Report on the internship carried out from March 29 to August 31

In the society: Bejing steelflow Electromagnetic Ltd

in Beijing

Numerical simulation

SFEM
北京钢流电磁

Xiyi Li

student in

Master2 Modeling and Solid Mechanical Simulation

Year 2021

At University Paris-Est Créteil

UPEC
Connaissance - Action
UNIVERSITÉ
PARIS-EST CRÉTEIL
VAL DE MARNE

The Supervisor: Dr. Jin Li

The UPEC referent teacher : Vu-Hieu Nguyen

SUMMARY

<u>THANKS</u>

Before any development on this professional experience, it seems appropriate to begin this internship report with thanks, to those who taught me a lot during this internship, and even to those who were kind enough to make this internship a very enriching.

Also, I thank Doctor Jin Li, my internship supervisor who trained and supported me throughout this professional experience with a lot of patience and pedagogy. Finally, I would like to thank all the employees for the advice they were able to give me during these five months.

INTRODUCTION

Summary of the internship

Host organization: Beijing Steelflow Electromagnetic Ltd

Internship duration: Five months

My internship concerns the optimization of Fortran code. This code can calculate the approximate values of physical quantities such as the characteristics and the power of the axial or centrifugal compressor when the corresponding parameters are entered in the code. The code is divided into two parts, which respectively deal with the problems of 1D and 2D compressors. In the 1D part, there is a part to calculate the characteristics of the compressor, which is the most important and difficult part of all the code.

I need to understand the code and change the source code from format 77 to format 90, then write a document to explain the logic and utility of subroutines. During the internship I read and tried to understand the code, optimizing some equations, removing unnecessary variables, and rewriting some of the irrelevant logic loop. The variables in the code are mostly named by Russian abbreviations, I did not fully understand the physical meaning of each variable. Finally, I wrote a document to explain the logic of the code in part 1D, and I modified the code of the characteristic part and optimized its structure.

During the internship, I acquired theoretical knowledge about compressors, I know more about FORTRAN, using company resources, I acquired several skills using various software tools (e.g. Python, etc.)

Announcement of the plan

The follow-up report is divided into three parts, the first part, I present my internship company, and in the second part, I will present the objectives, the progress and the results of the internship, finally, I will make a summary.

I / THE COMPANY AND ITS SECTOR OF ACTIVITY

A. Company history

Steel Flow Electromagnetic (SFEM) is a company that uses mathematical models to help customers solve practical engineering problems. The company was founded in 2013, it has decades of experience in the field of numerical computation. Its scope of activity covers the metallurgical industry, electrical energy, aerospace, nuclear industry, turbine machines, ships, etc. Technological reserves are mainly concentrated in CFDs, electromagnetic calculations, analysis of heat transfers and materials, multi-physics coupling calculations and numerical algorithms.

Over ten years ago, Chinese industry began to modernize, and traditional industries increased the demand for digitization and digital simulation. The company seized this opportunity, it invested a lot of resources in the development of mathematical models, after signing cooperation contracts with several companies, the company participated in the research and development of many important projects.

In order to better present the company where I did my internship, I show you some projects that the company has carried out.

This project is the design of the moving blade of a high temperature and high pressure steam turbine. The development includes the design of the blade shape, the calculation of the aerodynamic efficiency of the flow field and the verification of the vibration intensity. Based on the existing 30 MW grade steam turbine blades, the company has designed 38 MW grade steam turbine blades that meet the requirements of thermal performance, strength and vibration.

The company's engineers modeled and simulated the blade, as shown in Figures 1 and 2.

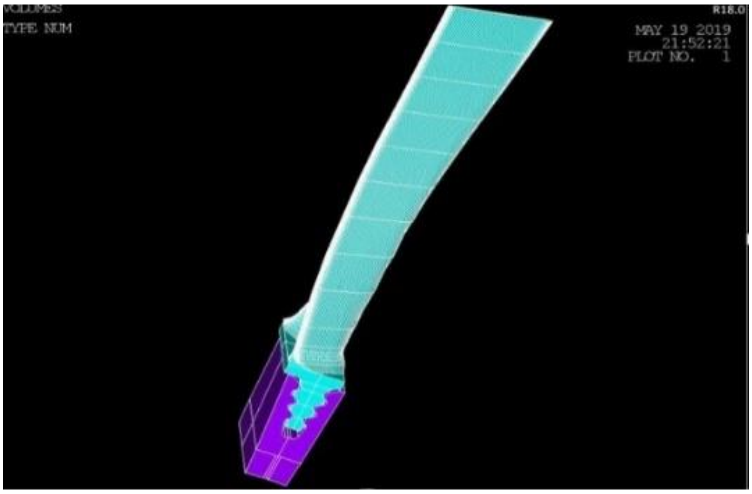


Figure1: Finite element model for the control of the resistance of the blade

In Figure 1, the engineers used the finite element method to calculate the deformation of the blade under the effect of stress.

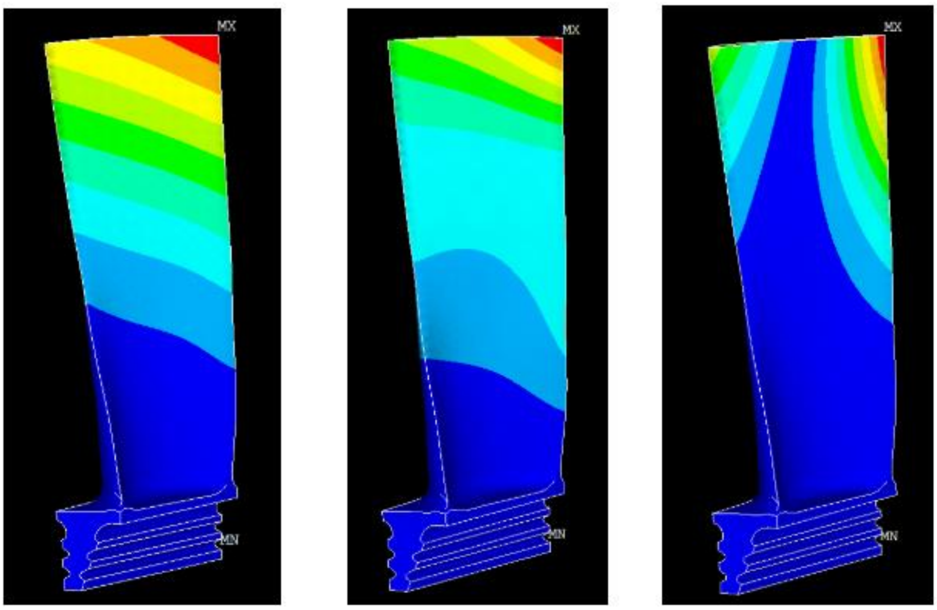


Figure 2: the 1st to 3rd order vibration diagram of the blade

The first order mode is bending vibration, the second order mode is bending vibration with a pitch line, and the third order mode is torsional vibration.

Flow field and aerodynamic calculation and grid construction. The grid is a hexahedral structured grid.

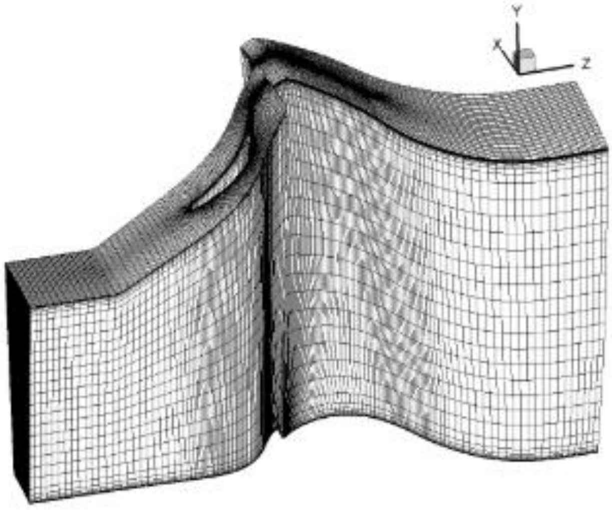


Figure3: grid CFD

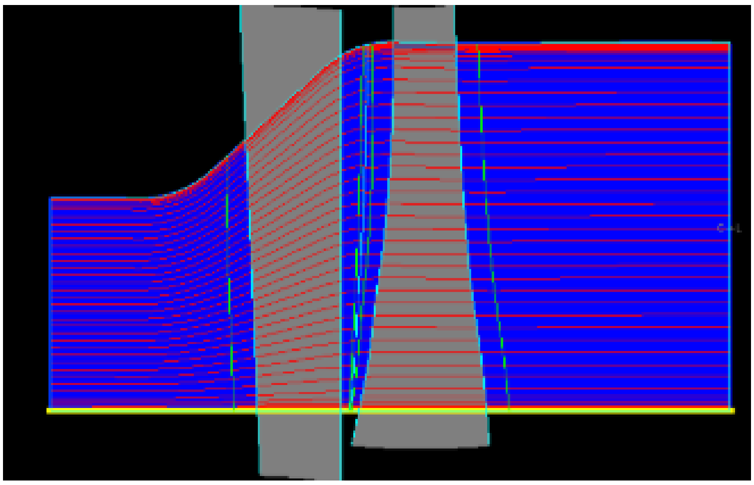


Figure 4: Distribution of the grid in the direction of the surface S2

By applying the above blade to actual engineering projects, the customer has fully realized the independent design, manufacture and service of 38MW Guard Steam Turbine products.

B. The company today

Today, the company is in a phase of rapid development and the company has recruited many students who have studied abroad, they have degrees in solid mechanics, fluid mechanics, aerodynamics and other specializations. In the future, the company will always focus on the development of mathematical models and numerical simulations, the company is also paying attention to the development of intelligent manufacturing. With the advent of the era of 5G and the era of artificial intelligence, the company will set foot in this field.

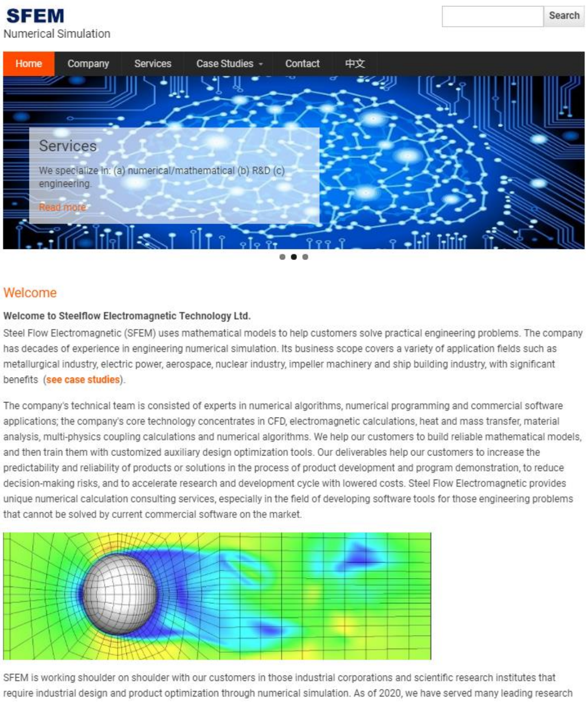


Figure5: Company web page

When the company was established, the company primarily served the Chinese market and its customers were other manufacturing companies, such as steel mills and shipyards. Today the company has cooperated with foreign companies. Recently, the company will recruit a large number of foreign graduates and foreign engineers to enhance overseas business. The current scale of the company is still small, but it is very dynamic and creative.

Going forward, the company plans to recruit more graduates in areas such as algorithms, artificial intelligence and smart manufacturing, expand the size of the company, and reserve young talent.

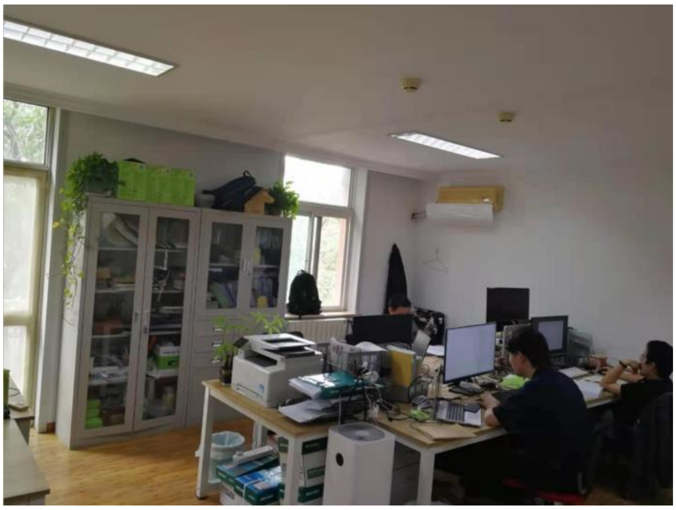
II / THE INTERNSHIP FRAMEWORK

A. Description of the social structure of the company / organization

The sector has four young employees, two experienced employees and an intern. Among them, with the exception of one who obtained a doctorate, the others all obtained a master's degree in science and technology. Among the young employees, two are graduates of the University of Leeds and one is a graduate of the University of California, the two colleagues graduating from the University of Leeds are all in fluid mechanics and more experienced in liquid simulation, a colleague graduated from the University of Leeds University of California in aerodynamics, he is also responsible for guiding my internship, the doctor graduated from the Chinese Academy of Sciences, He is an expert in applied mathematics and physics.

Every Friday, the company will hold a small, regular meeting. The theme and content of the meeting is not limited, the purpose of the regular meeting is to enable employees to communicate, so that employees can acquire new knowledge in communication.

Figure6: My trainee sector



The corporate culture advocates free communication between employees and the free development of the employee, in this company, even if you are a novice, the company will still entrust you with the project, the company considers that the training of new employees is very important, so she does not hesitate to invest resources, even if there is no short-term benefit.

The company has an international student from Pakistan. Its strength is Fortran programming and the calculation of various properties of the fluid. When I encountered difficulties in FORTRAN programming, it helped me solve many bugs.

B. Display of the sector project

In this chapter, I will focus on the projects that are handled by my colleagues in order to better demonstrate the daily work of the company, I will show some examples, firstly, I present the Multigrid Method.

When we do simulation processing, meshing is a necessary process, the mesh size is directly related to the accuracy of the simulation results, the result calculated using a fine mesh is more accurate than a mesh coarse, but fine size requires many iterations, although coarse mesh is not accurate due to fine mesh, but it saves the computational cost. Therefore, by combining the two mesh sizes, the multigrid method is widely used in practical engineering problems.

The characteristics of Multigrid are as follows:

(a) Different meshes can reduce errors; the fine mesh reduces the short wavelength error, the coarse mesh reduces the long wavelength error. A correct combination of the two can reduce errors on both scales in a single cycle.

(b) The shape functions associated with a node common to the coarse mesh-H and the fine mesh-h show how information (example speed, stress or displacement etc) of different distances can be brought to a node given using different mesh scales.

(b) We can design different schemes to iterate through the different meshes.

We use different grids for an iterative process which makes it possible to get rid of low wavenumber residuals with a smaller number of iterations, as shown in Figure 7.

Based on existing multigrid articles, my colleague developed a FORTRAN code which automatically generates the multigrid. Using this code, he was able to solve engineering problems.

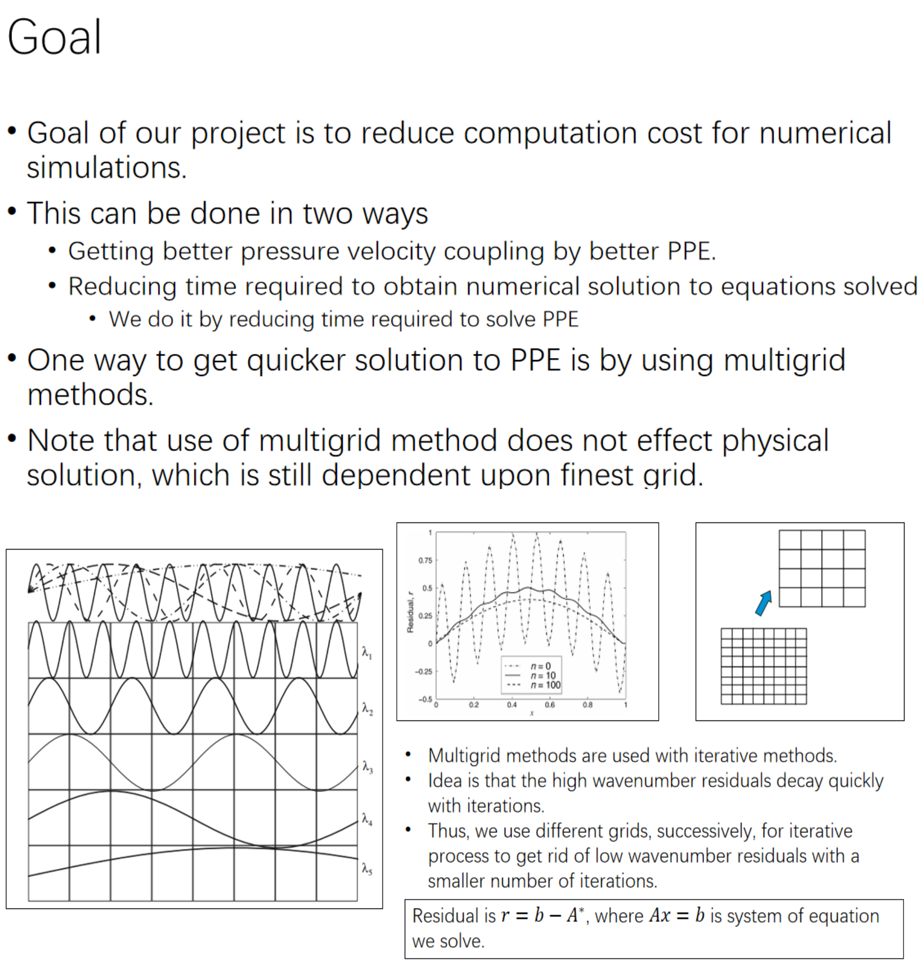


Figure7: Conclusions on The Multigrid Method

The company participates in an aircraft manufacturing work with another company, our company has to research the engines that were chosen by the customer. Next, the company should select a suitable engine, conduct research on this engine and show customers the performance and economy of showing, The following figure is a cross-sectional view of a structure of an engine, my colleague will show customers the process of running this engine.

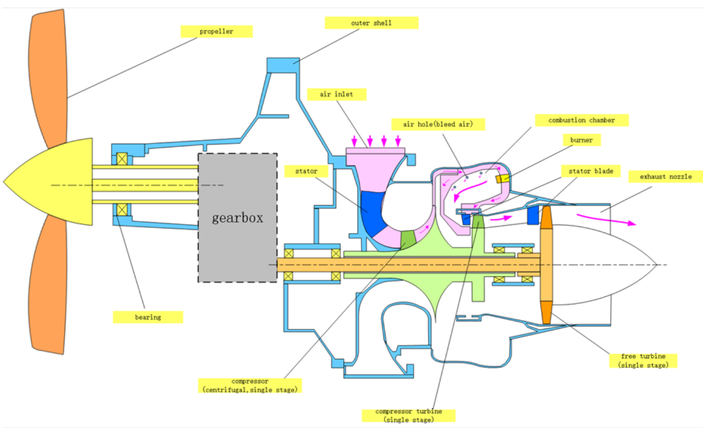


Figure 8: Sectional view of the motor

III / WORK CARRIED OUT AND CONTRIBUTIONS OF THE INTERNSHIP

A - The work carried out

1. The tools at my disposal

In the first week I learned the latex tool and used latex to write the project document. After that, I installed the necessary software tools to write Fortran code, and used the ifort file to compile the Fortran code.

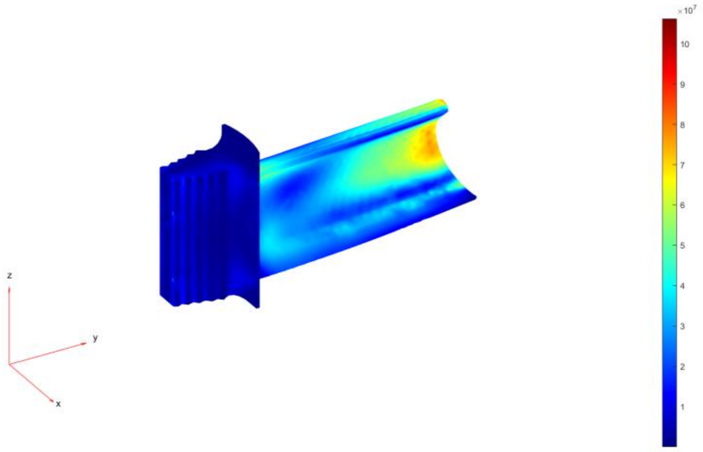


Figure12: Example of analyzing a structure using the finite element method

In figure 12, I used the application's toolbox to build the model, then build the mesh, finally use the finite element method to solve the deformation of the structure under pressures.

2. Missions of the position held

2.1 Fortran code optimization

During the internship, my main job is to optimize and modify the code. This code uses the old fortran77 format, the main function of the code is to calculate the approximate values of the physical quantities such as the characteristics and the power of the compressor. The source code I received has been modified by other programmers. The format of the code mixes the fortran 77 format and the fortran90 format. The code is divided into 4 files, the main program: Main; the variable file: Var; the subroutine file: Routine and the input and output file: IO.

Figure13: the code files

In the Main program, it is divided into 2 parts, one part calculates the compressor structure values such as the size of the compressor input working wheel, the optimum installation angle of the compressor blades in the 1D situation. The second part consists in calculating the characteristic value of the compressor, for example the input speed of the compressor, the output speed, the flow rate etc., the number of lines of code in this part represents about 70% of the code. The total number of lines of code is approximately 7,000 lines.

VAR	2021/7/16 星期五 11:...	F90 文件	6 KB
IO	2021/7/16 星期五 11:...	F90 文件	18 KB
MAIN	2021/7/16 星期五 17:...	F90 文件	181 KB
ROUTINE	2021/7/16 星期五 13:...	F90 文件	26 KB

When calculating the characteristic part, the output data of the 1D part is needed, so the program must first execute the first part, the 1D and characteristic data will be saved in the AX file.

Figure 14: Code selection interface

Figure 15: Code selection interface after 1D

管理员: Intel Parallel Studio XE 2015 Composer Edition Intel(R) 64

```
-----------^
MAIN.F90(2372): warning #7148: Arguments incompatible with intrinsic procedure, assume EXTERNAL.   [E
      CALL EXIT(HZ,HZ1,C1A,PIK,QLA1,&
-----------^
MAIN.F90(5992): warning #7532: The number of arguments is incompatible with intrinsic procedure, assu
       CALL EXIT(HZ,HZ1,C1A,PIK,QLA1,&
------------^
MAIN.F90(5992): warning #7148: Arguments incompatible with intrinsic procedure, assume EXTERNAL.   [E
       CALL EXIT(HZ,HZ1,C1A,PIK,QLA1,&
------------^
Microsoft (R) Incremental Linker Version 12.00.21005.1
Copyright (C) Microsoft Corporation.  All rights reserved.

-out:1Dnew.exe
-subsystem:console
VAR.obj
IO.obj
ROUTINE.obj
MAIN.obj
   IR=1: design
   IR=2: characteristic
   IR=
```

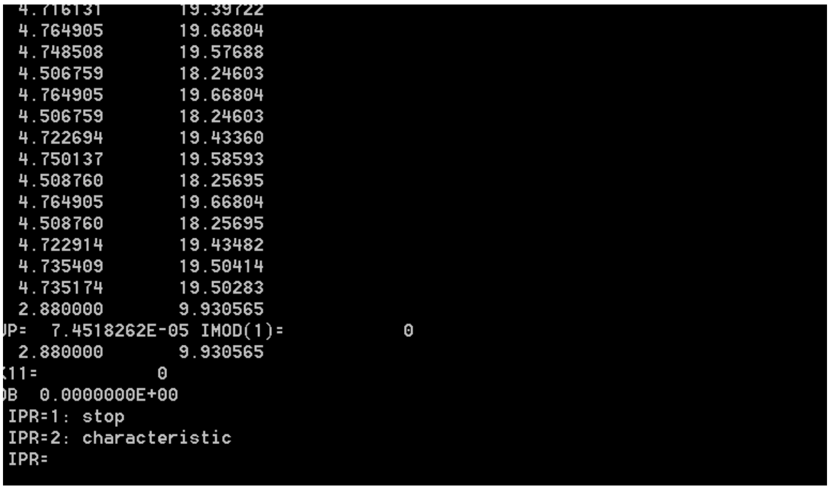


Figure16: The AX file

My main task is to understand the code and optimize the code, the code is from Russia, so the comments in the code are all in Russian. In the first month of my internship I understood the code of 1D, after that I started to write an explanation document, the document included an explanation of the logic of the 1D code and pseudo-codes also after writing the document i started to edit the code, the first thing i changed were the main program variables, in the source code the code uses COMMON to create a public area, COMMON only says that the variable assigns in the public area, but does not indicate the type and precision of the variable, the public area uses bytes of memory instead of variable names, this is extremely difficult to troubleshoot and debug.

So I changed the COMMON statement to the MODULE statement in code, and used the USE statement to call the variable. the source code did not use IMPLICIT NONE, so it used many variables directly in the program, I added the IMPLICIT NONE statement and declared the variables that were not declared in the main program.

After I finished modifying the 1D code, I started writing the report, the document includes the explanation of the code logic, pseudo-code and the explanation of the characteristic parameters of the compressor.

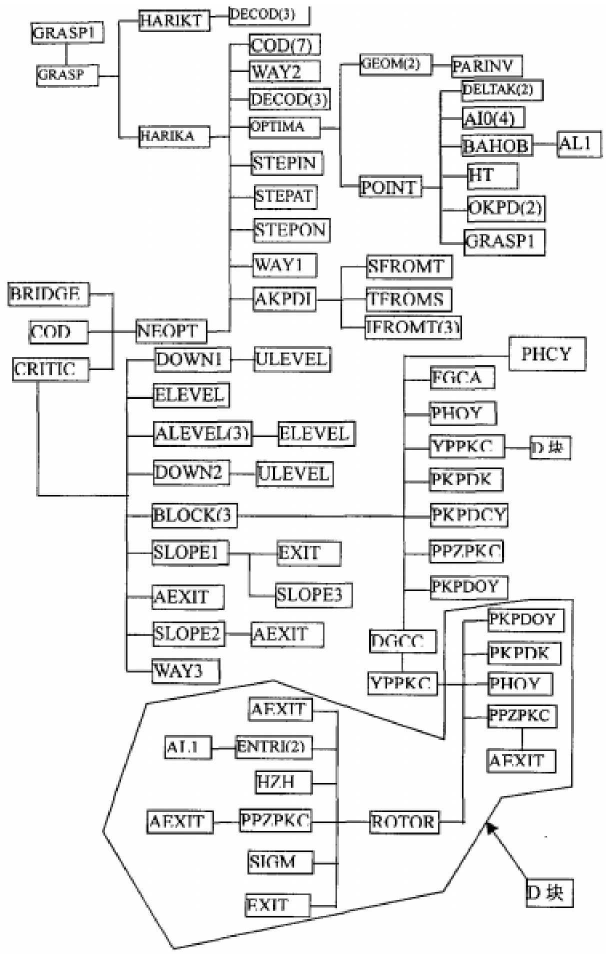


Figure 17: Program call diagram of the feature part

Figure 17 shows us the logic of the characteristic part of the code, we see that to carry out the characteristic calculation of the compressor, the program uses a lot of subroutines, and the logic of the code is complex, the code has many branches, this shows that the code of the characteristic part is very important.

I also wrote the important routines in pseudo-code format, which has the advantage of being modular, simple and clear, and easy to show to clients. Examples are as follows:

4.5 PCPBYX - 计算末级尺寸与轴向速度

输出变量	符号	描述	单位
HZK	L_u	实际功	J/kg
TB	T_b^*	出口总温	K
DBTB	$\bar{d}_b$	出口轮毂比	–
DPB	D_{tb}	末级外径	m
DCB	D_{mb}	末级中径	m
DBB	D_{hb}	末级内径	m
CAB	c_{ab}	出口轴向速度	m/s

4.5.1 等熵功实际功计算

用增压比即可求出出口总熵，出口总温，出口总焓，因为这部分计算未包括能量损失，故计算出来的结果为等熵功。

$$s_b^* = s^* + Rln(\frac{P_b^*}{P^*}) \tag{49}$$

$$T_b^{*\prime} = TXP(s_b^*) \tag{50}$$

$$h_b^{*\prime} = XHA(T_b^*) \tag{51}$$

$$L_{ad} = h_b^* - h^* \tag{52}$$

定义效率(η)初值为0.85来计算实际功，实际出口总焓与实际出口总温。

$$L_u = \frac{L_{ad}}{\eta} \tag{53}$$

$$h_b^* = h_{in}^* + L_u \tag{54}$$

$$T_b^* = TXH(h_b^*) \tag{55}$$

轴向速度估值，再通过轴向速度估值所计算的流通面积与实际尺寸计算的流通面积进行对比。若相同则退出子程序，否则调整轴向速度，直到计算流通面积与已知尺寸相同。

Algorithm 4 末级轴向速度计算

for do
  $c_{ab} = \frac{1.05\dot{m}RT_b^*}{A_b P_b^*}$
  $c_b = \frac{c_{ab}}{\sin\alpha_b}$
  $A_b^{\prime} = \frac{RT_b\dot{m}}{Pc_b}$
  if $abs(A_b - A_b^{\prime}) <= 0.0001 or A_b = A_b^{\prime}$ **then**
    RETURN
  end if
  if $A_b^{\prime} < A_b$ **then**
    $c_{ab} = c_{ab}\frac{A_b}{A_b^{\prime}}$
  else
    $c_{ab} = c_{ab}\frac{A_b^{\prime}}{A_b}$
  end if
end for

Figure18: Pseudo-code of part 1D of the code

The table of figure 18 shows the output variables which will be used in the following programs, in chapter 4.5.1 are the formulas for calculating the real isentropic work, finally, the pseudo code to calculate the axial speed of the compressor output, the rest of the document is similar to this part, after showing it to the client, the client was very satisfied, for me it was a good exercise, it gave me an idea of how to write explanatory documents that orient towards the customer.

Before the internship, I thought that the document was only a tool to make a summary or a presentation, but during the internship my colleagues told me that a good document is important, each client has their own requirements in document matter, the right documents can not only meet the needs of the customer, but also create a communication bridge between people, a good document equals good communication, good communication between people is very difficult, writing documentation is therefore also a very important job.

After finishing the documents I started to modify the code in the feature part, the lines of code in this part are about 70% for total lines of code, this part contains a lot of subroutines, as shown in the figure 19.

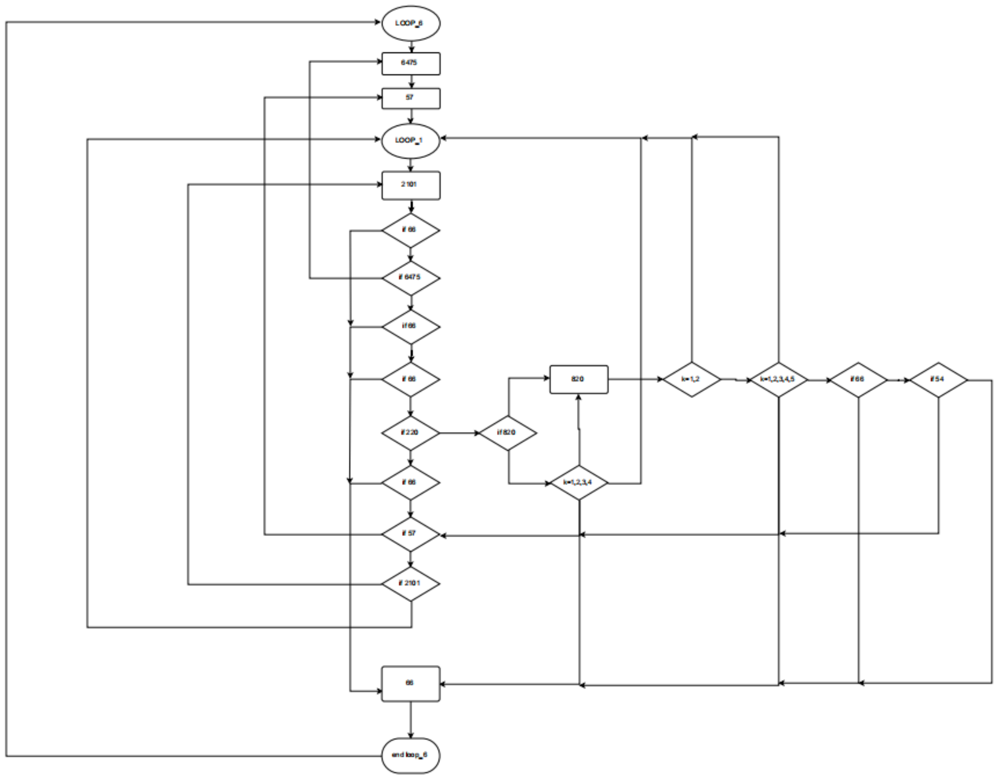


Figure 19: Logic diagram of the characteristic part of the code

In figure 19, the diamond represents a selection (IF or GOTO), the rectangle represents the content of the code and the circle represents the start of the loop or the end of the loop, we can see the complexity of this part of the code from figure 19, we have seen that the stability of the code is not good, due to the complexity of the code, the code is very sensitive to changes in the values of the variables. The source code did not declare double precision for the variables, so when we declare the double precision variables, the code will have bugs, in order to eliminate the bugs, we have optimized the logic of the code, for the error systematic code, the error range should be 0.0001, and the number of errors can not be much, as shown in figure 20, in figure 20, the data after code modification is on the left, we can see that the errors are all 0.0001 and the total number of errors is 17, there is approximately 2000 output data, so this number of errors is acceptable.

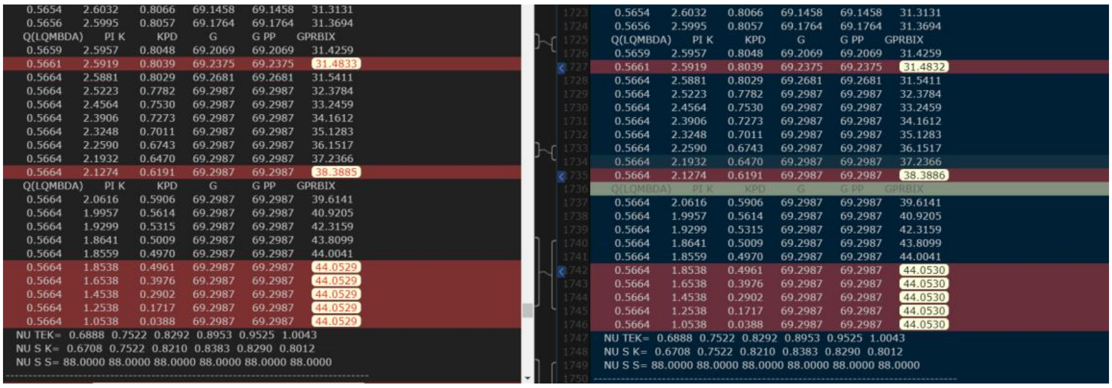


Figure 20: Comparison of results after and before code modification

In the work of modifying the code, I also learned a lot about compressors. The compressor can generate pressure difference and are widely used in airplanes and ships etc. Compressor vanes are very important, when other conditions remain unchanged, when the compressor efficiency is highest, the vane angle at this time is called the optimum vane angle.

The compressor pipe is generally irregular, as follows, this is a side view of the compressor pipe. it can be seen that the compressor inlet on the left is larger, while the compressor outlet on the right is narrower. For an axial compressor with coaxial rotation, the best vane angle at the inlet and outlet is different, for the working rotors and guide stators at all stages of the compressor, the vane angle is different. also, what is depends on the summary of the experience.

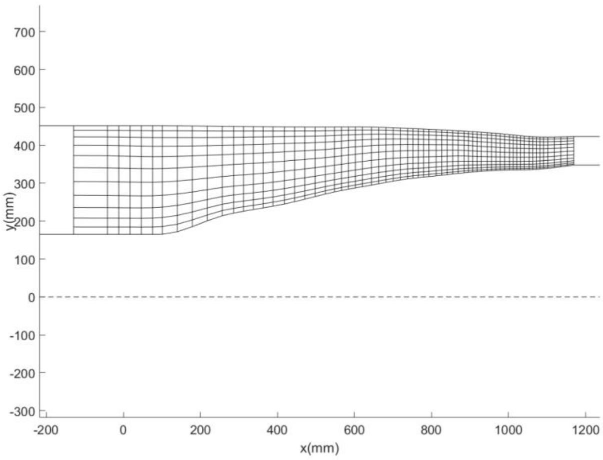


Figure 21: Compressor channel diagram

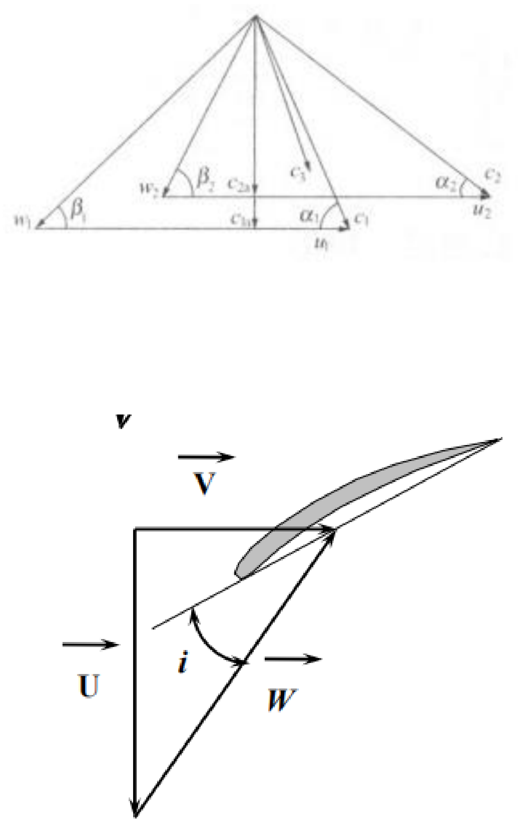


Figure 22: Speed triangle

Figure 22 shows us the speed triangle, the speed triangle is important to analyze the flow inside the compressor, the energy transfer process and the changes in the size and direction of the flow, the speed triangle graph can describe the type of blade, for the flow through the stator the absolute speed is important, for the flow through the rotor one has to take into account the relative speed of the rotating blades, which is called relative velocity, when the relative velocity and other related parameters are used, the flow in the row of rotating blades can be treated as the flow in a static flow channel.

The internship experience on compressors and FORTRAN codes is very important to me. I not only learned new knowledge, but I also learned how to organize work plans. My colleagues explained to me how to distribute the working time, manage the difficulties encountered, etc. these are all very important experiences, which have greatly helped me to improve my overall quality.

2.2 study the ant colony algorithm

In this part, I present the ant colony algorithm, the ant colony algorithm is a bionic algorithm, it simulates the natural method of ant path finding, an ant releases pheromones during movement and others can detect the pheromone to determine the direction of its movement. Under the condition of the same speed of the ants, more pheromones will be left on the short road. Therefore, the collective behavior of a large number of ants is a phenomenon of positive feedback.

The Ant Colony Algorithm is a new evolutionary bionic algorithm in the field of optimization. the algorithm uses a distributed parallel computing system, which is easy to combine with other methods, It has a strong robustness, but its main drawback is the long search time and easily limit the optimal local solution.

In a practical example I am using this algorithm to solve the problem of the extreme values of the equation, here is the equation we need to solve:

$$f(x,y) = 20(x^2 - y^2)^2 - (1-y)^2 - 3(1+y)^2 + 0.3$$

$$x, y \in [-5,5]$$

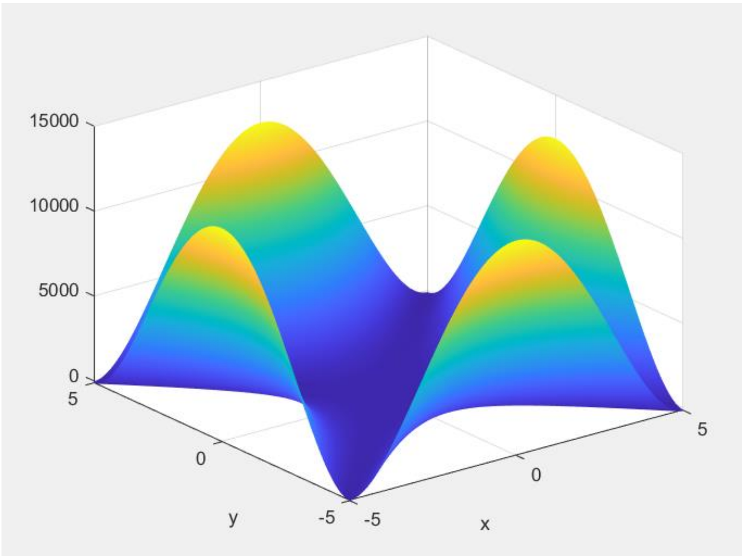
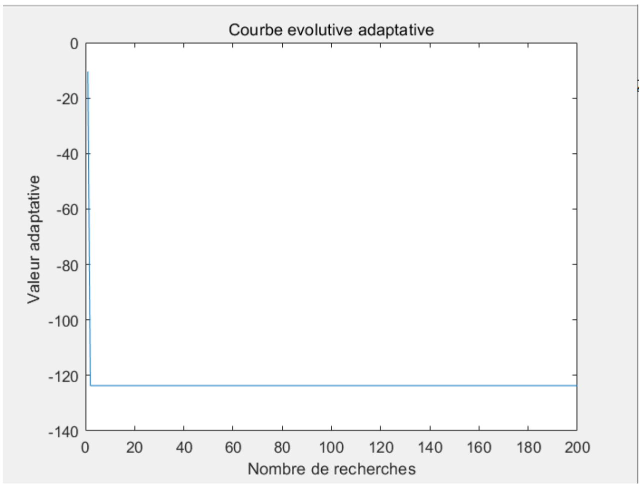
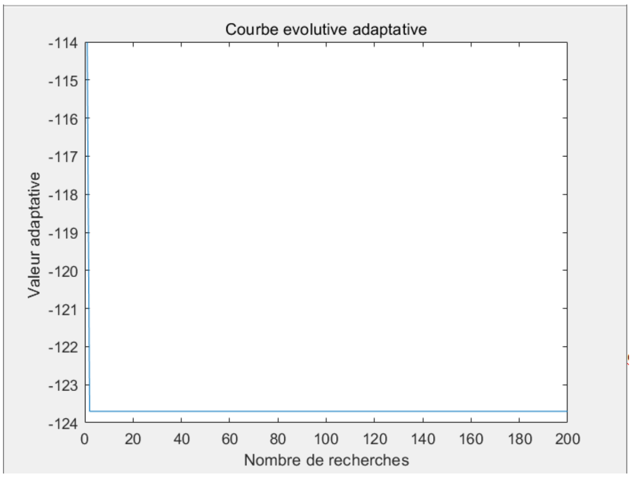


Figure 23: 3D diagram of the equation

In the code, it is assumed that there are 20 ants scattered randomly on the plane formed by the equation, every time the program is rerun, the position of the ant changes, this can ensure the correctness of the code.

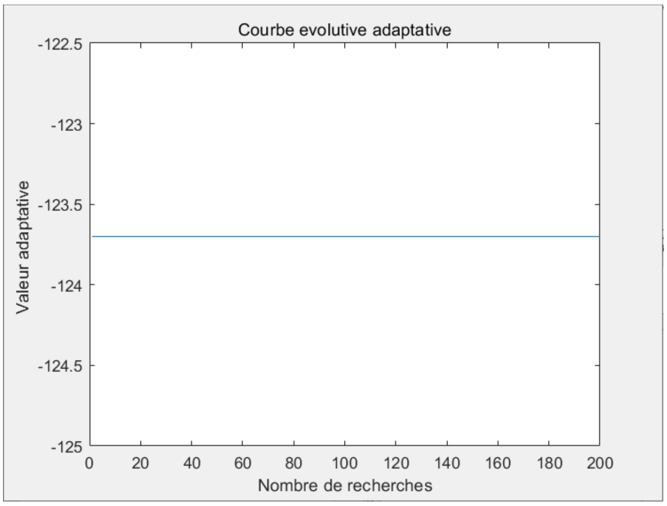


Results 1

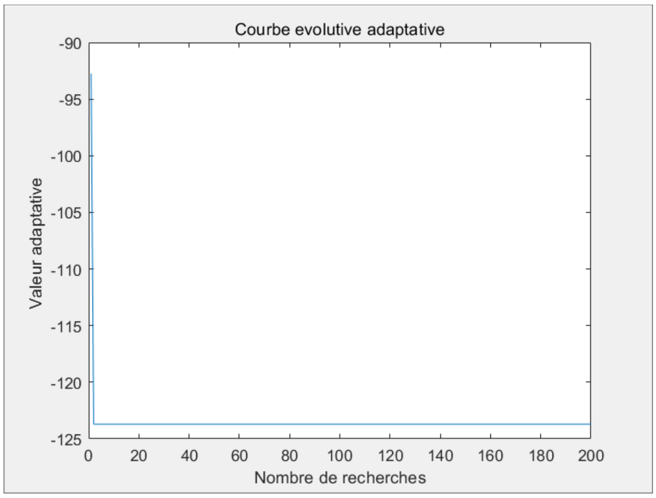


Results 2

Figure 24: Results of the ant colony algorithm 1 and 2

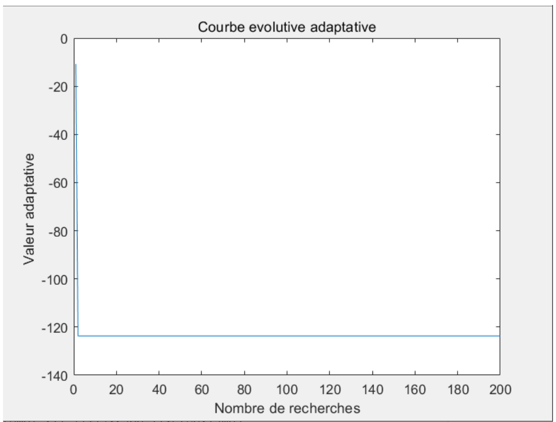


Results 3



Results 4

Figure25: Résultats de l'algorithme de colonie de fourmis 3 et 4



Résultats 5

Figure 26: Results of ant colony algorithm 5

We get 5 different figure, the figures are called the short adaptive evolutionary, where the x coordinate is the number of searches, and the y coordinate is the adaptive value, in our problem, the adaptive value is the limit value, here we calculate the minimum limit value, the curve shows the number of searches necessary for the code to find the extreme value, this simulates the number of steps necessary for the ants to find the best path, in our results, the result 5 is very close to the result 1, it maybe that the initial positions of the ants are close, or the initial positions of several ants are in the same area, these two situations will lead to similar results, the curve of the result 3 is a straight line, which is a special case, that is, at first more ants are distributed to the area near the limit point, due to the positive feedback effect, the program will directly determine t the solution, in addition to figure 3, we can see that the program found the optimal solution after less iterations, but that's because the range of x and y we set is small, when we increase the definition range of x and y, the program needs more iterations to find the optimal solution.

The important parameters in the ant colony algorithm are the empirical coefficients, when the definition range of x or y becomes larger, we only have to modify the parameters at the same time, for example, the number of ants, when we define the range of x and y from 100 to -100, if the number of ant parameters is too small, the optimal solution will not be found and the calculated value will be different each time, the code has the problem of falling into the local optimality, as shown in the following figure:

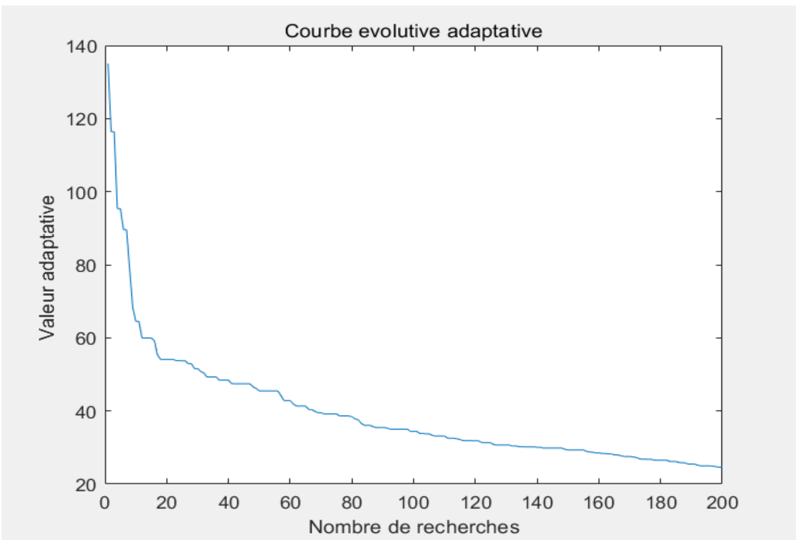


Figure 27: Results of falling into local optimality

In order to solve this problem, the code can be optimized or combined with other algorithms.

3. Difficulties Encountered and Solutions Provided

In the work of modifying the code, I was stuck in the part of calculating the characteristics of the compressor, I modified the variables of this part of the code and succeeded in compiling, but the program could not run, the reason could be an error in the assignment of the variable, or an implicit bug. For the understanding of the code, I don't know about compressors so the code is more difficult for me to understand, many formulas are used in the code, a lot of these formulas are obtained by experiments, which makes it hard to know the physical meaning of formulas.

The first time i changed the code i changed all the variables but the program couldn't run, due to i changed a lot of code, i don't know how to check the error, after trying to fix the errors I failed. When I changed the code the second time I changed the variables one by one, every time I changed a variable I ran the code to check for errors so I found the errors that I made by modifying the code.

```
!COMMON/A38/IND8R
COMMON/A40/C4A
!COMMON/A41/TTI
```

```
!COMMON/A38/IND
COMMON/A40/LA4
!COMMON/C1/III,
!COMMON/C3/HQK,
```

```
 CALL CRITIC(II,HQ,QLA1,C1AO,
&RK,HA,RKI,HAI,STAGE,AN,GB,GK
&INDEX2,INDEX3,C1U,C2A,BETA1,
&LA4,C4A,PI,PIK,PIT,HT,AKPX)
```

Figure 28: Code irregularities

As shown in figure 28, the source code uses the COMMON instruction to create a common area A40, this common area has a variable, which is named C4A in the main HARIKA program, but in some subroutines of HARIKA, this variable is named LA4, this is done to avoid duplicate names, such as the CALL CRITIC code. For the first time I did not find this problem, which made the program not work normally, regarding modifying A40 there are two methods, the first method is to change the names of all C4A variables in HARIKA , the second method is to change the name of the variable LA4 in the subroutines, I chose the first method because the corresponding variables in HARIKA are easier to modify.

In the source program many variables have the same problem, I can only check the subroutines one by one to see which subroutines these duplicate named variables are used in, so the workload is huge in this part , but in order to ensure the correctness of the code, this method is very necessary.

There are also illogical problems in the code, I have seen a lot of GOTO statements in the source code. In order to optimize the logical structure of the code, it is necessary to modify the GOTO instruction, however, when the code is modified correctly and the logic of the code is correct, there are errors between the calculation result of the modified code and the result got before modification, these errors are still 0.0001, I did not find the cause of the errors, but I suspect it may be caused by some computer problem, for example, the data is changing while entering and exiting computer memory, resulting in an error of 0.0001. The source code does not use double precision, when we only changed the variables to double precision, the code could not work, it is very illogical, I think the code is sensitive, when the variable becomes double precision, the number of data bits changes resulting in a small error, and these small errors prevent the program from jumping out of some loops, I did not fix this problem at that time, I have tried other methods, but the problem was not resolved. I also searched the internet for a lot of information, but there was no specific answer to this question.

Colleagues helped me a lot to understand the code. As I said before, many code comments are in russian, so I don't know what it means, to help understand the code, a colleague made a variable table, thanks to this variable table, j Understood the code bit by bit, after that I wrote a document explaining the logic of the code, which deepened my understanding of the logic of the code. In addition to the above issues I made some bad fixes, and an implicit BUG was generated due to my error, with the help of my colleagues I finally fixed the issue.

B - Contributions of the internship

2. Life in society

During the internship, the daily life of the company is also interesting, every Tuesday, the company organizes a lunch, everyone goes to the restaurant and discusses their experiences, through conversation, the atmosphere between colleagues is very relaxed , and interns like me can quickly integrate into the company, every Friday has a regular meeting of the company, on that day, everyone will discuss the progress of the work, the new knowledge acquired or the difficulties encountered in during labor.

For me this is a good opportunity to learn some knowledge, often I am curious about the content of my colleagues' conversations in meeting, then I check the documents to learn related knowledge etc even if these are not not my professional knowledge.

Most of my colleagues are young people who returned to China after studying abroad, so we often share the experience of studying abroad with each other, I found that my colleagues are very interested in knowledge of computer programming and image processing. At work, we often shared the difficulties encountered, sometimes we tried to solve the problem together, during this time I learned how they analyze the problems. Although sometimes we could not solve the problem, but e, I have gained experience anyway.

CONCLUSION

This internship is very rich for me, because it allows me to discover in detail the role of simulation in the field of intelligent manufacturing, and allows me to participate in challenges through different tasks, during the internship, I have gained a lot of knowledge about mathematical models, which can help me deal with problems related to structural optimization, and learned python and other software tools, which can help me develop better in the future.

During the internship, our company is in a period of rapid development. I am proud to be able to make my contribution, participating in the work of the company, I have a better understanding of what I want to do in the future and help me to have a better plan for my future. With this experience, I want to enter the automotive industry in the future, today the traditional automotive industry is transforming into intelligent electronic vehicles, and this industry still needs a large number of CAE engineers.

Finally, this internship experience is a very valuable life experience for me, it made me realize that a lot of knowledge and skills can only be acquired in real work, even if you get into the social you have to keep going to learn and grow. I am very satisfied with my internship experience, I not only learned a lot of knowledge, but also learned how to solve and think about problems when I encounter difficulties, it makes me a great improvement, I am very grateful for this experience of the traineeship.

Reference

[1] Bonabeau E, Dorigo M, Theraulaz G. Swarm intelligence: From natural to artificial systems[M]. New York: Oxford University Press, Inc, 1999

[2] M. Dorigo and G. Di Caro. The Ant Colony Optimization meta-heuristic. In D. Corne, M. Dorigo, and F. Glover, editors, New Ideas in Optimization, McGraw Hill, UK, 1999.

[3] Ines Alaya, Christine Solnon, Khaled Ghedira. Optimisation par colonies de fourmis pour le problème du sac-à-dos multi-dimensionnel. Revue des Sciences et Technologies de l'Information - Série TSI : Technique et Science Informatiques, Lavoisier, 2007

[4] Julien Lainé. Optimisation d'aubes de turbomachines : minimisation de la ré ponse vibratoire à la suite de contacts avec le carter. Génie mécanique [physics.class-ph]. 2017

[5] Millecamps Antoine. Interaction aube-carter: contribution de l'usure de l'abradable et de la thermomécanique sur la dynamique d'aube. Sciences de l'ingé nieur [physics]. laboratoire de mécanique de Lille, Français. 2010

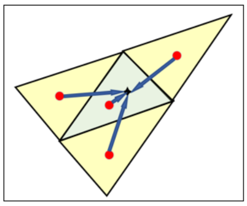
Annexe

Restriction

- Used to obtain residuals at coarser grid.

$$\mathcal{R} r_{fine} = r_{coarse}$$

- Restriction stencil $\mathcal{S}^{\mathcal{R}}$ contains IDs of the fine cell used for purpose of restriction.
- Coefficients of $\mathcal{R}$ are arbitrary. Could be defined by plain sum or weighted average.
- Necessary condition: $\sum \mathcal{R} = 1$

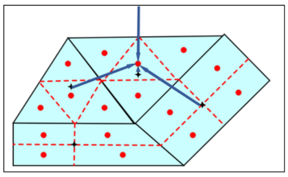


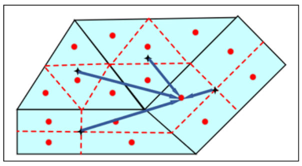
Prolongation

- Used to interpolate 'correction,' say $\boldsymbol{c}$, from coarse grid to fine grid

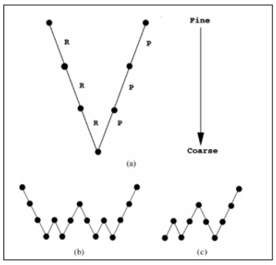
$$\mathcal{P} \boldsymbol{c}_{coarse} = \boldsymbol{c}_{fine}$$

- Prolongation stencil $\mathcal{S}^{\mathcal{P}}$ contains IDs of the coarse cell used for purpose of interpolation.
  - Cell IDs of the coarse cell sharing nearest node are used (with Taylor expansion for interpolation.)
  - Fine cell surrounded by other fine cells can either be interpolated from coarse grid face tensor (with Barycentric interpolation) or injected directly.
- Coefficients of $\mathcal{P}$ are defined by the method used for interpolation .
- Necessary condition: $\sum \mathcal{P}_i = 1$





Elementary theory of multigrid methods



- Assume we want to solve system of equation

$$A\boldsymbol{x} = \boldsymbol{b}$$

- Let, $\boldsymbol{x}^*$, be solution obtained from iterative method on the fine grid. Residual $\boldsymbol{r}$, then, is given by.

$$A(\boldsymbol{x} - \boldsymbol{x}^*) = A\boldsymbol{c} = \boldsymbol{r}$$

- To get rid of low wavenumber residuals, we solve for correction, $\boldsymbol{c}$, on coarser grid. Here, A^c needs be defined for coarse grid.

$$A^c(\boldsymbol{c}) = \mathcal{R}\boldsymbol{r}$$

- This correction is then used to correct $\boldsymbol{x}^*$at the fine grid as

$$\boldsymbol{x}^* = \boldsymbol{x}^* + \mathcal{P}\boldsymbol{c}$$

Multigrid method for PPE (for FVM)

- For FVM, PPE is given as (as obtained before). m is the face ID of i^{th} fine cell.

$$\sum_m A_m \hat{v}_m = \sum_m \frac{\mathcal{C}_m A_m}{d_m}(p_m - p_i)$$

- We need to re-write or re-discretize this equation for coarse grid. n is the face ID of j^{th} coarse cell.

$$\sum_n A_n \hat{v}_n = \sum_n \frac{\mathcal{C}_n A_n}{d_n}(p_n - p_j)$$

- This equation can directly be solved by using coarse grid. $\hat{v}_n$ can be obtained by restriction on coarse cell face.

PPE coarse grid equation (R.H.S)

- Another way to discretize PPE is through solving it by using fine grid variables.
- R.H.S of the PPE can be given as

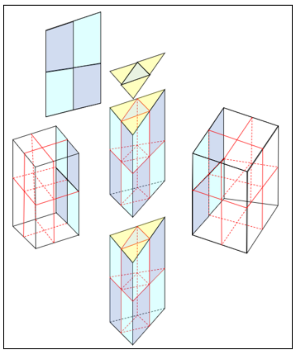
$$\sum_n \frac{c_n A_n}{d_n}(p_n - p_j) \approx \sum_{i \in \mathcal{S}^{\mathcal{R}}_{(j)}} \sum_m \frac{c_m A_m}{d_m}(p_m - p_i)$$

And

$$\frac{c_n A_n}{d_n} \approx \sum_{m \in \mathcal{S}^{\mathcal{R}}_{(n)}} \frac{c_m A_m}{2 d_m}$$

Thus

$$\sum_n \frac{c_n A_n}{d_n}(p_n - p_j) = \sum_n \sum_{m \in \mathcal{S}^{\mathcal{R}}_{(n)}} \frac{c_m A_m}{2 d_m}(p_n - p_j)$$



PPE coarse grid equation (L.H.S)

- L.H.S of PPE can equate as

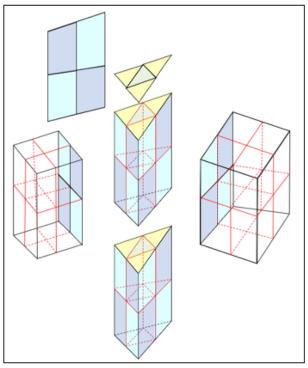
$$\sum_n A_n \hat{v}_n = \sum_{i \in \mathcal{S}^{\mathcal{R}}_{(j)}} \sum_m A_m \hat{v}_m$$

- Coarse grid PPE thus becomes

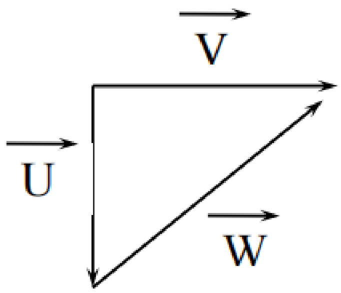
$$\sum_{i \in \mathcal{S}^{\mathcal{R}}_{(j)}} \sum_m A_m \hat{v}_m = \sum_n \sum_{m \in \mathcal{S}^{\mathcal{R}}_{(n)}} \frac{c_m A_m}{2 d_m}(p_n - p_j)$$

Or, using idea of continuum.

$$\sum_n \sum_{m \in \mathcal{S}^{\mathcal{R}}_{(n)}} A_m \hat{v}_m = \sum_n \sum_{m \in \mathcal{S}^{\mathcal{R}}_{(n)}} \frac{c_m A_m}{2 d_m}(p_n - p_j)$$



Théorème de composition des vitesses.



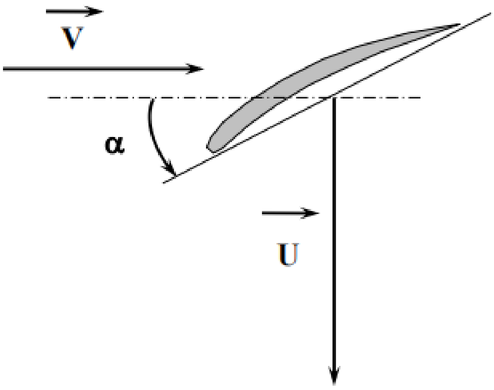
Vitesse absolue $\overrightarrow{\mathbf{V}}$ (observateur fixe).

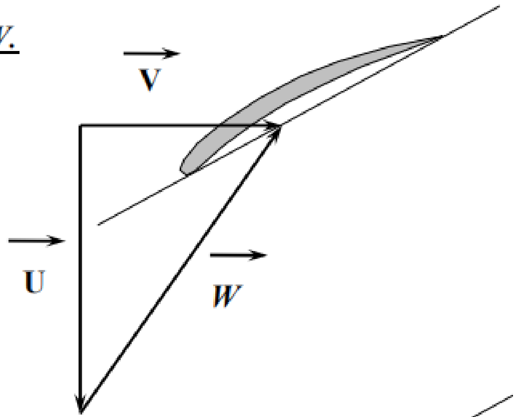
Vitesse relative $\overrightarrow{\mathbf{W}}$ (observateur solidaire de la pièce mobile).

Vitesse d'entraînement. $\overrightarrow{\mathbf{U}}$.

$$\overrightarrow{V} = \overrightarrow{W} + \overrightarrow{U}$$

☞ ***Construction de la vitesse relative W.***

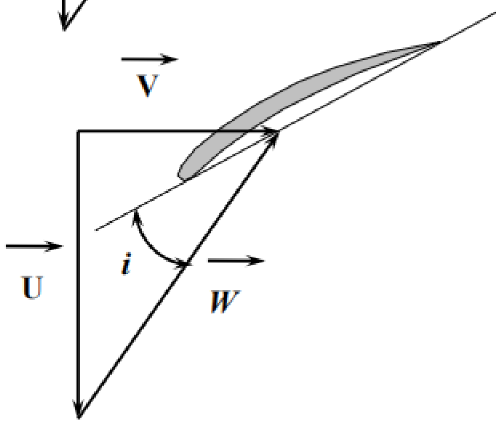




☞ ***Angle d'incidence d'une ailette.***

L'ailette fonctionne comme un profil d'aile.

La valeur de l'angle d'incidence i conditionne l'écoulement autour de l'ailette.



L'efficacité de l'ailette dépend directement de l'angle d'incidence i. Dans les compresseurs actuels $14° \leq i \leq 17°$.

Energie de pression.

On montre que l'énergie de pression gagnée dans un étage s'écrit :

$$P_{s3} - P_{s1} = \rho . U . \Delta W$$

Pour accroître le gain de pression, il suffit d'augmenter soit U soit ΔW.

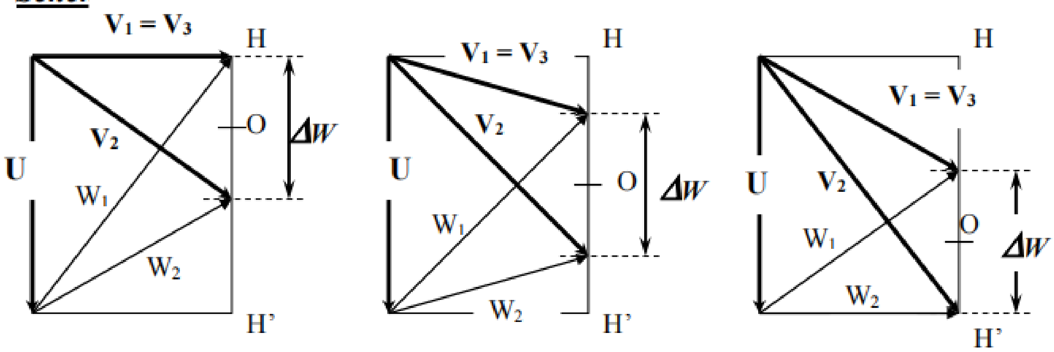
☞ U est limitée par des contraintes d'ordre mécaniques et par des l'apparition de phénomènes soniques en bout d'ailette.

☞ ΔW est limitée par l'apparition d'instabilités de fonctionnement du compresseur.

degré de réaction.

Le produit U.ΔW ne suffit pas à définir de façon précise un étage de compression.

Schéma :



Dans les trois cas présentés ci-dessus U et ΔW ne varie pas. Le travail de compression reste donc constant. Ce qui diffère, c'est la façon dont est obtenue la compression.