

Secondary Development and Application of Satellite Dynamic Simulation Platform Based on ABAQUS

Xiaotong Zhang^a, Wenlai Ma^b and Xiaolei Jiao

Harbin Institute of Technology, Harbin, 150001, China

^azhangxiaotong0627@163.com; ^bmawenlai@hit.edu.cn

Abstract. As a strong coupling system, structure of the whole satellite has too many design parameters which are undetermined before initial design. In order to overcome the existing problems such as high update frequency of related parameters, complexity of the mouse operation and the low effectiveness of manual modelling, based on secondary development of ABAQUS, a satellite dynamic simulation platform is carried out. First of all, a convenient and friendly user interface is established by writing registration file and the dialog interface creation file based on Python scripting language. Besides, kernel execution scripts is also developed based on Python scripting language, therefore, the parametric and automation modelling as well as simulating process are realized and the efficiency of analysis can be improved.

1. Introduction

In the process of launching, orbiting and returning, thus the satellite undergoes many physical and chemical changes such as high temperature, acceleration and vibration, causing excitation to the structure which directly affects the stability of the satellite structure and the accuracy of navigation and communication. Therefore, the dynamic simulation analysis of the whole satellite is the foundation to verify the correctness of the model establishment[1]. Based on the ABAQUS software, a new method of modelling and dynamic simulation is put forward in this paper, which will support the further design and optimization.

The engineers need to study dynamic characteristics of the whole satellite under the conditions of different dimensions and physical properties at the beginning of its design. However, there are too many cumbersome mouse operations when using ABAQUS for modelling, and once the model needs to be modified, the mouse operations have to be repeated. In order to avoid these unnecessary manual operations which further increase the computational cost and error probability, this paper uses the application program interface provided by ABAQUS to write a new user graphical interface and the pre and post processing kernel script in the Python language. Users only need to input the corresponding parameters in the interface, then the modelling and simulation of the whole satellite can be realized, which can effectively improve the efficiency and provide a simple, efficient and universal platform for satellite design[2].

2. Key technology of ABAQUS secondary development

This paper realizes satellite parametric modelling and simulation based on kernel script and graphical user interface. It mainly involves the following three types of files[3]: kernel execution file xx.py is the core of the plug-in, and it is used to drive ABAQUS/CAE to execute internal commands to carry



Content from this work may be used under the terms of the [Creative Commons Attribution 3.0 licence](https://creativecommons.org/licenses/by/3.0/). Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.

out CAE modelling and data processing; registration file `xx_plugin.py` is used to register various kinds of control keywords, check the legitimacy of data, and register plug-in tools to the Plug-ins menu in ABAQUS/CAE; the dialog interface creation file `xxDB.py` is used to form a friendly input interface by defining and associating various forms and controls, so that users can manage all the parameters in a unified graphical interface. The running process of the platform is shown in figure 1.

2.1 ABAQUS/CAE kernel script

ABAQUS provides the script interface for users, and the script that is transferred will directly access the kernel script of the program, thus implementing the pre- and post-processing. This not only facilitates the management of data, but also makes it more convenient to use. Users only need to set and modify relevant parameters in the script program and execute it, and various complicated operations in the pre- and post-processing can be completed quickly and accurately. The ABAQUS kernel execution program is based on Python language[4], and its script interface involves about 500 objects which can be divided into three categories[5], as shown in figure 2. These objects in ABAQUS are crucial for the kernel script to run efficiently.

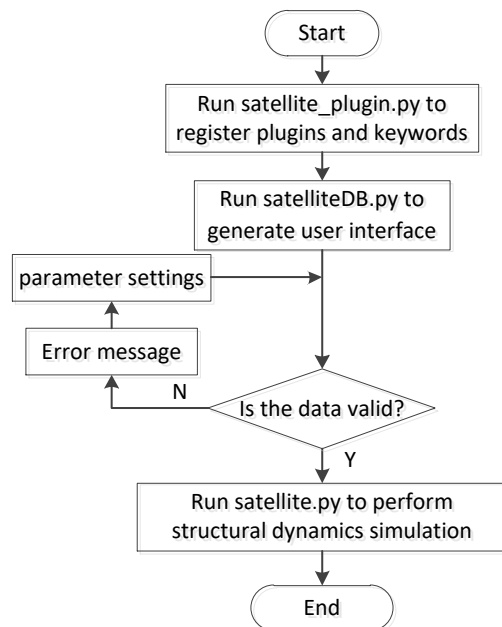


Figure 1. running process of the platform

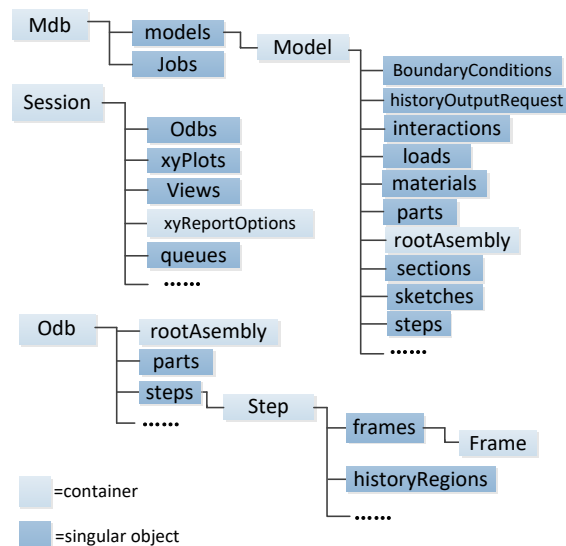


Figure 2. ABAQUS object model

2.2 ABAQUS graphical user interface

The graphical user interface mainly includes two parts: kernel process and GUI process[6]. ABAQUS realizes the human-computer interaction through the data interaction between these two parts, as shown in figure 3. There are two ways to create a GUI graphical interface based on ABAQUS/CAE: the RSG Dialog Builder and the ABAQUS GUI Toolkit. The former is embedded in ABAQUS/CAE, which provides an efficient and convenient method to create user interfaces. However it has a very limited variety of controls. In order to meet the complex needs of satellite modeling, this paper uses ABAQUS GUI toolkit to create user interfaces, which is to edit GUI commands directly in the source file. This method can realize many functions which cannot be realized by the RSG Dialog Builder[7], such as display error prompts, control usage status of the controls, etc.

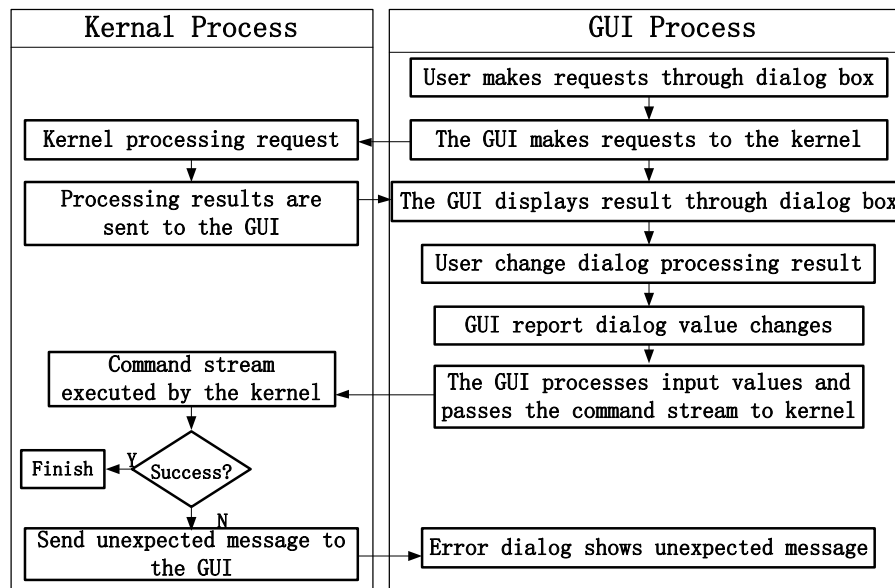


Figure 3. interaction between GUI and kernel in ABAQUS/CAE

2.3 Model simplification technology

In order to reduce weight and ensure structural safety of satellite, the honeycomb sandwich plate is applied widely in the modern spacecraft design because of its excellent capability[8]. When the finite element calculation of the satellite is carried out by the general simulation software, the lack of the honeycomb cell data library makes the processing of the honeycomb sandwich plates a challenging problem. This paper combines the secondary development of ABAQUS and the existing equivalent principle of honeycomb sandwich plates to get the equivalent model of honeycomb sandwich plates to complete the modelling and dynamic solution of the whole satellite.

At present, there are three kinds of equivalent methods commonly used in engineering: the sandwich plate theory, the honeycomb plate theory and the equivalent plate theory[9]. Here we use the equivalent plate theory as an example to describe the equivalent principle. Let's assume the total thickness of the honeycomb sandwich plate is $2H$, the thickness and the density of the upper and lower plates is d and ρ_f respectively, the density of the honeycomb core is ρ_c , the elastic modulus is E , and the Poisson's ratio is μ . The honeycomb sandwich plate is equivalent to the isotropic shell element with unequal thickness. It is assumed that the equivalent plate satisfies the Kirchhoff hypothesis of the thin plate bending theory[10]. The parameters of equivalent thickness, equivalent elastic modulus, equivalent Poisson's ratio and equivalent density can be obtained as follows:

$$t_{eq} = \left[d^2 + 12 \left(H - \frac{d}{2} \right)^2 \right]^{1/2} \quad (1)$$

$$E_{eq} = \frac{2Ed}{t_{eq}} \quad (2)$$

$$\mu_{eq} = \mu \quad (3)$$

$$\rho_{eq} = \frac{2\rho_f d + 2\rho_c(H-d)}{t_{eq}} \quad (4)$$

3. Development of the whole satellite dynamic simulation platform

3.1 Plug-in operation

Save the prepared satellite.py, satelliteDB.py, satellite_plugin.py files and the required graphics files into the plug-in folder under the ABAQUS installation directory, and users can find the satellite

parametric dynamic simulation platform plugin in the plugins drop-down menu. After that the user interface (as shown in figure 4) created by the dialog interface creation file and registration file will be displayed. In the interface, users can set the structure parameters of the satellite cabin, solar arrays, antenna and material properties according to their own needs. If the input parameters are invalid, such as the length parameter is negative, the system will pop up an error prompt dialog to remind users to input the correct and effective parameters. Click the OK button after completing the parameter input and the plug-in will automatically run the kernel program file which includes the establishment of the satellite model, the assignment of material properties, the application of constraints and loads, the automation of the grid and the corresponding simulation calculations as needed.

3.2 Geometric modelling

The plug-in kernel program performs the whole satellite modeling, material properties attributing and assembly processing after users finish inputting the parameters needed as shown in figure 4 and the material parameters shown in figure 5 input by the user. After that, users will get the whole satellite model, as shown in figure 6, some of the key codes are as follows:

```
p= mdb. models['Model-1'].Part(name='Part-4',dimensionality=THREE_D,type=DEFORMABLE_BODY) # create a three-dimensional model
p. Section Assignment(region=region,sectionName='Section-1',offset=0.0,offsetType=MIDDLE_SURFACE, offset Field="",thickness Assignment=FROM_SECTION)#assign section properties
a.WirePolyLine(points=((v1[vnum1[j]],v4[vnum2[j]]),),mergeType=IMPRINT, meshable=OFF) # create a rope unit
```

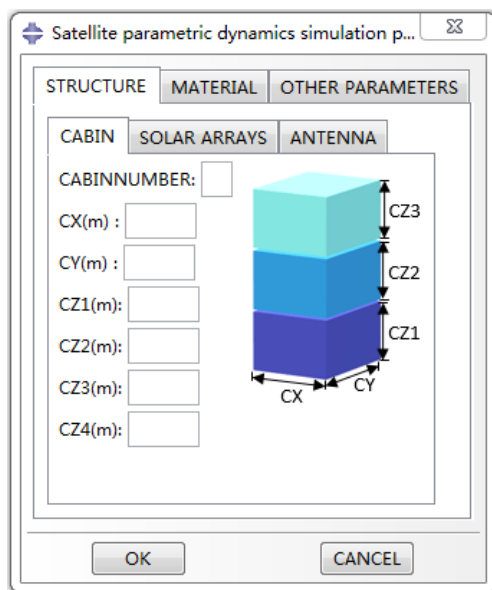


Figure 4. user interface of structure size setting

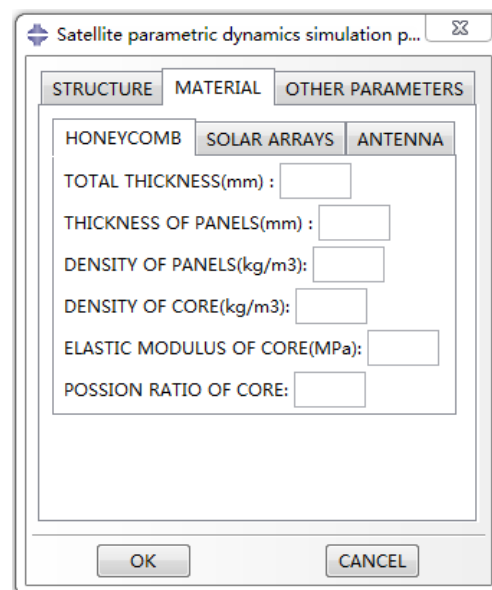


Figure 5. user interface of material property setting

3.3 Constraint exertion

After the satellite modelling is completed, the interactive constraints of different parts of the satellite are exerted, such as the rope connection between the ribs of the antenna and the connection between the cabin and the antenna. The parameters of the constraints discussed above can be set by users in the interface shown in figure 7. The model with constraints is shown in figure 8. Some of the key codes are as follows:

```
a.ConnectorOrientation(region=csa.getSet(),localCsys1=d1[1]) #Connection section assignment
```

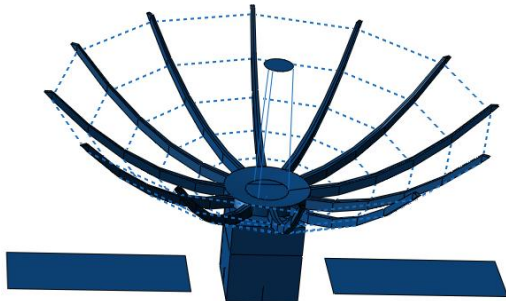


Figure 6. the whole satellite model

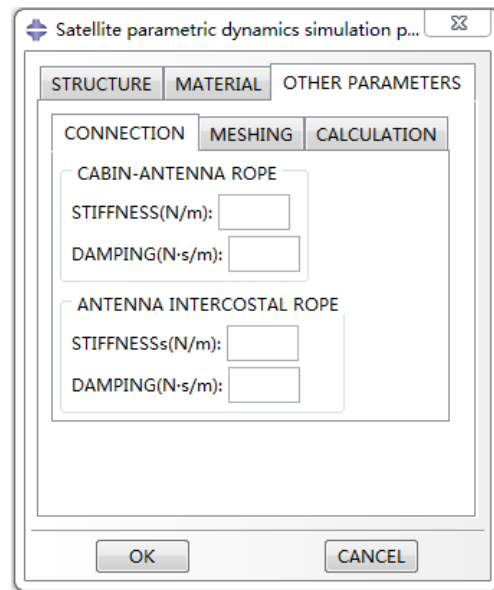


Figure 7. user interface of constraint attribute setting

3.4 Meshing

Next, the plug-in program automatically performs meshing according to the meshing parameters set by users. The finite element model of the whole satellite is shown in figure 9.

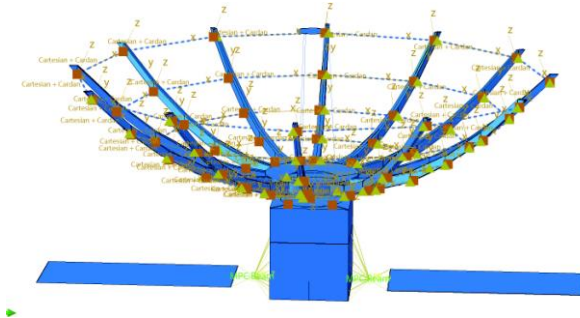


Figure 8. satellite model with constraint

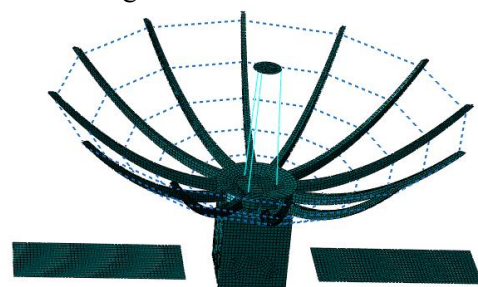


Figure 9. finite element model of satellite

3.5 Structural dynamic analysis

Modal analysis is an important approach to analyzing the dynamic characteristics of satellites, such as vibration. The modal calculation of the satellite model established in the previous section is carried out. Some of the key codes are as follows:

```
mdb.Job(name='satellite', model='Model-1',.....)#create a job
mdb.jobs['satellite'].submit(consistencyChecking=OFF)#submit calculation
session.writeImageAnimation(fileName='mode-%i%i',format=AVI,canvasObjects=(session.
viewports ['Viewport: 1'], ))# output mode shape
```

Conclusion can be drawn from the mode shapes shown in figure 10 that the equivalent displacement is mostly concentrated on the top of the antenna rib and the solar arrays, which indicates that it is necessary to pay more attention to strengthen the antenna rib and the solar arrays in the following design to improve the dynamic stability of the whole satellite.

4. Conclusion

This paper carries out the secondary development of ABAQUS based on Python language and the plug-in program is written to realize the model equivalent, model establishment and dynamic

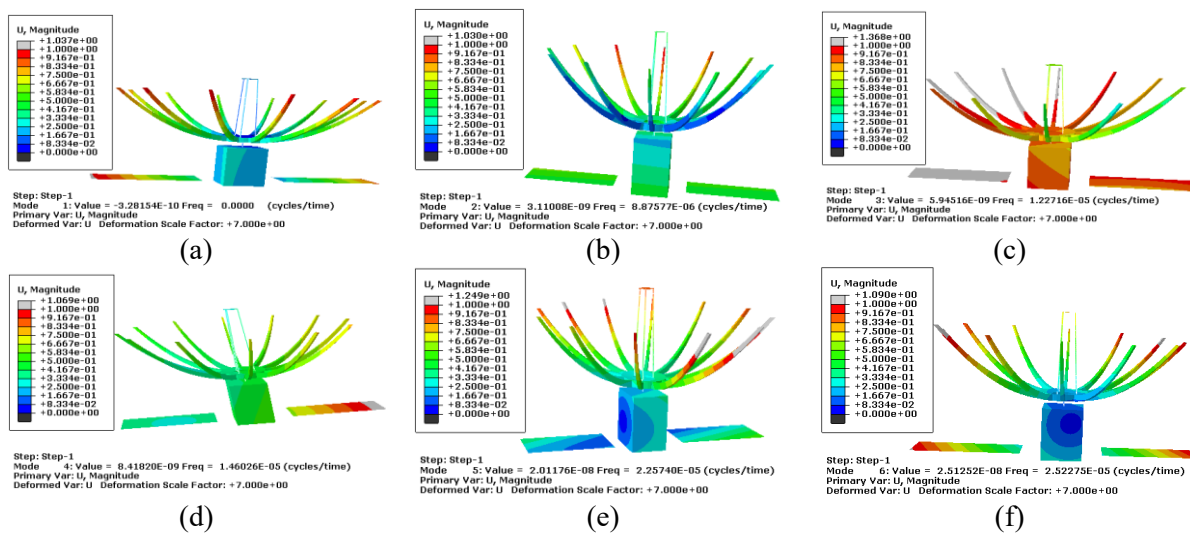


Figure 10. First 6th mode shapes

simulation of the satellite. The developed user interface is easy to operate, and enables the users who lack the experience of the ABAQUS application to carry out the dynamic simulation analysis of the whole satellite conveniently. While achieving unified management of parameters, it effectively improves simulation efficiency and avoids operation error. The plug-in can be extended to the design of the same type of satellites, and provides an effective reference for subsequent design and optimization of other types of satellites.

References

- [1] Pulecchi, T., Casella, F., & Lovera, M. 2010 Object-oriented modelling for spacecraft dynamics: tools and applications. *Simulation Modelling Practice & Theory* **18**(1) 63-86.
- [2] Song T, Cheng-Gong M A, Wang H, et al. 2018 Parametric modeling and harmonic response analysis of drum chest frame based on ABAQUS-Python secondary development *Wuhan. World of Building Materials*.
- [3] Wang, H., and Wei, H. 2013 Application of abaqus/plugin in the parametric modeling and mode analysis of gun barrel. *Ordnance Industry Automation*.
- [4] Luo L, Zhao M. 2011 The secondary development of ABAQUS by using python and the application of the advanced GA *Hong Kong. Physics Procedia* **22** 68-73.
- [5] Zhang X D, Liu J, Zou S G, et al. 2017 Application of Abaqus secondary development to the simulation of intermediate frequency hot pipe-bending *Beijing. Journal of Plasticity Engineering*.
- [6] Wang, J. L., Ping, L. I. 2009 Secondary development for gui of box girder bridge based on abaqus. *Journal of Chongqing Jiaotong University*.
- [7] Meng, L. I., Cun-Gui, Y. U., Cui, E. W., and Xian-Wei, Q. I. 2013 Secondary development of abaqus and its application in creating parametric model of artillery. *Journal of Sichuan Ordnance*.
- [8] Boudjemai, A., Amri, R., Mankour, A., Salem, H., Bouanane, M. H., and Boutchicha, D. 2012 Modal analysis and testing of hexagonal honeycomb plates used for satellite structural design. *Materials & Design* **35** 266-275.
- [9] Luo, H., Liu, G., Ma, S., & Liu, W. 2011 Dynamic analysis of the spacecraft structure on orbit made up of honeycomb sandwich plates. *IEEE International Conference on Computer Science and Automation Engineering* **Vol.1** pp.83-87.
- [10] Xia L J, Jin X D, Wang Y B. 2003 Equivalent analysis of honeycomb sandwich plates for satellite structure *Shanghai. Journal of Shanghai Jiaotong University* **37**(7) 999-1001.