



저작자표시-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.
- 이 저작물을 영리 목적으로 이용할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

공학석사학위논문

**LU-SGS 수치기법을 적용한 OpenFOAM
기반의 압축성 코드 개발 및 검증**

**A Development and Verification of Density Based
Solver Using LU-SGS Algorithm in OpenFOAM**

2014 년 2 월

서울대학교 대학원

기계항공공학부

김 정 현

LU-SGS 수치기법을 적용한 OpenFOAM 기반의 압축성 코드 개발 및 검증

A Development and Verification of Density Based
Solver Using LU-SGS Algorithm in OpenFOAM

지도교수 김 규 홍

이 논문을 공학석사 학위논문으로 제출함

2013 년 10 월

서울대학교 대학원

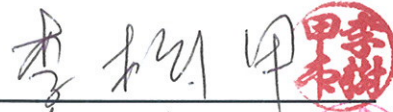
기계항공공학부

김 정 현


김정현의 공학석사 학위논문을 인준함

2013 년 12 월

위 원 장



부위원장



위 원



초 록

본 연구에서는 오픈소스(Open source)기반의 전산유체역학 프로그램인 오픈폼(OpenFOAM)에 LU-SGS(Lower Upper Symmetric Gauss Seidel) 수치기법을 적용한 압축성코드를 개발하고 이에 대한 검증을 수행하였다.

기본적으로 비압축성유동 해석을 위해 압력기반으로 개발된 오픈폼(OpenFOAM)의 지배방정식을 밀도기반의 지배방정식으로 수정하였고 기존의 오픈폼(OpenFOAM)에서 제공하고 있는 Explicit 시간적분법을 LU-SGS 수치기법을 적용한 Implicit 시간적분법으로 수정하였다. 또한 현재 오픈폼(OpenFOAM)에서 개발되지 않은 특성경계방정식인 리만(Riemann)경계조건을 추가하였고 기존에 오픈폼(OpenFOAM)에서 제공하고 있는 벽함수(Wall function) 및 일부 Library 에 대한 수정을 수행하였다.

개발된 코드를 기반으로 코드검증 시 널리 사용되고 있는 검증모델을 선정하여 검증을 수행하였고 이를 기존의 오픈폼(OpenFOAM) 해석자와 비교함에 따라 개발된 코드의 활용가능성을 확인하였다.

또한 이를 바탕으로 대한민국에서 개발하고 운용중인 비행기의 전산해석을 수행하였고 이를 풍동실험값과 비교함에 따라 향후 수행할 연구 방향을 제시하였다.

주요어 : 오픈폼(OpenFOAM), 오픈소스, LU-SGS, 전산유체역학

학 번 : 2012-20655

목 차

1. 서론	10
1.1 연구배경	10
1.2 오픈폼(OpenFOAM) 이란	13
1.3 연구목표	14
2. OpenFOAM Standard Solvers	16
2.1 비압축성(Incompressible) 유동	16
2.1.1 원형실린더 주위의 본 카르만 와류의 전산해석	16
2.1.2 해석조건 및 경계조건	17
2.1.3 코드검증 및 분석	18
2.1.3.1 이론적인 접근	18
2.1.3.2 Time Resolution Study	19
2.1.3.3 Grid Resolution Study	21
2.1.4 실험값과의 검증	23
2.2 압축성(Compressible) 유동	24
2.2.1 2차원 썰기(Wedge) 경사충격파 해석	24
2.2.2 해석조건 및 전산해석결과	24
3. Code Development and Validation	26
3.1 Density Based Solver	27
3.1.1 밀도 기반의 보존형 지배방정식	27
3.1.2 Oblique shock on a 2D wedge at Mach 2.5	28
3.2 Implicit Time Integration (LU-SGS)	30
3.2.1 Time Integration	30
3.2.2 Implicit LU-SGS	32
3.2.3 Transonic flow over a Bump in a channel	34
3.3 Characteristic Boundary Condition (Riemann Invariant)	35
3.3.1 Riemann Invariant	35

3.4 Turbulence model validation	40
3.4.1 RAE-2822 Transonic Airfoil	40
3.5 Automatic Wall Function	42
3.5.1 Wall treatment for a turbulence model	42
3.5.2 k-omega SST Model	43
3.5.3 Flat plate turbulent boundary layer	44
4. Applications	48
4.1 Shock Boundary Layer Interaction	48
4.1.1 해석조건 및 경계조건	48
4.1.2 충격파-경계층 상호작용의 유동현상	49
4.1.3 전산해석결과	50
4.2 Flat Plate	55
4.2.1 해석조건 및 경계조건	55
4.2.2 전산해석결과	57
4.3 Aircraft-1	60
4.3.1 해석조건 및 경계조건	60
4.3.2 전산해석결과	61
5. 결론	64
6. 참고 문헌	67

표 차례

표 1. CFD Jobs	11
표 2. Numerical schemes used for the simulation	17
표 3. Flow conditions for cylinder	18
표 4. Boundary conditions for cylinder	18
표 5. Calculation of empirical results	19
표 6. Calculated wake periods for each of time steps	21
표 7. Calculated wake periods for grid size variation	22
표 8. Calculated wake periods for domain size variation	22
표 9. Validation on natural(Unforced) experimental data	23
표 10. Boundary conditions for wedge	25
표 11. Flow conditions for wedge	25
표 12. Boundary conditions used for the wedge	29
표 13. Flow conditions used for the wedge	29
표 14. Comparison between theory and computation	30
표 15. Flow conditions for airfoil	38
표 16. Flow conditions used for RAE-2822 airfoil	40
표 17. Grids information	44
표 18. Flow conditions used of SWBLI	48
표 19. Boundary conditions used for SWBLI	51
표 20. The results from Oblique shock relation	52
표 21. Flow conditions used for flat plate	55
표 22. Flow conditions used for the Aircraft-1	60
표 23. Aerodynamic coefficients	62

그림 차례

그림 1. Why we need to use open source code	12
그림 2. Overview of OpenFOAM structure	13
그림 3. Flow schematic with flow geometry	16
그림 4. Grid system for cylinder(47,000)	17
그림 5. Velocity oscillations of monitoring point 0.02m (Two diameters of cylinder)	20
그림 6. The monitoring point schematic	20
그림 7. Grid independence study	22
그림 10. Computational results of Von Karman vortex shedding	23
그림 11. Grid system for wedge	24
그림 12. Pressure distribution along the wall	25
그림 13. Overview of code development process	26
그림 14. Grid system used for the wedge	28
그림 15. Pressure distribution (Original vs. Modified code)	29
그림 16. Explicit and Implicit time integration	32
그림 17. Mach contours of Bump in a channel	34
그림 18. Convergence history (Current means Implicit)	35
그림 19. Introduction to Boundary condition	36
그림 20. Grid system for airfoil	37
그림 21. OpenFOAM Free-stream Boundary condition	38
그림 22. Riemann Boundary condition in OpenFOAM	39
그림 23. Convergence history (Free-stream vs. Riemann)	39
그림 24. Grid system used for RAE-2822 airfoil	41
그림 25. Pressure coefficient on RAE-2822 airfoil	41
그림 26. Law of the wall	42
그림 27. OpenFOAM default wall treatment	45

그림 28. OpenFOAM modified wall treatment	46
그림 29. Wall shear stress (Original code wall treatment)	47
그림 30. Wall shear stress (Modified code wall treatment)	47
그림 31. Schematics of Shock B.L. Interaction	49
그림 32. Grid system used for SWBLI	50
그림 33. In-house vs. Modified OpenFOAM(Cp)	53
그림 34. In-house vs. Modified OpenFOAM(Cf)	53
그림 35. Pressure contour (Modified OpenFOAM)	54
그림 36. Pressure contour (In-house code)	54
그림 37. Schematic of the flat plate with flow geometry	55
그림 38. X direction	56
그림 39. Y direction	56
그림 40. Z direction	56
그림 41. Pressure contour (In-house code)	57
그림 42. Pressure contour (Modified OpenFOAM)	57
그림 43. Skin friction coefficient (Laminar flow)	58
그림 44. Skin friction coefficient (Turbulent flow)	59
그림 45. Grid system used for the Aircraft-1	60
그림 46. Pressure contour (OpenFOAM vs. FLUENT)	61
그림 47. Pressure contour (OpenFOAM)	62

1. 서론

1.1 연구배경

대한민국은 지난 20년간 KC-100, T-50 과 같은 다양한 항공기 개발과제가 수행되어져 왔으며 개발 이후에도 중형민항기, 중고도무인기와 같은 향후 개발과제들이 기대되고 있는 현실이다.[29] 일반적으로 비행기의 개념설계 단계에서는 구조, 제어, 추진 그리고 공력에 대한 해석이 요구된다. 특히 공력해석과 같은 유체역학과 관련된 해석은 크게 이론적인 접근, 실험적인 방법, 그리고 전산유체역학을 통한 해석으로 구분할 수 있다.[30] 이론적인 접근은 물리적인 자연현상을 지배방정식(Navier-Stokes Equation)으로 표현할 수는 있지만 그 방정식이 가지고 있는 비선형과 같은 특징 때문에 이론적인 해를 도출해내는 데는 한계가 존재한다. 실험적인 방법 또한 시험설비의 설치 및 유지, 시험모델 제작과 같은 비용의 부담이 큰 단점으로 작용하고 있다.[1] 이러한 단점을 보완하고자 비행기의 개발단계에서 전산유체역학의 활용도는 급증하고 있으며, 이를 통한 유동해석 연구가 다양한 분야에서 활용되고 있다.

그렇다면 전산유체역학(CFD, Computational Fluid Dynamics)이란 무엇인가? 사실 이미 약 150년 전부터 Navier-Stokes 방정식은 널리 알려져 오고 있었으나 그 방정식을 직접 풀어 유동을 해석할 수 있는 경우는 매우 단순한 물체 주위의 유동에 한하고 경계조건 역시 매우 간단한 경우에만 해석이 가능하였다. 게다가 방정식이 가지고 있는 비선형성의 특징 때문에 대부분의 실제 공학문제에서는 Navier-Stokes 방정식을 직접 해석적인 방법으로 풀어서 유동 현상을 이해한다는 것은 거의 불가능하였다. 이에 따라 앞선 연구자들은 여러 가지 가정을 도입하여 비선형 방정식을 선형화하고 대상 물체 및 경계조건 등을 단순화하여 유동해석

을 하거나, 실제 대부분은 실험에 의존해 오고 있었다. 그러나 1950년대부터 컴퓨터의 획기적인 발전에 힘입어 공학의 여러 분야에서 컴퓨터를 이용한 학문의 발전이 계속적으로 눈부시게 이루어져 왔으며 전산유체역학은 이러한 발전에 힘입어 다양한 분야에서 활용되고 있다.[32]

전산유체역학이란 유동을 지배하는 여러 종류의 편미분 방정식을 대수방정식으로 변화한 후 다양한 수치기법을 이용하여 이를 풀어 해를 구함으로써 유동을 해석하는 학문이다. 전산유체역학의 역사는 비록 약 50년 정도에 불과하나 그 응용범위는 기계, 항공, 조선, 토목, 그리고 해양 등과 같이 널리 이용되어 오고 있으며 앞으로도 급속한 발전이 기대되는 학문이라 할 수 있다.[32]

이러한 전산유체역학을 통한 유동해석연구는 In-house 코드를 개발하여 사용하거나 FLUENT와 같이 상용 전산유체역학 해석 프로그램을 사용하는 것이 일반적이다.[31] 상용프로그램을 사용하는 경우에는 해석 코드를 개발해야 하는 시간과 노력을 절약할 수 있으며 사용방법 학습이 용이하여 사람들에게 쉽게 다가갈 수 있다. 그러나 이 경우에는 초기 구매 비용과 유지보수 비용이 상당히며 별도의 License 비용을 지불해야 하는 단점이 존재할 뿐만 아니라 코드의 수정이 불가능하여 특정 문제를 해결하는데 사용자가 제한을 받을 수 있다.[16] 가령, 한국항공우주산업(KAI)의 경우에도 최근 전산유체역학의 활용도가 급증하고 있으나 상용프로그램에 대한 의존도가 높은 산업체의 특성상, 비용 증가문제에 대한 고민도 커져가는 실정이다.[29]

CFD Jobs (KAI)	
2008	1967
2010	7185
2012	8606

표 1. CFD Jobs

그에 반해 In-house 코드 개발을 통한 전산유체역학 연구는 자신의 연구 분야에 적용하기에 적절한 해석기법을 사용하기 때문에, 높은 정확도를 가지며 개발자에 의해 수정이 용이하다는 장점을 지니고 있다. 그러나 해석 코드를 개발하기 위해서 필요한 시간과 노력이 상당하며 개발자 이외의 연구자가 사용하기에는 상당한 노하우가 개발자에 의해 전수되어야 하는 단점을 지니고 있다.[16]

In-house 코드와 상용프로그램 각각의 단점을 보완하고자 1990년대 초부터 비용이 들지 않고 공개 프로그램을 지향함에 따라 쉽게 코드에 접근할 수 있는 오픈소스 기반의 전산유체역학 프로그램에 관한 연구가 진행되어 왔으며 OpenFOAM(Open Field Operation And Manipulation)이 대표적인 예라고 할 수 있다.[1]

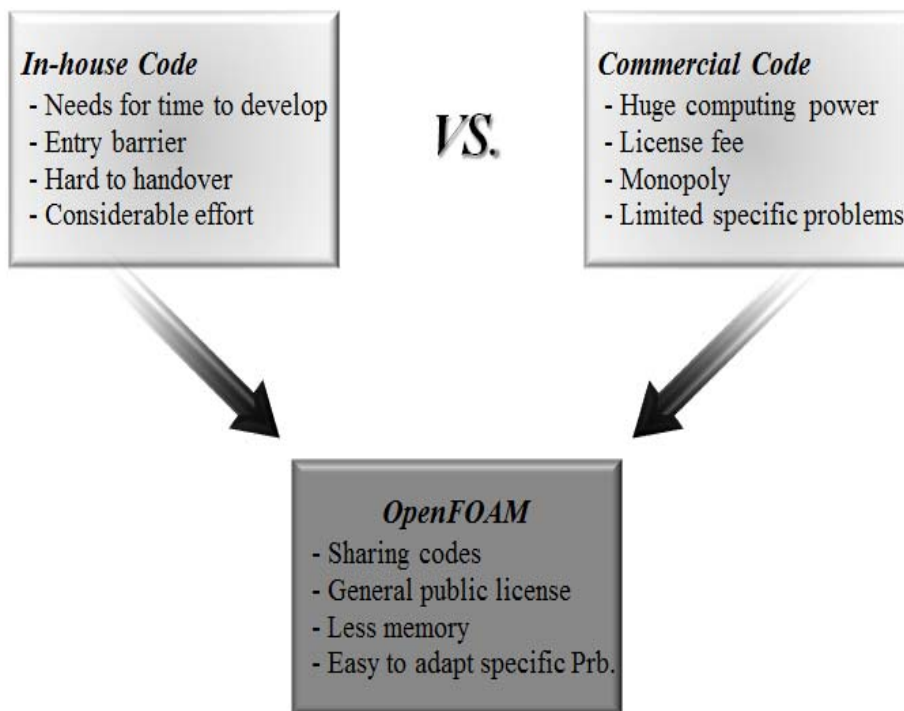


그림 1. Why we need to use open source code

1.2 오픈폼(OpenFOAM) 이란

오픈폼(OpenFOAM)은 1980년대 영국의 Imperial College에서 시작되었으며 LINUX OS하에서 C++ 언어 기반으로 개발되었다. 처음에는 FOAM(Field Operation And Manipulation)이란 이름으로 유료로 배포되었으나 2004년 OpenFOAM이란 이름으로 오픈소스 프로그램으로 변경되어 무상으로 배포되었고 현재에는 세계 각국에 많은 사용자층을 가지고 있다.[15]

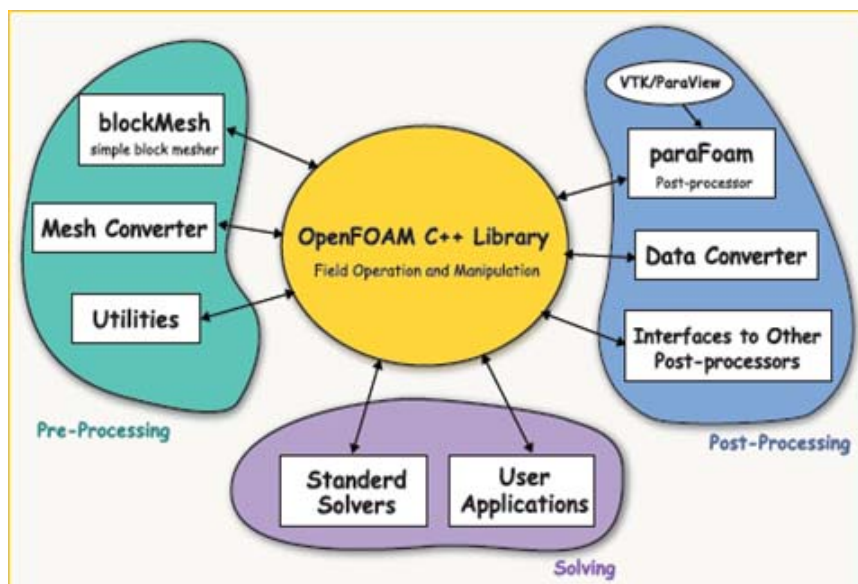


그림 2. Overview of OpenFOAM structure

오픈폼(OpenFOAM)은 일반 상용프로그램 혹은 In-house 코드와는 달리 편미분방정식의 수치적인 해를 구할 수 있는 객체 라이브러리를 제공하고 있고, 비용이 들지 않는 GPL(General Public License)를 사용하고 있어 새로운 해석모델의 이식성이 가장 뛰어난 코드로 알려져 있다.[6] 이러한 장점은 연구원 혹은 학교 연구실에서 개발된 In-house 코드를 상용코드 수준으로 향상

시킬 수 있는 잠재력을 가지고 있는 것으로 평가되어 외국에서는 이를 이용한 해석코드 개발이 활발하게 진행되고 있다. 국내에서는 현재 서울대학교, 한국과학기술원(KAIST), 그리고 부산대 등과 같은 학교 연구실을 포함해 넥스트폼(NextFOAM)과 같은 오픈폼(OpenFOAM) 전문 컨설팅 회사와 한국항공우주산업(KAI)와 같은 산업체가 주를 이루고 있다.

오픈폼(OpenFOAM)은 유한체적기법(Finite Volume Method, FVM)으로 유동현상을 해석한다. 오픈폼(OpenFOAM)의 핵심적인 장점 중 하나는 지배방정식을 사용자가 자유롭게 구성할 수 있으며 그 형식이 편미분방정식과 유사하여 사용자가 내부적인 문법을 알지 못해도 직관적으로 사용할 수 있다는 점이다. 한 예로 다음의 편미분방정식을 오픈폼(OpenFOAM) 상에서 표현하면 다음과 같다.[7]

$$\frac{\partial \rho U}{\partial t} + \nabla \cdot \phi - \nabla \cdot \mu U = -\nabla p$$



```
solve(fvm::ddt(rho,U)+fvm::div(phi,U)-fvm::laplacian(mu,U)
== -fv::grad(p));
```

1.3 연구목표

오픈폼(OpenFOAM)은 일반적으로 다음과 같은 장점 때문에 국외뿐만 아니라 국내에서도 많은 개발자 및 사용자들에게 널리 사용되고 있다: 1) 공개 프로그램을 지향함에 쉽게 코드에 접근할 수 있다. 2) 비교적 메모리를 적게 요구한다. 3) 비용이 들지 않는 GPL(General Public License)를 사용한다. 4) 특정한 연구 분야에 적용하기에 용이하다.[30]

그러나 이러한 오픈폼(OpenFOAM)은 일반 상용프로그램과 같이 사용자 환경을 제공하지 않고 있어 진입장벽이 높고 해석정밀도에 대해 공식적으로 검증되어 있지 않아 국내에서는 아직 많은 사용자층을 확보하지 못하고 있는 실정이다.

특히 오픈폼(OpenFOAM)은 비압축성유동에 대한 해석을 기반으로 개발되었기 때문에, 비압축성 유동에 대해서는 높은 정확도와 안정성을 가지고 있지만 압축성 유동에 대해서는 기본적인 해석자들이 공급되어 있지 않고 공개된 몇몇 해석자들 또한 압력기반의 지배방정식을 사용하기 때문에 압축성유동 해석 시 상당한 오차를 발생시킨다. 즉 현재 오픈폼(OpenFOAM)은 압축성 유동에 대한 해석자의 개발과 공식적인 검증이 이루어지지 않았다고 볼 수 있다.

이에 따라 본 논문에서는 기존의 오픈소스(Open source)기반의 전산유체역학 프로그램인 오픈폼(OpenFOAM)에 LU-SGS(Lower Upper Symmetric Gauss Seidel) 수치기법을 적용한 압축성코드를 개발하고 이에 대한 검증을 수행하였다. 기본적으로 비압축성 유동 해석을 위해 압력기반으로 개발된 오픈폼(OpenFOAM)의 지배방정식을 밀도기반의 지배방정식으로 수정하였고 기존의 오픈폼(OpenFOAM)에서 제공하고 있는 Explicit 시간적분법을 LU-SGS 수치기법을 적용한 Implicit 시간적분법으로 수정하였다. 또한 현재 오픈폼(OpenFOAM)에서 개발되지 않은 특성경계방정식인 리만(Riemann) 경계조건을 추가하였고 기존에 오픈폼(OpenFOAM)에서 제공하고 있는 벽함수(Wall function) 및 일부 라이브러리에 대한 수정을 수행하였다.

개발된 코드를 바탕으로 코드검증 시 널리 사용되고 있는 검증 모델을 선정하여 검증을 수행하였고 이를 기존의 오픈폼(OpenFOAM) 해석자와 비교함에 따라 개발된 코드의 활용가능성을 확인하였다.

본 논문의 본문 중 2장에서는 최근 국내외적으로 많은 주목을 받고 있는 오픈폼(OpenFOAM)의 기본해석자들을 이용한 비압축성유동에 대한 해석에 대해 다룰 것이고, 3장에서는 오픈폼(OpenFOAM)에 압축성효과를 고려한 코드개발 과정과 검증에 대해 다룰 것이다. 마지막으로 4장에서는 개발된 코드를 이용한 다양한 문제에 대한 응용부분을 다룰 것이다.

2. OpenFOAM Standard Solvers

2.1 비압축성(Incompressible) 유동

2.1.1 원형실린더 주위의 본 카르만 와류의 전산해석[1]

본 논문에서는 비압축성 유동 문제의 대표적인 검증문제로써 원형 실린더 주위에 발생하는 본 카르만 와류를 선정하여 오픈폼(OpenFOAM)에서 기본적으로 제공하는 icoFoam, pimpleFoam 등과 같은 비압축성 기반의 해석자에 대한 검증을 수행하였다.

2차원 Navier-Stokes 방정식을 수치적으로 접근하였고 실린더 후방부 유동의 불규칙적인 변화와 본 카르만 와류 현상을 확인할 수 있었다.

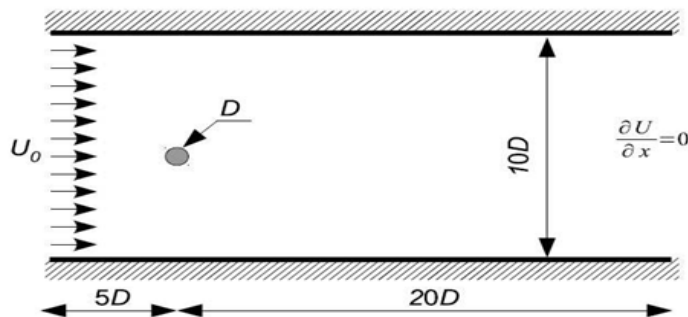


그림 3. Flow schematic with flow geometry

2.1.2 해석조건 및 경계조건

원형실린더 주변의 유동에 대한 전산해석을 위해 아래와 같은 격자와 경계조건 및 수치기법을 사용하였다. 해석격자는 약 47,000개로 상용격자 생성 프로그램인 Gambit 2.4를 사용하였고 오픈폼(OpenFOAM)에서 사용할 수 있도록 격자를 전환하였다.

Grid Resolution Study를 통해 아래 그림과 같은 격자를 선정하였고 이후 모든 해석영역에서 동일한 격자를 기반으로 해석을 수행하였다.

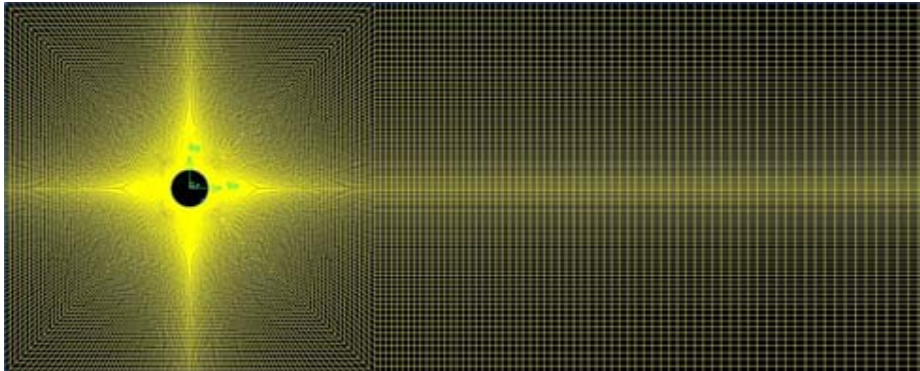


그림 4. Grid system for cylinder(47,000)

경계조건 및 해석조건은 다음과 같다.

Numerical schemes	
Time discretization	Implicit backward 2 nd order
Convective discretization	Total Variation Diminishing
Diffusion discretization	Central differencing scheme
Convergence criterion	1e-06

표 2. Numerical schemes used for the simulation

Flow Conditions	
μ_{∞}	$1.824E-05 \text{ Ns/m}^2$
ρ_{∞}	1.19 kg/m^3
<i>Reynolds Number</i>	100
U_{∞}	0.15 m/s
<i>D(Characteristic length)</i>	0.01 m
<i>Fluid</i>	<i>Air</i>

표 3. Flow conditions for cylinder

	Pressure	Velocity
Inlet	Zero gradient	Fixed value
Outlet	Fixed value	Zero gradient
Side	Zero gradient	Fixed value
Cylinder	Zero gradient	Fixed value

표 4. Boundary conditions for cylinder

전산해석 영역의 수치정확도는 2차정확도로 설정하였고 난류모델은 층류를 선정하였다. 비정상해석을 위해 Time Resolution Study를 통해 Time step은 0.00025s로 선정하였다. 이후 모든 해석영역에서 동일한 Time step를 기반으로 해석을 수행하였으며 전체해석 시간은 10초이고 Courant Number는 1보다 작게 설정하였다.

2.1.3 코드검증 및 분석

2.1.3.1 이론적인 접근

전산해석의 수행에 앞서 Williamson의 경험식(1989)을 이용하여 해석영역의 Reynolds number에서의 Strouhal number와 원형실

린더 후방부에서 발생하는 Von Karman vortex shedding의 주기를 예측하였다. 사용된 Willamson[2]의 경험식은 다음과 같다.

$$St = \frac{D}{U_\infty} = \frac{-3.3265}{Re_D} + 0.1816 + 0.00016Re_D$$

Willamson의 경험식에 의해 구해진 결과는 아래의 표와 같다.

Calculation of empirical results	
<i>StrouhalNumber</i>	0.1643
<i>ReynoldsNumber</i>	100
<i>Period</i>	0.4056s
<i>Frequency</i>	2.4650s

표 5. Calculation of empirical results

2.1.3.2 Time Resolution Study

비정상문제에 있어 물리적으로 정확한 Time step을 선정하기 위해 Time Resolution Study를 수행하였다.[22] 먼저 0.001s에 대하여 원형실린더 후방 유동이 안정화된 주기를 나타낼 까지 전산 해석을 수행하였다. 이때 원형실린더의 특성길이의 2배가 되는 x축에 Monitoring point를 선정하였고[3] 해당 지점에서 진동의 안정화가 나타날 때까지 해석을 수행한 뒤 다음과 같은 관계식을 이용하여 주기를 계산하였다.

$$t_{zero} = t_i - \frac{(t_{i+1} - t_i) V_i}{(V_{i+1} - V_i)}, \quad \tau = t_{zero_{i+1}} - t_{zero_i}$$

$$f = \frac{1}{\tau}, \quad St = \frac{D}{U_\infty \tau} = \frac{Df}{U_\infty}$$

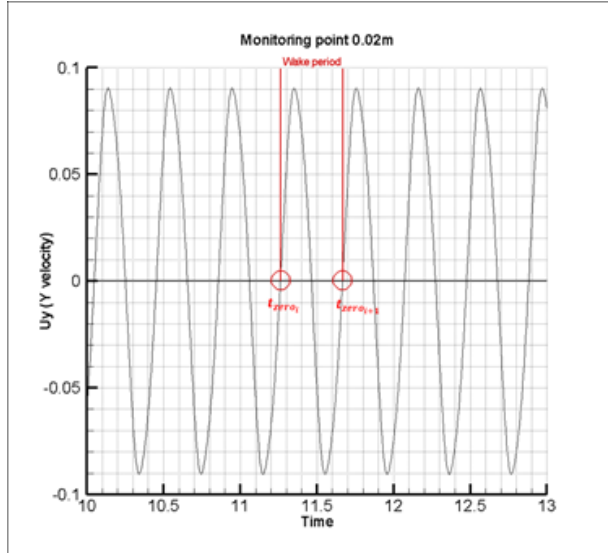


그림 5. Velocity oscillations of monitoring point 0.02m (Two diameters of cylinder)

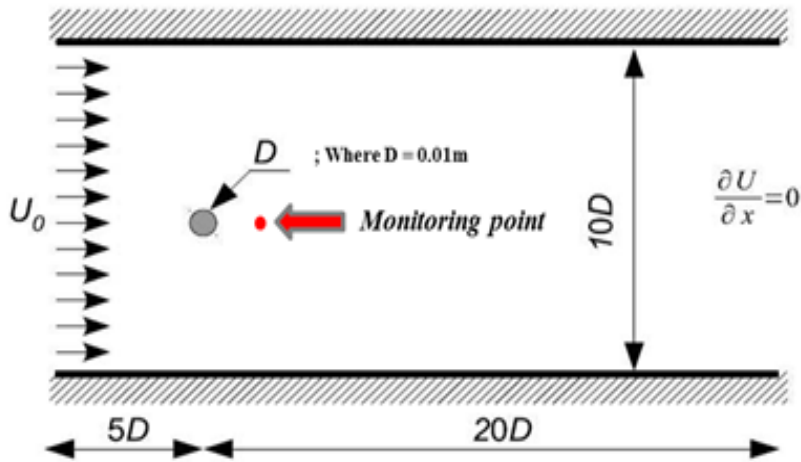


그림 6. The monitoring point schematic

Time step을 0.002s부터 0.00025s까지 변경함에 따라 앞서 언급한 방법을 동일하게 적용하여 주기를 계산하였다. 각각의 주기에 해당하는 Strouhal number는 아래의 표와 같다.

Time step	Wake period	Frequency	St #
Experiment	0.4056	2.4650	0.1643
0.002	Divergence		
0.001	0.4293	2.3293	0.1553
0.0005	0.4142	2.4142	0.1609
0.00025	0.4045	2.4716	0.1647

표 6. Calculated wake periods for each of time steps

표6에서 볼 수 있듯이 수치적으로 접근하여 계산한 Strouhal number는 Time step이 0.00025s 일 때 이론값과 가장 근접한 것을 확인할 수 있었다. 이후 모든 해석영역에서 동일한 Time step(t=0.00025s)를 기반으로 해석을 수행하였다.

2.1.3.3 Grid Resolution Study

Time Resolution Study와 마찬가지로 수치적으로 가장 효율적인 격자를 선정하기 위해 3개의 격자 시스템에 대해 Grid Resolution Study를 수행하였다. 3개의 시스템 모두 앞선 연구에서 선정한 0.00025s를 Time step으로 적용하였고 원형실린더 후방 유동이 안정화된 주기를 나타낼 때까지 전산해석을 수행하였다. 아래의 그림과 같이 격자 시스템은 각각 14850, 47000, 132600개 이며 계산된 주기는 아래의 표와 같다.

Cells	Wake period	Frequency	St #
Experiment	0.4056	2.4650	0.1643
14850	0.3980	2.5123	0.1674
47000	0.4045	2.4716	0.1647
132600	0.4048	2.4303	0.1646

표 7. Calculated wake periods for grid size variation

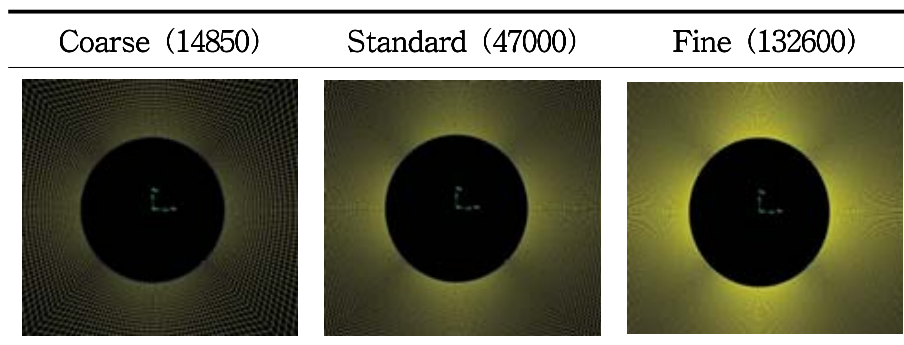


그림 7. Grid independence study

전산해석영역의 타당성 및 효율성의 검증을 위해 Domain Independence Study 또한 수행하였다. 아래의 표에서 볼 수 있듯이 1.5배가 큰 전산해석영역의 Strouhal number와 기존의 해석영역에서 전산해석으로 구해진 값에는 큰 차이가 없음을 확인할 수 있었고 이를 통해 기존의 해석영역으로 최종 격자시스템을 구성하였다.

Domain size	Wake period	Frequency	St #
Experiment	0.4056	2.4650	0.1643
Standard	0.4045	2.4716	0.1647
x1.5 times	0.4047	2.4709	0.1647

표 8. Calculated wake periods for domain size variation

2.1.4 실험값과의 검증

앞선 연구결과를 바탕으로 선정된 Time step, Grid system, 그리고 Domain 크기를 바탕으로 실험값과의 비교분석을 수행하였다. 아래의 표에서 볼 수 있듯이 실험값과 수치적인 해가 거의 일치하는 것을 확인할 수 있었고 이는 곧 오픈폼(OpenFOAM)이 기본적으로 제공하고 있는 비압축성 유동 해석자의 검증이 성공적으로 이루어졌다고 말할 수 있다.

(Re = 100)	Strouhal number
Experiment	0.1643
Computation (OpenFOAM)	0.1647

표 9. Validation on natural(Unforced) experimental data

아래 그림은 전산해석을 통한 원형실린더 후방부 유동의 Von Karman vortex shedding 현상을 보여주고 있다.

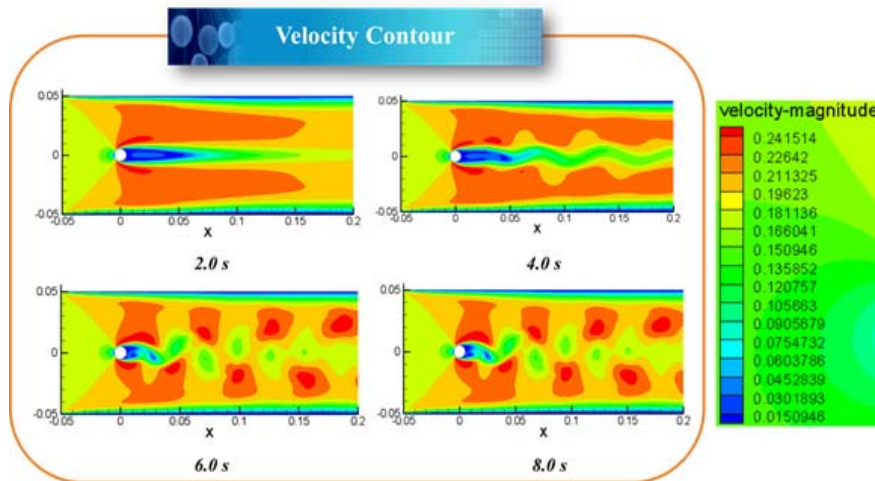


그림 10. Computational results of Von Karman vortex shedding

2.2 압축성(Compressible) 유동

2.2.1 2차원 썸기(Wedge) 경사충격파 해석

일반적으로 오픈폼(OpenFOAM)은 비압축성유동에 대한 해석을 기반으로 개발되었기 때문에 비압축성유동에 대해서는 높은 정확도와 안정성을 가지고 있는 반면 압축성유동에 대해서는 해석자의 개발과 공식적인 검증이 이루어지지 않은 단계이다. 본 논문에서는 현재 오픈폼(OpenFOAM)에서 제공하고 있는 압축성 유동에 대한 해석자에 대한 기본 검증을 2차원 썸기 경사충격파 문제를 선정하여 그 활용 가능성을 확인하였다.

2.2.2 해석조건 및 전산해석결과

2차원 썸기의 경사충격파에 대한 전산해석을 위해 다음과 같은 격자와 해석조건을 적용하였다.

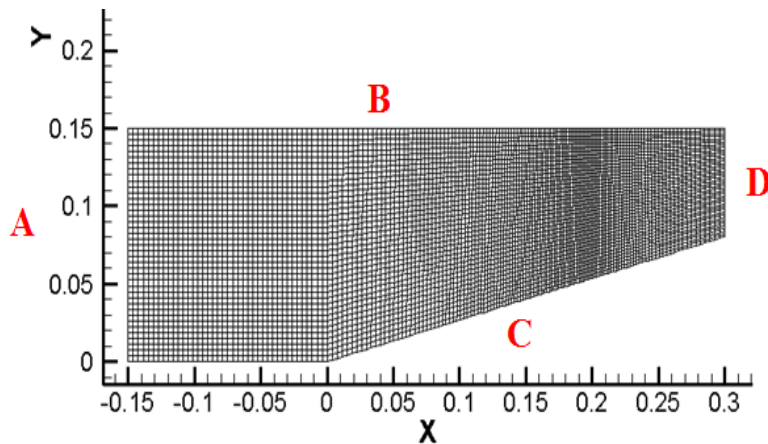


그림 11. Grid system for wedge

Boundary Conditions	
A	Fixed value (P,T,U)
B	1 st order extrapolation
C	Slip wall
D	1 st order extrapolation

표 10. Boundary conditions for wedge

Flow Conditions	
μ_{∞}	$1.78E-05 \text{Ns/m}^2$
T_{∞}	288.88K
P_{∞}	101325.58Pa
M_{∞}	2.5
Half Angle Of Wedge	15°
Fluid	Air

표 11. Flow conditions for wedge

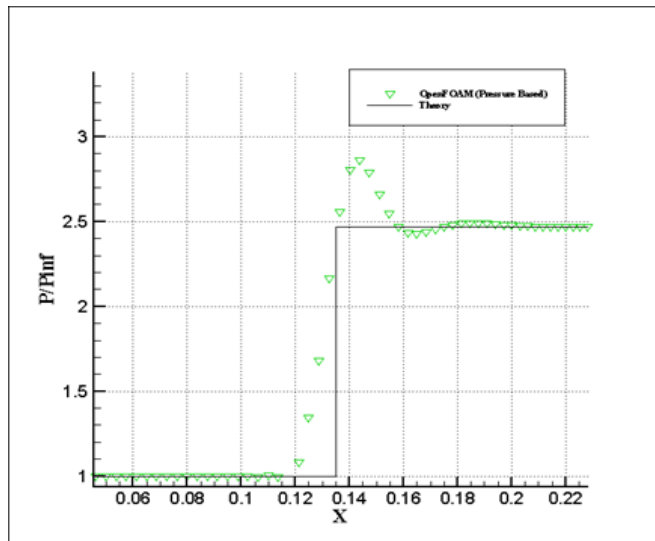


그림 12. Pressure distribution along the wall

초음속 영역의 해석 및 이론값과의 비교를 위해[12] 비점성, 압축성 유동의 조건으로 해석을 수행하였으며 충격파 전후의 압력 값은 그림12와 같다. 그림에서 볼 수 있듯이, 충격파 이후에 물리적인 값을 예측하는 데 있어서 수치적인 오류가 상당하고 충격파 전후 진동이 심한 것을 확인할 수 있었다. 이는 곧 압력기반으로 개발된 오픈폼(OpenFOAM)의 압축성 해석자에 대한 한계를 보여주고 있다.

다시 말해 현재 오픈폼(OpenFOAM)에서 개발된 해석자로는 압축성유동을 다룰 수 없으며 이는 곧 오픈소스(Open source)기반의 오픈폼(OpenFOAM)에 압축성코드의 개발의 필요성을 의미하며, 이것이 바로 본 논문의 가장 주된 목적이라 할 수 있다.

3. Code Development and Validation

앞서 언급한 오픈폼(OpenFOAM)의 압축성해석에 대한 한계를 극복하고 오픈소스기반의 코드를 이용한 유동해석의 효율성과 정확성을 위해 다음 그림과 같은 개발과정을 수행하였다.

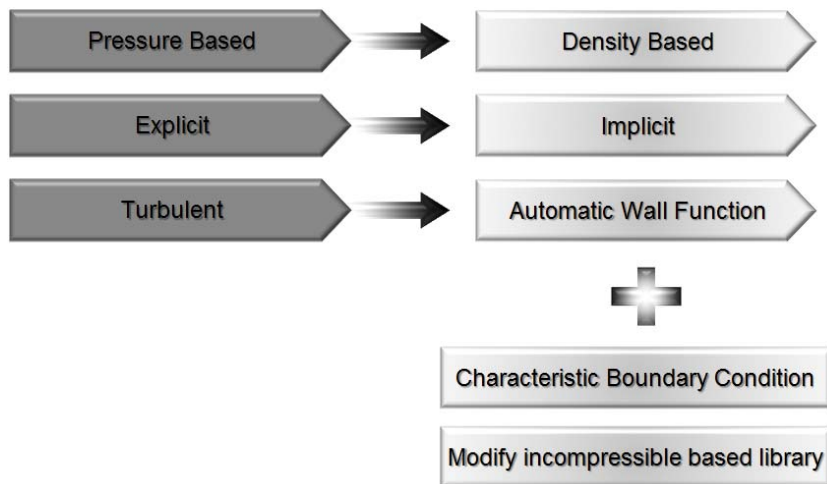


그림 13. Overview of code development process

3.1 Density Based Solver[4]

3.1.1 밀도 기반의 보존형 지배방정식

전산유체역학에서 지배 방정식은 일반적으로 연립된 편미분 방정식으로 유동해석 시에는 이를 이산화한 후 수치적인 해를 구하게 된다. 방정식의 이산화를 수행하면 공간 이산화와 시간 이산화로 나누게 되며 다음과 같은 형태를 갖는다.[23]

$$[Time\ integration]\delta Q = [Flux\ function]$$

여기서 δQ 는 각 시간단계에서 보정되는 값이고 Flux function은 수치해의 정확도에 영향을 미치는 공간 이산화부분이며, Time integration은 계산의 효율성에 영향을 미치는 시간 이산화부분이다. 본 논문에서는 공간 이산화에는 유한체적법을 사용하였고 수치기법으로는 Entropy가 fix된 Roe Scheme[10]을 선정하였다. 또한 정상 유동문제의 해석자로서 해의 수렴성을 증가시키기 위해 국소 시간 간격(Local time stepping)을 적용하였다. 본 논문에서 사용된 보존형 지배방정식은 다음과 같다.[9]

$$\int_V \frac{\partial W}{\partial t} dv + \int_V (F_c - F_v) ds = 0$$

수치적인 해를 얻기 위해 지배방정식을 공간에 대해 다음과 같이 이산화 하였다.

$$V_i \frac{\partial W_i}{\partial t} + \sum_{j \in N(i)} (F_{ij} - F_{v,ij}) S_{ij} = 0$$

여기서 보존량 벡터 W 와 플럭스(Flux) 벡터 F 는 다음과 같다.

$$W = \begin{bmatrix} \rho \\ \rho U \\ \rho E \end{bmatrix} \quad F_c = \begin{bmatrix} \rho U \\ \rho U_i U_j + \Pi \\ \rho H U \end{bmatrix}$$

$$F_v = \begin{bmatrix} 0 \\ \tau_{ij} \\ \nabla \cdot (\tau_{ij} U + \rho \alpha_{eff} \nabla h + (\mu + \frac{\mu_t}{\sigma_k}) \nabla k) \end{bmatrix}$$

3.1.2 Oblique shock on a 2D wedge at Mach 2.5

개발된 코드의 검증을 위해 2차원 쐐기(Wedge)의 경사충격파에 대한 전산해석을 수행하였고 기존에 오픈폼(OpenFOAM)에서 개발된 압축성 해석자와의 비교분석을 수행하였다. 기본적인 해석조건 및 경계조건은 아래와 같으며 초음속 영역의 해석 및 이론값과의 비교를 위해 비점성, 압축성 유동의 조건으로 해석을 수행하였으며 충격파 전후의 압력 값은 표14와 같다.

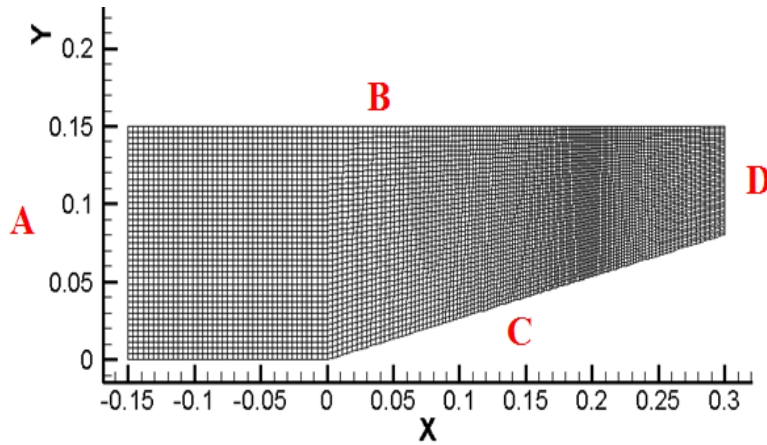


그림 14. Grid system used for the wedge

Boundary Conditions	
A	Fixed value (P,T,U)
B	1 st order extrapolation
C	Slip wall
D	1 st order extrapolation

표 12. Boundary conditions used for the wedge

Flow Conditions	
μ_{∞}	$1.78E-05 \text{ Ns/m}^2$
T_{∞}	288.88 K
P_{∞}	101325.58 Pa
M_{∞}	2.5
Half Angle Of Wedge	15°
Fluid	Air

표 13. Flow conditions used for the wedge

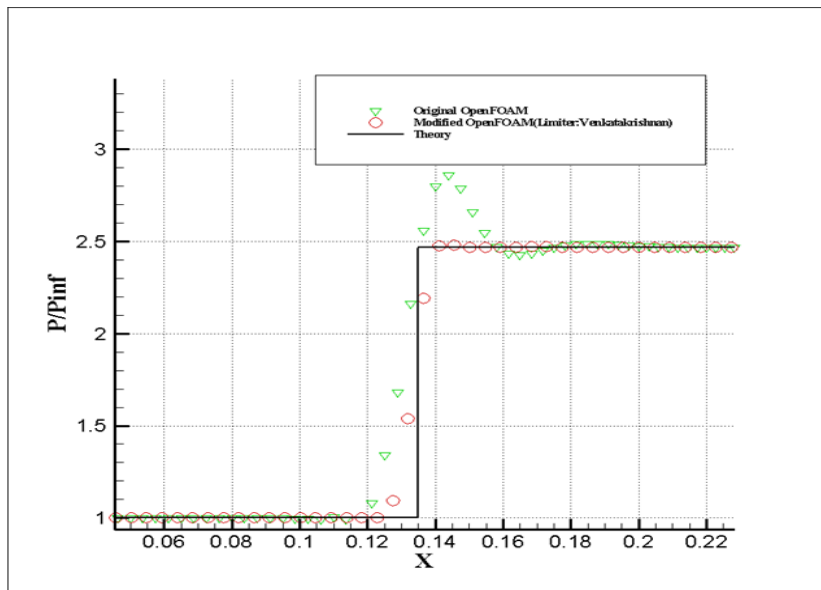


그림 15. Pressure distribution (Original vs. Modified code)

Inviscid and Compressible flow over a wedge			
	Theory[27]	OpenFOAM	Error (%)
Density ratio	1.8665	1.8641	0.12
Temp. ratio	1.3219	1.3236	0.13
Pressure ratio	2.4675	2.4655	0.08

표 14. Comparison between theory and computation
(Modified OpenFOAM code)

충격파 이후에 물리적인 값을 예측하는 데 있어서 오픈폼 (OpenFOAM)이 제공하는 압축성 해석자는 수치적인 오류가 상당히 많고 충격파 전후 진동이 심한 반면 개발된 코드의 경우에는 충격파 이후에 진동이 미세하며 이론값을 예측하는 데 있어서도 정확하게 예측하는 것을 확인할 수 있었다.

3.2 Implicit Time Integration (LU-SGS)

3.2.1 Time Integration

앞서 언급한 대로 편미분형태로 되어있는 지배방정식의 수치적인 해를 구하기 위해서는 공간이산화 뿐만 아니라 시간이산화 과정도 이루어져야 한다. 일반적으로 시간이산화를 위해 사용되는 시간 적분법은 외재적(Explicit)인 방법과 내재적(Implicit)인 방법이 있다.[18]

예를 들어 다음과 같은 지배방정식이 있을 때 시간 항에 관련된 Q항에 대해 고려해보자.

$$\frac{\partial Q}{\partial t} + \left(\frac{\partial E}{\partial x} + \frac{\partial F}{\partial y} + \frac{\partial G}{\partial z} \right) = \frac{\partial E_v}{\partial x} + \frac{\partial F_v}{\partial y} + \frac{\partial G_v}{\partial z}$$

이때 Q항에 대한 시간이산화를 위해서 외재적인 방법을 이용하면 다음과 같은 과정으로 n+1번째의 값을 구할 수 있다.

$$\frac{q^{n+1} - q^n}{\Delta t} + \left(\frac{\partial E}{\partial x}\right)^n = 0$$

$$\Delta q^{n+1} = \left(\frac{\partial E}{\partial x}\right)^n \Delta t$$

또한 아래의 식과 같이 내재적인 방법을 이용해도 동일한 원하는 값을 구할 수 있다.

$$\frac{q^{n+1} - q^n}{\Delta t} + \left(\frac{\partial E}{\partial x}\right)^{n+1} = 0$$

$$\Delta q^{n+1} = \left(\frac{\partial E}{\partial x}\right)^{n+1} \Delta t$$

외재적인 방법은 프로그램 코딩이 간편하고 한번 계산하는데 시간이 빠른 반면 Courant number에 제약을 받기 때문에 불안정한 특성을 가지고 있다. 그에 반해 내재적인 방법은 Matrix를 푸는 형태로 되어 있기 때문에 코딩이 복잡하고 한번 계산하는데 시간이 오래 걸리는 반면 Courant number에 제약을 받지 않기 때문에 수렴성이 좋으며 반복계산 시에는 외재적인 방법보다 같은 조건에서 우수한 효율성을 가지고 있다. 특히 내재적인 방법은 비정상(Unsteady)문제를 다루는 데는 한계가 존재하지만, 정상(Steady)문제를 접근하는 데 있어서는 계산의 효율성 및 수렴성 측면에서 유리한 것으로 알려져 있다. 시간적분법의 원리를 간단하게 그림으로 표현하면 아래와 같다.[24]

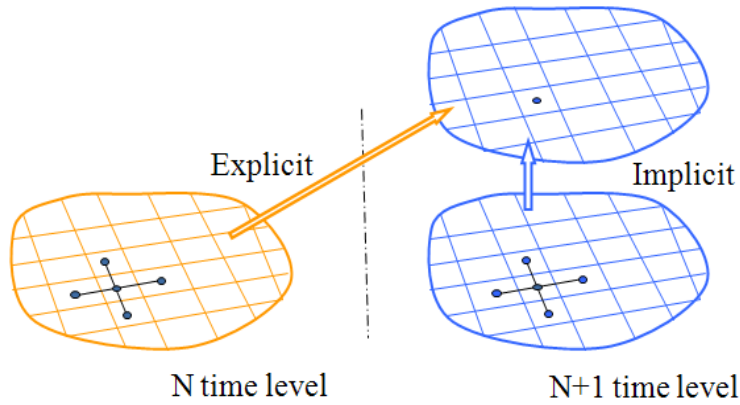


그림 16. Explicit and Implicit time integration

3.2.2 Implicit LU-SGS

본 논문에서는 시간 이산화를 위해 LU-SGS 수치기법을 적용한 내재적인 방법을 시간적분법으로 이용하였다. LU-SGS 수치기법은 벡터의 역 연산을 필요로 하지 않기 때문에 요구되는 메모리 사용량과 계산 시간이 작아져 외부유동을 해석하는 데 있어서 매우 효율적으로 수렴된 해를 얻을 수 있기 때문에 항공 분야에서 많이 사용되고 있다.[5] 본 논문에서 사용한 LU-SGS 수치기법은 Thin Shear Layer(TSL) 가정을 적용한 방법으로, 다음과 같은 과정으로 유도하여 적용하였다.

$$V_i \frac{\partial W_i}{\partial t} + \sum_{j \in N(i)} (F_{ij} - F_{v,ij}) S_{ij} = 0$$

$$\frac{V_i}{\Delta t_i} (W_i^{n+1} - W_i^n) + \sum_{j \in N(i)} (F_{c,ij}^{n+1} - F_{v,ij}^{n+1}) S_{ij} = 0$$

이때 Flux 벡터 F에 대한 항에 대해서는 다음과 같이 가정하였고 각각의 문자가 의미하는 바는 아래와 같다.

$$F_{ij}^{n+1} \approx F_{ij}^n + \left(\frac{\partial F}{\partial W}\right) \Delta W_{ij}^n$$

$$\frac{V_i}{\Delta t_i} \Delta W_i^n + \sum_{j \in N(i)} (A_{c,ij} - A_{v,ij}) \Delta W_{ij}^n S_{ij} = -Res_i^n$$

$$\Delta W_i^n = W_i^{n+1} - W_i^n$$

$$A_c = \frac{\partial F_c}{\partial W}: \text{Convective Flux Jacobian}$$

$$A_v = \frac{\partial F_v}{\partial W}: \text{Viscous Flux Jacobian}$$

여기서 앞서 언급한 TSL 가정을 통해 지배방정식을 다시 정리하면 다음과 같다.

$$\frac{V_i}{\Delta t_i} \Delta W_i^n + \sum_{j \in N(i)} (A_{c,i}^+ + A_{v,i}^*) \Delta W_i^n S_{ij} + \sum_{j \in N(i)} (A_{c,j}^- - A_{v,j}^*) \Delta W_j^n S_{ij} = -Res_i^n$$

LU Composition을 위해 이웃하는 면과 Owner 면에 대해 L과 U성분으로 분리하여 방정식을 정리하면 다음과 같다.

$$\frac{V_i}{\Delta t_i} \Delta W_i^n + \sum_{j \in N(i)} (A_{c,i}^+ + A_{v,i}^*) \Delta W_i^n S_{ij} + \sum_{j \in L(i)} (A_{c,j}^- - A_{v,j}^*) \Delta W_j^n S_{ij}$$

$$+ \sum_{j \in U(i)} (A_{c,j}^- - A_{v,j}^*) \Delta W_j^n S_{ij} = -Res_i^n$$

이를 통해 각각의 Matrix를 정의하고 정리한 식은 아래와 같이 쓸 수 있다.

$$(D+L+U)\Delta W^n = -R^n$$

$$(D+L)D^{-1}(D+U)\Delta W^n = -R^n + LD^{-1}U\Delta W^n$$

$$(D+L)\Delta W^* = -R^n ; \text{Forward}$$

$$(D+U)\Delta W^n = D\Delta W^* ; \text{Backward}$$

3.2.3 Transonic flow over a Bump in a channel

LU-SGS 수치기법을 적용한 밀도기반의 개발된 코드의 검증을 위해 오픈폼(OpenFOAM)에서 제공하고 있는 대표적인 외재적인 방법인 Runge-Kutta 수치기법과의 비교분석을 수행하였다. 검증 문제는 코드 검증 시 널리 사용되고 있는 Transonic flow over a Bump in a channel 문제로 선정하였다.

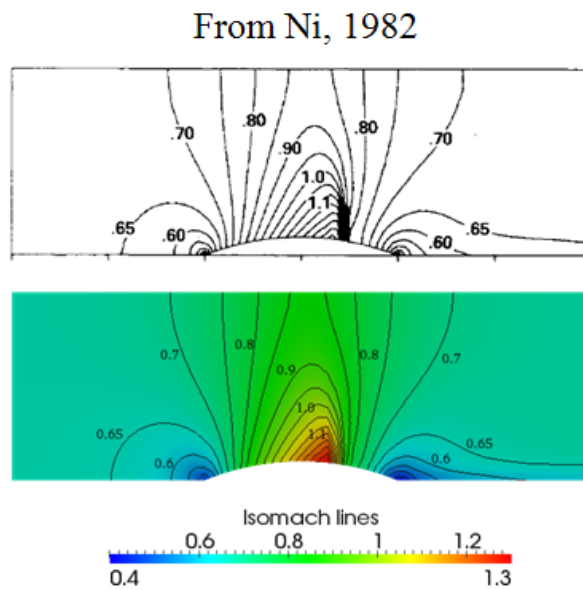


그림 17. Mach contours of Bump in a channel

검증문제에 대해 정상상태(Steady-state) 해석을 수행하였으며 아래 그림에서 볼 수 있듯이 주어진 조건에서 Residual이 Machine accuracy 영역까지 떨어지는 것을 확인할 수 있었다. 게다가 내재적인 방법인 LU-SGS 수치기법이 가지고 있는 빠른 수렴성과 높은 정확도의 특성을 반영하는 것을 확인할 수 있었다.

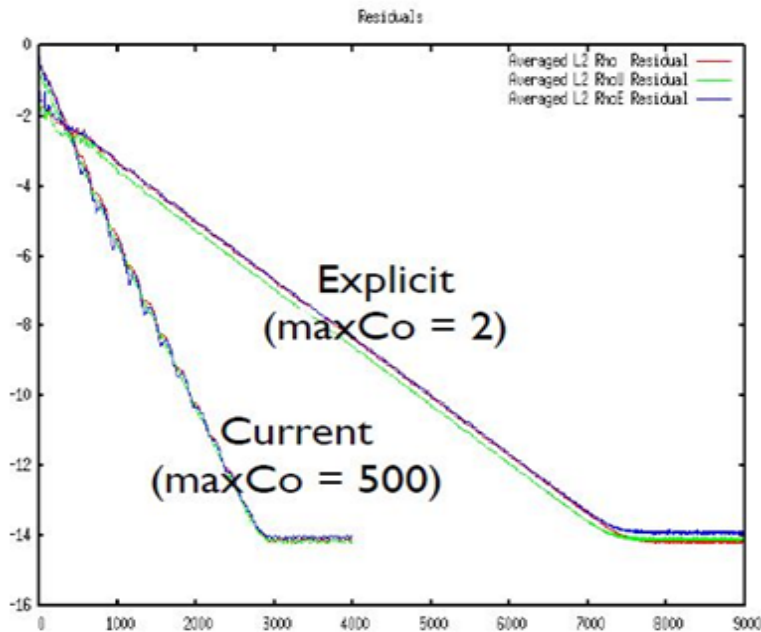


그림 18. Convergence history (Current means Implicit)

3.3 Characteristic Boundary Condition (Riemann Invariant)

3.3.1 Riemann Invariant

일반적으로 수치해석의 해를 구하기 위해서는 다양한 경계조건 및 초기조건들이 적용된다. 오픈폼(OpenFOAM)에서는 고정값 조건인 Dirichet 방식과 고정기울기 조건인 Neumann 방식을 기본으로 다양한 경계조건들을 제공하고 있다.[7]

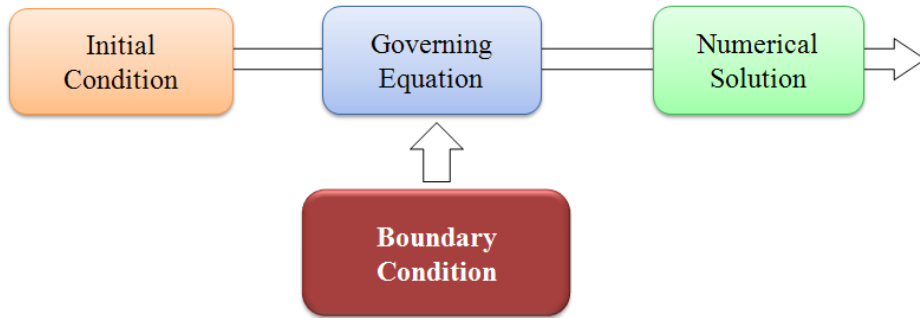


그림 19. Introduction to Boundary condition

본 연구에서는 오픈폼(OpenFOAM)에서 현재까지 제공하지 않았으며 항공분야에서 유용하게 사용되고 있는 대표적인 특성 경계조건인 리만 경계조건을 추가하고 이에 대한 검증을 수행하였다. 보존량을 Wave의 전과 방향에 따라 내삽 혹은 외삽하는 형태로 구성하였으며 이론적인 식은 다음과 같다.[23]

$$R^+ = U_{nB} + \frac{2C_B}{r-1}$$

$$R^- = U_{nB} - \frac{2C_B}{r-1}$$

이때 R^+ 는 전산해석영역으로 들어오는 Wave를 의미하며 이것은 외부조건을 이용하여 구하도록 코드를 구성하였다. 이와는 반대로 R^- 는 전산해석영역에서 나가는 Wave로 내부조건을 이용하여 구하도록 코드를 구성하였다. 기본적으로 리만 경계조건은 유입류와 유출류가 혼재되는 경우, 그 구분을 위해 사용되는데 이때 아음속에서 사용되는 유입류와 유출류의 정의는 아래의 식과 같다.

$$R^+ = U_{nB} + \frac{2C_B}{r-1} = U_{n\infty} + \frac{2C_\infty}{r-1} \quad (\text{Inflow})$$

$$R^- = U_{nB} - \frac{2C_B}{r-1} = U_{ni} - \frac{2C_i}{r-1} \quad (\text{Inflow})$$

$$R^+ = U_{nB} + \frac{2C_B}{r-1} = -U_{n\infty} + \frac{2C_\infty}{r-1} \quad (\text{Outflow})$$

$$R^- = U_{nB} - \frac{2C_B}{r-1} = -U_{ni} - \frac{2C_i}{r-1} \quad (\text{Outflow})$$

코드의 검증을 위해 아래와 같은 격자와 유동조건을 적용하였다. 해석격자는 상용격자 생성프로그램인 Gambit 2.4를 사용하였고 이를 오픈폼에서 사용할 수 있도록 격자를 전환하였다. 경계조건으로는 본 논문에서 오픈폼(OpenFOAM)에 추가시킨 리만 경계조건과 오픈폼(OpenFOAM)에서 제공하는 Free-stream 경계조건을 이용하여 비교분석을 수행하였다.

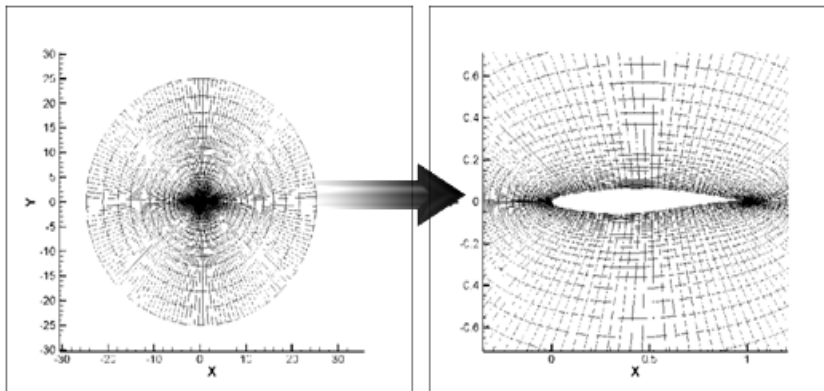


그림 20. Grid system for airfoil

Flow Conditions	
<i>Characteristic length</i>	<i>1 ft</i>
T_{∞}	255.5 K
P_{∞}	108987.393 Pa
M_{∞}	0.729
α	2.31 °

표 15. Flow conditions for airfoil

해석결과 아래의 그림과 같이 전산해석영역이 코드길이의 25배에서 10배로 감소함에 따라 기존에 오픈폼(OpenFOAM)에서 제공하는 경계조건(Free-stream)은 전산해석영역의 크기에 따라 오류를 발생하는 반면 개발된 리만 경계조건에서는 그 차이가 미미한 것을 확인할 수 있었다. 또한 수렴성의 차이에서도 그 차이가 분명한 것을 그림23에서 확인할 수 있다.

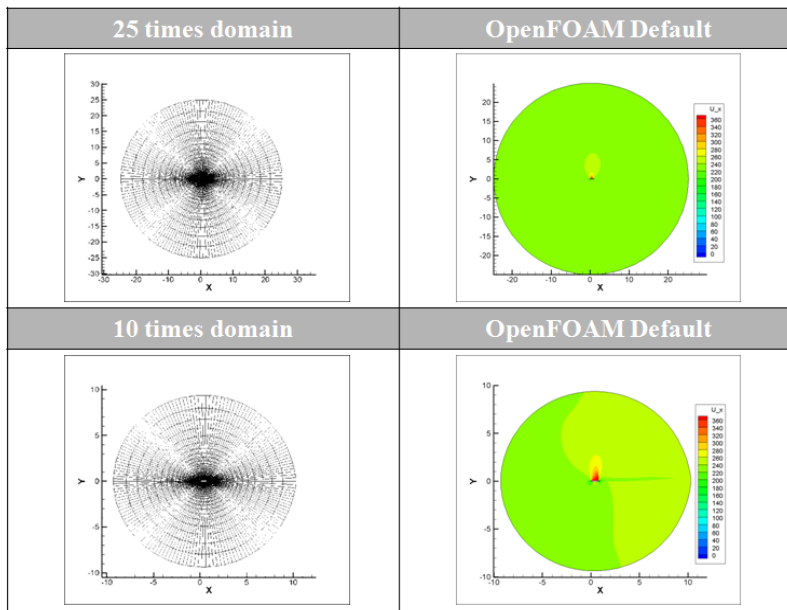


그림 21. OpenFOAM Free-stream Boundary condition

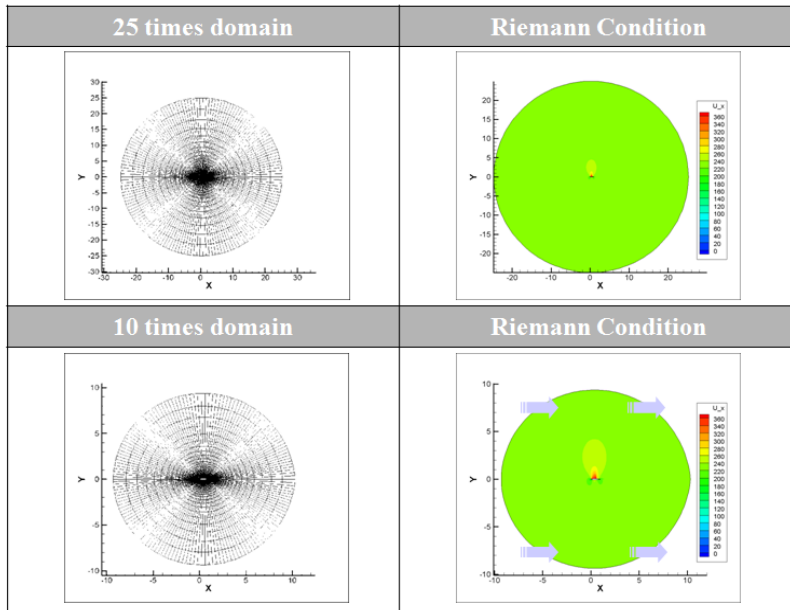


그림 22. Riemann Boundary condition in OpenFOAM

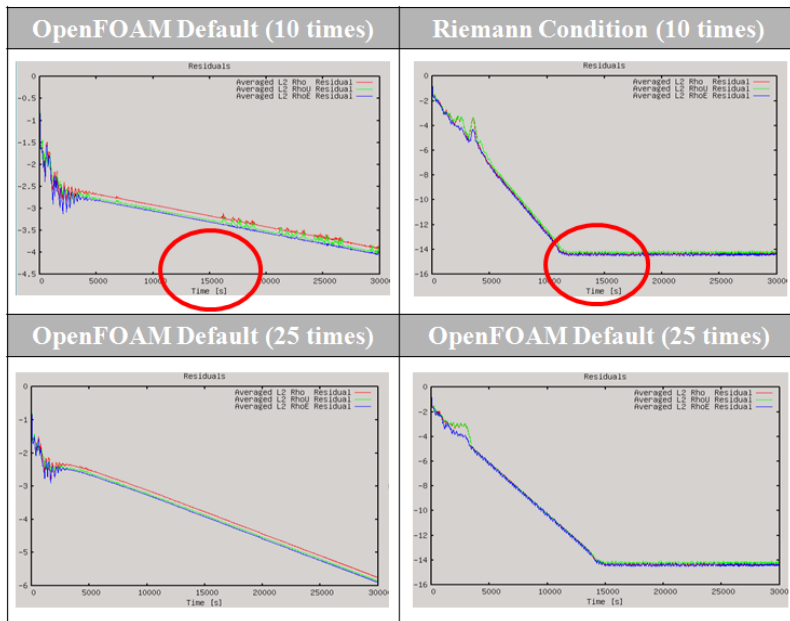


그림 23. Convergence history (Free-stream vs. Riemann)

3.4 Turbulence model validation

3.4.1 RAE-2822 Transonic Airfoil

천음속이란 음속에 거의 가까운 속도로, 보통 마하수 0.7에서 1.0에 이르는 속도를 일컫는다. 비행기 주변을 흐르는 공기가 마하수 1.0에 이르게 되면 충격파가 발생하게 되고 이는 곧 항력의 증가로 이어지게 된다. 이러한 이유 때문에 여객기는 보통 마하수 0.85에서 순항을 하게 되는데, 언급한 속도보다 비행속도가 조금만 증가해도 국부적으로 충격파가 발생하여 비행기의 효율적인 운영측면에서 큰 문제점으로 작용할 수 있다.[12]

본 논문에서는 앞서 개발된 압축성코드에 기본적으로 오픈폼(OpenFOAM)에서 제공하고 있는 k-omega SST 난류모델을 객체 라이브러리로 선택하여 RAE-2822 천음속 익형에 대해 전산 해석을 수행하고 이에 대한 검증을 시도하였다. 해석조건은 자유류 속도 마하 0.79 인 유동이 특성길이가 1인 조건에서 수치적으로 해석한 결과를 NASA의 전산해석 및 실험결과와 비교하였다. 해석을 위해 사용된 조건은 다음과 같다.

Flow Conditions	
<i>Characteristic length</i>	1 ft
T_{∞}	255.5 K
P_{∞}	108987.393 Pa
M_{∞}	0.729
α	2.31 °

표 16. Flow conditions used for RAE-2822 airfoil

계산에 사용된 격자계는 그림24와 같으며, 해석 결과 압력계수와 속도에 대한 결과 값이 NASA의 값과 비교적 잘 일치하는 것

을 확인할 수 있었다.

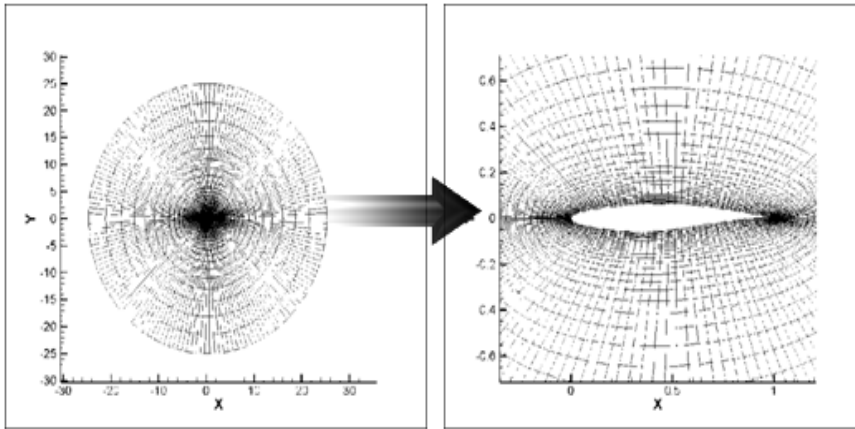


그림 24. Grid system used for RAE-2822 airfoil

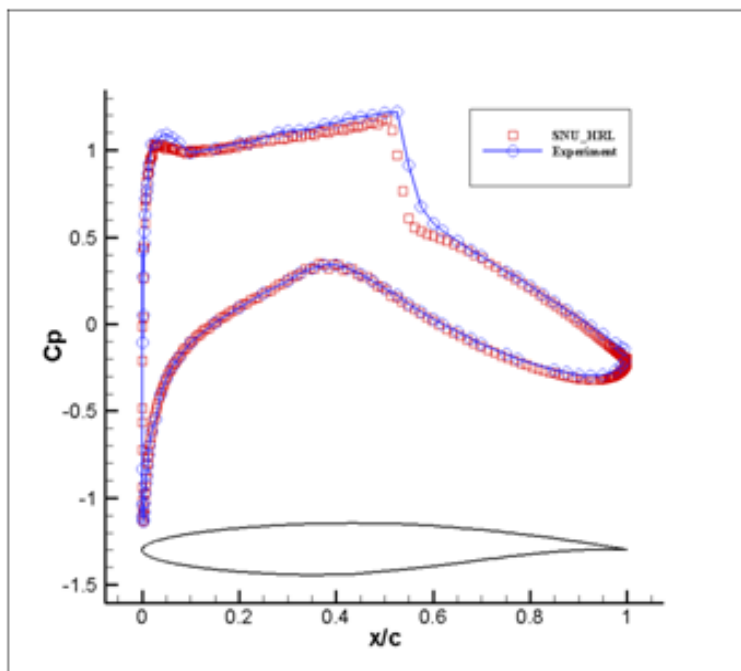


그림 25. Pressure coefficient on RAE-2822 airfoil

3.5 Automatic Wall Function

3.5.1 Wall treatment for a turbulence model

점성효과가 큰 벽면을 다룰 때는 보통 벽면으로 계산격자를 조밀하게 형성하여 해석을 수행한다.[19] 그러나 단순한 형상이 아닌 복잡한 형상에 대해서는 벽면으로 계산격자를 조밀하게 형성하는 것은 어려우며 이를 해결하기 위해 Von Karman이 제안한 벽법칙(Law of the wall)을 이용하기도 한다. 벽법칙은 일반적으로 다음과 같은 구분 하에 사용한다.

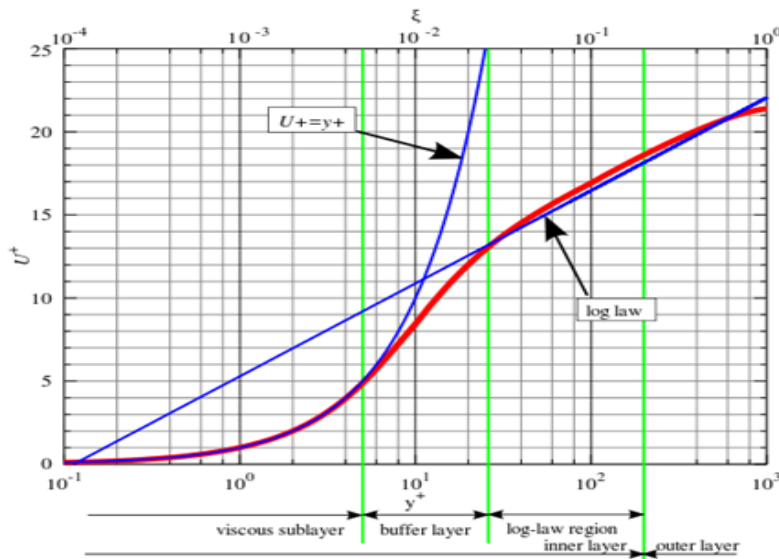


그림 26. Law of the wall

$$u^+ = y^+ ; \text{Viscous sub-layer}$$

$$u^+ = \frac{1}{k} \ln y^+ + C ; \text{Log layer}$$

Neither the law holds ; Buffer layer

3.5.2 k-omega SST Model

일반적으로 FLUENT/CFX와 같은 전산해석 상용프로그램은 난류모델 선정 시 벽법칙의 사용 유무를 결정할 수 있다. 그러나 오픈 소스 기반의 전산해석 프로그램인 오픈폼(OpenFOAM)에서는 난류모델 선정 시 반드시 벽법칙을 수반하게 되어 있다. 본 논문에서는 외부 공기역학(External Aerodynamics) 해석 시 널리 사용되고 있는 k-omega SST 난류모델을 선정하였다.

k-omega SST 난류모델은 일반적으로 Blending함수 F_1 을 도입하여 k-omega 난류모델 및 k-epsilon 난류모델을 결합시킴으로서 각각의 모델의 장점을 결합하여 다양한 유동에 대해서 정확한 난류해석을 가능하게 한다. k-omega SST 난류모델에서 에디(Eddy) 점성은 다음과 같이 Turbulent kinetic energy인 k 와 Specific dissipation rate인 w 의 함수로 나타낼 수 있으며 자세한 식은 다음과 같다.

$$\mu_t = \frac{\rho k / \omega}{\max[1, \Omega F_2 / a_1 \omega]}$$

$$F_2 = \tanh(\arg_2^2)$$

$$\arg_2 = \max\left(2 \frac{\sqrt{k}}{0.09\omega}; \frac{500\nu}{y^2\omega}\right)$$

$$\frac{\partial \rho k}{\partial t} + \frac{\partial}{\partial x_j} \left(\rho u_j k - (\mu + \sigma_k \mu_t) \frac{\partial k}{\partial x_j} \right) = \tau_{ij} \frac{\partial u_i}{\partial x_j} - \beta^* \rho \omega k$$

$$\frac{\partial \rho \omega}{\partial t} + \frac{\partial}{\partial x_j} \left(\rho u_j \omega - (\mu + \sigma_w \mu_t) \frac{\partial \omega}{\partial x_j} \right) = \frac{\gamma}{\nu_t} \tau_{ij} \frac{\partial u_i}{\partial x_j} - \beta \rho \omega^2 + 2(1 - F_1) \frac{\rho \sigma_{w2}}{\omega} \frac{\partial k \partial \omega}{\partial x_j \partial x_j}$$

$$\tau_{ij} = \mu_t \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3} \frac{\partial u_k}{\partial x_k} \delta_{ij} \right) - \frac{2}{3} \rho k \delta_{ij}$$

$$F_1 = \tanh \left[\left\{ \min \left(\max \left(\frac{\sqrt{k}}{0.09\omega y}, \frac{500\mu}{\rho\omega y^2} \right), \frac{4\rho\sigma_{w2}k}{CD_{kw}} \right) \right\} \right]$$

$$CD_{kw} = \max \left(2\rho\sigma_{w2} \frac{1}{\omega} \frac{\partial k}{\partial x_j} \frac{\partial \omega}{\partial x_j}, 10^{-20} \right)$$

3.5.3 Flat plate turbulent boundary layer

k-omega SST모델을 기반으로 벽법칙 부분을 수정한 코드의 검증을 위해 비압축성 유동에 대한 평판의 유동해석을 수행하였다. 난류모델은 k-omega SST를 선정하였으며 기존에 오픈폼 (OpenFOAM)이 제공하고 있는 벽함수를 사용했을 때와 수정된 벽함수를 사용했을 때, 전단응력과 같은 특정 물리 값을 가지고 비교분석을 수행하였다. Y+의 구분을 위해 격자는 아래의 표와 같이 설정하였다.

Grids Information			
GRID2	Y+1	GRID5	Y+20
GRID3	Y+5	GRID6	Y+30
GRID4	Y+10	GRID8	Y+70
GRID9	Y+90		

표 17. Grids information

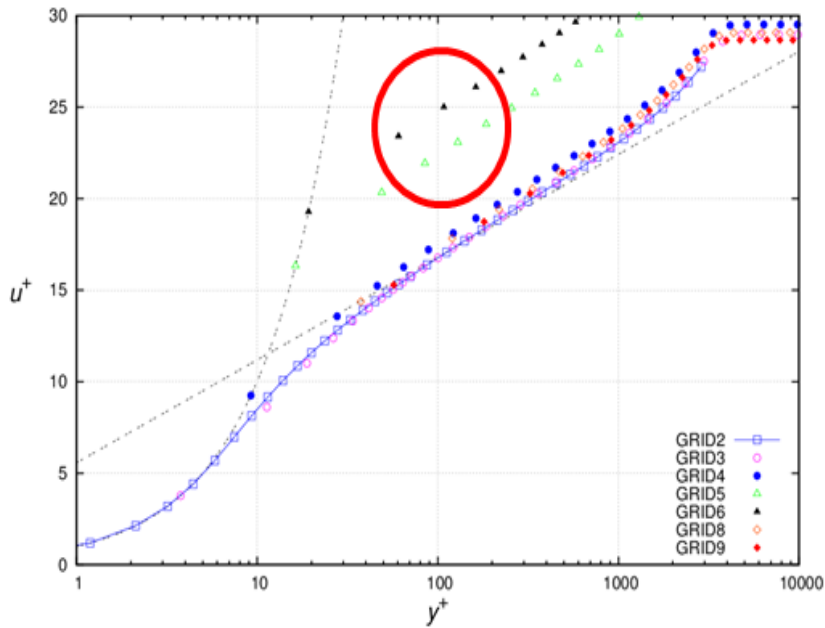


그림 27. OpenFOAM default wall treatment

위의 그림과 같이 기존에 오픈폼(OpenFOAM)이 제공하고 있는 벽함수를 사용한 난류모델은 y^+ 에 따라 u^+ 의 값이 특정구간인 Buffer layer 구간에서는 정확도의 측면에서 큰 오차를 기록하는 것을 확인할 수 있었다. 이는 곧 y^+ 값이 Log layer와 Viscous layer의 중간인 Buffer layer를 반영하지 못하고 있는 오픈폼(OpenFOAM)의 한계를 나타내고 있으며 이는 기존의 코드 구성자체가 if문으로 Log layer와 Viscous layer를 구분하고 있지만, 그 중간에 있는 Buffer layer에 대해서는 충분히 고려하지 못하고 있는 한계가 존재하기 때문이라 판단된다.

이에 따라 Spalding이 제안한 Universal 벽함수의 식을 코드 내에 수정함에 따라 y^+ 에 관계없이 u^+ 가 하나의 식으로 표현되도록 코드를 구성하였다.

$$\frac{u_{\tau}y}{v} = \frac{u}{u_{\tau}} + \frac{1}{E} \left[e^{\frac{ku_{\tau}y}{v}} - 1 - \frac{ku_{\tau}y}{v} - \frac{(ku_{\tau}y)^2}{2v^2} \right] - \frac{(ku_{\tau}y)^3}{6v^3}$$

이를 이용해 아래와 같은 식의 과정을 통해 y^+ 에 상관없이 원하는 일정한 값을 얻을 수 있었다.

$$\tau_w = (v + v_t) \left| \frac{dU}{dy} \right|$$

$$u_{\tau}^2 = (v + v_t) \left| \frac{dU}{dy} \right|$$

$$\therefore v_t = u_{\tau}^2 \left| \frac{dy}{dU} \right| - v$$

이러한 수정된 코드는 기존 오픈폼(OpenFOAM)코드에서 가지고 있었던 Buffer layer에서의 불일치를 개선하였고 이는 그림 29,30과 같이 일정한 점성계수 및 전단응력(Shear stress)를 예측하는데 있어서도 기존의 오픈폼(OpenFOAM)과의 확연한 차이를 보여주는 것을 확인할 수 있었다.

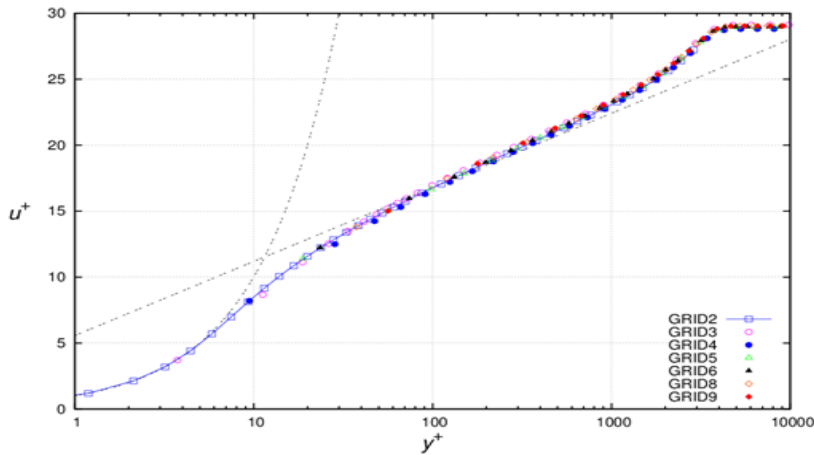


그림 28. OpenFOAM modified wall treatment

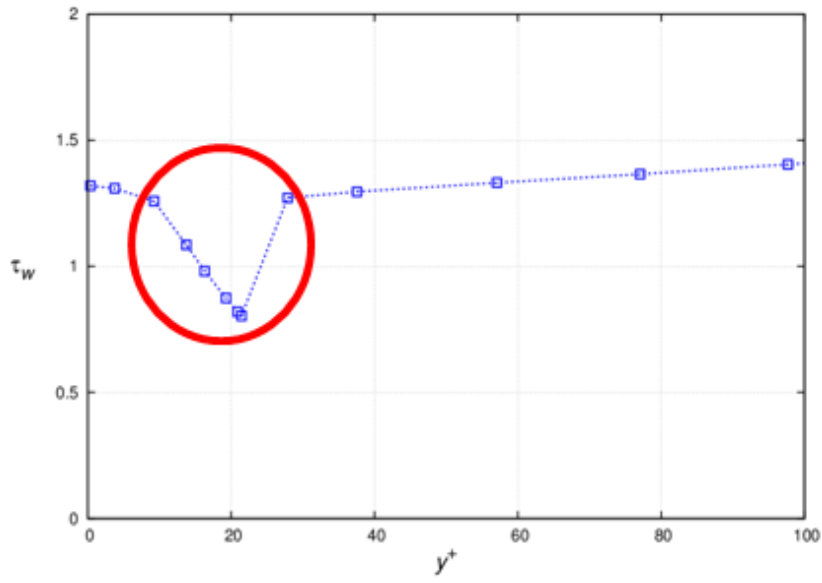


그림 29. Wall shear stress
(Original code wall treatment)

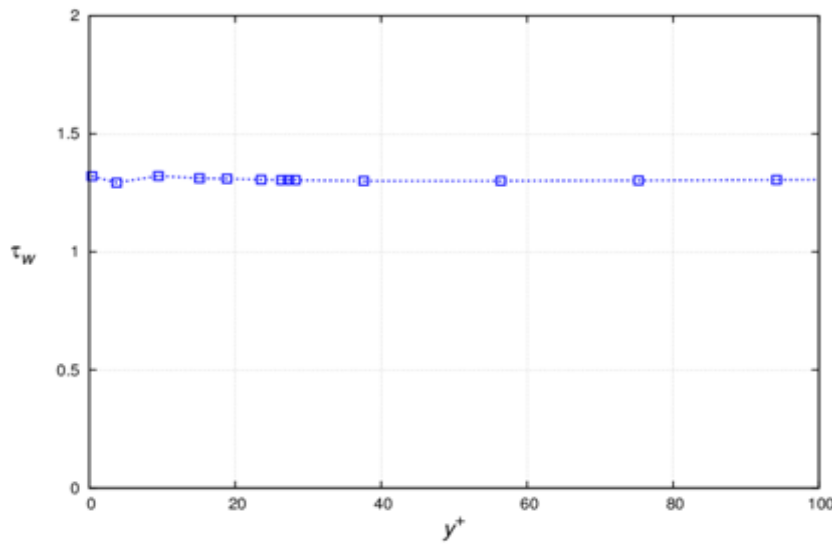


그림 30. Wall shear stress
(Modified code wall treatment)

4. Applications

본 논문에서는 압축성효과를 고려한 오픈폼(OpenFOAM)기반의 개선된 코드의 검증과 그 특성을 알아보기 위해 다음과 같은 여러 가지 문제에 대해 전산해석을 수행하였다. 수치기법으로는 Roe FDS, LU-SGS가 사용되었으며 Limiter는 Venkatakrisnan[4]이 사용되었다.

4.1 Shock Boundary Layer Interaction

4.1.1 해석조건 및 경계조건

충격파-경계층 상호작용문제는 압축성 및 점성유동에 관한 코드의 검증문제로 널리 사용되고 있으며[19], 본 연구에서 사용된 유동조건은 Hakkienen[21]의 실험조건과 동일하다.

Flow Conditions	
μ_{∞}	$1.78E-05 \text{ Ns/m}^2$
T_{∞}	288.815 K
P_{∞}	101325 Pa
M_{∞}	2
a_{∞}	340.28 m/s
<i>Characteristic length</i>	2
Re	2.96×10^5
Pr	0.72
<i>Impinging shock Angle</i>	32.585°
<i>Gas</i>	<i>Calorically perfect gas</i>

표 18. Flow conditions used of SWBLI

4.1.2 충격파-경계층 상호작용의 유동현상

충격파-경계층 상호작용(Shock Wave Boundary Layer Interaction, SWBLI) 문제는 아래의 그림과 같이 외부에서 생성된 경사충격파(Oblique shock)가 점성경계층내로 입사하는 문제로써 이러한 충격파-경계층 상호작용문제는 보통 충격파입사부근의 공력가열현상으로 인한 벽면의 열전달(Heat transfer) 증가와 역압력구배(Adverse pressure gradient) 형성으로 인한 유동박리현상(Flow separation)을 수반한다. 특히 극초음속(Hypersonic) 유동에서 공력가열현상(Aerodynamic heating)은 매우 중요한 문제로, 열전달로 인한 구조물의 파괴도 가져올 수 있기 때문에 이에 대한 해석이 필수적으로 요구된다.[25]

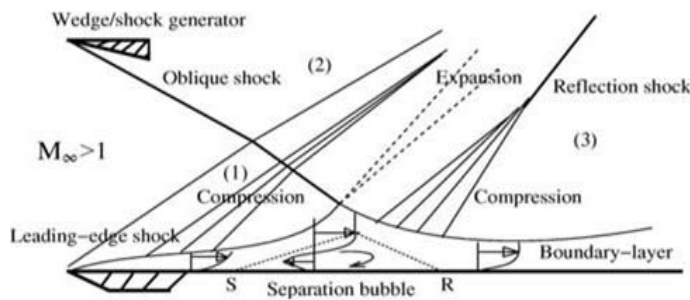


그림 31. Schematics of Shock B.L. Interaction

Hakkinen[21]의 실험은 충격파 입사각도(Impinging shock angle)가 32.585도인 충격파가 평판에 입사하는 대표적인 문제이다. 충격파가 입사한 영역에서는 급격한 압력증가를 가져오는 역압력구배가 형성되고, 경계층(Boundary layer) 내부에서는 점성의 영향에 의하여 유동의 운동량이 감소되어 작은 역압력구배에 의해서도 유동의 박리 현상이 발생하게 된다. 유동이 박리된 영역에서는 유동 박리에 의해 생기는 경계층의 급격한 증가로 인해

외부유동의 흐름이 바뀌게 되어 박리점(Separation point) 부근에서 충격파를 생성하게 된다. 또한 유동의 재부착점(Reattachment point)에서는 경계층 두께가 감소하게 되어 팽창파(Expansion)와 재 압축파(Compression)가 생기게 된다. 본 논문에서 선정한 검증문제는 극초음속 유동이 아니므로 점성소산 작용으로 생기는 온도의 증가가 작아 단열벽면조건(Adiabatic wall)을 만족하기 때문에 벽면 열전달과 같은 현상은 없는 것으로 설정하였다.

4.1.3 전산해석결과

본 논문에서는 기존의 In-house 코드(서울대학교 극초음속 및 희박기체 연구실 코드)와의 주된 비교대상을 입사충격파, 박리에 의한 충격파, 재부착에 의한 충격파와 같은 세 충격파를 정확히 잡아내는 것과 유동박리 영역을 정확하게 묘사하는 것을 기준으로 비교분석을 수행하였다. 해석에 사용된 계산격자와 해석조건은 다음과 같다.

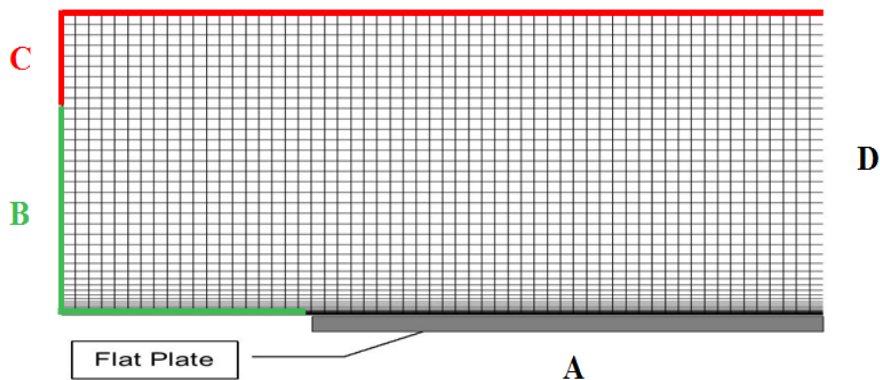


그림 32. Grid system used for SWBLI

Boundary Condition	
A	No slip wall
B	Inlet (1 condition)
C	Inlet (2 condition)
D	1st order extrapolation

표 19. Boundary conditions used for SWBLI

전산해석영역에서의 Impinging shock angle 32.585도의 모사를 위해 압축성유체역학의 이론식을 이용하여 경계조건 B,C에 각각 유동조건을 다르게 설정하였다. 사용된 이론식은 다음과 같다.[12]

$$\tan\theta = 2\cot\beta \frac{M_1^2 \sin^2\beta - 1}{M_1^2(r + \cos 2\beta) + 2}$$

$$\frac{P_2}{P_1} = \frac{2rM_1^2 \sin^2\beta - (r-1)}{r+1}$$

$$\frac{T_2}{T_1} = \frac{[2rM_1^2 \sin^2\beta - (r-1)][(r-1)M_1^2 \sin^2\beta + 2]}{(r+1)^2 M_1^2 \sin^2\beta}$$

$$\frac{\rho_2}{\rho_1} = \frac{(r+1)M_1^2 \sin^2\beta}{(r-1)M_1^2 \sin^2\beta + 2}$$

$$M_2^2 = \frac{(r-1)(M_1^2 \sin^2\beta) + 2}{[2rM_1^2 \sin^2\beta - (r-1)] \sin^2(\beta - \theta)}$$

Oblique shock relation	
$\frac{P_2}{P_1}$	1.186836216
$\frac{T_2}{T_1}$	1.050311456
$\frac{\rho_2}{\rho_1}$	1.129984652
M_2	1.88915403
Impinging shock angle	32.585

표 20. The results from Oblique shock relation

그림 31,32는 In-house 코드와 개발된 오픈폼(OpenFOAM)기반의 압축성 코드로 각각 해석한 Pressure 및 Skin Friction Coefficient의 그래프이다. 개발된 오픈폼(OpenFOAM)기반의 압축성 코드가 기존의 In-house 코드와 비교하였을 때 비교적 일치하는 것을 확인할 수 있었지만 박리 영역에서의 정확성은 다소 떨어지는 것을 확인할 수 있었다.

이는 곧 공간차분의 정확도의 차이로 판단되어 지며, 기존의 In-house 코드는 정렬기반의 코드로 3차 이상의 정확도를 갖는 반면 오픈폼(OpenFOAM)기반의 압축성 코드는 비정렬코드 기반으로 2차 정확도를 갖기 때문으로 판단된다. 또한 그림 33,34는 각각의 코드로 계산한 압력분포를 나타내며 박리 지점에서의 정확성은 다소 떨어지지만 해의 정확도면에서는 비교적 정확한 값을 예측하는 것을 확인할 수 있었다.

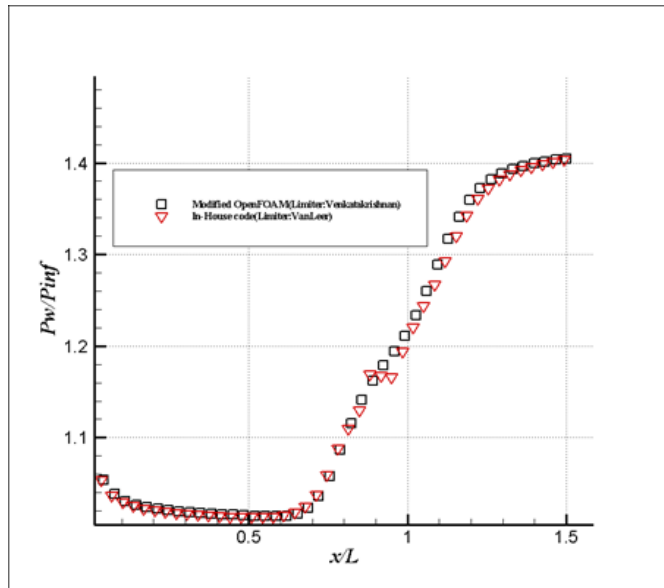


그림 33. In-house vs. Modified OpenFOAM(Cp)

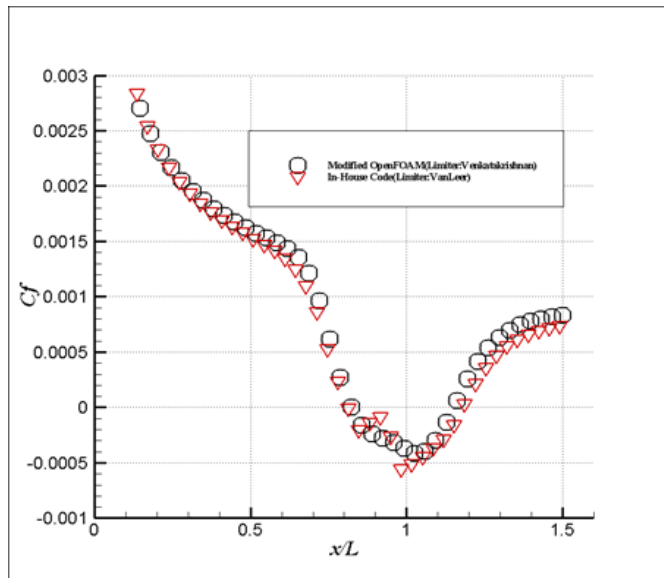


그림 34. In-house vs. Modified OpenFOAM(Cf)

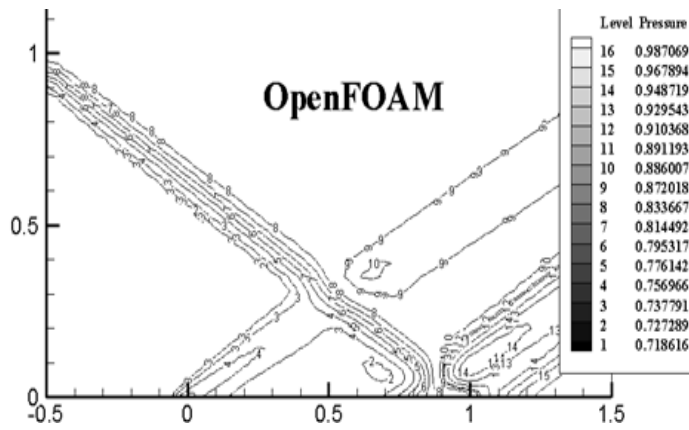


그림 35. Pressure contour (Modified OpenFOAM)

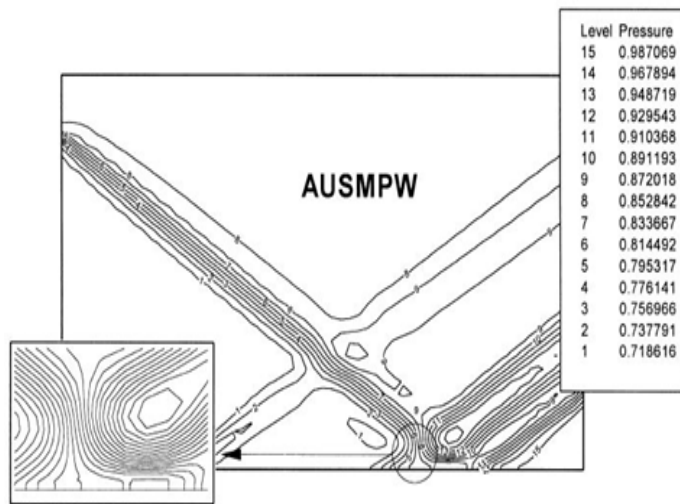


그림 36. Pressure contour (In-house code)

4.2 Flat Plate

4.2.1 해석조건 및 경계조건

개발된 오픈폼(OpenFOAM)기반의 압축성 코드의 검증을 위해 간단한 형상을 가진 평판을 선정하고 기존의 In-house코드와의 비교분석을 수행하였다. 해석조건은 자유류 속도 마하 8 인 유동이 특성길이가 1이고 표면 온도가 1000K 인 평판 위를 지날 때 고도가 약 15km 인 조건에서 수치적으로 해석한 결과를 Van Driest의 결과와 비교하였다.[13] 해석을 위해 사용된 조건은 다음과 같다.

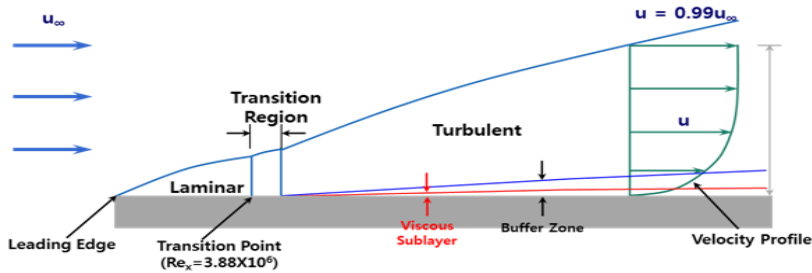


그림 37. Schematic of the flat plate with flow geometry

Flow Conditions	
μ_{∞}	$1.4216E-05 \text{ Ns/m}^2$
T_{∞}	216.65 K
P_{∞}	12111.4 Pa
M_{∞}	8
ρ_{∞}	0.193919 kg/m^3
a_{∞}	295.07 m/s

표 21. Flow conditions used for flat plate

계산에 사용된 격자계의 크기는 80x160x3 이며 코드의 정확성을 위해 유동방향을 x,y,z으로 변화 시켜 해석을 수행하였다. 이를 그림으로 나타내면 다음과 같다.

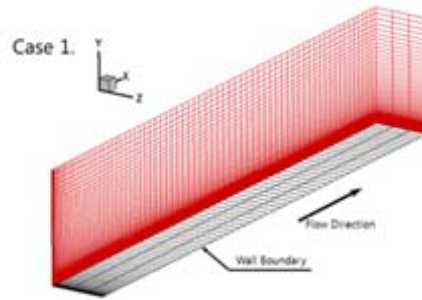


그림 38. X direction

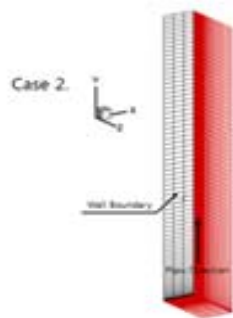


그림 39. Y direction

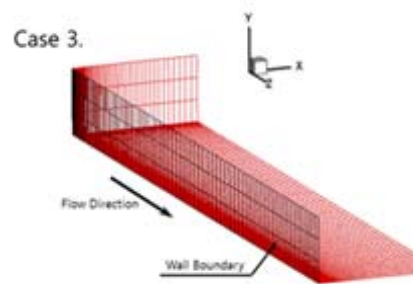


그림 40. Z direction

4.2.2 전산해석결과

해석 결과 아래의 그림과 같이 In-house 코드와 마찬가지로 평판의 시작부에서 충격파가 발생하는 것을 확인할 수 있었다.

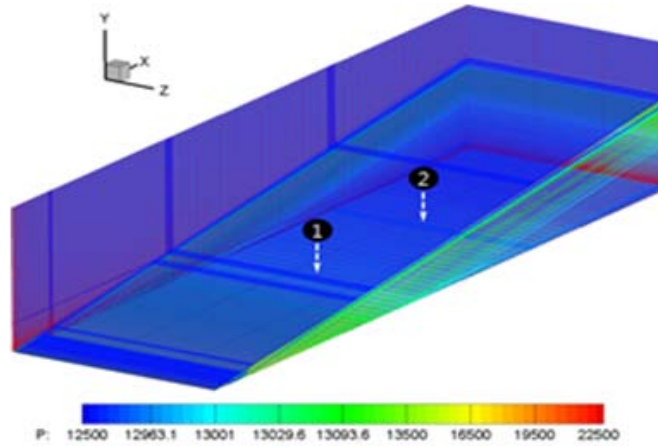


그림 41. Pressure contour (In-house code)

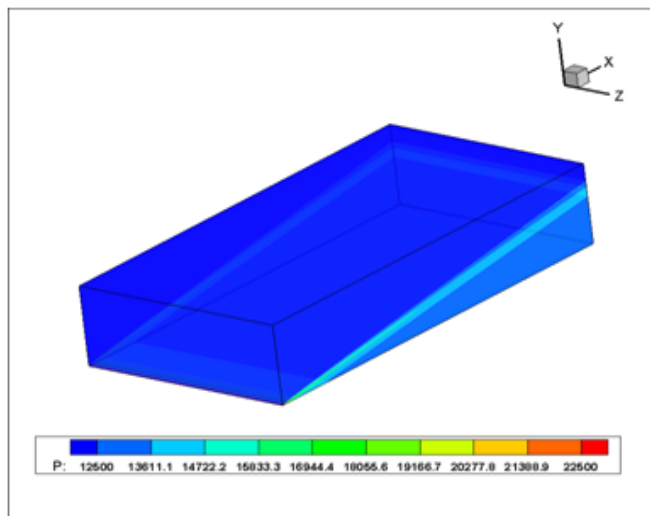


그림 42. Pressure contour (Modified OpenFOAM)

또한 유동방향을 x,y,z 으로 변화시킨 층류의 경우와 x 방향으로의 난류의 경우에도 Van Driest의 결과와 비교했을 때 정확하게 일치하는 것을 확인할 수 있었다.

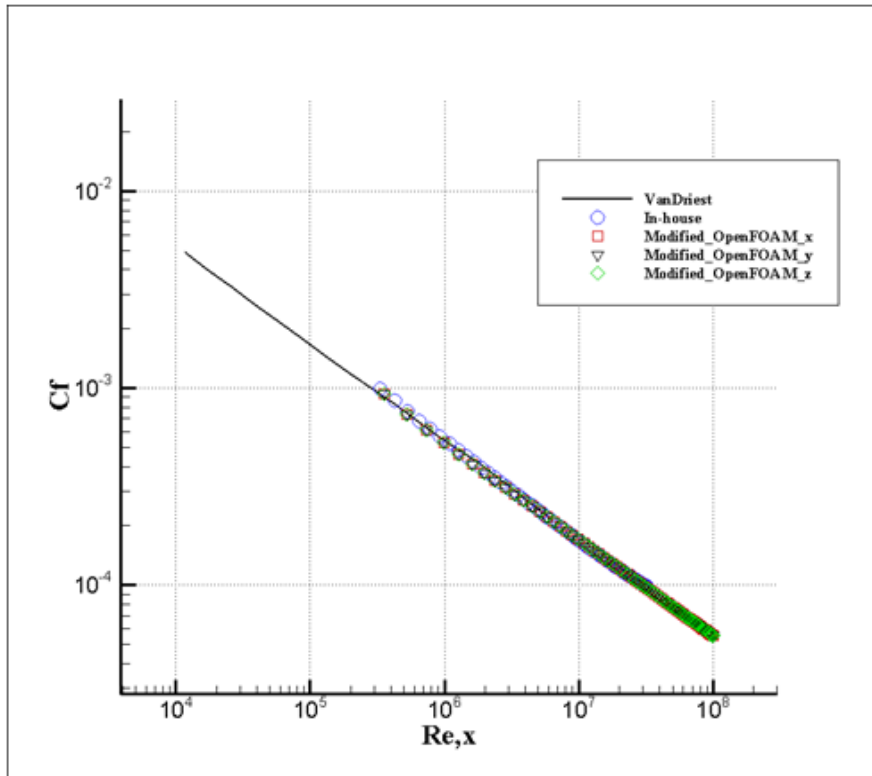


그림 43. Skin friction coefficient (Laminar flow)

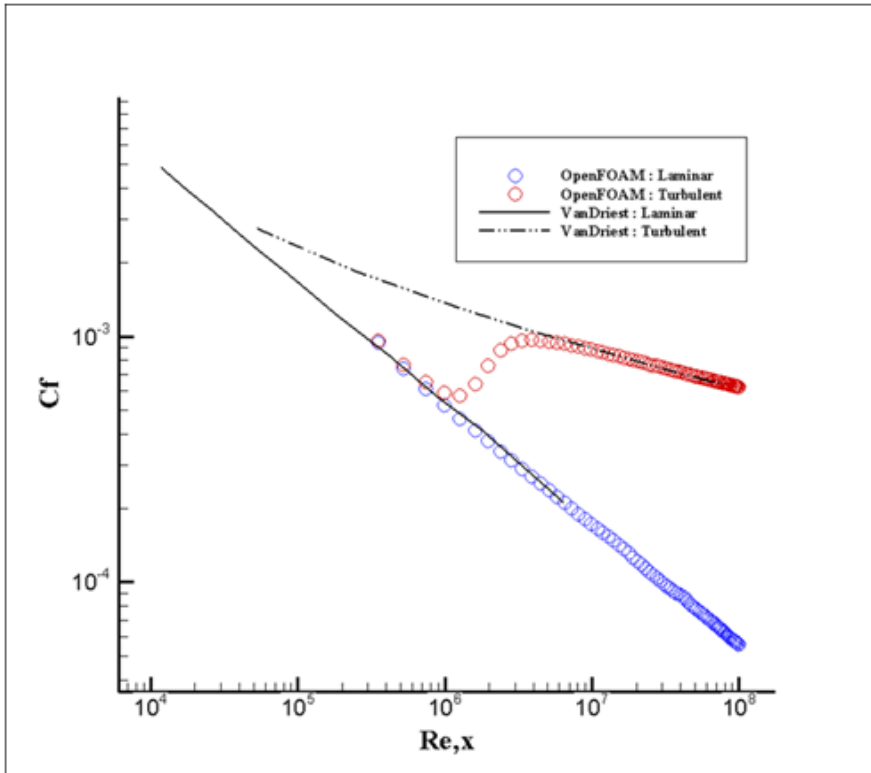


그림 44. Skin friction coefficient (Turbulent flow)

4.3 Aircraft-1

4.3.1 해석조건 및 경계조건

개발된 오픈폼(OpenFOAM)기반의 압축성 코드의 검증을 위해 현재 대한민국에서 개발하고 운용중인 초음속 비행기를 선정해 압축성 유동에 대한 해석을 수행하였다. 사용된 해석조건 및 경계조건은 다음과 같으며 보안상의 문제로 비행기의 이름은 편의상 “Aircraft-1” 이라고 칭한다.

Flow Conditions	
μ_{∞}	$1.731E-05 \text{ Ns/m}^2$
T_{∞}	273.3 K
P_{∞}	36580.52 Pa
M_{∞}	0.9
ρ_{∞}	0.4663062 kg/m^3

표 22. Flow conditions used for the Aircraft-1

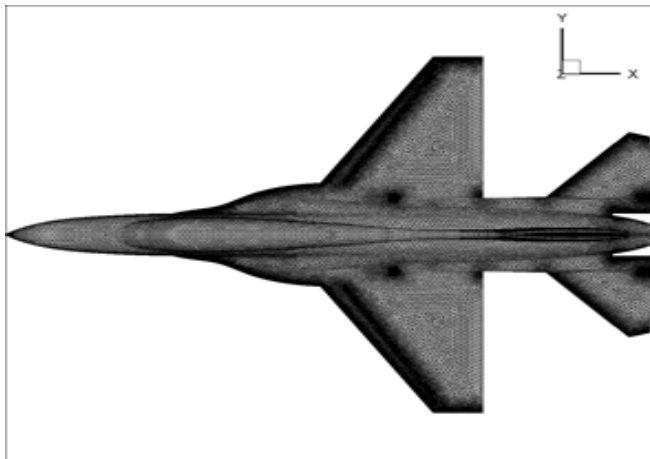


그림 45. Grid system used for the Aircraft-1

그림45는 3차원 전산유체해석을 위하여 형성된 격자계로서 상용격자 생성 프로그램인 Gambit 2.4과 T-grid를 사용하여 1개의 대칭면을 갖는 약 240만개의 비정렬 격자로 구성하였다. Aircraft-1의 3차원 날개를 해석하기 위한 수치해석 도구로는 개발된 오픈폼(OpenFOAM)기반의 압축성 코드와 상용 프로그램인 FLUENT 13.0을 사용하여 비교분석을 수행하였다.

4.3.2 전산해석결과

그림46은 상용 프로그램인 FLUENT로 해석한 결과와 개발된 오픈폼(OpenFOAM)기반의 압축성코드로 해석한 결과를 비교한 그림이다. 비행기의 공력특성예측에서 중요한 비행기 표면압력 값을 예측하는 데 있어서 FLUENT와 OpenFOAM에 큰 차이가 없는 것을 확인할 수 있다. (보안 상 구체적인 압력 값은 생략하였다.)

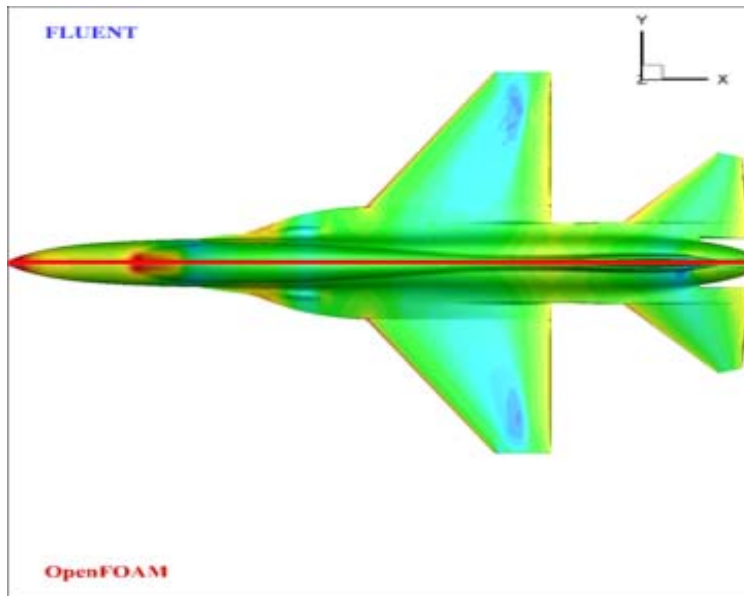


그림 46. Pressure contour (OpenFOAM vs. FLUENT)

전산해석결과의 보다 높은 정확성을 검증하기 위해 Aircraft-1에 대한 풍동 실험값과 전산해석을 통해 얻어진 공력계수 값에 대해 비교분석을 수행하였다. 보안상의 문제로 값을 무차원화 하였고, 그 결과는 아래의 표와 같다.

AoA = 1	Lift coefficient	Drag coefficient
OpenFOAM	1.07	1.05
Experiment	1	1
Error [%]	7.6	5.1

표 23. Aerodynamic coefficients

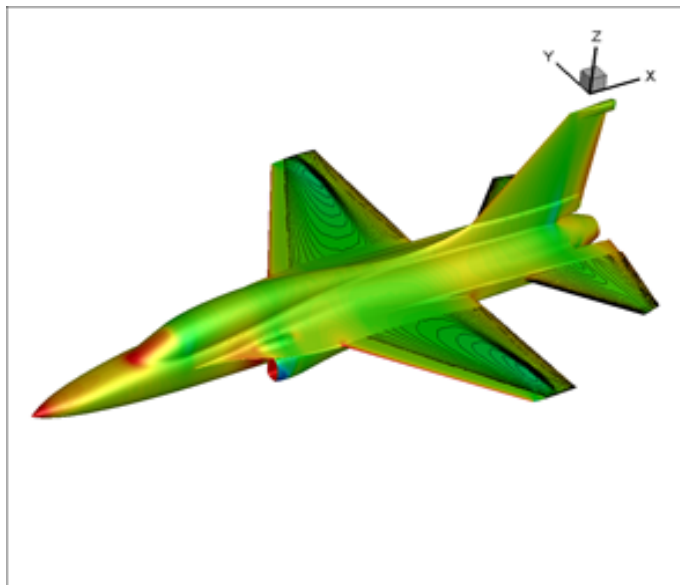


그림 47. Pressure contour (OpenFOAM)

위의 그림과 표에 대한 결과는 특정한 한 각도에 대한 해석결과이다. 각도가 높아질수록 박리현상에 의해 공력특성의 예측이 어렵다는 점을 감안했을 때, 각도가 낮은 영역에서는 그 오차가 더 줄어들 것으로 기대된다. 게다가 개발된 코드의 정확성 문제

를 보다 정량적으로 개선해 나간다면, 향후 오픈소스기반의 전산 해석 프로그램인 오픈폼(OpenFOAM)이 유지 및 보수비용이 상당한 FLUENT/CFX와 같은 상용프로그램의 대체 프로그램으로 사용될 수 있을 것으로 판단된다.

5. 결론

본 논문에서는 In-house 코드와 상용프로그램이 가지는 각각의 단점을 보완하고자 오픈소스기반의 전산유체역학 프로그램인 오픈폼(OpenFOAM)을 제안하였으며 오픈폼(OpenFOAM)에 압축성 유동에 대한 해석의 정확성을 높이기 위해 기존의 지배방정식을 밀도 기반의 지배방정식으로 수정하였고, LU-SGS(Lower Upper Symmetric Gauss-Seidal) 수치기법을 적용하여 압축성코드를 개발하였다. 개발된 내용을 요약하면 다음과 같다.

(1) Density-based solver (Conservative form of N-S Equation)

일반적으로 오픈폼(OpenFOAM)은 비압축성유동에 대한 해석을 기반으로 개발되었기 때문에 비압축성유동에 대해서는 높은 정확도와 안정성을 가지고 있는 반면 압축성유동에 대해서는 해석자의 개발과 공식적인 검증이 이루어지지 않았다고 볼 수 있다. 이에 따라 본 논문에서는 기본적으로 압력기반으로 개발된 오픈폼(OpenFOAM)의 지배방정식을 밀도기반의 보존형 지배방정식으로 수정하였고 기존의 오픈폼(OpenFOAM) 해석자와 이론값과의 비교분석을 통해 코드에 대한 검증을 수행하였다.

(2) Implicit time integration (LU-SGS)

계산의 효율성과 안정성을 위해 Runge-Kutta 와 같은 Explicit 시간적분 방법을 LU-SGS를 적용한 Implicit 시간적분 형태로 수정하였다. 동일한 문제(Transonic flow over a Bump in a channel)에 대해 같은 조건으로 해석을 수행하였으며 Explicit 방법에 비해 빠른 수렴성과 높은 안정성을 확인할 수 있었다.

(3) Automatic wall function (k-omega SST Turbulence model)

보통 벽면 근처의 난류유동에 대한 해석을 위해서는 격자를 조

밀하게 형성하거나 벽함수(Wall function)를 사용하게 된다. 기존에 오픈폼(OpenFOAM)에서 제공하고 있는 벽함수는 Buffer layer 구간에서의 상당한 오차를 가져왔고, 이를 해결하기 위해 Spalding이 제안한 Universal Function을 추가하여 벽함수를 수정하였다. 그 결과 평판 해석 시 전단응력과 같은 물리적인 값이 y^+ 에 Independent 한 값을 갖는 것을 확인할 수 있었다.

(4) Characteristic boundary condition (Riemann Invariant)

현재까지 오픈폼(OpenFOAM)의 경계조건에는 특성경계조건방정식이 포함되어 있지 않은 것으로 알려져 있다. 압축성유동의 해석뿐만 아니라 항공기와 같은 External Aerodynamics 해석을 위해 대표적인 특성경계조건방정식인 리만 경계조건을 추가하였고 기존에 오픈폼(OpenFOAM)에서 제공하고 있는 경계조건 (Free-stream)과 비교함에 따라 그 활용가능성을 확인하였다.

(5) Library modification (Wall shear stress and Etc.)

앞서 (1)에서 언급한 바와 같이 오픈폼(OpenFOAM)은 비압축성유동에 대한 해석을 기반으로 개발되었기 때문에 후처리 과정 (Post-processing)에서 사용되는 많은 객체 Library들이 비압축성의 물리적인 특성을 고려하여 개발되어 있는 상황이다. 이를 해결하기 위해 일부 객체 Library에 대해 압축성효과를 고려한 수정이 이루어졌으며 그 결과 원하는 물리적인 값을 얻을 수 있었다.

이와 같이 개발된 코드를 바탕으로 코드검증 시 널리 사용되고 있는 검증모델과 대한민국에서 개발하고 운용중인 비행기를 선정하여 이를 기존의 오픈폼(OpenFOAM)해석자와 In-house코드의 결과, 그리고 풍동실험값과 비교함에 따라 검증을 수행하였고 개발된 코드의 활용가능성을 확인하였다. 이를 통해 개발된 코드의

정확성 문제를 보다 정량적으로 개선해 나간다면, 향후 오픈소스 (Open source)기반의 전산해석 프로그램인 오픈폼(OpenFOAM)이 유지 및 보수비용이 상당한 상용프로그램인 FLUENT의 대체 프로그램으로 사용될 수 있을 것으로 기대된다.

6. 참고 문헌

- [1] J.H.Kim, C.F.Lange, C.R.Koch, A computational study of a circular cylinder at low Reynolds number for open loop control of Von Karman vortex shedding, 8th International OpenFOAM conference, 2013.
- [2] Williamson, C.H.K., Oblique and parallel modes of vortex shedding in the wake of a cylinder at low Reynolds number, J.Fluid Mech., 1989, p579-627.
- [3] Park D.S., Hendricks, Feedback control of Von Karman vortex shedding behind a circular at low Reynolds number, Phys. Fluids, 1994, p2390-2405.
- [4] Oliver Borm, Aleksandar Jemcov, Hans-Peter Kau, Density based Navier-Stokes solver for transonic flows, 6th International OpenFOAM conference, 2011.
- [5] R.F.Chen, Z.J.Wang, Block lower-upper symmetric Gauss-Seidel scheme for arbitrary grids, AIAA journal, 2000.
- [6] OpenFOAM User guide
- [7] OpenFOAM Programmer guide
- [8] Luis f. Gutierrez Marcantoni, Jose P. Tamagno, Sergio A. Elaskar, High speed flow simulation using OpenFOAM, Argentina journal, 2012.
- [9] Oliver Borm, Transonic density based flow solver, 5th International OpenFOAM conference, 2010.
- [10] P.L. Roe, Approximate Riemann solvers, parameter vectors, and difference schemes, Journal of computational physics, 1981, p43:357-372.
- [11] Catherine M. Maksymiuk, Thomas H. Pulliam, Viscous transonic airfoil workshop results using ARC2D, AIAA journal,

1987.

- [12] John. D.Anderson, Fundamentals of Aerodynamics 4th., McGraw-Hill.
- [13] Christopher J. Roy, Frederick G. Blottner, Methodology for turbulence model validation - Application to hypersonic flows, Journal of spacecraft and rockets, 2003.
- [14] T.W.Kim, S.J.Oh, K.J.Yee, Verification of the open source code, OpenFOAM to the external flows, KSAS journal, 2011.
- [15] Hrvoje Jasak, OpenFOAM: Introduction, capabilities and HPC needs, Cyprus advanced HPC workshop, 2012.
- [16] J.H.Kim, K.H.Kim, A development and verification of density based implicit Navier-Stokes solver using LU-SGS algorithm in OpenFOAM, APISAT, 2013.
- [17] MacCormack R.W., A numerical method for solving the equation of compressible viscous flow, AIAA journal, 1982, p1275-1281.
- [18] Hoffmann, K.A., Computational fluid dynamics for engineers, Engineering education system, 1989.
- [19] John. D.Anderson, Hypersonic and high temperature gas dynamics, McGraw-Hill, 1989.
- [20] White, F.M., Viscous fluid flow, McGraw-Hill, 1991.
- [21] Hakkinen, R. J., Greber, I. Trilling, The interaction of an oblique shock wave with a laminar boundary layer, Fluid dynamic research group, M.I.T., 1957.
- [22] C.F.Lange, Numerical Predictions of Heat and Momentum Transfer from a cylinder in Crossflow with Implications to Hot-Wire Anemometry, Ph.D.Thesis, 1997.
- [23] John D. Anderson, Computational Fluid Dynamics, McGraw-Hill, 1995.

- [24] John C. Tannehill, Computational fluid mechanics and heat transfer, Taylor&Francis, 1997.
- [25] 김규홍, 압력가중합수를 이용한 AUSM 수치기법 개선에 관한 연구, 서울대학교 석사학위논문, 1997.
- [26] 노오현, 점성유체역학, 2006.
- [27] 노오현, 압축성유체역학, 2004.
- [28] 박준권, 강관형, Source 공개 코드 OpenFOAM에 대한 리뷰, 전산유체공학회지, 2010, p46-53.
- [29] 신훈범, 정성기, 김지홍, 길재홍, 김병윤, 김정현, 김규홍, 오픈소스(OpenFOAM)에 기반한 내재적 방법의 압축성 전산유체역학 프로그램(ISACC)의 개발, 한국항공우주학회, 2013.
- [30] 김정현, 김규홍, C.F.Lange, C.R.Koch, 오픈폼(OpenFOAM)을 활용한 본 카르만 와류의 오픈 루프 및 피드백 제어를 위한 원형 실린더 주위의 전산해석 연구, 한국항공우주학회, 2013.
- [31] 김정현, 김규홍, 김지홍, 신훈범, 김병윤, 길재홍, 오픈폼(OpenFOAM)에 LU-SGS 알고리즘을 이용한 밀도 기반의 내재적 Navier-Stokes solver 코드 개발 및 검증, 한국항공우주학회, 2013.
- [32] 박원규, 전산유체역학1, 부산대학교 강의자료, 1999.

ABSTRACT

A development and verification of density based implicit solver using LU-SGS(Lower Upper Symmetric Gauss Seidel) algorithm in OpenFOAM(Open Field Operation And Manipulation) was performed.

A pressure-based solver in OpenFOAM for solving incompressible flow was modified to density-based solver for dealing with compressible flows. It was not only developed implicit LU-SGS algorithm instead of an explicit time integration in OpenFOAM but also implemented Riemann boundary condition which has not been developed in OpenFOAM. In addition, libraries such as wall shear stress dictionary in OpenFOAM were modified to solve and deal with compressible problems.

To validate the developed code, some validation models which are widely used were analyzed. Preliminary results showing the comparison between an experiment/reference and computation data indicated that our setup in OpenFOAM was correct.

Key Words : OpenFOAM, Open Source, LU-SGS, CFD

Student Number : 2012-20655

감사의 글

어느 덧 관악에서의 석사생활 2년도 지나갔습니다. 계속 배우면서 나를 갖추어 나가다보면 언젠가는 나에게도 기회가 올 것이라는 기대와 배움을 향한 갈망을 가지고 한 곳만 보고 달려왔던 순간이 었 그제 같은데 벌써 저도 졸업이라는 것을 하나 봅니다.

가장 먼저 항상 좋은 길로 인도해주시고 생각했던 것보다 더 큰 은혜 주시는 하늘의 아버지께 감사드립니다. 또한 육신의 아버지로 그 누구보다 자녀교육에 힘쓰셨던 나의 아버지, 본을 보여주셨던 것처럼 겸손하며 정직하게 그리고 성실하게 정도를 걸으며 살아가겠습니다. 사랑하는 어머니, 매일 새벽을 깨우며 아들을 위해 기도했던 그 은혜 잊지 않고 살아가겠습니다. 누구보다 든든한 누나, 동생의 기쁜 소식에 진심으로 함께 기뻐했던 순간들을 기억하며 더 열심히 살아가겠습니다. 항상 내편에 서서 나를 응원해준 신애, 꿈이 많았던 한 아이를 기다려주고 이해해줘서 너무 고맙습니다.

때로는 엄하게 때로는 칭찬으로 부족했던 저를 지도해주신 김규홍 교수님 그 은혜 평생 잊지 않고 살아가겠습니다. 교수님께 배웠다는 자부심 가지고 열심히 살아가겠습니다. 학부 시절 어머니의 마음으로 항상 저를 아껴주셨던 안존 교수님 그리고 아버지의 마음으로 지도해주신 이경태 교수님, 교수님들이 없이는 지금의 저도 없을 것 같습니다. 그 은혜 평생 갚으며 살아가겠습니다.

부족했던 저를 웃으며 감싸준 극초음속 연구실 동료들(가람이형, 준우형, 진선이, 평기, 병현이형, 준희형, 중근이형, 세일이형, 성민이, 진영이, 성준이, 재운이, 상아, 상준이, 원석이, 정호형, 정현이, 태경이형, 상훈이형) 진심으로 감사합니다. 연구실에서 만들었던 추억들 평생 간직하며 살아가겠습니다.

오픈폼(OpenFOAM)이라는 것을 소개시켜주시고, 진심으로 지도 해주신 C.F.Lange, C.R.Koch 교수님 감사합니다. 보여주신 학생과 가르침에 대한 열정들, 가슴 속에 새기고 살아가겠습니다.

코드 개발과정에서 많은 도움을 주셨던 김지홍 팀장님, 김병운 사장님, 길재홍 차장님, 신훈범 선임님 진심으로 감사합니다.

다 표현할 수는 없지만, 항상 저의 편에 서있는 고마운 친구들 그리고 교회식구들 진심으로 감사합니다.

2014년 2월 3일
한국항공우주연구원에서
김정현 올림