University of Dayton

AEE 553 — Compressible Flow

Department of Mechanical and Aerospace Engineering

Homework 6

Author:
Evan Burke

Instructor:
Dr. Carson Running

9 November 2022

AEE 553
Compressible Flow

Homework 6

Evan Burke
9 November 2022

Contents

Homework 6

Evan Burke
9 November 2022

Problem 1

Starting with $\dot{m} = \rho u A$, prove that the mass flowrate through an isentropic choked nozzle can be written in the form:

$$\dot{m} = \frac{p_0 A^*}{\sqrt{T_0}} \sqrt{\frac{\gamma}{R}\left(\frac{2}{\gamma+1}\right)^{\frac{(\gamma+1)}{(\gamma-1)}}}$$

Assumptions:

Isentropic flow through a nozzle with a choked (sonic) throat.

Solution:

The mass flow at a given cross section in a quasi 1-D flow is given by:

$$\dot{m} = \rho u A$$

Choosing the throat of a choked nozzle as the point of interest, we replace the conditions with sonic conditions, denoted by $*$ and indicating the flow property at the location where $M = 1$.

$$\dot{m} = \rho^* u^* A^*$$

In order to cast this in terms of properties that are more easily known ahead of time, we identify relationships involving the total conditions of the flow, beginning with ρ^*. Using the ideal gas law, we can cast the sonic density in terms of pressure and temperature:

$$p^* = \rho^* R T^*$$

$$\rho^* = \frac{p^*}{RT^*}$$

Now, we find relationships for p^* and T^*.

Beginning with the isentropic relationship between total and static pressure:

$$\frac{p_0}{p} = \left(1 + \frac{\gamma-1}{2}M^2\right)^{\frac{\gamma}{\gamma-1}}$$

Setting $M = 1$:

$$\frac{p_0}{p^*} = \left(1 + \frac{\gamma-1}{2}\right)^{\frac{\gamma}{\gamma-1}}$$

$$p^* = \frac{p_0}{\left(\frac{\gamma+1}{2}\right)^{\frac{\gamma}{\gamma-1}}}$$

For temperature:

$$\frac{T_0}{T} = \left(1 + \frac{\gamma-1}{2}M^2\right)$$

Setting $M = 1$:

$$\frac{T_0}{T^*} = \left(1 + \frac{\gamma-1}{2}\right)$$

$$T^* = \frac{T_0}{\left(\frac{\gamma+1}{2}\right)}$$

Substituting into the ideal gas equation yields an expression for ρ^* in terms of p_0, T_0, R, and γ:

$$\rho^* = \frac{p_0}{\left(\frac{\gamma+1}{2}\right)^{\frac{\gamma}{\gamma-1}}} \frac{\left(\frac{\gamma+1}{2}\right)}{T_0} \frac{1}{R}$$

Next, we examine the sonic velocity term, u^*. Noting that for choked flow $M = 1$, we observe that the flow velocity must be equal to the speed of sound, a.

$$M = \frac{u^*}{a^*} = 1 \rightarrow u^* = a^*$$

$$a^* = \sqrt{\gamma R T^*}$$

Subsituting our known equation for T^*:

$$a^* = \sqrt{\frac{\gamma R T_0}{\left(\frac{\gamma+1}{2}\right)}}$$

Substituting everything back into the original mass flow equation:

$$\dot{m} = \frac{p_0}{\left(\frac{\gamma+1}{2}\right)^{\frac{\gamma}{\gamma-1}}} \frac{\left(\frac{\gamma+1}{2}\right)}{T_0} \frac{1}{R} \sqrt{\frac{\gamma R T_0}{\left(\frac{\gamma+1}{2}\right)}} A^*$$

Rearranging:

$$\dot{m} = \frac{p_0 A^*}{RT_0} \frac{\frac{\gamma+1}{2}}{\left(\frac{\gamma+1}{2}\right)^{\frac{\gamma}{\gamma-1}}} \sqrt{\frac{\gamma R T_0}{\left(\frac{\gamma+1}{2}\right)}}$$

$$\dot{m} = p_0 A^* \frac{1}{\left(\frac{\gamma+1}{2}\right)^{\frac{\gamma}{\gamma-1}}} \sqrt{\frac{\gamma R T_0}{R^2 T_0^2} \frac{\left(\frac{\gamma+1}{2}\right)^2}{\left(\frac{\gamma+1}{2}\right)}}$$

$$\dot{m} = \frac{p_0 A^*}{\sqrt{T_0}} \frac{1}{\left(\frac{\gamma+1}{2}\right)^{\frac{\gamma}{\gamma-1}}} \sqrt{\frac{\gamma}{R} \left(\frac{\gamma+1}{2}\right)}$$

$$\dot{m} = \frac{p_0 A^*}{\sqrt{T_0}} \sqrt{\frac{\gamma}{R} \left(\frac{\gamma+1}{2}\right)^{1-\frac{2\gamma}{\gamma-1}}}$$

$$\dot{m} = \frac{p_0 A^*}{\sqrt{T_0}} \sqrt{\frac{\gamma}{R} \left(\frac{\gamma+1}{2}\right)^{\frac{-\gamma-1}{\gamma-1}}}$$

$$\boxed{\dot{m} = \frac{p_0 A^*}{\sqrt{T_0}} \sqrt{\frac{\gamma}{R} \left(\frac{2}{\gamma+1}\right)^{\frac{(\gamma+1)}{(\gamma-1)}}}}$$

Homework 6

Problem 2

Givens:

$p_0 = 4000\,\text{kPa}$
$T_0 = 500\,\text{K}$
$D_{throat,inviscid} = 4.114\,\text{in} = 0.1045\,\text{m}$
$D_{throat,real} = 3.71\,\text{in} = 0.0942\,\text{m}$
$M_e = 6$

Assumptions:

Flow is isentropic everywhere outside of any shock waves that are present. Steady, inviscid, quasi-1D flow through nozzle. Upstream reservoir is large enough to assume that static conditions are equal to total conditions. The nozzle is choked, i.e., $M_{throat} = 1$. $\gamma = 1.4$, $R = 287\,\text{J/kg}\cdot\text{K}$.

Note: All calculations performed in Python, see Appendix A.

(a) Mass flowrate through a choked nozzle is given by the following equation derived in Problem 1:

$$\dot{m} = \frac{p_0 A^*}{\sqrt{T_0}}\sqrt{\frac{\gamma}{R}\left(\frac{2}{\gamma+1}\right)^{\frac{(\gamma+1)}{(\gamma-1)}}}$$

p_0 and T_0 are given and A^* is easily calculated:

$$A^* = \frac{\pi D_{throat}^2}{4} = 0.0086\,\text{m}^2$$

$$\boxed{\dot{m} = 62.01\,\text{kg/s}}$$

(b) The Area-Mach relationship is given by the following equation:

$$\left(\frac{A}{A^*}\right)^2 = \frac{1}{M^2}\left[\frac{2}{\gamma+1}\left(1+\frac{\gamma-1}{2}M^2\right)\right]^{(\gamma+1)/(\gamma-1)}$$

With a known throat area (A^*) and exit Mach ($M_e = 6$) it is simple to calculate exit area, A_e:

$$A_e = \sqrt{\frac{{A^*}^2}{M^2}\left[\frac{2}{\gamma+1}\left(1+\frac{\gamma-1}{2}M^2\right)\right]^{(\gamma+1)/(\gamma-1)}}$$

$$\boxed{A_e = 0.4560\,\text{m}^2}$$

(c) The design back pressure, $p_{b,design}$, and temperature, $T_{b,design}$ are the static conditions associated with $M_e = 6$ exit flow assuming no shocks in the nozzle. Using isentropic relations and treating the reservoir conditions as total conditions for the flow, the design point conditions at the exit of the nozzle are as follows:

$$\boxed{p_{b,design} = 2533.45\,\mathrm{Pa} \qquad T_{b,design} = 60.97\,\mathrm{K}}$$

(d) The lowest back pressure for which there is only subsonic flow in the nozzle can be determined by first solving for the subsonic Mach number associated with the nozzle geometry. Using the Area-Mach relationship and a numerical solver, we can back out the subsonic solution associated with the given value of A_e/A^*.

$$M_{e,sub} = 0.0108$$

Then, we utilize isentropic relations to determine the back pressure associated with this subsonic exit Mach, assuming that the nozzle exit and region just downstream are in equilibrium.

$$\boxed{p_{b,sub} = 3999668\,\mathrm{Pa}}$$

This value is 99.9999% of the total condition.

(e) The back pressure for which there is a normal shock at the nozzle exit plane is given by the value of pressure associated with the design exit Mach goingi through a normal shock wave. We use normal shock relations to determine the post-normal shock static pressure for $M_e = 6$ flow.

$$\boxed{p_b = 105982\,\mathrm{Pa}}$$

(f) The critical value of back pressure below which there are no shock waves (OS or EW) in the nozzle is the same as the pressure for which the normal shock stands at the exit plane. There is a very exact value of pressure that will keep the shock at the nozzle exit, pressures on either side of this value can push shock waves into or out of the nozzle.

$$\boxed{p_b < 105982\,\mathrm{Pa}}$$

(g) The range of back pressures for which there are oblique shock waves in the nozzle exhaust is limited on the high end by the exit-plane normal shock value and on the lower end by the design exit pressure for $M_e = 6$. In this range, the exit flow has a lower static pressure than the downstream region and must go through an oblique shock to be in equilibrium.

$$\boxed{2533.45\,\text{Pa} < p_b < 105982\,\text{Pa}}$$

(h) The range of back pressures for which there are expansions waves is limited on the high end by the design exit pressure, p_e, and has no limit on the lower end. In this range, the exit flow has a higher static pressure than the downstream region and must go through an expansion wave to be in equilibrium.

$$\boxed{p_b < 2533.45\,\text{Pa}}$$

(i) To find the back pressure for which a normal shock wave occurs in the divergent section of the nozzle at the point where the cross-sectional area is equal to the average of the throat and exit planes, we start by calculating the average area.

$$A_{avg} = 0.2323\,\text{m}^2$$

Using the Area-Mach number relation and a numerical solver, we determine the supersonic velocity at this point.

$$M_{avg} = 5.1$$

Finally, we use normal shock relations to determine the post-normal shock pressure associated with this flow, which is equivalent to the back pressure that would hold a normal shock at this point in the nozzle.

$$\boxed{p_b = 203040\,\text{Pa}}$$

(j) To calculate the time until there is a normal shock at the exit plane of the nozzle, we must determine the pressures in the receiving tanks assuming a constant choked mass flow rate. We assume that transient effects are negligible and the temperature in the receiving tanks is a constant $T = 295\,\text{K}$. We also make the assumption that the receiving tanks are large enough that the air is essentially still and static conditions are equal to total conditions. For a constant tank temperature, the back pressure will be a function solely of the density of the air in the receiving tanks. Using the ideal gas law, $p = \rho RT$, we have a relationship between back pressure and the volume of air in the tank. Using the choked mass flow rate and the volume of the tanks, we can determine the density of the air in the receiving tanks at any time t.

$$\dot{m} = 62.01\,\text{kg/s}$$

$$\mathcal{V}_{tanks} = 4000\,\text{gal} = 15.14165\,\text{m}^3$$

$$p_b(t) = \frac{\dot{m}}{\rlap{-}V_{tanks}} RTt$$

Knowing the value of back pressure we want to solve for ($p_b = 105982\,\text{Pa}$) we can easily solve for the time at which the shock will exist at the nozzle exit.

$$t_{NS} = \frac{p_b \rlap{-}V_{tanks}}{\dot{m}RT}$$

$$\boxed{t_{NS} = 0.3056\,\text{s}}$$

(k) To determine the length of the driver tubes, we must first find calculate the fluid velocity during the time frame of interest. Assuming that conditions in the driver tube are equal to the stagnation conditions of the upstream reservoir, we first calculate the fluid density in the driver tube.

$$\rho_{tube} = \frac{P_0}{RT_0} = 27.87\,\text{kg/m}^3$$

Next, the amount of mass used during the time frame of interest is found using the known mass flow and time.

$$m_{used} = \dot{m}t = 18.95\,\text{kg}$$

The volume of air at this density associated with the known mass is then calculated.

$$\rlap{-}V = \frac{m_{used}}{\rho_{tube}} = 0.68\,\text{m}^3$$

From the paper, the driver tubes have an inner diameter of $D = 9.75\,\text{inch} = 0.24765\,\text{m}$. The area of the tube is easily calculated.

$$A_{tube} = \frac{\pi D_{tube}^2}{4} = 0.0482\,\text{m}^2$$

With a known volume and area, we can calculate the length of driver tube used during the time frame of interest.

$$L_{tube} = \frac{\rlap{-}V}{A_{tube}}$$

$$\boxed{L_{tube} = 14.12\,\text{m} = 46.31\,\text{ft}}$$

With a total internal driver tube length of 82 feet, this value is within reason.

Problem 3

The following equation can be used to calculate thrust from the SSME:

$$Thrust = \dot{m}u_{exit} + \left(p_{exit} - p_{ambient}\right) A_{exit}$$

Givens:

$A_{inlet} = 0.21\,\mathrm{m}^2$
$A_{throat} = 0.054\,\mathrm{m}^2$
$A_{exit} = 4.17\,\mathrm{m}^2$
$p_0 = 20.408\,\mathrm{MPa}$
$T_{inlet} = 3600\,\mathrm{K}$
$\gamma = 1.2$
$R = 287\,\mathrm{J/kg \cdot K}$

Assumptions:

The nozzle is choked and isentropic throughout the duration.

(a) Plot the thrust of the SSME as a function of altitude from sea level to 20 km above sea level. Include markers where the nozzle flow is under-expanded, over-expanded, and at design conditions (aerodynamically speaking). The following equation can be used to calculate ambient pressure:

$$p_{ambient} = 101325(1 - (2.25577 \cdot 10^{-5} \cdot h))^{5.25588}$$

where h is height (in meters) above sea level.

Solution:

Given the SSME thrust equation, the parameters of interest for the problem are:

- $\dot{m}$
- u_{exit}
- p_{exit}
- $p_{ambient}$
- A_{exit}

Of these, only the exit area is given.

We begin with $\dot{m}$. Mass flow through a choked nozzle is given by the following equation:

$$\dot{m} = \frac{p_0 A^*}{\sqrt{T_0}} \sqrt{\frac{\gamma}{R}\left(\frac{2}{\gamma+1}\right)^{\frac{(\gamma+1)}{(\gamma-1)}}}$$

p_0 is given, but T_0 is unknown. Given A_{inlet}/A_{throat} and recognizing that the flow is sonic at the throat (i.e., $A_{throat} = A^*$), we can numerically solve the Area-Mach relation for the (subsonic) Mach number at the inlet to the nozzle.

$$\left(\frac{A_{inlet}}{A^*}\right)^2 = \frac{1}{M_{inlet}^2}\left[\frac{2}{\gamma+1}\left(1+\frac{\gamma-1}{2}M_{inlet}^2\right)\right]^{(\gamma+1)/(\gamma-1)}$$

$$M_{inlet} = 0.1542$$

With the inlet Mach number and static temperature we can use isentropic relations to solve for the total temperature in the nozzle.

$$T_0 = 3608.56\,\mathrm{K}$$

Substituting known values into the mass flow equation yields the SSME choked mass flow:

$$\dot{m} = 702.29\,\mathrm{kg/s}$$

Next, we solve for p_{exit} and u_{exit}. Once again using the Area-Mach relation and a numerical solver, we utilize A_{exit}/A_{throat} to solve for the exit Mach number.

$$M_e = 4.704$$

Then, we can use isentropic relations to calculate p_e and T_e.

$$p_e = 18555.9\,\mathrm{Pa}$$

$$T_e = 1123.12\,\mathrm{K}$$

To calculate the exit velocity, we solve for the sonic velocity at the exit plane.

$$a_e = \sqrt{\gamma R T_e} = 621.95\,\mathrm{m/s}$$

The exit velocity is now easily found.

$$u_e = M_e a_e = 2925.7\,\mathrm{m/s}$$

The final piece of information required to calculate SSME thrust is $p_{ambient}$. Given the relation between ambient pressure and altitude we calculate ambient pressure at all altitudes between 0 and 20 km. Figure 1 shows the curve of pressure vs. altitude for the range of altitudes of interest.

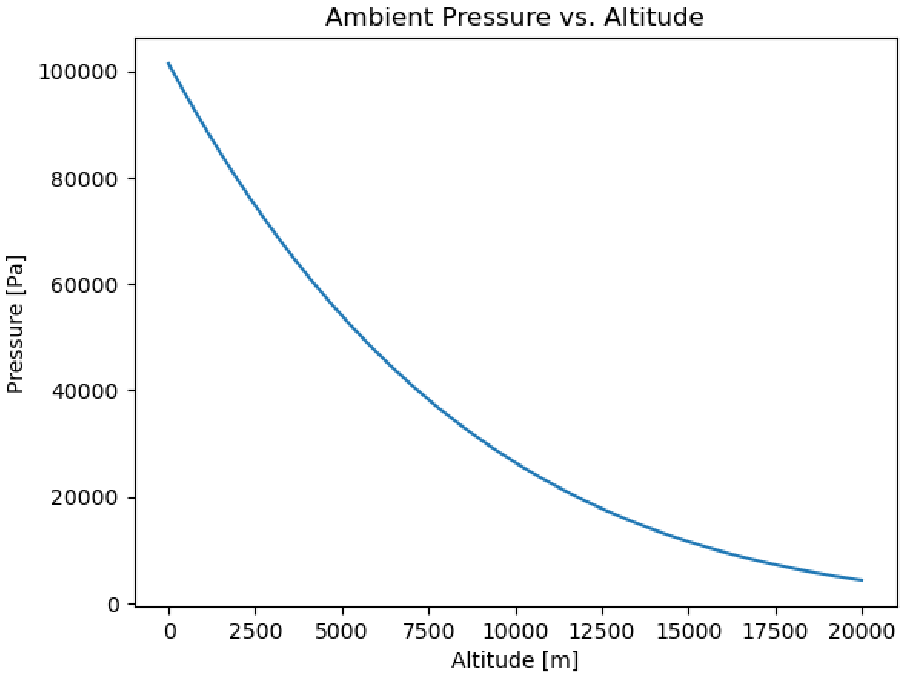

Figure 1: P_{amb} vs. Altitude

Using the calculated ambient pressures we can finally calculate thrust at all points of interest. Figure 2 shows the SSME thrust vs. altitude curve between 0 and 20 km. The SSME engine initially operates at an overexpanded condition, shown by the blue dotted line. The design condition of the engine (where there are no shocks or expansions in the nozzle exhaust) comes at approximately $h = 12236\,\text{m}$, with a design thrust of $Thrust = 2054.7\,\text{kN}$, noted with a red star. Beyond the design condition, the nozzle does not generate a sufficiently low static exit pressure value so the flow goes through expansion waves to reach equilibrium with the ambient pressure. The underexpanded portion of the thrust curve is shown by the dashed orange line.

(b) Do you expect to see shock-diamonds or a plume when the space shuttle takes off?

Discussion:

Because the SSME is initially overexpanded at takeoff, shock diamonds would be present in the exhaust of the SSME. The low exit pressure from the nozzle goes through a series of oblique shocks to reach equilibrium with the ambient pressure, beginning a train of shock-diamonds

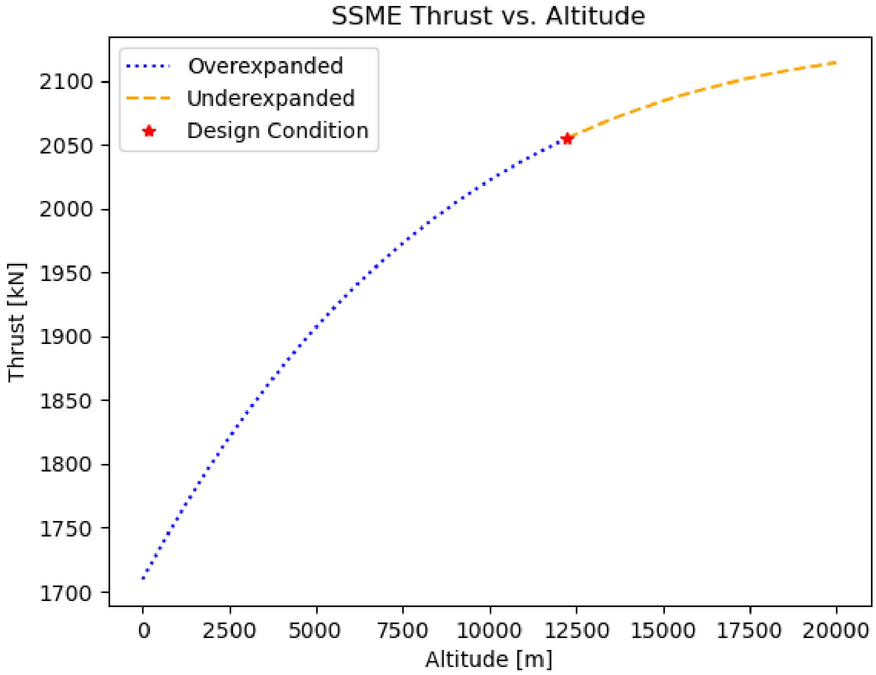

Figure 2: Thrust vs. Altitude

that dissipate as the difference between exit pressure and ambient pressure shrinks at higher altitudes.

(c) Does the "design condition" pertain to maximum thrust? Briefly explain.

Discussion:

The design condition does not indicate the point of maximum thrust. As shown in figure 2, the thrust continues to increase beyond the design point. The key difference is that the thrust is less efficient outside of the design point, potentially limiting the vehicle's performance far away from the design condition. The Space Shuttle did not fire the SSME at low altitudes, instead relying on the Solid Rocket Boosters (SRBs). The SRBs were designed to operate more efficiently at lower altitudes than the SSME which spent the majority of its time in flight at lower ambient pressures.

Appendix A Problem 2 Python Code

```python
# Compressible Flow
# AEE 553
# Homework 6 - Problem 2
# Evan Burke

import numpy as np
from matplotlib import pyplot as plt
import shocks as ns
import oblique as os
import isentropic as isen
from scipy.optimize import fsolve

pt = 4000*1000 # Pa
Tt = 500 # K
Me = 6 # Design exit Mach
D_th = 4.114 # throat diameter, inviscid, inches
D_th_r = 3.71 # throat diameter, real, viscous, inches
R = 287
gamma = 1.4

# Convert diameters to meters
D_th = D_th * 0.0254
D_th_r = D_th_r * 0.0254

print(f'Throat Diameter (Inviscid) = {D_th}')
print(f'Throat Diameter (Viscous) = {D_th_r}')

A_th = np.pi * D_th**2/4
print(f'A* = {A_th}')

def mass_flow(pt=None, A_star=None, Tt=None, gamma=1.4, R=287):
    mdot = pt*A_star / Tt**0.5 * (gamma/R * (2/(gamma+1))**((gamma+1)/(gamma-1)))**0.5
    print(f'Choked Mass Flow Rate = {mdot} kg/s')
    return mdot

def A_from_A_star(A_star=None,M=None,gamma=1.4):
    A = ((A_star**2/M**2) * (2/(gamma+1) * (1 + (gamma-1)/2 * M**2 ))**((gamma+1)/(gamma-1)))**0.5
    print(f'Area for M = {M}: {A} m^2')
    return A

# Part A
mdot = mass_flow(pt=pt, A_star=A_th, Tt=Tt, gamma=1.4, R=287)

# Part B
A_exit = A_from_A_star(A_star=A_th,M=Me,gamma=1.4)
```

```python

# Part C
p_exit = isen.get_static_pressure(M=Me,p_t=pt)
T_exit = isen.get_static_temperature(M=Me,T_t=Tt)

# Part D
def A_A_star(M=None,A_A_star=None,gamma=1.4):
    eq = ((1/M**2) * (2/(gamma+1) * (1 + (gamma-1)/2 * M**2 ))**((gamma+1)/(gamma-1)))**0.5 - A_A_star
    #print(f'A/A* = {A_A_star}')
    return eq

M_e_sub = float(fsolve(A_A_star,x0=0.01,args=(A_exit/A_th)))
print(f'Subsonic Exit Mach = {M_e_sub}')
p_e_sub = isen.get_static_pressure(M=M_e_sub,p_t=pt)

# Part E

# Normal shock will stand at nozzle exit when
# static pressure is equal to the static pressure
# across a normal shock at the nozzle design condition
p_exit_ns = ns.get_static_pressure_normal_shock(M1=Me,p1=p_exit)
print(f'Back Pressure for which exit NS= {p_exit_ns}')

# Part F
print(f'Back Pressure below which no shocks in nozzle = {p_exit_ns}')

# Part G
# Range of back pressures for which there are oblique shocks
# in nozzle exhaust
# pe < pb
print(f'Range of back pressures for oblique shocks: {p_exit} < p_b < {p_exit_ns}')

# Part H
# Range of back pressures for expansion waves
# pb < pe

print(f'Range of back pressures for expansion waves: p_b < {p_exit}')

# Part I

A_avg = (A_th + A_exit)/2
print(f'Average nozzle area = {A_avg} m^2')
M_avg = float(fsolve(A_A_star,x0=1.5,args=(A_avg/A_th)))
print(f'M_avg = {M_avg}')
p_avg = isen.get_static_pressure(M=M_avg,p_t=pt)
p_avg_NS = ns.get_static_pressure_normal_shock(M1=M_avg,p1=p_avg)

# Part J
```

```
T_tank = 295
tank_vol = 4000 # gal
tank_vol = tank_vol * 0.00378541 # mˆ3

ts = np.linspace(0,300,num=301,endpoint=True)
pbs = [mdot/tank_vol*t*R*T_tank for t in ts]

t_NS = p_exit_ns * tank_vol / (mdot * R * T_tank)
print(f’Time to fill receiving tanks to yield exit NS = {t_NS}’)

# Part K
mass_used =mdot*t_NS
print(f’Mass used until NS = {mass_used}’)
rho_tube = isen.get_static_density(p=pt,T=Tt)
D_driver = 9.75
D_driver = 0.24765
A_driver = np.pi*D_driver**2/4
volume_used = mass_used/rho_tube
L_used = mass_used/(rho_tube*A_driver)

print(f’Volume used = {volume_used}’)
print(f’Driver Tube cross sectional area = {A_driver}’)
print(f’Length used = {L_used} m = {L_used*3.28084} ft’)
```

Appendix B Problem 3 Python Code

```python
# Compressible Flow
# AEE 553
# Homework 6 - Problem 2
# Evan Burke

import numpy as np
from matplotlib import pyplot as plt
import shocks as ns
import oblique as os
import isentropic as isen
from scipy.optimize import fsolve

pt = 20.408 * 10**6 # MPa to Pa
T_i = 3600 # static temp at inlet of CD nozzle, K
A_i = 0.21 # area at inlet of CD nozzle, m^2
A_th = 0.054 # area at throat of CD nozzle, m^2
A_e = 4.17 # area at exit of CD nozzle, m^2
R = 287
gamma = 1.2 # this is different!!!

def mass_flow(pt=None, A_star=None, Tt=None, gamma=1.4, R=287):
    mdot = pt*A_star / Tt**0.5 * (gamma/R * (2/(gamma+1))**((gamma+1)/(gamma-1)))**0.5
    print(f'Choked Mass Flow Rate = {mdot} kg/s')
    return mdot

def A_from_A_star(A_star=None,M=None,gamma=1.4):
    A = ((A_star**2/M**2) * (2/(gamma+1) * (1 + (gamma-1)/2 * M**2 ))**((gamma+1)/(gamma-1)))**0.5
    print(f'Area for M = {M}: {A} m^2')
    return A

altitudes = np.linspace(0,20000,num=20001, endpoint=True)
print(altitudes)

pressures = [101325*(1-(2.25577*10**(-5)*h))**5.25588 for h in altitudes]
print(pressures[0:10])

def SSME_thrust(mdot=None,u_e=None,p_e=None,p_amb=None,A_e=None):
    thrust = mdot * u_e + (p_e-p_amb)*A_e
    return thrust

# To get mdot:
# Need pt, A_th, Tt, gamma, R
# pt is given, Tt unknown, M_th = 1, A_i/A_th known
# Solve for M_i from A_i/A_th given that M_th = 1

```

```python
def A_A_star(M=None,A_A_star=None,gamma=1.4):
    eq = ((1/M**2) * (2/(gamma+1) * (1 + (gamma-1)/2 * M**2 ))**((gamma+1)/(gamma-1)))**0.5 - A_A_star
    return eq

M_i = float(fsolve(A_A_star,x0=0.150,args=(A_i/A_th,gamma)))
print(f'Nozzle Inlet Mach = {M_i}')
Tt = isen.get_total_temperature(M=M_i,T=T_i,gamma=gamma)
print(f'Total temperature at nozzle inlet = {Tt} K')

mdot = mass_flow(pt=pt,A_star=A_th,Tt=Tt,gamma=gamma,R=R)

# mdot, A_e, p_ambs known, need u_e and p_e
# Use A/A* relationship to get exit Mach number, exit sonic velocity, exit velocity

M_e = float(fsolve(A_A_star,x0=6,args=(A_e/A_th,gamma)))
print(f'Exit Mach = {M_e}')
p_e = isen.get_static_pressure(M=M_e,p_t=pt,gamma=gamma)
print(f'Exit static pressure = {p_e} Pa')
T_e = isen.get_static_temperature(M=M_e,T_t=Tt,gamma=gamma)
print(f'Exit static temperature = {T_e} K')
a_e = isen.get_sonic_velocity(T=T_e,gamma=gamma)
print(f'Exit sonic velocity = {a_e} m/s')
u_e = M_e*a_e
print(f'Exit velocity = {u_e} m/s')

thrusts = [SSME_thrust(mdot=mdot,u_e=u_e,p_e=p_e,p_amb=p_i,A_e=A_e)/1000 for p_i in pressures]

fig,ax = plt.subplots()
ax.set_ylabel('Pressure [Pa]')
ax.set_xlabel('Altitude [m]')
ax.set_title('Ambient Pressure vs. Altitude')
plt.plot(altitudes,pressures)
plt.savefig('../images/problem_3/pamb_vs_alt.png', bbox_inches='tight')
plt.close()

thrust_design = SSME_thrust(mdot=mdot,u_e=u_e,p_e=p_e,p_amb=p_e,A_e=A_e)/1000
print(f'Design thrust = {thrust_design} kN')

def pvh(h=None,p=None):
    eq = 101325*(1-(2.25577*10**(-5)*h))**5.25588 - p
    return eq

alt_design = float(fsolve(pvh,x0=10000,args=(p_e)))
print(f'Design altitude = {alt_design} m')

fig,ax = plt.subplots()
```

```
ax.set_ylabel('Thrust [kN]')
ax.set_xlabel('Altitude [m]')
ax.set_title('SSME Thrust vs. Altitude')
#plt.plot(altitudes,thrusts)
plt.plot(altitudes[0:12235],thrusts[0:12235],'b',linestyle='dotted',label='Overexpanded')
plt.plot(altitudes[12237:],thrusts[12237:],color='orange',linestyle='dashed',label='Underexpanded')
plt.plot(alt_design,thrust_design,'r*',label='Design Condition')
plt.legend()
plt.savefig('../images/problem_3/t_vs_alt.png', bbox_inches='tight')
```