

Article

Adaptive Internal Model Control Based on Parameter Adaptation

Peng Chen ^{1,*}, Yunfeng Zhang ¹, Jianhong Wang ¹, Ahmad Táher Azar ^{2,3,*}, Ibrahim A. Hameed ^{4,*}, Ibraheem Kasim Ibraheem ⁵, Nashwa Ahmad Kamal ⁶ and Farah Ayad Abdulmajeed ⁷

¹ College of Electrical Engineering and Automation, Jiangxi University of Science and Technology, Ganzhou 341000, China

² College of Computer and Information Sciences, Prince Sultan University, Riyadh 11586, Saudi Arabia

³ Faculty of Computers and Artificial Intelligence, Benha University, Benha 13518, Egypt

⁴ Department of ICT and Natural Sciences, Norwegian University of Science and Technology, Larsgardsvegen, 2, 6009 Alesund, Norway

⁵ Computer Engineering Techniques Department, Al-Mustaqbal University College, Hilla 51001, Iraq

⁶ Faculty of Engineering, Cairo University, Cairo 12613, Egypt

⁷ Aeronautical Department, College of Technical Engineering, Alfarahidi University, Baghdad 10001, Iraq

* Correspondence: zaq18379663911@163.com (P.C.); aazar@psu.edu.sa or ahmad.azar@fci.bu.edu.eg or ahmad_t_azar@ieee.org (A.T.A.); ibib@ntnu.no (I.A.H.)

Abstract: In actual industrial control, many objects have the characteristics of large lag and time-varying parameters, which makes it difficult for traditional PID control to obtain satisfactory control effects. Therefore, engineers need a control algorithm with a better control effect and a simple structure, and thus the internal model control is introduced into industrial process control. Internal model control has received a lot of attention in the control field because of its excellent control effect, and in ideal conditions, it can perfectly suppress external disturbances, and the output strictly tracks the input. On the basis of the internal model control, a two-degrees-of-freedom internal model control was developed. Compared with the conventional feedback control, the internal model control structure mainly embeds an internal model consistent with the control object in the control object, so the deviation between the internal model and the control object determines the quality of the control effect. In this paper, the internal model is changed to an adjustable parameter model, and the model parameters are adjusted in real time using the parameter adaptive algorithm, so that the model output error is as small as possible, or even zero. In order to solve the influence of interference on the system, compensation based on MRAC theory is used. In order to verify the feasibility of the algorithm, it was applied to the landing process control of fixed-wing Unmanned aerial vehicle (UAV) and achieved satisfactory results.



Citation: Chen, P.; Zhang, Y.; Wang, J.; Azar, A.T.; Hameed, I.A.; Ibraheem, I.K.; Kamal, N.A.; Abdulmajeed, F.A. Adaptive Internal Model Control Based on Parameter Adaptation. *Electronics* **2022**, *11*, 3842. <https://doi.org/10.3390/electronics11233842>

Academic Editor: Davide Astolfi

Received: 5 September 2022

Accepted: 26 October 2022

Published: 22 November 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Due to the needs of various technical fields, automation technology has gradually developed. At the beginning, people imitated human operation process, put forward the automatic control system based on the feedback principle, and developed the classic proportional integral derivative (PID) control algorithm. The PID control algorithm has low requirements for the model, and online adjustment is also convenient, so it is favored by engineers. According to the estimation of experts, eighty percent of the control system still uses the PID algorithm. However, for some systems with large inertia and delay, using PID control cannot achieve good control results. In addition, there are also some complex systems (such as nonlinear, time-varying systems, large time-delay systems, and non-minimum phase systems). The effect of PID control is often poor and it is difficult to meet the process requirements. These complex systems have become the key to designing control systems. Coupled with the energy shortage in the 1960s and the rising prices of raw materials, in order to reduce costs and improve productivity, engineers gradually

developed a state space-based modern control theory with a cheap, high-performance computer, for the application of modern control technology, providing a material basis. In the actual process (especially in the chemical process), the object is often nonlinear and time-varying. Adaptive technology is an effective method to solve this kind of situation. So far, the modern control theory is basically the use of parametric models, such as state equations and transfer functions. The model requires a relatively high accuracy; in fact, the model order is a more difficult thing to determine.

In this case, researchers began to seek control approaches that did not need great model accuracy while still achieving the same optimization results. Recently developed artificial intelligence, self-organizing self-learning systems, etc., are branches of this development, and gradually reflect their importance. At the same time, there is a kind of algorithm, generally known as output predictive control, which has attracted more and more attention. This kind of algorithm has the advantages of low model requirements and high robustness, and has a great development prospect in process control. These algorithms mainly include DMC (dynamic matrix control), MAC (model algorithm control), IMC (internal model control), and GPC (generalized predictive control).

Internal model control is practical, and has the advantages of simple design, intuitive structure, and less adjustment coefficient. Especially in the system with time delay, it can effectively improve the anti-interference and robustness of the system. Since it was proposed, it has been applied to slow response control, and has achieved good application achievements in a large number of fast response control processes, such as motor control. After more than ten years of research and development, IMC has not only been applied in nonlinear and multivariable systems, but also produced many typical algorithms related to internal model, such as predictive control method, zero cancellation method, IMC-PID, and so on. The combination of IMC with other control methods, such as adaptive IMC, intelligent IMC using fuzzy decision making, humanoid control, neural networks, etc., is also easy. With the development of the internal model control to the present stage, it has been successfully applied in a variety of predictive algorithms, and can further become a powerful tool for forecasting research.

The internal model control algorithm was proposed by American professors Morari and Garcia [1]. They improved on the two very popular prediction algorithms at that time, DMC and MAC, and introduced the prediction space and control length of the prediction algorithm into the internal model control algorithm, and proposed a method for solving the model to approximate you, making it can effectively control the system with non-minimum phase. Therefore, compared with model algorithm control and dynamic matrix control, the internal model control algorithm is more competent.

Adaptive technology has been studied in predictive control. Larimore et al., in order to introduce adaptive technology in MAC, proposed a model in the form of state equation identified by the canonical variable method. Takamatsu used the structural model as the internal model, and proposed an internal model control based on model reference adaptive control. Both methods use structural models in identification, so it is necessary to determine the order of the model online. In order to avoid the difficulty of the on-line identification of the structural model, Richard put forward the online identification of the dynamic matrix, composed of unit impulse coefficient, and then constituted the adaptive dynamic matrix control. However, because the impulse response coefficient is often greater, and the dynamic matrix control technique and multi-step prediction require a significant amount of calculation, their applicability in engineering is limited. In order to reduce the amount of calculation in this case, this paper proposes a new adaptive internal model control algorithm. In this algorithm, the error model is identified online, and the error value of the output prediction is predicted by the error model, so as to achieve the adjustment of the parameters of the internal model controller. When the object drifts within a certain range, a satisfactory adaptive effect is obtained without having to calculate a large number of parameters each time, thereby greatly reducing the amount of calculation.

In the field of industrial process control, simple PID control has been able to solve about 90% of control problems, but most of the controlled processes have more complex mechanisms, such as pure lag, slow time-varying and highly nonlinear mechanisms, and so on. Under the influence of other conditions such as load or noise disturbance, the parameters and model structure of the process model may change, so that there are some uncertainties in the actual process system. When using a traditional PID controller, a set of fixed PID parameters can be used to adjust the system to changes in the controlled process parameters, interference, and other uncertainties. It is difficult to obtain a satisfactory control effect. When the parameter changes exceed the range, the control performance of the system will deteriorate significantly, even beyond the expected range. At the same time, for the PID regulator, the parameter tuning problem is also very difficult. The design level of the PID controller is improved by using the internal model control principle. Compared with the classical PID control, the internal model control has only one parameter, and the relationship between the parameter adjustment and the dynamic quality and robustness of the system is clear. In recent years, with the efforts of researchers, the internal model control principle has been extended to the field of nonlinear systems, and the internal model control has been further developed.

At present, the role of fixed-wing UAVs is not only becoming more and more important on the battlefield, but its potential in civilian and commercial fields is being further explored, ranging from unmanned reconnaissance aircraft and attack aircraft in the military and police industry to aerial photography and express delivery. The applications of these aspects all reflect the role of fixed-wing UAVs in today's world. The research on fixed-wing UAV technology is not only of great significance to the cause of national defense, but also of great significance for improving the quality of life of the general public. Because fixed-wing UAVs play a huge role in military and civilian use, domestic and foreign researchers have carried out a lot of in-depth research on them. The main research directions include formation control, cluster control, and autonomous operation of fixed-wing UAVs, as well as drop control, target tracking control, etc. Trajectory tracking control is one of the research directions, which aims to control the UAV to accurately track the desired trajectory. Trajectory-tracking control plays an important role in the field of fixed-wing UAV control, and is the basis for UAV tracking moving targets and trajectory planning. However, the main method of trajectory-tracking control of fixed-wing UAVs is still based on the model, supplemented by various improved control methods. This paper attempts to combine adaptive control and internal model control to design the control system of fixed-wing UAVs, in order to make some small contributions to the flight control of fixed-wing UAV and solve some flight problems. Some of the concerns are under control, but due to the amount of individual ability, only a subset of them have been investigated.

An adaptive controller should be one that modifies its properties to adapt to changes in the dynamic properties of objects and disturbances. This self-adaptive control method should achieve—during system operation and based on continuous collection of control process information—the determination of the current actual working state of the controlled object, optimization of the performance criteria, and generation of the self-adaptive control law, allowing the controller structure to be adjusted in real time. Or parameters, so that the system always automatically works in the optimal or sub-optimal operating state.

Since the first adaptive control system was proposed by the Massachusetts Institute of Technology in the late 1950s, many different forms of adaptive control systems have appeared successively. Model reference adaptive control and the self-tuning regulator are two mature adaptive control systems.

One of the fields where MRAC systems have been used most successfully in the past is in the field of electric traction. For example, the earliest application is the adaptive control to the thyristor-powered DC power drag system. Since the use of conventional PI regulators for speed feedback control cannot guarantee the required high-performance indicators, the adaptive control scheme can approximate the object as a second-order system, and only two parameters can be adjusted to ensure that the performance indicators will not change

when the parameters of the object change. It can also overcome the dead zone problem that cannot be solved by the PI regulator when the motor speed crosses zero.

The application of MRAC technology in automata is also very active, such as the literature based on the Lyapunov stable design of the MRAC system and its use in automata. The MRAC system designed based on the hyper stable theory in the literature is also used in automata. These applications basically solve the problem of nonlinearity and interference between degrees of freedom in automata.

The application of MRAC technology in ship autonomous driving is also very successful. For example, the application of MRAC technology in the automatic driving of ships is reported in the literature. It can simplify the nonlinear model into a second-order linear model. In this way, when the external environment (wind, waves, currents, etc.) changes, the dynamic characteristics of the ship will vary with the trim load and water depth, the autopilot with adaptive control can achieve the required performance, and the operation is safe and reliable. In addition, MRAC technology also has applications in other fields, such as internal combustion engines, oxygen-blown steel furnaces, and hydraulic servo systems.

The following summarizes the research progress of scholars at home and abroad in recent years. Reference [2] introduces virtual feedback correction control, where the output data is used to directly design the controller to avoid the system modeling process, and the model reference problem is transformed into the unknown parameters of the controller to solve the problem. Reference [3] first analyzes the robust stability conditions and parameter design criteria of the system when the models are mismatched, and then adaptively adjusts the filter parameters by introducing the optimization method of the quadratic performance index function to improve the robustness and control performance of the system. Reference [4] uses the all-pole approximation method to approximate the delay link in the control system, and the Kalman filter method is introduced as a nominal parameter model to update the gain of the internal model controller in real time. In reference [5], an adaptive disturbance observer is constructed in the vibration model of an internal combustion engine, which is used as an internal model of the control system to cancel the vibration disturbance in the control system. Reference [6] proposes a novel control scheme combining the inverse system approach and internal model control. The BPMSM inverse model is cascaded with the original BPMSM system to form a decoupled pseudo-linear system. In addition, in order to improve the robustness of the whole system and suppress the influence of unmodeled dynamics and system noise on the decoupling control accuracy, an additional closed-loop controller is designed for the pseudo-linear system using the internal model control method. In references [7,8] an internal model control is used as the main controller in the control algorithm of a hard disk drive, and the model reference adaptive control is added to correct the controller parameters. The biggest advantage of internal model control is that it can improve the robustness and anti-interference of the feedback system, and control those large time-delay systems. It has achieved incomparable results compared with PID in the process control of slow response and the control of servo motors, with fast-response results. The internal model control also has many typical generalized forms, such as the zero-pole cancellation method, predictive control method, and internal model control for PID control [9]. Internal model control is also combined with many other advanced control methods, such as internal model control and fuzzy control [10], internal model control and adaptive control [11], internal model control and optimal control [12], mixed H₂/H_∞ control, and the nonlinear tracking method [13]. Neural networks are also introduced into internal model control [14]. The combination of predictive control has made the internal model control continuously improve and become widely used in engineering practice, achieving good results.

In the field of UAV-trajectory tracking, the autonomous landing of UAV has been a key and complex process during all the stages of flight. Accidents appear more frequently due to severe wind and aerodynamic instability caused by the ground [15–18]. Adaptive backstepping control has been used for path tracking [19,20]. A novel method of adaptive internal model control has been proposed, which is an essential robust control strategy

based on the model [21–24]. In reference [25], two different multivariable control technologies are used, linear quadratic regulator and model predictive control, which, respectively, describe the impact on path planning and guidance system of UAVs. Reference [26] uses the YOLOv4 network to quickly identify unauthorized UAVs flying into power plants or airports, so as to timely expel these UAVs and avoid danger. A new YOLOv4 network has been proposed to identify various kinds of UAVs and birds and animals.

This paper first introduces PID control in the introduction and explains the shortcomings of PID control, in order to introduce internal model control. For some complex systems, such as time-varying systems, since internal model control is sensitive to model uncertainty, a parameter adaptive algorithm is introduced. In Section 2, the principle of internal model control and its two improvements are described in detail. The first point is to introduce the idea of adaptive control to adjust the filter parameters in internal model control. The second point is to change the internal model in internal model control to a variable parameter model and to introduce a parameter adaptive algorithm to modify the parameters of the internal model. In Section 3, the feasibility of the parameter adaptive algorithm and its fast convergence to the real value are verified by numerical simulation. In order to verify the control value effect of the improved internal model control, it is applied to the landing phase control of fixed-wing UAV. The simulation results show that the improved control effect is significantly better than the original control effect.

2. Internal Model Control Algorithm

The structure of a classic internal model control system is shown in Figure 1, where $G_p(s)$ is the controlled process, $G_m(s)$ is the process model, $G_{IMC}(s)$ is the internal model controller, and $r(s), d(s), y(s)$ are the input value, disturbance, and system output of the control system, respectively.

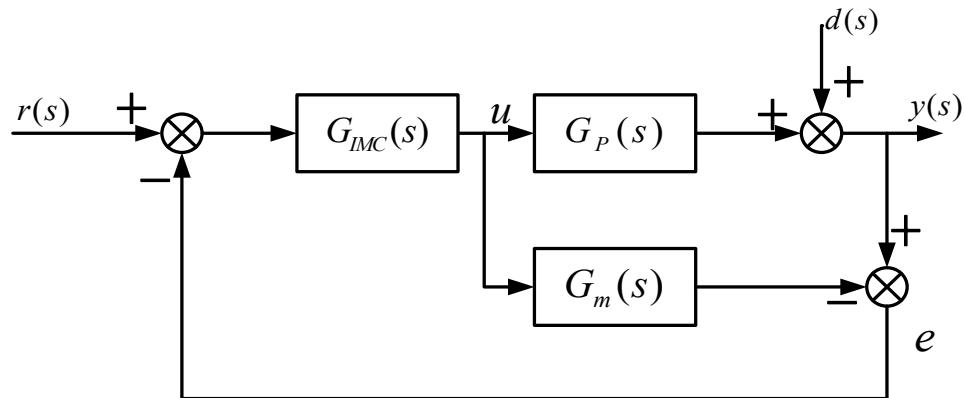


Figure 1. Design of internal model control structure.

In order to facilitate the calculation, the internal model control principal structure diagram shown in Figure 1 is equivalent to the equivalent simple feedback control structure diagram shown in Figure 2. For the equivalent internal model control structure diagram shown in Figure 2, the part in the dashed box $C(s)$ is the feedback controller, which can be obtained from Figure 2:

$$C(s) = \frac{G_{IMC}(s)}{1 - G_{IMC}(s)G_m(s)} \quad (1)$$

The input-output relationship in Figure 2 is:

$$\frac{y(s)}{r(s)} = \frac{C(s)G_p(s)}{1 + C(s)G_p(s)} \quad (2)$$

$$\frac{y(s)}{d(s)} = \frac{1}{1 + C(s)G_p(s)} \quad (3)$$

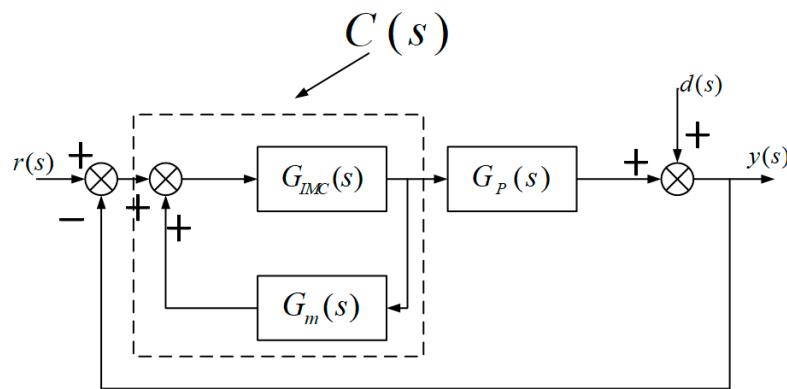


Figure 2. Design of equivalent internal model control structure.

Substitute Equation (1) into Equation (2) and Equation (3), and rewrite the system closed-loop state response as:

$$y(s) = \frac{G_{IMC}(s)G_p(s)}{1 + G_{IMC}(s)[G_p(s) - G_m(s)]}r(s) + \frac{[1 - G_{IMC}(s)G_m(s)]}{1 + G_{IMC}(s)[G_p(s) - G_m(s)]}d(s) \quad (4)$$

In addition, (5) is called the sensitivity function, the sensitivity of the system to external disturbance; similar (4) is called the complementary sensitivity function.

It can be seen from Figure 1 that the feedback signal is:

$$e(s) = [G_p(s) - G_m(s)]u + d \quad (5)$$

If the controlled process and the mathematical models of process model are equal, and there is no external disturbance, that is, $d = 0$. Then the output y_m of the process model is equal to the output y_p of the process, and the feedback signal is zero. It can be seen from Equation (6) that when the internal model controller $G_{IMC}(s)$ takes the inverse of the mathematical model of the process model, that is, $G_{IMC}(s) = G_m^{-1}(s)$, the output of the system will be equal to the output of the system, that is to say, the system has the characteristics of open-loop control.

2.1. Internal Model Controller Design

In an ideal situation, when $G_p(s) = G_m(s)$ and $G_{IMC}(s) = G_m^{-1}(s)$, its possible control effect can realize the output-to-input static-free tracking under any external disturbance. No adjustments to the controller are required in this case. However, in the actual industrial production control, the ideal internal model controller is difficult to realize. There are three reasons for this result: (1) the controlled process often contains a time-delay term $e^{-\tau s}$, which leads to a leading term in the inverse operation, which is physically inconsistent with causality, so it is difficult to achieve. (2) If $G_p(s)$ contains a zero point on the right half-plane, under open-loop conditions, both the internal control controller and the controlled process are unstable. (3) High-order differential terms easily appear when selecting the internal model controller, which will make the controller's ability to suppress noise worse.

In order to solve the above problems, the following steps are usually taken when designing the internal model controller:

Step 1: Decompose process model $G_m(s)$

$$G_m(s) = G_{m-}(s) \cdot G_{m+}(s) \quad (6)$$

where $G_{m+}(s)$ is the part of the model that contains pure lag and right half-plane zeros, and $G_{m-}(s)$ is the minimum phase part of the model.

Step 2: Design the IMC controller

Before designing the IMC controller, a controller $\tilde{G}_{IMC}(s)$ can be designed first. This controller can ignore the constraints and the uncertainty of the model. It is necessary to make the system have a better response under the input, and then add a low-pass filter f to ensure robust stability and robust performance.

For single-input single-output (SISO) systems, $\tilde{G}_{IMC}(s)$ is usually chosen to be H2-optimal for a particular input v , where $v = -r$ or $v = -d$. So $\tilde{G}_{IMC}(s)$ can be solved

$$\min_{\tilde{G}_{IMC}} \|e\|_2 = \min_{\tilde{G}_{IMC}} \|(1 - G_{IMC}G_p)v\|_2 \quad (7)$$

Equation (9) is stable and causal subject to $G_p(s)$. Equation (9) is the minimum that can be achieved when

$$\tilde{G}_{IMC}(s) = G_m^{-1}(s) \quad (8)$$

However, the inverse of the model is acceptable if and only if for the minimum phase system. For non-minimum phase systems, the model inverse obtained by Equation (10) is either unstable or acausal. Since the objective function cannot reach zero, it is necessary to find an approximate inverse which minimizes the 2-norm of the sensitivity function.

The decomposition of $G_m(s)$ and $r(s)$ is obtained by the first step. The H_2 optimal solution of the controller is

$$\tilde{G}_{IMC}(s) = (G_{m-} \cdot r_-)^{-1} \left\{ G_{m+}^{-1} \cdot r_- \right\}_* \quad (9)$$

where $\{\cdot\}_*$ means that after the partial fractional expansion of the internal expression of $\{\cdot\}$, the items involving the poles of G_{m+}^{-1} are all omitted.

Because \tilde{G}_{IMC} is an approximate inverse, in order to ensure robustness and stability, a low-pass filter f needs to be added after \tilde{G}_{IMC}

$$G_{IMC}(s) = \tilde{G}_{IMC}(s)f \quad (10)$$

where the filter f has the form $f = 1/(\lambda s + 1)^n$, where n represents the order of the filter, and n should be chosen to be a large enough number that $G_{IMC}(s)$ is achievable. λ is the filter time constant, which is also the only design parameter of the internal model controller. The size of λ affects the response time and robustness of the system, so how to choose a suitable λ is also a problem.

After introducing the above design, the output response of the system is:

$$y(s) = \frac{\tilde{H}(s)[1 + e_m(s)]}{1 + \tilde{H}(s)e(s)} [r(s) - y(s)] \quad (11)$$

$$e(s) = \frac{1 - \tilde{H}(s)}{1 + \tilde{H}(s)e_m(s)} [r(s) - d(s)] \quad (12)$$

In the Equations (10) and (11) $e_m(s) = (G_p(s) - G_m(s))/G_m(s)$, $\tilde{H}(s) = G_{m+}(s)f(s)$. Under non-ideal conditions, the controlled process deviates from the reference model. According to the robustness theorem, the closed-loop stability conditions of the internal model control system are:

$$|G_{m+}(s)f(s)| \leq \frac{1}{l_m}, \forall \omega$$

Among them, l_m is the upper bound of model uncertainty, and there is $e_m(s) \leq l_m(s)$ for any $e_m(s)$, which reflects the mismatch degree of the model. In order to facilitate the analysis, let $|G_{m+}(s)| = 1$ and $e_m(s)$ increase, the modulus $|f(s)|$ of the filter must be small. Therefore, the value range of the filter constant is limited by the mismatch between the object and the model. For the design method of the IMC, taking $G_{m+}(0)f(0) = 1$ and

$f(s) = 1/(\gamma s + 1)^\lambda$, it can be judged that they are very close in the normal frequency range by the following inequalities.

$$|e| = \frac{|1 - G_{m+}(s)f(s)|}{1 - |G_{m+}(s)f(s)|} |r(s) - d(s)|$$

When $\omega \leq 1/\lambda$, $G_{m+}(s)f(s) \approx 1$, then $e = 0$; when $\omega \geq 1/\lambda$, $|f(s)|$ is very small, approximately zero, at this time $|e| = |1 - G_{m+}(s)f(s)||r(s) - d(s)|$. So as long as λ is chosen when large enough, the closed-loop response is similar to that of the nominal system $G_{m+}(s)f(s)$ for high or low frequency inputs.

In conclusion, the advantage of IMC is that the controller parameters are related to the model parameters in a unique and direct way. Therefore, it is only necessary to select a suitable λ to achieve a good control effect within a certain error range.

2.2. Design of Internal Model PID Controller

According to Figures 1 and 2, the internal model control structure can be equivalent to a classical closed-loop control structure. Consider a process with unstable zeros, taking a first-order plus time-delay (FOPTD) process as an example

$$G_m(s) = \frac{K}{Ts + 1} e^{-\tau s} \quad (13)$$

From the content of the previous section, the equivalent structure of the internal model controller is:

$$C(s) = \frac{G_{m-}^{-1}(s)}{f(s)^{-1} - G_{m+}(s)} \quad (14)$$

Before designing the controller, the time-delay process part is first processed, and the first-order pada approximation is used here:

$$e^{-\tau s} = \frac{1 - 0.5\tau s}{1 + 0.5\tau s}$$

The approximated process model is

$$G_m(s) = \frac{K(1 - 0.5\tau s)}{(Ts + 1)(1 + 0.5\tau s)} \quad (15)$$

According to the content described by Equation (8), decompose the controlled object $G_m(s)$ into the minimum phase part G_{m-} and the with unstable zero part G_{m+} to obtain

$$G_{m-} = \frac{K}{(Ts + 1)(1 + 0.5\tau s)}$$

$$G_{m+} = 1 - 0.5\tau s$$

Since the process model is a first-order model, the form of the filter is a first-order low-pass filter

$$f(s) = \frac{1}{\lambda s + 1}$$

In summary, the expression form of the internal model controller can be rewritten into the transfer function form of the classic PID controller.

$$C(s) = K_p \left(1 + \frac{1}{T_i s} + T_d s \right) = \frac{K_p [T_i s + 1 + T_i T_d s^2]}{T_i s} \quad (16)$$

As can be seen from the above equations, the parameters of the PID controller are only related to the system parameters and the selected filter time parameters. Therefore, when

the controlled object model and the internal model are matched and known, the controller only needs to adjust the filter time constant, and the size of the filter affects the response speed and stability of the system.

Since the traditional IMC has only one degree of freedom, there is only a trade-off between the response speed of the system and the robustness of the system when selecting filter parameters. In order to improve the dynamic performance of the system and strengthen the robustness of the system, it is necessary to adjust the filter parameters in real time and correct the mathematical model of the controlled process in real time. We call the IMC of this structure adaptive IMC or AIMC. The AIMC structure is shown in Figure 3.

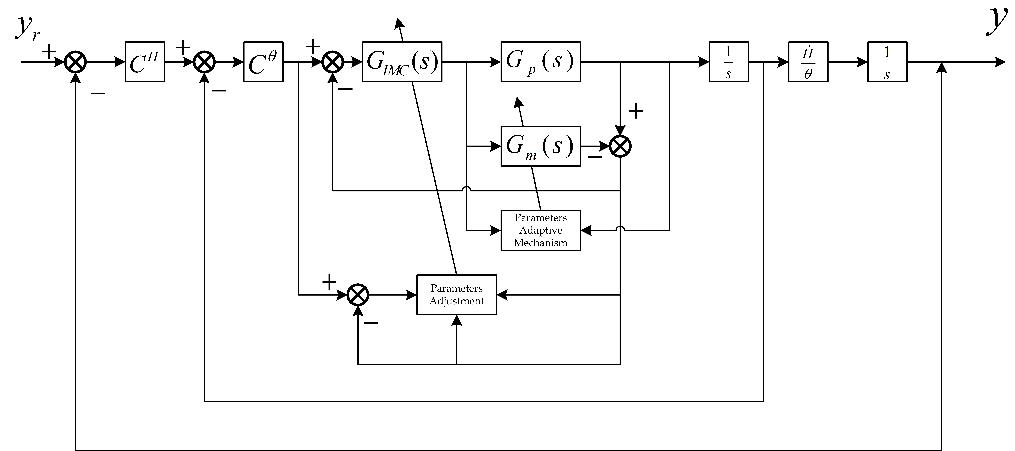


Figure 3. Design of adaptive internal model control structure based on parameter adaptation.

The adjustment rule of the filter parameters should be initialized with a smaller value to improve the system response speed. When the mode error becomes larger, the filter parameters are adjusted online, and the filter parameters should be adaptively amplified to ensure the stability of the system. The adaptive law of the filter parameters is:

$$\lambda = \lambda_0 \left(1 + \int_{t_r}^{t_s} |e| \left| \frac{y - y_m}{y_m} \right| dt \right) \quad (17)$$

where λ_0 denotes the initial filter parameter, $e = r - y$ denotes system error, y denotes system output, y_m denotes output of process model, t_r denotes the first time of achieving system setting point, and t_s denotes the system regulation time.

2.3. Parameter Adaptive Algorithm

In the second subsection, we know that the deviation between the controlled object and its corresponding mathematical model has a great influence on the control effect. Therefore, in this section, the parameter adaptive algorithm will be introduced to adjust the parameters in the mathematical model of the controlled object to minimize the deviation between them. By introducing the parameter adaptive algorithm, when the parameters of the controlled object are perturbed, the algorithm can adjust the mathematical model of the controlled object in time to ensure the good control effect of the control system and enhance the robustness of the system.

Assume a discrete-time process model

$$y(t+1) = -a_1 y(t) + b_1 u(t) = \theta^T \phi(t) \quad (18)$$

where ϕ denotes observation vector of input and output, θ denotes unknown parameter vector.

$$\begin{aligned} \theta^T &= [a_1, b_1] \\ \phi^T &= [-y(t), u(t)] \end{aligned}$$

Consider a tunable parameter model as

$$\hat{y}^0(t+1) = \hat{y}[(t+1)|\hat{\theta}(t)] = -\hat{a}_1(t)y(t) + \hat{b}_1(t)u(t) = \hat{\theta}^T(t)\phi(t) \quad (19)$$

$\hat{y}^0(t+1)$ is the a priori prediction output of the estimated parameters at time t . $\hat{\theta}$ denotes estimated parameter vector at time t .

$$\hat{\theta}^T(t) = [\hat{a}_1(t), \hat{b}_1(t)]$$

Define the posterior prediction output as

$$\hat{y}(t+1) = \hat{y}[(t+1)|\hat{\theta}(t+1)] = -\hat{a}_1(t+1)y(t) + \hat{b}_1(t+1)u(t) = \hat{\theta}^T(t+1)\phi(t) \quad (20)$$

The prior prediction error and posterior prediction error are defined as

$$\varepsilon^0(t+1) = y(t+1) - \hat{y}^0(t+1) \quad (21)$$

$$\varepsilon(t+1) = y(t+1) - \hat{y}(t+1) \quad (22)$$

Parameter adaptation algorithms usually have the following structure:

$$\begin{bmatrix} \text{New estimated} \\ \text{parameters} \\ (\text{vector}) \end{bmatrix} = \begin{bmatrix} \text{Previous estimated} \\ \text{parameters} \\ (\text{vector}) \end{bmatrix} + \begin{bmatrix} \text{Adaptation} \\ \text{gain} \\ (\text{matrix}) \end{bmatrix} \times \begin{bmatrix} \text{Measurement} \\ \text{function} \\ (\text{vector}) \end{bmatrix} \times \begin{bmatrix} \text{Prediction error} \\ \text{function} \\ (\text{scalar}) \end{bmatrix}$$

The prediction error is also called adaptive error. This structure is also called an integral adaptive algorithm.

Parameter adaptive control involves adjusting the parameters of the process model adaptively, according to the error of the control object and the process model, by establishing an adaptive mechanism. In order to minimize the prediction error, it is necessary to find a recursive parameter adaptive algorithm with memory. You can use the current parameter prediction value plus a correction term to get the next prediction value. Its structure should be:

$$\hat{\theta}(t+1) = \hat{\theta}(t) + \Delta\hat{\theta}(t+1) = \hat{\theta}(t) + f[\hat{\theta}(t), \phi(t), \varepsilon^0(t+1)] \quad (23)$$

Correction $f[\hat{\theta}(t), \phi(t), \varepsilon^0(t+1)]$ contains known and predicted information. The correction term can be solved by defining an optimization index.

In order to be able to find a better way to solve Equation (22), the easiest way is to use the gradient method to solve for the estimated parameter $\hat{\theta}$ that minimizes prior prediction error $\varepsilon^0(t+1)$. However, when using the gradient method to solve Equation (22), it is very difficult to choose an appropriate adaptive gain. If the adaptive gain is too large, the parameter will oscillate around the minimum value; if the adaptive parameter is too small, the time it takes for the parameter $\hat{\theta}$ to converge to the minimum value will be too long. Therefore, in this article, the least-squares method is used to find the optimal parameter $\hat{\theta}$.

The least-squares criterion is defined as

$$\min_{\hat{\theta}(t)} J(t) = \sum_{i=1}^t [y(i) - \hat{\theta}^T(i)\phi(i-1)]^2 \quad (24)$$

where $\hat{\theta}^T(i)\phi(i-1)$ as

$$\hat{\theta}^T(t)\phi(i-1) = -\hat{a}_1(t)y(i-1) + \hat{b}_1(t)u(i-1) = \hat{y}[i|\hat{\theta}(t)]$$

First, a parameter θ needs to be estimated at time t in order to calculate the parameter θ that minimizes the sum of the squares of the difference between the output of the controlled

object and the output of the prediction model within t measurement ranges. Calculate the value that minimizes the criterion function (23):

$$\frac{\delta J(t)}{\delta \hat{\theta}(t)} = -2 \sum_{i=1}^t [y(i) - \hat{\theta}^T(t)\phi(i-1)]\phi(i-1) = 0 \quad (25)$$

From the above equation

$$[\hat{\theta}^T(t)\phi(i-1)]\phi(i-1) = \phi(i-1)\phi^T(i-1)\hat{\theta}(t)$$

Therefore, it can be obtained

$$[\sum_{i=1}^t \phi(i-1)\phi^T(i-1)]\hat{\theta}(t) = \sum_{i=1}^t y(i)\phi(i-1)$$

Make:

$$F(t)^{-1} = \sum_{i=1}^t \phi(i-1)\phi^T(i-1)$$

Assuming that $F(t)$ is invertible, multiplying both sides of the above equation by $F(t)$ can get:

$$\hat{\theta}(t) = [\sum_{i=1}^t \phi(i-1)\phi^T(i-1)]^{-1} \sum_{i=1}^t y(i)\phi(i-1) = F(t) \sum_{i=1}^t y(i)\phi(i-1) \quad (26)$$

To change the estimation algorithm to a recursive form, consider the $\hat{\theta}(t+1)$ estimate:

$$\hat{\theta}(t+1) = F(t+1) \sum_{i=1}^{t+1} y(i)\phi(i-1) \quad (27)$$

$$F(t+1)^{-1} = \sum_{i=1}^{t+1} \phi(i-1)\phi^T(i-1) = F(t)^{-1} + \phi(t)\phi^T(t) \quad (28)$$

From Equation (26), we can get:

$$\hat{\theta}(t+1) = F(t+1)[\sum_{i=1}^t y(i)\phi(i-1) + y(t+1)\phi(t)] \quad (29)$$

Considering Equation (25)

$$\hat{\theta}(t+1) = F(t+1)[F(t)^{-1}\hat{\theta}(t) + y(t+1)\phi(t)] \quad (30)$$

Therefore, the recursive formula of parameter a can be obtained

$$\hat{\theta}(t+1) = \hat{\theta}(t) + F(t+1)\phi(t)\varepsilon^0(t+1) \quad (31)$$

The recurrence formula of $F(t+1)$ remains to be derived from the recurrence formula $F(t+1)^{-1}$ given in Equation (20). It can be obtained by the matrix inversion lemma.

Lemma 1. (matrix inversion lemma): Assuming that F is a $(n \times n)$ non-singular matrix, R is a $(m \times m)$ non-singular matrix, and H is a $(n \times m)$ matrix, the following formula holds:

$$(F^{-1} + HR^{-1}H^T)^{-1} = F - FH(R + H^TFH)^{-1}H^TF$$

For Equation (27), let $H = \phi(t)$, $R = 1$, we can get:

$$F(t+1) = F(t) - \frac{F(t)\phi(t)\phi^T(t)F(t)}{1 + \phi^T(t)F(t)\phi(t)} \quad (32)$$

In summary, the parameter adaptive algorithm of the recursive least squares method is:

$$\begin{aligned}\hat{\theta}(t+1) &= \hat{\theta}(t) + F(t+1)\phi(t)\varepsilon^0(t+1) \\ F(t+1) &= F(t) - \frac{F(t)\phi(t)\phi^T(t)F(t)}{1+\phi^T(t)F(t)\phi(t)} \\ \varepsilon^0(t+1) &= y(t+1) - \hat{\theta}^T(t)\phi(t)\end{aligned}\quad (33)$$

3. Simulation Example

In order to verify the performance of the adaptive internal model control system and the stability of the reference adaptive algorithm, two simulation experiments were designed in this paper. One is a numerical simulation experiment for the reference adaptive algorithm, and the other is an experiment that applies the adaptive internal model control to the landing control of a fixed-wing UAV.

3.1. Simulation of Parameter Adaptive Algorithm

In order to verify the validity of the parameter adaptive algorithm, it is assumed here that a controlled process is:

$$y(k) = -1.5y(k-1) - 0.6y(k-2) + 2u(k-1) - 1.4u(k-2)$$

$$\theta^T(k) = [1.5 \quad 0.6 \quad 2 \quad -1.4]$$

$$\phi^T(k) = [-y(k-1) \quad -y(k-2) \quad u(k-1) \quad u(k-2)]$$

In the simulation experiment, a square wave signal with an amplitude of 1 is used as the input excitation signal, and additional zero-mean white noise is added as the input noise. The experimental results obtained are shown in Figure 4. In the figure, a1, a2, b1, and b2 are the four parameters in $\theta^T(k)$, respectively.

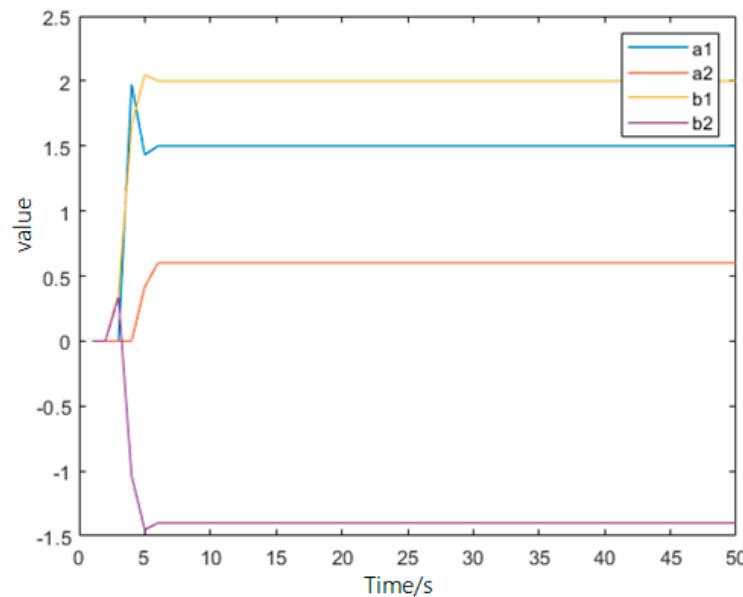


Figure 4. Parameter adaptive algorithm simulation results.

In this simulation, the simulation step size is 50 steps. It can be seen from Figure 4 that the convergence speed of the algorithm from the initial value to the real value is relatively fast, and it basically converges to the real value in the 7 steps.

3.2. Simulation of UAV Automatic Landing

In order to verify the effectiveness of the adaptive internal model control, it is applied to the automatic landing trajectory control of fixed-wing UAVs in this paper. The landing

process of fixed-wing UAVs can be divided into four stages: steep descent, shallow descent, touchdown, and slow down. The steep descent phase is characterized by a downward dive, that is, flying at a negative angle of attack. When the UAV descends to a certain altitude, in order to reduce the flight speed, it needs to fly at a positive small angle of attack until it touches the ground. Since the landing action is not a large deviation maneuver compared with the cruising action of the drone in the air, it is therefore possible to decouple the UAV mathematical model and design a linearized controller. Selecting the flight-level motion of the UAV as the reference state, the landing motion of the UAV can be decoupled into the longitudinal mode and the horizontal mode. Therefore, the longitudinal motion of the UAV represented by the state space model can be expressed as:

$$\dot{x} = H^{-1}Ax + H^{-1}Bu \quad (34)$$

where state vector $x = [\alpha \quad q]^T$, $H = \begin{bmatrix} V + \frac{L_\alpha}{m} & 0 \\ -c_7 \bar{M}_\alpha & 1 \end{bmatrix}$ is a Jacobian matrix, system state matrix $A = \begin{bmatrix} -\frac{L_\alpha}{m} - \frac{D}{m} & V - \frac{L_q}{m} \\ c_7 \bar{M}_\alpha & c_7 \bar{M}_q \end{bmatrix}$, system input matrix $B = \begin{bmatrix} -\frac{T_{\delta_T} \sin(\alpha_T + \alpha)}{c_7 \bar{M}_{\delta_t}} & -\frac{L_{\delta_e}}{c_7 \bar{M}_{\delta_e}} \end{bmatrix}$, input vector $u = [\delta_T \quad \delta_e]$. Where α is the angle of attack; q is the component of angular velocity at the body axis y ; \bar{M}_* is the moment component after linearization; c_7 is moment of inertia component; and δ_T δ_e are the throttle valve opening and the elevator declination angle, respectively.

In order to eliminate the steady-state error, AIMC is used as the UAV angular velocity loop controller. Its process model is Equation (34):

$$G_m = -\frac{q}{\delta_e} \quad (35)$$

The longitudinal control structure of the UAV landing is shown in Figure 5.

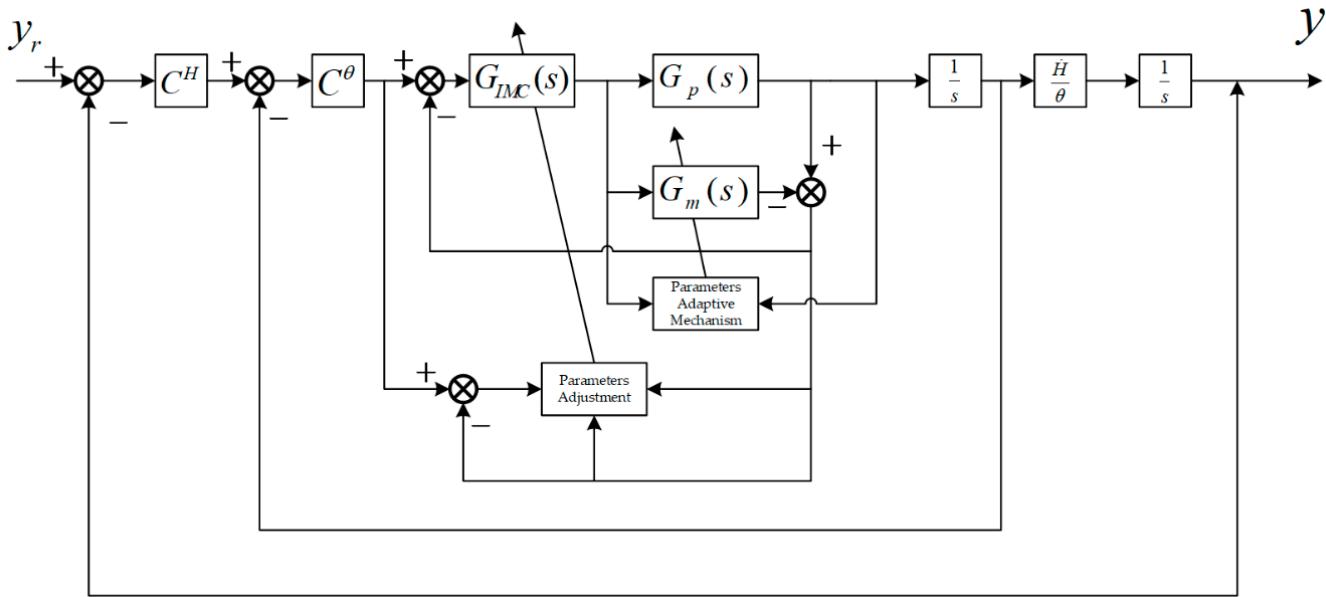


Figure 5. Control structure of longitudinal direction.

It should be noted that the pitch rate of the UAV is the control target of the inner loop control and the AIMC algorithm is applied in the inner loop. The outer loop of UAV control is pitch angle and height control, and the control method used in the outer loop is PID control. The parameter selection of PID control uses the Ziegler–Nichols closed-loop critical proportionality method to tune the PID parameters. In addition, Table 1 shows the simple parameters of fixed-wing UAV; Table 2 shows the parameters for tracking target

trajectory. Due to the limitation of experimental equipment and experimental site, it is difficult to deploy AIMC algorithm to real fixed-wing UAV. Therefore, this experiment was completed in the computer simulation software MATLAB/simulink. In addition, three different environmental conditions were set in the simulation as shown in Table 3. The first condition is no wind while in the second, the wind speed was $(6,10,0)$ (representation of wind speed in the north-east-geographic coordinate system). In the third condition, the wind speed is $(-6,-10,0)$.

Table 1. Nominal parameters in fixed-wing UAV.

Parameter	Value
Mean aerodynamic chord	$c_A = 0.54 \text{ m}$
Wing span	$b = 6.51 \text{ m}$
Wing area	$s_w = 3.11 \text{ m}^2$
Mass	$m = 200 \text{ kg}$
Cruise speed	$V_c = 120 - 200 \text{ km/h}$
Cruise altitude	$H_c = 4000 \text{ m}$
Endurance time	$T_e = 6 \text{ h}$

Table 2. Performances of standard track.

Parameter	Value
Arrival altitude	$H = 100 \text{ m}$
Steep sink rate	$V_z = 2 \text{ m/s}$
Shallow descend altitude	$H = 15 \text{ m}$
Shallow sink rate	$V_z = 0.5 \text{ m/s}$
Ground speed	$V = 45 \text{ m/s}$
Pitch angle of touchdown	$\theta = 0 \sim 6^\circ$
Wind speed	$V_d = (6, 10, 0) \text{ m/s}$

Table 3. State comparison under different environments.

Method	Wind Speed	Landing Coordinates	Air Speed	Ground Speed	Pitch
IMC	$(0,0,0)$	$(110,1.0,0)$	45	45	1.6
AIMC	$(0,0,0)$	$(0.3,0.6,0)$	45	45	1.6
IMC	$(6,10,0)$	$(12,3.5,0)$	40.5	45	2.5
AIMC	$(6,10,0)$	$(-27,3.2,0)$	40.5	45	2.5
IMC	$(-6,-10,0)$	$(206,-1.6,0)$	52	46	0.2
AIMC	$(-6,-10,0)$	$(25,-1,0)$	51.5	45.5	0.2

3.3. Simulation Analysis

Figures 6–9 are simulation diagrams of a fixed-wing UAV landing autonomously in a windless state. Figure 6 is the height control of the UAV, which is a comparison of the effects of the classical internal model control and the adaptive internal model control. It can be seen from the figure that when the height of the UAV under the control of the adaptive internal model decreases to zero, the lateral displacement correspondingly decreases to zero, while the UAV under the control of the classical internal model has a lateral displacement of a certain overshoot. It can be concluded that the UAV under the adaptive internal model control has a shorter landing distance.

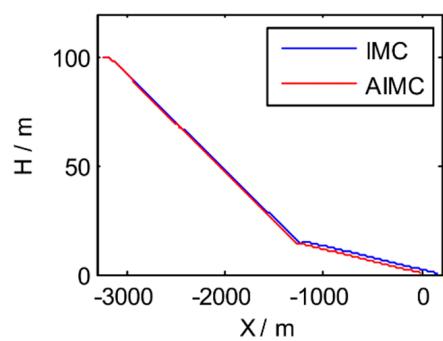


Figure 6. Height in windless condition.

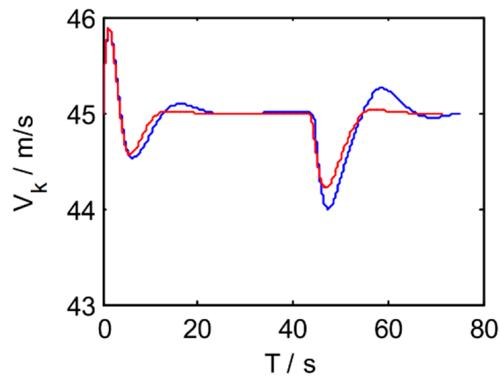


Figure 7. Air speed in windless conditions.

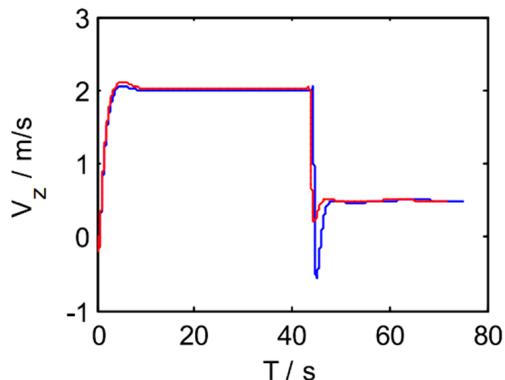


Figure 8. Landing speed in windless conditions.

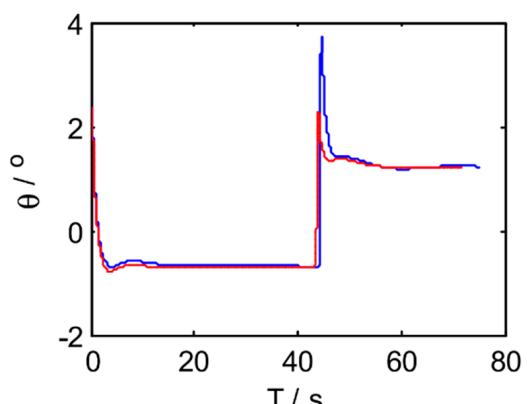


Figure 9. Pitch angle in windless conditions.

Figures 7–9 show the UAV speed after landing, the longitudinal speed during the landing process, and the pitch angle, respectively. It can be seen from the graphs of the longitudinal speed and pitch angle during the landing process that the adaptive internal model control has a smaller overshoot when there is a sharp change. This reflects the advantages of AIMC and also ensures the safety of drones.

Figures 10–13 are simulation diagrams of a fixed-wing UAV landing autonomously in a windward state. Figures 14–17 are simulation diagrams of a fixed-wing UAV landing autonomously in a headwind state. It can also be seen from Table 3 that the fixed-wing UAV under AIMC control has a smaller sideslip distance for lateral displacement

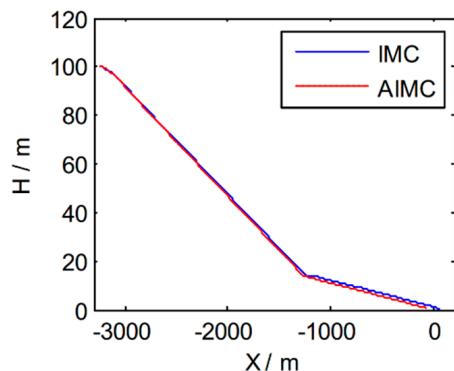


Figure 10. Hight in windward conditions.

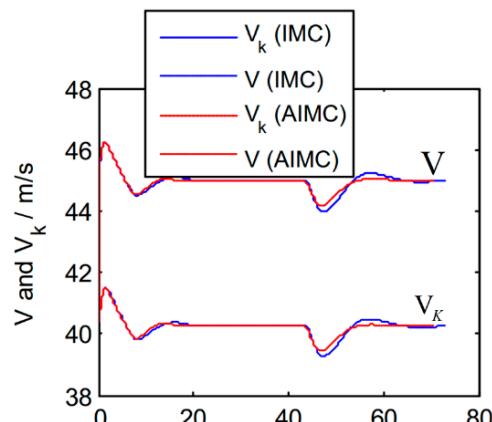


Figure 11. Air speed (V_k) and grand speed (V) in windward conditions.

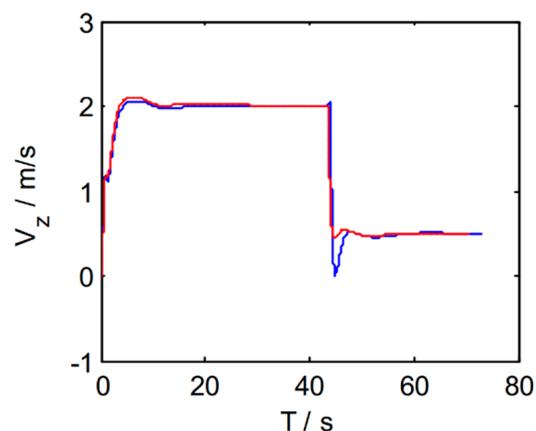


Figure 12. Landing speed in windward conditions.

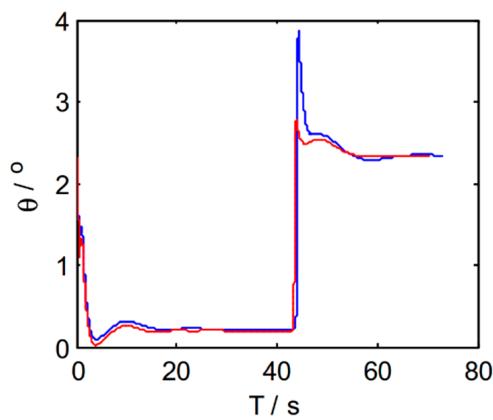


Figure 13. Pitch angle in windward conditions.

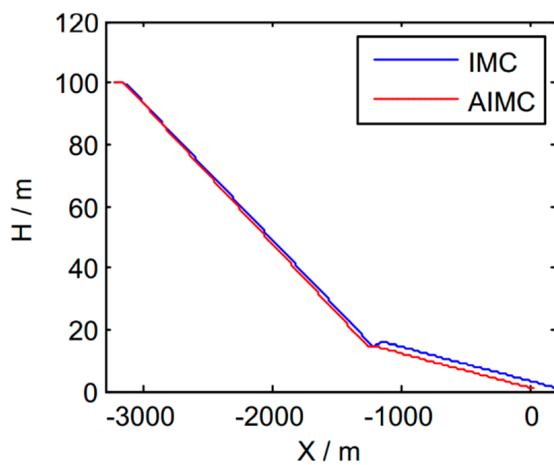


Figure 14. Height in headwind conditions.

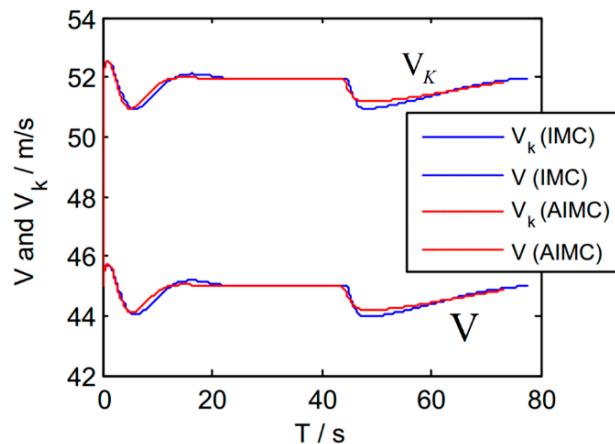


Figure 15. Air speed (V_k) and grand speed (V) in headwind conditions.

From the simulation results under three different environments, it can be seen that AIMC has superior performance to IMC. When the state changes, it can adjust to the target value faster, and has less overshoot.

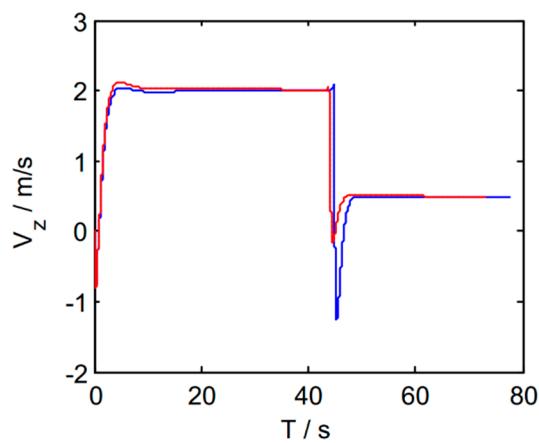


Figure 16. Landing speed in headwind conditions.

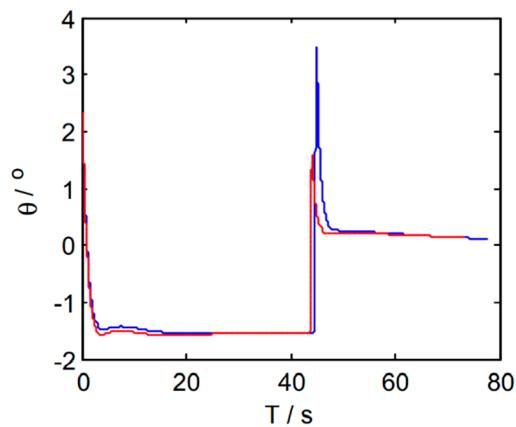


Figure 17. Pitch angle in headwind conditions.

4. Conclusions

This paper introduces the basic principles and advantages of internal model control in detail. From the basic principle of internal model control, we can see that internal model control has two shortcomings. One is the uncertainty of the controlled object model; the second is the selection of filter parameters. From the basic principle of internal model control, it can be seen that the ideal internal model controller is the inverse of the mathematical model of the controlled object, but since not all controlled processes are completely reversible, it will lose some control performance. Therefore, a low-pass filter is added to compensate the control performance. In addition, the selection of filter time constant will have a certain impact on the response time and dynamic performance of the system. Therefore, in order to solve these problems, this paper proposes two solutions. The first method is to adjust the time constant of the filter by adding an adaptive law. The adjustment principle is based on the product integration of system error and model uncertainty. The second method is to convert the internal model into an adjustable parameter model, and adjust the parameters in the parameter model through the parameter adaptive algorithm. When the parameters of the controlled object change, the control system will not lose control due to the increase of model mismatch. Through simulation, the improved AIMC has better performance than IMC. From the simulation results, it can be seen that AIMC can effectively restrain overshoot, reach the steady state faster, and have a smaller steady state error.

Author Contributions: Conceptualization, P.C., Y.Z. and J.W.; methodology, P.C., Y.Z., J.W., A.T.A., I.A.H., I.K.I., N.A.K. and F.A.A.; software, P.C., Y.Z., J.W., N.A.K., F.A.A.; validation, A.T.A., I.A.H., I.K.I., N.A.K. and F.A.A.; formal analysis, P.C., Y.Z., J.W., A.T.A., I.A.H., I.K.I., N.A.K. and F.A.A.; investigation, A.T.A., I.A.H., I.K.I., N.A.K. and F.A.A.; resources, A.T.A., I.A.H., I.K.I., N.A.K. and F.A.A.; data curation, P.C., Y.Z. and J.W.; writing—original draft preparation, P.C., Y.Z. and J.W.; writing—review and editing, P.C., Y.Z. and J.W.; visualization, A.T.A., I.A.H., I.K.I., N.A.K. and F.A.A.; supervision, A.T.A., I.A.H. and I.K.I.; project administration, P.C., Y.Z. and J.W.; funding acquisition, I.A.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research is funded by the Norwegian University of Science and Technology, Lars-gardsvegen, 2, 6009 Alesund, Norway.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors would like to acknowledge the support of the Norwegian University of Science and Technology for paying the Article Processing Charges (APC) of this publication. Special acknowledgement to Automated Systems & Soft Computing Lab (ASSCL), Prince Sultan University, Riyadh, Saudi Arabia. In addition, the authors wish to acknowledge the editorial office and anonymous reviewers for their insightful comments, which have improved the quality of this publication.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Garcia, C.E.; Morari, M. Internal Model Control. 1. A Unifying Review and Some New Results. *Int. J. Chem. Eng. Process. Des. Dev.* **1982**, *21*, 308–323. [[CrossRef](#)]
2. Tang, X.; Wang, D.; Wang, Y. Internal model control method based on virtual reference feedback correction method. *J. South China Univ. Technol.* **2016**, *44*, 66–72.
3. Ji, W.; Li, Q.; Xu, B. Design and Analysis of Robust Self-Adjusting Internal Model Control for Photoelectric Tracking System. *Opt. Precis. Eng.* **2009**, *17*, 2247–2254.
4. Liu, W.; Chen, L.; Chen, J. Adaptive Internal Model Control of Steering-by-Wire System Considering Delay. *J. Shanghai Jiaotong Univ.* **2021**, *55*, 1210–1218.
5. Raja, S.V.; Christian, B. Adaptive Internal Model-Based Harmonic Control for Active Torsional Vibration Reduction. *IEEE Trans. Ind. Electron.* **2019**, *67*, 3024–3032.
6. Xiaodong, S.; Zhou, S.; Long, C. Internal Model Control for a Bearingless Permanent Magnet Synchronous Motor Based on Inverse System Method. *IEEE Trans. Energy Convers.* **2016**, *31*, 1539–1548.
7. Qing, W.; Shinobu, Y. Design of HDD Servo Controller with Adaptive IMC Structure. In Proceedings of the International Conference on Mechatronics and Automation 2006, Luoyang, China, 25–28 June 2006; pp. 1280–1285. [[CrossRef](#)]
8. Jasmin, V.; Adnan, O.; Dinno, K.; Adnan, K. Adaptive Control of Hard Disk Drive Servo System. In Proceedings of the 2020 International Symposium ELMAR, Zadar, Croatia, 14–15 September 2020; pp. 91–96. [[CrossRef](#)]
9. Wahyu, D.; Reza, B.S.; Ngatelan; Budi, S. Design and simulation of PID controller based on internal model control. *J. Phys. Conf. Ser.* **2019**, *1175*, 012104.
10. Ren, F. *Application of Fuzzy Adaptive Internal Model Control in DCS*; Yanshan University: Qinhuangdao, China, 2011.
11. Jin, X.; Liu, J.; Niu, Y. Research on Application of Adaptive Internal Model Control in Main Steam Temperature Control System. *Chin. J. Electr. Eng.* **2003**, *10*, 225–229.
12. Tanaskovic, M.; Fagiano, L.; Smith, R. Adaptive receding horizon control for constrained MIMO systems. *Automatica* **2014**, *50*, 3019–3029. [[CrossRef](#)]
13. Zheng, J.; Fu, M.; Wang, Y.; Du, C. Nonlinear tracking control for a hard disk drive dual-stage actuator system. *IEEE/ASME Trans. Mechatron.* **2008**, *13*, 510–518. [[CrossRef](#)]
14. San, P.P.; Ren, B.; Ge, S.S.; Lee, T.H.; Liu, J.-K. Adaptive Neural Network Control of Hard Disk Drives with Hysteresis Friction Nonlinearity. *IEEE Trans. Control. Syst. Technol.* **2011**, *19*, 351–358. [[CrossRef](#)]
15. Xu, J.; Qinling, Z. Multivariable model reference adaptive control with application to flight control. *J. Beijing Univ. Aeronaut. Astronaut.* **2013**, *39*, 1048–1052. (In Chinese)
16. Astrom, K.J.A.; Wittenmark, B. *Adaptive Control*; Addison Wesley Longman Publishing Co., Inc.: Boston, MA, USA, 1995.
17. Rupp, D.; Guzzella, L. Iterative Tuning of Internal Model Controllers with Application to Air/Fuel Ratio Control. *IEEE Trans. Control. Syst. Technol.* **2010**, *18*, 177–184. [[CrossRef](#)]
18. Jin, Q.; Liu, Q.; Wang, Q.; Tian, Y.; Wang, Y. PID Controller Design Based on the Time Domain Information of Robust IMC Controller Using Maximum Sensitivity. *Chin. J. Chem. Eng.* **2013**, *21*, 529–536. [[CrossRef](#)]

19. Ju, H.-S.; Tsai, C.-C. Longitudinal auto-landing controller design via adaptive backstepping. *Int. J. Adapt. Control Signal Process* **2008**, *23*, 640–666. [[CrossRef](#)]
20. Rao, D.M.K.K.V.; Go, T.H. Automatic landing system design using sliding mode control. *Aerospace Sci. Technol.* **2014**, *32*, 180–187. [[CrossRef](#)]
21. Ndehedhe, C. Remotely Piloted Aircraft Systems. In *Satellite Remote Sensing of Terrestrial Hydrology*; Springer International Publishing: Cham, Switzerland, 2022; pp. 177–207.
22. Juang, J.G.; Chien, L.H.; Lin, F. Automatic Landing Control System Design Using Adaptive Neural Network and Its Hardware Realization. *IEEE Syst. J.* **2011**, *5*, 266–277. [[CrossRef](#)]
23. Mori, R.; Suzuki, S.; Sakamoto, Y.; Takahara, H. Analysis of Visual Cues During Landing Phase by Using Neural Network Modeling. *J. Aircr.* **2007**, *44*, 2006–2011. [[CrossRef](#)]
24. Salfi, M.I.; Ahsun, U.; Bhatti, A.H. Lateral and longitudinal guidance and control design of a UAV in auto landing phase. In Proceedings of the 6th International Bhurban Conference on Applied Sciences & Technology, Islamabad, Pakistan, 19–22 January 2009; pp. 162–168.
25. Alvaro, O.; Sergio, G.-N.; Raul, S. Comparative Study of Optimal Multivariable LQR and MPC Controllers for Unmanned Combat Air Systems in Trajectory Tracking. *Electronics* **2021**, *10*, 331. [[CrossRef](#)]
26. Farzaneh, D.J.; Farhad, S.; Mehrnaz, G. A Modified YOLOv4 Deep Learning Network for Vision-Based UAV Recognition. *Drones* **2022**, *6*, 160.