Implementation of Adaptive Partial Discharge Denoising in Resource-Limited Embedded Systems via Efficient Time-Frequency Matrix Factorization

Implementation of Adaptive Partial Discharge De-noising in Resource-limited Embedded Systems via Efficient Time-frequency Matrix Factorization

Yuan Yan（颜源）, Riccardo Trinchero, *Member*, *IEEE*, Igor Simone Stievano, *Senior Member*, *IEEE*, and Hongjie Li（李洪杰）, *Member*, *IEEE*

***Abstract*—As low-cost Internet-of-Thing-based PD monitors for medium-voltage apparatuses in distribution power systems increase, developing an effective PD de-noising algorithm is crucial to improve their robustness to on-site noise. Yet, de-noising PD signals in the monitoring devices is challenging primarily due to three critical reasons, i.e., high-level field noises, uncertain PD waveform, and limited computing resources. This work describes an adaptive and efficient PD de-noising algorithm based on the improved spectral decomposition of the noisy PD signal. PD pulses are accurately extracted from the noisy signal by selecting the dominant components via a low-rank singular value decomposition of the time-frequency spectrogram of the signal, thus reducing the size of the involved matrices and the computational complexity. The performance of the proposed de-noising algorithm is first demonstrated on a synthetic PD signal and compared with state-of-the-art alternatives implemented on three embedded systems commonly used for PD monitoring. Finally, the strength and the effectiveness of the proposed approach are further validated on experimental data based on the measurement of Internet-of-Thing-based PD monitors for 35-kV switchgears.**

***Index Terms*— Partial discharge de-noising, embedded systems, short-time Fourier transform, randomized singular value decomposition, kurtosis criteria.**

I. INTRODUCTION

A. Background

Partial discharge (PD) monitoring is a promising technique for assessing the health of electrical insulation within high-voltage apparatuses, capable of revealing incipient insulation failures [1, 2]. With the rapid development of advanced internet techniques, e.g., Internet-of-Thing (IoT), Edge Computing, etc., a mass of low-cost IoT-based PD monitoring devices (e.g., using a microcontroller with the Cortex-M4 core [3]) have been deployed on power apparatuses in power distribution networks, transportation systems, petrochemical plants, etc. [4], and they have successfully predicted many insulation failures.

Nevertheless, many application experiences show that the sensitivity and accuracy of PD diagnosis are significantly reduced by field interferences, with unavoidable detrimental effects induced by discrete spectral interference and white noise. Discrete spectral interference mainly arises from higher harmonics of the power system, high-frequency protection signals, carrier frequency signals (e.g., power line communication), and radio signals [5]. White noise is caused by equipment thermal noise, ground noise, rand noise, etc. [6].

Yet, de-noising PD signals in IoT-based monitoring devices with minimal computing resources is challenging primarily due to three critical reasons. The first one is that the magnitude of field noises (i.e., mV level) is often the same as or even higher than that of PD signals [5, 6], causing PD signals to be wholly drowned in noises. The second one is that the PD waveforms are always various, which depends on the size, location, and materials of the PD source, the transfer functions of the PD propagation path, and the used PD sensors [7, 8]. Such uncertainties on PD signal waveforms pose great difficulty in setting the algorithms' proper de-noising parameters. The last and most important one is that the hardware resources in the low-cost PD monitoring devices are minimal, e.g., the clock frequency and memory of the Cortex-M4 core used in [2,3] are only 80 MHz and Kbytes, leading to possible data-overrun error or too long computation time and too much power consumption of the de-noising algorithms. Therefore, developing an adaptive and efficient de-noising algorithm that can be embedded into PD monitors with minimal computing resources becomes vital.

B. Related works

In the past twenty years, various PD signal de-noising methods have been proposed to reduce or eliminate noise. Wavelet transform (WT) has demonstrated excellent performance in de-noising PD signals with high noise levels in cases where the mother wavelet and decomposition levels have been chosen appropriately [9]. However, identifying the ideal mother wavelet and decomposition level requires prior knowledge of the waveform characteristics of the PD signal, which is often difficult to determine in on-site measurements. To mitigate the above limitation, the paper [10] presents an adaptive empirical mode decomposition (EMD) algorithm in

This work was supported in part by China Scholarship Council. (corresponding author: Yuan Yan, y220928y@stu.xjtu.edu.cn)

Yuan Yan is with State Key Laboratory of Insulation and Power Equipment, School of the Electrical Engineering, Xi'an Jiaotong University, Xi'an, 710049, China, and also with Department of Electronics and Telecommunications, Politecnico di Torino, Turin, 10129, Italy. (y220928y@stu.xjtu.edu.cn).

Riccardo Trinchero, and Igor Simone Stievano are with the Department of Electronics and Telecommunications, Politecnico di Torino, Turin, Italy. (e-mail: riccardo.trinchero@polito.it; igor.stievano@polito.it).

Hongjie Li are with State Key Laboratory of Insulation and Power Equipment, School of the Electrical Engineering, Xi'an Jiaotong University, Xi'an, 710049, China. (e-mail: hjli@mail.xjtu.edu.cn)

which no prior knowledge is required. But, such a method suffers from mode-mixing issues. Variational mode decomposition (VMD) addresses this specific challenge [11], and its effectiveness has been successfully demonstrated in de-noising PD [12]. Nevertheless, iterations are required in the VMD algorithm, leading to a non-negligible computational cost. [13] explores using the machine learning method for PD de-noising, which is proven to extract PDs automatically in various noise environments. However, the technique requires lots of labeled data to train the de-noising model.

In recent years, singular value decomposition (SVD), a non-parametric and self-adaptive method, has been explored for PD de-noising. In [14], the PD signal is embedded into a Hankel matrix, and then PD de-noising is achieved via applying SVD and principal component analysis to the Hankel matrix. However, the survey in [15] reveals that the Hankel matrix-based SVD suffers from poor de-noising performance in a low signal-to-noise environment due to the difficulty of selecting the correct number of singular values. In [16], a short-time SVD algorithm is developed by reducing the rank of the input matrix to be decomposed and developing an automatic singular value selection approach based on the minimum description length criteria. A few hybrid methods that combine SVD with WT, EMD, and VMD are presented in [17, 18, 19]. However, these methods suffer from the same problems as WT, EMD, and VMD. In [20], a PD de-noising based on the so-called generalized S-transform and module time-frequency matrix is proposed, but the S-transform is expensive and leads to significant computational time. [21] offers an alternative solution that combines the SVD and the time-frequency complex matrix (or spectrogram) of a noisy PD signal obtained by the short-time Fourier transform (STFT). The STFT-SVD method can adaptively reduce noise in various PD pulses. However, the technique, as well as the above other algorithms, is time-consuming due to the expensive SVD, making it challenging to apply in low-cost PD monitors with minimal computing resources, e.g., the Cortex-M4 core.

C. Contributions and paper organization

The current study was motivated by the limitations of the above-mentioned PD de-noising methods in hardware resource-limited PD monitors, offering the following improvements:

1) **Algorithm improvement.** An adaptive and efficient STFT-randomized SVD (STFT-RSVD) algorithm is developed based on the previous work in [21] with the aim of heavily reducing the computational complexity of the proposed algorithm, including PD identification criteria to avoid unnecessary computation, decomposing the spectrogram using a more efficient RSVD algorithm, and selecting dominant components via a more straightforward approach.
2) **Hardware comparison and selection.** Embedded systems commonly used for PD monitoring are surveyed, and the implementation of the proposed algorithm in these systems is described in detail. To the best of our knowledge, no reports or publications addressed a systematic comparative study on implementing PD noise reduction in embedded systems used in PD monitors.
3) **Validation by simulation.** Simulation experiments of de-noising a representative synthetic PD signal are conducted based on the selected embedded systems. The effectiveness and feasibility of the proposed algorithm are thus validated, along with the discussion of the effects and sensitivity of its key parameters.
4) **Validation by measurements.** The proposed algorithm is successfully applied to de-noise real PD signals with an extremely low signal-to-noise ratio, measured by IoT-based PD monitors of medium-voltage switchgears.
5) **Cross-comparison and proposed proven solution.** A comparison with alternative state-of-the-art solutions is conducted in synthetic and measured PD signals, demonstrating the proposed algorithm's better de-noising performance and lower computational cost.

The rest of this paper is structured as follows: Section II introduces the principles of the improved STFT-RSVD de-noising algorithm. Section III describes the implementation of PD de-noising in embedded systems. Section IV discusses the simulation results of de-noising a synthetic signal in the embedded systems. In Section V, the feasibility of the developed algorithm is confirmed by applying it to de-noise PD data in real resource-limited PD monitors. Finally, conclusions and final remarks are drawn in Section VI.

II. Principles of the Improved STFT-RSVD De-noising Algorithm

This section describes the principles of the proposed STFT-RSVD de-noising algorithm. To explain the algorithm better, the previous STFT-SVD de-noising method is briefly reviewed, and then three improvements of the proposed algorithm in terms of computational complexity reduction are described in detail.

A. The STFT-SVD de-noising method

Figure 1 collects the principle of the STFT-SVD method based on a four-step procedure as follows [21]:

Step 1) The time-domain signal is first transformed into a time-frequency spectrogram via STFT.

Step 2) The spectrogram is preprocessed via a soft-masking function to reduce part of the white noise, and then it is decomposed into multiple components via SVD.

Step 3) Each component is transformed back to time-domain sub-signals via inverse STFT.

Step 4) The sub-signals with distinct pulsed characteristics are selected via Principal Component Analysis and kurtosis criteria, and they are finally summed to reconstruct the de-noised PD signal.

The STFT-SVD de-noising method has been validated to perform outstandingly in extremely low signal-to-noise ratio environments. However, the computational complexity of the STFT-SVD algorithm is too high to be executed in most embedded systems due to the two following reasons: i) the SVD algorithm is costly as it leads to colossal computational complexity and requires much memory; ii) The ISTFT requires to be used multiple times to transform all selected components (i.e., sub-spectrogram) back to time-domain signals (see the third step in Fig.1), leading to a huge computation.

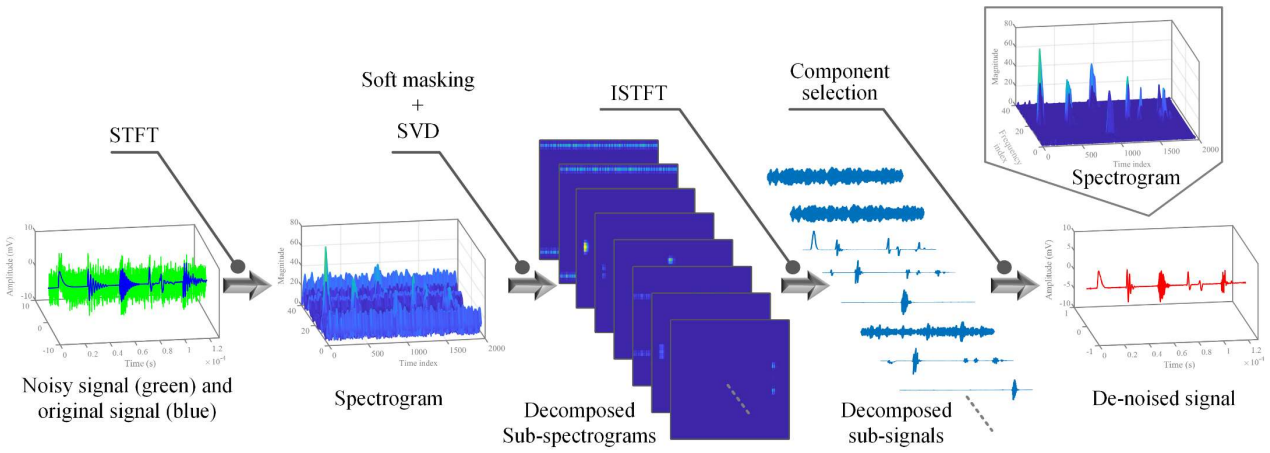


Fig. 1. Illustration of the basic principle of the STFT-SVD de-noising method in [21].

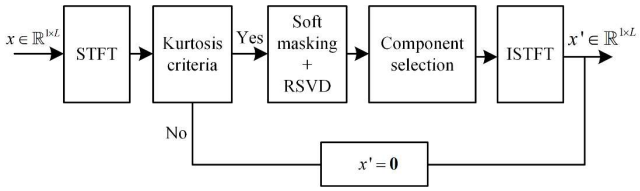


Fig. 2. Block diagram of the improved STFT-RSVD algorithm.

B. The proposed STFT-RSVD de-noising algorithm

To reduce the computational complexity of the STFT-SVD algorithm, the more efficient STFT-RSVD algorithm is developed, as shown in Fig. 2. Compared with the STFT-SVD algorithm, the proposed algorithm has three improvements:

- Kurtosis criteria is added between the STFT and SVD steps, aiming at avoiding applying the expensive SVD to the useless data without PDs.
- RSVD is used to decompose the time-frequency spectrogram automatically. Compared with the SVD, the RSVD requires less memory, leading to significantly smaller computational complexity.
- The component selection is carried out before the ISTFT step to avoid executing the computationally expensive ISTFT multiple times, significantly reducing the computational complexity of this step.

Based on the three improvements, the following paragraphs describe the proposed STFT-RSVD algorithm in detail according to the scheme in Fig. 2 and data processing following all the blocks involved from left to right.

1) Time-frequency transform via STFT (step 1)

The sampled noisy PD signal is stored in a vector $\boldsymbol{x} = [x(t_1), x(t_2), \dots, x(t_L)]^T$ defined as:

$$x(t_l) = s(t_l) + w(t_l), t_l = (l-1)\Delta t, l = 1,2,\dots,L, \quad (1)$$

where $s(t_l)$ is the discretized noise-free PD signal, $w(t_l)$ represents the superimposed noise (i.e., white or discrete spectrum noises), Δt is the sampling interval, and L is the number of time samples.

The STFT of the signal $\boldsymbol{x}$ is computed by sliding an analysis window $g(n)$ (e.g., a Gaussian window with a width factor of 2.5 in this work) of length M over the signal and calculating the fast Fourier transform of each segment of windowed data. The window hops over the original signal at intervals of H samples. The STFT matrix has $N = \lfloor (K - M + H)/H \rfloor$ columns, where the $\lfloor \cdot \rfloor$ symbol denotes the floor function. The n^{th} column of the STFT matrix $\boldsymbol{X} = [\boldsymbol{X}_1, \boldsymbol{X}_2, \dots, \boldsymbol{X}_n, \dots, \boldsymbol{X}_N]$ ($\boldsymbol{X} \in \mathbb{C}^{M\times N}$) contains the fast Fourier transform of the windowed data centered about time nH:

$$X_n(f_m) = \sum_{l=1}^{L} x(t_l)\, g(t_l - nH\Delta t) e^{-j2\pi t_l f_m} \Delta t, \quad (2)$$

where $f_m = (2m - M)/(2M\Delta t)$ ($m = 1,2,\dots,M$) are the discrete frequency, and $X_n(f_m)$ (also labeled as $X_{m,n}$) is the element in the m^{th} row and n^{th} column of the matrix $\boldsymbol{X}$.

2) Pulse identification via kurtosis criteria (step 2)

PD pulses have a short duration, leading to outliers in the local area of the spectrogram $\boldsymbol{X}$. At the same time, white and discrete spectral interferences tend to lack outliers in the time dimension (i.e., each row of the spectrogram). Such a statistical difference between PD pulses and the noises allows us to identify whether one PD pulse occurs by detecting these outliers. Since the real parts of the spectrogram have outliers similar to those of the spectrogram, PD identification can be achieved by applying kurtosis criteria directly to the real parts of the rows of the obtained spectrogram to reduce computational cost. The criteria is formulated as follows:

$$\begin{cases} \text{yes}, & \left(\sum_{i=1}^{M} \text{logical}(\text{kur}(\boldsymbol{X}_i^{\text{re}}) > \beta)\right) \geq \varepsilon \\ \text{no}, & \left(\sum_{i=1}^{M} \text{logical}(\text{kur}(\boldsymbol{X}_i^{\text{re}}) > \beta)\right) < \varepsilon \end{cases}, \quad (3)$$

where $\boldsymbol{X}_i^{\text{re}}$ is the real parts of the ith row of the spectrogram, M is the total number of rows of the spectrogram, β is a threshold set to 4, an empirical value suggested in [22], ε is a threshold to identify whether a PD occurs and is better to be selected as 2~5 since too small or large ε may lead to wrong or missed PD identification, kur$(\cdot)$ is the kurtosis calculation function given in [18], and logical$(\cdot)$ is the logical function and returns 0 or 1.

3) Signal enhancement via soft masking (step 3)

As shown in the spectrogram in Fig. 1, the PD pulses turn out to be localized in the specific zones and have significantly higher amplitude than the white noise. Such difference between the PD signal and the white noise allows us to pre-process the matrix $\boldsymbol{X} \in \mathbb{C}^{M\times N}$ to remove parts of the white noise via a soft masking function as [21,23]:

$$X_{m,n} = \begin{cases} T_{m,n} \cdot X_{m,n} & \text{if } |X_{m,n}| < 3\varepsilon \\ X_{m,n} & \text{if } |X_{m,n}| \geq 3\varepsilon \end{cases}, \quad (4)$$

where ε is the standard deviation of $\boldsymbol{X}$, and $T_{m,n}$ is the attenuation coefficients defined as:

$$T_{m,n} = \left| \frac{X_{m,n}}{3\varepsilon} \right|^q, \quad (5)$$

where q is the attenuation control factor, and it is set as 2, an empirical value suggested in [21]. After the soft masking, the spectrogram $\boldsymbol{X}$ is updated to a new spectrogram, in which parts of white noise are removed while discrete spectrum noise and some residual white noise still exist. It is essential to point out that soft masking improves the signal-to-noise ratio of the PD signal, thus facilitating the subsequent matrix factorization and component selection.

4) Matrix factorization via randomized SVD (step 4)

Since the elements belonging to the PD signal always concentrate at the local area of the spectrogram $\boldsymbol{X}$, we can perform low-rank approximation to reduce the size of the input matrix of the expensive SVD, which can be achieved via the randomized algorithm described in [24]. The randomized algorithm and SVD combination are named randomized SVD (RSVD) [25]. The noisy input matrix $\boldsymbol{X} \in \mathbb{C}^{M \times N}$ is decomposed into four matrices via RSVD, yielding:

$$\boldsymbol{X} = \boldsymbol{Q}\boldsymbol{U}_{\mathrm{Q}}\boldsymbol{S}_{\mathrm{Q}}\boldsymbol{V}_{\mathrm{Q}}^{H}, \quad (6)$$

where $\boldsymbol{Q} \in \mathbb{C}^{M \times P}$ is a given matrix with P ($P \ll M$, N) orthonormal columns which can be obtained via the randomized algorithm [25], $\boldsymbol{U}_{\mathrm{Q}}$ is a $P \times P$ left orthonormal matrix ($\boldsymbol{U}_{\mathrm{Q}}=[\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_P]$, $\boldsymbol{u}_i \in \mathbb{C}^{P \times 1}$), $\boldsymbol{S}_{\mathrm{Q}}$ is a real $P \times P$ rectangular matrix with the singular values $\sigma_1 > \sigma_2 > \cdots > \sigma_P$ in the diagonal entries, and $\boldsymbol{V}_{\mathrm{Q}}$ is a $P \times N$ right orthonormal matrix ($\boldsymbol{V}_{\mathrm{Q}} = [\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_P]$, $\boldsymbol{v}_i \in \mathbb{C}^{N \times 1}$). The total computational complexity of the RSVD is $\mathrm{O}(4MN\log P + 4(M+N)P^2)$ [25], which is much less than that of the classical SVD (i.e., $\mathrm{O}(4NM^2)$) as P can be set much less than M and N.

In RSVD, only one parameter (i.e., P) is required to be set in advance. On the one hand, we want the basis matrix $\boldsymbol{Q}$ to contain as few columns as possible to reduce the computational complexity. On the other hand, accurately approximating the input matrix is even more critical. Therefore, P can be slightly larger than the PD components' estimated rank to guarantee this approximation's accuracy. Accordingly, an empirical rule to roughly determine P can be formulated as follows:

$$P = M\frac{\Delta f_{\mathrm{PD}}}{f_{\mathrm{s}}} + r, \quad (7)$$

where Δf_{PD} is the estimated frequency bandwidth of the PD signal, f_{s} is the sampling frequency, the term of $M\Delta f_{\mathrm{PD}}/f_{\mathrm{s}}$ denotes the estimated rank of the PD components, and r is the redundancy factor in guaranteeing reliable noise reduction performance. It can be set from 3 to 5 [25].

5) Components selection and signal reconstruction (steps 5 and 6)

Since the right orthonormal eigenvectors $\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots$, and $\boldsymbol{v}_P$ (obtained via RSVD) also contain the time-dimension characteristics of the components, we can apply kurtosis criteria to the eigenvectors to select pulse-type components before ISTFT, thus avoiding executing the ISTFT multiple times. The components with larger kurtosis values are chosen to reconstruct the noiseless matrix $\boldsymbol{X}' \in \mathbb{C}^{M \times N}$, which can be formulated as:

$$\boldsymbol{X}' = a_1\sigma_1\boldsymbol{Q}\boldsymbol{u}_1\boldsymbol{v}_1^H + a_2\sigma_2\boldsymbol{Q}\boldsymbol{u}_2\boldsymbol{v}_2^H + \cdots + a_P\sigma_P\boldsymbol{Q}\boldsymbol{u}_P\boldsymbol{v}_P^H, \quad (8)$$

where $a_i = \mathrm{logical}\{kur(\boldsymbol{v}_i^{\mathrm{re}}) > 4\}$, where $\boldsymbol{v}_i^{\mathrm{re}}$ is the real parts of the right eigenvector $\boldsymbol{v}_i$. Finally, the noiseless matrix $\boldsymbol{X}'$ is transformed back to time-domain signal $\boldsymbol{x}' \in \mathbb{R}^{1 \times L}$ via inverse STFT.

An overall comparison between the computational complexity of the improved STFT-RSVD algorithm and the STFT-SVD algorithm in [21] at each algorithm step is listed in Tab. I. It can be observed that the added kurtosis criteria between STFT and RSVD leads to a negligible additional computational cost, while applying RSVD to decompose the spectrogram and exchanging the order between the component selection and ISTFT can significantly reduce the computational complexity. Moreover, it is essential to point out that although some other time-frequency transform tools (e.g., continuous wavelet transform, Wigner-Ville distribution, Hilbert-Huang transform, etc.) are optional to obtain the spectrogram, we still select the STFT since it is the most time-saving one [26].

TABLE I

Computational complexity comparison between the STFT-SVD and the proposed STFT-RSVD algorithms

The improved STFT-RSVD algorithm			The STFT-SVD algorithm in [21]	
Steps	Computational complexity		Steps	Computational complexity
STFT+ Kurtosis criteria	$\mathrm{O}(MN\log M + 3MN)$	$\approx$	STFT	$\mathrm{O}(MN\log M + MN)$
Soft masking	$\mathrm{O}(4MN)$	$=$	Soft masking	$\mathrm{O}(4MN)$
RSVD	$\mathrm{O}(4MN\log P + 4(M+N)P^2)$ $(P \ll M, N)$	$\ll$	SVD	$\mathrm{O}(4M^2N)$
Component selection	$\mathrm{O}(2NP + 4MNP')(P' \ll P, R)$	$\ll$	ISTFT	$\mathrm{O}(MNR\log M + 5MNR)$
ISTFT	$\mathrm{O}(MN\log M + MN)$		Component selection	$2NR$

M denotes the window length of the STFT (also the number of rows of the spectrogram); N denotes the number of columns of the spectrogram, determined by $N = \lfloor (K - M + H)/H \rfloor$, where K is the sampling number of the signal, and H is the window hops of the STFT; R denotes the number of the components obtained via principal component analysis in the STFT-SVD algorithm in [21]; P denotes the number of columns of the given matrix $\boldsymbol{Q}$ in RSVD; P' denotes the number of the selected components in the proposed STFT-RSVD algorithm.

III. Implementation of the Proposed STFT-RSVD Algorithm in Embedded Systems

In on-site PD monitoring, PD signal detection, including signal acquisition, de-noising, and feature extraction (or data compression), is always completed in embedded systems due to their nominal cost, volume, power consumption, etc. This can improve the edge computing capability of the whole monitoring system and reduce the amount of data that needs to be stored. Embedded systems with varying hardware resources have been used for PD monitoring in different application scenarios, which depend on the testing object and the budget cost.

Three representative types of embedded systems are often used for PD diagnosis: System on Chip (SoC), Microprocessor Unit (MPU), and Microprogrammed Control Unit (MCU). SoC is a system-level chip with very abundant hardware resources (i.e., field-programmable gate array, multiple cores, high-speed processors, and extensive random-access memories (RAM)), but on the contrary, it has a very high cost and power consumption; it is often used for advanced PD diagnosis of recognition, classification, and localization, e.g., the PD location system for medium-voltage switchgears and cables [27, 28, 29]. MPU is a kind of integrated Central Processing Unit with multiple cores, high-speed processors, and more functionalities but may have small random-access memories; it is often used for PD detection, recognition, and classification, e.g., online PD monitoring for high-voltage motors, gas insulation systems, and transformers [30, 31]. MCU is a chip-level chip with fewer hardware resources (i.e., a single core, a low-speed processor, less RAM), but it has a low cost and power consumption; it is often used for low-cost PD monitoring, e.g., online IoT-based PD monitors for widely distributed overhead lines, cables, switchgear, etc. [2, 3, 32,33].

On the embedded systems side, three representative chips corresponding to the above SoC, MPU, and MCU have been selected to test the performance of the proposed algorithm and its alternatives. Their characteristic parameters and reference prices are listed in Tab. II. The used evaluation boards integrate all the hardware needed for programming and debugging the chips, namely the JTAG interface for ZYNQ7035 and the ST-LINK/V2 interface for STM32MP157 and STM32L476. The input or output PD signals are stored in the program memory (flash memory), while the intermediate signals or matrices are stored in the RAM. All three devices have been clocked at their maximum speeds.

On the computer side, communication with the above interfaces has been established using KELL 5.0 (for STM32MP157 and STM32L476) and VIVADO (for ZYNQ7035), free software for debugging and programming ARM and other systems. Once the program runs for a series of tests, the numerical outputs can be examined or exported through the debuggers, interrupting the program at convenient points. The timing of the single routines is computed by the SysTick timer built into the ARM cores. Since the recommended software (i.e., KELL 5.0 and VIVADO) only supports debugging and programming in the embedded systems in C Language, we must implement the PD de-noising algorithms in C Language. However, the C-languages of the most cost-efficient versions of some sub-functions in the algorithms (e.g., STFT and SVD) are unavailable. Alternatively, we programmed the algorithm in MATLAB codes and used a transcoding tool named MATLAB CODER to translate the MATLAB codes into C codes. On the one hand, the MATLAB software platform can provide all the latest sub-functions required in the algorithm. On the other hand, the MATLAB CODER allows us to allocate the dynamic memory to execute the algorithm according to the size of the RAM in the embedded systems, and it can automatically generate corresponding C codes [34]. This work sets the dynamic memories of ZYNQ7035, STM32MP157, and STM32L476 as 900 Mbytes, 700 Kbytes, and 100 Kbytes, respectively. Moreover, the MATLAB CODER includes the OpenMV interface, which can generate the C codes that can be executed in parallel with multiple cores, e.g., in ZYNQ7035 or STM32MP157. This can further reduce the computational cost of the embedded algorithms.

IV. Laboratory Experiment

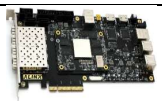
A generic PD signal with a low signal-to-noise ratio is synthesized to assess the de-noising algorithms. The feasibility of the proposed algorithm is validated via de-noising the synthetic PD signal in Device 1. The de-noising performance of the proposed algorithm and its alternative algorithms is compared in Devices 1, 2, and 3.

A. Synthetic noisy PD signal

Figure 3 collects the time-domain waveform of the synthetic PD signal. Since the synthetic signal includes multiple PD pulses with various waveforms, which simulate a general case in on-site PD measurement, it can be used as a qualified test sample to evaluate the de-noising performance of the proposed algorithm. The sampling rate of the synthetic signal is 500 MS/s, and the sampling number is 16000.

TABLE II

Features of three representative embedded systems commonly used for PD monitoring

Serial number	Device models	Pictures of the evaluation board	Types	ARM cores	Clock frequency	Memory (RAM)	Price of the device	Application scenarios
Device 1 (AX7350, ALINX)	ZYNQ7035		SoC	2×Cortex-A9+ Kintex-7	800 MHz	1 Gbytes	1280 USD	Advanced PD diagnosis: recognition, classification, and localization [27, 28, 29]
Device 2 (ATK-DLM P157M, ALIENTEK)	STM32MP157		MPU	Cortex-M4+ Cortex-A7	209 MHz +800 MHz	708 Kbytes	109 USD	PD detection, recognition, and classification [30, 31]
Device 3 (STM32 Nucleo-L47 6RG, ST)	STM32L476		MCU	Cortex-M4	80 MHZ	125 Kbytes	12 USD	Low-cost and low-power PD detection and recognition [2, 3, 32, 33]

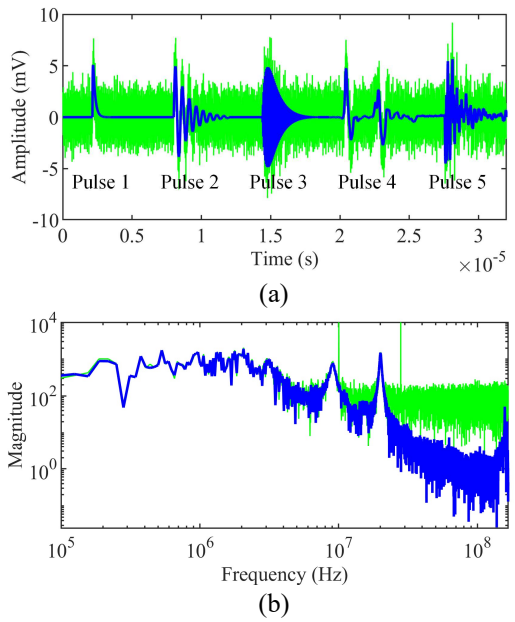


Fig. 3. Time-domain (a) and frequency-domain (b) waveforms of the synthetic noiseless and noisy PD signals. The green and blue lines denote the noisy and original signals, respectively.

Table III collects the models and parameters of the PD pulses in Fig. 3. At the PD source, the PD current pulse can be modeled by a double exponential pulse [4], e.g., pulse 1 in Fig. 3. However, most detected PD signals are oscillating pulses due to the effects of both the propagation path and the transfer function of the used sensor. Therefore, single exponential and double exponential attenuation oscillation pulses, e.g., pulses 2 and 3 in Fig. 3, are used. Moreover, it is important to point out that these three pulses are recognized as typical models of PD signals and have been widely used in the literature, such as in [6, 8, 9, 10, 11, 14, 17, 18, 19, 21]. Nevertheless, the real PD signals can be more complex than these three models, e.g., pulses 4 and 5 in Fig. 3. Pulse 4 originates from a fouling insulator and is detected by the high-frequency current transformer, as provided in [35]. Pulse 5 arises from the PD defect caused by a tree leaning on a 10-kV overhead covered conductor line and is detected by the same sensor. It is essential to point out that the sensor was deployed as close (about 20 cm) as possible to the defects to avoid the signal attenuation caused by the overhead line, to reduce the detrimental effects of noise, and to produce a reference response which is eventually compared with the reconstructed PD. In addition, the amplitudes of pulses 4 and 5 are reduced by several tens of times, giving them a similar amplitude to the first three pulses.

According to [4,5], discrete spectral interference and white noise always influence on-site PD measurement. Therefore, they are added to the synthetic PD signal. The frequencies of the two harmonics in the discrete spectral interference are set to 10 and 28 MHz, and their corresponding amplitudes are set to 1.5 and 1 mV, respectively, while the standard deviation of the white noise is set to 1 mV. It is essential to note that the synthetic noise includes very high-level noise with a similar amplitude to the PD pulses. Thus, it can be used to validate and assess the de-noising performance of the proposed algorithm for a PD signal with an extremely low signal-to-noise ratio.

TABLE III
Models and parameters of the PD pulses in Fig. 3

Series number	Models	Parameter values
Pulse 1	$A_1(e^{-\frac{t}{\tau_1}}-e^{-\frac{t}{\tau_2}})$	A_1=5 mV, τ_1=100 ns, and τ_2=10 ns.
Pulse 2	$A_2e^{-\frac{t}{\tau_3}}\sin(2\pi f_{c1}t)$	A_2=4 mV, τ_3=800 ns, and f_{c1}=2 MHz
Pulse 3	$A_3(e^{-\frac{t}{\tau_4}}-e^{-\frac{t}{\tau_5}})\sin(2\pi f_{c2}t)$	A_3=9 mV, τ_4=300 ns, τ_5=700 ns, and f_{c2}=20 MHz
Pulse 4	*(measured)*	
Pulse 5	*(measured)*	

The typical evaluation metric, i.e., signal-to-noise ratio (*SNR*), is defined below to quantify the quality of the estimated noiseless PD signal:

$$SNR = 10\log_{10}\frac{\sum_{l=1}^{L}[x'(t_l)]^2}{\sum_{l=1}^{L}[x'(t_l)-s(t_l)]^2} \quad (9)$$

where $x'(t_l)$ and $s(t_l)$ denote the discrete noise-free and de-noised signals, respectively, and t_k is equal to $(l-1)\Delta t$, where Δt is the sampling interval. Higher *SNR* represents better de-noising performance. *SNR* of the synthetic noisy signal is calculated as -5.849 dB.

B. Validation experiment (in Device 1)

In Section II, three improvements in the proposed STFT-RSVD algorithm are developed, aiming at reducing its computational complexity without influencing its de-noising performance. To validate the feasibility of the improvements, the de-noising experiment of the proposed STFT-RSVD algorithm was carried out in the synthetic PD signal in Device 1, which was compared with the STFT-SVD method in [21]. The parameters of the STFT-RSVD algorithm were set as M=320, H=10, and P =20, according to the empirical equations in Section II and [21].

Figure 4 collects the plots of the kurtosis values of the real parts of rows of the spectrograms of the noise and the noisy PD signal. The index axis denotes the serial number of rows of the spectrograms. It can be observed that the kurtosis values of the noise spectrogram are less than the threshold, while parts of the noisy PD signal are significantly larger than the threshold. According to Equation (1), the class of the spectrogram of the noise will be identified as "No," while that of the noisy PD signal will be "Yes," validating the feasibility of the kurtosis criteria.

Figure 5 collects the plots of the kurtosis values of the reconstructed sub-signals obtained via applying ISTFT to each component in [21] and the real parts of the right orthonormal eigenvectors obtained via the RSVD. The index axel denotes the serial number of the components obtained via the RSVD. It can be observed that the kurtosis values of the real parts of the right orthonormal eigenvectors are generally similar to that of the reconstructed sub-signals, verifying the feasibility of selecting the pulse-type components by applying kurtosis criteria to the right orthonormal eigenvectors. Although the difference between the kurtosis values of the two curves tends to be more prominent as the index increases, the result of the component selection via the threshold criteria is hardly influenced. Moreover, since the components' energy significantly reduces as the index rises,

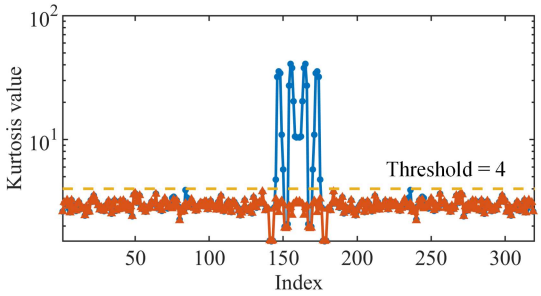


Fig. 4. Kurtosis values of real parts of rows of the spectrograms of the noise (triangle labels) and noisy PD signal (round labels) (M=320, H=10, and P =20).

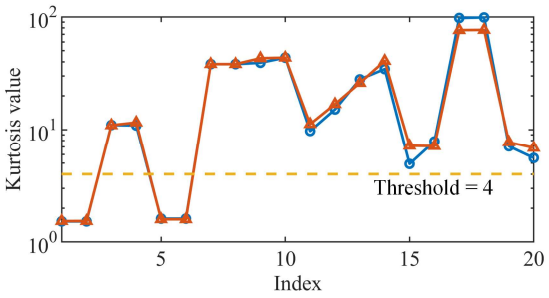


Fig. 5. Kurtosis values of the reconstructed sub-signals obtained via applying ISTFT to each component [21] (round labels) and the real parts of the right eigenvectors obtained via the RSVD (triangle labels) (M=320, H=10, and P =20).

discriminant error on the element with a larger index hardly affects the final de-noising result.

Figures 6 and 7 collect the de-noised signals and computation time of the STFT-SVD method in [21] and the proposed STFT-RSVD algorithm, respectively. In Fig. 6, it can be observed that their de-noising results are almost coincident as their differences are negligible. *SNR*s of the de-noising signals obtained via the STFT-SVD and STFT-RSVD algorithms are calculated as 18.34 dB and 18.11 dB, respectively, which are significantly higher than that of the original synthetic noisy signal, i.e., -5.849 dB. In contrast, the computation time of the proposed STFT-RSVD algorithm is significantly less than that of the STFT-SVD algorithm, especially in terms of the SVD or RSVD, component selection, and ISTFT steps, as shown in Fig. 7. The results demonstrate the essential benefits of the RSVD and exchanging the order between component selection and ISTFT in reducing computational complexity.

C. Parameter effect simulation (in Device 1)

The computational complexity of the proposed STFT-RSVD algorithm is associated with its three parameters, i.e., window length M of STFT, window hops H of STFT, and the estimated rank P of the spectrogram of the noiseless PD signal. Therefore, this section will investigate their effect on the computing cost of the proposed algorithm via de-noising the synthetic PD signal in Device 1. Moreover, since down-sampling the PD signal is commonly applied to reduce the computational burden of the used embedded systems, its effect on the proposed algorithm is also evaluated.

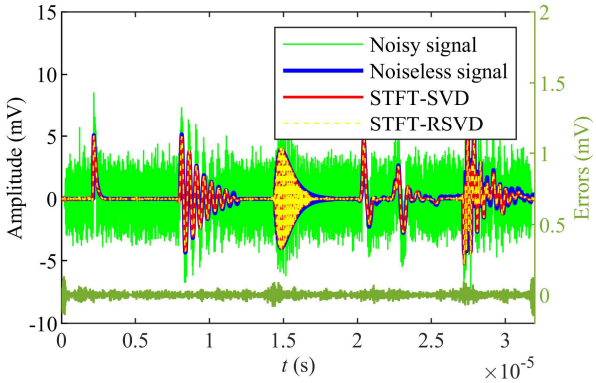


Fig. 6. De-noising results of the STFT-SVD and the improved STFT-RSVD algorithms (M=320, H=10, and P=20) and their differences (i.e., the subtraction between the two de-noised signals).

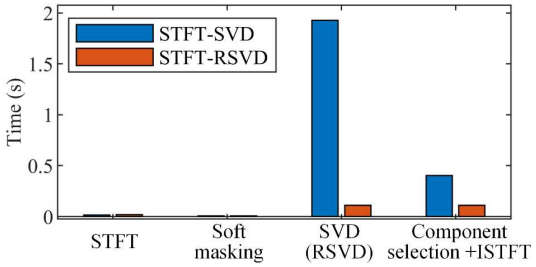


Fig. 7. Computation times of all steps in the STFT-SVD (a) and STFT-RSVD (b) algorithms (M=320, H=10, and P =20).

Figure 8 collects the curves of *SNR*s and the overall computational time of the proposed algorithm against the window length M. H and P are set as 10 and 20, respectively. It can be observed that the de-noising performance improves as M increases in the initial part of the curves. It becomes nearly flat once the number of sampling points exceeds the critical value of 100, and finally, it declines slowly as M continues to increase. The computation time of the proposed algorithm gradually increases as M increases since the size of the spectrogram (i.e., the number of rows) increases, too. Moreover, the increased size can lead to a significantly increased memory requirement. Therefore, considering both the de-noise performance and efficiency, selecting the smallest value of M may lead to sufficiently good performance indexes.

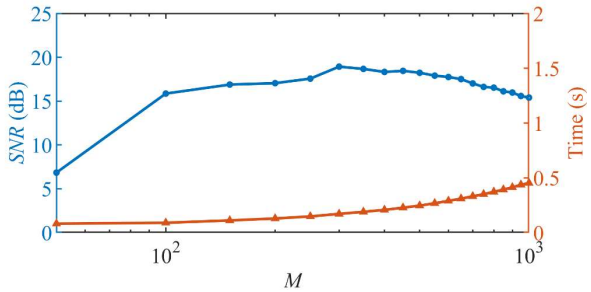


Fig. 8. Effect of window length M on SNR and the overall computational time required by the proposed STFT-RSVD algorithm (H=10, P = 20).

Figure 9 collects the curves of *SNR*s and the overall computational time of the proposed algorithm against the window hops H. M and P are set as 320 and 20, respectively. It can be observed that the de-noising performance stays nearly flat as H increases in the initial part of the curve. It decreases slowly, and finally, it oscillates and decreases faster once H exceeds the critical value of 20. The computation time of the proposed algorithm significantly decreases as H increases since the size of the spectrogram (i.e., the number of columns) decreases. Considering the de-noise performance and efficiency, selecting the critical value of H may lead to sufficiently good performance indexes.

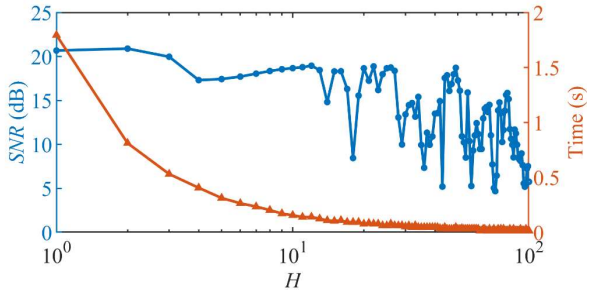


Fig. 9. Effect of window hops H on SNR and the overall computational time required by the proposed STFT-RSVD algorithm ($M = 320$, $P = 20$).

Figure 10 collects the curves of *SNR*s and the overall computational time of the proposed algorithm against the estimated rank P. H and M are set as 10 and 320, respectively. It can be observed that the de-noising performance improves as P increases in the initial part of the curves, and then it becomes nearly flat once P exceeds the critical value of 12. Therefore, as suggested in equation (3), P should be set to be somewhat more significant than the estimated rank of the PD components in the proposed STFT-RSVD algorithm, as such a value of P can lead to the same de-noising performance as the STFT-SVD algorithm but significantly reduced computational cost, which verifies the critical benefit of the RSVD.

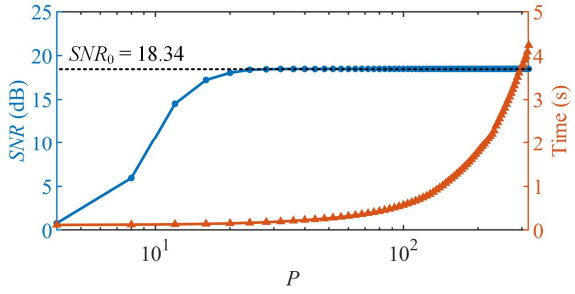


Fig. 10. Effect of P on SNR and the overall computational time required by the proposed STFT-RSVD algorithm ($M = 320$, $H = 10$). (SNR_0 is the SNR of the de-noised signal obtained via the STFT-SVD algorithm [21]).

Figure 11 collects the curves of *SNR*s and the overall computational time of the proposed algorithm against the down-sampling intervals. M, H, and P are set as 160, 10, and 20, respectively. It can be observed that as the down-sampling intervals increase, the de-noising performance decreases linearly while its computation time decreases exponentially. This means that down-sampling the input noisy PD signal can be a feasible trade-off solution if the computing resources of the used embedded system are severely insufficient.

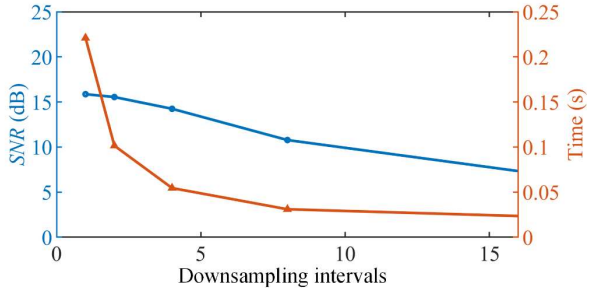


Fig. 11. Effect of down-sampling intervals on SNR and the overall computational time required by the proposed STFT-RSVD algorithm ($M = 160$, $H = 10$, and $P = 20$).

D. Comparison result (in Devices 1, 2, and 3)

De-noising the synthetic noisy PD signal via the proposed algorithm and its alternatives (i.e., WT [9], EMD [10], H-FSVD [14], AST-SVD [15], S-SVD [20], SVD-EWT[17], and SVD-VMD [19]) was carried out in Devices 1, 2, and 3. We selected three group parameters for the proposed STFT-RSVD algorithm in this experiment: (i) M=80, H=1, and P=20; (ii) M=40, H=5, P=12; (iii) M=20, H=10, P=8, which all satisfy the empirical equations in Section II and in [21]. The first group of parameters was used in the proposed algorithm executed in Device 1, while the second and last were in Device 2 and 3, respectively. Moreover, since the RAMs of Devices 2 and 3 are minimal, the synthetic signal is down-sampled at intervals of eight sampling points before de-noising, i.e., the length of the synthetic signal is compressed from 16,000 to 2,000 by sampling at equal intervals. This operation simulates real-world applications where high sampling rates have to be abandoned due to the limited hardware resources of the monitoring devices. *SNR* of the synthetic signal is re-calculated as -5.796 dB according to Equation (5).

The de-noising results are shown in Fig. 12. In Fig. 12(a), the WT algorithm behaves well for the first pulse but fails to reconstruct the third pulse. The difference in the noise reduction effect on the PD pulses is attributed to the selected mother wavelet (i.e., dB8). In Fig. 12(b), the EMD technique can only discriminate the PD signal vaguely; discrete spectral interferences and waveform distortions remain; it fails to reconstruct the third and last pulses. In Fig. 12(c), the H-FSVD fails to reduce the discrete spectral interference since its principal component analysis cannot distinguish PD and discrete spectral interference. In Fig. 12(d), the AST-SVD fails to reduce the discrete spectral interference, similar to the H-FSVD. In Fig. 12 (e), the S-SVD fails to de-noise the first pulses, and waveform distortions remain. In Fig. 12 (f), the SVD-EWT can de-noise all pulses, but visible distortions remain. In Fig. 12 (g), the SVD-VMD behaves well for the first, second, and fourth pulses, but obvious distortions of the third and last pulses remain. In Fig. 12(h), (i), and (j), compared with the results of the other algorithms, the proposed STFT-RSVD

algorithm can effectively reduce both the white noise and the discrete spectral interference in all five types of PD pulses, demonstrating its more robust adaptability to de-noise various PD signal with an extremely low signal-to-noise ratio.

All the algorithms' evaluation metrics (SNR and computing times) are listed in Tab. IV. For de-noising performance, it can be observed that the proposed STFT-RSVD algorithm with parameter settings of M=80, H=1, and P=20 has the highest *SNR*. For computing time, in Device 1, it can be observed that as M decreases and H increases (i.e., the size of the time-frequency matrix decreases), the computational time of the proposed STFT-RSVD algorithm decreases exponentially; the STFT-RSVD algorithm requires significantly less computing time than H-FSVD; although the STFT-RSVD algorithm requires slightly more computational time than the WT technique, it provides considerably better de-noising performance, as discussed in the last paragraph. Moreover, it is essential to note that the H-FSVD algorithm cannot be implemented in Devices 2 and 3 due to the limited memories, which cannot execute the expensive SVD operation in these algorithms. In contrast, the proposed STFT-RSVD algorithm can be implemented in Devices 2 and 3 as long as the proper parameters are selected. This demonstrates the most crucial benefit: the proposed algorithm can be implemented in different embedded systems with varying hardware resources.

In addition, the parameters of the synthetic signal in Tab. III were changed to test the algorithms in Device 1 further. *SNR*s of the de-noised signal obtained via the algorithms are listed in Tab. V. It can be observed that the *SNR*s of the proposed algorithm are still the highest, despite slight fluctuations as the parameters of the synthetic signal change, verifying the robustness of the proposed algorithm.

V. Validation of the Proposed Algorithm in IoT-Based PD Monitors for MV Switchgears

In this section, the feasibility of the proposed algorithm is validated via de-noising real PD data collected by the IoT-based PD monitors on the surface of a row of 35-kV switchgears, as shown in Fig. 13. The monitor consists of a transient earth voltage coupler [36, 37] as the PD sensor, a data acquisition and processing module, a communication module, and a battery. Since the number of 35-kV switchgears is enormous, which leads to the limited budget of the monitor for one switchgear, the data acquisition and processing module use the low-cost and low-power STM32L476 as their processor and the built-in analog-to-digital device with a sampling ratio of 5 Ms/s to collect data. The analog-to-digital device captures the PD signal component from tens of kHz to 2.5 MHz, sufficient to achieve the primary and fast PD diagnosis [37, 38] despite losing some signal waveform details. Compared with Device 3,

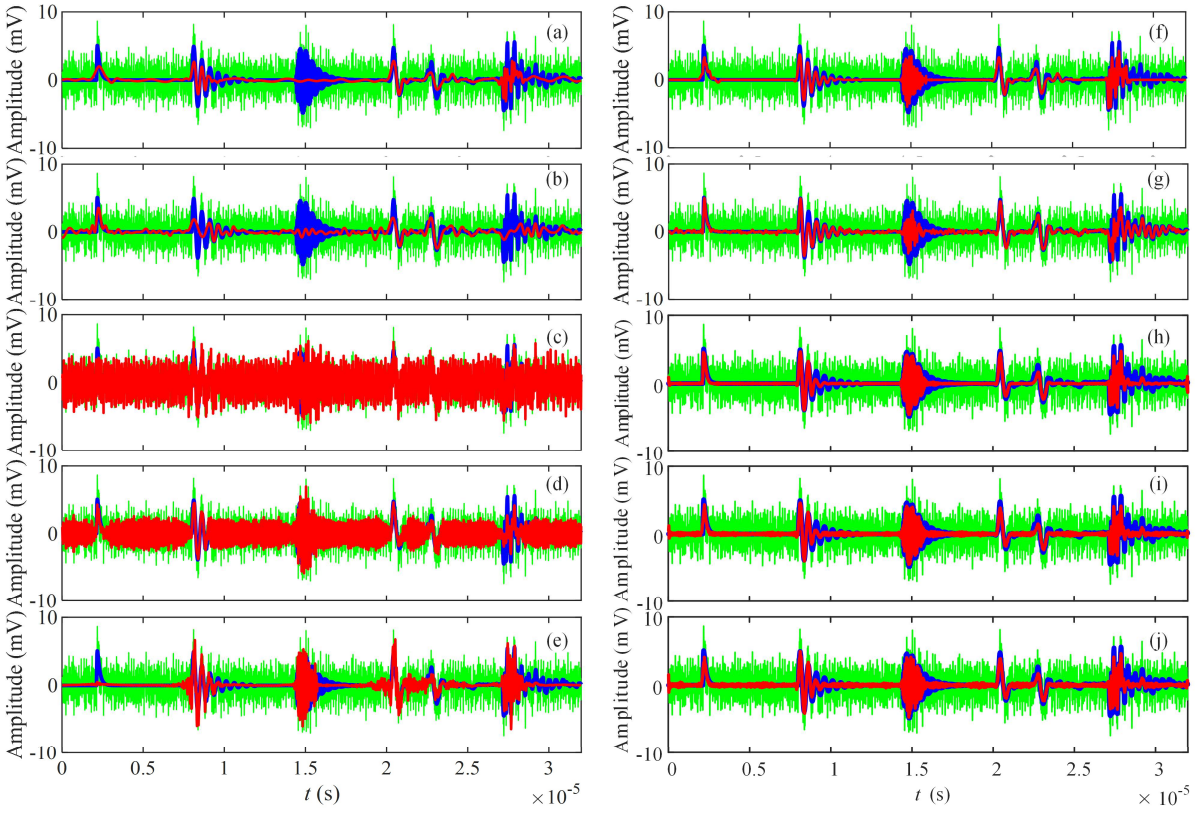


Fig. 12. De-noising results of (a) WT (the mother wavelet: dB8; decomposition level: 5) in Device 1, (b) EMD in Device 1, (c) H-FSVD (the size of Hankel matrix: 500×2000) in Device 1, (d) AST-SVD (the size of the sliding window: 200; the size of Hankel matrix: 100×200) in Device 1, (e) S-SVD (the adjustable factor of S-transform: 1) in Device 1, (f) SVD-EWT in Device 1, (g) SVD-VMD in Device 1, (h) the proposed STFT-RSVD ($M = 80$, $H = 1$, and $P = 20$) in Device 1, (i) the proposed STFT-SVD ($M = 40$, $H = 5$, and $P = 12$) in Device 2, (j) the proposed STFT-SVD ($M = 20$, $H = 10$, and $P = 6$) in Device 3. The green, blue, and red lines denote the noisy, original, and de-noised signals.

TABLE IV
Evaluation metrics comparison of the algorithms

Method	*SNR* (dB)	Computation time (s)		
		Device 1	Device 2	Device 3
STFT-RSVD (*M*=80, *H*=1, *P*=20)	11.819	0.133	/	/
STFT-RSVD (*M*=40, *H*=5, *P*=12)	9.128	0.021	0.049	/
STFT-RSVD (*M*=20, *H*=10, *P*=6)	7.633	0.010	0.016	0.847
WT[9]	2.348	0.005	0.011	0.483
EMD[10]	2.857	0.007	0.014	0.726
H-FSVD[14]	-0.547	0.112	/	/
AST-SVD[15]	4.157	0.437	/	/
S-SVD[20]	0.669	0.842	/	/
SVD-EWT[17]	7.106	0.375	/	/
SVD-VMD[19]	6.357	1.024	/	/

*Remark: "/" means that the memory is insufficient to run the algorithm. Moreover, since the minimal memory of Device 1, the signal is divided into five segments of equal length (i.e., 400 sampling points), which are de-noised one by one in Device 1; on the contrary, the whole signal is directly de-noised in Device 2 and 3 without segmentation.

TABLE V
De-noising performance comparison of the algorithms under difference parameter settings of the synthetic signal in Tab. III

Method	*SNR* (dB)		
	τ_1=200 ns τ_2=20 ns. τ_3=800 ns τ_4=150 ns τ_5=300 ns f_{c1}=8 MHz f_{c2}=25 MHz (-6.8218 dB)	τ_1=400 ns τ_2=40 ns τ_3=1000 ns τ_4=100 ns τ_5=200ns f_{c1}=4 MHz f_{c2}=20 MHz (-6.0280 dB)	τ_1=800 ns τ_2=80 ns τ_3=1200 ns τ_4=200ns τ_5=400 ns f_{c1}=2 MHz f_{c2}=15 MHz (-4.6559 dB)
STFT-RSVD (M=80, H=1, P=20)	10.923	11.618	12.168
STFT-RSVD (M=40, H=5, P=12)	8.175	8.515	9.211
STFT-RSVD (M=20, H=10, P=6)	6.697	7.214	7.417
WT[9]	2.654	3.248	5.348
EMD[10]	2.059	3.924	4.857
H-FSVD[14]	-0.575	-0.125	-0.547
AST-SVD[15]	1.157	2.114	2.235
S-SVD[20]	0.813	0.257	1.547
SVD-EWT[17]	6.447	6.663	7.024
SVD-VMD[19]	6.456	7.157	7.154

an external memory card of 2 Mbytes is added in the module to store the collected data with at least a power-frequency cycle (i.e., 20 ms or 100000 sampling points). The PD monitors are enabled to complete PD detection in a time interval of half an hour for two main reasons. On the one hand, the limited computation hardware resources and communication bandwidth of the monitors lead to difficulty in real-time PD monitoring. On the other hand, most PD activities are often dynamically stable within a short interval, e.g., tens of minutes [31,39]. The devices' minor deficiencies that may miss some

Fig. 13. Online PD monitoring of MV switchgears via IoT-based PD monitors.

special PD events (e.g., PDs caused by the overvoltage transients) can be tolerated in practical industrial application due to the devices' outstanding advantages of small volume, low cost, and independent of external power supply.

The proposed STFT-SVD algorithm and its alternative (i.e., the WT and EMD algorithms) are implemented in another PD monitor with the same hardware as that in Fig. 13. Since the memory of the processor is minimal, causing it impossible to directly de-noise the whole data at one time, we divided the data into 250 segmentations with the same length of 200 via a sliding window. These segmentations are long enough to cover a PD pulse and are de-noised one by one. Although this division may be done in the middle of a few pulses, the final PD diagnosis is hardly influenced by this problem since it is always based on the statistical results of all de-noised PD pulses, e.g., the phase-resolved PD spectrum [40]. Since the duration of a PD pulse in 35-kV switchgear is always more than 1 μs [37], which means that the signal always contains a frequency component less than 1 MHz, the window length M should be set as at least 15 (i.e., more than $\alpha \times 5$ MHz/1 MHz according to the empirical equation in [21], where α is set as 3) to guarantee a sufficient frequency resolution of the spectrogram. On the other hand, M cannot be set too large due to the limit of its random access memory (i.e., 125 Kbytes) in STM32L476. Therefore, we selected three group parameters for the proposed STFT-RSVD algorithm in this experiment: (i) M=25, H=2, and P=10; (ii) M=20, H=4, P=8; (iii) M=15, H=6, P=5, where the possible values are set according to the empirical equation (4) provided in Section II.

The de-noising results of the proposed algorithm and its alternatives are collected in Fig. 14. It is important to point out that all SVD-based algorithms were not implemented in the PD monitor since the SVD requires a huge RAM that the used monitor cannot provide. It can be observed that the WT algorithm can only discriminate the PD signal vaguely, which leads to significant distortion and energy loss. The EMD algorithm almost fails to de-noising most PD pulses. In contrast, the proposed algorithm yields good reconstructed PD pulses, causing most PD pulses (some even completely drowned in the noise) to be detected and their energy loss significantly less than the WT algorithm. It makes applying to various industrial applications accessible without intensively tuning the parameters. The total computing times of using the proposed algorithm and its alternatives to the whole data in Fig. 14 are listed as Tab. VI. It can be observed that the proposed

algorithms with the second and third parameter settings require significantly less computation time than that of the WT and EMD algorithms, primarily due to the unnecessary calculations of the data segment without any PD pulses (i.e., no PD pulse in the sliding time window) being avoided in the proposed algorithm. Moreover, it is essential to point out that the total computation time of the proposed algorithm (i.e., tens of seconds) is significantly less than the test interval (i.e., half an hour) of the PD monitors, allowing it to be used in these monitors.

Out of caution, the relevant maintenance department carried out the opening inspection of the two switchgears on the far right several days later. They found the bush between the two, as shown in Fig. 13. Immediately, they cleared up the fouling on the surface of the bush, and the PD pulses disappeared, as shown in Fig. 15, which verified that the fouling is the cause of the PD and the proposed algorithm can effectively extract the PD pulses drown in the noises.

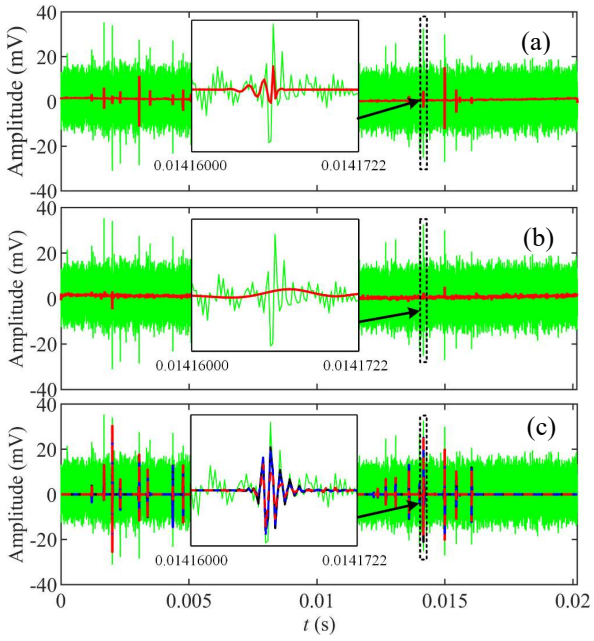


Fig. 14. De-noising results of (a) WT (the mother wavelet: dB8; decomposition level: 5) in Device 3, (b) EMD in Device 3, and (c) the proposed STFT-RSVD in Device 3 (black line: $M = 25$, $H = 2$, and $P = 10$; blue line: $M = 20$, $H = 4$, and $P = 8$; red line: $M = 15$, $H = 6$, and $P = 5$;). The green and red lines denote the noisy and de-noised signal.

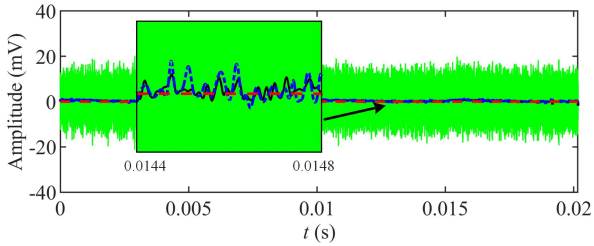


Fig. 15. De-noising results of WT (black line), EMD (blue line), and the proposed STFT-RSVD (red line) in Device 3 after removing the PD defect ($M = 25$, $H = 2$, and $P = 10$). The green denotes the noisy and de-noised signal.

TABLE VI

The computing times of applying the algorithms to de-noising the 20-ms data in Fig. 11

Method	Total computation time (s)
WT[9]	48.052
EMD[10]	45.381
STFT-RSVD (M=25, H=2, P=10)	50.089
STFT-RSVD (M=20, H=4, P=8)	36.592
STFT-RSVD (M=15, H=6, P=5)	22.693

Before removing the PD defect, 33 original PD data (each data contains the 20-ms noisy PD signal like that in Fig. 14) were stored, and they will be used further to validate the de-noising performance of the proposed algorithm (with parameters of M =15, H =6, P =5). Figure 16 contains the statistical phase-resolved spectrum of the original data and the de-noised results, the acknowledged data form widely used to diagnose PD magnitude and types [40]. It can be observed that it is indeed difficult to identify any PD activities in the spectrum of the original signal due to the severe noise; visible PD activities can be observed in the spectrum obtained via the WT algorithm, but it causes a significant energy loss and misses to detect most of PD pulses; the spectrum of the EMD algorithm reveals a small number of PD events; in contrast, the proposed algorithm yields a significantly more precise and more accurate phase-resolved spectrum, which can be undoubtedly identified as one with significant PD activities. It reveals that the proposed algorithm can substantially improve the sensitivity and accuracy of PD detection via the monitors and has a better de-noising performance than the state-of-the-art alternatives.

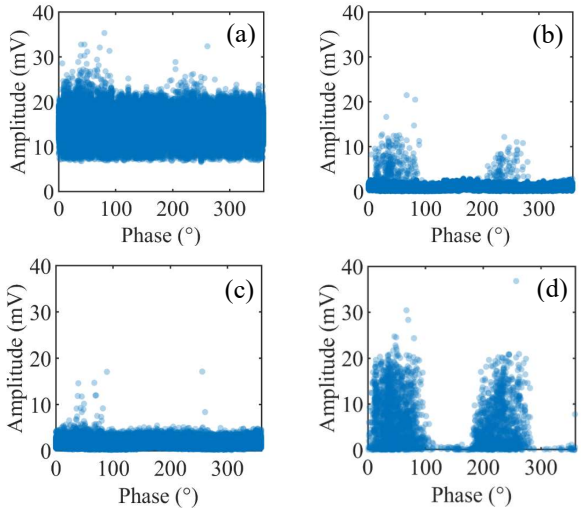


Fig. 16. Statistical phase-resolved spectrum of PD events detected in (a) the measured noisy signals; (b) the de-noised signals obtained via WT; (c) the de-noised signals obtained via EMD; (d) the de-noised signals obtained via the proposed STFT-RSVD.

VI. Conclusion

This paper proposes an adaptive and efficient PD de-noising algorithm based on the joint application of STFT and RSVD.

The proposed algorithm can be integrated into computing resource-limited embedded systems (e.g., Cortex-M4) used for PD monitoring to reduce both the white noise and the discrete spectral interferences in PD pulses with uncertain waveform parameters. The feasibility and effectiveness of the proposed algorithm are validated via comprehensive simulation and measurement experiments. To sum up,

1) The comparison with the previous STFT-SVD algorithm in the simulation experiment shows that the computational efficiency of the proposed algorithm is significantly improved without sacrificing its de-noising performance, validating the effectiveness of its three improvements, i.e., adding kurtosis criteria before matrix factorization, decomposing the spectrogram using the RSVD algorithm, and selecting dominate components by applying kurtosis criteria to the right eigenvectors.
2) The parameter effect simulation experiments show that the de-noising performance of the proposed algorithm is quite robust with respect to an empirical tuning of its internal parameters (i.e., window length and hops of the STFT and estimated rank of the spectrogram of the noiseless PD signal). On the other hand, such parameters strongly affect the computation time. Therefore, setting these parameters around their critical values can yield the highest cost performance.
3) The comprehensive comparison with other state-of-the-art alternatives in simulation experiments of de-noising PD pulses with various waveforms in three typical embedded systems shows that the proposed algorithm yields minimal waveform distortions of all considered PD signals and minimal computing times, proving its superior de-noising performance and computational efficiency.
4) The application of the algorithm in resource-limited devices (i.e., using low-cost and -power STM32L476 as their processers) for PD monitoring of 35-kV switchgears demonstrates its effectiveness and superiority, as it causes significantly less waveform distortion and energy loss than other state-of-the-art alternatives, i.e., WT and EMD. In this case, the proposed algorithm yields a more accurate statistical phase-resolved PD spectrum, which is fundamental to achieving reliable PD recognition and classification in the subsequent diagnosis process.

Future works will investigate the proposed algorithm's possible application or improvement for more PD sensors and more severe noise environments, e.g., colored or impulse noises.

REFERENCES

[1] G. C. Montanari and A. Cavallini, “Partial discharge diagnostics: from apparatus monitoring to smart grid assessment,” in *IEEE Electrical Insulation Magazine*, vol. 29, no. 3, pp. 8-17, May-June 2013.

[2] M. Riera-Guasp, J. A. Antonino-Daviu and G. -A. Capolino, “Advances in Electrical Machine, Power Electronic, and Drive Condition Monitoring and Fault Detection: State of the Art,” in *IEEE Transactions on Industrial Electronics*, vol. 62, no. 3, pp. 1746-1759, March 2015.

[3] Sikorski, Wojciech, and Artur Wielewski, “Low-Cost Online Partial Discharge Monitoring System for Power Transformers,’’ Sensors 23.7 23, no. 7: 3405, Mar. 2023.

[4] T. Zhuang, M. Ren, X. Gao, M. Dong, W. Huang and C. Zhang, “Insulation Condition Monitoring in Distribution Power Grid via IoT-Based Sensing Network,” in *IEEE Transactions on Power Delivery*, vol. 34, no. 4, pp. 1706-1714, Aug. 2019.

[5] Xie, Jun, et al.,”Suppressing the discrete spectral interference of the partial discharge signal based on bivariate empirical mode decomposition,” in *International Transactions on Electrical Energy Systems* 27.10 (2017): e2407.

[6] S. Sriram, S. Nitin, K. M. M. Prabhu and M. J. Bastiaans, “Signal denoising techniques for partial discharge measurements,” in *IEEE Transactions on Dielectrics and Electrical Insulation*, vol. 12, no. 6, pp. 1182-1191, Dec. 2005.

[7] F. Alvarez, J. Ortego, F. Garnacho and M. A. Sanchez-Uran, “A clustering technique for partial discharge and noise sources identification in power cables by means of waveform parameters,” in *IEEE Transactions on Dielectrics and Electrical Insulation*, vol. 23, no. 1, pp. 469-481, February 2016, doi: 10.1109/TDEI.2015.005037.

[8] A. R. Mor, L. C. C. Heredia and F. A. Muñoz, “Estimation of charge, energy and polarity of noisy partial discharge pulses,” in *IEEE Transactions on Dielectrics and Electrical Insulation*, vol. 24, no. 4, pp. 2511-2521, 2017, doi: 10.1109/TDEI.2017.006381.

[9] J. Li, C. Cheng, T. Jiang and S. Grzybowski, “Wavelet de-noising of partial discharge signals based on genetic adaptive threshold estimation,” in *IEEE Transactions on Dielectrics and Electrical Insulation*, vol. 19, no. 2, pp. 543-549, April 2012.

[10] S. Zhang *et al*., “An Adaptive CEEMDAN Thresholding Denoising Method Optimized by Nonlocal Means Algorithm,” in *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 9, pp. 6891-6903, Sept. 2020.

[11] K. Dragomiretskiy and D. Zosso, “Variational Mode Decomposition,” in *IEEE Transactions on Signal Processing*, vol. 62, no. 3, pp. 531-544, Feb.1, 2014.

[12] Zhang, Jun, Junjia He, Jiachuan Long, Min Yao, and Wei Zhou. 2019. “A New Denoising Method for UHF PD Signals Using Adaptive VMD and SSA-Based Shrinkage Method,” *Sensors* 19, no. 7: 1594..

[13] M. Krátký, S. Mišák, P. Gajdoš, P. Lukáš, R. Bača and P. Chovanec, “A Novel Method for Detection of Covered Conductor Faults in Medium Voltage Overhead Line Systems,” in *IEEE Transactions on Industrial Electronics*, vol. 65, no. 1, pp. 543-552, Jan. 2018.

[14] S. Govindarajan, J. Subbaiah, A. Cavallini, K. Krithivasan and J. Jayakumar, “Partial Discharge Random Noise Removal Using Hankel Matrix-Based Fast Singular Value Decomposition,” in *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 7, pp. 4093-4102, July 2020.

[15] H. Karami and G. B. Gharehpetian, “Limitations of Partial Discharge De-noising of Power Transformer Using Adaptive Singular Value Decomposition,” *2019 International Power System Conference (PSC)*, Tehran, Iran, 2019, pp. 777-781.

[16] K. Zhou, M. Li, Y. Li, M. Xie, Y. Huang, “An improved denoising method for partial discharge signals contaminated by white noise based on adaptive short-time singular value decomposition,” in *Energies*. Vol. 12, no. 18, pp. 3465-3481, Jan. 2019, doi: 10.3390/en12183465.

[17] L. Li, X. Wei, “Suppression Method of Partial Discharge Interferences Based on Singular Value Decomposition and Improved Empirical Mode Decomposition,” in *Ener*gies. vol. 14, no. 24, pp. 8579-8601, Jan. 2021.

[18] J. Zhong, X. Bi, Q. Shu, M. Chen, D. Zhou and D. Zhang, “Partial Discharge Signal Denoising Based on Singular Value Decomposition and Empirical Wavelet Transform,” in *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 11, pp. 8866-8873, Nov. 2020.

[19] Z. Lei, F. Wang and C. Li, “A denoising method of partial discharge signal based on improved SVD-VMD,” in *IEEE Transactions on Dielectrics and Electrical Insulation*, doi: 10.1109/TDEI.2023.3269725.

[20] Y. Liu, W. Zhou, P. Li, S. Yang, Y. Tian, “An ultrahigh frequency partial discharge signal de-noising method based on a generalized S-transform and module time-frequency matrix,” in Sensors. Vol. 16, no. 6, pp. 941-960, Jun. 2016, doi: 10.3390/s16060941.

[21] Y. Yan, R. Trinchero, I. S. Stievano, H. Li and Y. -Z. Xie, “An Automatic Tool for Partial Discharge De-Noising via Short-Time Fourier Transform and Matrix Factorization,” in *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1-12, 2022, Art no. 3528512.

[22] DeCarlo, Lawrence T., "On the meaning and use of kurtosis," Psychological methods 2, no. 3, pp. 292-307, 1997.

[23] Sawada, Hiroshi, Shoko Araki, Ryo Mukai, and Shoji Makino, "Blind extraction of dominant target sources using ICA and time-frequency masking," IEEE Transactions on Audio, Speech, and Language Processing, 14, no. 6, pp. 2165-2173, NOV, 2006.

[24] Li, H., Kluger, Y. & Tygert, M., "Randomized algorithms for distributed computation of principal component analysis and singular value decomposition," *Adv Comput Math* 44, 1651–1672, 2018.

[25] N. Halko, P. G. Martinsson and J. A. Tropp, "Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions," *SIAM Rev.*, vol. 53, no. 2, pp. 217-288, 2011.

[26] E. H. Bouchikhi, V. Choqueuse, M. E. H. Benbouzid, J. F. Charpentier and G. Barakat, "A comparative study of time-frequency representations for fault detection in wind turbine," *IECON 2011 - 37th Annual Conference of the IEEE Industrial Electronics Society*, Melbourne, VIC, Australia, 2011, pp. 3584-3589.

[27] Smart Diagnostic Monitoring Technologies (SDMT), *PD700-A complete Partial Discharge (PD) monitoring system for Gas Insulated Switchgears (GIS)*. [Online]. Available (cited 2023 October 17): https://sdmt.co.uk/products/pd700/.

[28] Y. Yan, S. Ren, Y. Lu, S. Yang, K. Zhao and H. Li, "Efficient and Low-Cost PD Monitoring and Locating System for MV Switchgears Using TEV Detectors," in *IEEE Transactions on Power Delivery*, vol. 36, no. 5, pp. 3266-3269, Oct. 2021.

[29] Yan, Y., Zhao, Y., Zhao, K., Trinchero, R., Stievano, I. S., & Li, H., "A high‐efficiency portable system for insulation condition assessment of wind farm inter-array cables with double-sided partial discharge detection and localization," *IET Generation, Transmission & Distribution*, 2023.

[30] Lu, Yuefeng, et al. "Design and Application of Integrated Sensing Terminal for Power Transformer." *2021 3rd International Conference on Electrical Engineering and Control Technologies (CEECT)*. IEEE, 2021.

[31] L. A. Renforth, R. Giussani, M. T. Mendiola and L. Dodd, "Online Partial Discharge Insulation Condition Monitoring of Complete High-Voltage Networks," in *IEEE Transactions on Industry Applications*, vol. 55, no. 1, pp. 1021-1029, Jan.-Feb. 2019, doi: 10.1109/TIA.2018.2866983.

[32] NOOTEEG, *IoT Partial Discharge monitoring system based on transient earth voltage coupler and ultrasonic sensor*. [Online]. Available (cited 2023 October 17): http://www.ch-nt.com/products.html?product_id=52.

[33] Y. Yan, J. Tao, R. Trinchero, I. S. Stievano and H. -J. Li, "A Compact Detector for Flexible Partial Discharge Monitoring of 10-kV Overhead Covered Conductor Lines," in *IEEE Transactions on Power Delivery*, vol. 37, no. 6, pp. 5492-5495, Dec. 2022.

[34] Can-can, L. I. A. O., Shu-qun, Z. H. A. N. G., & Zhao-yi, L. E. I., "Research and Application of C Code Generated by Matlab Coder". Computer and Modernization, 1(3), 175, 2013.

[35] Y. Yan, Y. Zhao, W. He, I. S. Stievano and H. Li, "On-Line Partial Discharge Localization of 10-kV Covered Conductor Lines," in *IEEE Transactions on Power Delivery*, vol. 38, no. 3, pp. 1688-1698, 2023.

[36] K. -H. Ji, J. -X. Xiao, W. -B. Li, X. -F. Wang, P. Song and W. -R. Si, "Review of partial discharge detection technology for transient earth voltage of HV switchgear cabinet," *2020 13th International Conference on Intelligent Computation Technology and Automation (ICICTA)*, Xi'an, China, 2020, pp. 735-739, doi: 10.1109/ICICTA51737.2020.00160.

[37] W. Huang, C. Zhang, M. Dong and J. Zhou, "Research on partial discharge monitoring system of switchgear based on wireless distributed TEV sensors," *2017 1st International Conference on Electrical Materials and Power Equipment (ICEMPE)*, Xi'an, China, 2017, pp. 647-650.

[38] High-Voltage Test Techniques—Partial Discharge Measurements, IEC standard 60270, 2015.

[39] L. A. Renforth, R. Giussani, M. T. Mendiola and L. Dodd, "Online Partial Discharge Insulation Condition Monitoring of Complete High-Voltage Networks," in *IEEE Transactions on Industry Applications*, vol. 55, no. 1, pp. 1021-1029, Jan.-Feb. 2019, doi: 10.1109/TIA.2018.2866983.

[40] T. Shahsavarian *et al.*, "A Review of Knowledge-Based Defect Identification via PRPD Patterns in High Voltage Apparatus," in *IEEE Access*, vol. 9, pp. 77705-77728, 2021, doi: 10.1109/ACCESS.2021.3082858.

Yuan Yan was born in Sichuan, China, in 1996. He received the B.S. degree in electrical engineering from Xi'an Jiaotong University, Xi'an, China, in 2018.

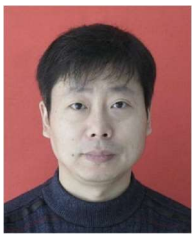
He is working toward a Ph.D. in electrical engineering with Xi'an Jiaotong University, Xi'an, China, and the Department of Electronics and Telecommunications, Politecnico di Torino, Turin, Italy. His research mainly focuses on the insulation condition monitoring of electrical equipment, electromagnetic modeling, and signal processing.

Riccardo Trinchero (M'16) received the M.Sc. and the Ph.D. degrees in Electronics and Communication Engineering from Politecnico di Torino, Torino, Italy, in 2011 and 2015, respectively.

He is currently an Associate Professor within the EMC Group with the Department of Electronics and Telecommunications at the Politecnico di Torino. His research interests include the analysis of switching DC-DC converters, machine learning, and statistical simulation of circuits and systems.

Igor S. Stievano (M'98–SM'08) received the master's degree in electronic engineering and the Ph.D. degree in electronics and communication engineering from the Politecnico di Torino, Turin, Italy, in 1996 and 2001, respectively.

He is currently a Professor of Electrical Engineering with the Department of Electronics and Telecommunications, Politecnico di Torino. He has authored or co-authored more than 130 papers published in international journals and conference proceedings. His research interests mainly relate to modeling and simulation of electrical and electronic systems.



Hongjie Li was born in Henan, China, in 1966. He received the B.S., M.S., and the Ph.D. degrees from Xi'an Jiaotong University, Xi'an, China, in 1989, 1992, and 1998, respectively. He is currently a professor at the High Voltage Division, School of Electrical Engineering, Xi' a Jiaotong University. His primary research interests include condition-based monitoring of electrical equipment, electromagnetic modeling, and numerical electromagnetic analysis.