

Modifying Parameters Online for Real-time Simulation of UAV Flight Control Sensor System Based on RTX

¹ Duan Xiaojun, ² Yu Changgui, ¹ Tian Zongchao, ¹ Wu Kai, ¹ Wang Zhe

¹ National Laboratory of UAV Special Technology, Northwestern Polytechnical University, Xi'an 710065, China

² Technology Center, AVIC Chengdu Aircraft Industrial Co. LTD, Chengdu, 610092, China

¹ E-mail: xiaojunduannpu@gmail.com

Received: 28 February 2014 / Accepted: 29 March 2014 / Published: 31 March 2014

Abstract: This paper brings forward a new technique to modify parameters online achieved by RTW for real-time simulation of UAV flight control sensor system, because RTX is unable to modify parameter online directly. The interface is C-API. By collecting model parameters and signal information, we set several programs in the simulations target to modify sensor model parameters about UAV flight control sensor system, including INS, vertical gyroscope and magnetic heading sensor. The simulation host transfers data with the simulation target by using shared memory. Then the main process achieves the function of modifying parameter online, thus improving the efficiency of simulation. *Copyright © 2014 IFSA Publishing, S. L.*

Keywords: Flight control sensor system, Modify parameters online, Real-time simulation, RTX.

1. Introduction

Using Matlab Simulink modeling toolbox to execute graphical modeling and simulation can not only simplify the control system design and simulation process, but also improve visualization and simulation integration. But the simulation of Simulink is not a real-time simulation. For this Matlab provides RTW (Real-Time Workshop) to achieve real-time simulation function. RTW can be generated optimized, portable and customizable ANSI C code from Simulink model. Using it can produce real-time code running under RTX. In order to verify the correctness of the model and optimize the model during real-time simulation processing, we need to change parameter constantly. For non-real-time simulation, changing the parameters only needs

to open the corresponding model that parameters can be modified. But it has too much trouble to modify parameters in real-time simulation model based on RTX because the model has been transformed to C code automatically. We often need to stop the simulation, and after RTW generates code, we cannot adjust parameter during the simulation. So, we propose using the RTW provides data exchange interface C-API for collecting model parameters and signal information to adjust parameter online. Simulation model parameters and signal information can be sent to external applications under Windows.[1] Meanwhile the external application can also be carried out depending on user needs for achieve online parameter adjustment in the process of simulation.[10] We can greatly improve simulation efficiency, and can achieve fault injection functions.

2. RTW Real Time Toolbox

a) RTW Profile.

RTW is an extension function of Simulink. It can be automatically generated, packed, and compiled code from the model. By configuring RTW, we generate C code running under real-time model for different real-time operating system. We could provide an implementation approach for the Simulink model to run in real-time hardware system. RTW-based real-time simulation can greatly reduce development costs and shorten the development cycle [2].

b) RTW code generation process.

RTW code generation and creation process need to combine the system target file with template files. The system target files contain information about the target platform and develop the corresponding template file, and describe particularly how the user to select the target platform to generate code. The document consists of three parts: 1) the annotation section; 2) rtwoptions structure configuration; 3) code generation. The annotation part mainly supplies templates files and code generation commands used during code generation. The rtwoptions structures are arranged mainly to complete the RTW configuration interface. And the code generation configuration specifies the path of code generation. On the other hand, RTW provides an inner template building file used to create a specific program. [12]The template building file gives the information of target platform and the simulation model that can guide binder program to compile and build programs from the source, library files, and user-defined modules. According to the system template file system.tmf, RTW could generate program binder file model.mk. Template building file is designed for a specific target environment platform [4].

For simplicity, we can build our own template file by changing an existing template. So we could make a personalized program creation process. After modifying system files and templates build target files, we can generate real-time applications that running under the target platform. If we want to exchange data with the other program, we must also modify the main framework document grt_main.c to achieve signal monitoring or online parameter adjustment.

3. C-API Data Exchange Interface

a) C-API documentation generation process

In practical applications, we often need to exchange data with some signals or parameters of simulation model that running under the RTX, which are generated form RTW. Its applications include signal monitoring display, parameter optimization changes and so on. By using the data interface C-API provided by RTW, we could change parameters, display signals easily when it is running on the

Windows client application. [8] When we set up the C-API data interface, RTW could generate another two files model_capi.c and model_capi.h during simulation model compiled process. "Model" represents the model name. These two documents are located at the compiled path set in RTW. C-API source files model_capi.c includes information of global parameters and signals which are located at model generation files. C-API header file model_capi.h is the interface of model source files with C-API files. C-API function code generation process is shown in Fig. 1.

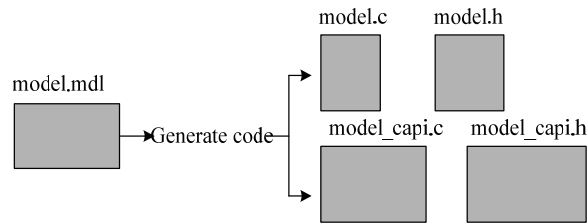


Fig. 1. C-API code generation process.

C-API file model_capi.c and model_capi.h provide the interface of external applications and simulation module. According to different needs, the data could be data of signal or parameter information. Each parameter or signal information is contained in the structure which include data names, data path in the module, data address, data type information, data dimensionality (scalar, vector or matrix), and the sampling time (only for signaling). The data type and data dimensions are stored in the form of index. According to the index values, we can get the data type in the dimensionality of the data structures and search for a specific data type and dimension. In this way, we could save memory space [9].

b) C-API data structures

The file of model_capi.c contains information of global signal and parameter generated by RTW, which is mainly stored in several important structures, such as the signal structure rtwC_API_Signals, module parameter structure rtwC_API_BlockParameters, model parameter structure rtwC_API_ModelParameters, data type structure rtwC_API_DataTypeMap, and the data dimension structure rtwC_API_Orientation. These structures are defined in matlabroot \ rtw \ c \ src \ rtw_capi.h. These structures are described below.

Signal structure rtwC_API_Signals: signal structure describes all the information of a signal defined in model. There are index of address, number of system identification, the module path, the signal name, port number, the data type index, the data dimension index, fixed-point index, and the sampling time index.

Module parameter structure rtwC_API_BlockParameters: describes specific information of parameter in module. There are

address index, the path of block parameters, parameter name, data type index, parameter dimension index, and fixed-point index.

Model parameter structure `rtwCAPI_ModelParameters`: is similar to the structure of module parameters. But there is no block parameter path, because the parameter is within model.

Data type structure `rtwCAPI_DataTypeMap`: describes data types of signal or parameter. There are data types defined in C language, data types defined in RTW, number of simple data included in the whole data, index in `rtwCAPI_ElementMap`, data types cited in Simulink, and whether it is a complex or a pointer.

Dimensionality of the data structure `rtwCAPI_Orientation` uses the enumerating form for definition. It includes scalar, vector, matrix with rows bigger than columns, and matrix with column bigger than rows.

In addition, `rtwCAPI_ModelMappingInfo` is an important structure to achieve online parameter adjustment. It is defined in the `model.h` and it covers all the information related with signals and parameters of the model. According to differences of models, C-API source file `model_capi.c` also generate the corresponding structures. So a mapping relationship is established between the simulation model and the C-API document.

4. Parameter Adjustment Online

a) The process of parameter adjustment online.

Commands of online parameter adjustment are produced by the host. The host receives model parameter information from the target side, and then gives the command of online parameter adjustment [3]. At the same time, the index of the parameter to modify the address and the new value are sent to the target machine. The target machine receives the instruction. The instruction is analyzed first, and if it is online parameter adjustment, then a new program will be created with the online parameter adjustment function to adjust parameter online. Process of parameter adjustment online shown in Fig. 2.

b) The implementation of online parameter adjustment.

As mentioned above, in order to achieve online parameter adjustment, the key is to get the pointer which is pointed at model structure in the main function, and then get adjustable parameter information. The parameter information is sent to the host side, and the parameter information will be displayed on the host side. Meanwhile on the host side we can get a real-time program after compiling and loading. After observing the changing of requested signal, we can optimize parameters according to the signal change. The parameter change commands will be sent via UDP port. When the main program receives the parameter adjustment command, it calls parameter adjustment function for

online parameter adjustment. The premise of online parameter adjustment under RTX is to get information stored in the model structure `rtwCAPI_ModelMappingInfo` [6].

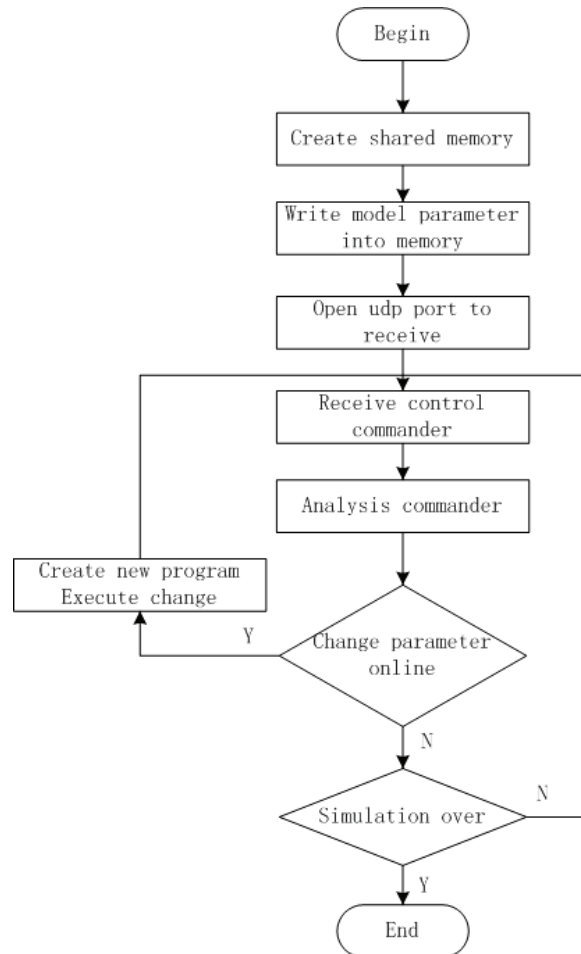


Fig. 2. Online Scheduling Process.

After get the pointer of information structure that points to the model structure, we have actually got all the signals and parameter information. And then we can get the adjustable parameter information by invoking the macro function, and store it in the shared memory, and then display it on the host. We change parameters according to different needs. Some of the key functions are shown below:

- 1) `GetTunableParaNum`: gets the number of adjustable parameters.
- 2) `RtCreateSharedMemory`: creates shared memory to store adjustable parameter information.
- 3) `FillParaInfoArray`: stores the obtained parameter information in `ONE_PARA_INFO` structure. The structure contains the index of the parameter address, the path name of the block, parameter names and parameter values.

In the model, each parameter has a unique address index of parameter, parameter adjustment online is using this only address index of parameter to find the parameters. When simulation is running, the target

machine receives the command to change the parameters sent from the host computer and extracts the address of the index and the values you want to update. And then we check module address index of each parameter [11]. If a parameter address index is the same with the received address index, then we change this parameter value corresponding to the index to the new value which is received. Since the structure of storage model is very complex, we can divide online parameters adjustment into two steps. The first step is to look up the address index values of the parameter which need to be modified. Based on the index corresponding to parameter types, the received new value will be converted into the requested type of data. Since the data is different in dimensions, the assignment process will also be different. For example, a scalar assignment only needs one statement, vector assignment is complex relatively. So in the second step we can use the subroutine for data assignment depending on different dimensionality. Meanwhile we can improve program modularity and enhance readability [7]. When the simulation model is relatively complex, there may be more parameters and signals. So when searching the parameter index it will take a long time and affect the normal conduct of real-time simulation. In order to achieve the purpose of real-time, we can open a new high-priority program. So when receiving the command of parameter adjustment online, it can immediately go to the new program to perform online parameter adjustment to ensure the normal operation of real-time programs. The main program of changing parameters is shown in Fig. 3.

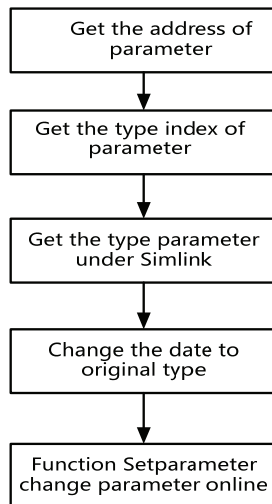


Fig. 3. Online parameter adjustment process.

5. Application and Validation

To verify the online parameter adjustment environment, we can build a simple simulation model in Simulink. After the model was building, we select

the data exchange interface as C-API mode in the simulation parameter configuration interface, configure signals and parameters, and set the simulation time for 20s. Then we compile the simulation model to generate the program with suffix. rts running under RTX [5]. And then we load the model in the host side and click “Run”. When the program runs to 8s we choose a square signal. We can see the parameters of the signal amplitude, period and pulse width. We modify any of the parameters in the modulation parameters shown in Fig. 4.

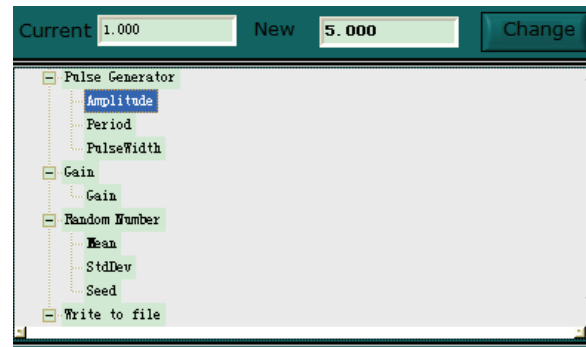


Fig. 4. Online parameter adjustment interface.

We click the amplitude of the parameters and can see the amplitude of the current parameter is 1. Then we type the new amplitude 5 in the text box. After that we click “Change parameter”, and we can see the change of a square signal shown in Fig. 5.

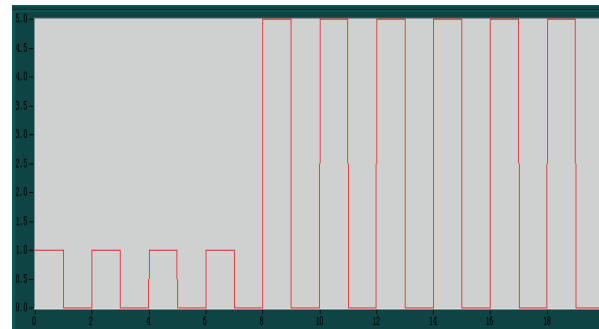


Fig. 5. Shows the simulation results online parameter adjustment.

6. Conclusions

By the C-API data interface generated from RTW, we can have access to the signal and parameter information of the simulation model. We modify the main frame files and control simulation process to achieve the simulation process of online parameter adjustment. In this way, we solve the problem of code generation and compilation which are caused by parameter variations under RTX. And we also achieve the real-time online parameter adjustment

function. It provides a great convenience to the parameter adjustment of large and complex flight control simulation system.

References

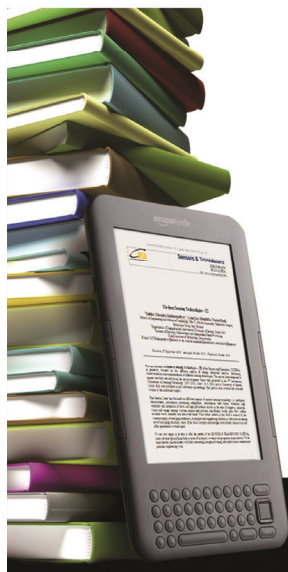
- [1]. Feng Lei, Yao Xinyu, etc., Using the C-API-based real-time simulation system RTW online parameter adjustment, *Software Technology*, Vol.25, 2006, pp. 87-90.
- [2]. Chen Yongchun, From Matlab / Simulink model to code implementation, *Tsinghua University Press*, October 2002, pp. 103-125.
- [3]. Shen Hui, Proficient SIMULINK simulation and control system, *Peking University Press*, 2003, pp. 63-75.
- [4]. Huang Yong An, Ma Lu, Liu Hui Min, MATLAB 7.0 / Simulink 6.0 modeling and simulation development and advanced engineering applications, *Tsinghua University Press*, 2005, pp. 23-25.
- [5]. Xu, Zhi, Tang, Shuo, Design of rapid control prototyping platform based on RTX, *Proceedings of the International Conference on Informatics, Cybernetics, and Computer Engineering (ICCE'2011)* November 19-20, 2011, Melbourne, Australia, *Advances in Intelligent and Soft Computing*, Vol. 110, 2012, pp. 427-434.
- [6]. Pineiro, Luis A., Biezdad, Daniel J., Real-time parameter identification applied to flight simulation, *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 29, 1993, pp. 290-301.
- [7]. Fu, Ho-Ling, Huang, Chin-Yuen, Chang, Wei-Chiang, Real time simulation for parameter optimization with Pentium D and Pentium core 2 T7400 microprocessor, in *Proceedings of the WRI World Congress on Computer Science and Information Engineering*, Vol.3, 2009, pp. 515-519.
- [8]. Lucas, D. E., Mirchandani, P. B., Head, K. L., Remote simulation to evaluate real-time traffic control strategies, *Transportation Research Record*, 2000, pp. 95-100.
- [9]. Kamali, C., Pashilkar, A. A., Raol, J. R., Real-time parameter estimation for reconfigurable control of unstable aircraft, *Defence Science Journal*, Vol.57, 2007, pp. 341-352.
- [10]. Hirose, Noriaki, Terachi, Yasuhisa, Kawafuku, Motohiro, Iwasaki, Makoto, Hirai, Hiromu, Real-time compensation for positioning performance using on-line parameter identification and initial value compensation, *Electrical Engineering in Japan*, Vol.171, April 30, 2010, pp. 40-49.
- [11]. Roozbahani, Hamid, Wu, Huapeng, Handroos, Heikki, Real-time simulation based robust adaptive control of hydraulic servo system, in *Proceedings of the IEEE International Conference on Mechatronics*, 13-15 April 2011, pp. 779-784.
- [12]. Kim, K., Kim, J., Huh, K. S., Yi, K., Cho, D., A real-time multi-vehicle simulator for longitudinal controller design, *Vehicle System Dynamics*, Vol. 44, May 2006, pp. 369-386.

2014 Copyright ©, International Frequency Sensor Association (IFSA) Publishing, S. L. All rights reserved. (<http://www.sensorsportal.com>)



International Frequency Sensor Association Publishing Call for Books Proposals

Sensors, MEMS, Measuring instrumentation, etc.



Benefits and rewards of being an IFSA author:

1

Royalties

Today IFSA offers most high royalty in the world: you will receive 50 % of each book sold in comparison with 8-11 % from other publishers, and get payment on monthly basis compared with other publishers' yearly basis.

2

Quick Publication

IFSA recognizes the value to our customers of timely information, so we produce your book quickly: 2 months publishing schedule compared with other publishers' 5-18-month schedule.

3

The Best Targeted Marketing and Promotion

As a leading online publisher in sensors related fields, IFSA and its Sensors Web Portal has a great expertise and experience to market and promote your book worldwide. An extensive marketing plan will be developed for each new book, including intensive promotions in IFSA's media: journal, magazine, newsletter and online bookstore at Sensors Web Portal.

4

Published Format: printable pdf (Acrobat).

When you publish with IFSA your book will never go out of print and can be delivered to customers in a few minutes.

You are invited kindly to share in the benefits of being an IFSA author and to submit your book proposal or/and a sample chapter for review by e-mail to editor@sensorsportal.com. These proposals may include technical references, application engineering handbooks, monographs, guides and textbooks. Also edited survey books, state-of-the-art or state-of-the-technology, are of interest to us. For more detail please visit: http://www.sensorsportal.com/HTML/IFSA_Publishing.htm