

Design of dynamic inversion control law based on numerical virtual flight for a supermaneuverable missile model

R. Ma^{1,2}, N.H. Wang², X. H. Chang^{1,2}, S. Zhang² and L. P. Zhang^{1,2}

Corresponding author: zhanglp_cardc@126.com

1. State Key Laboratory of Aerodynamics, China Aerodynamics Research and Development Center,
Mianyang Sichuan 621000, China;

2. Computational Aerodynamics Institute, China Aerodynamics Research and Development Center,
Mianyang Sichuan 621000, China.

Abstract: Increasing demands on maneuverability and agility of aero-vehicles require much superior unsteady performance and higher quality flight control systems. Traditional design methods of flight control laws based on linear assumption with small perturbations are no longer suitable. So numerical virtual flight (NVF), a multi-physics method coupling computational fluid dynamics (CFD), rigid body dynamics (RBD), flight control system (FCS) and other discipline simulation tools, has been paid more and more attention in recent years, due to its capability of obtaining the closed-loop response characteristics during maneuvering flight via numerical simulations. In this paper, a CFD-based integrated solver of unsteady aerodynamics, kinematics and flight control system is presented, and is applied to simulating the pitching maneuver process of a super maneuverable missile model. A parallel unstructured dynamic overlapping grid technique is adopted to generate proper moving grids over the deflecting afterbody rudders, while the unsteady flow field is simulated by a parallel URANS solver based on cell-centered finite volume method. For the flow/kinematic coupled problem, the 6-DOF equations are solved by an explicit or implicit method with the URANS CFD solver. A nonlinear dynamic inversion method is implemented to the control law design. Simulations and analysis of the pitching maneuver process are carried out to improve the flight control law. By adjusting the gain factors and adding an integrating link, we obtained a set of optimized control parameters, and realized good control performance.

Keywords: numerical virtual flight; nonlinear dynamic inversion; unstructured dynamic overlapping grids; super maneuverable missile pitching.

1 Introduction

In recent years, increasing demands on maneuverability and agility of aerospace vehicles require much superior performance under unsteady conditions and higher quality flight control systems. On one hand, the flight envelopes of next generation aircrafts are becoming wider and wider; the aerodynamic characteristics exceeds the linear interval and the traditional control laws based on linear assumptions with small perturbations are no longer suitable as a result. On the other hand, the next generation aircrafts will have much higher maneuvering performance, leading to stronger unsteady effects, and strong coupling of aerodynamics and kinematics, as well as the flight control law and even structural dynamics (aero-elastics). To solve these complex and strong coupled multi-physics problems, virtual flight experimental technologies [1-3] have been developed in recent years. With the rapid development of computer science and numerical methods, CFD-based multi-physics simulation methods have been paid more and more attention. These works focus on obtaining closed-loop response characteristics during maneuvering flight via numerical simulations, so this kind of

simulation methods is called as numerical virtual flight (NVF) or digital flight [4].

In the past decade, so many works on developing an integrated software system of numerical simulation, virtual flight and optimization have been reported, such as the AVT-161 program of the Research Treaty Organization [5], the SikMa (Simulation of Complex Maneuvers) program [6] and the Digital-X program [7-8] of DLR, the CREATE (Computational Research and Engineering Acquisition Tools and Environment) Program [9-10] of the DoD HPCMP in America. Especially in the CREATE-AV (aero-vehicle) sub-program, a 12-year long-term program since 2008, two multi-physics simulation systems, Kestral [11-12] and Helios [13-14], were developed for fixed-wing aircrafts and helicopters, respectively. The main goal of Kestral and Helios is to supply necessary virtual flight simulation tools for the next generation aircrafts. Some exciting results by these coupled solvers have been reported in a series of papers [11-14].

Using this kind of NVF tools, we can evaluate and further improve the flight control law for fast maneuvering flight, which will result in the revolution and innovation of aircraft design patterns. Even in the initial concept design phase, we can consider the effects of flight control system to avoid the risks of flight testing later, and further improve and optimize the flight control system in the detailed design phase.

For a CFD/RBD/FCS coupled NVF system, three key issues should be considered: dynamic mesh generation, unsteady flow simulation and multidisciplinary coupling method. In this work, we focus on CFD/RBD coupling method and control law design, based on the authors' previous work [15-18]. To simulate the pitching process of a super maneuverable missile model, we developed a CFD/RBD/FCS coupled NVF solver based on URANS simulation on unstructured dynamic grids and solution of flight mechanics (FM) equations. Due to the strong nonlinear effects of the missile model under high angle of attack and strong hysteresis effects caused by high-speed maneuvering, the traditional linear control law is no longer suitable. Hence, we adopted nonlinear dynamic inversion method [19] to design the control law, instead of the widely-used PID control approach [20-21]. The pitching process of the model with a control command (the angle of attack pitching-up from 0° to 30°) was simulated by the integrated solver and the influences of control parameters were analyzed. By adjusting the gain factor and adding an integrating link, we obtained a set of optimized control parameters, and realized good control performance.

2 Numerical methods of CFD-based NVF

2.1 Main procedure of the CFD/RBD/FCS coupled NVF solver

As discussed in the introduction, in order to simulate aerodynamics/kinematics/flight-control coupled problems, a multi-physics integrated solver should be set up firstly, in which the modules of dynamic grid generation, the unsteady flow simulation, the computation of 6-DOF equations and the flight control law should be coupled within a unified framework. Here in this work, we developed a CFD/RBD/FCS coupled solver as shown in Fig. 1. The procedure can be listed as follows:

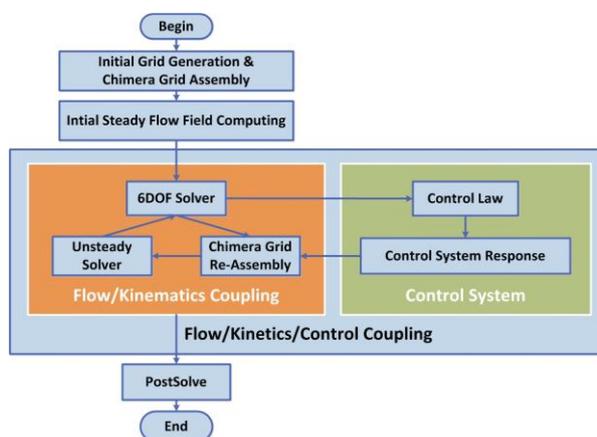


Figure1. Flow chart of the CFD/RBD/FCS coupled NVF solver

(a) The program starts with initial grid generation by some commercial grid generation software and unstructured overlapping grid assembly, and the initial steady flow field and the aerodynamic forces are obtained by the URANS solver for the initial steady case;

(b) Input the aerodynamic forces into the 6-DOF flight dynamics solver to obtain the 6-DOF information of the $(n+1)$ th time step;

(c) Input aerodynamic forces and the 6-DOF information of the model to the flight control law module to obtain the deflection of the control surfaces of the $(n+1)$ th time step;

(d) Assemble the overlapping grids according the 6-DOF information of the model body and control surfaces of the $(n+1)$ th time step;

(e) Obtain the $(n+1)$ th time step unsteady flow field and the aerodynamic forces by the URANS solver;

(f) Repeat steps (b)-(e) until the goal time.

In our previous work [15-17], the authors had developed a dynamic hybrid grid generation technique, which integrates the grid morphing and the local remeshing. For the maneuvering missile model considered in this work (as shown in Fig.2, and the 'X' configuration of the four rudders will be studied only), the afterbody rudders will deflect simultaneously by a same angle during pitching process. In order to generate the moving grids automatically, an unstructured dynamic overlapping grid technique is integrated further into our previous dynamic hybrid grid generator to handle the deflection of the control surfaces. An improved parallel implicit hole-cutting approach is developed to improve the efficiency and to enhance the robustness of overlapping grid assembling [22]. Figure 3 shows the initial hybrid grids. The grids of the body and the rudders are generated separately, and then are assembled by the parallel implicit hole-cutting approach. The initial hole-cutting results are shown in Figure 4a. During the pitching process, the dynamic overlapping grids over the afterbody rudders at some typical states are shown in Figure 4b-f, while the grids over the body are rotated with the pitching angles using a rigid body-fixed approach for simplicity.



Figure 2. The configuration of the maneuvering missile model

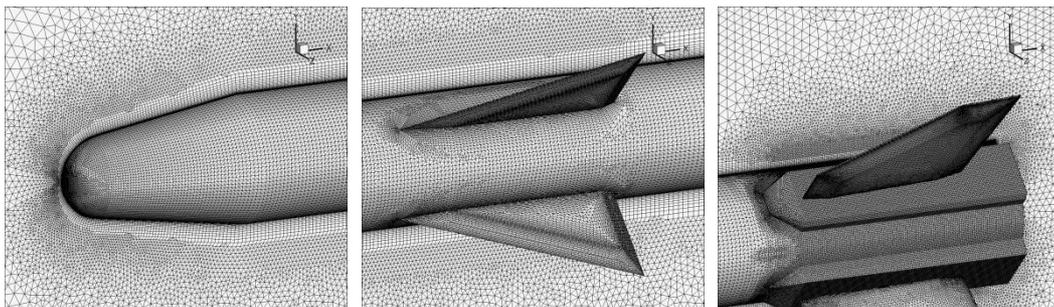
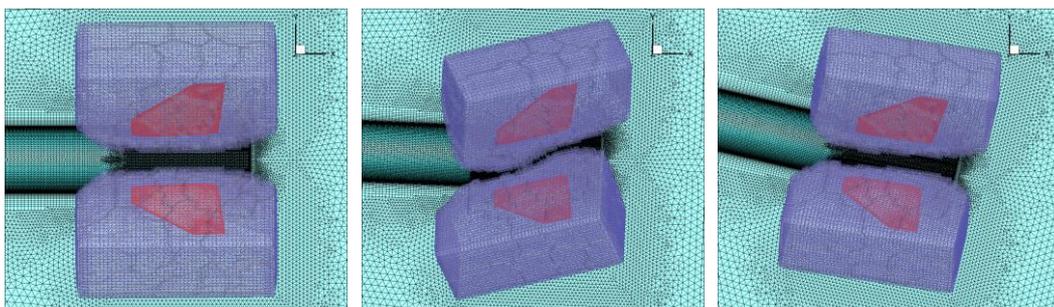


Figure 3. The initial grids over the missile model



(a) Initial case, $t = 0.0s$

(b) $t = 0.1s$

(c) $t = 0.2s$

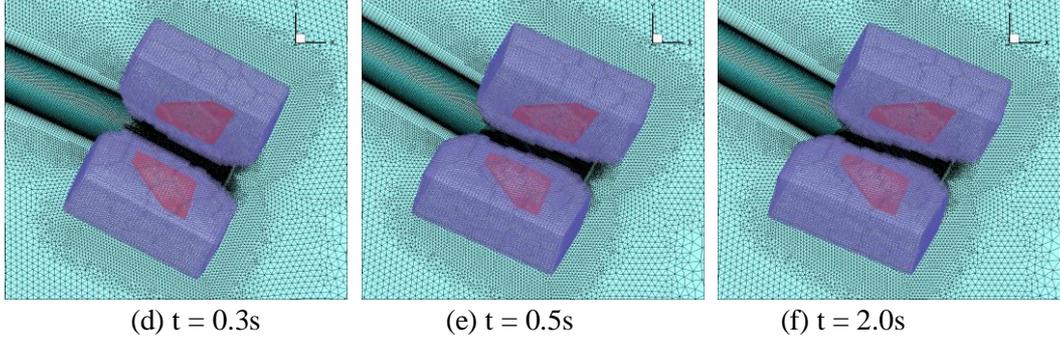


Figure 4. The dynamic overlapping grids over the deflecting rudders.

The unsteady flow solver is based on the well-known second-order finite volume discretization of the URANS equations in an ALE framework. The dual time stepping method is adopted to achieve second-order accuracy of temporal discretization. To improve the accuracy of unsteady flow simulations, the geometry conservation law (GCL) on moving grids is paid more attention in our previous work [18]. Some verification and validation test cases have been shown in these papers [15-18], which will not be repeated here.

2.2 Coupling strategy: Loose Coupling and Strong Coupling

For aerodynamics/kinematics/flight-control coupled problems, the right hand terms of URANS equations and the 6-DOF governing equations are related to both the flow variables and the kinematics variables; therefore, the governing equations for this coupled system can be written as:

$$\begin{cases} \frac{\partial(QV)}{\partial t} = R(Q, U) \\ \frac{dU}{dt} = P(Q, U) \end{cases} \quad (1)$$

Unfortunately, the flow variables Q are implicit in term P because the aerodynamic forces and moments are the integrated form of flow variables. And the 6-DOF variables U are implicit in term R too because the position, velocity and orientation, angular velocity of the moving body influence the computational mesh directly. Therefore, a uniform temporal scheme for Eq. (1) is hardly to achieve, a feasible way is to solve the URANS equations and the 6-DOF governing equations separately with time marching. In other words, the monolithic method is hardly to achieve and we have to deal with the problem by partitioned methods [23].

The conventional method is the loose coupling approach in which a general cycle can be described as following sequential procedure:

- 1) Suppose all the 6-DOF variables and the flow variables at the n th real-time step are already known (initial conditions);
- 2) Solve the 6-DOF equations for kinematics variables at the $(n+1)$ th time level according to the pressure field at the n th time level;
- 3) Adjust the computational mesh for the $(n+1)$ th real-time step according to the new kinematics variables;
- 4) Execute the URANS flow solver on the updated mesh and update the flow variables at the $(n+1)$ th time level.

So in the loose coupling approaches, the 6-DOF system is advanced explicitly. It's obvious that there always exist a lag between the information of the subsystems in time advancing. This may result in instability sometimes.

If an implicit method is adopted to solve the 6-DOF governing equations, sub-iterations should be carried out to get a convergent solution. The implicit method, in which the interaction between the URANS equations and the 6-DOF governing equations was carried out in each sub-iteration, is called as the 'strong coupling' approach (or 'fully implicit' method) [24].

At the beginning of the sub-iterations procedure, the initial value of U ($U(0)$) should be specified. The way to get $U(0)$ is generally called as the ‘prediction’ step, and various explicit methods can be employed here. The following sub-iterations are called as the ‘correction’ steps. The value of $U(0)$ may influence the convergence rate of sub-iterations; however, the converged solutions with different initial value are consistent. In this work we let $U(0)=U_n$.

The procedure of the fully implicit scheme is almost similar to the loose coupling one except for the updating of information between subsystems. For the loose coupling approaches, the information is updated in each real time step, but for the strong coupling ones (see Figure 5), the information is updated also in each sub-iteration. The 6-DOF governing equations are solved firstly in each sub-iteration, and then the computational mesh is updated according to the new kinematics variables. After that, the sub-iteration for URANS equations is executed on the updated mesh. Finally, a check for convergence is carried out at the end of each sub-iteration. If the convergence tolerance is reached, the variables at the $(n+1)$ th time-step is then updated; otherwise, the sub-iterations would be carried out continuously until reaching a convergence solution.

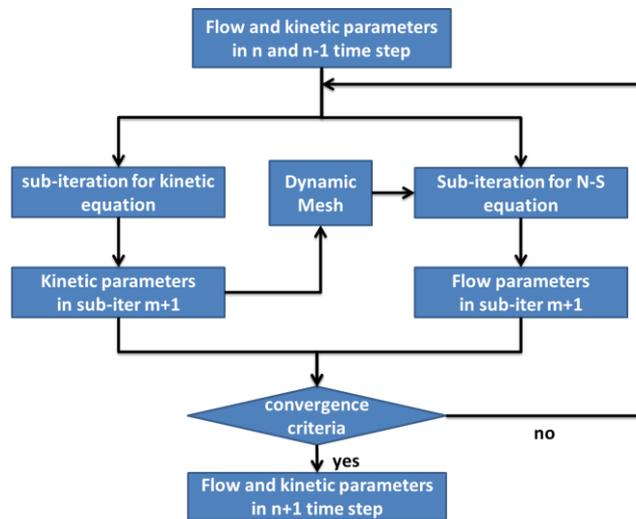


Figure 5. Sub-iteration procedure of the strong coupled solver.

Both the explicit and the implicit coupled strategies have been integrated into our NVF solver. What should be mentioned here is that the flight control law is coupled with an explicit manner. For each real-time step, the command from the flight control law will be updated according to the current kinematic parameters, and then the control law outputs the deflection of control surfaces to the next time step. In the next section, we will discuss the control law design approach in details.

3 Design of nonlinear dynamic inversion control law

As we know, the supermaneuverable missile model presents strong nonlinear characteristics at high angle of attack. And the rapid maneuver may also result in strong unsteady hysteresis effect, so the nonlinear dynamic inversion method is adopted to design the control law.

An aero-vehicle (controlled plant) is essentially an affine nonlinear system and can be decoupled by the feedback linearization method. This guarantees the feasibility of adopting the nonlinear dynamic inversion method in the design of flight control system. The fundamentals of dynamic inversion method can be described as follows: cascades an inversion system with the controlled plant, which can compensate it into linear system that can be designed with linear system theory. So it is very suitable to handle the nonlinear system. However, the mathematical model of the plant should be accurate enough for inversion system. But in actual engineering applications, we can hardly supply the absolute accurate aerodynamic model, some prediction and modelling errors in aerodynamic performance, especially in high angle of attack with strong flow separation, are unavoidable. This

may lead to the whole system instability in some cases, and brings challenges to adopt the dynamic inversion method in flight control system design. The goal of flight control law design is to ensure the flight stability, robustness, maneuverability and agility of the controlled plant in the condition of prediction errors. For the pitching-up process of the supermaneuverable missile model in this work, the specific goals of flight control law are 1) to reduce or even eliminate the static error to improve the control accuracy; 2) to control the overshoot phenomenon to decrease large structural overloads; 3) to reduce the response time and thus to improve the agility.

Dynamic inversion method can be divided into two types: output feedback and state feedback. The output feedback method requires the multistep derivatives of the output which is often difficult to get and even impossible to achieve sometimes. The state feedback method requires all the state variables, and is relatively easier to implement for the supermaneuverable missile pitching process. So we use this kind of feedback method in this work. For an affine nonlinear system in the following equation,

$$\dot{v}(t) = \dot{x}(t) = f(x(t)) + G(x(t))u(t) \quad (2)$$

the state feedback method can be described as follows.

If $G(x)$ is invertible, then:

$$u(t) = G^{-1}(x(t))[-f(x(t)) + v(t)] \quad (3)$$

so the system can be reduced to a linear dynamic system:

$$\dot{v}(t) = \dot{x}(t) = k[x_c(t) - x(t)] \quad (4)$$

For a given signal $x_c(t)$, its error between the actual state $x(t)$ and $x_c(t)$ is taken as the input of the linearized system, so that the signal can be tracked.

For the supermaneuverable missile pitching process, the time-scale separation technology is adopted when designing the dynamic inversion control law. The kinematics equations are considered as the slow loop (outer loop), while the kinetics equations are considered as the inner loop (fast loop). When designing the slow loop control law, the impacts of the fast variables are ignored; and for the fast loop, the slow variables are approximately considered constant.

Firstly, for the outer loop, introducing the kinematics equations into Equation (2), and rewriting it into the form of Equation (2), we have:

$$\frac{d\alpha}{dt} = q \Rightarrow \dot{\alpha}(t) = 0(t) + 1(t)q(t) \quad (5)$$

in which α is the angle of attack, q is the pitching angular velocity. Equation (5) can be further written as the form of Equation (3):

$$q(t) = 1(t)^{-1}[0(t) + q(t)] \quad (6)$$

Substituting Equation (4) into Equation (6) if we consider \dot{x} as q , the slow loop control law can be obtained:

$$q(t) = k[\alpha_c(t) - \alpha(t)] \Rightarrow q_c = k_\alpha(\alpha_c - \alpha) \quad (7)$$

where q_c is the command angular velocity, k_α is the slow loop control gain factor.

Then the inner loop is designed. Here the influences of the dynamic derivative terms are ignored, because dynamic derivative terms generally play a damping role, which only affect the response process, and have little effects on the final steady-state results. Meanwhile, the cost of calculating dynamic derivative terms by unsteady CFD method is relatively large. Therefore, the aerodynamic model of the missile's pitching channel can be simplified as follows:

$$m_a = QScC_m(\alpha, \delta) \quad (8)$$

in which M_a is the pitching moment, $Q = \rho V^2/2$ is the inflow dynamic pressure, ρ is the inflow density, V is the inflow velocity, S is the reference area, c is the reference length, C_m is the static pitching moment coefficient, which is a function of the angle of attack α and the rudder deflection angle δ . We can assume the pitching moment coefficient is approximately linear with the rudder deflection angle, i.e.:

$$C_m(\alpha, \delta) = \tilde{C}_m(\alpha) + C_{m\delta}(\alpha)\delta \quad (9)$$

where $\tilde{C}_m(\alpha)$ and $C_{m\delta}(\alpha)$ represent the pitching moment coefficient and the derivative of pitching moment with respect to deflecting angle of the rudder when $\delta = 0^\circ$, respectively. These two quantities are related only to the angle of attack α and can be obtained by linear interpolation or numerical differentiation from the static aerodynamic database.

Thus, the kinetics equation of fast loop is:

$$\frac{dq}{dt} = \frac{m_a}{I} = \frac{QSc[\tilde{C}_m(\alpha) + C_{m\delta}(\alpha)\delta]}{I} \quad (10)$$

in which I is the inertia of pitching channel. Similar to the slow loop design process, combining with Equations (2)~(4), the fast loop control law can be obtained with the fast loop control gain factor k_q :

$$\delta = \frac{-QSc\tilde{C}_m(\alpha) + Ik_q(q_c - q)}{QScC_{m\delta}(\alpha)} \quad (11)$$

Due to the approximation between the aerodynamic model and the actual plant, the aerodynamic moment formed by the rudder deflection according to Equations (7) and (11) is often different from the expected command value, so there usually exists a static error ($e_{ss} = \alpha_c - \alpha$) between the steady-state value of the angle of attack α and the command one α_c . The static error can be reduced by increasing the gain factor to some extent, but it cannot be completely eliminated. So an integrating link is introduced into the slow loop for reducing the static error, then Equation (7) can be written with the integrating gain factor $k_{i\alpha}$ as follows:

$$q_c = k_\alpha(\alpha_c - \alpha) + k_{i\alpha} \int (\alpha_c - \alpha) dt \quad (12)$$

In realistic engineering applications, the flight control devices also have some physical limitations, such as the maximum rudder angle δ_{Limit} , the maximum rudder deflection angular velocity $\omega_{\delta Limit}$. These nonlinear physical limitations may lead to divergence of the control system. Therefore, it is necessary to restrict the contribution of the integrating link and the command angular velocity in the control law. This can enhance the stability of the system and improve the dynamic response characteristics. In summary, the flight control system can be illustrated in Figure 6.

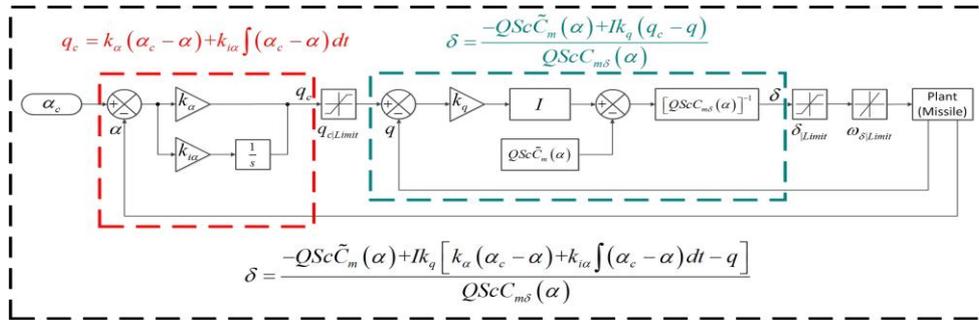


Figure 6. Sketch of the flight control system

4 Rapid pitching process simulation of the model

In this section, we use the aforementioned virtual flight numerical simulation platform combined with the control law designed in Section 3 to simulate the supermaneuverable missile rapid pitching process and trim the control law by adjusting the gain factors.

4.1 The static aerodynamic characteristics

The configuration of the supermaneuverable missile has been shown in Figure 2. The inflow Mach number Ma is 0.6 and the Reynolds number Re is 2×10^6 per meter. Since the Reynolds number is very large, so the well-known SA turbulence model is adopted in the following simulations. The inflow

temperature is set to 288.15K in the wind tunnel experimental environment. The reference length is taken as the missile's diameter $d=0.16002\text{m}$, and the reference area S is $\pi d^2/4=0.010062\text{m}^2$. The moment reference point (x_c, y_c, z_c) is $(9d, 0, 0) = (1.44018\text{m}, 0\text{m}, 0\text{m})$, and the inertia of pitching channel I is 63.5kgm^3 .

A half-model is adopted for saving computational cost since the pitching process is only considered in this study. The origin of coordinates is located at the apex of the missile's head. The boundary layer is discretized with prismatic elements and tetrahedral elements are adopted in farfield. The height of the first layer is $1 \times 10^{-6}\text{m}$ and the number of total cells is about 15 million to capture the flow separation more clearly.

Here the coordinate system adopts the one generally employed in flight mechanics, i.e., the x -axis is in the missile plumb symmetry plane and parallel to the missile design axis to the head, the y -axis is perpendicular to the missile plumb symmetry plane to the right of the missile body, and the z -axis is in the missile plumb symmetry plane and perpendicular to the xy plane pointing to the earth. In this coordinate system, the pitching-up moment and the upward deflection of the rudder leading edge are defined as positive values.

The pitching moment coefficients for different angles of attack and rudder deflection angles are shown in Table 1 and Figure 7. From Table 1, the trimmed deflection angle of the rudders δ_c is identified as -10.284° for the expected pitching angle of attack $\alpha_c=30^\circ$. The derivatives of pitching moment with respect to deflecting angle of the rudder at $\delta = 0^\circ$ can be derived from the values of the pitching moment coefficients (see the last column in Table 1). It can be seen that the missile model is basically static stable in the selected range ($-5^\circ < \alpha < 45^\circ$), and the pitching moment is relatively linear when the rudder deflections and the angles of attack are both small. However, in the cases of higher angles of attack or larger rudder deflections, the nonlinearity appears, and the missile even becomes static unstable. At these situations, it is difficult to design the control law by traditional linear method.

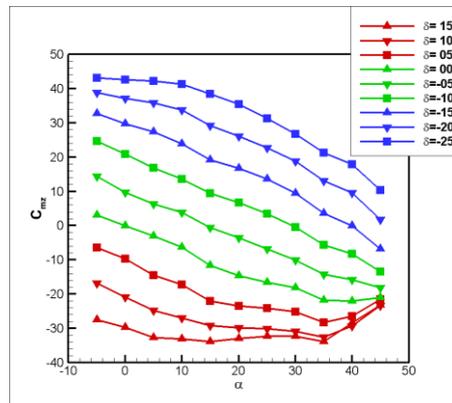


Figure 7. The static moment coefficients with different angles of attack of the missile and deflection angles of the rudders

Table 1. The static moment coefficients and the derivative with respect to the deflecting angle

	$\delta=15^\circ$	$\delta=10^\circ$	$\delta=5^\circ$	$\delta=0^\circ$	$\delta=-5^\circ$	$\delta=-10^\circ$	$\delta=-15^\circ$	$\delta=-20^\circ$	$\delta=-25^\circ$	$C_{m\delta}$
$\alpha=-5^\circ$	-27.474	-16.850	-6.3866	3.0357	14.394	24.703	32.661	38.844	43.145	-1.7655
$\alpha=0^\circ$	-29.761	-20.856	-9.7543	-0.0281	9.6999	20.811	29.712	37.179	42.588	-1.8087
$\alpha=5^\circ$	-32.764	-24.748	-14.451	-3.0909	6.3304	16.805	27.434	35.883	42.253	-1.8754
$\alpha=10^\circ$	-33.102	-26.959	-17.179	-6.2849	3.8229	13.571	23.827	33.693	41.351	-1.8613
$\alpha=15^\circ$	-33.886	-29.162	-22.069	-11.670	-0.6297	9.3983	19.151	29.196	38.439	-1.8081
$\alpha=20^\circ$	-32.943	-29.804	-23.517	-14.657	-3.6156	6.7183	16.645	26.033	35.379	-1.7081
$\alpha=25^\circ$	-32.388	-30.096	-24.173	-16.610	-6.8939	3.4084	13.532	22.639	31.219	-1.5902
$\alpha=30^\circ$	-32.359	-30.880	-25.174	-18.138	-10.059	-0.4689	9.3913	18.731	26.692	-1.4763
$\alpha=35^\circ$	-33.904	-32.556	-28.242	-21.784	-14.248	-5.6978	3.6054	13.117	21.210	-1.3779
$\alpha=40^\circ$	-28.442	-29.320	-26.527	-22.017	-15.874	-8.2208	-0.0934	9.5981	17.897	-1.1585
$\alpha=45^\circ$	-23.283	-23.380	-21.594	-20.994	-18.195	-13.462	-6.8650	1.7481	10.269	-0.8388

4.2 Typical cases with different gain factors

As mentioned before, in actual situations, there are some physical limitations for the flight control device. In this case, the maximum rudder deflection angle δ_{Limit} and the maximum rudder deflection velocity $\omega_{\delta Limit}$ are specified as $\pm 25^\circ$ and $250^\circ/s$, respectively. The initial value of control gain factors can be obtained through flight mechanics (FM) simulation. Generally, the fast loop gain factor is set to be no more than 1/5 of that of the rudder deflection velocity, and then the slow loop gain factor is set to be about 1/5 of the fast loop one to ensure the time-scale separation is established. According to the flight mechanics simulation, we have obtained a group of initial control gain factors, $k_\alpha=5$ and $k_q=25$ (named as Case1 in Table 2). For comparison, a case (Case2 in Table 2) with larger gain factors ($k_\alpha=10$ and $k_q=50$) is also considered. Because the static error cannot reach the expected value as shown in Section 4.3, then two cases (Case3 and Case4 in Table 2) with integrating link are simulated also. In spite of these cases, the overshoot phenomenon is very serious in Cases 2-4, so two other cases (Case5 and Case6 in Table 2) are considered with different limitations on control parameters.

Table 2. Six typical cases for different gain factors with and without integrating link

Case	k_α	k_q	$k_{i\alpha}$	$I k_q k_{i\alpha} I_{History}$ limitation	q_c limitation
1	5	25	-	-	-
2	10	50	-	-	-
3	10	50	1	-	-
4	10	50	5	-	-
5	10	50	50	± 250	-
6	10	50	50	± 250	± 120

4.3 Pitching-up process simulation without integrating link

Firstly for Case1, the NVF and FM results are shown for comparison. The angular variables (angle of attack of the model and deflection angle of the rudder) refer to the left vertical axis, the angular velocity of the model refers to the right vertical axis in Figure 8. The legend ‘CFD’ represents the NVF results, while ‘SIM’ represents the FM simulation results by Matlab-Simulink. The CFD-based results have a lag behind those by FM simulation, because the unsteady effects are fully taken into account in the NVF results. The unsteady aerodynamic forces will cause delayed response, and the FM simulation does not consider the influence of this factor. The phenomenon also illustrates the importance of using NVF to design the control laws. There exist some differences between the steady-state angles of attack of NVF and FM, and both of them have a static error (more than 1°) with the expected target because of the inaccuracy of the aerodynamic model.

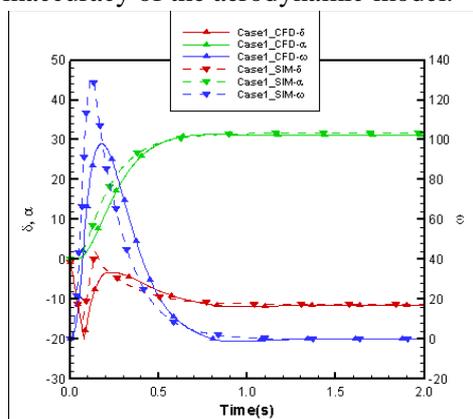


Figure 8. The NVF and FM simulation results of Case1

The static error is analyzed briefly as follows. Substituting Equation (7) into Equation (11), at the ideal trimmed state, namely the angle of attack is 30° and the pitching angular velocity is 0% , the steady-state rudder deflection angle from the control law is:

$$\delta_\infty = \frac{-QSc\tilde{C}_m(\alpha) + Ik_q[k_\alpha(\alpha_c - \alpha) - q]}{QScC_{m\delta}(\alpha)} = \frac{-\tilde{C}_m(30^\circ)}{C_{m\delta}(30^\circ)} \quad (13)$$

However, according to Table 1, $\tilde{C}_m(30^\circ) = -18.138$ and $C_{m\delta}(30^\circ) = -1.4763$, so the actual rudder deflection angle at this state is -12.286° , which is larger than the actual trimmed rudder deflection angle -10.284° . And this will lead to a larger steady-state angle of attack. For the steady state of Case1, the Equation (13) actually is:

$$\delta_\infty = \frac{-QSc\tilde{C}_m(30^\circ + e_{ss}) - Ik_q k_\alpha e_{ss}}{QScC_{m\delta}(30^\circ + e_{ss})} \approx \frac{-QSc\tilde{C}_m(30^\circ) - Ik_q k_\alpha e_{ss}}{QScC_{m\delta}(30^\circ)} \quad (14)$$

Since the static error $e_{ss} > 0$, the rudder deflection angle will increase with the control gain factors, and finally results in the reduction of the static error. Thus, we try to double the two control gain factors for comparison (Case2). The increment of the control gain factors can also accelerate the response, but too large gain factors may destabilize the whole system. For Case2, the FM simulation results have already tended to be divergent (the dash lines as shown in Figure 9), but the NVF results in Figure 9 are still able to maintain convergence due to the consideration of unsteady effects (namely the damping is larger). However, if we want to further reduce the static error or accelerate the response time, the way by increasing the control gain factors may fail, which means resulting in system divergence.

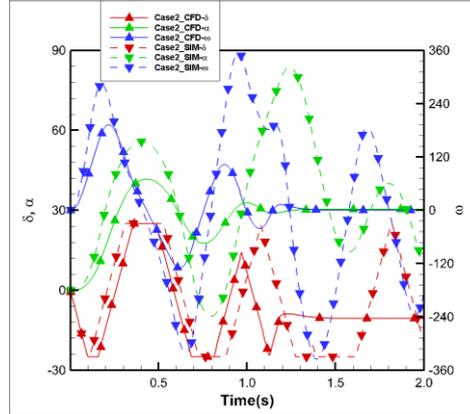


Figure 9. The NVF and FM simulation results of Case2

4.4 Maneuvering process simulation with integrating link

Inspired by equation (14), we can add an integrating link into the outer loop to further eliminate the static error. The steady-state rudder deflection angle after adopting the integrating link is:

$$\delta_\infty = \frac{-QSc\tilde{C}_m(30^\circ) - Ik_q k_{ia} I_{History}}{QScC_{m\delta}(30^\circ)} \quad (15)$$

where $I_{History} = \int (\alpha_c - \alpha) dt$ is the historical integral amplitude. When it reaches the ideal trimmed state, $\alpha - \alpha_c = 0$, $I_{History}$ will not change, and it can completely eliminate the static error, namely $\delta_\infty = \delta_c$.

We have tried two cases, in which the gain factor k_{ia} is set to 1 and 5, respectively (Case3 and Case4 in Table 2). The results compared with Case2 are shown in Figure 10. By introducing the integrating link, the response time of Case3 and Case4 are slightly reduced, but the overshoot phenomenon is more deteriorate. The approaching rate to the steady-state value is slightly faster with increasing k_{ia} (Figure 10(b)). However, the overshoot is obviously greater. This phenomenon can be analyzed from the property of the integrating link: when $\alpha < \alpha_c$, $I_{history}$ is negative and will make the rudder deflect larger, which will finally result in the missile model pitching-up stronger; when α

keeps growing up larger than α_c , $I_{history}$ will still negative in a certain range, so that the missile will continue to enhance its pitching-up trend. The analysis is similar for the pitching-down process. This is equivalent to increasing the ‘inertia’ of the system, and the damping is reduced relatively. Therefore, the response rate will become faster and the overshoot will increase as well.

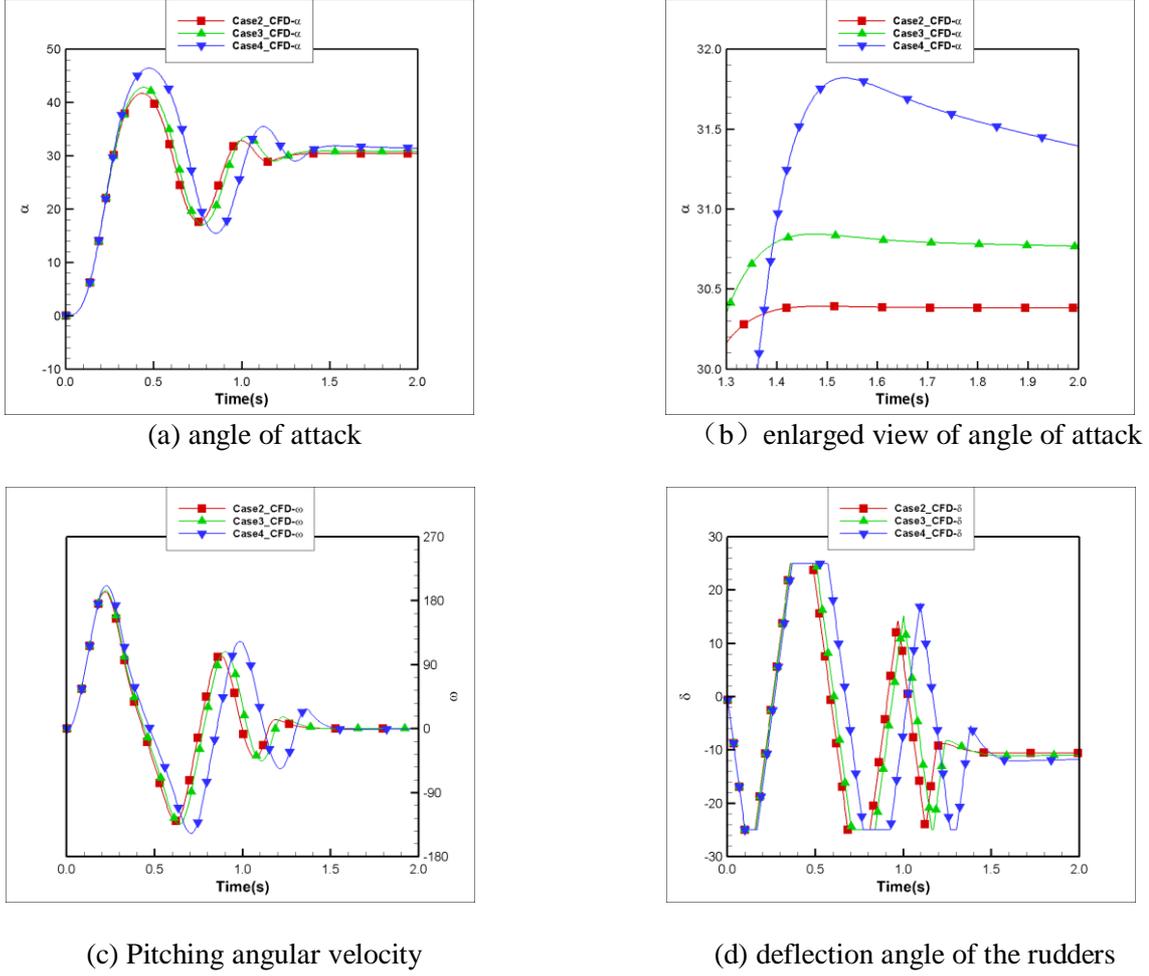


Figure 10. Comparison of Case2, Case3 and Case4

Because of the limited computing resources, we only simulate the pitching process within 2 seconds. When it reaches 2s, the response including the integrating link is even worse than the ones without. If the simulation keeps running forward, the final response including the integrating link can converge to the expected value, but it will take lots of time (more than 10s). If we want to approach the steady-state value faster, the gain factor $k_{i\alpha}$ should be larger. But this may cause excessive overshoot and eventually make the entire system out of control. In order to continue increasing $k_{i\alpha}$ the contribution of the integrating link $I k_q k_{i\alpha} I_{history}$ should be restricted. According to Equation (15) and the expected trimmed angle of attack, we can get $I k_q k_{i\alpha} I_{history} = 217.025$. So the limitation is set as $|I k_q k_{i\alpha} I_{history}| < 250$ and $k_{i\alpha}$ is taken as 50 (Case5 in Table 2). The response is shown in Figure 11. Comparing with Case3, although the integrating gain factor is increased by 50 times, the overshoot is almost unchanged, and the approaching rate to the steady-state value is increased slightly, so the performance of the controller is improved in some degree.

As shown in Figure 11, the overshoot phenomenon is not improved obviously. According to the maximum angular velocity in Case1 simulation process, we introduce a restriction condition to the rudder angular velocity (Case6 in Table 2) based on Case5. The maximum angular velocity in Case1 is $\omega_{max} = 98.137^\circ/s$, so we choose the limitation of $|q_c| < 120$. The CFD-based NVF results are plotted in Figure 12 and compared with those of Case1. The overshoot phenomenon has almost disappeared. Comparing with the baseline Case1, the static error is eliminated, the response time is shortened

substantially, so the control performance is improved obviously.

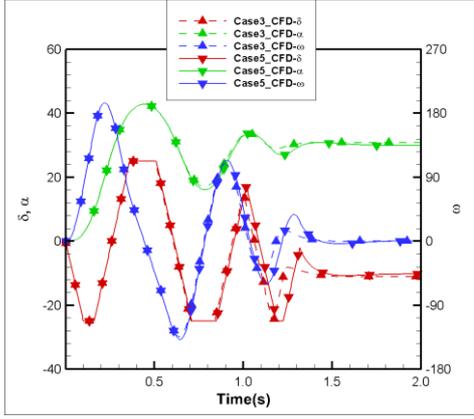


Figure 11. Comparison of the results between Case3 and Case5

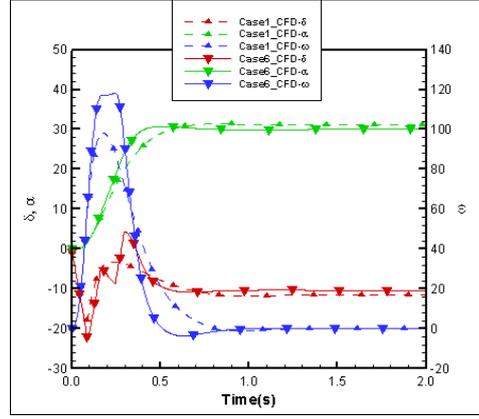


Figure 12. Comparison of the results between Case1 and Case6

Finally in summary, we list the response performance in Table 3 of all the six cases. The steady state angle of attack α_∞ , the steady-state rudder deflection angle δ_∞ , and the dynamic parameters (such as the rising time t_r , the setting time t_s and the overshoot σ_p) are compared. The rising time is defined as the response time when the model reaches the steady-state value for the first time. The setting time t_s is defined as the response time when the status of the model no longer exceeds a certain error band. Here we define the error band as $\pm 0.1^\circ$. The overshoot is defined as the ratio between the exceeding response and the steady-state value. It's obvious that the static error is significantly reduced and the rising time is greatly shortened with the increasing of the control gain factors. However, the overshoot is too large if we do not restrict the rudder angular velocity q_c , so the setting time increases finally. But in Case6, the overshoot is controlled very well after introducing a restriction condition of the rudder angular velocity, which is reasonable for realistic control device.

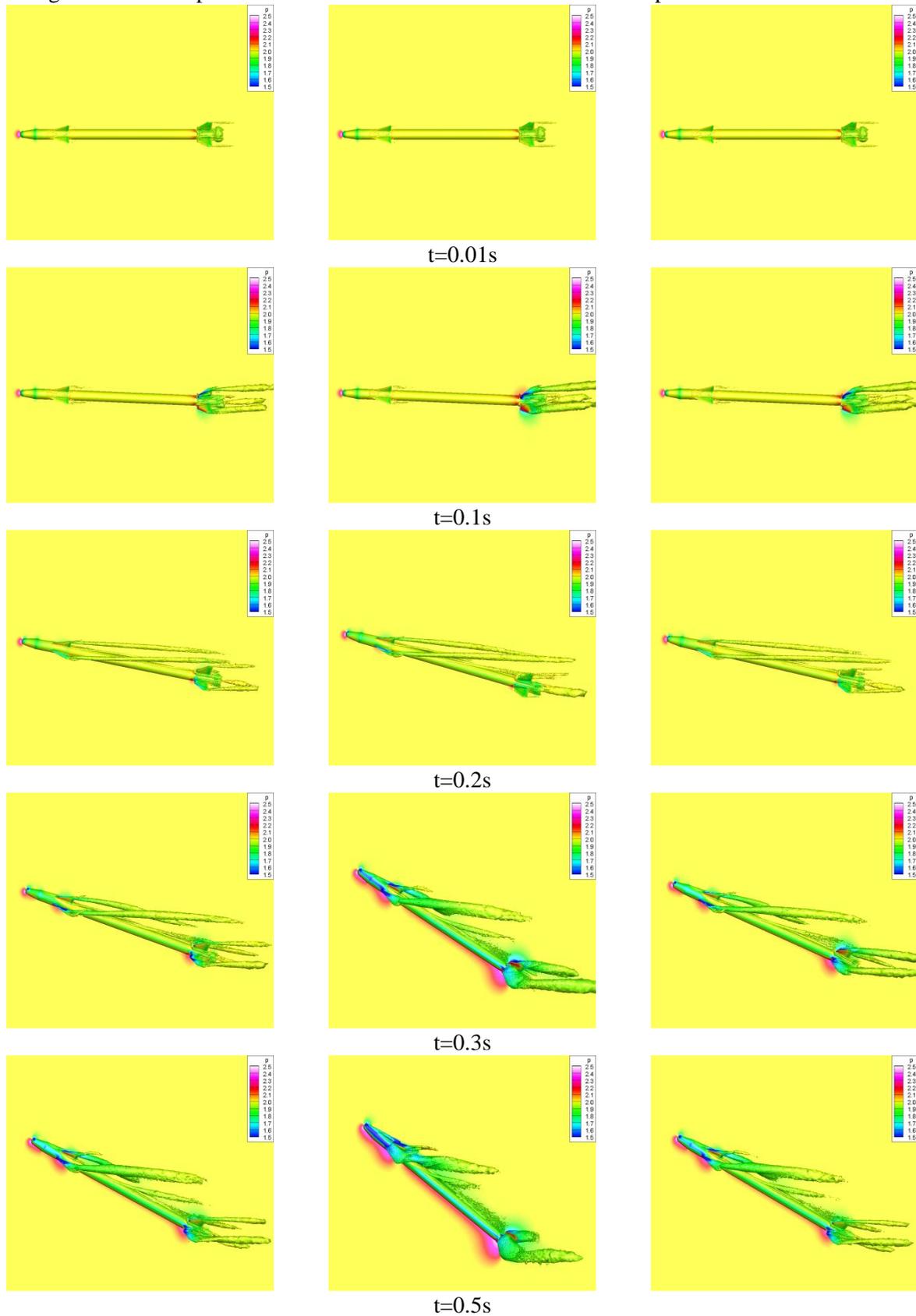
Table 3. Comparison of response parameters between Case1 and Case2

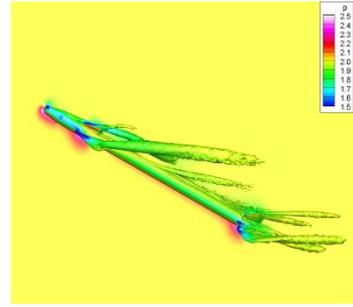
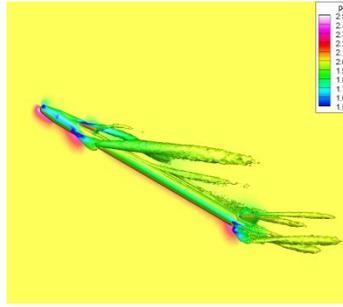
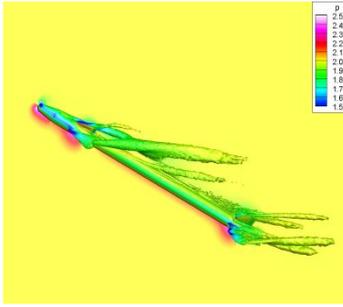
	α_∞ ($^\circ$)	δ_∞ ($^\circ$)	t_r (s)	t_s (s)	σ_p (%)
Case1-CFD	31.1608	-11.5552	0.6985	0.9715	0.7079
Case2-CFD	30.3823	-10.5750	0.2769	1.3374	38.1878
Case3-CFD	30.0000	-10.2844	0.2730	>2	42.7023
Case4-CFD	30.0000	-10.2844	0.2694	>2	54.7513
Case5-CFD	30.0000	-10.2844	0.2730	1.6810	42.7023
Case6-CFD	30.0000	-10.2844	0.4090	0.7704	2.5753

4.5 Comparison of the flow structure of Case1, Case2 and Case6

Since the pitching movements of Cases 2 – 5 are very similar, so we just compared the flow structures of Case1, Case2 and Case6. Figure 13 shows the Q contours (colored by pressure) over the model at six typical times for Case1, Case2 and Case6. It can be seen that the flow separation from the leading edge of the forebody delta wings and the rudders with angle of attack increasing. At lower angles of attack, the flow separation structures only occur at the tip of the rudders and within the model base (see the enlarged views in Figure 14). For Case2, because of the overshoot phenomenon, the maximum angle of attack is about 40° , so the flow separations are much more serious. At $t=0.5s$ for Case2, the flow over the forebody delta wings and the rudders is fully separated (see the enlarged views in Figure 14). Anyway, when approaching the final trimmed state (for example $t=2.0s$), the flow structures of Case1, Case2 and Case6 are almost same with each other.

Finally, Figure 15 shows the spatial streamlines over the model at different times with different angles of attach during pitching procedure. We also can see clearly the flow separation patterns with angle of attack increasing. It is difficult for conventional 2nd numerical methods to capture accurately the large-scale flow separation. We will validate the results with later experiments in the future.





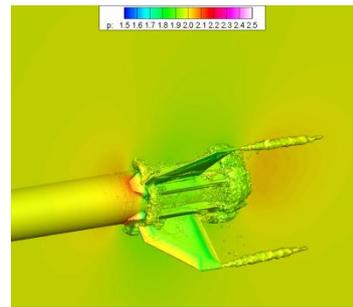
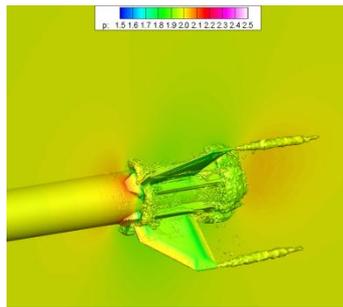
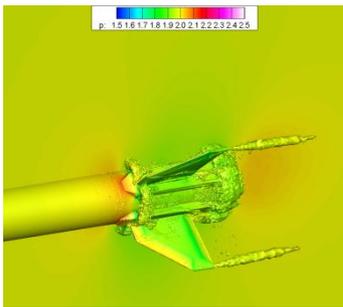
t=2.0s

Left: Case1

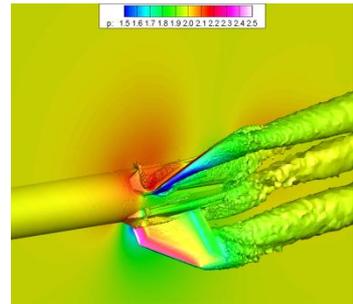
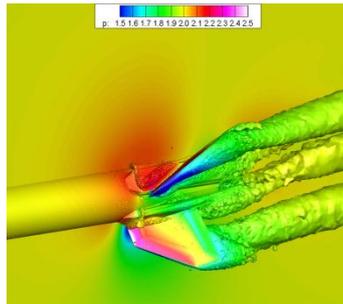
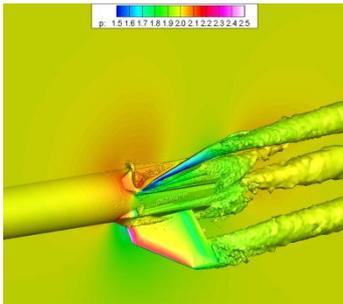
Middle: Case2

Right: Case6

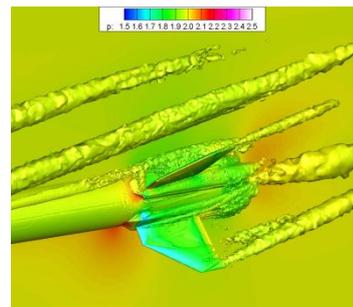
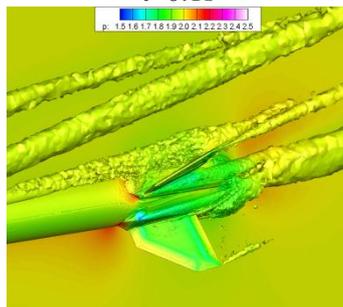
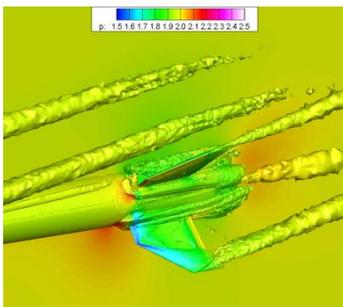
Figure 13. Q contours (colored by pressure) over the model for different cases.



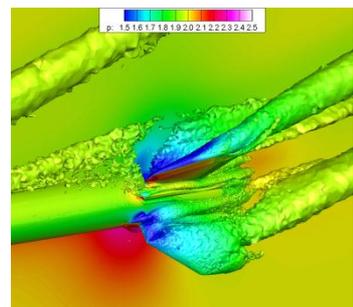
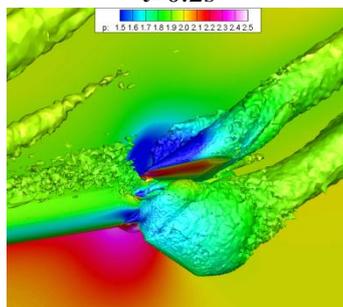
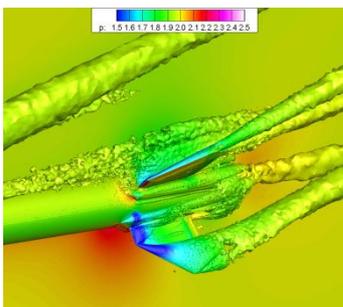
t=0.01s



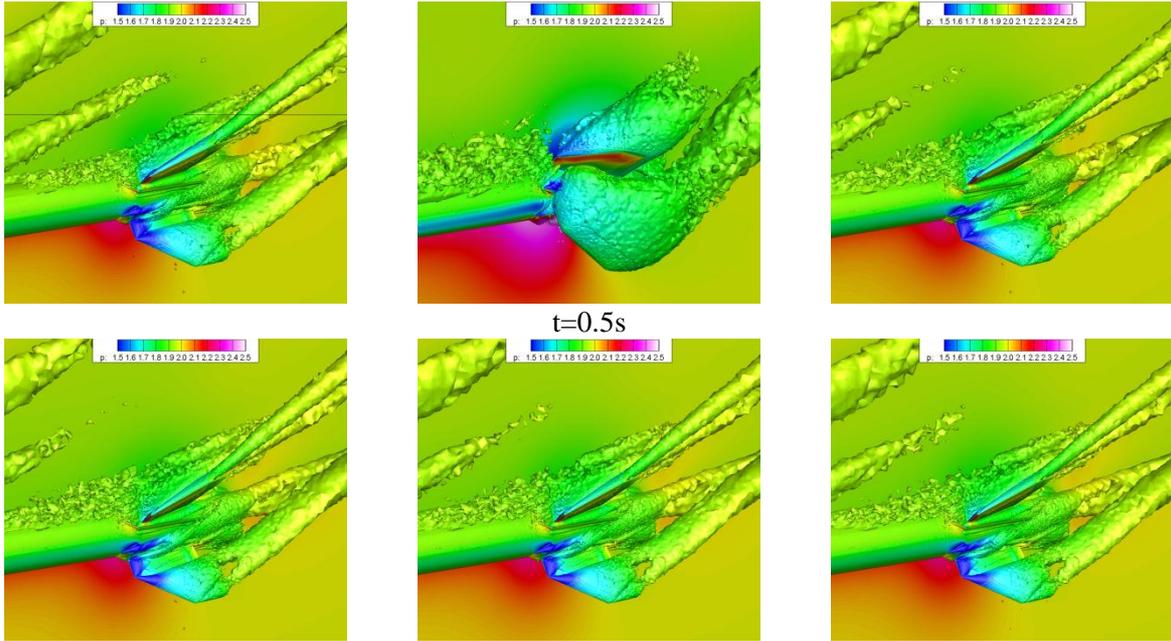
t=0.1s



t=0.2s



t=0.3s



Left: Case1
 Middle: Case2
 Right: Case6
 Figure 14. Enlarged views of Q contours (colored by pressure) near the rudders for different cases.

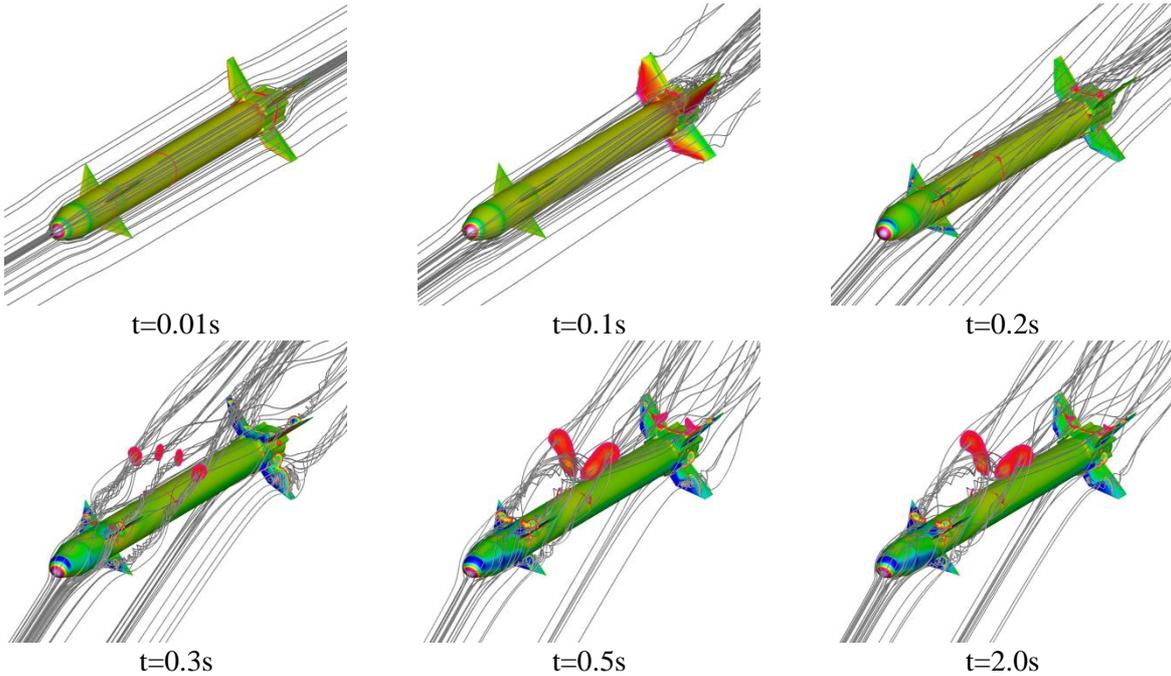


Figure 15. The spatial streamlines for Case6 at different times

5 Concluding remarks

A multi-physics coupled platform is presented for numerical virtual flight simulations, which integrates the high-efficient parallel unsteady RANS solver, dynamic hybrid grid generator, 6DOF flight mechanics solver, and flight control law. The nonlinear dynamic inversion method is adopted to design the flight control law of a super maneuverable missile model in the pitching channel. Through the simulation and analysis of the maneuvering process, the baseline dynamic inversion control law is improved, and a set of optimized control gain factors are obtained.

As illustrated by the test cases, for the rapid maneuvering of an aero-vehicle at large angles of

attack, flight mechanics simulations may often result in large errors due to the inaccurate approximation of aerodynamic model. The CFD-based NVF simulation is able to provide more reliable closed-loop response characteristics of the control system, because it has taken into account the strong unsteady effect of the controlled plant. More importantly, we can realize integrated multi-discipline optimization in the future, using the NVF tools.

Acknowledgements

This work is supported partially by National Key Research and Development Program (No.2016YFB0200700) and National Natural Science Foundation of China (Grant No. 11532016 and No. 11672324).

References

- [1] Ratliff C L, Marquart E J. Bridging the gap between ground and flight test: Virtual flight testing (VFT). AIAA paper 1995-3875, 1995.
- [2] Lawrence F C and Mills B H. Status update of the AEDC wind tunnel virtual flight testing development program. AIAA paper 2002-0168, 2002.
- [3] Murri D G, Nguyen L T, Grafton S B. Wind tunnel free-flight investigation of a model of a forward swept-wing fighter configuration. NASA TP-2230, 1984.
- [4] Salas M D. Digital Flight: The last CFD aeronautical grand challenge. *Journal of Scientific Computing*, 2006, 28(2/3): 479-505.
- [5] Boelens O J, Prananta B B, Soemarwoto B I, Allen M R, Badcock K J, Fritz W. Towards an aero-servo-elastic simulation capability for high-performance fighter aircraft. RTO-MP-AVT-2005.123(30).
- [6] Schütte A, Einarsson G, Raichle A, Schöning B, Orlt M, Neumann J, Arnold J, Mönnich W, Forkert T. Numerical simulation of maneuvering aircraft by aerodynamic, flight-mechanics, and structural-mechanics coupling. *Journal of Aircraft*, 2009, 46(1): 53-64.
- [7] Kroll N, Rossow C C. Digital-X: DLR's way towards the virtual aircraft. NIA CFD Conference, August 6-8, 2012, Hampton Virginia.
- [8] Kroll N, Abu-Zurayk M, Dimitrov D, Franz T, Führer T, et al. DLR project Digital-X: towards virtual aircraft design and flight testing based on high-fidelity methods. *CEAS Aeronautical Journal*, 2016, 7(1): 3-27.
- [9] Post D E, et al. A new DoD initiative: the computational research and engineering acquisition tools and environments (CREATE) program. *Journal of Physics: Conference Series* 125, 2008, 012090: 1-11.
- [10] Post D E, Atwood C, Newmeyer K, Kandsberg S, Shull F. The computational research and engineering acquisition tools and environments (CREATE) program. *IEEE Computing in Science & Engineering*, 2016: 10-12.
- [11] Morton S A, McDaniel D R, Sears D R, Tillman B, Tuckey T R. Kestrel V2.0-6DoF and control surface additions to a CREATE simulation tool. AIAA paper 2010-511, 2010.
- [12] Mortan S A, Meakin R L. HPCMP CREAET-AV Kestral architecture, capabilities, and long term plan for fixed-wing aircraft simulations. AIAA paper 2016-0565, 2016.
- [13] Narducci R. Industry assessment of HPCMP CREATE-AV Helios. AIAA paper 2015-0553, 2015.
- [14] Wong T C. Application of CREATE-AV Helios in an engineering environment: Hover prediction assessment. AIAA paper 2017-1667, 2017.
- [15] Zhang L P, Wang Z J. A block LU-SGS implicit dual time-stepping algorithm for hybrid dynamic meshes, *Computers & Fluids*, 2004, 33: 891-916.
- [16] Zhang L P, Chang X H, Duan X P, Zhang H X. A block LU-SGS implicit dual time-stepping algorithm on

- hybrid dynamic meshes for bio-fluid simulations. *Computers & Fluids*. 2009,38:290-308.
- [17] Zhang L P, Chang X H, Duan X P, Zhao Z, He X. Applications of dynamic hybrid grid method for three-dimensional moving/deforming boundary problems, *Computers & Fluids*, 2012, 62: 45-63.
- [18] Chang X H, Ma R, Zhang L P, He X, Li M. Further study on the geometric conservation law for finite volume method on dynamic unstructured mesh, *Computers & Fluids*, 2015, 120(5): 98-110.
- [19] Snell S A, Enns D F, Garrard Jr. W L. Nonlinear inversion flight control for a supermaneuverable aircraft. *Journal of Guidance, Control, and Dynamics*, 1992, 15(4): 976-984.
- [20] Bellman R E. *Adaptive control processes: A guided tour*. Princeton University Press, Dec., 2015.
- [21] Chen Q, Guo Y Y, XIE Y F, Chen J Q, Yuan X X. Research and application of coupled simulation techniques of PID controller and CFD. *Acta Aeronautica et Astronautica Sinica*, 2016, 37(8): 2507-2516.
- [22] Chang X H, Ma R, Wang N H and Zhang L P. Parallel implicit hole-cutting method for unstructured Chimera grid. The 10th International Conference on Computational Fluid Dynamics. Barcelona, Spain, July 9-13, 2018. ICCFD10-0051.
- [23] Kamakoti R, Shyy W. Fluid-structure interaction for aeroelastic applications. *Progress in Aerospace Sciences*, 2004, 40: 535-558.
- [24] Chang X H, Ma R, Zhang L P, He X. Fully implicit approach for flow/kinetic coupled problems based on dynamic hybrid mesh. The Eighth International Conference on CFD (ICCFD8), Chengdu, July, 2014, ICCFD8-2014-0329.