Politecnico di Milano

European Space Agency

Evolution and hazard analysis of orbital fragmentation continua

Doctoral dissertation of
Stefan J. Frey

Supervisors
Prof. Camilla Colombo
Stijn Lemmens
Dr Holger Krag

Academic year 2019/20 – Cycle XXXII

Doctoral dissertation of	Stefan J. Frey
Supervisors	Prof. Camilla Colombo Stijn Lemmens Dr Holger Krag
Coordinator doctoral program	Prof. Pierangelo Masarati
External reviewers	Prof. Darren McKnight Prof. Alexander Wittig

Frey, S. (2020), *Evolution and hazard analysis of orbital fragmentation continua*, Doctoral thesis, Politecnico di Milano, Supervisors: Colombo, C., Lemmens, S. and Krag., H.

Abstract

Today, space debris constitutes a palpable threat to satellites and astronauts alike. Owing to their large numbers and high orbital velocities, collisions between space objects, having devastating consequences, have become probable. Collisions produce clouds of fragments, with many of the particles too small to be observed or tracked regularly. Even a small fragment, sized in the order of centimetres, carries the kinetic energy equivalent to a medium sized car travelling at 100 km/h. The addition of new particles increases the collision risk further, potentially triggering follow-up collisions and eventually rendering the space environment around Earth unsafe for exploration and exploitation. Guidelines to mitigate the space debris problem were published at the beginning of this millennium, however, with meagre results. If the ramifications of a fragmentation can be accurately understood and quantified, the orbits of new missions could be designed to minimise the footprint on the space environment. To estimate the ramifications of a breakup, modelling of the evolution of the fragment cloud, and its interactions with other space objects, are required.

Here, such a model is proposed. Instead of considering each fragment in the cloud individually, the cloud is modelled fully probabilistically. Its evolution is predicted through application of the continuity equation, solved numerically along the characteristics of the system. At each epoch of interest, a Gaussian mixture surrogate model is fitted to the characteristics to obtain a quick to be evaluated approximation of the true fragment density across the whole domain, removing the need of performing many trajectory integrations. This novel approach is computationally efficient and allows the removal of many restrictive assumptions, on orbital geometries and force models, that plague existing evolutionary models. The proposed method is proven to be useful in various dimensions, making it applicable to estimate the short-, mid- and long-term evolution of fragmentation continua.

From the surrogate model, the spatial density and the number of impacts – and ultimately the collision probability – can be derived for target objects on any orbit. The proposed approach is sensitive enough to assess even tiny

collision probabilities. As such, the method, implemented in a new software suite, *Starling*, is uniquely equipped to estimate the global hazard emanating from evolving fragmentation clouds on all the current and future objects in orbit. The suite enables the study of the cloud evolution in a variety of target spaces, further bolstering the understanding of its evolution.

To showcase the potential of the method it is applied to model the evolution of clouds stemming from both, explosions and collisions, considering various orbital regimes and force models. The proposed method is benchmarked against traditional Monte Carlo sampling, which it outperforms both in terms of computational efficiency and accuracy. The application of the method is not limited to the evolution of fragmentation clouds. Its generic structure makes *Starling* a general partial differential solver, applicable to any problem that can be solved with the continuity equation.

Acknowledgements

First and foremost, I would like to express my gratitude to my supervisor Prof. Camilla Colombo. You made this doctoral thesis possible, provided me with your insights, guided me with patience and enthusiasm, and created an environment of trust and friendship, allowing me to grow both as a researcher and personally.

I would like to thank my second supervisor Stijn Lemmens, whose expertise and unsparing observations greatly improved my research. I admire your curiosity and ability to scrutinise every aspect of a solution simply to advance it. My gratitude goes to Dr Holger Krag, whose interest in my research topic and foresight for its potential application are highly motivational. I would also like to thank Dr Francesca Letizia, whose inputs and advice contributed further to the success of this thesis.

Many thanks go to the two reviewers, Prof. Darren McKnight and Prof. Alexander Wittig, who provided valuable suggestions and feedback. Thanks to you, the work gained considerably in consistency and rigour, and I gained the certainty that at least two people read this thesis in its entirety.

I want to thank the members of my teams both at the Politecnico di Milano, and the European Space Agency, for spending countless hours over excellent or dreadful coffees discussing about anything and everything. I also thank the staff of both organisations for their professional support.

I am grateful for the unconditional support of my parents and brothers. From you I have learned to have an open mind and see any new challenge as an opportunity to grow and gain new experiences. Thank you Noella for your many encouragements and your endless patience. I cannot wait to finally live in the same city as you.

The doctoral research presented in this thesis received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme as part of the COMPASS project (Grant agreement No 679086), and from the European Space Agency (ESA) through the Networking/Partnering Initiative (NPI) agreement.

Contents

List of Tables

List of Figures

Abbreviations

3BP	Third Body Perturbation.
API	Application Programming Interface.
BFGS	Broyden–Fletcher–Goldfarb–Shanno.
CG	Conjugate Gradient.
CIRA	COSPAR International Reference Atmosphere.
COSPAR	Committee on Space Research.
CPU	Central Processing Unit.
CSV	Comma-Separated Values.
DTM	Drag Temperature Model.
EM	Expectation Maximisation.
ESA	European Space Agency.
GEO	Geostationary Orbit.
GL	Gauss-Legendre.
GMM	Gaussian Mixture Model.
GP	Gaussian Process.
GTO	Geostationary Transfer Orbit.
IADC	Inter-Agency Space Debris Coordination Committee.
JSON	JavaScript Object Notation.
KDE	Kernel Density Estimation.
KH	King-Hele.
LEO	Low Earth Orbit.
MCMC	Markov Chain Monte Carlo.
NASA	National Aeronautics and Space Administration.
NN	Nearest Neighbour.

NRLMSISE	Naval Research Laboratory Mass Spectrometer, Incoherent Scatter Radar Extended.
PlanODyn	Planetary Orbital Dynamics.
RBF	Radial Basis Function.
SBM	Standard Breakup Model.
SI	International System of Units.
SI-KH	Superimposed King-Hele.
SRP	Solar Radiation Pressure.

Nomenclature

A	Cross-sectional area [m^2].
A/m	Area to mass ratio [m^2/kg].
B	Ballistic coefficient, $B = c_D A/m$ [m^2/kg].
$\mathcal{C}$	Cost function.
C	Number of characteristics [$-$].
D	Phase space dimension [$-$].
E	Eccentric anomaly [rad].
$\mathcal{E}$	Specific mechanical energy [$\mathrm{m}^2/\mathrm{s}^2$].
$\boldsymbol{F}$	Dynamics.
F	Solar flux [sfu].
$\boldsymbol{H}$	Hessian.
H	Scale height [m].
$\boldsymbol{I}$	Identity matrix.
$\mathcal{I}$	Inertial frame.
I_n	Modified Bessel function of the first kind.
$\boldsymbol{J}$	Jacobian.
J_2	Second-order zonal harmonic [$-$].
K	Number of kernels [$-$].
L	Fragment characteristic length [m].
$\boldsymbol{L}$	Lower triangular matrix.
M	Mean anomaly [rad].
$\mathcal{N}$	Normal distribution.
N	Number of fragments [$-$].
N_b	Number of bins [$-$].
N_p	Number of parameters [$-$].

N_s	Number of samples [−].
P	Period [s].
R	Regulariser [−].
R_E	Earth mean radius [m].
$\mathcal{T}$	Satellite frame.
T	Temperature [K].
T_∞	Exospheric temperature [K].
V	Phase space bin volume.
$\boldsymbol{Y}$	Training data.
Δv	Delta-v magnitude [m/s].
Ω	Right ascension of ascending node [rad].
$\boldsymbol{\Sigma}$	Covariance of normal distribution.
$\boldsymbol{\Theta}$	Surrogate model parameters.
$\boldsymbol{\alpha}$	Keplerian element set.
χ	Logarithm to base 10 of A/m [$\log_{10}(\mathrm{m}^2/\mathrm{kg})$].
δ	Dirac generalized function.
η	Number of impacts [−].
$\dot{\eta}$	Impact rate [1/s].
$\boldsymbol{\kappa}$	Logarithmic kernel weights.
λ	Longitude [rad].
λ	Logarithm to base 10 of L [$\log_{10}(\mathrm{m})$].
μ	Gravitational parameter [$\mathrm{m}^3/\mathrm{s}^2$].
$\boldsymbol{\mu}$	Mean of normal distribution.
ω	Argument of perigee [rad].
$\boldsymbol{\pi}$	Kernel weights.
ρ	Atmospheric density [kg/m^3].
θ	Latitude [rad].
$\boldsymbol{\theta}$	Parameters of normal distribution.
υ	Logarithm to base 10 of Δv [$\log_{10}(\mathrm{m}/\mathrm{s})$].
a	Semi-major axis [m].
c_D	Drag coefficient [−].
e	Eccentricity [−].
f	True anomaly [rad].
g	Source and sink terms.
h	Altitude [m].
h_a	Apogee height [m].

h_p	Perigee height [m].
$\hbar$	Specific angular momentum [m^2/s].
i	Inclination [rad].
l	Log-normal distribution.
m	Object mass [kg].
n	Phase space density.
$\bar{n}$	Mean motion [rad/s].
$\hat{n}$	Surrogate model.
p	Semi-parameter [m].
r	Magnitude of the orbital radius [m].
$\boldsymbol{r}$	Orbital radius [m].
r_a	Apogee radius [m].
r_p	Perigee radius [m].
$\boldsymbol{s}$	Cartesian coordinates.
t	Time [s].
u	Argument of latitude [rad].
v	Magnitude of the orbital velocity [m/s].
$\boldsymbol{v}$	Orbital velocity [m/s].
$\boldsymbol{x}$	Phase space.

1 Introduction

Space debris has become a major concern for satellite operators and space agencies alike. The number of such fragments has grown to an extent where they pose a serious collision risk to active missions. In case of a collision, complete loss of the mission could be the consequence. Understanding the ramifications of orbital fragmentations requires modelling of the evolution of space debris clouds. Such models allow to estimate the current and future risk emanating from fragmentations in terms of collision probability on orbiting payloads and rocket bodies.

Section 1.1 introduces the term space debris, discusses its proliferation and efforts to mitigate its risks. The scope of this thesis is outlined in Section 1.2 in the form of research questions. Section 1.3 reviews previous work that was undertaken to answer the questions. The contributions of this work in the form of novel techniques and publications are listed in Section 1.4. Finally, the structure of the thesis is explained in Section 1.5.

1.1 Background

The space environment is hostile to humans and their assets. Engineers designing and building satellites must tackle extreme temperature variations, space radiation and impacts from micro-meteoroids. They have done so largely successfully. From the first launch of Sputnik 1 in 1957 to the beginning of 2020, more than 9300 payloads[1] were inserted into orbit from over 5500 launches (ESA, 2020). The payloads enable a multitude of services such as telecommunications, broadcasting, navigation, weather forecasting and many more. The launch activities, however, have led to the rise of yet another danger encountered in Earth orbit: space debris.

Space debris comprises all man-made objects including fragments and elements thereof, in Earth orbit or re-entering the atmosphere, that are non-

[1] Payloads are space objects designed to perform a specific function in space excluding launch functionality.

functional (IADC, 2007). About half of the 9300 launched payloads have re-entered. Most of them have decayed naturally due to the drag effects induced by the atmosphere, with only a few performing controlled re-entry manoeuvres (ESA, 2019). Of the other half, 2200 are operational with the remaining 2700 payloads lingering in orbit as space debris (UCS, 2020). Other space debris are 1900 spent rocket bodies[2] used to launch the payloads and 1200 mission related objects, such as launch adapters and camera lids.

Large space debris like these used to serve a purpose earlier in their lives. They are tracked regularly through radar or optical observations, and their orbital states are maintained in catalogues. As a rule of thumb, objects in Low Earth Orbit (LEO), defined up to 2000 km altitude, are catalogued if they are larger than a tennis ball. Objects in Geostationary Orbit (GEO), at an altitude of 35786 km, are catalogued if they have the size of a dish washer or larger.

Most of the objects in Earth orbit today, however, came into existence without ever serving any function. Such fragments, spanning from sub-millimetre to decimetre sized objects, were generated in hundreds of disruptive explosion and collision events, due to degradation of materials, due to the ejection of propellant and from other sources (Smirnov, 2002; NASA, 2018). The 14200 fragments large enough to be catalogued are dwarfed by the population of non-catalogued space debris (ESA, 2019). It is estimated that several hundred thousand fragments sized larger than 1 cm are orbiting Earth, the number of fragments larger than 1 mm is estimated to exceed 100 million (IADC, 2013a).

The largest source of fragments are disruptive satellite disintegrations. Half of all catalogued objects, or 96% of catalogued objects other than payloads, rocket bodies or mission related objects are associated to accidental or intentional disassociations (NASA, 2018). Accidental breakups are collisions between space objects or explosions due to residual propellants, malfunctions during manoeuvres or battery discharges. Two breakups, the intentional destruction of Fengyun 1C in 2007 and the first collision between two intact objects, Cosmos-2251 and Iridium 33, in 2009 added more than 5700 fragments to the catalogue (NASA, 2007, 2009; ESA, 2020). Ten years later, almost three quarters of those fragments remain in orbit. According to the Standard Breakup Model (SBM) of the National Aeronautics and Space Administration (NASA), collisions or explosions of payloads and rocket bodies potentially add tens of thousands and hundreds of thousands new fragments larger than 1 cm and 1 mm, respectively (Johnson et al., 2001).

Millimetre sized fragments potentially penetrate exposed tanks or damage other equipment. Appropriate shielding can prevent such small fragments from harming the satellites (Drolshagen, 2001). The risk emanating

[2]Rocket bodies are space object designed to perform launch related functionality.

from larger, catalogued fragments can be assessed and even managed if orbital manoeuvring capabilities are available. The trajectories of these objects can be propagated into the future, allowing to screen active missions against potential close approaches. In the years 2014 and 2015, the European Space Agency (ESA) performed, on average, one collision avoidance manoeuvre per year and satellite it operates (Krag et al., 2016). Instead, non-catalogued fragments larger than 1 cm are dangerous as their whereabouts are not known well enough to predict close approaches and the effects of a collision could be dramatic. In LEO the average relative orbital velocity between two objects is 10 km/s, equivalent to 36000 km/h. At such speeds even an aluminium sphere with a radius of 1 cm – corresponding to a mass of 11.3 g only – carries the same kinetic energy as a 1.5 tons car travelling at 100 km/h (adapted from Smirnov, 2002).

The space debris situation is expected to deteriorate further. Space traffic has recently risen sharply as a result of miniaturisation and availability of commercial-of-the-shelf products (ESA, 2019). Space traffic might see another sharp increase due to the advent of large constellations. All new objects, if not disposed of by direct re-entry, will become space debris upon reaching the end of their mission. Additionally, the space debris generation is self-feeding. If the fragment density increases, so does the likelihood of collisions. More collisions lead to more fragments being added to the environment, increasing the density further. Such a cascade of collisions was already described in 1978 by Kessler and Cour-Palais (1978). Evolutionary studies performed more recently by major national space agencies and the ESA estimate catastrophic collisions[3] to happen in LEO every 5 to 9 years over the next 200 years (IADC, 2013b).

To mitigate the space debris problem, guidelines and standards were put in place (IADC, 2007; UN, 2010; ISO, 2019). The potential of accidental breakups should be minimised, both during operations and after reaching the end of the mission. Passivation measures, such as depleting residual fuel or discharging batteries, diminish the likelihood of explosions. Performing active avoidance manoeuvres during operations limits the probability of accidental collision in orbit. At the end of mission, payloads and rocket bodies in LEO should be placed in orbits where they decay naturally within 25 years.

However, adoption of the guideline addressing the removal of the object after reaching the end of its mission was slow in the past two decades (ESA, 2019). Even if better compliance was achieved, a risk of accidental breakups remains, e.g. due to collisions with uncatalogued objects. To protect the space environment, the insertion of new payloads and upperstages might require regulation to respect an upper limit, i.e. a capacity, certain orbital

[3] A catastrophic collision is characterized by an impactor kinetic energy-to-target mass-ratio of 40 J/g or higher (Krisko, 2007).

regions can sustain (Krag et al., 2017a). To define such a capacity, the evolution of the fragment cloud and its interactions with other objects in space need to be modelled and assessed. Finding a broad consensus among the international community on how to estimate the ramifications of a breakup is important to identify and define regions that need special protection. The authors of this work aspire to foster such a consensus.

1.2 Research questions

This work is motivated by two overarching and interrelated research questions. The first research question is how to accurately and efficiently model the mid- to long-term evolution of an orbital fragmentation cloud, originating from the catastrophic breakup of a parent object residing on an arbitrary Earth-bound orbit. The application to arbitrary orbits, and various timescales, requires the model to work for a wide range of forces acting on the fragment cloud. The computational resources required to employ the method should remain manageable enabling its application to a large set of scenarios.

The second research question is how to assess the ramifications of the breakup on current or future space missions residing, again, on arbitrary Earth-bound orbits, in terms of collision probability with the evolving fragment cloud. The sensitivity of the modelled evolution of the cloud should be high, as the collision probabilities are generally small.

1.3 Literature review

Many methods and engineering solutions were developed for the purpose of space debris propagation and estimation of their collision probability with existing missions. Here, an overview is given, stating advantages and disadvantages of each method. The methods modelling the evolution of the full space debris population or single fragment clouds can be broadly categorised into probabilistic and deterministic methods. Probabilistic methods were developed prior to deterministic models, hence, they are discussed first in Section 1.3.1. With ever increasing computational power, running deterministic simulations following many individual fragments became feasible. They are introduced in Section 1.3.2. For both categories, the force models are only briefly discussed. Various techniques, discussed in Section 1.3.3, exist to estimate the impact hazard, some of them designed specifically for the evolutionary method used to propagate the fragments.

Next to the methods directly addressing the research questions stated above, more literature needed reviewing to achieve an efficient implementation of the method proposed in this thesis. A considerable part is dedicated

to the semi-analytical integration of trajectories in an atmosphere. Semi-analytical propagation, reviewed in Section 1.3.4, enables the integration of many trajectories in a timely efficient way. Surrogate modelling, introduced in Section 1.3.5, alongside various interpolation techniques, can be applied to further reduce the number of required propagations. The review is summarised in Section 1.3.6.

1.3.1 Probabilistic evolutionary models

Probabilistic methods do not follow fragments individually. Rather, they employ statistical mechanics to estimate the evolution of the fragment density – or continuum – directly.

Origin

Historically, the idea of using statistical mechanics to model space debris as a density distribution was adopted from the community studying stellar dynamics. More than a century ago, the evolution of stars in galaxies were described with Liouville's equation and theorem (Chandrasekhar, 2005). Liouville's equation is a partial differential equation that governs the evolution of a phase space density in Hamiltonian systems. Liouville's theorem says that the density is constant along its trajectories. This is the microscopic equivalent of saying that the macroscopic number of stars – or fragments – is conserved.

Consider a D-dimensional conservative dynamical system with the Hamiltonian, $\mathcal{H}$, described in canonical coordinates, $\boldsymbol{q} \in \mathbb{R}^D$ and conjugate momenta, $\boldsymbol{p} \in \mathbb{R}^D$. The evolution in time, t, of the phase space density, $n(\boldsymbol{q}, \boldsymbol{p}, t)$, is then described through Liouville's equation

$$\frac{\partial n}{\partial t} + \sum_{j=1}^{D} \left(\frac{\partial n}{\partial q_j} \frac{\partial \mathcal{H}}{\partial p_j} - \frac{\partial n}{\partial p_j} \frac{\partial \mathcal{H}}{\partial q_j} \right) = 0 \tag{1.1}$$

Equivalently, given the equation of motion

$$\dot{q}_j = \frac{\partial \mathcal{H}}{\partial p_j} \qquad \dot{p}_j = -\frac{\partial \mathcal{H}}{\partial q_j} \tag{1.2}$$

the Liouville equation can be written as

$$\frac{\partial n}{\partial t} + \sum_{j=1}^{D} \left(\frac{\partial n}{\partial q_j} \dot{q}_j + \frac{\partial n}{\partial p_j} \dot{p}_j \right) = 0 \tag{1.3}$$

To model dissipative forces, such as dynamic friction, the stellar community uses the Fokker-Planck equation (Chandrasekhar, 1949). The Fokker-Planck equation, also called Kolmogorov forward equation, is used to model

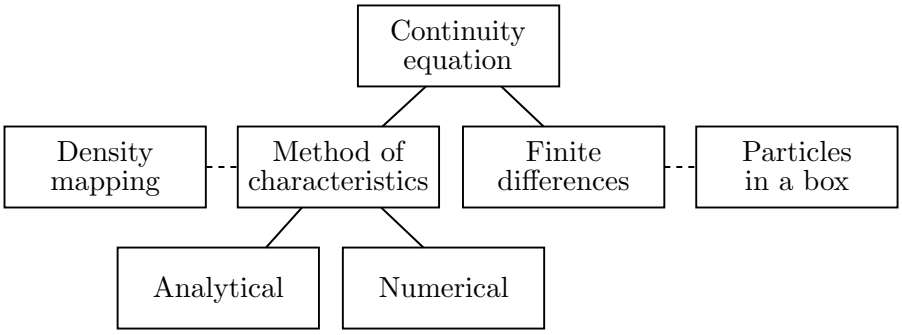


Figure 1.1: Overview of techniques employed to solve the continuity equation in space debris evolution modelling.

the evolution of densities subjected to Brownian motion. It reduces to the Liouville equation if the diffusion term is ignored. Both the Liouville equation and the Fokker-Planck equation are particular types of the continuity equation (González et al., 2016).

With the phase space density, $n = n(\boldsymbol{x}, t)$, as a function of the phase space, $\boldsymbol{x}$, and time, t, the continuity equation for a conserved n, i.e. in the absence of sources and sink terms, is given as

$$\frac{\partial n}{\partial t} + \boldsymbol{\nabla} \cdot (n\boldsymbol{F}) = 0 \tag{1.4}$$

with the divergence operator, $\boldsymbol{\nabla}\cdot$, and the flow defined by the dynamics, $\boldsymbol{F}$

$$\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}t} = \boldsymbol{F} \tag{1.5}$$

Just like the Liouville equation, the continuity equation states that, in the absence of sources or sink terms, the number of particles is conserved, i.e. the accumulation inside a control volume equals the difference between entering and exiting particles.

A more recent example for application of the statistical mechanics to model the evolution of interplanetary dust is given by Gor'kavyi et al. (1997). Using directly the continuum equation allows to incorporate the non-conservative Poynting-Robertson drag. Gor'kavyi et al. (1997) also show that orbital elements are well suited to model the evolution of long-term phenomena.

There are different ways of solving the continuity equation, the ones focusing on space debris are discussed in the following. Figure 1.1 gives an overview of the different techniques. They can be further sub-categorised into methods that solve the system along the trajectories, i.e. method of characteristics and density mapping, and others that solve it at fixed points in the phase space, i.e. finite differences.

Method of characteristics

For the convenience of the reader, a brief recap of the method of characteristics is given here, following the outline of John et al. (1981, Chapter 1.4), and applied to the continuity equation. Consider a general quasi-linear partial differential equation

$$a(x,t,n)\frac{\partial n}{\partial t} + b(x,t,n)\frac{\partial n}{\partial x} = c(x,t,n) \tag{1.6}$$

The characteristics are curves, which at each point are tangent to the characteristic direction, defined by (a, b, c). Along the characteristic curves the following relation holds

$$\frac{\mathrm{d}t}{a(x,t,n)} = \frac{\mathrm{d}x}{b(x,t,n)} = \frac{\mathrm{d}n}{c(x,t,n)} = \frac{\mathrm{d}\tau}{1} \tag{1.7}$$

introducing τ as a suitable parameter to refer to the curve. Equation (1.7) thus defines the following ordinary differential equation

$$\frac{\mathrm{d}t}{\mathrm{d}\tau} = a(x,t,n) \tag{1.8a}$$
$$\frac{\mathrm{d}x}{\mathrm{d}\tau} = b(x,t,n) \tag{1.8b}$$
$$\frac{\mathrm{d}n}{\mathrm{d}\tau} = c(x,t,n) \tag{1.8c}$$

If the derivatives of a, b and c exist and are continuous in a given domain, only one characteristic curve passes through each point in this domain. Hence, the method of characteristics solves first order partial differential equations by converting them into a set of ordinary differential equations.

The system of ordinary differential equations solving the continuity equation looks even simpler. First, note that the continuity equation, expressed in Equation (1.4) can be, by evaluating the divergence term and applying the chain rule, written as

$$\frac{\partial n}{\partial t} + \sum_{j=1}^{D}\left(\frac{\partial n}{\partial x_j}F_j\right) = -n\sum_{j=1}^{D}\frac{\partial F_j}{\partial x_j} \tag{1.9}$$

Second, as $a = 1$, it follows from Equation (1.8a) that $t = \tau$. Thus, the following set of ordinary differential equations solve the continuity equation

$$\frac{\mathrm{d}n}{\mathrm{d}t} = -n\sum_{j=1}^{D}\frac{\partial F_j}{\partial x_j} \tag{1.10a}$$
$$\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}t} = \boldsymbol{F} \tag{1.10b}$$

From Equation (1.10b) it is evident that the characteristics follow the trajectories governed by the dynamics. Equation (1.10a) can be solved for any dynamics independent of n via separation of variables

$$\frac{\mathrm{d}n}{n} = -\sum_{j=1}^{D} \frac{\partial F_j}{\partial x_j}\,\mathrm{d}t \tag{1.11}$$

Integration on the left-hand side, from $n(\boldsymbol{x}_0, t_0)$ to $n(\boldsymbol{x}, t)$, and the right-hand side from t_0 to t gives

$$\ln\left(\frac{n(\boldsymbol{x}, t)}{n(\boldsymbol{x}_0, t_0)}\right) = -\int_{t_0}^{t} \sum_{j=1}^{D} \frac{\partial F_j}{\partial x_j}\,\mathrm{d}t \tag{1.12}$$

which can be solved for $n(\boldsymbol{x}, t)$ as

$$n(\boldsymbol{x}, t) = n(\boldsymbol{x_0}, t_0) \exp\left(-\int_{t_0}^{t} \sum_{j=1}^{D} \frac{\partial F_j}{\partial x_j}\,\mathrm{d}t\right) \tag{1.13}$$

A simple example of an application of the method of characteristics is given in Appendix A.1.

The first to publish such a continuum approach in the context of space debris was Heard (1976). Contrary to stellar physics, he notes, the problem of modelling debris is less complicated because attractive forces between the fragments can be neglected. He uses the Liouville equation to model a continuum of non-interacting particles, such as fragments from an exploding satellite. The particles are assumed to originate from the same point but dispersed in initial velocities. Only the short-term evolution, subject to linearised two-body dynamics around a circular orbit, is considered, permitting an analytical solution. Although Heard (1976) does not specifically mention it, he solves the problem along the characteristics.

Jehn (1990) extends the approach by using satellite fragmentation models for the initial distribution and omitting the assumption of circular orbits. He uses an extended phased approach initially proposed by McKnight and Lorenzen (1989), which separates the evolution of the cloud into different phases. For the initial phase, where the distribution remains close to the parent object, and the second phase, where the fragments spread out into a torus along the orbit, Jehn (1990) uses the Liouville equation to model the evolution. For the mid- to long-term evolution, he utilises a deterministic approach based on samples to build a probabilistic density model for lack of an analytical solution.

An analytical model for the long-term evolution of space debris, subject to non-conservative forces, is described by McInnes (1993). He applies the continuity equation, considering various models for source terms and atmospheric drag, however, only in two dimensions, time and radius. Analytical

solutions are found through the method of characteristics for simplifying assumptions, such as a single exponentially decaying atmosphere model and circular orbits.

Letizia et al. (2015) combine the technique developed by McInnes (1993) with initial conditions derived through samples from the NASA SBM. This allows to model the evolution of fragmentation clouds stemming from collisions and explosions. Varying fragment sizes are considered by splitting the drawn fragments into different size bins and solving the continuity equation in each bin separately. The method is extended into multiple dimensions to simultaneously account for Earth's oblateness and drag (Letizia et al., 2016a). However, the assumption on quasi-circular orbits is maintained and a constant eccentricity is required to find analytical solutions. Additionally, the initial distribution in the required phase space is obtained from samples that are individually and numerically propagated, considerably reducing the speed improvements over deterministic models.

The techniques presented so far all obtain closed form solutions of the continuity equation, i.e. the integral in Equation (1.13) can be found analytically. The main advantage of closed form solutions is that the density at any point in time can be evaluated instantaneously without integration along the trajectory. However, these methods rely on simplified orbital geometries, e.g. assuming circular orbits, or simplified force models. More elaborate configurations and dynamics can be considered if the characteristics are numerically propagated.

Such a method was developed by Halder and Bhattacharya (2011) for uncertainty propagation in planetary entry, descent and landing. The space density function for numerically integrated characteristics is available only along the propagated trajectory. This constitutes also the biggest drawback of the method, as the density values in between the propagated trajectories require either further propagation, binning or interpolation. Similarly, Trisolini and Colombo (2019) numerically propagate the characteristics to model uncertainties of objects re-entering in the atmosphere. The methods these authors apply to obtain a density estimate over the full domain are discussed in Section 1.3.5.

Density mapping

Similarly, the solution of the continuity equation can be found through transformation of the density. Density mapping relies on the assumption that the density, or volume, is conserved locally, i.e. no source or sink terms are present. Given the solution of the dynamics in Equation (1.5), as

$$\boldsymbol{x} = \boldsymbol{\varphi}(\boldsymbol{x}_0) \tag{1.14}$$

the mapping of the density can be found as (Soong, 2004, Chapter 5.3)

$$n(\boldsymbol{x}, t) = \frac{n\left(\boldsymbol{\varphi}^{-1}(\boldsymbol{x}), t_0\right)}{|\det \boldsymbol{J}|} \tag{1.15}$$

with the Jacobian, $\boldsymbol{J} \in \mathbb{R}^{D \times D}$, defined as

$$J_{ij} = \frac{\partial \varphi_i}{\partial x_{0,j}} \qquad i, j = 1, 2, \ldots, D \tag{1.16}$$

and the determinant, det. For comparison of the density mapping technique with the method of characteristics, the former is applied in Appendix A.1 to solve the same initial value problem as in the previous example. For conservative systems, i.e. in the absence of sources and sinks, they provide equal solutions.

Ashenberg (1992) modelled the evolution of an isotropic explosion through density mapping of the debris cloud volume, considering the oblateness of Earth and atmospheric drag through linearised relative motion. Later, he applies the Liouville equation to the same problem, finding that only drag contributes to a change of the volume (Ashenberg, 1994). Gravitational forces only shift the volume in the phase space but do not change its magnitude.

Jenkin (1996) relies on volume mapping to calculate the short-term collision probability satellites are exposed to evolving fragment clouds. The solutions are found by application of an extended Lambert solver that finds the mappings for two-point boundary value problems considering perturbations from an oblate Earth. The spatial density is found through application of Equation (1.15) using the linearised state transition matrix. A similar approach is applied by Healy et al. (2016) using the NASA SBM for the initial fragment distribution. The method is applicable for short time frames in the order of multiple revolutions.

Long-term propagation of high area-to-mass ratio fragments is the subject of a method proposed by Wittig et al. (2017). They combine the density mapping method with differential algebra and semi-analytical integration of the trajectories. The advantage of using differential algebra is that the elements of the Jacobian in Equation (1.15), which are generally not available in a closed form solution, do not need to be integrated separately. Instead, the derivatives required to build the Jacobian are directly derived from the polynomial expansion of the solution of the dynamics. Additionally, the density can be evaluated not only along the trajectories, but also in the vicinity of it. However, the differential algebra framework is cumbersome, as each variable and operation needs to be replaced with its differential algebra representation, and its accuracy depends on a potentially high degree of the Taylor approximation.

Density mapping requires the integration of D state variables and, in the absence of differential algebra, D^2 Jacobian elements. Hence it is computationally more expensive than application of the method of characteristics which adds only one more state to be integrated to the state variables. Additionally, density mapping does not permit consideration of sources or sink terms, which make it less interesting for space debris modelling, where sources such as launch traffic and collisions exist.

Particles in a box and finite differences

Instead of finding the solution of the continuity equation along the trajectories, the solution can be approximated via discretisation of the phase space. Within each bin the continuity equation is – using finite-differencing – replaced by its discretised version and solved numerically. All the bins are simultaneously integrated for a carefully chosen step size. After the integration step, all the finite differences in each bin are updated according to its value of the density and that of its adjacent neighbours. This process is repeated until the result is found for the desired epoch.

For a mathematical description of the finite difference method solving the continuity equation, take Equation (1.4) and discretise the terms using forward and central differences

$$\frac{\partial n}{\partial t} \approx \frac{n(\boldsymbol{x}, t + \Delta t) - n(\boldsymbol{x}, t)}{\Delta t} \tag{1.17a}$$

$$\frac{\partial n}{\partial x_j} \approx \frac{n(\boldsymbol{x} + \boldsymbol{A}_j \Delta \boldsymbol{x}, t) - n(\boldsymbol{x} - \boldsymbol{A}_j \Delta \boldsymbol{x}, t)}{2\Delta x_j} \tag{1.17b}$$

$$\frac{\partial F_j}{\partial x_j} \approx \frac{F_j(\boldsymbol{x} + \boldsymbol{A}_j \Delta \boldsymbol{x}, t) - F_j(\boldsymbol{x} - \boldsymbol{A}_j \Delta \boldsymbol{x}, t)}{2\Delta x_j} \tag{1.17c}$$

where $\boldsymbol{A}_j \in \mathbb{R}^{D \times D}$ matrix with all zeros except at element jj, e.g.

$$\boldsymbol{A}_j = \mathrm{diag}(\delta_{1j}, \delta_{2j}, \ldots, \delta_{jj}, , \ldots, \delta_{Dj}) \tag{1.18}$$

and δ is the Kronecker delta

$$\delta_{ij} = \begin{cases} 1, & \text{if } i = j, \\ 0, & \text{if } i \neq j. \end{cases} \tag{1.19}$$

Replacing the terms in Equation (1.4) with the approximations found in Equation (1.17), noting the terms $n(\boldsymbol{x} + \boldsymbol{A}_j \Delta \boldsymbol{x}, t)$ and $n(\boldsymbol{x} - \boldsymbol{A}_j \Delta \boldsymbol{x}, t)$ are the values of the density adjacent to each box in j-direction, and solving for $n(\boldsymbol{x}, t + \Delta t)$ integrates the density for each bin for one step.

Often, instead of propagating directly the density, the number of objects in each bin – the number of particles in a box – are propagated. If the

bins are chosen to remain constant, a simple relation between the number of fragments, N, and the phase density, n, is found via the bin volume, V, as

$$N(V) = n(V)V \tag{1.20}$$

The first to apply such a finite differencing scheme in the context of space debris were, independently, Farinella and Cordelli (1991) and Talent (1992). They consider different species of objects, i.e. satellites and fragments thereof. The satellite and fragment populations are coupled through collision and explosion terms and even launches are modelled. Note that in the continuity equation such effects would be considered by introducing source and sink terms, i.e. with a non-vanishing right-hand side. However, they do not contain any dependence on the location of the orbit, hence, decay due to atmospheric decay is modelled rudimentarily.

A more elaborate approach, considering different height bands, was developed by Smirnov et al. (1993). It models perturbing forces such as drag, solar radiation pressure, and sources such as explosions and collisions. Later, the same model was extended to incorporate bins in fragment size, perigee altitude, eccentricity, inclination and ballistic coefficient (Smirnov et al., 2001). Similarly, Rossi et al. (1994) extended the approach of Farinella and Cordelli (1991) to space debris considering altitude and fragment size bins, and later semi-major axis, eccentricity and mass (Rossi et al., 1998). The model considers launches, explosions and collisions and accommodates elaborate force models.

More recently Letizia (2018) applied the finite difference method specifically to evolving fragmentation clouds considering semi-major axis and eccentricity. Different mass bins are considered, but as collision within the cloud is not modelled, the mass evolution is decoupled from each other. Somma et al. (2019) introduced a multi-species evolution model with discrete height bands, allowing to study the sensitivity of launch activities and post-mission disposal in LEO.

The issue with finite differences is that number of bins grows exponentially with the number of dimensions of the phase space. Generally, such methods are thus not applicable to phase spaces of four dimensions or more. Consider the propagation of a density in all six orbital elements. If each dimension is split into 100 bins, the number of bins would be 10^{12}. Next to memory issues, the 10^{12} equations to be solved for each time step would also challenge any processing unit. Such limitations led to the raise of deterministic models.

1.3.2 Deterministic evolutionary models

Deterministic evolutionary models do not aim to propagate the density. Instead, they propagate individual fragments that are sampled from a given

initial distribution. If the model is run with the same set of fragments, the same result is obtained, hence the term deterministic. Monte Carlo simulation, i.e. repeated sampling and execution of the model, enables to estimate expectation values of the variables of interest. E.g. the mean, $\boldsymbol{\mu} \in \mathbb{R}^D$, of a joint probability distribution function, $p_{\boldsymbol{x}}$, is estimated from N_s simulation runs as

$$\boldsymbol{\mu}_x = \mathbb{E}(\boldsymbol{x}) = \int_{\mathbb{R}^D} \boldsymbol{x}\, p_{\boldsymbol{x}}\, \mathrm{d}^D\boldsymbol{x} \approx \frac{1}{N_s} \sum_{j=1}^{N_s} \boldsymbol{x}_j \tag{1.21}$$

Over the last two decades, ESA and almost all the major national space agencies, or universities and research groups associated with them, have developed deterministic evolutionary models: DELTA (ESA, Walker et al., 2001), LEGEND (NASA, Liou et al., 2004), DAMAGE (United Kingdom Space Agency, Lewis et al., 2004), SDM (Agenzia Spaziale Italiana, Rossi et al., 2009), GEODEEM and LEODEEM (Japan Aerospace Exploration Agency, Hanada et al., 2009), MEDEE (Centre National d'Études Spatiales, Dolado-Perez et al., 2013), LUCA (Deutsches Zentrum für Luft- und Raumfahrt, Radtke et al., 2017) and SOLEM (China National Space Administration, Wang and Liu, 2019).

Some of them were initially designed to study either the LEO or GEO regime, but then eventually extended to either both regimes or the full space environment around Earth. All of them apply elaborate force models, incorporate launch traffic and mitigation measures and factor in collisions and explosions. They are usually applied to estimate the evolution of the total number of objects decades or centuries into the future (IADC, 2013b). Instead of sampling from an initial distribution, each Monte Carlo run starts with the known population of fragments. Collisions are introduced in a statistical manner. First, the collision probability is estimated using the Cube algorithm, which is discussed in Section 1.3.3. Then a collision is simulated or not by comparing the probability to a uniformly drawn number (Liou, 2006). The breakups are modelled according to the NASA SBM (Johnson et al., 2001), although some of the tools offer alternative breakup models. Note that despite having the same initial conditions, each Monte Carlo run will result in different outcomes. The predicted number of space debris is then estimated using Equation (1.21).

Being able to model any level of complexity is the main advantage of resorting to Monte Carlo simulation. Today, deterministic models are normative for the works of the Inter-Agency Space Debris Coordination Committee (IADC) when it comes to the prediction of future evolution of space debris or assessing the effects of large constellations on the space debris population (IADC, 2013b, 2017). However, there are drawbacks. Crude Monte Carlo sampling is lacking the sensitivity to capture rare events with low probabilities (Au and Beck, 2001). Even with large numbers of drawn samples, convergence towards an accurate value of the low density is slow.

The integration of the trajectories of large numbers of fragments is time consuming even if semi-analytical theory is used. Extensions to Monte Carlo sampling, such as importance sampling (Bishop, 2006, Chapter 11.1) or subset simulation (Au and Beck, 2001), could be used, however, at the expense of global applicability of the method. Using such a method, each target object would require a dedicated Monte Carlo simulation when calculating its collision probability.

To reduce the computational time, the models are usually run with a focus on objects larger than 10 cm only, e.g. simulating only $20000 - 30000$ fragments per run (IADC, 2013b). Such an approach ignores the danger emanating from the objects of size 1 cm or larger, of which there are an order of magnitude more fragments in orbit. Another way of reducing the number of fragments to be propagated is to use representative objects (Rossi et al., 1998). Instead of propagating all individual objects, the fragments are grouped together and represented by fewer objects carrying the appropriate weight. However, for statistical evaluation of the collision probability, this makes only partially sense. Grouped fragments are less likely to lead to close encounters. If a close encounter occurs anyway, the object carries the weight of multiple fragments therefore overestimating the collision probability. Hence, using representative fragments decreases the number of samples per run, but increases the variability of the outcome and thus the number of Monte Carlo simulations required for convergence.

Another drawback of deterministic integration, related to the sensitivity issue, is that accurate estimates of the underlying distributions in dimensions $D \geq 3$ are difficult to obtain. The distribution can be of any shape and is generally not be well described by a mean and variance. Instead, an estimate of the phase space density, $\hat{n}$, can be obtained by discretisation of the phase space as

$$\hat{n}(V) = \frac{N(V)}{V} \tag{1.22}$$

with a bin volume, V, and the number of fragments present in said volume, $N(V)$. However, suppose that the spatial density – a three-dimensional quantity – was to be inferred from samples through discretisation. The spherical LEO volume, from $r_0 = R_E + h_0$ to $r_1 = R_E + h_1$, with the radius of Earth, $R_E = 6371$ km, and the lower and upper altitudes of the LEO regime, $h_0 = 0$ km and $h_1 = 2000$ km (IADC, 2007), is

$$V_{\text{LEO}} = \frac{4\pi}{3}(r_1^3 - r_0^3) = 1.37 \times 10^{12} \text{ km}^3 \tag{1.23}$$

Discretising V_{LEO} into bins with edge lengths 10 km$\times$10 km$\times$10 km requires 1.37×10^9 bins. The number of propagated samples required to yield accurate estimates of the density in all those bins incurs a high computational cost. The extension to higher dimensions, elaborated quantitatively in Sec-

tion 2.1, becomes infeasible as the number of bins grows exponentially with dimension.

1.3.3 Collision probability estimation

Once an estimate of the density, or number of fragments, is available, the collision probability of a mission of interest can be calculated. In accordance with the many evolutionary models, a multitude of collision probability estimation techniques exist. The collision probability is always a probabilistic measure, but it can be obtained from both probabilistic and deterministic evolutionary models.

As for the evolutionary models, the methods developed to calculate the collision probability with space debris fragments were adapted from the study of collisions on a grander scale. Öpik (1951) studies the collision probability of planets with interplanetary dust particles – modelled as densities – over time frames of hundreds of millions of years. He finds that the secular perturbations lead to progressive changes in node and argument of perigee that are much shorter than the lifetime of the particles. This justifies the assumption of modelling these variables as uniformly distributed, i.e. randomised. Using geometrical considerations, he finds the collision probability of dust particles in eccentric and inclined orbits with Earth in circular orbit.

In their seminal work about the formation of a debris ring around Earth Kessler and Cour-Palais (1978) employ a similar statistical approach (explained in more detail in Kessler, 1981). The flux, F, within a given control volume, ΔV, is, analogous to the kinetic gas theory, estimated as

$$F = n_{\boldsymbol{r}} \Delta \bar{v} \tag{1.24}$$

with the fragment spatial density, $n_{\boldsymbol{r}}$, and the average relative velocity, $\Delta \bar{v}$, between the particles in the control volume and a target object of interest. The number of impacts, η, for an object traversing the volume within a duration of Δt is

$$\eta = F A_c \Delta t \tag{1.25}$$

where A_c is the average collisional cross-sectional area between the fragments and the target. To calculate the collision probability between two objects, Kessler and Cour-Palais (1978) first estimate the spatial density of a single object, $n_{\boldsymbol{r}}^{(i)}$, as

$$n_{\boldsymbol{r}}^{(i)} = \frac{\Delta t^{(i)}}{P^{(i)}} \frac{1}{\Delta V} \tag{1.26}$$

with the orbital period, P. The collision rate between the two objects in a single volume is then calculated as

$$\dot{\eta}_{ij}(\Delta V) = n_{\boldsymbol{r}}^{(i)} n_{\boldsymbol{r}}^{(j)} \Delta \bar{v} A_c \Delta V \tag{1.27}$$

To estimate the total collision rate between the two, Equation (1.27) is summed over all the volumes accessible to both objects.

Later, Su and Kessler (1985) apply the Poisson function to obtain an estimate of the collision probability. The Poisson distribution is used to describe probabilistically how often an event occurs in an interval of time over which η is accumulated. Hence, the distribution

$$p_k(\eta) = \frac{\eta^k \exp(-\eta)}{k!} \tag{1.28}$$

gives a measure of the probability of observing k collisions. The probability of witnessing one or more collision is thus

$$p_{1+} = 1 - p_0 = 1 - \exp(-\eta) \tag{1.29}$$

Note that if $\eta \ll 1$ is small, Equation (1.29) can be approximated as

$$p_{1+} \approx \eta \tag{1.30}$$

A similar statistical and geometrical approach is taken by Chobotov (1983), McKnight (1990) and Letizia et al. (2016b) to estimate the number of collisions of a payload or rocket body with a background of tracked objects or a cloud of fragments.

Liou (2006) introduces the Cube algorithm to find a computationally efficient extension of the method introduced by Kessler (1981) for processing of the collisions between a large set of objects. The name comes from the discretisation of the physical space into spatial cubes. Instead of randomising the node and argument of perigee, the method randomly samples in time, i.e. in mean anomaly. Whenever two objects end up in the same Cube, the collision rate is estimated according to Equation (1.27). Doing so, the complexity of the problem only rises linearly in number of objects, rather than quadratically. However, the collision probability scales with the cube size (Lewis et al., 2019). To be accurate, the cubes need to be small and the sampling intervals short.

Instead of resorting to geometrical considerations or sampling, the spatial density can also be found directly through application of the mapping of the density from a velocity distribution at breakup to a spatial distribution at the epoch of interest. Jenkin (1996) finds the probability of impact of a single fragment from a distribution of fragments originating from an explosion through such a mapping. If the fragment spread velocities at breakup are independent and noting that only one fragment can be involved in a collision, he models the probability of impact considering all fragments with a binomial distribution. If the number of fragments is large, the binomial distribution can be approximated with the Poisson distribution. For hypervelocity collisions or energetic explosions, the assumptions are satisfied, he

notes. Thus, he shows that the Poisson distribution is also applicable to the estimation of the short-term hazard, despite the correlated motion of the cloud fragments not resembling the motion of gas particles.

The kinetic gas theory is applied when the exact location of a fragment is not known. It is then assumed that the likelihood to find the fragment is equally spread across a sufficiently small volume. This does not mandate its velocity distribution to be random. Correlation of the fragment directions can be considered through proper weighting with the relative velocities. It is indeed important that not only the spatial distribution of the fragments is known, but also its dependence on the velocity. If the control volume is chosen to be much smaller than the uncertainties – ideally as small as the target object itself, assuming its orbit is known precisely – the assumption of the position of the fragment being uniformly distributed over the volume is satisfied.

Izzo and Valente (2004) use the Dirac delta function to transform space object distributions given in orbital elements into spatial densities, required for computation of the collision probability. This enables the evaluation of the collision probability without any hypothesis on the orbit geometry or discretisation of the phase space. However, they do not extend the method to uncertainties over multiple chaser fragments. Contrary to the change of variables method in Equation (1.15), which is only applicable to transformations of phase spaces with the same dimensionality, the Dirac delta function offers a direct way of transforming higher dimensional phase spaces to lower dimensional ones. The method is explained in greater detail in Section 2.2.1.

1.3.4 Semi-analytical propagation in an atmosphere

Objects in Earth orbit do not move in perfect Keplerian motion, i.e. along repeating ellipses. Instead, they are subject to perturbing forces. Earth is not a perfect sphere and its mass is unequally distributed within its body. Hence, the attractive gravitational field is not spherical either. Other solar system bodies – such as the Moon and the Sun – further disturb the gravitational field. Non-conservative perturbations are induced due to atmospheric drag and solar radiation pressure.

Any perturbing force can be accounted for in continuum modelling if the trajectories in Equation (1.5) are numerically integrated. An overview of numerical integration schemes can be found in Vallado (2013, Chapter 8.5). Numerical integration of the full force model is used for precise orbit propagation over the course of a few orbits or days only. As tens or hundreds of steps need to be evaluated for each orbital revolution, such an integration scheme quickly becomes inefficient for mid- to long-term propagation.

This subsection briefly introduces semi-analytical integration techniques and discusses its advantages and disadvantages. The focus is put on trajectory integration subject to atmospheric drag, which was identified as not

adequately modelled in the propagator used throughout this work. The propagator suite itself, and its extension to increase the accuracy of integration, are introduced in Sections 3.3 and 3.4.

Semi-analytical integration

To improve the efficiency of orbit integration while maintaining a high level of accuracy, semi-analytical propagation techniques were developed (e.g. Liu, 1974). Such techniques remove the short-periodic effects by averaging the dynamics. This allows to increase the step size of the integration routine, decreasing the computational time required for the full propagation. Short-periodic effects can be identified through the dependence on fast variables in the force model. Fast variables show large variations over a single orbital revolution, while slow variables change only marginally (Vallado, 2013). Radius and velocity written in Cartesian coordinates are all fast variables. Orbital elements, instead, usually have a single fast variable and are thus predestined for averaging techniques. For Keplerian elements, $\boldsymbol{\alpha}$, the fast moving variable is the anomaly term, e.g. the mean anomaly, M.

The averaged dynamics are obtained by integrating the dynamics over one orbit and dividing by the time it takes to complete the revolution as

$$\frac{\mathrm{d}\overline{\boldsymbol{\alpha}}}{\mathrm{d}t} = \frac{1}{P}\int_0^P \left(\frac{\mathrm{d}\boldsymbol{\alpha}}{\mathrm{d}t}\right)\mathrm{d}t \tag{1.31a}$$

$$= \frac{1}{2\pi}\int_0^{2\pi} \left(\frac{\mathrm{d}\boldsymbol{\alpha}}{\mathrm{d}t}\right)\mathrm{d}M \tag{1.31b}$$

where the orbital period, P, and M are found as

$$P = \frac{2\pi}{\bar{n}} \qquad M = M_0 + \bar{n}t \tag{1.32}$$

given the mean motion, $\bar{n}$. The singly averaged state is propagated numerically using the averaged dynamics from Equation (1.31) which are independent of the fast variables. Doubly averaged dynamics are obtained by performing the same averaging over the long-periodic effects. There are different ways of solving Equation (1.31), as is discussed below, right after introducing the most common atmosphere models.

Atmosphere models

The atmosphere models discussed here can be divided into reference models and the derivatives thereof. The reference models commonly give the temperature, T, and – more importantly for calculating the drag force – the density, ρ, of Earth's atmosphere as a function of the altitude, h, and other input parameters. Examples are, in increasing degree of complexity, the COSPAR

International Reference Atmosphere (CIRA) model (e.g. CIRA-86, Rees, 1988), the Jacchia atmosphere (Jacchia, 1977), the Drag Temperature Model (DTM) (Bruinsma, 2015) and the Naval Research Laboratory Mass Spectrometer, Incoherent Scatter Radar Extended (NRLMSISE) model (Picone et al., 2002), all of which are (semi-)empirical models.

Of these reference models, reduced derivatives can be obtained through fitting for two purposes: appropriate simplification of the mathematical formulation can lead to significant speed increases for a density evaluation; and adequate reformulation of the model enables averaging for semi-analytical propagation. The Jacchia-77 atmosphere model and a commonly used derivation of the CIRA model are discussed in more detail.

Jacchia-77 The Jacchia-77 reference atmosphere (Jacchia, 1977) estimates the temperature and density profiles of the relevant atmospheric constituents as a function of the exospheric temperature, T_∞. The density profile, ρ_J, is based on the barometric equation and an empirically derived temperature profile calibrated with observations of satellite decay. The model is valid for altitudes 90 km $< h <$ 2500 km and exospheric temperatures 500 K $< T_\infty <$ 2500 K. The computation of ρ_J cannot be performed analytically and requires numerical integration for each of the 4 constituents, nitrogen, oxygen, argon and helium, plus integration of atomic nitrogen and oxygen. The scale height, H, is defined as

$$H = -\rho \left(\frac{\mathrm{d}\rho}{\mathrm{d}h} \right)^{-1} \tag{1.33}$$

and is numerically approximated here as

$$H_J(h) = -\frac{\rho_J(h)\Delta h}{\rho_J(h+\Delta h) - \rho_J(h)} \qquad \Delta h = 1 \text{ m} \tag{1.34}$$

Several thermospheric variations can be considered, such as solar cycle, solar activity, seasonal or daily variations. Generally, the objects of interest for semi-analytical propagation dwell on-orbit for several months to hundreds of years. Thus, only the variation with the 11-year solar cycle is of interest here. The Jacchia reference uses the solar radio flux at 10.7 cm, F, as an index for the solar activity (see Figure 1.2, source for data: Goddard Space Flight Center, 2020). From F, T_∞ can be inferred as (Jacchia, 1977)

$$T_\infty = 5.48\overline{F}^{\frac{4}{5}} + 101.8F^{\frac{2}{5}} \tag{1.35}$$

where $\overline{F}$ is a smoothed F, commonly centred over an interval of several solar rotations. Jacchia recommends using a smooth Gaussian mean based on weights, w, which decay exponentially with time, i.e.

$$\overline{F} = \frac{\sum_t w_t F_t}{\sum_t w_t} \qquad \text{where } w_t = \exp\left(-\left(\frac{t - t_0}{\sigma} \right)^2 \right) \tag{1.36}$$

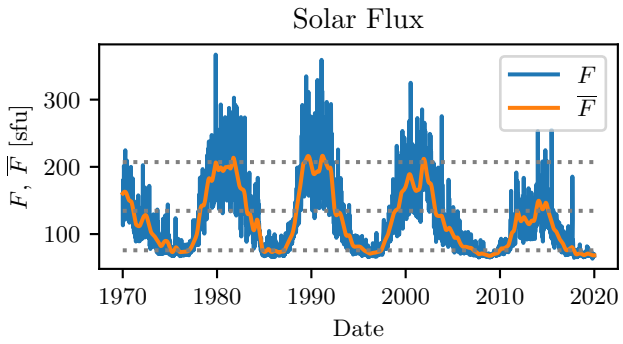


Figure 1.2: Daily 10.7 cm solar flux, and a Gaussian mean with $\sigma = 81$ days and a window of $w = 486$ days, since beginning of 1970. The dashed lines correspond to $T_\infty = 750$, 1000 and 1250 K, respectively, assuming $F = \overline{F}$.

with the time of interest, t_0, and the standard deviation, σ. Note that $\overline{F}$ requires the knowledge of the future F. Figure 1.2 shows F and its Gaussian mean, $\overline{F}$, with a standard deviation of $\sigma = 81$ days, considering a cut-off window for the calculation of $\overline{F}$ of $t \in t_0 \pm 3\sigma$. More recent models such as NRLMSISE or DTM require $\overline{F}$ to be a moving mean, centred in a 81 days window (ISO, 2013).

Non-Smooth Exponential One very simple representation of the atmosphere density is using a piecewise exponentially decaying model, by dividing the altitude range into bins. Each bin is defined by a lower altitude (base) and an upper altitude (base of the next bin), h_i and h_{i+1}, respectively, the base density, $\hat{\rho}_i$, at h_i and a scale height, H_i, chosen such that the density is continuous over the limits of each bin. Then, within each altitude bin, the density, ρ_{NS}, can be evaluated at each altitude h as follows

$$\rho_{NS}(h) = \hat{\rho}_i \exp\left(\frac{h - h_i}{H_i}\right) \qquad h_i < h < h_{i+1} \tag{1.37}$$

Such a model can be derived from any atmospheric model. For this work, the values given in Vallado (2013, Chapter 8.6) – fitting the CIRA-72 model at T_∞= 1000 K – are used for a comparison of models.

A problem with the non-smooth atmosphere model is that it is non-physical, with discontinuities in H. At each change of altitude bin, H jumps from H_i to H_{i+1}. This non-smooth behaviour potentially poses a problem to the (variable-step size) integrator, as the step size needs to be reduced to accurately describe the sudden change in contraction rate of the orbit. Thus, the number of function evaluations and the total time to propagate the orbit increases, if the discontinuities are not handled properly.

Averaging techniques for atmospheric drag effects

During semi-analytical propagation of an object trajectory subject to air-drag forces, the integrated rate of change in the orbital element space, i.e. the average rate of change over a full revolution is of interest. The evaluation of the average rate can either be achieved numerically using quadrature, or analytically. Many quadrature rules exist (e.g. see Abramowitz and Stegun, 1972, p. 885–895) and they are independent of the underlying function, making them versatile. However, they require the evaluation of the density at multiple nodes along the orbit, increasing the computational load of the function evaluations during integration. Note that in this context numerical integration describes the evaluation of an integral and not the propagation of a trajectory.

Analytical formulations, such as the one derived by King-Hele more than half a century ago (King-Hele, 1964) require the density to be evaluated only once per iteration in correspondence of the perigee altitude. Other examples of analytical formulations are the ones derived by Vinh et al. (1979), Sharma (1999) and Xavier James Raj and Sharma (2006). While offering improvements to the classical formulation of King-Hele (KH), such as being mathematically more rigorous and non-singular, they are either only available as closed-form solution or suffer from the same assumption of a fixed scale height. Here, only the numerical Gauss-Legendre (GL) quadrature and the analytical KH method are briefly discussed. The dynamical model is presented in Section 3.4.

Gauss-Legendre quadrature The integrals in Equation (1.31) can be approximated numerically using quadrature, e.g. GL quadrature (Abramowitz and Stegun, 1972, p. 887)

$$\int_0^{2\pi} f(y)dy \approx \pi \sum_i w_i f(y_i), \quad y_i = (x_i + 1)\pi \tag{1.38}$$

where the node x_i is the i^{th} root of the Legendre Polynomial $P_n(x)$. The weights w_i are given as

$$w_i = \frac{2}{(1 - x_i^2)[P_n'(x_i)]^2} \tag{1.39}$$

and P_n' is the derivative of $P_n(x)$ with respect to x. The nodes and weights remain constant during the propagation, so they are calculated (or read from a table) only once upon initialisation of a program. Routines to calculate (x_i, w_i) are available for various scientific programming tools.

Advantages of a numerical approximation of the integrals is that it can be found for any atmospheric model and that no series expansions are required. Disadvantages are the need of multiple density evaluations and the loss of

an analytic formulation. The Jacobian of the dynamics, which is important for the integration of the phase space density, cannot be inferred analytically but requires yet another quadrature.

King-Hele method Here, only an overview of the formulation is given. The treatment of the full theory can be found in King-Hele (1964). The mathematical formulas are re-derived and extended in Appendix B.3. The idea is to perform a series expansion of the integrand in Equation (1.31) in a suitable variable, truncate at a given degree and find analytical solutions for each integral of the expanded terms. With the assumption that ρ decreases strictly exponentially with altitude, i.e. with a fixed H, each expanded integrand can be represented by the modified Bessel function of the first kind, I_n, which for $n \in \mathbb{N}_0$ is given as (Abramowitz and Stegun, 1972, p. 376)

$$I_n(x) = \frac{1}{\pi} \int_0^\pi \exp(x \cos\theta) \cos(n\theta) d\theta \tag{1.40}$$

The KH formulation enables fast propagation as it can be evaluated analytically and requires only a single density evaluation for each computation of the rate of change. The main problem with the fixed H assumption is the underestimation of ρ at altitudes above the perigee altitude, h_p, which for eccentric orbits can induce large errors exceeding the uncertainties inherent in atmospheric modelling (see Section 3.4). This is true for any object in a non-circular orbit subject to a non-strictly exponentially decaying atmosphere. KH was aware of this problem and suggested a way to calculate the contraction of an orbit with a varying scale height (see King-Hele, 1964, Chapter 6). To keep the equations analytically integrable, he approximates the varying H linearly, with a constant slope parameter. Linear approximation of the true H is valid only locally. For low eccentric orbit configurations this might be sufficient, but high eccentricities will re-introduce the errors. Using a constant slope parameter will thus lead to a new over- or underestimation of the drag depending on the eccentricity, e.

1.3.5 Interpolation and surrogate modelling

Probabilistic propagation of an initial density, using the method of the characteristic, enables the exact evaluation of the density along the characteristic curve. However, no density estimate is available outside of the characteristic curve. Backpropagation could be employed for each point of interest. Equation (1.13) shows that the trajectory of a characteristic is independent of its density, n. Hence, a single backpropagation would suffice to obtain the density value at the epoch of interest. However, if the density is required over the full phase space, many backpropagations are necessary.

Instead, the density can efficiently be estimated from the scattered set of forward propagated characteristics. Wittig et al. (2017) employ differential

algebra to find an approximation of the density around the characteristics. Nevertheless, the solution is only valid in the vicinity of each characteristic. Halder and Bhattacharya (2011) estimate the density outside of the characteristic curves by discretising the phase space. Instead of simply counting the samples, as in Equation (1.22), the average over the characteristics present in the bin is taken to represent the density over the bin. Still, discretisation leads to non-smooth, inaccurate density estimates requiring a large set of characteristics. Trisolini and Colombo (2019) employ Delaunay triangulation to interpolate in between the characteristic points. In Delaunay triangulation, the function is interpolated from D-simplexes, defined by the characteristics (Lee and Schachter, 1980). The advantage is that the estimated function, evaluated at the interpolation points, remains exact. However, the resulting interpolation is non-smooth over the simplexes, extrapolation of the density is not feasible and careful treatment of non-convex distributions is required.

Approximation of the output of a function, given its evaluations on a scattered set of points, can also be obtained through surrogate modelling (e.g. Forrester et al., 2008). A surrogate model approximates a function of interest, allowing significantly faster evaluation of the approximated function, while maintaining sufficient accuracy. Any interpolation scheme, such as Delaunay triangulation, can be thought of as a surrogate model. A short list of surrogate modelling techniques is given here, allowing to estimate densities from scatter samples, with or without information about n. Selection of a method applicable to the problem of interpolating the characteristics of a fragmenting cloud is elaborated and justified in Section 4.1.

The density estimation methods can be categorised into samples-based and function-based methods, and each of the two categories can further be divided into parametric and non-parametric methods. Samples-based methods work with N_s samples

$$\boldsymbol{X} = \{\boldsymbol{x}_i\}_{i=1,\dots,N_s} \tag{1.41}$$

that do not carry information about the function itself, i.e. can be obtained through Monte Carlo sampling. They can only be used to approximate probability distribution functions. Instead, function-based methods approximate any function through interpolation of samples carrying information about the response, y, i.e. the training data is

$$\boldsymbol{Y} = \{(\boldsymbol{x}_i, y_i)\}_{i=1,\dots,N_s} \tag{1.42}$$

Parametric methods prescribe the functional form of the response function, i.e. the fitting is reduced to the selection of optimal parameters obtained through regression. The advantage of parametric methods is the near instantaneous evaluation of the fitted surface, making it suitable for post-processing steps such as the evaluation of the spatial density or collision

probability. Disadvantages are the loss of accuracy as residual errors are introduced even at the points of fitting, and potentially large discrepancies between estimate and underlying function if an unsuitable model is chosen.

Non-parametric methods estimate the function using directly the samples and are thus more flexible in approximating any underlying distribution. However, they generally require many samples. A promising method to obtain large sample sets without the added propagation effort is polynomial chaos expansion (Wiener, 1938; Xiu and Karniadakis, 2002). Sampling of such a surrogate model enables efficient sample generation – given any initial distribution – at the desired target space. Hence, it can be used to improve any samples-based method for the estimation of the resulting distribution, at a drastically reduced propagation cost (e.g. Jones et al., 2015; Vittaldev et al., 2016). Still, it is hard to interpret a multivariate distribution from samples only, where visual inspection becomes intractable. An option is to fall back on the mean and sample covariance, which, however, do not tell the full picture. E.g. for a banana-shaped distribution or a density with multiple peaks, the mean does not give an accurate picture of the distribution. Indeed, the initial as well as the propagated fragmentation cloud distribution does not resemble normally distributed fragments (see Chapter 6). Another drawback of samples-based methods is that all observations are required for each function evaluation.

An example of parametric, samples-based estimation methods is the Expectation Maximisation (EM) algorithm (Bishop, 2006, Chapter 9.2). This is an iterative method to maximise the likelihood function of a statistical model with respect to its parameters. Examples of non-parametric, samples-based estimators are the binning of samples (see Equation (1.22)), or Kernel Density Estimation (KDE) (Bishop, 2006, Chapter 2.5). KDE approximates the probability distribution function of random variable, $x \in \mathbb{R}$, as

$$\hat{p}_x(x) = \frac{1}{N_s} \sum_{i=1}^{N_s} k(x, x_i) \tag{1.43}$$

given a kernel, k, which itself can again be parametric or non-parametric. A common selection of k is the Gaussian kernel

$$k(x, x_i) = \frac{1}{\sqrt{2\pi\beta^2}} \exp\left(-\frac{(x - x_i)^2}{2\beta^2}\right) \tag{1.44}$$

with the bandwidth, β. The larger β, the smoother is the estimated distribution.

Examples for function-based, parametric methods are polynomial regression or fitting of a Gaussian Mixture Model (GMM). The latter is elaborated in more detail in Chapter 4. Non-parametric, function-based estimation methods are more flexible in fitting to any underlying distribution. Examples are Nearest Neighbour (NN) methods, Radial Basis Function (RBF)

interpolation and Gaussian Process (GP) regression. NN takes the average value of its k nearest neighbours (Bishop, 2006, Chapter 2.5). RBF interpolation is performed by evaluating (Forrester et al., 2008, Chapter 2.3)

$$\hat{y}(x) = \sum_{i=1}^{N_s} w_i k(||x - x_i||) \tag{1.45}$$

where, as for KDE, k is the kernel, and the weights, $\boldsymbol{w} = \{w_i\}_{i=1,\dots,N_s}$, are found through

$$\boldsymbol{w} = \boldsymbol{K}^{-1}\boldsymbol{y} \tag{1.46}$$

given the observations, $\boldsymbol{y} = \{y_i\}_{i=1,\dots,N_s}$, and the kernel evaluated for each pair of training samples

$$K_{i,j} = k(||x_i - x_j||) \quad i,j = 1,\dots,N_s \tag{1.47}$$

Among the possible variations of k are the non-parametric, linear kernel

$$k(r) = r \tag{1.48}$$

the parametric, multiquadric kernel

$$k(r) = \sqrt{(r/\epsilon)^2 + 1} \tag{1.49}$$

or the parametric, Gaussian kernel

$$k(r) = \exp\left(-\frac{r^2}{2\beta^2}\right) \tag{1.50}$$

where ϵ and β are hyperparameters.

A GP is an infinite-dimensional object, defined by a set of random variables. A short recap is given here, following (Rasmussen and Williams, 2005, Chapter 2). The notation of a GP is

$$g(x) \sim \mathcal{GP}(m(x), k(x, x^*)) \tag{1.51}$$

with the mean function, $m(x)$, and covariance function, $k(x, x')$, of the function to be interpolated, $g(x)$, as

$$m(x) = \mathbb{E}\left[g(x)\right] \tag{1.52a}$$
$$k(x, x') = \mathbb{E}\left[(g(x) - m(x))(g(x') - m(x'))\right] \tag{1.52b}$$

As the characteristics carry the true value of the density to be interpolated, the discussion here is restricted to prediction with noise-free observations. GP regression makes assumptions on the outputs of the function, $\boldsymbol{y}$, via the prior distribution

$$\boldsymbol{y} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X})) \tag{1.53}$$

where $\boldsymbol{K}$ is again the matrix of the covariance function evaluated at each pair of $\boldsymbol{X}$. A zero mean is chosen, together with a non-periodic kernel (see below), as the approximation of the function at data points far from the observations should vanish. Then, if the function values evaluated at the observations, $\boldsymbol{y} = \boldsymbol{g}(\boldsymbol{X})$, and at the test points, $\boldsymbol{y}^* = \boldsymbol{g}(\boldsymbol{X}^*)$, are jointly Gaussian, the joint distribution is

$$\begin{pmatrix} \boldsymbol{y} \\ \boldsymbol{y}^* \end{pmatrix} = \mathcal{N}\left(\boldsymbol{0}, \begin{bmatrix} \boldsymbol{K}(\boldsymbol{X},\boldsymbol{X}) & \boldsymbol{K}(\boldsymbol{X},\boldsymbol{X}^*) \\ \boldsymbol{K}(\boldsymbol{X}^*,\boldsymbol{X}) & \boldsymbol{K}(\boldsymbol{X}^*,\boldsymbol{X}^*) \end{bmatrix}\right) \tag{1.54}$$

Finally, a statistical description of $\boldsymbol{y}^*$, the posterior distribution, can be found as the conditional probability

$$\boldsymbol{y}^*(\boldsymbol{X}^*|\boldsymbol{X},\boldsymbol{y}) \sim \mathcal{N}\left(\boldsymbol{\mu}_K, \boldsymbol{\sigma}_K^2\right) \tag{1.55}$$

with

$$\boldsymbol{\mu}_K = \boldsymbol{K}(\boldsymbol{X}^*,\boldsymbol{X})\boldsymbol{K}^{-1}(\boldsymbol{X},\boldsymbol{X})\boldsymbol{y} \tag{1.56a}$$

$$\boldsymbol{\sigma}_K^2 = \boldsymbol{K}(\boldsymbol{X}^*,\boldsymbol{X}^*) - \boldsymbol{K}(\boldsymbol{X}^*,\boldsymbol{X})\boldsymbol{K}^{-1}(\boldsymbol{X},\boldsymbol{X})\boldsymbol{K}(\boldsymbol{X},\boldsymbol{X}^*) \tag{1.56b}$$

For interpolation, the mean is taken, i.e. $\boldsymbol{y}^* = \boldsymbol{\mu}_K$. A common kernel is the squared-exponential covariance function (also known as radial basis function)

$$k(x,x') = \sigma_s \exp\left(-\frac{(x-x')^2}{2\beta^2}\right) \tag{1.57}$$

with the length scale, β, and the signal variance, σ_s^2, two hyperparameters that need to be optimised. For the evaluation of the mean, only the length scale is important, having a similar effect on the smoothness of the resulting interpolation as the bandwidth in KDE.

1.3.6 Summary

A large pool of literature and techniques is available for the propagation of space debris and the evaluation of the collision probability. Probabilistic evolutionary models allow to directly map the density of a continuum, making their evaluation time independent of the actual number of fragments the continuum represents. Analytical solutions permit fast evaluations of the density at any point in time. However, they are based on simplified orbital geometries and require simple force models. Numerical solutions, instead, can overcome these constraints but find the density only along the characteristics of the dynamics. This drawback can be addressed by interpolating the scattered set of characteristics. By training a surrogate modelling, an estimate of the density is available over the full domain. Methods based on finite differences inherently give a solution across the full domain and for

any force model. Yet, they require computational power that restricts them from venturing into phases spaces with dimensionality higher than three.

Deterministic evolutionary models can consider any force model and any dimensionality making them useful for many applications. Today, they are the standard when it comes to defining new mitigation guidelines. Nonetheless, the large number of objects that need to be propagated to obtain statistically relevant results makes them computationally expensive. Additionally, low probability events cannot accurately be studied using deterministic models. Hence, such models are generally used globally to study the evolution of single-dimensional quantities, such as the number of objects or collisions, and are usually not used for the estimation of the ramifications of a single satellite breakup.

The collision probability estimation tools employed in evolutionary models suffer from the shortcomings of the density propagation tools. Simplifications made during propagation, e.g. enforcing circular orbits or discretising the phase space, are applied also for the evaluation of the collision probability. If, however, the phase space density as a function of position and velocity can accurately be estimated, easy derivation of an accurate collision probability follows.

Lastly, semi-analytical propagation increases the speed of the integration of the trajectories. Present techniques considering drag are either dependent on numerical quadrature or – to find analytical solutions of the integral – on simplified atmosphere models that are not valid for eccentric orbits. No analytical solution of the averaging integrals is available for atmospheric density profiles that do not decay strictly exponentially in altitude.

1.4 Contribution

The contribution of this doctoral thesis is presented in two sections. First, novel techniques and software developed during the doctoral studies are discussed in Section 1.4.1. The publications and presentations related to these studies are summarised in Section 1.4.2.

1.4.1 Novel techniques and software

The proposed method to model the evolution of a fragmentation cloud is fully probabilistic, based on the semi-analytical propagation of the characteristics and the training of surrogate models. Due to its probabilistic nature, the evolution of the fragment cloud is accurately predicted even for low probability events, leading to an order of magnitude speed increase if compared to deterministic models set for a comparable sensitivity. The semi-analytical application of the method of characteristics allows to propagate the continuity equation given any orbit geometry, force model or dimension.

This constitutes a considerable improvement over methods based on finite differences, which cannot account for higher dimensions, or analytical solutions with their restricted applicability. The surrogate model is used to find an estimate of the fragment phase space density over the full domain, thus removing the need for additional propagations. The model is subsequently used for post-processing tasks such as the estimation of the fragment spatial density or the collision probability.

At the beginning of every virtual fragmentation cloud is a model of the fragment distribution. Being the *de facto* standard among space agencies, the NASA SBM (Johnson et al., 2001) is utilised for obtaining the initial conditions for the examples shown herein. The SBM was designed to be sampled for representative objects, i.e. the samples come without a density value. For application in the probabilistic cloud propagation, sampling of the characteristics is required. Hence, an extended form of the SBM, reformulated as a probability distribution function, is developed and presented in this work. The probabilistic approach to the breakup model also allows to calculate estimation values, e.g. of the fragment mass, which can be used to enforce conservation of mass during a fragmentation. The transformation into various subsets of Keplerian elements enables the application of the method in various combinations of phase space variables. Representing the fragmentation distribution in Keplerian elements facilitates the incorporation of assumptions on the distribution of the fragments, such as randomisation of the mean anomaly. Hence, the approach is well suited to model fragmentations over various timespans.

The fitting of the surrogate model, a Gaussian Mixture Model (GMM), is performed in any user-defined phase space, instead of fitting directly in orbital elements. The phase space variables can be defined through symbolic mathematical expressions as a function of the propagated states. This permits a normalisation of the phase space, aiding the fitting of the surrogate model and allowing to explore a large combination of states and derived phase spaces without requiring manual derivations needed for density mappings. Still, the non-linearities that are inherent in orbital dynamics potentially lead to phase space distributions that are hard to represent with GMMs. A new framework, called reference hypersurfaces is introduced to further normalise such distributions, thereby absorbing some of the non-linearities. Through transformation of the fitted phase space density, the density in the phase space of origin can be recovered easily.

The estimation of the collision probability is performed considering not only the spatial density, but also the relative impact velocity and a variable cross-sectional area. The integration over all the incident velocities is performed directly in Keplerian elements, simplifying the selection of the integration limits. The collision probability is estimated along the orbit of a target object, without the need to define finite volumes, increasing its

fidelity.

Furthermore, this thesis introduces a new analytical method to calculate the effect of atmospheric drag on the trajectory of the objects. It is based on the introduction of a new smooth exponential model that accurately catches the density profile of Earth's atmosphere. The analytical solution permits application of semi-analytical propagation, speeding up the integration of trajectories.

Next to presenting the techniques, the *Starling* software suite is introduced. It represents an implementation of the developed methods presented in this thesis. Making use of parallel programming techniques, it propagates density distributions in a few minutes only. Its open structure facilitates the integration of any initial condition and dynamics and as such is a generic partial differential equation solver. The software suite is used to propagate the evolution of the fragmentation cloud for various scenarios, covering different eccentricities and force models. The found distribution is then subsequently used to estimate the collision probability of its fragments with Sentinel-1A and two other target objects.

1.4.2 Publications and outreach

The following journal publications and conference proceedings are directly related to the research presented in this thesis. As were the following seminars and other presentations (excluding conferences) given during the time of the doctoral studies.

Journal publications

- S. Frey, C. Colombo, and S. Lemmens. Extension of the King-Hele orbit contraction method for accurate, semi-analytical propagation of non-circular orbits. *Advances in Space Research*, 64:1–17, 2019. doi:10.1016/j.asr.2019.03.016.

  As outlined in Section 1.3.4, the analytical King-Hele method does not accurately model the drag forces for highly eccentric orbits. This paper presents an extension to the method, maintaining a closed-form solution of the averaged dynamics while removing the inaccuracies due to eccentric orbits.

- S. Frey and C. Colombo. Transformation of satellite breakup distribution for probabilistic orbital collision hazard analysis. *Journal of Guidance, Control, and Dynamics*, accepted for publication in July, 2020. doi:10.2514/1.G004939

  This paper provides a framework for the fully probabilistic treatment of orbital fragmentation clouds and their effects on other space ob-

jects in terms of collision probability, while leaving the selection of the uncertainty propagation method up to the reader.

- S. Frey, C. Colombo, and S. Lemmens. Evolution of orbital fragmentation cloud via continuum and surrogate modelling. To be submitted to the *Journal of Guidance, Control, and Dynamics* in September, 2020.

  Finally, this paper discusses the application of the framework, combined with the propagation of the characteristics and the surrogate modelling, to estimate the evolution of orbital fragmentation clouds.

Conference proceedings

- S. Frey, C. Colombo, S. Lemmens, and H. Krag. Evolution of fragmentation cloud in highly eccentric orbit using representative objects. In *Proceedings of the 68th International Astronautical Congress*, 2017

- S. Frey, C. Colombo, and S. Lemmens. Evolution of fragmentation cloud in highly eccentric Earth orbits through continuum modelling, In *Proceedings of the 69th International Astronautical Congress*, 2018.

- S. Frey, C. Colombo, and S. Lemmens. Interpolation and integration of phase space density for estimation of fragmentation cloud distribution, In *Proceedings of the 29th AAS/AIAA Space Flight Mechanics Meeting*, 2019

- S. Frey, C. Colombo, and S. Lemmens. Application of density-based propagation to fragment clouds using the Starling suite, In *Proceedings of the 1st International Orbital Debris Conference*, 2019.

Seminars

- S. Frey and C. Colombo. Orbit evolution of space debris: modelling and applications. At the *Tsinghua University*, Beijing, China, July 2018.

- S. Frey and C. Colombo. Space debris continuum modelling. At the *University of Texas*, Austin, USA, December 2019.

Other presentations

- S. Frey, C. Colombo, and S. Lemmens. Superimposition of the atmosphere density for fast and accurate semi-analytical propagation. At the *Key Topics in Orbit Propagation Applied to Space Situational Awareness (KePASSA) workshop*, Noordwijk, The Netherlands, July 2017.

- S. Frey and C. Colombo. Evolution of a fragment cloud in highly eccentric orbit using representative objects. At the *Research Summer School for Satellite Dynamics and Space Missions (SDSM): Theory and Applications of Celestial Mechanics*, San Martino al Cimino, Italy, September 2017.

- S. Frey, I. Gkolias, and C. Colombo. Design of end-of-life of disposal manoeuvres and re-entry modelling with PlanODyn. At the 7^{th} *European Workshop on Satellites End-of-Life*, Paris, France, January 2018.

- S. Frey, C. Colombo, D. Gondelbach, and R. Armellin. Extension of the King-Hele orbital contraction method and application to the geostationary transfer orbit re-entry prediction. At the 4^{th} *International Workshop on Space Debris Re-entry*, Darmstadt, Germany, March 2018.

- S. Frey, C. Colombo, and S. Lemmens. Density based modelling and indication of break-up location and epoch from fragments using backwards propagation. At the 5^{th} *European Workshop on Space Debris Modelling and Remediation*, Paris, France, June 2018.

- S. Frey, C. Colombo, and S. Lemmens. Density based method for space debris applications. At the *ReDSHIFT Space Debris Day*, Milan, Italy, October 2018.

1.5 Structure of thesis

This dissertation is, next to the introduction chapter, divided into six chapters. The next three chapters are dedicated to the developed theory, and the remaining three chapters are dedicated to the software implementation, applications, and conclusions, respectively.

Chapter 2 introduces the continuum formulation. It explains how phase space densities given in Cartesian coordinates can be converted into orbital elements and vice versa. This is important as the initial distributions and the propagation are not given in the same phase space. The concept of randomisation for mid- to long-term propagation is revisited and extended to obtain spatial densities and the number of impacts from distributions defined in a subset of orbital elements only. Then, the extension of the NASA SBM to obtain a probabilistic initial breakup distribution is discussed. Additionally, two examples of fragmentation distributions are given.

Chapter 3 discusses the propagation of the continuum via the continuity equation and the method of characteristics. It shows how initial characteristics are selected and integrated. The orbit propagator suite and the force

model used in this work are introduced. Additionally, it contains the description of the smooth exponential atmosphere and the superimposed King-Hele method developed in this research work.

Chapter 4 explains the interpolation technique used to obtain an estimate of the density across the full phase space. The surrogate model is introduced, and its fitting is discussed. Furthermore, normalisation of the phase space and the concept of reference hypersurfaces are introduced helping in finding an optimal density approximation.

The implementation of the theory in a software suite is presented in Chapter 5. The structure and interfaces are discussed and its applicability to other partial differential equations is elaborated. The chapter further contains a validation part where the method is compared to Monte Carlo sampling and another validation method that is applicable to phase spaces with any dimensions.

Chapter 6 shows applications of the proposed method and the software suite. Fragmentation clouds from collision and explosion scenarios are propagated in various orbital regimes considering $3-6$ dimensional phase spaces. The collision probability of one of the clouds is calculated for different target objects. Examples of how to apply the theory hopefully make it more appealing to the reader.

Finally, Chapter 7 concludes this dissertation with a summary of the work, the limitations of the method and potential future work.

2 Continuum formulation

The space debris cloud is modelled in this work as a continuum of fragments. In this chapter, the mathematical description of such a continuum formulation is discussed in Section 2.1. Section 2.2 introduces means of transforming densities into phase spaces of same dimensionality and of lower dimensionality. In Section 2.3 randomisation of some of the orbital elements is applied to obtain the full phase space density from densities defined in lower dimensions. Section 2.4 explains how to derive the spatial density and the number of impacts from a given phase space density. The derivation of the initial cloud distribution is shown in Section 2.5 and depicted in the form of an example in Section 2.6. Additionally, Section 2.6 shows an example of evaluating the spatial density directly from a distribution given in Keplerian elements.

This chapter does not yet introduce the propagation of the continuum, which is discussed, instead, in Chapter 3. Note that parts of this chapter are adapted from (Frey and Colombo, 2020)[1].

2.1 Formulation

The position of a particle in the phase space, $\boldsymbol{x} \in \mathbb{R}^D$, in dimension, D, is probabilistically described using a probability density function, $p_{\boldsymbol{x}} = p(\boldsymbol{x}, t)$. The dependence on time, t, is dropped from the notation, as $p_{\boldsymbol{x}}$ is studied at fixed epochs of interest. The integral of $p_{\boldsymbol{x}}$ over the full phase space is unity, i.e.

$$\int_{\mathbb{R}^D} p_{\boldsymbol{x}} \, \mathrm{d}^D \boldsymbol{x} = 1 \tag{2.1}$$

Similarly, the phase space density distribution, $n_{\boldsymbol{x}} = n(\boldsymbol{x}, t)$, describes the distribution of the total number of fragments, N. If the particles within the

[1] S. Frey and C. Colombo. Transformation of satellite breakup distribution for probabilistic orbital collision hazard analysis. *Journal of Guidance, Control, and Dynamics*, accepted for publication in July, 2020. doi:10.2514/1.G004939

cloud are independently described by $p_{\boldsymbol{x}}$, the two distributions are connected through

$$n_{\boldsymbol{x}} = N p_{\boldsymbol{x}} \tag{2.2}$$

Hence, the integral of $n_{\boldsymbol{x}}$ over the full phase space results in

$$\int_{\mathbb{R}^D} n_{\boldsymbol{x}} \, \mathrm{d}^D \boldsymbol{x} = N \tag{2.3}$$

The applicability of the theory in this thesis is valid for both, probability density functions and phase space density distributions. Any result simply needs to be weighted with N to convert either into the other. Hence, uncertainties described as probability distribution functions can be propagated as well as fragment clouds.

Why is a probabilistic formulation so much more powerful than simply deriving the density estimate from random samples? Assume a multivariate normal distribution, $\mathcal{N}$, given in D dimensions as (Bishop, 2006)

$$p_{\mathcal{N}} = \frac{1}{(2\pi)^{\frac{D}{2}} \det(\boldsymbol{\Sigma})^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{x}-\boldsymbol{\mu})\right) \tag{2.4}$$

with the mean, $\boldsymbol{\mu} \in \mathbb{R}^D$, and the covariance, $\boldsymbol{\Sigma} \in \mathbb{R}^{D\times D}$. The density value can be directly obtained through evaluation of Equation (2.4), corresponding to the probabilistic approach. Alternatively, it can be estimated as

$$\hat{p} = \frac{N_s(V)}{N_s V} \tag{2.5}$$

by sampling N_s instances from the distribution and counting the number of samples, $N_s(V)$, present in the volume of interest, V, corresponding to the deterministic approach. Suppose the following statistics are selected

$$\boldsymbol{\mu} = \boldsymbol{0} \qquad \boldsymbol{\Sigma} = \boldsymbol{I} \tag{2.6}$$

with the identity matrix, $\boldsymbol{I}$, and the density is evaluated around the bin centre, $\hat{\boldsymbol{x}}$, set at

$$\hat{x}_j = D^{-\frac{1}{2}} \quad \forall j \in [1, \ldots, D] \tag{2.7}$$

chosen such that the Mahalanobis distance, defined as (Bishop, 2006, Chapter 2.3)

$$d_M(\boldsymbol{x}) = \sqrt{(\boldsymbol{x}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{x}-\boldsymbol{\mu})} \tag{2.8}$$

is unity. As the covariance matrix is the identity matrix, d_M simply describes the distance from $\boldsymbol{\mu}$. The bin, centred around $\hat{\boldsymbol{x}}$, is selected to have a fixed width, w, in each direction of $w = 0.05$ and thus a volume of $V = w^D$. For the one-dimensional case, this corresponds to a grid of 120 bins over $x \pm 3$, covering 99.7% of the total probability. For the five-dimensional case,

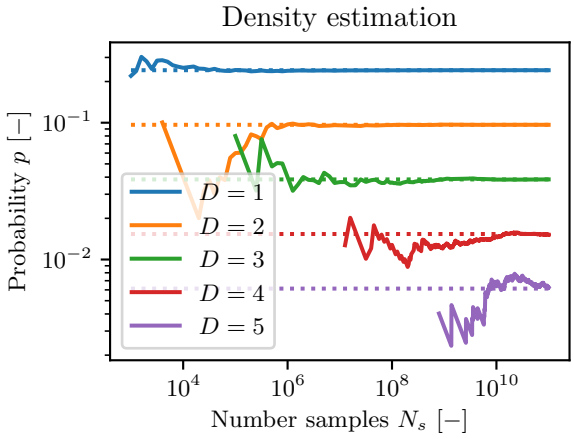


Figure 2.1: Convergence for phase space density estimate with increasing number of samples, N_s, for different dimension of the domain, D. The solid and dotted lines represent the density estimates, $\hat{p}$, and exact function evaluations of $p_{\mathcal{N}}$, respectively.

this corresponds to a regular grid of 40 bins over $x_j \pm 1$ in each dimension $j = 1, \ldots, 5$, covering, again, 99.7% of the total probability.

As can be seen in Figure 2.1, relatively few samples are required to accurately estimate the density for one-dimensional distributions. However, with increasing dimension, corresponding to an increase of number of bins, the density estimate becomes inefficient as many samples are required. For an accurate estimate, the number of samples needs to be $2 - 3$ orders of magnitude larger than the number of bins required to cover most of the distribution. E.g. in two dimensions, corresponding to $76^2 = 5776$ bins to cover 99.7% within $x_{1/2} \pm 1.9$, it requires 10^6 samples for convergence of the estimate with the real probability. For four dimensions and $48^4 = 5308416$ bins to cover 99.7% within $x_j \pm 1.2$ for $j = 1, \ldots, 4$, it requires $10^9 - 10^{10}$ samples.

Note that, from $D \geq 2$, the bulk of the distribution is found outside of the D-sphere with radius $||\hat{\boldsymbol{x}}||$, $S^D = \{\boldsymbol{x} \in \mathbb{R}^D : ||\boldsymbol{x}|| = ||\hat{\boldsymbol{x}}||\}$. E.g. for the three-dimensional case, more than 80% of the probability is found outside of the 3-sphere. For the five-dimensional case, this share increases to more than 96%. If the bin is positioned farther away from the peak of the distribution, where the probability is lower, even more samples would be required to find a decent estimate of the underlying distribution.

Propagation of many deterministic fragments is computationally expensive and inaccurate. Software suits based on Monte Carlo simulations generally propagate about $20000 - 30000$ individual objects per run only, without considering uncertainties despite propagation over hundreds of years (IADC,

2013a). That is not sufficient to accurately estimate the spatial density of the fragments on a fine three-dimensional grid as the volume of each bin vanishes, let alone directional quantities in six dimensions. Instead, direct evaluation of Equation (2.4) is exact and nearly instantaneous even for high D.

2.2 Transformation

Generally, initial distribution for orbital fragmentations are given in Cartesian coordinates. To profit from the fast propagation enabled through averaged dynamics, these distributions need to be transformed into orbital elements. Herein, different methods to perform density transformations are discussed in general and applied to transformations between Cartesian coordinates and Keplerian elements. Transformations into Keplerian elements are discussed here due to their widespread use and physical interpretability of the elements. However, the theory is equally applicable to any orbital element set. Note that for all the transformations described in this section, dependencies on variables not describing the position and velocity, i.e. the area-to-mass ratio, are dropped as they are not affected by the change of frame.

2.2.1 Methods

Two transformation methods are discussed here; one based on the change of variables and the other based on the Dirac generalised function. The former is suited for transformations involving phase spaces with equal dimensions. The latter is useful for marginalisation, i.e. the reduction of distributions from higher to lower-dimensional spaces.

Change of variables

The transformation through the change of variables method is already introduced in Section 1.3.1 to describe the mapping of densities over time. Here, it is introduced again for the application of transformation between phase spaces at the same instance of time.

Given a transformation $\boldsymbol{y} = \boldsymbol{\varphi}(\boldsymbol{x})$, $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^D$, $\boldsymbol{\varphi}$ being invertible, surjective and differentiable, the density $p_{\boldsymbol{y}}$ can be derived from $p_{\boldsymbol{x}}$ by integrating over an infinitesimal volume

$$p_{\boldsymbol{y}} \, \mathrm{d}V_{\boldsymbol{y}} = p_{\boldsymbol{x}} \, \mathrm{d}V_{\boldsymbol{x}} \tag{2.9}$$

considering that the integral is invariant under transformation and using integration by substitution, such that (Soong, 2004, Chapter 5)

$$p_{\boldsymbol{y}}(\boldsymbol{y}) = \frac{p_{\boldsymbol{x}}(\boldsymbol{\varphi}^{-1}(\boldsymbol{y}))}{|\det \boldsymbol{J}|} \tag{2.10}$$

with the Jacobian, $\boldsymbol{J} \in \mathbb{R}^{D\times D}$, defined as

$$J_{ij} = \frac{\partial \varphi_i}{\partial x_j} \tag{2.11}$$

The assumption on $\boldsymbol{\varphi}$ being invertible can be relaxed. If an output $\boldsymbol{y}$ permits multiple different inputs $\boldsymbol{x}_k$, the transformed probability is the sum of all the possible inputs

$$p_{\boldsymbol{y}}(\boldsymbol{y}) = \sum_k \frac{p_{\boldsymbol{x}}(\boldsymbol{x}_k)}{|\det \boldsymbol{J}_k|} \tag{2.12}$$

Transformations of densities for $\boldsymbol{\varphi} : \mathbb{R}^{D_x} \to \mathbb{R}^{D_y}$ with $D_x > D_y$ using the change of variables approach involves integration over joint probabilities to find the marginal probability. Such joint probabilities cannot always be found.

Dirac generalised function

Alternatively, the variables can be transformed using the Dirac generalised function (Au and Tam, 1999). Given a continuous transformation $\boldsymbol{y} = \boldsymbol{\varphi}(\boldsymbol{x})$, with $\boldsymbol{\varphi} : \mathbb{R}^{D_x} \to \mathbb{R}^{D_y}$ and $D_x > D_y$, the distribution function in $\boldsymbol{y}$ can be found by integrating over $\mathbb{R}^{D_x}$ as

$$p_{\boldsymbol{y}} = \int_{\mathbb{R}^{D_x}} p_{\boldsymbol{x}}\, \delta[\boldsymbol{\varphi}(\boldsymbol{x}) - \boldsymbol{y}]\, \mathrm{d}^{D_x}\boldsymbol{x} \tag{2.13a}$$

$$= \int_{\mathbb{R}^{D_x}} p_{\boldsymbol{x}}\, \delta[\varphi_1(\boldsymbol{x}) - y_1] \dots\, \delta[\varphi_{D_y}(\boldsymbol{x}) - y_{D_y}]\, \mathrm{d}x_1 \dots \mathrm{d}x_{D_x} \tag{2.13b}$$

with the Dirac delta function, δ, which has the following properties

$$\delta(x) = \begin{cases} 1, & \text{if } x = 0 \\ 0, & \text{otherwise} \end{cases} \tag{2.14a}$$

$$f(a) = \int_{-\infty}^{\infty} f(x)\delta(x-a)\,\mathrm{d}x \tag{2.14b}$$

$$\delta\left(g(x)\right) = \sum_i \left|\frac{\partial g}{\partial x}(x_i)\right|^{-1} \delta(x - x_i) \qquad g(x_i) = 0,\ \frac{\partial g}{\partial x}(x_i) \neq 0 \tag{2.14c}$$

Thus, the evaluation requires the roots of $\boldsymbol{\varphi}(\boldsymbol{x}) - \boldsymbol{y} = 0$. This method intrinsically deals with non-invertible functions. Each root in the Dirac generalised function corresponds to a possible instance in $\boldsymbol{x}_k$ that is transformed into $\boldsymbol{y}$. An example of application of the Dirac generalised function is given in Appendix A.2.

Both the transformation methods, through change of variables and the Dirac generalised function, give the same result. For the case where no

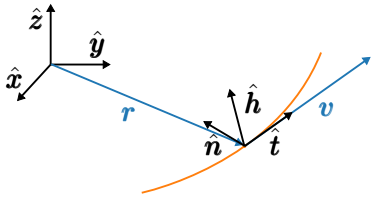


Figure 2.2: Satellite frame, $\mathcal{T} = \{\hat{\boldsymbol{t}}, \hat{\boldsymbol{n}}, \hat{\boldsymbol{h}}\}$, relative to the inertial frame, $\mathcal{I} = \{\hat{\boldsymbol{x}}, \hat{\boldsymbol{y}}, \hat{\boldsymbol{z}}\}$.

marginalisation is required, i.e. $D_x = D_y$, they collapse to the same formulation. Still, the transformation in Equation (2.13) is useful when integration over the space of origin is more convenient than integration over the target space.

2.2.2 Frames

For the derivation of the transformation between Cartesian coordinates and Keplerian elements, two different frames are required, which are described herein and depicted in Figure 2.2.

Inertial frame

The inertial frame, $\mathcal{I} \in \{\hat{\boldsymbol{x}}, \hat{\boldsymbol{y}}, \hat{\boldsymbol{z}}\}$, is centred at the central body, and is independent of the orbit configuration. The Keplerian elements, $\boldsymbol{\alpha}$

$$\boldsymbol{\alpha} = (a,\ e,\ i,\ \Omega,\ \omega,\ f)^T \tag{2.15}$$

consisting of semi-major axis, a, eccentricity, e, inclination, i, right ascension of ascending node, Ω, argument of perigee, ω, and true anomaly, f, are defined with respect to $\mathcal{I}$.

The elements in $\boldsymbol{\alpha}$, at a given point in time, can be found from a state in Cartesian coordinates described in any frame, $\boldsymbol{s} = (\boldsymbol{r}^T\ \boldsymbol{v}^T)^T$, with the orbital radial vector, $\boldsymbol{r} \in \mathbb{R}^3$, and the orbital velocity vector, $\boldsymbol{v} \in \mathbb{R}^3$. The function relating the two sets, $\boldsymbol{\alpha} = \boldsymbol{\varphi}_{\boldsymbol{\alpha s}}(\boldsymbol{s})$, can be found in literature (e.g. Vallado, 2013, Chapter 2.5).

Satellite frame

The satellite frame, $\mathcal{T} \in \{\hat{\boldsymbol{t}}, \hat{\boldsymbol{n}}, \hat{\boldsymbol{h}}\}$, is centred at the orbiting body, with its primary $\hat{\boldsymbol{t}}$-axis aligned with the orbital velocity vector, $\boldsymbol{v}$. The tertiary $\hat{\boldsymbol{h}}$-axis is aligned with the orbit angular momentum, and the secondary $\hat{\boldsymbol{n}}$-axis, lying in the orbital plane pointing inwards, completes the coordinate system. Thus, the rotation matrix from $\mathcal{T}$ to $\mathcal{I}$ is defined as

$$\boldsymbol{A}_{\mathcal{I}\mathcal{T}} = \left[\hat{\boldsymbol{t}}_{\mathcal{I}}\ \hat{\boldsymbol{n}}_{\mathcal{I}}\ \hat{\boldsymbol{h}}_{\mathcal{I}}\right] \tag{2.16}$$

with

$$\hat{\boldsymbol{t}}_{\mathcal{I}} = \frac{\boldsymbol{v}_{\mathcal{I}}}{||\boldsymbol{v}||} \qquad \hat{\boldsymbol{h}}_{\mathcal{I}} = \frac{\boldsymbol{r}_{\mathcal{I}} \times \boldsymbol{v}_{\mathcal{I}}}{||\boldsymbol{r} \times \boldsymbol{v}||} \qquad \hat{\boldsymbol{n}}_{\mathcal{I}} = \hat{\boldsymbol{h}}_{\mathcal{I}} \times \hat{\boldsymbol{t}}_{\mathcal{I}} \tag{2.17}$$

2.2.3 Cartesian coordinates from/to Keplerian elements

If the partial derivatives of a transformation between two spaces of the same dimension are known, the transformation of the density function can be obtained through application of Equation (2.10). It is convenient to work in the $\mathcal{T}$-frame if impulses are to be added, as most manoeuvres are executed relative to $\boldsymbol{v}$. The Jacobian of the Keplerian elements with respect to $\boldsymbol{s}$, evaluated at $\boldsymbol{s}_{\mathcal{T}} = (r_t,\ r_n,\ 0,\ v,\ 0,\ 0)^T$, is (Gonzalo et al., 2020)

$$\boldsymbol{J}_{\boldsymbol{\alpha s}_{\mathcal{T}}} = \begin{pmatrix} \frac{2a^2}{r^3} r_t & \frac{2a^2}{r^3} r_n & 0 & \frac{2a^2 v}{\mu} & 0 & 0 \\ G_{21} r_t + \frac{r s_f}{a\hbar} v & G_{21} r_n & 0 & \frac{2(e+c_f)}{v} & -\frac{r s_f}{a v} & 0 \\ 0 & 0 & \frac{s_u + e s_\omega}{p} & 0 & 0 & \frac{r c_u}{\hbar} \\ 0 & 0 & -\frac{c_u + e c_\omega}{p s_i} & 0 & 0 & \frac{r s_u}{\hbar s_i} \\ G_{51} r_t + G_{52} v & G_{51} r_n & \frac{(c_u + e c_\omega) c_i}{p s_i} & \frac{2 s_f}{e v} & \frac{2}{v} + \frac{r c_f}{e a v} & -\frac{r s_u}{\hbar \tan i} \\ G_{61} r_t + G_{62} v & G_{61} r_n & 0 & -\frac{2 s_f}{e v} & -\frac{2}{v} - \frac{r c_f}{e a v} & 0 \end{pmatrix} \tag{2.18}$$

with the argument of latitude, $u = \omega + f$, the shortened notation, $s_\gamma = \sin\gamma$ and $c_\gamma = \cos\gamma$, for $\gamma \in [\omega, f, u]$, the gravitational parameter, μ, the specific angular momentum, $\hbar$, that can be calculated via the semi-parameter, p, as

$$p = a(1 - e^2) \tag{2.19a}$$

$$\hbar = \sqrt{\mu p} \tag{2.19b}$$

and with the parameters

$$G_{21} = \frac{1}{\mu a e} \left(\frac{\hbar^2 a}{r^3} - v^2 \right) \tag{2.20a}$$

$$G_{51} = -\frac{r \sin f}{\hbar^2 e} \left(\frac{\hbar^2 (p + e^2 r)}{p r^3} - \frac{(p + r) v^2}{r^2} \right) \tag{2.20b}$$

$$G_{52} = -\frac{r(c_f + e)}{\hbar e p} \tag{2.20c}$$

$$G_{61} = \frac{r \sin f}{\hbar^2 e} \frac{\mu}{a r^3} \left((r - a)(p + r) - r a \right) \tag{2.20d}$$

$$G_{62} = \frac{r}{\hbar^2 e} \left(\frac{\hbar (c_f + e)}{p} + \frac{\hbar e}{r} \right) \tag{2.20e}$$

Note that $\boldsymbol{J}_{\boldsymbol{\alpha s}_{\mathcal{T}}}$ is singular for $e = 0$ and/or $i = 0/180$ deg, for which some of the Keplerian elements are ill-defined. Some submatrices, e.g. for the transformation of only the velocity components into (a, e, i), instead are

singular at the perigee and apogee, i.e. at $f = 0/180$ deg, and 90 deg away from the crossing of the equatorial plane, i.e. at $u = \pm 90$ deg. Even for non-singular elements, such as equinoctial elements (Danielson et al., 1995), the singularities in the submatrices remain. Hence, non-singular elements do not solve the problem of singularity in the transformation.

Practically, the transformation of a distribution of fragments stemming from a parent object residing on or close to one of those singularities can be performed in Keplerian elements, as the fragments themselves are ejected out of the singularity. An example for a fragmentation close to the perigee is given in Section 6.1.2. However, a more rigorous treatment of the transformation would be necessary if it used for applications such as stability estimation, where some of the required assumptions on the distribution might not apply given the singularities. For some of these applications, it might be sufficient to work in non-singular elements. A compilation of partial derivatives of transformations between Cartesian coordinates, spherical coordinates, Keplerian elements and equinoctial elements is given in Vallado and Alfano (2015) for the spaces defined relative to the $\mathcal{I}$-frame.

The density in Keplerian elements, $n_{\boldsymbol{\alpha}}$, can now be found from the density given in Cartesian coordinates, $n_{\boldsymbol{s}}$, by plugging $\boldsymbol{J}_{\boldsymbol{\alpha s}_{\mathcal{T}}}$ into Equation (2.10)

$$n_{\boldsymbol{\alpha}} = \frac{n_{\boldsymbol{s}}(\boldsymbol{\varphi}^{-1}_{\boldsymbol{\alpha s}_{\mathcal{T}}}(\boldsymbol{\alpha}))}{|\det \boldsymbol{J}_{\boldsymbol{\alpha s}_{\mathcal{T}}}|} \tag{2.21}$$

Equation (2.21) is valid for any Cartesian frame that is obtained by rotation from $\mathcal{T}$, as the volume is invariant under rotation. Thus, no specific frame is mandated for the transformation if it matches the frame in which $n_{\boldsymbol{s}}$ is defined. To increase the readability, the subscript $\mathcal{T}$ is dropped from the equations for the remainder of this work. The inverse density transformation is derived by inverting Equation (2.21) as

$$n_{\boldsymbol{s}} = n_{\boldsymbol{\alpha}} \left(\boldsymbol{\varphi}^{-1}_{\boldsymbol{s\alpha}}(\boldsymbol{s})\right) |\det \boldsymbol{J}_{\boldsymbol{\alpha s}}| \tag{2.22}$$

where

$$\boldsymbol{\varphi}^{-1}_{\boldsymbol{s\alpha}} = \boldsymbol{\varphi}_{\boldsymbol{\alpha s}} \tag{2.23}$$

relates a Cartesian state to its Keplerian elements (e.g. Vallado, 2013, Chapter 2.6).

It is important to mention that evaluating $\boldsymbol{J}_{\boldsymbol{\alpha s}}$ in units compliant with the International System of Units (SI) leads to an ill-conditioned matrix. Calculating the determinant of such a matrix might induce numerical errors. Therefore, it is advised to evaluate a normalised $\boldsymbol{J}_{\boldsymbol{\alpha s}}$ using adimensional variables (depicted with a hat), e.g.

$$\hat{a} = \hat{\mu} = 1 \quad \rightarrow \quad \hat{\boldsymbol{r}} = \frac{\boldsymbol{r}}{a}, \quad \hat{\boldsymbol{v}} = \frac{\boldsymbol{v}}{\sqrt{\mu/a}} \quad \text{etc.} \tag{2.24}$$

The resulting determinant then needs to be renormalised again as

$$\det \boldsymbol{J}_{\boldsymbol{\alpha s}}(a, \mu, \boldsymbol{r}, \boldsymbol{v}) = \frac{1}{a^{1/2}\mu^{3/2}} \det \boldsymbol{J}_{\boldsymbol{\alpha s}}(\hat{a}, \hat{\mu}, \hat{\boldsymbol{r}}, \hat{\boldsymbol{v}}) \tag{2.25}$$

to be used in Equation (2.22).

Equations (2.21) and (2.22) transform a distribution from Cartesian coordinates into oscillating Keplerian elements, $\boldsymbol{\alpha}$, and vice-versa. The propagation, discussed in Chapter 3, is performed in mean Keplerian elements, $\overline{\boldsymbol{\alpha}}$. Oscillating elements can be obtained from mean orbital elements through

$$\boldsymbol{\alpha} = \overline{\boldsymbol{\alpha}} + \Delta\boldsymbol{\alpha}(\overline{\boldsymbol{\alpha}}) \tag{2.26}$$

where $\Delta\boldsymbol{\alpha}$ contains the short-period variations, which depend on the considered force model. E.g. in his artificial satellite theory, Brouwer (1959) finds $\Delta\boldsymbol{\alpha}$ analytically considering the perturbations from the second-order zonal harmonic, J_2. Usually no explicit transformations from oscillating to mean elements are available making iterations necessary. Given Equation (2.26), a distribution given in $\boldsymbol{\alpha}$, $n_{\boldsymbol{\alpha}}$, can be transformed into a distribution given in $\overline{\boldsymbol{\alpha}}$, $n_{\overline{\boldsymbol{\alpha}}}$, using Equation (2.10) as

$$n_{\overline{\boldsymbol{\alpha}}} = n_{\boldsymbol{\alpha}}(\overline{\boldsymbol{\alpha}} + \Delta\boldsymbol{\alpha}) \left|\det \boldsymbol{J}_{\boldsymbol{\alpha}\overline{\boldsymbol{\alpha}}}\right| \tag{2.27}$$

with

$$\boldsymbol{J}_{\boldsymbol{\alpha}\overline{\boldsymbol{\alpha}}} = \left(\frac{\partial \alpha_i}{\partial \overline{\alpha}_j}\right)_{i,j=1,\ldots,6} \tag{2.28}$$

Plugging in Equation (2.26), shows that

$$\frac{\partial \alpha_i}{\partial \overline{\alpha}_j} = \delta_{ij} + \frac{\partial \Delta\alpha_i}{\partial \overline{\alpha}_j} \qquad \forall i, j = 1, \ldots, 6 \tag{2.29}$$

where δ_{ij} is the Kronecker delta. Noticing that $\frac{\partial \Delta\alpha_i}{\partial \overline{\alpha}_j} << 1$, e.g. of the order of J_2 in Brouwer's theory, the Jacobian relating the mean and oscillating state approximates the 6×6 identity matrix, $\boldsymbol{I}$ (Schaub et al., 2000). Hence, Equation (2.27) can be approximated as

$$n_{\overline{\boldsymbol{\alpha}}} \approx n_{\boldsymbol{\alpha}}(\overline{\boldsymbol{\alpha}} + \Delta\boldsymbol{\alpha}) \tag{2.30}$$

The fragments obtained from the breakup model discussed in Section 2.5 span large domains in $\boldsymbol{\alpha}$, resulting in radial distributions covering 10^3 km (see Chapter 6). Herein, the relatively small displacements of $\Delta\boldsymbol{\alpha}$ between oscillating and mean elements, e.g. in the order of 10 km in radial direction, are thus ignored. However, to study the collision probability between two known objects with comparably small uncertainties, this simplification does not hold.

2.3 Randomisation

As observed by McKnight and Lorenzen (1989), the evolution of a cloud of fragments can be considered in phases. In the first phase, shortly after a breakup, the fragments remain close to each other in an elliptical cloud. Differences in a, induced by the disintegration itself, means the time for each fragment to complete one revolution varies with respect to the others. The second phase is reached once the fragments are randomly distributed along the orbit, e.g. the same density of fragments is found for any mean anomaly, M. In this phase, the fragments form a torus around Earth.

Simultaneously the fragments start dispersing in Ω and ω due to an oblate gravitational field and differences in the orbital elements. As this occurs over a longer time span, the separation of the phases is justified. Over the course of a few years, fragments in a LEO cloud are found in a band around Earth, loosely limited in latitude by the parent object inclination.

Similarly, Sykes (1990) assumes randomisation to model the formation of dust bands associated to asteroid families. He derives the volume density, as a function of radius and latitude, of dust tori and bands through geometrical considerations, analogous to Kessler (1981), who finds the volume density of a single object. These derivations are tedious and not flexible to incorporate less strict assumptions, e.g. they cannot model distributions that are not yet fully randomised in Ω, limiting the derivations of the spatial density to long-term scenarios only.

In this work, a less strict variant is presented that allows to model any distribution in $\boldsymbol{\alpha}$ and derive its phase space density describing the position and velocity distribution or the spatial density through transformation of densities. The application of the assumptions directly in Keplerian elements is straight-forward. Each of the randomisations can be applied independently of the other, thus the extensions are valid for distributions given in any combination of $\boldsymbol{\alpha}$, making derivations of the spatial density possible for scenarios considering any time frame.

Start with a distribution, $n_{a,e}$, given as a function of a and e. Often, the inclination, i, of the fragments can be approximated as the parent inclination, i_0. This assumption is reasonable because changes in i require large out-of-plane Δv components. Even over time, the effects of perturbations on the inclination of LEO objects is small (McKnight and Lorenzen, 1989). Thus, the distribution can be extended to

$$n_{a,e,i} = \delta(i - i_0) n_{a,e} \tag{2.31}$$

where δ is, again, the Dirac generalised function, defined in Equation (2.14). Randomisation of Ω due to nodal precession results in a uniform distribution over $[0, 2\pi)$

$$n_{a,e,i,\Omega} = \frac{1}{2\pi} n_{a,e,i} \qquad \forall \Omega \in [0, 2\pi) \tag{2.32}$$

Similarly, apsidal precession leads to the randomisation of ω

$$n_{a,e,i,\Omega,\omega} = \frac{1}{2\pi} n_{a,e,i,\Omega} \qquad \forall \omega \in [0, 2\pi) \tag{2.33}$$

Analogously, if the fragment is assumed equally likely to be found between $M \in [0, 2\pi)$

$$n_{a,e,i,\Omega,\omega,M} = \frac{1}{2\pi} n_{a,e,i,\Omega,\omega} \qquad \forall M \in [0, 2\pi) \tag{2.34}$$

Using the following transformations between the mean anomaly, M, eccentric anomaly, E, and true anomaly, f (Vallado, 2013, Chapter 2.2)

$$M = E - e \sin E \tag{2.35a}$$

$$\cos E = \frac{e + \cos f}{1 + e \cos f} \tag{2.35b}$$

$$\sin E = \frac{\sqrt{1 - e^2} \sin f}{1 + e \cos f} \tag{2.35c}$$

the derivative of f with respect to M is found as

$$\frac{\mathrm{d}f}{\mathrm{d}M} = \frac{(1 + e \cos f)^2}{(1 - e^2)^{3/2}} \tag{2.36}$$

Thus, the distribution in the full Keplerian element set $\boldsymbol{\alpha}$, using Equation (2.10), is

$$n_{\boldsymbol{\alpha}} = n_{a,e,i,\Omega,\omega,f} = \left| \frac{\mathrm{d}f}{\mathrm{d}M} \right|^{-1} n_{a,e,i,\Omega,\omega,M} \tag{2.37a}$$

$$= \frac{(1 - e^2)^{3/2}}{(1 + e \cos f)^2} n_{a,e,i,\Omega,\omega,M} \qquad \forall f \in [0, 2\pi) \tag{2.37b}$$

Equation (2.22) transforms a distribution in the full Keplerian element set into Cartesian coordinates. Hence, the phase space density in Cartesian coordinates, $n_{\boldsymbol{s}}$, can be found from any subset of $\boldsymbol{\alpha}$, given the dependence on a and e and assumptions on randomisation.

2.4 Products

Given the phase space density fully describing position and velocity in either Cartesian or Keplerian, or in fact any other phase space, enables the derivation of products. Here, it is shown how to derive the spatial density and the impact rate from a cloud of fragments.

2.4.1 Spatial density

Given the phase space density in Cartesian coordinates, $n_{\boldsymbol{s}} = n_{\boldsymbol{r},\boldsymbol{v}}$, the spatial density is obtained by integration over the velocities

$$n_{\boldsymbol{r}} = \int_{\mathbb{R}^3} n_{\boldsymbol{r},\boldsymbol{v}} \, \mathrm{d}^3\boldsymbol{v} \tag{2.38}$$

The spatial density describes the number of objects within an infinitesimal spatial volume. Instead, the phase space density describes the number of objects within an infinitesimal phase space volume. For a phase space describing a position in space, the two are equivalent if they are defined in the same physical space. It is worth noting that a fixed inclination constraint reduces the possible incident velocities into a plane. Hence, the volume integral in Equation (2.38) can be reduced to an area integral by rotating into the satellite frame, $\mathcal{T}$, where the probability for non-zero out-of-plane velocities, v_h, is zero. Thus,

$$n_{\boldsymbol{r}_\mathcal{T}} = \int_{\mathbb{R}^2} n_{\boldsymbol{r}_\mathcal{T},\boldsymbol{v}_\mathcal{T}} \, \mathrm{d}v_t \, \mathrm{d}v_n \tag{2.39}$$

Instead, the spatial density can be directly obtained from Keplerian elements using the Dirac generalised function (see Section 2.2.1)

$$n_{\boldsymbol{r}}(\boldsymbol{r}^*) = \int_{\mathbb{R}^6} n_{\boldsymbol{\alpha}} \, \delta\left(\boldsymbol{r}(\boldsymbol{\alpha}) - \boldsymbol{r}^*\right) \mathrm{d}^6\boldsymbol{\alpha} \tag{2.40}$$

where $\boldsymbol{r}^*$ is the position of interest. To find the roots of $\boldsymbol{r}(\boldsymbol{\alpha}) - \boldsymbol{r}^*$, it is important to find a three-dimensional subset of $\boldsymbol{\alpha}$ – the integrands – that together with $\boldsymbol{r}^*$ permit to find all elements of $\boldsymbol{\alpha}$. Not all combinations of the elements are possible. It needs to contain one of the elements defining the orientation of the orbital plane, i.e. i and Ω. And it requires any two of the other four elements, excluding the combination of ω and f, because this combination does not permit to calculate the angular momentum or mechanical energy of the orbit.

Here, the subset of a, e and i is chosen because the limits are defined with respect to the point of interest and the integral over i is trivial if assumed fixed. Given this subset, four possible element sets are found, corresponding to two orbital planes and two pairs of (ω, f) each arising from the symmetry

$$\boldsymbol{\alpha}_j^*(\boldsymbol{r}^*, a, e, i) = (a, e, i, \Omega_j^*, \omega_j^*, f_j^*) \qquad j = 1, \ldots, 4 \tag{2.41}$$

E.g. with a given a point, $\boldsymbol{r}^*$, and fixed (a, e, i)

$$\boldsymbol{r}^* = \begin{pmatrix} r^* \\ \lambda^* \\ \theta^* \end{pmatrix} = \begin{pmatrix} 8000 \text{ km} \\ 45 \text{ deg} \\ 45 \text{ deg} \end{pmatrix} \qquad \begin{pmatrix} a \\ e \\ i \end{pmatrix} = \begin{pmatrix} 10000 \text{ km} \\ 0.3 \\ 70 \text{ deg} \end{pmatrix} \tag{2.42}$$

Table 2.1: Required subset of elements to cross $\boldsymbol{r}^*$ given (a, e, i).

j	Ω_j [deg]	ω_j [deg]	f_j [deg]
1	23.7	346.1	62.7
2	246.3	68.5	62.7
3	23.7	111.5	297.3
4	246.3	193.9	297.3

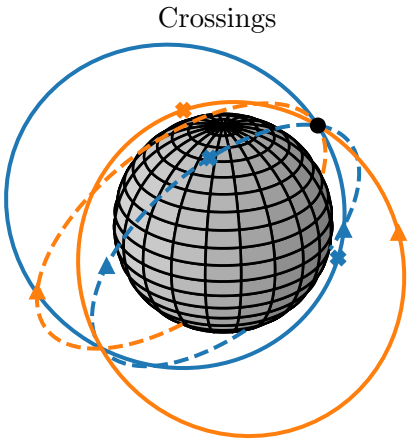


Figure 2.3: Orbital configurations in two orbital planes cross the point of interest (black dot) in ascending (solid) and descending (dashed) direction. The crosses and triangles show the perigee and ascending node, respectively.

with the radius, r, the longitude, λ, and the latitude, θ, defined relative to $\mathcal{I}$, the missing elements given in Table 2.1 can be found (see Appendix B.1 for more information). The corresponding orbits are depicted in Figure 2.3. Solutions are not always available, e.g. if the apogee radius is lower than r^*, the perigee radius is larger than r^*, or the inclination is too close to the equator, $\boldsymbol{r}^*$ cannot be reached.

Given the full element sets, Equation (2.40) simplifies to

$$n_{\boldsymbol{r}}(\boldsymbol{r}^*) = \int_{\frac{r^*+R_E}{2}}^{\infty} \int_{\left|1-\frac{r^*}{a}\right|}^{\left|1-\frac{R_E}{a}\right|} \int_{|\theta|}^{\pi-|\theta|} \sum_{j=1}^{4} \frac{n_{\boldsymbol{\alpha}}\left(\boldsymbol{\alpha}_j^*\right)}{\left|\det \boldsymbol{J}_j^*\right|} \,\mathrm{d}i \,\mathrm{d}e \,\mathrm{d}a \tag{2.43}$$

where the Jacobian, $\boldsymbol{J}^*$, relates $\boldsymbol{r}^*$ to $(\Omega^*, \omega^*, f^*)$ as

$$\boldsymbol{J}^* = \left(\begin{matrix} \frac{\partial r_x}{\partial \Omega} & \frac{\partial r_x}{\partial \omega} & \frac{\partial r_x}{\partial f} \\ \frac{\partial r_y}{\partial \Omega} & \frac{\partial r_y}{\partial \omega} & \frac{\partial r_y}{\partial f} \\ \frac{\partial r_z}{\partial \Omega} & \frac{\partial r_z}{\partial \omega} & \frac{\partial r_z}{\partial f} \end{matrix} \right) \Bigg|_{\boldsymbol{\alpha} = \boldsymbol{\alpha}^*} \tag{2.44}$$

After some derivations, the determinant is found as

$$\det \boldsymbol{J}^* = \frac{a^3 \left(1 - e^2\right)^3 \cos\left(\omega^* + f^*\right) \sin f^* \sin i}{\left(1 + e \cos f^*\right)^4} \tag{2.45}$$

Equation (2.43) converts a phase space density given in Keplerian elements directly into a spatial density. As before, the volume integral simplifies to an area integral if the inclination of the fragments in the cloud are fixed.

2.4.2 Impact rate

Given the phase space density in Cartesian coordinates, the impact rate, $\dot{\eta}$, i.e. the number of fragments impacting on a surface of a target located at $(\boldsymbol{r}^*, \boldsymbol{v}^*)$ at a given epoch, can accurately be estimated as

$$\dot{\eta}(\boldsymbol{r}^*, \boldsymbol{v}^*) = \int_{\mathbb{R}^3} A_c \left(\frac{\boldsymbol{v} - \boldsymbol{v}^*}{||\boldsymbol{v} - \boldsymbol{v}^*||} \right) n_{\boldsymbol{r},\boldsymbol{v}}(\boldsymbol{r}^*, \boldsymbol{v}) \; ||\boldsymbol{v} - \boldsymbol{v}^*|| \, \mathrm{d}^3\boldsymbol{v} \tag{2.46}$$

with the cross-sectional area, A_c, exposed to the incoming flux. As before, $\dot{\eta}$ can be estimated through integration directly in the Keplerian space. Equation (2.46) is generally valid, without any assumptions on the geometries of chaser or target orbit and includes the possibility to consider the spacecraft orientation. The only assumptions are that the area of the fragments is small compared to the area of the target and can thus be ignored, and that the flux is constant over A_c. The first assumption is valid, as $\geq 97.5\%$ of all fragments are smaller than $L = 1$ cm (see Section 2.5). Given the large variance of fragment velocities, the second assumption is not only reasonable, but constitutes a great improvement compared to a default cube size of 10 km $\times$ 10 km $\times$ 10 km used in the Cube algorithm (Liou, 2006).

The number of impacts, η, can be calculated by integrating Equation (2.46) over time, t

$$\eta(t) = \int_{t_0}^{t} \dot{\eta} \left(\boldsymbol{r}^*(t), \boldsymbol{v}^*(t) \right) \mathrm{d}t \tag{2.47}$$

Alternatively, the average number of impact rate, $\dot{\bar{\eta}}$, over the course of one target orbit might be of interest and can be found as

$$\dot{\bar{\eta}} = \frac{1}{P} \int_0^P \dot{\eta} \left(\boldsymbol{r}^*(t), \boldsymbol{v}^*(t) \right) \mathrm{d}t \tag{2.48a}$$

$$= \frac{1}{2\pi} \int_0^{2\pi} \dot{\eta} \left(\boldsymbol{r}^*(M), \boldsymbol{v}^*(M) \right) \mathrm{d}M \tag{2.48b}$$

with the orbital period, P.

Note that assuming A_c to be constant in any direction, i.e. having a spherical target, and given

$$n_{\boldsymbol{r},\boldsymbol{v}} = p_{\boldsymbol{v}|\boldsymbol{r}} n_{\boldsymbol{r}} \tag{2.49}$$

where $p_{\boldsymbol{v}|\boldsymbol{r}}$ is the conditional distribution of $\boldsymbol{v}$ given $\boldsymbol{r}$ and $n_{\boldsymbol{r}}$ is the spatial density that can be inferred from Equation (2.38), Equation (2.46) can be reduced to

$$\dot{\eta} = A_c n_{\boldsymbol{r}} \Delta\overline{v} \tag{2.50}$$

where

$$\Delta\overline{v} = \int_{\mathbb{R}^3} ||\boldsymbol{v} - \boldsymbol{v}^*|| p_{\boldsymbol{v}|\boldsymbol{r}} \, \mathrm{d}^3\boldsymbol{v} \tag{2.51}$$

is the average collision velocity. Equation (2.50) is equivalent to the time derivative of Equation (1.25). Still, Equation (2.46) is preferred to calculate $\dot{\eta}$ because it requires only one integration and A_c is not required to be constant in any direction.

2.5 Initial cloud continuum

The NASA Standard Breakup Model (SBM) (Johnson et al., 2001) can be used to sample deterministic fragments following an explosion or collision event in orbit. It is an empirical model based on evidence compiled from historical orbital data and ground-based impact tests. The samples are described in characteristic length, L, area-to-mass ratio, A/m, and the ejection velocity, Δv, imparted in random direction relative to the parent orbital velocity. Here, the NASA SBM is reformulated as a probability distribution function. As such, it allows not only to draw samples, but also to assign their probabilities, a requirement for the definition of the characteristics. Another advantage of a probabilistic description of the breakup model is that expectation values can be derived, which is done herein, and used to check the conservation of mass or energy.

Note that the techniques developed throughout this work are independent of the exact formulation of the initial distribution and are applicable using any other distribution. Semi-empirical breakup models, still fast enough for application over many collision scenarios, are the Fragmentation Algorithms for Strategic and Theater Targets (FASTT) model (McKnight et al., 1995), the IMPACT model (Sorge and Mains, 2016), and the more recent and higher fidelity Collision Simulation Tool (CST) (Francesconi et al., 2019). From the CST, however, it is not possible to directly obtain an analytical description of the fragment distribution, as it is based on discrete macroscopic elements.

2.5.1 Reformulation of the NASA SBM

The number of objects with L larger than a lower bound, L_0, produced in the fragmentation event is

$$N_L\left(L_0\right) = k \left(\frac{L_0 \ [\mathrm{m}]}{1 \ [\mathrm{m}]} \right)^{-\beta} \qquad k, \beta > 0 \tag{2.52}$$

where k and β are unitless parameters dependent on the type of fragmentation and the physical characteristics of the involved objects. All the parameters introduced in this Section are given in Appendix C.1. In the following, the normalisation in the units, e.g.

$$\frac{L_0\ [\mathrm{m}]}{1\ [\mathrm{m}]} \tag{2.53}$$

is omitted. Each variable is, if not stated otherwise, defined in SI units. Equation (2.52) can be converted into a probability density function as

$$p_L = \frac{k\beta}{N} L^{-(\beta+1)} \qquad \forall L_0 \leq L \leq L_1 \tag{2.54}$$

where L_1 is the upper boundary on the characteristic length, and

$$N = N_L(L_0) - N_L(L_1) \tag{2.55}$$

is the total number of fragments within this range. Combining Equation (2.54) and Equation (2.55) as

$$p_L = \beta \frac{L^{-(\beta+1)}}{L_0^{-\beta} - L_1^{-\beta}} \qquad \forall L_0 \leq L \leq L_1 \tag{2.56}$$

reveals that p_L is independent of the number of fragments involved and thus solely dependent on the power factor, β. Still, all the probability distribution functions derived herein can be multiplied by N to obtain the fragment phase space densities.

The conditional probabilities introducing the dependence on L, A/m, and Δv, are given as a function of the logarithm to base 10 of the characteristic length, $\lambda = \log_{10}(L)$, the logarithm to base 10 of the area-to-mass ratio, $\chi = \log_{10}(A/m)$, and the logarithm to base 10 of the absolute impulse, $\upsilon = \log_{10}(\Delta v)$. Transforming Equation (2.54) into a function dependent on λ using Equation (2.10) yields

$$p_\lambda = \log(10) 10^\lambda p_L(10^\lambda) \qquad \forall \lambda_0 \leq \lambda \leq \lambda_1 \tag{2.57a}$$

$$= \log(10)\beta \frac{10^{-\beta\lambda}}{10^{-\beta\lambda_0} - 10^{-\beta\lambda_1}} \tag{2.57b}$$

where $\lambda_0 = \log_{10}(L_0)$ and $\lambda_1 = \log_{10}(L_1)$. To sample λ from Equation (2.57b), the cumulative distribution function

$$P_\lambda = \int_{\lambda_0}^{\lambda} p_\lambda(\lambda')\,\mathrm{d}\lambda' = \frac{10^{-\beta\lambda_0} - 10^{-\beta\lambda}}{10^{-\beta\lambda_0} - 10^{-\beta\lambda_1}} \tag{2.58}$$

is solved for λ

$$\lambda = -\frac{1}{\beta} \log_{10}\left[10^{-\beta\lambda_0} - P_\lambda\left(10^{-\beta\lambda_0} - 10^{-\beta\lambda_1}\right)\right] \tag{2.59}$$

Equation (2.59) enables the sampling of λ by sampling P_λ uniformly $\in [0, 1]$. The conditional probabilities of χ dependent on λ and v dependent on χ are as sum of normal distributions, $\mathcal{N}$

$$p_{\chi|\lambda} = \sum_i \alpha_i(\lambda)\mathcal{N}\left(\mu_\chi^{(i)}(\lambda), \sigma_\chi^{(i)}(\lambda)\right) \qquad \sum_i \alpha_i(\lambda) = 1 \tag{2.60a}$$

$$p_{v|\chi} = \mathcal{N}\left(\mu_v(\chi), \sigma_v(\chi)\right) \tag{2.60b}$$

where $p_{y|x}$ denotes a conditional probability in y given x, and $\alpha_i, \mu_\chi^{(i)}, \sigma_\chi^{(i)}, \mu_v$ and σ_v are dependent on the type of fragmentation, i.e. explosion or fragmentation, and the type of the involved objects, i.e. payload or rocket body. Most relevant mathematical libraries provide functions to sample normal distributions. The joint probability in λ, χ and v is

$$p_{\lambda,\chi,v} = p_{v|\chi}p_{\chi|\lambda}p_\lambda \tag{2.61}$$

Marginal distributions can be obtained by numerical integration, e.g.

$$p_{\chi,v} = p_{v|\chi}\int_{\lambda_0}^{\lambda_1} p_{\chi|\lambda}p_\lambda \,\mathrm{d}\lambda \tag{2.62a}$$

$$p_v = \int_{\mathbb{R}} p_{\chi,v} \,\mathrm{d}\chi \tag{2.62b}$$

The distribution in L, A/m and Δv can be recovered through application of Equation (2.10) as

$$p_{L,\frac{A}{m},\Delta v} = \frac{p_{\lambda,\chi,v}\left(\log_{10}(L), \log_{10}\left(\frac{A}{m}\right), \log_{10}(\Delta v)\right)}{\log^3(10)L\frac{A}{m}\Delta v} \tag{2.63}$$

Assuming an isotropic impulse direction, the distribution is extended to model three-dimensional velocity changes (see Appendix B.2)

$$p_{L,\frac{A}{m},\Delta \boldsymbol{v}} = \frac{p_{L,\frac{A}{m},\Delta v}(L, \frac{A}{m}, ||\Delta \boldsymbol{v}||)}{4\pi||\Delta \boldsymbol{v}||^2} \tag{2.64}$$

Finally, a simple translation results in the desired density function

$$p_{L,\frac{A}{m},\boldsymbol{v}}(L, \frac{A}{m}, \boldsymbol{v}^+) = p_{L,\frac{A}{m},\Delta\boldsymbol{v}}(L, \frac{A}{m}, \boldsymbol{v}^+ - \boldsymbol{v}^-) \tag{2.65}$$

where $\boldsymbol{v}^-$ and $\boldsymbol{v}^+$ are the orbital velocities immediately before and after the impulse. As the fragments are ejected equally likely in any direction, the frame can be freely chosen.

The fragment cross-sectional area, A, and the fragment mass, m, follow directly from the previously introduced random variables as

$$A = bL^\gamma \tag{2.66a}$$

$$m = \frac{A}{A/m} \tag{2.66b}$$

where b and γ are parameters dependent on L, given in Appendix C.1.

The distribution $n_{\Delta \boldsymbol{v}}$ describes a three-dimensional distribution. It can be converted into a three-dimensional distribution given in a subset of $\boldsymbol{\alpha}$ using Equation (2.21). However, $\boldsymbol{J}_{\boldsymbol{\alpha s}}$ needs to be chosen accordingly. If, for example, a distribution in $\boldsymbol{v}$ is to be transformed into a distribution in (a, e, i), only the upper right 3×3 sub matrix of the full Jacobian, relating velocities to the chosen Keplerian elements, is required. Not all permutations are feasible, as some of them lead to a decreased rank of the sub matrix.

Marginalisation, i.e. obtaining lower dimensional distributions in the target space, can also be achieved through integration or direct application of the Dirac generalised function. However, it is not feasible to go from the three-dimensional distribution to a higher dimensional one because it would lead to a distribution that has lower rank than its phase space.

To tackle this problem, consider that also the position of the fragmentation, $\boldsymbol{r}$, is afflicted with an uncertainty, $p_{\Delta \boldsymbol{r}|\Delta \boldsymbol{v}}$. The additional uncertainty can be modelled to be dependent on $\Delta \boldsymbol{v}$. To obtain the uncertainty in the full Cartesian phase space, $\boldsymbol{s}$, multiplication is required as

$$p_{\Delta \boldsymbol{s}} = p_{\Delta \boldsymbol{r}|\Delta \boldsymbol{v}} p_{\Delta \boldsymbol{v}} \tag{2.67}$$

Note that such an extension is only required if Keplerian phase spaces with $D > 3$ are considered. Herein, the uncertainty in the position is assumed to be independent of the velocity, i.e. $p_{\Delta \boldsymbol{r}|\Delta \boldsymbol{v}} = p_{\Delta \boldsymbol{r}}$.

2.5.2 Expectation values

The NASA SBM does not inherently conserve physical quantities such as mass and kinetic energy (Finkleman et al., 2008). Given a fully probabilistic description of the breakup model permits to statistically check the conservation, or enforce it, via the calculation of expectation values. In general, the expectation value, $\mathbb{E}$, of a continuous random variable, $x \in \mathbb{R}$ with the probability density function, p_x, is

$$\mathbb{E}(x) = \int_{\mathbb{R}} x p_x \, \mathrm{d}x \tag{2.68}$$

For the D-dimensional case for a set of continuous random variables, $\boldsymbol{x} \in \mathbb{R}^D$ with a joint probability density function, $p_{\boldsymbol{x}}$, the expectation value of a function, $g(\boldsymbol{x})$ is

$$\mathbb{E}\left(g(\boldsymbol{x})\right) = \int_{\mathbb{R}^D} g(\boldsymbol{x}) p_{\boldsymbol{x}} \, \mathrm{d}^D \boldsymbol{x} \tag{2.69}$$

The integration can be performed in any frame. As the distributions are mostly defined with regards to $(\lambda, \chi, \upsilon)$, the expectation values are most

easily integrated in this frame too. The mass, m, and kinetic energy of the impulse, $\Delta\epsilon$, are calculated as

$$m = \frac{A}{A/m} = b10^{\lambda\gamma-\chi} \tag{2.70a}$$

$$\Delta\epsilon = \frac{1}{2}m\Delta v^2 = \frac{b}{2}10^{\lambda\gamma-\chi+2v} \tag{2.70b}$$

Hence, their expectation values are

$$\mathbb{E}(m) = \int_{\mathbb{R}}\int_{\lambda_0}^{\lambda_1} b10^{\lambda\gamma-\chi}p_{\chi|\lambda}p_\lambda \,\mathrm{d}\lambda\,\mathrm{d}\chi \tag{2.71a}$$

$$\mathbb{E}(\Delta\epsilon) = \frac{1}{2}\int_{\mathbb{R}}\int_{\mathbb{R}}\int_{\lambda_0}^{\lambda_1} b10^{\lambda\gamma-\chi+2v}p_{v|\chi}p_{\chi|\lambda}p_\lambda \,\mathrm{d}\lambda\,\mathrm{d}\chi\,\mathrm{d}v \tag{2.71b}$$

The expectation values for the different fragmentation scenarios are given in Table 2.2 for different bins in L, alongside the respective shares, P_L, which is the integral over the given bins assuming $L_0 = 1$ mm and $L_1 = 1$ m. Note that these values are independent of the total number of fragments, N, given from Equation (2.55). Instead, they can be used to set N to enforce conservation in either mass or kinetic energy. Conservation of mass means the sum of the mass of all the fragments equals, on average, the fragmenting mass, M_i, i.e.

$$M_i = N_m\,\mathbb{E}(m) \tag{2.72}$$

where N_m is the total number of fragments respecting mass conservation. The fragmenting mass is not simply the mass of the parent object, as $2-8$ fragments with $L > 1$ m, not following the power law in Equation (2.52), comprise the bulk of the mass (NASA, 2011). E.g. for a payload collision, the mean mass for 1 mm $< L <$ 1 m is $\mathbb{E}(m) = 4.13\times10^{-4}$ kg. If the mass is set at $M_i = 100$ kg it follows from Equation (2.72) that $N_m = 2.4\times10^5$, given that the combined mass in fragments smaller than $L < 1$ mm is negligible. The number of fragments required to assure conservation of kinetic energy is harder to estimate as losses due to heat and rotational velocities are not known in this simple model (Finkleman et al., 2008). Luckily, the cloud evolution, from the initial distribution, through propagation to estimation of the number of impacts can be performed based on the distribution of a single object only. Once an accurate estimate of N is available, the results can be updated through multiplication with N.

2.5.3 Covariance in velocity

For uncertainty propagation methods based on covariance propagation (e.g. see Luo and Yang, 2017), knowledge about the mean and covariance of the distribution is required. Here, the mean and variance in Δv_h are derived

Table 2.2: Expectation values for the mass and kinetic energy, and variance for the velocity impulse of the fragments according to the NASA SBM.

Case		L [m]	P_L [−]	$\mathbb{E}(m)$ [kg]	$\mathbb{E}(\Delta\epsilon)$ [J]	$\sigma^2_{\Delta v_h}$ $\left[\frac{\mathrm{m}^2}{\mathrm{s}^2}\right]$
Collision	Payload	$10^{-3} - 10^{-2}$	9.81×10^{-1}	8.46×10^{-6}	3.75×10^{0}	7.23×10^{5}
		$10^{-2} - 10^{-1}$	1.91×10^{-2}	3.67×10^{-3}	2.40×10^{2}	1.21×10^{6}
		$10^{-1} - 10^{-0}$	3.73×10^{-4}	8.97×10^{-1}	1.05×10^{4}	7.62×10^{4}
		$10^{-3} - 10^{-0}$	1	4.13×10^{-4}	1.22×10^{1}	7.31×10^{5}
	Rocket body	$10^{-3} - 10^{-2}$	9.81×10^{-1}	8.46×10^{-6}	3.75×10^{0}	7.23×10^{5}
		$10^{-2} - 10^{-1}$	1.91×10^{-2}	3.47×10^{-3}	2.51×10^{2}	1.22×10^{6}
		$10^{-1} - 10^{-0}$	3.73×10^{-4}	2.78×10^{-1}	2.55×10^{4}	1.72×10^{6}
		$10^{-3} - 10^{-0}$	1	1.78×10^{-4}	1.80×10^{1}	7.32×10^{5}
Explosion	Payload	$10^{-3} - 10^{-2}$	9.75×10^{-1}	9.24×10^{-6}	7.81×10^{-2}	7.18×10^{3}
		$10^{-2} - 10^{-1}$	2.45×10^{-2}	4.16×10^{-3}	1.70×10^{1}	6.83×10^{3}
		$10^{-1} - 10^{-0}$	6.15×10^{-4}	9.92×10^{-1}	2.86×10^{3}	3.95×10^{3}
		$10^{-3} - 10^{-0}$	1	7.21×10^{-4}	2.25×10^{0}	7.15×10^{3}
	Rocket body	$10^{-3} - 10^{-2}$	9.75×10^{-1}	9.24×10^{-6}	7.81×10^{-2}	7.18×10^{3}
		$10^{-2} - 10^{-1}$	2.45×10^{-2}	3.91×10^{-3}	1.64×10^{1}	6.84×10^{3}
		$10^{-1} - 10^{-0}$	6.15×10^{-4}	3.07×10^{-1}	1.52×10^{3}	6.07×10^{3}
		$10^{-3} - 10^{-0}$	1	2.93×10^{-4}	1.41×10^{0}	7.15×10^{3}

from the NASA SBM. As isotropic directionality is assumed, the presented values are also applicable in any other direction. To derive any moment in Δv_h, its probability distribution function is required. As becomes apparent later, it is beneficial to transform the distribution in Equation (2.64) into cylindrical coordinates

$$\Delta v_\perp = \sqrt{(\Delta v)^2 - (\Delta v_h)^2} \tag{2.73a}$$

$$\phi = \sin^{-1}\left(\frac{\Delta v_h}{\Delta v}\right) \tag{2.73b}$$

$$\Delta v_h = \Delta v_h \tag{2.73c}$$

yielding, under application of Equation (2.10), the distribution

$$p_{\Delta v_\perp,\phi,\Delta v_h} = \Delta v_\perp p_{\Delta \boldsymbol{v}} = \frac{\Delta v_\perp}{4\pi(\Delta v)^2} p_{\Delta v} \tag{2.74}$$

Noticing that $p_{\Delta v_\perp,\phi,\Delta v_h}$ is not dependent on ϕ, it can be readily reduced as

$$p_{\Delta v_\perp,\Delta v_h} = \int_0^{2\pi} p_{\Delta v_\perp,\phi,\Delta v_h}\,\mathrm{d}\phi = \frac{\Delta v_\perp}{2(\Delta v)^2} p_{\Delta v} \tag{2.75}$$

Thanks to the reduction, the distribution in Δv_h requires integration in only one, rather than two, variables

$$p_{\Delta v_h} = \int_0^\infty p_{\Delta v_\perp,\Delta v_h}\,\mathrm{d}\Delta v_\perp \tag{2.76}$$

From Equation (2.73a) it follows that $\Delta v_\perp\,\mathrm{d}\Delta v_\perp = \Delta v\,\mathrm{d}\Delta v$. Putting together the pieces allows to find

$$p_{\Delta v_h} = \int_{|\Delta v_h|}^\infty \frac{p_{\Delta v}}{2\Delta v}\,\mathrm{d}\Delta v \tag{2.77}$$

The expectation value of Δv_h is always zero as $p_{\Delta_h}(\Delta_h) = p_{\Delta_h}(-\Delta_h)$ is symmetrical, i.e.

$$\mathbb{E}(\Delta v_h) = \int_{-\infty}^\infty \Delta v_h p_{\Delta v_h}\,\mathrm{d}\Delta v_h = 0 \tag{2.78a}$$

The variance, $\sigma^2_{\Delta v_h}$, can be found as

$$\sigma^2_{\Delta v_h} = \mathrm{Var}(\Delta v_h) = \int_{-\infty}^\infty \left(\Delta v_h - \mathbb{E}(\Delta v_h)\right)^2 p_{\Delta v_h}\,\mathrm{d}\Delta v_h \tag{2.79a}$$

$$= 2\int_0^\infty (\Delta v_h)^2 p_{\Delta v_h}\,\mathrm{d}\Delta v_h \tag{2.79b}$$

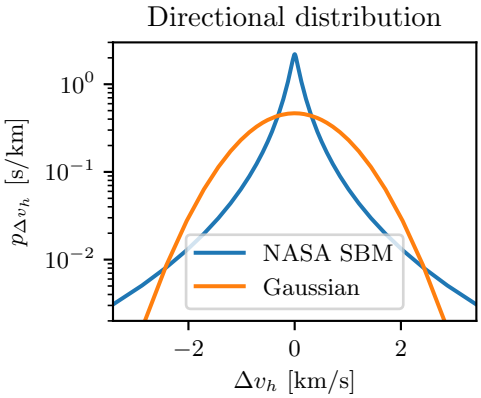


Figure 2.4: Transformed and sampled distribution in Δv_h according to the NASA SBM for a payload collision and fragments sized 1 mm $< L <$ 1 m, and its comparison with the Gaussian approximation with zero mean and variance, $\sigma^2_{\Delta v_h}$.

where the symmetry is exploited again. Finally, the mean, $\boldsymbol{\mu}$, and covariance matrix, $\boldsymbol{\Sigma}$, of the NASA SBM velocity distribution are given as

$$\boldsymbol{\mu}_{\Delta \boldsymbol{v}} = \mathbf{0} \qquad \boldsymbol{\Sigma}_{\Delta \boldsymbol{v}} = \operatorname{diag}\left(\sigma^2_{\Delta v_t}\ \sigma^2_{\Delta v_n}\ \sigma^2_{\Delta v_h}\right) \tag{2.80}$$

where $\sigma^2_{\Delta v_t} = \sigma^2_{\Delta v_n} = \sigma^2_{\Delta v_h}$. The corresponding values for the different cases and various characteristic length regimes – again valid for any N – are listed in Table 2.2. Note that a normal distribution with the given covariance matrix does not accurately describe $p_{\Delta \boldsymbol{v}}$, as can be seen in Figure 2.4 for the case of a payload collision. It underestimates both the peak and the tail end of the distribution, while overestimating the domain in between. Hence, application of the covariance needs to be treated with caution.

2.6 Examples

Two examples are discussed here to illustrate the derived transformations. The first transforms an initial distribution given in Cartesian coordinates into a subset of Keplerian elements. It illustrates the difficulties of using a phase space that is not suited for discretisation and introduces another phase space that is. The second example shows how to apply randomisation to obtain the spatial density, and how the analytical transformation is superior to inferring a multi-dimensional distribution function through discretisation and samples.

For comparison in terms of sensitivity and scalability, the analytical transformations are compared to densities estimated from sampling. To calculate the phase space density from samples, the individual instances, not

the density, are transformed into the target space, where they are binned according to Equation (2.5). The sampling method stands for two seemingly very different methods, Monte Carlo sampling (e.g. Liou et al., 2004) and methods based on finite differences (e.g. Letizia, 2018). They have both in common that discretisation of the phase space is required to estimate densities, such as the phase space or spatial density. As such, they both suffer from poor sensitivity. First, if the number of samples is too small compared to the number of bins, N_b, the bin might end up with a small number of samples not well representing the actual distribution. Second, binning spreads out highly concentrated distributions over the full bin volume, leading to inaccurate and averaged results, as will become evident herein. Thus, the ideal bin size is a trade-off between the sensitivity (many small bins) and the accuracy (enough samples in each bin), while keeping in mind that finite differencing methods suffer from poor scalability in dimensions (Sun and Kumar, 2012). Hence, binning requires a careful selection of the bin limits, which is dependent on the underlying distribution. In the simplest case, the bins are equally spaced in each separate dimension. Here, the outer bin limits, the range limits, are chosen such as to contain a considerable fraction, $\epsilon > 90\%$, of the overall distribution. Mathematically speaking, ϵ is defined as

$$\epsilon = \frac{\int_B p_{\boldsymbol{x}} \, \mathrm{d}^D \boldsymbol{x}}{\int_{\mathbb{R}^D} p_{\boldsymbol{x}} \, \mathrm{d}^D \boldsymbol{x}} = \int_B p_{\boldsymbol{x}} \, \mathrm{d}^D \boldsymbol{x} \tag{2.81}$$

where $B \in \mathbb{R}^D$ is the domain defined by the limits. The domain is regularly discretised with equal number of bins in each dimension.

2.6.1 LEO initial continuum

In February 2009, the two satellites Cosmos-2251 and Iridium 33 collided, resulting in two fragment clouds with hundreds to thousands of fragments large enough to be detected (NASA, 2009). Here, the initial Cosmos-2251 fragment cloud is modelled and discussed. The orbit for the Cosmos-2251 parent object at the epoch of collision, is (Space-Track, 2020)

$$\begin{aligned} \boldsymbol{\alpha} &= (a,\, e,\, i,\, \Omega,\, \omega,\, f) \\ &= (7166 \text{ km},\, 0.0016,\, 74 \text{ deg},\, 19.5 \text{ deg},\, 98.7 \text{ deg},\, 358.6 \text{ deg}) \end{aligned}$$

The initial distribution is obtained using the NASA SBM distribution, which is given as a function of L, A/m and Δv. Here, only the distribution in $\Delta v = 10^v$ is utilised, as the phase space transformation in A/m is a one-to-one relationship and thus trivial. To facilitate reproduction of the results, the initial distribution in v is assumed to be normally distributed as

$$v \sim \mathcal{N}(\mu_v, \sigma_v^2) \tag{2.83}$$

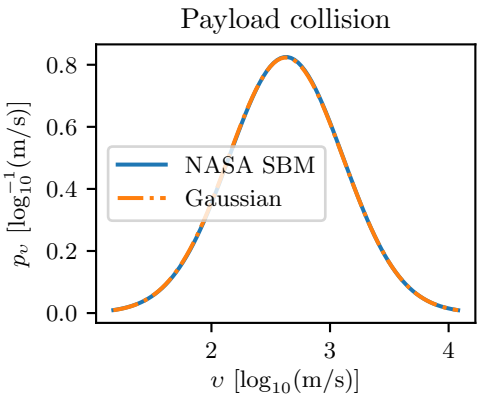


Figure 2.5: Integration of the NASA SBM over $L \in [1\text{ mm}, 1\text{ m}]$ and A/m, and its Gaussian approximation, given as a function of the logarithm to base 10 of Δv.

with the mean, $\mu_v = 2.63\ \log_{10}$ (m/s), and variance, $\sigma_v^2 = 0.48^2\ (\log_{10}\text{ m/s})^2$. The normal distribution is almost identical to the actual NASA SBM distribution for a payload collision, considering fragments with a characteristic length between 1 mm and 1 m (see Figure 2.5). The reason is that small fragments, at around 1 mm, dominate the overall distribution. Note that the resulting distribution gives the likelihood of a fragment to be ejected with a certain impulse, not the fragment density. The latter can be calculated by multiplying the likelihood with the number of fragments, N. Given the distribution in Equation (2.83), 50% of the fragments are modelled to be ejected with an impulse larger than 423 m/s, and 5% with an impulse larger than 2652 m/s. No cut-off velocity is required.

Marginalised target space

First, the transformed distribution is studied in a and e, useful when only the long-term evolution of the cloud is to be studied. As it requires a marginalisation from $\mathbb{R}^3 \to \mathbb{R}^2$ it is found through integration using the Dirac generalised function described in Section 2.2.1. The range limits can be found in Table 2.3. The resulting fragment cloud distribution, $p_{a,e}$, is depicted in Figure 2.6 for both the sampled and analytically derived transformation. Bins with zero-probability, i.e. not affected by the collision or with a lack of samples, are transparent. Large parts of the probability are confined to a small region in the $a - e$ domain (note that the scale is logarithmic). Within just a few bin lengths, the density grows and shrinks by orders of magnitudes.

This is a result of both, the parent orbit geometry and the log-based breakup model. From the Jacobian in Equation (2.18), it becomes evident

Table 2.3: Range limits for bin geometries. The fraction, ϵ, describes the share of the distribution located within the limits.

Space $\boldsymbol{x}$	x_1	x_2	x_3	ϵ
(a, e)	$4800 - 17000$ km	$0 - 0.65$	$-$	97%
(a, e, Ω)	$4800 - 17000$ km	$0 - 0.65$	$0 - 40$ deg	96%
(ξ_p, ξ_a, Ω)	$1.5 - 6.7$ $\log_{10}$(m)	$1.8 - 7.3$ $\log_{10}$(m)	$0 - 40$ deg	94%

that in a first-order approximation, $(\Delta a, \Delta e)$ are independent of the out-of-plane impulse, Δv_h, and a is independent of the cross-track impulse, Δv_n. Furthermore, the dependence of Δe on Δv_n nearly vanishes at the point of fragmentation, i.e.

$$\begin{pmatrix} \Delta a \\ \Delta e \end{pmatrix} \approx \begin{bmatrix} \frac{2a^2 v}{\mu} & 0 \\ \frac{2(e+c_f)}{v} & -\frac{r s_f}{a v} \end{bmatrix} \begin{pmatrix} \Delta v_t \\ \Delta v_n \end{pmatrix} \tag{2.84a}$$

$$= \begin{bmatrix} 1.92 \times 10^3 & 0 \\ 2.68 \times 10^{-4} & 3.35 \times 10^{-6} \end{bmatrix} \begin{pmatrix} \Delta v_t \\ \Delta v_n \end{pmatrix} \tag{2.84b}$$

Hence, impulses in the along-track direction, v_t, have a large effect, dominating the effect of impulses in other directions and producing nearly linearly dependent $(\Delta a, \Delta e)$. Thus, many samples are found near the physical boundary defined by the radius of the parent orbit, e.g. either the perigee or apogee radius of the sample is close the parent orbit radius. As the impulse follows a log-normal distribution, much of the distribution remains close to the origin, while the tail end of the distribution produces few fragments that are far away from it. It is noteworthy that large parts of the distribution re-enters directly, i.e. the fragments have a perigee below the radius of Earth, $R_E = 6371$ km.

The underlying function poses problems to methods based on discretisation of the phase space (a, e). The physical boundary is not well captured by the bins. Some of the bins extend outwards of the boundary spreading probability into a forbidden domain, e.g. fragment probability is added for a perigee above the parent orbit perigee, and vice-versa for the apogee, which is physically not feasible. The steep increase and decrease of the underlying distribution leads to order of magnitude jumps in the estimated density. Further from the boundary, where the underlying density becomes small, more and more empty bins can be discerned. A few bins still contain a sample or two, but as a result overestimate their densities. Additionally, only a small strip in the $a - e$ domain is useful for subsequent processing of the fragment cloud, once re-entered fragments are removed. The analytically derived distribution, instead, exactly represents the underlying distribution in any region and captures the boundary smoothly.

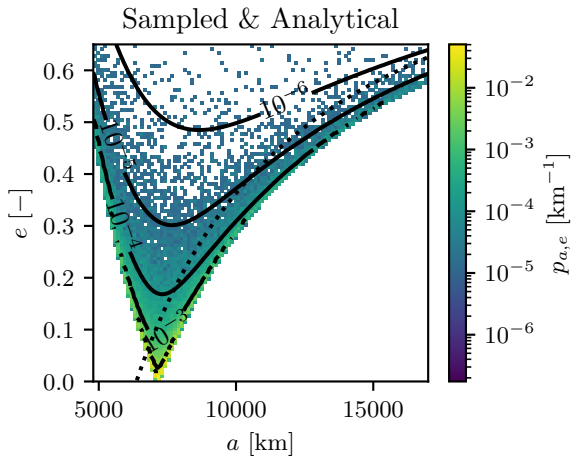
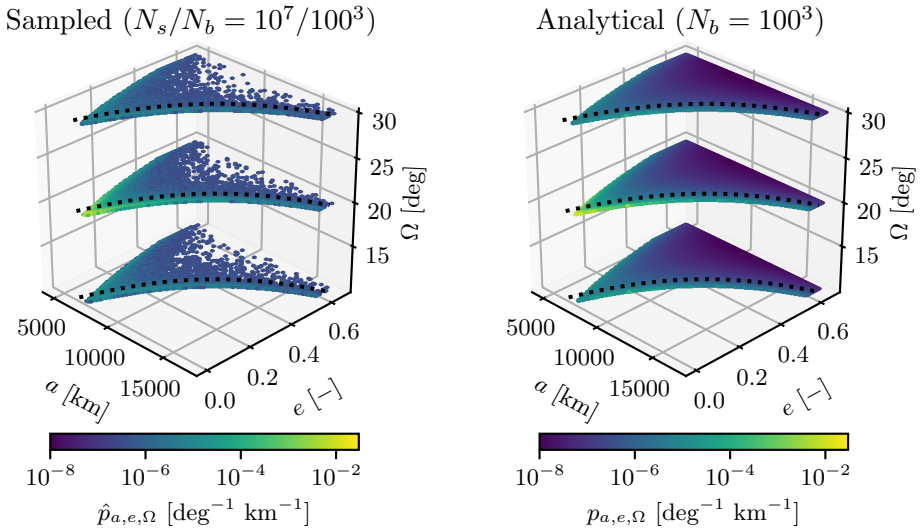


Figure 2.6: The marginalised distribution in (a, e) estimated through binning of $N = 10^5$ samples (colourmap) and analytical transformation using the Dirac generalised function (contours). The dotted line marks $r_p = R_E$, i.e. fragments with $e > e(r_p = R_E)$ re-enter within a single orbit from the fragmentation.

Keplerian target space

To study the spreading of the node and thus the mid-term evolution of the cloud, the distribution is required in a, e and Ω. The distribution, $p_{a,e,\Omega}$, is found through the transformation described in Section 2.2.3, by using a sub-matrix of $\boldsymbol{J_{\alpha s}}$ defined by the first, second and fourth row and the last three columns, relating $\boldsymbol{\Delta v}$ to (a, e, Ω). The distribution $p_{a,e,\Omega}$ is shown in Figures 2.7a and 2.7b for both, the estimated and analytical solution, for $N = 10^7$ samples and $n = 100 \times 100 \times 100 = 10^6$ bins. Only three slices are shown in the Ω-direction at $\Omega = 10$ deg, 20 deg and 30 deg. The colour bar is again logarithmic, and zero probability is depicted through transparent bins. The distribution is largely identical to the marginalised distribution, with the major part of the distribution again located near the radial boundary. As small impulses are more likely, the majority remains close to the parent node.

To answer the question about the proper bin size, it is worth investigating the convergence rates at various locations in the domain as the bin volume decreases, i.e. $N_b \to \infty$, and the number of samples increases, i.e. $N_s \to \infty$. The bins for the comparison are defined around a shortlist of $N_s = 1000$ random samples representing a wide range of density values, equally spaced in the logarithmic weight up to the sample carrying the maximum weight, $p^{max}_{a,e,\Omega}$. Figure 2.8a shows a projection of the samples and the selected bins, with a bin volume that would accommodate $N_b = 10^3$,



(a) Many of the $N_s = 10^7$ samples end up along the boundary, thus leaving large parts of the domain with empty (transparent) bins.

(b) The analytically derived transformation finds a probability even in the domains with near zero probability, resulting in a high sensitivity.

Figure 2.7: Comparison between the samples-based density, $\hat{p}_{a,e,\Omega}$, and analytically transformed density, $p_{a,e,\Omega}$. Only three slices are shown in Ω-direction.

i.e. 10 bins in each dimension. The diagonal subplots show the samples projected into a single dimension. Off-diagonal subplots show the projection into 2 dimensions. All the subplots use a logarithmic (colour-) scale in order to distinguish features. The v-shape in the $a - e$ plane can be readily observed. As before, the rectangular bins extend over the physical boundary. Hence, the estimated value is expected to underestimate the true density, as the bin incorporates a domain of zero probability.

The convergence of the binned distribution as a function of N_b and N_s are shown in Figure 2.8b and Figure 2.8c for $N_s = 10^9$ samples and $N_b = 100^3 = 10^6$, respectively. The estimates for bins around small densities converge already with a relatively large bin volume, corresponding to $N_b \geq 10^5$, well capturing the underlying density. This is because these samples are produced at the tail end of the log-normal distribution where change in the density is slow, i.e. the density is nearly constant over the bin. Bins closer to the peak of the distribution do not converge even for small bins corresponding to $N_b \geq 1000^3 = 10^9$ because the underlying distribution is too variable. For $N_b = 100^3 = 10^6$, the density around the bin with the maximum density is averaged out to a degree that evaluation of the estimated density underestimates the real density by 3 orders of magnitude.

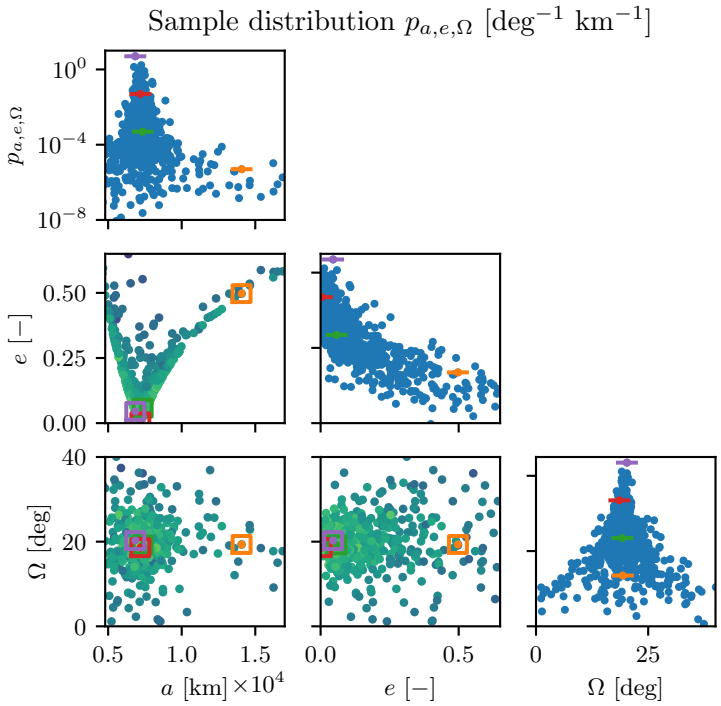
Any product that is derived from such an averaged density in (a, e), with or without Ω, such as collision probability estimations, will suffer from this inaccuracy. The averaging of the peak over the bin volume will lead to the under- or overestimation of the collision probability, again by orders of magnitude. Hence, for algorithms based on finite differences, the choice of target space is critical.

Bin-optimised target pace

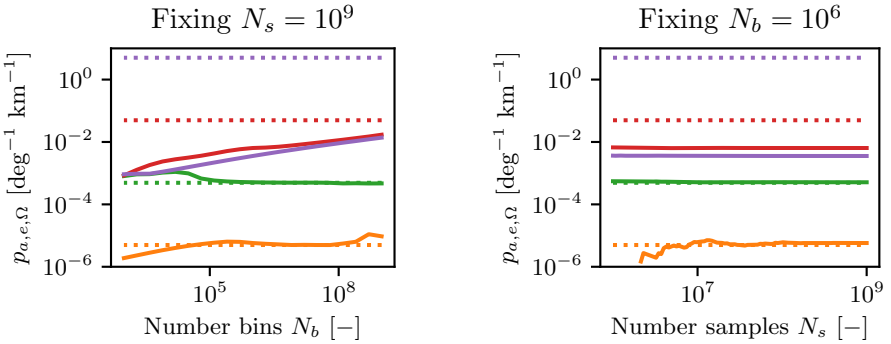
For near-circular orbits, or also for eccentric orbits fragmenting close to the apsides, a large part of the transformed distribution is concentrated into a small subspace of the $a - e$ domain. A more suitable subspace, capturing the logarithmic nature of the distribution is required. Thus, the distribution in a, e, and Ω, is further transformed into, $p_{\xi_p,\xi_a,\Omega}$, in ξ_a, ξ_p and Ω, with the auxiliary variables defined as

$$\xi_p = \log_{10}(r_0 - r_p) = \log_{10}(r_0 - a(1 - e)) \tag{2.85a}$$
$$\xi_a = \log_{10}(r_a - r_0) = \log_{10}(a(1 + e) - r_0) \tag{2.85b}$$

where r_0 is the radius of fragmentation, and r_p and r_a are the perigee and apogee radius of the parent orbit, respectively. The determinate of the Jacobian of the transformation, omitting the one-to-one relationship of Ω, is



(a) Projection of samples and bins chosen for comparison of the two estimation techniques. The represented bin limits show the bin volume accommodating $N_b = 10^3$.



(b) The distribution, being unequally distributed, converges only in domains with small densities.

(c) Hence, it is not possible to find a single bin volume that fits all the domain.

Figure 2.8: Distribution and convergence rates of the samples-based method (solid) against the analytical solution (dotted) in $\boldsymbol{x} = (a\ e\ \Omega)$.

$$\det \boldsymbol{J} = \det\left(\begin{bmatrix} \frac{\partial \xi_p}{\partial a} & \frac{\partial \xi_p}{\partial e} \\ \frac{\partial \xi_a}{\partial a} & \frac{\partial \xi_a}{\partial e} \end{bmatrix}\right) \tag{2.86a}$$

$$= \frac{1}{\log^2(10)} \det\left(\begin{bmatrix} \frac{-1+e}{r_0-r_p} & \frac{a}{r_0-r_p} \\ \frac{1+e}{r_a-r_0} & \frac{a}{r_a-r_0} \end{bmatrix}\right) \tag{2.86b}$$

$$= -\frac{2a}{(r_0 - r_p)(r_a - r_0)\log^2(10)} \tag{2.86c}$$

The following relationships can be found by taking the exponential of base 10 and summing and subtracting both sides of Equation (2.85)

$$r_0 - r_p = 10^{\xi_p} \tag{2.87a}$$

$$r_a - r_0 = 10^{\xi_a} \tag{2.87b}$$

$$2r_0 - 2a = 10^{\xi_p} - 10^{\xi_a} \tag{2.87c}$$

$$2ae = 10^{\xi_p} + 10^{\xi_a} \tag{2.87d}$$

Hence, the distribution function in the new space is obtained according to Equation (2.10) as

$$p_{\xi_p,\xi_a,\Omega} = \frac{10^{\xi_p+\xi_a}\log^2(10)}{2r_0 - 10^{\xi_p} + 10^{\xi_a}} p_{a,e,\Omega}\left(a, e, \Omega\right) \quad \forall\, 10^{\xi_p} < r_0 \tag{2.88}$$

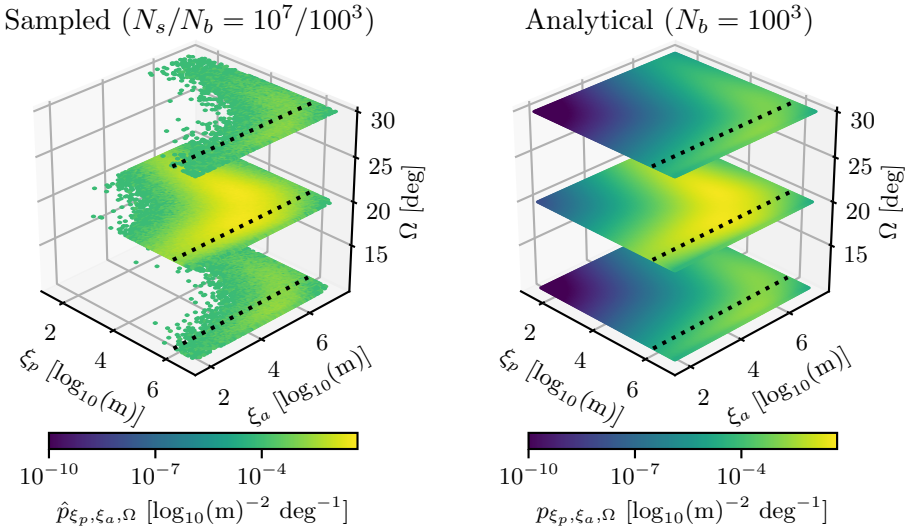
where

$$a = \frac{2r_0 - 10^{\xi_p} + 10^{\xi_a}}{2} \tag{2.89a}$$

$$e = \frac{10^{\xi_p} + 10^{\xi_a}}{2r_0 - 10^{\xi_p} + 10^{\xi_a}} \tag{2.89b}$$

with the domain restricted to positive values of the perigee radius, i.e. $r_p = a(1-e) > 0$.

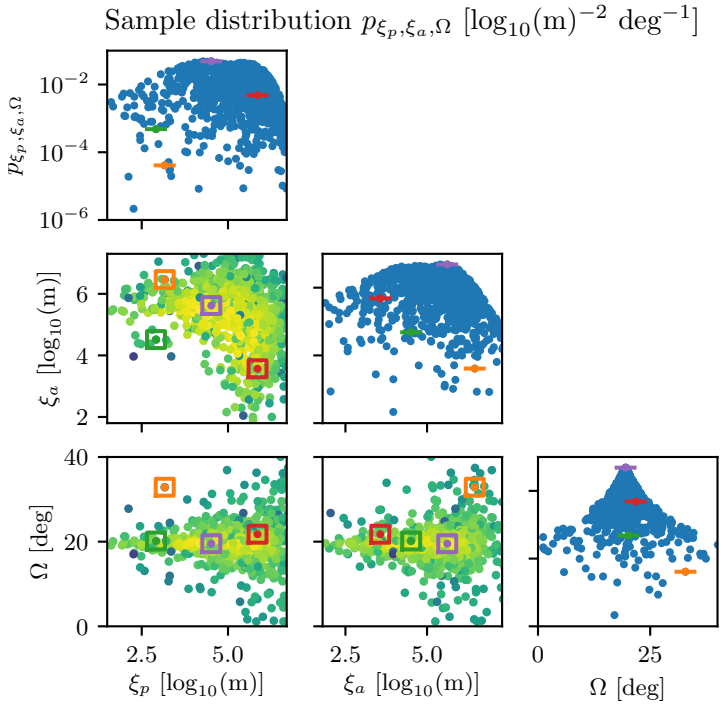
Figure 2.9 shows the distribution in the (ξ_p, ξ_a, Ω) space. The changes in density to neighbouring bins are small compared to the previous space. Additionally, only a small part of the domain describes re-entering fragments, as depicted with the dotted line corresponding to $\xi_p(r_p = R_E)$. Still, large parts of the domain contain empty bins, dotted by non-empty bins with very few samples. The slow convergence for low-probability events is evident when comparing the densities derived from sampling and analytically in selected bins. Figure 2.10a shows again $N_s = 1000$ transformed samples of the distribution and the projections of the 4 bins, with a volume corresponding to $N_b = 10^3$ or 10 bins in each dimension, selected to represent regions of different probabilities. As before, the selected samples are spread equally in the logarithmic density up to the maximum sample density $p^{max}_{\xi_p,\xi_a,\Omega}$. The rate



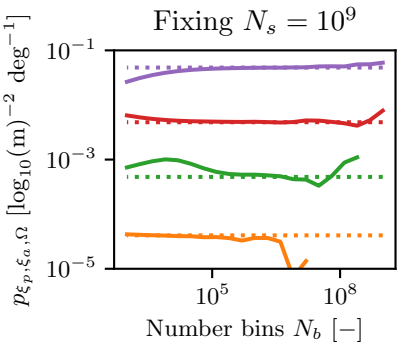
(a) The sampling leads to a non-smooth estimate of the distribution.

(b) Transformed analytically, the distribution can be estimated smoothly.

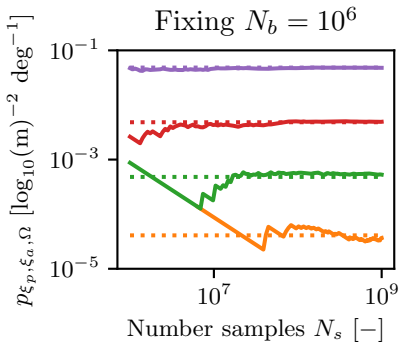
Figure 2.9: Comparison between the samples-based density, $\hat{p}_{\xi_p,\xi_a,\Omega}$, and analytically transformed density, $p_{\xi_p,\xi_a,\Omega}$. Only three slices are shown in Ω-direction. Fragments with $\xi_p > \xi_p(r_p = R_E)$ re-enter within a single orbit from the fragmentation.



(a) Projection of samples and bins chosen for comparison of the two estimation techniques. The represented bin limits show the bin volume accommodating $N_b = 10^3$.



(b) The distribution, being smoothly distributed, converges using a $N_b \geq 10^5$ for all regions.



(c) However, for $N_b = 100^3 = 10^6$ bins, convergence still requires $N_s \geq 10^8$.

Figure 2.10: Distribution and convergence rates of the samples-based method (solid) against the analytical solution (dotted) in $\boldsymbol{x} = (\xi_p\ \xi_a\ \Omega)$.

of convergence as $N_b \to \infty$ can be seen in Figure 2.10c for $N_s = 10^9$ samples. As the underlying distribution is not concentrated in small parts of the domain, convergence is achieved for a relatively large bin volume accommodating $N_b \geq 10^5$ over all regions with large or small probability. Hence, this space is well suited for estimating the underlying distribution function through sampling, as concentrated peaks are non-existent. Looking at the convergence rate in terms of N_s for $N_b = 10^6$ (see Figure 2.10c), shows that the estimated densities require more than $N_s \geq 10^8$ for convergence, showing the lack of sensitivity to capture low probabilities – or rare events – from deterministic samples (Au and Beck, 2001). Still, such optimised phase spaces are also important for the definition and fitting of the surrogate model, as will become apparent in Chapter 4.

2.6.2 Spatial density and directional distribution

For this example, the inverse transformation from a subset of Keplerian space to Cartesian coordinates is studied. To facilitate reproduction of the results, the example is detached from the results of the previous example. Instead, take a simple multivariate normal distribution in $(a,\, e,\, \Omega) \in \mathbb{R}^3$

$$p_{a,e,\Omega} = \mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \tag{2.90}$$

with

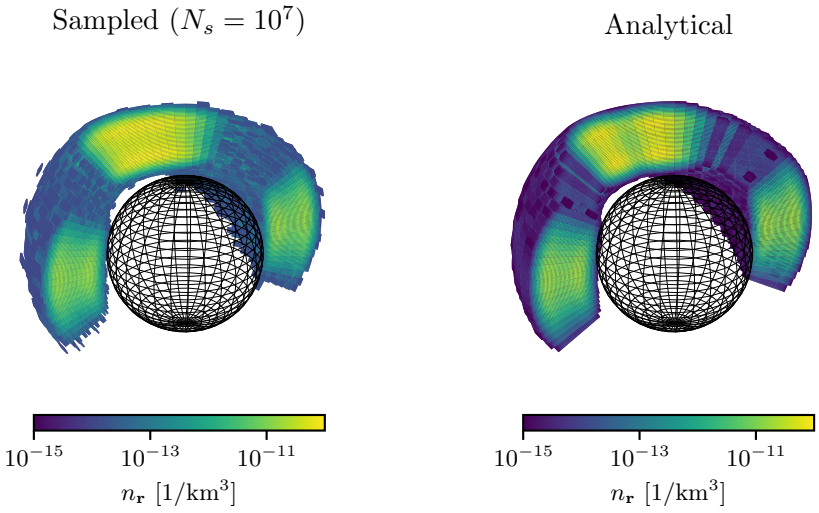
$$\boldsymbol{\mu}_k = (7200 \text{ km},\ 0.05,\ 180 \text{ deg})^T \tag{2.91a}$$

$$\boldsymbol{\Sigma}_k = \text{diag}(10^2 \text{ km}^2,\ 0.01^2,\ 9^2 \text{ deg}^2) \tag{2.91b}$$

The numbers are chosen as to resemble a breakup of an object in a slightly eccentric orbit. If the breakup occurs far from the apsides, it leads to a smooth distribution in a and e (e.g. see Section 6.1.1). The inclination is assumed to be fixed at $i_0 = 45$ deg, and randomisation is considered in ω and f according to Section 2.3. The spatial density, $p_{\boldsymbol{r}}$, may be evaluated through integration using Equation (2.43).

For comparison with the sampling method of estimating the density, $\hat{p}_{\boldsymbol{r}}$, the physical space is discretised into spherical bins. The radial direction, r, is split into 76 bins, with 25 km bin width, from $R_E + 100$ km to $R_E + 2000$ km. The longitude, λ, and latitude, θ, dimensions are split into 60 bins each, resulting in bin widths of 6 deg and 3 deg, respectively. At $r = R_E + 1000$ km, this corresponds to a very large bin size of 25 km $\times$ 775 km $\times$ 386 km.

The comparison of the analytical and estimated spatial density is depicted in Figure 2.11. Note that the figure is not drawn to scale to increase the visibility of the density distribution. Only half of the distribution is shown to allow insight into the cloud. Further hidden are empty bins for the estimated density and bins with density below $p_{\boldsymbol{r}} < 10^{-15}$ km^{-3} for the analytical density.



(a) Even using $N_s = 10^7$ samples, the sampling method can only estimate the densities over two orders of magnitude. Densities below are estimated to be zero.

(b) The analytically derived spatial density gives an estimate across the full domain. Densities below 10^{-15} km^{-3} are not shown.

Figure 2.11: Comparison between the samples-based spatial density, $\hat{p}_{\boldsymbol{r}}$, and analytically transformed spatial density, $p_{\boldsymbol{r}}$.

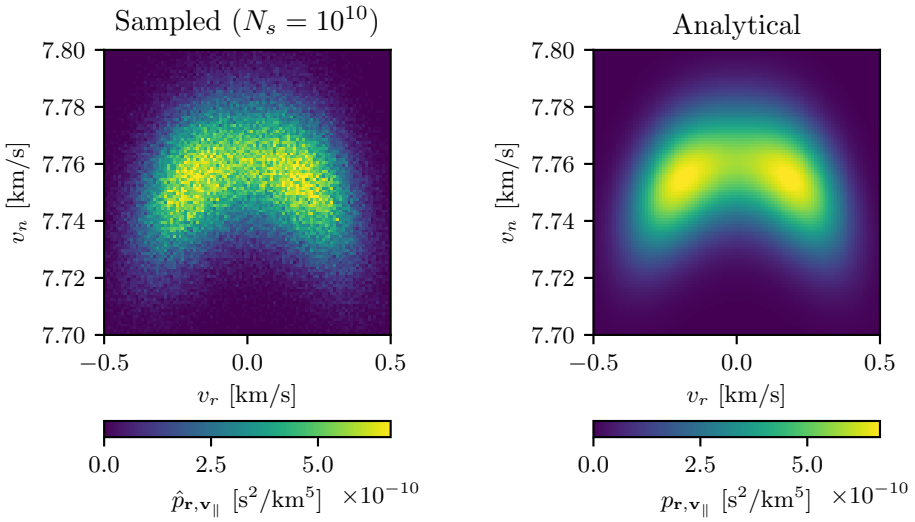
Not surprisingly, the density is highest at the latitudes close to the inclination because large parts of the distribution cross these bins. The estimated density, using $N_s = 10^7$, manages to predict the values only over two orders of magnitudes. Hence, effects of fragmentations are not felt if the density is below 1% of the maximum density. The analytical transformation, instead, gives a value of the density across the full domain, i.e. even for the smallest density regions.

Instead, if the impact rate, $\dot{\eta}$, is to be estimated from Equation (2.46), the directional distribution is required. Suppose a spectator sitting at

$$r = R_E + 530 \text{ km} \qquad \lambda = 0 \text{ deg} \qquad \theta = 0 \text{ deg} \tag{2.92}$$

At this point, there are two planes with non-zero incident fragments, in the planes rotated $\pm i_0$ around the x-axis. The first plane, at $+i_0$ corresponds to fragments at $\Omega \approx 0$ deg. The distribution, $p_{a,e,\Omega}$, vanishes for such an Ω. Instead, the second plane rotated for $-i_0$ corresponds to an $\Omega \approx 180$ deg. The plane orientation can be thought as the satellite frame, $\mathcal{T}$, for a circular orbit at r, i_0, $\Omega = 180$ deg and $u = 180$ deg.

In such a frame, the six-dimensional distribution collapses to a five-dimensional distribution. The analytical transformation, $p_{\boldsymbol{r},\boldsymbol{v}_\parallel}$, in positional and in-plane velocity components, is obtained through Equation (2.22). Instead of using the full Jacobian, $\boldsymbol{J}_{\boldsymbol{\alpha s}}$, the third row, corresponding to i, and



(a) Many samples are required to accurately estimate the underlying density.

(b) The analytical density evaluation is accurate and instantaneous.

Figure 2.12: Comparison between the samples-based, $\hat{p}_{\boldsymbol{r},\boldsymbol{v}_\parallel}$, and analytically transformed directional density, $p_{\boldsymbol{r},\boldsymbol{v}_\parallel}$.

the last column, corresponding to the out-of-plane velocity, v_h, are dropped. For the estimated density, $\hat{p}_{\boldsymbol{r},\boldsymbol{v}_\parallel}$, a box with the dimensions

$$\Delta r \pm 10 \text{ km} \qquad \Delta\lambda \pm 1 \text{ deg} \qquad \Delta\theta \pm 1 \text{ deg} \tag{2.93}$$

is defined around the point of interest. Only the samples that fall into the box are considered for further processing. Figure 2.12 compares the two densities. It takes hours to sample $N_s = 10^{10}$ and bin them accordingly. Still, the estimated distribution is not smooth. The analytical evaluation of the directional density, instead, is nearly instantaneous. Hence, the analytically derived phase space density is superior to an estimated one through sampling. Given an accurate value of the directional phase space density allows to accurately evaluate the impact rate.

In this chapter, the formulation of the fragmentation cloud as a continuum was discussed, including its advantages towards methods based on discretisation of the phase space. If the transformation of the phase space is known, so too can the phase space density be transformed. Assuming randomisation, the dimensionality of the phase space density can be chosen such as to match the desired time horizon. Without effort, the full orbital or Cartesian phase space density can be obtained from a density defined in a subset of it. Given the phase space density at an instance in time, the spatial density or the impact rate can readily be estimated without the need

of any limiting assumptions on the orbits of the target or chaser fragments. Now the question is, how does the phase space density evolve over time? This question will be answered in the following chapter.

3 Continuum propagation

In this chapter, the means to predict the evolution of the continuum describing the fragmentation cloud are presented. It is based on the continuity equation and solved using the method of the characteristics discussed in more detail in Section 3.1. The selection of the initial characteristics is presented in Section 3.2. Section 3.3 presents the semi-analytical propagator used for the numerical integration of the characteristics and discusses the perturbations from conservative forces and solar radiation pressure. Finally, Section 3.4 introduces a new atmosphere model and extended averaging technique for the fast and accurate propagation of objects subject to atmospheric drag. The recovery of an approximation of the continuum over the full domain is presented in Chapter 4. Note that parts of this chapter are adapted from (Frey et al., 2019b)[1].

3.1 Continuity equation

The cloud is modelled as a phase space density function, n. Hence, it is propagated using statistical mechanics. The continuity equation, accounting for sources and sink terms, g, is given as

$$\frac{\partial n}{\partial t} + \boldsymbol{\nabla} \cdot (n\boldsymbol{F}) = g \tag{3.1}$$

where $n(\boldsymbol{x}, t)$ is a function of the phase space variables, $\boldsymbol{x}$, and time, t, and $g(\boldsymbol{x}, t, n)$ is a function of $\boldsymbol{x}$, t and n. The dynamics, $\boldsymbol{F}(\boldsymbol{x}, t)$, define the trajectory of a phase state as

$$\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}t} = \boldsymbol{F} \tag{3.2}$$

[1] S. Frey, C. Colombo, and S. Lemmens. Extension of the King-Hele orbit contraction method for accurate, semi-analytical propagation of non-circular orbits. *Advances in Space Research*, 64:1–17, 2019. doi:10.1016/j.asr.2019.03.016. The datasets presented in Section 3.4 are available under https://doi.org/10.5281/zenodo.3048773.

Extending the divergence, $\boldsymbol{\nabla}\cdot$, in Equation (3.1) results in

$$\frac{\partial n}{\partial t} + \sum_{j=1}^{D}\left(\frac{\partial n}{\partial x_j}F_j\right) = g - n\sum_{j=1}^{D}\frac{\partial F_j}{\partial x_j} \tag{3.3}$$

where D is the dimension of the phase space. In Section 1.3, it is argued that the only viable way of solving the continuity equation for more than three dimensions and including sources and sink terms, is through the method of characteristics. The method of characteristics converts the partial differential equation in Equation (3.3) into a set of ordinary differential equations along the characteristic curves (see Equation (1.8))

$$\frac{\mathrm{d}n}{\mathrm{d}t} = g - n\sum_{j=1}^{D}\frac{\partial F_j}{\partial x_j} \tag{3.4a}$$

$$\frac{\mathrm{d}x_1}{\mathrm{d}t} = F_1 \tag{3.4b}$$

$$\vdots$$

$$\frac{\mathrm{d}x_D}{\mathrm{d}t} = F_D \tag{3.4c}$$

Which be be written in condensed form as

$$\frac{\mathrm{d}n}{\mathrm{d}t} = g - n\sum_{j=1}^{D}\frac{\partial F_j}{\partial x_j} \tag{3.5a}$$

$$\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}t} = \boldsymbol{F} \tag{3.5b}$$

The sources and sink term, g, can be used to model launch traffic, active debris removal or explosions and collisions (McInnes, 1993). Collisions, however, need to be treated with care. The source and sink term g can be modelled only as a function of $n(\boldsymbol{x}, t)$, i.e. the evolution is solved along the characteristic curve without awareness of n outside its current state, $\boldsymbol{x}$. McInnes (1993), for example, models the collision term, g_c, as

$$g_c \sim n^2 \tag{3.6}$$

This is acceptable for his model which considers only the radial distance, r, and assumes random relative collision velocities. However, if higher dimensional phase spaces are considered, the approximation is not valid. Consider a Keplerian phase space, i.e. $\boldsymbol{x} = \boldsymbol{\alpha}$. The approximation in Equation (3.6) then states that fragments which fly in the same orbit collide. This is physically infeasible as their relative velocity is zero. Fragments that have the potential to collide are to be found at different locations within the phase

space. E.g. considering circular, polar orbits: the most dangerous encounters for a fragment at Ω_0 would come head on from fragments at $\Omega_0 + \Delta\Omega$, with $\Delta\Omega = 180$ deg.

The collision rate could be incorporated through estimation of the impact rate (see Section 2.4.2). However, the update requires the knowledge of the density across the full domain. This knowledge is not available during the integration of a single characteristic which is only aware of its immediate surroundings. Instead, at fixed time steps, a full domain density estimate, introduced in Chapter 4, could be used to find a collision rate approximation. This approximation could be assumed to be constant over the next time span.

For the application of cloud propagation, inter-particle collisions are neglected. Nor are, analogously to Letizia et al. (2015), any other source or sink terms considered, i.e.

$$g = 0 \qquad \forall \boldsymbol{x}, t \tag{3.7}$$

Note that re-entering fragments are not considered a sink, rather they are a boundary condition. Hence, the solution along the characteristics in Equation (3.5) simplifies to

$$\frac{\mathrm{d}n}{\mathrm{d}t} = -n \operatorname{tr} \boldsymbol{J} \tag{3.8a}$$

$$\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}t} = \boldsymbol{F} \tag{3.8b}$$

with the trace, tr, defined as

$$\operatorname{tr} \boldsymbol{A} = \sum_{j=1}^{D} A_{jj} \tag{3.9}$$

and the Jacobian of the dynamics, $\boldsymbol{J}$, obtained through

$$J_{ij} = \frac{\partial F_i}{\partial x_j} \tag{3.10}$$

Thus, the integration of the characteristics requires not only the dynamics, but also its derivatives with respect to the phase space variables.

A simple example on how to find the evolution of an orbital continuum by solving the continuity equation through the method of characteristics is given in Appendix A.3. The example, considering only circular orbits and a simple atmospheric density model, even permits to find an analytical approximation of the solution. The evolution of the cloud is depicted in Figure 3.1. It shows both, the analytical solution as a surface plot and the numerically integrated characteristic curves as solid lines. An analytical solution can only be found if an analytical mapping of the trajectory is available. This is only possible for simplified dynamics and orbital geometry. Here, instead,

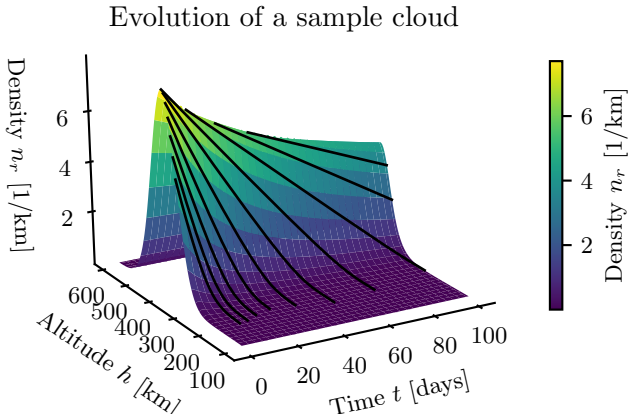


Figure 3.1: Example evolution of a cloud of fragments in circular orbit subject to atmospheric drag. The black lines are numerically integrated characteristic curves and the surface is obtained from the analytical solution.

selected initial states are integrated numerically, thus permitting any force model and geometries. The following sections discuss the selection of the characteristics and their propagation.

3.2 Selection of initial characteristics

The continuum is not propagated as a whole, but along the characteristics. Thus, a proper set of initial states that well represents the underlying distribution needs to be chosen. The trivial solution is to apply a regular grid to the phase space and assign a sample and evaluate its density at each grid corner. However, this method quickly runs into the same scaling issues inherent to discretisation methods: the number of samples grows exponentially in dimensions. Other methods covering the full domain are uniform random sampling or more elaborate techniques such as Latin hypercube sampling (McKay et al., 1979). E.g. in two dimensions, the plane is split into a square grid. If only one sample is present for each row and column, then it is a Latin square. The Latin hypercube is the extension to higher dimensions. The problem with such methods is that they do not consider the distribution they are supposed to represent. Many samples might end up in spaces that contain zero density.

It would be useful to have the most samples where the dynamics are the most divergent. However, this is not usually known a priori. Instead, sampling directly the distribution, rather than the domain, assures that samples are present where the bulk of the density is found. As such, the majority of the fragments is represented accurately, at least initially. If the inverse of the cumulative distribution function is known, uniform samples

can be converted into samples of the density function. The NASA SBM (see Section 2.5) is based on normal distributions and can easily be sampled using existing scientific computational routines. Other distributions might not permit finding the cumulative distribution function required for sampling.

A Metropolis-Hastings algorithm can be used to sample such initial conditions (e.g. see Chib and Greenberg, 1995). This algorithm describes a Markov Chain Monte Carlo (MCMC) method to sample multivariate distributions. The method is initialised at an arbitrary point, $\boldsymbol{x}_0$. A next step is randomly selected from a candidate-generating density, $q(\boldsymbol{x}_{s+1}|\boldsymbol{x}_s)$, probabilistically defining the likelihood of reaching a subsequent point, $\boldsymbol{x}_{s+1}$, given the current point, $\boldsymbol{x}_s$. Often, q is selected to be a multivariate normal distribution centred at $\boldsymbol{x}_s$. Then, q is symmetric, i.e. $q(\boldsymbol{x}_s|\boldsymbol{x}_{s+1}) = q(\boldsymbol{x}_{s+1}|\boldsymbol{x}_s)$, and the method reduced to the Metropolis algorithm. The step is accepted if the acceptance ratio

$$\alpha = \frac{n(\boldsymbol{x}_{s+1})}{n(\boldsymbol{x}_s)} \tag{3.11}$$

is higher than a uniform random sample $u \in [0, 1]$, else the step $\boldsymbol{x}_{s+1}$ is rejected. If the chain moves into a more probable point, i.e. $\alpha > 1$, the step is always accepted. If it moves into a less probable point, i.e. $0 < \alpha < 1$, it is accepted or rejected.

The Metropolis-Hastings requires some tuning, e.g. for the step size. The samples in a chain are correlated, hence only every k-th sample should be considered. Initial samples in the burn-in phase, i.e. the steps required to arrive from a low probable domain to a peak, should be dropped too as they do not accurately represent the distribution. For a sufficient number of iterations, the chain well represents the underlying distribution.

The problem with any density sampling method, such as direct sampling of a uniform or normal distribution or the Metropolis-Hastings algorithm, is that it lacks samples representing low probability regions. Especially samples obtained from non-convex distributions might give a misleading picture, as it is not always clear if gaps in the domain are due to the distribution or simply lack of samples. Similarly, initially convex distributions could be stretched and twisted into non-convex shapes by the dynamics. Still, as covering the full domain is not feasible in higher dimensions, density sampling is used herein. In particular, direct sampling of the normal distributions that make up the NASA SBM is performed to obtain initial characteristics. The issue related to the lack of samples in low probable domains is tackled in Chapter 4 concerned with the interpolation of the density from a cloud of points.

The selection of the initial points through sampling of a distribution resembles the selection of representative objects (Rossi et al., 1998). However, instead of assigning the same weight to all the samples, the initial density function is evaluated at the initial points, $n_c = n(\boldsymbol{x}_c)$ This results in the full description of the initial characteristics $(n_c, \boldsymbol{x}_c)$.

No easy measure for the required number of initial characteristics, C_0, is available. Figure 3.1 shows that the characteristics, following the trajectories, eventually decay and re-enter into the atmosphere, removing themselves from the cloud of fragments. Yet, being sampled from the initial distribution, the number of remaining characteristics, C, gives an estimate of the total remaining number of fragments, N, through

$$\widetilde{N} = \frac{C}{C_0} N_0 \tag{3.12}$$

E.g. if only $C = 10$ characteristics remain from an initial $C_0 = 1000$, that means that only 1% of the fragments remain, constituting a possible end point of the analysis. The number of required characteristics to train the surrogate model is discussed in Chapter 4. Once the initial points are selected, the characteristics can be numerically integrated using Equation (3.5). The propagator and the dynamics, i.e. the force model, used for the integration are discussed in the following two sections.

3.3 PlanODyn

So far, no phase space was mandated for the integration of the continuity equation along the characteristics. Many problems in varying phase spaces and dimensions can be modelled with the continuity equation. Examples are the evolution of interplanetary dust in semi-major axis and eccentricity (Gor'kavyi et al., 1997), of space debris in the orbital radius (McInnes, 1993; Letizia et al., 2015), spacecraft uncertainty during re-entry in nondimensionalised three- and six-state vectors (Halder and Bhattacharya, 2011) based on the equations derived by Vinh et al. (1979) and constellation satellites in the true anomaly (McInnes and Colombo, 2013), just to name a few.

Here, the interest is on the mid- to long-term propagation of fragmentation clouds. Orbital elements are well suited for this purpose as they permit integration in averaged dynamics and facilitate physical interpretation of the states. Thus, they enable quick propagation of states over tens to hundreds of years. Additionally, the Jacobian of the dynamics, $\boldsymbol{J}$, is needed for the integration of the characteristics. A well-suited propagator is the Planetary Orbital Dynamics (PlanODyn) suite (Colombo, 2016), briefly introduced here.

The PlanODyn suite is based on single averaged dynamics written in Keplerian elements, $\boldsymbol{\alpha}$. Conservative forces, i.e. gravitational forces of the non-spherical central body and Third Body Perturbations (3BPs) are considered using the Lagrange planetary equations (Battin, 1999, Section 10.2). Under some assumptions, also the perturbations from Solar Radiation Pressure (SRP) can be considered quasi-conservative. The non-conservative per-

turbations induced by atmospheric drag is considered through application of the Gauss planetary equation (Battin, 1999, Section 10.3).

The averaging is performed over the orbital revolution of the object of interest around the central body. Averaged perturbing effects from the geopotential are modelled up to a first-degree expansion for zonal (J_2, J_3, J_4) and tesseral (J_{22}) harmonics according to Kaula (1966, Chapter 3), and up to second-degree expansion for the second-order zonal harmonic (J_2^2) following Brouwer (1959).

3BPs are considered using the averaged approximation of Kaufman and Dasenbrock (1972). The ephemerides of the perturbing bodies, i.e. the Sun and Moon, are computed analytically according to Meeus (1998). The averaged effect of SRP is considered using the cannonball model, i.e. assuming the object of interest to be a sphere, and ignoring eclipses (Kaula, 1962; Krikov and Getino, 1997). The modelling of the perturbing effect of atmospheric drag is explained in more detail in Section 3.4.

The force model in PlanODyn is flexible and allows to turn on and off any perturbation. The dimension, D, of the phase space, $\boldsymbol{x}$, depends on the initial orbit, the force model and time-frame of interest. If only the long-term evolution of a cloud in LEO, subject to drag in a non-rotating atmosphere, is considered, only the semi-major axis, a, eccentricity, e, and area-to-mass ratio, A/m, are the required variables, as the other elements remain constant or can be assumed randomised under the effect of J_2 (see Section 2.3). The derivative of the variation of a constant element with itself is zero. Hence, it has no effect on the variation of n, which is dependent on the trace of $\boldsymbol{J}$ of the dynamics.

If, instead, the mid-term evolution of LEO objects, and the mid- to long-term evolution of non-LEO objects are of interest, the right ascension of the ascending node, Ω, and the argument of perigee, ω, become dependent variables. Furthermore, if 3BP and/or SRP are considered, the inclination, i, ceases to be constant. Lastly, if short-term evolution is of interest, an anomaly term, such as the true anomaly, f, needs to be considered. Hence, the propagation is performed in a subset of $\boldsymbol{\alpha}$ plus physical properties of the fragments, in the dimension that suits the scenario considered.

The flexible approach in dimensionality makes the method presented herein very powerful. Long-term propagation in a full orbital element set introduce large non-linearities. Hence, reducing the dimension and thus removing some of the non-linearities aids the recovery of the phase space density (see Chapters 4 and 6). However, there are limitations to the reduction of dimensionality. Depending on the type of orbit, certain perturbations cannot be ignored. E.g. for objects in geosynchronous and high area-to-mass ratios, SRP can have a considerable effect on the long-term evolution of the eccentricity and inclination (Valk et al., 2008). Thus, a careful trade-off between the accuracy and the benefit achieved through dimensionality

reduction is required for each scenario.

3.4 Atmospheric drag

The King-Hele (KH) method, derived anew in Appendix B.3, can be used to find an analytical expression of the effect of drag on the semi-major axis, a, and eccentricity, e, by averaging over the eccentric anomaly, E. I.e. the method describes a closed form solution of the integrals in Equation (B.32), repeated here in a simplified format as

$$F_a = \frac{\mathrm{d}\bar{a}}{\mathrm{d}t} = \frac{1}{2\pi}\int_0^{2\pi} \rho(h(a,e,E))g_a(a,e,E,B)\,\mathrm{d}E \tag{3.13a}$$

$$F_e = \frac{\mathrm{d}\bar{e}}{\mathrm{d}t} = \frac{1}{2\pi}\int_0^{2\pi} \rho(h(a,e,E))g_e(a,e,E,B)\,\mathrm{d}E \tag{3.13b}$$

with the density, ρ, at altitude, h, and the ballistic coefficient, B. The latter is defined as

$$B = c_D\frac{A}{m} \tag{3.14}$$

with the drag coefficient, c_D, the cross-sectional area, A, and the mass, m. Note that depending on the source, B is sometimes defined as the inverse of the definition given in Equation (3.14). The functions g_a and g_e, given in Equation (B.32), are independent of ρ. Using the KH approximation, the averaged variational equations require only a single evaluation of ρ at the perigee height, h_p. The method also permits to analytically find the contribution to the Jacobian of the dynamics, derived in Appendix B.4, which is important for the propagation of the phase space density, n.

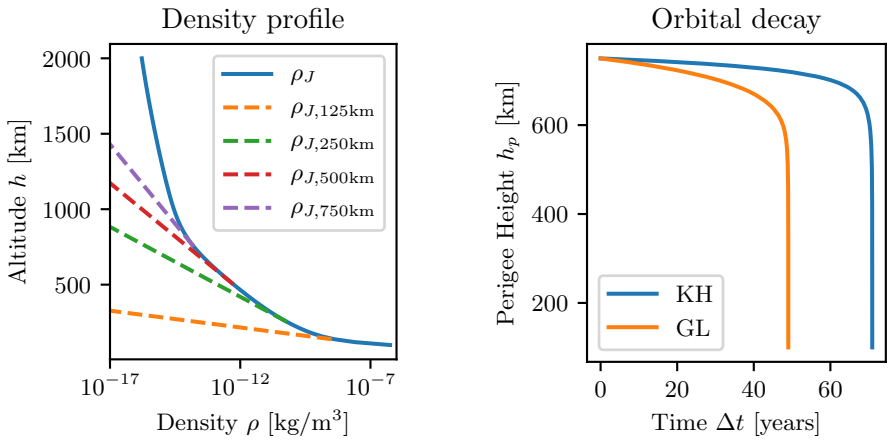
The caveat of the method is that, in order to obtain the closed form solution, it requires ρ to decay strictly exponentially in h, i.e.

$$\rho \propto \exp\left(-\frac{h-h_0}{H}\right) \tag{3.15}$$

with a base height, $h > h_0$, and a fixed scale height, H. The parameters can be obtained from any atmospheric density model, e.g. the Jacchia-77 model, ρ_J, as

$$h_0 = h_p \qquad H = -\frac{\rho_J(h_p)\Delta h}{\rho_J(h_p+\Delta h) - \rho_J(h_p)} \tag{3.16}$$

with $\Delta h = 1$ m and the perigee height, h_p. However, strictly exponential decay – a consequence of a fixed H – underestimates the true density at higher altitudes, such as the apogee height, h_a. Figure 3.2a shows the density profile according to ρ_J (solid), and the exponential approximation for parameters fixed at $h_p = 125$ km, 250 km, 500 km and 750 km (dashed). E.g. for an



(a) Fixing the density profile parameters at various altitudes leads to underestimation of ρ at higher altitudes.

(b) Underestimation of the density results in overestimation of the lifetime.

Figure 3.2: The KH approximation, fixing the parameters for the atmosphere density at h_p, underestimates the density at $h > h_p$ resulting in inaccurate lifetime estimates for eccentric orbits.

orbit at $h_p \times h_a = 500$ km $\times 1000$ km, the density value, $\rho_{J,500\text{ km}}$, fixed at $h_p = 500$ km, underestimates ρ_J at $h_a = 1000$ km by a factor of 10. Hence, the drag effect at altitudes above h_p are underestimated if strict exponential decay is assumed.

This, in turn, results in an overestimation of the lifetime of the object, if compared to numerical averaging techniques such as Gauss-Legendre (GL), that estimate the contraction rate by evaluating the true ρ_J at multiple heights. Figure 3.2b shows the effect of using the KH method by fixing the parameters in Equation (3.15) at different perigee altitudes on the lifetime estimate. The initial orbit configuration is set at $h_p \times h_a = 750$ km $\times$ 2000 km and the ballistic coefficient is set at, $B = 1$ m^2/kg. The difference in lifetime for the propagation using the numerical and analytical averaging techniques is +45%. Note that the same relative error is induced for any value of B. The biased errors introduced by the KH averaging method for highly eccentric orbits are of the same order of magnitude as uncertainties present in the estimation of the atmospheric density itself (Sagnieres and Sharf, 2017). Still, one of them is a modelling error of a highly complex environment, which is difficult to accurately predict. The other is a mathematical approximation error, which should and can be corrected.

Here, an extension of the KH method is presented. The new method is applicable – and accurate – to all eccentricity regimes, from circular to

Table 3.1: Non-exhaustive list of existing and newly proposed (in bold) atmospheric models and averaging methods. All analytical averaging methods make assumption on the shape of the atmosphere model.

Atmosphere models		Averaging methods	
References	Derivatives	Analytical	Numerical
CIRA	Non-smooth exponential	King-Hele (1964)	Simpson's rule
Jacchia (1977)	**Smooth exponential**	Vinh et al. (1979)	Gauss-Legendre (GL)
NRLMSISE		Sharma (1999)	
DTM		**Superimposed KH (SI-KH)**	

highly eccentric orbits. The idea is to find a mathematical formulation of the density model that suits the assumption of exponential decay without underestimating the density at higher altitudes. The proposed method thus consists of two parts: an atmosphere model based on constant scale heights, introduced in Section 3.4.1; and the extension of the KH formulation to reduce the errors induced by an atmosphere which in its sum does not decay exponentially, described in Section 3.4.2. The method is validated in Appendix D.

Table 3.1 shows an overview of how the proposed extension fits into the existing scheme of atmosphere reference models and derivatives thereof, and averaging methods. The technique presented here is not limited to the KH method but can be applied to any analytical averaging method based on a fixed scale height assumption.

3.4.1 Smooth exponential atmosphere model

The smooth atmosphere model proposed here does not in any way attempt to replace existing atmosphere density models. Instead, it is a derivation of those models. Nor is the idea of modelling the atmosphere as a sum of exponentials new: the Jacchia-77 reference model reduces – for each atmospheric constituent – to such a mathematical formulation if the vertical flux terms are neglected (Bass, 1980). The novelty of the method proposed here is the combination of the atmosphere model with the extended, superimposed KH formulation. A static and variable atmosphere model are presented in the following.

Static Model

The smooth exponential atmosphere model, ρ_S, is modelled by superimposing exponential functions as

$$\rho_S(h) = \sum_{j=1}^{N_p} \rho_j(h) = \sum_{j=1}^{N_p} \hat{\rho}_j \exp\left(-\frac{h}{H_j}\right) \tag{3.17}$$

where the number of partial atmospheres, N_p, the partial base densities, $\hat{\rho}_j > 0$, and the partial scale heights, $H_j > 0$, are fitting parameters. Note that here, the subscript j does not stand for altitude bins, but for one of the partial atmospheres, each of which is valid for the whole altitude range used for the fitting. It could potentially stand for a single atmosphere constituent, but it is not restricted as such. The parameters are strictly positive, as negative (base) densities are non-physical and negative scale heights lead to a density increase at higher h. However, the selection of positive parameters restricts the application of the model to certain altitude regimes. The rate of change and acceleration of ρ_S with respect to h are

$$\frac{\mathrm{d}\rho_S}{\mathrm{d}h} = -\sum_{j=1}^{N_p} \frac{\rho_j(h)}{H_j} < 0 \quad \forall h \tag{3.18a}$$

$$\frac{\mathrm{d}^2\rho_S}{\mathrm{d}h^2} = \sum_{j=1}^{N_p} \frac{\rho_j(h)}{H_j^2} > 0 \quad \forall h \tag{3.18b}$$

Hence, the model can only be applied to density profiles that monotonically decrease in h at ever slower rates.

Instead of finding the range of applicability through the acceleration in ρ, a qualitative study of the rate of change of the scale height is performed. The superimposed scale height, H_S, is

$$H_S(h) = -\frac{\rho_S(h)}{\mathrm{d}\rho_S/\,\mathrm{d}h} = \frac{\sum_{j=1}^{N_p} \rho_j(h)}{\sum_{j=1}^{N_p} \rho_j(h)/H_j} \tag{3.19}$$

Its derivative, i.e. its rate of change, with respect to h can be approximated through finite differences as

$$\frac{\mathrm{d}H_S(h)}{\mathrm{d}h} \approx \frac{H_S(h+\Delta h) - H_S(h)}{\Delta h} \tag{3.20a}$$

$$\approx -\frac{\rho_S(h+\Delta h)}{\rho_S(h+2\Delta h) - \rho_S(h+\Delta h)} + \frac{\rho_S(h)}{\rho_S(h+\Delta h) - \rho_S(h)} \tag{3.20b}$$

where $\Delta h > 0$ is a small step and $\frac{\mathrm{d}\rho_S}{\mathrm{d}h}$ is approximated using finite differences,

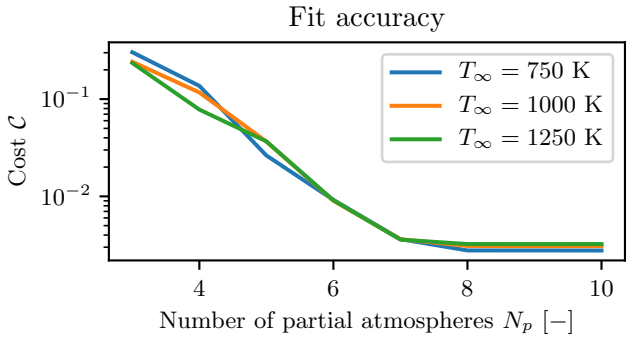


Figure 3.3: Cost function depending on number of partial atmospheres, N_p. The cost functions stop improving for $N_p > 8$.

too. From Equation (3.18), it follows, using again finite differences, that

$$\frac{\mathrm{d}\rho_S}{\mathrm{d}h} \approx \frac{\rho_S(h+\Delta h) - \rho_S(h)}{\Delta h} < 0 \quad \rightarrow \quad \rho_S(h+\Delta h) < \rho_S(h) \tag{3.21a}$$

$$\frac{\mathrm{d}^2\rho_S}{\mathrm{d}h^2} \approx \frac{\rho_S(h+2\Delta h) - 2\rho_S(h+\Delta h) + \rho_S(h)}{\Delta h^2} > 0 \tag{3.21b}$$

$$\rightarrow \quad \rho_S(h+2\Delta h) - \rho_S(h+\Delta h) > \rho_S(h+\Delta h) - \rho_S(h) \tag{3.21c}$$

Given the inequalities in Equation (3.21) and Equation (3.20b), it becomes evident that also H_S is monotonically increasing, i.e.

$$\frac{\mathrm{d}H_S(h)}{\mathrm{d}h} > 0 \tag{3.22}$$

Above $h = 110$ km, the Jacchia-77 density model, ρ_J, satisfies both requirements, a monotonically increasing ρ and H, for a wide range of exospheric temperatures, T_∞. Even if the underlying model shows slightly negative H at the lower boundary $h_0 = 100$ km (see Figure 3.5b), a partial atmosphere with a small positive H_j can still be fitted as is shown in the following. However, at lower altitudes, $h < 100$ km, the proposed model is not applicable.

To find the parameters, H_j and $\hat{\rho}_j$, the model in Equation (3.17) is fitted to ρ_J for three different T_∞: in accordance to a low solar activity, $T_\infty = 750$ K; mean solar activity, $T_\infty = 1000$ K; and high solar activity, $T_\infty = 1250$ K (see Figure 1.2). The fit is performed in the logarithmic space as not to neglect lower densities at higher altitudes, using least squares minimisation at heights between $h_0 = 100$ km and the upper boundary, $h_1 = 2500$ km. To put more weights on the edges of the fit interval, the densities are evaluated at $N_f = 100$ heights, h_i, distributed as Chebyshev

Table 3.2: Relative density fitting errors $\forall h \in [100\text{ km}, 2500\text{ km}]$.

ϵ_ρ	$T_\infty = 750$ K	$T_\infty = 1000$ K	$T_\infty = 1250$ K
$< 0.1\%$	$\forall h > 239$ km	$\forall h > 308$ km	$\forall h > 306$ km
$< 0.5\%$	$\forall h > 134$ km	$\forall h > 153$ km	$\forall h > 154$ km
$< 1.0\%$	$\forall h > 119$ km	$\forall h > 119$ km	$\forall h > 130$ km
$\epsilon_{\rho,max}$	1.6% ($h = 115$ km)	1.8% ($h = 115$ km)	1.9% ($h = 115$ km)

nodes (Abramowitz and Stegun, 1972, p. 889)

$$h_i = \frac{h_0 + h_1}{2} + \frac{h_1 - h_0}{2} \cos\left(\frac{2i-1}{2N_f}\pi\right) \qquad i = 1, \ldots, N \tag{3.23}$$

The number of partial atmospheres, N_p, is chosen to be 8, as the cost function

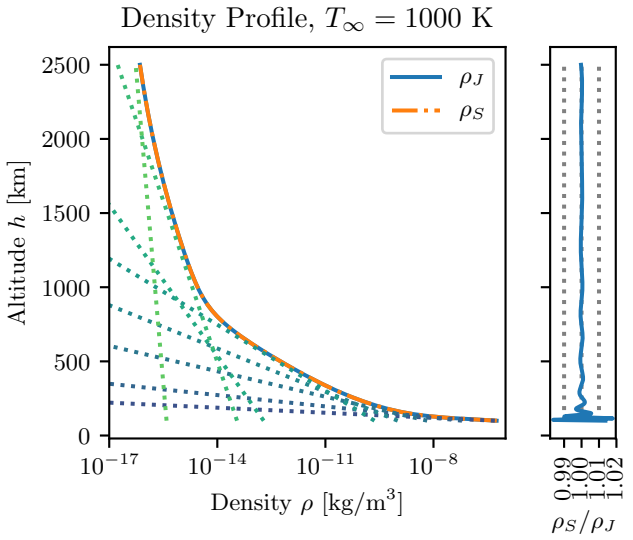
$$\mathcal{C} = \sqrt{\frac{1}{N_f} \sum_{i=1}^{N_f} \ln\left(\frac{\rho_S(h_i)}{\rho_J(h_i)}\right)^2} \tag{3.24}$$

which is the root mean square of the logarithmic density fit residuals, converges (see Figure 3.3). For $T_\infty \in [750\text{ K}, 1000\text{ K}, 1250\text{ K}]$, the relative error, ϵ_ρ, calculated as

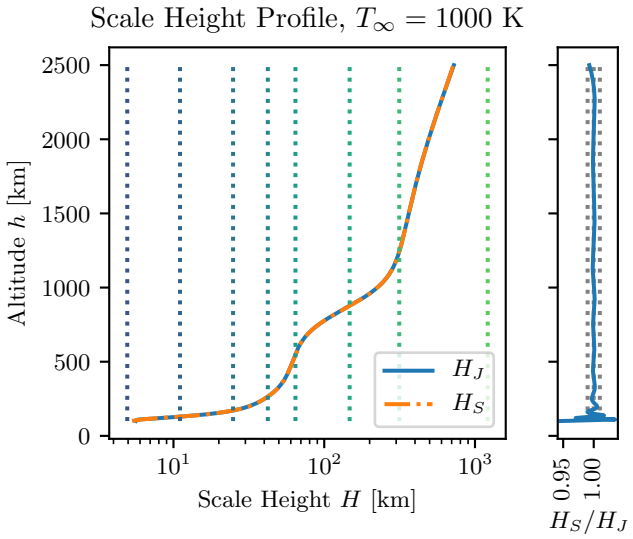
$$\epsilon_\rho(h) = \frac{|\rho_S(h) - \rho_J(h)|}{\rho_J(h)} \tag{3.25}$$

always remains below 0.1% and 1% for all $h > 308$ km and $h > 130$ km, respectively, and the maximum relative error, $\epsilon_{\rho,max}$, does not exceed 2%, as can be seen in Table 3.2. Hence, the density fit accurately represents the underlying model. The model parameters can be found in Table 3.3. Figure 3.4 shows a comparison between the underlying and fitted model, for T_∞= 1000 K. Next to the superimposed model, ρ_S, the individual contributions of the partial atmospheres, ρ_j, are shown. Note that despite fixed partial scale heights, H_j, the overall scale height, H_S, is variable and accurately follows the Jacchia-77 model. Figure 3.5 shows the fits at low altitudes, $h \in [100\text{ km}, 130\text{ km}]$ to show that despite increased inaccuracies in H, the fit, ρ_S, still accurately represents the Jacchia-77 density profile, ρ_J.

A speed test for hundreds of density and scale height evaluations over the range $100\text{ km} \leq h \leq 2500$ km shows a near 60-fold decrease in evaluation time for ρ_S compared to ρ_J. The implementation of the Jacchia-77 model used herein is written in the coding language C (taken from Instituto Nacional De Pesquisas Espaciais, 2020), and called from within MATLAB, while the routine to calculate ρ_S is implemented and called directly in MATLAB. Thus, a further decrease of computational time could be expected if also the latter was implemented in C. The speed tests were performed using the same processor architecture.



(a) Density profiles and relative error ρ_S/ρ_J.

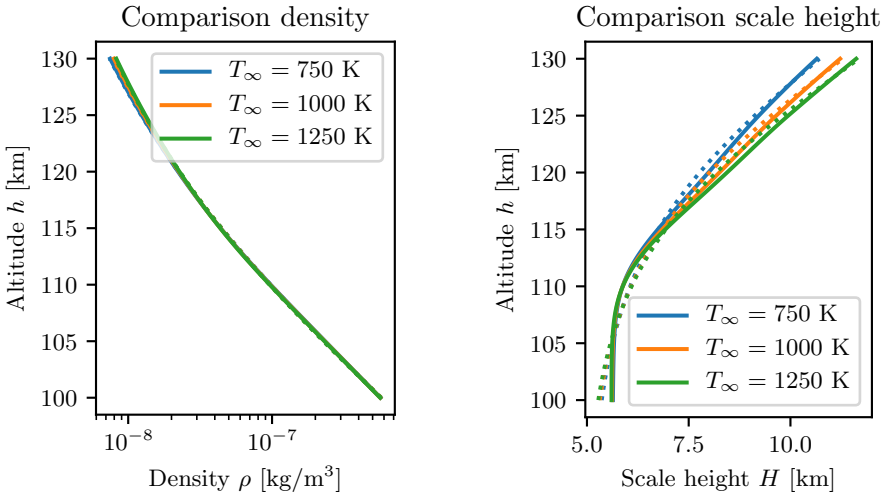


(b) Scale height profiles and relative error H_S/H_J. Note that the partial scale heights (dotted) are constant in h, while the superimposed scale height is not.

Figure 3.4: Fit of ρ_S to ρ_J for $T_\infty = 1000$ K. Additionally, the different contributions of each partial atmosphere are shown (dotted) from $j = 1$ (dark) to $j = 8$ (light).

Table 3.3: Smooth atmosphere model parameters resulting from a fit to the Jacchia-77 model, valid for altitudes $h \in [100\ \text{km}, 2500\ \text{km}]$.

	$T_\infty = 750$ K		$T_\infty = 1000$ K		$T_\infty = 1250$ K	
j	H_j [km]	$\hat{\rho}_j$ [kg/m^3]	H_j [km]	$\hat{\rho}_j$ [kg/m^3]	H_j [km]	$\hat{\rho}_j$ [kg/m^3]
1	4.995	2.496×10^{2}	4.936	3.163×10^{2}	4.903	3.640×10^{2}
2	10.47	8.465×10^{-4}	11.05	5.270×10^{-4}	11.44	3.818×10^{-4}
3	21.61	9.188×10^{-7}	24.85	3.735×10^{-7}	25.57	2.893×10^{-7}
4	37.81	1.253×10^{-8}	46.46	1.084×10^{-8}	44.92	1.246×10^{-8}
5	49.97	1.375×10^{-9}	64.44	1.088×10^{-9}	76.08	9.253×10^{-10}
6	174.2	1.593×10^{-13}	147.5	3.812×10^{-13}	111.1	1.667×10^{-11}
7	315.2	1.129×10^{-14}	314.5	4.843×10^{-14}	354.2	5.923×10^{-14}
8	1318.	3.807×10^{-16}	1215.	4.233×10^{-16}	892.2	1.738×10^{-15}



(a) The differences between the two models is hard to distinguish in the density profile.

(b) The errors are evident in H, where its derivative nearly vanishes or becomes negative in h.

Figure 3.5: Zoom into the lower altitude range, comparing ρ_J (solid) with its fit, ρ_S (dotted) for the different T_∞.

Variable Model

Possible extensions to the smooth exponential atmosphere model are the inclusion of a temporal dependence, such as the solar cycle, annual or daily variations. Here, the model is extended to incorporate the variability in the atmosphere density due to a variable T_∞. To conserve the mathematical formulation of the static model, the temperature dependence is introduced in the fitting parameters, $\hat{\rho}_j = \hat{\rho}_j(T_\infty)$ and $H_j = H_j(T_\infty)$.

T_∞ is a function of the solar proxy F (see Equation (1.35)), so the range of interest in T_∞ for the fitting is defined by F. Generally, the long-term predictions for F, based on various numbers of previous solar cycles, remain between 60 sfu and 230 sfu (Vallado and Finkleman, 2014; Dolado-Perez et al., 2015; Radtke and Stoll, 2016). As the smoothed solar proxy, $\overline{F}$, per definition remains in the same range as F, the given range translates into $T_\infty \in [669\ \text{K}, 1321\ \text{K}]$. The parameters for the variable smooth exponential atmosphere model derived below, and listed in Appendix C.2, are valid for any $T_\infty \in [T_0 = 650\ \text{K}, T_1 = 1350\ \text{K}]$. They should not be used for a T_∞ outside this range, as polynomial fits tend to oscillate strongly outside the fitting interval.

The dependence on T_∞ is incorporated using a polynomial least squares fit. Each partial atmosphere, ρ_j, is fitted separately on the values obtained through fitting static models at different T_∞. The static parameters, $\hat{\rho}_j$ and H_j in Equation (3.17), fitted to $i = 1, 2, \ldots, N_m$ static atmospheres, are converted

$$a_j^i = -1/H_j^i \tag{3.26a}$$

$$b_j^i = \ln\left(\hat{\rho}_j^i\right) \tag{3.26b}$$

where the fit space, a_j and b_j, is chosen such as to cope with high variability in densities. Each time-variable partial atmosphere is fitted to two independent polynomials of order l and m respectively

$$a_j(\widetilde{T}_\infty) = \sum_{k=0}^{l} a_{jk} \widetilde{T}_\infty^k \tag{3.27a}$$

$$b_j(\widetilde{T}_\infty) = \sum_{k=0}^{m} b_{jk} \widetilde{T}_\infty^k \tag{3.27b}$$

using a normalised and unit-less $\widetilde{T}_\infty$, defined as

$$\widetilde{T}_\infty = \frac{T_\infty - T_0}{T_1 - T_0} \tag{3.28}$$

In vector notation, Equation (3.27) can be written as

$$\boldsymbol{a} = \begin{bmatrix} a_1 \\ \vdots \\ a_{N_p} \end{bmatrix} = \begin{bmatrix} a_{10} & \dots & a_{1l} \\ \vdots & \ddots & \\ a_{N_p 0} & \dots & a_{N_p l} \end{bmatrix} \begin{bmatrix} \widetilde{T}_\infty^0 \\ \vdots \\ \widetilde{T}_\infty^l \end{bmatrix} \tag{3.29a}$$

$$\boldsymbol{b} = \begin{bmatrix} b_1 \\ \vdots \\ b_{N_p} \end{bmatrix} = \begin{bmatrix} b_{10} & \dots & b_{1m} \\ \vdots & \ddots & \\ b_{N_p 0} & \dots & b_{N_p m} \end{bmatrix} \begin{bmatrix} \widetilde{T}_\infty^0 \\ \vdots \\ \widetilde{T}_\infty^m \end{bmatrix} \tag{3.29b}$$

To prevent over-fitting, the order of the polynomials should remain well below the number of fitted static atmospheres. Here, the model in Equation (3.27) is fitted to $N_m = 50$ statically fitted models, distributed again as Chebyshev nodes between T_0 and T_1

$$T_i = \frac{T_0 + T_1}{2} + \frac{T_1 - T_0}{2} \cos\left(\frac{2i-1}{2N_f}\pi\right) \qquad i = 1, \dots, N_f \tag{3.30}$$

The orders are chosen to be $l = m = 8$ such that the error remains below 0.5% for all $h > 155$ km and $T_\infty \in [650\text{ K}, 1350\text{ K}]$. If more accuracy is needed, the polynomial order can be increased and/or spline polynomial interpolation applied. Finally, the time-dependent atmosphere is recovered by inverting Equation (3.26)

$$H_j(T_\infty) = -1/a_j(\widetilde{T}_\infty) \tag{3.31a}$$

$$\hat{\rho}_j(T_\infty) = \exp\Big(b_j(\widetilde{T}_\infty)\Big) \tag{3.31b}$$

Figure 3.6 compares the accuracy of the T_∞-variable smooth exponential atmosphere model, $\rho_S(T_\infty)$, against the original temperature dependent Jacchia-77 model, $\rho_J(T_\infty)$. It shows the ratio between $\rho_S(T_\infty)/\rho_J(T_\infty)$ for T_∞ in the range from 650 K to 1350 K (left), and the corresponding parameters, $\hat{\rho}_j$ and H_j as a function of T_∞, including the underlying parameters of the static fits (right). Towards the lower edge of the temperature range (i.e. $T_\infty \to 650$ K), the polynomial fits for components $j = 5 - 7$ do not well represent the underlying data. This leads to increased but still tolerable errors in the altitude range between 500 km and 1500 km. It is interesting to note that the parameters corresponding to the $j = 1 - 3$ partial atmospheres dominating the low altitude region $h < 200$ km (see Figure 3.4a) do not vary much in T_∞.

The advantage of this approach is that the original structure of the model is maintained, so it can be used with the contraction model introduced in the next section.

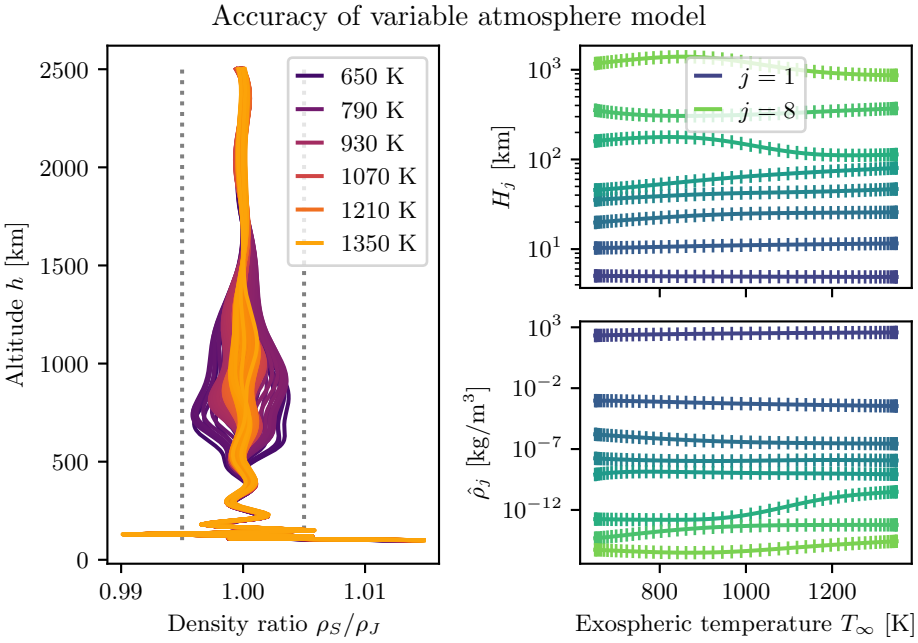


Figure 3.6: Quality of temperature dependent fit. Left: comparison for different T_∞. Right: evolution of H_j (top) and $\hat{\rho}_j$ (bottom) as a function of T_∞. The vertical lines indicate the value of the parameters of the static fits used to fit the variable model.

3.4.2 Superimposed King-Hele approximation

The extension of the KH contraction formulation into the Superimposed King-Hele (SI-KH) formulation with a superimposed atmosphere is straightforward. Substituting ρ from Equation (3.13) with the one defined in Equation (3.17) results in

$$F_a = \sum_{j=1}^{N_p} F_{a,j} = \frac{1}{2\pi} \sum_{j=1}^{N_p} \int_0^{2\pi} \rho_j(h) g_a(a, e, E, B) \, \mathrm{d}E \tag{3.32a}$$

$$F_e = \sum_{j=1}^{N_p} F_{e,j} = \frac{1}{2\pi} \sum_{j=1}^{N_p} \int_0^{2\pi} \rho_j(h) g_e(a, e, E, B) \, \mathrm{d}E \tag{3.32b}$$

i.e. each partial contraction reduces to the classical KH formulation with the partial exponential atmosphere $\rho_j(h)$. The important difference is that now H_j is constant over the whole altitude range. Hence, the resulting rates of change in a and e, averaged over a full orbit, can be obtained analytically through the KH method with only the approximation errors, introduced by the series expansion, remaining. The classical KH approximations for circular, low and highly eccentric orbits – and extended up to 5^{th} order – can be found in Appendix B.3. An extensive validation of the newly proposed averaging technique is given in Appendix D.

King-Hele (1964) introduced a simple fixed boundary condition $e_b = 0.2$ to select between the approximation method for low eccentric and high eccentric orbits, given in Appendix B.3.2 and Appendix B.3.3, respectively. However, as H_j can be large, this switch condition is not always sufficient. The low and highly eccentric series expansions are cut at 5^{th} order meaning the truncation errors are of order

$$\mathcal{O}\left(e^6\right) \qquad \text{and} \qquad \mathcal{O}\left(\frac{1}{z^6}\right) \tag{3.33}$$

where $\mathcal{O}$ is the order notation and

$$z = \frac{ae}{H} \tag{3.34}$$

For low a and high H, z can approach unity at $e = 0.2$, making the series expansion in $1/z$ inaccurate. Instead, it is proposed here to define e_b based on the truncation errors found in the formulations for the low and high eccentric orbits. The series truncation errors for the low eccentric orbit approximation in Equation (B.43) are of the order of

$$\mathcal{O}_a^{low}\left(e^6\right) = k_a \exp(-z) I_0 e^6 + \mathcal{O}\left(e^7\right) \tag{3.35a}$$

$$\mathcal{O}_e^{low}\left(e^6\right) = k_e \exp(-z) I_1 e^6 + \mathcal{O}\left(e^7\right) \tag{3.35b}$$

If z is large, $I_{0/1}(z) \to \exp(z)/\sqrt{2\pi z}$ (see justification below) and Equation (3.35) becomes

$$\mathcal{O}_a^{low}(e^6) \approx k_a \frac{e^6}{\sqrt{z}} \tag{3.36a}$$

$$\mathcal{O}_e^{low}(e^6) \approx k_e \frac{e^6}{\sqrt{z}} \tag{3.36b}$$

For the high eccentric orbit approximation in Equation (B.55), the truncation errors are in the order of

$$\mathcal{O}_a^{high}\left(\frac{1}{z^6}\right) = k_a \frac{1}{\sqrt{z}} \frac{(1+e)^{\frac{3}{2}}}{(1-e)^{\frac{1}{2}}} \frac{1}{z^6(1-e^2)^6} + \mathcal{O}\left(\frac{1}{z^7}\right) \tag{3.37a}$$

$$\mathcal{O}_e^{high}\left(\frac{1}{z^6}\right) = k_e \frac{1}{\sqrt{z}} \left(\frac{1+e}{1-e}\right)^{\frac{1}{2}} \frac{1}{z^6(1-e^2)^5} + \mathcal{O}\left(\frac{1}{z^7}\right) \tag{3.37b}$$

If the terms

$$\frac{(1+e)^{\frac{3}{2}}}{(1-e)^{\frac{1}{2}}} \frac{1}{(1-e^2)^6} \qquad \text{and} \qquad \left(\frac{1+e}{1-e}\right)^{\frac{1}{2}} \frac{1}{(1-e^2)^5} \tag{3.38}$$

are dominated by $1/z^6$ (see again below for a justification), Equation (3.37) simplifies to

$$\mathcal{O}_a^{high}\left(\frac{1}{z^6}\right) \approx k_a \frac{1}{\sqrt{z}} \frac{1}{z^6} \tag{3.39a}$$

$$\mathcal{O}_e^{high}\left(\frac{1}{z^6}\right) \approx k_e \frac{1}{\sqrt{z}} \frac{1}{z^6} \tag{3.39b}$$

Equating the truncation errors from Equations (3.36) and (3.39), using Equation (3.34) and solving for e results in the following, newly introduced, condition

$$e_b = \sqrt{\frac{H}{a}} \tag{3.40}$$

This switch condition allows to find the eccentricity value where the difference in the averaged rate of change calculated from the low and high eccentric approximation is smallest. Consequently, integrators using a variable step size experience a reduced effort in passing through the condition, if compared to a fixed boundary condition. Note that this boundary is most exact if the series expansions in both the low and high eccentric regimes are of the same order.

The derivation of the boundary condition required the assumptions of z to be large, such that $I_{0/1}(z) \to \exp(z)/\sqrt{2\pi z}$ and such that $1/z^6$ dominates the other e terms in Equation (3.37). To validate the assumptions, replace a

in Equation (3.40) with $a = (h_p + R_E)/(1 - e_b)$ and solve for e_b, neglecting the negative solution of the quadratic equation

$$e_b = \frac{1}{2}\left[-y + \sqrt{y^2 + 4y}\right] \qquad \text{where} \quad y = \frac{H}{h_p + R_E} \tag{3.41}$$

Given $H_{min/max} = 4.9/1320$ km (see Table 3.3) and the valid range for $h_p \in h \in [100\text{ km}, 2500\text{ km}]$, the extrema in e_b and $z_b = 1/e_b$, are found to be

$$e_{b,min/max} = 0.023/0.361$$
$$z_{b,min/max} = 2.77/43$$

For any $z > z_{b,min}$, $I_{0/1}$ remains close to $\exp(z)/\sqrt{2\pi z}$, being off only $+6\%$ and -16%, respectively, at $z_{b,min}$. At the same time, $1/z^6$ dominates the terms dependent on e in Equation (3.37) by two to three orders of magnitude $\forall e < e_{b,max}$. Thus, the assumptions made to derive e_b are valid.

An advantage of an analytical expression of the dynamics is that the Jacobian of the dynamics can be derived analytically too, which can be used for continuum and uncertainty propagation. The partial derivatives of the dynamics with respect to a and e, are given in Appendix B.4 (again dropping the subscript j). As the SI-KH method is simply a summation of the individual contributions of the partial atmosphere, the derivatives can equally be summed up as

$$\frac{\partial F_x}{\partial y} = \sum_{j=1}^{N_p} \left(\frac{\partial F_x}{\partial y}\right)_j \qquad (x, y) \in [a, e] \tag{3.43}$$

In this chapter, the initial selection and the propagation of the characteristic curves, containing the information about the fragment continuum, was discussed. The density can be propagated for any initial condition and force model. The force model and a new methodology of analytically incorporating the drag through averaged dynamics was presented. It removes the bias of underestimating the drag forces in the classical KH orbital contraction method, and achieves accuracies similar to averaging techniques based on numerical quadrature. Yet, the analytical formulation allows to analytically evaluate the Jacobian of the dynamics, which is required for the propagation of the dynamics. Hence, no numerical quadrature is required on the right-hand side of the ordinary differential equation, speeding up the integration process. However, the method of characteristics solves the continuity equation along the characteristic curves only. The next chapter shows how an approximation of the density, valid over the full domain, can be obtained.

4 Continuum interpolation

The previous two chapters describe how to formulate a cloud of fragments as a continuum and propagate them along the characteristics. The density, however, is only available along the characteristics. If propagated to the same epoch, the characteristics thus form a scattered point cloud in the phase space domain. If the spatial density or the number of impacts is to be estimated using Equations (2.39) and (2.46), an integration over all the incident velocity directions is required. The scattered point cloud, offering only limited knowledge about the full phase space density, is not sufficient to estimate the desired products.

Backpropagation of the density, i.e. starting from a point at the epoch of interest, propagating the characteristics backwards, and updating the density value using the initial distribution, could be an option. However, at each point in the three-dimensional physical space, there are an infinite number of directions and physical parameters, e.g. the area-to-mass ratio, to be considered. Most of these directions would be backpropagated to a domain far away from the initial density peak. Additionally, if the density is to be evaluated many times along an orbit and/or for many objects, the computational effort would quickly exceed the available resources.

Another solution is to interpolate between the scattered point cloud of characteristics to obtain the density across the full domain and thus avoiding performing costly propagations. An interpolation method suitable to the problem at hand is selected in Section 4.1. The fitting of the model and the selected optimisation scheme are discussed in Section 4.2. The initial condition, the dynamics and the selection of the phase space induce discontinuities and non-linearities. Section 4.3 shows how to deal with such distortions. Finally, the approach is validated against brute force sampling in Section 4.4. Additionally, a validation method is presented that is applicable in any dimension, D, as estimation from samples becomes inaccurate and inefficient for $D \geq 3$. Note that parts of this chapter are adapted from (Frey et al., 2020)[1].

[1] S. Frey, C. Colombo, and S. Lemmens. Evolution of orbital fragmentation cloud via

4.1 Selection of density estimation method

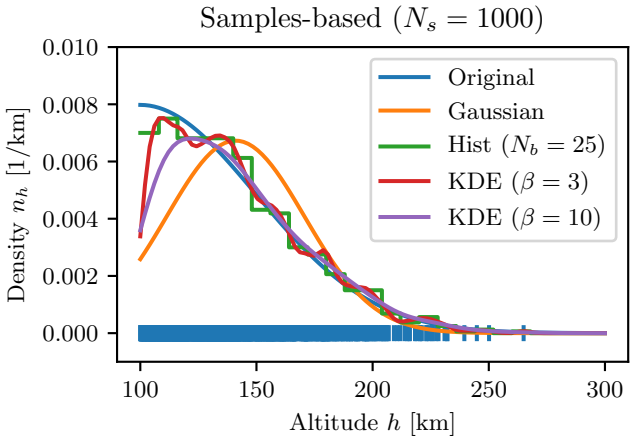
To select an appropriate surrogate model, the density estimation methods introduced in Section 1.3.5 are applied to a hypothetical breakup and discussed in terms of advantages and disadvantages. The hypothetical case is a breakup close to the re-entry altitude, $h_{re} = 100$ km, resulting in a cut-off distribution as shown in Figure 4.1. Albeit being a hypothetical case, such cut-off distributions are possible. The Cosmos-2251 collision fragmentation cloud, discussed in Section 6.1.2, shows exactly such a profile, as the energy of the collision is so high, that large parts of the fragments re-enter directly.

Samples-based, parametric estimation models cannot cope well with such distributions, as can be seen in Figure 4.1a for a Gaussian approximation found by taking the sample mean and variance, using a simple histogram obtained through discretisation of the phase space, and Kernel Density Estimation (KDE). The KDE solutions are obtained using Gaussian kernels with the bandwidths, $\beta = 3$ and $\beta = 10$, respectively. The density estimates, being probability density functions, are multiplied by a factor of $1/2$ in order to overlap with the original density that integrated $\forall h > h_{re}$ sums to $1/2$. As part of the samples are removed, the mean found using parametric estimation does not accurately describe the underlying distribution. Discretisation of the phase space, instead, produces a non-smooth density profile. KDE can cope better with the cut-off distribution than the Gaussian approximation. However, a general drawback of samples-based methods is that they cannot model distributions other than probability density functions. Additionally, estimates from non-parametric methods are hard to be interpreted in higher dimensions.

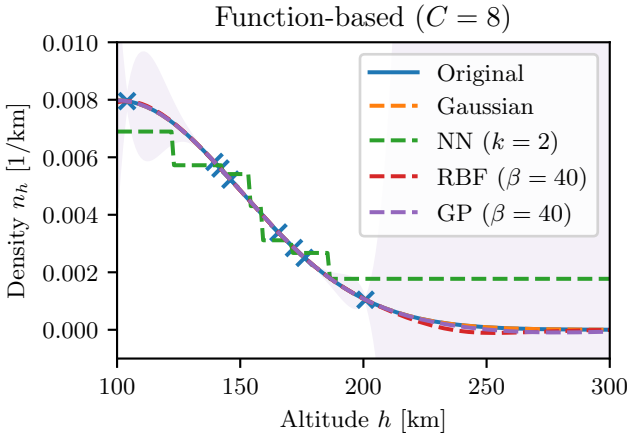
Figure 4.1b shows again the same underlying distribution together with function-based approximations based on $C = 8$ characteristics only. For each of the studied methods using hyperparameters, a manual selection of the optimal parameters was performed. The Gaussian fit is obtained through regression, the NN estimation uses $k = 2$ neighbours, the RBF interpolation uses a Gaussian kernel with $\beta = 40$, and the GP regressions uses the squared-exponential covariance kernel with a fixed $\sigma_s^2 = 1$ and $\beta = 40$. On top of the mean value obtained from the GP regression, the uncertainty is plotted as a shaded area, showing $\boldsymbol{\mu}_K \pm \boldsymbol{\sigma}_K$. Decent approximations are achieved with all the methods, despite the low number of observations.

However, NN produces non-smooth estimates. The non-continuous distribution present problems to subsequent integrations required for collision probability calculations. Another drawback is that the approximation is only accurate in the vicinity of samples. An advantage of RBF interpolation and GP regression is that, at the training points, the true function is approxi-

continuum and surrogate modelling. To be submitted to the *Journal of Guidance, Control, and Dynamics* in September, 2020.



(a) Sample-based methods, estimating the density from samples only, cannot accurately model re-entry distributions, even for a large set of samples, N_s (vertical lines).



(b) Density-based methods, taking the function value into account for interpolation, require only few data points, here $C = 8$ characteristics (crosses), to converge.

Figure 4.1: Density estimation for re-entry configuration for samples-based (upper) and function-based (lower) methods.

mated exactly. However, being non-parametric, the full set of training data and the inverse of a potentially large matrix is required every time a new data point is evaluated. Hence, these methods slow down with increasing number of training data. Additionally, functions approximated using RBF interpolation and GP regression are not inherently probability distribution functions, and thus might result in negative values. Negative fragment phase space densities or spatial densities are not feasible and should be avoided. GP regression is particularly useful, where an estimate of the output function, the prior, is available. This is not the case for the distribution describing fragments.

Here a GMM regression is chosen, i.e. a parametric, function-based method, for multiple reasons. First, it is a universal density function approximator (Goodfellow et al., 2016). Hence, it not only enables the estimation of probability distribution functions, but any density function, i.e. there are not restrictions on the value of the total integral. Additionally, the density estimate is easily restricted to positive values only. Second, it can handle multiple peaks, a necessity if multiple clouds or even the full debris population is to be modelled. Third, the Jacobian and even the Hessian of the cost function with respect to its parameters can easily be calculated. This helps the optimiser to converge to a minimum more quickly (see Section 4.2). Fourth, the total number of fragments can easily be inferred without the need of integration of the density. Finally, the GMM parameters can be interpreted even in higher dimensions.

The GMM is defined as (Bishop, 2006, Chapter 9)

$$\hat{n}_{\boldsymbol{x}} = \sum_{k=1}^{K} \pi_k \mathcal{N}_{\boldsymbol{x}|\boldsymbol{\theta}_k} \tag{4.1}$$

with the number of kernels, K, the kernel weights, $\boldsymbol{\pi} = (\pi_1, \pi_2, \cdots, \pi_K)$, the multivariate normal distribution, $\mathcal{N}$, as

$$\mathcal{N}_{\boldsymbol{x}|\boldsymbol{\theta}_k} = \frac{\exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right)}{\sqrt{(2\pi)^D \det(\boldsymbol{\Sigma})}} \tag{4.2}$$

and the parameters, $\boldsymbol{\theta}$, where $\boldsymbol{\theta}_k = (\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$ are the mean, $\boldsymbol{\mu} \in \mathbb{R}^D$, and the covariance, $\boldsymbol{\Sigma} \in \mathbb{R}^{D\times D}$, of the k^{th} kernel. The parameters of the surrogate model, $\boldsymbol{\Theta}$, are thus

$$\boldsymbol{\Theta} = \{(\pi_k, \boldsymbol{\theta}_k)\}_{k=1,\ldots,K} \tag{4.3}$$

The number of parameters, N_p, is

$$N_p = K\left(1 + \frac{3D + D^2}{2}\right) \tag{4.4}$$

which is a modest growth in D, considering that D remains below 10 for most of the envisioned applications of the method. The number of kernels, K, is

a hyperparameter to be chosen through cross-validation (e.g. Bishop, 2006, Chapter 1.3). A typical value for a well defined phase space is $K = 8$, resulting in $N_p = 80$ and $N_p = 120$ for $D = 3$ and $D = 4$, respectively.

Naturally, estimating the density by fitting a GMM comes with drawbacks too. If the underlying distribution is highly non-Gaussian, many kernels are required for an adequate fit. This problem can sometimes be alleviated by a proper selection of the fit space, as will be discussed in Section 4.3. Additionally, as the number of parameters grows in dimension and number of kernels, optimisation of the parameters can become slow. Hence, a proper selection of the optimiser is crucial, a topic which is addressed in Section 4.2

There is a distinct difference between the proposed method and previously developed GMM-based techniques for uncertainty quantification in non-linear systems (Terejanu et al., 2008; Horwood et al., 2011; Vishwajeet et al., 2014). There, extended or unscented Kalman filters are used to propagate the means and covariances of the kernels. Then, the weights of the kernels are updated by minimising the error of the fit versus the true value from the Fokker-Planck equation. Instead, the method proposed here leaves all the parameters, i.e. weights, means and covariances up for optimisation. This gives more freedom to the optimiser and ultimately results in a more accurate fit, especially for mid- to long-term propagation where non-linear effects become strong. The Kalman filters, however, could be used to provide an educated initial guess for the optimisation routine.

Still, the propagation of the continuum is closely related to non-invasive uncertainty propagation. Tardioli et al. (2015) compared multiple such approaches to study the evolution of space debris orbit uncertainties over the timespan of 10s of orbital revolutions. A more general review of uncertainty propagation techniques for problems using orbital mechanics was compiled by Luo and Yang (2017). Instead of considering all the fragments at once, only a single fragment could be considered probabilistically assuming large uncertainties around its physical state. However, such techniques usually do not consider boundary conditions, i.e. the total integral remains constant. Nor are they generally designed for long-term propagations where strong non-linear effects come into play. The method outlined in this thesis, instead, promises flexibility towards modelling of uncertainties and continua for any time frame. The application of the method of the characteristic, however, cannot be used to solve stochastic differential equations. Hence, it is not applicable to the propagation of a state in uncertain dynamics. Here, the initial fragment distributions are very large, justifying the neglect of uncertainties in the dynamics.

4.2 Model fitting

Initially, C_0 characteristics are sampled from a given fragmentation model. The characteristics are propagated to predefined epochs where the GMM is to be trained. At each epoch, the remaining C characteristics – i.e. the ones that did not re-enter in the meantime – are transformed into a feature space suitable for fitting. The transformed characteristics form the training data, $\boldsymbol{Y} \in \mathbb{R}^{C \times D+1}$. The training data consists of the extended phase space, i.e.

$$\boldsymbol{Y} = \{(\boldsymbol{x}_i, n_{\boldsymbol{x}_i})\}_{i=1,\dots,C} \tag{4.5}$$

with the phase space, $\boldsymbol{x}$, and the phase space density, n. Hence, a new GMM is trained at each epoch. Temporal interpolation between the models can be considered point-wise, e.g. using linear or piece-wise quadratic polynomials. If the time interval from epoch to epoch is short relative to the rate of evolution of the cloud, i.e. the changes in the shape of the cloud is marginal, the fit can even be assumed constant until the next epoch. The size of the time-step itself depends on whether the mid-term cloud evolution – considering Ω or ω in the phase space – or the long-term cloud evolution – not considering the two angles – is studied. Generally, quarter-yearly snapshots are sufficient for the long-term evolution scenarios, as the relative change in density is of the order of the modelling approximation errors.

The following three sections discuss the normalisation of the phase space, the cost function and its derivatives with respect to the parameters and the selection of the optimisation scheme.

4.2.1 Normalisation

Before running the optimisation algorithm, the characteristics are normalised. This step is required because the data is composed of features with various ranges, e.g. compare a and e for a near-circular fragmentation case, with their ranges in the order of 10^6 m and 0.1, respectively. The covariance of features with such large range differences can become nearly singular, creating difficulties for the computer routines handling them. The normalisation is performed as

$$\bar{\boldsymbol{x}} = \sqrt{\boldsymbol{\Lambda}}^{-1} \boldsymbol{V}^T (\boldsymbol{x} - \boldsymbol{\mu}_X) \tag{4.6a}$$

$$\bar{n}_{\boldsymbol{x}} = n_{\boldsymbol{x}} \prod_{d=1}^{D} \sqrt{\lambda_d} \tag{4.6b}$$

with the diagonal eigenvalue matrix, $\boldsymbol{\Lambda}$, and the eigenvector matrix, $\boldsymbol{V}$, as

$$\boldsymbol{\Lambda} = \operatorname{diag}(\lambda_1, \dots, \lambda_D) \qquad \boldsymbol{V} = [\boldsymbol{v}_1, \dots, \boldsymbol{v}_D] \tag{4.7}$$

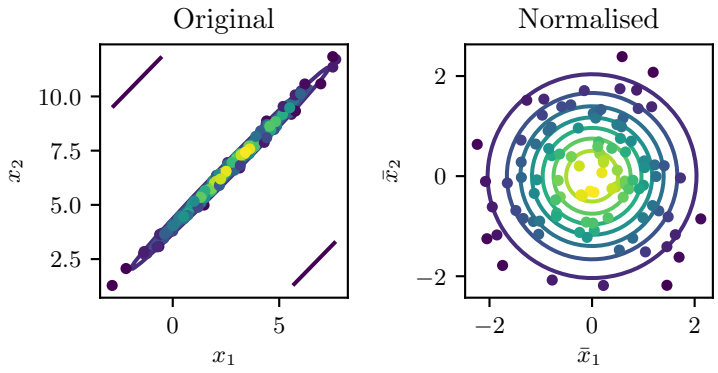


Figure 4.2: Example for normalisation of data. The original, highly correlated data on the left is hard to fit. Instead, the normalised data is not.

and the i^{th} eigenvalue, λ_i, corresponding to the i^{th} eigenvector, $\boldsymbol{v}_i$. The matrices $\boldsymbol{\Lambda}$ and $\boldsymbol{V}$ are obtained through eigenvalue decomposition of the centred and biased sample covariance $\boldsymbol{\Sigma}_{\boldsymbol{X}-\boldsymbol{\mu}_X}$, defined as

$$\boldsymbol{\mu}_X = \frac{1}{C}\sum_{c=1}^{C}\boldsymbol{x}_c \qquad \boldsymbol{\Sigma}_{\boldsymbol{X}-\boldsymbol{\mu}_X} = \frac{1}{C}\sum_{c=1}^{C}\left(\boldsymbol{x}_c - \boldsymbol{\mu}_X\right)\left(\boldsymbol{x}_c - \boldsymbol{\mu}_X\right)^T \tag{4.8}$$

Hence, the normalised characteristics have zero mean and a sample covariance corresponding to the identity matrix. An example of random characteristics distributed as

$$\boldsymbol{X} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) \qquad \boldsymbol{\mu} = \begin{pmatrix} 3 \\ 7 \end{pmatrix} \qquad \boldsymbol{\Sigma} = \begin{pmatrix} 5.01 & 4.99 \\ 4.99 & 5.01 \end{pmatrix} \tag{4.9}$$

and its normalised equivalent are shown in Figure 4.2. Note that the normalisation procedure in Equation (4.6) can be used for a principal component analysis (Bishop, 2006, Chapter 12). The study of the eigenvalues is useful for the feature selection and dimensionality reduction. If some of the eigenvalues nearly vanish, the samples can potentially be projected on the lower-dimensional space defined by the eigenvalues of the remaining eigenvalues. A reduction in dimensionality reduces the complexity of the fitting algorithm. However, the reduction could also lead to erroneous projections, as small variations at one instance in time could have large effects later on. Throughout this work, no principal component analysis is performed, as the dimensionality of the phase spaces is chosen to match the assumptions on the force model and duration of propagation.

The initial means for the fitting routine are found through a weighted k-means (Bishop, 2006, Chapter 9.1). The initial covariances are identity

matrices and the initial kernel weights are weighted with the number of samples assigned to each kernel. Optionally, few characteristics with very small relative density values can be discarded in a data pruning step before the fit. This step is executed by omitting the characteristics with the smallest densities, whose combined sum is less than a user-defined threshold, e.g. 10^{-5}, relative to the sum of the densities over all the characteristics. After the fit, each kernel is renormalised according to

$$\boldsymbol{\mu}_k = \boldsymbol{V}\sqrt{\boldsymbol{A}}\overline{\boldsymbol{\mu}}_k + \boldsymbol{\mu}_X \tag{4.10a}$$

$$\boldsymbol{\Sigma}_k = \boldsymbol{V}\sqrt{\boldsymbol{A}}\overline{\boldsymbol{\Sigma}}_k\sqrt{\boldsymbol{A}}\boldsymbol{V}^T \tag{4.10b}$$

giving the distribution in the original data set.

4.2.2 Cost function

The surrogate model is fitted to the normalised characteristics through minimisation of a cost function. The cost function, $\mathcal{C}$, using the parameters, $\boldsymbol{\Theta}$, given the observations, $\boldsymbol{Y}$, is defined as a regularised, logarithmic least squares function

$$\mathcal{C}_{\boldsymbol{\Theta}|\boldsymbol{Y}} = \mathcal{C}_C + \mathcal{C}_R = \frac{1}{C}\sum_{i=1}^{C}\left(\ln \hat{n}_{\boldsymbol{x}_i} - \ln n_{\boldsymbol{x}_i}\right)^2 + R\sum_{k=1}^{K}\pi_k^2 \tag{4.11}$$

with the regulariser, R, a second hyperparameter. Optimising in the logarithmic space allows to equally consider all characteristics. This is important because considerable parts of the distribution are to be found at the tails of the GMM. The regularisation term is required to inhibit the creation of kernels with large covariance values and large weights. Such kernels potentially lead to accurate fits with the centre of the kernel being outside of the actual cloud of characteristics.

To simplify the selection of R, i.e. to find a value that is valid for any distribution, the density values of the characteristics are divided by the estimate of the total integral, i.e. the estimated total number of fragments $\widetilde{N}$ from Equation (3.12), converting $n_{\boldsymbol{x}}$ into an approximation of a probability density function, i.e.

$$\int_{\mathbb{R}^D}\frac{n_{\boldsymbol{x}}}{\widetilde{N}}\,\mathrm{d}^D\boldsymbol{x} = \frac{1}{\widetilde{N}}\int_{\mathbb{R}^D} n_{\boldsymbol{x}}\,\mathrm{d}^D\boldsymbol{x} = \frac{N}{\widetilde{N}} \approx 1 \tag{4.12}$$

As such, both, $\mathcal{C}_C$ and $\mathcal{C}_R$ defined in Equation (4.11), remain nearly constant independent of the value of N. Thus, a R that works for $N = 10^6$ fragments works also for $N = 10^3$ fragments. Note that the optimised kernel weights must be multiplied by $\widetilde{N}$ to obtain the correct density distribution. If the number of fragments estimated from the model

$$\hat{N} = \int_{\mathbb{R}^D}\hat{n}_{\boldsymbol{x}}\,\mathrm{d}^D\boldsymbol{x} = \sum_{k=1}^{K}\pi_k \tag{4.13}$$

is very different from $\widetilde{N}$, that either means that the fit is not accurate, or that the peak of the distribution lies close to the re-entry altitude, as shown for the example depicted in Figure 4.1.

The optimal parameters, $\boldsymbol{\Theta}^*$, are the ones minimising the cost function

$$\boldsymbol{\Theta}^* = \arg\min_{\boldsymbol{\Theta}} c_{\boldsymbol{\Theta}|\boldsymbol{Y}} \tag{4.14}$$

The optimisation problem is bounded in kernel weights, $\pi_k > 0$, and in the diagonal of the covariances, $\operatorname{diag}(\boldsymbol{\Sigma}_k) > 0$, increasing the complexity of the optimisation. Instead, the cost function can be optimised in logarithmic weights, $\boldsymbol{\kappa}$, and lower triangular matrices, $\boldsymbol{L}$, defined as

$$\kappa_k = \ln \pi_k \qquad \boldsymbol{\Sigma}_k = \boldsymbol{L}_k \boldsymbol{L}_k^T \tag{4.15}$$

As such, the parameters are unbounded. Given that the derivatives of the cost function with respect to its parameters can be easily calculated (see Appendix B.5), a wealth of optimisation schemes becomes applicable.

A suitable optimisation method to fit the GMM surrogate model to the scattered cloud of is the memory limited version of the Broyden–Fletcher–Goldfarb–Shanno (BFGS) algorithm (Nocedal and Wright, 2006, Chapter 6.1). It is an implementation of the *Quasi-Newton method*, that instead of calculating the expensive Hessian analytically at every optimisation step maintains its approximation using gradient information only. More information about BFGS and various other optimisation methods and their benchmarking are given in Appendix E.

4.3 Reference hypersurface

The GMM is a universal density function approximator, i.e. as $K \to \infty$, any distribution can be represented with it. However, the number of parameters to be fitted grows linearly with K, hence, keeping K low ensures a fast convergence of the optimisation routine.

For the modelling of evolving fragmentation clouds, three different causes lead to distributions that do not resemble normal distributions or sums thereof. First, the initial fragmentation cloud, following an explosion or collision, can be of any shape. According to the NASA SBM, the initial distribution both in Δv and A/m are given as log-normal distributions. Second, the transformation from Cartesian coordinates to a subset of Keplerian elements can introduce bounded distributions. E.g. an explosion of an object in a circular orbit produces the typical V-shape distribution in a and e present in Figure 2.7. An elegant way to address this is to convert the distribution into other phase space variables, such as one defined relative to the apogee radius, r_p, and the perigee radius, r_a, in Figure 2.9. However, such a phase space cannot always be found, especially for evolving fragment

clouds. Third, the dynamics are non-linear and linear approximations are not applicable for long-term propagation. Thus, even an initially normally distributed fragmentation cloud will eventually get distorted.

Here it is proposed to use hypersurfaces with which a suitable relative phase space for the fitting is defined. The reference hypersurface is defined through samples. These samples are mapped from the initial characteristics upon the initially analytically defined hypersurface. The reference hypersurface samples are propagated alongside the characteristics and subsequently interpolated to obtain the reference hypersurface.

An example of an evolving distribution using a reference hypersurface is given in Figure 4.3 for a two-dimensional distribution. The initial distribution in Figure 4.3a is obtained by sampling the breakup model for an explosion of an object in circular orbit. The distribution, shown in r_p and r_a, shows clear boundedness in both directions. This follows from the fact that no fragment can have a perigee above or an apogee below the initial radius, r_0. The characteristics are propagated subject to atmospheric drag, leading to a decrease in r_p and r_a and distorting the distribution as is depicted in Figure 4.3b. It would be hard to fit a GMM to any of those distributions because of their non-smooth boundedness and their logarithmic distribution. Instead, a reference hypersurface is introduced to obtain the same samples in a new phase space, more suited for fitting the GMM. The reference samples defining the hypersurface are introduced by mapping the initial characteristics to $r_a = r_0$ and, separately, $r_p = r_0$ (Figure 4.3c), and propagated to the same epochs as the characteristics (Figure 4.3d). The relative phase space

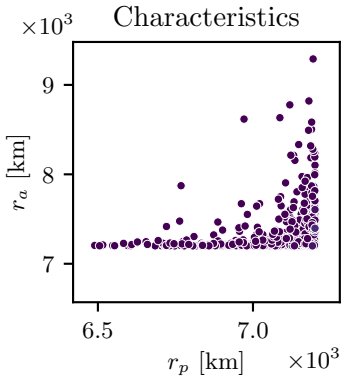
$$\xi_p = \log_{10}(r_0 - r_p) \tag{4.16a}$$

$$\xi_a = \log_{10}(r_a - \hat{r}_a(r_p)) \tag{4.16b}$$

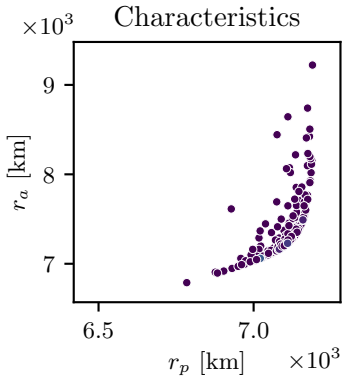
where $\hat{r}_a$ is the reference hypersurface interpolated from the reference samples reveals a well-behaved fragmentation cloud distribution which can be fitted effortlessly at any epoch (Figures 4.3e and 4.3f).

Forward and backward transformation of the phase space density still enables the evaluation of the density in the original phase space. However, the introduction of the reference hypersurface comes with a drawback. The final distribution is not available in a purely analytical fashion but requires the interpolation of reference samples. Theoretically, also the reference hypersurface could be obtained through a parametric fit of the samples, enabling a fully analytical surrogate model. However, this would add more complexity to an already complicated procedure.

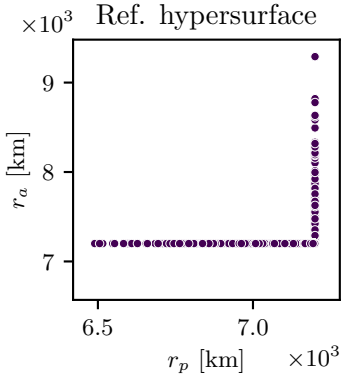
The introduction of reference hypersurfaces is useful for bounded initial conditions. Fragmentations close to the perigee and close to the apogee can be fit more easily with reference samples near the perigee and apogee, respectively. For long-term propagation of highly eccentric orbits, the non-



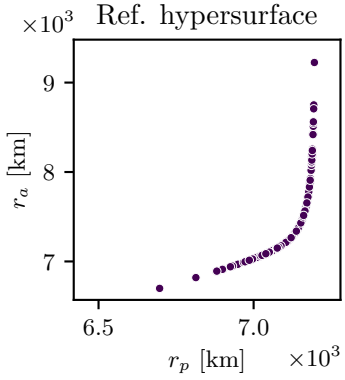
(a) Initial characteristics.



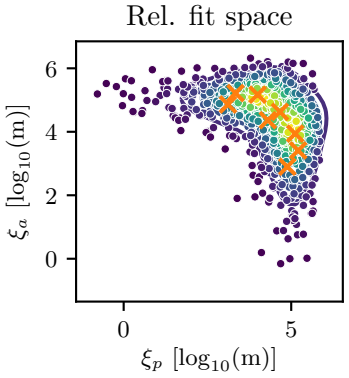
(b) Propagated characteristics.



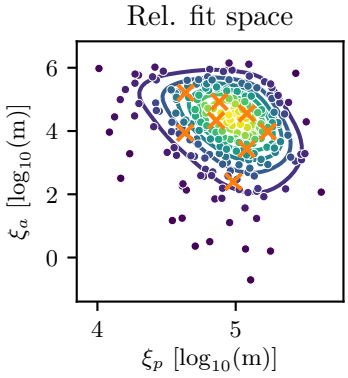
(c) Initial hypersurface samples.



(d) Propagated hypersurface samples.



(e) Initial characteristics mapped to the relative phase space.



(f) Evolved characteristics mapped to the relative phase space.

Figure 4.3: Initial (left) and propagated (right) characteristics (top), hypersurface samples (centre), and relative phase space fits (bottom).

linearities become very strong but can be mitigated by introducing a hypersurface within the initial scattered cloud of characteristics. An example of such a case is given in Section 6.1.4.

4.4 Validation methods

Two methods are presented here to validate the interpolation of the phase space density from a scattered cloud of characteristics. The first validation method compares the density estimate to observations not used during the fitting. For this, a validation set of $N_s = 10^4$ characteristics is sampled from the initial condition. It is propagated forward in time to the same snapshots as for the fits. Hence, a true value of the density is available for comparison with the approximations obtained from the surrogate model. For each characteristic at each snapshot, the relative error is calculated as

$$\epsilon_n = \frac{\hat{n}_{\boldsymbol{x}} - n_{\boldsymbol{x}}}{n_{\boldsymbol{x}}} \tag{4.17}$$

where $(\boldsymbol{x}, n_{\boldsymbol{x}})$ is the validation set and $\hat{n}_{\boldsymbol{x}}$ is the fitted surrogate model evaluated at the validation sample point, $\boldsymbol{x}$. Inspection of the quantiles of $\epsilon_n^{(j)}, j = 1, \cdots, 10^4$ over the full validation set allows to draw conclusions about the quality of the fit.

A second validation is performed following the Monte Carlo approach, using $N_s = 10^6$ initial samples. The sample density for comparison is obtained through binning, as discussed in Section 2.6, performed in the fit space. This allows the estimate obtained to converge with a comparatively small number of samples. The bins are centred around a few validation characteristics representing a wide range of density values, from the minimum to maximum, logarithmically spaced in the density value.

The application of validation methods is performed here for a single fragmentation evolution example in three dimensions. More scenarios, each validated according to the first validation technique outline here, are presented in Chapter 6. Validation through binning is not performed for these $D \geq 3$ cases, as the number of required samples to obtain convergence in the estimate exceeds the computational resources at hand. The characteristics and hypersurface samples were propagated, and the fits performed, on a parallelised computer architecture. The setup consisted of 4 Intel® Xeon® E5-4620 v4 Central Processing Units (CPUs), with a total of 40 cores. The propagation of $N_s = 10^6$ samples requires nearly 4 hours even on the parallelised setup.

The simulated fragmentation is a rocket body explosion, modelled with the NASA SBM, on a circular orbit at $r_0 = 7200$ km. A normalised phase space density is used, i.e. the distribution initially defines a probability density function. Integration of the characteristics is performed considering

atmospheric drag only in the phase space, $\boldsymbol{x} = (a,\ e,\ A/m)$. The target phase space for the fitting

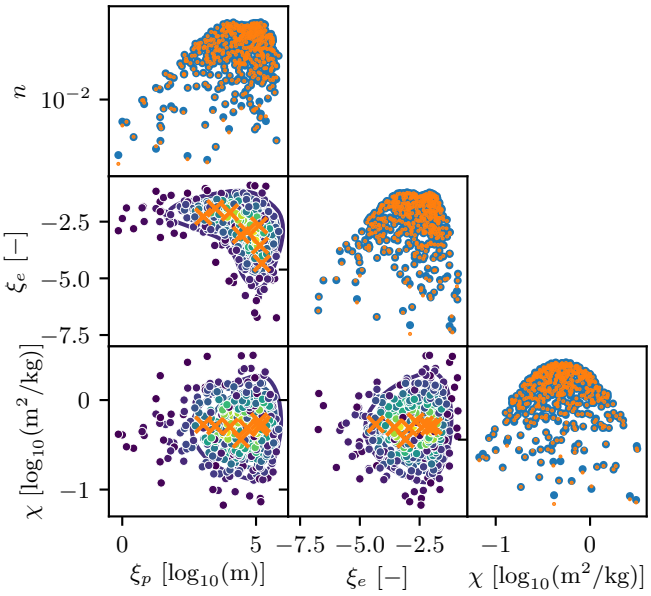
$$\xi_p = \log_{10}\left(r_0 - a(1-e)\right) \tag{4.18a}$$
$$\xi_e = \log_{10}\left(e - \hat{e}(\xi_p, \chi)\right) \tag{4.18b}$$
$$\chi = \log_{10}\left(A/m\right) \tag{4.18c}$$

is defined relative to a two-dimensional hypersurface, $\hat{e}$, which is obtained from hypersurface samples through a cubic spline interpolation.

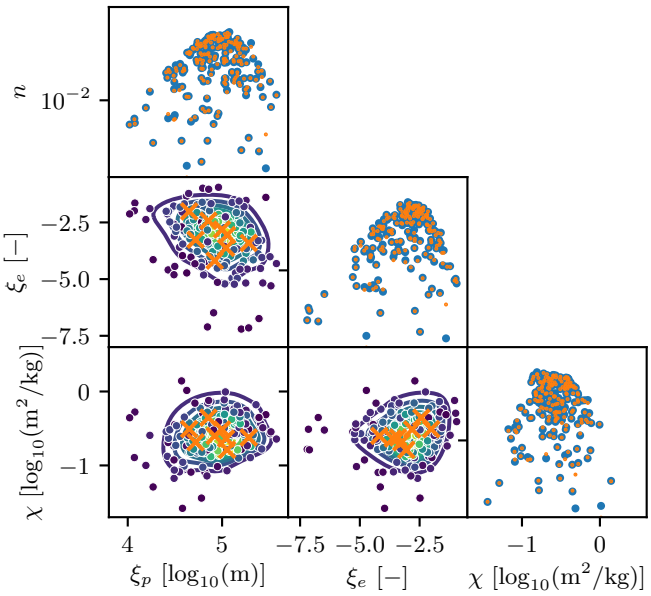
The selection of the number of characteristics, C, and hypersurface samples requires careful consideration. As a rule of thumb, twice as many characteristics as number of parameters, N_p, to be fitted, are required, i.e. $C > 2N_p$. As parts of the characteristics, following the trajectories of regular fragments, re-enter during the propagation, the number of characteristics available at later stages cannot be known initially. Here, $C_0 = 1000$ initial characteristics are sampled and propagated for $\Delta t = 15$ years and collected in 181 monthly snapshots. After 15 years, $C = 186$ characteristics remain, the rest have re-entered. The fits at each snapshot are performed using a regulariser of $R = 10^{-2}$ and $K = 8$ kernels, corresponding to $N_p = 80$ fit parameters. The hyperparameters, R and K, are found through cross-validation and valid for any of the snapshots. The characteristics are propagated alongside $2C_0 = 2000$ hypersurface samples, obtained through mapping of the characteristics to the initial radius, r_0, on the perigee and apogee radius. The propagation of the characteristics and reference samples is performed within 2 minutes on the parallelised setup. The optimisation routine terminates if the new cost function improves less than a relative tolerance of 10^{-5} compared to the previous iteration. On average, the $N_p = 80$ parameters are fit to each of the 181 snapshots in 3.1 s. Hence, the fitting requires 14 s on the parallel setup. The total process of propagating and interpolating the density takes 2.25 minutes, outperforming the Monte Carlo approach in terms of computational time by two orders of magnitude.

The initial and final distribution of the characteristics, given in the fit space and including the approximation from the surrogate model, are presented in Figure 4.4. Each figure shows a one-dimensional representation of n as a function of each variable in the diagonal. The value of $\hat{n}$ in orange is superimposed on the true value of n in blue. The off-diagonal sub-figures show a two-dimensional representation of the scattered characteristics and the fit as a contour plot. The centres of the kernels are marked with a cross. Through visual inspection, the fits seem accurate for both snapshots and for any density regime.

Figure 4.5 shows the percentiles of the errors as a function of time since the fragmentation, Δt. It shows that 50% of the errors (between $q_{25\%}$ and $q_{75\%}$) remain within an error range of -13% and $+7\%$. The 80% from $q_{10\%}$ and $q_{90\%}$ remain between -33% and $+22\%$. The upper percentile, $q_{95\%}$,



(a) Initial characteristics right after the fragmentation.



(b) Characteristics $\Delta t = 15$ years after the fragmentation event.

Figure 4.4: Evolution of a fragmentation cloud in the phase space suitable for fitting. Of the initially $C = 1000$ characteristics, only $C = 180$ remain after $\Delta t = 15$ years of propagation.

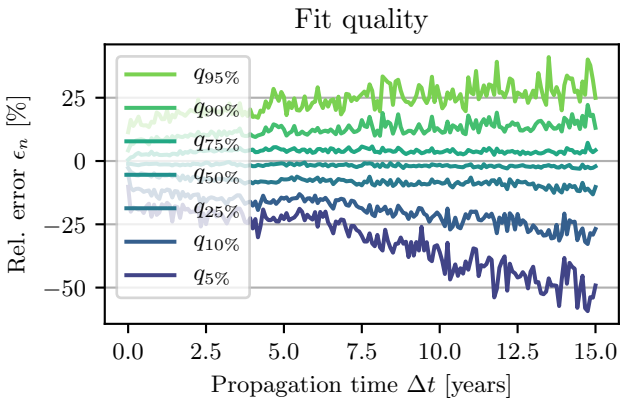
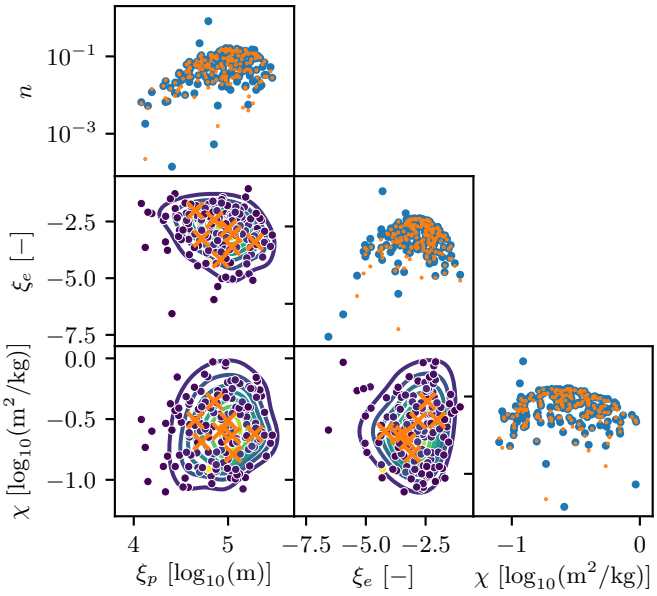


Figure 4.5: Evolution of the relative fitting error percentiles. Due to the underestimation of small density regions, the relative error profile is biased.

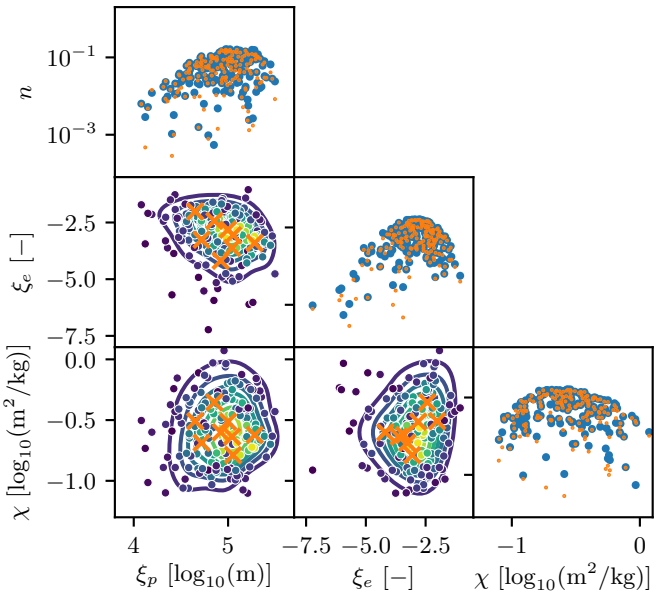
grows steadily from an initial 12% but remains below 40% for all snapshots. The lower percentile, $q_{5\%}$, grows slowly at first from initially -20% to -30% after $\Delta t = 7.5$ years, but then starts to grow faster. Still, $|q_{5\%}|$ remains below 60% for all snapshots. Hence, despite a bias towards underestimating the true density, the surrogate model represents the underlying density well.

To understand where the bias comes from, a new set of $N_s = 10^3$ validation characteristics is propagated to the final snapshot. Additionally, its mapped values are propagated as hypersurface samples. Figures 4.6a and 4.6b compare the new validation characteristics against the original fit, using the hypersurface samples mapped from the fit characteristics, and the ones mapped from the validation characteristics, respectively. In Figure 4.6a a few validation characteristics hover above the distribution seemingly as outliers. They are not present in Figure 4.6b. This indicates that the reference hypersurface is not accurate across the full domain, a consequence of re-entering hypersurface samples. Another observation is that characteristics far from the peak of the distribution, i.e. with low densities, are underestimated given either hypersurface, leading to the bias discerned in Figure 4.5. What about the outliers?

Figure 4.7 shows the comparison of the estimate from the surrogate model against the true density value from the validation set and the estimate from the Monte Carlo sampling. Four snapshots at $\Delta t = 0$ years, 5 years, 10 years and 15 years are shown. The bin sizes are varied with the number of bins, N_b, i.e. a large N_b corresponds to a small bin size and vice-versa. The fitted density from the surrogate model is integrated over the bin size, hence it changes with increasing volume. Note that from the initially drawn 10^6 samples, more than 6×10^5 re-enter or escape directly, leaving just under $N_s = 940000$ samples remaining for the initial snapshot. For the first three



(a) Relative to the hypersurface sampled from the fit characteristics.



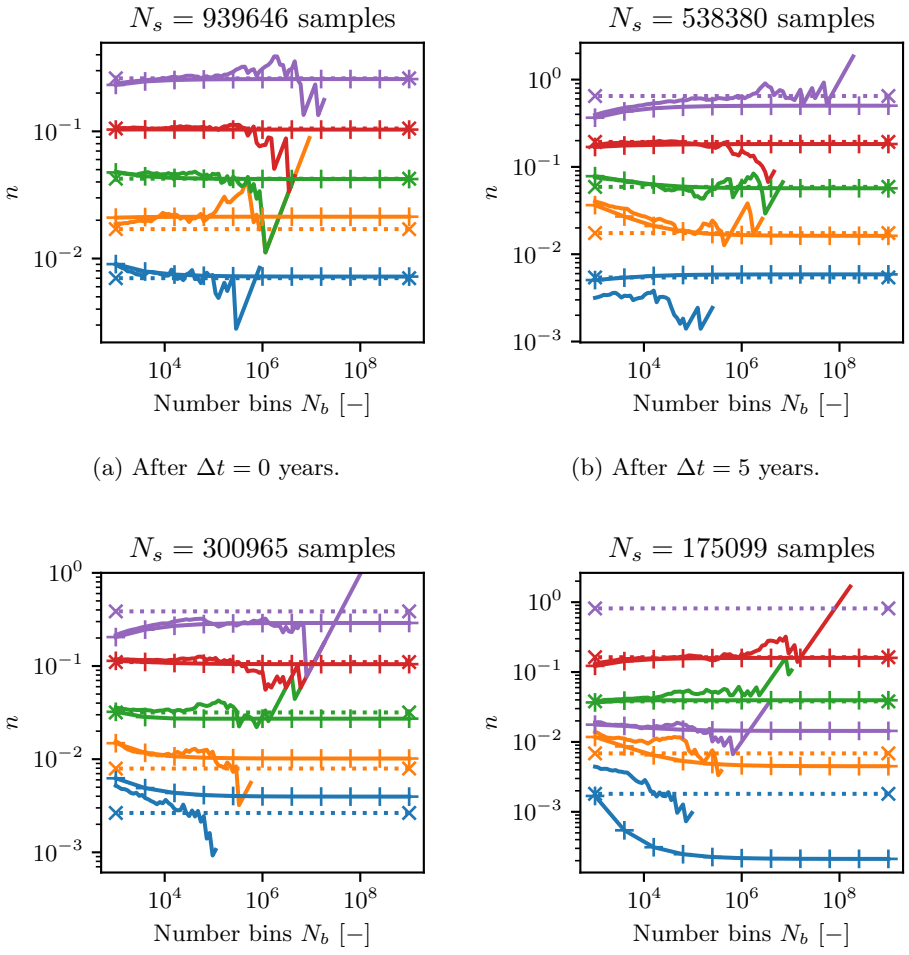
(b) Relative to the hypersurface sampled from the validation characteristics.

Figure 4.6: Snapshot of the validation characteristics at $\Delta t = 15$ years (blue), and its comparison to the surrogate model (orange).

snapshots, the true and fitted density values overlap nearly perfectly. For large enough bins even the Monte Carlo approach converges to the same values, thanks to the selection of the fit space being optimal for discretisation. However, estimates in low density regions remain inaccurate.

More interesting is Figure 4.7d where a single validation characteristic with the maximum density value corresponds to one of the outliers in Figure 4.6a. However, the density estimated through the Monte Carlo approach converges on the value given by the surrogate model. This observation indicates that the inaccuracies due to the hypersurface are highly localised and averaging over even a small region around such inaccuracies results in the correct value. In summary, except for very small density regions, the surrogate model performs well both against true values and against Monte Carlo. In addition, it outperforms the Monte Carlo approach in terms of computational time by two orders of magnitude. This shows the potential of the proposed fitting method to be an accurate technique for the evolution modelling of fragmentation clouds. However, it cannot be concluded that the errors will always remain bounded within similar bands for any fragmentation geometry or dynamical model. Thankfully, an easy and efficient way of assessing the accuracy of the surrogate model is available through the sampling of validation characteristics.

In this chapter, a way of interpolating the density from a cloud of scattered characteristics was presented. Several different density estimation techniques and methods were discussed. A GMM surrogate model was selected to be fitted through regression. It was shown to be a promising method for fragmentation cloud interpolation and facilitating interpretation of the distribution. Different optimisation schemas, from steepest descent to quasi Newton methods, were benchmarked, allowing to find a method to quickly fit the surrogate model. The concept of reference hypersurfaces was introduced to tackle the problem of non-smooth, bounded, and non-linear distributions. Finally, an efficient method of validating the resulting surrogate model was presented. The fragmentation propagation and interpolation method is implemented in a software suite, presented in the next chapter.



(a) After $\Delta t = 0$ years.

(b) After $\Delta t = 5$ years.

(c) After $\Delta t = 10$ years.

(d) After $\Delta t = 15$ years.

Figure 4.7: Comparison between exact density (dotted, crosses), the surrogate density estimate (solid, pluses) and the density estimate obtained through binning of the N_s samples (solid), for selected characteristics (coloured).

5 The Starling suite

Chapters 2 to 4 contain the theory behind the continuum approach used to simulate the evolution of a fragmentation cloud in orbit. Here, the implementation of the theory in the form of the *Starling* suite is presented. The suite, estimating evolving continua subject to non-linear dynamics, is being developed at the Politecnico di Milano and funded by the COMPASS European Research Council project and ESA.

The name *Starling* is inspired by the murmurations of starlings, a spectacle of nature where millions of birds fly together in perfect synchrony, seemingly as one ever-changing continuum (see Figure 5.1). Starlings have the capability to react and change their direction based on their immediate surroundings. Similarly, solving the continuity equation using the method of characteristics – the core of the *Starling* continuum propagation – can consider actuation based on the current density value of each characteristic curve.

The suite itself is designed as a generic partial differential equation solver. I.e. the propagation and fitting routines are not restricted to orbital dynamics nor space debris. Instead, the force model is incorporated through a fixed but generic interface allowing to plug in any dynamics. Any initial condition can be considered through yet another interface describing a density distribution that can be sampled for characteristics. As such, it is applicable to a wide range of problems involving the solution of the continuity equation. The implementation of the fragmentation model and the calculation of the spatial density and the collision probability are separate modules.

The structure of *Starling* is presented in Section 5.1. Section 5.2 introduces the interpreter class which enables the software suite to make sense of its data. Finally, Section 5.3 discusses the definition of the interfaces, the configuration and the data files. Note that parts of this chapter are adapted from (Frey et al., 2019a)[1].

[1] S. Frey, C. Colombo, and S. Lemmens. Application of density-based propagation to fragment clouds using the Starling suite, In *Proceedings of the 1st International Orbital Debris Conference*, 2019.

Figure 5.1: Murmurations of starling birds recalling an ever-changing continuum. Image credit: Fayez Nureldine/AFP/Getty Images.

5.1 Structure

Starling is written in the high-level and general-purpose PYTHON programming language (PYTHON 3.7, Python Software Foundation, 2020). It makes use of object-oriented programming and is highly modular. The execution of the programme is performed through a command-line interface and input files. The computationally expensive tasks are performed using the NUMPY and SCIPY scientific computing libraries, which utilise compiled C and FORTRAN libraries (Oliphant, 2006; Virtanen et al., 2020). High computational efficiency is achieved through vectorisation, using NUMPY structures and functions, and parallelisation, using the MULTIPROCESSING package.

An overview of the process chain of *Starling* is shown in Figure 5.2. It can be split into three independent parts: sampling of the characteristics, the propagation of the continuum and the model fitting routine. The separation of the processes decreases the complexity of the program. However, the loop could also be closed through backpropagation. If only a few characteristics remain, the operator might choose to sample the fitted model. The values of the sampled characteristics are updated through integration backwards in time. Such a procedure allows to create samples where most of the continuum remains, which is not known a-priori. The closed-loop process chain, albeit already implemented, is not yet at an operational maturity.

The initial distribution, e.g. the NASA SBM, the integrator, e.g. PlanODyn, and the surrogate model, e.g. a GMM, are incorporated in a modular fashion through interfaces (see Section 5.3). Hence, different initial conditions, dynamics or models can be explored. Each of the sub processes is controlled through JavaScript Object Notation (JSON) configuration files. Such files are easy and safe to read for humans and computers alike. The

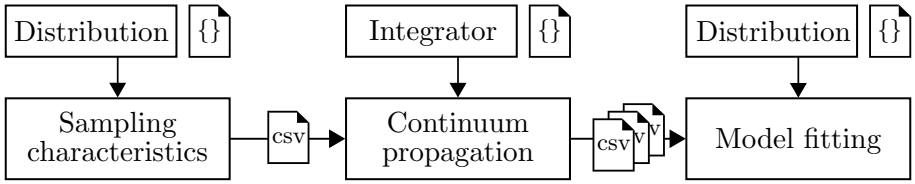


Figure 5.2: Schematic overview of the process chain implemented in *Starling*. The initial condition, integrator and model distribution can be interfaced with any program or definition.

processes communicate among each other using Comma-Separated Values (CSV) files, each of them representing snapshots of the characteristics – or the hypersurface samples – in time. Giving meaning to the snapshot files is enabled through the interpreter class, explained in more detail in the next section. The output of the fitting process is, again, a JSON file. Note that the initial condition is given in a single CSV, while multiple propagated snapshot files, at epochs defined by the user, are passed to the model fitting module that makes use of parallel computing.

The resulting fits can be used for post-processing. Graphical representations of the results, e.g. Figure 4.4, are enabled through the Matplotlib library (Hunter, 2007). Spatial density and number of impacts are computed and estimated through an additional package, which is an extension to *Starling*. The code documentation is generated automatically from the source code using Doxygen (van Heesch, 2018). Each method is commented detailing purpose, input arguments, return values and types and optional information.

5.2 Interpreter

Whenever the snapshot files are loaded, or saved, they pass through the interpreter class. Each of the three sub processes shown in Figure 5.2 have access to the interpreter. Next to simple loading and storing tasks, the interpreter oversees the transformation of the characteristics, i.e. the phase space and its density, into different phase spaces, e.g. for fitting or plotting. This includes calculating relative spaces through consideration of hypersurface samples. It also provides the possibility of adding constants that are required for the integrator but are not part of the phase space. Lastly, it enables to put a label on each dimension of the phase space, which is important for plotting purposes.

To make the interpreter impartial towards any problem definition, it is defined through a JSON configuration file, containing information that describe the data. Symbolic mathematics, enabled through SymPy (Meurer et al., 2017), eases the exploration of new phase spaces without the need of

analytically deriving the transformation of the density. An example for a two-dimensional interpreter configuration file is given in Figure 5.3.

Each interpreter configuration file consists of the following keywords: *data*, *states*, *weight*, *features* and *variables*. If one or multiple hypersurfaces are present, they are defined through an additional *surfaces* keyword. The *data* describes the full data set, which is required for the integrator routine, i.e. it is a mix of *states* relevant for the phase space derivation and default values. E.g. the inclination, required by the integrator, might not be needed to define the phase space, as it is assumed constant. The value describing the density is fixed via the *weight* keyword. The phase space is defined with the *features*, whose relationship with the *states* is given in the *variables* section.

Each variable is described through a *value* or a *default*. E.g. in Figure 5.3, the perigee radius, *rper*, and apogee radius, *rapo*, are defined through the semi-major axis, *sma*, and eccentricity, *ecc*, and vice-versa. Upon reading the data, the interpreter calculates the perigee and apogee radii and automatically transform the density according to Equation (2.10). Upon storing the snapshots, the inverse transformation is performed. Default values, such as the area-to-mass ratio, *amr*, are defined directly in the configuration file too. Note that the other Keplerian elements required by the propagator are not listed to increase readability of the figure. Each variable is further named through the *variable* and *unit* keywords, used for plotting purposes only. Note that the snapshot CSV files contain the *states*. Any feature space can be defined, the only restriction is that the feature space is of the same dimensionality as the *states*.

The *surfaces* are defined through a *dependent* feature and one or multiple *independent* features. Further keywords can be passed describing the interpolation method and its parameters. Subsequent processing of the data is performed as a recursive application of another layer of interpreter. The new *states* are the independent features and the new dependent features defined relative to the hypersurface. They and the new *weight* are transformed again into the new *features*, according to the rules set in the *variables*. In the given example, only one relative feature, *drapo*, is present dependent on the single feature *rper* only. The relative feature is subsequently transformed into the logarithmic *xiapo*. The recursive loop is currently restricted to one level of depth only. However, this restriction could be lifted easily.

5.3 Interfaces

The sampling of the characteristics from a distribution is trivial if the cumulative distribution function is available. If not, MCMC sampling methods, such as Metropolis-Hastings introduced in Section 3.2, can be employed. Hence, the minimum requirement for the sampling of the characteristics is that the distribution provides the probability of each sample. A direct

```
{
  "data": ["sma", "ecc", "amr", "dens"],
  "states": ["sma", "ecc"],
  "weight": "dens",
  "features": ["rper", "rapo"],
  "surfaces": {
    "interpolation": [{
      "dependent": "rapo",
      "independent": ["rper"],
      "method": "cubic"
    }],
    "states": ["rper", "drapo"],
    "weight": "dens",
    "features": ["rper", "xiapo"]
  },
  "variables": {
    "sma": {
      "value": "(rper+rapo)/2",
      "variable": "$a$",
      "unit": "m" },
    "ecc": {
      "value": "(rapo-rper)/(rper+rapo)",
      "variable": "$e$",
      "unit": "$-$" },
    "amr": {
      "default": 1.0,
      "variable": "$AMR$",
      "unit": "m$^2$/kg"},
    "dens": {
      "variable": "$n$",
      "unit": "m$^{-1} \\log_{10}$(m)"},
    "rper": {
      "value": "sma*(1-ecc)",
      "variable": "$r_p$",
      "unit": "m" },
    "rapo": {
      "value": "sma*(1+ecc)",
      "variable": "$r_a$",
      "unit": "m" },
    "drapo": {
      "value": "10**xiapo",
      "variable": "$\\Delta r_a$",
      "unit": "m"},
    "xiapo": {
      "value": "log(drapo, 10)",
      "variable": "$\\xi_a$",
      "unit": "$\\log_{10}$(m)"}
  }
}
```

Figure 5.3: Code snippet of interpreter configuration for a two-dimensional, relative features space.

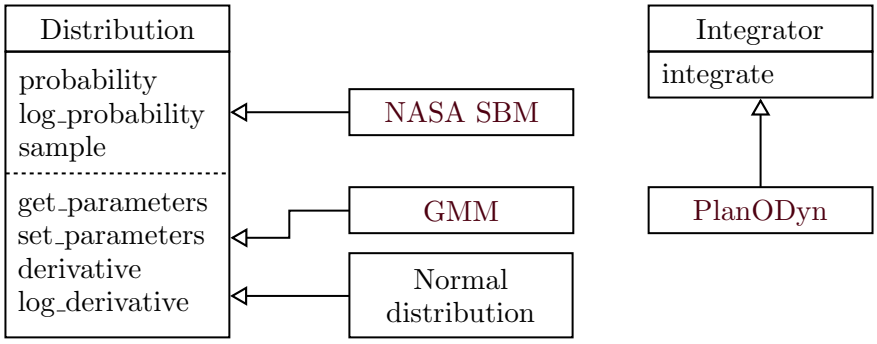


Figure 5.4: Interfaces of the distribution and integrator classes. For the distribution some methods are optional depending on the use case.

method to sample the distribution is preferred, however, as it is faster and more accurate than Metropolis-Hastings. Figure 5.4 shows the methods that are required for the interfacing of distributions and integrators. Any of the distributions is a potential initial continuum.

For the fitting routine, the distribution needs to provide handles to get and set the optimisation parameters. During each evaluation of the cost function, the derivative of the distribution is required to calculate the gradient and thus the search direction. The second derivative of the distribution with respect to its parameters is not listed here as only an estimate of the Hessian is maintained rather than evaluated exactly. Currently, only the normal distribution and the GMM provide the methods required for optimisation. Still, even the NASA SBM could be extended such as to re-fit it to newly available data.

The parameters for the optimisation routine are set via a configuration file. It contains information about the sample pruning, the hyperparameters and the optimisation method. Additionally, the covariances of the kernels of the GMM can be restricted to be diagonal. An example fitting configuration file is given in Figure 5.5a. The optimisation can be run in parallel for different snapshots given multiple CPUs.

The integrator class simply contains one function: integrate. Note that the integration does not only entail the evolution of the phase space, but the extended phase space, i.e. including the density. Additional parameters such as the force model settings are passed with keyword arguments during initialisation of the class. The settings are integrator specific and specified via JSON configuration files.

As of now, only one propagator – PlanODyn – is interfaced with *Starling*. PlanODyn is written in the MATLAB programming language and environment (MathWorks, Inc., 2020). The class instance utilises the MATLAB engine Application Programming Interface (API) for PYTHON. How to properly set up the API is explained in the code documentation. A snippet of a

```
{
    "worker": 40,
    "samples": {
        "prune":1e-5
    },
    "fit": {
        "nkernels": 8,
        "regulariser": 1e-2,
        "method": "L-BFGS-B",
        "jac": true,
        "hess": false,
        "diag": false
    }
}
```

(a) Optimisation configuration.

```
{
    "worker": 40,
    "intpar": {
        "jacobian": true,
        "t0": 0.0,
        "tf": 3.15576e9,
        "dt": 7889400.0,
        "solver": 113,
        "reltol": 1e-9,
        "abstol": 1e-12,
        ...
    },
    "physpar": { ... }
}
```

(b) PlanODyn configuration.

Figure 5.5: Examples of *Starling* configuration files.

```
# First line: time [s], total integral
# Following lines: a [m], e [-], n
0.00000000e+00,9.84577618e-01,0.00000000e+00
7.19657436e+06,5.35524603e-04,4.28024342e-03
7.18935665e+06,1.48892948e-03,1.74265701e-02
7.18851325e+06,1.66857467e-03,6.10812587e-03
7.19725275e+06,1.12085522e-03,1.77279593e-03
7.21592628e+06,2.21165399e-03,2.54483805e-02
...
```

Figure 5.6: Examples of a snapshot data file for a two-dimensional case.

PlanODyn configuration file is shown in Figure 5.5b, containing the integration and physical parameters. The number of workers tells PlanODyn how many CPUs can be used.

The snapshot files contain all the remaining characteristics at a given epoch. It consists of a short comment section, with information about the data extracted from the interpreter. The first line contains the epoch and an estimate of the total integral estimated with Equation (3.12). From the second line onward, it describes the phase space and density of the characteristics. An example of a snapshot file is given in Figure 5.6. The hypersurface sample file follows the same format with the last density row set to zeros. Following the same format allows to use the interpreting and plotting routines developed for the characteristics also on the hypersurface samples.

In this chapter, a software suite combining the techniques developed in this doctoral thesis was presented. Thanks to its generic implementation, it can be applied to many different problems requiring solving the continuity

equation in any dimension. The treatment of phase space densities through symbolic mathematics facilitates the exploration of many target spaces by simply rewriting a configuration file. Finally, given the theory and the software suite, it is time to make use of it. The next chapter shows applications of *Starling* to scenarios in various dimensions.

6 Applications

Together with the theory and the software suite, the cloud continuum evolution model can finally be applied to ever more complex orbit geometries and force models. Section 6.1 shows examples of fragment cloud evolutions in three-, four-, five- and six-dimensional phase spaces covering a wide range of eccentricities from LEO to the Geostationary Transfer Orbit (GTO). Once the evolution of the fragment cloud is known, products can be derived. Section 6.2 shows how the evolution model can be used to calculate the spatial density and the collision probability of the fragmentation cloud with other resident space objects in any orbital regime.

These examples are just a small subset of all the possible applications of the method. Other uses of the fragmentation cloud evolution method are parent identification, i.e. employing backpropagation of a cloud to pinpoint its origin (Heard, 1976), estimating the distribution and risk of re-entering objects (Trisolini and Colombo, 2019), rate current and future missions according to their hazard potential for space debris mitigation purposes (Letizia et al., 2019), or find optimal spacecraft attitudes to avoid incoming space debris. The method is not only applicable to fragmentation clouds, but can be applied to any distribution and uncertainty propagation problem, such as observation uncertainties in initial orbit determination, manoeuvring uncertainties, asteroid deflection, etc. Note that parts of this chapter are adapted from (Frey et al., 2020)[1].

6.1 Fragmentation cloud evolution

Five different scenarios are modelled based on three fragmentation cases, selected from NASA (2018). The first case, a Delta 2$^{\text{nd}}$ stage explosion, occurred in 1977 due to stored energy from residual propellant. It resided on an orbit spanning nearly the full LEO, from $h_p = 535$ km to $h_a = 2025$ km.

[1] S. Frey, C. Colombo, and S. Lemmens. Evolution of orbital fragmentation cloud via continuum and surrogate modelling. To be submitted to the *Journal of Guidance, Control, and Dynamics* in September, 2020.

Table 6.1: Orbit and location of fragmentation cases, extracted from (NASA, 2018). The international identifiers of the Delta stage, the Cosmos satellite and the Ariane stage are 1977-065B, 1993-036A and 1988-040B, respectively.

Object	a [km]	e [−]	i [deg]	Ω [deg]	ω [deg]	f [deg]
Delta 2$^{\text{nd}}$ stage	7659	0.0973	29.05	95.7	42.2	107.9
Cosmos-2251	7166	0.0016	74.04	19.5	98.7	358.6
Ariane 3$^{\text{rd}}$ stage	24369	0.7163	7.03	328.1	254.5	143.3

Off the 177 fragments catalogued, 59 remained in orbit in 2018. The second case is the Cosmos-2251 satellite collision with the Iridium 33 satellite taking place in February 2009 (NASA, 2009). Originating from a nearly circular orbit just below $h = 800$ km, 1076 of the 1668 catalogued fragments were still in orbit 9 years later. The third case is the explosion of an Ariane 3$^{\text{rd}}$ stage in GTO in 2002, orbiting from $h_p = 535$ km to $h_a = 35445$ km, coming within less than 150 km of the GEO domain. The breakup, caused again by on-board propellants, produced only 8 catalogued fragments, all of which were still on-orbit in 2018. The three corresponding parent orbits, chosen to span a wide range of eccentricities, are listed in Table 6.1. The breakups are modelled according to the NASA SBM as a payload collision in the case of the Cosmos-2251, and rocket body explosions for the other two cases.

The selection of the integration phase space, $\boldsymbol{x}$, and hence its dimension, D, depend on the initial orbit configuration, timeframe of interest and the considered force model as is discussed in Section 3.3. The first two scenarios, based on the Delta explosion and the Cosmos collision, model long-term evolutions subject to drag only. For all the scenarios, a non-rotating atmosphere is considered. Hence, the phase space is reduced to (a, e, A/m). The spreading of Ω and ω due to J_2 is implicitly considered via randomisation. It has no effect on the evolution of the density, n, as it induces a translation rather than resizing of an infinitesimal phase space volume. Thus, propagation can be performed without considering J_2.

The third scenario is an extension of the second scenario from long- to mid-term evolution, i.e. the dependence on Ω is considered by propagation of the characteristics subject to drag and, explicitly, J_2. The dependence on ω is not considered, as it is not well defined given the nearly circular parent orbit. Randomisation of ω is achieved at the point of fragmentation, or shortly thereafter.

The fourth and fifth scenarios are based on the Ariane explosion. As the cloud originates from a parent object in GTO, randomisation of Ω and ω cannot be assumed to occur in a few months or years only. The force model of the fourth scenario is considering drag and J_2. Hence, the inclination remains constant and the phase space is (a, e, Ω, ω, A/m). Given the orbit,

Table 6.2: Considered scenarios. The phase space is dependent on the initial orbit, and the time-frame and force model considered.

Case	Timeframe	Force model	Phase space
Delta	long-term	drag	(a, e, A/m)
Cosmos	long-term	drag	(a, e, A/m)
Cosmos	mid-term	drag, J_2	(a, e, Ω, A/m)
Ariane	mid- to long-term	drag, J_2	(a, e, Ω, ω, A/m)
Ariane	mid- to long-term	drag, J_2, 3BP, SRP	(a, e, i, Ω, ω, A/m)

forces induced by 3BP of the Moon and Sun, and SRP cannot be ignored. The fifth scenario is thus an extension of the fourth scenario, considering the full force model and the variability of i. An overview of the five scenarios is given in Table 6.2.

The scenarios are discussed and benchmarked in terms of accuracy and computational efficiency. One of the great advantages of using the method of characteristics is that it allows to validate the quality of the surrogate model. By sampling and propagating an additional set of verification characteristics, the density estimate obtained from the surrogate model can be compared against the true density using Equation (4.17).

The results of each of the scenarios are presented following the same structure. First, the scenario is repeated briefly. Second, the characteristics, sampled from the NASA SBM, are shown as a function of integration phase space, replacing A/m with $\chi = \log_{10}(A/m)$. Both, the initial and final distribution, i.e. propagated to a certain epoch in the future, are presented to highlight the non-linearities of the evolution. Third, the fit space, including the definition of the reference hypersurface, as well as the hyperparameters are introduced. Fourth, the characteristics used for the fitting and $C_0 = 1000$ validation characteristics are compared to the approximated density and presented in the fit space. Fifth, the evolution of the accuracy over time is shown by comparing the approximated density values with $C_0 = 10^4$ validation characteristics. Lastly, information about the computational efficiency are given. The simulations are performed on a parallelised setup, using 4 Intel® Xeon® E5-4620 v4 CPUs, with a total of 40 cores.

Note that for all the examples in this section, the density of a single fragment, i.e. its probability, is depicted in the figures, rather than the density of the continuum. To compute the fragment continuum, simply multiplication with the total number of initial fragments needs to be performed.

6.1.1 Delta stage explosion: LEO considering drag

This scenario describes the evolution of a fragmentation cloud stemming from the Delta stage, i.e. on an initially eccentric orbit in LEO, and subject to atmospheric drag only. The long-term evolution is modelled, hence a three dimensional fit space, derived from a, e and A/m is considered. The fragmentation takes place far away from the line of apsides. Hence, the initial distribution in the subset of the Keplerian elements, shown in Figure 6.1a, does not have the boundedness typical for clouds originating from circular parent orbits. As for the example in Section 4.4, the diagonal sub-figures show a one-dimensional representation of n as a function of each phase space variable. The off-diagonal sub-figures show a two-dimensional representation of the scattered characteristics. Their colour variations correspond to n, from low (dark purple) to high (light green). Of the initially $C_0 = 1000$ sampled characteristics, 33 either re-enter directly, i.e. their perigee altitude is lower than the radius of the Earth, or the Δv raises their orbital velocities to above the escape velocity, v_∞, defined as (Vallado, 2013, Chapter 1.3)

$$v_\infty = \sqrt{\frac{2\mu}{r}} \tag{6.1}$$

and the fragments described by the characteristics directly escape from the gravitational pull of the Earth.

Figure 6.1b shows the same characteristics propagated for $\Delta t = 25$ years. Only $C = 141$ characteristics remain, the rest – mostly the high A/m fragments – have re-entered in the meantime due to atmospheric drag. In the $a - \chi$ or $e - \chi$ sub figures, a tilting away from high χ fragments towards smaller a and e can be observed. The $a-e$ sub figure reveals that the dynamics force the characteristics from an initially distributed cloud into a narrow band. Even though fitting of the GMM in the Keplerian space would result in a good fit for the initial distribution, it would fail to accurately capture the propagated distribution.

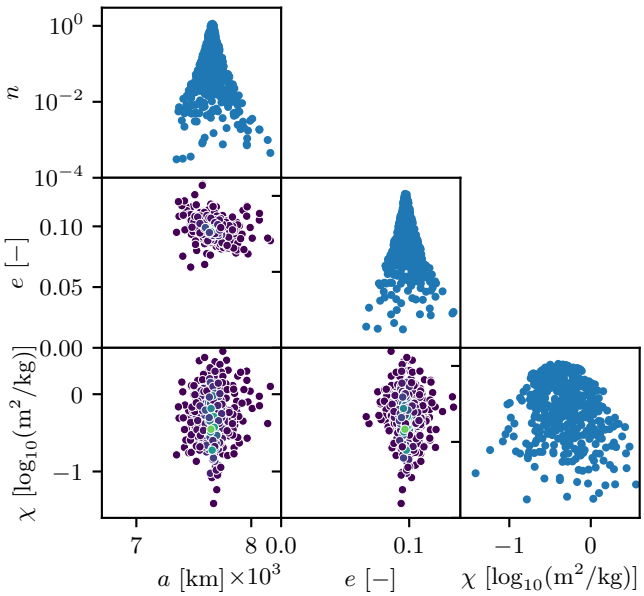
Instead, the fit is performed in a more suitable target space

$$\zeta_p = \log_{10}\left(a(1-e) - (R_E + \Delta h_{re})\right) \tag{6.2a}$$

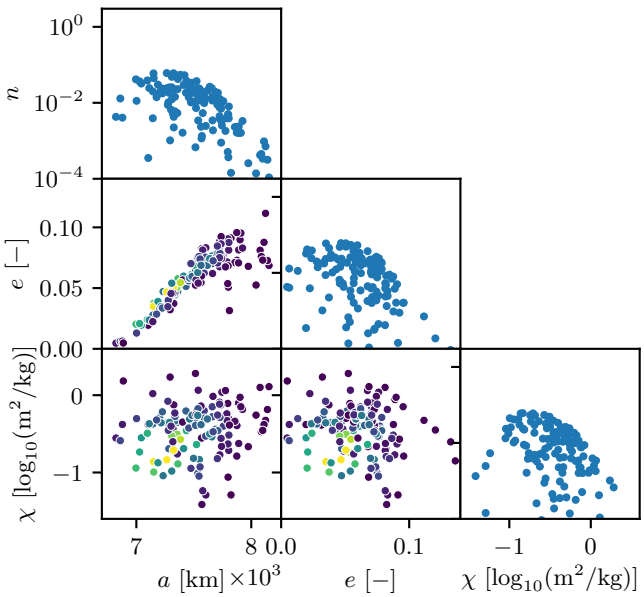
$$\Delta e = e - \hat{e}\left(\zeta_p, \chi\right) \tag{6.2b}$$

$$\chi = \log_{10}(A/m) \tag{6.2c}$$

where $\Delta h_{re} = 100$ km coincides with the stop condition of the propagations, i.e. the fragments are considered re-entered. Note that this is not the only possible target space definition, but it was found to work well for the problem at hand, as the definition of ζ_p helps in keeping the distribution compact. The reference hypersurface, $\hat{e}$, is found with a cubic spline in the variables ζ_p and χ. The initial hypersurface samples are obtained by mapping the initial characteristics into the parent eccentricity, i.e. $\hat{e}_0 = e_0$.



(a) Initial distribution, with $C = 967$ characteristics, at $\Delta t = 0$ years.



(b) Final distribution, with $C = 141$ remaining, at $\Delta t = 25$ years.

Figure 6.1: Characteristics in Keplerian elements for the Delta explosion scenario.

Instead of using the hypersurface to remove a boundary outside of the cloud of initial characteristics, as discussed in Section 4.4, the hypersurface defined here is defined within the cloud. As such, it reduces the non-linearities accumulating over time, facilitating the fitting of the surrogate model. For the fitting of the model, $K = 8$ kernels, corresponding to 80 fit parameters, are used with a regulariser of $R = 10^{-2}$. The fit is performed at quarterly epochs, corresponding to 101 snapshots spanning the 25 years.

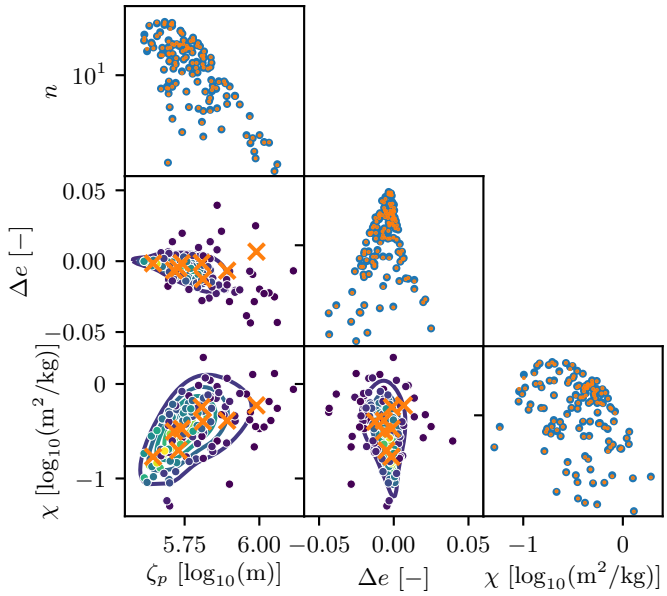
The characteristics used for the fit (in blue) and the corresponding density estimates from the surrogate model (in orange) at $\Delta t = 25$ years are shown in the target space in Figure 6.2a. In the off-diagonal sub-figures, the density fit is projected into the two-dimensional spaces in terms of the Gaussian kernel means (orange crosses) and estimated density (contour lines). Two observations can be made. First, the non-linearities in the target space are not present, as the hypersurface, itself propagated, absorbs the non-linearities of the dynamics. Secondly, a good agreement between the estimate and the characteristics is found.

That the surrogate model well approximates the true density distribution is also evident in the good agreement between the validation characteristics and the surrogate model presented in Figure 6.2b. Only towards the low ζ_p domain, i.e. for characteristics close to re-entry, small inaccuracies are creeping in, a consequence from only a few characteristics being present in that high atmospheric density region.

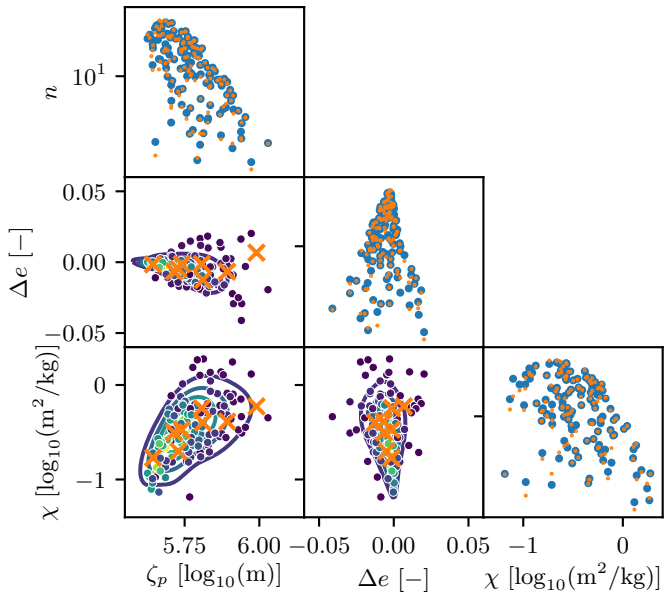
Finally, the fit quality over all the snapshots is presented in the form of quantiles of the relative errors in Figure 6.3. The errors remain bounded between $\pm 55\%$ for the 5% and 95% quantiles, respectively. Half of the validation characteristics, from $q_{25\%}$ to $q_{75\%}$ remain within -14% to $+7\%$ for all the snapshots. Hence, the surrogate model provides an accurate estimate of the underlying true distribution, even if only few characteristics are present. Increasing the number of fitting characteristics and especially hypersurface samples would further increase the accuracy of the fit. The propagation of the $C_0 = 1000$ characteristics and equal number of hypersurface samples is performed in less than 1 min. The average fit time for the 101 snapshots is 10.8 s, requiring about 30 s on the parallelised setup.

6.1.2 Cosmos-2251 collision: LEO considering drag

The second scenario models the long-term evolution of the Cosmos-2251 fragmentation cloud. As for the previous scenario, only perturbations due to drag are considered. Hence, the integration phase space is identical to the previous one with (a, e, A/m). The cloud is ejected from a parent object on a nearly circular orbit. Thus, the initial distribution in the Keplerian elements, given in Figure 6.4a, exhibits the typical v-shape in the $a - e$ domain. Of the initially drawn $C_0 = 1000$, more than 300 characteristics either re-enter within one orbit or escape, resulting in a cut-off of the v-shape.



(a) A very accurate fit is achieved with the training characteristics.



(b) Also, the validation characteristics are in good agreement with the fit.

Figure 6.2: Remaining characteristics in the fit space after $\Delta t = 25$ years for the Delta explosion scenario.

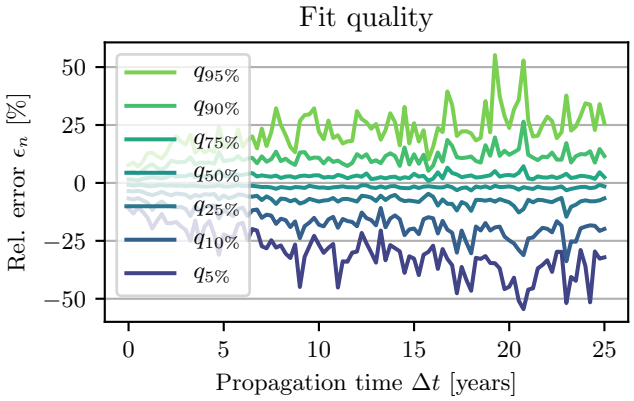


Figure 6.3: Fit quality for the Delta explosion scenario over the $\Delta t = 25$ years: 80% of the validation samples are accurately estimated within $\pm 25\%$ for most of the snapshots.

This is a consequence of the high energy involved in the catastrophic collision, spreading the fragments into high eccentricity orbits. After $\Delta t = 25$ years, only $C = 188$ characteristics remain, the rest have re-entered. Especially fragments with $a < a_0$ have re-entered as is visible in Figure 6.4b.

The fit is performed in the target space

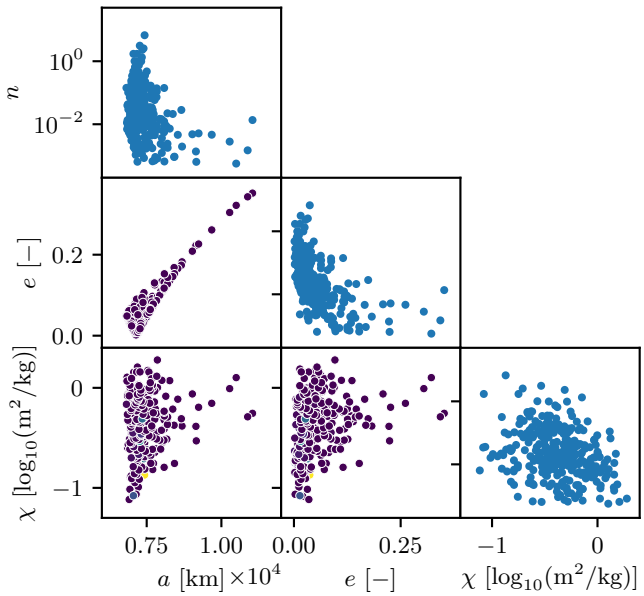
$$\xi_p = \log_{10}(r_0 - a(1-e)) \tag{6.3a}$$

$$\xi_e = \log_{10}(e - \hat{e}(\xi_p, \chi)) \tag{6.3b}$$

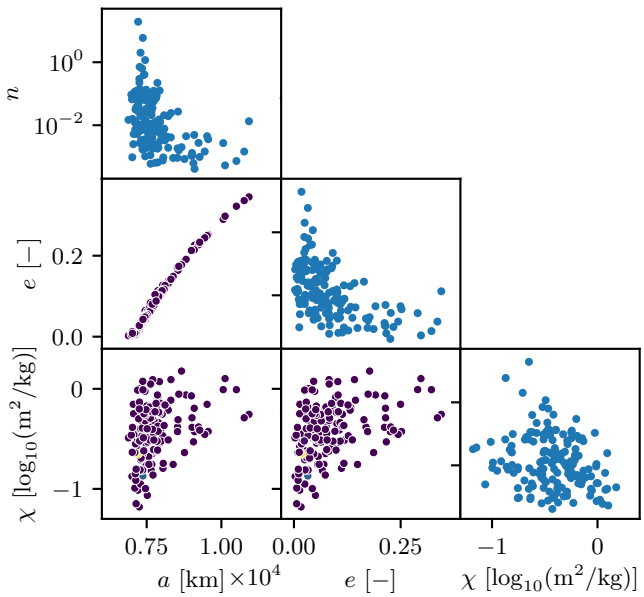
$$\chi = \log_{10}(A/m) \tag{6.3c}$$

where $r_0 = a_0(1-e_0)$ is the fragmentation radius. The reference hypersurface, $\hat{e}$, is acting as a boundary and obtained from the samples using a cubic spline in ξ_p and χ. The initial hypersurface samples are found by mapping the characteristics into $r_p = r_0$ and $r_a = r_0$, resulting in twice as many samples as characteristics. The given fit space is well suited to fit fragmentations from circular or near-circular orbits, as it removes the boundedness of the space. As before, $K = 8$ kernels and a regulariser of $R = 10^{-2}$ are used to fit 101 quarterly snapshots.

The fit characteristics used for the optimisation of the surrogate model are presented in the fit space at $\Delta t = 25$ years in Figure 6.5a. Visually, the surrogate model seems to capture the underlying distribution well. The model performs well too when compared to the verification characteristics in Figure 6.5b. Using twice as many characteristics as optimisation parameters seems to be sufficient to prevent overfitting in the given phase space. Quantitatively, the fit is accurate too, as is evident in Figure 6.6 showing the fit quality. Half of all the density estimates between $q_{25\%}$ and $q_{75\%}$ always remain within -6% and $+2\%$. Even the 90% between $q_{5\%}$ and $q_{95\%}$

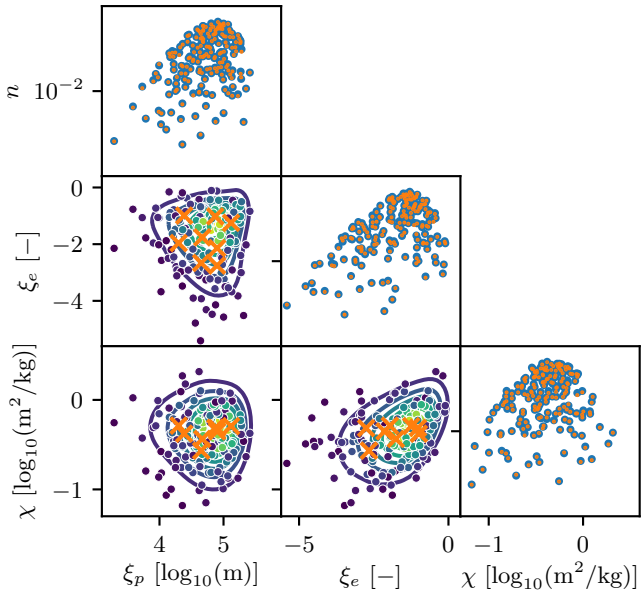


(a) Initial distribution, with $C = 689$ characteristics, at $\Delta t = 0$ years.

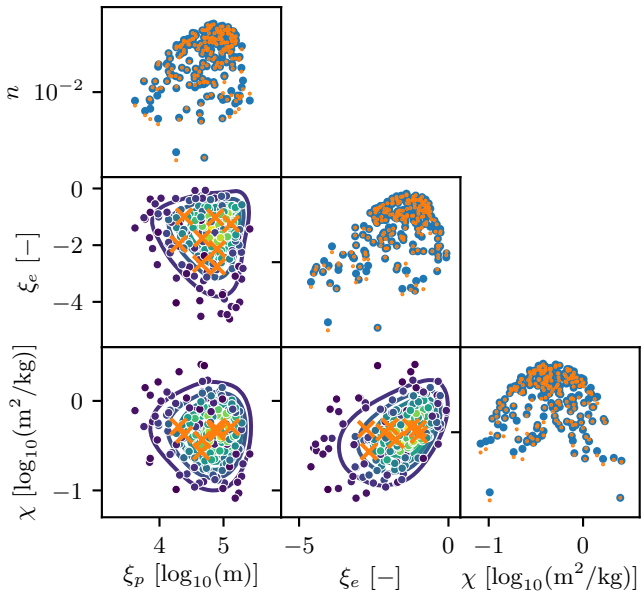


(b) Final distribution, with $C = 188$ remaining, at $\Delta t = 25$ years.

Figure 6.4: Characteristics in Keplerian elements for the long-term Cosmos-2251 collision scenario.



(a) Again, a very accurate fit is achieved with the training characteristics.



(b) As before, the validation characteristics are in good agreement with the fit.

Figure 6.5: Remaining characteristics in the fit space after $\Delta t = 25$ years for the long-term Cosmos-2251 collision scenario.

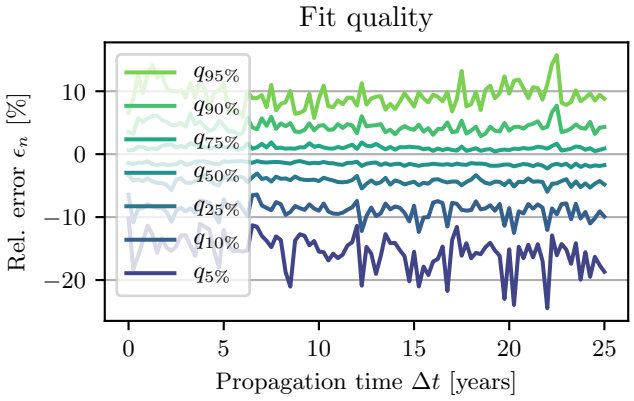


Figure 6.6: Fit quality for the long-term Cosmos-2251 collision scenario over the $\Delta t = 25$ years: again, 90% of the validation samples are accurately estimated within −24% and +16% for all the snapshots.

remain within error bounds of −24% and +16%. Hence, the surrogate model performs excellently.

The propagation of the characteristics and the hypersurface samples is performed within 70 s. On average, each model is fitted within 2.3 s. On the parallelised setup, the 101 surrogate models are fitted within less than 10 s. The fast convergence rate and the high accuracy show that the fit space is ideal to model such a distribution.

Generally, the continuum approach performs well for scenarios with drag only and thus in fit spaces derived from a, e and A/m. This is true for any parent orbit. As a rule of thumb, if fragmentation occurs close to the apsides, a logarithmic relative space is required for the GMM to be accurately fitted. In the case of a fragmentation from a (near-)circular parent orbit, two logarithmic target spaces are required, as shown here. In case the fragmentation occurs far away from the apsides, as shown in the previous example, a target space defined relative to a hypersurface absorbing the non-linearities is sufficient. Further simplifying the application of the method is the fact that the hyperparameters, K and R, are robust and can be kept constant for different scenarios, number of fit characteristics and geometries.

6.1.3 Cosmos-2251 collision: LEO considering drag and J_2

The third scenario extends the previous example by one dimension to study the mid-term evolution of the collision cloud. Next to atmospheric drag, the spreading of Ω due to J_2 is modelled in (a, e, Ω, A/m). Note that the distribution is not modelled in ω because the parent orbit is nearly circular. Hence, ω is not well defined and randomised almost instantly after the

fragmentation event. The initial characteristics are shown in Figure 6.7a. Of the drawn $C_0 = 1500$ fragments, more than 25% re-enter or escape directly, leaving only 1103 characteristics. The spreading in Ω is initially about $\Omega_0 \pm 5$ deg.

The characteristics are propagated for only $\Delta t = 1$ year and shown in Figure 6.7b. Only $C = 841$ characteristics remain. The most evident change comes from the precession of Ω due to the oblate Earth. The Ω of the fastest fragments revolve 2 times around Earth, the one of the slowest less than 0.5 times. Simultaneously, the evolution of Ω is highly correlated in a and e. Care needs to be taken to not prematurely declare randomisation in Ω. Despite having fragments overtaking other fragments in Ω, randomisation is not yet achieved as the overtaking takes place in different altitude regimes.

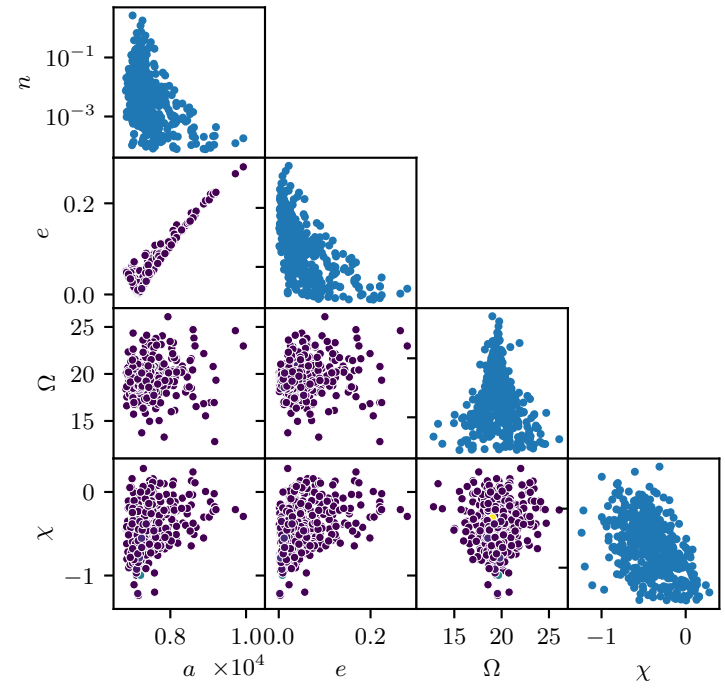
The fit is performed in the same target space as before but extended by one dimension

$$\xi_p = \log_{10}\left(r_0 - a(1-e)\right) \tag{6.4a}$$
$$\xi_e = \log_{10}\left(e - \hat{e}\left(\xi_p, \chi\right)\right) \tag{6.4b}$$
$$\Delta\Omega = \Omega - \hat{\Omega}\left(\xi_p, e, \chi\right) \tag{6.4c}$$
$$\chi = \log_{10}(A/m) \tag{6.4d}$$

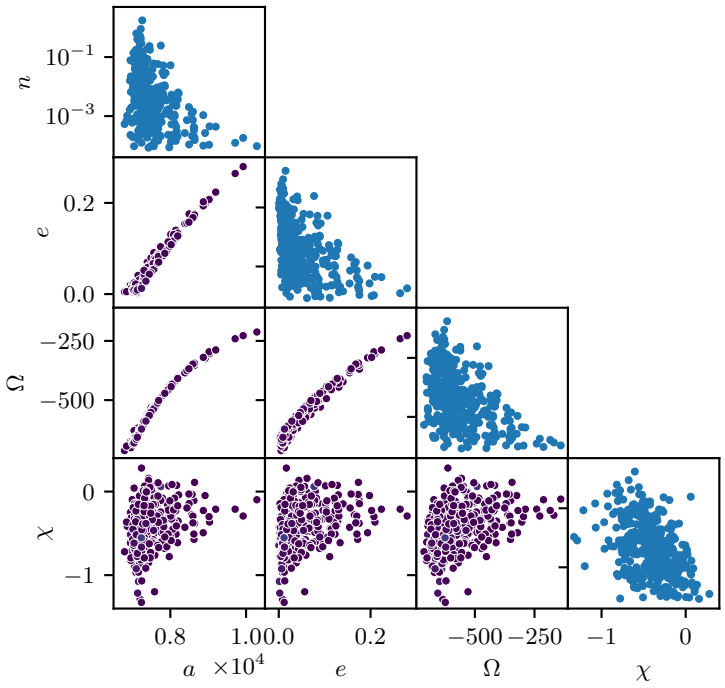
where r_0 is again the fragmentation radius. For this scenario, two hypersurfaces are required. The first is equivalently defined as in the previous example. Note that the evolution of e is independent of Ω, hence $\hat{e}$ does not depend on it either. The second hypersurface, $\hat{\Omega}$, is obtained through a quadratic RBF interpolation in ξ_p, e and χ. The initial hypersurface samples are found by mapping the characteristics into the initial $\Omega = \Omega_0$. Again, $K = 8$ kernels, or 120 parameters, are fitted at each epoch, using a regulariser of $R = 10^{-2}$. The fit is performed at 101 snapshots, equally spaced over the year.

The characteristics used for the fit, and its surrogate model are shown in the target space at $\Delta t = 1$ year in Figure 6.8a. As in the previous example, the hypersurfaces manage to absorb the non-linearities introduced by the dynamics. A decent fit is achieved using the surrogate model with $K = 8$ kernels only. Despite the spreading of Ω in Figure 6.7b, if considered as a function of a, e and A/m, the $\Delta\Omega$ remains constant. However, the hypersurface elongates over time, i.e. even small changes in the initial condition eventually result in large changes in Ω.

A comparison using the validation characteristics in Figure 6.8b reveals an issue with the definition of the hypersurface in $\hat{\Omega}$. The hypersurface, spread over 1.5 revolutions, or 540 deg is required to map three dependent variables into a narrow band of $\Delta\Omega \pm 25$ deg. It does so insufficiently for parts of the validation characteristics. This is also evident in Figure 6.9 comparing the estimates from the surrogate model against the validation

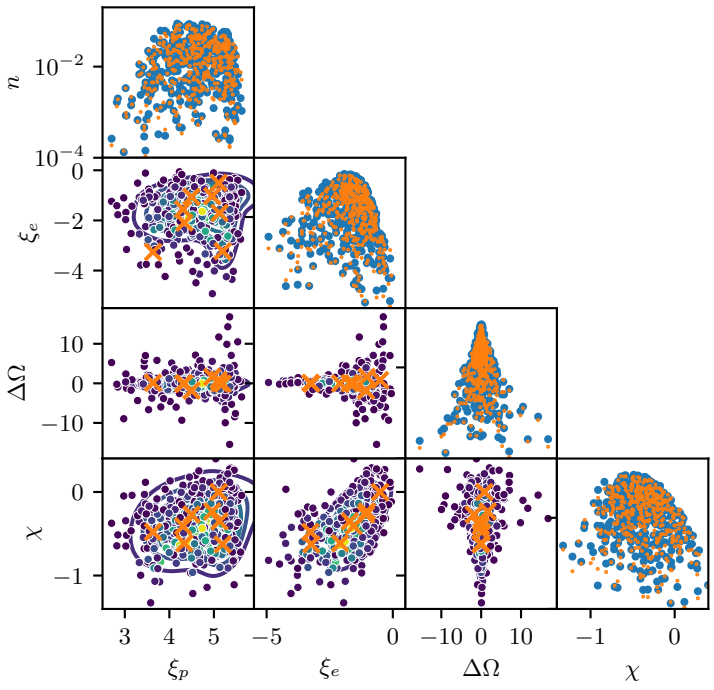


(a) Initial distribution, with $C = 1103$ characteristics, at $\Delta t = 0$ years.

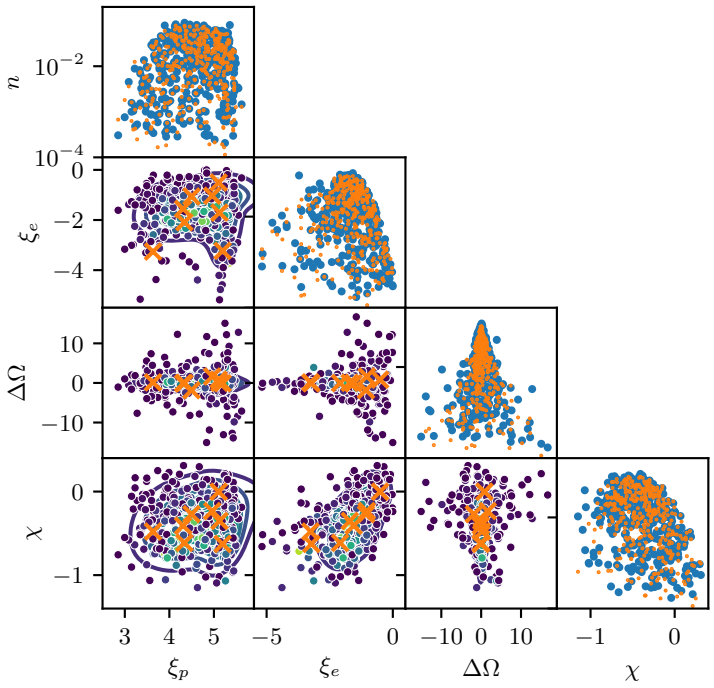


(b) Final distribution, with $C = 841$ remaining, at $\Delta t = 1$ year.

Figure 6.7: Characteristics in Keplerian elements for the mid-term Cosmos-2251 scenario. Note the different limits in Ω. Units: a [km], e [−], Ω [deg], χ [$\log_{10}(\mathrm{m}^2/\mathrm{kg})$].



(a) A decent fit is achieved with the training characteristics.



(b) However, the distance to the hypersurface in Ω is inaccurate.

Figure 6.8: Remaining characteristics in the fit space after $\Delta t = 1$ year for the mid-term Cosmos-2251 collision scenario. Units: ξ_p [$\log_{10}$(m)], ξ_e [−], $\Delta\Omega$ [deg], χ [$\log_{10}(\mathrm{m}^2/\mathrm{kg})$].

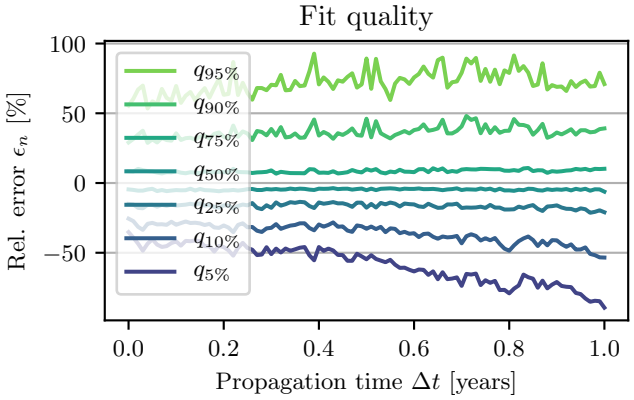
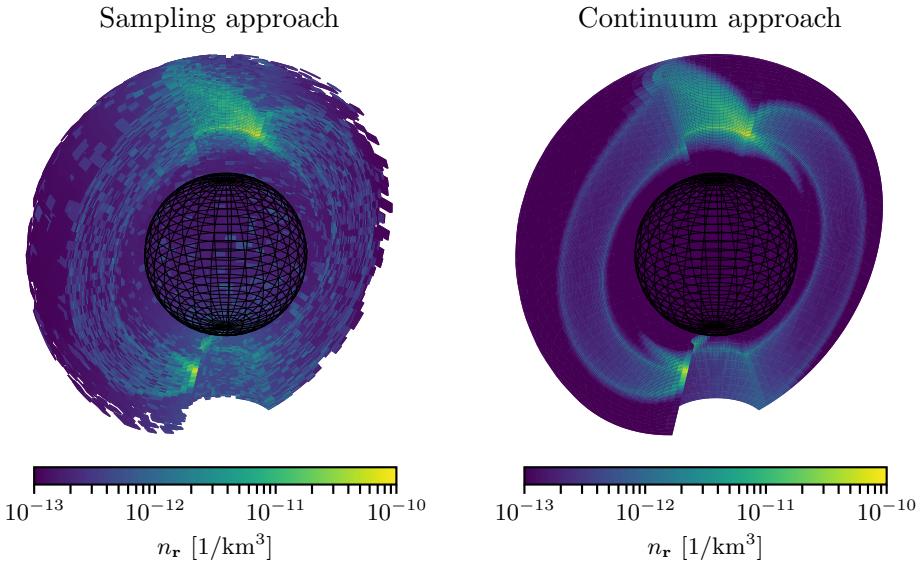


Figure 6.9: Fit quality for the mid-term Cosmos-2251 collision scenario over the $\Delta t = 1$ year: the fast spreading of Ω poses difficulties to the hypersurface interpolation.



(a) Using Monte Carlo sampling, only the first layer is visible.

(b) Using the continuum approach, distinct layers in r corresponding to the evolution in Ω can be observed.

Figure 6.10: Spatial distribution, not drawn to scale, of the probability of a single fragment of the Cosmos-2251 collision cloud after $\Delta t = 1$ year.

characteristics. The 50% of the samples within $q_{25\%}$ and $q_{75\%}$ remain within -22% and $+12\%$ relative errors. Instead, the lower and upper 5% come close to $\pm 100\%$ relative error. This seems worse than it is. Given the exponential decay of the underlying distribution, misplacing a fragment in $\Delta\Omega$ by just 1 deg results in a median absolute error of 70% over all the fit characteristic. So rather than giving wrong density values, it slightly misplaces them in Ω.

The shift is so small, it can hardly be discerned visibly. Figure 6.10 shows the spatial density of the probability of a single object of the fragment cloud at $\Delta t = 1$ year for the continuum method and one obtained from sampling with $N_s = 10^6$ instances. The density, covering the altitude domain from $R_E + 100$ km to $R_E + 2000$ km, is not drawn to scale and split in half to allow peeking into the domain. Note that the density derived from the individual samples is subject to the same assumptions as the continuum approach, i.e. randomisation in ω and M as well as a fixed $i = i_0$. Distinct layers can be discerned from the continuum approach corresponding to the cloud crossing the revealed bins at $\Delta t = 1$ year. The density obtained through samples shows some of the same features, but they are harder to discern. For statistical evaluations of the collision probability, the errors in the displacement are not expected to result in erroneous probabilities. However, this depends on the target orbit. E.g. analysis of target orbits synchronised in the motion of Ω with the bulk of the cloud might result in under- or overestimation of the collision probability.

The propagation of the characteristics and the hypersurface samples is performed within less than 2 mins. On average, each snapshot is fitted within 5.3 s. The total fitting time thus requires less than 15 s on the parallelised setup.

6.1.4 Ariane explosion: GTO considering drag and J_2

This scenario shows the evolution of a fragmentation cloud from the Ariane stage explosion in GTO. Contrary to objects in LEO, randomisation of Ω and ω is slow and cannot be assumed to be completed after a few years only. Here, only the simplified force model with atmospheric drag and J_2 is employed, thus, variations in the inclination can still be ignored and the integration phase space is (a, e, Ω, ω, A/m).

If integrated over the characteristic length, the NASA SBM is given as a function of $\Delta\boldsymbol{v} \in \mathbb{R}^3$ and $A/m \in \mathbb{R}$. The studied phase space, however, is five-dimensional. The transformed initial condition is thus defined on a four-dimensional hypersurface in the five-dimensional phase space. This corresponds to a Dirac delta function, i.e. the distribution is zero anywhere in the domain except on the hypersurface. It is not feasible to fit a GMM to the Dirac delta function. Ideally, a fit space defined on the hypersurface was used. A potential phase space suitable for this task is proposed as future work in Section 7.2.

Instead, the initial distribution from the breakup model is extended into a proper five-dimensional distribution. An additional uncertainty is introduced in the position of fragmentation according to Equation (2.67). The normally distributed uncertainty, of zero mean and standard deviation of $\sigma_r = 50$ km, is introduced in the orbital plane in the direction perpendicular to the velocity vector, i.e. along the $\hat{t}$-axis in the satellite frame, $\mathcal{T}$. A smaller displacement could be used, however, at the expense of an increased number of required kernels during the fitting.

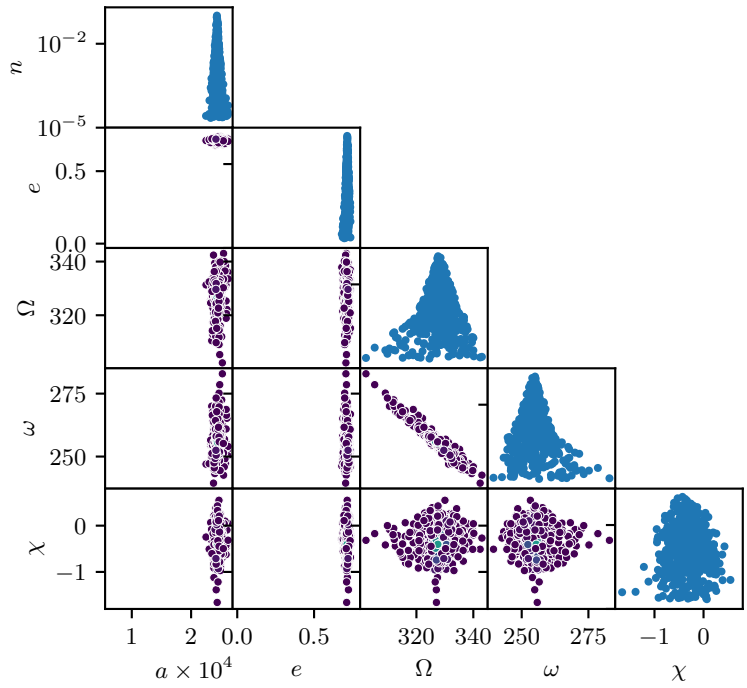
The initial distribution and the distribution after $\Delta t = 100$ years are shown in Keplerian elements in Figure 6.11. Of the initially sampled $C_0 = 2000$ characteristics, 167 re-enter directly or escape. After $\Delta t = 100$ years, only 1176 characteristics remain. Initially, the distribution could be fitted directly in the Keplerian space. However, after a few years, non-linearities start to dominate the distribution, thus making a fit difficult. Additionally, a linear dependence in $\Omega - \omega$ can be observed. The fastest and slowest fragments travel 140 and 23 times around Earth, respectively, in Ω, and 53 and 270 times, respectively, in ω.

The fit is performed in the target space

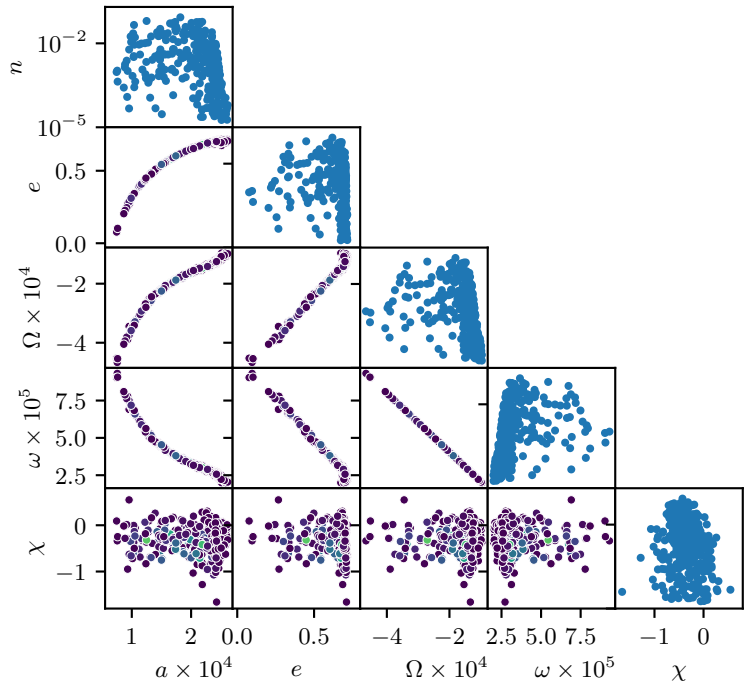
$$\zeta_p = \log_{10}\left(a(1-e) - (R_E + \Delta h)\right) \tag{6.5a}$$
$$\Delta e = e - \hat{e}\left(\zeta_p, \chi\right) \tag{6.5b}$$
$$\Delta\Omega = \Omega - \hat{\Omega}\left(\zeta_p, e, \chi\right) \tag{6.5c}$$
$$\Delta\omega = \omega - \hat{\omega}\left(\zeta_p, e, \chi\right) \tag{6.5d}$$
$$\chi = \log_{10}(A/m) \tag{6.5e}$$

The variable ζ_p is, as for the Delta scenario, logarithmic towards the reentry altitude to allow the many scattered fragments to be bundled together. Three reference hypersurfaces, $\hat{e}$, $\hat{\Omega}$ and $\hat{\omega}$, are defined. The hypersurface samples in $\hat{e}$ are found by mapping the characteristics into the apogee radius of the parent orbit, i.e. the sample perigee remains the same as the characteristic perigee, and the sample apogee corresponds to the parent orbit apogee. The samples for $\hat{\Omega}$ are defined analogously to the third scenario. No additional samples are required for $\hat{\omega}$ as its hypersurface is built from the same samples as for $\hat{\Omega}$. The initial hypersurface samples are set to $\omega = \omega_0$. The fits are performed using $K = 16$ kernels, corresponding to 336 parameters, and a regulariser of $R = 10^{-2}$. Yearly snapshots are fitted, resulting in 101 surrogate models.

The density estimates from the surrogate models and the characteristics used for the fitting at $\Delta t = 100$ years are shown in the target space in Figure 6.12a. As for the previous two examples, the non-linearities are largely absorbed by the hypersurfaces. Hence, even a relatively small number of kernels manages to achieve a decent fit. However, the hypersurface defined in the angles suffers from the same inaccuracy as for the previous example,

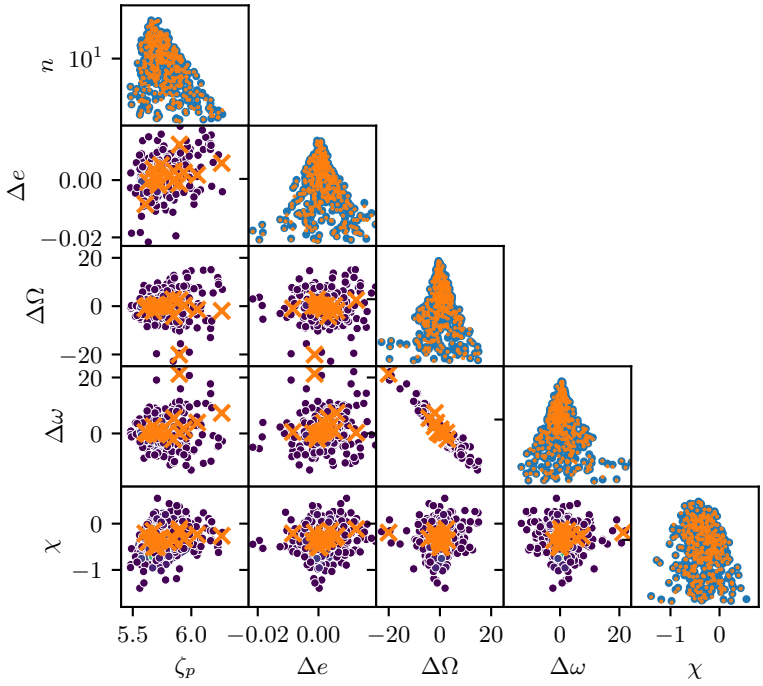


(a) Initial distribution, with $C = 1833$ characteristics, at $\Delta t = 0$ years.

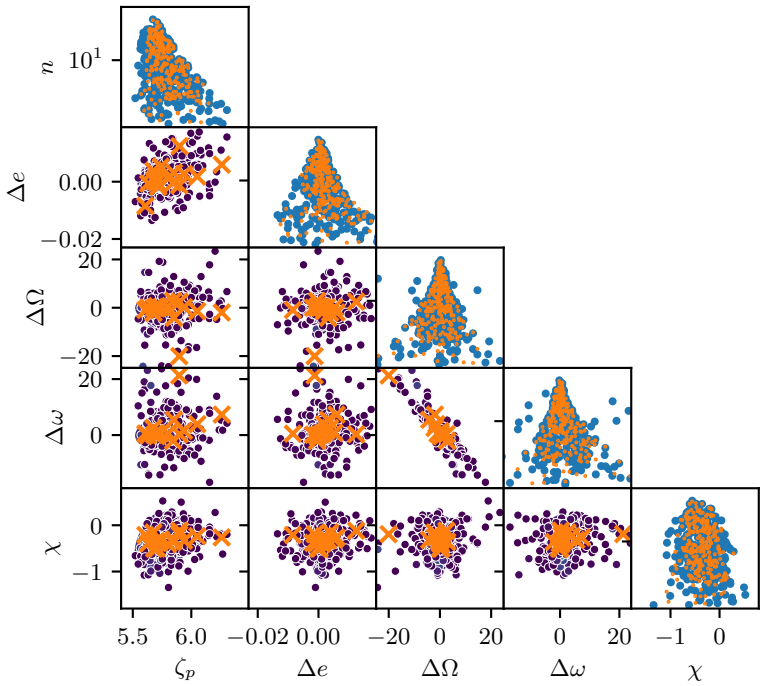


(b) Final distribution, with $C = 1176$ remaining, at $\Delta t = 100$ years.

Figure 6.11: Characteristics in Keplerian elements for the simplified Ariane explosion scenario. Note the different limits in Ω and ω. Units: a [km], e [−], Ω [deg], ω [deg], χ [$\log_{10}(\mathrm{m}^2/\mathrm{kg})$].



(a) A decent fit is achieved with the training characteristics.



(b) However, the hypersurfaces in Ω and ω are inaccurate.

Figure 6.12: Remaining characteristics in the fit space after $\Delta t = 100$ years for the simplified Ariane explosion scenario. Units: ζ_p [$\log_{10}$(m)], Δe [−], $\Delta\Omega$ [deg], $\Delta\omega$ [deg], χ [$\log_{10}(\mathrm{m}^2/\mathrm{kg})$].

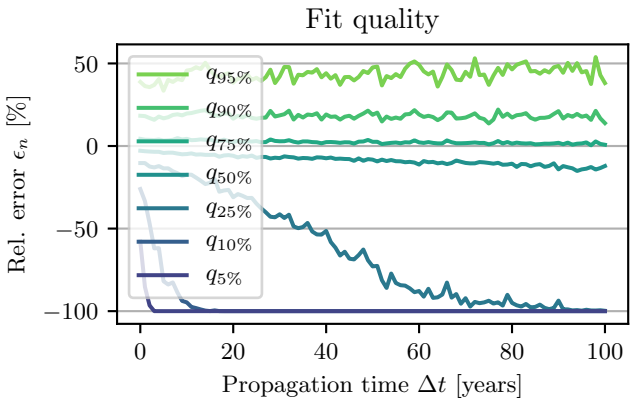


Figure 6.13: Fit quality for the simplified Ariane explosion scenario over the $\Delta t = 100$ years: the spread in Ω and ω pose difficulties to the hypersurface interpolation.

evident in Figure 6.12b. Away from the zero $\Delta\Omega$ or $\Delta\omega$, erroneous density values are depicted. Again, the hypersurface requires to map a Ω spread over 42000 deg and a ω spread over more than 78000 deg into the target space which is defined over only a few degrees.

As in the previous example, the error is not so much in the density estimation, but in the exact location of it along the nodes. The quantiles depicting the fit accuracy is given in Figure 6.12a. The inaccuracies in the hypersurfaces introduce large errors. For the first three decades, the fit is accurate enough for a qualitative analysis of the evolution. Until there, 50% of the estimates, from $q_{25\%}$ to $q_{75\%}$ remain between -43% and $+5\%$. After that, randomisation in the angles could possibly be considered.

The propagation of the $C_0 = 2000$ characteristics and the 4000 hypersurface samples is performed within less than 3 mins. The average processing time of the 101 snapshots is 41 s each, or about 2 mins in total on the parallelised setup.

6.1.5 Ariane explosion: GTO considering full force model

For highly eccentric orbits, 3BPs and SRP should be considered. Here, the previous Ariane explosion scenario is extended to consider the full force model, i.e. atmospheric drag, J_2, 3BPs from the Sun and Moon and SRP. Note that the full force model now depends on the location of the Sun and Moon, the evolution of the cloud is highly dependent on the fragmentation epoch, set to be the 9 July, 2002 (NASA, 2018). The integration phase space is (a, e, i, Ω, ω, A/m). Only a short propagation interval is considered as the errors of the approximation grow quickly. To avoid the collapse of the

dimensionality, additional uncertainties in the position are introduced perpendicular to the velocity vector along in-plane and out-of-plane directions. The uncertainties are modelled as normal distributions with zero mean and standard deviations of $\sigma_r = 50$ km in each direction.

The initial distribution in Keplerian elements for $C_0 = 2500$ sampled characteristics (minus the 234 that re-enter or escape directly) is shown in Figure 6.14a. The final distribution, at $\Delta t = 1$ years, with its $C = 1381$ remaining characteristics is shown in Figure 6.14b. After just one year, and given the lowly inclined orbit, the effects of 3BP and SRP are difficult to discern (e.g. in the $i-\chi$ sub figure), as it only slightly affects the inclinations for the fragments.

For the fitting, the same target space as for the previous scenario is utilised, extended with the i-dimension. The hypersurfaces are also extended to be functions of i

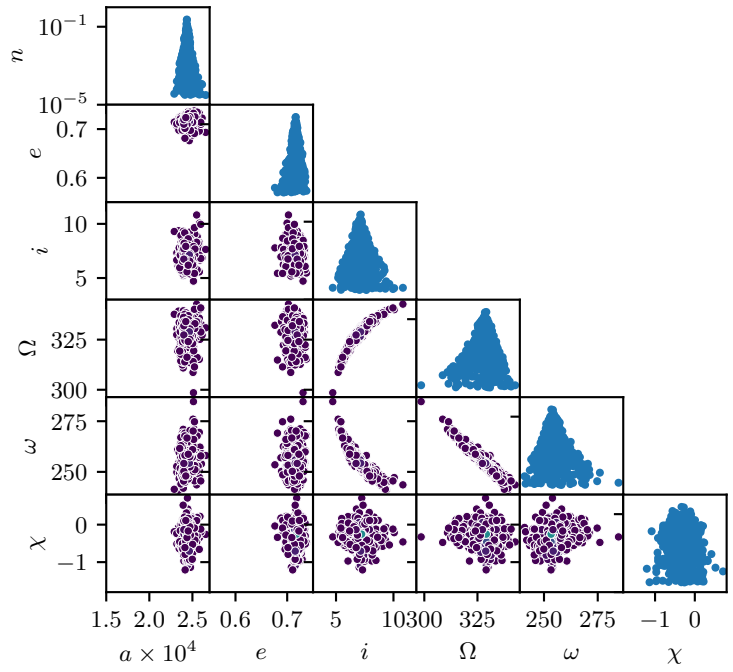
$$\hat{e} = \hat{e}\left(\zeta_p, i, \chi\right) \qquad \hat{\Omega} = \hat{\Omega}\left(\zeta_p, e, i, \chi\right) \qquad \hat{\omega} = \hat{\omega}\left(\zeta_p, e, i, \chi\right) \tag{6.6}$$

However, in this scenario, the hypersurfaces are obtained through linear simplex interpolation. Because of the additional perturbations, the quadratic interpolation scheme induces larger errors than the linear one. The hypersurfaces are not valid for long-term propagations because they miss some of the dependencies. For the fitting, $K = 20$ kernels, corresponding to 560 parameters, are optimised using $R = 10^{-2}$. Fitting is performed for 101 snapshots equally distributed in time over the year.

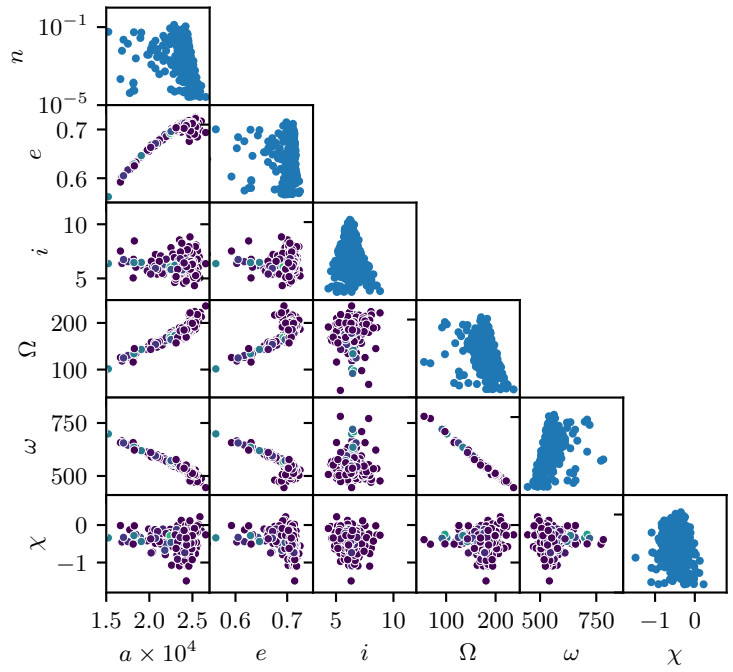
The resulting density estimates at $\Delta t = 1$ year for both, the fit characteristics and verification characteristics, are shown in Figure 6.15. Contrary to the previous cases, characteristics breaking out of the cloud can now be discerned also with the fit characteristics. This is because the hypersurfaces are not defined as a function of all its dependent variables, i.e. the characteristics are defined relative to an insufficiently accurate hypersurface. The optimisation procedure, however, is robust enough to converge to a decent solution, as most of the fitting points, and thus the underlying distribution, remain within a compact cloud.

The evolution of the quality fit can be seen in Figure 6.16, comparing the estimates to the true density given by the verification characteristics. The 50% of verification characteristics errors between $q_{25\%}$ and $q_{75\%}$ remain between -35% and $+12\%$ for the first quarter of the year but then grow quickly towards the end of the propagation span. Especially the lower quartile greatly underestimates the true density value at the verification characteristics. Some of the characteristics are pushed out of the cloud due to locally acting forces. If this local space is undersampled in terms of fitting characteristics, the resulting surrogate model is not able to catch such behaviour.

Another problem is the increase of the total integral over density ap-

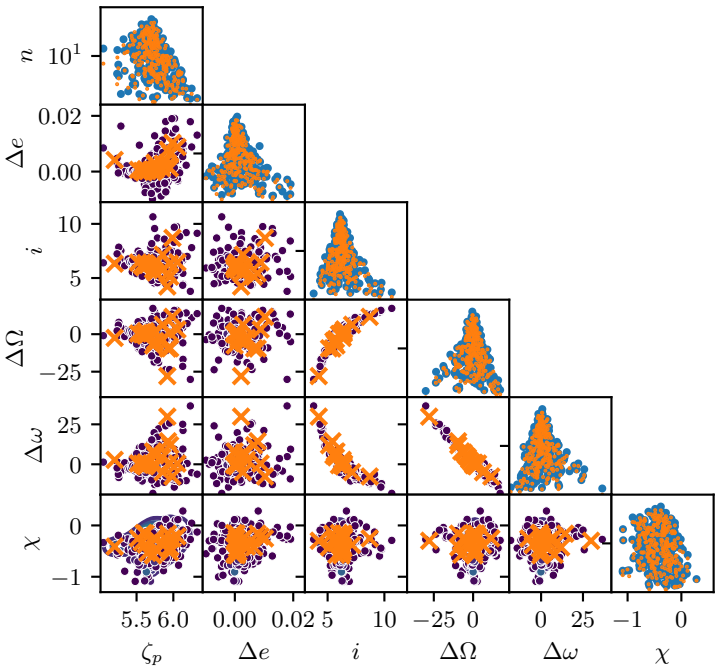


(a) Initial distribution, with $C = 2266$ characteristics, at $\Delta t = 0$ years.

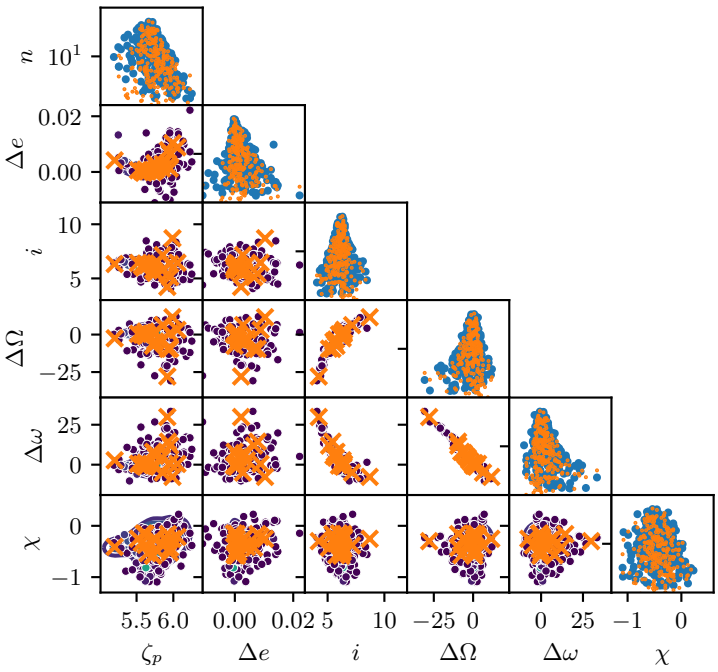


(b) Final distribution, with $C = 1381$ remaining, at $\Delta t = 1$ year.

Figure 6.14: Characteristics in Keplerian elements for the full Ariane explosion scenario. Note the different limits in Ω and ω. Units: a [km], e [−], i [deg], Ω [deg], ω [deg], χ [$\log_{10}(\mathrm{m}^2/\mathrm{kg})$].



(a) The optimisation procedure is robust enough to find a fit.



(b) However, the reference hypersurfaces are inaccurate.

Figure 6.15: Remaining characteristics in the fit space after $\Delta t = 1$ year for the full Ariane explosion scenario. Units: ζ_p [$\log_{10}$(m)], Δe [−], i [deg], $\Delta\Omega$ [deg], $\Delta\omega$ [deg], χ [$\log_{10}(\mathrm{m}^2/\mathrm{kg})$].

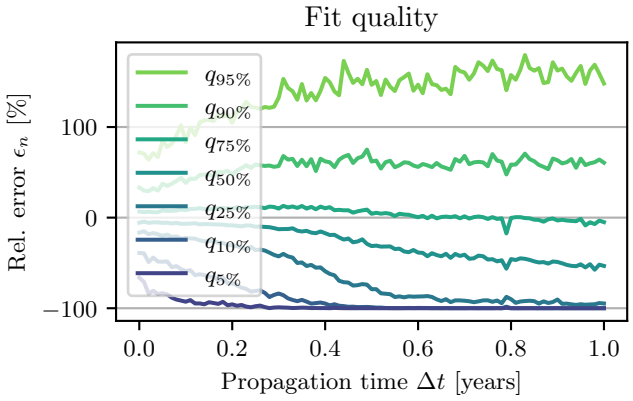


Figure 6.16: Fit quality for the full Ariane explosion scenario over $\Delta t = 1$ year: localised perturbations lead to inaccurate hypersurface definitions and error growth.

proximation from the surrogate model. Over the course of the simulated year, the integral remains nearly constant despite the apparent reduction with re-entering fragments. The origin of the error in the total integral is not entirely clear at this point. One possible explanation could come from the forces acting locally, lead to branching out of some of the characteristics. The model simply fits over parallel branches, ignoring lower density regions in between. This theory is supported by the considerable part of the verification characteristics where the model overestimates the true density.

The propagation of the $C_0 = 2500$ characteristics and the 5000 hypersurface samples is performed within 3 mins. On average, the 101 snapshots are fitted within 118 s each. However, some of the optimisation runs do not converge properly as the fitting leads to near-singular covariances. A better fit can generally be found through re-initialisation of the fitting routine.

Three final remarks conclude the part on the fragment cloud evolution. Trying to understand the dynamics in six dimensions is challenging as the phase spaces can only be studied in one dimensional and two-dimensional projected subspaces. Correlations between multiple variables cannot always be spotted. Therefore, it is important to start in low dimensions as presented in this section. Once the dynamics is understood in three dimensions, and appropriate hypersurfaces are defined, the next dimension can be considered. *Starling* natively handles any dimensionality and is thus suitably equipped for such a process chain.

The second remark is regarding automation. In three dimensions, in spaces derived from a, e and A/m, it is easy to come up with a set of rules for the target space and hypersurface depending on the fragmentation scenario. For the higher dimensional cases, however, studying of different

target spaces is required. Again, the proposed framework in *Starling* based on symbolical mathematics simplifies the exploration of a large set of target spaces. Nevertheless, automatically assigning target spaces depending on the scenario is not yet possible.

Finally, the extension of the dimensionality into 5 or 6 dimensions is tricky, because addition of an uncertainty in radial direction merely covers the fact that the dimension of the distribution is close to collapsing. Fitting of such a distribution is expensive and erroneous. Instead, a better approach would be to model the distribution only in 4 dimensions, i.e. transforming the three velocity and the area-to-mass ratio dependencies only. To recover the dependency on the full orbital element set, the remaining elements can be treated as dependent variables. As was shown above, Ω and ω can be interpolated from the other elements, constituting the hypersurface on which the 4 dimensional distribution itself is defined. This extension will be part of a future work, as outlined in Section 7.2.

6.2 Collision probability estimation

Once the evolution of the fragmentation cloud is known, the spatial density and the number of impacts can be estimated according to the method outlined in Section 2.4. No definition of a control volume, as is needed for the Cube algorithm (see Section 1.3.2), is required because the density is known at each point in space rather than estimated from deterministic instances. Here, the collision probability is estimated for different target orbits with the long-term Cosmos-2251 fragmentation cloud presented in Section 6.1.2.

The following figures are all weighted with the number of fragments, N, predicted by the NASA SBM for payload collisions. For catastrophic collisions, the model suggests that the involved mass is the combined mass of target and chaser. Here, the interest is on one of the fragment clouds only, hence, the mass is set to be $m = 900$ kg corresponding to the mass of Cosmos-2251 only (Kelso, 2009). Thus, the number of fragments released in the collision from Cosmos-2251 and sized between 1 mm and 1 m is estimated to be $N = 2.2 \times 10^6$.

The spatial density is calculated according to Equation (2.43), with an additional integration over the area-to-mass ratio, A/m, or to speed up the convergence of the integration, its logarithm to base 10, χ. The modelled spatial density of the Cosmos-2251 fragments down to 1 mm is presented in Figure 6.17 as of beginning of the year 2020, or $\Delta t = 11$ years after the event. The distribution is independent of the longitude, λ, as randomisation in Ω is assumed. The peak of the density is found around $h = 750$ km at high latitudes. The cloud is bounded by the parent inclination, $i_0 = 74$ deg. Of the initial fragments, 33%, or 0.73×10^6 fragments remain after $\Delta t = 11$ years. Of the remaining fragments, about 75%, or 0.55×10^6 fragments

Table 6.3: Selected targets and their orbits at beginning of the year 2020. The area, A, constitutes an average cross section. Sources: Space-Track (2020); ESA (2020).

Name	Int. Des.	h_p [km]	h_a [km]	i [deg]	ω [deg]	A [m^2]
Sentinal-1A	2014-016A	695	697	98.18	–	23.45
Proba-1	2001-049B	531	631	97.70	298.35	0.80
Ariane 5	2015-039C	254	34684	5.06	131.10	42.12

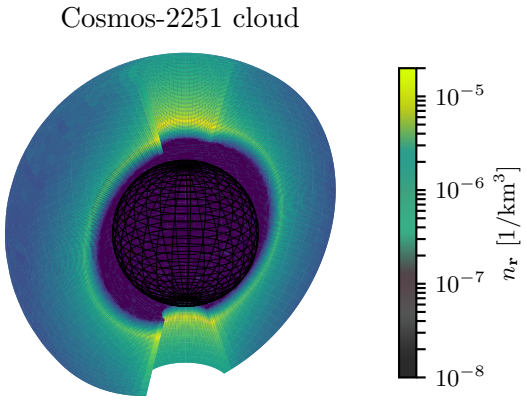
are found in LEO, the rest above. I.e. a collision in LEO also affects orbital regimes above it.

Before studying the impact rates for different target orbits, an example of the velocity distribution within the orbital plane (i.e. rotated around $\hat{x}$ for $i_0 = 74$ deg) at $h = 750$ km and $\theta = 0$ deg is shown in Figure 6.18 at the ascending node ($u = 0$ deg). The distribution for the descending node is the same, as randomisation is assumed. The highest density corresponds to near-circular objects at $a = R_E + h$, corresponding to a circular velocity of $v = 7.48$ km/s. Lower velocities are possible, however, the population goes to zero for perigee heights below 400 km. The transparent part of the figure includes such orbits and others that cross the Earth, i.e. are not feasible. Objects with high apogee altitudes, instead, are present and contribute significantly to the collision probability with their increased velocities.

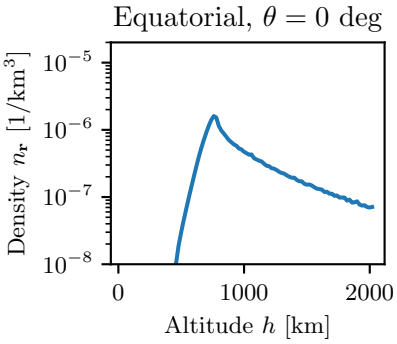
The studied target orbits, selected to cover a broad range in eccentricities, are given in Table 6.3. Note that as the fragmentation cloud is assumed to be randomised over Ω, ω and M, the corresponding values of the target orbits are not required for the calculation of the impact rate. The impact rate is calculated according to Equation (2.46) but in Keplerian elements, and again, integrated over χ. A constant target cross-sectional area, A, is assumed and given in Table 6.3. The impact rate for the three target objects at $\Delta t = 11$ years and as a function of the argument of latitude, u, is given in Figure 6.19a.

Both, Sentinel-1A and Proba-1 are on orbits inclined higher than the fragmentation cloud. Hence, close to the poles the satellites are outside of the cloud and receive zero impacts. Just before crossing in or out of the band, the fragment density and thus the impact rate is highest. The Ariane 5 stage, instead, always remains within the inclination band of the cloud. Close to the perigee, the Ariane 5 goes so low that it enters a fragment free region. Close to the apogee, the impact rate becomes tiny, but remains non-zero, as few fragments are in such highly eccentric orbits. This nicely shows the potential of the proposed method: the effect of an explosion in LEO can be felt even at GEO altitudes.

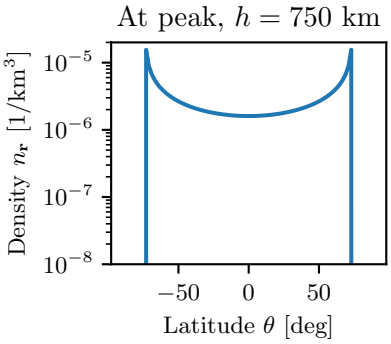
Sentinel-1A collects the most impacts as it operates close to the peak



(a) Spatial distribution, being independent of λ because of the assumed randomisation in Ω.



(b) Radial distribution in LEO with a peak density at $h = 750$ km.



(c) The cloud is bounded by the parent inclination, $i_0 = 74$ deg.

Figure 6.17: Modelled spatial density of Cosmos-2251 fragments as of the beginning of the year 2020, i.e. $\Delta t = 11$ years after the fatal event, for fragments down to 1 mm.

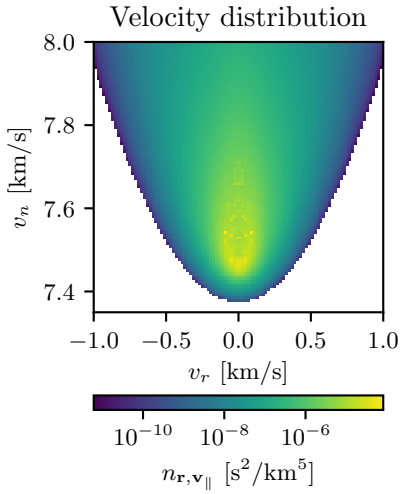
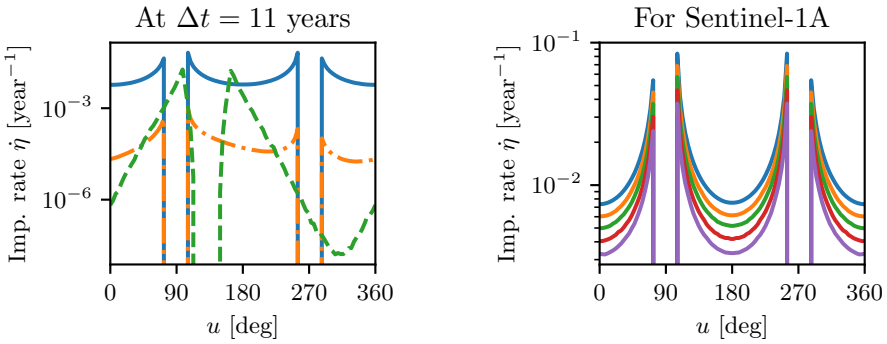


Figure 6.18: Velocity distribution of Cosmos-2251 fragments, sitting in the orbital plane at $h = 750$ km and $\theta = 0$ deg.



(a) Impact rates for Sentinel-1A (solid), Proba-1 (dash-dot), and Ariane 5 (dashed) along the orbits.

(b) Historical and future impact rates on Sentinel-1A at different times. From top to bottom: $\Delta t = 5, 10, 15, 20, 25$ years.

Figure 6.19: Impact rates of Cosmos-2251 fragments on selected target orbits.

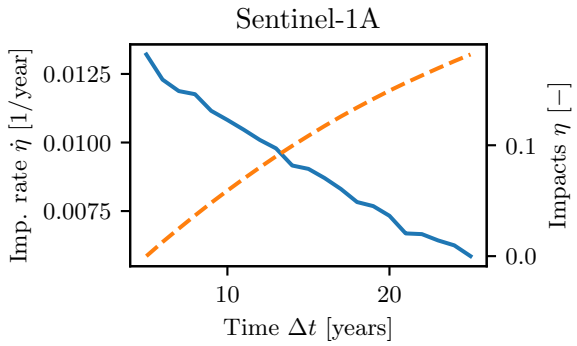


Figure 6.20: Impact rates (solid) and number of impacts (dashed) from launch in 2014, i.e. after $\Delta t = 5$ years of the event, up to the year 2034.

density altitude. Averaged over one orbit, and extrapolated to one year, the fragment rate incident on Sentinel-1A is 0.0105 per year. Proba-1 resides in lower altitudes. Given the slightly eccentric orbit, the impact rate profile is less symmetric than for Sentinel-1A. Averaged over one orbit, Proba-1 receives $\dot{\eta} = 7.37 \times 10^{-5}$ impacts per year from Cosmos-2251 fragments larger than 1 mm. For the Ariane 5, most of the impacts are collected during crossing of the peak density altitude. Averaged over one orbit, the number of impacts is $\dot{\eta} = 1.62 \times 10^{-4}$ per year.

The time evolution of the impact rate for the case of the Sentinel-1A satellite – assuming a constant mission orbit – is shown in Figure 6.19b. The snapshots start at $\Delta t = 5$ years, i.e. at the launch of the satellite in 2014, and go up to $\Delta t = 25$ years, i.e. to the year 2034, in steps of 5 years. The yearly impact rate and the cumulative number of impacts – calculated by summing the average value between two snapshots, taken to be constant for the time interval –, and weighted with the area, is given in Figure 6.20. Note that the impact rate shows some jitter, which is likely a consequence of the model fitting. Still, a high level of consistency is achieved over the full range.

The cumulative number of impacts of Cosmos-2251 fragments on Sentinel-1A is $\eta = 0.067$ and $\eta = 0.18$ after $\Delta t = 11$ years and $\Delta t = 25$ years, respectively, corresponding to exposure of the satellite to the cloud for 6 years and 20 years. The cumulative probability of having one or more collision during this time, i.e. evaluating Equation (1.29), is therefore $p_{1+} = 6.5\%$ and $p_{1+} = 16.5\%$, at the two epochs. Coincidentally, Sentinel-1A suffered an impact in 2016, probably caused by space debris (Krag et al., 2017b). However, the impactor is thought to be approximately 1 cm in characteristic length, making a Cosmos-2251 fragment less likely than the probabilities reported above. Repeating the process of propagating the cloud and estimat-

ing the impact rate against Cosmos-2251 fragments of 1 cm and larger shows that the total number of impacts after $\Delta t = 25$ years is 30 times smaller than the one reported for fragments sized larger than 1 mm. For fragments sized 10 cm and larger, the number of impacts is estimated to 3 orders of magnitude lower. Even if there is no way of knowing exactly the origin of the fragment, the proposed continuum method can be used to probabilistically assign chasers to potential parent objects and fragmentation events.

7 Conclusions

Fragmentation clouds in Earth orbit pose a threat to current and future space missions. Owing to the high orbital velocities, even small fragments can have devastating effects if they collide with other space objects. Collisions potentially introduce new clouds of space debris leading to self-perpetuating growth in number of objects. To protect this delicate environment, guidelines and standards are conceptualised and introduced. They are refined regularly to adapt to new developments. The tools that currently underpin such efforts estimate the number of collisions or objects for a variety of input variables over long timespans. However, they fail to foster the understanding about the spatial density evolution, let alone the evolution of the full phase space density, in an efficient and accurate manner. Nor can they accurately assess the effects of single fragmentation events.

This thesis introduced new techniques and methodologies sensitive enough to estimate the negative ramifications of fragmentation clouds on single missions. At the same time, restricting assumptions on orbit geometries and force models are dropped making the method applicable to a large set of fragmentation scenarios. A more extensive summary of the work contained in this thesis is given in Section 7.1. Current limitations, and how to address them in the future, are discussed in Section 7.2, together with suggestions on how extend the applicability of the method to problems beyond space debris.

7.1 Summary of work

The main objectives of this research are to find a way of propagating a fragmentation cloud into the future, irrespective of the orbit geometry, the force model, or the time horizon, and to quantify its ramifications on other space missions. Simultaneously, high sensitivity is required to estimate the implications of a single fragmentation on other space objects. In addition, the algorithm estimating the evolution of the cloud should be computationally inexpensive to allow for large scale deployment of the method. Here, it is

summarised how these objectives were reached.

7.1.1 Methodology

In a first step, existing methods are studied herein, and advantages and disadvantages are discussed. The continuum method, based on the continuity equation, is identified as a suitable candidate because of its statistical representation of the distribution, scalability in dimensions and sensitivity and accuracy along the characteristics. Combined with numerical integration, it can predict the density evolution from any orbital region and subject to any type and number of perturbations.

The propagation of the object trajectories is performed numerically in averaged elements. Hence, an analytical method is presented to transform the initial fragment distribution from Cartesian coordinates to Keplerian elements and vice-versa. The analytical formulation allows to remain in a fully probabilistic framework, making the transformation process nearly instantaneous while retaining exact density values. To this end, the NASA Standard Breakup Model is extended to provide the full density distribution function rather than simply representative samples.

Given the density as a function of orbital elements allows to employ randomisation in some of the elements of the distribution in a straightforward manner. Randomisation in tandem with transformation into Cartesian coordinates enables the calculations of the spatial density or number of impacts – and thus collisions risk – from a distribution given in a subset of, or full orbital element set, without resorting to any geometrical assumptions. This makes the method not only highly versatile in its application to short-, mid- and long-term scenarios, but also highly efficient, as reduced phase spaces can be considered.

Next to advancements in the cloud evolution methodology, an extension of the King-Hele orbital contraction method is presented. The extension, based on a simplified but accurate density profile, provides an analytical formulation of the averaged decay rate in semi-major axis and eccentricity for any orbital regime, removing inaccuracies of the original formulation. The superimposed approach is also applicable to other analytical averaging techniques. Furthermore, the formulation enables to analytically evaluate the Jacobian of the dynamics, facilitating the propagation of the density along the characteristics. Additionally, a new boundary condition is introduced to switch between the different averaging formulations for low and highly eccentric orbits. The new boundary condition is derived by comparing the approximation errors of the series expansion in both formulations, constituting an optimal boundary. More terms are added in the series expansion to further increase the accuracy of the method.

For quick evaluation of the phase space density over the full domain, it is proposed to fit a surrogate model to the cloud of characteristics. The

characteristics are sampled from the initial distribution to well represent the bulk of the fragments. A Gaussian Mixture Model is selected as the surrogate model for multiple reasons; it is an universal density function approximator, positive function values are easily enforced across the full domain, multiple peaks can be represented, its derivatives with respect to the parameters are readily obtained, the total integral can be calculated analytically, and it can be interpreted even in higher dimensions. The propagation of the characteristics and the model fitting are separated, allowing to test various target spaces and number of kernels without having to integrate repeatedly.

Multiple optimisation methods, with various degrees of complexity, are compared and benchmarked to find an optimisation algorithm well suited to fit the surrogate model to the characteristics. The limited-memory Broyden–Fletcher–Goldfarb–Shanno algorithm based on the *Quasi-Newton* method is shown to rapidly converge to a local minimum. Instead of evaluating the second partial derivatives of the cost function with respect to the optimisation parameters, which is computationally expensive, it keeps an approximation of the inverse of the Hessian, built from gradient information only. As such, it can fit hundreds of optimisation parameters within a couple of seconds only. However, the comparison of the methods is not conclusive, as none of the methods was fine-tuned in terms of its parameters.

The dynamics lead to non-linear displacements of the fragmentation cloud, even if expressed in orbital elements. For long-term propagation, this can lead to phase space densities that are condensed into curved tubes, impeding the surrogate model from appropriately representing the underlying density. Further complicating the fitting are initial cloud distributions that exhibit boundedness in the phase space used for propagation. To tackle these problems, normalisation of the phase space through reference hypersurfaces is performed. These hypersurfaces are defined through and interpolated from hypersurface samples that are propagated alongside the cloud characteristics. They are proven to be an effective means of simultaneously absorbing non-linearities and boundedness. In the resulting, normalised target spaces, efficient and accurate fitting of the surrogate model can be performed. Transformation of the density from a phase space defined relative to the space of interest is straightforward as it is described by a simple translation.

The methodology of propagating the characteristics and fitting the surrogate model in a target space relative to a hypersurface is successfully validated against two methods. The first method is based on the propagation of a set of verification characteristics, identically sampled as the characteristics used for the fitting. The verification characteristics represent the exact value of the density and thus allow to compare the approximation from the surrogate model against a truth value. This verification method is powerful as it can be easily applied to problems of any dimension. Application of the continuum propagation method should always be accompanied with

such a verification to estimate inaccuracies. The second verification method is brute force sampling and binning of the samples to estimate the density. Although it is inherently flawed as it does not provide an exact estimate of the true density it can still be used in low dimensions to validate not only the surrogate model but the continuity propagation itself.

7.1.2 Starling and Applications

The newly developed techniques are implemented in *Starling*, the software suite for the propagation of continua, such as orbital fragment distributions, uncertainties or other phenomena that can be described probabilistically. The suite is highly modular, and as such is not restricted to orbital dynamics only. Clearly defined interfaces and input files allow to study the evolution of a continuum given any initial distribution and any dynamics.

Symbolic representations of phase spaces enables the study of many target spaces. Removing the need of manually deriving and implementing the transformation of the density not only increases the efficiency in the manipulation of the data, but also cuts out a source of errors that potentially lead to the premature exclusion of suitable spaces.

The proposed methodologies and its implementation in *Starling* achieve high accuracies in a computationally efficient way. Parallelisation of all the demanding tasks reduces the time required to simulate the evolution of a fragmentation cloud. Given 40 cores, problems in three dimensions can be studied in less than two minutes. Compared to a Monte Carlo simulation that propagates one million samples on the same parallelised setup, requiring 4 hours, that equates to two orders of magnitude speed up using the same computational setup. Even in six dimensions, the propagation and fitting to 100 snapshots can be performed and in less than 10 minutes. The continuum approach thus gives a measure of the directional components, for which even millions of samples are not sufficient in the Monte Carlo simulation.

Starling is applied to a large set of evolution scenarios for various initial conditions, force models and orbital geometries. It is proven to be adept in switching to any dimension suitable for the problem. Accurate results are achieved over long time spans for low dimensional problems. The solution of higher dimensional problems still permits to draw qualitative conclusions on the evolution of the cloud. For the three-dimensional case derived from the semi-major axis, eccentricity, and area-to-mass ratio, suitable for the long-term propagation of clouds in Low Earth Orbit, rules are devised that permit automation of the method. Hence, the method can be established for indexing purposes or analysis of the capacity, where the ramifications of orbital breakups from many parent objects need to be evaluated.

Examples are given showing how the resulting evolution can be converted into spatial densities and collision probability estimates for targets in any orbit, highlighting the sensitivity of the presented method. Next to esti-

mating the ramifications of fragmentations, it can be used for visualisation and educational purposes, identification of parent objects, optimisation of flight trajectories and attitudes of spacecraft to avoid fragments and the enforcement of space debris mitigation measures.

7.2 Limitations & future work

Despite the achievements of the research presented in this thesis, some limitations remain. They are presented here including possible future work to address them. Currently, the main limitation of the method is that fitting of the surrogate model is not applicable to long-term scenarios of fragmentations in orbits where solar radiation pressure and third body perturbations cannot be neglected. Even though the density can be accurately estimated along the characteristics for any force model and geometry, the surrogate model cannot accurately be fitted once the perturbations distorted the initially compact scattered cloud of characteristics to a large extent. Consequently, the surrogate model adds density to regions where it should not. This leads to a faulty increase in the total number of fragments.

A possible solution to this problem is decreasing the complexity of the fitting routine by exploiting the fact that the initial distribution is limited to four dimensions: the three velocity components and the area-to-mass ratio. Instead of artificially inflating the dimensionality of the problem by adding uncertainties in radial direction, the missing elements could be inferred from the dependent variables and modelled as Dirac delta distribution functions. E.g. the cloud density is evaluated and fitted in the four-dimensional phase space variables semi-major axis, eccentricity, inclination and area-to-mass ratio only, but the trajectories of the characteristics are propagated considering the full element set. To obtain an approximation of the distribution in the full element set, the node and argument of perigee, i.e. the dependent variables, are subsequently, and independently of the phase space density, interpolated from the independent phase space variables. First tests of this simplification have shown promising results and will be explored further in a dedicated journal publication.

Another possible remedy to the distortion of the cloud distribution subject to solar radiation pressure and third body perturbations is automated domain splitting. Wittig et al. (2015) applied this technique for the long-term propagation of orbit uncertainties using differential algebra. In the differential algebra environment, errors build up during the propagation, requiring the domain splitting to occur on-line. Here, the density values along the characteristics are exact even for highly non-linear dynamics. Hence, the domain splitting, trying to isolate branching from parts of the distribution, can be applied off-line in a post-processing step. I.e. if the errors of the surrogate model are found to be too large with respect to the fitting charac-

teristics, the domain can be split into sub domains, and fitted individually.

A third possible solution is to avoid using a surrogate model in the first place. A solution is outlined here that simultaneously addresses a second shortcoming present in the estimation of the products. The integrals to calculate the spatial density and the number of impacts – involving marginalisation in two or higher dimensions depending on the number of parameters – cannot be analytically evaluated. Instead, they are solved through numerical integration on a regular grid. Inaccuracies arise from the problem that the target object might only be affected by the tail-end of the fragment cloud distribution, and a grid over the fragment cloud results in many function evaluations with negligible contributions to the integral. To avoid evaluating the density on a regular grid, the integral could be evaluated using importance sampling (Bishop, 2006, Chapter 11). Hence, instead of sampling the initial distribution, the integrand is sampled using a Markov Chain Monte Carlo method. As such, function evaluations would be performed where it is useful for the estimation of the integral, i.e. where the density is high, reducing the computational effort and simultaneously increasing the accuracy of the integration. If it can be shown that the number of required evaluations of the integrand can be reduced to a sufficiently low number, the surrogate modelling step can be skipped, and the densities directly evaluated through backpropagation of the characteristics. Instead, a combined method could be developed, using an imperfect surrogate model to draw initial samples of the integrand, which are subsequently updated through backpropagation.

In the current implementation of *Starling*, the optimisation is performed using a local minimisation schema. This could be problematic if multimodal distributions are to be modelled, where a local optimal minimum would approximate the full distribution only partially. The fitting of the surrogate model could greatly be improved by investigating global optimisation methods, such as genetic algorithms or particle swarms (e.g. Hendrix and G.-Tóth, 2010, Chapter 7.5).

Another shortcoming is that the current usage of reference hypersurfaces renders the originally parametric definition of the model non-parametric. Thus, it is not sufficient to describe the distribution with parameters only, but it needs to be accompanied with hypersurface samples. Finding a parametric description of the reference hypersurfaces would allow to remove the hypersurface samples after fitting, and possibly to obtain an analytical solution of the integrals giving spatial density and number of impacts. Instead of interpolating the samples, a hypersurface could be fitted instead, by minimising a distance measure.

Finding suitable target spaces for the model fit in higher dimension is a non-trivial task. Understanding of the initial distribution and the evolution governed by the dynamics is required. In the absence of this understanding, the user is required to resort to trial and error, which can be time consuming

without guarantee of success. An additional interesting feature could be for the software to automatically find and propose suitable target spaces. Various features spaces could be combined, tested, and discarded based on error measures from cross validation. This might even lead to the discovery of new element sets that are suitable to describe the evolution of orbital uncertainties.

Furthermore, *Starling* could profit from becoming more user-friendly. To make it useful for educational purposes, for example, a graphical user-interface needs to be developed. Currently, the software is designed to study the evolution of single fragmentation clouds on single target objects. To study the effect of potential fragmentations on the full satellite population, a multiple-target feature needs to be added to reduce the workload on the analyst.

Finally, the methods developed herein can be applied to various other problems. Short-term propagation of uncertainties, i.e. considering the fast variable, is a possible field of interest, as is the application of the method to the full space debris population. In the latter case, collisions among the continuum could be considered through addition of a collision source term during the propagation of the characteristics. This requires a regular update of the collision probability enabled through the methods presented in this thesis.

Bibliography

M. Abramowitz and I. A. Stegun. *Handbook of Mathematical Functions: with Formulas, Graphs, and Mathematical Tables*. Dover Publications, 10^{th} edition, 1972. ISBN 978-0486612720.

J. Ashenberg. On the short-term spread of space debris. In *Proceedings of the AIAA/AAS Astrodynamics Conference*, 1992.

J. Ashenberg. Perturbed volume of orbiting debris. *Journal of Spacecraft and Rockets*, 31(3):526–528, 1994. doi:10.2514/3.26471.

C. Au and J. Tam. Transforming variables using the Dirac generalized function. *The American Statistician*, 53(3):270–272, 1999. doi:10.2307/2686109.

S.-K. Au and J. L. Beck. Estimation of small failure probabilities in high dimensions by subset simulation. *Probabilistic Engineering Mechanics*, 16 (4):263–277, 2001. doi:10.1016/S0266-8920(01)00019-4.

J. N. Bass. Condensed storage of diffusion equation solution for atmospheric density model computations. Technical report, Air Force Geophysics Laboratory, 1980.

R. H. Battin. *An Introduction to the Mathematics and Methods of Astrodynamics, Revised Edition*. American Institute of Aeronautics and Astronautics, Inc., 1999. ISBN 978-1563473425.

C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006. ISBN 978-0-387-31073-2.

D. Brouwer. Solution of the problem of artificial satellite theory without drag. *Astronomical Journal*, 64(1274):378–396, 1959. doi:10.1086/107958.

S. Bruinsma. The DTM-2013 thermosphere model. *Journal of Space Weather and Space Climate*, 5(A1):1–8, 2015. doi:10.1051/swsc/2015001.

S. Chandrasekhar. Brownian motion, dynamical friction, and stellar dynamics. *Reviews of modern physics*, 21(3):383–388, 1949. doi:10.1103/RevModPhys.21.383.

S. Chandrasekhar. *Principles of Stellar Dynamics*. Dover Publications, 2005. ISBN 978-0486442730.

S. Chib and E. Greenberg. Understanding the Metropolis–Hastings algorithm. *The American Statistician*, 49(4):327–335, 1995. doi:10.2307/2684568.

V. A. Chobotov. Classification of orbits with regard to collision hazard in space. *Journal of Spacecraft and Rockets*, 20(5):484–490, 1983. doi:10.2514/3.25633.

C. Colombo. Planetary orbital dynamics (PlanODyn) suite for long term propagation in perturbed environment. In *Proceedings of the 6th International Conference on Astrodynamics Tools and Techniques*, 2016.

D. A. Danielson, C. P. Sagovac, B. Neta, and L. W. Early. Semianalytic satellite theory. Technical report, Naval Postgraduate School, 1995.

J. C. Dolado-Perez, R. Di Costanzo, and B. Revelin. Introducing MEDEE - a new orbital debris evolutionary model. In *Proceedings of the 6th European Conference on Space Debris*, 2013.

J. C. Dolado-Perez, B. Revelin, and R. Di Costanzo. Sensitivity analysis of the long-term evolution of the space debris population in LEO. *Journal of Space Safety Engineering*, 2(1):12–22, 2015. doi:10.1016/S2468-8967(16)30035-0.

G. Drolshagen. Hypervelocity impact effects on spacecraft. In *Proceedings of the Meteoroids Conference*, 2001.

ESA. Space Debris Office. Annual Space Environment Report, 2019.

ESA. DISCOSweb: A web based frontend to ESA's DISCOS, 2020. https://discosweb.esoc.esa.int/.

P. Farinella and A. Cordelli. The proliferation of orbiting fragments: a simple mathematical model. *Science and Global Security*, 2:365–378, 1991. doi:10.1080/08929889108426373.

D. Finkleman, D. Oltrogge, A. Faulds, and J. Gerber. Analysis of the response of a space surveillance network to orbital debris events. In *Proceedings of the AAS/AIAA Astrodynamics Specialist Conference*, 2008.

A. I. J. Forrester, A. Sóbester, and A. J. Keane. *Engineering Design via Surrogate Modelling, A Practical Guide*. John Wiley & Sons, Ltd., 2008. ISBN 978-0470060681.

A. Francesconi, C. Giacomuzzo, L. Olivieri, G. Sarego, M. Duzzi, F. Feltrin, A. Valmorbida, K. D. Bunte, M. Deshmukh, E. Farahvashi, J. Pervez, M. Zaake, T. Cardone, and D. de Wilde. CST: a new semi-empirical tool for simulating spacecraft collisions in orbit. *Acta Astronautica*, 160: 195–205, 2019. doi:10.1016/j.actaastro.2019.04.035.

S. Frey and C. Colombo. Transformation of satellite breakup distribution for probabilistic orbital collision hazard analysis. *Journal of Guidance, Control, and Dynamics*, accepted for publication in July, 2020. doi:10.2514/1.G004939.

S. Frey, C. Colombo, and L. Lemmens. Application of density-based propagation to fragment clouds using the starling suite. In *Proceedings of the 1st International Orbital Debris Conference*, 2019a.

S. Frey, C. Colombo, and S. Lemmens. Extension of the King-Hele orbit contraction method for accurate, semi-analytical propagation of non-circular orbits. *Advances in Space Research*, 64(1):1–17, 2019b. doi:10.1016/j.asr.2019.03.016.

S. Frey, C. Colombo, and S. Lemmens. Evolution of orbital fragmentation cloud via continuum and surrogate modelling. To be submitted to the *Journal of Guidance, Control, and Dynamics* in September, 2020.

Goddard Space Flight Center. NASA's space physics data facility OMNIWeb plus, 2020. https://omniweb.gsfc.nasa.gov/.

D. González, D. Díaz, and S. Davis. Continuity equation for probability as a requirement of inference over paths. *European Physical Journal B*, 89 (214):1–5, 2016. doi:10.1140/epjb/e2016-70307-0.

J. L. Gonzalo, C. Colombo, and P. Di Lizia. Analysis and design of collision avoidance manoeuvres. To be submitted to the *Journal of Guidance, Control, and Dynamics* in 2020, 2020.

I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. The MIT Press, 2016. ISBN 978-0262035613.

N. N. Gor'kavyi, L. M. Ozernoy, and J. C. Mather. A new approach to dynamical evolution of interplanetary dust. *The Astrophysical Journal*, 474(1):496–502, 1997. doi:10.1086/303440.

A. Halder and R. Bhattacharya. Dispersion analysis in hypersonic flight during planetary entry using stochastic Liouville equation. *Journal of Guidance, Control, and Dynamics*, 34(2):459–474, 2011. doi:10.2514/1.51196.

T. Hanada, Y. Ariyoshi, K. Miyazaki, K. Maniwa, J. Murakami, and S. Kawamoto. Orbital debris modeling at Kyushu university. *The Journal of Space Technology and Science*, 24(2):23–35, 2009. doi:10.11230/jsts.24.2_23.

L. Healy, S. Kindl, and C. Binz. Spatial density maps from a debris cloud. In *Proceedings of the Advanced Maui Optical and Space Surveillance Technologies Conference*, 2016.

W. B. Heard. Dispersion of ensembles of non-interacting particles. *Astrophysics and Space Science*, 43:63–82, 1976. doi:10.1007/BF00640556.

E. M. T. Hendrix and B. G.-Tóth. *Introduction to Nonlinear and Global Optimization*. Springer, 2010. ISBN 978-0-387-88669-5.

J. T. Horwood, N. D. Aragon, and A. B. Poore. Gaussian sum filters for space surveillance: theory and simulations. *Journal of Guidance, Control, and Dynamics*, 34(6):1839–1851, 2011. doi:10.2514/1.53793.

J. D. Hunter. Matplotlib: a 2D graphics environment. *Computing in Science and Engineering*, 9(3):90–95, 2007. doi:10.1109/MCSE.2007.55.

IADC. Space debris mitigation guidelines. Action Item number 22.4, 2007.

IADC. Space debris. Assessment Report for 2011, 2013a.

IADC. Stability of the future LEO environment. Action Item 27.1, 2013b.

IADC. Statement on large constellations of satellites in low earth orbit, 2017.

Instituto Nacional De Pesquisas Espaciais. A (huge) collection of interchangeable empirical models to compute the thermosphere density, temperature and composition, 2020. http://www.dem.inpe.br/.

ISO. Space environment (natural and artificial) - Earth upper atmosphere. ISO 14222:2013, 2013.

ISO. Space systems - space debris mitigation requirements. ISO 24113:2019, 2019.

D. Izzo and C. Valente. A mathematical model representing the statistical properties of sets of orbits. *Acta Astronautica*, 54(8):541–546, 2004. doi:10.1016/S0094-5765(03)00224-8.

L. G. Jacchia. Thermospheric temperature, density, and composition: New models. *SAO Special Report*, 375, 1977.

R. Jehn. Dispersion of debris clouds from on-orbit fragmentation events. In *Proceedings of the 41st International Astronautical Congress*, 1990.

A. B. Jenkin. Probability of collision during the early evolution of debris clouds. *Acta Astronautica*, 38(4–8):525–538, 1996. doi:10.1016/0094-5765(96)00059-8.

F. John, J. P. LaSalle, and L. Sirovich. *Partial Differential Equations*. Springer, 4th edition, 1981. ISBN 978-0387906096.

N. L. Johnson, P. H. Krisko, J.-C. Liou, and P. D. Anz-Meador. NASA's new breakup model of EVOLVE 4.0. *Advances in Space Research*, 28(9): 1377–1384, 2001. doi:10.1016/S0273-1177(01)00423-9.

B. A. Jones, N. Parrish, and A. Doostan. Postmaneuver collision probability estimation using sparse polynomial chaos expansions. *Journal of Guidance, Control, and Dynamics*, 38(8):1425–1437, 2015. doi:10.2514/1.G000595.

E. Jones, T. Oliphant, and P. Peterson. SciPy: Open source scientific tools for Python, 2001. http://www.scipy.org/.

B. Kaufman and R. Dasenbrock. Higher order theory for long-term behaviour of earth and lunar orbiters. Naval Research Laboratory, 1972.

W. M. Kaula. Development of the lunar and solar disturbing functions for a close satellite. *Astronomical Journal*, 67:300–303, 1962. doi:10.1086/108729.

W. M. Kaula. *Theory of Satellite Geodesy: Applications of Satellites to Geodesy*. Blaisdell Publishing Company, 1966. ISBN 978-0486414652.

T. S. Kelso. Analysis of the Iridium 33–Cosmos 2251 collision. In *Proceedings of the AAS/AIAA Astrodynamics Specialist Conference*, 2009.

D. J. Kessler. Derivation of the collision probability between orbiting objects: the lifetimes of Jupiter's outer moons. *Icarus*, 48(1):39–48, 1981. doi:10.1016/0019-1035(81)90151-2.

D. J. Kessler and B. G. Cour-Palais. Collision frequency of artificial satellites: The creation of a debris belt. *Journal of Geophysical Research: Space Physics*, 83(A6):2637–2646, 1978. doi:10.1029/JA083iA06p02637.

D. King-Hele. *Satellite Orbits in an Atmosphere: Theory and Application*. London: Butterworths Mathematical Texts, 1964. ISBN 978-0216922525.

H. Krag, K. Merz, T. Flohrer, S. Lemmens, B. Bastida Virgili, Q. Funke, and V. Braun. ESA's modernised collision avoidance service. In *Proceedings of the 14th International Conference on Space Operations*, 2016.

H. Krag, S. Lemmens, and F. Letizia. Space traffic management through the control of the space environment's capacity. In *Proceedings of the 1st IAA Conference on Space Situational Awareness*, 2017a.

H. Krag, M. Serranoa, V. Braun, P. Kuchynka, M. Catania, J. Siminski, M. Schimmerohn, X. Marc, D. Kuijper, I. Shurmer, A. O'Connell, M. Otten, I. Muñoza, J. Morales, M. Wermuth, and D. McKissock. A 1 cm space debris impact onto the Sentinel-1A solar array. *Acta Astronautica*, 137:434–443, 2017b. doi:10.1016/j.actaastro.2017.05.010.

A. V. Krikov and J. Getino. Orbital evolution of high-altitude balloon satellites. *Astronomy and Astrophysics*, 318:308–314, 1997.

P. H. Krisko. The predicted growth of the low-earth orbit space debris environment – an assessment of future risk for spacecraft. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 221(6):975–985, 2007. doi:10.1243/09544100JAERO192.

D. T. Lee and B. J. Schachter. Two algorithms for constructing a Delaunay triangulation. *International Journal of Computer and Information Sciences*, 9(3):219–242, 1980. doi:10.1007/BF00977785.

F. Letizia. Extension of the density approach for debris cloud propagation. *Journal of Guidance, Control, and Dynamics*, 41(12):2650–2656, 2018. doi:10.2514/1.G003675.

F. Letizia, C. Colombo, and H. G. Lewis. Analytical model for the propagation of small-debris-object clouds after fragmentations. *Journal of Guidance, Control, and Dynamics*, 38(8):1478–1491, 2015. doi:10.2514/1.G000695.

F. Letizia, C. Colombo, and H. G. Lewis. Multidimensional extension of the continuity equation method for debris clouds evolution. *Advances in Space Research*, 57(8):1624–1640, 2016a. doi:10.1016/j.asr.2015.11.035.

F. Letizia, C. Colombo, and H. G. Lewis. Collision probability due to space debris clouds through a continuum approach. *Journal of Guidance, Control, and Dynamics*, 39(10):2240–2249, 2016b. doi:10.2514/1.G001382.

F. Letizia, S. Lemmens, B. Bastida Virgili, and H. Krag. Application of a debris index for global evaluation of mitigation strategies. *Acta Astronautica*, 161:348–362, 2019. doi:10.1016/j.actaastro.2019.05.003.

H. G. Lewis, G. G. Swinerd, C. E. Martin, and W. S. Campbell. The stability of disposal orbits at super-synchronous altitudes. *Acta Astronautica*, 55 (3–9):299–310, 2004. doi:10.1016/j.actaastro.2004.05.022.

H. G. Lewis, S. Diserens, T. Maclay, and J. P. Sheehan. Limitations of the cube method for assessing large constellations. In *Proceedings of the 1st International Orbital Debris Conference*, 2019.

J.-C. Liou. Collision activities in the future orbital debris environment. *Advances in Space Research*, 38(9):2102–2106, 2006. doi:10.1016/j.asr.2005.06.021.

J.-C. Liou, D. T. Hall, P. H. Krisko, and J. N. Opiela. LEGEND - a three-dimensional LEO-to-GEO debris evolutionary model. *Advances in Space Research*, 34(5):981–986, 2004. doi:10.1016/j.asr.2003.02.027.

J. J. F. Liu. Satellite motion about an oblate earth. *AIAA Journal*, 12(11): 1511–1516, 1974. doi:10.2514/3.49537.

Y.-Z. Luo and Z. Yang. A review of uncertainty propagation in orbital mechanics. *Progress in Aerospace Sciences*, 89:23–39, 2017. doi:10.1016/j.paerosci.2016.12.002.

J. R. Magnus and H. Neudecker. Matrix differential calculus with applications to simple, Hadamard, and Kronecker products. *Journal of Mathematical Psychology*, 29(4):474–492, 1985. doi:10.1016/0022-2496(85)90006-9.

MathWorks, Inc. MATLAB R2019b, 2020. https://www.mathworks.com/.

C. R. McInnes. An analytical model for the catastrophic production of orbital debris. *ESA Journal*, 17:293–305, 1993.

C. R. McInnes and C. Colombo. Wave-like patterns in an elliptical satellite ring. *Journal of Guidance, Control, and Dynamics*, 36(6):1767–1771, 2013. doi:10.2514/1.55956.

M. D. McKay, R. J. Beckham, and W. J. Conover. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 21(2):239–245, 1979. doi:10.2307/1268522.

D. McKnight. A phased approach to collision hazard analysis. *Advances in Space Research*, 10(3-4):385–388, 1990. doi:10.1016/0273-1177(90)90374-9.

D. McKnight and G. Lorenzen. Collision matrix for low earth orbit satellites. *Journal of Spacecraft and Rockets*, 26(2):90–94, 1989. doi:10.2514/3.26037.

D. McKnight, R. Maher, and L. Nagl. Refined algorithms for structural breakup due to hypervelocity impact. *International Journal of Impact Engineering*, 17(4–6):547–558, 1995. doi:10.1016/0734-743X(95)99879-V.

J. Meeus. *Astronomical Algorithms*. Willmann-Bell, Inc., 2nd edition, 1998. ISBN 978-0943396613.

A. Meurer, C. P. Smith, M. Paprocki, O. Čertík, S. B. Kirpichev, M. Rocklin, A. Kumar, S. Ivanov, J. K. Moore, S. Singh, T. Rathnayake, S. Vig, B. E. Granger, R. P. Muller, F. Bonazzi, H. Gupta, S. Vats, F. Johansson, F. Pedregosa, Curry M. J., Terrel A. R., Š. Roučka, A. Saboo, I. Fernando, S. Kulal, R. Cimrman, and A. Scopatz. SymPy: symbolic computing in Python. *PeerJ Computer Science*, 3, 2017. doi:10.7717/peerj-cs.103.

NASA. Chinese anti-satellite test creates most severe orbital debris cloud in history. Orbital Debris Quarterly News, 11(2):2–3, 2007.

NASA. Satellite collision leaves significant debris clouds. Orbital Debris Quarterly News, 13(2):1–2, 2009.

NASA. Proper implementation of the 1998 NASA breakup model. Orbital Debris Quarterly News, 15(4):4–5, 2011.

NASA. Orbital debris program office. History of on-orbit satellite fragmentations, 15th edition, 2018.

J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer, 2006. ISBN 978-0387303031.

T. E. Oliphant. *A guide to NumPy*, volume 1. Trelgol Publishing USA, 2006.

E. J. Öpik. Collision probabilities with the planets and the distribution of interplanetary matter. *Proceedings of the Royal Irish Academy. Section A: Mathematical and Physical Sciences*, 54:165–199, 1951.

K. B. Petersen and M. S. Pedersen. The matrix cookbook, 2012. http://www.math.uwaterloo.ca/~hwolkowi//matrixcookbook.pdf.

J. M. Picone, A. E. Hedin, D. P. Drob, and A. C. Aikin. NRLMSISE-00 empirical model of the atmosphere: statistical comparisons and scientific issues. *Journal of Geophysical Research: Space Physics*, 107(A12):1–16, 2002. doi:10.1029/2002JA009430.

Python Software Foundation. Python language reference, version 3.x., 2020. https://www.python.org/.

J. Radtke and E. Stoll. Comparing long-term projections of the space debris environment to real world data – looking back to 1990. *Acta Astronautica*, 127:482–490, 2016. doi:10.1016/j.actaastro.2016.06.034.

J. Radtke, S. Müller, V. Schaus, and E. Stoll. LUCA2 - an enhanced long-term utility for collision avoidance. In *Proceedings of the 7th European Conference on Space Debris*, 2017.

C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, 2005. ISBN 978-0262182539.

D. Rees. COSPAR international reference atmosphere: 1986. Part 1: thermosphere models. *Advances in Space Research*, 8(5–6), 1988.

A. Rossi, A. Cordelli, P. Farinella, and L. Anselmo. Collisional evolution of the earth's orbital debris cloud. *Journal of Geophysical Research: Planets*, 99(E11):23195–23210, 1994. doi:10.1029/94JE02320.

A. Rossi, L. Anselmo, Cordelli A., P. Farinella, and C. Pardini. Modelling the evolution of the space debris population. *Planetary and Space Science*, 46(11–12):1583–1596, 1998. doi:10.1016/S0032-0633(98)00070-1.

A. Rossi, L. Anselmo, C. Pardini, R. Jehn, and Valsecchi G. B. The new space debris mitigation (SDM 4.0) long term evolution code. In *Proceedings of the 5th European Conference on Space Debris*, 2009.

L. Sagnieres and I. Sharf. Uncertainty characterization of atmospheric density models for orbit prediction of space debris. In *Proceedings of the 7th European Conference on Space Debris*, 2017.

H. Schaub, S. R. Vadali, J. L. Junkins, and K. T. Alfriend. Spacecraft formation flying control using mean orbit elements. *Journal of the Astronautical Sciences*, 48(1):69–87, 2000.

L. F. Shampine and M. W. Reichelt. The MATLAB ODE suite. *SIAM Journal on Scientific Computing*, 18(1):1–22, 1997. doi:10.1137/S1064827594276424.

R. K. Sharma. Contraction of high eccentricity satellite orbits using KS elements in an oblate atmosphere. *Advances in Space Research*, 23(4): 693–698, 1999. doi:10.1016/S0273-1177(99)00139-8.

N. N. Smirnov. *Space Debris: Hazard Evaluation and Mitigation*. Taylor and Francis, 2002. ISBN 978-0415279079.

N. N. Smirnov, V. R. Dushin, I. I. Panfilov, and V. V. Lebedev. Space debris evolution mathematical modelling. In *Proceedings of 1st European Conference on Space Debris*, 1993.

N. N. Smirnov, A. I. Nazarenko, and A. B. Kiselev. Modelling of the space debris evolution based on continua mechanics. In *Proceedings of 3rd European Conference on Space Debris*, 2001.

G. L. Somma, H. G. Lewis, and C. Colombo. Sensitivity analysis of launch activities in low earth orbit. *Acta Astronautica*, 158:129–139, 2019. doi:10.1016/j.actaastro.2018.05.043.

T. T. Soong. *Fundamentals of Probability and Statistics for Engineers*. John Wiley & Sons, Ltd., 2004. ISBN 978-0470868133.

M. E. Sorge and D. L. Mains. IMPACT fragmentation model developments. *Acta Astronautica*, 126:40–46, 2016. doi:10.1016/j.actaastro.2016.04.011.

Space-Track. Historical TLE search provides orbital states of spacecraft and other objects, 2020. https://www.space-track.org/.

S.-Y. Su and D. J. Kessler. Contribution of explosion and future collision fragments to the orbital debris environment. *Advances in Space Research*, 5(2):25–34, 1985. doi:10.1016/0273-1177(85)90384-9.

Y. Sun and M. Kumar. A Markov chain Monte Carlo particle solution of the initial uncertainty propagation problem. In *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, 2012. doi:10.2514/6.2012-4937.

M. V. Sykes. Zodiacal dust bands: their relation to asteroid families. *Icarus*, 85(2):267–289, 1990. doi:10.1016/0019-1035(90)90117-R.

D. L. Talent. Analytic model for orbital debris environmental management. *Journal of Spacecraft and Rockets*, 29(4):508–513, 1992. doi:10.2514/3.25493.

C. Tardioli, M. Kubicek, M. Vasile, E. Minisci, and A. Riccardi. Comparison of non-intrusive approaches to uncertainty propagation in orbital mechanics. In *Proceedings of the 2015 AAS/AIAA Astrodynamics Specialist Conference*, 2015.

G. Terejanu, P. Singla, T. Singh, and Scott P. D. Uncertainty propagation for nonlinear dynamic systems using gaussian mixture models. *Journal of Guidance, Control, and Dynamics*, 31(6):1623–1633, 2008. doi:10.2514/1.36247.

M. Trisolini and C. Colombo. A density-based approach to the propagation of re-entry uncertainties. In *Proceedings of the 29th AAS/AIAA Space Flight Mechanics Meeting*, 2019.

UCS. Satellite database, 2020. https://www.ucsusa.org/resources/satellite-database.

UN. Office for Outer Space Affairs. Space Debris Mitigation Guidelines of the Committee on the Peaceful Uses of Outer Space, 2010.

S. Valk, A. Lemaître, and L. Anselmo. Analytical and semi-analytical investigations of geosynchronous space debris with high area-to-mass ratios. *Advances in Space Research*, 41(7):1077–1090, 2008. doi:10.1016/j.asr.2007.10.025.

D. A. Vallado. *Fundamentals of Astrodynamics and Applications*. Microcosm Press, 4th edition, 2013. ISBN 978-1881883180.

D. A. Vallado and S. Alfano. Updated analytical partials for covariance transformations and optimization. In *Proceedings of the 2015 AAS/AIAA Astrodynamics Specialist Conference*, 2015.

D. A. Vallado and D. Finkleman. A critical assessment of satellite drag and atmospheric density modeling. *Acta Astronautica*, 95:141–165, 2014. doi:10.1016/j.actaastro.2013.10.005.

D. van Heesch. Doxygen: generate documentation from source code, 2018. http://www.doxygen.nl/.

N. X. Vinh, J. M. Longuski, A. Busemann, and R. D. Culp. Analytic theory of orbit contraction due to atmospheric drag. *Acta Astronautica*, 6(5–6): 697–723, 1979. doi:10.1016/0094-5765(79)90028-6.

P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. Jarrod, M., N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and Contributors. SciPy 1.0: Fundamental algorithms for scientific computing in Python. *Nature Methods*, 17:261–272, 2020. doi:10.1038/s41592-019-0686-2.

K. Vishwajeet, P. Singla, and M. Jah. Nonlinear uncertainty propagation for perturbed two-body orbits. *Journal of Guidance, Control, and Dynamics*, 37(5):1415–1425, 2014. doi:10.2514/1.G000472.

V. Vittaldev, R. P. Russell, and R. Linares. Spacecraft uncertainty propagation using gaussian mixture models and polynomial chaos expansions. *Journal of Guidance, Control, and Dynamics*, 39(12):2615–2626, 2016. doi:10.2514/1.G001571.

R. Walker, C. E. Martin, P. H. Stokes, J. E. Wilkinson, and H. Klinkrad. Analysis of the effectiveness of space debris mitigation measures using the DELTA model. *Advances in Space Research*, 28(9):1437–1445, 2001. doi:10.1016/S0273-1177(01)00445-8.

X.-w. Wang and J. Liu. An introduction to a new space debris evolution model: SOLEM. *Advances in Astronomy*, 2019:1–11, 2019. doi:10.1155/2019/2738276.

R. M. Weisman, M. Majji, and K. T. Alfriend. Application of the transformation of variables technique for uncertainty mapping in nonlinear filtering. *Celestial Mechanics and Dynamical Astronomy*, 118:129–164, 2014. doi:10.1007/s10569-013-9527-2.

N. Wiener. The homogeneous chaos. *American Journal of Mathematics*, 60 (4):897–936, 1938. doi:10.2307/2371268.

W. E. Wiesel. *Spaceflight Dynamics*. CreateSpace Independent Publishing Platform, 3rd edition, 2010. ISBN 978-1452879598.

A. Wittig, P. Di Lizia, R. Armellin, K. Makino, F. Bernelli-Zazzera, and M. Berz. Propagation of large uncertainty sets in orbital dynamics by automatic domain splitting. *Celestial Mechanics and Dynamical Astronomy*, 122:239–261, 2015. doi:10.1007/s10569-015-9618-3.

A. Wittig, C. Colombo, and R. Armellin. Long-term density evolution through semi-analytical and differential algebra techniques. *Celestial Mechanics and Dynamical Astronomy*, 128:435–452, 2017. doi:10.1007/s10569-017-9756-x.

M. Xavier James Raj and R. K. Sharma. Analytical orbit predictions with air drag using KS uniformly regular canonical elements. *Planetary and Space Science*, 54(3):310–316, 2006. doi:10.1016/j.pss.2005.12.006.

D. Xiu and G. E. Karniadakis. The Wiener–Askey polynomial chaos for stochastic differential equations. *SIAM Journal on Scientific Computing*, 24(2):619–644, 2002. doi:10.1137/S1064827501387826.

A Examples

A short collection of examples is given here to improve the understanding of the mathematical concepts applied throughout the thesis by way of illustration.

A.1 Solving the continuity equation

The example for the comparison of the method of characteristics and density mapping was inspired by Halder and Bhattacharya (2011) and Weisman et al. (2014). The initial value problem

$$\dot{x} = F(x) = -x^2 \tag{A.1a}$$
$$x_0 = x(t_0 = 0) \tag{A.1b}$$

has the following solution

$$x = \varphi(x_0, t) = \frac{x_0}{x_0 t + 1} \tag{A.2}$$

dropping the dependence on t_0 without loss of generality. Solving for x_0 results in

$$x_0 = \varphi^{-1}(x, t) = \frac{x}{1 - xt} \tag{A.3}$$

In the absence of sources or sink terms, there are different ways of solving the continuity equation.

A.1.1 Method of characteristics

Evaluating the integral of Equation (1.13) using Equation (A.1a) and (A.2) gives

$$\int_0^t \frac{\partial F}{\partial x}\,\mathrm{d}t' = -2\int_0^t x\,\mathrm{d}t' \tag{A.4a}$$

$$= -2\int_0^t \frac{x_0}{x_0 t + 1}\,\mathrm{d}t' \tag{A.4b}$$

$$= -2\ln(x_0 t + 1) \tag{A.4c}$$

Plugging this into Equation (1.13) using Equation (A.3) results in

$$n(x,t) = n(x_0)(1 + x_0 t)^2 \tag{A.5a}$$

$$= n\left(\frac{x}{1 - xt}\right)\frac{1}{(1 - xt)^2} \tag{A.5b}$$

A.1.2 Density mapping

The same result can also be found through mapping of the density. Evaluating the determinate of the Jacobian in Equation (1.15) gives

$$\det J = \frac{1}{(1 + x_0 t)^2} = (1 - xt)^2 \tag{A.6}$$

Hence, Equation (1.15) for the given sample dynamics results in

$$n(x,t) = n\left(\frac{x}{1 - xt}\right)\frac{1}{(1 - xt)^2} \tag{A.7}$$

which is equivalent to the solution obtained through the method of characteristics.

A.2 Application of the Dirac generalised function

Here, an example is given transforming the probability function in $\boldsymbol{v} = (v_x\, v_y\, v_z)^T \in \mathbb{R}^3$ at $\boldsymbol{r} = (r_x\, r_y\, r_z)^T$ to (a, e). From Equation (2.13) it follows that

$$p_{a,e} = \int_{\mathbb{R}^3} p_{\boldsymbol{v}}\; \delta[\varphi_a(v_x, v_y, v_z) - a]\; \delta[\varphi_e(v_x, v_y, v_z) - e]\,\mathrm{d}v_x\,\mathrm{d}v_y\,\mathrm{d}v_z \tag{A.8}$$

The roots in, say, v_x and v_y, given $\boldsymbol{r}, a, e$ and v_z are found using the following relationships (Battin, 1999; Vallado, 2013, p. 501 and p. 27, respectively)

$$\boldsymbol{r} \cdot \boldsymbol{v} = \frac{\mu}{\hbar} re \sin f = c \tag{A.9a}$$

$$v^2 = v_x^2 + v_y^2 + v_z^3 = \frac{2\mu}{r} - \frac{\mu}{a} \tag{A.9b}$$

with the gravitational parameter, μ, and arbitrarily introduced constant, c, and the specific angular momentum, $\hbar$, that can be calculated via the semi-parameter, p, as

$$p = a(1 - e^2) \tag{A.10a}$$
$$\hbar = \sqrt{\mu p} \tag{A.10b}$$

Inserting Equation (A.9b) in the squared form of Equation (A.9a) allows to find a quadratic expression in, say, v_y, e.g.

$$\left(r_x^2 + r_y^2\right) v_y^2 + \left(2r_y(r_z v_z - c)\right) v_y + (c - r_z v_z)^2 + r_x^2(v_z^2 - v^2) = 0 \tag{A.11}$$

Solving for $v_y^{(1)}$ and $v_y^{(2)}$ and inserting it into either of the Equation (A.9b) or Equation (A.9a) to solve for $v_x^{(k,1)}$ and $v_x^{(k,2)}$ gives the 4 roots of the system. Special care needs to be taken for cases where r_x vanishes, which in the $\mathcal{T}$-frame occurs for circular orbits or around the apsides. Then, Equation (A.9a) becomes decoupled from v_x, and v_y can be directly inferred.

The derivatives of a and e with respect to v_x and v_y are straightforward to find. Rearranging Equation (A.9b), the following derivatives of a can be found

$$\frac{\partial a}{\partial v_j} = \frac{2a^2 v_j}{\mu} \qquad j = 1, 2, 3 \tag{A.12}$$

Instead, for the derivative of e, the following formula is introduced (Vallado, 2013, p. 98)

$$\boldsymbol{e} = \left(\frac{v^2}{\mu} - \frac{1}{r}\right)\boldsymbol{r} - \frac{1}{\mu}(\boldsymbol{r} \cdot \boldsymbol{v})\boldsymbol{v} \tag{A.13}$$

to find

$$\frac{\partial e}{\partial v_j} = \frac{1}{2e}\frac{\partial \boldsymbol{e}^T \boldsymbol{e}}{\partial v_j} = \frac{\boldsymbol{e}^T}{e}\frac{\partial \boldsymbol{e}}{\partial v_j} \tag{A.14}$$

where

$$\frac{\partial e_l}{\partial v_j} = \frac{1}{\mu}\begin{cases} v_j r_j - \boldsymbol{r} \cdot \boldsymbol{v}, & \text{for } j = l \\ 2v_j r_l - r_j v_l, & \text{otherwise} \end{cases} \qquad j, l = 1, 2, 3 \tag{A.15}$$

Finally, putting everything together, the distribution $p_{a,e}$ is found as

$$p_{a,e} = \int_{\mathbb{R}} \sum_{k=1}^{2} \sum_{m=1}^{2} \frac{1}{\left|\det \boldsymbol{J}\left(v_x^{(k,m)}, v_y^{(k)}\right)\right|} p_{\boldsymbol{v}}\left(v_x^{(k,m)}, v_y^{(k)}, v_z\right) \mathrm{d}v_z \tag{A.16}$$

where

$$\boldsymbol{J}\left(v_x^{(k,m)}, v_y^{(k)}\right) = \left.\begin{pmatrix} \frac{\partial a}{\partial v_x} & \frac{\partial a}{\partial v_y} \\ \frac{\partial e}{\partial v_x} & \frac{\partial e}{\partial v_y} \end{pmatrix}\right|_{\substack{v_x = v_x^{(k,m)} \\ v_y = v_y^{(k)}}} \tag{A.17}$$

The same result could be achieved by integration of $p_{a,e,i}$ over i. However, the presented method is more generally applicable, even if the joint distribution in $p_{a,e,i}$ was not available. Furthermore, it can be easily converted into the v-space, which improves the convergence speed of numerical integration.

A.3 Simplified orbital cloud evolution

The example given here is adapted from McInnes (1993) and shows the evolution of a cloud of fragments in circular orbit, subject to drag only. The drag force, a_d, acting in the direction of the orbital velocity, v, is (Vallado, 2013)

$$a_d = -\frac{1}{2} B \rho v^2 \tag{A.18}$$

with the ballistic coefficient, B and the atmospheric density, ρ. This formula is only valid for a non-rotating atmosphere.

Following Wiesel (2010, Section 3.6), the calculation of drag effect on the trajectory evolution is performed via the specific energy, $\mathcal{E}$, given as

$$\mathcal{E} = -\frac{\mu}{2a} \tag{A.19}$$

where a is the semi-major axis and μ is the gravitational parameter. Taking the derivative of $\mathcal{E}$ with respect to time, t, gives

$$\frac{\mathrm{d}\mathcal{E}}{\mathrm{d}t} = \frac{\mu}{2a^2} \frac{\mathrm{d}a}{\mathrm{d}t} \tag{A.20}$$

Air drag induces a dissipative force decreasing $\mathcal{E}$ over time such that

$$\frac{\mathrm{d}\mathcal{E}}{\mathrm{d}t} = a_d v \tag{A.21}$$

Substitution of Equations (A.19) to (A.21) gives the dynamics in a as

$$\frac{\mathrm{d}a}{\mathrm{d}t} = F_a = -B \rho v^3 \frac{a^2}{\mu} \tag{A.22}$$

Assuming circular orbits, e.g. $a = r$, with the orbit radius, r, and the circular orbital velocity, v_r, as

$$v = \sqrt{\frac{\mu}{r}} \tag{A.23}$$

the dynamics in r, are found to be

$$\frac{\mathrm{d}r}{\mathrm{d}t} = F_r = -B \rho \sqrt{\mu r} \tag{A.24}$$

Considering a simplified atmospheric density model

$$\rho = \rho_0 \exp\left(-\frac{r - R_E}{H}\right) \tag{A.25}$$

with the base density, ρ_0 at the radius of Earth, R_E, and the scale height, H, the dynamics can be written as

$$F_r = -B\rho_0\sqrt{\mu r}\exp\left(-\frac{r - R_E}{H}\right) \tag{A.26}$$

The partial derivative of F_r with respect to r is

$$J_r = \frac{\partial F_r}{\partial r} = \left(\frac{1}{2r} - \frac{1}{H}\right) F_r \tag{A.27}$$

Hence, the characteristics can be found through application of Equation (3.8) as

$$\frac{\mathrm{d}n}{\mathrm{d}t} = -nJ_r \tag{A.28a}$$

$$\frac{\mathrm{d}r}{\mathrm{d}t} = F_r \tag{A.28b}$$

Using a further approximation of $\sqrt{R_E} \approx \sqrt{r}$, which is accurate to 1% for $h < 1000$ km, a closed form solution of Equation (A.28) can be found (McInnes, 1993). Lumping the constant factors together in k

$$k = B\rho_0\sqrt{\mu R_E}\exp\left(\frac{R_E}{H}\right) \tag{A.29}$$

the dynamics can be approximated as

$$\frac{\mathrm{d}r}{\mathrm{d}t} \approx -k\exp\left(-\frac{r}{H}\right) \tag{A.30}$$

permitting – using separation of variables – the solution

$$r = H\ln\left[\exp\left(\frac{r_0}{H}\right) - \frac{k}{H}t\right] \tag{A.31}$$

with initial position, $r_0 = r_0(t_0 = 0)$. Or, solved for r_0

$$r_0 = H\ln\left[\exp\left(\frac{r}{H}\right) + \frac{k}{H}t\right] \tag{A.32}$$

The density can also be found with a closed form solution. Integration of Equation (A.28a) with separation of variables results in

$$n = n_0\exp\left(-\int_0^t J_r\,\mathrm{d}t\right) \tag{A.33}$$

with the integral, using substitutions from Equations (A.27), (A.30) and (A.31), resulting in

$$-\int_0^t J_r \, \mathrm{d}t = \frac{r - r_0}{H} + \frac{1}{2} \ln\left(\frac{r_0}{r}\right) \tag{A.34}$$

Both, the closed form and the integrated solutions are depicted in Figure 3.1, for an initial distribution of

$$n_{r,0} = N_0 p_{r_0}(r_0) \qquad r_0 \sim \mathcal{N}(\mu_r = R_E + 400 \text{ km}, \sigma = 50^2 \text{ km}^2) \tag{A.35}$$

and the parameters $N_0 = 10^3$, $\mu = 398600.4\text{km}^3/\text{s}^2$, $R_E = 6371$ km, $B = 0.021$ m^2/kg, $\rho_0 = 2.765 \times 10^{-5}$ kg/m^3 and $H = 25.284$ km.

B Derivations

In the following, the derivations to obtain the orbits intersecting with a given point in space (Appendix B.1), adding directionality to an isotropic delta velocity distribution (Appendix B.2), the King-Hele (KH) contraction rates and Jacobian of the dynamics (Appendices B.3 and B.4), and the Jacobian and the Hessian of the cost function with respect to its parameters (Appendix B.5) are described in detail.

B.1 Intersecting orbits

Given a point in space, $\boldsymbol{r}$, that needs to be intersected, puts a limit on the Keplerian elements. If the point itself is described through Keplerian elements, the radius, $r = ||\boldsymbol{r}||$, can be found via

$$r = \frac{p}{1 + e\cos f} \tag{B.1}$$

the latitude, $\theta \in [-\pi/2, \pi/2]$, via

$$\sin\theta = \sin i \sin u \tag{B.2}$$

with the argument of latitude, u, defined as

$$u = \omega + f \tag{B.3}$$

and the longitude, $\lambda \in [0, 2\pi)$, is found via

$$\sin(\lambda - \Omega) = \frac{\tan\theta}{\tan i} \qquad \cos(\lambda - \Omega) = \frac{\cos u}{\cos\theta} \tag{B.4}$$

using any two-argument arctangent routine. The point is then defined as

$$\boldsymbol{r} = \begin{pmatrix} r\cos\theta\cos\lambda \\ r\cos\theta\sin\lambda \\ r\sin\theta \end{pmatrix} \tag{B.5}$$

All the orbits that cross $\boldsymbol{r}$ can be found with simple geometrical considerations.

B.1.1 Orbital planes

First, note that the inclination, i, needs to be larger than the latitude of the point to be evaluated, resulting in the following permissible range

$$i \in [i_0 = |\theta|, i_1 = \pi - |\theta|] \tag{B.6}$$

The node, Ω, is a function of λ, θ and i. From spherical trigonometry, the following dependence can be derived

$$\sin(\lambda - \Omega) = \frac{\tan\theta}{\tan i} \tag{B.7}$$

For each i, there exists another orbital plane at

$$\Omega' = \Omega(\lambda, \theta, \pi - i) + \pi \tag{B.8}$$

for ascending/descending crossing of $\boldsymbol{r}$. If the point is located over the poles, Ω is ill-defined.

B.1.2 Orbit geometries

The perigee radius, r_p, is limited from below by the radius of Earth, R_E, and from above by r. The apogee radius, r_a, needs to be larger than r. These limits

$$R_E \leq r_p = a(1-e) \leq r \leq r_a = a(1+e) \tag{B.9}$$

translate into limits on the semi-major axis, a, and the eccentricity, e, as

$$a \in [a_0 = \frac{R_E + r}{2}, a_1 = \infty) \tag{B.10a}$$

$$e \in [e_0 = |1 - \frac{r}{a}|, e_1 = 1 - \frac{R_E}{a}) \tag{B.10b}$$

Fixing a and e imposes the following rule on the true anomaly, f

$$\cos f = \frac{1}{e}\left(\frac{p}{r} - 1\right) \tag{B.11}$$

Due to symmetry, another $f' = 2\pi - f$ exists. To find the argument of perigee, ω, the argument of latitude is required, which can be obtained unambiguously through

$$\sin u = \frac{\sin\theta}{\sin i} \qquad \cos u = \cos(\lambda - \Omega)\cos\theta \tag{B.12}$$

To intersect with r, the orbit needs to be rotated into the right angle with

$$\omega = u - f \tag{B.13}$$

B.2 Introducing directionality

The dependence on the directional component is considered via the directional distribution, $p_{\phi,\varphi|\Delta v}$, with the spherical angles

$$\phi \in [0, 2\pi) \qquad \varphi \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right] \tag{B.14}$$

defined relative to the $\mathcal{T}$-frame. In general, the theory introduced in this thesis is applicable to any directional distribution. However, the NASA SBM does not contain any information about the impulse direction. Hence, to find $p_{\phi,\varphi|\Delta v}$, an isotropic impulse direction is assumed. Each infinitely small area, $\mathrm{d}S$, on the sphere defined by the radius, Δv, is equally likely, hence

$$p_S = \frac{1}{4\pi(\Delta v)^2} \tag{B.15}$$

Using $\mathrm{d}S = (\Delta v)^2 \cos\varphi \,\mathrm{d}\phi \,\mathrm{d}\varphi$ and application of Equation (2.9), gives

$$p_{\phi,\varphi|\Delta v} \,\mathrm{d}\phi \,\mathrm{d}\varphi = p_S \,\mathrm{d}S = \frac{\cos\varphi}{4\pi} \,\mathrm{d}\phi \,\mathrm{d}\varphi \tag{B.16}$$

from which it follows that

$$p_{\phi,\varphi|\Delta v} = p_{\phi,\varphi} = \frac{\cos\varphi}{4\pi} \tag{B.17}$$

The marginals

$$p_\phi = \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} p_{\phi,\varphi} \,\mathrm{d}\varphi = \frac{1}{2\pi} \tag{B.18a}$$

$$p_\varphi = \int_0^{2\pi} p_{\phi,\varphi} \,\mathrm{d}\phi = \frac{\cos\varphi}{2} \tag{B.18b}$$

show that only ϕ is uniformly distributed over $\phi \in [0, 2\pi)$, but φ is not uniformly distributed. To sample φ, the cumulative distribution function, P_φ, given as

$$P_\varphi = \int_{-\frac{\pi}{2}}^{\varphi} p_\varphi(\varphi') \,\mathrm{d}\varphi' = \frac{1}{2}\left(\sin\varphi + 1\right) \tag{B.19}$$

needs to be solved for φ, such that

$$\sin\varphi = 2P_\varphi - 1 \tag{B.20}$$

From Equation (B.17), the distribution in spherical velocity coordinates can be calculated as

$$p_{\Delta v,\phi,\varphi} = p_{\phi,\varphi|\Delta v} p_{\Delta v} = \frac{\cos\varphi}{4\pi} p_{\Delta v} \tag{B.21}$$

Transformation of $p_{\Delta v,\phi,\varphi}$ into Cartesian velocity coordinates in the $\mathcal{T}$-frame, given as

$$\Delta v_t = \Delta v \cos\varphi \cos\phi \tag{B.22a}$$
$$\Delta v_n = \Delta v \cos\varphi \sin\phi \tag{B.22b}$$
$$\Delta v_h = \Delta v \sin\varphi \tag{B.22c}$$

and, using Equation (2.10), results in the directional distribution

$$p_{\Delta \boldsymbol{v}} = \frac{p_{\Delta v,\phi,\varphi}}{(\Delta v^2)\cos\varphi} = \frac{p_{\Delta v}}{4\pi(\Delta v)^2} \tag{B.23}$$

B.3 Averaged drag variational equations

Following the derivations of King-Hele (1964), the perturbing effect of the drag is considered using the Gauss planetary equations in Keplerian elements (Battin, 1999, Chapter 10.3). Only the tangential acceleration induced by the aerodynamic drag is considered, i.e.

$$a_d = -\frac{1}{2} B \rho v^2 \tag{B.24}$$

with the density, ρ, the orbital velocity, v, and the ballistic coefficient, B, defined as

$$B = c_D \frac{A}{m} \tag{B.25}$$

and the drag coefficient, c_D, and area-to-mass ratio, A/m. Atmospheric rotation is ignored here but could be taken into account by multiplying the right-hand side of Equation (B.24) with the appropriate factor.

The Gauss' variational equations in semi-major axis, a, and eccentricity, e, considering the acceleration in Equation (B.24) can be written as

$$\frac{\mathrm{d}a}{\mathrm{d}t} = -\frac{a^2 v^3}{\mu} B\rho \tag{B.26a}$$
$$\frac{\mathrm{d}e}{\mathrm{d}t} = -v(e + \cos f) B\rho \tag{B.26b}$$

with Earth's gravitational parameter, μ, and the true anomaly, f. Using the identities for the radius, r, velocity, v, and $\cos f$ given as

$$r = a(1 - e\cos E) \tag{B.27a}$$
$$v = \sqrt{2\frac{\mu}{r} - \frac{\mu}{a}} \tag{B.27b}$$
$$\cos f = \frac{\cos E - e}{1 - e\cos E} \tag{B.27c}$$

the variational equations, given in Equation (B.26), can be expressed as a function a, e and the eccentric anomaly, E, as

$$\frac{\mathrm{d}a}{\mathrm{d}t} = -\sqrt{\mu a} B \rho \left(\frac{1 + e\cos E}{1 - e\cos E}\right)^{\frac{3}{2}} \tag{B.28a}$$

$$\frac{\mathrm{d}e}{\mathrm{d}t} = -\sqrt{\frac{\mu}{a}} B \rho \frac{(1 + e\cos E)^{\frac{1}{2}}}{(1 - e\cos E)^{\frac{3}{2}}} (1 - e^2)\cos E \tag{B.28b}$$

Given Kepler's equation

$$M = E - e\sin E \tag{B.29}$$

it follows that

$$\mathrm{d}M = (1 - e\cos E)\,\mathrm{d}E \tag{B.30}$$

Hence, the averaging of the variational equations from Equation (1.31) can be given in terms of E as

$$\frac{\mathrm{d}\bar{\boldsymbol{\alpha}}}{\mathrm{d}t} = \frac{1}{2\pi}\int_0^{2\pi} \left(\frac{\mathrm{d}\boldsymbol{\alpha}}{\mathrm{d}t}\right)(1 - e\cos E)\,\mathrm{d}E \tag{B.31}$$

Finally, plugging Equation (B.28) into Equation (B.31) results in

$$F_a = \frac{\mathrm{d}\bar{a}}{\mathrm{d}t} = -\frac{\sqrt{\mu a} B}{2\pi}\int_0^{2\pi} \rho(h) \frac{(1 + e\cos E)^{\frac{3}{2}}}{(1 - e\cos E)^{\frac{1}{2}}}\,\mathrm{d}E \tag{B.32a}$$

$$F_e = \frac{\mathrm{d}\bar{e}}{\mathrm{d}t} = -\sqrt{\frac{\mu}{a}}\frac{B}{2\pi}\int_0^{2\pi} \rho(h) \left(\frac{1 + e\cos E}{1 - e\cos E}\right)^{\frac{1}{2}} (1 - e^2)\cos E\,\mathrm{d}E \tag{B.32b}$$

The integrals in Equation (B.32) can be solved analytically if the atmosphere model describes strict exponential decay as in Equation (3.15). Rewriting the altitude, h, as

$$\begin{aligned} h &= a(1 - e\cos E) - R_E \\ &= a - ae\cos E - R_E \\ &= a - ae + ae - ae\cos E - R_E \\ &= a(1 - e) - R_E + ae(1 - \cos E) \\ &= h_p + ae(1 - \cos E) \end{aligned}$$

permits to rewrite the density as

$$\rho(h) = \rho_0 \exp\left(-\frac{h - h_0}{H}\right) \tag{B.34a}$$

$$= \rho_0 \exp\left(-\frac{h_p - h_0}{H}\right)\exp\left(-\frac{ae}{H}\right)\exp\left(\frac{ae}{H}\cos E\right) \tag{B.34b}$$

$$= \rho(h_p)\exp(-z)\exp(z\cos E) \tag{B.34c}$$

with the auxiliary variable, z, defined as

$$z = \frac{ae}{H} \tag{B.35}$$

With the definition of ρ from Equation (B.34c) and exploiting the symmetry of the integrand around E, the averaged dynamics from Equation (B.32) become

$$F_a = -\frac{k_a}{\pi} \exp(-z) \int_0^\pi \exp(z \cos E)\ d_a(E)\, \mathrm{d}E \tag{B.36a}$$

$$F_e = -\frac{k_e}{\pi} \exp(-z) \int_0^\pi \exp(z \cos E)\ d_e(E)\, \mathrm{d}E \tag{B.36b}$$

with

$$k_a = \sqrt{\mu a} B \rho(h_p) \tag{B.37a}$$

$$k_e = k_a / a \tag{B.37b}$$

$$d_a(E) = \frac{(1 + e\cos E)^{\frac{3}{2}}}{(1 - e\cos E)^{\frac{1}{2}}} \tag{B.37c}$$

$$d_e(E) = \left(\frac{1 + e\cos E}{1 - e\cos E}\right)^{\frac{1}{2}} (1 - e^2)\cos E \tag{B.37d}$$

The closed-form approximation of Equation (B.36) can be found for three different eccentricity regimes. King-Hele (1964) splits the derivations into four different regimes of which one is omitted here. The regimes are circular orbits, low eccentric and highly eccentric orbits. He uses a fixed boundary condition $e_b = 0.2$ to switch between highly and low eccentric orbits.

B.3.1 Circular orbits

For circular orbits, i.e. $e = 0$, the terms simplify to

$$\exp(-z) = 1 \qquad \exp(-z\cos E) = 1 \qquad d_a = 1 \qquad d_e = \cos E$$

Hence, an exact solution of Equation (B.36) is found as

$$F_a = -k_a \tag{B.38a}$$

$$F_e = 0 \tag{B.38b}$$

B.3.2 Low eccentric orbits

For low eccentric orbits, i.e. $e < e_b$, the terms d_a and d_e can be expanded as a power series in e, e.g. up to second order

$$d_a = 1 + 2\cos(E)e + \frac{3}{2}\cos^2(E)e^2 + \mathcal{O}(e^3) \tag{B.39a}$$

$$d_e = \cos(E) + \cos^2(E)e + \left(\cos^{\frac{3}{2}}(E) - \cos(E)\right)e^2 + \mathcal{O}(e^3) \tag{B.39b}$$

which, using trigonometric identities and dropping the error, $\mathcal{O}(e^3)$, results in

$$d_a \approx 1 + 2\cos(E)e + \frac{3}{4}(1+\cos(2E))e^2 \tag{B.40a}$$

$$d_e \approx \cos(E) + \frac{1}{2}(1+\cos(2E))e + \frac{1}{8}\left(\cos(3E) - 5\cos(E)\right)e^2 \tag{B.40b}$$

After plugging the series expansion into Equation (B.36), the structure of the individual terms in the integral follows the modified Bessel function of the first kind, I_n, which for $n \in \mathbb{N}_0$ is given as (Abramowitz and Stegun, 1972, p. 376)

$$I_n(z) = \frac{1}{\pi}\int_0^\pi \exp(z\cos\theta)\cos(n\theta)d\theta \tag{B.41}$$

Hence, the averaged dynamics for low eccentric orbits can be calculated as

$$F_a = -k_a\exp(-z)\left[I_0 + 2I_1 e + \frac{3}{4}(I_0+I_2)e^2 + \mathcal{O}(e^3)\right] \tag{B.42a}$$

$$F_e = -k_e\exp(-z)\left[I_1 + \frac{1}{2}(I_0+I_2)e + \frac{1}{8}(I_3-5I_1)e^2 + \mathcal{O}(e^3)\right] \tag{B.42b}$$

Equivalently, it can be extended into higher orders, such that

$$F_a = -k_a\exp(-z)\left[\boldsymbol{e}^T\boldsymbol{K}_a^l\boldsymbol{I} + \mathcal{O}(e^6)\right] \tag{B.43a}$$

$$F_e = -k_e\exp(-z)\left[\boldsymbol{e}^T\boldsymbol{K}_e^l\boldsymbol{I} + \mathcal{O}(e^6)\right] \tag{B.43b}$$

with

$$\boldsymbol{e}^T = \begin{pmatrix} 1 & e & e^2 & e^3 & e^4 & e^5 \end{pmatrix}$$

$$\boldsymbol{I}^T = \begin{pmatrix} I_0 & I_1 & I_2 & I_3 & I_4 & I_5 & I_6 \end{pmatrix}$$

and

$$\boldsymbol{K}_a^l = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 & 0 \\ \frac{3}{4} & 0 & \frac{3}{4} & 0 & 0 & 0 & 0 \\ 0 & \frac{3}{4} & 0 & \frac{1}{4} & 0 & 0 & 0 \\ \frac{21}{64} & 0 & \frac{28}{64} & 0 & \frac{7}{64} & 0 & 0 \\ 0 & \frac{30}{64} & 0 & \frac{15}{64} & 0 & \frac{3}{64} & 0 \end{bmatrix} \tag{B.45a}$$

$$\boldsymbol{K}_e^l = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 & 0 & 0 & 0 \\ 0 & -\frac{5}{8} & 0 & \frac{1}{8} & 0 & 0 & 0 \\ -\frac{5}{16} & 0 & -\frac{4}{16} & 0 & \frac{1}{16} & 0 & 0 \\ 0 & -\frac{18}{128} & 0 & -\frac{1}{128} & 0 & \frac{3}{128} & 0 \\ -\frac{18}{256} & 0 & -\frac{19}{256} & 0 & \frac{2}{256} & 0 & \frac{3}{256} \end{bmatrix} \tag{B.45b}$$

B.3.3 Highly eccentric orbits

To find an analytical expression of the integral for highly eccentric orbits, King-Hele (1964) performs a substitution via

$$\cos E = 1 - \frac{\lambda^2}{z} \tag{B.46}$$

such that

$$\mathrm{d}E = \sqrt{\frac{2}{z\left(1-\frac{\lambda}{2z}\right)}}\,\mathrm{d}\lambda \tag{B.47}$$

Plugging Equations (B.46) and (B.47) into Equation (B.36) gives

$$F_a = -\frac{k_a}{\pi}\sqrt{\frac{2}{z}}\int_0^{\sqrt{2z}} \exp\left(-\lambda^2\right)\hat{d}_a(\lambda)\,\mathrm{d}\lambda \tag{B.48a}$$

$$F_e = -\frac{k_e}{\pi}\sqrt{\frac{2}{z}}(1-e^2)\int_0^{\sqrt{2z}} \exp\left(-\lambda^2\right)\hat{d}_e(\lambda)\,\mathrm{d}\lambda \tag{B.48b}$$

with

$$\hat{d}_a(\lambda) = \frac{\left(1+e\left(1-\frac{\lambda^2}{z}\right)\right)^{\frac{3}{2}}}{\left(1-e\left(1-\frac{\lambda^2}{z}\right)\right)^{\frac{1}{2}}}\frac{1}{\left(1-\frac{\lambda^2}{2z}\right)^{\frac{1}{2}}} \tag{B.49a}$$

$$\hat{d}_e(\lambda) = \left(\frac{1+e\left(1-\frac{\lambda^2}{z}\right)}{1-e\left(1-\frac{\lambda^2}{z}\right)}\right)^{\frac{1}{2}}\frac{1-\frac{\lambda^2}{z}}{\left(1-\frac{\lambda^2}{2z}\right)^{\frac{1}{2}}} \tag{B.49b}$$

These two terms can be expanded in λ^2/z, e.g. in second order

$$\hat{d}_a(\lambda) = \frac{(1+e)^{\frac{3}{2}}}{(1-e)^{\frac{1}{2}}}\left(1+\frac{k_{a1}}{(1-e^2)}\frac{\lambda^2}{z}+\frac{k_{a2}}{(1-e^2)^2}\frac{\lambda^4}{z^2}+\mathcal{O}\left(\frac{\lambda^6}{z^3}\right)\right) \tag{B.50a}$$

$$\hat{d}_e(\lambda) = \left(\frac{1+e}{1-e}\right)^{\frac{1}{2}}\left(1+\frac{k_{e1}}{(1-e^2)}\frac{\lambda^2}{z}+\frac{k_{e2}}{(1-e^2)^2}\frac{\lambda^4}{z^2}+\mathcal{O}\left(\frac{\lambda^6}{z^3}\right)\right) \tag{B.50b}$$

with

$$\begin{aligned}
k_{a1} &= \frac{1}{4}\left(1-8e+3e^2\right)\\
k_{a2} &= \frac{1}{32}\left(3-16e+50e^2+16e^3-5e^4\right)\\
k_{e1} &= \frac{1}{4}\left(-3-4e+3e^2\right)\\
k_{e2} &= \frac{1}{32}\left(-5+24e+26e^2+8e^3-5e^4\right)
\end{aligned}$$

Replacing the upper integration limit in Equation (B.48) of $\sqrt{2z}$ with ∞, which introduces small relative errors, and thus noting that the individual integrands can be solved analytically through integration by parts (King-Hele, 1964)

$$\int_0^\infty \lambda^2 \exp\left(-\lambda^2\right) \mathrm{d}\lambda = \frac{1}{2}\int_0^\infty \exp\left(-\lambda^2\right) \mathrm{d}\lambda = \frac{1}{4}\sqrt{\pi} \tag{B.52a}$$

$$\int_0^\infty \lambda^4 \exp\left(-\lambda^2\right) \mathrm{d}\lambda = \frac{3}{8}\sqrt{\pi} \tag{B.52b}$$

the averaged dynamics for highly eccentric orbits can be calculated as

$$F_a = -k_a c_a \left[\frac{1}{2} + \frac{1}{4}\frac{k_{a1}}{z(1-e^2)} + \frac{3}{8}\frac{k_{a2}}{z^2(1-e^2)^2} + \mathcal{O}\left(\frac{1}{z^3}\right)\right] \tag{B.53a}$$

$$F_e = -k_e c_e \left[\frac{1}{2} + \frac{1}{4}\frac{k_{e1}}{z(1-e^2)} + \frac{3}{8}\frac{k_{e2}}{z^2(1-e^2)^2} + \mathcal{O}\left(\frac{1}{z^3}\right)\right] \tag{B.53b}$$

with

$$c_a = \sqrt{\frac{2}{\pi z}}\frac{(1+e)^{\frac{3}{2}}}{(1-e)^{\frac{1}{2}}} \tag{B.54a}$$

$$c_e = \sqrt{\frac{2}{\pi z}}(1-e^2)\left(\frac{1+e}{1-e}\right)^{\frac{1}{2}} \tag{B.54b}$$

As before, the expansion can equivalently be performed considering higher terms such that

$$F_a = -k_a c_a [\boldsymbol{k}^T \boldsymbol{K}_a^h \boldsymbol{e} + \mathcal{O}\left(\frac{1}{z^6}\right)] \tag{B.55a}$$

$$F_e = -k_e c_e [\boldsymbol{k}^T \boldsymbol{K}_e^h \boldsymbol{e} + \mathcal{O}\left(\frac{1}{z^6}\right)] \tag{B.55b}$$

with

$$\boldsymbol{k}^T = \begin{pmatrix} 1 & \frac{1}{z(1-e^2)} & \frac{1}{z^2(1-e^2)^2} & \frac{1}{z^3(1-e^2)^3} & \frac{1}{z^4(1-e^2)^4} & \frac{1}{z^5(1-e^2)^5} \end{pmatrix}$$

$$\boldsymbol{e}^T = \begin{pmatrix} 1 & e & e^2 & e^3 & e^4 & e^5 & e^6 & e^7 & e^8 & e^9 & e^{10} \end{pmatrix}$$

and the constant matrices

$$\left(\boldsymbol{K}_a^h\right)^T = \begin{bmatrix} \frac{1}{2} & \frac{1}{16} & \frac{9}{256} & \frac{75}{2048} & \frac{3675}{65536} & \frac{59535}{524288} \\ 0 & -\frac{1}{2} & -\frac{3}{16} & -\frac{45}{256} & -\frac{525}{2048} & -\frac{33075}{65536} \\ 0 & \frac{3}{16} & \frac{75}{128} & \frac{675}{2048} & \frac{5985}{16384} & \frac{288225}{524288} \\ 0 & 0 & \frac{3}{16} & -\frac{75}{128} & -\frac{105}{2048} & \frac{10395}{16384} \\ 0 & 0 & -\frac{15}{256} & -\frac{3735}{2048} & \frac{21945}{32768} & -\frac{344925}{262144} \\ 0 & 0 & 0 & -\frac{45}{256} & \frac{13545}{2048} & -\frac{129465}{32768} \\ 0 & 0 & 0 & \frac{105}{2048} & \frac{110985}{16384} & -\frac{7687575}{262144} \\ 0 & 0 & 0 & 0 & \frac{525}{2048} & -\frac{836325}{16384} \\ 0 & 0 & 0 & 0 & -\frac{4725}{65536} & -\frac{16288965}{524288} \\ 0 & 0 & 0 & 0 & 0 & -\frac{33075}{65536} \\ 0 & 0 & 0 & 0 & 0 & \frac{72765}{524288} \end{bmatrix}$$

$$\left(\boldsymbol{K}_e^h\right)^T = \begin{bmatrix} \frac{1}{2} & -\frac{3}{16} & -\frac{15}{256} & -\frac{105}{2048} & -\frac{4725}{65536} & -\frac{72765}{524288} \\ 0 & -\frac{1}{4} & \frac{9}{32} & \frac{75}{512} & \frac{735}{4096} & \frac{42525}{131072} \\ 0 & \frac{3}{16} & \frac{39}{128} & -\frac{405}{2048} & \frac{525}{16384} & \frac{152145}{524288} \\ 0 & 0 & \frac{3}{32} & -\frac{375}{256} & \frac{735}{4096} & -\frac{31185}{32768} \\ 0 & 0 & -\frac{15}{256} & -\frac{1515}{2048} & \frac{123585}{32768} & -\frac{530145}{262144} \\ 0 & 0 & 0 & -\frac{45}{512} & \frac{31605}{4096} & -\frac{1165185}{65536} \\ 0 & 0 & 0 & \frac{105}{2048} & \frac{40845}{16384} & -\frac{10235295}{262144} \\ 0 & 0 & 0 & 0 & \frac{525}{4096} & -\frac{1505385}{32768} \\ 0 & 0 & 0 & 0 & -\frac{4725}{65536} & -\frac{5716305}{524288} \\ 0 & 0 & 0 & 0 & 0 & -\frac{33075}{131072} \\ 0 & 0 & 0 & 0 & 0 & \frac{72765}{524288} \end{bmatrix}$$

B.4 Jacobian of drag variational equations

The partial derivatives of the dynamics with respect to a and e are given here for the three different regimes discussed in Appendix B.3. The partial derivatives of a partial atmosphere defined in Equation (3.17) (dropping the

subscript j), and given $h_p = a(1-e) - R_E$, can be found as

$$\frac{\partial \rho(h_p)}{\partial a} = -\frac{1-e}{H}\rho(h_p) \tag{B.57a}$$

$$\frac{\partial \rho(h_p)}{\partial e} = \frac{a}{H}\rho(h_p) \tag{B.57b}$$

Thus, the partial derivatives of k_a and k_e (see Equation (B.37)) with respect to a and e are

$$\frac{\partial k_a}{\partial a} = \delta\sqrt{\mu}\left(\frac{\rho(h_p)}{2\sqrt{a}} + \sqrt{a}\frac{\partial \rho(h_p)}{\partial a}\right) = k_a\left(\frac{1}{2a} - \frac{1-e}{H}\right) \tag{B.58a}$$

$$\frac{\partial k_a}{\partial e} = \delta\sqrt{\mu a}\frac{\partial \rho(h_p)}{\partial e} = k_a\frac{a}{H} \tag{B.58b}$$

$$\frac{\partial k_e}{\partial a} = \delta\sqrt{\mu}\left(-\frac{\rho(h_p)}{2a^{\frac{3}{2}}} + \frac{1}{\sqrt{a}}\frac{\partial \rho(h_p)}{\partial a}\right) = k_e\left(-\frac{1}{2a} - \frac{1-e}{H}\right) \tag{B.58c}$$

$$\frac{\partial k_e}{\partial e} = \delta\sqrt{\frac{\mu}{a}}\frac{\partial \rho(h_p)}{\partial e} = k_e\frac{a}{H} \tag{B.58d}$$

B.4.1 Circular Orbit

For circular orbits, the rate and derivative in e vanishes and the partial derivative of F_a with respect to a, combining Equation (B.38) and Equation (B.58), is

$$\frac{\partial F_a}{\partial a} = \left(\frac{1}{2a} - \frac{1}{H}\right)F_a \tag{B.59}$$

B.4.2 Low Eccentric Orbit

For low eccentric orbits, i.e. $e \leq e_b$, the partial derivative of F_a and F_e with respect to a and e, combining Equations (B.43) and (B.58) and using the product rule, are

$$\frac{\partial F_a}{\partial a} = \left(\frac{1}{2a} - \frac{1}{H}\right)F_a - k_a\exp(-z)\boldsymbol{e}^T\boldsymbol{K}_a^l\frac{e}{H}\frac{\partial \boldsymbol{I}}{\partial z} \tag{B.60a}$$

$$\frac{\partial F_a}{\partial e} = -k_a\exp(-z)\left[\frac{\partial \boldsymbol{e}^T}{\partial e}\boldsymbol{K}_a^l\boldsymbol{I} + \boldsymbol{e}^T\boldsymbol{K}_a^l\frac{a}{H}\frac{\partial \boldsymbol{I}}{\partial z}\right] \tag{B.60b}$$

$$\frac{\partial F_e}{\partial a} = \left(-\frac{1}{2a} - \frac{1}{H}\right)F_e - k_e\exp(-z)\boldsymbol{e}^T\boldsymbol{K}_e^l\frac{e}{H}\frac{\partial \boldsymbol{I}}{\partial z} \tag{B.60c}$$

$$\frac{\partial F_e}{\partial e} = -k_e\exp(-z)\left[\frac{\partial \boldsymbol{e}^T}{\partial e}\boldsymbol{K}_e^l\boldsymbol{I} + \boldsymbol{e}^T\boldsymbol{K}_e^l\frac{a}{H}\frac{\partial \boldsymbol{I}}{\partial z}\right] \tag{B.60d}$$

where

$$\frac{\partial I_n(z)}{\partial z} = \frac{1}{2}\left(I_{n-1}(z) + I_{n+1}(z)\right) \qquad \frac{\partial I_0(z)}{\partial z} = I_1(z) \tag{B.61}$$

and

$$\frac{\partial e^d}{\partial e} = de^{d-1} \tag{B.62}$$

B.4.3 High Eccentric Orbit

Using the partial derivatives of c_a and c_e from Equation (B.54) with respect to a and e

$$\frac{\partial c_a}{\partial a} = c_a \left(-\frac{1}{2a} \right) \tag{B.63a}$$

$$\frac{\partial c_a}{\partial e} = c_a \left(-\frac{1 - 4e + e^2}{2e(1 - e^2)} \right) \tag{B.63b}$$

$$\frac{\partial c_e}{\partial a} = c_e \left(-\frac{1}{2a} \right) \tag{B.63c}$$

$$\frac{\partial c_e}{\partial e} = c_e \left(-\frac{1 - 2e + 3e^2}{2e(1 - e^2)} \right) \tag{B.63d}$$

it follows that

$$\frac{\partial}{\partial a}\left(k_a c_a\right) = k_a c_a \left(-\frac{1 - e}{H} \right) \tag{B.64a}$$

$$\frac{\partial}{\partial e}\left(k_a c_a\right) = k_a c_a \left(\frac{a}{H} - \frac{1 - 4e + e^2}{2e(1 - e^2)} \right) \tag{B.64b}$$

$$\frac{\partial}{\partial a}\left(k_e c_e\right) = k_e c_e \left(-\frac{1}{a} - \frac{1 - e}{H} \right) \tag{B.64c}$$

$$\frac{\partial}{\partial e}\left(k_e c_e\right) = k_e c_e \left(\frac{a}{H} - \frac{1 - 2e + 3e^2}{2e(1 - e^2)} \right) \tag{B.64d}$$

and the partial derivatives of F_a and F_e for high eccentric orbits (see Equation (B.55)), with $e \geq e_b$, with respect to a and e become

$$\frac{\partial F_a}{\partial a} = \left(-\frac{1 - e}{H} \right) F_a - k_a c_a \frac{\partial \boldsymbol{k}^T}{\partial a} \boldsymbol{K}_a^h \boldsymbol{e} \tag{B.65a}$$

$$\frac{\partial F_a}{\partial e} = \left(\frac{a}{H} - \frac{1 - 4e + e^2}{2e(1 - e^2)} \right) F_a - k_a c_a \left[\frac{\partial \boldsymbol{k}^T}{\partial e} \boldsymbol{K}_a^h \boldsymbol{e} + \boldsymbol{k}^T \boldsymbol{K}_a^h \frac{\partial \boldsymbol{e}}{\partial e} \right] \tag{B.65b}$$

$$\frac{\partial F_e}{\partial a} = \left(-\frac{1}{a} - \frac{1 - e}{H} \right) F_e - k_e c_e \frac{\partial \boldsymbol{k}^T}{\partial a} \boldsymbol{K}_e^h \boldsymbol{e} \tag{B.65c}$$

$$\frac{\partial F_e}{\partial e} = \left(\frac{a}{H} - \frac{1 - 2e + 3e^2}{2e(1 - e^2)} \right) F_e - k_e c_e \left[\frac{\partial \boldsymbol{k}^T}{\partial e} \boldsymbol{K}_e^h \boldsymbol{e} + \boldsymbol{k}^T \boldsymbol{K}_e^h \frac{\partial \boldsymbol{e}}{\partial e} \right] \tag{B.65d}$$

where

$$r_d = l^{-d} = \left(\frac{ae}{H}(1-e^2)\right)^{-d} \tag{B.66a}$$

$$\frac{\partial r_d}{\partial a} = -dl^{-(d+1)}\frac{e}{H}(1-e^2) = -\frac{d}{a}r_d \tag{B.66b}$$

$$\frac{\partial r_d}{\partial e} = -dl^{-(d+1)}\frac{a}{H}(1-3e^2) = -\frac{d(1-3e^2)}{e(1-e^2)}r_d \tag{B.66c}$$

B.5 Derivatives of cost function with respect to parameters

The following subsections show how to derive the Jacobian, $\boldsymbol{J}$, i.e. the gradient, and the Hessian, $\boldsymbol{H}$, of the cost function in Equation (4.11) with respect to its parameters. This requires the first and second derivative of the normal distribution (Equation (4.2)) and GMM (Equation (4.1)), given in the following.

B.5.1 Derivatives of the normal distribution

The derivatives of the normal distribution with respect to its parameters are more easily derived in the logarithmic space. Using the shortened notation $\mathcal{N}_{\boldsymbol{x}|\boldsymbol{\theta}_k} = \mathcal{N}(\boldsymbol{x}|\boldsymbol{\theta}_k)$, the conversion can be achieved using the chain rule

$$\frac{\partial \mathcal{N}_{\boldsymbol{x}|\boldsymbol{\theta}}}{\partial \boldsymbol{\theta}} = \mathcal{N}_{\boldsymbol{x}|\boldsymbol{\theta}}\frac{\partial \ln \mathcal{N}_{\boldsymbol{x}|\boldsymbol{\theta}}}{\partial \boldsymbol{\theta}} \tag{B.67}$$

for the first derivative and, using the chain rule again, as

$$\begin{aligned}\frac{\partial^2 \mathcal{N}_{\boldsymbol{x}|\boldsymbol{\theta}}}{\partial \boldsymbol{\theta}^2} &= \frac{\partial \mathcal{N}_{\boldsymbol{x}|\boldsymbol{\theta}}}{\partial \boldsymbol{\theta}}\frac{\partial \ln \mathcal{N}_{\boldsymbol{x}|\boldsymbol{\theta}}}{\partial \boldsymbol{\theta}} + \mathcal{N}_{\boldsymbol{x}|\boldsymbol{\theta}}\frac{\partial^2 \ln \mathcal{N}_{\boldsymbol{x}|\boldsymbol{\theta}}}{\partial \boldsymbol{\theta}^2} \\ &= \mathcal{N}_{\boldsymbol{x}|\boldsymbol{\theta}}\left\{\left(\frac{\partial \ln \mathcal{N}_{\boldsymbol{x}|\boldsymbol{\theta}}}{\partial \boldsymbol{\theta}}\right)^2 + \frac{\partial^2 \ln \mathcal{N}_{\boldsymbol{x}|\boldsymbol{\theta}}}{\partial \boldsymbol{\theta}^2}\right\}\end{aligned} \tag{B.68}$$

for the second derivative. The logarithm of the multivariate normal distribution, l, dropping the parameter $\boldsymbol{\theta}$ for better readability, is

$$l_{\boldsymbol{x}} = \ln \mathcal{N}_{\boldsymbol{x}} = -\frac{1}{2}\left\{D \ln 2\pi + \ln|\boldsymbol{\Sigma}| + (\boldsymbol{x}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right\} \tag{B.69}$$

Bounded parameters

The partial derivative of l with respect to $\boldsymbol{\mu}$ is

$$\frac{\partial l_{\boldsymbol{x}}}{\partial \boldsymbol{\mu}} = \boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu}) = \boldsymbol{\Sigma}^{-1}\boldsymbol{z} \tag{B.70}$$

with $\boldsymbol{z} = \boldsymbol{x} - \boldsymbol{\mu} \in \mathbb{R}^D$. Using the following identities (Petersen and Pedersen, 2012)

$$\frac{\partial \ln |\boldsymbol{Y}|}{\partial \boldsymbol{Y}} = \boldsymbol{Y}^{-T} \tag{B.71a}$$

$$\frac{\partial \boldsymbol{a}^T \boldsymbol{Y}^{-1} \boldsymbol{b}}{\partial \boldsymbol{Y}} = -\boldsymbol{Y}^{-T} \boldsymbol{a} \boldsymbol{b}^T \boldsymbol{Y}^{-T} \tag{B.71b}$$

and using the fact that the inverse of a symmetric matrix is also symmetric, the partial derivative of l with respect to $\boldsymbol{\Sigma}$ can be found as

$$\frac{\partial l_{\boldsymbol{x}}}{\partial \boldsymbol{\Sigma}} = -\frac{1}{2} \left(\boldsymbol{\Sigma}^{-1} - \boldsymbol{\Sigma}^{-1} \boldsymbol{S} \boldsymbol{\Sigma}^{-1} \right) \tag{B.72}$$

where

$$\boldsymbol{S} = \boldsymbol{z}\boldsymbol{z}^T \in \mathbb{R}^{D \times D} \tag{B.73}$$

In the following, the notation of Magnus and Neudecker (1985) is used for the derivation of the second partial derivatives of l. Given any function $\boldsymbol{U}(\boldsymbol{Y})$ where $\boldsymbol{Y} \in \mathbb{R}^{n \times q}$ and $\boldsymbol{U} \in \mathbb{R}^{m \times r}$, the derivative of $\boldsymbol{U}$ with respect to $\boldsymbol{Y}$ is

$$\frac{\partial \boldsymbol{U}}{\partial \boldsymbol{Y}} = \frac{\partial \operatorname{vec} \boldsymbol{U}}{\partial \operatorname{vec}(\boldsymbol{Y})^T} \in \mathbb{R}^{mr \times nq} \tag{B.74}$$

where the vectorisation, vec, linearly transforms a matrix by stacking its columns vertically. The derivation of a matrix multiplication between $\boldsymbol{UV}$ with respect to $\boldsymbol{Y}$, with $\boldsymbol{Y}$ and $\boldsymbol{U}$ as before and $\boldsymbol{V} = \boldsymbol{V}(\boldsymbol{Y}) \in \mathbb{R}^{r \times p}$, can be found as (Magnus and Neudecker, 1985)

$$\frac{\partial \boldsymbol{UV}}{\partial \boldsymbol{Y}} = (\boldsymbol{V}^T \otimes \boldsymbol{I}_m) \frac{\partial \operatorname{vec} \boldsymbol{U}}{\partial (\operatorname{vec} \boldsymbol{Y})^T} + (\boldsymbol{I}_p \otimes \boldsymbol{U}) \frac{\partial \operatorname{vec} \boldsymbol{V}}{\partial (\operatorname{vec} \boldsymbol{Y})^T} \in \mathbb{R}^{mp \times nq} \tag{B.75}$$

with the Kronecker product, $\otimes$, and the identity matrix, $\boldsymbol{I}$. The derivation of a Kronecker product, using $\boldsymbol{Y}$ as before and $\boldsymbol{U} = \boldsymbol{U}(\boldsymbol{Y}) \in \mathbb{R}^{m \times p}$ and $\boldsymbol{V} = \boldsymbol{V}(\boldsymbol{Y}) \in \mathbb{R}^{r \times s}$, is (Magnus and Neudecker, 1985)

$$\frac{\partial \boldsymbol{U} \otimes \boldsymbol{V}}{\partial \boldsymbol{Y}} = (\boldsymbol{I}_p \otimes \boldsymbol{G}) \frac{\partial \operatorname{vec} \boldsymbol{U}}{\partial (\operatorname{vec} \boldsymbol{Y})^T} + (\boldsymbol{H} \otimes \boldsymbol{I}_r) \frac{\partial \operatorname{vec} \boldsymbol{V}}{\partial (\operatorname{vec} \boldsymbol{Y})^T} \in \mathbb{R}^{mprs \times nq} \tag{B.76}$$

where

$$\boldsymbol{G} = (\boldsymbol{K}_{sm} \otimes \boldsymbol{I}_r)(\boldsymbol{I}_m \otimes \operatorname{vec} \boldsymbol{V}) \in \mathbb{R}^{mrs \times m} \tag{B.77a}$$

$$\boldsymbol{H} = (\boldsymbol{I}_r \otimes \boldsymbol{K}_{sm})(\operatorname{vec} \boldsymbol{U} \otimes \boldsymbol{I}_s) \in \mathbb{R}^{mps \times m} \tag{B.77b}$$

with the commutation matrix, $\boldsymbol{K}_{sm} \in \mathbb{R}^{ms \times ms}$.

Thus, the second derivative of l with respect to $\boldsymbol{\mu}$ can be found by taking the derivative of Equation (B.70) with respect to $\boldsymbol{\mu}$ and applying Equation (B.75) as

$$\frac{\partial^2 l_{\boldsymbol{x}}}{\partial \boldsymbol{\mu}^2} = (\boldsymbol{I}_1 \otimes \boldsymbol{\Sigma}^{-1}) \frac{\partial \boldsymbol{z}}{\partial \boldsymbol{\mu}} = -\boldsymbol{\Sigma}^{-1} \tag{B.78}$$

The derivative of the inverse of $\boldsymbol{\Sigma}$ with respect to $\boldsymbol{\Sigma}$, required below, can be derived using Equation (B.71b) and going through all combinations of $\boldsymbol{a}$ and $\boldsymbol{b}$, as

$$\frac{\partial \boldsymbol{\Sigma}^{-1}}{\partial \boldsymbol{\Sigma}} = -(\boldsymbol{\Sigma}^{-1} \otimes \boldsymbol{\Sigma}^{-1}) \tag{B.79}$$

The second derivative of l with respect to $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ can be found by taking the derivative of Equation (B.70) with respect to $\boldsymbol{\Sigma}$ and applying Equation (B.75) as

$$\frac{\partial^2 l_{\boldsymbol{x}}}{\partial \boldsymbol{\mu} \partial \boldsymbol{\Sigma}} = (\boldsymbol{z}^T \otimes \boldsymbol{I}_D) \frac{\partial \boldsymbol{\Sigma}^{-1}}{\partial \boldsymbol{\Sigma}} \tag{B.80}$$

The second derivative of l with respect to $\boldsymbol{\Sigma}$ can be found by taking the derivative of Equation (B.72) with respect to $\boldsymbol{\Sigma}$, setting

$$\boldsymbol{B} = \boldsymbol{\Sigma}^{-1} \boldsymbol{S} \tag{B.81a}$$

$$\frac{\partial \boldsymbol{B}}{\partial \boldsymbol{\Sigma}} = (\boldsymbol{S} \otimes \boldsymbol{I}_D) \frac{\partial \boldsymbol{\Sigma}^{-1}}{\partial \boldsymbol{\Sigma}} \tag{B.81b}$$

and applying Equation (B.75) as

$$\frac{\partial^2 l(\boldsymbol{x})}{\partial \boldsymbol{\Sigma}^2} = -\frac{1}{2} \left(\frac{\partial \boldsymbol{\Sigma}^{-1}}{\partial \boldsymbol{\Sigma}} - (\boldsymbol{\Sigma}^{-1} \otimes \boldsymbol{I}_D) \frac{\partial \boldsymbol{B}}{\partial \boldsymbol{\Sigma}} - (\boldsymbol{I} \otimes \boldsymbol{B}) \frac{\partial \boldsymbol{\Sigma}^{-1}}{\partial \boldsymbol{\Sigma}} \right) \tag{B.82a}$$

$$= -\frac{1}{2} \left(\boldsymbol{I}_{D^2} - \boldsymbol{B} \otimes \boldsymbol{I}_D - \boldsymbol{I}_D \otimes \boldsymbol{B} \right) \frac{\partial \boldsymbol{\Sigma}^{-1}}{\partial \boldsymbol{\Sigma}} \tag{B.82b}$$

using

$$(\boldsymbol{\Sigma}^{-1} \otimes \boldsymbol{I}_D)(\boldsymbol{S} \otimes \boldsymbol{I}_D) = (\boldsymbol{\Sigma}^{-1} \boldsymbol{S} \otimes \boldsymbol{I}_D) = (\boldsymbol{B} \otimes \boldsymbol{I}_D) \tag{B.83}$$

Unbounded parameters

Using the lower triangular matrix, $\boldsymbol{L}$, to define $\boldsymbol{\Sigma}$ as follows

$$\boldsymbol{\Sigma} = \boldsymbol{L} \boldsymbol{L}^T \tag{B.84}$$

the optimisation can be performed unbounded. The derivative of l with respect to $\boldsymbol{L}$ – using the chain rule – is

$$\frac{\partial l_{\boldsymbol{x}}}{\partial \boldsymbol{L}} = \operatorname{vec}\left(\frac{\partial l_{\boldsymbol{x}}}{\partial \boldsymbol{\Sigma}} \right)^T \frac{\partial \boldsymbol{\Sigma}}{\partial \boldsymbol{L}} \tag{B.85}$$

where the derivative of $\boldsymbol{\Sigma}$ with respect to $\boldsymbol{L}$, using Equation (B.75), is

$$\begin{aligned} \frac{\partial \boldsymbol{\Sigma}}{\partial \boldsymbol{L}} &= \frac{\partial \boldsymbol{L} \boldsymbol{L}^T}{\partial \boldsymbol{L}} \\ &= (\boldsymbol{L} \otimes \boldsymbol{I}_D) \frac{\partial \boldsymbol{L}}{\partial \boldsymbol{L}} + (\boldsymbol{I}_D \otimes \boldsymbol{L}) \frac{\partial \boldsymbol{L}^T}{\partial \boldsymbol{L}} \\ &= \boldsymbol{L} \otimes \boldsymbol{I}_D + (\boldsymbol{I}_D \otimes \boldsymbol{L}) \boldsymbol{K}_{D,D} \end{aligned} \tag{B.86}$$

The second derivative of l with respect to $\boldsymbol{\mu}$ and $\boldsymbol{L}$ can be found by taking the derivative of Equation (B.70) with respect to $\boldsymbol{L}$, and using Equation (B.75) and the chain rule, as

$$\frac{\partial^2 l_{\boldsymbol{x}}}{\partial \boldsymbol{\mu} \partial \boldsymbol{L}} = (\boldsymbol{z}^T \otimes \boldsymbol{\Sigma}^{-1}) \frac{\partial \boldsymbol{\Sigma}^{-1}}{\partial \boldsymbol{\Sigma}} \frac{\partial \boldsymbol{\Sigma}}{\partial \boldsymbol{L}} \tag{B.87}$$

The second derivative of l with respect to $\boldsymbol{L}$ can be found by taking the derivative of Equation (B.85) with respect to $\boldsymbol{L}$ using Equation (B.75) and the chain rule as

$$\begin{aligned} \frac{\partial^2 l_{\boldsymbol{x}}}{\partial \boldsymbol{L}^2} &= \left(\left(\frac{\partial \boldsymbol{\Sigma}}{\partial \boldsymbol{L}} \right)^T \otimes \boldsymbol{I}_1 \right) \frac{\partial^2 l_{\boldsymbol{x}}}{\partial \boldsymbol{\Sigma} \partial \boldsymbol{L}} + \left(\boldsymbol{I}_{D^2} \otimes \operatorname{vec} \left(\frac{\partial l_{\boldsymbol{x}}}{\partial \boldsymbol{\Sigma}} \right)^T \right) \frac{\partial^2 \boldsymbol{\Sigma}}{\partial \boldsymbol{L}^2} \\ &= \left(\frac{\partial \boldsymbol{\Sigma}}{\partial \boldsymbol{L}} \right)^T \frac{\partial^2 l_{\boldsymbol{x}}}{\partial \boldsymbol{\Sigma}^2} \frac{\partial \boldsymbol{\Sigma}}{\partial \boldsymbol{L}} + \left(\boldsymbol{I}_{D^2} \otimes \operatorname{vec} \left(\frac{\partial l_{\boldsymbol{x}}}{\partial \boldsymbol{\Sigma}} \right)^T \right) \frac{\partial^2 \boldsymbol{\Sigma}}{\partial \boldsymbol{L}^2} \end{aligned} \tag{B.88}$$

The second derivative of $\boldsymbol{\Sigma}$ in $\boldsymbol{L}$ can be found by taking the derivative of Equation (B.86) with respect to $\boldsymbol{L}$ and applying Equation (B.75) once and Equation (B.76) twice as

$$\begin{aligned} \frac{\partial^2 \boldsymbol{\Sigma}}{\partial \boldsymbol{L}^2} &= (\boldsymbol{I}_D \otimes \boldsymbol{G}) \frac{\partial \boldsymbol{L}}{\partial \boldsymbol{L}} + (\boldsymbol{K}_{D,D}^T \otimes \boldsymbol{I}_{D^2})(\boldsymbol{H} \otimes \boldsymbol{I}_D) \frac{\partial \boldsymbol{L}}{\partial \boldsymbol{L}} \\ &= \boldsymbol{I}_D \otimes \boldsymbol{G} + (\boldsymbol{K}_{D,D}^T \otimes \boldsymbol{I}_{D^2})(\boldsymbol{H} \otimes \boldsymbol{I}_D) \end{aligned} \tag{B.89}$$

where $\boldsymbol{G}$ and $\boldsymbol{H}$, defined in Equations (B.77a) and (B.77b), are identical for the given case and can be found as

$$\boldsymbol{G} = \boldsymbol{H} = (\boldsymbol{K}_{D,D} \otimes \boldsymbol{I}_D)(\boldsymbol{I}_D \otimes \operatorname{vec} \boldsymbol{I}_D) \tag{B.90}$$

B.5.2 Derivatives of the Gaussian mixture model

The partial derivatives of $\hat{n}$ with respect to $\boldsymbol{\pi}$ and $\boldsymbol{\theta}$ are

$$\frac{\partial \hat{n}_{\boldsymbol{x}}}{\partial \pi_k} = \mathcal{N}_{\boldsymbol{x}|\boldsymbol{\theta}_k} \tag{B.91a}$$

$$\frac{\partial \hat{n}_{\boldsymbol{x}}}{\partial \boldsymbol{\theta}_k} = \pi_k \frac{\partial \mathcal{N}_{\boldsymbol{x}|\boldsymbol{\theta}_k}}{\partial \boldsymbol{\theta}_k} \tag{B.91b}$$

The second partial derivatives of $\hat{n}$ with respect to $\boldsymbol{\pi}$ and $\boldsymbol{\theta}$ are

$$\frac{\partial^2 \hat{n}_{\boldsymbol{x}}}{\partial \pi_k^2} = 0 \tag{B.92a}$$

$$\frac{\partial^2 \hat{n}_{\boldsymbol{x}}}{\partial \pi_k \partial \boldsymbol{\theta}_k} = \frac{\partial \mathcal{N}_{\boldsymbol{x}|\boldsymbol{\theta}_k}}{\partial \boldsymbol{\theta}_k} \tag{B.92b}$$

$$\frac{\partial^2 \hat{n}_{\boldsymbol{x}}}{\partial \boldsymbol{\theta}_k^2} = \frac{\partial^2 \mathcal{N}_{\boldsymbol{x}|\boldsymbol{\theta}_k}}{\partial \boldsymbol{\theta}_k^2} \tag{B.92c}$$

Introducing the unbounded logarithmic weight, $\boldsymbol{\kappa} \in \mathbb{R}^K$, such that

$$\boldsymbol{\pi}(\boldsymbol{\kappa}) = \exp \boldsymbol{\kappa} \tag{B.93}$$

the partial derivatives of $\boldsymbol{\pi}$ with respect to $\boldsymbol{\kappa}$ are

$$\frac{\partial^2 \pi_k}{\partial \kappa_k^2} = \frac{\partial \pi_k}{\partial \kappa_k} = \pi_k \tag{B.94}$$

The partial derivative of $\hat{n}$ with respect to $\boldsymbol{\kappa}$ is thus

$$\frac{\partial \hat{n}_{\boldsymbol{x}}}{\partial \kappa_k} = \frac{\partial \hat{n}_{\boldsymbol{x}}}{\partial \pi_k} \frac{\partial \pi_k}{\partial \kappa_k} = \pi_k \mathcal{N}_{\boldsymbol{x}|\boldsymbol{\theta}_k} \tag{B.95}$$

The second partial derivatives of $\hat{n}$ with respect to $\boldsymbol{\kappa}$ can be found as

$$\frac{\partial^2 \hat{n}_{\boldsymbol{x}}}{\partial \kappa_k^2} = \pi_k \mathcal{N}_{\boldsymbol{x}|\boldsymbol{\theta}_k} = \frac{\partial \hat{n}_{\boldsymbol{x}}}{\partial \kappa_k} \tag{B.96a}$$

$$\frac{\partial^2 \hat{n}_{\boldsymbol{x}}}{\partial \kappa_k \partial \boldsymbol{\theta}_k} = \pi_k \frac{\partial \mathcal{N}_{\boldsymbol{x}|\boldsymbol{\theta}_k}}{\partial \boldsymbol{\theta}_k} \tag{B.96b}$$

B.5.3 Derivatives of the cost function

The derivative of $\mathcal{C}_C$ with respect to any parameter, p, is

$$\frac{\partial \mathcal{C}_C}{\partial p} = \frac{2}{C} \sum_{i=1}^{C} (\ln \hat{n}_i - \ln n_i) \frac{\partial \ln \hat{n}_i}{\partial p} \tag{B.97}$$

For $\mathcal{C}_R$, all the derivatives other than with respect to $\boldsymbol{\kappa}$ are zero. The latter is

$$\frac{\partial \mathcal{C}_R}{\partial \kappa_k} = \frac{\partial \mathcal{C}_R}{\partial \pi_k} \frac{\partial \pi_k}{\partial \kappa_k} = 2R\pi_k^2 \tag{B.98}$$

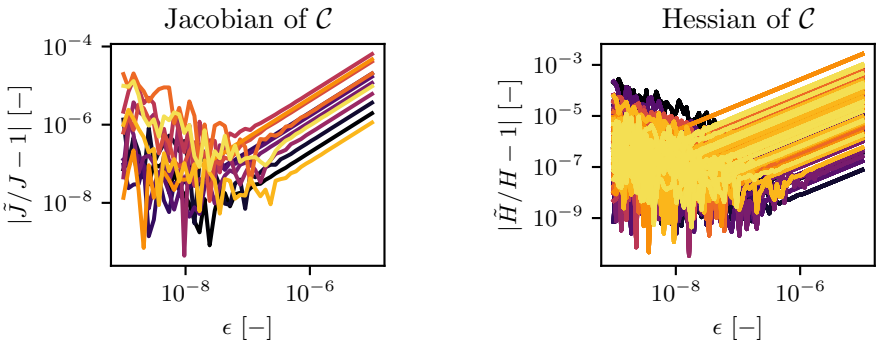
The Jacobian, $\boldsymbol{J}$, is the sum of the two

$$\boldsymbol{J} = \frac{\partial \mathcal{C}}{\partial \boldsymbol{p}} = \frac{\partial \mathcal{C}_C}{\partial \boldsymbol{p}} + \frac{\partial \mathcal{C}_R}{\partial \boldsymbol{p}} \tag{B.99}$$

To validate the formulae and its implementation, a comparison against numerical derivation is performed. Each individual component of the numerically derived Jacobian, $\tilde{\boldsymbol{J}}$, can be found with

$$\tilde{J}_i = \frac{\mathcal{C}(\boldsymbol{p}_0 + \epsilon \boldsymbol{e}_i) - \mathcal{C}(\boldsymbol{p}_0)}{\epsilon} \qquad i = 1, \ldots, N_p \tag{B.100}$$

with some small value of $\epsilon \ll 1$ and with the vector, $\boldsymbol{e}_i$, which contains all zeros apart from the i-th position which is 1. Figure B.1a shows the comparison for different values of ϵ for a random GMM with $D = K = 2$, evaluated at $C = 100$ randomly chosen sample points. The comparison



(a) Comparison of $\boldsymbol{J} = \frac{\partial \mathcal{C}}{\partial \boldsymbol{p}}$.

(b) Comparison of $\boldsymbol{H} = \frac{\partial \boldsymbol{J}}{\partial \boldsymbol{p}}$.

Figure B.1: Relative error between numerically estimated derivation and the analytical, exact derivation. For better readability, the legend is omitted.

shows a very typical pattern, where the $N_p = 12$ components become more accurate as $\epsilon \to 0$, until a certain point where the accuracy reaches a floating point rounding limit.

The second derivative of $\mathcal{C}_C$ with respect to any two parameters, p_k and p_l, can be found as

$$\frac{\partial^2 \mathcal{C}_C}{\partial p_k \partial p_l} = \frac{2}{N} \sum_{i=1}^{N} \frac{\partial \ln \hat{n}_i}{\partial p_k} \frac{\partial \ln \hat{n}_i}{\partial p_l} + (\ln \hat{n}_i - \ln n_i) \frac{\partial^2 \ln \hat{n}_i}{\partial p_k \partial p_l} \tag{B.101}$$

For $\mathcal{C}_R$, the derivatives with respect to κ_k and κ_l is

$$\frac{\partial^2 \mathcal{C}_R}{\partial \kappa_k \partial \kappa_l} = 4R\pi_k^2 \delta_{kl} \tag{B.102}$$

where

$$\delta_{kl} = \begin{cases} 1, & \text{if } k = l \\ 0, & \text{otherwise} \end{cases} \tag{B.103}$$

The Hessian, $\boldsymbol{H}$, is the sum of the two

$$H_{ij} = \frac{\partial J_i}{\partial p_j} = \frac{\partial^2 \mathcal{C}}{\partial p_i \partial p_j} = \frac{\partial^2 \mathcal{C}_C}{\partial p_i \partial p_j} + \frac{\partial^2 \mathcal{C}_R}{\partial p_i \partial p_j} \qquad i, j = 1, \ldots, N_p \tag{B.104}$$

Again, the analytical implementation can be cross checked by comparing against numerical derivation of the Hessian, $\tilde{\boldsymbol{H}}$, with

$$\tilde{H}_{ij} = \frac{J_i(\boldsymbol{p}_0 + \epsilon \boldsymbol{e}_j) - J_i(\boldsymbol{p}_0)}{\epsilon} \qquad i, j = 1, \ldots, N_p \tag{B.105}$$

Figure B.1b compares the exact and approximate Hessians for different values of ϵ and the same scenario as for the comparison of the Jacobian. As before, the pattern for each of the $N_p^2 = 144$ elements validates $\boldsymbol{H}$.

C Parameters

In the following, the parameters of the NASA SBM (Appendix C.1) and the variable smooth exponential atmosphere model (Appendix C.2) are listed.

C.1 NASA SBM Parameters

The parameters presented here are compiled from Johnson et al. (2001). Table C.1 contains the parameters required to calculate the number of fragments with Equation (2.52). The fragmenting mass in the collision, M_c, is dependent on the impact energy per target mass (Krisko, 2007)

$$\epsilon = \frac{1}{2}\frac{m_c}{m_t}v_c^2 \tag{C.1}$$

with the mass of the chaser, m_c, the mass of the target, $m_t > m_c$, and the impact velocity, v_c. If $\epsilon > 40$ J/g, the collision is considered catastrophic, i.e. both chaser and target are completely fragmented, and M_c is the sum of both objects

$$M_c = m_c + m_t \tag{C.2}$$

If $\epsilon < 40$ J/g, a non-catastrophic collision occurred, i.e. the target is cratered only. The fragmenting mass is composed of the chaser mass scaled with the square of v_c, as

$$M_c = m_c \left(\frac{v_c \text{ [km/s]}}{1 \text{ [km/s]}}\right)^2 \tag{C.3}$$

Note that this formulation differs from the one given in the original publication, however, was corrected to be of the form given in Equation (C.3) in NASA (2011). The empirically derived, unitless scaling factor, $0.1 \leq s \leq 1$, is dependent of the explosion body type.

The parameters required to calculate the logarithm to base 10, χ, of the area-to-mass ratio, A/m, as a function of the logarithm to base 10, λ, of the characteristic length, L, in Equation (2.60a) are dependent on the parent type, i.e. rocket body or payload, and on the size regime. The distributions

Table C.1: Unitless parameters related to the number of fragments, with the mass, M_c, and a scaling factor, s.

Fragmentation	k	β
Collision	$0.1\left(\frac{M_c \text{ [kg]}}{1 \text{ [kg]}}\right)^{0.75}$	1.71
Explosion	$6s$	1.6

for objects larger than $L > 11$ cm were derived by analysing decay rates of orbital fragments. For both, rocket bodies and payloads, the distribution for large objects is bimodal, composed of two normal distributions. For rocket bodies, the parameters for large objects are

$$\alpha_1 = \begin{cases} 1 & \lambda \leq -1.4 \\ 1 - 0.3571(\lambda + 1.4) & -1.4 < \lambda < 0 \\ 0.5 & 0 \leq \lambda \end{cases} \tag{C.4a}$$

$$\mu_\chi^{(1)} = \begin{cases} -0.45 & \lambda \leq -0.5 \\ -0.45 - 0.9(\lambda + 0.5) & -0.5 < \lambda < 0 \\ -0.9 & 0 \leq \lambda \end{cases} \tag{C.4b}$$

$$\sigma_\chi^{(1)} = 0.55 \tag{C.4c}$$

$$\alpha_2 = 1 - \alpha_1 \tag{C.4d}$$

$$\mu_\chi^{(2)} = -0.9 \tag{C.4e}$$

$$\sigma_\chi^{(2)} = \begin{cases} 0.28 & \lambda \leq -1.0 \\ 0.28 - 0.1636(\lambda + 1) & -1.0 < \lambda < 0.1 \\ 0.1 & 0.1 \leq \lambda \end{cases} \tag{C.4f}$$

For payloads, the parameters for large objects are

$$\alpha_1 = \begin{cases} 0 & \lambda \leq -1.95 \\ 0.3 + 0.4(\lambda + 1.2) & -1.95 < \lambda < 0.55 \\ 1 & 0.55 \leq \lambda \end{cases} \tag{C.5a}$$

$$\mu_\chi^{(1)} = \begin{cases} -0.6 & \lambda \leq -1.1 \\ -0.6 - 0.318(\lambda + 1.1) & -1.1 < \lambda < 0 \\ -0.95 & 0 \leq \lambda \end{cases} \tag{C.5b}$$

$$\sigma_\chi^{(1)} = \begin{cases} 0.1 & \lambda \leq -1.3 \\ 0.1 + 0.2(\lambda + 1.3) & -1.3 < \lambda < -0.3 \\ 0.3 & -0.3 \leq \lambda \end{cases} \tag{C.5c}$$

$$\alpha_2 = 1 - \alpha_1 \tag{C.5d}$$

$$\mu_\chi^{(2)} = \begin{cases} -1.2 & \lambda \leq -0.7 \\ -1.2 - 1.333(\lambda + 0.7) & -0.7 < \lambda < -0.1 \\ -2.0 & -0.1 \leq \lambda \end{cases} \tag{C.5e}$$

$$\sigma_\chi^{(2)} = \begin{cases} 0.5 & \lambda \leq -0.5 \\ 0.5 - (\lambda + 0.5) & -0.5 < \lambda < -0.3 \\ 0.3 & -0.3 \leq \lambda \end{cases} \tag{C.5f}$$

For small objects ($L < 8$ cm), the distribution – composed of one normal distribution only – is identical for rocket bodies and payloads. The parameters are

$$\alpha_1 = 1 \tag{C.6a}$$

$$\mu_\chi^{(1)} = \begin{cases} -0.3 & \lambda \leq -1.75 \\ -0.3 - 1.4(\lambda + 1.75) & -1.75 < \lambda < -1.25 \\ -1.0 & -1.25 \leq \lambda \end{cases} \tag{C.6b}$$

$$\sigma_\chi^{(1)} = \begin{cases} 0.2 & \lambda \leq -3.5 \\ 0.2 + 0.1333(\lambda + 3.5) & -3.5 \leq \lambda \end{cases} \tag{C.6c}$$

For the range between 8 cm $< L <$ 11 cm, the original paper simply states that a transition function is being used but does not elaborate on the type of function. For the sake of simplicity, here, a linear transition function in λ

$$\alpha_\lambda = \frac{\lambda - \lambda_l}{\lambda_u - \lambda_l} \qquad \forall \lambda_l = \log_{10}(0.08) \leq \lambda \leq \lambda_u = \log_{10}(0.11) \tag{C.7}$$

is employed, such that

$$p_{\chi|\lambda}^{\text{transition}} = \alpha_\lambda p_{\chi|\lambda}^{\text{large}} + (1 - \alpha_\lambda) p_{\chi|\lambda}^{\text{small}} \qquad \forall \lambda_l \leq \lambda \leq \lambda_u \tag{C.8}$$

The parameters required to calculate υ as a function of χ in Equation (2.60b), depending on the fragmentation type, are given in Table C.2. Finally, the parameters required to calculate A as a function of L in Equation (2.66a) are

$$b/\gamma = \begin{cases} 0.540424/2 & L \leq 0.00167 \text{ m} \\ 0.556945/2.0047077 & \text{else} \end{cases} \tag{C.9}$$

C.2 Variable atmosphere model parameters

Tables C.3 and C.4 list the parameters to calculate $\boldsymbol{a}$ and $\boldsymbol{b}$ according to Equation (3.29) as a function of the normalised $\widetilde{T}_\infty$. The two vectors are needed to recover $\hat{\rho}_j$ and H_j $\forall j$, according to Equation (3.31). Note that the model should only be used for $T_\infty \in [T_0 = 650, T_1 = 1350]$ K.

Table C.2: Parameters for the derivation of the ejection velocity.

Type	μ_υ	σ_υ
Collision	$0.9\chi + 2.9$	0.4
Explosion	$0.2\chi + 1.85$	0.4

Table C.3: Parameters to calculate $\boldsymbol{a}$ as a function of $\widetilde{T}_\infty$. The factors are of unit [km^{-1}].

j	a_{j0}	a_{j1}	a_{j2}
1	-1.98541×10^{-1}	-1.40701×10^{-2}	1.87647×10^{-2}
2	-9.71648×10^{-2}	7.16062×10^{-3}	4.77822×10^{-2}
3	-5.05069×10^{-2}	3.33725×10^{-2}	-1.85987×10^{-2}
4	-2.83356×10^{-2}	1.64584×10^{-2}	-3.32683×10^{-2}
5	-2.18893×10^{-2}	8.84693×10^{-3}	5.46460×10^{-2}
6	-6.24488×10^{-3}	4.90041×10^{-3}	-6.03999×10^{-3}
7	-2.82771×10^{-3}	-3.17505×10^{-3}	1.93697×10^{-3}
8	-8.53512×10^{-4}	7.92640×10^{-4}	-1.24063×10^{-3}
j	a_{p3}	a_{p4}	a_{j5}
1	-1.72925×10^{-2}	2.77798×10^{-2}	-9.95750×10^{-2}
2	-1.51184×10^{-1}	3.51432×10^{-1}	-7.02642×10^{-1}
3	-1.03728×10^{-1}	5.51289×10^{-1}	-1.41638×10^{-0}
4	8.69501×10^{-2}	-6.20406×10^{-2}	-3.36952×10^{-1}
5	-2.34999×10^{-1}	5.47095×10^{-1}	-8.27779×10^{-1}
6	-7.24190×10^{-2}	5.32824×10^{-1}	-1.79828×10^{-0}
7	4.29619×10^{-2}	-1.78919×10^{-1}	3.53528×10^{-1}
8	4.65874×10^{-3}	-1.87465×10^{-2}	8.70408×10^{-3}
j	a_{j6}	a_{j7}	a_{j8}
1	1.76679×10^{-1}	-1.37542×10^{-1}	3.94618×10^{-2}
2	9.01640×10^{-1}	-6.03103×10^{-1}	1.59691×10^{-1}
3	1.87770×10^{-0}	-1.22379×10^{-0}	3.11852×10^{-1}
4	8.28293×10^{-1}	-6.99209×10^{-1}	2.06734×10^{-1}
5	7.76841×10^{-1}	-4.02671×10^{-1}	8.74533×10^{-2}
6	2.85818×10^{-0}	-2.11311×10^{-0}	5.91400×10^{-1}
7	-3.82857×10^{-1}	2.16923×10^{-1}	-5.02721×10^{-2}
8	3.62357×10^{-2}	-4.73838×10^{-2}	1.66805×10^{-2}

Table C.4: Parameters to calculate $\boldsymbol{b}$ as a function of $\widetilde{T}_\infty$. The factors are of unit $[\ln(\mathrm{kg/m^3})]$.

j	b_{j0}	b_{j1}	b_{j2}
1	5.35674×10^{-0}	1.36142×10^{-0}	-1.71993×10^{-0}
2	-6.96022×10^{-0}	-1.71534×10^{-1}	-6.26282×10^{-0}
3	$-1.33334 \times 10^{+1}$	-4.29240×10^{-0}	1.12545×10^{-0}
4	$-1.78792 \times 10^{+1}$	-2.89047×10^{-0}	3.93500×10^{-0}
5	$-2.09320 \times 10^{+1}$	8.52674×10^{-0}	$-5.08863 \times 10^{+1}$
6	$-2.93700 \times 10^{+1}$	5.68339×10^{-2}	$-2.61029 \times 10^{+1}$
7	$-3.29807 \times 10^{+1}$	4.90080×10^{-0}	$1.78391 \times 10^{+1}$
8	$-3.51561 \times 10^{+1}$	-2.66659×10^{-0}	1.73783×10^{-0}
j	b_{j3}	b_{j4}	b_{j5}
1	1.48408×10^{-0}	-2.43815×10^{-0}	9.19988×10^{-0}
2	$1.70218 \times 10^{+1}$	$-3.66333 \times 10^{+1}$	$7.26606 \times 10^{+1}$
3	$1.41418 \times 10^{+1}$	$-6.27283 \times 10^{+1}$	$1.53398 \times 10^{+2}$
4	$1.67754 \times 10^{+1}$	$-1.15289 \times 10^{+2}$	$3.24667 \times 10^{+2}$
5	$1.56893 \times 10^{+2}$	$-3.21951 \times 10^{+2}$	$4.61948 \times 10^{+2}$
6	$2.90804 \times 10^{+2}$	$-1.47321 \times 10^{+3}$	$3.87334 \times 10^{+3}$
7	$-9.35850 \times 10^{+1}$	$2.24591 \times 10^{+2}$	$-3.60868 \times 10^{+2}$
8	-4.98942×10^{-0}	$2.71676 \times 10^{+1}$	$4.15537 \times 10^{+1}$
j	b_{j6}	b_{j7}	b_{j8}
1	$-1.64492 \times 10^{+1}$	$1.28147 \times 10^{+1}$	-3.67526×10^{-0}
2	$-9.47544 \times 10^{+1}$	$6.43396 \times 10^{+1}$	$-1.72245 \times 10^{+1}$
3	$-2.00134 \times 10^{+2}$	$1.29740 \times 10^{+2}$	$-3.30267 \times 10^{+1}$
4	$-4.59063 \times 10^{+2}$	$3.15704 \times 10^{+2}$	$-8.42405 \times 10^{+1}$
5	$-4.34126 \times 10^{+2}$	$2.32404 \times 10^{+2}$	$-5.27733 \times 10^{+1}$
6	$-5.21125 \times 10^{+3}$	$3.43718 \times 10^{+3}$	$-8.85649 \times 10^{+2}$
7	$3.73065 \times 10^{+2}$	$-2.15221 \times 10^{+2}$	$5.18052 \times 10^{+1}$
8	$-1.88208 \times 10^{+2}$	$1.86631 \times 10^{+2}$	$-5.96266 \times 10^{+1}$

D Validation

The validation of the newly proposed averaging method section is split into two parts. In Appendix D.1, the smooth exponential atmosphere, ρ_S, is validated against the Jacchia-77 model, ρ_J, during semi-analytical propagation using the GL contraction method. In Appendix D.2, the SI-KH is validated by comparing the contraction approximation along a full orbit, Δa and Δe, defined as

$$\Delta a = P\,F_a \qquad \Delta e = P\,F_e \tag{D.1}$$

with the orbital period, P, to numerical quadrature. For completeness, propagations of a grid of initial conditions are performed using the GL and SI-KH methods and full numerical, i.e. non-averaged integration. The latter does not resort to any averaging technique, instead integrates the full dynamics of Equation (B.26), including the fast variable: the eccentric anomaly, E.

D.1 Smooth exponential atmosphere model

To validate the smooth exponential atmosphere, ρ_S, against the Jacchia-77 density, ρ_J, for $T_\infty = 750$ K, 1000 K and 1250 K and at the same time distinguish it from the effects introduced by the SI-KH method, the following orbits are propagated using the GL method only. All unique initial orbit configurations on a 46×46 grid from 250 km $\leq h_p \leq$ 2500 km and 250 km $\leq h_a <$ 2500 km are propagated, using $B = 1$ m^2/kg. The lower limit, 250 km, is selected as an object with such a large B on a circular orbit at this altitude survives for a fraction of a day only at which point propagation in averaged dynamics becomes inaccurate. The upper limit, 2500 km, is being imposed by definition of ρ_J, but can be overcome by fitting to another model. The chosen B is large, but does not limit the validity of this validation, as inaccuracies from the averaged approach affect the propagation equally for both atmosphere models.

The integration is performed using MATLAB's ODE113 – a variable step, variable order Adams-Bashforth-Moulton integrator (Shampine and Reichelt,

Table D.1: Comparison of propagations of orbits in terms of computational efficiency and accuracy. With the total number of function evaluations, N_f^{tot}, the total integration evaluation time, t_{CPU}^{tot}, and the minimum and maximum lifetime estimation error, $\epsilon_{t_L,min/max}$.

T_∞ [K]	ρ	N_f^{tot} [−]	t_{CPU}^{tot} [s]	$\epsilon_{t_L,min/max}$ [%]
750	ρ_J	593255	1086.9	
	ρ_S	592124	169.5	−0.060/0.051
1000	ρ_J	568140	986.3	
	ρ_S	568140	153.6	−0.077/0.056
1250	ρ_J	550021	789.9	
	ρ_S	549063	149.3	−0.074/0.048

1997) – and a relative and absolute error tolerance, γ_{rel}= 10^{-6} and γ_{abs}= 10^{-12}, respectively. Figure D.1 shows the orbit lifetime, t_L, for the initial orbit grid, for propagations subject to ρ_S, and the relative error, ϵ_{t_L}, defined as

$$\epsilon_{t_L} = \frac{t_L(\rho_S) - t_L(\rho_J)}{t_L(\rho_J)} \tag{D.2}$$

comparing the propagations for each grid point using ρ_S and ρ_J, respectively. Table D.1 contains information about the maximum lifetime estimation error, $\epsilon_{t_L,min/max}$, and the computational effort in terms of the total number of function evaluations, N_f^{tot}, and the total integration evaluation time, t_{CPU}^{tot}. Over the whole specified domain and for all $T_\infty \in [750\text{ K}, 1000\text{ K}, 1250\text{ K}]$, ϵ_{t_L} remains within $[-0.1\%, 0.1\%]$, which considering the uncertainties in atmospheric density modelling is more than accurate enough (Sagnieres and Sharf, 2017). Towards low perigees ($h_p < 500$ km), the fitted ρ_S starts to wobble around the underlying model (see Figure 3.4a), which is also apparent for the propagated orbits. A 6-fold speed improvement can be observed, as no numerical integration is required when calculating the density with ρ_S.

The reduction in function evaluations and computational time observable with an increasing T_∞ is a consequence of the different density profiles. Increasing T_∞ leads to an increased ρ, which increases the drag force and thus decreases the lifetime. However, the variable-step size integration method can compensate this by increasing the step size. Two possible explanations are: as the integrator is initialised with the same properties for all three cases, the initially set (small) step size favours shorter lifetimes; and the shape of the density profiles with high T_∞ are more smooth, decreasing the number of failed function evaluation attempts.

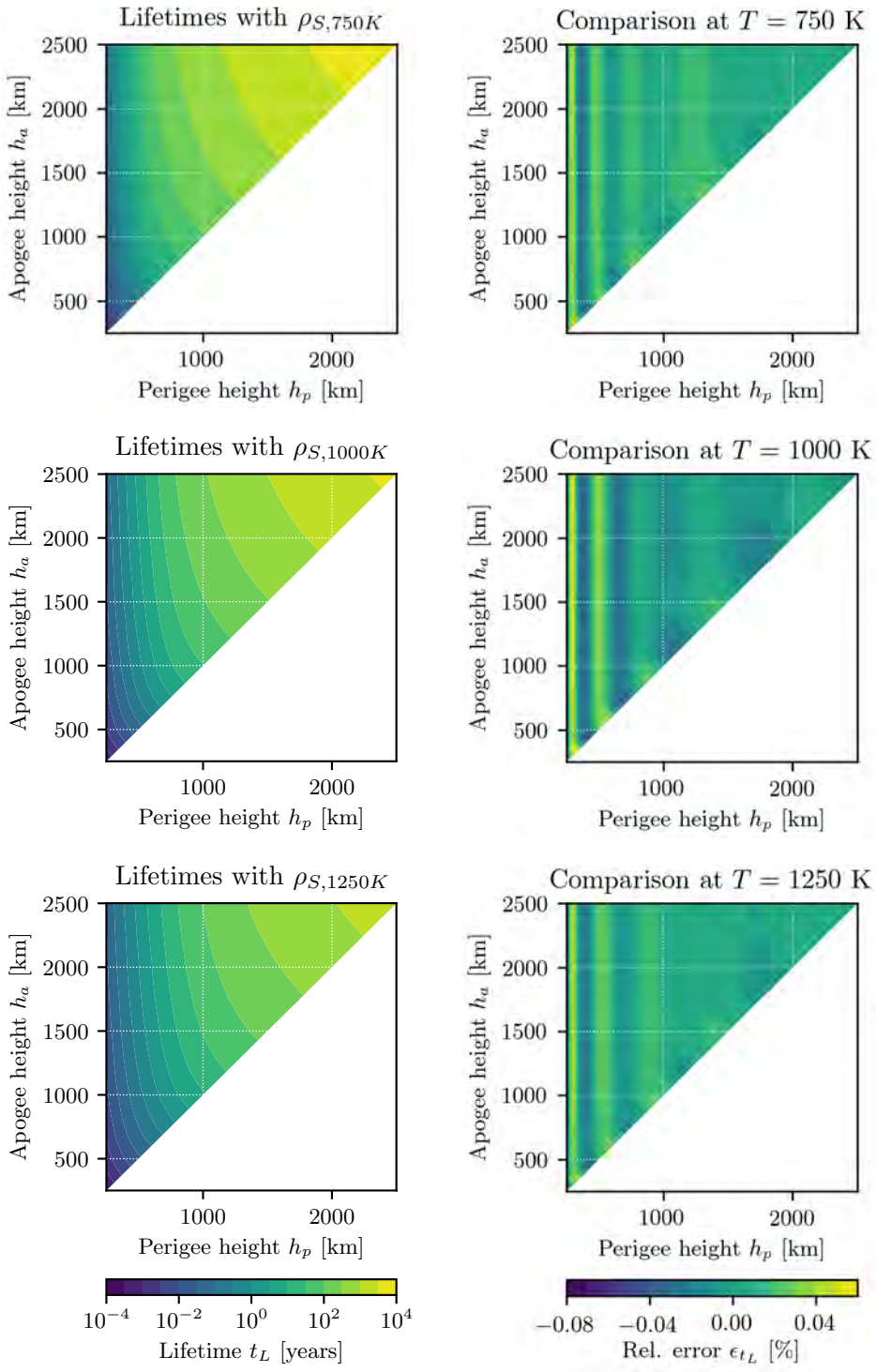


Figure D.1: Lifetimes (left) and comparison of accuracy for lifetime estimation (right) for objects being subject to ρ_J and ρ_S at $T_\infty = 750$ K (top), $T_\infty = 1000$ K (middle) and $T_\infty = 1250$ K (bottom).

D.2 Superimposed King-Hele method

The semi-analytical propagation relies on an accurate approximation of Δa and Δe. Figure D.2 shows – for different orbital configurations – the relative integral approximation error, $\epsilon_{\Delta x}$, defined as

$$\epsilon_{\Delta x} = \frac{\Delta x(\mathcal{C}_2) - \Delta x(\mathcal{C}_1)}{\Delta x(\mathcal{C}_1)} \tag{D.3}$$

where Δx is the averaged variation over a full orbital revolution for a and e, respectively. $\mathcal{C}$ is the selected contraction method: $\mathcal{C}_1$ is the numerical GL method computed using 65 nodes; and $\mathcal{C}_2$ describes the analytical formulation, KH or SI-KH, using series expansion up to 5^{th} order.

Figure D.2a shows again why orbits are predicted to re-enter much later using the classical KH contraction method: the density is underestimated at altitudes above h_p. The largest errors occur around $h_p = 125$ km and 800 km, where the rate of change in H with respect to h is large. There, the density profile is far from following a strictly exponentially decaying profile (e.g. see Figure 3.2a). Around these two altitudes, the densities at apogee altitude and thus the contraction rate in a is underestimated by more than 10% and 20%, respectively, if $e > 0.03$.

Instead, using the SI-KH method, the relative error remains well below 0.1% for all $h_p \in [100\text{ km}, 2500\text{ km}]$ and $h_a \in [100\text{ km}, 100000\text{ km}]$ (see Figure D.2b), a range that includes the vast majority of all Earth orbiting objects. Discontinuities can be found whenever e passes through $e_b = e_b(H_p)$. The biggest step occurs for the largest H_j. Those discontinuities slightly increase the number of steps required during the integration. However, given the averaged dynamics, γ_{rel} can be chosen large enough during integration mitigating the effects of the discontinuities.

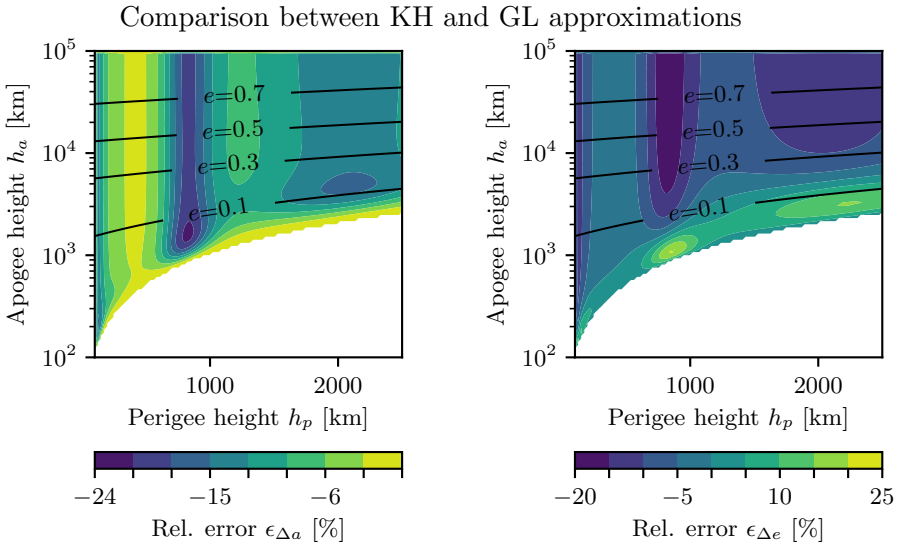
To see how the SI-KH compares against GL and non-averaged propagation in terms of accuracy and computational power, the results from different initial orbit conditions are compared, for two scenarios:

a) Short-term re-entry duration: $t_L = 30$ days

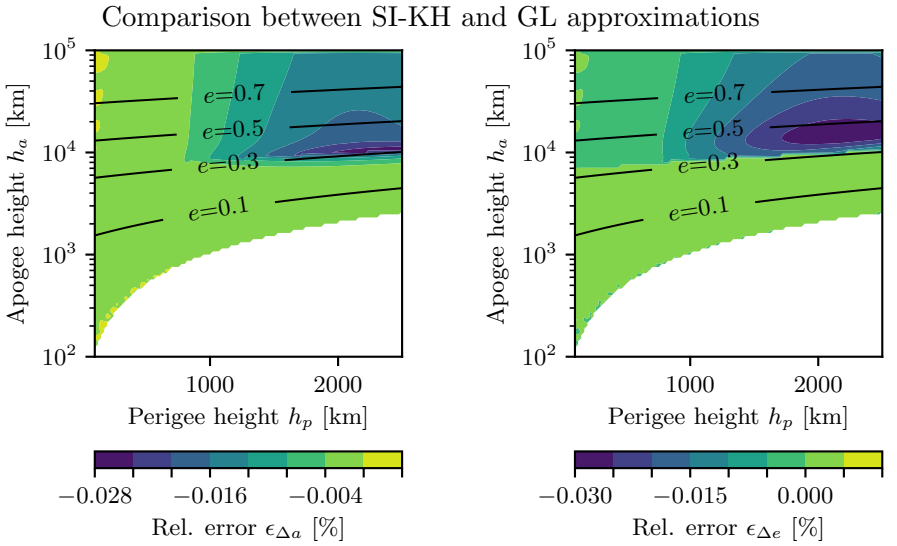
b) Mid-term re-entry duration: $t_L = 360$ days

The reasons why long-term re-entry cases are not discussed here are twofold: First, for long time spans, the non-averaged integration requires small relative tolerances. If they are not met, the result cannot be trusted; Secondly, the longer the time spans, i.e. the smaller B, the more accurate the assumptions made for the averaged propagation.

The initial conditions are spaced in $h_p \in [250\text{ km}, 2500\text{ km}]$ and $h_a \in [250\text{ km}, 100000\text{ km}]$ and consist of all unique solutions on a 46×46 grid, where the grid spacing in h_a is chosen to be logarithmic, as opposed to the

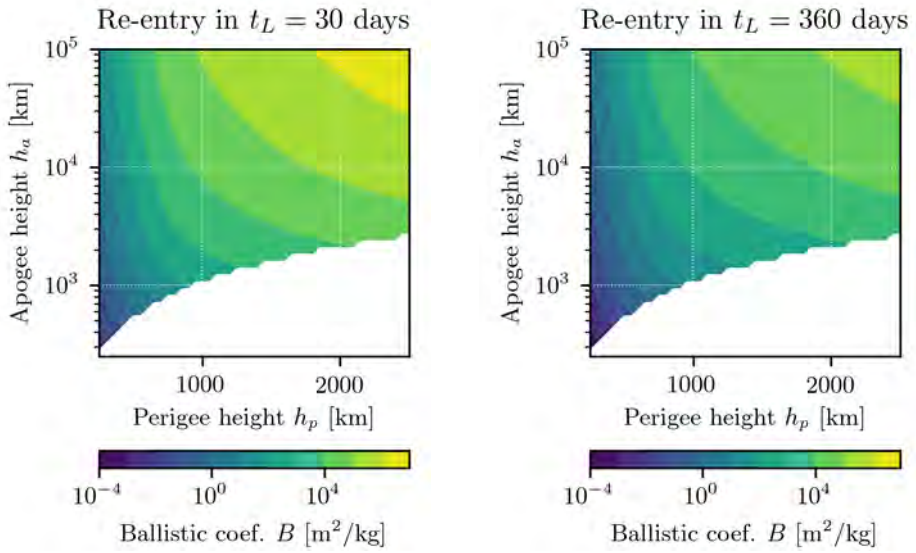


(a) Analytical KH approximation compared against GL quadrature. Differences of up to 25% can occur for certain orbital configurations.



(b) Analytical SI-KH approximation compared against GL quadrature. The error remains below 0.1% across the domain.

Figure D.2: Comparison for accuracy in Δa (left) and Δe (right) for different approximation methods. The underlying atmosphere model is ρ_S at T_∞ = 1000 K. Note that the colour bar range of the lower figure is 3 orders of magnitudes smaller than the one of the upper figure.



(a) Effective area-to-mass ratio required to re-enter in 30 days.

(b) Effective area-to-mass ratio required to re-enter in 360 days.

Figure D.3: The minimum ballistic coefficient is $B_{min} = 1.5 \times 10^{-4}$ m/kg^2 in order to remain in orbit for 360 days from a circular orbit at $h_p = h_a = 250$ km. The maximum, in order to re-enter in 30 days from $h_p/h_a = 250/100000$ km, is $B_{max} = 3.0 \times 10^6$ m/kg^2.

equidistant grid in h_p. Two preliminary runs were performed using the SI-KH method to calculate the lifetimes. This way, the B required to re-enter within the given timespan can be estimated. Figure D.3 shows the grids of the resulting B for both scenarios. Note that B varies by almost 13 orders of magnitude.

The accuracy is described again as the relative lifetime, ϵ_{t_L}, this time defined as

$$\epsilon_{t_L}^{ij}(\mathcal{M}_1, \mathcal{M}_2, h_{p,i}, h_{a,j}) = \frac{t_L(\mathcal{M}_1, h_{p,i}, h_{a,j}) - t_L(\mathcal{M}_2, h_{p,i}, h_{a,j})}{t_L(\mathcal{M}_2, h_{p,i}, h_{a,j})} \tag{D.4}$$

where $\mathcal{M}$ is the selected contraction and integration method, combined with a given relative integrator tolerance, γ_{rel}, during integration. To give a feeling for the accuracy across all the different initial conditions, the 50%- and 100%-quantiles, i.e. the median and maximum denoted as $\epsilon_{t_L,50\%}$ and $\epsilon_{t_L,100\%}$, respectively, over all the $|\epsilon_{t_L}^{ij}|$ are given. The computation effort is compared via the total number of function calls, N_f^{tot}, and time required for

Table D.2: Performance of the different propagation and contraction methods, for a) $t_L = 30$ days and b) $t_L = 360$ days and various relative integration tolerances, γ_{rel}. NA stands for not averaged, i.e. full numerical propagation. All figures are unitless.

	$\mathcal{M}_1$	$\mathcal{M}_2$	$\epsilon_{t_L,50\%}$	$\epsilon_{t_L,100\%}$	ϵ_{N_f}	$\epsilon_{t_{CPU}}$
a)	SI-KH/10^{-6}	SI-KH/10^{-12}	$3.2e-6$	$8.4e-5$	$3.0e-1$	$2.9e-1$
	GL/10^{-6}	GL/10^{-12}	$3.3e-6$	$7.0e-5$	$3.7e-1$	$3.7e-1$
	NA/10^{-6}	NA/10^{-12}	$1.3e-3$	$2.5e-2$	$3.4e-1$	$3.4e-1$
	NA/10^{-9}	NA/10^{-12}	$1.6e-6$	$3.1e-5$	$6.0e-1$	$6.2e-1$
	SI-KH/10^{-6}	NA/10^{-12}	$8.7e-4$	$1.8e-3$	$1.1e-2$	$2.2e-2$
	GL/10^{-6}	NA/10^{-12}	$8.7e-4$	$1.7e-3$	$1.0e-2$	$3.6e-2$
b)	SI-KH/10^{-6}	SI-KH/10^{-12}	$3.2e-6$	$6.9e-5$	$3.1e-1$	$3.2e-1$
	GL/10^{-6}	GL/10^{-12}	$3.7e-6$	$6.9e-5$	$3.9e-1$	$4.0e-1$
	NA/10^{-6}	NA/10^{-12}	$1.6e-2$	$2.6e-1$	$3.4e-1$	$3.7e-1$
	NA/10^{-9}	NA/10^{-12}	$1.9e-5$	$4.1e-4$	$6.1e-1$	$6.4e-1$
	SI-KH/10^{-6}	NA/10^{-12}	$7.0e-5$	$3.2e-4$	$5.8e-4$	$1.1e-3$
	GL/10^{-6}	NA/10^{-12}	$7.2e-5$	$4.9e-4$	$5.8e-4$	$2.1e-3$

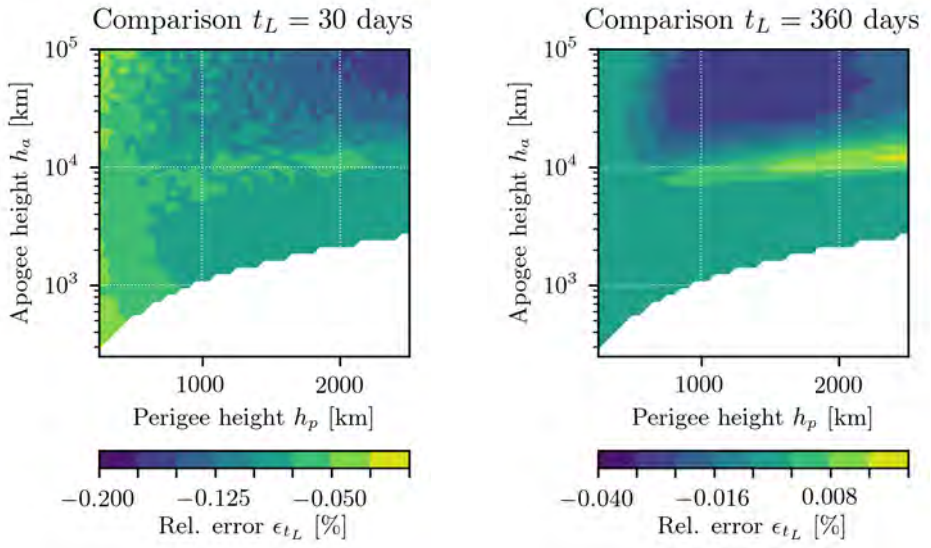
the integration itself, t^{tot}_{CPU}

$$\epsilon_{N_f}(\mathcal{M}_1, \mathcal{M}_2) = \frac{N_f^{tot}(\mathcal{M}_1)}{N_f^{tot}(\mathcal{M}_2)} \tag{D.5a}$$

$$\epsilon_{t_{CPU}}(\mathcal{M}_1, \mathcal{M}_2) = \frac{t_{CPU}^{tot}(\mathcal{M}_1)}{t_{CPU}^{tot}(\mathcal{M}_2)} \tag{D.5b}$$

Table D.2 shows the performance of propagating with the various contraction methods and tolerances. For both SI-KH and GL, the absolute maximum error over the whole grid and over both scenarios remains below 0.01%, when decreasing γ_{rel} from 10^{-6} to 10^{-12}. Given this force model, it is therefore sufficient to use $\gamma_{rel} = 10^{-6}$. For the Non-Averaged (NA) integration, this is not the case. While the maximum error remains modest (0.18%) in the short-term case, it becomes large when the re-entry span is increased to one year (26%), when decreasing γ_{rel}. Decreasing $\gamma_{rel} = 10^{-9}$ and comparing to integration with $\gamma_{rel} = 10^{-12}$, reduces the maximum error for the non-averaged propagation in the mid-term case to 0.032%.

For the comparison of the averaging techniques against non-averaged propagation, the tolerance of the latter is set to $\gamma_{rel} = 10^{-12}$. Again, SI-KH and GL fare very similar. For the short-term case, the boundaries of the averaged propagation can be recognised for very high B, leading to still small maximum errors of 0.18% and 0.17%, respectively. Figure D.4a shows the



(a) Small errors occur for large effective area-to-mass ratios ($B > 10^4$ m^2/kg).

(b) Two areas of very small errors can be distinguished, stemming from the series truncation.

Figure D.4: Relative error ϵ_{t_L} when comparing averaged propagation using SI-KH with γ_{rel}= 10^{-6} against non-averaged integration with $\gamma_{rel} = 10^{-12}$.

resulting lifetime comparison for SI-KH and $t_L = 30$ days. As B increases to values above 10^4 m^2/kg, the assumption of constant a and e over one orbit starts to break down and small relative errors up to 0.2% are introduced. This might be an issue for small debris such as multi-layer insulation fragments and paint flakes. For the mid-term scenario, the maximum error reduces by one order of magnitude for both averaging methods tested. For high $h_a > 10000$ km, the relative errors introduced due to the series expansion applied in the SI-KH method remain below 0.04% (see Figure D.4b).

E Optimisation

Two frameworks for the minimisation of the cost function are studied here: line search and trust-region (Nocedal and Wright, 2006). Both use a quadratic approximation of the objective function. The two frameworks and various optimisation methods are presented in Appendix E.1 and compared in Appendix E.2 in terms of speed of convergence.

E.1 Methods

In the line search framework, the model is used to find a suitable search direction. The search direction generally follows a descent direction, i.e. is within 90 deg of the steepest descent. The function is then minimised along this line. In an iterative procedure, the distance along the search direction is given by a variable step length.

The trust-region framework uses the model, which is trusted within the region, to optimise the step direction and size simultaneously. In case a step is not acceptable, i.e. the model did not appropriately approximate the function, the step is rejected, and the trust-region shrinks. If, however, the model approximates the function adequately, the trust-region can be increased to accelerate convergence.

Different optimisation methods exist that can be employed in either framework. Here, a very brief overview is given, following Nocedal and Wright (2006) and mentioning the corresponding algorithm implemented in the SciPy package (Jones et al., 2001). An overview of the methods is given in Table E.1

One of the conceptually easiest methods is the one using the *steepest descent direction*. To find the direction, the gradient of the function is required at each iteration, but not the Hessian. However, for difficult problems, convergence to the minimum can be slow.

The *Newton method* makes use of a quadratic model approximation. Thus, it requires the Hessian around each point, which is used to obtain the

Table E.1: Overview of optimisations algorithms implemented in SciPy. Optional means that – if the derivatives are not given – the algorithm approximates them.

Algorithm	Method	$\boldsymbol{J}$ req.	$\boldsymbol{H}$ req.	stop tol.
CG	Non-linear CG	optional	no	10^{-5}
BFGS	Quasi-Newton	optional	no	10^{-5}
L–BFGS–B	Quasi-Newton	optional	no	10^{-5}
Newton-CG	Newton CG	yes	optional	3×10^{-3}
TNC	Newton CG	optional	no	2×10^{-4}
trust-ncg	Newton CG	yes	yes	10^{-4}
trust-krylov	Lanczos	yes	yes	10^{-4}

search direction. Such methods converge typically very fast. However, evaluating the Hessian, if given analytically, can be computationally expensive as the number of elements grows with $\propto D^4$. Another class of methods to find search directions is the *non-linear Conjugate Gradient (CG)* method, that was originally developed to solve linear systems. In this algorithm, subsequent search directions are found that are conjugate to each other. Only the gradient information is required to derive the directions.

To mitigate the expensive calculation of the Hessian to find the directions defined by the *Newton method*, the *Quasi-Newton method* can be helpful. Instead of calculating the true Hessian, an approximation of it is maintained using gradient information only. The approximation of the Hessian is updated at each iteration. Thus, a better search direction can be derived leading to a fast, superlinear convergence rate. An example is the BFGS algorithm (Nocedal and Wright, 2006, Chapter 6.1), or its memory limited version, the L–BFGS–B, which continuously forgets previous steps.

Other methods for large-scale unconstrained optimisation are *inexact Newton methods*, where approximations to the *Newton method* are found using the CG method or the Lanczos method. The storage of the full Hessian approximation is not required, as the Hessian-vector product, i.e. the matrix multiplication between the Hessian and the search direction, can be found directly using finite differences. Examples in the line search framework are the Newton-CG and the truncated Newton (TNC) algorithm and in the trust-region framework the trust-ncg and trust-krylov.

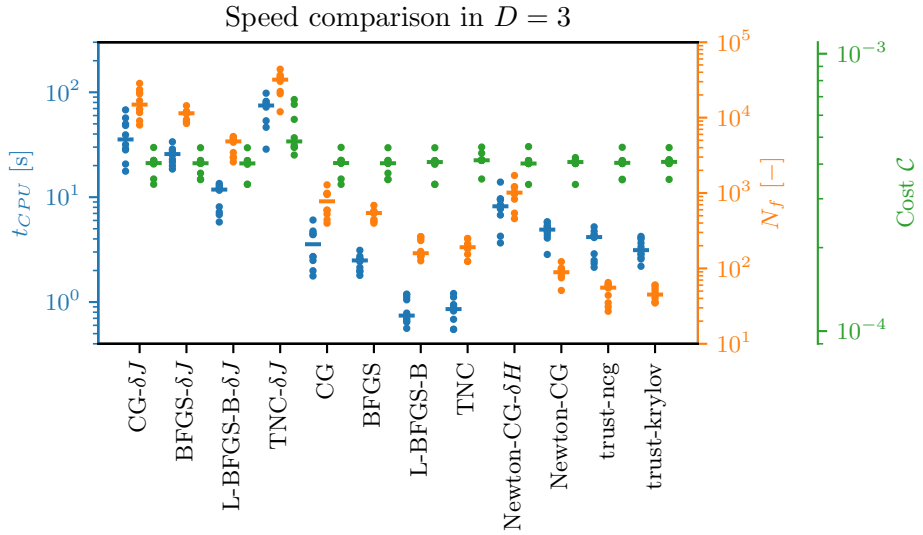
E.2 Benchmarking

To find a suitable candidate algorithm for the minimisation in Equation (4.14), a random GMM is generated, sampled and fitted using the different methods given in Table E.1. The number of kernels of the randomly sampled model is set as $K = 2D$ and that of the surrogate model as $K = 2(D-1)$ to allow for inexact results. The tolerances for the stop criteria were selected such as to achieve comparable cost function values. The fitting process was repeated 10 times to find robust results. The benchmarking tests were performed on Intel® Xeon® E5-4620 v4 CPUs.

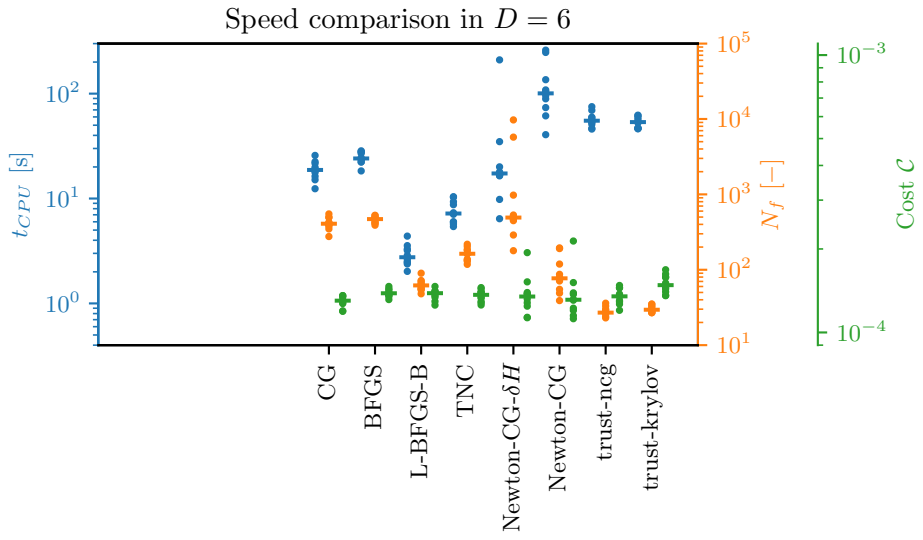
The results in terms of computational time required, t_{CPU}, number function evaluation, N_f, and cost, $\mathcal{C}$, for a three- and six-dimensional case can be found in Figure E.1, corresponding to optimisation in $N_p = 40$ and $N_p = 280$ parameters, respectively. The dimensionality of the two cases is chosen such as to allow a wide range of applications to fragmentation modelling (see Chapter 6). For all the algorithms where using an analytical function of $\boldsymbol{J}$ or $\boldsymbol{H}$ is optional, the finite-differences approximation is benchmarked too (marked with δJ or δH). Note that due to their poor performance in $D = 3$, gradient-based algorithms using finite differences, are omitted in the $D = 6$ benchmarking test. There is a lot of information in this figure, allowing to draw a few preliminary conclusions. Note that the application of each of the methods could be fine-tuned to further increase their efficiency. Hence, the following considerations are to be taken with caution.

First, as required, all algorithms terminate around a similar accuracy, except the TNC-δJ algorithm. It performs worse than the other algorithms in terms of computational cost despite stopping early. The linear search of TNC-δJ failed 2 out of 10 times, probably due to insufficient accuracy in estimating the gradient. None of the other methods ever failed to converge. Second, algorithms that use finite differences to derive the gradient require nearly 2 orders of magnitude more function evaluations than the ones that use the analytical expression. Thus, despite the expensive evaluation of $\boldsymbol{J}$, the latter outperform the former easily. Third, fully estimating $\boldsymbol{H}$ with finite differences versus using the exact formulation has mixed results. The number of function evaluations is lower with the analytical expression, however, the higher the dimensions, the slower the evaluation of $\boldsymbol{H}$. Even though fully Hessian-based algorithms use the least amount of N_f, they are being outperformed by analytical gradient-based algorithms in terms of t_{CPU}. This is related to the specific implementation of the evaluation of $\boldsymbol{H}$, however, it is also a general observation made by Bishop (2006).

Surprisingly, the L–BFGS–B outperforms the BFGS algorithm by an order of magnitude in $D = 6$. The speed increase arises not from an increase in efficiency, but from finding a better path. One reason might be, that the initial guess is far from the minimum. Thus, being able to forget previous



(a) For the 3-dimensional case.



(b) For the 6-dimensional case.

Figure E.1: Comparison of the algorithms in terms of speed for $D = 3$ and $D = 6$. The points indicate the result of each optimisation run, the dashes the median.

estimates of $\boldsymbol{H}$ – the L–BFGS–B takes only information of the last iteration – could be an advantage. Together with the TNC, the L–BFGS–B seems best suited to the task. The latter is chosen as the default optimiser, solving an optimisation problem in $N_p = 280$ parameters within $1 - 4$ s.