Virginia Polytechnic
Institute And State University

Orbital Launch Vehicle Team — Trajectory and Analysis

Critical Design Review Document

# Two–Stage Rocket Motion Simulation

*Author*
M. A. Werner

Fall – Winter 2019

CONTENTS

# LIST OF SYMBOLS

## *Matlab*

### – *Inputs* –

#### *Switches*

| | |
|---|---|
| `odestats` | Show solver statistics |
| `oderefine` | Amount of points to interpolate between time steps |
| `odereltol` | Solver relative tolerance |
| `odeabstol` | Solver absolute tolerance |
| `odemaxdt` | Solver maximum time step |
| `tableseqs` | Range of sequences to display in tables |
| `plotseqs` | Range of sequences to display in plot |
| `angleu` | Units of angles to display in plots |
| `SampleTable` | Display a short table |
| `FullTable` | Display the full table |
| `DispTable` | Display the table for a desired range of sequences |
| `Plots` | Show plots |
| `RealTime` | Show runtime |
| `ODEsTime` | Show solver integration time |
| `Stochastics` | Consider Brownian motion |
| `StochasticN` | Consider bounded limit on randomly determined stochastic variable |

#### *Parameters*

| | |
|---|---|
| `n` | Stochastic variable bound |
| `nMax` | Bound on randomly determined `n` |
| `lat` | Geodetic latitude of launch site |
| `elev` | Elevation above sea level of launch site |
| `T0` | Ground-level temperature at launch site |
| `Ltower` | Length of launch tower |
| `L` | Length of launch vehicle |
| `fn` | Fin number |
| `fw` | Fin width |
| `fz` | Fin depth |
| `od` | Outer diameter |
| `mf` | Fuel mass |
| `m0` | Initial mass |
| `tbFT` | Thrust as a function of burn time |
| `MaCd` | Drag coefficient as a function of Mach number |
| `delay` | Time to coast between staging |

| | |
|---|---|
| `diamp` | Flat parachute diameter |
| `tb` | Burn time |
| `dmdt` | Mass flow rate |
| `A` | Characteristic area |
| `Ap` | Characteristic area of parachute |
| `Lcm` | Mass center distance from base |

*Initial Conditions*

| | |
|---|---|
| `delr` | Shortest distance separating launch rail axis from vehicle axis |
| `delphi` | Vertical offset angle |
| `deltheta` | Horizontal offset angle |
| `veps` | Initial velocity |
| `t0` | Initial time |
| `tf` | Final time |
| `xx0` | Initial state |

*– Outputs –*

*Flight Characteristics and Performance*

| | |
|---|---|
| `t` | Time |
| `xx` | State |
| `te` | Event times |
| `xxe` | Event states |
| `x` | Displacement in the $x$ direction |
| `y` | Displacement in the $y$ direction |
| `z` | Displacement in the $z$ direction |
| `vx` | Velocity in the $x$ direction |
| `vy` | Velocity in the $y$ direction |
| `vz` | Velocity in the $z$ direction |
| `ax` | Acceleration in the $x$ direction |
| `ay` | Acceleration in the $y$ direction |
| `az` | Acceleration in the $z$ direction |
| `r` | Radial displacement from the origin |
| `v` | Speed relative to the observer origin |
| `a` | Acceleration relative to the observer origin |
| `FT` | Force of thrust |
| `FD` | Force of drag |
| `FG` | Force of gravity |
| `FXx` | Force of stochastic process in the $x$ direction |
| `FXy` | Force of stochastic process in the $y$ direction |
| `FXz` | Force of stochastic process in the $z$ direction |
| `vxv` | Directionality function in the $x$ direction |
| `vyv` | Directionality function in the $y$ direction |
| `vzv` | Directionality function in the $z$ direction |
| `theta` | Horizontal displacement directionality angle |
| `phi` | Vertical displacement directionality angle |
| `thetap` | Horizontal velocity directionality angle |
| `phip` | Vertical velocity directionality angle |

| | |
|---|---|
| `psi` | Angle of alignment between the displacement and velocity vectors |
| `El` | Angle of elevation |
| `thetad` | `theta` in degrees |
| `phid` | `phi` in degrees |
| `thetapd` | `thetap` in degrees |
| `phipd` | `phip` in degrees |
| `psid` | `psi` in degrees |
| `Eld` | `El` in degrees |
| `m` | Mass |
| `Cd` | Drag coefficient |
| `q` | Dynamic pressure |
| `Ma` | Mach number |
| `Re` | Reynolds number |

*Earth and the Atmosphere*

| | |
|---|---|
| `g` | Gravitational acceleration |
| `T` | Air temperature (corrected) |
| `p` | Air pressure |
| `rho` | Air density (corrected) |
| `c` | Local speed of sound |
| `mu` | Kinematic viscosity of air |
| `nu` | Dynamic viscosity of air |

*Theory*

*– Coordinate Frames –*

*Generic*

| | |
|---|---|
| $\mathcal{O}$ | Coordinate system origin |
| $\xi$ | Cartesian coordinate direction |
| $\eta$ | Cartesian coordinate direction |
| $\zeta$ | Cartesian coordinate direction |
| $w$ | Spherical coordinate magnitude |
| $\vartheta$ | Spherical coordinate angle |
| $\varphi$ | Spherical coordinate angle |

*Observer*

| | |
|---|---|
| $O$ | Observer origin located at the launch tower's base |
| $x$ | Coordinate direction emanating parallel to the ground towards the rocket's mass center |
| $y$ | Coordinate direction completing the right-hand rule |
| $z$ | Coordinate direction emanating completely upwards |
| $r$ | Radial displacement from the origin |
| $\theta$ | Horizontal displacement directionality angle |
| $\phi$ | Vertical displacement directionality angle |

*Body*

| | |
|---|---|
| $O'$ | Body origin located at the rocket's mass center |
| $x'$ | Coordinate direction aligned with $x$ |
| $y'$ | Coordinate direction aligned with $y$ |
| $z'$ | Coordinate direction aligned with $z$ |
| $v$ | Speed relative to the observer origin |
| $\theta'$ | Horizontal velocity directionality angle |
| $\phi'$ | Vertical velocity directionality angle |

*– Earth and the Atmosphere –*

| | |
|---|---|
| $\varphi$ | Geodetic latitude |
| $R_{eq}$ | Equatorial radius |
| $R_{po}$ | Polar radius |
| $R$ | Radius to sea level at latitude $\varphi$ |
| $h$ | Local elevation above sea level |
| $Z_g$ | Geopotential altitude |
| $\Phi$ | Geopotential |
| $g_\varphi$ | Nominal gravitational acceleration |
| $g$ | Gravitational acceleration |
| $T$ | Air temperature |
| $p$ | Air pressure |
| $\rho$ | Air density |
| $T_0$ | Ground-level air temperature |
| $\Delta T$ | Real and standard air temperature difference |
| $\gamma$ | Heat capacity ratio of air |
| $R$ | Gas constant of air (differentiated from Earth radius by context) |
| $c$ | Local speed of sound |
| $\mu$ | Dynamic viscosity of air |
| $\nu$ | Kinematic viscosity of air |

*– Initial Conditions –*

| | |
|---|---|
| $\ell_{cm}$ | Mass center distance from base |
| $\delta_r$ | Shortest distance separating launch rail axis from vehicle axis |
| $\delta_\phi$ | Launch angle |
| $\delta_\theta$ | Offset angle of the launch rail from the $x$ axis |

*– Launch Vehicle –*

| | |
|---|---|
| $m$ | Mass |
| $\dot{m}$ | Mass flow rate |
| $m_0$ | Initial mass |
| $L$ | Characteristic length |
| $A$ | Characteristic area |
| $C_d$ | Drag coefficient |
| $u$ | Relative exhaust velocity |

*– Flight Characteristics –*

| | |
|---|---|
| $t$ | Time |
| $\mathbf{x}$ | State containing the position ($\vec{r}$) and velocity ($\vec{v}$) |
| $F_t$ | Force of thrust |
| $F_d$ | Force of drag |
| $F_g$ | Force of gravity |
| $F_\chi$ | Force of stochastic processes |
| $\psi$ | Angle of alignment between the displacement and velocity vectors |
| El | Angle of elevation |
| Ma | Mach number |
| Re | Reynolds number |
| $q$ | Dynamic pressure |
| $\Delta v$ | Velocity change |

*– Descent Systems –*

| | |
|---|---|
| $r_d$ | Flat parachute radius |
| $A_d$ | Flat parachute area |
| $r_p$ | Inflated parachute radius |
| $A_p$ | Inflated parachute characteristic area |
| $C_{d,p}$ | Parachute drag coefficient |
| $\tau_p$ | Half-time to deploy parachute |
| $\tau_p^*$ | Time at which parachute deploys |
| $z_m^*$ | Main parachute deployment altitude |

# 1. USER GUIDE

## 1.1 Introduction and General Overview

This Matlab program "`rtraj`", was written for the Orbital Launch Vehicle Team by a member of the Trajectory and Analysis subteam. It aims to provide a means of comparing trajectory solutions to those obtained from commercially-available software and enable the ability to calculate additional flight parameters that are often otherwise unobtainable from commercial solutions.

The model considers three driving forces – gravitation, thrust, and drag – and a small-scale stochastic force representing random thrust and atmospheric effects on the vehicle while ascending at a rate in excess of the speed the sound. The magnitude of this stochastic force depends on a user-input ratio of the gravitational force experienced at local altitude, including zero, and is further scaled by the density of air at altitude relative to the local launch site elevation relative to sea level.

The equations of motion are solved numerically with variable step size and order via Adams-Bashforth-Moulton methods. The solution is split into seven regions, referred to as sequences, corresponding to critical regimes of flight wherein flight parameters and characteristics are smooth. Particularly, flight parameters spanning across sequences are most generally only piecewise continuous and possibly piecewise smooth with the exception of nonzero stochastic forcing, which is discontinuous everywhere outside of a set of zero measure.

The previously mentioned sequences are specified in the following order.

1 The first stage is firing and the rocket is on the launch rail
2 The first stage is firing and the rocket is clear of the launch rail, i.e. in free-flight
3 The rocket is coasting with the first stage depleted but still attached
4 The second stage is firing with the first stage now detached
5 The second stage is depleted and the rocket is coasting to apogee
6 The rocket is descending from apogee with its drogue parachute until main deployment
7 The rocket is descending with its main parachute until touchdown

The identifier "0" is specially reserved for the initial conditions, which are represented only by a singular point in time, thus a flight regime of zero measure and therefore not considered a sequence.

Once the solution is fully obtained, the flight characteristics and parameters are cloned using the final, converged state variables. Because of its nature, the stochastic force can only be estimated at best by algebraic means after a finite difference computation to estimate acceleration. These estimates can, therefore, be quite rough, but nonetheless provide a means of obtaining the general magnitudes of what the stochastic forcing could have been. If the stochastic force is known to be zero otherwise, the acceleration may be directly obtained through recalculation of the experienced forces.

## 1.2 User Inputs

### 1.2.1 Switches

The beginning of `rtraj` is dedicated to variable inputs. The first block acts as a collection of switches that indicate how the program should behave when performing the numerical integration and displaying its progress and results. These options and switches are listed below, where the pair $(i, j)$ are integers satisfying $1 \leq i \leq j \leq 7$, and the values $x, i$, and $j$ per option are independent from one another.

| | | |
|---|---|---|
| `odestats` | 'on'/'off' | Provide a short summary of statistics from each ODE solver |
| `oderefine` | $x \in \mathbb{Z}^+$ | Specify the amount of points evaluated between time steps |
| `odereltol` | $x \in (0, 1)$ | Specify the relative accuracy of numerically calculated solution |

| | | |
|---|---|---|
| `odeabstol` | $x \in (0, 1)$ | Specify the absolute accuracy of numerically calculated solution |
| `odemaxdt` | $x \in \mathbb{R}^+$ | Specify the maximum allowable time step by solver |
| `tableseqs` | $i{:}j$ | Specify range of sequences to display in `DispTable` (↓) |
| `plotseqs` | $i{:}j$ | Specify range of sequences to display in `Plots` (↓) |
| `angleu` | 'rad'/'deg' | Specify units of angles to display in `Plots` (↓) |
| `SampleTable` | true/false | Show a short table of the solution and parameters |
| `FullTable` | true/false | Show entire table of solution and parameters |
| `DispTable` | true/false | Show table of solution and parameters via `tableseqs` (↑) |
| `Plots` | true/false | Show plots of the solution and parameters via `plotseqs` (↑) |
| `RealTime` | true/false | Show the runtime |
| `ODEsTime` | true/false | Show the integration time at each time step |
| `Stochastics` | true/false | Apply stochastic forcing |
| `StochasticN` | true/false | Permit the maximum value of `n` (↓) to be random |

### *1.2.2 Vehicle and Launch Site Parameters*

The random variable in the stochastic force represents a percentage of the current vehicle weight by which the force can be scaled at each step. The value `n` (corresponding with `Stochastics`) determines a hard static bound on values that this random variable can attain at each time step. Furthermore, the value `stochasticNMax` (corresponding with `StochasticN`) provides a hard static bound as to what the hard bound on the random variable can be at each step. This feature is only used if and only if `Stochastics` is already true. Otherwise, the variable `n` is set to zero, eliminating any consideration of the stochastic force.

| | | |
|---|---|---|
| `n` | $\lvert x\rvert < 1$ | Maximum value of random variable per time step |
| `nMax` | $\lvert x\rvert < 1$ | Maximum value of randomly determined `n` per time step |

The parameters determining characteristics about the launch site are then listed. Particular values that specify the site characteristics are its geodetic latitude and elevation, which determine the local atmospheric conditions and radius of Earth for the gravitational force.

| | | | |
|---|---|---|---|
| `lat` | $\lvert x\rvert \leq 90$ | [deg] | Geodetic latitude of launch site |
| `elev` | $-86 < x < 6200$ | [m] | Elevation above sea level of launch site |
| `T0` | $x > -459.67$ | [°F] | Local temperature at launch site |
| `T00` | `atmos(elev, lat, 0)` | [K] | Temperature via atmosphere model `atmos` |
| `delT` | `convtemp(T0, 'F', 'K') - T00` | [K] | Temperature offset from `atmos` model |
| `rho0` | `atmos(elev, lat, delT)` | [kg/m$^3$] | Local surface air density via `atmos` |
| `Ltower` | $x > 0$ | [m] | Overall length of launch rail |

It is noted that the values `T00`, `delT`, and `rho0` follow from function definitions dependent upon previously input data and are not explicitly required to be manually assigned a numerical value. However, because `T0` is only used to define `delT`, the standard atmosphere conditions may be used if `delT` is set to zero.

The parameters of the vehicle itself are defined next. These parameters are defined for each of the two stages, where the first entry corresponds to the first stage and the second entry for the second stage. The notation [US] → [SI] indicates that the parameters are input in (Standard) units [US] and immediately converted into (metric) units [SI] during runtime. Additionally, the notation [SI] [SI] indicates that the input is related to a matrix with row size greater than unity for which the units apply to its second dimension, i.e. to the individual columns ($n \times 2$ with $n > 1$ in this case).

| | | | |
|---|---|---|---|
| `L` | $\begin{bmatrix}6.35 & 3.71\end{bmatrix}$ | [m] | Overall length of the vehicle |
| `fn` | $\begin{bmatrix}4 & 4\end{bmatrix}$ | [ ] | Number of fins considered for drag |

| | | | |
|---|---|---|---|
| `fw` | $\begin{bmatrix}9.25 & 6.75\end{bmatrix}$ | [in] → [m] | Fin width |
| `fz` | $\begin{bmatrix}0.435 & 0.44\end{bmatrix}$ | [in] → [m] | Fin thickness |
| `od` | $\begin{bmatrix}8.5 & 6\end{bmatrix}$ | [in] → [m] | Largest outer diameter of casing |
| `mf` | $\begin{bmatrix}71.22 & 20.374\end{bmatrix}$ | [kg] | Mass of fuel at the beginning of each stage |
| `m0` | $\begin{bmatrix}194.43 & 65.33\end{bmatrix}$ | [kg] | Initial mass at the beginning of each stage |
| `tbFT` | $\mathbf{F} \in \mathbb{R}^{n_{1,2}\times 2\times 2}$ | [s] [lbf] → [N] | Thrust as a function of burn time from csv files |
| `MaCd` | $\mathbf{F} \in \mathbb{R}^{n_{1,2}\times 2\times 2}$ | [ ] [ ] | Drag coefficient as a function of Mach number from csv files |
| `delay` | $x \geq 0$ | [s] | Time between first stage burnout and second stage firing |
| `diamp` | $\begin{bmatrix}36 & 108\end{bmatrix}$ | [in] → [m] | Flattened parachute diameters |
| `zm` | $x \geq 0$ | [m] | Altitude at which the second stage's parachute deploys |

It is important that the comma separated values (csv) files are supplied to consider variable thrust (obtained from *BurnSim*) and drag coefficients (obtained from *RasAero*). For thrust, any trailing times for which thrust is constantly zero should be removed, and the drag coefficient should have a sufficient Mach number domain so that drag coefficients are not extrapolated. Should thrust or drag coefficient be desired to be held constant, the respective files need only two rows of information. The values `tbFT` and `MaCd` are combined to form a $1 \times 2$ cell of $1 \times 2$ cells that each contain two matrices, each of which have 2 columns but different numbers of rows, which is passed into the solver in place of passing both `tbFT` and `MaCd`.

| | | | |
|---|---|---|---|
| `csvs` | $\{\{$`tbFT{1}  MaCd{1}`$\}$ $\{$`tbFT{2}  MaCd{2}`$\}\}$ | [s] [N] [ ] [ ] $\times 2$ | Combined csv files |

Several additional parameters may be computed using these above-defined values. The notation of Matlab functions and element-wise operations are used below.

| | | | |
|---|---|---|---|
| `tb` | `tbFT{:}(end,1)` | [s] | Burn time (note: pseudo-notation) |
| `dmdt` | `-mf./tb` | [kg/s] | Mass flow rate |
| `A` | `pi/4*od.^2+fw.*fz.*fn` | [m$^2$] | Characteristic rocket cross-sectional areas |
| `Ap` | `pi/8*diamp.^2` | [m$^2$] | Characteristic drogue and main parachute areas |

The final section of parameters are those related to the initial conditions of launch. As such, the initial conditions are evaluated with a fully loaded, unfired launch vehicle attached to the launch rail at time $t = 0$.

| | | | |
|---|---|---|---|
| `Lcm` | 2.2 | [m] | Distance from the first stage nozzle exit to the mass center |
| `delr` | $x \geq 0$ | [in] → [m] | Perp. distance from the launch rail center to the vehicle center |
| `delphi` | $x \in [0, 90]$ | [deg] | Offset angle of the launch rail from the true $z$-vertical |
| `deltheta` | $x \in [0, 360]$ | [deg] | Offset angle of the launch rail from the $x$-horizontal |
| `veps` | `eps` | [m/s] | Very small initial velocity |

All of these parameters are categorized by tables (`iWant` (3), `stochasticParams` (2), `launchSiteParams` (4), `rocketParams` (15), and `initParams` (5) respectively in order of previous appearance) and combined into the structure `args` to be passed into the solver. This action is done as opposed to passing individual values to the solver due to the structure having the distinct advantages of organization and flexibility.

The last values `t0`, `tf`, and `xx0` are not passed into the solvers per se, but rather used as arguments. The initial time `t0` specifies the initial integration time and is fixed to zero. Accordingly, the final time `tf` is the maximum allowable integration time per sequence. To guarantee sufficient time then, an arbitrarily large value can be chosen since the solvers are split into sequences and stop automatically upon sequence changes. Finally, `xx0` is vector of initial conditions corresponding to `t0`. It is noteworthy to mention that `t0` may be nonzero, but every value must be updated accordingly to compensate for beginning the solution in mid-flight.

The values left completely to user interpretation are truly open and have no correlation with the physical build of the rocket, but particular values are given below.

| | Black Rock Desert (Nevada) | Spaceport America (New Mexico) |
|---|---|---|
| `lat` | 40.9107°N | 32.9903°N |
| `elev` | 1191 [m] | 1401 [m] |
| `Ltower` | 360 [in] | 360 [in] |
| `delay` | 7 [s] | 7 [s] |
| `zm` | 609 [m] | 609 [m] |
| `delr` | 7.25 [in] | 7.25 [in] |
| `delphi` | 5° | 5° |
| `deltheta` | 10° | 10° |

In reality, the values `delr`, `delphi`, and `deltheta` are measurable values to be found at the launch site, and `delphi` will be close to zero, but is set to consider a worst-case scenario.

## *1.3 Assumptions*

The implicit assumptions made within `rtraj` are explicitly listed here.

The atmosphere model is taken to be the 1976 U.S. Standard Atmosphere defined by the World Meteorological Organization that varies in latitude from Earth's surface up to 1000 km.

The gravitational acceleration varies along with the radius of Earth as a function of latitude.

The amount of thrust is sufficient to lift the specified mass `m0` so that the rocket is able to sustain vertical motion overcoming the influence of gravity.

The thrust force acts in the direction of the velocity vector, antiparallel to the drag force.

The mass flow rate is a constant while staging and coasting

The stochastic force is only considered at supersonic speeds.

The rocket is inherently stable.

The second stage fires immediately following separation.

The rocket's horizontal displacement is not sufficiently large enough to account for the curvature of Earth's surface nor changes in latitude and local elevation.

The rocket deploys the drogue parachute at apogee in which drag increases as $\operatorname{erf} t$ over a span of 4 seconds and releases the main chute at the specified altitude `zm`, increasing drag similarly over a span of 7 seconds.

The parachutes take on a hemispherical form when fully deployed and exhibit a constant drag coefficient.

The main parachute deployment altitude `zm`, while positive, is less than the attained altitude at apogee.

## *1.4 Warnings*

A smooth run of `rtraj` is not guaranteed for all possible input values, and obtaining a solution does not necessarily mean that the solution is converged. The convergence of a solution is determined by `odereltol`

and `odeabstol`. These values increase solution accuracy as each decrease at the cost of speed. Additionally, solution resolution is determined by `oderefine` and `odemaxdt` in which resolution increases as `oderefine` increases and `odemaxdt` decreases at the cost of speed. If considering stochastic forcing, using any value greater than one for `oderefine` creates artificial, possibly incorrect, trends in acceleration.

There is no case handling for flights in which the set parameters result in touchdown before the final sequence, or more generally, any future sequence that happens during the current one. The program may therefore either break in the solver or finish executing. Thus if execution finished, special attention must be given to the altitude, for any negative altitude indicates that the simulation failed.

# 2. DERIVATION

## 2.1 Generalized Coordinates and Mappings

Consider a point in a non-rotating orthogonal coordinate system at a specified time $t$. The trajectory history of this point as $t$ varies can be written in both spherical and Cartesian coordinates. In spherical coordinates, this history can be established by tracking the vector $\vec{w} \in \mathbb{R}^3$ by its magnitude $0 \leq w < \infty$ and associated directions $0 \leq \vartheta < 2\pi$ and $0 \leq \varphi \leq \pi$ for all times $t$ in the associated time interval of tracking.

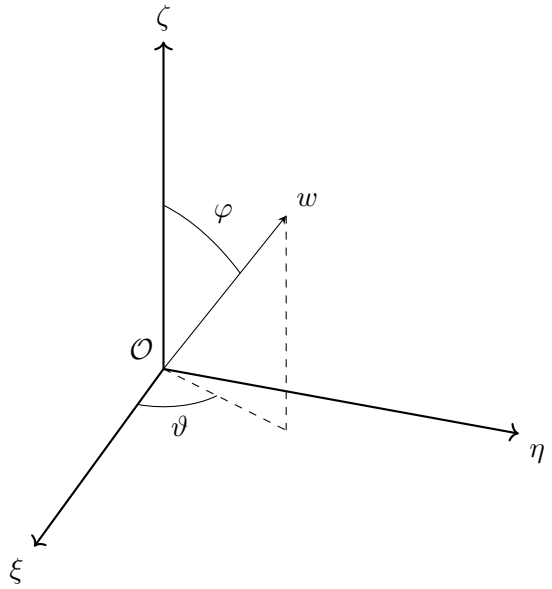


*Fig. 2.1.1:* Spherical coordinate system containing a point at $(w, \vartheta, \varphi)$ at time $t$

The mapping from spherical to Cartesian coordinates follows from the projection of $\mathbb{R}^3 \to \mathbb{R}^2 \to \mathbb{R}$ onto the coordinate axes $\xi, \eta, \zeta$ as

$$\xi = w \sin\varphi \cos\vartheta \tag{2.1.1}$$
$$\eta = w \sin\varphi \sin\vartheta \tag{2.1.2}$$
$$\zeta = w \cos\varphi. \tag{2.1.3}$$

Conversely, the inverse mapping is

$$w^2 = \xi^2 + \eta^2 + \zeta^2 \tag{2.1.4}$$
$$\tan\vartheta = \frac{\eta}{\xi} \tag{2.1.5}$$
$$\cos\varphi = \frac{\zeta}{w}. \tag{2.1.6}$$

In general, the point of interest will have a radius $r$ from an inertial frame $O$ and a velocity $v$ in the non-inertial body frame $O'$. This frame $\mathcal{O} \to O$ takes $\xi \to x$, $\eta \to y$, and $\zeta \to z$, and is related to $O'$ ($\xi \to x'$, $\eta \to y'$, $\zeta \to z'$) by

$$O' = O + \begin{bmatrix} x & y & z \end{bmatrix}^T, \tag{2.1.7}$$

where $O$ is the observer location, which can be taken to be zero, and $O'$ is non-rotating with respect to the $O$ frame. Finally, it is noted that all vectors $\vec{w}$ are free such that $\vec{r}$ can be expressed in the O′ frame beside $\vec{v}$ and there exists an angle of separation that determines how aligned $\vec{r}$ is with $\vec{v}$ and vice versa.

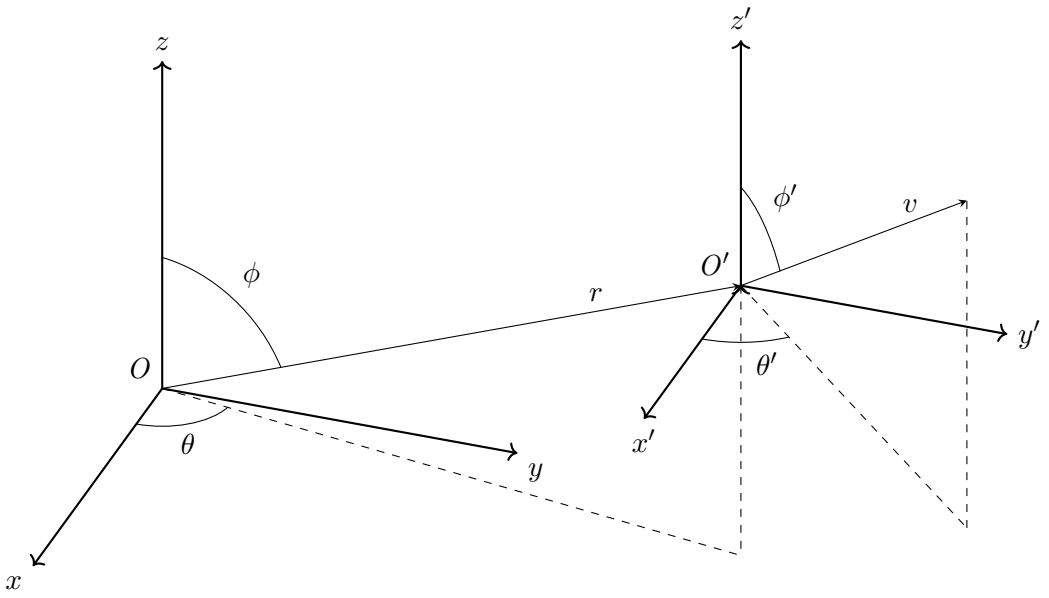


*Fig. 2.1.2:* Relation between $O$ and $O'$ coordinate systems

## *2.2 Application To Rocket Motion*

The main, deterministic forces acting on the point, which will be considered as the rocket's center of mass, in the $O'$ frame are those of thrust $\vec{F}_t$, drag $\vec{F}_d$, and gravity $\vec{F}_g$. Because the drag force always acts in opposition to the velocity vector $\vec{v}$, which by assumption shares the same direction as the thrust force, the forces of thrust and drag act in opposing directions at all times except when either or both forces go to zero, in which case directionality is not defined. A perturbitive stochastic force $\vec{F}_\chi$ is also included to account for the Brownian motion of fluid particles in close proximity to the vehicle which can be defined to model random thrusting or atmosphere effects.

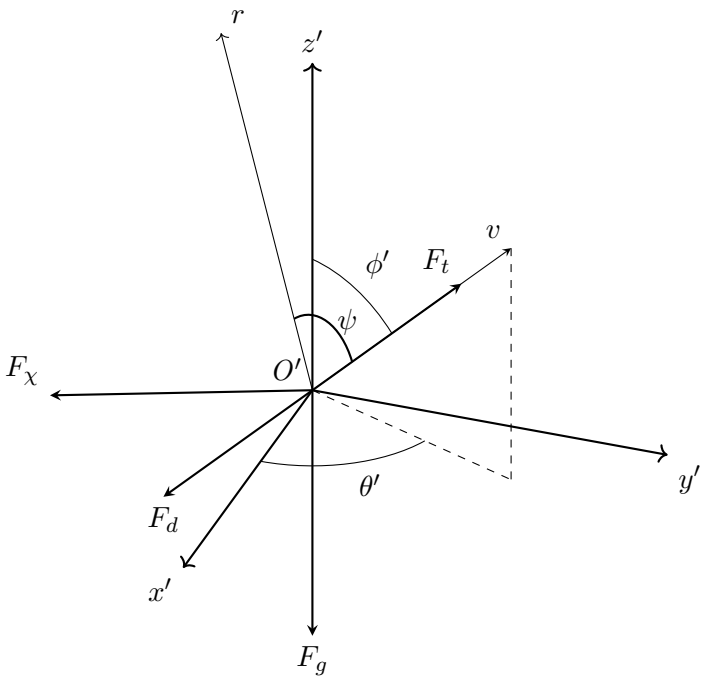


*Fig. 2.2.1:* Free-body diagram of the rocket's center of mass while in flight

With only information regarding the magnitude of the involved forces, the governing equations of motion to be solved for $\vec{r}$ may be expressed as

$$m\ddot{x} = (F_t - F_d)\sin\phi'\cos\theta' + F_{\chi_x} \quad [\text{N}] \tag{2.2.1}$$
$$m\ddot{y} = (F_t - F_d)\sin\phi'\sin\theta' + F_{\chi_y} \quad [\text{N}] \tag{2.2.2}$$
$$m\ddot{z} = (F_t - F_d)\cos\phi' - F_g + F_{\chi_z} \quad [\text{N}], \tag{2.2.3}$$

where every variable is implicitly a function of time $t$ and the symbol $\dot{}$ denotes a time derivative. The angles $\phi'$ and $\theta'$ can be rewritten in terms of $\vec{v}$ from Fig. (2.2.1) as

$$\sin\phi' = \frac{\sqrt{\dot{x}^2 + \dot{y}^2}}{v} \qquad \cos\theta' = \frac{\dot{x}}{\sqrt{\dot{x}^2 + \dot{y}^2}} \tag{2.2.4}$$
$$\cos\phi' = \frac{\dot{z}}{v} \qquad \sin\theta' = \frac{\dot{y}}{\sqrt{\dot{x}^2 + \dot{y}^2}}. \tag{2.2.5}$$

As a corollary, the angles $\phi$ and $\theta$ in the $O$ frame must follow the same form since the orientation of each frame is the same and are written

$$\sin\phi = \frac{\sqrt{x^2 + y^2}}{r} \qquad \cos\theta = \frac{x}{\sqrt{x^2 + y^2}} \tag{2.2.6}$$
$$\cos\phi = \frac{z}{r} \qquad \sin\theta = \frac{y}{\sqrt{x^2 + y^2}}, \tag{2.2.7}$$

where the displacement and velocity vectors are formally and respectively defined

$$\vec{r} = \begin{bmatrix} x & y & z \end{bmatrix}^T \quad [\text{m}] \tag{2.2.8}$$
$$\vec{v} = \dot{\vec{r}} \quad [\text{m/s}]. \tag{2.2.9}$$

The angle $\psi$ measures the deviation of the velocity vector $\vec{v}$ from the displacement vector $\vec{r}$ and is obtained for all times $t$ through direct use of the inner and cross products as

$$\tan\psi = \frac{\|\vec{r} \times \vec{v}\|}{\vec{r}^T\vec{v}}, \tag{2.2.10}$$

or in terms of the coordinate angles $\phi, \theta, \phi', \theta'$,

$$\tan\psi = \frac{\sqrt{\sin^2\phi\cos^2\phi' + (\cos^2\phi + \sin^2(\theta - \theta')\sin^2\phi)\sin^2\phi' - (\cos(\theta - \theta')\sin 2\phi \sin 2\phi')/2}}{\cos\phi\cos\phi' + \cos(\theta - \theta')\sin\phi\sin\phi'}. \tag{2.2.11}$$

If launching solely for altitude (complete vertical ascent), $\psi$ is identically zero until apogee when $\dot{z} \sim 0$, after which $\vec{r}$ and $\vec{v}$ are antiparallel. Thus, with the assumption of purely vertical motion, i.e. $\phi \equiv 0$, $\psi$ is a step function for all times $t$ behaving like

$$\psi = \pi H(t - t^*) \quad [\text{rad}], \tag{2.2.12}$$

where $t^*$ is the time at apogee uniquely satisfying

$$\dot{z}(t^*) = 0 \quad [\text{m/s}]. \tag{2.2.13}$$

Conversely, if $\phi \sim 0$ ($\phi' \sim 0$ consequently), an assumption that holds well at least until apogee for near-vertical launches, then (2.2.11) is approximated by

$$\tan\psi \sim \left|\tan\phi' - \phi\,\sec^2\phi'\cos\big(\theta - \theta'\big)\right|. \tag{2.2.14}$$

Here, the quantity $\tan\psi$ is seen to be sensitive to $\tan\phi'$ for near vertical launches where (2.2.11) is linearly expanded in favor of $\phi$. So a rocket launching with a nonzero angle $\phi'$ should generally expect its position and velocity vectors to be misaligned. However, there are indeed particular combinations of $\phi, \theta, \phi', \theta'$ with $\phi \sim 0$ and $\phi' \neq 0$ that still satisfy $\psi = 0$, a less desirable case to complete vertical ascent.

For completion, (2.2.11) is approximated by (2.2.14) but without absolute value during descent of a near vertical launch for which $\phi \sim 0$ and $\phi' \sim \pi$.

## 2.3 Quantifying the Equations of Motion

The nozzle for each flight sequence will either have no mass flow or be choked while producing thrust, so the mass flow rate $\dot{m} \leq 0$ during each sequence can be considered constant so that the mass of the rocket is a linear function of time,

$$m = \dot{m}t + m_0 \quad [\text{kg}], \tag{2.3.1}$$

where $m_0$ is the rocket's mass at the beginning of the sequence. Additionally, the forces experienced during flight are assumed to take the forms

$$F_t = -u\dot{m} \quad [\text{N}] \tag{2.3.2}$$

$$F_d = \frac{1}{2}\rho v^2 C_d A \quad [\text{N}] \tag{2.3.3}$$

$$F_g = mg \quad [\text{N}] \tag{2.3.4}$$

$$F_{\chi_{n,\Theta}} = \frac{\rho}{\rho_0}\chi_{n,\Theta}F_g \quad [\text{N}], \tag{2.3.5}$$

where

| | | |
|---|---|---|
| $u$ | [m/s] | Speed of exhaust relative to the rocket |
| $\dot{m}$ | [kg/s] | Mass flow rate |
| $\rho$ | [kg/m$^3$] | Atmospheric density at altitude $z$ |
| $\rho_0$ | [kg/m$^3$] | Atmospheric density at launch site elevation |
| $C_d$ | [ ] | Coefficient of drag |
| $A$ | [m$^2$] | Maximum cross-sectional area during a sequence |
| $g$ | [m/s$^2$] | Gravitational acceleration |
| $\chi_{n,\Theta}$ | [ ] | Uniformly distributed number on $[-n, n]$ for $\lvert n\rvert < 1$ applied to axis $\Theta = x, y, z$ |
| $n$ | [ ] | Scaling factor of the stochastic force as a percentage of the gravitational force. |

It is noted that the thrust force $F_t$ may be given as an explicit function of time, the stochastic force $F_{\chi_{n,\Theta}}$ may be revised to take on a different form later, and the value $n$ can be interpreted many ways (including being a uniformly distributed number at each time $t$ itself). Therefore, inserting these variables with the exception of $F_t$ and $F_\chi$ into the equations of motion (2.2.1 – 2.2.3) and expressing them in terms of zeroth, first, and second order derivatives of the coordinate directions $x, y, z$ yields

$$\ddot{x} = \frac{1}{\dot{m}t + m_0}\left(F_t - \frac{\rho(\dot{x}^2 + \dot{y}^2 + \dot{z}^2)}{2}C_d A\right)\frac{\dot{x}}{\sqrt{\dot{x}^2 + \dot{y}^2 + \dot{z}^2}} + \frac{F_{\chi_{n,x}}}{\dot{m}t + m_0} \quad [\text{m/s}^2] \tag{2.3.6}$$

$$\ddot{y} = \frac{1}{\dot{m}t + m_0}\left(F_t - \frac{\rho(\dot{x}^2 + \dot{y}^2 + \dot{z}^2)}{2}C_d A\right)\frac{\dot{y}}{\sqrt{\dot{x}^2 + \dot{y}^2 + \dot{z}^2}} + \frac{F_{\chi_{n,y}}}{\dot{m}t + m_0} \quad [\text{m/s}^2] \tag{2.3.7}$$

$$\ddot{z} = \frac{1}{\dot{m}t + m_0}\left(F_t - \frac{\rho(\dot{x}^2 + \dot{y}^2 + \dot{z}^2)}{2}C_d A\right)\frac{\dot{z}}{\sqrt{\dot{x}^2 + \dot{y}^2 + \dot{z}^2}} - g + \frac{F_{\chi_{n,z}}}{\dot{m}t + m_0} \quad [\text{m/s}^2]. \tag{2.3.8}$$

Explicitly, the air density $\rho$ depends on altitude $z$, the coefficient of drag $C_d$ depends on Mach number Ma for a fixed body geometry, and the characteristic area $A$ depends solely on time $t$ for the same body geometry. These equations, from which the position $\vec{r}$ must be found, in their full forms are classified as nonautonomous, nonlinear, coupled, second-order ordinary differential equations.

### 2.3.1 Initial Conditions

With the equations of motion established, the attention now turns towards studying the initial conditions. It is well-established that a second order ordinary differential equation requires two initial conditions, namely

an initial position and initial velocity. A total of six initial conditions are thus sought as a result of carrying a set of three second order ordinary differential equations.

Consider the rocket statically attached to the launch rail before launch. Suppose the ground is slightly uneven so there exists a nonzero angle between the launch rail and the true vertical $z$. Let the coordinate system $O$ be located at the booster's nozzle exit oriented with the $z$ axis aligned truly upwards, $x$ perpendicular to $z$ and in the direction of the rocket's center of mass, and $y$ satisfying the right-hand rule. Consider the case in which the launch rail is misaligned with the vertical. In general, such a case moves the initial position of the center of mass and gives nonzero initial acceleration components in the $x$ and $y$ directions. Fig. (2.3.1) displays the special two-dimensional case when no $y$-displacement is given.

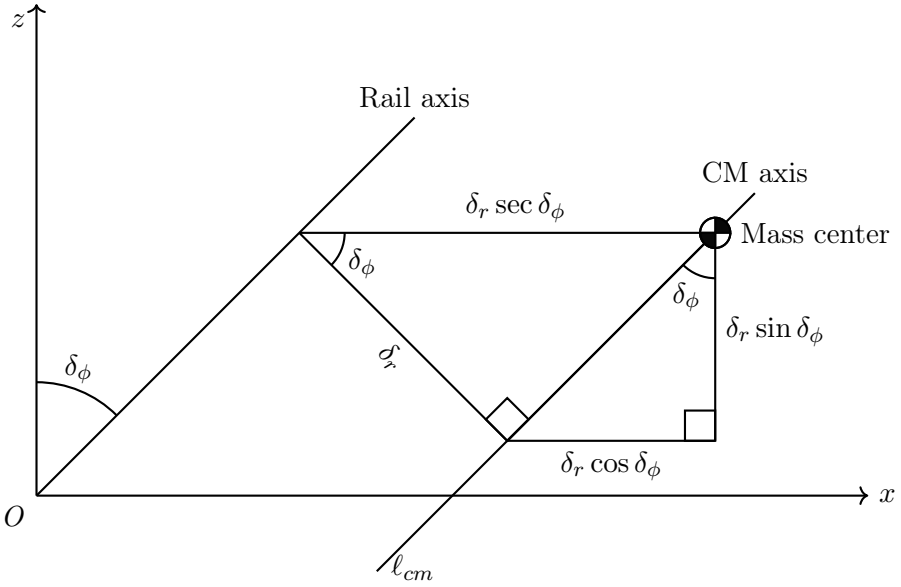


*Fig. 2.3.1:* Geometric diagram to examine how the initial conditions vary with exaggerated nonzero $\delta_\phi$

The variables appearing, as well as the special-case, non-appearing angle $\delta_\theta$ made with the $x$ axis, are all strictly measurable values and defined as

| | | |
|---|---|---|
| $\ell_{cm}$ | [m] | Distance from first stage nozzle exit to the vehicle's center of mass |
| $\delta_r$ | [m] | Perpendicular distance between launch rail and mass center axes |
| $\delta_\phi$ | [rad] | Offset angle of launch rail with the true vertical $z$ |
| $\delta_\theta$ | [rad] | Offset angle of launch rail with the horizontal relative to the $x$ axis |

The problem of finding the new mass center location in the O frame can be solved geometrically by Fig. (2.3.1) with a simple extension to include $y$-displacement through the angle $\delta_\theta$. Alternatively, the problem may be solved analytically by posing it as one of vector transformation. The solution is uniquely determined and independent of the method of solution in either case.

To consider the transformation problem, suppose first that a Cartesian frame with origin $\widetilde{\mathrm{O}}$ has its $\tilde{z}$ axis oriented along the launch rail. Now consider the O frame such that $\mathrm{O} = \widetilde{\mathrm{O}}$ but $z$ and $x$ are separated from $\tilde{z}$ and the $\tilde{x}$-$\tilde{y}$ plane respectively by an angle $\delta_\phi$. Further suppose that the $\tilde{x}$ axis is separated by an angle $\delta_\theta$ from the projection of the $x$ axis onto the $\tilde{x}$-$\tilde{y}$ plane. Then the position of the mass center in the O frame is related to the position of the mass center in the $\widetilde{\mathrm{O}}$ frame by

$$\vec{r}_{cm,\mathrm{O}} = \mathbf{R}_{\tilde{z}}(\delta_\theta)\mathbf{R}_{\tilde{y}}(\delta_\phi)\vec{r}_{cm,\widetilde{\mathrm{O}}} \tag{2.3.9}$$

$$= \begin{bmatrix} \cos\delta_\theta & -\sin\delta_\theta & 0 \\ \sin\delta_\theta & \cos\delta_\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\delta_\phi & 0 & \sin\delta_\phi \\ 0 & 1 & 0 \\ -\sin\delta_\phi & 0 & \cos\delta_\phi \end{bmatrix} \begin{bmatrix} \delta_r \\ 0 \\ \ell_{cm} \end{bmatrix} \tag{2.3.10}$$

$$= \begin{bmatrix} \big(\ell_{cm}\sin\delta_\phi + \delta_r\cos\delta_\phi\big)\cos\delta_\theta \\ \big(\ell_{cm}\sin\delta_\phi + \delta_r\cos\delta_\phi\big)\sin\delta_\theta \\ \ell_{cm}\cos\delta_\phi - \delta_r\sin\delta_\phi \end{bmatrix} \quad [\mathrm{m}]. \tag{2.3.11}$$

Furthermore, if the launch vehicle is prescribed a small initial velocity $v_\epsilon$ such that the equations of motion are nonsingular for times near $t = 0$, then a local solution only might exist for times $t > 0$ until touchdown. It is therefore advantageous to convert the three, second-order ordinary differential equations into six, first-order ordinary differential equations to place them in convenient form to study the existence and uniqueness of any potential solutions. If solutions do exist and are unique, then this form is also the desired form for obtaining numerical solutions. With this motivation, let the vehicle's state be defined as

$$\mathbf{x} := \begin{bmatrix} x & y & z & \dot{x} & \dot{y} & \dot{z} \end{bmatrix}^T \tag{2.3.12}$$

such that $\dot{\mathbf{x}} = \boldsymbol{f}(\mathbf{x}, t)$. The equations of motion are then written

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \frac{\left(F_t - \frac{1}{2}\rho(\dot{x}^2+\dot{y}^2+\dot{z}^2)C_dA\right)\dot{x}}{(\dot{m}t+m_0)\sqrt{\dot{x}^2+\dot{y}^2+\dot{z}^2}} + \frac{F_{\chi n,x}}{\dot{m}t+m_0} \\ \frac{\left(F_t - \frac{1}{2}\rho(\dot{x}^2+\dot{y}^2+\dot{z}^2)C_dA\right)\dot{y}}{(\dot{m}t+m_0)\sqrt{\dot{x}^2+\dot{y}^2+\dot{z}^2}} + \frac{F_{\chi n,y}}{\dot{m}t+m_0} \\ \frac{\left(F_t - \frac{1}{2}\rho(\dot{x}^2+\dot{y}^2+\dot{z}^2)C_dA\right)\dot{z}}{(\dot{m}t+m_0)\sqrt{\dot{x}^2+\dot{y}^2+\dot{z}^2}} - g + \frac{F_{\chi n,z}}{\dot{m}t+m_0} \end{bmatrix}, \qquad \mathbf{x}(0) = \begin{bmatrix} \big(\ell_{\mathrm{cm}}\sin\delta_\phi + \delta_r\cos\delta_\phi\big)\cos\delta_\theta \\ \big(\ell_{\mathrm{cm}}\sin\delta_\phi + \delta_r\cos\delta_\phi\big)\sin\delta_\theta \\ \ell_{\mathrm{cm}}\cos\delta_\phi - \delta_r\sin\delta_\phi \\ v_\epsilon\sin\delta_\phi\cos\delta_\theta \\ v_\epsilon\sin\delta_\phi\sin\delta_\theta \\ v_\epsilon\cos\delta_\phi \end{bmatrix}. \tag{2.3.13}$$

## 2.4 Earth and Atmospheric Models

The Earth is assumed to take on the shape of a spheroid with major and minor axes equivalent to the equatorial and polar radii ($R_{eq}$ and $R_{po}$) respectively. As such, the geocentric radius to the (smooth) surface at a geodetic latitude $\varphi$ is given by

$$R = \sqrt{\frac{R_{eq}^4\cos^2\varphi + R_{po}^4\sin^2\varphi}{R_{eq}^2\cos^2\varphi + R_{po}^2\sin^2\varphi}} \quad [\mathrm{m}]. \tag{2.4.1}$$

Local changes in elevation relative to sea level are accounted for in the altitude as to avoid false positives between sea level and ground level. In other words, the altitude $z$ references zero at the local geographic elevation $h$ such that the rocket is a distance $R + h + z$ from the Earth's center, i.e. $h + z$ from the surface (sea level).

Earth's atmosphere is modelled using the World Meteorological Organization's U.S. 1976 Standard Atmosphere extended to 1000 km. The theoretical gravitational acceleration at a sea-level geodetic latitude $\varphi$ is

$$g_\varphi \sim 9.780356\,(1 + 0.0052885\sin^2\varphi - 0.0000059\sin^2 2\varphi) \quad [\mathrm{m/s^2}] \tag{2.4.2}$$

such that the gravitational acceleration is

$$g = \frac{R^2}{(R+h+z)^2}\,g_\varphi \quad [\mathrm{m/s^2}]. \tag{2.4.3}$$

Also, the geopotential altitude is

$$Z_g = \frac{R\,(h+z)}{R+h+z} \quad [\mathrm{m}], \tag{2.4.4}$$

which determines the atmospheric regime from which to take values for the density, pressure, and temperature. These values for $Z_g \leq 84.852\,\text{km}$ compose the U.S. 1976 Standard Atmosphere and are tabulated in functional form in Tab. (2.4.1).

*Tab. 2.4.1:* Temperatures and pressures according to the 1976 U.S. Standard Atmosphere as a function of geopotential altitude

| Geop. Alt. $Z_g$ [km] | Temperature $T$ [K] | Pressure $p$ [Pa] |
|---|---|---|
| $0-11$ | $288.15-6.5Z_g$ | $101325.0\left(\frac{288.15}{288.15-6.5Z_g}\right)^{-34.1632/6.5}$ |
| $11-20$ | $216.65+0.0Z_g$ | $22632.06\,e^{-34.1632(Z_g-11)/216.65}$ |
| $20-32$ | $196.65+1.0Z_g$ | $5474.889\left(\frac{216.65}{(216.65+(Z_g-20)}\right)^{34.1632}$ |
| $32-47$ | $139.05+2.8Z_g$ | $868.0187\left(\frac{228.65}{228.65+2.8(Z_g-32)}\right)^{34.1632/2.8}$ |
| $47-51$ | $270.65+0.0Z_g$ | $110.9063\,e^{-34.1632(Z_g-47)/270.65}$ |
| $51-71$ | $413.45-2.8Z_g$ | $66.93887\left(\frac{270.65}{270.65-2.8(Z_g-51)}\right)^{-34.1632/2.8}$ |
| $71-84.852$ | $356.65-2.0Z_g$ | $3.956420\left(\frac{214.65}{214.65-2(Z_g-71)}\right)^{-34.1632/2}$ |

Under this model, air is assumed to be a perfect gas with gas constant $R = 287.057$ [J/kg K], understood not to be the Earth's spheroidal radius by context, with heat capacity ratio $\gamma = 1.4$ such that the air density within these geopotential altitudes follows the ideal gas law.

$$\rho = \frac{p}{RT} \quad [\text{kg/m}^3] \tag{2.4.5}$$

The extended atmosphere in Tabs. (2.4.2 – 2.4.4) are used for higher geopotential altitudes, though for these values, the geocentric altitude $z$ is used instead. The values are listed below in in order of previous appearance (temperature, pressure, density).

*Tab. 2.4.2:* Temperatures according to the extended U.S. 1976 Standard Atmosphere

| Alt. $z$ [km] | Temperature $T$ [K] |
|---|---|
| $86-91$ | 186.8673 |
| $91-110$ | $263.1905-76.3232\sqrt{1+\left(\frac{z-91}{19.9429}\right)^2}$ |
| $110-120$ | $240+12(z-110)$ |
| $120-1000$ | $1000-640\,e^{-0.01875(z-120)\frac{6356.766+120}{6356.766+z}}$ |

The expression for air pressure is an exponential power law of the form

$$p = e^{A_p z^4+B_p z^3+C_p z^2+D_p z+E_p} \quad [\text{Pa}], \tag{2.4.6}$$

whose coefficients are given below in Tab. (2.4.3).

*Tab. 2.4.3:* Pressure exponential power law coefficients for geocentric altitudes within the extended U.S. 1976 Standard Atmosphere

| Alt. $z$ [km] | $10^{11} \cdot A_p$ [km$^{-4}$] | $10^7 \cdot B_p$ [km$^{-3}$] | $10^4 \cdot C_p$ [km$^{-2}$] | $D_p$ [km$^{-1}$] | $E_p$ [ ] |
|---|---|---|---|---|---|
| $86-91$ | 0.000000 | 21.59582 | $-4.836957$ | $-0.1425192$ | 13.47530 |
| $91-100$ | 0.000000 | 330.4895 | $-90.62730$ | 0.6516698 | $-11.03037$ |
| $100-110$ | 0.000000 | 0.6693926 | $-194.5388$ | 1.719080 | $-47.75030$ |
| $110-120$ | 0.000000 | $-0.6539316$ | 248.5568 | $-3.223620$ | 135.9355 |
| $120-150$ | 22835.06 | $-1343.221$ | 299.9016 | $-3.055446$ | 113.5764 |
| $150-200$ | 1209.434 | $-96.92458$ | 30.02041 | $-0.4523015$ | 19.19151 |
| $200-300$ | 81.13942 | $-9.822568$ | 4.687616 | $-0.1231710$ | 3.067409 |
| $300-500$ | 9.814674 | $-1.654439$ | 1.148115 | $-0.05431334$ | $-2.011365$ |
| $500-750$ | $-7.835161$ | 1.964589 | $-1.657213$ | 0.04305869 | $-14.77132$ |
| $750-1000$ | 2.813255 | $-1.120689$ | 1.695568 | $-0.1188941$ | 14.56718 |

Similarly, the air density follows the identical form

$$\rho = e^{A_\rho z^4 + B_\rho z^3 + C_\rho z^2 + D_\rho z + E_\rho} \quad [\mathrm{kg/m^3}], \tag{2.4.7}$$

whose coefficients are given below in Tab. (2.4.4).

*Tab. 2.4.4:* Density exponential power law coefficients for geocentric altitudes within the extended U.S. 1976 Standard Atmosphere

| Alt. $z$ [km] | $10^{12} \cdot A_\rho$ [km$^{-4}$] | $10^9 \cdot B_\rho$ [km$^{-3}$] | $10^6 \cdot C_\rho$ [km$^{-2}$] | $D_\rho$ [km$^{-1}$] | $E_\rho$ [ ] |
|---|---|---|---|---|---|
| $86-91$ | 0.000000 | $-3322622$ | 911.1460 | $-0.2609971$ | 5.944694 |
| $91-100$ | 0.000000 | 28734.05 | $-8492.037$ | 0.6541179 | $-23.62010$ |
| $100-110$ | $-12407740$ | 5162063 | $-804834.2$ | 55.55996 | $-1443.338$ |
| $110-120$ | 0.00000 | $-88541.64$ | 33732.54 | $-4.390837$ | 176.5294 |
| $120-150$ | 366177.1 | $-215434.4$ | 48092.14 | $-4.884744$ | 172.3597 |
| $150-200$ | 19060.32 | $-15277.99$ | 4724.294 | $-0.6992340$ | 20.50921 |
| $200-300$ | 1199.282 | $-1451.051$ | 691.0474 | $-0.1736220$ | $-5.321644$ |
| $300-500$ | 114.0564 | $-213.0756$ | 157.0762 | $-0.07029296$ | $-12.89844$ |
| $500-750$ | 8.105631 | $-2.358417$ | $-2.635110$ | $-0.01562608$ | $-20.02246$ |
| $750-1000$ | $-3.701195$ | $-8.608611$ | 51.18829 | $-0.06600998$ | $-6.137674$ |

The standard atmosphere can be shifted in temperature and density to account for real offsets from the model. If $T_0$ is the real temperature at ground-level and $T_{0,0}$ is that predicted by the model, then the temperature correction is $\Delta T = T_0 - T_{0,0}$ such that the temperature for all altitudes then becomes

$$T_c = T + \Delta T \quad [\mathrm{K}]. \tag{2.4.8}$$

Accordingly, the density follows (2.4.5) and is written

$$\rho_c = \frac{p}{RT_c} \quad [\mathrm{kg/m^3}]. \tag{2.4.9}$$

The subscript notation for the corrected temperature and density, however, is dropped and subsequently are respectively designated as temperature $T$ and density $\rho$ for simplicity. Furthermore, the specific potential at altitude $z$ provides a measure of the physical, as opposed to chemical, potential energy contained within the rocket and is given by

$$\Phi = 9.80665\, Z_g \quad [\mathrm{m^2/s^2}]. \tag{2.4.10}$$

## 2.5 Parachute Model

Suppose a circular parachute exhibits a radius $r_d$ when flattened and an average radius $r_p$ when fully unfurled and inflated during descent. Further suppose that the fabric is rigid enough that negligibly small amounts of stretching occur when met with varying air resistance. Let $S_p$ be the surface area of the hemisphere induced by the inflated parachute and $A_p$ be the characteristic area enclosed by the dashed curve in Fig. (2.5.1).

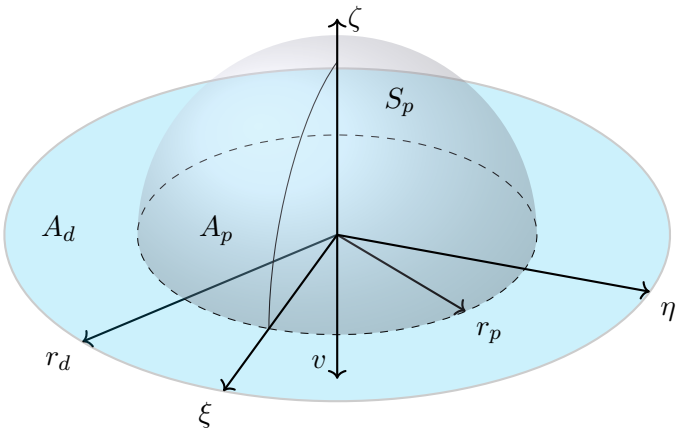


*Fig. 2.5.1:* Depiction of flattened and inflated parachute modeled as a disk and hemisphere

With a fabric that has negligible stretching, any geometric transformation yields a surface area invariant. In particular, the area of the disk $A_d$ and surface area of the hemisphere $S_p$ are identically the same. Consequently,

$$\frac{r_p}{r_d} = \frac{1}{\sqrt{2}} \iff \frac{A_p}{A_d} = \frac{1}{2}. \tag{2.5.1}$$

Thus, the characteristic area associated with drag during flight is related to the measurable parachute area when unfurled and uninflated as

$$A_p = \frac{\pi}{2} r_d^2 \quad [\mathrm{m}^2]. \tag{2.5.2}$$

Both the drogue and main parachutes fit within this framework and are rated with a drag coefficient of $C_{d,p} = 0.97$.

The drogue parachute is first deployed as the rocket reaches apogee and begins its descent to the surface. The main parachute deployment is delayed until the rocket reenters the atmosphere to maximize the usage of atmospheric braking while approaching terminal velocity, simultaneously reducing the amount of stress to be handled by the shock cords. If the main chute deployment altitude is $z_m^* = \text{const.} < z(t^*)$, then the times $\tau_p^*$ at which the parachutes are deployed are defined

$$\tau_p^* = \begin{cases} t^* & \text{(drogue)} \\ z^{-1}(z_m^*) & \text{(main for } t > t^*\text{)} \end{cases} \quad [\mathrm{s}]. \tag{2.5.3}$$

The deployment dynamics of the characteristic areas and drag coefficients for the drogue and main parachutes are assumed to take on the form

$$A = A_p \operatorname{erf}\left(\frac{t - \tau_p^*}{\tau_p}\right) \quad [\mathrm{m}^2] \tag{2.5.4}$$

$$C_d = C_{d,p} \operatorname{erf}\left(\frac{t - \tau_p^*}{\tau_p}\right) \quad [\,], \tag{2.5.5}$$

which are added to the characteristic area and drag coefficient of the vehicle itself, where

$$\operatorname{erf}(t) = \frac{2}{\sqrt{\pi}} \int_0^t e^{-s^2} \, \mathrm{d}s \tag{2.5.6}$$

is the error function and

$$\tau_p = \begin{cases} 2 & \text{(drogue)} \\ 3.5 & \text{(main)} \end{cases} \quad \text{[s]}. \tag{2.5.7}$$

These forms were chosen to simulate the drogue and main parachutes opening in approximately $2\tau_p$ seconds (the error function is odd, monotonically increasing, and $\operatorname{erf}(2) \sim 0.9953$) respectively without overcomplicating the deployment dynamics.

# 3. ANALYSIS

## 3.1 Solution Existence and Uniqueness

Studying the existence and uniqueness of a potential solution relies upon the Picard-Lindelhöf theorem which states that a unique local $C^0$ solution exists if $\boldsymbol{f}(\mathbf{x},t)$ is uniformly Lipschitz in $\mathbf{x}$ and continuous in $t$. The term uniformly Lipschitz indicates that a function is bounded in its first derivative.

The state equation contains $t$ explicitly only in the mass term, which is bounded but continuous at all points except when staging. Thus, if a solution exists, then it must solely belong to times before or after first stage separation. Now let $V^2_{\pm\Theta} := v^2 \pm v^2_\Theta$ and $F^\Theta_\chi$ denote $F_{\chi_n,\Theta}$ for convenience, where $\Theta = x, y, z$, such that the gradient of $\boldsymbol{f}(\mathbf{x},t)$ without mass terms (i.e. $m = 1$), denoted $\boldsymbol{f}_{\not m}(\mathbf{x})$, is written

$$\frac{\partial \boldsymbol{f}_{\not m}}{\partial \mathbf{x}} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ \frac{\partial F^x_\chi}{\partial x} & \frac{\partial F^x_\chi}{\partial y} & \frac{\partial F^x_\chi}{\partial z} - \frac{\rho'(z)}{\rho(z)}\frac{\dot{x}}{v}F_d & \frac{\partial F^x_\chi}{\partial \dot{x}} + \frac{F_t V^2_{-x} - V^2_x F_d}{v^3} & \frac{\partial F^x_\chi}{\partial \dot{y}} - \frac{\dot{x}\dot{y}(F_t+F_d)}{v^3} & \frac{\partial F^x_\chi}{\partial \dot{z}} - \frac{\dot{x}\dot{z}(F_t+F_d)}{v^3} \\ \frac{\partial F^y_\chi}{\partial x} & \frac{\partial F^y_\chi}{\partial y} & \frac{\partial F^y_\chi}{\partial z} - \frac{\rho'(z)}{\rho(z)}\frac{\dot{y}}{v}F_d & \frac{\partial F^y_\chi}{\partial \dot{x}} - \frac{\dot{x}\dot{y}(F_t+F_d)}{v^3} & \frac{\partial F^y_\chi}{\partial \dot{y}} + \frac{F_t V^2_{-y} - V^2_y F_d}{v^3} & \frac{\partial F^y_\chi}{\partial \dot{z}} - \frac{\dot{y}\dot{z}(F_d+F_t)}{v^3} \\ \frac{\partial F^z_\chi}{\partial x} & \frac{\partial F^z_\chi}{\partial y} & \frac{\partial F^z_\chi}{\partial z} - \frac{\rho'(z)}{\rho(z)}\frac{\dot{z}}{v}F_d - \frac{2g}{R+h+z} & \frac{\partial F^z_\chi}{\partial \dot{x}} - \frac{\dot{x}\dot{z}(F_t+F_d)}{v^3} & \frac{\partial F^z_\chi}{\partial \dot{y}} - \frac{\dot{y}\dot{z}(F_t+F_d)}{v^3} & \frac{\partial F^z_\chi}{\partial \dot{z}} + \frac{F_t V^2_{-z} - V^2_z F_d}{v^3} \end{bmatrix}. \tag{3.1.1}$$

Performing this operation in the classical sense is meaningless due to the everywhere-discontinuity of the stochastic force, but under a weak derivative interpretation,

$$\frac{\partial \boldsymbol{f}_{\not m}}{\partial \mathbf{x}} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & -\frac{\rho'(z)}{\rho(z)}\frac{\dot{x}}{v}F_d & \frac{F_t V^2_{-x} - V^2_x F_d}{v^3} & -\frac{\dot{x}\dot{y}(F_t+F_d)}{v^3} & -\frac{\dot{x}\dot{z}(F_t+F_d)}{v^3} \\ 0 & 0 & -\frac{\rho'(z)}{\rho(z)}\frac{\dot{y}}{v}F_d & -\frac{\dot{x}\dot{y}(F_t+F_d)}{v^3} & \frac{F_t V^2_{-y} - V^2_y F_d}{v^3} & -\frac{\dot{y}\dot{z}(F_t+F_d)}{v^3} \\ 0 & 0 & -\frac{\rho'(z)}{\rho(z)}\frac{\dot{z}}{v}F_d - \frac{2g}{R+h+z} & -\frac{\dot{x}\dot{z}(F_t+F_d)}{v^3} & -\frac{\dot{y}\dot{z}(F_t+F_d)}{v^3} & \frac{F_t V^2_{-z} - V^2_z F_d}{v^3} \end{bmatrix}. \tag{3.1.2}$$

Thus, for all values $\mathbf{x}$ with $v > 0$, the Jacobian satisfies

$$||\nabla \boldsymbol{f}_{\not m}(\mathbf{x})||_{\max} < \infty. \tag{3.1.3}$$

This inequality holds since $\rho(z) > 0$ and is bounded, $\rho'(z)$, though piecewise continuous, is bounded since $\rho$ is bounded, and $R + h + z > 0$. Also, the velocity components, and therefore drag, are bounded given a finite amount of thrust.

Consequently, $\boldsymbol{f}(\mathbf{x},t)$ is uniformly Lipschitz for $v > 0$ and continuous in $t$ for flight regimes before and after stage separation. So by Picard-Lindelhöf, there exists a smooth, unique solution in each of the flight regimes. These regimes are from launch until stage separation, from stage separation until apogee, and from apogee until landing, for which, at all times, there is a finite amount of thrust and nonzero velocity.

## 3.2 Numerical Methods

The solution $\mathbf{x}(t)$ is obtained numerically in *Matlab* and is done so using Adams-Bashforth-Moulton methods if the system is not stiff and a modified Rosenbrock formula of order 2 otherwise, where the stiffness of the system is determined by the operator based on observation of propagation speed. In most applications, however, the system is generally not stiff if stochastic forcing is zero. Otherwise, the system may be stiff depending on whether the requested error tolerances are too small or not.

During numerical integration, the values for thrust (if thrusting) and coefficient of drag are interpolated using cubic Hermite polynomials. This interpolation method was chosen due to its low-oscillation behavior to avoid gross overestimation and underestimation in regions undergoing large changes in small amounts of times, particularly near the beginning and ending of a staging sequence.

The solvers are split into seven regions in which each solver has a stopping criterion. The values to be output by the solver and those that come from computations are preallocated for memory in $7 \times 1$ cells. The values of time and the state as each stopping criterion is met are exported from the solver into the according base variables. These values are the event states and are used to extract information regarding: launch rail exit velocity, burnout altitudes, changes in velocity due to each stage and any delay between stages, altitude at apogee, descent rates, estimates of the recovery location (at least within a radial vicinity), etc., and the times at which these events occur.

## *3.3 Data Acquisition*

The numerical solution contains seven components of information evaluated at converged time steps in the ordinary differential equation solver. All flight characteristics and parameters, with the exception of nonzero stochastic forcing, may be exactly recalculated with these converged values at every step.

The evaluated times $t$ and the Cartesian position $\vec{r}$ and velocity $\vec{v}$ vectors coming directly from the solution $\mathbf{x}$ are direct outputs of the solver. The radial distance and speed relative to the origin O follow from taking the Euclidean norm of $\vec{r}$ and $\vec{v}$ respectively. The five angles defined across the O and O′ frames, $\phi, \theta, \phi', \theta'$, and $\psi$, fall from (2.2.4 — 2.2.7) and (2.2.10). Additionally, the angle $\phi$ can be converted into local elevation via

$$\mathrm{El} = \frac{\pi}{2} - \phi \quad [\mathrm{rad}]. \tag{3.3.1}$$

The vehicle mass $m$ is obtained from (2.3.1) and the main forces $F_t, F_d$, and $F_g$ according to their definitions (2.3.2 – 2.3.4).

The ratios modulating the balance of thrust and drag are defined from (2.1.1 – 2.1.3) as

$$\frac{v_x}{v} = \sin\phi' \cos\theta' \qquad \frac{v_y}{v} = \sin\phi' \sin\theta' \qquad \frac{v_z}{v} = \cos\phi'. \tag{3.3.2}$$

Some caution must be taken with these terms, however; they are actually constant during sequence 1 since the launch rail has a fixed orientation to which the rocket must oblige. Though after the launch rail is cleared, they may be taken to be truly as $v_{x,y,z}/v$ respectively.

The atmospheric model provides the evaluation of the gravitational acceleration and air properties - temperature $T$, pressure $p$, and density $\rho$. The dynamic pressure on the vehicle is then

$$q = \frac{1}{2}\rho v^2 \quad [\mathrm{Pa}]. \tag{3.3.3}$$

The local speed of sound is also found by computing

$$c = \sqrt{\gamma R T} \quad [\mathrm{m/s}], \tag{3.3.4}$$

where $\gamma = 1.4$ is the ratio of specific heats for air and $R = 287.057$ [J/kg K] is the gas constant for air, understood not to be Earth's radius from context. Additionally, the dynamic and kinematic air viscosities obey Sutherland's law such that

$$\mu = \mu_r \left(\frac{T}{T_r}\right)^{3/2} \frac{T_r + 110.4}{T + 110.4} \quad [\mathrm{Pa\,s}] \tag{3.3.5}$$

$$\nu = \frac{\mu}{\rho} \quad [\mathrm{m^2/s}], \tag{3.3.6}$$

where $\mu_r = 1.1716 \cdot 10^{-5}$ [Pa s] and $T_r = 273.15$ [K] are known reference values. Together, the Mach and Reynolds numbers are defined

$$\mathrm{Ma} = \frac{v}{c} \quad [\,] \tag{3.3.7}$$

$$\mathrm{Re} = \frac{vL}{\nu} \quad [\,], \tag{3.3.8}$$

where the reference length $L$ used in the Reynolds number is the overall length of the rocket during each sequence. The coefficient of drag may be obtained by smooth interpolation of the provided data embedded within `MaCd` or algebraically solved from (2.3.3).

The acceleration then can be directly computed if the stochastic force is zero from (2.3.6 – 2.3.8). Otherwise, the acceleration is estimated by finite difference as

$$\vec{a} \sim \frac{\Delta \vec{v}}{\Delta t} \quad [\mathrm{m/s^2}], \tag{3.3.9}$$

and the stochastic force components are estimated by their algebraic solutions to (2.3.6 – 2.3.8). From these values, the change in velocity over an interval of time $[t_1, t_2]$ during flight is

$$\Delta \vec{v}_{[t_1,t_2]} = \int_{t_1}^{t_2} \vec{a}(t)\,\mathrm{d}t \quad [\mathrm{m/s}] \tag{3.3.10}$$

such that

$$\Delta v_{[t_1,t_2]} = \left\| \Delta \vec{v}_{[t_1,t_2]} \right\| \quad [\mathrm{m/s}]. \tag{3.3.11}$$

The stochastic force is not considered in the expression (3.3.10) because it is assumed to be discontinuous everywhere, so its contribution at all points is nothing since every point is defined on a set of zero measure.

These values may finally be used in finding the state variables and flight parameters experienced by the vehicle occurring at the special times for which events outlined in §3.2 take place.

## *3.4 Solution Comparison*

The solution $\mathbf{x}(t) = \begin{bmatrix} \vec{r}^{\,T}(t) & \vec{v}^{\,T}(t) \end{bmatrix}^T$ may be compared with others obtained from various trajectory programs, most notably *RasAero*, at any and every time $t$ to determine a general sense of accuracy and agreement with different models. The modes for comparison make use of relative difference and various function norm metrics.

The metric of relative difference is useful in determining absolute quantities and thus works well in comparing particular values attained throughout the flight such as maximum Mach number, apogee altitude, time of flight, etc. If $x$ is such a comparable quantity from an accepted solution and $y$ the corresponding quantity generated by `rtraj`, where $x$ and $y$ are sufficiently far away from zero and understood by context to be not necessarily components of $\vec{r}$, then this metric is defined

$$d(x, y) = \frac{|x - y|}{\max(x, y)}. \tag{3.4.1}$$

A relative difference, as opposed to a relative error, is considered as a metric since an error measures the residual with respect to a true and accepted answer. Each and every available model, however, is both unique and does not exactly match reality, so comparing solutions in the sense of error is of little meaning.

The same metric in functional form is useful for measuring the relative difference between the two functions at every point in time. Similarly as before, if $\xi$ is a comparable function from an accepted solution and $\eta$ the corresponding function generated by `rtraj`, then the relative difference at each time $t$ is

$$\delta(\xi, \eta; t) = \frac{|\xi(t) - \eta(t)|}{1 - \min(0, \min(\min \xi, \min \eta)) + \max(\xi(t), \eta(t))}. \tag{3.4.2}$$

This function is chosen in such a way that $\delta(\xi, \eta; t) \leq |\xi(t) - \eta(t)|$ for all times $t$, so relative differences are bounded even when either $\xi$ or $\eta$ vanish nonsimultaneously. The effect of this boundedness is achieved via a positive vertical shift to both functions by an amount $1 - \min(0, \min(\min \xi, \min \eta)) = \text{const.}$ at the cost of changing the relative scaling particularly between small numbers.

The measurement of the accumulation of squared relative differences of the zeroth, first, and second orders to time $t$ against the maximum relative difference of unity to a reference time $\tau$ is then defined

$$\Psi(t; \tau) = \frac{1}{\tau} \int_0^t |\delta(\xi, \eta; s)|^2 + |\delta(\dot{\xi}, \dot{\eta}; s)|^2 + |\delta(\ddot{\xi}, \ddot{\eta}; s)|^2 \, \mathrm{d}s \,. \tag{3.4.3}$$

As such, the relative difference accumulation may be interpreted several ways depending on the chosen reference time. A cumulative function norm parameterized by $\tau$ is further obtained as

$$\varsigma(t; \tau) = \sqrt{\Psi(t; \tau)}. \tag{3.4.4}$$

Particular values of $\tau$ yield different measurements and meanings of this cumulative norm. Choosing $\tau = 1$ gives a naturally defined norm, but the relative differences are left unscaled and perhaps difficult to interpret. In contrast, the value $\tau = t$ gives a scaling to the total possible amount of accumulated relative difference until time $t$. This action has the effect of actively accounting for increases in measurement performance, thus giving ability to the norm to decrease. Using $\tau = t_1$ scales earlier relative difference accumulation to the total possible amount and may thus be interpreted as a sort of percentage of total accumulated relative difference.