Advances in Computer, Signals and Systems (2022)
Clausius Scientific Press, Canada
DOI: 10.23977/acss.2022.060407
ISSN 2371-8838 Vol. 6 Num. 4

Embedded Dynamic Intelligent Algorithm in Computer Software Testing

Ying He*, Yan Chen

Sichuan Vocational and Technical College, Suining, Sichuan 629000, China
heyingt11@163.com
**Corresponding author*

Keywords: Embedded Dynamics, Intelligent Algorithms, Computer Software, Test Applications

Abstract: Since entering the digital information age, ARM embedded systems equipped with high-performance embedded processors have obtained huge development opportunities and ushered in the most beautiful market prospects in history. The purpose of this article is to study the application of embedded dynamic intelligent algorithms in computer software testing. First, it analyzes the research background of the subject, expounds the purpose and significance of the research from different angles, and introduces the main content. Secondly, a dynamic matrix control algorithm is introduced into the predictive control algorithm. Modeling is convenient, internal mechanism analysis is simple, and model accuracy is low. Due to the contradiction between static and dynamic performance, the contradiction between tracking set value and suppressing disturbance, and the contradiction between robustness and control performance, it is designed as a PID-DMC control system. A detailed analysis and summary of the previous theoretical results of software testing, computer software testing based on the characteristics of the embedded software itself, the experimental results show that the cascade DMC-PID main steam temperature control system runs stably.

1. Introduction

With the rapid development of microelectronics and information network technology and the improvement of capabilities, not only has it promoted economic growth and social development, but also has made industry changes increasingly attractive [1-2]. The demand for enterprise space continues to grow. With the advancement of network technology, a single application can only be managed, and it is difficult to meet the security and reliability requirements of business processes [3]. ARM-based embedded systems are widely used in many fields due to their low power consumption, simple CPU architecture, high performance characteristics, embedded network interface, and other power consumption advantages [4-5].

Increasingly scholars are studying dynamic intelligent algorithms. Bat Vibration Algorithm (DVBA) is a new optimization algorithm that has been tested in a wide range of international landscape projects. However, it has not been tested on truly global issues. Some experts have used DVBA to reduce supply chain costs and other well-known algorithms, namely swarm particle optimization (PSO), bat algorithm (BA), genetic algorithm (GA), and taboo detection (TS) [6]. Some scholars have proposed an optimal particle swarm optimization algorithm with automatic weight correction to solve the premature integration and minimum optimization problems of the existing bus system algorithms and build a deficit planning model with the lowest possible cost and shorter travel delays [7]. It is of practical significance to study the application of embedded dynamic intelligent algorithms in computer software testing [8].

This article embeds the adaptive dynamic programming algorithm for the control of the single-stage inverted pendulum into the controller and displays the control results through the graphical user interface, so that you can easily view the control effect, and you can modify the undesirable result until you are satisfied. until. In addition, when the control of the single-stage inverted pendulum is realized, some parameters of the ADP algorithm can be modified or some partial modifications of the algorithm can be made to achieve the control of other problems, so that a general intelligent controller can be realized.

2. Research on Embedded Dynamic Intelligent Algorithm in Computer Software Testing

2.1 Embedded System

The embedded system is an independent computer system, composed of an executive and an embedded system [9-10]. The built-in system consists of an application layer, a central layer, a software layer, and a functional layer (application layer); the executor is a controlled object that receives control commands issued by an embedded computer to complete specific tasks or tasks [11].

Hardware layer: The hardware layer is composed of built-in microprocessors, memory (ROM, SDRAM, Flash, etc.), a general application interface, and a center/O interface.

Intermediate layer: The central layer is located between the hardware layer and the software layer, also known as the support layer package or hardware abstraction layer, which separates the top software from the bottom layer [12].

System software layer: In a broader concept, the system software layer is between the hardware layer and the functional layer (application), and it is the core of the entire embedded system. The system software layer can accept the data or external signals transferred from the functional layer and transfer these data or external signals to the hardware layer for storage. It can also process and process the data or external signals with the solution stored in the memory. After passing through the system software layer to the functional layer for display or corresponding actions, it can be seen from the above process that the system software layer is the heart of the entire embedded system.

Functional layer (application software layer): For the integrated system to run well, the operating layer must provide simple operations and a very human-machine interface. The functional layer is mainly composed of developed applications and is oriented towards controlled objects and users.

Execution device: Executable equipment mainly refers to the object to be controlled. It can receive control commands and data sent by the embedded system, and then respond to these control commands to complete the specified functions or tasks and meet the required control. Generally speaking, in different application fields, the executable devices of the integrated computer system are different, and we must choose different executable devices and devices according to the requirements of the specific system or application.

2.2 Software Testing

Software testing uses testing tools to design, run, visualize, and securely test software products based on specific testing concepts and procedures, create various testing tools as needed, and design and maintain testing procedures to identify possible problems. Start testing after occasional analysis and evaluation, after creating test cases, errors need to be identified to ensure that the developed product meets the requirements. In IEEE software technology software testing is defined as: "The process of running or testing a program using simulation or automated methods, the purpose of which is to determine whether it meets specific requirements or explain the difference between the expectations we want. Results and actual results can be, It can be seen that no matter what it is defined, the basis of software testing is to "check the consistency of software and requirements."

2.3 Dynamic Intelligent Algorithm

The DMC (Rotary Matrix Control) algorithm is a dynamic intelligent prediction algorithm based on the nonparametric model of the control object's single-step response. Through feedback correction and rotation, the current and future control variables are optimized to make the output consistent with the predetermined trajectory. The strong matrix control is a dynamic intelligent predictive control algorithm, which uses the progressive response characteristics of the controlled object to characterize the power model of the system.

Observe the process of PID and DMC, enter the steady-state error density in the general predictive function index, restore the general predictive control rules according to PID, that is, decompose the algorithm into PID format, and check the predictive characteristics of PID. To use two PID control systems and a strong matrix control, PID control and DMC control are combined, the field function used for constraints in the corresponding segment is changed to PID format, and then a new PID control system is developed.

3. Investigation and Research on Embedded Dynamic Intelligent Algorithms in Computer Software Testing

3.1 System Construction

The platform hardware includes accessories such as system power supply (2N redundancy), main control unit, communication unit, air switch, and terminals.

The main control unit CPU: adopts a low-power controller with high computing performance and integrated Ethernet function, and does not use fan heat dissipation; it has strong online operation capabilities to meet the requirements of high precision, fast sampling and real-time control; adopt a modular and expandable model with equipment expansion capabilities.

The communication module CP: CPU conducts safe data interaction with the conventional control system in the form of MODBUS communication. The main control unit directly receives the process parameters from the controller of the conventional control system and issues instructions to the scene through the conventional control system.

DCS communication module: The communication module here is the EDPF-COMII module. This unit is an intelligent communication unit used to measure high-speed EDPF power distribution and control networks. There are four RS485 ports, two of which use HDLC protocol to communicate with EDPF-NT DPU, and the other two ports can be independently configured 103. This is very useful for the system to communicate with third-party MODBUS applications to ensure electronic safety.

3.2 Parameter Selection

The advanced rotation of the dynamic smart matrix is to increase the control of the P-error error through the algorithm, and let the algorithm record the progress of the control P. The time M determines the number of control variables of the strong matrix control algorithm. Therefore, the number of control time zones should be less than or equal to the value of the predicted time zone P in the M control variable.

If the P value is too high or too low, the system test results will become worse. When choosing the P value, the stability and speed of the system should be considered. The characteristics of the control system are very complex, and the response voltage of the system does not depend on the P predicted time point and the power of the system. In this case, the P value must be adjusted according to the system's requirements for stability and speed.

3.3 DMC Dynamic Intelligent Algorithm

The DMC algorithm is mainly based on the assumptions of the line simulation characteristics and the dynamic intelligent prediction model of the capture demonstration. As an online monitoring system with the best performance, it obtains the elastic retention function for rolling optimization, calculates the separation degree of search information in time, and feeds and corrects the separation information. The prediction model based on the step response is shown in Formula 1:

$$y(k) = \sum_{i=1}^{N} a_i \Delta u(k-i) \tag{1}$$

Among them, $u(k-i)-u(k-i-1)$ is the control increment acting on the system at k-i moment.

The goal of realizing feedback is to enable every step of the algorithm to be optimized according to the actual state of the controlled object. The actual results of the measurement system, real-time results of the measurement provide an accurate and reliable information basis for the system to obtain the best control concept. Frequently obtained information is not only the predictive behavior of the control system, but also the uncertain information unknown in the actual system.

Feedback correction is to use the actual error to correct as shown in Equation 2:

$$y_c(k+j) = y(k+j) + he(k),\ j = 1,2,\ldots,P \tag{2}$$

$y_c(k+j)$ is the new predicted value obtained after feedback correction.

4. Investigation and Analysis of Embedded Dynamic Intelligent Algorithms in Computer Software Testing

4.1 Development of DMC Controller for Embedded System

The overall design scheme is shown in Figure 1, which can be roughly divided into the server part, development board part, and controlled object part. And the main development board part includes three links of A/D converter, D/A converter, DMC algorithm.

First, the user sets the parameters open loop gain K, time constant T, delay time τ, and set value through the graphical interface of the upper computer. After setting, we start to run the lower computer program, the A/D converter collects the output of the controlled object in real time, calculates the control quantity u through the DMC algorithm, and converts the control quantity into an analog signal through the D/A converter, which acts on the controlled object, the output of the controlled object is also sent to the graphic interface of the upper computer, and the output curve is displayed in real time.

While running the program, threads are also added to access the server. When the connection to the server is successful, you can accept the parameters and set values sent by the server, change the local parameters and set values in real time, and no longer accept the control of the machine. When the connection with the server is disconnected, it continues to accept local control.

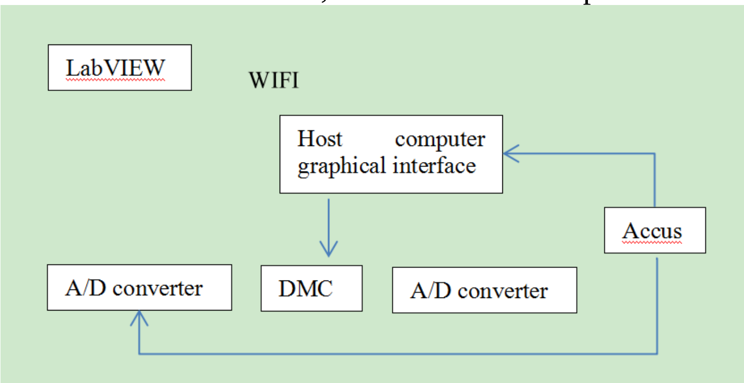

Figure 1: Design flow chart

4.2 Real-time Curve of Main Steam Temperature

Load the initial condition operation on the DCS side, and at the same time switch the PLC to the RUN state for operation. Open the real-time trend icon on the appbar to display the real-time trend curve. Drag the observation point to the real-time curve window to observe its real-time value change. The main steam temperature controlled object is selected, and the transfer function of the main steam temperature object is identified to obtain its dynamic characteristics, to adjust the parameters of the predictive control. According to the tuning prediction controller, the main steam temperature controlled object is predicted and controlled, and the upper and lower limits and time axis coordinates of the real-time curve interface are adjusted to obtain real-time data, as shown in Table 1.

Table 1: Constant disturbance of cascade DMC-PID main steam temperature control system

	Main steam temperature in real time	MATLAB simulation
10	40%	30%
20	41%	31%
30	48%	38%
40	45%	35%
50	47%	37%
60	1%	1%

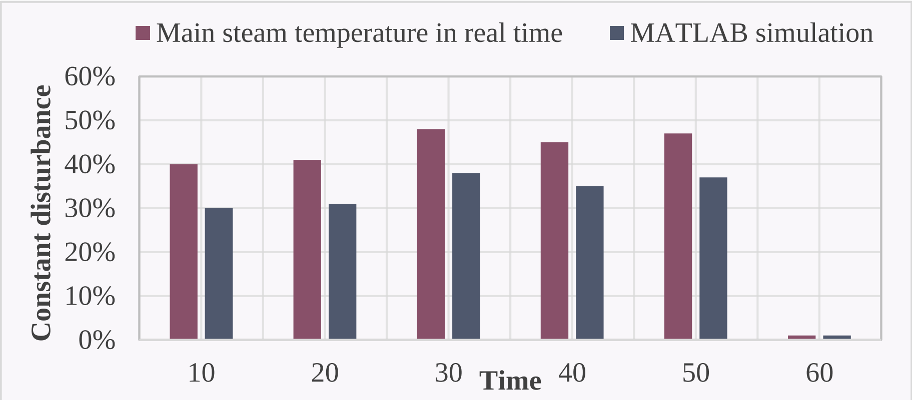

Figure 2: Constant disturbance test diagram of cascade DMC-PID main steam temperature control system

It can be seen that the real-time curve of the main steam temperature in the project is basically consistent with the MATLAB simulation curve, as shown in Figure 2, the control effect is also very good, the system runs stably, and satisfactory results can be achieved.

5. Conclusions

The rapid development of embedded technology has also accelerated the development of control technology. It has become a habit to integrate some advanced control technologies and control algorithms into the controller. Analyze the information, describe the advantages and disadvantages of the existing software test model, combine the characteristics of the built-in military control software, and create the built-in military test software. Analyzed the advantages and disadvantages of current designers testing in the real world, combined with the characteristics of embedded systems, created a software simulation software embedded embedded component. Based on the general understanding of the information and analysis of the controller of the dynamic intelligent automation algorithm, the graphical user interface is designed to determine the friendly and flexible interaction between the operator and the basic algorithm system.

References

[1] Tamilselvan K, Thangaraj P. Pods-A Novel Intelligent Energy Efficient and Dynamic Frequency Scalings for Multi-Core Embedded Architectures in an IoT Environment. Microprocessors and Microsystems, 2020, 72(Feb.):102907.1-102907.11.

[2] Divya G, Gummineni M. Implementation of FPGA based Decision Making Engine and Genetic Algorithm (GA) for Control of Wireless Parameters. Advances in Computational Sciences and Technology, 2018, 11(1):15-21.

[3] Lin G. Solving Hardware/Software Partitioning via a Discrete Dynamic Convexized Method. Wuhan University Journal of Natural Sciences, 2019, 24(004):341-348.

[4] Zhong Q, Liu Y, Guo X, et al. Dynamic Obstacle Detection and Tracking Based on 3D Lidar. Journal of Advanced Computational Intelligence and Intelligent Informatics, 2018, 22(5): 602-610.

[5] Kyrkou C, Bouganis C S, Theocharides T, et al. Embedded Hardware-Efficient Real-Time Classification With Cascade Support Vector Machines. IEEE Transactions on Neural Networks & Learning Systems, 2017, 27(1):99-112.

[6] Li Y, Dong W, Yang Q, et al. Automatic Impedance Matching Method With Adaptive Network Based Fuzzy Inference System for WPT. IEEE Transactions on Industrial Informatics, 2020, 16(2):1076-1085.

[7] Huang H C, Lin S K. A Hybrid Metaheuristic Embedded System for Intelligent Vehicles Using Hypermutated Firefly Algorithm Optimized Radial Basis Function Neural Network. IEEE Transactions on Industrial Informatics, 2019, 15(2):1062-1069.

[8] Shen Z, Han Y, Ling Y, et al. Dynamic Generation of Internet of Things Organizational Structures through Evolutionary Computing. IEEE Internet of Things Journal, 2018, PP(99): 1-1.

[9] Qin H, Zawad S, Y Zhou, et al. Reinforcement-Learning-Empowered MLaaS Scheduling for Serving Intelligent Internet of Things. IEEE Internet of Things Journal, 2020, 7(7): 6325-6337.

[10] Vavrek J, Viszlay P, Lojka M, et al. Weighted Fast Sequential Dtw for Multilingual Audio Query-By-Example Retrieval. Journal of Intelligent Information Systems, 2018, 51(2): 439-455.

[11] Rana M M. IoT-Based Electric Vehicle State Estimation and Control Algorithms Under Cyber Attacks. IEEE Internet of Things Journal, 2019, PP(99):1-1.

[12] F Angeletti, Ga Sbarri P, Sabatini M. Optimal Design and Robust Analysis of a Net of Active Devices for Micro-Vibration Control of an On-Orbit Large Space Antenna. Acta astronautica, 2019, 164(Nov.):241-253.