

Flow Over External Compression Inlet

Nima Fariborzi*

University of California, San Diego, La Jolla, 92093, United States of America

Michael Puso†

University of California, San Diego, La Jolla, 92093, United States of America

Christopher Lin‡

University of California, San Diego, La Jolla, 92093, United States of America

I. Nomenclature

β	=	wave angle or shock angle
δ	=	turn angle or deflection angle
M	=	Mach Number
P	=	pressure
P_o	=	stagnation pressure or total pressure
ρ	=	Density
T	=	Temperature
e	=	internal energy
C_v	=	Constant volume specific heat
C_p	=	Constant pressure specific heat
μ	=	Dynamic viscosity
u	=	Velocity in x-direction
v	=	Velocity in y-direction
k	=	Thermal conductivity
q	=	Heat flux
τ	=	Shear stress
ξ	=	Grid transformation variable
η	=	Grid transformation variable
n	=	Time value
t	=	Time step
d	=	distance between two spacial grid points

II. Introduction

ONE of the key objectives in air travel is to minimize cost—not just in monetary cost of operation, but in reduced cost in time for those using air travel. Currently most civilian transport aircraft travel at around Mach 0.6-0.9, just on the border of subsonic and transonic flight, due to the increased cost of operation when entering the supersonic flight regime caused by the significant increase in drag once the sound barrier is passed. This occurs as particles in air cannot move faster than the speed of sound, leading to build up of air molecules called shock waves, producing what is called wave drag. This additional form of drag means that the vehicles must produce higher thrust values that scale exponentially with the speed, unlike with subsonic flight. This increases the fuel burn for each flight and therefore increases operational costs, a significant barrier for the development of these vehicles. A prime example of this conundrum was the Concorde, an aircraft developed in 1969 to travel Mach 2, significantly reducing commute times. The Concorde could complete what was usually a seven to eight hour flight from LA to NYC in an average of three and a half hours.[1] This aircraft, while providing high speed travel, was eventually grounded due to a variety of

*Masters Student, Mechanical and Aerospace Engineering, 9500 Gilman Dr, La Jolla, CA 92093

†Masters Student, Mechanical and Aerospace Engineering, 9500 Gilman Dr, La Jolla, CA 92093

‡Masters Student, Mechanical and Aerospace Engineering, 9500 Gilman Dr, La Jolla, CA 92093

factors. One of the most significant being that the operational cost out-weighed the price customers were willing to pay for the convenience of reduced flight times. Since the Concorde's discontinuation, demand has persisted to increase the efficiency of the propulsion system to increase the viability of civilian supersonic transport aircraft—also known as SST.

To reduce operational cost of the aircraft, we can increase the efficiency of the propulsion system to decrease fuel burn through optimization. Most of the upcoming designs for SST's use a gas turbine engine, an engine similar to those in current commercial aircraft, with modifications for supersonic flight. These engines require high pressure slow moving air—around Mach 0.3—for ingestion, and the higher pressure the air is going into the engine, the more efficiently it can produce thrust.[2] To increase the pressure the engine receives, we can optimize the inlet design to create the most optimal conditions for the aircraft engine. Since this has previously been done by Nima Fariborzi, we know the best ramp angles to maximize pressure recovery across the inlet system.

This paper will discuss the flow of an external compression inlet for supersonic vehicles using gas turbine engines for SST's. These engines require inlets that slow the flow of air to subsonic conditions; however, this must be done while preserving the total pressure from ingestion to the end of the inlet. This can be done using one or more oblique shocks, followed by a normal shock. This system is called an external compression inlet and is shown in Fig.1 b) along with the internal compression inlet and mixed compression inlet.

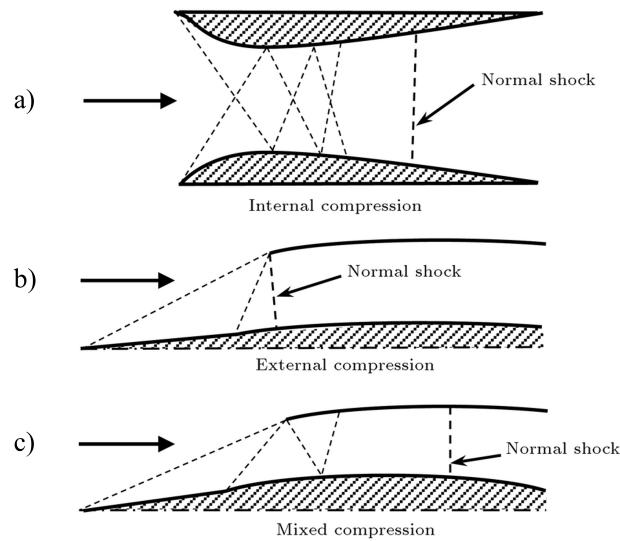


Fig. 1 Types of supersonic inlets.[3]

Using the external compression inlet we can use one or two oblique shocks as well as a normal shock to make the flow subsonic, as well as decrease total pressure loss from the start to the end of the inlet for the engine. The reason why oblique shocks are utilized in inlet design is that the losses are dependent on the normal component of the flow to the shock. The experienced loss is similar to a weaker normal shock that would typically form at a much lower mach number, minimizing the overall total pressure loss when compared to one strong normal shock. The oblique shocks are created with shock ramps and the normal shock is created from the back pressure from the engine face after the diffuser section. The section after the inlet takes high subsonic air and slows it down using changing areas so that it is low velocity high pressure air for the engine to ingest.

Our goal is to visualize the optimized ramp system previously found to show the pressure recovery across the shocks. We will solve the compressible Navier Stokes equations(NSE) using the MacCormack method to march in time while also using a grid transformation with block structuring to efficiently partition the domain according to the areas of highest importance, such as near the boundaries and shocks.

A. Problem Geometry

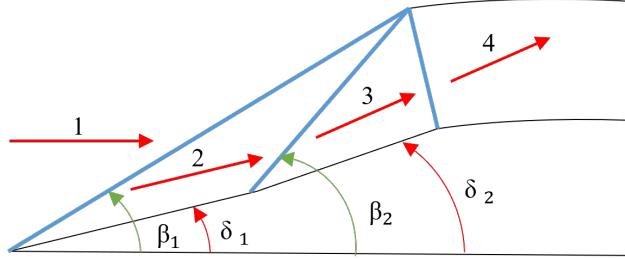


Fig. 2 Geometry of the physical domain

To fully specify the dimensions of the external compression inlet, the angles δ_1 , δ_2 , β_1 , and β_2 must be specified. For the case where free-stream Mach number is 3, the following angles were found to give optimal pressure recovery:

δ_1	14.98°
δ_2	18.81°
β_1	32.22°
β_2	45.15°

Table 1 Ramp angles for $M = 3$

With the specification of the above angles, the lengths of each ramp and the location of the leading edge of the cowl can be determined. Moreover, with these values we calculate with compressible flow theory that the stagnation pressure ratio across the inlet should be 92.1%.

III. Governing Equations

We first assume our air to be Newtonian and a calorically perfect gas. With this in mind, we will have eight unknowns to solve for: density, x- and y-velocity, pressure, temperature, energy, viscosity, and thermal conductivity. Four of those unknowns will be solved through the NSE leaving four unknowns found by the equations of state for perfect gas where R is air,

$$P = \rho RT \quad (1)$$

the equation for a calorically perfect gas,

$$e = C_v T \quad (2)$$

Sutherland's Law, where $\mu_0 = 1.735 * 10^{-5}$, $T_0 = 288K$ and $S_1 = 110.4K$,

$$\mu = \mu_0 \left(\frac{T}{T_0} \right)^{\frac{3}{2}} \frac{T_0 + S_1}{T + S_1} \quad (3)$$

and the specification of thermal conductivity k , where constant Prandtl number Pr and C_p is for air.

$$k = \frac{C_p}{Pr * \mu} \quad (4)$$

A. Navier-Stokes Equations

The general compressible Navier-Stokes equation in strong conservation form is:

$$\frac{\partial U}{\partial t} + \frac{\partial E}{\partial x} + \frac{\partial F}{\partial y} = 0 \quad (5)$$

where U is

$$U = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ E_t \end{bmatrix} \quad (6)$$

and E_t is defined as

$$E_t = \rho(e + \frac{u^2 + v^2}{2}) \quad (7)$$

The E vector flux is written as,

$$E = \begin{bmatrix} \rho u \\ \rho u^2 + P - \tau_{xx} \\ \rho uv - \tau_{xy} \\ (E_t + P)u - u\tau_{xx} - v\tau_{xy} + q_x \end{bmatrix} \quad (8)$$

The F vector flux is written as,

$$F = \begin{bmatrix} \rho v \\ \rho uv - \tau_{xy} \\ \rho v^2 + P - \tau_{yy} \\ (E_t + P)v - u\tau_{yy} - u\tau_{xy} + q_y \end{bmatrix} \quad (9)$$

where the shear stresses are,

$$\begin{bmatrix} \tau_{xy} = \tau_{yx} = \mu(\frac{du}{dy} + \frac{dv}{dx}) \\ \tau_{xx} = 2\mu(\frac{du}{dx} - \frac{1}{3}(\frac{du}{dx} + \frac{dv}{dy})) \\ \tau_{yy} = 2\mu(\frac{dv}{dy} - \frac{1}{3}(\frac{du}{dx} + \frac{dv}{dy})) \end{bmatrix} \quad (10)$$

and the heat fluxes q

$$\begin{bmatrix} q_x = -k \frac{dT}{dx} \\ q_y = -k \frac{dT}{dy} \end{bmatrix} \quad (11)$$

B. Navier-Stokes Equations on Transformed Grid

For our simulation, we will employ a grid transformation to locally refine the grid in regions of interest. In order to solve the problem with grid transformations, we will formulate a physical grid in the (x,y) domain that represents the real geometry of the problem as well as a computational grid in the (ξ,η) domain where the solution will actually be computed. The precise nature of these two grids will be discussed in more detail below; however, the governing equations must be slightly modified to account for the grid transformations. This requires the calculation of grid metrics that allow us to convert between the computational and physical domain. These are calculated using finite differences on the computational grid; the so-called inverse metrics are calculated as follows:

$$x_\xi = \frac{\partial x}{\partial \xi} \quad (12a)$$

$$y_\xi = \frac{\partial y}{\partial \xi} \quad (12b)$$

$$x_\eta = \frac{\partial x}{\partial \eta} \quad (12c)$$

$$y_\eta = \frac{\partial y}{\partial \eta} \quad (12d)$$

The Jacobian determinant at each grid point can then be calculated as:

$$J = x_\xi y_\eta - x_\eta y_\xi \quad (13)$$

Then, the conversion metrics can be calculated like so:

$$\xi_x = \frac{y\eta}{J} \quad (14a)$$

$$\eta_x = -\frac{y\xi}{J} \quad (14b)$$

$$\xi_y = -\frac{x\eta}{J} \quad (14c)$$

$$\eta_y = \frac{x\xi}{J} \quad (14d)$$

With these quantities known at each grid point in the computational domain, we can proceed to write the transformed compressible Navier-Stokes equation in strong conservation form:

$$\frac{\partial \mathcal{U}}{\partial t} + \frac{\partial \mathcal{E}}{\partial \xi} + \frac{\partial \mathcal{F}}{\partial \eta} = 0 \quad (15)$$

where

$$\mathcal{U} = JU \quad (16)$$

The transformed E flux vector becomes

$$\mathcal{E} = \xi_x JE + \xi_y JF = y\eta E - x\eta F \quad (17)$$

and the F flux vector becomes

$$\mathcal{F} = \eta_x JE + \eta_y JF = -y\xi E + x\xi F \quad (18)$$

Some terms in E and F , such as τ_{xy} , τ_{xx} and q_x contain $\frac{d}{dx}$ and $\frac{d}{dy}$. In order to do the grid transformation, we also need to transform the derivative to $\frac{d}{d\xi}$ and $\frac{d}{d\eta}$ respectively.

We use the following equations:

$$\begin{bmatrix} \frac{d}{dx} = \xi_x \frac{d}{d\xi} + \eta_x \frac{d}{d\eta} \\ \frac{d}{dy} = \xi_y \frac{d}{d\xi} + \eta_y \frac{d}{d\eta} \end{bmatrix} \quad (19)$$

For example, the $\frac{du}{dx}$ and $\frac{dv}{dy}$ in the τ_{xx} term would be replaced with

$$\frac{du}{dx} = \xi_x \frac{du}{d\xi} + \eta_x \frac{du}{d\eta} \quad (20)$$

and

$$\frac{dv}{dy} = \xi_y \frac{dv}{d\xi} + \eta_y \frac{dv}{d\eta} \quad (21)$$

yielding:

$$\tau_{xx} = 2\mu \left[\left(\xi_x \frac{du}{d\xi} + \eta_x \frac{du}{d\eta} \right) - \frac{1}{3} \left(\left(\xi_x \frac{du}{d\xi} + \eta_x \frac{du}{d\eta} \right) + \left(\xi_y \frac{dv}{d\xi} + \eta_y \frac{dv}{d\eta} \right) \right) \right] \quad (22)$$

IV. Numerical Methods

A. MacCormack's Method

In order to solve the compressible Naiver Stokes equations, we will need a method that will be both accurate and computationally fast. One such method which is well-suited for the solution of the NSE is MacCormack's method. We are applying this method because it solves all finite difference's in 1st order, but retains spatial second order accuracy due to the two-step nature, and is also second order accurate in time. The process to compute this method follows two steps.

First, we use the explicit Euler in time and applying the forward finite difference in x to get the so called, "predictor" which is a temporary solution.

$$\bar{U}_{i,j}^{n+1} = U_{i,j}^n - \frac{\Delta t}{\Delta x} (E_{i+1,j}^n - E_{i,j}^n) - \frac{\Delta t}{\Delta y} (F_{i,j+1}^n - F_{i,j}^n) \quad (23)$$

Second, implicit Euler in time is used to take a half-step from $t^{n+\frac{1}{2}}$ alongside backwards finite differences in x, using the values from the predictor to now solve the "corrector" value (which allows the step to be solved explicitly).

$$U_{i,j}^{n+1} = \frac{1}{2} [U_{i,j}^n + \bar{U}_{i,j}^{n+1} - \frac{\Delta t}{\Delta x} (\bar{E}_{i,j}^{n+1} - \bar{E}_{i-1,j}^{n+1}) - \frac{\Delta t}{\Delta y} (\bar{F}_{i,j}^{n+1} - \bar{F}_{i,j-1}^{n+1})] \quad (24)$$

The zig-zag like structure of this method from going forward and backward allows this code to be a second order accurate solution while only having to compute first order finite difference methods. It should be noted that the changes needed to make MacCormack's method work with a transformed grid are trivial. They are given below for completeness:

$$\bar{U}_{i,j}^{n+1} = U_{i,j}^n - \frac{\Delta t}{\Delta \xi} (\mathcal{E}_{i+1,j}^n - \mathcal{E}_{i,j}^n) - \frac{\Delta t}{\Delta \eta} (\mathcal{F}_{i,j+1}^n - \mathcal{F}_{i,j}^n) \quad (25)$$

$$U_{i,j}^{n+1} = \frac{1}{2} [\bar{U}_{i,j}^{n+1} + \bar{U}_{i,j}^{n+1} - \frac{\Delta t}{\Delta \xi} (\bar{\mathcal{E}}_{i,j}^{n+1} - \bar{\mathcal{E}}_{i-1,j}^{n+1}) - \frac{\Delta t}{\Delta \eta} (\bar{\mathcal{F}}_{i,j}^{n+1} - \bar{\mathcal{F}}_{i,j-1}^{n+1})] \quad (26)$$

B. Grid Transformations and Block Structuring

To improve the accuracy of the solution, we will apply a grid transformation so that the grid can be refined in regions with large spatial gradients, particularly in the boundary layers. The computational grid is constructed to have uniform grid spacing so that the computation of finite differences is straightforward. The physical domain, however, is constructed to conform to the physical geometry of the problem. Block structuring was used to further enable the use of different types of grid stretching in different regions of the domain, allowing us, to the maximum extent possible with a rectilinear grid, to refine the grid only where we need to. However, because the use of block structuring requires that grid points line up between adjacent blocks (unless hanging nodes are specially treated), it will be the case that regions of refinement propagate to the edges of the domain. The figure below portrays the transformed physical grid and the respective uniformly-spaced computational grid. Note that the grids used for the actual computations were much finer.

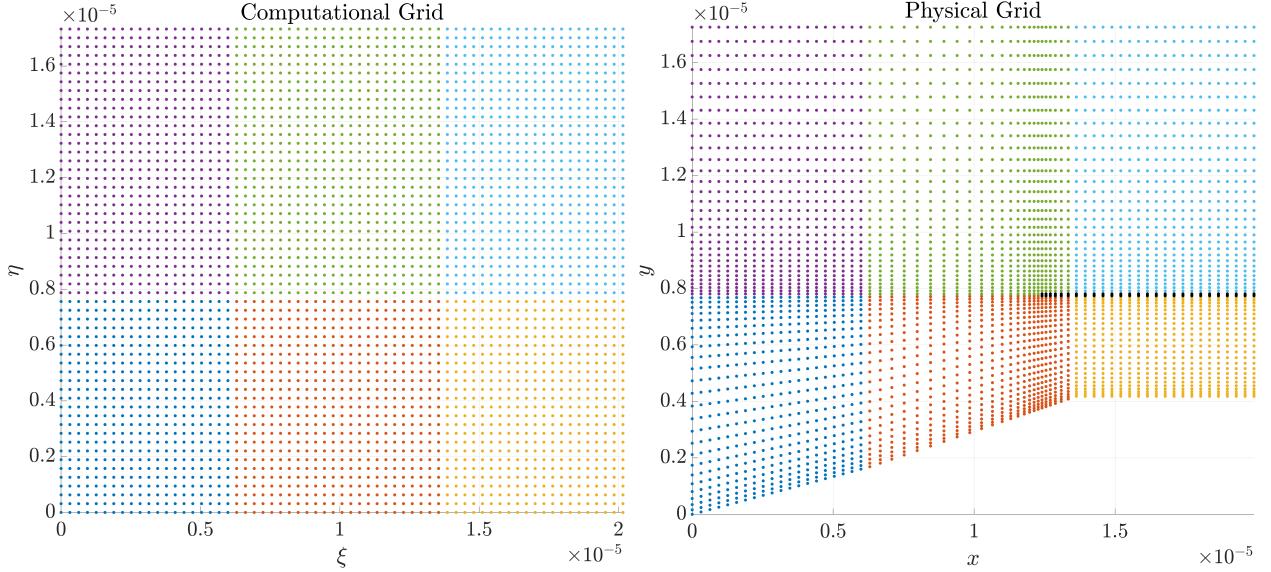


Fig. 3 Computational grid and Physical grid

The domain consists of six blocks, each with a unique color in the figure above. Blocks 1-3 are those nearest the bottom of the domain, from left to right, and blocks 4-6 are those above blocks 1-3, also from left to right. The grid transformations were applied to refine the grid near the no-slip boundaries: along the inlet ramp, above and below the cowl, and near the leading edge of the cowl. In order to refine the grid within the boundary layer both along the ramp and on the underside of the cowl, two-sided biasing was used in the y-direction in block 3. For compatibility, the same was applied to blocks 1 and 2, though this also accomplished the refining of the grid within the boundary layer on the ramps. Similarly, one-sided biasing was used in blocks 4 through 6 to give refinement of the boundary layer on the top

side of the cowl. Finally, a grid transformation was applied in the x -direction in blocks 2 and 5 to give refinement near the leading edge of the cowl.

Black points are those which will be treated as the cowl; the cowl must be two points thick in order to accommodate a difference in pressure above and below. It should also be noted that on the cowl surface, finite difference derivatives in the η direction were always calculated via backward difference on the bottom surface and via forward difference on the upper surface. This prevented the solution on one side of the cowl from influencing the solution on the other side.

C. Boundary Conditions

The boundary conditions are nearly identical to those used in the midterm project in which we treated supersonic flow over a flat plate. However, as noted above, we are using $M = 3$ for our case. The left edge of the domain was treated as the inlet, and the top edge of the domain was treated as the far-field; in both of these regions, u was specified to give $M = 3$, v was set to zero, and P and T were set to their freestream values (taken as STP in our case). The right side of the domain was treated as the outlet, and extrapolation boundary conditions were used for all four variables here. Finally, the ramp and cowl were treated as walls where the no-slip condition was applied ($u = v = 0$). The temperature on the walls could either be specified or treated as adiabatic. The pressure on the walls is the only non-trivial boundary condition due to the problem geometry. Extrapolation was used to prescribe pressure on the walls, however, the extrapolation must be in the normal direction; along the ramps, there are not grid points in the normal direction from a given point on the ramp. Instead, a combination of interpolation and extrapolation was used as shown in the following figure:

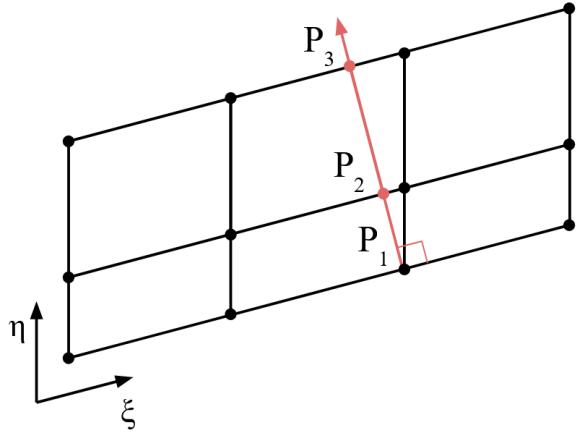


Fig. 4 Pressure boundary condition extrapolation

To compute the pressure P_1 using the extrapolation boundary condition, the steps are as follows. First, a ray is cast in the direction normal from the wall, shown as a red arrow in the above figure. The location where this ray intersects with the first two grid lines along ξ is calculated. Pressure at these positions, P_2 and P_3 , were calculated via interpolation between the nearest two neighbors along ξ . Then, P_1 could be calculated from the following extrapolation formula:

$$P_1 = P_2 - \frac{d_{1,2}}{d_{2,3}} (P_3 - P_2) \quad (27)$$

where $d_{1,2}$ and $d_{2,3}$ are the distances between point 1 and 2 and point 2 and 3, respectively. Pressure extrapolation along non-angled surfaces (such as on the bottom wall after the ramp section, and above and below the cowl) were treated normally.

V. Results

Solutions were generated on two grids, denoted as "coarse" and "fine." The coarse grid had dimensions of 129 points by 111 points, and the fine grid had dimensions of 257 by 221 points.

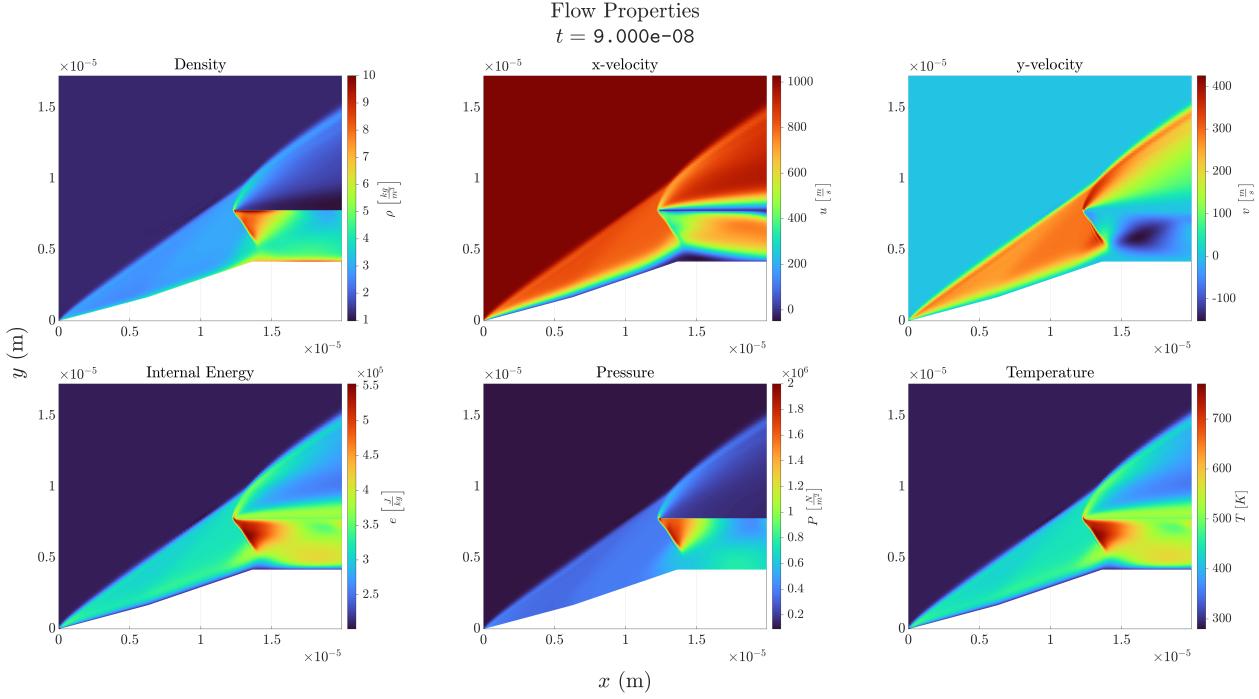


Fig. 5 Flow results on coarse mesh

Figure 5 shows the resulting solution fields calculated by our solver on the coarse grid. Results on the fine grid will be shown in higher detail below. They are generally as expected in terms of qualitative characteristics: two oblique shocks can be seen resulting from the two distinct ramp angles (the second is very faint due to the small difference in angle between the first and second ramp, but it is visible). We can also see shocks formed by the cowl. An oblique shock is formed above, which merges with the shock formed by the ramps. The analytical results show a normal shock formed by the cowl which sits inside the intake. Our results show a shock which is not quite normal. Reasons for this are discussed below.

Moreover, the stagnation pressure is plotted in Figure 6 based on the results on the fine grid, and it can be noted that stagnation pressure drops as it goes through the shocks, as expected. From the analytical results, we expect a stagnation pressure ratio of 92.1% from the freestream conditions to after the normal shock in the intake. Calculating the mass-averaged stagnation pressure at the end of the intake section in our results, we find a stagnation pressure ratio of 46.3% compared to the freestream value on the fine grid (44.4% on the coarse grid).

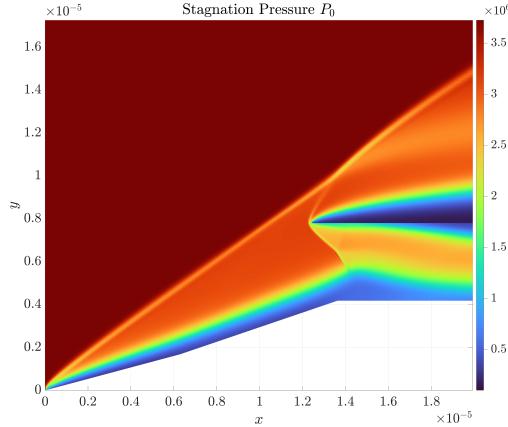


Fig. 6 Resulting stagnation pressure on the fine grid

Larger plots of the solution on the fine grid are shown below for the variables of interest.

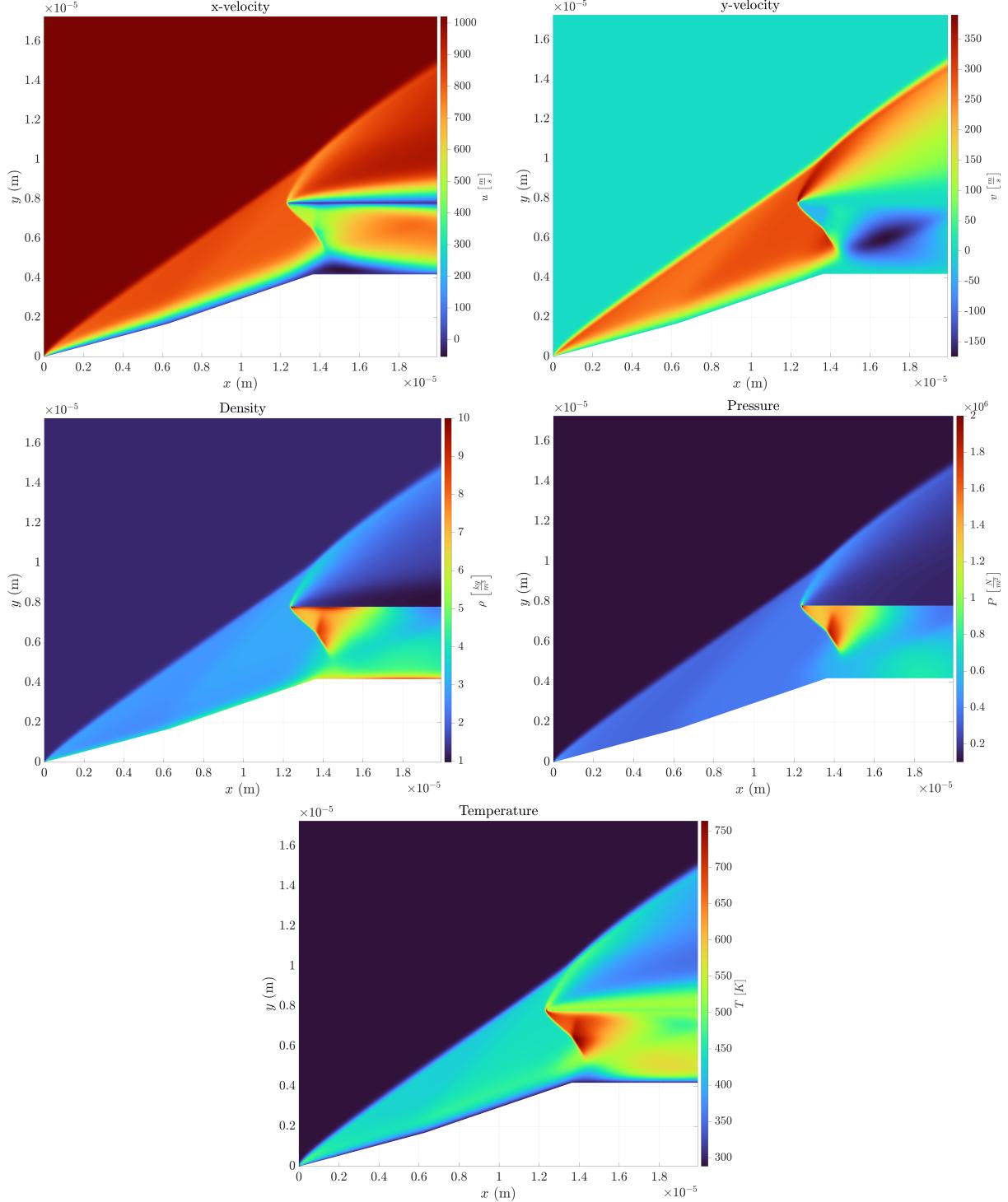


Fig. 7 Results on the fine grid

Grid independence was not truly verified for this problem, as we were only able to generate results on two different grids. Grids coarser than the coarse grid would not run without crashing the solver, and grids finer than the fine grid were untenable to run due to the computational time required. To claim grid independence we would have liked to run

one more case with an even finer grid, though we would require more computational resources (or a more efficient implementation in, say, C++ or FORTRAN). However, given that the difference in results between the coarse and fine grid are fairly minimal, we are confident that further refinement of the grid would only verify the current results.

VI. Discussion

A. Analysis of Results

As noted above, we do not observe a true normal shock at the location of the cowl lip. In the real intake, back pressure is generated further within the engine that propagates upstream to create this shock. This could have been simulated in our code by prescribing the pressure at the outlet. Further, the cowl should be curved such that the flow at the leading edge is parallel to the cowl. In our analysis, a flat cowl was used for ease of implementation, and this likely impacted the resulting shock structure.

Additionally, the stagnation pressure recovery was not as much as expected. This is most likely due to viscous losses in our simulation. The analytical analysis assumes inviscid flow, which is a valid assumption for supersonic flow at the physical scale. In our case, the DNS solver does not work well at scales above those shown in this report, with a total domain length of just under 20 microns. At this scale, the boundary layer makes up a significant portion of the flow entering the intake, and as seen in Figure 6, the boundary layer has very low stagnation pressure.

B. Future considerations

Several measures can be taken to improve the overall numerical accuracy and allow the simulation to resemble more to that of a real external compression inlet.

First, the normal shock in the inlet can be better simulated by implementing a curved cowl lip instead of just a flat surface, this would allow the normal shock to resemble more to that of Figure 1. Furthermore, more appropriate Boundary conditions can be enforced exclusively inside the inlet to simulate the back pressure, which would in turn generate the correct normal shock at the inlet.

Second, the numerical accuracy of the two oblique shocks can be improved greatly if grid refinements were implemented along the shock, i.e. doing grid transformation within the shock's vicinity. In addition, instead of using finite difference, unstructured finite volume method can be applied to create the grid. This would greatly increase the flexibility of the design of the grid and allow the grid refinements aforementioned.

Finally, by implementing the RANS or LES method, simulations in scales that are similar to the real-life physical scale can be better achieved. Some complications that originate from the current small scale domain can then be resolved. For example, the boundary layers would account for a much smaller portion of the flow going into the intake, and the measured pressure recovery would be closer to the analytical value.

Acknowledgements

We would like to thank Professor Schmidt for his advice about applying the concepts he taught during the course to the current analysis.

References

- [1] “How Fast Did Concorde Actually Fly From New York To London?”, 04 2022. URL <https://www.grc.nasa.gov/www/k-12/airplane/normal.html>.
- [2] Raymer, D. P., *Aircraft design: A conceptual approach*, 6th Ed, American Institute of Aeronautics and Astronautics, 2018.
- [3] Ebrahimi, A., and Chavoshi, M., “Numerical Investigation of Back Pressure and Free-stream Effects on a Mixed Compression Inlet Performance,” *Scientia Iranica*, Vol. 25, 2017. <https://doi.org/10.24200/sci.2017.4324>.