Documentation of Pre V1.0 KSP Physics

Version 1.3.0

https://github.com/mhoram-kerbin/ksp-physics-documentation
http://forum.kerbalspaceprogram.com/threads/93426-Physics-of-KSP

Mhoram Kerbin mhoram.kerbin@gmx.de

July 18, 2015

Abstract

This document summarizes the Pre V1.0 stock physics of Kerbal Space Program [1]. It should be valid for versions 0.21 to 0.25.0. Some equations are also applicable to real life situations - if you ever happen to stumble into one.

Aspects of Stock Aerodynamic have changed in KSP V 1.0 So these sections do not apply anymore.

Contents

Chapter 0

Preface

0.1 Introduction

During my attempts to use PSOPT as a method for generating ideal ascent trajectories for rockets, I needed to implement the physics of KSP in C/C++. So I had to gather all the physics related information again after my Ascent Optimizer Perl-project [2]. Since they are available at quite different places, this document is intended to summarize them in a single central place.

0.2 Document Structure

Part I describes how the Physics Engine of KSP works. Part II describes how some in game physical properties can be calculated. Appendix 14 contains a desription of the differences between the real world and KSP-Physics.

0.3 Contributions

I want to thank the following people for helping to improve this document.

- Arrowstar (Reference Frames)

0.4 Relevance for Physics-Changing Addons

FAR [20] and NEAR [21] implement a completely different atmosphere and only the non-atmospheric descriptions of this text can be applied to them.

While Principia [23] intends to change the orbital mechanics completely it remains to be seen how much of the rest of this text can be applied to it.

0.5 Basic Knowledge

For getting along with this document besides knowledge about mathmatics and physical units a basic understanding of the following aspects is required.

- Euclidian Vectors [24] and Vector Operations including scalar product [25] and cross product [26]
- Trigonometric functions [27]
- Differentiation including Newton's notation [28]
- Exponentiation [29] and Logarithm [30]

Part I

Inner Workings of KSP

Chapter 1

Conventions

We denote by $\overrightarrow{P}$, $\overrightarrow{V}$, d and M the position vector, velocity vector, drag coefficient and the mass of the ship respectively.

$\overrightarrow{P}_X$, $\overrightarrow{P}_Y$ and $\overrightarrow{P}_Z$ denote the X, Y and Z components of the vector $\overrightarrow{P}$.

The Z-Axis points from the planets core to the northpole.

Distance denotes the distance to the planets center and altitude the distance to the surface sea level.

Chapter 2

Physical Constants

2.1 Global Constants

The conversion factor between pressure and density is according to [3].

$$CF := 1.2230948554874 \frac{kg}{m^3 \cdot atm} \tag{2.1}$$

The gravitational constant G is according to [4].

$$G := 6.674E - 11N \left(\frac{m}{kg}\right)^2 \tag{2.2}$$

2.2 Planets

Planets are defined by the following parameters

- Name
- Mass of Planet $PMass^{(Name)}$ in kg
- Planetary Radius $PRadius^{(Name)}$ in m
- Scale Height $PSH^{(Name)}$ in m
- Pressure at Sealevel $p_0^{(Name)}$ in atm
- Rotation Period $RP^{(Name)}$ in s
- Radius of Sphere of Influence $Soi^{(Name)}$ in m [1]

The KSP Wiki [5] contains information about each celestial body.

[1] It should be noted that the Sphere of Influence is actually calculated as described in [34].

2.2.1 Kerbin

According to the KSP Wiki for Kerbin the values are:

$$\begin{aligned}
PMass^{(Kerbin)} &:= 5.2915793E22kg \\
PRadius^{(Kerbin)} &:= 600000m \\
PSH^{(Kerbin)} &:= 5000m \\
p_0^{(Kerbin)} &:= 1atm \\
RP^{(Kerbin)} &:= 21600s \\
Soi^{(Kerbin)} &:= 84159286m
\end{aligned}$$

Chapter 3

Atmosphere

3.1 Atmospheric Formula

Now we will show how several atmospheric parameters are calculated based on [3].

AtmosphericHeight

$$AH^{(Name)} := -\ln\left(10^{-6}\right) \cdot PSH^{(Name)} \tag{3.1}$$

Pressure at altitude

$$p(alt) := \begin{cases} p_0^{(Name)} \cdot \exp\left(\frac{-alt}{PSH^{(Name)}}\right) & \text{if } alt < AH^{(Name)} \\ 0 & \text{if } alt \geq AH^{(Name)} \end{cases} \tag{3.2}$$

Density at altitude

$$\rho(alt) := CF \cdot p(alt) \tag{3.3}$$

3.2 Lift

The best description of lift can be found in the "How to calculate lift?" thread [14]

3.3 Air Intake

Jet Engines need the resource Intake Air [40] to run. This resource is harvested by Air Intakes [41] while flying in an atmosphere that contains Oxygen.

There are several factors influencing the amount of IntakeAir gathered. While the exact formula is unknown yet, some relations are more or less established. [42], [43]

- Intake Area is directly proportional to the amount
- Surface Velocity is probably directly proportional to the amount
- Atmospheric Pressure is probably directly proportional to the amount
- Angle of Attack is probably in a cosinus relation to the amount

Chapter 4

Surface

4.1 Latitude and Longitude conversion

From a Position Vector $\overrightarrow{P}$ we can calculate latitude and longitude by the following formula:

$$\mathrm{Lon}(\overrightarrow{P}) := \arctan\left(\frac{\overrightarrow{P}_Y}{\overrightarrow{P}_X}\right) \tag{4.1}$$

$$\mathrm{Lat}(\overrightarrow{P}) := \arctan\left(\frac{\overrightarrow{P}_Z}{\sqrt{\overrightarrow{P}^2_X + \overrightarrow{P}^2_Y}}\right) \tag{4.2}$$

Be advised that this happens to be only true if the inertial X is aligned with body X at t = 0. That's not the case for most KSP planets and moons. [36]

4.2 Surface Velocity

Due to the rotation of the planet, the surface velocity is different from the orbital speed.

The difference between the two speed vectors $DiffV(\overrightarrow{P})$ is the projected planets rotation to the position $\overrightarrow{P}$ of the rocket.

We can convert orbital $\overrightarrow{V}$ to surface velocity $\overrightarrow{SV}$ vectors with the following equation if we know the position $\overrightarrow{P}$.

$$\begin{aligned} DiffV(\overrightarrow{P}) &:= \left|\left|\overrightarrow{P}\right|\right| \cdot \frac{2\pi}{RP^{(Name)}} \cdot \cos\left(\mathrm{Lat}\left(\overrightarrow{P}\right)\right) \cdot \begin{pmatrix} -\sin(\mathrm{Lon}(\overrightarrow{P}) \\ \cos(\mathrm{Lon}(\overrightarrow{P})) \\ 0 \end{pmatrix} \\ \overrightarrow{SV} &= \overrightarrow{V} - DiffV(\overrightarrow{P}) \end{aligned} \tag{4.3}$$

Chapter 5

Forces

There are three forces that effect the flightpath of a rocket, namely Drag, Gravity and Thrust. This chapter details them.

5.1 Drag

Drag is the force applied to rocket based on the movement through the atmosphere and it is directed in an opposite direction to the movement.

It should be notet that since the atmosphere rotates with the planet, the surface velocity is the relevant velocity and not the orbital velocity.

5.1.1 Drag Coefficient

In KSP d is the drag coefficient of the rocket. It is a mass-based average of the drag coefficients of all parts of the rocket and is dimensionless.

It indicates how much it is decellerated by means of air friction.

Usually it is near to 0.2.

Consider a rocket consisting of a Mk1 Cockpit (dry mass of 1.25 ton, drag coefficient of 0.1), a FL-T800 Fuel Tank (total mass of 4.5 ton, drag coefficient of 0.2) and a LV-909 Liquid Fuel Engine (total mass of 0.5 ton, drag coefficient of 0.2). For this rocket the drag coefficient is:

$$\begin{aligned} d &= \frac{1.25 \cdot 0.1 + 4.5 \cdot 0.2 + 0.5 \cdot 0.2}{1.25 + 4.5 + 0.5} \\ &= \frac{1.125}{6.25} \\ &= 0.18 \end{aligned}$$

5.1.2 Drag Force

Since the drag force $\overrightarrow{F_D}$ is directly proportional to the cross-sectional area A of the rocket and this value is approximated in KSP by $A = 0.008\frac{m^2}{kg} \cdot M$ we get

$$
\begin{aligned}
\overrightarrow{F_D} &:= -0.5 \cdot \rho(alt) \cdot \left|\left|\overrightarrow{SV}\right|\right|^2 \cdot d \cdot 0.008\frac{m^2}{kg} \cdot M \cdot \frac{\overrightarrow{SV}}{\left|\left|\overrightarrow{SV}\right|\right|} \\
&= -0.5 \cdot \rho(alt) \cdot d \cdot 0.008\frac{m^2}{kg} \cdot M \cdot \left|\left|\overrightarrow{SV}\right|\right| \cdot \overrightarrow{SV} \qquad (5.1)
\end{aligned}
$$

5.2 Gravity

The gravity force $\overrightarrow{F_G}$ is directed towards the planets core and gets lower with increasing distance.

$$
\begin{aligned}
\mu^{(Name)} = LocalGravityParameter^{(Name)} &:= G \cdot PMass^{(Name)} \\
LocalGravity(distance) &:= \frac{LocalGravityParameter}{distance^2} \qquad (5.2) \\
\overrightarrow{F_G} &:= -M \cdot \mu^{(Name)} \cdot \frac{\overrightarrow{P}}{\left|\left|\overrightarrow{P}\right|\right|^3} \qquad (5.3)
\end{aligned}
$$

5.3 Engines

5.3.1 Rocket Engines

The Specific impulse I_{SP} (in seconds) describes the engine efficiency and is parameterized for an engine by ISP at sealevel and ISP in vacuum.

The ISP is linear in the pressure p and cut off at $1atm$ (citation needed). We can calculate the ISP based on the pressure p by

$$np(p) := \min(1atm, p) \qquad (5.4)$$

$$ISP(p) := ISP_{1atm} \cdot np(p) + ISP_{VAC} \cdot (1 - np(p)) \qquad (5.5)$$

According to [7] the conversion factor g_0 is not $9.81\frac{m}{s^2}$, but

$$g_0 \approx 9.82\frac{m}{s^2} \qquad (5.6)$$

So we get a fuel consumption $\dot{M}$ (change of mass) [11] of:

$$\dot{M} := \frac{\left|\left|\overrightarrow{F_T}\right|\right|}{ISP \cdot g_0} \qquad (5.7)$$

5.3.2 Jet Engines

Jet Engines are described in a nonlinear way. Details can be found in the “Fuel consumption as a function of atmospheric pressure“ thread [15]

5.3.3 ISP Calculation

A ship with multiple engines that have different ISP-profiles has an averaged ISP that is based on the ISP of all running engines.

The calculation of the ships ISP is described on the wiki [13].

A ship with n running engines, where each engine i has a thrust of F_{T_i} and a specific impulse of I_{SP_i} has a total ISP of:

$$I_{SP} := \frac{\sum_i F_{T_i}}{\sum_i \frac{F_{T_i}}{I_{SP_i}}} \tag{5.8}$$

5.3.4 Engine Thrust

The total ship thrust $\overrightarrow{F_T}$ with n running engines with thrusts $\overrightarrow{F_{T_i}}$ is

$$\overrightarrow{F_T} := \sum_i \overrightarrow{F_{T_i}} \tag{5.9}$$

This equation assumes that the engines do not induce a rotation onto the rocket. If this is however the case then the total thrust is lower than calculated in (5.9).

It should also be noted that this equation is a vectoraddition, meaning that the orientation of the engines is important.

5.4 Nomenclature

We will use the following abbreviations for the scalar thrust.

$$F_D := \left|\left|\overrightarrow{F_D}\right|\right|$$
$$F_G := \left|\left|\overrightarrow{F_G}\right|\right|$$
$$F_T := \left|\left|\overrightarrow{F_T}\right|\right|$$

Chapter 6

Acceleration

The rocket is subect to the three forces drag $\overrightarrow{F_D}$, gravity $\overrightarrow{F_G}$ and thrust $\overrightarrow{F_T}$. The acceleration as a vector $\overrightarrow{a}$ that the rocket experiences based on these forces is:

$$\overrightarrow{a} := M^{-1} \cdot (\overrightarrow{F_D} + \overrightarrow{F_G} + \overrightarrow{F_T}) \tag{6.1}$$

Chapter 7

Ship Rotation

There are several forces that induce a rotation on a ship.

- Torque from SAS modules and most command pods
- Engines that are not aligned with the center of mass
- Gimbal of engines
- Drag on rocketparts
- Lift on rocketparts (Wings & control surfaces)
- Steerable wheels on the ground
- Colisions with other objects

At the time of this writing there is to my knowledge no community-based comparison of the quantities of these effects available.

The best description of Torque can be found in the “If something has an SAS torque of ‘20’, what does that actually mean?” thread [22].

Chapter 8

Coordinate Conversion

An Orbit can be described either by cartesian coordinates of a position vector $\overrightarrow{P}$ and a velocity vector $\overrightarrow{V}$ or by Kepler Elements [8] that consist of

- Semi-major axis (a)
- Eccentricity (e)
- Inclination (i)
- Longitude of the ascending node (Ω)
- Argument of periapsis (ω)
- True Anomaly (ν), Mean Anomaly (M) or Eccentric Anomaly (E)

In KSP, Kepler Elements are used for objects that are on rails and Cartesian Coordinates for the ship you currently steer and others in its vicinity.

This chapter contains a description of the conversion between Kepler and Cartesian coordinates.

In order to convert between the two representations, it is necessary to keep the reference frames in order.

8.1 Kepler-Orbit-Position Conversions

For a Kepler Orbit there are three aequivalent parameters that describe the position of an object on the orbit.

- True Anomaly (ν) [16]
- Mean Anomaly (M) [17]
- Eccentric Anomaly (E) [18]

All of them are useful for different purposes. This chapter deals with converting them.

Note that we use M for both, Rocketmass and Mean Anomaly. The context usually indicates which of both is the intended one.

There are other equivalent parameterizations possible, like for example time since Periapsis.

KSP savegames store [38] this information with a reference to epoch, namely

- MNA: Mean anomaly at Epoch - the position of the orbiting body along the ellipse at a specific time
- EPH: Epoch - the reference time for the orbit

8.1.1 True Anomaly to Eccentric Anomaly

The Eccentric Anomaly E can be calculated from the True Anomaly ν and Eccentricity e by the formula

$$\tan(E) = \frac{\sqrt{1-e^2} \cdot \sin(\nu)}{e + \cos(\nu)}$$

$$E = \operatorname{atan2}\left(\sqrt{1-e^2} \cdot \sin(\nu), e + \cos(\nu)\right) \tag{8.1}$$

8.1.2 Mean Anomaly to Eccentric Anomaly

We calculate the eccentric Anomaly E by the formula $M = E - e \cdot \sin(E)$. Since this equation does not have a closed-form solution for E, we use the Newton-Raphson method [19] to approximate the solution.

In order to start the iteration, we need some initial values.

$$\begin{aligned} MaxError &:= 10^{-13} \\ target &:= \begin{cases} \pi & \text{if } e > 0.8 \\ M & \text{otherwise} \end{cases} \\ error &:= target - e \cdot \sin(target) - M \end{aligned}$$

After that we apply the following steps (8.2) to (8.4) until $|error| < MaxError$ or the procedure has found no solution within a specified number of runs.

$$prev := target \tag{8.2}$$

$$target := prev - \frac{error}{1 - e \cdot \cos(prev)} \tag{8.3}$$

$$error := target - e \cdot \sin(target) - M \tag{8.4}$$

The resulting Eccentric anomaly is $E := target$.

8.1.3 Eccentric Anomaly to True Anomaly

We get the True Anomaly by the formula

$$\tan\left(\frac{\nu}{2}\right) = \sqrt{\frac{1+e}{1-e}} \cdot \tan\left(\frac{E}{2}\right)$$
$$\nu = 2 \cdot \text{atan2}\left(\sqrt{1+e} \cdot \sin\left(\frac{E}{2}\right), \sqrt{1-e} \cdot \cos\left(\frac{E}{2}\right)\right) \tag{8.5}$$

8.1.4 Eccentric Anomaly to Mean Anomaly

We get the Mean Anomaly from the Eccentric Anomaly and the Eccentricity by the formula

$$M = E - e \cdot \sin(E) \tag{8.6}$$

8.1.5 True Anomaly to Mean Anomaly

Use the methods in chapters 8.1.1 and 8.1.4.

8.1.6 Mean Anomaly to True Anomaly

Use the methods in chapters 8.1.2 and 8.1.3.

8.2 Cartesian to Kepler

We can convert cartesian to kepler coordinates by the following equations that are explained in [9], where $\mu^{(Body)}$ denotes the gravitational Parameter of the planet, $\overrightarrow{h}$ the orbital momentum vector and $\overrightarrow{e}$ the eccentricity vector.

As reference frames we assume that inertial X is aligned with body X at t = 0.

$$a := \left(2 \cdot \left|\left|\overrightarrow{P}\right|\right|^{-1} - \frac{\left|\left|\overrightarrow{V}\right|\right|^{2}}{\mu^{(Body)}} \cdot \right)^{-1} \tag{8.7}$$

$$\overrightarrow{h} := \overrightarrow{P} \times \overrightarrow{V}$$

$$\overrightarrow{e} := \frac{\overrightarrow{V} \times \overrightarrow{h}}{\mu^{(Body)}} - \left|\left|\overrightarrow{P}\right|\right|$$

$$e := ||\overrightarrow{e}|| \tag{8.8}$$

$$i := \arccos\left(\frac{\overrightarrow{h}_Z}{\left|\left|\overrightarrow{h}\right|\right|} \right) \tag{8.9}$$

$$\overrightarrow{n} := \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \times \overrightarrow{h} = \begin{pmatrix} -\overrightarrow{h}_Y \\ \overrightarrow{h}_X \\ 0 \end{pmatrix}$$

$$\Omega := \begin{cases} 0 & \text{if } ||\overrightarrow{n}|| = 0 \\ \arccos\left(\begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \cdot \frac{\overrightarrow{n}}{||\overrightarrow{n}||} \right) & \text{if } \overrightarrow{n}_Y = \overrightarrow{h}_X >= 0 \\ 2\pi - \arccos\left(\begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \cdot \frac{\overrightarrow{n}}{||\overrightarrow{n}||} \right) & \text{otherwise} \end{cases} \tag{8.10}$$

$$\omega := \begin{cases} 0 & \text{if } ||\overrightarrow{n}|| = 0 \\ \arccos\left(\frac{\overrightarrow{n} \cdot \overrightarrow{e}}{||\overrightarrow{n}|| \cdot ||\overrightarrow{e}||} \right) & \text{if } \overrightarrow{e}_Z >= 0 \\ 2\pi - \arccos\left(\frac{\overrightarrow{n} \cdot \overrightarrow{e}}{||\overrightarrow{n}|| \cdot ||\overrightarrow{e}||} \right) & \text{otherwise} \end{cases} \tag{8.11}$$

$$\nu := \begin{cases} \arccos\left(\frac{\overrightarrow{P} \cdot \overrightarrow{e}}{\left|\left|\overrightarrow{P}\right|\right| \cdot ||\overrightarrow{e}||} \right) & \text{if } \overrightarrow{P} \cdot \overrightarrow{V} >= 0 \\ 2\pi - \arccos\left(\frac{\overrightarrow{P} \cdot \overrightarrow{e}}{\left|\left|\overrightarrow{P}\right|\right| \cdot ||\overrightarrow{e}||} \right) & \text{otherwise} \end{cases} \tag{8.12}$$

8.3 Kepler to Cartesian

We can also convert kepler to cartesian coordinates by the following equations that are explained in [10], where $\mu^{(Body)}$ denotes the Gravitational Parameter of the planet.

Use the formulas in chapter 8.1 to get the True Anomaly ν and Eccentric Anomaly E. In this calculation $\overrightarrow{o}$ and $\dot{\overrightarrow{o}}$ denote the position and velocity vector within the orbital frame of reference.

As reference frames we assume that inertial X is aligned with body X at t = 0.

$$
\begin{aligned}
||\vec{o}|| &:= a(1 - e\cdot\cos(E)) \\
\vec{o} &:= ||\vec{o}|| \begin{pmatrix} \cos(\nu) \\ \sin(\nu) \\ 0 \end{pmatrix} \\
\dot{\vec{o}} &:= \frac{\sqrt{\mu^{(Body)}a}}{||\vec{o}||} \begin{pmatrix} -\sin(E) \\ \sqrt{1-e^2}\cos(E) \\ 0 \end{pmatrix} \\
\vec{P} &:= \begin{pmatrix} \vec{o}_X(\cos(\omega)\cos(\Omega) - sin\omega\cos(i)\sin(\Omega)) - \vec{o}_Y(\sin(\omega)\cos(\Omega) + \cos(\omega)\cos(i)\sin(\Omega)) \\ \vec{o}_X(\cos(\omega)\sin(\Omega) + sin\omega\cos(i)\cos(\Omega)) - \vec{o}_Y(\cos(\omega)\cos(i)\cos(\Omega) - \sin(\omega)\sin(\Omega)) \\ \vec{o}_X(\sin(\omega)\sin(i)) + \vec{o}_Y(cos\omega\sin(i)) \end{pmatrix} && (8.13) \\
\vec{V} &:= \begin{pmatrix} \dot{\vec{o}}_X(\cos(\omega)\cos(\Omega) - sin\omega\cos(i)\sin(\Omega)) - \dot{\vec{o}}_Y(\sin(\omega)\cos(\Omega) + \cos(\omega)\cos(i)\sin(\Omega)) \\ \dot{\vec{o}}_X(\cos(\omega)\sin(\Omega) + sin\omega\cos(i)\cos(\Omega)) - \dot{\vec{o}}_Y(\cos(\omega)\cos(i)\cos(\Omega) - \sin(\omega)\sin(\Omega)) \\ \dot{\vec{o}}_X(\sin(\omega)\sin(i)) + \dot{\vec{o}}_Y(cos\omega\sin(i)) \end{pmatrix} && (8.14)
\end{aligned}
$$

8.4 KSP coordinate conversion

Depending on the location of the ship, different references for the Cartesian coordinate system are used. This chapter describes them once it is finished.

Part II

Physics in the Game

Chapter 9

In Space

9.1 Orbital Mechanics

9.1.1 Apsides

We can calculate the apoapsis r_A and periapsis r_P (in the meaning distance to the Planets center) as

$$r_A := a \cdot (1 + e) \tag{9.1}$$

$$r_P := a \cdot (1 - e) \tag{9.2}$$

Conversely we can calculate the semi-major axis and eccentricity as

$$a := \frac{1}{2} \cdot (r_A + r_P) \tag{9.3}$$

$$e := \frac{r_A}{a} - 1 \tag{9.4}$$

$$e := 1 - \frac{r_P}{a} \tag{9.5}$$

9.1.2 Orbital Velocities

The velocity V of an object that orbits a Body and has a distance r to the Planets center is according to the Vis-viva equation [31]

$$V := \sqrt{\mu^{(Body)} \cdot \left(\frac{2}{r} - \frac{1}{a}\right)} \tag{9.6}$$

It should be noted that the speed reaches it's maximum at Periapsis and minimum at Apoapsis and that the disctance r lies within the bounds

$$r_P \leq r \leq r_A$$

In the case of a circular orbit, the speed is equal at all points on the orbit.

For the calculation of the velocity at the border of the SOI the above formula can be used, but the calculation of the semi-major axis a is different.

9.1.3 Orbital Period

The orbital period (with the meaning of sidereal period [12]) around a Body is

$$SiderealPeriod := 2\pi\sqrt{\frac{a^3}{\mu^{(Body)}}} \tag{9.7}$$

9.2 Orbital Darkness Time

A ship orbiting a Body will spend some time in the shadow of this body.

The maximal possible time T_d in darkness can be approximated as described in [32].

$$b := \sqrt{r_A \cdot r_P} \text{ the semi-minor axis} \tag{9.8}$$

$$l := 2\frac{r_A \cdot r_P}{r_A + r_P} \text{ the semi-latus rectum} \tag{9.9}$$

$$h := \sqrt{l\mu^{(Name)}} \text{ the specific angular momentum} \tag{9.10}$$

$$T_d :\approx \frac{2ab}{h}\left(\arcsin\left(\frac{PRadius^{(Name)}}{b}\right) + \frac{e \cdot PRadius^{(Name)}}{b}\right) \tag{9.11}$$

Note that this is the darkness time for the case that the Apoapsis is in the center of the shadow. The time can be much lower if the Periapsis is in the shadow.

In order to calculate the longest time that a ship can be without sunlight, the other planetary bodies have to be taken into acount.

9.3 Geostationary Orbits

A geosynchronous orbit [47] is an orbit with an orbital period that equals the duration of one sidereal day.

A geostationary orbit [48] is a special case of a geosynchronous orbit with an inclination of $i = 0$ that is also circular.

To calculate the altitude of a geostationary orbit a_G we use the following approach.

$$
\begin{aligned}
RotationPeriod &= OrbitalPeriod \\
RP^{(Name)} &= 2\pi\sqrt{\frac{(a_G + PRadius^{(Name)})^3}{\mu^{(Body)}}} \\
a_G &= \sqrt[3]{\left(\frac{RP^{(Name)}}{2\pi}\right)^2 \cdot \mu^{(Body)}} - PRadius^{(Name)} \qquad (9.12)
\end{aligned}
$$

Chapter 10

Orbital Transfers

10.1 Hohmann Transfer

10.1.1 Delta-V Calculations

10.1.2 Phase Angles

10.2 Bi-elliptic Transfer

10.3 Ideal Transfer Method

Chapter 11

Engine Calculations

11.1 Thrust to Weight Ratio

The Thrust to Weight ratio TWR [39] of a ship with mass M and thrust F_T is a simple yet complex measurement.

Simple in the meaning that it can be calculated easily as

$$TWR := \frac{F_T}{M \cdot g} \tag{11.1}$$

where g denotes the gravitational acceleration.

Complex because of the choice of g. g can be selected in different ways that makes it difficult to talk about, if it is not specified what is meant. Some versions are:

- g is the gravity at Sealevel on Kerbin
- g is the local gravity at the position of the ship $LocalGravity(distance)$ as calculated in (5.2)
- g is the gravity at the Sealevel of the Body, the ship is currently orbiting

For launching a rocket from the launchpad it is necessary that $TWR > 1$. In this case g is nearly equal for all variants. Within a single stage of a ship the TWR always increases because of the decreasing fuel mass.

11.2 Delta-V

Delta-V or Δv [44] is a measurement of how much fuel is needed to reach a destination. The thread “What Is Delta V?” [45] contains answers from different points of view that try to explain it.

Using Tsiolkovsky rocket equation [37], [46] we can calculate the Delta-V a ship has, given its initial mass m_0, its dry mass m_1 and its I_{SP}.

$$\Delta v := \ln\left(\frac{m_0}{m_1}\right) \cdot I_{SP} \cdot g_0 \tag{11.2}$$

This formula is much more useful in vacuum than in the atmosphere, because I_{SP} depends on the pressure and the pressure changes with altitude within the atmosphere. And note that this formula is only correct within a single stage.

11.3 Delta-V Upper Bound

The maximal theoretical possible amount of Delta-V Δv_M within a single stage can be calculated based on the I_{SP} of the engines within this stage and an upper bound of the full fueltank : dry fueltak mass ratio. Fuel tanks in KSP have an upper bound for this ratio of 9 : 1.

$$\begin{aligned}
\Delta v_M &< \ln\left(\frac{9}{1}\right) \cdot I_{SP} \cdot g_0 \\
&= \ln(9) \cdot I_{SP} \cdot g_0 \\
&\approx \begin{cases} 4747\frac{m}{s} & \text{for LV-N Atomic Rocket Motor on Kerbins Sealevel} \\ 6904\frac{m}{s} & \text{for Mainsail on Kerbins Sealevel} \\ 7551\frac{m}{s} & \text{for Rockomax 48-7S in vacuum} \\ 8414\frac{m}{s} & \text{for LV-909 or Toroidal Aerospike Rocket in vacuum} \\ 17261\frac{m}{s} & \text{for LV-N Atomic Rocket Motor in vacuum} \end{cases}
\end{aligned} \tag{11.3}$$

Note however that these bounds are far from reality, since they do not account for Engine or payload mass and induce a ridiculously low TWR. Ships that have an average vacuum Δv of $7000\frac{m}{s}$ per stage are feasible for rockets that launch from the spaceport. [53] Todo: Is $g_0 = 9.82\frac{m}{s^2}$ correct for this calculation?

11.4 Burn Duration

Given Δv and a ship configuration as mass of the ship before the burn M, Specific impulse I_{SP} [1] and the thrust F_T it is possible to calculate the duration of the burn T_B.

Using (5.7) we can calculate the Fuelconsumption $\dot{M}$ in $\frac{kg}{s}$.

In order to calculate the total amount of fuel used M_F during the burn, we use equation (11.2).

$$\Delta v = I_{SP} \cdot g_0 \cdot \ln\left(\frac{M}{M - M_F}\right)$$

[1] Remember that the Specific impulse depends not only on the engines but also on the pressure.

so we get

$$
\begin{aligned}
M_F &= M - \frac{M}{\exp\left(\frac{\Delta v}{I_{SP} \cdot g_0}\right)} \\
&= M\left(1 - \exp\left(-\frac{\Delta v}{I_{SP} \cdot g_0}\right)\right) \qquad (11.4)
\end{aligned}
$$

and can calculate the burn time

$$
\begin{aligned}
T_B &= \frac{M_F}{\dot{M}} \\
&= \frac{M}{F_T} \cdot I_{SP} \cdot g_0 \left(1 - \exp\left(-\frac{\Delta v}{I_{SP} \cdot g_0}\right)\right) \qquad (11.5)
\end{aligned}
$$

Note however that this is only valid if the current stage in the rocket has enough Δv for the complete burn.

Chapter 12

Ascent Calculations

12.1 General notes

We use "ideal" in the meaning optimized for fuel usage in this chapter.

For a rocket the amount of fuel needed for ascending from Kerbins Sealevel to an orbit depends on several factors including

- Rocket staging configuration
- Ascent Path

Since these factors are quite variable, it should be noted that there is no single ideal ascent trajectory that fits to all rockets.

That being said, Rockets that have similar TWR-configurations have similar ideal ascent paths.

12.2 Payload Fraction

The payload fraction f_P of a rocket is the payload mass m_P divided by the mass at launch m_L.

$$f_P := \frac{m_P}{m_L} \tag{12.1}$$

We define the payload mass as the mass of the parts of the rocket that stay in orbit after launch and are not used in any way to help with the ascent. For example a engine that is ignited before reaching orbit is not considered for the payload mass.

In stock KSP payload fractions above 20% have been reached on Kerbin. [49]

12.3 Delta-V Requirements

As stated in [50] the amount Δv_L of Delta-V needed to ascend from sealevel to a stable orbit above the atmosphere can be approximated by the sum of

- Orbital Speed v_o on an orbit above the atmosphere with a radius r_o
- drag losses during a vertical ascent Δv_D
- gravity losses during a vertical ascent Δv_G

While the orbital speed can be calculated as stated in chapter 9.1.2, the other two can be approximated in the following way. [51]

$$\Delta v_D :\approx \frac{v_o}{TWR - 1} \tag{12.2}$$

$$\Delta v_G :\approx \sqrt{G \cdot PMass^{(Name)} \left(\frac{1}{PRadius^{(Name)}} - \frac{1}{r_o} \right)} \tag{12.3}$$

$$\Delta v_L :\approx v_o + \Delta v_D + \Delta v_G \tag{12.4}$$

where TWR denotes the rockets TWR at the begin of the launch at the launchsite. As a replacement for the orbital speed the escape velocity [52] can be used.

Chapter 13

Atmosphere

13.1 Terminal Velocity

The terminal velocity [54] V_t of an object in vertical motion is the velocity at which the sum of drag force and buoyancy [55] equals the gravity force. Since buoyancy is not modeled in KSP, it is left out in the following calculation.

As described in equations (5.1) and (5.3) the forces are

$$\begin{aligned}
\overrightarrow{F_D} &= -0.5 \cdot \rho(alt) \cdot d \cdot 0.008 \frac{m^2}{kg} \cdot M \cdot \left|\left|\overrightarrow{SV}\right|\right| \cdot \overrightarrow{SV} \\
\overrightarrow{F_G} &= -M \cdot \mu^{(Name)} \cdot \frac{\overrightarrow{P}}{\left|\left|\overrightarrow{P}\right|\right|^3}
\end{aligned}$$

Setting them equal results in the terminal velocity for an object with drag coefficient d at altitude alt of

$$\begin{aligned}
0.5 \cdot \rho(alt) \cdot d \cdot 0.008 \frac{m^2}{kg} \cdot M \cdot \left|\left|\overrightarrow{SV}\right|\right|^2 \cdot \frac{\overrightarrow{SV}}{\left|\left|\overrightarrow{SV}\right|\right|} &= M \cdot \mu^{(Name)} \cdot \frac{\overrightarrow{P}}{\left|\left|\overrightarrow{P}\right|\right|^3} \\
0.5 \cdot \rho(alt) \cdot d \cdot 0.008 \frac{m^2}{kg} \cdot \left|\left|\overrightarrow{SV}\right|\right|^2 &= \mu^{(Name)} \cdot \frac{1}{\left|\left|alt + PRadius^{(Name)}\right|\right|^2} \\
\left|\left|\overrightarrow{SV}\right|\right| &= \frac{\sqrt{\frac{250 \frac{kg}{m^2} \cdot \mu^{(Name)}}{\rho(alt) \cdot d}}}{\left|\left|alt + PRadius^{(Name)}\right|\right|} \\
V_t(alt, d) &:= \frac{\sqrt{\frac{250 \frac{kg}{m^2} \cdot \mu^{(Name)}}{\rho(alt) \cdot d}}}{\left|\left|alt + PRadius^{(Name)}\right|\right|} \qquad (13.1)
\end{aligned}$$

13.1.1 Kerbin

For Kerbin the Terminal velocities according to this formula are like on the Wikipage [6].

Altitude (m)	Terminal Velocity (m/s)
75	
1000	

Part III

Appendices

Chapter 14

Differences to the real world

14.1 Atmosphere

In the real world the atmosphere does not stop suddenly at an altitude.

In KSP every Planetary Body with an atmosphere has an altitude above of which there is no atmosphere and this altitude can be calculated as described in equation (3.1).

The density of the atmosphere also depends on temperature, weather and humidity [35] but KSP does not model these.

Also Buoyancy [55] is not modeled in KSP, which describes the force that an object receives by displacing a liquid or air.

14.2 Drag

Drag is directly proportional to the cross-sectional area A of the ship. In KSP this value A is approximated by the ships mass M as described in chapter 5.1.2.

With the Add-Ons FAR [20] and NEAR [21] an effort is made to implement a more realistic atmosphere into KSP.

14.3 Lift

I don't feel comfortable describing effects I don't even have tried to comprehend yet. So I will just leave this chapter at "KSP lift differs greatly from the real world" and refer the interested reader to [20].

14.4 Gravitational Model

In the real world every Planetary Body exercises gravitation to all other bodies. In KSP a simplified model is used that is based on Patched Conics [33].

This simplification states that the space is divided into areas where only the gravitation of a single planetary body is in effect. These areas are mostly spheric, have the same center as the planetary body and they can contain other such areas, if the planetary body has other planetary bodies orbiting it.

The name “Sphere of Influence” can refer either to the area itself or to it’s radius. This radius can be calculated according to the formula in [34].

One effect of this model is that one can escape the SOI of a planet with a velocity that is smaller than the theoretical escape velocity, because it is sufficient to raise the Apoapsis above the planets SOI.

The Add-On Principia [23] intends to implement a more realistic N-Body-Physics model.

14.5 Local Gravity

The local gravity depends in the real world on the density of the continent below and the form of the planetary body. So the local gravity differs for locations at the same distance to the planets center while in KSP the local gravity is constant for all locations on the same distance.

14.6 Engines

In the real world Rockete-engines do not have a constant thrust and ISP has a different meaning...

Index

Bibliography

[1] http://kerbalspaceprogram.com/

[2] https://github.com/mhoram-kerbin/ascent-optimizer

[3] http://wiki.kerbalspaceprogram.com/wiki/Atmosphere

[4] http://wiki.kerbalspaceprogram.com/wiki/API:CelestialBody

[5] http://wiki.kerbalspaceprogram.com/wiki/Main_Page

[6] http://wiki.kerbalspaceprogram.com/wiki/Kerbin

[7] http://wiki.kerbalspaceprogram.com/wiki/Specific_impulse#Conversion_factor

[8] http://en.wikipedia.org/wiki/Kepler_orbit

[9] https://downloads.rene-schwarz.com/dc/category/20

[10] https://downloads.rene-schwarz.com/dc/category/19

[11] http://wiki.kerbalspaceprogram.com/wiki/Specific_impulse#Formula

[12] http://en.wikipedia.org/wiki/Sidereal_period

[13] http://wiki.kerbalspaceprogram.com/wiki/Specific_impulse#Multiple_engines

[14] http://forum.kerbalspaceprogram.com/threads/29788-How-to-calculate-lift

[15] http://forum.kerbalspaceprogram.com/threads/67606-Fuel-consumption-as-a-function-of-atmospheric-pressure

[16] http://en.wikipedia.org/wiki/True_anomaly

[17] http://en.wikipedia.org/wiki/Mean_anomaly

[18] http://en.wikipedia.org/wiki/Eccentric_anomaly

[19] http://en.wikipedia.org/wiki/Newton%27s_method

[20] http://forum.kerbalspaceprogram.com/threads/20451-Ferram-Aerospace-Research

[21] http://forum.kerbalspaceprogram.com/threads/86419-NEAR-A-Simpler-Aerodynamics-Model

[22] http://forum.kerbalspaceprogram.com/threads/75916-If-something-has-an-SAS-torque-of-20-what-does-that-actually-mean

[23] http://forum.kerbalspaceprogram.com/threads/68502-Principia-N-Body-Gravitation

[24] http://en.wikipedia.org/wiki/Euclidean_vector

[25] http://en.wikipedia.org/wiki/Dot_product

[26] http://en.wikipedia.org/wiki/Cross_product

[27] http://en.wikipedia.org/wiki/Trigonometric_functions

[28] http://en.wikipedia.org/wiki/Derivative

[29] http://en.wikipedia.org/wiki/Exponentiation

[30] http://en.wikipedia.org/wiki/Natural_logarithm

[31] http://en.wikipedia.org/wiki/Vis-viva_equation

[32] http://wiki.kerbalspaceprogram.com/wiki/Orbit_darkness_time

[33] http://en.wikipedia.org/wiki/Patched_conic_approximation

[34] http://wiki.kerbalspaceprogram.com/wiki/Sphere_of_influence

[35] http://en.wikipedia.org/wiki/Density_of_air

[36] http://forum.kerbalspaceprogram.com/threads/36476?p=1433092#post1433092

[37] http://en.wikipedia.org/wiki/Tsiolkovsky_rocket_equation

[38] http://wiki.kerbalspaceprogram.com/wiki/Orbit#Orbits_in_the_save_file

[39] http://wiki.kerbalspaceprogram.com/wiki/Thrust-to-weight_ratio

[40] http://wiki.kerbalspaceprogram.com/wiki/Intake_air

[41] http://wiki.kerbalspaceprogram.com/wiki/Air_intake

[42] http://forum.kerbalspaceprogram.com/threads/73472-Air-intake-model-stock-KSP

[43] http://www.reddit.com/r/KerbalSpaceProgram/comments/1bx1zd/

[44] http://en.wikipedia.org/wiki/Delta-v

[45] http://forum.kerbalspaceprogram.com/threads/60189-What-Is-Delta-V

[46] http://wiki.kerbalspaceprogram.com/wiki/Cheat_sheet#Delta-v_.28.CE.94v.29

[47] http://en.wikipedia.org/wiki/Geosynchronous_orbit

[48] http://en.wikipedia.org/wiki/Geostationary_orbit

[49] http://forum.kerbalspaceprogram.com/threads/55615-Payload-Fraction-Challenge

[50] http://forum.kerbalspaceprogram.com/threads/46194-I-need-someone-help-me-do-some-math-for-launch-optimization?p=987663#post987663

[51] http://forum.kerbalspaceprogram.com/threads/97801-I-need-help-making-a-delta-v-map?p=1498127#post1498127

[52] http://en.wikipedia.org/wiki/Escape_velocity

[53] http://forum.kerbalspaceprogram.com/threads/59823-Delta-V-Maximization-Challenge

[54] http://en.wikipedia.org/wiki/Terminal_velocity

[55] http://en.wikipedia.org/wiki/Buoyancy