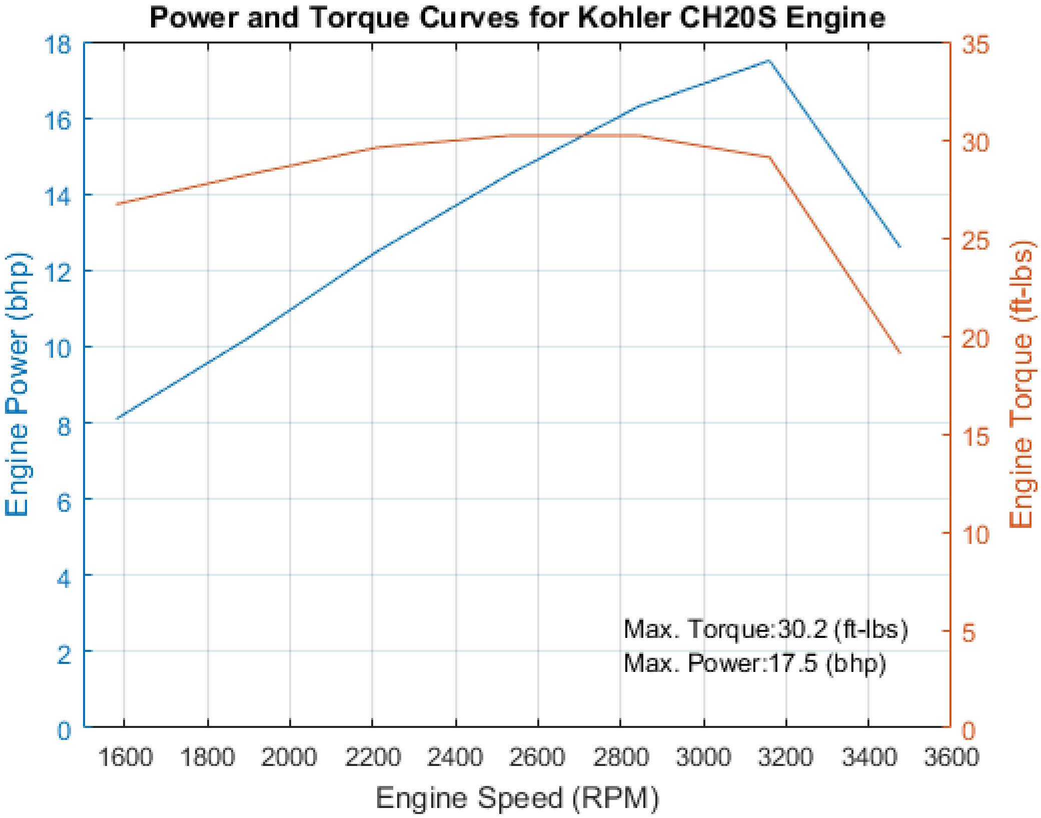

Figure 4 – The power band and torque curve of the Kohler engine. Power and torque both drop dramatically after 3,200 RPM. The ideal operating range of the engine is between 2,600 RPM and 3,200 RPM.

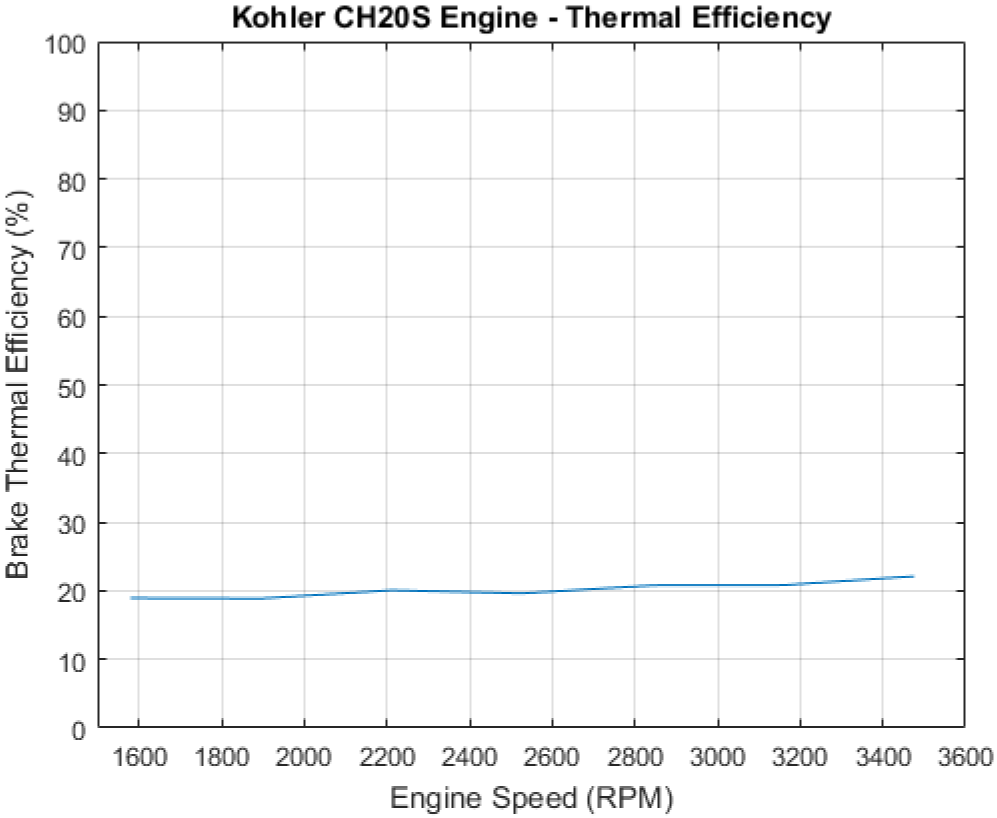

Figure 5 – The thermal efficiency of the engine is around 20% on average. Higher engine speed increases the efficiency slightly.

CODE:

```
%% Header

% Lab 5

% Simon Popecki

% 1 April 2017

% Note that the recorded RPMs are the results of a gear system - 1000 RPM

% is really 2,528 RPM (factore of 3.16)


clear all;

close all;

%% Engine Information

displacement = 38.1; %engine displacement, ci (two cylinder engine)

l = 4.55; %connecting rod length, in

bore = 3.03; %cylinder bore, inches

stroke = 2.64; %piston stroke, inches

compressionRatio = 8.5; %[NUM]:1, compression ratio of the engine

cylNum = 2; %Number of cylinders


%% Deliverable 1 Part a


% Optical sensor output voltage vs. time, crank angle (in radians) vs. time,

% Chamber Volume vs. time, and pressure vs. time for five cycles in four
% stacked plots for a single engine speed. b


%importing structured arrays
nheaderlines = 22; %Data starts on line 23
fileName = '800rpm.lvm';
impStruct = importdata(fileName,'¥t',nheaderlines); %data for the balloon with long tube
impData = impStruct.data; %taking only the shit we care about
time = impData(:,1); %time in column three is the same
% time = time(1:2000); %we are only looking at 5 cycles
time = time-(time(1));


pDataV = impData(:,2); %Pressure sensor data, in voltage
% pDataV = pDataV(1:2000); %taking only first 5 cycles
pData = pDataV/.0104; %converting pressure readings to psi
oData = impData(:,4); %Optical sensor voltage data
% oData = oData(1:2000); %taking only the first 5 cycles


[pks,~,TDCs,~] = peaks(time,oData,1); %finding the peak locations of TDC
% TDC = TDCs(1); %defining the first spike as TDC
% tp1 = TDCs(1); %defining the second spike as tp1 (see ICE Lab overview.pdf)
% tp2 = TDCs(2); %defining the third spike as tp2
[~,~,cycleStarts,~] = peakdet(oData,1); %cycleStarts is referenced later in the program
```

```
crankAngle = zeros(length(time),1);

for i = 1:1:20

    for j = cycleStarts(i):cycleStarts(i+1)-1

        tp2 = time(cycleStarts(i+1)-1);

        tp1 = time(cycleStarts(i));

        crankAngle(j) = 2*pi*(1-((tp2-time(j))/(tp2-tp1)));


    end

end


Vd = displacement/cylNum; %the volume of the combustion chamber at BDC


Vc = Vd/(compressionRatio-1); %clearance volume (cubic inches)

% - the volume of the cylinder when the piston is at TDC


a = stroke/2; %inches, finding the radius of crankshaft


R = l/a; %ratio of the length of the connecting rod to the radius of the crankshaft


cylinderVolume = Vc.*(1+.5.*(compressionRatio-1).*(R+1-cos(crankAngle)-((R.^2)-
(sin(crankAngle).^2)).^.5));


time = time(cycleStarts(1):cycleStarts(20));
time = time-time(1); %rezeroing the new time
pData = pData(cycleStarts(1):cycleStarts(20));
oData = oData(cycleStarts(1):cycleStarts(20));
cylinderVolume = cylinderVolume(cycleStarts(1):cycleStarts(20));
crankAngle = crankAngle(cycleStarts(1):cycleStarts(20));

figure(1)
subplot(4,1,1)
plot(time,oData)
title('Engine Data: 2,528 RPM')
legend('Optical Sensor Voltage')
xlabel('Time (s)')
ylabel('Voltage (V)')
grid on
xmin = 0;
xmax = .2;
ymin = 0;
ymax = 4.5;
axis ([xmin xmax ymin ymax])
text(.03*xmax,.9*ymax,'Peaks Indicate TDC')


subplot(4,1,2)
```

```
plot(time,crankAngle)

legend('Crank Angle')

xlabel('Time (s)')

ylabel('Crank Angle (rad)')

grid on

xmin = 0;

xmax = .2;

ymin = 0;

ymax = 7;

axis ([xmin xmax ymin ymax])


subplot(4,1,3)

plot(time,cylinderVolume)

legend('Chamber Volume')

xlabel('Time (s)')

ylabel('Chamber Volume (ci)')

grid on

xmin = 0;

xmax = .2;

ymin = 0;

ymax = 25;

axis ([xmin xmax ymin ymax])
```

```
subplot(4,1,4)
plot(time,pData)
legend('Chamber Pressure')
xlabel('Time (s)')
ylabel('Pressure (psi)')
grid on
xmin = 0;
xmax = .2;
ymin = -100;
ymax = 550;
axis ([xmin xmax ymin ymax])

%% Deliverable 1 Part b

startIndex = cycleStarts(1); %TDC of the beginning of a cycle
endIndex = cycleStarts(3); %TDC of the next cycle
sctime = time(startIndex:endIndex); %a new time array looking at only a single cycle
sctime = sctime - sctime(1); %rezeroing time to begin at the first TDC

%Corresponding values of interest for this new time interval
sccrankAngle = crankAngle(startIndex:endIndex);
scoData = oData(startIndex:endIndex);
sccylinderVolume = cylinderVolume(startIndex:endIndex);
```

```
scpData = pData(startIndex:endIndex);


% figure(2)

% subplot(4,1,1)

% plot(sctime,scoData)

% title('Single Cycle Data: 2,528 RPM')

% legend('Optical Sensor Voltage','location','southeast')

% xlabel('Time (s)')

% ylabel('Voltage (V)')

% grid on

% xmin = 0;

% xmax = .039;

% ymin = 0;

% ymax = 4.5;

% axis ([xmin xmax ymin ymax])

% text(.03*xmax,.9*ymax,'Peaks Indicate TDC')

%

%

% subplot(4,1,2)

% plot(sctime,sccrankAngle)

% legend('Crank Angle','location','southeast')

% xlabel('Time (s)')

% ylabel('Crank Angle (rad)')
% grid on
% xmin = 0;
% xmax = .039;
% ymin = 0;
% ymax = 7;
% axis ([xmin xmax ymin ymax])
%
% subplot(4,1,3)
% plot(sctime,sccylinderVolume)
% legend('Chamber Volume','location','southeast')
% xlabel('Time (s)')
% ylabel('Chamber Volume (ci)')
% grid on
% xmin = 0;
% xmax = .039;
% ymin = 0;
% ymax = 25;
% axis ([xmin xmax ymin ymax])
%
% subplot(4,1,4)
% plot(sctime,scpData)
% legend('Chamber Pressure','location','southeast')
% xlabel('Time (s)')
% ylabel('Pressure (psi)')
% grid on
% xmin = 0;
% xmax = .039;
% ymin = -100;
% ymax = 550;
% axis ([xmin xmax ymin ymax])

%plotting the P-V diagram for the combustion chamber
figure(3)
plot(sccylinderVolume,scpData)
title('Representative Single Cycle at 2,528 RPM')
xlabel('Chamber Volume (ci)')
ylabel('Chamber Pressure (psi)')
grid on

%% Deliverable 1 Part c

%plotting all the P-V diagrams on top of each other to get the average

figure(4)
plot(cylinderVolume,pData)
title('All Cycles at 2,528 RPM')
```

```
xlabel('Chamber Volume (ci)')
ylabel('Chamber Pressure (psi)')
grid on

%% Deliverable 2
[avgW400,avgP400,stdW400,stdP400] = del2('400rpm.lvm');
[avgW500,avgP500,stdW500,stdP500] = del2('500rpm.lvm');
[avgW600,avgP600,stdW600,stdP600] = del2('600rpm.lvm');
[avgW700,avgP700,stdW700,stdP700] = del2('700rpm.lvm');
[avgW800,avgP800,stdW800,stdP800] = del2('800rpm.lvm');
[avgW900,avgP900,stdW900,stdP900] = del2('900rpm.lvm');
[avgW1000,avgP1000,stdW1000,stdP1000] = del2('1000rpm.lvm');
[avgW1100,avgP1100,stdW1100,stdP1100] = del2('1100rpm.lvm');



function [avgWork,avgPower,stdWork,stdPower] = del2(fileName)

nheaderlines = 22; %Data starts on line 23
impStruct = importdata(fileName,'¥t',nheaderlines); %data for the balloon with long tube
impData = impStruct.data; %taking only the shit we care about
time = impData(:,1); %time in column three is the same
time = time-(time(1));
pDataV = impData(:,2); %Pressure sensor data, in voltage
```

```
pData = pDataV/.0104; %converting pressure readings to psi

oData = impData(:,4); %Optical sensor voltage data

[~,~,TDCs,~] = peaks(time,oData,1); %finding the peak locations of TDC

TDC = TDCs(1); %defining the first spike as TDC

tp1 = TDCs(1); %defining the second spike as tp1 (see ICE Lab overview.pdf)

tp2 = TDCs(2); %defining the third spike as tp2

[~,~,cycleStarts,~] = peakdet(oData,1); %cycleStarts is referenced later in the program


crankAngle = zeros(length(time),1);

for i = 1:1:20

    for j = cycleStarts(i):cycleStarts(i+1)-1

        tp2 = time(cycleStarts(i+1)-1);

        tp1 = time(cycleStarts(i));

        crankAngle(j) = 2*pi*((tp2-time(j))/(tp2-tp1));


    end

end


function [crankAngle] = crank(i,time,cycleStarts)

    for j = cycleStarts(i):cycleStarts(i+1)-1

        tp2 = time(cycleStarts(i+1)-1);

        tp1 = time(cycleStarts(i));

        crankAngle(j) = 2*pi*((tp2-time(j))/(tp2-tp1));
    end
end
```