# Reinforcement Learning based control law on PAPYRUS: simulations using different atmospheric conditions

Raissa Camelo[a], Jalo Nousiainen[b,c], Cedric Taïssir Heritier[a,d], Morgan Gray[a], and Benoit Neichel[a]

[a]Aix-Marseille Université, CNRS, CNES, LAM, Marseille, France
[b]Department of Computational and Process Engineering, Lappeenranta–Lahti University of Technology, Finland
[c]European Southern Observatory, Karl-Schwarzschild-Str. 2, 85748 Garching bei München, Germany
[d]DOTA, ONERA, F-13661 Salon cedex Air - France

## ABSTRACT

Predictive control laws for Adaptive Optics (AO) using Artificial Intelligence has been recently explored as an alternative to the classic methods, such as the integrator law. Reinforcement Learning excels in predictive control tasks by enabling systems to learn optimal control strategies through continuous interaction with their environment, adapting to dynamic conditions and achieving effective decision-making in real-time. In our previous work, a Model-based Reinforcement Learning (MBRL) method called Policy Optimization for Adaptive Optics (PO4AO) was used in conjunction with the Object-Oriented Python Adaptive Optics (OOPAO) to simulate the Provence Adaptive Optics Pyramid Run System (PAPYRUS) optical bench. PO4AO demonstrated high adaptability to turbulence and rapid convergence, achieving optimal corrections after just 500 frames of interaction, outperforming a simulated integrator in different atmospheric conditions. Building upon this, our current study explored PO4AO's capability to adapt to sudden atmospheric changes by worsening turbulence conditions during evaluation, notably the wind speed and the seeing. In the result's section, We compare PO4AO's performance in terms of Strehl Ratio (SR) to the integrator. Further description of the experiments are present in the paper.

**Keywords:** Adaptive Optics,Machine Learning, Predictive Control, Reinforcement Learning

## 1. INTRODUCTION

Observations made by ground-based telescopes suffer significantly from atmospheric turbulence, which distorts the phase of incoming light, resulting in blurred and contorted images. Adaptive Optics (AO) systems have emerged as a solution to mitigate these effects, enabling us to correct wavefront aberrations and improve observation quality. Traditionally, an AO system consists of three primary components: a wavefront sensor (WFS) to measure atmospheric-induced phase aberrations, a Real Time Computer (RTC) that calculates the corrections for the aberrations and a deformable mirror (DM) that applies said corrections. These systems can operate in a open-loop configuration, where the WFS assesses wavefront measurements before the DM corrections, or in a closed-loop configuration, where the WFS assesses wavefront distortions post-DM correction. All the simulations done in this work where closed-loop.

While traditional control algorithms have proven to be effective, they often rely on predefined models of atmospheric turbulence and can be hindered by errors in the WFS such as photon noise.[1–3] Recently, Reinforcement Learning (RL) has shown promise in AO systems by predicting the evolution of turbulence over time and adjusting for common modeling errors.[4] Beyond RL, advancements in AO techniques have also explored the application of artificial intelligence to enhance other tasks related to wavefront sensing.[5–7] AI is continuously being employed across multiple fields of research and the AO community can also benefit from its assistance.

Further author information:
R. Camelo : E-mail: raissa.camelo@lam.fr,
J. Nousiainen.: E-mail: jalo.nousiainen@lut.fi

Among the various Reinforcement Learning (RL) techniques applied to adaptive optics, Nousiainen et al.'s PO4AO[8](Policy Optimization for Adaptive Optics) stands out as a model-based policy optimization algorithm. This method estimates the control voltages necessary for the Deformable Mirror (DM) based on data from the wavefront sensor (WFS). Extensive numerical simulations and experiments[8,9] have demonstrated its promising performance, paving the way for potential on-sky applications. This study consists on the first steps into applying RL techniques on-sky using the PAPYRUS bench at Observatoire Haute de Provence (OHP).To prepare for on-sky tests, we first implement and validate the algorithm through numerical simulations that emulates the PAPYRUS configuration. This approach allows us to thoroughly understand the RL algorithm's interaction with the bench and provides flexibility to experiment with different optimizations before initiating the on-sky development.

The PO4AO algorithm has been noted for its ability to reduce wavefront error and its adaptability in on-sky conditions.[10,11] Furthermore, as AI-driven approaches in AO continue to grow, the demand for robust and compatible simulation platforms has increased. Between other AO tools, the Object-Oriented Python Adaptive Optics (OOPAO)[12] emerges as a valuable tool for end-to-end AO simulations due to its open-source nature and Python implementation. In this paper, we showcase OOPAO's suitability as an experimental environment for testing RL methods in AO.

## 2. PREDICTIVE CONTROL FOR AO: REINFORCEMENT LEARNING

Reinforcement Learning (RL)[13] belongs to the realm of machine learning that focuses on how intelligent algorithmic agents make decisions to influence their environment, aiming to maximize cumulative rewards. Unlike traditional supervised learning, where models learn from labeled input-output pairs, RL operates without pre-existing labeled data. Instead, RL agents learn through trial and error, optimizing their behavior by maximizing the rewards obtained for each action taken. The specific reward function is tailored to the problem at hand.

RL is particularly suited for tasks requiring real-time decision-making because it can be trained "online" as it actively solves the problem. These problems are typically formulated using a Markov Decision Process (MDP), a mathematical framework that defines states, actions, transition probabilities, and rewards. In an MDP, the state represents the current environment status, actions are choices made by the agent, transition probabilities determine the likelihood of moving between states based on actions, and rewards quantify the agent's performance.

During each time step, an agent selects an action from the available set, which alters the environment state according to defined probabilities. The agent then receives a reward based on this action. The final goal for the agent is to learn an optimal policy, a strategy or function that maps states to actions, maximizing cumulative rewards through effective interaction with the environment.

Next, we outline the MDP framework for Adaptive Optics (AO) as detailed by Nousiainen et al.[11] Various MDP formulations exist for AO control (see, e.g.,[14–16]), but this paper focuses on the approach used in PO4AO.

We denote the control voltages applied to a DM at a given time instant $t$ as $v_t$ and the WFS measurements, pre-processed to slopes as $w_t$. As such, an action effectuated at time step $t$ is defined as the differential control voltages applied to the DM at that instant $t$:

$$a_t = \Delta v_t, \tag{1}$$

while the the full control voltages are $\Delta v_t + v_{t-1}$. At each time step $t$, the WFS measurement $w_t$ is observed. The measurements are projected into voltage space by operating the reconstruction matrix, denoted as $C$. Since the AO system in our simulations is a closed loop, it corresponds to the residual voltages detected by the WFS. We define an observation (of the state) at time instant $t$ as

$$o_t = Cw_t. \tag{2}$$

To ensure the Markovian property, each state $S_t$ is represented by a concatenation of previous observations and actions, that is,

$$s_t = (o_t, o_{t-1}, .., o_{t-k}, a_{t-1}, a_{t-2}, ..., a_{t-m}), \tag{3}$$

where $k = m$, including data from the previous $m$ time steps and the reconstruction matrix $C$.

The reward function is defined as the residual voltages' negative squared norm as stated below:

$$r_t(s_{t+1}, s_t, a_t) = -1 * ||o_{t+1}||^2. \tag{4}$$

In the following section we briefly explain the PO4AO algorithm and its implementation.

## 3. PO4AO: POLICY OPTIMIZATION FOR ADAPTIVE OPTICS

Reinforcement Learning algorithms typically fall into two main categories: model-based and model-free approaches.[17] In model-based Reinforcement Learning, the agent constructs an explicit model of the environment, known as the dynamics model, to approximate how the environment transitions between states (e.g., controlling a DM). This model is essential for tasks such as model predictive control or optimizing a policy model, which determines the agent's actions based on the environment's dynamics (e.g., AO system, bench, WFS). In contrast, model-free Reinforcement Learning learns directly from interactions with the environment, deriving a policy without explicitly modeling its dynamics.[14,15]

The Policy Optimization for Adaptive Optics (PO4AO) algorithm, detailed by J. Nousiainen et al.,[8] exemplifies a model-based approach in RL. PO4AO employs two neural networks: a dynamics model and a policy. The dynamics model, implemented as a neural network, learns to predict the next state given the current state and action, $Dyn(s_t, a_t) = s'_{t+1}$, using previously stored state-action pairs from simulations. The policy network maps current states $s_t$ to actions $a_t$ to be executed by the DM, $\pi(s_t) = a'_t$. Both neural networks are compact convolutional models, each composed of three dense layers.

To train these models, data from simulations is collected into a dataset $\mathcal{D} = \{(s_t^i, a_t^i)s_{t+1}^i\}_{i=1}^N$, where each entry consists of a state-action pair and its subsequent state. The dynamics model $Dyn(s_t, a_t)$ undergoes supervised learning using $\mathcal{D}$ to predict $s'_{t+1}$ accurately. Later, the policy model $\pi(s_t)$ is optimized by using the dynamics model: starting from a state $s_t$ in $\mathcal{D}$, $\pi(s_t)$ predicts an action $a'_t$, which is then fed into $Dyn$ along with $s_t$ to predict $s_{t+1}$. This iterative process repeats over a fixed number of time steps $T$ (planning horizon) within each episode, gathering rewards that guide the policy's optimization through back-propagation.

In our implementation of PO4AO, each episode comprises 500 simulation frames, with a total of 20 episodes conducted. Initially, during the warm-up phase, the simulation runs without policy guidance, using random DM voltages to gather $\mathcal{D}$. After warm-up, the policy begins guiding actions, continually updating $\mathcal{D}$ and refining both the dynamics and policy models. For instance, we trained the policy over 60 instances during warm-up and 7 instances in subsequent rounds, parameters chosen based on experimental results.

## 4. RESULTS

In this section we describe the parameters used for the PAPYRUS simulation using OOPAO. We also describe the experiments realised with the atmospheric turbulence and display the results obtained. The experiments mainly consisted on changing the r0 and the Wind Speed (WS) values during each simulation. We elaborated three turbulence profiles for our experiments: increasing WS, decreasing r0 and "chaotic" WS, where we change the Wind Speed values randomly every second. The range of values used for each simulation profile are shown in the images 1 and 2. The chaotic WS profile employs the same range of values as the increasing WS profile, except that the values where "shuffled", randomizing the list of WS parameters.

### 4.1 OOPAO: PAPYRUS Simulation Parameters

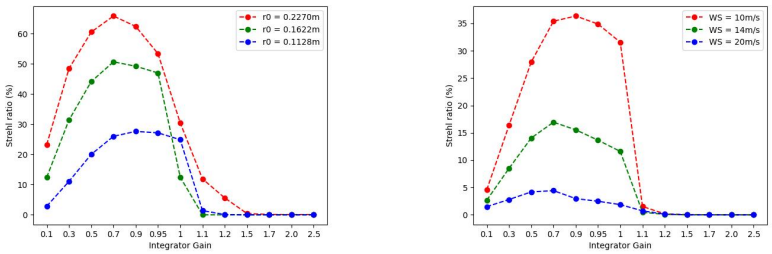
In order to simulate the PAPYRUS system we used OOPAO for both the bench and telescope T152, located at OHP. The simulation parameters for the optical system are provided in Table 1. The simulated atmospheric turbulence is presented as a sum of five frozen flow layers with Von Karman power spectra. The same system configuration was used for all the simulations, with PO4AO and the integrator. The number of Zernike modes was kept at 50 based on the results obtained in our previous work,[9] where we demonstrated that even using a few modes the RL was able to outperform the integrator. We did not optimise the number of modes or other AO simulation parameters besides optical gain as we restricted ourselves to analysing the behaviour of both RL and integrator under changing atmospheric conditions.

Table 1: PAPYRUS Simulation parameters for OOPAO

| | |
|---|---|
| N°Actuators | 17*17 |
| Telescope Diameter | 1.52m (OHP) |
| Sampling time | 2ms (500 Hz) |
| Delay (PO4AO) | 1 Frame |
| WFS | Pyramid (Modulated) |
| WFS Modulation | 3 $\lambda$/D |
| Modal Basis | Zernike (50 modes) |
| DM Mechanical Coupling | 35% |
| Wind Speed* | 10 (m/s) / layer |
| Magnitude (NGS) | 8 |
| Sensing Wavelength | "I" (806 nm) |

*

In the images bellow we show the results obtained in the simulations described in the previously. First, we run the atmospheric simulations on the integrator under different gain values, in order to optimize the integrator and select the best configuration to compare to the results obtained with PO4AO. We chose the decreasing r0 and the increasing WS profiles for the first simulation. We tested those profiles on optical gain values ranging from 0.1 to 2.5. In order to provide a more comprehensive view of the results, we opted to display the Strehl Ratio for three specific values of r0 and WS for the decreasing r0 and the increasing WS simulations, respectively. The results are shown in Figure 1.




(a) WS = 10m/s, Decreasing r0 @500nm

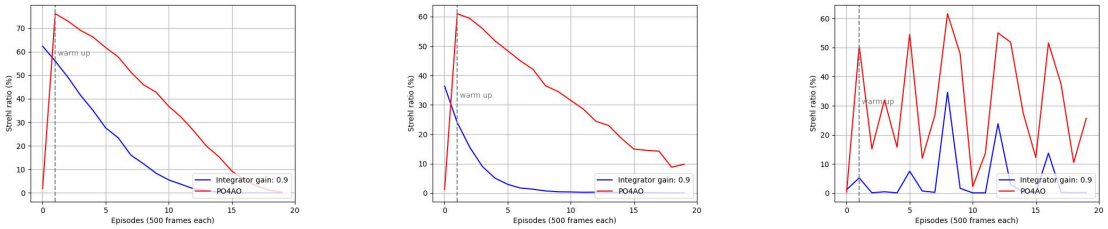
(b) Increasing WS , r0 = 0.13m @500nm

Figure 1: Integrator simulations: Strehl ratio for each gain in 10.000 frames

As seen in both simulations, the optimal integrator gain was between 0.7 and 0.9 for all r0 and WS values displayed (see Figure 1), with 0.9 being more frequently the best. The gain values bigger than 1 were unstable. We chose the gain value of 0.9 for the comparison to PO4AO (Figure 2). Further, we note here that we aimed to exploit the Pyramid WFS's optical gain by going above 0.5 in an attempt to optimize the integrator for a fair comparison against PO4AO.[15,18] We also note that in our previous work[9] the same value was the most optimal in all simulations, with this work being an extension of the previous one and maintaining the same simulation setup, except for the atmospheric turbulence.

Next we display the results for PO4AO under three different atmospheric profiles: decreasing r0, increasing WS and chaotic WS. In each image we compare PO4AO's performance to the Integrator at optical gain 0.9 over 10.000 frames. Similarly to the previous plots, we change the atmospheric conditions every second of the simulation (every 500 frames). The results are displayed in Figure 2.

*The simulated atmosphere consists of 5 layers of frozen flow with an initial speed of 10m/s each on average




(a) WS = 10m/s, Decreasing r0 from 0.227m @500nm to 0.0237m @500nm

(b) Increasing WS from 10m/s to 50m/s, r0 = 0.13m @500nm

(c) Chaotic WS , r0 = 0.13m @500nm

Figure 2: PO4AO vs Integrator (gain = 0.9): Average Strehl ratio per Episode (500 frames)

## 5. DISCUSSION

PO4AO continues to yield promising results, with the MBRL algorithm consistently outperforming the optimized integrator across all test cases after just one episode of training. Our simulation on the PAPYRUS bench further validated the efficacy of the Reinforcement Learning approach facilitated by OOPAO, allowing us to experiment with various turbulence profiles and gain valuable insights prior to full-scale implementation on PAPYRUS. OOPAO served as a testing ground for RL algorithm before progressing towards on-sky experiments.

The neural network's capability to detect and exploit hidden features allied to the MDP formulation and implementation of the RL technique has shown to improve correction. The PO4AO approach consistently optimizes the residual WFS measurement, and subsequently the SR, even in changing atmospheric conditions. Furthermore, neural networks indicate to be a promising technique for wavefront correction, emphasizing the potential of Reinforcement Learning to enable telescopes to dynamically optimize their corrections for atmospheric turbulence. The ability of being able to adapt to changing conditions through online learning is an major advantage of the RL method.

To refine our simulations, upcoming experiments will focus on enhancing the PAPYRUS model by incorporating physical features that were not present in this version. There is a new more accurate simulation of the PAPYRUS bench using OOPAO currently being developed at LAM that can be used in future experiments. We also aim to improve the neural network performance by fine-tuning the algorithm and through hyperparameter optimization. Additionally, transitioning from frozen flow to boiling simulations[19,20] and addressing misregistration and vibrations will introduce greater realism into our atmospheric turbulence tests. Other AO parameters could also be optimized in future experiments such as the number of Zernike modes used in the simulation, as noted before. There is a lot of room for experimenting with RL control for AO and further comparisons with other alternative methods such as the Kalman filter should be encouraged in the near future. Exploring the creating of hybrid methods, mixing both AI and classic algorithms could also be beneficial and has been investigated before.[21,22]

Finally, we aim to continue this project by proceeding with the mentioned experiments on the simulated bench and then deploying the algorithm on the PAPYRUS, under calibration source and, subsequently, on-sky. In order to enable the transition from simulation to on-sky experiments the software must be adapted and integrated into the PAPYRUS's RTC, which requires some significant changes to the current software interface. We hope to manage these challenges in the near future and further to explore the capabilities of Reinforcement Learning for adaptive optics.

## ACKNOWLEDGMENTS

This work benefited from the support of the the French National Research Agency (ANR) with APPLY (ANR-19-CE31-0011) and LabEx FOCUS (ANR-11-LABX-0013); the Programme Investissement Avenir F-CELT (ANR-21-ESRE-0008), the Action Spécifique Haute Résolution Angulaire (ASHRA) of CNRS/INSU co-funded by CNES, the ECOS-CONYCIT France-Chile cooperation (C20E02), the ORP-H2020 Framework Programme of the European Commission's (Grant number 101004719), STIC AmSud (21-STIC-09), the french government under

the France 2030 investment plan, the Initiative d'Excellence d'Aix-Marseille Université A*MIDEX, program number AMX-22-RE-AB-151. In accordance with SPIE's policy, we acknowledge the use of external AI-based tools, solely for proofreading and improving the grammar of this manuscript.

## REFERENCES

[1] Wong, A. P., Norris, B. R., Tuthill, P. G., Scalzo, R., Lozi, J., Vievard, S., and Guyon, O., "Predictive control for adaptive optics using neural networks," *Journal of Astronomical Telescopes, Instruments, and Systems* **7**(1), 019001–019001 (2021).

[2] Swanson, R., Lamb, M., Correia, C. M., Sivanandam, S., and Kutulakos, K., "Closed loop predictive control of adaptive optics systems with convolutional neural networks," *Monthly Notices of the Royal Astronomical Society* **503**(2), 2944–2954 (2021).

[3] Orban De Xivry, G., Quesnel, M., Vanberg, P., Absil, O., and Louppe, G., "Focal plane wavefront sensing using machine learning: performance of convolutional neural networks compared to fundamental limits," *Monthly Notices of the Royal Astronomical Society* **505**(4), 5702–5713 (2021).

[4] Nousiainen, J., Engler, B., Kasper, M., Helin, T., Heritier, C. T., and Rajani, C., "Advances in model-based reinforcement learning for adaptive optics control," in [*Adaptive Optics Systems VIII*], **12185**, 882–891, SPIE (2022).

[5] Wong, A. P., Norris, B. R., Deo, V., Guyon, O., Tuthill, P. G., Lozi, J., Vievard, S., and Ahn, K., "Machine learning for wavefront sensing," in [*Adaptive Optics Systems VIII*], **12185**, 791–809, SPIE (2022).

[6] Gray, M., Dumont, M., Beltramo-Martin, O., Lambert, J.-C., Neichel, B., and Fusco, T., "Deeploop: Deep learning for an optimized adaptive optics psf estimation," in [*Adaptive Optics Systems VIII*], **12185**, 1022–1030, SPIE (2022).

[7] Fowler, J. and Landman, R., "Tempestas ex machina: a review of machine learning methods for wavefront control," *Techniques and Instrumentation for Detection of Exoplanets XI* **12680**, 100–114 (2023).

[8] Nousiainen, J., Rajani, C., Kasper, M., Helin, T., Haffert, S., Vérinaud, C., Males, J., Van Gorkom, K., Close, L., Long, J., et al., "Toward on-sky adaptive optics control using reinforcement learning-model-based policy optimization for adaptive optics," *Astronomy & Astrophysics* **664**, A71 (2022).

[9] Camelo, R., Nousiainen, J., Heritier, C. T., Morgan, G., and Neichel, B., "Papyrus at ohp: Predictive control with reinforcement learning for improved performance," in [*Adaptive Optics for Extremely Large Telescopes 7th Edition*], (2023).

[10] Nousiainen, J., Rajani, C., Kasper, M., Helin, T., Haffert, S., Vérinaud, C., Males, J., Van Gorkom, K., Close, L., Long, J., et al., "Towards on-sky adaptive optics control using reinforcement learning," *arXiv preprint arXiv:2205.07554* (2022).

[11] Nousiainen, J., Rajani, C., Kasper, M., and Helin, T., "Adaptive optics control using model-based reinforcement learning," *Optics Express* **29**(10), 15327–15344 (2021).

[12] Héritier, C. T. et al., "Object oriented python adaptive optics (oopao)," *AO4ELT-7 proceedings* , https://github.com/cheritier/OOPAO (2023).

[13] Sutton, R. S. and Barto, A. G., [*Reinforcement learning: An introduction*], MIT press (2018).

[14] Landman, R., Haffert, S. Y., Radhakrishnan, V. M., and Keller, C. U., "Self-optimizing adaptive optics control with reinforcement learning for high-contrast imaging," *Journal of Astronomical Telescopes, Instruments, and Systems* **7**(3), 039002–039002 (2021).

[15] Pou, B., Ferreira, F., Quinones, E., Gratadour, D., and Martin, M., "Adaptive optics control with multi-agent model-free reinforcement learning," *Optics express* **30**(2), 2991–3015 (2022).

[16] Pou, B., Smith, J., Quinones, E., Martin, M., and Gratadour, D., "Model-free reinforcement learning with a non-linear reconstructor for closed-loop adaptive optics control with a pyramid wavefront sensor," in [*Adaptive Optics Systems VIII*], **12185**, 945–958, SPIE (2022).

[17] Plaat, A., [*Deep reinforcement learning*], vol. 10, Springer (2022).

[18] Landman, R., Haffert, S. Y., Radhakrishnan, V. M., and Keller, C. U., "Self-optimizing adaptive optics control with reinforcement learning," in [*Adaptive Optics Systems VII*], **11448**, 842–856, SPIE (2020).

[19] Prengère, L., Kulcsár, C., and Raynaud, H.-F., "Zonal-based high-performance control in adaptive optics systems with application to astronomy and satellite tracking," *JOSA A* **37**(7), 1083–1099 (2020).

[20] Berdja, A. and Borgnino, J., "Modelling the optical turbulence boiling and its effect on finite-exposure differential image motion," *Monthly Notices of the Royal Astronomical Society* **378**(3), 1177–1186 (2007).

[21] Nousiainen, J., Engler, B., Kasper, M., Rajani, C., Helin, T., Heritier, C. T., Quanz, S. P., and Glauser, A. M., "Laboratory experiments of model-based reinforcement learning for adaptive optics control," *Journal of Astronomical Telescopes, Instruments, and Systems* **10**(1), 019001–019001 (2024).

[22] Guo, Y., Zhong, L., Min, L., Wang, J., Wu, Y., Chen, K., Wei, K., and Rao, C., "Adaptive optics based on machine learning: a review," *Opto-Electronic Advances* **5**(7), 200082–1 (2022).