Wavefront Prediction Using Artificial Neural Networks for Adaptive Optics

LIU, XUEWEN

Wavefront Prediction Using Artificial Neural Networks for Adaptive Optics

Xuewen Liu

A thesis presented for the degree of

Doctor of Philosophy

Centre for Advanced Instrumentation

The University of Durham

United Kingdom

21st July 2021

Wavefront Prediction Using Artificial Neural Networks for Adaptive Optics

Xuewen Liu

Abstract

Latency in the control loop of Adaptive Optics (AO) systems can severely limit its performance. Theories describing the temporal evolution of the atmospheric turbulence, such as the frozen flow hypothesis, justify the feasibility of predicting the turbulence (or equivalently its measurements) to compensate for the resultant temporal error in the system. This will mostly benefit AO assisted High Contrast Imaging (HCI) instruments for enhanced contrast, or wide-field AO systems for improved sky coverage.

In this thesis, we explore the potential of an Artificial Neural Network (ANN) as a nonlinear tool for open-loop wavefront prediction. The ANN predictor composes mainly Long Short-Term Memory (LSTM) cells, an ANN type specialised in sequence modelling and prediction. We demonstrate the efficiency and robustness of an ANN predictor both with simulated and on-sky 7 × 7 Shack-Hartmann Wavefront Sensor (SHWFS) CANARY data measured at 150 Hz, an AO demonstrator on the 4.2 m William Herschel Telescope (WHT), La Palma. We provide evidence that in addition to accurately predicting the wavefronts, an ANN predictor is also filtering high temporal frequencies such as Wavefront Sensor (WFS) noise. We show that an ANN predictor is adaptive to time-variant turbulence on sub-second level without user tuning. Specifically, we show that an ANN predictor is capable of predicting both frozen flow and non-frozen flow such as dome seeing, and that the ANN prediction can be based on a per-subaperture basis. As a pioneer, this thesis examines in great detail the characteristics of an ANN wavefront predictor and provides implications towards an on-sky implementation.

Supervisors: Tim Morris, Lisa Bardou and Chris Saunter

Declaration

The work in this thesis is based on research carried out at the Centre for Advanced Instrumentation, Department of Physics, University of Durham, England. No part of this thesis has been submitted elsewhere for any other degree or qualification, and it is the sole work of the author unless referenced to the contrary in the text.

Some of the work presented in this thesis has been published in journals and conference proceedings - the relevant publications are listed below.

Journal publications

Xuewen Liu, Tim Morris, Chris Saunter, Francisco Javier de Cos Juez, Carlos González-Gutiérrez, Lisa Bardou. Wavefront prediction using artificial neural networks for open-loop adaptive optics. *Monthly Notices of the Royal Astronomical Society*, 496(1): 456–464, 2020.

Conference proceedings

Xuewen Liu, Tim Morris and Lisa Bardou. Wavefront prediction using artificial neural networks with CANARY telemetry. In *Proceedings of SPIE 11448, Adaptive Optics Systems VII*, 740-746, 2020.

Xuewen Liu, Tim Morris and Chris Saunter. Using long short-term memory for wavefront prediction in adaptive optics. In *Artificial Neural Networks and Machine Learning – ICANN 2019: Text and Time Series*, 11730: 537-542, 2019.

Xuewen Liu, Tim Morris and Chris Saunter. Reducing bandwidth error by predicting wavefronts using long short-term memory network. In *Proceedings of the 6th AO4ELT Conference*, 2019.

Acknowledgements

It has been splendid three and a half years for me as a PhD student in Durham. There are many people I must thank for making all this happen and I could not be more grateful.

First I would like to thank my supervisory team, Tim Morris, Lisa Bardou and Chris Saunter. Tim, thank you for guiding me through the journey from a curious but puzzled student to a researcher ready for her adventure. I could not have reached here without your intelligence, persistence, optimism in the seemingly worst situation, passion for perfection, and zeal for the scientific truth! Lisa, thank you for so kindly being my unofficial supervisor. This thesis could not have been what it is without your sharp wit and infinite patience. It has been such an enjoyable journey since you taught me about the error budget! Chris, thank you for all the unique perspectives and insightful comments I could never think of at our meetings. You said 'try LSTM' and it worked!

I have been very fortunate to work with many remarkably inspiring and respectful people. I would like to thank my collaborators at University of Oviedo, Spain, Francisco Javier de Cos Juez and Carlos González-Gutiérrez, for bearing me with my lack of understanding in neural networks when I first started and for showing me a bigger world of intelligence. Here in Durham, I would like to thank Richard Myers, for so kindly guiding me through the application; James Osborn, for proofreading the draft of my first paper and all his thought-provoking remarks and suggestions; Tim Butterley, for the generous and stimulating discussions on dome seeing; Nazim Ali Bharmal, for running the CfAI Seminars and then the Journal Club that provides a larger stage for us young students and for being my contact during the lockdown.

I would like to thank especially my most wonderful office mates, Abi, Mark, Ollie, Vlad, Lai and Katy, for making OCW11 the best office! Many many thanks to Huizhe, David, Nico, Zhentao, Wei, Amrit and Dougie for filling me with so much fun and memories. Thank you also to everyone in CfAI who have made my years in Durham so enjoyable and unforgettable.

I would also like to extend thanks to my supervisor when I did my undergraduate and Masters in China, Yonghui Liang. It has been almost ten years since she showed me the tiny and intriguing book, Lighter Side of Adaptive Optics, that got me interested in this field in the first place.

A big thank you to my dearest friends, Jiabiao and Nandi, for their encouraging philosophy. Also to my boyfriend, Han, for his endless understanding, respect, encouragement and support. Finally and most importantly, to my parents Jun and Hong, it is nothing else but their unspoken love and faith that accompanies me all the way, all these years.

Life is a pure flame and we live by an invisible sun within us.

Sir Thomas Browne, 1658

Contents

List of Figures

List of Tables

Nomenclature

ANN Artificial Neural Network

AO Adaptive Optics

BPTT Backpropagation Through Time

CCD Charged Coupled Device

CfAI Centre for Advanced Instrumentation

CNN Convolutional Neural Network

CoG Centre of Gravity

DM Deformable Mirror

ELT Extremely Large Telescope

ESO European Southern Observatory

FC Fully Connected

FFT Fast Fourier Transform

FOV Field of View

FWHM Full Width at Half Maximum

GD Gradient Descent

GPI Gemini Planet Imager

GS Guide Star

HCI High Contrast Imaging

LGS Laser Guide Star

LMMSE Linear Minimum Mean Squared Error

LQG Linear Quadratic Gaussian

LSTM Long Short-Term Memory

LWE Low Wind Effect

MLP Multilayer Perceptron

MOAO Multi-Object Adaptive Optics

MSE Mean Squared Error

MVM Matrix Vector Multiplication

NGS Natural Guide Star

POL Pseudo Open Loop

POLC Pseudo Open Loop Control

PSD Power Spectral Density

PSF Point Spread Function

ReLU Rectified Linear Unit

RMS Root Mean Squared

RMSE Root Mean Squared Error

RNN Recurrent Neural Network

RON Readout Noise

RTC Real-Time Control

SCAO Single-Conjugate Adaptive Optics

SGD Stochastic Gradient Descent

SHWFS Shack-Hartmann Wavefront Sensor

SNR Signal-to-Noise Ratio

STD Standard Deviation

TCoG Thresholded Centre of Gravity

TMT Thirty Metre Telescope

TS Truth Sensor

TT Tip and Tilt

VLT Very Large Telescope

WFE Wavefront Error

WFS Wavefront Sensor

WHT William Herschel Telescope

XAO Extreme Adaptive Optics

Chapter 1

Introduction

Tremendous success in Astronomy over the last few decades has been revolutionising our understanding of existence and being. In these findings, ground-based telescopes play a key role in delivering scientific information from the universe. Among recent technologies that significantly enhance the performance of optical/infrared ground-based telescopes, astronomical Adaptive Optics (AO) is the technique that mitigates image blurring caused by the turbulence in the Earth's atmosphere in real time, which otherwise worsens as the size of the telescope increases. With the success of modern AO, the resolving power of the largest telescopes ever built can be fully delivered to study smaller, further objects in far greater detail.

1.1 Motivation

All AO systems, in their basic form, consist of a Wavefront Sensor (WFS) that measures the wavefront aberrations, an adaptive optical element, usually a Deformable Mirror (DM), correcting the aberrations, and a real-time control system linking these two. The inevitable finite integration time of the WFS and the computation time within the control system induces a time lag between wavefront sensing and correction. This time lag is usually on the same order of the characteristic time of the atmospheric turbulence (milliseconds) that depicts how fast the turbulence evolves. The resultant temporal error in the system can severely limit the AO performance.

For Extreme Adaptive Optics (XAO) systems for High Contrast Imaging (HCI) of exoplanets, the temporal error results in broadening of the Point Spread Function (PSF) along dominant wind directions, which severely degrades contrast, especially at small star separations (Kasper, 2012; Males and Guyon, 2018). For wide-field AO systems dominated by tomographic errors, to keep temporal error tolerable, the integration time of wavefront sensing and thus guidable star magnitude (either natural or laser) is limited, limiting sky coverage of the system (Correia et al., 2014; Jackson et al., 2015). One way to overcome this is to attempt to predict the future wavefront based on recent past wavefront measurements. Under the frozen flow hypothesis (Taylor, 1938), the turbulence volume is modeled as a linear composition of static, independent layers, each translating across the telescope aperture with certain velocity as a result of dominant wind at that layer. Because of this temporal correlation, it is possible that the future wavefronts can be partially predicted using past measurements. This hypothesis is a reasonable simplification of the turbulence for wavefront prediction purposes.

1.2 Related Work

Predictive control in AO is an active research area that incorporates wavefront prediction based on the frozen flow hypothesis into controller design. One of the most popular schemes is the Kalman filter based Linear Quadratic Gaussian (LQG) control (Paschall and Anderson, 1993; Le Roux et al., 2004). Under this framework, the whole system (both turbulence and AO system) is represented by a small set of state variables. Linear assumptions, such as an autoregressive process, are used to describe temporal evolution of those variables as well as their links with system measurements. Priors from system telemetry and noise statistics are then combined to obtain the control law. Because of their flexibility in structure, LQG predictors allow for consideration of other system error sources such as static error and vibration. Numerical and laboratory implementations focusing on a single or a few Zernike modes show great improvement in terms of overall residual Wavefront Error (WFE) or Strehl ratio (Le Roux et al., 2004; Kulcsár et al., 2012), especially in vibration filtering (Petit et al., 2006, 2008). Poyneer et al. (2007) developed a computationally efficient Fourier based LQG predictive controller, which can be extended to non-integer loop delays (Poyneer and Véran, 2008), facilitating graceful formulation of wind-blown turbulence evolution under Fourier basis. Laboratory tests demonstrate a reduction of around 67% in temporal error using a full Fourier LQG controller (Rudy et al., 2015). Correia et al. (2014) incorporates open-loop wavefront prediction into a minimum Mean Squared Error (MSE) tomographic reconstructor design for Multi-Object Adaptive Optics (MOAO) systems. This tomographic predictor allows the use of guide stars one magnitude fainter (corresponding to an increase in the density of available stars by a factor of 1.8) in end-to-end simulations of RAVEN (Andersen et al., 2012), which is expected to be further improved if deployed within the LQG framework. LQG based predictive control has been deployed for AO systems on HCI instrument SPHERE (Petit et al., 2014) for both turbulence correction and vibration filtering in Tip and Tilt (TT) modes. Stability and robustness of LQG controller in full-mode Single-Conjugate Adaptive Optics (SCAO) control has also been verified on sky (Sivo et al., 2014), showing overall performance improvement over a standard integrator controller in conditions where temporal error is not dominant.

Data-driven approaches to predictive control remove the constraint of an explicit physical model describing the turbulence evolution, aiming at fully exploiting linear spatio-temporal correlations within input telemetry and improving controller robustness. Guyon and Males (2017) deployed the Empirical Orthogonal Functions framework. Numerical HCI simulations demonstrate significantly improved contrast and resistance against sensor noise. van Kooten et al. (2019) proposed an Linear Minimum Mean Squared Error (LMMSE) predictor and successfully tested its robustness using on-sky SPHERE data (van Kooten et al., 2020).

On the other hand, to adapt to varying turbulence conditions, frequent monitoring (e.g. at least every 10 s as suggested in Poyneer et al. (2007)) of some turbulence parameters such as the wind speed might be unavoidable to update the LQG control law. For the linear data-driven approaches, this would require the reset and on-line re-learning of the controller parameters. An estimate of the noise level for determining the LQG control law also suggests the controller requires updating when this condition changes. Similarly, the predictive reconstructor for MOAO systems requires re-computation when the Guide Star (GS) geometry varies (Correia et al., 2014). These will impose an additional complexity on the AO Real-Time Control (RTC) and system calibration.

In this thesis, we exploit the potential of Artificial Neural Network (ANN) as a nonlinear framework for wavefront prediction. It falls in the regime of the data-driven approach, learning the underlying model within the data through a training process. ANNs have the potential to take the calibration complexity out of operations, and replace it with a training scheme that imposes training constraints beforehand, but is insensitive to different conditions and does not impose an additional operational overhead on-sky. This however imposes further limitations on how accurate the training data needs to be, as will be investigated in this thesis.

ANNs have been applied to a variety of tasks within AO. Angel et al. (1990) and Sandler et al. (1991) successfully applied a simple feed-forward Multilayer Perceptron (MLP) network for wavefront sensing based on a pair of in-focus and slightly defocused images in simulation and with on-sky data respectively. Norris et al. (2020) used Convolutional Neural Network (CNN)s for the nonlinear wavefront reconstruction in the photonic lantern technique, a novel all-photonic focal-plane wavefront sensor. This nonlinear reconstructor suits the nonlinear relationship between the input phase and the output intensities. Osborn et al. (2012) successfully demonstrated an ANN tomographic reconstructor for MOAO systems, which can adapt to a wide range of atmospheric conditions and is more resistant to photon noise than linear reconstructors. This technique was later validated on-sky (Osborn et al., 2014), showing comparable performance with linear tomographic techniques with less sensitivity to the estimation error in turbulence layer heights.

For the use of ANN for wavefront prediction, early numerical simulations adopting a feed-forward MLP network demonstrate promise for using this nonlinear tool for open-loop slope prediction based on a time series of past noisy measurements by a Shack-Hartmann Wavefront Sensor (SHWFS) (Jorgenson and Aitken, 1992, 1994), with further improvement over a linear predictor when the Signal-to-Noise Ratio (SNR) of wavefront sensing gets lower (Lloyd-Hart and McGuire, 1996). The last few decades have seen significant advances in both the theory and applications of ANNs (LeCun et al., 2015), among which the Long Short-Term Memory (LSTM) network is well-suited to time series modeling and prediction by design (Hochreiter and Schmidhuber, 1997; Gers et al., 1999). The adaptive memory elements within the LSTM makes it competitive for predicting evolving turbulence without user tuning. Swanson et al. (2018) explored the prediction of wavefront phase map from its last few frames using convolutional LSTM, a combination of CNN and LSTM for simultaneously extracting spatial and temporal correlations from image sequences. This was later extended to the prediction of slopes in a simulated Pseudo Open Loop (POL) system, outperforming an integrator under a variety of WFS SNR conditions (Swanson et al., 2021). Landman et al. (2020) experimented with a closed-loop LSTM-based AO controller for the attenuation of both vibration and latency in TT control. They showed that the ANN can adapt to varying vibration frequency without retraining. The resurgence of ANN since early 2010s has brought more sophisticated ANN architectures and learning algorithms, however the potentials and properties of an ANN wavefront predictor remain largely under study.

1.3 Synopsis

This thesis characterises an ANN predictor in a 7 × 7 Shack-Hartmann SCAO system placed on a 4.2 metre telescope with open-loop wavefront sensing, both in simulations and with on-sky data. In addition to determining if an ANN can be used to accurately predict a future wavefront based on past measurements, we will address the following questions:

1. How robust is an ANN to changes in real turbulence conditions?
2. Is the knowledge of the spatial distribution of WFS subapertures required for ANN prediction?
3. Should an ANN predictor be trained with simulated or real data?

The reasons that we have adopted the open-loop wavefront sensing and control configuration, are mainly as follows,

1. The ANN will not need to learn the dynamics of the system such as the interaction between the WFS and the DM or the potential loop gain for closed-loop control, which will simplify the problem to predicting the temporal evolution of wavefronts only, saving the amount of training data and training time required.
2. A Pseudo Open Loop Control (POLC) scheme would be more realistic but complicated, introducing additional noise terms which can affect loop stability (Gilles, 2005). We will show that the ANN is resistant to noise. This will alleviate the stability issue, however we leave this to future study.
3. Limiting the problem to wavefront prediction only will also simplify the understanding of the ANN performance. The nonlinear nature of the ANN can complicate standard AO analyses that rely on independent error terms.

We adopt the low-order 7 × 7 AO configuration mainly for the ease of fast validation and exploitation of the concept of ANN wavefront prediction, but also because of the availability of open-loop on-sky data taken using the CANARY instrument. CANARY is an AO demonstrator built for Extremely Large Telescope (ELT) instruments, hosted by the 4.2 m William Herschel Telescope (WHT) on La Palma in the Canary islands.

In **Chapter 2** we provide theoretical background for atmospheric turbulence, SCAO systems and describe the AO simulation tool, Soapy (Reeves, 2016), that is used throughout this thesis.

In **Chapter 3** we provide necessary mathematical background for ANNs and describe the ANN training methodology. We provide the mathematical framework for the operation and training of ANNs. We present the ANN architectures used in this thesis. We then describe the simulated CANARY SCAO system and how the training is undertaken for this system. Using the simulated CANARY system we will generate the ANN training data and evaluate the prediction performance.

In **Chapter 4** we demonstrate the effectiveness and robustness of the ANN predictor within the simulated SCAO system under the frozen flow hypothesis. Both time-invariant and time-variant turbulence conditions are considered. The ANN can generalise to multi-layer profiles or systems with a two-frame latency. Specifically, we investigate the impact of training noise level on the performance. We explore a non-spatially aware ANN where the prediction is on a per-subaperture basis and the ANN has not been trained with any knowledge of the spatial distribution of WFS subapertures with respect to one another.

In **Chapter 5** we demonstrate the proficiency of the ANN predictor with CANARY on-sky telemetry taken by the on-axis SHWFS in open loop. We identify that CANARY experienced strong dome turbulence instead of frozen flow, and that strong vibrations existed in TT modes. We show the improved ANN performance brought by the use of an empirical stationary turbulence model, and how the vibration impacts ANN performance. Analyses with the training noise level and the spatial awareness of the ANN are conducted as in the previous chapter, revealing expectations and considerations for on-sky implementations.

Finally, in **Chapter 6** we summarise the conclusions drawn from these studies and discuss the future prospects for the use of ANN prediction with AO systems.

1.4 References

D. R. Andersen, K. J. Jackson, C. Blain, C. Bradley, C. Correia, M. Ito, O. Lardière, and J.-P. Véran. Performance modeling for the RAVEN multi-object adaptive optics demonstrator. *Publications of the Astronomical Society of the Pacific*, 124 (915):469–484, 2012.

J. R. P. Angel, P. Wizinowich, M. Lloyd-Hart, and D. Sandler. Adaptive optics for array telescopes using neural-network techniques. *Nature*, 348:221–224, 1990.

C. Correia, K. Jackson, J.-P. Véran, D. Andersen, O. Lardière, and C. Bradley. Static and predictive tomographic reconstruction for wide-field multi-object adaptive optics systems. *J. Opt. Soc. Am. A*, 31(1):101–113, 2014.

F. A. Gers, J. Schmidhuber, and F. Cummins. Learning to forget: continual prediction with lstm. In *1999 Ninth International Conference on Artificial Neural Networks ICANN 99. (Conf. Publ. No. 470)*, volume 2, pages 850–855, 1999.

L. Gilles. Closed-loop stability and performance analysis of least-squares and minimum-variance control algorithms for multiconjugate adaptive optics. *Appl. Opt.*, 44(6):993–1002, 2005.

O. Guyon and J. Males. Adaptive Optics Predictive Control with Empirical Orthogonal Functions (EOFs). 2017. `http://arxiv.org/abs/1707.00570`.

S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 9:1735–1780, 1997.

K. Jackson, C. Correia, O. Lardière, D. Andersen, and C. Bradley. Linear prediction of atmospheric wave-fronts for tomographic adaptive optics systems: modelling and robustness assessment. *Opt. Lett.*, 40(2):143–146, 2015.

M. B. Jorgenson and G. J. M. Aitken. Prediction of atmospherically induced wavefront degradations. *Opt. Lett.*, 17(7):466–468, 1992.

M. B. Jorgenson and G. J. M. Aitken. Wavefront Prediction for Adaptive Optics. *ESO Conference and Workshop Proceedings*, 48:143–148, 1994.

M. Kasper. Adaptive optics for high contrast imaging. In *Proc. of SPIE 8447*, volume 8447, page 84470B, 2012.

C. Kulcsár, H.-F. Raynaud, C. Petit, and J.-M. Conan. Minimum variance prediction and control for adaptive optics. *Automatica*, 48(9):1939–1954, 2012.

R. Landman, S. Y. Haffert, V. M. Radhakrishnan, and C. U. Keller. Self-optimizing adaptive optics control with reinforcement learning. In *Adaptive Optics Systems VII*, volume 11448, pages 842 – 856. SPIE, 2020.

B. Le Roux, J.-M. Conan, C. Kulcsár, H.-F. Raynaud, L. M. Mugnier, and T. Fusco. Optimal control law for classical and multiconjugate adaptive optics. *J. Opt. Soc. Am. A*, 21(7):1261–1276, 2004.

Y. LeCun, T. Bengio, and G. Hinton. Deep learning. *Nature*, 521:436–444, 2015.

M. Lloyd-Hart and P. McGuire. Spatio-temporal prediction for adaptive optics wavefront reconstructors. *ESO Conference and Workshop Proceedings*, 54:95, 1996.

J. R. Males and O. Guyon. Ground-based adaptive optics coronagraphic performance under closed-loop predictive control. *J. Astron. Telesc. Instrum. Syst*, 4(1): 019001, 2018.

B. R. M. Norris, J. Wei, C. H. Betters, A. Wong, and S. G. Leon-Saval. An all-photonic focal-plane wavefront sensor. *Nature Communications*, 11(1):5335, 2020.

J. Osborn, F. J. D. C. Juez, D. Guzman, T. Butterley, R. Myers, A. Guesalaga, and J. Laine. Using artificial neural networks for open-loop tomography. *Opt. Express*, 20(3):2420–2434, 2012.

J. Osborn, D. Guzman, F. J. de Cos Juez, A. G. Basden, T. J. Morris, E. Gendron, T. Butterley, R. M. Myers, A. Guesalaga, F. Sanchez Lasheras, M. Gomez Victoria, M. L. Sánchez Rodríguez, D. Gratadour, and G. Rousset. Open-loop tomography with artificial neural networks on CANARY: on-sky results. *Monthly Notices of the Royal Astronomical Society*, 441(3):2508–2514, 2014.

R. N. Paschall and D. J. Anderson. Linear quadratic gaussian control of a deformable mirror adaptive optics system with time-delayed measurements. *Appl. Opt.*, 32(31):6347–6358, 1993.

C. Petit, J.-M. Conan, C. Kulcsár, H.-F. Raynaud, T. Fusco, J. Montri, and D. Rabaud. First laboratory demonstration of closed-loop kalman based optimal control for vibration filtering and simplified mcao. In *Proc. of SPIE 6272*, volume 6272, page 62721T, 2006.

C. Petit, J.-M. Conan, C. Kulcsár, H.-F. Raynaud, and T. Fusco. First laboratory validation of vibration filtering with lqg control law for adaptive optics. *Opt. Express*, 16(1):87–97, 2008.

C. Petit, J.-F. Sauvage, T. Fusco, A. Sevin, M. Suarez, A. Costille, A. Vigan, C. Soenke, D. Perret, S. Rochat, A. Barrufolo, B. Salasnich, J.-L. Beuzit, K. Dohlen, D. Mouillet, P. Puget, F. Wildi, M. Kasper, J.-M. Conan, C. Kulcsár, and H.-F. Raynaud. Sphere extreme ao control scheme: final performance assessment and on sky validation of the first auto-tuned lqg based operational system. In *Proc. of SPIE 9148*, volume 9148, page 91480O, 2014.

L. A. Poyneer and J.-P. Véran. Predictive wavefront control for adaptive optics with arbitrary control loop delays. *J. Opt. Soc. Am. A*, 25(7):1486–1496, 2008.

L. A. Poyneer, B. A. Macintosh, and J.-P. Véran. Fourier transform wavefront control with adaptive prediction of the atmosphere. *J. Opt. Soc. Am. A*, 24(9): 2645–2660, 2007.

A. Reeves. Soapy: an adaptive optics simulation written purely in python for rapid concept development. In *Proc. of SPIE 9909*, volume 9909, page 99097F. International Society for Optics and Photonics, 2016.

A. Rudy, L. A. Poyneer, S. Srinath, S. M. Ammons, and D. Gavel. A laboratory demonstration of an LQG technique for correcting frozen flow turbulence in adaptive optics systems. 2015. `https://arxiv.org/abs/1504.03686`.

D. G. Sandler, T. K. Barrett, D. A. Palmer, R. Q. Fugate, and W. J. Wild. Use of a neural network to control an adaptive optics system for an astronomical telescope. *Nature*, 351:300–302, 1991.

G. Sivo, C. Kulcsár, J.-M. Conan, H.-F. Raynaud, Éric Gendron, A. Basden, F. Vidal, T. Morris, S. Meimon, C. Petit, D. Gratadour, O. Martin, Z. Hubert, A. Sevin, D. Perret, F. Chemla, G. Rousset, N. Dipper, G. Talbot, E. Younger, R. Myers, D. Henry, S. Todd, D. Atkinson, C. Dickson, and A. Longmore. First on-sky scao validation of full lqg control with vibration mitigation on the canary pathfinder. *Opt. Express*, 22(19):23565–23591, 2014.

R. Swanson, M. Lamb, C. Correia, S. Sivanandam, and K. Kutulakos. Wavefront reconstruction and prediction with convolutional neural networks. In *Proc. of SPIE 10703*, volume 10703, page 107031F, 2018.

R. Swanson, M. Lamb, C. M. Correia, S. Sivanandam, and K. Kutulakos. Closed Loop Predictive Control of Adaptive Optics Systems with Convolutional Neural Networks. *Monthly Notices of the Royal Astronomical Society*, 2021.

G. I. Taylor. The spectrum of turbulence. *Proceedings of the Royal Society A*, 164 (919):476–490, 1938.

M. van Kooten, N. Doelman, and M. Kenworthy. Impact of time-variant turbulence behavior on prediction for adaptive optics systems. *J. Opt. Soc. Am. A*, 36(5): 731–740, 2019.

M. van Kooten, N. Doelman, and M. Kenworthy. Robustness of prediction for extreme adaptive optics systems under various observing conditions. *Astronomy and Astrophysics*, 636:A81, 2020.

Chapter 2

Theory

2.1 Atmospheric Turbulence

As solar energy is transferred through Earth's atmosphere, the temperature of the air, hence the density, fluctuates spatially and temporally. Optical turbulence is caused by mechanical mixing of large air masses of differing temperatures. In a dense medium, the light travels more slowly, implying a larger refractive index. For the incoming light, atmospheric turbulence forms a continuous screen of constantly varying refractive indices, acting as arrays of lenses. When imaging through the atmosphere, different parts of the light from an astronomical source propagates through different paths, resulting in parts of the incoming wavefront being delayed with respect to others. The once-flat wavefront becomes distorted. The wavefront distortion spreads the energy received into a diffuse disk in the focal plane, considerably degrading the image quality.

Astronomical Adaptive Optics (AO) is the technique used to compensate for atmospheric turbulence for ground-based telescopes in real time. For the performance of an AO system to be predicted and optimised, it is essential to understand the theory behind the turbulence.

2.1.1 Spatial Properties of the Atmosphere

In 1941, Kolmorogov developed a theory to analyse the mechanical structure of turbulence (Kolmogorov, 1941), enabling statistical description of its effects. In this theory, atmospheric turbulence is formed by the cascade of thermal energy from large to smaller scales. The energy is injected on large spatial scale and forms eddies. As the turbulent flow breaks up, the energy is transferred to smaller scales until the eddies become small enough that the energy is dissipated by viscous friction.

Kolmogorov introduced the idea of a structure function to describe the refractive index of the atmosphere. The structure function of refractive index n is defined as the mean-square difference in refractive index between two points separated by $\boldsymbol{r}$ (Hardy, 1998)

$$\begin{aligned} D_n(r) &\equiv \langle [n(\boldsymbol{x}+\boldsymbol{r}) - n(\boldsymbol{x})]^2 \rangle \\ &= C_n^2 r^{2/3}, \end{aligned} \tag{2.1}$$

where $r = |\boldsymbol{r}|$, $\boldsymbol{x}$ is a point in space. C_n^2 is the refractive index structure constant in $\mathrm{m}^{-2/3}$, which can be used to quantify the turbulence strength. Though termed constant, C_n^2 is a function of both altitude and time (Hardy, 1998).

Roddier used the thin layer approximation to study the cumulative optical effects of the turbulence (Roddier, 1981). Assume a thin turbulence layer at altitude h with a thickness of δh. δh is large compared with the eddy size, but small enough for diffraction effects within the layer to be ignored. The phase structure function at the output of this layer is then

$$D_\phi(h,r) = 2.914 k^2 r^{5/3} C_n^2(h) \delta h, \tag{2.2}$$

where $k = 2\pi/\lambda$, with λ being the wavelength of the light. For a continuous distribution of turbulence, the phase structure function at the telescope pupil, $D_\phi(r)$, is the integral of $D_\phi(h,r)$ along the imaging path,

$$D_\phi(r) = \int \sec\zeta D_\phi(h,r), \tag{2.3}$$

where ζ is the zenith angle.

The Fried parameter r_0, also termed the coherence length, describes in a single parameter the integrated effect of the refractive-index fluctuations throughout the atmospheric volume (Fried, 1965),

$$r_0 = \left(0.423k^2 \sec\zeta \int C_n^2(h)\delta h\right)^{-3/5}. \tag{2.4}$$

Using this notation, $D_\phi(r)$ can be expressed in terms of r_0,

$$D_\phi(r) = 6.88\left(\frac{r}{r_0}\right)^{5/3}. \tag{2.5}$$

The significance of r_0 is that it defines the diameter of a pupil within which the Root Mean Squared (RMS) phase error is approximately 1 rad for a given wavelength (Noll, 1976). The phase error of 1 rad is the threshold above which the image quality deteriorates quickly (Hardy, 1998). Thus, small values of r_0 correspond to stronger turbulence. Eq. 2.4 shows that r_0 is wavelength dependent. r_0 is usually defined at 500 nm.

The phase power spectrum $\Phi_\phi(f)$ describes the distribution of the turbulence strength with respect to spatial frequency $\boldsymbol{f}$, and is related to the phase structure function by

$$D_\phi(r) = 2\int \Phi_\phi(f)(1-\cos(2\pi\boldsymbol{f}\cdot\boldsymbol{r}))\delta\boldsymbol{f}, \tag{2.6}$$

where $\cdot$ represents dot product. Combining Eqs. 2.5 and 2.6 yields Kolmogorov phase power spectrum

$$\Phi_\phi^{\mathrm{K}}(f) = 0.023 r_0^{-5/3} f^{-11/3}. \tag{2.7}$$

This power law breaks down at very large or very small spatial scales, i.e. $r > L_0$ or $r < l_0$, where l_0 is known as the inner scale and L_0 is the outer scale. Tatarskii modified this spectrum for $r < l_0$ (Tatarskii, 1961). However, since very little power is contained in small spatial scales (large f), the effect of a finite l_0 in AO simulation can be safely ignored. On large spatial scales (small f), the Kolmogorov spectrum tends towards infinity, implying unphysical infinite energy input. Besides, the measured value of L_0 vary between 10 and 100 m (Ziad et al., 2004; Ono et al., 2017), comparable to the size of a modern telescope. The effect of a finite L_0 should thus be considered in numerical simulations. This leads to the von Kármán spectrum,

$$\Phi_\phi^{\mathrm{vK}}(f) = 0.023 r_0^{-5/3} (f^2 + f_0^2)^{-11/6}, \tag{2.8}$$

where $f_0 = 1/L_0$.

2.1.2 Temporal Properties of the Atmosphere

Although the structure of the turbulence across the telescope aperture evolves with time due to temperature fluctuations, this will be on time scales much longer than those caused by wind-blowing turbulence translation. Thus, the temporal fluctuations of turbulence observed by the telescope are mainly the latter (Hardy, 1998). This is known as Taylor's frozen flow hypothesis (Taylor, 1938).

Taylor's hypothesis enables the conversion between spatial and temporal properties of turbulence. Given this hypothesis and Eq. 2.5 describing the spatial phase structure function, we can define a temporal phase structure function

$$\begin{aligned} D_\phi(\tau) &\equiv \langle [\phi(\boldsymbol{x}, t+\tau) - \phi(\boldsymbol{x}, t)]^2 \rangle \\ &= 6.88 \left(\frac{\bar{v}\tau}{r_0} \right)^{5/3}. \end{aligned} \tag{2.9}$$

This function describes the evolution of the phase at any point $\boldsymbol{x}$ over a telescope aperture between time t and $t+\tau$. $\bar{v}$ is the average wind speed across the atmospheric volume,

$$\bar{v} = \left[\frac{\int C_n^2(h) v(h)^{5/3} \delta h}{\int C_n^2(h) \delta h} \right]^{3/5}, \tag{2.10}$$

where $v(h)$ is the wind speed at height h.

From Eq. 2.9 we can define the coherence time τ_0, which is the time for the RMS phase error to reach 1 rad,

$$\tau_0 = 0.314 \frac{r_0}{\bar{v}}. \tag{2.11}$$

2.1.3 Zernike Polynomials for Modal Representation of Turbulence

Wavefronts over a disk (such as the telescope aperture without a central obscuration) generated by the atmospheric turbulence can be represented by a set of orthonormal functions of ascending spatial frequency. One commonly used set of two-dimensional orthogonal functions are Zernike polynomials, defined on a unit disk in polar coordinates by

$$Z_n^m = \sqrt{n+1}R_n^m(r)\begin{cases}\sqrt{2}\mathrm{cos}(m\theta),\ m \neq 0\\ \sqrt{2}\mathrm{sin}(m\theta),\ m \neq 0\\ 1{,}m = 0\end{cases} \tag{2.12}$$

where

$$R_n^m(r) = \sum_{S=0}^{(n-m)/2} \frac{(-1)^S(n-S)!r^{n-2S}}{S![(n+m)/2-S]![(n-m)/2-S]!}. \tag{2.13}$$

n and m are radial and azimuthal orders respectively. Noll (1976) introduced a numbering system where each Zernike polynomial can be indexed by a single variable j, which is a function of n and m. In Noll's notation, an even j corresponds to cos($m\theta$) while an odd j corresponds to sin($m\theta$).

Zernike polynomials defined in Eq. 2.12 satisfy

$$\int d^2rW(r)Z_jZ_{j'} = \delta_{jj'}, \tag{2.14}$$

where

$$\begin{aligned} W(r) &= 1/\pi, \qquad r \leq 1\\ &= 0, \qquad\quad r > 1 \end{aligned} \tag{2.15}$$

and $\delta_{jj'}$ is the Kronecker delta function. Eq. 2.14 implies that each mode has an RMS value of 1 over the unit disk. Zernike modes 2-36 (in Noll's notation) used within this thesis are shown in Fig. 2.1.

An arbitrary wavefront $\phi(\rho,\theta)$ over a circular aperture of radius R can be expanded as an infinite sum of these modes (Noll, 1976),

$$\phi(\rho,\theta) = \sum_j a_jZ_j(r,\theta), \tag{2.16}$$

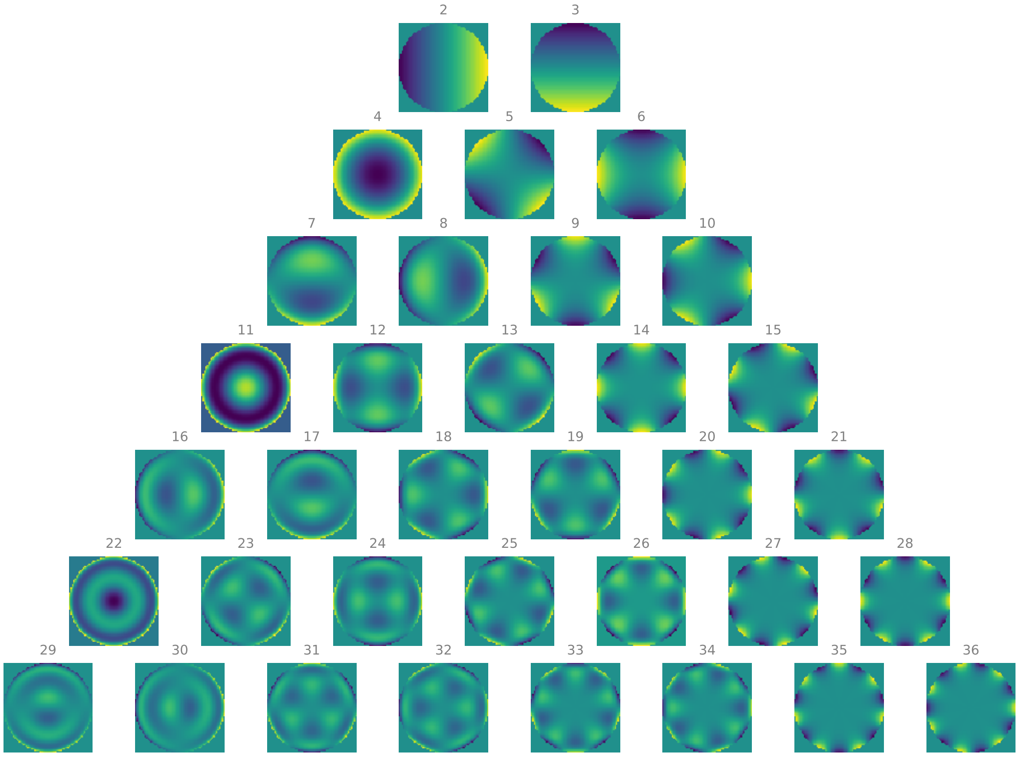


Figure 2.1: Zernike modes (in Noll's notation) 2-36 (from left to right, top to bottom). Modes 2 and 3 correspond to Tip and Tilt respectively. Modes in the same row have the same radial order.

where the normalised coordinate $r = \rho/R$ and the coefficient a_j is given by

$$a_j = \int d^2 r W(r)\phi(\rho, \theta) Z_j(r, \theta) \tag{2.17}$$

2.1.4 Imaging through Atmospheric Turbulence

The wavefront ϕ of a celestial point source, perturbed by the atmospheric turbulence, propagates through the telescope optics and forms a Point Spread Function (PSF) in the focal plane. Mathematically, this imaging process can be modelled as

$$PSF = |\mathcal{F}(Me^{i\phi})|^2, \tag{2.18}$$

where $|\cdot|$ takes the absolute value, $\mathcal{F}$ denotes the Fourier transform, and M represents the pupil mask, which is 1 inside the pupil and 0 elsewhere.

In the absence of atmospheric turbulence, the wavefront ϕ is considered flat (or equivalently a zero matrix). The resulting diffraction-limited PSF is an airy disk (shown in Fig 2.2 (a)). The Full Width at Half Maximum (FWHM) of this image is

$$\theta = 0.98\frac{\lambda}{D}, \tag{2.19}$$

where D is the diameter of the circular aperture, λ is the imaging wavelength.

The resolving power of a telescope is the minimum angular distance for two adjacent point sources of equal magnitude to be resolved. According to the Rayleigh criterion (Strutt, 1879), it is when the peak intensity of one source lies in the first minimum of the other. In the case of a circular aperture with the absence of turbulence,

$$R = 1.22\frac{\lambda}{D}. \tag{2.20}$$

Atmospheric turbulence induces an optical path difference between different parts of ϕ. The resulting focused image is no longer diffraction limited. Low- and high-order aberrations in ϕ have different impact on image formation. Short-exposure images (exposure time less than about 1/50 seconds) are composed of a number of speckles, the size of each approximating that of a diffraction-limited spot, produced by interference between rays separated by D. This structure is dominated by higher-order wavefront aberrations. The increase in the angular extent over which the speckles are spread are factored by D/r_0, as is shown in Fig 2.2 (b)-(d). For long-exposure imaging, the time-varying image motion induced by low-order aberrations (Tip and Tilt) causes speckles to add together, producing a blurred large halo.

When $D < r_0$, there is little distortion within the aperture. The image is effectively diffraction limited with an overall displacement. When $D > r_0$, the resolving power of the telescope is reduced to

$$R \approx 1.22\frac{\lambda}{r_0}, \tag{2.21}$$

which is equivalent to reducing the size of the telescope aperture to r_0. When D/r_0 is between 1 and 10, there is a modest improvement in the short-exposure

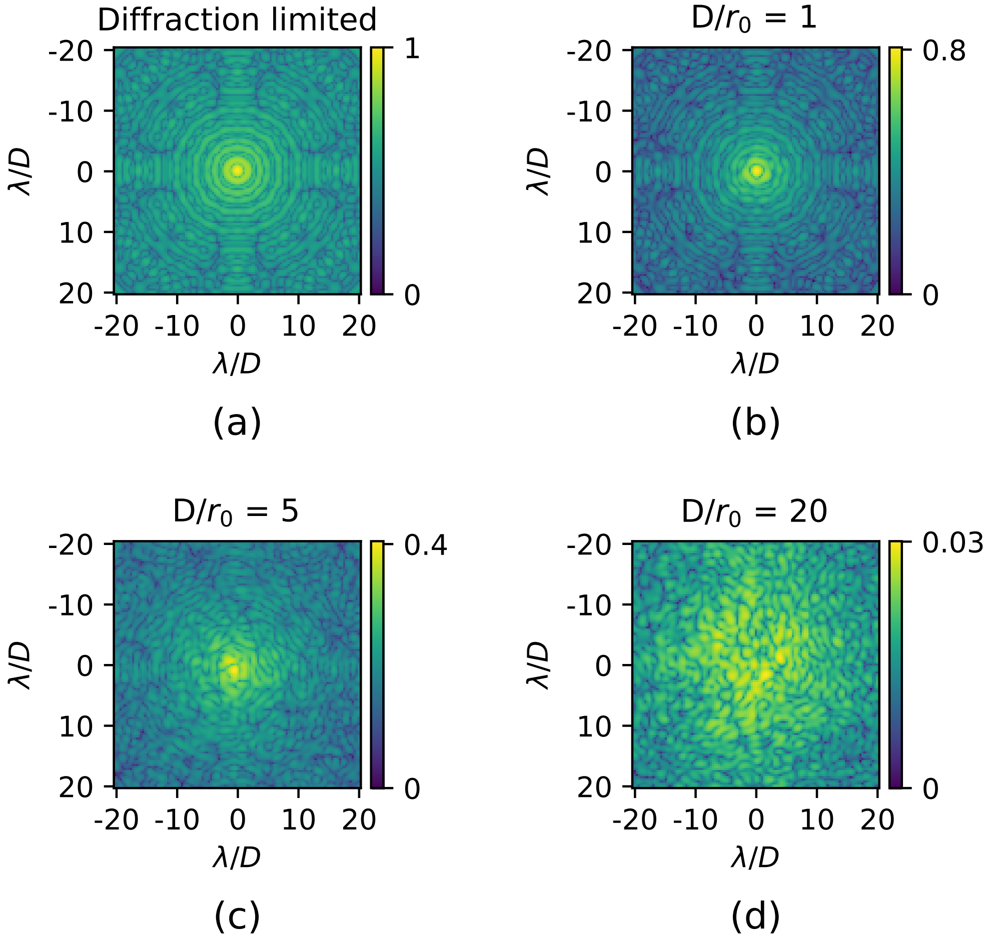


Figure 2.2: Simulated short-exposure diffraction-limited imaging of a point source at infinity (a) and turbulence-limited imaging when D/r_0 equals 1 (b), 5 (c) and 20 (d). The plots are shown in log scale. The normalised peak intensity decreases as r_0 decreases due to the spread of energy into more speckles over larger area.

telescope performance by image motion compensation, which is a simple form of AO. When $D > 10\ r_0$, the impact of image motion becomes less severe. The structure and size of the uncompensated image are mainly determined by r_0. In premier observing sites, r_0 typically varies between 5 and 20 cm (Hardy, 1998). Most modern ground-based optical observations fall within the $D > 10\ r_0$ regime. It is in this regime where the potential performance gain by the use of AO for full wavefront compensation is most prominent.

2.2 Single-Conjugate Adaptive Optics

Astronomical AO is the technique for ground-based telescopes to compensate for atmospheric turbulence in real time. This thesis focuses on Single-Conjugate Adaptive Optics (SCAO), which is a basic form of AO that corrects for the turbulence in a single observing direction.

2.2.1 Structure of an SCAO System

The layout of a closed-loop SCAO system imaging a natural celestial point source is illustrated in Fig. 2.3. The wavefront from the science target is considered flat before the turbulence introduces a wavefront perturbation. Here the science target itself is bright enough to be used as a Natural Guide Star (NGS) for wavefront sensing in addition to being imaged. The wavefront corrector, here a Deformable Mirror (DM), is deformed in such a way that the wavefront reflected by it is ideally flattened. A beamsplitter splits the residual wavefront into a wavefront sensing and control path and an imaging path. In the feedback control loop, a Real-Time Control (RTC) system converts the residual wavefront distortion measured by the Wavefront Sensor (WFS) to the driving signal (or command) of the DM in real time. In an SCAO system, the DM is optically conjugate to the telescope pupil, providing a small corrected Field of View (FOV).

An open-loop control scheme is shown in Fig. 2.4. In this scheme, the WFS measures the uncompensated instead of residual wavefront. This measurement is then converted to the DM driving signal via the RTC, however the DM correction is now unseen by the WFS, as well as errors in the correction such as those induced by WFS or DM non-linearity (Gilles, 2005). System calibration in an open-loop system is also more challenging and demanding than in a closed-loop system. This has limited the use of a pure open-loop AO systems in practice.

A third control scheme is the Pseudo Open Loop Control (POLC). This was developed for wide-field closed-loop AO systems where the minimum variance wavefront reconstruction technique is used and the Pseudo Open Loop (POL) slope measurement is required (Ellerbroek and Vogel, 2003). For such systems, POLC improves the system stability over closed-loop control. The POL measurement is recovered from the residual measurement, the DM command and the known DM-WFS response. This can be followed by a classical open-loop reconstructor. POLC can also be used in an SCAO system if the reconstruction is based on POL slopes. More details on open-loop or closed-loop implementation will be given in Section 2.2.5.

Throughout the thesis, we assume open-loop wavefront sensing and control configuration. Reasons for this have been given in Section 1.3. We will focus on the prediction power of the Artificial Neural Network (ANN) in this work, omitting calibration error and DM or WFS non-linearity for the rest of the thesis.

2.2.2 Shack-Hartmann Wavefront Sensor

There are many types of wavefront sensors used within AO, each with their strengths and weaknesses. This thesis focuses solely on the Shack-Hartmann Wavefront Sensor (SHWFS).

The SHWFS uses a lenslet array that is optically conjugate to the telescope pupil and divides it in subapertures. Each subaperture focuses a section of the incoming wavefront to form a spot, recorded for example by a Charged Coupled Device (CCD) positioned at the focal plane of all the lenslets (see the illustration in Fig. 2.5). All spots form a regular grid if the wavefront is unperturbed. In the presence of the atmospheric turbulence, there exists a local slope across each subaperture. This will deviate the centre of the spot, measured by centroid, from the central position and this amount of deviation is proportional to the local slope in the linear regime of WFS. The centroids in orthogonal directions within each subaperture can be calculated using centroiding algorithms from the focal plane

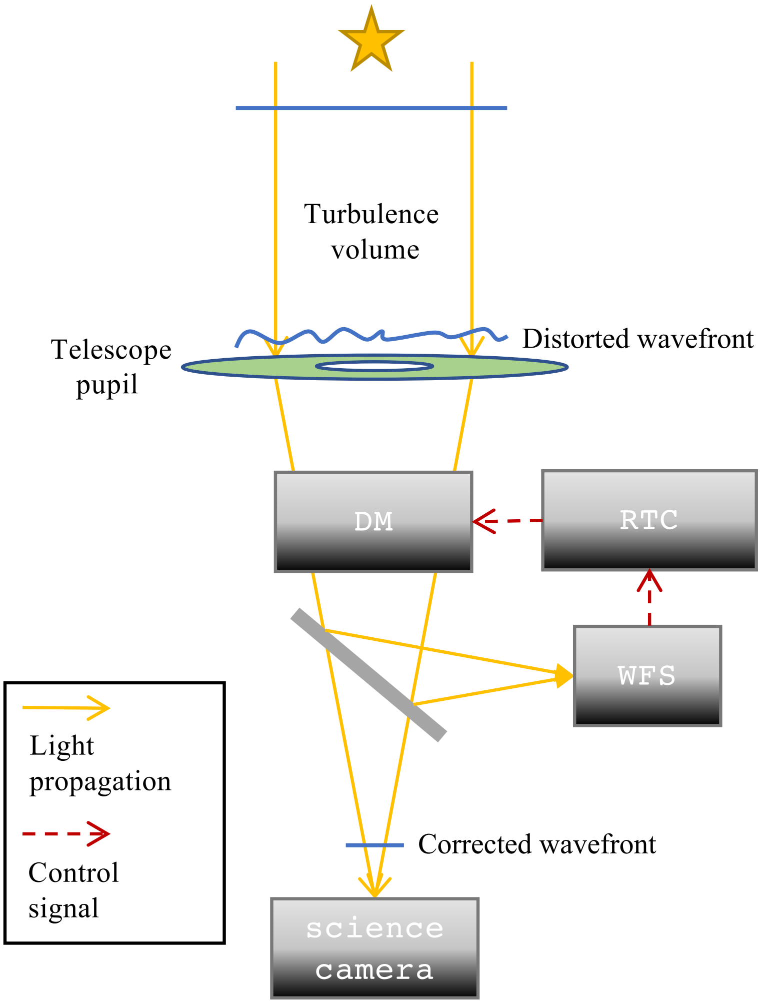


Figure 2.3: Diagram of a closed-loop SCAO system. In a closed-loop scheme, a wavefront measurement is made after a correction has been applied. In the feedback control loop, the RTC system converts the residual wavefront measured by a WFS to the driving signal of a DM. The DM is deformed in such a way that the wavefront distortion passing through it can be flattened. Mathematically, this is modelled as subtracting the DM shape from the incoming distortion.

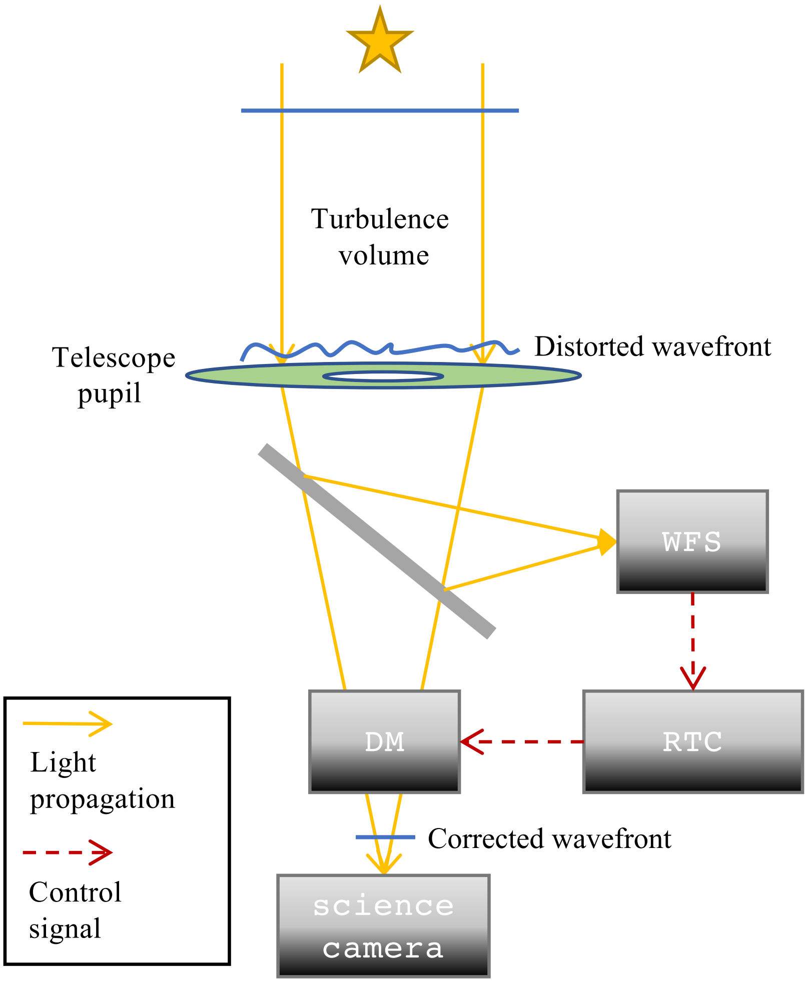


Figure 2.4: Diagram of an open-loop SCAO system. In an open-loop scheme, a wavefront measurement is made before the DM correction. There is no feedback in the control loop, adding difficulties to system calibration.

images (Platt and Shack, 2001). The wavefront can then be reconstructed from measured centroids from all subapertures (see Section 2.2.4)

The Centre of Gravity (CoG) algorithm represents the centroid (in number of CCD pixels) as a weighted average,

$$s_{x,\mathrm{CoG}} = \frac{\sum_{x,y} xI(x,y)}{\sum_{x,y} I(x,y)}, \tag{2.22}$$

with the weight $I(x,y)$ being the intensity of the subaperture image at CCD location (x,y). The summation is across all pixels within a single subaperture. s_y can be calculated by replacing x with y in the above equation.

The improved Thresholded Centre of Gravity (TCoG) algorithm is designed to suppress WFS noise (detailed in Section 2.2.6.4) for medium-flux observations (Arines and Ares, 2002; Thomas et al., 2006) by choosing brighter pixels for centroiding,

$$s_{x,\mathrm{TCoG}} = \frac{\sum_{I>I_T} x(I - I_T)}{\sum_{I>I_T}(I - I_T)}, \tag{2.23}$$

where I_T is the intensity threshold. It is set to $I_T = TI_{\mathrm{max}}$, where T is the thresholding coefficient between 0 and 1 and I_{max} is the maximum pixel value within one subaperture. T is usually fixed across subapertures. The summation is only across pixels with an intensity larger than I_T. A larger T reduces the amount of noise in the centroid calculation, but might introduce the truncation effect since only a fraction of pixels are considered, which can in turn introduce nonlinearities and additional measurement errors. Hence, there is a compromise in optimising T as a function of photon count.

Another thresholding method is the brightest pixel selection algorithm (Thomas et al., 2006; Basden et al., 2011). Similar to the TCoG algorithm, it involves selecting the N brightest pixels in each subaperture, setting all pixels below this level to 0, and subtracting from those selected pixels the $N + 1$th brightest pixel value. After the selection and subtraction, a standard centroiding method such as CoG proceeds. The benefit of this algorithm is that the threshold can be modified based on the actual intensity of pixels within the WFS image. From simulations based on the CANARY instrument, which is an AO demonstrator on the 4.2 m William Herschel Telescope (WHT) on La Palma, Basden et al. (2011) found that setting N to 20 is optimal (a trade-off between slope estimation accuracy and the linearity of measurements) for average seeing, while more pixels should be used for poorer seeing, dependant on the WFS Signal-to-Noise Ratio (SNR) level.

2.2.3 Deformable Mirrors

The DM is used for correcting the aberrations, ideally flattening the incoming wavefront. These are usually continuous-surface mirrors which can be mechanically

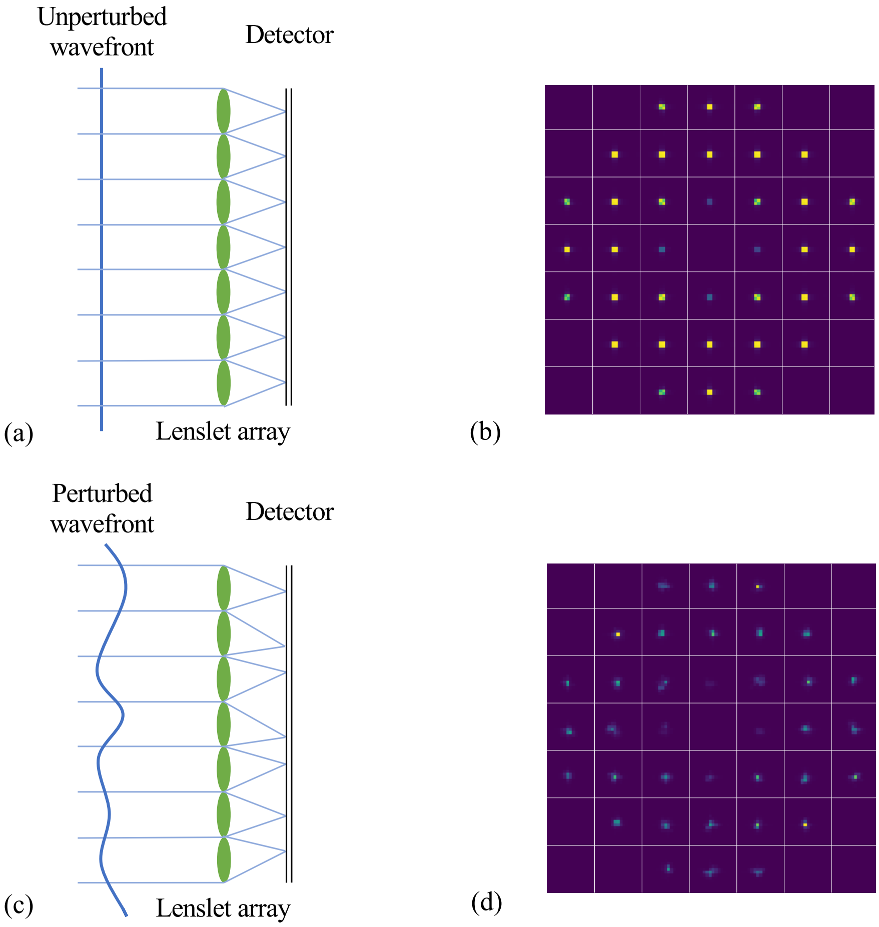


Figure 2.5: An illustration of the SHWFS. Each subaperture measures the local slopes in orthogonal directions from its focal plane images, from which the wavefront can be reconstructed. (a) and (c) show the operation of the WFS when observing flat and perturbed wavefronts respectively. (b) and (d) show the corresponding images in the WFS focal plane with an annular telescope aperture. White lines denote subaperture boundaries. Subapertures around aperture edges are partially illuminated, causing the lack of intensity of these spots.

deformed by devices called actuators behind the mirror surface (Tyson, 2011). The stroke of an actuator is its largest possible displacement. Actuators push or pull the mirror into the desired shape given a set of commands determined by the RTC system. The shape formed on the mirror by the activation of each individual actuator is known as the influence function of the DM, which is greatly influenced by the actuator technology.

Simulations in Chapters 3 and 4 assume a DM with a regular square grid geometry. The influence function of such DMs can be approximated using a super-Gaussian function with nearly zero response at the adjacent actuators (Hardy, 1998), providing zonal corrections, making them compatible with local tilt wavefront sensors such as SHWFSs.

We assume a Zernike DM in some of the analyses (see Section 4.2.1 and Chapter 5). In this thesis, such a DM is not used for correcting the wavefront. Instead its interaction (detailed in the next section) with a WFS facilitates performance quantification. The influence functions of Zernike DMs are Zernike polynomials up to an order that is equivalent to the spatial resolution of the matching WFS (detailed in Section 4.2).

The phase compensation provided by the DM can be modelled as

$$\mathbf{\Phi}_{\mathrm{DM}} = \sum_i c_i \mathbf{M}_{\mathrm{inf}}^i, \tag{2.24}$$

where c_i and $\mathbf{M}_{\mathrm{inf}}^i$ are respectively the command and the influence function of the i^{th} actuator.

2.2.4 Wavefront Reconstruction

The RTC uses reconstruction algorithms to reconstruct DM commands from WFS measurements in real time. The fundamental algorithm is the Matrix Vector Multiplication (MVM) approach which utilises the linear interaction matrix $\mathbf{M}_{\mathrm{int}}$ between the DM and the WFS.

The most common means of measuring $\mathbf{M}_{\text{int}}$ is to record the influence of each actuator on the measured wavefront during system calibration. $\mathbf{M}_{\text{int}}$ describes in each row the corresponding WFS measurements for a unit activation of each DM actuator. As a result, given the DM command vector $\boldsymbol{c}$ and the linear WFS-DM interaction, the WFS measurements should be

$$\boldsymbol{s} = \mathbf{M}_{\text{int}}\boldsymbol{c}. \tag{2.25}$$

During AO operations, the least-squares estimate of $\boldsymbol{c}$ given $\boldsymbol{s}$ can be computed using

$$\boldsymbol{c} = \mathbf{M}_{\text{rec}}\boldsymbol{s}, \tag{2.26}$$

where $\mathbf{M}_{\text{rec}}$ is the reconstruction, or control matrix. $\mathbf{M}_{\text{rec}}$ is the pseudo-inverse of $\mathbf{M}_{\text{int}}$, meaning $\mathbf{M}_{\text{rec}}\mathbf{M}_{\text{int}} = \mathbf{I}$, $\mathbf{I}$ being the identity matrix. The pseudo-inverse of $\mathbf{M}_{\text{int}}$ is usually obtained via singular value decomposition. It is possible that the DM can form shapes which the WFS will sense poorly. These DM modes correspond to small singular values of $\mathbf{M}_{\text{int}}$ and can degrade the reconstruction. One way to mitigate this is to set small singular values below a threshold to zero during the inversion process. Adjusting this threshold degrades the pseudo-inversion, but can make the system more stable, as it will not attempt to correct poorly sensed wavefront modes that may be dominated by WFS noise.

In a real system, to maintain loop stability, $\boldsymbol{c}$ is not directly applied. A basic integrator with gain g maintains a running average of $\boldsymbol{c}$ (see the next section).

2.2.5 Time Lines for SCAO

The operating frequency of an AO system, f_S, is usually the inverse of the effective WFS integration time, T, during which time enough photons are collected for wavefront sensing. The discrete nature of the RTC in a closed-loop system is illustrated in Fig. 2.6. During frame $[(k-2)T,\ (k-1)T]$, the residual wavefront is integrated by the WFS, yielding $\boldsymbol{\phi}^{\text{res}}_{k-1}$. After the WFS exposure, it takes another frame for the CCD readout and centroiding, the generation of the measured slope $\boldsymbol{s}_k^{\text{res}}$, and the calculation of the DM command $\boldsymbol{c}_k$. In the classical closed-loop AO control scheme this can be modelled as an integrator whose gain is noted g, and the control signal is given by

$$\boldsymbol{c}_k = \boldsymbol{c}_{k-1} + g\mathbf{M}_{\text{rec}}\boldsymbol{s}_k^{\text{res}}. \tag{2.27}$$

This can be understood as the sum of an applied correction $\boldsymbol{c}_{k-1}$ and a weighted residual correction $\mathbf{M}_{\text{rec}}\boldsymbol{s}_k^{\text{res}}$ determined using Eq. 2.26. g is for maintaining the closed-loop stability, typically between 0 and 1. The control is then applied on the next frame $[kT,\ (k+1)T]$. Here we omit the DM settling time taken for the DM to arrive at the desired shape, which can usually be safely neglected.

When the system is controlled in open loop, the time line shown in Fig. 2.6 is still valid. The main difference is that the WFS samples the uncompensated wavefront ϕ_{k-1}^{atm} instead of the residual ϕ_{k-1}^{res}. The control signal then becomes

$$\boldsymbol{c}_k = (1-g)\boldsymbol{c}_{k-1} + g\mathbf{M}_{\text{rec}}\boldsymbol{s}_k^{\text{OL}}, \tag{2.28}$$

where $\boldsymbol{s}_k^{\text{OL}}$ is the slope measurement of ϕ_{k-1}^{atm}.

This AO loop shown in Fig. 2.6 has a 1.5-frame latency, which is the time delay between the mid-point of the WFS exposure and the beginning of the application of DM correction. Overall system latency is typically dominated by the time taken for computations within the RTC system, with the most challenging Extreme Adaptive Optics (XAO) systems requiring sub-frame latency.

In a discrete, simulated system however, the physical processes shown in Fig. 2.6 are all assumed instantaneous. For example, as is shown in Fig. 2.7, the WFS samples the wavefront at the discrete timestep of $(k-1)T$. The integration time is considered infinitesimal. Slope and DM command calculation can be considered to take place between time steps $(k-1)T$ and kT. At kT, the DM correction is applied to the wavefront distortion at that instance. The average delay is thus one frame, which is the time between the 'mid-point' of the 'integration' and the availability of

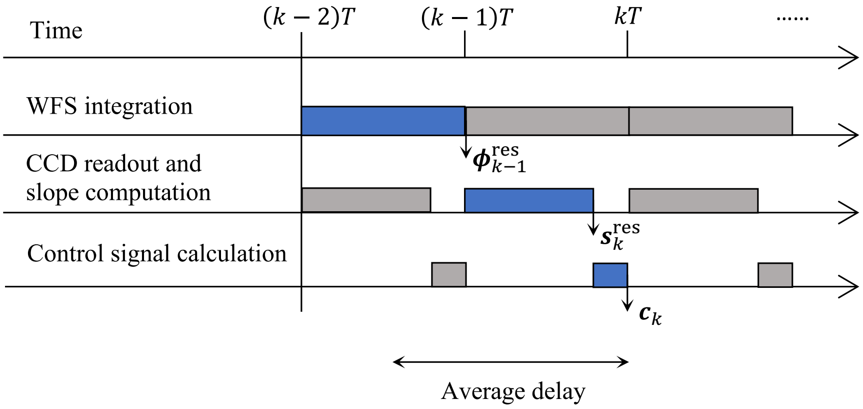

Time
$(k-2)T$
$(k-1)T$
kT
......
WFS integration
$\boldsymbol{\phi}_{k-1}^{\mathrm{res}}$
CCD readout and slope computation
$\boldsymbol{s}_{k}^{\mathrm{res}}$
Control signal calculation
$\boldsymbol{c}_k$
Average delay

Figure 2.6: Typical time line for a continuous closed-loop SCAO system. Here, the delay between the mid-point of the WFS integration and the application of the corresponding DM commands is about 1.5 frames due to time taken for the WFS exposure, CCD readout, calculation of slopes and control signals.

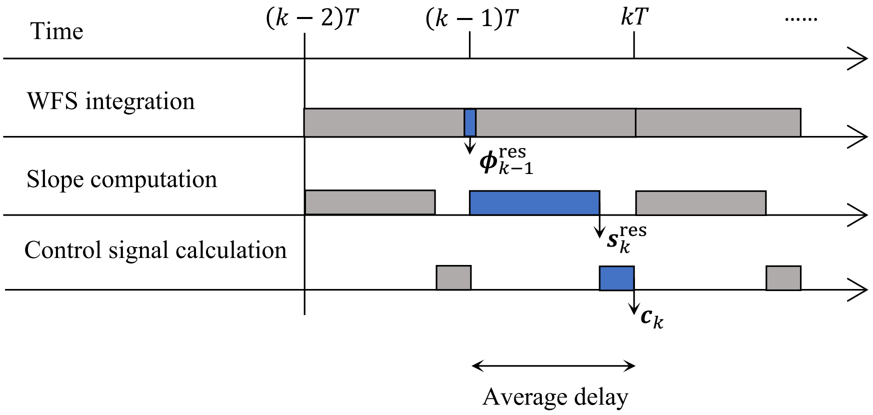

Time
$(k-2)T$
$(k-1)T$
kT
......
WFS integration
$\boldsymbol{\phi}_{k-1}^{\mathrm{res}}$
Slope computation
$\boldsymbol{s}_{k}^{\mathrm{res}}$
Control signal calculation
$\boldsymbol{c}_k$
Average delay

Figure 2.7: Typical time line for a discrete, simulated closed-loop SCAO system. Here, the WFS integration and DM correction is assumed instantaneous, taking place at discrete time steps. The average loop latency is one frame, accounting only for the time taken for slope and DM command calculation.

the DM commands. When the loop latency is extended to two frames, this means the slope and DM command calculation is considered to take up two frames.

2.2.6 Error Sources in SCAO

The performance of an SCAO system can be measured in varied ways for different purposes. This thesis mainly examines the RMS residual Wavefront Error (WFE), which we label as σ.

The sources of error in an AO system that contribute to the error variance, σ^2, depend on both the property of the atmosphere and the components of the system. For the SCAO system studied here, these are the temporal error, fitting error, WFS noise error and aliasing error. Assuming the contribution from individual sources are uncorrelated, σ^2 can be estimated using

$$\sigma^2 = \sigma^2_{\text{temporal}} + \sigma^2_{\text{fitting}} + \sigma^2_{\text{noise}} + \sigma^2_{\text{aliasing}}. \tag{2.29}$$

Eq. 2.29 is widely used in AO system design. In practice, there are correlations between error terms (especially spatial errors). This will lead to an overestimate of σ^2.

2.2.6.1 Temporal Error

Temporal errors are caused by the latency between wavefront sensing and correction as has been shown in Fig. 2.6, during which time the turbulence has evolved.

As is detailed in Section 2.1.2, for small time scales such as the crossing time of the turbulence across the telescope aperture due to the wind, the temporal evolution of the turbulence is dominated by the frozen flow translation. As a result, the temporal error due to pure loop delay is equivalent to the temporal phase structure function $D_\phi(\tau)$. Combining Eqs. 2.9 and 2.11 we have,

$$\sigma^2_{\text{temporal}} = \left(\frac{\tau_d}{\tau_0}\right)^{5/3}, \tag{2.30}$$

where τ_d is the loop delay in seconds.

2.2.6.2 Fitting Error

For a DM with zonal correction elements such as the piezoelectric DM (Hardy, 1998), the finite number of these elements (thus a finite degree of freedom) means that the DM is incapable of perfectly fitting an arbitrary shape of the wavefront, while there is no limit to the spatial frequency of a well-developed Kolmogorov turbulence as the power spectrum suggests. This sets a fundamental limit to the system. The fitting error is given by

$$\sigma^2_{\text{fitting}} = a_F \left(\frac{d_{\text{DM}}}{r_0}\right)^{5/3}, \tag{2.31}$$

where d_{DM} is the actuator spacing. The coefficient a_F is determined by the DM type. For a continuous phase sheet DM with super-Gaussian influence functions, a_F is between 0.28 and 0.34 rad^2 (Hardy, 1998).

2.2.6.3 Aliasing Error

Similar to the fitting error, aliasing is a phenomenon associated with spatial sampling caused by the finite number of sensing elements (e.g. the subapertures in a SHWFS) in the WFS (Poyneer and Macintosh, 2004). High-spatial frequencies in the turbulence that are above the Nyquist frequency of the WFS will be misinterpreted as low-frequency signals. This will lead to an error in the reconstruction and the resulting system performance. The aliasing error depends on the characteristics of the turbulence as well as the spatial resolution of the WFS and the control law (Kulcsár et al., 2012). Rigaut et al. (1998) shows that σ_{aliasing} can be estimated in the case of a continuous phase sheet DM using

$$\sigma^2_{\text{aliasing}} = 0.08 \left(\frac{d_{\text{DM}}}{r_0}\right)^{5/3}. \tag{2.32}$$

.

2.2.6.4 Noise Error

For a SHWFS, the focal plane image $\mathbf{I}_{\text{WFS}}$ is randomly contaminated by Poissonian photon noise and Readout Noise (RON),

$$\mathbf{I}_{\text{WFS}} = \mathcal{P}(\mathbf{I}_{\text{WFS}}) + n_e, \tag{2.33}$$

where $\mathcal{P}$ denotes a Poisson process and n_e denotes the RMS RON in number of electrons per CCD pixel, which can be modelled as a zero-mean Gaussian distribution. WFS noise, especially photon noise, is unavoidable and determines the precision of the slope measurements.

The noise-induced slope variance can be estimated from measured data by a SHWFS using the approach described in Gendron and Léna (1995), which is based on the temporal auto-correlation of open-loop slope measurements. The auto-correlation is defined as

$$\mathcal{A}_j(\Delta n) = \frac{1}{n_t - \Delta n} \sum_k s_j(k) \times s_j(k + \Delta n), \tag{2.34}$$

where j is the index of a subaperture, n_t is the length of the slope sequence in frames. It can be proven from the above equation that $\mathcal{A}$ is an even function.

Fig. 2.8 shows the auto-correlation of an open-loop slope sequence (10,000 frames) taken by the CANARY instrument in the x direction of a single WFS subaperture. At $\Delta n = 0$, the auto-correlation is the sum of the variances of the turbulence and the noise, $\mathcal{A}_j(0) = s^2_{\text{turb},j} + s^2_{\text{noise},j}$. When $\Delta n \neq 0$, because the noise is not correlated with itself over time, $\mathcal{A}_j(\Delta n)$ is the auto-correlation of the turbulence only, which degrades with time and can be approximated at small Δn by a parabola. The parabolic approximation was justified mathematically in Gendron and Léna (1995). It was verified with on-sky data by the correlation of noise variance measured by this approach and by a different approach based on the slope variance (Gendron and Léna, 1994), which shows the reliability of both methods with different principles. This parabola can be obtained by fitting $\mathcal{A}_j(1)$ and $\mathcal{A}_j(2)$. The fitted value at $\Delta n = 0$ then gives the slope variance induced by the turbulence only, $s^2_{\text{turb},j}$. Subtracting this value from $\mathcal{A}_j(0)$, we obtain $s^2_{\text{noise},j}$. This approach also provides a way of defining the noise as a component that is uncorrelated through time, which we will use to identify and quantify the noise in the predicted slopes.

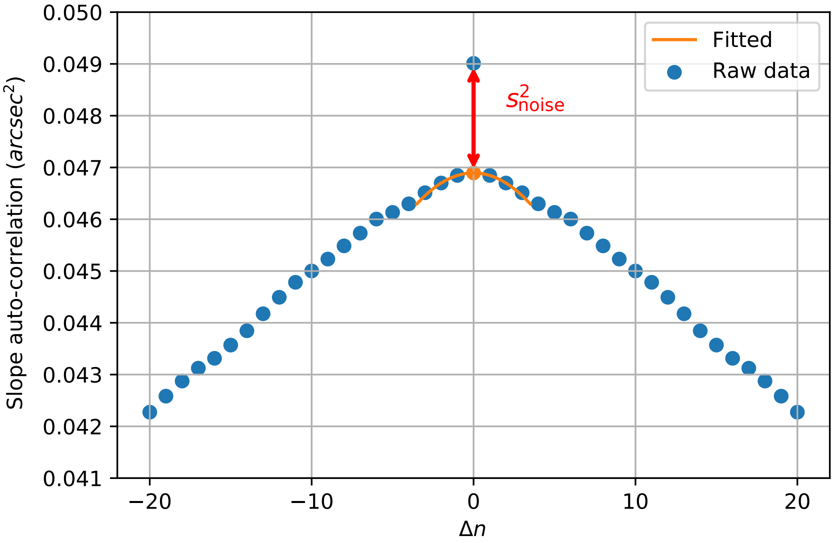


Figure 2.8: Time-lagged auto-correlation of an open-loop slope sequence in the x direction of the 5th WFS subaperture. At $\Delta n = 0$, the auto-correlation is the sum of the covariances of the turbulence and WFS noise. At small Δn, since the noise is uncorrelated over time, the auto-correlation contains the turbulence contribution only, which can be approximated by a parabola. The fitted parabola at $\Delta n = 0$ gives the turbulence variance. Subtracting this from the auto-correlation at $\Delta n = 0$ then gives the noise variance.

2.3 AO Simulation with Soapy

Numerical simulations of the turbulence and AO systems as a whole provide a powerful tool for analysing and predicting the performance of a given system, or validating new ideas and implementations. AO simulations within this thesis are performed using Soapy (Simulation 'Optique Adaptative' with Python) (Reeves, 2016). Soapy is a Monte-Carlo AO simulation tool written in the Python programming language. This section introduces modules of Soapy, with a focus on the principle of atmospheric turbulence simulation adopted by Soapy.

2.3.1 Layered Atmosphere Model

In many AO simulations, the atmospheric turbulence is treated as a finite number of infinitesimally thin, discrete and independent turbulence layers (Roddier, 1981).

This mathematical simplification has been grounded by many observations (Wang et al., 2008; Poyneer et al., 2009; Wilson et al., 2009), where the turbulence is found to exist in distinct thin layers at different heights, each translating with an independent wind vector in the case of frozen flow.

Under this assumption, Eq. 2.4 reduces to

$$r_0 = \left(0.423k^2 \sec\zeta \sum_i C_{n_i}^2 \Delta z_i\right)^{-3/5}, \tag{2.35}$$

where C_{n_i} and Δz_i are respectively the refractive index structure constant and thickness of the i^{th} layer. The effective strength of the i^{th} layer is defined as (Roggemann et al., 1995)

$$r_{0_i} = [0.423k^2 \sec\zeta C_{n_i}^2 \Delta z_i]^{-3/5}. \tag{2.36}$$

The fraction of the i^{th} layer in the total integrated strength, also known as the relative strength, is then

$$\rho_i = \frac{C_{n_i}^2 \Delta z_i}{\sum_i C_{n_i}^2 \Delta z_i}. \tag{2.37}$$

The relative strength satisfies $\sum_i \rho_i = 1$. From Eqs. 2.35-2.37 we have

$$r_{0_i} = \rho_i^{-3/5} r_0, \tag{2.38}$$

or r_0 can be determined from r_{0_i},

$$r_0 = \left(\sum r_{0_i}^{-5/3}\right)^{-3/5}. \tag{2.39}$$

The relationship between $\bar{v}$ (defined in Eq. 2.10) and the wind speed of each layer v_i can be derived in a similar fashion. Combining Eqs. 2.10 and 2.37, we have

$$\bar{v} = \left(\sum \rho_i v_i^{5/3}\right)^{3/5}. \tag{2.40}$$

In Soapy implementation, the optical effects of each layer is modelled as a phase screen. After individual properties of each layer are determined, the corresponding phase screen can be generated given r_{0_i} (detailed in the following section), which then translate with v_i, with different portions of the screen sampled by the telescope at each discrete timestep. The wavefront aberration ϕ at the telescope pupil is the sum from all layers along the propagation path.

2.3.2 Monte-Carlo Phase Screens

Turbulence-induced phase aberrations are a random process with a well-defined power spectrum (see Section 2.1.1). To generate a random realisation of such a process on a discrete two-dimensional sample grid, it is a common practice to filter a white Gaussian noise with the square root of the power spectrum in the frequency domain, and then transform the filtered spectrum to the spatial domain via Fast Fourier Transform (FFT) (McGlamery, 1976; Johansson and Gavel, 1994; Ellerbroek and Cochran, 2002).

With this approach, we can represent a discrete two-dimensional phase screen, $\phi(x,\ y)$, given its strength r_0, as

$$\phi(x,y) = \sum_{\kappa_x}\sum_{\kappa_y} g(\kappa_x,\kappa_y)\sqrt{F_\phi(\kappa_x,\kappa_y)}e^{i2\pi(\kappa_x x+\kappa_y y)}\Delta\kappa_x\Delta\kappa_y. \tag{2.41}$$

The spatial domain sample points x and y are $x = m\Delta x$ and $y = n\Delta y$, where $\Delta x = \Delta y = L/N$. L is the physical size of the square phase screen on each side, and N is the number of pixels along L. Usually N is a power of 2 for the ease of FFT implementation. The wavenumbers κ_x and κ_y are given by $\kappa_x = k\Delta\kappa_x$ and $\kappa_y = l\Delta\kappa_y$, where $\Delta\kappa_x = \Delta\kappa_y = 2\pi/L$. $g(\kappa_x,\kappa_y)$ is defined as

$$g(\kappa_x,\kappa_y) = \frac{g'(k,l)}{\sqrt{\Delta\kappa_x\Delta\kappa_y}}, \tag{2.42}$$

where g' is a complex Hermitian noise matrix, both the real and imaginary parts of which follow an independent zero-mean Gaussian distribution with a standard deviation of $1/\sqrt{2}$. Under von Kármán statistics,

$$F_\phi(\kappa_x,\kappa_y) = 0.490\ r_0^{-5/3}(\kappa^2+\kappa_0^2)^{-11/6}, \tag{2.43}$$

where $\kappa = \sqrt{\kappa_x^2+\kappa_y^2}$, $\kappa_0 = 2\pi/L_0$. This expression is consistent with Eq. 2.8, with a scaling factor being $(2\pi)^2(2\pi)^{-11/3}$ due to the conversion between spatial frequency κ and spatial wavenumber f, i.e. $\kappa = 2\pi f$.

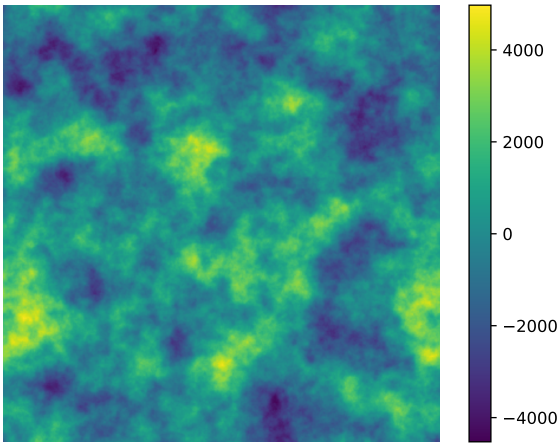


Figure 2.9: A sample phase screen (of size 1024 × 1024 pixels) showing phase variations (in nm) with von Kármán statistics generated using the FFT method.

Converting from wavenumber space to spatial frequency domain and combining Eqs. 2.41-2.43, we obtain

$$\begin{aligned}\phi(m,n) &= \sum_{k=-N/2}^{N/2-1} \sum_{l=-N/2}^{N/2-1} g'(k,l)h(k,l)e^{i2\pi(\frac{mk+nl}{N})} \\ &= N^2\mathcal{F}^{-1}[g'(k,l)h(k,l)],\end{aligned} \tag{2.44}$$

where $\mathcal{F}^{-1}$ denotes the inverse Fourier transform, with N^2 added to match the discrete numerical implementation. $h(k,l)$ is the turbulence spatial filter,

$$h(k,l) = \frac{0.1513}{L}\, r_0^{-5/6}(f^2 + f_0^2)^{-11/12}, \tag{2.45}$$

where $f_0 = \kappa_0/2\pi$. $h(0,0)$, corresponding to the direct component of the phase screen which has no effect on image formation, is set to zero. Both the real and imaginary part of ϕ are realisations of a von Kármán phase screen. Usually the real part is taken.

A sample phase screen generated using this method is shown in Fig. 2.9. $L = 67.2$ m and $N = 1024$. r_0 is 0.16 m. L_0 is 25 m.

The minimum and maximum spatial frequencies represented by this method are $f_{\min} = 1/L$ and $f_{\max} = \frac{\sqrt{2}}{2}N/L$. Thus, the maximum and minimum scale sizes represented are $s_{\max} = L$ and $s_{\min} = \sqrt{2}L/N$ respectively. For $L = 67.2$ m and $N = 1024$, we have $s_{\max} = 67.2$ m and $s_{\min} = 0.0928$ m. For typical inner (a few millimeters) and outer (tens of meters) scales of turbulent eddies, this means either the low- or high-frequency component of turbulence might be poorly sampled for trading off the computation requirements. While the improper sampling of the inner scale is relatively negligible (Johansson and Gavel, 1994), the generated phase screen is normally made much larger than the telescope aperture size in order to sufficiently sample the outer scale (Herman and Strugala, 1990). This also guarantees that the phase screen is large enough for the translation needed during a simulation run.

2.3.3 Soapy Modules

Soapy can be used for end-to-end simulations where the entire system parameters are defined in a configuration file. The real power lies in its modular nature. Each AO component is modelled as a Python object with intrinsic attributes and methods. Each object can be configured from the related block of parameters in the configuration file. This enables fast implementations and novel configurations using parts of the modules, from which this thesis has benefited hugely. New concepts implemented as a Python object can also be easily integrated. Below we introduce principal Soapy modules and related attributes and methods we have used. A simulated SCAO system built from these modules will be introduced in Section 3.3.

2.3.3.1 `Atmosphere`

The `atmosphere` class generates a given number of phase screens (the number of turbulence layers) with von Kármán statistics using the FFT method described in Section 2.3.2. The strength of each layer r_{0_i} can be determined using Eq. 2.38 from its fractional strength ρ_i and the integrated r_0 given in the configuration file. A phase screen generated using the FFT method is periodic and continuous across its opposite edges. The screen size should be at least 10 times of L_0 to better sample large spatial components. This also guarantees that the duration of the simulation is satisfied. Under the frozen flow hypothesis, the `moveScrns` method can be called on each timestep to move the phase screens forward, the size of which is determined by the system frequency and the wind speed of that layer, converted to pixels given the pixel scale (m/pixel) of the simulation. Sub-pixel movements are represented by bilinear interpolation in the phase screens. The phase aberration seen by the telescope at that timestep is the sum of the resulting sampled smaller screens from all layers.

2.3.3.2 `SHWFS`

The `SHWFS` class used in this thesis inherits from the base `WFS` class. The `frame` method calculates slopes from the input phase aberration, which calls three methods in sequence. The `calcFocalPlane` method computes individual focal plane image focused by each subaperture using Eq. 2.18. The `makeDetectorPlane` method combines all these images and fits them back into a single detector plane image as is shown in Fig. 2.5 (b) and (d). Desired amount of noise is added using Eq. 2.33. The `calculateSlopes` method calculates slopes across the subaperture from the detector image given the centroiding method. We mainly use TCoG and brightest pixel selection centroiding methods described in Section 2.2.2.

2.3.3.3 `DM`

The `DM` base class has a `dmFrame` method that computes the DM shape using Eq. 2.24 given its influence functions and the input DM commands. The influence function for each actuator is determined by the `makeIMatShapes` method defined within each subclass inheriting from the `DM` base. For the `Piezo` subclass simulating a piezoelectric DM, the influence function of an actuator is initialised as an $n \times n$

matrix where n is the number of actuators across telescope pupil, with only this actuator set to 1 and the rest to 0. This matrix is then bicubicly interpolated to the simulation size (number of pixels representing the pupil). For the `Zernike` subclass representing a Zernike DM, the influence functions are a set of Zernike polynomials up to a given order.

2.3.3.4 `Reconstructor`

The `reconstructor` base class has a `makeIMat` method that measures the interaction matrix between a given WFS and DM pair. Each row in the interaction matrix is the resulting WFS measurements given an individual DM influence function (multiplied by an activation value), representing the influence on the WFS measurement of a single activation. The `makeCMat` method calls the `calcCMat` method within each subclass to calculate the control matrix. For the `MVM` subclass, the `calcCMat` method calculates the pseudo inversion of the interaction matrix. Truncation of singular values involved in this process has been explained in Section 2.2.4. The `reconstruct` method outputs DM commands given measured slopes from `WFS` using Eq. 2.26.

2.4 References

J. Arines and J. Ares. Minimum variance centroid thresholding. *Opt. Lett.*, 27(7): 497–499, 2002.

A. G. Basden, R. M. Myers, and E. Gendron. Wavefront sensing with a brightest pixel selection algorithm. *Monthly Notices of the Royal Astronomical Society*, 419(2):1628–1636, 2011.

B. L. Ellerbroek and G. Cochran. Wave optics propagation code for multiconjugate adaptive optics. In R. K. Tyson, D. Bonaccini, and M. C. Roggemann, editors, *Adaptive Optics Systems and Technology II*, volume 4494, pages 104–120. International Society for Optics and Photonics, SPIE, 2002.

B. L. Ellerbroek and C. R. Vogel. Simulations of closed-loop wavefront reconstruction for multiconjugate adaptive optics on giant telescopes. In *Astronomical Adaptive Optics Systems and Applications*, volume 5169, pages 206–217. SPIE, 2003.

D. L. Fried. Statistics of a geometric representation of wavefront distortion. *J. Opt. Soc. Am.*, 55(11):1427–1435, 1965.

E. Gendron and P. Léna. Astronomical adaptive optics. i. modal control optimization. *Astronomy and Astrophysics*, 291:337–347, 1994.

E. Gendron and P. Léna. Astronomical adaptive optics. ii. experimental results of an optimized modal control. *Astronomy and Astrophysics Supplement*, 111: 153–167, 1995.

L. Gilles. Closed-loop stability and performance analysis of least-squares and minimum-variance control algorithms for multiconjugate adaptive optics. *Appl. Opt.*, 44(6):993–1002, 2005.

J. Hardy. *Adaptive Optics for Astronomical Telescopes.* Oxford Series in Optical & Imaging Sciences. Oxford University Press, 1998.

B. J. Herman and L. A. Strugala. Method for inclusion of low-frequency contributions in numerical representation of atmospheric turbulence. In P. B. Ulrich and L. E. Wilson, editors, *Propagation of High-Energy Laser Beams Through the Earth's Atmosphere*, volume 1221, pages 183–192. International Society for Optics and Photonics, SPIE, 1990.

E. M. Johansson and D. T. Gavel. Simulation of stellar speckle imaging. In J. B. Breckinridge, editor, *Amplitude and Intensity Spatial Interferometry II*, volume 2200, pages 372–383. International Society for Optics and Photonics, SPIE, 1994.

A. Kolmogorov. The Local Structure of Turbulence in Incompressible Viscous Fluid for Very Large Reynolds Numbers. *Akademiia Nauk SSSR Doklady*, 30:301–305, 1941.

C. Kulcsár, H.-F. Raynaud, C. Petit, and J.-M. Conan. Minimum variance prediction and control for adaptive optics. *Automatica*, 48(9):1939–1954, 2012.

B. L. McGlamery. Computer Simulation Studies Of Compensation Of Turbulence Degraded Images. In J. C. Urbach, editor, *Image Processing*, volume 0074, pages 225–233. International Society for Optics and Photonics, SPIE, 1976.

R. J. Noll. Zernike polynomials and atmospheric turbulence∗. *J. Opt. Soc. Am.*, 66(3):207–211, 1976.

Y. H. Ono, C. M. Correia, D. R. Andersen, O. Lardière, S. Oya, M. Akiyama, K. Jackson, and C. Bradley. Statistics of turbulence parameters at maunakea using the multiple wavefront sensor data of raven. *Monthly Notices of the Royal Astronomical Society*, 465(4):4931–4941, 2017.

B. Platt and R. Shack. History and principles of shack-hartmann wavefront sensing. *Journal of refractive surgery (Thorofare, N.J. : 1995)*, 17(5):S573–7, 2001.

L. A. Poyneer and B. Macintosh. Spatially filtered wave-front sensor for high-order adaptive optics. *J. Opt. Soc. Am. A*, 21(5):810–819, 2004.

L. A. Poyneer, M. van Dam, and J.-P. Véran. Experimental verification of the frozen flow atmospheric turbulence assumption with use of astronomical adaptive optics telemetry. *J. Opt. Soc. Am. A*, 26(4):833–846, 2009.

A. Reeves. Soapy: an adaptive optics simulation written purely in python for rapid concept development. In *Proc. of SPIE 9909*, volume 9909, page 99097F. International Society for Optics and Photonics, 2016.

F. J. Rigaut, J.-P. Veran, and O. Lai. Analytical model for Shack-Hartmann-based adaptive optics systems. In *Adaptive Optical System Technologies*, volume 3353, pages 1038–1048. SPIE, 1998.

F. Roddier. The effects of atmospheric turbulence in optical astronomy. *Progess in Optics*, 19:281–376, 1981.

B. M. Roggemann, D. Montera, and T. A. Rhoadarmer. Method for simulating atmospheric turbulence phase effects for multiple time slices and anisoplanatic conditions. *Applied Optics*, 34(20):4037–4051, 1995.

J. W. Strutt. Investigations in optics, with special reference to the spectroscope. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 8(49):261–274, 1879.

V. I. Tatarskii. *Wave Propagation in Turbulent Medium.* McGraw-Hill, 1961.

G. I. Taylor. The spectrum of turbulence. *Proceedings of the Royal Society A*, 164 (919):476–490, 1938.

S. Thomas, T. Fusco, A. Tokovinin, M. Nicolle, V. Michau, and G. Rousset. Comparison of centroid computation algorithms in a Shack–Hartmann sensor. *Monthly Notices of the Royal Astronomical Society*, 371(1):323–336, 2006.

R. K. Tyson. *Principles of Adaptive Optics (Third Edition).* CRC Press, 2011.

L. Wang, M. Schöck, and G. Chanan. Atmospheric turbulence profiling with slodar using multiple adaptive optics wavefront sensors. *Appl. Opt.*, 47(11):1880–1892, 2008.

R. W. Wilson, T. Butterley, and M. Sarazin. The Durham/ESO SLODAR optical turbulence profiler. *Monthly Notices of the Royal Astronomical Society*, 399(4): 2129–2138, 2009.

A. Ziad, M. Schöck, G. A. Chanan, M. Troy, R. Dekany, B. F. Lane, J. Borgnino, and F. Martin. Comparison of measurements of the outer scale of turbulence by three different techniques. *Appl. Opt.*, 43(11):2316–2324, 2004.

Chapter 3

ANN Training for Wavefront Prediction

3.1 Overview

This chapter mainly address two questions: how to develop the Artificial Neural Network (ANN) training methodology for the wavefront prediction task and how does the training work?

We first provide necessary theoretical background for understanding ANN technologies used in this thesis. We then introduce the software modules and data flow of a simulated 7 × 7 Single-Conjugate Adaptive Optics (SCAO) system. We will describe how the simulated system can be used to generate ANN training data and how the ANN training and hyperparameter tuning process works.

3.2 Artificial Neural Networks

3.2.1 Supervised Learning

ANN is a computational model that can be used to solve many supervised learning tasks, where input-target pairs are provided for their underlying mapping to be learnt. This learning scheme is based on the fact that it is sometimes easier to obtain examples from the desired mapping than defining the mapping directly (Karpathy, 2016). Let X be an input space and Y be an output space. D is the data distribution over $X \times Y$. The goal of supervised learning task is to find the mapping f from X to Y based on a *training set* consisting of n independent and identically distributed (i.i.d.) samples representative of D, $S_{\text{train}} = \{(x_i, y_i)\}_{i=1}^{n}$, in order to minimise the expected loss error $E(L)$ over the data generating distribution D,

$$f^* = \operatorname*{argmin}_{f \in F} E_{(x,y) \sim D}[L(f(x), y)]. \tag{3.1}$$

L is a scalar-valued objective function. E represents the expectation of L evaluated on D. F is a class of functions which f is restricted to.

Because not all elements of D are known, the expected loss error becomes intractable, but can be approximated by the *training error* under the i.i.d. assumption,

$$\begin{aligned} f^* &\approx \operatorname*{argmin}_{f \in F} E_{(x,y) \sim S_{\text{train}}}[L(f(x), y)] \\ &= \operatorname*{argmin}_{f \in F} \frac{1}{n} \sum_{i=1}^{n} L(f(x_i), y_i). \end{aligned} \tag{3.2}$$

The expected value of the error on a new input in D but not in S_{train} is defined as the *generalisation error*, which is usually estimated by measuring the performance on a *test set* S_{test}. The goal of the training process (which focuses on reducing the training error only) is to achieve the lowest possible generalisation error.

When ANNs are used as the computational model for input-output mapping, F will be a neural network with given architecture but unknown parameters $\boldsymbol{\theta}$ to be determined from S_{train}. The learning problem in Eq. 3.2 reduces to an optimisation problem,

$$\begin{aligned} \boldsymbol{\theta} &= \operatorname{argmin} g(\boldsymbol{\theta}) \\ &= \operatorname{argmin} \frac{1}{n} \sum_{i=1}^{n} L(f_{\boldsymbol{\theta}}(x_i), y_i), \end{aligned} \tag{3.3}$$

which is referred to as *training* in the ANN context.

For this thesis, the supervised learning task will be to predict the future open-loop Wavefront Sensor (WFS) measurements given a time sequence of the immediate past WFS measurements. In this case, x_i is the WFS measurement history. f is the mapping defined by an ANN. $\boldsymbol{\theta}$ is the parameter of the ANN to be learnt. $f_{\boldsymbol{\theta}}(x_i)$ is the predicted measurement made by the ANN while y_i is the actual or targeted measurement. For L we will use the Mean Squared Error (MSE) metric. More details will be given in Section 3.4.1.

3.2.2 Optimisation with ANNs

3.2.2.1 Algorithms with First Order Derivatives

When g in Eq. 3.3 is differentiable, algorithms based on first order derivatives can be used to solve this optimisation problem. The *gradient descent* (GD) algorithm is such a method. It is based on the first order approximation of g, from which the gradient vector $\nabla_{\boldsymbol{\theta}} g$ can be seen as a direction along which g increases, thus the opposite direction along which g can be decreased. Gradient Descent (GD) alternates between two steps: 1) calculate $\nabla_{\boldsymbol{\theta}} g$ with *backpropagation* (detailed in Section 3.2.3) given the current value of $\boldsymbol{\theta}$ and 2) update $\boldsymbol{\theta}$ by taking a small step along $-\nabla_{\boldsymbol{\theta}} g$,

$$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \epsilon \nabla_{\boldsymbol{\theta}} g, \tag{3.4}$$

where ϵ is called the *learning rate.* If ϵ is too large, the optimisation may not converge or even diverge. If ϵ is too small, the optimisation process will be too slow or converge to a local minimum.

In practice, $\nabla_{\boldsymbol{\theta}} g$ is approximated as the average gradient vector across the training set. This leads to the *batch gradient descent* algorithm. Batch GD is promised to reach the global minima with a well tuned learning rate. However, because gradients calculated from every training sample have to be computed at each optimisation step, batch GD is computationally expensive. A much more computationally efficient algorithm when the training set size is huge is the *stochastic gradient descent* algorithm, which evaluates a single example at each step. Compared with the batch approach, stochastic GD provides a much noisier estimate of $\nabla_{\boldsymbol{\theta}} g$, preventing the optimisation from converging to a local minima (LeCun et al., 1998). The *minibatch gradient descent* algorithm balances between finding the global minimum and avoiding the local minima by evaluating several samples simultaneously (called a minibatch) at each step. The size of the minibatch is known as the *batch size*. An *epoch* is the process during which all training samples have been evaluated once. For example, for a training set of 100,000 samples and a batch size of 100, $\boldsymbol{\theta}$ is updated once during one epoch with batch GD, 1,000 times with minibatch GD and 100,000 times with stochastic GD.

Many advanced algorithms exist that can accelerate convergence by modifying the update direction based on $\nabla_{\boldsymbol{\theta}} g$ (Goodfellow et al., 2016). For example, the addition of a *momentum* term acts as the inertia of the motion of the algorithm in the parameter space (Sutskever et al., 2013), guaranteeing that the optimisation is along consistent instead of zigzag directions in some cases,

$$\begin{aligned} \boldsymbol{v} &\leftarrow \alpha \boldsymbol{v} - \epsilon \nabla_{\boldsymbol{\theta}} g \\ \boldsymbol{\theta} &\leftarrow \boldsymbol{\theta} + \boldsymbol{v}. \end{aligned} \tag{3.5}$$

$\boldsymbol{v}$ (initialised as $\mathbf{0}$) is analogous to velocity in physics and becomes momentum if mass of the particle is considered unit, while $\nabla_{\boldsymbol{\theta}} g$ is regarded as the force on the particle at position $\boldsymbol{\theta}$. α is within 0 and 1 and determines how quickly the contribution of previous gradients decays. The larger α is, the more heavily the optimisation direction is influenced by accumulative previous steps. This also helps to avoid local minima.

While momentum is based on the running estimate of the first moment of $\nabla_{\boldsymbol{\theta}} g$ (its mean value), Tieleman and Hinton (2012) showed that the use of the running estimate of its second moment guarantees convergence in certain optimisation situations, leading to the *RMSProp* algorithm,

$$\begin{aligned} \boldsymbol{r} &\leftarrow \rho \boldsymbol{r} + (1-\rho)\nabla_{\boldsymbol{\theta}} g \odot \nabla_{\boldsymbol{\theta}} g \\ \Delta\boldsymbol{\theta} &\leftarrow -\frac{\epsilon}{\sqrt{\delta + \boldsymbol{r}}} \odot \nabla_{\boldsymbol{\theta}} g \\ \boldsymbol{\theta} &\leftarrow \boldsymbol{\theta} + \Delta\boldsymbol{\theta}, \end{aligned} \tag{3.6}$$

where δ is a small constant (e.g. 1e-8) avoiding division by 0, $\odot$ denotes elementwise multiplication, and ρ is a parameter controlling the speed of the running average. $\boldsymbol{r}$ can be considered an accumulation factor regulating the learning rate.

The algorithm used within this thesis is *Adam* (Kingma and Ba, 2014), which features the second moment modified to include momentum,

$$\begin{aligned} t &\leftarrow t + 1 \\ \boldsymbol{s} &\leftarrow \rho_1 \boldsymbol{s} + (1-\rho_1)\nabla_{\boldsymbol{\theta}} g \\ \boldsymbol{r} &\leftarrow \rho_2 \boldsymbol{r} + (1-\rho_2)\nabla_{\boldsymbol{\theta}} g \odot \nabla_{\boldsymbol{\theta}} g \\ \boldsymbol{s} &\leftarrow \frac{\boldsymbol{s}}{1-\rho_1^t} \\ \boldsymbol{r} &\leftarrow \frac{\boldsymbol{r}}{1-\rho_2^t} \\ \Delta\boldsymbol{\theta} &\leftarrow -\epsilon \frac{\boldsymbol{s}}{\sqrt{\boldsymbol{r}} + \delta} \\ \boldsymbol{\theta} &\leftarrow \boldsymbol{\theta} + \Delta\boldsymbol{\theta}, \end{aligned} \tag{3.7}$$

where $\boldsymbol{s}$, $\boldsymbol{r}$ or t are initialised as $\mathbf{0}$ or 0. The Adam algorithm is generally regarded robust to the choice of ρ_1 and ρ_2 (default is 0.9 and 0.999 respectively), while the learning rate ϵ can be tuned for better test performance.

All these advanced learning algorithms feature adaptive learning rate, such that parameters with smaller gradients take larger updates (larger effective learning rate) (Goodfellow et al., 2016).

3.2.2.2 Cross Validation

Other variables encountered in the ANN parameter optimisation process such as the batch size and learning rate as well as the topology of an ANN (e.g. the number of hidden layers and the number of neurons in each hidden layer, to be introduced below) have to be specified before the optimisation is implemented. These are usually called the *hyperparameters*. Hyperparameters have a direct impact on the network performance and the optimal values of these are task specific. However, because this impact cannot be quantified easily, it is difficult to optimise hyperparameters alongside $\boldsymbol{\theta}$ using algorithms mentioned above. Instead, a stochastic optimisation technique is adopted, i.e. training is repeated for each of the networks configured from a set of hyperparameters sampled from a search range. The trained network with the lowest error on a withheld *validation set*, $S_{\text{val}} \sim D$, yields the optimal set of hyperparameters. The validation set is independent from the training dataset. Similar to the training data, the validation set is used during the training process only. However, it is mainly for estimating the network performance on unseen data, not for updating the network parameters as the training dataset does. Thus the validation process guides the training towards minimising the generalisation error. It is most efficient to randomly sample the hyperparameter space rather than searching along its parameter grids (Bergstra and Bengio, 2012).

3.2.2.3 Generalisation

The ultimate goal of ANN training is to achieve the best performance on a *test set* $S_{\text{test}} \sim D$ that is not seen during training, which can be considered an estimate of the generalisation error. A trained network can have minimised training error by memorising all training samples but will perform poorly on S_{test}, which is referred to as *overfitting*. To avoid this many methods have been proposed, derived from either statistical or empirical principles. For the experiments in this thesis we use two methods: *early stopping* and *dropout*. The effect of the size of S_{train} on generalisation error is also evaluated qualitatively.

Early stopping. This is an empirical approach and perhaps the simplest for improved generalisation (Graves, 2012). The objective function is evaluated on the validation set S_{val} after a fixed number of epochs and the lowest validation error is recorded. If this error shows no improvement after a few evaluations (the number of which is referred to as *patience*), training is terminated. This approach is based on the fact that during training the validation error usually levels off or begins to rise after a certain point while the training error continues to decrease (see Fig. 3.6 in Graves (2012)). In this thesis we evaluate the validation error after each epoch and the patience is set to 5.

Dropout. Dropout is a regularisation technique (Srivastava et al., 2015; Gal and Ghahramani, 2016), which encodes preference to one group of f over others. For example, an L_2 regularisation favours $\boldsymbol{\theta}$ of small L_2 norm. Dropout works by randomly dropping a fixed proportion of neurons (alongside their connections with other neurons) within the network before each training epoch to reduce the physical capacity of the network. At test time, all neurons and connections are assumed active. This prevents the network from assigning significant connections to features specific to the training set only. A large network trained with dropout is proved to have lower validation error than a smaller one trained without dropout (Goodfellow et al., 2016).

The training set size. When the test error of a trained system is significant but the training error is much lower, the most efficient way of reducing the generalisation error is to enlarge S_{train}. As is shown in Fig. 5.4 in Goodfellow et al. (2016), as the training set size increases, the test error decreases rapidly while the training error increases slightly. When the amount of training data, the complexity of the optimisation problem, and the effective capacity of the network (determined by both its topology and the optimisation process) match with each other, both the training error and the test error will stabilise after certain point, with the test error slightly higher than the training error showing good generalisation property.

3.2.3 Multilayer Perceptron Networks

ANNs were originally developed to emulate the signal processing capability of biological neurons (McCulloch and Pitts, 1943; Rosenblatt, 1958), though nowadays the development of ANNs has deviated from pure understanding of neuroscience. Many varieties of ANNs have emerged over the years with distinct structures and properties. Depending on whether its outputs are dependent only on current inputs or on both current and past/future inputs, ANNs can be divided into feedforward networks and feedback, recursive or recurrent networks. The most widely used feedforward network is the Multilayer Perceptron (MLP) (Rumelhart et al., 1986). MLPs are said to be universal function approximators because an MLP with a hidden layer containing sufficient nonlinear neurons are proven to be able to approximate any continuous function on a compact input domain (Hornik et al., 1989).

3.2.3.1 Forward Pass

The structure of a Fully Connected (FC) MLP is shown in Fig. 3.1. It is composed of neurons (or nodes) in a layered structure. By fully connected we mean each neuron is connected to not partial but all neurons in adjacent layers. The shown network has two hidden layers. The depth of an MLP can be extended to multiple hidden layers for increased capacity of representation (Goodfellow et al., 2016).

Consider an MLP receiving n_0-element input vector $\boldsymbol{x}$. For each hidden neuron h in the first hidden layer, it is connected with input neurons by the weight vector $\boldsymbol{w}_h^1$, and a weighted sum of $\boldsymbol{x}$ is referred as the network input a_h^1 to neuron h. The activation function of the first layer, g_1, is then applied to this input, yielding the activation b_h^1,

$$\begin{aligned} a_h^1 &= \sum_{i=1}^{n_0} w_{ih}^1 x_i \\ b_h^1 &= g_1(a_h^1). \end{aligned} \tag{3.8}$$

Most used activation functions are the hyperbolic tangent

$$tanh(a) = \frac{e^{2a} - 1}{e^{2a} + 1}, \tag{3.9}$$

and the logistic sigmoid

$$\sigma(a) = \frac{1}{1 + e^{-a}}. \tag{3.10}$$

The sigmoid was used in the original biological model squashing the weighted sum between 0 and 1, representing the firing rate of a neuron. $tanh(a)$ and $\sigma(a)$ are linked via $tanh(a) = 2\sigma(2a) - 1$, and are found to provide the same results in practice (Graves, 2012).

Rectified Linear Unit (ReLU) is a piece-wise linear function,

$$ReLU(a) = max(0,\ a). \tag{3.11}$$

ReLU is more computationally efficient. It has fewer vanishing gradient problems than sigmoid and tanh that render training of deep neural networks difficult (Hochreiter, 1991; Bengio et al., 1994; Glorot et al., 2011), gaining vast popularity as learning with deep networks has dominated.

All these activations are considered nonlinear, providing far more power than linear networks when for example approximating nonlinear functions or finding nonlinear classification boundaries (Goodfellow et al., 2016). They are all differentiable functions, allowing the network to be trained with gradient-based optimisation algorithms. Their first derivatives are

$$\begin{aligned} tanh'(a) &= 1 - tanh^2(a) \\ \sigma'(a) &= \sigma(a)(1 - \sigma(a)) \\ ReLU'(a) &= H(a), \end{aligned} \tag{3.12}$$

where $H(a)$ is the Heaviside function, which returns 1 for non-negative inputs and 0 otherwise.

The calculation in Eq. 3.8 is repeated for subsequent hidden layers, i.e.

$$\begin{aligned} a_h^k &= \sum_{i=1}^{n_{k-1}} w_{ih}^k b_i^{k-1} \\ b_h^k &= g_k(a_h^k) \end{aligned} \tag{3.13}$$

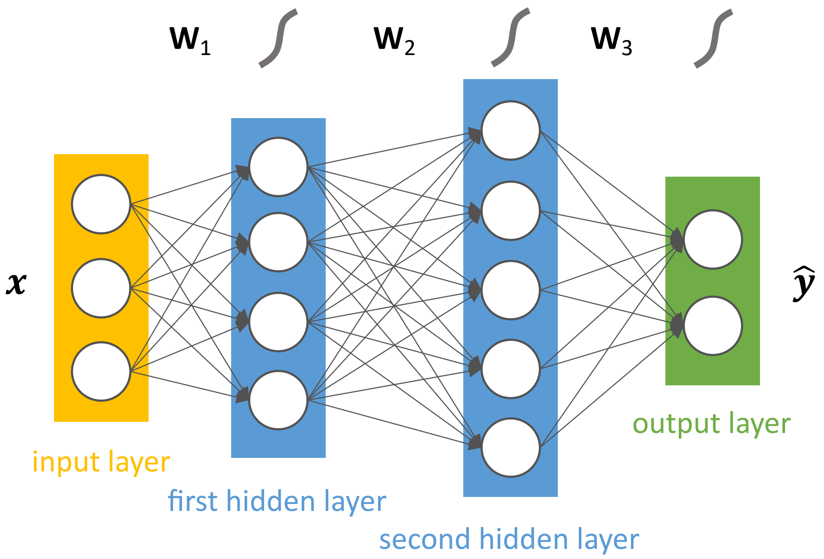


Figure 3.1: Example structure of a multilayer perceptron with two hidden layers. Each cell (represented by a white circle) represents a neuron. This network receives 3-element input vectors and outputs 2-element vectors. Each neuron in the same layer is connected to all neurons in adjacent layers but not connected to each other. $\mathbf{W}_i$ $(i = 1, 2, 3)$ is the weighting matrix. For example, $\mathbf{W}_1\boldsymbol{x}$ gives the 4-element input vector $\boldsymbol{a}_1$ to the first hidden layer. This vector is then transformed by an elementwise nonlinear function, producing the output vector $\boldsymbol{b}_1$ of this layer. This process continues until the output layer is reached.

for neuron h in hidden layer k.

The output layer is treated as a hidden layer for calculating the output vector $\tilde{\boldsymbol{y}}$,

$$\begin{aligned} a_h^{K+1} &= \sum_{i=1}^{n_K} w_{ih}^{K+1} b_i^K \\ \tilde{y}_h &= g_{K+1}(a_h^{K+1}), \end{aligned} \tag{3.14}$$

where K is the number of hidden layers. The number of neurons in the output layer (the length of output vector) and its activation function depend on the specific task (Goodfellow et al., 2016).

An initial value of L, the objective function of the training process, has to be determined for the optimisation process to begin. This requires the initialisation of $\boldsymbol{\theta}$. Gradient based algorithms require that $\boldsymbol{\theta}$ has small, random initial values.

For experiments in this thesis, we use Xavier uniform distribution to initialise the weights linking feedforward FC layers (Glorot and Bengio, 2010). Xavier distribution is a uniform distribution between $-\sqrt{6/(n_{in}+n_{out})}$ and $\sqrt{6/(n_{in}+n_{out})}$, where n_{in} and n_{out} are the numbers of neurons in adjacent layers. For example, for the network shown in Fig. 3.1, $n_{in}=3,\ n_{out}=4$ for elements in $\mathbf{W}_1$.

3.2.3.2 Objective Function

Under the supervised learning scheme where the targeted output $\boldsymbol{y}$ is provided for a given input $\boldsymbol{x}$, the objective function L can now be evaluated. For regression problems such as the wavefront prediction task considered in this thesis, the MSE is a commonly adopted metric, which can be calculated from a minibatch of m training samples,

$$L^{\mathrm{MSE}} = \frac{1}{m}||\boldsymbol{y}_h - \tilde{\boldsymbol{y}}_h||_2^2, \tag{3.15}$$

where $||\cdot||_2$ denotes the Euclidean distance between two vectors.

3.2.3.3 Backward Pass

The backward pass is the process of finding the derivatives of the objective function with respect to network weights, such that gradient based optimisation algorithms introduced in Section 3.2.2.1 can be used for updating these weights and decreasing the objective function. *Backpropagation* is the technique for efficiently calculating derivatives in neural networks (Rumelhart et al., 1986), which is a repeated application of chain rule for calculating partial derivatives in calculus.

The first step of backpropagation is to calculate the derivative of the objective function with respect to network outputs. For the objective function defined in Eq. 3.15, for a single training sample, this derivative is

$$\frac{\partial L^{\mathrm{MSE}}}{\partial \tilde{y}_h} = 2(\tilde{y}_h - y_h) \tag{3.16}$$

for the h^{th} element in the output vector.

Here we introduce the following notation:

$$\delta_h^k \equiv \frac{\partial L^{\mathrm{MSE}}}{\partial a_h^k}. \tag{3.17}$$

For the output layer, from the chain rule we have

$$\delta_h^{K+1} = \frac{\partial L^{\mathrm{MSE}}}{\partial \tilde{y}_h} \frac{\partial \tilde{y}_h}{a_h^{K+1}} = 2(\tilde{y}_h - y_h) g'_{K+1}(a_h^{K+1}). \tag{3.18}$$

By applying the chain rule recursively, working backwards through hidden layers, we have

$$\delta_h^k = \frac{\partial L^{\mathrm{MSE}}}{\partial b_h^k} \frac{\partial b_h^k}{\partial a_h^k} = \frac{\partial b_h^k}{\partial a_h^k} \sum_{i=1}^{n_{k+1}} \frac{\partial L^{\mathrm{MSE}}}{\partial a_i^{k+1}} \frac{\partial a_i^{k+1}}{\partial b_h^k}. \tag{3.19}$$

From Eq. 3.13 we have

$$\delta_h^k = g'_k(a_h^k) \sum_{i=1}^{n_{k+1}} \delta_i^{k+1} w_{hi}^{k+1} \tag{3.20}$$

for any unit h in hidden layer k. Then we can have the derivatives with respective to each of the weights

$$\frac{\partial L^{\mathrm{MSE}}}{\partial w_{ih}^k} = \frac{\partial L^{\mathrm{MSE}}}{\partial a_h^k} \frac{\partial a_h^k}{\partial w_{ih}^k} = \delta_h^k b_i^{k-1}, \tag{3.21}$$

where w_{ih}^k is the weight linking neuron h in layer k and neuron i in layer $k-1$.

For training on minibatches, the derivative is evaluated as the average derivative from each sample.

3.2.4 Long Short-Term Memory Networks

While the output of an MLP depends only on the current input, in a Recurrent Neural Network (RNN), a memory of the input history is retained. RNNs are specialised in sequence-to-sequence mapping, using a recurrence formula of the form

$$\tilde{\boldsymbol{y}}_t = f_{\boldsymbol{\theta}}(\boldsymbol{h}_{t-1}, \boldsymbol{x}_t), \tag{3.22}$$

where f is a function dependent on the network topology. Its parameter set $\boldsymbol{\theta}$ is fixed across all time steps. The state vector $\boldsymbol{h}_t$ can be understood as a running 'memory' of all previous inputs, and is updated at each time step. $\boldsymbol{h}_0$ can either be $\mathbf{0}$ or treated as a parameter and learnt during training. In this thesis we will set $\boldsymbol{h}_0$ to $\mathbf{0}$. RNNs are proven to be able to approximate any sequence-to-sequence mapping to arbitrary accuracy given a sufficient number of hidden neurons (Hammer, 2000).

In this thesis we focus on Long Short-Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997; Gers et al., 1999), which is a powerful variety of RNN. LSTM is the most efficient solution so far to the vanishing gradient problem in training RNNs (Hochreiter, 1991; Bengio et al., 1994), allowing long-term memories to be maintained within the network.

The inner structure of an LSTM cell is shown in Fig. 3.2. While the topology of an MLP is determined by the number of hidden layers and the number of neurons in each layer, an LSTM cell can be constructed once the number of its neurons m is known. Several LSTM cells can be stacked though to increase the depth of the resulting network. In addition to a hidden state vector $\boldsymbol{h}_t$, LSTMs maintain a cell state vector $\boldsymbol{c}_t$. The calculation is as follows,

$$\begin{gathered}
\begin{bmatrix} \boldsymbol{i} \\ \boldsymbol{f} \\ \boldsymbol{o} \\ \boldsymbol{g} \end{bmatrix} = \begin{bmatrix} \sigma \\ \sigma \\ \sigma \\ tanh \end{bmatrix} \mathbf{W} \begin{bmatrix} \boldsymbol{x}_t \\ \boldsymbol{h}_{t-1} \end{bmatrix} \\
\boldsymbol{c}_t = \boldsymbol{f} \odot \boldsymbol{c}_{t-1} + \boldsymbol{i} \odot \boldsymbol{g} \\
\boldsymbol{h}_t = \boldsymbol{o} \odot tanh(\boldsymbol{c}_t).
\end{gathered} \tag{3.23}$$

$\boldsymbol{i}$, $\boldsymbol{f}$ and $\boldsymbol{o}$ are the input gate, forget gate and output gate respectively, controlling whether each memory element is updated, reset, or output to update the hidden state. These are binary gates activated by the sigmoid function. These vectors alongside $\boldsymbol{g}$, $\boldsymbol{c}$ and $\boldsymbol{h}$ are all m-element vectors, where m is the number of neurons of this cell. $\boldsymbol{g}$ is for additively modifying the cell memory, and it is this additive interaction that allows gradients on $\boldsymbol{c}$ to flow backwards through time uninterruptedly for a long period, enabling the retaining of long-term dependencies. $\mathbf{W}$ is the weighting matrix of dimensions $4m \times (m + n)$, where n is the input vector size. In

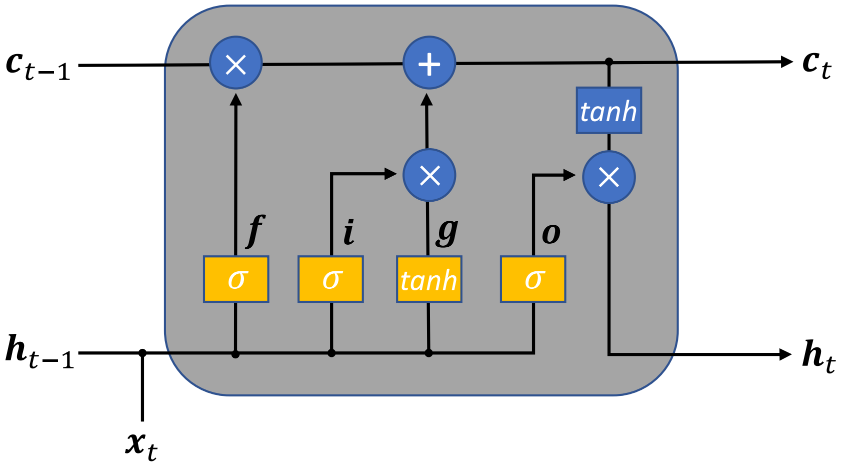


Figure 3.2: Data flow within an LSTM cell. $\boldsymbol{x}_t$ is the cell input. It has a hidden state $\boldsymbol{h}_t$, which is also its output vector, and a cell state $\boldsymbol{c}_t$ for retaining memory. $\boldsymbol{i}$, $\boldsymbol{f}$ and $\boldsymbol{o}$ are three gates activated by the sigmoid function, controlling whether the cell memory is updated, reset or output to update the hidden state. $\boldsymbol{g}$ is for additively modifying the memory content. All these vectors are updated at each time step.

this thesis, blocks in $\mathbf{W}$ linking $\boldsymbol{x}_t$ are initialised using Xavier distribution between $-\sqrt{6/(n_{in}+n_{out})}$ and $\sqrt{6/(n_{in}+n_{out})}$. Blocks linking $\boldsymbol{h}_{t-1}$ are initialised as an orthogonal matrix to preserve gradient norm for a long term (Saxe et al., 2014; Vorontsov et al., 2017).

The addition of a cell state with internal loops and the gating mechanism makes the time constant of an LSTM variable. This can change dramatically with inputs even when the parameters of this network are fixed after training (Goodfellow et al., 2016). The calculation in Eq. 3.23 is repeated for all time steps. When applied, an LSTM can process a sequence spanning for arbitrary time steps, contrary to an MLP receiving inputs of fixed size only.

The gradients of the objective function with respect to network weights in an RNN can be computed using the technique called Backpropagation Through Time (BPTT) (Werbos, 1990). Like backpropagation, BPTT is also a recursive application of the chain rule. The difference is that because the RNN parameters are shared across time the partial derivatives also have to take into account all previous time steps. Equations of gradient computation in an LSTM can be found in Graves (2012).

The forward and backward calculations of an MLP or an LSTM introduced so far have been implemented by Keras (Chollet et al., 2015), which is a high-level ANN library written in Python. For this thesis, we will use Keras for ANN training.

3.3 SCAO Simulation

In this section we describe the schematic of a simulated SCAO system built with Soapy. This system takes the configuration of CANARY low-order SCAO mode (Morris et al., 2010). We use this system to generate sequences of open-loop slope measurements as ANN training data, and to evaluate the performance of trained ANN predictor in terms of Root Mean Squared (RMS) Wavefront Error (WFE).

The architecture of the simulated SCAO system is shown in Fig. 3.3. Throughout the simulation we use a point source at infinity to act as a natural Guide Star (GS). To generate the training data, one single turbulence layer is assumed. Here, the use of a single layer is for the ease of training. We will show that an ANN predictor trained using one turbulence layer is capable of predicting in multi-layer conditions (see Section 4.6). A large random phase screen with von Kármán statistics is generated within the atmosphere module `Atmos` at the start of each loop run. Pure frozen flow is assumed, under which the large phase screen is translated over the telescope aperture with a given velocity due to the wind. At each time step, a smaller portion of the large phase screen, the part of which is seen by the telescope aperture, is output to `SHWFS`. `SHWFS` then outputs measured noisy wavefront slopes from the image plane using the Thresholded Centre of Gravity (TCoG) algorithm to suppress photon noise and readout noise. A single frame delay can be used in Soapy simulations (the center loop in Fig. 3.3) to account for the inevitable WFS

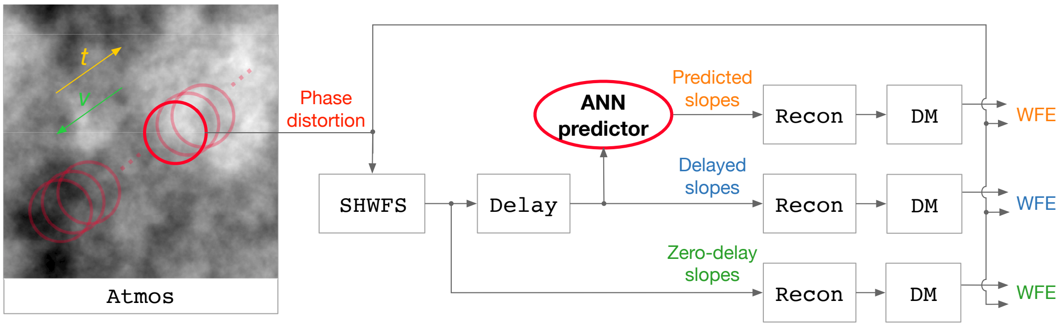


Figure 3.3: Composition of the simulated SCAO system and its data flow. RMS wavefront error of the predictive correction (upper) is expected to be between the 1-frame delay (center) and zero-delay (lower) corrections.

integration time. For the ANN training, we assume a 1-frame delay. This delay can be varied to any integer number of frames. Simulating sub-frame latency in Monte Carlo simulations involves additional iterations that increases the running time of the simulation and the time taken to generate the training data. This time lag between wavefront measurement and correction can be compensated either by applying slope measurements immediately (the lower loop) or by sending the prior slopes to an ANN predictor to extrapolate the current measurements (the upper loop). A reconstructor module (`Recon`) combines noisy slopes (either delayed, predictive, or delay-compensated) and control matrix generated during calibration to output Deformable Mirror (DM) commands, which are used by `DM` to generate the corrected phase. RMS error between the phase distortion and the DM shape is then output as the performance metric, RMS WFE.

The central loop can be seen as an open-loop integrator with the gain value in Eq. 2.28 set to 1. As can be seen from Fig. 3.4, this is the optimal gain for an open-loop integrator regardless of the noise condition. Different thresholding values (given in Table 3.1) of the TCoG algorithm were used at each guide star magnitude to suppress measurement noise. These thresholds gave the lowest RMS slope measurement error under each condition during the system calibration. The residual wavefront error is the average of 1,000 independent runs each lasting for 100 frames. We will keep the gain in the 1-frame delay loop as 1 in the following

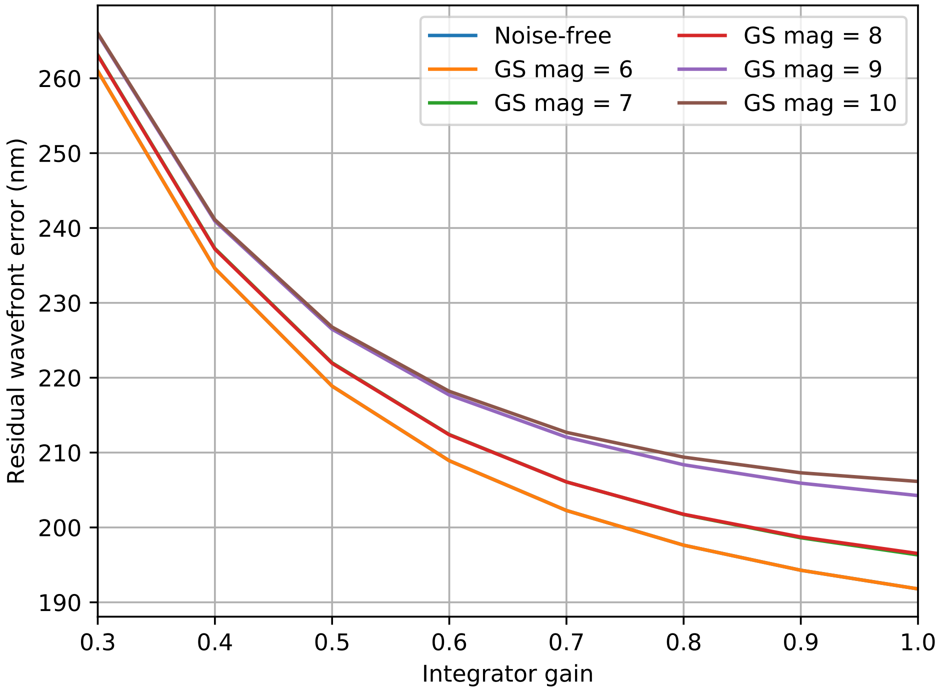


Figure 3.4: Residual wavefront error as the gain of the open-loop integrator changes. A unity-gain integrator corresponds to the central loop in Fig. 3.3, which we will refer to as the 1-frame (or 2-frame in Chapter 5) delay loop. Due to optimised noise suppression during the WFS measurement, the curves for the noise-free and the magnitude 6 conditions, or the magnitude 7 and 8 conditions, overlap.

Table 3.1: Thresholding value of the TCoG algorithm as the GS magnitude varies.

	Noise-free	6	7	8	9	10
Threshold	0	0	0.02	0.02	0.1	0.1

analyses.

We also note from Fig. 3.5 that an open-loop unity-gain integrator has better performance than a closed-loop optimised-gain integrator in the simulation setup regarding temporal, noise, aliasing and fitting errors only. This is because when the system is controlled in closed loop, any WFS measurement error such as noise and aliasing will be integrated with a non-zero gain, leading to performance degradation compared with an open-loop integrator of the same gain. The closed-loop gain was sampled from 0.3 to 1 with a step of 0.1. The optimal gain yields the lowest residual

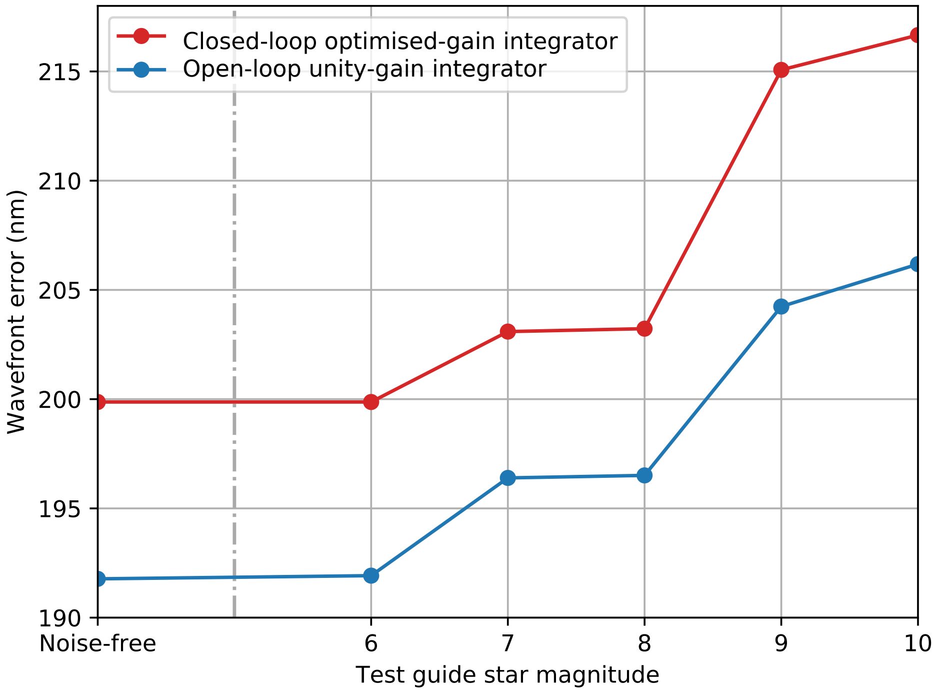


Figure 3.5: Performance comparison of a closed-loop optimised-gain integrator and an open-loop unity-gain integrator as guide star magnitude varies. Here the error sources are temporal, noise, aliasing and fitting errors.

wavefront error.

For the following analyses, we will use the 1-frame (or 2-frame as in Chapter 5) delay loop as our performance benchmark. This can be considered an open-loop unity-gain integrator. In reality, a pure open-loop system is rare and challenging because the DM is not seen by the WFS. Potential DM errors are difficult to calibrate and this can easily deviate the system performance from being optimal. The configuration of open-loop wavefront sensing and control adopted in this study is mainly for the ease of implementation and performance interpretation, as has been explained in Section 1.3.

Principal simulation parameters for generating ANN training data and for the performance analyses conducted in Chapter 4, unless stated otherwise, are listed in Table 3.2. A layout of the subapertures is shown in Fig. 3.6. A detailed introduction

Table 3.2: Principal parameters used with the Soapy SCAO simulation for ANN training and optimisation.

Parameter	Value
Telescope diameter	4.2 m
Central obscuration	1.2 m
System frequency	150 Hz
Loop latency	1 frame
Throughput	1
# of phase screens	1
Wind speed	10-15 m/s
Wind direction	0-360°
r_0 @ 500 nm	0.16 m
L_0	25 m
GS magnitude	10
Type of WFS	Shack-Hartmann
# of subapertures	7×7 (36 active)
# of pixels per subaperture	16× 16
Subaperture Field of View (FOV)	4.8 arcsec
Readout noise	1 e^- RMS
Photon noise	Yes
WFS wavelength	600 nm
Centroiding algorithm	TCoG
Thresholding value of TCoG	0.1
Type of DM	Piezo
# of DM actuators	8×8
Real-Time Control (RTC) in open loop	Yes

to CANARY can be found in Section 5.2. We typically train the predictor under similar atmospheric and system conditions to those where it will be validated. The impact of the WFS noise on ANN training and the predictor's robustness against changes in input statistics will be explored in Chapter 4.

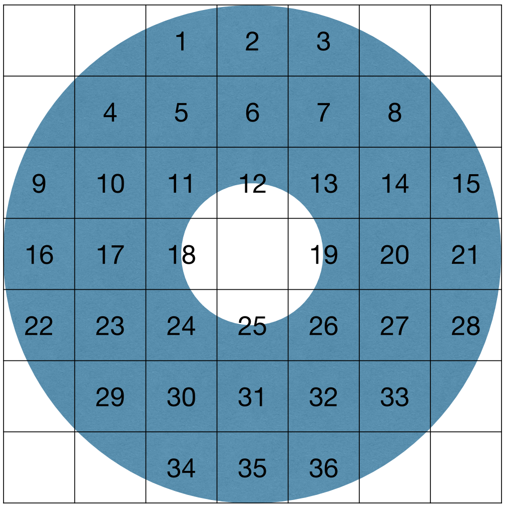


Figure 3.6: The layout of the subapertures of the simulated Shack-Hartmann Wavefront Sensor (SHWFS). The shaded area is the pupil, which is circular with a central obscuration. Each square cell with a number represents the position of an active subaperture, which has over 50% overlap in area with the pupil and is thus defined as well-illuminated.

3.4 ANN Training

3.4.1 Training Data Generation

The wavefront sensing subsystem consisting of `Atmos` and `SHWFS` modules is used to generate the first 100,000 training samples. Each sample is a time sequence of thirty 72-element vectors $(\boldsymbol{s}_1, \boldsymbol{s}_2, \dots , \boldsymbol{s}_{30})$, with each vector, $\boldsymbol{s}_i$, being the x and y slope for each of the 36 subapertures. $(\boldsymbol{s}_1, \boldsymbol{s}_2, \dots , \boldsymbol{s}_{29})$ will be ANN inputs sequentially during training, and $\boldsymbol{s}_{30}$ will be the targeted output. Wind velocity corresponding to each sample is a random vector, with its magnitude uniformly sampled from the range 10 to 15 m/s and its direction uniformly sampled from the range 0 to 360°. Wind velocity is constant within each sequence.

We then reverse each sequence to form the other half of the training set, with the last frame being the first and first being last. This corresponds to reversing the wind direction. We use this data augmentation approach to introduce variability in training data to improve model robustness. Resulting ANN input and targeted output sets are tensors of shape $(2\times10^5, 29, 72)$ and $(2\times10^5, 72)$ respectively. The amount of training data is decided by trial and error to match both ANN architecture complexity and problem complexity to balance between training data fitting and model generalisation. No further training data pre-processing is implemented. During training, 90% of the dataset forms the training dataset while the remaining 10% is reserved to form the validation set.

3.4.2 ANN Training and Hyperparameter Tuning

The ANN architecture consists of stacked LSTM cells and a final FC output layer. The activation function of the FC layer is ReLU. The depth of neural networks is associated with the depth of representations that can be learnt (Goodfellow et al., 2016), thus the stacking of LSTM cells in our case.

The ANN topology comprising two LSTM cells and a FC layer is shown in Fig. 3.7. The display is unrolled in time, which means all components in the same colour (or row) are duplicates in time and essentially identical to inputs at any time step. At each time step t ($t \geq 2$), the network can output a slope prediction $\tilde{\boldsymbol{s}}_t$ based on the current input $\boldsymbol{s}_{t-1}$ and two state vectors, the cell state and the hidden state. Both states are either initialised as all-zero vectors ($t = 2$) or updated at each time step ($t > 2$) using information in the input sequence so far.

10% dropout is deployed for each LSTM cell (Gal and Ghahramani, 2016). Batch size is set to 128. The training error is MSE between the targeted output $\boldsymbol{s}_{30}$ and the actual output $\tilde{\boldsymbol{s}}_{30}$ evaluated and averaged on the current minibatch. The Adam optimisation algorithm is used to optimise the network parameters in a direction that minimises the training error. During one epoch, every minibatch is evaluated once and the network parameters are updated accordingly multiple times. At the end of each epoch, the updated network is evaluated on the validation set for the assessment of the generalisation error. The default learning rate, 1e-3, of the

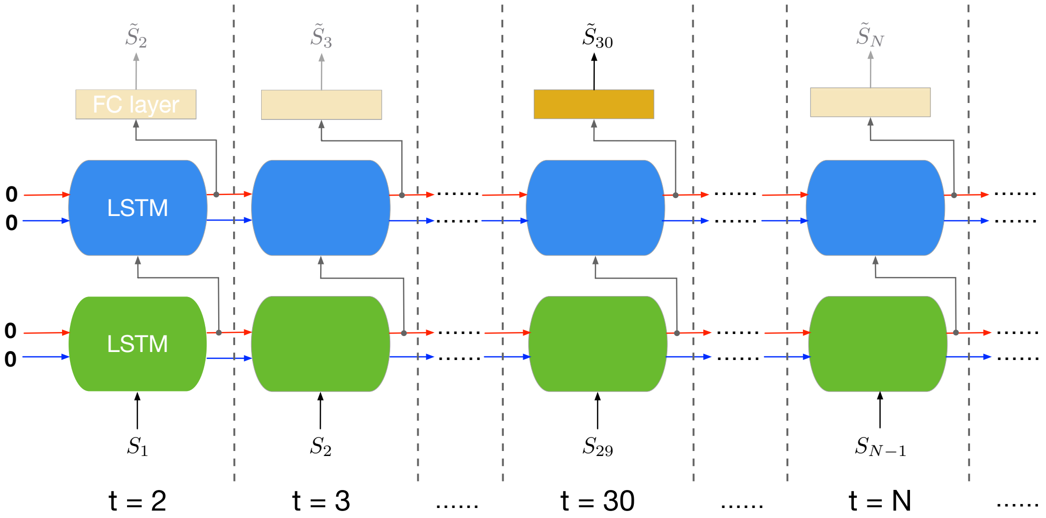


Figure 3.7: The ANN predictor structure unrolled in time. The predictor can start predicting from the 2nd time step, although initial predictions can be unstable and inaccurate due to limited temporal information. The two LSTM cells have the same inner structure, but different sets of parameters after training.

Adam algorithm is kept as no prominent performance improvement was observed by tuning this. If the validation error shows no improvement for 10 consecutive epochs, showing the potential of overfitting on the training set probably due to the optimisation process being too fast, the learning rate is reduced to its 1/5 unless reaching 1e-5. The reduced learning rate allows only small updates of the network parameters to prevent this optimisation process from early stagnation. Training is terminated after 40 epochs, at which point both training and validation errors have levelled off.

The hyperparameter tuning process is coupled with ANN training. We tune two hyperparameters that determine the physical capacity of the network: number of stacked LSTM cells (1 or 2) and length of output vectors of each LSTM cell (a random integer between 100 and 250, different for each cell). Every time a set of these two hyperparameters are chosen, the model is recompiled, re-initialised and re-trained as is described above. The random search routine is adopted (Bergstra and Bengio, 2012). In total 30 combinations of the hyperparameters were tested. The model that achieves the lowest validation error at the end of the 40th epoch

Table 3.3: The optimised ANN architecture.

Module	Input vector size	Output vector size
First LSTM	72	247
Second LSTM	247	226
FC	226	72

is composed of two LSTM cells and a final FC layer (as is shown in Fig. 3.7). The output vector of the first LSTM cell has 247 elements and the second cell has 226 elements. The resulting model has 761,000 trainable parameters in total.

After training, the optimised predictor is inserted between `SHWFS` and `Recon` to form part of a predictive correction loop. From this stage, the parameters within the ANN are fixed and inputs are now processed in a deterministic way.

3.5 Conclusions

In this chapter we have developed the training methodology of an ANN predictor with a simulated SCAO system. We have shown how the wavefront sensing subsystem can be used for generating training data and how the ANN predictor can be inserted between the WFS and the DM to form a predictive correction loop, the residual error of which can be compared with the delayed or zero-delay loop to evaluate the ANN performance. We described the principle and workflow of the ANN training and hyperparameter tuning process, and presented the architecture of the trained ANN predictor that will be examined in greater detail in the following chapters.

We will show in the following chapters that although originally developed with simulated frozen-flow turbulence, the training methodology presented here can be adapted to work with on-sky turbulence as well, showing that the training method is valid for a variety of problems.

3.6 References

Y. Bengio, P. Simard, and P. Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, 5(2):157–166, 1994.

J. Bergstra and Y. Bengio. Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13:281–305, 2012.

F. Chollet et al. Keras. `https://keras.io`, 2015.

Y. Gal and Z. Ghahramani. A theoretically grounded application of dropout in recurrent neural networks. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, pages 1027–1035, 2016.

F. A. Gers, J. Schmidhuber, and F. Cummins. Learning to forget: continual prediction with lstm. In *1999 Ninth International Conference on Artificial Neural Networks ICANN 99. (Conf. Publ. No. 470)*, volume 2, pages 850–855, 1999.

X. Glorot and Y. Bengio. Understanding the difficulty of training deep feedforward neural networks. *Journal of Machine Learning Research - Proceedings Track*, 9: 249–256, 2010.

X. Glorot, A. Bordes, and Y. Bengio. Deep sparse rectifier networks. In *AISTATS*, volume 15, pages 315–323, 2011.

I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning.* MIT Press, 2016. `http://www.deeplearningbook.org`.

A. Graves. *Supervised Sequence Labelling with Recurrent Neural Networks.* PhD thesis, Technische Universität at München, 2012.

B. Hammer. On the approximation capability of recurrent neural networks. *Neurocomputing*, 31(1-4):107–123, 2000.

S. Hochreiter. *Untersuchungen zu dynamischen neuronalen Netzen*. PhD thesis, Technische Universität at München, 1991.

S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 9:1735–1780, 1997.

K. Hornik, M. Stinchcombe, and H. White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5):359–366, 1989.

A. Karpathy. *Connecting images and natural language*. PhD thesis, Stanford University, 2016.

D. Kingma and J. Ba. Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference on Learning Representations*, 2014. `http://arxiv.org/abs/1412.6980`.

Y. LeCun, L. Bottou, G. B. Orr, and K.-R. Müller. Efficient backprop. In *Neural Networks: Tricks of the Trade*, pages 9–50. Springer-Verlag, 1998.

W. McCulloch and W. Pitts. A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, 5:115–133, 1943.

T. Morris, Z. Hubert, R. Myers, E. Gendron, A. Longmore, G. Rousset, G. Talbot, T. Fusco, N. Dipper, F. Vidal, D. Henry, D. Gratadour, T. Butterley, F. Chemla, D. Guzman, P. Laporte, E. Younger, A. Kellerer, M. Harrison, M. Marteaud, D. Geng, A. Basden, A. Guesalaga, C. Dunlop, S. Todd, C. Robert, K. Dee, C. Dickson, N. Vedrenne, A. Greenaway, B. Stobie, H. Dalgarno, and J. Skvarc. Canary: The ngs/lgs moao demonstrator for eagle. In *Proceedings of the First AO4ELT Conference*. EDP Sciences, 2010.

F. Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386–408, 1958.

D. E. Rumelhart, G. E. Hinton, and R. J.Williams. Learning representations by back-propagating errors. *Nature*, 323:533–536, 1986.

A. M. Saxe, J. L. McClelland, and S. Ganguli. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. *CoRR*, abs/1312.6120, 2014.

N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15:1929–1958, 2015.

I. Sutskever, J. Martens, G. Dahl, and G. Hinton. On the importance of initialization and momentum in deep learning. In *Proceedings of the 30th International Conference on International Conference on Machine Learning*, page III–1139–III–1147. JMLR.org, 2013.

T. Tieleman and G. Hinton. Lecture 6.5—RmsProp: Divide the gradient by a running average of its recent magnitude. COURSERA: Neural Networks for Machine Learning, 2012.

E. Vorontsov, C. Trabelsi, S. Kadoury, and C. Pal. On orthogonality and learning recurrent networks with long term dependencies. volume 70 of *Proceedings of Machine Learning Research*, pages 3570–3578, 2017.

P. J. Werbos. Backpropagation through time: what it does and how to do it. *Proceedings of the IEEE*, 78(10):1550–1560, 1990.

Chapter 4

Prediction Results in Simulation

4.1 Overview

In this chapter, we will investigate if an Artificial Neural Network (ANN) can be used for wavefront prediction in a simulated Single-Conjugate Adaptive Optics (SCAO) system and characterise system performance when an ANN-based predictor is used. We will also address the following questions:

1. Among the many turbulence and system parameters determined before training, which is an ANN predictor sensitive or insensitive to?
2. Can an ANN trained with a single turbulence layer generalise to multi-layer profiles?
3. Can the ANN-based predictor operate without knowledge of the spatial distribution of Wavefront Sensor (WFS) subapertures?

To investigate these, the performance of the ANN predictor obtained through the training workflow presented in the previous chapter is examined across seven scenarios:

- Guide Star (GS) magnitude is decreased from 10 (on which the predictor is trained) to 6, which decreases WFS noise in input slopes. We investigate the noise level of training data on ANN performance. We then examine the predictor's performance in the spatial and temporal frequency domain to assist our understanding of its performance.

- The turbulence strength is varied from an r_0 of 8 to 30 cm, for an ANN trained with an r_0 of 16 cm.
- A time-variant turbulence is considered by changing either the wind speed or the direction every 10 frames (15 Hz) after the predictor stabilises.
- We test the stability and robustness of the predictor under realistic r_0 conditions for a simulated time of one hour (540,000 frames).
- A multi-layer turbulence is considered to test the predictor's ability to track multiple wind vectors.
- We extend our approach to account for a more realistic 2-frame latency, where we trained a separate ANN predictor to predict two frames in advance directly, and compare that with applying the single-frame latency predictor twice.
- We compare the performance of spatially aware predictor and non-spatially aware predictor. For the former, the prediction of a single subaperture is based on the history of all subapertures. For the latter, each subaperture is treated independently and in parallel, although the same temporal evolution is assumed for all. This, alongside Section 5.7, will investigate if spatial awareness is necessary for predicting frozen flow and non-frozen flow turbulence such as dome seeing.

In most scenarios, statistics of the input slopes to the predictor are slightly different from that used during training to test the generalisation capability of the ANN. In each scenario, we used 1,000 simulated test slope sequences each of 100 frames (lasting for 0.67 s) for the test. The predictor's memory (both cell and hidden states) is zeroed before a new slope sequence. The predictor is expected to build up its memory and output stable predictions in 30 frames as the training was designed. Other simulation parameters are mostly the same as listed in Table 3.2, unless stated otherwise.

4.2 Performance with Varying WFS Noise Level

In this section, we examine the performance under varied WFS noise levels, and the sensitivity of an ANN predictor to the training noise level.

To investigate these, in addition to the ANN obtained in Section 3.4.2 that was trained on a guide star of magnitude 10, we include results from two networks that have been trained with different magnitudes of GSs (thus different noise levels) using the training methodology developed in Section 3.4. These three predictors are denoted as **Mag-10**, **Mag-8** (trained with a GS of magnitude 8) and **Noise-free** (trained without WFS noise) respectively. The training procedure and other simulation parameters were the same as detailed in Section 3.4, except the thresholding value of the Thresholded Centre of Gravity (TCoG) algorithm that was reduced to 0.02 for the **Mag-8** predictor, and 0 for the **Noise-free** predictor to accommodate for the lowered noise levels. The resulting ANN architectures are listed in Table 4.1. For each network, we see that the residual Wavefront Error (WFE) decreases until the 20$^{\text{th}}$ frame, after which the performance of each ANN stabilises. This is depicted in Fig. 4.1 when observing a bright GS of magnitude 6.

The RMS WFE of all predictors as GS magnitude varies is shown in Fig. 4.2. The corresponding thresholding values of the TCoG algorithm are listed in Table 3.1. We note that the ANN trained with the highest noise level performs far better than the ANNs trained in lower noise regime and this behaviour was observed irrespective of test GS magnitude. **Mag-10** and **Mag-8** predictors perform better than the 1-frame delay in all test conditions.

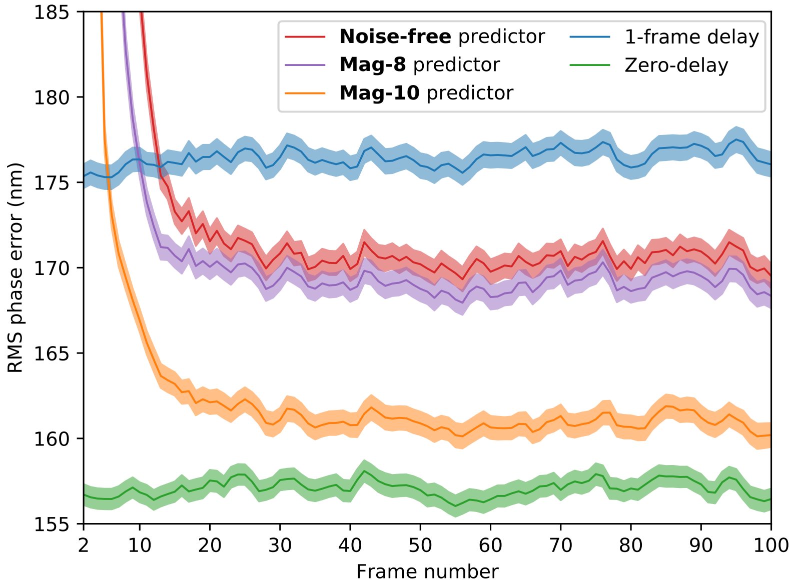


Figure 4.1: Mean Root Mean Squared (RMS) WFEs in an Adaptive Optics (AO) loop averaged across 1,000 test sequences. The GS used to generate test slopes has a magnitude of 6, which decreases the noise level of inputs to the predictor that is trained with a GS of magnitude 10 (**Mag-10**) compared with during its training. Wind speed is 15 m/s in a single direction. We also compare **Mag-10** predictor using the same set of test slopes with another two predictors that were trained with GS magnitude 8 (**Mag-8**) and trained without WFS noise (**Noise-free**).

Table 4.1: Training conditions and structures of **Mag-10**, **Mag-8** and **Noise-free** predictors.

ANN Predictor	**Mag-10**	**Mag-8**	**Noise-free**
GS magnitude during training	10	8	-
# of neurons of the first cell	247	247	162
# of neurons of the second cell	226	203	114

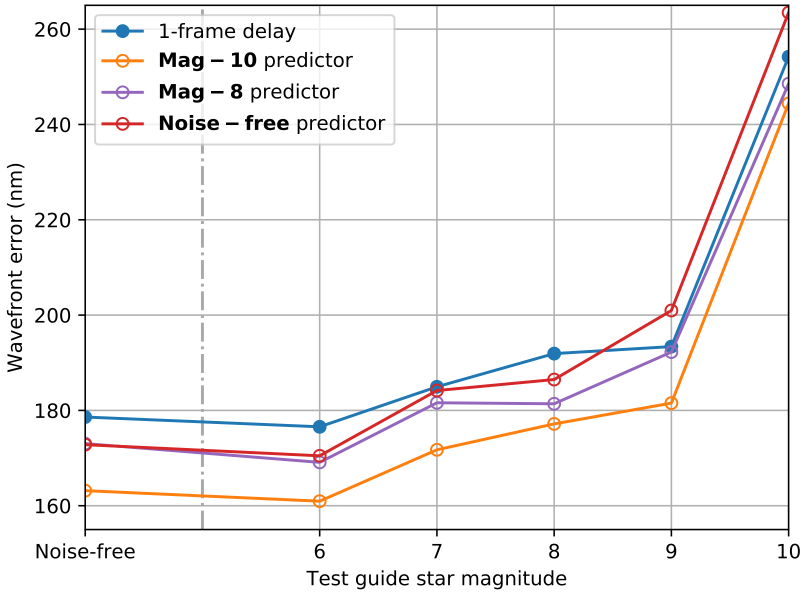


Figure 4.2: Performance of **Mag-10**, **Mag-8** and **Noise-free** predictors that were trained with decreasing WFS noise when the test guide star magnitude varies. The wavefront error is measured using data from the 30th frame after the ANNs have stabilised.

To further quantify the ANN performance, we introduce the prediction error, σ_p, as follows,

$$\sigma_p = \sqrt{\overline{\sigma}_{\text{pred}}{}^2 - \overline{\sigma}_{\text{zero-delay}}{}^2}, \tag{4.1}$$

where $\overline{\sigma}_*$ is the average WFE after the 30th frame and across all sequences. $\overline{\sigma}_{\text{pred}}{}^2$ is the WFE in the predictive correction loop and $\overline{\sigma}_{\text{zero-delay}}{}^2$ is the WFE in the zero-delay correction loop. The delay error is defined in a similar fashion, with $\overline{\sigma}_{\text{delay}}{}^2$ the WFE in the 1-frame delay loop,

$$\sigma_d = \sqrt{\overline{\sigma}_{\text{delay}}{}^2 - \overline{\sigma}_{\text{zero-delay}}{}^2}. \tag{4.2}$$

For the simulated SCAO system, the error sources are temporal, noise, aliasing and fitting errors, as described in Section 2.2.6. If we assume these terms are independent from one another, the wavefront error can be decomposed as,

$$\begin{aligned} \overline{\sigma}_{\text{zero-delay}}{}^2 &= \sigma^2_{\text{fitting}} + \sigma^2_{\text{noise}} + \sigma^2_{\text{aliasing}} \\ \overline{\sigma}_{\text{delay}}{}^2 &= \sigma^2_{\text{fitting}} + \sigma^2_{\text{noise}} + \sigma^2_{\text{aliasing}} + \sigma^2_{\text{temporal}}. \end{aligned} \tag{4.3}$$

Table 4.2: Delay error σ_d and prediction error σ_p (nm) of **Mag-10**, **Mag-8** and **Noise-free** predictors as GS magnitude varies.

GS magnitude	σ_d	**Mag-10** σ_p	**Mag-8** σ_p	**Noise-free** σ_p
Noise-free	82.6	39.3	69.8	69.2
6	80.6	34.9	62.5	66.2
7	77.4	35.7	69.0	75.5
8	82.1	36.0	53.0	68.4
9	71.1	24.6	67.9	89.7
10	72.6	19.6	49.2	100.3
Mean	77.7	31.7	61.9	78.2
STD	4.5	7.1	8.1	12.6

We make two assumptions about the error breakdown in the prediction case. Because both the input and targeted output slopes of the ANN are perturbed by noise and aliasing, we assume that the ANN predictions also contain these two error terms, although their values could have been altered. We use $\sigma_{\text{noise}'}$ and $\sigma_{\text{aliasing}'}$ to denote the ANN-filtered noise and aliasing errors. We will use the autocorrelation approach in Section 2.2.6.4 to identify and quantify $\sigma_{\text{noise}'}$. $\sigma_{\text{temporal}'}$ denotes the temporal error after the prediction and represents the ANN prediction power. The fitting error should remain unchanged before and after the prediction since in both cases this is limited only by the WFS sampling of the pupil. The second assumption is the filtered temporal, aliasing and noise errors are still independent. Correlations between errors introduced by the filtering may exist in practice, however this breakdown enables the quantitative examination of the ANN performance described here. As a result, $\overline{\sigma}_{\text{pred}}{}^2$ can be decomposed as

$$\overline{\sigma}_{\text{pred}}{}^2 = \sigma^2_{\text{fitting}} + \sigma^2_{\text{noise}'} + \sigma^2_{\text{aliasing}'} + \sigma^2_{\text{temporal}'}. \tag{4.4}$$

Combining Eqs. 4.1 to 4.4, we have

$$\begin{aligned} \sigma_d^2 &= \sigma^2_{\text{temporal}} \\ \sigma_p^2 &= (\sigma^2_{\text{noise}'} - \sigma^2_{\text{noise}}) + (\sigma^2_{\text{aliasing}'} - \sigma^2_{\text{aliasing}}) + \sigma^2_{\text{temporal}'} \end{aligned} \tag{4.5}$$

The delay error σ_d is the temporal error, σ_{temporal}. The aim of the prediction error σ_p is to give an insight into the prediction power of the ANN, represented by

$\sigma_{\text{temporal}'}$ when comparing it with σ_d. Calculated σ_d and σ_p as GS magnitude varies are listed in Table 4.2. The significantly decreased σ_d in much noisier conditions might be due to the inherent accuracy of the simulation.

However at this stage, a prediction error σ_p smaller than the delay error σ_d can indicate the suppression of noise and aliasing errors as well as the temporal error. We will examine the filtered error terms qualitatively and quantitatively in the following sections, and prove that $\sigma_{\text{temporal}'}$ is smaller than σ_{temporal}.

4.2.1 Modal Analysis of ANN Performance

In this section we analyse the Zernike breakdown of the residual WFEs to understand Fig. 4.2.

To access this, we can convert slope errors to phase errors via a perfectly calibrated control matrix $\mathbf{M}_{rz}$,

$$\begin{aligned} \boldsymbol{a}_{err}^{\text{delay}} &= \mathbf{M}_{rz}(\boldsymbol{s}_t - \boldsymbol{s}_{t-1}) \\ \boldsymbol{a}_{err}^{\text{pred}} &= \mathbf{M}_{rz}(\boldsymbol{s}_t - \tilde{\boldsymbol{s}}_t). \end{aligned} \tag{4.6}$$

$\mathbf{M}_{rz}$ is the pseudo inverse of a perfect interaction matrix $\mathbf{M}_{zi}$ *. Both matrices were developed for CANARY performance analysis. $\mathbf{M}_{zi}$ is the interaction matrix between a 7 × 7 Shack-Hartmann Wavefront Sensor (SHWFS), which takes the CANARY WFS configuration, and a 35-mode Zernike Deformable Mirror (DM). Each row of $\mathbf{M}_{zi}$, corresponding to the WFS response to a given Zernike mode input phase, were computed directly as the line integral of a high-resolution Zernike phase map along the perimeter of each subaperture (Akondi and Dubra, 2020), not from an image plane. This removes WFS noise and Charged Coupled Device (CCD) pixel sampling effects. The pseudo inverse was thus computed with a conditioning value of 0 without any regularisation or mode filtering. The limit of 35 Zernike terms represents those terms that are adequately sampled by the WFS and has been determined from the formula

$$n_Z = \frac{(N+1)(N+2)}{2}, \tag{4.7}$$

where N is the maximum Zernike radial order that a 7 × 7 WFS can sense, which is 7.

The temporal variance (after the 30th time step) of each Zernike error in each sequence, averaged across all sequences, gives an RMS error term on that mode z,

$$\begin{aligned}
&\sigma^2_{\text{delay},z}\\
&=< |a_t - a_{t-1}|^2 >\\
&=< |(a^t_{\text{atmos}} - a^{t-1}_{\text{atmos}}) + (a^t_{\text{noise}} - a^{t-1}_{\text{noise}}) + (a^t_{\text{aliasing}} - a^{t-1}_{\text{aliasing}})|^2 >\\
&=< |a^t_{\text{atmos}} - a^{t-1}_{\text{atmos}}|^2 > + < |a^t_{\text{noise}} - a^{t-1}_{\text{noise}}|^2 > + < |a^t_{\text{aliasing}} - a^{t-1}_{\text{aliasing}}|^2 > \quad (4.8)\\
&=< |a^t_{\text{atmos}} - a^{t-1}_{\text{atmos}}|^2 > + < |a^t_{\text{noise}}|^2 > + < |a^{t-1}_{\text{noise}}|^2 >\\
&\quad + < |a^t_{\text{aliasing}}|^2 > + < |a^{t-1}_{\text{aliasing}}|^2 >\\
&= \sigma^2_{\text{temporal},z} + \sigma^2_{\text{noise},z} + \sigma^2_{\text{noise},z} + \sigma^2_{\text{aliasing},z} + \sigma^2_{\text{aliasing},z}.
\end{aligned}$$

$< \cdot >$ denotes the temporal mean. The derivation uses the fact that $< |a^t_* \times a^{t-1}_*| >= 0$ where * denotes noise or aliasing, and that $< |a_{\text{a}} \times a_{\text{b}}| >= 0$ where a and b refer to two statistically independent terms. Similarly, we have

$$\sigma^2_{\text{pred},z} = \sigma^2_{\text{temporal}',z} + \sigma^2_{\text{noise},z} + \sigma^2_{\text{noise}',z} + \sigma^2_{\text{aliasing},z} + \sigma^2_{\text{aliasing}',z}. \quad (4.9)$$

Fig. 4.3 shows $\sigma_{\text{pred},z}$ of the **Mag-10** predictor and $\sigma_{\text{delay},z}$ of the 1-frame delay in the noise-free condition. We also plot the Zernike breakdown of the open-loop measurements, $< |a_t|^2 >$, and the RMS error in the 2-frame delay loop, $< |a_t - a_{t-2}|^2 >$, for comparison. The bump around Z30 indicates the aliasing. It is unclear why the 1-frame or 2-frame temporal error is especially low (e.g. Z8, Z16, Z25, Z30) or high (e.g. Z29) in some modes. The fact that the prediction follows this trend indicates that the delay compensation by the ANN is partial. Still, the ANN has a lower error in all modes than the 1-frame delay.

Fig. 4.4 shows the ratio between $\sigma_{\text{pred},z}$ and $\sigma_{\text{delay},z}$ as GS magnitude varies. A ratio less than unity indicates the system will benefit from the prediction of that

[*] Both matrices were obtained through the private communication with Lisa Bardou at Centre for Advanced Instrumentation (CfAI), Durham University.

mode, which is mostly the case. Figure 4.4 shows that the performance of the ANN predictor is highly dependent on the ANN training regime. The relative performance of the **Noise-free** predictor gets much worse as the magnitude increases, while the performance of the **Mag-8** and **Mag-10** predictors remains unchanged with variation in guide star magnitude. As the magnitude increases to 9 and 10, **Mag-10** predictor has the lowest error in all modes, followed by **Mag-8** predictor. Combining Figs 4.3 and 4.4 we can also see that the ANN reduction is most significant in poorly corrected modes (e.g. Z7, Z17, Z29) by the 1-frame delay.

These results show that:

- It is beneficial to train the ANN predictor using a dataset that includes sources of noise, and that ANNs trained in noise-free conditions can degrade performance when noise is present.
- Importantly, a predictor trained with noise can also operate in noise-free conditions.
- Predictors trained on noisy data are not insensitive to noise, as can be seen by the overall increase in ratio as the GS magnitude increases.
- ANN predictor performance appears to be dependent on the amount of noise present within the training data set, with noisier training data providing a better performance.

4.2.2 ANN Characteristics with Temporal Frequencies

The temporal behaviour of ANN predictors can be understood by analysing the temporal Power Spectral Density (PSD) of non-predicted and predicted slopes. The temporal PSDs were obtained using the Welch's average periodogram method (Welch, 1967), which groups the time sequence of each Zernike mode into subsequences of 1024 frames with an overlap of 512 frames, windowed by a Hanning window.

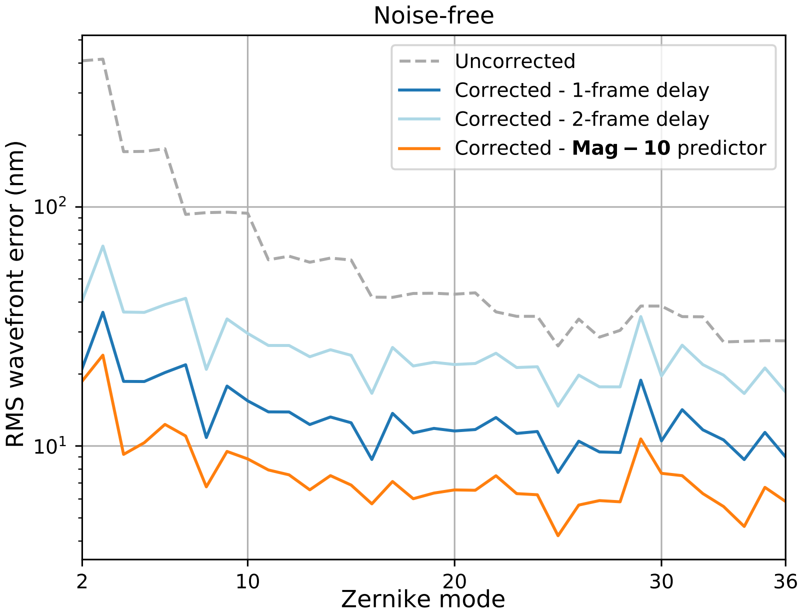


Figure 4.3: Zernike breakdown of the RMS wavefront error (nm) by the 1-frame delayed slopes, the 2-frame delayed slopes and the ANN predictive slopes compared with the zero-delay slopes. The Zernike breakdown of the uncorrected zero-delay slopes are also plotted for comparison.

Although an analysis of frequency transfer is typically implemented in a linear system where the input frequency is not altered by the system, this analysis can still be meaningful for a nonlinear system. For example, the comparison of residual PSDs was used in Landman et al. (2020) to demonstrate the ability of an ANN-based closed-loop controller in attenuating vibrations in Tip and Tilt (TT) modes.

4.2.2.1 Temporal Response with Sine Waves

We first input to the ANN simple sine waves to show its frequency transfer is nonlinear. The input is either a 49-Hz sine wave of unit amplitude, $\sin(98\pi t)$, or the sum of two sine waves with frequencies 45 and 49 Hz respectively, $\sin(90\pi t) + \sin(98\pi t)$. In each case the sequence lasts for 9,000 frames (60 s) with a sampling rate of 150 Hz. This sequence is then multiplied by a 72-element unit vector to match the input size of the ANN. This can be understood as a vibration in the

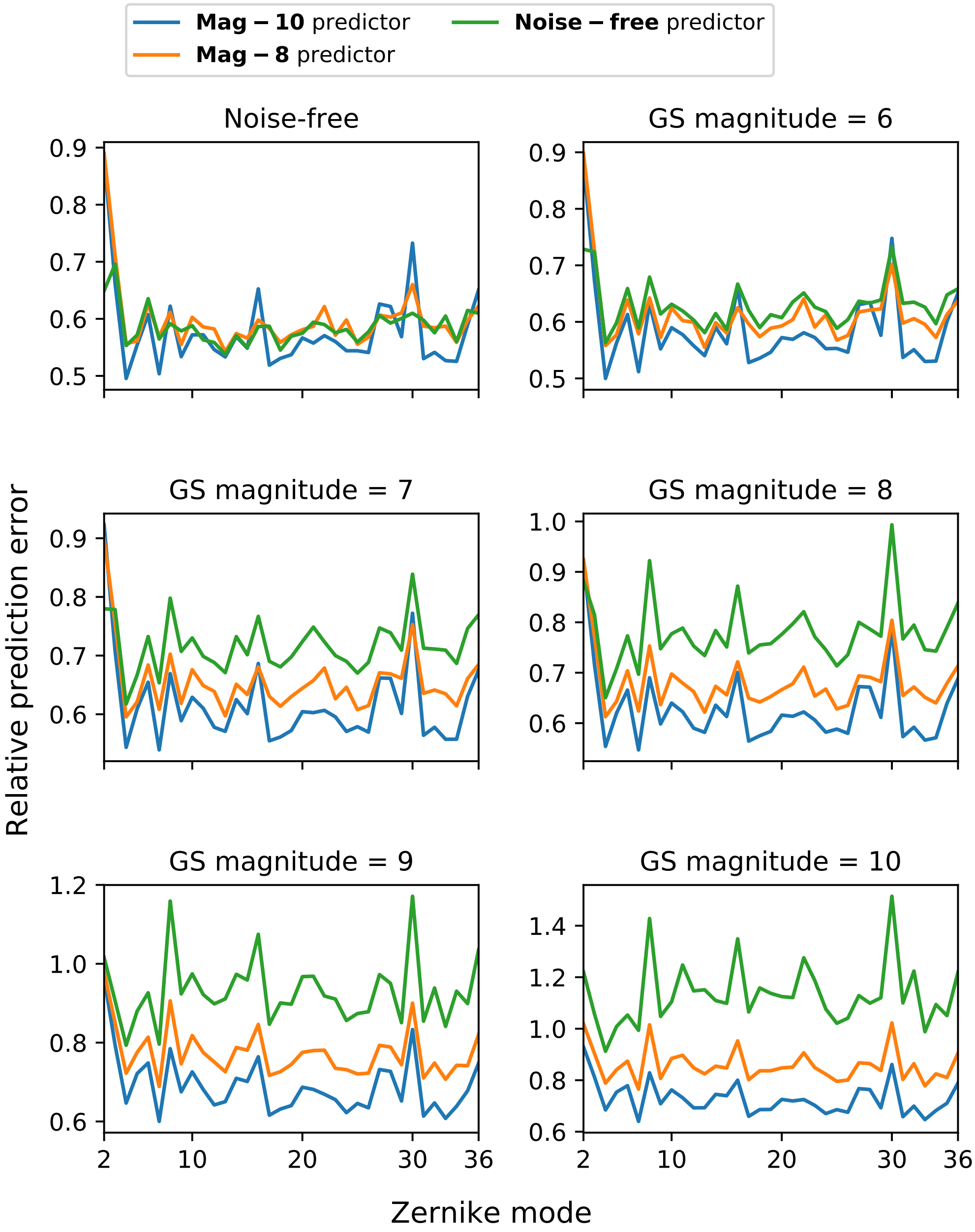


Figure 4.4: Ratio between the Zernike breakdown of the RMS wavefront error by the predicted slopes and the 1-frame delayed slopes. This is consistent with Fig. 4.2. A ratio below unity indicates the system will benefit from the ANN prediction of that mode.

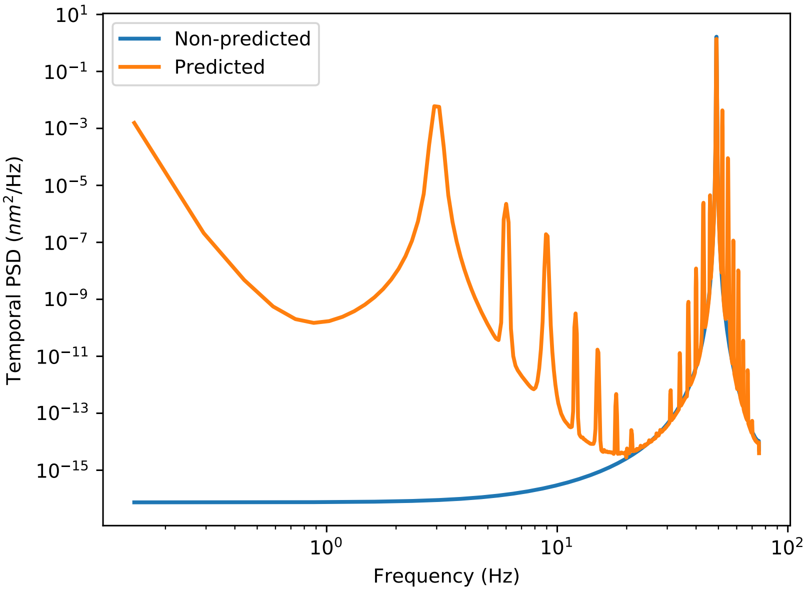


Figure 4.5: Temporal PSD of a sine wave (at 49 Hz) and that of the ANN prediction. The frequency limit is 75 Hz, with a temporal resolution of around 0.20 Hz.

TT modes.

The filtering of a single sine wave is shown in Fig. 4.5. The input has a distinct peak at 49 Hz as expected. After the ANN filtering, this peak is maintained, however significant amount of power is transferred to other frequencies, with a few peaks other than the 49 Hz. This is further complicated when the input contains more than one frequency (see Fig. 4.6 for the filtering of the sum of a 45-Hz and a 49-Hz sine wave). This has been observed with other combinations of frequencies. Here we define the prediction transfer as the ratio between the predicted PSD and the non-predicted. Because each frequency in the output is the contribution of many frequencies in the input including itself, if the prediction transfer at a certain frequency is below unity, we can safely say that such a frequency must have been attenuated by the ANN.

We have shown that the frequency transfer of the ANN predictor is nonlinear. This may complicate the interpretation as each frequency of the output may come from a variety of frequencies. However as explained above, the PSD of the output is a valid

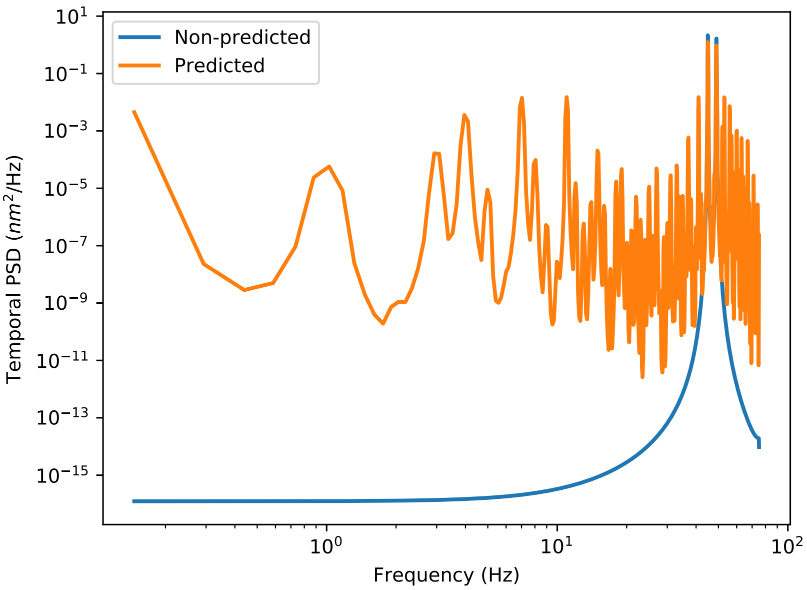


Figure 4.6: Temporal PSD of the sum of two sine waves (at 45 and 49 Hz respectively) and that of the ANN prediction.

concept regardless of the linearity of the system, and the prediction transfer of the ANN is reliable at least in indicating frequency attenuation. A prediction transfer of unity may also imply the capability of learning the input power spectrum. Thus, for the following analyses we regard the prediction transfer as a generalised transfer function for a nonlinear system.

4.2.2.2 Temporal Response with WFS Measurements

To obtain the PSDs of the predicted WFS slopes, in each of the GS magnitude conditions, we generated a 10,000-frame (66.7 s) slope sequence. All simulation parameters are as listed in Table 3.2, except that an infinite phase screen was used to satisfy the duration of this simulation (Assémat et al., 2006). The wind speed is 15 m/s along 0 deg. Initially, a 4096 × 4096 phase screen was generated using the Fast Fourier Transform (FFT) approach described in Section 2.3.2 as a stencil. This is roughly 11 times of L_0. Using the infinite phase screen technique, each new column of the phase screen is determined from several recently generated columns given the turbulence statistics. An infinitely long phase screen can be generated this way. Compared with the FFT approach, this technique is computationally expensive, but reduces memory required to store the phase screen, and allows Monte Carlo simulations lasting for minutes or even hours of simulated time. The slopes were then converted to Zernikes via $\mathbf{M}_{rz}$.

Fig. 4.7 displays the temporal PSDs of the zero-delay slopes of low-order Tilt (Zernike mode 3) and higher-order Trefoil (mode 10). As the noise level increases, PSDs at high temporal frequencies are gradually flattened, deviating from the exponential decay. This indicates the contamination of high frequencies by WFS noise. It is especially visible for frequencies higher than 30 Hz on the PSDs when the GS magnitude is 10.

Fig. 4.8 displays the prediction transfer of Tilt and Trefoil. Similar trends were observed in other modes. The ratio of the **Mag-10** predictor stabilises at 1, decreasing slightly above 10 Hz in noisier conditions, implying mild suppression of high temporal frequency aberrations in addition to predicting low frequencies well. It is clear that using the predictor trained on noisier data results in a better overall performance, and the negative slope in the transfer at frequencies above 10 Hz is indicative of noise filtering. This ratio of the **Mag-8** and the **Noise-free** predictors at high temporal frequencies exceeds 1, especially when the magnitude increases. Due to the nonlinear nature of the ANN frequency transfer, we cannot deduce at this stage that both predictors amplify noise from this plot. However, we will prove this in the next section.

This behaviour with high frequencies is observed in other Zernike modes as well. See Fig. 4.9 for the transfer of modes 2-36 in the noise-free condition. We have also observed that the low frequencies of some modes are magnified by all predictors and that this magnification can transit between modes as wind direction varies and can even vanish for certain directions. This might also explain the poorly corrected modes suggested in Fig. 4.3 where the wind direction is 0 deg. The ANN predictors all show this same property that we hypothesise is due in part to a Zernike term's

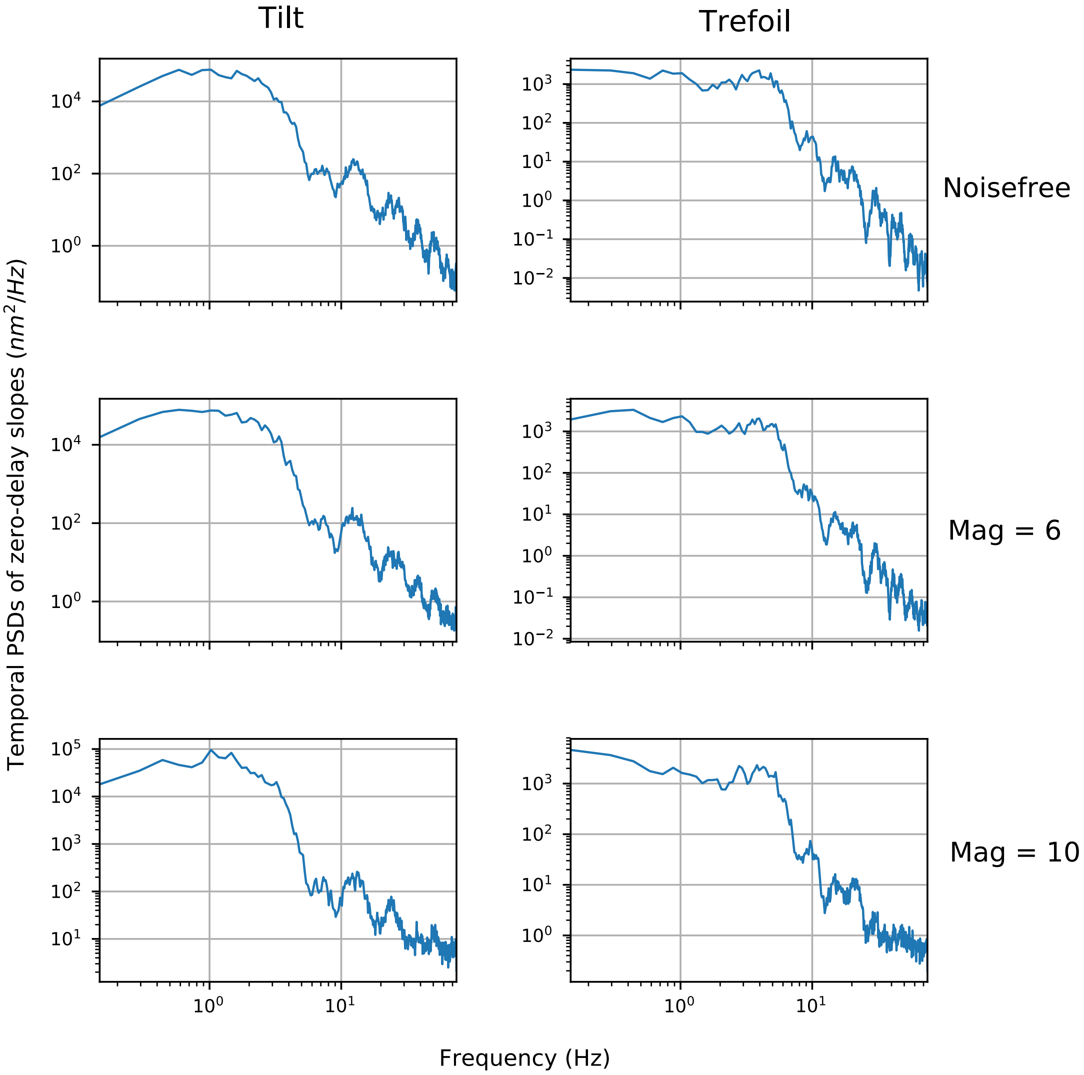


Figure 4.7: Temporal PSDs of Tilt and Trefoil of the zero-delay slopes as guide star magnitude varies. The flattening at high frequencies indicates the contamination by WFS noise.

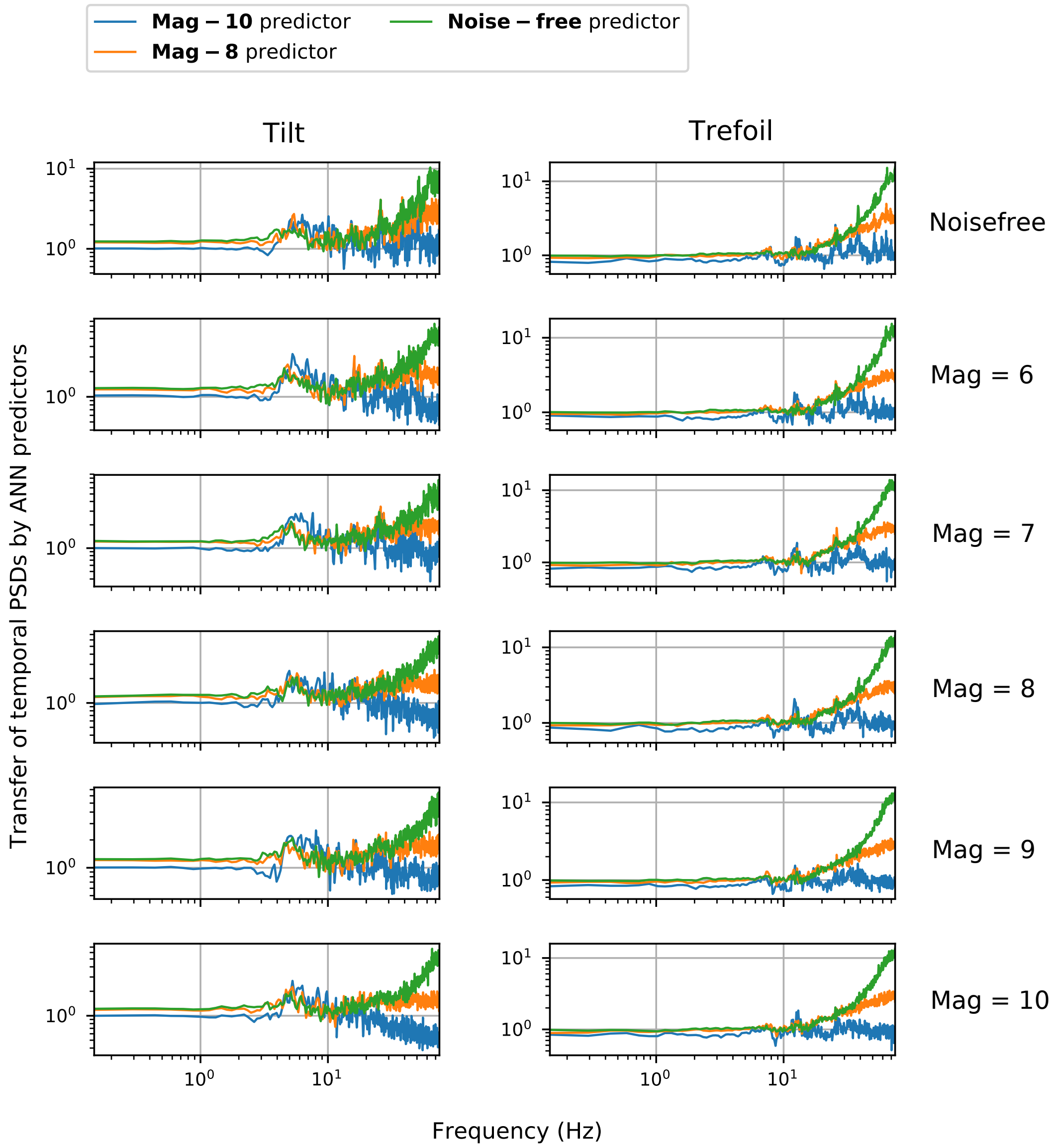


Figure 4.8: Prediction transfer in temporal PSDs of Tilt and Trefoil by **Mag-10**, **Mag-8** and **Noise-free** predictors as GS magnitude varies.

time-averaged amplitude being dependent on wind direction and not speed alone (Gordon et al., 2011). See Appendix A for the prediction transfer as wind direction varies and a brief description there.

4.2.3 Noise Propagation

Recall from Section 2.2.6.4 that the noise-induced error can be estimated as the difference between the actual auto-correlation of slopes at $\Delta n = 0$ and a fitted parabola (representing the turbulence) around $\Delta n = 0$. Fig. 4.10 shows the auto-correlation of slopes measured by the 10^{th} subaperture (fully illuminated) with and without WFS measurement noise. In the noise-free condition, the auto-correlation peak at $\Delta n = 0$, which contains the power of the turbulence only, is 1.51×10^{-2}. The fitted peak using values at $\Delta n = 1$ and 2 is 1.50×10^{-2}, reducing slightly to 1.49×10^{-2} when fitted with $\Delta n = 2$ and 3. The relative parabola fitting error is 0.5% and 1.7% respectively. Thus, we consider the parabola fit a valid approximation at $\Delta n \leq 3$ in our simulations.

In the magnitude 10 condition, We observed slight deviation from a parabola at $\Delta n = 1$ and 0 of the predicted slopes. The derivation at $\Delta n = 0$ suggests that ANN predictors suppress or magnify noise to different levels. The deviation at $\Delta n = 1$ might imply the noise propagation between adjacent frames after filtering by the ANN.

Due to the noise propagation at $\Delta n = 1$, we use the auto-correlation values at $\Delta n = 2$ and 3 (instead of 1 and 2) to fit the parabola, and subtract the fitted curve from the actual auto-correlation to obtain the propagated noise to $\Delta n = 1$. We think that the noise is propagated between adjacent frames, to a much lesser extent to further separated frames as is supported by Fig. 4.10. The fitted parabolas for the non-predicted and predicted slopes are also shown in this plot.

Fig. 4.11 shows that non-zero noise propagation exists even in the zero-delay slopes, and this is indicative of the inaccuracy in the parabolic fit approach. We can

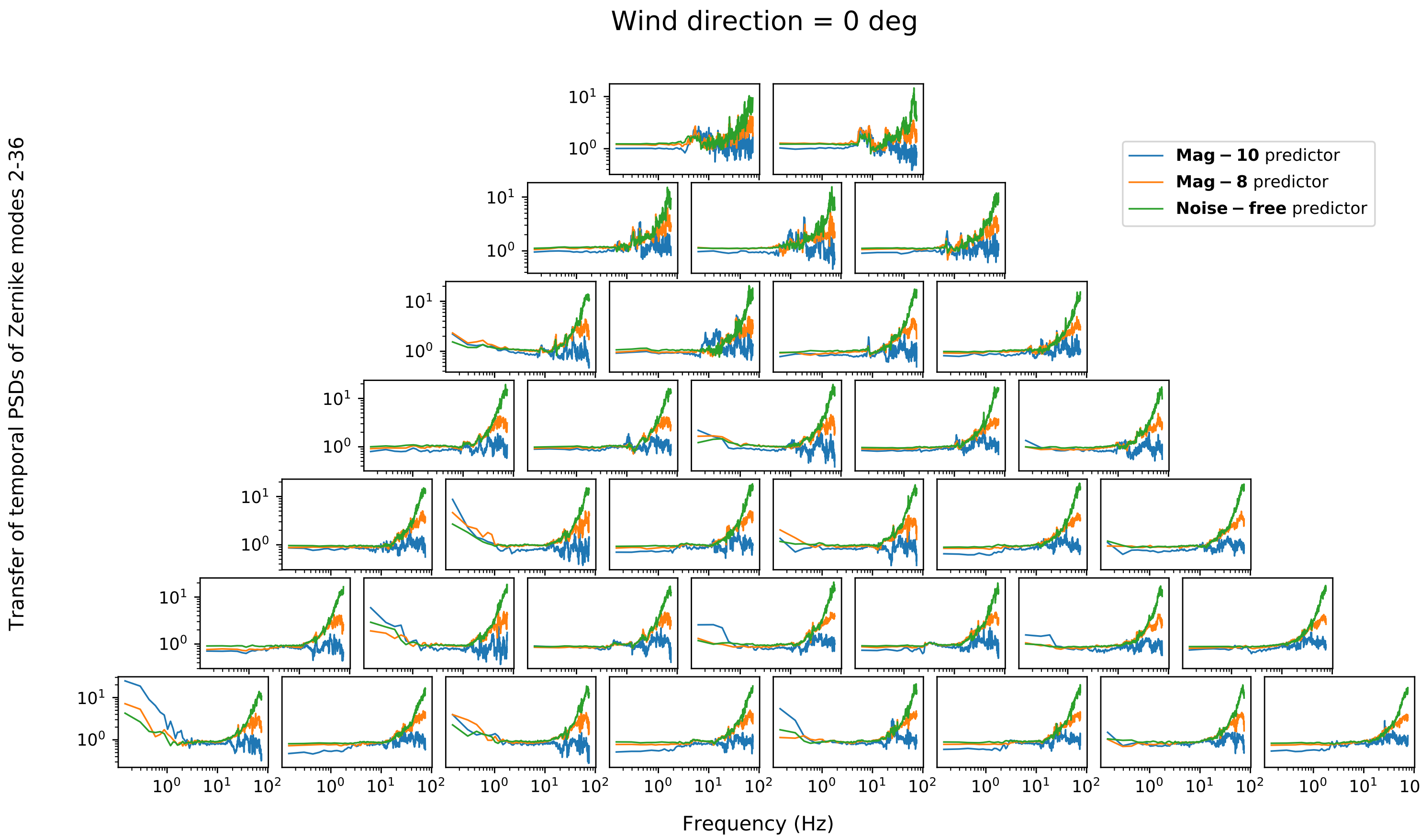


Figure 4.9: Prediction transfer of temporal PSDs in Zernike modes 2-36 (from top to bottom, left to right) by **Mag-10**, **Mag-8** and **Noise-free** predictors when the wind direction is 0 degree in the noise-free condition. This corresponds to the Zernike modes displayed in Fig. 2.1. Modes in the same row have the same radial order.

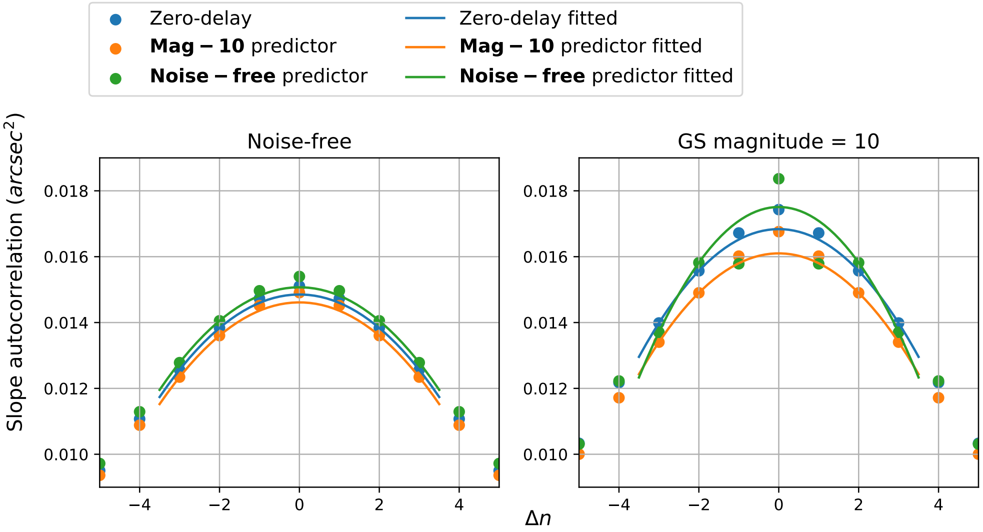


Figure 4.10: Auto-correlation of predicted and zero-delay slopes. The parabola is fitted using the auto-correlation values at $\Delta n = 2$ and 3. The gap between the actual auto-correlation at $\Delta n = 1$ and the fitted parabola indicates the noise propagation between adjacent frames.

therefore say that noise propagation by the **Mag-10** and **Mag-8** predictors are similar to that of the non-predicted slopes and their noise propagation can be considered negligible. Due to this, we can use the auto-correlation at $\Delta n = 1$ and 2 to fit the parabola and subtract the actual auto-correlation at $\Delta n = 0$ from this to obtain the noise-induced slope variance (see Fig. 4.12). The **Mag-10** predictor slightly suppresses WFS noise while the other two predictors amplify this. This together with Fig. 4.8 shows that the amplification in the power of the noise floor can be indicative of noise amplification, regardless of the linearity of the system transfer.

This is in line with Osborn et al. (2012), which shows that in noisy conditions an ANN tomographic reconstructor trained with photon noise has significantly better performance than an ANN reconstructor trained without any noise. The improved performance of the ANN tomographic reconstructor might as well have benefited from the noise filtering, potentially brought by the spatial averaging of multiple WFS measurements. This suggests that applying the ANN prediction technique to multi-GS systems, combining the ANN spatial filtering and temporal filtering,

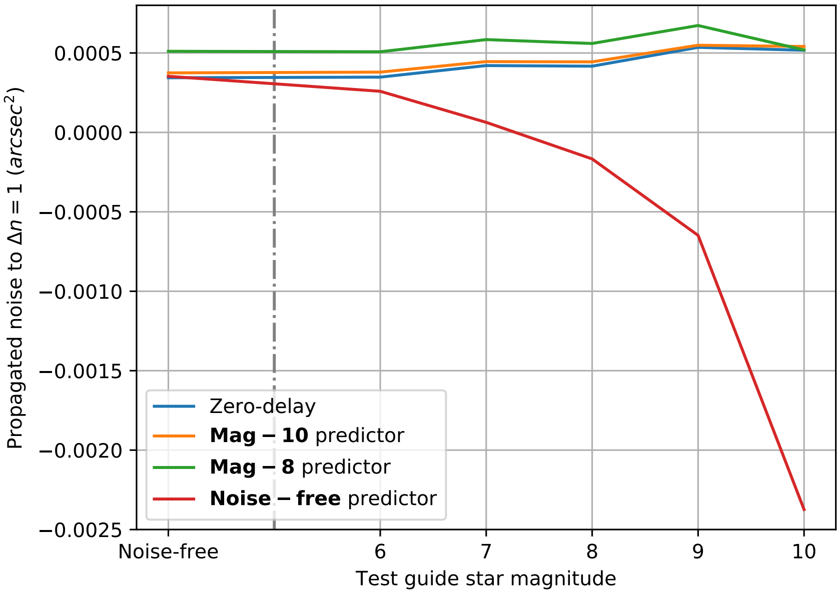


Figure 4.11: Noise propagation between adjacent frames by ANN predictors. This is represented by the deviation of the actual auto-correlation of slopes at $\Delta n = 1$ from the parabola fitted with values at $\Delta n = 2$ and 3 (shown in Fig. 4.10).

might improve the noise suppression even further.

4.2.4 Discussion

From the analyses presented in this section we have the first tests of our hypothesis that the ANN is capable of wavefront prediction. Whilst it is clear that using the ANN predictor results in a reduction in wavefront error compared to the slopes delayed by a single frame, this could also be due to a reduction in noise or aliasing errors. The nonlinear nature of the ANN does not allow us to easily investigate these errors in isolation, but we can compare some of the errors associated to the ANN prediction to that of the delayed slopes where the error budget can be more easily derived.

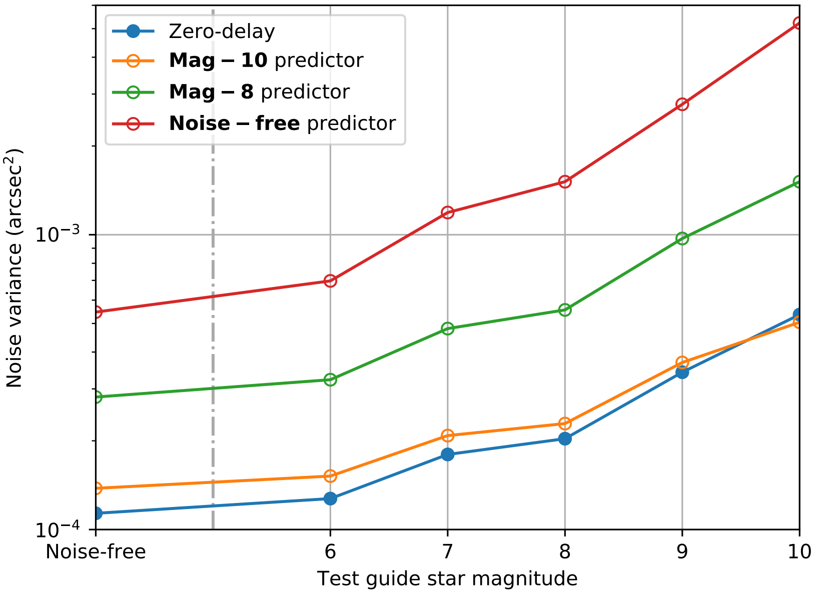


Figure 4.12: Measured noise variance of zero-delay and predicted slopes.

Recall from Eq. 4.5 the breakdown of the delay and prediction errors,

$$\begin{aligned}\sigma_d^2 &= \sigma_{\text{temporal}}^2 \\ \sigma_p^2 &= (\sigma_{\text{noise}'}^2 - \sigma_{\text{noise}}^2) + (\sigma_{\text{aliasing}'}^2 - \sigma_{\text{aliasing}}^2) + \sigma_{\text{temporal}'}^2.\end{aligned} \tag{4.10}$$

As a result,

$$\sigma_{\text{temporal}'}^2 = \sigma_p^2 + \sigma_{\text{noise}}^2 - \sigma_{\text{noise}'}^2 + \sigma_{\text{aliasing}}^2 - \sigma_{\text{aliasing}'}^2. \tag{4.11}$$

Consider the **Mag-10** predictor in the noise-free condition. σ_{noise} is 0 and $\sigma_{\text{noise}'}$ is thus 0. From Table 4.2 we have $\sigma_d = \sigma_{\text{temporal}} = 82.6$ nm and $\sigma_p = 39.3$ nm. The aliasing error can be estimated using Eq. 2.32. Given the DM spacing 0.6 m, $r_0 = 0.16$ m @ 500 nm, we have

$$\begin{aligned}\sigma_{\text{aliasing}}^2 &= 0.08 \left(\frac{d_{\text{DM}}}{r_0}\right)^{5/3} \\ &= 67.7^2 \ (\text{nm}^2).\end{aligned} \tag{4.12}$$

Eq. 4.11 then becomes

$$\begin{aligned}\sigma^2_{\text{temporal}'} &= \sigma_p^2 + \sigma^2_{\text{aliasing}} - \sigma^2_{\text{aliasing}'} \\ &= 39.3^2 + 67.7^2 - \sigma^2_{\text{aliasing}'} \\ &= 78.3^2 - \sigma^2_{\text{aliasing}'} \leq 78.3^2 \\ &< 82.6^2 = \sigma^2_{\text{temporal}},\end{aligned} \tag{4.13}$$

i.e. $\sigma_{\text{temporal}'} < \sigma_{\text{temporal}}$ and therefore we can conclude that ANN is predicting the wavefront.

In noisy conditions (GS magnitude = 6 to 10), Fig. 4.12 shows that $\sigma_{\text{noise}} \approx \sigma_{\text{noise}'}$ for the **Mag-10** predictor. As a result, the first equation in Eq. 4.13 still holds. Using σ_p and σ_d values in Table 4.2, the derivation in Eq. 4.13 can be easily followed. It can then be proven that $\sigma_{\text{temporal}'} < \sigma_{\text{temporal}}$ in all conditions except when GS magnitude is 9, where the sum of σ_p^2 and $\sigma^2_{\text{aliasing}}$ (72^2) is slightly higher than $\sigma^2_{\text{temporal}}$ (71.1^2). However, because Eq. 4.13 gives only the loose upper bound of $\sigma_{\text{temporal}'}$ and $\sigma_{\text{temporal}'} < \sigma_{\text{temporal}}$ holds for both magnitudes 8 and 10, the same conclusion should hold for magnitude = 9 as well.

The prediction power of the **Mag-8** and **Noise-free** predictors can be understood intuitively. Fig. 4.9 shows that high temporal frequencies will be amplified by both predictors. Noise and aliasing can both be considered high temporal frequency errors, but aliasing will be observed only on higher-spatial frequency terms. If these terms are amplified by the **Mag-8** and **Noise-free** predictors, as indicated in Fig. 4.9, we can assume that $\sigma^2_{\text{noise}} - \sigma^2_{\text{noise}'}$ and $\sigma^2_{\text{aliasing}} - \sigma^2_{\text{aliasing}'}$ should both be negative. In this case, from Eq. 4.11 and Table 4.2 we have,

$$\begin{aligned}\sigma^2_{\text{temporal}'} &= \sigma_p^2 + \sigma^2_{\text{noise}} - \sigma^2_{\text{noise}'} + \sigma^2_{\text{aliasing}} - \sigma^2_{\text{aliasing}'} \\ &< \sigma_p^2 < \sigma_d^2 = \sigma^2_{\text{temporal}},\end{aligned} \tag{4.14}$$

i.e. $\sigma_{\text{temporal}'} < \sigma_{\text{temporal}}$ in all conditions.

In the following six scenarios, we show the results obtained with **Mag-10** predictor only. We use a brighter guide star of magnitude 6 in all following scenarios to reduce the performance variations brought by WFS noise.

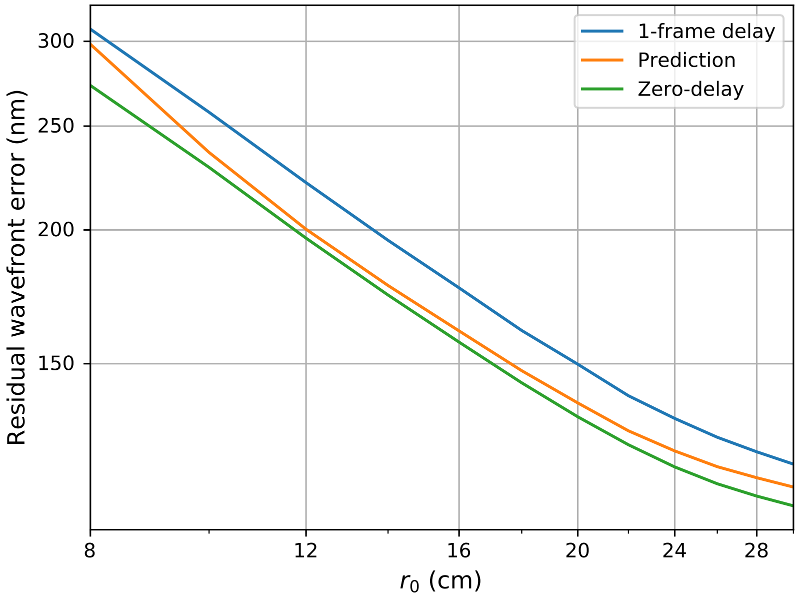


Figure 4.13: Residual WFE as a function of r_0. Guide star magnitude is 6. Wind speed is 15 m/s along a single direction.

4.3 Performance with Different Turbulence Strengths

Fig. 4.13 shows the residual WFE when r_0 of the test slope sequence varies from 8 to 30 cm. The predictor was trained with a median r_0 of 16 cm.

In this r_0 range the prediction performance is constantly better than the 1-frame delay. This implies that an ANN predictor trained with a fixed r_0 can generalise well to other turbulence strength levels, especially when the seeing condition is better than during training. When the seeing gets much worse(in our case below 10 cm), the sensitivity of the predictor to r_0 increases and a retraining dedicated for bad seeing should be considered once the performance becomes unacceptable.

4.4 Performance with Time-Variant Wind Velocity

In the above scenarios, we have assumed stationary turbulence. In this section, we demonstrate the agility and robustness of the ANN predictor against fluctuations

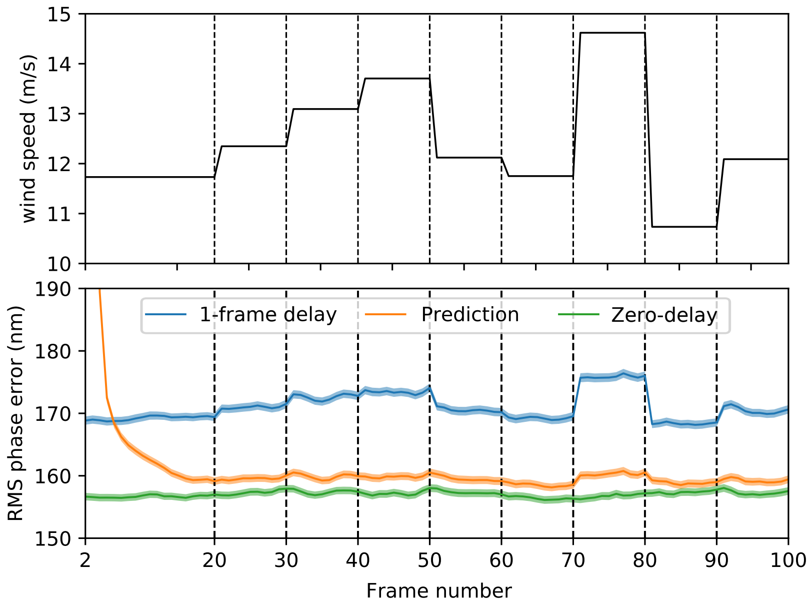


Figure 4.14: Robustness of the predictor against wind speed fluctuations between 10 and 15 m/s every 10 frames. Wind direction is 0 degree. Guide star magnitude is 6.

in wind velocity.

Here we use a synthetic wind speed sequence (upper panel in Fig. 4.14) in a relatively short time scale of 100 consecutive WFS frames (0.67 s). Once the prediction stabilises after the first 20 frames, wind speed changes every 10 frames (15 Hz) to a value within 10 and 15 m/s. This fluctuation is reflected in the dynamics of the 1-frame delayed correction, as a faster translation of the phase screen induces increased phase variations between adjacent frames under frozen flow.

Fig. 4.15 demonstrates robustness of the predictor against wind direction fluctuations between 0 and 45 degrees every 10 frames (upper panel). This corresponds to a maximum instantaneous change of 8.4 m/s in wind speed along a single direction.

Recently van Kooten et al. (2019) have used typical wind profiles from the Thirty Metre Telescope (TMT) site to demonstrate effects of wind velocity variations in a data-driven Linear Minimum Mean Squared Error (LMMSE) predictor over a period of 5 seconds in numerical simulations. Wind data are linearly interpolated

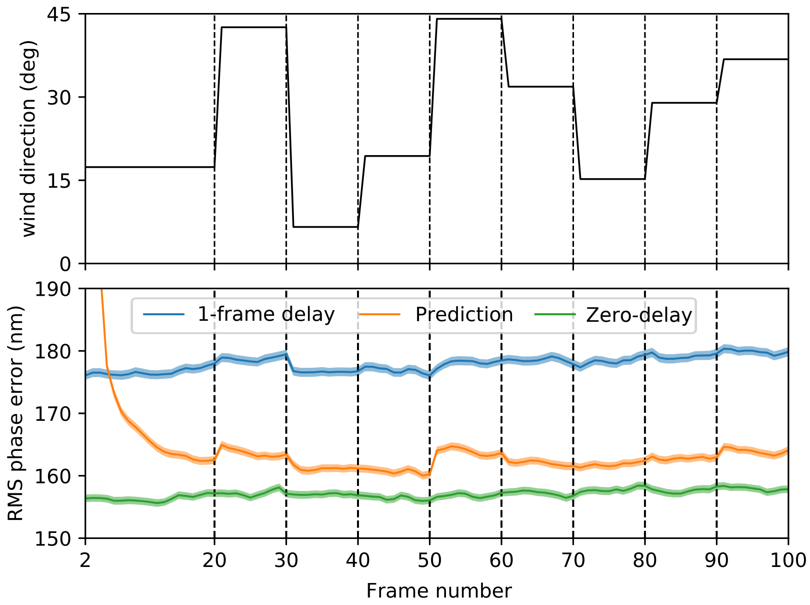


Figure 4.15: Robustness of the predictor against wind direction fluctuations between 0 and 45 degrees every 10 frames. Wind speed is 15 m/s. Guide star magnitude is 6.

to system frequency to allow for per-frame fluctuations. Two adaptive variations, resetting-batch LMMSE and forgetting LMMSE, along with LMMSE were tested. Compared with these linear predictors, the ANN predictor is more robust to wind fluctuations in that the variance of predicted WFEs did not increase significantly after the wind disturbance. This robustness can be explained as the ANN predictor is allowed to use more spatial and temporal information when making inferences. Furthermore, compared with the adaptive LMMSE the updating and forgetting mechanisms of the Long Short-Term Memory (LSTM) ANN are not fixed, but can constantly self-adjust according to the inputs, which by design allows for more flexible control on data flow.

4.5 Performance with Time-Variant Turbulence Strength

In this section we demonstrate the robustness of the predictor against realistic fluctuations in the turbulence strength r_0. The r_0 sequence was measured using the stereo-SCIDAR technique from on-sky data taken at La Palma in 2020 (Osborn et al., 2018). The time resolution is around 90 s, which is 13,500-frame long given the 150 Hz rate of our simulated system. The measured r_0 ranges from 12 to 23 cm (see the upper plot in Fig. 4.16).

The lower plot in Fig. 4.16 shows the phase errors for a continuous simulated time of one hour, with 40 different r_0 values. Each data point in this figure is the average value within 10 s. L_0 is 25 m. Wind speed is set to 30 m/s. An infinite phase screen with 4096 pixels in one dimension was simulated. The ANN predictor was not updated during this process. In the first 90-second slot and the fifth from last slot, the r_0 values are both around 12.6 cm. The average RMS errors in the prediction loop within both slots are 231.3 and 231.9 nm respectively, showing negligible degradation even with the ANN integration for a long period. This demonstrates the robustness of the ANN against realistic r_0 fluctuations without user tuning. Mean RMS WFEs of the delayed, predictive and zero-delay correction loops are 201.9, 189.4 and 148.5 nm respectively.

4.6 Performance with Multi-Layer Turbulence

Although we train the ANN with a single turbulence layer for the ease of training, there usually exist several layers at high altitudes in addition to a strong ground layer (Farley et al., 2018, 2019). It is thus meaningful to test the predictor's sensitivity to multiple layers moving with different velocities.

Here we show the results obtained with European Southern Observatory (ESO)

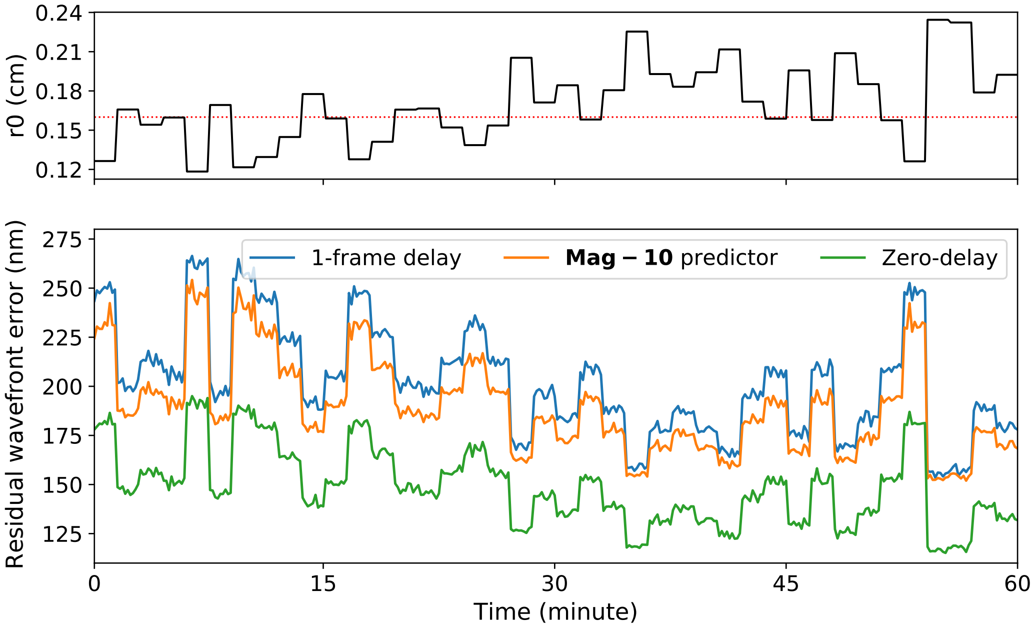


Figure 4.16: Robustness of the predictor against turbulence strength fluctuations every 90 s of simulated time (13,500 frames). Wind speed is 30 m/s. Guide star magnitude is 6. The internal states of the LSTMs were not reset during this process. The r_0 value (0.16 cm) at which the ANN was trained is represented by a red dotted line in the upper plot.

median 35-layer profile (Sarazin et al., 2013). r_0 is 0.157 m, slightly worse than during ANN training. L_0 is 25 m. We generated 1,000 slope sequences each of 100 frames with this profile. For comparison, we also generated the same amount of test data of a single ground layer and of a four-layer profile (detailed in Table 4.3), both moving at 9.21 m/s (slightly slower than the training range), which is equivalent to the dynamics of the 35-layer profile.

Fig. 4.17 shows residual WFEs when wind vectors of multi-layer profiles (either the 4-layer or the 35-layer) move in different directions. For the 35-layer profile, the moving direction of each layer is a random integer between 0 and 360 degrees. For the 4-layer profile, wind directions are listed in Table 4.3. The delayed and the zero-delay correction loops behave similarly regardless of the number of layers due to similar turbulence statistics used, thus only values obtained from the single-layer profile are displayed here. Mean RMS WFEs of the delayed, 35-layer predictive, 4-layer predictive, 1-layer predictive and zero-delay correction loop after the 20th

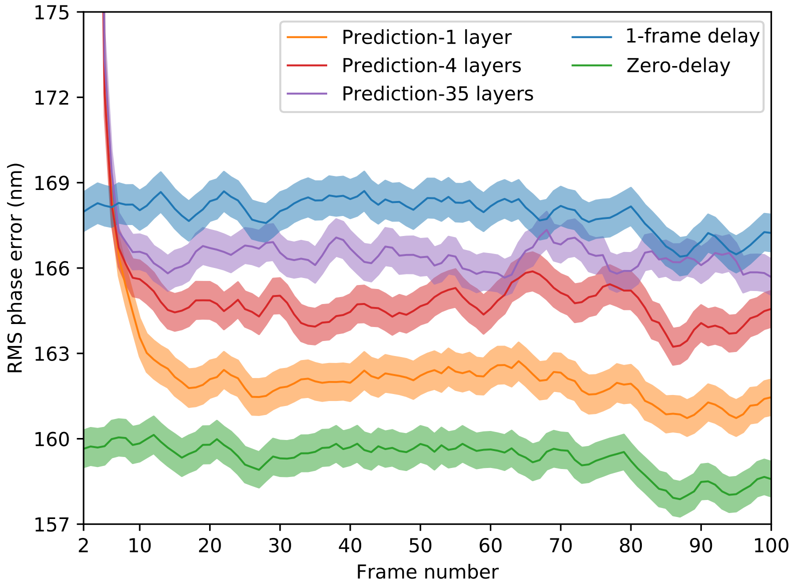


Figure 4.17: ANN performance with multiple turbulence layers moving along different directions. Wind speeds of either the 1- or 4-layer profile are scaled to maintain the same dynamics as that of the 35-layer profile. r_0 is 0.157 m.

frame are 167.9, 166.4, 164.6, 161.9 and 159.2 respectively.

Fig. 4.18 shows improved ANN performance when all layers in either multi-layer profile move in the same direction (wind speeds are the same as used in Fig. 4.17). Mean RMS WFE of the 35-layer predictive loop decreases to 164.0 nm, slightly better than the 4-layer predictive loop when wind vectors are largely distinct from each other. Mean RMS WFE of the 4-layer predictive loop decreases to 162.4 nm, approaching that of the 1-layer predictive loop.

We think that the wind directions adopted represent two extreme conditions, and that performance with real turbulence profiles would fall within these two cases. These results show that the predictor trained on a single layer frozen-flow conditions is capable of providing performance improvement even when complex profiles with random wind directions are encountered.

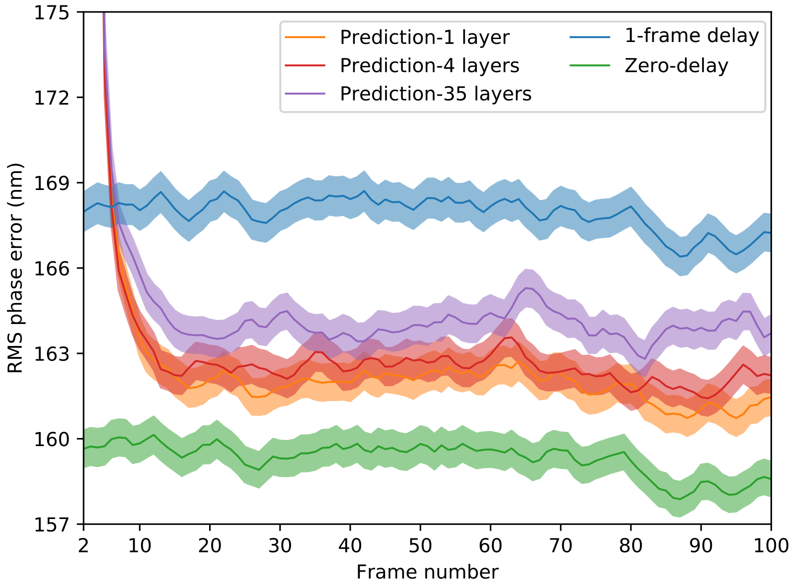


Figure 4.18: ANN performance with multiple turbulence layers moving along the same direction. Compared with Fig. 4.17, the ANN performance suffers from the increased number of wind vectors, but mainly from the variety among those vectors.

Table 4.3: Four-layer turbulence profile used for testing the predictor in a multi-layer scenario. r_0 is 0.157 m. L_0 is 25 m. Two sets of wind directions corresponding to Figs. 4.17 and 4.18 respectively are examined.

	Layer 1	Layer 2	Layer 3	Layer 4
Height (m)	0	4000	10000	15500
Relative strength	0.65	0.15	0.10	0.10
Wind speed (m/s)	7.6	9.5	11.4	15.2
Wind direction (degrees)	0	330	135	240
	0	0	0	0

4.7 Performance with Two-Frame Delay

Up to this point, we have considered only a 1-frame delay in the AO loop to account for WFS integration time only but ignored the time taken for real-time processing and the update of the surface shape of the DM.

In Fig. 4.19 we show the ANN performance when a more realistic loop delay of two frames is considered. We trained a separate ANN that was designed to predict two frames in advance in a single step. The training dataset described in Section 3.4.1

was re-utilised in the way that $(\boldsymbol{s}_1, \boldsymbol{s}_2, \dots, \boldsymbol{s}_{28})$ in each sequence are the ANN inputs and $\boldsymbol{s}_{30}$ is the training target. The training and hyperparameter tuning setup follows that described in Section 3.4.2. The resulting network comprises two stacked LSTM cells and a final Fully Connected (FC) layer. The output vector sizes of the two LSTMs are 122 and 171 respectively. The resulting mean RMS WFE of this single-step predictive loop after the 30th frame is significantly reduced to 166.9 nm, compared with 225.6 nm of the 2-frame delayed loop and close to 157.1 nm, the WFE corresponding to the zero-delay loop.

As a comparison, the 1-frame predictor can also be applied twice to provide a 2-frame prediction: first, the measured $(\boldsymbol{s}_1, \boldsymbol{s}_2, \dots, \boldsymbol{s}_t)$ $(t \geq 2)$ is fed into the predictor to generate the predicted $\tilde{\boldsymbol{s}}_{t+1}$ as it was designed; second, $\tilde{\boldsymbol{s}}_{t+1}$ is treated as its truth value $\boldsymbol{s}_{t+1}$ and forms part of the ANN input vector $(\boldsymbol{s}_1, \boldsymbol{s}_2, \dots, \boldsymbol{s}_t, \tilde{\boldsymbol{s}}_{t+1})$, which is then used to generate $\tilde{\boldsymbol{s}}_{t+2}$ as in the first step. This resulted in a WFE of 174.4 nm, slightly worse than the single-step prediction, however still significantly better than the 2-frame delay.

4.8 Performance of Non-Spatially Aware Predictor

For the **Mag-10**, **Mag-8** and **Noise-free** predictors we have examined so far, the ANN prediction of measurements by a single subaperture has been derived from past measurements of all subapertures. We call these *spatially aware* predictors. Including this level of spatial information allows for the exploitation of the spatio-temporal correlation of frozen flow. However, in van Kooten et al. (2020) measurements from surrounding subapertures did not bring performance benefits compared with a per-subaperture basis prediction by a linear data-driven predictor with on-sky SPHERE data, which is a High Contrast Imaging (HCI) instrument on the 8.2 m Very Large Telescope (VLT). In this section we examine the performance of a *non-spatially aware* predictor which predicts the slopes on a per-subaperture basis.

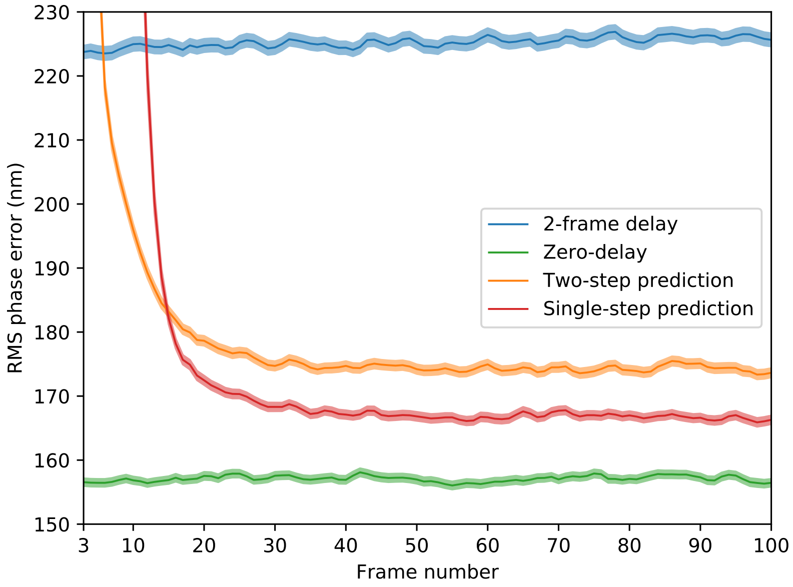


Figure 4.19: Prediction performance of a 1-frame predictor (two-step prediction) and a 2-frame predictor (single-step prediction) in a system with a 2-frame latency. The methodology adopted for the prediction in a 1-frame delay system is extended to training a separate ANN predictor to predict two frames in advance directly (single-step prediction). In this case, the 1-frame predictor can also be applied twice in sequence (two-step prediction), albeit with slightly worse performance. Both predictors significantly improve the system performance compared with the 2-frame delay. Guide star magnitude for test is 6. Wind speed is 15 m/s along a single direction.

Using the training dataset for **Mag-10** predictor, we trained an ANN predictor that receives the x and y slope of a single subaperture only, and predicts the x and y slope one frame in advance. In this case, slope measurements of a single subaperture are treated as a single training sample, thus one sample in the original dataset breaks into 36 samples (one for each subaperture). The first 10,000 sequences in the original dataset then yield a new training dataset of size 360,000, each sample being a 30-frame sequence composed of 2-element slope vectors. We again adopted the two stacked LSTMs structure. The number of neurons of each of the LSTMs were tuned in the range of 4 to 30 to match the reduced input/output size. Following the training and hyperparameter tuning procedure detailed in Section 3.4, we obtained what we refer to here as a non-spatially aware predictor. It has only 30 neurons in the first LSTM and 20 in the second. After training, 36 copies (one for each subaperture) of this predictor will be run in parallel, which is well suited to modern processor architectures.

Below we compare the performance of this predictor and its spatially aware counterpart, the **Mag-10** predictor, using approaches introduced in Section 4.2. Residual WFE is plotted in Fig. 4.20. Although not as performant as a spatially aware predictor, the non-spatially aware predictor improves system performance when GS magnitude is 9 or lower.

Fig. 4.21 displays the prediction transfer of temporal PSDs in Zernike modes Tilt and Trefoil by both predictors. The non-spatially aware predictor has a transfer gain of nearly unity below around 10 Hz, implying a nearly perfect restoration of slowly evolving components. However, it will significantly magnify fast-evolving components such as WFS noise or aliasing, which might explain its performance degradation as the GS magnitude increases. This is manifested in Figs. 4.22 and 4.23. These were measured using the auto-correlation approach detailed in Section 4.2.3. For the non-spatially aware predictor, both the noise propagation and the noise variance are much more severe than the spatially aware predictor under all test conditions.

4.9 Conclusions

We have shown in extensive numerical simulations the potential of ANNs as a nonlinear framework for wavefront prediction under the frozen flow hypothesis. The residual wavefront error of the simulated 7×7 subaperture SCAO system with 1-frame delay improves significantly after the predictor is incorporated irrespective of guide star magnitude and wind velocity. We have provided evidence that the ANN predictor reduces the temporal error.

For the ANN architecture studied within this thesis, the key parameter that affects ANN performance is the level of noise present in the training data set, with ANNs

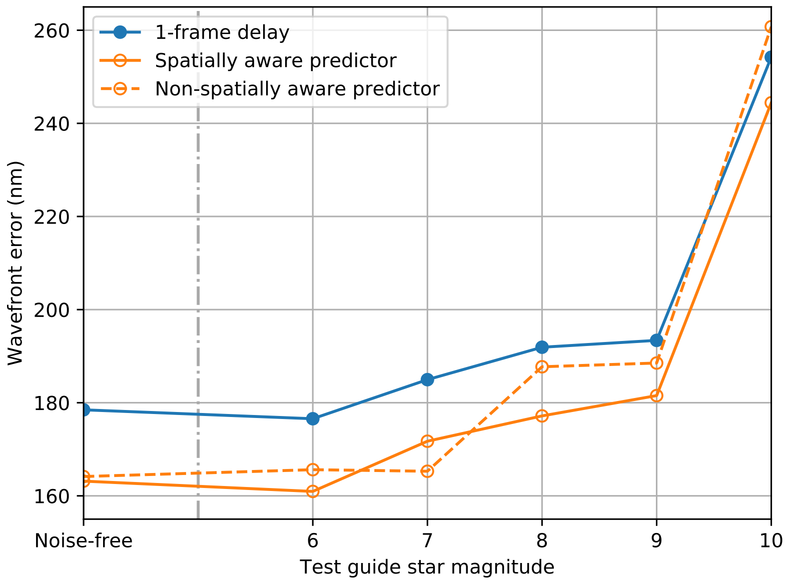


Figure 4.20: Performance of a spatially aware (**Mag-10** predictor) and a non-spatially aware predictor when the test guide star magnitude varies. Both predictors were trained with the same level of WFS noise.

trained on noisier data sets providing better system performance. The ANN is relatively insensitive to other parameters, such as turbulence strength and wind velocity. Sensitivity to the internal configuration of the ANN in terms of number of neurons and hidden layers was not explicitly investigated, but was optimised via the hyperparameter tuning step.

In addition to accurately predicting the wavefront, we have provided evidence that the ANN predictor also compensates for some noise and/or aliasing errors that can be temporally filtered from the wavefront. This behaviour however is dependent on the ANN training regime and was only observed when the system was trained on a faintest 10$^{\text{th}}$ magnitude guide star.

We have shown that the ANN predictor trained on a single atmospheric turbulence layer is also capable of predicting under more complex conditions with multiple layers with independent wind vectors, albeit with reduced performance. The ANN approach taken with a 1-frame delay is transferable to systems with a 2-frame

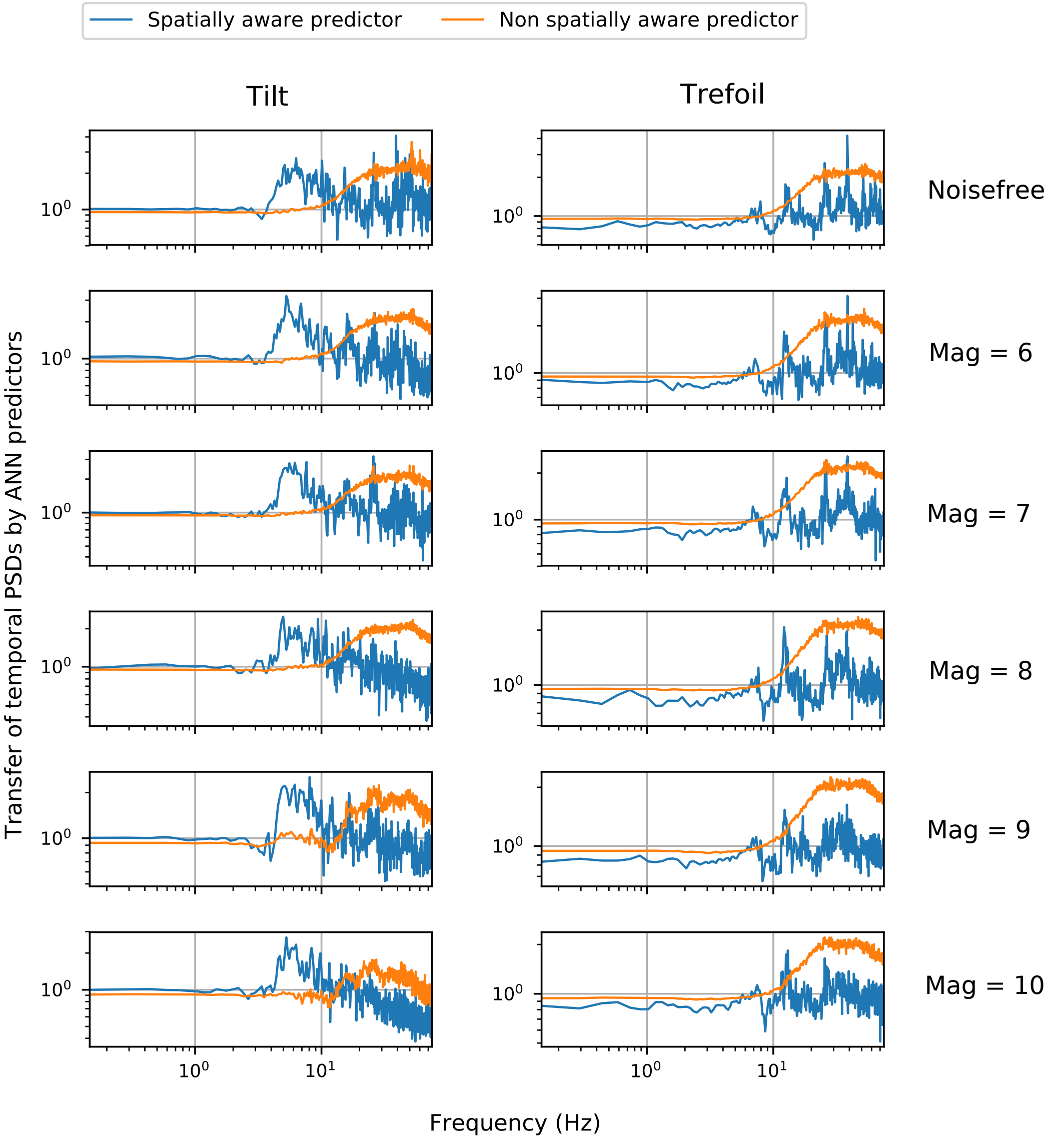


Figure 4.21: Prediction transfer of temporal PSDs of Tilt and Trefoil by a spatially aware predictor and a non-spatially aware predictor as guide star magnitude varies.

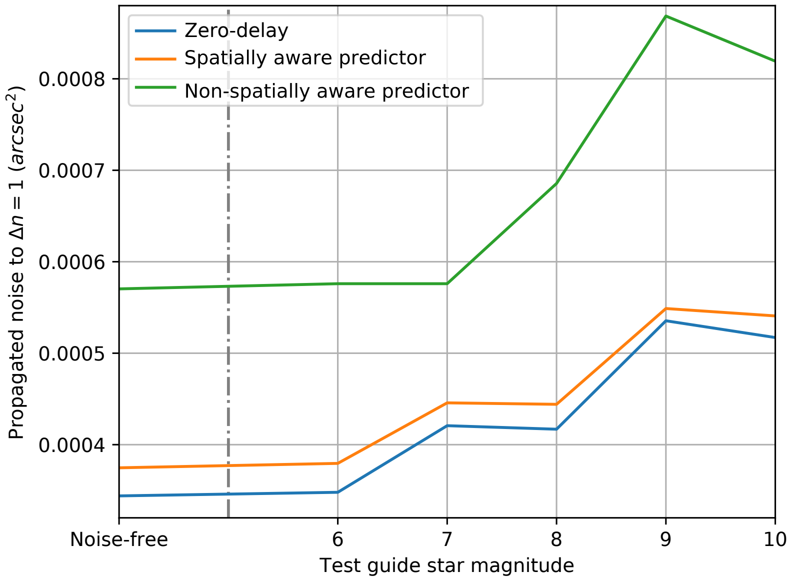


Figure 4.22: Noise propagation between adjacent frames by the spatially aware and the non-spatially aware predictors.

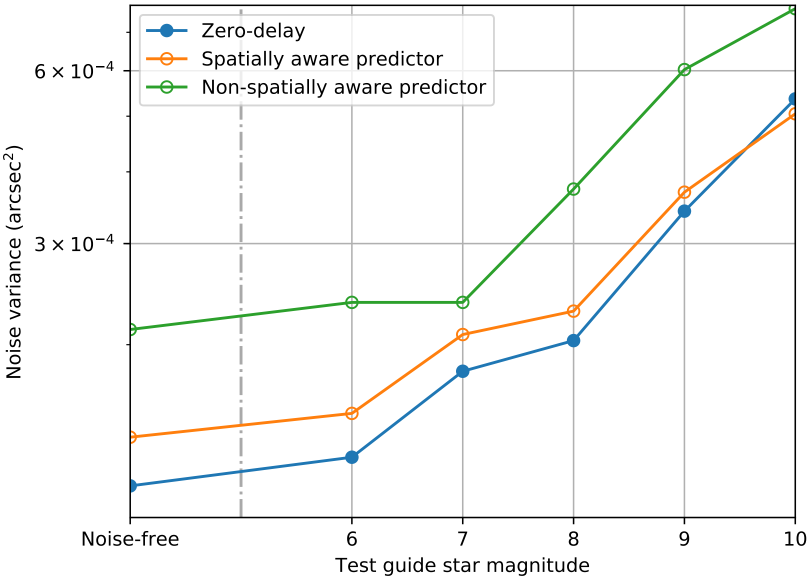


Figure 4.23: Measured noise variance of the zero-delay slopes and slopes predicted by the spatially aware and the non-spatially aware predictors as guide star magnitude varies.

delay.

We have also shown that a non-spatially aware predictor is capable of predicting the frozen flow turbulence. Although the non-spatially aware predictor tends to amplify the noise and aliasing which then degrades its performance, it can predict the slowly-evolving components (below around 10 Hz) with more accuracy in all test conditions compared with the spatially aware predictor.

4.10 References

V. Akondi and A. Dubra. Average gradient of zernike polynomials over polygons. *Opt. Express*, 28(13):18876–18886, 2020.

F. Assémat, R. W. Wilson, and E. Gendron. Method for simulating infinitely long and non stationary phase screens with optimized memory storage. *Opt. Express*, 14(3):988–999, 2006.

O. J. D. Farley, J. Osborn, T. Morris, M. Sarazin, T. Butterley, M. J. Townson, P. Jia, and R. W. Wilson. Representative optical turbulence profiles for ESO Paranal by hierarchical clustering. *Monthly Notices of the Royal Astronomical Society*, 481(3):4030–4037, 2018.

O. J. D. Farley, J. Osborn, T. Morris, T. Fusco, B. Neichel, C. Correia, and R. W. Wilson. Identifying optical turbulence profiles for realistic tomographic error in adaptive optics. *Monthly Notices of the Royal Astronomical Society*, 488(1): 213–221, 2019.

J. A. Gordon, D. F. Buscher, and F. Baron. Long-exposure filtering of turbulence-degraded wavefronts. *Appl. Opt.*, 50(27):5303–5309, 2011.

R. Landman, S. Y. Haffert, V. M. Radhakrishnan, and C. U. Keller. Self-optimizing adaptive optics control with reinforcement learning. In *Adaptive Optics Systems VII*, volume 11448, pages 842 – 856. SPIE, 2020.

J. Osborn, F. J. D. C. Juez, D. Guzman, T. Butterley, R. Myers, A. Guesalaga, and J. Laine. Using artificial neural networks for open-loop tomography. *Opt. Express*, 20(3):2420–2434, 2012.

J. Osborn, R. W. Wilson, M. Sarazin, T. Butterley, A. Chacón, F. Derie, O. J. D. Farley, X. Haubois, D. Laidlaw, M. LeLouarn, E. Masciadri, J. Milli, J. Navarrete, and M. J. Townson. Optical turbulence profiling with Stereo-SCIDAR for VLT and ELT. *Monthly Notices of the Royal Astronomical Society*, 478(1): 825–834, 2018.

M. Sarazin, M. Le Louarn, J. Ascenso, G. Lombardi, and J. Navarrete. Defining reference turbulence profiles for e-elt ao performance simulations. In *Proceedings of the Third AO4ELT Conference*, 2013.

M. van Kooten, N. Doelman, and M. Kenworthy. Impact of time-variant turbulence behavior on prediction for adaptive optics systems. *J. Opt. Soc. Am. A*, 36(5): 731–740, 2019.

M. van Kooten, N. Doelman, and M. Kenworthy. Robustness of prediction for extreme adaptive optics systems under various observing conditions. *Astronomy and Astrophysics*, 636:A81, 2020.

P. D. Welch. The use of Fast Fourier Transform for the estimation of power spectra: A method based on time averaging over short, modified periodograms. *IEEE Transactions on Audio and Electroacoustics*, AU-15(2):70–73, 1967.

Chapter 5

Prediction Results with CANARY Data

5.1 Overview

In this chapter, we will investigate if an Artificial Neural Network (ANN) can be used for wavefront prediction with on-sky data. We will apply the ANN prediction technique to Wavefront Sensor (WFS) data taken using the CANARY Adaptive Optics (AO) system, recorded between 28 September and 2 October, 2017, using an on-axis 7×7 Shack-Hartmann Wavefront Sensor (SHWFS) operating in open loop wavefront sensing. In this chapter we wish to address the following key questions:

1. Does the training methodology from Chapter 3 that works in simulation translate to real data?
2. Are there identifiable properties of the CANARY data that can explain any variations in between expected and actual performance?
3. Should an ANN predictor be trained in simulation or with real data?

We first introduce the CANARY instrument, with a focus on the system and atmospheric characteristics when the datasets were acquired. We show that CANARY sees strong evolving static turbulence that is indicative of dome seeing as opposed to frozen flow. We then present a static turbulence model that better describes the temporal evolution of observed data than the frozen flow hypothesis and demonstrate performance benefits brought by training with the fitted turbulence model. Specifically, we investigate how the training noise level and spatially awareness of the ANN can impact the prediction performance with dome seeing.

5.2 CANARY and Dataset Description

CANARY is a single-channel Multi-Object Adaptive Optics (MOAO) demonstrator for Extremely Large Telescope (ELT) instruments, hosted by the 4.2 m William Herschel Telescope (WHT), on La Palma in the Canary Islands (Myers et al., 2008; Morris et al., 2010). MOAO is an AO configuration correcting simultaneously for several individual lines of sight over a large Field of View (FOV), using wavefront information from multiple WFSs observing within that field (Assémat et al., 2007). CANARY was initially designed to demonstrate the concept of MOAO by correcting for on-axis turbulence using off-axis wavefront information.

The development of CANARY was phased. In its initial phase in 2010, CANARY implemented MOAO using a 52-actuator piezoelectric Deformable Mirror (DM) controlled in open loop. The open-loop control signal was generated using the Learn and Apply algorithm with information from three off-axis 7×7 Natural Guide Star (NGS) SHWFSs (Vidal et al., 2010). An on-axis reference Truth Sensor (TS) (a SHWFS) observed the corrected residual wavefront, providing a performance measurement. In its later phases, CANARY was upgraded with additional four off-axis Laser Guide Star (LGS) SHWFSs, bringing the total number of reference sources to seven, a higher-order DM and upgraded WFSs with double the number of subapertures across the pupil, gradually matching the MOAO architecture of the proposed MOSAIC ELT instrument. The concept of MOAO has been successfully demonstrated on-sky in both NGS-only and mixed NGS/LGS modes (Gendron

Table 5.1: Timestamps of CANARY Datasets 0-5.

Dataset No.	Timestamp
0	2017-09-28-21h20m20s
1	2017-09-28-23h10m34s
2	2017-09-29-03h46m09s
3	2017-09-29-05h26m20s
4	2017-10-02-01h43m38s
5	2017-10-02-02h04m08s

et al., 2011; Morris et al., 2013). CANARY was not designed for astronomical science use, but for the study of AO system performance. CANARY performance with the low-order DM results in an H-band Strehl ratio of 0.2-0.4, which is low compared to modern AO systems, but sufficient for the characterisation of MOAO performance. Its enormous flexibility by design has been used for the investigation of other ELT related problems such as the design of the real time controller (Basden and Myers, 2012) and the use of LGSs (Bardou et al., 2018), novel AO topics such as nonlinear tomographic wavefront reconstructors (Osborn et al., 2012), vibration mitigation techniques (Sivo et al., 2014) and automated wind velocity profiling (Laidlaw et al., 2019).

To quantify the ANN prediction performance with on-sky data, we used six open-loop slope Datasets taken by the TS of the CANARY instrument between 28 September and 2 October, 2017. These Datasets each last for 10,000 frames, which correspond to 66.7 s in time given the 150 Hz system frequency. These were recorded for turbulence profiling for the study of elongated LGS wavefront sensing technique taken around the same time (Bardou et al., 2018).

Timestamps for these Datasets are shown in Table 5.1, representing the time that the final frame in the dataset was recorded. For the ease of representation, we will refer to these as Datasets 0-5.

Characteristics of the TS are listed in Table 5.2. It has 7×7 subapertures, 36 active (each with over 50% illumination, providing a valid slope measurement). Each subaperture has a FOV of 3.87 arcsec, sampled by 16×16 pixels on the

Table 5.2: Principal AO system parameters of CANARY Datasets 0-5. This is also the configuration of the simulated Single-Conjugate Adaptive Optics (SCAO) system used within this chapter to generate ANN training data that are for the prediction of CANARY data, except that the loop latency is an integer 2 frames in the simulated system for simplicity.

Parameter	Value
System frequency	150 Hz
Loop latency	2.2 frames
Telescope diameter	4.2 m
Central obscuration	1.2 m
Truth Sensor type	SHWFS
# of subapertures	7×7 (36 active)
Readout noise	0.3 e^- RMS
# of pixels per subaperture	16×16
Subaperture pixel scale	0.24 arcsec
Subaperture FOV	3.87 arcsec
Real-Time Control (RTC) in open loop	Yes
Centroiding algorithm	Brightest pixel selection
# of brightest pixels selected	12

Charged Coupled Device (CCD) detector. The pixel scale is thus 0.24 arcsec. The detector has a readout noise of 0.3 electron Root Mean Squared (RMS). Wavefront slopes were determined using the brightest pixel selection algorithm described in Section 2.2.2, with 12 brightest pixels being used within any subaperture. For the 6 Datasets, the DM was inactive with actuators set to midrange values.

Observational parameters describing the turbulence conditions of the six Datasets are listed in Tables 5.3 and 5.4. Turbulence profiles were measured using the Learn 3 Steps (L3S) approach upgraded from the Learn and Apply algorithm (Martin et al., 2016), using information from the TS and 3 off-axis WFSs. The L3S approach dissociates the identification of slowly evolving turbulence terms and fast evolving terms in order to speed up the fitting process. The turbulence is measured for fixed altitudes between 0 and 18 km, with a resolution of 2 km. This procedure also yields turbulence strength of each layer and the integrated r_0 can be deduced using Eq. 2.39. All profiles show a strong ground layer turbulence.

[*] Wind direction 0 deg corresponds to wind blowing from the north while 90 deg corresponds to wind blowing from the east.

Table 5.3: Principal observational parameters of CANARY Datasets 0-5.

Dataset No.	0	1	2	3	4	5
GS magnitude (V)	9.66	10.83	10.83	9.47	10.92	10.92
r_0 (cm)	9.76	13.90	17.22	10.79	10.79	22.00
Ground-layer r_0 (cm)	10.90	14.67	21.45	11.57	12.09	30.12
Ground-layer wind speed (m/s)	3.23	5.34	8.21	5.14	3.89	0.87
Ground-layer wind direction (deg) *	-48.1	-66.6	-100.0	-79.0	89.5	138.3
# of photons/frame	8610	5060	4530	10030	2500	2560
# of photons/frame/fully illuminated subaperture	270	150	140	300	70	80
Measured noise variance ($\times 10^{-3}$) (arcsec^2)	2.5	2.4	8.3	12.0	20.1	12.8

Wind speeds and directions of the ground layer were taken from weather station archives of the Issac Newton Group of Telescopes [†], which were determined using the SLODAR technique (Wilson, 2002). High altitude wind information is not available. The number of photons collected per WFS frame was the average flux level of a separate 5,000-frame TS Dataset observing the same asterism, taken shortly before or after (within 17 minutes) the 10,000-frame Datasets used for the following prediction performance analyses. Table 5.3 also lists the measured noise variance from the 10,000-frame slopes using the approach detailed in Section 2.2.6.4. The discrepancy between the photon level and the noise level suggests the unreliability of the photon count, potentially due to the time difference between the Datasets 0-5 and the datasets used for determining the flux. We will therefore adopt the use of the excess noise variance when analysing the ANN performance with respect to the WFS noise level.

5.3 Temporal Properties of CANARY Data

In this section, we investigate if there is any temporal property of CANARY Datasets that deviates from the hypotheses used for the simulation in Chapter 4. We

[†] `http://catserver.ing.iac.es/weather/archive/index.php`

Table 5.4: Relative turbulence strength (%) of CANARY Datasets 0-5. Each column representing one Dataset sums up to 1.

Layer height (km)	0	1	2	3	4	5
0	83.4	91.4	69.3	89.0	82.7	59.4
2	6.9	3.1	11.4	3.7	9.3	11.6
4	0	0.1	0	0	0.1	2.0
6	1.4	0	0	0.1	1.4	3.3
8	2.9	2.1	0	0.1	0	5.0
10	0	0.1	0.1	2.3	0.1	4.3
12	0.1	0.1	0.5	0	2.3	9.8
14	0	0	0.1	0.1	0.6	0.8
16	0	3.0	0	0.2	2.6	0
18	5.3	0.1	18.6	4.5	0.9	3.8

have identified two of these through the examination of the covariance map and temporal Power Spectral Density (PSD) of the observed data respectively.

5.3.1 Covariance Map

For two two-dimensional discrete variables $\mathbf{A}$ and $\mathbf{B}$ of size $N \times N$, the element at (m, n) $(m, n = 0, 1, ..., 2N)$ of their covariance map $\mathbf{\Gamma}(\mathbf{A}, \mathbf{B})$ is defined as the correlation of their overlapping parts when one matrix is offset by $(m - N, n - N)$ with respect to the other, averaged across all time steps. This map describes the spatial correlation between these two variables. Values of large magnitude (either negative or positive) in the map implies a strong correlation between the corresponding parts of these two variables.

Under the frozen flow hypothesis, within short time scales the covariance map between two parts of the turbulence is equivalent to the temporal correlation of one part with itself given the correct time separation, as is detailed in Section 2.1.2 and in Wilson (2002). This spatio-temporal correlation can be reflected by the time-lagged auto-covariance map of open-loop slope measurements by a single WFS,

$$\mathbf{\Gamma}^{\mathbf{S}}_{\Delta t} = \mathbf{\Gamma}(\mathbf{S}(t), \mathbf{S}(t + \Delta t)), \tag{5.1}$$

where $\mathbf{S}(t)$ is a two-dimensional slope matrix with values being slope measurements projected on the corresponding subaperture positions. Either slopes in x or y direction can be used and the properties shown will be independent of this choice. In the analyses below we use slopes in x. Before calculating the covariance map of a slope sequence, the static component, which is the temporal mean of the slope measurement by an individual subaperture, should be subtracted from each subaperture to remove static aberrations that may be present in the Dataset. The global Tip and Tilt signal (different from the Zernike Tip and Tilt), which is the mean measurement of all subapertures at each time step, should also be subtracted to remove the effect of common motions such as wind shake (Butterley et al., 2006).

Comparison of the time-lagged covariance maps from CANARY Dataset 2 and those obtained through the simulation of a single frozen flow layer with Δt ranging from 0 to 0.2 s (30 frames) are shown in Fig. 5.1. The characteristics of the simulated layer are the same as the ground layer of Dataset 2 (which accounts for 69.3% of total turbulence). Other simulation parameters are the same as listed in Tables 5.2 and 5.3 for Dataset 2. In the lower panel in this figure, at $\Delta n = 0$, the covariance peak of the simulated layer appears at the centre, implying a perfect correlation with itself without any need of spatial displacement. This peak moves almost three pixels upward (which is slightly less than the size of three subapertures, 1.8 m) in 0.2 s. This is equal to the speed at which the turbulence moves (the wind speed is 8.21 m/s), though in the opposite direction (the wind direction is -100 deg, with 0 deg pointing to the right in the map and 90 deg upwards). Under the frozen flow hypothesis, when there exist multiple layers, several peaks of varied strengths moving with different velocities should be observed, each representing the relative movement of the WFS with respect to the corresponding layer (Wang et al., 2008).

Covariance maps of Dataset 2 are shown in the upper panel in Fig. 5.1. Contrary to the simulation assuming pure frozen flow, the covariance peak remains at the centre, although its intensity decays as Δn increases. The shape of the peak also varies with Δn, which might be indicative of a slowly moving layer. A static but

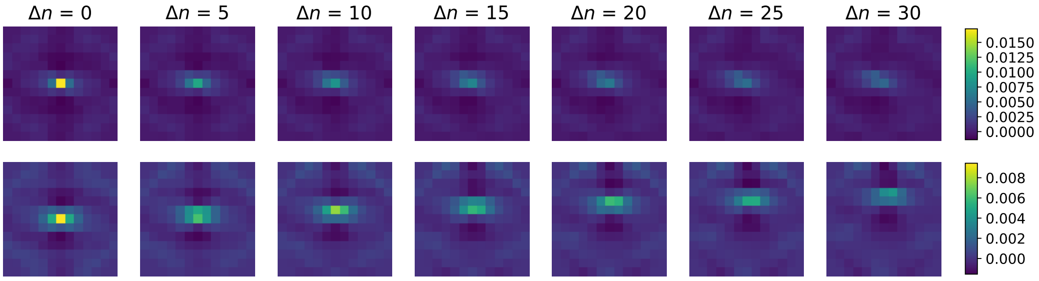


Figure 5.1: Covariance maps of CANARY Dataset 2 (upper row) and from the corresponding Soapy simulation (lower row) assuming a single frozen flow layer. The frozen flow layer simulates the ground layer of Dataset 2, moving at 8.21 m/s along the direction of -100 deg, which is in the opposite direction of the peak movement (in this plot, 0 deg corresponds to the right and 90 deg upwards). Δn in frame corresponds to a time lag of 0 to 0.2 s given a system frequency of 150 Hz. The peak movement shown with the simulation represents the frozen layer translation, while this is not observed with real data.

degrading central peak is observed for all Datasets. This absence of covariance translation implies that the turbulence observed by CANARY deviates from pure frozen flow matching the ground-layer wind speed measured at the local weather station.

5.3.2 Temporal Power Spectral Density

PSDs of Zernike modes Tip (left) and Tilt (right) are shown in Fig 5.2 for all Datasets (from top to bottom). Wide spikes around 1 Hz and a few narrow spikes between 10 and 30 Hz indicate perturbations induced by the telescope vibration. This can be caused by the telescope tracking error or wind shake of the telescope structure, and is common to most existing AO systems (Kulcsár et al., 2012). Such spikes were not observed in higher-order Zernike modes. This is another deviation from the temporal properties of the turbulence in simulation. Since we have not trained ANNs with vibrations, we will study how the ANN copes with these.

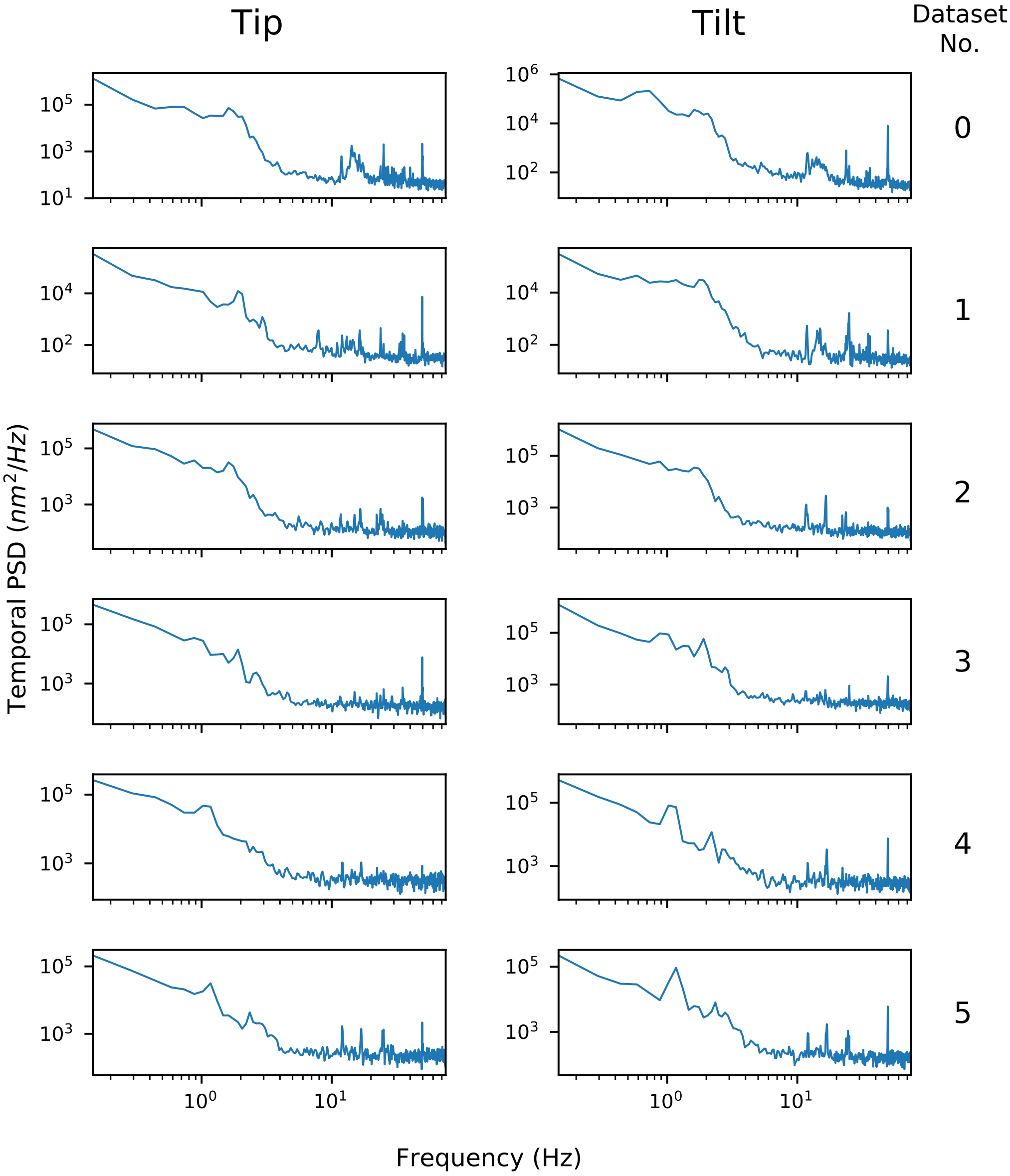


Figure 5.2: Temporal PSDs of CANARY Datasets 0-5 (from top to bottom) of Zernike modes Tip (left) and Tilt (right). There exist a wide peak at around 1 Hz and a few narrow peaks between 10 and 30 Hz in both modes in all Datasets, showing strong vibrations at these frequencies. This is not observed with higher-order modes.

5.4 Prediction Performance with Frozen Flow Predictor

We start with the methodology described in Section 3.4 to generate ANN training data in simulation with Soapy, and to train a frozen flow ANN predictor to predict the CANARY data.

We first determine the simulation parameters. We have adopted most of the CANARY parameters given in Table 5.2, except that a 2-frame instead of 2.2-frame latency is assumed for simplicity. We have shown in Section 4.7 that the ANN can be designed to predict two frames in advance directly. In all CANARY Datasets there exist multiple turbulence layers. Results in Section 4.6 have shown that the ANN predictor trained with a single turbulence layer can generalise to multi-layer profiles which have strong ground layers, albeit with reduced performance. Thus, here we train an ANN predictor that predicts two frames in advance directly assuming a single frozen layer. Sections 4.3 and 4.4 show that the ANN is much less sensitive to r_0 and wind speed than to the noise level. So here we maintain the choice of r_0 and wind velocity for training as was adopted in Section 3.4.

We have shown in Section 4.2 that the ANN training is highly sensitive to the WFS noise. Table 5.3 shows that CANARY data is much noisier than the simulated data used in Chapter 4 (recall from Fig. 4.12 that the noise variance of the noisiest Guide Star (GS) magnitude 10 condition is 5.4×10^{-4} arcsec^2, around one quarter of the least noisy CANARY Dataset 1). Here we configure the simulation to match the photon flux level of Dataset 3, which has a medium noise level among all Datasets. The resultant training data has a measured noise variance of roughly 2.8×10^{-4} arcsec^2, lower than all Datasets. This is roughly the noise variance of a magnitude 9 GS in simulations (see Fig. 4.12). Possible reasons for the discrepancy between flux level and measured noise variance have been discussed in Section 5.2. The impact of training noise level on the prediction performance will be explored in Section 5.7.2.

Table 5.5: Additional parameters (in addition to Table 5.2) used with the Soapy simulation for training the frozen flow predictor for predicting CANARY Datasets.

Parameter	Value
# of phase screens	1
Wind speed	10-15 m/s
Wind direction	0-360°
r_0 @ 500 nm	0.16 m
L_0	25 m
GS magnitude	9.47
Throughput	0.361
WFS wavelength	600 nm
RTC in open loop	Yes
Measured noise variance	2.8×10^{-4} arcsec^2

The simulation parameters used in this chapter are listed in Table 5.5.

In total we generated 200,000 training samples, with the second half of the dataset being the reversed sequence of the first half, which is equivalent to reversing the wind direction. Each sample is a time sequence of thirty 72-element vectors $(\boldsymbol{s}_1, \boldsymbol{s}_2, \dots, \boldsymbol{s}_{30})$. $(\boldsymbol{s}_1, \boldsymbol{s}_2, \dots, \boldsymbol{s}_{28})$ will be the ANN training input sequentially and $\boldsymbol{s}_{30}$ will be the training target.

The network comprises two stacked Long Short-Term Memory (LSTM) cells and an output Fully Connected (FC) layer as is shown in Fig. 3.7. The training and hyperparameter tuning principles are as described in Section 3.4.2, with a slight modification in the tuning of the hyperparameters: we use a recently developed hyperparameter tuning tool, Keras Tuner (O'Malley et al., 2019), to tune the numbers of neurons of both LSTM cells (an integer between 100 and 250, with a step of 5). The tuning routine set in this tool is random search (Bergstra and Bengio, 2012), similar in operation to that used in section 3.4.2. The total number of network configurations for evaluation is 20. The resulting optimal structure has 220 neurons in the first cell and 250 in the second.

After training, the predictor is applied to CANARY data directly. Its internal states are zeroed every time a different Dataset is input. At each time step, the

Table 5.6: The 2-frame delay error σ_{delay} (nm RMS) of CANARY Datasets and the corresponding prediction error σ_{pred} by a frozen flow ANN predictor.

Dataset No.	0	1	2	3	4	5
Two-frame delay	221.5	192.9	316.3	389.9	494.4	401.9
Frozen flow predictor	274.3	223.5	349.4	429.6	507.9	410.9

internal states from the previous time step and the input $\boldsymbol{s}_t$ $(t = 0, 1, ..., 9997)$ are combined to output the predicted $\tilde{\boldsymbol{s}}_{t+2}$.

Because the wavefront perturbation input to the AO system is not available as in Chapter 4, in this chapter we convert slope differences to an RMS wavefront error to quantify the prediction or delay performance, as was done in Section 4.2.1. This is obtained by first converting $\tilde{\boldsymbol{s}}_t - \boldsymbol{s}_t$ or $\boldsymbol{s}_{t-2} - \boldsymbol{s}_t$ to Zernikes via the calibrated control matrix $\mathbf{M}_{rz}$, then taking the square root of the sum of the temporal variance (averaged across each time sequence after the 30th time step by when the predictor has stabilised) of each Zernike mode (from mode 2 to 36). Recall from Eqs. 4.8 and 4.9 that

$$\begin{aligned}\sigma^2_{\text{delay}} &= \sigma^2_{\text{temporal}} + 2 \times \sigma^2_{\text{noise}} + 2 \times \sigma^2_{\text{aliasing}} \\ \sigma^2_{\text{pred}} &= \sigma^2_{\text{temporal}'} + \sigma^2_{\text{noise}} + \sigma^2_{\text{noise}'} + \sigma^2_{\text{aliasing}} + \sigma^2_{\text{aliasing}'},\end{aligned} \tag{5.2}$$

where σ_{delay} is the RMS 2-frame delay error and σ_{pred} is the RMS prediction error. $\sigma_{\text{temporal}'}$, $\sigma_{\text{noise}'}$ and $\sigma_{\text{aliasing}'}$ denote the ANN-filtered temporal, noise and aliasing errors respectively as was explained for Eq. 4.4. A prediction error σ_{pred} lower than the delay error σ_{delay} thus represents the ANN power in suppressing the temporal error as well as the noise and aliasing errors.

Table 5.6 gives the computed σ_{delay} of all Datasets and the corresponding σ_{pred} by the frozen flow predictor. The predicted performance is worse than the non-predicted in all cases. Degradation of the ANN performance could be due to the vibrations, or the inappropriate frozen flow hypothesis taken here for the ANN training. We will explore these in the following sections.

5.5 Modelling CANARY Data

In this section, we will concentrate on modelling the temporal characteristics of CANARY Datasets in order to generate more representative training data to improve the ANN performance.

Fig. 5.3 shows the covariance peak degradation of all Datasets in a time frame of 1 s (150 frames). At $\Delta n = 0$, this peak contains the power of both turbulence and WFS noise. At $\Delta n > 0$, because the noise is uncorrelated between frames, the peak contains power from turbulence only. For $\Delta n > 0$, Datasets 0 and 1 show an exponential decay while in other Datasets the decay is more curved.

We have suggested in Fig. 5.1 the presence of static turbulence and/or a slowly moving turbulence layer. In the following subsections, we will examine the above hypothesis by fitting the covariance peak profiles.

5.5.1 Fitting a Slow Frozen Flow Layer

We simulated two noise-free open-loop slope sequences lasting for 10,000 frames each, assuming CANARY wavefront sensing configuration. A single frozen layer is assumed, with wind speeds of 0.01 and 0.1 m/s along a single direction respectively. r_0 is 9.76 cm (that of Dataset 0). The screen size is 512 by 512 pixels, with the central 64 by 64 pixels being the position of the pupil. A log plot of the covariance peak degradation of both sequences is shown in Fig. 5.4. When the wind speed is 0.01 m/s, the covariance peak decay is exponential (as is observed in Datasets 0 and 1), however the decay is much slower with a gradient of magnitude 1.4×10^{-4} arcsec^2/s (regardless of the turbulence strength), compared with 10^{-2} of Dataset 0 and 4×10^{-3} of Dataset 1. When the wind speed increases to 0.1 m/s, the amount of decay also increases (1.4×10^{-3} in 1 s), but the shape of decay does not match any of these of CANARY Datasets shown in Fig. 5.3. When the wind speed is over around 1 m/s, the peak degrades rapidly within just a few frames. This implies that

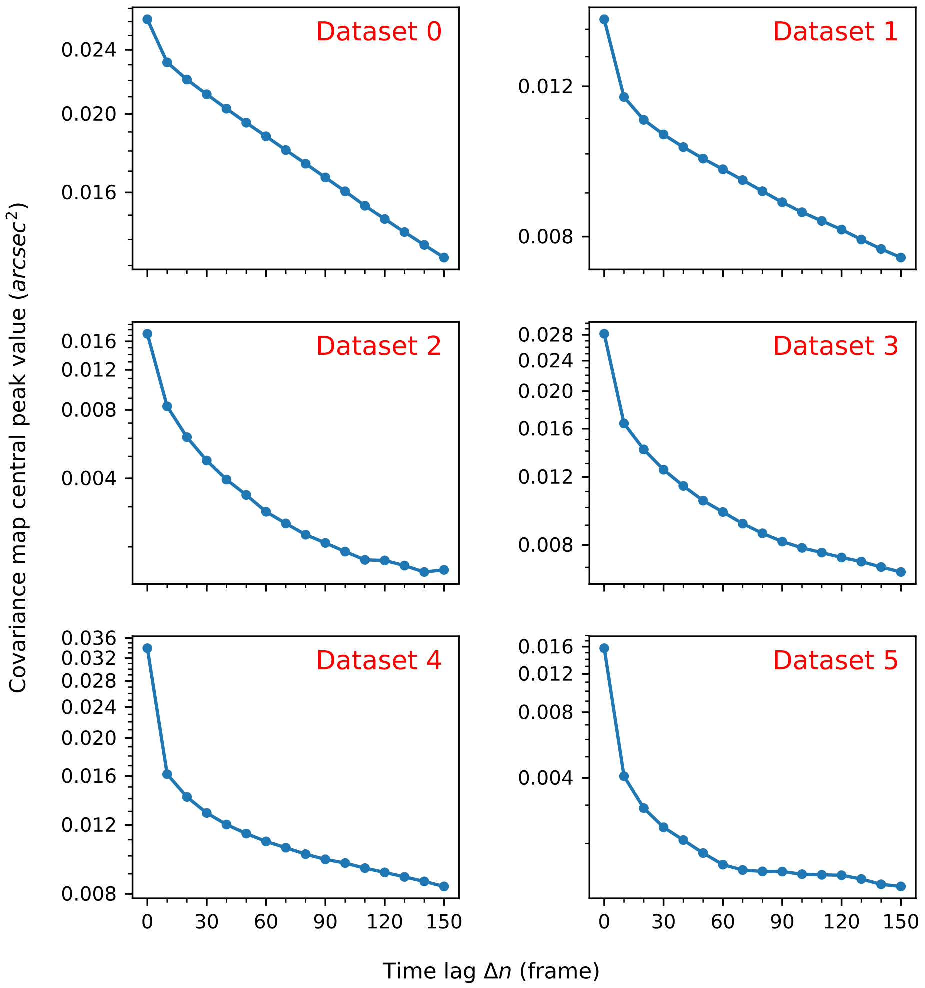


Figure 5.3: Peak value (arcsec^2) degradation of the covariance maps of CANARY Datasets 0-5. The time lag in frame corresponds to 0 to 1 s.

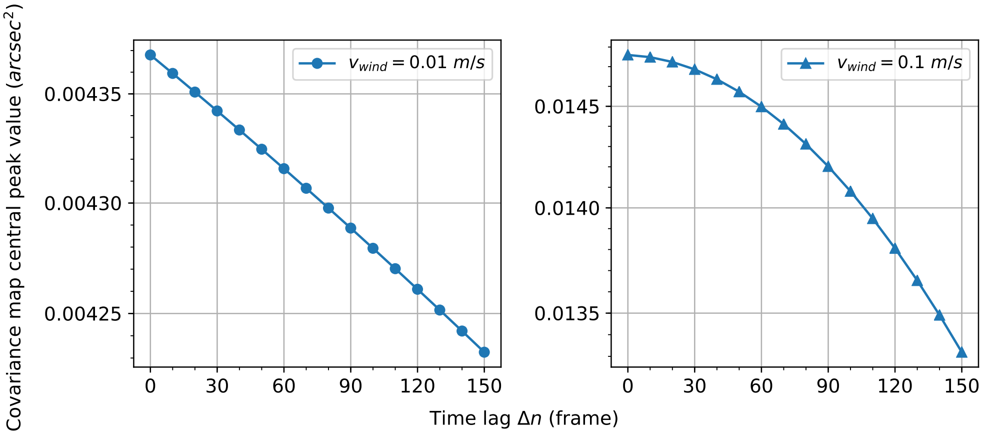


Figure 5.4: Covariance peaks assuming a slowly moving frozen flow layer with a wind speed of 0.01 (left) and 0.1 m/s. The y axis is in log scale. When the layer moves at 0.1 m/s or faster, the decay curve shape does not follow any of the CANARY Datasets. When the layer moves slowly enough, the decay is exponential, but its magnitude is far too small compared with CANARY Datasets.

the observed covariance peak decay profiles of CANARY data cannot be described by frozen flow alone, regardless of the wind speed.

One thing to note in Fig. 5.4 is the temporal mean across 10,000 frames of the slopes measured by one subaperture when the wind speed is 0.01 m/s is larger than when the speed is 0.1 m/s, due to the shorter spatial distance travelled by the turbulence in the former case, meaning that the average slopes are further away from zero. Since the temporal mean of slopes are subtracted before calculating the covariance, this results in the smaller covariance value at $\Delta n = 0$ when the wind speed is smaller, whereas we expect it to be affected by r_0 only when the slope sequence is long enough. Increasing the wind speed would make the temporal mean closer to the expectation, however the decay curve would deviate further from exponential. Thus the conclusion drawn from Fig. 5.4 is still valid. In this chapter all slope sequences (either observed or simulated) for covariance calculation will be of the same length (10,000 frames) to minimise this effect.

5.5.2 Modelling Dome Turbulence

A static covariance peak might be an indication of dome seeing, which has been identified by many on-sky observations (Avila et al., 2000; Shepherd et al., 2013; Guesalaga et al., 2014). Dome seeing is caused by mixing of air of different temperatures within and around a telescope dome structure (Basden et al., 2015), which introduces wavefront aberrations in addition to the intrinsic seeing by the atmospheric turbulence. One example of dome seeing effects is the Low Wind Effect (LWE) known in the High Contrast Imaging (HCI) community (Sauvage et al., 2015), which is due to radiatively supercooled telescope spiders. LWE causes a strong degradation of the instrument Point Spread Function (PSF) and thus the contrast. Dome turbulence often exhibits a deviation from the Kolmogorov/von Kármán power spectrum with more power at high spatial frequencies (Lai et al., 2020). High spatial frequencies are amplified by the AO system through aliasing. It is therefore important to identify the source of dome seeing and remove this effect either physically through modification of the telescope and/or dome, or by data processing. Still, the physical and statistical properties of dome seeing remain largely unknown and are open to further research.

Below we describe a method to simulate an evolving but non-translating turbulence which will try to match to the observed covariance peak profiles of CANARY Datasets. It should be noted that in this thesis, this model is used solely to compare the performance of ANNs trained on simulated frozen flow and fitted dome turbulence models and we draw no conclusions on the accuracy of this model in recreating the exact dome turbulence conditions encountered within CANARY. These are undoubtedly affected by the local telescope environment and not taken into account in the model here.

This approach uses the same framework for modelling boiling turbulence given in Glindemann et al. (1993), which was for explaining the short time scales (tens of milliseconds) of observed turbulence. The framework is based on the Fast Fourier

Transform (FFT) approach described in Section 2.3.2, while modelling the evolution of the turbulence as a Markov process. Similar to Eq. 2.44, at $t = 0$ the square root of a turbulence power spectrum (here the Kolmogorov spectrum is assumed) is filtered with a Gaussian white noise in the spatial frequency domain,

$$\mathbf{G}_0 = \mathbf{g}_0'\mathbf{h}, \tag{5.3}$$

where $\mathbf{g}'$ is a complex Hermitian matrix representing a Gaussian white noise in the frequency domain. $\mathbf{h}$ is defined as

$$\mathbf{h} = \frac{0.1513}{L}\, r_0^{-5/6}\mathbf{f}^{-11/6}, \tag{5.4}$$

where L is the physical size of the phase screen in metres, r_0 is the turbulence strength. $\mathbf{f}$ is a spatial wavenumber matrix with the values ranging from 0 to $\frac{\sqrt{2}N}{2L}$, where N is the number of pixels along one dimension of the phase screen. The real part of the inverse Fourier transform of $\mathbf{G}_0$ gives the initial phase screen.

When $t > 0$, $\mathbf{G}_t$ is updated as a weighted sum of its last state $\mathbf{G}_{t-1}$ and a newly generated filtered Kolmogorov spectrum $\mathbf{g}_t'\mathbf{h}$,

$$\mathbf{G}_t = \boldsymbol{\beta}\mathbf{G}_{t-1} + \sqrt{1 - |\boldsymbol{\beta}|^2}\mathbf{g}_t'\mathbf{h}. \tag{5.5}$$

In Glindemann et al. (1993), the weighting matrix $\boldsymbol{\beta}$ can be determined from the observed data. In this thesis, we calculate $\boldsymbol{\beta}$ from a scalar decay rate α [‡],

$$\boldsymbol{\beta} = [1 - (1 - \mathbf{f}_n^{-11/12})(1 - \alpha)]e^{-i\boldsymbol{f}\cdot\boldsymbol{v}}, \tag{5.6}$$

where $\mathbf{f}_n = L\mathbf{f}$, $\boldsymbol{v}$ is the wind vector, and $\cdot$ denotes the dot product. In both cases, the decay is assumed spatial frequency dependent. The real part of the inverse Fourier transform of $\boldsymbol{G}_t$ then yields the phase screen at time step t. When v is nonzero, this simulates a boiling and translating turbulence layer. When v is zero, this simulates an evolving but static layer. In the following simulations of the dome turbulence, we will set v to 0.

[‡] This is an empirical function fitted on on-sky data by Tim Butterley at Centre for Advanced Instrumentation (CfAI), Durham University. A rigorous mathematical theory of the dome seeing is still under study in the AO community.

When $\alpha = 1$, $\boldsymbol{\beta}$ will be a unit matrix and $\mathbf{G}_t = \mathbf{G}_0$, meaning a constant phase screen invariant of time. As α gets smaller, β is smaller for all f_n. From Eq. 5.6, $\mathbf{G}_t$ will be more influenced by the random $\mathbf{g}'_t$, indicating a more rapid degradation. Besides, in this model, higher spatial components degrade slightly faster (smaller β) (Conan et al., 1995). β corresponding to the piston of the turbulence is set to 0, indicating that this component is completely random and independent through time.

If the decay is assumed independent of the spatial frequency, as is found in Srinath et al. (2015) by fitting the open-loop temporal PSDs of the Gemini Planet Imager (GPI) telemetry, Eq. 5.5 reduces to

$$\mathbf{G}_t = \alpha \mathbf{G}_{t-1} + \sqrt{1-\alpha^2}\mathbf{g}'_t\mathbf{h}. \tag{5.7}$$

We call the dome model assuming frequency-dependent decay rates (Eqs. 5.5 and 5.6) dome *Model-A*, and the model assuming a constant decay rate (Eq. 5.7) *Model-B*.

5.5.3 Fitting the Dome Model

Degradation of the covariance peak of slopes generated assuming dome Model-A or Model-B follows a power law, as CANARY Datasets 0 and 1 in Fig. 5.3 exhibit. Such dome models have two parameters as described above, r_0 and α. r_0 determines how strong the peak is when $\Delta n = 0$ (noise peak excluded). α determines the gradient of peak degradation. By fitting the degradation profiles, we can find the set of dome model parameters that best describes CANARY Datasets.

Fig. 5.5 shows this fitting using Model-A (left) and Model-B (right) of CANARY Dataset 0. For the curve fitting, simulated slopes are generated using the wavefront sensing subsystem built by Soapy and relevant module parameters listed in Table 5.2, except that the frozen flow turbulence model is replaced with each dome model and wavefront sensing noise is turned off for the robustness of fitting. Generated phase screens are 512×512 pixels, with the central 64×64 pixels being the

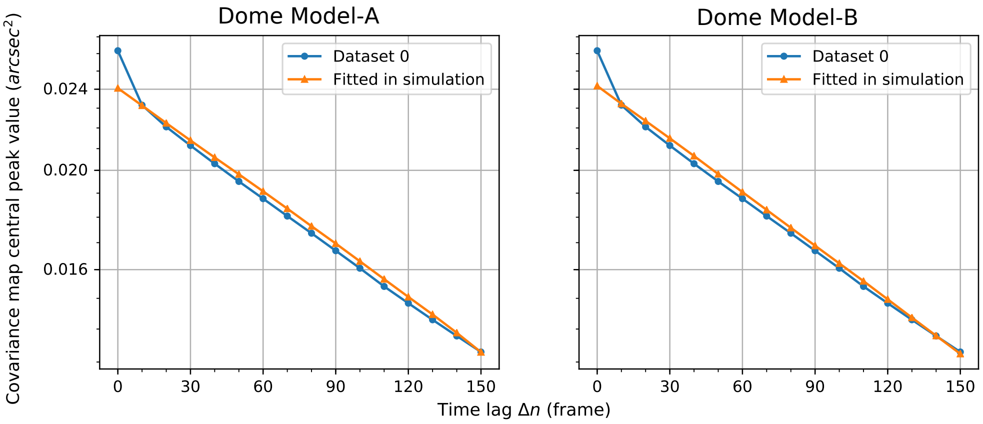


Figure 5.5: Covariance peak fitting of CANARY Dataset 0 in simulation using dome Model-A (left) and Model-B (right) with noise-free wavefront sensing. Model-A: the decay rate varies with spatial frequencies. Model-B: the decay rate is independent of spatial frequencies. The time lag corresponds to 0 to 1 s. The extra power shown in Dataset 0 at $\Delta n = 0$ is brought by WFS noise.

Table 5.7: Fitted parameters of CANARY Datasets. Model-1 and Model-2 were fitted on Dataset 0. Model-3 was fitted on Dataset 3.

Turbulence type	Parameter	Model-1	Model-2	Model-3
Dome	r_0 (m)	0.112	0.112	0.141
	α	0.9957	0.9962	0.97
Frozen flow	r_0 (m) v (m/s)	None	None	0.05 0.01

telescope pupil. Simulated slope sequences (assuming either Model-A or B) last for 10,000 frames as CANARY Datasets. Fitted parameters of both dome models are listed in Table 5.7 as *Model-1* and *Model-2* respectively. It should be noted that even with a very long slope sequence (100,000 frames), the peak curve did not converge and fluctuated slightly between different realisations. The fitted curve shown is a best fit, which gives the lowest RMS error between the fitted and the actual auto-correlations, evaluated at $\Delta n \geq 20$. Considering the amount of time taken for the peak fitting procedure we will not focus on quantifying the dome model fitting error here. However, we will later show that the ANN performance is tolerant to this error.

Decay trends shown in Datasets 2-5 are more curved than an exponential decay.

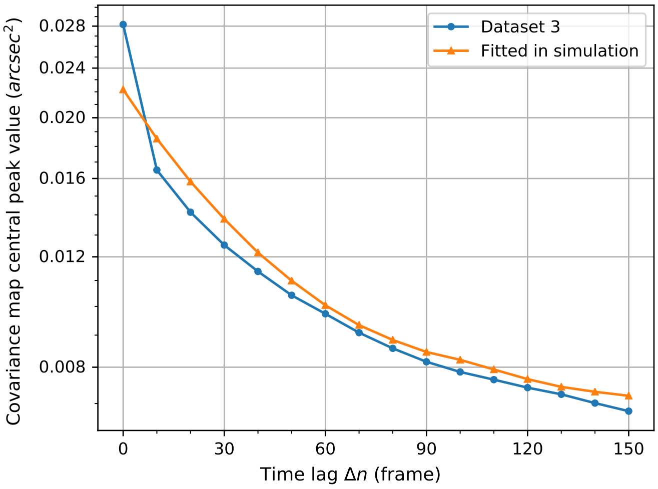


Figure 5.6: Covariance peak fitting of CANARY Dataset 3 in simulation assuming a dome layer (Model-A) and a slowly moving frozen flow with noise-free wavefront sensing.

To fit this trend, we consider two independent layers, one dome layer and one slowly moving frozen flow layer. Fig. 5.6 shows the fitting of Dataset 3 using this combination. For the dome layer Model-A is assumed. The fitted parameters are given in Table 5.7 as *Model-3*. The dome layer has an r_0 of 14.1 cm, accounting for 15% of the total turbulence strength. Its decay rate is 0.97. The frozen layer follows von Kármán statistics with an L_0 set to 25 m. The best fit wind speed was 0.01 m/s with an r_0 of 4.96 cm.

5.6 Training Models

In this section, we train ANN predictors using the turbulence models we fitted in the last section presented in Table 5.7. These turbulence models are:

1. Model-1, a single dome layer assuming dome Model-A fitted on Dataset 0, with parameters given in the third column of Table 5.7.
2. Model-2, a single dome layer assuming dome Model-B fitted on Dataset 0, with parameters given in the fourth column of Table 5.7.

3. Model-3, the sum of a dome layer and a slow frozen flow layer fitted on Dataset 3, with parameters given in the fifth column of Table 5.7.

From each turbulence model we generated 100,000 open-loop slope sequences. Each sequence is a 30-frame sequence consisting of 72-element slope vectors, $(\boldsymbol{s}_1, \boldsymbol{s}_2, \dots, \boldsymbol{s}_{30})$. $(\boldsymbol{s}_1, \boldsymbol{s}_2, \dots, \boldsymbol{s}_{28})$ are the ANN training inputs sequentially and $\boldsymbol{s}_{30}$ is the training target. The slope generation process is as described in Section 3.4.1, using the wavefront sensing subsystem built with Soapy. GS magnitude is 9.47. The calculated system throughput is 0.361, matching the photon count level of Dataset 3, as was set for training the frozen flow predictor in Section 5.4. See Table 5.2 for other related simulation parameters.

In Section 4.8 we have applied a non-spatially aware predictor to the frozen flow, which predicts each subaperture from its own history without the knowledge of the spatial distribution of WFS subapertures. Here we explore the potential of a non-spatially aware predictor with static turbulence. The first 10,000 sequences in each of the above three training datasets were reshaped into 360,000 sequences, each being a 30-frame slope sequence of 2-element vectors, the x and y slope of a single subaperture.

In total we have six training datasets, three for training spatially aware predictors and three for non-spatially aware predictors. Using each of the datasets we will train an ANN predictor. The training and hyperparameter tuning procedures follow that described in Section 5.4. The network is set to have two stacked LSTM cells and a final FC layer. In the hyperparameter tuning process, the searching range of neuron numbers of both LSTM cells is between 100 and 250 for spatially aware predictors, with a step of 5. For non-spatially aware predictors, the searching range is between 4 and 30, with a step of 2. The total trial number of the ANN topology for each predictor is 20.

Table 5.8 lists all six ANN predictors we obtained (namely *Predictor-I*, *-II*, *-III*, *-IV*, *-V* and *-VI*). Corresponding network structures are also given.

Table 5.8: Structures and training turbulence conditions of ANN dome predictors.

Predictor	I	II	III	IV	V	VI
Turbulence model assumed	Model-1	Model-1	Model-2	Model-2	Model-3	Model-3
Dome model used for training	A	A	B	B	A	A
An extra frozen layer for training	No	No	No	No	Yes	Yes
Input/output vector size	72	2	72	2	72	2
# of neurons in the first LSTM	185	22	155	24	230	18
# of neurons in the second LSTM	110	8	105	6	155	16

5.7 Prediction Performance with Dome Predictor

In this section we analyse the prediction performance using the approaches detailed in Section 4.2.

5.7.1 Performance of Different Turbulence Models

Prediction errors σ_{pred} (defined in Section 5.4) of each predictor alongside corresponding 2-frame and 1-frame delay errors are detailed in Table 5.9. We have observed the following:

1. All prediction errors by dome predictors are lower than corresponding 2-frame delay errors, showing a significant improvement in prediction performance brought by using a turbulence model that better describes the observed data than a pure frozen flow model.

2. For Datasets 2 to 5, prediction errors are even lower than corresponding 1-frame delay errors.

3. The use of a dome turbulence instead of a frozen flow model provides the best predictor performance, however the differences between different models (turbulence Model-1 to 3) are negligible. This indicates that the training is

Table 5.9: Prediction error (in nm RMS) of all six ANN predictors-I to -VI with CANARY Datasets compared with 1-frame and 2-frame delay errors. Performance of the frozen flow predictor from Section 5.4 is also listed here for comparison.

Dataset No.	0	1	2	3	4	5
2-frame delay	222	193	316	390	494	402
1-frame delay	199	179	305	380	484	391
Frozen flow predictor	274	224	349	430	508	411
Predictor-I	209	180	288	352	440	362
Predictor-II	203	173	283	343	433	355
Predictor-III	209	180	289	353	442	363
Predictor-IV	204	174	286	348	439	359
Predictor-V	215	186	298	363	457	374
Predictor-VI	**201**	**171**	**281**	**341**	**432**	**353**

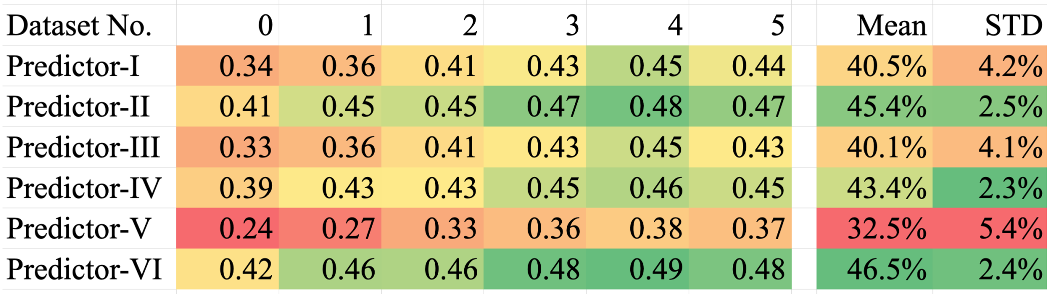
relatively robust across a range of different observing conditions, and should be tolerant to the dome turbulence model fitting error as well.

4. Non-spatially aware predictors (-II, -IV and -VI) perform better than their spatially aware counterparts (-I, -III and -V) regardless of turbulence model assumed or Dataset considered.

Table 5.10 displays the reduction in the 2-frame delay error deduced from Table 5.9. This is defined as $\sqrt{\sigma_{\text{delay}}^2 - \sigma_{\text{pred}}^2}/\sigma_{\text{delay}}$, where σ_{delay} and σ_{pred} are the RMS delay and prediction errors described in Eq. 5.2. This value represents the reduction of the combination of temporal, noise and aliasing errors by dome predictors. The non-spatially aware Predictor-VI has a largest average improvement of $46.5 \pm 2.4\%$. Although the ANN training data were fitted on either Dataset 0 or Dataset 3, all predictors have the best performance with Dataset 4, then Datasets 5 and 3. This could be an indication that training is not sensitive to turbulence strength, which is consistent with our findings in Section 4.3.

We also notice that the ANN cannot suppress vibrations detected in Fig. 5.2. See Fig. 5.7 for the temporal PSDs of the predicted slopes by the frozen flow predictor presented in Section 5.4 and by the dome Predictor-I for an example. Both predictors were trained with spatial awareness at the same noise level. All

Table 5.10: Reduction in the RMS 2-frame delay error by dome predictors



Dataset No.	0	1	2	3	4	5	Mean	STD
Predictor-I	0.34	0.36	0.41	0.43	0.45	0.44	40.5%	4.2%
Predictor-II	0.41	0.45	0.45	0.47	0.48	0.47	45.4%	2.5%
Predictor-III	0.33	0.36	0.41	0.43	0.45	0.43	40.1%	4.1%
Predictor-IV	0.39	0.43	0.43	0.45	0.46	0.45	43.4%	2.3%
Predictor-V	0.24	0.27	0.33	0.36	0.38	0.37	32.5%	5.4%
Predictor-VI	0.42	0.46	0.46	0.48	0.49	0.48	46.5%	2.4%

strong spikes indicative of vibrations were maintained in the predicted slopes PSD. Note from Section 4.2.2.1 that the ANN frequency transfer is highly nonlinear, which complicates the analysis with a single-frequency component. However, the reproduction of the power in that component is still undesired and can potentially degrade the system performance.

Fig. 5.8 displays the transfer of temporal PSDs of Tilt and Trefoil modes by the frozen flow predictor and dome Predictor-I, which is the PSD of the predicted slopes divided by that of the non-predicted slopes. The transfer of the first 36 Zernike modes can be found in Fig. A.5, where we found all modes were treated roughly the same by either predictor, unlike the dependence on the direction of the mode with respect to the wind which was present when frozen flow was assumed (see Section 4.2.2). Here a dome predictor can be interpreted as a low-pass filter in the temporal domain, with a response approximating unity with low frequencies implying its prediction power with these signals. The difference between the frozen flow predictor and the dome predictor in high frequencies beyond around 10 Hz suggests that the later might be attenuating noise or aliasing.

5.7.2 Performance with Varying WFS Noise Level

As is shown in Section 4.2, the WFS noise level impacts the ANN training and test significantly. In this section, we explore this with CANARY data. We compare the performance of spatially aware and non-spatially aware predictors (trained with

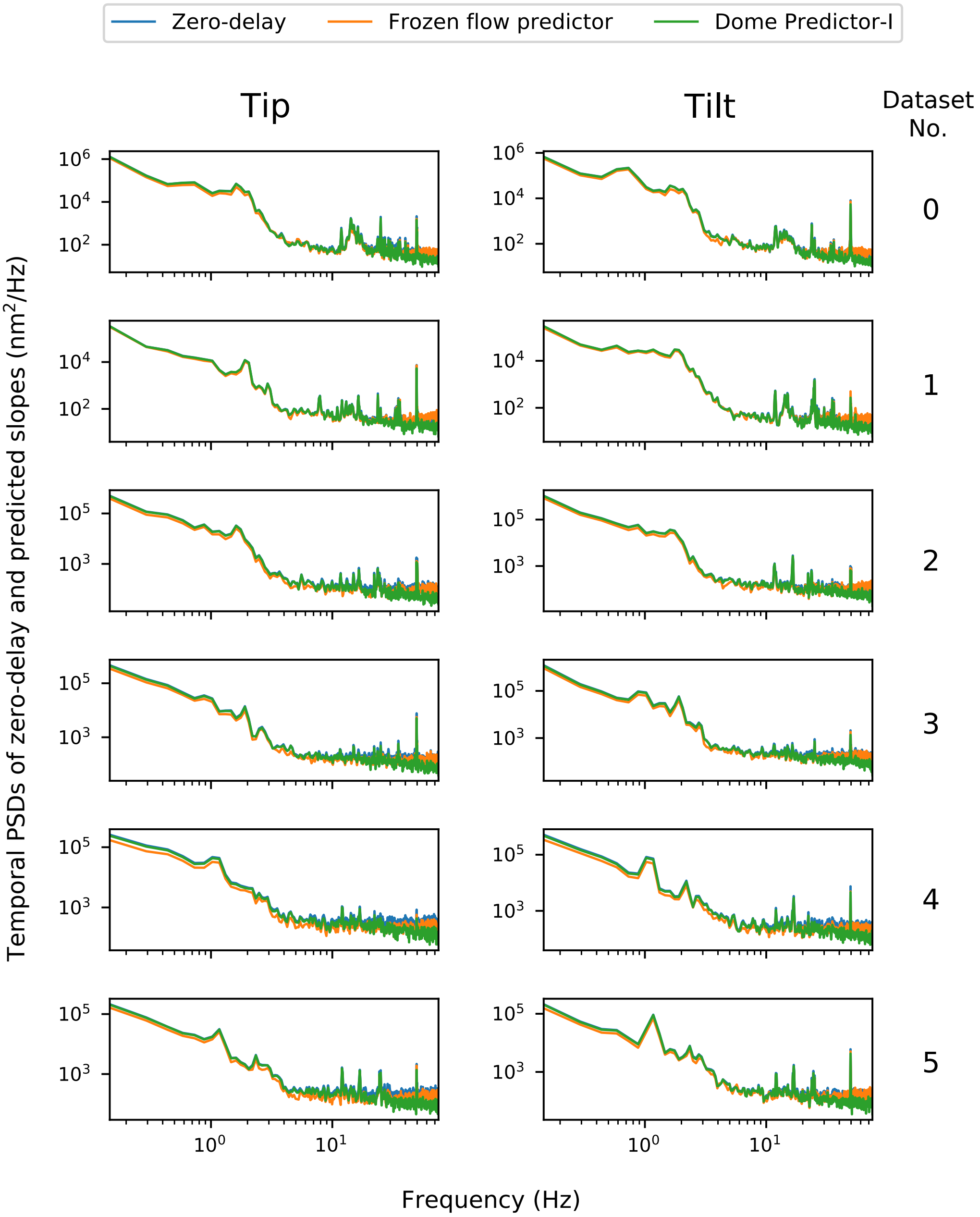


Figure 5.7: Temporal PSDs of CANARY Datasets 0-5 and of the corresponding predictions made by the frozen flow predictor and the dome Predictor-I.

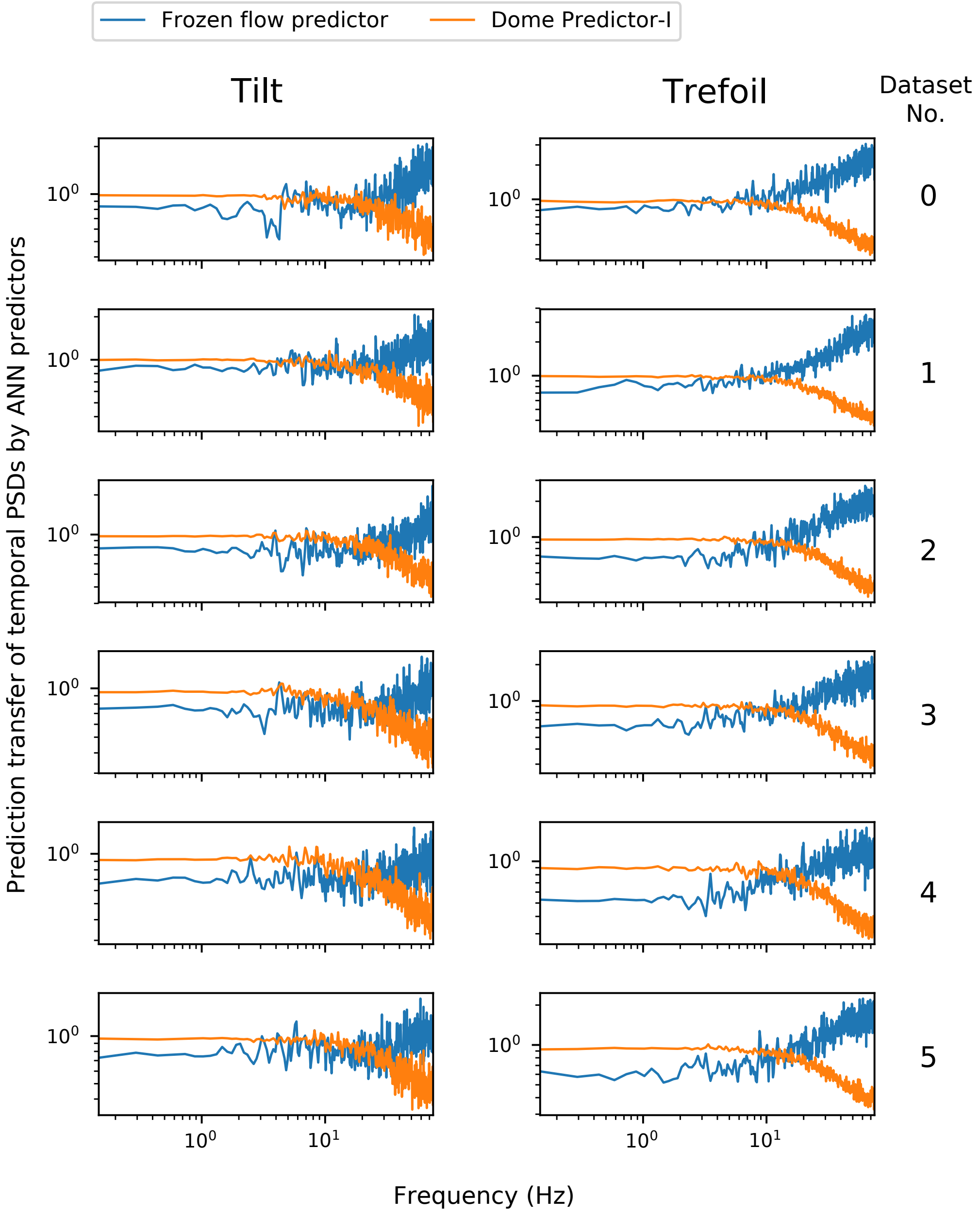


Figure 5.8: Prediction transfer of temporal PSDs of CANARY Datasets 0-5 in Tilt and Trefoil by the frozen flow predictor and dome Predictor-I.

Table 5.11: Structures of ANN dome predictors trained with different noise levels. From left to right, the predictors were trained with increasing noise variance.

Predictor	II	II(2)	II(3)
Turbulence model used for training	Model-1	Model-1	Model-1
Input/output vector size	2	2	2
# of neurons in the first LSTM	22	14	26
# of neurons in the second LSTM	8	10	12

Table 5.12: Measured noise variance ($\times 10^3$) (arcsec^2) of CANARY Datasets 0-5 and of training data for Predictors-II, -II(2) and -II(3).

Predictor-II	Predictor-II(2)	Predictor-II(3)	Dataset 0	1	2	3	4	5
0.3	1.6	4.2	2.5	2.4	8.3	12.0	20.1	12.8

the same noise level) under different noise conditions, and the performance of non-spatially aware predictors trained with different noise levels.

For the latter, we generated another two training datasets with more noise. The first dataset was acquired by using all WFS pixels for centroiding (instead of the 12 brightest ones). The second dataset was obtained by using all the pixels for centroiding and reducing the GS magnitude from 9.47 to 10.2. For both datasets, turbulence Model-1 was assumed.

Using these two training datasets, we trained another two non-spatially aware dome predictors, *Predictor-II(2)* and *Predictor-II(3)* respectively. Structures of both predictors alongside Predictor-II can be found in Table 5.11. To be explicit, the only difference in training conditions among these three predictors is the noise level. Measured noise variances of the training data for Predictor-II, -II(2) and -II(3) alongside those of CANARY Datasets copied from Table 5.3 are given in Table 5.12.

To test the predictors in different noise conditions, additional Gaussian noise was added to Dataset 0. Standard deviation of this noise ranges from 0.1 to 1.5 pixels, with a step of 0.2 pixel. The resultant delay and prediction errors are plotted in

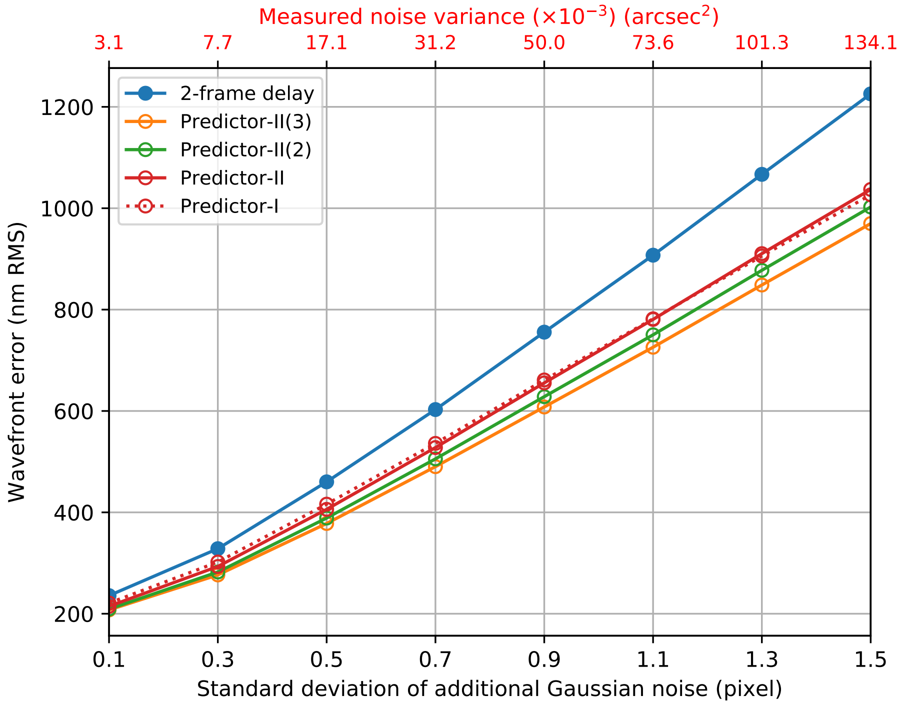


Figure 5.9: Prediction error (in nm RMS) of Predictors-II, -II(2) and -II(3) along with 2-frame delay error as levels of Gaussian noise added to CANARY Dataset 0 increases. The corresponding measured noise variance is also labelled for reference.

Fig. 5.9. The corresponding measured noise variance is also labelled. Compared with Table 5.12, the measured noise variance of the noisiest CANARY Dataset is equivalent to an additional noise of approximately 0.5 pixel RMS to Dataset 0. The smaller gradients shown with the dome predictor curves compared to the 2-frame delay curve suggest that such predictors are resistant to WFS noise.

Fig. 5.10 plots $\sigma_{\text{pred},z}/\sigma_{\text{delay},z}$ (defined in Section 4.2.1), the ratio between the Zernike breakdown of the RMS prediction error and the 2-frame delay error. Recall that Predictors-II, -II(2) and -II(3) are non-spatially aware predictors trained with increasing WFS noise. Predictors-I and II were trained with the same amount of noise, the former with spatial awareness. There are several observations we can make from Fig. 5.10:

1. The spatially aware predictor exhibits an improvement in performance (i.e. a lower relative prediction error) for higher-order Zernike terms. This behaviour

is not observed for any of the three non-spatially aware predictors.

2. The performance of the non-spatially aware predictors is dependent on the level of training noise, with better performance achieved by predictors trained in noisier regimes. This is in line with earlier results (see Section 4.2), but here we see that the correction is broadly uniform across all Zernike modes of higher-order than Tip and Tilt (TT).
3. At the highest noise level (1.5 pixel standard deviation of additional slope noise added), the spatially aware and non-spatially aware predictors perform the same for Zernike terms above Z = 28.
4. The prediction accuracy of Tip and Tilt for non-spatially aware predictors is dependent on the additional noise level.

Fig. 5.11 shows the prediction transfer of temporal PSDs by non-spatially aware dome predictors trained with different noise levels. The higher the noise present during training, the higher the transfer is at high temporal frequency. This implies that increasing the training noise improves the noise suppression and the resultant prediction performance. Fig. 5.12 compares the transfer of temporal PSDs by the spatially aware dome Predictor-I and its non-spatially aware counterpart Predictor-II. Comparing this with Fig. 5.10, we can see that the transfer of a non-spatially aware predictor is uniform across different Zernike modes, while the spatially aware predictor shows stronger rejection of high frequencies in higher-order Zernike modes. This implies stronger filtering of WFS noise in higher-order modes, which contributes to the lower error of the spatially aware predictor with these modes compared with the non-spatially aware predictor. Still, the non-spatially aware predictor predicts components below around 10 Hz with more accuracy regardless of the noise level. These trends shown in Fig. 5.12 are consistent with Fig. 4.21.

Fig. 5.13 displays the auto-correlation of predicted and non-predicted slopes when the additional noise is 0.1 and 0.5 pixel RMS. The auto-correlation values at

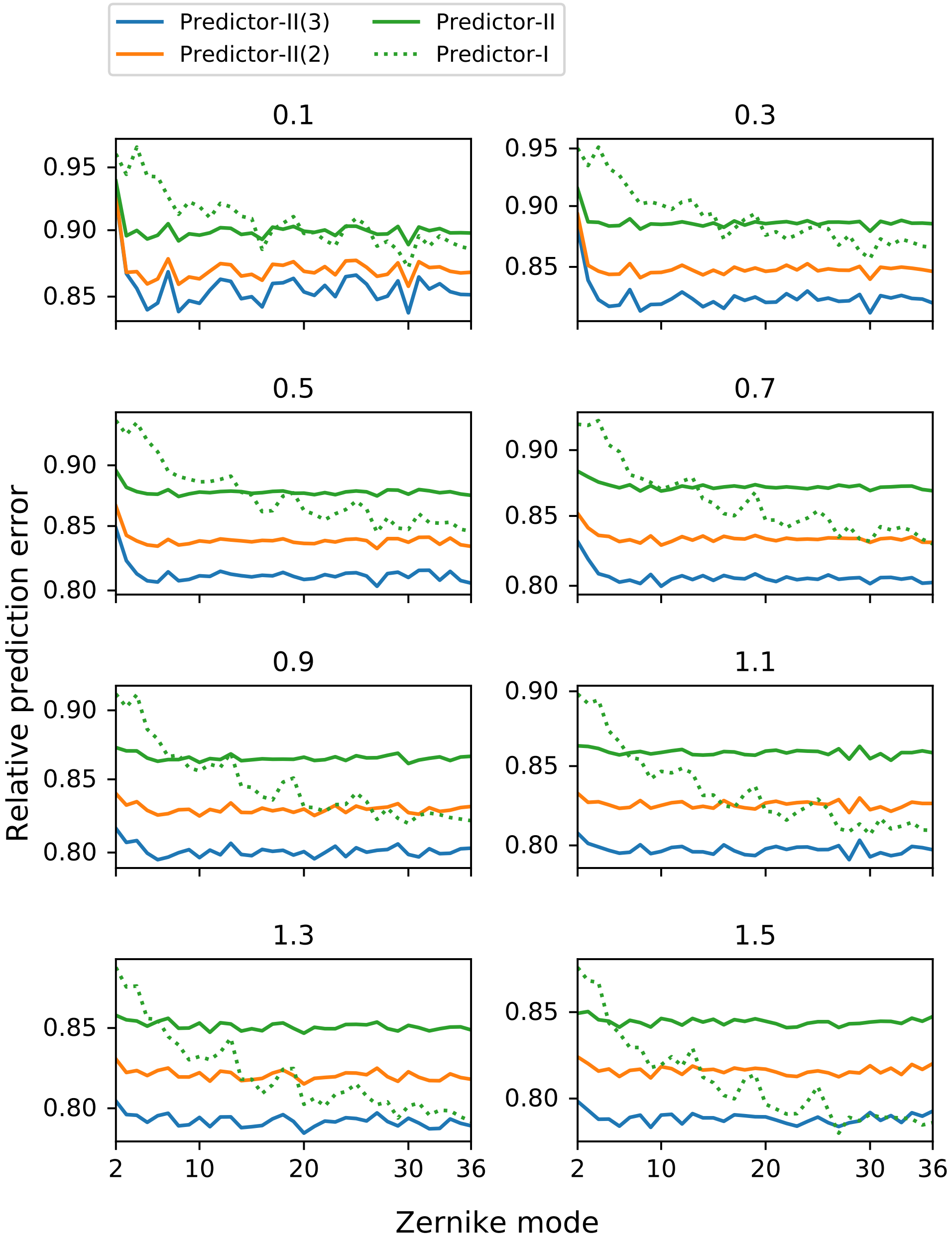


Figure 5.10: Ratio between the Zernike breakdown of the error in the predicted slopes and in the 2-frame delay slopes compared with the zero-delay slopes as the level of additional noise increases. Numbers above each figure denote the standard deviation (# WFS CCD pixel) of additional Gaussian noise added to Dataset 0.

$\Delta n = 0$ are further away from the fitted parabolas compared with Fig. 4.10, due to the much more noise present. From the auto-correlation fit we obtain the measured noise variance of the predicted and non-predicted slopes (shown in Fig. 5.14) and the noise propagation between adjacent frames (shown in Fig. 5.15), where this propagation is taken as the difference between the measured auto-correlation at $\Delta n = 1$ and the value of the fitted parabola at the same Δn (with the parabola fitted using the values measured at $\Delta n = 2$ and $\Delta n = 3$). A spatially aware predictor (Predictor-I) has lower noise variance and less propagated noise than a non-spatially aware one (Predictor-II) trained with the same amount of noise.

These results show that:

- An ANN trained with more noise shows better prediction performance irrespective of the spatial frequency (Zernike mode), which implies it is better at rejecting noise. This is consistent with the conclusion in Section 4.2.1.
- A spatially aware predictor is better at suppressing noise. However, the non-spatially aware predictor predicts slowly-evolving components (below around 10 Hz) with more accuracy. This is consistent with the conclusion in Section 4.8.
- For the Datasets dominated by a static layer studied in this chapter, a non-spatially aware predictor always shows better performance than the spatially aware predictor trained in the same noise condition.

5.8 Conclusions

We have provided the first evidence that the ANN can improve an on-sky AO system performance and that the methodology for training an ANN predictor taken in simulations in Chapter 3 can transfer to real systems. We have trained ANNs with simulated data to predict six 10,000-frame slope datasets taken in open loop

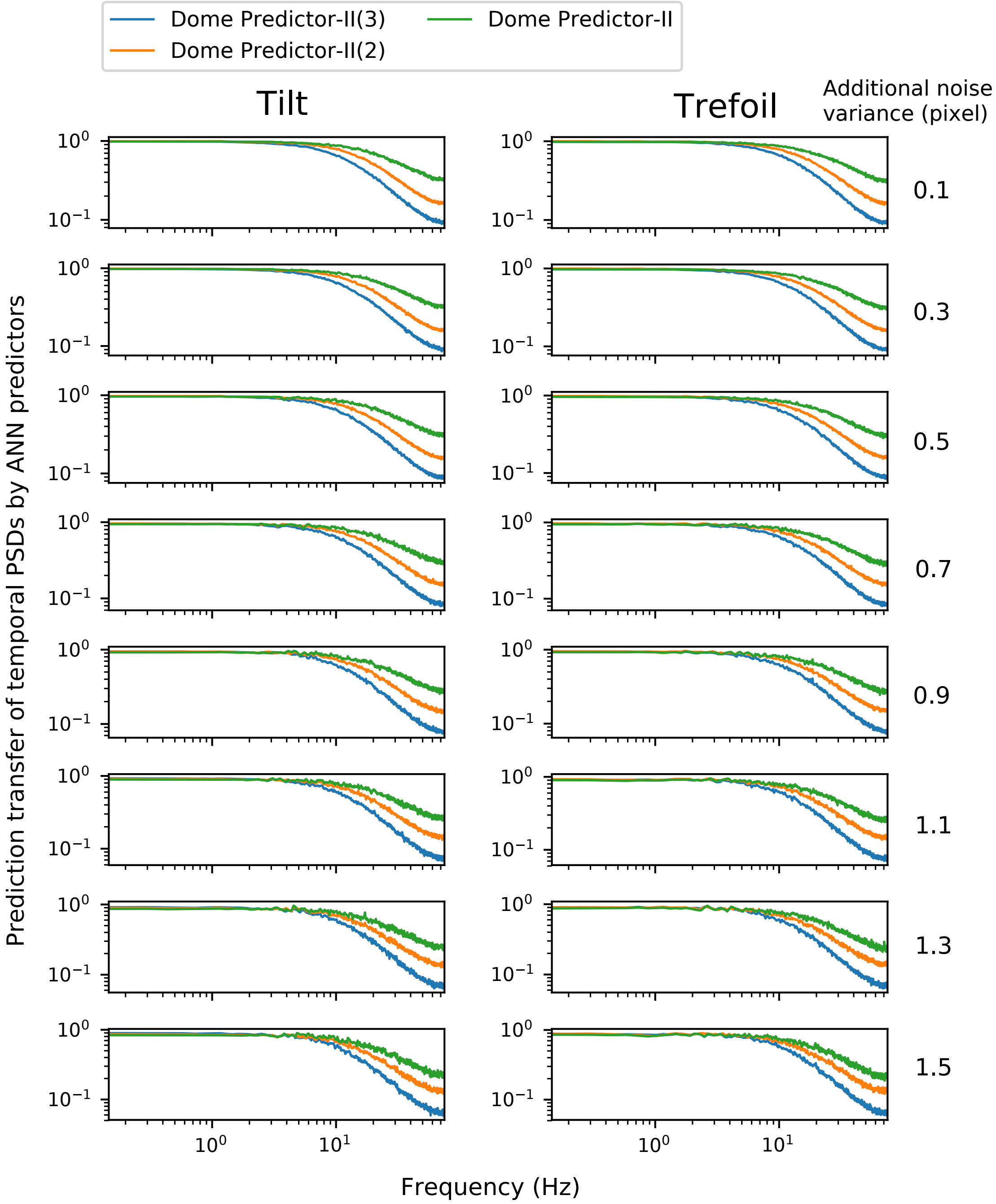


Figure 5.11: Prediction transfer of temporal PSDs of Tilt and Trefoil by dome Predictors-II(3), -II(2) and -II that were trained with decreasing noise.

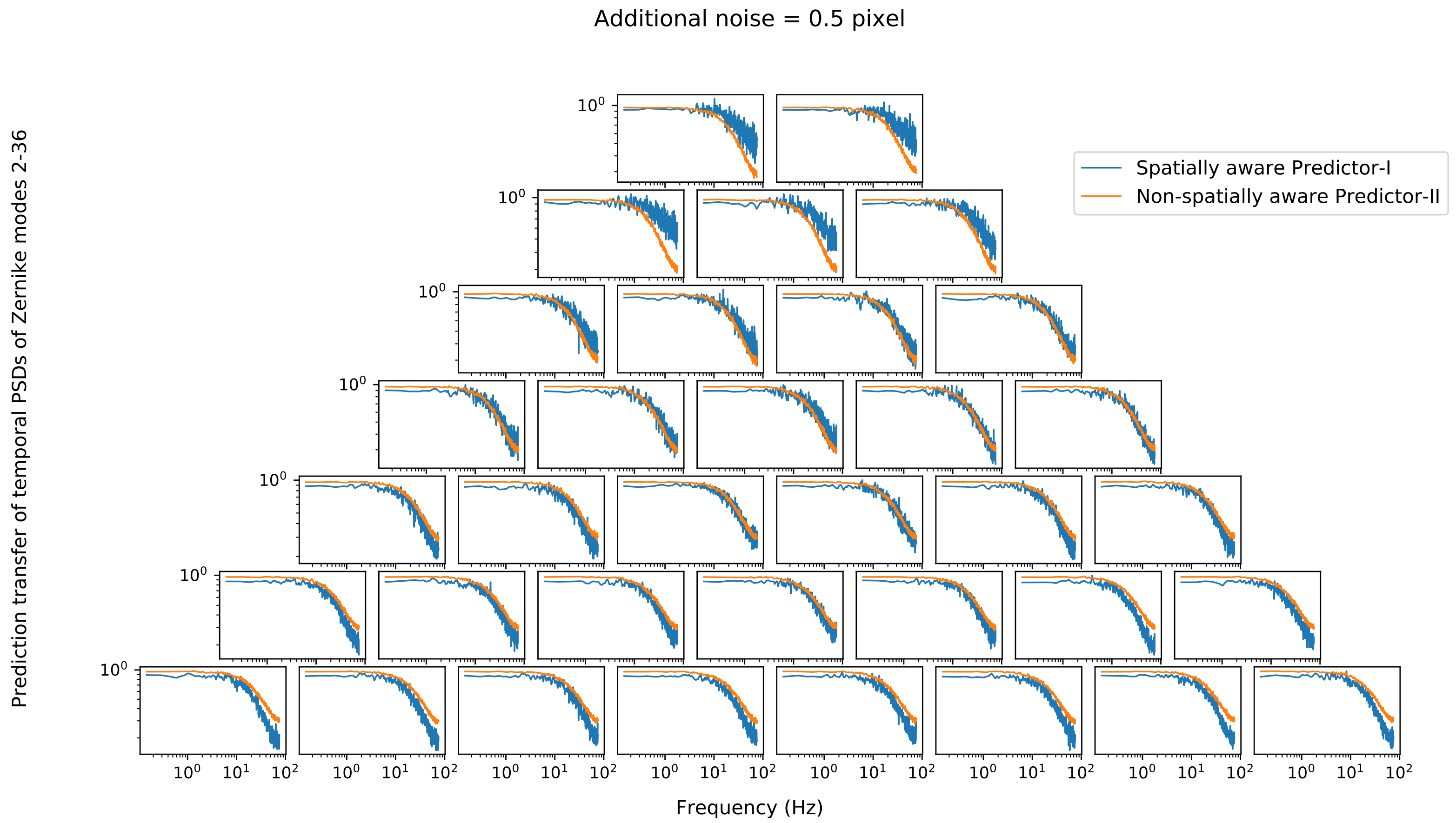


Figure 5.12: Prediction transfer of temporal PSDs in Zernike modes 2-36 by the spatially aware dome Predictor-I and non-spatially aware dome Predictor-II when 0.5 pixel RMS of Gaussian noise is added to Dataset 0.

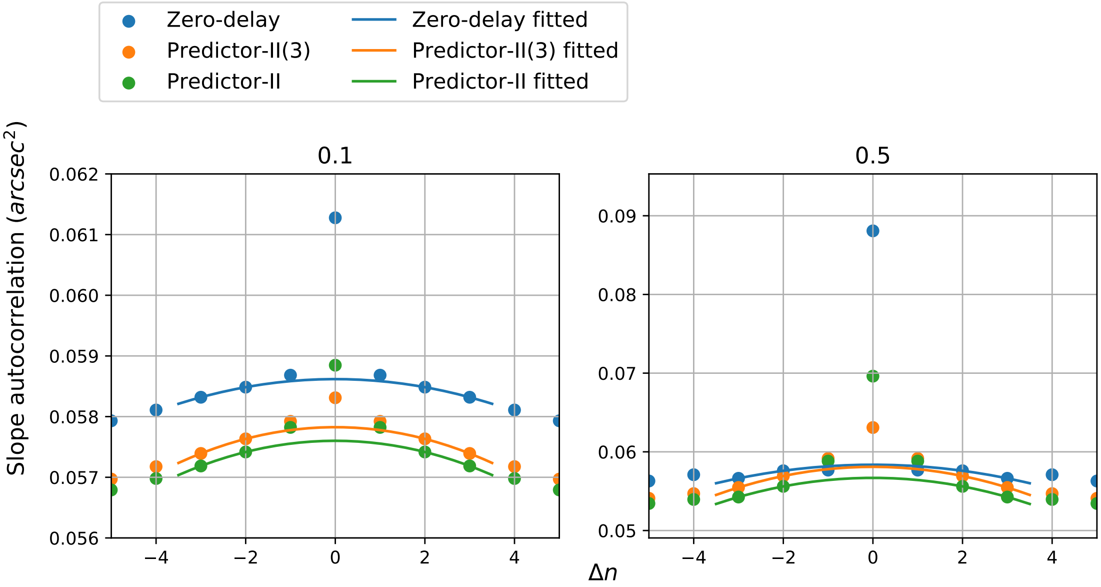


Figure 5.13: Auto-correlation of predicted and non-predicted slopes with an additional Gaussian noise of 0.1 (left) and 0.5 (right) pixel RMS added to CANARY Dataset 0 respectively. The parabolas were fitted using the values at $\Delta n = 2$ and 3. Deviation at $\Delta n = 0$ indicates the noise present within the slopes (shown in Fig. 5.14). Deviation from this parabola at $\Delta n = 1$ indicates the propagated noise between adjacent frames (shown in Fig. 5.15).

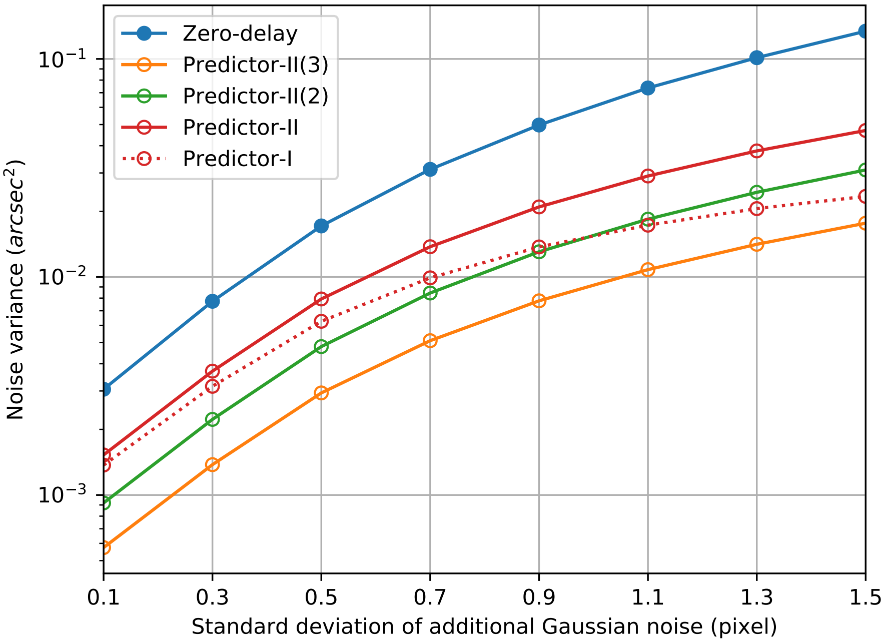


Figure 5.14: Measured noise-induced slope variance (arcsec^2) of the predicted and non-predicted slopes.

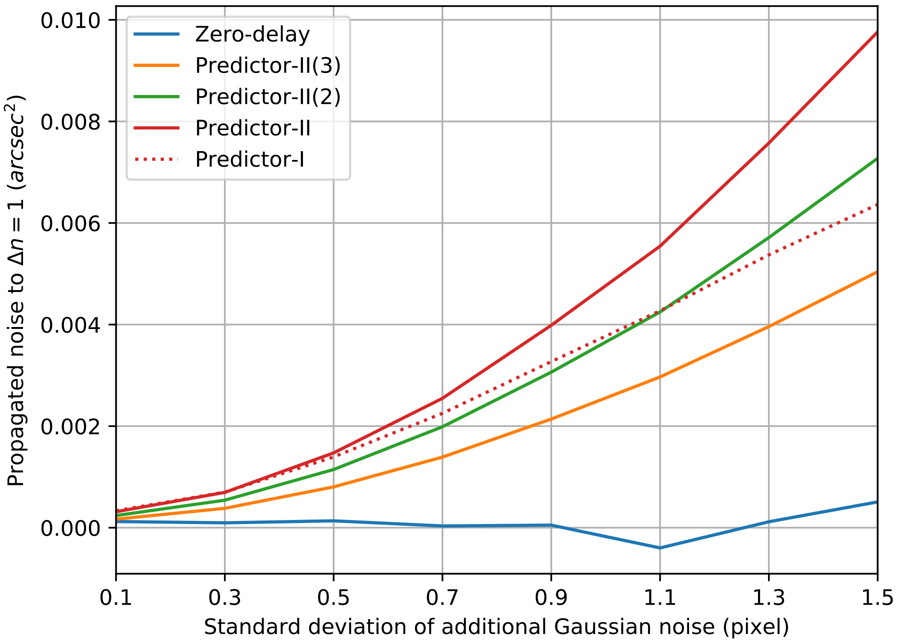


Figure 5.15: Noise propagation (arcsec^2) between adjacent frames by ANN predictors. Noise propagation in the non-predicted slopes is negligible, represented by the nearly flat curve approaching 0.

at 150 Hz by the 7×7 SHWFS of the CANARY instrument between 28 September and 2 October, 2017. The largest reduction in the combination of the temporal, noise and aliasing errors filtered by the ANN compared with those associated to the 2-frame latency is 48%.

In addition to the training noise levels, we have identified two factors that significantly impact the ANN performance: vibrations and temporal dynamics of the observed turbulence. We have observed vibrations in TT modes, and that the observed data was dominated by a strong static layer indicative of dome seeing, instead of frozen flow assumed in simulations. We have used an empirical static turbulence model for training ANNs. The improved performance over a frozen flow predictor suggests that this model provides a better representation of the observed data, however the ANN performance is tolerant of the error in fitting this model. We have shown that the ANN cannot predict the vibrations.

For the CANARY Datasets dominated by a static layer, the non-spatially aware predictor shows better performance than the spatially aware predictor trained with the same amount of noise in all conditions. The non-spatially aware predictor predicts low-temporal frequencies (below around 10 Hz) with improved accuracy.

An ANN predictor should be trained offline in simulation due to the amount of training data or the variability (e.g. wind speed, decay rate) within for the robustness and adaptability. However, the training data should be representative of on-sky observations by matching the noise condition and the dynamics of the turbulence such as the existence of vibration and/or frozen flow. System and turbulence characterisation is thus necessary before the ANN training.

5.9 References

F. Assémat, E. Gendron, and F. Hammer. The FALCON concept: multi-object adaptive optics and atmospheric tomography for integral field spectroscopy – principles and performance on an 8-m telescope. *Monthly Notices of the Royal Astronomical Society*, 376(1):287–312, 2007.

R. Avila, J. Vernin, M. R. Chun, and L. J. Sanchez. Turbulence and wind profiling with generalized scidar at Cerro Pachon. In *Adaptive Optical Systems Technology*, volume 4007, pages 721–732. SPIE, 2000.

L. Bardou, E. Gendron, G. Rousset, D. Gratadour, A. G. Basden, D. B. Calia, T. Buey, M. Centrone, F. Chemla, J.-L. Gach, D. Geng, Z. Hubert, D. J. Laidlaw, T. J. Morris, R. M. Myers, J. Osborn, A. P. Reeves, M. J. Townson, and F. Vidal. Error breakdown of ELT-elongated LGS wavefront-sensing using CANARY on-sky measurements. In *Adaptive Optics Systems VI*, volume 10703, pages 614–632. SPIE, 2018.

A. Basden, S. Wells, and R. Myers. Simulation and laboratory demonstration of measurement and mitigation of dome seeing. *Journal of Physics: Conference Series*, 595:012002, 2015.

A. G. Basden and R. Myers. The durham adaptive optics real-time controller: capability and extremely large telescope suitability. *Monthly Notices of the Royal Astronomical Society*, 424:1483–1494, 2012.

J. Bergstra and Y. Bengio. Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13:281–305, 2012.

T. Butterley, R. W. Wilson, and M. Sarazin. Determination of the profile of atmospheric optical turbulence strength from SLODAR data. *Monthly Notices of the Royal Astronomical Society*, 369(2):835–845, 2006.

J.-M. Conan, G. Rousset, and P.-Y. Madec. Wave-front temporal spectra in high-resolution imaging through turbulence. *J. Opt. Soc. Am. A*, 12(7):1559–1570, 1995.

E. Gendron, F. Vidal, M. Brangier, T. Morris, Z. Hubert, A. Basden, G. Rousset, R. Myers, F. Chemla, A. Longmore, T. Butterley, N. Dipper, C. Dunlop, D. Geng, D. Gratadour, D. Henry, P. Laporte, N. Looker, D. Perret, A. Sevin, G. Talbot, and E. Younger. Moao first on-sky demonstration with canary. *Astronomy and Astrophysics*, 529(L2), 2011.

A. Glindemann, R. G. Lane, and J. C. Dainty. Simulation of time-evolving speckle patterns using kolmogorov statistics. *Journal of Modern Optics*, 40(12):2381–2388, 1993.

A. Guesalaga, B. Neichel, A. Cortés, C. Béchet, and D. Guzmán. Using the Cn2 and wind profiler method with wide-field laser-guide-stars adaptive optics to quantify the frozen-flow decay. *Monthly Notices of the Royal Astronomical Society*, 440 (3):1925–1933, 2014.

C. Kulcsár, G. Sivo, H.-F. Raynaud, B. Neichel, F. Rigaut, J. Christou, A. Guesalaga, C. Correia, J.-P. Véran, E. Gendron, F. Vidal, G. Rousset, T. Morris, S. Esposito, F. Quiros-Pacheco, G. Agapito, E. Fedrigo, L. Pettazzi, R. Clare, R. Muradore, O. Guyon, F. Martinache, S. Meimon, and J.-M. Conan. Vibrations in AO control: a short analysis of on-sky data around the world. In *Adaptive Optics Systems III*, volume 8447, pages 529–542. International Society for Optics and Photonics, SPIE, 2012.

O. Lai, J. K. Withington, M. R. Chun, and R. Laugier. Investigating ground layer and dome turbulence in astronomical observatories using a localized optical turbulence sensor. In *Imaging and Applied Optics Congress*, page JW1G.2. Optical Society of America, 2020.

D. J. Laidlaw, J. Osborn, T. J. Morris, A. G. Basden, E. Gendron, G. Rousset, M. J. Townson, and R. W. Wilson. Automated wind velocity profiling from adaptive optics telemetry. *Monthly Notices of the Royal Astronomical Society*, 491(1):1287–1294, 2019.

O. A. Martin, C. M. Correia, E. Gendron, G. Rousset, F. Vidal, T. J. Morris, A. G. Basden, R. M. Myers, Y. H. Ono, B. Neichel, and T. Fusco. William Herschel Telescope site characterization using the MOAO pathfinder CANARY on-sky data. In *Adaptive Optics Systems V*, volume 9909, pages 1143–1157. SPIE, 2016.

T. Morris, Z. Hubert, R. Myers, E. Gendron, A. Longmore, G. Rousset, G. Talbot, T. Fusco, N. Dipper, F. Vidal, D. Henry, D. Gratadour, T. Butterley, F. Chemla, D. Guzman, P. Laporte, E. Younger, A. Kellerer, M. Harrison, M. Marteaud, D. Geng, A. Basden, A. Guesalaga, C. Dunlop, S. Todd, C. Robert, K. Dee, C. Dickson, N. Vedrenne, A. Greenaway, B. Stobie, H. Dalgarno, and J. Skvarc. Canary: The ngs/lgs moao demonstrator for eagle. In *Proceedings of the First AO4ELT Conference*. EDP Sciences, 2010.

T. Morris, E. Gendron, A. Basden, O. Martin, J. Osborn, D. Henry, Z. Hubert, G. Sivo, D. Gratadour, F. Chemla, A. Sevin, M. Cohen, E. Younger, F. Vidal, R. Wilson, T. Butterley, U. Bitenc, A. Reeves, N. Bharmal, H.-F. Raynaud, C. Kulcsar, J.-M. Conan, D. Guzman, J. de Cos Juez, J.-M. Huet, D. Perret,

C. Dickson, D. Atkinson, T. Baillie, A. Longmore, S. Todd, G. Talbot, S. Morris, R. Myers, and G. Rousset. Multiple object adaptive optics: Mixed ngs/lgs tomography. In *Proceedings of the Third AO4ELT Conference*, volume 1, 2013.

R. M. Myers, Z. Hubert, T. J. Morris, E. Gendron, N. A. Dipper, A. Kellerer, S. J. Goodsell, G. Rousset, E. Younger, M. Marteaud, A. G. Basden, F. Chemla, C. D. Guzman, T. Fusco, D. Geng, B. L. Roux, M. A. Harrison, A. J. Longmore, L. K. Young, F. Vidal, and A. H. Greenaway. CANARY: the on-sky NGS/LGS MOAO demonstrator for EAGLE. In *Adaptive Optics Systems*, volume 7015, pages 52–60. International Society for Optics and Photonics, SPIE, 2008.

T. O'Malley, E. Bursztein, J. Long, F. Chollet, H. Jin, L. Invernizzi, et al. Keras Tuner. `https://github.com/keras-team/keras-tuner`, 2019.

J. Osborn, F. J. D. C. Juez, D. Guzman, T. Butterley, R. Myers, A. Guesalaga, and J. Laine. Using artificial neural networks for open-loop tomography. *Opt. Express*, 20(3):2420–2434, 2012.

J.-F. Sauvage, T. Fusco, A. Guesalaga, P. Wizinowitch, J. O'Neal, M. N'Diaye, A. Vigan, J. Girard, G. Lesur, D. Mouillet, J.-L. Beuzit, M. Kasper, M. Le Louarn, J. Milli, K. Dohlen, B. Neichel, P. Bourget, P. Heigenauer, and D. Mawet. Low wind effect, the main limitation of the sphere instrument. In *Adaptive Optics for Extremely Large Telescopes 4 – Conference Proceedings*, 2015.

H. W. Shepherd, J. Osborn, R. W. Wilson, T. Butterley, R. Avila, V. S. Dhillon, and T. J. Morris. Stereo-SCIDAR: optical turbulence profiling with high sensitivity using a modified SCIDAR instrument. *Monthly Notices of the Royal Astronomical Society*, 437(4):3568–3577, 2013.

G. Sivo, C. Kulcsár, J.-M. Conan, H.-F. Raynaud, Éric Gendron, A. Basden, F. Vidal, T. Morris, S. Meimon, C. Petit, D. Gratadour, O. Martin, Z. Hubert, A. Sevin, D. Perret, F. Chemla, G. Rousset, N. Dipper, G. Talbot, E. Younger, R. Myers, D. Henry, S. Todd, D. Atkinson, C. Dickson, and A. Longmore. First on-sky scao validation of full lqg control with vibration mitigation on the canary pathfinder. *Opt. Express*, 22(19):23565–23591, 2014.

S. Srinath, L. A. Poyneer, A. R. Rudy, and S. M. Ammons. Computationally efficient autoregressive method for generating phase screens with frozen flow and turbulence in optical simulations. *Opt. Express*, 23(26):33335–33349, 2015.

F. Vidal, E. Gendron, and G. Rousset. Tomography approach for multi-object adaptive optics. *J. Opt. Soc. Am. A*, 27(11):A253–A264, 2010.

L. Wang, M. Schöck, and G. Chanan. Atmospheric turbulence profiling with slodar using multiple adaptive optics wavefront sensors. *Appl. Opt.*, 47(11):1880–1892, 2008.

R. W. Wilson. SLODAR: measuring optical turbulence altitude with a Shack–Hartmann wavefront sensor. *Monthly Notices of the Royal Astronomical Society*, 337(1):103–108, 2002.

Chapter 6

Conclusions

6.1 Thesis Aim

This thesis has been dedicated to exploring the potential of Artificial Neural Network (ANN) as a nonlinear tool for open-loop wavefront prediction, in order to compensate for the inevitable temporal error in Adaptive Optics (AO) systems. We have successfully demonstrated the robustness and effectiveness of ANN-based predictors both with simulated and on-sky CANARY data, recorded by the 7×7 Shack-Hartmann Wavefront Sensor (SHWFS) running at 150 Hz.

In this chapter we present conclusions drawn from this study by answering the questions posed in Chapter 1, and discuss paths forward.

6.2 Characteristics of an ANN Wavefront Predictor

We provided the workflow for training and optimising an ANN predictor with simulated training data. The predictor receives a time sequence of past slope measurements by a SHWFS and predicts directly the future measurement either one or two frames in advance. The predictor is composed mainly of stacked Long Short-Term Memory (LSTM) cells. This workflow was first developed for simulations, but later translated well to real systems. Predicting only a single frame in advance provides a more accurate prediction, but a two frame prediction still shows a large performance benefit compared to a two-frame temporal delay and more accurately represents the latency encountered within a real AO system.

We have demonstrated that the ANN is capable of wavefront prediction both with simulated and real data. In simulations, the system performance in terms of residual Root Mean Squared (RMS) Wavefront Error (WFE) improved significantly after the predictor is incorporated, irrespective of guide star magnitude or turbulence strength. The prediction error is within 19.6 to 39.3 nm RMS of a latency-free system operating under the same conditions compared to a temporal error of 77.7 ± 4.5 nm RMS. We provided evidence that the ANN reduces the temporal error in a system that is mainly corrupted by the noise, aliasing, fitting and temporal errors. For on-sky data, we achieved a $46.5 \pm 2.4\%$ reduction in the temporal, noise and aliasing errors. Disentangling these three errors from one another was complicated by the presence of the ANN that modified some of the standard AO analysis techniques that allow, for example, an independent estimation of noise.

Apart from accurately predicting the wavefront, we have provided evidence that the predictor is also filtering high temporal frequency components such as Wavefront Sensor (WFS) noise and aliasing errors. This behaviour is however dependent on the training noise level of the WFS and was observed only with predictors trained with a noise variance of 2.8×10^{-4} arcsec^2 or higher. This has the potential to alleviate the stability issue caused by additional noise in the Pseudo Open Loop Control (POLC). How this optimal training noise limit may vary for systems other than low-order 7×7 SHWFS of CANARY was not investigated however, but we leave this open for future investigation.

We have shown in simulations that the predictor is robust to changes in wind velocity on sub-second timescales, and that the ANN is insensitive to changes in turbulence strength r_0. The predictor trained with a single turbulence layer is capable of predicting in more complex conditions with up to 35 layers each with distinct wind vectors, albeit with reduced performance. However, the performance of a frozen-flow trained ANN when applied to on-sky CANARY data with a ground-layer dominated turbulence profile did not show a significant performance improvement. We identified that the CANARY Datasets were contaminated by vibrations in Tip and Tilt (TT) modes and that the observed turbulence was dominated by an evolving static layer (indicative of dome seeing) instead of frozen flow assumed in simulations. We adopted an empirical model of dome turbulence to fit to the observed temporal properties of the CANARY data (excluding vibrations), and showed that an ANN trained using this model significantly outperforms a frozen flow predictor trained in the same noise condition.

We have shown that both the frozen-flow or dome turbulence model trained ANN cannot filter or predict vibrations. This can potentially degrade the control system performance. The removal of vibrations before the ANN prediction can thus be beneficial. We note that it is likely the ANN architecture used within this thesis could also be used to identify and/or predict vibrations, but this study was not performed due to limited time and can be investigated at a later date.

A non-spatially aware predictor proves competitive both with frozen flow and dome seeing. The knowledge of the spatial distribution of WFS subapertures is most beneficial in suppressing high temporal frequencies such as noise and aliasing. A non-spatially aware predictor can predict the slowly-evolving components (below around 10 Hz) with more accuracy in all test conditions. Additionally, the parallel nature of the operation of a non-spatially aware predictor may make it well suited for high-order Extreme Adaptive Optics (XAO) systems. We have not compared the computational load of the ANN approach presented here with existing linear predictive techniques applied to higher-order AO systems. The difficulty lies in that the computational load of a spatially aware ANN scales in an unknown form with the AO order due to the nonlinear nature of its implementation, and that different techniques may benefit from parallelism to varied degrees. We will leave this to future study.

6.3 Implications for On-Sky Implementation

An ANN wavefront predictor should be trained with simulated data. The training method presented here uses simulated SHWFS slope data which could be applied to any real WFS data as has been shown. However, the sensitivity of the ANN performance to the training regime and the requirement for a large amount of training data acquired under the same conditions means that the collection of a real WFS dataset may take a significant amount of time. It may therefore be best to initially train in simulation and convert real WFS slopes to ensure that the subaperture geometry and pixel scale matches that encoded within the ANN. However system and turbulence characterisation is necessary before the ANN training for the identification of WFS noise level and the dynamics of the turbulence such as the existence of vibrations and/or frozen flow.

The technique proposed here inherently scales to multiple guide star systems through parallelism. An on-sky implementation of the ANN presented here is equivalent to an additional processing step for each WFS before reconstruction within the system. The turbulence profile can be derived from multi-WFS data that may provide information on frozen-flow wind velocities and non-frozen ground conjugated turbulence conditions in real time. This may make a system more robust to complex multi-layer turbulence, but at the expense of a more complex training regime. This will be subject to future study.

Appendix A

The Dependence of ANN Frequency Transfer on Wind Direction

We have observed the sensitivity of a spatially aware Artificial Neural Network (ANN) predictor to the relative direction between the wind and a Zernike mode in simulations, where the frozen flow hypothesis is assumed. This is reflected as the magnification of small temporal frequencies (below 10 Hz) by the ANN in some modes and the transition of ANN response between modes when the wind direction varies.

Figs. A.1 and A.2 display the transfer of simulated temporal Power Spectral Density (PSD)s of Zernike modes 2-36 (from top to bottom, left to right) by the **Mag-10**, **Mag-8** and **Noise-free** predictors in the noise-free condition. Wind directions are 10 and 90 degrees respectively. We have seen the same kind of plot (Fig. 4.9) when the wind direction is 0 deg. In a few modes (for example modes 17 and 23 in this figure), all predictors especially the **Mag-10** predictor exhibit some magnification. We have observed that this magnification transits between modes when the wind direction varies. When the wind direction changes from 0 to 90 degrees, as is shown in Figs. 4.9 and A.2, the magnification transits between adjacent modes with the same odd azimuthal order (for example, from mode 17 to 16). This can be interpreted as these two modes interchange when the axis rotates odd multiples of 90 degrees (equivalent to the change of wind direction). For two modes with the same azimuthal order ($m = 1, 2, ..., 7$), when the rotation angle equals $\frac{n\pi}{2m}$, where $n = 1, 3, 5, ...$, these two modes interchange. Based on this, another observation is when the wind direction (e.g. 10 deg) is away from any of these angles, the magnification becomes much less severe in all modes (see Fig. A.1).

However, a non-spatially aware predictor is less sensitive to the wind direction and treats different Zernikes roughly the same. See Figs. A.3 and A.4 for the transfer by spatially aware and non-spatially aware predictors in the noise-free condition. The wind direction is 0 and 90 degrees respectively.

For CANARY telemetry, which has been shown to be dominated by a non-translating, evolving turbulence, we did not observe the discrepancy in the transfer of PSDs across Zernike modes (see Fig. A.5). The wind speed is effectively 0. The two predictors shown are both spatially aware.

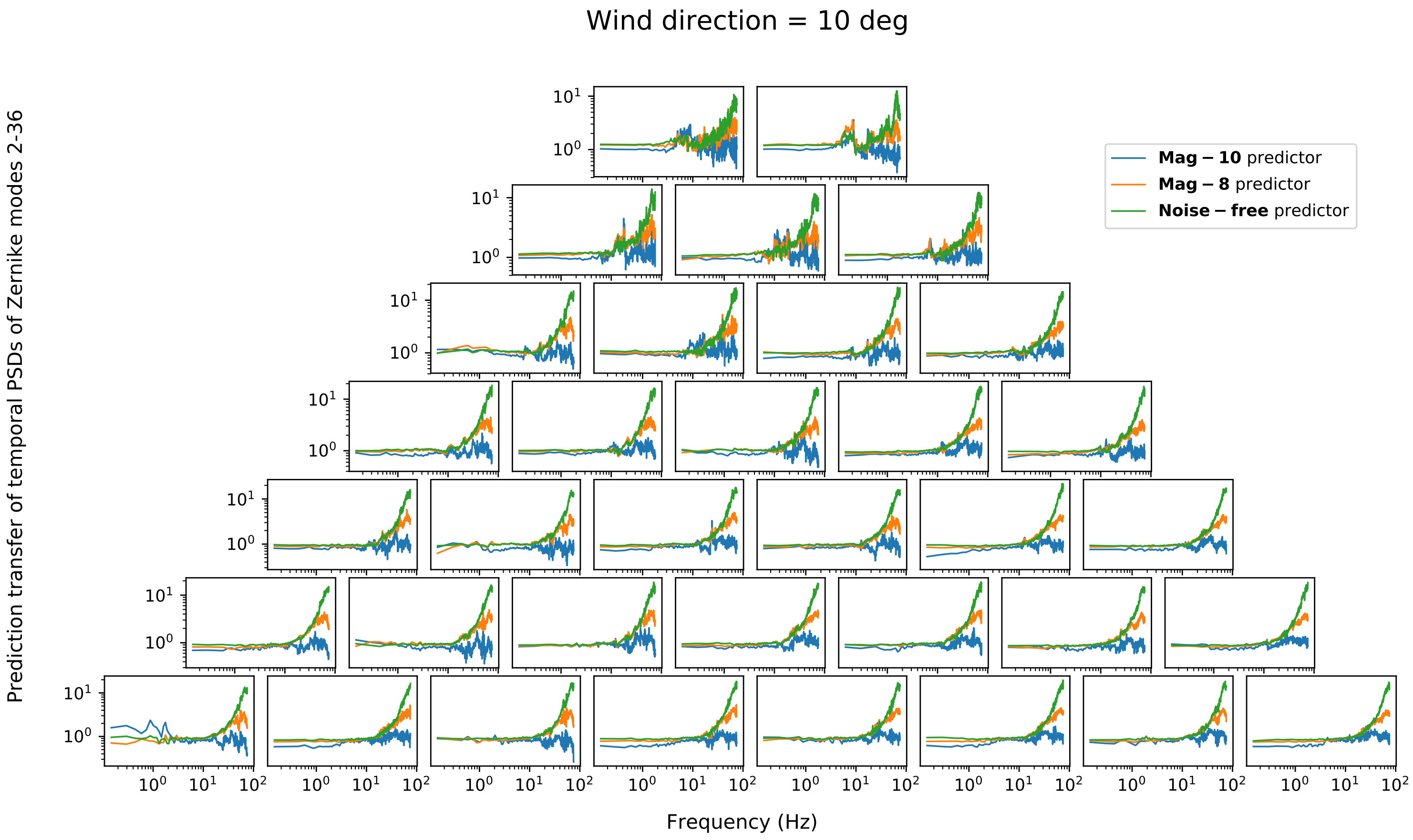


Figure A.1: Transfer of temporal PSDs in Zernike modes 2-36 (from top to bottom, left to right) by **Mag-10**, **Mag-8** and **Noise-free** predictors when the wind direction is 10 degrees.

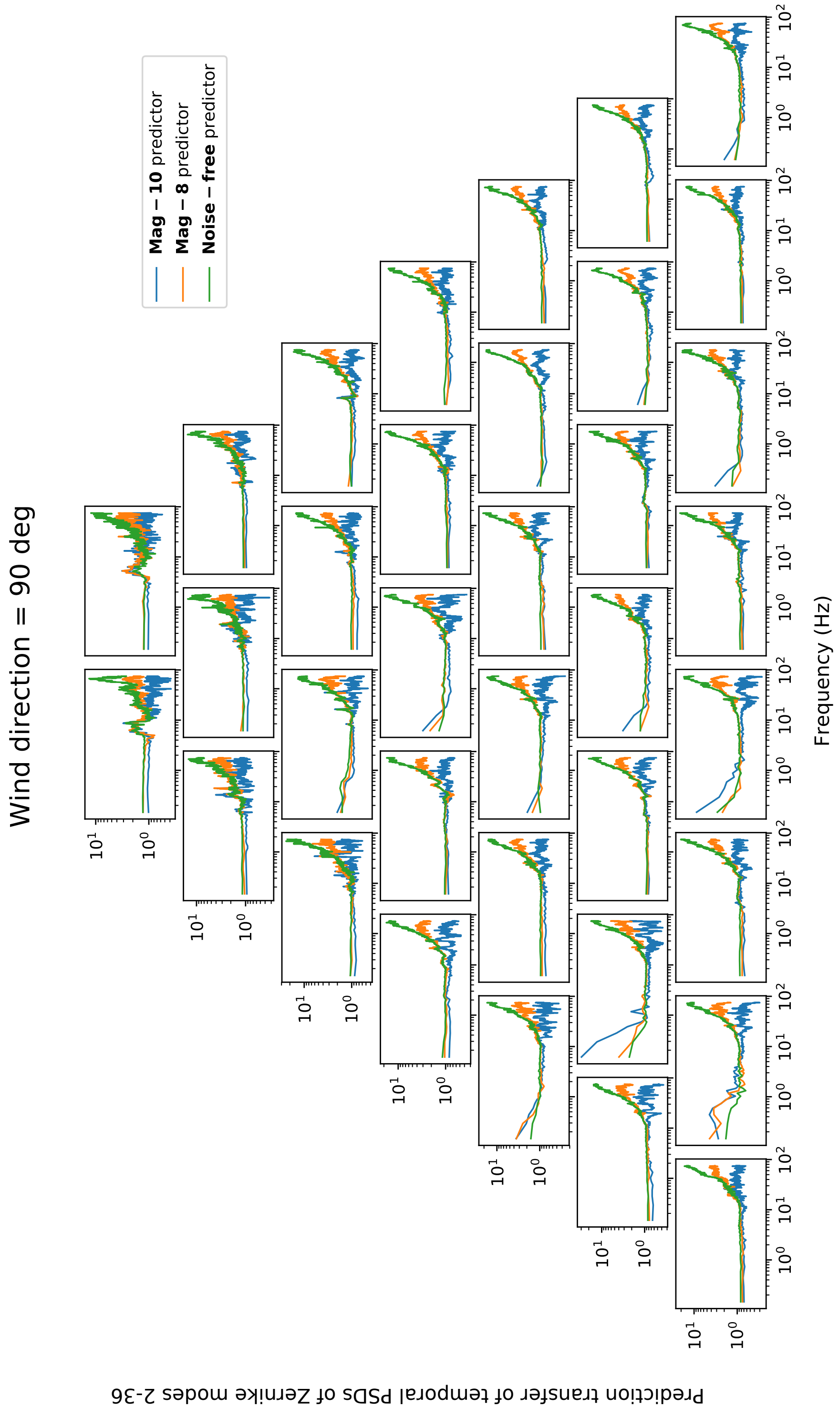


Figure A.2: Transfer of temporal PSDs in Zernike modes 2-36 (from top to bottom, left to right) by **Mag-10**, **Mag-8** and **Noise-free** predictors when the wind direction is 90 degrees.

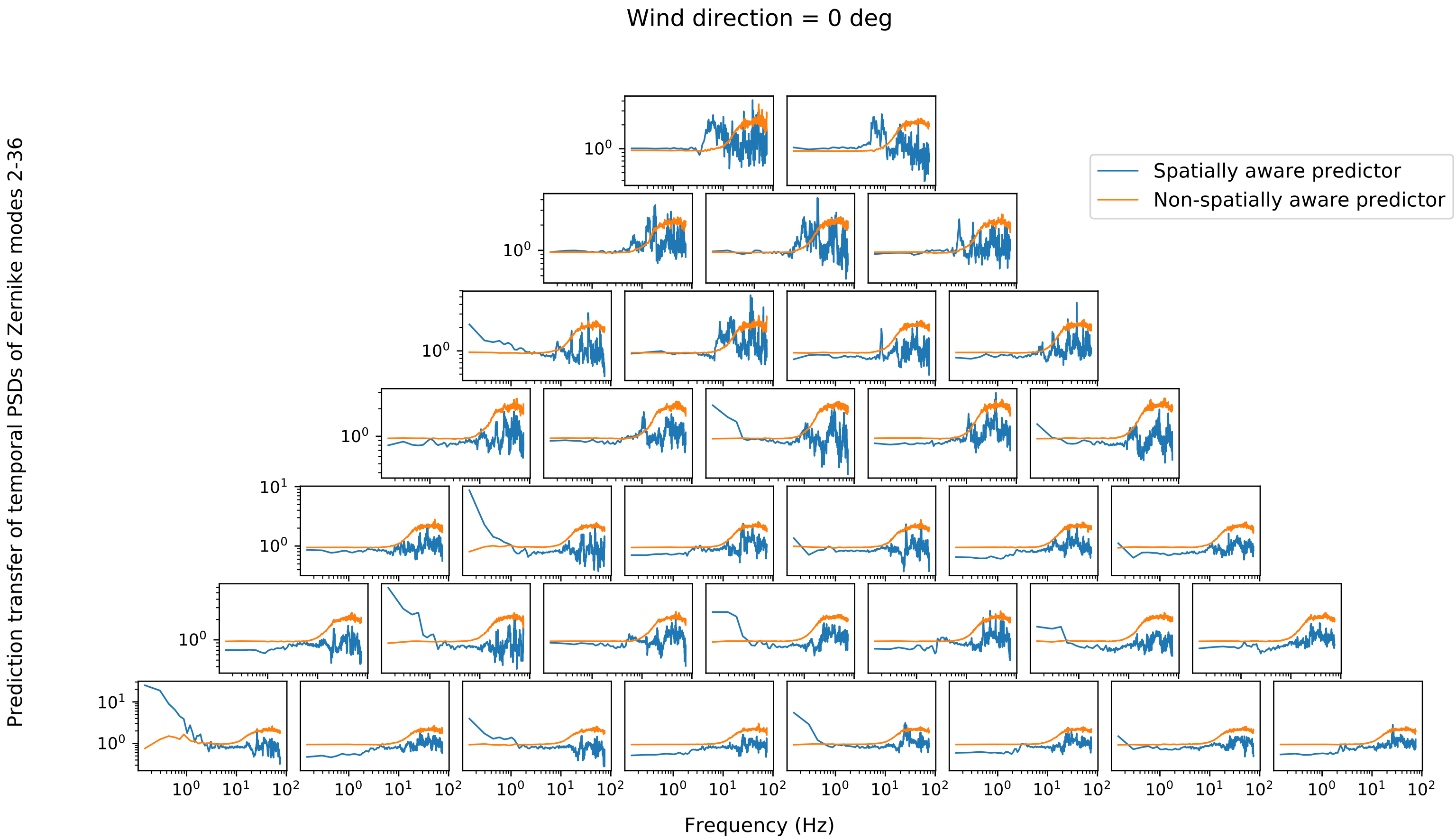


Figure A.3: Transfer of temporal PSDs in Zernike modes 2-36 by the spatially and non-spatially aware predictors when the wind direction is 0 degree.

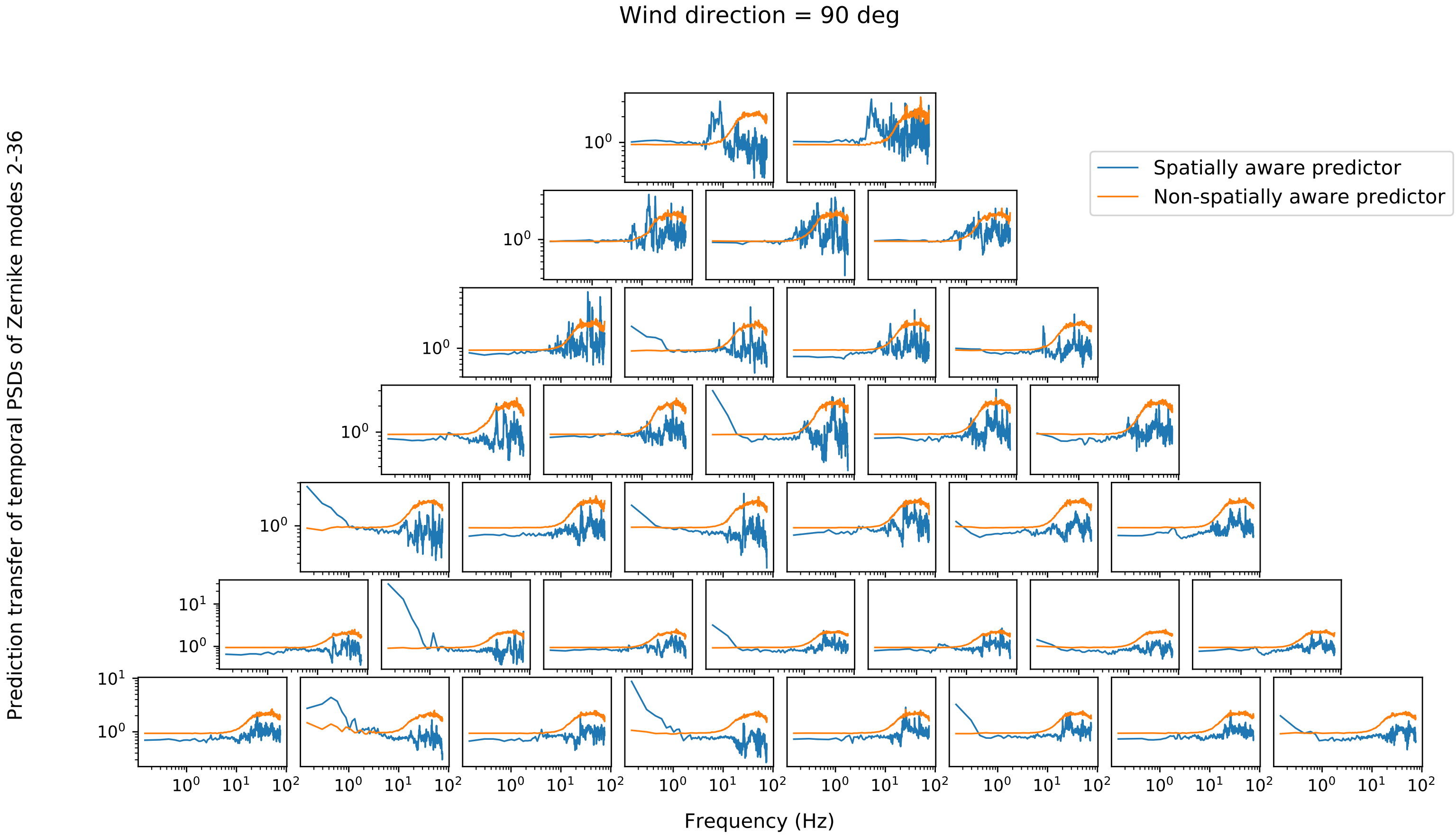


Figure A.4: Transfer of temporal PSDs in Zernike modes 2-36 by the spatially and non-spatially aware predictors when the wind direction is 90 degrees.

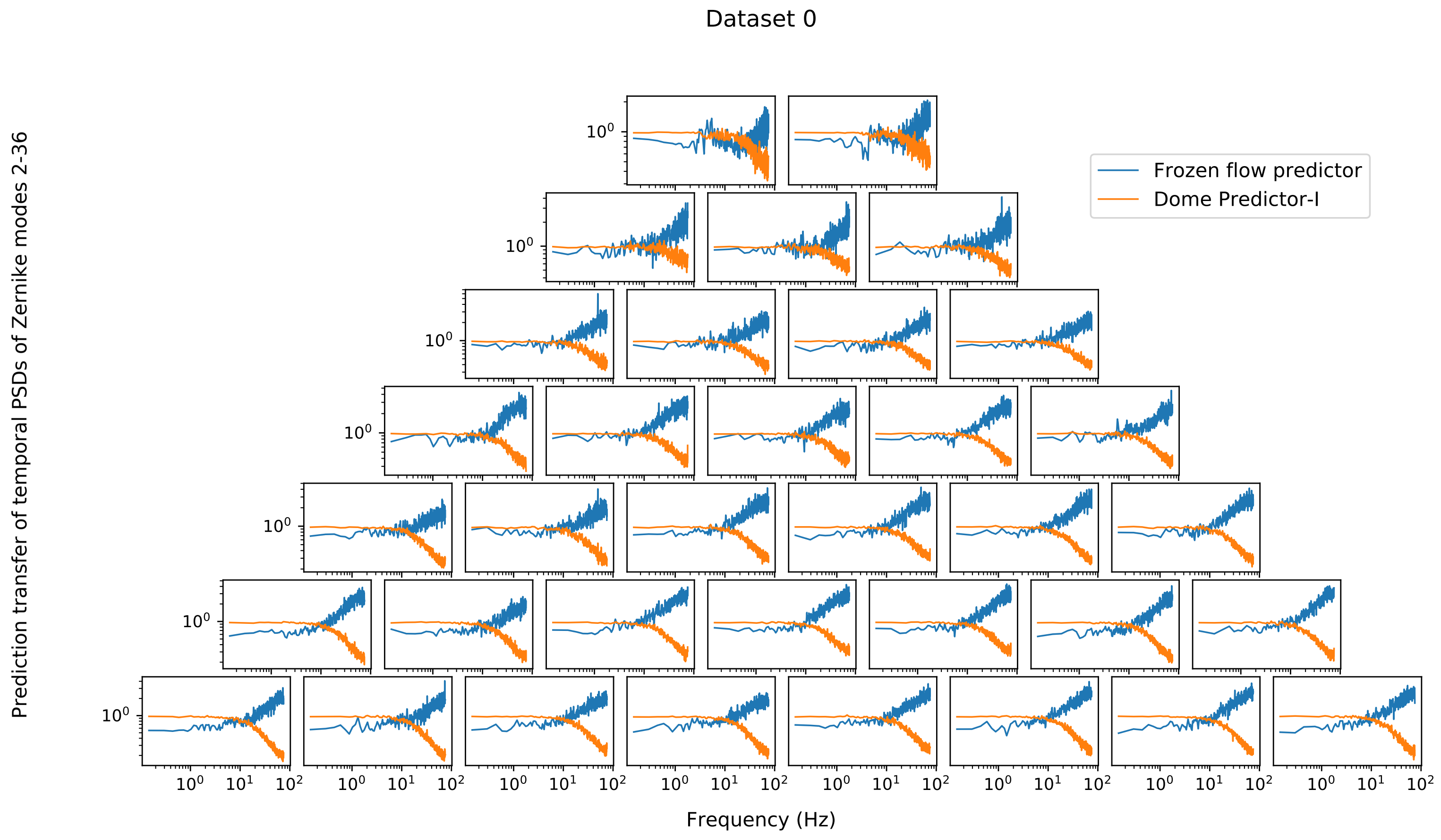


Figure A.5: Transfer of temporal PSDs in Zernike modes 2-36 by the frozen flow predictor and dome Predictor-I with CANARY Dataset 0. With CANARY Datasets, no distinct wind translation of turbulence layers was detected.