Generalized approach to compensate for low- and high-frequency errors in fast Fourier transform-based phase screen simulations

Sorabh Chhabra,[a,]* Jyotirmay Paul,[a,b]
Anamparambu N. Ramaprakash◉,[a,c,d] and Avinash Surendran◉[e]

[a]Inter-University Center for Astronomy and Astrophysics, Pune, Maharashtra, India
[b]University of Liège, Space Sciences Technologies & Astrophysics Research Institute, Liège, Belgium
[c]California Institute of Technology, Cahill Center for Astronomy and Astrophysics, Pasadena, California, United States
[d]Institute of Astronomy, Foundation for Research and Technology-Hellas, Heraklion, Greece
[e]W. M. Keck Observatory, Kamuela, Hawaii, United States

Abstract. Fast Fourier transform-based phase screen simulations give accurate results only when the screen size (G) is much larger than the outer scale parameter (L_0). Otherwise, they fall short in correctly predicting both the low and high frequency behaviors of turbulence-induced phase distortions. Subharmonic compensation is a commonly used technique that aids in low-frequency correction but does not solve the problem for all values of screen size to outer scale parameter ratios (G/L_0). A subharmonics-based approach will lead to unequal sampling or weights calculation for subharmonics addition at the low-frequency range and patch normalization factor. We have modified the subharmonics-based approach by introducing a Gaussian phase autocorrelation matrix that compensates for these shortfalls. We show that the maximum relative error in structure function with respect to theoretical value is as small as 0.5% to 3% for (G/L_0) ratio of 1/1000 even for screen sizes up to 100 m diameter. © *2021 Society of Photo-Optical Instrumentation Engineers (SPIE)* [DOI: [10.1117/1.JATIS.7.2.025007](https://doi.org/10.1117/1.JATIS.7.2.025007)]

1 Introduction

Accurately simulating the atmospheric turbulence behavior is well recognized as very challenging. For a variety of purposes such as the design and development of adaptive optics systems, speckle imaging techniques, and atmospheric propagation studies, it is essential to simulate good atmospheric phase screen models. Methods based on Zernike polynomial expansions,[1] fast Fourier transform (FFT)-based methods,[2–8] and low-frequency optimization method[9] have been in use for this purpose. The Zernike polynomial method, which is widely in use, has a limitation due to the maximum number of coefficients needed for accurate compensation. The optimization method, which compensates accurately for low-frequency part of the spectrum using unequal sampling and unequal weight in low-frequency region, does not cover high-frequency deficiencies. Among these, FFT-based methods are computer memory size friendly and widely accepted. But, FFT operators assume uniform sampling for the non-uniformly distributed phase power spectrum, which can lead to underestimation in the low- and high-frequency out of band regions, as shown in Fig. 1. Thus, it has limitations in recreating the true phase power spectrum. To compensate for low-frequency components,
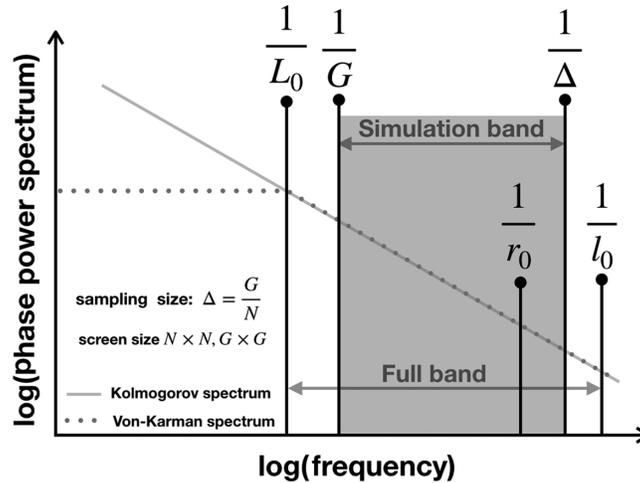
Fig. 1 Comparison between simulation band and full band.

Johansson and Gavel[3] suggested employing the modified subharmonics equation (an adaptation from Lane et al.[10]), which works well up to an infinite outer scale length. Sedmak[6] later compared the performance of this method with that of Lane et al.[10] by actually calculating the phase structure function from the simulated screen. He improved upon Lane et al.[10] by employing different fine-tuned subharmonic weights for different G/L_0 ratios. Results from his analysis show that these FFT-based simulations are accurate for large screen size (G) to outer scale parameter (L_0) ratios. For a screen size of $G = 200$ m and outer scale of $L_0 = 25$ m, the maximum relative error in the simulation approaches 1%. Our simulations demonstrate that the errors from low-frequency components start shooting up once we move to smaller G/L_0 ratios, even after compensating with modified subharmonics.

In Fig. 1, we show[11] a typical situation where the simulation band $(\frac{1}{G} - \frac{1}{\Delta})$ is actually smaller than full band $(\frac{1}{L_0} - \frac{1}{l_0})$, where Δ is the sampling size defined as the ratio of screen size G to sampling number N and l_0 is the inner scale parameter. In practice, the simulations are often curtailed at the low-frequency end, to a few times the optical beam size (say as determined by the telescope or laser beam diameter), whereas at the high frequency end, they often extend to only a few times that determined by the Fried parameter r_0. Clearly, the larger the simulation band to full band ratio, the more accurate the simulated results will be.

On the one hand, the apertures of upcoming and future astronomical telescopes are often of the same order or even larger than the typical median outer scale sizes of about 20 to 25 m.[12] On the other hand, wavefront sensing and compensation technologies are fast progressing that Nyquist sampling at r_0 scales even for large aperture telescopes are becoming quite possible. Thus, atmospheric turbulence simulations have to deal with a wide range in a multi-dimensional parameter space.

For working with very small apertures relative to the outer scale, it may appear that we need to simulate only a relatively small screen size. But cutting out small apertures from a larger screen introduces deviation from phase structure function due to misrepresentation of low-frequency components present in the small screen power spectrum.

In this paper, we present an approach and a corresponding algorithm to deal with phase screen simulations for a wide range of G/L_0 ratios, using the FFT-based method.[13] Our technique builds upon the modified subharmonic approach of Johansson and Gavel[3] and is inspired by Jingsong Xiang's work.[7] It works well for space- and time-invariant, zero intermittency atmospheric turbulence. Section 2 explains how to obtain phase autocorrealtion matrix using phase power spectrum, Sec. 3 presents the algorithm part to compensate for the remaining error in phase structure function calculation, Sec. 4 steps through the implementation of the algorithm with the help of a flow chart, Sec. 5 covers the validation of the technique using results from simulated phase screens, and Sec. 6 provides the concluding remarks.

2 Obtaining Phase Autocorrelation Matrix Using Phase Power Spectrum

The 2D phase structure function and phase autocorrelation matrix are related as follows:[14]

$$D_\phi(m,n) = 2(B_\phi(0,0) - B_\phi(m,n)),$$ (1)

where $B_\phi(m,n)$ is the phase autocorrelation matrix and (m,n) is the coordinates along x and y axis. The 2D phase autocorrelation matrices for the FFT-based phase screen and the modified subharmonic method by Johansson and Gavel[3] are represented as follows.

$$B_\phi^{FFT}(m,n) = \sum_{m'=-N_x/2}^{N_x/2-1} \sum_{n'=-N_y/2}^{N_y/2-1} f_{FFT}^2(m',n') e^{i2\pi\left(\frac{m'm}{N_x}+\frac{n'n}{N_y}\right)},$$ (2)

$$B_\phi^{SUB}(m,n) = \sum_{p=1}^{N_p} \sum_{m'=-3}^{2} \sum_{n'=-3}^{2} f_{SUB}^2(m',n') e^{i2\pi3^{-p}\left(\frac{(m'+0.5)m}{N_x}+\frac{(n'+0.5)n}{N_y}\right)},$$ (3)

where $f_{FFT}^2(m',n')$ and $f_{SUB}^2(m',n')$ are the von-Kármán spectrum and subharmonic power spectrum as explained by Johansson and Gavel. (N_x, N_y) are sample points, p is the p'th subharmonic, and N_p is the total number of subharmonics. Set $f_{FFT} = 0$, for $(m',n') = (0,0)$ and $f_{SUB} = 0$, for $(m',n') = (-1,0)$ and $(0,-1)$ as originally proposed by Lane et al.[10] There will be an overlap between subharmonic energy sample and secondary lobes from first sample of high-frequency spectrum or harmonic sample during subharmonic addition. Earlier this leakage of energy has been dealt using patch normalization factor, where first patch of high-frequency spectrum is weighted by 0.707 for $(m',n') = (\pm1,0)$ and $(m',n') = (0,\pm1)$ and 0.866 for $(m',n') = (\pm1,\pm1)$ in the original method of Johansson and Gavel.[3] Similarly, the original method of Lane et al.,[10] Sedmak[6] proposed the corresponding weights to be 0.935 and 0.998, respectively. Our simulations show that these weights do not fit perfectly for different G/L_0 ratios and hence need to be tuned on a case by case basis. We have made our approach independent from these weights assignments. The weight factor has been set equal to 1 in our approach. Section 3 explains this approach in detail.

The 2D phase autocorrelation matrix after compensating with subharmonics is represented as

$$B_\phi(m,n) = B_\phi^{FFT}(m,n) + B_\phi^{SUB}(m,n).$$ (4)

3 Algorithm to Compensate for Residual Error in Phase Structure Function

To calculate the remaining error in the final $B_\phi(m,n)$, Eq. (4) is converted to phase structure matrix $D_\phi(m,n)$ with the help of Eq. (1) with the assumption that $B_\phi^{FFT}(0,0)$ and $B_\phi^{SUB}(0,0)$ are zero because we are not concerned about the piston component. This gives the following equation

$$D_{error}(m,n) = D_{theory}(m,n) - D_\phi(m,n),$$ (5)

where $D_{theory}(m,n)$ is the well-known theoretical von-Kármán phase structure matrix,[3] given as follows:

$$D_{theory}(r) = 6.16 r_0^{5/3}\left[0.6(L_0/2\pi)^{5/3} - \frac{(rL_0/4\pi)^{5/6}}{\gamma(11/6)} K_{5/6}(2\pi r/L_0)\right],$$ (6)

where $r^2 = (m\Delta)^2 + (n\Delta)^2$, $\Delta = G/N$.

We need to compensate D_ϕ so that D_{error} is minimized. However, simply adding error correction terms in the D_ϕ matrix directly would only introduce further error into the system while

taking the Fourier transform. This is because any matrix or curve in general will have higher order moments. Thus, if we take the Fourier transform of the adjusted equation, the resultant curve will have completely different moments and hence power spectrum. This is because the transition between two steps in the error matrix will not be smooth, which introduces additional errors due to Gibb's phenomena such as overshoots. Just curve fitting with any function does not satisfy the additional requirement of leaving the power spectrum unaffected by the process. What we really need is to introduce a smoothening operator such as a Gaussian function in the phase autocorrelation matrix, which exactly compensates for D_{error}.

For that we have developed an iterative algorithm (see the flow chart shown in Fig. 4) and implemented it in Matlab. The algorithm looks for the perfect Gaussian curve that minimizes the D_{error} matrix. We use Matlab cftool to initially determine the correct 1D Gaussian matrix and later convert it into a 2D matrix by exploiting the fact that $B_\phi(r)$, $B_{\text{theory}}(r)$, and $B_{\text{SUB}}(r)$ all are dependent on r only and hence are center symmetric functions. We call the fitted Gaussian phase structure matrix D_{gauss} and the corresponding Gaussian phase autocorrelaiton matrix B_{gauss} [using Eq. (1)].

The final equation for B_{tot} can then be written as

$$B_{\text{tot}}(m, n) = B_\phi^{FFT}(m, n) + B_\phi^{SUB}(m, n) - D_{\text{gauss}}(m, n)/2. \tag{7}$$

Here, we have used $B_{\text{gauss}} = -D_{\text{gauss}}/2$ from Eq. (1). A look at the power spectrum of $B_{\text{tot}}(m, n)$ in Fig. 2 shows that it contains negative terms[7] for the case of $G/L_0 < 1$. Directly putting those frequency terms equal to zero leads to a loss in the energy spectrum. Hence, $B_{\text{tot}}(m, n)$ matrix needs to be preprocessed to eliminate most of these negative values in the power spectrum. Over small frequencies, piston and tip/tilt components account for most of these high magnitude negative elements. Therefore, we first extract the piston and tip/tilt components from the phase autocorrelation matrix B_{tot}. The tip/tilt component from phase autocorrelation matrix is given as[7]

$$B_{\text{tilt}}(r) = B_{\text{tilt}}(0) - r^2 \sigma_{\text{tilt}}^2/2, \tag{8}$$

where σ_{tilt}^2 is the variance of the random tilt angle in the x or y directions and given as follows:[7]
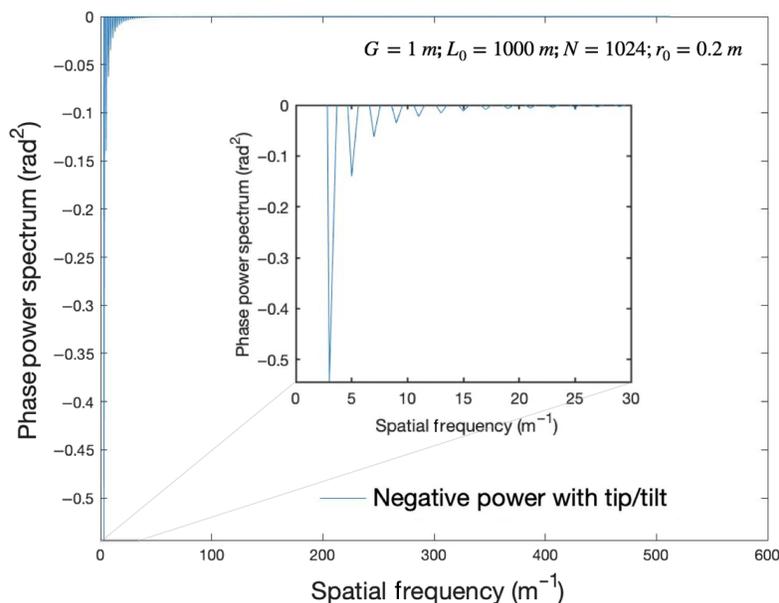
Fig. 2 Negative power spectrum values for small G/L_0 ratios.
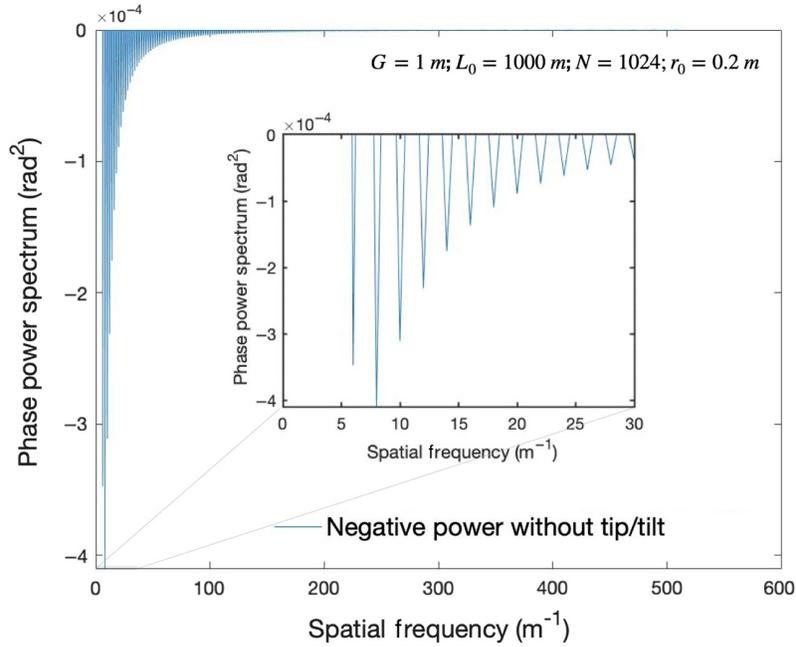
Fig. 3 Residual negative power spectrum values after removing tip/tilt from B_{tot} for small G/L_0 ratios.

$$\sigma_{\text{tilt}}^2 = \frac{B_{\text{tot}}(G/2 + \Delta) - B_{\text{tot}}(G/2)}{\Delta(G - \Delta)/2}. \tag{9}$$

After setting, $B_{\text{tilt}}(0) = 0$, the remaining phase autocorrelation matrix is given as follows:

$$B_{\text{high}}(r) = B_{\text{tot}}(r) - B_{\text{tilt}}(r). \tag{10}$$

The power spectra f_{high}^2 and f_{tilt}^2 of the phase autocorrelation matrices $B_{\text{high}}(r)$ and $B_{\text{tilt}}(r)$ are obtained by standard Fourier transformation. Figure 3, shows the remaining negative power elements present in the power spectrum of B_{high} matrix. In comparison to Fig. 2, the largest negative power contributions fall by factor of three order of magnitude. Now, we set the negative values in f_{high}^2 equal to zero by hand. The new error matrix is given as

$$B_{\text{high}}^{err}(r) = B_{\text{high}}'(r) - B_{\text{high}}(r), \tag{11}$$

where $B_{\text{high}}'(r)$ is the phase autocorrelation matrix obtained after setting the negative elements in f_{high}^2 to zero. The residual error that is present in the high-frequency region can then be reduced with the help of a Gaussian smoothing operator, using Matlab fmincon tool. The high-frequency compensated matrix is given as

$$B_{\text{high}}^{\text{comp}}(r) = B_{\text{high}}(r) - H_{\text{high}}^{\text{comp}}(r)B_{\text{high}}^{\text{err}}(r), \tag{12}$$

where $H_{\text{high}}^{\text{comp}}(r)$ is the smoothening operator, multiplied with error matrix to reduce the high frequency errors. fmincon gives the optimized parameter for smoothening operator by calculating the final error in the $D_\phi(r)$ matrix with respect to $D_{\text{theory}}(r)$.

4 Implementation of the Compensation Algorithm

In this section, we explain the error compensation algorithm with the help of the flow chart shown in Fig. 4. Brief explanations of each of the steps from L_1 to L_{12} are given below.
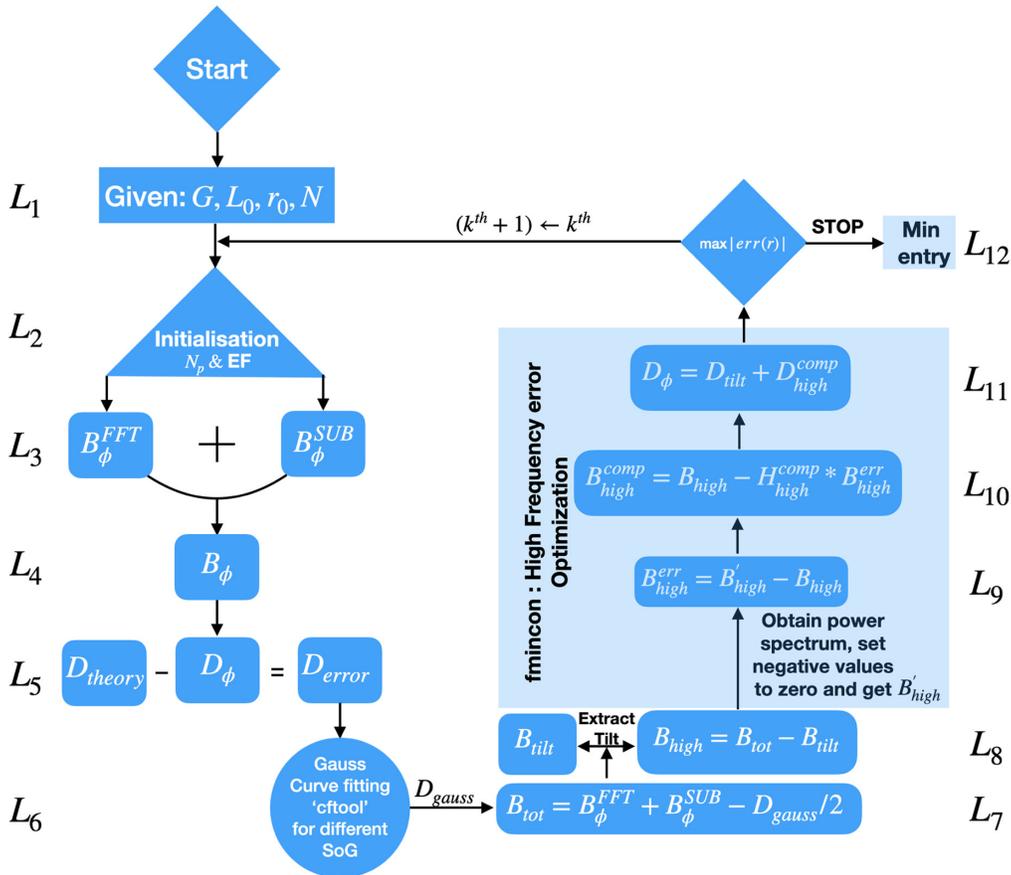
Fig. 4 Flow chart for error compensation.

L_1: Input screen size G, outer scale size L_0, Fried parameter r_0, and number of samples N.

L_2: Initialize the algorithm with N_p, the total number of subharmonics, and extrapolation factor EF, both ranging from 1 to 10. The EF factor is relevant while performing curve fitting, e.g., EF $= 2$ means curve fitting will work from 3 to $N/2$ points and later that curve will be extrapolated from 1 to $N/2$ points (example shown in Sec. 4.1 and Fig. 6)

L_3: Obtain B_ϕ^{FFT} and B_ϕ^{SUB} based upon L_1 and L_2 parameters.

L_4: Add the matrices that were calculated in L_3 layer, call that B_ϕ.

L_5: Obtain D_ϕ from B_ϕ using Eq. (1) and also produce D_{theory} matrix based upon L_1 and L_2. Then obtain error matrix D_{error} using Eq. (5).

L_6: Obtain 1D array from D_{error} matrix from the center and give as input to the curve fitting tool cftool, which works on 1D data. The output of the tool will be a best fitted curve in terms of sum of Gaussian's (SOG), called D_{gauss}.

L_7: The final expression for B_{tot} is shown Eq. (7).

L_8: Extract tilt component from B_{tot} matrix using Eqs. (8) and (9).

fmincon: High Frequency error Optimization

$L_9 - L_{11}$: Obtain error matrix $B_{\text{high}}^{\text{err}}$ after setting negative elements in the power spectrum to zero. Compensate for high frequency error by multiplying error matrix with smoothening operator-SOG. Calculate the maximum remaining error in structure function matrix relative to the D_{theory} matrix. Thus fmincon will give parameters for the smoothing operator, which gives the lowest possible residual error.

L_{12}: Update the entry for N_p and EF to next value, evaluation from $L_2 - L_{12}$ would go in loop, and maximum relative error (MRE) value stored in vector form. At the end of the

iterations, min entry will get extracted out from stored vector and accepted for final analysis.

Table 1 shows the result of curve fitting using cftool for different cases of G/L_0 and N, which demonstrates that the Gaussian error matrix can compensate for a wide range of G/L_0 ratios and under different sampling constraints.

4.1 Example

To illustrate the robustness of the above algorithm, we have taken an example with $G = 80$ m, say for a large future telescope, $N = 256$, and median value of $L_0 = 20$ m.

The output from the above algorithm corresponding to minimum error entry as in (step L_{12}), has been plotted against $EF = 5$ and $N_p = 8$. Figure 5 shows a 3D rendering of D_{error} matrix with a maximum separation of up to 40 m, corresponding to Eq. (4).

Figure 6 shows 1D D_{error} matrix (radial section from 3D D_{error} matrix) along with 1D fitted curve D_{gauss} including the extrapolated part, for a maximum separation of up to 40 m.

Lastly, MRE values are stored against 500 entries corresponding to N_p ranging from 1 to 10, EF ranging from 1 to 10 and SoG ranging from 2 to 6 after performing cftool fitting. This has been arranged in descending order and shown in Fig. 7, which illustrates a large set of iterations where errors are <1% and entry with minimum MRE has been picked up. Typical time required to perform each iteration for this case is ≈4.9 s on 2.3 GHz quad-core Intel Core i5 Macbook Pro 2018 model.

Table 1 Result of curve fitting against Gaussian function for different cases of G/L_0 and N in terms of MRE for fixed $r_0 = 0.2$ m.

G	L_0	N_p	EF	SOG	MRE (%)	N_p	EF	SOG	MRE (%)	N_p	EF	SOG	MRE (%)	N_p	EF	SOG	MRE (%)
		N = 128				N = 256				N = 512				N = 1024			
1	20	2	1	3	0.25	2	2	4	0.61	2	2	3	0.36	2	1	4	1
5	20	1	1	5	0.19	6	3	5	0.23	4	2	3	0.27	7	1	3	1
10	20	9	5	4	0.25	9	1	5	0.29	1	1	3	0.26	7	3	6	0.35
20	20	3	1	4	0.53	10	1	6	0.4	4	5	5	0.97	3	5	3	0.22
40	20	5	0	4	0.95	8	0	6	0.75	8	0	3	1	7	1	3	2.99
60	20	5	0	4	0.17	3	2	6	0.42	9	8	5	0.41	3	1	5	0.94
80	20	8	1	4	0.24	7	5	3	0.17	5	6	3	0.32	6	4	3	0.3
100	20	8	1	4	0.24	7	5	3	0.17	5	6	3	0.32	6	4	3	0.3
1	10	3	1	4	0.53	10	1	6	0.4	4	5	5	0.97	3	5	3	0.22
1	100	3	1	6	0.21	4	1	4	0.49	3	6	5	0.96	3	6	5	0.25
1	1000	5	0	4	0.95	8	0	6	0.75	8	0	3	1	7	1	3	2.99
10	100	2	0	6	0.28	3	0	4	0.22	8	1	4	0.30	7	0	3	0.27
10	1000	5	0	4	0.17	3	2	6	0.42	9	8	5	0.41	3	1	5	0.94
100	100	3	0	6	0.21	2	3	5	0.21	7	0	3	0.26	1	1	3	0.26
100	1000	8	1	4	0.24	7	5	3	0.17	5	6	3	0.32	6	4	3	0.3
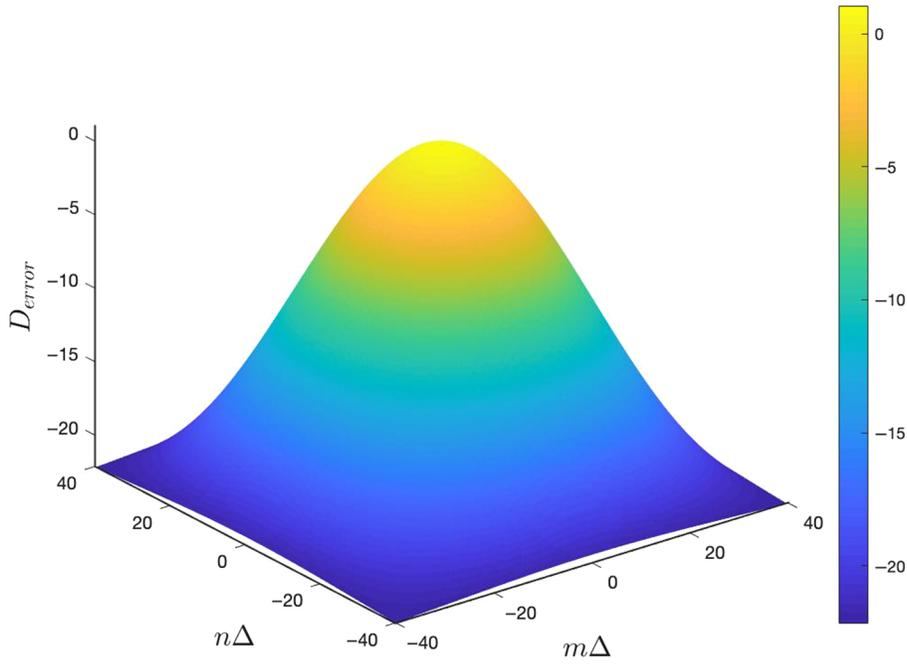
Fig. 5 3D D_{error} matrix for case $G = 80$ m, $L_0 = 20$ m, $r_0 = 0.2$ m, and $N_x = N_y = 256$.
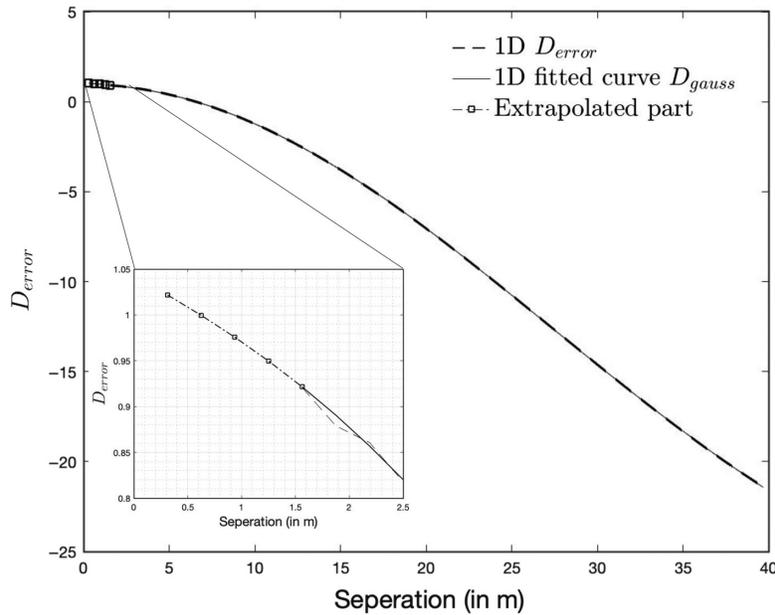
Fig. 6 1D D_{error} matrix fitted against D_{gauss} matrix, along with an extrapolated part of the curve. Here $G = 80$ m, $L_0 = 20$ m, $r_0 = 0.2$ m, and $N_x = N_y = 256$.

5 Validation via Phase Structure Function Calculated from Simulated Phase Screen

To obtain the phase screen $\phi(m, n)$ from the power spectrum, the following relation is used:[7]

$$\phi(m\Delta, n\Delta) = \sum_{m'=-N/2}^{N/2-1} \sum_{n'=-N/2}^{N/2-1} [R_a(m', n')$$
$$+ iR_b(m', n')] f(m'\Delta', n'\Delta') \exp[i2\pi(m'm + n'n)/N], \qquad (13)$$
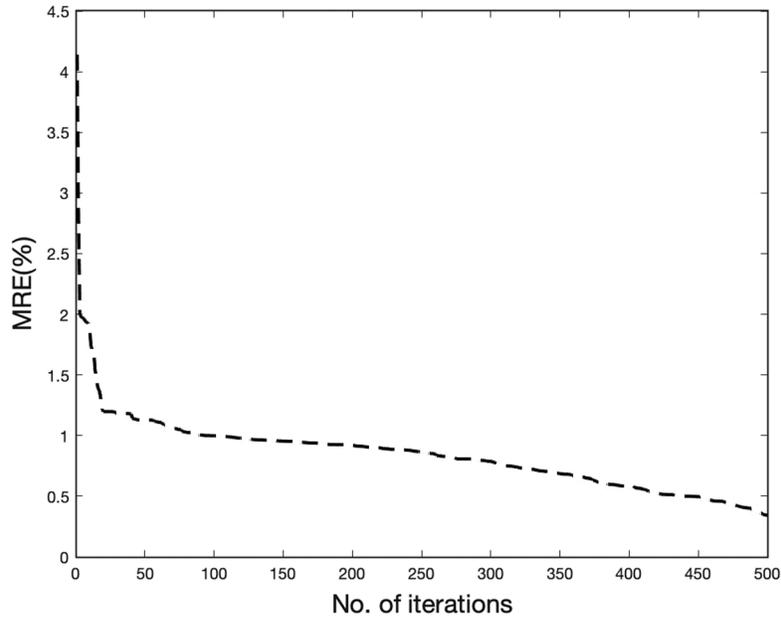
Fig. 7 Maximum relative error MRE with the maximum number of iterations for $G = 80$ m, $L_0 = 20$ m, $r_0 = 0.2$ m, and $N_x = N_y = 256$.

where $R_a(m', n')$ and $R_b(m', n')$ are zero-mean and unity-variance Gaussian random number generator. We get ϕ_{high} and ϕ_{tilt}, by replacing f with f_{high} and f_{tilt}, which are square roots of the power spectrum corresponding to autocorrelation matrix $B_{\text{high}}^{\text{comp}}$ and B_{tilt}, respectively.

For validation, we consider scenarios of apertures up to 40 m, i.e., $G = 80$ m, at a median $L_0 = 20$ m for two different sampling levels $N = 256$ and 512. The phase structure function, defined as an ensemble average of differences of phases at various separation,[14] has been averaged over 100 K independent frames. The relative error in phase structure function is calculated as follows:
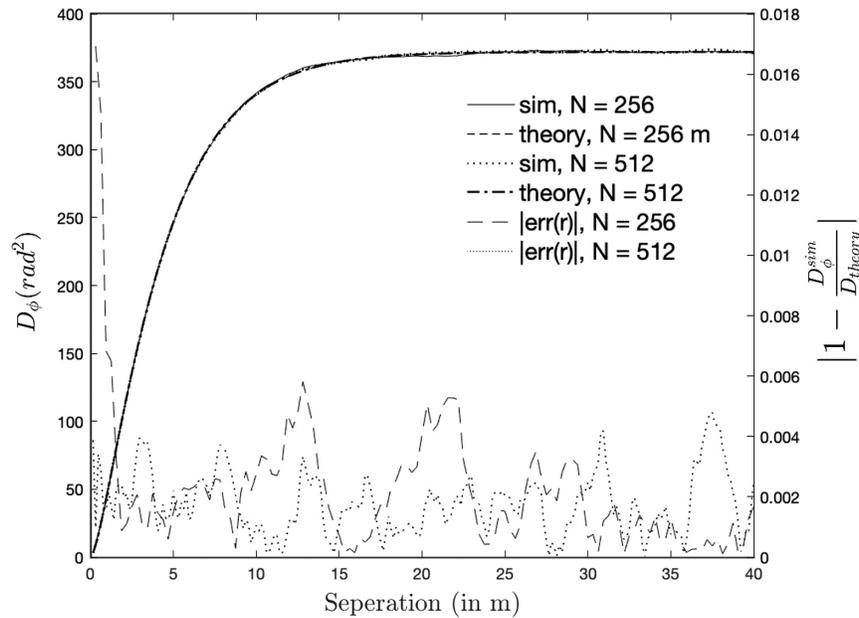
Fig. 8 Left: Compares simulated structure function with respect to theoretical structure function for maximum separation of $G/2$ for two different cases $N = 256$ and 512, for fixed $G = 80$ m, $r_0 = 0.2$ m, and $L_0 = 20$ m. Right: Calculates the magnitude of relative error in simulated structure function for maximum separation of $G/2$, for both cases.

$$err(r) = \frac{D_\phi^{\text{sim}}(r) - D_{\text{theory}}(r)}{D_{\text{theory}}(r)}. \tag{14}$$

Here, $D_\phi^{\text{sim}}(r)$ is the phase structure function from the simulated phase screen. The magnitude of the peak relative error $\max(|err(r)|)$ is $<1.6\%$ for $N = 256$ and $<0.5\%$ for $N = 512$ as shown in Fig. 8.

We also illustrate the performance (shown in Fig. 9) with parameters $G = 1$ m, $L_0 = 100$ m, and 1000 m, $N = 128$, $r_0 = 0.2$ m, which cover the extreme cases (very low G/L_0 ratios) which leads to the maximum error in the simulation. The magnitude of the peak relative error $\max(|err(r)|)$ is $<1.6\%$ for $L_0 = 100$ m and $<1.8\%$ for $L_0 = 1000$ m. Figure 10 shows one realization of the corresponding phase screen plots for $L_0 = 100$ m.

Figure 11 contains results of magnitude of the peak relative error in $D_\phi^{\text{sim}}(r)$ for the case of different sampling points $N = 128/256/512/1024$, for L_0 ranges up to 1024 m, $r_0 = 0.2$ m and $G = 2$ m. Similarly, Fig. 12 contains results of magnitude of the peak relative error in $D_\phi^{\text{sim}}(r)$ for
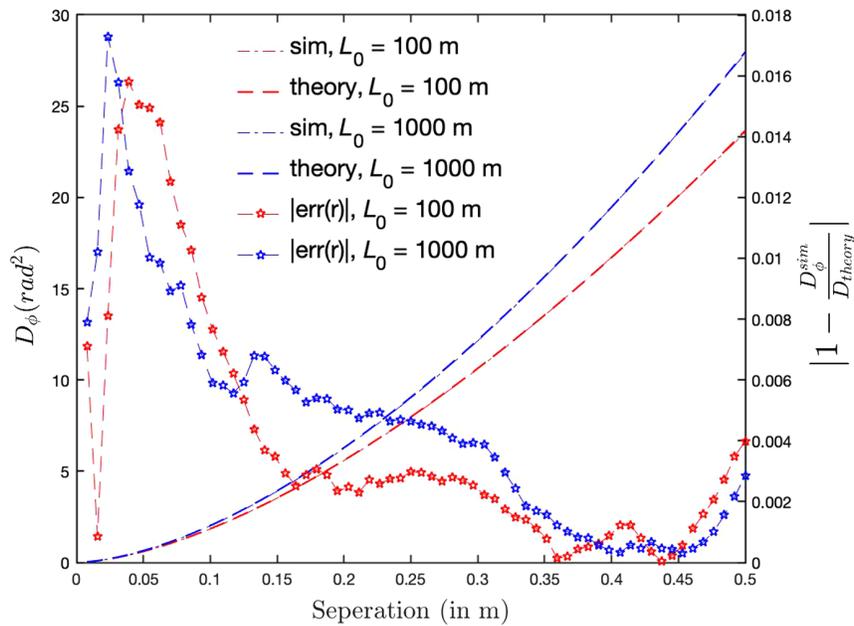
Fig. 9 (a) Compares simulated structure function with respect to theoretical structure function for maximum separation of $G/2$ for two different cases $L_0 = 100$ and 1000 m, for fixed $G = 1$ m, $r_0 = 0.2$ m, and $N = 128$. (b) Calculates the magnitude of relative error in simulated structure function for maximum separation of $G/2$, for both cases.
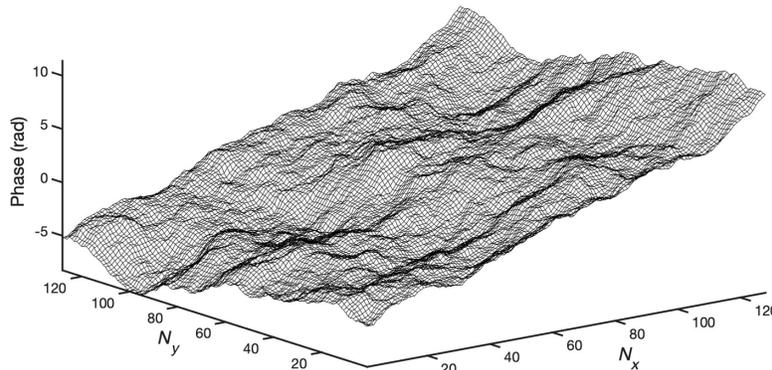
Fig. 10 Phase screen for case $G = 1$ m, $L_0 = 100$ m, $r_0 = 0.2$ m, and $N_x = N_y = 128$.
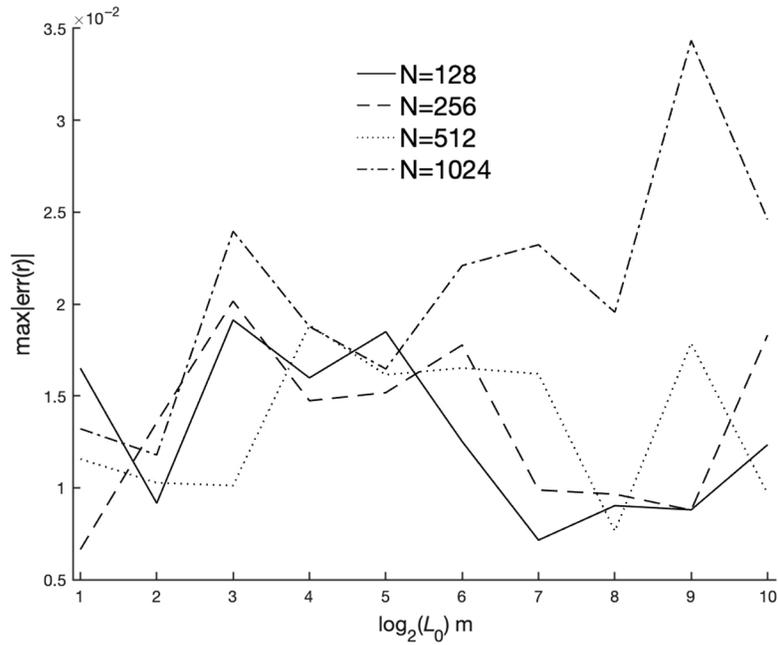
Fig. 11 The magnitude of the peak relative error for $N = 128, 256, 512$, and 1024 for L_0 ranges up to 1024 m. Here $G = 2$ m and $r_0 = 0.2$ m.
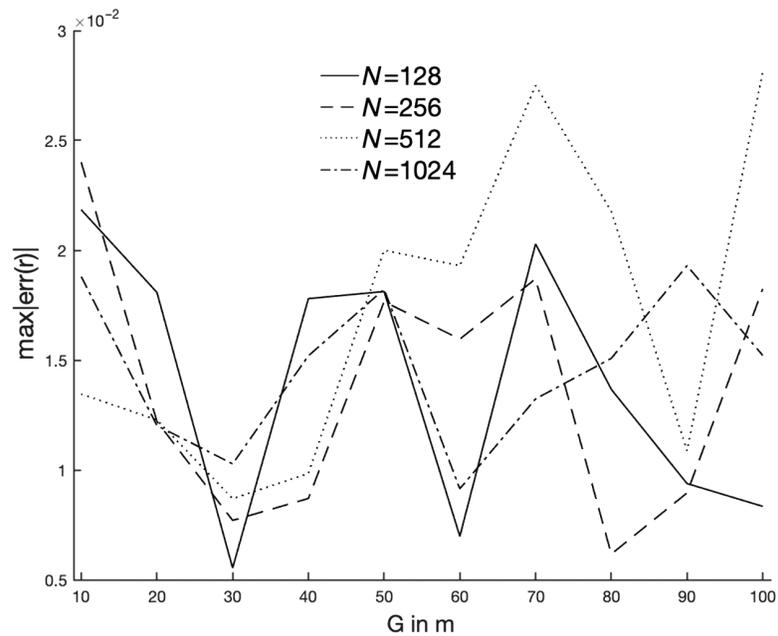
Fig. 12 The magnitude of the peak relative error for $N = 128, 256, 512$, and 1024 for screen size up to 100 m. Here, $L_0 = 25$ m and $r_0 = 0.2$ m.

the case of different sampling points $N = 128/256/512/1024$, for G ranges up to 100 m, $r_0 = 0.2$ m, and $L_0 = 25$ m.

There are some outliers that have a high residual error as shown in Figs. 11 and 12, because we have not set the phase autocorrelation matrix to zero for $r > G/2$. The reason for this stems from the non-zero value of $B_{high}(r)$ at $r > G/2$, where $B_{high}(r)$ is formed from the removal of piston and tilt from $B_{tot}(r)$. This can be resolved using a better smoothening operator, which we can multiply with B_{high} so that it falls to zero progressively and not sharply. This can provide further improvement in the compensation.

6 Conclusion

In this paper, we put forward a new method to compensate for the residual error in both the low and/or high-frequency region of FFT simulated phase screens that remain even after compensating with the modified subharmonic method. This method provides accurate phase screen structure for even G/L_0 ratios as small as 1/1000 plus screen sizes as large as 100 m. No patch normalization factor is needed, no need to calculate subharmonic weight coefficient[10] and weights to compensate for high-frequency components, as done by Sedmak.[6] While adequately large G/L_0 ratios may be the natural choice for modern large telescopes, simulations that deal with applications such as laser beam propagation through turbulent atmospheres would tend to have very small G/L_0 ratios. The method we propose is independent of the G/L_0 ratio choice. However, we emphasize that properly sampling r_0 and the high-frequency phase spectrum forces N to be at least larger than $(2G/r_0)$ and preferably up to the inner scale limit $(2G/l_0)$. Currently, we have demonstrated this technique for only circular screens. We have used a GPU processor with total number of 128 cores, such that each iteration runs independently on each core. We have fixed the number of iterations to 500, although increasing this will lead to improvement of errors in some cases. Each core takes about ∼0.06, ∼0.1, ∼0.36, and ∼1.1 min for sampling sizes of 128, 256, 512, and 1024. On the above GPU system, this translates to total computing times for error minimization of about ∼0.2, ∼0.5, ∼1.25, and ∼4 min for sampling sizes of 128, 256, 512, and 1024, respectively. Once the coefficients are determined, generating multiple phase screen realizations from the corresponding power spectrum takes a few milliseconds at most on 2.3 GHz quad-core Intel Core i5 Macbook Pro 2018 model. Then it takes <1 min to ∼10 min for averaging over 100k phase screens, for sampling sizes ranging from 128 to 1024.

The uniqueness of our approach is its ability to deal with any G/L_0 ratio within a very broad range, in an automated iterative process with little human intervention needed for tuning of parameters. Any standard FFT-based approach (say Sedmak's[6] compensated approach) for a given computer platform is computationally fast, only if we already have determined proper measures of the various compensating components such as the patch normalization factor, subharmonic weights, and high-frequency weights. Typically, determining these compensations is where the difficulty is due to mathematical complexity, algorithmic limitations, and/or computational power requirements. Our algorithm accomplishes the determination of the required compensation in very little time, with fairly reasonable computational power while at the same time keeping the residual errors competitively low using an appropriate compensator. Other existing FFT based approaches have limitations in their operable G/L_0 range. For example, Xiang et al.[7] offer a very computationally fast approach but does not apply subharmonic compensation. Zhang et al.[9] does not consider compensation for the high-frequency error, thus leaving a residual error of more than 100% in the high-frequency region. Sedmak's[6] approach needs the determination of accurate subharmonic weights for different G/L_0 ratios. The accuracy of our method from low-frequency to the high-frequency range is between 0.5% and 3% for G/L_0 as low as 1/1000 and screen size up to 100 m in diameter.

Acknowledgments

References

1. N. A. Roddier, "Atmospheric wavefront simulation and Zernike polynomials," *Proc. SPIE* **1237**, 668–679 (1990).
2. B. J. Herman and L. A. Strugala, "Method for inclusion of low-frequency contributions in numerical representation of atmospheric turbulence," *Proc. SPIE* **1221**, 183–192 (1990).

3. E. M. Johansson and D. T. Gavel, "Simulation of stellar speckle imaging," *Proc. SPIE* **2200**, 372–383 (1994).
4. G. Sedmak, "Performance analysis of and compensation for aspect-ratio effects of fast-Fourier-transform-based simulations of large atmospheric wave fronts," *Appl. Opt.* **37**(21), 4605–4613 (1998).
5. B. L. McGlamery, "Computer simulation studies of compensation of turbulence degraded images," *Proc. SPIE* **0074**, 225–233 (1976).
6. G. Sedmak, "Implementation of fast-Fourier-transform-based simulations of extra-large atmospheric phase and scintillation screens," *Appl. Opt.* **43**(23), 4527–4538 (2004).
7. J. Xiang, "Fast and accurate simulation of the turbulent phase screen using fast Fourier transform," *Opt. Eng.* **53**(1), 016110 (2014).
8. J. Xiang, "Accurate compensation of the low-frequency components for the FFT-based turbulent phase screen," *Opt. Express* **20**(1), 681–687 (2012).
9. D. Zhang et al., "Accurate simulation of turbulent phase screen using optimization method," *Optik* **178**, 1023–1028 (2019).
10. R. Lane et al., "Simulation of a Kolmogorov phase screen," *Waves Random Media* **2**(3), 209–224 (1992).
11. G. Sedmak, private communication (2019).
12. A. Ziad, "Review of the outer scale of the atmospheric turbulence," *Proc. SPIE* **9909**, 99091K (2016).
13. S. Chhabra et al., "Gaussian phase autocorrelation as an accurate compensator for FFT-based atmospheric phase screen simulations," *Proc. SPIE* **11448**, 114487U (2020).
14. F. Roddier, *Adaptive Optics in Astronomy*, Cambridge University Press (1999).

Sorabh Chhabra is a PhD student at Inter University Center for Astronomy and Astrophysics (IUCAA), Pune. He received his B. Tech degree in electronics and communication from Delhi Technological University (formally Delhi College of Engineering) in 2016 and joined for his PhD in the same year in the Instrumentation Department at IUCAA.

Biographies of the other authors are not available.