Two-stage phase reconstruction using DNN and von Mises distribution-based maximum likelihood

NGUYEN Binh Thien*, Yukoh WAKABAYASHI†, Kenta IWAI*, Takanobu NISHIURA*
* Ritsumeikan University, Shiga, Japan
E-mail: {gr0398xe@ed, iwai18sp@fc, nishiura@is}.ritsumei.ac.jp
† Tokyo Metropolitan University, Tokyo, Japan
E-mail: wakayuko@tmu.ac.jp

***Abstract*—This paper presents an improvement to a two-stage algorithm for estimating the phase from only the amplitude with deep neural networks (DNNs). Rather than directly estimating the phase, the two-stage method estimates phase derivatives, i.e., instantaneous frequency (IF) and group delay (GD), by using DNNs in the first stage, and it then reconstructs the phase from those derivatives using a least-squares (LS) method in the second stage. A problem with the algorithm is that the periodicity of the estimated IF and GD significantly affects the results of LS estimation. In this paper, we replace the LS method in the second stage with a new maximum-likelihood method using *von Mises* distribution. The error function is minimized by using a regularized Newton's method. Experimental results demonstrate that the proposed method can reduce the IF and GD errors of the reconstructed phase in the second stage and achieve a higher overall speech quality than conventional methods.**

I. Introduction

In the last decade, phase reconstruction for a given amplitude has gained popularity as recent studies [1], [2] have demonstrated its importance. It has been shown that phases reconstructed from the amplitude and observed noisy/mixed phases help to produce higher-quality time-domain signals in speech enhancement [2]–[4] and source separation [5]–[7]. In other applications such as speech synthesis [8]–[10], in which observed phases are unavailable, phase reconstruction relies only on the information from the amplitude. Various algorithms have been proposed. The Griffin-Lim (GL) algorithm [11] and its modifications [12], [13] are well-known iterative-based methods using the consistency property of the short-time Fourier transform (STFT). Recent studies [14]–[16] use deep neural networks (DNNs) to benefit from the prior knowledge of a signal.

Instead of estimating the phase directly from the amplitude, [17] proposed a two-stage phase reconstruction algorithm utilizing the time and frequency derivatives of the phase, i.e., instantaneous frequency (IF) [18] and group delay (GD) [19]. Fig. 1 shows a block diagram of this method. In the first stage, the IF and GD are reconstructed from the amplitude by using DNNs. By differentiating, these phase derivatives become more structured and less sensitive; therefore, they can be reconstructed much more easily than the phase itself. In the second stage, the phase is estimated from the IF and GD by using a least-squares (LS) method. Experimental results have demonstrated that this two-stage strategy is more efficient than directly reconstructing the phase. However, the periodicity of the estimated IF and GD may hurt the performance of the LS-based algorithm. For instance, the values of $-\pi$ and π are identical for the IF and GD, but they make a big distance for the LS error. Although [17] proposed a GD-modification scheme for dealing with this problem, this scheme seems to be less effective when the errors of the estimated IF and GD are high. Starting with the same first stage, a more recent study [20] proposed an alternative simple algorithm for the second stage in which the phase estimate at each time-frequency (TF) bin is the weighted average of its estimates calculated from the previously estimated phase elements in its vicinity.

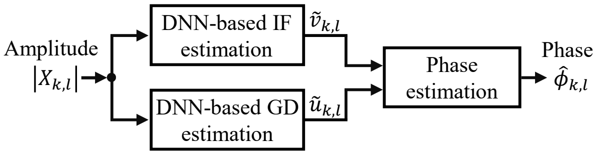


Fig. 1. Block diagram of two-stage phase reconstruction algorithm.

In this paper, adopting the same approach as in [17], we tackle the problem of the LS method described above by replacing the quadratic error function with a cosine error function. In other words, the LS problem becomes a maximum-likelihood (ML) problem with *von Mises* distribution. We also utilize amplitudes as weights to emphasize the importance of phases at high-amplitude positions. The error function is minimized by using a regularized Newton's method.

The remainder of the paper is organized as follows. In Section II, notation, formulation, and related works are described. In Section III, we present the proposed phase-reconstruction method. Section IV outlines the experiments and presents the results. Finally, Section V concludes the paper.

II. Notation, formulation, and related works

In this section, we first define the notation and formulation. Then, we present brief descriptions of the conventional two-stage phase reconstruction algorithms.

A. *Notation and formulation*

Let $X_{k,l}$ be the STFT of a discrete-time speech signal, where $l = 0, \ldots, L-1$ and $k = 0, \ldots, K-1$ are the time frame index and frequency bin index, respectively. Then, its phase and amplitude spectra are denoted as $\phi_{k,l} = \angle[X_{k,l}]$ and $|X_{k,l}|$, respectively, where $\angle$ is an angle operator.

IF is defined as the time-derivative of a phase, which can be estimated by

$$v_{k,l} = \text{princ}\{\phi_{k,l+1} - \phi_{k,l}\}, \tag{1}$$

where princ$\{\cdot\}$ is a function mapping the phase difference into the range of $(-\pi, \pi]$. Similarly, the GD, which is a negative frequency-derivative of a phase, can be calculated as

$$u_{k,l} = \text{princ}\{\phi_{k,l} - \phi_{k+1,l}\}. \tag{2}$$

Let $\boldsymbol{\phi}_l = (\phi_{0,l}, \cdots, \phi_{K-1,l})^\mathsf{T}$ be the phase spectrum at frame l, where $(\cdot)^\mathsf{T}$ is a matrix transposition operator; the vector-based formulas of IF and GD at frame l are given as

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

and

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

respectively, where $\boldsymbol{D}$ is a $(K-1) \times K$ matrix defined by

$$D_{i,j} = \begin{cases} 1, & \text{if } i = j \\ -1, & \text{if } i+1 = j \ . \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

Note that the princ$\{\cdot\}$ function is omitted in (3) and (4) (and from now on) for the sake of convenience.

The main objective of the two-stage phase reconstruction algorithms is to estimate the IF $\boldsymbol{v}$ and GD $\boldsymbol{u}$ from a given amplitude $|\boldsymbol{X}|$ and then reconstruct the phase $\boldsymbol{\phi}$ that preserves as much IF and GD information as possible.

B. IF and GD estimation from amplitude using DNNs

In the first stage of the two-stage phase reconstruction algorithm [17], the IF and GD are modeled by fully connected DNNs. The DNNs are trained to reconstruct the IF and GD for each frame l by minimizing the following loss function:

$$\mathcal{L}_{\text{DNN}}(\boldsymbol{y}_l, \tilde{\boldsymbol{y}}_l) = -\sum_{k=0}^{K-1} \cos(y_{k,l} - \tilde{y}_{k,l}), \tag{6}$$

where $\boldsymbol{y}_l$ and $\tilde{\boldsymbol{y}}_l$ are the original and estimated values of the output, which is either the IF or GD. The inputs are vectors consisting of the log amplitude at the current and ±2 frames.

C. Estimation of phase from its derivatives

Regarding the second stage, in which the phase is reconstructed from the estimated IF and GD, two methods, the baseline LS and weighted-average, are reviewed below.

1) Baseline LS method: The authors of [17] proposed recursively reconstructing each phase spectrum by minimizing the following frame-wise quadratic error function:

$$\mathcal{L}_{\text{LS}}(\boldsymbol{\phi}_l) = \|\boldsymbol{\phi}_l - \boldsymbol{\phi}_{l-1} - \boldsymbol{v}_{l-1}\|_2^2 + \|\boldsymbol{D}\boldsymbol{\phi}_l - \boldsymbol{u}_l\|_2^2, \tag{7}$$

where $\|\cdot\|_2$ is a Euclidean norm, $\boldsymbol{\phi}_{l-1}$ is the previously estimated phase spectrum at frame $l-1$, and $\boldsymbol{v}_l$ and $\boldsymbol{u}_l$ are replaced by their estimates $\tilde{\boldsymbol{v}}_l$ and $\tilde{\boldsymbol{u}}_l$ in the first stage, respectively. The solution to the LS problem in (7) is

$$\hat{\boldsymbol{\phi}}_l = (\boldsymbol{I}_K + \boldsymbol{D}^\mathsf{T}\boldsymbol{D})^{-1}(\hat{\boldsymbol{\phi}}_{l-1} + \tilde{\boldsymbol{v}}_{l-1} + \boldsymbol{D}^\mathsf{T}\tilde{\boldsymbol{u}}_l), \tag{8}$$

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

From (6), it is clear that the estimated IF and GD are also periodic. This 2π ambiguity significantly affects the LS solution. To solve this problem, [17] proposed a GD-modification scheme fixing the IF and modifying the GD so that they are consistent with each other.

2) Weighted-average method: Utilizing amplitude information, [20] proposed a simple method for calculating each phase element as a weighted average of its estimates as

$$\hat{\phi}_{k,l} = \angle \sum_{p=1}^{P} \alpha_{k,l}^{(p)} \cdot e^{j\varphi_{k,l}^{(p)}}, \tag{9}$$

where $\varphi_{k,l}^{(p)}$ is an estimate of $\phi_{k,l}$ calculated by using the IF, GD, and the pth previously estimated phase element near the TF bin (k,l). $\alpha_{k,l}^{(p)}$ denotes the weight, and P is the number of the neighbors involved. [20] empirically determined that $P = 3$ yields the best result, corresponding to the TF bins of $(k-1,l)$, $(k,l-1)$, and $(k+1,l-1)$.

III. Proposed estimation of phase from its derivatives

We start with the same first stage as in [17] for reconstructing the IF and GD from a given amplitude. To overcome the problem of the LS method when estimating the phase from the IF and GD in the second stage, taking the idea of [14], we propose an ML estimation method using *von Mises* distribution, whose probability density function is given by

$$f(x|\mu,\kappa) = \frac{e^{\kappa\cos(x-\mu)}}{2\pi I_0(\kappa)}, \tag{10}$$

where μ is a measure of location, κ is a measure of concentration, and $I_0(\kappa)$ is a modified Bessel function of order 0. Because the contribution of each TF bin of the STFT to the reconstructed time-domain signal highly depends on the amplitude of that bin, we use the amplitude as weights to emphasize the importance of the phase at high-amplitude positions. By taking the negative logarithm of (10), we define the loss function as

$$\begin{aligned} \mathcal{L}_{\text{ML}}(\boldsymbol{\phi}_l) = -\sum_{k=0}^{K-1} \Big(&|X_{k,l}|\cos(u_{k,l} - \hat{u}_{k,l}) \\ &+ |X_{k,l-1}|\cos(v_{k,l-1} - \hat{v}_{k,l-1}) \\ &+ |X_{k,l}|\cos(v_{k,l} - \hat{v}_{k,l})\Big). \end{aligned} \tag{11}$$

Similar to (7), $u_{k,l}$ and $v_{k,l}$ in (11) are also replaced by their estimates $\tilde{u}_{k,l}$ and $\tilde{v}_{k,l}$ from the first stage, and the phase at frames other than l are assumed to be constant. A novel point of (11) is that we also consider the contribution of the IF at the current frame $v_{k,l}$ (in other words, the phase at the next frame $\phi_{k,l+1}$) in addition to that at the previous frame $v_{k,l-1}$. This point makes the relationship between consecutive phase frames more solid. With several simple mathematics transformations, the gradient vector $\nabla_{\boldsymbol{\phi}_l}\mathcal{L}_{\text{ML}}(\boldsymbol{\phi}_l)$ can be calculated with the kth element defined as

$$\frac{\partial \mathcal{L}_{\text{ML}}(\boldsymbol{\phi}_l)}{\partial \phi_{k,l}} = \sin(\phi_{k,l})C_{k,l} - \cos(\phi_{k,l})S_{k,l}, \tag{12}$$

where

$$C_{k,l} = |X_{k,l}|\cos(\hat{\phi}_{k,l+1} - \tilde{v}_{k,l}) + |X_{k,l-1}|\cos(\hat{\phi}_{k,l-1} + \tilde{v}_{k,l-1}) + |X_{k,l}|\cos(\phi_{k+1,l} + \tilde{u}_{k,l}) + |X_{k-1,l}|\cos(\phi_{k-1,l} - \tilde{u}_{k-1,l}), \tag{13}$$

and $S_{k,l}$ is defined the same as $C_{k,l}$, except that all the cosine functions are replaced by sine functions. Note that, at the boundaries of $k = 0$, $k = K - 1$, $l = 0$, and $l = L - 1$, we remove the terms containing the indices of $k - 1$, $k + 1$, $l - 1$, and $l + 1$, respectively, from $C_{k,l}$ and $S_{k,l}$. We can see from (12) that the partial derivative of $\mathcal{L}_{\mathrm{ML}}(\boldsymbol{\phi}_l)$ with respect to $\phi_{k,l}$ contains only two phase elements of the same frame, i.e., $\phi_{k+1,l}$ and $\phi_{k-1,l}$. Consequently, the Hessian matrix of $\mathcal{L}_{\mathrm{ML}}(\boldsymbol{\phi}_l)$ is a symmetric tridiagonal matrix whose element on the main diagonal is

$$\frac{\partial^2 \mathcal{L}_{\mathrm{ML}}(\boldsymbol{\phi}_l)}{\partial \phi_{k,l}^2} = \cos(\phi_{k,l}) C_{k,l} + \sin(\phi_{k,l}) S_{k,l}, \tag{14}$$

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

$$\frac{\partial^2 \mathcal{L}_{\mathrm{ML}}(\boldsymbol{\phi}_l)}{\partial \phi_{k,l} \partial \phi_{k+1,l}} = -|X_{k,l}|\cos(\phi_{k,l} - \phi_{k+1,l} - \tilde{u}_{k,l}). \tag{15}$$

This fact motivates us to use Newton's method to iteratively update the phase estimate as

$$\hat{\boldsymbol{\phi}}_l^{(i)} = \hat{\boldsymbol{\phi}}_l^{(i-1)} - \boldsymbol{H}^{-1} \nabla_{\boldsymbol{\phi}_l} \mathcal{L}_{\mathrm{ML}}(\hat{\boldsymbol{\phi}}_l^{(i-1)}), \tag{16}$$

where $\hat{\boldsymbol{\phi}}_l^{(i)}$ is the estimate of $\boldsymbol{\phi}_l$ at the i^{th} update, and $\boldsymbol{H}$ is the Hessian matrix of $\mathcal{L}_{\mathrm{ML}}(\boldsymbol{\phi}_l)$ measured at $\hat{\boldsymbol{\phi}}_l^{(i-1)}$. Since $\boldsymbol{H}$ is tridiagonal, the vector $\boldsymbol{H}^{-1} \nabla_{\boldsymbol{\phi}_l} \mathcal{L}_{\mathrm{ML}}(\boldsymbol{\phi}_l)$ can be calculated with the complexity of $O(n)$ [21] (instead of the $O(n^3)$ required by Gaussian elimination for a non-tridiagonal matrix $\boldsymbol{H}$). A problem with (16) is that $\boldsymbol{H}$ is often not positive definite as $\mathcal{L}_{\mathrm{ML}}(\boldsymbol{\phi}_l)$ is periodic. To deal with this, we apply a regularization strategy as in [22]. The update becomes

$$\hat{\boldsymbol{\phi}}_l^{(i)} = \hat{\boldsymbol{\phi}}_l^{(i-1)} - (\boldsymbol{H} + \gamma \boldsymbol{I}_K)^{-1} \nabla_{\boldsymbol{\phi}_l} \mathcal{L}_{\mathrm{ML}}(\hat{\boldsymbol{\phi}}_l^{(i-1)}), \tag{17}$$

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 (17) approximates the standard gradient descent at a rate of $1/\gamma$. Ideally, γ is adaptive to each update so that it is large enough to offset the negative eigenvalues of $\boldsymbol{H}$. Because $\boldsymbol{H}$ is symmetric tridiagonal, we can efficiently estimate only its smallest eigenvalue as in [23]. We propose calculating $\gamma^{(i)}$ by using the estimated smallest eigenvalue $\lambda^{(i)}$ of the matrix $\boldsymbol{H}$ at the i^{th} update as

$$\gamma^{(i)} = \begin{cases} -\beta \lambda^{(i)}, & \text{if } \lambda^{(i)} < 0 \\ 0, & \text{otherwise} \end{cases}, \tag{18}$$

where β is a scaling constant.

Before minimizing (11), we propose recursively calculating the phase for each frame in a similar way to the LS method in [17] by minimizing the following error function:

$$\mathcal{L}_{\mathrm{RML}}(\boldsymbol{\phi}_l) = -\sum_{k=0}^{K-1} \Big(|X_{k,l}|\cos(u_{k,l} - \hat{u}_{k,l}) + |X_{k,l-1}|\cos(v_{k,l-1} - \hat{v}_{k,l-1}) \Big). \tag{19}$$

Algorithm 1 Proposed phase reconstruction from IF and GD

```
Input: Amplitude spectrogram $|\boldsymbol{X}|$, estimated IF $\tilde{\boldsymbol{v}}$ and GD $\tilde{\boldsymbol{u}}$, number of loops $N_1$ and $N_2$ for optimization
Output: Phase spectrogram $\hat{\boldsymbol{\phi}}$
  Calculate $\hat{\boldsymbol{\phi}}_0$ as in (20)
  for $l \in \{1, \dots, L-1\}$ do
    $\hat{\boldsymbol{\phi}}_l \leftarrow \hat{\boldsymbol{\phi}}_{l-1} + \tilde{\boldsymbol{v}}_{l-1}$
    for $i \in \{1, \dots, N_1\}$ do
      Update $\hat{\boldsymbol{\phi}}_l$ as in (17) replacing $\mathcal{L}_{\mathrm{ML}}(\boldsymbol{\phi}_l)$ by $\mathcal{L}_{\mathrm{RML}}(\boldsymbol{\phi}_l)$
    end for
  end for
  for $i \in \{1, \dots, N_2\}$ do
    for $l \in \{0, \dots, L-1\}$ do
      Update $\hat{\boldsymbol{\phi}}_l$ as in (17)
    end for
  end for
```

The error function (19) is equivalent to (11) removing the terms containing $v_{k,l}$. In other words, (19) doesn't require $\hat{\phi}_{k,l+1}$, which is not available at the beginning. Let $\hat{\phi}_{0,0} = 0$; the initial phase for the first frame can be simply calculated from the estimated GD as

$$\hat{\phi}_{k,0} = \hat{\phi}_{k-1,0} - \tilde{u}_{k-1,0}. \tag{20}$$

The minimization of (19) will start from the second frame, in which the initial phase is calculated from the estimated phase at the previous frame and IF as

$$\hat{\boldsymbol{\phi}}_l = \hat{\boldsymbol{\phi}}_{l-1} + \tilde{\boldsymbol{v}}_{l-1}. \tag{21}$$

After minimizing (19) with N_1 loops, the phase estimate is then updated by minimizing the full version of the error function in (11) with N_2 loops.

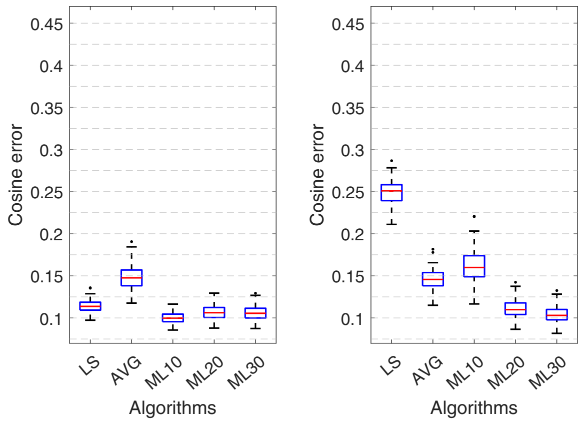
The pseudo-code for the proposed algorithm is given in Algorithm 1. Although using iteration, this method is much faster than the GL method as its calculation for each loop is simple thanks to the symmetry and tridiagonality of the Hessian matrix.

IV. Experiments and results

A. Experimental setup

We conducted experiments to compare two-stage phase reconstruction algorithms, including the baseline LS method (`LS`) [17], the weighted average method (`AVG`) [20], and the proposed ML method with 10 loops (`ML10`, $N_1 = 5$ and $N_2 = 5$), 20 loops (`ML20`, $N_1 = 10$ and $N_2 = 10$), and 30 loops (`ML30`, $N_1 = 10$ and $N_2 = 20$). All of these algorithms shared the same IF and GD estimated from the amplitude by using DNNs in the first stage. We also included the Griffin-Lim method with 100 loops (`GL100`) [11] for comparison.

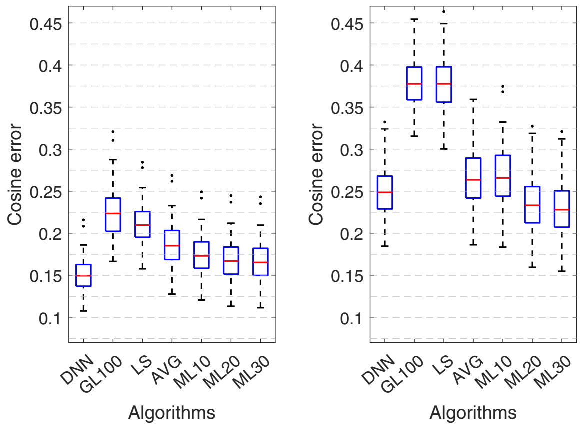
As all of the two-stage phase reconstruction algorithms rely on the same result of the first stage, we compared their second stage by measuring the mean cosine error between the IF and GD estimated from the amplitude by using DNNs and those



(a) IF error

(b) GD error

Fig. 2. Cosine errors between IF and GD calculated from reconstructed phases and those estimated from amplitude using DNNs.



(a) IF error

(b) GD error

Fig. 3. Cosine errors between IF and GD calculated from reconstructed phases and those calculated from the original phase. DNN denotes IF and GD estimated from amplitude using DNNs.

calculated from the reconstructed phases. The cosine error is given as

$$\epsilon(\tilde{\boldsymbol{y}}, \hat{\boldsymbol{y}}) = 1 - \frac{1}{KL}\sum_{k=0}^{K-1}\sum_{l=0}^{L-1}\cos\left(\tilde{y}_{k,l} - \hat{y}_{k,l}\right), \qquad (22)$$

whose range is $[0, 2]$. These second-stage errors indicate how much information of the IF and GD estimated in the first stage is preserved by the second stage. We also used the same error function to calculate the errors between the IF and GD calculated from the reconstructed phases and those calculated from the original phase. That is to replace $\tilde{\boldsymbol{y}}$ in (22) with $\boldsymbol{y}$. In other words, these errors show the overall IF and GD accuracy of the phases estimated by the two-stage algorithms. By comparing the second-stage and overall errors, we can see how the result of the first stage affects the overall result. Finally, we measured the PESQ [24] and STOI [25] of the reconstructed signals. The higher these values, the better the quality of the reconstructed speech.

In our implementation, the STFT was calculated by using a Hamming window with a 32-ms length, 4-ms shift, and 512-point DFT. For estimating the IF and GD, the same as in [17], we used fully connected DNNs with 4 hidden layers, each layer containing 1024 gated tanh units [26], and the last layer containing linear units. These models were trained by the Adam optimizer for 400 epochs. The Linear Algebra Package (LAPACK) [27] was used to calculate the inverse and find the smallest eigenvalue of the Hessian matrix as in [21] and [23], respectively. The weight β of the proposed method was empirically set to 2.4.

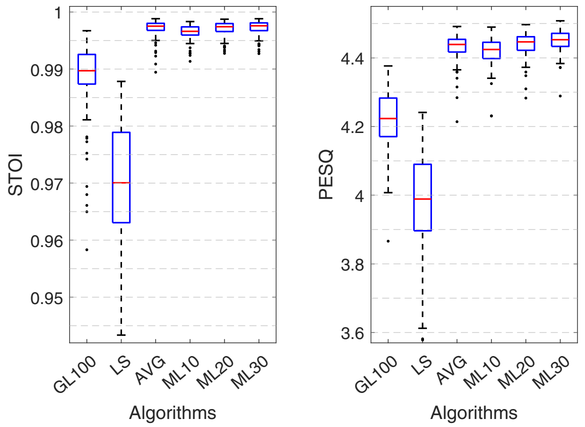
The data used for training were from the training set of the TIMIT dataset [28], which contains broadband recordings of various speakers of eight major dialects of American English. The sampling rate is 16 kHz. The tests were performed on 100 speech samples (50 from male and 50 from female speakers) randomly selected from the test set of the same dataset.

B. Experimental results

Fig. 2 depicts the errors of the second stage in the two-stage algorithms. It can be seen that the proposed ML methods achieved the lowest IF and GD errors in most cases. ML10 is a special case in that it yielded lower IF and higher GD errors than ML20 and ML30. That is because, for the proposed method, we start with zero IF error as in (21), and it seems that 10 loops are not enough to compromise between the IF and GD errors.

Fig. 3 illustrates the overall errors the two-stage algorithms, where the DNN denotes the errors of the IF and GD estimated in the first stage or, in other words, the errors of the DNNs. Fig. 3 reveals a similar pattern as in Fig. 2 showing that the proposed ML methods surpassed the other methods. One difference is that the IF error of ML10 became higher than those of ML20 and ML30, even though it was lower in Fig. 2. We can also see that all of the errors in Fig. 2 were lower than the corresponding errors in Fig. 3. These observations demonstrate that the estimated IF and GD in the first stage play an important role in the two-stage algorithms: even if the second stage yields a low error, the overall error will be high if the error of the first stage is high. Another remarkable point in Fig. 3 is that the IF error of the DNN set a lower bound, while the GD errors of ML20 and ML30 were even lower than that of the DNN. The reason is that, by minimizing both the IF and GD errors at the same time, the more-accurate IF helped to correct the GD of the estimated phase.

Fig. 4 shows the overall performances of the phase reconstruction algorithms. In comparison with the AVG method, the ML20 and ML30 methods had similar STOI scores and higher PESQ scores. For both indices, the proposed ML methods were superior to the 100-loop GL method and significantly outperformed the baseline LS method. These results show the importance of using the amplitude weight and the advantage of the cosine loss function over the quadratic loss function in



(a) STOI score

(b) PESQ score

Fig. 4. Performances of phase reconstruction algorithms.

the two-stage phase reconstruction algorithm.

V. Conclusion

In this paper, we proposed an algorithm reconstructing the STFT phase from its estimated derivatives by using an ML estimation with the assumption of the *von Mises* distribution. The error function was minimized by using a regularized Newton's method. The iterations are fast because the Hessian is a symmetric tridiagonal matrix. Experimental results confirmed the superior performance of the proposed method in comparison with other methods both in terms of IF and GD errors and the quality of the reconstructed signals.

In the future, we will improve on the results by applying other optimization approaches for the ML estimation. We will also reduce the error of the estimated IF and GD by using other neural network architectures.

Acknowledgment

This work was supported by JSPS KAKENHI Grant numbers JP20K19827, JP18K19829, and JP19H04142.

References

[1] P. Mowlaee and J. Kulmer, "Phase estimation in single-channel speech enhancement: Limits-potential," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 8, pp. 1283–1294, 2015.

[2] T. Gerkmann, M. Krawczyk-Becker, and J. Le Roux, "Phase processing for single-channel speech enhancement: History and recent advances," *IEEE Signal Processing Magazine*, vol. 32, no. 2, pp. 55–66, 2015.

[3] 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 Transactions on Audio, Speech and Language Processing*, vol. 26, no. 9, pp. 1559–1569, 2018.

[4] P. Mowlaee and J. Kulmer, "Harmonic phase estimation in single-channel speech enhancement using phase decomposition and SNR information," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 9, pp. 1521–1532, 2015.

[5] J. Le Roux, G. Wichern, S. Watanabe, A. Sarroff, and J. R. Hershey, "The phasebook: Building complex masks via discrete representations for source separation," in *2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 66–70.

[6] Z. Q. Wang, J. Le Roux, and J. R. Hershey, "Multi-channel deep clustering: Discriminative spectral and spatial embeddings for speaker-independent speech separation," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 1–5.

[7] P. Magron, R. Badeau, and B. David, "Model-based STFT phase recovery for audio source separation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 6, pp. 1095–1105, 2018.

[8] 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 *INTERSPEECH*, 2017, pp. 1128–1132.

[9] T. Kaneko, S. Takaki, H. Kameoka, and J. Yamagishi, "Generative adversarial network-based postfilter for STFT spectrograms." in *INTERSPEECH*, 2017, pp. 3389–3393.

[10] Y. Saito, S. Takamichi, and H. Saruwatari, "Text-to-speech synthesis using STFT spectra based on low-/multi-resolution generative adversarial networks," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 5299–5303.

[11] D. Griffin and J. Lim, "Signal estimation from modified short-time Fourier transform," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 32, no. 2, pp. 236–243, 1984.

[12] X. Zhu, G. T. Beauregard, and L. L. Wyse, "Real-time signal estimation from modified short-time Fourier transform magnitude spectra," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 5, pp. 1645–1653, 2007.

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

[14] 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 *2018 International Workshop on Acoustic Signal Enhancement (IWAENC)*, 2018, pp. 286–290.

[15] N. Takahashi, P. Agrawal, N. Goswami, and Y. Mitsufuji, "Phasenet: Discretized phase modeling with deep neural networks for audio source separation." in *INTERSPEECH*, 2018, pp. 2713–2717.

[16] J. Le Roux, G. Wichern, S. Watanabe, A. Sarroff, and J. R. Hershey, "Phasebook and friends: Leveraging discrete representations for source separation," *IEEE Journal of Selected Topics in Signal Processing*, vol. 13, no. 2, pp. 370–382, 2019.

[17] Y. Masuyama, K. Yatabe, Y. Koizumi, Y. Oikawa, and N. Harada, "Phase reconstruction based on recurrent phase unwrapping with deep neural networks," in *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 826–830.

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

[19] R. M. Hegde, H. A. Murthy, and V. R. R. Gadde, "Significance of the modified group delay feature in speech recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 1, pp. 190–202, 2007.

[20] L. Thieling, D. Wilhelm, and P. Jax, "Recurrent phase reconstruction using estimated phase derivatives from deep neural networks," in *2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 7088–7092.

[21] B. N. Datta, *Numerical linear algebra and applications*. Siam, 2010, vol. 116.

[22] D. W. Marquardt, "An algorithm for least-squares estimation of nonlinear parameters," *Journal of the society for Industrial and Applied Mathematics*, vol. 11, no. 2, pp. 431–441, 1963.

[23] W. Kahan, "Accurate eigenvalues of a symmetric tri-diagonal matrix," Computer Science Dept., Stanford University, Tech. Rep., 1966.

[24] 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 *2001 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 2, 2001, pp. 749–752.

[25] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen, "An algorithm for intelligibility prediction of time–frequency weighted noisy speech," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 7, pp. 2125–2136, 2011.

[26] A. V. D. Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, "Wavenet: A generative model for raw audio," *arXiv preprint arXiv:1609.03499*, 2016.

[27] E. Anderson, Z. Bai, C. Bischof, L. S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney *et al.*, *LAPACK users' guide*. SIAM, 1999.

[28] J. S. Garofolo, *TIMIT acoustic phonetic continuous speech corpus*. Linguistic Data Consortium, 1993.