AIAA SciTech Forum
11–15 & 19–21 January 2021, VIRTUAL EVENT
AIAA Scitech 2021 Forum

10.2514/6.2021-0386

Optimization of an Axisymmetric Mach 6 Reflected-Shock Tunnel Nozzle at Flight Enthalpy

A. Hameed * M. A. Mustafa † D. Shekhtman ‡ N. J. Parziale §
Stevens Institute of Technology, Hoboken, NJ 07030, USA

In this work, we present an optimization process for the design of converging-diverging nozzle contours. The process is a brute-force algorithm that runs CFD simulations for various contours until the one with the most uniform flow properties is obtained. The contour is described using a Bezier curve, with control points serving as the independent variables in the optimization process. The function that is minimized is a penalty function that characterizes the non-uniformity of the flow properties. The optimization is carried out in three steps: contour definition, inviscid optimization, and viscous optimization. The inviscid optimization is performed with a coarse grid and a CFD model with no viscosity or turbulence. This serves to rapidly design a contour that is close to the desired performance. Using the result of the inviscid optimization as a starting point, further refinement is carried out in the viscous optimization, with a finer grid and a CFD model that accounts for all relevant physical phenomena. Different CFD solvers are used at each step, and results are presented for the design of a Mach 6 nozzle for the Stevens Shock Tunnel.

Nomenclature

M = Mach number, (-)
P = Static pressure, (Pa)
P_{pitot} = Pitot pressure, (Pa)
T = Temperature, (K)
ρ = Density, (kgm^{-3})
u = Streamwise velocity, (ms^{-1})
v = Wall-normal velocity, (ms^{-1})
x = Streamwise coordinate, (m)
r = Radial coordinate, (m)
F = Penalty function, (-)
θ = Flow angle, (degrees)
f_M = Exit Mach number penalty, (-)
f_θ = Exit flow angle number penalty, (-)
f_{M_c} = Centerline Mach number penalty, (-)
f_p = Contour penalty, (-)
f_P = Exit pressure penalty, (-)

I. Introduction

The rapidly advancing field of hypersonics has given rise to the development of high-speed research facilities such as shock tubes, shock tunnels, wind tunnels, and Ludwieg tubes. The purpose of these facilities is to recreate the conditions experienced by hypersonic vehicles. A common feature of all such facilities is a

*Graduate Student, Mechanical Engineering, Castle Point on Hudson, Hoboken, New Jersey, 07030.
†Postdoctoral Researcher, Mechanical Engineering, Castle Point on Hudson, Hoboken, New Jersey, 07030.
‡Graduate Student, Mechanical Engineering, Castle Point on Hudson, Hoboken, New Jersey, 07030.
§Associate Professor, Mechanical Engineering, Castle Point on Hudson, Hoboken, New Jersey, 07030, AIAA Senior Member.

converging-diverging nozzle that accelerates the flow to high velocities. The design of converging-diverging nozzles in these research facilities is driven by the need for accurate measurements, requiring the flow to be uniform, unidirectional, and shock-free.

A common method used to design the diverging section of a supersonic nozzle is the Method of Characteristics (MOC). It is computationally inexpensive and there have been various improvements to the method, such as incorporating vibrational effects.[1] However, the MOC requires an iterative design process as viscous effects are not properly accounted for. An accurate CFD simulation is used to check the MOC results and a corresponding boundary-layer correction is applied to the MOC for the next iteration. Therefore, the MOC is not a replacement for solving the complete set of flow equations using an appropriate solver. Although it is a useful tool when computing power is not readily available (as was the case when it was first developed) this is no longer a limiting factor with even commercially available modern personal computers able to perform the simulations described later in this work.

In this work, a methodology is presented to design nozzle contours without using the MOC. Building upon work from the University of Queensland,[2] it is a brute force algorithm that evaluates numerous contours by solving for the flow each time using a CFD solver with all the relevant flow physics accounted for. For each contour, a penalty function is calculated that quantifies the non-uniform nature of the flow. The contour with the smallest value of the penalty function is selected as the optimal contour. This is effectively an optimization problem, where the contour is the independent variable and the penalty function is being minimized. This design methodology is comprised of three steps, outlined below.

1. Contour Definition: Choose a suitable mathematical definition for the contour of the nozzle. In this work, we use a Bezier curve which defines the contour by a set of control points. These control points are adjusted during the algorithm to change the contour until their optimal values are determined.
2. Inviscid Optimization: Using a CFD package, perform the optimization assuming the flow is inviscid. The target Mach number here is slightly higher than the desired value, to account for the decrease in expansion caused by the boundary layers on the wind-tunnel walls. An inviscid solver runs much faster, and the purpose of this step is to get a contour that is close to the desired conditions.
3. Viscous Optimization: Using the results of step 2 as a starting point and a suitable CFD package, perform the optimization with viscous and turbulent effects included. These simulations take longer than their inviscid counterparts, and any other physical effects (thermochemical nonequilibrium, etc.) that need to be accounted for must be added. The target Mach number is the desired value as this simulation represents the actual conditions. The accuracy of this algorithm depends on the CFD model used in this step.

The steps are detailed in the following sections and results are presented for the design of a Mach 6 nozzle for the Stevens Shock Tunnel. The Stevens Shock Tunnel is currently being constructed to advance our understanding of hypersonic fluid mechanics phenomena such as boundary-layer instability,[3–5] turbulence,[6] and shock-wave boundary-layer interaction[7,8] via non-intrusive optical diagnostics.[9–17]

II. Characterization of Nozzle Contour

The first step in the design process is selecting a suitable mathematical description for the supersonic portion of the nozzle contour. Following Chan et al.,[2] in this work, we use a Bezier curve to define the axial, x, and radial, r, coordinates of the contour. The general form for the nozzle contour using an n^{th} order Bezier curve is given in Eq. 1,

$$\begin{aligned} x(t) &= \sum_{i=0}^{i=n} \binom{n}{i} (1-t)^{n-i} t^i X_i \\ r(t) &= \sum_{i=0}^{i=n} \binom{n}{i} (1-t)^{n-i} t^i R_i \end{aligned} \tag{1}$$

where, $0 \leq t \leq 1$, $\binom{n}{i}$ is the binomial term and X_i and R_i are the control points. The order of the curve, n,

is fixed at the beginning and the control points, (X_i, R_i), are the parameters that are varied until an optimal contour is achieved. As the form shows, the number of control points is always $n+1$.

Bezier curves have the property that they always pass through the first and last control points, (X_0, R_0) and (X_n, R_n), respectively. We exploit this property to limit the nozzle's size to meet any spatial constraints imposed by a new or pre-existing set up and adhering to required start-up times. The nozzle length can be fixed by setting $X_0 = 0$ and X_n to the design length. Similarly, R_n can be set to the desired value to fix the nozzle's exit radius. Therefore, X_0 has a fixed value of 0 and denotes the throat location. Furthermore, the last control point, (X_n, R_n) is fixed and is a constraint that is imposed on the optimization problem.

An example of a nozzle contour defined using a Bezier curve is given in Fig. 1. This is the optimized curve that was obtained for the design of a 1.5 m long 16 in exit diameter Mach 6 nozzle. The Bezier curve (in black) in Fig. 1 is a 9^{th} order curve and its 10 control points are given in Table 1.

The subsonic portion of the nozzle contour is not subject to the optimization problem as there are no standing waves in that section. Hence, its selection is somewhat ad-hoc. In this work, we chose a 2 in radius quarter circle to ensure that the slope at the throat is 0.

In addition to fixing the values of X_0, X_n, and R_n, there are other constraints that the contour must satisfy. First, at the throat, the slope must be 0. This is achieved by enforcing that $R_0 = R_1$ (as shown in Table 1). Second, at the nozzle exit, it might be of interest to ensure that the contour is nearly flat so that the flow is fully axial. This is achieved by enforcing either $R_n = R_{n-1}$ or $R_n - R_{n-1} \leq tol$, where tol is some tolerance selected by the designer.

Consequently, in this work, there were a total of 10 control points that could be varied to change the nozzle profile. This corresponds to 20 variables (each control point has an X and R coordinate). Out of these 20, 3 are fixed (X_0, X_n and R_n) by design requirements, and 1 more, R_1 is fixed to ensure the slope is 0 at the throat. Therefore, the optimization problem is reduced from 20 to 16 variables. This could have been reduced further to 15 if $R_n = R_{n-1}$ had been enforced.

Table 1: Control points for 9^{th} order Bezier curve.

i	X_i (m)	R_i (m)
0	0.0000	0.0256
1	0.0704	0.0256
2	0.1007	0.0945
3	0.3872	0.1458
4	0.5775	0.1695
5	0.7707	0.1691
6	1.0207	0.2105
7	1.2601	0.1892
8	1.4451	0.2028
9	1.5000	0.2032

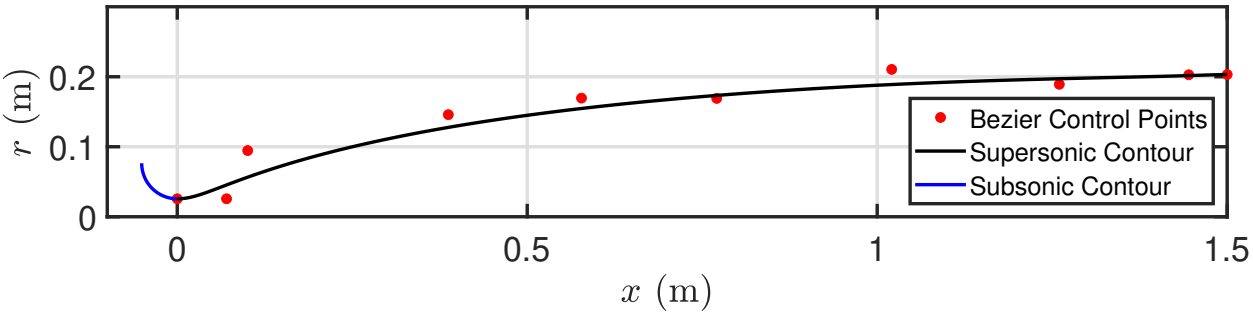


Figure 1: Paramaterization of nozzle contour. The Bezier control points are shown in red. The corresponding optimized Bezier curve, which defines the supersonic portion of the nozzle, is shown in black. The subsonic portion is shown in blue, which is defined by a 2 in quarter circle.

III. Algorithm Overview

In this section, we present the general structure of the algorithm. A schematic is shown in Fig. 2. The general form of the algorithm is the same for both the inviscid and viscous optimization processes. The differences lie in the CFD models. The steps are detailed below.

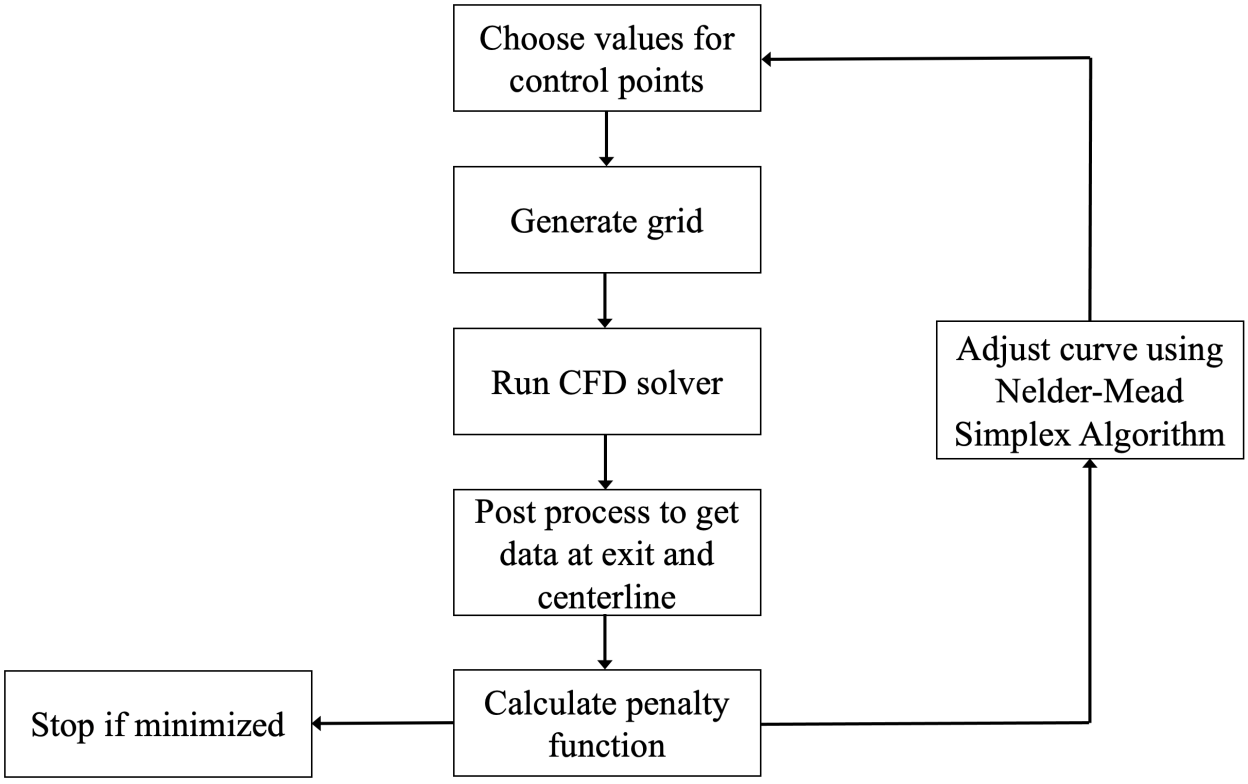


Figure 2: Schematic of the general structure of the algorithm.

1. Choose values for control points: To start the optimizer, an initial Bezier curve is used. This initial curve is selected by taking an existing nozzle, scaling it to the desired length and exit radius, and fitting a Bezier curve to its contour. The control points of this contour are then used to initiate the

optimization process. As the algorithm runs, the optimizer varies the control points to get the best results.

2. Generate grid: Once the contour has been generated, a corresponding grid is created for the solver. The grid includes both the converging and diverging sections and is different for the inviscid and viscous optimizations. In the inviscid case, the grid is coarse and not clustered, as the point is to speed through the calculations. The viscous version is refined enough to result in a converged solution and is clustered near the walls and throat.

3. Run CFD solver: After the grid is generated, the chosen CFD solver is run to compute the flow. As is done in this work, one may choose to use different solvers for the inviscid and viscous cases to speed up the process. In the inviscid case, the no-slip condition is not enforced at the wall. At the exit of the nozzle, a supersonic outlet boundary condition is enforced. At the nozzle's inlet, constant pressure/temperature reservoir conditions are enforced; the velocity is allowed to vary, as this is not a supersonic inlet.

4. Post process to get data at the nozzle's exit and centerline: Once the flow solution is available, it is processed to extract the data at the nozzle exit and the centerline. The exit data is used to determine the uniformity of the conditions and the centerline data is used to get an appropriate core size. In the viscous case, all of the data at the exit cannot be used as the boundary layer will always introduce non-uniformities. Therefore, only the data points in the freestream must be extracted and used to determine the quality of the flow.

5. Calculate penalty function: After extracting the data, the penalty function is computed. The value of this function describes the off-design nature of the nozzle, and the optimizer's goal is to minimize this value. Various functions can be designed, depending on how many flow properties (pressure, velocity, etc.) one wants to control.

6. Adjust curve using Nelder-Mead Simplex Algorithm: This is the optimization step. Following Chan et al.,[2] the algorithm used is a Nelder-Mead Simplex Algorithm, which varies the control points until the penalty function is minimized. After every iteration, the process loops back to step 1 in this list. The convergence of the minimization depends on how off-design the starting curve is. A starting curve that is extremely off-design will naturally require more iterations to reach its minimum.

7. Stop if minimized: If the penalty function has reached its minimum value or is within the prescribed tolerance limit, the optimizer stops and outputs the corresponding values of the control points that define the Bezier curve. This curve is then selected as the nozzle contour.

The penalty function, F, is evaluated as follows.

$$F = (f_\theta + f_M + f_{M_c} + f_P + f_p)^2 \tag{2}$$

Where,

$$\begin{aligned}
f_\theta &= \frac{1}{N\tan(\theta_{tol})^2}\sum_{i=1}^{N}(v_i/u_i)^2 \\
f_M &= \frac{1}{NM_{tol}^2}\sum_{i=1}^{N}(M_i - M_{tar})^2 \\
f_{M_c} &= \frac{1}{NM_{c,tol}^2}\sum_{j=1}^{N}(M_{c,j} - M_{tar})^2 \\
f_P &= \mathrm{PRESS}(\max(P_i) - \min(P_i)) \\
f_p &= 1e5, \quad \text{if}\frac{\partial r}{\partial x} < 0 \\
&= 0, \quad \text{otherwise}
\end{aligned} \tag{3}$$

Here, f_θ quantifies the angularity of the flow at the exit and f_M quantifies the deviation from the target Mach number, M_{tar}. v and u are the radial and axial velocities respectively and the summations for f_θ and f_M are carried out over the data points at the nozzle exit, N being the total number of points. θ_{tol} and M_{tol} are the weighting parameters. f_P quantifies the non-uniformity of the pressure at the exit, where the max and min operations are applied to all the data points at the exit and PRESS is the weighting parameter. f_p is a flag that returns a large number if the algorithm outputs a contour that curves inward (after the converging portion). f_{M_c} quantifies the deviation of the centerline Mach number from the target value. This summation is applied to the data points at the centerline of the nozzle from a chosen x location onward. It cannot be applied to the entire centerline as the Mach number has to start from ≈0 at the reservoir and then increase, therefore the centerline Mach number cannot be the target value throughout. N for f_{M_c} is then the number of data points from the chosen x location till the nozzle end and $M_{c,tol}$ is the weighting parameter. The values of the weighting parameters are chosen to ensure that all the penalties are approximately equal in value and that a single penalty is not given preference.

IV. Inviscid Optimization

In this section, the details of the inviscid optimization process are presented. The goal of the inviscid optimization is to create a contour that is close to the end result to minimize the run-time of the viscous optimization. The target Mach number for the inviscid optimization must be sightly higher than the desired value to account for the lack of a boundary layer. In this work, as the desired Mach number at the nozzle exit was 6, the target Mach number for the inviscid optimization was selected to be 6.2. The CFD package used for this step (meshing and solving) is Eilmer 3,[18] from the University of Queensland. During the inviscid optimization, both X_n and R_n coordinates of the control points are varied, resulting in 16 variables for the optimizer, as mentioned in the previous section.

The first step in the inviscid optimization is to create a starting Bezier curve. This was done by using the contour from an existing nozzle and scaling it to the desired length and exit radius. Then a Bezier curve was fit to the contour. The order of the Bezier curve in this work is 9, as this fits the existing nozzle well. Generally, Bezier curves of orders 6-9 work well for a variety of nozzle sizes; the choice mainly depends on which order fits the contour of the existing nozzle after it has been scaled to the desired size. The Borges-Pastva algorithm [a] was used to fit an initial Bezier curve to the existing nozzle contour, scaled to size. Following this initial curve, the subsequent curves were generated by the Nelder-Mead Simplex Algorithm.

The grid in the inviscid optimization was kept very coarse, roughly 70 x 30 in the axial and radial directions, respectively, as shown in Fig. 3. The grid was also specified to be orthogonal to the nozzle wall and included both the converging and diverging sections. The converging section was a quarter circle with a 2 in radius.

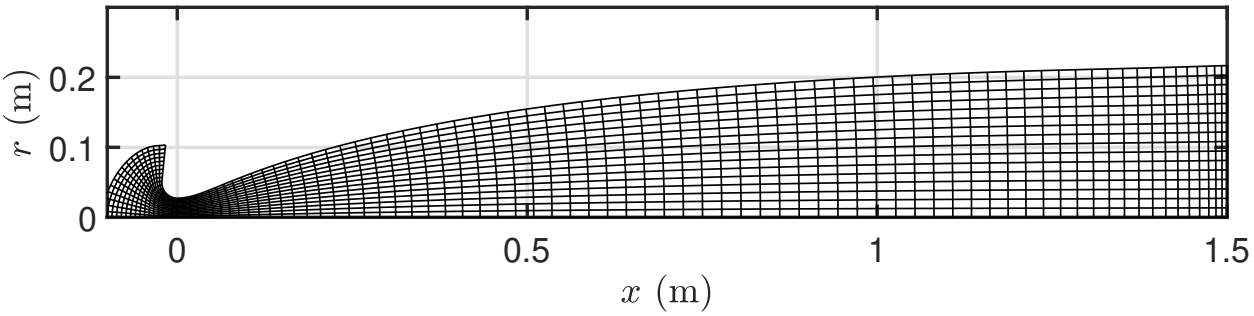


Figure 3: Grid for inviscid optimization process.

The CFD solver for the inviscid case was run with the desired reservoir conditions. These conditions appear in the boundary condition on the left boundary of the grid in Fig. 3. The proper boundary condition at this location is a subsonic inlet or constant pressure/temperature reservoir. The boundary condition at the nozzle exit is a supersonic outlet. At the top wall, the no penetration condition is enforced. The bottom wall, $r = 0$, is the axisymmetric line. In this work, the reservoir pressure and temperature were 1 MPa and 1800 K respectively, and the gas was air. The total simulation time was selected to be 3 ms, as that was

[a] https://www.mathworks.com/matlabcentral/fileexchange/46406-borgespastva-m

sufficient for the start-up process. The simulation was run in parallel on 16 cores and took around 30 seconds to finish.

The post-processing step of the inviscid optimization extracted the raw data from the nozzle exit and centerline and saved it in two separate files.

Fig. 4 shows how the penalty function changes as the optimization proceeds. The value changes by approximately 6 orders of magnitudes over the entire process and the CFD simulation requires approximately 1000 iterations to converge to a minimum. This process takes ≈6-8 hrs, depending on how off-design the starting contour is. Once the optimizer has converged, the resulting Bezier curve is used as the starting point for the viscous optimization, detailed in the next section.

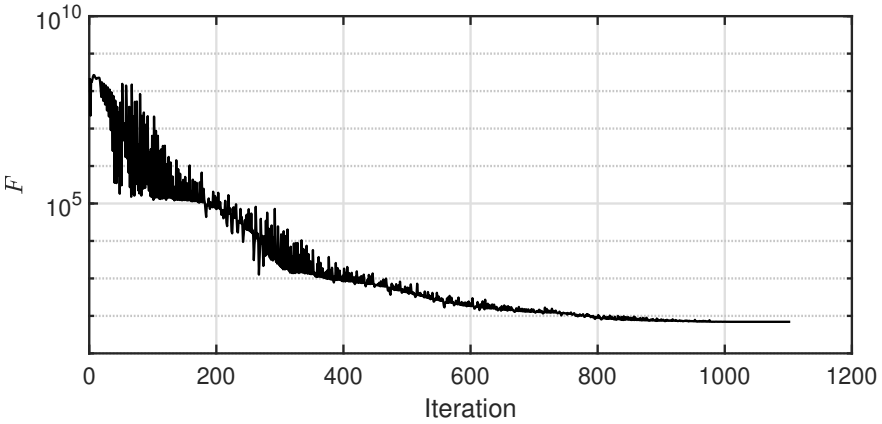


Figure 4: Variation of penalty function, F, with algorithm iteration for the inviscid optimization process.

V. Viscous Optimization

In this section, the viscous optimization process is detailed. During this step, all relevant physical phenomena must be accounted for, such as chemical reactions, nonequilibrium effects, etc. In this work, two CFD packages were used. Eilmer 3 was used to generate the grid, and a modification of DPLR[19–22] was used to solve the flow equations. DPLR was chosen over Eilmer 3 because it allows for large CFL numbers that reduce the simulation time significantly. The target Mach number for this step was now the desired Mach number, 6.

The starting Bezier curve for the viscous optimization is the optimized Bezier curve from the inviscid optimization. Furthermore, at this point, only the R_n coordinates of the control points are varied and the X_n coordinates are kept fixed at the starting values. This speeds up the process as fewer variables require fewer iterations and it is generally also not necessary to vary the X_n coordinates. Therefore, the number of variables in the viscous optimization was reduced to 8.

The grid for the viscous optimization is shown in Fig. 5. Unlike its inviscid counterpart, it is refined enough to be converged, 500x100 in the axial and radial directions, respectively. There is also clustering near the throat and walls to resolve the large gradients in the flow properties. It is also kept orthogonal to the nozzle walls. The grid was generated in Eilmer 3. However, the output from Eilmer 3 had to be converted into a format compatible with DPLR. This was done by adding ghost nodes at the boundaries and writing the file in a specific format.

The CFD simulation for the viscous optimization is run with viscosity, turbulence, chemical kinetics, and vibrational effects accounted for. The turbulence model used is the one-equation Spalart-Allmaras model[23] with the Catrisa and Aupoix[24] compressibility correction. The gas model is 5 species air with an excluded volume equation of state and a simple harmonic oscillator model to evaluate the vibrational energy. The boundary conditions are the same as the inviscid optimization, except that at the top wall the no-slip condition is enforced and the wall temperature is kept fixed at 300 K. The CFL number was progressively increased during the simulation until it reached a value of 1000. This ramp-up of the CFL number allowed the simulation to run in 10 min.

The post-processor for the viscous optimization is similar to the inviscid version in that it extracts the data at the exit and centerline.

The penalty function is similar to the one used in the inviscid optimization. The difference between the penalty functions is that the summations for f_θ, f_M, and f_P are carried out only in the freestream of the nozzle exit during the viscous optimization. The freestream is quantified as the region beyond which $\partial M/\partial r < 20$.[2]

Fig. 6 shows the variation of the penalty function as the optimization progresses. The final value is approximately 4 orders of magnitude less than the starting value. Also, the number of iterations is significantly less than the inviscid version, 180 vs. 1200. This takes ≈30 hrs to converge.

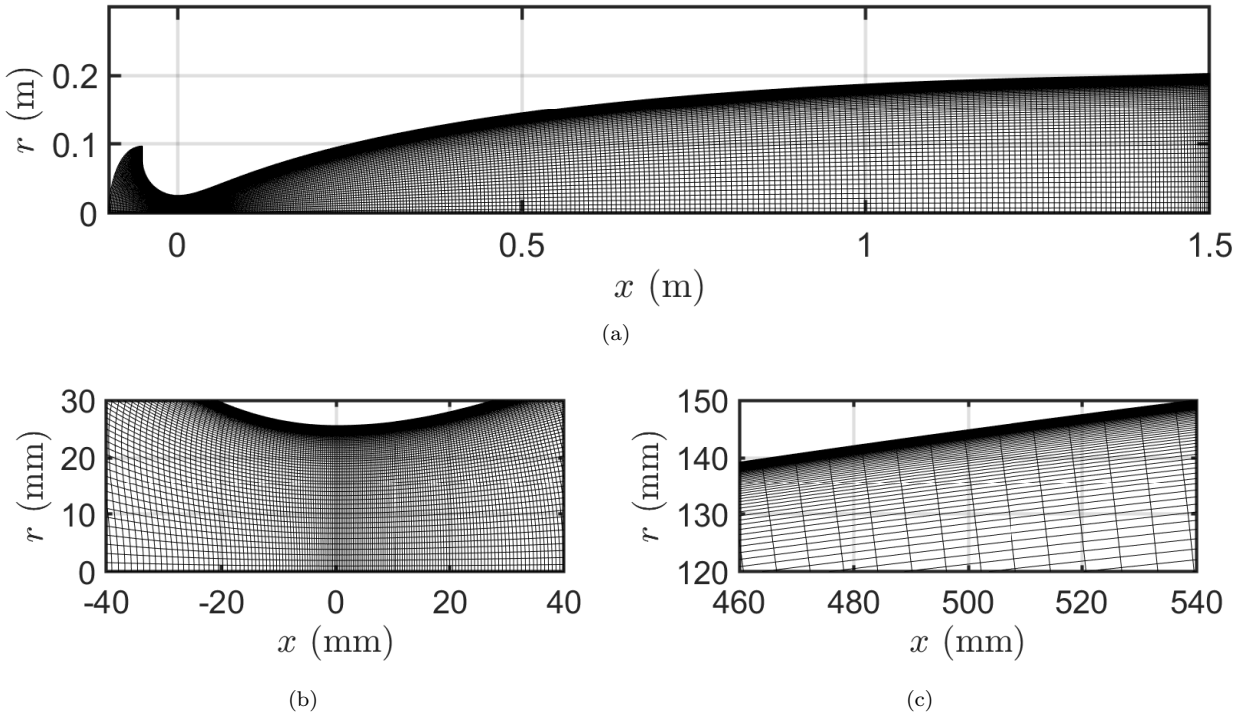


Figure 5: Nozzle grid for viscous optimization. a) Overall view, b) close-up of throat showing axial clustering and c), close-up of nozzle wall showing clustering and orthogonality at the wall.

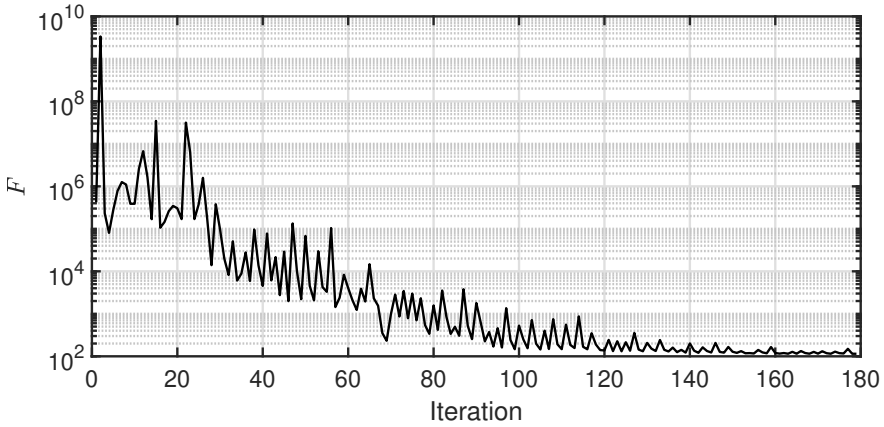


Figure 6: Variation of penalty function, F, with algorithm iteration for the viscous optimization process.

VI. Results

In this section, the results of the optimization process are presented. The final contour is given in table 1 and is plotted in Fig. 1, which is the result of the viscous optimization. The results in this section correspond to a nozzle reservoir at 1 MPa and 1800 K, with air as the gas.

Fig. 7 shows a contour plot of the Mach number in the nozzle. The plot shows that there is a core region of uniform Mach number starting from $x \approx 0.6$ m. There are also no standing waves in the nozzle.

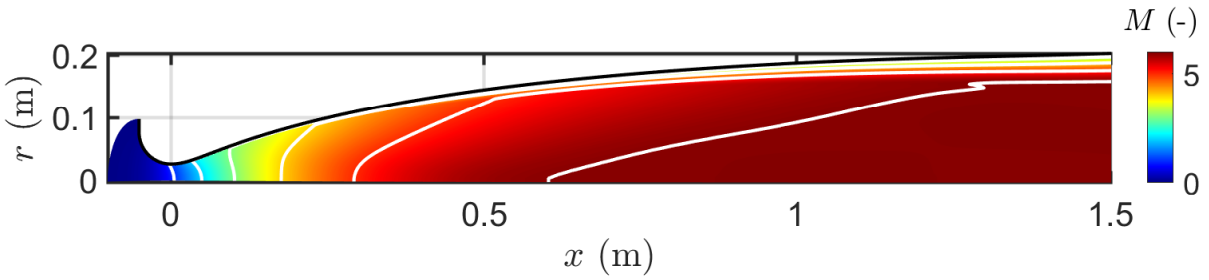


Figure 7: Contour plot of Mach number. White lines denote M =1, 2, 3, 4, 5 and 5.93 from left to right.

Figs. 8 and 9 show the various flow properties at the nozzle exit and centerline. The Mach number is uniform along the centerline after $x \approx 0.6$ m and at the nozzle exit. The size of the inviscid core at the nozzle exit is approximately 0.3 m in diameter. The flow angularity at the nozzle exit is essentially negligible at less than 1 degree. Other flow properties such as pressure, temperature, and velocity are also fairly uniform at the exit.

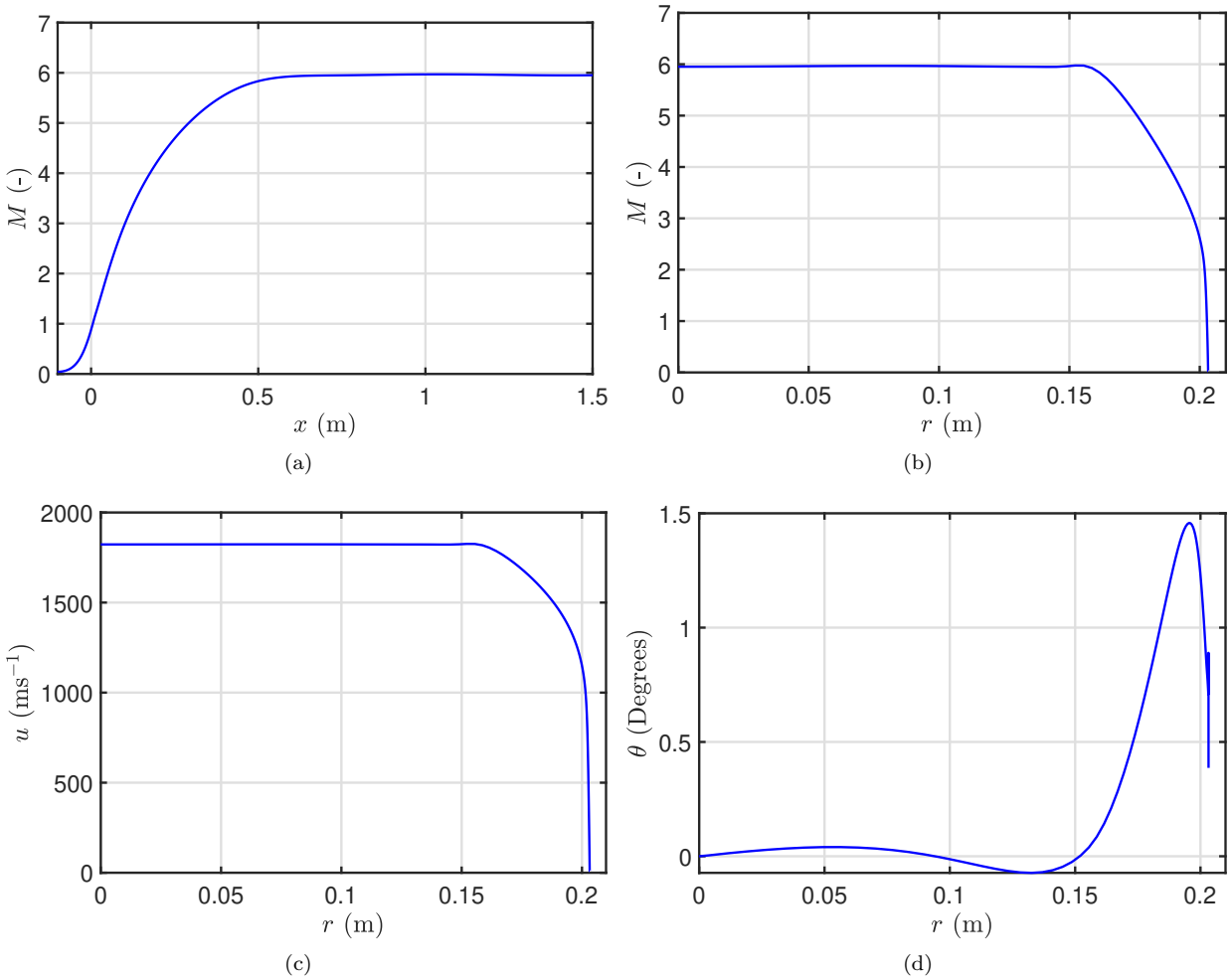


Figure 8: a) Nozzle centerline Mach number profile. b) Nozzle exit Mach number profile. c) Nozzle exit axial velocity profile. d) Nozzle exit flow angularity profile.

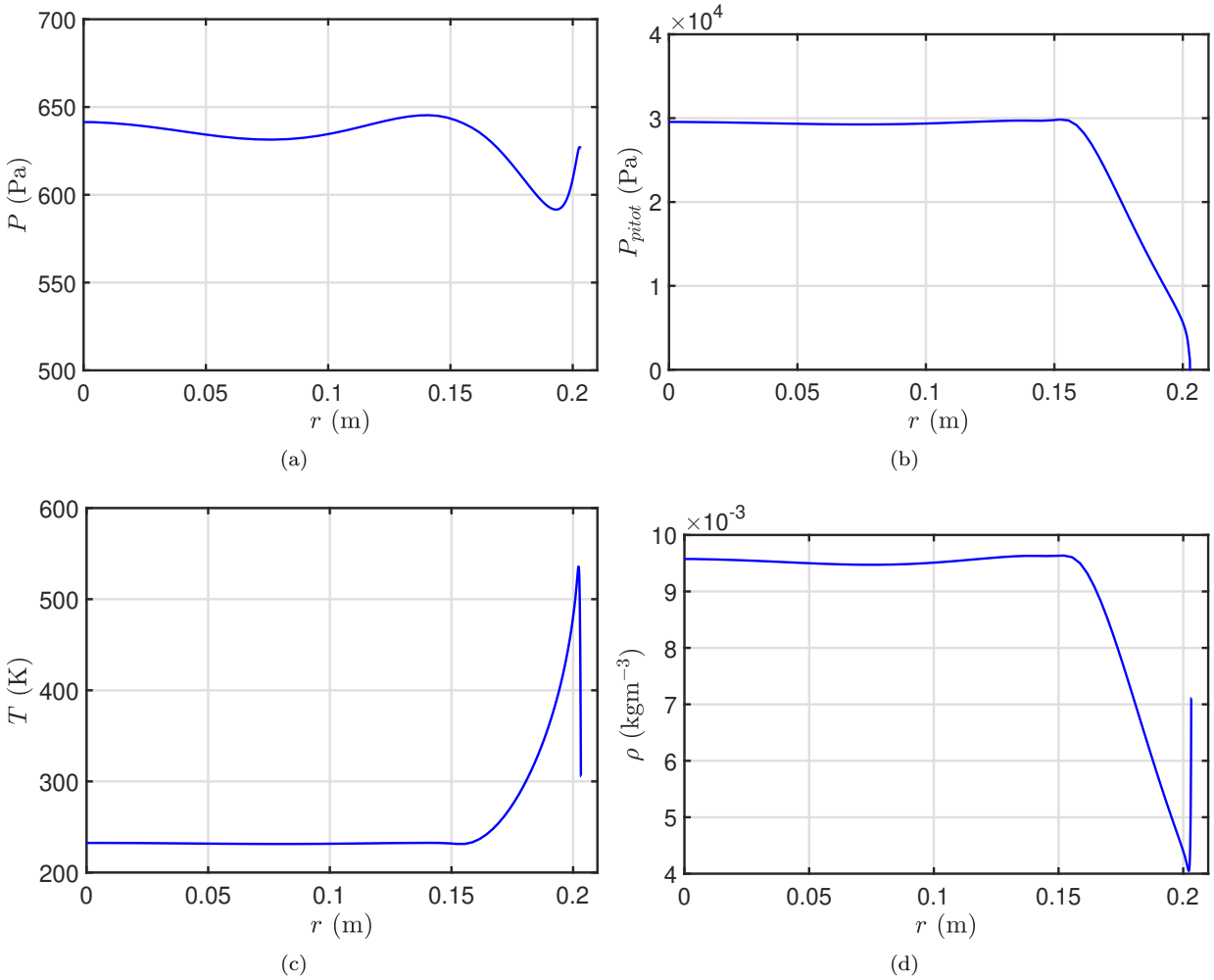


Figure 9: a) Nozzle exit pressure profile. b) Nozzle exit pitot pressure profile. c) Nozzle exit temperature profile. d) Nozzle exit density profile.

VII. Model Validation

In this section, the results from the DPLR simulations are validated. Since the accuracy of this design process depends on the accuracy of the CFD models used, it is important to ensure that the chosen model is validated.

The first method of validation is a convergence check. In Fig. 10, the results from a simulation with a 1000x200 grid are plotted against the results from a 500x100 grid (the grid used in the optimizer). As the figure shows, the Mach number profiles overlap and the grid used in the optimizer is refined enough to give a converged solution.

To further validate the model, a comparison between the results from DPLR and Eilmer 3 is presented in Fig. 11. Two simulations were run on Eilmer 3, one with the $k\omega$ turbulence model and one with the Baldwin Lomax model. The model used in DPLR is the Spalart-Allmaras model. The figure shows that all three simulations are essentially equivalent and that varying the turbulence model does not change the solutions significantly, bringing confidence to the DPLR results.

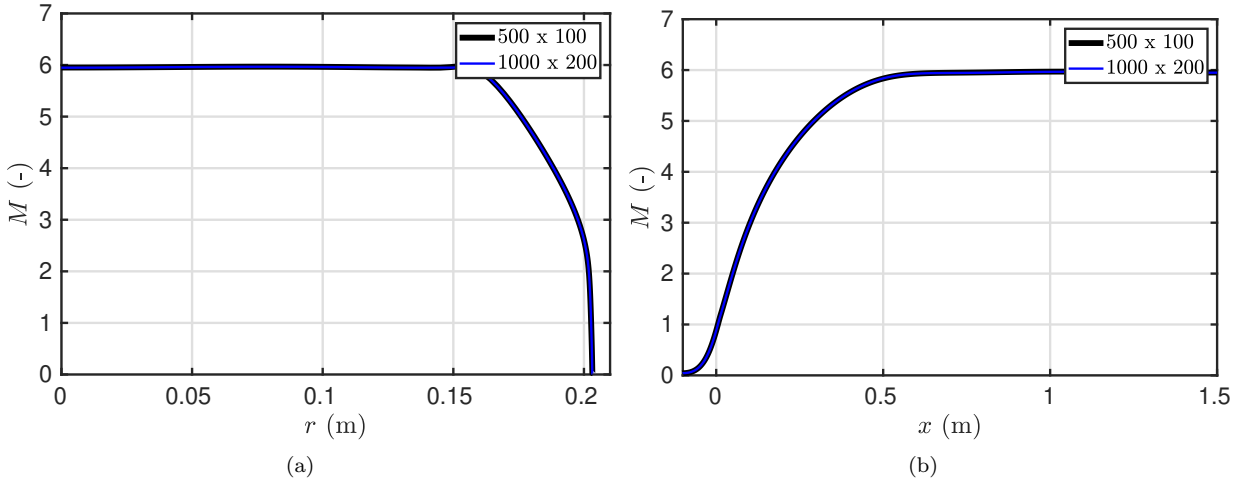


Figure 10: Mach number profiles at a) nozzle exit and b) nozzle centerline for two grid sizes.

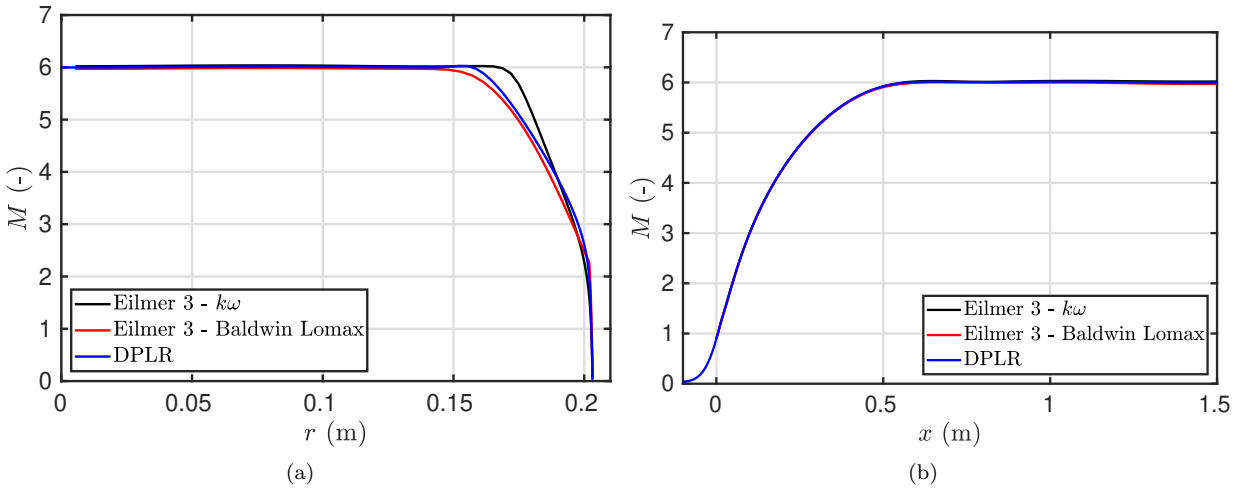


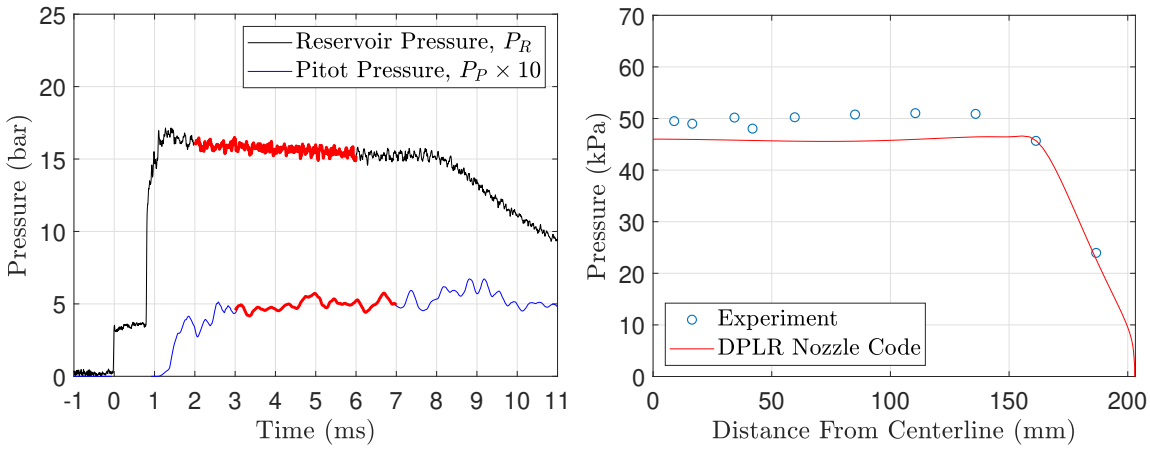
Figure 11: Mach number profiles at a) nozzle exit and b) nozzle centerline for Eilmer and DPLR simulations using different turbulence models.

VIII. Preliminary Experimental Results

We present preliminary results from some of the first experiments in the Stevens Shock Tunnel operated in reflected-shock tunnel mode. Reservoir conditions are listed in Table 2. Reservoir conditions are calculated from the incident shock speed, driven-section fill pressure, and average reservoir pressure trace using Cantera[25] and the Shock and Detonation Toolbox.[26] The reservoir conditions are then used as the input to the University of Minnesota DPLR[19] Nozzle code. In Figs. 12,13-a, we present the reservoir pressure and the Pitot pressure along with centerline. The red portion of the trace indicates the test time, noting the delay of the Pitot probe is delayed by 1 millisecond to account for expansion from the reservoir through the 1.5 m nozzle. In either case, slight under-tailoring is observed in the reservoir pressure; this will be worked on in the future. In Figs. 12,13-b we present the averaged experimental Pitot-pressure response over the test time versus distance from the Nozzle centerline and compare this result to the DPLR code output. The nozzle code somewhat under-predicts the Pitot pressure, and rectifying this discrepancy will be a focus of future work. A key positive takeaway from these experiments is the approximately 320 mm (12.5 inch) nominally inviscid core. More experimental results will be made available in forthcoming works.

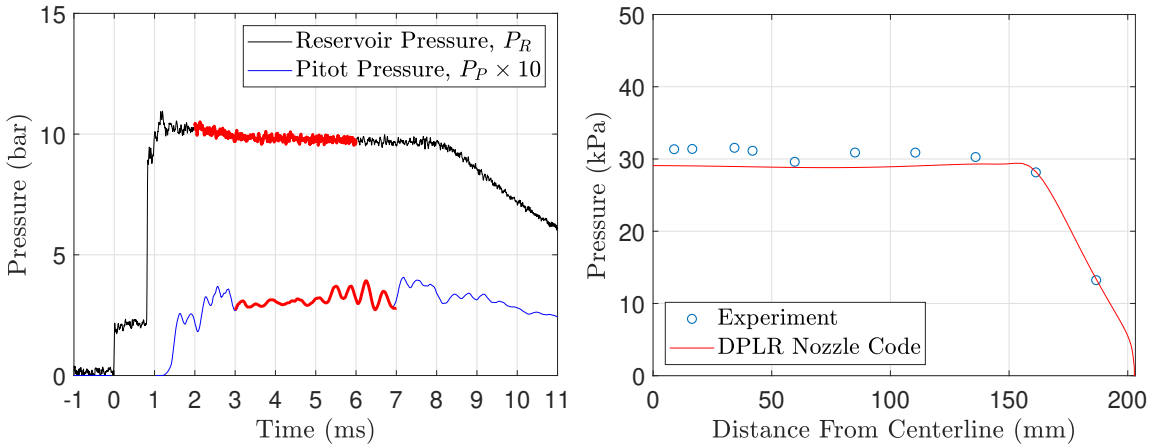
Table 2: Conditions in Preliminary Stevens Shock Tunnel Experiments. P_R, T_R, and h_R are the reservoir pressure, temperature, and mass-specific enthalpy, respectively. , U_∞, P_∞, T_∞, Tv_∞, ρ_∞, M_∞, Re_U^E are the freestream velocity, pressure, translational/rotational temperature, vibrational temperature, density, Mach number, and unit Reynolds number, respectively.

Shot	P_R	T_R	h_R	U_∞	P_∞	T_∞	Tv_∞	ρ_∞	M_∞	Re_U^E
	(bar)	(K)	(MJ/kg)	(km/s)	(kPa)	(K)	(K)	(kg/m^3)	(-)	(1/m)
70	15.7	1471	1.32	1.65	1.00	192	1035	0.018	5.94	2.23e6
72	15.0	1446	1.28	1.64	0.95	187	1038	0.018	5.95	2.20e6
73	9.8	1471	1.31	1.65	0.63	191	1101	0.011	5.94	1.42e6
74	9.8	1470	1.31	1.65	0.63	190	1101	0.011	5.94	1.42e6



(a) (b)

Figure 12: Shot 70 Results. a) Reservoir pressure and centerline Pitot pressure vs. time b) Comparison of DPLR Nozzle code output with time-average Pitot-pressure results.



(a) (b)

Figure 13: Shot 73 Results. a) Reservoir pressure and centerline Pitot pressure vs. time b) Comparison of DPLR Nozzle code output with time-average Pitot-pressure results.

IX. Conclusions

In this work, a methodology for the design of converging-diverging nozzles was presented and demonstrated with the design of an axisymmetric Mach 6 nozzle. The methodology relies on an optimization that is effectively a brute force algorithm in that it keeps changing the contour and runs the simulations and until a desirable flow is achieved in the nozzle. This is meant as an alternative to the traditional nozzle design process using the Method of Characteristics.

The basic structure of the algorithm is a three-step process. The first step is to mathematically define the contour using a Bezier curve. This is done by taking an existing contour, scaling it to the desired length and diameter, and then fitting a Bezier curve of an appropriate order to it. The Bezier curve is described by a set of control points that can be varied to change the curve. These control points are the variables that the optimizer modifies to change the curve. A penalty function is defined that quantifies the off-design nature of the flow. The goal is then to minimize the value of this penalty function. The second step is to perform the optimization while using a simple inviscid model with a coarse grid. This optimization runs quickly and results in a contour that has flow properties close to the desired conditions. The third step is to take the result of the inviscid optimization and using it start the viscous optimization. This takes longer to complete than its inviscid counterpart but gives an accurate description of the flow physics in the nozzle. The grid for the viscous optimization is more refined and clustered near the walls and throat. The result of the viscous optimization is then the Bezier curve that describes the contour of the nozzle.

The inviscid optimization in this work was run using Eilmer 3 at reservoir conditions of 1 MPa and 1800 K. The viscous optimization was run using Eilmer 3 and DPLR. Eilmer 3 was used to generate the grid and DPLR was used to run the simulation. In the DPLR simulation, all relevant flow physics such as chemical reactions and vibrational effects were accounted for.

The result was a contour that created uniform flow properties at the exit with a Mach number of 6 as computed with Eilmer 3 and DPLR. Furthermore, it was demonstrated that the grid used in the viscous optimization is refined enough to give a converged solution and that the contour gives the same results when different turbulence models and CFD solvers are used.

Future work on this design methodology involves further experimentally validating the simulation. This will be done with more Pitot pressure measurements using probes and velocity measurements using tagging velocimetry.

Acknowledgments

We thank Professor Stuart Lawrence's group at the University of Maryland for providing the contour of their nozzle for us to use as a starting point.

Mustafa and Parziale were supported by AFOSR Young Investigator Program Grant FA9550-16-1-0262, and equipment for this work was supported by AFOSR DURIP Grant FA9550-15-1-0325 and FA9550-19-1-0182. Support was also provided by U.S. Air Force Small Business Innovation Research Grants (FA9101-17-P-0094 and FA2487-19-C-0013). Shekhtman was supported by the Stevens Institute of Technology Provost Fellowship and ONR Young Investigator Program Grant N00014-20-1-2549, and equipment was supported by ONR DURIP Grants N00014-19-1-2523 and N00014-20-1-2637. Hameed was supported by AFOSR Grant FA9550-18-1-0403.

References

[1]C., S. W., Butler, C., and Laurence, S. J., "Method of Characteristics Design of High-Temperature Wind Tunnel Nozzles with Vibrational Relaxation," *2018 2018 Aerodynamic Measurement Technology and Ground Testing Conference*, AIAA-2018-3565, Atlanta, Georgia, 2018. doi: 10.2514/6.2018-3565.

[2]Chan, W. Y. K., Jacobs, P. A., Smart, M. K., Grieve, S., Craddock, C. S., and Doherty, L. J., "Aerodynamic Design of Nozzles with Uniform Outflow for Hypervelocity Ground-Test Facilities," *Journal of Propulsion and Power*, Vol. 34, No. 6, 2018, pp. 1467–1478. doi: 10.2514/1.B36938.

[3]Parziale, N. J., Shepherd, J. E., and Hornung, H. G., "Differential Interferometric Measurement of Instability in a Hypervelocity Boundary Layer," *AIAA Journal*, Vol. 51, No. 3, 2013, pp. 750–754. doi: 10.2514/1.J052013.

[4]Parziale, N. J., Shepherd, J. E., and Hornung, H. G., "Observations of hypervelocity boundary-layer instability," *Journal of Fluid Mechanics*, Vol. 781, 2015, pp. 87–112. doi: 10.1017/jfm.2015.489.

[5]Hameed, A., Parziale, N. J., Paquin, L., Butler, C., and Laurence, S. J., "Hypersonic Slender-Cone Boundary Layer Instability in the UMD HyperTERP Shock Tunnel," *Proceedings of AIAA SciTech 2020*, AIAA-2020-0362, Orlando, Florida, 6-10 January 2020. doi: 10.2514/6.2020-0362.

[6]Zahradka, D., Parziale, N. J., Smith, M. S., and Marineau, E. C., "Krypton tagging velocimetry in a turbulent Mach 2.7 boundary layer," *Experiments in Fluids*, Vol. 57, 2016, pp. 62. doi: 10.1007/s00348-016-2148-2.

[7]Mustafa, M. A., Parziale, N. J., Smith, M. S., and Marineau, E. C., "Two-Dimensional Krypton Tagging Velocimetry (KTV-2D) Investigation of Shock-Wave/Turbulent Boundary-Layer Interaction," *Proceedings of AIAA SciTech 2018*, AIAA-2018-1771, Kissimmee, Florida, 8-12 January 2018. doi: 10.2514/6.2018-1771.

[8]Mustafa, M. A., Parziale, N. J., Smith, M. S., and Marineau, E. C., "Amplification and structure of streamwise-velocity fluctuations in compression-corner shock-wave/turbulent boundary-layer interactions," *Journal of Fluid Mechanics*, Vol. 863, 2019, pp. 1091–1122. doi: 10.1017/jfm.2018.1029.

[9]Parziale, N. J., Shepherd, J. E., and Hornung, H. G., "Free-stream density perturbations in a reflected-shock tunnel," *Experiments in Fluids*, Vol. 55, No. 2, 2014, pp. 1665. doi: 10.1007/s00348-014-1665-0.

[10]Parziale, N. J., Schmidt, B. E., Damazo, J. S., Wang, P. S., Hornung, H. G., and Shepherd, J. E., "Pulsed Laser Diode for Use as a Light Source for Short-Exposure, High-Frame-Rate Flow Visualization," *Proceedings of AIAA SciTech 2015*, AIAA-2015-0530, Kissimmee, Florida, 5-9 January 2015. doi: 10.2514/6.2015-0530.

[11]Parziale, N. J., Smith, M. S., and Marineau, E. C., "Krypton tagging velocimetry of an underexpanded jet," *Applied Optics*, Vol. 54, No. 16, 2015, pp. 5094–5101. doi: 10.1364/AO.54.005094.

[12]Jewell, J. S., Parziale, N. J., Lam, K.-L., Hagen, B. J., and Kimmel, R. L., "Disturbance and Phase Speed Measurements for Shock Tubes and Hypersonic Boundary-Layer Instability," *Proceedings of 32nd AIAA Aerodynamic Measurement Technology and Ground Testing Conference*, AIAA-2016-3112, Washington, D. C., 13-17 June 2016. doi: 10.2514/6.2016-3112.

[13]Mustafa, M. A., Parziale, N. J., Smith, M. S., and Marineau, E. C., "Nonintrusive Freestream Velocity Measurement in a Large-Scale Hypersonic Wind Tunnel," *AIAA Journal*, Vol. 55, No. 10, 2017, pp. 3611–3616. doi: 10.2514/1.J056177.

[14]Mustafa, M. A. and Parziale, N. J., "Simplified read schemes for krypton tagging velocimetry in N_2 and air," *Optics Letters*, Vol. 43, No. 12, 2018, pp. 2909–2912. doi: 10.1364/OL.43.002909.

[15]Mustafa, M. A., Shekhtman, D., and Parziale, N. J., "Single-Laser Krypton Tagging Velocimetry Investigation of Supersonic Air and N_2 Boundary-Layer Flows over a Hollow Cylinder in a Shock Tube," *Physical Review Applied*, Vol. 11, No. 6, 2019, pp. 064013. doi: 10.1103/PhysRevApplied.11.064013.

[16]Jewell, J. S., Hameed, A., Parziale, N. J., and Gogineni, S. P., "Disturbance Speed Measurements in a Circular Jet via Double Focused Laser Differential Interferometry," *Proceedings of AIAA Scitech 2019*, AIAA-2019-2293, San Diego, California, 7-11 January 2019. doi: 10.2514/6.2019-2293.

[17]Shekhtman, D., Mustafa, M. A., and Parziale, N. J., "Two-photon cross-section calculations for krypton in the 190-220 nm range," *Applied Optics*, Vol. 59, No. 34, 2020, pp. 10826–10837. doi: 10.1364/AO.410806.

[18]Gollan, R. J. and Jacobs, P. A., "About the formulation, verification and validation of the hypersonic flow solver Eilmer," *International Journal for Numerical Methods in Fluids*, Vol. 73, No. 1, 2013, pp. 19–57. doi: https://doi.org/10.1002/fld.3790.

[19]Wright, M. J., Candler, G. V., and Prampolini, M., "Data-parallel Lower-upper Relaxation Method for the Navier-Stokes Equations," *AIAA Journal*, Vol. 34, No. 7, 1996, pp. 1371–1377. doi: 10.2514/3.13242.

[20]Candler, G. V., "Hypersonic Nozzle Analysis Using an Excluded Volume Equation of State," *Proceedings of 38th AIAA Thermophysics Conference*, AIAA-2005-5202, Toronto, Ontario Canada, 2005. doi: 10.2514/6.2005-5202.

[21]Johnson, H. B., *Thermochemical Interactions in Hypersonic Boundary Layer Stabity*, Ph.D. thesis, University of Minnesota, Minneapolis , Minnesota, 2000.

[22]Wagnild, R. M., *High Enthalpy Effects on Two Boundary Layer Disturbances in Supersonic and Hypersonic Flow*, Ph.D. thesis, University of Minnesota, Minnesota, 2012.

[23]Spalart, P. R. and Allmaras, S. R., "A One-equation Turbulence Model for Aerodynamic Flows," *Proceedings of 30th AIAA Aerospace Sciences Meeting and Exhibit*, AIAA 1992-439, Reno, Nevada, 1992. doi: 10.2514/6.1992-439.

[24]Catrisa, S. and Aupoix, B., "Density Corrections for Turbulence Models," *Aerospace Science and Technology*, Vol. 4, No. 1, 2000, pp. 1–11. doi: 10.1016/S1270-9638(00)00112-7.

[25]Goodwin, D. G., "An Open-Source, Extensible Software Suite for CVD Process Simulation," *Proceedings of CVD XVI and EuroCVD Fourteen, M Allendorf, F Maury, and F Teyssandier (Eds.)*, 2003, pp. 155–162.

[26]Browne, S., Ziegler, J., and Shepherd, J. E., "Numerical Solution Methods for Shock and Detonation Jump Conditions," GALCIT - FM2006-006, 2006.