LENDI INSTITUTE OF ENGINEERING AND TECHNOLOGY

STUDENT MANUAL

COMPUTATIONAL FLUID DYNAMICS

B.Tech -Mechanical Engineering

Department of Mechanical Engineering

COMPUTATIONAL FLUID DYNAMICS

(SKILL ORIENTED COURSE)

STUDENT MANUAL

For

MECHANICAL ENGINEERING

Prepared By

M. GANGADHAR RAO

DEPARTMENT OF MECHANICAL ENGINEERING

LENDI INSTITUTE OF ENGINEERING AND TECHNOLOGY
An Autonomous Institution
(Approved by A.I.C.T.E & Permanently Affiliated to JNTU-GV, Vizianagaram
Accredited by NAAC with "A" Grade & NBA)
Jonnada ,Denkada (Mandal), Vizianagaram Dist – 535 005 Phone
No. 08922-241111, 24166
E-Mail: lendi_2008@yahoo.comWebsite: www.lendi.org

LENDI INSTITUTE OF ENGINEERING AND TECHNOLOGY
An Autonomous Institution
(Approved by A.I.C.T.E & Permanently Affiliated to JNTU- GV, Vizianagaram
Accredited by NAAC with "A" Grade & NBA)
Jonnada, Denkada (Mandal), Vizianagaram Dist – 535 005
Phone No. 08922-241111, 24166
E-Mail: lendi_2008@yahoo.com Website: www.lendi.org

DEPARTMENT OF MECHANICAL ENGINEERING

COMPUTATIONAL FLUID DYNAMICS (SDC) MASTER MANUAL

DEGREE	B. Tech (U.G)
COURSE WITH CODE	Computational Fluid Dynamics **(Skill Oriented Course-5)** R20MEC-SC4101 (C422)
REGULATION	R20
PROGRAM	**MECHANICAL ENGINEERING**
YEAR & SEMESTER	IV B. TECH- I SEM
COURSE AREA/ DOMAIN	THERMAL ENGINEERING
CREDITS	2

LENDI INSTITUTE OF ENGINEERING AND TECHNOLOGY
An Autonomous Institution
(Approved by A.I.C.T.E & Permanently Affiliated to JNTU- GV, Vizianagaram
Accredited by NAAC with "A" Grade & NBA)
Jonnada, Denkada (Mandal), Vizianagaram Dist – 535 005
Phone No. 08922-241111, 24166
E-Mail: lendi_2008@yahoo.com Website: www.lendi.org

INSTITUTE

VISION

- Producing globally competent and quality technocrats with human values for the holistic needs of industry and society.

MISSION

- Creating an outstanding infrastructure and platform for enhancement of skills, knowledge and behavior of students towards employment and higher studies.

- Providing a healthy environment for research, development and entrepreneurship, to meet the expectations of industry and society.

- Transforming the graduates to contribute to the socio-economic development and welfare of the society through value-based education.

LENDI INSTITUTE OF ENGINEERING AND TECHNOLOGY
An Autonomous Institution
(Approved by A.I.C.T.E & Permanently Affiliated to JNTU- GV, Vizianagaram
Accredited by NAAC with "A" Grade & NBA)
Jonnada, Denkada (Mandal), Vizianagaram Dist – 535 005
Phone No. 08922-241111, 24166
E-Mail: lendi_2008@yahoo.com Website: www.lendi.org

DEPARTMENT OF MECHANICAL ENGINEERING

VISION

Envisions Mechanical Engineers of highly competent and skilled professionals to meet the needs of the modern society.

MISSION

- Providing a conductive and inspiring learning environment to become competent engineers.
- Providing additional skills and training to meet the current and future needs of the industry.
- Providing a unique environment towards entrepreneurship by fostering innovations, creativity, freedom and empowerment.

PROGRAM EDUCATIONAL OBJECTIVES (PEOs)

PEO1: Graduates will have strong knowledge and skills and attitudes towards employment, higher studies and research.

PEO2: Graduates shall comprehend latest tools and techniques to analyze, design and develop novel systems and products and solutions for real life problems.

PEO3: Graduates shall have multidisciplinary approach, professional attitude and ethics, good communication and teamwork and engage in life-long learning and professional development to adapt to rapidly changing technology.

PROGRAM SPECIFIC OUTCOMES (PSOs)

PSO1: Capable of design, develop and implement sustainable mechanical environmental systems.

PSO2: Quality in national and international competitive examinations for successful higher studies and employment.

LENDI INSTITUTE OF ENGINEERING AND TECHNOLOGY
An Autonomous Institution
(Approved by A.I.C.T.E & Permanently Affiliated to JNTU- GV, Vizianagaram
Accredited by NAAC with "A" Grade & NBA)
Jonnada, Denkada (Mandal), Vizianagaram Dist – 535 005
Phone No. 08922-241111, 24166
E-Mail: lendi_2008@yahoo.com Website: www.lendi.org

PROGRAM OUTCOMES (POs)

Engineering Graduates will be able to:

PO1: Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

PO2: Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO3: Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

PO4: Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO5: Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

PO6: The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

P07: Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

PO8: Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

P09: Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO10: Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO11: Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO12: Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change

LENDI INSTITUTE OF ENGINEERING AND TECHNOLOGY
An Autonomous Institution
(Approved by A.I.C.T.E & Permanently Affiliated to JNTU- GV, Vizianagaram
Accredited by NAAC with "A" Grade & NBA)
Jonnada, Denkada (Mandal), Vizianagaram Dist – 535 005
Phone No. 08922-241111, 24166
E-Mail: lendi_2008@yahoo.com Website: www.lendi.org

COMPUTATIONAL FLUID DYNAMICS SYLLABUS
(SKILL ORIENTED COURSE)

Course Objective:

The objectives of the course are to

1. Understand the fundamentals and relevance of fluid mechanics in the broader context of engineering sciences in general.
2. Analyze fluid flows through different configurations along with the measurement of flow parameters.
3. Expertise of experimentation, simulation and the fundamental concepts those are required to translate a novel engineering idea to reality through dimensional analysis and similitude.
4. Expose a wide variety of research areas and concerns in and around fluid mechanics such as energy, health etc. across multidisciplinary domains.
5. Explain engineering skills such as solving engineering problems in a professional way, using commercial software packages such as ANSYS Fluent etc. for data analysis and presentation, numerical simulations etc.

Course Outcomes:

After completion of this lab the student will be able to

1. ***Select*** the suitable computational method for specified Fluid flow conditions (Level-3)
2. ***Analyze*** the critical parameters of flow for the specified fluid flow conditions. (Level-4)
3. ***Apply*** appropriate solution strategy for the given incompressible fluid system (Level-3)
4. ***Develop*** CFD problem for the given boundary conditions of fluid flow. (Level-6)
5. ***Design*** computational solutions using software tools to analyze and solve specified fluid flow problems. (Level -6)

Introduction: Illustration of the CFD approach, CFD as an engineering analysis tool, Review of governing equations, Modeling in engineering, Partial differential equations- Parabolic, Hyperbolic and Elliptic equation, CFD application in Mechanical Engineering, CFD software packages and tools.

Principles of Solution of the Governing Equations: Finite difference and Finite volume Methods, Convergence, Consistency, Error and Stability, Accuracy, Boundary conditions, CFD model formulation.

Mesh generation: Overview of mesh generation, Structured and Unstructured mesh, Guideline on mesh quality and design, Mesh refinement and adaptation.

Solution Algorithms: Discretization schemes for pressure, momentum and energy equations - Explicit and implicit Schemes, First order upwind scheme, second order upwind scheme, QUICK scheme, SIMPLE, SIMPLER and MAC algorithm, pressure-velocity coupling algorithms, velocity-stream function approach, solution of Navier-Stokes equations.

CFD Solution Procedure: Problem setup – creation of geometry, mesh generation, selection of physics and fluid properties, initialization, solution control and convergence monitoring, results reports and visualization

LIST OF EXPERIMENT:

Numerical simulation of the following flow problems using commercial software packages:

1. 2D and 3D structured grid generation – flat plates, aerofoil
2. 3D unstructured grid generation – pipe, external aerodynamics
3. Incompressible internal laminar flows
4. Incompressible external laminar flows
5. Incompressible internal turbulent flows
6. Incompressible external turbulent flows
7. Forced Convection flows
8. Fluid Structure Interaction (Flow past a cylinder, Flow over an airfoil.)
9. Heat transfer analysis in heat exchanger
10. Heat transfer analysis in solar flat plate collector
11. Multiphase flows
12. Compressible flows

Note: Any Ten Experiments

INDEX

S.NO	NAME OF THE EXPERIMENT	PAGE NO.	CO	PO
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				
11				

LENDI INSTITUTE OF ENGINEERING AND TECHNOLOGY
An Autonomous Institute
Approved by A.I.C.T.E & Permanently Affiliated to JNTU-GV, Vizianagaram
(Accredited by NAAC with 'A' Grade & NBA)
Jonnada (Village), Denkada (Mandal), Vizianagaram Dist – 535 005
Phone No. 08922-241111, 241112
E-Mail: lendi_2008@yahoo.com Website: www.lendi.org

COURSE DATA SHEET

Academic Year: 2024-25

PROGRAM: Mechanical Engineering	**DEGREE:** UG (B Tech)
COURSE: Computational Fluid Dynamics (Skill Oriented Course)	**YEAR &SEMESTER:** IV & I **CREDITS:** 3
COURSE CODE: R20MEC-SC4101 (C422) **REGULATION:** R20	**COURSE TYPE:** Skill Oriented Course
COURSE AREA/DOMAIN: Soft Computing	**CONTACT HOURS:** 3 Periods/Week.
CORRESPONDING LAB COURSE CODE (IF ANY):----	**LAB COURSE NAME (IF ANY):---**

SYLLABUS

Module	DETAILS	HOURS
I	**Introduction**: Illustration of the CFD approach, CFD as an engineering analysis tool, Review ofgoverning equations, Modeling in engineering, Partial differential equations- Parabolic, Hyperbolic and Elliptic equation, CFD application in Mechanical Engineering, CFD software packages and tools. **Principles of Solution of the Governing Equations:** Finite difference and Finite volume Methods, Convergence, Consistency, Error and Stability, Accuracy, Boundary conditions, CFD model formulation.	**6**
II	**Mesh generation**: Overview of mesh generation, Structured and Unstructured mesh, Guideline on mesh quality and design, Mesh refinement and adaptation. **List of Experiments:** 1. 2D and 3D structured grid generation – flat plates, aerofoil 2. 3D unstructured grid generation – pipe, external aerodynamics	**9**
III	**Solution Algorithms**: Discretization schemes for pressure,	**33**

	momentum and energy equations - Explicit and implicit Schemes, First order upwind scheme, second order upwind scheme, QUICK scheme, SIMPLE, SIMPLER and MAC algorithm, pressure-velocity coupling algorithms, velocity-stream function approach, solution of Navier-Stokes equations. **CFD Solution Procedure:** Problem setup – creation of geometry, mesh generation, selection of physics and fluid properties, initialization, solution control and convergence monitoring, results reports and visualization. **LIST OF EXPERIMENT:** Numerical simulation of the following flow problems using commercial software packages: 1. Incompressible internal laminar flows 2. Incompressible external laminar flows 3. Incompressible internal turbulent flows 4. Incompressible external turbulent flows 5. Forced Convection flows 6. Fluid Structure Interaction (Flow past a cylinder, Flow over an airfoil.) 7. Heat transfer analysis in heat exchanger 8. Heat transfer analysis in solar flat plate collector 9. Multiphase flows 10. Compressible flows	
	TOTAL HOURS	**48**

TEXT/REFERENCE BOOKS:

T/R	BOOK TITLE/AUTHORS/PUBLICATION
T	P.S. Ghosdastidar, Computer Simulation of Flow and Heat Transfer, Tata McGraw-Hill.
T	Muralidhar, K.,and Sundararajan,T. Computational Fluid Flow and Heat Transfer, Narosa Publishing. House .
R	Niyogi, P. Chakrabarty, S.K. and Laha, M.K., Introduction to computational fluid dynamics, Pearson education
R	S K Gupta. Numerical Methods for Engineers, New Age Publishers.
R	Anderson J.D. Computational Fluid Dynamics, Mc-Graw Hills.
R	Numerical Heat Transfer and Fluid Flow: Suhas V. Patankar
R	An introduction of computation fluid dynamics: Versteeg & Malalasekera

COURSE PRE-REQUISITES:

COURSE CODE	COURSE NAME	DESCRIPTION	SEM
R20MEC-PC2202 (C213)	Fluid Mechanics& Hydraulic Machinery	Fluid Mechanics & Hydraulic Machinery explores the behavior of fluids and the design and operation of hydraulic systems, covering principles of fluid flow and machinery like pumps and turbines.	**II-II**
R20MEC-PC3202 (C318)	Heat Transfer	Heat transfer involves the exchange of thermal energy between objects with different temperatures, driven by mechanisms like conduction, convection, and radiation. It plays a crucial role in various engineering applications and natural processes.	**III-II**

COURSE OBJECTIVES:

The objectives of the course are to

1	Understand the fundamentals and relevance of fluid mechanics in the broader context of engineering sciences in general.
2	Analyze fluid flows through different configurations along with the measurement of flow parameters.
3	Expertise of experimentation, simulation and the fundamental concepts those are required to translate a novel engineering idea to reality through dimensional analysis and similitude.
4	Expose a wide variety of research areas and concerns in and around fluid mechanics such as energy, health etc. across multidisciplinary domains.
5	Explain engineering skills such as solving engineering problems in a professional way, using commercial software packages such as ANSYS Fluent etc. for data analysis and presentation, numerical simulations etc.

COURSE OUTCOMES:

At the end of the course students will be able to

SNO	DESCRIPTION	PO (1..12) MAPPING	PSO(1,2) MAPPING
CO.1	***Select*** the suitable computational method for specified Fluid flow conditions **(Apply)**	PO1, PO2, PO3, PO4, PO5, PO6, PO9, PO10, PO12	PSO1, PSO2
CO.2	***Analyze*** the critical parameters of flow for the specified fluid flow conditions. **(Analyze)**	PO1, PO2, PO3, PO4, PO5, PO6, PO7, PO8, PO9, PO10, PO12	PSO1, PSO2
CO.3	***Apply*** appropriate solution strategy for the given incompressible fluid system **(Apply)**	PO1, PO2, PO3, PO4, PO5, PO8, PO9, PO10, PO12	PSO1, PSO2
CO.4	***Develop*** CFD problem for the given boundary conditions of fluid flow. **(Create)**	PO1, PO2, PO3, PO4, PO5, PO6, PO7, PO8, PO9, PO10, PO12	PSO1, PSO2
CO.5	***Design*** computational solutions using software tools to analyze and solve specified fluid flow problems. (**Create)**	PO1, PO2, PO3, PO4, PO5, PO6, PO9, PO10, PO12	PSO1, PSO2
COURSE OVERALL PO/PSO MAPPING: PO1, PO2, PO3, PO4, PO5, PO6, PO7, PO8, PO9, PO10, PO12, PSO1, PSO2			

COURSE OUTCOMES VS POs MAPPING (DETAILED; HIGH:3; MEDIUM:2; LOW:1**):**

SNO	DESCRIPTION	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
1.	***Select*** the suitable computational method for specified Fluid flow conditions **(Apply)**	3	3	3	2	3	2	-	-	2	3	-	2	3	3
2.	***Analyze*** the critical parameters of flow for the specified fluid flow conditions. **(Analyze)**	3	3	3	2	3	2	2	2	2	3	-	2	3	3
3.	***Apply*** appropriate solution strategy for the given incompressible fluid system **(Apply)**	3	3	3	2	3	-	-	2	2	3	-	2	3	3
4.	***Develop*** CFD problem for the given boundary conditions of fluid flow. **(Create)**	3	3	3	3	3	2	2	2	2	3	-	2	3	3
5.	***Design*** computational solutions using software tools to analyze and solve specified fluid flow problems. (**Create)**	3	3	3	3	3	2	-	-	2	3	-	2	3	3
C422*		**3**	**3**	**3**	**3**	**3**	2	2	2	**2**	**3**		**2**	**3**	3

POs & PSO REFERENCE:

PO1	Engineering Knowledge	PO7	Environment & Sustainability	PSO1	Capable of design, develop and implement sustainable mechanical and environmental systems.
PO2	Problem Analysis	PO8	Ethics	PSO2	Qualify in national and international competitive examinations for successful higher studies and employment.
PO3	Design & Development	PO9	Individual & Team Work		
PO4	Investigations	PO10	Communication Skills		
PO5	Modern Tools	PO11	Project Mgt. & Finance		
PO6	Engineer & Society	PO12	Life Long Learning		

COs VS POs MAPPING JUSTIFICATION:

S.NO	PO/PSO MAPPED	LEVEL OF MAPPING	JUSTIFICATION
CO1	PO1	3	Understanding fluid dynamics and computational methods requires fundamental engineering principles in mathematics, physics, and numerical analysis.
	PO2	3	Selecting appropriate computational techniques for different fluid flow conditions involves problem identification, formulation, and critical evaluation of numerical solutions.
	PO3	3	Computational Fluid Dynamics (CFD) methods aid in designing solutions for complex fluid flow problems, ensuring accuracy and efficiency in engineering applications.
	PO4	2	The process of selecting computational methods involves analyzing complex fluid flow behavior, turbulence modeling, and numerical stability, requiring in-depth investigation.
	PO5	3	CFD relies on modern computational tools, software, and simulations, making proficiency in numerical methods and software applications essential.
	PO6	2	CFD is widely applied in industries such as aerospace, automotive, and environmental engineering, impacting society by optimizing designs for safety, efficiency, and sustainability.
	PO9	2	CFD-based projects require collaboration among engineers, researchers, and industry professionals, fostering teamwork and interdisciplinary learning.
	PO10	3	Effective interpretation and presentation of CFD results

			require strong technical communication skills, including report writing, graphical representation, and oral presentations.
	PO12	2	Given the continuous advancements in computational techniques and CFD tools, engineers must engage in lifelong learning to stay updated with evolving methodologies.
	PSO1	3	CFD applications in energy-efficient HVAC systems, aerodynamics, and fluid transport contribute to sustainability by reducing energy consumption and improving performance.
	PSO2	3	Mastery of CFD techniques enhances problem-solving skills, analytical thinking, and technical expertise, which are essential for excelling in competitive exams like GATE, GRE, and industry certifications.
CO2	PO1	3	Understanding flow parameters requires applying fundamental engineering principles in fluid mechanics, thermodynamics, and numerical analysis.
	PO2	3	Identifying and analyzing critical flow parameters involves systematic problem formulation and evaluation of fluid behavior under different conditions.
	PO3	3	Accurate analysis of flow parameters contributes to designing efficient fluid flow systems and optimizing engineering solutions.
	PO4	2	Fluid flow analysis requires investigating turbulence, pressure distribution, and heat transfer, which are essential for solving complex engineering problems.
	PO5	3	CFD tools and simulation software are essential for analyzing flow parameters, requiring expertise in computational techniques.
	PO6	2	Understanding flow behavior helps in designing safe and efficient systems in industries like aerospace, automotive, and environmental engineering.
	PO7	2	Analyzing flow parameters contributes to designing energy-efficient and eco-friendly fluid systems, promoting sustainability.
	PO8	2	Ethical considerations are essential in CFD analysis to ensure accuracy, reliability, and responsible engineering practices.
	PO9	2	Flow analysis in CFD projects requires collaboration among engineers, researchers, and industry professionals.
	PO10	3	Presenting and interpreting flow analysis results effectively requires strong communication skills, including technical reporting and data visualization.
	PO12	2	Advancements in CFD tools and fluid dynamics require continuous learning to stay updated with emerging technologies and methodologies.
	PSO1	3	Flow analysis is crucial for optimizing mechanical and environmental systems for sustainability and efficiency.
	PSO2	3	Expertise in analyzing flow parameters enhances problem-solving skills, improving success in competitive exams and

			career opportunities in CFD-related industries.
CO3	PO1	3	Requires applying fundamental principles of fluid mechanics and numerical methods to solve incompressible fluid flow problems.
	PO2	3	Involves identifying and formulating appropriate solution strategies for different incompressible fluid systems.
	PO3	3	Helps in developing efficient and optimized computational models for fluid flow applications.
	PO4	2	Requires analyzing flow characteristics, boundary conditions, and stability to develop accurate solution strategies.
	PO5	3	Utilizes CFD software and computational techniques to solve and simulate incompressible fluid systems.
	PO8	2	Ensures ethical considerations in CFD modeling, including accuracy, reliability, and responsible engineering practices
	PO9	2	Encourages collaboration and teamwork in solving real-world fluid flow problems using CFD tools.
	PO10	3	Requires effective presentation and interpretation of CFD results through reports, graphs, and simulations.
	PO12	2	Continuous learning is essential to stay updated with evolving CFD techniques and computational methodologies.
	PSO1	3	Applying appropriate solution strategies enhances the development of efficient and sustainable fluid flow systems.
	PSO2	3	Strengthens analytical and problem-solving skills, improving success in competitive exams and career opportunities in CFD-related industries.
CO4	PO1	3	Requires applying fundamental principles of fluid dynamics, thermodynamics, and numerical analysis to define a CFD problem.
	PO2	3	Involves analyzing the given boundary conditions to correctly formulate the problem for computational modeling.
	PO3	3	Helps in designing an appropriate CFD model that accurately represents the physical fluid flow system.
	PO4	3	Requires evaluating boundary conditions, selecting numerical methods, and ensuring stability and convergence of the solution.
	PO5	3	Utilizes CFD software and computational techniques to develop and solve fluid flow problems.
	PO6	2	CFD solutions contribute to real-world applications in industries like aerospace, automotive, and energy sectors.
	PO7	2	Develops CFD models that aid in designing energy-efficient and environmentally friendly fluid systems.
	PO8	2	Ensures ethical considerations in defining problem constraints, model validation, and result interpretation.
	PO9	2	Encourages teamwork and collaboration in defining, modeling, and validating CFD problems.
	PO10	3	Requires effective documentation, visualization, and

			presentation of the CFD problem and simulation results.
	PO12	2	Continuous learning is essential to stay updated with advanced CFD problem-solving techniques and industry practices.
	PSO1	3	Developing CFD problems contributes to optimizing sustainable and energy-efficient fluid flow systems.
	PSO2	3	Strengthens analytical and computational skills, improving success in research, higher studies, and CFD-related careers.
CO5	PO1	3	Requires applying fundamental fluid mechanics and computational principles to solve fluid flow problems using software tools.
	PO2	3	Involves analyzing fluid flow problems and selecting appropriate computational methods for accurate solutions.
	PO3	3	Helps in developing numerical models and simulations for optimized fluid flow system designs.
	PO4	2	Requires in-depth investigation of flow behavior, turbulence modeling, and boundary conditions for accurate CFD solutions.
	PO5	3	Utilizes advanced CFD software and computational tools for fluid flow analysis and simulation.
	PO6	2	Supports real-world engineering applications in industries like aerospace, automotive, and environmental sectors, ensuring safety and efficiency.
	PO9	2	Encourages collaborative problem-solving and teamwork in CFD projects.
	PO10	3	Requires effective communication of simulation results through technical reports, presentations, and graphical data representation.
	PO12	2	Encourages continuous learning to stay updated with emerging CFD tools and evolving computational techniques.
	PSO1	3	Enables the use of software tools to design and optimize sustainable and efficient fluid flow systems.
	PSO2	3	Enhances technical expertise and problem-solving skills, improving success in competitive exams and career opportunities in CFD-related industries.

TOPICS BEYOND SYLLABUS/ADVANCED TOPICS/DESIGN:

S.NO	TOPIC	PROPOSED ACTIONS	RELEVANCE TO POS, PSOS
1	Natural Convection Flows	Taught the concept and addressed a related Computational Fluid Dynamics (CFD) problem as an additional experiment.	PO1, PO2, PO3, PO4, PO5, PO6, PO7, PO8, PO9, PO10, PO12, PSO1, PSO2

TOPICS BEYOND SYLLABUS/ADVANCED TOPICS WEB SOURCE REFERENCES:

1	https://archive.nptel.ac.in/courses/112/108/112108246/
2	https://www.sfu.ca/~mbahrami/ENSC%20388/Notes/Natural%20Convection.pdf
3	https://www.youtube.com/watch?v=wxKkHI7GHbs
4	https://www.youtube.com/watch?v=Vha2Ak7-jdk
5	https://www.thermopedia.com/de/content/786/

WEB SOURCE REFERENCES:

1	https://www.cfdsupport.com/references.html
2	https://www.cfd-online.com/Links/
3	https://www.youtube.com/playlist?list=PL30F4C5ABCE62CB61
4	https://help.sim-flow.com/tutorials
5	https://cfd.ninja/ansys-tutorials/ansys-fluent-tutorials/

DELIVERY/INSTRUCTIONAL METHODOLOGIES/PEDAGOGICAL INITIATIVES:

☑CHALK & TALK	☑ICT TOOLS	☑WEB REFERENCES	☑STUDENT SEMINARS
☐ INDUSTRIAL VISITS	☐ INTERNSHIPS	☑LABORATORY LEARNING	☐ MODEL-BASED LEARNING
☐ GUEST LECTURES	☑COLLABORATIVE LEARNING	☐ MINI/MAJOR PROJECTS	☑CASE STUDIES/REAL LIFE EXAMPLES

ASSESSMENT METHODOLOGIES-DIRECT

☐ ASSIGNMENTS	☐ STUD. SEMINARS	☑TESTS/MODEL EXAMS	☑ END SEM EXAMINATION
☑ STUD. LAB PRACTICES	☑ STUD. VIVA	☐ MINI/MAJOR PROJECTS	☐ CERTIFICATIONS
☐ ADD-ON COURSES	☐ OTHERS		

ASSESSMENT METHODOLOGIES-INDIRECT

☑ COURSE EXIT SURVEY

Prepared by
M Gangadhar Rao

Approved By

1. Department IQAC co-ordinator **2. Head Of the Department**

INDEX SHEET

NAME OF THE LAB:

S.NO	EXERCISE NO	DATE	NAME OF THE EXERCISE	PAGE NO	GRADE MARKS	SIGN & DATE OF THE FACULTY
1						
2						
3						
4						
5						
6						
7						
8						
9						
10						
11						
12						

SIGNATURE OF THE FACULTY

INTRODUCTION

INTRODUCTION TO COMPUTATIONAL FLUID DYNAMICS (CFD)

Computational Fluid Dynamics (CFD) is a branch of fluid mechanics that uses numerical analysis and computational power to simulate fluid flow, heat transfer, and related phenomena. It has become a vital tool for engineers and researchers across multiple disciplines, including aerospace, automotive, chemical, and biomedical engineering.

CFD allows engineers to analyze complex fluid flow problems without physical experiments, making it a cost-effective and efficient alternative to traditional testing. By solving the governing equations of fluid motion numerically, CFD provides detailed insights into velocity, pressure, turbulence, and thermal properties of the flow.

CFD AS AN ENGINEERING ANALYSIS TOOL

CFD plays a significant role in design optimization, troubleshooting, and performance evaluation of engineering systems. Some advantages of CFD as an engineering tool include:

1. **Cost Reduction**
   - Reduces the need for expensive prototypes and wind tunnel testing.
   - Minimizes material waste in experimental setups.
2. **Time Efficiency**
   - Faster than physical experiments.
   - Allows quick testing of multiple design variations.
3. **Detailed Insights**
   - Provides a complete field solution (velocity, pressure, temperature) at every point in the domain.
   - Offers visualization of flow characteristics through contour plots, vector fields, and streamlines.

4. **Safety & Risk Reduction**
    - Allows simulation of hazardous conditions, such as explosions, high-speed aerodynamics, and chemical reactions, without endangering human lives.
5. **Performance Optimization**
    - Used to improve efficiency and reduce losses in aerospace, turbomachinery, automotive aerodynamics, and HVAC systems.

REVIEW OF GOVERNING EQUATIONS IN CFD

The behaviour of fluids and heat transfer in CFD is governed by fundamental conservation laws formulated as partial differential equations (PDEs). These include:

1. **Mass Conservation (Continuity Equation)**

    This equation ensures mass is conserved in the system.

$$\partial\rho/\partial t + \nabla \cdot (\rho V) = 0$$

2. **Momentum Equations (Navier-Stokes Equations)**
    Derived from Newton's Second Law, these equations describe the motion of fluids.

$$\rho\ (\partial V/\partial t + V \cdot \nabla V) = -\nabla P + \nabla \cdot \tau + \rho g$$

3. **Energy Equation (Heat Transfer Analysis)**

    Used to study thermal effects in fluids.

$$\rho cp\ (\partial T/\partial t + V \cdot \nabla T) = \nabla \cdot (k\ \nabla T) + \Phi$$

MODELLING IN ENGINEERING

Modelling in engineering is a crucial process that involves developing mathematical and computational representations of physical systems. This approach helps engineers analyze, optimize, and predict the performance of real-world applications in a virtual environment before conducting experimental tests. Computational Fluid Dynamics (CFD) uses modelling techniques to simulate fluid flow, heat transfer, and related physical phenomena, providing engineers with detailed insights into system behavior.

CFD modelling consists of several key steps, each of which contributes to accurately representing the physical system under investigation:

1. Geometry Creation

The first step in CFD modelling is defining the physical domain using Computer-Aided Design (CAD) software. The geometry represents the system where fluid flows, such as an airfoil, heat exchanger, pipeline, or combustion chamber.

- The accuracy of the geometry plays a crucial role in the reliability of CFD results.
- Complex geometries may require simplification while maintaining the essential features of the model.
- CAD tools such as SolidWorks, CATIA, and AutoCAD are commonly used to create the required geometries.
- The defined geometry is then imported into a CFD pre-processing tool for further modifications, such as surface cleaning and defeaturing to remove unnecessary details.

2. Meshing

Once the geometry is created, it is divided into a finite number of smaller regions called control volumes, elements, or cells. This process, known as meshing, is essential for solving the governing equations of fluid flow numerically.

- The quality of the mesh directly affects the accuracy and computational efficiency of the CFD simulation.
- There are different types of meshes, including structured, unstructured, hybrid, and adaptive meshes.
- A fine mesh is required in regions with complex flow characteristics (such as boundary layers and vortices), while a coarser mesh can be used in areas with relatively uniform flow.
- Meshing tools such as ANSYS Mesher, ICEM CFD, and Open Foam's meshing utilities are used to generate high-quality meshes.

3. Boundary Conditions

After meshing, boundary conditions must be assigned to define how the fluid interacts with the system. Boundary conditions specify the flow properties at the domain boundaries, ensuring that the simulation accurately represents real-world conditions.

- **Inlet Conditions:** Define how the fluid enters the domain, including velocity, pressure, temperature, and turbulence parameters.
- **Outlet Conditions:** Specify how the fluid exits the domain, typically using pressure or mass flow boundary conditions.

- **Wall Conditions:** Define surfaces where no fluid passes through, such as solid walls with no-slip or slip conditions. Heat transfer properties can also be applied to walls.
- **Symmetry and Periodic Conditions:** Used to reduce computational effort by exploiting geometric or flow symmetry.
- **Moving and Rotating Boundaries:** Essential for modelling rotating machinery such as turbines and fans.

4. Solver Selection

Once the pre-processing steps (geometry creation, meshing, and boundary condition setup) are completed, the next step is selecting a numerical solver to compute the fluid flow and thermal properties. The solver is responsible for discretizing and solving the governing equations of fluid mechanics.

- Different solvers are used for different types of flows (laminar, turbulent, compressible, incompressible, multiphase, etc.).
- The **Finite Volume Method (FVM)** is the most common approach in CFD, converting partial differential equations into algebraic equations solved iteratively.
- Common turbulence models include **k-ε**, **k-ω**, and **Large Eddy Simulation (LES)**, which approximate turbulent effects in fluid flows.
- The choice of solver and numerical scheme significantly impacts the stability, accuracy, and convergence of the simulation.

5. Post-Processing

Once the solver has completed the calculations, the results must be analyzed and visualized to extract meaningful insights. This step, known as post-processing, involves interpreting simulation outputs to evaluate system performance.

- **Contour Plots:** Used to visualize scalar quantities such as pressure, velocity, and temperature distributions.
- **Streamlines and Vector Fields:** Display fluid flow patterns and velocity direction across the domain.
- **Cut Planes and Isosurfaces:** Help in analyzing 3D flow behavior by slicing through the simulation domain.
- **Flow Animation:** Enables visualization of time-dependent simulations to track fluid movement.

- **Quantitative Analysis:** Includes lift and drag coefficients, heat transfer rates, pressure drop, and turbulence intensity to assess design performance.
- Tools such as **ANSYS Fluent, ParaView, Tecplot, and MATLAB** are widely used for CFD post-processing.

PARTIAL DIFFERENTIAL EQUATIONS (PDES) IN CFD

PDEs describe the relationship between multiple variables in fluid flow and heat transfer. Based on their nature, PDEs are classified as:

1. Parabolic Equations

Parabolic equations represent diffusion-dominated problems where the solution evolves over time. These equations are commonly used in heat conduction and transient fluid flow problems.

Example: Heat Conduction Equation

$$\partial T/\partial t = \alpha \nabla^2 T$$

Where:

- T is the temperature

- α is the thermal diffusivity

- $\nabla^2 T$ represents the Laplacian operator indicating spatial diffusion

2. Hyperbolic Equations

Hyperbolic equations describe wave propagation and convection-dominated flows. These equations govern the behavior of compressible fluids and shock waves.

Example: Euler Equations (Inviscid Flow)

$$\partial u/\partial t + a\, \partial u/\partial x = 0$$

Where:

- u represents the velocity

- a is the wave speed

- The equation describes how disturbances propagate through the medium

3. Elliptic Equations

Elliptic equations are used for steady-state problems, where time dependency is absent. They describe equilibrium conditions in fluid flow and heat conduction.

Example: Laplace's Equation

$$\nabla^2\varphi = 0$$

Where:

- φ represents a scalar field such as velocity potential or temperature distribution

- The equation is used in electrostatics, incompressible fluid flow, and steady-state heat transfer

CFD APPLICATIONS IN MECHANICAL ENGINEERING

1. Aerospace Engineering
   - Drag reduction in aircraft wings.
   - Shockwave analysis in supersonic flight.
2. Automobile Industry
   - Aerodynamics and fuel efficiency optimization.
3. HVAC and Refrigeration
   - Air conditioning system design.
4. Combustion Analysis
   - Internal combustion engine simulations.
5. Biomedical Engineering
   - Blood flow simulation in arteries.

CFD SOFTWARE PACKAGES AND TOOLS

Various CFD software tools are used for simulations, including:

1. ANSYS Fluent
    - Industry-leading software for fluid dynamics and heat transfer simulations.
2. OpenFOAM
    - Open-source CFD package with extensive customization capabilities.
3. COMSOL Multiphysics
    - Used for coupled physics simulations (fluid-structure interactions).
4. CFX (ANSYS CFX)
    - Specializes in turbomachinery and rotating machinery applications.
5. Star-CCM+
    - Used in automotive and aerospace CFD analysis.
6. SimScale
    - A cloud-based CFD simulation platform.

PRINCIPLES OF SOLUTION OF THE GOVERNING EQUATIONS IN CFD

Finite Difference and Finite Volume Methods

Finite Difference Method (FDM)

- The Finite Difference Method (FDM) is a numerical technique for solving differential equations by approximating derivatives using difference equations.
- The domain is discretized into a structured grid of points, and derivatives are replaced by finite difference approximations.
- Common finite difference schemes:
    - **Forward Difference:** $f'(x) \approx f(x+\Delta x) - f(x) / \Delta x$
        - Accuracy: First-order
    - **Backward Difference:** $f'(x) \approx f(x) - f(x-\Delta x) / \Delta x$
        - Accuracy: First-order
    - **Central Difference:** $f'(x) \approx f(x+\Delta x) - f(x-\Delta x) / 2\Delta x$

  - Accuracy: Second-order (more accurate than forward and backward difference schemes)

- Higher-order schemes (e.g., fourth-order, sixth-order) are used for increased accuracy.
- FDM is primarily used in structured grid problems where nodes are arranged in a regular manner.

Finite Volume Method (FVM)

- The Finite Volume Method (FVM) is based on the integral form of conservation laws.
- The computational domain is divided into small control volumes (CVs), with governing equations integrated over each control volume.
- Fluxes across the control volume faces are evaluated using interpolation schemes.
- **Steps in FVM:**

1. Divide the domain into control volumes.
2. Integrate the governing equation over a control volume.
3. Apply Gauss's divergence theorem to convert volume integrals into surface integrals.
4. Approximate the surface integrals using numerical schemes.
5. Solve the algebraic equations iteratively.

- FVM is widely used in CFD due to its conservation properties and ability to handle unstructured grids.

CONVERGENCE, CONSISTENCY, AND STABILITY

Convergence

- A numerical method is said to be convergent if the approximate solution approaches the exact solution as the grid is refined.
- **Indicators of convergence:**
  - Residuals decrease to a predefined threshold.
  - The solution remains unchanged with further refinement.
  - The numerical error diminishes with decreasing grid size.
- Convergence can be accelerated using techniques such as under-relaxation, multigrid solvers, and improved initial guesses.

Consistency

- Consistency ensures that the discretized equations reduce to the original differential equations as the grid spacing approaches zero.
- It is evaluated using Taylor series expansion:
  - A method is consistent if the truncation error vanishes as $\Delta x \to 0$.
  - For example, in FDM, the truncation error for a central difference scheme is $O(\Delta x2)$.

Stability

- Stability ensures that errors introduced at any stage do not grow unbounded during computations.
- **Von Neumann Stability Analysis:**
  - Based on Fourier analysis of error propagation.
  - A numerical scheme is stable if its amplification factor satisfies: $|G| \leq 1|$
  - If $|G|>1$, errors amplify, leading to divergence.
- **Courant-Friedrichs-Lewy (CFL) Condition:**
  - For explicit schemes, stability requires: $CFL=u\Delta t/\Delta x \leq Cmax$

    where Cmax depends on the numerical scheme.
  - Ensures that numerical disturbances do not propagate faster than the physical wave speed.

ERROR AND ACCURACY

Error

- **Truncation Error:** The difference between the discretized and exact equations.
- **Round-off Error:** Due to finite precision in floating-point arithmetic.
- **Discretization Error:** Caused by grid resolution and numerical approximation.
- **Iterative Error:** Arises from incomplete convergence in iterative solvers.

Accuracy

- The order of accuracy determines how the numerical error decreases with grid refinement.
- A method is said to be of order $O(\Delta x^n)$ if the error decreases proportionally to Δx^n .
- **Higher-order schemes offer better accuracy but may require finer grids.**

BOUNDARY CONDITIONS

- Boundary conditions are essential for well-posed CFD problems.
- **Types:**
  - **Dirichlet Boundary Condition:** Specifies the variable value at the boundary (e.g., temperature at a wall).
  - **Neumann Boundary Condition:** Specifies the derivative of the variable (e.g., heat flux at a surface).
  - **Mixed Boundary Condition:** A combination of Dirichlet and Neumann conditions.
  - **Periodic Boundary Condition:** Used when the solution repeats over the domain.
  - **Symmetry Boundary Condition:** Used when a problem is symmetric along an axis.
  - **Inlet/Outlet Boundary Condition:** Defines flow parameters at inlets and outlets.

CFD MODEL FORMULATION

- **Step 1: Define Governing Equations**
  - **Continuity Equation:** Conservation of mass
  - **Momentum Equations:** Navier-Stokes equations
  - **Energy Equation:** Conservation of energy
- **Step 2: Discretization**
  - Choose FDM, FVM, or FEM.
  - Convert differential equations into algebraic equations.
- **Step 3: Define Boundary and Initial Conditions**
  - Apply appropriate boundary conditions.
- **Step 4: Solve the Discretized Equations**
  - Use iterative solvers like:
    - Gauss-Seidel Method
    - Multigrid Solvers
    - SIMPLE Algorithm (for pressure-velocity coupling)
- **Step 5: Post-processing**
  - Visualize results using contour plots, streamlines, and vector plots.
  - Validate the numerical results with analytical or experimental data.

Experiment No: - 1

2D STRUCTURED MESH OF PIPE

To perform a 2D structured Mesh for the pipe of 30 cm diameter and 200 cm long. Find the Mesh quality factors after meshing.

AIM

SOFTWARE REQUIREMENT

THEORY

Mesh generation is the practice of creating a mesh, a subdivision of a continuous geometric space into discrete geometric and topological cells. Often these cells form a simplicial complex. Usually the cells partition the geometric input domain. Mesh cells are used as discrete local approximations of the larger domain. Meshes are created by computer algorithms, often with human guidance through a GUI , depending on the complexity of the domain and the type of mesh desired. A typical goal is to create a mesh that accurately captures the input domain geometry, with high-quality (well-shaped) cells, and without so many cells as to make subsequent calculations intractable. The mesh should also be fine (have small elements) in areas that are important for the subsequent calculations.

It is common practice to classify meshes into two main types: structured and unstructured.

1. **Structured meshes**

Structured meshes are meshes with implicit connectivity whose structure allows for easy identification of elements and nodes. Often structured meshes have orthogonal quadrilateral (2D) or hexahedral (3D) elements.

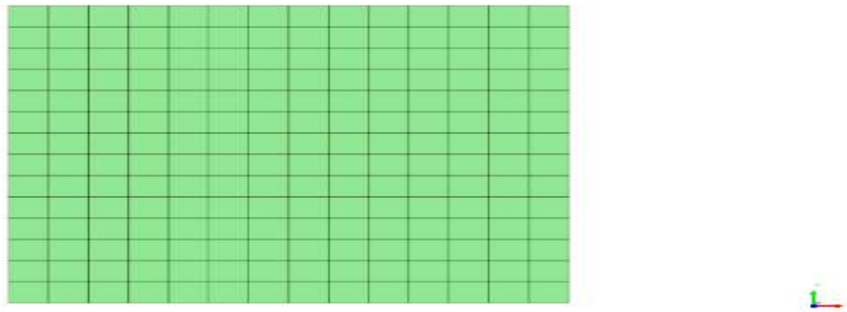


Figure: Structured Mesh of Plane

Structured mesh advantages

- Memory efficient
- Fast to Solve

Structured mesh disadvantages

- Angled and curved geometries are approximated (leads to stair stepping)
- Solutions are not accurate

1. Unstructured meshes

Unstructured meshes are meshes with general connectivity (GCON) whose structure is arbitrary and therefore the connectivity of elements must be defined and stored. GCON element types are non-orthogonal, such as triangles (2D) and tetrahedra (3D).

Unstructured meshes require programmers to map more data to each node, such as adjacency lists and coordinate lists.

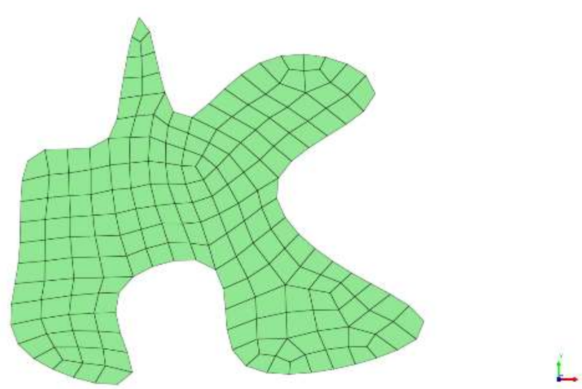


Figure: Unstructured Mesh of Plane

Unstructured mesh advantages

- Complex geometries easier to mesh
- Arbitrary positions

Unstructured mesh disadvantages

- Greater memory requirement
- slower to solve

Mesh Metrics

When it comes to mesh elements, symmetry and uniformity are of prime importance. Symmetrical elements with uniform sides are considered high quality elements.

The "quality" of a mesh cell can be quantified in several ways. ANSYS performs several geometrical checks on mesh elements in order to determine their quality. These checks, or metrics are:

- Element Quality
- Aspect Ratio Calculation for Triangles
- Aspect Ratio Calculation for Quadrilaterals
- Jacobian Ratio
- Warping Factor
- Parallel Deviation
- Maximum Corner Angle
- Skewness
- Orthogonal Quality

i) Element Quality

The Element Quality is based on the ratio of the volume to the sum of the square of the edge lengths for 2D quad/tri elements, or the square root of the cube of the sum of the square of the edge lengths for 3D elements.

- The Element Quality is a composite quality metric that ranges between 0 and 1.
- A value of 1 indicates a perfect cube or square while a value of 0 indicates that the element has a zero or negative volume.

ii) Aspect Ratio Calculation for Triangles

The aspect ratio of a triangle provides a comparison of the "height" and "width" of a triangle.
It varies from 1 to infinite. A value of 1.0 indicate an equilateral triangle.
The image below shows triangles with aspect ratios of 1 and 2.

iii) Aspect Ratio Calculation for Quadrilaterals

The aspect ratio of quadrilaterals provides a comparison of a long side to a short side of the quadrilaterals.It varies from 1 to infinite. A value of 1.0 indicates a square.
The image below shows quadrilaterals with aspect ratios of 1 and 20.

Quadrilaterals with Aspect Ratios

iv) Jacobian Ratio

The Jacobian ratio is a measurement of the shape of a given element compared to that of an ideal element. The ideal shape of an element depends on element type. The ideal Jacobian ratio is 1.0 and a good quality mesh has a Jacobian ratio between 1 and 10 for the majority of its elements (90% and above) [2].

v) Warping Factor

Warping factor is computed and tested for some quadrilateral shell elements, and the quadrilateral faces of bricks, wedges, and pyramids.
The ideal warping factor is 0. Warping is a measure of twisting and distortion and is best understood by looking at quadrilaterals with varying warping factors.

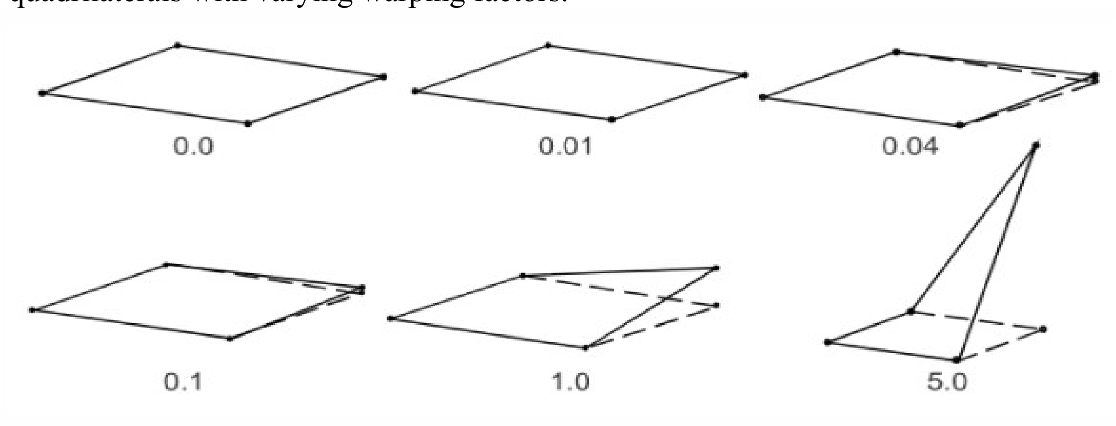


Figure: Quadrilateral Shells with various Warping Factors

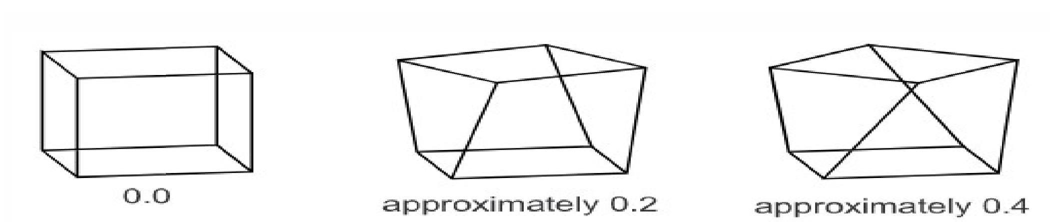


Figure: Bricks with various Warping Factors

vi) Parallel Deviation

Parallel deviation is a measure of how much two parallel sides of a shape deviate. The ideal parallel deviation is 0 for a square.

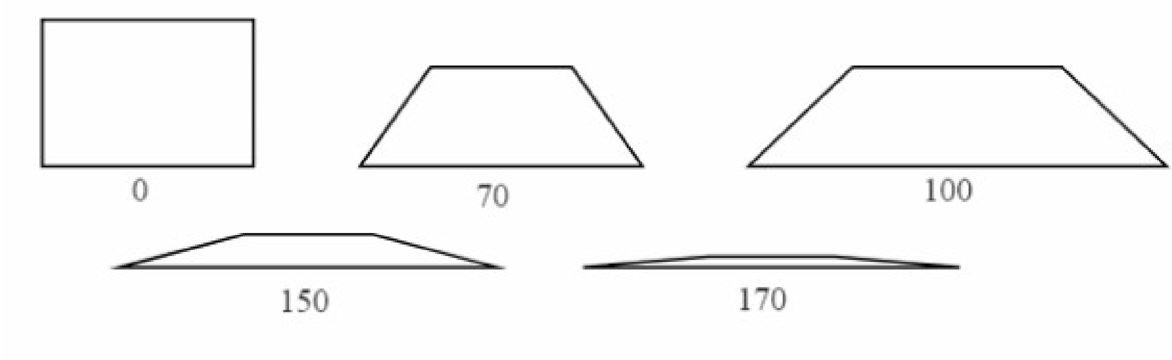


Figure: Parallel Deviations for Quadrilaterals

vii) Maximum Corner Angle

This is the maximum angle between adjacent edges of an element. For a triangle the best maximum angle is 60 degrees. For a quadrilateral it is 90 degrees.

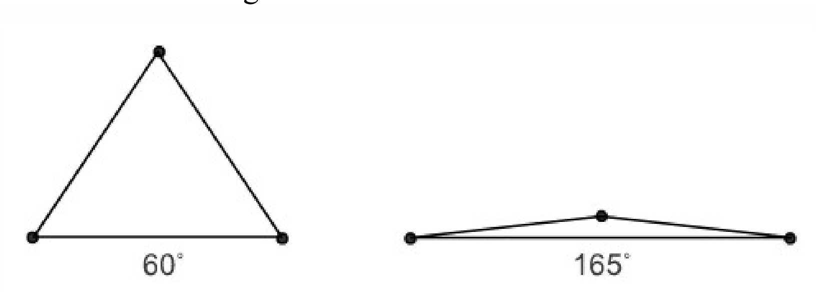


Figure: Maximum Corner Angle for Triangles

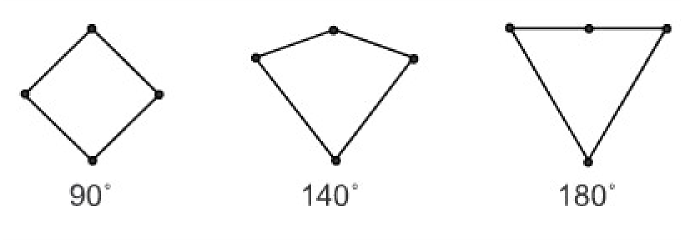


Figure: Maximum Corner Angles for Quadrilaterals

viii) Skewness

Skewness is one of the primary quality measures for a mesh. Skewness determines how close to ideal (equilateral or equiangular) a face or cell is.

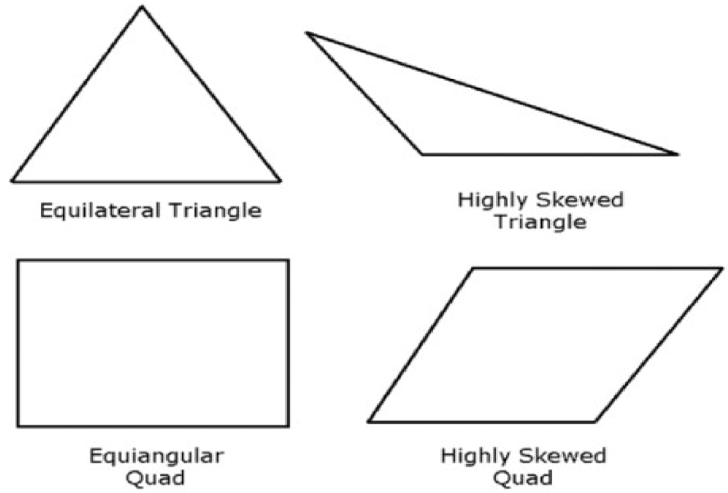


Figure: Triangle and Quadrilateral Skewness

Value of Skewness	Cell Quality
1	degenerate
0.9 — <1	bad (sliver)
0.75 — 0.9	poor
0.5 — 0.75	fair
0.25 — 0.5	good
>0 — 0.25	excellent
0	equilateral

Table: ANSYS recommended scale for skewness

ix) Orthogonal Quality

The range for orthogonal quality is 0-1, where a value of 0 is worst and a value of 1 is best.

Summary

It is useful to summarize the information that we have seen so far in tabular form. You may use this table when **verifying the quality of your mesh**.

Metric	Best	Worst
Element Quality	1	0
Aspect Ratio - Triangle	1	20
Aspect Ratio - Quadrilaterals	1	20
Jacobian Ratio	1	10
Warping Factor - Shell	0	5
Warping Factor - Brick	0	0.4
Parallel Deviation	1	170
Maximum Corner Angle - Triangular	60°	165
Maximum Corner Angle - Quadrilateral	90°	180°
Skewness - Triangular	0	1
Skewness - Quadrilateral	0	1
Orthogonal Quality	1	0

Table: ANSYS Mesh Metrics Summary

S.NO	Mesh Metrics	Average Quality	Comment
1			
2			
3			
4			
5			

Table: Mesh Metric Results

SOLUTION ALGORITHMS:

DISCRETIZATION SCHEMES FOR PRESSURE, MOMENTUM AND ENERGY EQUATIONS

1. PRESSURE EQUATION (CONTINUITY EQUATION)

The continuity equation for incompressible flow is:

$$\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}+\frac{\partial w}{\partial z}=0$$

Discretization schemes for pressure include:

a) Central Differencing Scheme (CDS):

- Second-order accurate.
- Uses values at neighbouring grid points to approximate derivatives.
- For a 1D grid, the derivative at point i*i* is:

$$\left.\frac{\partial \phi}{\partial x}\right|_i \approx \frac{\phi_{i+1}-\phi_{i-1}}{2\Delta x}$$

- **Advantages:** High accuracy for smooth flows.
- **Disadvantages:** Can cause oscillations in regions with sharp gradients.

b) Upwind Differencing Scheme (UDS):

- First-order accurate.
- Uses the value from the upstream (upwind) direction to approximate derivatives.
- For a 1D grid, if u>0:

$$\left.\frac{\partial \phi}{\partial x}\right|_i \approx \frac{\phi_{i}-\phi_{i-1}}{\Delta x}$$

- **Advantages:** Stable and avoids oscillations.
- **Disadvantages:** Introduces numerical diffusion, reducing accuracy.

c) Rhie-Chow Interpolation:

- Prevents pressure-velocity decoupling on collocated grids.
- Combines pressure and velocity interpolation to ensure smooth pressure fields.

- Expression for velocity at the cell face:

$$u_f = \overline{u}_f - D_f \left(\frac{\partial p}{\partial x} \right)_f$$

where D_f is a coefficient derived from the momentum equation.

2. MOMENTUM EQUATIONS (NAVIER-STOKES EQUATIONS)

The momentum equations in 1D (for simplicity) are:

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = -\frac{1}{\rho}\frac{\partial p}{\partial x} + \nu\frac{\partial^2 u}{\partial x^2}$$

Discretization schemes include:

a) Convective Term:

1. **Central Differencing Scheme (CDS):**

$$u\frac{\partial u}{\partial x} \approx u_i \left(\frac{u_{i+1} - u_{i-1}}{2\Delta x} \right)$$

   - **Advantages:** Second-order accurate.
   - **Disadvantages:** Can cause oscillations in high Reynolds number flows.

2. **Upwind Differencing Scheme (UDS):**

$$u\frac{\partial u}{\partial x} \approx u_i \left(\frac{u_i - u_{i-1}}{\Delta x} \right) \quad (\text{if } u > 0)$$

   - **Advantages:** Stable and robust.
   - **Disadvantages:** First-order accurate, introduces numerical diffusion.

3. **QUICK Scheme:**
   - Third-order accurate.
   - Uses a quadratic interpolation for the convective term:

$$u\frac{\partial u}{\partial x} \approx u_i\left(\frac{3u_{i+1} - 7u_i + 3u_{i-1} + u_{i-2}}{8\Delta x}\right)$$

- **Advantages:** Higher accuracy than CDS and UDS.
- **Disadvantages:** Can cause oscillations in complex flows.

b) Diffusive Term:

- Typically discretized using **Central Differencing Scheme (CDS):**

$$\nu\frac{\partial^2 u}{\partial x^2} \approx \nu\left(\frac{u_{i+1} - 2u_i + u_{i-1}}{\Delta x^2}\right)$$

- **Advantages:** Second-order accurate and stable for diffusion-dominated flows.

c) Pressure Gradient Discretization

The pressure gradient term $\left(\frac{1}{\rho}\frac{\partial P}{\partial x}\right)$ is discretized using finite difference or finite volume methods.

Common schemes include:

1. **Central Differencing Scheme (CDS):**
   - Second-order accurate.
   - For a 1D grid, the pressure gradient at point *ii* is:

$$\left.\frac{\partial p}{\partial x}\right|_i \approx \frac{p_{i+1} - p_{i-1}}{2\Delta x}$$

Advantages: High accuracy for smooth pressure fields.

- **Disadvantages:** Can cause oscillations in regions with sharp pressure gradients.

2. **Staggered Grid Arrangement:**
   - Pressure is stored at cell centres, while velocities are stored at cell faces.
   - The pressure gradient is computed as:

$$\left.\frac{\partial p}{\partial x}\right|_f \approx \frac{p_E - p_W}{\Delta x}$$

where P_E and P_W are pressures at the east and west faces, respectively.

- **Advantages:** Prevents pressure-velocity decoupling and ensures stability.

3. **Rhie-Chow Interpolation:**

- Used on colocated grids (where pressure and velocity are stored at the same locations).
- Interpolates velocities to cell faces to prevent oscillations:

$$u_f = \overline{u}_f - D_f \left(\frac{\partial p}{\partial x}\right)_f$$

where D_f is a coefficient derived from the momentum equation.

b) Pressure-Velocity Coupling

- In incompressible flows, pressure and velocity are coupled through the continuity equation $(\nabla.u = 0)$.
- Common algorithms for pressure-velocity coupling include:
  1. **SIMPLE (Semi-Implicit Method for Pressure-Linked Equations):**
     - Iteratively solves for pressure and velocity.
  2. **PISO (Pressure-Implicit with Splitting of Operators):**
     - Faster convergence for transient problems.
  3. **Fractional Step Method:**
     - Decouples pressure and velocity updates.

3. ENERGY EQUATION

The energy equation in 1D (for simplicity) is:

$$\frac{\partial T}{\partial t} + u\frac{\partial T}{\partial x} = \alpha\frac{\partial^2 T}{\partial x^2}$$

Discretization schemes include:

a) Convective Term:

- Similar to the momentum equations, schemes like **CDS**, **UDS**, or **QUICK** are used:

$$u\frac{\partial T}{\partial x} \approx u_i \left(\frac{T_{i+1} - T_{i-1}}{2\Delta x} \right) \quad \text{(CDS)}$$

$$u\frac{\partial T}{\partial x} \approx u_i \left(\frac{T_i - T_{i-1}}{\Delta x} \right) \quad \text{(UDS, if } u > 0\text{)}$$

b) Diffusive Term:

- Discretized using **Central Differencing Scheme (CDS):**

$$\alpha\frac{\partial^2 T}{\partial x^2} \approx \alpha \left(\frac{T_{i+1} - 2T_i + T_{i-1}}{\Delta x^2} \right)$$

c) Source Terms:

- Discretized based on their mathematical form. For example, a heat source S is directly added to the discretized equation.

4. Temporal Discretization

For transient simulations, temporal discretization schemes are used:

a) Explicit Schemes:

- **Forward Euler:**

$$\frac{\partial \phi}{\partial t} \approx \frac{\phi^{n+1} - \phi^n}{\Delta t}$$

  - **Advantages:** Simple to implement.
  - **Disadvantages:** Subject to CFL stability condition.

b) Implicit Schemes:

- **Backward Euler:**

$$\frac{\partial \phi}{\partial t} \approx \frac{\phi^{n+1} - \phi^n}{\Delta t}$$

  - **Advantages:** Unconditionally stable.

- **Disadvantages:** Requires solving a system of equations.

c) Crank-Nicolson Scheme:

- Second-order accurate in time:

$$\frac{\partial \phi}{\partial t} \approx \frac{\phi^{n+1} - \phi^n}{\Delta t} = \frac{1}{2}\left(F^{n+1} + F^n\right)$$

where F represents the spatial terms.

Summary of Schemes

Scheme	Advantages	Disadvantages
CDS	Second-order accurate	Oscillations in high-gradient regions
UDS	Stable, robust	First-order, introduces numerical diffusion
QUICK	Third-order accurate	Oscillations in complex flows
Rhie-Chow	Prevents pressure-velocity decoupling	Requires additional computation
Forward Euler	Simple to implement	Subject to **CFL condition**
Backward Euler	Unconditionally stable	Requires solving a system of equations

FIRST-ORDER UPWIND SCHEME (FOU)

a) Description

- A first-order accurate scheme for discretizing convective terms in the governing equations.
- Uses the valueto left):

$$\left.\frac{\partial \phi}{\partial x}\right|_i \approx \frac{\phi_{i+1} - \phi_i}{\Delta x}$$

c) Advantages

- Simple to implement.
- Highly stable and robust.

d) Disadvantages

- First-order accurate, leading to numerical diffusion (smearing of sharp gradients).
- Not suitable for high-accuracy simulations.

2. SECOND-ORDER UPWIND SCHEME (SOU)

a) Description

- A second-order accurate scheme for discretizing convective terms.
- Uses a linear interpolation of values from the upwind direction.

b) Mathematical Expression

- For a 1D grid, if $u > 0$:

$$\left.\frac{\partial \phi}{\partial x}\right|_i \approx \frac{3\phi_i - 4\phi_{i-1} + \phi_{i-2}}{2\Delta x}$$

- If $u < 0$:

$$\left.\frac{\partial \phi}{\partial x}\right|_i \approx \frac{-3\phi_i + 4\phi_{i+1} - \phi_{i+2}}{2\Delta x}$$

c) Advantages

- Higher accuracy than first-order upwind.
- Reduces numerical diffusion.

d) Disadvantages

- Can cause oscillations in regions with sharp gradients.
- Slightly more complex to implement than first-order upwind.

3. QUICK SCHEME (QUADRATIC UPSTREAM INTERPOLATION FOR CONVECTIVE KINEMATICS)

a) Description

- A third-order accurate scheme for discretizing convective terms.

- Uses a quadratic interpolation of values from the upwind direction.

b) Mathematical Expression

- For a 1D grid, if $u > 0$:

$$\phi_f = \frac{3}{8}\phi_{i+1} + \frac{6}{8}\phi_i - \frac{1}{8}\phi_{i-1}$$

- If $u < 0$:

$$\phi_f = \frac{3}{8}\phi_{i-1} + \frac{6}{8}\phi_i - \frac{1}{8}\phi_{i+1}$$

Where ϕ_f is the value at the cell face.

c) Advantages

- Higher accuracy than first- and second-order upwind schemes.
- Suitable for resolving sharp gradients.

d) Disadvantages

- Can cause oscillations in complex flows.
- Requires more computational effort.

PRESSURE-VELOCITY COUPLING SCHEMES (UNSTAGGERED GRID)

SIMPLE ALGORITHM

(Semi-Implicit Method for Pressure-Linked Equations)

a) Description

- SIMPLE is an iterative algorithm for solving the pressure-velocity coupling in incompressible flows.
- It solves the momentum and continuity equations sequentially.

b) Mathematical Expressions

1. **Momentum Equations:**
   Solve for intermediate velocities $u*$ and $v*$ using the guessed pressure field $p*$:

$$u^* = u^n + \Delta t \left(-\frac{1}{\rho} \frac{\partial p^*}{\partial x} + \nu \nabla^2 u^n \right)$$

$$v^* = v^n + \Delta t \left(-\frac{1}{\rho} \frac{\partial p^*}{\partial y} + \nu \nabla^2 v^n \right)$$

2. **Pressure Correction Equation:**
   Solve for the pressure correction p':

$$\nabla \cdot \left(\frac{1}{\rho} \nabla p' \right) = \nabla \cdot \mathbf{u}^*$$

3. **Velocity and Pressure Correction:**
   Correct the velocities and pressure:

$$u^{n+1} = u^* - \frac{\Delta t}{\rho} \frac{\partial p'}{\partial x}$$

$$v^{n+1} = v^* - \frac{\Delta t}{\rho} \frac{\partial p'}{\partial y}$$

$$p^{n+1} = p^* + p'$$

4. **Repeat until convergence.**

c) Advantages

- Robust and widely used.
- Suitable for steady-state and transient problems.

d) Disadvantages

- Slow convergence due to approximate pressure correction.
- Requires under-relaxation for stability.

SIMPLEC ALGORITHM

(SIMPLE-Consistent)

a) Description

- SIMPLEC is an improved version of the SIMPLE algorithm.
- It uses a more consistent approximation for the velocity correction terms, leading to faster convergence.

b) Mathematical Expressions

1. **Momentum Equations:**
   Solve for intermediate velocities $u*$ and $v*$ using the guessed pressure field $p*$:

$$u^* = u^n + \Delta t \left(-\frac{1}{\rho} \frac{\partial p^*}{\partial x} + \nu \nabla^2 u^n \right)$$

$$v^* = v^n + \Delta t \left(-\frac{1}{\rho} \frac{\partial p^*}{\partial y} + \nu \nabla^2 v^n \right)$$

2. **Pressure Correction Equation:**
   Solve for the pressure correction p':

$$\nabla \cdot \left(\frac{1}{\rho} \nabla p' \right) = \nabla \cdot \mathbf{u}^*$$

3. **Velocity and Pressure Correction:**
   Correct the velocities and pressure:

$$u^{n+1} = u^* - \frac{\Delta t}{\rho} \frac{\partial p'}{\partial x}$$

$$v^{n+1} = v^* - \frac{\Delta t}{\rho} \frac{\partial p'}{\partial y}$$

$$p^{n+1} = p^* + p'$$

4. **Repeat until convergence.**

c) Advantages

- Faster convergence than SIMPLE.
- More consistent approximation for velocity corrections.

d) Disadvantages

- Slightly more complex to implement than SIMPLE.

SIMPLER ALGORITHM

(SIMPLE REVISED)

a) Description

- SIMPLER is an improved version of the SIMPLE algorithm.
- It solves a separate pressure equation to obtain a better initial guess for pressure, leading to faster convergence.

b) Mathematical Expressions

1. **Pressure Equation:**
   Solve for an initial pressure field $p*$:

$$\nabla \cdot \left(\frac{1}{\rho} \nabla p^* \right) = \nabla \cdot \mathbf{u}^n$$

2. **Momentum Equations:**
   Solve for intermediate velocities $u*$ and $v*$ using $p*$:

$$u^* = u^n + \Delta t \left(-\frac{1}{\rho} \frac{\partial p^*}{\partial x} + \nu \nabla^2 u^n \right)$$

$$v^* = v^n + \Delta t \left(-\frac{1}{\rho} \frac{\partial p^*}{\partial y} + \nu \nabla^2 v^n \right)$$

3. **Pressure Correction Equation:**
   Solve for the pressure correction p':

$$\nabla \cdot \left(\frac{1}{\rho}\nabla p'\right) = \nabla \cdot \mathbf{u}^*$$

4. **Velocity Correction:**
   Correct the velocities:

$$u^{n+1} = u^* - \frac{\Delta t}{\rho}\frac{\partial p'}{\partial x}$$
$$v^{n+1} = v^* - \frac{\Delta t}{\rho}\frac{\partial p'}{\partial y}$$

5. **Repeat until convergence.**

c) Advantages

- Faster convergence than SIMPLE.
- More accurate initial pressure guess.

d) Disadvantages

- Slightly more complex to implement than SIMPLE.

PISO ALGORITHM

Mathematical Formulation

The **PISO** (Pressure-Implicit with Splitting of Operators) algorithm is an extension of the **SIMPLE** algorithm. It is particularly useful for **transient flows** because it performs **multiple pressure corrections** within each time step, improving accuracy and convergence.

Steps in PISO:

1. **Momentum Prediction**:

$$a_P \mathbf{u}_P^* = \sum_{nb} a_{nb}\mathbf{u}_{nb}^* + \mathbf{b} - \nabla p^{n-1}$$

   - **u∗**: Intermediate velocity.
   - p^{n-1}: Pressure from the previous time step.

2. **Pressure Correction**:

$$\nabla \cdot \left(\frac{1}{a_P} \nabla p' \right) = \nabla \cdot \mathbf{u}^*$$

   - p': Pressure correction.

3. **Velocity Correction**:

$$\mathbf{u}^{**} = \mathbf{u}^* - \frac{1}{a_P} \nabla p'$$

   - $\mathbf{u}^{**}$: Corrected velocity.

4. **Second Pressure Correction (Optional)**:

$$\nabla \cdot \left(\frac{1}{a_P} \nabla p'' \right) = \nabla \cdot \mathbf{u}^{**}$$

   - p'': Second pressure correction.

5. **Update Pressure**:

$$p^n = p^{n-1} + p' + p''$$

Advantages of PISO

- **Higher Accuracy**: Multiple pressure corrections improve accuracy for transient flows.
- **Stability**: Suitable for time-dependent problems with large time steps.

Disadvantages of PISO

- **Computational Cost**: Multiple corrections increase the computational effort per time step.
- **Memory Usage**: Requires additional storage for intermediate variables.

Coupled Algorithm

Mathematical Formulation

The **Coupled algorithm** solves the momentum and continuity equations **simultaneously** as a single system of equations. This approach is more robust and converges faster than segregated methods like SIMPLE or PISO, especially for **steady-state problems**.

Steps in Coupled Algorithm:

1. **Combined System**:

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ p \end{bmatrix} = \begin{bmatrix} \mathbf{f} \\ g \end{bmatrix}$$

   - A: Momentum equation coefficients.
   - B: Pressure gradient terms.
   - C: Continuity equation coefficients.
   - D: Pressure-velocity coupling terms.

2. **Solve the System**:

$$\begin{bmatrix} \mathbf{u} \\ p \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{f} \\ g \end{bmatrix}$$

3. **Update Variables**: $\mathbf{u}^n = \mathbf{u}$, $p^n = p$

Advantages of Coupled Algorithm

- **Faster Convergence**: Solves momentum and continuity equations simultaneously, reducing the number of iterations.
- **Robustness**: Handles strong pressure-velocity coupling more effectively.

Disadvantages of Coupled Algorithm

- **Memory Usage**: Requires significantly more memory than segregated methods.
- **Computational Cost**: Higher per-iteration cost due to the larger system of equations.

Comparison Table

Feature	SIMPLE	SIMPLEC	SIMPLER	PISO	Coupled
Equations Solved	Segregated	Segregated	Segregated	Segregated	Simultaneous
Pressure Correction	Single	Single	Direct pressure solution	Multiple	None
Convergence Speed	Slow	Moderate	Fast	Fast (transient)	Very Fast
Memory Usage	Low	Low	Moderate	Moderate	High
Stability	Requires under-relaxation	Requires less under-relaxation	Stable	Stable	Very Stable
Applications	Steady-state, simple flows	Steady-state, complex flows	Steady-state, complex flows	Transient flows	Steady-state, transient flows

When to Use Which Algorithm?

- **Use SIMPLE**:
  - For simple, steady-state problems with low computational resources.
- **Use SIMPLEC**:
  - For steady-state problems with strong pressure-velocity coupling.
- **Use SIMPLER**:
  - For steady-state problems requiring higher accuracy in the pressure field.
- **Use PISO**:
  - For transient flows with large time steps.
- **Use Coupled**:
  - For steady-state or transient flows requiring fast convergence and robustness.

MAC (MARKER-AND-CELL) ALGORITHM

The **MAC (Marker-and-Cell) algorithm** is a classic method for solving the incompressible Navier-Stokes equations, particularly for free-surface and multiphase flows. It uses a **staggered grid** arrangement to prevent pressure-velocity decoupling and employs marker particles to track fluid interfaces. Below is a detailed explanation of the MAC algorithm, including **mathematical expressions** and **figures** to illustrate the staggered grid and marker particles.

1. Staggered Grid Arrangement

a) Description

- In the MAC algorithm, **velocities** are stored at the **cell faces**, while **pressure** is stored at the **cell centres**.
- This arrangement ensures that pressure and velocity are naturally coupled, preventing oscillations in the pressure field.

b) Staggered Grid

```
      u (i-1/2, j)      u(i+1/2,j)
           ↓                ↓
p(i-1,j) ---●---------------- ●--- p(i,j)
            |                 |
            |                 |
            |                 |
            ●-----------------●
      v(i,j-1/2)        v(i,j+1/2)
```

- ●: Pressure nodes (cell centres).
- ↓: Horizontal velocity components (u) stored at vertical cell faces.
- →: Vertical velocity components (v) stored at horizontal cell faces.

2. Governing Equations

The incompressible Navier-Stokes equations are:

1. **Momentum Equations:**

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = -\frac{1}{\rho}\frac{\partial p}{\partial x} + \nu\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right)$$

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} = -\frac{1}{\rho}\frac{\partial p}{\partial y} + \nu\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right)$$

2. **Continuity Equation:**

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0$$

3. MAC Algorithm Steps

Step 1: Initialize Velocities and Pressure

- Guess initial velocities u, v and pressure p.

Step 2: Solve Momentum Equations

- Discretize the momentum equations using finite differences on the staggered grid.
- For example, the u-momentum equation at cell face (i+1/2, j) (i+1/2, j):

$$u^{n+1}_{i+1/2,j} = u^n_{i+1/2,j} + \Delta t\left[-\frac{1}{\rho}\frac{p^n_{i+1,j} - p^n_{i,j}}{\Delta x}\right.$$
$$\left.+\nu\left(\frac{u^n_{i+3/2,j} - 2u^n_{i+1/2,j} + u^n_{i-1/2,j}}{\Delta x^2} + \frac{u^n_{i+1/2,j+1} - 2u^n_{i+1/2,j} + u^n_{i+1/2,j-1}}{\Delta y^2}\right)\right]$$

Step 3: Solve Continuity Equation

- Use the intermediate velocities to enforce mass conservation.
- The continuity equation is discretized as:

$$\frac{u^{n+1}_{i+1/2,j} - u^{n+1}_{i-1/2,j}}{\Delta x} + \frac{v^{n+1}_{i,j+1/2} - v^{n+1}_{i,j-1/2}}{\Delta y} = 0$$

Step 4: Pressure Correction

- Solve the pressure Poisson equation to correct the pressure field:

$$\nabla^2 p^{n+1} = \frac{\rho}{\Delta t}\left(\frac{\partial u^*}{\partial x} + \frac{\partial v^*}{\partial y}\right)$$

where $u*$ and $v*$ are the intermediate velocities.

Step 5: Update Velocities

- Correct the velocities using the updated pressure field:

$$u_{i+1/2,j}^{n+1} = u_{i+1/2,j}^{*} - \frac{\Delta t}{\rho} \frac{p_{i+1,j}^{n+1} - p_{i,j}^{n+1}}{\Delta x}$$

$$v_{i,j+1/2}^{n+1} = v_{i,j+1/2}^{*} - \frac{\Delta t}{\rho} \frac{p_{i,j+1}^{n+1} - p_{i,j}^{n+1}}{\Delta y}$$

Step 6: Marker Particles

- For free-surface flows, use marker particles to track the fluid interface.
- Update the position of marker particles using the velocity field:

$$\mathbf{x}_p^{n+1} = \mathbf{x}_p^n + \mathbf{u}_p^n \Delta t$$

where $\mathbf{u}^{p}{}_{n}$ is the velocity interpolated at the marker particle location.

5. Advantages of MAC Algorithm

- Prevents pressure-velocity decoupling due to the staggered grid.
- Suitable for free-surface and multiphase flows.
- Robust for incompressible flows.

6. Disadvantages of MAC Algorithm

- Requires careful handling of boundary conditions.
- Less efficient than modern algorithms like SIMPLE or SIMPLER.

CFD SOLUTION PROCEDURE

The **CFD solution procedure** involves a systematic process to simulate fluid flow and heat transfer problems. Below is a detailed explanation of each step, including **mathematical expressions** where applicable.

1. Problem Setup

a) Creation of Geometry

- Define the physical domain of the problem (e.g., a pipe, airfoil, or room).
- Use CAD software or built-in tools in CFD software to create the geometry.
- Ensure the geometry accurately represents the physical problem.

b) Mathematical Representation

- The geometry defines the boundaries of the computational domain, where the governing equations (Navier-Stokes, energy, etc.) are solved:
  Ω=Computational Domain, $\partial\Omega$=Boundary of the Domain

2. Mesh Generation

a) Description

- Discretize the geometry into small cells or elements (mesh).
- The mesh can be structured (regular grid) or unstructured (irregular grid).

b) Mathematical Representation

- The domain Ω is divided into N cells or elements:

$$\Omega = \bigcup_{i=1}^{N} \Omega_i$$

where Ωi is the i-th cell.

c) Mesh Quality

- Ensure the mesh has good quality (e.g., low skewness, aspect ratio close to 1).
- Use finer mesh in regions with high gradients (e.g., near walls, in boundary layers).

3. Selection of Physics and Fluid Properties

a) Physics Models

- Choose the appropriate governing equations and models:
  - **Navier-Stokes Equations** for fluid flow:

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} = -\frac{1}{\rho}\nabla p + \nu \nabla^2 \mathbf{u} + \mathbf{f}$$

  - **Energy Equation** for heat transfer:

$$\frac{\partial T}{\partial t} + (\mathbf{u} \cdot \nabla)T = \alpha \nabla^2 T + \frac{q}{\rho c_p}$$

- **Turbulence Models** (e.g., k-ϵ, k-ω, SST) for turbulent flows.

b) Fluid Properties

- Specify fluid properties such as density (ρ), viscosity (v), thermal conductivity (k), and specific heat (c_p).

4. Initialization

a) Description

- Set initial conditions for the flow field (e.g., velocity, pressure, temperature).
- Initial conditions are required to start the simulation.

b) Mathematical Representation

- For example, initialize velocity and pressure fields:

$$\mathbf{u}(\mathbf{x}, t=0) = \mathbf{u}_0(\mathbf{x}), \quad p(\mathbf{x}, t=0) = p_0(\mathbf{x})$$

5. Solution Control and Convergence Monitoring

a) Solution Control

- Set numerical parameters such as:
  - Time step size (Δt) for transient simulations.
  - Under-relaxation factors for iterative solvers.
  - Convergence criteria (e.g., residual thresholds).

b) Residuals

- Monitor residuals to check convergence. For example, the residual for the momentum equation is:

$$R_u = \left\| \frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} + \frac{1}{\rho}\nabla p - \nu \nabla^2 \mathbf{u} - \mathbf{f} \right\|$$

- Convergence is achieved when residuals fall below a specified threshold (e.g. $R_u < 10^{-6}$).

c) Iterative Solvers

- Use iterative solvers (e.g., Gauss-Seidel, Conjugate Gradient) to solve the discretized equations.

6. Results Reports and Visualization

a) Post-Processing

- Extract and analyze results such as:
    - Velocity fields ($\mathbf{u}$).
    - Pressure distribution (p).
    - Temperature distribution (T).
    - Turbulence quantities (e.g., k, ϵ).

b) Mathematical Representation

- For example, calculate the drag force on a body:

$$F_D = \int_{\partial\Omega} (-p\mathbf{n} + \tau \cdot \mathbf{n})\, dA$$

where τ is the viscous stress tensor and $\mathbf{n}$ is the unit normal vector.

c) Visualization

- Use visualization tools to create:
    - Contour plots (e.g., pressure contours).
    - Streamlines or path lines.
    - Vector plots (e.g., velocity vectors).
    - Animations for transient simulations.

PRESSURE-BASED VS. DENSITY-BASED SOLVERS IN ANSYS FLUENT

In ANSYS Fluent, the choice between **pressure-based** and **density-based** solvers depends on the type of flow problem you are solving. Both solvers have distinct approaches to solving the governing equations of fluid flow (Navier-Stokes equations). Below is a detailed comparison of the two solvers, including their applications, advantages, and limitations.

1. PRESSURE-BASED SOLVER

The pressure-based solver is the traditional approach used in Fluent and is well-suited for incompressible and low-speed compressible flows.

Key Features:

- **Primary Variable**: Pressure is the primary variable, and the continuity equation is used to enforce mass conservation.
- **Algorithm**: Uses a segregated approach, where the equations for momentum, pressure, and other scalars are solved sequentially.
- **Pressure-Velocity Coupling**: Methods like SIMPLE, SIMPLEC, or PISO are used to couple pressure and velocity.
- **Applications**:
    - Incompressible flows (e.g., water flow, air at low speeds).
    - Low-Mach-number compressible flows (Mach number < 0.3).
    - Steady-state or transient simulations.
    - Multiphase flows (e.g., VOF, Eulerian models).
- **Advantages**:
    - Efficient for incompressible and low-speed flows.
    - Robust and stable for a wide range of problems.
    - Lower memory usage compared to the density-based solver.
- **Limitations**:
    - Less accurate for high-speed compressible flows (e.g., supersonic flows).
    - Slower convergence for compressible flows compared to the density-based solver.

2. DENSITY-BASED SOLVER

The density-based solver is designed for high-speed compressible flows, where density variations are significant.

Key Features:

- **Primary Variable**: Density is the primary variable, and the continuity, momentum, and energy equations are solved simultaneously.
- **Algorithm**: Uses a coupled approach, where the governing equations are solved together as a system.
- **Applications**:
    - Compressible flows (e.g., supersonic or hypersonic flows).
    - High-Mach-number flows (Mach number > 0.3).
    - Flows with strong shocks or discontinuities.
    - Transient simulations with rapid changes in flow properties.
- **Advantages**:
    - Accurate and efficient for high-speed compressible flows.
    - Better handling of shocks and discontinuities.
    - Faster convergence for compressible flows.
- **Limitations**:
    - Higher memory usage due to the coupled solution approach.
    - Less efficient for incompressible or low-speed flows.
    - May require more tuning for stability in complex flows.

When to Use Which Solver?

1. **Pressure-Based Solver**:
    - Use for incompressible flows (e.g., water, air at low speeds).
    - Use for multiphase flows (e.g., VOF, Eulerian models).
    - Use for steady-state or transient simulations with low Mach numbers.

2. **Density-Based Solver**:
    - Use for compressible flows (e.g., supersonic or hypersonic flows).
    - Use for flows with strong shocks or discontinuities.
    - Use for high-Mach-number flows (Mach > 0.3).

Switching Between Solvers

In Fluent, you can switch between pressure-based and density-based solvers in the **General** task page under the **Solver** section. However, this switch may require reinitializing the solution and adjusting solver settings.

Tips for Using Each Solver

Pressure-Based Solver:

- Use SIMPLE or SIMPLEC for steady-state problems.
- Use PISO for transient problems.
- Adjust under-relaxation factors for stability (e.g., lower values for pressure and momentum).

Density-Based Solver:

- Use the implicit formulation for better stability.
- Use explicit formulation for transient problems with rapid changes.
- Monitor Courant number (CFL) for stability (lower values for better stability).

Understanding and Optimizing Relaxation Factors in ANSYS Fluent for Stable and Efficient CFD Simulations

In ANSYS Fluent, relaxation factors are used to control the rate at which the solution variables (such as pressure, velocity, temperature, etc.) are updated during the iterative solution process. Properly setting relaxation factors is crucial for achieving convergence in computational fluid dynamics (CFD) simulations. Here's an overview of relaxation factors in Fluent:

What are Relaxation Factors?

Relaxation factors are numerical parameters that determine how much the solution variables are adjusted in each iteration. They help stabilize the solution process, especially in cases where the equations are highly nonlinear or coupled.

- **Under-relaxation**: Slows down the change in solution variables to improve stability.

- **Over-relaxation**: Speeds up the change in solution variables, which can accelerate convergence but may lead to instability.

In Fluent, under-relaxation is typically used for most variables.

Key Relaxation Factors in Fluent

The following are the primary relaxation factors you can adjust in Fluent:

1. **Pressure**:
    - Controls the update of the pressure field.
    - Typical range: 0.2 to 0.7 (default is often 0.3).
2. **Density**:
    - Used for compressible flows.
    - Typical range: 0.5 to 1.0 (default is 1.0).
3. **Body Forces**:
    - Affects the treatment of body forces (e.g., gravity).
    - Typical range: 0.5 to 1.0 (default is 1.0).
4. **Momentum**:
    - Controls the update of velocity components.
    - Typical range: 0.2 to 0.8 (default is often 0.7).
5. **Turbulence Kinetic Energy (k)**:
    - Affects the update of the turbulence kinetic energy.
    - Typical range: 0.4 to 0.8 (default is often 0.8).
6. **Turbulence Dissipation Rate (ε) or Specific Dissipation Rate (ω)**:
    - Controls the update of turbulence dissipation rate or specific dissipation rate.
    - Typical range: 0.4 to 0.8 (default is often 0.8).
7. **Energy**:
    - Used for temperature or energy equations.

- Typical range: 0.5 to 1.0 (default is often 1.0).

8. **Volume Fraction (for multiphase flows)**:
    - Controls the update of phase volume fractions.
    - Typical range: 0.2 to 0.8 (default is often 0.5).

How to Set Relaxation Factors

1. In Fluent, go to **Solution Controls** in the Solution tab.
2. Adjust the relaxation factors for the relevant variables.
3. Monitor the residuals and convergence behavior to determine if further adjustments are needed.

Guidelines for Adjusting Relaxation Factors

- **For Stability**:
    - If the solution is diverging or oscillating, reduce the relaxation factors (especially for pressure and momentum).
- **For Faster Convergence**:
    - If the solution is stable but converging slowly, gradually increase the relaxation factors.
- **Default Values**:
    - Start with the default values and adjust only if necessary.
- **Multiphase Flows**:
    - Use lower relaxation factors for volume fractions to ensure stability.

Common Issues and Fixes

1. **Divergence**:
    - Reduce relaxation factors, especially for pressure and momentum.
2. **Slow Convergence**:
    - Increase relaxation factors slightly, but monitor for instability.
3. **Oscillations**:
    - Reduce relaxation factors and ensure proper mesh quality and boundary conditions.

Advanced Techniques

- **Coupled Solvers**:
    - When using coupled solvers (e.g., pressure-based coupled or density-based coupled), relaxation factors are often handled differently, and fewer adjustments may be needed.
- **Adaptive Relaxation**:
    - Some solvers or custom schemes can adapt relaxation factors dynamically based on the solution behaviour.

BOUNDARY CONDITIONS

In computational fluid dynamics (CFD), **boundary conditions** are essential for defining the behaviour of the flow at the boundaries of the computational domain. Here's an overview of the **different boundary conditions** commonly used in ANSYS Fluent and other CFD solvers:

1. Inlet Boundary Conditions

These define the flow entering the domain.

a. Velocity Inlet

- Specifies the velocity components at the inlet.
- **Equation**: $\mathbf{u}=(u_x,u_y,u_z)$
- **Use Case**: Known velocity profile (e.g., uniform or parabolic).

b. Pressure Inlet

- Specifies the total pressure at the inlet.
- **Equation**: $P_{\text{total}}=P_{\text{static}}+0.5\rho|\mathbf{u}|^2$
- **Use Case**: Flow rate not known, but pressure is.

c. Mass Flow Inlet

- Specifies the mass flow rate at the inlet.
- **Equation:** $\dot{m} = \rho u A$
- **Use Case**: Known mass flow rate (e.g., pipe flow).

2. Outlet Boundary Conditions

These define the flow exiting the domain.

a. Pressure Outlet

- Specifies the static pressure at the outlet.
- **Equation**: $P_{static}=P_{outlet}$
- **Use Case**: Flow exits to a known pressure (e.g., atmospheric pressure).

b. Outflow

- Assumes zero gradients for all flow variables at the outlet.
- **Use Case**: Fully developed flow (no reverse flow allowed).

c. Pressure Far-Field

- Used for external flows (e.g., aerodynamics).
- Specifies freestream pressure and Mach number.
- **Use Case**: Compressible flows (e.g., supersonic or subsonic).

3. Wall Boundary Conditions

These define the interaction between the fluid and solid surfaces.

a. No-Slip Wall

- Velocity at the wall is zero (fluid sticks to the wall).
- **Equation**: $\mathbf{u}=0$
- **Use Case**: Most common for viscous flows.

b. Slip Wall

- Velocity at the wall is non-zero (no friction).
- **Equation**: $\partial n/\partial \mathbf{u}=0$
- **Use Case**: Inviscid or free-slip flows.

c. Moving Wall

- Wall moves with a specified velocity.
- **Equation**: $\mathbf{u}=\mathbf{u}_{wall}$
- **Use Case**: Rotating or translating surfaces (e.g., fan blades).

4. Symmetry Boundary Conditions

These are used to reduce computational cost by exploiting symmetry.

a. Symmetry Plane

- Assumes zero normal velocity and zero gradients across the plane.
- **Equation**: $\mathbf{u}\cdot\mathbf{n}=0$
- **Use Case**: Symmetric geometries (e.g., flow over a cylinder).

5. Periodic Boundary Conditions

These are used for flows with repeating patterns.

a. Periodic

- Links two boundaries with identical flow conditions.
- **Equation**: $\mathbf{u}(x)=\mathbf{u}(x+L)$

Where L is the periodicity length.

- **Use Case**: Turbomachinery, heat exchangers.

6. Special Boundary Conditions

a. Axis Boundary

- Used for axisymmetric flows.
- **Equation**:$\partial r/\partial \mathbf{u}=0$ at $r=0$
- **Use Case**: Axisymmetric geometries (e.g., nozzles, pipes).

b. Fan Boundary

- Models a fan with a pressure jump.
- **Equation**: $\Delta P=f$ (flow rate)
- **Use Case**: HVAC systems, cooling fans.

c. Radiative Boundary

- Models heat transfer via radiation.
- **Equation**: $q_{\text{rad}}=\epsilon\sigma(T^4-T_\infty{}^4)$
- **Use Case**: High-temperature flows (e.g., combustion).

CFD SOLUTION PROCEDURE

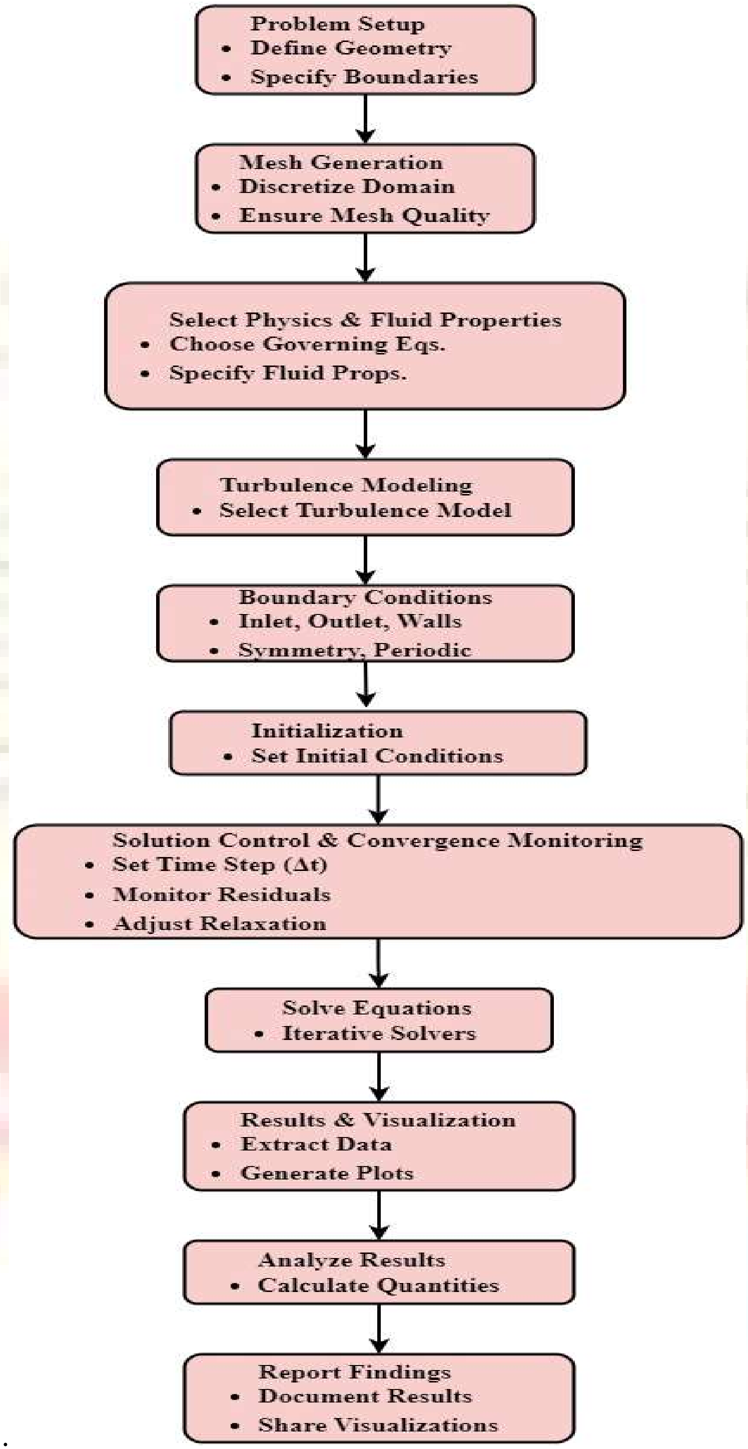


Figure: CFD Solution Procedure Flow Chat

Turbulence Modelling in ANSYS Fluent

Turbulence modelling is essential for simulating complex flows where chaotic, unsteady vortices dominate. Fluent provides several turbulence models to approximate the effects of turbulence without resolving all scales directly.

1. Reynolds-Averaged Navier-Stokes (RANS) Equations

RANS models decompose flow variables into mean and fluctuating components:

$$u_i = \overline{u_i} + u_i'$$

where:

- $\overline{u_i}$: Time-averaged velocity component.
- u_i' : Fluctuating velocity component.

The RANS equations for incompressible flow are:

$$\frac{\partial \overline{u_i}}{\partial t} + \overline{u_j}\frac{\partial \overline{u_i}}{\partial x_j} = -\frac{1}{\rho}\frac{\partial \overline{p}}{\partial x_i} + \nu \frac{\partial^2 \overline{u_i}}{\partial x_j \partial x_j} - \frac{\partial \overline{u_i' u_j'}}{\partial x_j}$$

where:

- $\overline{u_i u_j}$ Reynolds stress tensor (requires modelling).

2. Common Turbulence Models in Fluent

a. k-ε (k-Epsilon) Model

- **Transport Equations**:

$$\frac{\partial(\rho k)}{\partial t} + \frac{\partial(\rho k u_i)}{\partial x_i} = \frac{\partial}{\partial x_j}\left[\left(\mu + \frac{\mu_t}{\sigma_k}\right)\frac{\partial k}{\partial x_j}\right] + P_k - \rho\epsilon$$

$$\frac{\partial(\rho \epsilon)}{\partial t} + \frac{\partial(\rho \epsilon u_i)}{\partial x_i} = \frac{\partial}{\partial x_j}\left[\left(\mu + \frac{\mu_t}{\sigma_\epsilon}\right)\frac{\partial \epsilon}{\partial x_j}\right] + C_{1\epsilon}\frac{\epsilon}{k}P_k - C_{2\epsilon}\rho\frac{\epsilon^2}{k}$$

where:

- k: Turbulent kinetic energy.
- ϵ: Turbulent dissipation rate.
- μ_t: Turbulent viscosity ($\mu_t = \rho C_\mu \frac{k^2}{\epsilon}$).
- P_k: Production of turbulent kinetic energy.

- **Strengths**: Robust and computationally efficient for high-Reynolds-number flows.
- **Weaknesses**: Poor performance in flows with strong separation, curvature, or adverse pressure gradients.

b. k-ω (k-Omega) Model

- **Transport Equations**:

$$\frac{\partial(\rho k)}{\partial t} + \frac{\partial(\rho k u_i)}{\partial x_i} = \frac{\partial}{\partial x_j}\left[\left(\mu + \frac{\mu_t}{\sigma_k}\right)\frac{\partial k}{\partial x_j}\right] + P_k - \beta^* \rho k \omega$$

$$\frac{\partial(\rho \omega)}{\partial t} + \frac{\partial(\rho \omega u_i)}{\partial x_i} = \frac{\partial}{\partial x_j}\left[\left(\mu + \frac{\mu_t}{\sigma_\omega}\right)\frac{\partial \omega}{\partial x_j}\right] + \alpha \frac{\omega}{k} P_k - \beta \rho \omega^2$$

where:

- ω: Specific dissipation rate.
- $\mu_t = \rho \frac{k}{\omega}$.

- **Strengths**: Better performance in near-wall regions and low-Reynolds-number flows.
- **Weaknesses**: Sensitive to free-stream conditions.

c. SST (Shear Stress Transport) Model

- Combines k-ε (free stream) and k-ω (near-wall) models using a blending function.
- **Transport Equations**:

$$\frac{\partial(\rho k)}{\partial t}+\frac{\partial(\rho k u_i)}{\partial x_i}=\frac{\partial}{\partial x_j}\left[\left(\mu+\frac{\mu_t}{\sigma_k}\right)\frac{\partial k}{\partial x_j}\right]+P_k-\beta^*\rho k\omega$$

$$\frac{\partial(\rho\omega)}{\partial t}+\frac{\partial(\rho\omega u_i)}{\partial x_i}=\frac{\partial}{\partial x_j}\left[\left(\mu+\frac{\mu_t}{\sigma_\omega}\right)\frac{\partial \omega}{\partial x_j}\right]+\alpha\frac{\omega}{k}P_k-\beta\rho\omega^2+2(1-F_1)\rho\sigma_{\omega 2}\frac{1}{\omega}\frac{\partial k}{\partial x_j}\frac{\partial \omega}{\partial x_j}$$

where:

- F_1: Blending function.

- **Strengths**: Accurate for flows with adverse pressure gradients and separation.
- **Weaknesses**: Slightly more computationally expensive than k-ε or k-ω.

d. LES (Large Eddy Simulation)

- Resolves large-scale eddies directly and models small-scale eddies using a sub grid-scale (SGS) model.
- **Filtered Navier-Stokes Equations**:

$$\frac{\partial \overline{u_i}}{\partial t}+\frac{\partial(\overline{u_i u_j})}{\partial x_j}=-\frac{1}{\rho}\frac{\partial \overline{p}}{\partial x_i}+\nu\frac{\partial^2 \overline{u_i}}{\partial x_j \partial x_j}-\frac{\partial \tau_{ij}}{\partial x_j}$$

where:

- τ_{ij}: Subgrid-scale stress tensor.

- **Strengths**: Captures transient flow features accurately.
- **Weaknesses**: Computationally expensive.

e. DES (Detached Eddy Simulation)

- Hybrid model combining RANS (near-wall) and LES (free stream).
- **Strengths**: Balances accuracy and computational cost for separated flows.
- **Weaknesses**: Requires careful grid design.

3. Selection Criteria

- **k-ε**: High-Reynolds-number flows, simple geometries.
- **k-ω**: Near-wall accuracy, low-Reynolds-number flows.
- **SST**: Complex flows with separation and adverse pressure gradients.
- **LES/DES**: Transient flows with large-scale vortices.

Y+ Value in Turbulence Modelling

The **Y+ value** is a non-dimensional distance used to characterize the resolution of the mesh near the wall. It is defined as:

$$Y^+ = \frac{y u_\tau}{\nu}$$

where:

- y: Distance from the wall to the first mesh node.
- u_τ: Friction velocity $\left(u_\tau = \sqrt{\frac{\tau_w}{\rho}} \right)$.
- ν: Kinematic viscosity of the fluid.
- τ_w: Wall shear stress.

Y+ Guidelines for Turbulence Models

1. **Low Y+ (Y+ < 5)**:
    - Required for **low-Reynolds-number models** (e.g., k-ω SST, k-ε low-Re).
    - Mesh resolves the **viscous sublayer**.
    - Suitable for accurate near-wall flow predictions.
2. **Moderate Y+ (5 < Y+ < 30)**:
    - Used with **wall functions** (e.g., standard k-ε, k-ω with wall functions).
    - Mesh skips the viscous sublayer but resolves the **log-law region**.
    - Balances accuracy and computational cost.
3. **High Y+ (Y+ > 30)**:
    - Used with **wall functions** in high-Reynolds-number flows.
    - Mesh does not resolve the near-wall region.
    - Less accurate but computationally efficient.

Turbulence Modeling Notes with Y+ Considerations

1. k-ε Model

- **Y+ Range**: 30 < Y+ < 300 (with wall functions).
- **Equation for Turbulent Viscosity**:

$$\mu_t = \rho C_\mu \frac{k^2}{\varepsilon}$$

- **Wall Treatment**: Standard wall functions or scalable wall functions.

2. k-ω SST Model

- **Y+ Range**: Y+ < 5 (low-Re) or 30 < Y+ < 300 (with wall functions).
- **Equation for Turbulent Viscosity**:

$$\mu_t = \rho \frac{k}{\omega}$$

- **Wall Treatment**: Automatic wall treatment (blends low-Re and wall functions).

3. LES (Large Eddy Simulation)

- **Y+ Range**: Y+ < 1 (resolves viscous sublayer).
- **Subgrid-Scale Stress**:

$$\tau_{ij} = -2\mu_t \overline{S}_{ij}$$

where $\overline{S}_{ij}$ is the resolved strain rate tensor.

- **Wall Treatment**: Requires fine mesh near the wall.

4. DES (Detached Eddy Simulation)

- **Y+ Range**: Y+ < 1 (near-wall RANS) or Y+ > 30 (LES region).
- **Wall Treatment**: Hybrid approach (RANS near wall, LES in free stream).

Steps to Ensure Proper Y+ Values in Fluent

1. **Estimate Y+**:
    - Use the formula:

$$y = \frac{Y^+ u_\tau}{\nu}$$

- Estimate u_τ based on flow conditions.

2. **Generate Mesh**:
   - Use inflation layers near walls to achieve the desired Y+.
   - Ensure sufficient layers (e.g., 10–20 layers) with a growth ratio of 1.2–1.5.
3. **Check Y+ After Simulation**:
   - Use Fluent's post-processing tools to verify Y+ values on the walls.

Summary of Y+ and Turbulence Models

Model	Y+ Range	Wall Treatment	Application
k-ε	30 < Y+ < 300	Standard wall functions	High-Re flows, simple geometries
k-ω SST	Y+ < 5 or 30 < Y+	Automatic wall treatment	Complex flows, separation, adverse Pressure gradient
LES	Y+ < 1	Resolve viscous sublayer	Transient flows, large-scale vortices
DES	Y+ < 1 (RANS)	Hybrid RANS-LES	Separated flows, unsteady simulations

Experiment No: - 2

INCOMPRESSIBLE INTERNAL LAMINAR FLOWS IN PIPE

Q) **Consider the flow of oil at 20^{0}C in a 30cm diameter pipeline at an average velocity of 2m/s. A 200 m long section of the pipe line passes through a very cold water of a lake at 0^{0}C. Measurements indicate that the surface temperature of the pipe is very nearly 0^{0}C. Disregarding the thermal resistance of the pipe material, Determine**

(a) The temperature of the oil when the pipe leaves the lake

(b) The pumping powder required to overcome the pressure losses and to maintain the flow of the oil in the pipes and

(c) Examine the velocity and thermal boundary layers

Properties of the oil as

ρ = 888 Kg/m^3, v= 901 x 10^{-6} m^2/ s, k= 0.145 w/m. k, c_p = 1880 J/kg-k, Pr = 10,400

AIM

SOFTWARE REQUIREMENT

THEORY

Convective heat transfer, often referred to simply as convection, is the transfer of heat from one place to another by the movement of fluids. Convection is usually the dominant form of heat transfer in liquids and gases. Although often discussed as a distinct method of heat transfer, convective heat transfer involves the combined processes of unknown conduction (heat diffusion) and advection (heat transfer by bulk fluid flow). Internal and external flow can also classify convection. Internal flow occurs when a fluid is enclosed by a solid boundary such when flowing through a pipe. An external flow occurs when a fluid extends indefinitely without encountering a solid surface. Both of these types of convection, either natural or forced, can be internal or external because they are independent of each other. The bulk temperature, or the average fluid temperature, is a convenient reference point for evaluating properties related to convective heat transfer, particularly in applications related to flow in pipes and ducts. The below figures shows velocity distribution and thermal distribution of fluid flow in a pipe.

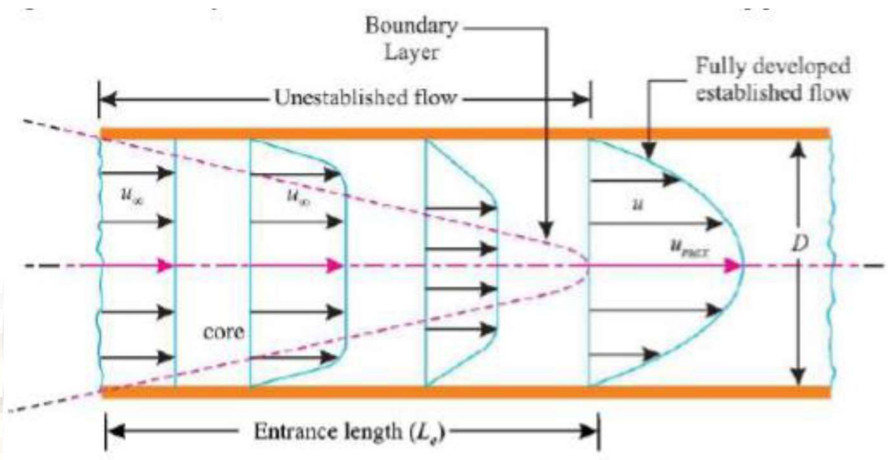


Figure: Velocity Boundary Layer Formation in a Pipe

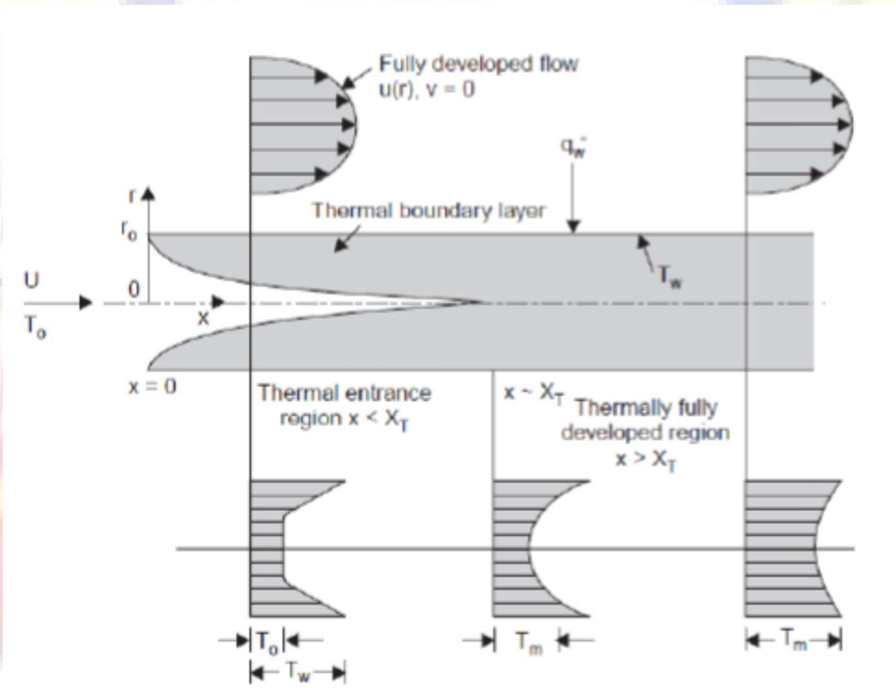


Figure: Thermal Boundary Layer Formation in a Pipe

PROCEDURE

RESULT AND CONCLUSION:

Experiment No: - 3

3D UNSTRUCTURED MESH OF PIPE

1 Q) **Generate a high-quality 3D unstructured mesh for a cylindrical pipe with a consistent radius, having a diameter of 0.2 meters and a length of 3 meters, while accounting for a pipe thickness of 0.02 meters. Enhance mesh quality parameters to meet acceptable standards and determine the final mesh metrics.**

AIM

SOFTWARE REQUIREMENT

THEORY

Inflation mesh is an option to use when we need to refine the zone near to the wall. Inflation is use when analysis of boundary condition is done accurately by create finer mesh near to wall or near to boundary. Inflation is also useful for CFD boundary layer resolution, electromagnetic air gap resolution or resolving high stress concentrations.

Global Inflation Controls:

The Inflation group of global mesh controls available in the Details View when the Mesh option is selected in the Tree Outline. There are many different basic options available in inflation. The options in the Inflation group provide global control over all inflation boundaries.

These options are below :

1. Use Automatic Inflation
2. Inflation Option
3. Transition Ratio
4. Maximum Layers
5. Growth Rate
6. Number of Layers
7. Maximum Thickness
8. First Layer Height

9. First Aspect Ratio
10. Aspect Ratio (Base/Height)
11. Inflation Algorithm
12. View Advanced Options

Use Automatic Inflation:

If you are doing simple analysis near to boundary & no much accuracy is necessary, then you can use "Use Automatic Inflation" option. You can set the Use Automatic Inflation control so that inflation boundaries are selected automatically depending on whether or not they are members of Named Selections groups. While using inflation, keep in mind that

The following options are available in Use Automatic Inflation:

None

Program Controlled

All Faces in Chosen Named Selection

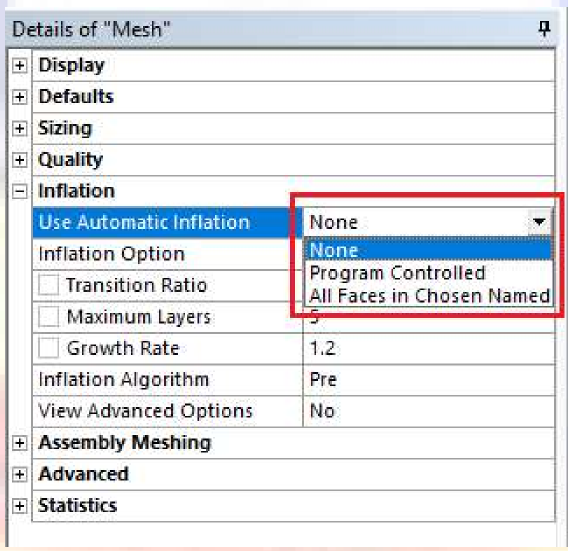


Figure: Mesh Details

While using inflation, keep in mind that Automatic inflation is supported only for 3D inflation on volume models. It is not supported for 2D inflation on the shell model. You cannot select Program Controlled or All Faces in Chosen Named Selection option for the Use Automatic Inflation option to mesh a 2D model. To apply 2D inflation on a shell model, use local inflation mesh controls.

Here, I create basic geometry & assigned a name on the face of an object, showing how inflation work & what change occurs in an object. This geometry is just for example for understanding inflation.

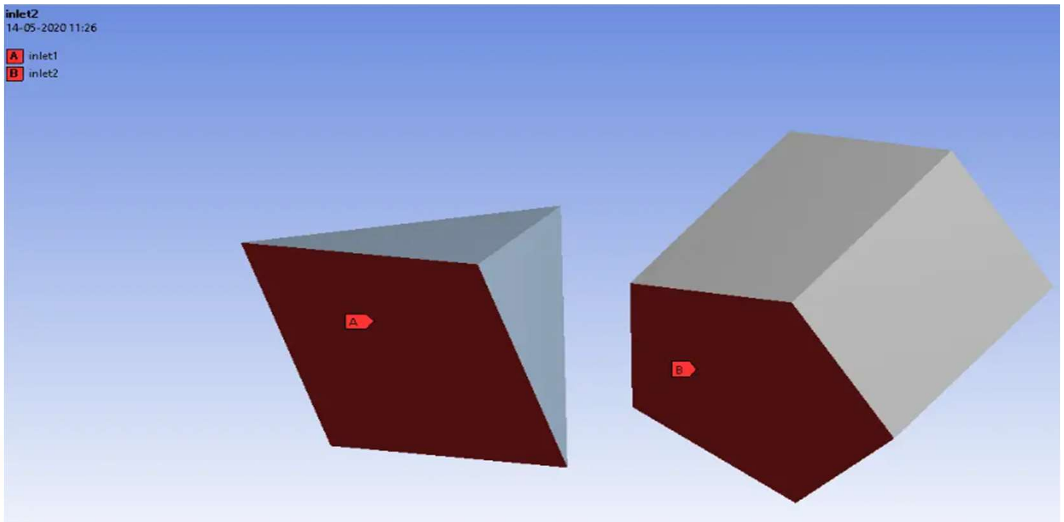


Figure: Geometry Model for Inflation

None

If you select None, inflation boundaries are not selected globally.

Program Controlled

If you select Program Controlled, all faces in the model are selected to be inflation boundaries, except Faces which is used in Named Selection(s).

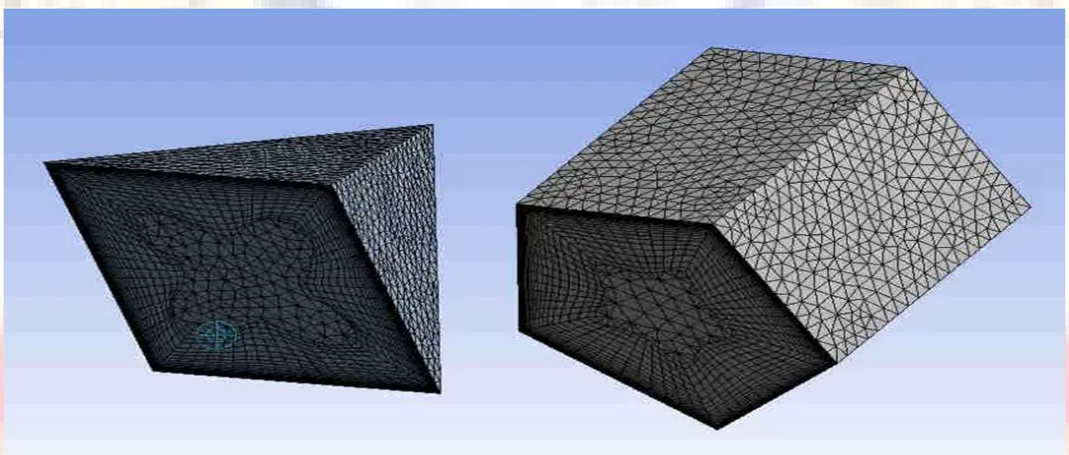
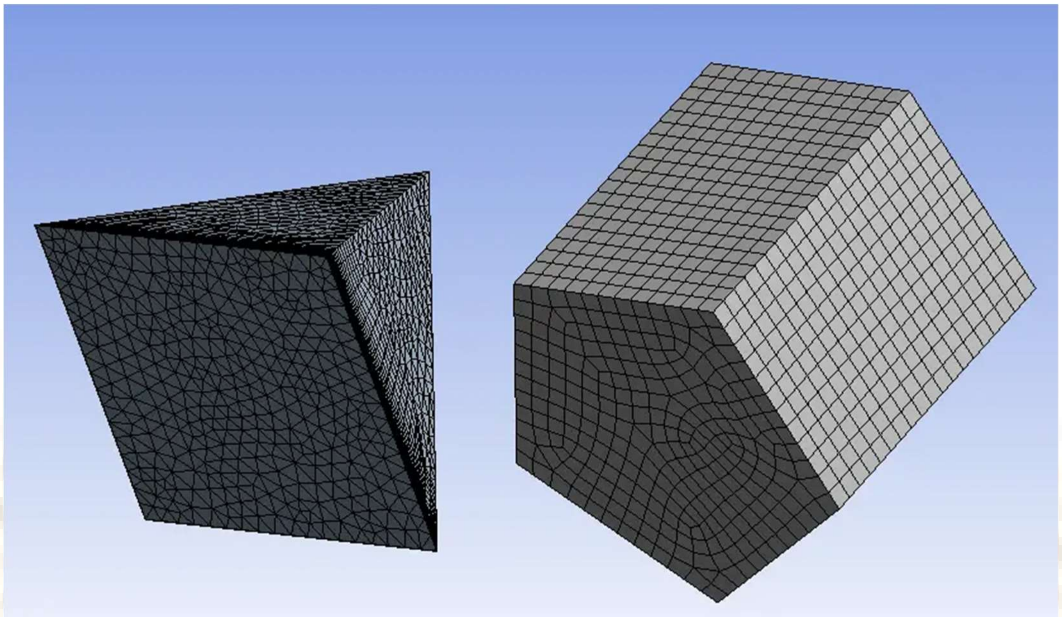


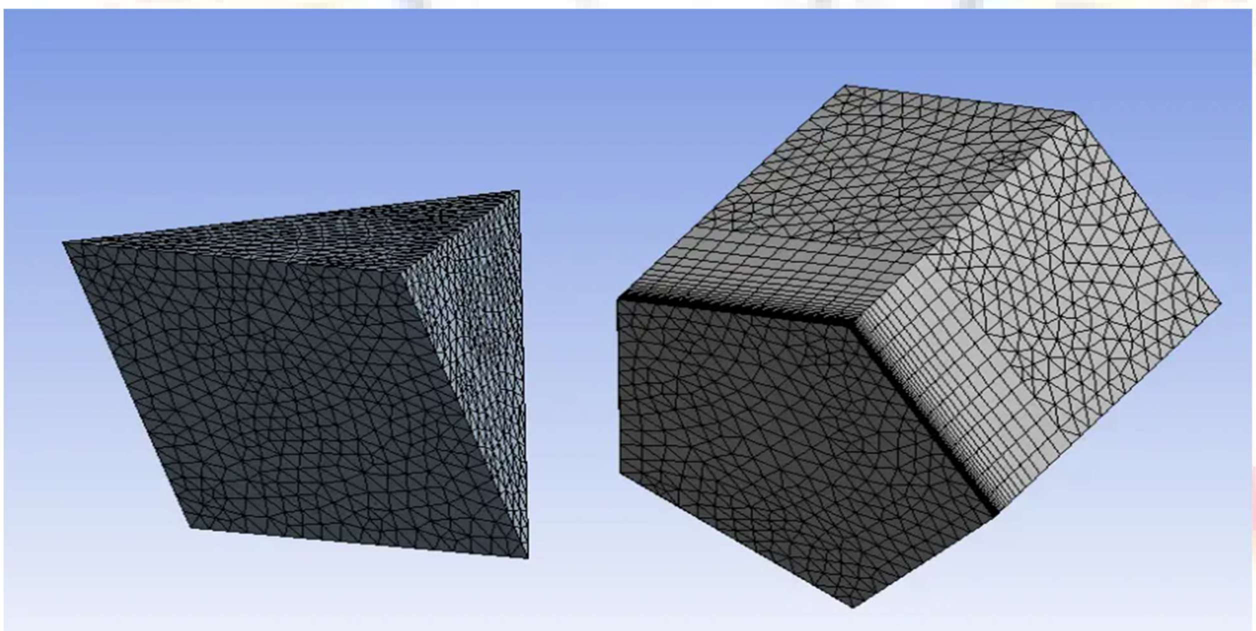
Figure: Inflation on faces of model

All Faces in Chosen Named Selection

If you select All Faces in Chosen Named Selection, then you can able to apply inflation on Face(s) on which name is assigned.



'All Faces in Chosen Named Selection' inflation option



'All Faces in Chosen Named Selection' inflation option

you can choose only one face at one time on which name is assigned.

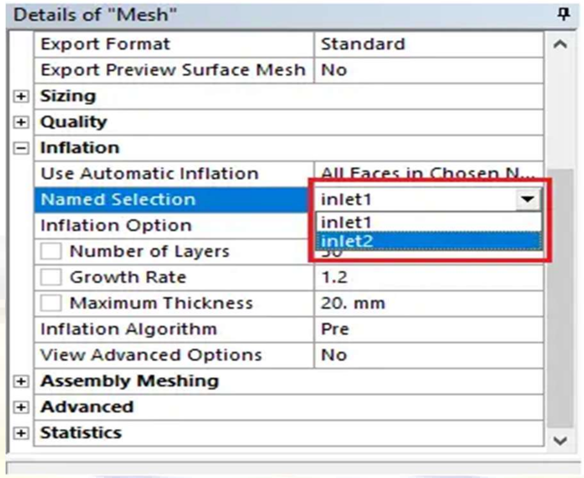


Figure: Name selection

The effect of inflation is applied to the Named Selections group is depends on values that you enter for the following options:

- Inflation Option
- Inflation Algorithm
- View Advanced Options

Inflation Option:

Total Thickness

The Total Thickness option creates inflation layers using the values of the Number of Layers, Growth Rate controls & Maximum thickness. By this option, we can create no. of inflation layers in particular thickness. for example, I want to create inflation layers in 10 mm thickness from selected face & no. of inflation layers is 5.

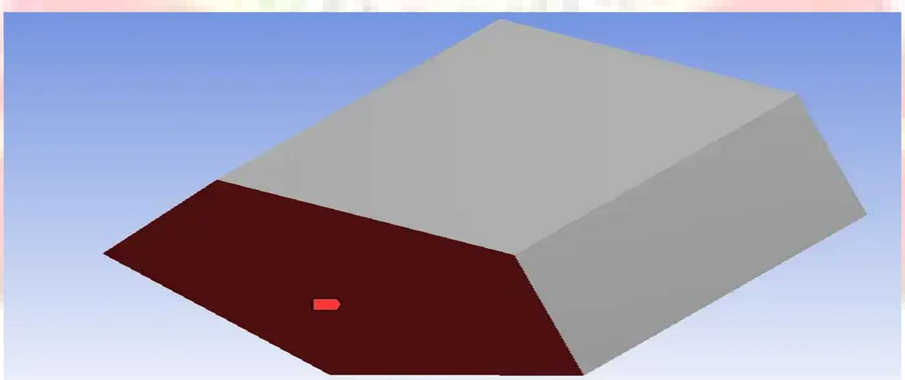


Figure: Geometry for Inflation

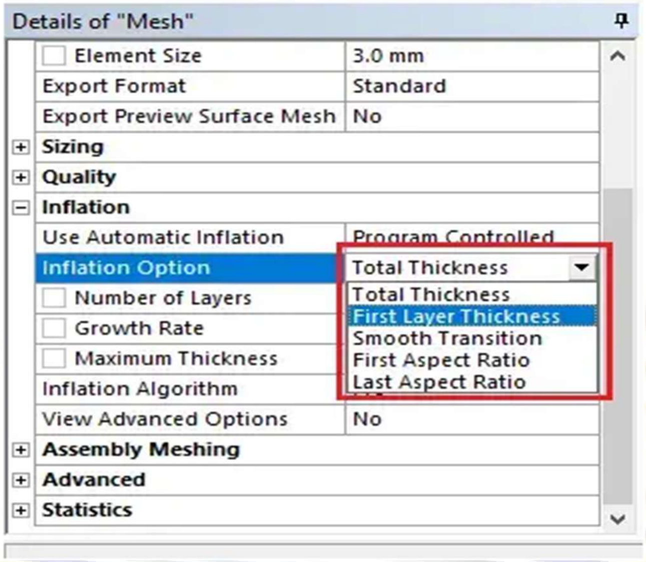


Figure: Inflation Details

So, i will enter Number of Layers = 5, & Maximum Thickness = 10 mm and the result is….

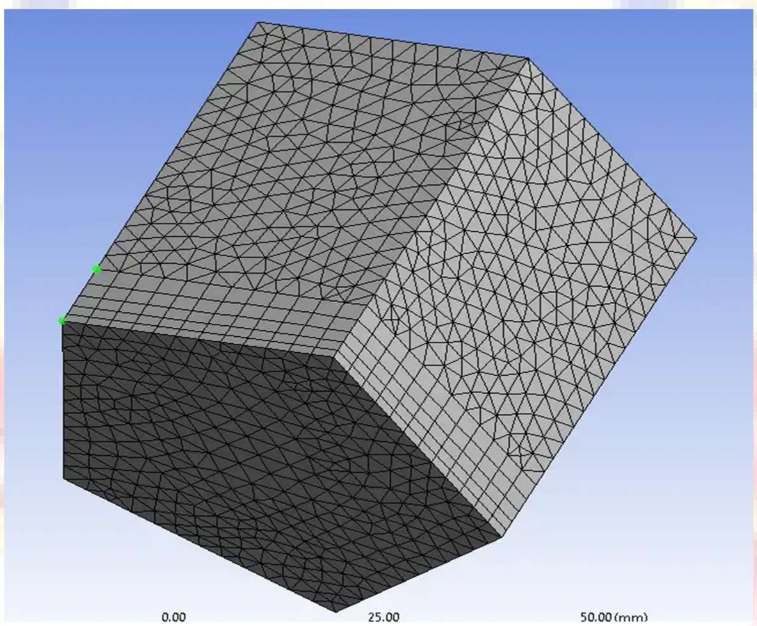
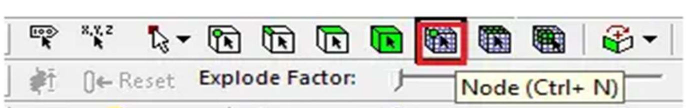


Figure: Geometry with 5 layers of Inflation

You can check also by measuring the distance between nodes. just click on Node option & select 2 nodes by press ctrl key as shown in the image.



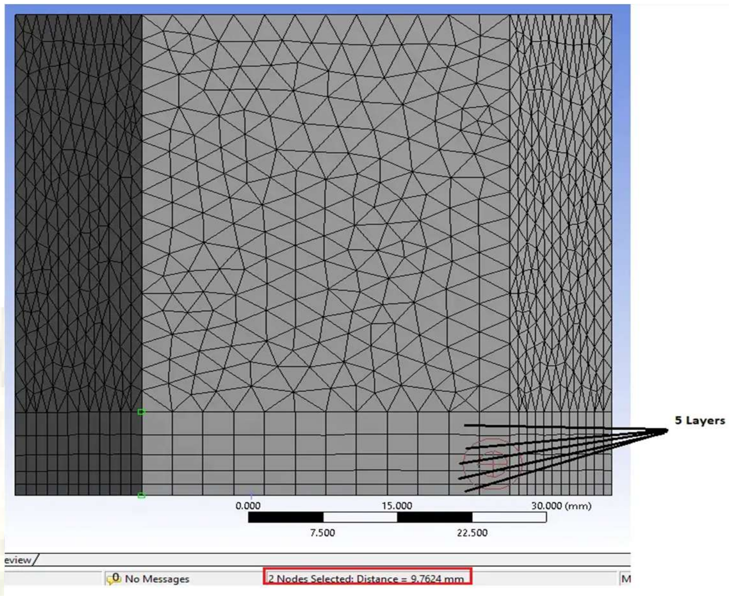


Figure: layers in Meshing

First Layer Thickness

This is the same as the total thickness option. The change is only, By this option, you can set the thickness of the 1st layer from the selected face. For example, I want to create a thickness of the 1st layer is 0.5 mm.

Inflation	
Use Automatic Inflation	All Faces in Chosen N...
Named Selection	inlet
Inflation Option	First Layer Thickness
First Layer Height	0.5 mm
Maximum Layers	5
Growth Rate	1.2
Inflation Algorithm	Pre
View Advanced Options	No
Assembly Meshing	
Advanced	
Statistics	

Figure: Inflation Options

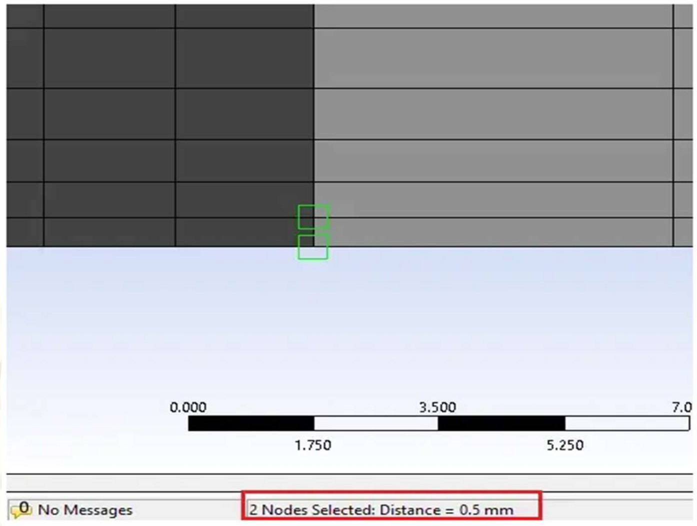


Figure: Meshing with layers

Smooth Transition

In this option, a new parameter appears called the 'Transition ratio'. The transition ratio is the ratio between the cell area of the last layer of inflation and the first cell area out of the inflation. The Transition Ratio determines the rate at which adjacent elements grow. As transition ratio increases, the gap between the adjacent layers is increased. The Transition Ratio control is applicable only when the Inflation Option is Smooth Transition. The transition Ratio is valid from 0 to 1. When Physics Preference is set to CFD and Solver Preference is set to CFX, the default for Transition Ratio is 0.77. For all other physics preferences, including CFD when Solver Preference is set to either Fluent or POLYFLOW, the default is 0.272.

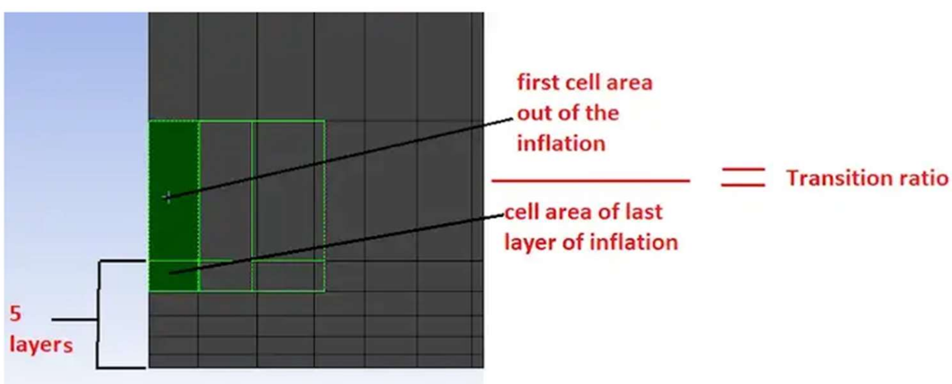


Figure: Transition ration in meshing

Inflation Algorithm

The Inflation Algorithm control determines which inflation algorithm will be used. There are two types of algorithm.

- Pre
- Post

Pre

- When Pre is selected, the surface mesh will be inflated first, and then the rest of the volume mesh will be generated. This is the default for all physics types.

Post

- A benefit of this option is that the tetrahedral mesh does not have to be generated each time the inflation options are changed.
- Post inflation is not supported when there is a mixture of the tetrahedron and non-tetrahedron mesh methods applied to the bodies which are made from multi-part. If you want to apply Post inflation to a multi-body part, all bodies in the part must have a tetrahedron mesh method applied to them.
- Post inflation is not supported when Inflation Option is either First Aspect Ratio or Last Aspect Ratio.

PROCEDURE

RESULT AND CONCLUSION:

S.NO	Mesh Metrics	Average Quality	Comment
1			
2			
3			
4			
5			

Table: Mesh Metric Results

Experiment No: - 4

INCOMPRESSIBLE INTERNAL LAMINAR FLOWS IN PIPE

Q) **An incompressible liquid is flowing though the cylindrical pipe with a consistent radius, having a diameter of 0.2 meters and a length of 3 meters, while accounting for a pipe thickness of 0.02 meters. The Inlet velocity of fluid is 2m/s and outlet pressure of 1 atm. solve the problem numerically using ANSYS FLUENT and show the following result.**

i) Pressure Contour

ii) Velocity Contour

iii) Velocity Stream Lines

Assumptions

Density of liquid = 1 kg/m3

Viscosity of liquid = 2e-5 kg/m-s

AIM

SOFTWARE REQUIREMENT

PROCEDURE

RESULT AND CONCLUSION:

S.NO	Mesh Metrics	Average Quality	Comment
1			
2			
3			
4			
5			

Table: Mesh Metric Results

Experiment No: - 5

INCOMPRESSIBLE EXTERNAL LAMINAR FLOW ON A FLAT PLATE

Q) **Engine oil at 40 ^{0} C flows over a 5m long flat plate with a free stream velocity of 2m/s. The density and kinematic viscosity of engine oil at 40 ^{0} C are**

Density = 870 kg/m3, Kinematic Viscosity = 2.485 x 10^{-4} m²/s

Simulate the flow over the flat plate. Determine the drag force acting on the plate per unit width.

AIM:

SOFTWARE REQUIREMENT

THEORY

Assumptions:

1. The flow is steady and incompressible.
2. The critical Reynolds number is Re = 5 × 105.

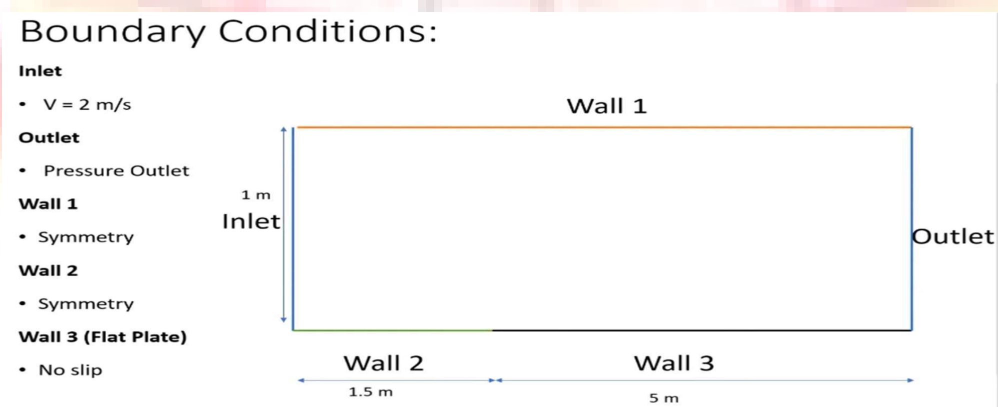
Properties:

The density and kinematic viscosity of engine oil at 40°C are = 876 kg/m³ and v = 2.485 × 10-4 m²/s.

Analysis

Noting that L = 5 m, the Reynolds number at the end of the plate is

Re = (2 m/s)(5 m) / 2.485 x 10^{-4} m²/s = 4.024 × 10^{-4} which is less than the critical Reynolds number. Thus we have laminar flow



Temperature:

- Skin Friction Coefficient =0.0071647433
- Drag Force =

Analytical solution:

$$C_f = 1.328 Re_L^{-0.5} = 1.328 \times (4.024 \times 10^4)^{-0.5} = 0.00662$$

Noting that the pressure drag is zero and thus $C_D = C_f$ for parallel flow over a flat plate, the drag force acting on the plate per unit width becomes

$$F_D = C_f A \frac{\rho V^2}{2} = 0.00662(5 \times 1\ m^2)\frac{(876\ kg/m^3)(2\ m/s)^2}{2}\left(\frac{1\ N}{1\ kg \cdot m/s^2}\right) = 58.0\ N$$

The total drag force acting on the entire plate can be determined by multiplying the value just obtained by the width of the plate.

All values calculated using simulation are within % error bar with analytical solution, Hence results can be considered.

PROCEDURE

RESULT AND CONCLUSION:

The incompressible external laminar flow over a flat plate was successfully simulated. The drag force acting on the plate during the flow was determined, providing insights into the flow behaviour and resistance experienced by the surface.

S.NO	Mesh Metrics	Average Quality	Comment
1			
2			
3			
4			
5			

Table: Mesh Metric Results

Experiment No: - 6

INCOMPRESSIBLE EXTERNAL TURBULENT FLOW ON FLAT PLATE

1 Q) **Simulate the flow of water over a 0.2x0.2x0.05 m flat plate with a velocity of 2 m/s for 2 seconds. Calculate the drag (force) on the plate within this time frame and analyze the variations in velocity and pressure along the flow.**

AIM

SOFTWARE REQUIREMENT

THEORY

Boundary layer is a layer adjacent to a surface where viscous effects are important.

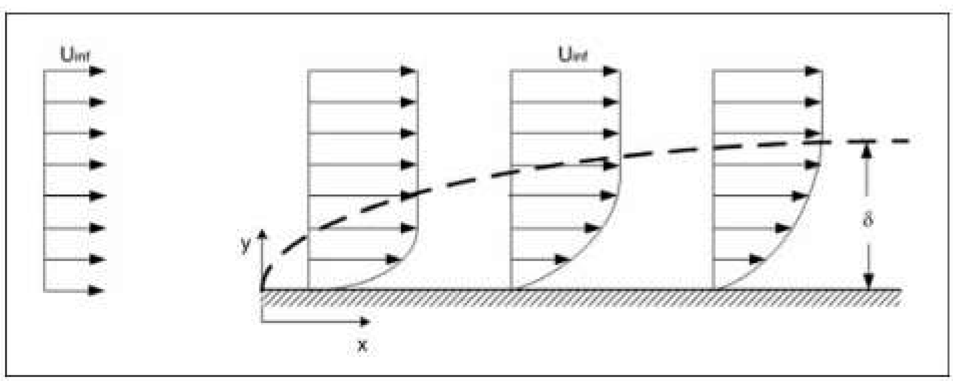


Figure 1: Flow over a flat plate

The fluid particles at the flat plate surface have zero velocity and they act as a retardant to reduce velocity of adjacent particles in the vertical direction. Similar actions continue by other particles until at the edge of the boundary layer where the particles' velocity is 99% of the free stream velocity. Boundary layers can also be measured by more significant parameters.

The Reynolds number is a measure of the ratio of inertiIONNa forces to viscous forces. It can be used to characterize flow characteristics aver a flat plate. Values under 500,000 are classified as Laminar flow where values from 500,000 to 1,000,000 are deemed Turbulent flow. Is it important to distinguish between turbulent and non-turbulent flow since the boundary layer thickness varies

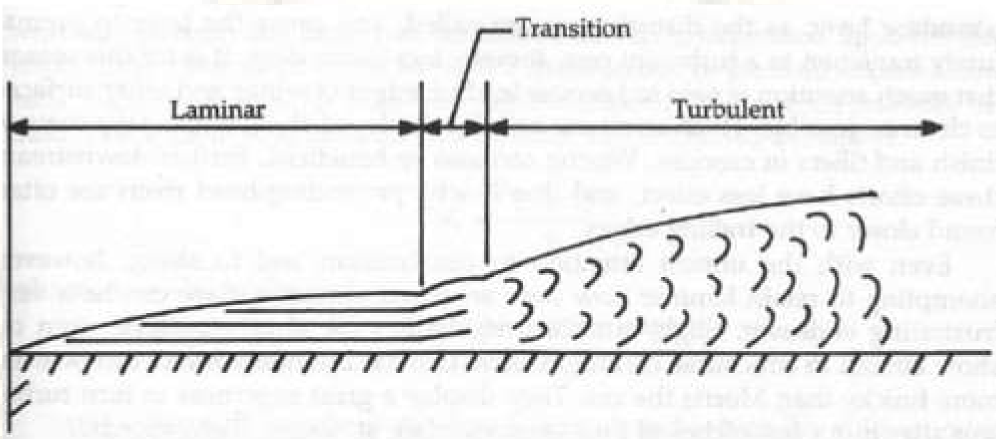


Figure 2: Flow over a flat plate

Drag coefficients, CD, for several bluff and streamline shapes are shown as a function of Reynolds number. The drag force,

$$F_D = C_D A(rU^2/2),$$

For the flat plate normal to the flow, C_D is independent of Reynolds number over the range shown because flow separation occurs at the sharp corners of the plate. When the plate is oriented parallel to the flow direction the drag coefficient is reduced by more than an order of magnitude and becomes Reynolds number dependent with transition and turbulence causing C_D to increase at higher Re.

PROCEDURE

RESULT AND CONCLUSION:

S.NO	Mesh Metrics	Average Quality	Comment
1			
2			
3			
4			
5			

Table: Mesh Metric Results

Experiment No: - 7

FLOW PAST A CYLINDER

1 Q) **Simulate the flow of water around a cylinder with a diameter of 0.2m, where the water is moving at a velocity of 0.001 m/s. Investigate the forces of drag and lift that arise due to this flow and also delve into the variations in velocity and pressure along the path of the flow.**

AIM

SOFTWARE REQUIREMENT

THEORY

The flow past a two-dimensional cylinder is one of the most studied of aerodynamics. It is relevant to many engineering applications. The flow pattern and the drag on a cylinder are functions of the Reynolds number $Re_D = UD/$□, based on the cylinder diameter D and the undisturbed free-stream velocity U. Recall that the Reynolds number represents the ratio of inertial to viscous forces in the flow. The drag is usually expressed as a coefficient $C_d = d/(\frac{1}{2}\rho U^2 D)$, where d is the drag force per unit span

PROCEDURE

RESULT AND CONCLUSION:

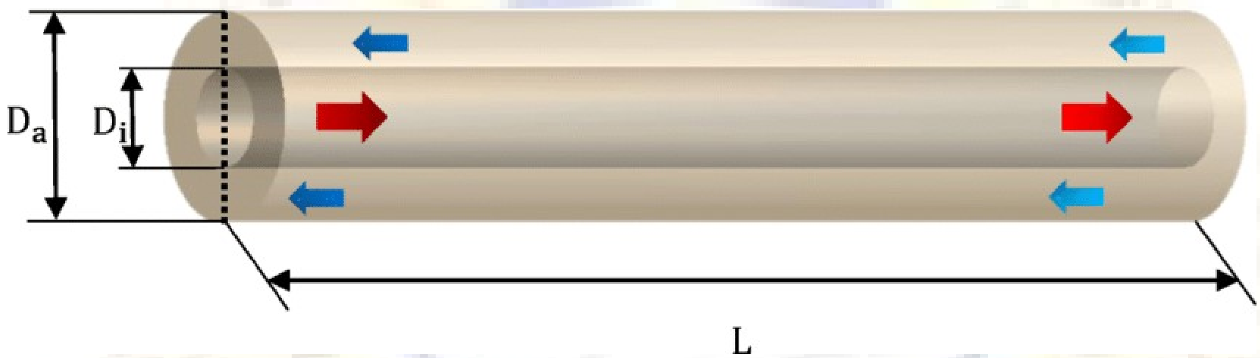
S.NO	Mesh Metrics	Average Quality	Comment
1			
2			
3			
4			
5			

Table: Mesh Metric Results

Experiment No: - 8

HEAT TRANSFER ANALYSIS OF HEAT EXCHANGER

1 Q) **A counter flow double pipe heat exchanger having a length of 1m. The inner pipe of the heat exchanger has an inner diameter of 40 mm and a wall thickness of 5mm whereas the outer pipe has an inner diameter of 70 mm and a wall thickness of 5mm. Water having a temperature of 80^0C enters the inner pipe with a velocity of 0.25 m/s and water having a temperature of 15^0C enters the outer region with a velocity of 0.3 m/s. Calculate the area-weighted average temperature of water at the exists. Both Pipes are made of Aluminium.**



AIM

SOFTWARE REQUIREMENT

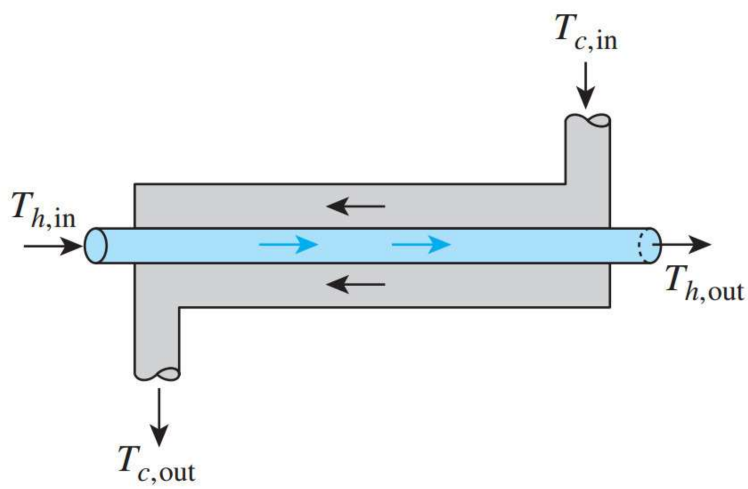
THEORY

Heat exchangers are devices that transfer or exchange heat between two fluids without mixing and include various types depending on the design, application, required space, and the fluid flows in the system. All the heat exchangers have a barrier that separates the fluids and allows the heat transfer simultaneously. The double pipe heat exchanger is one of the basic kinds of exchangers with a very flexible configuration. There are two types of counterflow or parallel flow for this type that are the basis of design and calculation for determining pipe size, length, and a number of bends.

Double Pipe Heat Exchangers

In double pipe heat exchangers, we have a large pipe with a small pipe inside it concentrically, and all the heat transfer process occurs inside the larger pipe. One fluid flows through the inner of a small pipe, and another fluid is between the two pipes, and that is how the inner pipe acts as a conductive barrier. The outside or shell side includes fluid flow passing on the inner side or tube side.

This type of heat exchanger is known as hairpin, jacketed pipe, jacketed u-tube, and pipe in pipe exchanger. They can contain one pipe or pipe bundle (less than 30), and the outer pipe must have a diameter of less than 200mm. In some cases, to increase the rate of heat transfer between working fluids, there are longitudinal fins in the inner tube.



PROCEDURE

RESULT AND CONCLUSION:

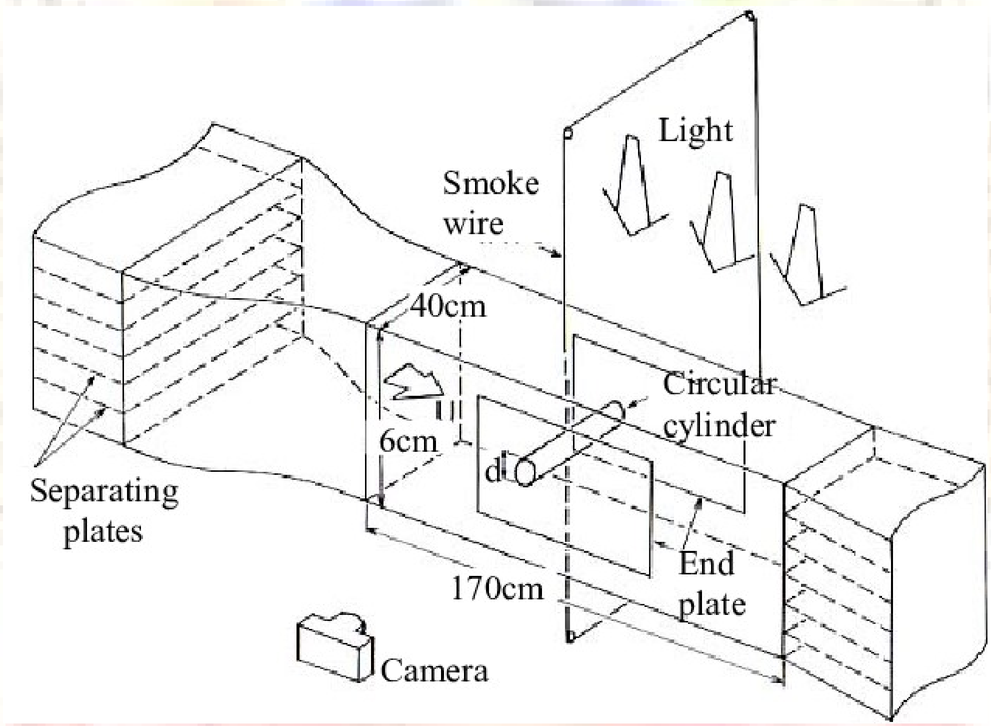
S.NO	Mesh Metrics	Average Quality	Comment
1			
2			
3			
4			
5			

Table: Mesh Metric Result

Experiment No: - 9

FORCED CONVECTION FLOWS

1 Q) **Conduct a simulation to analyze the temperature changes of air, which is initially at 15^0C moving with a velocity of 2m/s in a wind tunnel over a cylinder of diameter 0.2m. The Surface temperature of cylinder is 100^0 C. Explore how the air temperature varies along the flow path and find the exit temperature of air.**



AIM

SOFTWARE REQUIREMENT

THEORY

Forced Convection Heat Transfer

Convection is the mechanism of heat transfer through a fluid in the presence of bulk fluid motion. Convection is classified as natural (or free) and forced convection depending on how the fluid motion is initiated. In natural convection, any fluid motion is caused by natural means such as the buoyancy effect, i.e. the rise of warmer fluid and fall the cooler fluid. Whereas in forced convection, the fluid is forced to flow over a surface or in a tube by external means such as a pump or fan.

Mechanism of Forced Convection

Convection heat transfer is complicated since it involves fluid motion as well as heat conduction. The fluid motion enhances heat transfer (the higher the velocity the higher the heat transfer rate).

The rate of convection heat transfer is expressed by Newton's law of cooling:

$$\dot{q}_{conv} = h(T_s - T_\infty) \qquad (W/m^2)$$
$$\dot{Q}_{conv} = hA(T_s - T_\infty) \qquad (W)$$

The convective heat transfer coefficient *h* strongly depends on the fluid properties and *roughness* of the solid surface, and the type of the fluid flow (*laminar* or *turbulent*).

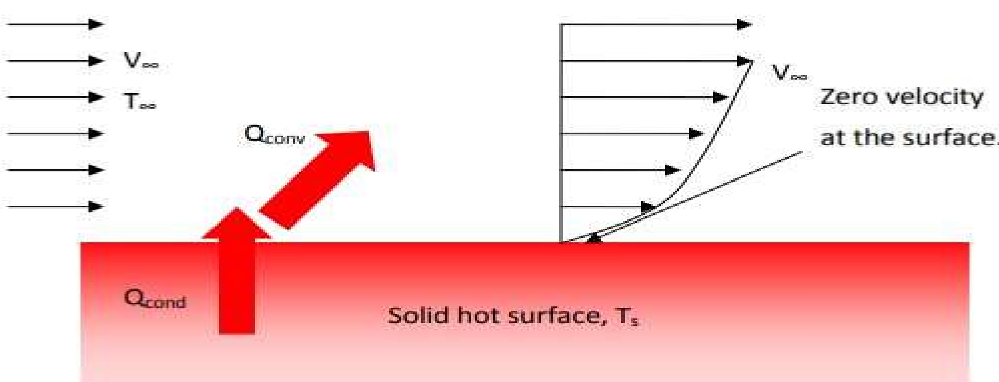


Fig. 1: Forced convection.

It is assumed that the velocity of the fluid is zero at the wall, this assumption is called no- slip condition. As a result, the heat transfer from the solid surface to the fluid layer adjacent to the surface is by pure conduction, since the fluid is motionless. Thus,

$$\left.\begin{aligned} \dot{q}_{conv} = \dot{q}_{cond} &= -k_{fluid} \left.\frac{\partial T}{\partial y}\right|_{y=0} \\ \dot{q}_{conv} &= h(T_s - T_\infty) \end{aligned}\right\} \rightarrow h = \frac{-k_{fluid} \left.\frac{\partial T}{\partial y}\right|_{y=0}}{T_s - T_\infty} \qquad (W/m^2.K)$$

The convection heat transfer coefficient, in general, varies along the flow direction. The mean or average convection heat transfer coefficient for a surface is determined by (properly) averaging the local heat transfer coefficient over the entire surface.

PROCEDURE

RESULT AND CONCLUSION:

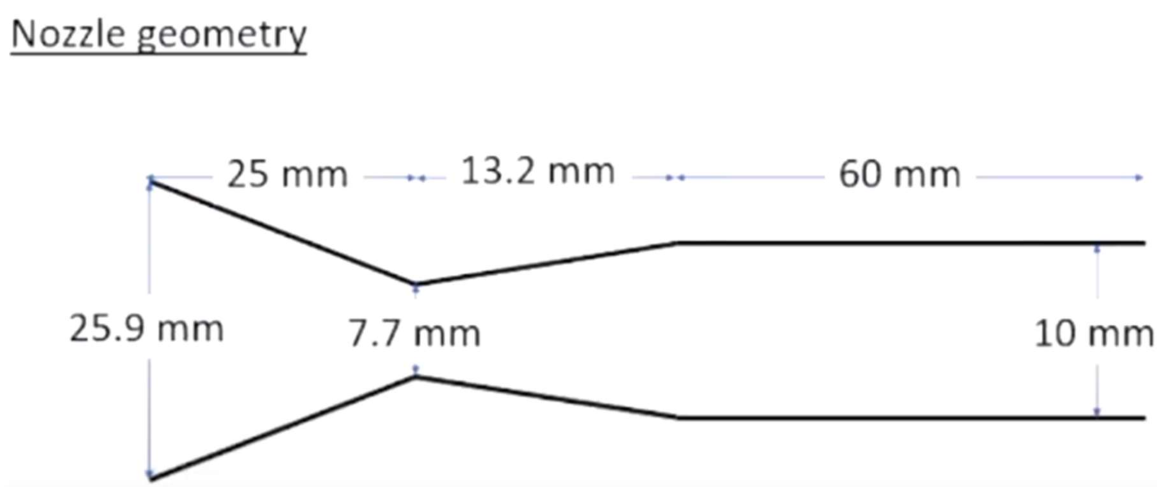
S.NO	Mesh Metrics	Average Quality	Comment
1			
2			
3			
4			
5			

Table: Mesh Metric Result

Experiment No: - 10

COMPRESSIBLE FLOW IN CONVERGENT-DIVERGENT NOZLE

1 Q) **Conduct a simulation to compressible flow in a convergent –divergent nozzle using ANSYS Fluent. The dimensions of the nozzle are given below.**



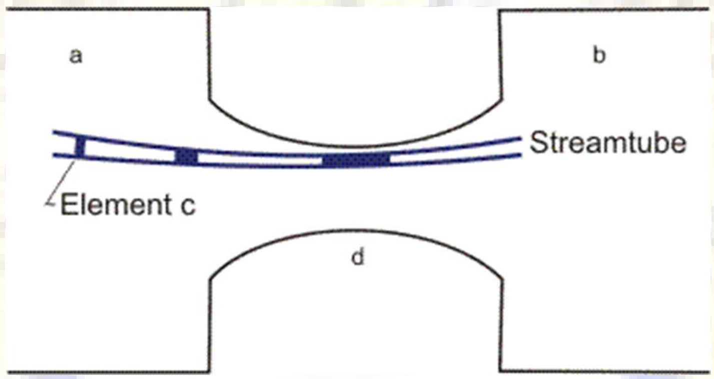
AIM

SOFTWARE REQUIREMENT

THEORY

All fluids are compressible and when subjected to a pressure field causing them to flow, the fluid will expand or be compressed to some degree. The acceleration of fluid elements in a given pressure gradient is a function of the fluid density, ρ, whereas the degree of compression is determined by the isentropic bulk modulus of compression, κ. The speed of sound in a medium is given by, a = (κ/ρ)1/2 and compressibility effects are apparent when the flow velocity, u, becomes significant compared to the local speed of sound. The local Mach number M = u/a is the primary parameter which characterizes the effects of compressibility. Under normal atmospheric conditions, the speed of sound in water is 1500 ms−1 and that in air is 345 ms−1. Thus, it can be expected that compressibility manifests itself in gas flows more readily than in liquid flows and the discussion below deals predominantly with gas flows. Transients in hydraulic systems are an example of compressible liquid flow which is of some importance. The case of liquid-gas mixtures is of interest and is discussed below.

The role of Mach number in compressible gas flow may be derived from the governing equations of motion and state. However, the physics of these processes are clear when gas flow from one chamber to another is considered. Flow from a constant pressure reservoir, a, is produced by reducing the pressure in chamber b below that in a (Figure 1).



PROCEDURE

RESULT AND CONCLUSION

.NO	Mesh Metrics	Average Quality	Comment
1			
2			
3			
4			
5			

Table: Mesh Metric Result

Experiment No: - 11

NATURAL CONVECTION FLOWS

1 Q) Simulate Natural Convection heat transfer for a solid Aluminium cylinder. The diameter of cylinder is 50 mm. The Cylinder has been kept in a rectangular chamber. The cylinder wall temperature is 500K, whereas the surrounding air temperature is 300K. These temperature difference causes a natural convection heat transfer between the cylinder surface and surrounding air. This heat transfer bas been analyzed for an unsteady (Transient) case for a time period of 60 seconds. The detail variations of temperature along with time has to been captured a solution.

AIM

SOFTWARE REQUIREMENT

THEORY

In natural convection, the fluid motion occurs by natural means such as buoyancy. Since the fluid velocity associated with natural convection is relatively low, the heat transfer coefficient encountered in natural convection is also low.

Mechanisms of Natural Convection

Consider a hot object exposed to cold air. The temperature of the outside of the object will drop (as a result of heat transfer with cold air), and the temperature of adjacent air to the object will rise. Consequently, the object is surrounded with a thin layer of warmer air and heat will be transferred from this layer to the outer layers of air.

The temperature of the air adjacent to the hot object is higher, thus its density is lower. As a result, the heated air rises. This movement is called the natural convection current. Note that in the absence of this movement, heat transfer would be by conduction only and its rate would be much lower

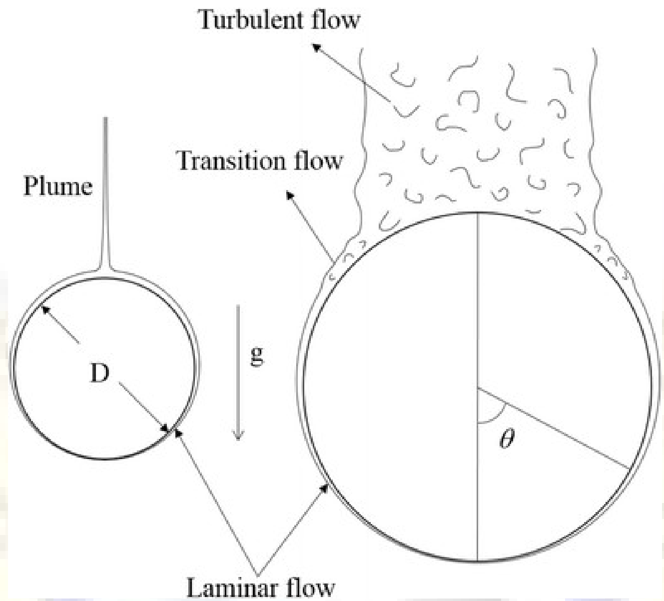


Figure: Natural Convection of flow over cylinder

PROCEDURE

RESULT AND CONCLUSION

COMPUTATIONAL FLUID DYNAMICS

LAB MANUAL

lendi

LIET-MED

DEPARTMENT OF MECHANICAL ENGINEERING

Do's:

- ✔ Follow the lab schedule and be punctual.
- ✔ Use only authorized software and tools for programming.
- ✔ Handle lab equipment and systems with care.
- ✔ Report any system or software issues to the lab supervisor.

Don'ts:

- ✘ Do not bring food, drinks, or USB drives without permission.
- ✘ Do not modify system settings or install unauthorized software.
- ✘ Do not leave the system on or workspace untidy after use..