Orbitális pályára állás

Gillich Ádám

2026. január 14.

Bevezetés

Munkám célja az alacsony Föld körüli és Kerbin körüli orbitális pályára (LEO és LKO) álláshoz szükséges Δv kiszámítása. A projekt lényegesen változott az eredeti célhoz képest. A programozott röppálya el lett vetve, mert túl új és komplikált, valamint amint kiderült Teofilatto és tsai. munkája kiváló, de nem pontosan erre a célra tervezték. Így a kísérleteknél kézi vagy modok segítségével fél automata irányítást alkalmaztam, az alapján, hogy mi adta a jobb eredményt.

A dokumentum felépítése:

1. Elmagyarázom az alapfogalmakat, amik kellenek a megértéshez.
2. Egyszerűen elmagyarázom, hogy hogyan állítanak pályára egy rakétát, amit összevetek azzal, amit a számításaimban használtam.
3. Elmagyarázom a Python kódom lényegi részét.
4. Bemutatom a kísérlethez használt rakétákat, és a kísérletet. Bemutatom az eredményeket.
5. Ezek után kiértékelem, összevetem az eredményeket.

A projektben használt és keletkezett anyag minden formáját tőlem telhető legrészletesebben igyekszem feltölteni a [14] forrás alatt található GitHub repository-ba.[1].

1. Fogalomtár

- **Orbitális pálya (pálya)** — olyan általánosan ellipszis alakú pálya ami egy égitest körül helyezkedik el, nem érintve annak felszínét. Kialakításához elsősorban sebességre van szükség, magasság csak a légkör miatt követelmény. Lehet rá folyamatos szabadesésként is gondolni, ami azért lesz folyamatos, mert olyan sebességgel haladunk a bolygó felszíne felett, hogy kifordul alattunk a bolygó. Így sosem esünk vissza.
- **Δv** — Egyszerűen úgy lehet rá tekinteni, mint a rakéta üzemanyaga. Hasonló módon, mintha egy autóban ülve nem a megmaradt liternyi üzemanyagot néznéd meg, hanem azt, hogy hány km utat tud még megtenni. És más autóba átülve nem kell gondolkodni, hogy az autó tulajdonságait tekintve mennyire elég 5 $liter$ benzin, ha tudod, hogy még 50 km-t meg tudsz tenni vele. A Tziolkowski egyenletből következik, hogy a rakéta az űrben a sebességét tudja megváltoztatni, és arra a következtetésre lehet jutni, hogy érdemes egy rakéta "üzemanyagát" m/s-ban mérni, és a Δv pedig az a sebességváltozás, amit a rakéta ideális esetben vákuumban el tud érni. Általában nem is kell foglalkozni azzal, mi van, ha nem ideális esetben vagyunk, de itt pont ez lesz a helyzet.

[1] A projekt "kódneve" *'Orbit Project'* volt, így ezen a néven is előfordulhat

- **Gravitációs Forduló** — A pályára állás az a része amikor hagyjuk, hogy a gravitáció fordítsa a rakétát. Azért fordítja, mert ha bedöntjük a rakétát, akkor a gravitáció a sebességvektorral nem lesz párhuzamos, így azon kívül, hogy annak ellentart, még forgatja is azt.
- **Hohmann módszer** — Egy nagyon hatékony módszer a Δv kiszámítására, amikor egyik körpályáról egy másikra megyünk. De ennek két fele van, ugyanis az űrben nem tudjuk egy ponton megemelni a pályánkat, így megemeljük egy ponton, majd fölmegyünk az *Apoapsis*-ra (lásd lejjebb) és ott is gyorsítunk egyet, így lesz újra körpályánk. Ebből nekünk az utóbbi eset kell majd.
- **Apoapsis** — a pálya legmagasabb pontja (általánosan)
- **Periapsis** — a pálya legalacsonyabb pontja (általánosan)
- **Prograde** — haladási irány
- **liftoff** — a rakéta elengedésének pillanata
- **TWR** — Tolóerő–Tömeg arány, a *Thrust-to-Weight Ratio* rövidítése
- $\mathbf{I_{sp}}$ — A specifikus impulzus, tolóerő osztva a súllyal (I/mg) [3], ez egy hajtóműre jellemző mennyiség, minden hajtóműnek van sajátja. A legegyszerűbben úgy lehet rá gondolni mint a hajtómű teljesítménye, hatékonysága. A nagyobb I_{sp} jobb.
- **R** — a bolygó sugara
- μ — GM, ahol G a gravitációs állandó, M pedig a bolygó tömege, neve *standard gravitációs paraméter*, minden égitestnek van sajátja. Ezt a jelölést bevezetve kicsit rövidebbé, és egyszerűbben érthetővé válnak a formulák.
- **Analitikus megoldás** — Egy olyan megoldás, ami pontosan adja vissza az értéket
- **Numerikus megoldás** — Ha a problémának nincs analitikus megoldása, vagy nehéz megtalálni az analitikus megoldást, a közelítő számítások mellett (például: Taylor-sorfejtés) numerikus módszerrel is célt lehet érni. A numerikus módszerek nem adnak pontos eredményt, de ismerve a mérések pontatlanságát és a rendszert befolyásoló faktorokat, az eredmény kellően pontos lehet. A legtöbb módszernél meg lehet adni egy δ vagy ε paramétert, hogy mennyire közelítse meg a valós megoldást.
- **KSP, és RSS** — ez talán a legfontosabb. A kísérleteket a Kerbal Space Program alap és modolt változatában végeztem el, amikor azt írom KSP akkor az alap játékra gondolok, és amikor azt írom RSS akkor a modoltra. RSS onnan jön hogy az egyik fő mod a *Real Solar System*.

2. Elméleti háttér

2.1. Hogyan működik egy rakéta fellövése?

Ha körpályára akarunk állni a Föld körül, akkor 3+1 lépésből áll.

1. Függőleges emelkedés: A rakéta *liftoff* után nem csinál semmit csak emelkedik egy bizonyos pontig.[2]

2. Gravitációs forduló: A rakéta egy bizonyos, jól optimalizált szöggel bedől, és azután követi a *Prograde* irányát. Hagyja, hogy a gravitáció segítsen a fordulásban, amíg a légkör hatása elhanyagolhatóvá nem válik.

3. Irányított szakasz: Ha a légkör már elhanyagolható, a rakétát átadják egy algoritmusnak, ami valós időben korrigálja a pályát, és nagy valószínűleg hamar áttér közel vízszintes repüléshez, azért hogy sebességet gyűjtsön az orbitális pálya kialakításához.
A [9] forrásban Reddy az űrsikló dokumentációi alapján készített egy ilyen algoritmust pályára álláshoz.[3]

4. Siklás[4] és körpálya: Ha megtette az algoritmus amit tudott, akkor vár, ez a siklási szakasz. Addig vár amíg el nem éri a rakéta az *Apoapsis*-t, ott újra begyújtja a hajtóművét és kialakítja a körpályát.

2.2. A számítások fizikai háttere

Változások

Eredetileg a [1] forrás 57. fejezetében található módszert terveztem használni a Δv_{total} kiszámítására, ami még mindig egy lehetőség, de én végül eltértem tőle. Majd később részletezem a pontos módszert.
Az [1] forrásban szereplő módszer:

$$\Delta v_{total} = \Delta v_K + \Delta v_U + \Delta v_g + \Delta v_d + \Delta v_{steer} - \Delta v_{lat} \tag{1}$$

Az általam írt Python kód és a hozzá tartozó fizika erősen összekapcsolódik, de megpróbálom a lehető legjobban szétválasztani őket.

Teofilatto és tsai. munkájának célja

Teofilatto és tsai. munkájának a célja egy analitikus megoldás volt a *Gravitációs Fordulóra*, ehhez néhány egyszerűsítést kénytelenek voltak bevezetni. Például, hogy a légellenállás elhanyagolódik, de később látjuk majd, hogy nem is olyan nagy mértékű. Valamint, hogy a tolóerő-tömeg arány (*TWR*) egy átlagos értékkel leírható. Ami azért nem pontos, mert a hajtómű tolóereje nem konstans, és a tömeg sem. A [4] forrás alapján a tolóerő, ahogy a nyomás csökken, úgy növekszik, míg a tömeg folyamatosan fogy. Így a *TWR* folyamatosan nő. Ezt bizonyos körülmények közt akár egy konstans értéken lehetne tartani amennyiben a hajtómű tolóereje manuálisan is állítható.

A rakéta összes Δv-jének kiszámítása

A két fokozat összesen kifejthető Δv-je kiszámítható az [1] és [2] forrásban is található Tziolkowski formula egy változatával. A két fokozatnak külön kell kiszámítani a Δv-jét, majd összeadni őket ahhoz, hogy megtudjuk, mennyi Δv kifejtésére képes a rakéta.

$$\Delta V_{\text{Tz}} = g_0 \, I_{\text{sp}\,k} \, \ln\left(\frac{M_k}{M_{kf}}\right) \tag{2}$$

[2] Meghatározható sebességben, időben, magasságban, és talán másban is de ezek a gyakori kézenfekvő opciók.
[3] Amit nekem sajnos nem sikerült működésbe hozzak.
[4] *Coasting Flight*

Ahol a g_0 a gravitációs gyorsulás föld közelben, $I_{\mathrm{sp}\,k}$ a hajtómű specifikus impulzusa, M_k a fokozat teljes tömege[5], M_{kf} pedig vég tömeg.[5]

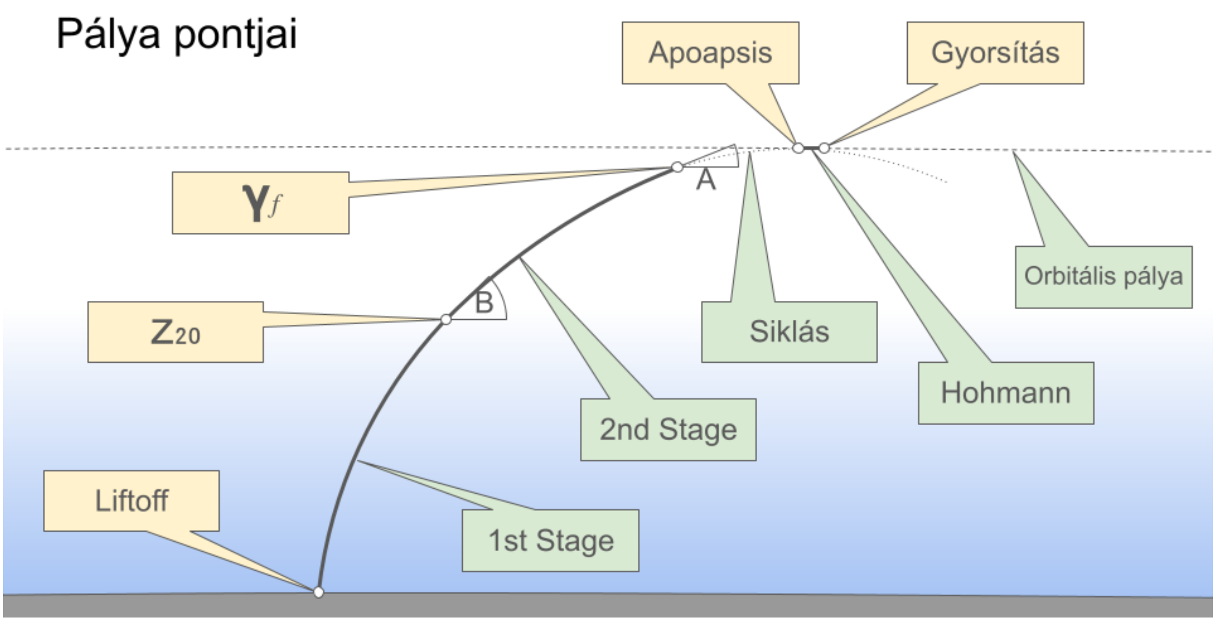
A mérés logikája

Maga a cél a Föld körül 250 km, és 120 km magasságú körpályát kialakítani a lehető legkevesebb elhasznált Δv-vel. Ezt elnevezem Δv_{total}-nak.

A rakéta képzeletbeli útja:

- Emelkedünk
- Gravitációs forduló, Teofilatto és tsai. analitikus módszere alapján
- Az *Apoapsis* eléri a 250 km-t
- A kialakult *Apoapsis* és *Periapsis* segítségével meghatározzuk, hogy Hohmann módszerrel mennyi Δv körpályává tenni a jelenlegi ellipszisünket.
- Összeadjuk a fokozatok által elhasznált Δv-t, meghatározva a Δv_{total}-t

A feltevés az, hogy lesz egy ideális pálya γ_f vég-röppályaszöggel, $v(z_{2f})$ sebességgel, és $R_{Föld} + h(z_{2f})$ sugárral aminek az *Apoapsis*-a éppen 250 km lesz. És az ezt követő Hohmann módszer után a Δv_{total} a minimum lesz.



1. ábra. A röppálya elméleti ábrája

Teofilatto és tsai. módszere erre nem ad választ, viszont ki tudom számolni, hogy a hajtóművek égetése után végül mekkora sebességgel, $v(z_{2f})$ megyek milyen magasságban,

[5] M_k: üzemanyag, struktúra, hasznos teher; M_{kf}: struktúra, hasznos teher (száraztehernek is nevezik)

$h(z_{2f})$, a végső szög z_{2f} függvényében, ami a γ_f, a vég-röppályaszög, egy ügyesen átalakított formája.[6]

A sebesség a mozgás végén:

$$v(z_{2f}) = A_2\, z_{2f}^{\bar{n}-1}\,(1 + z_{2f}^2) \tag{3}$$

A magasság a mozgás végén:

$$h(z_{2f}) = h(z_{20}) + \frac{A_2^2}{g}\left[\frac{z^{2\bar{n}-2}}{2\bar{n}-2} - \frac{z^{2\bar{n}+2}}{2\bar{n}+2}\right]_{z_{20}}^{z_{2f}} \tag{4}$$

Ahol $\bar{n}$ az átlagos *TWR*, és A_2 nincs megnevezve de abból ahogy használva van úgy lehetne elnevezni, mint egy röppálya skálázó paraméter, ami mindig egy a fokozatra jellemző.

Teofilatto és tsai. azzal számoltak, hogy γ_f a mozgás végén ismert. De mivel a mi helyzetünkben nem ismert, ezért a problémát numerikusan kell megoldani.

A módszerükkel kiszámítható a Δv_g, tehát az a Δv amit a gravitáció miatt elvesztünk.

$$\Delta v_g = \Delta v_{Tz1} + \Delta v_{Tz2} - v(z_{2f}) \tag{5}$$

Tehát a két fokozat által összesen elhasznált Δv mínusz a ténylegesen elért sebesség.

De még kellenek a pálya adatok.
A két fontos adat a Hohmann módszer szempontjából az *Apoapsis* és a *Periapsis*
Ha megvan az excentricitás akkor a [5] forrás alapján kettőt úgy számoljuk ki, hogy:

$$r_{Ap} = a(1+e); \qquad r_{Pe} = a(1-e) \tag{6}$$

Ahol a a félnagytengely és e az excentricitás.

Szintén a [5] forrásból a félnagytengely a következő módon felírható:

$$a = -\frac{\mu}{2E} \tag{7}$$

Ahol E a mechanikai energia.

Excentricitás vektor

A [8] forrásból megállapítható, hogy az excentricitás vektor a következő:

$$\mathbf{e} = \frac{\mathbf{v}\times\mathbf{L}}{\mu} - \frac{\mathbf{r}}{r}, \qquad \mathbf{L} = \mathbf{r}\times\mathbf{v} \tag{8}$$

Négyzetezzük:

$$\begin{aligned} e^2 &= \left(\frac{\mathbf{v}\times\mathbf{L}}{\mu} - \frac{\mathbf{r}}{r}\right)^2, \qquad (a-b)^2 \\ &= \frac{(\mathbf{v}\times\mathbf{L})^2}{\mu^2} - 2\cdot\frac{1}{\mu}\frac{\mathbf{r}}{r}\cdot(\mathbf{v}\times\mathbf{L}) + 1, \end{aligned}$$

$$\mathbf{v}\perp\mathbf{L}, \qquad \mathbf{r}\cdot(\mathbf{v}\times\mathbf{L}) = \mathbf{r}\cdot(\mathbf{v}\times(\mathbf{r}\times\mathbf{v})) = (\mathbf{r}\times\mathbf{v})\cdot(\mathbf{r}\times\mathbf{v}) = (\mathbf{r}\times\mathbf{v})^2 = L^2$$

[6] Az indíték ilyen módú jelölésére és az átalakítás módszerére a [2] forrás 7. oldalán található válasz.

$$e^2 = \frac{v^2L^2}{\mu^2} - \frac{2L^2}{\mu r} + 1 = 1 + \frac{L^2}{\mu^2}\left(v^2 - \frac{2\mu}{r}\right) = 1 + \frac{L^2}{\mu^2}2E$$

Bevezetve az $E = \frac{v^2}{2} - \frac{\mu}{r}$-t az excentricitás vektor nagysága a következő:

$$e = \sqrt{1 + \frac{2E\,L^2}{\mu^2}} \tag{9}$$

Ahol E a mechanikai energia:

$$E = \frac{v^2}{2} - \frac{\mu}{r}$$

És L pedig a perdület:

$$L = \mathbf{r} \times \mathbf{v} = r\,v\cos\gamma$$

Ez a mi esetünkre átalakítva:

$$E = \frac{v(z_{2f})^2}{2} - \frac{\mu}{R + h(z_{2f})}, \qquad L = \left[R + h(z_{2f})\right]\,v(z_{2f})\,\cos\gamma_f \tag{10}$$

Miután megvannak a pályaadataink még mindig ellipszis pályán keringünk, amit körpályává kell alakítani. Ehhez a Hohmann módszert alkalmazzuk, a [7] forrás alapján:

$$\Delta v_H = \sqrt{\frac{\mu}{r_{Pe}}}\left(\sqrt{\frac{2r_{Ap}}{r_{Ap} + r_{Pe}}} - 1\right) \tag{11}$$

Nagyon jó mostmár meg tudjuk állapítani a pályaadatainkat a mozgás végi adatokból, és körpályára tudunk állni.
Tehát mégegyszer: A feltevés az, hogy lesz egy ideális pálya γ_f vég-röppályaszög, $v(z_{2f})$ sebességgel, és $R + h(z_{2f})$ sugárral aminek az *Apoapsis*-a éppen 250 km lesz. És az ezt követő Hohmann módszer után a Δv_{total} a minimum lesz.

Ehhez numerikusan optimalizálni kell öt paramétert:
Δv_{total}, r_{Ap}, γ_f, z_{20}, M%-et.
Δv_{total}, r_{Ap}, γ_f-val már találkoztunk, z_{20} a röppályaszög az első fokozat leválasztásánál M% pedig abból következik, hogy nem minden üzemanyagunkat akarjuk elégetni csak azt amennyi kell. A név abból származik hogy ez tényleg egy százalék tehát például 56.66%, ebben az esetben a második fokozat 56.66%-át használtuk fel.

2.3. Numerikus számítási folyamat

Az számítási folyamatot szinte teljesen én készítettem, apukám, Gillich Péter segédkezett a numerikus optimalizációban[7].

A mérés módja az (1) egyenletben leítrakból úgy változott, hogy most azt mérem mennyi Δv lett elhasználva, nem pedig tételesen kiszámolom mennyi lenne a Δv_{total}.

Kis kiegészítés

Azzal, hogy tudjuk állítani a $M\%$-et, két dolog fog változni **1)** M_{2f}, **2)** t_{b2}, tehát a második fokozat vég tömege és az idő ameddig égetjük az üzemanyagot.
M_{2f} kilogikázható:

$$M_{ü} = M_{20} - M_{2\,\mathrm{dry}}$$

$$M_{ü\,maradt} = M_{ü}[1 - (M\%/100)]$$

$$M_{2f} = M_{2\,\mathrm{dry}} + M_{ü\,maradt} \tag{12}$$

Ahol $M_{ü}$ az üzemanyag tömege, $M_{2\,\mathrm{dry}}$ a száraz tömeg, M_{2f} pedig a vég tömeg.

A másik a t_{b2} azaz a második fokozat kiégésének ideje. A [3] és [4] forrásból meg lehet határozni $\dot{m}$-t, ebből következik:

$$t_{b2} = \frac{M_{ü\,használt}}{\dot{m}} = \frac{M_{20} - M_{2f}}{F_{\mathrm{Tolóerő}}/(g_0\, I_{sp})} = \frac{(M_{20} - M_{2f})\, g_0\, I_{sp}}{F_{\mathrm{Tolóerő}}} \tag{13}$$

Numerikus módszerek összefoglalója

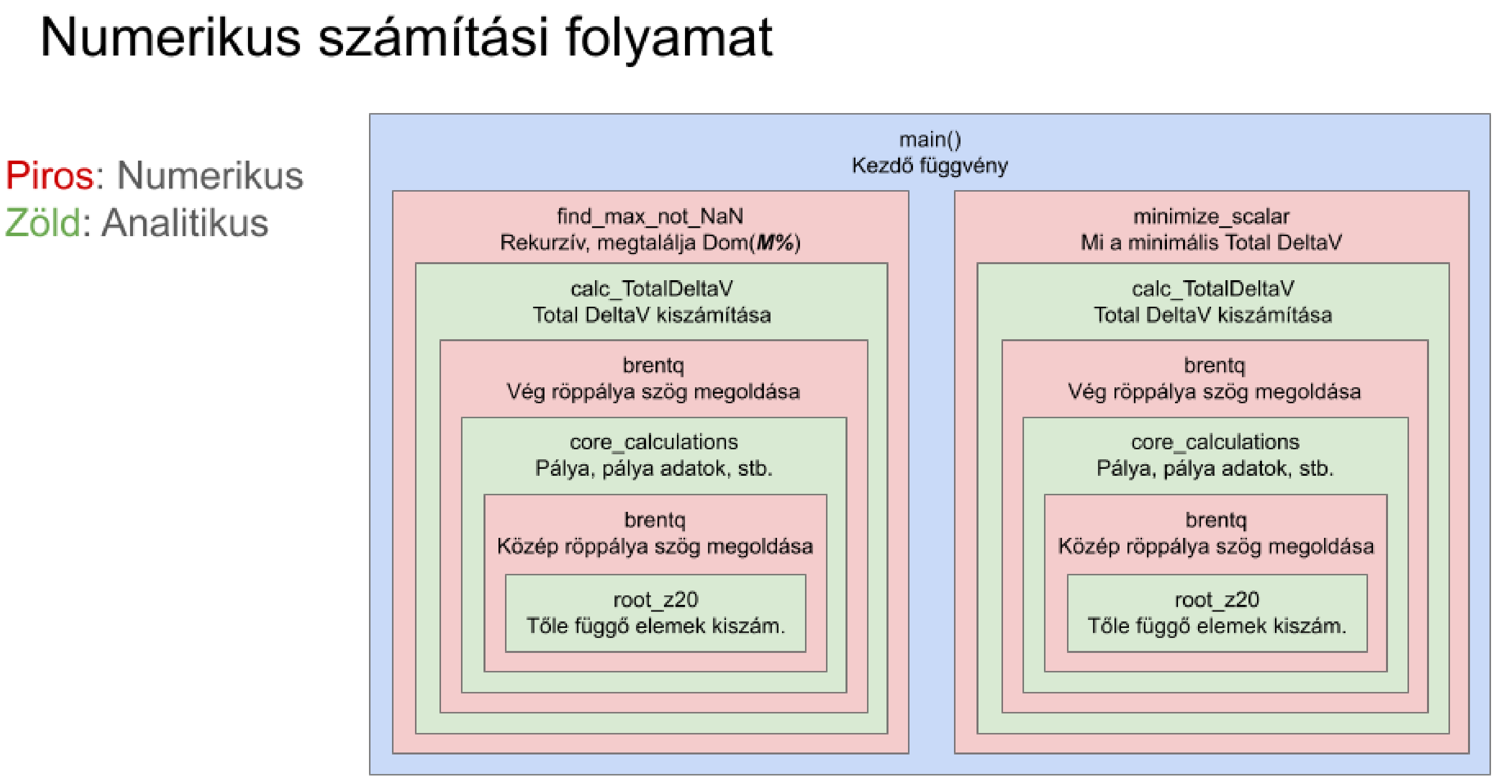
Az öt paraméter, amit optimalizálni kell: Δv_{total}, r_{Ap}, γ_f, z_{20}, $M\%$
A számítás menete, sorrendben:

1. Meg kell találni $M\%$-nek azt az intervallumát, ahol még van olyan γ_f, amivel elérjük a kívánt r-t.
2. Választunk egy $M\%_i$-t és egy γ_i-t
3. Kiszámoljuk a γ_i-hez tartozó z_{20}, közép röppályaszöget
4. Megnézzük, hogy ezzel a γ_i-vel eléri-e az r_{Ap} a kívánt r-t // Ha nem, akkor másik γ_i
5. Ha igen, akkor megnézzük, mennyi a Δv_{total} // Kevesebb vagy több
6. Választunk egy másik $M\%_i$-t

Az optimalizáció végén megkapjuk a minimális Δv-t, amit el kell használnunk a pályára álláshoz.

---

[7]Például a `find_max_not_NaN` függvény egy pair codinggal készített, erre az esetre szabott rekurzív függvény.



2. ábra. A számításokat végző program ábrája

Alkalmazott numerikus módszer típusok

Brent: A paraméterek közül kettő (z_{20} és a r_{Ap}) a scipy.optimize/*brentq* függvényével lett kiszámítva.
A *brentq* függvény nullára optimalizáláskor lehet használni, ami összetett, de hatékony. Egy biztonságos felező módszert ötvöz inverz kvadratikus extrapolációval.[8]

Ezek közül az r_{Ap} a legkézenfekvőbb, hiszen van egy r amit el akarunk érni, így:

$$f_{Ap} = r - r_{Ap} \tag{14}$$

Ha ez a különbség 0, akkor megvan az a γ_i, amivel el lehet érni a kívánt r-t.

A másik a z_{20}, amit Teofilatto és tsai. számításaiból származik.
A lényeg, hogy létezik olyan $f_{1,2}$ függvény, ami:

$$t_{b1} = f_1(z_{20})$$
$$t_{b2} = f_2(z_{20})$$
$$\frac{t_{b1}}{t_{b2}} = \frac{f_1}{f_2}$$

Tehát:

$$f(z_{20}) = \frac{t_{b1}}{t_{b2}} - \frac{f_1}{f_2} \tag{15}$$

[8] A módszer Richard Brent által lett kidolgozva [15]

Ha ez a különbség 0, akkor megvan a γ_i-hez tartozó z_{20} szög.

minimize_scalar: Maga a Δv_{total} nem számítható ki *brentq*-val, minimumot kell keresni, nem zéruspontot. Ezért a scipy.optimize egy másik függvényét hívtuk segítségül: *minimize_scalar*.

find_max_not_NaN: Ez a függvény találja meg a $M\%$-nek azt az intervallumát, ahol a $f_{Ap} = 0$ értelmezett. Ez egy külön erre a célra készített függvény, ami a felezőmódszerhez hasonlóan működik, de nem zéruspontot keres, hanem az értelmezési tartomány határát.

A megoldások: γ_f és $M\%$ nincs használva ilyen numerikus módszerben, csupán az intervallumukat keressük meg numerikusan. Tehát ezek a keresett változók és ha megvan a γ_f és $M\%$, akkor megkapom a megoldást, ami a Δv_{total}.

A számítás általánossága

Természetesen valamennyire általános ez a megoldás. A két fokozatú rakétákra, amiknek legalább az első fokozat egészére szükségük van a pályára álláshoz. Teofilatto és tsai. módszere kiterjeszthető N fokozatra is, és a kellő tudással meg lehet azt csinálni, hogy több fokozatból engedjünk állítani felhasznált százalékot. De egy általánosítása volt a számításnak, hogy az első fokozatra szükség van. Ennek az indoklása annyi, hogy sem KSP-ben, sem RSS-ben nem volt olyan első fokozat, ami elérni az első kozmikus sebességet a bolygó körül, de lehetne. Vagy éppen három fokozata van, valóságban vannak rá példák.[9]
Illetve nem kizárt hogy ki lehet terjeszteni magas Föld kürüi pályákra is a számítást, de ez már nem tartozik a projekt kereteibe.

Talált hibák a [2] forrásban

Teofilatto és tsai. munkája lenyűgöző, de nem hibátlan. Ami megnehezítette a munkámat és akár több nappal is hátravetett.
Mivel az én példáim is két fokozatra épülnek, gyorsan le is tekertem a 4. fejezetig (*Analytic Derivation of Launcher Trajectories*). Ahol pont két fokozatos példát ecsetelgetnek. Itt a (20) egyenlet teljesen el van írva. $\bar{n}_{1,2}$ (átlagos *TWR*) össze-vissza van írva, és hiányoznak a zárójelek.

$$t_{b2} = \frac{A_2}{g}\left(\frac{z_{2f}^{\bar{n}_2-1}}{\bar{n}_2-1} + \frac{z_{2f}^{\bar{n}_2+1}}{\bar{n}_1+1} - \frac{z_{20}^{\bar{n}_2-1}}{\bar{n}_2-1} + \frac{z_{20}^{\bar{n}_2+1}}{\bar{n}_1+1}\right) \tag{16}$$

Ez a ChatGPT-nek rögtön feltűnt, mikor rákérdeztem, hiszen mint arra felhívta a figyelmemet, ez egy Newton-Leibniz tételes kiértékelés eredménye. Mint az megfigyelhető a (26) egyenletben is, ahol 3 fokozatra írnak példát.
És itt szerencsére a (25) egyenletben javítják, amit elírtak.

$$t_{b2} = \frac{A_2}{g}\left[\left(\frac{z_{2f}^{\bar{n}_2-1}}{\bar{n}_2-1} + \frac{z_{2f}^{\bar{n}_2+1}}{\bar{n}_2+1}\right) - \left(\frac{z_{20}^{\bar{n}_2-1}}{\bar{n}_2-1} + \frac{z_{20}^{\bar{n}_2+1}}{\bar{n}_2+1}\right)\right] \tag{17}$$

Ahol a t_{b2} a második fokozat kiégésének ideje; A_2 nincs megnevezve, de abból, ahogy használva van, úgy lehetne elnevetni, mint egy röppálya skálázó paraméter, ami mindig egy a fokozatra

[9] Az indíték az, hogy a hajtóművek I_{sp}-je általában 280-350 köré esik. És körülbelül $I_{sp} \cdot 10$ érdemes egy fokozat Δv-jét tenni, és ehhez Föld körül három fokozatra van szükség. Ilyen a Proton rakéta család, és a méltán híres Saturn V, és a Soyuz rakéták is.

jellemző; z pedig a röppályaszög, z_{2f} a második fokozat kifogyásánál, z_{20} pedig az első fokozat leválasztásánál.
Egy másik hibát szintén ugyanezen a helyen én vettem észre: $f_{1,2}$ meg volt cserélve:

$$\frac{t_{b2}}{t_{b1}} = \frac{f_1}{f_2} \tag{18}$$

Helyesen az egyenlet:

$$\frac{t_{b2}}{t_{b1}} = \frac{f_2}{f_1} \tag{19}$$

Egy harmadikat megint a ChatGPT. A $h(z)$ a (14) egyenlettől végig rosszul van írva. Ha megfigyeljük a 8. oldal legalját, akkor ott van a $\partial_z(h)$, amit integrálni kell z szerint, hogy megkapjuk $h(z)$-t. Az integrálást elvégezve:

$$\int \frac{dh}{dz} dz = \frac{A^2}{g} \int \left(z^{2\bar{n}-3} - z^{2\bar{n}+1} \right) dz \tag{20}$$

$$h(z) = h(z_0) + \frac{A^2}{g} \left[\frac{z^{2\bar{n}-2}}{2\bar{n}-2} - \frac{z^{2\bar{n}+2}}{2\bar{n}+2} \right]_{z_0}^{z_f} \tag{21}$$

Náluk a pirosra színezett blokkban tűnt el a kettes:

$$h(z) = h(z_0) + \frac{A^2}{g} \left[\frac{z^{2\bar{n}-2}}{2\bar{n}-2} - \frac{z^{2\bar{n}+2}}{\bar{n}+2} \right]_{z_0}^{z_f} \tag{22}$$

3. A kísérletek

A kísérletek elvégzéséhez a Kerbal Space Program nevű "játékos szimulátort" használtam. Egy alap játékos konfigurációban és egy modokkal megváltoztatott konfigurációban, ami lehetővé tette, hogy a Földön is teszteljem azt, amit kiszámítottam, illetve mindezt igazi hajtóművekkel tegyem.

3.1. A rakéták

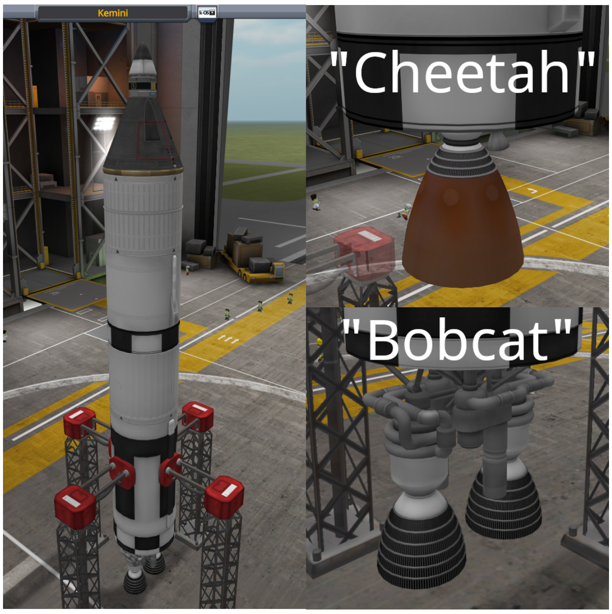
A rakéták pontos számítás szempontjából lényeges adata megtalálható egy YAML fájlban [14] GitHub repository-ban a Python kód mellett.

KSP

Az alap játékban egy a Gemini programban használtra hajazó rakétát raktam össze.
A fokozatok hajtóművei:

1. "Bobcat"

2. "Cheetah"



3. ábra. Képek a rakétáról a hangárban.

KSP-ben a kilövőállás hajszálra rajta van az egyenlítőn, így nem kell egyszerűsíteni sem a Δv_{lat}-ön.

RSS

Itt nem céloztam semmit, amire hasonlítania kéne a rakétának. A hajtóművek orosz gyártásúak, de csak azért, mert azok tűntek a legmegfelelőbbnek.
A fokozatok hajtóművei:

1. RD-253
2. RD-0124

4. ábra. Képek a rakétáról a hangárban.

Az RD-253 hajtóművek a Proton rakéta első fokozatát alkották, egy nagyon megbízható, sokat használt hajtómű. A "Salyut", "Mir" programokban is alkalmazták. 2012-ben lett kivezetve, azóta a fejlesztett RD-275M van szolgálatban. [16]

Az RD-0124 a Soyuz rakéták harmadik fokozatát alkotja, és még a mai napig használatban van. És én is ugyanarra a célra használom, ami az hogy egy felsőbb fokozatos hajtómű legyen. A hajtómű egy érdekes konstrukció, ugyanis egy turbópumpa van, ami négy fúvóka égésterébe pumpálja az üzemanyagot. Tehát a négy fúvóka ellenére ez mégis egy hajtómű. [17]

Fontos szempont volt a hajtómű választásnál, hogy a második fokozat minimum kétszer gyújtható legyen. Mert az RD-253, mivel első fokozatra tervezett, ezért nem számolnak azzal, hogy le lesz állítva, míg az RD-0124 többször (kétszer) begyújtható a körpálya kialakítása miatt.

RSS-ben a legtöbb valóságban használt kilövőállás a rendelkezésemre áll, így a Francia Guyanában található Kourou kilövőállást használtam. De még az elején Cape Canaveral-nél is szórakoztam. [14]

3.2. Az irányítás

Mint azt a legelején említettem vagy kézi vagy fél automata irányítást fogok használni.
Kézi KSP-ben, de csak mert az hozta a legkisebb Δv_{total}-t.
És RSS-ben a *MechJeb2* nevű mod segítségével egy fél automata vezérlést.

Az eredményeket szintén a *MechJeb2* rögzítette.

4. Kiértékelés

A kiértékelést a számításokkal kezdem, és azután következik a kísérlet, majd az összehasonlítás, tanulságok levonása.

4.1. Számítások eredménye

Sok adat volt végül kiírva, segítve a hibák megtalálását ezt megtartottam a végső verzióban is de itt csak a legfontosabbakat fogom listázni.[10]

Paraméter	**KSP**	**RSS**	**M.egység**
Δv_{total}	3223.72	8930.49	m/s
Δv_g	651.5	1102.75	m/s
Δv_H	282.05	42.2	m/s
γ_f	7.16	0.31	fok
$M\%$	64.251	85.234	%

1. táblázat. Fontosabb mért adatok eredménye.

A számításokban azt figyeltük meg, hogy $M\%$ szélső értékhez közelít, ezért volt szükség a *find_max_not_NaN* függvényre. Mert egy bizonyos százalék felett nincs olyan γ_f amivel az *Apoapsisunk* elérné a 250 km-t. Ekkor a *brentq* hibát ad, miszerint az intervallum nem halad át a nullán.
Ez azt jelenti, hogy a körpálya kialakítására szükséges Δv-t minimalizálni kell. Tehát az első gyorsítási szakasz közben minnél inkább késleltetni kell a kívánt *Apoapsis* elérését, így minnél

inkább megemelve a *Periapsis*-t.
Ez feltételezésem szerint akár lehet az *Oberth-effektus* miatt, de ennek az igazolása nem tartozik bele a kísérlet kereteibe.

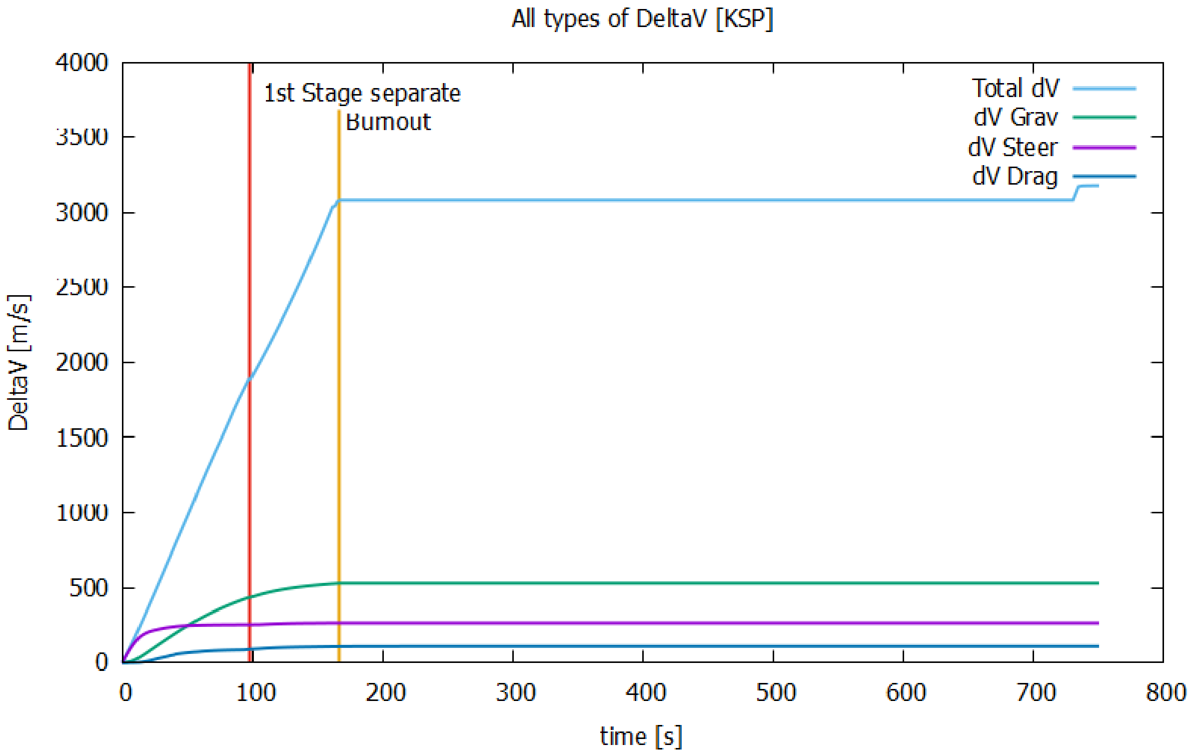
4.2. A kísérletek eredménye

Itt a *MechJeb2* sokkal részletesebben bontja le a Δv_{total}-t, így az (1) egyenletben megemlített részek is bekerülnek. Ez egy eredmény a sok közül. Több mérést is végeztem de egyik sem érte el a $\Delta v_{total} \leq 8900\, m/s$-ot.

Paraméter	**KSP**	**RSS**	**M.egység**
Δv_{total}	3176.07	8956.02	m/s
Δv_g	529.71	803.89	m/s
Δv_d	109.96	198.37	m/s
Δv_{steer}	264.77	672.12	m/s
Δv_H	96.5	369.9	m/s
γ_f	1.73	0	fok

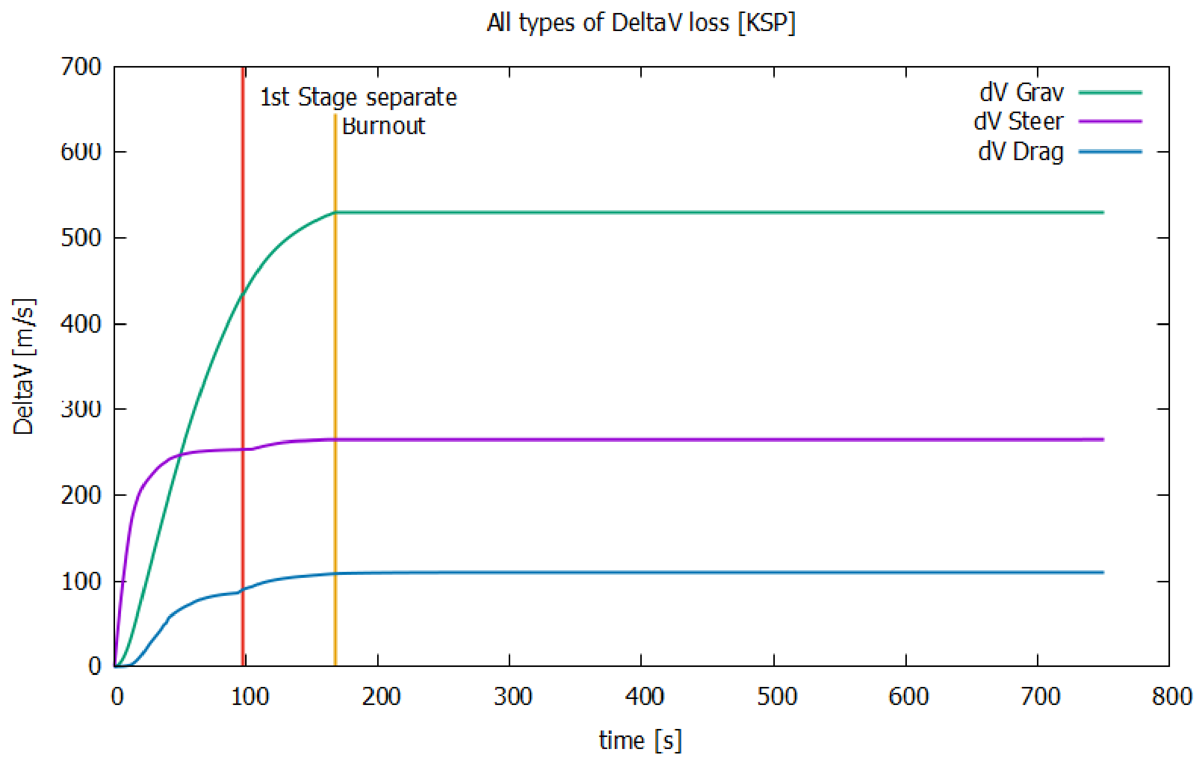
2. táblázat. Fontosabb mért adatok eredménye.

Itt a mérés módja miatt ábrák is készíthetők. Jó sok érdekes ábrát sikerült készítsek ami érdekes lehet, de az összesen 14 kép lenne, így ide csak a Δv szempontjából fontosakat rakom be. Ha pedig valakit érdekel a többi is, [14]-ben megtalálja. Ha pedig egyéb más összehasonlítást szeretne végezni akkor szintén fel lesznek töltve a csv fájlok amiket a *MechJeb2* készített.



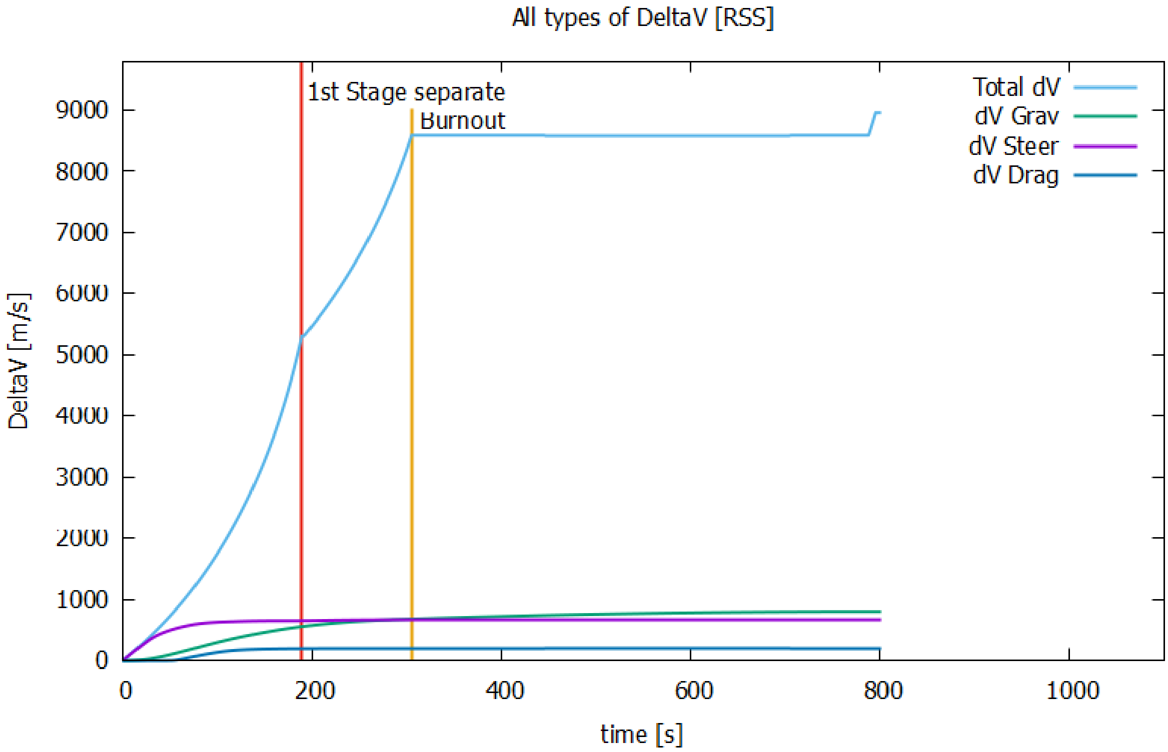
5. ábra. Minden Δv fajta.

[10]Akit érdekel minden mért adat az a [14]-ben megtalálja.



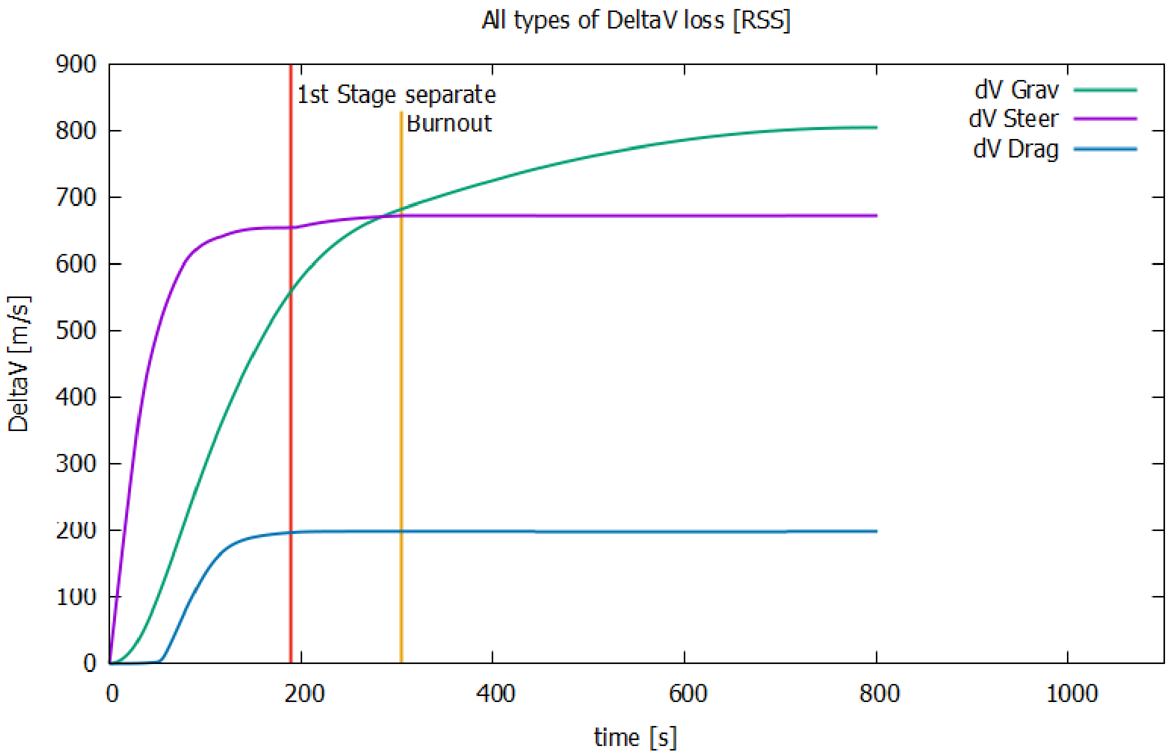
6. ábra. Minden elvesztett Δv fajta.

Már a fenti 5 és 6 KSP ábrákból is látszik, hogy a Δv_d, a légellenállás 100 m/s-os nagyságával igazán elhanyagolható, a teljes $\approx$ 3200 m/s-hoz képest. És a legnagyobb a Δv_g, mint amire számítottunk. De már itt is felüti a fejét a Δv_{steer}, ami abból származik, hogyha nem pontosan *Prograde* irányba égetünk hanem valamilyen szögben tőle. És az elvesztett Δv onnan származik, hogy a sebességvektorunkat össze kell hoznunk a haladási iránnyal. Megfigyelhető a 8 ábrán is, hogy ez a Δv_{steer} a mozgás elején jelenik meg és egy ideig több mint a Δv_g, amíg a Δv_{steer} konstanssá nem válik és végül a Δv_g nagyobb lesz nála.



7. ábra. Minden Δv fajta.

A 7 ábrán megfigyelhető, hogy az arányok megváltoztak, míg az 5 ábrán $\Delta v_{total} \approx 3200\,m/s$, addig egyik Δv veszteség sem jut túlságosan túl az $500\,m/s$-on. Viszont a 7 ábrán a Δv_{total} megháromszorozódott, de a Δv veszteségek nagyjából megkétszereződtek. Ez még jobban látszik a 6 és a 8 ábrák összevetésével.



8. ábra. Minden elvesztett Δv fajta.

A nagyjából megkétszereződés alól kivételt képez a Δv_{steer}, ami majdnem a fő égetési szakasz egészén nagyobb, mint a Δv_g.

4.3. Összevetés

Az én modellemben a légellenállást elhanyagoltam, mert nehéz kiszámolni és nem akkora mértékű, sokkal fontosabb a Δv_g kiszámolni [2], [1]. Ez egy jó döntésnek bizonyult.

De miért is lesz ilyen kis mértékű? Mert minél előbb el akarjuk hagyni a légkört. Természetesen ha olyan rakétákkal számolnánk amik a Földhöz közel mozognak nagy sebességgel akkor nem lehetne elhanyagolni. Még felvetülhet a kérdés: De ha minél gyorsabbak vagyunk, akkor nem vesztünk több Δv-t a légellenállás miatt? Ami igaz is, de ha lassan mennénk, akkor a gravitáció miatt vesztenénk többet, mert a Δv_g függ az időtől [1]. És mint azt a saját adataink a 6 és a 8 ábrán mutatják a Δv_g mindig nagyobb lesz mint a Δv_d. Illetve ha akarnánk sem tudnánk a legsűrűbb rétegeken túl gyorsan végig menni mert a rakéták 1.2-1.8 *TWR*-el kezdenek, ami temészetesen majd növekszik, de a tengerszint közeli szakaszon szépen "végigsétál" a rakéta mielőtt elég sebességet gyűjetene ahhoz hogy a légellenállás nagyobb szerepet kapna.

Viszont amivel nem számoltam, hogy ekkora mértékű lesz az a Δv_{steer}. Tehát beszéljünk ennek a lehetséges okairól:

Mind az [1] és [2] forrás említést tesz róla de nem állítja hogy ekkora mértékű lenne. Én a kísérlet közben egy már az *Elméleti háttér* első bekezdésében leírt módszerhez hasonlóan végeztem el a pályára állást:[11]

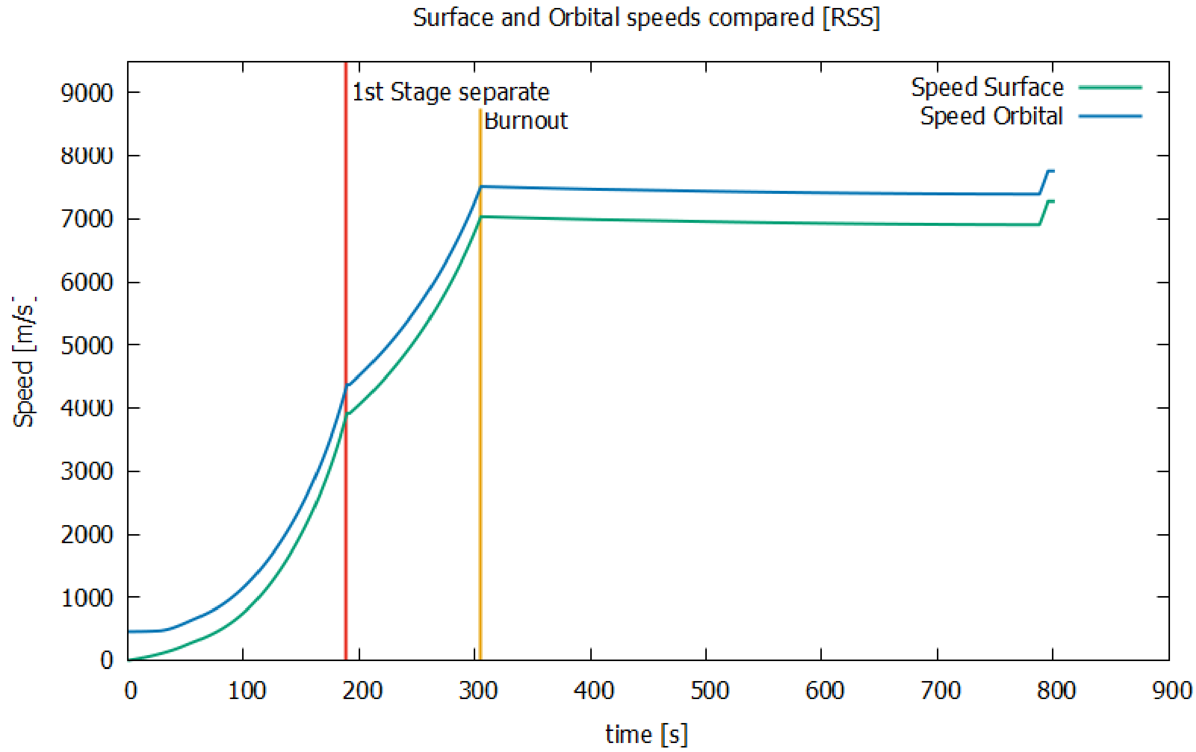
1. Kis emelkedés
2. Bedöntés
3. Gravitációs forduló
4. Körpálya

Én nem alkalmaztam külön algoritmust, hanem *MechJeb2*-nek egy félautomata irányítását ahol én tudom állítani a dőlés mértékét, így követve egy közel optimális pályát.
Így lehet, hogy a kezdeti bedőlése a *MechJeb2*-nek túl erős. Viszont ha megfigyeljük az eredményeket nem veszünk észre akkora különbséget. Ez lehet azért mert Teofilatto és tsai. belekalkulálták a Δv_{steer} a számításaikba, vagy inkább éppen jól hibás a számítás, hogy ezt ellensúlyozza, de bizosra nem tudom megállapítani. Természetesen mivel ez csak egy szimuláció előfordulhat, hogy nem pontos, vagy éppen a *MechJeb2* számolja rosszul a Δv_{steer}-et.

A Bolygó forgása

Sokáig tartogattam a Δv_{lat}-öt, hogy most kedvünkre értékelhesük ki annak a hatását.
A logika mögötte, hogy a Föld forog a talpunk alatt tehát van egy bizonyos kezdő kerület sebességünk amennyit a Föld "segít" nekünk a pályára állásban[12].
Ahhoz hogy ezt vizualizálni tudjuk segít a következő 9 ábra. Figyeljül meg, hogy az orbitális sebesség mindig nagyobb és a felszíni sebeség a nulla. Ez a különbség amit a Föld segít.

[11] A [14]-ben fel vannak videó is egy ilyen mérésről.



9. ábra. Földi és Orbitális sebességek összehasonlítva.

A kiszámolása:

$$\Delta v_{lat} = v_{lat} = \omega R \cos \vartheta = \frac{2\pi R}{T} \cos \vartheta \tag{23}$$

Ahol R a bolygó sugara, T egy nap a bolygón, és $\cos \vartheta$ pedig a kilövőállás magasságszöge[13]. Mivel ez mástól nem függ ezért egy konstans így később is le lehet vonni.
Mivel KSP-ben az egyenlítőről indítottunk, RSS-ben pedig a Francia Guyana-ból indított adatokat értékeltük, ezért a $\cos \vartheta$ tagot szimplán elhanyagoljuk.

Pamaméter	**KSP**	**RSS**	**M.egység**
Δv_{lat}	174.53	463.31	m/s

3. táblázat. A bolygók forgásából származó sebesség.

Adatok összehasonlítása

Ha az előbb kiszámolt eredményket egymás mellé rakjuk a következő táblázatot kapjuk:

Eredménye	**KSP**	**RSS**	**M.egység**
Kísérlet	3176.07	8956.02	m/s
Számítás	3223.72	8930.49	m/s
$-\Delta v_{lat}$	3049.19	8467.18	m/s

4. táblázat. Δv_{total} a bolygók forgását beszámítva és az eredeti eredményekkel öszehasonlítva.

[12] Természetesen akkor ha a forgással megegyező irányba indulunk. De hát ki akarná még azt is leküzdeni amikor pont segítene. Ezen logikát követve a keletre indítás a konvenció.
[13] Vagy Alt-szög, szélesség, vagy latitude

Így a számítások azt sugallják, hogy ezek a Δv_{total} értékek elegendőek lennének a pályára álláshoz. De ezt kísérlettel nem tudom alátámasztani. Nem kizárt hogy KSP-ben lehetne még $\approx 150\,m/s$-ot faragni, de ez a Föld közel $\approx 500\,m/s$-ra már nem mondható el. De nem is feltétlen az a kérdés, hogy fizikailag lehetséges-e, hanem az hogy érdemes lenne egy ennyi Δv_{total}-vel rendelkező rakétát tervezni. Amire a válasz, hogy nem, hiszen ezek a minimum értékek. Tehát *lehet*, de a biztonság és egyéb ki nem számítható nehézségek és hibák miatt nem érdemes cseppre kimérni az üzemanyagot.
Valamint a számítás, részben analitikus mivolta miatt bizonyos egyszerűsítéseket alkalmaz, tehát lehet azt mondani, hogy a kimért adatok alapján pont annyi a hiba amennyi a Δv_{lat} lenne.
De amennyiben a játékkal játszunk, még érdemes ennek a tetejébe is egy kis plusz üzemanyagot vinni $\approx 200 - 400\,m/s$-ot. Ezeket a Δv térképek adatai alapján mondom, [18], [19] amiket a Kerbal Space Program közösség készített az évek alatt.

Alternatív ötlet

Egyszer a kód hibás volt és úgy viselkedett ami a Hohmann módszerre emlékeztetett engem. Így jutott eszembe ez a megoldás.

Mi lenne, ha Hohmann módszerrel számolnánk Δv_{total}-t?
Semmi mást nem kell tennünk, mint kiszámoljuk, mennyi a kerületi sebesség R magasságban, majd a Hohmann módszert alkalmazva felmegyünk egy $R + h$ magasságú kör pályára.

Ez lenne a leghatékonyabb módja a pályára állásnak, ha nem lenne mindenféle veszteségünk, azért mert ki kell kerüljük a légkört, illetve fel kell emelkedjünk. Mert nem egy pillanat alatt fogunk 7.9 km/s-ra szert tenni. A Python kód ezt is kiszámolja, így adódik a következő táblázat:

Eredménye	KSP	RSS	M.egység
Hohmann	2735.33	8134.84	m/s
Hohmann + Δv_g	3386.82	9237.58	m/s
Hohmann + Δv_g - Δv_{lat}	3212.29	8774.27	m/s

5. táblázat. Tisztán Hohmann módszerrel a Δv_{total}, és Δv_g beleszámolva.

Az eredmény bíztató, mert nem kaptunk tisztán Hohmann módszer alatti értéket a számításainkban, ami azt jelentette volna, hogy hibás a számításunk. A másik két sor pedig alkalmazza az (1) gondolatmenetét, miszerint, hogyha valahogyan tudnánk Δv_g-t akkor a Δv fajták összeadásával megkapjuk Δv_{total}-t. Így megint csak egészen közeli értékeket kapunk a valósághoz.[14]

Ha még pontosabb eredményeket akarunk kapni arra egy teljesen numerikus megközelítés, ami nem hanyagolja el a légellenállást és a Δv_{steer}-et megoldást nyújthat.

5. Összefoglalás

A projekt célja az volt, hogy állapítsuk meg az alacsony Föld és Kerbin körüli pályára álláshoz szükséges Δv-t. Teofilatto és tsai. munkája alapján a pályára állás egy részét analitikusan megoldva, majd a kezdőfeltételeket numerikusan hozzáigazítva a kívánt pályaadatokhoz sikerült megállapítani a Föld és Kerbin körüli pályára állás Δv szükségleteit. Ezeket a körülbelüli eredményeket pedig Kerbal Space Programmal sikerült alátámasztani, és visszaigazolni a közösség által ajánlott értékeket.

[14] A Δv_g a Python kód által megállapított érték.

Hivatkozások

[1]J. G. Leishman, *Introduction to Aerospace Flight Vehicles*, eBook ISBN: 979-8-9852614-0-0, https://eaglepubs.erau.edu/introductiontoaerospaceflightvehicles/front-matter/4811/ (Embry-Riddle Aeronautical University, Daytona Beach, Florida, 2023).

[2]P. Teofilatto, S. Carletta és M. Pontani, “Analytic Derivation of Ascent Trajectories and Performance of Launch Vehicles”, Applied Sciences **12**, 10.3390/app12115685 (2022).

[3]NASA Glenn Research Center, *Specific Impulse*, https://www.grc.nasa.gov/www/k-12/airplane/rockth.html, NASA K-12 educational resource.

[4]NASA Glenn Research Center, *Rocket Thrust Equasion*, https://www.grc.nasa.gov/www/k-12/airplane/specimp.html, NASA K-12 educational resource.

[5]B. Weber, *Orbital Mechanics & Astrodynamics*, https://orbital-mechanics.space/intro.html, Online educational textbook.

[6]M. A. Karabeyoglu, *AA284A Lecture 7: Rocket Nozzles and Performance*, https://web.stanford.edu/~cantwell/AA284A_Course_Material/Karabeyoglu\%20AA\%20284A\%20Lectures/AA284a_Lecture7.pdf, AA284A: Space Propulsion.

[7]W. contributors, *Hohmann transfer orbit*, https://en.wikipedia.org/wiki/Hohmann_transfer_orbit, Accessed: January 11, 2026.

[8]W. contributors, *Eccentricity vector*, https://en.wikipedia.org/wiki/Eccentricity_vector, Accessed: January 11, 2026.

[9]f. Reddy, *PEGAS - Powered Explicit Guidance Ascent System - devlog*, https://forum.kerbalspaceprogram.com/topic/142213-pegas-powered-explicit-guidance-ascent-system-devlog/, Accessed: January 11, 2026.

[10]KSP-RO Group, *Ferram on Ascent Profile and TWR*, https://github.com/KSP-RO/RealismOverhaul/wiki/Ferram-on-Ascent-Profile-and-TWR, Accessed: January 11, 2026.

[11]Space Stack Exchange, *How to calculate apoapsis of sub-orbital trajectory?*, https://space.stackexchange.com/questions/4727/how-to-calculate-apoapsis-of-sub-orbital-trajectory, Accessed: January 11, 2026.

[12]Reddit KerbalAcademy, *How do I calculate delta V losses due to drag?*, https://www.reddit.com/r/KerbalAcademy/comments/63nzq6/how_do_i_calculate_delta_v_losses_due_to_drag/, Accessed: January 11, 2026.

[13]M. Aben, *The Vis-Viva Equation | KSP Let’s Do The Math*, YouTube, https://www.youtube.com/watch?v=7AkakB5TOSw, Accessed: January 11, 2026.

[14]G. Ádám, *Orbit Project*, https://github.com/Adam-Gillich/Orbit-Project, GitHub repository, release v1.0, 2026.

[15]docs.scipy.org, *scipy.optimize.brentq*, https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.brentq.html, Accessed: January 11, 2026.

[16]Wikipedia contributors, *RD-253*, https://en.wikipedia.org/wiki/RD-253, Accessed: January 11, 2026.

[17]Wikipedia contributors, *RD-0124*, https://en.wikipedia.org/wiki/RD-0124, Accessed: January 11, 2026.

[18] Kerbal Space Program Wiki, *KSP Subway DeltaV map*, https://wiki.kerbalspaceprogram.com/wiki/Cheat_sheet, Accessed: January 11, 2026.

[19] ucarion, *I made a delta-v subway map of the Solar System*, https://www.reddit.com/r/space/comments/29cxi6/i_made_a_deltav_subway_map_of_the_solar_system/, A képet meg kell nyitni új lapon és olvashatóvá válik. Accessed: January 11, 2026.