Tutorial: Engine Performance Modeling

Christopher Chinske

May 20, 2019

Tutorial: Engine Performance Modeling

Copyright 2019 Christopher Chinske. This work is licensed under the Creative Commons Attribution-ShareAlike 4.0 International License. To view a copy of this license, visit http://creativecommons.org/licenses/by-sa/4.0/ or send a letter to Creative Commons, PO Box 1866, Mountain View, CA 94042, USA.

1 Introduction

One of the more powerful applications of the Gas Turbine Propulsion Toolbox (GTPT) is the ability to build a feasible engine performance model (EPM) for an engine using limited engine parameter data. This tutorial will describe how to use GTPT to build an EPM for the CFM56-5B1. The CFM56-5B1 is used on the Airbus A321.

Before we begin, let's briefly discuss the problem at hand. If you want to model an aircraft's performance, you will need a reasonable engine performance model. The engine performance model, also known as the *engine deck*, returns thrust, T, and specific fuel consumption, s, as a function of flight condition (e.g., altitude, Mach number, and thrust lever angle).

$$T = f(h, M, TLA) \tag{1}$$

$$s = f(h, M, TLA) \tag{2}$$

However, aircraft engine suppliers generally do not publicly release the engine deck. Even when working within a leading research organization, it may be difficult or cost-prohibitive to obtain this data. Engine parameter data, such as component efficiencies, are even more difficult to come by. Only limited data about an engine may be publicly available.

A good source of engine parameter data is the *ICAO Aircraft Engine Emissions Databank* [1]. This data set lists bypass ratio, pressure ratio, rated output, and fuel flow for many engines. Since you have engine performance (i.e., rated output and fuel flow) for a single flight condition (rated output, sea level static thrust), you can use this limited engine performance data and GTPT to determine a feasible set of component efficiencies that match this performance point. This allows you to build a feasible engine deck.

Now that we have an idea of where we are going, we can concisely state the problem statement.

Problem Statement Use GTPT to build an engine performance model (EPM) for the CFM56-5B1 engine. Determine maximum thrust at 10,000 ft MSL and Mach 0.4.

2 Solution

2.1 Configuration

From the propulsion top level directory, add the following paths.

```
addpath optimize
addpath epm
```

2.2 Known Engine Parameters

From the *ICAO Aircraft Engine Emissions Databank* [1], the bypass ratio, compressor pressure ratio, rated output, and takeoff fuel flow are known.

$$\alpha = 5.7 \tag{3}$$
$$\pi_c = 30.2 \tag{4}$$
$$F = 133.45 \times 10^3 \text{ N} \tag{5}$$
$$f = 1.359 \text{ kg/s} \tag{6}$$

Therefore, the thrust specific fuel consumption (TSFC) is

$$s = \frac{1.359 \text{ kg/s}}{133.45 \times 10^3 \text{ N}} \frac{10^6 \text{ mg}}{1 \text{ kg}} = 10.184 \frac{\frac{\text{mg fuel}}{\text{s}}}{\text{N thrust}} \tag{7}$$

2.3 Build Optimization Script

Edit the optimization script `optimize/s_seek.m`. The optimization script includes four distinct parts.

1. The first part defines the lower and upper bounds for each free (i.e., unknown) parameter. For example:

   ```
   lb_pi_d = 0.98;
   ub_pi_d = 0.998;
   ```

   defines the lower bound and upper bound of the inlet pressure ratio, π_d. Table 6.3 in Oates [2] provides typical ranges of parameters. Definitions for these parameters may be found in the Interface Control Document (ICD) provided with the program.

2. Then, vectors of the lower bound and upper bound values are built.

```
lb = [lb_pi_d; ...
lb_pi_b; ...
lb_pi_n; ...
lb_pi_n2; ...
lb_eta_b; ...
lb_e_c; ...
lb_e_c2; ...
lb_e_t; ...
lb_tau_lam; ...
lb_pi_c2];

ub = [ub_pi_d; ...
ub_pi_b; ...
ub_pi_n; ...
ub_pi_n2; ...
ub_eta_b; ...
ub_e_c; ...
ub_e_c2; ...
ub_e_t; ...
ub_tau_lam; ...
ub_pi_c2];
```

3. Next, an initial guess vector is built. A good initial guess puts the guess values between the lower bound and upper bound values.

```
x0 = (lb+ub)./2;
```

4. Finally, the function `sqp` is called. **For this tutorial, change the function handle to `@phi_a321`, and save the file.**

```
x = sqp(x0,@phi_a321,[],[],lb,ub);
```

This script will return the vector `x`, which will contain the values of the free parameters in the same order as they were defined in the lower/upper

bound vectors. Therefore, in this tutorial, `x` will consist of the following parameters.

π_d
π_b
π_n
$\pi_{n'}$
η_b
e_c
$e_{c'}$
e_t
τ_λ
$\pi_{c'}$

2.4 Build Objective Function

Copy the file `optimize/phi.m` to `optimize/phi_a321.m`. Open the file `phi_a321.m` for editing.

The objective function defines the target TSFC and the fixed (i.e., known) parameters. The objective function then calls the appropriate cycle analysis function and computes the absolute difference between the computed TSFC and the target TSFC. In this tutorial, we are considering a subsonic transport turbofan engine. The cycle analysis function `nonideal_turbofan2` applies specifically to this case — a turbofan with convergent exit nozzles. Specifically, `nonideal_turbofan2` applies to subsonic aircraft, separate stream turbofan engines, no afterburning, and cases where the pressure ratio across nozzles is small.

Rename the function to match the file name.

```
function s_err = phi_a321(x)
```

Update the target TSFC to match (7).

```
s_goal = 10.184;
```

Update the compressor pressure ratio and bypass ratio to match (4) and (3), respectively.

```
inputs.pi_c = 30.2;
inputs.alpha = 5.7;
```

Then, save the file.

2.5 Run Optimization Script

In the command window, run the optimization script.

```
s_seek
```

The variable `x` contains the determined values for the free parameters.

$$
\begin{aligned}
\pi_d &= 0.98110 \\
\pi_b &= 0.93547 \\
\pi_n &= 0.99066 \\
\pi_{n'} &= 0.99073 \\
\eta_b &= 0.96728 \\
e_c &= 0.92144 \\
e_{c'} &= 0.87762 \\
e_t &= 0.86339 \\
\tau_\lambda &= 5.35964 \\
\pi_{c'} &= 1.34782
\end{aligned}
\tag{8}
$$

2.6 Run the Cycle Analysis Function

The cycle analysis function outputs additional information that will be required for the engine performance model. First, we will build an inputs structure array. Then, we will call the cycle analysis function.

To build an inputs structure array, call the `build_inputs` function.

```
inputs = build_inputs;
```

Enter the inputs values at the prompts.

```
Primary stream afterburning flag:  0
Secondary stream afterburning flag:  0
T0 (K): 303.15
Ratio of specific heats, compressor:  1.4
Ratio of specific heats, turbine:  1.35
Specific heat of air Cp (J/(kg*K)), compressor:  1004.9
Specific heat of air Cp (J/(kg*K)), turbine:  1004.9
Fuel heating value (J/kg):  4.4194E7
Pressure ratio, inlet:  0.98110
Pressure ratio, burner:  0.93547
Pressure ratio, PRI nozzle:  0.99066
Pressure ratio, SEC nozzle:  0.99073
Efficiency, burner:  0.96728
Efficiency, mechanical:  1
Polytropic efficiency, PRI compressor:  0.92144
Polytropic efficiency, SEC compressor:  0.87762
Polytropic efficiency, turbine:  0.86339
p9/p0, PRI: 1
p9/p0, SEC: 1
tau_lam:  5.35964
Compressor pressure ratio:  30.2
Fan pressure ratio:  1.34782
Flight Mach number:  0
Bypass ratio:  5.7
```

Call the cycle analysis function in verbose mode.[1]

```
[f_mdot, s, inputs] = nonideal_turbofan2(inputs,1);
```

Note the outputs of the cycle analysis function.

```
Etta ch:  0.88872
Etta c':  0.87234

M0*u9/u0:  1.481
M0*u9'/u0:  0.63971

Turbine temperature ratio:  0.54958
Turbine efficiency:  0.90069
Fan temperature ratio:  1.1021
Fuel-to-air ratio:  0.018396
```

2.7 Build Engine Performance Model

Copy the file `epm/example_epm.m` to `epm/epm_a321.m`. Open the file `epm_a321.m` for editing. Rename the function to match the file name.

```
function [s,f] = epm_a321(atmos,altitude,mach,throttle_fraction)
```

The engine performance model function takes as inputs: altitude (ft), Mach number, throttle fraction (1 implies maximum TT4), and the reference atmosphere `atmos`. GTPT includes a reference atmosphere lookup table. However, if you have a functional atmosphere (e.g., `atmosisa`), you could build the engine performance model to utilize that function instead of taking `atmos` as an input.

Line 7 includes a comment noting the engine. Update the comment for the present engine.

```
% Engine Performance Modeling, CFM 56-5B1
```

[1] You may want to turn pagination off before calling the cycle analysis function by executing the command `more off`.

Next, we assume that the engine parameters remain constant throughout its operating envelope. Reference, or on-design, quantities are denoted by the suffix `r`. Using the values in sections 2.2, 2.5, and 2.6, update the "build inputs" content of `epm_a321.m`.

```
% build inputs
inputs.gam_t = 1.35;
inputs.cp_c = 1004.9;
inputs.cp_t = 1004.9;
inputs.eta_ch = 0.88872;
inputs.eta_c2 = 0.87234;
inputs.eta_b = 0.96728;
inputs.pi_d = 0.98110;
inputs.pi_b = 0.93547;
inputs.pi_n = 0.99066;
inputs.pi_n2 = 0.99073;
inputs.h = 4.4194E7;
inputs.m0r = 0;
inputs.tau_lamr = 5.35964;
inputs.pi_c2r = 1.34782;
inputs.pi_cr = 30.2;
inputs.alphar = 5.7;
inputs.eta_chr = 0.88872;
inputs.eta_c2r = 0.87234;
inputs.eta_br = 0.96728;
inputs.pi_dr = 0.98110;
inputs.pi_br = 0.93547;
inputs.pi_nr = 0.99066;
inputs.pi_n2r = 0.99073;
inputs.t0r = 303.15;
inputs.m0u9u0r = 1.481;
inputs.m0u92u0r = 0.63971;
inputs.tau_tr = 0.54958;
inputs.eta_tr = 0.90069;
inputs.tau_c2r = 1.1021;
inputs.fr = 0.018396;
```

Update the "performance conditions" content of `epm_a321.m`. `F_REF` in lbf (from (5)) is 30,000. `TT4_MAX` is given by

$$T_{t4} = T_0 \; \tau_\lambda \tag{9}$$

$$T_{t4\ \mathtt{MAX}} = (303.15)(5.35964) = 1624.8 \text{ K} \tag{10}$$

The updated code is given below.

```
% performance conditions
f_ref = 30000;
tt4_max = 1624.8;
```

Then, save the file.

2.8 Utilize the Engine Performance Model

To determine the maximum thrust at 10,000 ft MSL and Mach 0.4, utilize the engine performance model by executing the following commands.

```
load atmos.dat
[s, thrust] = epm_a321(atmos,10000,0.4,1)
```

You should find that the thrust is 19,243 lbf.

3 Additional Notes

In this example, we used a T_0 of 303.15 K, which is Std + 15 ℃. Per the engine's type certificate data sheet (TCDS) [3], takeoff thrust is nominally independent of ambient temperature (flat rated) up to Std + 10 ℃. If you run the EPM for sea level, static thrust at Std (i.e., 288.15 K), you will find that the EPM over-predicts the takeoff thrust, which is rated at 30 000 lbf. How would you modify the EPM to output 30 000 lbf up to Std + 15 ℃?

Similarly, according to [3], maximum continuous is nominally independent of ambient temperature (flat rated) up to ambient temperature of Std + 10 ℃, and maximum continuous sea level, static thrust is rated at 29 090 lbf. How would you modify the EPM to employ maximum continuous thrust?

References

[1] European Union Aviation Safety Agency. (2018). *ICAO Aircraft engine emissions databank* [Data File]. Available from https://www.easa.europa.eu/easa-and-you/environment/icao-aircraft-engine-emissions-databank

[2] Oates, G. C. (1997). *Aerothermodynamics of Gas Turbine and Rocket Propulsion* (3rd ed.). American Institute of Aeronautics and Astronautics.

[3] U.S. Department of Transportation. (2018). *Type Certificate Data Sheet E37NE* (TCDS Number E37NE, Revision 14).