World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:15, No:10, 2021

Practical Challenges of Tunable Parameters in Matlab/Simulink Code Generation

Authors : Ebrahim Shayesteh, Nikolaos Styliaras, Alin George Raducu, Ozan Sahin, Daniel Pombo VáZquez, Jonas Funkquist, Sotirios Thanopoulos

Abstract : One of the important requirements in many code generation projects is defining some of the model parameters tunable. This helps to update the model parameters without performing the code generation again. This paper studies the concept of embedded code generation by MATLAB/Simulink coder targeting the TwinCAT Simulink system. The generated runtime modules are then tested and deployed to the TwinCAT 3 engineering environment. However, defining the parameters tunable in MATLAB/Simulink code generation targeting TwinCAT is not very straightforward. This paper focuses on this subject and reviews some of the techniques tested here to make the parameters tunable in generated runtime modules. Three techniques are proposed for this purpose, including normal tunable parameters, callback functions, and mask subsystems. Moreover, some test Simulink models are developed and used to evaluate the results of proposed approaches. A brief summary of the study results is presented in the following. First of all, the parameters defined tunable and used in defining the values of other Simulink elements (e.g., gain value of a gain block) could be changed after the code generation and this value updating will affect the values of all elements defined based on the values of the tunable parameter. For instance, if parameter K=1 is defined as a tunable parameter in the code generation process and this parameter is used to gain a gain block in Simulink, the gain value for the gain block is equal to 1 in the gain block TwinCAT environment after the code generation. But, the value of K can be changed to a new value (e.g., K=2) in TwinCAT (without doing any new code generation in MATLAB). Then, the gain value of the gain block will change to 2. Secondly, adding a callback function in the form of “pre-load function,” “post-load function,” “start function,” and will not help to make the parameters tunable without performing a new code generation. This means that any MATLAB files should be run before performing the code generation. The parameters defined/calculated in this file will be used as fixed values in the generated code. Thus, adding these files as callback functions to the Simulink model will not make these parameters flexible since the MATLAB files will not be attached to the generated code. Therefore, to change the parameters defined/calculated in these files, the code generation should be done again. However, adding these files as callback functions forces MATLAB to run them before the code generation, and there is no need to define the parameters mentioned in these files separately. Finally, using a tunable parameter in defining/calculating the values of other parameters through the mask is an efficient method to change the value of the latter parameters after the code generation. For instance, if tunable parameter K is used in calculating the value of two other parameters K1 and K2 and, after the code generation, the value of K is updated in TwinCAT environment, the value of parameters K1 and K2 will also be updated (without any new code generation).

Keywords : code generation, MATLAB, tunable parameters, TwinCAT
Conference Title : ICCGO 2021 : International Conference on Code Generation and Optimization
Conference Location : Paris, France
Conference Dates : October 28-29, 2021

Open Science Index, Computer and Information Engineering Vol:15, No:10, 2021 publications.waset.org/abstracts/130623/pdf