**Table of Contents**

[Abstract 6](#_Toc80425096)

[1 Airplane Details 7](#_Toc80425097)

[1.1 Overall Dimensions 7](#_Toc80425098)

[1.2 Engine details 7](#_Toc80425099)

[1.3 Weights 7](#_Toc80425100)

[1.4 Wing geometry 7](#_Toc80425101)

[1.5 Fuselage geometry 9](#_Toc80425102)

[1.6 Nacelle geometry 9](#_Toc80425103)

[1.7 Horizontal tail geometry 9](#_Toc80425104)

[1.8 Vertical tail geometry 9](#_Toc80425105)

[1.9 Other details 10](#_Toc80425106)

[1.10 Flight Condit 10](#_Toc80425107)

[2 Steady Level Flight 11](#_Toc80425108)

[2.1 Equation for steady level 11](#_Toc80425109)

[2.2 Vertical Forces 12](#_Toc80425110)

[2.3 Longitudinal Forces 13](#_Toc80425111)

[2.4 Drag polar 15](#_Toc80425112)

[2.5 Best lift to drag ratio 16](#_Toc80425113)

[2.6 Thrust required 17](#_Toc80425114)

[2.7 Minimum drag speed 18](#_Toc80425115)

[2.8 Drag polar for B737 19](#_Toc80425116)

[2.9 Remarks 20](#_Toc80425117)

[2.10 Stalling speed 21](#_Toc80425118)

[2.11 Variations of and with altitude 23](#_Toc80425119)

[3 Steady climb 29](#_Toc80425120)

[3.1 Equations of steady climb 29](#_Toc80425121)

[3.2 Steady climb performance 29](#_Toc80425122)

[4 Turning performance. 35](#_Toc80425123)

[4.1 Equations of turning fligth 35](#_Toc80425124)

[4.2 Turning performance 35](#_Toc80425125)

[5 Range and Endurance 46](#_Toc80425126)

[5.1 Range for jet propelled Aircraft 46](#_Toc80425127)

[5.2 Endurance for jet propelled aircraft 48](#_Toc80425128)

[6 Take off 50](#_Toc80425129)

[6.1 Calculation of Ground Roll 50](#_Toc80425130)

[6.2 Calculation of Distance While Airborne to Clear an Obstacle 54](#_Toc80425131)

[6.3 Total take off distance 54](#_Toc80425132)

[7 Landing 55](#_Toc80425133)

[7.1 Calculation of Approach Distance 55](#_Toc80425134)

[7.2 Calculation of flare distance 55](#_Toc80425135)

[7.3 Calculation of Ground Roll 55](#_Toc80425136)

[7.4 Total landing distance 55](#_Toc80425137)

[8 Python code 56](#_Toc80425138)

[8.1 Steady level flight 56](#_Toc80425139)

[8.1.1 Stalling speed code 56](#_Toc80425140)

[8.1.2 Variations of and with altitude code 57](#_Toc80425141)

[8.2 Steady climb and Turning flight code 58](#_Toc80425142)

[8.3 Range and Endurance code 65](#_Toc80425143)

[8.4 Take-off distance code 67](#_Toc80425144)

[8.5 Landing distance code 68](#_Toc80425145)

[9 Conclusions and remarks 70](#_Toc80425146)

[10 References 71](#_Toc80425147)

**Table of figures**

[Figure 1.4:A.(Three-view drawing of the airplane) 8](#_Toc80423414)

[Figure 2.1:A. Equilibrium Forces 12](#_Toc80423415)

[Figure 2.4:A. Drag polar 16](#_Toc80423416)

[Figure 2.7:A Thrust required vs TAS 19](#_Toc80423417)

[Figure 3.2:B. Maximum rate of climb vs altitude Table 2‑A. Variation of stalling speed with altitude 21](file:///C:\Users\Suresh%20Ragireddy\OneDrive\Desktop\report9.docx#_Toc80423418)

[Figure 2.10:A. Variation of stalling with altitude 22](#_Toc80423419)

[Figure 2.10:B. Stalling speed Vs Altitude 22](#_Toc80423420)

[Figure 2.10:C. Variation of density with Altitude 23](#_Toc80423421)

[Figure 2.11:A. Available and required at H=0km 25](#_Toc80423422)

[Figure 2.11:B. Available and required at H=1km 25](#_Toc80423423)

[Figure 2.11:C. Available and required at H=2km 25](#_Toc80423424)

[Figure 2.11:D. Available and required at H=4km 26](#_Toc80423425)

[Figure 2.11:E. Available and required at H=5km 26](#_Toc80423426)

[Figure 2.11:F. Available and required at H=6km 27](#_Toc80423427)

[Figure 2.11:G. Available and required at H=7km 27](#_Toc80423428)

[Figure 2.11:H. Available and required at H=8km 27](#_Toc80423429)

[Figure 3.1:A Forces acting on steady climb flight 29](#_Toc80423430)

[Figure 3.2:A. Rate of climb vs Velocity for various altitudes 32](file:///C:\Users\Suresh%20Ragireddy\OneDrive\Desktop\report9.docx#_Toc80423431)

[Figure 3.2:B. Maximum rate of climb vs altitude 32](file:///C:\Users\Suresh%20Ragireddy\OneDrive\Desktop\report9.docx#_Toc80423432)

[Figure 3.2:C. Angle of climb vs Velocity for various altitudes 33](file:///C:\Users\Suresh%20Ragireddy\OneDrive\Desktop\report9.docx#_Toc80423433)

[Figure 3.2:D.Velocity at max rate of climb vs altitude 33](file:///C:\Users\Suresh%20Ragireddy\OneDrive\Desktop\report9.docx#_Toc80423434)

[Figure 3.2:E. Velocity at max angle of climb vs altitude 34](file:///C:\Users\Suresh%20Ragireddy\OneDrive\Desktop\report9.docx#_Toc80423435)

[Figure 3.2:F. Maximum angle of climb Vs altitude 34](file:///C:\Users\Suresh%20Ragireddy\OneDrive\Desktop\report9.docx#_Toc80423436)

[Figure 3.2:A. Forces acting on turning Flight 35](file:///C:\Users\Suresh%20Ragireddy\OneDrive\Desktop\report9.docx#_Toc80423437)

[Figure 4.2:A Radius of turn vs Velocity at various altitudes 38](#_Toc80423438)

[Figure 4.2:B.( Rate of turn and radius of turn) Vs Flight velocity 38](file:///C:\Users\Suresh%20Ragireddy\OneDrive\Desktop\report9.docx#_Toc80423439)

[Figure 4.2:C Velocity at Rmin vs altitude 39](file:///C:\Users\Suresh%20Ragireddy\OneDrive\Desktop\report9.docx#_Toc80423440)

[Figure 4.2:D Min radius of turn vs altitude 39](#_Toc80423441)

[Figure 4.2:E rate of turn Vs velocity at various altitudes 40](#_Toc80423442)

[Figure 4.2:F velocity at rate of turn max vs altitude 40](#_Toc80423443)

[Figure 4.2:G maximum rate of turn vs altitude 41](#_Toc80423444)

[Figure 5.1:A Variation of range with altitude 47](file:///C:\Users\Suresh%20Ragireddy\OneDrive\Desktop\report9.docx#_Toc80423445)

[Figure 5.1:B Variation of range with altitude 47](file:///C:\Users\Suresh%20Ragireddy\OneDrive\Desktop\report9.docx#_Toc80423446)

[Figure 5.2:A variation vemax vs altitude 48](file:///C:\Users\Suresh%20Ragireddy\OneDrive\Desktop\report9.docx#_Toc80423447)

[Figure 5.2:B variation of endurance with altitude 49](file:///C:\Users\Suresh%20Ragireddy\OneDrive\Desktop\report9.docx#_Toc80423448)

**Tables**

[Table 2‑A. Variation of stalling speed with altitude 21](file:///C:\Users\Suresh%20Ragireddy\OneDrive\Desktop\report9.docx#_Toc80423603)

[Figure 3.2:B. Maximum rate of climb vs altitude Table 2‑A. Variation of stalling speed with altitude 21](file:///C:\Users\Suresh%20Ragireddy\OneDrive\Desktop\report9.docx#_Toc80423604)

[Table 2‑B. Variation of Vstall, Vmax, Tmax, with altitude 28](file:///C:\Users\Suresh%20Ragireddy\OneDrive\Desktop\report9.docx#_Toc80423605)

[Table 2‑B. Variation of Vstall, Vmax, Tmax, with altitude 28](file:///C:\Users\Suresh%20Ragireddy\OneDrive\Desktop\report9.docx#_Toc80423606)

[Table 3‑A climb performance 31](#_Toc80423607)

[Table 4‑A. variation of turning performance with altitude 37](file:///C:\Users\Suresh%20Ragireddy\OneDrive\Desktop\report9.docx#_Toc80423608)

[Table 5‑A. Performance of Range and Endurance 46](file:///C:\Users\Suresh%20Ragireddy\OneDrive\Desktop\report9.docx#_Toc80423609)

# Abstract

This report contains details of the performance estimation of a medium range jet airplane like B737. The following aspects are considered.

1. Drag polar.
2. Level flight performance - stalling speed, maximum and minimum speeds.
3. Steady climb performance – maximum rate of climb, maximum angle of climb, service ceiling and absolute ceiling.
4. Range and endurance.
5. Steady level co-ordinated turn - minimum radius of turn, maximum rate of turn.
6. Take-off and landing distances.

The report is intended to serve as an example of performance calculation of a typical jet airplane.

# Airplane Details

## Overall Dimensions

Length : 34.32 m

Wingspan : 32.22 m

Height above ground : 11.17 m

Wheelbase : 13.2 m

Wheel track : 5.8 m

## Engine details

Like CFM 56 - 2B

Seal level static thrust : 97.9 kN per engine

Bypass ratio : 6.5 (For which the engine characteristics are given in Ref.3\*)

SFC : 0.6 hr-1 at M = 0.8 and h = 10973 m (36000 ft)

## Weights

Gross weight : 59175 kgf (580506.8 N)

Empty weight : 29706 kgf (291415.9 N)

Fuel weight : 12131 kgf (119005.1 N)

Payload : 17338 kgf (170085.8 N)

Maximum landing weight : 50296 kgf (493403.8 N)

## Wing geometry

Planform shape : Cranked wing

Span : 32.22 m

Area (Sref) : 111.63 m2

Airfoil : NASA - SC (2) series, t/c=14%, Clopt=0.5

Root chord : 5.59 m (Equivalent trapezoidal wing)

Tip chord : 1.34 m (Equivalent trapezoidal wing) Root chord of cranked wing : 7.44 portion of wing with straight

trailing edge : 11.28 m

Mean aerodynamic chord : 3.9 m Quarter chord sweep : 27.69o

Dihedral : 5 o

Twist : 3 o

Incidence : 1.4 o

Taper ratio : 0.24 (Equivalent trapezoidal wing)

Aspect ratio : 9.3

Figure :.(Three-view drawing of the airplane)

## Fuselage geometry

Length : 33 m

Maximum diameter : 3.59 m

## Nacelle geometry

No. of nacelles 2

Nacelle diameter : 1.62 m

Cross sectional area : 2.06 m2

Length of nacelle : 3.3 m (based on B737 Nacelle)

## Horizontal tail geometry

Span : 11.98 m

Area : 28.71 m2

Mean aerodynamic chord : 2.67 m Quarter chord sweep : 32 o

Root chord : 3.80 m

Tip chord : 0.99 m

Taper ratio : 0.26

Aspect ratio 5

## Vertical tail geometry

Span : 6.58 m

Area : 25.43 m2

Root chord : 5.90 m

Tip chord : 1.83 m

Mean aerodynamic chord : 4.22 m Quarter chord sweep : 37 o

|  |  |
| --- | --- |
| Taper ratio | : 0.31 |
| Aspect ratio | : 1.70 |

|  |  |
| --- | --- |
| Other details |  |
| Taper ratio | : 0.31 |
| Aspect ratio | : 1.70 |
| Taper ratio | : 0.31 |
| Aspect ratio | : 1.70 |
| Flight Condit |  |
| Altitude | : 10973 m (36000 ft) |
| Mach number | : 0.8 |
| Kinematic viscosity | : 3.90536 x10-5 m2/s |
| Density | : 0.3639 kg/m3 |
| Speed of sound | : 295.07 m/s |
| Flight speed | : 236.056 m/s |
| Weight of the airplane | : 59175 kgf (580506.8 N) |

*L*

# Steady Level Flight

The discipline of Aircraft Flight Mechanics requires the formulation of relationships between aircraft forces, and aircraft motion. To define motion, it was necessary to define the different airspeeds in the preceding section.

Aircraft have six degrees of freedom - three translational (𝑥, 𝑦, 𝑧), and three rotational (𝜙, 𝜃, 𝜓), and to develop the expressions describing aircraft flight, nine coupled equations are required. This course will get to that point, and those equations will be derived and utilised - but before that, some handy relationships can be defined for flight constrained to a single direction.

## Equation for steady level

Let’s explore what this regime means, and the assumptions we make

* We assume that the aircraft is a **point mass**, whereby we assume that the aircraft dimensions are negligible when compared to the dimensions of motion.
* **Steady** flight means no acceleration, so we can infer from Newton’s first law that the sum of forces acting on the aircraft is zero ∑𝐹⃗ =0. This is the **equilibrium steady flight condition**.

The definition of forces on the aircraft can change depending on the purpose - and it is only convention that we define lift and drag in the directions we do.

The semantics notwithstanding, it is traditional to define four mutually-orthogonal forces - see [Equilibrium Forces](https://www.aircraftflightmechanics.com/AircraftPerformance/SteadyLevelFlight.html#cruiseforces).

* Two aerodynamic; **lift and drag**.
* One propulsive; **thrust**.
* One inertial; **weight**.

A black and white photo of a rifle

Description automatically generated with low confidence

Figure :. Equilibrium Forces

For this regime, it is further assumed that the aerodynamic incidence is small, and that the thrust offset is negligible. Therefore, we assume that lift and weight are *perpendicular* to aircraft motion, and that thrust, and drag are parallel to aircraft motion.

## Vertical Forces

Looking at the vertical forces, , therefore :

rearranging to find flight speed:

Equation above is the Aircraft Speed Equation for steady level flight, and some basic aerodynamic behaviour may be inferred from it:

* Slower flight is possible by reducing wing loading - reducing aircraft mass or increasing wing area. Or by increasing - increasing
* The minimum possible flight speed occurs at - just before stall
* Flight speed may be increased by reducing - by flying at increased altitude  
  Stall speed

From the stall speed may be determined if is known.

## Longitudinal Forces

Looking at the longitudinal forces, , therefore :

Drag estimation is complex and can be performed via a variety of means from datasheets, CFD, wind tunnel testing or - more commonly - a combination of all. A good breakdown of

drag sources is given by McCormick, and a reproduction of the breakdown given is found in the dropdown below - but this is far beyond the complexity required for Aircraft Performance.

The following is an extract from MacCormick:

**Induced Drag** The drag that results from the generation of a trailing vortex system downstream of a lifting surface of finite aspect ratio.

**Parasite Drag** The total drags of an airplane minus the induced drag. Thus, it is the drag not directly associated with the production of lift. The parasite drag is composed of many drag components, the definitions of which follow.

**Skin Friction Drag** The drag on a body resulting from viscous shearing stresses over its wetted surface.

**Form Drag** (Sometimes Called Pressure Drag) The drag on a body resulting from the integrated effect of the static pressure acting normal to its surface resolved in the drag direction.

**Interference Drag** The increment in drag resulting from bringing two bodies in proximity to each other. For example, the total drag of a wing-fuselage combination will usually be greater than the sum of the wing drag, and fuselage drag independent of each other.

**Trim Drag** The increment in drag resulting from the aerodynamic forces required to trim the airplane about its centre of gravity. Usually this takes the form of added induced and form drag on the horizontal tail.

**Profile Drag** Usually taken to mean the total of the skin friction drag and form drag for a two-dimensional aerofoil section.

**Cooling Drag** The drag resulting from the momentum lost by the air that passes through the power plant installation for purposes of cooling the engine, oil, and accessories.

**Base Drag** The specific contribution to the pressure drag attributed to the blunt after-end of a body.

**Wave Drag** Limited to supersonic flow, this drag is a pressure drag resulting from non-cancelling static pressure components to either side of a shock wave acting on the surface of the body from which the wave is emanating.

In flight performance, we assume that the aircraft is operating in the region of linear aerodynamics, and utilise a drag model as given by Equation (6):

were

The first term represents the drag that is independent of aerodynamic incidence, whilst the second term is proportional to , which represents the induced drag and the component of form drag that varies with incidence.

The parameter for most aircraft, and is the wing aspect ratio usually assumed constant, but can depend on:

* configuration changes (flap deployment)
* Reynolds Number (speed and height)
* Compressibility (shock waves)

Or sometimes can be presented as the Oswald efficiency factor, , which can be related to through the aspect ratio.

## Drag polar

The relationship between lift and drag is given through the aircraft drag polar - often plotted as vs . Values for and are commonplace in the literature - a quick search yielded data for the Cessna

You will see that the drag model works well for low values of lift - that is, in the linear aerodynamic region. The behaviour at the ‘bottom’ of the curve is sometimes called the ‘drag bucket’ - and I spent many years *collecting* these data in a wind tunnel.

The drag polar shows the aerodynamic efficiency of a given aircraft - that is, it represents the lift-to-drag ratio. You can find the best (highest) lift to drag ratio as the tangent of a line drawn from the origin to the curve.

It would be great to be able to find the best lift to drag ratio without having to draw the drag polar. If you look at the source code for the image below, then it’s obvious that I haven’t drawn the tangent to find the best lift to drag ratio - rather I have drawn the tangent after having found the best lift to drag ratio.

Expand the code to see the source data and coefficients used in the drag model.

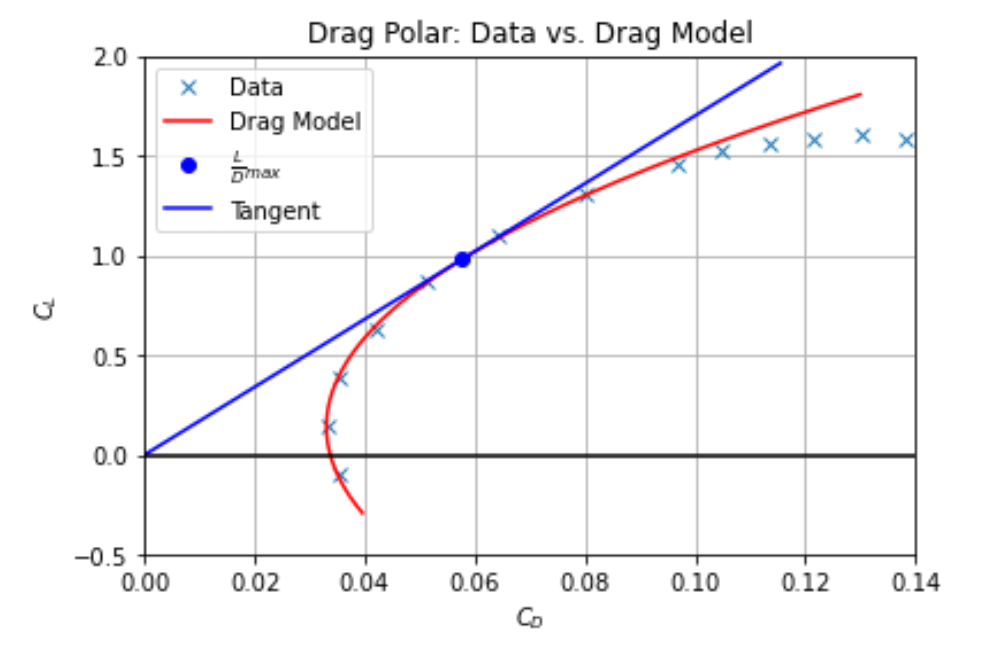


Figure :. Drag polar

## Best lift to drag ratio

In the following, the values of and for optimum aerodynamic efficiency - but before diving into the mathematics, some consideration of their significance might be necessary for some readers.

The best lift to drag ratio can be determined. Consider that the lift to drag ratio is equal in dimensional and coefficient form:

Into which Equation (6) may be inserted

The best lift to drag ratio can then be found from minimising , so differentiate by :

A bit of elementary calculus gives the minima of the right-hand side as given by

## Thrust required

The drag equation can be multiplied by the numerator of the lift and drag coefficients, , to yield dimensional drag - this can be considered as the thrust required to fly at a certain speed:

In steady level flight, , so the lift coefficient can be expressed as

and hence

where and are functions of density (and therefore functions of altitude).

* represents the profile drag, which gets larger with forward speed squared
* represents the induced drag, which gets smaller with forward speed squared.

The above should make sense to you intuitively. Profile drag is largely viscous drag, which will get larger in proportion to the dynamic pressure. The  
induced drag is proportional to the bound vortex maintaining lift, which will be proportional to which, for a steady flight, is inversely proportional to the dynamic pressure.

## Minimum drag speed

From the drag equation in dimensional form, the minimum drag speed can be shown

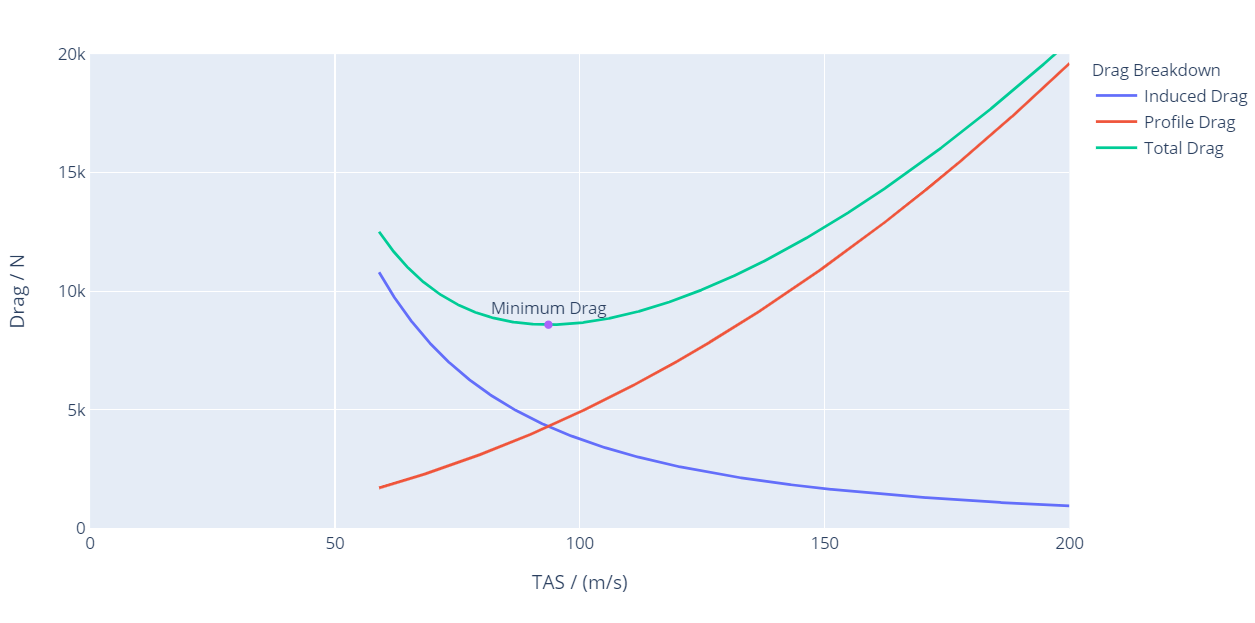


Figure : Thrust required vs TAS

## Drag polar for B737

The drag polar is assumed to be of the form:

The quantity is assumed to be given by:

where suffices WB, V, H, Misc. denote wing-body combination, vertical tail,  
horizontal tail, and miscellaneous contributions respectively.

Therefore, calculating Drag polar is complex at present, I have directly taken from openVSP software**.** We calculatemanuallyfrom the above data

But my goal for this project is to draw the performance plots. Indeed, I am taking directing the drag polar value

**CD = 0.0159 + 0.04244 *CL* 2** ...…………. (a)

Where, CD0 = 0.0159 & K = 0.04244

## Remarks

* The polar given above is valid at subcritical Mach numbers. The increase in CDO  and K at higher Mach numbers is not considered by me for this project.
* The maximum lift to drag ratio is given by:

Using and above equation, we get (L/D) max is , which is typical of modern jet transport airplanes.

* It may be noted that the parabolic polar is an approximation and is not valid beyond  
  . It is also not accurate close to and

## Stalling speed

In level flight,

Since, cannot exceed , there is a flight speed below which level  
flight is not possible. The flight speed at is called the stalling  
speed and is denoted by

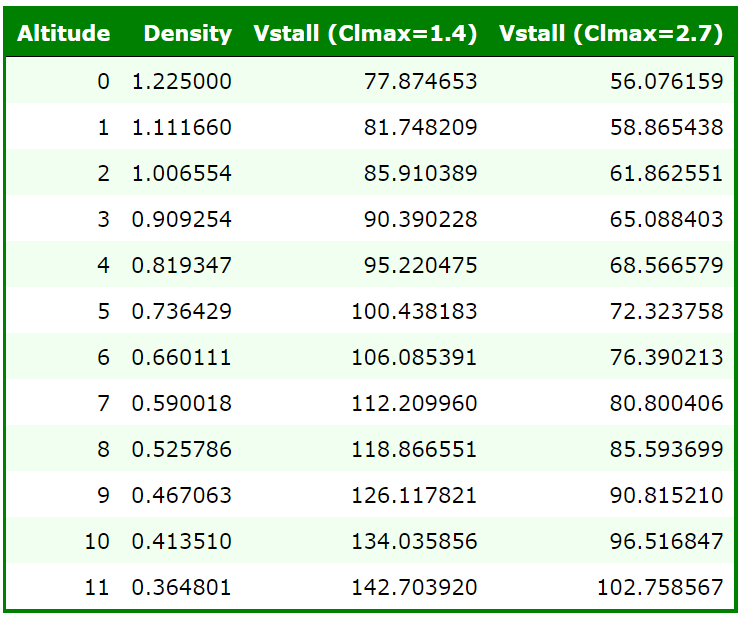
Since, decreases with altitude, increases with height. It may be noted that  
 with landing flaps and without flaps. The  
values of stalling speed at different altitudes and flap settings are tabulated in Table 1 and

Table ‑. Variation of stalling speed with altitude

Figure :. Maximum rate of climb vs altitude Table ‑. Variation of stalling speed with altitude

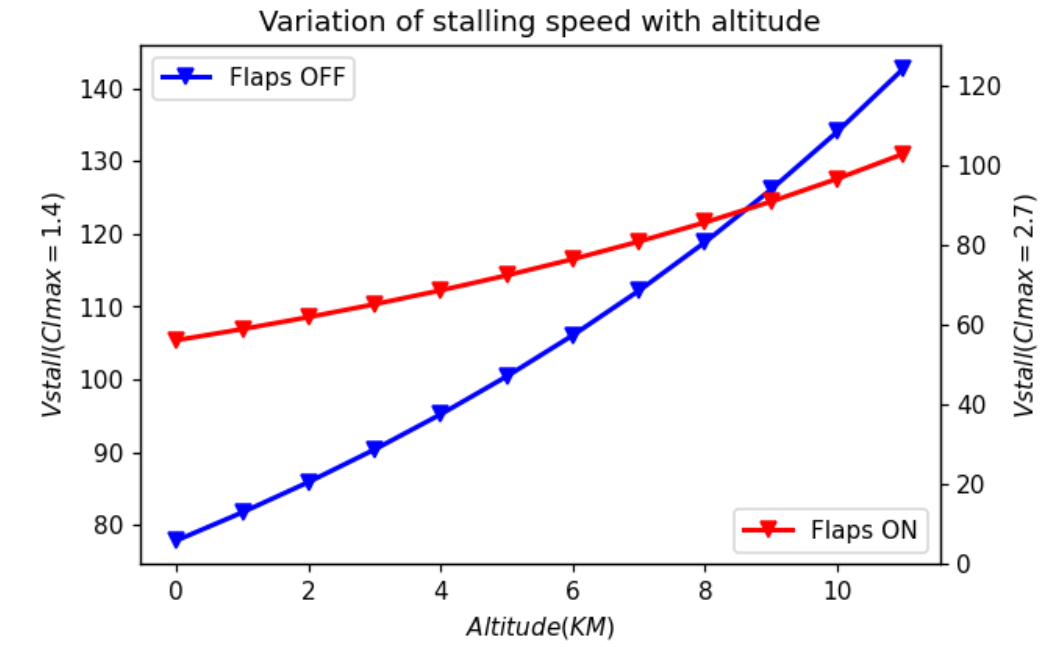


Figure :. Variation of stalling with altitude

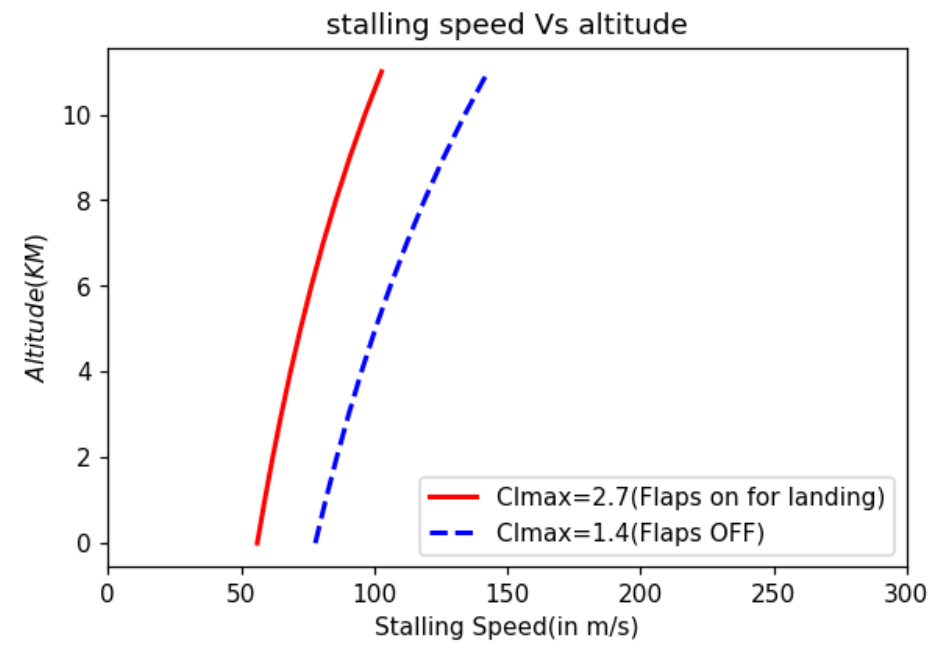


Figure :. stalling speed Vs Altitude

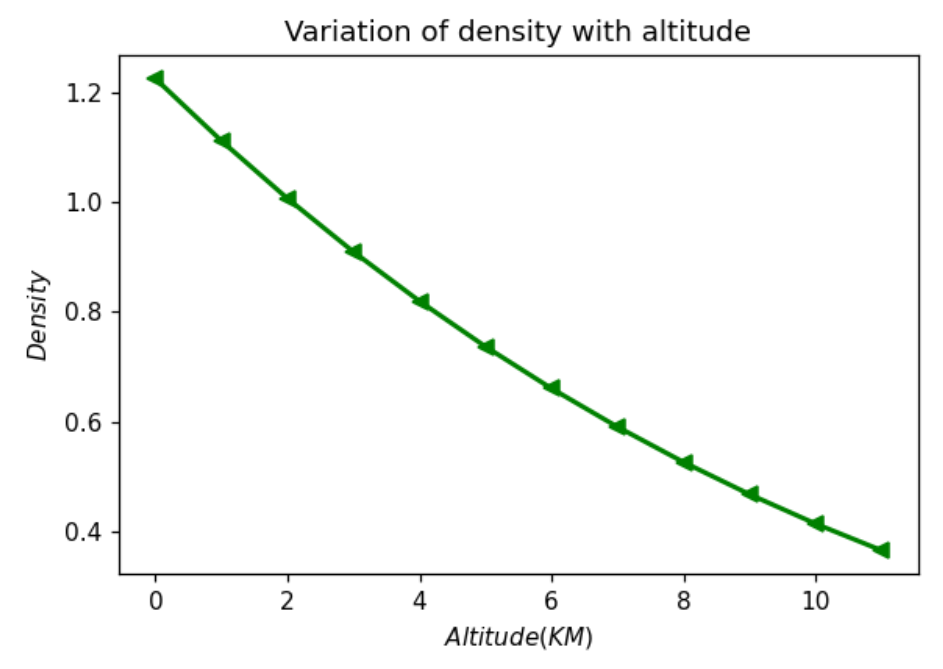


Figure :. Variation of density with Altitude

## Variations of and with altitude

To determine the and at each altitude, the following procedure is adopted. The  
engine thrust as a function of velocity at each altitude is obtained from the smoothed data.  
The drag at each altitude is obtained as a function of velocity using the drag polar and the  
level flight formulae given below.

Thrust available

The variation of Thrust available, according to my approximation are calculated at various altitudes are followed by following relations:

At sea level:

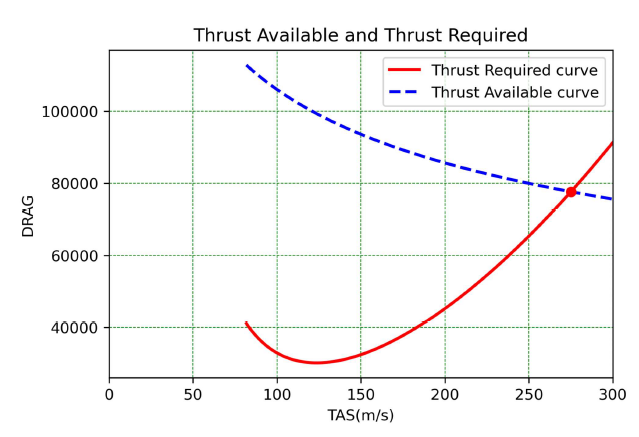
At

Recall that is the thrust at sea level at zero velocity.

NOTE:

* These are approximate formulas for finding the Tavailable , you can go by engine characteristics by following research papers and find the Tavailable, at various altitudes.
* The plots which I have plotted are done in python, and the code is attached at the end of my report.  
  where, and .

(Engine specifications)  
However, the cruise Mach number for this airplane is . Hence, and are expected to become functions of Mach number above To get some guidelines



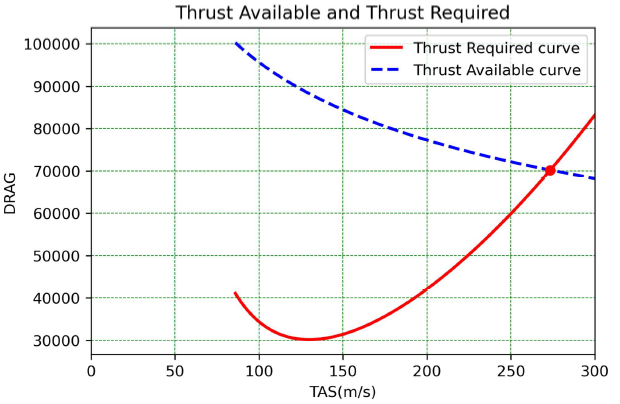
Figure 2.11:A. Available and required at H=0km

Figure 2.11:B. Available and required at H=1km

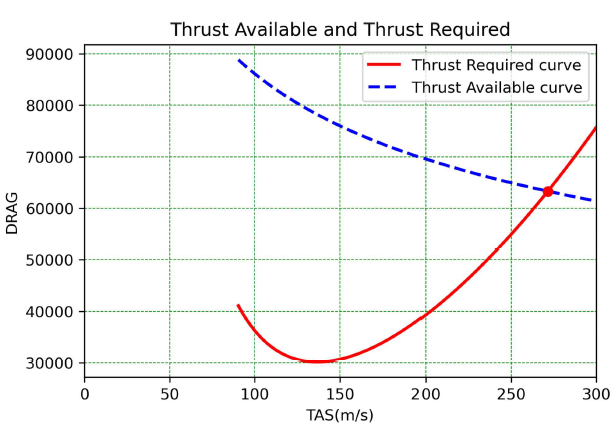
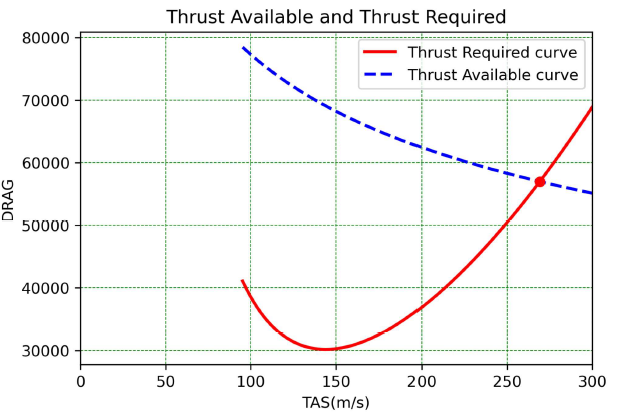
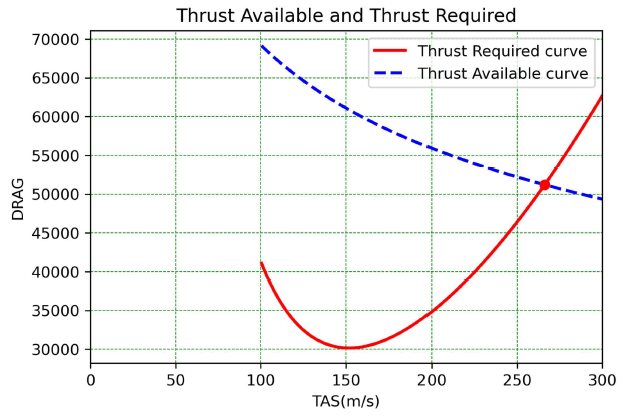
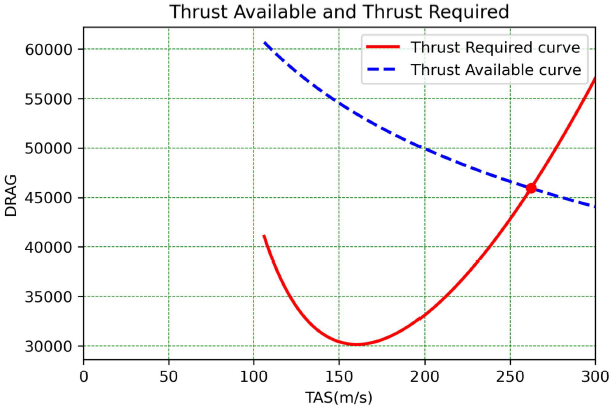
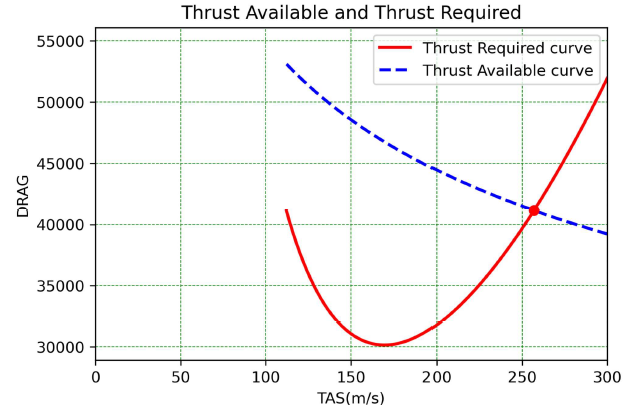


Figure 2.11:C. Available and required at H=2km



Figure 2.11:D. Available and required at H=4km

Figure 2.11:E. Available and required at H=5km

Figure 2.11:F. Available and required at H=6km

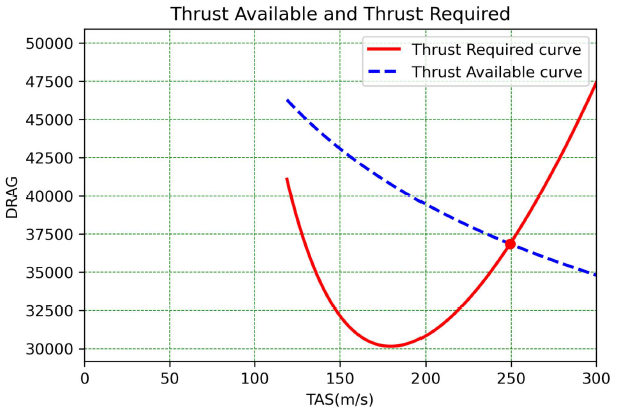
Figure 2.11:G. Available and required at H=7km

Figure 2.11:H. Available and required at H=8km

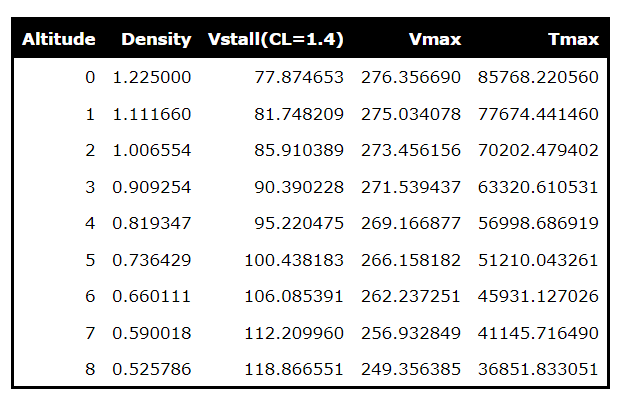


Table 2‑C. Variation of Vstall, Vmax, Tmax, with altitude

Table 2‑D. Variation of Vstall, Vmax, Tmax, with altitude

# Steady climb

## Equations of steady climb

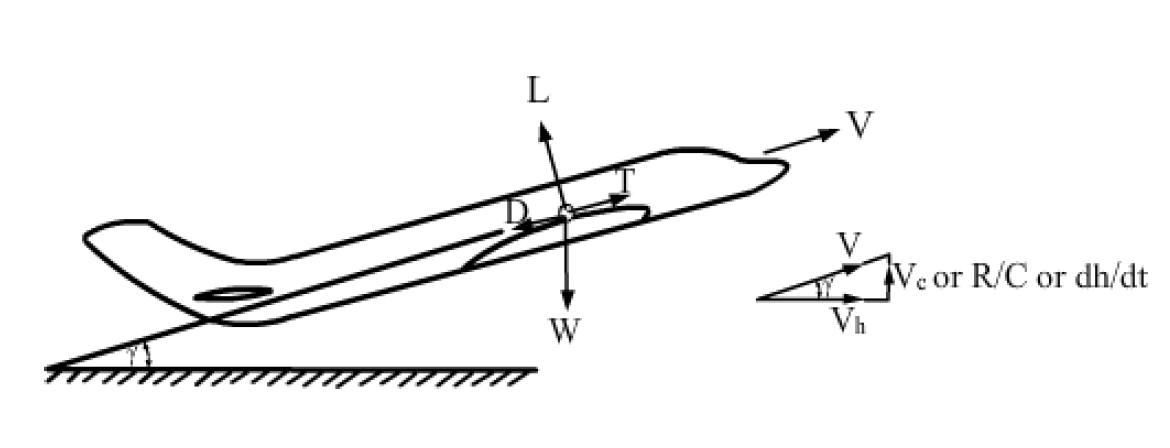


Figure : Forces acting on steady climb flight

In this flight, the C.G of the airplane moves along a straight line inclined to the horizontal  
at an angle . The velocity of flight is assumed to be constant during the climb. Since,  
the flight is steady, the acceleration is zero and the equations of motion can be written as:

## Steady climb performance

To calculate the variation of rate of climb with flight velocity at different altitudes, the  
following procedure is adopted.  
Choose an altitude.  
Choose a flight speed.  
Noting that

, gives:

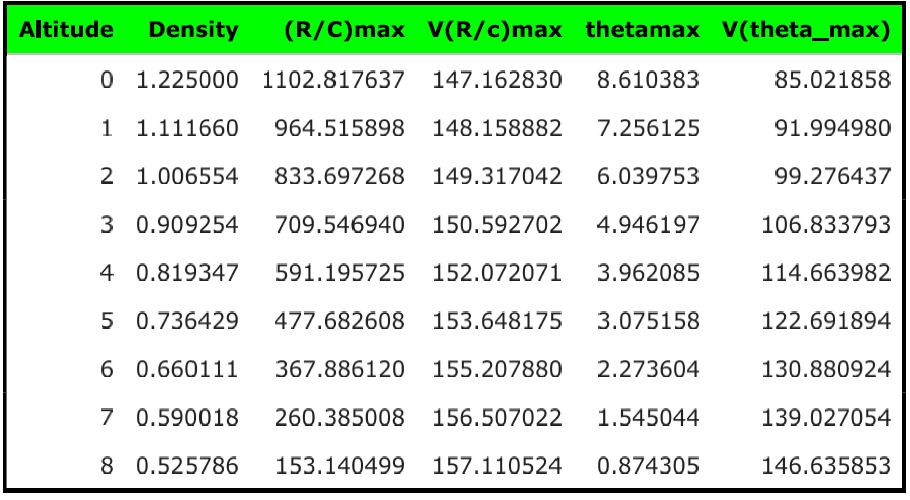
Also,  
Hence,

Substituting various quantities in Eq. yields:  
   
Or

Since, altitude and flight velocity have been chosen, the thrust available is read from the  
climb thrust curves in Fig@2AtoG. Further, the variation of and with Mach number is taken. The value which is less than is chosen, as cannot be greater than unity. Hence,

This procedure is repeated for various speeds between and . The entire  
procedure is then repeated for various altitudes. The variations of and with  
velocity and with altitude as parameter are shown in Figs3: b and 3: c. The variations of  
 and with altitude are shown in Figs 3: d and 3: e. The variations of   
and with altitude are shown in Figs.3: f and 3: g. A summary of results is presented in  
Table

Table ‑ climb performance



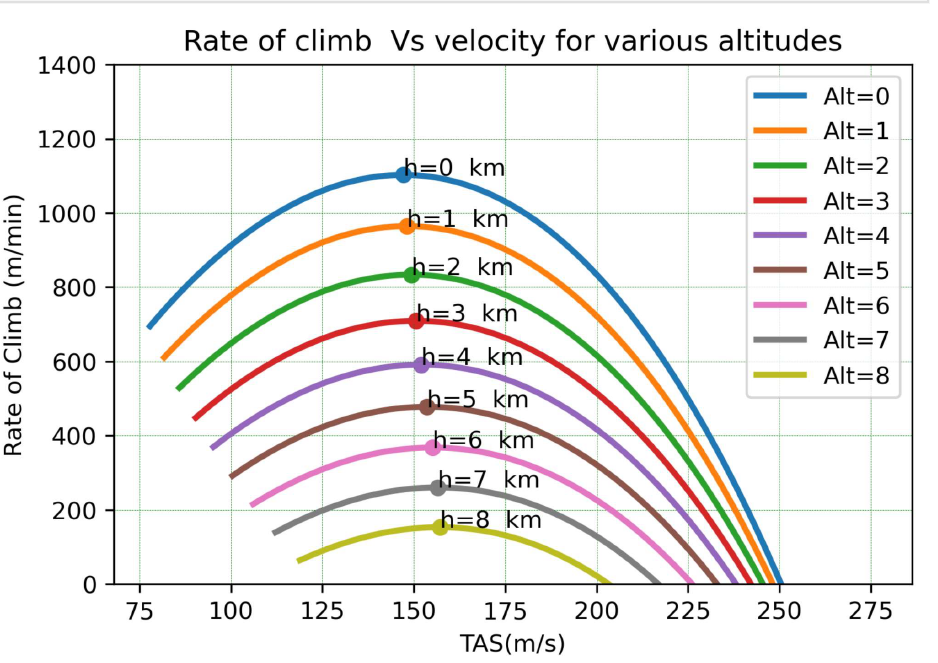


Figure :. Rate of climb vs Velocity for various altitudes

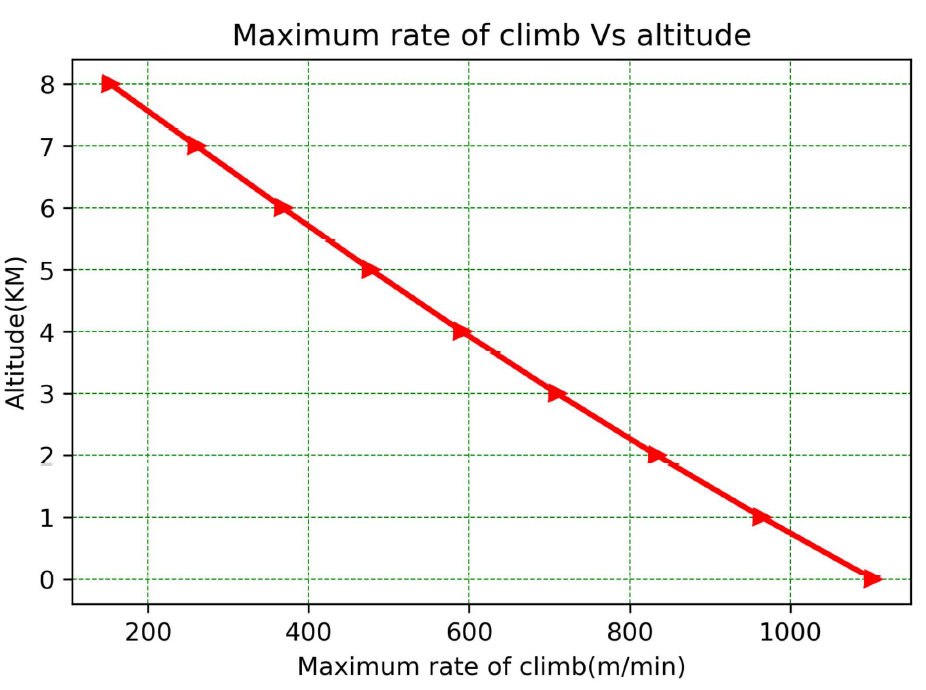


Figure :. Maximum rate of climb vs altitude

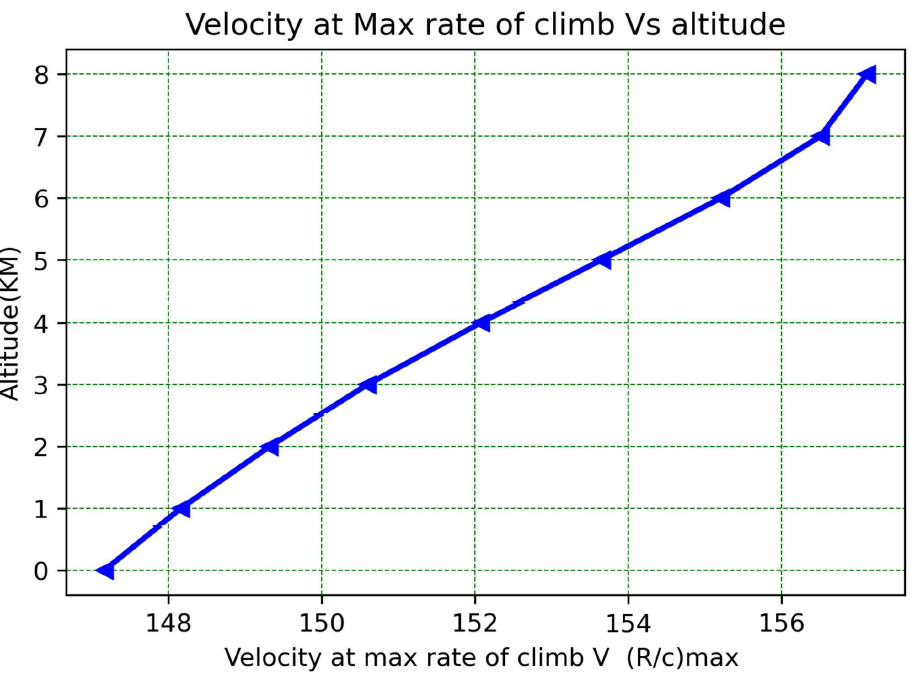
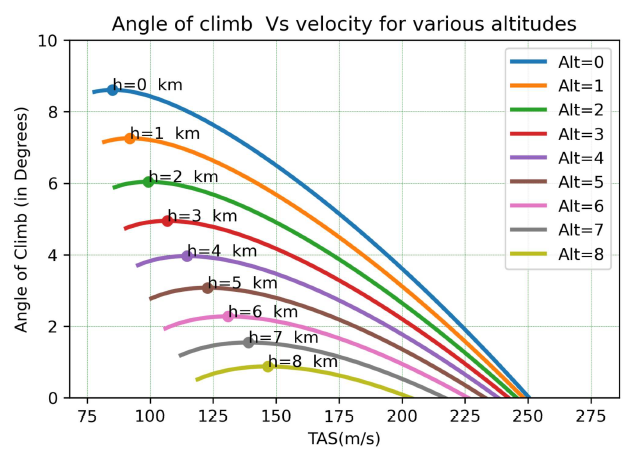


Figure :. Angle of climb vs Velocity for various altitudes

Figure :.Velocity at max rate of climb vs altitude

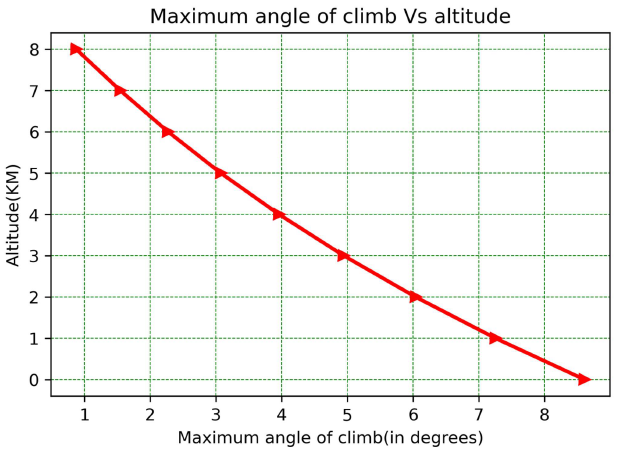
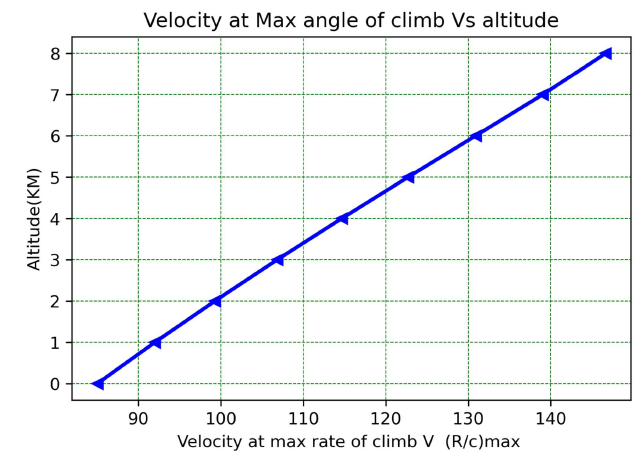


Figure :. Velocity at max angle of climb vs altitude

Figure :. Maximum angle of climb Vs altitude

# Turning performance.

## Equations of turning fligth

Figure :. Forces acting on turning Flight

In this section, the performance of the airplane in a steady level, co-ordinated turn is  
studied. The equations of motion in this case are:

## Turning performance

where is the angle of bank? These equations give:

where, is the rate of turn and is the radius of turn.  
The following procedure is used to obtain and .

1. A flight speed and altitude are chosen and the level flight lift coefficient  
    is obtained as:
2. If , where is the maximum load factor for which the airplane is  
   designed, then the turn is limited by and . However, if  
   , then the turn is limited by , and .
3. From the drag polar, is obtained corresponding to . Then,

If , where is the available thrust at that speed and altitude, then the turn is limited by the engine output. In this case, the maximum permissible value of in  
turning flight is found from

From drag polar, the value of is calculated as

However, if , then the turn is not limited by the engine output and the value of  
 calculated in step (2) is retained.  
4. Once is known, the load factor during the turn is determined as

Once is known, the values of and can be calculated using the equations given  
above.

The above steps are repeated for various speeds and altitudes. A typical turning flight  
performance estimation is presented in Table In these calculations, and  
 are assumed. The variation of turning performance with altitude is shown in  
Table 4.2-A.

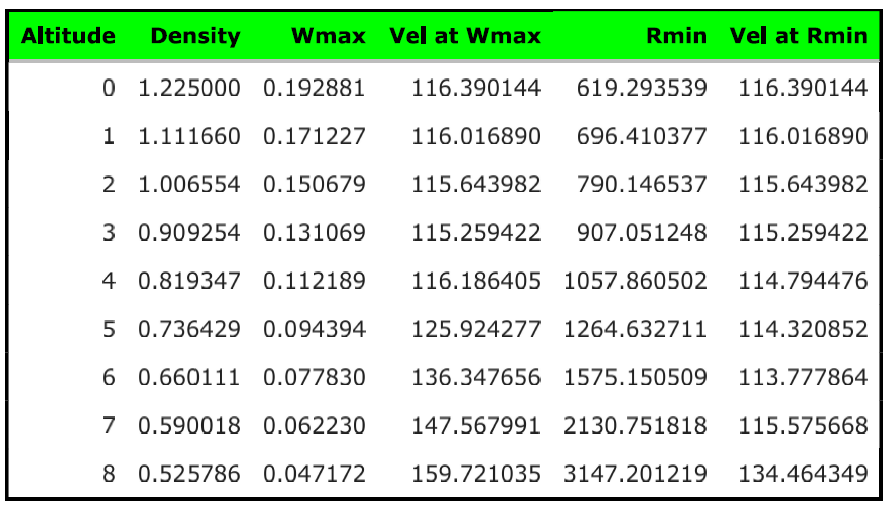


Table ‑. variation of turning performance with altitude

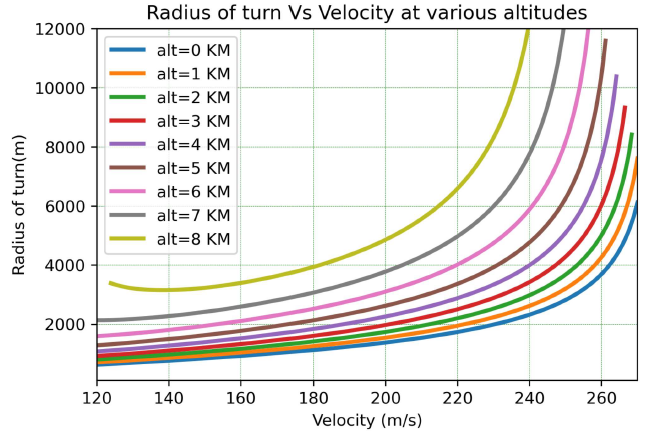


Figure : Radius of turn vs Velocity at various altitudes

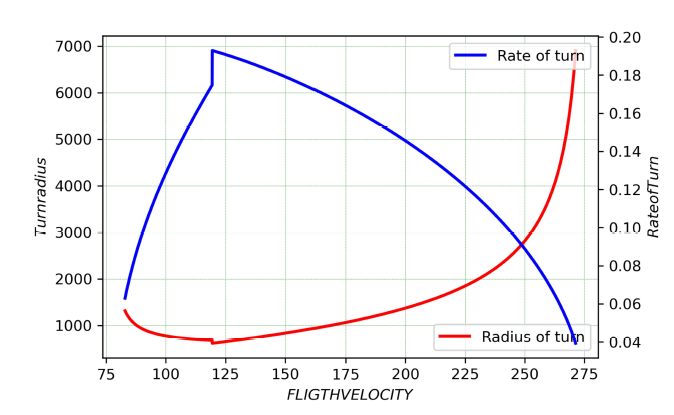


Figure :.( Rate of turn and radius of turn) Vs Flight velocity



Figure : Velocity at Rmin vs altitude

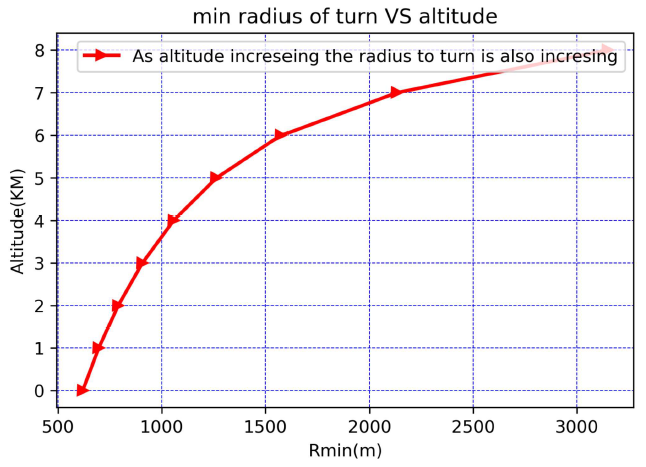


Figure : Min radius of turn vs altitude



Figure : rate of turn Vs velocity at various altitudes

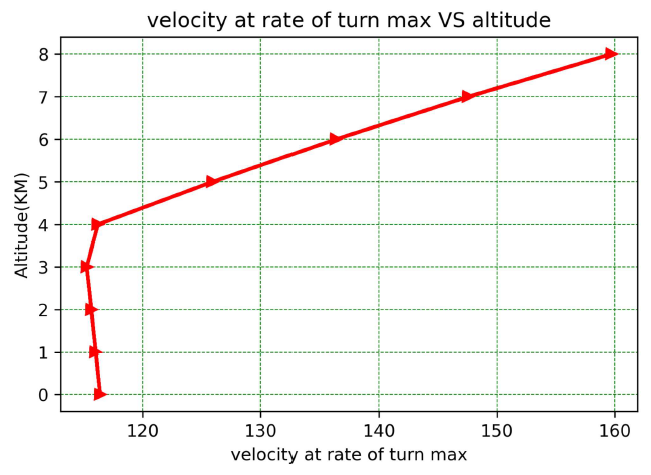
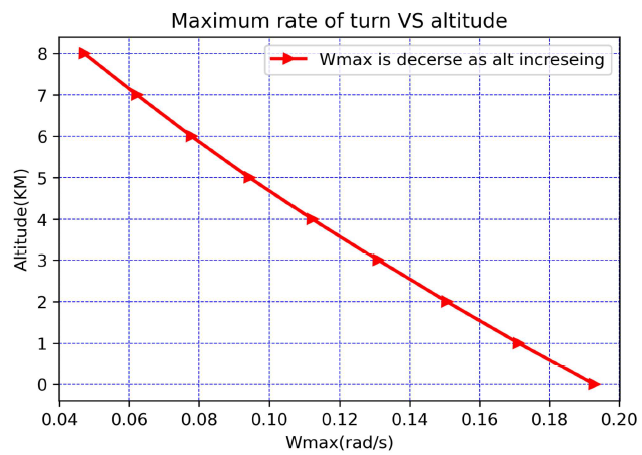
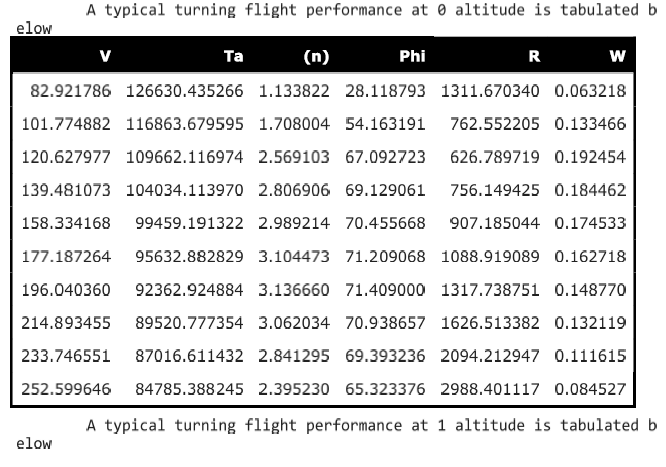
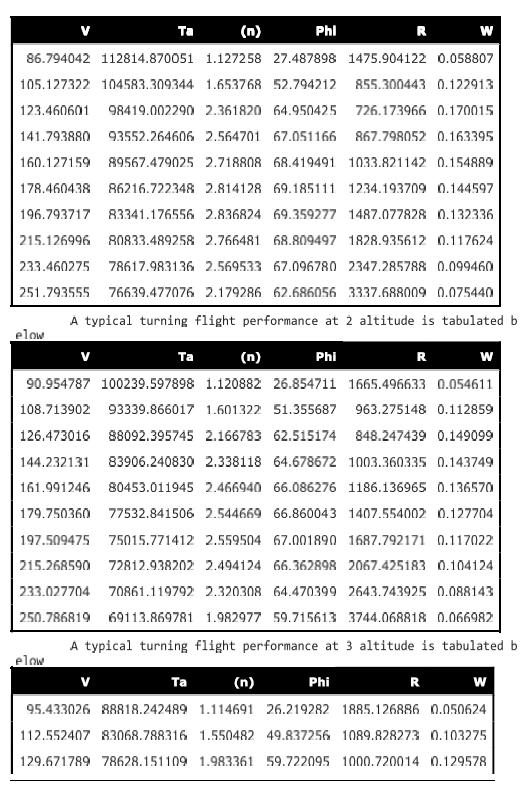
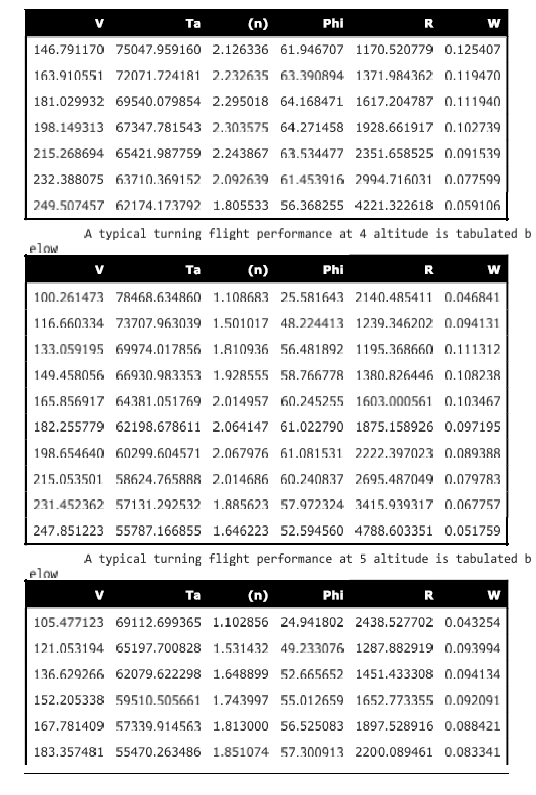


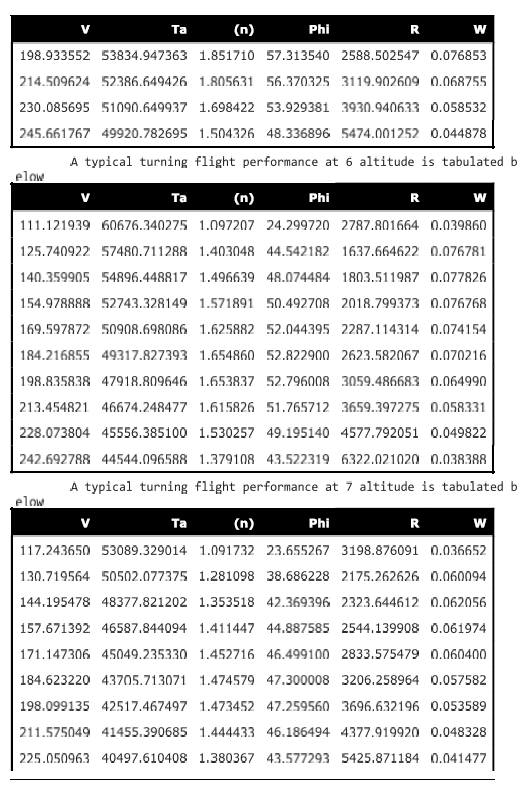
Figure : velocity at rate of turn max vs altitude

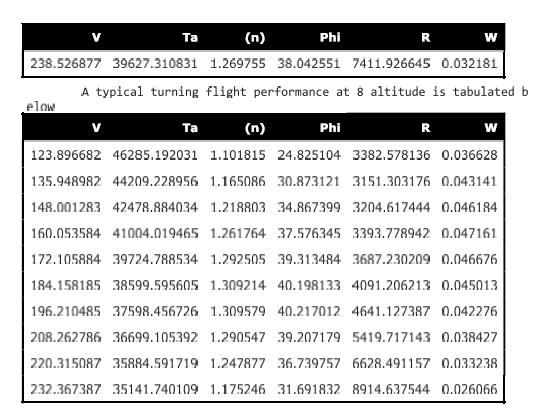


Figure : maximum rate of turn vs altitude









# Range and Endurance

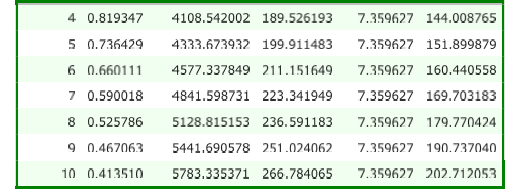
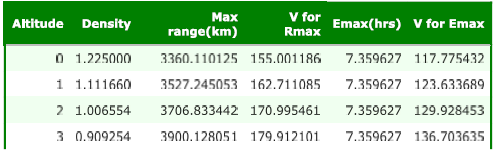
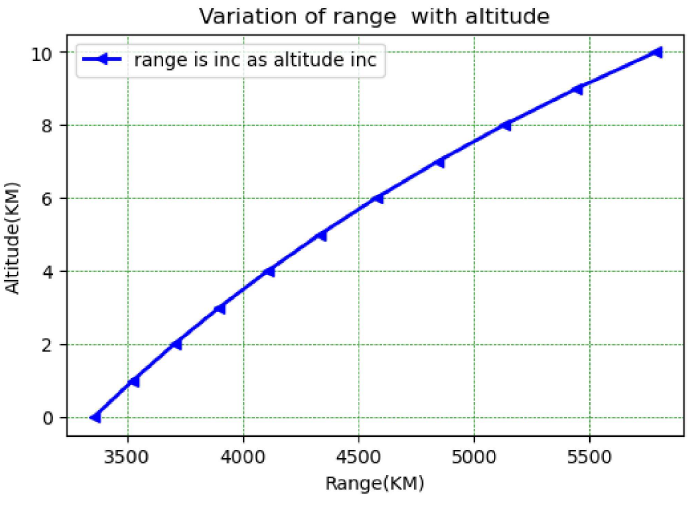


Table ‑. Performance of Range and Endurance

## Range for jet propelled Aircraft

Equation is a simplified range equation for a jet-propelled airplane. From this equation, the flight conditions for maximum range for a jet-propelled airplane are

1. Fly at maximum .
2. Have the lowest possible thrust specific fuel consumption.
3. Fly at high altitude, where is small.
4. Carry a lot of fuel.

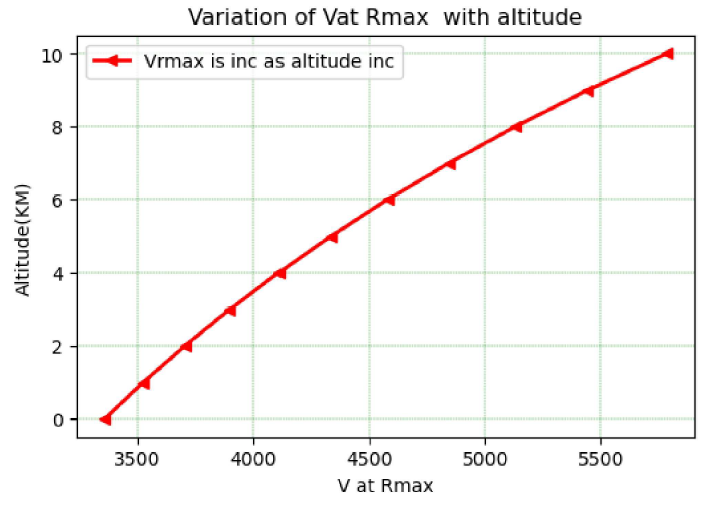


Figure : Variation of range with altitude

Figure : Variation of range with altitude

## Endurance for jet propelled aircraft

Equation is already expressed in terms of thrust specific fuel consumption, and it gives the endurance for a jet-propelled airplane directly. We repeat Eq. for convenience:

Note from Eq. (5.167) that maximum endurance for a jet-propelled airplane corresponds to the following conditions:

1. Fly at maximum
2. Have the lowest possible thrust specific fuel consumption.
3. Have the highest possible ratio of to (i.e., carry a lot of fuel).

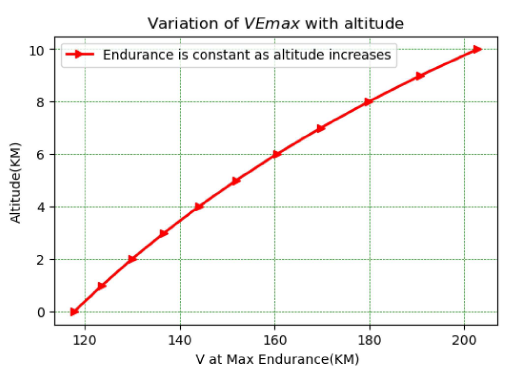


Figure : variation VEmax vs altitude

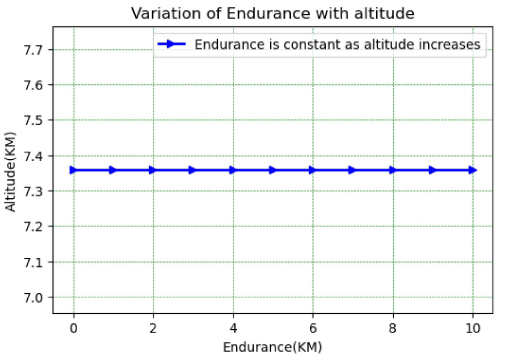
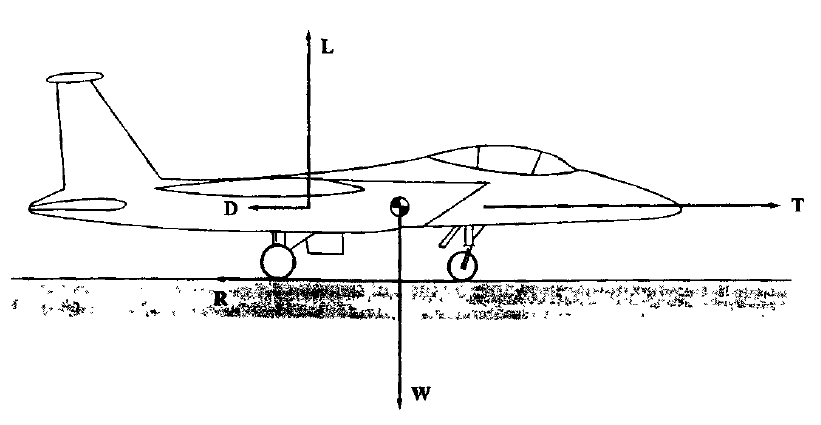


Figure : variation of endurance with altitude

# Take off

## Calculation of Ground Roll

  
The forces acting on the airplane during take-off. In addition to the familiar forces of thrust, weight, lift, and drag, there is a rolling resistance , caused by friction between the tires and the ground. This resistance force is given by

where , is the coefficient of rolling friction and is the net normal force exerted hetween the tires and the ground. Summming forces parallel to the ground and employing Newton's second law, we have from

This is for two primary reasons: (1) With the landing gear fully extended, is larger than when the landing gear is retracted; and (2) there is a reduction in the induced drag due to the proximity of the wings to the ground-part of the "ground effect." An approximate expression for the increase in due to the extended landing gear is given as

where is the wing loading, is the maximum mass of the airplane, and the factor depends on the amount of flap deflection? With flap deflection, the average airflow velocity over the bottom of the wing is lower than it would be with no flap deflection; that is, the deflected flap partially blocks the airflow over the bottom surface. Hence, the landing gear drag is less with flap deflection than its value with no flap deflection. When is in units of newtons per square meter and is in units of kilograms, for a zero-flap deflection and for maximum flap deflection. These values are based on correlations for several civil transports and are approximate only. Regarding the induced drag during the ground roll, the downwash is somewhat inhibited by the proximity of the ground, and hence the induced drag contribution is less than that included in Eq. (2.47); that is, for the ground roll, must be reduced below that for the airplane in flight. The reduction in the induced drag coefficient can be approximated by the relation from Ref. 50 given here

where is the height of the wing above the ground and is the wingspan?

or

Let us now construct an appropriate expression for to be inserted into Eq... Returning

, we have

or

In Eq., is given by Eq. Hence, recalling that , we have

This is the expression for that is inserted into Eq... To simplify the appearance of the following equations, we define the symbols and as

Integrating Eq between where and where , we have

consider into be a constant equal to its value at

Thus, by assuming and are constant in performing the integration, and allowing a distance equal to for the rotation phase (where for large aircraft and for small aircraft), the ground roll can be approximated by

With Eq., a quick analytical evaluation of the ground roll can be made. An analytic form for that more clearly illustrates the design parameters that govern takeoff performance can be obtained by

Integrating Eq. (6.91) from point 0 to lift off, and again noting that , we have

In , is the net force acting in the horizontal direction on the airplane during takeoff. In Fig. , a schematic 1 shown of the variation of the forces acting during take-off as a function of distance along the ground. Note that the net force , specifically identified in, does not vary greatly. This gives some justification to assurning that the expression is constant up to the point of rotation.

Net force value equal to its value at , then Eq. is easily integrated, giving

where the term has been added to account for that part of the ground roll during rotation, as noted earlier.

The velocity at lift off should be no less than

(C is that value with the flaps extended for takeoff; also keep in mind that may be a smaller value if the angle of attack is limited by tail clearance with the ground. Setting and inserting in Eq.

## Calculation of Distance While Airborne to Clear an Obstacle

## Total take off distance

After calculation, total take-off distance = 1311.14 meters

(Note: go through code)

# Landing

## Calculation of Approach Distance

## Calculation of flare distance

## Calculation of Ground Roll

## Total landing distance

After calculation, total landing distance = 1075.61 meters

(Note: go through code)

# Python code

## Steady level flight

### Stalling speed code

**import** **numpy** **as** **np**

**import** **pandas** **as** **pd**

**import** **matplotlib.pyplot** **as** **plt**

**import** **math**

**from** **ambiance** **import** Atmosphere

**from** **shapely.geometry** **import** LineString

w=580506.8

s=111.63

clmax1=1.4 *# flaps off*

clmax2=2.7 *#flaps on*

CD0=0.0159

K=0.04244

K1=0.41065

K2=-0.3078

T0=(2\*97900)

view=100

alt=[0,1,2,3,4,5,6,7,8,9,10,11]

rho=np.zeros(len(alt))

Vs1=np.zeros(len(alt))

Vs2=np.zeros(len(alt))

**for** j **in** alt:

mosphere = Atmosphere(alt[j]\*1000)

*#density at various altitudes calculations*

rho[j] = mosphere.density

A=CD0\*0.5\*rho[j]\*s

B=(K\*(w\*\*2))/(rho[j]\*0.5\*s)

*#vstall at vaious altitudes*

Vs1[j]=((2\*w)/(rho[j]\*clmax1\*s))\*\*(0.5)

Vs2[j]=((2\*w)/(rho[j]\*clmax2\*s))\*\*(0.5)

dfa=pd.DataFrame({ 'Altitude':alt,'Density':rho,'Vstall (Clmax=1.4)':Vs1,'Vstall (Clmax=2.7)':Vs2})

*#plotting code*

fig,ax1=plt.subplots(num=1,dpi=view)

ax1.set\_xlabel('$Altitude(KM)$')

ax1.set\_ylabel('$Vstall (Clmax=1.4)$')

ax1.plot(alt,Vs1,label='Flaps OFF',linewidth=2,color='blue',marker='v')

ax2=ax1.twinx()

ax2.set\_ylabel('$Vstall (Clmax=2.7)$')

ax2.plot(alt,Vs2,label='Flaps ON',linewidth=2,color='red',marker='v')

*#fig.tight layout()*

plt.title("Variation of stalling speed with altitude")

plt.figure()

ax1.legend(loc ="upper left")

ax2.legend(loc ="lower right")

ax2.set\_ylim([0, 130])

plt.show()

*#method 2 to show*

plt.figure(num=3,dpi=view)

plt.plot(Vs2,alt,label='Clmax=2.7(Flaps on for landing)',linewidth=2,color='red')

plt.plot(Vs1,alt,label='Clmax=1.4(Flaps OFF)',linewidth=2,color='blue',linestyle='dashed')

plt.title("stalling speed Vs altitude")

plt.ylabel('$Altitude(KM)$')

plt.xlabel('Stalling Speed(in m/s)')

plt.legend()

plt.xlim([0,300])

*#printing table*

*# use DFA*

*# use DFA to print table*

plt.figure(num=2,dpi=view)

plt.plot(alt,rho,linewidth=2,color='green',marker='<')

plt.title("Variation of density with altitude")

plt.xlabel('$Altitude(KM)$')

plt.ylabel('$Density$')

### Variations of and with altitude code

**import** **numpy** **as** **np**

**import** **pandas** **as** **pd**

**import** **matplotlib.pyplot** **as** **plt**

**import** **math**

**from** **ambiance** **import** Atmosphere

**from** **shapely.geometry** **import** LineString

**import** **sympy** **as** **sp**

alt=[0,1,2,3,4,5,6,7,8]

*#alt=np.linspace(0,7,8)*

n=100*# size of velocity array*

w=580506.8

s=111.63

clmax=1.4

CD0=0.0159

K=0.04244

K1=0.41065

K2=-0.3078

view=200

a=340

Tsv=(2\*97900) *#static thrust at sea level*

v=np.zeros([len(alt),n])

Tr=np.zeros([len(alt),n])

Ta=np.zeros([len(alt),n])

rho=np.zeros(len(alt))

tem=np.zeros(len(alt))

Vmax1=[0]\*len(alt)

Tmax1=[0]\*len(alt)

y=np.zeros(len(alt))

a=np.zeros(len(alt))

Vs=np.zeros(len(alt))

**def** f(V):

**return** (A\*V\*\*2)+(B\*V\*\*(-2))

**def** ff(T):

**return** (1.4\*288\*T)\*\*(0.5)

**def** fff(X):

**return** Tsv\*(X/1.22500002)

**for** i **in** range(len(alt)):

mosphere = Atmosphere(alt[i]\*1000)

*#density at various altitudes calculations*

rho[i] = mosphere.density

tem[i]= mosphere.temperature

a1=ff(tem[i])

a[i]=a1

A=CD0\*0.5\*rho[i]\*s

B=(K\*(w\*\*2))/(rho[i]\*0.5\*s)

*#vstall at vaious altitudes*

Vs[i]=((2\*w)/(rho[i]\*clmax\*s))\*\*(0.5)

v[i]=np.linspace(Vs[i],310,n)

**for** j **in** range(n):

z=f(v[i][j])

Tr[i][j]=z

*#thrust available*

M=v[i][j]/a[i]

TVV=fff(rho[i]) *# thrust variation logic*

yy=TVV\*K1\*M\*\*(K2)

Ta[i][j]=yy

*# plt.plot(v[i],Tr[i])*

*# plt.plot(v[i],Ta[i])*

plt.figure(num=3,dpi=view)

plt.plot(v[i],Tr[i],label='Thrust Required curve',linewidth=2,color='red')

plt.plot(v[i],Ta[i],label='Thrust Available curve',linewidth=2,color='blue',linestyle='dashed')

plt.title("Thrust Available and Thrust Required")

plt.ylabel('DRAG')

plt.xlabel('TAS(m/s)')

plt.legend()

plt.xlim([0,300])

plt.grid(color = 'green', linestyle = '--', linewidth = 0.5)

line\_1 = LineString(np.column\_stack((v[i],Tr[i])))

line\_2 = LineString(np.column\_stack((v[i],Ta[i])))

intersection = line\_1.intersection(line\_2)

plt.plot(\*intersection.xy, 'ro')

plt.show()

x,y = intersection.xy

**from** **array** **import** \*

Vmax1[i]= x.tolist()

Tmax1[i]= y.tolist()

print("Vmax and Thrust available at %s km Altitude is %s and %s respectively." % (alt[i],Vmax1[i],Tmax1[i]))

## Steady climb and Turning flight code

**import** **numpy** **as** **np**

**import** **pandas** **as** **pd**

**import** **matplotlib.pyplot** **as** **plt**

**import** **math**

**from** **ambiance** **import** Atmosphere

**from** **shapely.geometry** **import** LineString

**import** **sympy** **as** **sp**

alt=[0,1,2,3,4,5,6,7,8]

*#alt=np.linspace(0,7,8)*

n=4000*# size of velocity array for turn n=5 keep for printing tables*

w=580506.8

s=111.63

clmax=1.4

CD0=0.0159

K=0.04244

K1=0.41065

K2=-0.3078

g=9.81

view=100

a=340

Tsv=(2\*97900) *#static thrust at sea level*

v=np.zeros([len(alt),n])

Tr=np.zeros([len(alt),n])

Ta=np.zeros([len(alt),n])

rho=np.zeros(len(alt))

tem=np.zeros(len(alt))

Vmax1=[0]\*len(alt)

Tmax1=[0]\*len(alt)

y=np.zeros(len(alt))

a=np.zeros(len(alt))

Vs=np.zeros(len(alt))

**def** f(V):

**return** (A\*V\*\*2)+(B\*V\*\*(-2))

**def** ff(T):

**return** (1.4\*288\*T)\*\*(0.5)

**def** fff(X):

**return** Tsv\*(X/1.22500002)

**for** i **in** range(len(alt)):

mosphere = Atmosphere(alt[i]\*1000)

*#density at various altitudes calculations*

rho[i] = mosphere.density

tem[i]= mosphere.temperature

a1=ff(tem[i])

a[i]=a1

A=CD0\*0.5\*rho[i]\*s

B=(K\*(w\*\*2))/(rho[i]\*0.5\*s)

*#vstall at vaious altitudes*

Vs[i]=((2\*w)/(rho[i]\*clmax\*s))\*\*(0.5)

v[i]=np.linspace(Vs[i],310,n)

**for** j **in** range(n):

z=f(v[i][j])

Tr[i][j]=z

*#thrust available*

M=v[i][j]/a[i]

TVV=fff(rho[i]) *# thrust variation logic*

yy=TVV\*K1\*M\*\*(K2)

Ta[i][j]=yy

*# plt.plot(v[i],Tr[i])*

*# plt.plot(v[i],Ta[i])*

plt.figure(num=3,dpi=view)

plt.plot(v[i],Tr[i],label='Thrust Required curve',linewidth=2,color='red')

plt.plot(v[i],Ta[i],label='Thrust Available curve',linewidth=2,color='blue',linestyle='dashed')

plt.title("Thrust Available and Thrust Required")

plt.ylabel('DRAG')

plt.xlabel('TAS(m/s)')

plt.legend()

plt.xlim([0,300])

plt.grid(color = 'green', linestyle = '--', linewidth = 0.5)

line\_1 = LineString(np.column\_stack((v[i],Tr[i])))

line\_2 = LineString(np.column\_stack((v[i],Ta[i])))

intersection = line\_1.intersection(line\_2)

plt.plot(\*intersection.xy, 'ro')

plt.show()

x,y = intersection.xy

**from** **array** **import** \*

Vmax1[i]= x.tolist()

Tmax1[i]= y.tolist()

print("Vmax and Thrust available at %s km Altitude is %s and %s respectively." % (alt[i],Vmax1[i],Tmax1[i]))

arr = np.asarray(Vmax1)

V111=arr.transpose()

Vmax=V111.flatten()

arrr = np.asarray(Tmax1)

T111=arrr.transpose()

Tmax=T111.flatten()

dfb=pd.DataFrame({ 'Altitude':alt,'Density':rho,'Vstall(CL=1.4)':Vs,'Vmax':Vmax,'Tmax':Tmax})

).hide\_index()

display(DFB)

v1=np.zeros([len(alt),n])

Ta1=np.zeros([len(alt),n])

theta=np.zeros([len(alt),n])

Vc=np.zeros([len(alt),n])

**for** k **in** range(len(alt)):

v1[k]=np.linspace(Vs[k],Vmax[k],n)

**for** l **in** range(n):

*#thrust available*

M1=v1[k][l]/a[k]

TVV1=fff(rho[k]) *# thrust variation logic*

yyy=TVV1\*K1\*M1\*\*(K2)

Ta1[k][l]=yyy

A1=(K\*w\*\*2)/(0.5\*rho[k]\*v1[k][l]\*\*2\*s)

B1=-w

Caa=0.5\*rho[k]\*v[k][l]\*\*2\*s\*CD0

C1=Ta1[k][l]-Caa-A1

www=(B1\*\*2)-(4\*A1\*C1)

wwww=www\*\*(0.5)

*#W1=(-B1+wwww)/(2\*A1)*

W2=(-B1-wwww)/(2\*A1)

W22=np.arcsin(W2)

theta[k][l]=W22\*(180/math.pi)

Vc[k][l]=v[k][l]\*W2\*60

RCmax=np.zeros(len(alt))

V\_RCmax=np.zeros(len(alt))

theta\_max=np.zeros(len(alt))

V\_theta\_max=np.zeros(len(alt))

**for** Q **in** range(len(alt)):

*#RCmax and V\_rcmax finding*

vvvvv=np.max(Vc[Q])

index= np.argmax(Vc[Q], axis=0)

ccccc=v1[Q][index]

V\_RCmax[Q]=ccccc

RCmax[Q]=vvvvv

*#thetamax and V at theta max*

vvvvv1=np.max(theta[Q])

index1=np.argmax(theta[Q], axis=0)

ccccc1=v1[Q][index1]

V\_theta\_max[Q]=ccccc1

theta\_max[Q]=vvvvv1

dfc=pd.DataFrame({ 'Altitude':alt,'Density':rho,'(R/C)max':RCmax,'V(R/c)max':V\_RCmax,'thetamax':theta\_max,'V(theta\_max)':V\_theta\_max})

).hide\_index()

display(DFC)

**for** Q1 **in** range(len(alt)):

plt.figure(num=4,dpi=view)

plt.plot(v1[Q1],Vc[Q1],linewidth=2.5,label='Alt=%d' %(alt[Q1]))

plt.ylim([0,1400])

*#plt.legend(loc ="lower right")*

plt.text(V\_RCmax[Q1],RCmax[Q1],'h=%d km' %(alt[Q1]))

plt.scatter(V\_RCmax[Q1],RCmax[Q1])

plt.title("Rate of climb Vs velocity for various altitudes")

plt.ylabel('Rate of Climb (m/min)')

plt.xlabel('TAS(m/s)')

plt.legend()

plt.grid(color = 'green', linestyle = '--', linewidth = 0.2)

**for** Q2 **in** range(len(alt)):

plt.figure(num=4,dpi=view)

plt.plot(RCmax,alt,linewidth=2,color='red',marker='>')

plt.grid(color = 'green', linestyle = '--', linewidth = 0.5)

plt.title("Maximum rate of climb Vs altitude")

plt.ylabel('Altitude(KM)')

plt.xlabel('Maximum rate of climb(m/min)')

**for** Q3 **in** range(len(alt)):

plt.figure(num=4,dpi=view)

plt.plot(V\_RCmax,alt,linewidth=2,color='blue',marker='<')

plt.grid(color = 'green', linestyle = '--', linewidth = 0.5)

plt.title("Velocity at Max rate of climb Vs altitude")

plt.ylabel('Altitude(KM)')

plt.xlabel('Velocity at max rate of climb V (R/c)max ')

**for** Q4 **in** range(len(alt)):

plt.figure(num=4,dpi=view)

plt.plot(v1[Q4],theta[Q4],linewidth=2.5,label='Alt=%d' %(alt[Q4]))

plt.ylim([0,10])

*#plt.legend(loc ="lower right")*

plt.text(V\_theta\_max[Q4],theta\_max[Q4],'h=%d km' %(alt[Q4]))

plt.scatter(V\_theta\_max[Q4],theta\_max[Q4])

plt.title("Angle of climb Vs velocity for various altitudes")

plt.ylabel('Angle of Climb (in Degrees)')

plt.xlabel('TAS(m/s)')

plt.legend()

plt.grid(color = 'green', linestyle = '--', linewidth = 0.2)

**for** Q5 **in** range(len(alt)):

plt.figure(num=4,dpi=view)

plt.plot(theta\_max,alt,linewidth=2,color='red',marker='>')

plt.grid(color = 'green', linestyle = '--', linewidth = 0.5)

plt.title("Maximum angle of climb Vs altitude")

plt.ylabel('Altitude(KM)')

plt.xlabel('Maximum angle of climb(in degrees)')

**for** Q6 **in** range(len(alt)):

plt.figure(num=4,dpi=view)

plt.plot(V\_theta\_max,alt,linewidth=2,color='blue',marker='<')

plt.grid(color = 'green', linestyle = '--', linewidth = 0.5)

plt.title("Velocity at Max angle of climb Vs altitude")

plt.ylabel('Altitude(KM)')

plt.xlabel('Velocity at max rate of climb V (R/c)max ')

v11=np.zeros([len(alt),n])

Ta11=np.zeros([len(alt),n])

R1=np.zeros([len(alt),n])

w1=np.zeros([len(alt),n])

DT1=np.zeros([len(alt),n])

phi=np.zeros([len(alt),n])

CLL=np.zeros([len(alt),n])

S=np.zeros([len(alt),n])

CLT1=np.zeros([len(alt),n])

CDT1=np.zeros([len(alt),n])

CDT=np.zeros([len(alt),n])

CLT=np.zeros([len(alt),n])

n1=np.zeros([len(alt),n])

**def** ff(T):

**return** (1.4\*288\*T)\*\*(0.5)

**def** fff1(X):

**return** Tsv\*(X/1.22500002)

**def** fx1(V,rho):

**return** w/(0.5\*rho\*V\*\*2\*s)

**def** ffx1(cl):

**return** CD0+cl\*\*2\*0.05

**for** k **in** range(len(alt)):

v11[k]=np.linspace(Vs[k]+nn,Vmax[k]-nn,n)

**for** l **in** range(n):

*#thrust available*

M1=v11[k][l]/a[k]

TVV1=fff(rho[k]) *# thrust variation logic*

yyy=TVV1\*K1\*M1\*\*(K2)

Ta11[k][l]=yyy

**for** m1 **in** alt:

**for** m2 **in** range(n):

X=fx1(v11[m1][m2],rho[m1])

CLL[m1][m2]=X

Y=clmax/CLL[m1][m2]

S[m1][m2]=Y

**if** S[m1][m2]<3.5:

CLT1[m1][m2]=1.4

**else**:

CLT1[m1][m2]=3.5\*CLL[m1][m2]

XX345=ffx1(CLT1[m1][m2])

CDT1[m1][m2]=XX345

YY345=0.5\*rho[m1]\*v11[m1][m2]\*\*2\*s\*CDT1[m1][m2]

DT1[m1][m2]=YY345

**if** DT1[m1][m2]<Ta11[m1][m2]:

CDT[m1][m2]=CDT1[m1][m2]

CLT[m1][m2]=CLT1[m1][m2]

**else**:

zz345=(Ta11[m1][m2])/(0.5\*rho[m1]\*v11[m1][m2]\*\*2\*s)

CDT[m1][m2]=zz345

zzz345=(CDT[m1][m2]-CD0)/K

zzzz=zzz345\*\*(0.5)

CLT[m1][m2]=zzzz

e345=CLT[m1][m2]/CLL[m1][m2]

n1[m1][m2]=e345

ee345=1/(e345)

eee345=np.arccos(ee345)

eeee345=eee345\*(180/math.pi)

phi[m1][m2]=eeee345 *#phi value*

rrrr=v11[m1][m2]\*\*2

rrr=n1[m1][m2]\*\*2-1

rr=g\*rrr\*\*0.5

r345=rrrr/rr

R1[m1][m2]=r345

*#finding rate of turn*

ww=v11[m1][m2]/R1[m1][m2]

w1[m1][m2]=ww

Wmax=np.zeros(len(alt))

V\_Wmax=np.zeros(len(alt))

V\_Rmin=np.zeros(len(alt))

Rmin=np.zeros(len(alt))

**for** we **in** range(len(alt)):

c=np.amax(w1[we])

index=np.argmax(w1[we], axis=0)

ccccc=v1[we][index]

V\_Wmax[we]=ccccc

Wmax[we]=c

cc=np.amin(R1[we])

index1=np.argmin(R1[we], axis=0)

ccc=v1[we][index1]

V\_Rmin[we]=ccc

Rmin[we]=cc

dfe=pd.DataFrame({ 'Altitude':alt,'Density':rho,'Wmax':Wmax,'Vel at Wmax':V\_Wmax,'Rmin':Rmin,'Vel at Rmin':V\_Rmin})

,

]

).hide\_index()

display(DFE)

fig,ax1=plt.subplots(num=3,dpi=100)

plt.grid(color = 'green', linestyle = '--', linewidth = 0.2)

ax1.set\_xlabel('$FLIGTH VELOCITY$')

ax1.set\_ylabel('$Turn radius$')

ax1.plot(v11[0],R1[0],label='Radius of turn',linewidth=2,color='red')

ax2=ax1.twinx()

ax2.set\_ylabel('$Rate of Turn$')

ax2.plot(v11[0],w1[0],label='Rate of turn',linewidth=2,color='blue')

*#fig.tight layout()*

plt.figure()

ax1.legend(loc ="lower right")

ax2.legend(loc ="upper right")

plt.show()

**for** h **in** range(len(alt)):

plt.figure(num=2,dpi=view)

plt.plot(v11[h],R1[h],linewidth=2.5,label='alt=%d KM' %(h))

plt.xlim(120,270)

plt.ylim(100,12000)

plt.title("Radius of turn Vs Velocity at various altitudes")

plt.ylabel('Radius of turn(m)')

plt.xlabel('Velocity (m/s)')

plt.grid(color = 'green', linestyle = '--', linewidth = 0.25)

plt.legend(loc ="upper left")

**for** Q7 **in** range(len(alt)):

plt.figure(num=4,dpi=view)

plt.plot(V\_Rmin,alt,linewidth=2,color='red',marker='>')

plt.grid(color = 'green', linestyle = '--', linewidth = 0.5)

plt.title('velocity at Rmin vs Altitude')

plt.ylabel('Altitude(KM)')

plt.xlabel('velocity at Rmin')

plt.figure(num=4,dpi=view)

plt.plot(Rmin,alt,linewidth=2,label='As altitude increseing the radius to turn is also incresing',color='red',marker='>')

plt.grid(color = 'blue', linestyle = '--', linewidth = 0.5)

plt.title("min radius of turn VS altitude")

plt.ylabel('Altitude(KM)')

plt.xlabel('Rmin(m)')

plt.legend()

**for** h **in** range(len(alt)):

plt.figure(num=1,dpi=view)

plt.plot(v11[h],w1[h],linewidth=2.5,label='alt=%d KM' %(h))

plt.grid(color = 'green', linestyle = '--', linewidth = 0.3)

*#plt.ylim(0.04,0.14)*

*#plt.xlim(100,180)*

plt.title("Rate of turn Vs Velocity at various altitudes")

plt.ylabel('Rate of turn(rad/s)')

plt.xlabel('Velocity (m/s)')

plt.legend(loc ="upper right")

*#plt.plot(v,R[6])*

**for** Q5 **in** range(len(alt)):

plt.figure(num=4,dpi=view)

plt.plot(V\_Wmax,alt,linewidth=2,label='V\_Wax inc as Alt inc',color='red',marker='>')

plt.grid(color = 'green', linestyle = '--', linewidth = 0.5)

plt.title("velocity at rate of turn max VS altitude")

plt.ylabel('Altitude(KM)')

plt.xlabel('velocity at rate of turn max')

plt.figure(num=4,dpi=view)

plt.plot(Wmax,alt,linewidth=2,label='Wmax is decerse as alt increseing',color='red',marker='>')

plt.grid(color = 'blue', linestyle = '--', linewidth = 0.5)

plt.title("Maximum rate of turn VS altitude")

plt.ylabel('Altitude(KM)')

plt.xlabel('Wmax(rad/s)')

plt.legend()

**for** iii **in** alt:

print("**\t\t\t\t**A typical turning flight performance at %s altitude is tabulated below" % (alt[iii]))

ddddf=pd.DataFrame({'V':v11[iii],'CLL':CLL[iii],' Clmax/CLL ':S[iii],'CLT1':CLT1[iii],'CDT1':CDT1[iii],'DT1':DT1[iii],'Ta':Ta[iii],'CDT':CDT[iii],'CLT':CLT[iii],'CLT/CLL':n1[iii],'Phi':phi[iii],'R':R1[iii],'W':w1[iii]})

dddf=ddddf[1:n:400]

## Range and Endurance code

**import** **numpy** **as** **np**

**import** **pandas** **as** **pd**

**import** **matplotlib.pyplot** **as** **plt**

**import** **math**

**from** **ambiance** **import** Atmosphere

*# weigths for boeing 737*

W0=580506.8 *# gross weigth in newtons*

empty\_weigth= 291415.9

fuel\_weigth=119005.1

payload=170085.6

maximum\_landing\_weigth=493403.8

W1=W0-fuel\_weigth

s=111.63

K=0.04244

ct=0.6 *#thrust specific fuel consumption in hr^-1*

CD0=0.0159

view=100

**def** f(CL):

**return** CD0+(K\*CL\*\*2)

**def** ff(rho):

**return** ((((3\*K)/(CD0))\*\*(0.5))\*(W0/s)\*(2/rho))\*\*(0.5) *#for range*

**def** fff(rho):

**return** ((((K)/(CD0))\*\*(0.5))\*(W0/s)\*(2/rho))\*\*(0.5) *# for endurance*

*# for range*

*#cd0=3kcl^2*

*# cl^(1/2)/cd maximum range condition*

CL1=(CD0/(3\*K))\*\*(0.5)

E1=(CL1\*\*(0.5))/(f(CL1)) *#where E1 is the cl^(1/2)/cd ratio*

*# for endurance*

*#CD0=KCL^2*

*# CL/CD max condition for maximum endurance*

CL2=(CD0/(K))\*\*(0.5)

E2=(CL2)/(f(CL2)) *# where E2 CL/CD max condition for maximum endurance*

alt=[0,1,2,3,4,5,6,7,8,9,10]

rho=np.zeros(len(alt))

R=np.zeros(len(alt))

VR=np.zeros(len(alt)) *# at this velocity range is maximum*

EN=np.zeros(len(alt))

VE=np.zeros(len(alt)) *# at this velocity range is endurance*

**for** i **in** range(len(alt)):

mosphere = Atmosphere(alt[i]\*1000)

*#density at various altitudes calculations*

rho[i] = mosphere.density

*#calculation of range*

a1=(W0\*\*(0.5)-W1\*\*(0.5))\*E1\*((2\*3600)/ct)\*((2/(rho[i]\*s))\*\*(0.5))

R[i]=a1/1000 *#for KM convertion*

VR[i]=ff(rho[i])

*#calculation of endurance*

a2=(1/ct)\*(E2)\*(math.log(W0/W1))

EN[i]=a2

VE[i]=fff(rho[i])

plt.figure(num=2,dpi=view)

plt.plot(R,alt,linewidth=2,label='range is inc as altitude inc',color='blue',marker='<')

plt.title("Variation of range with altitude")

plt.xlabel('Range(KM)')

plt.ylabel('Altitude(KM)')

plt.legend()

plt.grid(color = 'green', linestyle = '--', linewidth = 0.45)

plt.figure(num=2,dpi=view)

plt.plot(R,alt,linewidth=2,label='Vrmax is inc as altitude inc',color='red',marker='<')

plt.title("Variation of Vat Rmax with altitude")

plt.xlabel('V at Rmax')

plt.ylabel('Altitude(KM)')

plt.legend()

plt.grid(color = 'green', linestyle = '--', linewidth = 0.3)

plt.figure(num=2,dpi=view)

plt.plot(alt,EN,linewidth=2,label='Endurance is constant as altitude increases',color='blue',marker='>')

plt.title("Variation of Endurance with altitude")

plt.xlabel('Endurance(KM)')

plt.ylabel('Altitude(KM)')

plt.legend()

plt.grid(color = 'green', linestyle = '--', linewidth = 0.45)

plt.figure(num=2,dpi=view)

plt.plot(VE,alt,linewidth=2,label='Endurance is constant as altitude increases',color='red',marker='>')

plt.title("Variation of $VEmax$ with altitude")

plt.xlabel(' V at Max Endurance(KM)')

plt.ylabel('Altitude(KM)')

plt.legend()

plt.grid(color = 'green', linestyle = '--', linewidth = 0.45)

## Take-off distance code

**import** **math**

w=580506.8

s=111.63 *# wing span area M^2*

clmax=2.7 *#flaps on*

CD0=0.0159

K=0.04244

K1=0.41065

K2=-0.3078*# see negative value is there*

g=9.81

rho=1.225 *#sealevel takeoff condition kg/m^3*

a =340 *#speed of sound at sea level*

TSV= (2\*97900) *#static thrust at sea level in newtons*

mu1=0.04 *# friction of road for takeoff*

Kuc1=4.5e-5 *# for moderate flap setting use in takeoff*

b=32.22 *# wing span in meters*

h=11.17 *# wing heigth above ground in meters*

AR=9.3 *# aspect ratio*

e=0.8064

CL=0.1 *# assumed value for takeoff*

wbys=w/s *# weigth by span ratio*

*# vstall caluation*

vstall=((2\*wbys)/(rho\*clmax))\*\*(0.5)

*# VLO caluculation*

VLO=1.1\*vstall

*# free stream velocity*

V=0.7\*VLO

*#mach number*

M=V/a

*#thrust at 0.7VLO*

T=TSV\*K1\*(M\*\*K2)

*# T /w ratio*

Tbyw=T/w

*# constants 1 in takeoff formula*

KT=Tbyw-mu1

*#DELTHA cd0*

m=w/g

delthaCD0=wbys\*Kuc1\*(m\*\*(-0.215))

*# GRound effect calculuation*

a1=(16\*(h/b))\*\*2

G=a1/(1+a1)

*# constants 2 in takeoff formula*

KA=((K+(G/(3.14\*e\*AR)))\*CL\*\*2-(mu1\*CL)+CD0+delthaCD0)\*(-rho/(2\*wbys))

*#ground roll*

a2=math.log(1+(KA/KT)\*VLO\*\*2)

a3=1/(2\*g\*KA)

sg1=a2\*a3

*#sg1 take N=3 seconds*

N=3

sg2=N\*VLO

sg=sg1+sg2

*# calculation of distance coverd by air borne*

R=(6.96\*vstall\*\*2)/g

*# 35 feet to meters 10.668*

hob=10.668

a4=(1-(hob/R))

theta1=math.acos(a4)

theta=theta1\*180/math.pi

*#distance of airbrone*

sa=R\*math.sin(theta1)

*#total takeoff distance*

Total\_Takeoff\_distance=sg+sa

## Landing distance code

**import** **math**

w=580506.8

s=111.63 *# wing span area M^2*

clmax=2.7 *#flaps on*

CD0=0.0159

K=0.04244

K1=0.41065

K2=-0.3078*# see negative value is there*

g=9.81

rho=1.225 *#sealevel landing condition kg/m^3*

a =340 *#speed of sound at sea level*

TSV= (2\*97900) *#static thrust at sea level in newtons*

mu1=0.04 *# friction of road for takeoff*

mu2=0.4 *# for landing*

Kuc1=4.5e-5 *# for moderate flap setting use in takeoff*

KUc2=3.16e-5 *# for landing flap setting*

b=32.22 *# wing span in meters*

h=11.17 *# wing heigth above ground in meters*

AR=9.3 *# aspect ratio*

e=0.8064

CL=0.1 *# assumed value for takeoff*

Theta\_a1=3 *#in degress # approach angle*

theta\_a=((3\*math.pi)/180)

wbys=w/s *# weigth by span ratio*

*# vstall caluation*

vstall=((2\*wbys)/(rho\*clmax))\*\*(0.5)

*# flare velcity*

VF=1.23\*vstall

*#VTD touch down velocity*

VTD=1.15\*vstall

*#R*

R=(VF\*\*2)/(0.2\*g)

hf=R\*(1-math.cos(theta\_a))

*#50 feet is 15 meters*

*#airbrone distance*

sa=((15-hf)/math.tan(theta\_a))

*#flare distance*

sf=R\*math.sin(theta\_a)

*#DELTHA cd0*

m=w/g

delthaCD0=wbys\*KUc2\*(m\*\*(-0.215))

*# GRound effect calculuation*

a1=(16\*(h/b))\*\*2

G=a1/(1+a1)

*# constants 2 in landing formula*

JA=((K+(G/(3.14\*e\*AR)))\*CL\*\*2-(mu2\*CL)+CD0+delthaCD0)\*(rho/(2\*wbys))

*# constants 1 in landing formula*

JT=mu2

*#ground roll*

a2=math.log(1+(JA/JT)\*VTD\*\*2)

a3=1/(2\*g\*JA)

sg1=a2\*a3

*#sg1 take N=3 seconds*

N=3

sg2=N\*VTD

sg=sg1+sg2

Total\_landing\_distance=sa+sf+sg

# Conclusions and remarks

1. Performance of a typical commercial airliner has been estimated for stalling speed,  
   maximum speed, minimum speed, steady climb, range, endurance, turning, take-off  
   and landing.
2. The performance approximately corresponds to that of B737-200.
3. Variations with altitude of the characteristic velocities  
   corresponding to :  
   stalling speed, Vs  
   maximum speed,   
   minimum speed as dictated by thrust,   
   maximum rate of climb,   
   maximum angle of climb,   
   maximum rate of turn,   
   minimum radius of turn,

Have been tabulated and plotted using Jupiter notebook.

# References

* Aircraft fligth mechanics by Harry Smith <https://www.aircraftflightmechanics.com/index.html>
* NPTEL Fligth dynamics 1, Aerospace Engineering, Prof. E.G. Tulapurkara, IITM

<https://nptel.ac.in/courses/101/106/101106041/>

* Aircraft performance and design, John D. Anderson, Jr, McGraw-Hill.