宇宙ロケットの設計

1116170035

　宇宙ロケットの設計をする.1段ロケットで,機体に積み込む燃料の最適な量だけを計算で求めた.

1. ロケットの条件
2. 燃料なしの機体重量は人工衛星を積んだ状態でK = 500[kg]とする.
3. 宇宙探査機のロケットエンジンは,燃料を1[kg]燃焼させると300[kgf]の推力を1[s]間出すことができる.そのようなエンジンを4台搭載している.
4. ロケットは地球に対して垂直に打ち上げられ,そのままの方向で高度160[km]において秒速7.9[km/s]となり,そこで人工衛星を軌道に投入する.実際は地球と平行になるようにする.この状態で燃料がなくなるようになれば良い.
5. シミュレーションの条件
6. ロケットはエンジンの噴射で速度を速めながら高度H=160[km]に達する.また,エンジンの最大出力は1台につき10kgとする.
7. 地球の重力は高度によって変化するのだが,今回は簡易のため一定として計算し,G=9.81[m/]とする.
8. 積み込みの燃料の値を入力し,燃料の無くなった時間での高度が160[km]を超えるための燃料消費量を割り出す.

(1)~(3)をもとに計算し結果を縦軸に高度と速度,横軸に時間をプロットしたグラフを作成した. 図1はスペースX社が開発した低コストであるにもかかわらず,離陸したそのままの体制で着陸もできるロケットの離陸する瞬間である.



図１　再利用ロケットスペースXの離陸する瞬間

<https://www.hazardlab.jp/contents/post_info/2/7/5/27522/Dtry3yzXgAA70D7.jpg>

1. 考え方

プログラムはPython3.7.3を使った.各秒で燃料消費量X[kg]が入力されると,推進力Fが計算され,そのFを用いて運動方程式よりaを算出.その後,その加速度から速度と高度を順に計算し,それをwhile文で高度が160[km]を超えるまで計算した.計算した値を逐次,リストにまとめ最後にグラフ化した.表1に主な計算部分を載せる.

表1　計算部分(著者作成)

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22 | while height < 160000 :  syohi = int(input('消費量はいくらにしますか(max:10 kg/s)')  M2 = M1 - syohi\*4  M = 0.5\*(M1+M2)  F = 300 \* 9.81 \*4\*syohi - M\*g  a = F/M  M1 = M2  print('M:'+str(M))  v2 = v1+a  print('v2:'+str(v2))  height += 0.5\*(v1+v2)  v1 = v2  E -= 0.5\*syohi\*4  print('E:'+str(E))  if E <= 0 :  print('燃料なくなりました')  break  t += 1  V.append(v2)  H.append(height)  time.append(t) |

1. 結果

図2は赤線を速度,青線を高度とし,横軸時間としたときのグラフである.手探りで値を代入し続けたところ初期燃料4000[kg],毎秒10[kg]の燃料消費で160[km]を超えるロケットを探すことができた.(図2左) 100sの時に約164.4[km],5.5[km/s]という結果が出た.

初期燃料5000[kg],毎秒10[kg]の燃料消費で160[km]を超えるロケットを探すことができた.(図2左) 100sの時に約162.3[km],4.5[km/s]という結果が出た.

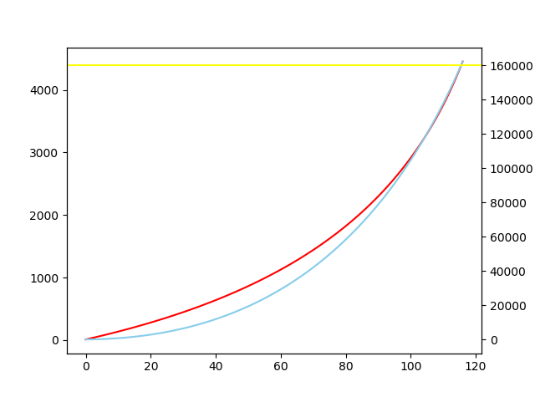
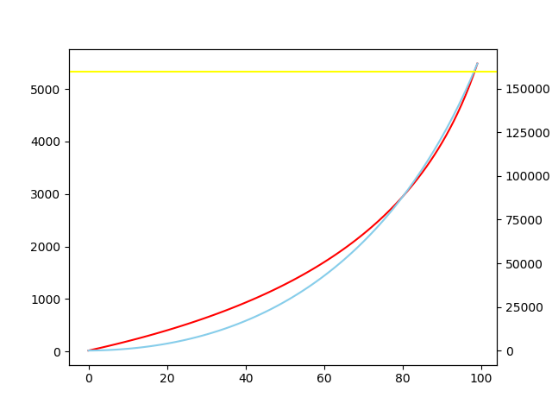


図2　高度と速度のグラフ(著者作成)

1. 考察

ロケットのシミュレートする際には,緻密な計算が絡んでくると推察する.今回は,空気抵抗や重力が高度によって変化することを考慮せず,また機体を垂直のまま飛ばすことを考えたが実際にはもっと複雑に変数を用意し,その時々によって変わる気候も考慮しなくてはならないと考えると大変な労力がかかると考えた.

スペースの例では,垂直に降下してくるロケットを作成している.シミュレーション上で降下して地上0[m]地点で速度が0[km/s]になるようなシミュレータをプログラムしても面白いと考えた.

1. グラフ作成方法

今回,グラフ作成に用いたのはmatplotlibというライブラリである.このライブラリはPythonにおけるグラフ描画のためのライブラリである.ヒストグラムや散布図,円グラフ,箱ひげ図など様々なグラフを描画することができ,用途に合わせて調整できる.また,引数を調整することで,例えば,直線の色を変えたり,一度に任意の枚数の図を描画することもできる. 次ページ図3は一枚の画像を9分割し,それぞれの分割された画像に

どのような割合でRGBが含まれているのかを調べたものである.

これを使えば,同じような割合のRGB値を持つ画像を検出することができ類似画像として検索する手段にも使える.

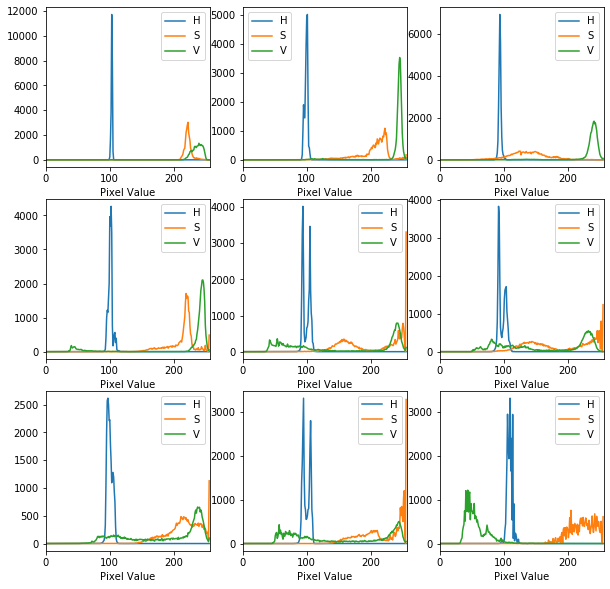


図3 RGBの割合

1. 参考

すぐわかるmatplotlibライブラリの使いかた<https://www.sejuku.net/blog/54285>

ドキュメントmatplotlib <https://matplotlib.org/>