A real-time gas turbine simulation model for control logic evaluation

Ye Tao, Yining Pan, Xiaopeng Xin¹, Yunbing Xu and Yongfeng Sui

Hangzhou Steam Turbine & Power Group Co. Ltd.: No. 357 Shiqiao Road, Hangzhou, China

¹ E-mail: xinxp@htc.net.cn

Abstract. During our development of a gas turbine control system, we need to evaluate the correctness and completeness of the control logic before conducting a hardware-in-the-loop (HIL) test. For this purpose, a pure software simulation framework which includes virtual DPU (VDPU) and gas turbine model interfacing through OPC protocol was proposed. A real-time gas turbine simulation model for this framework was developed. This gas turbine model was written in C++ and employs the object-oriented philosophy for maintenance and further development and it uses a non-iterative method to ensure the real-time capability to run simultaneously with the control logic. From the result of preliminary tests on this model, it shows to be able to simulate the response of real gas turbine to the instructions from the control system and suitable to evaluate the control logic.

1. Introduction

As the development of gas turbines, the parameters went higher and higher and the operation point of each component become closer and closer to the unstable boundary. This brings new challenges to the gas turbine control system. The control system needs to react much faster and more accurate than before and the control and protect logic became more complicated. All the changes happened to the control system and control logic made it more error-prone than before, since software simulation and hardware-in-loop simulation are proved to be helpful tools to test the control system [1, 2], our control system development team has proposed a pure software simulation framework to test the correctness and completeness of the control logic.

One of the key parts of the simulation framework is a gas turbine model which can run in real time and take the output of the VDPU as input and return the calculated gas turbine parameters as sensor readings back to the VDPU. A lot of gas turbine simulation models were developed by other researchers like CUI Ning [3], M. T. Schobeiri [4] and Chaibakhsh [5] et al. Those models work fine for predicting the transient behavior of gas turbine but none of them can meet all of the following requirements we need for our framework in one model:

- The model should be based on the components performance maps and basic thermodynamic principles within the gas turbine to calculate the accurate the overall performance. Although the artificial intelligence method become more and more popular in other field to simulate and estimation data [6, 7] and it begin to be applied to predict the performance of gas turbine components [8, 9] and even the overall performance [10], as manufacturer the traditional way of interpolating the characteristic map to get the performance of a component is still preferred for absolute accuracy;
The model should take every command from the VDPU and simulate the corresponding impact of that command, including the rotation of VIGV, change of the fuel mass flow, open and close of bleed valves et al;

The model should run at a fixed step length and each simulation step is synchronized with physical time on PC with the VDPU since the VDPU is running in real-time and will exchange input and output data with the model. This requirement means that we cannot build our model in Simulink or OpenModelica like some other models because the model build in those platforms can only run real-time after compiled and send to a dedicated simulator. This requirement also means the gas turbine model should be simple enough to minimize the calculation time during each time step while being precise enough to evaluate the control logic.

2. Control logic test framework and modelling platform

Our control logic test framework (Figure 1) uses the OPC protocol to exchange data between VDPU and the gas turbine model running in the Modelling platform. Since OPC protocol employs a client-server model the Modeling platform runs as the server and the VDPU runs as the client.

![Figure 1. Control logic test framework.](image)

Our gas turbine model was built in the Modeling platform called SimuWorks, the platform can not only provide the OPC data exchange capability but also make sure the gas turbine model was called exactly at the time intervals of simulation time step length. When developing the gas turbine in this platform we only need to write down the calculation code for the gas turbine at each time step and the platform will handle the rest.

![Figure 2. Overview of the gas turbine model.](image)

3. The gas turbine model

Most of the gas turbine models use the Newton iteration method to solve the non-linear equations of gas turbine model. But this method can be very computation expensive and the iteration number depends on the initial guess of each time step which means it is hard to guarantee the computation time is smaller than the step size and therefore compromise the real-time demand of the model. To solve this problem we used a non-iterative method described by YANG Gang [11]. As we can see in Figure 2, a volume module was introduced between the compressor module and combustor module to
avoid the iteration. The model was written in C++ and to ensure the maintainability and develop of the model, each component in the gas turbine was written as a class, the whole gas turbine model was built by instantiating each class into a corresponding module and assembled together.

3.1. Compressor class
Since the compressor performance is influenced by the environment conditions, VIGV openness and each of the bleed valves’ openness. The compressor performance map was designed to contain the compressor character line at each VIGV openness and each bleed valve openness with dimensionless parameters. At each time step, a series of interpolation was applied on the map to find the corresponding efficiency and mass flow of the compressor. Since the map is dimensionless, the impact of the environment on the compressor is considered automatically during the process.

After find out the operating point on the performance map, the thermal property of air was used to determine the compressor output temperature.

3.2. Combustor class
Combustor takes the mass flow rate, thermal property, and composition of inlet air and fuel to calculate the mass flow rate, thermal property and composition of the hot gas as output. There are two import factors for the calculation, the efficiency of the combustor ($\eta_{\text{comb}}$) and the pressure loss ($\Delta P$).

The off-design efficiency of the combustor was calculated by the following equations. In the equation $b$ mean the part-load coefficient and differs from one combustor to another, the recommended value for $b$ is 6.0. At low-speed condition, the $\eta_{\text{comb}}$ calculated from this equation can be ridiculously low, a temporary method to solve this problem is by restricting $\eta_{\text{comb}}$ to be larger or equal to 80%.

$$\Omega = \frac{W_{31}}{P_{3}^{1.8} * T_{300K} * Vol}$$

$$\eta_{\text{comb}} = 1 - \left( \frac{\Omega}{\Omega_{ds}} \right)^{b} \left( 1 - \eta_{\text{comb-ds}} \right)$$

The pressure loss of the combustor are divided into cold pressure loss ($\Delta P_{\text{cold}}$) and hot pressure loss ($\Delta P_{\text{hot}}$), the constant in the following equation can be calculated from the design condition [12].

$$\Delta P_{\text{cold}} = K_{\text{cold}} * \frac{P_{31}}{W_{31}} \left( \frac{\sqrt{T_{31}}}{P_{31}} \right)^{2}$$

$$\Delta P_{\text{hot}} = K_{\text{hot}} * \frac{P_{31}}{T_{31}} \left( \frac{T_{4} - 1}{T_{31}} \right) * \left( \frac{\sqrt{T_{31}}}{P_{31}} \right)^{2}$$

3.3. Turbine class
Turbine class uses the same performance map interpolation method as the compressor to determine the efficiency and mass flow rate of the turbine, but the performance map is much simpler than the
compressor’s performance map. In order to simplify the calculation in the turbine, each secondary air is considered to expand with the same efficiency as the mainstream.

3.4. Rotor class
The rotor class was designed to calculate two parameters of the gas turbine: the mechanic loss and the speed.

The off-design point mechanic loss was calculated using the following equation, as we can see the mechanic loss is determined by the current rotating speed.

\[ W_{\text{loss}} = \left( \frac{n}{n_{ds}} \right)^2 W_{ds} \]

The speed of rotor was calculated by integrating the equation below, both Euler backward difference and 4th order Runge-Kutta (RK4) method was implemented in the rotor class and can be chosen based on precision requirement.

\[ \frac{dn}{dt} = \frac{900}{J\pi^2 n} W_{\text{eff}} \]

3.5. Volume class
The volume class was introduced to bring complementary variables and equations, then the whole set of original equations become closed form and can be solved without iteration. Since the volume is introduced to eliminate the iteration, the volume class did not consider the change of gas constant and use the input temperature as the temperature of the whole volume. The input and output parameter can be seen in Figure 3.

![Figure 3. Input and output data of volume class.](image)

The differential equation for the pressure in the volume has shown below. Since the effect of volume dynamic on simulation precision is much smaller than the effect of rotor dynamic, only Euler backward difference method was implemented to integrate the pressure.

\[ \frac{dP}{dt} = \left( m_{\text{in}} - m_{\text{out}} \right) \frac{RT_m}{V} \]

3.6. Fluid thermodynamic property class
A common way to calculate the thermal properties of the determined working fluid in a computer program is to call the NIST Reference Fluid Thermodynamic and Transport Properties Database (REFPROP). This method is very accurate but also very time consuming since our gas turbine model will calculate synchronize with the physical time it is critical to ensure each time step can be calculated by computer within the same amount of physical time.

To save a calculation time, we used the traditional way which is described in an LIU [13], the entropy and enthalpy were described as polynomial equations to temperature. By our comparison, the difference between the results calculated by this method and REFPROP for the delta enthalpy within compressor and turbine in a typical condition is around 1.35% and 0.18%, so this method is considered to be precise enough for dynamic gas turbine model.
Figure 4. Input and output data of volume class.
4. Result and conclusions
Since our control logic is under development and it is very hard to “manually” start a gas turbine, so we tested the model after the startup process by suddenly change the fuel mass flow rate from 2.78kg/s to 2.98kg/s (pictures in the left) and from 2.98kg/s to 2.78kg/s (pictures in the right).

As we can see in Figure 4, that the gas turbine model handles the sudden change of input parameter very well except there is a “sharp point” at the first step after the change of input parameter, this will not be a problem when testing the control logic since the output parameter from the VDPU will be divided into very small time steps and the change within each time step is much smaller.

The response of gas turbine during a trip process is also simulated by manual shut down the fuel gas supply and open all the bleed valves at the same time. The speed change during this process is shown in Figure 5, the speed line is quite smooth which means that the compressor class and turbine class handles the interpolations in their performance maps well.

From the result mentioned before we can see that this model works fine and can return the trend of gas turbine’s response to the instruction of the control system. After the development of control logic, more study will be carried out by a combination of the control logic and this gas turbine model, such as the simulation of the gas turbine and control logic during the starting process, shutdown process and emergency trip process. The study on improving the accuracy of the model itself will also be carried out by implementing even higher order Runge-Kutta method and the introduction of heat transfer dynamic.

References
[1] Pugi L, Galardi E, Carcasci C, Rindi A., Lucchesi N 2015 Preliminary design and validation of a Real Time model for hardware in the loop testing of bypass valve actuation system Energy Conversion and Management 92 366-384
[2] Pugi L, Galardi E, Pallini G, Paolucci L, Lucchesi N 2016 Design and testing of a pulley and cable actuator for large ball valves, Proceedings of the Institution of Mechanical Engineers Part I: Journal of Systems and Control: Engineering 230 (7) 622-639
[3] CUI Ning 2008 Study and Application On Real-time Dynamic Simulation Model for Heavy-Duty Gas Turbine Combined Cycle Power Unit North China Electric Power University
[4] Schobeiri M T, Attia M, Lippke C 1994 GETRAN: A generic, modularly structured computer code for simulation of the dynamic behavior of aero-and power generation gas turbine engines. Journal of Engineering for Gas Turbines and Power 116(3): 483-494
[5] Chaibakhsh A, Amirkhani S 2018 A simulation model for transient behaviour of heavy-duty gas
turbines Applied Thermal Engineering 132: 115-127

[6] Aliabad F A, Shojaei S, Zare M & Ekhtesasi M R 2018 Assessment of the fuzzy ARTMAP neural network method performance in geological mapping using satellite images and Boolean logic International Journal of Environmental Science and Technology 1-10

[7] Dehghani A A, Bahremand A R and Shojaei S 2017 Intelligent estimation of flood hydrographs using an adaptive neuro–fuzzy inference system (ANFIS) Modeling Earth Systems and Environment 3(1) 35

[8] Uzol O 2011 A new high-fidelity transient aerothermal model for real-time simulations of the t700 helicopter turboshaft engine Isi bilimi ve teknigi dergisi-journal of thermal science and technology 31(1) 37-44

[9] Ghorbanian K and Gholamrezaei M 2009 An artificial neural network approach to compressor performance prediction Applied Energy 86(7-8) 1210-1221

[10] Asgari, H and Chen X 2015 Gas turbines modeling simulation, and control: using artificial neural networks CRC Press

[11] Yang Gang, Sun Jianguo, Huang Xianghua and Su Weisheng 2003 A Non-Iterative Method of Aero-Engine Modeling Using Complementary Variables Journal of Aerospace Power 18(2)

[12] P Walsh, P Fletcher 2004 Gas Turbine Performance Blackwell publishing

[13] Liu Zhigang 1992 The programming and usage of the thermodynamic property of working fluid Beijing Science publishing