Frequency-Domain Signal-to-Noise Ratios Illuminate the Effects of the Spectral Consistency Constraint and Griffin-Lim Algorithms

Stephen Voran and Jaden Pieper

{svoran, jpieper}@ntia.gov, NTIA, Institute for Telecommunication Sciences, Boulder, CO, USA

***Abstract*—The restoration of degraded audio signals is often performed on complex-valued frequency-domain (FD) representations. This requires manipulation of either magnitudes and phases or real and imaginary parts. In general, these manipulations do not produce consistent representations. The consequence is that the magnitudes and phases (or real and imaginary parts) of the restored time-domain signal (which are always consistent) do not match the generally inconsistent values imposed during FD restoration. In colloquial terms, "What we get is not what we asked for." The enforcement of consistency is always heard in the resulting audio and is known in principle, but it can be better understood. We present two-dimensional FD SNR frameworks (e.g., magnitude/phase or real/imaginary) that visually reveal how consistency enforcement changes the *applied* FD restorations to arrive at the *achieved* FD restorations. We also show how extended Griffin-Lim algorithms can reduce and direct, but not eliminate, the changes produced by consistency enforcement. We apply objective estimators to connect this work to estimated speech quality and intelligibility. This work can inform machine learning training and architecture choices that must balance restoration efforts across two dimensions (e.g., magnitude and phase) to arrive at the best possible speech quality.**

1. INTRODUCTION

Restoring an audio signal that has been degraded by other audio signals, reverberation, coding, or other means is an important and enduring problem that has received much attention. There are viable time domain (TD) approaches to audio restoration (e.g., [1]–[6]) but the use of the frequency-domain (FD) opens up additional opportunities. Hundreds of examples are referenced in [7], [8]. Audio signals are often transformed by the discrete-time short-time Fourier transform (STFT) so that restoration can be accomplished in the FD. The audio signal x is windowed and transformed to the FD representation $X_{k,t}$:

$$X_{k,t} = \sum_{i=0}^{N-1} w_i x_{(t \cdot N_s + i)} e^{-j2\pi ki/N} \ , \tag{1}$$

where w_i, $i = 0$ to $N-1$ are window values, $t = 0, 1, \ldots, N_t - 1$, is the window index, and $k = 0$ to $N-1$ is the STFT frequency index. The window stride is N_s. The spectral values $X_{k,t}$ are complex.

Let F represent the windowing and STFT function given in (1). This function maps a set of real TD audio samples $\{x_i\}$ to a set of complex FD values (a spectral representation) $\{X_{k,t}\}$. We denote this (somewhat informally) as $\{X_{k,t}\} = F(\{x_i\})$. Window length, stride, and shape are chosen so that the windowing operation can be inverted using overlap-and-add. The STFT is always invertible. Thus F^{-1} exists and is used to recreate the original audio signal: $\{x_i\} = F^{-1}(\{X_{k,t}\})$.

FD restoration may be accomplished through signal processing algorithms, machine learning (ML) architectures, or combinations of these. If $X_{k,t}$ come from an audio signal and $\hat{X}_{k,t}$ come from a degraded version of that signal, then FD restoration involves finding a restoration function f such that

$$f(\hat{X}_{k,t}) = \tilde{X}_{k,t} \approx X_{k,t}. \tag{2}$$

The notion of consistency is key because it controls the relationship between the FD restorations *applied* by f and the FD restorations that are actually *achieved*. If a set of values $\{X_{k,t}\}$ is produced by a signal via $\{X_{k,t}\} = F(\{x_i\})$, then it is easy to show that

$$F(F^{-1}(\{X_{k,t}\})) = \{X_{k,t}\}. \tag{3}$$

When $\{X_{k,t}\}$ has the property given in (3) we say that $\{X_{k,t}\}$ is *consistent*. $\{\tilde{X}_{k,t}\}$ will be consistent, but the restoration process f in (2) will not preserve consistency in general, so

$$F(F^{-1}(\{\tilde{X}_{k,t}\})) \neq \{\tilde{X}_{k,t}\}. \tag{4}$$

Equation (4) says that when we convert the restored signal $\{\tilde{X}_{k,t}\}$ to a TD signal, then convert that signal back to the FD to check its spectral properties, those spectral properties will not match those imposed by the restoration process — "What we get is not what we asked for." The consequences of enforcing consistency when creating the TD output $\{x_i\}$ are unavoidable. The process has been investigated and described via orthogonal projections [9]–[11] but can still be better understood. We develop and apply two-dimensional FD SNR tools to better reveal the inner workings of this process.

In the next section we present complete error functions that measure the success of restoration. These allow us to build pairs of FD SNRs in Section 3 and apply them to show how TD SNRs translate to FD SNRs for various audio signals and window lengths, strides, and shapes. In Section 4 we define a simple yet realistic FD restoration simulator and in Section 5 we show that it moves a degraded audio signal through the magnitude/phase SNR plane in a controlled fashion. We also show how enforcing consistency shifts those restored signals and that the constraints and iterations imposed by extended Griffin-Lim Algorithms (GLA) can reduce and direct those shifts, but not eliminate them. Finally, in Section 6 we link the example two-dimensional FD SNR results to estimated speech quality and show how one can focus restoration efforts to achieve the best estimated speech quality.

2. ERROR FUNCTIONS

The tightness of the approximation in (2) gives the effectiveness of the restoration algorithm, and this is quantified by error (aka cost or loss) functions. The spectral values are often written as magnitudes $|X_{k,t}|$, and phases $\angle X_{k,t}$. This representation has a solid physical motivation: $|X_{k,t}|^2$ can be related to audio power at frequency k and time t. So one well-motivated pair of error functions is

$$\epsilon^p_{\text{mag}}(k,t) = \left| \, |\tilde{X}_{k,t}| - |X_{k,t}| \, \right|^p, \text{ and} \tag{5}$$

$$\epsilon^p_{\text{ph}}(k,t) = \left| W_\pi(\angle \, \tilde{X}_{k,t} - \angle X_{k,t}) \right|^p, \tag{6}$$

where W_π is the function that wraps angles into $[-\pi, \pi)$ and $p > 0$. Real and imaginary errors are also an option:

$$\epsilon^p_{\text{re}}(k,t) = \left| \Re(\tilde{X}_{k,t}) - \Re(X_{k,t}) \right|^p, \text{ and} \tag{7}$$

$$\epsilon^p_{\text{im}}(k,t) = \left| \Im(\tilde{X}_{k,t}) - \Im(X_{k,t}) \right|^p. \tag{8}$$

And another option is proposed in [12]:

$$\epsilon^p_{\text{rc}}(k,t) = \left| \tilde{X}_{k,t} \frac{|X_{k,t}|}{|\tilde{X}_{k,t}|} - X_{k,t} \right|^p = \left| 2|X_{k,t}| \sin\left(\frac{\epsilon^1_{\text{ph}}(k,t)}{2} \right) \right|^p, \tag{9}$$

using $p = 2$. We call this the squared-length of the residual chord because after $\tilde{X}$ has been scaled to have the same magnitude as X, the remaining error is a chord of the circle defined by $|X|$. We define a "complete error function" to be a function $\epsilon(\tilde{X}_{k,t}, X_{k,t})$ such that

$$\epsilon(\tilde{X}_{k,t}, X_{k,t}) = 0 \implies \tilde{X}_{k,t} = X_{k,t}, \forall k, t. \tag{10}$$

The building blocks introduced in (5)-(9) can be used to build three different complete error functions:

$$\epsilon^p_{\text{mp}} = \epsilon^p_{\text{mag}} + \lambda\epsilon^p_{\text{ph}}, \quad \epsilon^p_{\text{ri}} = \epsilon^p_{\text{re}} + \lambda\epsilon^p_{\text{im}}, \quad \epsilon^p_{\text{mrc}} = \epsilon^p_{\text{mag}} + \lambda\epsilon^p_{\text{rc}}, \tag{11}$$

with $\lambda > 0$ (arguments k and t omitted here). A complete error function can be used train, tune, or evaluate restoration algorithms.

Much early work focused on restoring magnitudes only (minimizing ϵ^p_{mag}) and a few different examples are in [13]–[18]. The success of this approach shows magnitude restoration is key, but it is also true that magnitude restoration alone cannot lead to complete restoration. As more sophisticated modeling and algorithms have been developed, researchers have addressed including phase restoration (minimizing ϵ^p_{mp}) [19]–[25] and also restoration of complex spectral coefficients directly (minimizing ϵ^p_{ri}) [26], [27]. Minimization of ϵ^2_{mrc} was used to remove reverberation in [12]. $\lambda = 0.1$, 1.0, and 2.0 were considered, $\lambda = 1.0$ was best when reverberation times were below one second, and $\lambda = 0.1$ showed a slight advantage for longer reverberation times. Note that ML training can benefit from *over-complete* loss functions, for example when ϵ_{ri} is augmented with ϵ_{mag} and even additional TD loss as discussed in [28] and seen in various forms in [29]–[34].

In ML training, λ balances how much a network concerns itself with different components of the restoration problem in order to create the best sounding restoration. One way to determine the best value for λ is through a hyperparameter search. Another approach is to develop an understanding of how the constituent errors (e.g., ϵ^p_{mag} and ϵ^p_{ph}) relate to sound quality both individually and jointly. Our work serves the second approach and thus may help to guide the selection of λ and similar weights, or to better focus hyperparameter searches.

3. SNRS BUILT FROM ERROR FUNCTIONS

We now use the error functions in (5) to (9) to build a family of FD SNRs. We use SNRs because they correctly recognize that the importance of a given perturbation depends on the power of the signal that is being perturbed. Ratios of this type are used extensively in audio signal processing. Note that the "noise" in SNR is just a difference which can be due to coding, reverberation, filtering, etc., not just added acoustic noise. The magnitude, phase, real part, imaginary part, and residual chord SNRs are defined, respectively as:

$$\text{SNR}_{\text{mag}} = m_k\left(10\log_{10}\left(\frac{m_t(|X_{k,t}|^2)}{m_t(\epsilon^2_{\text{mag}}(k,t))}\right)\right), \tag{12}$$

$$\text{SNR}_{\text{ph}} = m_k\left(10\log_{10}\left(\frac{v_t(\angle X_{k,t})}{m_t(\epsilon^2_{\text{ph}}(k,t))}\right)\right), \tag{13}$$

$$\text{SNR}_{\text{re}} = m_k\left(10\log_{10}\left(\frac{m_t(\Re(X_{k,t})^2)}{m_t(\epsilon^2_{\text{re}}(k,t))}\right)\right), \tag{14}$$

$$\text{SNR}_{\text{im}} = m_k\left(10\log_{10}\left(\frac{m_t(\Im(X_{k,t})^2)}{m_t(\epsilon^2_{\text{im}}(k,t))}\right)\right), \tag{15}$$

$$\text{SNR}_{\text{rc}} = m_k\left(10\log_{10}\left(\frac{m_t(|X_{k,t}|^2)}{m_t(\epsilon^2_{\text{rc}}(k,t))}\right)\right), \tag{16}$$

where the function m_t is the mean over time, m_k is the mean over the frequency index k, and v_t returns the variance of the phase angles over time. This variance is calculated after wrapping phase values into the 2π range that is centered on the mean of the phase angles. These equations can produce SNRs for each frequency which are useful in general, but for the present application we have applied a mean over frequency to define a single SNR value.

We use FD SNRs to view degraded signals and the restoration process. To have a complete view we must use a pair of SNRs (taken from (12) to (16)) that correspond to a complete error function (see (11)). Given this, the three pairs of FD SNRs we have considered are (SNR_{mag}, SNR_{ph}), (SNR_{re}, SNR_{im}), and (SNR_{mag}, SNR_{rc}). First we show the FD restoration problem in terms of FD SNRs for different signal types and TD SNRs.

Our tools and visualizations work for any type of degradation, but the examples we present here are limited to degradation by the addition of a second audio signal. The desired signal is either fullband speech or music, and the degrading signal can be speech, music, office noise, street noise, or white noise, always with sample rate 48,000 smp/s. We calculate TD SNR (SNR_{TD}) using total power, except in the case of speech where we use active speech power [35], [36]. Figure 1 shows the resulting SNR_{mag} and SNR_{ph} values for the ten cases with SNR_{TD} values from 5 to 30 dB. Results shown are means over 60 audio files, each 5 seconds long.

As expected, increasing SNR_{TD} increases both SNR_{mag} and SNR_{ph} for all signals and degradation, but each case brings unique SNR relationships. Depending on signal types, at $\text{SNR}_{\text{TD}} = 5$ dB, SNR_{mag} ranges from -16.9 to 13.2 dB, while SNR_{ph} ranges from 0.5 to 7.5 dB. For fixed SNR_{TD} and degrading signal type, music has better SNR_{ph} than speech but worse SNR_{mag}. This phase advantage increases as SNR_{TD} increases but the magnitude penalty is fairly constant. Across the five degrading signals, speech harms phase the least while white noise harms phase (and magnitude) the most.

Figure 1 results used the 20 ms square-root periodic Hamming window with 10 ms stride. The effect of changing these parameters depends on the signal combination, but we summarize here by reporting upper bounds. Across the five SNRs, varying window length from 5 to 80 ms results in changes no greater than 6.7 dB. Switching between Hamming and Hann windows gives changes no greater than 3.6 dB. With the 20 ms Hamming window, switching the window stride between 10 and 5 ms produces almost no changes at all, with the exception of phase SNR where change was limited to 0.24 dB.

As expected, repeating Fig. 1 for the (SNR_{re}, SNR_{im}) plane shows all points close to the line $y = x$ and linearly related to SNR_{TD}, so we do not expend any space to display it here. Finally, repeating the figure for SNR_{mag} and SNR_{rc} shows less variation due to signal types than Fig. 1. This reduced variation suggests ϵ^p_{rc} might be a more stable and versatile error function than ϵ^p_{ph}. On the other hand, the residual chord cannot be manipulated separately from magnitude, so it cannot serve as an independent restoration dimension as magnitude, phase, real part, and imaginary part can. For the remainder of this paper we present results in the (SNR_{mag}, SNR_{ph}) plane only.

4. AUDIO RESTORATION SIMULATIONS

We have simulated magnitude, phase, real part, and imaginary part restoration. These simulations are controlled by a single parameter r that ranges from zero (no restoration) to one (full restoration). This simulation is motivated by simplified generalizations on real restoration algorithms: with mild restoration the largest errors are detected and partially corrected, while more aggressive settings cause attempted correction of more and more of the time-frequency elements.

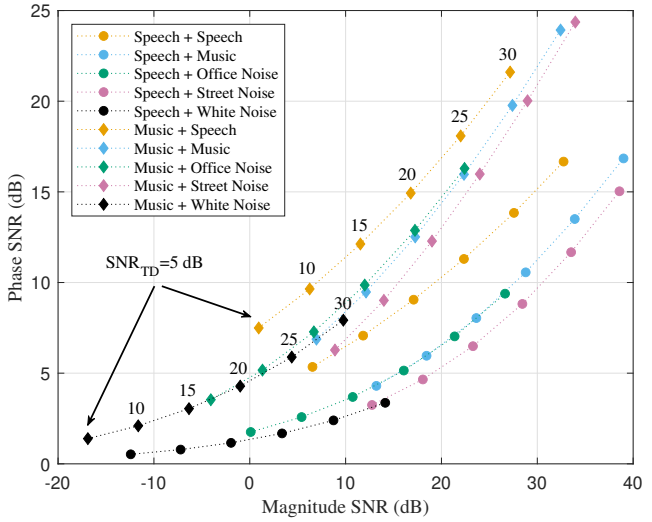

Fig. 1: SNR_{mag} and SNR_{ph} values for speech and music, each degraded by addition of one of five other signals. Markers correspond to TD SNRs from 5 dB (leftmost of a group) to 30 dB (rightmost of a group) in 5 dB steps.

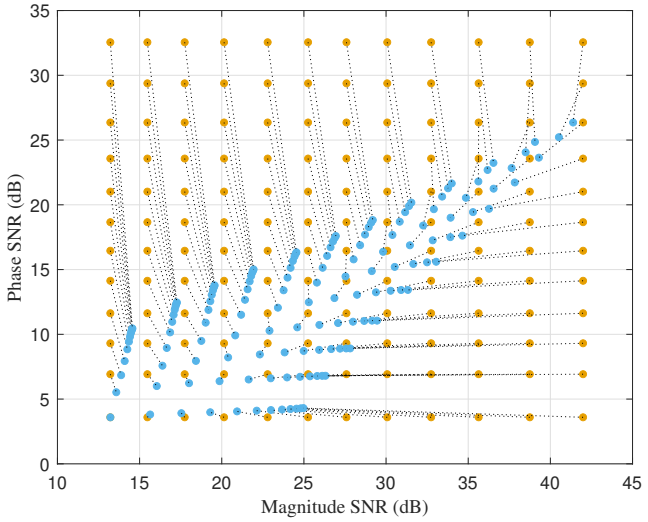

Fig. 2: Gold circles show FD SNR locations of (inconsistent) restored spectra and blue circles show new locations after consistency has been enforced.

For a degraded signal $\{X_{k,t}\}$ with n_t time-frequency elements, we achieve this by defining a subset of $\lceil n_t\sqrt{r}\,\rceil$ time-frequency elements to restore, where $\lceil\cdot\rceil$ rounds to the nearest integer. We call the set of elements to restore R. Let $Y_{k,t}$ be the magnitude, phase, real part, or imaginary part of $X_{k,t}$, and similarly for $\tilde{Y}_{k,t}$ and $\hat{Y}_{k,t}$. We construct R so that it contains the time-frequency elements with the largest values of $|\hat{Y}_{k,t} - Y_{k,t}|$, i.e., the largest deviations between clean and degraded signals. The oracle-based simulated restoration is a mixture of the clean and degraded signals in the domain of interest:

$$\tilde{Y}_{k,t} = \begin{cases} \sqrt{r}Y_{k,t} + (1-\sqrt{r})\,\hat{Y}_{k,t}, & \forall(k,t) \in R, \\ \hat{Y}_{k,t}, \quad \text{otherwise.} \end{cases} \tag{17}$$

Thus, $\sqrt{r}$ is the proportion of elements that are restored and for each of these elements $\sqrt{r}$ is also the amount of restoration applied. The level of restoration can be viewed as the product of these two effects, so controlling each effect with $\sqrt{r}$ instead of r helps to linearize the relationship between restoration level and r. The parallel roles played by real and imaginary parts reduces the motivation to explore that restoration space, so we report only on magnitude/phase restoration.

5. CONSISTENCY AND GRIFFIN-LIM ALGORITHM VIEWED IN THE MAGNITUDE SNR - PHASE SNR PLANE

We can now view simulated magnitude/phase restoration, the effects of enforcing consistency, and the effects of extended GLAs in the (SNR_{mag},SNR_{ph}) plane. All results going forward are averages over 24 speech files, each 5 seconds long, with street noise at SNR_{TD}= 5 dB, using a 20 ms Hamming window with 10 ms stride. The grid of gold circles in Fig. 2 show restoration cases obtained by setting r to 12 carefully selected values from 0.0 to 0.93 for both magnitude restoration and phase restoration. Figure 1 shows that the (SNR_{mag}, SNR_{ph}) point for the unrestored speech in this case is (13.2, 3.6) and this corresponds to the lower left point in Fig. 2 where no restoration was performed. For each restored spectral representation (gold circle) we converted to a TD signal that would be heard by listeners, then converted it back to the FD and measured FD SNRs. These new FD SNRs are shown by blue circles in Fig. 2, and each is connected to the corresponding gold circle by a dotted line. This yields a graphical view of the inconsistent nature of the restored FD representations.

Figure 2 shows that consistency enforces a natural relationship between between SNR_{mag} and SNR_{ph} — for this example, SNR_{mag} averages 12.2 dB greater than SNR_{ph}. So larger imbalances between magnitude restoration and phase restoration produce results that must move farther (longer dotted line) when consistency is enforced. That spread of consistent points is smaller in the upper right because there are few consistent ways to *closely* approximate the original audio signal. The spread is larger in the lower left because there are many different consistent ways to *roughly* approximate the original audio signal. The practical value of Fig. 2 is that it shows the gaps between applied restorations and the restorations that are actually achieved. Before we use Fig. 2 to deduce which restoration points to pursue, we must also consider what follows below — these gaps can often be moderated, but not eliminated.

If we are more confident about the restoration in one dimension than the other, we can use iteration to minimize the change in the favored dimension when generating the TD signal. The classic example of this process is to restore only magnitudes, then use the original GLA [37] to find a phase solution that is consistent with the restored magnitudes. The algorithm uses F^{-1} to convert the inconsistent $\{\tilde{X}_{k,t}\}$ to $\{\tilde{x}_i\}$, then F to convert $\{\tilde{x}_i\}$ back to a new consistent set of $\{\tilde{X}_{k,t}\}$. $\{\tilde{X}_{k,t}\}$ is then forced to have the desired magnitudes, generally creating inconsistency, and the process repeats. When the consistent magnitudes are suitably close to the desired magnitudes, the process is complete, resulting in a TD signal that has (almost) the desired magnitudes. This approach can be used to enforce other constraints as well.

We used GLA to enforce the restored magnitude and, in separate experiments, to enforce the restored phases. Figure 3 shows the results in the (SNR_{mp}, SNR_{ph}) plane. As before, gold circles show FD SNR locations after restoration and blue circles show the new locations after consistency is enforced. The sequence of magenta circles shows results of 2, 4, 8, ... 128 iterations of the GLA when the restored magnitude is enforced at every iteration. The sequence of green circles show the same but when restored phase is enforced instead of magnitude. To maintain legibility, we show only eight example restoration cases.

In six of these cases, enforcing the restored magnitude causes GLA iterations to increase SNR_{mag} with each iteration (magenta), approaching, but not reaching, the SNR_{mag} value of the the original

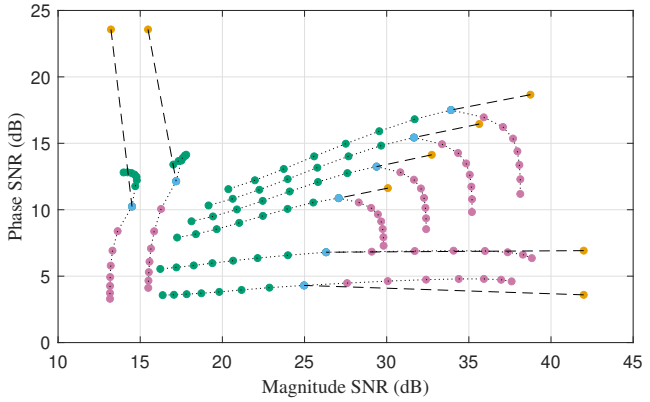

Fig. 3: Gold shows FD SNR locations after restoration (inconsistent), blue after consistency is enforced, magenta for 2, 4, 8, ... 128 iterations of GLA with restored magnitude enforced, green for GLA with restored phase enforced.

restoration (gold). This comes at some cost to the SNR_{ph}, especially if it has been significantly improved by restoration. In the remaining two cases, there is significant phase restoration but little magnitude restoration (an unlikely situation in practice given that phase restoration is significantly more difficult than magnitude restoration) and GLA that enforces the restored phase (green circles) does improve SNR_{ph} a small amount. Every blue, magenta, and green circle corresponds to a TD signal, but the gold circles do not. A gold circle is an example of "what we asked for," and the associated blue, magenta, and green circles show a range of options for "what we get." Any of these options can be selected by picking the GLA constraint and the number of GLA iterations performed. As a practical matter, figures like this can inform which restoration points we pursue, since they show how a restoration point and GLA work together to produce a final result.

6. ESTIMATED SPEECH QUALITY

We connect the (SNR_{mag}, SNR_{ph}) plane to estimated speech quality using WB-PESQ [38] and ViSQOL [39] and to estimated intelligibility using STOI [40]. Figure 4 color codes ViSQOL speech quality estimates (ver. 238, nominal range 1.0 to 5.0) for every restoration case and every GLA option in the current example. It shows how each achieved point in the plane relates to speech quality. It is clear that improving either SNR can improve speech quality, and that both SNRs must be improved to obtain the highest speech quality.

Each point in Fig. 4 represents a combination of a magnitude restoration level, a phase restoration level, and a GLA choice. Figure 5 provides a simplified view. For each restoration case, Fig. 5 color codes highest speech quality that can be obtained by selecting the best GLA strategy. To acknowledge ViSQOL estimation noise we use a "soft best" that includes GLA strategies that give ViSQOL scores that are within 0.2 (5% of the nominal full scale) of the mathematical best. A "P" marks restoration cases where GLA should enforce restored phase to get best results. "M" marks those where enforcing restored magnitude is best. For unmarked cases, the best result is achieved without GLA. The figure also shows approximate contours of constant quality for ViSQOL scores of 3.5, 4.0 and 4.5. Figures like 5 can be used to set restoration expectations and pick GLA options. For example, SNR_{mag}= 30 dB, SNR_{ph}= 20 dB gives estimated speech quality of 4.0. If we seek to increase that to 4.5, the figure suggests multiple options. Three of those would be to increase SNR_{mag} by 15 dB, increase SNR_{ph} by 8 dB, or increase both by 5 dB.

Figures 4 and 5 do not change much when we show PESQ in place of ViSQOL. When we show STOI there is much less variation in

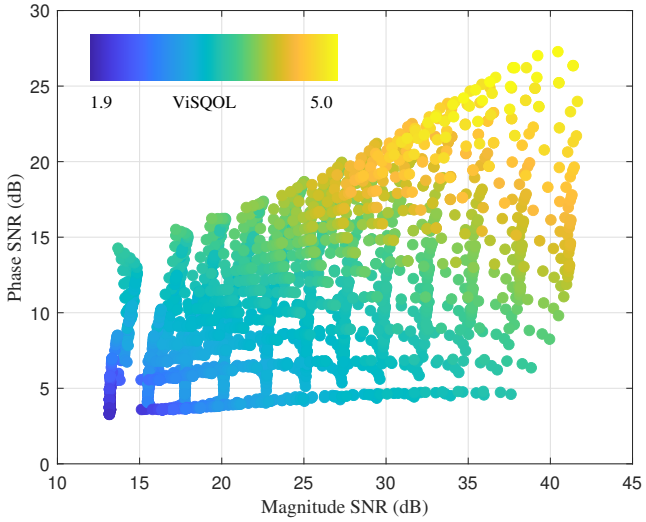

Fig. 4: Relationship between FD SNRs and speech quality estimated by ViSQOL for all combinations of restoration cases and GLA options.

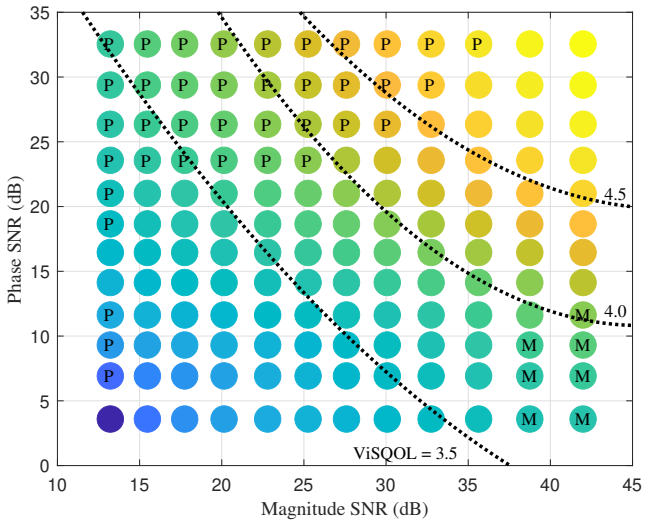

Fig. 5: Previous figure simplified by picking best GLA option for each restoration case and using the FD SNR coordinates of restorations before consistency is enforced. "P" and "M" indicate that the best GLA option enforces restored phase or magnitude, resp. Dotted lines are approximate constant quality contours for ViSQOL values 3.5, 4.0 and 4.5.

the upper right, consistent with the fact that many different levels of higher quality produce full intelligibility.

7. CONCLUSION

We have developed an FD SNR analysis framework and tools that allow us to see how the consistency constraint and GLA function as intermediaries between FD restorations and the TD signal that a listener will hear. Understanding the actions of these intermediaries affords the opportunity to plan for and work with their effects, rather than simply tolerate their effects. We provided example results for a single case: speech with street noise at SNR_{TD}= 5 dB. The patterns seen here, but not the specific values, do extend to other cases, as one would expect based on Fig. 1. Our reporting here focuses on the (SNR_{mag},SNR_{ph}) plane, but additional insights can be gained from considering SNR_{re}, SNR_{im}, and SNR_{rc}, and all of these are supported by the tools we offer at https://github.com/NTIA/Audio-FD-SNRs.

REFERENCES

[1] S. Canazza, G. De Poli, and G. A. Mian, "Restoration of audio documents by means of extended Kalman filter," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 6, pp. 1107–1115, 2010.

[2] A. Pandey and D. Wang, "A new framework for CNN-based speech enhancement in the time domain," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 7, pp. 1179–1188, 2019.

[3] ——, "Dense CNN with self-attention for time-domain speech enhancement," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 1270–1279, 2021.

[4] T. Nakamura, S. Kozuka, and H. Saruwatari, "Time-domain audio source separation with neural networks based on multiresolution analysis," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 1687–1701, 2021.

[5] M. Kolbæk, Z.-H. Tan, S. H. Jensen, and J. Jensen, "On loss functions for supervised monaural time-domain speech enhancement," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 825–838, 2020.

[6] A. A. Nugraha, D. Di Carlo, Y. Bando, M. Fontaine, and K. Yoshii, "Time-domain audio source separation based on gaussian processes with deep kernel learning," in *Proc. 2023 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 2023.

[7] P. Loizou, *Speech Enhancement, Theory and Practice*. Boca Raton, Florida: CRC Press, 2013.

[8] E. Vincent, T. Virtanen, and S. Gannot, Eds., *Audio Source Separation and Speech Enhancement*. Hoboken, New Jersey: Wiley, 2018.

[9] J. Le Roux, N. Ono, and S. Sagayama, "Explicit consistency constraints for STFT spectrograms and their application to phase reconstruction," in *Proc. ISCA Tutorial and Research Workshop on Statistical and Perceptual Audition*, Sep. 2008, pp. 23–28.

[10] T. Peer, S. Welker, and T. Gerkmann, "Beyond Griffin-Lim: Improved Iterative Phase Retrieval for Speech," in *Proc. 2022 International Workshop on Acoustic Signal Enhancement*, Sep. 2022.

[11] T. Peer and T. Gerkmann, "Intelligibility prediction of speech reconstructed from its magnitude or phase," in *Proc. Speech Communication; 14th ITG Conference*, 2021.

[12] J. Zhang, M. D. Plumbley, and W. Wang, "Weighted magnitude-phase loss for speech dereverberation," in *Proc. 2021 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2021, pp. 5794–5798.

[13] S. Boll, "Suppression of noise in speech using the SABER method," in *Proc. 1978 IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 3, Apr 1978, pp. 606–609.

[14] B. L. Sim, Y. C. Tong, J. S. Chang, and C. T. Tan, "A parametric formulation of the generalized spectral subtraction method," *IEEE Transactions on Speech and Audio Processing*, vol. 6, no. 4, pp. 328–337, Jul. 1998.

[15] L.-P. Yang and Q.-J. Fu, "Spectral subtraction-based speech enhancement for cochlear implant patients in background noise," *The Journal of the Acoustical Society of America*, vol. 117, no. 3, pp. 1001–1004, 2005.

[16] J. Li, S. Sakamoto, S. Hongo, M. Akagi, and Y. Suzuki, "Noise reduction based on adaptive beta-order generalized spectral subtraction for speech enhancement," in *Proc. Interspeech 2007*, Aug 2007, pp. 802–805.

[17] T. Virtanen, J. Gemmeke, B. Raj, and P. Smaragdis, "Compositional models for audio processing: Uncovering the structure of sound mixtures," *IEEE Signal Processing Magazine*, vol. 32, no. 2, pp. 125–144, Mar. 2015.

[18] Z. Zhong, H. Shi, M. Hirano, K. Shimada, K. Tateishi, T. Shibuya, S. Takahashi, and Y. Mitsufuji, "Extending audio masked autoencoders toward audio restoration," in *Proc. 2023 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 2023.

[19] H. Erdogan, J. R. Hershey, S. Watanabe, and J. Le Roux, "Phase-sensitive and recognition-boosted speech separation using deep recurrent neural networks," in *Proc. 2015 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2015, pp. 708–712.

[20] D. Yin, C. Luo, Z. Xiong, and W. Zeng, "PHASEN: A phase-and-harmonics-aware speech enhancement network," in *Proc. AAAI Conference on Artificial Intelligence*, 2019.

[21] M. Krawczyk and T. Gerkmann, "STFT phase reconstruction in voiced speech for an improved single-channel speech enhancement," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 12, pp. 1931–1940, 2014.

[22] 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.

[23] Z. Ni and M. I. Mandel, "Mask-dependent phase estimation for monaural speaker separation," in *Proc. 2020 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2020, pp. 7269–7273.

[24] D. Kim, H. Han, H.-K. Shin, S.-W. Chung, and H.-G. Kang, "Phase continuity: Learning derivatives of phase spectrum for speech enhancement," in *Proc. 2022 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2022, pp. 6942–6946.

[25] Y. Masuyama, K. Yatabe, K. Nagatomo, and Y. Oikawa, "Online phase reconstruction via DNN-based phase differences estimation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 31, pp. 163–176, 2023.

[26] D. S. Williamson, Y. Wang, and D. Wang, "Complex ratio masking for monaural speech separation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 3, pp. 483–492, 2016.

[27] Z.-Q. Wang, G. Wichern, S. Watanabe, and J. Le Roux, "STFT-domain neural speech enhancement with very low algorithmic latency," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 31, pp. 397–410, 2023.

[28] Z.-Q. Wang, G. Wichern, and J. Le Roux, "On the compensation between magnitude and phase in speech separation," *IEEE Signal Processing Letters*, vol. 28, pp. 2018–2022, 2021.

[29] S.-W. Fu, T.-y. Hu, Y. Tsao, and X. Lu, "Complex spectrogram enhancement by convolutional neural network with multi-metrics learning," in *Proc. 2017 IEEE 27th International Workshop on Machine Learning for Signal Processing*, 2017.

[30] S. Wisdom, J. R. Hershey, K. Wilson, J. Thorpe, M. Chinen, B. Patton, and R. A. Saurous, "Differentiable consistency constraints for improved deep speech enhancement," in *Proc. 2019 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2019, pp. 900–904.

[31] J. Lee and H.-G. Kang, "A joint learning algorithm for complex-valued T-F masks in deep learning-based single-channel speech enhancement systems," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 6, pp. 1098–1108, 2019.

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

[33] A. Defossez, G. Synnaeve, and Y. Adi, "Real time speech enhancement in the waveform domain," in *Proc. Interspeech 2020*, Oct. 2020, pp. 3291–3295.

[34] H. Wang and D. Wang, "Neural cascade architecture with triple-domain loss for speech enhancement," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 30, pp. 734–743, 2022.

[35] *ITU-T Recommendation P.56, Objective measurement of active speech level*, International Telecommunication Union: Geneva, 2011.

[36] *ITU-T Recommendation P.191, Software tools for speech and audio coding standardization*, International Telecommunication Union: Geneva, 2005.

[37] 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.

[38] *ITU-T Recommendation P.862.2, Wideband Extension to Recommendation P.862 for the Assessment of Wideband Telephone Networks and Speech Codecs*, International Telecommunication Union: Geneva, 2007.

[39] A. K. A. Hines, J. Skoglund and N. Harte, "ViSQOL: an objective speech quality model," *EURASIP Journal on Audio, Speech, and Music Processing*, May 2015.

[40] 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.

