Journal of Applied Science and Engineering Innovation, Vol.6, No.1, 2019, pp. 36-38
ISSN (Print): 2331-9062
ISSN (Online): 2331-9070

Study on the Open Architecture of Industrial Robot Control Systems

Yifan Zhang
Automation and Electronic Engineering Institute, Qingdao University of Science & Technology, Qingdao, Shandong, 266071, China.

Abstract*:* As an important part of industrial robots, control systems restrict the development of robot technology to some extent. This paper, firstly, it summarizes the research of open robot control system, and focuses on the structural characteristics of industrial robot control system and the open software and hardware implementation methods of the system. Then it also summarizes the application status of open control system in multi-robot collaborative control, and last the future development direction of the control system is forecasted.

Keywords Robot Control System, Industrial Robot, Open Architecture

INTRODUCTION

With the continuous improvement of the world's technological level, the powerful vitality of industrial robot technology is becoming more and more significant. It combines human and machine expertise to form an intelligent mechanical device. [Michalos, *et. al.*, 2015] The robot can be reprogrammed, human-like, automatic, and multi-degree-of-freedom positioning devices that can perform various functions in three-dimensional space. Because it is difficult to accurately obtain the motion model of the robot system, and its control implementation is also very complicated, how to effectively control the robot system with uncertainty becomes a research hotspot of current robot technology. [Martinova, *et. al.*, 2015] At present, the main problem facing the robot control problem is the complexity of the robot dynamics model and the uncertainty of the system itself. [Krüger, *et. al.*, 2017]

OVERVIEW OF ROBOT CONTROL SYSTEM

The robot system is mainly composed of three parts: the execution part, the detection part and the control system. The execution part of the robot mainly has mechanical parts such as mechanical arms, wrists and fingers and their actuators; The detection part refers to various types of sensors that acquire the controlled object, the dynamic information of the robot itself and the surrounding working environment. [Vick, *et. al.*, 2015] The control system, as the core of the robot system, is controlled by a certain control algorithm according to the feedback information of the detection part. The part that issued the command to complete the desired operation. The robot is a complex system with strong coupling, high nonlinearity and great uncertainty. It is very difficult to achieve effective control of the robot. Usually, the excellent control of the robot control technology directly affects the advanced or not of the robot system. The strength of the function, therefore, the design of the control system is an important part of the level of robot technology.

For robot control problems with modeling errors and external disturbances, the main control methods of robots are:

Traditional control methods commonly used in robot control problems include PID control, feed forward control, and calculated torque. Most of these methods are simple and easy to implement. The development of traditional control methods is relatively mature, and it also plays an important role in practical applications. The control algorithm of this method is mainly represented by robust control and sliding mode variable structure control. Robust control is a control method that can ensure the stability of an uncertain system and achieve satisfactory control effects. Sliding mode control is a special kind of nonlinear control. The basic idea is to find a hyperplane in the state space of the error system, and all the state trajectories in the hyperplane finally converge to zero. In terms of intelligent control, the most popular methods at present are adaptive control, neural network control, fuzzy control, and genetic algorithm control. The above four control methods have obvious advantages, but in the control design of the robot system, many of them are combined with other control methods, and the advantages are complementary, so that some mixed control methods appear.

OPEN ROBOT CONTROL SYSTEM

At present, the openness of the robot control system is not strictly defined. According to the IEEE's official definition of “open”, the open system should satisfy the application of the system and can be transplanted between different platforms, interacting with other application systems to provide users with Consistent interaction. Lothar Rossol, founder of the KUKA Robot Group, defines an open system as: Running on a commercial, standard computer and operating system with open software and hardware interfaces. For an open controller, it should have a modular, interface-standard open structure, so users don't have to understand the internal structure of the robot, and only a few simple robot knowledge can operate the robot. Once the process changes, the system can be modified to meet new applications with minimal cost and as little time as possible. The robot system is mainly composed of three parts: the execution part, the detection part and the control system. The system management and communication module is mainly responsible for communication with the host computer and the module, and manages the information scheduling and sharing of the entire system, as well as teaching programming, and communicates with the host computer through the M CP2515 independent CAN controller of the SPI interface; The data processing module mainly initializes the motion control board and the servo controller, and processes the input data and parameters and transmits them to the system management and communication module; the motion control and the trajectory interpolation module are responsible for completing the trajectory interpolation algorithm of the robot according to requirements.[Wang, *et. al.*, 2016] Including linear, arc, S-curve interpolation, and real-time feedback of position and velocity to dynamic tracking and detection module, and real-time correction of trajectory and interpolation path according to data processing module; dynamic tracking and detection module mainly dynamic real-time tracking of each axis. The corner and position data buffers are fed back to the system management and data processing module to obtain the current position coordinates of the robot's end for real-time adjustment of the interpolation trajectory.

The overall architecture model of the dual-core real-time system based on uC linux and RTA I mainly includes the basic kernel uC linux, the real-time kernel RTA I, the hardware abstraction layer RTHAL, and the SMP-860 motion control board based on S3C 44B0X. The basic kernel (uC linux) and the real-time kernel (RTA I) handle the scheduling and execution of non-real-time and real-time tasks, respectively, while the exchange of information between real-time tasks and non-real-time tasks is done through pipes (FIFO) or shared memory (MBUFF). to fulfill. When the real-time task runs, the base kernel is blocked by the hardware abstraction layer. That is, the real-time kernel runs the basic kernel as the lowest priority task, and only schedules it when there is no real-time task running. The real-time kernel RTA I is dynamically loaded. Only when the real-time task is activated, the basic kernel is taken over by the real-time kernel. Otherwise, the basic kernel transparently accesses the hardware through RTHAL, just as RTHAL does not exist. The basic kernel interrupt and real-time interrupt are processed by scheduling. When the interrupt arrives, the real-time kernel determines whether it is a basic kernel interrupt or a real-time interrupt, and processes it separately. At the same time, real-time interrupts can preempt the execution of basic kernel tasks.

Many experts put forward their own opinions on how to implement an open control system. K. University of London Nilson and Rolf Johanssonb proposed a hierarchical structure of the control system, and based on this, proposed the structure of the open robot controller. So far, many countries or institutions have funded research from the concept of open control systems, and three of them have more influential plans: The OSACA (Open System Architecture for Control with Automation System) It was jointly proposed by some European research institutes and controller manufacturers, and successfully completed the development of the entire standard system platform in 1996; some large mechanical and electromechanical manufacturers in Japan also jointly proposed the OSEC (Oregon Solar Energy Conference) plan in 1994. The ambitious goal is to develop a standard for automation companies around the world; the United States also proposed the OMAC (the Organization for Machine Automation and Control) program in 1994. Unlike the previous two plans, the control system proposed by OMAC has a standardized interface layer, which users can tailor according to their own needs. A series of specific function modules with the concept of "plug and play", the specific function modules can be easily connected to the control system. With the rapid development of computers, research on robot control software with platform versatility has also made corresponding progress. In 2001, the Jet Propulsion Laboratory designed the autonomous robotic system CLARAty. In 2003, the robot control software CARMEN developed by Carnegie Mellon University made the experiment and simulation of control algorithms easier. The open source software project Orca proposed in 2006 realized cross-platform development. In 2007, Stanford University proposed an open source ROS system (robot operating system) and cooperated with many universities and companies, such as MIT, Samsung and so on. In 2010, Willow Garage released the open source robotic operating system ROS, which is open source and uses a distributed processing architecture (also known as Nodes), which allows executables to be designed separately and loosely coupled at runtime.

The advantage of this system is that it is not constrained by the robot hardware. The system itself has many robot control algorithms and instructions that can be applied to different robots. Although there is currently no strictly open control system in the strict sense, research on this aspect has made great progress.

There are two main types of hardware structures for open control systems: PC-based and VME (Versa Module Eurocard)-based systems. PC has the advantages of low cost, openness, complete software development environment and good communication functions. At present, some large robot manufacturers regard PC-based robot control system as the main development object. In the "PC + Motion Control Card" mode, the PC is mainly responsible for the human-computer interaction interface, and the motion control card is responsible for kinematics solution and trajectory interpolation calculation. In this mode, the PC can run on Windows or Linux systems, while the motion control card provides standard interface functions in Visual C++ and Qt environments, which is convenient for users to program with standard software. The research carried out by this model is relatively early. Luo Weitao et al. combined ARM, DSP and FPGA to design a robot motion controller, which can realize reliable motion control of the robot. In the United States, Seok S, Hyun DJ, etc. developed the control system platform system of multi-degree-of-freedom robots by using the advantages of modern parallel real-time computing technology such as multi-core CPU, FP GA and distributed processor, so that the robot can realize Dynamically move and interact quickly with the environment. The Robotics Research Institute of Harbin Institute of Technology developed the hardware control structure of PC + DSP + FPGA and applied it to the satellite teleoperation four-degree-of-freedom manipulator ware resources used are rich, and open, but there are also disadvantages of high requirements on the control card. Therefore, a DSP is needed to improve the data processing speed of the motion card. The upgrade cost of the controller is high and the motion control card does not currently have a uniform interface standard.

CONCLUSION

This paper proposes the implementation methods of several mainstream open control systems and analyzes their characteristics. The development trend of open control system in the future: Committed to the development of general modular structure software platform, the platform should adopt international normative standards to design software interface, users can conveniently carry out secondary development on the above, realize software portability; based on real-time Industrial robot open control system with operating system and high-speed bus; Industrial Ethernet has broad application prospects in the production workshop. The open control system will develop towards the network in the future; it is committed to realize the seamless connection between different functions of the robot system. Reduce the difficulty of robot system integration and improve the real-time and compatibility of the robot control system software system.

REFERENCES

Krüger J, Wang L, Verl A, et al. Innovative control of assembly systems and lines[J]. CIRP annals, 2017, 66(2): 707-730.

Martinova L I, Sokolov S S, Nikishechkin P A. Tools for monitoring and parameter visualization in computer control systems of industrial robots[C]//International Conference in Swarm Intelligence. Springer, Cham, 2015: 200-207.

Michalos G, Makris S, Chryssolouris G. The new assembly system paradigm[J]. International Journal of Computer Integrated Manufacturing, 2015, 28(12): 1252-1261.

Vick A, Vonásek V, Pěnička R, et al. Robot control as a service—towards cloud-based motion planning and control for industrial robots[C]//2015 10th International Workshop on Robot Motion and Control (RoMoCo). IEEE, 2015: 33-39.

Wang L, Haghighi A. Combined strength of holons, agents and function blocks in cyber-physical systems[J]. Journal of manufacturing systems, 2016, 40: 25-34.