ATLANTIS PRESS

Advances in Engineering Research, volume 181

International Conference on Electronical, Mechanical and Materials Engineering (ICE2ME 2019)

Macro Micro - Coupling Simulation Front - End Processing Tools Based on SALOME Platform

Jun Hu[1,2]

[1]Nanjing University of Science and Technology, Nanjing, Jiangsu, China

[2]Huangpi District Mission Hill Tower 95942 troops, Wuhan City, Hubei Province

***Abstract*—Micro-Macroscopic Coupled Simulation is an important method in the field of microfluidics. Pre-Post processing is an important process of macro and micro coupling simulation method. Nowadays, there is no unified and effective pre-post processing method in Micro-Macroscopic Coupled Simulation. Based on the research status of the pre-post processing method and the requirement of Micro-Macroscopic Coupled Simulation. Paper deeply analyzes the SALOME software platform and focuses on the research of design, implementation and performance optimization of pre-post processing tool in Micro-Macroscopic Coupled Simulation. paper completes the requirement analysis of the pre-post processing workflow and deeply analyzes the architecture of SALOME platform, and designs and implements the pre-processing and post-processing modules of MPT module. Finally, paper completes the parallel optimization for MPT tool. Paper designs and implements the MPICH parallel optimization strategies, achieves good parallel effect and improves the performance of the MPT tool.**

Keywords—macro-microcosmic coupled method; pre-post processing; SALOME; MPT

I. INTRODUCTION

Macro-micro coupling simulation is a typical method of multi-scale coupling calculation, and has wide application value in complex system simulation. The macro-microscopic coupling simulation method refers to the calculation of the physical model using different scales, the use of smaller scale microscopic simulation method for the sensitive region, to ensure the accuracy of the simulation requirements; the results of the region is not sensitive to the use of large scale macro simulation The calculation method, improve the calculation efficiency, reduce the calculation cost.

The front-end processing is an indispensable part of the macro-micro coupling simulation platform, which is mainly responsible for the data input and data output processing of the core solver. For the macroscopic computational fluid dynamics simulation, the front-end processing is to perform the grid generation, parallel division and boundary condition definition of the simulation area. The back-end processing is the data processing and display of the solver's calculation results, which is convenient for the user to visualize the physical phenomena For the molecular dynamics simulation, the front-end processing needs to specify the information of the simulation area, configure the physical properties of the particle information, back-end processing needs from the micro-data to the macro data conversion, user-friendly from the macro level and Study the physical phenomenon.

This paper is based on SALOME software platform for front and back processing tools development. This paper first studies the basic architecture of SALOME software platform, the operation mechanism and development method of the module. On this basis, the demand analysis of macro and micro-coupled front-end and back-end processing is carried out, and the unified front-end processing tool. Finally, the subject of the front and back processing tools to start the performance bottleneck analysis, and focus on the back-end processing to optimize the acceleration and improve the efficiency of the tool.

The second part is the design and implementation of the MPT pre-processing module. The fourth section is the design and implementation of the MPT post-processing module. The fifth part is the design and implementation of the MPT post-processing module. The fifth section is the design and implementation of the MPT post- The section uses MPICH to perform performance optimization on MPT tools in parallel. Section 6 is related work; Section VII is a summary of this article.

II. SALOME SOFTWARE PLATFORM ANALYSIS

SALOME is an open source software that provides a common numerical simulation of the front and rear processing platform, the use of open, flexible, reusable components of the software architecture [2]. The SALOME software platform is designed to address the need for numerical simulation, to provide a simple user interface, to support the integration of specific computing solvers, to provide effective pre-processing and post-processing solutions to support the integration of numerical simulation of the relevant algorithm code. The software platform provides users with numerical simulation of the cross-platform solution, its distributed architecture, so that the software platform can run on different platforms, truly cross-platform, cross-operating system, cross-language.

A. CORBA Software Standards

SALOME software platform cross-platform, cross-operating system, cross-language advantage is the use of CORBA (Common Object Request Broker Architecture, Common Object Request Broker Architecture) [3] software standards to achieve. The CORBA standard defines a set of common API interfaces, communication protocols, objects, and services that can be invoked and manipulated by different

applications. These applications can be developed using different programming languages that can run on different platforms and run on different Operating system, which is the biggest advantage of CORBA standards [4].

Figure I is the CORBA architecture, CORBA use Client / Server mode, the client is called IDL Stub, the server is called IDL Skeleton. The client accesses the object on the server through the IDL stub, and the object on the server implements its specific service on the basis of IDL Skeleton, and the IDL stub and IDL Skeleton pass through the ORB (Object Request Broker) bus Communication.

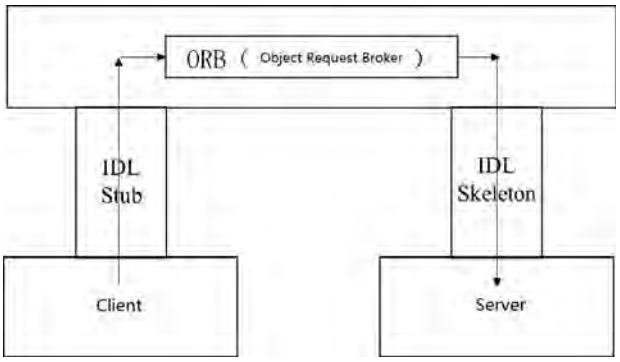

FIGURE I. CORBA ARCHITECTURE

B. *SALOME Software Architecture*

SALOME software architecture based on CORBA technology, and the use of distributed system application modeling [5], as shown in Figure II, through CORBA technology, SALOME software platform will need to use the software packages are integrated together to reduce the system Of the dependencies, therefore, SALOME software platform installation depends on the relatively small software package, software independence is strong.

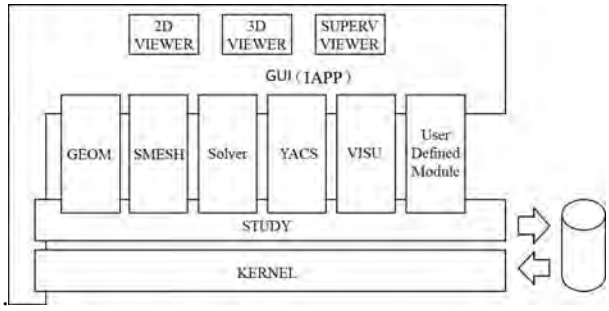

FIGURE II. SALOME SOFTWARE ARCHITECTURE

The module is the basic concept of the SALOME software platform. Modules can be understood as separate independent software applications that are part of the SALOME software that specializes in specific function functions. SALOME software platform integrated software modules are:

KERNEL module: the core module is the basis and core of the SALOME software platform, provides the realization of the module runtime class, and for other modules to provide the necessary interface to run.

GUI module: Salome to provide a unified user interface, each module can be integrated design their own interface and interface.

GEOM module: Provides the basic function of creating, importing, and modifying any CAD model (IGES format, STEP format, BRep format, etc.).

SMESH module: Provides the use of standard grid generation algorithms or the use of external grid generators to create a grid for CAD models.

PARAVIEW module: The module is a macro data analysis and display tool, the original module is a separate operation of the software, computational fluid dynamics simulation field is widely used, SALOME software platform will also be integrated into the software platform.

III. MACRO MICRO-COUPLING SIMULATION OF THE FRONT AND REAR PROCESSING TOOLS OVERALL DESIGN

A. *Macro Micro-coupling Simulation before and after the End of Business Process Analysis*

The macro - micro coupling coupled simulation process can be divided into pre - processing, simulation and post - processing. Front-end back-end processing is separated from time, so the pre-processing and post-processing without functional overlap and interaction can be designed and implemented separately, and finally merged into a complete front and rear processing tools.

The coupling solution process is closely related to the pre-and post-processing process. The solver obtains the configuration file from the pre-processing process and outputs the calculation result for the post-processing process. Macro micro-coupling solver to solve the brief process shown in Figure III, the first is the relevant configuration file input, and CFD, MD, coupling area and attribute initialization; Second, a CFD calculation time step for the calculation cycle based on configuration information And then the time step correlation simulation calculation result is output after the end of each step to determine whether the simulation solving process is completed.

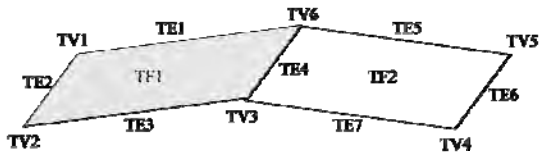

FIGURE III. SHELL DATA STRUCTURE

B. *Macro Micro-coupling Simulation of a Unified Front and Rear Processing Tools Needs Analysis*

In the front-end processing phase, there is no front-end processing tool for coupling simulation, resulting in domain experts in the front-end processing process to spend a lot of time. In the processing part of the CFD simulation area, the grid program provided by the CFD platform is relatively simple, and the simple geometric region of the rectangle and the circle is supported by the grid. For the complex geometric

region, Deal with more complex. The simple geometric region is only an ideal condition, and most of the practical engineering applications need to simulate the geometric regions of the complex boundaries. Therefore, the CFD area front-end processing requires the support of third-party geometric modeling and grid generation. In the MD area information processing stage, there is no tool or script suitable for configuring the MD region coupling information, which reduces the efficiency of the domain expert and affects the promotion and use of the macro micro coupling solver. Based on the above problems in front-end processing, we need to study the unified pre-processing tools to achieve complex geometric regions and grid generation, and deal with MD area information, automated manual configuration to complete the work.

In the back-end processing stage, there is no macro micro-unified display of the back-end processing tools. As the macro micro-scale is different, the information displayed can not be unified, the need for uniformity of the data scale to observe the phenomenon, to achieve the purpose of macro-micro coupling simulation. Macroscopic information and micro-information is not uniform, making the observation of obstacles, hinder the summary of the conclusions of the simulation and summary. Therefore, this paper needs to study a unified back-end processing methods to achieve data scale and format of the unified and unified display.

C. MPT Tool Overall Structure Design

Based on the thinking and analysis of business process and demand, this paper presents MPT (Macro-Micro Pre-Post Processing Tool) tool based on SALOME software platform. The MPT tool is based on the SALOME software platform, including the GEOM module, the SMESH module, the PARAVIEW module and the MPT tool in the software framework. The MPT tool includes the MPT preprocessing module implemented in this paper. The MPT tool is based on the SALOME software platform, Together with other modules constitute the entire macro micro-coupling simulation of a unified front-end processing tools. (See Figure IV)

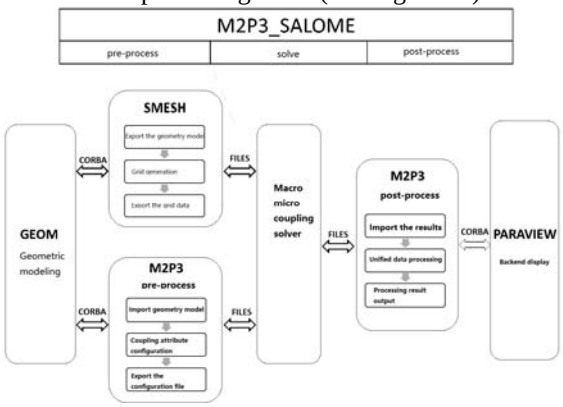

FIGURE IV. MPT FRONT AND REAR PROCESSING TOOL ARCHITECTURE

IV. Design and Implementation of MPT Pretreatment Module

A. MPT Preprocessing Module Timing Diagram

In this paper, MPT pre-processing module for the use case analysis, according to the use case analysis of this design of the MPT pre-processing module timing diagram. MPT pre-processing module core business process is mainly from the GEOM module to obtain data, configure the MD area information and output. For the two core business processes, create a timing diagram, as Figure V shown.

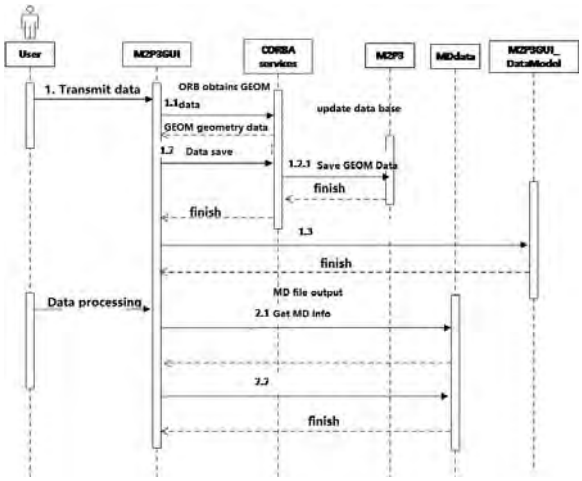

FIGURE V. MPT PRE-PROCESSING MODULE CORE TIMING DIAGRAM

B. MPT Pre-processing Module Implementation

MPT preprocessing module includes CORBA engine and graphical interface (GUI) two parts. The module's CORBA engine defines the interface of the module and the core service, which is the interface of the module's GUI and other modules.

MPT pre-processing module CORBA engine definition is through the idl folder under the MPT.idl implementation, M2P2.idl only defines the interface, not the interface code implementation. Table I describes the definition of the M2P2.idl interface.

The MPT GUI class contains functions such as module creation, activation, and revocation in the module lifecycle, as well as interface functions that interact with the user.

MPT pre-processing module MPTGUI class, as shown in Figure VI, inherited from the KERNEL module SalomeApp_Module class. The following describes the MPT module activation or revocation is called the relevant function.

TABLE I. MPT_ORB PSEUDO-CODE

	MPT_ORB
1	Defines the Property structure
2	Define the core data structure
3	Define the MDobject queue, MDobjectList
4	Define the interface MPT_Gen
4.1	Define the interface function setData ()
4.2	Define the interface function getData ()
4.3	Define the interface function processData ()

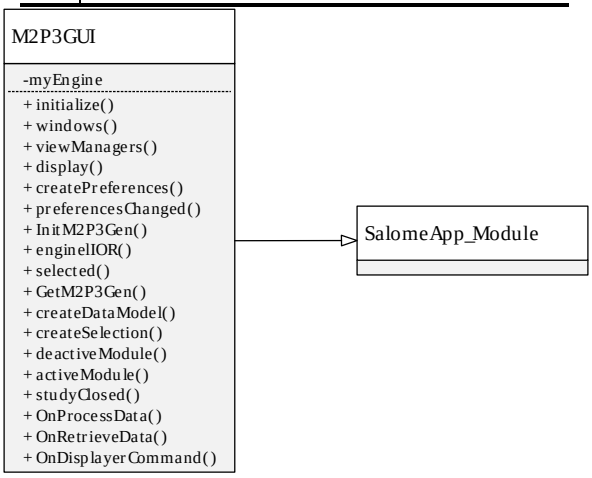

FIGURE VI. MPTGUI CLASS DIAGRAM

Initialize () function: The MPT preprocessing module is called when it is first activated, and the GUI module object only executes the initialization function once it is created.

activateModule () function: is called every time the module is activated (from other modules to the MPT pre-processing module), the main function is to make the initialization function to create the menu, toolbar and other visible.

DeactivateModule () function: In contrast to the activateModule () function, the main function is the menu that makes the activateModule () function visible, and the toolbar is not visible when the module is revoked or when the software is turned off.

GetMPTGen () function: returns a static variable, and other GUI classes can access the services in the MPT preprocessing module engine by returning static variables.

The function of the OnRetrieveData () function is to obtain the selected MD simulation area geometry data from the GEOM module through the interface defined by the CORBA protocol in the form of GEOM: GEOM_Object and store the obtained GEOM: GEOM_Object object in its own Data structure, the completion of data acquisition function.

V. MPT POST-PROCESSING MODULE DESIGN AND IMPLEMENTATION

The data in the MPT post-processing module need to convert the microscopic physical quantity into the macroscopic physical quantity in the region, such as the microscopic velocity statistics of the micro-particles, which is based on the knowledge of statistical physics.

A. Data from Microcosmic to Macroscopic Data

In statistical physics, there is a concept of the distribution function [6]. Statistical physics is called a bridge between microcosmic physical quantities and macroscopic physical quantities, and the partition function is the cornerstone of the bridge. In this paper, the system is a set of systems of regular systems (the same number of molecules N, the same volume V and the same temperature T, and the partition function of the regular system is:

$$Q(N,V,T)=\sum_i e^{-E_i/k_BT}$$

Where, k_B for the Boltzmann constant, $k_B=1.0806\times10^{-16}erg/(mol\square K)$ for the energy of the system. The probability of system energy is:

$$P_i=\frac{e^{-E_i/k_BT}}{\sum_j e^{-E_j/k_BT}}=\frac{e^{-E_i/k_BT}}{Q}$$

It is possible to set $E_i<E_j$ the number of systems, and E_i corresponding to the number of systems n_i, corresponding to:

$$\frac{\overline{n_k}}{\overline{n_i}}=\frac{P_k}{P_i}=\frac{e^{-E_k/k_BT}}{e^{-E_i/k_BT}}=e^{-(E_k-E_i)/k_BT}=e^{-\Delta E/k_BT}$$

In the formula, $\Delta E=E_k-E_i$ the difference between the two states of energy, $\overline{n_k}=\langle n_k\rangle$ is the sign of the mean.

Then the average of any physical quantities B of the regular system is:

$$\overline{B}=\sum_i P_iB_i=\sum_i\frac{B_ie^{-E_i/k_BT}}{Q}=\frac{1}{Q}\sum_i B_ie^{-E_i/k_BT}$$

Using this relationship and the thermodynamic correlation formula, we can deduce the various thermodynamic characteristics of the regular system, such as entropy S, pressure P, velocity V, etc., calculated by the statistical physics of the value of the mechanical average, if the number of molecules enough, Then the mechanical average is consistent with the thermodynamic measure.

B. MPT Post-processing Module Design

In the same way as MPT preprocessing module, this paper analyzes the post-processing function of the module on the

basis of case analysis, and the core business process is the macro-micro data coupling process. The core business process is created in this paper, As shown in Figure VII.

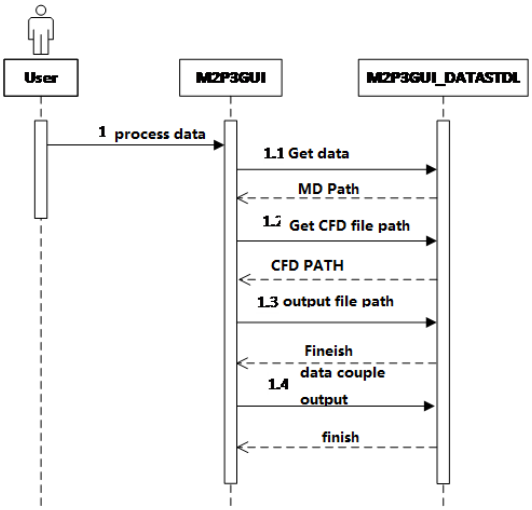

FIGURE VII. MPT POST-PROCESSING TIMING DIAGRAM

Will MPT the entire module class diagram created, as shown in Figure VII.

There are two input files for the module, one is the VTD file of the CFD analog output, one is the OUT file of the MD simulation calculation output, which is the key to the data conversion operation data acquisition in the DATASTDL class. Only a clear understanding of the input file format, In order to correctly obtain the data, so as to carry on the unified operation to the data.

The DDIASTDL () function is used to instantiate the DATASTDL class in the OnDATASTDL () function; the OnDATASTDL () function calls the setMDinfilepath () function in the DATASTDL class to get the DATASTDL class in the function of the MPT postprocessing module. MD is calculated by calling the setCFDinfilepath () function in the DATASTDL class to get the storage path of the CFD calculation result file; call the setoutfilepath () function in the DATASTDL class to get the storage path of the output processing result file; finally call the MDTOVTK in the DATASTDL class () Function for data conversion and data output operations.

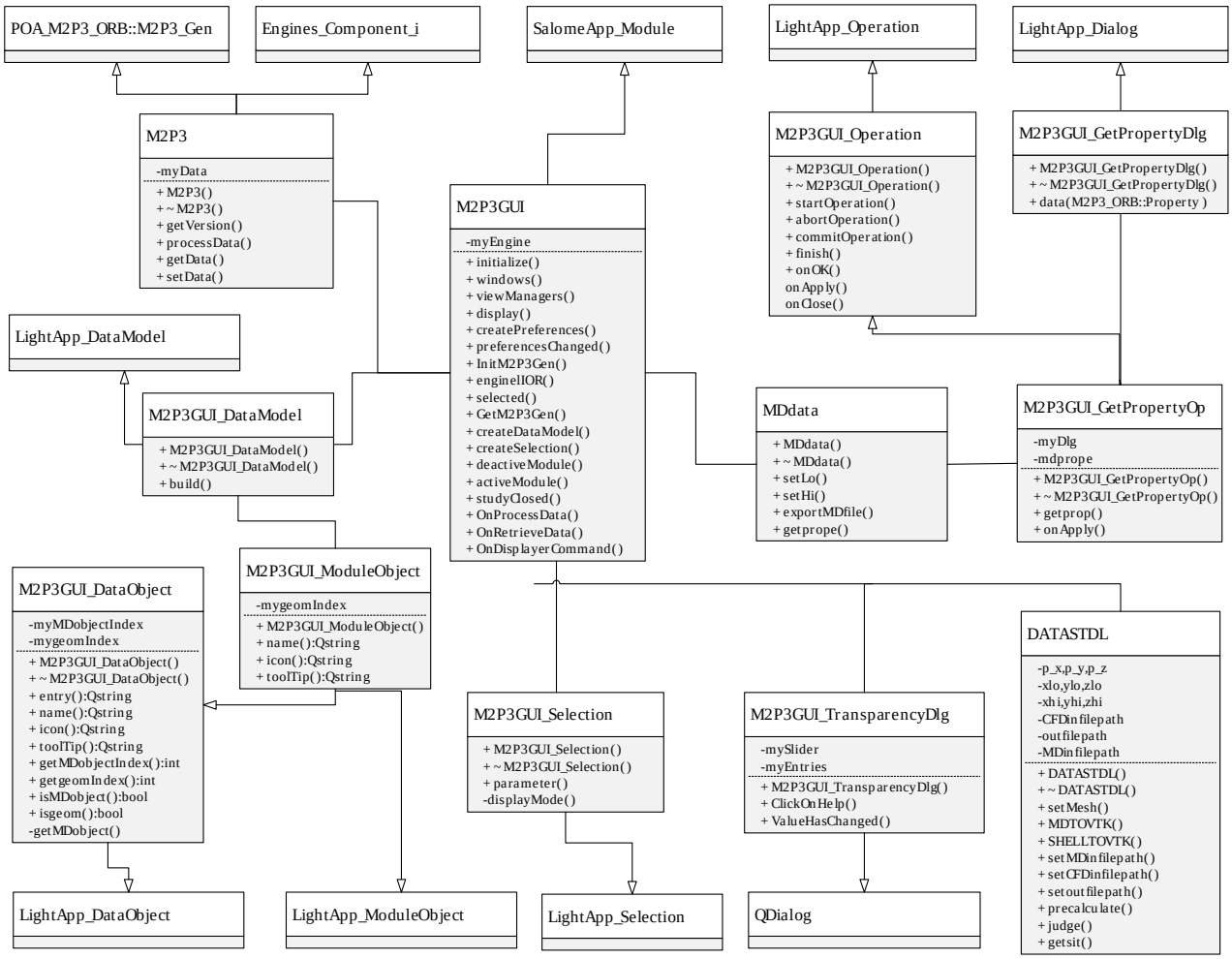

FIGURE VIII. MPT MODULE CLASS DIAGRAM

VI. MPT TOOLS ARE OPTIMIZED IN PARALLEL

With the arrival of multi-core era, people are increasingly demanding the speed of the software, parallel computing more and more applied to the design and development of software, macro micro-coupling simulation front and back processing there is a parallel acceleration needs. In particular, the microscopic data statistics process of the MPT post-processing module is large because the number of microscopic particles is large and the statistical process is time-consuming. The computational complexity increases with the increase of the statistical time step.

The main function of the pre-processing module is to obtain the geometric objects of the MD simulation area from the GEOM module, obtain the configuration information of the MD simulation area from the user, and output the configuration file. Each step of the preprocessing section needs to interact with the user. The data processing time of the processing program is O (1) time complexity. From the pre-processed code analysis, the preprocessing module does not calculate the intensive part, Parallel optimization.

The main function of the post-processing module is the unified processing of the CFD and MD simulation results data and the output of the VTK format file. Which in the implementation of data unified processing due to the larger number of particles, processing more time-consuming; In addition, each time step need to deal with the number of files more, if you need to deal with more time step of the file, the size of the problem will Rapid increase, the processing time will be greatly increased, therefore, post-processing module optimization space is larger.

The post-processing is based on a CFD time step as a processing cycle. There is no data interaction between the time step and the time step. The processing of each time step is the same, and the data processing between the time step and the time step has a natural data independent Sex. In MPT post-processing module optimization, this paper first adopted the strategy is based on MPICH multi-process coarse-grained parallel. After the post-processing program uses MPI for parallel processing, a time-step data processing is considered as a separate task, and different time-step tasks are assigned to different processes according to the process number. Due to the data independence between time steps, There is no need for data communication between processes, reducing the communication overhead of the process. After the calculation, the process can be revoked. Assuming that there are n processes for running the program, the task assignment diagram is shown in Figure XI.

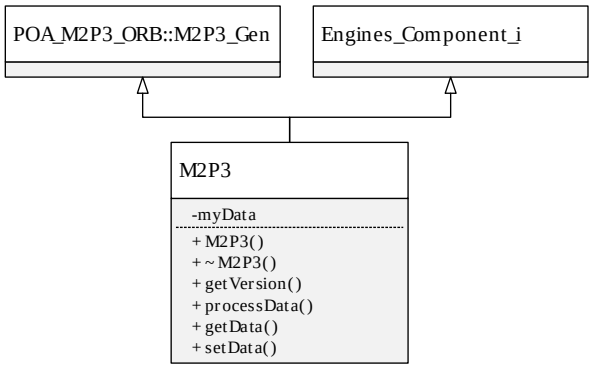

FIGURE IX. PROCESS ASSIGNMENT DIAGRAM

In theory, regardless of the process of creating overhead, the time step for the granularity of the process of task allocation and MPI procedures used in the number of processes is a linear relationship between the number of processes, processing tasks, the more powerful, The shorter the time, the next step will be tested on the proposed parallel strategy.

The above is the design and analysis of the optimization strategy of the post-processing part of the MPT module. The following section will test and evaluate the optimization effect. The MPICH optimized parallel program is tested. This paper tests the four different file sizes of 400, 800, 1600, and 3200, and runs the program at 1, 2, 4, 8, 16, and 32 processes. The test results are shown in Figure X.

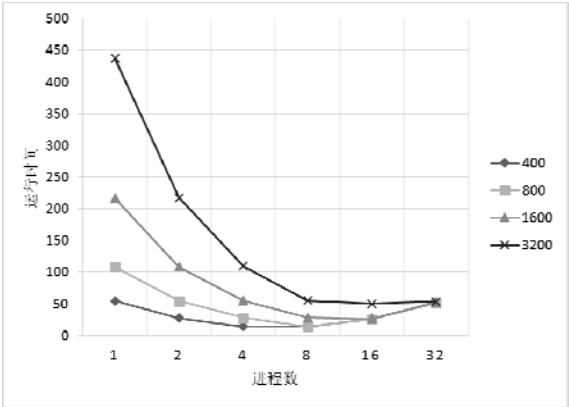

FIGURE X. DIFFERENT FILE SIZE MULTI-PROCESS PROGRAM RUNNING TIME

It can be seen from Figure XI, in the same file size, the program execution time with the number of changes in the first increase after the increase, in 8 or 16 processes when the program run time to a minimum. The speed of the program is shown in Figure XI.

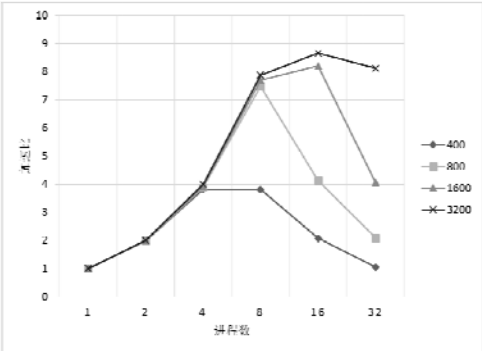

FIGURE XI. DIFFERENT FILE SIZE MULTI-PROCESS PROGRAM RUNNING SPEED RATIO

From the results found in the figure, the file size of the same circumstances, the program to run the speed of the increase with the number of processes increased first and then reduce the number of processes in the 16 when the program faster than the maximum.

To sum up, this paper concludes that the MPICH parallel optimization of the post-processing procedures, the number of processes in 16 to achieve the best acceleration effect, the acceleration ratio can reach 8.5.

VII. RELATED WORK

The front-end processing is an indispensable part of the macro-micro coupling simulation platform, which is mainly responsible for the data input and data output processing of the core solver. For the macroscopic computational fluid dynamics simulation, the front-end processing is to perform the grid generation, parallel division and boundary condition definition of the simulation area. The back-end processing is the data processing and display of the solver's calculation results, which is convenient for the user to visualize the physical phenomena For the molecular dynamics simulation, the front-end processing needs to specify the information of the simulation area, configure the physical properties of the particle information, back-end processing needs from the micro-data to the macro data conversion, user-friendly from the macro level and Study the physical phenomenon.

There are many mature front and rear processing software, but most are single-scale processing tools, there are few multi-scale front and rear processing tools. In the field of computational fluid dynamics front-end processing, ICEM CFD, PointWise and Gridpro are commonly used in the generation of computational fluid dynamics [1]. These software can perform grid generation on solid or liquid simulation regions. Some software, like Gridpro, Generating a structured grid, TGrid is a professional, completely unstructured grid generation software that generates grids that are not geometrically complex and dimensioned. In the field of computational fluid dynamics, the software function is to reanalyze the simulation results data. Visualization, backward processing of the typical software are ParaView, GLview [12-14]. In the field of molecular dynamics research, there are few front-end processing software, back-end processing software such as MDAnalysis and other software is generally a laboratory product, for the theoretical researchers, after the MD simulation, according to their own experimental needs to write back-end processing Script, but scripting is limited.

VIII. CONCLUSION

This paper analyzes the architecture and module programming method of SALOME software platform, designs the unified front-end processing method of macro micro-coupling simulation in the field of microfluidics, and realizes the unified MPT front-end processing tool based on SALOME software platform. The data processing program is effectively optimized. Finally, the validity of the backend program optimization is verified by experiments.

REFERENCES

[1] Dong Yuxin, He Zheng. Integration of CAD / CAM and Grid Generation and CFD Software [J]. Journal of Applied Science, 2007, 34 (6): 43-45. DOI: 10.3969 / j.issn.1009-671X .2007.06.012.

[2] Zhang Hao. Based on the Salome platform for finite element analysis process automation software framework design [D]. Wuhan University of Technology, 2013.

[3] Henning M, Vinoski S. Advanced CORBAr Programming with C[C]. //ISBN 9-780201379273, Addison-Wesley A.7 INTERNET 83 A.7 Internet [39] The World Wide Web Consortium http://www.w3.org/ Publications of many standards and drafts related the the Internet and World Wide Web. 1999.

[4] Borgley. CORBA technology and its development [J]. Journal of Chongqing Dianli University, 2007, 12 (1): 15-17. DOI: 10.3969 / j.issn.1008-8032.2007.01.005.Hébert A. Integration of the DRAGON5/DONJON5 codes in the SALOME platform for performing multi-physics calculations in nuclear engineering[J]. Joint International Conference on Supercomputing in Nuclear Applications + Monte Carlo, 2014.

[5] [5] Song Shixue. Statistical physics in the distribution function [J]. Journal of Jinan University: Social Science Edition, 2000, (2): 35-37.

[6] The Visualization Toolkit (VTK). http://www.vtk.org/

[7] Schroeder W J, Avila L S, Hoffman W. Visualizing with VTK: a tutorial[J]. Computer Graphics & Applications IEEE, 2000, 20(5):20 - 27.

[8] Committee M. MPI: a message-passing interface standard[J]. Internat.j.supercomputer Applications, 1994, 20(2):179.

[9] Dagum, L, Menon, R. OpenMP: an industry standard API for shared-memory programming[J]. Omaonal N & Ngnrng, 1998, (1):46-55.

[10] Hébert A. Integration of the DRAGON5/DONJON5 codes in the SALOME platform for performing multi-physics calculations in nuclear engineering[J]. Joint International Conference on Supercomputing in Nuclear Applications + Monte Carlo, 2014.

[11] Ribes A, Bruneton A. Visualizing results in the SALOME platform for large numerical simulations: an integration of ParaView[J]. IEEE Symposium on Large Data Analysis & Visualization, 2014:119 - 120.

[12] Nunio F, Manil P. SALOME as a Platform for Magneto-Mechanical Simulation[J]. IEEE Transactions on Applied Superconductivity, 2012, 22(3):108-110.

[13] David G, Chevalier T, Meunier G. Unification of Physical Data Models. Application in a Platform for Numerical Simulation: SALOME[J]. IEEE Transactions on Magnetics, 2007, 43(4):1661 - 1664.