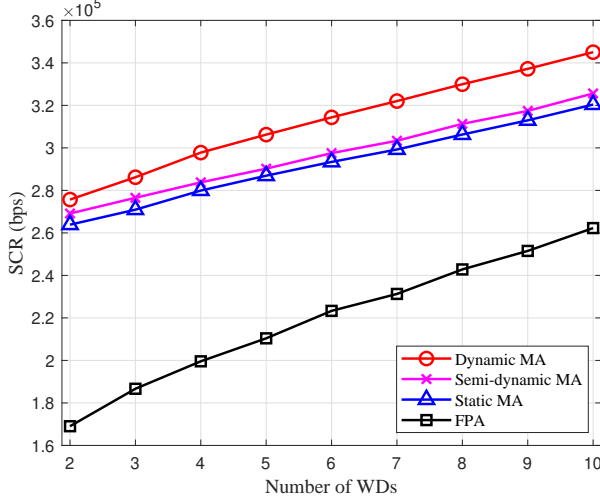Fig. 6 shows the SCR versus the number of WDs, i.e., K.

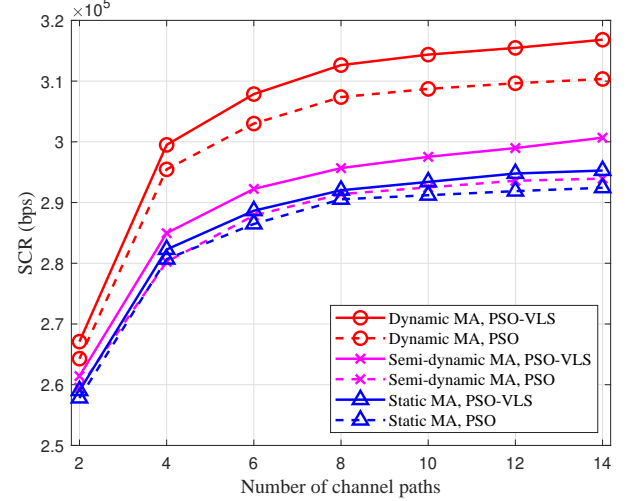Fig. 6. SCR versus the number of WDs.

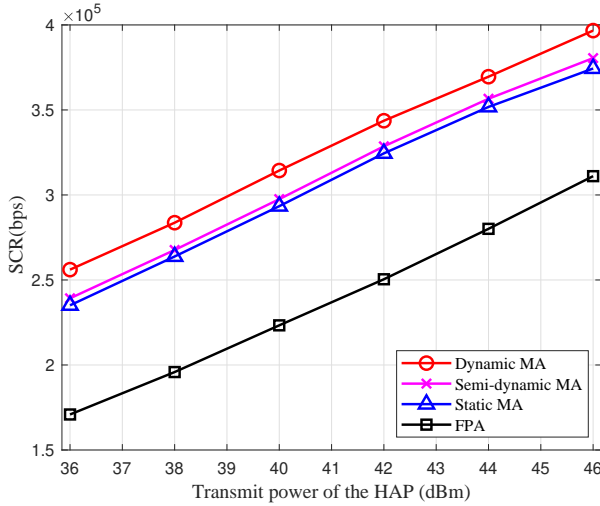Fig. 8. SCR versus the number of channel paths.

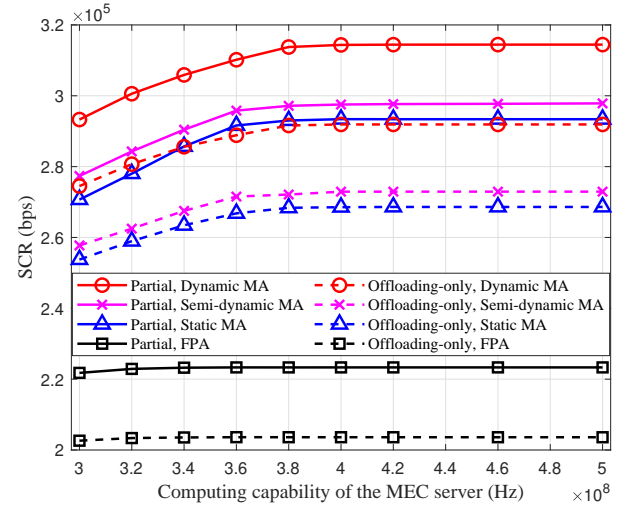Fig. 7. SCR versus the power budget of HAP.

Fig. 9. SCR versus the edge computing capability.

It is observed that the SCR performance increases with K for all schemes. The MA schemes outperform the FPA scheme, which again demonstrates the superiority of MAs over the FPAs in the WP-MEC systems. Specifically, the schemes with dynamic, semi-dynamic, and static MA positioning achieve about 42.4%, 35.3% and 33.2% SCR gains over the FPA scheme, respectively. In addition, as K increases, the performance gaps between the dynamic MA scheme and the other two MA schemes increase. It is due to the fact that the APV for each WD in the dynamic MA scheme is optimized for easy task offloading.

Fig. 7 depicts the SCRs of different schemes versus the power budget of HAP, i.e., $P_{\max}$. As expected, with the increase of $P_{\max}$, the SCRs of all schemes increase. The MA schemes achieve more performance gains over the FPA scheme, which again demonstrates the superiority of MA-enhanced WP-MEC over the conventional WP-MEC with FPAs. Significantly, the MA schemes achieve about 42.5% and 23.4% performance gains over the FPA scheme when

$P_{\max} = 36$ dBm and $P_{\max} = 46$ dBm, respectively. This indicates that the MA schemes are more applicable for the WP-MEC systems with low power budget over the FPA scheme.

Fig. 8 illustrates the SCR versus the number of channel paths of each WD, i.e., L_k. The SCRs of all schemes increase with L_k because higher multi-path diversity can be exploited and more design flexibility of energy beams and receive combing vectors at the HAP can be obtained. In addition, the performance gaps between the PSO-VLS schemes and their counterpart PSO schemes increases with L_k, which further validates the superiority of PSO-VLS algorithm over the standard PSO algorithm, especially there exist lots of channel paths.

Fig. 9 shows the SCR versus the edge computing capability, i.e., f_E. The benefits of exploiting MAs over FPAs are further conformed, especially when the edge computing capability is sufficient. The SCRs of all schemes increase with the edge computing capability firstly and become stable afterwards. This is due to the fact that the improvement in edge computing
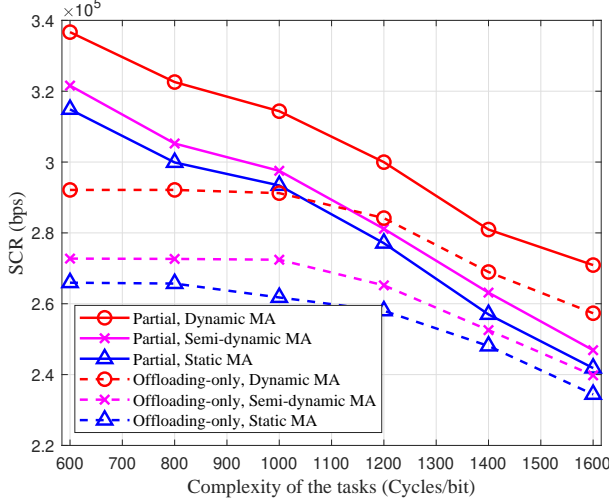
Fig. 10. SCR versus the complexity of the tasks.

capability permits more tasks to be handled by task offloading, yet restricted by the WPT efficiency and channel state for task offloading. Therefore, for cost saving, it is crucial to take into account the network topology and equip the edge server with proper computing capability. Moreover, the partial schemes outperforms the counterpart offloading-only schemes because the partial schemes generally permit local computing and task offloading in parallel.

Fig. 10 depicts the SCR versus the complexity of the tasks, φ_k. The SCRs of all schemes decrease as φ_k increases, because the energy required to process 1-bit raw task via local computing is amplified and the time consumption via the edge computing becomes longer. The partial scheme outperforms the counterpart offloading-only schemes, which further validates the benefits of optimizing offloading strategies at the WDs.

VI. CONCLUSION

In this paper, we have investigated three types of MA positioning configurations for the MA-enhanced WP-MEC systems, aiming to balance the enhancement of performance against the augmentation in implementation intricacy cased by the MA positioning optimization. For practical purposes, A non-linear EH model has been applied to characterize the power conversion of EH circuit at each WD. In addition, due to the practical hardware limitations, the finite edge computing capability has been considered. The SCR maximization problems for the three cases with different MA positioning configurations have been formulated by jointly optimizing the time allocation, positions of MAs, energy beamforming matrix, receive combing vectors, and offloading strategies of WDs. The dynamic MA positioning optimization has been studied at first, and the efficient AO framework with the PSO-VLS algorithm has been proposed. Then, the PSO-VLS algorithm has been extended to optimize the semi-dynamic and static MA positioning. Numerical results have revealed the superiority of exploiting MAs in WP-MEC systems over the conventional FPAs. Furthermore, analytical results have

demonstrated that compared to the standard PSO algorithm, the PSO-VLS algorithm can jump from local optima.

Finally, we discuss some future research directions. Significantly, it is optimistic that the positions of MAs can be adjusted freely within a given region. In practice, the motion control of the stepper motor is discrete with finite precision. Thus, the given region is quantized in several paces, leading to non-convex mixed integer programming (MIP). To tackle this intractable issue, the novel deep reinforcement learning (DRL)-based algorithms can be applied to optimize the resource allocation and offloading strategy in the MA-enhanced WP-MCE systems. Moreover, in MA-enhanced systems, it is impractical to acquire the instantaneous CSI due to the latency caused by the MA array movement. In this case, only statistical CSI is available, which may bring a great challenge to the MA positioning optimization.

REFERENCES

[1] L. Kong, J. Tan, J. Huang, G. Chen, S. Wang, X. Jin, P. Zeng, M. Khan, and S. K. Das, "Edge-computing-driven Internet of things: A survey," *ACM Comput. Surv.*, vol. 55, no. 8, pp. 1–41, Dec. 2022.

[2] S. Xu, J. Liu, N. Kato, and Y. Du, "Intelligent reflecting surface backscatter enabled multi-tier computing for 6G Internet of Things," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 2, pp. 320–333, Feb. 2023.

[3] Y. Siriwardhana, P. Porambage, M. Liyanage, and M. Ylianttila, "A survey on mobile augmented reality with 5G mobile edge computing: Architectures, applications, and technical aspects," *IEEE Commun. Surveys Tut.*, vol. 23, no. 2, pp. 1160–1192, Secondquarter 2021.

[4] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," *IEEE Commun. Surveys Tut.*, vol. 19, no. 4, pp. 2322–2358, Dec. 2017.

[5] H. Zhang, N. Shlezinger, F. Guidi, D. Dardari, M. F. Imani, and Y. C. Eldar, "Near-field wireless power transfer for 6G Internet of Everything mobile networks: Opportunities and challenges," *IEEE Commun. Mag.*, vol. 60, no. 3, pp. 12–18, Mar. 2022.

[6] Z. Zhang, H. Pang, A. Georgiadis, and C. Cecati, "Wireless power transfer: An overview," *IEEE Trans. Ind. Electron.*, vol. 66, no. 2, pp. 1044–1058, Feb. 2019.

[7] X. Wang, J. Li, Z. Ning, Q. Song, L. Guo, S. Guo, and M. S. Obaidat, "Wireless powered mobile edge computing networks: A survey," *ACM Comput. Surv.*, vol. 55, no. 13s, pp. 1–37, Jul. 2023.

[8] L. Zhu, W. Ma, and R. Zhang, "Movable antennas for wireless communication: Opportunities and challenges," *IEEE Commun. Mag.*, pp. 1–7, Oct. 2023.

[9] L. Zhu, W. Ma, and R. Zhang, "Modeling and performance analysis for movable antenna enabled wireless communications," *IEEE Trans. Wireless Commun.*, pp. 1–1, Nov. 2023.

[10] W. Ma, L. Zhu, and R. Zhang, "MIMO capacity characterization for movable antenna systems," *IEEE Trans. Wireless Commun.*, vol. 23, no. 4, pp. 3392–3407, Apr. 2024.

[11] Z. Cheng, N. Li, J. Zhu, X. She, C. Ouyang, and P. Chen, "Sum-rate maximization for movable antenna enabled multiuser communications," *arXiv preprint arXiv:2309.11135*, 2023.

[12] Y. Sun, H. Xu, C. Ouyang, and H. Yang, "Sum-rate optimization for RIS-aided multiuser communications with movable antenna," *arXiv preprint arXiv:2311.06501*, 2023.

[13] H. Teng, Z. Li, K. Cao, S. Long, S. Guo, and A. Liu, "Game theoretical task offloading for profit maximization in mobile edge computing," *IEEE Trans. Mob. Comput.*, vol. 22, no. 9, pp. 5313–5329, Sept. 2023.

[14] T. Bahreini, H. Badri, and D. Grosu, "Mechanisms for resource allocation and pricing in mobile edge computing systems," *IEEE Trans. Parallel Distrib. Syst.*, vol. 33, no. 3, pp. 667–682, Mar. 2022.

[15] W. Zhou, J. Xia, F. Zhou, L. Fan, X. Lei, A. Nallanathan, and G. K. Karagiannidis, "Profit maximization for cache-enabled vehicular mobile edge computing networks," *IEEE Trans. Veh. Technol.*, vol. 72, no. 10, pp. 13 793–13 798, Oct. 2023.

[16] A. Asghari and M. K. Sohrabi, "Multiobjective edge server placement in mobile-edge computing using a combination of multiagent deep Q-network and coral reefs optimization," *IEEE Internet Things J.*, vol. 9, no. 18, pp. 17 503–17 512, Sept. 2022.

[17] F. Guim, T. Metsch, H. Moustafa, T. Verrall, D. Carrera, N. Cadenelli, J. Chen, D. Doria, C. Ghadie, and R. G. Prats, "Autonomous lifecycle management for resource-efficient workload orchestration for green edge computing," *IEEE Trans. Green Commun. Netw.*, vol. 6, no. 1, pp. 571–582, Mar. 2022.

[18] S. Bi and Y. J. Zhang, "Computation rate maximization for wireless powered mobile-edge computing with binary computation offloading," *IEEE Trans. Wireless Commun.*, vol. 17, no. 6, pp. 4177–4190, Jun. 2018.

[19] S. Mao, N. Zhang, L. Liu, J. Wu, M. Dong, K. Ota, T. Liu, and D. Wu, "Computation rate maximization for intelligent reflecting surface enhanced wireless powered mobile edge computing networks," *IEEE Trans. Veh. Technol.*, vol. 70, no. 10, pp. 10 820–10 831, Oct. 2021.

[20] B. Li, F. Si, W. Zhao, and H. Zhang, "Wireless powered mobile edge computing with NOMA and user cooperation," *IEEE Trans. Veh. Technol.*, vol. 70, no. 2, pp. 1957–1961, Feb. 2021.

[21] W. Feng, J. Tang, N. Zhao, X. Zhang, X. Wang, K.-K. Wong, and J. A. Chambers, "Hybrid beamforming design and resource allocation for UAV-aided wireless-powered mobile edge computing networks with NOMA," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 11, pp. 3271–3286, Nov. 2021.

[22] M. Wu, Q. Song, L. Guo, and I. Lee, "Energy-efficient secure computation offloading in wireless powered mobile edge computing systems," *IEEE Trans. Veh. Technol.*, vol. 72, no. 5, pp. 6907–6912, May 2023.

[23] Y. Ye, L. Shi, X. Chu, D. Li, and G. Lu, "Delay minimization in wireless powered mobile edge computing with hybrid BackCom and AT," *IEEE Wireless Commun. Lett.*, vol. 10, no. 7, pp. 1532–1536, Jul. 2021.

[24] Y. Ye, L. Shi, X. Chu, R. Q. Hu, and G. Lu, "Resource allocation in backscatter-assisted wireless powered MEC networks with limited MEC computation capacity," *IEEE Trans. Wireless Commun.*, vol. 21, no. 12, pp. 10 678–10 694, Dec. 2022.

[25] L. Shi, Y. Ye, X. Chu, and G. Lu, "Computation bits maximization in a backscatter assisted wirelessly powered MEC network," *IEEE Commun. Lett.*, vol. 25, no. 2, pp. 528–532, Feb. 2021.

[26] J. Han, G. H. Lee, S. Park, and J. K. Choi, "Joint subcarrier and transmission power allocation in OFDMA-based WPT system for mobile-edge computing in IoT environment," *IEEE Internet Things J.*, vol. 9, no. 16, pp. 15 039–15 052, Aug. 2022.

[27] P. Chen, B. Lyu, S. Gong, H. Guo, J. Jiang, and Z. Yang, "Computational rate maximization for IRS-assisted full-duplex wireless-powered MEC systems," *IEEE Trans. Veh. Technol.*, vol. 73, no. 1, pp. 1191–1206, Jan. 2023.

[28] P. Chen, B. Lyu, Y. Liu, H. Guo, and Z. Yang, "Multi-IRS assisted wireless-powered mobile edge computing for Internet of Things," *IEEE Trans. Green Commun. Netw.*, vol. 7, no. 1, pp. 130–144, Mar. 2023.

[29] P. Chen, Y. Yang, J. Jiang, B. Lyu, Z. Yang, and A. Jamalipour, "Computational rate maximization for IRS-assisted multi-antenna WP-MEC systems with finite edge computing capability," *IEEE Internet Things J.*, vol. 11, no. 4, pp. 6607–6621, Feb. 2024.

[30] Z. Xiao, S. Cao, L. Zhu, Y. Liu, B. Ning, X.-G. Xia, and R. Zhang, "Channel estimation for movable antenna communication systems: A framework based on compressed sensing," *IEEE Trans. Wireless Commun.*, pp. 1–1, Apr. 2024.

[31] W. Ma, L. Zhu, and R. Zhang, "Compressed sensing based channel estimation for movable antenna communications," *IEEE Commun. Lett.*, vol. 27, no. 10, pp. 2747–2751, 2023.

[32] X. Chen, B. Feng, Y. Wu, D. W. Kwan Ng, and R. Schober, "Joint beamforming and antenna movement design for movable antenna systems based on statistical CSI," *in Proc. IEEE GLOBECOM*, pp. 4387–4392, Feb. 2023.

[33] W. Ma, L. Zhu, and R. Zhang, "Multi-beam forming with movable-antenna array," *IEEE Commun. Lett.*, vol. 28, no. 3, pp. 697–701, Mar. 2024.

[34] L. Zhu, W. Ma, and R. Zhang, "Movable-antenna array enhanced beamforming: Achieving full array gain with null steering," *IEEE Commun. Lett.*, vol. 27, no. 12, pp. 3340–3344, Dec. 2023.

[35] Z. Xiao, X. Pi, L. Zhu, X. Xia, and R. Zhang, "Multiuser communications with movable-antenna base station: Joint antenna positioning, receive combining, and power control," *arXiv preprint arXiv:2308.09512*, 2023.

[36] L. Zhu, W. Ma, B. Ning, and R. Zhang, "Movable-antenna enhanced multiuser communication via antenna position optimization," *IEEE Trans. Wireless Commun.*, pp. 1–1, Dec. 2023.

[37] G. Hu, Q. Wu, J. Ouyang, K. Xu, Y. Cai, and N. Al-Dhahir, "Movable-antenna array-enabled wireless communication with CoMP reception," *arXiv preprint arXiv:2311.11814*, 2023.

[38] H. Qin, W. Chen, Z. Li, Q. Wu, N. Cheng, and F. Chen, "Antenna positioning and beamforming design for movable-antenna enabled multi-user downlink communications," *arXiv preprint arXiv:2311.03046*, 2023.

[39] G. Hu, Q. Wu, K. Xu, J. Si, and N. Al-Dhahir, "Secure wireless communication via movable-antenna array," *IEEE Signal Proc. Let.*, vol. 31, pp. 516–520, Jan. 2024.

[40] J. Guo and X. Zhu, "An improved analytical model for RF-DC conversion efficiency in microwave rectifiers," *in Proc. IEEE MTT-S Int. Microw. Symp. Dig.*, pp. 1–3, Jun. 2012.

[41] T. Bai, C. Pan, H. Ren, Y. Deng, M. Elkashlan, and A. Nallanathan, "Resource allocation for intelligent reflecting surface aided wireless powered mobile edge computing in OFDM systems," *IEEE Trans. Wireless Commun.*, vol. 20, no. 8, pp. 5389–5407, Aug. 2021.

[42] D. Mishra and G. C. Alexandropoulos, "Transmit precoding and receive power splitting for harvested power maximization in MIMO SWIPT systems," *IEEE Trans. Green Commun. Netw.*, vol. 2, no. 3, pp. 774–786, Sept. 2018.

[43] Z. Li, W. Chen, Q. Wu, H. Cao, K. Wang, and J. Li, "Robust beamforming design and time allocation for IRS-assisted wireless powered communication networks," *IEEE Trans. Commun.*, vol. 70, no. 4, pp. 2838–2852, Apr. 2022.

[44] M. Hua and Q. Wu, "Throughput maximization for IRS-aided MIMO FD-WPCN with non-linear EH model," *IEEE J. Sel. Top. Signal Process.*, vol. 16, no. 5, pp. 918–932, Aug. 2022.

[45] Y. Wang, M. Sheng, X. Wang, L. Wang, and J. Li, "Mobile-edge computing: Partial computation offloading using dynamic voltage scaling," *IEEE Trans. Commun.*, vol. 64, no. 10, pp. 4268–4282, Oct. 2016.

[46] J. Chen, Z. Qin, Y. Liu, and J. Lu, "Particle swarm optimization with local search," *in Proc. ICNN&B*, pp. 481–484, 2005.

[47] M. Grant, S. Boyd, and Y. Ye, "CVX: Matlab software for disciplined convex programming," 2008.

[48] Z. Luo, W. Ma, A. M. So, Y. Ye, and S. Zhang, "Semidefinite relaxation of quadratic optimization problems," *IEEE Signal Process. Mag.*, vol. 27, no. 3, pp. 20–34, May 2010.

[49] S. Mao, L. Liu, N. Zhang, M. Dong, J. Zhao, J. Wu, and V. C. M. Leung, "Reconfigurable intelligent surface-assisted secure mobile edge computing networks," *IEEE Trans. Veh. Technol.*, vol. 71, no. 6, pp. 6647–6660, Jun. 2022.