

Distributed Self-triggered Control for Frequency Restoration and Active Power Sharing in Islanded Microgrids

Yulin Chen, *Member, IEEE*, Keng-Weng Lao, *Senior Member, IEEE*, Donglian Qi, *Senior Member, IEEE*, Hongxun Hui, *Member, IEEE*, Shaohua Yang, Yunfeng Yan, Yi Zheng

Abstract—Distributed event-triggered secondary control in microgrids have been widely investigated to improve system efficiency. But most of them are based on consecutive triggering condition monitor, which would in turn increase the computation burden of the system. To this end, this paper presents distributed self-triggered algorithmic solutions to the frequency restoration control and active power sharing control of islanded microgrids. Different from event-triggered control schemes, in our self-triggered solutions, each distributed generator is equipped with a local algorithm that enables it to pre-compute the next triggering time instant according to the states at previous one. Our starting point is to design an triggering condition with a novel estimate error. Then, the next triggering time instant is determined by solving a quadratic equation established based on the triggering condition, rather than monitoring the triggering condition consecutively. Theoretical analysis and simulation results show that the proposed distributed self-triggered secondary controllers can highly reduce the communication and computation cost simultaneously.

Index Terms—Self-triggered control, Distributed Control, Frequency restoration, Active power sharing, Islanded microgrid.

I. INTRODUCTION

EXPLOITATION of distributed renewable energies in an effective way plays an important role in the decarbonization objective. Microgrid (MG) is considered as a feasible way to better utilize the distributed renewable energies [1]–[3]. To coordinate the contradiction between the renewable energies' fluctuation and the stability of the power system, a multilevel control structure is used [4], [5]. In the structure,

Yulin Chen is with Hainan Institute of Zhejiang University, Sanya, Hainan, 572024, China, and also with the State Key Laboratory of Internet of Things for Smart City and Department of Electrical and Computer Engineering, University of Macau, Macao, 999078, China.

Keng-Weng Lao, Hongxun Hui and Shaohua Yang are with the State Key Laboratory of Internet of Things for Smart City and Department of Electrical and Computer Engineering, University of Macau, Macao, 999078, China.

Donglian Qi is with Hainan Institute of Zhejiang University, Sanya, Hainan, 572024, China, and also with the College of Electrical Engineering, Zhejiang University, Hangzhou, Zhejiang, 310027, China.

Yunfeng Yan and Yi Zheng are with the College of Electrical Engineering, Zhejiang University, Hangzhou, Zhejiang, 310027, China.

(Corresponding author: Keng-Weng Lao).

Manuscript received July 25, 2022; revised November 26, 2022; accepted December 30, 2022. This work was supported in part by the Science and Technology Development Fund, Macau SAR (Project no. SKL-IOTSC-2021-2023, FDCT/0022/2020/A1, FDCT/0076/2019/AMJ), the National Natural Science Foundation of China (Grant No. U1909201,62001416,62101490) and the Research Startup Funding from Hainan Institute of Zhejiang University (Grant No. 0210-6602-A12202).

it requires the primary control to stabilize the frequency and the voltage quickly. Then, the secondary control is needed to achieve frequency and voltage restoration and accurate power sharing. For the secondary control in MGs, conventionally, it is always realized though centralized control structure [6] with consecutive periodic communication. However, this control mode requires higher data transmission rate and has poor scalability. In fact, the performance of the secondary control in MGs can be improved by the distributed cooperative control approach.

Distributed secondary control in MGs can be achieved by using multi-agent system-based consensus algorithms [7], which can achieve the global objective with only local control and neighboring communication, providing more scalability, flexibility and reliability. Therefore, the distributed secondary control is a superior choice for MGs to manage decentralized DGs. The first distributed secondary control in MGs refers to [8], in which the DGs are considered as agents, and a fair utilization profile is achieved among DGs by employing average consensus algorithm. Then, the authors in [9] propose the earliest distributed secondary control of frequency restoration by using leader-follower consensus algorithm. After that, a series of distributed secondary control algorithms in MGs are proposed for better performances relating to finite-time convergence [10], optimal operation [11], attack resistance [12]–[14], etc.

Notice that distributed secondary control relies on communication network to achieve the global objective. Conventionally, the data transmission of distributed control is implemented through sampled data system, resulting in a consecutive periodic information exchange among DGs. To meet the requirement in the worst possible extreme situations, the fixed sample period may be selected to be very small, which may cause conservative usage of the computation and communication resources during both transient process and steady state. On the other hand, considering the limitation of the resources in the local of DGs, it would increase the communication burden and lead to time delay or packet loss when the system scale is expanded, especially for the MGs with rapid growth of DGs. In this context, if the data transmission occurs only when it is needed, the efficiency of the distributed secondary control can be improved a lot. To this end, the event-triggered mechanism starts to rise [15]–[17].

By event-triggered control, the controller only updates and transmits its new states to its neighbors when a designed

triggering condition is satisfied [17], [18], which is reactive in nature. In this way, the communication burden will be greatly reduced. For example, the authors in [19] designed a distributed event-triggered consensus-like nonlinear state feedback control to achieve reactive power allocation distributedly with reduced communications. A distributed event-triggered control scheme for frequency restoration with the aid of an extra distributed estimator was proposed in [20]. In [21], centralized and distributed event-triggered solutions are proposed for frequency and voltage restoration in MGs to reduce controller updates. To tolerate communication time delay, a novel event-triggered control law is proposed in [22] for the voltage restoration in AC MGs. In [23], [24], both power sharing control and restoration control were achieved with event-triggered mechanism simultaneously. After that, the authors in [25] used the event-triggered average consensus algorithm proposed in [17] to the power sharing control, and achieved the overall secondary control objectives of islanded MGs by decoupling the power sharing and the restoration control objectives. In [26], distributed event-triggered control strategies are designed for the current sharing and bus voltage regulation in DC MGs. However, there exists convergence error due to the design of the triggering condition. To achieve better convergence performance, a PI-based distributed event-triggered secondary controller with a constant threshold was proposed in [27]. To make the system converge to the objective in a desired period of time, a finite-time distributed event-triggered secondary frequency and voltage control for islanded AC MGs is proposed in [28]. In [29], the authors investigate the secondary control of DC MGs with a distributed dynamic event-triggering mechanism, with which the minimum inter-event time between triggers can be enlarged. Similarly, a distributed dynamic event-triggered protocol for the secondary control in islanded AC MGs is proposed in [30].

However, the event-triggered control strategies mentioned above require for continuous monitoring of a triggering condition, which would result in a significant computational overhead. To save the computation cost, self-triggered mechanism is investigated. For the self-triggered control, the controller evaluates the next triggering time instant ahead of time without involving triggering condition monitoring [17], [31]–[33], which is proactive in nature. Therefore, self-triggered control has better performance in reducing both communication and computation requirements when performing the secondary control in MGs. But, the study of self-triggered secondary control is insufficient. Several relating researches include: a discrete-time self-triggered controller is proposed in [34] to solve load current sharing problem in DC MGs considering data dropouts and communication delay. However, it mandates complex calculation for acquiring the next event time. In addition, a distributed self-triggered secondary control in MGs relating to reactive power sharing is proposed in [35]. The proposed controller is designed based on the ternary control in [31]. However, it leads to convergence error for resulting fewer events of the controller. To the best of the author's knowledge, for the AC MGs, the existing studies only focus on power sharing problem. Besides, the computation of the existing self-triggered controllers is complex, which motivates

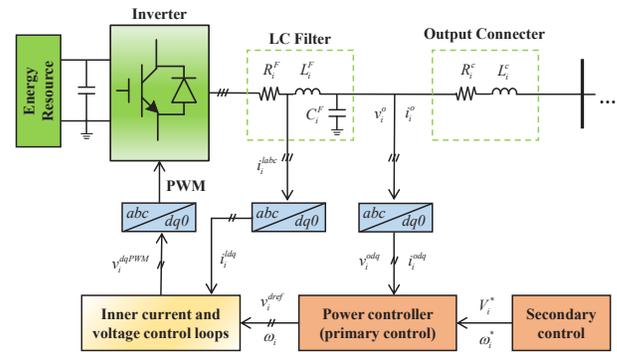


Fig. 1. Schematic view of the control structure of an inverter-based DG.

our research in this paper.

In this paper, we proposed a novel distributed self-triggered control algorithm for both frequency restoration control and power sharing control in islanded MGs. The novelties of this paper are summarized as follows:

- 1) A distributed self-triggered control algorithm for frequency restoration is developed based on a novel distributed event-triggered condition considering a new kind of estimate error. On this basis, both communication and computation requirements are significantly reduced.
- 2) Following the same design principle, active power sharing is realized by distributed self-triggered control mechanism as well. Then, both frequency restoration and active power sharing objectives are achieved with reduced communication and computation costs.
- 3) The designed distributed self-triggered controllers only need to iteratively compute quadratic equations to obtain the next triggering time instants, which is simpler and more efficient than that of the existing self-triggered controllers.

The reminder of this paper is organized as follows: In Section II, a brief introduction of the primary droop control and distributed secondary control of MGs is presented. In Section III, our main result of this work relating the self-triggered controller design for the frequency restoration and active power sharing of islanded MGs is proposed. Simulation results are provided in Section IV to validate the effectiveness and advantages of the proposed controllers. Finally, Section V concludes our work in this paper.

II. PRIMARY AND SECONDARY CONTROL IN MICROGRIDS

In MGs, distributed renewable energies can be modeled as three-phase inverter-based DGs, which individually connect to the MG through a DC/AC converter and an LC filter via different inductive dominated line impedances [36]. The inverter-based DGs are regulated by inner voltage control, inner current control, and PWM control to generate the desired power with very short time scales [37], which require the reference signals from the primary control. The schematic view of the control structure can be illustrated in Fig. 1.

A. Droop-based Primary Control

For the primary control, droop technique is usually implemented since it mimics the characteristic of the traditional synchronous generator. It describes the relationship between active power and frequency, and the relationship between reactive power and voltage magnitude. DG i 's droop mechanism can be given by, for $i \in \{1, 2, \dots, N\}$,

$$\begin{cases} \omega_i = \omega_i^* - m_i P_i \\ V_i = V_i^* - n_i Q_i, \end{cases} \quad (1)$$

where N is the total number of DGs. ω_i and V_i are the frequency and voltage magnitude of DG i , respectively. P_i and Q_i are the DG i 's output active and reactive power. m_i and n_i are the frequency and voltage droop coefficients. ω_i^* and V_i^* are the references of the droop control.

However, the droop mechanism may cause frequency and voltage deviations due to the mismatch between the power supply and demand. Thus, the secondary control is required to compensate for the deviations.

B. Distributed Secondary Frequency Restoration Control and Active Power Sharing Control

In this paper, we focus on frequency restoration and active power sharing objectives in the secondary control level.

As mentioned previously, distributed cooperative control mode is a better choice for implementing the secondary control compared with the centralized and decentralized control modes. In distributed secondary control, the controller is deployed at each DG's local and only exchanges information with its neighbors, which provides more flexibility and better scalability. Therefore, the communication network is important for the distributed control system. Typically, the communication network can be described by an undirected graph $\mathcal{G} = (\mathcal{V} \times \mathcal{E})$, in which \mathcal{V} represents the agent set corresponding to the DG set in the MG, and $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$ denotes the set of edges corresponding to the communication links between DGs. An existing edge $(i, j) \in \mathcal{E}$ means that agent i can communicate with agent j . Then, agent j is called the neighbor of agent i . We denote by \mathcal{N}_i the neighbor set of DG i . An adjacency matrix $A = [a_{ij}]_{N \times N}$ is required to formulate the existing of edges, where $a_{ij} = 1$ if $(i, j) \in \mathcal{E}$, and $a_{ij} = 0$, otherwise. Since the graph is undirected, we have $A = A^T$. The degree of agent i is defined as $d_i = \sum_{j=1}^N a_{ij}$ representing the number of agent i 's neighbors. Then, the Laplacian matrix of graph \mathcal{G} is defined as $L = D - A$, where $D = \text{diag}\{[d_1, \dots, d_N]\}$.

The secondary frequency restoration control and active power sharing control are usually realized by turning the set point ω_i^* of primary control [9], [38], which can be derived from (1) by

$$\omega_i^* = \int (\dot{\omega}_i + m_i \dot{P}_i) dt. \quad (2)$$

For the frequency restoration control, it is worth emphasizing that the reference value may be changed due to the higher level control objective. Therefore, it is applicable to

use the leader-follower consensus-based controller to achieve the frequency restoration objective [38], which is given by:

$$\dot{\omega}_i = u_i^\omega = -k_\omega \sum_{j \in \mathcal{N}_i} a_{ij} (\omega_i - \omega_j) - k_\omega b_i (\omega_i - \omega^r), \quad (3)$$

where u_i^ω is the control input. ω^r is the reference value of frequency. $k_\omega > 0$ is the control gain. $b_i = 1$ indicates DG i can receive the reference signal, $b_i = 0$, otherwise.

For the active power sharing control, average consensus algorithm is usually employed such that each DG generates active power with the same utilization profile, which is given by:

$$m_i \dot{P}_i = u_i^P = -k_P \sum_{j \in \mathcal{N}_i} a_{ij} (m_i P_i - m_j P_j). \quad (4)$$

By these implementations, with a connected communication network, the following frequency restoration and power sharing objectives can be achieved in distributed manners [9] as follows:

$$\lim_{t \rightarrow \infty} |\omega_i(t) - \omega_i^r| = 0, \quad (5a)$$

$$\lim_{t \rightarrow \infty} [m_i P_i(t) - m_j P_j(t)] = 0. \quad (5b)$$

III. DISTRIBUTED SELF-TRIGGERED SECONDARY CONTROLLER DESIGN

In this section, we first design a self-triggered controller for frequency restoration in islanded MGs. Then, following the similar principle, we design a self-triggered controller for active power sharing control. Theoretical analysis is conducted to prove the correctness.

A. Distributed Self-triggered Controller Design for Frequency Restoration

1) *Triggering Condition Design:* For the DG i 's controller, $i \in \{1, 2, \dots, N\}$, the triggering time instants are denoted by a sequence $t_0^i, t_1^i, \dots, t_k^i, \dots$, and we denote the following disagreement as

$$q_i(t) = - \sum_{j \in \mathcal{N}_i} a_{ij} (\omega_i(t) - \omega_j(t)) - b_i (\omega_i(t) - \omega^r). \quad (6)$$

Then, the secondary control input is designed as:

$$u_i^\omega(t) = k_\omega \hat{q}_i(t). \quad (7)$$

where $\hat{q}_i(t)$ satisfies the following update rule

$$\hat{q}_i(t) = q_i(t_k^i), \quad t \in [t_k^i, t_{k+1}^i). \quad (8)$$

Rule (8) implies that DG i only updates $\hat{q}_i(t)$ by calling for $\omega_i(t)$ and $\omega_j(t)$, $j \in \mathcal{N}_i$, at its triggering time instant t_k^i or at the time instant when any of its neighboring DG j , ($j \in \mathcal{N}_i$), triggers. Otherwise, $\hat{q}_i(t)$ remains unchanged. Therefore, $\hat{q}_i(t)$ is piecewise continuous. This rule also means that the communication is only executed at the event time instants of DG i or its neighbors, rather than continuously.

Therefore, how to determine t_k^i is the key to the controller design.

To find the triggering condition that determines when to conduct a trigger, we define the following estimate error

$$e_i(t) = \hat{q}_i(t) - q_i(t), \quad (9)$$

and the following tracking error,

$$\varepsilon_i(t) = \omega_i(t) - \omega^r. \quad (10)$$

Notice the estimate error (9) defined here is different from the existing ones in the literature [18], [25], [27].

Then, the triggering time instant that maintains the convergence correctness can be designed by

$$t_{k+1}^i = \min_{t > t_k^i} \{t \mid e_i^2(t) \geq \sigma_i q_i^2(t)\}, \quad (11)$$

where $e_i^2(t) \geq \sigma_i q_i^2(t)$ is the triggering condition. $0 < \sigma_i < 1$ is an adjustable parameter that affects the number of triggering time instants.

The following Lemma shows how the triggering time instant defined in (11) achieves the frequency restoration.

Lemma 1: For an islanded MG with N DGs, assume that the DGs are connected by an undirected connected communication network and at least one DG can receive the reference frequency signal, then the DGs group will asymptotically achieve frequency restoration in (5a) under the control law (7) with the triggering time instants defined in (11). Moreover, there doesn't exist Zeno behavior for each DG's controller.

Proof: To conduct the proof, we first provide two relationships in compact forms.

From (6), (7) and (9), one can derive

$$\dot{\omega}(t) = k_\omega \mathbf{q}(t) + k_\omega \mathbf{e}(t), \quad (12)$$

and from (6) and (10), we have

$$\mathbf{q}(t) = -H\varepsilon(t), \quad (13)$$

where $H = L + B$, and $B = \text{diag}([b_1, \dots, b_N])$. Since the communication links are undirected, $H = H^T$ is hold. In the compact form, the bold lowercase variable represents the vector with proper dimension, for example $\mathbf{q}(t)$ in (12) is referred to as $\mathbf{q}(t) = [q_1, q_2, \dots, q_N]^T$.

Consider the Lyapunov function candidate: $W(t) = \frac{1}{2} \varepsilon^T(t) H \varepsilon(t)$. Differentiating the Lyapunov function yields

$$\dot{W}(t) = \varepsilon^T(t) H \dot{\varepsilon}(t). \quad (14)$$

Due to $\dot{\varepsilon}(t) = \dot{\omega}(t)$ and $H = H^T$, by invoking (12) and (13), we have

$$\begin{aligned} \dot{W}(t) &= \varepsilon^T(t) H \dot{\omega}(t) \\ &= -\mathbf{q}^T(t) [k_\omega \mathbf{q}(t) + k_\omega \mathbf{e}(t)] \\ &= -k_\omega \mathbf{q}^T(t) \mathbf{q}(t) - k_\omega \mathbf{q}^T(t) \mathbf{e}(t). \end{aligned} \quad (15)$$

Using the inequality $-xy \leq \frac{1}{2}x^2 + \frac{1}{2}y^2$ and expanding (15) out, we have

$$\begin{aligned} \dot{W}(t) &\leq -k_\omega \sum_{i=1}^N q_i^2(t) + k_\omega \sum_{i=1}^N \left[\frac{1}{2} q_i^2(t) + \frac{1}{2} e_i^2(t) \right] \\ &\leq -\frac{1}{2} k_\omega \sum_{i=1}^N [q_i^2(t) - e_i^2(t)]. \end{aligned} \quad (16)$$

If we guarantee

$$e_i^2(t) < \sigma_i q_i^2(t) \quad (17)$$

with $0 < \sigma_i < 1$, which is referred to as the stability condition, then one has

$$\dot{W}(t) \leq -\frac{1}{2} k_\omega \sum_{i=1}^N (1 - \sigma_i) q_i^2(t) < 0. \quad (18)$$

Here, we finish the controller stability proof.

Since the triggers should occur when the stability condition is violated, we define the triggering condition by:

$$e_i^2(t) \geq \sigma_i q_i^2(t). \quad (19)$$

Next, we prove the exclusion of Zeno behavior by evaluating the lower bound on the time intervals between triggering time instants.

From definition (9), one can derive that

$$|\hat{q}_i(t)| = |q_i(t) + e_i(t)| \leq |q_i(t)| + |e_i(t)|. \quad (20)$$

Combining (19) with (20), we have

$$|\hat{q}_i(t)| \leq \left(1 + \frac{1}{\sqrt{\sigma_i}}\right) |e_i(t)|. \quad (21)$$

For $t \in [t_k^i, t_{k+1}^i)$, considering no neighbor's information is received, we have the following relationship from (9),

$$\dot{e}_i(t) = -\dot{q}_i(t) = -h(t), \quad (22)$$

where

$$h_i(t) = -k_\omega \sum_{j \in \mathcal{N}_i} a_{ij} (\hat{q}_i(t) - \hat{q}_j(t)) - k_\omega b_i \hat{q}_i(t). \quad (23)$$

Since $h(t)$ is constant in $[t_k^i, t_{k+1}^i)$ and $e_i(t_k^i) = 0$, integrating (22) from t_k^i to t , we have

$$e_i(t) = -(t - t_k^i) h(t). \quad (24)$$

Thus, by using (21) and (24), one can derive that

$$(t - t_k^i) |h(t)| \geq \frac{\sqrt{\sigma_i}}{1 + \sqrt{\sigma_i}} |\hat{q}_i(t)|. \quad (25)$$

Then, since $\hat{q}_i(t) \neq 0$, it can be derived from (25) that

$$\begin{aligned}
 t_{k+1}^i - t_k^i &\geq t - t_k^i \geq \frac{\sqrt{\sigma_i}|\hat{q}_i(t)|}{(1 + \sqrt{\sigma_i})|h(t)|} \\
 &\geq \frac{\sqrt{\sigma_i}|\hat{q}_i(t)|}{(1 + \sqrt{\sigma_i})(|\mathcal{N}_i| + b_i)|\hat{q}_i(t)| + |\sum_{j \in \mathcal{N}_i} a_{ij}\hat{q}_j(t)|} \\
 &= \frac{\sqrt{\sigma_i}}{k_\omega(1 + \sqrt{\sigma_i})(|\mathcal{N}_i| + b_i + \frac{|\sum_{j \in \mathcal{N}_i} a_{ij}\hat{q}_j(t)|}{|\hat{q}_i(t)|})} > 0.
 \end{aligned} \tag{26}$$

where $|\mathcal{N}_i|$ is the cardinal number of \mathcal{N}_i . Here we have used the inequality of $|h(t)| \leq k_\omega(|\mathcal{N}_i| + b_i)|\hat{q}_i(t)| + k_\omega|\sum_{j \in \mathcal{N}_i} a_{ij}\hat{q}_j(t)|$. Thus, Zeno behavior is proved non-existent for each DG's controller.

The proof is complete.

Notice that this controller requires each DG to monitor the trigger condition all the time. Moreover, each controller needs to call for $\omega_j(t)$ from its neighbors to calculate $q_i(t)$ continuously. This is not an effective event-triggered controller. But we can use this condition to design a self-triggered controller.

2) *Self-triggered Controller Design*: In this subsection, we provide our self-triggered frequency restoration controller design based on the proposed triggering condition (19).

The self-triggered mechanism can overcome the drawbacks of the event-triggered mechanism due to the following feature: The next triggering time instant t_{k+1}^i can be pre-computed at the previous event time t_k^i without involving consecutive triggering condition monitoring. Therefore, our main job is to derive t_{k+1}^i by using the states in $[t_k^i, t_{k+1}^i)$ for DG i 's controller, $i = \{1, 2, \dots, N\}$.

From definition (23), we know that

$$h_i(t) = \frac{d}{dt}q_i(t). \tag{27}$$

Integrating both-hand sides of (22) from t_k^i to t , $t \in (t_k^i, t_{k+1}^i)$, we have

$$e_i(t) = - \int_{t_k^i}^t h_i(t)dt. \tag{28}$$

Invoking (9), one has

$$q_i(t) = \hat{q}_i(t) + \int_{t_k^i}^t h_i(t)dt. \tag{29}$$

Considering (28), (29) and the triggering condition (19), if we can compute $\int_{t_k^i}^t h_i(t)dt$, then we can deduce the time instant t_{k+1}^i by solving t . Therefore, we need to express $\int_{t_k^i}^t h_i(t)dt$ as simple as possible so that t can be solved easily.

Notice that $h_i(t)$ is piecewise continuous, namely $h_i(t)$ remains constant unless one of its neighbor is triggered, at which a new $\hat{q}_j(t)$ is received. Thus,

$$\Omega(t) = \int_{t_k^i}^t h_i(t)dt = \int_{t_k^i}^s h_i(t)dt + h_i(s)(t - s), \tag{30}$$

is always hold, where $s = \max_{j \in \mathcal{N}_i} \{t_k^j \leq t\}$ is the latest event of neighboring DGs.

Since $h_i(t)$ is a piecewise constant and we know all time instants in $[t_k^i, t_s^i]$, we can use the following iterative procedure to obtain $\Omega(t)$.

We need a flag p and a temporary variable s_p to conduct our analysis. They are initialized by $p = 0$ and $s_0 = t_k^i$. If any new state $\hat{q}_j(t)$ from neighbor is received, p is updated by $p = p + 1$ and s_p is updated by the current time instant. Then, in $[t_k^i, t)$, $\Omega(t)$ can be represented by:

$$\Omega(t) = \sum_{l=1}^p h_i(s_{l-1})(s_l - s_{l-1}) + h_i(s_p)(t - s_p). \tag{31}$$

Thus, (31) is the iterative implementation of (30). Then, we further simplify (31) by

$$\Omega(t) = \Omega_p + h_i(s_p)(t - s_p), \tag{32}$$

where $\Omega_p = \sum_{l=1}^p h_i(s_{l-1})(s_l - s_{l-1})$, $p = 1, 2, \dots$, and $\Omega_0 = 0$.

Since $e_i(t)$ and $q_i(t)$ can be determined by (28), (29) and hence by (32), t_{k+1}^i thus can be obtained by solving the following equation, which is derived from the critical condition of the triggering condition (19),

$$\Omega^2(t) = \sigma_i[\Omega(t) + \hat{q}_i(t)]^2. \tag{33}$$

To solve t , at which $t = t_{k+1}^i$. We first denote $z = t - s_p$, $h_p = h_i(s_p)$. Thus (33) becomes

$$(\Omega_p + zh_p)^2 = \sigma_i[\Omega_p + zh_p + \hat{q}_i(t)]^2. \tag{34}$$

Then, one can obtain the following quadratic equation of z by expanding (34) as:

$$az^2 + bz + c = 0, \tag{35}$$

where $a = (1 - \sigma_i)h_p^2$, $b = 2\Omega_p h_p - 2\sigma_i h_p[\Omega_p + \hat{q}_i(t)]$, $c = \Omega_p^2 - \sigma_i[\Omega_p + \hat{q}_i(t)]^2$.

Notice that $a > 0$ since $0 < \sigma_i < 1$ and $h_p \neq 0$, and $\Delta = b^2 - 4ac = 4\sigma_i h_p^2 \hat{q}_i^2(t) > 0$ if $\hat{q}_i(t) \neq 0$, then (35) have two solutions. From (33), we know that $c < 0$ before $t = t_{k+1}^i$. Thus, (35) has exactly one positive solution denoted by

$$z_+ = \begin{cases} \frac{\sigma_i \hat{q}_i(t) + \sqrt{\sigma_i}|\hat{q}_i(t)| - (1 - \sigma_i)\Omega_p}{(1 - \sigma_i)h_p}, & \text{if } \hat{q}_i(t) > 0 \\ \frac{\sigma_i \hat{q}_i(t) - \sqrt{\sigma_i}|\hat{q}_i(t)| - (1 - \sigma_i)\Omega_p}{(1 - \sigma_i)h_p}, & \text{if } \hat{q}_i(t) < 0. \end{cases} \tag{36}$$

The next triggering time instant t_{k+1}^i can be determined by the following process. For DG i 's controller, the controller records the last triggering time instant t_k^i of DG i or t_k^j of its neighbors by s_p . Then, the controller computes z_+ according to (36) and obtains a candidate next triggering time instant as $t_{k+1}^i = s_p + z_+$. If a new state is received from any neighbor before t_{k+1}^i , then update s_p by the receiving moment, e.g. t_{k+1}^j . At the same time, the controller computes (36) again to get a new z_+ and takes $t_{k+1}^i = s_p + z_+$ as a new candidate next triggering time instant. Repeat the process until no new state is received from neighboring controllers before t_{k+1}^i , then, the

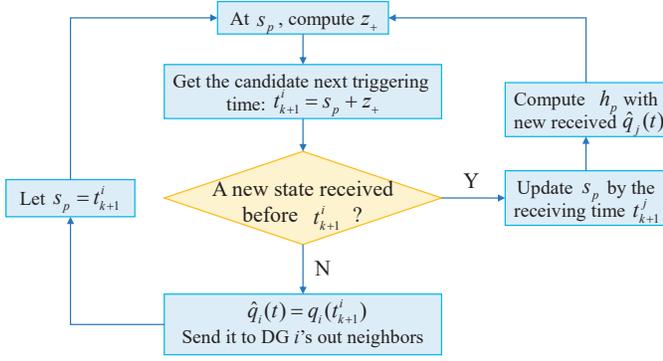


Fig. 2. The flow chart diagram of the frequency restoration and active power sharing distributed self-triggered controllers.

next trigger will be executed when t reaches t_{k+1}^i . At that moment, DG i 's controller will update $\hat{q}_i(t)$ by $q_i(t_{k+1}^i)$ and send it to its out neighbors. The flow chart of DG i 's self-triggered control can be illustrated by the in Fig. 2. And the overall procedure can be summarized in **Algorithm 1**.

It is worth mentioning that, to obtain the next event time in advance, each DG only needs to iteratively solving the quadratic equation (36), rather than computing the triggering condition continuously. Thus, it can reduce a lot of computation operations compared with event-triggered controllers. Moreover, it is also simpler than most of the existing self-triggered controllers [17], [39] due to its computation simplicity.

Algorithm 1 Self-triggered control algorithm for DG i

INITIALIZE:

$$\begin{aligned} k &\leftarrow 0; t_k^i \leftarrow 0; \\ p &\leftarrow 0; s_p \leftarrow t_k^i; \Omega_p \leftarrow 0; h_p \leftarrow h_i(s_p) \text{ as in (23);} \end{aligned}$$

ENSURE:

```

while t < T do
  if h_p = 0 then
    Continue;
  else
    compute z_+ as (36); tau_k = s_p + z_+;
  end if
  if t < tau_k & a new state q_j(t) is received then
    h_temp leftarrow h_p; Omega_temp leftarrow Omega_p; p = p + 1;
    Omega_p leftarrow Omega_temp + h_temp(t - s_p);
    s_p leftarrow t; h_p leftarrow h_i(s_p); Continue;
  else if t = tau_k then
    k = k + 1; t_k^i = tau_k;
    update q_i(t) by acquiring the local states omega_i(t);
    propagate q_i(t) to all neighbors;
    p leftarrow 0; s_p leftarrow t_k^i; Omega_p leftarrow 0; h_p leftarrow h_i(s_p);
  end if
end while
  
```

The following Theorem concludes the correctness of the proposed distributed self-triggered controller.

Theorem 1: For an islanded MG with N DGs, assume

that the DGs are connected by an undirected connected communication network and at least one DG can receive the reference frequency signal, then the DGs group will asymptotically achieve frequency restoration in (5a) under the control algorithm in Algorithm 1.

Proof: Since Algorithm 1 is designed according to the proposed triggering condition (19), the outputs of all DGs determined by Algorithm 1 are exactly equivalent to those determined by the control in Lemma 1 if both of them have the same initial conditions. In addition, since we have proved that the proposed triggering condition can exclude Zeno behavior, the self-triggered controller does not exhibit Zeno behavior, either.

For the implementation of the proposed controller, the neighboring communication and computation of z_+ are not difficult to realize. For each DG's controller, a memory unit is needed to store s_p , $\hat{q}_i(t)$, $\hat{q}_j(t)$, $j \in \mathcal{N}_i$, and Ω_p . Then, when computing z_+ , a calculation unit is needed to solve a quadratic equation with the new received $\hat{q}_j(t)$ (one of the neighboring DGs), which does not mandate too much computation resources. While the time reaching t_{k+1}^i , the DG calls for $\omega_i(t)$ and $\omega_j(t)$, $j \in \mathcal{N}_i$ to update $\hat{q}_i(t)$. At the same time, the controller sends $\hat{q}_i(t)$ to the neighboring DGs, which can be realized by a communication unit. Therefore, each controller only needs to integrate the memory module, calculation module and communication module to realize the self-triggered control.

B. Distributed Self-triggered Controller Design for Active Power Sharing

The distributed self-triggered active power sharing controller can be designed by following the similar procedure as in subsection A. Firstly, we denote by $p_i = m_i P_i$ for notation conciseness. Then, for DG i 's power sharing controller, the active power disagreement is defined by

$$q_i^p(t) = - \sum_{j \in \mathcal{N}_i} a_{ij} (p_i(t) - p_j(t)). \quad (37)$$

The control input is then designed by $u_i^p = k_p \hat{q}_i^p(t)$ with $\hat{q}_i^p(t) = q_i^p(t_{k+1}^i)$, $t \in [t_k^i, t_{k+1}^i)$.

Similarly, define the power sharing estimate error as $e_i^p(t) = \hat{q}_i^p(t) - q_i^p(t)$, we can obtain the following triggering time instant to achieve the power sharing objective in (4) with no Zeno behavior.

$$t_{k+1}^i = \min_{t > t_k^i} \{t \mid (e_i^p)^2(t) \geq \sigma_i^p (q_i^p)^2(t)\}. \quad (38)$$

The proof is omitted here since it is similar to the proof of Lemma 1. The only difference is the Lyapunov function should be changed into $W(t) = \mathbf{p}^T L \mathbf{p}$.

Then, following the same procedure of subsection A 2), we have the following theorem.

Theorem 2: For an islanded MG with N DGs, assume that the DGs are connected by an undirected connected communication network, then the DGs group will asymptotically achieve active power sharing objective in (5b) under the

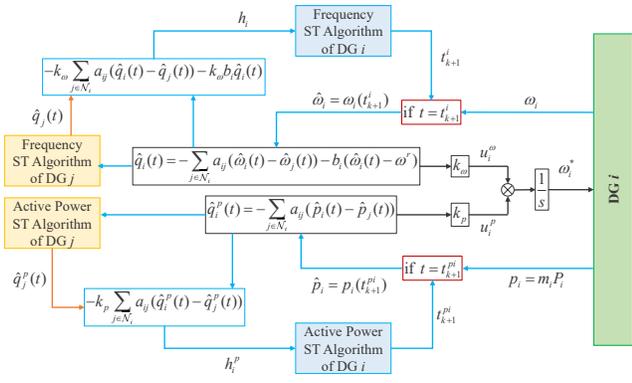


Fig. 3. The diagram of the frequency restoration and active power sharing distributed self-triggered controllers.

same self-triggered control procedure in Algorithm 1 with $h_i(t) = -k_p \sum_{j \in \mathcal{N}_i} a_{ij}(\hat{q}_i^p(t) - \hat{q}_j^p(t))$.

Proof: Following the same design procedure in subsection A 2), the distributed self-triggered controller for the power sharing can be designed by only change (23) into $h_i(t) = -k_p \sum_{j \in \mathcal{N}_i} a_{ij}(\hat{q}_i^p(t) - \hat{q}_j^p(t))$, then we can achieve the power sharing objective with the same procedure in Algorithm 1.

Accordingly, the frequency restoration and active power sharing secondary controls can be achieved based on self-triggered mechanism. The diagram of the controllers for DG i can be depicted as Fig. 3.

IV. SIMULATION RESULTS AND DISCUSSION

In this section, the effectiveness of the proposed distributed self-triggered secondary control algorithms are validated by simulation results. The test MG system with 6 DGs is illustrated in Fig. 4, in which the communication network is represented by blue dash lines. In the test MG system, the DGs are simulated as inverter-based DGs with inner voltage & current control loops, PWM mechanisms and LC filters in detail. The parameters of the test MG are provided in Table. I. Without loss of generality, the reference frequency of the islanded MG is set to be $\omega^r = 50$ Hz. The control gains are set to be $k_\omega = 30$ and $k_p = 20$, respectively. The parameters σ_i and σ_i^p are all set to be 0.9, $i \in \{1, 2, 3, 4, 5, 6\}$.

The simulation process is: 1) At $t = 0$ s, the MG works in islanded mode with only primary control. The total load is 50 kW at the beginning. 2) At $t = 1$ s, the proposed self-triggered secondary controller is activated. 3) At $t = 3$ s, the total load suddenly reduces 5 kW. 4) At $t = 5$ s, the total load suddenly increases 5 kW. The simulation time lasts 7 s in total.

A. Step Load Change

In this subsection, the effectiveness of the proposed distributed secondary control algorithms are verified by investigating the operation of the MG with step load change.

Fig. 5 shows the simulation results, in which Fig. 5 (a) exhibits the frequencies of DGs, Fig. 5 (b) exhibits the active power outputs of DGs. As observed, after the MG islanding at the beginning, the frequencies of DGs are soon synchronized

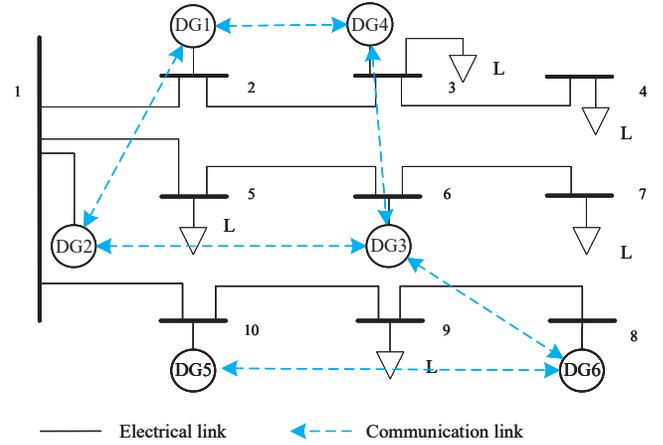


Fig. 4. The diagram of the test MG system.

TABLE I
PARAMETER SETTINGS OF THE TEST MG

	Line	Resistance	Reactance
Lines	Line 1,2	$R_{12} = 0.4 \Omega$	$X_{12} = 1.13 \Omega$
	Line 2,3	$R_{23} = 0.3 \Omega$	$X_{23} = 0.597 \Omega$
	Line 3,4	$R_{34} = 0.2 \Omega$	$X_{34} = 0.628 \Omega$
	Line 1,5	$R_{15} = 0.25 \Omega$	$X_{15} = 0.565 \Omega$
	Line 5,6	$R_{56} = 0.2 \Omega$	$X_{56} = 0.565 \Omega$
	Line 6,7	$R_{67} = 0.2 \Omega$	$X_{67} = 0.628 \Omega$
	Line 1,10	$R_{110} = 0.2 \Omega$	$X_{110} = 0.628 \Omega$
	Line 10,9	$R_{109} = 0.25 \Omega$	$X_{109} = 0.565 \Omega$
	Line 9,8	$R_{98} = 0.2 \Omega$	$X_{98} = 0.628 \Omega$
	DGs	DG 1	$P_{m1} = 20 \text{ kW}$
DG 2		$P_{m2} = 10 \text{ kW}$	$m_2 = 1 \times 10^{-4}$
DG 3		$P_{m3} = 20 \text{ kW}$	$m_3 = 5 \times 10^{-5}$
DG 4		$P_{m4} = 10 \text{ kW}$	$m_4 = 1 \times 10^{-4}$
DG 5		$P_{m5} = 20 \text{ kW}$	$m_5 = 5 \times 10^{-5}$
DG 6		$P_{m6} = 10 \text{ kW}$	$m_6 = 1 \times 10^{-4}$

(in 0.5 s) but are deviated from the reference value owing to the primary droop control. The active power of DGs are allocated inversely proportional to their droop coefficients, as $P_1 : P_2 : P_3 : P_4 : P_5 : P_6 = 2 : 1 : 2 : 1 : 2 : 1$ (the outputs of DG 1, DG 3 and DG 5 are stable at about 11.11 kW, and the outputs of DG 2, DG 4 and DG 6 are stable at about 5.55 kW). Then, the self-triggered secondary controls are applied at $t = 1$ s. It can be seen that the frequencies are gradually compensated to the reference value 50 Hz. Meanwhile, the active power of DGs are allocated to the same utilization profile accurately, which coincides with the droop control since we choose m_i in (4) to be the same as the corresponding droop coefficient. When the step load changes occur at $t = 3$ s, the frequencies of DGs can still be maintained at the reference value after a short transient process. The outputs of DGs decrease due to the load reduction while still maintain the same utilization profile

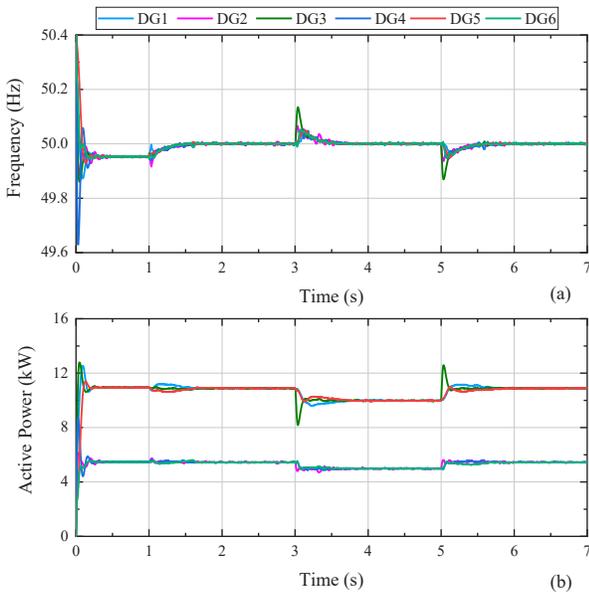


Fig. 5. Control performance under step load change: (a) the frequencies of DGs; (b) the active power outputs of DGs.

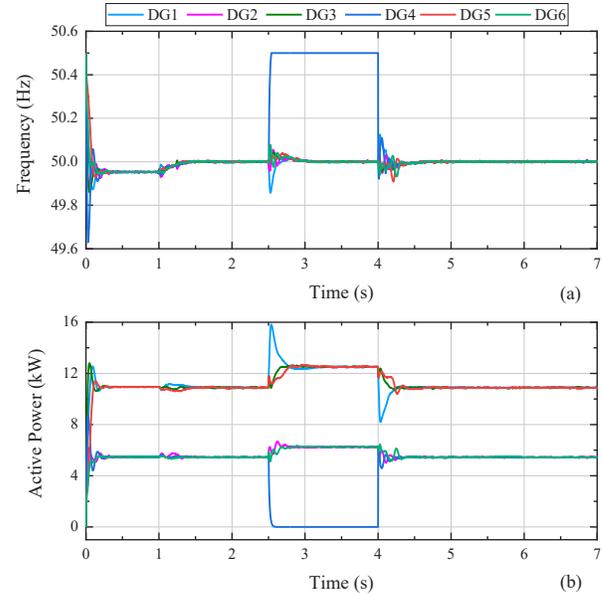


Fig. 7. Control performance under plug-and-play operation: (a) the frequencies of DGs; (b) the active power outputs of DGs.

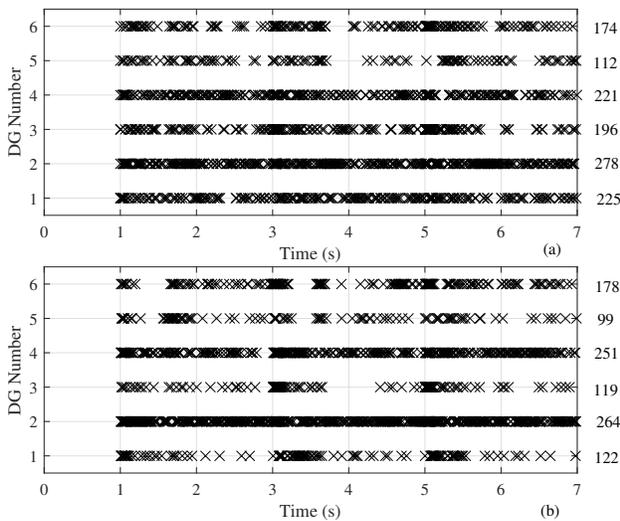


Fig. 6. Triggering time instants: (a) the triggering time instants of DGs' frequencies; (b) the triggering time instants of DGs' active power outputs.

(the outputs of DG 1, DG 3 and DG 5 are stable at about 10 kW, and the outputs of DG 2, DG 4 and DG 6 are stable at about 5 kW). And when the load increases 5 kW at $t = 5$ s, the proposed controllers can still compensate the frequency to 50 Hz and achieve the accurate power sharing with the same utilization profile.

Fig. 6 provides the triggering time instants of DGs' frequency and active power sharing controllers. The numbers in the right-hand side is the total triggering time of each DG's controller. It is observed that the triggers for each DG's controller are aperiodic and intermittent rather than consecutively. Moreover, each controller only triggers several

hundreds of times during the whole simulation process. From Fig. 6 we can see that the triggers are rather sparse, which means the controllers require quite few control updates and communications.

This scenario illustrates the effectiveness of the proposed distributed self-triggered frequency restoration and active power sharing control algorithms.

B. Plug-and-play Operation

In this subsection, we test the plug-and-play operation ability of the proposed distributed self-triggered control algorithms. In this simulation, the MG is islanded at $t = 0$ s and the secondary control is actuated at $t = 1$ s. Then, at $t = 2.5$ s, DG 4 gets disconnected from the MG, and is plugged back at $t = 4$ s. The total simulation time is set to be 6 s. The results are provided in Fig. 7 and Fig. 8.

It can be observed that, after DG 4 being plugged out at $t=2.5$ s, the active power of DG 4 soon decreases to 0. And its frequency increases to 50.5 Hz due to the primary droop control on idle-load operation. During 2.5s to 4 s, because of DG 4 getting disconnected, the load in the MG (50 kW) is supported only by DG 1, DG 2, DG 3, DG 5 and DG 6. Their outputs all increase to compensate the power shortage caused by the outage of DG 4. Whereas, because of the proposed self-triggered secondary controllers, the load is shared by DG 1, DG 2, DG 3, DG 5 and DG 6 according to the fair utilization profile as $P_1 : P_2 : P_3 : P_5 : P_6 = 2 : 1 : 2 : 2 : 1$ (the outputs of DG 1, DG 3 and DG 5 are stable at about 12.5 kW, and the outputs of DG 2 and DG 6 are stable at about 6.25 kW), and the frequencies of the connected DGs can also be synchronized to 50 Hz. While after DG 4 being plugged in, the MG operation is supported by all six DGs again. The MG returns to its original operating state before 2.5 s. As shown in Fig. 8, the triggers for all DGs' controllers are aperiodic and

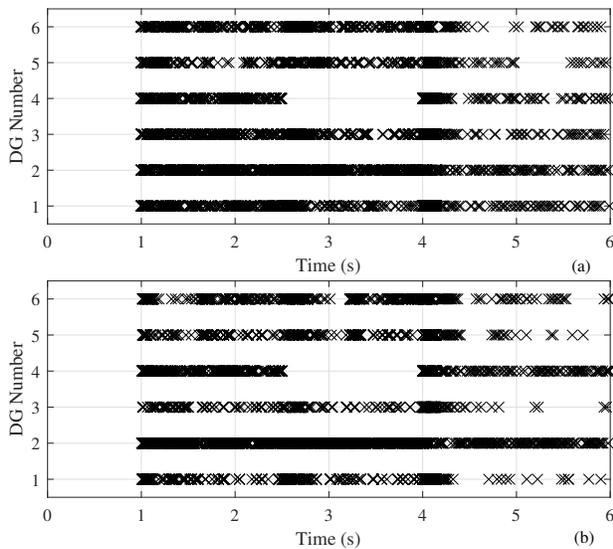


Fig. 8. Triggering time instants under plug-and-play operation: (a) the triggering time instants of DGs' frequencies; (b) the triggering time instants of DGs' active power outputs.

intermittent as well. When DG 4 is plugged out, its controllers do not trigger any more until DG 4 is plugged back at $t = 4.5$ s.

This simulation result illustrates that the proposed distributed self-triggered secondary controllers have plug-and-play operation ability and hence can provide more operation flexibility and scalability for DGs.

C. Operation with Communication Delay

Notice that the proposed self-triggered controllers are designed under the assumption of ideal communication, i.e., without time delay. However, communication delay may occur in practice. Recalling the principle of the self-triggered mechanism, the triggering time instant is determined when the stable condition is about to be violated. Time delay would cause the controller to trigger after the critical time of stable condition, which may make the system lose stability. In order to study the influence of communication delay on the self-triggered control systems preliminarily, in this subsection, we test the proposed self-triggered controllers with different communication delay.

Without loss of generality, we only provide the control performance of frequency restoration control due to the space limitation.

Fig. 9 illustrates the control performance of the proposed self-triggered frequency restoration controller with different communication delay during 1s to 2.5 s. The time delay in Fig. 9 (a) is set to be $\tau = 1ms$, in Fig. 9 (b) is $\tau = 5ms$, in Fig. 9 (c) is $\tau = 5ms$. We can see from Fig. 9 (a) and (b) that the responses of the controllers get more and more oscillatory as the communication delay increases. But, the system can maintain stability under short communication delay. Therefore, a shorter communication delay does not destroy the stability of the system, but it does reduce the control performance of the self-triggered controller.

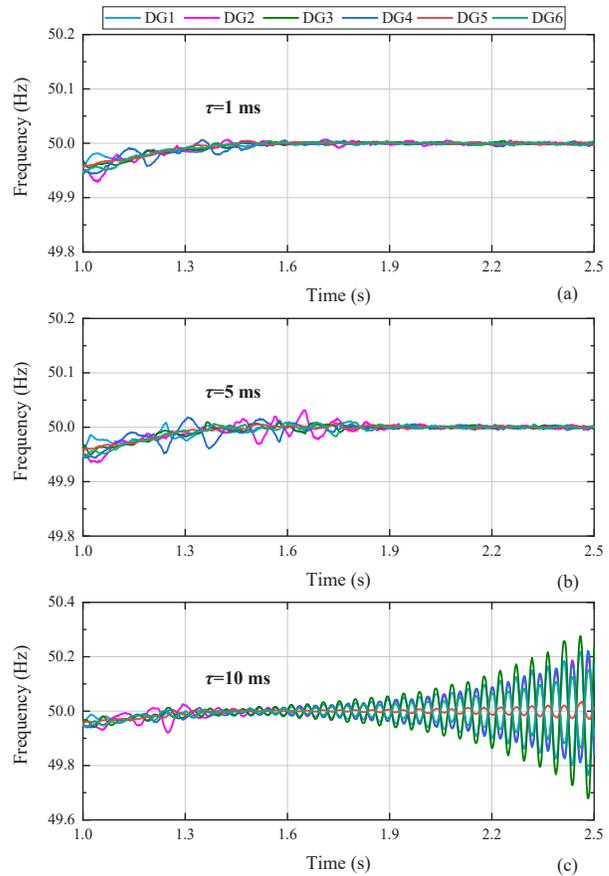


Fig. 9. Control performance of the proposed self-triggered frequency restoration controller with communication delay: (a) $\tau = 1ms$; (b) $\tau = 5ms$; (c) $\tau = 10ms$.

However, as shown in Fig. 9 (c), when the communication delay increases to 10 ms, the system loses its stability. In [40], it has been demonstrated that the traditional distributed secondary control system is able to endure the communication delay of tens of milliseconds. Thus, it is reasonable to state that the self-triggered mechanism reduces the stability margin of the distributed secondary control system. Stability analysis of distributed self-triggered secondary control is a complicated, interesting and worth studying issue, which we would like to discuss in our next paper.

D. Comparison with the Conventional Time-Triggered and the State-of-the-art Event-Triggered Controller

In this subsection, to further verify the effectiveness of the proposed self-triggered control, we compare the proposed self-triggered control with the conventional time-triggered control, and the state-of-the-art event-triggered control in [18]. For the sake of conciseness, we only provide the comparison results of the frequency restoration control as well.

The sampling period for the conventional controller and the controller in [18] is set to be 1 ms. The parameter σ_i for the triggering condition of the event-triggered controller in [18] is set to be 0.9 as well.

TABLE II
THE TOTAL NUMBER OF TRIGGERING TIME INSTANTS FOR DGs UNDER DIFFERENT CONTROLLERS

DGs	Number of times	Controllers		
		Conventional	event-triggered	self-triggered
DG 1	Trigger	5000	212	122
	Computation	5000	5000	198
DG 2	Trigger	5000	293	264
	Computation	5000	5000	421
DG 3	Trigger	5000	146	119
	Computation	5000	5000	208
DG 4	Trigger	5000	289	251
	Computation	5000	5000	408
DG 5	Trigger	5000	134	99
	Computation	5000	5000	163
DG 6	Trigger	5000	221	178
	Computation	5000	5000	305

Table. II depicts the total number of triggering times and computation times for each DGs under the proposed self-triggered controller, the conventional controller and the event-triggered controller in [18]. It can be clearly seen that the total number of triggering times for the event-triggered controller and the self-triggered controller are far more less than that of the conventional controller. This demonstrates the advantage of the proposed self-triggered controller for its ability to reduce communication requirement significantly. However, the total number of triggering times for the self-triggered controller and event-triggered controller are nearly the same. Therefore, the self-triggered controller does not have great advantages in reducing communication requirement compared with the event-triggered controller. But for the computation times, the conventional controller requires computing the control input consecutively, so each DG conducts 5000 times computation. The event-triggered controller needs to monitor the triggering condition all the time, thus it also needs to compute the triggering condition 5000 times for each DGs. While the self-triggered controller only requires computing z_+ 284 times on average. Although the computation times are larger than the triggering times due to the new states received from neighboring DGs in between the events, it is still far more less than that of the conventional controller and the event-triggered controller. In this sense, the self-triggered controller can reduce both communication and computation cost significantly.

E. Comparison with the State-of-the-art Self-Triggered Controller

To further verify the superiority of the proposed self-triggered control strategy, in this subsection, we provide a comparison result with the state-of-the-art self-triggered control strategy in [35]. For the sake of conciseness, we only provide the comparison results of the active power sharing control.

For the self-triggered active power sharing control in [35], it introduces a signum function $\text{sign}(\cdot)$ to prescribe the control step of the controller. In addition, a linear clock is designed based on the maximum value of a predefined convergence error ε and the current estimate error to determine the triggering time instants, hence this controller will result in a small

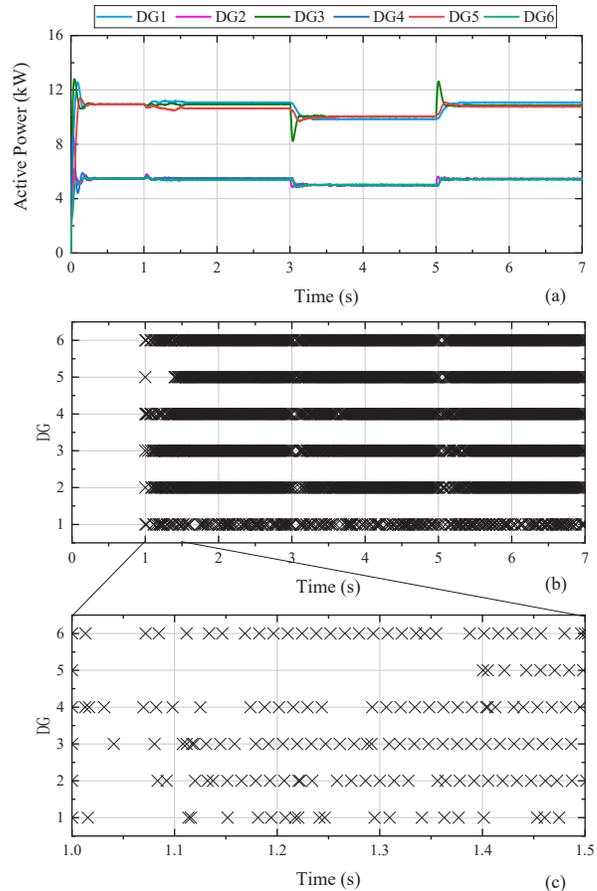


Fig. 10. Control performance of the self-triggered active power sharing controller in [35]: (a) the active power outputs of DGs; (b) the triggering time instants of DGs; (c) the enlarged detail of the triggering time instants of DGs.

convergence error. It is shown that the choice of ε prescribes the trade-off between the number of triggers and the accuracy of convergence. A larger ε means fewer triggers but a larger convergence error. For more details of this controller, one can refer to [35] or [31].

Fig. 10 illustrates the control performance of the self-triggered active power sharing control in [35]. As shown in Fig. 10 (a), the load can be allocated according to the fair utilization profile. However, compared with Fig. 5 (b), there exists a small convergence error at the steady states. Fig. 10 (b) exhibits the triggering time instants of DGs in the whole simulation process, Fig. 10 (c) shows the detailed triggering time instants of DGs during 1 s to 1.5 s. It can be seen that it also shows an aperiodic and intermittent triggering mode. However, as shown in Table. III, compared with our proposed self-triggered active power sharing control, the self-triggered active power sharing control in [35] generates more triggers and requires more computation for each controller. It is worth to point out that the triggering time and computation time of the self-triggered active power sharing control in [35] are equal because of the mechanism of the linear clock.

This simulation result illustrates the superiority of the proposed self-triggered control algorithm.

TABLE III
THE TOTAL NUMBER OF TRIGGERING TIME INSTANTS FOR THE PROPOSED SELF-TRIGGERED CONTROLLERS AND THE SELF-TRIGGERED CONTROLLERS IN [35]

DGs	Number of times	Controllers	
		the controller in [35]	the proposed controller
DG 1	Trigger	331	122
	Computation	331	198
DG 2	Trigger	488	264
	Computation	488	421
DG 3	Trigger	416	119
	Computation	416	208
DG 4	Trigger	413	251
	Computation	413	408
DG 5	Trigger	397	99
	Computation	397	163
DG 6	Trigger	421	178
	Computation	421	305

V. CONCLUSIONS

In this paper, we have developed distributed self-triggered control algorithms for frequency restoration and active power sharing of islanded MGs. It has been demonstrated that the proposed self-triggered controllers not only can reduce the communication cost of DGs, but also reduce the computation burden significantly, which highly improves the efficiency of the secondary control system in MGs. The controllers are fully distributed, and thus can support the plug-and-play operation of DGs. Compared with the existing distributed self-triggered secondary controller, the proposed controllers show better performance on the accuracy of convergence and the communication and computation cost reduction. Future work includes the analysis of the proposed self-triggered secondary controller under communication delay.

REFERENCES

- [1] C. Wang, *Analysis and Simulation Theory of Microgrids*. Beijing: Science Press, 2013.
- [2] Y. Khayat, Q. Shafiee, R. Heydari, M. Naderi, T. Dragievi, J. W. Simpson-Porco, F. Drfler, M. Fathi, F. Blaabjerg, and J. M. a. Guerrero, "On the secondary control architectures of ac microgrids: An overview," *IEEE Transactions on Power Electronics*, vol. 35, no. 6, pp. 6482–6500, 2020.
- [3] H. Hui, Y. Chen, S. Yang, H. Zhang, and T. Jiang, "Coordination control of distributed generators and load resources for frequency restoration in isolated urban microgrids," *Applied Energy*, vol. 327, p. 120116, 2022.
- [4] J. M. Guerrero, J. C. Vasquez, J. Matas, L. G. de Vicuna, and M. Castilla, "Hierarchical control of droop-controlled ac and dc microgrids—a general approach toward standardization," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 1, pp. 158–172, 2011.
- [5] A. Bidram and A. Davoudi, "Hierarchical structure of microgrids control system," *IEEE Transactions on Smart Grid*, vol. 3, no. 4, pp. 1963–1976, 2012.
- [6] K. T. Tan, X. Y. Peng, P. L. So, Y. C. Chu, and M. Z. Q. Chen, "Centralized control for parallel operation of distributed generation inverters in microgrids," *IEEE Transactions on Smart Grid*, vol. 3, no. 4, pp. 1977–1987, 2012.
- [7] Z. Qu, *Cooperative control of dynamical systems: applications to autonomous vehicles*. London, U.K.: Springer-Verlag, 2009.
- [8] H. Xin, Z. Qu, J. Seuss, and A. Maknouninejad, "A self-organizing strategy for power flow control of photovoltaic generators in a distribution network," *IEEE Transactions on Power Systems*, vol. 26, no. 3, pp. 1462–1473, 2010.

- [9] A. Bidram, A. Davoudi, F. L. Lewis, and Z. Qu, "Secondary control of microgrids based on distributed cooperative control of multi-agent systems," *IET Generation, Transmission & Distribution*, vol. 7, no. 8, pp. 822–831, 2013.
- [10] C. Li, Z. Qu, D. Qi, and F. Wang, "Distributed finite-time estimation of the bounds on algebraic connectivity for directed graphs," *Automatica*, vol. 107, pp. 289–295, 2019.
- [11] Y. Wang, X. Wang, Z. Chen, and F. Blaabjerg, "Distributed optimal control of reactive power and voltage in islanded microgrids," *IEEE Transactions on Industry Applications*, vol. 53, no. 1, pp. 340–349, 2017.
- [12] S. Abhinav, H. Modares, F. L. Lewis, and A. Davoudi, "Resilient cooperative control of dc microgrids," *IEEE Transactions on Smart Grid*, vol. 10, no. 1, pp. 1083–1085, 2018.
- [13] Y. Chen, D. Qi, H. Dong, C. Li, Z. Li, and J. Zhang, "A fdi attack-resilient distributed secondary control strategy for islanded microgrids," *IEEE Transactions on Smart Grid*, vol. 12, no. 3, pp. 1929–1938, 2021.
- [14] A. Bidram, B. Poudel, L. Damodaran, R. Fierro, and J. M. Guerrero, "Resilient and cybersecure distributed control of inverter-based islanded microgrids," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 6, pp. 3881–3894, 2020.
- [15] M. Chen, X. Xiao, and J. M. Guerrero, "Secondary restoration control of islanded microgrids with a decentralized event-triggered strategy," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 9, pp. 3870–3880, 2018.
- [16] S. Weng, D. Yue, C. Dou, J. Shi, and C. Huang, "Distributed event-triggered cooperative control for frequency and voltage stability and power sharing in isolated inverter-based microgrid," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 49, no. 4, pp. 1427–1439, 2019.
- [17] D. V. Dimarogonas, E. Frazzoli, and K. H. Johansson, "Distributed event-triggered control for multi-agent systems," *IEEE Transactions on Automatic Control*, vol. 57, no. 5, pp. 1291–1297, 2012.
- [18] Y. Chen, C. Li, D. Qi, Z. Li, Z. Wang, and J. Zhang, "Distributed event-triggered secondary control for islanded microgrids with proper trigger condition checking period," *IEEE Transactions on Smart Grid*, vol. 13, no. 2, pp. 837–848, 2022.
- [19] Y. Fan, G. Hu, and M. Egerstedt, "Distributed reactive power sharing control for microgrids with event-triggered communication," *IEEE Transactions on Control Systems and Technology*, vol. 25, no. 1, pp. 118–128, 2017.
- [20] Y. Chen, D. Qi, and Z. Li, "Distributed event-triggered control for frequency restoration in islanded microgrids with reduced trigger condition checking," *CSEE Journal of Power and Energy System*, vol. 1, no. 1, pp. 1–1, 2021.
- [21] T. Qian, Y. Liu, W. Zhang, W. Tang, and M. Shahidehpour, "Event-triggered updating method in centralized and distributed secondary controls for islanded microgrid restoration," *IEEE Transactions on Smart Grid*, vol. 11, no. 2, pp. 1387–1395, 2020.
- [22] Y. Xie and Z. Lin, "Distributed event-triggered secondary voltage control for microgrids with time delay," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 49, no. 8, pp. 1582–1591, 2019.
- [23] B. Abdolmaleki, Q. Shafiee, M. M. Arefi, and T. Dragičević, "An instantaneous event-triggered hz-watt control for microgrids," *IEEE Transactions on Power Systems*, vol. 34, no. 5, pp. 3616–3625, 2019.
- [24] K. Mohammadi, E. Azizi, J. Choi, M.-T. Hamidi-Beheshti, A. Bidram, and S. Bolouki, "Asynchronous periodic distributed event-triggered voltage and frequency control of microgrids," *IEEE Transactions on Power Systems*, vol. 36, no. 5, pp. 4524–4538, 2021.
- [25] Y. Wang, T. L. Nguyen, Y. Xu, Z. Li, Q. Tran, and R. Caire, "Cyber-physical design and implementation of distributed event-triggered secondary control in islanded microgrids," *IEEE Transactions on Industry Applications*, vol. 55, no. 6, pp. 5631–5642, 2019.
- [26] L. Xing, Q. Xu, F. Guo, Z.-G. Wu, and M. Liu, "Distributed secondary control for dc microgrid with event-triggered signal transmissions," *IEEE Transactions on Sustainable Energy*, vol. 12, no. 3, pp. 1801–1810, 2021.
- [27] B. Abdolmaleki, Q. Shafiee, A. R. Seifi, M. M. Arefi, and F. Blaabjerg, "A zero-free event-triggered secondary control for ac microgrids," *IEEE Transactions on Smart Grid*, vol. 11, no. 3, pp. 1905–1916, 2020.
- [28] J. Choi, S. I. Habibi, and A. Bidram, "Distributed finite-time event-triggered frequency and voltage control of ac microgrids," *IEEE Transactions on Power Systems*, vol. 37, no. 3, pp. 1979–1994, 2022.
- [29] Y.-Y. Qian, A. V. P. Premakumar, Y. Wan, Z. Lin, Y. A. Shamash, and A. Davoudi, "Dynamic event-triggered distributed secondary control of dc microgrids," *IEEE Transactions on Power Electronics*, vol. 37, no. 9, pp. 10226–10238, 2022.
- [30] F. Han, X. Lao, J. Li, M. Wang, and H. Dong, "Dynamic event-triggered protocol-based distributed secondary control for islanded microgrids,"

International Journal of Electrical Power & Energy Systems, vol. 137, p. 107723, 2022.

- [31] C. De Persis and P. Frasca, "Robust self-triggered coordination with ternary controllers," *IEEE Transactions on Automatic Control*, vol. 58, no. 12, pp. 3024–3038, 2013.
- [32] Y. Fan, L. Liu, G. Feng, and Y. Wang, "Self-triggered consensus for multi-agent systems with zero-free triggers," *IEEE Transactions on Automatic Control*, vol. 60, no. 10, pp. 2779–2784, 2015.
- [33] X. Yi, K. Liu, D. V. Dimarogonas, and K. H. Johansson, "Dynamic event-triggered and self-triggered control for multi-agent systems," *IEEE Transactions on Automatic Control*, vol. 64, no. 8, pp. 3300–3307, 2019.
- [34] J. Peng, B. Fan, H. Xu, and W. Liu, "Discrete-time self-triggered control of dc microgrids with data dropouts and communication delays," *IEEE Transactions on Smart Grid*, vol. 11, no. 6, pp. 4626–4636, 2020.
- [35] G. Xu and L. Ma, "Resilient self-triggered control for voltage restoration and reactive power sharing in islanded microgrids under denial-of-service attacks," *Applied Sciences*, vol. 10, no. 11, p. 3780, 2020.
- [36] Y. Wang, C. Deng, D. Liu, Y. Xu, and J. Dai, "Unified real power sharing of generator and storage in islanded microgrid via distributed dynamic event-triggered control," *IEEE Transactions on Power Systems*, vol. 36, no. 3, pp. 1713–1724, 2021.
- [37] A. Bidram, F. L. Lewis, and A. Davoudi, "Distributed control systems for small-scale power networks: Using multiagent cooperative control theory," *IEEE Control systems magazine*, vol. 34, no. 6, pp. 56–77, 2014.
- [38] A. Bidram, A. Davoudi, F. L. Lewis, and J. M. Guerrero, "Distributed cooperative secondary control of microgrids using feedback linearization," *IEEE Transactions on Power Systems*, vol. 28, no. 3, pp. 3462–3470, 2013.
- [39] M. Tahir and S. K. Mazumder, "Self-triggered communication enabled control of distributed generation in microgrids," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 2, pp. 441–449, 2015.
- [40] L. Xu, Q. Guo, Z. Wang, and H. Sun, "Modeling of time-delayed distributed cyber-physical power systems for small-signal stability analysis," *IEEE Transactions on Smart Grid*, vol. 12, no. 4, pp. 3425–3437, 2021.

Yulin Chen (M' 22) received the Ph.D. degree in Electrical Engineering from Zhejiang University, Hangzhou, China, in 2021. He was a Postdoctoral Fellow with University of Macau, Macau, China, from September 2021 to September 2022. He is currently an Associate Research Fellow with Hainan Institute of Zhejiang University, Sanya, China. His current research interests include distributed control of renewable energy and cyber-physical security with application in smart grid.



Keng-Weng Lao (S'09, M'17, SM'20) received his B.Sc., M.Sc., and Ph.D. degrees in electrical and electronics engineering from the Faculty of Science and Technology, University of Macau, Macau, China, in 2009, 2011, and 2016, respectively. He is currently an assistant professor with the Department of Electrical and Computer Engineering, as well as the State Key Laboratory of Internet of Things for Smart City, University of Macau. He was a Research Scholar in the Department of Electrical and Computer Engineering, The University of Texas at

Austin, Austin, TX, USA from June 2017 to June 2019. His research interests include cyber-physical security, renewable energy integration, energy internet of things, smart energy system protection and smart grid. He was the recipient of the Macao Science and Technology Development Fund Postgraduate Award for Ph.D. Student 2016, first runner-up of the Challenge Cup National Inter-university Science and Technology Competition 2013, and Championship of the Postgraduate Section in the IET Young Professionals Exhibition 2013.



and Economics, and the Journal of Robotics, Networking and Artificial Life.

Donglian Qi (M' 19, SM' 21) received her Ph.D. degree in control theory and control engineering from Zhejiang University, Hangzhou, China, in March 2002. Since then, she has been with the College of Electrical Engineering, Zhejiang University where she is currently a Professor. Her current research interests include the basic theory and application of cyber physical power system (CPPS), digital image processing, artificial intelligence, and electric operation and maintenance robots. She is an Editor for the Clean Energy, the IET Energy Conversion



demand response, and Internet of Things technologies for smart energy.

Hongxun Hui (S'17, M'20) received the B.E. and Ph.D. degrees in electrical engineering from Zhejiang University, Hangzhou, China, in 2015 and 2020, respectively. From 2018 to 2019, he was a visiting scholar at the Advanced Research Institute in Virginia Tech, and the CURENT Center in University of Tennessee. He is currently a Research Assistant Professor with the State Key Laboratory of Internet of Things for Smart City, University of Macau, Macao SAR, China. His research interests include optimization and control of power system,



Shaohua Yang (S'19) received the B.S. degree from Hefei University of Technology, Hefei, China, in 2017 and the M.S. degree from Fuzhou University, Fuzhou, China, in 2020, both in electrical engineering. He is currently working toward the Ph.D. degree at University of Macau, Macao SAR, China. His research interests include cyber physical system, demand response, and power quality.

Yunfeng Yan received the Ph.D. degree in electrical engineering from Zhejiang University, Hangzhou, China, in 2019. She is currently an Associate Research Fellow with Zhejiang University. Her research interests include computer vision and machine learning systems, and distributed estimation and control of networked systems.



Yi Zheng received the Ph.D. degree in Chinese philology and linguistics from Zhejiang University, Hangzhou, China, in June 2019. He is currently a postdoctoral fellow with Zhejiang University. His research interests include computer vision, palaeography and phonology of ancient Chinese.

