Table of Contents

Setting Up

ME140 - Advanced Thermal systems Reginald Mitchell Spring 2018 Project #3: SR30 Combustion Report
Alex Casas, Ian Gunady, Minh Ngo, Zane Zook

```
close all; clear all ; clc;
in2m = 0.0254;
C2K = 273.15;
lbf2N = 4.44822;
P_atm = 101325;             % [Pa]
R = 287;                    % [J/kg K]
Cp_const = 1.005e3;         % [J/kgK]

G1.A1 = 27.3 * in2m^2;
G1.A2 = 6.4 * in2m^2;
G1.A3 = 9.0 * in2m^2;
G1.A4 = 7.2 * in2m^2;
G1.A5 = 4.7 * in2m^2;
G1.A8 = 3.87 * in2m^2;

G2.RF_T2 = 0.68;           % Cross Flow RF
G2.RF_T3 = 0.68;           % Cross Flow RF
G2.RF_T4 = 0.68;           % Cross Flow RF
G2.RF_T5 = 0.86;           % Axial Flow RF
G2.RF_T8 = 0.68;           % Cross Flow RF

G3.LHV = 42.8e6;        % J/kg

% Given properties of Jet A and Dodecane
JetA.HC_ratio = 1.8;
JetA.mol_mass = 170;
JetA.h_vap = 270;
JetA.LHV = 42800;            % [kJ/kg] vapor, stp

Dod.HC_ratio = 2.17;
Dod.mol_mass = 170;
```

```
Dod.h_vap = 331;
Dod.LHV = 44560;            % [kJ/kg] vapor, stp

% Importing Data & Reassigning the columns to corresponding names

filename = 'Data.xlsx';
data = xlsread(filename);

D.RPM = data(:,1)';
D.T2m = data(:,2)' + C2K;                               % [K]
D.T3m = data(:,3)' + C2K;                               % [K]
D.T4m = data(:,4)' + C2K;                               % [K]
D.T5m = data(:,5)' + C2K;                               % [K]
D.T8m = data(:,6)' + C2K;                               % [K]
D.DP2 = (data(:,7)')*1e3; % difference b/w static and stagnation, [Pa]
D.PT3 = (data(:,8)')*1e3 + P_atm;                       % [Pa]
D.P4 = (data(:,9)')*1e3 + P_atm;                        % [Pa]
D.PT5 = (data(:,10)')*1e3 + P_atm;                      % [Pa]
D.PT8 = (data(:,11)')*1e3+ P_atm;                       % [Pa]
D.FuelFlow = data(:,12)';                               % [kg/s]
D.Thrust = data(:,13)'*lbf2N;                           % [N]
```

Project 2, Problem 2

```
% State 2
P2.Po2 = P_atm*ones(size(D.DP2));                   % [Pa]
P2.P2 = P2.Po2 - D.DP2;                             % [Pa]
[~, P2.k2] = specHeatAir(D.T2m);
P2.M2 = ((2./(P2.k2-1)).*((P2.Po2./P2.P2).^((P2.k2-1)./
P2.k2)-1)).^(1/2);

P2.T2 = Tm2T(D.T2m, G2.RF_T2, P2.M2, P2.k2);        % [K]
P2.To2 = T2To(P2.T2, P2.M2, P2.k2);                 % [K]

P2.V2 = P2.M2.*sqrt(P2.k2*R.*P2.T2);
P2.ro2 = P2.Po2./(R.*P2.To2);
P2.r2 = P2.ro2.*(P2.P2./P2.Po2).*(P2.To2./P2.T2);
P2.mdot_air = P2.r2.*P2.V2.*G1.A2;                  % [kg/s]
P2.MFP2 = (P2.mdot_air.*sqrt(R*P2.To2)./(G1.A2.*P2.Po2));

% State 1
P2.Po1 = P2.Po2;                                    % [Pa]
P2.To1 = P2.To2;                                    % [K]
P2.MFP1 = P2.MFP2*G1.A2/G1.A1;
P2.k1 = P2.k2;
for i = 1:length(P2.k1)
    P2.M1(i) = MFP2M(P2.k1(i),P2.MFP1(i));
end
P2.Tratio1 = isenRatioT(P2.k1, P2.M1);
P2.T1 = P2.To1./P2.Tratio1;
P2.V1 = P2.M1.*sqrt(P2.k1.*R.*P2.T1);

% State 3
```

```
P2.Po3 = D.PT3;
[~, P2.k3] = specHeatAir(D.T3m);
P2.To3 = D.T3m; % estimated value; first guess
for j = 1:2
    P2.MFP3 = (P2.mdot_air.*sqrt(R*P2.To3)./(G1.A3.*P2.Po3));   %
 estimated MFP
    for i = 1:length(P2.MFP3)
        P2.M3(i) = MFP2M(P2.k3(i),P2.MFP3(i));   % estimated value
    end
    P2.T3 = D.T3m./(1+G2.RF_T3.*(P2.k3-1)./2.*P2.M3.^2);
    P2.To3 = P2.T3.*(1+(P2.k3-1)./2.*P2.M3.^2);
end
P2.V3 = P2.M3.*sqrt(P2.k3.*R.*P2.T3);
```

Project 3, Problem 1

Chemistry stuff

```
C.x = 12.3;         C.y = 22.2;

C.b = C.x;
C.c = C.y/2;
C.a = C.b+0.5*C.c;
C.d = 3.76*C.a;

% M = Molar Mass
C.Mb = 44.01e-3;        C.Mc = 18.01e-3;        C.Md = 14.01e-3; %[kg/
mol]
C.MProd = C.b*C.Mb + C.c*C.Mc + C.d*C.Md;

C.M_O2 = 32;        C.M_N2 = 28.013;    C.M_C = 12.01;      C.M_H =
 1.01;
C.M_JetA = C.x*C.M_C  + C.y*C.M_H;

C.AFs = (C.a*C.M_O2 + C.d*C.M_N2)/(C.x*C.M_C + C.y*C.M_H);
C.AF = P2.mdot_air ./ D.FuelFlow;

C.Phi = C.AFs ./ C.AF;

% N = Number of Moles
C.N_fuel = C.Phi;
C.N_CO2 = C.b * C.Phi;
C.N_H2O = C.c * C.Phi;
C.N_N2 = C.d * ones(size(C.Phi));
C.N_O2 = C.a * (1-C.Phi);

C.N_prod = C.N_CO2 + C.N_H2O + C.N_N2 + C.N_O2;

% y = Mole Fraction
C.y_CO2 = C.N_CO2 ./ C.N_prod;
C.y_H2O = C.N_H2O ./ C.N_prod;
C.y_N2 = C.N_N2 ./ C.N_prod;
C.y_O2 = C.N_O2 ./ C.N_prod;
```

```
C.molFrac = [C.y_CO2; C.y_H2O; C.y_N2; C.y_O2];
```

State 4

```
P2.P4 = D.P4;
P2.QDotComb = G3.LHV * D.FuelFlow; %[J/s]
P2.mDotTot = D.FuelFlow + P2.mdot_air;

% For original states at 4, assuming all air
P2.To4_air = D.T4m;         % INITIAL GUESS
[~, P2.k4_air] = specHeatAir(D.T4m);
for j = 1:2
    for i = 1:length(P2.To4_air)
        [P2.MFP4_air(i),P2.Po4_air(i),P2.M4_air(i)] =
 ToP2MFP(P2.mDotTot(i),...
              R, P2.To4_air(i), G1.A4, P2.k4_air(i), P2.P4(i));
    end
    P2.T4_air = D.T4m./(1+G2.RF_T4.*(P2.k4_air-1)./2.*P2.M4_air.^2);
    P2.To4_air = P2.T4_air.*isenRatioT(P2.k4_air,P2.M4_air);
    P2.V4_air = P2.M4_air.*sqrt(P2.k4_air.*R.*P2.T4_air);
end

% For new states at 4, accounting for air and fuel mixture
[~, P2.k4_mix] = specHeatMix(D.T4m, C.molFrac);
P2.To4_mix = D.T4m;         % INITIAL GUESS
for j = 1:2
    for i = 1:length(P2.To4_mix)
    [P2.MFP4_mix(i),P2.Po4_mix(i), P2.M4_mix(i)] =
 ToP2MFP(P2.mDotTot(i),...
             R, P2.To4_mix(i), G1.A4,P2.k4_mix(i), P2.P4(i));
    end
    P2.T4_mix = D.T4m./(1+G2.RF_T4.*(P2.k4_mix-1)./2.*P2.M4_mix.^2);
    P2.To4_mix = P2.T4_mix.*isenRatioT(P2.k4_mix,P2.M4_mix);
    P2.V4_mix = P2.M4_mix.*sqrt(P2.k4_mix.*R.*P2.T4_mix);
end

% State 5
P2.Po5 = D.PT5;

% mixtures
P2.To5_mix = D.T5m;  % initilizing to an estimated value
[~, P2.k5_mix] = specHeatMix(D.T5m, C.molFrac);
for j = 1:2
    P2.MFP5_mix = (P2.mDotTot.*sqrt(R*P2.To5_mix)./(G1.A5.*P2.Po5));
    for i = 1:length(P2.k5_mix)
        P2.M5_mix(i) = MFP2M(P2.k5_mix(i),P2.MFP5_mix(i));
    end
    P2.T5_mix = D.T5m./(1+G2.RF_T5.*(P2.k5_mix-1)./2.*P2.M5_mix.^2);
    P2.To5_mix = P2.T5_mix.*isenRatioT(P2.k5_mix,P2.M5_mix);
end
P2.V5_mix = P2.M5_mix.*sqrt(P2.k5_mix.*R.*P2.T5_mix);

% assuming just air
```

```
P2.To5_air = D.T5m;  % initilizing to an estimated value
[~, P2.k5_air] = specHeatAir(D.T5m);
for j = 1:2
    P2.MFP5_air = (P2.mDotTot.*sqrt(R*P2.To5_air)./(G1.A5.*P2.Po5));
    for i = 1:length(P2.k5_air)
        P2.M5_air(i) = MFP2M(P2.k5_air(i),P2.MFP5_air(i));
    end
    P2.T5_air = D.T5m./(1+G2.RF_T5.*(P2.k5_air-1)./2.*P2.M5_air.^2);
    P2.To5_air = P2.T5_air.*isenRatioT(P2.k5_air,P2.M5_air);
end
P2.V5_air = P2.M5_air.*sqrt(P2.k5_air.*R.*P2.T5_air);

% State 8
P2.P8 = P2.Po2;         % P8 is the same as Patm because subsonic
P2.Po8 = D.PT8;
[~, P2.k8_air] = specHeatAir(D.T8m);
P2.M8_air = sqrt(2./(P2.k8_air-1).*((P2.Po8./P2.P8).^((P2.k8_air-1)./
P2.k8_air)-1));
P2.T8_air = D.T8m./(1+G2.RF_T8.*(P2.k8_air-1)./2.*P2.M8_air);
P2.To8_air = P2.T8_air.*isenRatioT(P2.k8_air,P2.M8_air);

P2.V8_air = (2*dh(P2.To8_air, P2.T8_air)).^(1/2);
P2.F_T_air = P2.mDotTot .* P2.V8_air;       % [N] Total thrust

[~, P2.k8_mix] = specHeatMix(D.T8m, C.molFrac);
P2.M8_mix = sqrt(2./(P2.k8_mix-1).*((P2.Po8./P2.P8).^((P2.k8_mix-1)./
P2.k8_mix)-1));
P2.T8_mix = D.T8m./(1+G2.RF_T8.*(P2.k8_mix-1)./2.*P2.M8_mix);
P2.To8_mix = P2.T8_mix.*isenRatioT(P2.k8_mix,P2.M8_mix);

P2.V8_mix = (2*dhMix(C.molFrac, P2.To8_mix, P2.T8_mix)).^(1/2);
P2.F_T_mix = P2.mDotTot .* P2.V8_mix;       % [N] Total thrust
```

Plotting Portion of Problem 2

```
Plot2.Po_arr = [P2.Po1; P2.Po2; P2.Po3; P2.Po4_mix; P2.Po5; P2.Po8];
Plot2.To_arr = [P2.To1; P2.To2; P2.To3; P2.To4_mix; P2.To5_mix;
 P2.To8_mix];
Plot2.M_arr = [P2.M1; P2.M2; P2.M3; P2.M4_mix; P2.M5_mix; P2.M8_mix];
Plot2.V_arr = [P2.V1; P2.V2; P2.V3; P2.V4_mix; P2.V5_mix; P2.V8_mix];
Plot2.leg = ['Station 1     '; 'Station 2     '; 'Station 3     ';...
             'Station 4, mix'; 'Station 5, mix'; 'Station 8, mix'];
Plot2.line = ['+-'; 'o-'; 's-'; 'd-'; '^-'; '*-'];
% Stagnation Temp Plot
figure('units','normalized','outerposition',[0 0 .75 .75]); % for
 larger plot
hold on;
for i = 1:6
    Plot2.h(i) = plot(D.RPM/(1e3),
 Plot2.To_arr(i,:),Plot2.line(i,:),...
                        'LineWidth',2,'MarkerSize',10);
end
legend(Plot2.h,Plot2.leg, 'Location', 'best');
```

```
xlabel('Spool Speed, \omega
 [kRPM]','FontSize',18,'FontWeight','bold');
ylabel('Stagnation Temperature, T_o
 [K]','FontSize',18,'FontWeight','bold');
set(gca,'FontSize',18);

% Stagnation Pressure Plot
figure('units','normalized','outerposition',[0 0 .75 .75]); % for
 larger plot
hold on;
for i = 1:6
    Plot2.h(i) = plot(D.RPM/(1e3), Plot2.Po_arr(i,:)/(1e3),
 Plot2.line(i,:),...
                     'LineWidth',2,'MarkerSize',10);
end
legend(Plot2.h,Plot2.leg, 'Location', 'best');
xlabel('Spool Speed, \omega
 [kRPM]','FontSize',18,'FontWeight','bold');
ylabel('Stagnation Pressure, P_o
 [kPa]','FontSize',18,'FontWeight','bold');
set(gca,'FontSize',18);

% Mach Number Plot
figure('units','normalized','outerposition',[0 0 .75 .75]); % for
 larger plot
hold on;
for i = 1:6
    Plot2.h(i) = plot(D.RPM/(1e3), Plot2.M_arr(i,:),
 Plot2.line(i,:),...
                     'LineWidth',2,'MarkerSize',10);
end
legend(Plot2.h,Plot2.leg, 'Location', 'best');
xlabel('Spool Speed, \omega
 [kRPM]','FontSize',18,'FontWeight','bold');
ylabel('Mach Number, M ','FontSize',18,'FontWeight','bold');
set(gca,'FontSize',18);

% Velocity Plot
figure('units','normalized','outerposition',[0 0 .75 .75]); % for
 larger plot
hold on;
for i = 1:6
    Plot2.h(i) = plot(D.RPM/(1e3), Plot2.V_arr(i,:),
 Plot2.line(i,:),...
                     'LineWidth',2,'MarkerSize',10);
end
legend(Plot2.h,Plot2.leg, 'Location', 'best');
xlabel('Spool Speed, \omega
 [kRPM]','FontSize',18,'FontWeight','bold');
ylabel('Velocity, V [m/s]','FontSize',18,'FontWeight','bold');
set(gca,'FontSize',18);

% Air Fuel Ratio:
P2.AF = P2.mdot_air./D.FuelFlow;
```

```
% air flow rate, fuel flow rate,
figure('units','normalized','outerposition',[0 0 .75 .75]); % for
 larger plot
plot(D.RPM/
(1e3),P2.mdot_air*1e2, 'o--','LineWidth',2,'MarkerSize',10);
hold on;
plot(D.RPM/(1e3),D.FuelFlow*1e4, '^-','LineWidth',2,'MarkerSize',10);
plot(D.RPM/(1e3),P2.AF,'s-','LineWidth',2,'MarkerSize',10);
legend('100 x Air Flow Rate', '10000 x Fuel Flow Rate', 'Air-Fuel
 Ratio', 'Location', 'best')
xlabel('Spool Speed, \omega
 [kRPM]','FontSize',18,'FontWeight','bold');
ylabel('Flow Rates [kg/s] and Air-Fuel
 Ratio','FontSize',18,'FontWeight','bold');
set(gca,'FontSize',18);

% calculated thrust and recorded thrust
figure('units','normalized','outerposition',[0 0 .75 .75]); % for
 larger plot
plot(D.RPM/(1e3),D.Thrust,'o-','LineWidth',2,'MarkerSize',10);
hold on;
plot(D.RPM/(1e3),P2.F_T_mix, 's--','LineWidth',2,'MarkerSize',10);
legend('Measured Thrust','Calculated Thrust','Location', 'best')
xlabel('Spool Speed, \omega
 [kRPM]','FontSize',18,'FontWeight','bold');
ylabel('Thrust Force, F_T [N]','FontSize',18,'FontWeight','bold');
set(gca,'FontSize',18);

% specific thrust
P3.ST = P2.F_T_mix ./ P2.mdot_air;

% thrust specific fuel consumption
P3.TSFC = D.FuelFlow ./ P2.F_T_mix;

figure('units','normalized','outerposition',[0 0 .75 .75]); % for
 larger plot
plot(D.RPM/(1e3),P3.ST,'s-','LineWidth',2,'MarkerSize',10);
hold on;
plot(D.RPM/(1e3),P3.TSFC*(1e6),'o-','LineWidth',2,'MarkerSize',10);

ylabel('ST [N/kg/s] and TSFC [\mu N/kg/
s]','FontSize',18,'FontWeight','bold');
xlabel('Spool Speed, \omega
 [kRPM]','FontSize',18,'FontWeight','bold');
set(gca,'FontSize',18);
legend('ST [N/kg/s]', 'TSFC [\muN/kg/s]', 'Location', 'best')

% thermal efficiency  = net work / heat input; heat input =
 P2.QDotComb
% W_net kinetic energy
P3.W_net_mix = 0.5*P2.mDotTot.*P2.V8_mix.^2;
P3.Eff_therm_mix = P3.W_net_mix./P2.QDotComb;
```

```
figure('units','normalized','outerposition',[0 0 .75 .75]); % for
 larger plot
plot(D.RPM/(1e3),P3.Eff_therm_mix,'o-','LineWidth',2,'MarkerSize',10);
xlabel('Spool Speed, \omega
 [kRPM]','FontSize',18,'FontWeight','bold');
ylabel('Thermal Efficiency,
 \eta_{therm}','FontSize',18,'FontWeight','bold');
set(gca,'FontSize',18);
```

Problem 4

```
P4.W_comp_1 = P2.mdot_air .* dh(P2.To3, P2.To2);
P4.W_turb_1 = -1*P2.mDotTot .* dh(P2.To5_air, P2.To4_air);

P4.W_comp_2 = P2.mdot_air .* dhMix(C.molFrac, P2.To3, P2.To2);
P4.W_turb_2 = -1*P2.mDotTot .* dhMix(C.molFrac, P2.To5_mix,
 P2.To4_mix);

P4.W_comp_3 = P2.mdot_air .* Cp_const .* (P2.To3 - P2.To2);
P4.W_turb_3 = -1*P2.mDotTot .* Cp_const .* (P2.To5_air -P2.To4_air);

figure('units','normalized','outerposition',[0 0 .75 .75]); % for
 larger plot
hold on;
plot(D.RPM/(1e3),P4.W_turb_1/
(1e3),'s-','LineWidth',2,'MarkerSize',7.5);
plot(D.RPM/(1e3),P4.W_comp_1/
(1e3),'o:','LineWidth',2,'MarkerSize',7.5);

xlabel('Spool Speed, \omega
 [kRPM]','FontSize',18,'FontWeight','bold');
ylabel('Compressor and Turbine Power,
 [kW]','FontSize',18,'FontWeight','bold');
set(gca,'FontSize',18);
legend('Turbine Power','Compressor Power', 'Location', 'best')

% compressor adiabatic efficiencies
for i = 1:length(P4.W_comp_1)
    P4.To3s(i) = presRatio2Temp(P2.Po2(i), P2.Po3(i), P2.To2(i));
end
P4.W_comp_s = P2.mdot_air .* dh(P4.To3s, P2.To2);
P4.eta_comp = P4.W_comp_s./P4.W_comp_1;

% turbine adiabatic efficiencies
for i = 1:length(P4.W_comp_1)
    P4.To5s(i) = presRatio2TempInit(P2.Po5(i), P2.Po4_mix(i),
 P2.To4_mix(i));
end
P4.W_turb_s = -1*P2.mDotTot .* dh(P4.To5s, P2.To4_mix);
P4.eta_turb = P4.W_turb_1./P4.W_turb_s;

% nozzle adiabatic efficiencies
P4.T8s = P2.To5_mix.*(P2.P8/P2.Po5).^((P2.k8_mix-1)./P2.k8_mix);
```

```
P4.M8s = sqrt((2./(P2.k8_mix - 1)).*((P2.Po5./P2.P8).^((P2.k8_mix -
 1)./P2.k8_mix)-1));
P4.V8s = P4.M8s.*sqrt(P2.k8_mix.*R.*P4.T8s);
% the way reggie suggested
P4.V8s_alt = (-2*dh(P2.To5_mix, P2.To8_mix)).^(1/2);

% nozzle efficiency is:
P4.eta_noz = (P2.V8_mix.^2)./(P4.V8s.^2);

% combustor stagnation pressure loss
P4.CombPressLoss = P2.Po4_mix ./ P2.Po3;

% apparent combustion efficiency vs spool speed
P4.Q_comb = P2.mDotTot.* dh(P2.To4_mix, P2.To3);
P4.eta_comb = (P2.mDotTot.* dh(P2.To4_mix, P2.To3))./P2.QDotComb;

figure('units','normalized','outerposition',[0 0 .75 .75]); % for
 larger plot
plot(D.RPM/(1e3),P4.CombPressLoss,'o-','LineWidth',2,'MarkerSize',10);
hold on;
plot(D.RPM/(1e3),P4.eta_comp,'^-','LineWidth',2,'MarkerSize',10);
plot(D.RPM/(1e3),P4.eta_turb,'s-','LineWidth',2,'MarkerSize',10);
plot(D.RPM/(1e3),P4.eta_noz,'h-','LineWidth',2,'MarkerSize',10);
%plot(D.RPM/(1e3),P4.eta_comb,'o--','LineWidth',2,'MarkerSize',10);
xlabel('Spool Speed, \omega
 [kRPM]','FontSize',18,'FontWeight','bold');
ylabel('Efficiency, Pressure
 Ratio','FontSize',18,'FontWeight','bold');
set(gca,'FontSize',18);
% legend('Apparent Combustion Efficiency', 'Compressor Efficiency',...
%      'Turbine Efficiency', 'Nozzle Efficiency', 'Combustor
 Efficiency',...
%      'Location', 'best');
legend('\pi_b, Combustor Pressure Loss','\eta_c Compressor
 Efficiency',...
    '\eta_t Turbine Efficiency', '\eta_n Nozzle Efficiency',...
    'Location', 'best');
```

Deliverable 3

```
H_bar.CO2 = -393520;     %J/mol
H_bar.H20 = -241820;     %J/mol
H_bar.O2 = 0;            %J/mol
H_bar.MJetA = 0.17;      %kg/mol

H_bar.JetA = C.b * H_bar.CO2 + C.c * H_bar.H20 - C.a * H_bar.O2 + ...
            G3.LHV * H_bar.MJetA ; % J/mol
H.JetA = H_bar.JetA./H_bar.MJetA;  % J/kg
```

Deliverable 4

```
P3.Phi = linspace(0.05,0.65,13);
```

```
P3.H_bar_Dod = -1.712e6*0.17; %J/mol

P3.T_react = 300; %K
for i = 1:length(P3.Phi)
    P3.T_a_JetA(i) = findTA(H_bar.JetA, H_bar.CO2, H_bar.H20,
 P3.Phi(i), P3.T_react);
    P3.T_a_DoD(i) = findTA(P3.H_bar_Dod, H_bar.CO2, H_bar.H20,
 P3.Phi(i), P3.T_react);
    P3.T_react = P3.T_a_JetA(i);
end

figure('units','normalized','outerposition',[0 0 .75 .75]); % for
 larger plot
hold on;
plot(P3.Phi, P3.T_a_DoD,'s-','LineWidth',2,'MarkerSize',10)
plot(P3.Phi, P3.T_a_JetA, 'o:','LineWidth',2,'MarkerSize',10);
legend('JetA', 'Dodecane', 'Location', 'best');
xlabel('Equivalence Ratio, \Phi','FontSize',18,'FontWeight','bold');
ylabel('Adiabatic Flame Temp, T_a','FontSize',18,'FontWeight','bold');
set(gca,'FontSize',18);
```

Deliverable 5 (and 2)

```
P3.T_guess = 300; %[K]
for i = 1:length(C.Phi)
    P3.T_A(i) = findTA_P3(H_bar.JetA, H_bar.CO2, H_bar.H20,
 C.Phi(i),...
                            P3.T_guess, P2.To3(i));
    P3.T_guess = P3.T_A(i);
end
for i = 1:length(P3.T_A)
    P3.To5s(i) = presRatio2TempMix(P2.Po4_mix(i), P2.Po5(i),
 P3.T_A(i), ...
        C.molFrac(:,1));
end
P4.W_turb_flame = -1 * P2.mDotTot .* dhMix(C.molFrac, P3.To5s,
 P3.T_A);
% added power curve to the turbine/compressor power plot
figure('units','normalized','outerposition',[0 0 .75 .75]); % for
 larger plot
hold on;
plot(D.RPM/(1e3),P4.W_turb_1/
(1e3),'^-','LineWidth',2,'MarkerSize',10);
plot(D.RPM/(1e3),P4.W_turb_3/
(1e3),'s-','LineWidth',2,'MarkerSize',10);
plot(D.RPM/(1e3),P4.W_turb_2/
(1e3),'d-','LineWidth',2,'MarkerSize',10);

xlabel('Spool Speed, \omega
 [kRPM]','FontSize',18,'FontWeight','bold');
ylabel('Turbine Power, [kW]','FontSize',18,'FontWeight','bold');
legend('Turbine Power w/ air, variable c_p',...
```

```
    'Turbine Power w/ air, constant c_p', 'Turbine Power w/ JetA
 combustion',...
    'Location', 'best');
set(gca,'FontSize',18);

figure('units','normalized','outerposition',[0 0 .75 .75]); % for
 larger plot
plot(D.RPM/(1e3), P4.W_turb_flame/
(1e3),'o-','LineWidth',2,'MarkerSize',10);
hold on
plot(D.RPM/(1e3),P4.W_turb_2/
(1e3),'d-','LineWidth',2,'MarkerSize',10);
plot(D.RPM/(1e3),P4.W_comp_1/
(1e3),'d-','LineWidth',2,'MarkerSize',10);
xlabel('Spool Speed, \omega
 [kRPM]','FontSize',18,'FontWeight','bold');
ylabel('Turbine and Compressor Power,
 [kW]','FontSize',18,'FontWeight','bold');
legend('Ideal Turbine Power', 'Turbine Power w/ JetA combustion',...
    'Compressor Power w/ air, variable c_p','Location', 'best');
set(gca,'FontSize',18);

Plot5.To_arr = [P2.To1; P2.To2; P2.To3; P2.To4_mix; P2.To5_mix;
 P2.To8_mix;...
                P3.T_A; P3.To5s];
Plot5.leg = ['Station 1      '; 'Station 2      '; 'Station 3
 ';...
            'Station 4, mix '; 'Station 5, mix '; 'Station 8, mix
 '; ...
            'Station 4, adia'; 'Station 5, isen'];
Plot5.line = ['+-'; 'o:'; 's-'; 'd-'; '^-'; '*-';'h-'; 'p-'];

% Stagnation Temp Plot
figure('units','normalized','outerposition',[0 0 .75 .75]); % for
 larger plot
hold on;
for i = 1:8
    Plot5.h(i) = plot(D.RPM/(1e3),
 Plot5.To_arr(i,:),Plot5.line(i,:),...
                    'LineWidth',2,'MarkerSize',10);
end

legend(Plot5.h, Plot5.leg, 'Location', 'best');
xlabel('Spool Speed, \omega
 [kRPM]','FontSize',18,'FontWeight','bold');
ylabel('Stagnation Temperature, T_o
 [K]','FontSize',18,'FontWeight','bold');
set(gca,'FontSize',18);
```

Improvements to SR 30

```
I1.A8 = [3.87, 3.75, 3.5, 3.25].*in2m^2;
for i = 1:length(I1.A8)
```

```
    I1.MFP8(i,:) = (P2.mdot_air./I1.A8(i)).*sqrt(R*P2.To5_mix)./
P2.Po5;
    for j = 1:length(I1.MFP8)
        I1.M8(i,j) = MFP2M(P2.k8_mix(j), I1.MFP8(i,j));
    end
    I1.T8s(i,:) = P2.To5_mix./isenRatioT(P2.k8_mix,I1.M8(i,:));
    I1.V8s(i,:) = (2*dhMix(C.molFrac, P2.To5_mix,
 I1.T8s(i,:))).^(1/2);
    I1.F_Ts(i,:) = P2.mDotTot .* I1.V8s(i,:);       % [N] Total thrust
end

% calculated thrust and recorded thrust
figure('units','normalized','outerposition',[0 0 .75 .75]); % for
 larger plot
plot(D.RPM/(1e3),I1.F_Ts(1,:),'o-','LineWidth',2,'MarkerSize',10);
hold on;
plot(D.RPM/(1e3),I1.F_Ts(2,:), 's-','LineWidth',2,'MarkerSize',10);
plot(D.RPM/(1e3),I1.F_Ts(3,:), '*-','LineWidth',2,'MarkerSize',10);
plot(D.RPM/(1e3),I1.F_Ts(4,:), '^-','LineWidth',2,'MarkerSize',10);
legend('Area = 3.87 m^2','Area = 3.75 m^2', 'Area = 3.5 m^2', 'Area =
 3.25 m^2','Location', 'best')
xlabel('Spool Speed, \omega
 [kRPM]','FontSize',18,'FontWeight','bold');
ylabel('Thrust Force, F_T [N]','FontSize',18,'FontWeight','bold');
set(gca,'FontSize',18);
```

Improvement 2 Increase Fuel Richness

```
multiplier = 1:0.25:3;
for j = 1:length(multiplier)
    I2.Fuelflow = multiplier(j)*D.FuelFlow;
    I2.Phi  = multiplier(j)*C.Phi;
    I2.mDotTot = P2.mdot_air + I2.Fuelflow;

    % N = Number of Moles
    I2.N_fuel = I2.Phi;
    I2.N_CO2 = C.b * I2.Phi;
    I2.N_H2O = C.c * I2.Phi;
    I2.N_N2 = C.d * ones(size(I2.Phi));
    I2.N_O2 = C.a * (1-I2.Phi);

    I2.N_prod = I2.N_CO2 + I2.N_H2O + I2.N_N2 + I2.N_O2;

    % y = Mole Fraction
    I2.y_CO2 = I2.N_CO2 ./ I2.N_prod;
    I2.y_H2O = I2.N_H2O ./ I2.N_prod;
    I2.y_N2 = I2.N_N2 ./ I2.N_prod;
    I2.y_O2 = I2.N_O2 ./ I2.N_prod;
    I2.molFrac = [I2.y_CO2; I2.y_H2O; I2.y_N2; I2.y_O2];
    I2.T_guess = 300; %[K]
    for i = 1:length(I2.Phi)
        I2.T_A(j,i) = findTA_P3(H_bar.JetA, H_bar.CO2, H_bar.H20,
 I2.Phi(i),...
```

```
                                I2.T_guess, P2.To3(i));
        I2.T_guess = I2.T_A(j,i);
    end
    for i = 1:length(I2.Phi)
        I2.To5s(j,i) = presRatio2TempMix(P2.Po4_mix(i), P2.Po5(i),
 I2.T_A(j,i), ...
            I2.molFrac(:,i));
    end
    I2.W_turb_flame(j,:) = -1 * I2.mDotTot .* dhMix(I2.molFrac,
 I2.To5s(j,:), I2.T_A(j,:));

end
figure('units','normalized','outerposition',[0 0 .75 .75]); % for
 larger plot
hold on;
xlabel('Spool Speed, \omega
 [kRPM]','FontSize',18,'FontWeight','bold');
ylabel('Turbine Power, [kW]','FontSize',18,'FontWeight','bold');
set(gca,'FontSize',18);
for i = 1:length(multiplier)
    name = num2str(multiplier(i)) + " * Original \Phi";
    plot(D.RPM/(1e3),I2.W_turb_flame(i,:)/
(1e3),'d-','LineWidth',2,'MarkerSize',10,'DisplayName',name);
end
legend('Location', 'northwest');
```

Improvement 3

```
multiplier = 0.9:0.05 :1.1;

I3.QDotComb = G3.LHV * D.FuelFlow; %[J/s]

for z = 1:length(multiplier)
    % State 3
    I3.Po3 = D.PT3 * multiplier(z);
    [~, I3.k3] = specHeatAir(D.T3m);
    I3.To3 = D.T3m; % estimated value; first guess
    for j = 1:2
        I3.MFP3 = (P2.mdot_air.*sqrt(R*I3.To3)./(G1.A3.*I3.Po3));   %
 estimated MFP
        for i = 1:length(P2.MFP3)
            I3.M3(i) = MFP2M(I3.k3(i),I3.MFP3(i));   % estimated value
        end
        I3.T3 = D.T3m./(1+G2.RF_T3.*(I3.k3-1)./2.*I3.M3.^2);
        I3.To3 = I3.T3.*(1+(I3.k3-1)./2.*I3.M3.^2);
    end

    I3.W_comp = P2.mdot_air .* dh(I3.To3, P2.To2);
    for i = 1:length(I3.W_comp)
        I3.To3s(i) = presRatio2Temp(P2.Po2(i), I3.Po3(i), P2.To2(i));
    end
    I3.W_comp_s = P2.mdot_air .* dh(I3.To3s, P2.To2);
    I3.eta_comp(:,z) = I3.W_comp_s ./ I3.W_comp;
```

```
end

figure('units','normalized','outerposition',[0 0 .75 .75]); % for
 larger plot
hold on;
xlabel('Spool Speed, \omega
 [kRPM]','FontSize',18,'FontWeight','bold');
ylabel('Compressor Efficiency,
 \eta','FontSize',18,'FontWeight','bold');
set(gca,'FontSize',18);
for i = 1:length(multiplier)
    name = num2str(multiplier(i)) + " * Original Pressure Ratio";
    plot(D.RPM/
(1e3),I3.eta_comp(:,i)','d-','LineWidth',2,'MarkerSize',10,'DisplayName',name);
end
legend('Location', 'best');
```

Published with MATLAB® R2017b