Table of Contents

Project 2 Analysis

This script is the main controller for the SR30 analysis in Project 2 of ME140. It should import data, and call the component functions for each of the individual parts of the project.

Authors: Jean-Christophe Perrin, Beck Goodloe, Richard Randall, Jason Trinidad

Created: 2018-04-11 Edited: 2018-04-11

```
clear all;
clc;
```

Usefull Constants

These are mostly conversion factors so that we can convert from imperial units collected into metric.

```
const.insqToMsq = 0.00064516; % [m^2/in^2]
const.lbfToN = 4.44822; % [N/lbf]
const.R = 287; % [kJ/kg/k] Gas constant of air
const.KCdiff = 273; % [deg]
const.kPaToPa = 1e3; % [Pa/kPa]
const.kJkmol2Jkg = 1e3/28.97; %[J/kg * kmol/kJ]
const.airCoefs = const.kJkmol2Jkg .* ...
    fliplr([28.11, 0.1967e-2, 0.4802e-5,-1.966e-9]); % [J/kg/k] cp air
```

Given Data

The following values were supplied to us in the original project specifications and should not be changed.

```
given.pitotEffectiveArea = 6.4*const.insqToMsq; % [m^2]
given.jetAHeating = 42.8e6; % [J/kg/K]
given.A = [27.3, 6.4, 9.0, 7.2, 4.7, 3.87]'.*const.insqToMsq; % [m^2]
```

Import Collected Data

We import the collected data into an table for use throughout our function. This table should never be changed, compromising the validity of our collected data. Instead, all deried values should be copied out the table into a seperate vector or table.

```
fname = 'data.txt';
```

```
collectedData = readtable(fname);
collectedData.Properties.VariableUnits =
 {'', 'C', 'C', 'C', 'C', 'C', ...
    'C', 'kPa', 'kPa', 'kPa', 'kPa', 'kPa', 'kg/s', 'lbs'};

nObservations = height(collectedData); % Number of observations
% disp(collectedData);
```

Warning: Variable names were modified to make them valid MATLAB identifiers. The original names are saved in the VariableDescriptions property.

We also measured the following values seperately.

```
T1 = 23; % [C] room temp
P1 = 101.4e3; % [Pa] atmospheric pressue
```

Plots

Use your performance data and area measurements (listed in Appendix) to construct a series of plots showing how the following quantities vary with spool speed:

- station stagnation temperature (K),
- station stagnation pressure (kPa, absolute),
- station Mach number
- station velocity (m/s).

Make sure to include Station 1 in all of your plots.

```
krpm = collectedData.RPM/1000;
```

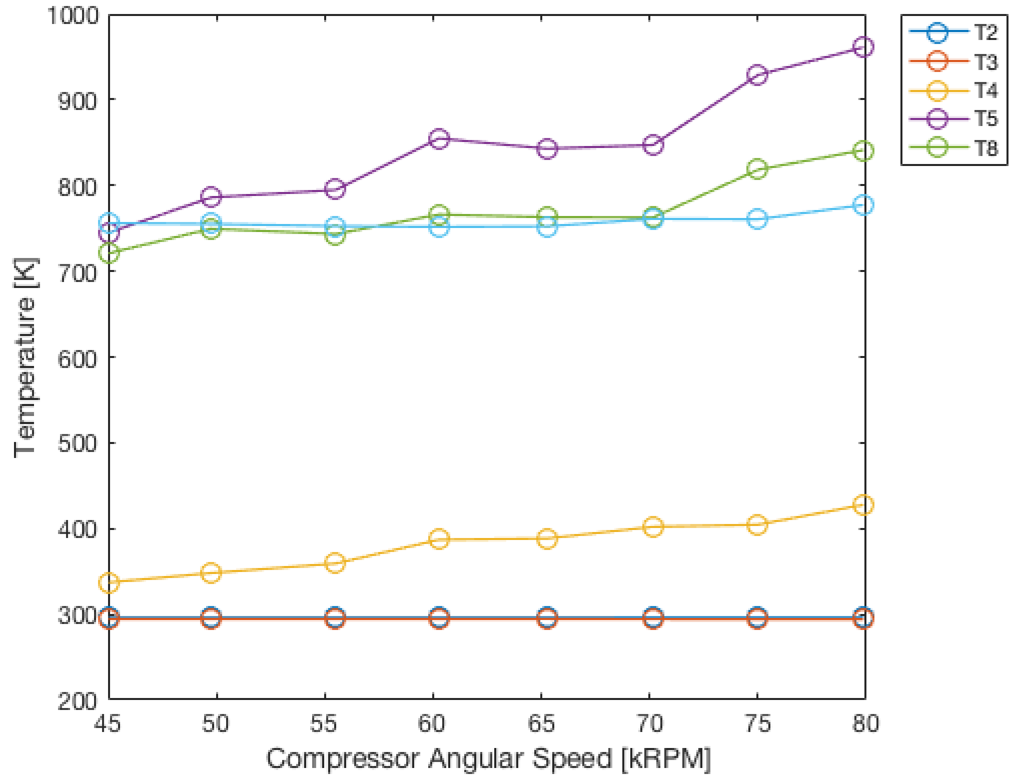
Stagnation Temperature v. RPM

Let's collect all of the stagnation temperatures that we collect into a table. We can slice the variables of interest out of the table we imported earlier. However, we are still missing T1. This is the same for all trials -- the ambient air temperature of the lab. We generate a vector of this measurement (one for each trial) and collected all measurements in an array.

```
vars = {'T2', 'T3', 'T4', 'T5', 'T8'};
stagTemps = collectedData{:,vars} + const.KCdiff; % [K]
T1_vec = T1 * ones(nObservations, 1) + const.KCdiff; % [K]
stagTemps = [T1_vec, stagTemps]; % [K] all T0 observations
```

Now plot those stagnation temperatures as a function of RPM.

```
plot(krpm, stagTemps, '-o');
xlabel('Compressor Angular Speed [kRPM]');
ylabel('Temperature [K]');
legend(vars, 'Location', 'BestOutside');
plotFixer();
```



Published with MATLAB® R2017b