Master of Science Thesis

Development of a data fusion framework for the aerodynamic analysis of launchers

Q.Zhang

January 24, 2017

Aerodynamics

Faculty of Aerospace Engineering

TU Delft

Delft University of Technology

Development of a data fusion framework for the aerodynamic analysis of launchers

Master of Science Thesis

For obtaining the degree of Master of Science in Aerospace Engineering
at Delft University of Technology

Q.Zhang

January 24, 2017

Faculty of Aerospace Engineering · Delft University of Technology

Delft University of Technology

DELFT UNIVERSITY OF TECHNOLOGY
DEPARTMENT OF AERODYNAMICS

The undersigned hereby certify that they have read and recommend to the Faculty of Aerospace Engineering for acceptance the thesis entitled **"Development of a data fusion framework for the aerodynamic analysis of launchers"** by **Q.Zhang** in fulfillment of the requirements for the degree of **Master of Science**.

Dated: January 24, 2017

Supervisors:

Dr.ir. F.F.J. Schrijer

Dr.ir. B.W. van Oudheusden

Ir. B.T.C. Zandbergen

Preface

This thesis concludes my Flight Performance and Propulsion master program in the faculty of Aerospace Engineering of Delft University of Technology. The time spent on this thesis is longer than expected and the reason is multiple, but I really learnt a lot. If I kept working as an engineer as two years ago, I will never feel the bitter and sweet of the master study.

I would like to show my sincere gratitude to my supervisor Dr. ir. Ferry Schrijer and Dr. ir. Bas van Oudheusden for providing me this experimental topic, thanks for their kindness when they introduced me to this project. I'm grateful to Dr. ir. Ferry Schrijer for his great guidance through the project, and for the understanding and encouragement he gave me. Special thanks to Dr. ir. Sander van Zuijlen, who provided me great support on the application of CFD software. Thanks Ir. B.T.C. Zandbergen for his help at the beginning of my thesis and all the useful literature provided.

I would like to thank my family, especially my dear wife Fanyi Yang, for always having faith in me and supporting my foreign study from the beginning to the end. Without her love, I will never finish the master study. I also want to show my gratitude to my dear friends who have given me a lot of help during this thesis. They are Li Mo, Ye Zhang, Li Cui, William Mulkens, Qingqing Ye, Rucheet Tipnis, Zaoxu Zhu, Qingxi Li, Remco van de Meerendonk, Lin Liu, Lijian Qi, Shuanghou Deng, etc.

Quan Zhang
Delft, The Netherlands
24^{th} January, 2017

Abstract

During the launching vehicles design and their optimization process, the implementing of aerodynamic characteristics (AC) database is quite important. Accuracy of such database determines quality of the pre-flight simulation, trajectory and even the whole design direction. However, the database building process is one of the most challenging steps. For most launchers, the flight envelopes cover a wide range of Mach number and Reynolds number. Traditionally, two basic practices are mostly adopted to carry out this kind of job, the experimental-based wind tunnel test (WTT) and the computational fluid dynamics (CFD) simulation. However, both methods have their own weaknesses, either being economically or computationally expensive. These disadvantages disenable them to be the best routines. There is another kind of technique to achieve the AC database, the industrial semi-empirical computer programs, such as DATCOM from U.S. Air Force and Aeroprediction from U.S. Naval Surface Warfare Center. For scenarios where WTT is not applicable, the computational intensity of CFD can be reduced by the combination of CFD and industrial semi-empirical methods. In this project, such approach is proposed to develop the aerodynamic database for launchers, in an efficient and accurate way.

The basic idea is to apply the Co-Kriging-based data fusion algorithm on CFD data and estimation results. Generally, the results from semi-empirical methods tend to have low fidelity. By performing the data fusion, the accuracy of low fidelity estimation data (from DATCOM) can be improved significantly by adding a small sum of expensive data (from CFD). During the fusion process, one important step is the sampling process, namely determining the simulation points among all the flight regime where CFD simulations should be performed.

Before adopting the CFD simulation results as the accurate and expensive benchmark, validation of the CFD codes should be carried out in advance. The evaluation is performed on a U.S. launcher, of which some WTT data is available from public literature. The CFD codes are based on the Navier-Stocks governing equations and are dealt with the Reynolds-Average Navier-Stocks (RANS) concept. In this project, the one-equation turbulence model, Sparlart-Allmaras Model, is adopted for its robustness in the chosen CFD codes. After the grid-independence study, the simulation results prove satisfactory, with an error lower than 10% for the normal force coefficient (C_N) and axial force coefficient (C_A). For the pitching-up moment coefficient (C_M), the error can be as large as 24%. The reason for it is complicated, and the most possible reason is the existence of the double backward-facing steps at the end of the payload stage, flow around which is difficult for the CFD code to estimate and simulate.

After the evaluation, the data fusion is implemented to the VEGA launcher. The first step is to establish the low-fidelity AC database from DATCOM. Three kinds of information are written into the input files, the geometry of the launcher, the definition of the flight conditions, the desired outputs. The second step is performing CFD simulations on flight scenarios which are chosen based on the low-fidelity AC database and the Latin Hypercube Sampling (LHS) scheme. The final step, called the fusion process, is carried out based on the Co-Kriging algorithm. The fusion results are evaluated by comparing them with corresponding new CFD simulations. The evaluation demonstrates that the proposed method is capable of improving the low-fidelity AC database to a great extent, with only a small quantity of computationally expensive CFD simulations. In this project, the error is reduced from as large as 60% to lower than 15%. Should more expensive data is added, the accuracy can be further improved.

This project suggests an efficient way of establishing a high-fidelity AC database by harnessing Co-Kriging algorithm. Data from industrial semi-empirical methods is improved to high accuracy with a small quantity of CFD simulations. The suggested scheme has a big advantage in saving time. In this project, 90% CFD computation time is saved to achieve the same accurate AC database.

Table of Contents

Nomenclature

Abbreviations

AC Aerodynamic Characteristics
ANOVA analysis of variance
AoA Angle of attack
APVCT Allen and Perkins Viscous Crossflow Theory
AST Adaptive Space Transformation
CCD Central Composite Design
CFD Computational Fluid Dynamics
CFL Courant number
Corr Correlation
DACE Design and Analysis of Computer Experiments
DATCOM Data Compendium
DES Detached Eddy Simulation
DNS Direct Numerical Simulation
DOE Design of Experiment
EIF Expected Improvement Function
GECK Gradient-Enhanced recursive CoKriging
HTVD Van Dyke Hybrid Theory
LES Large Eddy Simulation
LHS Latin Hypercube Sampling
MLE Maximum Likelihood Estimation
MNT Modified Newtonian Theory
MSE Mean Squared Error
NS Navier-Stocks
OFAT one-factor-at-a-time
RANS Reynolds-Averaged Navier-Stocks
RSM Response Surface Methodology
SOSE Second Oder Shock Expansion
WTT Wind Tunnel Test

Coefficients

C_A	Axial force coefficient	[-]
C_f	Friction coefficient	[-]
C_M	Pitching up moment coefficient	[-]
C_N	Normal force coefficient	[-]

Greek Symbols

ϵ	random error	[-]
μ	Dynamic viscosity	$[kg/m^3]$
ρ	Density	$[kg/m^3]$
σ^2	variance of the normally distribution	[-]

Latin Symbols

τ	viscosity	[kg/(sm)]
F_A	Axial force	[N]
F_N	Normal force	[N]
s^2	mean square error	[-]
u_τ	friction velocity	$[m^2]$
y^+	non-dimensional wall distance	[-]
F	Thrust	[N]
G	Gravity force	[N]
M	pitching up moment	[Nm]
Ma	Mach number	[-]
Re	Reynolds number	[-]
S	reference area	$[m^2]$

Superscripts

$*$	new evaluation point

Subscripts

c	computationally cheap data
d	difference between cheap and expensive data
e	computationally expensive data

Chapter 1

Introduction

The research for this project is about the development of aerodynamic framework of the launchers. This chapter serves as a general introduction to the background of this project, which includes the introduction to launchers, aerodynamic characteristics(AC) of launchers, the importance of AC of launchers, AC database building methodology. Also, the main objective and research approach of this project are generally explained. Finally, the structure of this thesis is summarized.

1.1 Launchers

In spaceflight, a launching vehicle or a launcher is a rocket that sends certain payload from earth surface to space. Space can be sub-orbital space such as that of sounding rockets (sometimes called a research rocket), orbital space such as that of the most usual man-made satellites, and also orbits that lead to escape the gravitational attraction of the earth.

Launchers are usually classified by the number of their stages. For instance, earth orbital launchers typically have at least two stages, and some of them can have as many as four stages or even more. Besides various stages, some boosters can be attached around the first stage, which supply high early thrust and reduce the burnout mass of later stages to make a larger payload. For various purposes, there are many types of launchers, examples of which are illustrated in Figure 1.1. Sometimes, modifications were performed on certain kind of classical launchers, based on various performance and missions, famous of which are Delta family, Titan family, Ariane family, etc. Figure 1.2 gives an example of the Atlas V family. During the modification process, a big part of the design can be based on already developed parts, subsystems, components and materials, which can save a great deal of resource.

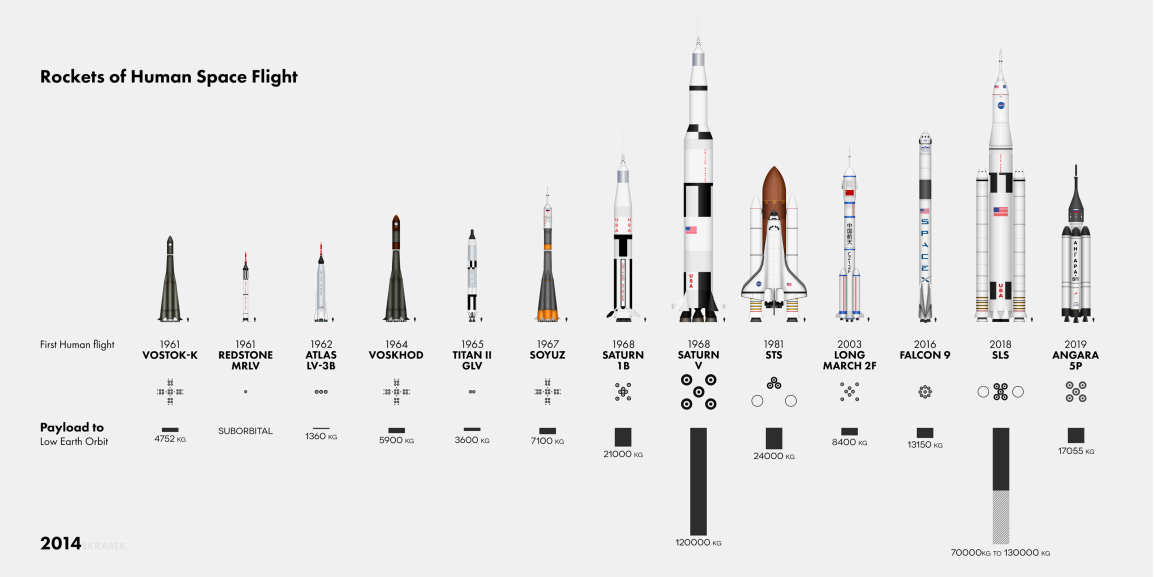


Figure 1.1: Schematic sketches of rocket examples of human space flight (Tyler Skrabek (2014))

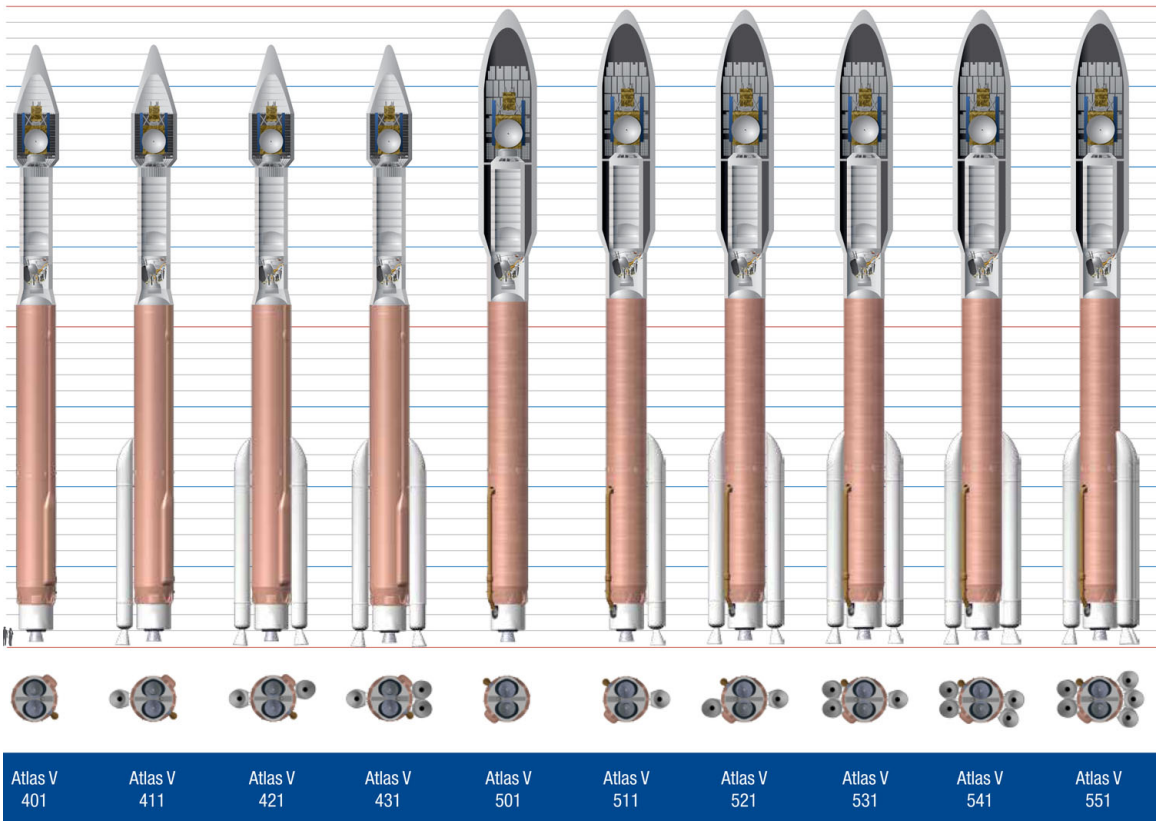


Figure 1.2: Simplified geometric configuration of Atlas V family (ULA (2016))

1.2 Flight conditon and launcher dynamics

During the launch process, the flight conditions of the launcher is much more complicated than that of traditional airborne flight vehicle (airplane), it changing significantly in a short time. As mentioned by Nicolì et al. (2006), for the VEGA (a Europe small commercial laucher), it can reach the velocity of 1800 m/s and an altitude of 46 km in just 120 seconds. During this period, the Mach number (Ma) and Reynolds number (Re) change from 0.5 and $1 * 10^6$ to 6 and $4.3 * 10^7$, respectively. For another example, the typical launch profile of the Ariane 4 is shown in Figure 1.3.

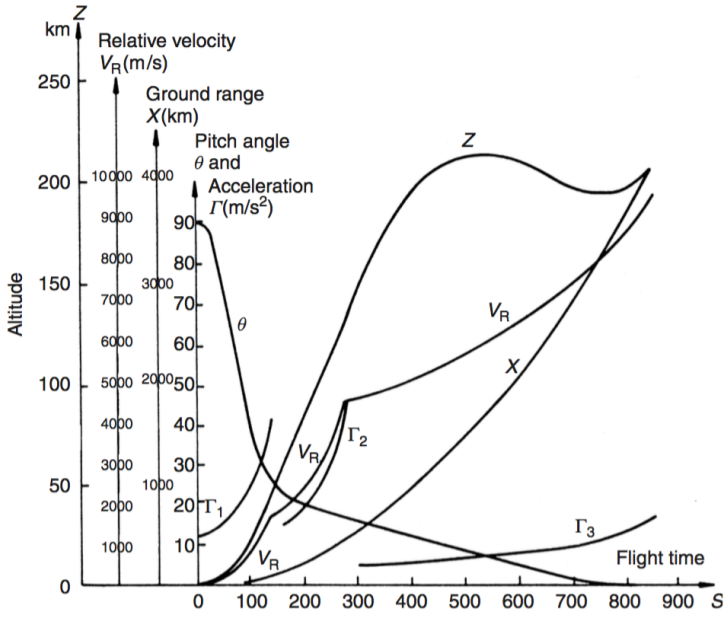


Figure 1.3: Typical Ariane 4 launch profile (Fortescue et al. (2011))

This figure mainly gives information on the variation of height conditions as a function of altitude. In real flight, the flight conditions are much more complex, for example, there is wind which adds a cross-flow that the launcher experiences during lift-off. The strength of the ground wind changes from time to time, introducing challenges w.r.t. stability control. Besides those, the exhaust plume at the launcher base can affect the flow around the launcher. During the flight, separation process between various stages and attached boosters can also bring in shock wave and consequent flow separation. Sometimes, the effects mentioned above can interact with each other, which is extremely difficult to simulate all at once by modern CFD tools.

In terms of the launcher dynamics, motion equations of the vertical plane, which is parallel to the flight direction and passes the centre of mass, is illustrated in Figure 1.4. In this project, we only consider the dynamics in 2D plane (axial force, normal force and pitching moment),

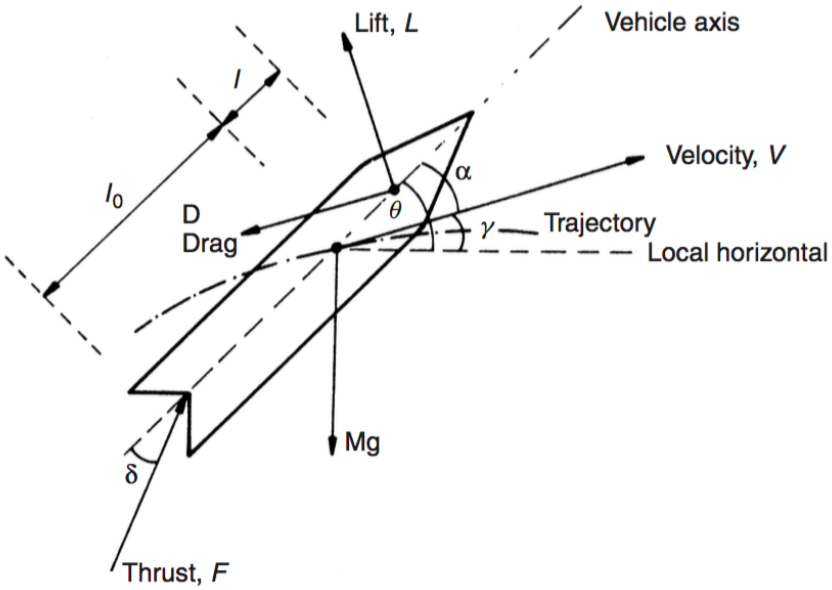


Figure 1.4: Configuration and nomenclature for rocket motion in the vertical plane (Fortescue et al. (2011))

ignoring the more complex 3D scenarios (yaw, roll, etc.). In the direction of flight, we have,

$$M\frac{dV}{dt} = Fcos(\alpha+\delta) - Mgsin\gamma - D, \tag{1.1}$$

and normal to the flight direction (inside the vertical plane),

$$MV\frac{d\gamma}{dt} = Fsin(\alpha+\delta) - Mgcos\gamma + L + \frac{MV^2}{r}cos\gamma. \tag{1.2}$$

Finally, the accompanying angular motion is dominated by,

$$I_p\frac{\mathrm{d}^2\theta}{\mathrm{d}x^2} = (Lcos\alpha + Dsin\alpha)l - Fl_0sin\delta. \tag{1.3}$$

1.3 Importance of aerodynamics

As introduced in 1.2, the motion of launchers is determined by forces act on the vehicles, the normal force F_N, the axial force F_A, the thrust F and the gravity G. Among them, the normal and axial force are extremely important, for they are not directly controllable like thrust force and they change quickly with flight time, altitude and AoA. it is of great importance to get a clear view of the F_N and F_A before the launching mission. To evaluate the effect of these

forces, the normal force coefficient C_N, axial force coefficient C_A and pitching up moment coefficient C_M is introduced:

$$\begin{aligned} C_N &= \frac{F_N}{\frac{1}{2}\rho_\infty V^2}, \\ C_A &= \frac{F_A}{\frac{1}{2}\rho_\infty V^2}, \\ C_M &= \frac{M}{\frac{1}{2}\rho_\infty V^2 S}, \end{aligned} \tag{1.4}$$

where F and M are force and moment exerted on the launcher, and $\frac{1}{2}\rho_\infty V^2$ and S are the dynamic pressure and reference area of the launcher, respectively.

The combination of the three coefficients mentioned above as a function of Ma, Re and AoA forms the launcher AC database. Such database is essential for the design and/or optimization of launchers. It determines the launchers' controllability, stability and trajectory. The AC under various flight condition directly determines corresponding control laws, size and quantity of control engines and so on. Furthermore, the aerodynamic force and moment have a great effect on the launchers structure. They may cause structure vibrations and strength fatigue, both of which may cause structural failure and subsequently launch failure.

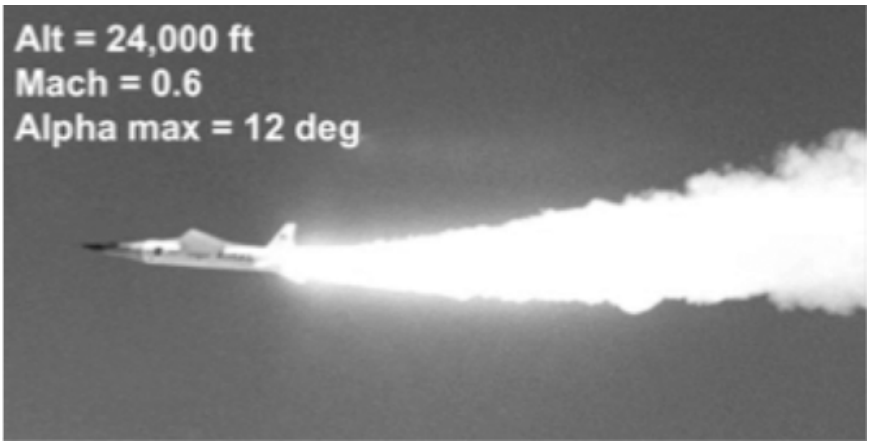


Figure 1.5: Pegasus XL with X-42 in flight Hall et al. (2012)

In the aerospace industry, an accurate AC database is extremely important. Without it, setbacks,delay, even project cancelation may happen. One example is the Pegasus XL launch system, see Figure 1.5. During its first flight to launch the NASA X-43 vehicle, to keep project cost under control, the wind tunnel test is compressed. During the AC database building process, one mistake was made. The rear fin was assumed symmetrical, which leads to higher fin effectiveness than really needed. Besides that, the thermal protection system was added after the wind tunnel test (WTT), adding more control authority. The demand and supply data are illustrated in Figure 1.6. During the real flight, the control system was

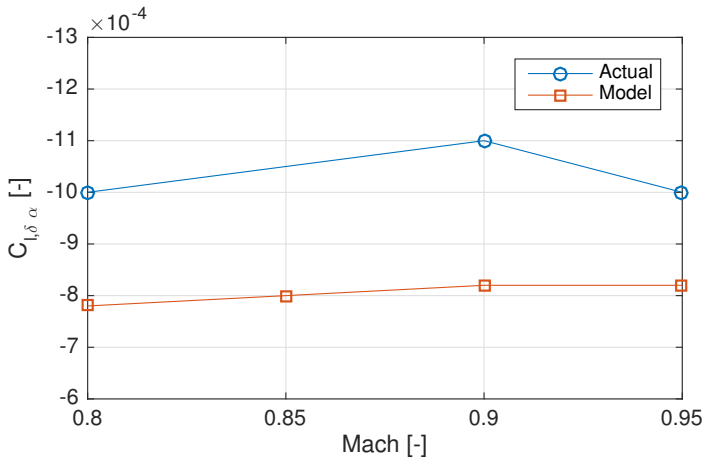


Figure 1.6: Modeling error in roll stability for Pegasus XL with X-43 Hall et al. (2012)

inadvertently over reacting and brought divergent roll oscillation, which directly caused failure of the fins and loss of the vehicle.

The second similar case deals with the Delta II heavy launcher, which has problems caused by unsteady flow during flight. In this case, Boeing had followed the standard industry practice to perform the WTT. The only thing they missed was taking into account the unsteady effect, which finally caused the main engine gimbaling more significantly than being expected. After a long time of investigation, Boeing and its partner Kennedy Spaceflight Centre found out that unsteady separation occurred under the solid booster attached to the launcher and corresponding steps were taken to solve this problem.

From those two cases, we realize that it is critical to establish an ***accurate*** AC database of the launcher to achieve a safe and successful mission. In real practice, every effort possible should be made to improve the fidelity of the AC database.

1.4 AC database building

In the early days, when the computational resource is not so rich as nowadays, the traditional WTT is the prime practice to achieve the indispensable AC data during the design process. Usually, before the real flight of aero-vehicles, a large number of wind tunnel experiments should be carried out, to get sufficient understand of the AC over the whole flight envelope. However, WTT has its limitation. Firstly, due to the limitation of laboratory equipment, the in-flight Re number can be too high for the traditional wind tunnel to realistically duplicate, as in the case mentioned before by Nicolì et al. (2006). Secondly, considering the experiment budget, the whole flight range may be too wide and it is not realistic to make all the experiments, for the huge finance consumption of the WTT. Also, the WTT is usually performed in relatively late period of the design process and before that, WTT can not provide full data.

An alternative to wind tunnel testing are the Computational Fluid Dynamics (CFD) simulations, which have seen dramatic developments over the past years. Advantages of CFD are apparent. It can be carried out even before the design process and can provide sufficient guide to the concept design process. Also, the geometry of test object can be changed easily. However, there are also some weaknesses in this technique. Generally, the accuracy of CFD is still not satisfactory when dealing with transonic, turbulent and separation flow. For these scenarios, without validation of WTT, one can hardly able to tell the accuracy of CFD results. Another important issue about CFD is that the simulation can be computationally intensive and time-consuming especially when the quantity of cell numbers is huge in the meshing file. As mentioned by Langone and Bermüdez (2009) and Kitamura et al. (2013), one high-precision computation may take from tens of hours to several days, based on the quantity of the grids and the flight Mach number.

Besides WTT and CFD, there is another kind of industrial method, which estimates AC by way of semi-empirical correlations. The codes are based on the combination of theoretical equations and experimental databases. The most popular aero estimating codes are United States Air Force Stability and Control Digital DATCOM (DATCOM) from U.S. Air Force and Aero Prediction (AP) from U.S. Naval Surface Warfare. Both codes have up-to-date versions, but not publicly available. For the DATCOM code, there is a branch, which is especially suitable for missile-shaped vehicle, called Missile DATCOM. One version of Missile DATCOM code can be found in Hammond (2001). The estimating codes generally produce the AC database by applying approximation to the exact motion equations (the Navier-Stokes equations). When that is not possible, for example at a high incidence angle, empirical data will be implemented to create an estimate. The advantage of such approximation codes is the high efficiency, usually taking a few seconds to carry out the results, Blake (1998). The shortcoming, however, is that the estimating accuracy is not quite high. Work performed by Sooy and Schmidt (2005), Blake (1997) and Castellini (2012) show that error of Missile DATCOM can be about 20% in terms of normal and axial force coefficients when comparing to the WTT data. Moore et al. (2000) also shows that the AeroPrediction 02/98 had an error as large as 35% in the subsonic or transonic regime. EngelenF.M. (2012) pointed out an obvious problem with Missile DATCOM(w.r.t the 97 version): for the transonic estimation, the code would give two drag peaks, which is not physically possible. This error may be caused by implementation when the code switches between physics based modelling and empirical corrolations.

As mentioned by Anderson Jr (2010), pure experiment, pure theory and CFD are the three equal partners in modern aerodynamics. To get a clearer view of the modern aerodynamics, one should try to combine more schemes rather than utilizing only one. As illustrated in Figure 1.7, the double arrows mean that in modern aerodynamics, these equal partners do not stand alone. These techniques can help each other to resolve the complicated aerodynamic problems. In this project, an attempt is made to combine the CFD simulation results and the estimation results. From the fusion of these two type (fidelity) of data, we can achieve the accurate enough AC database, meanwhile, reduce the quantity of CFD simulation. Improving efficiency and ensure accuracy as well.

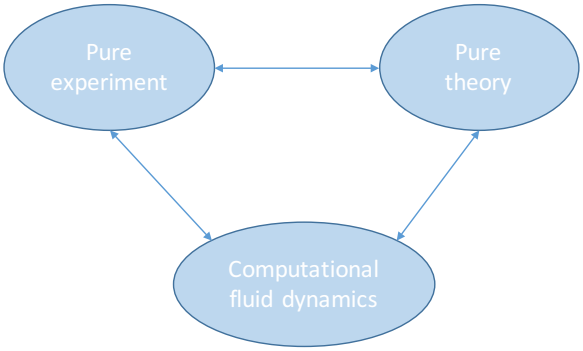
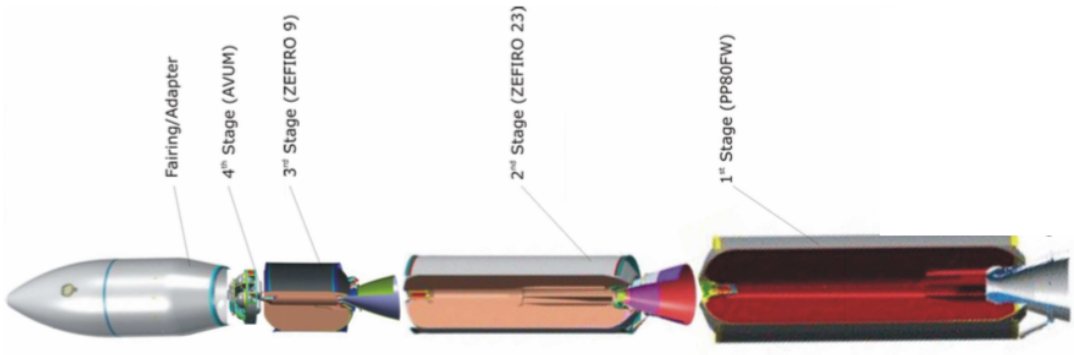


Figure 1.7: The three equal partners of modern aerodynamics

1.5 VEGA description

The researching object of this project is the VEGA launcher, which is a small launch vehicle stemming from the Vettore Europeo di Generazione Avanzata program from the early 1990s. It starts as a national Italian concept and serves as a complement to Ariane family, which is optimized for missions with large/medium geostationary-transfer-orbit and low-earth-orbit satellites. It takes advantage of expertise from Ariane by using its solid booster technology. VEGA has four tailed stages, focusing on a payload-lifting capability of 1500 kg to a 700 km circular polar orbit. These four stages are three solid propellant stages (first to third) and a restartable Attitude and Vernier Upper Module (AVUM). Besides those, there is an upper composite, including the payload stage faring, payload adapter and separation systems. The exterior and inner parts are illustrated in Figure 1.8.

(a) Exterior ESA (2016)



(b) Interior Perez (2006)

Figure 1.8: VEGA launcher illustration

1.6 Thesis objective

The main objective of this thesis is to develop a data framework for the aerodynamic analysis of launchers. In this project, the development will be performed on the VEGA launcher. To implement the target framework, a scheme which can make the correlation between low-fidelity and high-fidelity data sources should be applied. With such correlation, the global accuracy of the low-fidelity data should be improved significantly when a small sum of high-fidelity data is added. In order to achieve the objective, some sub-problems should be solved, which include:

1. Carry out the literature study to find the proper scheme that can correlate the low-fidelity and high fidelity data in an accurate and efficient way.
2. Validate the CFD code to be used in this project, which can be done by comparing WTT data of a launcher to that from the CFD simulations.
3. Build the low-fidelity AC database by executing the semi-empirical method (DATCOM) and perform corresponding CFD simulations. Carry out the fusion between DATCOM and CFD data with the formerly mentioned correlation scheme.
4. Evaluate the fusion outcome by comparing the data from fusion and that from corresponding CFD simulations on new flight conditions.

It is important to mention in advance that in this project, we only consider the *ideal* case, which means the ideal lift-up process is adopted, no consideration on jet/exhaust plume effect, corresponding separation and the cross wind/wind shear effect.

1.7 Thesis structure

This document is divided into 3 parts. The first part is composed of Chapter 1 to 3. In Chapter 2, the methods of implementing AC framework are reviewed, discussed and compared. Chapter 3 focuses on explaining the theories, which contains approximation scheme of Missile DATCOM, theoretical framework of NUMECA and data fusion technique of multi-fidelity database. In the second part, Chapter 4, validation of the CFD software, NUMECA, is performed. Simulation results will be compared with that from corresponding WTT, meanwhile, the accuracy of the CFD result will be checked and discussed. For part three, comprising Chapter 5 and Chapter 6, the final results of this project are presented and discussed. Details about DATCOM approximation, CFD simulation on the VEGA will be given. Fusion of two type of database is performed and analyzed. After that, the fusion result will be evaluated and discussed. Finally, in Chapter 6, the main conclusions of this project are given and recommendations for further investigations are summarized.

Chapter 2

Methods of implementing AC database

To implement an AC database, there are various kinds of schemes. In this chapter, typical methods are going to be reviewed. These methods contain experimental and computational based practices and the combinations of them. Besides those techniques, other industrial approximation-based schemes will also be introduced. All of the methods will be presented, compared and discussed.

2.1 Full-degree/full-factorial design

To give a full review of various kinds of AC database building schemes, we shortly mention the full-degree/full-factorial method even though it is not the important one in this thesis. For this basic concept, all AC data is from full-factorial WTT or CFD simulation. No other correlation, interpolation and extrapolation is performed. The most used method is the one-factor-at-a-time (OFAT) scheme, i.e. experiments or simulations will be performed on all the possible scenarios, each time changing one parameter. The OFAT is suitable for simple models which do not have a high quantity of design variables, and cases where the WTT is not expensive or the simulation is not computationally intensive.

WTT is widely used to explore the aerodynamics, especially in the early days when the CFD technique is not as advanced as nowadays. In fact, even now, WTT is still a main tool in the aerodynamic discipline. By adopting WTT, we should mention its limitation. There are some kinds of uncertainty inside WTT that can give rise to inaccuracy. Hall et al. (2005) lists some examples, which include scaling and Re effect, cross coupling, model flexibility issue and the support structure interference. For those projects with low quantity of design variables, meanwhile demanding a high-quality AC database, full-degree WTT should be applied.

With the development of computer technology and computational technique, CFD simulation is adopted more and more frequently and widely, and is a good addition besides WTT in

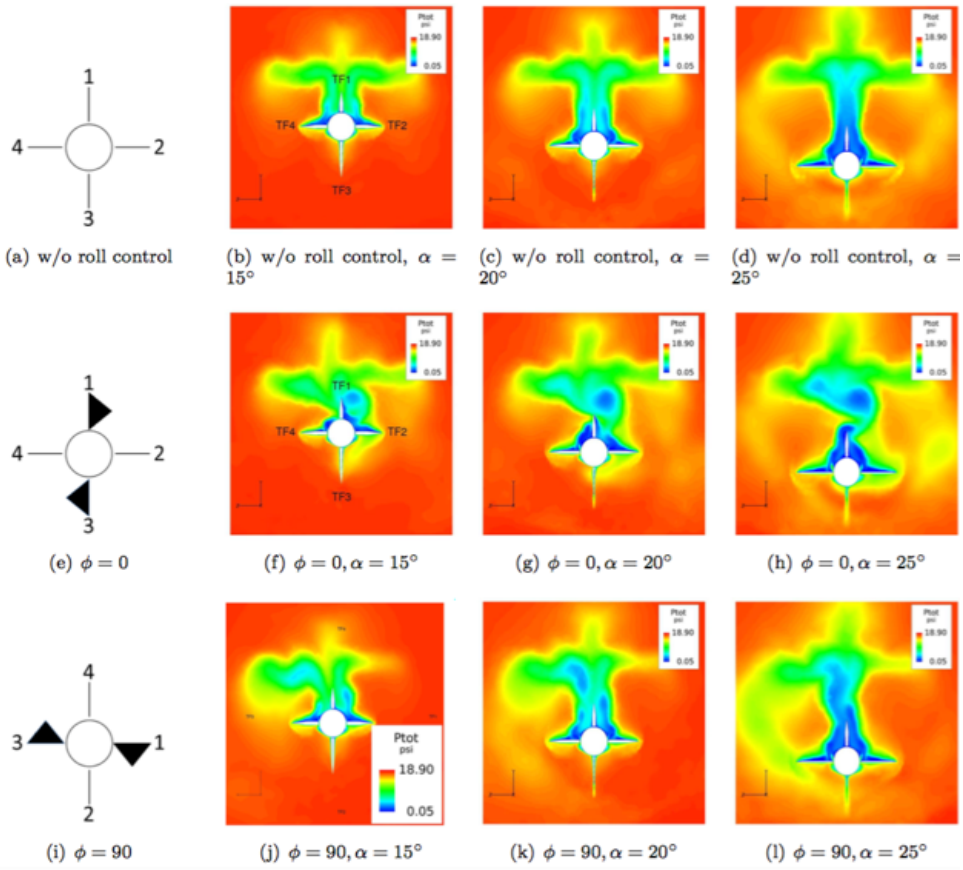


Figure 2.1: Total pressure near tailfin region, aft looking forward. Ma=2.87 Nelson et al. (2015)

aerospace engineering. For the preliminary design period, the aerodynamic database can be built totally with help of CFD. Since for this stage, the need for the accuracy is not quite high. Nelson et al. (2015) set up a non-linear missile rolling moment model of Sparrow missile to evaluate the wing-wing, wing-body interaction in the general design process, to provide a roadmap for the follow up design. The computation result is based on the RavenCFD solver, which is a three-dimensional, Favre-averaged compressible Navier-Stokes based solver. From the view of Design of Experiment (DOE), the authors choose the OFAT approach to set up the computation. The reason for that is the process is not computationally expensive, and they want to catch the AC trends in every dimension accurately. One of the computation results, the total pressure distribution contour near-tail region is demonstrated in Figure 2.1. The nine cases, a combination of three roll angles and three AoAs are all computed and listed in the figure. As the AoA and roll angle vary, we can find the effect on the total pressure distribution and identify the most important effects..

The same strategy is chosen by Coirier et al. (2014), in which AC of various-shaped transonic fins are compared between DATCOM and CFD. For the transonic regime, the flow phenomena are complex, unstable and nonlinear. Therefore it is hard for DATCOM to give decent aerodynamic predictions. The error of lift coefficient slope from DATCOM was as large as 50% Coirier et al. (2014). To establish a high-fidelity database, CFD is applied to compute

Table 2.1: Fin geometry parameter matrix Coirier et al. (2014)

Parameter	Values
Taper Ratio	0, 0.25, 0.5, 0.75, 1
Panel Aspect Ratio	0.25, 0.5, 1, 2, 3, 4
Trailing edge sweep	0, -15, -30, 15, 30
Thickness Ratio	0.5, 0.10
Leading Edge Setback	0, 0.01, 0.02

AC in this regime. The planform of the transonic fin is created by systematically changing five key parameters, see Table 2.1.

2.2 Integration of CFD and WTT

On large scale, modern aerospace industry chooses the integration of WTT and CFD to implement AC databases. The WTT have high fidelity and CFD simulation is more economic (Provided the fidelity of the CFD solver is not quite high and the quantity of test case is large.). The proper integration can generate decent AC database both efficiently and economically. There are many integration schemes, which will be discussed in details in the following subsections.

Sometimes wind tunnel data has overlapped region with CFD results, as in the case of building Orion Crew Module static aerodynamic database Bibb et al. (2011). To establish the drag coefficient curve, wind tunnel data is available in certain region, AoA between 140 and 170 degrees, see blue points in Figure 2.2. And CFD data is generated over the whole range. To blend the two kinds of data, some strategies are applied. First, to use all the data from wind tunnel (because of the high-fidelity versus CFD), corresponding data of CFD between AoA 140 to 170 degrees is eliminated. Then for left and right boundaries of the wind tunnel data range, CFD data was removed to smoothly handle the transition between CFD and wind tunnel data. The final data blending (basic fusion) result is shown in Figure 2.2, the black curve. In fact, this is the most basic blending approach. For some cases the accuracy of such an approach is not high enough. More advanced metamodeling, such as data fusion is needed in that case which will be explained in Section 2.4.

For some scenarios, integration of wind tunnel and CFD data can be performed by way of CFD-based interpolation plus taking the wind tunnel data into account as a guideline. As is presented in Figure 2.3, CFD data at Ma 0.5 and 0.7 are both available on the whole AoA range, and wind tunnel data at Ma 0.6 (blue circle) is available in parts of the interval. Interpolation can be performed by combing all the available data, interpolate inside CFD data and meanwhile using the wind tunnel data as the benchmark. The final integrated data is illustrated in black curve. Such method is also adapted by Nicolì et al. (2006) to perform ground-to-flight extrapolation on launcher VEGA. In case of WTT the in-flight Reynolds number is too large to be realistically duplicated. To achieve AC in such scenario, data extrapolation should be carried out, which should be based on both WTT data and CFD

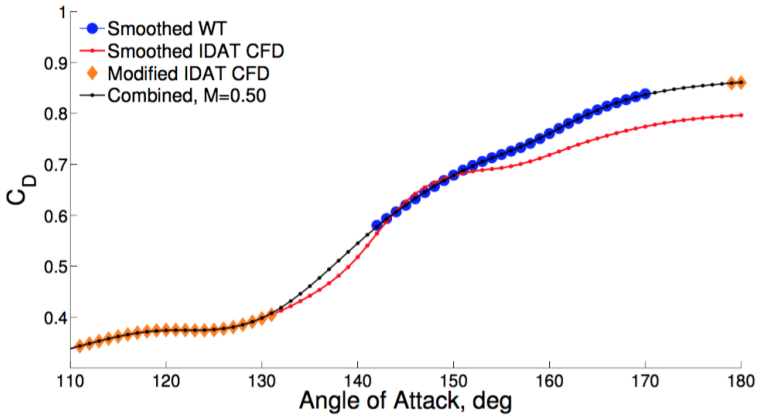


Figure 2.2: Blending of wind tunnel and CFD results, where wind tunnel data dominant, M=0.5 Bibb et al. (2011)

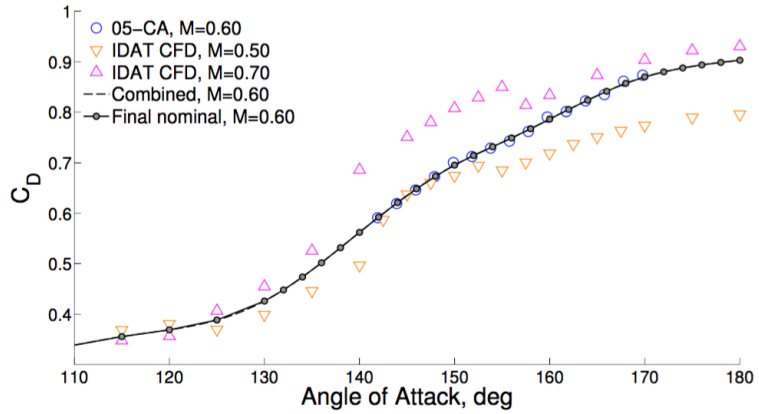


Figure 2.3: Combining interpolated CFD and wind tunnel Bibb et al. (2011)

results. Trends of AC vs Re can be obtained from WTT data. Based on the comparison between WTT and CFD results for lower Re, AC for higher Re can be achieved through extrapolation, see Figure 2.4.

This method is essentially a WTT-supported CFD extrapolation. For approximation between pure CFD data sometimes can lead to low fidelity result. With help of WTT data, the accuracy is highly guaranteed. Of course, the premise of this method is the availability of the WTT data.

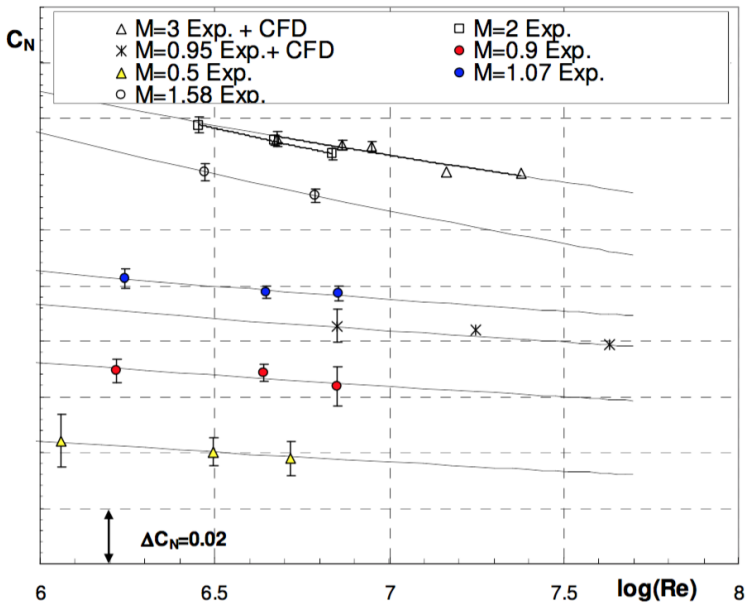


Figure 2.4: Ground-to-flight extrapolation of C_N of VEGA Nicolì et al. (2006)

2.3 Analytical expressions

To prepare for flight simulation, Keshmiri et al. (2005) developed an AC database for a hypersonic air vehicle, using the analytical expression method. The input data, AC of the Generic Hypersonic Vehicle (GHV) is obtained from both wind tunnel and high fidelity CFD computations. Those data cover various flight conditions.

The AC can be expressed as a function of Ma, AoA, surface deflection, and so on. The increment coefficient caused by deflection of the control surface is added to AC of the basic configuration to form the total AC. To achieve the analytical expressions, a 5^{th} order polynomial is fitted to the data. Part of the results Keshmiri et al. (2005), equations of lift coefficient under subsonic flight condition, are shown below:

$$C_L = f(Ma, \alpha, \delta_a, \delta_e) = C_{L_\alpha} + C_{L_{\delta_\alpha}} + C_{L_{\delta_e}}, \tag{2.1}$$

$$\begin{aligned} C_{L_\alpha} = -5.29e^{-4} + 1.57e^{-2}\alpha + 6.02e^{-3}M * \alpha - 3.44e^{-4}\alpha^2 + 1.45e^{-4}(\alpha * M)^2 \\ -5.20e^{-5}\alpha^3 + 3.48e^{-5}\alpha^4 + 2.77e^{-3}M^4 - 2.30e^{-6}\alpha^5. \end{aligned} \tag{2.2}$$

The other ACs are all in the similar formations. The idea behind the analytical expressions is not same as that mentioned in Section 2.2: in this application, the analytical expressions are based on global regression (vs. local regression) by using Matlab codes; variable interactions

are all considered (vs. assuming none interaction between various variables). This concept is especially suitable for building databases for flight simulation, where only part of the AC data is available.

2.4 Approximation Modeling/Metamodeling

With the development of modern computational techniques, numerical computations can be carried out faster and more efficiently. However, for some multi-variable designs and optimization processes, when full-factorial experiment design is applied, there are still quite too many computations to perform, especially where complex computer codes are used. Even with fast computational resources, long computation times are inevitable. In those scenarios, strategies like metamodeling is favorable. Such concept is good at reducing expensive computer analyses meanwhile keeping or even improving the accuracy of the database.The literature on metamodeling is enormous. Prevalent examples are: response surfaces(Knill et al. (1999), Landman et al. (2007)), least square regression (Morelli and DeLoach (2003)), Kriging models (Tang et al. (2005), Laurenceau and Sagaut (2008), Ghoreyshi et al. (2008)), artificial neural network (Rajkumar et al. (2002)), radial basis functions, extrapolation (Nicolì et al. (2006)), multivariate adaptive regression splines etc. Most of the literature will be reviewed hereinafter.

Generally, the metamodeling process contains three steps, performing design of experiment (DOE), choosing a model to present the data and fitting the model to the obtained data. For each step, there are various strategies. Figure 2.5 illustrates the most frequently used methods in each step. In this section, we will review two popular metamodeling methods, response surface methodology (RSM) and Kriging method.

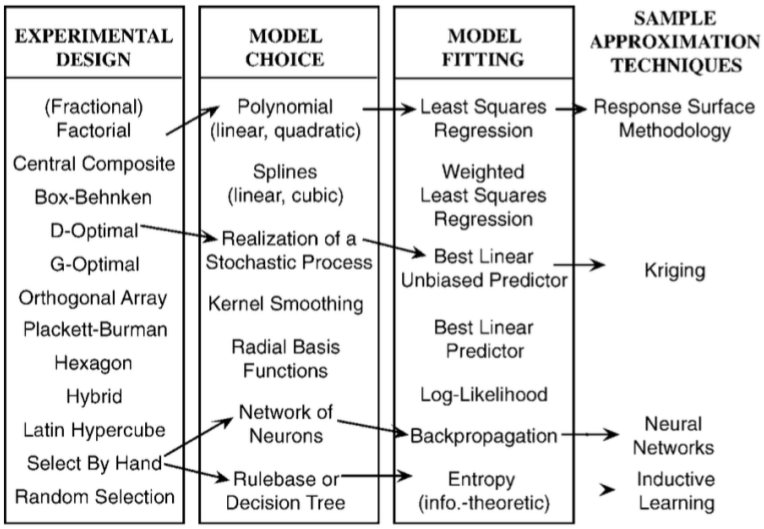


Figure 2.5: Techniques for metamodeling. Simpson et al. (2001)

2.4.1 Response surface methodology (RSM)

For the definition of RSM, different authors describe it differently. Myers et al. (1986) defined RSM as "a collection of tools in design or data analysis that enhance the exploration of a region of design variables in one or more responses". Box et al. (1987) stated that "Response surface methodology comprises a group of statistical techniques for empirical model building and model exploitation. By careful design and analysis of experiments, it seeks to relate a response, or output variable, to the levels of a number of predictors, or input variables, that affects it". Generally speaking, RSM is a strategy that takes advantage of DOE and analytical expertise to find the relation(s) between input and output, in a more efficient and accurate way.

For various input x, there will be different response y. The RSM tries to approximate the unknown response surface function $f(x)$. As in the following equation,

$$y = f(x) + \epsilon, \tag{2.3}$$

ϵ represent random error, which is assumed to be normally distributed and has mean zero and standard deviation σ. The most widely used RMS are low-order-polynomial models. The following two equations:

$$\widehat{y} = \beta_0 + \sum_{i=1}^{k} \beta_i x_i, \tag{2.4}$$

$$\widehat{y} = \beta_0 + \sum_{i=1}^{k} \beta_i x_i + \sum_{i=1}^{k} \beta_{ii} {x_i}^2 + \sum_{i=1}^{k} \sum_{j=1, i<j}^{k} \beta_{ij} x_i x_j, \tag{2.5}$$

give an example of first and second order RSM, respectively. The polynomials in these equations are frequently determined by way of fitting the existing data with the least square method. Survey of Simpson et al. (1997) showed that most engineering applications use second order response surface models, which gave accurate enough approximation results.

For an n-factor problem, when adopting second-order approximation scheme, the quantity of polynomials is $\frac{(n+1)(n+2)}{2}$. To get values for these polynomials, DOE should be applied and a test matrix should be defined. For DOE, there are three fundamental concepts, replication, randomization and blocking (Montgomery (2008)). Randomization is the most important one in experimental design, which means both the experimental factor choosing (the value of experimental variables) and order of various runs should be random. Replication allows

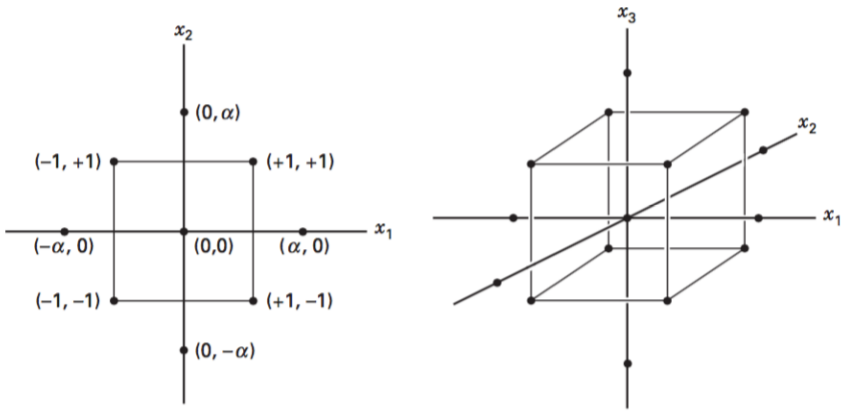


Figure 2.6: Central composite designs for k=2 and k=3 Montgomery (2008)

the experimenter to determine an internal estimate of system error and uncertainty. For the blocking concept, it is a strategy to reduce the potential impact of nuisance factors.

The two-level factorial design is fundamental to DOE. In the design, the values of the various interested factors change between predetermined high-level +1 and low-level −1. The first thing to building a design is to define the region of interest. Then the DOE scheme should be chosen. To fit a second-order model, the central composite design (CCD) is most used, which is illustrated in Figure 2.6. Generally, the CCD consists of 2k full/ fractional factorial runs, 2k axial/star runs and some center runs. After the necessary runs being performed, the analysis of variance (ANOVA) should be carried out. Finally, with results from ANOVA, the regression model can be achieved.

Table 2.2: ANOVA for axial force coefficient response surface Montgomery (2008)

Source	Sum of squares	DF	Mean square	F value	Prob >F
Block	0.005985	1	0.005984528	–	–
Model	0.229783	24	0.009574301	4653.35	<0.0001
A	0.173022	1	0.173022038	84092.96	<0.0001
C	0.019821	1	0.019820604	9633.30	<0.0001
...					
AC	0.000714	1	0.000713931	346.99	<0.0001
...					
Residual	0.000263	128	2.05751E-06	–	–
Cor. total	0.236031	153	–	–	–

Landman et al. (2007) provided an RSM example, evaluating AC of a blended-wing-body jet with eight design factors (multi-elevons, AoA, sideslip angle). In the DOE process, the face-centered CCD (FCD) is used and a 1/2 fractional factorial design is generated. Part of the axial force coefficient ANOVA is listed in Table 2.2. With ANOVA, the RSM model can be generated. The comparison of C_N between RSM and the traditional OFAT is illustrated in Figure 2.7. The comparison shows that the RSM method has almost the same accuracy as that of OFAT, but with a much lower experimental cost, 128 sampling points for RSM vs. 287 for OFAT.

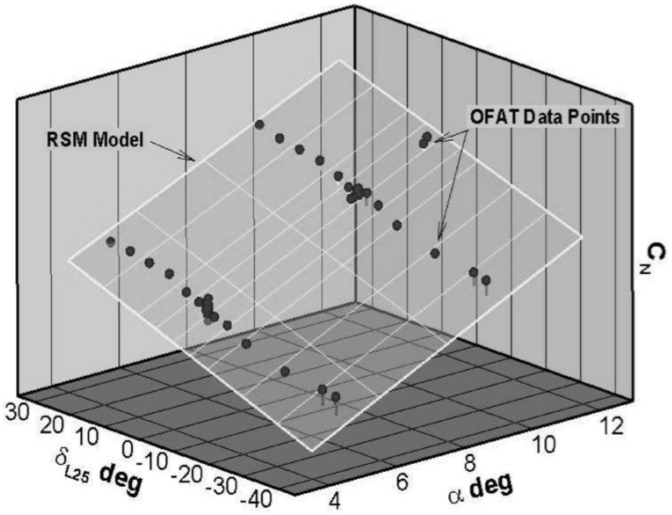


Figure 2.7: Comparison between RSM and OFAT, C_N Montgomery (2008)

RSM is an efficient way to get an approximate AC database, provided that the noise in the sourse data is properly deleted. From the review of Simpson et al. (2001), RSM tends to fail in cases where there are more than 10 factors or cases where highly nonlinear responses exist. In deterministic applications where there are some fairly well-behaved factors, another option for exploration is suggested to adopt the standard RSM approach augmented by the robust design, applying a Taguchi noise-filter to reduce the misleading noise data.

2.4.2 Kriging technique

Another important metamodeling scheme is the Kringing technique. The theory of Kriging is developed by the French mathematician Georges Matheron in the early 1960s. It is based on the seminal work of D. G. Krige on mining data. Sometimes it is referred to as DACE (Design and Analysis of Computer Experiments). Different from traditional wind tunnel experiment, for the DACE, the experiment is deterministic and there are no randomly emerging errors. Strategies applied in DOE, like replication, randomization, and blocking, are not suitable to DACE anymore. As the RSM, the DACE uses a Kriging function to estimate the target function, $\widehat{y}(x^*) = \mu + \epsilon$, where μ is the average value and ϵ is the same as that in the RSM, normally distributed error term.

For the metamodeling methods, sampling strategy is quite important. It affects the efficiency and accuracy of the approximation process. An effective pattern should put samples near regions where nonlinearity emerges. The sampling place should be located where mean squared error (MSE) is largest. And with the new sampling point, a new MSE-max point can be found, which is to be chosen as next sampling point. In this way, after the maximum MSE meeting some standard, the Kriging can produce an accurate enough target function. The closing-in process is illustrated in Figure 2.8, where pressure coefficient over an airfoil is approximated.

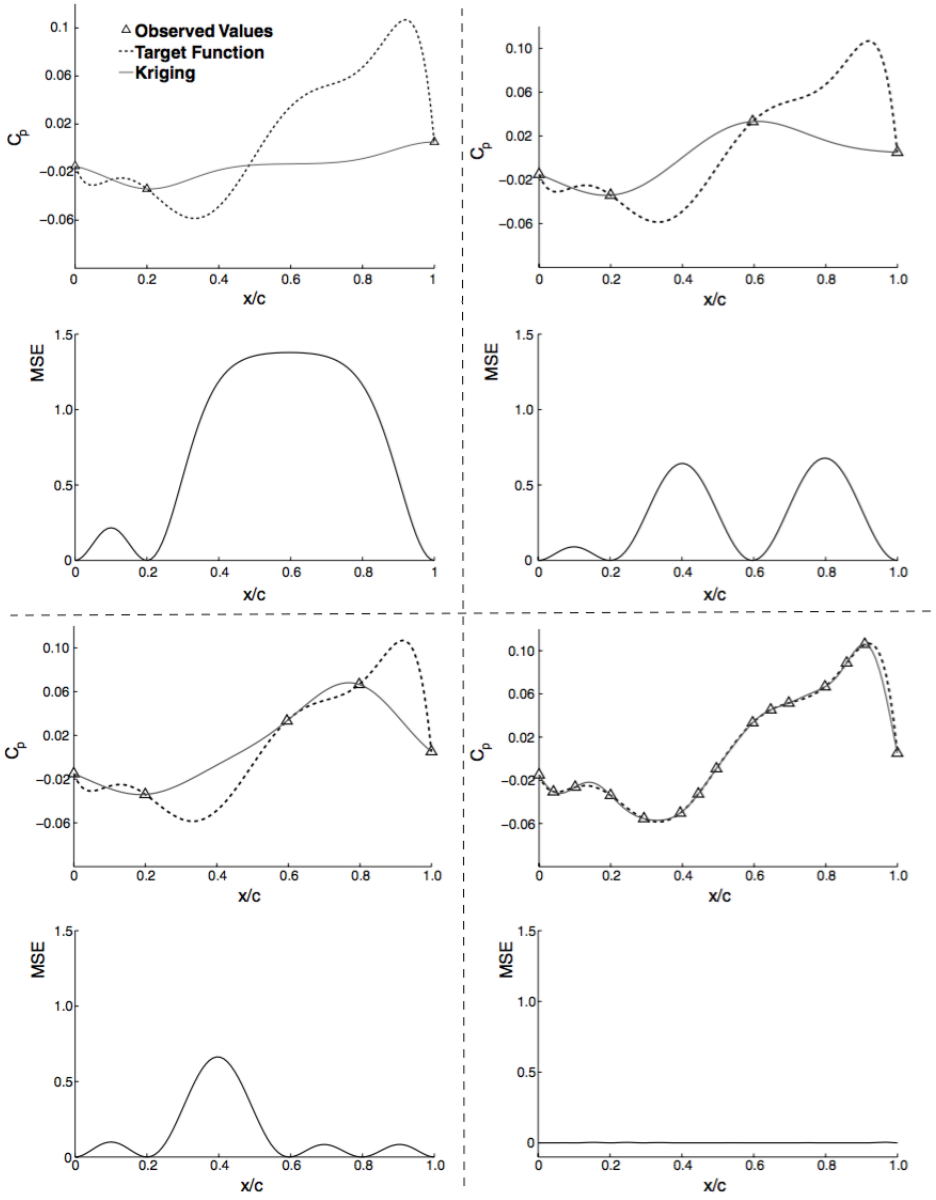


Figure 2.8: Kriging approximation with MSE sampling methods Ghoreyshi et al. (2009)

The MSE sampling method pays attention to the global search for the target function, based on the weighted distance correlation for the error term. Different from it, there is another improved sampling method, combining local and global search. This strategy takes advantage of the expected improvement function (EIF), defined in

$$E[I(x)] = \begin{cases} (y_{min} - \widehat{y})\Phi(\frac{y_{min}-\widehat{y}}{s}) + s\phi(\frac{y_{min}-\widehat{y}}{s}) & s > 0 \\ 0 & s = 0 \end{cases} \tag{2.6}$$

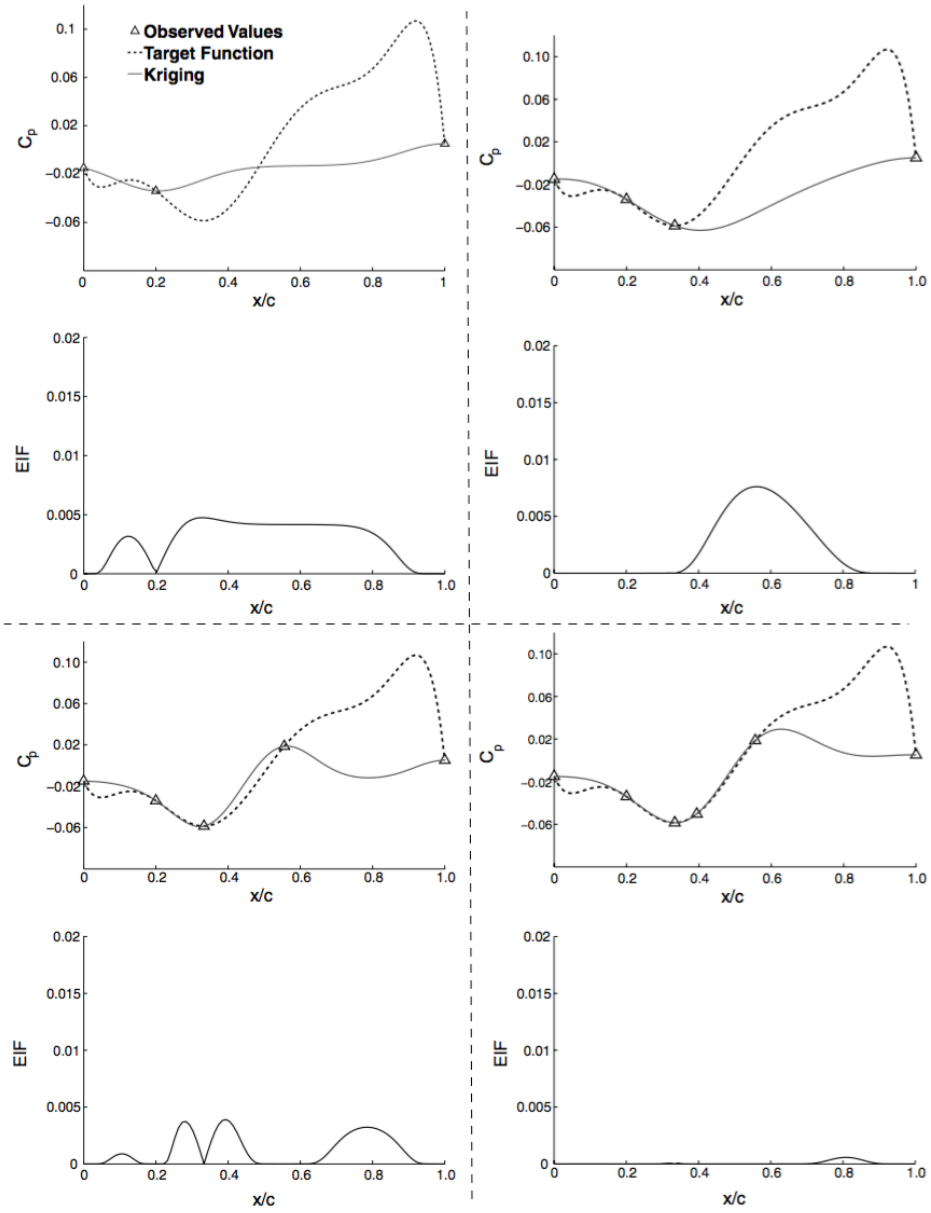


Figure 2.9: Kriging approximation with EIF sampling methods Ghoreyshi et al. (2009)

where s is the standard deviation. In this equation, the y_{min} equals $min(y^{(1)}, y^{(2)}, \cdots, y^{(n)}$, and ϕ and Φ are normal density and distribution functions, respectively. The new sampling point locates where EIF has a maximum value. Kriging approximation using EIF is illustrated in Figure 2.9 (same background as Figure 2.8). From the comparison between MSE and EIF, we can find that the MSE method is trying to reduce the uncertainty caused by gaps in the parameter space. However, the EIF focuses on finding local samples where global maximum can be reduced.

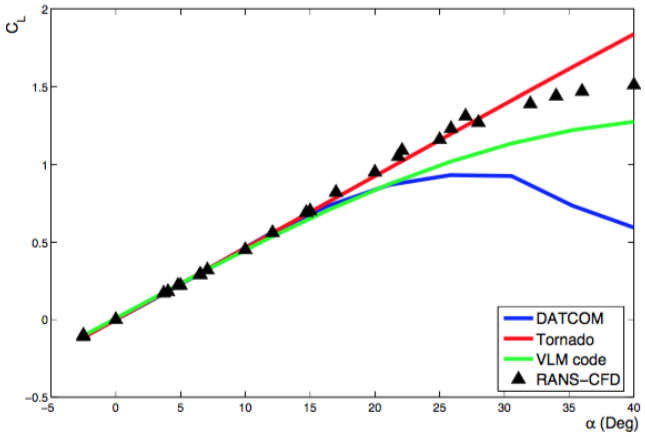
Ghoreyshi et al. (2008) explains an example of fusion of DATCOM, VLM (Approximating

code based on linearization of the compressibility effects) and CFD (RANS-based code) to get AC of a unmanned aerial vehicle (UAV). The assessment shows that the fusion result achieves decent prediction when compared to pure RANS results, which is illustrated in Figure 2.10. Of the three data sources, DATCOM is good at low AoA prediction, and VLM provides accurate AC trends at high AoA, and RANS-based CFD has the best accuracy in the full range. In the project, it is claimed that, with a priori knowledge of the AC trends, to get a proper AC database, only one RANS sample is needed. Work of Tang et al. (2005), using a similar scheme, has a database established with only 31% of full RANS-based CFD computation. It is no doubt that good application of the data fusion strategy leads to a significant reduction of computation time.

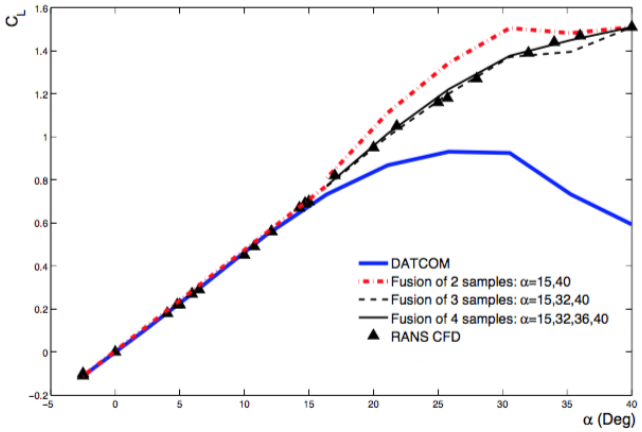
For the Kriging approximation scheme, a Matlab-based code can be generated by theories mentioned earlier in this section. Or some ready-to-use toolbox can be chosen, famous of which is the publicly-available Matlab-based DACE toolbox from Lophaven et al. (2002).

From this section review, we can see that both RSM and Kriging modeling can give decent estimating results. In fact, as in work of Simpson et al. (1997), which carries out a preliminary comparison between a second-order RSM and Kriging modeling, on an aerospike nozzle which has three geometry design variables, neither of them can consistently outperform the other. A similar result is from Giunta (1997), and he presents an investigation on multidisciplinary design optimization of a high-speed civil transporter, which has 5-10 design variables. RSM and Kriging modeling yield similar results.

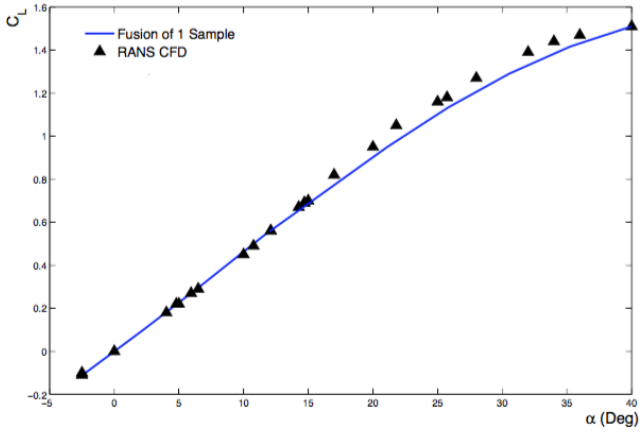
As suggested by Simpson et al. (2001), for the deterministic and highly nonlinear projects, meanwhile the design factors is moderate (e.g. less than 50), the Kriging modeling is the better choice to perform the approximation, even though it is more complex than the RSM. Also as mentioned hereinbefore, in deterministic projects with a few well-behaved factors and noise-affected factors, one can choose the standard RSM augmented by a Taguchi noise-filter.



(a) DATCOM vs. RANS



(b) VLM vs. RANS



(c) Fusion vs. RANS

Figure 2.10: Evaluation of data fusion with DATCOM, VLM and RANS-based CFD Ghoreyshi et al. (2008)

2.5 Adaptive space transformation

Aforementioned methods all tend to have the same deficit. They can only be used inside the convex hull of known database, a.k.a. interpolation. Outside of the familiar region, the accuracy of those estimating methods tends to be poor, and may well lead to totally wrong directions. To solve this issue, a new concept, known as adaptive space transformation (AST) is promoted by Luo et al. (2015). The method tries to detect the underlying invariant relation with the available database. By using 'invariant', we describe the consistent law that can be applied on various Ma regime, say from transonic, supersonic to hypersonic. During the detecting process, real physics is referred as a guidance. Once the relation is detected, the prediction can be performed, both inside and outside the known regions, as long as corresponding physics is clear.

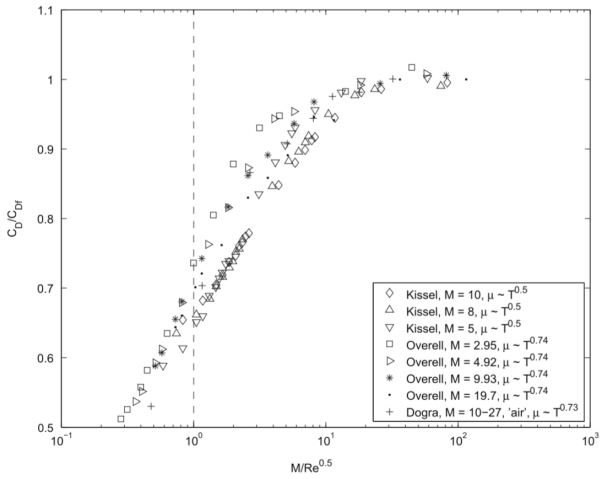
Different from other correlation methods, AST takes advantage of the scaling parameters, which are functions of aerodynamic parameters. Such examples are Tsien's parameter: (Tsien (2012)) and Cheng's rarefaction parameter (Cheng (1961)). Macrossan (2007) compared those parameters by correlating drag force under rarefied air conditions. The evaluation results are illustrated in Figure 2.11. From the figure, it is clear that the target coefficient is proportional to the scaling parameters mentioned above.

As shown in Figure 2.11, the principle behind AST is the recovery capability of the invariant relation, which can provide unseen parts of the database. In reality, only fractional data is available, because of restriction of experimental equipment, budget or computational source, like shown in Figure 2.12(b). By way of applying parameter correlation, the key relationship of the variables can be obtained. In this case, an ellipse is the plane curve surrounding two focal points and sum of the distances to the two points is same for points on the curve. After obtaining relation mentioned above, the missing parts of the data (ellipse) can be found easily, Figure 2.12(a).

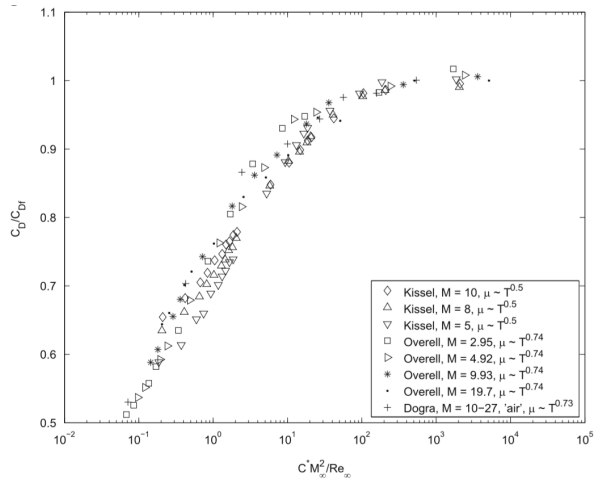
As mentioned before, one important shortcomings of the traditional approximation based methods is that they can only be applied inside of the known data region. Inside the region, the accuracy is satisfactory. Otherwise, the estimated results can be quite misleading. For example, aerodynamics of the supersonic regime is quite different from that of the hypersonic regime. If database in the former regime is known, it is totally wrong to pursue hypersonic database by way of extrapolation. New physical phenomenons could appear, such as dissociation and ionization. And those physics can become the new dominant force in the aerodynamics. That is why many famous aerodynamicists, like von Karman and Tsien, prefer to use the scaling parameters, which can get over such changes.

However, to find the proper scaling parameters is an expertise-based job. It needs complicated theoretical aerodynamic knowledge, rich researching experience and verification from a large amount of experimental data. Thanks to the development of computational ability, it is now possible to detect the kernel of space transformation automatically by way of genetic programming. In this way, the AST method is suggested.

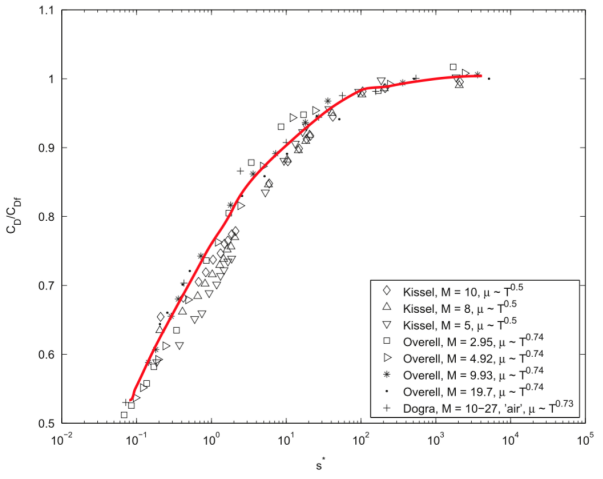
To attain an AST scaling parameter, four steps need to be carried out, see Figure 2.13. The



(a) Tsien's



(b) Cheng's



(c) Ideal correlation

Figure 2.11: Correlation results with methods Luo et al. (2015)

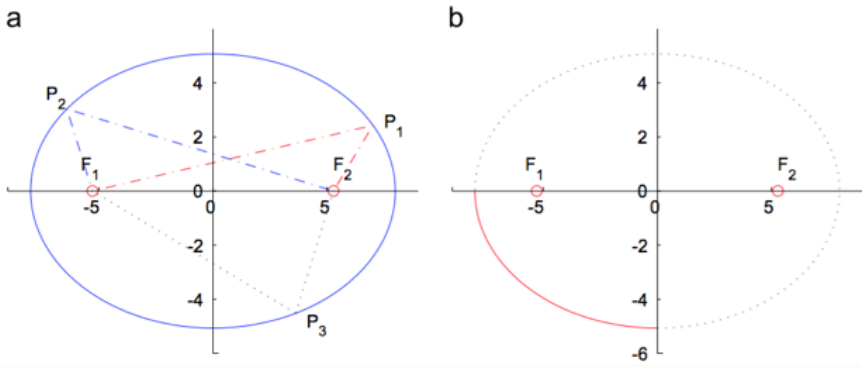


Figure 2.12: Variations and invariant of the ellipse. (a) Detection of an invariant relation. (b) Recovery of the unknown part by the invariant relation. Luo et al. (2015)

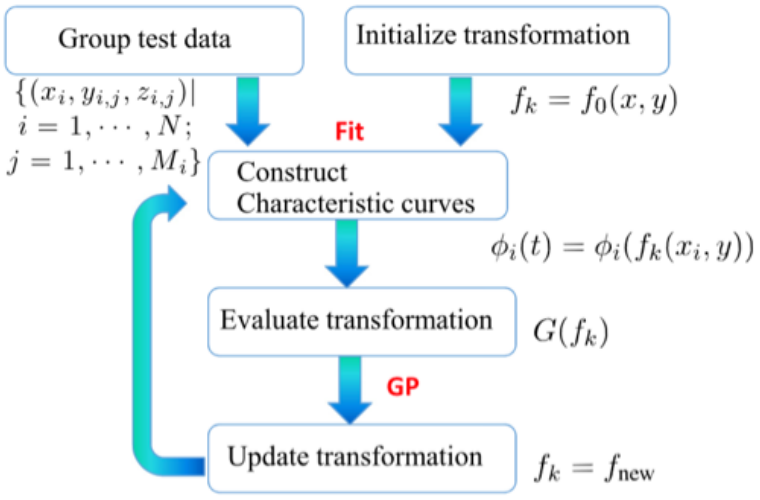
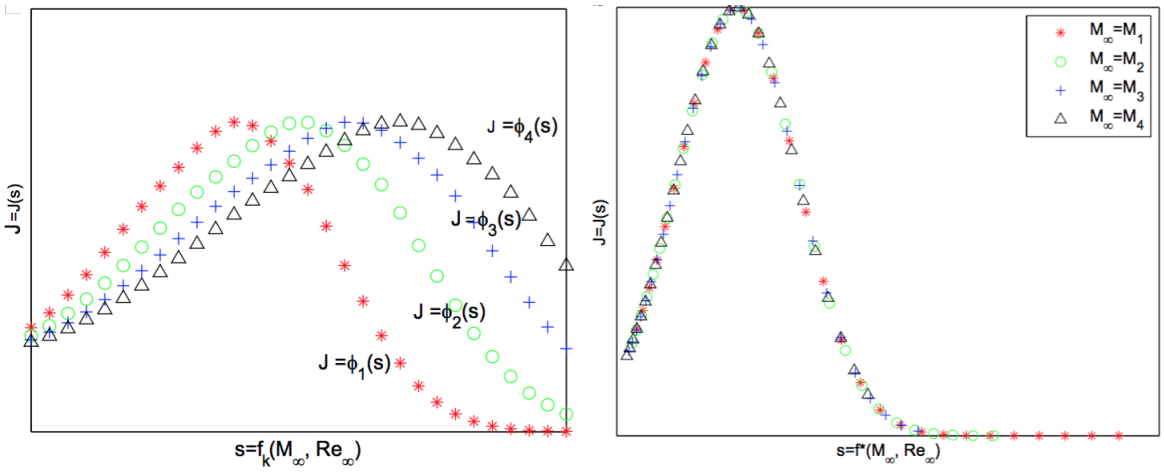


Figure 2.13: Flow chart of AST. Luo et al. (2015)



(a) Constructing initial characteristic curves

(b) Achieving optimization result through iteration

Figure 2.14: AST step two, Luo et al. (2015)

example in literature is introduced here briefly. The case is about detecting invariant of Mach number. Here are the four steps:

1. Divide the available database to several groups, e.g. by various Mach regime.
2. Construct the characteristics curves in each Mach regime and overlap them in the same coordinates. See Figure 2.14 (Left).
3. Modify and optimize the transformation kernel by genetic programming to make the characteristics curve identical with each other. See Figure 2.14 (Right).
4. Repeat step two and three until certain error criterion is met.

Among the four steps, the third one is the most critical in AST. Optimization function should be set up to evaluate the overlapping-degree of the characteristics curves. For a simple case, only one factor x needs to correlate, a typical optimization (Luo et al. (2015)) is like

$$minG(f) = \sum_{i=2}^{N} \sum_{j=1}^{i-1} \int_{a}^{b} \left\| \phi_i[f(x_i, y] - \phi_j[f(x_j, y] \right\| ds / S_{convhull}, \tag{2.7}$$

where the function describes characteristic curve and $s = f(x, y)$ is the transformation kernel in the optimization process.

For the AST, if the scaling parameter is found, it can give a good approximating result. However, the transformation kernel is not easy to achieve and complex genetic programming

should be introduced, tested and modified. The pursue process can be much longer than expected, especially when there are a large number of design variables. For the project with not many design factors, this method can be applied.

Chapter 3

Numerical flow modeling and data fusion

The numerical tools and concepts used in this project is going to be discussed in this chapter. Section 3.1 provides a brief introduction to the semi-empirical method, Missile DATCOM. Section 3.2 focuses on the application of the CFD software, NUMECA. In this section, the selection of turbulence model, definition of computation domain, boundary layer treatment, mesh independence study, etc., will be discussed in detail. In Section 3.3, the data fusion theory adopted in this project will be fully explained. In the last section, the sampling strategy which is used during the data fusion will be briefly introduced.

3.1 Missile DATCOM

To have a better understanding of the semi-empirical method, we firstly give a short introduction of Missile DATCOM. Missile DATCOM is a branch of Data Compendium (DATCOM), which is developed by the McDonnell Douglas Corporation under contract with the United States Air Force in 1970s. DATCOM is a 1500-page methodology for determining stability and controllability of aircraft. In 1979, the methodology was programmed in form of Fortran language which was named USAF Digital DATCOM (There is a branch with the name Missile DATCOM). The semi-empirical method offers an easy-to-use, rapid and economical estimation of aerodynamic derivatives. Typically, the DATCOM input file consists of flight conditions (Ma, Altitude, Re, etc.), geometry data of aircraft configuration and desired outputs.

DATCOM can only estimate AC of stability and controllability in the subsonic and supersonic regime. For the transonic regime, corresponding data can be obtained by way of pure correlation. Accuracy of such correlation result usually tends to be quite low and therefore requires special attention when a given accuracy is required. Another aspect need to note is that DATCOM is based on inviscid-flow model and the viscous contributions are added later. This may sometimes lead to an erroneous drag force prediction.

DATCOM is based on low-Re and out-dated (in terms of both aircraft geometry and the flow conditions) aircraft databases . The out-dated empirical data can lead to serious error when predicting on modern-configured aircraft/missiles. Besides that, the accuracy of DATCOM is not good in flow separation region. As mentioned by Coirier et al. (2014), the error of lift coefficient slope from DATCOM can be as large as 50%. One typical example w.r.t. DATCOM error is shown in Figure 3.1. The centre of pressure of a triangular wing computed by both CFD and Missile DATCOM 3/11 is shown. We can see that the center of pressure is estimated a bit forward than the WTT result at low Mach, and beyond Mach 1, it is backward. The discrepancy is quite large in transonic regime, especially for the t/c=0.1 case, the shock-induced-separation-caused pressure center fluctuation is not captured by the Missile DATCOM. The explanation can be Missile DATCOM's linear-theory-based approximate modeling in the transonic regime. However, even with these shortcomings, DATCOM is still of great value, since for concept or preliminary design purpose, it can provide fast and moderately accurate AC data. When needed the AC data can be further refined and corrected by CFD computations.

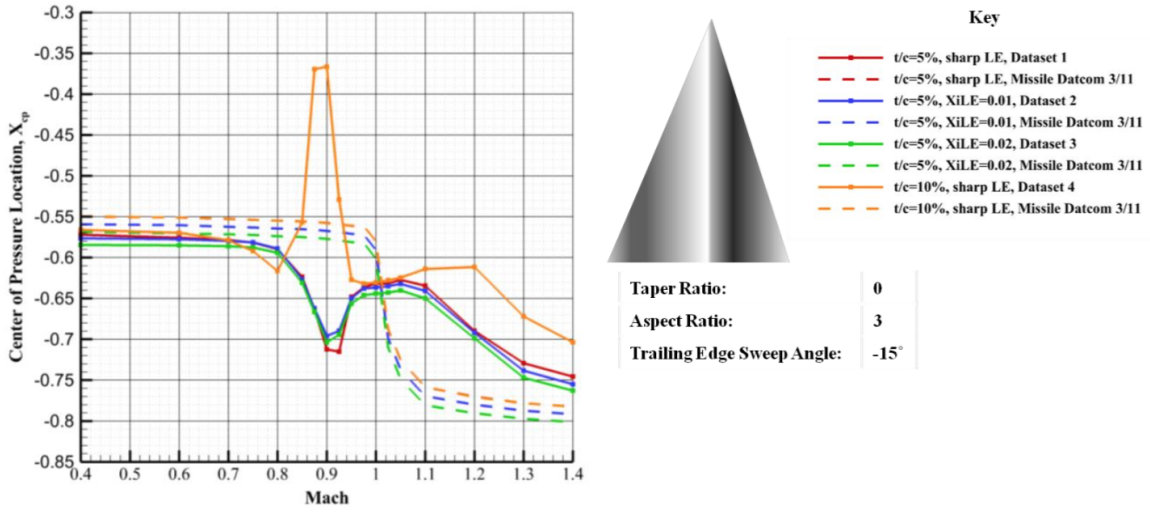


Figure 3.1: Centre of pressure from both CFD and Missile DATCOM 3/11, dash lines and solid lines representing the results from DATCOM and WTT, respectively Coirier et al. (2014)

The concept behind the Missile DATCOM is that it uses a modular approach towards the aerodynamic coefficients. For AC estimation of the body-alone model, there are several contributions that need to be computed. These are the potential and viscous normal force, skin friction, pressure and wave drag, potential and viscous pitching moment. The total lift, drag and pitching moment are combinations of the contribution mentioned above. Estimation of those lumped parts is based on various aerodynamic theories. From Blake (1998) and EngelenF.M. (2012), there are about ten types of methodologies involved in the body alone aerodynamic estimation. Most important ones are the Second Oder Shock Expansion (SOSE), the Van Dyke Hybrid Theory (HTVD), the Modified Newtonian Theory (MNT), the Allen and Perkins Viscous Crossflow Theory (APVCT), the Van Driest II Theory, and the Jorgensen viscous crossflow theory. Table 3.1 gives the applicability of those theories.

The application of Missile DATCOM will be further explained in Section 5.1.

Table 3.1: Applicability of body-alone aerodynamic theories Blake (1998)

Characteristics	**Subsonic/Transonic**($M < 1.2$)	**Supersonic** ($M > 1.2$)
CN-potential	Empirical data/ SBT	SOSE/HTVD/MNT
CN-viscous	Jorgensen viscous crossflow	
CM-potential	Empirical data/ SBT	SOSE/HTVD/MNT
CM-viscous	Jorgensen viscous crossflow	
CA-friction	Turbulent: Van Driest II, MDAC West Handbook Laminar: Blasius, Hoerner Fluid Dynamic Drag	
CA-pressure/wave	$M < M_{crit}$: USAF DATCOM section4.2.3.1 $M > M_{crit}$: Transonic area rule	SOSE/HTVD/MNT
CA-alpha	APVCT	SOSE / HTVD / MNT

3.2 Set-up of the numerical simulation

3.2.1 NUMECA

$NUMECA^{TM}$ is the CFD software that is used in this project which is a fully integrated CFD platform. It is designed to solve both internal and external flow problems. It can handle incompressible flows as well as supersonic and hypersonic, compressible-flow problems. NUMECA consists of three modules. These three modules are integrated in a user-friendly environment.

The first module is the $HEXPRESS^{TM}$, which is able to mesh complex geometries in 2D and 3D. It mainly serves as an auto all-hexahedral unstructured meshing system. Alternatively the user can also choose the hybrid module where tetrahedral, pyramid and prism cells can be used. The fact that for most cases only hexahedral cell can be used may lead to high cell numbers during the meshing process. The second system is $Fine^{TM}/Open$ and this is the module where CFD computational parameters such as physical configuration, boundary condition, numerical schemes and control parameters are defined. This is the most used module during the CFD simulation. Computation starting/ending is performed through $Fine^{TM}/Open$ and also important parameters like the convergence history are shown in this module. The final system is $CFView^{TM}$ module, where visualization and data processing (like flow characteristics checking, y^+ viewing etc.,) can be carried out.

3.2.2 Numerical method and turbulence modeling

Most modern CFD software is based on Navier-Stocks (NS) equations (Anderson Jr (2010)):

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{V}) = 0, \tag{3.1}$$

$$\begin{aligned}
\frac{\partial(\rho u)}{\partial t} + \nabla \cdot (\rho u \mathbf{V}) &= -\frac{\partial \rho}{\partial x} + \rho f_x + (F_x)_{viscous}, \\
\frac{\partial(\rho v)}{\partial t} + \nabla \cdot (\rho v \mathbf{V}) &= -\frac{\partial \rho}{\partial y} + \rho f_y + (F_y)_{viscous}, \\
\frac{\partial(\rho w)}{\partial t} + \nabla \cdot (\rho w \mathbf{V}) &= -\frac{\partial \rho}{\partial z} + \rho f_z + (F_z)_{viscous},
\end{aligned} \tag{3.2}$$

$$\frac{\partial}{\partial t}[\rho(e + \frac{V^2}{2})] + \nabla \cdot [\rho(e + \frac{V^2}{2})\mathbf{V}] = \rho \dot{q} - \nabla \cdot (p\mathbf{V}) + \rho(\mathbf{f} \cdot \mathbf{V}) + \dot{Q}'_{viscous} + \dot{W}'_{viscous}. \tag{3.3}$$

There are several ways to deal with the NS equations, one of which solves the equations directly, called Direct Numerical Simulation (DNS). DNS solves the equations on all turbulent length scales, which demands rather small grid and time steps, leading to a tremendous consumption of computational resource. For most aerodynamic problems including this project, DNS is not feasible. Another method to deal with the NS equations is to simplify them by only solving the large scale eddies and it is called Large Eddy Simulation (LES). For this project, even this simplified method is not suitable, since it still requires a large amount of computational resources. Then comes to the most used simplified method, the Reynolds-Averaged Navier-Stocks (RANS) concept, which splits the velocity component into a fluctuating value and a time-averaged value. The split concept brings much lower computational resource, compared to DNS and LES. Due to its low cost in terms of computation power and running time, RANS model is used commonly.

There are also other approaches such as NS equation based Detached Eddy Simulation (DES) and non-NS-based Lattice Boltzmann Method, which is based on the Boltzmann equation and considers the aerodynamic flow as microscopic particles. However, when compared to RANS, these methods are either computationally expensive or less mature, when applied to the launcher scale models. Considering all the factors mentioned above, RANS is the most suitable method for this project.

The RANS model gives rise to one unknown value, the Reynolds Stress. This should be resolved by way of applying turbulence models. (The need for additional equations to model the new unknowns is called Turbulence Modeling.)There are two ways to resolve the Reynolds Stress. The first one is using an isotropic value for the turbulent viscosity, which is called Eddy Viscosity Model. This is the most commonly used. Another method regards the Reynolds Stresses (6 values) are anisotropic. And those values can be achieved by solving the Reynolds Stress Model. For the second method, extra equations need to be solved, increasing the computation time considerably. In this project, the first method will be adopted. For the past ten years, there have been many Eddy Viscosity Models, the most popular ones are:

- `Sparlart-Allmaras Model`
  The one-equation model is more suitable for attached flow. it is famous for its robustness.

- $\kappa - \epsilon$ `Model`
  The two-equation model, for free shear and non-wall bounded flow. It was the former industrial standard.

- $\kappa - \omega$ `Model`
  Another two-equation model, for wall-bounded flow. it is not commonly used in industry.

- `Shear Stress Transport (SST) Model`
  Another two-equation model combining the advantage of $\kappa - \epsilon$ model in free stream and $\kappa - \omega$ model in wall bounded flow. it is the new industry standard in recent years.

In NUMECA, several turbulence models are available: Spalart-Allamras model, $\kappa - \epsilon$ model, $\kappa - \omega$ SSt model, SARC model, etc. All these models have both extended-wall function model and near-wall modeling model. From research of Kok (2000), the $\kappa - \omega$ SST turbulence model has the best accuracy when applied to launchers. The accuracy of this turbulence model is also proved in work of Catalano et al. (2007). However, when the author tried to apply the $\kappa - \omega$ SST model in NUMECA, it turned out that it is hard to achieve the convergence during the simulations. No matter how the discretizing scheme or the CFL number is adjusted or mesh refinement is applied etc., it is proved that only the Spalart-Allamras turbulence model is suitable for applying on the launchers (for it is stability and robustness). By now, this can not be explained. One of the possible reason is that NUMECA is mainly designed to simulate the flow inside the turbines and when applied to supersonic or hypersonic external flow, some potential error can happen. Though the chosen turbulence model is not perfect, the validation of NUMECA, which will be further explained in Chapter 4 shows that accuracy of the simulation results is decent when comparing the WTT data.

3.2.3 Size of computational domain and Boundary conditions

First of all, a computational domain where flow simulation is performed should be defined. In this domain, the volume is discretized into small cells in which the flow aerodynamic characteristics are computed. A good choice of the simulation domain is critical to the accuracy and efficiency of the simulation. The domain size should be defined properly to ensure there is sufficient distance upstream and downstream from the object of interest so that the boundary conditions do not directly affect the flow solution. The choice of the domain can be determined by experience and trial-and-error. In this project, the geometry size of the computational domain is set with referring to work of Kok (2000). For the supersonic test case, the total length is set to 10.1 times of the launcher's length. In front of the nose, 0.1 time's length is set to settle the mesh in front of the nose. For the subsonic case, the total length is set to 15 times of the launcher's length, with 5 times in front of the nose and 9 times length behind the launcher base. The difference between the two settings is that for the supersonic flow, the flow disturbance can't propagate upstream and its effect is limited to the region of the downstream. With respect to width and depth, the values are both set as 10 times of the first-stage diameter of the launcher. In this project, unstructured meshing strategy is adopted and the larger size of the computational domain will not increase the number of the

total cell to a large extent, since the cell size near the outer domain of the region is quite large compared to that near the launcher surface. In this project, we only consider effect of AoA, so only half of the mesh is enough for the simulation. We will choose the left part (when facing the flight direction). Besides the definition of the computation domain, we also need to define several refine boxes in corresponding regions, the sizes of which are based on experience and trial and error during the simulations. when using NUMECA, the author found that if some regions (e.g. regions after the backward-facing step or regions after the nozzle) are not resolved fine enough, divergence tends to happen. It is highly suggested that enough refine boxes are placed in such regions. Generally, the refine boxes won't bring high increase in cell numbers and thus adding simulation workload.

The domain can be set symmetrically, with the pitching surface as the symmetry plane. Overview of the domain for VEGA launcher is illustrated in Figure 3.2. Five surfaces of the domain (front, back, left, top and bottom) are set as external boundary condition, where flow property is set the same as the free stream. The right surface of the domain, which is symmetrical surface of the domain is set as mirror surface.

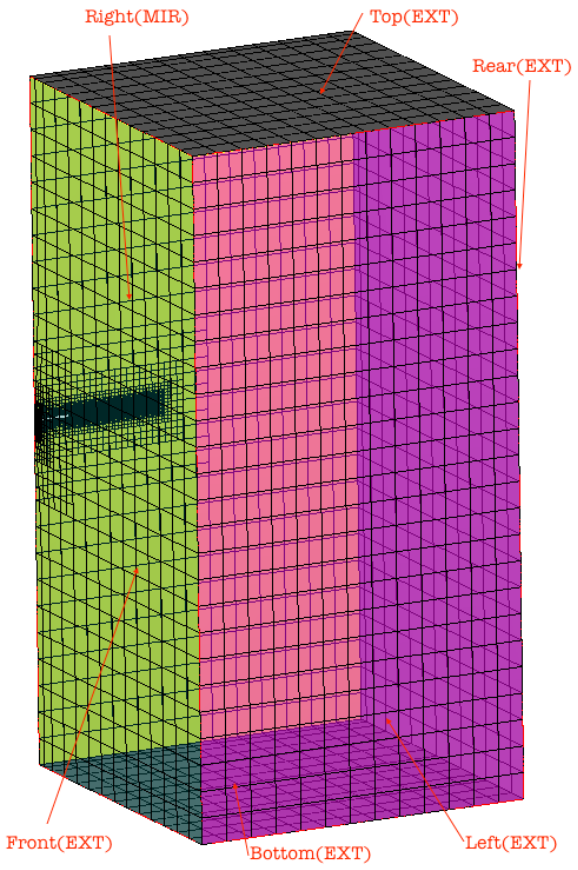


Figure 3.2: Overview of computational domain and boundary settings of VEGA

3.2.4 Wall treatment and boundary layer

When simulating the flow around the surface of the launchers, one needs to handle the near-wall region carefully. Perpendicular to the surface, the flow region can be split into three layers, the outer layer, the inner layer and the overlap layer, see Figure 3.3. These three layers have totally different characters. The outer layer is faraway enough from the object surface and the flow inside this layer is not affected by the viscous effect. The viscous layer is the region closest to the launcher surface and it is mainly affected by the viscosity effect rather than the free stream. There is a connecting part between the outer and inner layer called the overlap/log layer, where both free stream characteristics and the viscous effect can have influence on the flow in this region. For the inner layer, it can be subdivided into viscous sublayer and buffer layer.

These different kinds of layers can be categorized by the vertical distance from the object surface, which is evaluated by the non-dimensional wall distance, y^+. This value is defined (Pope (2001)) in

$$y^+ = \frac{u_\tau * y_{\frac{1}{2}}}{\mu}, \tag{3.4}$$

where the u_τ is the friction velocity shown in

$$u_\tau = \sqrt{\frac{\tau_{wall}}{\rho}}, \tag{3.5}$$

where τ is the wall shear stress

$$\tau_{wall} = \frac{1}{2} C_f \rho U^2, \tag{3.6}$$

and the $y_{\frac{1}{2}}$ is half of the first cell off the surface of the launcher. In Equation 3.6, the friction coefficient is related to Re, which is based on estimation from experience, the friction coefficient of turbulent boundary flow over a flat plate. One example is the 1/7th velocity profile based on flat-plate boundary layer theory from White (2003) and is expressed in

$$C_f \approx 0.027 * Re^{-\frac{1}{7}}. \tag{3.7}$$

There are two ways to resolve the boundary layer near the object surface, the near wall modeling and the wall function. For the first method, it solves the complete boundary layer

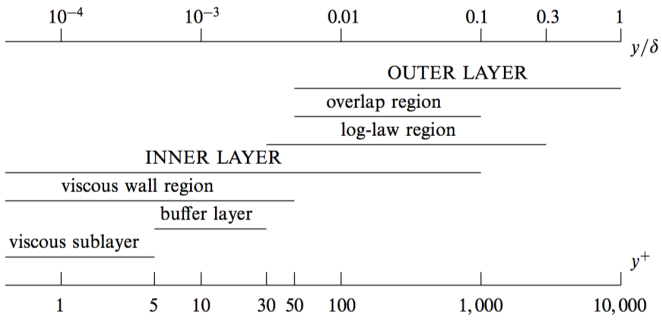


Figure 3.3: various wall regions and layers defined in terms of y^+ (Pope (2001))

and the cell size should be fine enough, the corresponding y^+ should be lower than 5. For the wall function method, the boundary layer is solved based on experience-based approximation, so it does not need the discretization like near wall modeling. The corresponding y^+ should be larger than 20 and lower than 100 in $Fine^{TM}/Open$ (NUMECA (2012)). From work of Catalano et al. (2007), for most part of the launcher, the y^+ should be set to 4-6. Work of Levy et al. (2014) also provides some gridding guidelines for meshing of the launcher. Those values can be used as the initial points in the gridding process of the simulation.

After the simulation is performed, one should verify if the real y^+ over the object surface is inside the acceptable range. In NUMECA this can be checked after approximately one hundred iterations on the fine grid within $CFView^{TM}$, rather than waiting until the simulation finishes.

3.2.5 General process

In this part, the general process of performing the CFD simulation is described. The whole process can be divided into three parts: computation domain generating and meshing, solver settings and post-simulation visualization.

- `Domain generating`

First, the computation domain should be generated. This is carried out in the $HEXPRESS$ module. $HEXPRESS$ accepts several kinds of CAD models as input, like $Parasolid^{TM}$ model (files with extension '$.x\_t$') and $CATIA$ model. Due to license issue, it is only available to use the $Parasolid^{TM}$ model. The CAD model can be generated from the 3-D CAD software $SOLIDWORKS^{TM}$. Geometry configuration of the launcher VEGA in this project is illustrated in Figure 3.4. After the CAD model is read into $HEXPRESS$, corresponding editing such as Boolean operation can be carried out to generate the desired geometry. Since the CFD simulation would only consider the pitching movement, only one half of the axial-symmetric launcher is enough. In this project, we will keep the left-hand side (w.r.t the flight direction) of the model. Geometrical size of the domain according to discussion shown in

Section 3.2.3. The example of the supersonic computational domain is illustrated in Figure 3.5.

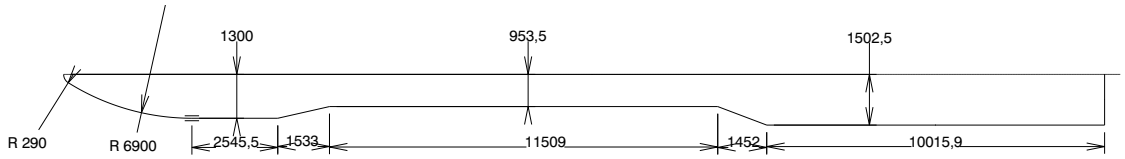


Figure 3.4: Geometry of launcher VEGA (in mm, Perez (2006))

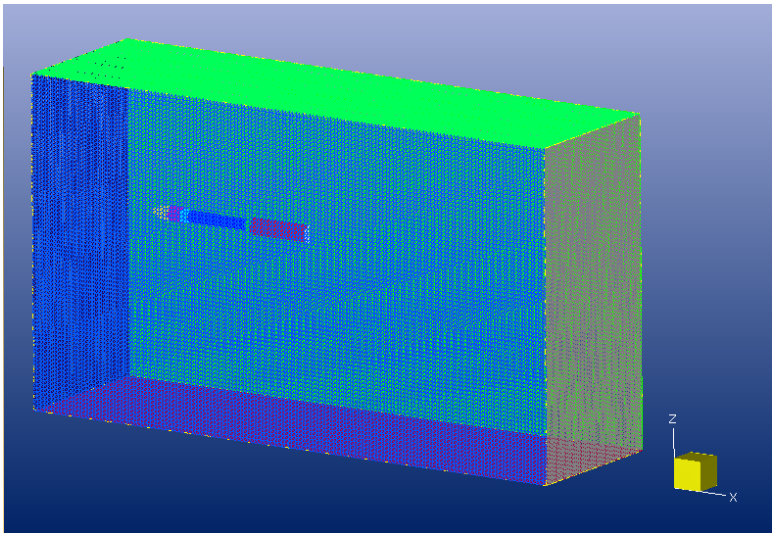


Figure 3.5: Computational domain of launcher VEGA at supersonic

- `Meshing`

After the domain was successfully created and imported into *HEXPRESS*, the meshing process can be started. Before meshing, some domain manipulation can be done if needed. The manipulation contains splitting or merging of edge & faces, adding or deleting of blocks. From experience of this project, it is suggested that on the smooth nose of the launcher, extra curves on connection region between two surfaces can be deleted by surfaces merging. Because near such curves, the cells have to be reallocated during the meshing process, which tends to cause divergence during simulation. One example is illustrated in Figure 3.6, where the red dash line is suggested to be deleted by surface merge. The whole meshing process can be divided into five sub-steps: initial mesh, adapt to geometry, snap to geometry, optimization and viscous layer insert. Corresponding illustration is shown in Figure 3.7.

The initial mesh step gives a rough meshing in each axis. The divisions along each cartesian axis are automatically suggested by the software. But it is suggested to set by the user, since it directly affects the final cell quantity. Because all the further meshing step is based on the initial mesh result, and the suggested values are generally not suitable for launchers which

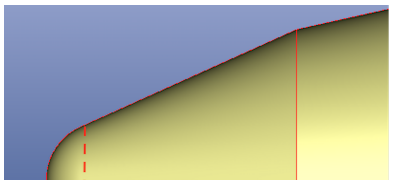


Figure 3.6: Recommended surface merging before meshing

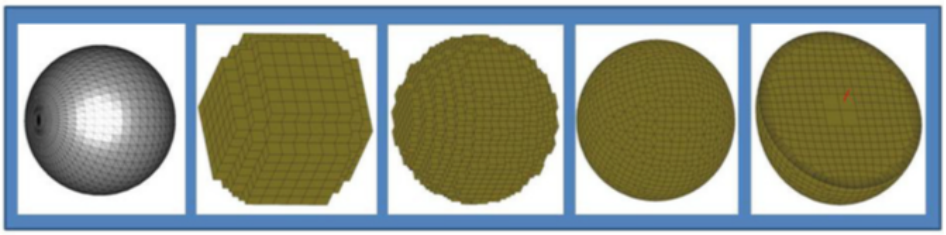


Figure 3.7: Example of meshing process (NUMECA (2012))

can be longer than 30 meters. The initial mesh step can also serve as a way of refining the mesh globally, and it is especially useful when performing the mesh independent evaluations.

Then comes to the most important step of the meshing process, the geometry adaption. In this step, curve & surface and box refinements are defined accordingly. First, the global maximum number of refinements should be selected. This value supersedes any other selected adaption criteria and limits the maximum refine times and has a direct constraint on the final cell quantity. Refinement can be implemented on both surfaces and curves. The surface refinement is applied to the whole surface, for example, the whole nose surface. The curve refinement is exerted on a certain curve, like sharp edge etc. For regions where significant flow compression happens, like the nose surface and the flare of the first stage, surface refinement should be done intensively. For curves where kinks emerge, the refinement should be reinforced by adopting higher refinement level than that of the surface refinement.

There are three types of adaption criteria available, the distance, curvature and target cell size criteria. The aim of the distance criterion is to allow enough refinements of the cells in order to fill any gaps between very close surfaces with a sufficient amount of cells. The aim of the curvature criterion is compute suitable cell size for resolving the curved surface. The last one, target size criterion is the most straightforward and it specify exact cell size to each refinement. From experience of this project, the target cell size criterion should be most used, since it gives a better control of the refinement than any other two. After both surface and curve refinement have been performed, box refinement should be added, to better discretize the space around the launcher. These regions contain backward facing steps region after the payload stage and the region around the base of the first stage (In this project, the nozzle is not taken into account.). Without proper resolving, high pressure, temperature or turbulence viscosity gradient can happen in such regions, which leads to computational divergence. NUMECA seems to have a higher request on the meshing file than other CFD tools like $FLUENT^{TM}$ and effort should be made on such local refinements. Sometimes, ill-discretization in one spot can cause divergence in other places and it is difficult to locate the

region needing further refinement. So it is strongly suggested to make proper refinement to regions where large gradients may happen. Besides suggestions mentioned above, one useful tip is to take good advantage of the Refinement diffusion, which controls cell region that should be refined. This parameter is available under all kinds of refinement and the default value is 2. In some scenarios where further refinement is required, increasing the refinement level may lead to large increasing in cell numbers, which can be avoided by adjusting corresponding refinement diffusion value higher. This process is explained in Figure 3.8, where only two cells over the object surface will be refined from the default value. Generally, to what extent the refinement should be is based on trial-and-error, just like the viscous layer inserting, which is to be discussed hereinafter.

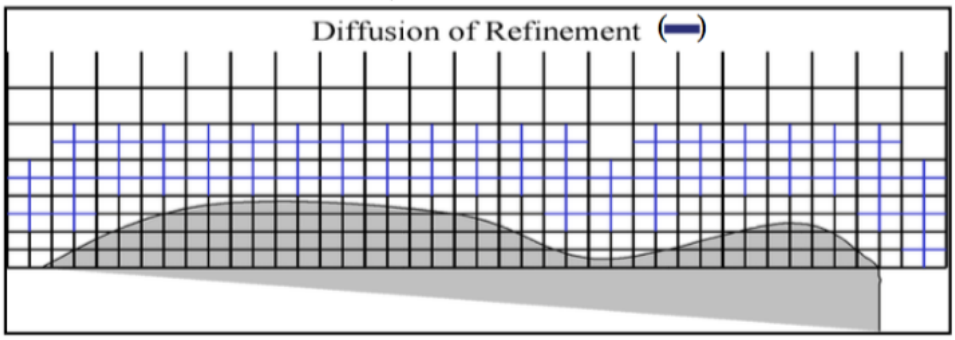


Figure 3.8: Diffusion of refinement from surface intersecting cells (NUMECA (2012))

The following steps are cell snapping and optimization. In both steps, the default settings are adequate and no extra adjustment is needed. The last meshing step is the insertion of the viscous layer. This step is critical for viscous flow simulation, especially where separation occurs. As discussed in Section 3.2.4, the y^+ in this project should be set below 5 to capture the viscosity effect correctly. Based on the y^+, the first layer thickness can be roughly computed, see Equation 3.4. Another important parameter is the stretching ratio, which specifies the inflation rate of the viscous layers. When all the parameters are chosen, the meshing can be executed.

After the meshing, the mesh quality must be checked, as shown in Figure 3.9. The histogram presenting the cell quality distribution for the various quality criterion. The most import ones are the quantity of the concave cells, the cell aspect ratio and the mesh expansion ratio. In NUMECA, the recommended aspect ratio for inviscid-flow simulation is 5 and can be extended to 2000 under viscous flow. The recommended value for expansion ratio is 5. For both criteria, a larger value is allowable. But it is strongly suggested they are controlled to certain low values, or else afflicting divergences can happen during the simulation process and will take you a great deal of time to locate the error.

- `Solver settings`
  - `Physical configuration`

    In the physical configuration, the fluid model and flow model will be defined. For the former one, the perfect gas is chosen for this project. The characters of the free

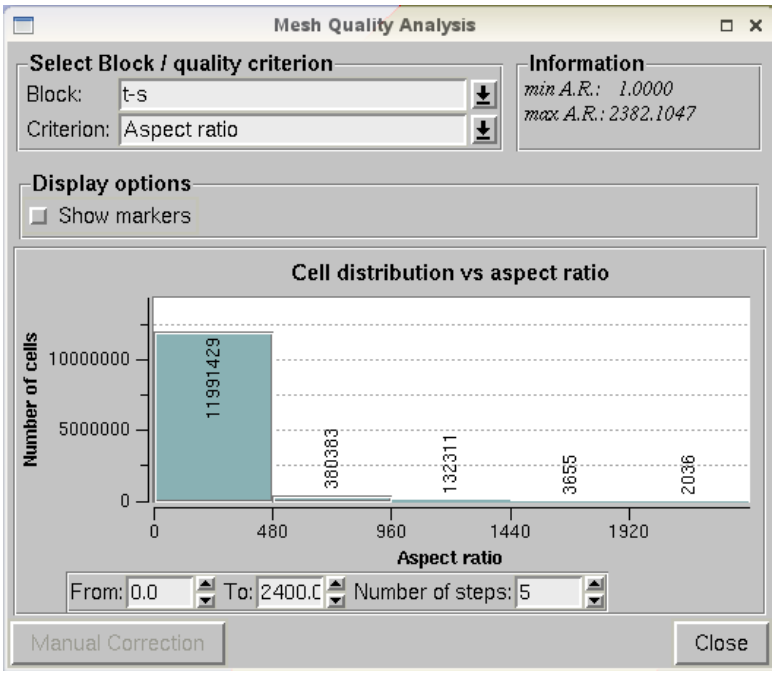


Figure 3.9: Meshing quality checking (Screen shot from NUMECA)

stream need to be defined are the static pressure, static temperature and dynamic viscosity. In this project, all these parameters can be achieved from the VEGA flight profile, see Figure 5.2. The density is automatically computed based on the obligatory parameters. This value can be used as double-check if the fluid properties are defined consistently. Then the flow model needs to be chosen. As discussed in Section 3.2.2, the Spalart-Allmaras extended wall function turbulence model is chosen for this project. In this step, when input the reference length and velocity, the Reynolds number can be computed automatically, which can also be used as double-check. After that, the boundary condition, in this project the external flow properties should be set. The values are the same as that of the fluid property.

The only property needs to specify in the Spalart-Allmaras model is the turbulence viscosity. In this model, a parameter called Spalart viscosity $\tilde{\nu}$ is needed. The ratio $\tilde{\nu}_t/\nu$ is equivalent to the ratio between the turbulent viscosity and the kinematic viscosity μ_t/ν, for values higher or equal to 1, and for external flow the value is suggested to set to 1, (NUMECA (2012)). With the relation mentioned above, the turbulence viscosity can be estimated and input into the boundary conditions. For the initial solution setting, one can set it to the free flow constant values or set it to former results from files such as results based on the first-order discretization.

- `Numerical schemes`

For the numerical schemes, several aspects need to be set. These contain the CPU booster scheme setting, the multi-grid parameters, the spacial discretization, the CFL number and so on.

$FINE^{TM}/Open$ includes an innovative technique which helps to achieve the conver-

gence. It reduces the computation time significantly. If this item is checked, the default CFL number will be set to 1000. In this project, all the computation cases adopt the CPU booster mode and the corresponding CFL is set to below 50 to ensure the convergence. From the author's experience, for the first-order spacial discretization, the CFL can be set to around 30 and for the second-order, the value is suggested to be not higher than 20. Generally, with Mach increasing, the CFL should be reduced accordingly. In NUMECA, the multi-grid method is another scheme to accelerate convergent speed. it is said to be better than the Runge-Kutta strategy (NUMECA (2012)). it is recommended to select more than 4 grid levels for the multi-grid strategy.

For the spacial discretization, the first-order is more robust than the second-order. For the subsonic and transonic region, one can choose the second-order accuracy directly. As the Ma and AoA rising, the convergence is hard to achieve and it is suggested to apply the first-order scheme to get a rough simulation result and then based on that start the simulation with the second-order discretization. With respect to the discretization scheme, the choice depends. Generally, the Roe upwind scheme is harder to converge than the central scheme. But it does not always happen. For some cases where Central scheme fails to give convergence, however, Roe upwind can make it. Even sounds weird, it is suggested to change one scheme to the other when divergence happens. Note that, the Roe upwind scheme does not suitable for high-Ma simulation for the potential of divergence.

Then comes to the topic of mesh adaption. Generally, finer grids lead to more accurate results than the coarser ones. Before the simulation, the flow features in the interested region is unknown. To capture the flow phenomena by way of using uniform finer grid is computationally expensive. A better way is using the mesh adoption strategy. First, mesh the target region with coarser grid and then start the simulation. Then based on the preliminary result, regions where large gradients (with respect to density, velocity, pressure and temperature) occur will be flagged to be refined. By adopting this scheme, the refinement is more efficient. However, even though the scheme can reduce the growth of the grid number to a large extent, one mesh adaption step will get the mesh size doubled in NUMECA, which is not suitable for this project, since the cell number is already very high. Remedy to this can be adding refine boxes to regions where large gradient can happen in advance.

3.2.6 Mesh independence study

Before taking the simulation result from CFD as acceptable, the mesh independence study should be carried out. This step is critical to ensure the accuracy of the simulations. A converged solution is not necessarily a correct one. For some scenarios, even convergence is achieved, the result can still be misleading. The mesh independence study should at least be carried out once for each type of simulation.

A model can be considered converged if 3 criteria have been met. These criteria include residual targets, monitor invariance, and mass balance. For the residual root mean square value, it should be reduced to an acceptable degree, the most used standard is 10^{-4} or 10^{-5}.

And the target values in the monitor window should reach a steady state. In terms of the imbalance, the species imbalance should be lower than 1%.

In NUMECA, from the user's manual, it is suggested to add approximately 9 nodes across a free shear-layer and approximately 15 across a boundary layer to get a grid-independent result for turbulent flow. In cases where the wall function is adopted, only 9 nodes are enough. it is highly suggested that those steps are performed so as to save time in the mesh independent study.

To perform the study, the simulator should at least build three meshes with different refinements. From coarser to finer, the cell number ratio should be at least 1.5 (for unstructured meshing, best ratio is 3), which is also suggested by Levy et al. (2014). As in this project, the viscous sublayer should be caught, the y^+ value is set to below 5, the total cell number is huge. So the cell-increasing ratio in this study will take 1.5. For each mesh, the simulation should be carried out and the best way to check the mesh independence is to plot a graph of the target values v.s. the cell numbers. In this case, the target values include C_A, C_N, C_M, C_p, and C_f. The mesh should be refined until the grid-independent result is achieved. Corresponding results will be further shown in Chapter 4 (for Taurus) and Chapter 5 (for VEGA).

In the mesh independent study, some procedures can be applied to accelerate the convergence. Setting a good initial condition is quite important for achieving a faster convergence. For cases where Ma and AoA have large values, it is not easy to get a converge. it is suggested to use the first-order discretization scheme to set up the flow field and based on that start the second-order simulation. Another tip is that, as simulation goes on, the Courant number or CFL can be increased gradually to accelerate the convergence. In terms of the cell size, a uniform mesh usually has a lower truncation error, and we should try to keep the cell-size variance to a certain extent. At last, the aspect ratio, expansion ratio and skewness should be near the suggested value. With those tips, the time-consuming mesh independence study can be finished with high efficiency.

3.3 Data fusion methodology

As discussed in the literature review in Chapter 2, there are many methods to perform correlation and interpolation. However, only a few of them can handle multi-fidelity data source. The most suitable method for this project is the Kriging & Co-Kriging. The former one can handle single-fidelity data source and the latter one which is based on Kriging, is able to handle multi-fidelity data. In the following paragraphs, the Kriging and Co-Kriging methods will be discussed in detail.

3.3.1 Kriging process

Kriging is a statistical interpolation method suggested by Kbiob (1951) and mathematically formulated by Matheron (1963). Kriging estimation depends on the spatial correlations between the given sample points to be interpolated. It is mostly depicted as a way of 'modeling the function as a realization of a stochastic process'. The Kriging algorithm to be discussed in this report is based on Jones (2001). The standard derivation can be found in Sacks et al. (1989). Compared with other basis function methods, the Kriging stands out for the characteristics of *tunable*, thanks to its statistical interpolation. This kind of interpolation can not only provide the *predictor* but also allow us to measure the possible error of the predictor.

Suppose we are trying to make a prediction on point $\mathbf{X}$ in one domain. For two random points x_i and x_j, when assuming the target function is continuous and the distance between x_i and x_j, $||x_i - x_j||$, is small enough, we can model this statistically by saying the corresponding $Y(x_i)$ and $Y(x_i)$ is highly correlated. The correlation between $Y(x_i)$ and $Y(x_i)$ is defined in

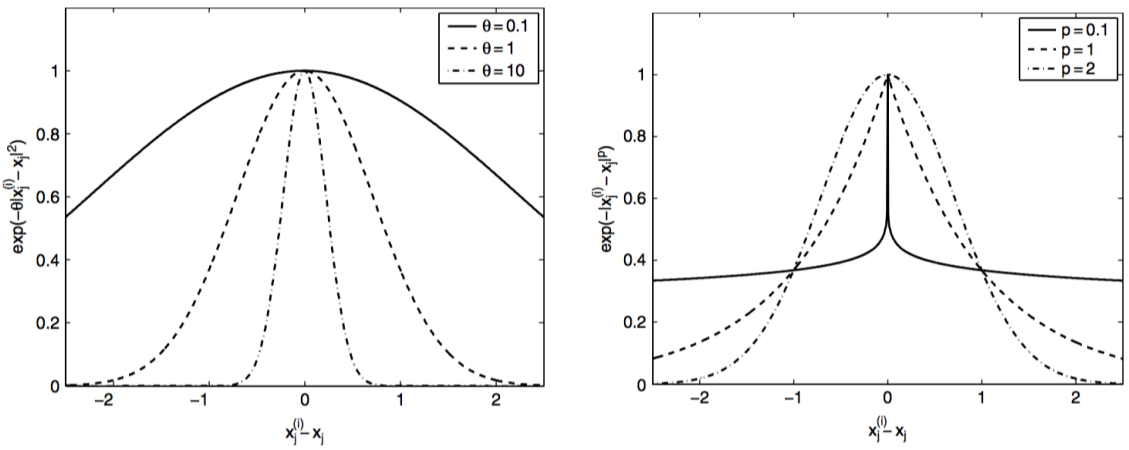
$$Corr[Y(x_i), Y(x_j)] = exp(-\sum_{l=1}^{d} \theta_l |x_{i_l} - x_{j_l}|^{p_l}). \tag{3.8}$$

In the above equation, if $x_i = x_j$, the correlation is 1 and if the distance between those two points is infinite, the correlation is zero, which is intuitive. In this equation, the parameter $\boldsymbol{\theta}$ means how fast the correlation decreases when these two points move away from each other in the l^{th} dimension. The effect of $\boldsymbol{\theta}$ is illustrated in Figure 3.10(a). The bigger value implies even small change in this direction can lead to a large change in the correlation. The parameter $\boldsymbol{p}$ describes the smoothness of the function in the l^{th} dimension, which is illustrated in Figure 3.10(b). A larger value indicates smoother. Most chosen value for this parameter is 2.

The Kriging model treats the observed responses as if they're from a stochastic process. Even though in the computational simulation, the responses are deterministic, we should begin with this concept. Based on this concept, the uncertainty about the responses can be represented in vector as $(Y(x_1) \dots Y(x_n))^T$. The vector has a mean of $\mathbf{1}\mu$, where the $\mathbf{1}$ is a $n \times 1$ vector and n is the quantity of the responds ($\boldsymbol{Y}$). The covariance of the vector equals to Cov(Y)=$\sigma^2\boldsymbol{R}$, where σ^2 is the variance of the normally distributed $\boldsymbol{Y(X)}$ and the $\boldsymbol{R}$ is a $n \times n$ vector, shown below:

$$\boldsymbol{R} = \begin{pmatrix} Corr[Y(x_1), Y(x_1)] & \dots & Corr[Y(x_1), Y(x_n)] \\ \vdots & \dots & \vdots \\ Corr[Y(x_n), Y(x_1)] & \dots & Corr[Y(x_n), Y(x_n)] \end{pmatrix}. \tag{3.9}$$

The distribution of $\boldsymbol{Y}$ depends on μ(the mean of the normally distributed $\boldsymbol{Y(X)}$), σ^2, θ_l and



(a) Effect of $\boldsymbol{\theta}$ (b) Effect of $\boldsymbol{p}$

Figure 3.10: Tunable parameters in Kriging (Forrester et al. (2008))

p_l (l=1,..., d where d is the dimension of $\boldsymbol{X}$). To identify the parameters, $\mu, \sigma^2, \boldsymbol{\theta}$ and $\boldsymbol{p}$, we maximize the likelihood function of the observed data , $(y_1 \dots y_n)^T$, shown below:

$$Likelihood\ function = \frac{1}{(2\pi)^{\frac{n}{2}}(\sigma^2)^{\frac{n}{2}}|\mathbf{R}|^{\frac{1}{2}}} exp[\frac{-(y-\mathbf{1}\mu)^T\mathbf{R}^{-1}(y-\mathbf{1}\mu)}{2\sigma^2}]. \tag{3.10}$$

By maximizing the likelihood function, it means we want the target function has the similar character as that seen from the observed response. An easy way to maximize the likelihood is to evaluate its log function, after ignoring the constant part, the log likelihood is shown as:

$$Log\ likelihood\ function = -\frac{n}{2}log(\sigma^2) - \frac{1}{2}log(|\mathbf{R}|) - \frac{(y-\mathbf{1}\mu)^T\mathbf{R}^{-1}(y-\mathbf{1}\mu)}{2\sigma^2}. \tag{3.11}$$

We can achieve corresponding σ^2 and μ by setting the first order derivatives of the log likelihood (w.r.t. σ^2 and μ) to zero. The results are

$$\begin{aligned} \hat{\mu} &= \frac{\mathbf{1}^T\boldsymbol{R}^{-1}y}{\mathbf{1}^T\boldsymbol{R}^{-1}\mathbf{1}}, \\ \hat{\sigma^2} &= \frac{(y-\mathbf{1}\hat{\mu})^T\boldsymbol{R}^{-1}(y-\mathbf{1}\hat{\mu})}{n}, \end{aligned} \tag{3.12}$$

where the circumflex symbol ˆ means the maximum likelihood estimation (MLE). Substituting

these two parameters to Equation 3.11, we get the so-called 'concentrated log-likelihood' function. Again, ignoring the constant part, we have $\frac{n}{2}log(\hat{\sigma}^2) - \frac{1}{2}log(|\boldsymbol{R}|)$.

The concentrate function is only about $\boldsymbol{R}$, which is a function of θs and ps. The θs and ps can be estimated by maximizing the concentration function. With the $\hat{\boldsymbol{\theta}}$ and $\hat{\boldsymbol{p}}$, we can compute $\hat{\mu}$ and $\hat{\sigma}^2$ from Equation 3.12.

To estimate $y^*(x)$ at some new point x^*, we should add (x^*,y^*) to the dataset as the $(n+1)^{th}$ observation and compute the 'augmented' likelihood function. This time the new likelihood is a function of y^* and shows the consistency of (x^*,y^*) with the formerly observed pattern of variation. The new vector of function value is $\tilde{y} = (y^T, y^*)^T$ and the new correlation matrix $\tilde{\mathbf{R}}$ turns to the form of $\tilde{\mathbf{R}} = \begin{pmatrix} \mathbf{R} & \mathbf{r} \\ \mathbf{r}^T & 1 \end{pmatrix}$, where the vector correlation $\mathbf{r}$ is correlation of $Y(x^*)$ with the former $Y(x_i)$ $(i = 1, \ldots, n)$, shown below:

$$\mathbf{r} = \begin{pmatrix} Corr[Y(x^*), Y(x_1)] \\ \vdots \\ Corr[Y(x^*), Y(x_n)] \end{pmatrix} \tag{3.13}$$

As in Equation 3.11, only part of the augmented log-likelihood function, $-\frac{(\tilde{y}-\mathbf{1}\hat{\mu})^T \tilde{\mathbf{R}}^{-1}(\tilde{y}-\mathbf{1}\hat{\mu})}{2\hat{\sigma}^2}$, is dependent on y^*. Substituting the $\tilde{y}$ and $\tilde{\mathbf{R}}$, the equation turns into the form

$$Augmented\ log\ likelihood\ function = \frac{-\begin{pmatrix} y-\mathbf{1}\hat{\mu} \\ y^*-\hat{\mu} \end{pmatrix}^T \begin{pmatrix} \mathbf{R} & \mathbf{r} \\ \mathbf{r}^T & 1 \end{pmatrix}^{-1} \begin{pmatrix} y-\mathbf{1}\hat{\mu} \\ y^*-\hat{\mu} \end{pmatrix}}{2\hat{\sigma}^2} \tag{3.14}$$

With the partition inverse formula by Theil and Theil (1971), the $\tilde{\mathbf{R}}^{-1}$ can be expanded in the form

$$\tilde{\mathbf{R}}^{-1} = \left(\begin{array}{c|c} \mathbf{R}^{-1} + \mathbf{R}^{-1}\mathbf{r}(1-\mathbf{r}^T\mathbf{R}^{-1}\mathbf{r})^{-1}\mathbf{r}^T\mathbf{R}^{-1} & -\mathbf{R}^{-1}\mathbf{r}(1-\mathbf{r}^T\mathbf{R}^{-1}\mathbf{r})^{-1} \\ \hline -(1-\mathbf{r}^T\mathbf{R}^{-1}\mathbf{r})^{-1}\mathbf{r}^T\mathbf{R}^{-1} & (1-\mathbf{r}^T\mathbf{R}^{-1}\mathbf{r})^{-1} \end{array} \right). \tag{3.15}$$

Substitute this to Equation 3.14, we further rewrite the augmented likelihood function to the form

$$Augmented\ log\ likelihood\ function = [\frac{-1}{2\hat{\sigma}^2(1-\mathbf{r}^T\mathbf{R}^{-1}\mathbf{r})}](y^*-\hat{\mu})^2 + [\frac{\mathbf{r}^T\mathbf{R}^{-1}(y-\mathbf{1}\hat{\mu})}{\hat{\sigma}^2(1-\mathbf{r}^T\mathbf{R}^{-1}\mathbf{r})}](y^*-\hat{\mu}),$$

(3.16)

which only consisting the part with $y*$. Then the first order derivative w.r.t. y^* should be set to zero to achieve the maximum likelihood. After that, we get

$$[\frac{-1}{2\hat{\sigma}^2(1-\mathbf{r}^T\mathbf{R}^{-1}\mathbf{r})}](y^*-\hat{\mu})+[\frac{\mathbf{r}^T\mathbf{R}^{-1}(y-\mathbf{1}\hat{\mu})}{\hat{\sigma}^2(1-\mathbf{r}^T\mathbf{R}^{-1}\mathbf{r})}]=0, \tag{3.17}$$

from which the y^* can be solved.

Now we get the standard form of the Kriging predictor

$$\hat{y}(x^*)=\hat{\mu}+\mathbf{r}^T\mathbf{R}^{-1}(y-\mathbf{1}\hat{\mu}). \tag{3.18}$$

When setting $a=\hat{\mu}$ and b_i as the i^{th} element of $\mathbf{R}^{-1}(y-\mathbf{1}\hat{\mu})$ and setting $\phi(x^*-x_i)$ as the i^{th} element of $\mathbf{r}^T(Corr[Y(x^*),Y(x_i)]$, expansion of which see Equation 3.8), the Kriging predictor can be transformed into

$$\hat{y}(x^*)=a+\sum_{i=1}^{n}b_i\phi(x^*-x_i). \tag{3.19}$$

With this form, we find that the Kriging predictor is essentially a linear combination of basic functions and polynomials (in this equation, a is a constant).

The mean square error of the Kriging predictor, denoted by $s^2(x^*)$, can be derived by way of a standard stochastic-process method. The result is given directly as below:

$$s^2(x^*)=\hat{\sigma}^2[1-\mathbf{r}^T\mathbf{R}^{-1}\mathbf{r}+\frac{(1-\mathbf{r}^T\mathbf{R}^{-1}\mathbf{r})^2}{\mathbf{1}^T\mathbf{R}^{-1}\mathbf{1}}]. \tag{3.20}$$

In this value, the former part $\hat{\sigma}^2(1-\mathbf{r}^T\mathbf{R}^{-1}\mathbf{r})$ is the reciprocal of the second-order derivative of the likelihood function (w.r.t. y^*). The larger value of the derivative means the likelihood function has a larger curvature at corresponding y^*. In Figure 3.11, the high-curvature line means a lower potential error, which is intuitive. The latter part in Equation 3.20 can be explained as the error caused by the uncertainty of the μ.

As said in the beginning of this section, the Kriging outperforms other basis function methods in terms of prediction accuracy. That is because only the Kriging has tunable parameters, θs and ps included in the estimating process.

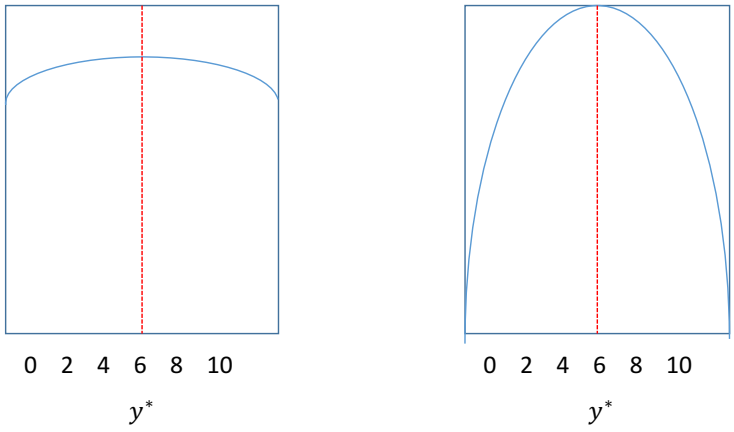


Figure 3.11: Augmented log likelihood function at guessed y^*. (Jones (2001)) (a) With low confidence (b) With high confidence.

3.3.2 Co-Kriging process

In reality, there will be more than one kind of information available in the engineering problems. The information can have various fidelities, which can come from empirical estimations, finite element analysis, full-scale simulation or pure experimental test. For these various data source, corresponding accuracy diverse with each other. Generally, the low-fidelity data tends to be easy to obtain and is called cheap data. The opposite is named expensive data. Both kinds of data are useful to some extent. Ideally, we wish to use more cheap and less expensive data to figure out the target problems (or functions). In this section, we will discuss the Co-Kriging algorithm, which can handle multi-fidelity data. The Co-Kriging is essentially based on Kriging. And it can be interpreted as performing two or more times of Kriging. The following derivation is concentrated on two-dataset Co-Kriging, for its relative simplicity. The derivation is based on work of Forrester et al. (2007) and Forrester et al. (2008). The latter provides a more detailed process.

We denote the cheap data $\boldsymbol{Y_c}$ from sample points $\boldsymbol{X_c}$ and expensive data $\boldsymbol{Y_e}$ at points $\boldsymbol{X_e}$, where we assume the cheap points cover the expensive points ($\boldsymbol{X_e} \subset \boldsymbol{X_c}$). First, we concatenate the two type of data, $\boldsymbol{X} = (\boldsymbol{X_c}, \boldsymbol{X_e})^T = (X_c^{(1)}, \ldots, X_c^{(n_c)}, X_e^{(1)}, \ldots, X_e^{(n_e)})^T$ and $\boldsymbol{Y} = (Y_c(\boldsymbol{X_c}), Y_e(\boldsymbol{X_e}))^T = (Y_c(X_c^{(1)}), \ldots, Y_c(X_c^{(n_c)}), Y_e(X_e^{(1)}), \ldots, Y_e(X_e^{(n_e)}))^T$. Before start of the derivation, we need to declare the important assumption from Kennedy and O'Hagan (2000), $cov\{Y_e(X^i), Y_c(\boldsymbol{X})|Y_c(X^i)\} = 0, \forall X \neq X^{(i)}$, which is also called Auto-regression assumption and it means *no more can be learnt about* $Y_e(X^i)$ *from the cheaper code if the value of the expensive function at* $X^{(i)}$ *is known.* This is also known as the *Markov property* and can be interpreted as all the inaccuracy lies in the cheaper data, ie. all the expensive data is correct.

Let $\boldsymbol{Z_c}()$ and $\boldsymbol{Z_e}()$ stand for the local features at $\boldsymbol{X_c}$ and $\boldsymbol{X_e}$ respectively. By using the Auto-regression assumption, we actually approximate the expensive response as a combination of

cheap response multiplied a coefficient and an error part, as shown below:

$$\boldsymbol{Z_e}(\boldsymbol{x}) = \rho \boldsymbol{Z_c}(\boldsymbol{x}) + \boldsymbol{Z_d}(\boldsymbol{x}), \tag{3.21}$$

where ρ is a scaling factor. Similar with the Kriging covariance matrix (like Cov($\boldsymbol{Y}$)=$\sigma^2\mathbf{R}$, where the $\mathbf{R}$ is an $n \times n$ vector, see Equation 3.9), we have the covariance matrix

$$\begin{aligned}
Cov\{\boldsymbol{Y_c}(\boldsymbol{X_c}), \boldsymbol{Y_c}(\boldsymbol{X_c})\} &= Cov\{\boldsymbol{Z_c}(\boldsymbol{X_c}), \boldsymbol{Z_c}(\boldsymbol{X_c})\} \\
&= \sigma_c^2 \boldsymbol{R_c}(\boldsymbol{X_c}, \boldsymbol{X_c}) \\
Cov\{\boldsymbol{Y_e}(\boldsymbol{X_e}), \boldsymbol{Y_e}(\boldsymbol{X_c})\} &= Cov\{\rho \boldsymbol{Z_c}(\boldsymbol{X_c}) + \boldsymbol{Z_d}(\boldsymbol{X_c}), \boldsymbol{Z_c}(\boldsymbol{X_e})\} \\
&= \rho\sigma_c^2 \boldsymbol{R_c}(\boldsymbol{X_c}, \boldsymbol{X_e}) \\
Cov\{Y\boldsymbol{Y_e}(\boldsymbol{X_e}), \boldsymbol{Y_e}(\boldsymbol{X_e})\} &= Cov\{\rho \boldsymbol{Z_c}(\boldsymbol{X_e}) + \boldsymbol{Z_d}(\boldsymbol{X_e}), \rho\boldsymbol{Z_c}(\boldsymbol{X_e}) + \boldsymbol{Z_d}(\boldsymbol{X_e})\} \\
&= \rho^2\sigma_c^2 \boldsymbol{R_c}(\boldsymbol{X_e}, \boldsymbol{X_e}) + \sigma_d^2 \boldsymbol{R_d}(\boldsymbol{X_e}, \boldsymbol{X_e})
\end{aligned} \tag{3.22}$$

where $\boldsymbol{R_c}(\boldsymbol{X_c}, \boldsymbol{X_e})$ stands for correlation in form of $\boldsymbol{R_c}$, between $\boldsymbol{X_e}$ and $\boldsymbol{X_c}$. The complete covariance matrix is given as:

$$\boldsymbol{C}(ov) = \left(\begin{array}{c|c} \sigma_c^2 \boldsymbol{R_c}(\boldsymbol{X_c}, \boldsymbol{X_c}) & \rho\sigma_c^2 \boldsymbol{R_c}(\boldsymbol{X_c}, \boldsymbol{X_e}) \\ \hline \rho\sigma_c^2 \boldsymbol{R_c}(\boldsymbol{X_e}, \boldsymbol{X_c}) & \rho^2\sigma_c^2 \boldsymbol{R_c}(\boldsymbol{X_e}, \boldsymbol{X_e}) + \sigma_d^2 \boldsymbol{R_d}(\boldsymbol{X_e}, \boldsymbol{X_e}) \end{array} \right) \tag{3.23}$$

The correlations are in the form of Equation 3.8. For the Co-Kriging, there are more parameters to determine than the Kriging case, since there are two kinds of correlation $\boldsymbol{R_c}$ and $\boldsymbol{R_d}$.

Since the cheap data is independent of the expensive data, to determine the μ_c, σ_c, $\boldsymbol{\theta_c}$ and $\boldsymbol{p_c}$, we should maximize the likelihood as did in the Kriging process. The corresponding results are same as Equation 3.12. With $\hat{\mu_c}$ and $\hat{\sigma_c}$ available, the $\hat{\boldsymbol{\theta_c}}$ and $\hat{\boldsymbol{p_c}}$ can be achieved by maximizing the reduced concentrated (ln-)likelihood. For the corresponding process of the expensive data, it is a bit complicated. We define $\boldsymbol{d} = \boldsymbol{y_e} - \rho \boldsymbol{y_c}(\boldsymbol{X_e})$, and based on that, the ln-likelihood of $\boldsymbol{d}$ turns to $-\frac{n_e}{2} log(\sigma_d^2) - \frac{1}{2} log(|\boldsymbol{R_c}(\boldsymbol{X_c}, \boldsymbol{X_c})|) - \frac{(\mathbf{d}-\mathbf{1}\mu_d)^T \boldsymbol{R_d}(\boldsymbol{X_e}, \boldsymbol{X_e})^{-1}(\mathbf{d}-\mathbf{1}\mu_d)}{2\sigma_d^2}$.

Accordingly, the $\hat{\mu_d}$ and $\hat{\sigma_d}^2$ are available, shown below,

$$\begin{aligned}
\hat{\mu_d} &= \frac{\mathbf{1}^T \boldsymbol{R_d}(\boldsymbol{X_e}, \boldsymbol{X_e})^{-1}\mathbf{d}}{\mathbf{1}^T \boldsymbol{R_d}(\boldsymbol{X_e}, \boldsymbol{X_e})^{-1}\mathbf{1}}, \\
\hat{\sigma_d^2} &= \frac{(\mathbf{d} - \mathbf{1}\hat{\mu_d})^T \boldsymbol{R_d}(\boldsymbol{X_e}, \boldsymbol{X_e})^{-1}(\mathbf{d} - \mathbf{1}\hat{\mu_d})}{n_e}.
\end{aligned} \tag{3.24}$$

Then the $\hat{\boldsymbol{\theta}}_{\boldsymbol{d}}$, $\hat{\rho}$ and $\hat{\boldsymbol{p}}_{\boldsymbol{d}}$ can be achieved by maximizing the corresponding likelihood. During the process (also for the cheaper case), a suitable global searching algorithm, like the generic algorithm, should be adopted. An example of Matlab codes on this topic can be found in Forrester et al. (2008). Note that, if the dimension of $\boldsymbol{X}$ is very high, the multiple matrix inversion in the searching process can be time-consuming. For such cases, the $\boldsymbol{\theta}$ and $\boldsymbol{p}$ (both for cheap and expensive data) can use the constant value, say mostly used value 2, to reduce the time-consuming process. Only this can reduce the accuracy to some extent.

Like discussed in the Kriging process, to derive the Co-Kriging predictor, we should add the new expensive point to the observed data (concatenation of cheap and expensive), to form new 'augmented' likelihood, and try to maximize it. The essential of the step is predict value at the new expensive point by making it consistent with the observed data. The new $\boldsymbol{X}$, $\boldsymbol{Y}$ dataset is $\tilde{\boldsymbol{X}} = \{\boldsymbol{X}^T, X^*\}^T = \{\boldsymbol{X_c}^T, \boldsymbol{X_e}^T, X^*\}^T$ and $\tilde{\boldsymbol{Y}} = \{\boldsymbol{Y^T}, Y^*\}^T = \{\boldsymbol{Y_c^T}, \boldsymbol{Y_e^T}, Y^*\}^T$, respectively. The corresponding $\tilde{\boldsymbol{C}}$ is

$$\tilde{\boldsymbol{C}} = \begin{pmatrix} \hat{\sigma_c}^2 \boldsymbol{R_c}(\boldsymbol{X_c}, \boldsymbol{X_c}) & \rho\hat{\sigma_c}^2 \boldsymbol{R_c}(\boldsymbol{X_c}, \boldsymbol{X_e}) & \rho\hat{\sigma_c}^2 \boldsymbol{R_c}(\boldsymbol{X_c}, X^*) \\ \rho\hat{\sigma_c}^2 \boldsymbol{R_c}(\boldsymbol{X_e}, \boldsymbol{X_c}) & \rho\hat{\sigma_c}^2 \boldsymbol{R_c}(\boldsymbol{X_e}, \boldsymbol{X_e}) + \hat{\sigma_d}^2 \boldsymbol{R_d}(\boldsymbol{X_e}, \boldsymbol{X_e}) & (\rho^2\hat{\sigma_c}^2 + \hat{\sigma_d}^2)\boldsymbol{R_d}(\boldsymbol{X_e}, X^*) \\ \rho\hat{\sigma_c}^2 \boldsymbol{R_c}(\boldsymbol{X_c}, X^*)^T & (\rho^2\hat{\sigma_c}^2 + \hat{\sigma_d}^2)\boldsymbol{R_d}(\boldsymbol{X_e}, X^*)^T & \rho^2\hat{\sigma_c}^2 + \hat{\sigma_d}^2 \end{pmatrix}, \tag{3.25}$$

which can be reduced to

$$\tilde{\boldsymbol{C}} = \begin{pmatrix} \mathbf{C} & \mathbf{c} \\ \mathbf{c}^T & \rho^2\hat{\sigma_c}^2 + \hat{\sigma_d}^2 \end{pmatrix}, \tag{3.26}$$

where the $\mathbf{c}$ is the column vector of covariance of $\boldsymbol{X}$ and X^*. The part in the 'augmented' likelihood function that contains the expensive data is $-\frac{1}{2}(\tilde{\boldsymbol{Y}} - \mathbf{1}\mu)^T \tilde{\mathbf{C}}^{-1}(\tilde{\boldsymbol{X}} - \mathbf{1}\mu)$, and it can be expanded in the form of $-\frac{1}{2}\begin{pmatrix} \boldsymbol{Y} - \mathbf{1}\mu \\ \hat{Y}_e(X^*) - \mu \end{pmatrix}^T \begin{pmatrix} \mathbf{C} & \mathbf{c} \\ \mathbf{c}^T & \rho^2\hat{\sigma_c}^2 + \hat{\sigma_d}^2 \end{pmatrix}^{-1} \begin{pmatrix} \boldsymbol{Y} - \mathbf{1}\mu \\ \hat{Y}_e(X^*) - \mu \end{pmatrix}$.

During the expanding process, the $\tilde{\mathbf{C}}^{-1}$ can be achieved by partition inversion formula, similar with Equation 3.15. Substitute the $\tilde{\mathbf{C}}^{-1}$ into the expanded augmented likelihood function and ignore items without $\hat{Y}_e(X^*)$, we get the 'concentrated' likelihood $(\frac{-1}{\rho^2\hat{\sigma_c}^2 + \hat{\sigma_d}^2 - \mathbf{c}^T\mathbf{C}^{-1}\mathbf{c}})(\hat{Y}_e(X^*) - \hat{\mu}) + (\frac{\mathbf{c}^T\mathbf{C}^{-1}(\boldsymbol{Y} - \mathbf{1}\mu)}{\rho^2\hat{\sigma_c}^2 + \hat{\sigma_d}^2 - \mathbf{c}^T\mathbf{C}^{-1}\mathbf{c}})(\hat{y_e}(\mathbf{x}) - \hat{\mu})$.

Set the first-order derivative (w.r.t. $\hat{Y}_e(X^*)$) to zero, we can get the Co-Kriging predictor

$$\hat{Y}_e(X) = \hat{\mu} + \mathbf{c}^T\mathbf{C}^{-1}(\boldsymbol{Y} - \mathbf{1}\hat{\mu}), \tag{3.27}$$

where $\hat{\mu}$ is given as

$$\hat{\mu} = \frac{\mathbf{1}^T \boldsymbol{C}^{-1} \boldsymbol{Y}}{\mathbf{1}^T \boldsymbol{C}^{-1} \mathbf{1}}, \tag{3.28}$$

deriving from maximizing the 'concentrated' likelyhood w.r.t $\hat{\mu}$. The estimated estimation error is similar with that of Kriging process and is computed as

$$\hat{s}^2 \approx \rho^2 \hat{\sigma_c}^2 + \hat{\sigma_d}^2 - \mathbf{c}^T \mathbf{C}^{-1} \mathbf{c} + \frac{1 - \mathbf{1}^T \mathbf{C}^{-1} \mathbf{c}}{\mathbf{1} \mathbf{C}^{-1} \mathbf{1}}. \tag{3.29}$$

3.4 Sampling method

In the last section, we have discussed the mathematic theories of Kriging and Co-Kriging. In this section, we are going to give a short review on methods of sampling. Both Kriging and Co-Kriging models depend on a set of sample data to explore the design space. What data to choose and how many in total to choose strongly affect the accuracy of the exploration results. The ideal samples should cover as large surface/volume (or hypercube) of the design space as possible to get as much potential information, meanwhile, the sampling quantity should be controlled to some extent. Because on the chosen points, potentially expensive (expensive data in Co-Kriging) simulations are needed. For the DACE, the Latin hypercube sampling (LHS) method proposed by McKay et al. (2000) is most used. In this concept, to get n_p samples in a n_v variables dimension, each dimension is divided into n_p equal pieces and there is only one point (out of n_p) that falls in each of these divided pieces. However, the LHS result is based on random procedure and sometimes it can give ill-located choice, as shown in Figure 3.12. To avoid such bad result, the LHS needs to be optimized.

it is not easy to make the best sampling, since it can be a time-consuming combinatorial optimization problem. For example, to make an optimal of 20 samples with two dimensions, a number of 10^{36} combinations should be evaluated. If the dimension increased to 3, the quantity can be more than 10^{55}. Many researches have been carried out on the LHS optimization fields, popular of which are listed in Table 3.2. In terms of the time cost, the results are various. Kenny et al. (2000) reported to spend several hours on a Sun SPARC 20 workstation to generate an optimal LHS of 25 points in four dimensions. Jin et al. (2005) used much less time, several minutes to generate 100 points in 10 dimensions by using a PC with a Pentium III 650 MHZ CPU.

For the assignment in this project, there is no need to use such profound schemes to carry out such optimal LHS. The reasons are as follows. The most important dimension in this project is the Mach regime, which ranges from 0.5 to larger than 5.0. In this range, the characteristics of the target parameters are not regular. The characters near the transonic region, from

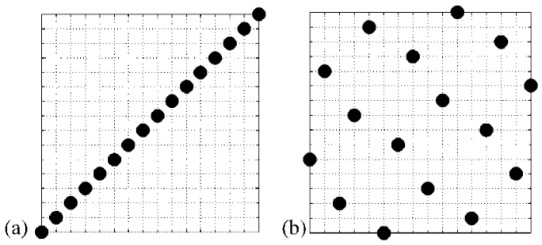


Figure 3.12: Examples of Latin hypercube designs: (a) ill-suited LHS with $n_v = 2$ and $n_p = 16$ and (b) reasonable LHS with n_v=2 and n_p=16. (Viana et al. (2010))

Table 3.2: Approaches for constructing the optimal LHS design Viana et al. (2010)

Researchers/Years	**Algorithm**	Objective functions
Audze and Eglajs 1977	Coordinates Exchange Algorithm	Potential energy
Morris and Mitchell	Simulated annealing	Φ_p criterion
Ye et al. 2000	Columnwise pairwise	Φ_p and entropy criteria
Fang et al. 2002	Threshold accepting algorithm	Centered L_2-discrepancy
Bates et al. 2004	Genetic algorithm	Potential energy
Jin et al. 2005	Enhanced stochastic evolutionary algorithm	Φ_p criterion, entropy and L_2,discrepancy
Liefvendahl and Stocki 2006	Columnwise-pairwise and, genetic algorithms	Minimum distance and, Audze Eglajs functions
van Dam et al. 2007	Branch-and-bound algorithm	1-norm and infinite norm, distances
Grosso et al. 2008	Iterated local search and simulated annealing,algorithms	Φ_p

Mach 0.9 to 1.5, change significantly. The traditional sampling method may not suitable for such scenarios. For the AoA dimension, the LHS criterion suggested by Morris and Mitchell (1995) is adequate. In such $M\&M$ method, the possibly best space-filling can be achieved by maximizing the minimum distance between any pair of points inside the sample. In Matlab, the corresponding command is *lhsdesign* and by default, the criteria *'Maximize minimum distance between points'* is applied. For future work, say when performing multi-disciplinary optimization and considering the change of the geometry, the optimal LHS strategy should be further applied and the effect of it should be evaluated.

During the improvement of data fusion process, a complementary sampling scheme can be applied besides the LHS. As discussed in Section 3.3, by using the Co-Kriging, we can get the estimated estimation error, see Equation 3.29. With this information, we can locate where the approximating results have highest estimating error, see Figure 3.13. In the figure, the red asterisks represent the relative error of the estimation. For the Ma 4-5 region, extra expensive data source is needed to reduce the inaccuracy. With the new expensive CFD data, Co-Kriging process should be carried out again. Iteration like this should be continued until certain stop criteria is met. In fact, this is also the sampling scheme in the Kriging-based optimization process.

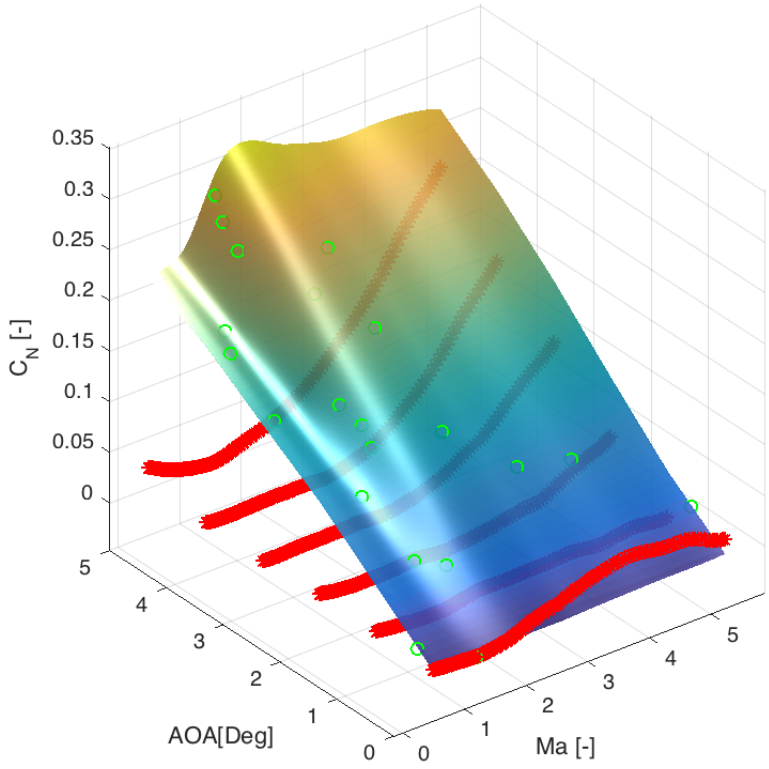


Figure 3.13: Sampling by the estimation error

Chapter 4

Validation of NUMECA

Validation of the CFD code should be performed before its result can be used. This is going to be done by comparing the CFD simulation results (C_N, C_A and C_M) with those from the WTT. Before the comparison, the grid independence tests will be carried out to ensure that the simulation results are not affected by the meshing schemes. Results from at least three meshes (with different refinements) will be compared, in terms of global data as C_A, C_N. Local data such as C_p and C_f over the launcher's surface will be considered as well.

Generally, the axial force prediction is harder than the normal force. As mentioned by Anderson Jr (2010), at least three aspects should be met to get an accurate axial force prediction. First, to get an accurate skin friction drag, an accurate calculation of the velocity gradient in the wall surface should be performed. This requires very fine grid adjacent the wall to get the flow velocity at the first few grid points above the wall surface. Second, the treatment of the turbulence directly affects the final result. As discussed in Section 3.2.2, the most popular method (for large-dimension objects) is modeling the turbulence by the RANS concept. The modeling itself can bring a great deal of uncertainty. The third aspect is the position where flow separation starts. The separation position can also be influenced by the uncertainty in the turbulence models. With these uncertainties, it is easier to understand the discrepancy in C_D in Table 4.1, which shows some very recent CFD results of a complete airplane, F-16XL, from Rizzi et al. (2009). These results are from the different state-of-the-art CFD solvers with various algorithms and turbulence models. From the table, we can see that the lift coefficient has a higher accuracy than the drag coefficient. The maximum error of the C_D is 16% (with respect to the average, meanwhile neglect the most inaccurate Investigation 3) v.s. 7% of the C_L, which indicates that axial force is hard to simulate.

With this example, we can see that the validation of NUMECA is of great importance before the results can be used for further analysis.

Table 4.1: AC discrepancy of F-16XL based on various CFD solvers ($Ma = 0.36, Re = 46.8 * 10^6, AoA = 11.85deg$ Rizzi et al. (2009))

Investigator No.	C_L	C_D
1	0.43846	0.13289
2	0.44693	0.13469
3	0.37006	0.11084
4	0.43851	0.15788
5	0.46798	0.13648
6	0.44190	0.16158
7	0.44590	0.14265

4.1 Object

The validation will be performed on the U.S. launcher, geometry of which is illustrated in Figure 4.1. Need to mention in advance that this launcher is not a perfect model to perform the validation, for its complex geometry, especially its two continuous backward-facing steps at the end of the payload stage. From experience of this project ,these two steps bring great difficulty in achieving convergence and grid-independence test during the CFD simulation. The detailed analysis will be given in the following paragraphs. Even with these disadvantage, we have to choose this launcher. Since (maybe) for some confidential reasons, this is the only traditional launcher (hammer-head and multi-stages with various diameters) whose corresponding WTT database can be found in public.

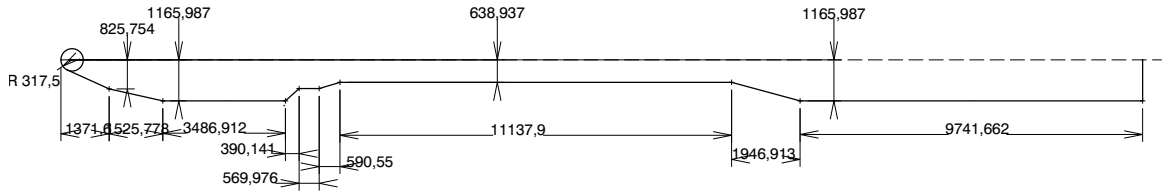


Figure 4.1: Distributions of the friction coefficient for different step heights (Unit mm, Langone and Bermüdez (2009))

Partial flight conditions of Taurus are available from the literature, and they are listed in Table 4.2. These flight conditions are converted from the corresponding environment in the wind tunnel tests. In the table, the Re is based on the diameter of the first stage, which is 2.38 meter.

Table 4.2: Flight condition of Taurus, modified from WTT Langone and Bermüdez (2009)

Mach	**P (pa)**	**T (K)**	**V (m/s)**	ρ	μ	**Re(based on D)**
0.899	81683	264	290	1.078	1.68E-05	4.43E+07
1.189	69378	246	370	0.981	1.59E-05	5.44E+07
2.41	25147	151	589	0.580	1.04E-05	7.80E+07
4.75	4970	62	744	0.279	4.16E-06	1.19E+08

4.2 Grid independence study

As discussed in Section 3.2.6, the grid independence test will be performed on several meshes with different cell sizes. From coarse, fine to more fine, etc., the cell numbers increase with a factor of about 1.5. The interested global coefficients, C_A, C_N v.s. iteration times for various meshes can be plotted to check the independence, as such is shown in Figure 4.2 for the case of Ma 1.189 (AoA 2 Deg). From the figure, we can see that when the cell number is 14.5 million (with initial mesh $16 * 16 * 16$ in the x, y, z dimension), the normal and axial force coefficients are accurate enough. When the cell number rises to 30.3 million (with initial mesh $24 * 24 * 48$), the results are almost the same. The grid independence on axial and normal forces is therefore proven.

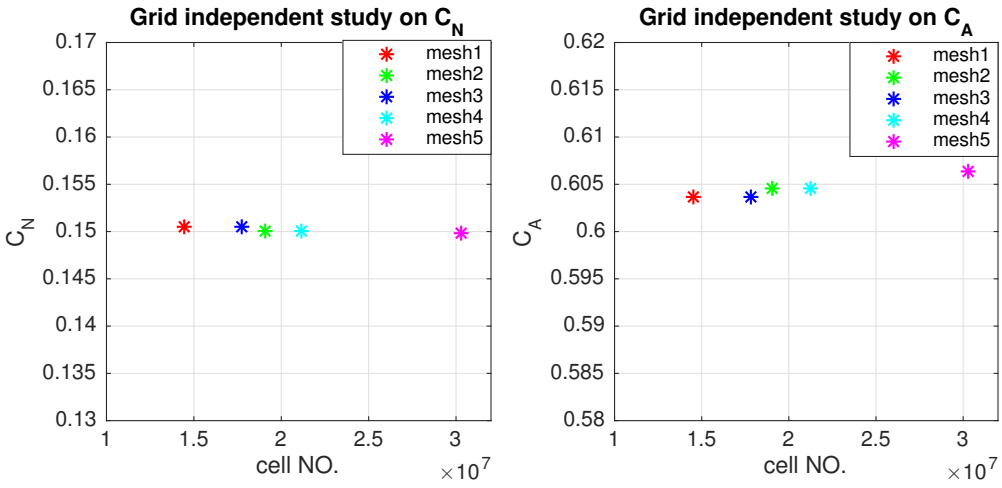


Figure 4.2: Grid independence study of axial, normal force and pitching moment coefficient, @Ma 1.189

The study mentioned in the last paragraph is about the global effect and sometimes it is not enough to prove the total grid independence. Local characteristics like C_P and C_f over surfaces of the launcher can change with various meshes, meanwhile, the global characteristics like C_N and C_A can be stable, discrepancy being canceled with each other between some surfaces. Based on this possibility, in the grid independent study, we should also check pressure coefficient and coefficient of viscosity over the surfaces. During the flight, we only consider the AoA, so we choose to evaluate AC in the region where launcher surface intersecting the symmetry plane. One of the grid independent study result, at Ma 1.189 (AoA 2 Deg), is shown in Figure 4.3. Generally, with various cell numbers, the changes in pressure ratio over the chosen points is fairly small. With respect to the coefficient of viscosity, the changes are larger, especially around the long cylinder after the payload stage and the first stage. That is because the computation of the attaching point after the backward-facing step and forward-facing step is quite sensitive to the meshing, especially in the scenario where AoA is not zero. Reasons for the relatively larger discrepancy on C_f (w.r.t. cell numbers) is explained in next section.

After the grid independence study, the final mesh is obtained, shown in Figure 4.4.

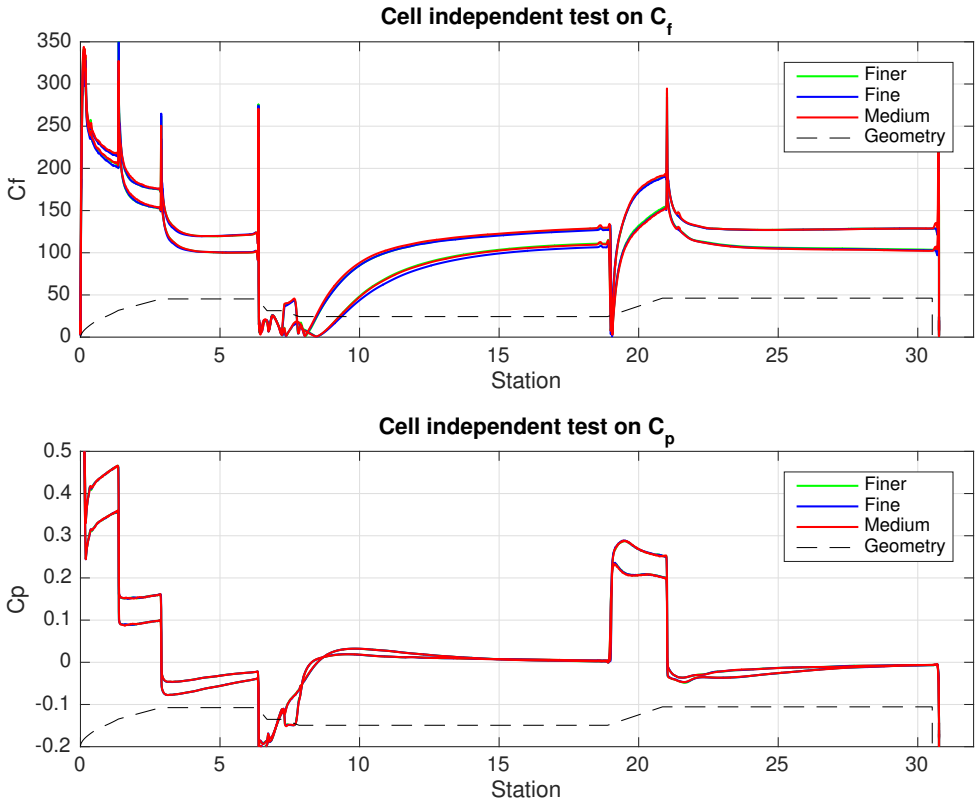
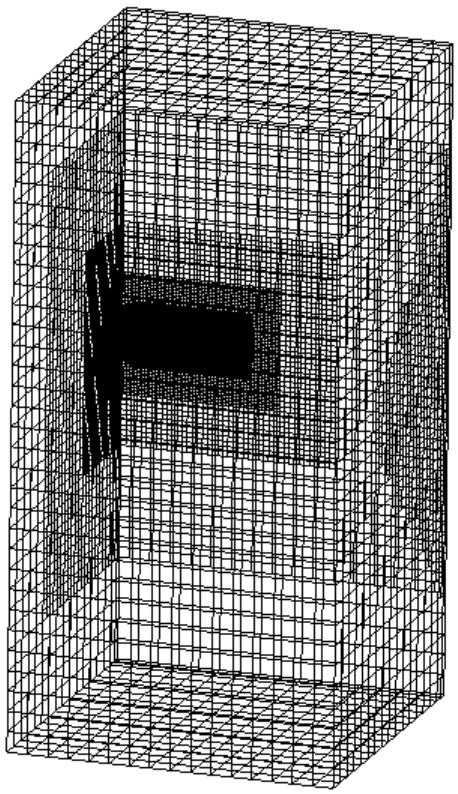
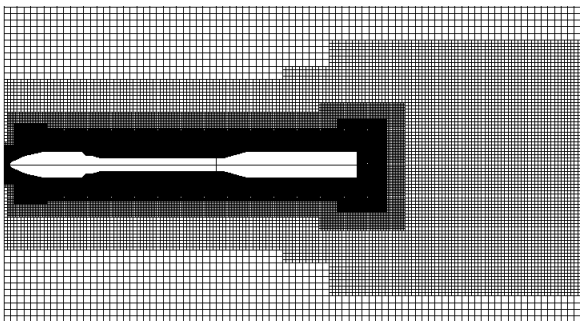


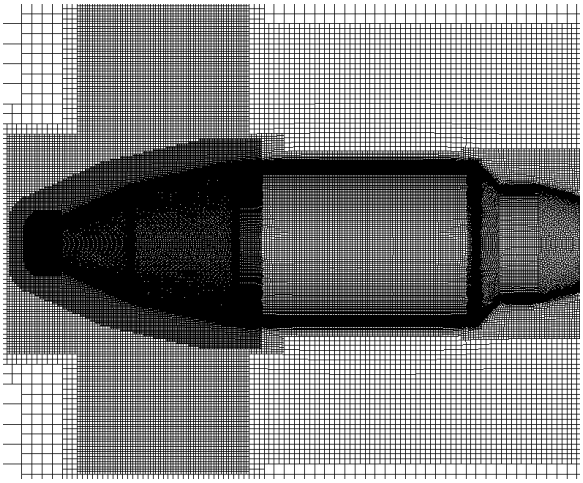
Figure 4.3: Grid-independente study on pressure and friction coefficient over the plane of symmetry of Taurus, @Ma 2.41



(a) Zoom-out view of the mesh



(b) Close-up of the mesh



(c) Local view of the mesh

Figure 4.4: Mesh of Taurus @ supersonic

4.3 Comparison of results from CFD and WTT

In this section, some flow topologies from the simulation will be shown and some of them will be compared with those from WTT. Also, the global AC from the simulations will be compared with WTT data and potential reasons for the discrepancy will be discussed.

Figure 4.5 and 4.6 gives two types of flow separation from the simulation, the shock-induced and the foci-type separation, respectively. Both flow topologies show good agreement with the oil flow visualizations shown in Figure 4.7. For the shock-induced separation, it happens at low AoA and is caused by the shock ahead of the forward-facing step. This type of separation is relatively simple. With the more complicated foci-type separation, it is from the interaction between the shock and vortex field on the cylinder ahead of the flare. This kind of separation is typical for the cylinder-flare configuration at mediate AoA (in this case 5 Deg).

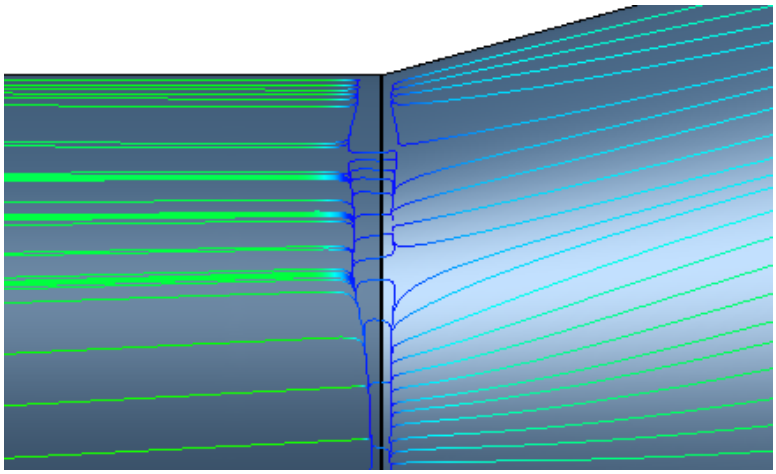


Figure 4.5: Shock-induced separation near leeward flare between first and second stages of Taurus, @Ma 2.41 and AoA 2 Deg

Table 4.3: CFD simulation results compared with wind tunnel data

Mach	WTT			CFD			ERROR		
	C_N	C_A	C_M	C_N	C_A	C_M	C_N	C_A	C_M
0.899	0.102	0.366	0.437	0.104	0.340	0.531	1%	-7%	22%
1.189	0.109	0.774	0.584	0.115	0.764	0.461	6%	2%	-21%
2.41	0.147	0.641	0.616	0.151	0.602	0.702	3%	6%	14%
4.75	0.145	0.437	0.606	0.146	0.399	0.752	1%	9%	24%

With CFD simulation results, comparison on the global AC can be made between CFD and the WTT data (Langone and Bermüdez (2009)). The result is shown in Table 4.3. In the table, we can see that both the lift and drag coefficient from the simulation have relatively good accuracy, with the maximum error lower than 10 %. But for the pitching moment coefficient, the error can be as large as 24% (for the Ma 4.75 case). The discrepancy can be explained in terms of the uncertainty in pressure coefficient C_p and friction coefficient C_f, especially in the recirculation region of the continuous double-backward-facing steps at the end of the payload stage.

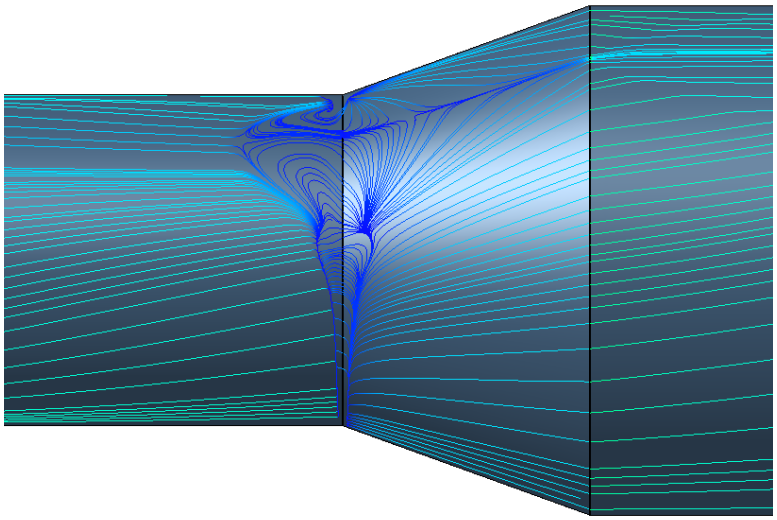


Figure 4.6: Foci-type separation near leeward flare between first and second stages of VEGA, @Ma 2.89 and AoA 5 Deg

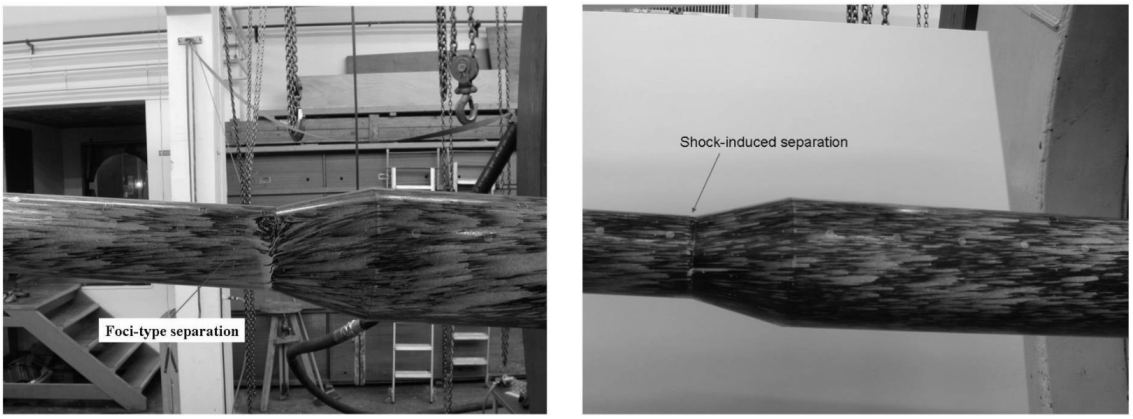


Figure 4.7: Oil flow visualizations of VEGA. Left, foci-type separation @Ma 1.58, AoA 5 Deg; Right, shock-induced separation @Ma 2.01, AoA 2 Deg (Catalano et al. (2007))

As illustrated in Figure 4.8, a single backward-facing step has a primary and a secondary recirculation regions after the sharp edge of the step. The effect of the secondary recirculation is relatively less important to the global characteristics. Figure 4.9 gives the recirculation flow near the double-backward-facing steps of Taurus at Ma 2.41 and AoA 2 degree. Each step gives rise to a recirculation region (there being no secondary recirculation region simulated on both steps in this case) and the former one interacts with the latter one. This interaction leads to more complex C_p and C_f distribution along the two continuous steps, which is shown in Figure 4.10 and Figure 4.11. In Figure 4.10, we can see the C_N changes significantly along these surfaces. In Figure 4.11, the C_f fluctuates sharply downward the flow, which is similar to that of the 2D backward-facing step (see Figure 4.12). The 3D case, C_f over Tauras given in Figure 4.13 is much more complex for the existence of the three-dimensional relieving effect.

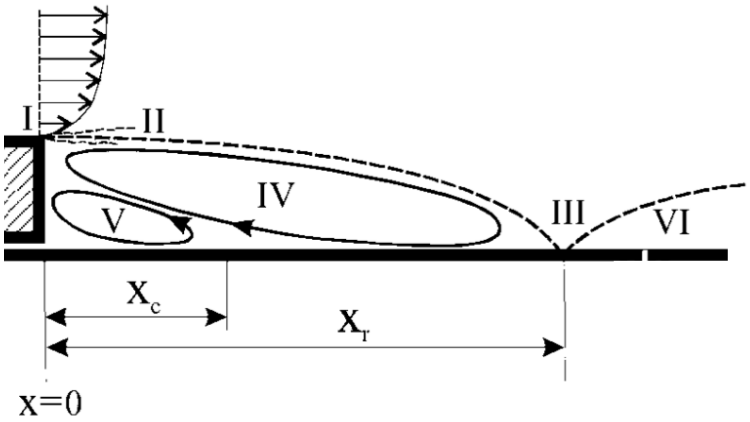


Figure 4.8: Flow pattern in the downstream of a backward-facing step (Tihon et al. (2001))

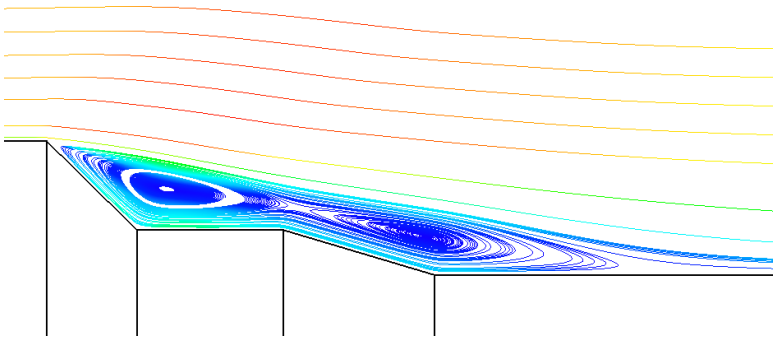


Figure 4.9: Recirculation near the double backward facing steps of Taurus, @Ma 2.41 and AoA 2 Deg

By now, there are many researches on 2D backward-facing steps, but not much attention has been paid to the 3D models. The modern CFD codes have difficulty in simulating flow over these geometries. Both the peak value of C_f and reattachment point are not easy to simulate. The simulation inaccuracy in this region may have further influence on downstream flow. All of these can be the reasons for the inaccuracy in C_M.

Figure 4.14 shows the time-dependent buffet in this region from Langone and Bermüdez (2009) and simulation result from NUMECA. By now, after several rounds of attempt (by way of refining the mesh), this error in C_M can't be reduced to the same level with that of C_N and C_A, and it is suggested in the future to find another WTT database of certain 'normal' launcher to test the accuracy on the pitching up coefficient.

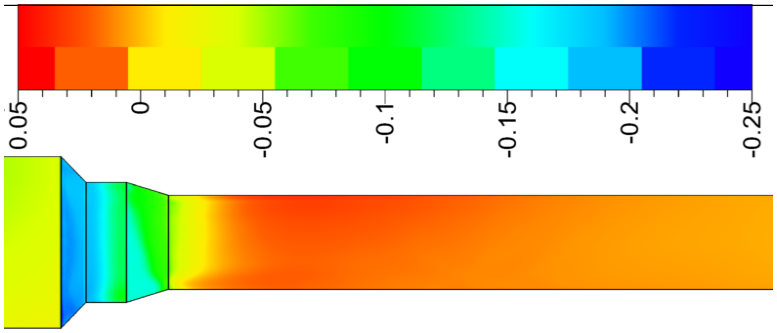


Figure 4.10: Pressure coefficient distribution over the surfaces of backward-facing steps of Taurus , @Ma2.41 and AoA 2Deg

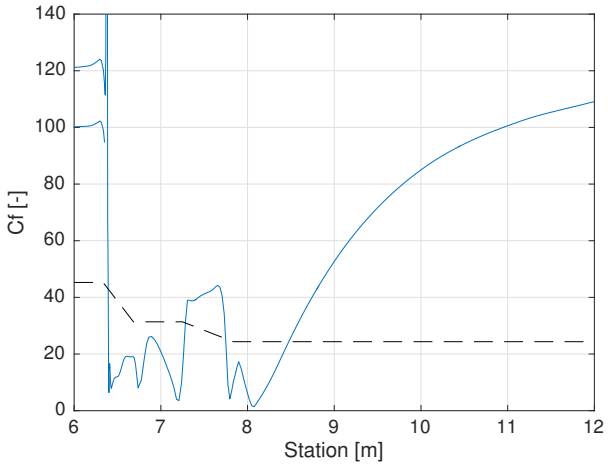


Figure 4.11: Closeup of C_f over the plane of symmetry along the backward-facing steps of Taurus, @Ma2.41 and AoA 2Deg. Two lines, both upside and downside.

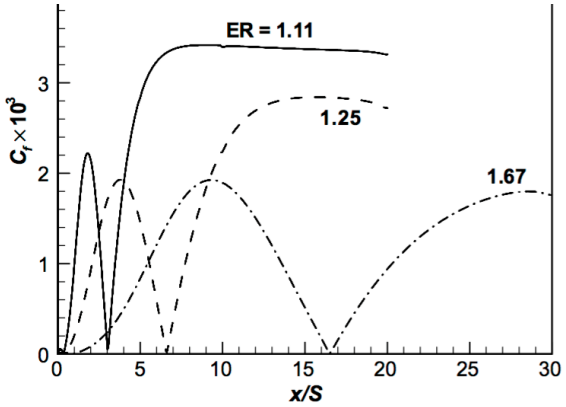


Figure 4.12: Distributions of the friction coefficient for dif-ferent step heights Chen et al. (2006), where ER stands for expansion ratio

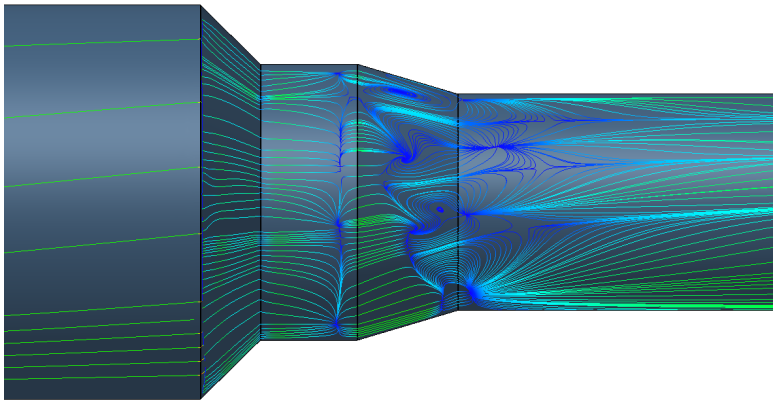
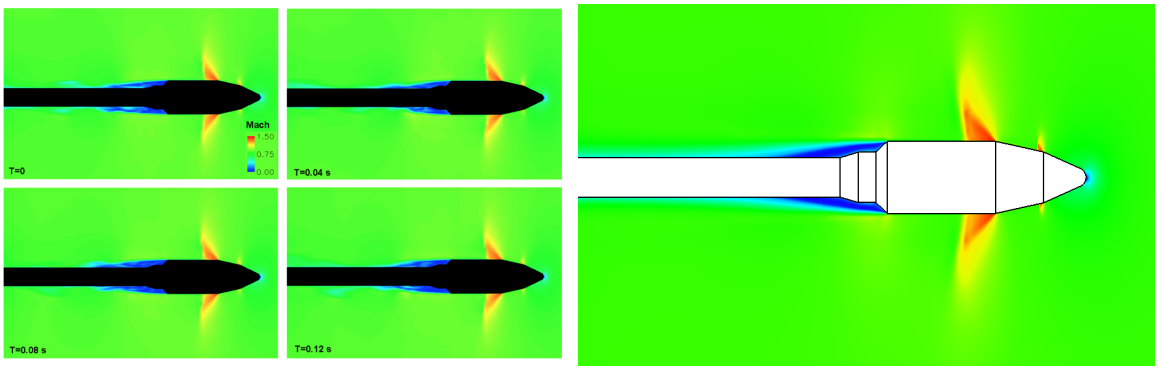


Figure 4.13: Skin friction lines on the double backward facing steps of Taurus, @Ma 2.41 and AoA 2 Deg



(a) Unsteady flow after the payload stage from Langone and Bermüdez (2009)

(b) Time-averaged flow after the payload stage from NUMECA

Figure 4.14: Mach distribution behind backward facing steps of Taurus @Ma 0.89 and AoA 2 Deg

Chapter 5

Results and analysis

VEGA AC from both Missile DATCOM and NUMECA will be computed, analyzed and fused in this chapter. In Section 5.1, input and output of Missile DATCOM will be described. In Section 5.2, the samples will be determined by way of LHS and manual selection, and the CFD simulations on those points will be carried out. During the simulation, the grid independent study on global and local values with be performed. In Section 5.3, data fusion will be carried out based on DATCOM and CFD results. Meanwhile, corresponding analysis will be given. In the end, the fidelity of the data fusion will be evaluated by extra CFD simulation on new samples.

5.1 DATCOM estimation

5.1.1 Input file

- `Flight condition`
  As mentioned in Chapter 3, to perform the DATCOM approximation, the flight condition, launcher geometry and desired output are necessary to define the DATCOM input file. For the flight condition, which contains Ma, Re and AoA etc., they can be (roughly) achieved from the VEGA flight profile from the users' manual of VEGA (Perez (2006)). In this project, we suppose the trajectory is perpendicular to the horizontal plane. Based on such assumption, only one parameter, the relative speed, is enough to define the flight profile. Distance from sea level (altitude) is the integration of the speed. With the altitude, important properties of the free stream, like density, temperature, laminar viscosity, static pressure, speed of sound, etc., can be computed based on the International Standard Atmosphere model. The corresponding Ma and Re can then be attained. The flight condition and flow property during the VEGA flight is illustrated in Figure 5.1 and Figure 5.2, respectively. Note that, due to the confidential issue, the flight data from Perez (2006) is to some extent different from other literature,

such as Nicolì et al. (2006). However, since the purpose of this project is focusing on multi fidelity data fusion, the input data is not that critical. The flight profile data from Perez (2006) is decent enough for this project.

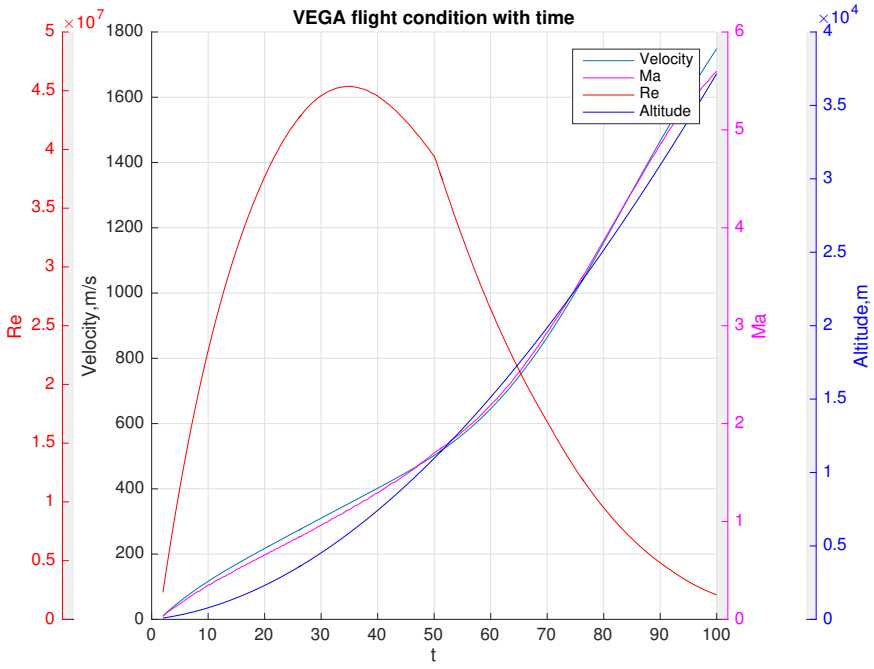


Figure 5.1: Part of the flight condition of VEGA

- `Geometry`
  The geometry data can also be obtained from users' manual of VEGA (Perez (2006)). In reality, the geometry is not built in a simple shape, e.g. there are small steps between stages and also there are other instruments such as retro-rockets locating around combination region of stages. Since this project is focusing on the data fusion application rather than 100 per cent simulation as done in work of Kitamura et al. (2013), we decide to use the simplified geometry contour, see Figure 5.3.

- `Input file`
  With the geometry data, flight profile and the output data (C_A, C_N and C_M), the input file for the Missile DATCOM can be built manually or from Matlab coding.

5.1.2 Result and analysis

The semi-approximation AC of VEGA can be achieved by executing Missile DATCOM. The result is shown in Figure 5.4.

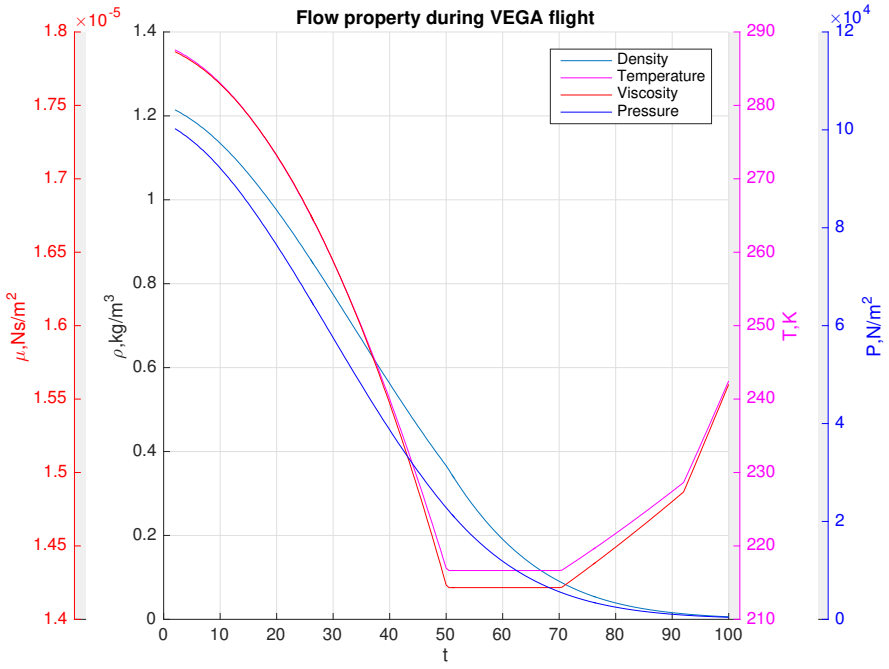


Figure 5.2: Flow property during VEGA flight

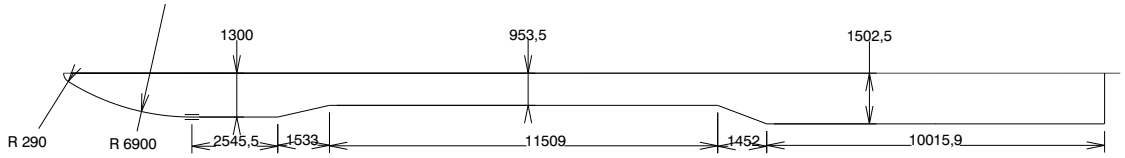
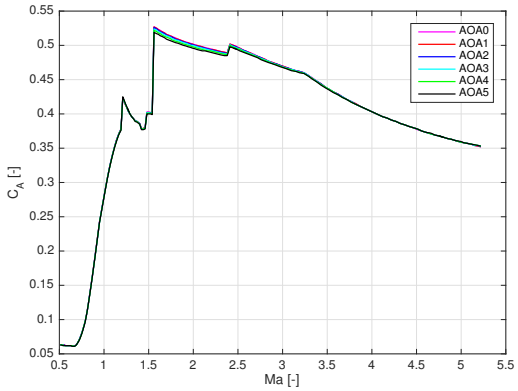
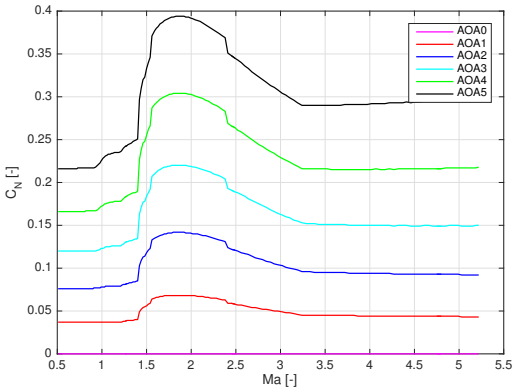


Figure 5.3: Simplified geometry of VEGA (in mm) (Perez (2006))

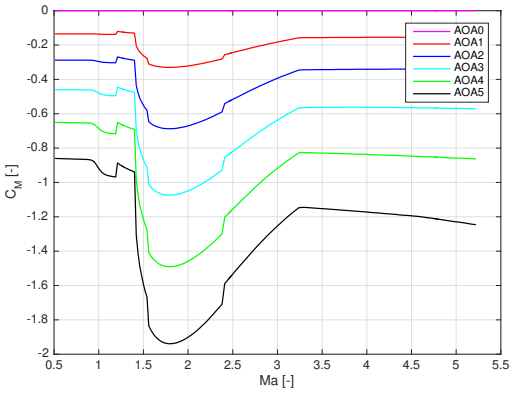
Trends of the results are generally consistent with those of the corresponding WTT data from Nicolì et al. (2006) and Catalano et al. (2007), which are shown in Figure 5.5. The C_A from DATCOM generally follows the same trend as the WTT data. Below Mach 1, the coefficient keeps increasing with the Mach number and reaches the maximum value around Ma 1.5. After that, it drops gradually. In terms of magnitude, the DATCOM results show that C_A is underestimated between Ma 0.91 and Ma 2.41. This is further illustrated in Figure 5.9 in the next section. With respect to normal force coefficient, the C_N from DATCOM does not show the same trend at Mach 1.4 to 2.1 as that in the WTT data. The values in this region are overestimated to a large extent by DATCOM. Trends in C_M reflect the behavior of C_N, as can be seen from Figure 5.5. Similar to C_N, C_M is overestimated in region between Ma 1.4 to 2.1. The accuracy of results from DATCOM will be further checked and improved by the following CFD simulations and Co-Kriging-based data fusion process. Note that an obvious error in the axial force figure is that there are multiple peaks as Ma increasing from 1 to 1.5, which is physically impossible. That can be explained by the various correlations that are used by DATCOM inside this Ma regime.



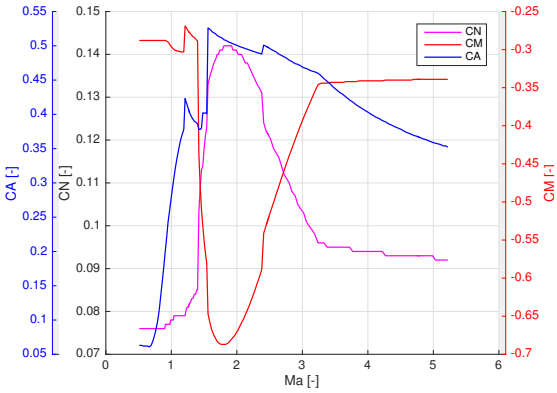
(a) C_A



(b) C_N

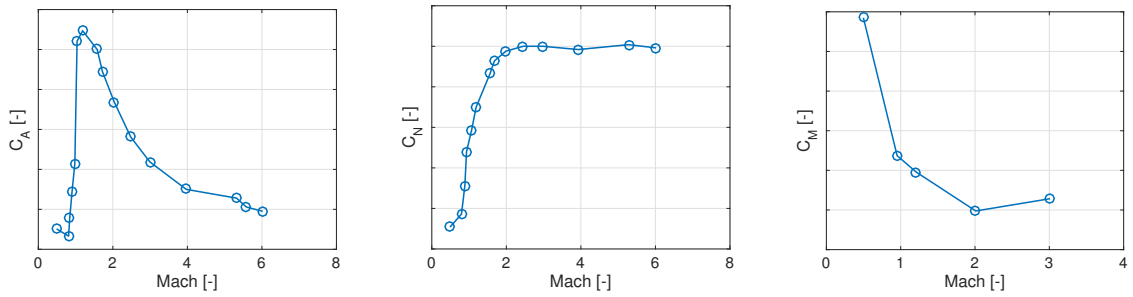


(c) C_M



(d) AC @ AoA=2 Deg

Figure 5.4: AC of VEGA from computation by Missile DATCOM



(a) C_A trends from WTT Nicolì et al. (2006)

(b) C_N trends from WTT Nicolì et al. (2006)

(c) C_M trend from WTT Catalano et al. (2007)

Figure 5.5: Partial AC data of VEGA from literature, Error bar being omitted; only trends are available for the confidential issue.

5.2 CFD simulations

5.2.1 Sampling points

In this project, the simulation data from CFD serves as the high-fidelity source. As this kind of data is computationally expensive, it should be used sparingly. We will perform the sampling in two directions, dimensions of AoA and Ma. From Figure 5.4, we can see that as AoA increases, the change of AC is simple (almost linear for C_N and C_M, and independent with C_A). So in the AoA-dimension, we can use the LHS scheme as mentioned in Section 3.4. In the Ma dimension, the samples will be chosen manually, which focuses mainly on the transonic region and takes Missile DATCOM data as a reference. The Ma, at which AC change sharply will all be taken into account. With the AoAs and Mas, we can combine them together and the final flight conditions to be simulated are listed in Table 5.1.

5.2.2 Simulation result

To perform efficient and accurate CFD simulations, the grid generation is of great importance. The quality and quantity of the grid directly influence the running time and final outcomes. Actually, the simple guideline in the CFD computation is to use less resource to get better simulation. As mentioned in work Langone and Bermüdez (2009) and Levy et al. (2014), for the initial grid, the first cell of the viscous layer is set to a height of about 6.1E-6 m and the cell-stretching ratio should be set to 1.1 to properly get the viscous layer over the solid surfaces. Also, the whole computing region is suggested to set as 100 times of the LV base radius in all direction. From experience of the project, the suggestion is reasonable and realistic. The total cell numbers for fine meshing is 18 million. The meshing for supersonic flow is shown in Figure 5.6.

Table 5.1: Sample points based on Ma and corresponding flow property

No.	**AoA(°)**	**Ma(-)**	**Re(-)**	**P(Pa)**	**T(K)**	μ (Ns/m^2)	ρ (kg/m^3)	**V** (m/s)
1	0.3	0.52	3.21E+07	84146	278.2	1.74E-05	1.05E+00	174
2	3.0	0.69	3.87E+07	74692	271.9	1.71E-05	9.57E-01	227
3	3.8	0.72	3.96E+07	72835	270.6	1.70E-05	9.38E-01	236
4	4.1	0.91	4.40E+07	60466	261.2	1.66E-05	8.06E-01	296
5	1.9	1.09	4.53E+07	49793	251.7	1.61E-05	6.89E-01	347
6	0.0	1.19	4.52E+07	44137	246.0	1.58E-05	6.25E-01	374
7	1.1	1.21	4.51E+07	43227	245.1	1.57E-05	6.14E-01	379
8	4.4	1.39	4.36E+07	34385	234.6	1.52E-05	5.11E-01	425
9	4.7	1.42	4.32E+07	32720	232.4	1.51E-05	4.90E-01	435
10	2.7	1.46	4.27E+07	31076	230.1	1.49E-05	4.70E-01	445
11	4.9	1.48	4.24E+07	30257	229.0	1.49E-05	4.60E-01	449
12	2.2	1.52	4.19E+07	28714	226.7	1.48E-05	4.41E-01	459
13	0.9	1.54	4.16E+07	27904	225.5	1.47E-05	4.31E-01	465
14	2.4	1.56	4.13E+07	27153	224.3	1.46E-05	4.22E-01	470
15	3.6	1.89	3.31E+07	17189	216.7	1.42E-05	2.76E-01	559
16	3.9	2.38	2.32E+07	9587	216.7	1.42E-05	1.54E-01	702
17	3.1	2.41	2.27E+07	9247	216.7	1.42E-05	1.49E-01	712
18	2.0	2.49	2.17E+07	8590	216.7	1.42E-05	1.38E-01	733
19	1.5	3.24	1.39E+07	4267	218.2	1.43E-05	6.81E-02	960
20	1.4	4.02	8.65E+06	2192	222.5	1.45E-05	3.43E-02	1201
21	0.6	5.22	3.36E+06	695	232.8	1.51E-05	1.04E-02	1598

With the meshing files, the grid independence of VEGA can be performed as that has done to the Taurus in Section 4.2. Both globally and local checks are carried out, examples of which are shown in Figure 5.7 and Figure 5.8, respectively. In the both figures, we can see that with the mesh being refined, there is some obvious fluctuation near regions of the two long cylinders. This is caused by the forward and backward facing steps, which can not be resolved by simple cell refinement. From both figures, we can say that the result of grid independence study is acceptable.

After the NUMECA simulation, AC of VEGA under the 21 selected fight conditions can be achieved. The results are listed in Table 5.2.

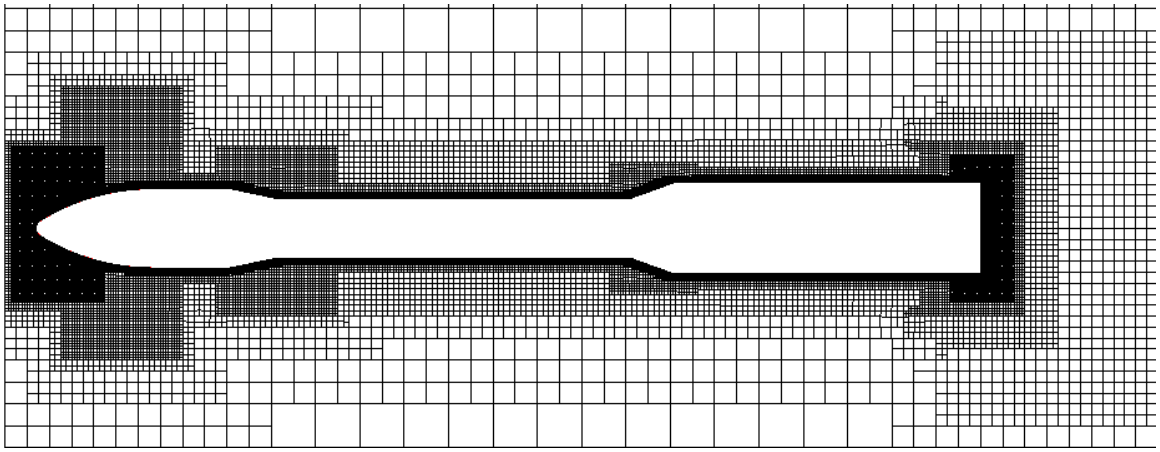


Figure 5.6: Finer meshing of VEGA for supersonic (partial)

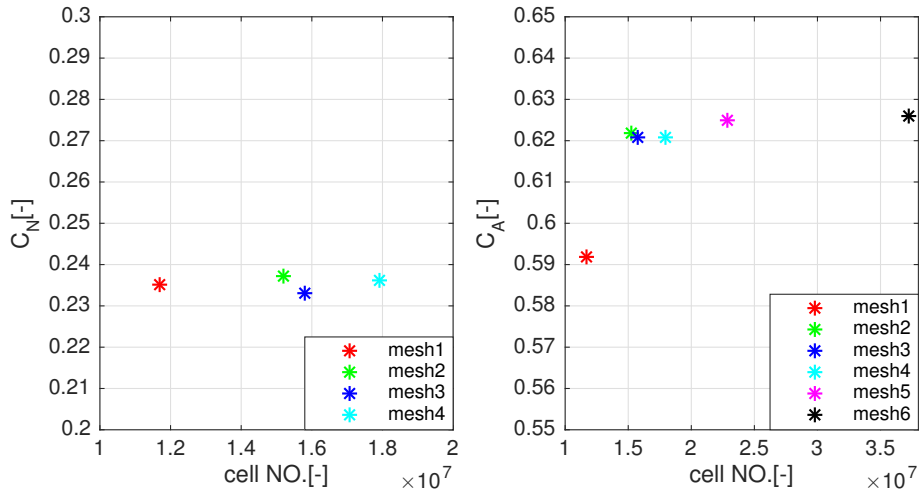


Figure 5.7: Grid independent study, normal force coefficient @Ma 1.09 AoA5 Deg and axial force coefficient @Ma 1.02 AoA0 Deg

Table 5.2: AC of VEGA from NUMECA simulation

No.	AoA	Ma	C_N	C_A	C_M	No.	AoA	Ma	C_N	C_A	C_M
1	0.3	0.52	0.0149	0.0878	0.0822	12	2.2	1.52	0.1190	0.6182	0.4750
2	3	0.69	0.1457	0.1049	0.7623	13	0.9	1.54	0.0497	0.6095	0.1985
3	3.8	0.72	0.1844	0.1145	0.9552	14	2.4	1.56	0.1323	0.6081	0.5300
4	4.1	0.91	0.1847	0.2667	0.8619	15	3.6	1.89	0.2104	0.5591	0.8488
5	1.9	1.09	0.0914	0.6198	0.3517	16	3.9	2.38	0.2337	0.4957	0.9549
6	0	1.19	0.0001	0.6454	0.0000	17	3.1	2.41	0.1829	0.4897	0.7454
7	1.1	1.21	0.0544	0.6404	0.2228	18	2	2.49	0.1189	0.4878	0.4831
8	4.4	1.39	0.2354	0.6482	0.8996	19	1.5	3.24	0.0853	0.4215	0.3360
9	4.7	1.42	0.2525	0.6396	0.9739	20	1.4	4.02	0.0777	0.3755	0.3016
10	2.7	1.46	0.1441	0.6268	0.5678	21	0.6	5.22	0.0325	0.3231	0.1313
11	4.9	1.48	0.2701	0.6360	1.0612						

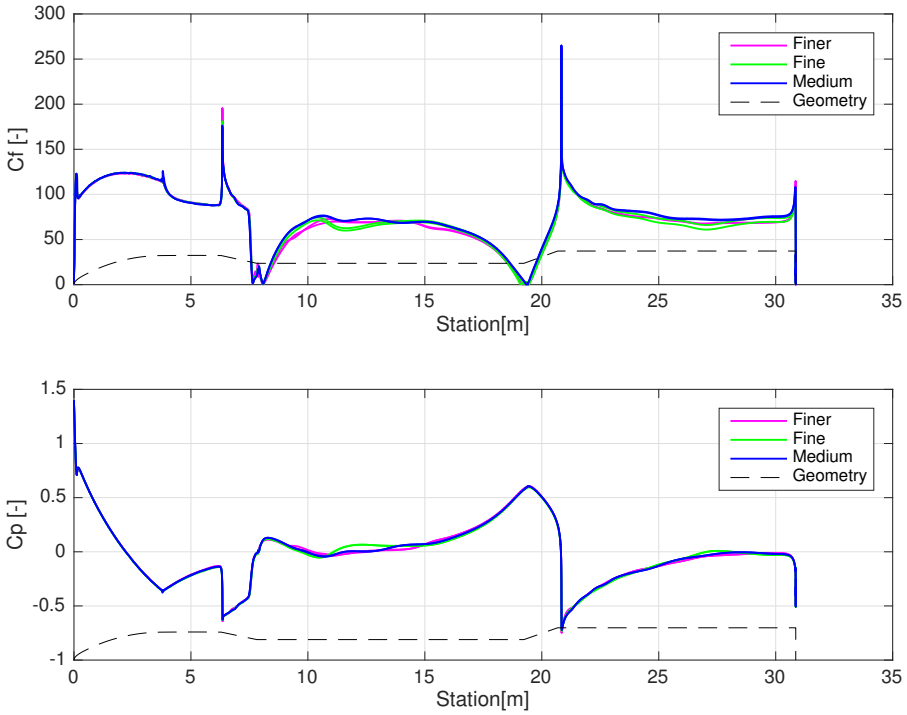
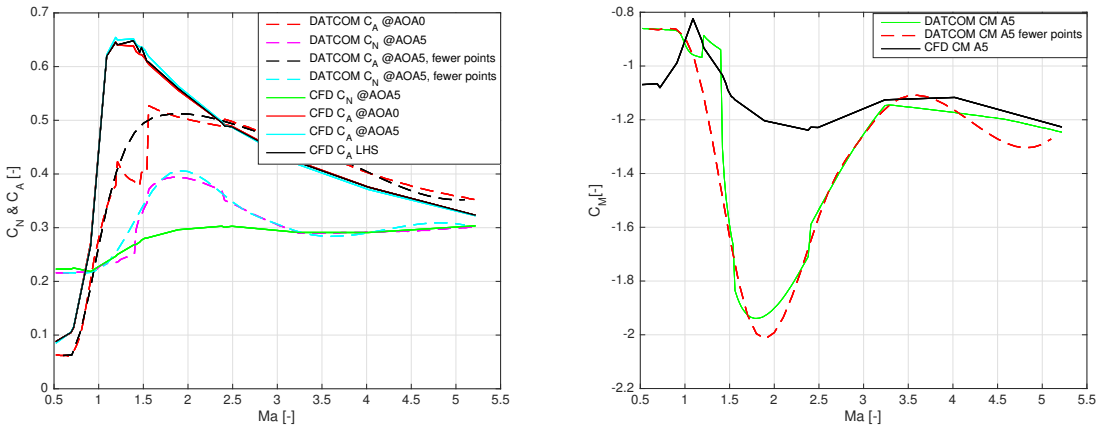


Figure 5.8: Grid independent study on pressure and friction coefficient over the plane of symmetry of VEGA, @Ma 1.09 AoA0 Deg



(a) Normal and axial force coefficient

(b) Pitching-up moment coefficient

Figure 5.9: Comparison of data from DATCOM and CFD

5.3 Data fusion and analysis

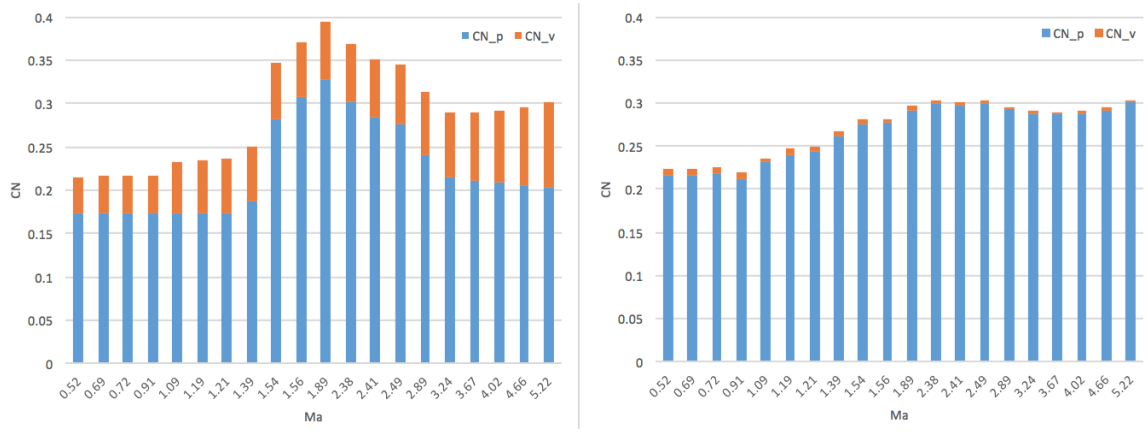
With data from Missile DATCOM and results on sampling points from NUMECA, the data fusion based on Co-Kriging can be performed. Before applying the Co-Kriging scheme, some preparation work should be done. Figure 5.9 shows the comparison of data from DATCOM and CFD simulations. In the figure, we can see that kinks of the DATCOM results are faulty when comparing results from the CFD simulations. Those kinks fail to provide useful information, on the contrary, they can be misleading during the data fusion process. In the region of Ma 1.2 to Ma 2.41, fewer points should be taken into account to alleviate the potential influence. As shown in Figure 5.9, only a few points will be adopted to serve as the low-fidelity data to avoid the useless discontinuities. In the following fusion process, it proved to save a great deal of time by the preliminary kink-cancelation.

Beside the total data comparison shown in Figure 5.9, a detailed analysis can be performed, as illustrated in Figure 5.10. In Figure 5.10, the coefficients are broken down into non-viscous and viscous parts, where the legend $\_v$ represents the viscous contribution and $\_p$ stands for non-viscous contribution. By comparing each part, we can track where the inaccuracy comes from. Note that the values on the abscissa are not averagely distributed and these figures don't provide exact trends of the AC.

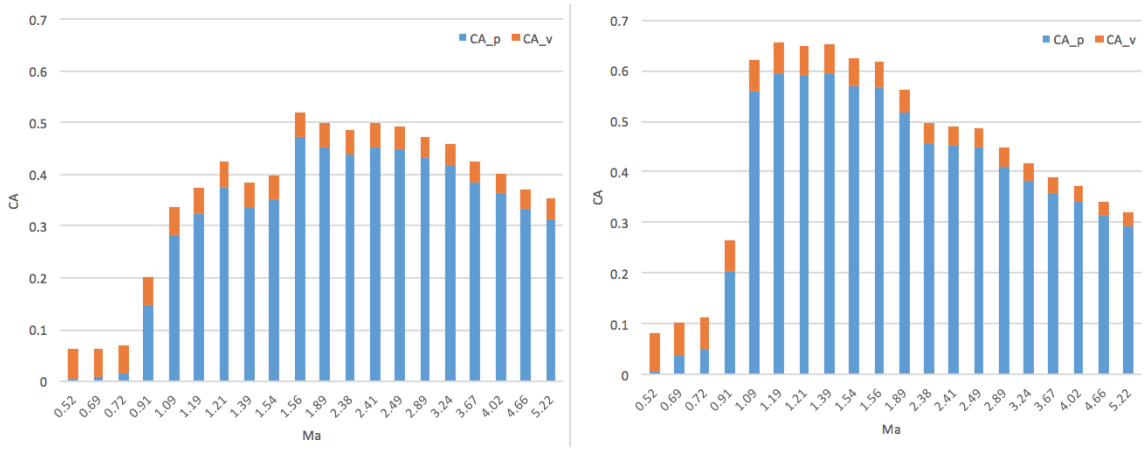
In Figure 5.10(a), it is clear that the viscous effect for the C_N is overestimated by DATCOM. In terms of quantity, it is tens of times higher than that of the CFD results. This is caused by the concept behind DATCOM, supposing inviscid flow during the estimation and adding the viscous effect later. With the potential flow contribution, DATCOM overestimated it around Ma 1.56 to 2.38. For other Ma regime, the coefficient is underestimated. The exact reason for this inaccuracy is not clear at the moment. In Figure 5.10(a), we can see that the viscous effect for the C_A is generally accurately estimated by DATCOM and the main error lies in the non-viscous part. From Ma 0.91, DATCOM failed to capture the dramatic increase, which is mainly wave drag. After Ma 2.38, the result from DATCOM turns relatively accurate again. C_M from DATCOM almost has the same trends as that of C_N, which has an overestimated viscous effect. With the C_M from CFD, there is a slight drop around Ma 1.09 (or the values lower than Ma 1.09 are slightly higher than they should be; see experiment trends in Figure 5.5(c)). This inconsistency is from the inaccuracy of the CFD simulations in C_M.

5.3.1 Normal force coefficient

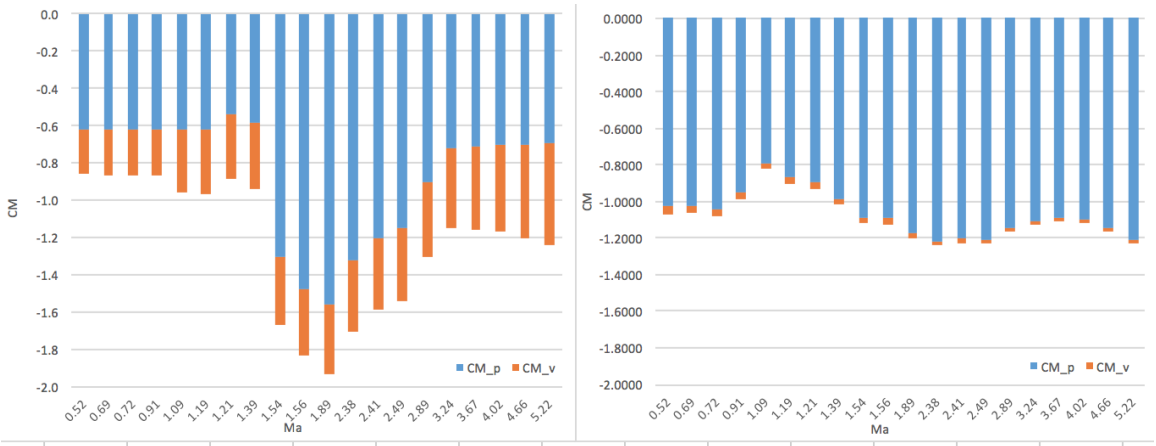
In this subsection, we perform data fusion on the normal force coefficient C_N. At the beginning, we only take the CFD simulation data on the LHS samples as the expensive data source. The fusion result is shown in Figure 5.11 and Figure 5.12. In Figure 5.11, comparison of data from DATCOM and data fusion is illustrated. We can see that with only 21 expensive points, we improve the DATCOM data globally. In Figure 5.12, the green circle stands for expensive data from CFD and red asterisk stands for estimation error of the fusion on corresponding point. To make it clear, the error is multiplied by a factor of 100. The most obvious error we can see in this figure is that the lift coefficient at zero AoA is not zero, especially in the



(a) Normal and axial force coefficient. Left from DATCOM, right from CFD.



(b) Normal and axial force coefficient. Left from DATCOM, right from CFD.



(c) Pitching-up moment coefficient. Left from DATCOM, right from CFD.

Figure 5.10: Lumped comparison of data from DATCOM and CFD, cases with AoA 5 Deg.

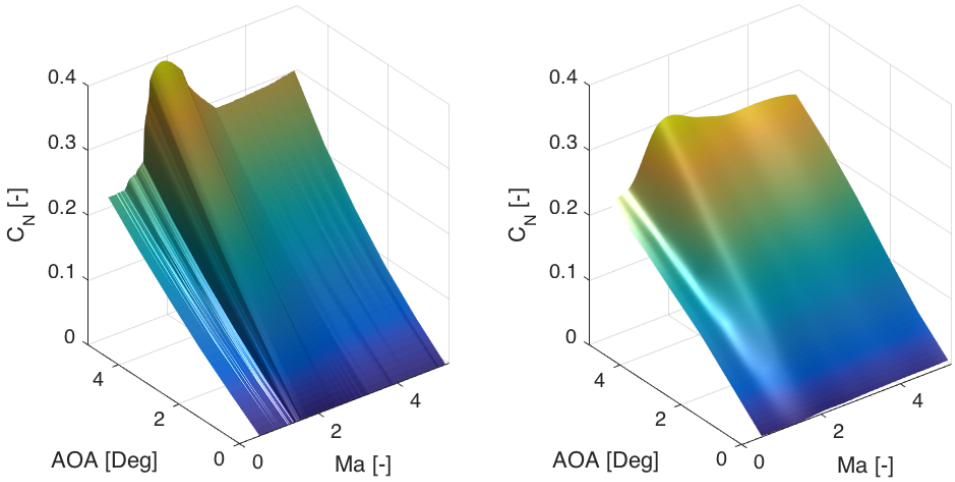


Figure 5.11: Raw result from DATCOM (Left) V.S. Fusion result of C_N, with LHS samples as expensive data source (Right)

region between Ma 2 and Ma 5. This error information is indicated by the corresponding error asterisks (amplitudes at AoA 1 are smaller compared to those at AoA 0. The reason for this inaccuracy is that there is only one point is sampled at AoA 0, the case of Ma 1.19, see Table 5.2. To reduce such error, extra expensive data at high Mach and AoA 0 should be added. In fact, these data isn't *expensive*, for all C_N is zero at zero-AoA. Besides error at zero-AoA region, the error at Ma 3 to Ma 5 is relatively large. That is because most of the samples located in the 'low' Mach region (Ma 0.5-3). Such distribution is caused by the uneven distribution of samples in the Ma-dimension. From the error matrix, the current max value (Co-Kriging from LHS samples) locates at Ma 4.66 and AoA 5. With help of the error information, extra samples can be added accordingly to reduce corresponding estimating error.

With the error info, extra samples are added. One point is added at each time and then the estimating error is evaluated based on the new result. After several rounds of improvement, the error is significantly reduced, from original $2.1E-3$ to $9.8E-5$, as shown in Figure 5.13. The accuracy of the improvement will be further evaluated in Section 5.4. The extra samples is plotted in Figure 5.14. The 21 green circles stand for LHS samples and 19 red asterisks represent extra-added points. From the final sample distribution (which leads to a decent estimating error), we can say that an evenly-distributed sample is indispensable to perform a good Co-Kriging process. In this project, the sampling scheme in the Ma-dimension is misled by results from DATCOM which shows many faulty kinks in the AC curves. The improper sampling scheme gives rise to the sample concentration in the region between Ma 1 and 2.4, which is some kind of waste of the expensive CFD simulations. In the future work, sample designation along the Ma-dimension should be uniform-distributed to reduce the quantity of the expensive samples.

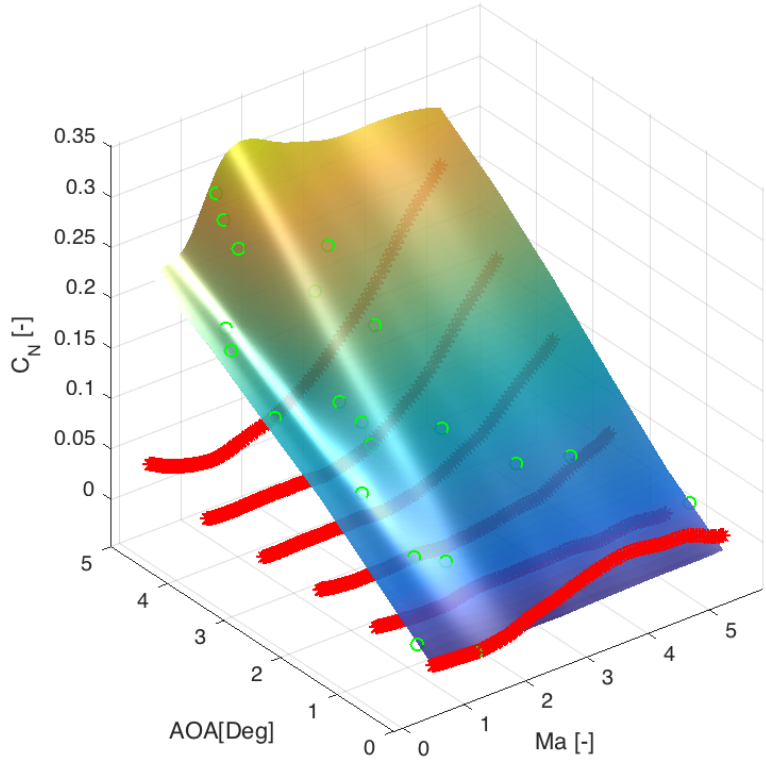


Figure 5.12: Fusion result of C_N, with LHS samples (green circles) as expensive data source, and the error is multiplied by a factor of 100

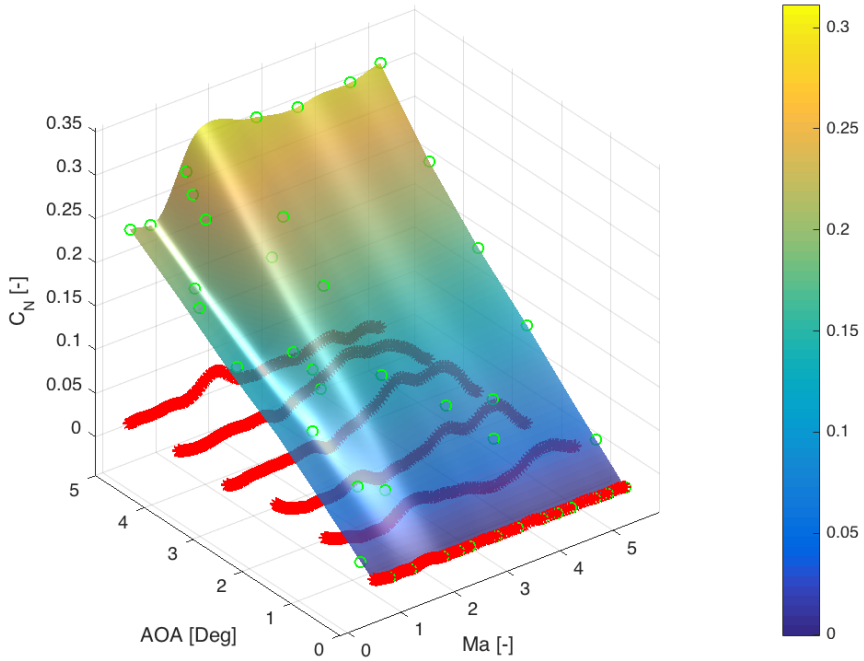


Figure 5.13: Improved fusion result of C_N, green circles represent the expensive samples and the error is multiplied by a factor of 500

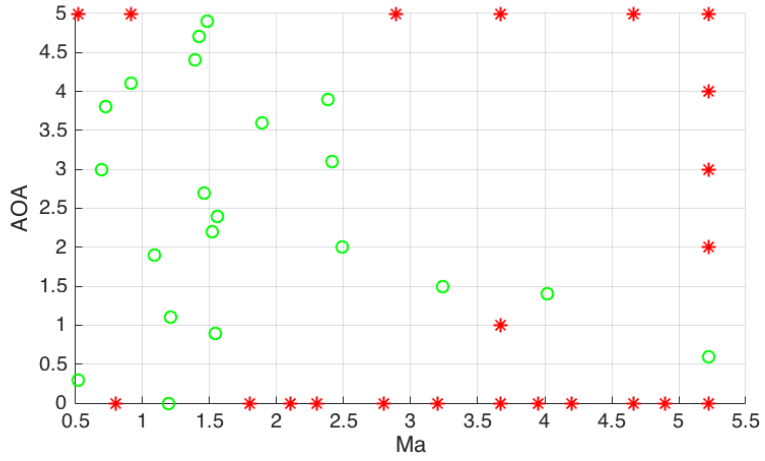


Figure 5.14: Initial LHS and extra added points during the C_N Co-Kringing process; green circle and red asterisk representing LHS sample and extra-added point, respectively

5.3.2 Axial force coefficient

As did in Section 5.3.1, data fusion is performed on the C_A. In the first fusion, the 21 LHS samples are used. The fusion results are compared with DATCOM data, shown in Figure 5.15. From the figure, we can see that the DATCOM data has many faulty kinks in the region between Ma 1.21 to 2.38. Those kinks are caused by the inner estimating algorithm shifting. They add much uncertainty to the fusion process and fail to provide useful information, instead, can mislead the final results. To improve the efficiency and accuracy of the fusion process, DATCOM output in this region is partially adopted, as shown in Figure 5.9. Only a few points are chosen to describe the trends in this region. In the region from Ma 0.7 to 1.2, both DATCOM and CFD show that the C_A changes significantly as Ma increases. The estimation from DATCOM is relatively accurate in this section. For cases where Ma is higher than 2.4, DATCOM gives good trends and values, with a relative error of around 10%. Note that the fusion result fluctuates in this region, though the variations are not large. That is because only 3 samples are used (see Figure 5.18) in this area, Ma from 2.4 to 5.22 and AoA from 0 to 5. Such points are not adequate to improve C_A globally (at all evaluated points). Even though, the outcome is not bad. Only three expensive samples have changed the characteristics to a large extent. In the following steps, as more expensive samples added in this region, the fluctuation can be well reduced.

Figure 5.16 gives both the fusion result and error matrix in the whole Ma-AoA region . The max estimating error is $3.53E-2$, locating at Ma 4.66 and AoA 5. As performed in the former section, corresponding expensive samples should be added to reduce the maximum error. With the error data showing the direction, a few new samples are added step by step. The improved fusion result has a maximum error of $4.2E-3$, ten times reduced compared to the original fusion. The C_A and corresponding error values are illustrated in Figure 5.17. In the figure, the surface fluctuation amplitude is reduced to a quite small degree. And this can be further decreased when more samples are added. However, extra samples are quite expensive (in terms of computational intensity) and also, to this step, the fusion improvement

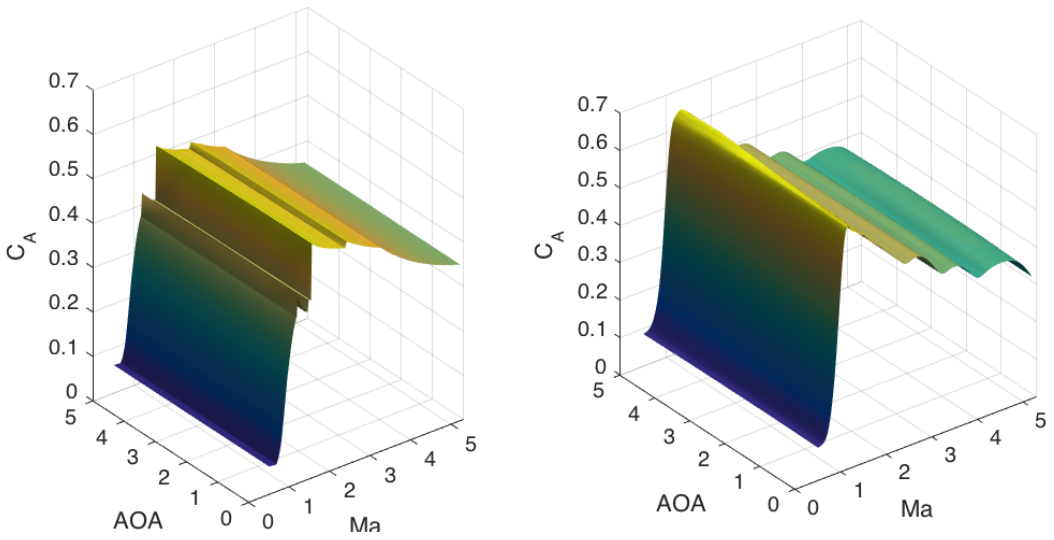


Figure 5.15: Raw result from DATCOM (Left) V.S. Fusion result of C_A, with LHS samples as expensive data source (Right)

is accurate enough. The improvement in terms of accuracy will be checked in Section 5.4. For this fusion-improving process, 8 samples are added, which are plotted in Figure 5.18. Like the C_N improving process, the extra samples in this process mainly locate in the upright corner of the figure, where few samples are inserted in the LHS. The quantity of extra samples is fewer than that of the C_N process, that because the C_A characteristics surface is relatively less complicated than C_N and a few points are enough to get the same accuracy.

One indispensable step for the C_A fusion process is that special attention should be paid to the region between Ma 0.72 to Ma 1.21, for the gradient, $\frac{dC_A}{dM\alpha}$, is extraordinarily large. From experience of this project, fusion error in this region can be large even corresponding estimating error S^2 (see Equation 3.29) is shown small. More samples should be added or evaluated in this region. Alternatively, the improved Co-Kriging, Gradient-Enhanced recursive CoKriging (GECK) can be applied, as discussed by Ulaganathan et al. (2015). By using GECK, gradient information behind the samples (expensive data) can be utilized in the Kriging process and reduce the quantity of samples needed. This application is suggested to carry out in the future work.

5.3.3 Pitching moment coefficient

The data fusion process and results for C_M share similar outcome as that of C_N. Initial C_M fusion result based on LHS is illustrated in Figure 5.19. The difference between the two characteristics surface is quite similar with that of the C_N. That is reasonable since the pitching moment is mostly determined by the normal force. The C_M fusion result and corresponding error is shown in Figure 5.20. For the initial case, the maximum error is $3.39E-2$, locating at Ma 4.71, AoA 5. Based on the error, extra samples are added step

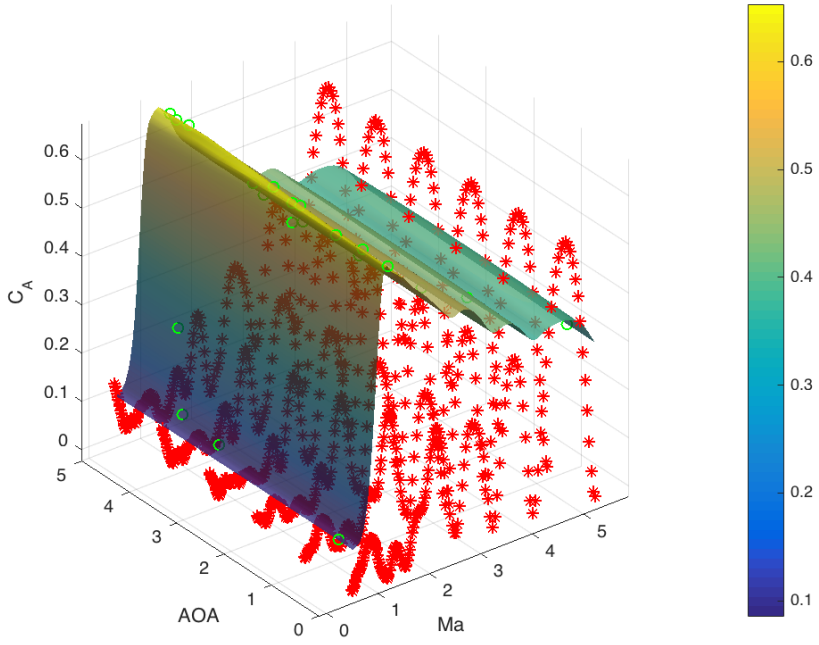


Figure 5.16: Fusion result of C_A, with LHS samples as expensive data source, and the error is multiplied by a factor of 50.

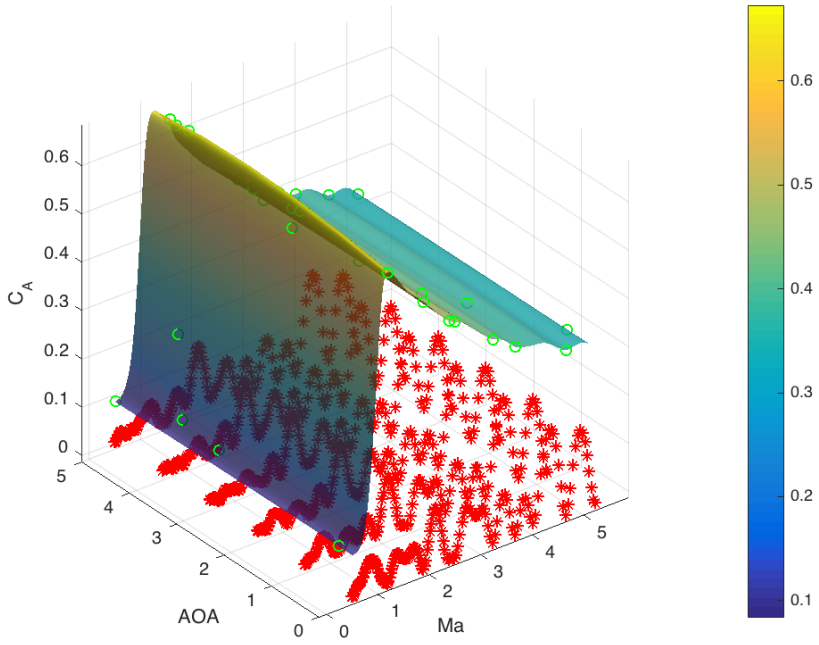


Figure 5.17: Improved fusion result of C_A, and the error is multiplied by a factor of 50.

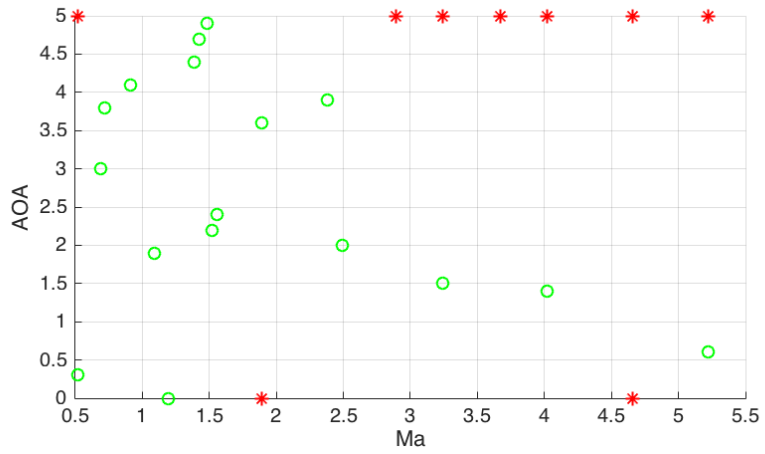


Figure 5.18: Initial LHS and extra added points during the C_A Co-Kringing process; green circle and red asterisk representing LHS sample and extra-added point, respectively

by step and the final improved C_M result is given in Figure 5.21. The improved maximum estimating error is reduced to $3.3E-3$. The extra samples contain 9 points and are plotted in Figure 5.22.

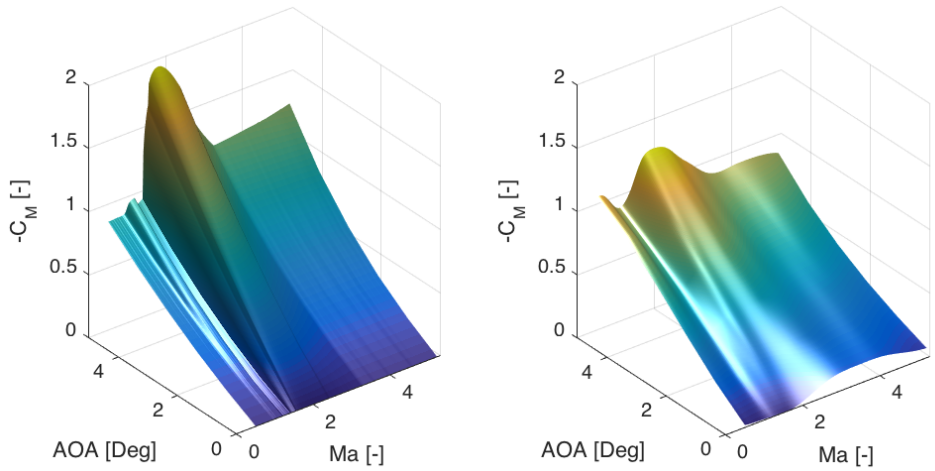


Figure 5.19: Raw result from DATCOM (Left) V.S. Fusion result of C_M, with LHS samples as expensive data source (Right)

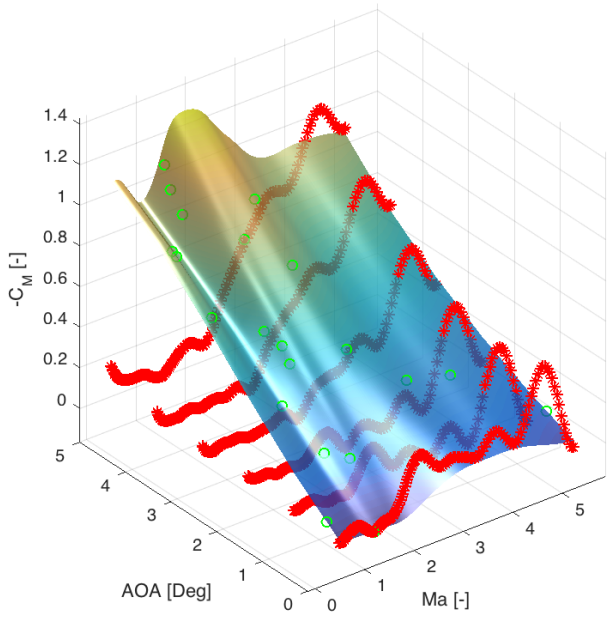


Figure 5.20: Fusion result of C_M, with LHS samples as expensive data source, and the error is multiplied by a factor of 30.

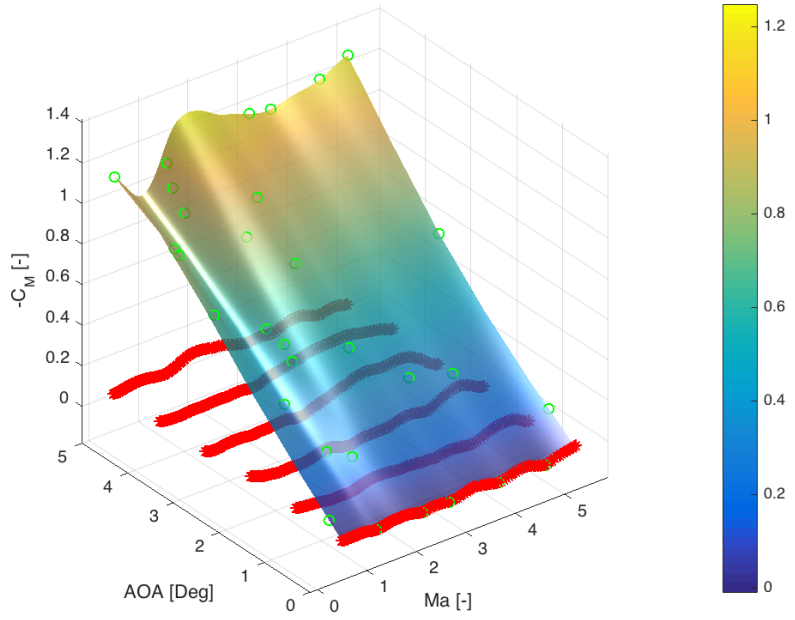


Figure 5.21: Improved fusion result of C_M, and the error is multiplied by a factor of 30.

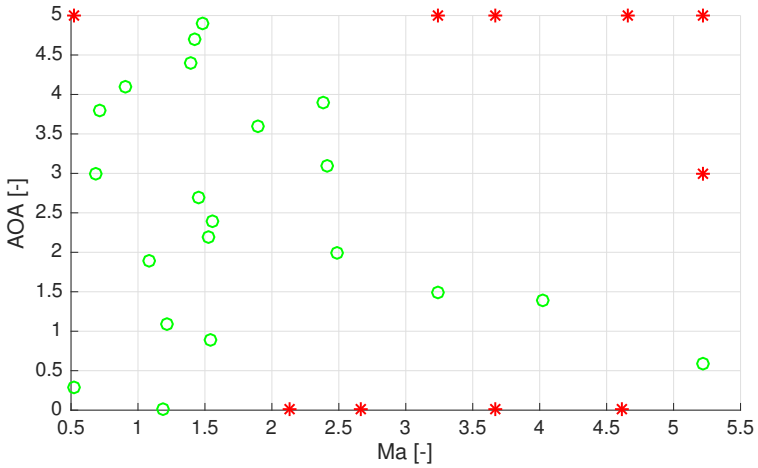


Figure 5.22: Initial LHS and extra added points during the C_M Co-Kringing process; green circle and red asterisk representing LHS sample and extra-added point, respectively

5.4 Fusion result evaluation

In this section, the Co-Kriging based data fusion will be evaluated by way of comparing new CFD results on random samples with corresponding results from the fusion process. Also, the comparison between DATCOM estimation and data fusion will be given to prove the advantage of applying the data fusion. After that, the fusion will be remarked in terms of time-saving.

5.4.1 Accuracy analysis

Table 5.3 lists the AC from DATCOM, data fusion and CFD simulations. The new flight conditions in the table are randomly selected. From the table, we can see that for most of the evaluated points the accuracy of data fusion is quite high, with an error not big than 8%. Much better than that from DATCOM. However, for the Ma 0.99 and AoA 5 Deg case, the accuracy for the fusion is lower than DATCOM, with a value 7.2% vs. 5%. That means sometimes the fusion can also degrade the accuracy, even though only to a small degree. This degradation actually is not a problem, for one thing, the result is still accurate enough within the current step and for another, the degradation can be eliminated when extra samples are added. In a nutshell, the fusion result for the C_N is satisfactory.

Similar results are achieved in terms of C_A and C_M, which are shown in Table 5.4 and Table 5.5, respectively. In those two tables, even for points that have an error of around 60% (C_M case with Ma 1.56 and AoA 5 Deg), the error can be reduced to less than 15%. This reduction is decent to the pitching-up moment coefficient.

Table 5.3: Improvement of C_N after Data Fusion, error comparison

No.	Ma	AoA	DATCOM	Fusion	No.	Ma	AoA	DATCOM	Fusion
1	0.69	5	3.1%	0.8%	12	1.54	5	23.3%	1.0%
2	0.72	5	4.0%	1.9%	13	1.56	5	31.7%	1.5%
3	0.99	5	5.0%	7.2%	14	1.89	5	33.0%	4.6%
4	1.09	5	1.5%	4.5%	15	1.97	5	30.7%	3.5%
5	1.19	5	4.7%	4.4%	16	2.38	5	21.8%	1.0%
6	1.21	5	5.6%	4.8%	17	2.41	5	16.7%	1.5%
7	1.39	5	6.6%	1.9%	18	2.49	5	14.0%	0.0%
8	1.42	5	10.5%	0.7%	19	3.24	5	0.4%	0.0%
9	1.46	5	17.1%	0.3%	20	4.02	5	0.4%	1.1%
10	1.48	5	16.6%	0.6%	21	4.8	3	10.2%	0.8%
11	1.52	5	21.2%	0.1%					

Table 5.4: Improvement of C_A after Data Fusion, error comparison

No.	Ma	AoA	DATCOM	Fusion	No.	Ma	AoA	DATCOM	Fusion
1	0.52	0	28.2%	0.5%	24	1.54	0.9	34.0%	0.8%
2	0.69	5	39.1%	2.1%	25	1.54	5	36.0%	0.2%
3	0.69	0	39.7%	6.1%	26	1.54	0	34.0%	2.1%
4	0.72	5	38.9%	0.7%	27	1.56	5	16.1%	0.4%
5	0.72	0	38.4%	7.0%	28	1.56	0	12.8%	1.9%
6	0.91	5	23.5%	2.9%	29	1.89	5	11.3%	2.2%
7	0.99	5	40.6%	0.4%	30	1.97	5	10.5%	2.7%
8	1.09	5	45.8%	0.0%	31	2.38	5	2.5%	1.3%
9	1.09	0	45.5%	4.7%	32	2.38	0	1.4%	0.0%
10	1.19	5	42.5%	3.7%	33	2.41	3.1	2.3%	0.9%
11	1.21	1.1	34.0%	4.1%	34	2.41	5	1.4%	0.4%
12	1.21	5	34.6%	4.5%	35	2.41	0	2.0%	0.3%
13	1.39	5	40.8%	1.3%	36	2.49	5	1.4%	0.6%
14	1.39	0	39.4%	2.5%	37	2.49	0	1.8%	0.2%
15	1.42	5	41.2%	0.7%	38	3.24	0	8.8%	0.8%
16	1.42	0	39.8%	1.8%	39	3.67	1	7.8%	2.6%
17	1.46	2.7	39.5%	1.3%	40	4.02	0	7.0%	0.8%
18	1.46	5	40.6%	0.1%	41	4.8	3	7.8%	3.3%
19	1.46	0	39.0%	2.0%	42	5.22	0	9.0%	0.3%
20	1.48	5	37.2%	0.2%	43	5.22	2	8.8%	0.5%
21	1.48	0	35.3%	2.5%	44	5.22	3	9.0%	0.7%
22	1.52	5	36.5%	0.3%	45	5.22	4	9.4%	0.3%
23	1.52	0	34.6%	2.4%					

5.4.2 Time-saving advantage

The accuracy improvement of data fusion is remarkable as discussed in Section 5.4.1. Besides the accuracy, the advantage of time-saving is also worth to mention. From experience of this project, the average NUMECA computation time for each CFD simulation is about 100

Table 5.5: Improvement of C_M after Data Fusion, error comparison

No.	Ma	AoA	DATCOM	Fusion	No.	Ma	AoA	DATCOM	Fusion
1	0.69	5	-19.0%	-4.4%	14	1.56	5	-63.0%	-0.9%
2	0.72	5	-20.0%	-6.6%	15	1.89	5	-60.3%	-3.2%
3	0.91	5	-12.1%	-4.2%	16	1.97	5	-55.7%	-1.6%
4	0.99	5	-2.8%	-0.7%	17	2.38	5	-38.0%	-1.8%
5	1.09	5	-16.3%	-12.5%	18	2.41	5	-29.4%	-1.4%
6	1.19	5	-6.9%	-4.3%	19	2.49	5	-25.3%	-2.5%
7	1.21	5	-4.9%	-2.0%	20	2.89	5	-12.1%	-0.8%
8	1.39	5	-8.1%	-1.3%	21	3.67	1	-27.3%	-0.8%
9	1.42	5	-26.3%	-1.5%	22	4.02	5	-5.0%	-1.0%
10	1.46	5	-39.8%	-0.8%	23	4.8	3	-12.0%	-0.4%
11	1.48	5	-41.7%	-0.1%	24	5.22	2	-23.8%	-8.6%
12	1.52	5	-46.7%	-0.1%	25	5.22	3	-11.3%	0.0%
13	1.54	5	-49.1%	-0.6%	26	5.22	4	-3.4%	-4.0%

hours, computed parallelly with 8 cores (CPU Intel Xeon E5-2670 v2; clock speed 2.6 GHz). For cases with high Mach and AoA, the time consumed is relatively longer.

For this project, if we want to get AC of VEGA on a two dimensional flight condition matrix of 46*11 (Ma ranging from 0.5 to 5 with a step of 0.1, AoA ranging from 0 to 5 with a step of 0.5), we need to simulate 504 cases. By applying data fusion scheme mentioned in the former chapters, 40 cases are enough to achieve the same accurate results. 90% of the computation time can be saved.

Chapter 6

Conclusions and Recommendations

6.1 Conclusions

In this thesis, a new approach to implementing an aerodynamic framework of launchers is suggested. The basic idea behind is to apply the Co-Kriging based data fusion algorithm to industrial estimation result and CFD simulation data. The estimation result is the AC data from industrial semi-empirical methods like U.S. Air Force DATCOM. Such data is based on the empirical experience and approximating mathematic equations abstracted from history data. As a result, accuracy of such data tends to be relatively low. From experience of this project, the DATCOM error is as high as 60% (in terms of C_M, for cases of drag coefficient at Ma 1.56 with AoA 5 Deg). The other type of data source is AC from the CFD simulations, which are based on RANS in this project. The fidelity of the CFD code is evaluated by comparing simulation results with WTT data on a U.S. launcher. The accuracy turns out to be relatively decent. In terms of expenses, the first kind of data is easy to achieve, taking several seconds to get the multi-case results; the second kind of data, however, is much more time-consuming and computationally intensive, average time for one simulation being 100 hours in this project (8 CPUs, Intel Xeon E5-2670 V2; Clock speed 2.6 GHz). By performing the data fusion, the accuracy of low fidelity data (DATCOM data) can be improved significantly by adding a small quantity of expensive data (CFD data).

Before adopting the CFD simulation results as the accurate and expensive data, validation of the CFD code is carried out in advance. The fidelity of the CFD code (in this project, NUMECA) is evaluated by comparing AC results from both simulations and WTT on a U.S. launcher, Taurus. The reason for choosing it as the evaluation target is that this launcher is the only traditional launcher whose corresponding WTT data can be found in the public literature. During the CFD simulations, the one-equation viscous model, Sparlart-Allmaras Model, is adopted for its robustness, especially under the NUMECA code. After applying proper meshing scheme and the grid independence study on both local (pressure and friction coefficient over surfaces of the launcher) and global parameters (total C_N, C_A, C_M), the

accuracy of the CFD simulation turns out to be decent when compared with WTT data. For both C_N and C_A over subsonic and supersonic, the errors are lower than 10%, which is satisfactory for the simulations. With respect to the C_M, the error is high, being as large as 24%. Reason for this discrepancy is complicated and the important one can be the existence of the double-backward-facing steps after the payload stage of the Taurus. Flow near these regions is hard to predict and simulate by the nowadays CFD solvers. In fact, there is few WTT on such 3D continuous backward-facing steps. Uncertainty in such regions gives rise to uncertainty in forces and the location where forces exerted. By now, the author has tried both refining the mesh and changing turbulence models, but such discrepancy can not be reduced. Special effort should be made in this direction in the future work.

After the evaluation of the CFD code, the data fusion is implemented to the VEGA launcher. The first step is to establish the low-fidelity aerodynamic database from DATCOM. Three kinds of information are written into the input files: the geometry of the launcher, the definition of the flight condition and the desired outputs. The second step is performing CFD simulations on flight scenarios which are determined based on the low-fidelity aerodynamic database and the Latin Hypercube Sampling (LHS) scheme. With respect to Ma, points where kinks appear or where gradients (w.r.t. Ma and AoA) are high are chosen. In the dimension of the AoA, LHS is applied. In this project, to get all the AC in the full target flight regime, 40 cases turn out to be chosen, being simulated by CFD. Note that the quantity of the CFD simulations can be lower, since in this project the author was misled by the DATCOM result, adding several unnecessary simulations at places where AC kink appears. The extra added points near certain region directly lead to an ill-distributed sampling, which gives rise to more work in the fusion improving session. In the future work, it is strongly suggested to check if all the potential scenarios suggested by DATCOM are necessary. The final step, called the fusion process, is carried out based on the Co-Kriging algorithm. And the fusion results are evaluated by comparing them with corresponding new CFD simulations. The comparison demonstrates that the proposed data fusion is capable of improving the low-fidelity aerodynamic database from DATCOM to a great extent, with only a small quantity of computationally expensive CFD simulations. In this project, the maximum error is reduced from as large as 60% to 15%. Should more expensive data is added, the accuracy can be further improved.

In conclusion, this project suggests an efficient way of establishing a high-fidelity AC database by harnessing Co-Kriging algorithm. Data from industrial estimation is improved to high accuracy with a small quantity of CFD simulations. The suggested method has a big advantage in saving time. In this project, 90% computation time is saved to achieve the same accurate database.

6.2 Recommendations for future work

In this section, recommendations and suggestions are given on aspects where, after this project, problems still remain and where improvement can be performed. They contain three topics.

The first one is the CFD simulation on flow near the single/double (continuous) backward-facing steps. Effort should be made in this region to achieve satisfactory results when compared to that from WTT. This contains the critical situation when the AoA is relatively high and separation is significant.

The second suggestion is adopting the more complex Co-Kriging algorithm, the gradient-enhanced Co-Kriging (GECK), as suggested by Han et al. (2013) and Ulaganathan et al. (2015). The name itself suggests that the concept is good at handling data where the gradient is high. Rather than adding extra more points in such region, only a few sampling can be enough. Another advantage of GECK can be, by using such scheme, the noise in the data source (see the kinks of the CFD results in Figure 5.9) can be filtered before the fusion process, which will improve the fusion accuracy, meanwhile, decreases the need for extra simulations in the improving process of the fusion.

Finally, the data fusion concept is suggested to be applied to three-fidelity or even more data source. The extra data source can be WTT data, CFD simulation from LES and DNS or other highly accurate simulation concepts. Both computational intensity and accuracy of the fusion are suggested to be evaluated.

Bibliography

John David Anderson Jr. *Fundamentals of aerodynamics*. Tata McGraw-Hill Education, 2010.

K. L. Bibb, E. L. Walker, G. J. Brauckmann, and P. E. Robinson. Development of the orion crew module static aerodynamic database, part ii: Supersonic/subsonic. In *29th AIAA Applied Aerodynamics Conference 2011*, 2011.

William B Blake. Missile datcom–1997 status and future plans. In *the AIAA Applied Aerodynamics Conference, Atlanta, Georgia*, 1997.

William B Blake. Missile datcom: User's manual-1997 fortran 90 revision. Technical report, DTIC Document, 1998.

George EP Box, Norman Richard Draper, et al. *Empirical model-building and response surfaces*, volume 424. Wiley New York, 1987.

George Hartley Bryan. *Stability in aviation*. Hardpress Publishing, 2012.

F. Castellini. *Multidisciplinary Design Optimization for Expendable Launch Vehicles*. PhD thesis, Politecnico di Milano, 2012.

P. Catalano, M. Marini, A. Nicoli, and A. Pizzicaroli. Cfd contribution to the aerodynamic data set of the vega launcher. *Journal of Spacecraft and Rockets*, 44(1):42–51, 2007.

YT Chen, JH Nie, BF Armaly, and Hsuan-Tsung Hsieh. Turbulent separated convection flow adjacent to backward-facing step effects of step height. *International Journal of Heat and Mass Transfer*, 49(19):3670–3680, 2006.

HK Cheng. Hypersonic shock-layer theory of the stagnation region at low reynolds number. In *Proceedings of the 1961 Heat Transfer and Fluid Mechanics Institute*, pages 161–175. Stanford Univ. Press Chicago, 1961.

W. J. Coirier, J. Stutts, and C. Rosema. Development of a transonic fin aerodynamic database for incorporation into missile datcom. In *32nd AIAA Applied Aerodynamics Conference*, 2014.

EngelenF.M. Quantitative risk analysis of unguided rocket trajectories. Master's thesis, TU Delft, 2012.

ESA. Vega, 2016. URL http://www.esa.int/spaceinimages/Images/2016/11/Artist_s_view_of_Vega_C. [Online; accessed November 23, 2016].

Alexander Forrester, Andras Sobester, and Andy Keane. *Engineering design via surrogate modelling: a practical guide*. John Wiley & Sons, 2008.

Alexander IJ Forrester, András Sóbester, and Andy J Keane. Multi-fidelity optimization via surrogate modelling. In *Proceedings of the royal society of london a: mathematical, physical and engineering sciences*, volume 463, pages 3251–3269. The Royal Society, 2007.

Peter Fortescue, Graham Swinerd, and John Stark. *Spacecraft systems engineering*. John Wiley & Sons, 2011.

M. Ghoreyshi, K. J. Badcock, and M. A. Woodgate. Integration of multi-fidelity methods for generating an aerodynamic model for flight simulation. In *46th AIAA Aerospace Sciences Meeting and Exhibit*, 2008.

M. Ghoreyshi, K. J. Badcock, and M. A. Woodgate. Accelerating the numerical generation of aerodynamic models for flight simulation. *Journal of Aircraft*, 46(3):972–980, 2009.

Anthony A Giunta. *Aircraft multidisciplinary design optimization using design of experiments theory and response surface modeling methods*. PhD thesis, virginia polytechnic institute and state university, 1997.

R. M. Hall, S. X. Bauer, and J. A. Blevins. Aerodynamic characterization of the ares launch vehicles. *Journal of Spacecraft and Rockets*, 49(4):558–563, 2012.

Robert M Hall, Robert T Biedron, Douglas N Ball, David R Bogue, James Chung, Bradford E Green, Matthew J Grismer, Gregory P Brooks, and Joseph R Chambers. Computational methods for stability and control (comsac): the time has come. *AIAA paper*, 6121, 2005.

Walter Edward Hammond. *Design methodologies for space transportation systems*. Aiaa, 2001.

Zhong-Hua Han, Stefan Görtz, and Ralf Zimmermann. Improving variable-fidelity surrogate modeling via gradient-enhanced kriging and a generalized hybrid bridge function. *Aerospace Science and Technology*, 25(1):177–189, 2013.

Ruichen Jin, Wei Chen, and Agus Sudjianto. An efficient algorithm for constructing optimal design of computer experiments. *Journal of Statistical Planning and Inference*, 134(1): 268–287, 2005.

Donald R Jones. A taxonomy of global optimization methods based on response surfaces. *Journal of global optimization*, 21(4):345–383, 2001.

D Kbiob. A statistical approach to some basic mine valuation problems on the witwatersrand. *Journal of Chemical, Metallurgical, and Mining Society of South Africa*, 1951.

Marc C Kennedy and Anthony O'Hagan. Predicting the output from a complex computer code when fast approximations are available. *Biometrika*, 87(1):1–13, 2000.

Q Ye Kenny, William Li, and Agus Sudjianto. Algorithmic construction of optimal symmetric latin hypercube designs. *Journal of statistical planning and inference*, 90(1):145–159, 2000.

S. Keshmiri, R. Colgren, and M. Mirmirani. Development of an aerodynamic database for a generic hypersonic air vehicle. In *Collection of Technical Papers - AIAA Guidance, Navigation, and Control Conference*, volume 5, pages 3978–3998, 2005.

K. Kitamura, S. Nonaka, K. Kuzuu, J. Aono, K. Fujimoto, and E. Shima. Numerical and experimental investigations of epsilon launch vehicle aerodynamics at mach 1.5. *Journal of Spacecraft and Rockets*, 50(4):896–916, 2013.

D. L. Knill, A. A. Giunta, C. A. Baker, B. Grossman, W. H. Mason, R. T. Haftka, and L. T. Watson. Response surface models combining linear and euler aerodynamics for supersonic transport design. *Journal of Aircraft*, 36(1):75–86, 1999.

Johan C Kok. Resolving the dependence on freestream values for the k-turbulence model. *AIAA journal*, 38(7):1292–1295, 2000.

D. Landman, J. Simpson, D. Vicroy, and P. Parker. Response surface methods for efficient complex aircraft configuration aerodynamic characterization. *Journal of Aircraft*, 44(4): 1189–1195, 2007.

K. Langone and L. Bermüdez. Use of cfd to rapidly characterize aerodynamics for a hammerhead launch vehicle. In *Collection of Technical Papers - AIAA Applied Aerodynamics Conference*, 2009.

J. Laurenceau and P. Sagaut. Building efficient response surfaces of aerodynamic functions with kriging and cokriging. *AIAA Journal*, 46(2):498–507, 2008.

David W Levy, Kelly R Laflin, Edward N Tinoco, John C Vassberg, Mori Mani, Ben Rider, Christopher L Rumsey, Richard A Wahls, Joseph H Morrison, Olaf P Brodersen, et al. Summary of data from the fifth computational fluid dynamics drag prediction workshop. *Journal of Aircraft*, 51(4):1194–1213, 2014.

Søren Nymand Lophaven, Hans Bruun Nielsen, and Jacob Søndergaard. Dace-a matlab kriging toolbox, version 2.0. Technical report, 2002.

C. Luo, Z. Hu, S. . Zhang, and Z. Jiang. Adaptive space transformation: An invariant based method for predicting aerodynamic coefficients of hypersonic vehicles. *Engineering Applications of Artificial Intelligence*, 46:93–103, 2015.

MN Macrossan. Scaling parameters for hypersonic flow: correlation of sphere drag data. In *25th International Symposium on Rarefied Gas Dynamics*, volume 1, pages 759–764. Siberian Branch of the Russian Academy of Sciences, 2007.

Georges Matheron. Principles of geostatistics. *Economic geology*, 58(8):1246–1266, 1963.

Michael D McKay, Richard J Beckman, and William J Conover. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 42(1):55–61, 2000.

Douglas C Montgomery. *Design and analysis of experiments*. John Wiley & Sons, 2008.

F. G. Moore, R. M. McInville, and T. C. Hymer. Evaluation and improvements to the aeroprediction code based on recent test data. *Journal of Spacecraft and Rockets*, 37(6): 720–730, 2000.

E. A. Morelli and R. DeLoach. Wind tunnel database development using modern experiment design and multivariate orthogonal functions. In *41st Aerospace Sciences Meeting and Exhibit*, 2003.

Max D Morris and Toby J Mitchell. Exploratory designs for computational experiments. *Journal of statistical planning and inference*, 43(3):381–402, 1995.

Raymond H Myers, Andre I Khuri, and Walter H Carter. Response surface methodology: 1966-1986. Technical report, DTIC Document, 1986.

Anthony C Nelson, Gregory Z McGowan, and Frank G Moore. Cfd database for the development of a non-linear missile rolling moment model. In *53rd AIAA Aerospace Sciences Meeting*, 2015.

A. Nicolì, B. Imperatore, M. Marini, P. Catalano, A. Pizzicaroli, and D. Perigo. Ground-to-flight extrapolation of the aerodynamic coefficients of the vega launcher. In *Collection of Technical Papers - 25th AIAA Aerodynamic Measurement Technology and Ground Testing Conference*, volume 2, pages 1155–1167, 2006.

NUMECA. Fine/open v2. 12 (including openlabs) flow integrated environment. *NUMECA International, Brussels*, 2012.

E Perez. Vega user's manual. *ARIANESPACE, issue*, 3:154, 2006.

Stephen B Pope. Turbulent flows, 2001.

T. Rajkumar, C. Aragon, J. Bardina, and R. Britten. Prediction of aerodynamic coefficients for wind tunnel data using a genetic algorithm optimized neural network. In *Management Information Systems*, volume 6, pages 473–487, 2002.

Arthur Rizzi, Adam Jirásek, John E Lamar, Simone Crippa, Kenneth J Badcock, and Okko J Boelens. Lessons learned from numerical simulations of the f-16xl aircraft at flight conditions. *Journal of Aircraft*, 46(2):423–441, 2009.

Jerome Sacks, William J Welch, Toby J Mitchell, and Henry P Wynn. Design and analysis of computer experiments. *Statistical science*, pages 409–423, 1989.

T. W. Simpson, J. D. Peplinski, P. N. Koch, and J. K. Allen. Metamodels for computer-based engineering design: Survey and recommendations. *Engineering with Computers*, 17 (2):129–150, 2001.

Timothy W Simpson, Jesse Peplinski, Patrick N Koch, and Janet K Allen. On the use of statistics in design and the implications for deterministic computer experiments. *Design Theory and Methodology-DTM'97*, pages 14–17, 1997.

T. J. Sooy and R. Z. Schmidt. Aerodynamic predictions, comparisons, and validations using missile datcom (97) and aeroprediction 98 (ap98). *Journal of Spacecraft and Rockets*, 42 (2):257–265, 2005.

C. Y. Tang, K. Gee, and S. L. Lawrence. Generation of aerodynamic data using a design of experiment and data fusion approach. In *43rd AIAA Aerospace Sciences Meeting and Exhibit - Meeting Papers*, pages 4341–4349, 2005.

Henri Theil and Henri Theil. Principles of econometrics. Technical report, 1971.

J Tihon, J Legrand, and P Legentilhomme. Near-wall investigation of backward-facing step flows. *Experiments in Fluids*, 31(5):484–493, 2001.

Hsue-Shen Tsien. Superaerodynamics, mechanics of rarefied gases. *Journal of the Aeronautical Sciences*, 2012.

Tyler Skrabek. Rockets of human spaceflight, 2014. URL http://www.universetoday.com/118742/the-worlds-rockets-to-scale/#. [Online; accessed September 26, 2016].

ULA. Atlas v family, 2016. URL http://www.spaceflightnow.com/atlas/av045/infosheet.html. [Online; accessed September 26, 2016].

Selvakumar Ulaganathan, Ivo Couckuyt, Francesco Ferranti, Eric Laermans, and Tom Dhaene. Performance study of multi-fidelity gradient enhanced kriging. *Structural and Multidisciplinary Optimization*, 51(5):1017–1033, 2015.

Felipe AC Viana, Gerhard Venter, and Vladimir Balabanov. An algorithm for fast optimal latin hypercube design of experiments. *International journal for numerical methods in engineering*, 82(2):135–156, 2010.

Frank M White. Fluid mechanics. 5th. *Boston: McGraw-Hill Book Company*, 2003.