Inter-Frequency Phase Difference for Phase Reconstruction Using Deep Neural Networks and Maximum Likelihood

Nguyen Binh Thien, Yukoh Wakabayashi, *Member, IEEE*, Kenta Iwai, *Member, IEEE*, and Takanobu Nishiura, *Member, IEEE*

***Abstract*—This paper presents improvements to two-stage algorithms for estimating the short-time Fourier transform (STFT) phase from only the amplitude by using deep neural networks (DNNs). The phase is difficult to reconstruct due to its sensitivity to the waveform shift and wrapping issue. To mitigate these problems, two-stage approaches indirectly estimate the phase through phase derivatives, i.e., instantaneous frequency (IF) and group delay (GD). In the first stage, the IF and GD are estimated from the amplitude using DNNs, and then in the second stage, the phase is reconstructed by maintaining the IF/GD information. Conventional methods for the second stage do not consider the importance of high-amplitude time–frequency bins, e.g., the least squares-based method, or lack a solid model, e.g., the average-based method. To address these problems, we propose improvements to the second stage of two-stage algorithms by using *von Mises* distribution-based maximum likelihood and weighted least squares. We also provide theoretical discussions for the phase reconstruction, including the investigations of the properties of the GD and roles of the IF/GD information in the inverse STFT. On the basis of the analysis, we propose a new phase-based feature, i.e., inter-frequency phase difference (IFPD), and demonstrate its application in two-stage phase reconstruction algorithms. We conducted subjective and objective experiments to compare the performances of our proposed and conventional methods. The results confirm that the proposed method using the IFPD performs better than other methods for all metrics.**

***Index Terms*—Two-stage phase estimation, instantaneous frequency, group delay, weighted least squares, von Mises distribution.**

Manuscript received 7 July 2022; revised 24 January 2023; accepted 10 April 2023. Date of publication 20 April 2023; date of current version 1 May 2023. This work was supported in part by the Ritsumeikan Advanced Research Academy (RARA), in part by Ritsumeikan Global Innovation Research Organization (R-GIRO), and in part by JSPS KAKENHI under Grants JP20K19827, JP19H04142, JP21H03488, JP21H04427 and JP21K18372. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Hakan Erdogan. *(Corresponding auhtor: Nguyen Binh Thien.)*

Nguyen Binh Thien is with the Graduate School of Information Science and Engineering, Ritsumeikan University, Shiga 525-8577, Japan (e-mail: gr0398xe@ed.ritsumei.ac.jp).

Yukoh Wakabayashi is with the Department of Computer Science and Engineering, Toyohashi University of Technology, Toyohashi 441-8580, Japan (e-mail: wakayuko@cs.tut.ac.jp).

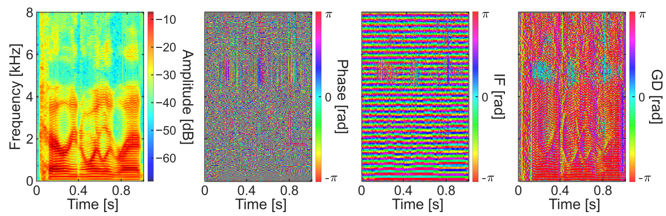
Kenta Iwai and Takanobu Nishiura are with the College of Information Science and Engineering, Ritsumeikan University, Shiga 525-8577, Japan (e-mail: iwai18sp@fc.ritsumei.ac.jp; nishiura@is.ritsumei.ac.jp).

Digital Object Identifier 10.1109/TASLP.2023.3268577

I. Introduction

PHASE processing, especially phase reconstruction, has gained considerable attention [1], [2], [3], [4], [5], [6], [7], [8], [9], [10] in the short-time Fourier transform (STFT)-based audio processing area. Most conventional STFT-based applications focus on reconstructing/modifying the amplitude, while the phase is largely neglected as it is difficult to handle due to the wrapping issue. However, a low-quality phase spectrogram may degrade the perceptual quality of the reconstructed signal [11], [12]. The phase also contains important information, which can be combined with the amplitude information as inputs for the DNN models to improve their performances [13], [14], [15], [16]. As a target to estimate, the phase reconstructed using the amplitude and observed noisy/mixed phase has been demonstrated to be useful in many applications including source separation [17], [18], [19] and speech enhancement [20], [21], [22]. In other contexts, when the amplitude spectrogram is artificially constructed (e.g., time-scale modification [23], speech synthesis [24], [25], [26], and audio restoration [27]), the observed phase does not exist, and the phase reconstruction has to be done using only the amplitude information.

Most former phase reconstruction approaches rely on the consistency property of the STFT, which originates from the redundancy of the information caused by the overlap of analysis windows. The approach proposed by Griffin and Lim [23] is the most well-known, which iteratively updates the phase estimate using the STFT and inverse STFT (ISTFT) while holding the amplitude information. Alternatively, [28] explicitly defines an inconsistency criterion and minimizes it with simplifications. Although yielding relatively good results, consistency-based approaches have several drawbacks; the whole amplitude spectrogram is required for each iteration, the convergence can be slow, and the reconstructed signals may contain artifacts such as echo or reverberation. Other phase reconstruction approaches based on signal modeling, including harmonic modeling, have been reported to achieve higher performance with a lower complexity in comparison with consistency-based approaches in various applications [19], [20], [21], [22], [27]. More recently, iterative algorithms use alternating direction method of multipliers [29] and direction map [30] to improve the reconstruction quality and convergence rate. In a non-iterative manner, [31] utilizes the direct relationship between the logarithm of the amplitude



(a) Amplitude (b) Phase (c) IF (d) GD

Fig. 1. Example of (c) instantaneous frequency and (d) group delay of a speech, where (a) and (b) are log-amplitude and phase spectrograms.

and partial derivatives of the phase of the un-sampled STFT with respect to the Gaussian window. Additional features, such as instantaneous frequency (IF) [32] and group delay (GD) [33], have also been used to assist the phase reconstruction [34], [35]. Other approaches [17], [18], [36], [37] model the phase by using deep neural networks (DNNs) to further benefit from the prior knowledge of the target signals.

One difficulty with DNN-based phase reconstruction is the wrapping issue. As the phase is wrapped in the range of $(-\pi, \pi]$, the conventional loss functions for regression, e.g., mean squared errors, become inefficient as they do not handle the periodicity. A solution for the wrapping issue is to use the *von Mises* distribution, which is a circular distribution. [38] and [39] are among the first studies to model the phase using the *von Mises* distribution for deriving a joint estimator of the amplitude and phase. Later, by using the same distribution, [36] proposed a cosine loss function for DNN-based phase estimation. Other approaches to deal with the wrapping issue are to cast the phase-regression problem into a classification problem of the quantized version of the phase [17], [18] or estimate the real and imaginary parts of the complex spectrogram instead of its amplitude and phase [40], [41]. However, there are also other problems for modeling the phase using DNNs, including the phase sensitivity to the waveform shift, i.e., only a small shift in the time domain can lead to a significant change in the phase spectrogram, especially at high frequencies. Another problem is sign indetermination [42], i.e., the STFTs of two signals $x(n)$ and $-x(n)$ have the same amplitudes but different phases. In other words, a given amplitude spectrogram may be consistent with both phase spectrograms $\boldsymbol{\Phi}$ and $\boldsymbol{\Phi} + \boldsymbol{J}\pi$, where $\boldsymbol{J}$ is an all-one matrix. The $\boldsymbol{\Phi}$ and $\boldsymbol{\Phi} + \boldsymbol{J}\pi$ usually yield very different values for most phase reconstruction loss functions; they are even opposite for the cosine loss function proposed in [36].

To mitigate these problems, two-stage approaches were proposed [4], [5], [6] for indirectly reconstructing the phase through the phase derivatives, i.e., the IF and GD. Although the phase changes quickly along the time and frequency, its change rate between neighboring elements is more stable. The IF and GD extract that change rate through the derivative operation, thereby reducing the sensitivity and wrapping issues and revealing the underlying structure of the phase, as illustrated in Fig. 1(c) and (d). The first stage is almost the same for two-stage phase reconstruction algorithms in that the IF and GD are estimated from the amplitude using DNNs. Not only are the IF and GD more structured than the phase, they are also not affected by the sign-indetermination problem because the ambiguity of $\boldsymbol{J}\pi$ becomes zero after the derivative operation. Therefore, the IF and GD are reconstructed much more easily than the phase itself. In the second stage, the phase is reconstructed from the IF and GD. Conventional methods for the second stage include the least squares (LS) [4], circular average [5], and maximum likelihood (ML) [6]. Experimental results [4] indicated the efficacy of such a two-stage approach over directly reconstructing the phase.

We focus on the two-stage approach for phase reconstruction from only the amplitude. The contributions of this paper are extensions of our preliminary study [6], including improvements to the current methods, theoretical discussions, and the introduction of our new phase-based feature, called inter-frequency phase difference (IFPD), that can be applied to phase reconstruction. In our previous study [6], we used Newton's method for solving the ML problem in the second stage. In this paper, we present improvements to this ML-based method in terms of calculation speed and convergence rate of the optimization algorithm by using a coordinate-descent strategy, which takes advantage of the separability of variables in the objective function. We also improve upon the LS-based method [4] by introducing amplitude weights, which reflect the importance of each time–frequency (TF) bin, to the error function and applying a tridiagonal system algorithm to reduce the calculation time. Comparisons among the approaches for the second stage are then discussed from theoretical aspects. We also present several new analyses for two-stage phase reconstruction algorithms. More specifically, we examine the properties of the GD and propose a GD normalization method for facilitating the training process of the GD-estimation model in the first stage. Unlike the method in [13] that subtracts the peak of the GD histogram, we introduce an analytic formula for normalizing the GD without requiring the training data for the histogram calculation. Another analysis is the investigation of a narrow-band signal to interpret the effects of the IF and GD information on the reconstructed waveforms using the ISTFT. From the discussions, we define the IFPD and use it to improve two-stage phase reconstruction algorithms. We further conduct a listening test in addition to other objective measurements to verify the performance of our proposed methods.

The remainder of the paper is organized as follows. We review related work in Section II, and analyze the phase properties for two-stage phase reconstruction algorithms in Section III. In Section IV, we describe our proposed methods and present the comparisons of them with conventional methods. In Section V, we present the IFPD and illustrate its application in phase reconstruction. In Section VI, we discuss the experiments on the efficacy of our proposed methods and present the results. Finally, we conclude the paper in Section VII.

II. Conventional Two-Stage Phase Reconstruction

This section starts by defining the notation and formulation. Then, we review the conventional two-stage phase reconstruction algorithms.

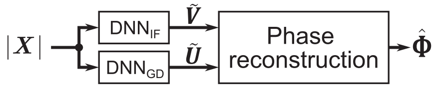


Fig. 2. Diagram of two-stage phase reconstruction algorithms. First stage consists of two DNNs for estimating IF $\tilde{\boldsymbol{V}}$ and GD $\tilde{\boldsymbol{U}}$. Second stage reconstructs phase $\hat{\boldsymbol{\Phi}}$ from $\tilde{\boldsymbol{V}}$ and $\tilde{\boldsymbol{U}}$.

A. Notation and Formulation

Let $X_{k,\ell}$ be the STFT of a discrete-time signal calculated with an M-sample window length, R-sample window shift, and N-point discrete Fourier transform (DFT), where $\ell \in \{0, \ldots, L-1\}$ and $k \in \{0, \ldots, K-1\}$ are the time frame index and frequency bin index, respectively. Its phase and amplitude are then denoted as $\Phi_{k,\ell} = \angle X_{k,\ell}$ and $|X_{k,\ell}|$, respectively, where $\angle$ is the angle operator.

IF is defined as the derivative of the phase with respect to time, which can be estimated by the phase difference as

$$V_{k,\ell} = \mathcal{P}(\Phi_{k,\ell+1} - \Phi_{k,\ell}), \tag{1}$$

where $\mathcal{P}(\cdot)$ is a wrapping function mapping a value into the principal range of $(-\pi, \pi]$. Similarly, GD, which is a negative frequency derivative of the phase, can be calculated as

$$U_{k,\ell} = \mathcal{P}(\Phi_{k,\ell} - \Phi_{k+1,\ell}). \tag{2}$$

We denote the vector notations of phase spectrum, IF, and GD at frame ℓ as $\boldsymbol{\phi}_\ell = (\Phi_{0,\ell}, \ldots, \Phi_{K-1,\ell})^\mathsf{T}$, $\boldsymbol{v}_\ell$, and $\boldsymbol{u}_\ell$, respectively, where $(\cdot)^\mathsf{T}$ is a matrix transposition operator. Thus, we have

$$\boldsymbol{v}_\ell = \boldsymbol{\phi}_{\ell+1} - \boldsymbol{\phi}_\ell, \tag{3}$$

$$\boldsymbol{u}_\ell = \boldsymbol{D}\boldsymbol{\phi}_\ell, \tag{4}$$

where $\boldsymbol{D}$ is a $(K-1)\times K$ upper bidiagonal matrix defined as

$$(\boldsymbol{D})_{i,j} = \begin{cases} 1, & \text{if } i = j \\ -1, & \text{if } i+1 = j \\ 0, & \text{otherwise} \end{cases}. \tag{5}$$

The matrix notations for the amplitude, phase, IF, and GD spectrograms are $|\boldsymbol{X}|$, $\boldsymbol{\Phi}$, $\boldsymbol{V}$, and $\boldsymbol{U}$, respectively. In two-stage phase reconstruction algorithms, the notations $\tilde{\cdot}$ and $\hat{\cdot}$ denote the estimates in the first and second stages, respectively, and $\cdot^*$ denotes the normalization (which is described in Sections II-B and III-A).

Two-stage phase reconstruction algorithms are aimed at estimating the phase $\boldsymbol{\Phi}$ from a given amplitude $|\boldsymbol{X}|$ indirectly through the IF $\boldsymbol{V}$ and GD $\boldsymbol{U}$, as illustrated in Fig. 2.

B. First Stage: IF/GD Estimation Using DNN

The first stage is similar for two-stage phase reconstruction algorithms, in which the IF $\tilde{\boldsymbol{V}}$ and GD $\tilde{\boldsymbol{U}}$ are reconstructed from the amplitude using *von Mises* distribution-based DNNs [36]. The *von Mises* distribution is also known as the circular normal distribution, which can be used to model circular data like the phase. Its probability density function is defined as

$$f(x|\mu,\kappa) = \frac{e^{\kappa\cos(x-\mu)}}{2\pi I_0(\kappa)}, \tag{6}$$

where x is a circular variable, μ is a measure of location, κ is a measure of concentration, and $I_0(\kappa)$ is a modified Bessel function of order 0. μ and $1/\kappa$ are analogous to the mean and variance of the normal distribution. The negative logarithm of (6) is given as

$$-\log f(x|\mu,\kappa) = -\kappa\cos(x-\mu) + \mathcal{C}, \tag{7}$$

where $\mathcal{C}$ is a constant to x. By modeling the IF/GD by the *von Mises* distribution with the assumption that κ is constant for all the data points, the error function for the DNNs reconstructing the IF/GD is defined as

$$\mathcal{L}_{\mathrm{DNN}}(\boldsymbol{y}_\ell, \tilde{\boldsymbol{y}}_\ell) = -\sum_k \cos(Y_{k,\ell} - \tilde{Y}_{k,\ell}), \tag{8}$$

where $\boldsymbol{y}_\ell$ and $\tilde{\boldsymbol{y}}_\ell$ are the original and estimated values of the output, which is either the IF or GD. To improve the DNN training process, the IF and GD are normalized so that their distributions have peaks at zero. [13] proposed an IF normalization method as

$$V^*_{k,\ell} = \mathcal{P}\left(V_{k,\ell} - 2\pi kR/N\right), \tag{9}$$

which removes the between-frame phase shift of $2\pi kR/N$ from the IF. [13] also proposed a GD normalization scheme that subtracts π from all of its elements, which is based on the observation that the GD histogram has a peak near π. [5] demonstrated that the IF and GD normalizations in [13] are useful for the DNN training process of two-stage phase reconstruction algorithms.

C. Second Stage: Phase Estimation From IF and GD

In the second stage, the phase $\hat{\boldsymbol{\Phi}}$ is reconstructed from the estimated IF $\tilde{\boldsymbol{V}}$ and GD $\tilde{\boldsymbol{U}}$. We briefly describe two conventional methods for the second stage, LS-based [4] and weighted circular average-based [5].

1) Least Squares: Inspired by the LS-based method for 2D-phase unwrapping in [43], [4] proposed recursively reconstructing the phase from the IF and GD for each frame by minimizing the quadratic error function defined as

$$\mathcal{L}_{\mathrm{LS}}(\boldsymbol{\phi}_\ell) = \|\boldsymbol{\phi}_\ell - \boldsymbol{\phi}_{\ell-1} - \tilde{\boldsymbol{v}}_{\ell-1}\|^2 + \|\boldsymbol{D}\boldsymbol{\phi}_\ell - \tilde{\boldsymbol{u}}_\ell\|^2. \tag{10}$$

When estimating $\boldsymbol{\phi}_\ell$, $\boldsymbol{\phi}_{\ell-1}$ is replaced with the wrapped version of its previously estimated value, i.e., $\hat{\boldsymbol{\phi}}^{\mathcal{P}}_{\ell-1} = \mathcal{P}(\hat{\boldsymbol{\phi}}_{\ell-1})$. Since $\tilde{\boldsymbol{v}}_\ell$ and $\tilde{\boldsymbol{u}}_\ell$ are also wrapped, which may lead to a detrimental effect on the LS solution, [4] proposed modifying $\tilde{\boldsymbol{u}}_\ell$ in (10) as

$$\tilde{\boldsymbol{u}}_\ell \leftarrow \boldsymbol{D}(\hat{\boldsymbol{\phi}}^{\mathcal{P}}_{\ell-1} + \tilde{\boldsymbol{v}}_{\ell-1}) + \mathcal{P}\left(\tilde{\boldsymbol{u}}_\ell - \boldsymbol{D}(\hat{\boldsymbol{\phi}}^{\mathcal{P}}_{\ell-1} + \tilde{\boldsymbol{v}}_{\ell-1})\right). \tag{11}$$

(11) adds 2π jumps to $\tilde{\boldsymbol{u}}_\ell$ to make it more consistent with the GD calculated from the previously estimated phase and IF, i.e., $\boldsymbol{D}(\hat{\boldsymbol{\phi}}^{\mathcal{P}}_{\ell-1} + \tilde{\boldsymbol{v}}_{\ell-1})$. The solution for minimizing (10) is

$$\hat{\boldsymbol{\phi}}_\ell = (\boldsymbol{I}_K + \boldsymbol{D}^\mathsf{T}\boldsymbol{D})^{-1}(\hat{\boldsymbol{\phi}}^{\mathcal{P}}_{\ell-1} + \tilde{\boldsymbol{v}}_{\ell-1} + \boldsymbol{D}^\mathsf{T}\tilde{\boldsymbol{u}}_\ell), \tag{12}$$

where $\boldsymbol{I}_K$ is a $K \times K$ identity matrix.

2) Circular Average: By incorporating the amplitude information, [5] proposed a simple weighted circular average-based method that estimates the phase for each TF bin as

$$\hat{\Phi}_{k,\ell} = \angle \sum_{q=1}^{Q} W_{k,\ell}^{(q)} \exp\left(j\varphi_{k,\ell}^{(q)}\right), \tag{13}$$

where $\varphi_{k,\ell}^{(q)}$ is an estimate of $\Phi_{k,\ell}$ computed from the IF, GD, and the qth previously estimated phase element near $\Phi_{k,\ell}$. $W_{k,\ell}^{(q)}$ is the amplitude weight, and Q is the number of the neighbors involved. [5] also empirically determined that $Q = 3$ yields the best result, i.e., $\hat{\Phi}_{k,\ell}$ is calculated from $\hat{\Phi}_{k-1,\ell}$, $\hat{\Phi}_{k,\ell-1}$, and $\hat{\Phi}_{k+1,\ell-1}$.

III. Phase Analysis for Two-Stage Phase Reconstruction Algorithms

In this section, we provide theoretical discussions for two-stage phase reconstruction algorithms. Section III-A analyzes the properties of the phase and GD calculated with two types of the window functions and introduces our analytic GD normalization formula. Section III-B investigates the effects of phase modifications on the ISTFT of a sinusoidal wave to interpret how the IF and GD are useful for phase reconstruction.

A. GD Analysis and Normalization

For normalizing the GD, [13] proposed subtracting the peak π of the GD histogram. However, this peak varies with the window length and number of DFT points. Instead of following the method in [13], we analyze the GD values and introduce an analytic formula for GD normalization as follows.

When calculating the STFT, we usually multiply each signal frame by a window function before calculating the DFT. This is equivalent to a convolution in the frequency domain, as

$$\boldsymbol{x}_\ell = \boldsymbol{s}_\ell * \boldsymbol{w}, \tag{14}$$

where $\boldsymbol{x}_\ell$, $\boldsymbol{s}_\ell$, and $\boldsymbol{w}$ are the DFT spectra of the windowed signal, target signal at frames ℓ, and window function, respectively. Equivalently, elements of $\boldsymbol{x}_\ell$ can be calculated as

$$x_\ell(k) = \sum_{m=0}^{N-1} s_\ell(m) w(k-m). \tag{15}$$

From (14), we can see that the phase of $\boldsymbol{x}_\ell$ is affected by $\boldsymbol{w}$. The phase of $\boldsymbol{w}$ relies not only on the type of the window function but also on the position (in other words, the time origin) of the window function in the DFT formula. Fig. 3(a) to (c) shows an example of the Hamming window at two positions in the time and frequency domains, i.e., that starting from zero (typically used in STFT implementations, as shown with blue lines) and that centered at zero (as shown with red lines). When the window function is symmetric and centered at zero (in other words, the time origin of the DFT formula is at the center of the frame), $\boldsymbol{w}$ is a real-valued vector. From (15), we see that each element $x_\ell(k)$ is a linear combination of all the elements of $\boldsymbol{s}_\ell$. However, the contributions of the elements of $\boldsymbol{s}_\ell$ at frequency bins far from bin k are scaled down by the low side lobes of $\boldsymbol{w}$, as shown in Fig. 3(b). As a result, the phase of $\boldsymbol{x}_\ell$ at each frequency bin is mostly dominated by the nearest strong component of $\boldsymbol{s}_\ell$. By shifting the zero-centered window to the right side by $(M-1)/2$ samples, we obtain a zero-starting window, i.e., the time origin of the DFT formula is at the beginning of the frame. In this case, the amplitude is the same, but the phase at each frequency bin k will be shifted by $-\frac{2\pi k}{N}\frac{M-1}{2}$ compared with the case of a zero-centered window, as illustrated in Fig. 3(b) and (c).

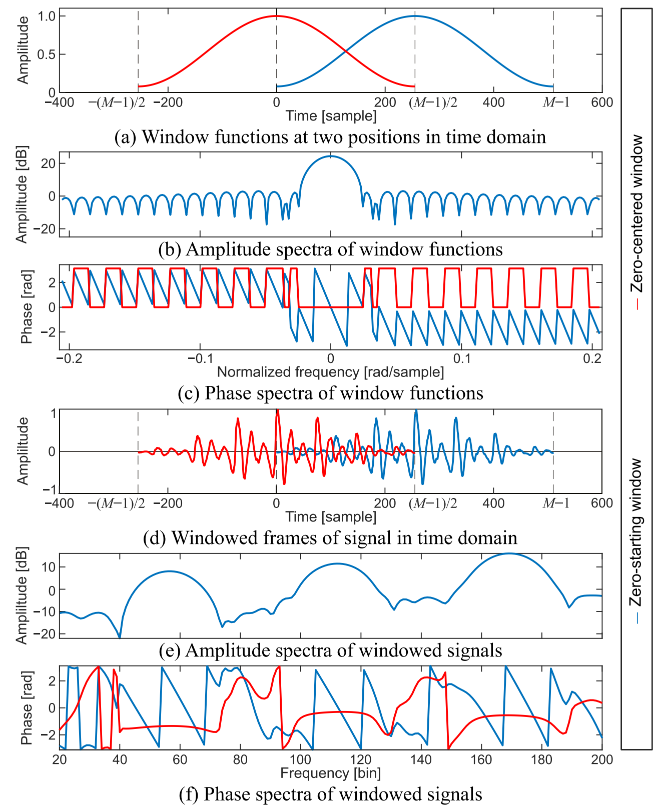


Fig. 3. Example of Hamming window at two positions and windowed frame of speech signal in time and frequency domains. Window length and number of DFT points are $M = N = 511$.[1]

Fig. 3(d) to (f) shows an example of a windowed frame of a speech sample calculated using the zero-centered and zero-starting windows. For the zero-centered window, we see that phase elements around a strong component (an amplitude peak), e.g., around the frequency bin of 56, are dominated by that component, hence nearly constant. As a result, the GD, which is a negative frequency-derivative of the phase, is approximately zero. For the zero-starting window, the phase at those frequency bins becomes an oblique line due to the linear phase shift, and the GD will be a non-zero constant. These properties are similar for weak components of the signal, although the affected area around a weak component is narrower than that around a strong component.

[1]In this example, M was set to an odd number so that the zero-centered window is calculated by setting the time origin to the center of the windowed frame when calculating the DFT. However, M can also be even.

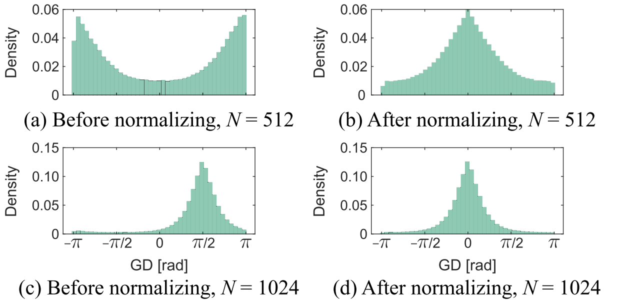


Fig. 4. Examples of GD histograms of speech sample before and after normalization. STFT is calculated with $M = 512$ and various N.

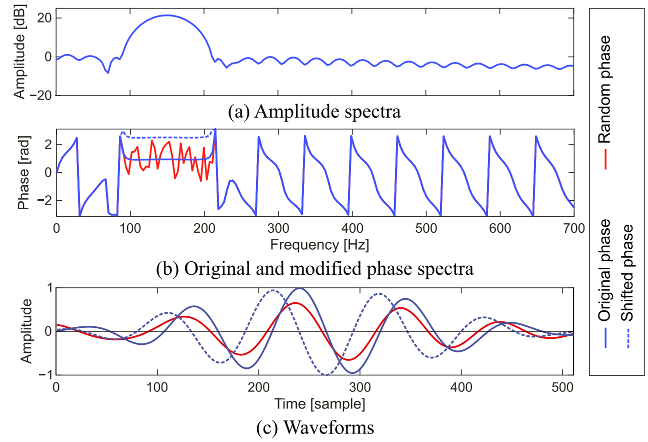


Fig. 5. Example of effects of phase modifications on IDFT. DFT of 32-ms frame of 150-Hz sinusoidal wave is calculated with Hamming window. Sampling frequency is 16 kHz. Original phase spectrum (blue solid line) is modified by adding random numbers (red solid line) or constant of $\pi/2$ (blue dashed line). Note that only area around 150 Hz is modified because other areas do not have much of effect on IDFT.

From the above discussion, we can see that the GD values calculated using the zero-centered window naturally concentrate around zero. The linear phase shift introduced by the commonly used zero-starting window shifts the peak of the GD histogram to a non-zero constant. We hence propose normalizing the GD by compensating for that linear phase shift using either the following methods.

- Circular-shift the windowed signal to the left by $(M-1)/2$ samples when calculating the DFT.
- Compensate for the phase shift by

$$\Phi^*_{k,\ell} = \mathcal{P}\left(\Phi_{k,\ell} + \frac{2\pi k}{N}\frac{(M-1)}{2}\right). \tag{16}$$

- Directly modify the GD by

$$U^*_{k,\ell} = \mathcal{P}\left(U_{k,\ell} - \frac{\pi(M-1)}{N}\right). \tag{17}$$

It is worth noting that $(M-1)/2$ is not necessarily an integer; hence, M can be either even or odd. (17) is similar to the GD normalization formula proposed by [13] in terms of subtracting a number from the GD. However, when $M = N$, instead of subtracting π as with the method in [13], we can see from (17) that the subtrahend is $\pi(M-1)/N$. The advantage of (17) is that it can be applied to other settings of M and N without requiring calculating the GD histogram of the training data. Fig. 4 shows examples of GD histograms of a speech sample before and after normalizing using (17).

B. IF and GD Information in Phase Reconstruction

Two-stage phase reconstruction algorithms indeed reconstruct the phase by maintaining the phase relationships between TF bins along time and frequency through the IF and GD, respectively. Since the ISTFT has the form of a sum of complex numbers, if the phase relationships between those complex numbers are maintained, i.e., the phase differences between TF bins remain unchanged, the amplitude of the reconstructed signal will be consistent even if the phase is shifted. To investigate the role of the phase relationships in the ISTFT, we modify the phase spectra calculated from a sinusoidal wave and observe the effects on the reconstructed waveform as follows.

Fig. 5 shows the DFT spectra and waveforms of the sinusoidal wave, in which the phase spectrum is modified in a frequency range by using two methods: 1) adding random numbers to each element, i.e., the phase relationships along the frequency are broken, and 2) adding a constant of $\pi/2$ to all the elements, i.e., the phase relationships along the frequency are maintained. Each phase spectrum is then combined with the amplitude spectrum to reconstruct the waveform using the inverse DFT (IDFT). We can see from Fig. 5(c) that the reconstructed waveform for the maintained phase relationships has almost the same amplitude as the original waveform, although it is shifted in the time domain. In contrast, the waveform of the randomly modified phase has a lower amplitude due to the misalignment of the complex spectral bins in the IDFT calculation.

Fig. 6 shows an example of two consecutive frames in the same signal used above. The phase spectra are shifted by adding a constant, which is $\pi/2$ in this example, to all the elements. The IDFTs of the original and modified versions of the first and second frames are then combined for each pair using the overlap-add method. We can see from this figure that, if the phase spectra of both frames are shifted the same way, i.e., the IF information is maintained, their waveforms are aligned well for the overlap-add to yield the same amplitude as the original signal. In other situations, when the phase spectrum of only one frame is modified, the misalignments between the reconstructed waveforms of the two frames decrease the amplitude of the overlap-added signals. Those misalignments also cause a frequency modulation to the overlap-added signals, as illustrated by the period changes in Fig. 6(e).

From the above discussion, we can see that the distortions in the IF and GD information result in the degradation in both the amplitude and frequency of the signal. When the phase relationships between TF bins are maintained, even if the phase is shifted, the reconstructed signal still has the same amplitude as the original signal. The shifted phase only introduces a shift of the signal in the time domain, which, based on our observation, makes no difference in perception of sound quality.

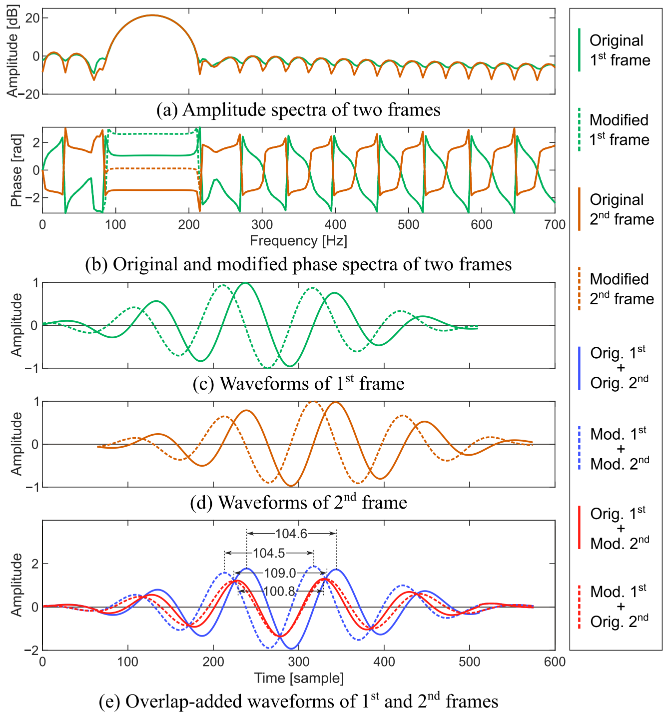


Fig. 6. Example of effects of phase modifications on overlap-add. Signal and DFT setting are same as in Fig. 5, in which two consecutive frames with hop of 4 ms are shown. Phase spectra are modified by adding $\pi/2$ to all elements.

IV. Proposed Phase Reconstruction Methods

All two-stage phase reconstruction algorithms have the same first stage that estimates the IF and GD from the amplitude using DNNs. In this section, we propose two approaches for reconstructing the phase from the IF and GD in the second stage, i.e., weighted LS-based (Section IV-A) and ML-based using *von Mises* distribution (Section IV-B). The ML-based methods are also divided into two optimization approaches, i.e., using Newton's method and coordinate descent. We then present the comparison of the proposed and conventional methods in theoretical aspects (Section IV-C).

A. Weighted Least Squares

The contribution of each TF bin to the ISTFT depends highly on its amplitude. We also observed that errors of the IF and GD reconstructed in the first stage are low at high-amplitude positions. Therefore, we improved the conventional LS-based method [4] by adding amplitude weights to the error function (10) to emphasize the importance of the high-amplitude TF bins. The weighted error function is defined as

$$\mathcal{L}_{\mathrm{WLS}}(\boldsymbol{\phi}_\ell) = \left\| \sqrt{\breve{\boldsymbol{W}}^{v}_{\ell-1}} (\boldsymbol{\phi}_\ell - \boldsymbol{\phi}_{\ell-1} - \tilde{\boldsymbol{v}}_{\ell-1}) \right\|^2 + \left\| \sqrt{\breve{\boldsymbol{W}}^{u}_{\ell}} (\boldsymbol{D}\boldsymbol{\phi}_\ell - \tilde{\boldsymbol{u}}_\ell) \right\|^2, \tag{18}$$

where $\breve{\boldsymbol{W}}^{v}_{\ell}$ and $\breve{\boldsymbol{W}}^{u}_{\ell}$ are diagonal weight matrices of the IF $\tilde{\boldsymbol{v}}_\ell$ and GD $\tilde{\boldsymbol{u}}_\ell$, respectively. The kth element on the main diagonal of both $\breve{\boldsymbol{W}}^{v}_{\ell}$ and $\breve{\boldsymbol{W}}^{u}_{\ell}$ are empirically set to $|X_{k,\ell}|^p$. The power of $p\,(p \geq 1)$ is used to further separate the low- and high-amplitude TF bins. We also use the GD modification method (11) to address the wrapping issue.

The analytic solution for minimizing (18) is

$$\hat{\boldsymbol{\phi}}_\ell = (\breve{\boldsymbol{W}}^{v}_{\ell-1} + \boldsymbol{D}^\mathsf{T}\breve{\boldsymbol{W}}^{u}_{\ell}\boldsymbol{D})^{-1} \cdot (\breve{\boldsymbol{W}}^{v}_{\ell-1}(\hat{\boldsymbol{\phi}}^{\mathcal{P}}_{\ell-1} + \tilde{\boldsymbol{v}}_{\ell-1}) + \boldsymbol{D}^\mathsf{T}\breve{\boldsymbol{W}}^{u}_{\ell}\tilde{\boldsymbol{u}}_\ell). \tag{19}$$

The derivation of this solution is explained in Appendix. Most of the calculation time of (19) is spent in calculating the inverse of a $K \times K$ matrix. However, we can see that (19) has a form of $\hat{\boldsymbol{\phi}}_\ell = \boldsymbol{A}^{-1}\boldsymbol{b}$, where $\boldsymbol{A}$ is a matrix and $\boldsymbol{b}$ is a vector. Moreover, the matrix $\boldsymbol{A} = \breve{\boldsymbol{W}}^{v}_{\ell-1} + \boldsymbol{D}^\mathsf{T}\breve{\boldsymbol{W}}^{u}_{\ell}\boldsymbol{D}$ is a symmetric tridiagonal matrix. For that reason, we apply the tridiagonal system algorithm [44] to calculate $\boldsymbol{A}^{-1}\boldsymbol{b}$ with a complexity of $O(K)$ (instead of the $O(K^3)$ required by the Gaussian elimination for a non-tridiagonal matrix $\boldsymbol{A}$), thus significantly reducing the calculation time. The pseudo-code for this method is given in Algorithm 1.

Algorithm 1: Pseudo-Code of Weighted LS-Based Method for Reconstructing Phase from IF and GD.

Input: Amplitude $|\boldsymbol{X}|$, estimated IF $\tilde{\boldsymbol{V}}$ and GD $\tilde{\boldsymbol{U}}$
Output: Phase spectrogram $\hat{\boldsymbol{\Phi}}$
  $\hat{\Phi}_{0,0} \leftarrow 0$
  $\hat{\Phi}_{k,0} \leftarrow \hat{\Phi}_{k-1,0} - \tilde{U}_{k-1,0}$, for $k \in \{1, \ldots, K-1\}$
  for $\ell \in \{1, \ldots, L-1\}$ **do**
    $\hat{\boldsymbol{\phi}}^{\mathcal{P}}_{\ell-1} \leftarrow \mathcal{P}(\hat{\boldsymbol{\phi}}_{\ell-1})$
    Update $\tilde{\boldsymbol{u}}_\ell$ as in (11)
    Calculate $\hat{\boldsymbol{\phi}}_\ell$ as in (19)

B. Maximum Likelihood Using Von Mises Distribution

The LS-based methods for the second stage are greatly affected by the wrapping issue with the periodic variables such as the phase, IF, and GD. To address this issue, we propose an ML-based approach using a circular distribution, i.e., the *von Mises* distribution. In addition, the use of the *von Mises* distribution for the second stage makes it consistent with the first stage since, in the first stage, the IF/GD are also modeled using the same distribution.

We define a model as

$$\tilde{y} = \boldsymbol{d}^\mathsf{T}\boldsymbol{\psi} + \varepsilon, \tag{20}$$

where $\tilde{y}$ is the element of either $\tilde{\boldsymbol{V}}$ or $\tilde{\boldsymbol{U}}$, $\boldsymbol{\psi}$ is the flattened vector of the phase spectrogram $\boldsymbol{\Phi}$, $\boldsymbol{d}$ is a corresponding vector consisting of 0, 1, and -1 [similar to the matrix $\boldsymbol{D}$ in (5)], and ε is the residual of the model. Unlike the first stage that models the distribution of the IF/GD conditioned on the amplitude to train the DNN parameters, we define a *von Mises* distribution over $\tilde{y}$ conditioned on $\boldsymbol{d}$, and the phase $\boldsymbol{\psi}$ becomes the parameter of the model to be fitted. In other words, $p(\tilde{y}|\boldsymbol{d};\boldsymbol{\psi})$ is equal to a *von Mises* distribution, the measure of location of which is $\hat{y} = \boldsymbol{d}^\mathsf{T}\boldsymbol{\psi}$.

From (6), by taking the negative logarithm of $p(\tilde{y}|\boldsymbol{d};\boldsymbol{\psi})$ of all the elements of $\tilde{\boldsymbol{V}}$ and $\tilde{\boldsymbol{U}}$ with the assumption of a constant concentration κ, we derive an error function for the whole phase spectrogram as

$$\mathcal{L}_{\mathrm{ML}}(\boldsymbol{\Phi}) = -\sum_{k,\ell}\Big(W^{u}_{k,\ell}\cos(\tilde{U}_{k,\ell}-\hat{U}_{k,\ell}) + W^{v}_{k,\ell}\cos(\tilde{V}_{k,\ell}-\hat{V}_{k,\ell})\Big), \tag{21}$$

where $W^u_{k,\ell}$ and $W^v_{k,\ell}$ are the weights of the GD and IF at TF bin (k,ℓ), respectively, which are empirically selected as $W^u_{k,\ell} = W^v_{k,\ell} = |X_{k,\ell}|$. Thanks to the cosine functions, (21) is not affected by the wrapping issue of $\tilde{U}_{k,\ell}$ and $\tilde{V}_{k,\ell}$ as with the LS-based methods.

The partial derivative of (21) is given by

$$\frac{\partial \mathcal{L}_{\mathrm{ML}}}{\partial \Phi_{k,\ell}} = \sin(\Phi_{k,\ell})C_{k,\ell} - \cos(\Phi_{k,\ell})S_{k,\ell}, \tag{22}$$

where

$$C_{k,\ell} = W^v_{k,\ell}\cos(\Phi_{k,\ell+1}-\tilde{V}_{k,\ell}) + W^v_{k,\ell-1}\cos(\Phi_{k,\ell-1}+\tilde{V}_{k,\ell-1}) + W^u_{k,\ell}\cos(\Phi_{k+1,\ell}+\tilde{U}_{k,\ell}) + W^u_{k-1,\ell}\cos(\Phi_{k-1,\ell}-\tilde{U}_{k-1,\ell}), \tag{23}$$

and $S_{k,\ell}$ is defined the same as $C_{k,\ell}$, in which all the cosine functions are replaced with sine functions. Note that, at the boundaries of $k=0$, $k=K-1$, $\ell=0$, and $\ell=L-1$, we remove the terms containing the indices of $k-1$, $k+1$, $\ell-1$, and $\ell+1$, respectively, from $C_{k,\ell}$ and $S_{k,\ell}$.

It is impossible to find the analytic solution of the equation setting the gradient vector of $\mathcal{L}_{\mathrm{ML}}(\boldsymbol{\Phi})$ to zero. On the basis of several properties of the error function, we propose the following two optimization approaches.

1) Using Newton's Method: The first approach is to break (21) into frames so that the Hessian matrix becomes tridiagonal. Considering only the terms containing the phase at frame ℓ in $\mathcal{L}_{\mathrm{ML}}(\boldsymbol{\Phi})$, the frame-wise error function is given by

$$\mathcal{L}_{\mathrm{MLF}}(\boldsymbol{\phi}_\ell) = -\sum_{k}\Big(W^{u}_{k,\ell}\cos(\tilde{U}_{k,\ell}-\hat{U}_{k,\ell}) + W^{v}_{k,\ell-1}\cos(\tilde{V}_{k,\ell-1}-\hat{V}_{k,\ell-1}) + W^{v}_{k,\ell}\cos(\tilde{V}_{k,\ell}-\hat{V}_{k,\ell})\Big). \tag{24}$$

The gradient vector $\nabla_{\boldsymbol{\phi}_\ell}\mathcal{L}_{\mathrm{MLF}}(\boldsymbol{\phi}_\ell)$ can be calculated with the kth element identical to (22). We can see from (22) that the partial derivative with respect to $\Phi_{k,\ell}$ contains only two phase elements of the same frame, i.e., $\Phi_{k+1,\ell}$ and $\Phi_{k-1,\ell}$. Consequently, the Hessian matrix of $\mathcal{L}_{\mathrm{MLF}}(\boldsymbol{\phi}_\ell)$, denoted as $\boldsymbol{H}$, is a symmetric tridiagonal matrix, the element on the main diagonal of which is given by

$$\frac{\partial^2 \mathcal{L}_{\mathrm{MLF}}}{\partial \Phi^2_{k,\ell}} = \cos(\Phi_{k,\ell})C_{k,\ell} + \sin(\Phi_{k,\ell})S_{k,\ell}, \tag{25}$$

and the element on the first diagonal above (or below) is

$$\frac{\partial^2 \mathcal{L}_{\mathrm{MLF}}}{\partial \Phi_{k,\ell}\partial \Phi_{k+1,\ell}} = -W^u_{k,\ell}\cos(\Phi_{k,\ell}-\Phi_{k+1,\ell}-\tilde{U}_{k,\ell}). \tag{26}$$

Algorithm 2: Pseudo-Code of ML-Based Method Using Newton's Method for Reconstructing Phase from IF and GD.

Input: Amplitude spectrogram $|\boldsymbol{X}|$, estimated IF $\tilde{\boldsymbol{V}}$ and GD $\tilde{\boldsymbol{U}}$, number of iterations N_1 and N_2
Output: Phase spectrogram $\hat{\boldsymbol{\Phi}}$
$\hat{\Phi}_{0,0} \leftarrow 0$
$\hat{\Phi}_{k,0} \leftarrow \hat{\Phi}_{k-1,0} - \tilde{U}_{k-1,0}$, for $k \in \{1,\ldots,K-1\}$
for $\ell \in \{1,\ldots,L-1\}$ **do**
  $\hat{\boldsymbol{\phi}}_\ell \leftarrow \hat{\boldsymbol{\phi}}_{\ell-1} + \tilde{\boldsymbol{v}}_{\ell-1}$
  for $i \in \{1,\ldots,N_1\}$ **do**
    Update $\hat{\boldsymbol{\phi}}_\ell$ as in (27) removing terms containing $\Phi_{k,\ell+1}$ from $C_{k,\ell}$ and $S_{k,\ell}$
for $i \in \{1,\ldots,N_2\}, \ell \in \{0,\ldots,L-1\}$ **do**
  Update $\hat{\boldsymbol{\phi}}_\ell$ as in (27)

The tridiagonality of the Hessian matrix motivates us to use Newton's method to update the phase estimate. However, there is a problem that $\boldsymbol{H}$ is often not positive definite as $\mathcal{L}_{\mathrm{MLF}}(\boldsymbol{\phi}_\ell)$ is periodic. To solve this problem, we apply a regularization strategy, as in a previous study [45], to update the phase estimate as

$$\hat{\boldsymbol{\phi}}_\ell \leftarrow \hat{\boldsymbol{\phi}}_\ell - (\boldsymbol{H}+\gamma\boldsymbol{I}_K)^{-1}\nabla_{\boldsymbol{\phi}_\ell}\mathcal{L}_{\mathrm{MLF}}(\hat{\boldsymbol{\phi}}_\ell), \tag{27}$$

where γ is a damping factor. $\gamma=0$ is equivalent to no regularization. When γ is large, $\boldsymbol{H}$ is dominated by $\gamma\boldsymbol{I}_K$, and (27) approximates the standard gradient descent with the updating rate of $1/\gamma$. Ideally, γ is adaptive so that it is large enough to offset the negative eigenvalues of $\boldsymbol{H}$. We calculate γ from the smallest eigenvalue λ of $\boldsymbol{H}$ for each update as

$$\gamma = \begin{cases} -\beta\lambda, & \text{if } \lambda < 0 \\ 0, & \text{otherwise} \end{cases}, \tag{28}$$

where β is a scaling constant. We can efficiently estimate only the smallest eigenvalue of the tridiagonal matrix $\boldsymbol{H}$ as in [46]. As $\boldsymbol{H}+\gamma\boldsymbol{I}_K$ is also tridiagonal, the complexity of (27) can be reduced from $O(K^3)$ to $O(K)$ by using the tridiagonal system algorithm, similar to what was mentioned in Section IV-A.

In the update (27) for $\boldsymbol{\phi}_\ell$, the phase at the next and previous frames for calculating $C_{k,\ell}$ and $S_{k,\ell}$ are replaced with their estimates, i.e., $\hat{\boldsymbol{\phi}}_{\ell+1}$ and $\hat{\boldsymbol{\phi}}_{\ell-1}$. However, those estimates are not available at the beginning. We found that a random initialization may lead to slow convergence and poor results. Therefore, for the first several iterations, we recursively reconstruct the phase $\boldsymbol{\phi}_\ell$ by using only $\hat{\boldsymbol{\phi}}_{\ell-1}$. In other words, we remove the terms containing $\Phi_{k,\ell+1}$ from $C_{k,\ell}$ and $S_{k,\ell}$ when calculating (27). This is equivalent to removing the terms of $W^v_{k,\ell}\cos(\tilde{V}_{k,\ell}-\hat{V}_{k,\ell})$ from the error function (24). The full version of (27) is then used to smooth the phase estimate, i.e., $\boldsymbol{\phi}_\ell$ is updated using both $\hat{\boldsymbol{\phi}}_{\ell-1}$ and $\hat{\boldsymbol{\phi}}_{\ell+1}$. The pseudo-code for this method is given in Algorithm 2.

2) Using Coordinate Descent: Another approach for minimizing (21) is based on its separability property. From (1) and (2), we can see that each phase element $\Phi_{k,\ell}$ is only present in at most four terms in the sum of $\mathcal{L}_{\mathrm{ML}}$, i.e., the terms containing

Algorithm 3: Pseudo-Code of ML-Based Method Using Coordinate Descent for Reconstructing Phase from IF and GD.

Input: Amplitude spectrogram $|\boldsymbol{X}|$, estimated IF $\tilde{\boldsymbol{V}}$ and GD $\tilde{\boldsymbol{U}}$, number of iterations N_1 and N_2
Output: Phase spectrogram $\hat{\boldsymbol{\Phi}}$
$\hat{\Phi}_{0,0} \leftarrow 0$
$\hat{\Phi}_{k,0} \leftarrow \hat{\Phi}_{k-1,0} - \tilde{U}_{k-1,0}$, for $k \in \{1,\ldots,K-1\}$
for $\ell \in \{1,\ldots,L-1\}$ **do**
  $\hat{\boldsymbol{\phi}}_\ell \leftarrow \hat{\boldsymbol{\phi}}_{\ell-1} + \tilde{\boldsymbol{v}}_{\ell-1}$
  for $i \in \{1,\ldots,N_1\}$, $k \in \{0,\ldots,K-1\}$ **do**
    Update $\hat{\Phi}_{k,\ell}$ as in (29) removing terms containing $\Phi_{k,\ell+1}$ from $C_{k,\ell}$ and $S_{k,\ell}$
for $i \in \{1,\ldots,N_2\}$, $\ell \in \{0,\ldots,L-1\}$, $k \in \{0,\ldots,K-1\}$ **do**
  Update $\hat{\Phi}_{k,\ell}$ as in (29)

$\hat{V}_{k,\ell}$, $\hat{V}_{k,\ell-1}$, $\hat{U}_{k,\ell}$, and $\hat{U}_{k-1,\ell}$. In other words, varying $\Phi_{k,\ell}$ will change only those terms and will not have much of an effect on the optimal states of other phase elements in $\mathcal{L}_{\mathrm{ML}}$. Therefore, we use a coordinate-descent strategy [47] that sequentially minimizes $\mathcal{L}_{\mathrm{ML}}$ for each $\Phi_{k,\ell}$ where all other phase elements are fixed.

As $S_{k,\ell}$ and $C_{k,\ell}$ are independent from $\Phi_{k,\ell}$, we can easily set the first derivative $\partial\mathcal{L}_{\mathrm{ML}}/\partial\Phi_{k,\ell}$ to zero and check the second derivative to find the minimum. The solution is

$$\hat{\Phi}_{k,\ell} = \begin{cases} \arctan\left(S_{k,\ell}/C_{k,\ell}\right), & \text{if } f'' > 0 \\ \arctan\left(S_{k,\ell}/C_{k,\ell}\right) + \pi, & \text{otherwise} \end{cases}, \tag{29}$$

where $f'' = \partial^2\mathcal{L}_{\mathrm{ML}}/\partial\Phi_{k,\ell}^2$, which is identical to (25).

(29) is sequentially calculated throughout the whole spectrogram. Because the update of $\hat{\Phi}_{k,\ell}$ affects the optimal states of other phase elements around it, we need several iterations to make all the elements converge. Similar to the approach using Newton's method, for the first several iterations, we remove from $C_{k,\ell}$ and $S_{k,\ell}$ the terms containing $\Phi_{k,\ell+1}$ when calculating the solution of (29). The pseudo-code for this coordinate-descent approach is illustrated in Algorithm 3.

C. Comparison of Methods for Second Stage

In this subsection, we present several theoretical comparisons among the methods for the second stage of two-stage phase reconstruction algorithms, i.e., the conventional LS-based method, conventional circular average-based method, our weighted LS-based method, and our ML-based method with the two optimization schemes.

1) Least squares [4] and Weighted Least Squares: Our weighted LS-based method differs from the conventional LS-based method only in terms of the weights in the error function, which results in changes in the calculation of the solution, especially the matrix inversion. In the solution (12) of the conventional LS-based method, we can calculate the inverse of the matrix $(\boldsymbol{I}_K + \boldsymbol{D}^\mathsf{T}\boldsymbol{D})$ in advance as it is constant. For our weighted LS-based method, the matrix $(\boldsymbol{W}^v_{\ell-1} + \boldsymbol{D}^\mathsf{T}\boldsymbol{W}^u_\ell\boldsymbol{D})$ in (19) depends on the weights; hence, its inverse must be computed for each frame. However, thanks to the tridiagonality property of the matrix, (19) can be calculated with a complexity of $O(K)$, which is the same as (12).

2) Least Squares and Maximum Likelihood: As the LS-based methods can also be interpreted as the ML, LS- and ML-based methods differ in the distributions used, i.e., Gaussian and *von Mises* distributions. The *von Mises* distribution seems to be more efficient as it handles the wrapping issue and is the same distribution used in the first stage. The LS-based method is greatly affected by the wrapping issue. Although (11) is used to mitigate this issue, it may not be reliable when the errors of $\tilde{U}$ and/or $\tilde{V}$ are high. However, the advantage of the LS-based methods is that they yield a unique solution, while the ML-based methods require iterative methods for the optimization.

3) Circular average [5] and Maximum Likelihood: Like our *von Mises* distribution-based ML-based method, the circular average-based method is not affected by the wrapping issue. However, it consists of a single pass of recursively calculating each phase element using the average operation. This makes the phase estimate at each TF bin heavily dependent on the IF, GD, and other previously estimated phase elements nearby. Therefore, the circular average-based method may not be efficient when the estimated IF and GD have high errors, or when the underlying components of the signal are not stable. In contrast, our ML-based methods define a solid optimization problem, which can be solved using various optimization techniques. Although for the first several iterations, our ML-based methods also reconstruct the phase recursively, the later iterations help to compromise the IF and GD errors at all TF bins, thus smoothing the phase estimate. Regarding the calculation speed, the circular average-based method has the same complexity as one iteration of our ML-based methods, which is $O(K)$.

4) Maximum Likelihood Using Newton's Method and Coordinate Descent: Our ML-based methods using Newton's method (MLN) and using coordinate descent (MLC) are two different strategies to solve the same optimization problem: MLN breaks the error function into frames, while MLC breaks it into elements. The update scope of MLN seems to be more advanced than that of MLC as it modifies more elements at the same time. However, with the use of amplitude weights, the high-amplitude TF bins may restrict the change of the low-amplitude ones when they are updated simultaneously. An advantage of MLC is that it does not require tuning the parameters such as β in MLN. Regarding the calculation speed, although one iteration of both MLN and MLC has the complexity of $O(K)$, MLN is slower due to the eigenvalue estimation.

V. Inter-Frequency Phase Difference and Its Application to Phase Reconstruction

This section presents our new phase-based feature, the IFPD, and its application in improving two-stage phase reconstruction algorithms.

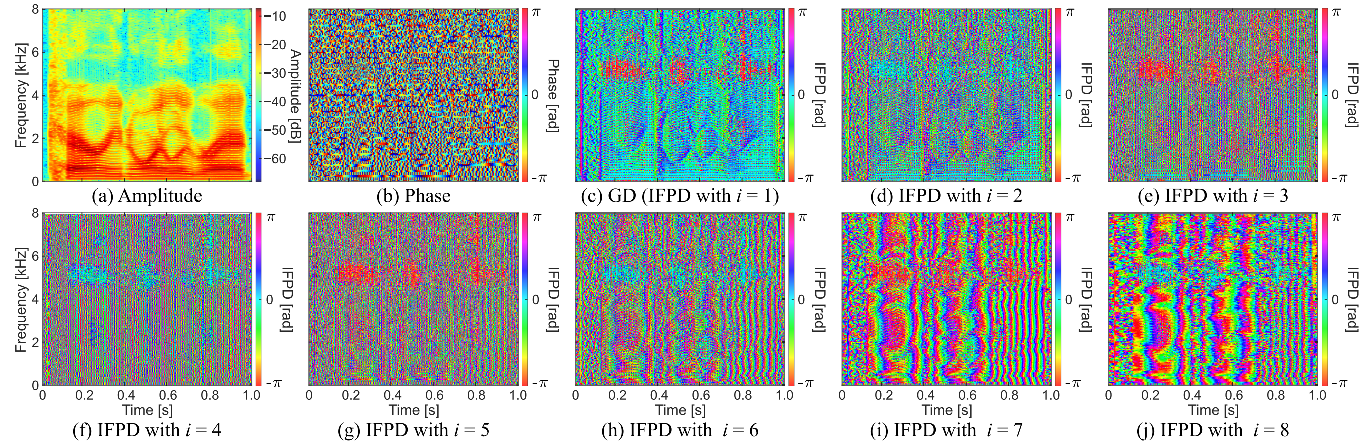


Fig. 7. Example of IFPD of speech signal with various frequency hops i. STFT is calculated using Hamming window with 32-ms length, 4-ms shift, and 512-point DFT. Sampling rate is 16 kHz. Linear phase shift is removed from phase using (16).

A. *Inter-Frequency Phase Difference*

In two-stage phase reconstruction algorithms, the phase relationships between TF bins along the frequency are maintained through the GD. However, as a phase difference between two consecutive TF bins, the GD only represents the local relationships. Meanwhile, all TF bins in the same frame are interdependent because each depends on all the underlying components of the signal, as discussed in Section III-A. In addition, we can see from Fig. 3 that the range of the area affected by a strong component is probably wider than two bins.

For the above reasons, to better represent the phase relationships along frequency, we generalize the calculation of the GD to the phase difference between two frequency bins with the frequency hop of i bins. We define a feature called the IFPD as

$$U_{k,\ell}^{(i)} = \mathcal{P}(\Phi_{k,\ell} - \Phi_{k+i,\ell}). \tag{30}$$

For $i = 1$, $U_{k,\ell}^{(i)}$ is identical to $U_{k,\ell}$. It is worth noting that the subtraction operation in the GD calculation is an estimation of the phase derivative, while for the IFPD, it is merely the phase difference between two frequency bins. For the frequency hops smaller than the main-lobe width of the window function, the IFPD has similar properties to the GD in that its value at the strong components is close to zero, as discussed in Section III-A. For larger frequency hops, the IFPD may capture the phase difference between two harmonic components if the hop is close to the multiple of the fundamental frequency. This property relates to two other phase-based features, i.e., relative phase shift [8] and phase distortion [9], which also reflect the phase relationships between harmonic components. However, the calculation of these features relies on the estimation of the fundamental frequency and harmonic model, while the IFPD is literally based on the DFT. An example of the IFPD of a speech signal with the frequency hop varying from 1 to 8 bins is shown in Fig. 7.

To illustrate how the IFPD captures the phase difference between harmonic components, we analyze the example of the IFPD in Fig. 7. Let us represent two harmonic components by sinusoids as

$$x_1(n) = A_1 \cos(2\pi k_1 n/N + \phi_1), \tag{31}$$

$$x_2(n) = A_2 \cos(2\pi k_2 n/N + \phi_2), \tag{32}$$

where A_1 and A_2 are the amplitudes, ϕ_1 and ϕ_2 are the initial phases, and k_1 and k_2 are the frequencies of the sinusoids. The phase difference between the two components at frame ℓ is

$$\Delta\varphi_\ell = \mathcal{P}\left(\frac{2\pi\Delta k}{N} R\ell + \Delta\varphi_0\right), \tag{33}$$

where $\Delta\varphi_0 = \phi_1 - \phi_2$ is the phase difference at the time origin, and $\Delta k = k_1 - k_2$ is the frequency difference. Since the signal in Fig. 7 has the fundamental frequency close to 4 bins, the IFPD with the frequency hops of 4 bins shows the phase difference between two consecutive harmonic components. With $\Delta k = -4$, $R = 64$, and $N = 512$, from (33), we have $\Delta\varphi_\ell = \mathcal{P}(-\ell\pi + \Delta\varphi_0)$. Along the time frame, as ℓ increases, $\Delta\varphi_\ell$ switches between $\Delta\varphi_0$ and $\Delta\varphi_0 + \pi$, resulting in the vertical stripes in the IFPD spectrogram shown in Fig. 7(f). Considering two more distant harmonic components with the frequency hop of 8 bins, corresponding to $\Delta k = -8$, the phase difference is nearly constant along the time frame as $\Delta\varphi_\ell = \mathcal{P}(-2\ell\pi + \Delta\varphi_0) = \Delta\varphi_0$. Although $\Delta\varphi_\ell$ is a constant, the IFPD spectrogram shown in Fig. 7(j) changes slowly along the time in accordance with the variations of the fundamental frequency. The IFPD spectrograms with other frequency hops in Fig. 7 also exhibit similar patterns, although not as clearly as those with frequency bin hops that are multiple of the fundamental frequency.

B. *IFPD for Two-Stage Phase Reconstruction*

The IFPD can be used to enhance the phase relationships along the frequency in two-stage phase reconstruction algorithms. In the first stage, the IFPD with various hops is reconstructed from the amplitude using DNNs, similar to the IF and GD. In the second stage, we penalize the IFPD errors in addition to the IF and GD errors in the loss function for reconstructing the phase.

Because the IFPD is also wrapped, which may aggravate the wrapping issue in the LS-based method, the ML-based method is used. With the IFPD, the error function (21) becomes

$$\mathcal{L}_{\text{ML\_IFPD}}(\phi_\ell) = -\sum_{k,\ell}\left(\sum_{i\in \boldsymbol{S}} W^{u^{(i)}}_{k,\ell}\cos(\tilde{U}^{(i)}_{k,\ell} - \hat{U}^{(i)}_{k,\ell}) + W^{v}_{k,\ell}\cos(\tilde{V}_{k,\ell} - \hat{V}_{k,\ell})\right), \tag{34}$$

where $\boldsymbol{S}$ is the set of the frequency hops used for calculating the IFPD, including $i = 1$ for the GD. The weight for $U^{(i)}_{k,\ell}$ is defined as

$$W^{u^{(i)}}_{k,\ell} = \alpha^{(i)}|X_{k,\ell}|, \tag{35}$$

where the scalar $\alpha^{(i)}$ is used to adjust the contribution of $U^{(i)}_{k,\ell}$ in the error function.

Because the use of the IFPD makes the Hessian matrix no longer tridiagonal, which increases the computational complexity of Newton's method, (34) is minimized using the coordinate-descent strategy. The solution of $\hat{\Phi}_{k,\ell}$ for minimizing $\mathcal{L}_{\text{ML\_IFPD}}$ is the same as (29), except that we include terms containing $\hat{U}^{(i)}_{k,\ell}$ to the calculation of $C_{k,\ell}$ and $S_{k,\ell}$. We found that errors of the IFPD estimated in the first stage, $\tilde{U}^{(i)}_{k,\ell}$, become higher as the frequency hop i increases. This means that the reconstructed phase will get these errors if it is completely fitted with those IFPD estimates. Therefore, we only minimize the error function (34) with the IFPD for the first several iterations. After that, (21) is used to smooth the phase estimates with only the IF and GD.

In summary, the algorithm for reconstructing the phase with the IFPD is the same as Algorithm 3 with two modifications, i.e., the IFPD is required as an input, and, in the inner for-loop of the first for-loop, $\hat{\Phi}_{k,\ell}$ is updated to minimize $\mathcal{L}_{\text{ML\_IFPD}}$ instead of $\mathcal{L}_{\text{ML}}$. As discussed above, the update $\hat{\Phi}_{k,\ell}$ in the last for-loop in Algorithm 3 is still for minimizing $\mathcal{L}_{\text{ML}}$.

VI. Experiments and Results

A. Experimental Setup

We conducted experiments to evaluate the performances of two-stage phase reconstruction algorithms. All such algorithms share the same IF and GD estimated in the first stage. The methods used for the second stage include the conventional LS-based method [4] (`LS`), conventional circular average-based method [5] (`AVG`), the proposed weighted LS-based method (`WLS`), and the proposed ML-based methods with 30 iterations using Newton's method (`MLN`; $N_1 = 10$ and $N_2 = 20$) and coordinate descent (`MLC`; $N_1 = 5$ and $N_2 = 25$). The proposed algorithm using the IFPD (`ML+IFPD`) was also evaluated with 30 iterations ($N_1 = 5$ and $N_2 = 25$). In addition, we included conventional non-two-stage phase reconstruction algorithms for comparison. These are the Griffin-Lim method [23] with 100 iterations (`GL`), the phase gradient heap integration method [31] (`PGHI`), and the iterative method using alternating direction method of multipliers [29] with 100 iterations (`ADMMGLA`).

The data used for training were from the training set of the TIMIT dataset [48]. The sampling rate is $16\,\text{kHz}$. The tests were performed on 300 samples (150 males and 150 females) randomly selected from the test set of the TIMIT dataset.

In the implementation, the STFT was calculated using a Hamming window with a 32-ms length, 4-ms shift, and 512-point DFT. To reconstruct the IF, GD, and IFPD in the first stage, we used fully connected feedforward DNNs with 4 hidden layers, each layer containing 1024 gated tanh units [49], and the last layer containing linear units. This DNN architecture is similar to those in [4], [5], [50]. In addition, the authors of [50] claimed in their work that, based on their experiments, there was no difference between the gated layers and LSTM (long short-term memory) layers in terms of the quality of the reconstructed speech. For these reasons, we decided to use this DNN architecture. It is worth noting that a separate DNN is used to estimate each of the IF, GD, and IFPD. The input of the DNN was joint vectors consisting of the log amplitude at the current and ± 2 frames and was normalized to zero mean and unit variance. The output of the DNN was one frame of the phase feature (IF, GD, or IFPD), which was also normalized using (9) for the IF and (16) for the GD and IFPD. These models were trained using the Adam optimizer for 400 epochs. The parameters of the methods in the second stage were determined by fine-tuning, which are described as follows. The power p in `WLS` was set to 10. The weight β in `MLN` was set to 2.4. For `ML+IFPD`, we used a set of six frequency hops of $\boldsymbol{S} = \{1, 2, 3, 4, 5, 6\}$ with the corresponding set of weights $\alpha^{(i)}$ of $\{1.0, 0.4, 0.3, 0.2, 0.1, 0.1\}$. The Linear Algebra Package (LAPACK) [51] was used for the tridiagonal system algorithm and eigenvalue estimation.

For the objective metrics, we measured the perceptual evaluation of speech quality (PESQ) [52] and short-time objective intelligibility (STOI) [53] of the reconstructed signals. The higher those scores, the higher the quality of the signal. We also calculated the consistency measure [28] as

$$C(\hat{\boldsymbol{X}}) = 10\log_{10}\frac{\|\hat{\boldsymbol{X}} - \text{STFT}(\text{ISTFT}(\hat{\boldsymbol{X}}))\|^2}{\|\hat{\boldsymbol{X}}\|^2}, \tag{36}$$

where $(\hat{\boldsymbol{X}})_{k,\ell} = |X_{k,\ell}|e^{j\hat{\Phi}_{k,\ell}}$. The consistency measure indicates how much the phase spectrogram is consistent with the amplitude spectrogram, which is expected to be low.

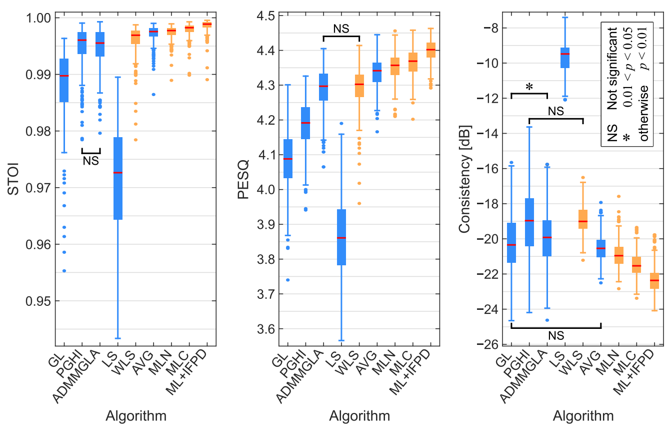
To further compare the subjective performances among the two-stage algorithms, we conducted the BS.1116 test [54] using webMUSHRA [55], which is a web-based listening test framework. In each BS.1116 trial, the subject is presented with three stimuli labeled `A`, `B`, and `C`. `A` is always the reference (original signal), while `B` and `C` are randomly assigned by the hidden reference and reconstructed signal. The subject is asked to assess the impairments (if any) on `B` and `C` compared to `A` using a continuous 5-grade scale with anchors defined as

- (5.0) Imperceptible,
- (4.0) Perceptible, but not annoying,
- (3.0) Slightly annoying,
- (2.0) Annoying,
- (1.0) Very annoying.

TABLE I
ERRORS OF DNNs IN FIRST STAGE

	IF	GD	IFPD				
			$i=2$	$i=3$	$i=4$	$i=5$	$i=6$
Training	0.133	0.231	0.290	0.432	0.530	0.729	0.717
Testing	0.148	0.245	0.307	0.450	0.575	0.804	0.811

Note: Error range is $[0, 2]$



(a) STOI (b) PESQ (c) Consistency

Fig. 8. Objective scores of phase reconstruction algorithms, where blue and red respectively indicate conventional and proposed methods.

Because one of B and C is identical to A, there must be at least one point of 5.0. As a general rule, if a subject rates the hidden reference with a score of less than 5.0 for more than 15% of the test, all the results of this subject will not be considered. The samples presented to each subject are randomly selected from the test set, in which the number of samples depends on the subject's demand (maximum 15 samples per subject, corresponding to 105 trials for 7 methods). The subjects participating in the test were all students ranging from 20 to 30 in age. Apart from the results of 3 subjects excluded by the test rules, 245 samples (which may be duplicated) were tested by 20 subjects in total.

B. Results

Table I lists the errors of the DNNs in the first stage, in which an error is defined as

$$\epsilon(\boldsymbol{Y}, \tilde{\boldsymbol{Y}}) = 1 - \frac{1}{KL}\sum_{k,\ell}\cos(Y_{k,\ell} - \tilde{Y}_{k,\ell}). \quad (37)$$

$\epsilon(\boldsymbol{Y}, \tilde{\boldsymbol{Y}})$ is similar to $\mathcal{L}_{\text{DNN}}$ in (8), however, (37) is for the whole spectrogram, while (8) is for each frame. The error range is $[0, 2]$. We can see from Table I that the higher the frequency hop i, the higher the errors of the DNNs for reconstructing the IFPD. The reason is most likely because, when the frequency hop is large, the connections between TF bins are weak due to the low side lobes of the window function. In such a case, the IFPD becomes less structured, hence, more difficult to estimate.

Fig. 8 shows the STOI, PESQ, and consistency measure of the reconstructed signals of the phase reconstruction algorithms. The results were analyzed with the paired sample t-test, which shows that the differences between the scores are statistically significant with a few exceptions. It can be seen from Fig. 8 that the two-stage methods, except the LS-based methods, performed better than the conventional non-two-stage methods, in which the ML+IFPD yielded the highest results. By using the IFPD in addition to the IF and GD for only several first iterations, ML+IFPD improved the results of MLC for all metrics. Although both MLN and MLC minimize the same error function with the same number of iterations, MLC achieved better results than MLN. A possible reason is that the update of MLN is an approximation while MLC directly solves the equation setting the derivative to zero. Although the solution in MLC is local, the separability of the error function motivates it. We can also see from Fig. 8 that, by adding the amplitude weights, WLS significantly improved the results of its baseline method LS. However, WLS was still worse than AVG and our ML-based methods. This is most likely due to the LS-based methods being affected by the wrapping issue.

Fig. 9 shows an example of the phase differences between the original and estimated phases, i.e., $\mathcal{P}(\boldsymbol{\Phi} - \hat{\boldsymbol{\Phi}})$. We may expect that the phase difference spectrogram has large regions of the same color, at which $\boldsymbol{\Phi}$ and $\hat{\boldsymbol{\Phi}}$ change at the same rates. In other words, the phase relationships in those regions are preserved, even if the absolute values of the phase are changed. In addition, we only focus on the high-amplitude regions since the phases of low-amplitude TF bins have little effect on the ISTFT. We can see from Fig. 9(b) and (e) that the same-color regions in the phase-difference spectrograms of GL and LS are small. At the boundaries of those regions, the phase relationships are distorted, resulting in impairments in the amplitude and modulations in the frequency of the reconstructed time-domain signals. As a consequence, the signals estimated using GL and LS often contains artifacts, such as reverberation and buzz. The same-color regions at high-amplitude TF bins became larger for other methods. Especially, by using the IFPD, ML+IFPD clearly improved the phase relationships between TF bins, illustrated with the smooth phase difference spectrogram in Fig. 10(j). This finding in this example is reflected in, and consistent with, the objective results.

Fig. 10 illustrates the results of the BS.1116 test for the two-stage algorithms, i.e., the measures of perceptual impairments on the reconstructed signals compared with the original signal. The subjective scores in Fig. 10 expose a similar trend to the objective scores in Fig. 8, in which ML+IFPD surpassed other methods. The differences between the scores are also confirmed with the small p-values of the paired sample t-test.

The above observations confirmed that ML+IFPD outperforms the other methods. In addition, for the second stage of two-stage phase reconstruction algorithms, the ML-based methods are better than the LS-based and circular average-based methods. The experimental results also indicate the efficacy of using amplitude weights in improving the conventional LS-based method.

Although achieving high objective and subjective scores, a limitation of the two-stage approaches is that the waveform may differ from the original signal as we focus on the phase relationship between the TF bins but not the absolute value of the phase. This is a common problem for phase reconstruction when only the amplitude information is available [42]. In other applications, when the noisy/mixed phases are available, they can be used as

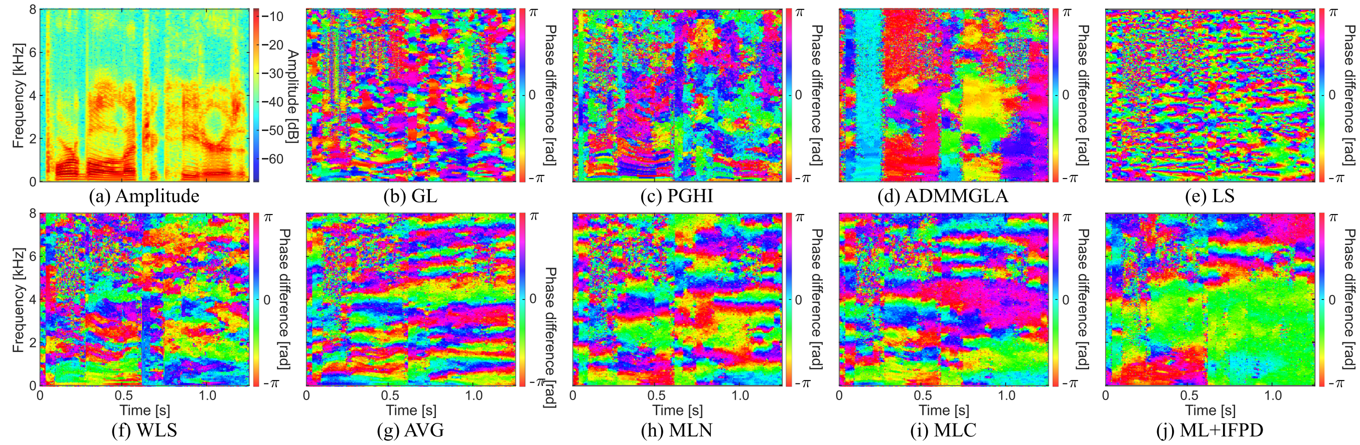


Fig. 9. Examples of (a) log-amplitude, and (b)–(j) phase differences between original phase $\mathbf{\Phi}$ and estimated phase $\hat{\mathbf{\Phi}}$.

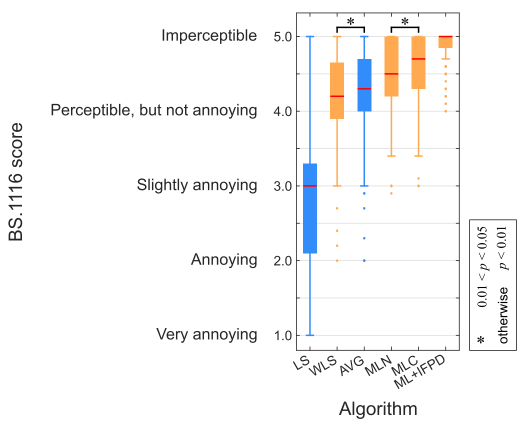


Fig. 10. Subjective scores of phase reconstruction algorithms.

an initialization for the proposed methods to reduce the problem. Another limitation of the proposed methods is that they require multiple models to estimate the phase features in the first stage, which may be a drawback in real-time applications. A possible solution is to use multitask learning, i.e., to combine all the models in the first stage into one model with multiple outputs.

VII. Conclusion

We presented two approaches for the second stage of two-stage phase reconstruction algorithms. The first method is to add the amplitude weights to a conventional LS-based method. The second method is based on the ML with the *von Mises* distribution, which is optimized using the regularized Newton's method and coordinate descent. In the theory discussion, we analyzed the GD properties and introduced a GD-normalization formula by compensating for the phase shift introduced by the commonly used zero-starting window function. We also investigated the roles of the phase relationships between TF bins in the ISTFT. On the basis of the analysis, we proposed a new phase-based feature, i.e., IFPD, and applied it to the phase reconstruction. Both objective and subjective experiments showed that the performance of our ML-based method using the IFPD is superior to other methods that use only the IF and GD. The results also suggest that ML-based methods perform better than other methods in the second stage, and the use of amplitude weights significantly improves the results of the conventional LS-based method. In the future, we will investigate effects of the first stage on the final results, including using other advanced DNN architectures and combining the models in the first stage into one model with multiple outputs. We will also apply the proposed methods to other fields of speech processing such as STFT-based speech synthesis and speech enhancement.

Appendix
Derivation of Weighted Least Square Solution (19)

We can rewrite the loss function (18) as

$$\mathcal{L}_{\mathrm{WLS}}(\boldsymbol{\phi}_\ell) = (\boldsymbol{\phi}_\ell - \boldsymbol{\phi}_{\ell-1} - \tilde{\boldsymbol{v}}_{\ell-1})^\mathsf{T} \breve{\boldsymbol{W}}^v_{\ell-1} (\boldsymbol{\phi}_\ell - \boldsymbol{\phi}_{\ell-1} - \tilde{\boldsymbol{v}}_{\ell-1}) + (\boldsymbol{D}\boldsymbol{\phi}_\ell - \tilde{\boldsymbol{u}}_\ell)^\mathsf{T} \breve{\boldsymbol{W}}^u_\ell (\boldsymbol{D}\boldsymbol{\phi}_\ell - \tilde{\boldsymbol{u}}_\ell). \tag{38}$$

Its first derivative can be calculated as

$$\begin{aligned}\frac{\mathcal{L}_{\mathrm{WLS}}}{\boldsymbol{\phi}_\ell} &= 2\breve{\boldsymbol{W}}^v_{\ell-1}(\boldsymbol{\phi}_\ell - \boldsymbol{\phi}_{\ell-1} - \tilde{\boldsymbol{v}}_{\ell-1}) + 2\boldsymbol{D}^\mathsf{T}\breve{\boldsymbol{W}}^u_\ell(\boldsymbol{D}\boldsymbol{\phi}_\ell - \tilde{\boldsymbol{u}}_\ell) \\ &= 2(\breve{\boldsymbol{W}}^v_{\ell-1} + \boldsymbol{D}^\mathsf{T}\breve{\boldsymbol{W}}^u_\ell\boldsymbol{D})\boldsymbol{\phi}_\ell \\ &\quad - 2(\breve{\boldsymbol{W}}^v_{\ell-1}(\boldsymbol{\phi}_{\ell-1} + \tilde{\boldsymbol{v}}_{\ell-1}) + \boldsymbol{D}^\mathsf{T}\breve{\boldsymbol{W}}^u_\ell\tilde{\boldsymbol{u}}_\ell).\end{aligned} \tag{39}$$

By setting the first derivative to zero, we achieve the solution (19).

References

[1] P. Mowlaee and J. Kulmer, "Phase estimation in single-channel speech enhancement: Limits-potential," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 23, no. 8, pp. 1283–1294, Aug. 2015.
[2] K. Vijayan and K. S. R. Murty, "Analysis of phase spectrum of speech signals using allpass modeling," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 23, no. 12, pp. 2371–2383, Dec. 2015.
[3] T. Gerkmann, M. Krawczyk-Becker, and J. Le Roux, "Phase processing for single-channel speech enhancement: History and recent advances," *IEEE Signal Process. Mag.*, vol. 32, no. 2, pp. 55–66, Mar. 2015.
[4] Y. Masuyama, K. Yatabe, Y. Koizumi, Y. Oikawa, and N. Harada, "Phase reconstruction based on recurrent phase unwrapping with deep neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2020, pp. 826–830.
[5] L. Thieling, D. Wilhelm, and P. Jax, "Recurrent phase reconstruction using estimated phase derivatives from deep neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2021, pp. 7088–7092.

[6] B. T. Nguyen, Y. Wakabayashi, K. Iwai, and T. Nishiura, "Two-stage phase reconstruction using DNN and von mises distribution-based maximum likelihood," in *Proc. IEEE Asia-Pacific Signal Inf. Process. Assoc. Annu. Summit Conf.,*2021, pp. 995–999.

[7] Y. Masuyama, K. Yatabe, K. Nagatomo, and Y. Oikawa, "Online phase reconstruction via DNN-Based phase differences estimation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 31, pp. 163–176, 2022.

[8] I. Saratxaga, I. Hernaez, D. Erro, E. Navas, and J. Sanchez, "Simple representation of signal phase for harmonic speech models," *Electron. Lett.*, vol. 45, no. 7, pp. 381–383, Mar. 2009.

[9] G. Degottex and D. Erro, "A uniform phase representation for the harmonic model in speech synthesis applications," *EURASIP J. Audio, Speech, Music Process.*, vol. 2014, no. 1, pp. 1–16, Oct. 2014.

[10] B. T. Nguyen, Y. Wakabayashi, K. Iwai, and T. Nishiura, "Analysis of derivative of instantaneous frequency and its application to voice activity detection," *Appl. Acoust.*, vol. 181, pp. 108–116, Oct. 2021.

[11] M.-V. Laitinen, S. Disch, and V. Pulkki, "Sensitivity of human hearing to changes in phase spectrum," *J. Audio Eng. Soc.*, vol. 61, no. 11, pp. 860–877, Nov. 2013.

[12] K. K. Paliwal and L. Alsteris, "Usefulness of phase spectrum in human speech perception," in *Proc. 8th Eur. Conf. Speech Commun. Technol.*, 2003, pp. 2117–2120.

[13] J. Muth, S. Uhlich, N. Perraudin, T. Kemp, F. Cardinaux, and Y. Mitsufuji, "Improving DNN-Based music source separation using phase features," in *Joint Workshop Mach. Learn. Music*, 2018. [Online]. Available: https://sites.google.com/site/faimmusic2018/program

[14] Y. Luo and N. Mesgarani, "Conv-TasNet: Surpassing ideal time–frequency magnitude masking for speech separation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 27, no. 8, pp. 1256–1266, Aug. 2019.

[15] Z.-Q. Wang, P. Wang, and D. Wang, "Complex spectral mapping for single- and multi-channel speech enhancement and robust ASR," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 28, pp. 1778–1787, 2020.

[16] D. Ma, N. Hou, V. T. Pham, H. Xu, and E. S. Chng, "Multitask-based joint learning approach to robust ASR for radio communication speech," in *Proc. IEEE Asia-Pacific Signal Inf. Process. Assoc. Annu. Summit Conf.*, 2021, pp. 497–502.

[17] N. Takahashi, P. Agrawal, N. Goswami, and Y. Mitsufuji, "Phasenet: Discretized phase modeling with deep neural networks for audio source separation," in *Proc. Conf. Int. Speech Commun. Assoc.,*2018, pp. 2713–2717.

[18] J. Le Roux, G. Wichern, S. Watanabe, A. Sarroff, and J. R. Hershey, "Phasebook and friends: Leveraging discrete representations for source separation," *IEEE J. Sel. Topics Signal Process.*, vol. 13, no. 2, pp. 370–382, May 2019.

[19] P. Magron, R. Badeau, and B. David, "Model-based STFT phase recovery for audio source separation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 26, no. 6, pp. 1095–1105, Jun. 2018.

[20] Y. Wakabayashi, T. Fukumori, M. Nakayama, T. Nishiura, and Y. Yamashita, "Single-channel speech enhancement with phase reconstruction based on phase distortion averaging," *IEEE/ACM Trans. Audio, Speech Lang. Process.*, vol. 26, no. 9, pp. 1559–1569, Sep. 2018.

[21] P. Mowlaee and J. Kulmer, "Harmonic phase estimation in single-channel speech enhancement using phase decomposition and SNR information," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 23, no. 9, pp. 1521–1532, Sep. 2015.

[22] Y. Wakabayashi, T. Fukumori, M. Nakayama, T. Nishiura, and Y. Yamashita, "Phase reconstruction method based on time-frequency domain harmonic structure for speech enhancement," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2017, pp. 5560–5564.

[23] D. Griffin and J. Lim, "Signal estimation from modified short-time fourier transform," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 32, no. 2, pp. 236–243, Apr. 1984.

[24] S. Takaki, H. Kameoka, and J. Yamagishi, "Direct modeling of frequency spectra and waveform generation based on phase recovery for DNN-based speech synthesis," in *Proc. Conf. Int. Speech Commun. Assoc.,*2017, pp. 1128–1132.

[25] T. Kaneko, S. Takaki, H. Kameoka, and J. Yamagishi, "Generative adversarial network-based postfilter for STFT spectrograms," in *Proc. Conf. Int. Speech Commun. Assoc.,*Aug. 2017, pp. 3389–3393.

[26] Y. Saito, S. Takamichi, and H. Saruwatari, "Text-to-speech synthesis using STFT spectra based on low-/multi-resolution generative adversarial networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2018, pp. 5299–5303.

[27] P. Magron, R. Badeau, and B. David, "Phase reconstruction of spectrograms with linear unwrapping: Application to audio signal restoration," in *Proc. IEEE 23rd Eur. Signal Process. Conf.*, 2015, pp. 1–5.

[28] J. Le Roux, H. Kameoka, N. Ono, and S. Sagayama, "Fast signal reconstruction from magnitude STFT spectrogram based on spectrogram consistency," in *Proc. Int. Conf. Digit. Audio Effects*, vol. 10, 2010, pp. 397–403.

[29] Y. Masuyama, K. Yatabe, and Y. Oikawa, "Griffin–Lim like phase recovery via alternating direction method of multipliers," *IEEE Signal Process. Lett.*, vol. 26, no. 1, pp. 184–188, Jan. 2019.

[30] T. Peer, S. Welker, and T. Gerkmann, "Beyond griffin-lim: Improved iterative phase retrieval for speech," in *Proc. Int. Workshop Acoust. Signal Enhancement*, 2022, pp. 1–5.

[31] Z. Průša, P. Balazs, and P. L. Søndergaard, "A noniterative method for reconstruction of phase from STFT magnitude," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 25, no. 5, pp. 1154–1164, May 2017.

[32] B. Boashash, "Estimating and interpreting the instantaneous frequency of a signal. I. fundamentals," *Proc. IEEE*, vol. 80, no. 4, pp. 520–538, Apr. 1992.

[33] R. M. Hegde, H. A. Murthy, and V. R. R. Gadde, "Significance of the modified group delay feature in speech recognition," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 15, no. 1, pp. 190–202, Jan. 2007.

[34] N. Zheng and X.-L. Zhang, "Phase-aware speech enhancement based on deep neural networks," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 27, no. 1, pp. 63–76, Sep. 2018.

[35] A. A. Nugraha, K. Sekiguchi, and K. Yoshii, "A deep generative model of speech complex spectrograms," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2019, pp. 905–909.

[36] S. Takamichi, Y. Saito, N. Takamune, D. Kitamura, and H. Saruwatari, "Phase reconstruction from amplitude spectrograms based on von-mises-distribution deep neural network," in *Proc. IEEE Int. Workshop Acoust. Signal Enhancement*, 2018, pp. 286–290.

[37] Y. Masuyama, K. Yatabe, Y. Koizumi, Y. Oikawa, and N. Harada, "Deep Griffin–Lim iteration: Trainable iterative phase reconstruction using neural network," *IEEE J. Sel. Topics Signal Process.*, vol. 15, no. 1, pp. 37–50, Jan. 2021.

[38] T. Gerkmann, "Bayesian estimation of clean speech spectral coefficients given a priori knowledge of the phase," *IEEE Trans. Signal Process.*, vol. 62, no. 16, pp. 4199–4208, Aug. 2014.

[39] T. Gerkmann, "MMSE-optimal enhancement of complex speech coefficients with uncertain prior knowledge of the clean speech phase," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2014, pp. 4478–4482.

[40] K. Tan and D. Wang, "Learning complex spectral mapping with gated convolutional recurrent networks for monaural speech enhancement," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 28, pp. 380–390, 2019.

[41] Z. Ouyang, H. Yu, W.-P. Zhu, and B. Champagne, "A fully convolutional neural network for complex spectrogram processing in speech enhancement," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2019, pp. 5756–5760.

[42] N. Sturmel et al., "Signal reconstruction from STFT magnitude: A state of the art," in *Proc. Int. Conf. Digit. Audio Effects*, 2011, pp. 375–386.

[43] D. C. Ghiglia and M. D. Pritt, *Two-Dimensional Phase Unwrapping: Theory, Algorithms, and Software*. Hoboken, NJ, USA: Wiley, 1998.

[44] B. N. Datta, *Numerical Linear Algebra and Applications*, vol. 116. Philadelphia, PA, USA: SIAM, 2010.

[45] D. W. Marquardt, "An algorithm for least-squares estimation of nonlinear parameters," *J. Soc. Ind. Appl. Math.*, vol. 11, no. 2, pp. 431–441, Jun. 1963.

[46] W. Kahan, "Accurate eigenvalues of a symmetric tri-diagonal matrix," Computer Science Dept., Stanford University, Tech. Rep., Jul. 1966. [Online]. Available: https://apps.dtic.mil/sti/citations/AD0638796

[47] S. J. Wright, "Coordinate descent algorithms," *Math. Program.*, vol. 151, no. 1, pp. 3–34, Mar. 2015.

[48] J. S. Garofolo, "TIMIT Acoustic Phonetic Continuous Speech Corpus," *Linguistic Data Consortium*, 1993. [Online]. Available: https://catalog.ldc.upenn.edu/LDC93s1

[49] A. V. D. Oord et al., "Wavenet: A generative model for raw audio," in *Proc. 9th ISCA Workshop Speech Synthesis Workshop*, vol. 125, 2016. [Online]. Available: https://www.isca-speech.org/archive/ssw_2016/vandenoord16_ssw.html

[50] S. Takamichi, Y. Saito, N. Takamune, D. Kitamura, and H. Saruwatari, “Phase reconstruction from amplitude spectrograms based on directional-statistics deep neural networks,” *Signal Process.*, vol. 169, Apr. 2020, Art. no. 107368.

[51] E. Anderson et al., *LAPACK Users’ Guide*. Philadelphia, PA, USA: SIAM, 1999.

[52] A. W. Rix, J. G. Beerends, M. P. Hollier, and A. P. Hekstra, “Perceptual evaluation of speech quality (PESQ)–a new method for speech quality assessment of telephone networks and codecs,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2001, pp. 749–752.

[53] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen, “An algorithm for intelligibility prediction of time–frequency weighted noisy speech,” *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 19, no. 7, pp. 2125–2136, Sep. 2011.

[54] Recommendation ITU-R BS.*1116-3, Methods for the Subjective Assessment of Small Impairments in Audio Systems*, Feb. 2015. [Online]. Available: https://www.itu.int/rec/R-REC-BS.1116

[55] M. Schoeffler et al., “webMUSHRA–A comprehensive framework for web-based listening tests,” *J. Open Res. Softw.*, vol. 6, no. 1, Feb. 2018. [Online]. Available: https://openresearchsoftware.metajnl.com/articles/10.5334/jors.187

Nguyen Binh Thien received the B.E. degree from the Danang University of Science and Technology, Danang, Vietnam, in 2018, and M.E. degree in 2020 from Ritsumeikan University, Shiga, Japan, where he is currently working toward the Ph.D. degree with the Graduate School of Information Science and Engineering. His research interests include acoustic signal processing, and speech phase processing. He is a Member of the Acoustical Society of Japan.

Yukoh Wakabayashi (Member, IEEE) received the B.E. and M.E. degrees from Osaka University, Osaka, Japan, in 2008 and 2010, respectively, and the Ph.D. degree from Ritsumeikan University, Shiga, Japan, in 2017. He joined Rohm, Inc., Kyoto, Japan, in 2010. From 2012 to 2014, he was an Assistant Researcher with Kyoto University. From 2016 to 2017, he was a recipient of the JSPS Research Fellowship for Young Scientists DC2. From 2018 to 2020, he was an affiliate Assistant Professor with Ritsumeikan University. He is currently an Assistant Professor with the Department of Computer Science and Engineering, Toyohashi University of Technology, Aichi, Japan, and the Faculty of Systems Design, Tokyo Metropolitan University, Tokyo, Japan. His research interests include acoustic signal processing, speech phase processing, array signal processing, and speaker diarization. He is a Member of the Institute of Electrical and Electronics Engineers, the Institute of Electronics, Information and Communication Engineers, and Acoustical Society of Japan.

Kenta Iwai (Member, IEEE) received the B.Eng., M.Eng., and Ph.D. degrees from Kansai University, Suita, Japan, in 2010, 2012, and 2015, respectively. He was a part-time Researcher in 2015 and post-doctoral Fellow from 2016 to 2018 with Kansai University. From 2018 to 2020, he was a specially appointed Assistant Professor with Ritsumeikan University, Kusatsu, Japan. From 2021 to 2022, he was an Assistant Professor with Ritsumeikan University. Since 2023, he has been a Lecturer with Ritsumeikan University. His research interests include acoustic signal processing, in particular, active noise control, acoustic echo cancellation, and nonlinear distortion reduction for electro-acoustic transducers. He is a Member of the Institute of Electronics, Information and Communication Engineers, and the Acoustical Society of Japan.

Takanobu Nishiura (Member, IEEE) was born in 1974. He received the B.E. degree from the Nara National College of Technology, Yamatokoriyama, Japan, in 1997, and the M.E. and Ph.D. degrees from Nara Institute of Science and Technology, Ikoma, Japan, in 1999 and 2001, respectively. From 2001 to 2004, he was a Research Associate with Wakayama University, Wakayama, Japan. From 2004 to 2014, he was an Associate Professor with Ritsumeikan University. He is currently a Professor with Ritsumeikan University, Kusatsu, Japan. His research interests include parametric loudspeaker, optical laser microphone, and auditory scene analysis. He is a Member of the Institute of Electronics, Information and Communication Engineers, and Acoustical Society of Japan.