An aerodynamics-centric framework for multidisciplinary coupling analysis and its application

Xiong Jiang¹, Ziwei Wang²,¹, Fengshun Lu¹*, Duoneng Liu¹, Bendong Zhao¹ and Wenkai Wang¹

¹China Aerodynamics Research and Development Center, Mianyang, Sichuan, 621000, China
²National University of Defense Technology, Changsha, Hunan, 410073, China
*Corresponding author’s e-mail: lufengshun@cardc.cn

Abstract. There are many multifaceted problems in the aerospace engineering domain, which demand the multidisciplinary simulations for coupling analysis and the development of coupling frameworks. In order to address the issue of lacking domestic coupling framework, we propose an aerodynamics-centric framework for multidisciplinary coupling analysis, and then demonstrate the usage of it in developing a structured-unstructured grid coupling software, which is validated with an external store separation case study.

1. Introduction

The issues in aerospace engineering usually belong to multifaceted problems, which demand the multidisciplinary simulations for coupling analysis at high fidelity. Besides, new algorithms, solvers, physical models, and techniques with better mathematical and numerical properties are being developed and gradually mature. Consequently, standardization of disciplinary interfaces and the development of coupling frameworks will increase in importance with added simulation complexity [1]. For instance, Kestrel[2] is an integrating product that allows crossover between simulation of aerodynamics, structures, propulsion, and store separation targeting fixed-wing aircraft, which validates many of the goals outlined in the CFD (computational fluid dynamics) Vision 2030 report[1].

The domestic researchers have already performed much investigation on multidisciplinary coupling analysis. Guo et al.[3] studied the nonlinearity of forces and moments in a variable-sweep morphing aircraft and performed an unsteady flow simulation using unsteady CFD coupled with the flight control system. Huang et al.[4] derived a precise integration method (PIM) formula for the structural modal equations and then proposed a PIM-based CFD/CSD (computational structural dynamics) coupling method to study the aeroelasticity (i.e., the interaction between aerodynamic loads and the flexible structures). Qin et al.[5] presented an aerothermoelasticity computing method by coupling an in-house code HUNS3D and open source CSD analysis solver CalculiX[6] to predict the aerothermoelastic performance of hypersonic flight. However, to the best of our knowledge, there have been none domestic frameworks to cover the need in seamless integration with other disciplinary codes for enabling complex multidisciplinary analyses.

In order to address the aforementioned issue, we propose an aerodynamics-centric framework for multidisciplinary coupling analysis (hereinafter called ACEMUCA). The rest of the manuscript is organized as follows. Section 2 describes the architecture of ACEMUCA and the multidisciplinary coupling analysis method based on ACEMUCA. Section 3 demonstrates the usage of ACEMUCA in
developing a structured-unstructured grid coupling software, which is validated with an external store separation case study. Section 4 concludes our paper.

2. Aerodynamics-centric framework for multidisciplinary coupling analysis
ACEMUCA is an aerodynamics-centric framework for multidisciplinary coupling analysis and its architecture is depicted in figure 1. It can be seen that ACEMUCA has a hierarchical architecture.

Figure 1. Aerodynamics-centric framework.

2.1. ACEMUCA’s four layers
There are four layers in ACEMUCA, namely the client layer, the scientific computing middleware layer, the server layer and the hardware layer.

2.1.1. Client layer.
The client layer provides the interfaces between users and multidisciplinary coupling software, which can be divided into two parts. On the one hand, users can manage the analysis job and the execution commands through the friendly graphical user interface. On the other hand, users can also customize execution parameters for different discipline software.

2.1.2. Scientific computing middleware layer.
The scientific computing middleware layer is located between the client and server layers, and performs as a bridge for them. It is responsible for data transmission, job scheduling management, computational status monitoring, computational load balancing and other functions, which shields the complexity of underlying hardware environment for operational discipline software.

2.1.3. Server layer.
The server layer is critical for ACEMUCA and it contains four parts, namely public library, discipline plugin, interface adapter and coupling analysis software.

The public library part is a collection of several basic functions involved in multidisciplinary analysis. Each basic function exists in the form of an independent tool library, which can be called by upper modules. It consists of the following four libraries:
surface-surface interpolation library: it is utilized to realize the data exchange (e.g., aerodynamic force and displacement) between grid surfaces for coupling disciplines during their execution;

• surface-body interpolation library: it is employed to implement the data exchange between surface grid and space grid;

• grid deformation library: it fulfils the functionality of space grid deformation for the optimization design of aircraft shape and structural deformation;

• overset grid library: it performs the calculation of interpolation relationship for overlapped grids in the scenario of generating grids for rather complex configurations.

The **interface adapter** part builds a bridge between the public library and the discipline plugin. Since each public library has a standard interface specification and most of the discipline modules have complex and diverse internal data structure, it is necessary to realize the data conversion between the public library interfaces and the input/output interfaces of the discipline plugins through a well-defined interface adapter.

The **discipline plugin** part organizes the analysis modules of different disciplines into plugins, such as aerodynamics, structure, kinematics, acoustics, and control. It can bring much convenience for the further assembly of these plugins into specific coupling analysis software. Concretely speaking, a discipline module calls certain public library through the interface adapter, and then they are compiled into an independent plugin. Programmers can quickly construct coupling software for typical complex problems by assembling relevant discipline plugins.

The **coupling analysis software** part can solve multidisciplinary coupling problems in a loosely or tightly coupled manner. It consists of three pieces of coupling software, namely the CFD/CSD coupling software, the aeroacoustics coupling software and the aerodynamics/kinematics/flight-control coupling software. When designing coupling analysis software, users are required to write a simple work-flow program to organize specific discipline plugins in such an appropriate order that the multidisciplinary coupling problems can be correctly simulated.

### 2.1.4. Hardware layer.

The hardware layer is the fundamental infrastructure for multidisciplinary coupling simulations. It includes large-scale supercomputers and provides the abundant computational resources for various coupling analysis software.

### 2.2. Multidisciplinary coupling analysis method based on ACEMUCA

Figure 2 demonstrates the workflow of multidisciplinary coupling analysis based on ACEMUCA, which mainly includes six steps:

• **coupling strategy analysis**: determining the coupling strategy after the relevant disciplines of the problem is analysed;

• **data communication pattern evaluation**: evaluating the data communication pattern (i.e., content, scale and frequency) through theoretical analysis;

• **interface adapter construction**: based on the coupling strategy and the data communication pattern, selecting the appropriate public library and constructing the interface adapter between disciplines;

• **discipline module standardization**: implementing the discipline plugin by calling interface adapter for each discipline module;

• **coupling software development**: customizing the execution parameters for coupling software, and implementing the main program that calls the discipline plugins to accomplish various coupling analysis tasks;

• **coupling analysis calculation**: performing the coupling analysis calculation on large-scale supercomputers and obtaining the desired results.
3. Structured-unstructured grid coupling software development with ACEMUCA

It is well known that the time-consuming mesh generation process needed by the flow field solver is a great obstacle to widespread use of CFD methods. At present, the hybrid grids method is usually utilized to cope with the simulation of complex configurations, which can effectively combine the advantages of structured, unstructured, or Cartesian grid approach [7-10]. Consequently, a monolithic solver is developed to handle the hybrid grids that discrete the computational domain.

From the perspective of modern software engineering, this manner is not flexible enough to cope with the ever-changing demand and timely insertion of new techniques. Different from that, we take advantage of two in-house solvers (i.e., a structured-grid aerodynamic solver and an unstructured-grid aerodynamic solver) and develop the hybrid grid coupling software based on ACEMUCA. Note that we utilize overlapped grids to perform the coupling analysis.

3.1. Analyzing coupling strategy

In the aerodynamic simulations with hybrid grids, we usually generate a body-fitted structured grid around the wall and an unstructured grid in the background so as to improve the simulation accuracy on the premise of the same number of grid cells. Note that the aerodynamic analysis needs to exchange data between the unstructured and structured grids in the internal iteration of each physical time step. Therefore, the aerodynamic analysis adopts the tightly-coupled strategy.

3.2. Evaluating data communication pattern

Based on the overset grid library, the structured-grid aerodynamic solver and the unstructured-grid aerodynamic solver conduct the data exchange between each other in the sub-iteration of each physical time step. The relevant communication content is the flow field variables falling in the overlapped area. The scale of communicated data volume depends on the size of the overlapped area, which is generally an order of magnitude lower than that of the entire grid.

3.3. Constructing interface adapter

According to the coupling strategy and the data communication pattern, we have to construct interface adapters for the overset grid public library.
The name of the interface adapter for overset grid is *KovalevskayaData* and its parameters are listed in Table 1.

| name   | type | description                              | name   | type | description                              |
|--------|------|------------------------------------------|--------|------|------------------------------------------|
| ng     | int  | number of grid groups                    | ctp(nc)| int  | volume cell types                        |
| np(ng) | int  | number of processes for each grid group  | nws    | int  | number of wall surfaces                  |
| npt    | int  | number of grid points                    | wsf(nws)| int | surface cell index for each wall surface |
| xx(npt)| double| x coordinates of all grid points         | dct    | int  | number of donor cells in current process |
| yy(npt)| double| y coordinates of all grid points         | srk(dct)| int | process index of \(i^{th}\) interpolation cell |
| zz(npt)| double| z coordinates of all grid points         | scidx(dct)| int | cell index in the source process of \(i^{th}\) interpolation cell |
| nsf    | int  | number of surface cells                  | tcidx(dct)| int | cell index in current process for donor cell of \(i^{th}\) interpolation cell |
| nd(4,nsf)| int | node index for each surface cell        | lg(8*dct)| double | interpolation coefficients for each donor cell |
| mat(2,nsf)| int | adjacent volume cell indexes of surface cell | blk(nc)| int | calculation type for each cell |
| nc     | int  | number of volume cells                   |        |      |                                          |

### 3.4. Standardizing discipline module

The coupling software is related to a structured-grid aerodynamic solver and an unstructured-grid aerodynamic solver, whose flow field exchange relevant to the overlapped area is realized through the overset grid interface adapter between them. Either of them calls the overset grid library and then is compiled into a standard plugin in the form of a dynamic library.

### 3.5. Developing coupling software

The development of coupling software mainly includes two parts: customizing the execution parameters and implementing the main program that accomplishes various coupling analysis tasks.

The parameters consist of the path for input/output files, operating options for aerodynamic/structure discipline plugin, and execution controls of the coupling software. These parameters can be demonstrated on the graphical user interface with graphic controls such as lists, checkboxes, and tables provided by the QT graphic library, which is convenient for users to operate the entire coupling analysis software.

The workflow of coupling analysis is implemented in a main program by calling the structured-grid plugin, the unstructured-grid plugin and the overset grid public library. Then the main program is compiled into an executable file, which can be triggered to perform various coupling analysis tasks.

### 3.6. Performing coupling analysis

#### 3.6.1. External store separation configuration.

The selected configuration is the standard model [11] of the U.S. Air Force Laboratory. It has relatively complete wind-tunnel experimental data and has been repeatedly used in the evaluation of numerical methods. The parameters of the configuration are presented in figure 3.
3.6.2 Hybrid grids.
Figure 4 shows the hybrid grids for the configuration. An unstructured grid is generated for the wing and multi-block structured grid for the external store. Note that the background grid is also unstructured. The structured grid is moving following the trajectory of the external store since its separation.

3.6.3 Results.
Figure 5 compares the computational results and the experimental data relevant to the centroid displacement of the external store with time. It can be seen that under the action of gravity and ejection force, the external store moves to the bottom of the wing (z coordinate), and its movement amplitude is the largest among the three directions. At the same time, the external store moves to the
Figure 5. Comparison of the store position (x,y,z) between simulation and experiment.

Figure 6. Comparison of the parameters STA between simulation and experiment.
Figure 7. Comparison of the parameters CSA between simulation and experiment.

rear of the wing (x direction) under the action of resistance. In the lateral direction (y direction), the external store first moves to the inner side of the wing, and then after about 0.35 seconds, the external store starts to move to the outside of the wing, which is accurately captured by the simulation although the moving range is small. Generally speaking, the computational linear velocity and displacement of the centroid both well match those from the experimental data within the displayed time range (0.5 seconds).

Figure 6 and Figure 7 show the attitude angle of the external store over time. Note that STA and CSA denote pitch angle and yaw angle, respectively.

From figure 6 we observe that the calculated maximum pitch angle and starting downward-pitch position are in good agreement with the experimental values. Concretely speaking, the external store moves in the following manner: (1) in the initial stage of separation, the ejection force produces a larger head-up moment; (2) after about 0.055 seconds, the ejection force disappears, then the aerodynamic force and torque acting on the external store begin to play a leading role, which produces a head-down moment; (3) after 0.2 seconds, the external store starts to move downward.

Figure 8. Static pressure distribution at certain slice in the x-direction

Figure 7 presents that the external store deviates to the outside of the wing at separation, and the deviation begins to appear after 0.35 seconds, which affects the accuracy of subsequent calculations.
Figure 8 shows the instantaneous static pressure distribution of the flow field in the x-direction at a certain moment. It can be seen that the static pressure contours smoothly transition between the computational domain of external store and that of the wing, indicating that the structured grid for external store and the unstructured grid for wing are correctly assembled.

4 Conclusions
The multidisciplinary simulations for coupling analysis and the development of coupling frameworks are in great demand for the aerospace engineering domain. In order to address the issue of lacking domestic coupling framework, we have proposed an aerodynamics-centric framework named ACEMUCA for multidisciplinary coupling analysis. Based on ACEMUCA, we have developed a structured-unstructured hybrid grid solver and validated it with an external store separation case study.

Acknowledgments
This work was supported by the Science and Technology Innovation 2030 Major Project under grant no. 2020AAA0104800 and the Natural Science Foundation of China under grant no.61903364.

References
[1] Slotnick, J., Khodadoust, A., Alonso, J., et al. (2014) CFD Vision 2030 Study: A Path to Revolutionary Computational Aerosciences. Hampton: Langley Research Center, NASA/CR–2014-218178.
[2] McDaniel, D.R., Tuckey, T.R., Morton, S.A. (2017) The HPCMP CREATE™-AV Kestrel Computational Environment and its Relation to NASA’s CFD Vision 2030. In: 55th AIAA Aerospace Sciences Meeting. Grapevine. pp. 1-25.
[3] Guo, Z.Z., Dai, G., Yang, H., et al. (2020) Unsteady flow simulation of a variable-sweep morphing aircraft coupled with flight control system. International Journal of Modern Physics B, 34(14-16): 2040073.
[4] Huang, C.D., Huang, J., Song, X., et al. (2020) Aeroelastic Simulation Using CFD/CSD Coupling Based on Precise Integration Method. International Journal of Aeronautical and Space Sciences, 21(3): 750-767.
[5] Qin, M.Z., Cheng, M., Mian, H.H., et al. (2016) Fluid-Thermal-Structural Interaction Simulation with coupled CFD/CSD method. In: 13th International Bhurban Conference on Applied Sciences and Technology. Islamabad. pp. 440-446.
[6] Dhondt, G., Wittig, K. (2020) CALCULIX - A Free Software Three-Dimensional Structural Finite Element Program. http://www.calculix.de .
[7] Sang, W., Li, F. (2007). An unstructured/structured multi-layer hybrid grid method and its application. Int. J. Numer. Meth. Fluids, 53: 1107-1125.
[8] He, X., Zhang L., Zhao, Z., et al. (2012) The research and development of structured-unstructured hybrid CFD software. In: 9th Asian Computational Fluid Dynamics Conference. Nanjing. pp. 1-5.
[9] He, X., He, X.Y., He, L., et al. (2015) HyperFLOW: A structured/unstructured hybrid integrated computational environment for multi-purpose fluid simulation. Procedia Engineering, 126: 645-649.
[10] Corrigan, A., Kailasanath, K., Liu, J.H., et al. (2012) A Hybrid Grid Compressible Flow Solver for Large-Scale Supersonic Jet Noise Simulations on Multi-GPU Clusters. In: the 50th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition. Nashville. pp. 1-6.
[11] Heim, E.R. (1991) CFD wing/pylon/finned store mutual interference wind tunnel experiment. Tennessee: Arnold Engineering Development Center.
[12] Xiao, Z.Y., Jiang, X., Mou, B., et al. (2010) Numerical Simulation of Dynamic Process of Store Separation in Parallel Environment. Acta Aeronautica et Astronautica Sinica, 31(8): 1509-1516.