Citation information: DOI10.1109/TCYB.2022.3168839, IEEE Transactions on Cybernetics

Combining Lyapunov Optimization with Evolutionary Transfer Optimization for Long-Term Energy Minimization in IRS-Aided Communications

Pei-Qiu Huang, Yong Wang, *Senior Member, IEEE*, Kezhi Wang, *Senior Member, IEEE*, and Qingfu Zhang, *Fellow, IEEE*

***Abstract*—This paper studies an intelligent reflecting surface (IRS)-aided communication system under the time-varying channels and stochastic data arrivals. In this system, we jointly optimize the phase shift coefficient and the transmit power in sequential time slots to maximize the long-term energy consumption for all mobile devices while ensuring queue stability. Due to the dynamic environment, it is challenging to ensure queue stability. In addition, making real-time decisions in each short time slot also needs to be considered. To this end, we propose a method (called LETO) that combines Lyapunov optimization with evolutionary transfer optimization to solve the above optimization problem. LETO first adopts Lyapunov optimization to decouple the long-term stochastic optimization problem into deterministic optimization problems in sequential time slots. As a result, it can ensure queue stability since the deterministic optimization problem in each time slot does not involve future information. After that, LETO develops an evolutionary transfer method to solve the optimization problem in each time slot. Specifically, we first define a metric to identify the optimization problems in past time slots similar to that in the current time slot, and then transfer their optimal solutions to construct a high-quality initial population in the current time slot. Since the evolutionary transfer optimization effectively accelerates the search, we can make real-time decisions in each short time slot. Experimental studies verify the effectiveness of LETO by comparison with other algorithms.**

***Index Terms*—Evolutionary algorithm, dynamic environment, evolutionary transfer optimization, Lyapunov optimization, intelligent reflecting surface**

I. Introduction

With the emergence of various innovative applications such as immersive virtual reality, autonomous systems, and industrial Internet of Things, the demand for high-data-rate and low-latency communications continues to grow [1]–[4]. Therefore, a lot of innovations for wireless communications have been proposed in recent years, such as massive multiple-input-and-multiple-output (MIMO) and millimeter wave (mmWave) communications. Although these technologies can significantly improve the spectral efficiency of wireless communication systems, they require high hardware costs and/or energy consumption [5].

Intelligent reflecting surface (IRS) [6], also known as reconfigurable intelligent surface, has been recently proposed as a promising technology to enhance the performance of wireless communication systems through intelligently reconfiguring the channel environment. Specifically, IRSs are two-dimensional meta-surfaces composed of a large array of passive reflecting elements, each of which can impose the required phase shift on the incoming signal. By carefully adjusting the phase shifts of the reflecting elements, the reflected signals can be reconfigured to propagate towards their desired directions, thereby improving the propagation environment [7]. Since IRSs do not require any sophisticated signal processing operations but simply rely on passive signal reflection, they greatly reduce the hardware cost and energy consumption. In addition, IRSs exhibit great flexibility and compatibility in practical deployment as they can be easily attached to building facades, ceilings, road signs, etc.

To explore the benefits of IRSs in wireless communication systems, several efforts have been devoted to system design and optimization for IRS-aided communication systems [8]. For instance, Wu and Zhang [6] studied an IRS-aided multiple-input single-output (MISO) communication system, where the downlink information from an access point (AP) to multiple users is transmitted with the help of an IRS. To minimize the total transmit power at the AP, a power minimization problem is formulated by jointly optimizing the active and passive beamforming. Xu *et al.* [9] designed a phase shift optimization method in an IRS-aided MISO downlink scenario without the assumption of knowing channel state information at the IRS. Souto *et al.* [10] employed particle swarm optimization to study beamforming optimization without channel state information. Yang *et al.* [11] considered an IRS-aided orthogonal frequency division multiplexing system. They then jointly optimized the transmit power allocation and the passive array reflection coefficient to maximize the achievable rate for the user. To compensate for the severe path loss in mmWave communication systems, Wang *et al.* [12] deployed multiple IRSs to enhance paths for mmWave signals in a MISO downlink mmWave communication system. In terms of IRS-aided MIMO communication systems, Zhang and Zhang [13] jointly optimized the reflection coefficient and the MIMO transmit covariance matrix to increase the capacity of systems. To mitigate inter-cell interference, Pan

P.-Q. Huang is with the School of Automation, Central South University, Changsha 410083, China, and also with the Department of Computer Science, City University of Hong Kong, Hong Kong (Email: pqhuang@csu.edu.cn).

Y. Wang is with the School of Automation, Central South University, Changsha 410083, China (Email: ywang@csu.edu.cn).

K. Wang is with the Department of Computer and Information Sciences, Northumbria University, Newcastle NE1 8ST, UK (Email: kezhi.wang@northumbria.ac.uk).

Q. Zhang is with the Department of Computer Science, City University of Hong Kong, Hong Kong (Email: qingfu.zhang@cityu.edu.hk).

et al. [14] deployed an IRS at the cell boundary of a multi-cell system to assist multi-user multi-cell MIMO downlink transmissions to cell-edge users. In addition, IRSs have been applied to other systems. Wu and Zhang [15] studied an IRS-aided simultaneous wireless information and power transfer system, where an IRS is deployed to improve the wireless power transfer efficiency for a group of energy harvesting receivers. Bai *et al.* [16] exploited an IRS-aided mobile edge computing system, where mobile devices (MDs) can offload a part of computational tasks to the edge computing node with the aid of an IRS. Peng *et al.* [17] adopted a genetic algorithm (GA) to optimize the phase shift coefficient in an IRS-aided multi-pair communication system.

Although the success of the above studies has been reported, they all focus on the short-term performance of IRS-aided communication systems. It is vital to study the long-term performance of IRS-aided communication systems under dynamic environments as it can provide an aggregate of short-term performance over a long period, which is of great economic and commercial significance [18]. Against the above background, this paper studies an IRS-aided uplink communication system, where MDs send data to an AP with the aid of an IRS under the time-varying channels and stochastic data arrivals. We aim to minimize the long-term energy consumption for all MDs. Note that, such long-term consideration introduces significant complexity to the system optimization as we need to take queue stability and real-time decision making into account. The main contributions of this paper are summarized as follows.

- We propose a method, named LETO, to jointly optimize the phase shift coefficient of the IRS and the transmit power of all MDs in sequential time slots to minimize the long-term energy consumption for all MDs while ensuring queue stability.
- To make decisions in each time slot without the assumption of knowing the future channels and data arrivals, LETO applies Lyapunov optimization to decouple the original long-term stochastic optimization problem into deterministic optimization problems in sequential time slots. Since no future information is involved, it can ensure queue stability by solving the deterministic optimization problem in each time slot.
- To satisfy real-time decision making in each short time slot, we adopt evolutionary transfer optimization (ETO) in LETO to solve the deterministic optimization problems sequentially. Specifically, we first define a metric to identify the optimization problems in past time slots that are similar to that in the current time slot. Then, the optimal solutions of these similar optimization problems are transferred to construct an initial population in the current time slot. In this way, we can accelerate the search and improve computational efficiency.
- Extensive experiments are conducted to evaluate the performance of LETO. The results show that LETO can improve the overall performance of the studied system in terms of energy consumption and queue stability.

The rest of this paper is organized as follows. Section II establishes the system model and formulates the energy minimization problem. In Section III, we describe the details of LETO. The experimental studies are given in Section IV. Finally, Section V concludes this paper.

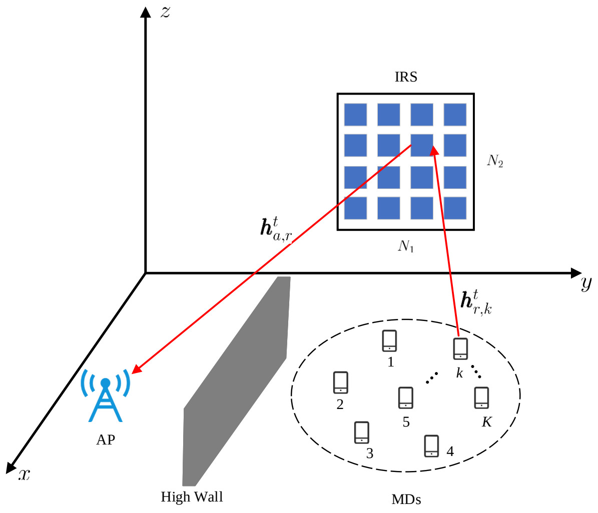


Fig. 1. An IRS-aided uplink communication system involving K MDs, an IRS, and an AP.

II. System Model and Problem Formulation

A. System Model

As shown in Fig. 1, we consider an IRS-aided uplink communication system, which consists of a set of K MDs (denoted as $\mathcal{K} = \{1, 2, \ldots, K\}$), an IRS, and an AP. Due to the obstacle of the high wall, there is no direct link from each MD to the AP. Thus, an IRS equipped with a set of N reflecting elements (denoted as $\mathcal{N} = \{1, 2, \ldots, N\}$) is deployed to assist the uplink communication from K MDs to the AP.

We assume that the system operates in a time-slotted structure, i.e., $t \in \mathcal{T} = \{1, 2, \ldots\}$ with equal duration τ. $\boldsymbol{h}_{r,k}^t \in \mathbb{C}^{N\times 1}$ and $\boldsymbol{h}_{a,r}^t \in \mathbb{C}^{N\times 1}$ represent channels of links from the kth MD to the IRS and from the IRS to the AP in the tth time slot, respectively. It is assumed that they remain near-constant within each time slot and can be estimated [16]. The phase shift coefficient of the IRS in the tth time slot is represented by $\boldsymbol{\theta}^t = [\theta_1^t, \theta_2^t, \ldots, \theta_N^t]^{\mathrm{T}}$. Note that, in this paper, the discrete phase shifts are considered as they are more cost-effective to be implemented than the continuous phase shifts [5], [19]. When each reflecting element of the IRS has X levels, the set of discrete phase-shift values is $\{0, \Delta\theta, \ldots, (X-1)\Delta\theta\}$, where $\Delta\theta = \frac{2\pi}{X}$. Then, the phase shift coefficient matrix of the IRS in the tth time slot is $\boldsymbol{\Theta}^t = \mathrm{diag}\{e^{j\theta_1^t}, e^{j\theta_2^t}, \ldots, e^{j\theta_N^t}\}$, where j represents the imaginary unit. In addition, we assume that the amplitude reflection coefficient is 1 for all reflecting elements [16].

We assume that the Rician fading channel model is used for links from the MD to the IRS and from the IRS to the AP [20]. Therefore, the channel of the link from the IRS to the AP in the tth time slot can be expressed as [21]

$$\boldsymbol{h}_{a,r}^t = PL_{a,r}^t \left(\sqrt{\frac{\varepsilon}{\varepsilon+1}} \boldsymbol{a}_r(\phi_r^t, \theta_r^t) + \sqrt{\frac{1}{\varepsilon+1}} \overline{\boldsymbol{h}}_{a,r}^t \right), \quad (1)$$

where $PL_{a,r}^t$ denotes the path-loss from the AP to the IRS in the tth time slot, ε denotes the Rician factor, $\boldsymbol{a}_r(\phi_r^t, \theta_r^t) \in \mathbb{C}^{N\times 1}$ is the array response of the IRS, ϕ_r^t (θ_r^t) denotes the azimuth (elevation) angle of departure for the link from the IRS to the AP in the tth time slot, and $\overline{\boldsymbol{h}_{a,r}^t}$ denotes the non-line-of-sight components in the tth time slot and its elements are chosen from $\mathcal{CN}(0,1)$ [1]. Similarly, $\boldsymbol{h}_{r,k}^t$ is defined as

$$\boldsymbol{h}_{r,k}^t = PL_{r,k}^t\left(\sqrt{\frac{\varepsilon}{\varepsilon+1}}\boldsymbol{a}_r(\phi_r^{t'}, \theta_r^{t'}) + \sqrt{\frac{1}{\varepsilon+1}}\overline{\boldsymbol{h}_{r,k}^t}\right), \quad (2)$$

where $PL_{r,k}^t$ denotes the path-loss from the kth MD to the IRS in the tth time slot, and $\phi_r^{t'}/\theta_r^{t'}$ denotes the azimuth/elevation angle of arrival for the link from the kth MD to the IRS in the tth time slot. According to [22] and [23], we have

$$\boldsymbol{a}_r(\phi_r^t, \theta_r^t) = \boldsymbol{a}_x(u) \otimes \boldsymbol{a}_y(v), \quad (3)$$

where $\otimes$ denotes the Kronecker product, $u = 2\pi d\cos(\theta_r^t)/\lambda$, $v = 2\pi d \sin(\theta_r^t)\cos(\phi_r^t)/\lambda$, d denotes the antenna spacing, and λ is the signal wavelength. In addition,

$$\boldsymbol{a}_x(u) = \left[1, e^{ju}, \ldots, e^{j(N_1-1)u}\right]^{\mathrm{T}}, \quad (4)$$

and

$$\boldsymbol{a}_y(v) = \left[1, e^{jv}, \ldots, e^{j(N_2-1)v}\right]^{\mathrm{T}}, \quad (5)$$

where N_1 and N_2 represent the numbers of reflecting elements in the horizontal and vertical directions of the IRS, respectively.

As a result, the channel of the link from the kth MD to the IRS to the AP in the tth time slot is

$$h_k^t = (\boldsymbol{h}_{a,r}^t)^{\mathrm{H}}\boldsymbol{\Theta}^t\boldsymbol{h}_{r,k}^t. \quad (6)$$

Remark 1: *We assume that the locations of MDs are fixed within each time slot, but they can be changed between different time slots. More importantly, their locations in the future time slots are unknown. Therefore, the future channels are not available as they are related to the future locations of MDs.*

In this paper, all MDs communicate simultaneously with the AP on different orthogonal sub-bands of equal size. Therefore, the data rate from the kth MD to the AP in the tth time slot can be given by

$$R_k^t = B\log_2\left(1 + \frac{p_k^t|h_k^t|^2}{\sigma^2}\right), \quad (7)$$

where B denotes the sub-band bandwidth, p_k^t denotes the transmit power of the kth MD in the tth time slot, and σ^2 denotes the variance of complex Gaussian channel noise.

Further, as shown in Fig. 2, we assume that the raw data with size $\boldsymbol{A}^t = [A_1^t, A_2^t, \ldots, A_K^t]^{\mathrm{T}}$ (in bits) arrives at the queues of MDs in the tth time slot. Besides, $A_k^t \in [A_{k,\min}^t, A_{k,\max}^t]$ follows an independent and identically uniformed distribution in different time slots with non-negative integer values, with $\mathbb{E}\{A_k^t\} = \lambda_k$ [24]. As stated in [25], A_k^t is dynamic and unpredictable. Then, let $\boldsymbol{Q}^t = [Q_1^t, Q_2^t, \ldots, Q_K^t]^{\mathrm{T}}$ denote the length of queues of all MDs at the beginning of the tth time slot. Therefore, the queue dynamic of the kth MD can be modeled as

$$Q_k^{t+1} = \max\{Q_k^t - D_k^t, 0\} + A_k^t, \quad (8)$$

where $D_k^t = R_k^t\tau$ represents the size of the data of the kth MD sent to the AP in the tth time slot. Note that, $Q_k^1 = 0$.

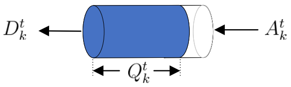


Fig. 2. Queue of the kth MD in the tth time slot.

B. Problem Formulation

In this paper, we aim to minimize the long-term energy consumption for all MDs by jointly optimizing the phase shift coefficient (i.e., $\boldsymbol{\theta} = \{\boldsymbol{\theta}^1, \boldsymbol{\theta}^2, \ldots, \boldsymbol{\theta}^t, \ldots\}$) and the transmit power (i.e., $\boldsymbol{p} = \{\boldsymbol{p}^1, \boldsymbol{p}^2, \ldots, \boldsymbol{p}^t, \ldots\}$) in sequential time slots while ensuring queue stability, where $\boldsymbol{p}^t = [p_1^t, p_2^t, \ldots, p_K^t]^{\mathrm{T}}$ denotes the transmit power of MDs in the tth time slot. As a result, the optimization problem is formulated as

$$\begin{aligned}
\min_{\boldsymbol{\theta},\boldsymbol{p}} \quad & \lim_{T\to\infty}\frac{1}{T}\sum_{t=1}^{T}\sum_{k=1}^{K} p_k^t\tau && (9a)\\
\text{s.t.} \quad & 0 \le p_k^t \le p_{k,\max}, \quad \forall k,t, && (9b)\\
& \theta_n^t \in \{0, \Delta\theta, \ldots, (X-1)\Delta\theta\}, \quad \forall n,t, && (9c)\\
& \lim_{T\to\infty}\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\{Q_k^t\} < \infty, \quad \forall k, && (9d)
\end{aligned}$$

where $p_{k,\max}$ denotes the maximal transmit power of the kth MD and (9d) ensures queue stability [26], [27].

Fig. 3 presents the flowchart of the long-term energy minimization problem in the studied IRS-aided uplink communication system. Owing to the time-varying channels (i.e., $\boldsymbol{h}_{a,r}^t$ and $\boldsymbol{h}_{r,k}^t$) and the stochastic data arrival (i.e., $\boldsymbol{A}^t$), we need to make decisions (i.e., $\boldsymbol{\theta}^{t*}$ and $\boldsymbol{p}^{t*}$) in sequential time slots. Due to the dynamic information in future time slots and the coupling between optimization problems in different time slots, myopic optimization in each time slot may compromise queue stability [28]. The reason is that MDs may not or less transmit data to the AP to save energy, which will cause the queue length to increase continuously. Under this condition, to effectively solve (9), there are two challenges:

- How to make decisions in each time slot for the long-term energy minimization while ensuring queue stability without the assumption of knowing the future channels and data arrivals?
- How to make real-time decisions in each short time slot?

Remark 2: *Although the studied optimization problem in this paper and the most common dynamic optimization problems [29]–[33] have the time-slotted structure, the latter focus on the short-term performance in each time slot.*

[1] $\mathcal{CN}(x,\sigma)$ represents the distribution of a complex Gaussian variable with mean x and covariance σ.

Citation information: DOI10.1109/TCYB.2022.3168839, IEEE Transactions on Cybernetics

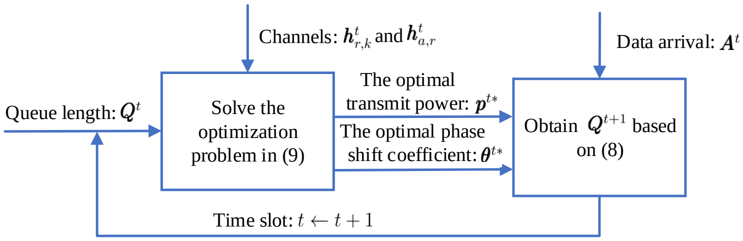

Channels: $\boldsymbol{h}_{r,k}^t$ and $\boldsymbol{h}_{a,r}^t$
Data arrival: $\boldsymbol{A}^t$
Queue length: $\boldsymbol{Q}^t$
Solve the optimization problem in (9)
The optimal transmit power: $\boldsymbol{p}^{t*}$
The optimal phase shift coefficient: $\boldsymbol{\theta}^{t*}$
Obtain $\boldsymbol{Q}^{t+1}$ based on (8)
Time slot: $t \leftarrow t+1$

Fig. 3. Flowchart of the long-term energy minimization problem.

III. Proposed Approach

To tackle the above two challenges, we propose a method (called LETO), which combines Lyapunov optimization with ETO to solve (9).

A. Lyapunov Optimization-Based Problem Decoupling

Regarding the first challenge, LETO applies Lyapunov optimization to decouple the long-term stochastic optimization problem in (9) into deterministic optimization problems in sequential time slots. Lyapunov optimization is particularly suitable for optimizing the long-term performance in stochastic network systems. By using it, system optimization can only depend on current information [34], [35].

We first define Lyapunov function $L(\boldsymbol{Q}^t)$ and Lyapunov drift $\Delta\boldsymbol{Q}^t$ as [24]

$$L(\boldsymbol{Q}^t) = 0.5\sum_{k=1}^{K}(Q_k^t)^2, \tag{10}$$

and

$$\Delta\boldsymbol{Q}^t = \mathbb{E}\left\{L(\boldsymbol{Q}^{t+1}) - L(\boldsymbol{Q}^t)|\boldsymbol{Q}^t\right\}, \tag{11}$$

where $L(\boldsymbol{Q}^t)$ represents a scalar measure of the length of queues, and $\Delta\boldsymbol{Q}^t$ represents the expected change in the Lyapunov function from one time slot to the next. To stabilize the queues, we need to make $\Delta\boldsymbol{Q}^t$ as small as possible [36].

To minimize the long-term energy consumption for all MDs while stabilizing the queue, we then minimize the following drift-plus-penalty function in each time slot [34]:

$$\Delta\boldsymbol{Q}^t + V\sum_{k=1}^{K}\mathbb{E}\left\{p_k^t\tau|\boldsymbol{Q}^t\right\}, \tag{12}$$

where V is a predefined tradeoff between energy consumption and queue stability.

Since it is not a trivial task to minimize the drift-plus-penalty function, similar to [27], [37], and [38], we instead minimize the upper bound of the drift-plus-penalty function in each time slot, which can be obtained as follows. First, we have

$$\begin{aligned}\Delta\boldsymbol{Q}^t =& \mathbb{E}\left\{L(\boldsymbol{Q}^{t+1}) - L(\boldsymbol{Q}^t)|\boldsymbol{Q}^t\right\}\\ =& 0.5\sum_{k=1}^{K}\mathbb{E}\left\{(Q_k^{t+1})^2 - (Q_k^t)^2\big|\boldsymbol{Q}^t\right\}\\ =& 0.5\sum_{k=1}^{K}\mathbb{E}\left\{\{\max\{Q_k^t - D_k^t, 0\} + A_k^t\}^2 - (Q_k^t)^2\big|\boldsymbol{Q}^t\right\}\\ \leq& \sum_{k=1}^{K}Q_k^t\mathbb{E}\left\{A_k^t - R_k^t\tau\big|\boldsymbol{Q}^t\right\} + 0.5\sum_{k=1}^{K}\mathbb{E}\left\{(A_k^t)^2 + (R_k^t\tau)^2\right\}\\ \leq& U + \sum_{k=1}^{K}Q_k^t\mathbb{E}\left\{A_k^t - R_k^t\tau\big|\boldsymbol{Q}^t\right\}.\end{aligned} \tag{13}$$

Herein, U is a constant obtained as

$$\begin{aligned}&0.5\sum_{k=1}^{K}\mathbb{E}\left\{(A_k^t)^2 + (R_k^t\tau)^2\right\}\\ \leq& 0.5\sum_{k=1}^{K}\left\{(A_{k,\max}^t)^2 + (R_{k,\max}^t\tau)^2\right\} \triangleq U,\end{aligned} \tag{14}$$

where $R_{k,\max}^t = B\log_2\left(1 + \frac{p_{k,\max}|h_k^t|^2}{\sigma^2}\right)$ is the maximum data rate from the kth MD to the AP in tth time slot.

Substituting (13) into (12), we have

$$\begin{aligned}&\Delta\boldsymbol{Q}^t + V\sum_{k=1}^{K}\mathbb{E}\left\{p_k^t\tau|\boldsymbol{Q}^t\right\}\\ \leq& U + \sum_{k=1}^{K}Q_k^t\mathbb{E}\left\{A_k^t - R_k^t\tau|\boldsymbol{Q}^t\right\} + V\sum_{k=1}^{K}\mathbb{E}\left\{p_k^t\tau|\boldsymbol{Q}^t\right\}\\ =& U + \sum_{k=1}^{K}Q_k^t\lambda_i - \sum_{k=1}^{K}Q_k^t\mathbb{E}\left\{R_k^t\tau|\boldsymbol{Q}^t\right\} + V\sum_{k=1}^{K}\mathbb{E}\left\{p_k^t\tau|\boldsymbol{Q}^t\right\}.\end{aligned} \tag{15}$$

Note that, the first two terms on the right-hand side of (15) can be removed as they are not related to the decision variables (i.e., $\boldsymbol{\theta}^t$ and $\boldsymbol{p}^t$). According to opportunistic expectation minimization [24], we can obtain the following optimization problem in each time slot as:

$$\begin{aligned}\min_{\boldsymbol{\theta}^t,\boldsymbol{p}^t}\;& \sum_{k=1}^{K}\left(-Q_k^tR_k^t\tau + Vp_k^t\tau\right)\\ \text{s.t.}\;\;& 0 \leq p_k^t \leq p_{k,\max}, \quad \forall k, t,\\ & \theta_n^t \in \{0, \Delta\theta, \ldots, (X-1)\Delta\theta\}, \quad \forall n, t.\end{aligned} \tag{16}$$

Algorithm 1 Overall process of DE for solving (21)

```
for $t = 1, 2, \ldots, t_{\max}$ do
    $g = 0$; // $g$ denotes the generation number
    Randomly generate $NP$ solutions to form an initial population (denoted as $\mathcal{P}_g^t$), where each solution represents the phase shift coefficient in the $t$th time slot (i.e., $\boldsymbol{\theta}^t$);
    Evaluate each solution in $\mathcal{P}_g^t$ according to (21);
    while $g < g_{\max}$ do
        Implement the crossover and mutation operators of DE on $\mathcal{P}_g^t$ to generate an offspring population (denoted as $\mathcal{Q}_g^t$);
        Evaluate each solution in $\mathcal{Q}_g^t$ according to (21);
        Perform the selection operator of DE to produce $\mathcal{P}_{g+1}^t$ from $\mathcal{P}_g^t$ and $\mathcal{Q}_g^t$;
        $g \leftarrow g + 1$;
    end while
    Obtain the optimal phase shift coefficient (denoted as $\boldsymbol{\theta}^{t*}$) from $\mathcal{P}_g^t$;
    Calculate the optimal transmit power (denoted as $\boldsymbol{p}^{t*}$) according to (20);
    Use $\boldsymbol{p}^{t*}$, $\boldsymbol{\theta}^{t*}$, $\boldsymbol{Q}^t$, and $\boldsymbol{A}^t$ to obtain $\boldsymbol{Q}^{t+1}$ according to (8);
    $t \leftarrow t + 1$;
end for
```

Then, differentiating $F = \sum_{k=1}^{K} \left(-Q_k^t R_k^t \tau + V p_k^t \tau\right)$ by p_k^t, we have

$$\frac{dF}{dp_k^t} = \frac{-Q_k^t \tau B |h_k^t|^2}{ln2(\sigma^2 + p_k^t |h_k^t|^2)} + V\tau. \tag{17}$$

The local minimizer (denoted as $\bar{p}_k^t$) can be obtained from $\frac{dF}{dp_k^t} = 0$ by

$$\bar{p}_k^t = \frac{Q_k^t B}{V ln2} - \frac{\sigma^2}{|h_k^t|^2}. \tag{18}$$

In addition, since we have

$$\frac{d^2 F}{d(p_k^t)^2} = \frac{Q_k^t \tau B \sigma^2 |h_k^t|^4}{ln2(\sigma^2 + p_k^t |h_k^t|^2)^2} > 0, \tag{19}$$

$\bar{p}_k^t$ is the global minimizer of F. Considering the bound constraint of p_k^t, the optimal p_k^t is obtained by

$$p_k^{t*} = \min\{\max\{0, \bar{p}_k^t\}, p_{k,\max}\}. \tag{20}$$

Thus, (16) can be simplified to the following optimization problem in each time slot:

$$\begin{aligned} &\min_{\boldsymbol{\theta}^t} \sum_{k=1}^{K} \left(-Q_k^t R_k^t \tau + V p_k^{t*} \tau\right) \\ &\text{s.t.} \quad \theta_n^t \in \{0, \Delta\theta, \ldots, (X-1)\Delta\theta\}, \quad \forall n, t. \end{aligned} \tag{21}$$

It can be seen that the optimization problem in (21) does not involve future channels and data arrivals. Moreover, we also consider queue stability in the objective function. As a result, we can make decisions in each time slot for the long-term energy minimization while ensuring queue stability without the assumption of knowing the future channels and data arrivals.

B. ETO-Based Problem Solving

Due to the discreteness of the phase shift coefficient, it is generally hard to solve the optimization problem in (21). Although deterministic optimization algorithms, such as the branch-and-bound method, can obtain the optimal solution of (21), they suffer from high computational complexity [39]. In contrast, evolutionary algorithms (EAs) [10], [17], [40], [41], such as GA, differential evolution (DE), and particle swarm optimization provide higher efficiency for solving complex optimization problems [42].

Algorithm 1 presents the overall process of DE to solve (21) [2], which sequentially solves the optimization problems in $t_{\max}$ time slots. The process of DE to solve the optimization problem in the tth time slot is introduced as follows. In the initialization, we first randomly generate NP solutions to form an initial population (denoted as $\mathcal{P}_g^t$), where each solution represents the phase shift coefficient in the tth time slot (i.e., $\boldsymbol{\theta}^t$). Then, we evaluate each solution in $\mathcal{P}_g^t$ based on (21), During the evolution, $\mathcal{Q}_g^t$ is generated from $\mathcal{P}_g^t$ by using the crossover and mutation operators of DE. Subsequently, each solution in $\mathcal{Q}_g^t$ is evaluated based on (21). Finally, the selection operator of DE is performed to obtain the next-generation population (denoted as $\mathcal{P}_{g+1}^t$) based on $\mathcal{P}_g^t$ and $\mathcal{Q}_g^t$. The above procedure is repeated until the stopping criterion is met, i.e., the maximal generation (denoted as $g_{\max}$) is reached. Afterward, we obtain the optimal phase shift coefficient (denoted as $\boldsymbol{\theta}^{t*}$) and the optimal transmit power (denoted as $\boldsymbol{p}^{t*}$). After that, $\boldsymbol{Q}^{t+1}$ is updated by using $\boldsymbol{p}^{t*}$, $\boldsymbol{\theta}^{t*}$, $\boldsymbol{Q}^t$, and $\boldsymbol{A}^t$ according to (8).

It can be seen that DE starts the search process from scratch in each time slot. Therefore, it is difficult to satisfy real-time decision making in each short time slot since each run usually requires many generations [43]. In practice, the optimization problems in different time slots share some similarities and rarely exist in isolation. Using the knowledge gained from past time slots may reduce the computational cost and improve real-time performance in future time slots. Based on this consideration, LETO adopts ETO to solve (21) in each time slot. ETO is a paradigm that integrates EAs and the experience gained in past related optimization problems to achieve better optimization efficiency and performance [44]. It has been applied to various optimization problems, such as multitask optimization problems [45]–[47], dynamic optimization problems [48]–[50], and complex practical optimization problems [51], [52].

When using ETO, LETO needs to address the following three issues: when, what, and how to transfer [53].

- As we all know, a high-quality initial population in EAs can speed up the evolutionary process. Along this line, in the initialization of each time slot, we utilize the experience obtained from the optimization problems in past time slots to generate an initial population for solving the optimization problem in the current time slot, which corresponds to the first issue.
- For the second issue, we transfer the optimal solutions of the optimization problems in past time slots. Note, however, that to avoid performance degradation caused by negative transfer, the optimization problems in past time slots should be similar to that in the current time slot. Therefore, a question which naturally arises is: how to measure their similarity. Indeed, the optimal phase shift coefficient is usually related to the locations of MDs [20]. Therefore, we design a metric to measure the similarity

[2] Other EAs can also be used as the search algorithm.

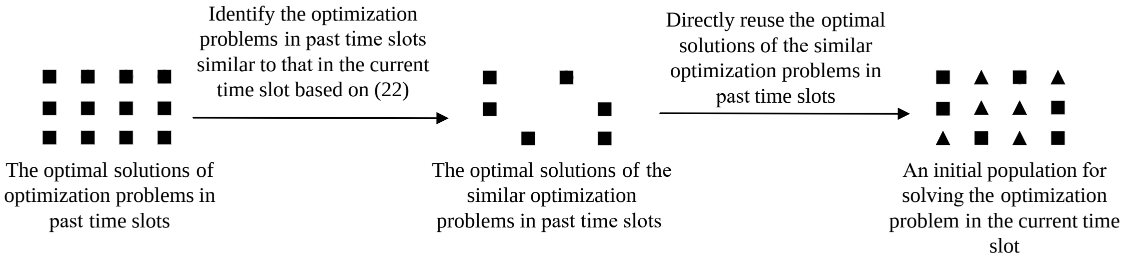

Identify the optimization problems in past time slots similar to that in the current time slot based on (22)
Directly reuse the optimal solutions of the similar optimization problems in past time slots
The optimal solutions of optimization problems in past time slots
The optimal solutions of the similar optimization problems in past time slots
An initial population for solving the optimization problem in the current time slot

Fig. 4. Overall process of LETO to generate an initial population in the current time slot, where ■ represents the optimal solution of the optimization problem in past time slots and ▲ represents the randomly generated solution.

between the optimization problems in different time slots based on the locations of MDs:

$$m(\mathcal{L}^{t_a}, \mathcal{L}^{t_b}) = \min\left\{\frac{1}{K}\sum_{k=1}^{K} d_k^{t_a,t_b}, \frac{1}{K}\sum_{k=1}^{K} d_k^{t_b,t_a}\right\} \tag{22}$$

where $\mathcal{L}^{t_a} = \{\boldsymbol{l}_1^{t_a}, \boldsymbol{l}_2^{t_a}, \ldots, \boldsymbol{l}_K^{t_a}\}$ and $\mathcal{L}^{t_b} = \{\boldsymbol{l}_1^{t_b}, \boldsymbol{l}_2^{t_b}, \ldots, \boldsymbol{l}_K^{t_b}\}$ represent the set of locations of all MDs in the t_ath and t_bth time slots, respectively, $\boldsymbol{l}_k^{t_a}$ and $\boldsymbol{l}_k^{t_b}$ represent the location of the kth MD in the t_ath and t_bth time slots, respectively, and $d_k^{t_a,t_b}$ denotes the Euclidean distance between $\boldsymbol{l}_k^{t_a}$ and the closest location in $\mathcal{L}^{t_b}$. Therefore, $\frac{1}{K}\sum_{k=1}^{K} d_k^{t_a,t_b}$ is used to calculate the average Euclidean distance between the locations of all MDs in the t_ath time slot and their closest locations in $\mathcal{L}^{t_b}$, and $\frac{1}{K}\sum_{k=1}^{K} d_k^{t_b,t_a}$ is used to calculate the average Euclidean distance between the locations of all MDs in the t_bth time slot and their closest locations in $\mathcal{L}^{t_a}$. The smaller the value of m, the higher the similarity. By comparing the value of m, we can identify the optimization problems in past time slots which are similar to that in the current time slot. Note that, to improve the real-time performance, for the optimization problem in the tth slot, we only search the similar optimization problems between the $(t-1-tlast)$th to the $(t-1)$th time slots.

- With respect to the last issue, several novel methods have been devised in ETO, such as autoencoding [50], [54], transfer component analysis [48], and neural network-based information transfer [55]. However, transfer component analysis and neural network-based information transfer may require a long training time [48], [55], and autoencoding is usually used to model the linear relationship among different optimization problems [50], [54]. As a result, they are difficult to use in this paper due to the highly complex correlation between the optimization problems in different time slots and the requirements of real-time decision making. To this end, we directly reuse the optimal solutions of similar optimization problems in past time slots to construct a high-quality initial population for solving the optimization problem in the current time slot. Note that, at most $NP/2$ transferred solutions are employed while the remaining solutions are randomly generated to maintain the initial population's diversity.

Fig. 4 shows the overall process of LETO to generate an initial population in the current time slot. The overall process of LETO for solving (21) is presented in Algorithm 2. It is similar to Algorithm 1 except for one modification. That is, we directly reuse the optimal solutions of the similar optimization problems as a part of the initial solutions of $\mathcal{P}_g^t$, as shown in line 16 of Algorithm 2.

Algorithm 2 Overall process of LETO for solving (21)

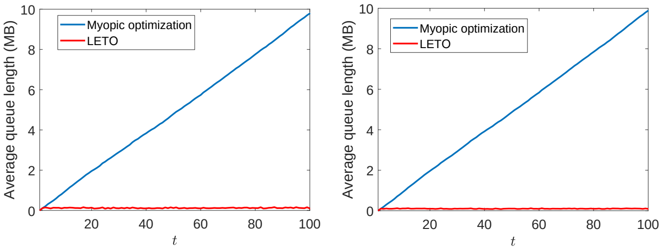
```
1: $t = 1$ and $g = 0$; // $g$ denotes the generation number
2: Randomly generate an initial population (denoted as $\mathcal{P}_g^t$);
3: Evaluate each solution in $\mathcal{P}_g^t$ according to (21);
4: while $g < g_{\max}$ do
5:     Implement the crossover and mutation operators of DE on $\mathcal{P}_g^t$ to
       generate an offspring population (denoted as $\mathcal{Q}_g^t$);
6:     Evaluate each solution in $\mathcal{Q}_g^t$ according to (21);
7:     Perform the selection operator of DE to produce $\mathcal{P}_{g+1}^t$ from $\mathcal{P}_g^t$ and
       $\mathcal{Q}_g^t$;
8:     $g \leftarrow g + 1$;
9: end while
10: Obtain the optimal phase shift coefficient (denoted as $\boldsymbol{\theta}^{t*}$) from $\mathcal{P}_g^t$;
11: Calculate the optimal transmit power (denoted as $\boldsymbol{p}^{t*}$) based on (20);
12: Use $\boldsymbol{p}^{t*}$, $\boldsymbol{\theta}^{t*}$, $\boldsymbol{Q}^t$, and $\boldsymbol{A}^t$ to obtain $\boldsymbol{Q}^{t+1}$ according to (8);
13: $t \leftarrow t + 1$;
14: while $t \leq t_{\max}$ do
15:     $g = 0$;
16:     Directly reuse the optimal solutions of the similar optimization prob-
        lems in past time slots as a part of the initial solutions of $\mathcal{P}_g^t$;
17:     Randomly generate the remaining solutions of $\mathcal{P}_g^t$;
18:     Evaluate each solution in $\mathcal{P}_g^t$ according to (21);
19:     while $g < g_{\max}$ do
20:         Implement the crossover and mutation operators of DE on $\mathcal{P}_g^t$ to
            generate $\mathcal{Q}_g^t$;
21:         Evaluate each solution in $\mathcal{Q}_g^t$ according to (21);
22:         Perform the selection operator of DE to produce $\mathcal{P}_{g+1}^t$ from $\mathcal{P}_g^t$
            and $\mathcal{Q}_g^t$;
23:         $g \leftarrow g + 1$;
24:     end while
25:     Obtain $\boldsymbol{\theta}^{t*}$ from $\mathcal{P}_g^t$;
26:     Calculate $\boldsymbol{p}^{t*}$ based on (20);
27:     Use $\boldsymbol{p}^{t*}$, $\boldsymbol{\theta}^{t*}$, $\boldsymbol{Q}^t$, and $\boldsymbol{A}^t$ to obtain $\boldsymbol{Q}^{t+1}$ according to (8);
28:     $t \leftarrow t + 1$;
29: end while
```

Remark 3: *Although reusing the optimal solutions of similar optimization problems in past time slots is a simple transfer method, its effectiveness has been investigated in experimental studies by comparison with autoencoding.*

IV. Experimental Studies

A. Parameter Settings

To evaluate the performance of LETO, we studied an IRS-aided uplink communication system, in which all MDs move at random speeds between 0 and 5 m/s on a circle centered at



(a) (b)

Fig. 5. Dynamics of the average queue length of all MDs obtained by LETO and myopic optimization. (a) $N = 25$ and $K = 30$. (b) $N = 100$ and $K = 50$.

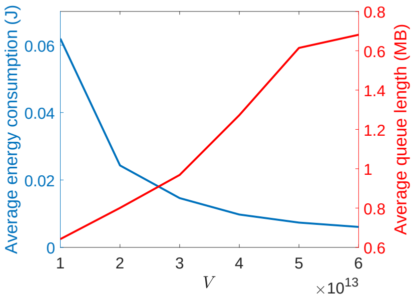
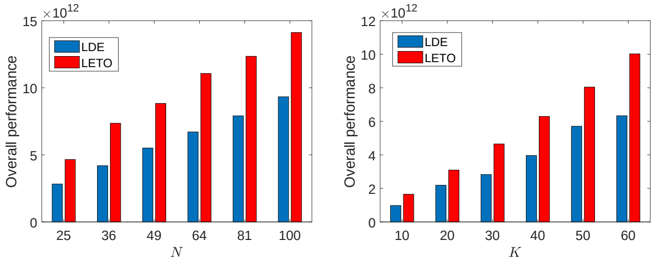


Fig. 6. Results of LETO with six different values of V when $N = 25$ and $K = 30$.



(a) (b)

Fig. 7. Overall performance of LETO and LDE in terms of energy consumption and queue stability. (a) Different numbers of reflecting elements when $K = 30$. (b) Different numbers of MDs when $N = 25$.

[100, 100, 0] m with a radius of 80 m, and the IRS and the AP are located at [0, 100, 2] m and [100, 1, 1] m, respectively. A_k^t obeys the uniform distribution, whose range is between [1, 200] KB. As for the channels, the path loss in dB is expressed as [16]

$$PL^t = PL_0 - 10\beta \log(\frac{d^t}{d_0}) \tag{23}$$

where $PL_0 = 30$ dB is the path loss at the reference distance (i.e., $d_0 = 1$ m), $\beta = 2.2$ represents the path loss exponent, and d^t denotes the distance from the transmitter to the receiver in the tth time slot. In addition, the duration of each time slot was set to $\tau = 1$ s, the number of levels of each reflect element was set to $X = 16$, the sub-band bandwidth was set to $B = 1$ MHz, the maximal transmit power was set to $p_{k,\text{max}} = 0.1$ W, the number of time slots was set to $t_{\text{max}} = 100$, and the tradeoff between energy consumption and queue stability was set to $V = 1 \times 10^{13}$.

In LETO, the "DE/best/1" mutation operator and the binomial crossover operator [56] were adopted. The population size was set to $NP = 30$, and the crossover control parameter and the scaling factor of DE were set to $CR = 0.9$ and F =0.9, respectively. The maximal generation of each time slot was set to a small number (i.e., g_{max} =10) since LETO has a high convergence speed. The number of optimization problems in past time slots was set to $tlast = 30$. Besides, 20 independent runs were implemented on LETO. We implemented all the experiments in MATLAB and tested them on a computer with Intel Core i7-6700 CPU @3.40 GHz and 16 GB of RAM.

B. Effectiveness of Lyapunov Optimization in LETO

To ensure queue stability, we adopted Lyapunov optimization to decouple the original long-term stochastic problem into deterministic optimization problems in sequential time slots. To verify its effectiveness, we compared the performance of LETO with that of myopic optimization. Myopic optimization removed Lyapunov optimization from LETO to minimize the long-term energy consumption for all MDs in each time slot while ignoring queue stability. Fig. 5 presents the dynamics of the average queue length of all MDs obtained by LETO and myopic optimization. It can be observed that the average queue length of all MDs obtained by myopic optimization continues to increase over time slots while LETO can keep the average queue length stability, indicating that LETO can achieve long-term queue stability; thus, the effectiveness of Lyapunov optimization is verified.

Lyapunov optimization introduces an important parameter V in (12) to balance queue stability and energy consumption for all MDs. One may be interested in its effect on the performance of LETO. We tested LETO with six different values of V: $V = 1 \times 10^{13}, 2 \times 10^{13}, \ldots, 6 \times 10^{13}$ on an instance with $N = 25$ and $K = 30$. Two performance indicators were employed here: (1) the average energy consumption for all MDs: $\frac{1}{K{*}t_{\text{max}}} \sum_{t=1}^{t_{\text{max}}} \sum_{k=1}^{K} p_k^t \tau$, and (2) the average queue length of all MDs: $\frac{1}{K{*}t_{\text{max}}} \sum_{t=1}^{t_{\text{max}}} \sum_{k=1}^{K} Q_k^t$. Fig. 6 presents the results obtained by LETO with six different values of V. It can be seen from Fig. 6 that as the values of V increase, the average energy consumption for all MDs continues to decrease, while the average queue length of all MDs continues to increase. This is because when the values of V increase, LETO prefers the second term in (12), which leads to lower energy consumption and longer queues.

C. Effectiveness of ETO in LETO

To make decisions quickly, LETO made use of ETO to solve the optimization problem in (21) in each time slot. To study the effectiveness of ETO, ETO in LETO was replaced with DE, resulting in a variant of LETO (called LDE). Then, we compared their overall performance in terms of energy consumption and queue stability: $\frac{1}{t_{\text{max}}} \sum_{t=1}^{t_{\text{max}}} \sum_{k=1}^{K} (Q_k^t R_k^t \tau - V p_k^t \tau)$. Note here that a larger value indicates better overall performance. As shown in Fig. 7(a), we compared the overall performance of LDE and LETO under different numbers of reflecting elements

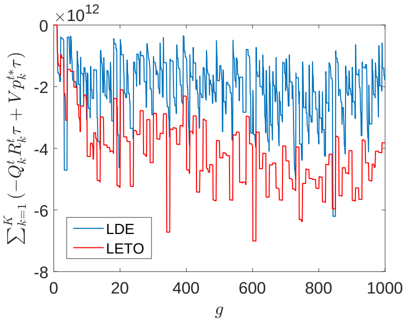
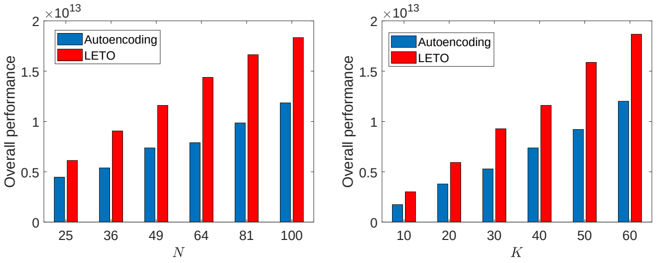


Fig. 8. Dynamics of the objective values of LETO and LDE when $N = 25$ and $K = 30$, where the maximum value of g is equal to $g_{\max}{*}t_{\max}$ (i.e., 1,000).

TABLE I
AVERAGE RUNTIME OF LETO TO MAKE DECISIONS IN EACH TIME SLOT.

N	25	36	49	64	81	100
T(s)	0.014	0.016	0.016	0.017	0.018	0.018



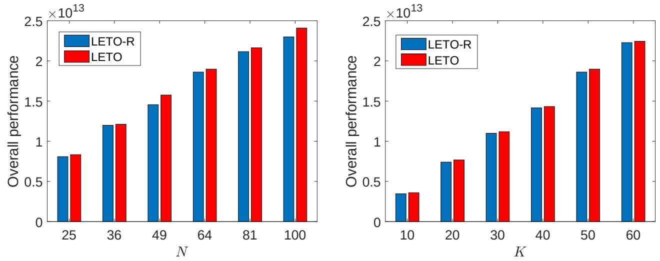
(a)

(b)

Fig. 9. Overall performance of LETO and autoencoding in terms of energy consumption and queue stability. (a) Different numbers of reflecting elements when $K = 40$. (b) Different numbers of MDs when $N = 49$.

when $K = 30$. We can observe that LETO performs better than LDE on each instance. In addition, the advantage of LETO consistently increases as the number of reflecting elements increases. To be specific, their gap is about 2×10^{12} when $N = 25$, but when $N = 100$, their gap reaches up to 5×10^{12}. Moreover, we compared LDE and LETO under different numbers of MDs when $N = 25$ in Fig. 7(b). From Fig. 7(b), LETO provides an increasing advantage over LDE as the number of MDs grows. Fig. 8 presents the dynamics of the objective values of LETO and LDE when $N = 25$ and $K = 30$. As shown in Fig. 8, the main advantage of LETO can be attributed to the fact that it starts from some good initial solutions instead of totally randomly generated solutions. In addition, we recorded the average runtime of LETO to make decisions in each time slot. As shown in Table I, the average runtime of LETO to make decisions in each time slot is much less than the duration of each time slot (i.e., $\tau = 1$ s), which means that LETO can meet the requirements of real-time decision making.

In order to utilize the previous experience, LETO directly reused the optimal solutions of the optimization problems in past time slots. As stated in Section III-B, many methods have been proposed, such as autoencoding, transfer component analysis, and neural network-based information transfer. In this subsection, we compared the performance of LETO and autoencoding. Autoencoding adopts a single-layer denoising autoencoder to transfer the optimal solutions in past time slots to the initial solutions in the current time slot [50]. Fig. 9 presents the overall performance of LETO and autoencoding in terms of energy consumption and queue stability. As shown in Fig. 9, LETO outperforms autoencoding on all instances. This phenomenon can be explained as follows. In general, autoencoding models the linear relationship among different optimization problems. However, the studied optimization problems between different time slots have highly complex nonlinear relationships. Besides, in the studied optimization problems, the optimal solutions of the similar optimization problems may also be similar; therefore, it is effective to directly reuse the optimal solutions of the optimization problems in past time slots.



(a)

(b)

Fig. 10. Overall performance of LETO and LETO-R in terms of energy consumption and queue stability. (a) Different numbers of reflecting elements when $K = 50$. (b) Different numbers of MDs when $N = 64$.

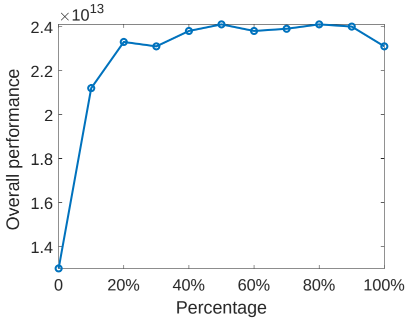


Fig. 11. Overall performance of LETO with different ratios of transferred solutions when $K = 50$ and $N = 100$.

D. Discussions

1) Effectiveness of the Similarity Metric: A similarity metric was proposed in LETO to identify the optimization problems in past time slots similar to that in the current time slot. To verify the effectiveness of the proposed metric, an additional experiment was performed, in which the similar optimization problems in past time slots were randomly selected, and the resultant algorithm was recorded as LETO-R. As shown in Fig. 10, the overall performance of LETO is better than LETO-R in terms of energy consumption and queue

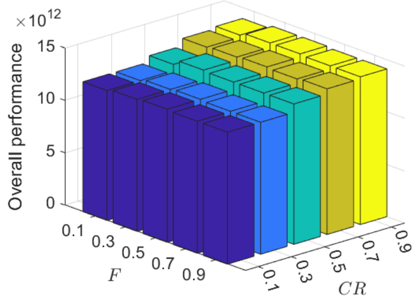
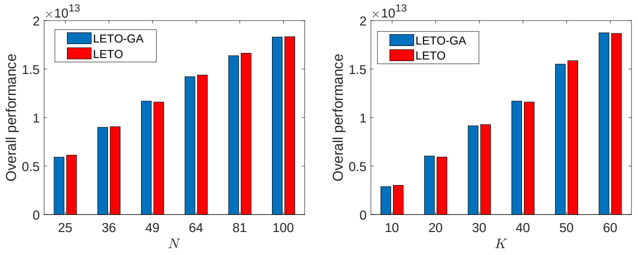


Fig. 12. Overall performance of LETO with 25 different combinations of F and CR when $K = 30$ and $N = 100$.



(a)

(b)

Fig. 13. Overall performance of LETO and LETO-GA in terms of energy consumption and queue stability. (a) Different numbers of reflecting elements when $K = 40$. (b) Different numbers of MDs when $N = 49$.

stability. The above results suggest that the proposed similarity metric can effectively search the similar optimization problems in past time slots.

2) Effect of the Ratio of Transferred Solutions: In LETO, some initial solutions in the current time slot were transferred from the optimal solutions in past time slots. To study the effect of the ratio of the transferred solutions in the initial population, we tested 11 different cases: 0, 10%, …, 100%. Fig. 11 presents the overall performance of LETO in terms of energy consumption and queue stability when $K = 50$ and $N = 100$. It can be seen from Fig. 11 that LETO performs well when the ratio of the transferred solutions in the initial population is between 40% and 90%, and performs the worst when no solution is transferred.

3) Sensitivities of F and CR to the Performance of LETO: We tested LETO with 25 combinations of F and CR to investigate their sensitivity. Specifically, we selected F and CR from two sets $\{0.1, 0.3, 0.5, 0.7, 0.9\}$ and $\{0.1, 0.3, 0.5, 0.7, 0.9\}$, respectively. Fig. 12 shows the overall performance obtained by LETO with 25 combinations of F and CR when $K = 30$ and $N = 100$. As shown in Fig. 12, LETO performs better as the value of CR increases, and its performance remains almost the same when $F \geq 0.3$.

4) Effect of the Search Engine: In this subsection, we studied the effect of the search engine on the performance of LETO by replacing DE with GA. The resultant variant is called LETO-GA. As shown in Fig. 13, there is no significant performance difference between LETO and LETO-GA, which means that LETO is not sensitive to the search engine.

V. Conclusion

In this paper, we studied an IRS-aided uplink communication system. To minimize the long-term energy consumption for all MDs in the studied system, a method combining Lyapunov optimization with ETO, called LETO, was proposed to jointly optimize the phase shift coefficient and the transmit power in sequential time slots while ensuring queue stability. LETO first decomposed the long-term stochastic optimization problem into deterministic optimization problems in sequential time slots. To quickly solve the optimization problem in each time slot, we then transferred the optimal solutions of the similar optimization problems in past time slots to the initial solutions in the current time slot. LETO was applied to various instances with different numbers of MDs and reflecting elements. The results demonstrated the effectiveness of LETO.

References

[1] C. Pan, H. Ren, K. Wang, J. F. Kolb, M. Elkashlan, M. Chen, M. Di Renzo, Y. Hao, J. Wang, A. L. Swindlehurst, X. You, and L. Hanzo, "Reconfigurable intelligent surfaces for 6g systems: Principles, applications, and research directions," *IEEE Communications Magazine*, vol. 59, no. 6, pp. 14–20, 2021.

[2] Y. Wang, Z.-Y. Ru, K. Wang, and P.-Q. Huang, "Joint deployment and task scheduling optimization for large-scale mobile users in multi-uav-enabled mobile edge computing," *IEEE Transactions on Cybernetics*, vol. 50, no. 9, pp. 3984–3997, 2020.

[3] P.-Q. Huang, Y. Wang, K. Wang, and Z.-Z. Liu, "A bilevel optimization approach for joint offloading decision and resource allocation in cooperative mobile edge computing," *IEEE Transactions on Cybernetics*, vol. 50, no. 10, pp. 4228–4241, 2020.

[4] P.-Q. Huang, Y. Wang, and K. Wang, "A divide-and-conquer bilevel optimization algorithm for jointly pricing computing resources and energy in wireless powered mec," *IEEE Transactions on Cybernetics*, 2021, to be published, doi:10.1109/TCYB.2021.3103840.

[5] C. You, B. Zheng, and R. Zhang, "Channel estimation and passive beamforming for intelligent reflecting surface: Discrete phase shift and progressive refinement," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 11, pp. 2604–2620, 2020.

[6] Q. Wu and R. Zhang, "Intelligent reflecting surface enhanced wireless network via joint active and passive beamforming," *IEEE Transactions on Wireless Communications*, vol. 18, no. 11, pp. 5394–5409, 2019.

[7] ——, "Towards smart and reconfigurable environment: Intelligent reflecting surface aided wireless network," *IEEE Communications Magazine*, vol. 58, no. 1, pp. 106–112, 2020.

[8] S. Gong, X. Lu, D. T. Hoang, D. Niyato, L. Shu, D. I. Kim, and Y. C. Liang, "Toward smart wireless communications via intelligent reflecting surfaces: A contemporary survey," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 4, pp. 2283–2314, 2020.

[9] J. Xu, W. Xu, and A. L. Swindlehurst, "Discrete phase shift design for practical large intelligent surface communication," in *2019 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing (PACRIM)*, 2019, pp. 1–5.

[10] V. D. Pegorara Souto, R. D. Souza, B. F. Uchôa-Filho, A. Li, and Y. Li, "Beamforming optimization for intelligent reflecting surfaces without csi," *IEEE Wireless Communications Letters*, vol. 9, no. 9, pp. 1476–1480, 2020.

[11] Y. Yang, S. Zhang, and R. Zhang, "IRS-enhanced ofdm: Power allocation and passive array optimization," in *2019 IEEE Global Communications Conference (GLOBECOM)*, 2019, pp. 1–6.

[12] P. Wang, J. Fang, X. Yuan, Z. Chen, and H. Li, "Intelligent reflecting surface-assisted millimeter wave communications: Joint active and passive precoding design," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 12, pp. 14 960–14 973, 2020.

[13] S. Zhang and R. Zhang, "Capacity characterization for intelligent reflecting surface aided MIMO communication," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 8, pp. 1823–1838, 2020.

[14] C. Pan, H. Ren, K. Wang, W. Xu, M. Elkashlan, A. Nallanathan, and L. Hanzo, “Multicell MIMO communications relying on intelligent reflecting surfaces,” *IEEE Transactions on Wireless Communications*, vol. 19, no. 8, pp. 5218–5233, 2020.

[15] Q. Wu and R. Zhang, “Weighted sum power maximization for intelligent reflecting surface aided SWIPT,” *IEEE Wireless Communications Letters*, vol. 9, no. 5, pp. 586–590, 2020.

[16] T. Bai, C. Pan, Y. Deng, M. Elkashlan, A. Nallanathan, and L. Hanzo, “Latency minimization for intelligent reflecting surface aided mobile edge computing,” *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 11, pp. 2666–2682, 2020.

[17] Z. Peng, T. Li, C. Pan, H. Ren, W. Xu, and M. D. Renzo, “Analysis and optimization for RIS-aided multi-pair communications relying on statistical CSI,” *IEEE Transactions on Vehicular Technology*, vol. 70, no. 4, pp. 3897–3901, 2021.

[18] R. Lin, Z. Zhou, S. Luo, Y. Xiao, X. Wang, S. Wang, and M. Zukerman, “Distributed optimization for computation offloading in edge computing,” *IEEE Transactions on Wireless Communications*, vol. 19, no. 12, pp. 8179–8194, 2020.

[19] Q. Wu and R. Zhang, “Beamforming optimization for wireless network aided by intelligent reflecting surface with discrete phase shifts,” *IEEE Transactions on Communications*, vol. 68, no. 3, pp. 1838–1851, 2020.

[20] L. Wang, K. Wang, C. Pan, W. Xu, and N. Aslam, “Joint trajectory and passive beamforming design for intelligent reflecting surface-aided UAV communications: A deep reinforcement learning approach,” *arXiv: 2007.08380*, 2020.

[21] H. Guo, Y.-C. Liang, J. Chen, and E. G. Larsson, “Weighted sum-rate maximization for reconfigurable intelligent surface aided wireless networks,” *IEEE Transactions on Wireless Communications*, vol. 19, no. 5, pp. 3064–3076, 2020.

[22] W. Zhang, J. Xu, W. Xu, D. W. K. Ng, and H. Sun, “Cascaded channel estimation for IRS-assisted mmwave multi-antenna with quantized beamforming,” *IEEE Communications Letters*, vol. 25, no. 2, pp. 593–597, 2021.

[23] P. Wang, J. Fang, H. Duan, and H. Li, “Compressed channel estimation for intelligent reflecting surface-assisted millimeter wave systems,” *IEEE Signal Processing Letters*, vol. 27, pp. 905–909, 2020.

[24] M. J. Neely, *Stochastic Network Optimization with Application to Communication and Queueing Systems*. Morgan and Claypool Publishers, 2010.

[25] X. Wang, K. Wang, S. Wu, S. Di, H. Jin, K. Yang, and S. Ou, “Dynamic resource scheduling in mobile edge cloud with cloud radio access network,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 29, no. 11, pp. 2429–2445, 2018.

[26] L. Huang and M. J. Neely, “Utility optimal scheduling in energy-harvesting networks,” *IEEE/ACM Transactions on Networking*, vol. 21, no. 4, pp. 1117–1130, 2013.

[27] X. Lyu, W. Ni, H. Tian, R. P. Liu, X. Wang, G. B. Giannakis, and A. Paulraj, “Optimal schedule of mobile edge computing for internet of things using partial information,” *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 11, pp. 2606–2615, 2017.

[28] X. Lyu, C. Ren, W. Ni, H. Tian, and R. P. Liu, “Distributed optimization of collaborative regions in large-scale inhomogeneous fog computing,” *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 3, pp. 574–586, 2018.

[29] S. Jiang, M. Kaiser, S. Yang, S. Kollias, and N. Krasnogor, “A scalable test suite for continuous dynamic multiobjective optimization,” *IEEE Transactions on Cybernetics*, vol. 50, no. 6, pp. 2814–2826, 2020.

[30] Q. Chen, J. Ding, S. Yang, and T. Chai, “A novel evolutionary algorithm for dynamic constrained multiobjective optimization problems,” *IEEE Transactions on Evolutionary Computation*, vol. 24, no. 4, pp. 792–806, 2020.

[31] Y. Liu, J. Liu, Y. Jin, F. Li, and T. Zheng, “An affinity propagation clustering based particle swarm optimizer for dynamic optimization,” *Knowledge-Based Systems*, vol. 195, p. 105711, May 2020.

[32] Y. Liu, J. Liu, and Y. Jin, “Surrogate-assisted multipopulation particle swarm optimizer for high-dimensional expensive optimization,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2021, to be published, doi:10.1109/TSMC.2021.3102298.

[33] G. Ruan, J. Zheng, J. Zou, Z. Ma, and S. Yang, “A random benchmark suite and a new reaction strategy in dynamic multiobjective optimization,” *Swarm and Evolutionary Computation*, vol. 63, p. 100867, 2021.

[34] M. Neely, “Energy optimal control for time-varying wireless networks,” *IEEE Transactions on Information Theory*, vol. 52, no. 7, pp. 2915–2934, 2006.

[35] N. Liu, X. Yu, W. Fan, C. Hu, T. Rui, Q. Chen, and J. Zhang, “Online energy sharing for nanogrid clusters: A Lyapunov optimization approach,” *IEEE Transactions on Smart Grid*, vol. 9, no. 5, pp. 4624–4636, 2018.

[36] Y. Cui, V. K. N. Lau, R. Wang, H. Huang, and S. Zhang, “A survey on delay-aware resource control for wireless systems—large deviation theory, stochastic Lyapunov drift, and distributed stochastic learning,” *IEEE Transactions on Information Theory*, vol. 58, no. 3, pp. 1677–1701, 2012.

[37] L. Zheng and L. Cai, “A distributed demand response control strategy using Lyapunov optimization,” *IEEE Transactions on Smart Grid*, vol. 5, no. 4, pp. 2075–2083, 2014.

[38] S. Mao, S. Leng, S. Maharjan, and Y. Zhang, “Energy efficiency and delay tradeoff for wireless powered mobile-edge computing systems with multi-access schemes,” *IEEE Transactions on Wireless Communications*, vol. 19, no. 3, pp. 1855–1867, 2020.

[39] Q. Wu, S. Zhang, B. Zheng, C. You, and R. Zhang, “Intelligent reflecting surface-aided wireless communications: A tutorial,” *IEEE Transactions on Communications*, vol. 69, no. 5, pp. 3313–3351, 2021.

[40] R. Jiao, B. Xue, and M. Zhang, “Investigating the correlation amongst the objective and constraints in gaussian process-assisted highly-constrained expensive optimization,” *IEEE Transactions on Evolutionary Computation*, 2021, to be published, doi: 10.1109/TEVC.2021.3120980.

[41] R. Jiao, S. Zeng, C. Li, S. Yang, and Y.-S. Ong, “Handling constrained many-objective optimization problems via problem transformation,” *IEEE Transactions on Cybernetics*, vol. 51, no. 10, pp. 4834–4847, 2021.

[42] M. Asim, Y. Wang, K. Wang, and P.-Q. Huang, “A review on computational intelligence techniques in cloud and edge computing,” *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 4, no. 6, pp. 742–763, 2020.

[43] S. Das, S. S. Mullick, and P. Suganthan, “Recent advances in differential evolution - an updated survey,” *Swarm and Evolutionary Computation*, vol. 27, pp. 1–30, 2016.

[44] K. C. Tan, L. Feng, and M. Jiang, “Evolutionary transfer optimization - a new frontier in evolutionary computation research,” *IEEE Computational Intelligence Magazine*, vol. 16, no. 1, pp. 22–33, 2021.

[45] A. Gupta, Y.-S. Ong, and L. Feng, “Multifactorial evolution: Toward evolutionary multitasking,” *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 3, pp. 343–357, 2016.

[46] J. Ding, C. Yang, Y. Jin, and T. Chai, “Generalized multitasking for evolutionary optimization of expensive problems,” *IEEE Transactions on Evolutionary Computation*, vol. 23, no. 1, pp. 44–58, 2019.

[47] S. Yao, Z. Dong, X. Wang, and L. Ren, “A multiobjective multifactorial optimization algorithm based on decomposition and dynamic resource allocation strategy,” *Information Sciences*, vol. 511, pp. 18–35, 2020.

[48] M. Jiang, Z. Huang, L. Qiu, W. Huang, and G. G. Yen, “Transfer learning-based dynamic multiobjective optimization algorithms,” *IEEE Transactions on Evolutionary Computation*, vol. 22, no. 4, pp. 501–514, 2018.

[49] M. Jiang, Z. Wang, L. Qiu, S. Guo, X. Gao, and K. C. Tan, “A fast dynamic evolutionary multiobjective algorithm via manifold transfer learning,” *IEEE Transactions on Cybernetics*, vol. 51, no. 7, pp. 3417–3428, 2021.

[50] L. Feng, W. Zhou, W. Liu, Y.-S. Ong, and K. C. Tan, “Solving dynamic multiobjective problem via autoencoding evolutionary search,” *IEEE Transactions on Cybernetics*, 2020, to be published, doi:10.1109/TCYB.2020.3017017.

[51] R. Santana, A. Mendiburu, and J. A. Lozano, “Structural transfer using edas: An application to multi-marker tagging SNP selection,” in *2012 IEEE Congress on Evolutionary Computation*, 2012, pp. 1–8.

[52] L. Feng, Y.-S. Ong, M.-H. Lim, and I. W. Tsang, “Memetic search with interdomain learning: A realization between CVRP and CARP,” *IEEE Transactions on Evolutionary Computation*, vol. 19, no. 5, pp. 644–658, 2015.

[53] S. J. Pan and Q. Yang, “A survey on transfer learning,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.

[54] L. Feng, Y.-S. Ong, S. Jiang, and A. Gupta, “Autoencoding evolutionary search with learning across heterogeneous problems,” *IEEE Transactions on Evolutionary Computation*, vol. 21, no. 5, pp. 760–772, 2017.

[55] X.-F. Liu, Z.-H. Zhan, T.-L. Gu, S. Kwong, Z. Lu, H. B.-L. Duh, and J. Zhang, “Neural network-based information transfer for dynamic optimization,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 31, no. 5, pp. 1557–1570, 2020.

[56] X. Wang, Z. Dong, and L. Tang, “Multiobjective differential evolution with personal archive and biased self-adaptive mutation selection,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 12, pp. 5338–5350, 2020.

Pei-Qiu Huang received the B.S. degree in automation and the M.S. degree in control theory and control engineering both from the Northeastern University, Shenyang, China, in 2014 and 2017, respectively, and the Ph.D. degree in control science and engineering, Central South University, Changsha, China, in 2021.

He is a Research Assistant with the Department of Computer Science, City University of Hong Kong, Hong Kong. His current research interests include evolutionary computation, bilevel optimization, and mobile edge computing.

Yong Wang (M'08–SM'17) received the Ph.D. degree in control science and engineering from the Central South University, Changsha, China, in 2011.

He is a Professor with the School of Automation, Central South University, Changsha, China. His current research interests include intelligent learning and optimization and their interdisciplinary applications.

Dr. Wang is an Associate Editor of the IEEE Transactions on Evolutionary Computation and the Swarm and Evolutionary Computation. He was a recipient of Cheung Kong Young Scholar by the Ministry of Education, China, in 2018, and a Web of Science highly cited researcher in Computer Science in 2017 and 2018.

Kezhi Wang (M'15–SM'20) received the B.E. and M.E. degrees from the School of Automation, Chongqing University, China, in 2008 and 2011, respectively, and the Ph.D. degree in engineering from the University of Warwick, U.K., in 2015. He was a Senior Research Officer with the University of Essex, U.K., from 2015 to 2017.

He is currently a Senior Lecturer with the Department of Computer and Information Sciences, Northumbria University, U.K. His research interests include mobile edge computing, intelligent reflection surface, and machine learning.

Qingfu Zhang (M¡¯01-SM¡¯06-F¡¯17) received the BSc degree in mathematics from Shanxi University, China in 1984, the MSc degree in applied mathematics and the PhD degree in information engineering from Xidian University, China, in 1991 and 1994, respectively.

He is a Chair Professor of Computational Intelligence at the Department of Computer Science, City University of Hong Kong. His main research interests include evolutionary computation, optimization, neural networks, machine learning and their applications.

Dr. Zhang is an Associate Editor of the IEEE Transactions on Evolutionary Computation and the IEEE Transactions on Cybernetics. He is a Web of Science highly cited researcher in Computer Science.