Proceedings of the 2023 International Conference on Mechatronics and Smart Systems
DOI: 10.54254/2755-2721/12/20230367

PID control based on machine learning algorithm in robot path optimization

Shijie Guo

College of Maritime, Tianjin University of Technology, Tianjin 300000, China

20214962@stud.tjut.edu.cn

Abstract. Robot path optimization in the robotic arm, robot movement of autonomous navigation and motion control plays a great role, PID control is a traditional, simple and widely used control algorithm. in recent years, machine learning has become increasingly popular, and combining machine learning with PID control has become a potential way to optimize PID control deficiencies. However, the results of the research are scattered in different research teams and different research fields, and in order to integrate the existing research results, we have integrated and analyzed articles from different research teams and research fields. This paper aims to discuss the recent perspectives of robot path optimization based on machine learning algorithms and PID control, summarize the existing problems, research progress and challenges, and put forward the research directions and prospects that can be carried out in the future. In addition, the systematic analysis of research studies contained in this paper helps to consolidate the research results that are still in the final stage and to set the research agenda for future development. This article focuses on robotic arms.

Keywords: PID control system, machine learning, robot path optimization.

1. Introduction

Proportional-integral-derivative (PID) controller is at the heart of the early development of control theory. In 1936, Albert Cal-lender and AIlan Stevenson of Imperial Chemical Limited in Northwich, England, England, and others proposed a PID controller for a temperature control system. Countless researchers have since devoted themselves to its research. Efforts are being made to combine machine learning and PID control, and the optimal path of robots is also being developed. It can be combined with the KNN algorithm to monitor the PID control system and quickly adjust the PID parameters [1]. It is also possible to improve the PID control system in combination with a support vector machine [2]. Improving the problem of automatic tuning of the PID controller [3]. And a machine learning approach using multivariate regression with gradient descent and the normal equation [4]. Use machine learning methods to find the best PID parameters [5]. Application of PID system in robot path planning [6]. In terms of path planning, to use the MATLAB, a microcontroller and servo motors to control robotic arm [7]. The Effect of degrees of freedom on controllers [8]. In addition, there are many recent studies in different subfields, which will not be listed here.

© 2024 The Authors. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/).

DOI: 10.54254/2755-2721/12/20230367

1.1. Problem description and motivation

PID control system is widely used in industrial control systems, the reason is obvious, because it has a simple structure and is easy to implement, but PID control systems also have some problems, including low precision, low degree of automation, and now most engineers use the situation, more use of trial and error methods, PID parameter automatic adjustment is an existing problem, now more and more people have proposed many methods or algorithms to simplify or eliminate the process of trial and error, but in recent years of research, Still not possible to achieve a high degree of automation. And, we can see from the literature cited above that although they all combine machine learning for improvement, they don't know anything about the innovative approaches they come up with in their respective fields due to different fields and teams.

1.2. Research approach and contribution

In order to connect and summarize the current technical level and development status, this paper systematically reviews the methods of optimizing PID control systems combined with machine learning, through this, we can summarize the current problems in the field, and combine the research results of different teams and different fields to develop or apply to the path optimization of robots, in general, we aim to achieve the following goals:

1) It provides a basic classification framework in the form of classification, and classifies existing methods that combine machine learning algorithms to improve PID control systems.

2) An overview of the current state of the art in this area.

3) Point out current trends, gaps, and directions for future research.

We collected papers in several subfields, most of which were published in journals or conferences in the field of artificial intelligence or control theory. We sort out, categorize. Based on the classification basis, we first summarize and classify according to the title, keywords and conclusions of this paper, and comprehensively summarize the current research on machine learning and control theory through careful reading and general classification and analysis. These data are then used to cross-analyze different concepts in taxonomy and derive gaps and possible directions for further research.

1.3. Organization

The rest of the paper is structured as follows: In the 2 sections, we outline the fundamental knowledge about the paper and as the first contribution of this article, four possible starting points are proposed to combine with simple implementation steps. In the 3 sections, we make the second contribution of this paper, which proposes possible challenges and research directions in the future. In the 4 sections, we discussed the approaches and challenges presented. The mind map of the overall structure was shown in the Figure 1.

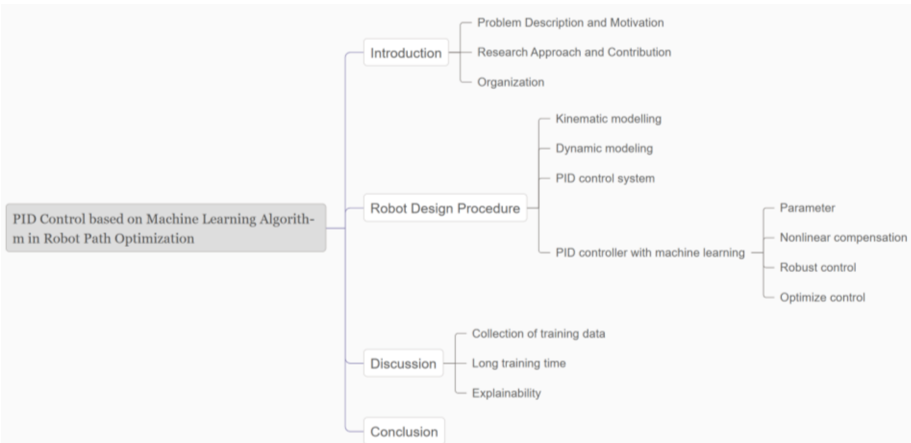

Figure 1. General structure (Photo/Picture credit: original).

2. Robot design procedure

With the continuous development of technology, robots are developing rapidly and there are more and more types of robots. This section mainly discusses robotic arms, which are widely used in daily applications. Like the factory [9] and medicine [10]. Section 2.1 and 2.2 are all based on Matlab to model the robotic arm.

2.1. Kinematic modelling

Kinematic modeling of a robotic arm is the process of mathematical modeling and analysis of its kinematic properties with the aim of describing the position and posture of the robotic arm in space.

Robotic arms are usually made up of multiple joints connected by each joint that can perform rotational or telescopic movements. Kinematic modeling of robotic arms focuses on how to describe and calculate the position and posture of the end effectors of robotic arms, such as robotic jaws, based on a given joint angle or length. In kinematic modeling, Cartesian coordinate systems are often used to represent the pose of a robotic arm.

The core of kinematics modeling is to establish the forward kinematics and inverse kinematics equations of the robotic arm. The forward kinematics equation is used to calculate the position and attitude of the end effector of the robotic arm based on the angle or length of the joint. The inverse kinematics equation solves the angle or length of the joint based on the position and posture of a given end effector. These two equations are key in robotic arm control, enabling precise control and path planning of robotic arms.

In addition to forward and inverse kinematics equations, kinematic modeling of robotic arms involves the calculation of Jacobian matrices. The Jacobi matrix describes the kinematic relationship of the robotic arm, which correlates the velocity of the end effector and the velocity of the joint, and is used to control the movement and trajectory tracking of the robotic arm.

Kinematic modeling of the robotic arm is the basis for the control and planning of the robotic arm and is essential for tasks such as precise positioning, obstacle avoidance, and path planning of the robotic arm. Kinematic modeling provides theoretical guidance for robotic arm design and control, as well as support for robotic arm automation tasks in a variety of application areas.

2.2. Dynamic modeling

The purpose of robotic arm dynamics modeling is to describe its behavior under force and motion control. Robotic arm dynamics modeling involves the following things: mechanical principles, kinematics, and the combined application of dynamics. We need to take into account the structure, inertial characteristics and other dynamic characteristics of the robotic arm. First, kinematic analysis is used to describe the kinematic relationship between the individual links of the robotic arm. Through kinematic analysis, the position, velocity and acceleration of the robotic arm, as well as the relative motion between the individual joints, can be determined.

Secondly, the kinetic analysis studies the mechanical behavior and motion response of the robotic arm. Dynamics modeling takes into account the external forces and moments to which the robotic arm is subjected, as well as the joint forces and moments inside the robotic arm. Through dynamic analysis, the motion equation and mechanical characteristics of the robotic arm can be derived, so as to predict and control the motion and mechanical behavior of the robotic arm.

Dynamic modeling is important for motion control and path planning of robotic arms. By establishing the dynamic model of the robotic arm, the design of motion simulation and control algorithm can be carried out. In addition, dynamic modeling can be applied to torque optimization, collision detection, and safety analysis of robotic arms.

In dynamic modeling, commonly used methods include the Lagrange kinetic method, the Newton-Euler method, and the iterative Newton-Euler method. These methods are based on mechanical principles and kinematic relationships, and the kinematic and dynamic equations of the robotic arm are obtained by modeling and solving various parts of the robotic arm system.

DOI: 10.54254/2755-2721/12/20230367

2.3. PID control system

The PID controller generates a control signal based on the weights of the three parts: proportional, integral, and differential by measuring the difference between the feedback signal and the setpoint. These three parts represent the current error of the system, the accumulation of past errors, and the rate of change of error, respectively. When it comes to PID control algorithms, the following PID formula is usually used to represent the output of the controller: the 'u(t)' stands for the output signal of the controller at the moment 't', the 'e(t)' stands for the error at moment 't', the 'K_p' stands for the Proportional Gain, the 'K_i' stands for the Integral Gain, the 'K_d' stands for the Derivative Gain, the '$\int_0^t e(t)dt$' stands for the integral of the error 'e(t)', the 'de(t)/dt' stands for the reciprocal of the error e(t).

$$u(t) = K_p e(t) + K_i \int_0^t e(t)dt + K_d \frac{de(t)}{dt} \quad (1)$$

1) The proportional part generates a control signal by multiplying the error by a scale factor. Proportional control makes the response of the controller proportional to the error. When the error is large, the proportional controller provides greater correction force.

2) The integral part is by integrating errors to account for past accumulated errors of the system. Integral controllers are used to eliminate steady-state errors that exist in the system, such as those due to friction or imperfect models.

3) The Derivative section predicts the trend of future errors by measuring the rate of change of error. Differential controllers are used to suppress system overshoot and reduce response time to improve system stability.

The output signal of the PID controller is the weighted sum of these three parts. The choice of weights is critical to the performance of the control system, and the optimal weight parameters can be determined through experimentation and adjustment. The principal diagram of conventional PID controller is shown in Figure 2.

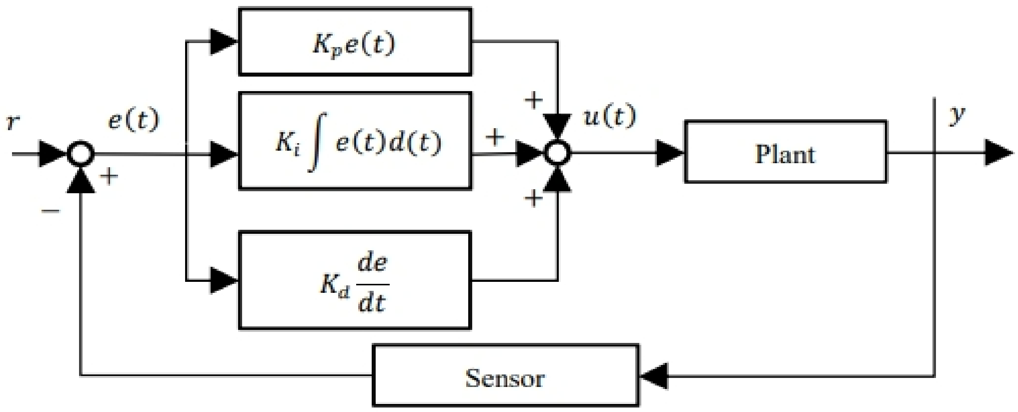

Figure 2. General block diagram for PID controller [11].

2.4. PID controller with machine learning

There are already some applications for PID combined with machine learning algorithms, and the following is a specific example in Figure 3.

DOI: 10.54254/2755-2721/12/20230367

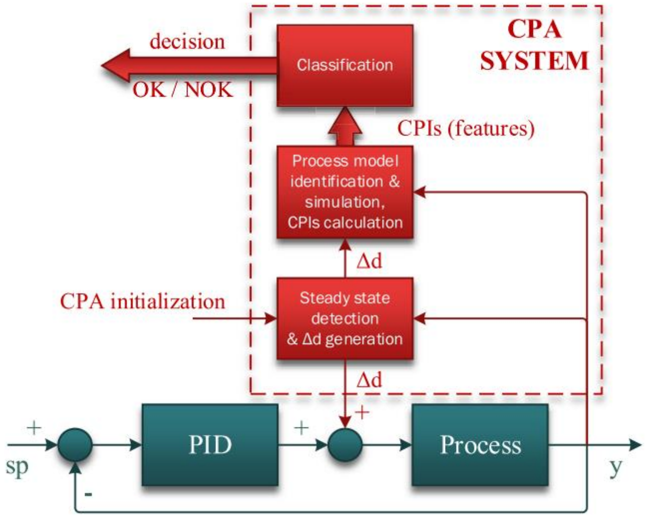

Figure 3. PID-based closed-loop system with schematic of designed CPA system [12].

The PCA process can be started manually by the user or applied periodically by a supervisory control system predefine schedules. After starting the PCA system, the system continuously monitors the value of the output to ensure that its steady state can be monitored. The load disturbance step change Δd will then participate in the adjustment of the system and the process response output until this interference is completely suppressed and a new steady state is detected. Then, interference Δd is canceled and the entire system returns to normal work. The CPA system calculates certain characteristics of the collected response and classifies whether the control performance is acceptable (OK) or unacceptable (NOK).

Although PID control system is a widely used control system, it is not perfect. It includes: difficulty in debugging and optimization, sensitivity to system changes, static error and steady-state performance limitations, sensitivity to noise and disturbances, and a single control strategy. For certain shortcomings, we can try to combine PID control systems with machine learning to solve or optimize these shortcomings. Here are some optimization ideas:

2.4.1. Parameter. As is well known, the performance of PID controllers largely depends on the selection of parameters. In section 1.1, we mentioned that the adjustment of PID parameters is largely done through trial and error or experience. This method is undoubtedly inefficient and has poor generalization performance. Whenever in a new environment, PID parameters need to be readjusted. Moreover, for inexperienced researchers, the time and effort required are enormous. However, machine learning can provide more intelligent and convenient methods to automatically adjust PID parameters. For example, we can select appropriate PID parameters by using basic algorithms such as K-means algorithm, OCSVM algorithm, SVDD algorithm, etc. to process and analyze the returned data. You can also use Reinforcement learning algorithms, such as Q-learning or Deep reinforcement learning, to automatically adjust parameters in the control system. Machine learning models can learn in real-time environments, continuously adjusting parameters to optimize performance. This can be achieved through the following basic steps. In the first step, we need to collect the data returned by the PID control system, which needs to cover various responses of various working conditions to ensure the universality of the dataset. In the second step, we can perform feature extraction. Extract features from the data returned by the PID control system, and note that these features should describe the performance of the control system well, including but not limited to response time, steady-state error, etc. In the third step, we can perform data preprocessing. Preprocess the data to ensure its accuracy and consistency, facilitating model training. Including data cleaning, normalization, and other operations. In the fourth step, we train the model. Choose an appropriate algorithm model to integrate with the PID control system. Using feature values as inputs to the model and PID parameters as outputs, establish their relationships through model training.

DOI: 10.54254/2755-2721/12/20230367

In the fifth step, we optimize the model. Choose an appropriate algorithm model to integrate with the PID control system. Using feature values as inputs to the model and PID parameters as outputs, establish their relationships through model training. In the sixth step, we verify and adjust. Apply the optimized PID parameters to the control system and perform real-time verification. Based on the actual operation results, the model parameters can be further adjusted.

2.4.2. Nonlinear compensation. In PID control, we often encounter a problem about nonlinear systems, in which the dynamic characteristics of the system may change under different working conditions. Machine learning can be used to compensate for the output of the controller to improve system performance. This can be achieved through the following basic steps. In the first step, we can collect data, we need to collect the data returned by the PID control system, which needs to cover various responses of various working conditions to ensure the universality of the dataset. In the second step, we can build a nonlinear model. Using machine learning to establish nonlinear models. The input data is the system's state variables and input signals, while the output data is compensation or correction silver. In the third step, we can preprocess and train the data. Preprocess the data to ensure its accuracy and consistency, facilitating model training. Including data cleaning, normalization, and other operations. Use these data to train nonlinear models. In the fourth step, we do nonlinear compensation. In operation, we embed the nonlinear model into the PID controller and calculate the compensation or correction factor based on the current system state and input signal. Add compensation to the output of the PID controller to improve the compensation ability for system nonlinearity.

Overall, we learn the nonlinear characteristics of the system through machine learning and adjust the output of the PID to improve system performance.

2.4.3. Robust control. The purpose of this method is to make the system robust to changes in parameters, external changes, etc. Simply put, it means being able to maintain good working conditions and performance in different or complex situations. This can be achieved through the following basic steps. In the first step, we can design a model. We can add the compensation part of the robust controller to the PID controller to deal with parameter changes and external disturbances. In the second step, we can model the range of parameters. In the third step, we can design a controller. According to the parameter uncertainty Deterministic system and the system's requirements for robustness, the compensation part is designed. This part can be based on machine learning methods. In the fourth step, adapting the parameters. Through real-time monitoring of system performance and parameter changes, the parameters can be adjusted adaptively, which can use machine learning algorithms, including Reinforcement learning or recursive least squares to achieve parameter optimization. In the fifth step, providing feedback and adjustments through real-time output to achieve goals.

2.4.4. Optimize control. It is based on mathematical optimization theory and technology, which optimizes the performance indicators of the system by adjusting the parameters or input signals of the controller. This can be achieved through the following basic steps. In the first step, we can define optimization objectives. To optimize control, we usually need to first define optimization objectives. This goal needs to basically meet our needs. In the second step, we can establish a mathematical model, including system dynamic equations, constraints, and performance indicators. In the third step, we can optimization algorithm. Choose a suitable optimization algorithm to solve the optimization problem of the objective function. Common optimization algorithms include Gradient descent, genetic algorithm, particle swarm optimization, etc. In the fourth step, we can optimize the input or output signals of the controller. In the fifth step, we can provide feedback and adjustments through real-time output to achieve goals.

3. Discussion

This paper integrates and analyzes articles from different research teams and research fields, discusses the latest views of robot path optimization based on machine learning algorithms and PID control,

DOI: 10.54254/2755-2721/12/20230367

summarizes the existing problems, research progress, and proposes four possible methods for combining machine learning and PID control systems.

For Section 2.4.1, it is feasible and effective to adjust parameters in combination with machine learning, and machine learning algorithms are applied in other fields, such as training models on contaminated data to achieve the effect of classifiers, and can judge unknown data, similarly, the data fed back by the PID system, we can compare it to pollution data, of course, we can also achieve the purpose of training the model to adjust the parameters of the PID system. For Section 2.4.3, the same principle as in 2.4.1, the core is to build the model or controller. For sections 2.4.2 and 2.4.4, the steps are roughly the same, except that they tend to build mathematical models. Although slightly different, these four improvement methods are based on the characteristics of machine learning to improve the performance of PID control system, or pre-processing for the built data set to achieve the purpose of improvement.

At the same time, in the process of combination, there will be some problems and challenges. The performance of PID control systems should be improved by combining machine learning with PID control systems, but we propose the following challenges that can be studied in the future in view of the weaknesses of machine learning.

3.1. Collection of training data

Machine learning algorithms need a large number of training samples to train, for some real-time systems, obtaining training samples is a challenge, for example, for training models, we usually need a lot of abnormal samples and normal samples to form a training set and a test set, but in the real-time operation of the system, abnormal samples are few and not easy to obtain, which leads to us not having enough abnormal samples to train the model, resulting in the model cannot be established. So for this challenge, it is worth studying for researchers.

3.2. Long training time

For machine learning, we know that it usually takes a long time, hours or even days to train a model, and for real-time control systems, in the application, it is impossible to wait for a long time to train the model, which seriously affects the real-time nature of the control system, resulting in a good application in reality. This is a question worth studying.

3.3. Explainability

Traditional PID controllers have good interpretability, compared to some machine learning methods, it is difficult to explain their decision-making process and internal logic. This may result in its inability to be applied in some special areas. In summary, we can think that the combination of machine learning algorithms and PID algorithms to improve the performance of PID control systems is effective and promising, and it is easy to implement, because the specific operation steps have been given, and the problems are also worthy of researchers' research.

4. Conclusion

The use environment of the robotic arm is sometimes very complex, requiring its control system to be well adapted and adjusted to the optimal working performance in various environments. The construction of the robotic arm is easy to implement, the difficulty lies in its control, PID control system is a mature and widely used control system. Machine learning is a new and hot field in recent years. At the same time, there are many algorithms and methods for machine learning algorithms that can be used by us to achieve research goals. After the discussion in Section 2.4, we think that the combination of PID system and machine learning has potential development prospects and advantages in achieving its intelligent, adaptive and robust control, although there are some challenges, such as the acquisition of the dataset we mentioned in section 3, which requires a lot of training time and interpretability, but with the continuous development of technology and researchers' continuous research, these problems will eventually be solved or optimized, and the combination of PID system and machine learning will be well applied and play an important role in various fields.

References

[1] M. Xu and P. Wang, "Evidential KNN-based Performance Monitoring Method for PID Control System," 2020 5th International Conference on Mechanical, Control and Computer Engineering (ICMCCE), Harbin, China, 2020, pp. 597-601.
[2] W. Yue, "Study on PID Control of Aquatic Product Dryer based on Support Vector Machine Optimized by Improved Particle Swarm Algorithm," 2020 2nd International Conference on Applied Machine Learning (ICAML), Changsha, China, 2020, pp. 366-370.
[3] D. Kucherov, A. Kozub, V. Tkachenko, G. Rosinska and O. Poshyvailo, "PID Controller Machine Learning Algorithm Applied to the Mathematical Model of Quadrotor Lateral Motion," 2021 IEEE 6th International Conference on Actual Problems of Unmanned Aerial Vehicles Development (APUAVD), Kyiv, Ukraine, 2021, pp. 86-89.
[4] A. Zulu, "Towards explicit PID control tuning using machine learning," 2017 IEEE AFRICON, Cape Town, South Africa, 2017, pp. 430-433.
[5] R. Büchi, "Optimal ITAE Criterion PID Parameters for PTn Plants Found with a Machine Learning Approach," 2021 9th International Conference on Control, Mechatronics and Automation (ICCMA), Belval, Luxembourg, 2021, pp. 50-54.
[6] S. C. Suhaimin, N. L. Azmi, M. M. Rahman and H. M. Yusof, "Analysis of Point-to-Point Robotic Arm Control using PID controller," 2019 7th International Conference on Mechatronics Engineering (ICOM), Putrajaya, Malaysia, 2019, pp. 1-6.
[7] S. C. Suhaimin, N. L. Azmi, M. M. Rahman and H. M. Yusof, "Analysis of Point-to-Point Robotic Arm Control using PID controller," 2019 7th International Conference on Mechatronics Engineering (ICOM), Putrajaya, Malaysia, 2019, pp. 1-6.
[8] P. Yeptho, K. Katheria, S. Swain and P. Gaur, "Analyzing the Impact of Degrees of Freedom on the Performance of Fractional Order PID Controllers in Robotic Manipulators with Payloads," 2020 IEEE 17th India Council International Conference (INDICON), New Delhi, India, 2020, pp. 1-6.
[9] R. Rahman, M. S. Rahman and J. R. Bhuiyan, "Joystick controlled industrial robotic system with robotic arm," 2019 IEEE International Conference on Robotics, Automation, Artificial-intelligence and Internet-of-Things (RAAICON), Dhaka, Bangladesh, 2019, pp. 31-34.
[10] C. L. Fall et al., "Intuitive wireless control of a robotic arm for people living with an upper body disability," 2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Milan, Italy, 2015, pp. 4399-4402.
[11] Y. He, Y. Tang, D. Lee and J. Ahn, "Suspending control scheme of 8/10 bearingless SRM based on adaptive fuzzy PID controller," in Chinese Journal of Electrical Engineering, 2016, 2(2), pp. 60-67.
[12] P. Grelewicz, T. T. Khuat, J. Czeczot, P. Nowak, T. Klopot and B. Gabrys, "Application of Machine Learning to Performance Assessment for a Class of PID-Based Control Systems," in IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2023, 53(7): pp. 4226-4238.