

# 오픈소스 기반 미사일 공력해석 자동화

유강국, 이상철



# 목차

---

- 서론
- 공력 해석 자동화
- Benchmark Test
- 결론

## ❖ 미사일 공력 설계

- 다양한 미사일 형상 타입
  - ✓ Body 형상, Finset 개수
- 넓은 유동조건 범위
  - ✓ 아음속 ~ 초음속, 고 받음각



## ❖ 반 경험식 기반 solver

- 텍스트 기반 입력 파일 사용
- 적은 계산 시간
- 부정확한 결과 도출

```
CASEID Missile Geometry
```

```
DIM M
```

```
DERIV RAD
```

```
$FLTCON
```

```
  NALPHA = 4.00000,
```

```
  ALPHA = 0.00000, 1.00000, 2.00000, 3.00000,
```

```
  NMACH = 4.00000,
```

```
  MACH = 0.60000, 1.20000, 1.50000, 2.00000,
```

```
  ALT = 0.00000,
```

```
  PHI = 45.00000,
```

```
$END
```

# 서론

## ❖ 반 경험식 기반 solver

- 텍스트 기반 입력 파일 사용
- 적은 계산 시간
- 부정확한 결과 도출

CASEID Missile Geometry

DIM M

DERIV RAD

\$FLTCON

NALPHA = 4.00000,

ALPHA = 0.00000, 1.00000, 2.00000, 3.00000,

NMACH = 4.00000,

MACH = 0.60000, 1.20000, 1.50000, 2.00000,

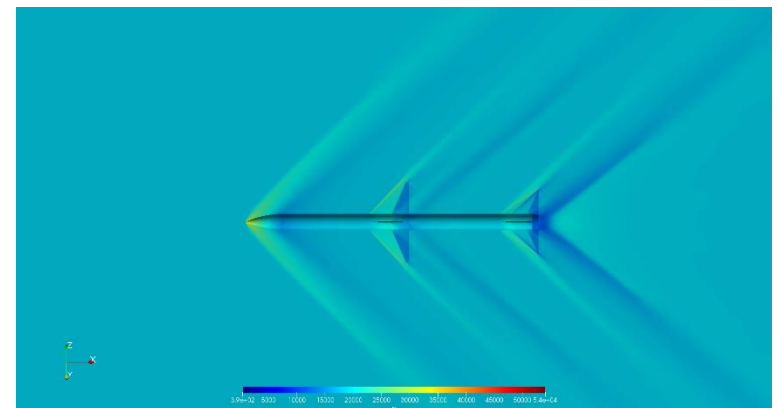
ALT = 0.00000,

PHI = 45.00000,

\$END

## ❖ CFD

- 신뢰성 있는 결과 도출
- CAD 및 격자 생성 필요

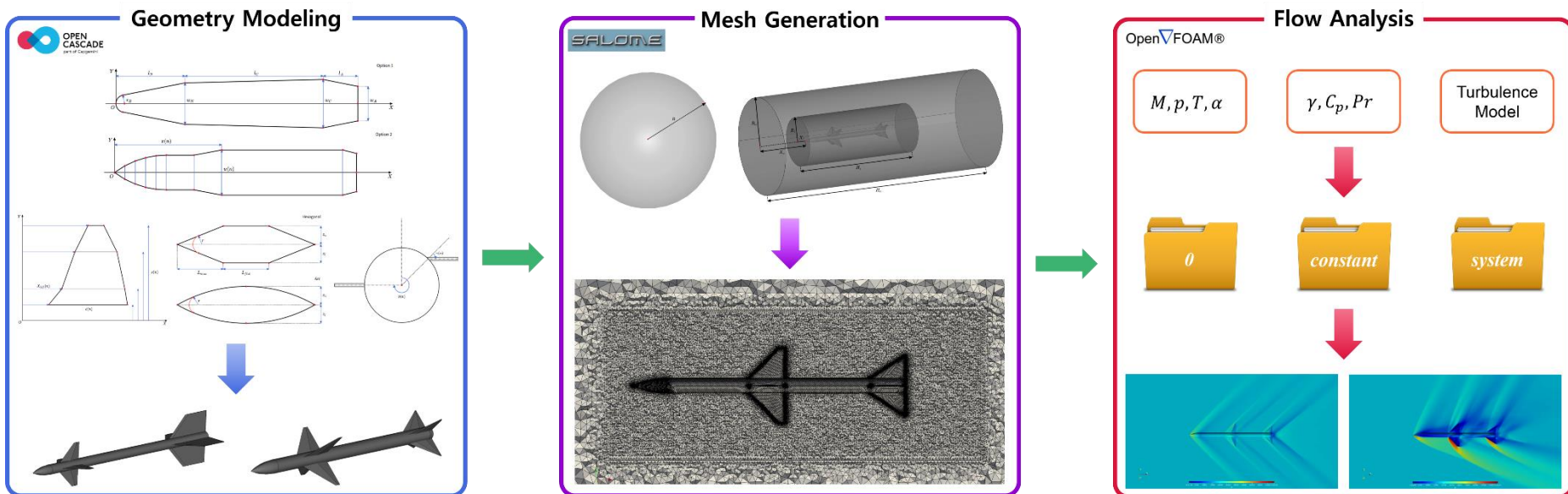


- ❖ 초기 설계 단계에 CFD 사용의 문제점
  - 다양한 형상에 대한 많은 해석 필요
  - CAD 및 격자 생성 작업의 어려움
    - ✓ 일반적으로 사람 손으로 작업 진행
  - 라이선스 비용

# 서론

## ❖ 연구 목적

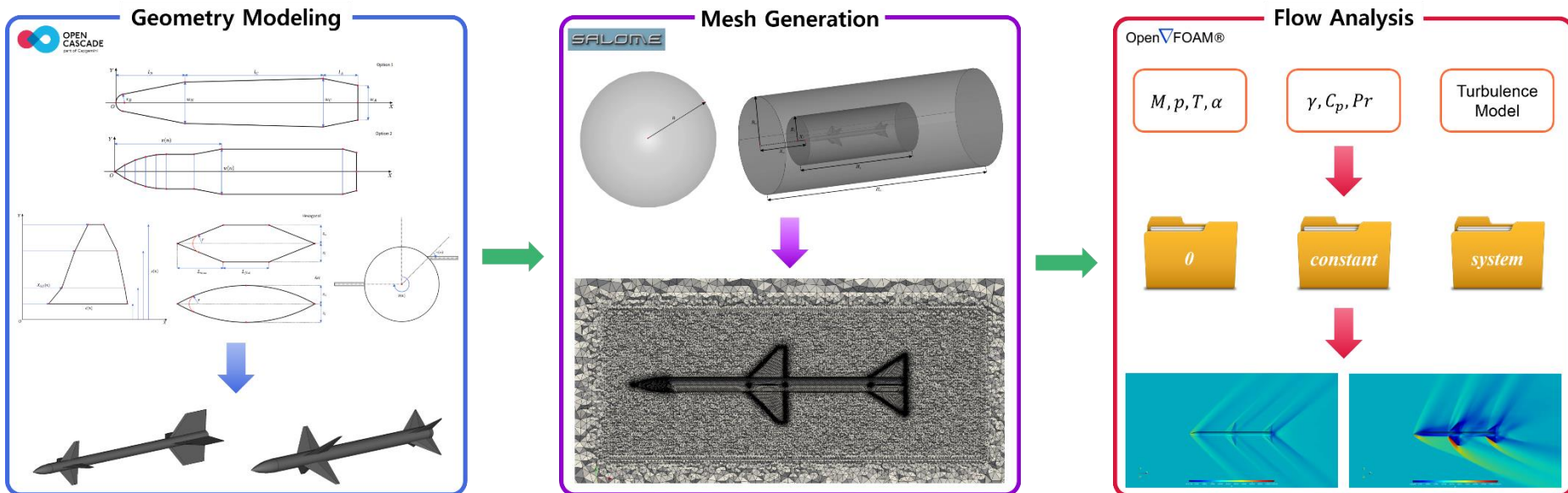
- **오픈소스 기반 미사일 공력해석 자동화 기술 개발**
  - ✓ 형상 생성 모듈
  - ✓ 격자 생성 모듈
  - ✓ 유동 해석 모듈



# 공력 해석 자동화

## ❖ 개요

- 오픈소스 라이브러리를 사용하여 개발
- **XML** 파일 형식의 입력 파라미터 사용
- 입력 파라미터로부터 **사용자 개입 없이** 해석 수행





# 공력 해석 자동화

## ❖ 형상 생성

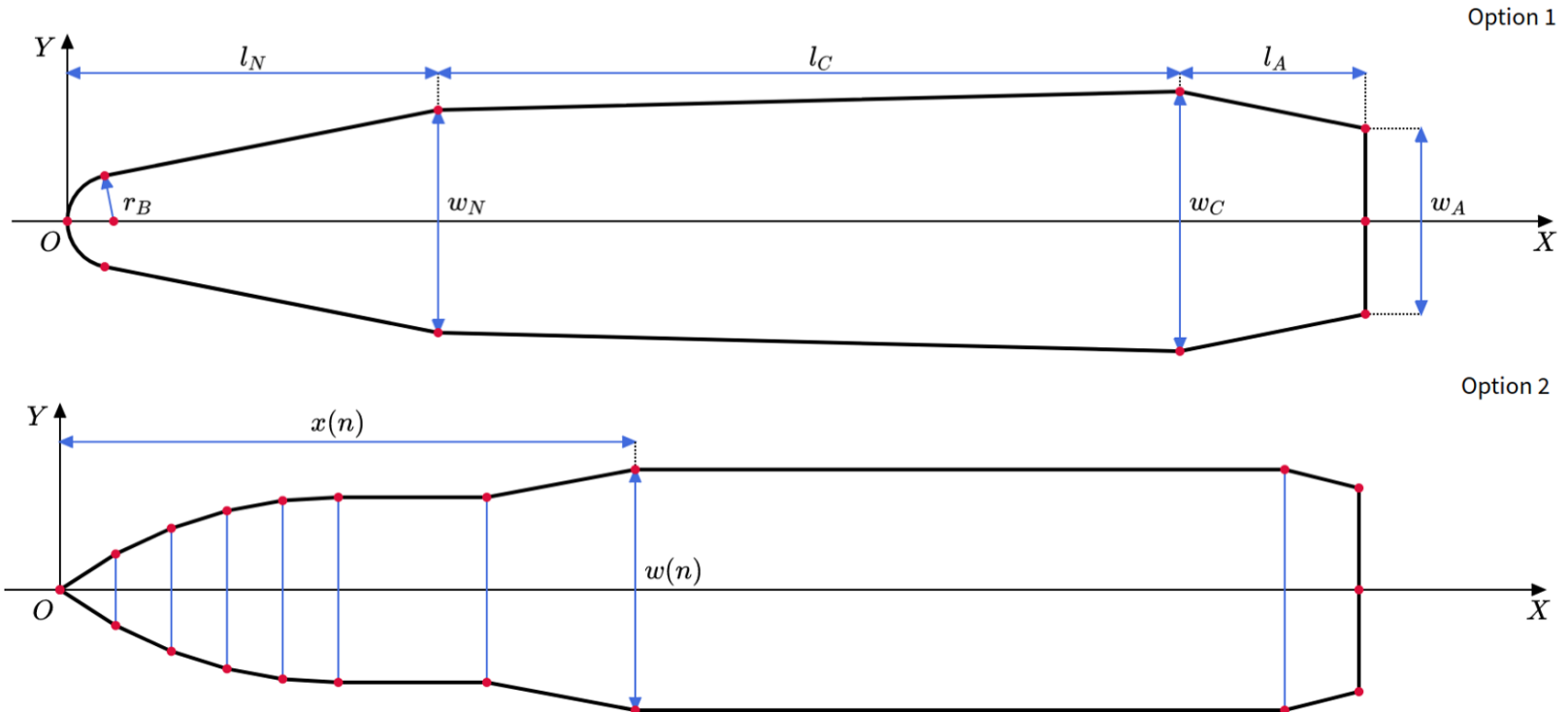
- **pythonOCC** 라이브러리 사용하여 개발
  - ✓ OpenCASCADE의 python 버전
- Missile DATCOM의 형상 정의법 사용
  - ✓ Body + Finset 의 일반적인 미사일 형상 정의 가능
- 격자 생성을 위해 sharp edge를 **blunted edge**로 수정

# 공력 해석 자동화

## ❖ 형상 생성

- Body

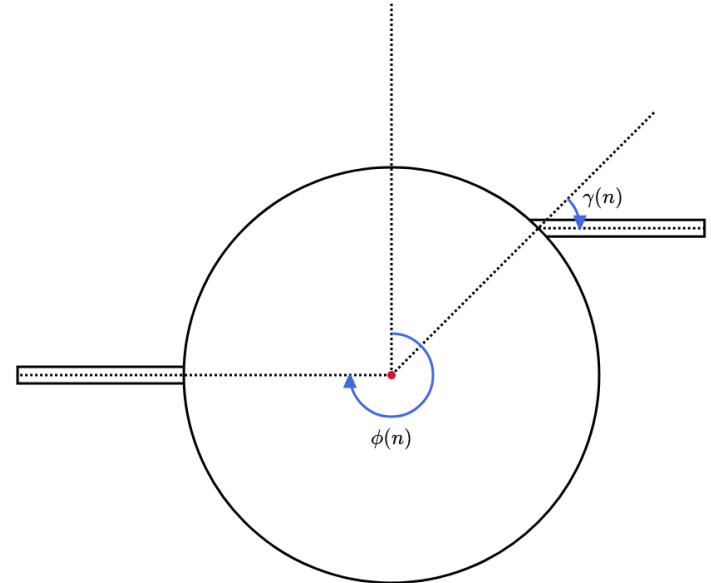
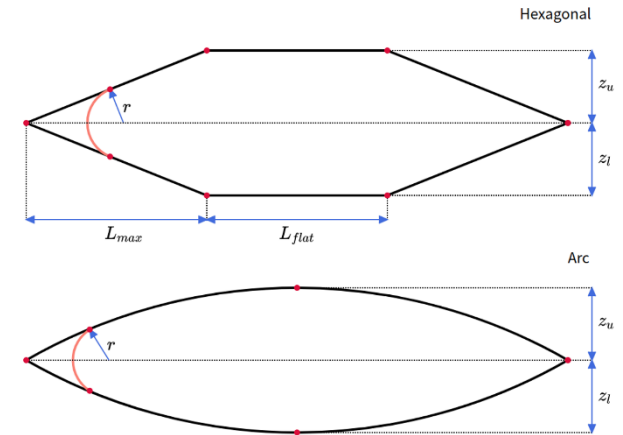
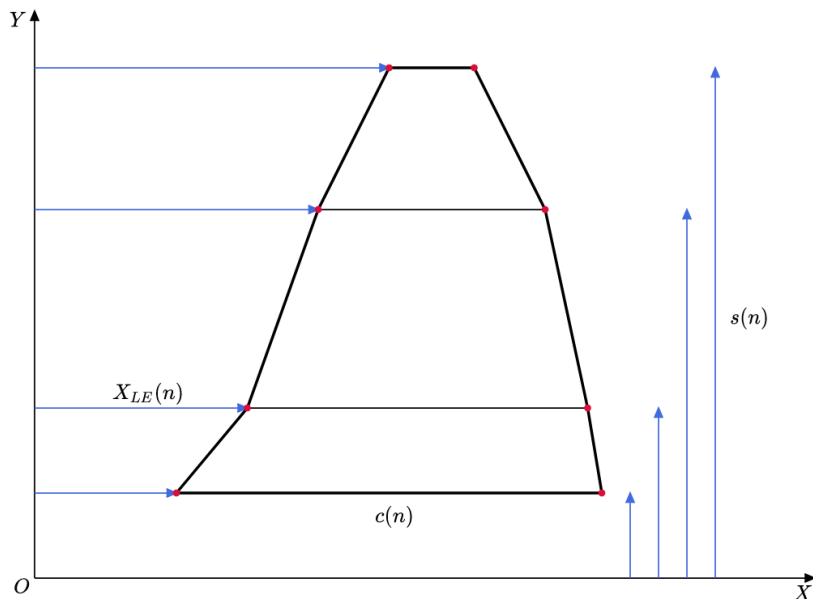
- ✓ Option 1 : Nose, center body, after body의 조합
- ✓ Option 2 :  $(X, R)$  data



# 공력 해석 자동화

## ❖ 형상 생성

- Finset
  - ✓ Planform : chord, semi-span 등
  - ✓ 익형
  - ✓ 부착 방법 : 부착 각도 등

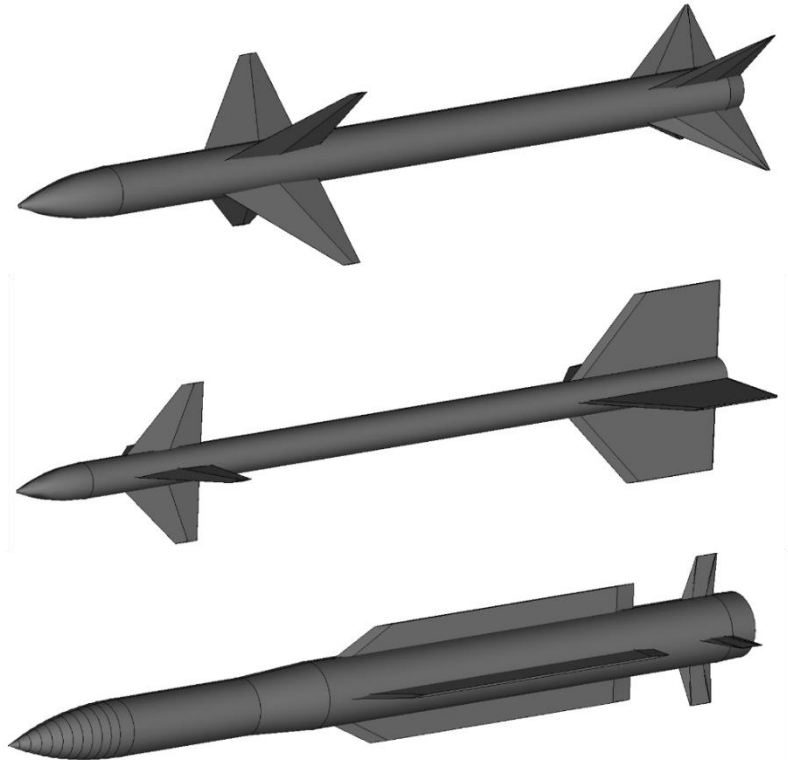


# 공력 해석 자동화

## ❖ 형상 생성

- 다양한 타입의 미사일 형상 생성 가능
- 1~2초의 생성 시간 소요

```
<Item Name="Body" Type="Body1">  
  <Item Name="Nose" Type="NOSE">  
    <X0>0.000000</X0>  
    <TNOSE>OGIVE</TNOSE>  
    <LNOSE>0.068600</LNOSE>  
    <WNOSE>0.030480</WNOSE>  
    <BNOSE>0.001524</BNOSE>  
    <ENOSE>1.000000</ENOSE>  
    <TRUNC>.FALSE.</TRUNC>  
  </Item>  
  <Item Name="Centr" Type="CENTR" ...>  
  <Item Name="Aft" Type="AFT" ...>  
</Item>
```



# 공력 해석 자동화

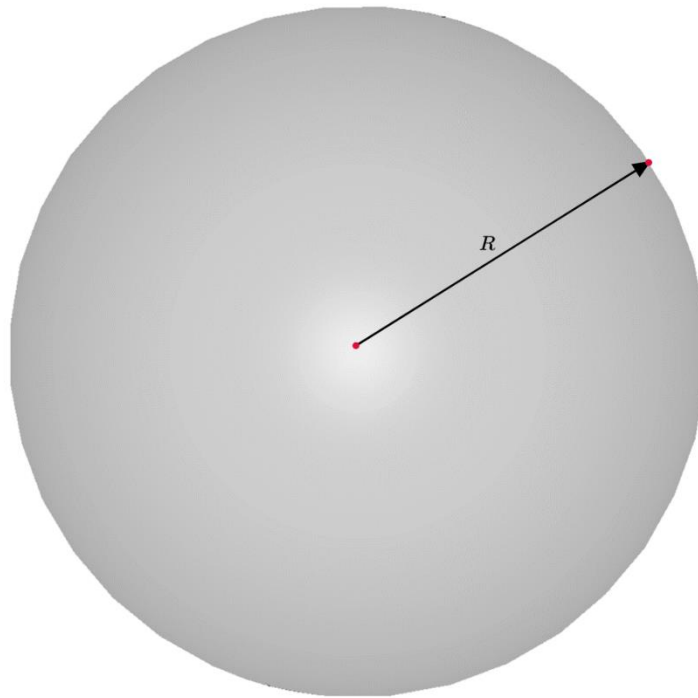
## ❖ 격자 생성

- **SALOME** 프로그램을 사용하여 격자 생성
  - ✓ Open-source platform for numerical simulation
  - ✓ 한계점 : 격자 생성 병렬화 지원 X
- Unstructured tetrahedral mesh
  - ✓ Prism layer
- Ideas UNV 파일 형식으로 격자 저장

# 공력 해석 자동화

## ❖ 격자 생성

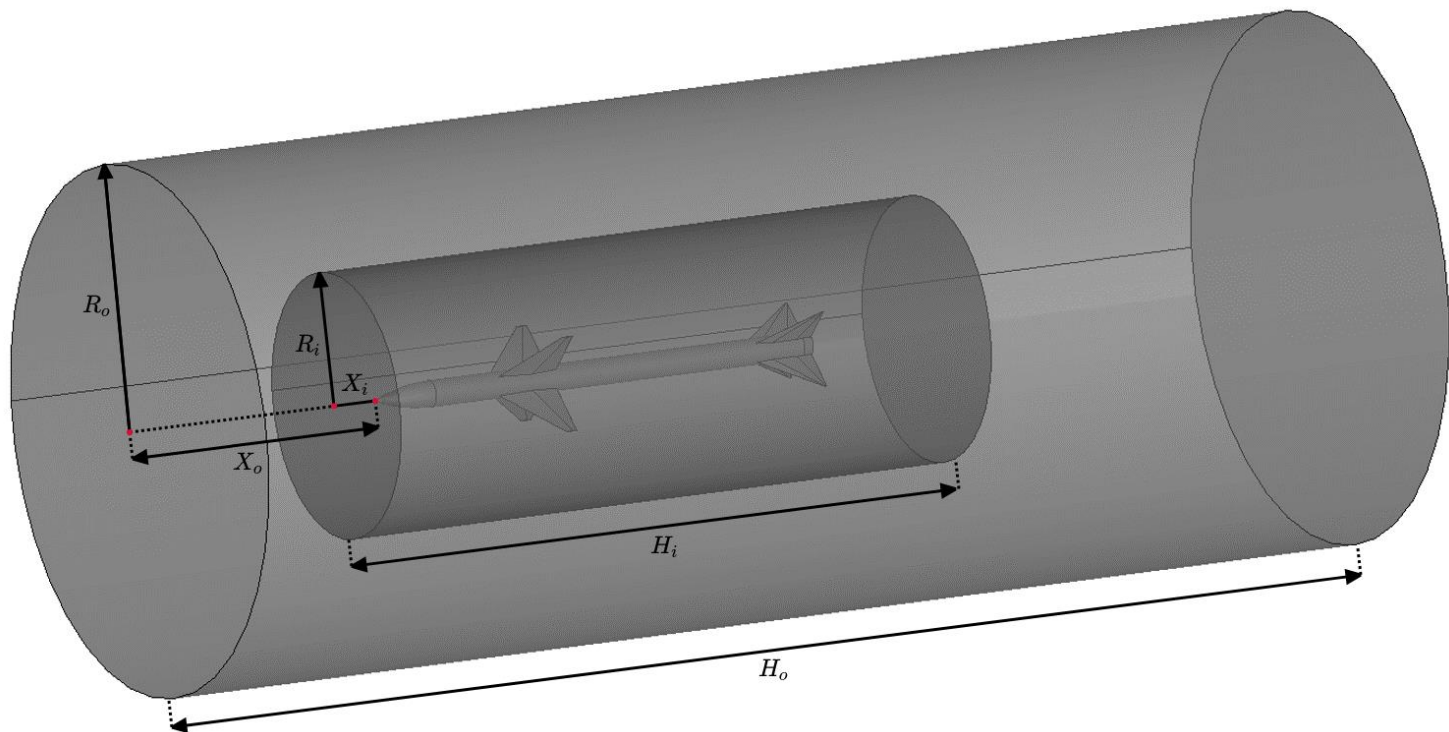
- Domain
  - ✓ 구 형태의 Domain 사용



# 공력 해석 자동화

## ❖ 격자 생성

- Refinement regions
  - ✓ 충격파 및 후류에 대한 정확한 해석
  - ✓ 형상의 변화에 관계없이 격자 quality 유지



# 공력 해석 자동화

## ❖ 격자 생성

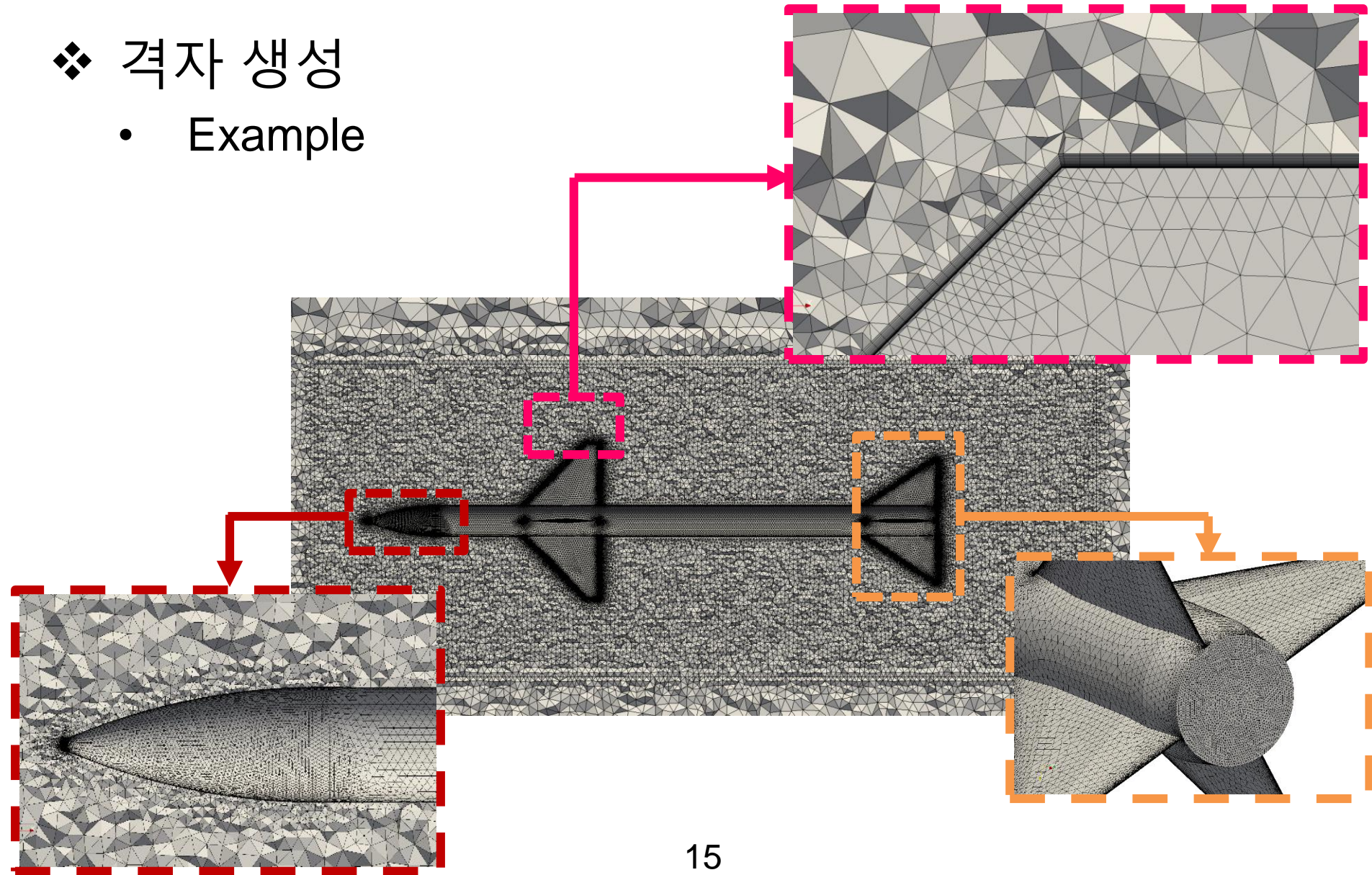
- 격자 크기는 각 region에 대해 입력 파라미터로 정의
  - ✓ Min, max, growth rate
- 특정 surface에 대해서는 자동으로 크기 설정
  - ✓ Body – Nose / Base surface
  - ✓ Finset – Tip surface



# 공력 해석 자동화

## ❖ 격자 생성

- Example



# 공력 해석 자동화

## ❖ 유동 해석

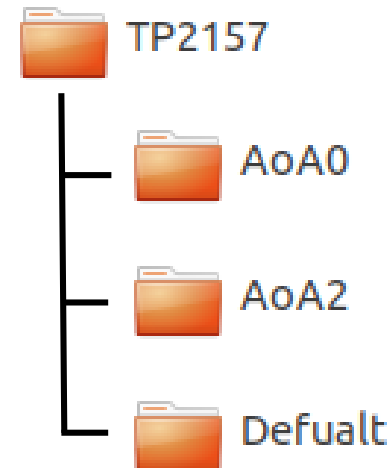
- OpenFOAM을 사용하여 유동 해석 수행
  - ✓ *0, constant, system* 의 디렉토리 구조
- 공력해석자로 **TSLAeroFoam** 사용
  - ✓ Density-based compressible coupled solver
- 난류모델로  **$k - \omega$  SST** model 사용

# 공력 해석 자동화

## ❖ 유동 해석

- 자유류 조건 및 기체 상수를 입력 파라미터로 정의
  - ✓ 자유류 조건 :  $M, P, T, AoA$
  - ✓ 기체 상수 :  $\gamma, C_p, Pr$
- 각 받음각에 대한 케이스 디렉토리 자동 생성

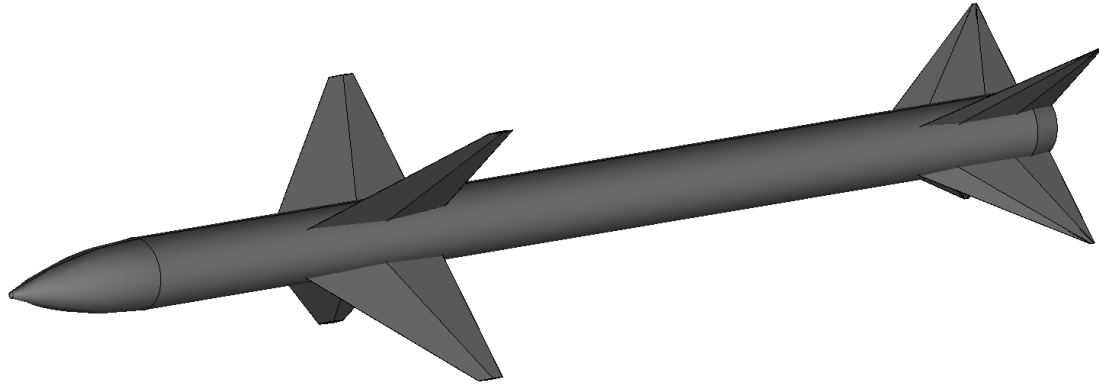
```
<Item Name="FLTCON" Type="FLTCON">  
  <MACH>3.500000</MACH>  
  <PFREE>1696.550000</PFREE>  
  <TFREE>94.202900</TFREE>  
  <ALPHA>  
    <param>0.000000</param>  
    <param>2.000000</param>  
  </ALPHA>  
</Item>
```



# Benchmark Test

## ❖ Case 1 (NASA Sparrow)

- Finset 배치 : X Configuration



- 자유류 조건

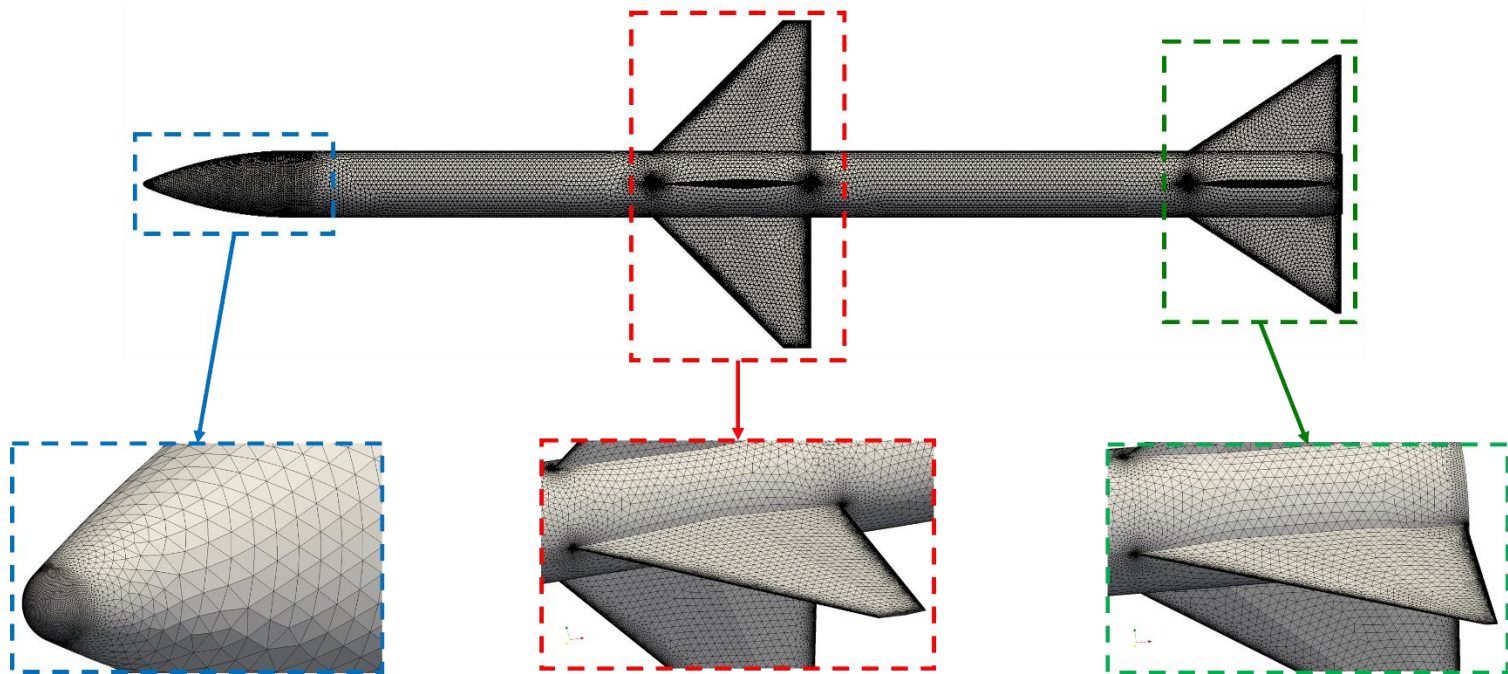
	Value
$M$	1.5
$P(Pa)$	18114.8
$T(K)$	233.793



# Benchmark Test

## ❖ Case 1 (NASA Sparrow)

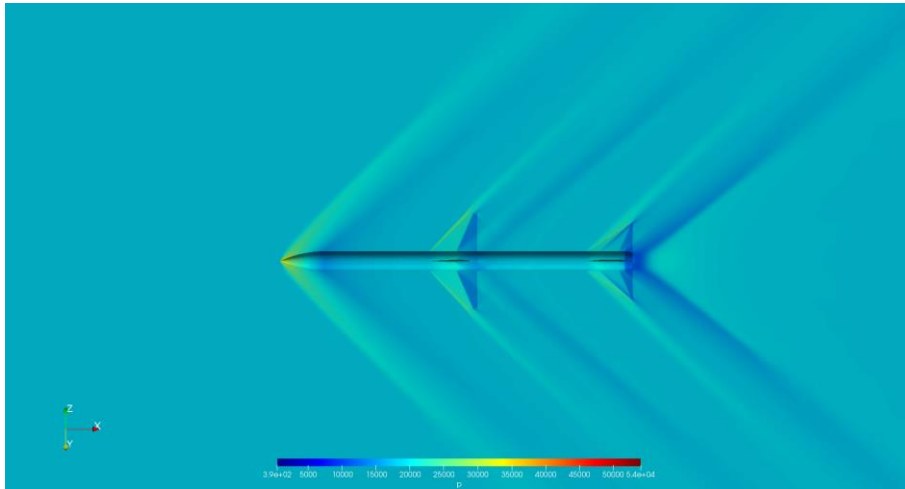
- 격자
  - ✓ Prism layers : 20층 ( $y^+ = 1$ )
  - ✓ Volume mesh : ~2330만



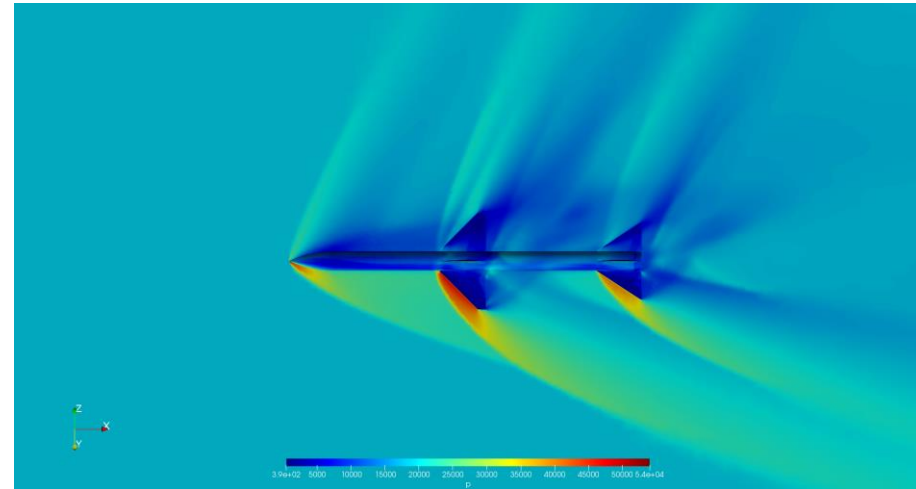
# Benchmark Test

## ❖ Case 1 (NASA Sparrow)

- 결과
  - ✓ 미사일 주위의 충격파가 잘 예측됨



$$\alpha = 0^\circ$$

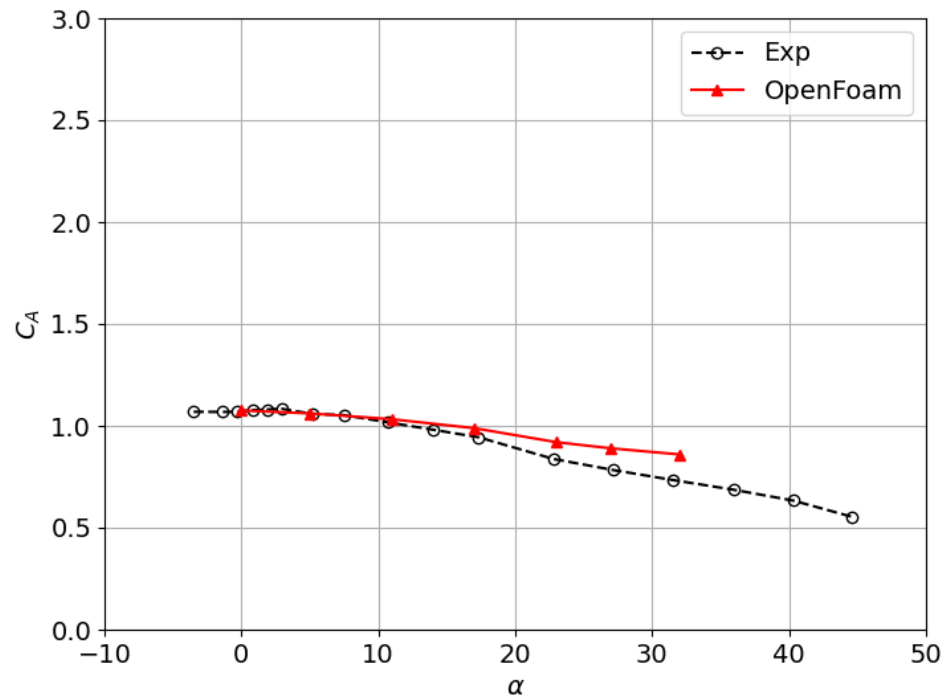


$$\alpha = 32^\circ$$

# Benchmark Test

## ❖ Case 1 (NASA Sparrow)

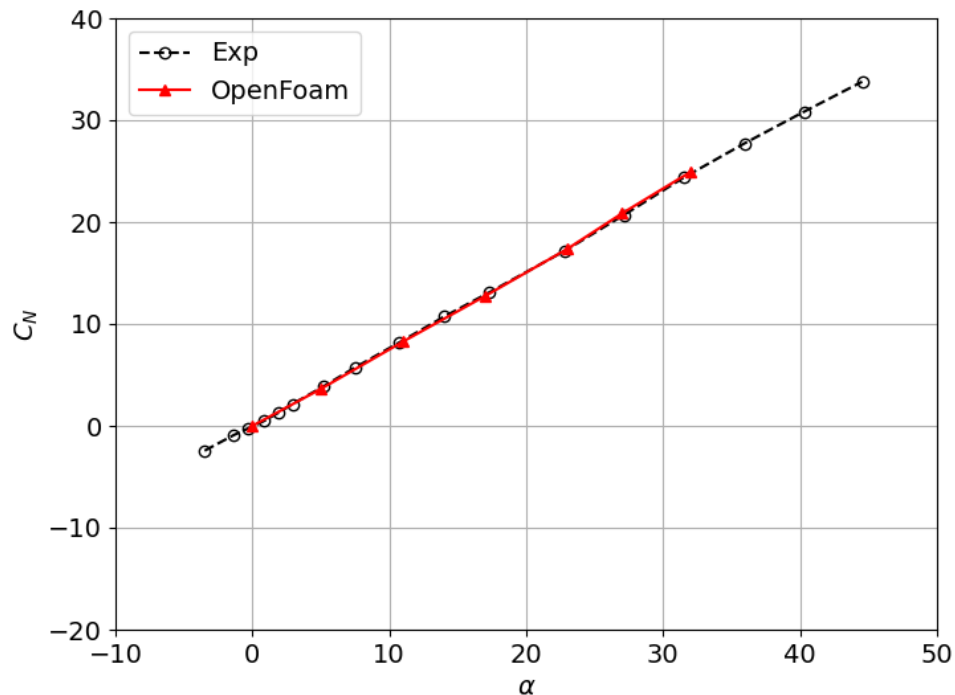
- 결과
  - ✓ 고받음각에서  $C_A$ 가 크게 예측됨



# Benchmark Test

## ❖ Case 1 (NASA Sparrow)

- 결과
  - ✓  $C_N$  은 모든 받음각에서 매우 일치함

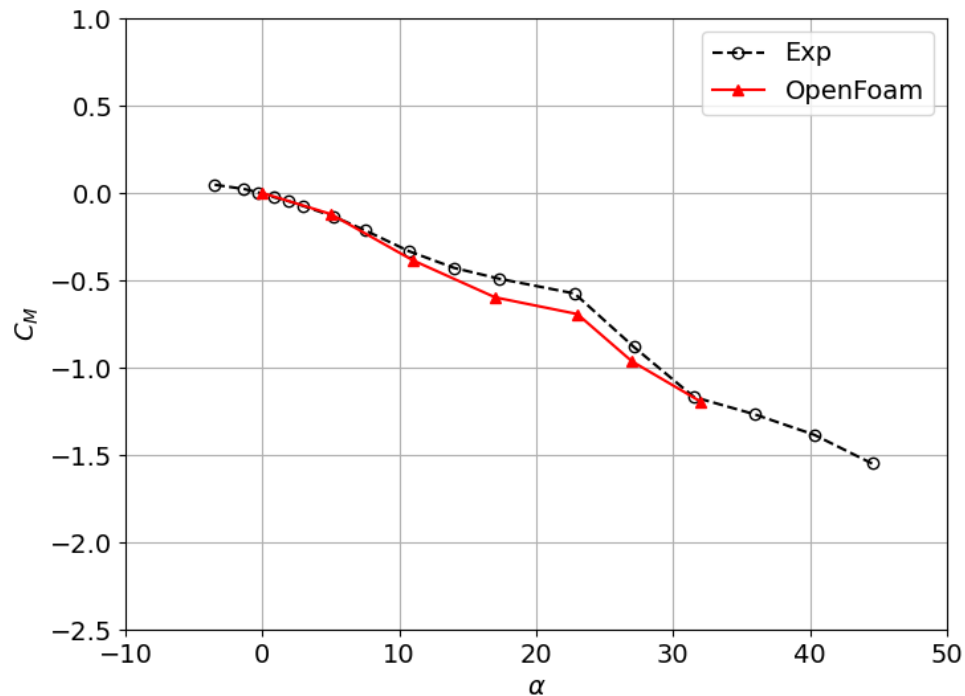




# Benchmark Test

## ❖ Case 1 (NASA Sparrow)

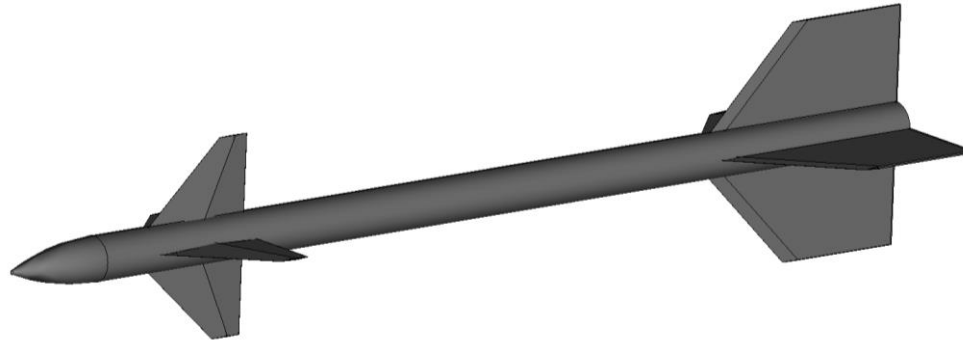
- 결과
  - ✓  $C_M$  은 실험과 경향이 잘 맞음



# Benchmark Test

## ❖ Case 2 (NASA TCM)

- Finset 배치 : + Configuration



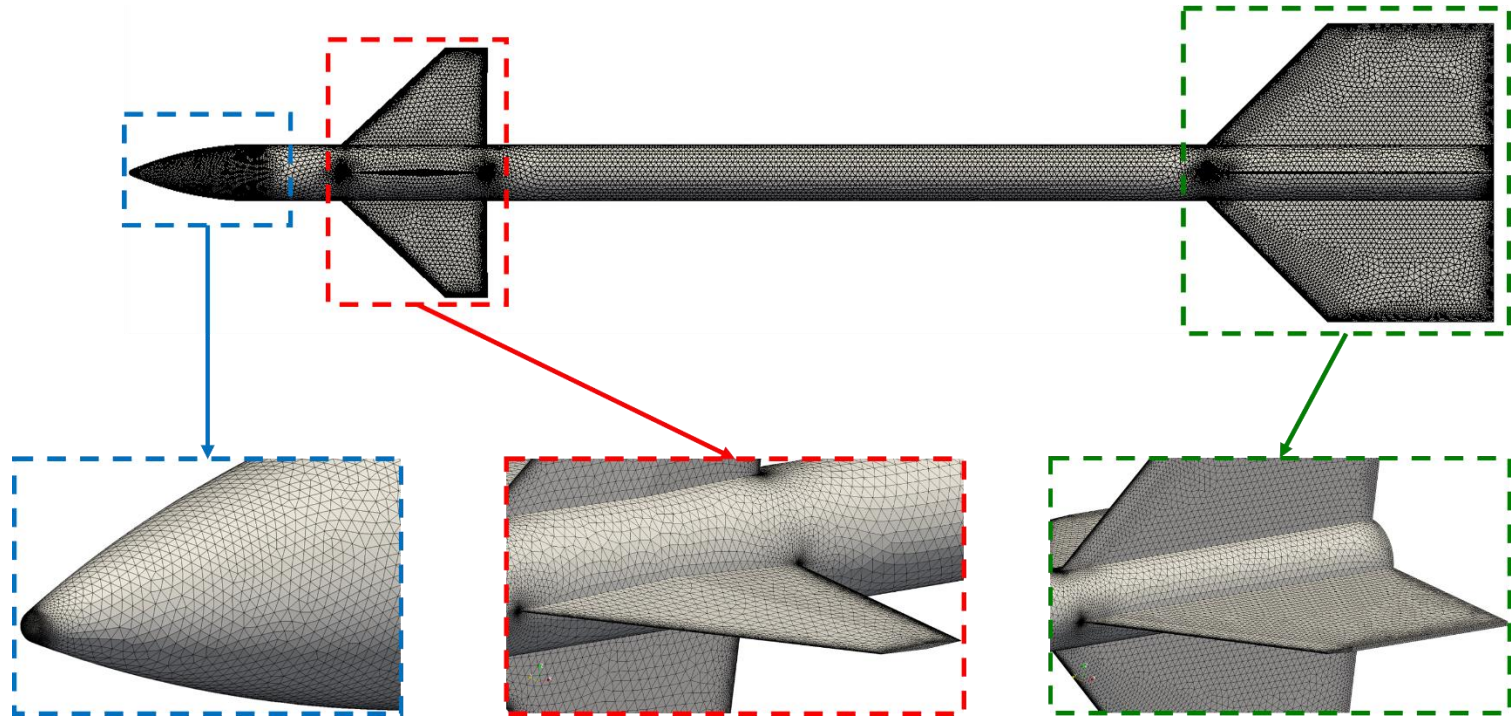
- 자유류 조건

	Value
$M$	3.5
$P(Pa)$	1696.55
$T(K)$	94.2029

# Benchmark Test

## ❖ Case 2 (NASA TCM)

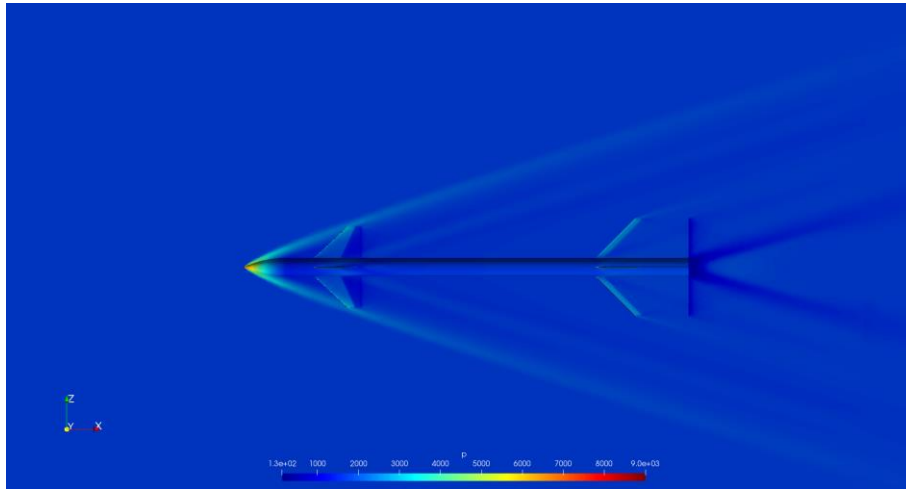
- 격자
  - ✓ Prism layers : 20층 ( $y^+ = 1$ )
  - ✓ Volume mesh : ~2550만



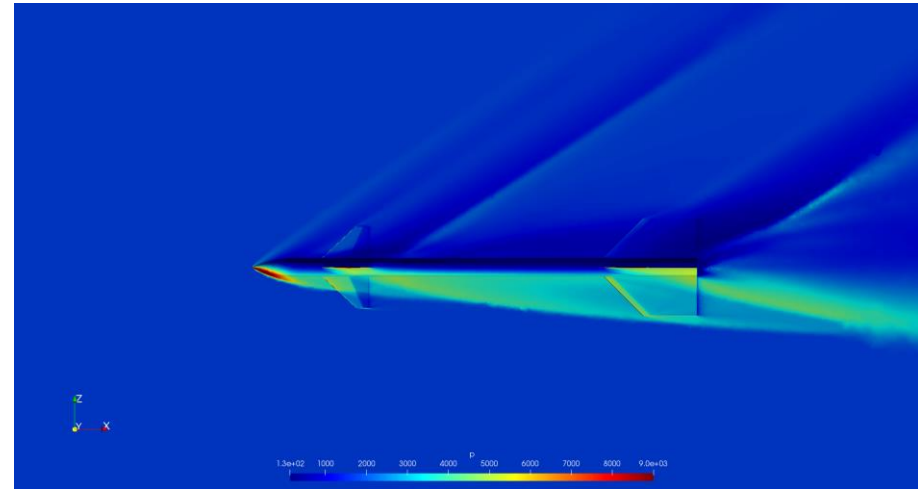
# Benchmark Test

## ❖ Case 2 (NASA TCM)

- 결과
  - ✓ 미사일 주위의 충격파가 잘 예측됨



$\alpha = 0^\circ$

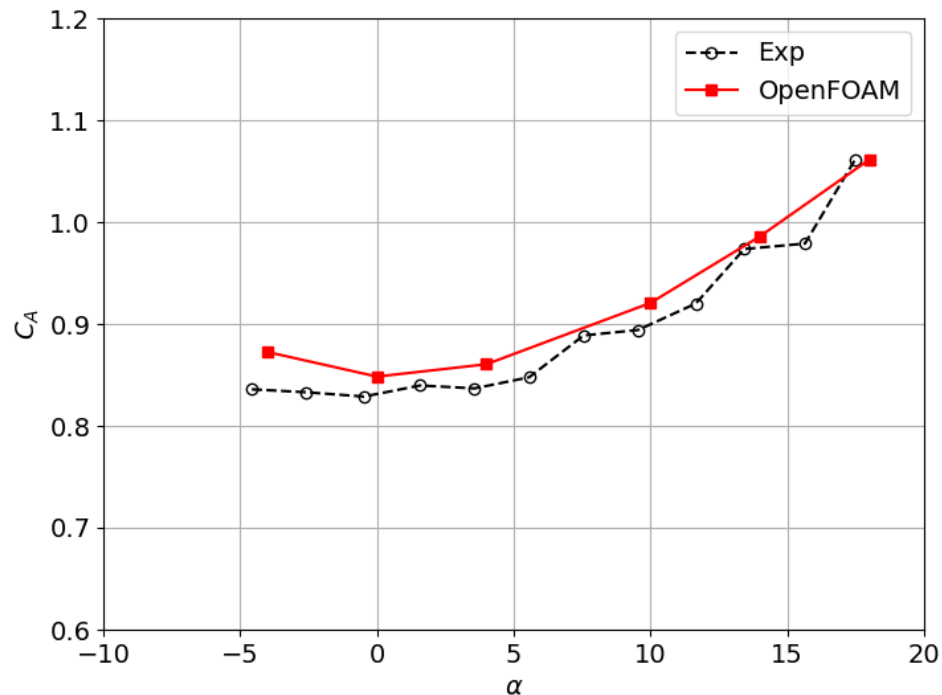


$\alpha = 18^\circ$

# Benchmark Test

## ❖ Case 2 (NASA TCM)

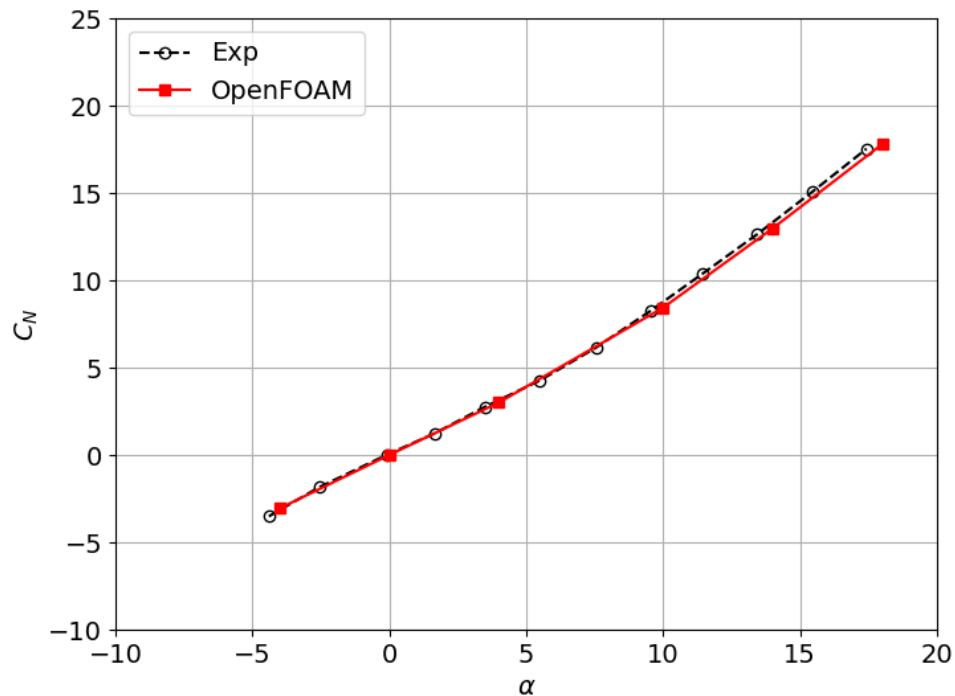
- 결과
  - ✓  $C_A$ 는 다소 과예측하나 경향이 잘 맞음



# Benchmark Test

## ❖ Case 2 (NASA TCM)

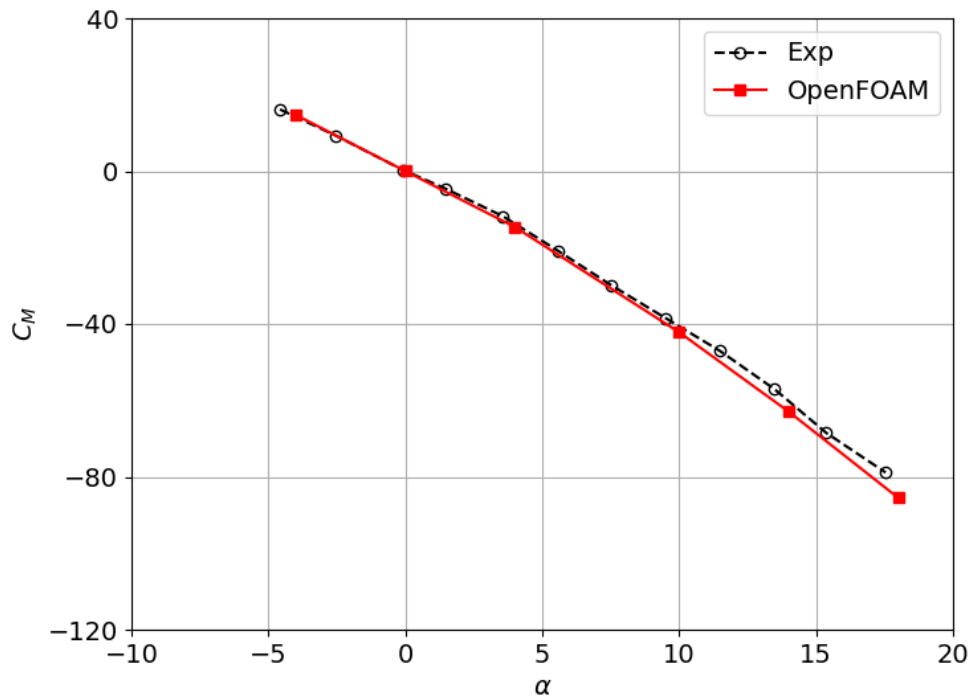
- 결과
  - ✓  $C_N, C_M$  은 모든 받음각에서 매우 일치함



# Benchmark Test

## ❖ Case 2 (NASA TCM)

- 결과
  - ✓  $C_N, C_M$  은 모든 받음각에서 매우 일치함



# 결론 & Future Works

- ❖ 오픈소스 기반 **미사일 공력 해석 자동화 기술 개발**
  - 형상 생성 모듈
  - 격자 생성 모듈
  - 유동 해석 모듈
- ❖ Benchmark Test를 통해 개발된 기술 검증
  - 유동 해석 결과가 실험과 잘 맞음을 확인
- ❖ Finset Control에 대한 자동화 기술 개발 예정



---

**Thank You.**