

Data-driven subspace predictive control of adaptive optics for high-contrast imaging

Sebastiaan Y. Haffert¹,^{a,*}† Jared R. Males,^a Laird M. Close,^a
Kyle Van Gorkom¹,^{a,b,c} Joseph D. Long¹,^a Alexander D. Hedglen,^{a,b}
Olivier Guyon¹,^{a,b,d,e} Lauren Schatz,^{a,b} Maggie Kautz¹,^{a,b}
Jennifer Lumbres¹,^{a,b} Alex Rodack,^{a,b} Justin M. Knight,^{a,b}
He Sun¹,^f and Kevin Fogarty^{g,h}

^aUniversity of Arizona, Steward Observatory, Tucson, Arizona, United States

^bUniversity of Arizona, Wyant College of Optical Science, Tucson, Arizona, United States

^cNASA Goddard Space Flight Center, Greenbelt, Maryland, United States

^dNational Institutes of Natural Sciences, Astrobiology Center, Mitaka, Tokyo, Japan

^eNational Astronomical Observatory of Japan, Subaru Telescope,

National Institutes of Natural Sciences, Hilo, Hawaii, United States

^fCalifornia Institute of Technology, Department of Computing and Mathematical Science,
Pasadena, California, United States

^gCalifornia Institute of Technology, The Division of Physics, Mathematics and Astronomy,
Pasadena, California, United States

^hNASA Ames Research Center, Moffett Field, California, United States

Abstract. The search for exoplanets is pushing adaptive optics (AO) systems on ground-based telescopes to their limits. One of the major limitations at small angular separations, exactly where exoplanets are predicted to be, is the servo-lag of the AO systems. The servo-lag error can be reduced with predictive control where the control is based on the future state of the atmospheric disturbance. We propose to use a linear data-driven integral predictive controller based on subspace methods that are updated in real time. The new controller only uses the measured wavefront errors and the changes in the deformable mirror commands, which allows for closed-loop operation without requiring pseudo-open loop reconstruction. This enables operation with non-linear wavefront sensors such as the pyramid wavefront sensor. We show that the proposed controller performs near-optimal control in simulations for both stationary and non-stationary disturbances and that we are able to gain several orders of magnitude in raw contrast. The algorithm has been demonstrated in the lab with MagAO-X, where we gain more than two orders of magnitude in contrast. © 2021 Society of Photo-Optical Instrumentation Engineers (SPIE) [DOI: [10.1117/1.JATIS.7.2.029001](https://doi.org/10.1117/1.JATIS.7.2.029001)]

Keywords: adaptive optics; exoplanets; high-contrast imaging; coronagraph; spectroscopy.

Paper 20143 received Sep. 18, 2020; accepted for publication Mar. 9, 2021; published online Apr. 6, 2021.

1 Introduction

The upcoming generation of Giant Segmented Mirror Telescopes (GSMT) has the light gathering capability and angular resolution to directly image Earth-like planets around other stars, which allow us to search for bio-signatures. However, these ground-based telescopes do not operate at their diffraction-limit due to turbulence in the Earth's atmosphere. And even if these large telescopes could be used close to their diffraction-limit, the star is usually orders of magnitude brighter than the planet, making it difficult to distinguish the planet from the star.¹ High-contrast imaging (HCI) instruments are designed to overcome these challenges by using extreme adaptive optics (ExAO) to compensate for atmospheric disturbances and recover the angular resolution, while advanced coronagraphs are used to remove the influence of starlight.²

*Address all correspondence to Sebastiaan Y. Haffert, shaffert@arizona.edu

†NASA Hubble Fellow

With the current generation of HCI instruments,^{3–5} we can routinely reach post-processed contrast levels of 10^{-4} to 10^{-6} , depending on the angular distance from the host star. With these contrast levels, we are sensitive to hot and massive self-luminous planets.⁶ Even though we are sensitive to massive Jupiter-like planets, only a few planets have been imaged and spectroscopically characterized. The results from large surveys that targeted massive Jupiters on wide orbits indicate that the planet occurrence rate drops sharply between 1 and 10 AU.^{7–10}

More planets could be directly imaged if the sensitivity close to the star is improved. The main limiting factor close to the star is the performance of the adaptive optics (AO) systems.^{11–13} During the operation of an AO system, a wavefront sensor measures the wavefront aberrations, which are then fed back to a DM for correction. This causes an inevitable delay because the system can only compensate after it has measured the disturbance. There are two options to reduce the servo-lag error in AO systems, either run the full system at a higher speed, which is currently being tried¹⁴ or by the use of predictive control. In predictive control, a model of the system is used to make a prediction about the future state of the system. This allows the predictive controller to anticipate future behavior and mitigate errors before they happen. Predictive control has been proposed some time ago as a solution to the servo-lag error.¹⁵ If predictive control is successfully implemented it could lead to a gain of two or three orders of magnitude in contrast, under the assumption that much of the temporal evolution is predictable.^{16–18}

Initial results from predictive control focused on low-order modes, which was mainly due to restrictions in computing power and the availability of high-order deformable mirrors (DM).^{19,20} From that point two distinct paths of predictive control appeared, one focused on physical modeling of the atmosphere and its dynamics^{21–23} and the other focused on data-driven methods.^{16,24} Recently, a large amount of effort has been put into investigating the application of deep reinforcement learning for predictive control.^{25,26} There are also algorithms where model-based and data-driven approaches are merged.²⁷

Currently, both paths are maturing and are being implemented in laboratory and on-sky demonstrations. The initial results are encouraging, showing increased Strehl ratio in many conditions.^{27,28} The gain in contrast for ExAO systems however has been limited.²⁹ A major limitation for ExAO systems that is common in the previous methods is that they require a sequence of open-loop wavefront sensor measurements to estimate certain aspects of the system, e.g. the power spectral density (PSD),²⁰ correlation matrix of all inputs and outputs,¹⁶ or the wind velocity of turbulent layers.²¹ The wavefront sensor of interest for HCI is the pyramid wavefront sensor,³⁰ which has limited dynamic range. Due to the limited dynamic range, it may not be possible to reconstruct the disturbance PSD from open-loop measurements. Furthermore, variable calibration and inherent non-linearity of the PWFS can change the optical gain, which effectively changes the modal sensitivity.³¹ If there is no correction for the optical gain, the pseudo-open loop reconstruction will under or overestimate the wavefront. Online gain tracking is necessary to compensate for this effect. And finally, the atmosphere itself is dynamic and exhibits non-stationary turbulence. This is limiting the application of open-loop predictive controllers.³²

Another complication stems from the use of multi-stage AO systems, where several dynamical components are operated at the same time, but each may be operated at different frequencies. The temporal dynamics of each component need to be accurately accounted for to reconstruct the full open-loop disturbance. Any model error in the DM dynamics immediately folds into the disturbance reconstruction. Additionally, ExAO systems are being operated at very high speeds close to the operating frequency of DMs, which means that the DM cannot be modeled with a fixed-lag step response and the temporally resolved dynamics need to be taken into account.

In this work, we propose a completely model-free data-driven method that identifies the atmospheric and system dynamics in closed-loop and is updated online to track non-stationarity. Our method is based on a closed-loop data-driven subspace identification algorithm, which has been gaining traction recently.^{33,34} The principle of the data-driven closed-loop subspace predictive controller (DDSPC) is to obtain the predictive controller directly from the measured input-output pairs during closed-loop operations, without any intermediate parametric model identification. In this work, we will derive a predictive controller based on the DDSPC method that uses integral-action, which will drastically lower the influence of model errors and allows it to be used in closed-loop.

In Sec. 2, we derive the closed-loop data-driven subspace identification algorithm and its controller. We numerically demonstrate the power of the algorithm for predictive control in Sec. 3, and in Sec. 4, we show the verification of the algorithm in a lab setting. Section 5 concludes the paper and gives an outlook on the implementation for on-sky.

2 Data-Driven Closed-Loop Subspace Identification

The proposed DDSPC controller works in a mode space of the DM, e.g., actuator space or Zernike space. We assume that the modal coefficients are retrieved from a WFS that measures the wavefront error in closed-loop at regular time intervals. The reconstructed wavefront error at timestep i is then defined as Δy_i . At each time step i there is also a new command that will be sent to the DM to compensate for the wavefront errors, which is defined as Δu_i . It is important to note here that the DDSPC controller is working with the changes in the wavefront and not the full wavefront and actuator commands. This will allow the predictive controller to retain integral action.

2.1 Defining the Auto-Regressive Model

The key aspect of the predictive controller lays in its ability to predict the future WFS measurements. The future measurements of the WFS depend on the past wavefront sensor measurements from the atmosphere changing between time steps and past DM commands due to non-instantaneous DM dynamics. And finally, the future measurements also depend on the future commands, because at every time step a new command will be sent to the DM which has temporal dynamics that need to be taken into account. We will assume that a linear auto-regressive (AR) model is enough to capture all dynamics of the system. The AR model for the measurement at the next time step is

$$\Delta y_{i+1} = \sum_{n=i-N}^{n=i} a_n \Delta y_n + \sum_{n=i-N}^{n=i} b_n \Delta u_n + \sum_{n=i+1}^{n=i+M} c_n \Delta u_n. \quad (1)$$

This equation consists of three parts, the AR model for the past N measurements with coefficient a_n , the AR model for the past N commands with coefficients b_n ; and finally, the AR model for the M future commands with coefficients c_n . The order (N) for the past commands and measurements do not have to be equal. For practical purposes, we kept them equal. The model of Eq. 1 can be rewritten in a clearer vector notation by defining the data vector which contains a sequence of length N of either measurements or commands as

$$x_{i+N:i} = (x_{i+N} \quad \dots \quad x_i)^T. \quad (2)$$

Here, x_i is the value of variable x at time step i . If a prediction horizon of M steps is considered with N past measurements, the notation can be further simplified. To do this, the future measurements at time step i are defined as $y_f^i = \Delta y_{i+M:i+1}$ and the past measurements as $y_p^i = \Delta y_{i:i-N}$. The future and past commands are defined in a similar manner, $u_f^i = \Delta u_{i+M:i+1}$ and $u_p^i = \Delta u_{i:i-N}$. With the simplified notation the AR model becomes,

$$y_f^i = A y_p^i + B u_p^i + C u_f^i. \quad (3)$$

With A the prediction matrix of size $M \times N$ that correlates the past N measurements to the future M measurements, B the prediction matrix of size $M \times N$ that correlates the past N commands to the future M measurements, and C is the prediction matrix of size $M \times M$ that correlates the future commands to the future measurements.

2.2 Online Estimation of the Prediction Matrices

The AR model needs to be updated every time step to take into account non-stationary statistics. Therefore, we need to do an online identification of the AR model. The problem can be reduced to a single matrix-vector multiplication because it is a linear problem,

$$y_f^i = \begin{bmatrix} A^i & B^i & C^i \end{bmatrix} \begin{bmatrix} y_p^i \\ u_p^i \\ u_f^i \end{bmatrix} = \Theta^i \phi^i. \quad (4)$$

Here, ϕ is the concatenation of y_p^i , u_p^i , and u_f^i , while Θ is the concatenation of A , B , and C . This can be solved in a straightforward way by defining the least-squares solution,

$$\Theta = \arg \min_{\Theta} \|y_f^i - \Theta \phi^i\|_i. \quad (5)$$

Recursive least-squares (RLS) are used to update our prediction matrix at every time step. RLS consists of four steps, of which the first is the determination of the update gain,

$$K^i = \frac{\gamma^{-1} \phi^i P^{i-1}}{1 + \gamma^{-1} \phi^{iT} P^{i-1} \phi^i}. \quad (6)$$

Here, K^i is the gain matrix at time step i , γ is the forgetting factor that is included to reduce the weight of measurements far in the past, and P^i is the inverse covariance matrix between all features at time step i . The next step is to calculate the prediction error at step i ,

$$e^i = y_f^i - \Theta \phi^i. \quad (7)$$

After which the prediction matrix can be updated,

$$\Theta^{i+1} = \Theta^i + K^i e^i. \quad (8)$$

And finally, the inverse covariance matrix is updated

$$P^i = \gamma^{-1} P^{i-1} - \gamma^{-1} K^i \phi^i P^{i-1}. \quad (9)$$

The RLS algorithm is closely related to the Kalman filter. A benefit of the RLS algorithm is that it is an $\mathcal{O}(n^2)$ algorithm, as opposed to the Kalman filter, which is an $\mathcal{O}(n^3)$ algorithm. This difference in algorithmic complexity allows the RLS algorithm to update its matrices in real time, while for the Kalman filter the gain matrix is almost always calculated offline.³⁵

2.3 Derivation of the Controller

The results from the previous two sections can now be used to derive the finite horizon least-squares optimal controller. First, a cost function has to be defined that the controller will optimize. A logical choice is the mean square error of the future M wavefront errors because those are the errors that are predicted. This leads to the following cost function,

$$J_i = \sum_{k=1}^M y_{i+k}^T y_{i+k} + \lambda \sum_{k=1}^M u_{i+k}^T u_{i+k} = y_f^{iT} y_f^i + \lambda u_f^{iT} u_f^i. \quad (10)$$

The future commands have been added to the cost function with a scalar λ to penalize large commands, which effectively regularizes the control algorithm. The best estimate that is available at time step i of the future measurements is the AR model described in Eq. (3), which upon substituting into Eq. (10) will provide a relation between the past measurements and control signals and the future control signals.

$$J_i = y_f^{iT} y_f^i + \lambda u_f^{iT} u_f^i = \begin{bmatrix} y_p^{iT} & u_p^{iT} & u_f^{iT} \end{bmatrix} \begin{bmatrix} A^T A & A^T B & A^T C \\ B^T A & B^T B & B^T C \\ C^T A & C^T B & C^T C \end{bmatrix} \begin{bmatrix} y_p^i \\ u_p^i \\ u_f^i \end{bmatrix} + \lambda u_f^{iT} u_f^i. \quad (11)$$

The regularisation λ can be absorbed into the cost matrix,

$$J_i = \begin{bmatrix} y_p^{iT} & u_p^{iT} & u_f^{iT} \end{bmatrix} \begin{bmatrix} A^T A & A^T B & A^T C \\ B^T A & B^T B & B^T C \\ C^T A & C^T B & C^T C + \lambda I \end{bmatrix} \begin{bmatrix} y_p^i \\ u_p^i \\ u_f^i \end{bmatrix} \quad (12)$$

The optimal controller can now be derived by finding the future control commands that minimize the cost function

$$u_f^i = \arg \min_{u_f^i} J_i. \quad (13)$$

This is relatively straightforward because the cost function is quadratic. To find the minimum, we take the derivative of the cost function with respect to the future commands and equate it to zero,

$$\frac{\partial J_i}{\partial u_f^i} = 2 \begin{bmatrix} A^T C & B^T C \end{bmatrix} \begin{bmatrix} y_p^i \\ u_p^i \end{bmatrix} + 2(C^T C + \lambda I) u_f^i = 0. \quad (14)$$

After some algebra we find the optimal control signal as

$$u_f^i = -(C^T C + \lambda I)^{-1} \begin{bmatrix} A^T C & B^T C \end{bmatrix} \begin{bmatrix} y_p^i \\ u_p^i \end{bmatrix}. \quad (15)$$

The controller itself is $K = -(C^T C + \lambda I)^{-1} \begin{bmatrix} A^T C & B^T C \end{bmatrix}$. This controller calculates the optimal control sequence for the next M steps. However, because the optimal control sequence is recalculated at every time step, only the next command is important.

2.4 Computational Considerations

Even though the RLS is more efficient than the Kalman filter, it is still a computational challenge to apply the DDSPC with coupling between all modes for a high-order AO system with thousands of modes. To ease the computation time, we implemented a distributed control scheme where we create a controller for each mode. This has the advantage that it is naturally parallel and therefore very easy to implement on distributed computing architectures, such as CPU or GPU clusters. A basis that is close to orthogonal is required for the distributed control. This will prevent the spatial correlations from folding into temporal correlations, which is otherwise very difficult to correct. The Gaussian actuator basis with low actuator cross talk or the Fourier basis are examples of possible mode bases. In this work, we will use the actuator basis.

The distributed controller works so long as spatio-temporal correlations between modes can be ignored. Recent work³⁶ showed with telemetry from VLT/SPHERE⁴ that using the past history of a mode is enough to predict the future state. Spatial correlations do not improve the prediction accuracy if the system is driven at high speeds because it takes too long for the wind to move the phase across a sub-aperture. For an AO system such as SPHERE, the time history needs to be at least 13 steps. For this, we assumed a 1-kHz loop speed, a windspeed of 15 m/s, and 41 wfs pixels across the pupil. And this is only the case for single-layer frozen flow turbulence. The influence of spatial correlations will be lowered even more if multiple (possibly boiling) layers are moving across the aperture. This implies that the proposed distributed approach is valid, and we do not have to consider modal correlations for predictive control. Distributed model predictive control (MPC) also has been shown to work well in simulations.^{21,35} Later sections in this work will show that the distributed approximation still results in near-perfect correction in our end-to-end simulations.

2.5 Parameter Investigation and Stability

It is not immediately clear that data-driven methods are guaranteed to be stable. A lot of research has focused on stability of data-driven methods in the past years, of which a lot has been driven by the advances in (deep) reinforced learning. At first, closed-loop data-driven MPC was shown to have stability guarantees.³⁷ Later, the stability was expanded to the data-driven subspace predictive control scheme.³⁸ The DDSPC that is proposed here is slightly different because we use the closed-loop residuals and the input-output model is update online. The data-driven MPC control is guaranteed to be stable if three conditions are met: the cost function needs to have a quadratic upper bound, the number of total number of data samples needs to be larger than the number of free parameters in the model, and the history length N needs to be larger than the future length M . While we have not proven that these conditions guarantee a stable controller for the DDSPC, they can still be used as a guideline when choosing the parameters of the controller. It is easy to meet the conditions. The quadratic upper bound is guaranteed because we use a quadratic cost function. The number of data samples puts a constraint on the forgetting factor. Each time, a new data point is added the old data will be multiplied by γ , which means that after T steps the weight of the data will be γ^T . We now assume that the data does not contribute anymore if its weight is smaller or equal to a threshold value ϵ . So, if we want to retain at least T samples, the inequality $\gamma^T \geq \epsilon$ must hold. The proposed model has $M \times (2N + M)$ free parameters. Consequently, the constraint on the forgetting factor is then $\gamma \geq e^{1/(2NM+M^2)}$. A typical model has between 40 and 100 free parameters. With a threshold value of $\epsilon = 0.1$, we find a lower bound of about 0.955 for the small models and a lower bound of 0.975 for the larger models. In future work, we will investigate if these conditions are also sufficient to guarantee stability for the DDSPC algorithm. Here, the stability is analyzed with end-to-end simulations.

The stability is tested by applying the DDSPC algorithm with varying history length (N) and injecting a range of power-law disturbances. The disturbances have the form of $P \propto f^\beta$, with f the temporal frequency and β the power-law index. Each generated time-series is normalized to a standard deviation of 1. For these simulations, we used a single frame delay. The prediction horizon (M) was set to three for these tests. The relative RMS as function of history length is shown in Fig. 1 and compared against a gain optimized integrator. This Figure shows that the DDSPC works significantly better than an integrator for power-law indices $\beta \leq -2$. For flat power spectra ($\beta \geq -1$), the integrator and predictor have about equal performance. Longer history lengths decrease the performance of the controller for the steeper power laws. While the shallow power laws prefer a longer history length that can be used to average out the noise. This also implies that there is an optimal history length when a mixture of shallow and steep power laws are present in the disturbance.

The DDSPC has the regularization parameter λ as hyper parameter, which was optimized for each power-law index and each history length. The optimal λ is shown in Fig. 2, which shows that the optimal regularization parameter should be very small $\lambda = 10^{-3}$ for most situations. For the noisiest cases, a strong regularization is preferred. This is expected because the optimal control is no control when either the measurements or the disturbance itself consist of pure noise.

The stability of the controllers is verified by inspecting their Nyquist Plot. The open-loop error transfer function (ETF) is created from the controller (K) using the equation for the frequency response of general linear filter.¹⁷ The Nyquist plot of each controller is shown in Fig. 3, which plots the real versus the imaginary part of the open-loop ETF. The system is stable in closed-loop if the ETF does not contain $-1 + 0i$ as a pole, which can be determined by the number of encirclements of the point $(-1,0)$. All systems do not encircle the unstable pole, which means that all derived controllers provide stable feedback. From this, we determine that the DDSPC is stable over a wide range of disturbances and parameter choices. These results do not guarantee stability for all systems. However, we have not yet encountered a situation, either in simulation or lab experiments, where the controller is unstable.

2.6 Learning Time

Data-driven algorithms will only have an advantage if they can learn the system parameters fast enough, which makes it important to examine the learning time scale. A long time sequence of

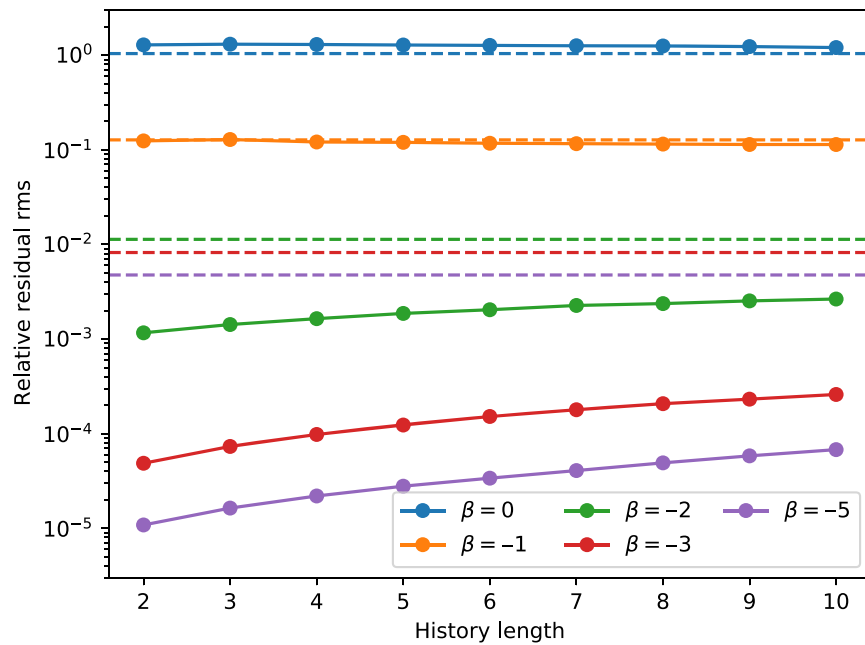


Fig. 1 The standard deviation of the residuals during closed-loop operations. The different colors indicate different power-law indices. The dashed lines represent the residuals for the gain optimized integrator. The solid lines show the residuals of the DDSPC. For $\beta \leq -2$, the predictive control has residuals that are at least an order of magnitude smaller. The residuals increase as the history length is increased for the steeper power-laws. For the shallow power-law disturbances, $\beta \geq -1$, the integrator and predictor behave quite similar.

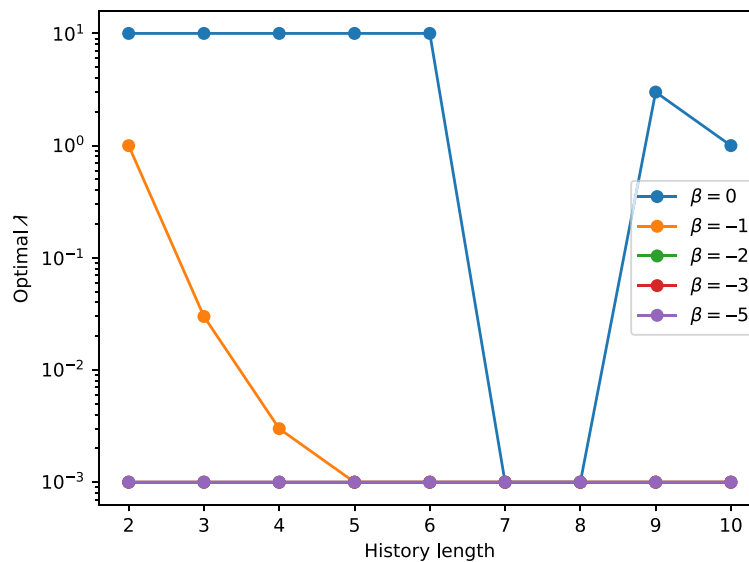


Fig. 2 The optimal regularization parameter for different power-law indices and different history lengths. The different colors represent different power-law indices. The optimal value is the smallest value that was considered for the steep power-laws ($\beta < -2$). For the shallow power-laws, a strong regularization is preferred.

30 s is simulated to estimate the scaling behavior of the algorithm. Frozen-flow turbulence with the parameters from 1 is used as the input disturbance. The initial prediction matrix is set to zero and inverse covariance matrix is initialized as a diagonal matrix with a large number, such as 10^9 , on the diagonal. A system identification (SI) approach is used to let the algorithm get familiar with the system it is controlling. In SI, the system is excited with a known disturbance on the

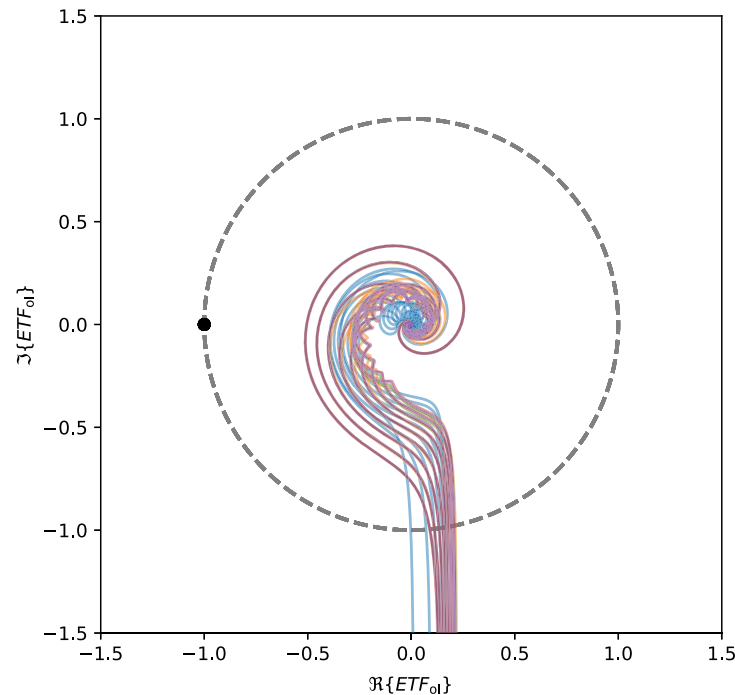


Fig. 3 Nyquist plot demonstrating the stability of the DDSPC controller for the cases considered in this section. The open-loop ETFs are plotted in the imaginary plane. It is noteworthy that frequency increases as the lines move out from the center. The controllers are stable so long as they do not encircle the black point $(-1,0)$. The gray dashed circle has a radius of 1. We show the open-loop ETFs for each controller. The color represents different power-law indices. All controllers are stable because they do not encircle the point $(-1,0)$.

control commands. For large systems, it is important to construct information-rich signals that are able to persistently excite all relevant frequencies. A popular choice is a random binary signal (RBS). This is a key point of the identification process because it is possible for the algorithm to get stuck in the null space of the instantaneous prediction matrices. For example, if the system starts with a zero prediction matrix and no control signal is injected, the algorithm will think that all temporal effects are intrinsic to the turbulence. This will keep the B and C matrices at zero, which means that the controller will not be learned regardless of the amount of training steps. RBSs will allow the control algorithm to explore all possible states and push the controller out of such null spaces. Therefore, we start the control loop with an additional exploration signal on top of the atmospheric disturbance. In principle, this can also be done during the day to minimize the loss of observing time during the night. The identification is most efficient if the amplitude of the RBS signal is larger than the measurement noise.

An example of the identification process can be seen in Fig. 4. The initial RMS of the DDSPC shoots up because the system is not yet known. After several tens of iteration, the controller is already performing better than an integral controller. Even with the RBS included, the DDSPC has lower RMS than a classical integrator. The integrator does not need to train, so the RBS is not necessary. The RBS signal was still injected to create an equal disturbance between both controllers. After the exploration period has ended at the 5 s mark, we see that the predictor is performing substantially better than an integrator. It is important to mention here that the considered RMS is not the RMS of the wavefront, but of the reconstructed modal coefficients. This means that the contribution of higher-order modes, e.g., the fitting error, has not been accounted for and the RMS in the figure cannot be used to determine the improvement in Strehl. The RMS can however be used to determine the contrast improvement because the modal coefficients determine the RMS within the control radius to first order. The results show that the RMS is reduced by more than an order of magnitude, which implies an improvement of almost a factor 100 in contrast. This is the major gain that predictive control can bring for HCI; while the

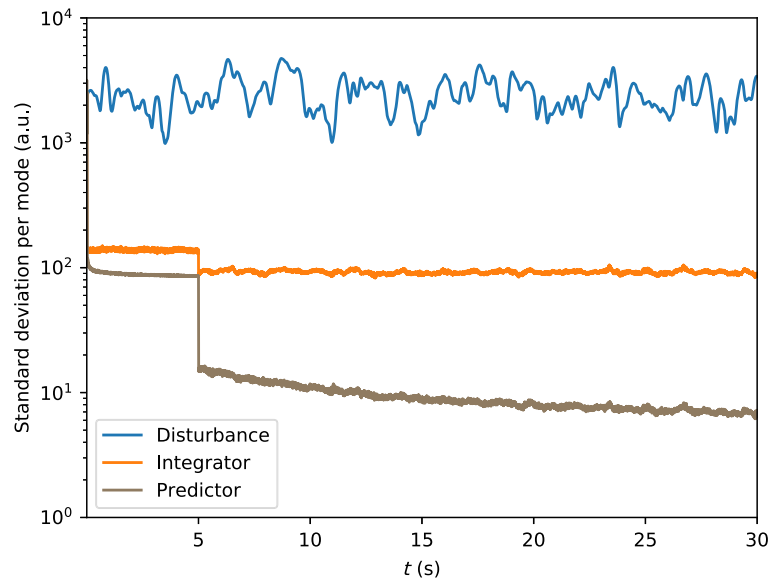


Fig. 4 The standard deviation of the residual modal coefficients during closed-loop operations. The first 5 s were used for exploration of the system with an RBS, which adds additional noise to the residuals. The influence of the RBS noise is visible through the drop of the RMS after the exploration is stopped. The DDSPC controller has learned most of the structure of the system within the first 100 ms, where the RMS of the DDSPC crosses the RMS of the integrator.

Strehl improvement is not large, there can be a substantial improvement in the contrast gain. This has been shown before in previous work.^{16,17,35}

After the exploration period, the DDSPC keeps learning how to improve the RMS even further, as can be seen in the part between 5 and 30 s of Fig. 4. The region from 5 to 30 s is plotted separately in Fig. 5 on a log–log scale to demonstrate the scaling behavior. A second line that

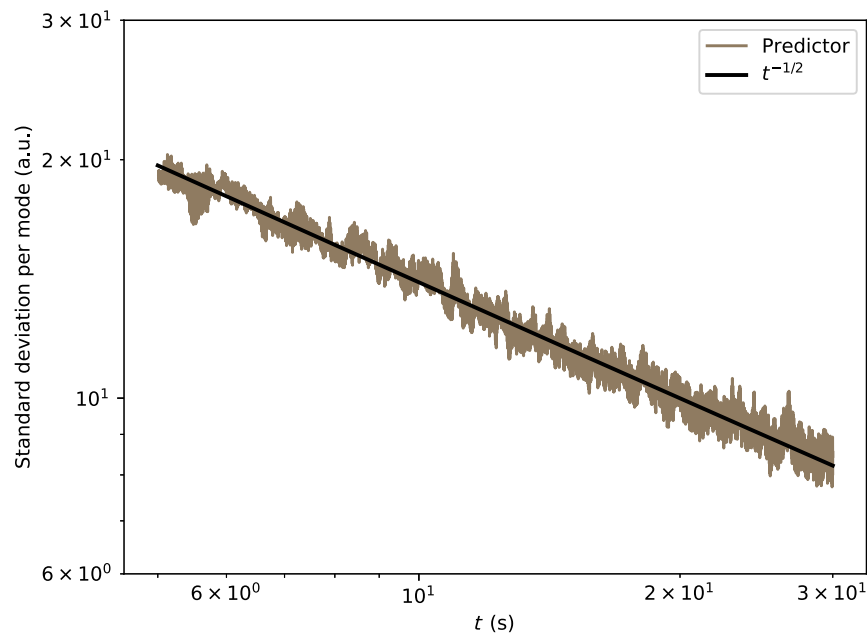


Fig. 5 A log–log plot of the RMS of the residual modal coefficients between 5 and 30 s after starting the control. In black, we show the expected statistical learning curve that scales as $1/\sqrt{t}$. The RMS of the DDSPC follows the expected learning curve even after more than 29,000 iterations. This indicates that the algorithm will keep learning to improve the RMS over time until all correlations are learned.

scales as $1/\sqrt{t}$ has been added for comparison. From this figure, we can see that the DDSPC algorithm is learning close to the rate of $1/\sqrt{t}$. The scaling also puts limits on how fast data-driven algorithms, such as the DDSPC, can adapt to changing conditions and the precision with which it adapts on that timescale. It is not clear from the data when DDSPC stops learning, a considerably longer time series is necessary to see if the model becomes more accurate or not and at what time it stops learning. This was not explored due to the required computation time.

3 Adaptive Optics Simulations

In this section, the performance of DDSPC is tested with end-to-end Monte Carlo simulations. We consider two cases: stationary and non-stationary frozen-flow turbulence. Both simulations use a direct wavefront sensor with infinite SNR. All simulations make use of the High-Contrast Imaging for Python (HCIPy) package.³⁹ We simulate an AO system similar to MagAO-X, see Table 1 for all parameters. The temporal response of the DM is simulated as a step response with one frame delay. The total system delay is then two frames because the act of measuring the wavefront error also delays the response by 1 frame.

3.1 Stationary Turbulence

In this section, we apply the DDSPC to single-layer atmospheric disturbances which evolve according to Taylor's frozen flow.⁴⁰ The DDSPC is compared to two other controllers. The first is what we will define as the perfect controller. The perfect controller is an instantaneous controller that perfectly removes all modes from the atmosphere at each time step. The second controller is a classical integrator with a fixed gain of 0.4. For the simulations in this section, we will use the same parameters as Table 1 except the wind speed that will vary between simulations. For each wind speed, we simulate 5 s (5000 frames) of data sampled at 1 kHz. The first 2 s (2000 frames) are used for exploration with an exploration noise strength of 2 rad per mode.

Table 1 Parameters of the end-to-end AO simulations.

AO parameter	Value	Comment
Pupil diameter	6.5 m	—
Actuators across pupil	50	Gaussian influence functions.
Integration time	1 ms	—
Total loop delay	2 frames	—
Atmospheric parameter	Value	Comment
r_0	0.16 m	Median 0.62" at 0.5 μm
Mean wind speed	15 m/s	—
WFS wavelength	800 nm	—
Integral control parameter	Value	Comment
g	0.4	Integrator gain.
Predictive control parameter	Value	Comment
M	3	The prediction horizon.
N	4	The number of past samples.
λ	0.01	Regularization parameter
γ	1.0	Forgetting factor

The history and future length of the DDSPC are 4 and 3, respectively. This choice of the horizon lengths is driven by the delay of the system. The simulated delay is two frames and to solve for the future command trajectory, we used a future horizon length that is one frame longer. Longer horizons can be used, but the states farther in the future are generally harder to predict, which will effectively introduce additional noise in the future trajectory. The history horizon was chosen based on the results from Sec. 2.5, which showed that longer history horizons do not necessarily improve the residuals due to over fitting. The history length was set to $N = 4$ to make sure that there is more input data for the prediction than outputs that need to be predicted. The additional benefit of smaller filters is also that they require less computation time, which is sparse at operating frequencies of 1 kHz or higher. Section 2.5 also showed that the regularization parameter λ should be set to something small, so we chose $\lambda = 0.01$.

The wavefront residuals for each controller are propagated through a perfect coronagraph^{41,42} to evaluate the contrast in the focal plane. We choose to evaluate the contrast at a wavelength of $1 \mu\text{m}$. An example of such a focal plane contrast map for a windspeed of 15 ms^{-1} can be seen in Fig. 6. The contrast of the perfect controller is on the order of 10^{-6} . There is still some residual structure in the dark hole, which comes from artifacts of the wavefront reconstruction step. In principle, these residuals could be removed by focal-plane wavefront sensing techniques. The integrator shows the typical wind-driven halo, which in this case severely limits the contrast. The DDSPC reaches a contrast that is very similar to the perfect controller. The only difference is a small decrease in contrast at the smallest angular separations. This can be better seen in the radial profile, as shown in Fig. 7. The radial profile shows that the contrast of the predictive controller is within a factor of 2 of the perfect controller.

Even though the DDSPC does not reach the fundamental limit, it still performs order of magnitude better than the integrator. The strength of the wind-driven halo is directly proportional to the wind speed. Figure 8 shows how the mean contrast within the dark hole depends on the wind speed. For very low wind speeds, the integrator reaches the same average contrast as the DDSPC. Because the atmospheric phase screen changes very slowly, the integrator has enough time to catch up. But as the velocity increases, the contrast degrades. The degradation as function of velocity follows the expected trend for Taylor's frozen flow approximation. Under Taylor's frozen flow, the phase screen is shifted by $\Delta r = \Delta v \Delta t$, while the integral controller assumes the phase screen stays static. The variance of the error is then proportional to the phase structure function because we are essentially comparing the phase screen with a shifted copy. The phase structure function itself has a power-law dependence under Taylor's frozen flow,

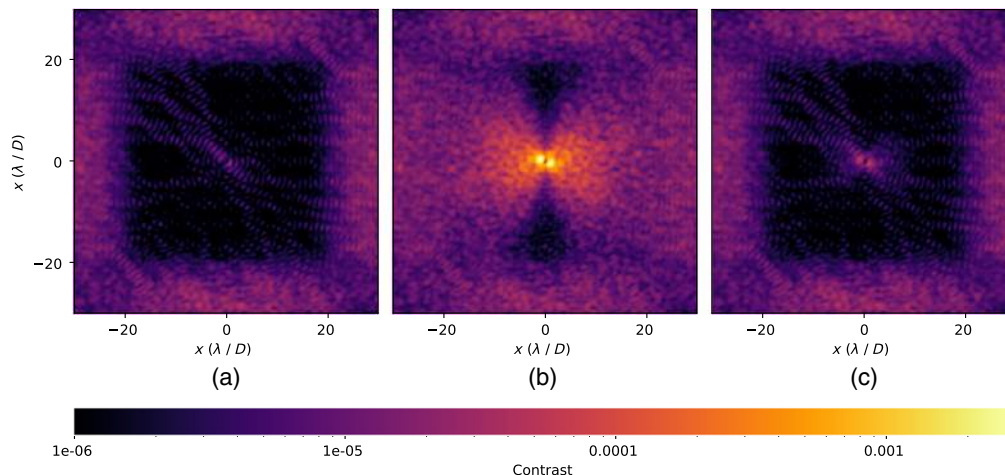


Fig. 6 (a) Post-coronagraphic residuals for different controllers. The post-coronagraphic stellar residuals for a perfect controller. The perfect controller still shows some residuals within the dark hole that are due to spatial fitting errors. (b) The results of a classical integrator where a strong wind-driven halo is limiting the contrast. (c) The DDSPC contrast map, where the majority of the wind driven halo is removed. The contrast has improved by almost two orders of magnitude. Most of the residual structure is similar to the structure in the dark hole of the perfect controller, which indicates that the contrast is also limited by the spatial fitting errors.

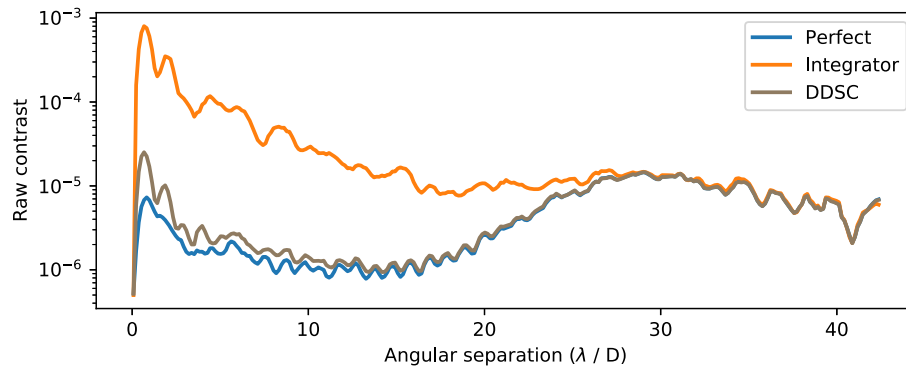


Fig. 7 The radial profiles of the post-coronagraphic contrast maps. The DDSPC shows two orders of improvement in contrast over the integral controller. The performance of the DDSPC at the diffraction-limit is still a factor 2 higher than the perfect controller.

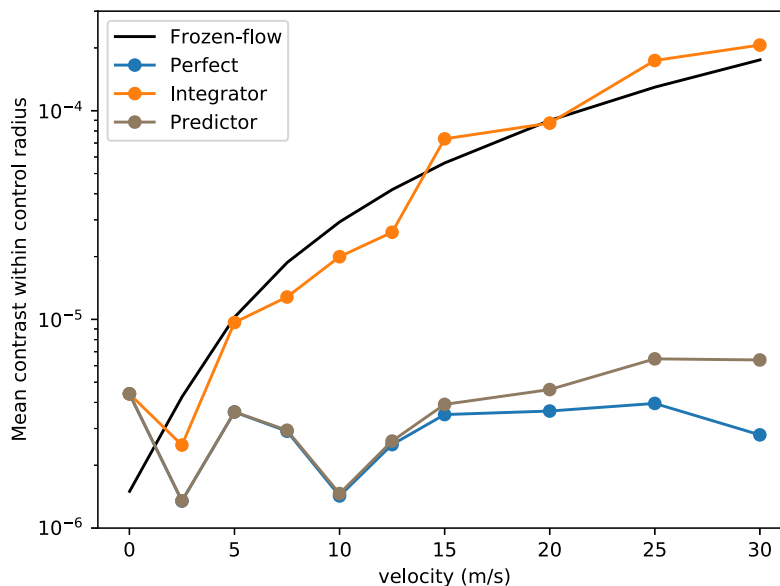


Fig. 8 The mean contrast within the control radius of the DM after a perfect coronagraph as function of the wind speed. The contrast is shown for the three different controllers. The black line shows the expected behavior for an integrator as function of velocity for frozen flow turbulence based on the phase structure function. The proportionality between contrast and velocity is derived in the Sec. 3.1. The actual measured contrast of the integrator (orange) follows the frozen-flow relation. The DDSPC controller stays very close to the perfect controller. Only at high wind velocities does the DDSPC start to degrade in performance, but it is still within a factor of 2 of perfect control.

$D(\Delta r) \propto (\Delta r)^{5/3} \propto (\Delta v)^{5/3}$. The residual contrast of the integrator follows this proportionality, as can be seen in Fig. 8. The contrast of an optimal predictive controller will also depend on the wind speed in the same way although with a lower constant of proportionality,⁴³ which is why the contrast starts to degrade at higher wind speeds. At lower wind speed ($<15 \text{ ms}^{-1}$), the DDSPC stays very close to the perfect controller. For the DDSPC, we kept the forgetting factor at 1, which effectively means that we are keeping the full history. The performance could still be improved by optimizing the forgetting factor or adding more data (like Fig. 4).

3.2 Non-Stationary Turbulence

One of the largest challenges of the model-based predictive controllers is the effect of changes in model parameters such as wind speed and direction.³² Therefore, it is crucial to test the algorithm

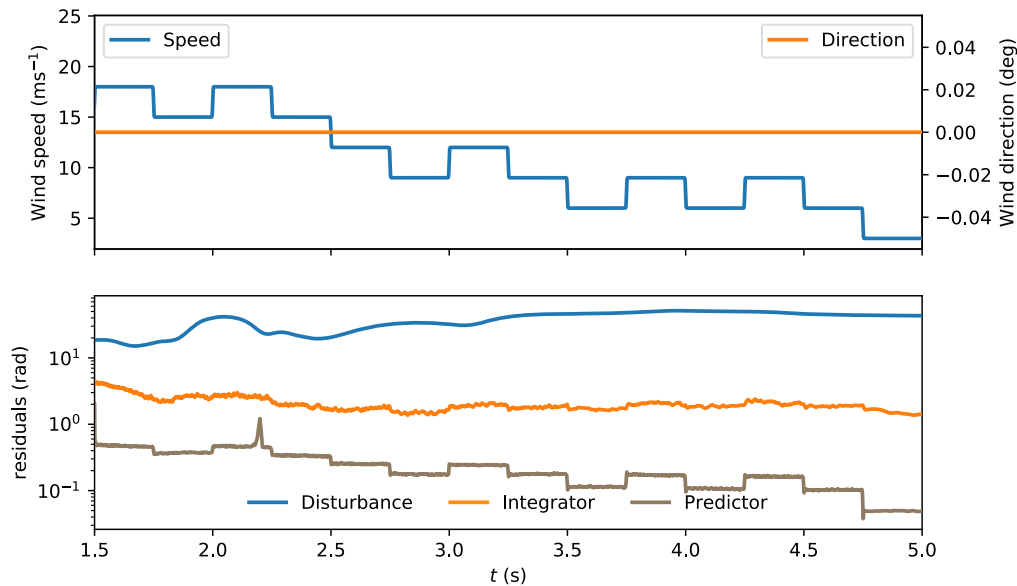


Fig. 9 The performance during sudden changes in wind speed. The top figure shows the wind speed and direction. The wind speed is changed by 3 ms^{-1} every 250 ms. The bottom shows the RMS of the modal coefficients as function of time for the input disturbance (blue), the integrator (Orange), and the DDSPC controller (Brown). The predictive controller has significant better performance than the integrator at all times. The performance of both the integrator and predictor is correlated with the wind speed.

against non-stationary turbulence. Several different simulations were performed to test the robustness of the DDSPC controller.

In the first simulation, the wind speed is changed by 3 ms^{-1} every 250 ms. Whether the speed increases or decreases is randomly chosen. Except for the wind speed, all other parameters of the simulation are still the same as Table 1. The results can be seen in Fig. 9. The first 1.5 s are used for training the controller and are not shown. The DDSPC controller is systematically performing better than the integral controller by one or two orders of magnitude. The RMS of the DDSPC is strongly correlated with the wind speed, which is expected based on the structure function. The integrator RMS is also correlated with the wind speed, but the relative changes are smaller than for the DDSPC controller. This indicates that a large fraction of the RMS of the integrator consists of the temporal dynamics of the DM. For the simulations here, the temporal dynamics of the DM is the delay because no additional temporal dynamics are included. This shows that the non-stationary wind velocities are not the limiting factor, but the delay itself.

Figure 10 shows a zoom-in of the response of the DDSPC between 3 and 4 s. This highlights the wind step response of the controller. We see that slight over and undershoots happen when the wind speed suddenly changes. The residuals quickly converge within four frames, which is the length of the history that is being used. When the wind speed changes, the history vector contains information about the wind speed before and after the jump. Only after four frames has the previous wind speed been completely removed from the history vector. Therefore we see that the RMS converges in about four frames.

The over and undershoots can also be explained from the frequency domain. The DDSPC controller uses the knowledge of the system dynamics to flatten the PSD of the disturbance. An optimal predictive controller tries to bring the output PSD as close as possible to a white output spectrum. This means that a predictive controller acts similar to a temporal deconvolution. Because we only approximate the system by a finite number of samples (the history length), we are susceptible to the Gibbs phenomenon. The Gibbs phenomenon shows up when a discontinuity is encountered and creates strong oscillations at the discontinuity. These initial simulations

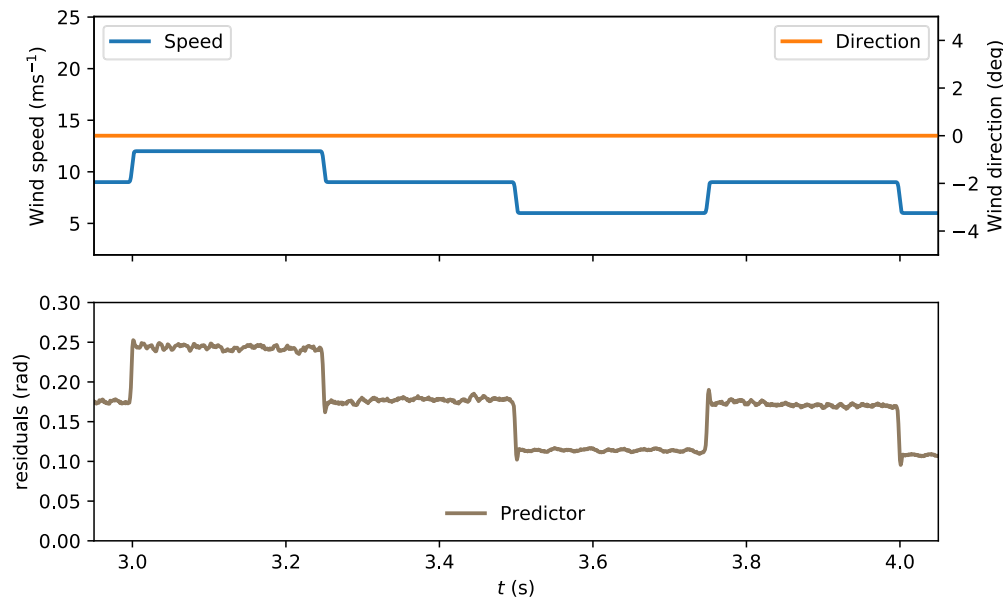


Fig. 10 A zoom in on the predictive controller residuals during step changes in the wind speed. The top figure shows the wind speed and direction. The bottom figure shows the RMS of the modal coefficients. There is a strong correlation between the wind speed and the controller performance. The response during a step change shows a small over or undershoot which settles within three frames. The step response shows behavior similar to the Gibbs phenomenon.

show that the DDSPC can handle sudden changes in the atmosphere quite well and only very few frames are required to take the new conditions into account.

The second simulation we perform includes both changes in the wind speed and wind direction. To stress-test the controller, we decided to create completely random changes in the wind speed and direction. Both variables perform a random walk. The average wind speed is 21 ms^{-1} with a standard deviation of 3 ms^{-1} . The wind direction changes by more than 270 deg during the simulation. This is a significant amount over 5 s . Because there is no pattern to the wind speed changes, the changes in the atmospheric disturbance become more random. This makes it more difficult to predict. Consequently, the predictive controller will show less gain over the integrator. The results are shown in Fig. 11. The predictive controller still performs better than the integrator, but the difference is smaller than in the previous test. We do however still see improvement as the time increases. The wind speed is relatively constant at 20 ms^{-1} around the 2 s mark with a residual RMS of 0.6 rad . At 4.5 s , the wind speed is back to 20 ms^{-1} again but this time the residuals are around 0.35 rad RMS. The improvement is roughly a factor of 1.7 . This improvement is very similar to the expected gain based on the learning rate, \sqrt{t} , from Sec. 2.6. That learning rate predicts a gain of $\sqrt{\frac{4.5}{2.0}} \approx 1.5$. The difference can be explained by the small differences in the actual wind speed. The wind speed is a little bit higher on average at 2 s if we compared it against the wind speeds at 4.5 s . This suggests that the controller has been constantly learning over this period, even under varying conditions. Globally, the residuals show that the performance is correlated with the wind speed similar to the behavior in the wind speed step response simulation.

We have not explicitly modeled the effects of changing r_0 , but this was included in all simulations. Due to the way the phase screens are generated, the r_0 that is specified is a temporal average, which means that there are variations in time. Therefore, we assume that explicitly adding non-stationarity of r_0 will not influence the presented results. The simulations show that the DDSPC controller can handle non-stationary atmospheric turbulence. The second set of simulations shows that even completely random changes in the conditions can be handled. The actual wind speed has more temporal structure,³² therefore, we expect that better performance can be achieved with more realistic wind models.

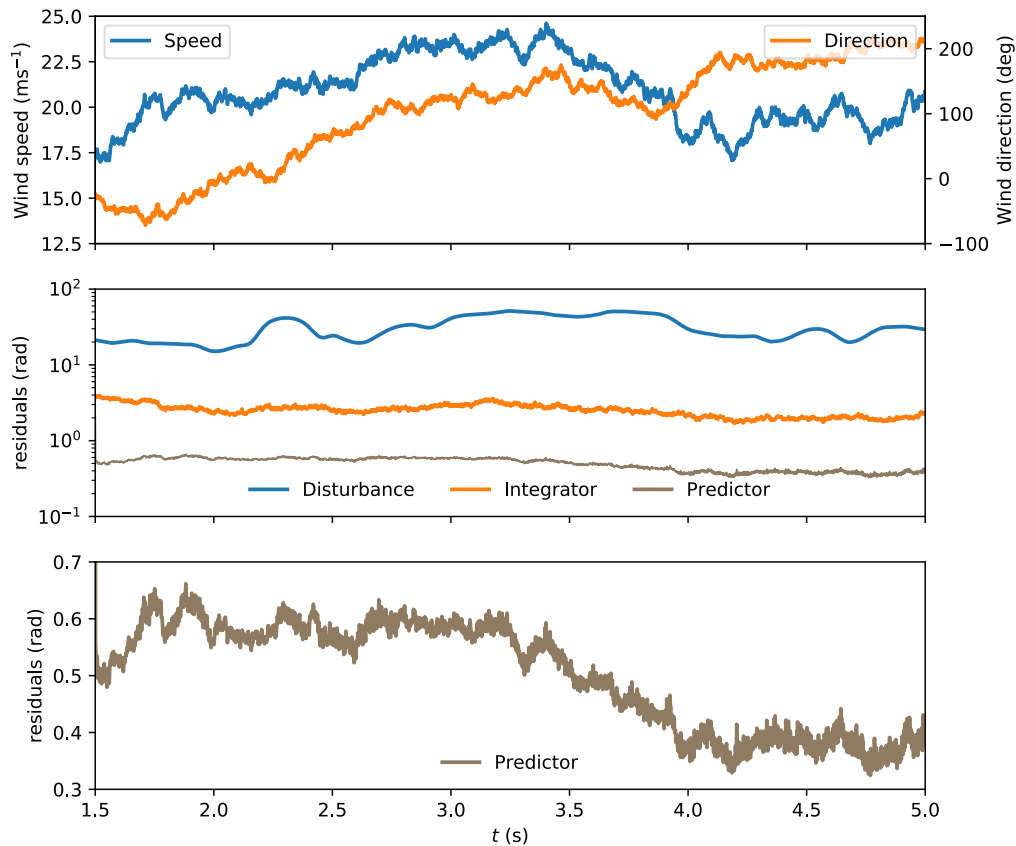


Fig. 11 The performance with a non-stationary disturbance. The top figure shows the wind speed and direction, which are both randomly changed at each time step. The bottom shows the RMS of the modal coefficients as function of time for the input disturbance (blue), the integrator (brown), and the DDSPC (orange). The predictive controller has significant better performance than the integrator at all times. The performance of both the integrator and predictor is correlated with the wind speed.

4 Low-Order Control Verification with MagAO-X

4.1 Setup Description

In this section, we verify the DDSPC algorithm in the lab with MagAO-X. A schematic of MagAO-X can be seen in Fig. 12. MagAO-X uses a woofer-tweeter architecture with an ALPAO-97 DM as woofer and a Boston Micromachines 2K tweeter.^{14,44} The system accepts an $f/11$ beam that first hits the woofer and then the tweeter. This beam is relayed to the lower bench where the pyramid wavefront sensor (PWFS) and the science cameras sit. Similar to previous tests of predictive control, we used the woofer DM as the disturbance creator by running simulated phase screens across the DM.²⁹ In all our tests, we use the PWFS without modulation and with a disturbance outside of the linear regime to show that non-linearity is not a problem for the proposed algorithm. The main advantage of the DDSPC algorithm is that it operates in closed-loop. In closed-loop, the residual wavefront errors are small and within the linear range of the PWFS. We monitor the PSF with the CAMTIP camera, which monitors the (modulated) PSF that hits the pyramid prism. All tests were done with an i' filter that transmits the wavelength range from 700 to 820 nm. The internal light source of MagAO-X creates a brightness equivalent to a 0th magnitude star in the i' filter.

The current implementation of the control algorithm is written in Python, which limited the number of actuators that we could control and the loop speed at which we could run. Therefore, we tested the algorithm at a frame speed of 200 Hz with the 97-actuator woofer DM. To calibrate the response of wavefront to the woofer's actuators, we applied 2000 random patterns on the DM

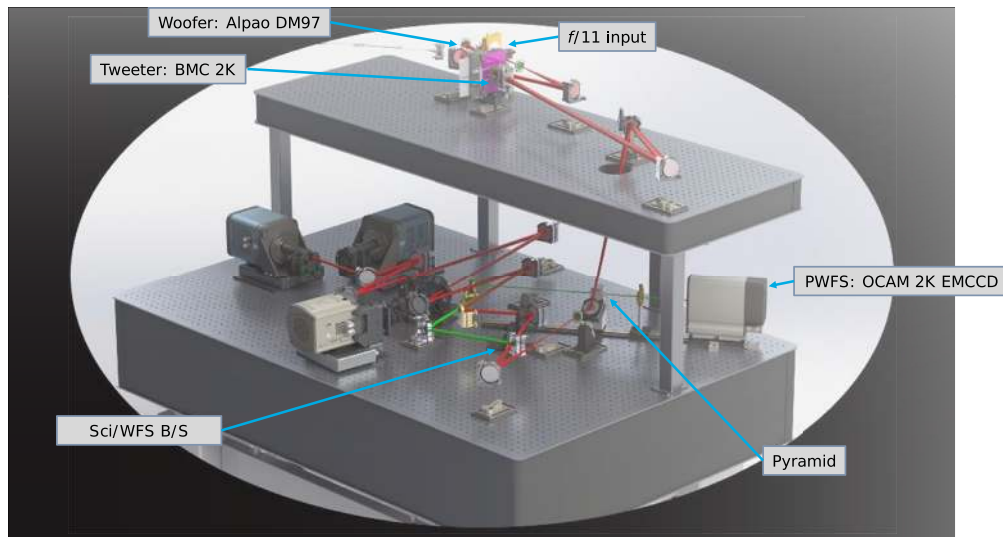


Fig. 12 A solid works render of MagAO-X. On the top bench, the beam enters at $f/11$ and then passes the woofer DM and the tweeter DM. After the periscope, the beam is split into a science path and a wavefront sensor path (green rays). The green rays hit the pyramid prism and the subsequent pupils are imaged onto the OCAM 2K. Next to the pyramid prism is the CAMTIP camera, which is not shown in the figure that monitors the PSF that hits the tip of the pyramid prism.

with a surface RMS of 40 nm, which becomes 80 nm in total reflected wavefront RMS. This data set was then divided into a training part (1500 samples) and a validation part (500 samples). From the training data, we derived the reconstruction matrix as

$$M = V_T S_T^T (S_T S_T^T + \alpha I)^{-1}. \quad (16)$$

With M the reconstruction matrix, V_T the applied commands of the training data, S_T the measured slopes of the training data, and α is the regularization parameter. The optimal regularization parameter is chosen as the α which minimizes the residuals of the reconstructed coefficients in the validation dataset,

$$\arg \min_{\alpha} |V_V - M(\alpha) S_V|^2. \quad (17)$$

Here, V_V and S_V are the applied commands and measured slopes of the validation data set. After finding the optimal regularization parameter, we removed the actuators that were not fully illuminated, which reduced the number of actuators that were controlled from 97 to 86. For the illuminated actuators, the reconstruction residuals were below 10%.

The predictive controllers are initially trained by applying a RBS of 2000 samples in time, with an individual random series for each actuator. Because the controller does not know anything about the system, we started the controller with a strong regularization of $\lambda = 10$. This heavily penalizes the system for any control command, which is desirable if nothing is known yet. After the first round of RBS, we lowered the regularization parameter by a factor 100 and ran another RBS. And finally, we reduced λ again by another factor of 100 and did the last training iteration. The final value of $\lambda = 0.001$ was used for the rest of the tests.

4.2 Results

A series of Kolmogorov phase screens generated with HCIPy³⁹ that evolved according to frozen flow were used to mimic atmospheric disturbances. The phase screens were created with a $r_0 = 0.16$ m and $L_0 = 40$ m. The PSF was recorded by the CAMTIP camera at 30 Hz while the phase screens ran across the DM. Examples of the instantaneous PSFs for a wind speed of 15 ms^{-1} can

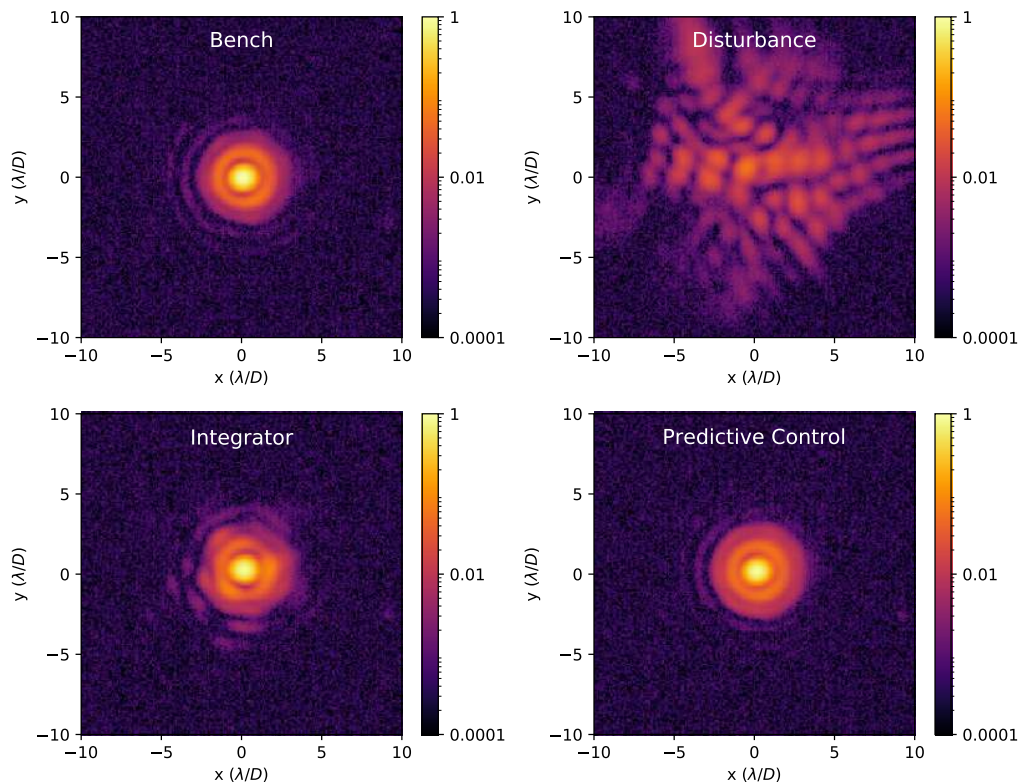


Fig. 13 The PSF during the lab experiments when running a 15 ms^{-1} wind across the DM. The top left figure shows the PSF of the system at rest, without any disturbance or control. There is a visible defocus of the PSF. The top right is a snapshot during the replay of the generated atmospheric phase screens. The bottom left shows the instantaneous PSF with an integral controller with a gain of 0.3. The integrator PSF still has distortions of the Airy rings. The bottom right shows the predictive control PSF which has no visible distortions. A video of the PSFs is available online ([Video 1](https://doi.org/10.1117/1.JATIS.7.2.029001.1), MP4, 11 Mb [URL: <https://doi.org/10.1117/1.JATIS.7.2.029001.1>]).

be seen in Fig. 13. There is a defocus visible in the PSFs because the CAMTIP camera is slightly out of focus. The initial PSF without any correction is heavily distorted and well outside of the PWFS linear regime. The non-linearity is an important aspect to test for predictive control because previous data-driven methods relied on the open-loop reconstruction of the wavefront^{16,29} and the non-linearities will introduce strong model errors due to saturation or optical gain variations. Therefore, it is important to test whether the proposed DDSPC will be influenced or not by non-linearity. Looking at the PSF of the integrator shows that there are residual aberrations, while the PSF of the DDSPC looks perfect by eye. This first image already demonstrates the performance boost of DDSPC, even with the non-linearity of the PWFS included.

Figure 14 shows the performance of both the integrator and DDSPC under a range of wind speeds. The integrator's declining performance with increased wind speed follows the expectation from frozen flow. On the other hand, the performance of the DDSPC seems to be independent of the wind speed. Another major benefit is also apparent in the stability of the Strehl, the DDSPC has a significant smaller spread than the integrator. This spread directly translates into PSF stability, and a more stable PSF will lead to an increased sensitivity to faint companions after PSF subtraction techniques. However, Fig. 14 also shows that the Strehl of the integrator is sometimes higher than that of the predictive controller. We think that this is due to the focus offset of the CAMTIP camera. The DDSPC creates a well-corrected wavefront on the PWFS and due to the focus NCPA, the CAMTIP PSF will not have the highest possible Strehl. The integrator is not able to reach this performance of control and will have some leftover defocus that can compensate for the focus NCPA. Therefore, this creates a situation where a less-performing controller can reach a higher Strehl.

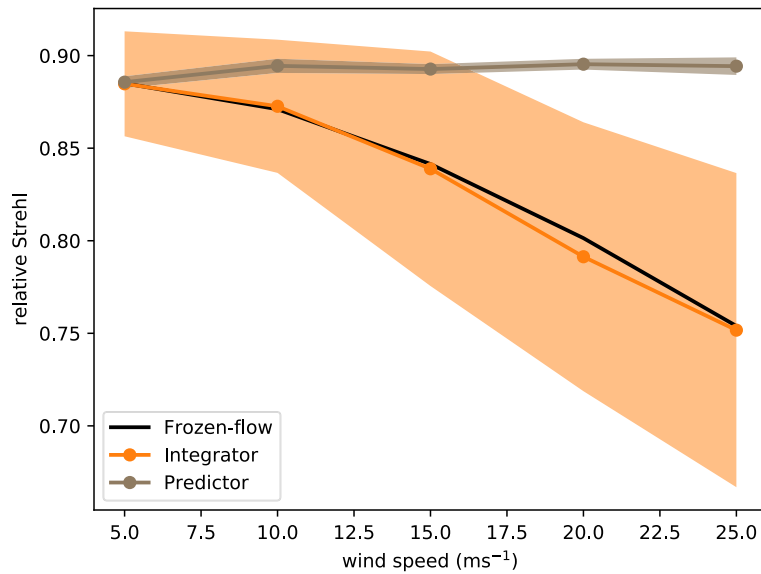


Fig. 14 Strehl of the measured PSFs relative to the bench PSF versus wind speed. The shaded area around each curve is the 16th and 84th percentile of the Strehl in time and corresponds to the temporal stability of the PSF. The median performance of the integrator follows the theoretical Strehl according to frozen-flow. The Strehl of the predictive controller has a smaller spread than the integrator and is independent of wind speed.

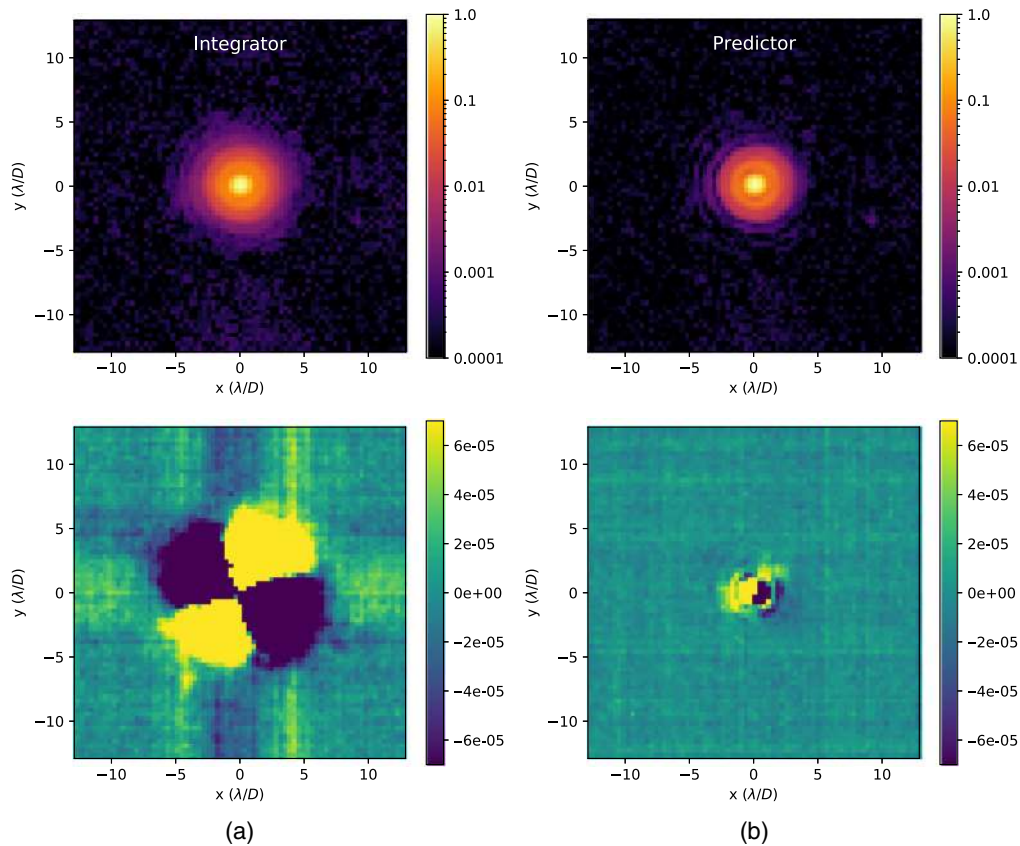


Fig. 15 (a) The measured PSFs and (b) post-processed residuals for a wind speed of 25 ms⁻¹. The left column shows the results for the integrator while the right shows the results for the predictor. The control region of the DM extends to 5.5λ. The integrator shows a strong wind-driven halo, while the halo is gone with the DDSPC.

A better performance metric would be the post-processed contrast that can be reached. At the time of the experiments, MagAO-X did not have coronagraphs installed, so it was not possible to directly measure the raw contrast. To still estimate the gain in contrast between the two controllers, we created two data sets for each wind speed. The first data set has a wind moving horizontally, while the second has a wind moving vertically. The temporal error of the integrator will create a wind-driven halo that is oriented in the direction of the wind (e.g., see Fig. 6). Because the wind is rotated between the exposures, the wind driven halo also rotates. This creates a typical butterfly pattern that decrease the post-processed contrast.⁴⁵ The difference between the two data sets will remove the static diffraction pattern and show the residuals due to the time lag. An example of the post-processed focal plane is shown in Fig. 15 where we show the residuals for a wind speed of 25 ms^{-1} . For both wind directions, we stack all PSFs in time (10 s of total integration time) to create a single PSF per wind direction. These two PSFs are then combined to estimate the average PSF. The diffraction pattern is removed by taking the difference between the PSFs of the two wind directions.

To quantify the contrast gain, we determined the post-processed contrast curve for several wind speeds with aperture photometry. An aperture diameter of $1.4 \lambda/D$ was used. The contrast curves are shown in Fig. 16 for three different wind speeds, 5, 15, and 25 ms^{-1} . The mean wind speed at the Las Campanas Observatory site is 18 ms^{-1} ,¹⁴ which means that the three chosen wind speeds are a good range for the actual on-sky conditions. The contrast of the integrator degrades as the wind speed increases, which is similar to the simulated behavior. The DDSPC however has close to no dependence on the wind speed. In all cases, the DDSPC has superior performance. For the low speed (5 ms^{-1}), the predictor gains a factor of 10 to 20 in contrast. However, for the higher speeds, there is a gain of more than a factor of 100, which was the prediction for predictive control in earlier work.^{16–18} These results also show a much higher gain in contrast than earlier tests that only showed a moderate improvement of a factor of 2.²⁹ Our lab results clearly show the benefit of predictive control for HCI.

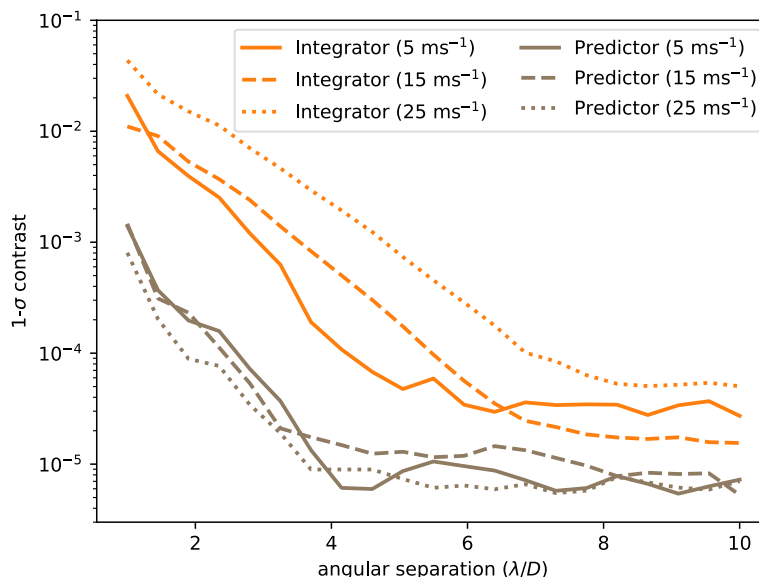


Fig. 16 The contrast curve for different wind speeds. The integrator is shown in orange, while the predictor is shown in brown. The different line styles correspond to different wind speeds. The contrast curve for both controllers flattens out at the larger angular separations due to detector noise. In general, we see a much deeper contrast for the DDSPC than for the integrator. The achieved contrast of DDSPC is almost independent of the wind speed, similar to the behavior of the Strehl. For the integrator, we see a clear degradation of the contrast with increasing wind speed.

5 Discussion and Conclusion

In this paper, we presented the DDSPC for AO for HCI. The method is completely data-driven; no *a priori* knowledge of the system is necessary, and it learns while operating in closed-loop. To lower the computational burden, we used a distributed implementation of DDSPC where we control each mode independently. The hyperparameters of the controller have been explored for various power-law disturbances. This showed that small models result in the lowest closed-loop residuals and that regularization of the controller is only necessary when shallow power laws (or noise) are the dominant input. The stability of the controller has been verified numerically for a wide range of power laws and hyper parameters. This gives us confidence that the algorithm is stable in most situations.

The controller was tested both in an end-to-end AO simulation with stationary and non-stationary turbulence and shows an improvement of more than two order of magnitude in contrast. To verify that the algorithm can work in real systems, we used it to control the woofer of MagAO-X with an PWFS. On the bench, we have achieved a contrast improvement of a factor 10 to 20 for low wind speeds, while we gained more than two orders of magnitude for high wind speeds. This was done in combination with an unmodulated PWFS, showing that the DDSPC can work even when strong non-linearities are involved.

Several steps need to be taken to move from the bench tests to the on-sky with the DDSPC algorithm on MagAO-X. The most important step is to completely port the Python implementation to a lower-level programming language. Because the algorithm mainly consists of matrix-vector multiplications and matrix inversions, a major speed up is expected if we run the algorithm on the GPU. The first speed and timing tests with a GPU implementation of the DDSPC have already shown that it is possible to control 1600 modes faster than 1.5 kHz in double precision and faster than 3 kHz in single precision. Because of the distributed nature of the algorithm, we expect that it is trivial to switch from a single GPU to multiple GPUs, which would allow us to run DDSPC for the ExAO systems of the future GSMTs on current hardware.

The major focus in this work has been on the data-driven control to suppress the servo-lag error. The current controller does not explicitly take into account that the measurements contain noise, however, it can handle noisy measurements because all our lab measurements contained noise. The exact behavior of the algorithm with noisy measurements has not been looked at. The proposed cost function minimizes the future predicted residuals, which would be the correct cost function for deterministic systems. For a stochastic system with unequal variance on the future predictions [Eq. (10)], this does not create the optimal controller. For the optimal estimator, each measurement will need to be weighted by its inverse variance, leading to a modification of Eq. (15). It has been shown that the DDSPC can converge to the optimal Linear-quadratic-Gaussian (LQG) controller for an infinite prediction horizon.³³ For this, two steps are necessary: the first step estimates the prediction matrices, while the second step applies a filter based on the singular values of the prediction matrix. This second step creates the optimal Kalman filter for the future states. Our implementation with the RLS does not apply this second filter step and retains all temporal modes. Adding such a step may increase the robustness of the DDSPC controller against measurement noise. Finally, we could add spatial coupling back into the control matrices. We originally chose to remove those for computational reasons, but the addition of some spatial coupling may help to reduce the effects of noise. All three modifications will lead to increased computational demands, so for future work we will need to balance the complexity of our model against available computational resources.

An on-sky demonstration of the DDSPC with MagAO-X is planned in the near future, which will be a step toward detecting Earth-like planets around nearby stars.

Acknowledgments

Support for this work was provided by NASA through the NASA Hubble Fellowship grant #HST-HF2-51436.001-A awarded by the Space Telescope Science Institute, which is operated by the Association of Universities for Research in Astronomy, Incorporated, under NASA contract NAS5-26555. MagAO-X is funded by the NSF MRI program, award #1625441. The authors declare that they have no conflict of interest.

References

1. W. A. Traub and B. R. Oppenheimer, *Direct Imaging of Exoplanets*, pp. 111–156, University of Arizona Press, Tucson (2010).
2. O. Guyon, “Extreme adaptive optics,” *Annu. Rev. Astron. Astrophys.* **56**, 315–355 (2018).
3. B. Macintosh et al., “First light of the Gemini Planet Imager,” *Proc. Natl. Acad. Sci. U.S.A.* **111**, 12661–12666 (2014).
4. J. L. Beuzit et al., “SPHERE: the exoplanet imager for the very large telescope,” *Astron. Astrophys.* **631**, A155 (2019).
5. N. Jovanovic et al., “The Subaru coronagraphic extreme adaptive optics system: enabling high-contrast imaging on solar-system scales,” *Publ. Astron. Soc. Pac.* **127**, 890 (2015).
6. M. S. Marley et al., “On the luminosity of young Jupiters,” *Astrophys. J.* **655**, 541–549 (2007).
7. B. P. Bowler et al., “Planets around Low-mass Stars (PALMS). IV. The Outer Architecture of M Dwarf Planetary Systems,” *Astrophys. J. Suppl. Ser.* **216**, 7 (2015).
8. E. L. Nielsen et al., “The Gemini planet imager exoplanet survey: giant planet and brown Dwarf demographics from 10 to 100 au,” *Astronomical Journal* **158**, 13 (2019).
9. R. B. Fernandes et al., “Hints for a turnover at the snow line in the giant planet occurrence rate,” *Astrophys. J.* **874**, 81 (2019).
10. K. Wagner, D. Apai, and K. M. Kratter, “On the mass function, multiplicity, and origins of wide-orbit giant planets,” *Astrophys. J.* **877**, 46 (2019).
11. M. Kasper, “Adaptive optics for high contrast imaging,” *Proc. SPIE* **8447**, 84470B (2012).
12. J. Milli et al., “Performance of the extreme-AO instrument VLT/SPHERE and dependence on the atmospheric conditions,” arXiv:1710.05417 (2017).
13. F. Cantalloube et al., “High contrast imaging with ELT/METIS: The wind driven halo, from SPHERE to METIS,” arXiv:1911.11241 (2019).
14. J. R. Males et al., “MagAO-X: project status and first laboratory results,” *Proc. SPIE* **10703**, 1070309 (2018).
15. D. T. Gavel and D. Wiberg, “Toward Strehl-optimizing adaptive optics controllers,” *Proc. SPIE* **4839**, 890–901 (2003).
16. O. Guyon and J. Males, “Adaptive optics predictive control with empirical orthogonal functions (EOFs),” arXiv:1707.00570 (2017).
17. J. R. Males and O. Guyon, “Ground-based adaptive optics coronagraphic performance under closed-loop predictive control,” *J. Astron. Telesc. Instrum. Syst.* **4**, 019001 (2018).
18. C. M. Correia et al., “Performance limits of adaptive-optics/high-contrast imagers with pyramid wavefront sensors,” *Mon. Not. R. Astron. Soc.* **495**, 4380–4391 (2020).
19. C. Dessenne, P.-Y. Madec, and G. Rousset, “Modal prediction for closed-loop adaptive optics,” *Opt. Lett.* **22**, 1535–1537 (1997).
20. C. Dessenne, P.-Y. Madec, and G. Rousset, “Sky implementation of modal predictive control in adaptive optics,” *Opt. Lett.* **24**, 339–341 (1999).
21. 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**, 2645 (2007).
22. L. A. Poyneer and J.-P. Véran, “Kalman filtering to suppress spurious signals in adaptive optics control,” *J. Opt. Soc. Am. A* **27**, A223 (2010).
23. C. Kulcsár et al., “Optimal control, observers and integrators in adaptive optics,” *Opt. Express* **14**, 7464–7476 (2006).
24. K. Hinnen, M. Verhaegen, and N. Doelman, “H₂-optimal control of an adaptive optics system: part I, data-driven modeling of wavefront disturbance,” *Proc. SPIE* **5903**, 590309 (2005).
25. Z. Xu et al., “Deep learning control model for adaptive optics systems,” *Appl. Opt.* **58**(8), 1998–2009 (2019).
26. S. L. S. Gómez et al., “Experience with artificial neural networks applied in multi-object adaptive optics,” *Publ. Astron. Soc. Pac.* **131**(1004), 108012 (2019).
27. G. Sivo et al., “First on-sky SCAO validation of full LQG control with vibration mitigation on the CANARY pathfinder,” *Opt. Express* **22**, 23565–23591 (2014).

28. J. Tesch et al., “On-sky demonstration of optimal control for adaptive optics at Palomar Observatory,” *Opt. Lett.* **40**, 1575 (2015).
29. R. Jensen-Clem et al., “Demonstrating predictive wavefront control with the Keck II near-infrared pyramid wavefront sensor,” *Proc. SPIE* **11117**, 111170W (2019).
30. R. Ragazzoni, “Pupil plane wavefront sensing with an oscillating prism,” *J. Mod. Opt.* **43**, 289–293 (1996).
31. V. Deo et al., “A telescope-ready approach for modal compensation of pyramid wavefront sensor optical gain,” *Astron. Astrophys.* **629**, A107 (2019).
32. 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**, 731 (2019).
33. W. Favoreel et al., “Model-free subspace-based LQG-design,” in *Proc. Am. Control Conf.*, Vol. 5, pp. 3372–3376 (1999).
34. B. Huang and R. Kadali, *Dynamic Modeling, Predictive Control and Performance Monitoring: A Data-Driven Subspace Approach*, Springer, London (2008).
35. C. M. Correia et al., “Modeling astronomical adaptive optics performance with temporally filtered Wiener reconstruction of slope data,” *J. Opt. Soc. Am. A* **34**, 1877 (2017).
36. M. A. M. van Kooten, N. Doelman, and M. Kenworthy, “Robustness of prediction for extreme adaptive optics systems under various observing conditions. An analysis using VLT/SPHERE adaptive optics data,” *Astron. Astrophys.* **636**, A81 (2020).
37. J. Berberich et al., “Data-driven model predictive control with stability and robustness guarantees,” *IEEE Trans. Autom. Control* (2020).
38. J. Coulson, J. Lygeros, and F. Dörfler, “Distributionally robust chance constrained data-enabled predictive control,” arXiv:2006.01702 (2020).
39. E. H. Por et al., “High contrast imaging for Python (HCIPy): an open-source adaptive optics and coronagraph simulator,” *Proc. SPIE* **10703**, 1070342 (2018).
40. G. I. Taylor, “The spectrum of turbulence,” *Proc. R. Soc. London Ser. A* **164**, 476–490 (1938).
41. C. Cavarroc et al., “Fundamental limitations on Earth-like planet detection with extremely large telescopes,” *Astron. Astrophys.* **447**, 397–403 (2006).
42. O. Guyon et al., “Theoretical limits on extrasolar terrestrial planet detection with coronagraphs,” *Astrophys. J. Suppl. Ser.* **167**, 81–99 (2006).
43. N. Doelman, “The minimum of the time-delay wavefront error in adaptive optics,” *Mon. Not. R. Astron. Soc.* **491**, 4719–4723 (2020).
44. L. M. Close et al., “Optical and mechanical design of the extreme AO coronagraphic instrument MagAO-X,” *Proc. SPIE* **10703**, 107034Y (2018).
45. F. Cantalloube et al., “Wind-driven halo in high-contrast images. I. Analysis of the focal-plane images of SPHERE,” *Astron. Astrophys.* **638**, A98 (2020).

Sebastiaan Y. Haffert is a NASA Hubble postdoctoral fellow at the University of Arizona’s Steward Observatory. His research focuses on high-spatial and high-spectral resolution instrumentation for exoplanet characterization.

Jared R. Males is an assistant astronomer in the University of Arizona’s Steward Observatory and is the principal investigator of MagAO-X. His research is focused on using HCI to study extrasolar planets.

Joseph D. Long is a PhD student at the University of Arizona’s Steward Observatory and a member of the MagAO-X instrument team researching exoplanet observation post-processing techniques.

Biographies of the other authors are not available.