



**HAL**  
open science

# Accuracy and Computational Efficiency of Various Methods for Atmospheric Turbulence Structure Function Computation

Deborah Malone, Nicholas Devaney

► **To cite this version:**

Deborah Malone, Nicholas Devaney. Accuracy and Computational Efficiency of Various Methods for Atmospheric Turbulence Structure Function Computation. Adaptive Optics for Extremely Large Telescopes 7th Edition, ONERA, Jun 2023, Avignon, France. 10.13009/AO4ELT7-2023-085 . hal-04402826

**HAL Id: hal-04402826**

**<https://hal.science/hal-04402826>**

Submitted on 18 Jan 2024

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



# Accuracy and Computational Efficiency of Various Methods for Atmospheric Turbulence Structure Function Computation

Deborah Malone<sup>a</sup> and Nicholas Devaney<sup>a</sup>

<sup>a</sup>Applied Optics, University of Galway, Ireland

## ABSTRACT

Structure functions are used to characterise spatial variations in turbulent quantities such as air temperature, density, refractive index and wavefront phase. They are often used to verify numerically generated phase screens for use in numerical simulations or with, for example, spatial light modulators. The usual method of calculating the structure function of such a screen in principle requires the comparison of each pixel with every other pixel, a computationally complex and time-consuming process. This paper will compare alternative methods for calculating the structure function of numerically generated atmospheric turbulence. Specifically, it evaluates the efficiency and accuracy of the methods and highlights the strengths and limitations of each. The study aims to help researchers select the appropriate method for their specific needs, choosing between faster computation, or higher accuracy.

**Keywords:** atmospheric turbulence, numerical simulations, structure functions, turbulence characterization

## 1. INTRODUCTION

Atmospheric turbulence primarily occurs in the lower part of the atmosphere, known as the planetary boundary layer (PBL), and refers to the small scale chaotic flow of air [3]. The turbulence in the PBL transfers heat, moisture, and momentum between the Earth's surface and the atmosphere. Since the atmosphere is a system in hydrostatic equilibrium, changes in temperature affect the density of the atmosphere in that region, and as a consequence, the refractive index also changes.

Changes in refractive index cause a difference in phase across any wavefront travelling through the atmosphere. The statistics of the phase changes can be described by the Kolmogorov theory of turbulence, which describes the atmosphere as a series of energy cascades from larger to smaller structures, known as eddies [4]. We can describe the average phase difference squared between any two points using a structure function, defined by Equation 1;

$$D_\phi(r) = \langle [\phi(r_1 + r) - \phi(r_1)]^2 \rangle \quad (1)$$

where  $\phi(r_1)$  is the phase at a point  $r_1$ , and  $\phi(r_1 + r)$  is the phase a distance  $r$  away. The angle brackets indicate the statistical expected value [5]. It can be shown that Kolmogorov turbulence gives rise to a power law given by Equation 2.

$$D_\phi(r) = 6.88 \left( \frac{r}{r_0} \right)^{5/3} \quad (2)$$

---

Further author information: (Send correspondence to D.M)

D.M: E-mail: [deborah.malone@universityofgalway.ie](mailto:deborah.malone@universityofgalway.ie)

N.D: E-mail: [nicholas.devaney@universityofgalway.ie](mailto:nicholas.devaney@universityofgalway.ie)

where  $r$  is the separation between two points, and  $r_0$  is the Fried parameter [1]. By measuring the structure function of phase screens, one can compare to the Kolmogorov power law to see how accurate the phase screen simulates real atmospheric turbulence.

## 2. EXPERIMENT SETUP

For the experiment outlined in this paper, phase screens were generated using the code developed by Johnston, Harding, and Lane, which uses a midpoint displacement method to increase the resolution of a simulated low-resolution phase screen [2]. The hardware setup of the computer used for timed comparisons is given in Table 1.

Table 1: Computer set-up for running calculations.

<b>Operating System</b>	Windows 11 Home, version 10.0.22621
<b>Processor</b>	Intel <sup>R</sup> Core <sup>TM</sup> i5-13600K (14 Cores, up to 5.10GHz)
<b>RAM</b>	32 GB DDR5 @ 6000 MHz
<b>MATLAB Version</b>	9.13.0.2126072 (R2022b) Update 3

The purpose of this experiment is to find fast ways of measuring the structure function of many phase screens when used either numerically for simulations, or with optical devices which can display them, such as Spatial Light Modulators (SLMs). Because of this, the experiments will use a reference phase screen diameter of 100 pixels for initial testing, and a diameter of 512 pixels for the final experiments, which is a commonly used resolution for off-the-shelf SLMs.

## 3. PERFORMANCE OF BASELINE STRUCTURE FUNCTION ALGORITHM

In this paper, we define the Baseline Structure Function (BSF) as the structure function which compares every pixel to every other pixel. It is the most computationally expensive, but also the one which will give the most accurate result. The number of calculations in a phase screen goes according to Equation 3, where  $p$  is the number of pixels within the area used to calculate the structure function.

$$n_{calc} = \sum_{i=1}^{p-1} (p-i) \quad (3)$$

Since the function is defined over a circular aperture, a phase screen with a diameter of 512 pixels would contain 205887 pixels within the aperture (using  $\pi r^2$  where  $r = 512/2$ ), requiring  $2.1 \times 10^{10}$  calculations.

The first experiment uses phase screen sizes from 10 to 200 pixels in diameter in steps of 10, and measures the time taken for the structure function to be calculated using the BSF approach. The resulting Time vs Screen Diameter graph can be found in Figure 1. Fitting a power curve to the results gives the following equation;

$$T(x) = (5.991 \times 10^{-9})x^{5.344} \quad (4)$$

where  $x$  is the screen diameter in pixels, and  $T(x)$  is the time to perform the calculation. For a screen size of 512 pixels, this gives 20.9 days. This length of time is not practical, and so a faster method must be used to deal with large numbers of phase screens.

## 4. SPEEDING UP THE STRUCTURE FUNCTION ALGORITHM

The simplest way to speed up the structure function calculation is to reduce the number of pixel comparisons. However, we also consider whether the geometry of the positions of the comparison pixels has an effect on the resulting structure function. For the sake of this investigation, three different geometries are used. They are defined as grid, circular, and random. Examples of these geometries can be seen in Figure 2. In each case, every pixel within the defined structure function area (the circular aperture shown in white) is compared to a sub-set

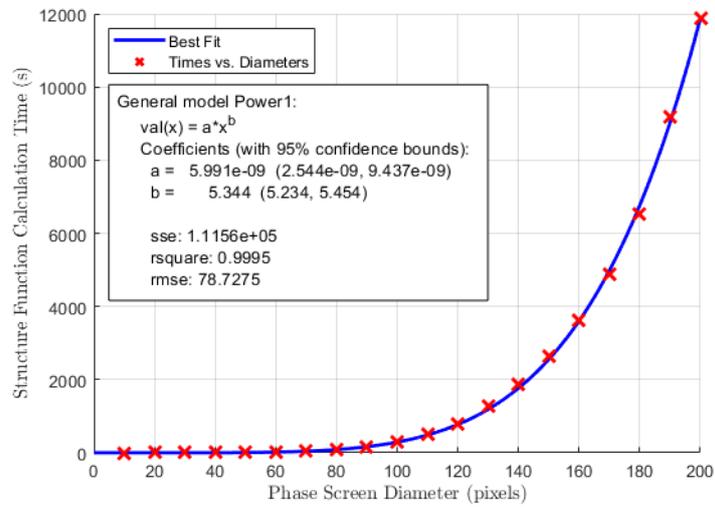
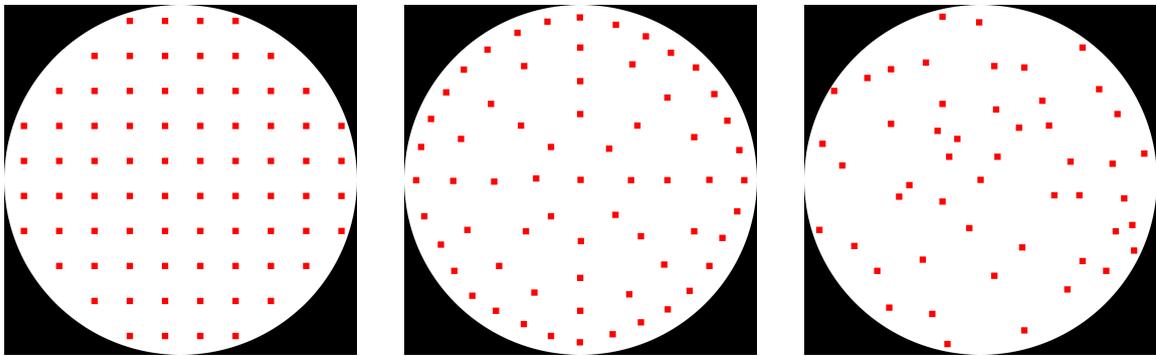


Figure 1: Time to complete the structure function of phase screens of diameters varying between 10 and 200.



(a) Grid layout – The pixels are arranged in a uniform grid throughout the defined area.  
 (b) Circle layout - The pixels are arranged in equally spaced concentric circles.  
 (c) Random layout - The positions of the pixels are chosen randomly.

Figure 2: The three layouts used for the investigation of the reduced pixel comparisons method. Each method will have approximately the same number of comparisons. The area over which the structure function is calculated is shown in white.

of pixels (shown in red). Each method has approximately the same number of comparisons, the only difference being the position of the pixels which are compared.

For these comparisons, the variable  $n$  is introduced, defined as the number of columns and rows skipped between comparison pixels in the grid algorithm. This means for every value of  $n$ , one comparison is made in each  $n \times n$  grid. For example, for  $n = 2$ , one comparison is made in each  $2 \times 2$  grid, or  $1/4$  of the total number of pixels are compared. The number of comparisons,  $C(n)$ , therefore, is given by Equation 5;

$$C(n) = \frac{p}{n^2} \quad (5)$$

where  $C(n)$  is the number of comparisons,  $p$  is the number of pixels contained within the aperture, and  $n$  is the number of columns and rows to skip between comparisons. While this value is based on the grid algorithm, the same number of comparison pixels is used for the circular and random algorithms to keep the number of computations the same.

#### 4.1 Preliminary Testing

First, we wish to compare the speed and the accuracy of the algorithms on a smaller data set. To achieve this, a single  $100 \times 100$  pixel phase screen with a  $D/r_0$  value of 10 is generated, and its structure function calculated using the baseline algorithm ( $n = 1$ ). This is shown in Figure 3. While structure functions are expected to fit Equation 2, this law is an average over many realisations and a single screen is not expected to match the law exactly. This can be seen by comparing the fit result in Figure 3b with Equation 2. With a diameter of 100 and a  $D/r_0$  value of 10, the values of  $a$  and  $b$  if the structure function fit the theoretical exactly would be 0.688 and 1.6667 respectively. This is not the case, as the measurement is based on only one screen.

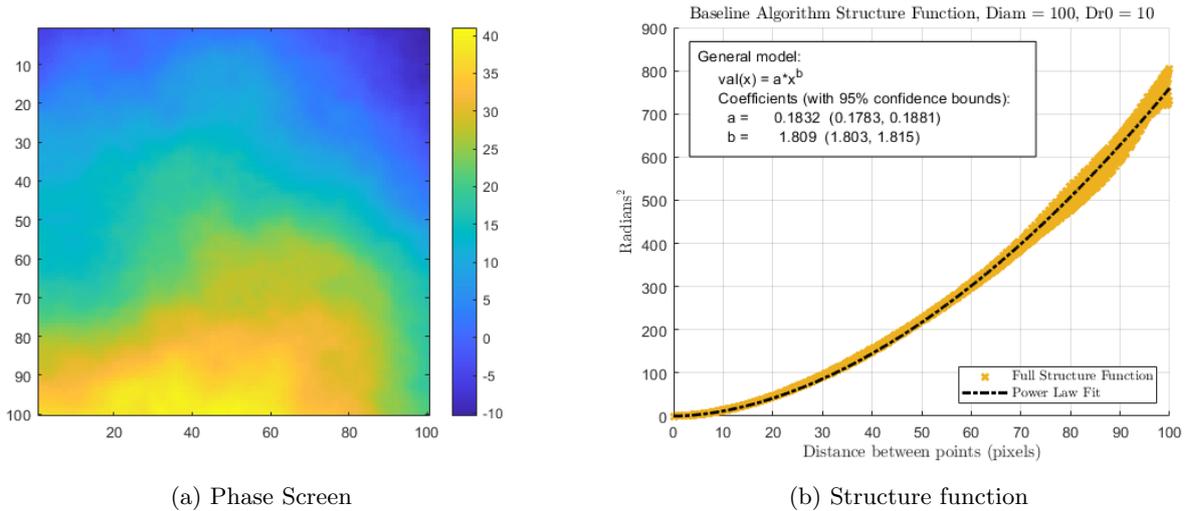


Figure 3:  $100 \times 100$  Phase Screen with resulting structure function produced using the baseline algorithm, which compares every pixel to every other pixel. The value of  $D/r_0$  is 10.

Instead of comparing results to the Kolmogorov power law (Eqn. 2) for this smaller dataset, the results from the fast algorithms will be compared only to the resulting structure function from the baseline algorithm.

##### 4.1.1 Computation time

The time to calculate a structure function with the three different algorithms on a single phase screen with different values of  $n$  can be found in Figure 4. The speed of the three algorithms are comparable for all values of  $n$  considered here, which shows that approximately the same number of comparisons are made for each algorithm, as required.

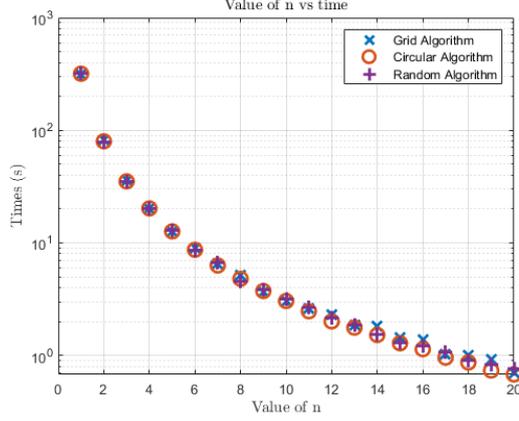


Figure 4: Comparison of the time to complete and for the grid, circular, and random algorithms in calculating a structure function for a single  $100 \times 100$  phase screen. Note: y-axis is log scale.

#### 4.1.2 Accuracy of multiple screens for different values of n

As the Structure Function is defined as the statistical average, it must be used on a larger data set. For this, 100 phase screens created with a  $D/r_0$  value of 10 and dimensions of  $100 \times 100$  pixels are created. Each algorithm is tested on the phase screens with values of  $n$  equal to 5, 10, 15, and 20. The resulting structure functions are shown in Figure 5.

In figures 5a and 5b, the resulting structure functions for each algorithm closely match the reference curve. In Figure 5c, we can see the Grid and Random algorithms start to deviate from the reference, while the Circular one remains very close. In Figure 5d, all the algorithms are significantly deviating, suggesting that this value of  $n = 20$  is too high for this method. In Figure 6, we can see that the time to compute all 100 phase screens remained consistent for each algorithm.

It was noted during these measurements that the random algorithm gave different results for the structure functions every time. While this was expected, it meant that the calculations would not be repeatable, and this algorithm was discarded in favour of the other two.

#### 4.1.3 Errors

Using the Grid and Circular algorithms, the errors of the structure functions were calculated. The errors on these graphs are calculated using the Standard Error (SE), given by Equation 6.

$$SE = \frac{\sigma}{\sqrt{y}} \quad (6)$$

where:

$$\sigma = \sqrt{\frac{\sum(x^2) - (\sum x)^2}{y}} \quad (7)$$

and  $y$  is the number of samples for that value of  $x$ . These errors are shown as a shaded region in Figure 7.

Both Grid and Circular algorithms performed very well for  $n$  values of 5 and 10, while the grid algorithm begins to fail at the smaller distances for  $n \geq 15$ . The base curve remained within the errors for the circular algorithm for  $n = 15$ , but not so for the grid algorithm at distances less than 20 pixels. It can also be seen that the circular algorithm works better than the grid algorithm for separations  $\approx > 80$  pixels. This is due to the structure function being defined on the circle rather than the square. However, the grid algorithm could be adjusted so that the longest distances between pixels are sampled.

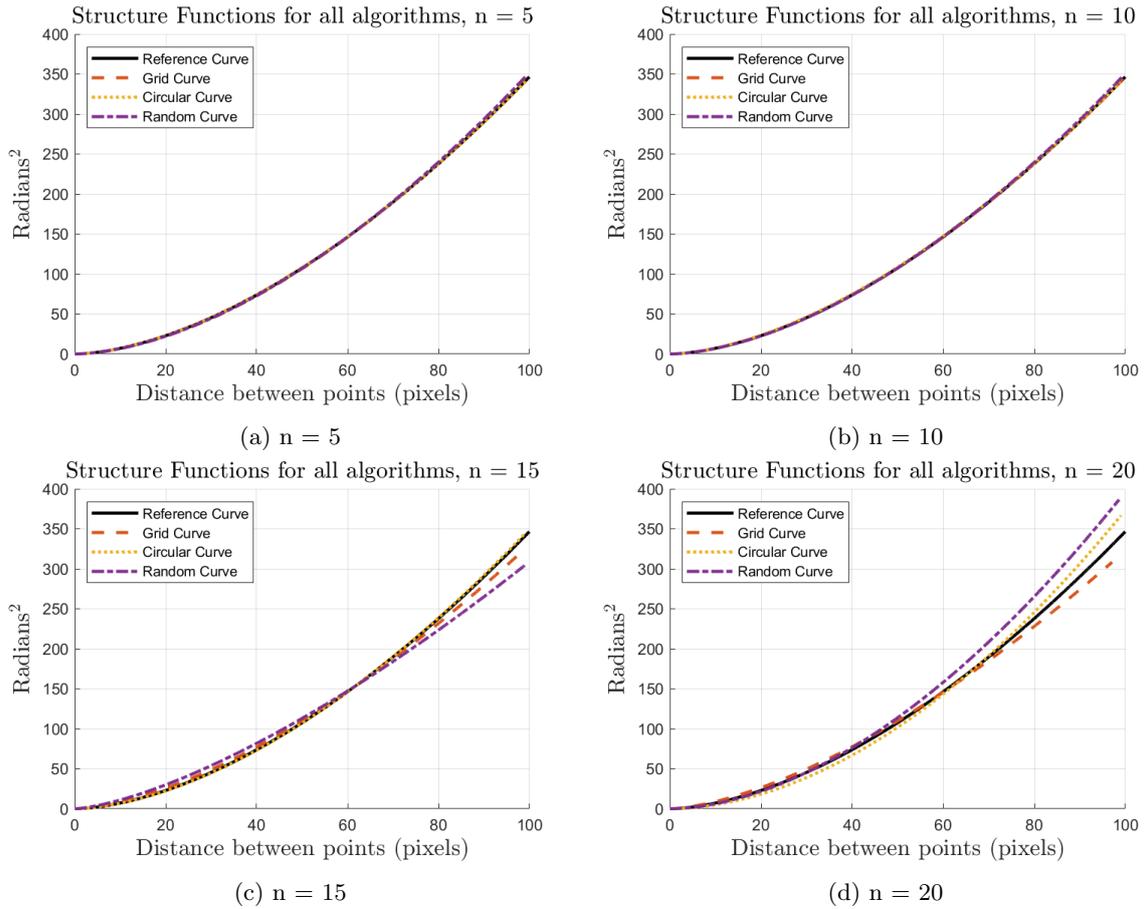


Figure 5: Structure functions for a data set of 100 phase screens of dimensions  $100 \times 100$  pixels. The reference curve corresponds to the BSF where  $n = 1$

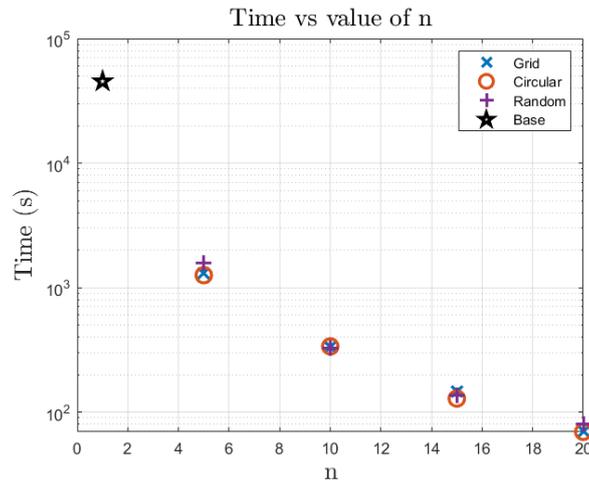


Figure 6: Time to complete 100 structure functions of the  $100 \times 100$  screens for all three algorithms for the structure functions shown in Figure 5

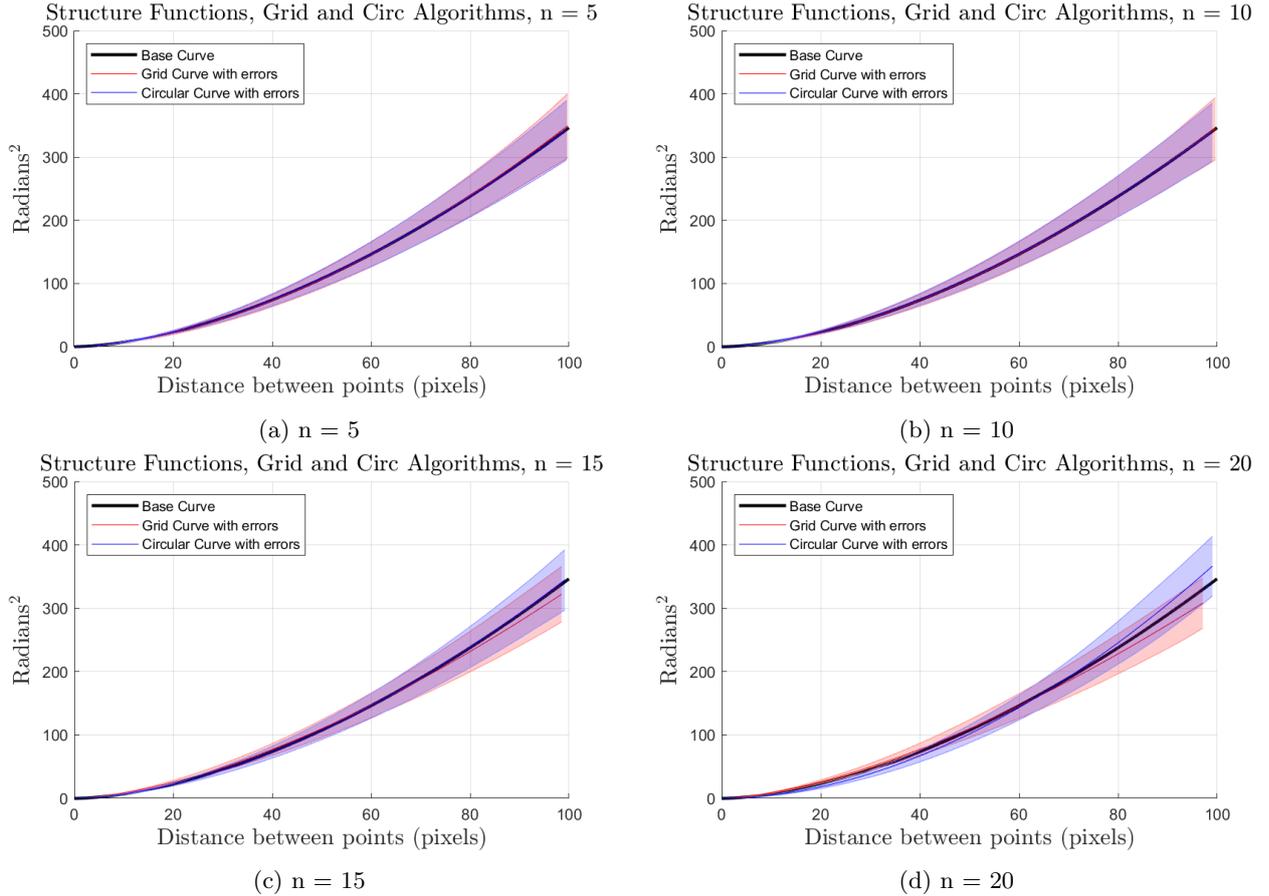


Figure 7: Resulting structure functions of the  $100 \times 100$  phase screens with errors for  $n$  values of 5, 10, 15, and 20.

## 5. 512 X 512 PHASE SCREEN TESTING AND COMPARISON WITH FOURIER METHOD

The grid and circular algorithms are then tested on  $100 \times 512 \times 512$  phase screens. Due to the time constraint of calculating a base curve to compare to, the results are instead compared to the theoretical curve generated from the Kolmogorov Theory (Equation 2).

An alternative method of calculating the structure function, is to use a Fourier transform method. This is based on the following relationship [6]

$$D_\phi(r) = 2[B_\phi(0) - B_\phi(r)] \quad (8)$$

where  $B_\phi(r)$  is the covariance function of the phase. In this case, the Fourier power spectrum of the phase screen (with mean subtracted) is inverse Fourier transformed to obtain the autocorrelation of the phase screen. The structure function estimate is given by taking a radial average and using 8.

The results show that both grid and circular algorithms give very similar results. Both resulting structure functions are a little higher than the theoretical, but still remaining within the error. Again noted here is the lack of values at higher separations for the grid algorithm for  $n = 150$  and  $n = 200$ . This is due to the geometry of the grid not providing pixels with enough separation to make a measurement at these distances. The circular algorithm, however, which contains at least one pixel at, or close to, the edge of the aperture, was able to provide a measurement at these separations. The grid algorithm could be improved by ensuring that at least one comparison pixel is at the edge of the aperture.

In Figure 8, the resulting Structure functions from the Grid and Circular algorithms are compared with the Fourier method. The results show that the Fourier results overestimate the phase difference squared for the

shorter distances; however, it does it much faster than the comparison algorithms, as shown in Table 2.

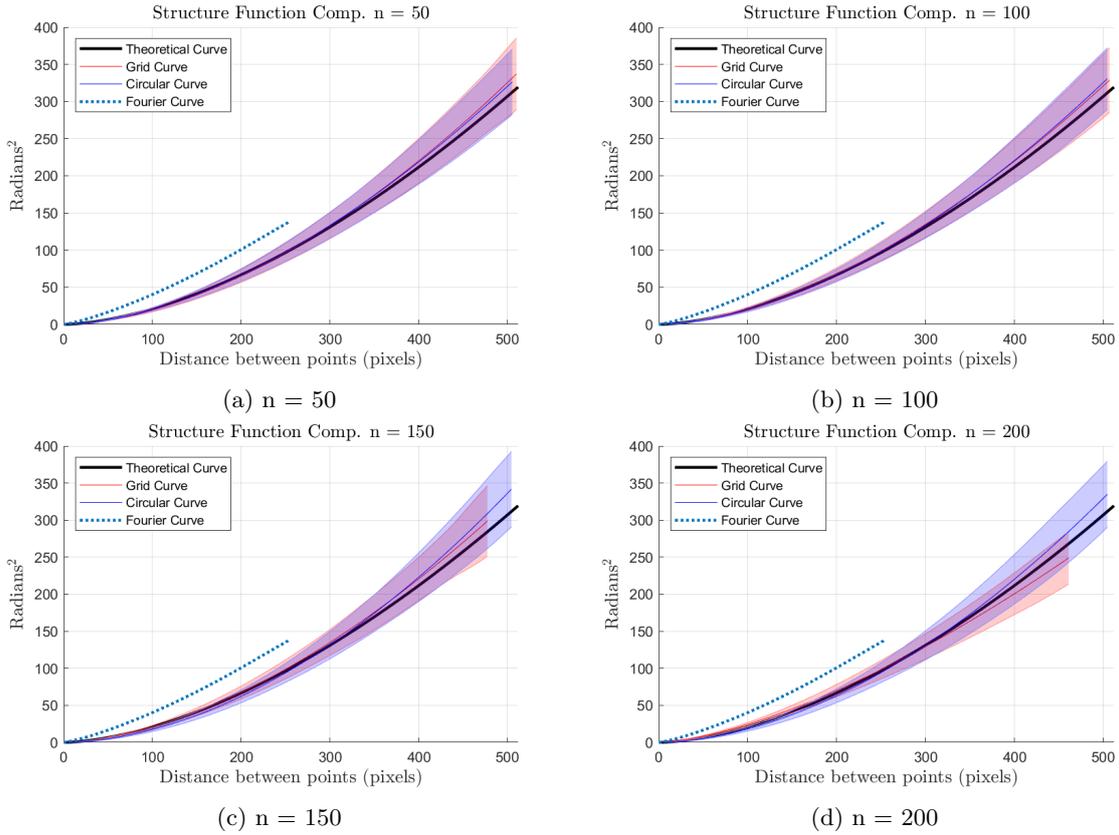


Figure 8: Resulting structure functions with errors for n values of 50, 100, 150, and 200.

Table 2: Time taken to calculate a structure function of a single  $512 \times 512$  phase screen

Algorithm	Time (s)
Fourier	1.2
Grid (n = 50)	658
Grid(n = 200)	39
Circular (n = 50)	731
Circular (n = 200)	81

The Fourier method is much faster than the algorithms that just compare the pixels themselves, but the algorithm doesn't produce the expected structure function. This may be due to the effect of windowing, i.e. the fact that the phase screen has a finite size. It is also clear that the value of the central pixel of the estimated autocorrelation directly affects the estimated structure function scaling.

## 6. CONCLUSION

In this work, we have presented a comparison of three algorithms for the calculation of structure functions. The three algorithms focus on the comparison of the phase values between pixels with different numbers of difference calculations and pixel location geometries. The grid algorithm, which compares pixels in an equally spaced grid, performed comparably to the circular algorithm, which uses pixels in concentric circles. The random algorithm, which compares to randomly location pixels, produced different results for each iteration and so was discarded as the results were not repeatable.

The two remaining algorithms were tested on  $512 \times 512$  pixel phase screens and compared to the theoretical structure function given by the Kolmogorov Power Law. The functions produced by the algorithms matched with the theoretical within the errors for  $n$  values of 50 and 100, but began to deviate for 150 and 200. The grid algorithm also did not provide comparisons at the larger distances ( $> \approx 470$  pixels). These results were also compared to a Fourier method of measuring the structure function, which showed that the Fourier method over-estimated the squared phase difference between two pixels. While this may be due to window effects for the Fourier transform, further work is needed to confirm whether this is the case or whether a different mechanism is producing the discrepancy.

## ACKNOWLEDGMENTS

This work has been supported by the Irish Research Council under the Government of Ireland Postgraduate Scholarship Programme (Project ID GOIPG/2020/537).

## References

- [1] David L Fried. “Optical resolution through a randomly inhomogeneous medium for very long and very short exposures”. In: *JOSA* 56.10 (1966), pp. 1372–1379.
- [2] Cressida M Harding, Rachel A Johnston, and Richard G Lane. “Fast simulation of a Kolmogorov phase screen”. In: *Applied optics* 38.11 (1999), pp. 2161–2170.
- [3] A.A.M. Holtslag. “Atmospheric Turbulence”. English. In: *Encyclopedia of Physical Science and Technology, Third Edition, volume 1*. Ed. by R.A. Meyers. (contribution is refereed). Academic Press, 2001, pp. 707–719.
- [4] Andrey Nikolaevich Kolmogorov. “The local structure of turbulence in incompressible viscous fluid for very large Reynolds numbers”. In: *Cr Acad. Sci. URSS* 30 (1941), pp. 301–305.
- [5] Rachel Rampy et al. “Production of phase screens for simulation of atmospheric turbulence”. In: *Applied optics* 51.36 (2012), pp. 8769–8778.
- [6] François Roddier. “V the effects of atmospheric turbulence in optical astronomy”. In: *Progress in optics*. Vol. 19. Elsevier, 1981, pp. 281–376.