UNIVERSITY OF MINNESOTA
Mechanical Engineering
ME 4053W

ME4053
Mechanical Engineering Modeling

DATE: 10/15/2021
TO: Frank Kelso, Matt Anderson
CC: Berik Kallevig, Krishna Chandra Bavandla, Samyak Jain, Ben Szot.

FROM: Ryker Zierden
SUBJECT: Project 1

Abstract

In this experiment, a divide-and-conquer Schmidt analysis was performed on a Stirling Engine system with given initial constraints and parameters. The divide-and-conquer method found torque over a crank angle range (with various intermediate steps in between) then used numerical integration to find the kinetic energy that the flywheel would provide, which was used to find the flywheel parameters. From the analysis, the designed flywheel was found to have a moment of inertia of 3.19 kg*m^2, a mass of 25.48 kg, an inner diameter of 0.69 meters, and an outer diameter of 0.79 meters. The torque rating for the system was found to be 6.44 Nm and the average power rating for the system was found to be 555 Watts. The efficiency of the system was found to be 38.6% when compared to an ideal Stirling Engine Cycle. This efficiency is likely low due to the isochoric assumptions made for the heating and cooling of the system in the ideal analysis. In a real-life application, this efficiency would likely be lower than the calculated value since the simulated analysis performed in this experiment did not account for some real-life losses such as leaking air, environmental heat loss, and inconsistent temperatures.

Table of Contents

1 Introduction 1
2 Approach 1
2.1 Apparatus 1
2.2 Experimental Methods 1
3 Results 3
3.1 Volume Analysis 3
3.2 Pressure Analysis 3
3.3 Force Analysis 4
3.4 Torque Analysis 4
3.5 Cycle Analysis 5
3.6 Flywheel Design Parameters 5
4 Discussion 6
4.1 Volume Analysis 6
4.2 Pressure Analysis 6
4.3 Force Analysis 6
4.4 Torque Analysis 6
4.5 Cycle Analysis 6
4.6 Flywheel Design Parameters 7
5 Conclusion 7
6 References 8
APPENDICES 9
Appendix A – Given System Design Parameters 9
Appendix B – Derivations 10
Appendix C – Validation Calculation 16
Appendix D – MATLAB Code 16
fly.m 16
getVolume.m 21
getForce.m 22
getTorque.m 23
getRoots.m 24
isothermal.m 24

1 Introduction

Stirling Engines have many modern use cases where efficiency and low noise output are desired (as opposed to something like a combustion engine). Some of these use cases include cooling systems and low-noise automotive systems. Learning how to analyze and design Stirling Engines increases the amount of power-generating systems that can be considered when working in industry.

The purpose of this experiment was to design a flywheel for a beta Stirling Engine. The flywheel is an important part in a Stirling Engine design since it allows the engine to maintain function during the parts of the cycle where sufficient torque cannot be supplied to the piston assembly. A well-designed flywheel eliminates the need to "overdesign" the engine to supply the maximum torque of the required use case, but rather enables the engine to be used based on the *average* required torque. To design the flywheel, a comprehensive MATLAB simulation was performed to operate around given design parameters (see Appendix A for details).

2 Approach

2.1 Apparatus

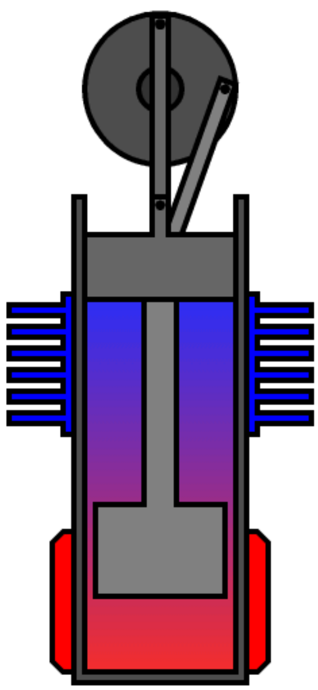


Figure 1 – Beta Stirling Engine with piston, displacer, and flywheel. Hot and cold regions shown with red and blue colors, respectively [1]

2.2 Experimental Methods

To analyze the Stirling Engine and obtain the desired design parameters for the flywheel, a divide-and-conquer approach was used, calculating each necessary value one at a time and working through the problem in pieces. The first value that was found was volume. This value was found by simply using the geometry of the system to find the volume under the cylindrical piston. The same method was then used to find the volume under the displacer, with the area of the two sections seen in figure one being found from these two values. The area in the regenerator was assumed to be constant. The derivation for the equations used to find volume, as well as those used for the later pieces of the divide-and-conquer analysis, can be found in Appendix B.

Next, the pressure in the piston was found. Five assumptions were used in this step: the ideal gas assumption for the air in the engine, the assumption that the hot and cold region temperatures (in the compression and expansion regions of the engine, respectively) remained constant, the assumption that the temperature in the regenerator was constant and equal to the average between the hot and cold region temperatures, the assumption that the pressure was constant throughout the system, and the assumption that no air leaked from the system (the mass of the air was constant). The assumptions of constant pressure and temperature are considered to be lumped analysis assumptions. The ideal gas law was then applied to the entire system to find the pressure of the air contained throughout. To use the ideal gas law, however, the mass of the air in the system had to be found. This was accomplished by using the ideal gas law with the minimum pressure that was given (also equal to the atmospheric pressure). Lastly, the pressure was converted to relative pressure by simply subtracting the atmospheric pressure (same as the minimum pressure) from the calculated pressure values.

From pressure, force was calculated by using trigonometry and free body diagrams and obtaining the magnitude of the force along the connecting rod of the piston. The force from the pressure of the system was accounted for in this step as well as the force from friction at the piston (which varied in direction depending on the point in the cycle). Torque was then calculated using trigonometry and the force calculated in the previous step.

Next, the kinetic energy that was provided from the flywheel was calculated. This was accomplished by integrating the torque function for any points over the average torque in terms of crank angle in radians. Numerical methods were used for this integration, and the mean-value-theorem was applied to find the average.

Now that the energy that the flywheel had to provide was known, the moment of inertia of the flywheel could be found with the following equation:

$$\Delta KE = J * \omega_{ave}^2 * C_f \ (1)$$

Where omega represents the average angular velocity, J represents the moment of inertia, and C_f represents the coefficient of fluctuation of the system. Omega average and C_f were provided in the initial problem statement. A complete listing of the given values can be found in Appendix A.

From the moment of inertia, the mass, inner, and outer diameter of the flywheel were determined using equations that model the geometry and inertia of a hollow cylinder [2]. The density of steel was also necessary for moving between mass and volume in this step [3].

The work for the cycle was found by using numerical integration with pressure as a function of volume for the cycle. This was used to find the average power of the cycle. Additionally, the efficiency of the cycle was found by using a value for ideal work determined by numerical integration of the pressure-volume plots in an ideal, isothermal cycle analysis. The efficiency of the cycle was determined by finding the ratio between the actual work of the cycle and the work of the ideal cycle:

$$\eta = \frac{W_{actual}}{W_{ideal}} * 100\% \ (2)$$

3 Results

3.1 Volume Analysis

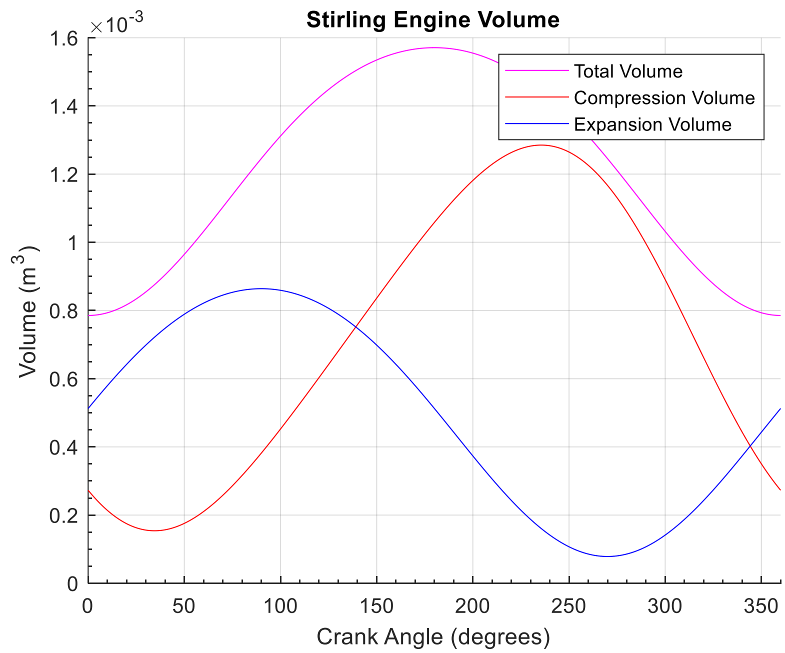


Figure 2 – Total volume under the piston, volume in compression region of Stirling Engine, volume in expansion region of the Stirling engine plotted as a function of crank angle

3.2 Pressure Analysis

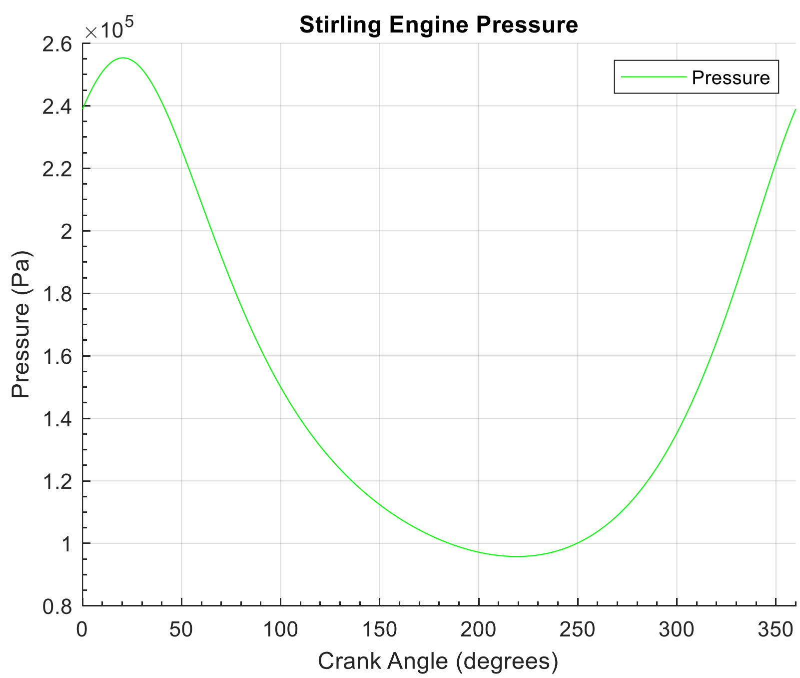


Figure 3 – Pressure in the system as a function of crank angle. The total mass of the air was needed for this step and found to be 0.0012 kg based on the minimum pressure.

3.3 Force Analysis

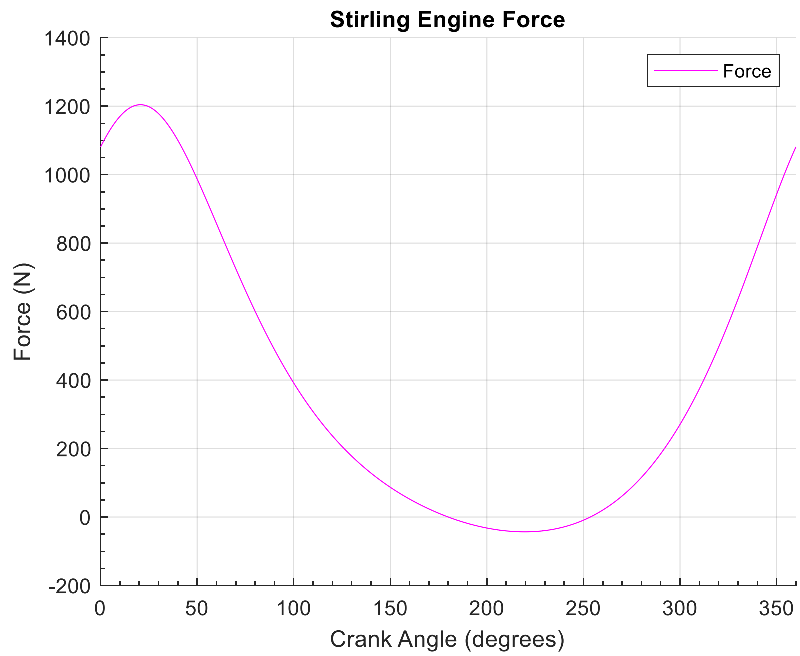


Figure 4– Force in the system as a function of crank angle

3.4 Torque Analysis

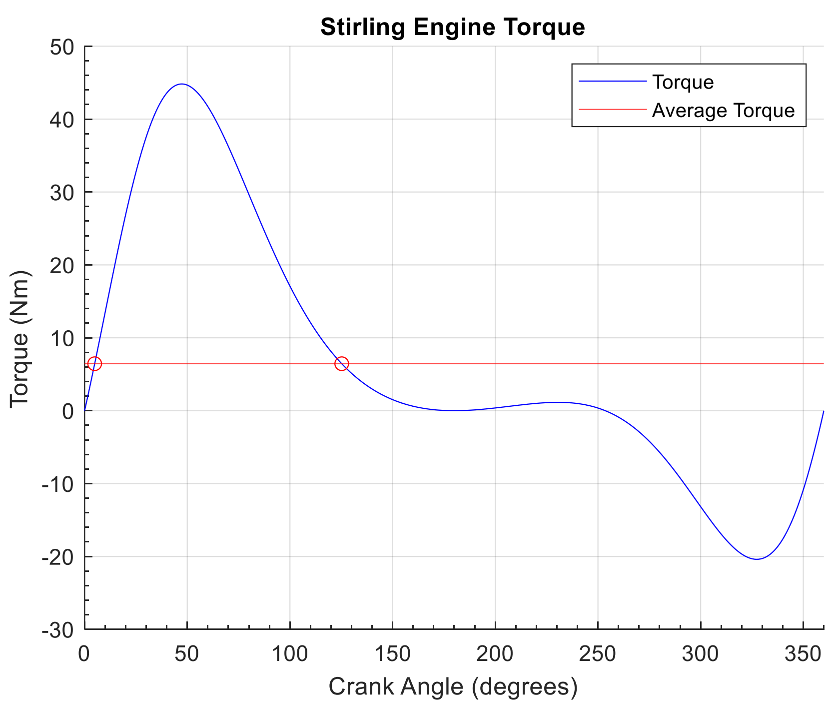


Figure 5 – Torque of the system as a function of crank angle plotted with the average torque and roots shown. The change in kinetic energy provided by the flywheel is the area under the torque curve that lies above the average line and is equal to 44.7 Joules

3.5 Cycle Analysis

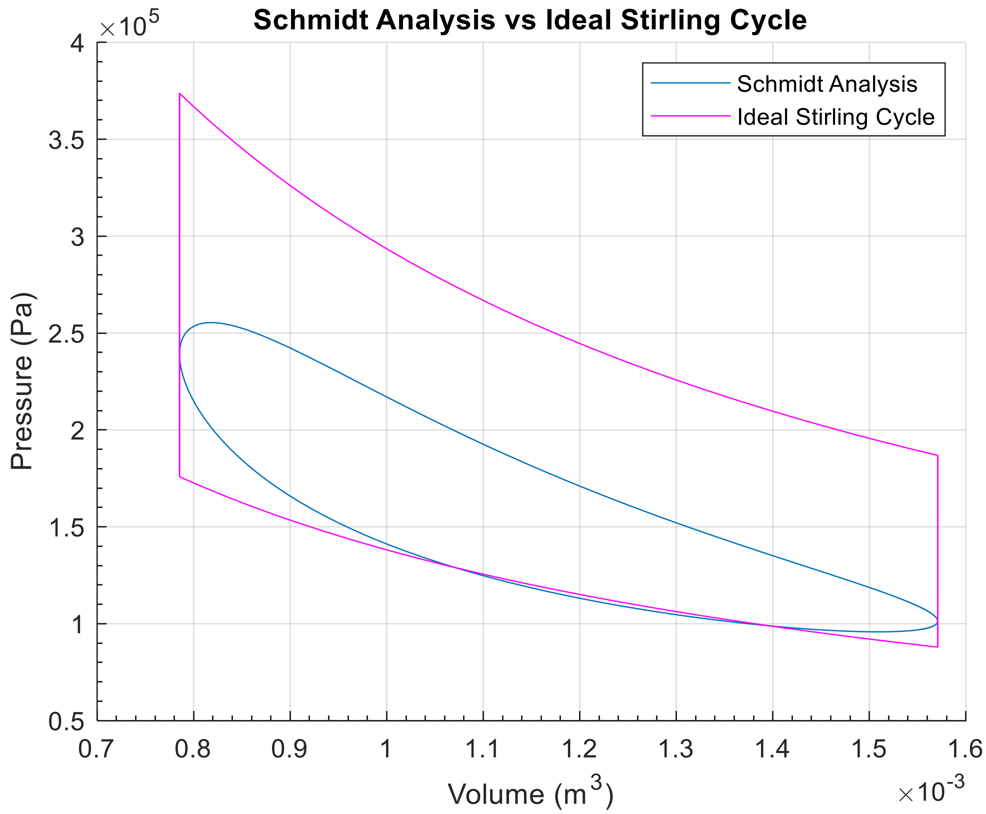


Figure 6 – Plot of the thermodynamic cycle using a Schmidt Analysis plotted alongside the Ideal Stirling Cycle. The vertical lines seen on the left and right sides of the Ideal Stirling Cycle represent isochoric cooling and heating, respectively, and the bottom and top lines of the ideal cycle represent isothermal compression and expansion, respectively.

Work (Schmidt) (J)	Work (Ideal Cycle) (J)	Stirling Cycle Efficiency
41.62	107.8	38.6%

Table 1 – Actual work, ideal work, and cycle efficiency for the Stirling Cycle.

3.6 Flywheel Design Parameters

Parameter	Value
Outer Diameter (m)	0.79
Inner Diameter (m)	0.69
Flywheel Mass (kg)	25.48
Flywheel Moment of Inertia (kg*m^2)	3.19
Average Torque Rating (N*m)	6.44
Average Power Rating (W)	554.98

Table 2 – Pressure in the system as a function of crank angle

4 Discussion

4.1 Volume Analysis

The first piece of the divide-and-conquer analysis was the volume inside the cylinder. As can be seen from Figure 1, three volumes were found. The volumes in the compression and expansion regions of the system, and the total volume under the piston. The resulting graph of the volume went along with expectations, with the total volume starting at a minimum at the top dead center position, gradually increasing as the piston expanded, then decreasing again to its initial value. Since the piston and displacer are out of phase by 90 degrees, the compression and expansion zones show similar curves that are out of phase with the crank angle. A validation step was also performed at this step to ensure that the piston volume was being calculated correctly in the MATLAB script and can be found in Appendix C.

4.2 Pressure Analysis

The pressure graph, found in Figure 3, represents the pressure in the system, which was considered to be constant throughout all regions in the lumped analysis approach to solving this problem. This value reached a peak near to where compression volume value reached a trough, which matched expectations. Since the pressure is a function of the regenerator volume, compression volume, and expansion volume, the graph shape does not precisely oppose that of the piston volume like may be seen in simpler systems.

4.3 Force Analysis

The force graph for the cycle is found in Figure 4 generally matched the shape of the pressure graph. This is because the major component of the force on the conrod depends on the pressure on the piston, though the change in the angle between the piston and the conrod (which is dependent on the crank angle) and the friction force does slightly change the shape of the output. The force on the conrod reached a maximum at approximately 1200 Newtons. Interestingly, the force on the conrod is negative for crank angles roughly between 180 and 270 degrees. This represents the part of the motion where the crankshaft begins to rise from bottom dead center and the conrod is being pulled at the piston connection instead of pushed.

4.4 Torque Analysis

As seen in Figure 5, the torque of the system varied widely both in direction and magnitude throughout the cycle and reached a max near the point of maximum compression for the compression region. This goes along with expectations, as the system is at its greatest pressure at this point and therefore can provide the largest force. In contrast, the graph shows negative torque near where the compression region was at its greatest volume and the expansion region was at its largest volume. This represents the angles at which work is being done on the system. The kinetic energy was also calculated at this step from the area between the torque curve and the average torque. This represents the energy per cycle that needs to be provided by the inertia of the flywheel to keep the system moving when operating at the average torque and came out to be 44.7 Joules. This value is a critical step in defining the design parameters for the flywheel needed for this system.

4.5 Cycle Analysis

Figure 6 shows the pressure-volume graphs for both the ideal analysis of the Stirling Cycle and the Schmidt analysis of the Stirling Cycle. The Schmidt analysis cycle plot was much smaller in size than the ideal plot. This represents the difference in work done between the two analyses, and stems from the inefficiencies that exist in the non-idealized model of the system. This efficiency difference can be quantified by comparing the work done by each of the cycles, which can be found in Table 1 and represents the area contained within the pressure-volume lines of each cycle on the graph. The efficiency of the actual cycle was found to be just 38.6%. While both the Schmidt and ideal cycles assume isothermal expansion and compression, the isochoric heating and cooling portions of the cycle are present only in the idealized cycle since the volume is always changing during heating and cooling. For the volume to stay constant during heating and cooling with a moving crankshaft, the temperature change would have to happen instantaneously, which is not realistic for this cycle. In a real-life system, efficiency would likely be even lower than what was observed in this analysis due to factors such as air-leakage, non-ideal gas behavior, and inconsistent temperatures in each region.

4.6 Flywheel Design Parameters

Using the kinetic energy found from the torque analysis, the moment of inertia for the flywheel was calculated using Equation 1. From this, all other design parameters for the flywheel were calculated. These can be found in Table 2. The inner and outer diameters of the flywheel were found to be 0.69 and 0.79 meters, respectively, and the flywheel mass was found to be 25.48 kg. These dimensions are reasonable for a medium-sized engine system, and the design could likely be implemented without difficulty. The average torque rating for the system was 6.44 Nm (found in the torque portion of the analysis), and the average power rating was 555 Watts. These ratings represent the necessary specifications of a motor that could drive the Stirling Engine (or the amount of driving power that the Stirling Engine could provide), and, along with the flywheel size, are important design parameters for a real-world implementation of this system.

5 Conclusion

In this experiment, a divide-and-conquer Schmidt analysis was performed on a Stirling Engine system with given initial constraints and parameters. From the analysis, the designed flywheel was found to have a moment of inertia of 3.19 kg*m^2, a mass of 25.48 kg, an inner diameter of 0.69 meters, and an outer diameter of 0.79 meters. The torque rating for the system was found to be 6.44 Nm and the average power rating for the system was found to be 555 Watts.

The efficiency of the system was found to be 38.6% when compared to the ideal Stirling Engine Cycle. This efficiency is likely low due to the isochoric assumptions made for the heating and cooling of the system in the ideal analysis. For a future design, greater efficiency could be achieved by experimenting with different linkage designs. Additionally, the efficiency of an alpha Stirling Engine with a similar design could be compared to this efficiency value. In a real-life application, this efficiency would likely be lower than the calculated value since the simulated analysis performed in this experiment did not account for some real-life losses such as leaking air, environmental heat loss, and inconsistent temperatures.

All goals of the experiment were met, with a reasonable flywheel design obtained that would allow the system to drive or be driven by loads that are at times greater than the average torque of a cycle. This analysis has potential future applications related to designing Stirling Engines, which have various use cases in modern day industries related to heating and cooling and (in some cases) automobiles. The techniques used in the analysis can also be applied to a variety of different engineering and design problems in the real world extending far beyond the scope of this system.

6 References

[1] Mishra, Pankaj. “What Is Stirling Engine - Types, Main Parts, Working and Application?” *Mechanical Booster*, 2 Feb. 2018, https://www.mechanicalbooster.com/2018/01/stirling-engine-types-main-parts-working-application.html.

[2] Juvinall, R. C., & Marshek, K. M. (2017). *Fundamentals of Machine Component Design*. John Wiley & Sons.

[3] *Density of Steel*, https://amesweb.info/Materials/Density_of_Steel.aspx.

APPENDICES

Appendix A – Given System Design Parameters

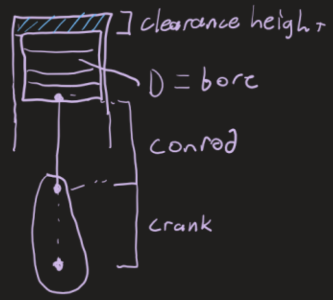
Parameter	Value
Piston Crankshaft Length (m)	0.050
Piston Conrod Length (m)	0.150
Piston Clearance (m)	0.1
Displacer Crankshaft Length (m)	0.050
Displacer Conrod Length (m)	0.24
Displacer Clearance (m)	0.01
Bore Diameter (m)	0.10
Phase Shift (degrees)	90
Hot Temperature (K)	850
Cold Temperature (K)	400
Minimum Pressure (Pa)	101300
Regenerator Dead Volume (m^3)	0.0001
Flywheel Width (m)	0.025
Flywheel Rim Thickness (m)	0.050
Coefficient of Fluctuation (AU)	0.002
Coefficient of Friciton (AU)	0.10
Angular Velocity (RPM)	800

Appendix B – Derivations

Hand calculation for gas volume

@ $\theta = 0°, 30°, \& \ 180°$

↳ sketch of piston @ 0°

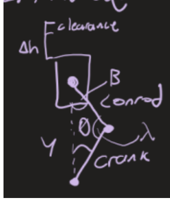


↳ need volume of shaded region

↳ cylinder-shaped region → $V_{cyl} = \frac{\pi D_{cyl}^2}{4} \cdot h$

↳ h changes based on clearance height, conrod, crank, & crank angle (θ)

↳ find equation for h



$h = \text{clearance} + \Delta h$

$\Delta h = (\text{conrod} + \text{crank} - y)$

↳ get y from law of sines/cosines

Drawing of crank-slider mechanism:

Given: θ, l_1, l_2

Find: y

Analysis: use law of cosines to solve for y

↳ $c = \sqrt{a^2 + b^2 - 2ab\cos(\gamma)}$

↳ angle opposite c

Sub our variables ↳ $y = \sqrt{l_1^2 + l_2^2 - 2 l_1 l_2 \cos(180 - \theta - \beta)}$

↳ only missing β → law of sines

↳ $\frac{l_1}{l_2} = \frac{\sin\beta}{\sin\theta} \rightarrow \beta = \sin^{-1}\left(\frac{l_1}{l_2}\sin\theta\right)$

↳ substitute for equation for y

↳ $y = \sqrt{l_1^2 + l_2^2 - 2 l_1 l_2 \cos\left(180 - \theta - \sin^{-1}\left(\frac{l_1}{l_2}\sin\theta\right)\right)}$

↳ $l_1 = \text{crank}, \ l_2 = \text{conrod}$

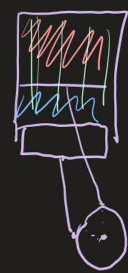
↳ $\text{Bore} = D_{cyl}$

↳ $V_{gas} = \frac{\pi \, \text{Bore}^2}{4} \cdot \left(\text{clearance} + \text{conrod} + \text{crank} - \sqrt{\text{crank}^2 + \text{conrod}^2 - 2 \cdot \text{crank} \cdot \text{conrod} \cdot \cos\left(180 - \theta - \sin^{-1}\left(\frac{\text{crank}}{\text{conrod}} \cdot \sin\theta\right)\right)}\right)$

Project 1 Equation Derivation

Thursday, October 14, 2021 11:22 PM

compression, expansion volume from volume under piston, displacor



- compression, displacer volume
- piston volume
- expansion volume

- get piston and displacer volume using volume eq

↳ $\boxed{\text{compression volume} = \text{displacer volume}}$

↳ $\boxed{\text{expansion volume} = \text{piston volume} - \text{displacer volume}}$

---

mass, pressure using ideal gas law

↳ Find air mass first

↳ $PV = mRT \rightarrow m = \frac{PV}{RT}$

$m_{air} = m_{cair} + m_{eair} + m_{rair}$

↳ $$\boxed{m_{air} = \frac{P}{R}\left(\frac{T_c}{V_c} + \frac{T_e}{V_e} + \frac{T_r}{V_r}\right)}$$

↳ solve at 180° where piston is at maximum volume (piston is TDC @ 0°)

↳ $T_c = T_{hot}$, $T_e = T_{cold}$, $T_r = \underbrace{\frac{T_{hot} + T_{cold}}{2}}_{\text{average of two areas}}$

* full calculation in sample calcs section

↳ m_{air} is constant → solve gas law eq for P

↳ $$\boxed{P = \frac{R\, m_{air}}{\left(\frac{T_c}{V_c} + \frac{T_e}{V_e} + \frac{T_r}{V_r}\right)}}$$

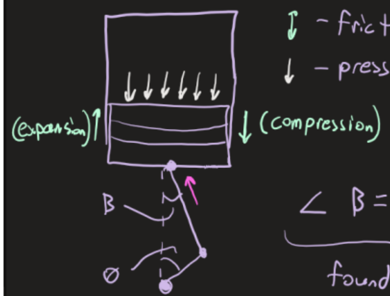
Find Force from pressure

↳ focus on piston force, draw FBD

↕ - friction force

↓ - pressure force

↑ - conrod force



$$\underbrace{\angle\ \beta = \sin^{-1}\left(\frac{l_1}{l_2}\sin\varnothing\right)}$$

found using law of sines, see volume calc for details

↳ $\Sigma F_y = 0 = F_{conrody} \pm F_\mu - F_{pressure}$

↳ $0 < \varnothing < 180$ F_μ is negative

↳ $180 < \varnothing < 360$ F_μ is positive

↳ $F_{\mu\,mag} = \mu N = \mu \cdot F_{conrodx}$

$F_{conrodx} = F_{conrod} \cdot \sin(\beta)$

determined intuitively

$F_{conrody} = F_{conrod} \cdot \cos(\beta)$

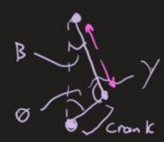
$F_{pressure} = P \cdot A$ $A = \frac{\pi \cdot D_{bore}^2}{4}$

↳ $0 = F_{conrod} \cdot \cos(\beta) \pm \mu F_{conrod} \sin(\beta) - P \cdot A$

↳ $$\boxed{F_{conrod} = \frac{P \cdot A}{\cos(\beta) \pm \mu \sin(\beta)}}$$

use sign(sin(∅)) to get "±" on friction force in MATLAB

Find Torque from Force

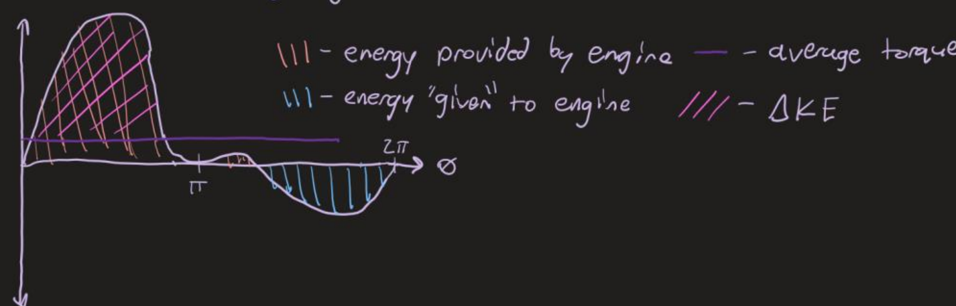


$\gamma = 180 - \beta - \emptyset$

when $\gamma = 90$, torque is at its max

↳ $\boxed{\text{Torque} = \text{crank} \cdot \text{force} \cdot \sin(\gamma)}$

ΔKE from torque graph



↳ energy is area under T – ø curve

↳ need ΔKE, or energy needed from the flywheel to provide necessary torque to system to function

↳ ΔKE corresponds to torque greater than the average torque of the system

↳ average torque from mean-value theorem

↳ $\underbrace{\frac{1}{2\pi}}_{\text{range of theta values}} \cdot \underbrace{\int_0^{2\pi} T \, d\emptyset}_{\text{total energy under curve}}$

↳ Use MATLAB to solve integrals/roots (trapz for integrals, for loop for roots in my code)

Find J needed to provide ΔKE

↳ $KE = \int_{w_o}^{w_f} J w \, dw \rightarrow KE = \frac{J(w_f^2 - w_o^2)}{2}$

↳ $\Delta KE_{flywheel} = -KE$

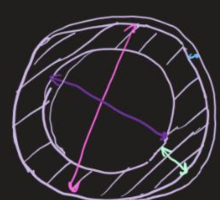
$C_f = \frac{w_{max} - w_{min}}{w_{ave}} \rightarrow w_{max} = w_o, \; w_{min} = w_f$

↳ $\Delta KE = \frac{J(w_o^2 - w_f^2)}{2} \rightarrow \frac{J(w_{max}^2 - w_{min}^2)}{2} = J(w_{max} - w_{min})\underbrace{\frac{(w_{min} + w_{max})}{2}}_{= w_{ave}}$

↳ $\Delta KE = J(w_{max} - w_{min}) \cdot w_{ave} \rightarrow \Delta KE = J w_{ave}^2 C_f$

↳ $\boxed{J = \frac{\Delta KE}{w_{ave}^2 C_f}}$

Find m, D_o, D_i from J



Flywheel

Known:

t, w, J, ρ_{steel}

Find:

D_o, D_i

↳ For cylindrical shell

$J = \frac{1}{2} m (r_o^2 + r_i^2) = \frac{1}{8} m (D_o^2 + D_i^2)$

↳ need m of disk

↳ $\rho = \frac{m}{V} \rightarrow m = V \cdot \rho_{steel}$

↳ $V = \underbrace{V_{outer} - V_{inner}}_{\text{for cylinder}} = w \cdot \left(\frac{\pi D_o^2}{4} - \frac{\pi D_i^2}{4}\right)$

↳ $D_i = D_o - 2t$

$$\boxed{m = \frac{\rho_{steel}\,\pi}{4}\left(D_o^2 - (D_o - 2t)^2\right)}$$

$$\boxed{J = \tfrac{1}{8} m\left(D_o^2 + D_i^2\right) = \frac{\rho_{steel}\,\pi}{32}\left(D_o^4 - (D_o - 2t)^4\right)}$$

Average Power

- $P = \frac{W}{t}$, $W = \int P\,dV$
- already have graph of PdV, use trapz to get area under curve → $W = \mathrm{trapz}(V_{piston}, \mathrm{Pressure})$
- need t for one cycle → $t_{cycle} = T$
- $T = \frac{1}{rpm} \cdot \frac{60\ s}{m} = \frac{60}{\omega_{(rpm)}}$
- $P = \frac{\mathrm{trapz}(V_{piston}, \mathrm{pressure})}{(60/\omega_{(rpm)})}$

Isothermal (ideal) pressure, work, efficiency

- $1 = \frac{mRT}{PV}$, $1 = \frac{mRT_i}{P_iV_i}$, $T_i = T$
- $\frac{\cancel{mRT}}{PV} = \frac{\cancel{mRT_i}}{P_iV_i} \rightarrow \boxed{P = \frac{V}{P_iV_i}}$

Ideal Work → use trapz to find area between ideal pressure lines

- $W_{ideal} = \mathrm{trapz}(V_{ideal}, P_{comp}) - \mathrm{trapz}(V_{ideal}, P_{exp})$

$$\boxed{\eta_{stirling} = \frac{W_{schmidt}}{W_{ideal}} \cdot 100}$$

Appendix C – Validation Calculation

To verify the validity of the MATLAB code, a validation calculation was done for three different total piston gas volumes for angles of 0, 30, and 180 degrees:

$$\hookrightarrow V_{gas} = \frac{\pi \, Bore^2}{4} \cdot \left(clearance + conrod + crank - \sqrt{crank^2 + conrod^2 - 2 \cdot crank \cdot conrod \cdot \cos\left(180 - \theta - \sin^{-1}\left(\frac{crank}{conrod} \cdot \sin\theta\right)\right)} \right)$$

Solve manually for crank = 0.050 m, conrod = 0.150 m, Bore = 0.1 m, clearance = 0.02 m, $\theta = 0, 30, 180$

$\hookrightarrow V_{gas}(0) \approx 1.6 \cdot 10^{-4}\ m^3$ (rounded to 2 significant figures, assumes Bore, clearance have 2 sig figs of accuracy)

$V_{gas}(30) \approx 2.3 \cdot 10^{-4}\ m^3$ " "

$V_{gas}(180) \approx 9.4 \cdot 10^{-4}\ m^3$ " "

These values matched that of the MATLAB output for the getVolume function:

```
>> getVolume(0,0.050,0.150,0.1,0.02)

ans =

   1.5708e-04

>> getVolume(30,0.050,0.150,0.1,0.02)

ans =

   2.2617e-04

>> getVolume(180,0.050,0.150,0.1,0.02)

ans =

   9.4248e-04
```

This validates that the volume function is working correctly.

Appendix D – MATLAB Code

fly.m

```
% Flywheel Calculation Script
% ME4053 Project 1
```

```
% Author: Ryker Zierden
% Date: 10 12 2021

%% Initialization
%Initialize this run
clear
clc

%initialize crank angles, NOTE: 0 degrees is TDC
piston_crank_angle = 0:0.01:360;
displacer_crank_angle = 90:0.01:450;
%% Design parameters

% Piston
piston_crankshaft = 0.050; %m
piston_conrod = 0.150; %m
piston_clearance = 0.1; %m

% Displacer
disp_crankshaft = 0.050; %m
disp_conrod = 0.24; %m
disp_clearance = 0.01; %m

% Other parameters
bore_diameter = 0.10; %m
phase_shift = pi/2; %radians
T_high = 850; %K
T_low = 400; %K
P_min = 101300; %Pa, BDC
regen_dead_volume = 0.0001; %m^3
flywheel_width = 0.025; %m
flywheel_rim_thickness = 0.050; %m
Cf = 0.002; %AU
mu = 0.10; %AU
angular_velocity = 800; %RPM

%% Get Volumes
% use getVolume function to get the volume across all angles for piston and
% displacer
piston_volume =
getVolume(piston_crank_angle,piston_crankshaft,piston_conrod,bore_diameter,piston_clearance);
displacer_volume =
getVolume(displacer_crank_angle,disp_crankshaft,disp_conrod,bore_diameter,disp_clearance);
% the displacer volume represents the volume if the piston was not there, we
% want the volume between the piston and displacer -> subtract the two
compression_volume = piston_volume - displacer_volume; %m^3
% expansion volume is just displacer volume, but it will work more intuitively
% if the variable is named differently
expansion_volume = displacer_volume; %m^3
% generate volume plot
hold on
```

```
plot(piston_crank_angle,piston_volume,"m-")
plot(piston_crank_angle,compression_volume,"r-")
plot(piston_crank_angle,expansion_volume,"b-")
legend("Total Volume","Compression Volume", "Expansion Volume")
grid on
xlabel("Crank Angle (degrees)")
ylabel("Volume (m^3)")
xlim([0 360])
title("Stirling Engine Volume")
hold off
%% Get Pressure
% to get the pressure, we first must find the mass -> ideal gas law
% assumption analysis at BDC position (assume mass is constant
% throughout cycle)PV = mRT -> m = PV/RT, assume pressure is constant
% across all three areas. mtotal = (P/R)(Vc/Tc + Ve/Te + Vr/Tr)
R_air = 287; %J/kgK
% find T_regen, needed for mass calculation
T_regen = (T_high + T_low)/2;
% find m_total, 100 data points taken per degree of rotation -> index =
% theta * 100. BDC is 180 for piston, comp volume in terms of piston angle
% when indexing
m_total = (P_min/R_air) * (compression_volume(round(size(compression_volume,2)/2))/T_low +
expansion_volume(round(size(expansion_volume,2)/2))/T_high +regen_dead_volume/T_regen); %kg
% assuming mass, temperature is constant, P = mRT/V = mR(Tc/Vc  + Te/Ve +
% Vr/Tr)
regen_dead_volume_matrix = (regen_dead_volume.*ones(1,size(expansion_volume,2)));
pressure = R_air * m_total./(compression_volume./T_low + expansion_volume./T_high +
(regen_dead_volume_matrix)./T_regen);
% generate pressure plot
figure()
hold on
plot(piston_crank_angle,pressure,"g-")
legend("Pressure")
grid on
xlabel("Crank Angle (degrees)")
ylabel("Pressure (Pa)")
xlim([0 360])
title("Stirling Engine Pressure")
legend("Pressure")
hold off

%% Get Force
% run the force function on the crank angle and pressure to get
% the force on the conrod throughout the entire motion of the mechanism
force = getForce(piston_crank_angle,piston_crankshaft,piston_conrod,bore_diameter,pressure,mu);
% generate force plot
figure()
hold on
plot(piston_crank_angle,force,"m-")
legend("Force")
grid on
```

```
xlabel("Crank Angle (degrees)")
ylabel("Force (N)")
xlim([0 360])
title("Stirling Engine Force")
legend("Force")
hold off

%% Get Torque
% run the torque function on the crank angle and force to get the torque on
% the motor throughout the motion of the mechanism

torque = getTorque(piston_crank_angle,piston_crankshaft,piston_conrod,force);%Nm



%% Get Change in Kinetic Energy (Above Average)
% the area under the torque/crank angle curve represents energy provided by
% the crank shaft. By finding the average, we can see where the energy is that we
% need to restore to the system with the inertia of the flywheel


% calculate average torque using mvt

average_torque = (1/360) * trapz(piston_crank_angle,torque) %Nm


% now get the roots using the getRoots function, which is a custom function
% that avoids the need to use fzero and thus circumvents the need to
% get the function feature to work in MATLAB (which is quite tedious most
% of the time)
roots = getRoots(piston_crank_angle, torque, average_torque); %Nm
root1 = roots(1);
root2 = roots(2);

% now that we have the roots, we can use the mean value theorem to once
% again to find the area over the average torque, which is the energy that
% must be supplied by the flywheel

% get subset of crank angles and torque (NOTE: there are steps here to make
% the step changeable at the top of the function for more or less precision
% that make this messier than it has to be. The result is the ability to
% change the precision/number of angles at will/even use negative steps if
% desired in a later use case)
angle_step = abs(piston_crank_angle(2) - piston_crank_angle(1));
root_crank_angles = deg2rad(root1):deg2rad(angle_step):deg2rad(root2);
root_torque = torque((root1)*(1/angle_step)+1:(root2)*(1/angle_step)+1);

% generate plot of torque with average torque and roots shown
figure()
hold on
plot (piston_crank_angle,torque,"b-")
```

```
yline(average_torque,"r-")
plot ([root1 root2], average_torque,"ro")
grid on
xlabel("Crank Angle (degrees)")
ylabel("Torque (Nm)")
xlim([0 360])
title("Stirling Engine Torque")
legend("Torque","Average Torque")
hold off
% plot(root_crank_angles, (root_torque))
% use mean value theorum to find delta KE
delta_KE = trapz(root_crank_angles,root_torque-average_torque); % J

%% Find J from KE/theta
% J = KE/(w^2 * Cf)
% need to convert average angular velocity to radians
omega_average = angular_velocity * 1/60 * 2 * pi; % rad/s

% solve for J
J = delta_KE/(omega_average^2 * Cf) % kg * m^2

%% Find Do, Di, m_flywheel from J
% the flywheel will be made of steel, so we'll need to get the density of
% steel for this calculation
steel_density = 8050;  % kg/m^3

% Do has a fairly complicated equation -> solve with fzero
% combine constants
c = (steel_density * flywheel_width/32)*pi;
% define a function to find the zeros of
Do_func = @(Do)(c * (Do^4 - (Do - 2*flywheel_rim_thickness)^4)-J);
% call fzero to find Do
Do = fzero(Do_func, 0)
Di = Do - 2*flywheel_rim_thickness
% calculate m_flywheel using volume * density
% m = volume*density
m_flywheel = (Do^4 - (Do - 2*flywheel_rim_thickness)^4)* (flywheel_width/4)*pi*steel_density

%% Find Average Power
% P = integral under pdV curve/time -> use trapz with piston_volume and pressure
% to find work then divide by time for one rotation

period = 1/800 * 60; % seconds/cycle
work = trapz(piston_volume,pressure); %J
average_power = work/period %W


%% Plot PdV
figure()
hold on
 plot(piston_volume,pressure) % plot the actual curve (Schmidt analysis)
```

```
% find minimum and maximum volumes from piston_volume array
max_volume = max(piston_volume);
min_volume = min(piston_volume);

% use ideal gas law to get max pressure from minimum volume
max_pressure = m_total*R_air*T_high/min_volume;
min_pressure = m_total*R_air*T_low/max_volume;
% set ideal_volume_range based on min and max values
ideal_volume = round(min_volume,6):0.000001:round(max_volume,6);
% calculate the isothermal pressure for expansion and compression using
% isothermal function
ideal_pressure_expansion = isothermal(min_pressure,max_volume,ideal_volume);
ideal_pressure_compression = isothermal(max_pressure,min_volume,ideal_volume);
% plot the ideal stirling cycle, set plot values
plot(ideal_volume,ideal_pressure_expansion,"m-","DisplayName","isothermal expansion")
plot(ideal_volume,ideal_pressure_compression,"m-","DisplayName","isothermal compression")

plot(min_volume * ones(2), [max(ideal_pressure_expansion) max_pressure],"m-
","DisplayName","isochoric cooling")
plot(max_volume * ones(2), [min_pressure min(ideal_pressure_compression)],"m-","DisplayName",
"isochoric heating")

legend("Force")
grid on
xlabel("Volume (m^3)")
ylabel("Pressure (Pa)")
title("Schmidt Analysis vs Ideal Stirling Cycle")
legend("Schmidt Analysis","Ideal Stirling Cycle")
hold off
% finally, calculate the ideal work of the cycle using trapz

ideal_work = trapz(ideal_volume,ideal_pressure_compression) -
trapz(ideal_volume,ideal_pressure_expansion);
```

getVolume.m

```
function Vol = getVolume(crankAngle,crank,conrod,bore,clearanceHeight)

% Calculate the volume of the gas contained in a piston cylinder device

% as a function of the crank angle.

%

% Note: The zero crank angle reference position is the

% same as the piston top dead center (TDC) position, and

% the crank rotates in the clockwise direction.
```

```
% Author: RJZ

% Date: 2021.09.27

%

% Key Variables:

% Provided in function parameters - crankAngle (theta), crank (length of
% the crank), conrod (length of the connecting rod, bore (diameter of the
% piston), clearanceHeight (distance between top of piston and top of
% enclosure at TDC position)

% calculated intermediate values - y (distance between ground-crank
% connection and the conrod-piston connection)



%% Calculate the gas volume

% convert crank angle to radians, store in theta
theta = pi*crankAngle/180;
% calculate y
y = sqrt(crank^2 + conrod^2 - 2*crank*conrod*cos(pi-theta-asin((crank/conrod)*sin(theta))));
% calculate and return the volume of the cylinder
Vol = (pi*bore^2/4)*(clearanceHeight + crank + conrod - y);
```

getForce.m

```
function Force = getForce(crankAngle,crank,conrod,bore,pressure,mu)

% Calculate the force on the connecting rod of the piston cylinder

% as a function of the crank angle and pressure.

%

% Note: The zero crank angle reference position is the

% same as the piston top dead center (TDC) position, and

% the crank rotates in the clockwise direction.

% Author: RJZ

% Date: 2021.10.13
% define atmospheric pressure to account for relative pressure in piston
P_atm = 101300; %Pa

%% Calculate the Force
```

```
% convert crank angle to radians, store in theta
theta = pi*crankAngle/180;
% calculate beta, the third angle of the triangle made between height and
% the two links connecting the motor to the piston
beta = asin((crank/conrod).*sin(theta)); %radians
bore_area = pi * bore^2/4; %m^2
% find the pressure force, accounting for the fact that we need relative
% pressure
pressure_force = (pressure - P_atm).*bore_area; %N
% find the conrod_force obtained from force analysis in y direction
conrod_force = pressure_force./((cos(beta))+sign(sin(theta)).*mu.*sin(beta));

Force = conrod_force; % N
```

getTorque.m

```
function Torque = getTorque(crankAngle,crank,conrod,force)

% Calculate the torque on the connecting rod of the piston cylinder

% as a function of the crank angle and force.

%

% Note: The zero crank angle reference position is the

% same as the piston top dead center (TDC) position, and

% the crank rotates in the clockwise direction.

% Author: RJZ

% Date: 2021.10.13


%% Calculate the Torque

% convert crank angle to radians, store in theta
theta = pi*crankAngle/180;
% calculate beta, the third angle of the triangle made between height and
% the two links connecting the motor to the piston
beta = asin((crank/conrod).*sin(theta)); %radians
% calculate gamma, the angle between the conrod and crank
gamma = pi - theta - beta;
% T = F*r where F is the component of the conrod force that is normal to
% the crank
torque = force.*sin(gamma)*crank;

Torque = torque;
```

getRoots.m

```
function Roots = getRoots(piston_crank_angle,torque, average_torque)

% Author: RJZ

% Date: 2021.09.27

% the following for loop finds the exact roots of the function. This is
% to avoid the sometimes quite tedious syntax needed to get fzero to work
% properly (even though it's more lines of code)
minTheta1 = 0;
minTheta2 = 0;
torqueSize = size(torque,2);
for i = 1:torqueSize
    % set lastTorque variable for comparison as well as initialize our
    % minimum difference values (only on first loop through)
    if i < 2
      lastTorque = torque(i);
      minDiff1 = abs(average_torque - torque(i));
      minDiff2 = minDiff1;
    else
      lastTorque = torque(i - 1);
    end
    % calculate the current distance between the torque and the average
    % torque
    currDiff = abs(torque(i) - average_torque);
    % compare the current distance with the minimum difference with
    % separate cases for when the curve is climbing vs falling (to obtain
    % both roots). If this value is the current minimum, write the theta
    % value to the minTheta variable and update the minDiff variable for
    % this case
    if torque(i) - lastTorque > 0
        if (currDiff < minDiff1)
            minDiff1 = currDiff;
            minTheta1 = piston_crank_angle(i);
        end
    else
        if currDiff < minDiff2
            minDiff2 = currDiff;
            minTheta2 = piston_crank_angle(i);
        end
    end
end

Roots = [minTheta1 minTheta2];
```

isothermal.m

```
% Calculates the pressure for an isobaric process given the volume and intial
% conditions
%
```

```
% Date: 10/6/2021
% Author: Ryker Zierden (in collaboration with in-class group 5)


function pressure = isothermal(pressure_initial, volume_initial, volume)

hold = pressure_initial*volume_initial;

pressure = hold ./ volume;

end
```