Evaluating Adaptive Control of a 6DoF Manipulator

Jason Pettinato
UMass Amherst Computer Science
Amherst, USA
jpettinato@umass.edu

Dejvi Rrapi
UMass Amherst Computer Science
Amherst, USA
email

Gordon Hatcher
UMass Amherst Computer Science
Amherst, USA
ghatcher@umass.edu

***Abstract*—This report reviews works regarding conventional and adaptive methods of control for robotic manipulators. It then goes on to create a PID controller for use with the Frank Emirka Panda Robotic Manipulator Model. This controller is tuned using both the Ziegler-Nichols and Chao Oscillatory Methods to find the proportional gain (K_p), integral gain (K_i), and derivative gain (K_d) for each axis. An Lyapunov-based adaptive controller is also proposed, which utilizes the lyapunov function to continuously tune K_p, K_i, and K_d. The three controllers performances are then compared in a stable and windy enviorment, provided a circular desired trajectory. The Chao tuned PID controller was found to perform worse than both the Ziegler-Nichols conventional tuned controller and Adaptive Controller recieving a performance index of 0.3 compared to the 0.56 and 0.59 of the Ziegler-Nichols and Adaptive controllers respectively. Under stable conditions the Ziegler-Nichols was comprable to Adaptive controller, but once the wind force was applied to the manipulator the conventional Ziegler-Nicholas needed far more controller effort to achieve similar performance to the adaptive controller while still displaying a lower recovery time.**

I. Introduction

Robotic manipulators are among the most widely utilized robotic systems, employing multiple revolute and/or prismatic joints to control the position of a single end effector. These systems are implemented across a variety of industries, including medical applications and manufacturing processes. The task of guiding such robots to their desired positions is both complex and critical, relying on the system's controller to achieve precise motion. The controller must accomplish two essential functions: first, it must compute the trajectory of the end effector from its current to its desired position, and second, it must determine and apply the torques necessary to execute this motion [1]. The approach to solving these problems varies based on the specific controller, which is selected primarily according to the intended function of the robot. The study of controller design and optimization forms the foundation of control theory.

Control theory, a multidisciplinary field rooted in computer science and engineering, explores methodologies for predicting and managing the behavior of dynamic systems. Controllers are often evaluated based on trade-offs between efficiency and performance, and extensive research has been devoted to comparing their effectiveness [1]–[5]. Proportional-Integral-Derivative (PID) controllers are among the most commonly employed due to their simplicity [2] and reliable performance [3]. PIDs rely on predefined parameters to compensate for a system nonlinearities and dynamic behaviors. Conventional tuning of these parameters is often conducted using observational methods such as the Ziegler-Nichols method, employed in studies by Zhang et al. [1], Noventino et al. [3], and Slati et al. [4], or the Chau method, as demonstrated by Amiri et al. [2]. Noventino et al specifcally utilized the Ziegler-Nichols method to tune a PID controller for a 3-DOF manipulator and evaluated its performance. Slati et al. applied the same method to a 6-DOF robotic system finding the proper gains through observation of their system in oscillation. They found that their system wasn't capable of achieving a steady state condition despite optimal tuning [4]. This conveys one of the short comings of PID controllers as, despite their effectiveness, conventional PID controllers cannot compensate for nonlinearities within more complex systems, quickly losing accuracy in unpredictable scenarios. These more complex systems produce the need for Adaptive Controllers.

Adaptive controllers dynamically adjust system parameters to maintain optimal control [1]. This adaptability can be achieved through various approaches, each with unique advantages. Zhang et al. reviewed the use of Model Reference Adaptive Controllers (MRACs), highlighting their ability to achieve ideal control without precise system parameters [2]. The work by Montanaro et al. deepens this concept by introducing an Enhanced Model Reference Adaptive Controller (EMRAC) to improve the tracking of reference dynamics [5]. This EMRAC controller implemented additional adaptive controls to further improve the closed loop tracking of robotic manipulators. Most relevnat to this study, Amiri et al. implemented a Lyapunov-based Adaptive Controller (LAC) to continuously optimize the parameters of a 5-DOF robotic arm. Showing its performance benifits compared to conventional controllers, specifically when handling step responses.

This work aims to compare the performance of a Lyapunov-based adaptive controller with conventionally tuned PID controllers, using the Ziegler-Nichols and Chau methods in applications of continous tracking. The controllers are applied to a 6-DOF Franka Emika Panda robotic manipulator within the Mujoco physics simulation environment. Their effectiveness is evaluated under two scenarios: an unloaded idea scenario and a scenario simulating the external disturbance of wind, by applying a varying magnitude force from a single dirction.

II. Mujoco Physics Environment

In order to test these controllers a test enviorment needed to be established. The Mujoco physics engine was selected as the simulation environment due to its robust actuator functionalities and extensive library of open-source robotic models. Among these, the Franka Emika Panda robotic manipulator was chosen as the ideal candidate for a 6-DOF system, given its widespread use in Mujoco learning resources and prior exposure to the model. One significant advantage of the Mujoco environment is its built-in functionality, including the capability to compute Jacobians, which greatly simplifies the controller design process.

III. Conventional PID Control

To establish a baseline with which the Lyapunoc Based Adaptive controller, a PID controller first needed to be created. To implement control with a conventional PID controller, it is necessary to determine the proportional gain (K_p), integral gain (K_i), and derivative gain (K_d) for each axis. For this study, these gains were calculated using both the Ziegler-Nichols and Chau methods, providing a basis for comparison. The following procedure was employed for each method in order to find the neccessary gains:

1) Define the range of K_p values to test.
2) Establish the step size for K_p and set thresholds for selecting K_u (ultimate gain) and P_u (oscillation period), such as minimum oscillations and maximum divergence attempts.
3) Simulate robotic behavior by iterating through K_p values:
   a) Compute the error and its derivative.
   b) Map controller effort to joint velocities using the Jacobian matrix.
   c) Monitor divergence and adjust parameters accordingly.
4) Identify peaks in the error signal and calculate the number of oscillations.
5) If the minimum oscillation criterion is satisfied, set $K_u = K_p(i)$ and $P_u =$ Period of the error signal.
6) Repeat the process for each axis.

Once the parameters K_u and P_u were determined, the selected tuning rules were applied. The tuning relationships for the Ziegler-Nichols (Eq. 1) and Chau (Eq. 2) methods are as follows:

$$K_p = 0.6 \cdot K_u, \quad K_i = 0.6 \cdot \frac{K_u}{P_u}, \qquad K_d = 0.075 \cdot K_u \cdot P_u \tag{1}$$

$$K_p = 0.2 \cdot K_u, \quad K_i = 0.3636 \cdot \frac{K_u}{P_u}, \quad K_d = 0.066 \cdot K_u \cdot P_u \tag{2}$$

These gains can be calculated for the enitre manipulator as one, or as seen with some works, gains can be calcuated by joints and contained matrix, describing the complex interrealational dynamics that each joint has with one-another. The per joint gain solution was not used to maintain simplicity. Instead the end effector position was controlled in the task space, where each axis, x, y, z, has its own controller. Originally the task space had one controller, however the individual control enabled the robot to track the trajectory much better than it did before. Thus, this project uses three controllers for each axis shown in the results.

IV. Lyapunov Based Adaptive Control

Traditional control methods are effective for environments with well-defined inputs and outputs, and minimal disturbances. However, there are many scenarios where basic controllers exhibit shortcomings, such as when lifting objects of varying weights or moving along a trajectory under the influence of wind disturbances. In such cases, adaptive control methods can be employed, wherein the controller continuously updates its parameters in response to the system's state. In this paper, Lyapunov Adaptive Control (LAC) is implemented for comparison with traditional methods.

Lyapunov control leverages the inherent nonlinear nature of the robotic arm and ensures asymptotic stability through the existence of a Lyapunov function [1]. Amiri et al. derived an expression for the LAC controller and demonstrated that such a function exists. It is defined as:

$$V = X^T P X + \widetilde{K}_p^T \tau^{-1} \widetilde{K}_p + \widetilde{K}_i^T \tau^{-1} \widetilde{K}_i + \widetilde{K}_d^T \tau^{-1} \widetilde{K}_d, \tag{3}$$

Where X is the state vector and P is the solution to the Lyapunov equation $PA + A^T P = -Q$. Here, Q is a positive definite matrix chosen as a design parameter to aid in constructing the Lyapunov function. The parameter τ represents the adaptive rate, while $\widetilde{K}_p$, $\widetilde{K}_i$, and $\widetilde{K}_d$ denote the differences between the true and estimated controller parameters. To determine the true controller parameters $\hat{\mathbf{K}}$, the derivative of the Lyapunov function is set to $-X^T Q X$, ensuring that the Lyapunov function has a negative derivative. This negative definiteness guarantees stability. Further derivations are provided in [1], but they are omitted here for brevity. The final controller parameters are given by:

$$K_p = \int_0^t \dot{K}_p \, dt + K_{p0}, \tag{4}$$

$$K_i = \int_0^t \dot{K}_i \, dt + K_{i0}, \tag{5}$$

$$K_d = \int_0^t \dot{K}_d \, dt + K_{d0}, \tag{6}$$

where K_{p0}, K_{i0}, and K_{d0} are the initial values of the controller, determined in this work using Ziegler-Nichols tuning. The parameters $\dot{K}_p$, $\dot{K}_i$, and $\dot{K}_d$ are expressed as:

$$\dot{\widetilde{K}}_p = \dot{K}_p = \frac{-\Gamma B^T P X E_r - \Gamma X^T P B E_r}{2}, \tag{7}$$

$$\dot{\widetilde{K}}_i = \dot{K}_i = \frac{-\Gamma B^T P X Z - \Gamma X^T P B Z}{2}, \tag{8}$$

$$\dot{\widetilde{K}}_d = \dot{K}_d = \frac{-\Gamma B^T P X \dot{E}_r - \Gamma X^T P B \dot{E}_r}{2}. \tag{9}$$

The values of $\widetilde{K}_p$, $\widetilde{K}_i$, and $\widetilde{K}_d$ are continuously updated throughout the simulation as the system state evolves. This adaptive setup aims to achieve superior performance compared to traditional control methods.

V. Results

Both conventional controllers and the adaptive controller were provided a circular trajectory to follow. Each controller's performance was first observed under ideal conditions where there was no disturbances. Then the controllers were compared when the arm's end effector was subject to simulation of random wind forces on the end effector. These wind forces are sampled from a normal distribution with $N(500, 250)$. The goal was to see how well the controllers are able to maintain stability when the arm is randomly given unexpected forces.

The control effort metric is used alongside error to demonstrate how efficent each controller is - alongside error reduction. Effort demonstrates that the motion of the arm is minimized to obtain the task.

A. No Wind Conditions

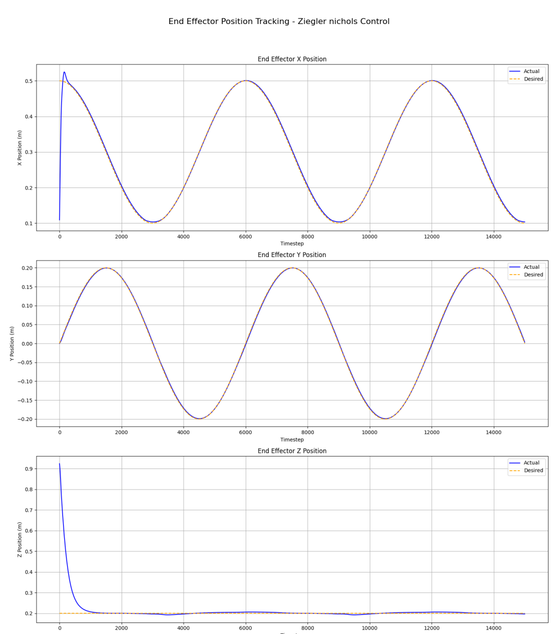


Fig. 1: Control effort for Z-N method in no-wind.

Under no-wind conditions, the Ziegler-Nichols tuned PID controller performed well in tracking the desired trajectory. The controller also demonstrated good efficiency, as shown in the control effort graph in Figure 2.

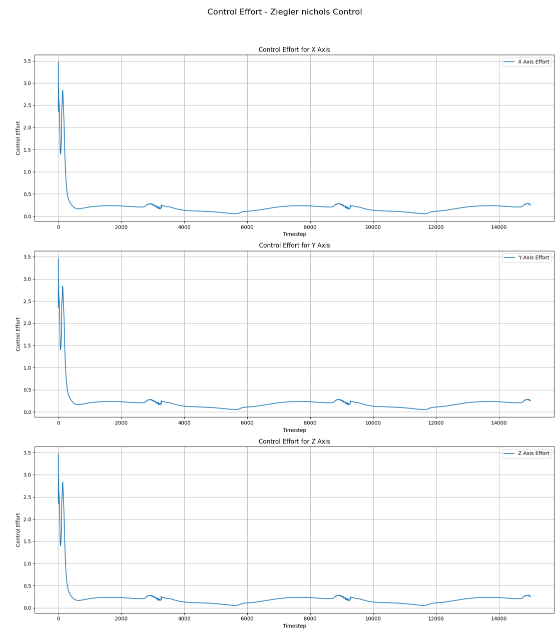


Fig. 2: Control effort for Z-N method in no-wind.

At the start of the simulation, the control effort was higher before settling, a behavior consistent across all controllers. The Chao method exhibited slightly worse performance, consistently overpredicting the desired position, resulting in the end effector leading ahead of the desired Trajectory.

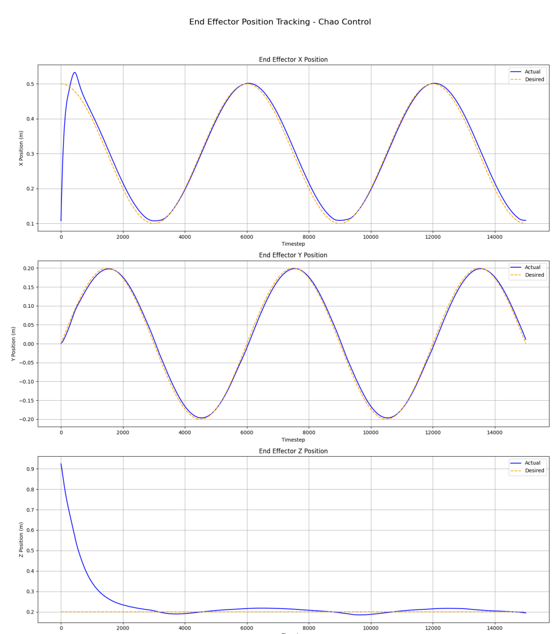


Fig. 3: Trajectory tracking for Chao method in no-wind.

The Chao method also proved less efficient, taking longer than the Z-N method to reduce control effort.

Finally, the adaptive controller's results are shown below.

It exhibited a short settling time, minimal overshoot, and consistently low error throughout the trajectory, as illustrated in Figure 4. Furthermore, the controller gains over time can be seen, and they have a fixed nature after more optimal parameters are found via the luyoponov method.

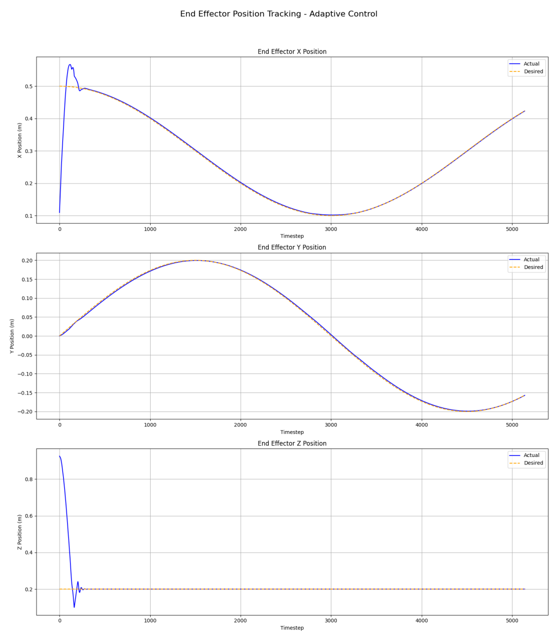


Fig. 4: Trajectory tracking for the adaptive method in no-wind.

Although the adaptive controller's control effort settled slightly more slowly than the PID controllers, it achieved lower steady-state effort, as shown in Figure 5.

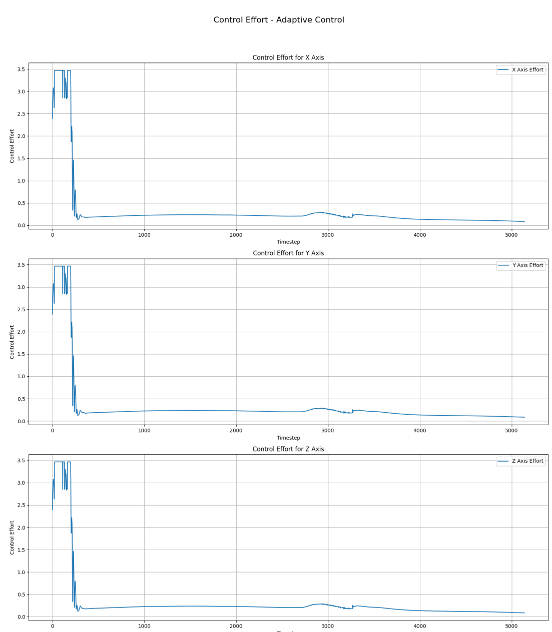


Fig. 5: Control effort for adaptive method in no-wind.

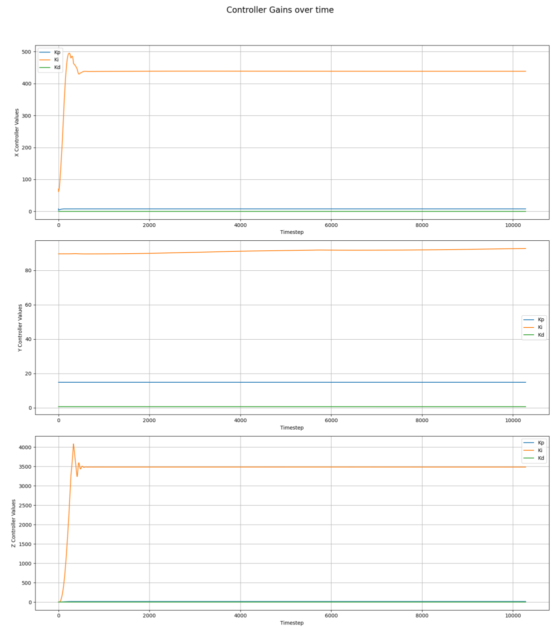
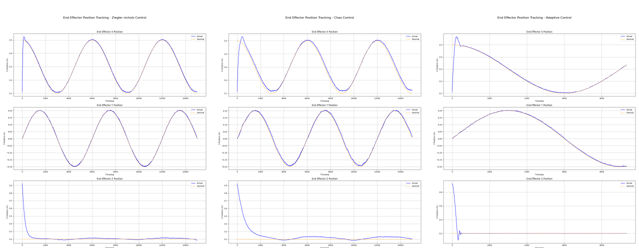


Fig. 6: Gain history for the adaptive method in a no-wind environment.

Each controller was then tested in windy conditions, simulated by a distributed directional force applied randomly based upon a standard distrubution. The results

B. Windy Conditions

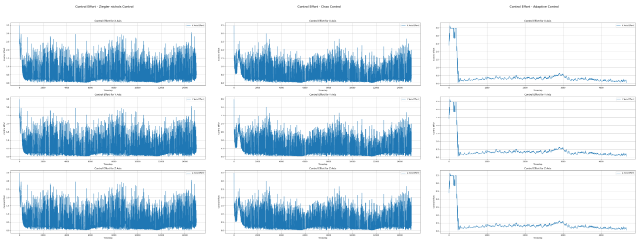
The same trajectory was again provided to the manipulator. The trajectories seen in Figure 7 were produced. Similarly to the no wind scenarios. The Adaptive controller produced the best tracking under wind, followed by the Z-N controller, and then the Chao method.



(a) Z-N (b) Chao (c) Adaptive

Fig. 7: Trajectory tracking methods in wind.

All three of the trajectories appear noticably more noisy than the no wind scenario. The Adaptive Method is once more viewed to produce the fastest convergance to the desired trajectory from the inital conditions.



(a) Z-N (b) Chao (c) Adaptive

Fig. 8: Comparison of control efforts in wind.

When viewing the control efforts under windy scenarios as seen in Figure 8, it is observed that neither the Z-N method or Chao method ever experience a settling of control effort, while the Adaptive controller settles to far lower effort in a similar manner to its no wind scenario.

For the windy condition, the gain history is also seen. best shows the advantages of adaptive control as the gain shifts during the fast convergence to the trajectory before settling as the manipulator performs the stable tracking.

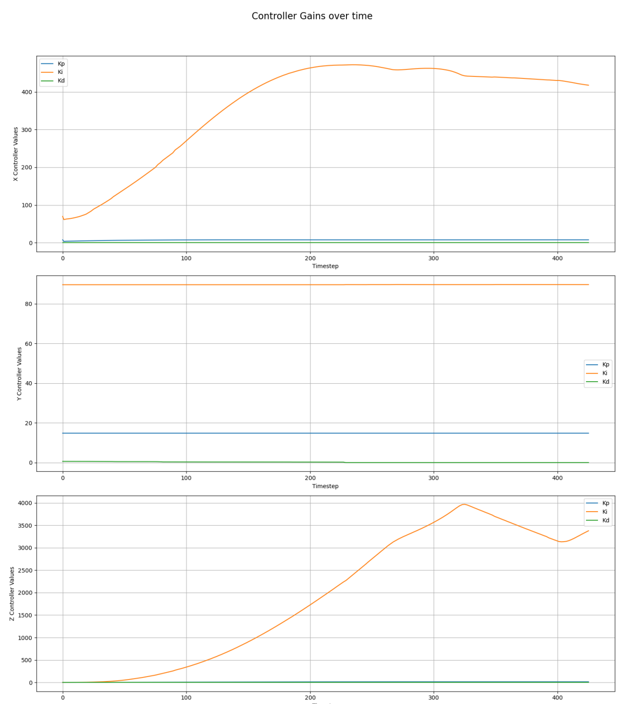


Fig. 9: Gain history for the windy condition.

VI. Discussion

Based on the initial no-wind simulations, it is evident that the Ziegler-Nichols (Z-N) method outperforms the Chao method for conventionally tuning a PID controller in a high-degree-of-freedom (DOF) manipulator. As shown in Figure 10, a comparison of the error norms between the Z-N and adaptive controllers under no-wind conditions reveals that their performance is comparable.

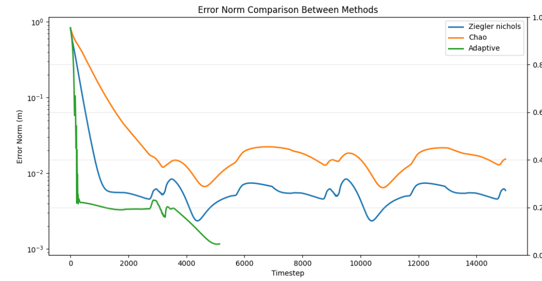


Fig. 10: Error norm under no-wind conditions.

The results from the dynamically stable environment indicate that either the Z-N or adaptive controller is suitable for this scenario, with the choice depending primarily on the specific application. However, it is important to note that these simulations do not account for manipulator joint friction effects. The additional nonlinearities introduced by friction could make the adaptive controller a more favorable choice in practice.

Under windy conditions, as shown in Figure 11, the Chao-based controller consistently exhibited the highest error, while both the Z-N and adaptive controllers demonstrated significantly lower errors.

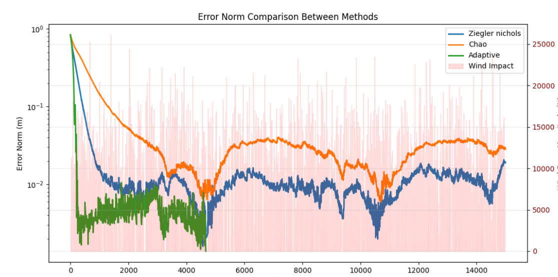


Fig. 11: Error norm under windy conditions.

Using data from both the windy and no-wind simulations, an overall performance score was calculated for each controller based on error norm and control effort. As seen in Figure 12, the adaptive and Z-N controllers achieved relatively high scores of 0.59 and 0.56, respectively, while the Chao controller scored poorly at 0.30.

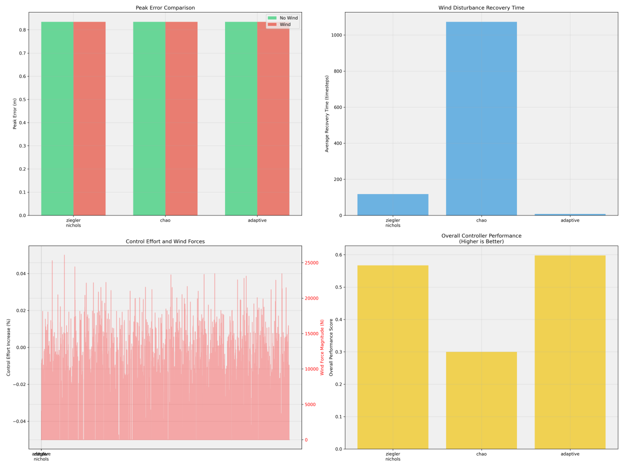


Fig. 12: Overall performance analysis of the controllers.

VII. Conclusion

Based upon this work, the advantages of using an adaptive controller are clear, as the benefits in control effort aswell as in continous tracking accuracy under unpreditable conditions is unmatched by conventional PIDs. Furthermore, the Lyapunov-Based Adaptive controller implented in this study also consistently produced the most accurate trajectory tracking with the least overshoot under ideal conditions further supporting this controller's selection. Neither conventional PID controller was able to achieve convergence showing displaying the inablity of system based conventially tuned PID's to control high degree of freedom systems like the one utilized in this study.

References

[1] M. S. Amiri and R. Ramli, "Utilisation of Initialised Observation Scheme for Multi-Joint Robotic Arm in Lyapunov-Based Adaptive Control Strategy," *Mathematics*, vol. 10, no. 17, p. 3126, 2022. doi: [10.3390/math10173126](https://doi.org/10.3390/math10173126).

[2] Zhang, Dan, and Bin Wei. A review on model reference adaptive control of robotic manipulators. Annual Reviews in Control, vol. 43, 2017, pp. 188198, https://doi.org/10.1016/j.arcontrol.2017.02.002.

[3] Noventino, Timotius Heries, et al. PID control design and kinematic modelling of 3-DOF robot manipulator. 2022 International Conference on Electrical Engineering, Computer and Information Technology (ICEECIT), 22 Nov. 2022, pp. 8894, https://doi.org/10.1109/iceecit55908.2022.10030325.

[4] Wahyuningtri, Slati, et al. Motion Control Design and analysis of UR5 collaborative robots using proportional integral derivative (PID) method. 2021 International Conference on Advanced Mechatronics, Intelligent Manufacture and Industrial Automation (ICAMIMIA), 8 Dec. 2021, pp. 157161, https://doi.org/10.1109/icamimia54022.2021.9807805. '

[5] Montanaro, Umberto, et al. MultiInput enhanced model reference adaptive control strategies and their application to space robotic manipulators. International Journal of Robust and Nonlinear Control, vol. 33, no. 10, 13 Mar. 2023, pp. 52465272, https://doi.org/10.1002/rnc.6639.