



**UiT** The Arctic University of Norway

Faculty of Science and Technology  
Department of Physics and Technology

## **Interstellar Dust in the Inner Heliosphere and Impact Detection Capabilities with ESA's Solar Orbiter Spacecraft**

FYS-3931

Aleksander Johan Henriksen

Master thesis in Space Physics - June 2022



“Never tell me the odds!”  
–Han Solo



# Abstract

ESA's Solar Orbiter spacecraft provides a new opportunity to investigate the inner heliosphere close to sun, by taking measurements with state-of-the-art instruments it will help uncover the mysteries of the Sun. Solar Orbiter is planned for a 7 year mission, reaching as close as  $0.28 AU$  to the sun. Using a process called impact ionization, dust fluxes can be measured during the mission. Some dust particles that reside outside our solar system in the local interstellar clouds can enter the solar system from an upstream direction of  $258^\circ$  Heliocentric Ecliptic Longitude. These dust particles are referred to as interstellar dust.

This Master Thesis looks at the prospect of using ESA's Solar Orbiter for interstellar dust detection through dust impact measurements. Models have been created in this thesis to describe the interstellar dust trajectories and impact velocities on the spacecraft. Comparing possible impact velocities during the orbit to measured dust fluxes helps to interpret the observational data. This has been done for interstellar dust and for interplanetary dust particles in unbound orbits around the sun. The trajectories are determined by solar radiation pressure force and solar gravity. This work does not consider Lorentz Forces in any of the calculations, as it can be assumed all calculated particles have a low enough charge to mass ratio to not be heavily affected by magnetic forces.

Impact velocities from interstellar dust are found to range from  $40 \text{ km/s}$  to  $80 \text{ km/s}$  when Solar Orbiter travels in the upstream direction ( $80^\circ$  to  $270^\circ$ ), with the more gravitational dominated particles reaching the highest velocities. Dust impact data show that interplanetary dust particles contribute more to the measured dust flux than interstellar dust when Solar Orbiter is closer to the sun, and the interstellar dust upstream direction ( $258^\circ$ ) could not be measured with the data. The model calculations show that the gravitational focusing region behind the sun ( $78^\circ$ ) has a high population density of interstellar dust and high impact velocities and future work should be directed to this region.



# Acknowledgements

I would like to give my gratitude to my supervisor Ingrid Mann, for guiding me through the semester and helping with difficult problems, both as an advisor and semi-therapist.

I also want to thank Andreas Kvammen and Samuel Kociścák for the insight I gained through our discussions, help received with data, scripts, proof reading and all other help provided for this Master Thesis.

Most of all I want to thank my family and Kristine for always being there and giving me the support I needed.





# Contents

<b>Abstract</b>	<b>iii</b>
<b>Acknowledgements</b>	<b>v</b>
<b>List of Figures</b>	<b>ix</b>
<b>Acronyms</b>	<b>xiii</b>
<b>1 Introduction</b>	<b>1</b>
<b>2 ESA's Solar Orbiter Spacecraft</b>	<b>3</b>
2.1 Mission . . . . .	3
2.1.1 Launch and Orbit . . . . .	3
2.1.2 Mission Objectives . . . . .	5
2.2 Instruments . . . . .	6
2.2.1 The Physics of Dust Impact Ionization . . . . .	6
2.2.2 Antenna signal shapes . . . . .	7
2.2.3 RPW Dust Detection Algorithms . . . . .	9
<b>3 What is Interstellar Dust?</b>	<b>11</b>
3.1 Background . . . . .	11
3.2 Forces that determine the flow of ISD . . . . .	13
3.2.1 Lorentz Forces . . . . .	16
<b>4 Methods for Calculating Impact Velocity</b>	<b>21</b>
4.1 Using Equations to Calculate Impact Velocity . . . . .	21
4.2 $\beta$ -meteoroids . . . . .	27
4.3 Alternative Method using Initial Value Problem Solvers . . . . .	27
4.4 Discussion . . . . .	29
<b>5 Impact Velocities and ISD Dynamics</b>	<b>31</b>
5.1 Impact Velocities . . . . .	31
5.2 Information of ISD in the inner heliosphere . . . . .	34
5.3 Discussion . . . . .	36

<b>6</b>	<b>A Brief Look at the RPW Dust Impact Data</b>	<b>39</b>
6.1	Information on the data . . . . .	39
6.2	Dust Impacts per day . . . . .	40
6.3	Relation to Impact Velocity . . . . .	44
<b>7</b>	<b>Conclusion</b>	<b>47</b>
	<b>Bibliography</b>	<b>49</b>
<b>A</b>	<b>appendix</b>	<b>53</b>
A.1	Special Curriculum . . . . .	53
A.1.1	Abstract from Project Paper A. Henriksen 2020 . . . . .	53
A.2	Extra equations . . . . .	53
A.3	Code . . . . .	55
A.3.1	Solar Orbiter Ephemeris . . . . .	55
A.3.2	Analytical Impact Velocity . . . . .	59
A.3.3	Numerical calculations . . . . .	66
A.4	Dust Impact Data . . . . .	76

# List of Figures

2.1	The planned trajectory of Solar Orbiter in the Heliocentric Aries Ecliptic model, from February 11th 2020 to November 20th 2030. Color indicates the days after launch, in the later parts the inclination increases which is not visible in the flat image . . . . .	4
2.2	Heliocentric Aries Ecliptic model shown from the sun with its respective axes, and an image showing where the first point of Aries is located relative to Earth. Figures from (Suresh and Sivan, 2015) (Lang, 2000) . . . . .	5
2.3	(Zaslavsky <i>et al</i> 2015) Simulation of signal caused by a dust impact. Red line indicates the antenna potential, green line indicates the potential of the spacecraft body that was hit by dust particle. Black line is the difference in potential measured between the spacecraft body and antenna in a monopole mode. . . . .	8
3.1	Calculated ratios of radiation pressure to gravity for local interstellar cloud dust models explained in the figure and shown as lines and dots. The shaded areas depict the range of values that are consistent with flux measurements.(Mann, 2010) . . . . .	12
3.2	XY Plane showing gravitational focusing of $\beta = 0$ and exclusion zones for $\beta > 1$ . . . . .	14
3.3	Numerically calculated ISD velocities starting from 50 AU away with impact parameters containing a width and height of 2 AU. Note that the color axis for $\beta = 1.3$ go lower than the others to display the decrease in velocity when the ISD reaches close to the sun . . . . .	16
3.4	ISD can be found to exist with parameters where Lorentz forces does not heavily dictate the flow. . . . .	18
3.5	Focusing and defocusing of charged ISD, from (Sterken <i>et al.</i> , 2012). The particle trajectories are for $\beta = 1$ and $Q/m = 0.5$ C/kg starting at 50 AU away, the start year is 1990 on 3.5a, and 2000 on 3.5b . . . . .	19

3.6	Solar Cycle 25 predictions from NOAA from 1995 to 2035, published December 9th 2019 . . . . .	20
4.1	Transforming the coordinate system from HAE to an ISD fixed system will change the orbit of the spacecraft as we see in Figure 4.1b. . . . .	22
4.2	Hyperbolic parameters $\delta$ and $\theta_\infty$ describe the sharpness of the hyperbola based on the angle between the asymptotes. They are dependant on the eccentricity of the hyperbola and show deflection angle of the trajectory. Note that the semi-minor axis is equal to the impact parameter, the shortest distance from the focus point to the incoming asymptote. The position $(r_s, f_s)$ is the spacecraft polar coordinates an ISD is needs to reach to impact. . . . .	24
4.4	Numerically calculated impact events for Silicate Spheres, with a histogram showing the ISD velocity at the marked locations. (Henriksen A. 2021) . . . . .	28
5.1	The speed of Solar Orbiter is shown in the left image with the color indicating the orbital velocity of the spacecraft, on the right impact vectors on Solar Orbiter from ISD is shown. Red arrows indicate the vectors from ISD and white arrows represent vectors from Solar Orbiter . . . . .	32
5.2	ISD impact velocities analytically calculated for different $\beta$ values. The Impact velocity has only been calculated for incoming particles, leaving a big edge in the focused region due to the seperation of two different directions the ISD is travelling and which one the impact velocity has been calculated for. . . . .	33
5.3	Impact Velocities for ISD with $\beta = 1$ is shown on the left and IDPs are shown on the right with a constant radial velocity of 80 km/s . . . . .	34
5.4	Impact Velocity of ISD and IDPs compared on the same orbit. Red line represent ISD impact velocity for $\beta = 0.5$ , black line represent IDPs with $V_r = 80$ km/s . . . . .	34
5.5	Gaussian kernel Density Estimation of ISD population with $\beta$ values between 0.3 and 0.9 inside a region from 0.5 AU to 1.1 AU in the Ecliptic plane. Colors indicate hotspots where ISD trajectories overlap. . . . .	35
5.6	The histogram on the left has 500 bins of ISD velocities in the inner heliosphere covering 0.5 AU to 1.1 AU in the ecliptic plane, The right image shows the speed of ISD particles as a function of the heliocentric distance, $\beta$ ratios are seperated by color with the values shown in the colorbar on the right . . . . .	36

6.1	The data has measured dust impacts per day using two different flagging algorithms. The CNN classification detects about 26% more dust than TDS flagging (Kvammen et al., 2022). All figures show the dust flux as a function of a spacecraft parameter, Fig 6.1a is a function of the heliocentric longitude in HAE, Fig 6.1b as heliocentric distance, 6.1c heliocentric longitude with spacecraft velocity on a separate Y axis, and 6.1d as a function of the spacecraft velocity . . . . .	41
6.2	Averaged dust flux over five years as function of solar ecliptic longitude. Red curve is a fitted model. From (Belheouane et al., 2012) . . . . .	42
6.3	As the spacecraft got closer to the sun, its velocity is increased and so would the measured dust impacts. . . . .	42
6.4	Dust impacts averaged over multiple days, as a function of spacecraft velocity . . . . .	43
6.5	Dust impacts per day as a function of heliocentric longitude. Color is represented as the velocity at each recorded day, on the left is the impact velocity from ISD with $\beta = 0.5$ and on the right the color is the impact velocity produced from IPDs . . . . .	44
6.6	The ratio of dust impacts per day over the spacecraft velocity, shown as a scatter plot with the impact velocity from ISD $\beta = 0.5$ on the left and a 10 day averaged impacts on the right . . . . .	45
A.1	Very big equation to calculate one of the Velocity components for an ISD particle at a spacecraft coordinate . . . . .	54



# Acronyms

**AU** Astronomical Unit

**CNN** Convolutional Neural Network

**HAE** Heliocentric Aries Ecliptic

**IDP** Interplanetary Dust Particle

**IMF** Interplanetary Magnetic Field

**ISD** Interstellar Dust

**ISM** Interstellar Medium

**LIC** Local Interstellar Cloud

**RPW** Radio and Plasma Waves

**TDS** Time Domain Sampler







# Introduction

Since the 1970s consistent research on Interstellar Dust (ISD) has been conducted, but the local interstellar clouds they originate from have probably been detected by ancient astronomers and scientists a long time ago and simply ignored as “noise” blocking the stars. The local interstellar clouds (LIC) surround the solar system and can help reveal information about early formations of solar systems and the content of elements in our galaxy. By looking at the extinction of light from the interstellar medium (ISM) the presence of dust can be inferred (Mann, 2010), and while a lot of information can be gathered from remote sensing through measuring the scattering, polarization and thermal emissions of ISD, in-situ measurements provide a whole different kind of detailed data with the right equipment. ISD comes into our heliosphere when our solar system travels through the LIC in our galaxy, the relative speed between the LIC and our solar system is roughly 26 km/s and ISD enters with the same velocity from a direction of  $258^\circ$  ecliptic longitude and  $8^\circ$  ecliptic latitude, this direction is therefore known as the upstream direction of ISD.

In-Situ measurements of ISD has been done on multiple missions into interplanetary space, some spacecraft have been equipped with instruments that allow for the dust composition to be revealed, but even on spacecraft without such equipment we can still take in-situ measurements using other techniques, as we will mention later on. The STEREO<sup>1</sup> spacecraft was able to detect ISD and measure the upstream direction over a period of 5 years at  $1 AU$  (Astronomical

1. For more info on STEREO click on this Web link

Unit) with techniques known as dust impact ionization (Belheouane et al., 2012), using the same techniques this paper will look at the Solar Orbiter spacecraft and its potential for detecting ISD in the inner heliosphere. Previously most measurements have been outside of  $1 AU$  and the environment close to the sun still remains an area where ISD has not been thoroughly investigated.

Solar Orbiter orbits the sun in an ellipse reaching as close as 60 Solar Radii over time, it will achieve this by making multiple gravity assists by Venus and Earth. During this time the spacecraft travels between different regions in the heliosphere and provides an ample opportunity to map the dust environment in these regions, and possibly provide data on ISD flow close to the sun. Some of the challenges with trying to detect ISD this close to the sun is the obscurity of data caused by interplanetary dust particles that crowd the region, the distinction between these will be the determining factor in confirming whether the spacecraft has detected ISD particles or not. A second challenge is the interaction between charged particles and the interplanetary magnetic field (IMF), determined by the solar cycle, which can change when detection will be very hard close to the sun.

This Thesis will focus on the flow of ISD into the inner heliosphere, how to calculate the impact velocity from ISD on a spacecraft and compare dust impact data with the calculations made. First an overview over the Solar Orbiter mission and spacecraft is given in the next chapter, along with a detailed description on the physics that makes dust impact detection possible. In chapter 3 a brief background of ISD is given and some of the most important dynamics controlling the flow of ISD in our solar system is explained. The theory for calculating impact velocities on Solar Orbiter is given in chapter 4, where it also mention interplanetary dust particles and discuss the pros and cons for different methods of calculating ISD flow. The figures and models produced for impact velocities are discussed in chapter 5, and chapter 6 includes dust impact data from the Solar Orbiter. Finally the conclusion is presented in chapter 7.

# /2

## ESA's Solar Orbiter Spacecraft

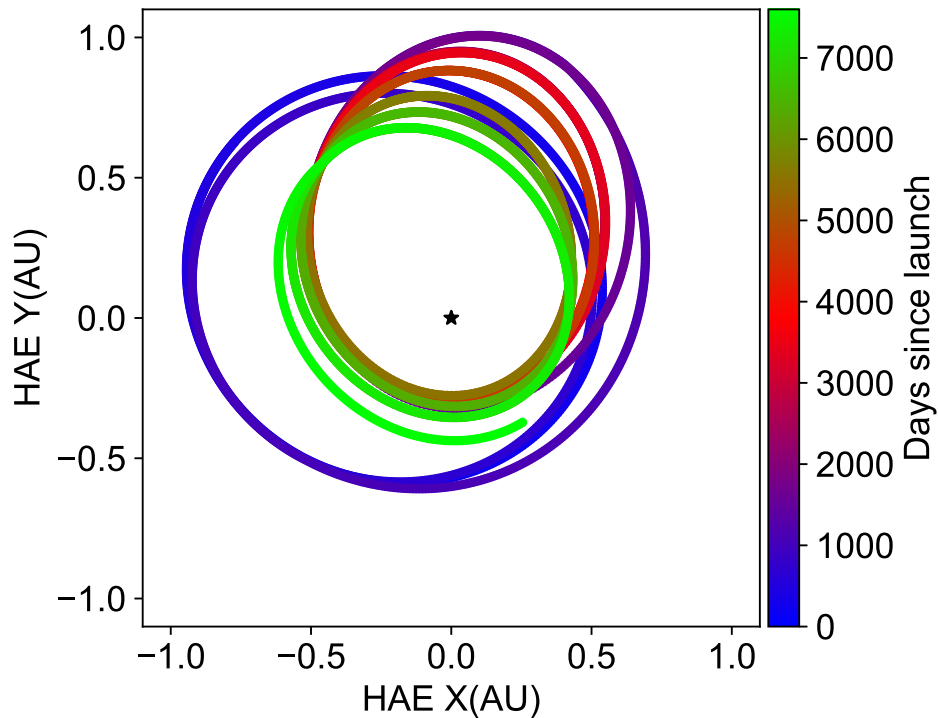
This chapter will introduce the main objectives and goals for the Solar Orbiter spacecraft, when it was launched from Earth, the planned orbit for the spacecraft and the duration of its mission. Then some discussion of the instruments aboard Solar Orbiter will follow, how they are used to detect dust in interplanetary space and how they might be used to detect ISD through a physical process called dust impact ionization.

### 2.1 Mission

#### 2.1.1 Launch and Orbit

On February 10th 2020 the Solar Orbiter was launched from Cape Canaveral Florida, aboard an Atlas V rocket. The mission is planned to last seven years, with a minimum distance of  $0.28 AU$  from the sun and during the mission it will increase its inclination to  $24^\circ$  above the ecliptic plane. The trajectory of Solar Orbiter has been meticulously planned to use multiple gravity assists from both Earth and Venus to get closer to the sun, saving a large amount of fuel. Mainly it will orbit the sun between  $1.2 AU$  and  $0.3 AU$ , but the trajectory is not fixed over time and the spacecraft will perform multiple maneuvers in space to change

its orbit. The first two orbits have an aphelion in between Venus and Mercury's orbit, but during the third orbit two gravity assists will be executed with both Venus and Earth to reduce the aphelion inside Mercury's orbit and change the right ascension of the ascending node with  $-30^\circ$ . For the next flybys the inclination will be slowly increased, and the aphelion will decrease, allowing the spacecraft to sweep a vast variety of regions in interplanetary space, see Figure 2.1 for the full orbit in the ecliptic plane.

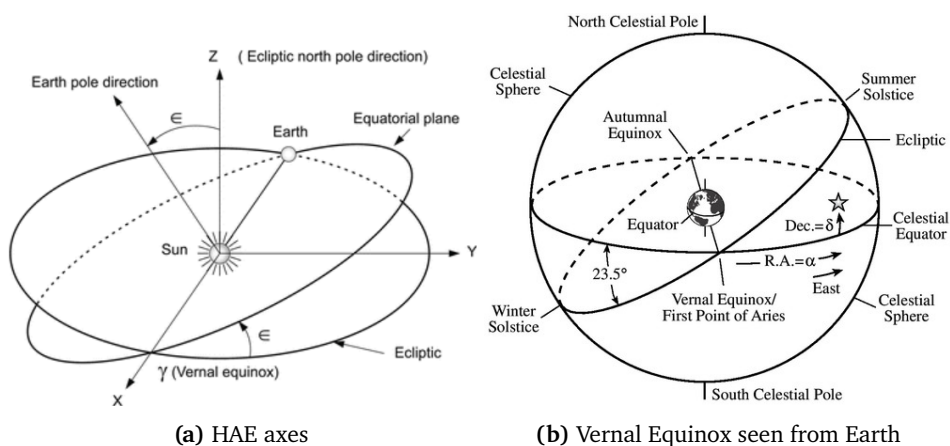


**Figure 2.1:** The planned trajectory of Solar Orbiter in the Heliocentric Aries Ecliptic model, from February 11th 2020 to November 20th 2030. Color indicates the days after launch, in the later parts the inclination increases which is not visible in the flat image

### Coordinate System

For interplanetary missions it is not very useful to maintain the same coordinate system used for Earth orbiting satellites, which uses Earth fixed relative vectors to determine the position of a spacecraft, rather we need a coordinate system that is fixed relative to the sun. The Heliocentric Aries Ecliptic (HAE) model is a system with the center of the sun at origin, the Z-axis points North from Earth's orbital plane around the sun (The ecliptic plane) and the X-axis points from the sun towards the vernal equinox, when Earth's equatorial plane crosses

the orbital ecliptic plane, and the Y-axis finishes the right hand rule. Figures 2.2 show how the first point of Aries goes through the X-axis when the sun aligns with the equatorial plane on Earth. The longitudes start from 0 on the X-axis and increases towards the Y-axis, Latitudes start on the ecliptic plane and increases towards the Z-axis to  $90^\circ$  and  $-90^\circ$  under the ecliptic plane. The exact time of the vernal equinox changes slowly over time due to an imperfect ratio of Earths rotation around its own axis to Earths orbital period around the sun, but for Solar Orbiter the J2000 Epoch; how the coordinate system was exactly on January 1st 2000, is used.



**Figure 2.2:** Heliocentric Aries Ecliptic model shown from the sun with its respective axes, and an image showing where the first point of Aries is located relative to Earth. Figures from (Suresh and Sivan, 2015) (Lang, 2000)

### 2.1.2 Mission Objectives

Solar Orbiter will as ESA said "Take the closest ever images of the Sun... Unraveling the mysteries of the solar cycle". Four scientific questions help break down the overall mission of Solar Orbiter (Mueller et al., 2013):

1. How and where do the solar wind plasma and magnetic field originate in the corona?
2. How do transients drive heliospheric variability?
3. How do solar eruptions produce energetic particle radiation that fills the heliosphere?
4. How does the solar dynamo work and drive connections between the Sun and heliosphere?

Scientists hope to answer some of these questions with high-resolution remote sensing of the sun, the solar wind and measuring the changes in the sun's magnetic field. With a higher inclination Solar Orbiter will also be able to explore the polar regions of the sun, a region not yet fully explored, but one which plays an important role in the activity of the sun and solar cycles.

While at the perihelion Solar Orbiter will image the surface of the sun in multiple bands<sup>1</sup>, but the most interesting in UV and EUV (Extreme Ultraviolet) a way to get never seen before super high resolution images of the sun. This allows small scales in features like solar flares and the corona to be studied with great detail.

Solar Orbiter will also help map the IMF and the polar magnetic regions of the sun in the later stages of the mission. If the mission gets an extension it might also be able to capture magnetic measurements of the sun during the changing of solar cycles.

## 2.2 Instruments

Included on the spacecraft is the Radio and Plasma Waves experiment (RPW) instrument (Maksimovic et al., 2020), consisting of three orthogonal antennas that can operate in multiple configurations together with the spacecraft body. A subsystem of this instrument is the Time Domain Sampler (TDS) which is designed to capture snapshots of electromagnetic waveforms from 200 Hz to 200 kHz. The RPW can measure magnetic and electric fields with a high temporal resolution, normally used for measurements of the solar wind, it can also detect tiny perturbations caused by dust particles impacting on the spacecraft or antenna.

### 2.2.1 The Physics of Dust Impact Ionization

When a dust particle impacts a spacecraft with a hypervelocity impact (20+ km/s), the dust particle instantly vaporises along with some of the surface material. The vaporisation generates a cloud of charged particles, ions and electrons, from the dust and surface material. The charge  $Q$  that is generated by the impact can be described by the mass of the dust particle  $m$  and its velocity  $v$  with the equation (eq 2.1). The exponents  $\alpha$  and  $\beta$  are found in laboratory

1. A band is a wavelength (ex. 300 nm) or range of wavelengths (ex. 630-650nm) of the electromagnetic spectra, usually used in remote sensing the number of bands describe the range of light or microwaves an instrument can measure.

experiments and vary based on the particle & surface materials,  $\xi$  represents a proportionality that is dependent on the impact parameters.

$$Q = \xi m^\alpha v^\beta \quad (2.1)$$

Based on the floating potential of the spacecraft, the charges can either be rejected or attracted back to the surface for recollection. The spacecraft will attract the charges of opposite charge and recollect them until they all either escape or the potential reaches its equilibrium again with the ambient plasma surrounding the spacecraft.

Spacecrafts are very often placed in locations in space where they are exposed to many surface charging processes, like strong UV radiation from the sun or the solar wind, this gives the spacecraft an electric potential. Zaslavsky (2015) gives expressions for the charge of a conductive object embedded in the solar wind (eq 2.2).

$$\frac{dq}{dt} = I_{ph} + I_{sw} + I_{sec} \quad (2.2)$$

Further the floating potential can be found from a steady state solution to the equation (eq 2.3) that depends on the local plasma parameters  $n_e$ ,  $T_e$  and  $v_e$ .

$$e^{-\varphi/T_{ph}} = \frac{en_e v_e S (1 + \varphi/T_e)^\alpha}{J_{ph0} S_{lit} (1 + \varphi/T_{ph})^\beta} \quad (2.3)$$

A fully detailed description of the charging process is beyond the scope of this thesis and is not critical for reading and comprehending the rest of the thesis. It is included to help explain the physics behind how an electric field antenna can measure dust flux.

## 2.2.2 Antenna signal shapes

Spacecrafts like the Solar Orbiter are equipped with electric field antennas that are used to take measurements of the solar wind and electric fields in space. Different configurations for measurements exist: “Electric field antennas can be operated as a dipole, where the voltage difference between two antenna booms is measured, or operated in a monopole mode, where the voltage difference between an antenna boom and the spacecraft body is measured” (Mann et al., 2019). When a dust particle impact occurs and charges are recollected, the potential difference will spike before it relaxes. If the charge generated is big enough the dust impact can be detected by the TDS as a visible spike in the signal and it will be saved as a “Triggered Snapshot WaveForm”.

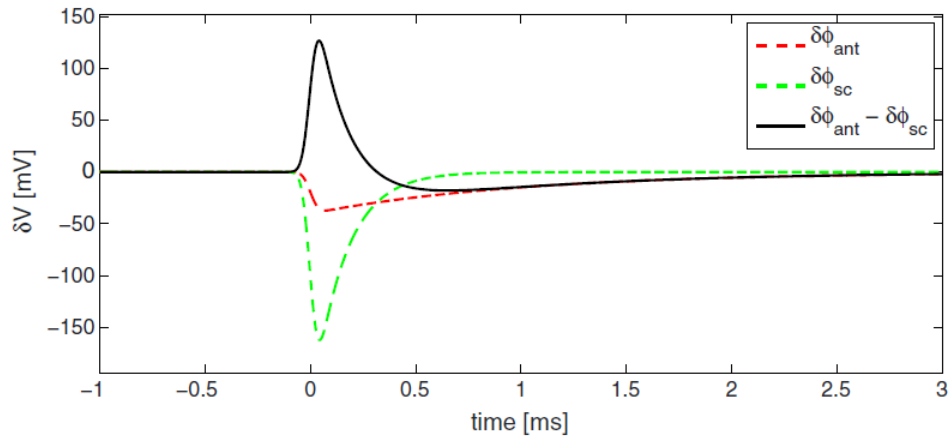
The shape of the signal depends heavily on the electric potential bias of the spacecraft before the impact. Electrons are faster than ions and will escape much

quicker, leaving a different signal if the spacecraft has a negative bias compared to a positive bias. The antennas will also sometimes collect charges from impact, and having a different geometry than the spacecraft body, they have different charging processes due to their thin cylindrical shape approximating a 1D charging process in the ambient plasma (Zaslavsky, 2015).

Zaslavsky (2015) assumes a positively charged spacecraft that collects electrons as a Gaussian function of time (eq 2.4).  $\Delta t$  is the typical collection timescale.

$$I_{dust}(t) = \frac{-Q}{\sqrt{2\pi}\Delta t} e^{-(t-t_0)^2/2\Delta t^2} \quad (2.4)$$

Simulating a dust impact that produces a charge of  $Q = 50$  pC creates a signal spike in the potential readings (Fig 2.3). Signals like these can be detected with algorithms that flag them as dust impacts, and by gathering the data each day we receive a list of the number of detected dust impacts per day. To detect ISD they need to produce enough charge for the snapshot to be triggered as a dust impact, this requires them to have sufficient impact velocity and mass to generate the charges. We therefore expect a higher flux of ISD detection in the regions with the highest impact velocity.



**Figure 2.3:** (Zaslavsky *et al* 2015) Simulation of signal caused by a dust impact. Red line indicates the antenna potential, green line indicates the potential of the spacecraft body that was hit by dust particle. Black line is the difference in potential measured between the spacecraft body and antenna in a monopole mode.



### 2.2.3 RPW Dust Detection Algorithms

The Time Domain Sampler (TDS) records waveform snapshots periodically at configured intervals of time, a wave detection algorithm is hard coded into the TDS to create a subset of snapshots called Triggered Snapshot WaveForms (TSWF) when the signal strength is above the noise level and has some presence of waves or dust impacts (Soucek et al., 2021). They are then given a quality factor by the algorithm and ranked in a queue for data transmission based on the quality ranking. If the TDS flagging algorithm classifies the TSWF as a dust impact it gets a quality ranking of  $Q = 65535$  (*dust*) and if  $Q < 65535$  the TSWF is flagged as *no dust*. Using this hard coded algorithm the TDS is able to classify most dust impacts correctly as *dust*, but higher classification accuracy can be achieved with machine learning techniques.

Kvammen et al. (2022) applied advanced machine learning techniques to TDS data and was able to detect more dust impacts than the hard coded algorithm, with an overall classification accuracy of  $96\% \pm 1\%$  compared to the on-board accuracy of 85%. This was achieved using a Convolutional Neural Network (CNN) trained on a data set of 2400 labeled observations and tested on 600 labeled observations.

The CNN does not use pre-defined feature extraction routines, but rather advanced chains of operations that optimizes the convolution filters automatically based on the associated labels in the training data. The 3-layer Fully Convolutional Network used is presented in Wang et al. (2017).

After training, the CNN was employed for detecting the number of dust impacts per day in data from June 15th 2020 to November 25th 2021. In Chapter 6 this data is analysed for ISD detection.



# / 3

## What is Interstellar Dust?

This chapter provides some background information on Interstellar Dust (ISD) and how it interacts with our sun through the heliosphere. In section 3.2 equations used for calculating the flow of ISD is explained together with figures illustrating how the flow changes based on intrinsic ISD parameters. The chapter ends with a discussion on including Lorentz forces into the calculation and why this paper will not include them.

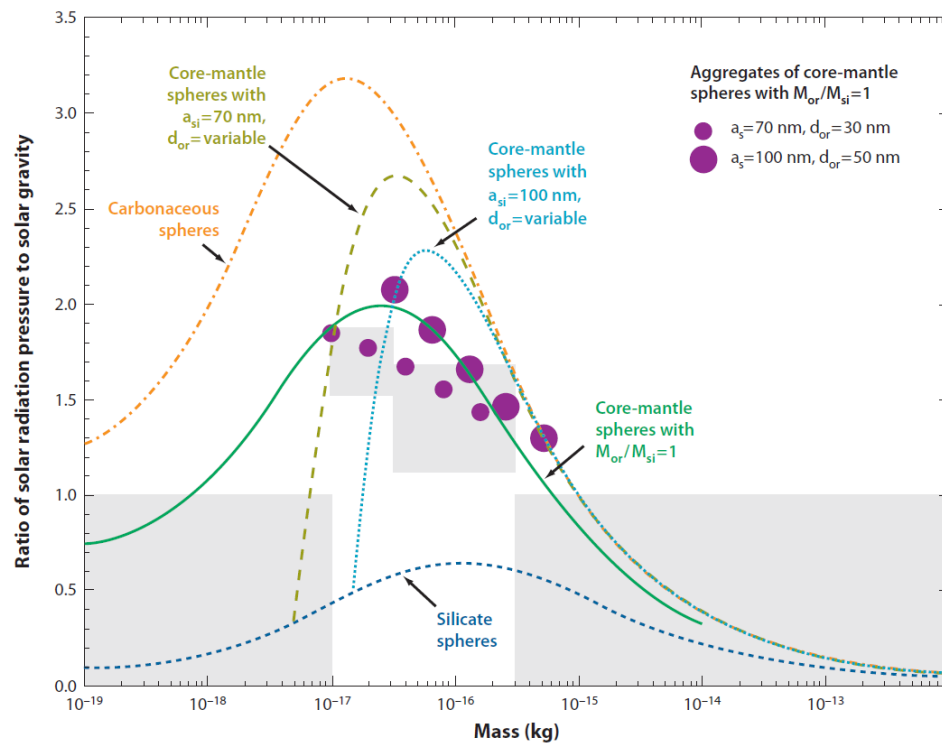
### 3.1 Background

Interstellar Dust (ISD) refers to small solid particles, ranging in size from nanometers to millimeters, that reside in the empty space between stars, i.e. the Interstellar Medium (ISM). Local Interstellar Clouds (LIC) in the galaxy contain a variety of gasses, dust particles and volatile material. When the solar system travels through some of these local dust clouds, tiny dust particles from the clusters with little to no charge can penetrate through the heliopause<sup>1</sup> and enter the solar system from roughly  $260^\circ$  HAE Longitude as ISD with a velocity of 26 km/s.

1. A border in space where the Solar Wind meets the Interstellar Wind and creates a bow shock front. Acting as a barrier towards charged particles travelling towards the solar system.

Here in the solar system they can be attracted by the sun's gravity and start accelerating further into the heliosphere, or the pure pressure from the sun's radiation can push the tiny dust particles away from the sun. The ratio between these two forces; the solar radiation pressure force and gravity of the sun is called the  $\beta$  ratio, and is a deciding factor for whether the ISD can travel deep into the heliosphere or whether they are deflected from reaching the vicinity of the Sun.

Factors that determine the  $\beta$  ratio of an ISD particle are the composition, the shape, and the bulk density. Changing the shape of a particle also changes the scattering effects from the sun's radiation (Mann, 2010), resulting in non linear  $\beta$  variation for the mass of a particular species of ISD (Fig 3.1).



**Figure 3.1:** Calculated ratios of radiation pressure to gravity for local interstellar cloud dust models explained in the figure and shown as lines and dots. The shaded areas depict the range of values that are consistent with flux measurements. (Mann, 2010)

### 3.2 Forces that determine the flow of ISD

ISD has through in-situ measurements been proved to have mass (Altobelli, 2004; Altobelli et al., 2006; Frisch et al., 1999; Krüger et al., 2019; Grün et al., 2000), therefore they must also interact with the suns gravitational field. The force felt by the gravity of the sun on an ISD particle can be described with the following equation

$$\mathbf{F}_G = -\frac{GM_\odot m_{ISD}}{|\mathbf{r}|^3} \mathbf{r} \quad (3.1)$$

Where  $G$  is the gravitational constant,  $M_\odot$  is the mass of the sun and  $m_{ISD}$  is the mass of an ISD particle,  $\mathbf{r}$  is the heliocentric position vector of the ISD particle.

The sun illuminates radiation radially outwards with the speed of light, the photons emitted carry momentum. If this radiation hits a surface the momentum from the photons will be transferred, cumulatively this can be described as a force that exerts an acceleration on the absorbed surface, we call this the Solar Radiation Pressure force. Small objects ( $\mu m$  sized) will be noticeably affected by this force, and we can describe the force from Solar Radiation exerted on an ISD particle as

$$\mathbf{F}_{rad} = \frac{A_p Q_{pr} S_0}{|\mathbf{r}|^3 c} \mathbf{r} \quad (3.2)$$

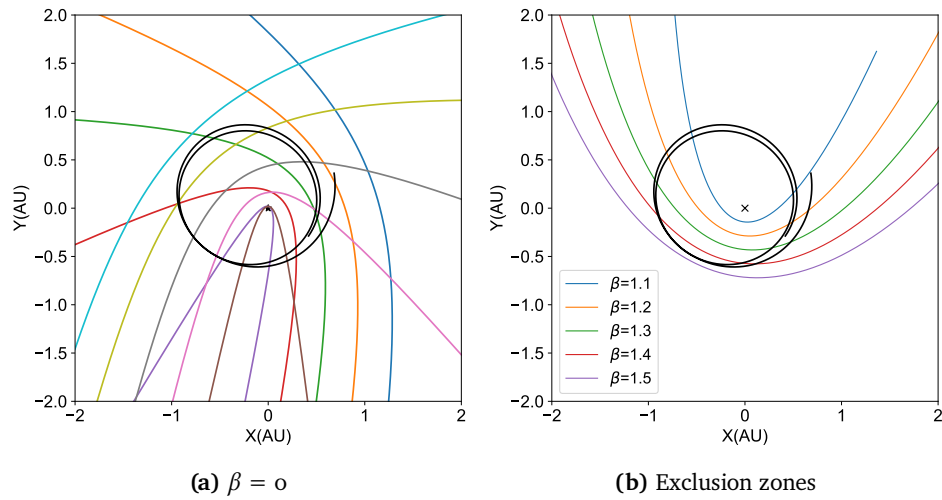
For a cross section  $A_p$  of the particle,  $Q_{pr}$  is the efficiency factor of the radiation pressure weighted by the solar spectrum and  $S_0$  is the solar flux weighted for the distance to the sun (Sterken et al., 2012).  $\mathbf{r}$  is again the heliocentric position vector of the ISD particle, and  $c$  is the speed of light.

Solar radiation pressure decreases with the distance of the sun squared, it spreads radially outward from the sun and will thus work against the force of gravity. The relationship between these forces are represented by  $\beta = \frac{F_{rad}}{F_G}$  as the ratio of the solar radiation pressure force over gravity. With this ratio, the sum of the forces acting on the particle is therefore.

$$\mathbf{F} = \frac{GM_\odot m_{ISD}(1 - \beta)}{|\mathbf{r}|^3} \mathbf{r} \quad (3.3)$$

The  $\beta$  determines the trajectories of ISD particles and therefore the ISD flux in the solar system, particles with  $\beta > 1$  are generally rejected from the sun

and create a density reduction in a cone-shaped volume behind the sun. This reduction is due to the radiation pressure force being greater than the gravity acting on the particle, and as they reach closer the sun they are repelled from the sun by this force. ISD with higher  $\beta$  are unable to reach certain areas behind the sun as a result of the Radiation Pressure Force being greater than the gravity, leaving behind a cone shaped volume excluding particles of a given  $\beta$ . We can draw a line in space illustrating the exclusion zone for ISD, seen in Fig 3.2b the different colors are the exclusion zones for different  $\beta$  values, ISD with those values can not go past those lines and from the figure it can be seen that ISD with  $\beta > 1.5$  can not reach the Solar Orbiter spacecraft.



**Figure 3.2:** XY Plane showing gravitational focusing of  $\beta = 0$  and exclusion zones for  $\beta > 1$ .

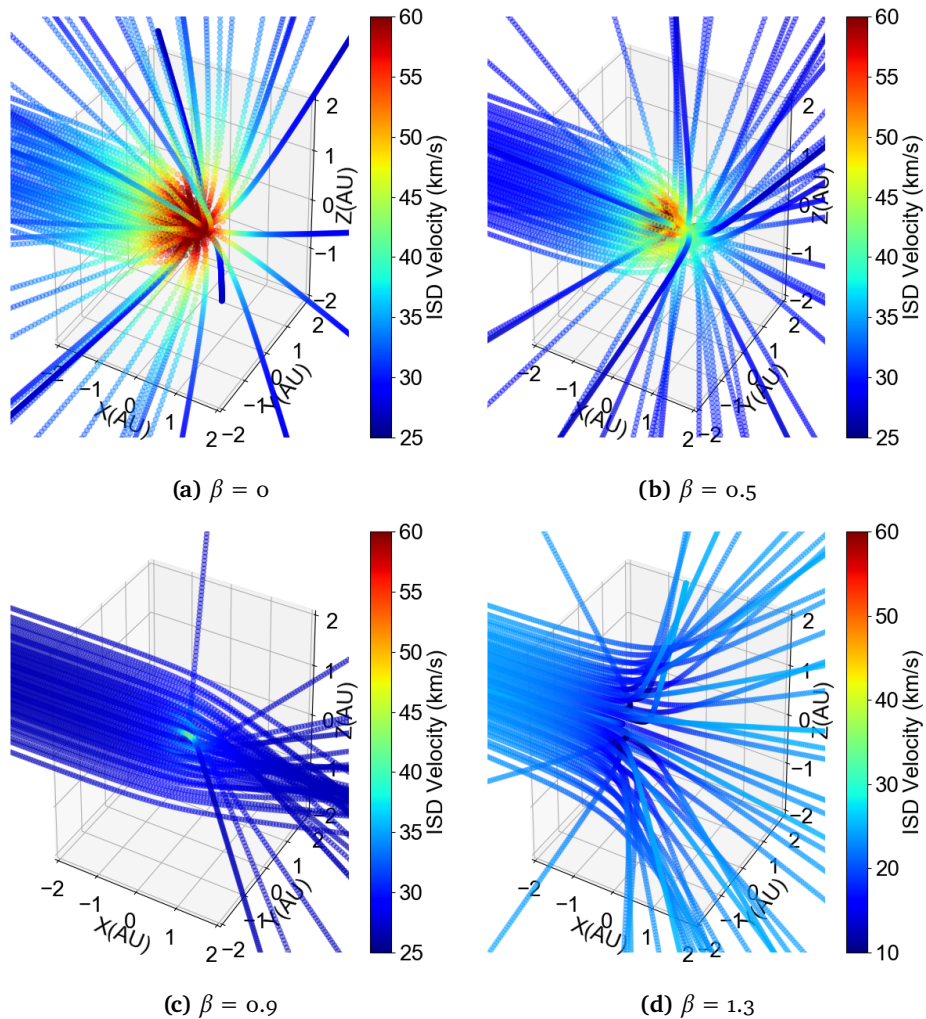
If  $\beta < 1$  the particles are attracted by the sun instead of being repelled, and focus behind the sun in the downstream direction increasing the density of particles in a line following the axis of the ISD downstream. Due to the hyperbolic nature of the interstellar dust orbits we see a big increase in the velocity as the particle gets closer to the sun, with the direction of travel changing as it flies past the sun we see a spread of particles behind the sun in multiple directions (see Figure 3.3). This makes it practically very difficult to accurately detect ISD particles through impact ionization in regions far behind the sun, due to exclusion zones for all  $\beta > 1$  particles behind the sun leading to a deficiency in ISD distributions, but also in the downstream region ISD with  $\beta < 1$  are no longer travelling parallel to the upstream direction and are spread out from the sun.

In the case when a particle has a  $\beta = 1$  it will not experience any acceleration from gravity as the solar radiation pressure force cancels it out. The particle

will continue to travel through the heliosphere with the same velocity until it is disturbed. Almost no particle is exactly  $\beta = 1$ , but models assuming this ratio can be used as a baseline for impact velocities on spacecrafts. This is very simple since we can cancel out gravity and solar radiation pressure forces, and only need to calculate the relative velocity of the spacecraft in the ISD upstream direction.

To simulate how ISD with different  $\beta$  ratios flow in the inner heliosphere, I implemented a time-dependent numerical solver in Python to calculate trajectories from a distance of  $50 AU$  with impact parameters from  $\pm 2 AU$  in height and width. This was done using a forward Euler's Method on a single ISD at a time and simply changing the initial values for each run, this was then repeated until 50 trajectories had been calculated, the code for this can be found in the appendix. Using a numerical calculation like the one just mentioned can be very computationally heavy when you want precise trajectories because it would require very many tiny steps, which in turn creates a large amount of data points that require more memory.

One way of reducing computational time is to reduce the number of steps the trajectory is calculated for, and interpolate in between steps of interest. Using fewer steps may not preserve the conservation of energy, therefore it can be wise to take values with a pinch of salt before relying on them for further calculations. For purposes where only visual illustration is the motivation, this will not be as big of a concern. In Figures 3.3 the changing of ISD flow based on the  $\beta$  value is visualised in 3D with the method just described. For other calculations done using IVP solvers, as in section 4.3, the number of steps were increased for this effect to minimal.



**Figure 3.3:** Numerically calculated ISD velocities starting from 50 AU away with impact parameters containing a width and height of 2 AU. Note that the color axis for  $\beta = 1.3$  goes lower than the others to display the decrease in velocity when the ISD reaches close to the sun

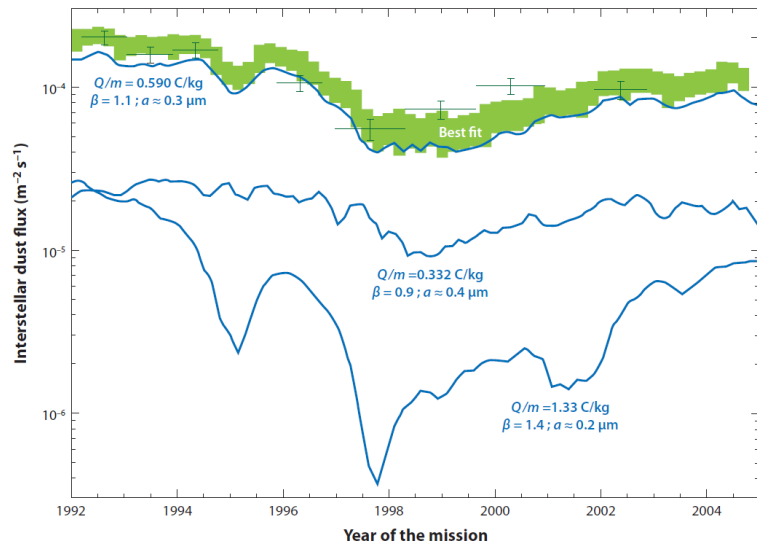
### 3.2.1 Lorentz Forces

When the dust particles travel in the solar system they can pick up charges from moving through the ambient solar wind, and they will also experience photo ionization from the sun's UV radiation. The electron emission through photo-ionization is more prevalent than collection of electrons, this results in a net positive charge on the dust particles. And when charged particles move through the IMF they will experience forces from the magnetic and electric fields, these are called Lorentz forces. The force is dependent on the relation

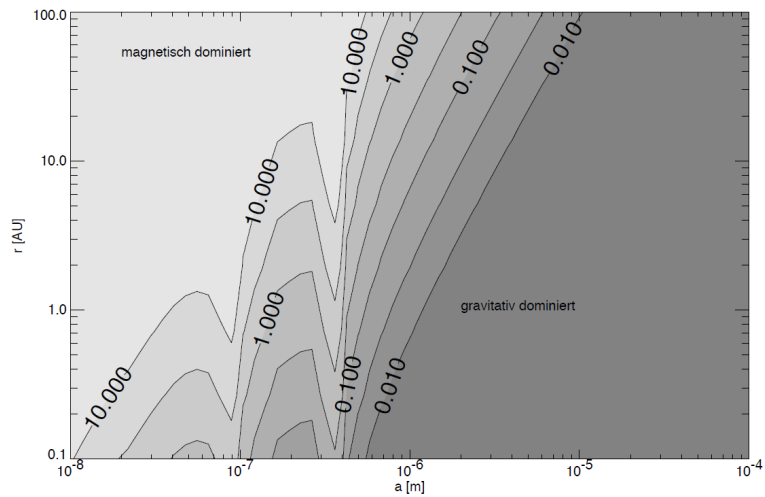


between the particle charge and its mass ( $Q/M$ ), its velocity relative to the solar wind and the strength of the magnetic field. This section will look at the two main effects this force has on ISD, focusing and defocusing particles from the ecliptic plane.

Research done on the effect of Lorentz forces on the flow of ISD has shown that based on the Sun's solar cycle it will either focus or defocus particles from the ecliptic plane (Sterken et al., 2012) see figures 3.5, but this effect is mostly dominant on smaller charged particles. The Focusing and defocusing periods changes with every solar cycle, with the last focusing period being around 2010, and Solar Cycle 25 starts in a defocusing period and will turn towards a new focusing period sometime in 2029+ (Fig 3.6). Bigger particles can have a smaller charge to mass ratio and be dominated mostly by gravitational and solar radiation forces, these particles can be assumed to be less affected by the Lorentz forces and travel along previously established trajectories (Sterken et al., 2012). For the sake of simplicity this paper will not consider Lorentz forces affecting flow of ISD, and will therefore only be looking at the possibility for detecting ISD that have a sufficiently small charge to mass ratio be gravitational dominant, and with small flux variation during the changing of solar cycles, from (Landgraf, 2000) it is found that ISD with a charge to mass ratio of  $Q/M = 0.332C/kg$  has only tiny variation in the dust flux.

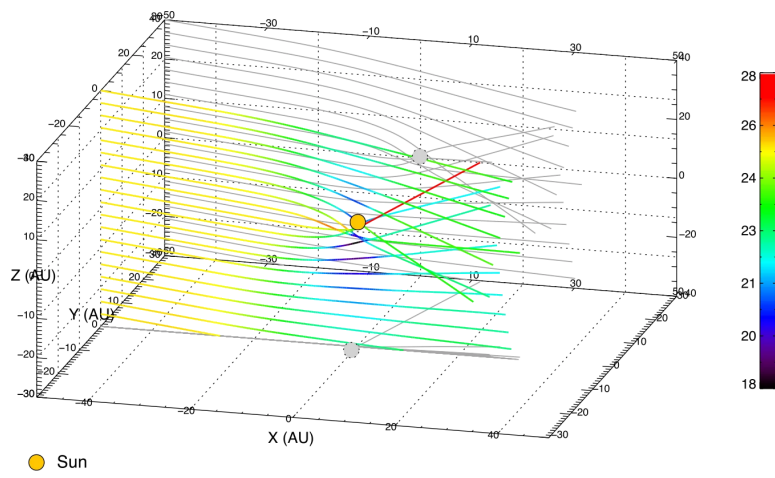


(a) ISD flux from 1992 to 2004 along Ulysses orbit with calculated fluxes for different  $Q/m$  ratio. (Mann, 2010) adapted from (Landgraf, 2000).

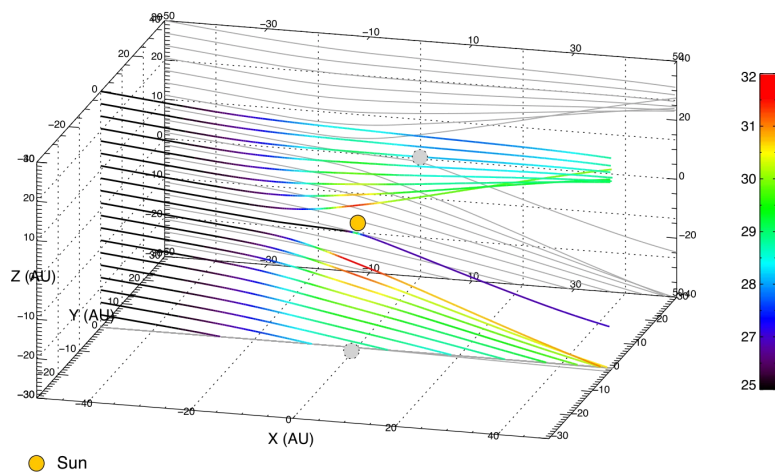


(b) Ratio of Lorentz force to effective gravitation. (Sterken et al., 2012)

**Figure 3.4:** ISD can be found to exist with parameters where Lorentz forces does not heavily dictate the flow.

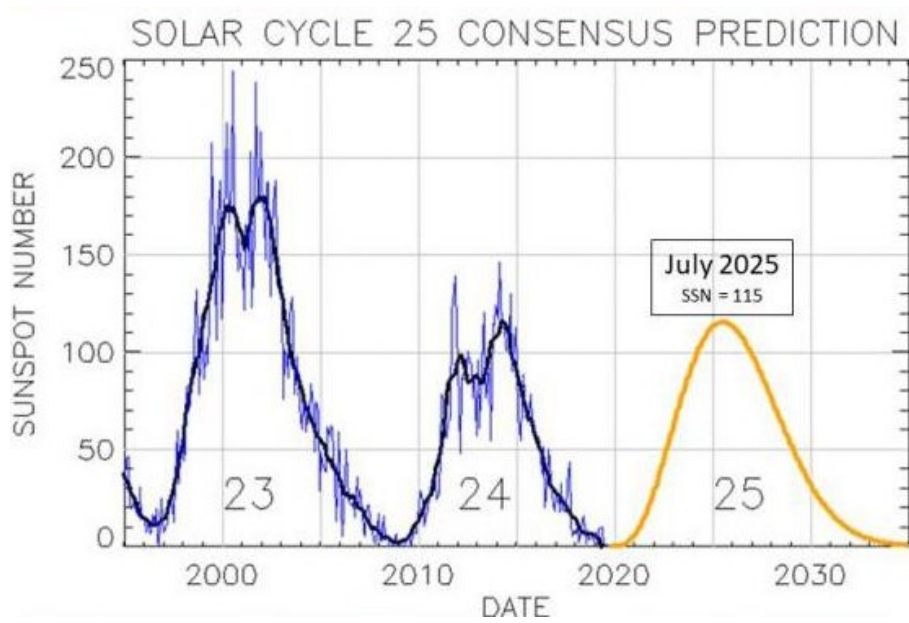


(a) Focusing



(b) Defocusing

**Figure 3.5:** Focusing and defocusing of charged ISD, from (Sterken et al., 2012). The particle trajectories are for  $\beta = 1$  and  $Q/m = 0.5$  C/kg starting at 50 AU away, the start year is 1990 on 3.5a, and 2000 on 3.5b



**Figure 3.6:** Solar Cycle 25 predictions from NOAA from 1995 to 2035, published December 9th 2019

# /4

## Methods for Calculating Impact Velocity

This chapter presents equations that find analytical solutions for calculating impact velocities from ISD on a spacecraft. Included in the sections are discussions on complications and limitations certain equations might have and how to deal with them. Section 4.2 mentions Interplanetary Dust Particles and how they can complicate dust detection from ISD, and in section 4.3 it is debated on the use of numerical calculations for impact velocities.

### 4.1 Using Equations to Calculate Impact Velocity

To find the impact velocity of an ISD particle, we start by calculating the parameters for an ISD particle that passes through a known coordinate and solve the initial conditions that fulfill a trajectory through the known coordinate. For this the coordinates and the equations need to be in the same coordinate system. While the HAE model is often used for interplanetary spacecrafts, the equations needed are not always in the same coordinate system. A decision has to be made whether to rotate the equations to your preferred coordinate system, rotate to an ISD fixed coordinate system, or rotate the orbits and ISD flow to the coordinate system of the equations. Changing back and forth between coordinate systems become very tricky and requires a firm notation

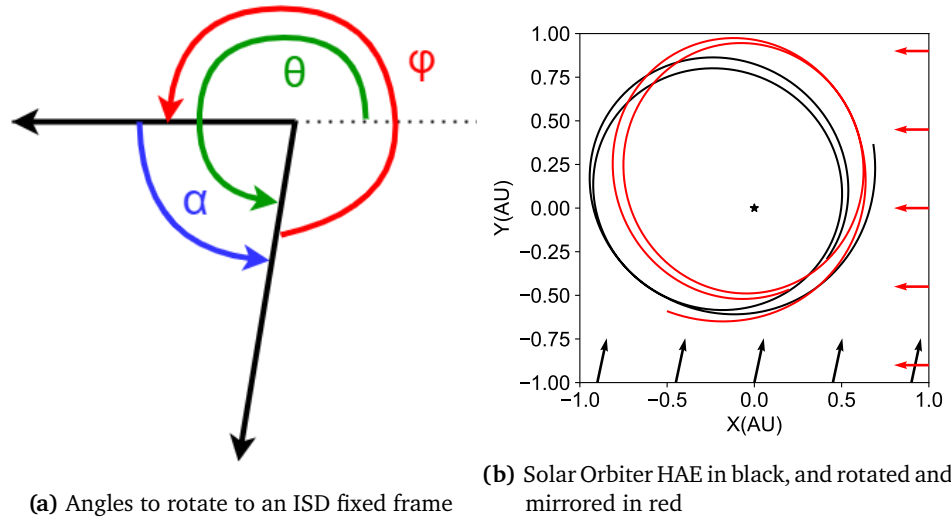
of the order of calculation to avoid miscalculations.

Let  $\theta = 260^\circ$  be the direction of the ISD upstream, if I want to rotate our coordinate system so that the upstream direction is fixed to the X-axis ( $180^\circ$ ) we can see from Fig 4.1a that  $\varphi$  must be as eq 4.1. To rotate back to HAE I need to rotate everything with  $\alpha$  degrees (eq 4.3).

$$\varphi - \pi = 2\pi - \theta \quad (4.1)$$

$$\varphi = 3\pi - \theta \quad (4.2)$$

$$\alpha = \theta - \pi \quad (4.3)$$



**Figure 4.1:** Transforming the coordinate system from HAE to an ISD fixed system will change the orbit of the spacecraft as we see in Figure 4.1b.

Start by making an ISD fixed coordinate system where it is assumed that ISD particles are injected into the solar system parallel to the X-axis with an initial velocity  $V_\infty = 26$  km/s and with an impact parameter  $b$  in the Y-axis. With  $G$  being the gravitational constant,  $M$  the mass of the sun, the variable  $\mu = GM(1 - \beta)$  is chosen as the effective gravitational parameter (Altobelli, 2004). Because Solar Orbiter is mostly confined within exclusion zones of  $\beta = 1.5$  and under (Fig 3.2b), for the following equations the assumption that  $\beta < 1$  is made. ISD enter with a velocity greater than the escape velocity, so they follow a hyperbolic orbit where the eccentricity has to be greater than 1. The

eccentricity and deviation angle are given as:

$$e = \sqrt{\frac{b^2 V_\infty^4}{\mu^2} + 1} \quad (4.4)$$

$$\frac{\delta}{2} = \sin^{-1} \left( \frac{1}{e} \right) \quad (4.5)$$

$b$  is the impact parameter of the particle,  $V_\infty$  is the velocity at infinity (entrance velocity) and  $\delta$  is the deviation angle, parameters are illustrated in the Figure 4.2 In polar coordinates  $(r, f)$  the equation for a hyperbola, with  $p = \frac{b^2 V_\infty^2}{\mu}$ , is:

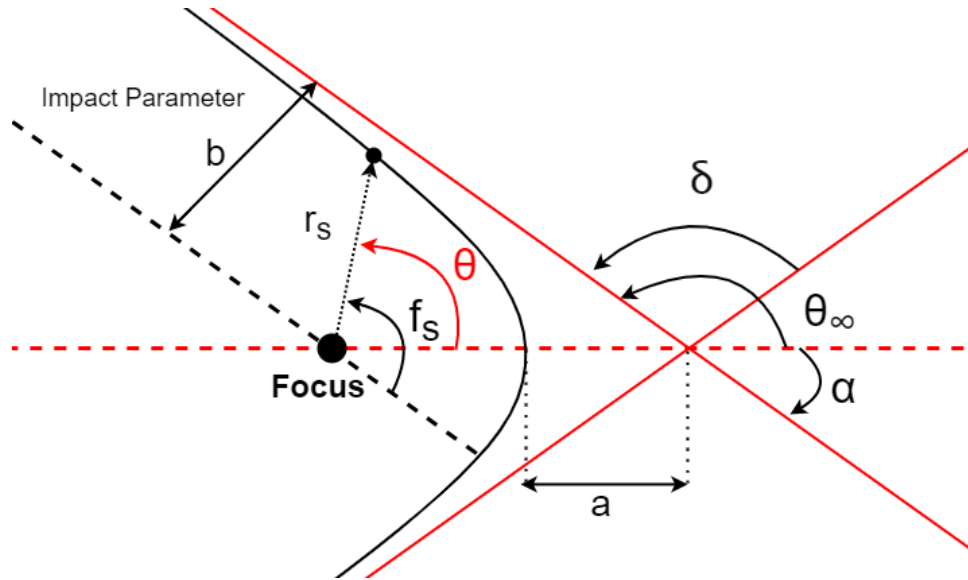
$$r = \frac{p}{1 + e \cos \left( f - \frac{\delta}{2} - \frac{\pi}{2} \right)} \quad (4.6)$$

If we then take a spacecraft position in polar coordinates with the Sun at the origin  $(r_s, f_s)$  an ISD particle can reach that position if the equation 4.6 is true at those coordinates. This approach starts by taking every spacecraft position and verifying if an ISD particle can hit the spacecraft and solving for the parameters if a solution is true. To solve the hyperbolic equation the impact parameter  $b$  (eq 4.7) for an ISD with a given  $\beta < 1$  has to be found first, there will always be two solutions with opposite signs from the fact that the sun can slingshot a particle after the focusing point. For simplicity this thesis will remain to only use impact parameters with the same sign as the spacecraft Y coordinate in the new coordinate system, this reduces the equation to look at incoming particles instead of outgoing particles.

$$b_{1,2} = \frac{r_s \sin(f_s)}{2} \left[ 1 \pm \left( 1 + \frac{4\mu}{V_\infty^2 r_s (1 + \cos(f_s))} \right)^{\frac{1}{2}} \right] \quad (4.7)$$

Note that the equation to calculate the impact parameter does not know which direction the ISD is coming from, and it might not be in your desired direction. In my case the coordinate system had to be mirrored around the Y axis to calculate ISD impact parameters from the correct direction, but it is important to remember to mirror back again afterwards.

From this the eccentricity and deviation angle can be easily calculated, which gives us the full equation for the hyperbolic trajectory. All the parameters to describe the trajectory of an ISD particle can now be calculated, so the next step



**Figure 4.2:** Hyperbolic parameters  $\delta$  and  $\theta_\infty$  describe the sharpness of the hyperbola based on the angle between the asymptotes. They are dependant on the eccentricity of the hyperbola and show deflection angle of the trajectory. Note that the semi-minor axis is equal to the impact parameter, the shortest distance from the focus point to the incoming asymptote. The position  $(r_s, f_s)$  is the spacecraft polar coordinates an ISD is needs to reach to impact.

is to find the ISD velocity. To find the velocity of the particle at the spacecraft position we take use of the Vis-Viva equation 4.8

$$V^2 = \mu \left( \frac{2}{r_s} - \frac{1}{a} \right) \quad (4.8)$$

where  $a = -\frac{b}{\sqrt{e^2-1}}$  using the impact parameter as the semi-minor axis.

The Vis-Viva equation only gives us the orbital speed of the ISD particle at a position  $(r, \theta)$  but to calculate the impact velocity we require the relative velocity between the spacecraft and the particle. When  $\beta < 1$  the trajectory curves when it's distance to the sun decreases, so the angle between the spacecraft and the particle will not be constant through the spacecraft orbit and will change the scale of the relative velocity at impact.

The angle of the velocity  $\varphi$  (Fig 4.3a) relative to the perpendicular to the radial direction is given by the equation (Bate et al., 2020):



$$\tan(\varphi) = \frac{e \sin(\theta)}{1 + e \cos(\theta)} \quad (4.9)$$

This is an important equation that allows us to get the impact vector for ISD, as is probably known a vector needs both a scalar and direction. A different method could be to find the angle of velocity as the direction from a small change in position  $\delta p = v$ .

We find  $\theta$  from conic parameters of a hyperbola used to describe the orbital elements from the semi-major axis, semi-minor axis and the eccentricity. In the Figure 4.2 we assume an ISD particle has already reached a spacecraft position  $(r_s, f_s)$  and we need to find the relation between the  $\theta$  angle and spacecraft angle  $f_s$ .

In the coordinate system I have previously used, I rotated to an ISD-frame with the downstream direction being parallel to the X-axis travelling in the positive direction. The X-axis of that system is now parallel to the incoming asymptote shown in red in the Figure 4.2, with the impact parameter being the distance from the asymptote to the sun.

Looking at the figure we can see that the relationship between the coordinate system is found by the following equations

$$f_s = \theta + \alpha \quad (4.10)$$

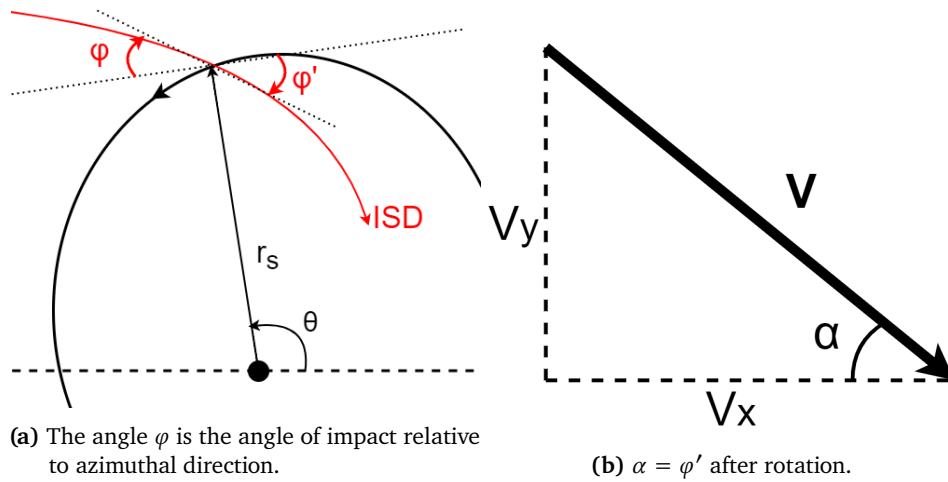
$$\alpha = -(\theta_\infty - \delta) \quad (4.11)$$

$$f_s = \theta - \theta_\infty + \delta \quad (4.12)$$

The two angles  $\theta_\infty$  and  $\delta$  are given by the eccentricity:

$$\theta_\infty = \cos^{-1} \left( \frac{1}{-e} \right) \quad (4.13)$$

$$\delta = 2 \cdot \sin \left( \frac{1}{e} \right) \quad (4.14)$$



Then we finally find the angle to be:

$$\varphi = \tan^{-1} \left( \frac{e \sin(f_s + \theta_\infty - \delta)}{1 + e \cos(f_s + \theta_\infty - \delta)} \right) \quad (4.15)$$

The velocity components can then be calculated:

$$V_x = V \cdot \cos(\varphi) \quad (4.16)$$

$$V_y = V \cdot \sin(\varphi) \quad (4.17)$$

When trying to solve for impact velocities analytically you are limited by the number of equations, and the approach taken to find a solution. Working backwards I start with a set of coordinates for Solar Orbiter  $(x, y)$  in HAE and transform them into the ISD-fixed polar coordinate system  $(r_s, f_s)$ , then I input the new coordinates into equation 4.7 to get the impact parameters for an ISD trajectory that intersects the coordinates. With the impact parameter  $b$  the eccentricity (eq 4.4) and deviation angle (eq 4.5) can be solved. With the impact parameter and eccentricity the Vis-Viva equation 4.8 gives the Velocity of the ISD particle at the spacecraft coordinates  $(r_s, f_s)$ , and the angle of the velocity is calculated with equation 4.15. After rotating the velocity vector back to a HAE coordinate system, the impact velocity is calculated as the relative velocity between Solar Orbiter and the ISD particle  $V_{SO} - V_{ISD}$ . It is recommended to do the calculation in the steps as explained, but if so desired the entirety can be inserted into one very big equation, but this is not recommended, see appendix A.2 for a full equation.

## 4.2 $\beta$ -meteoroids

Most of the dust observed within  $1 AU$  are interplanetary dust particles (IDPs) produced in the inner heliosphere. IDPs can come from multiple sources like comet trails, fragments produced by asteroid or meteoroid collisions, and fragments that are ejected with the gas emission from comets. They form the highest number density near the ecliptic because their orbits have similar low inclinations to the ecliptic as most comets and asteroids. The IDPs experience the same forces as ISD, i.e. primarily the solar gravitational force and radiation pressure force. Because the Poynting-Robertson effect slows down particles in bound orbits, their orbital angular momentum decreases over time, allowing us to find more of them closer to the sun. Many of the interplanetary dust particles with nanometer to micrometer size are in hyperbolic orbits, those are often called  $\beta$ -meteoroids, and they can be observed moving away from the sun.

In the orbits closer to the sun the IDPs have a very high velocity, hence we can expect that close to the sun there will be considerably higher dust fluxes than at further distances. This results in high collision rates, high collision velocities and small sizes of the fragments that form when the dust particles collide with each other or with larger meteoroids. These smaller fragments have a high  $\beta$  ratio ( $>1$ ) so that they are repelled from the sun which causes them to travel radially outwards from the sun. The dust-dust collision rate is inversely proportional to the distance to the sun (Mann et al., 2019), hence most of the observed  $\beta$ -meteoroids form close to the sun.

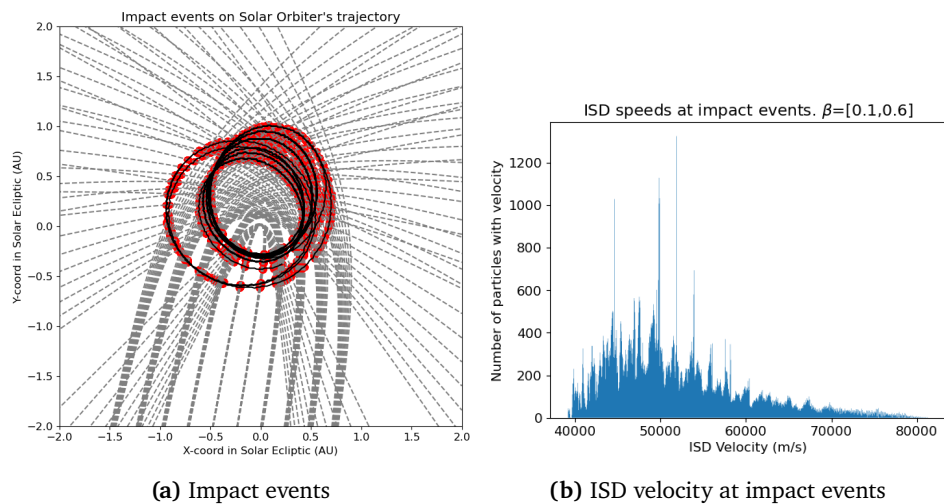
Out of all the different kinds of IDPs this paper will only look at  $\beta$ -meteoroids that travel radially outwards from the sun. For the rest of this paper when IDPs are mentioned, it will only consider  $\beta$ -meteoroids. We assume they have a constant velocity of  $80 \text{ km/s}$  radially outwards from the sun “because this was found to be a good approximation for dust distances larger than  $0.3 AU$ ” (Henriksen, 2020).

## 4.3 Alternative Method using Initial Value Problem Solvers

One approach for calculating impact velocities is by numerically calculating the trajectories of ISD, this was the approach I took in my project paper (sec A.1). One of the aspects that can make a numerical approach intriguing is that it is very straight forward and easy to understand. Time evolutions are easy to make numerically and can help illustrate the overall dynamics in a system without

searching for equations. The accuracy of a numerical calculation depends on the numerical method and the size of the time steps used. Increasing the number of steps can increase the accuracy but requires longer computation times, which might be undesirable.

In my project paper the ISD trajectories were calculated one at a time with different initial values for a range of values. To try and find impact velocities I first implemented a check to see when the trajectory was close enough to the spacecraft orbit to be considered an impact, and then find the velocity of the ISD at that point. This was done by first taking the ISD position and vector subtracting the entire orbit of Solar Orbiter to see if they would impact, if the distance was under a certain threshold a red marker would be placed at that location in the trajectory to indicate an impact. After an impact had been marked, the ISD velocity would be saved in a list and after all trajectories had been calculated they would be binned in a histogram, as seen in the figures 4.4.



**Figure 4.4:** Numerically calculated impact events for Silicate Spheres, with a histogram showing the ISD velocity at the marked locations. (Henriksen A. 2021)

The method used in my project paper is not recommended, it is not straight forward to implement and is computationally not efficient. A better option would be to calculate the trajectories of multiple particles at the same time, and iterate the coordinates of the spacecraft orbit to find an ISD particle in close vicinity instead of checking the entire particle trajectory.

An improvement on my previous work was attempted for this Thesis, to get better numbers on ISD in the environment around the Solar Orbiter spacecraft. With a Forward Euler Method IVP solver the trajectories of ISD are calculated

from the sum of forces that act on it. Using Runge Kutta 45 the accuracy was increased and the flow simulation could be expanded. Specifically I simulated the flow of ISD in 3 dimensions and then applied restrictions on the ISD trajectories to only exist within the borders of Solar Orbiter's maximum and minimum position in all 3 axes, so from roughly 1.5 AU to 0.3 AU in the XY plane and  $\pm 0.15$  AU in the Z plane. For the simulation it was calculated with a smaller number of steps (200-400), then after the restrictions were applied I increased the steps through 3rd order interpolation to save computational time, but this might will have an effect on the accuracy of the particles energy.

## 4.4 Discussion

Finding a method for calculating the impact velocity is not trivial, especially finding out how to calculate the components for the ISD velocity vector. Some equations for azimuthal and radial velocity were found, but produced inconsistent results and were quickly discarded, another method involved taking the time derivative of the position but since they were in polar coordinates and the spacecraft components were in cartesian it would involve a lot of coordination transformations that were not desirable. Before the angle of velocity equation was found, I simply made a vector from the change in coordinates from one data point to the very next, and this seemed to work for a while. The problem was if there was no next data point then it would create a vector from the coordinates of the last element to the coordinates of the first element in the array, due to how python handles arrays.

A problem will appear when using eq 4.15, this stems from the fact that in the focusing region behind the sun, ISD trajectories will cross paths and solutions for impact velocities will exists for multiple directions and only one impact velocity can be used at a time. Only incoming trajectories are able to be calculated and as shown in the Figure 5.1b where the red arrows indicate ISD vectors and on the top right they change direction.



# /5

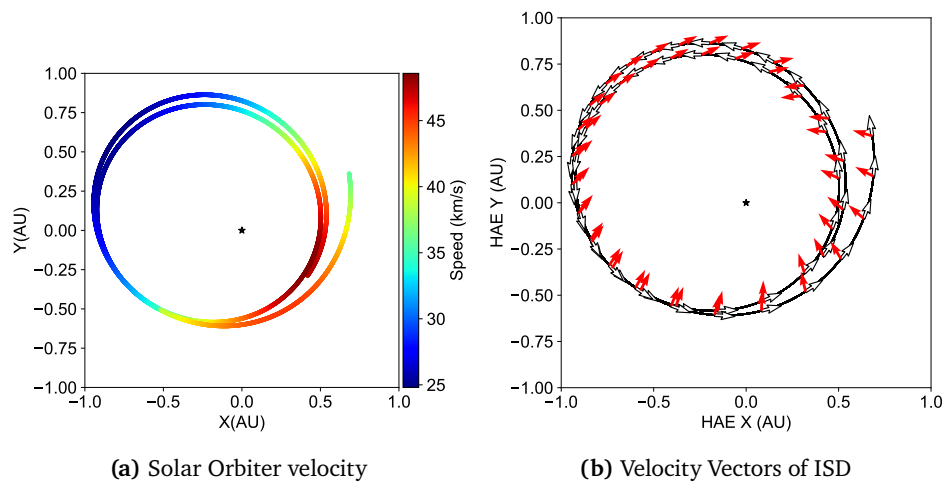
## Impact Velocities and ISD Dynamics

In this chapter the equations and methods discussed in chapter 4 are applied, and models are presented for impact velocities on Solar Orbiter by both ISD and IDPs. The impact velocities are compared together and discussed in the end of the chapter. Results from calculations made using the alternative IVP solver are also presented in section 5.2.

### 5.1 Impact Velocities

To get an understanding of the areas Solar Orbiter can potentially detect ISD in, I calculated the impact velocity on the Solar Orbiter spacecraft from ISD with multiple different  $\beta$  values to illustrate the importance this parameter has in the dust detection. Combining the velocities with measured charges from the RPW it is possible to estimate the mass of ISD from the equation 2.1, but this is outside the scope of this thesis and no such estimations has been made. Theoretical impact velocities are calculated using the analytical equations in section 4.1 for incoming ISD, on the trajectory of Solar Orbiter from June 15th 2020 to November 25th 2021. We see in figures 5.1 the prograde orbital velocity for Solar Orbiter with the intensity of the color referring to the colorbar as velocity in km/s, and the relation between velocity and heliocentric distance

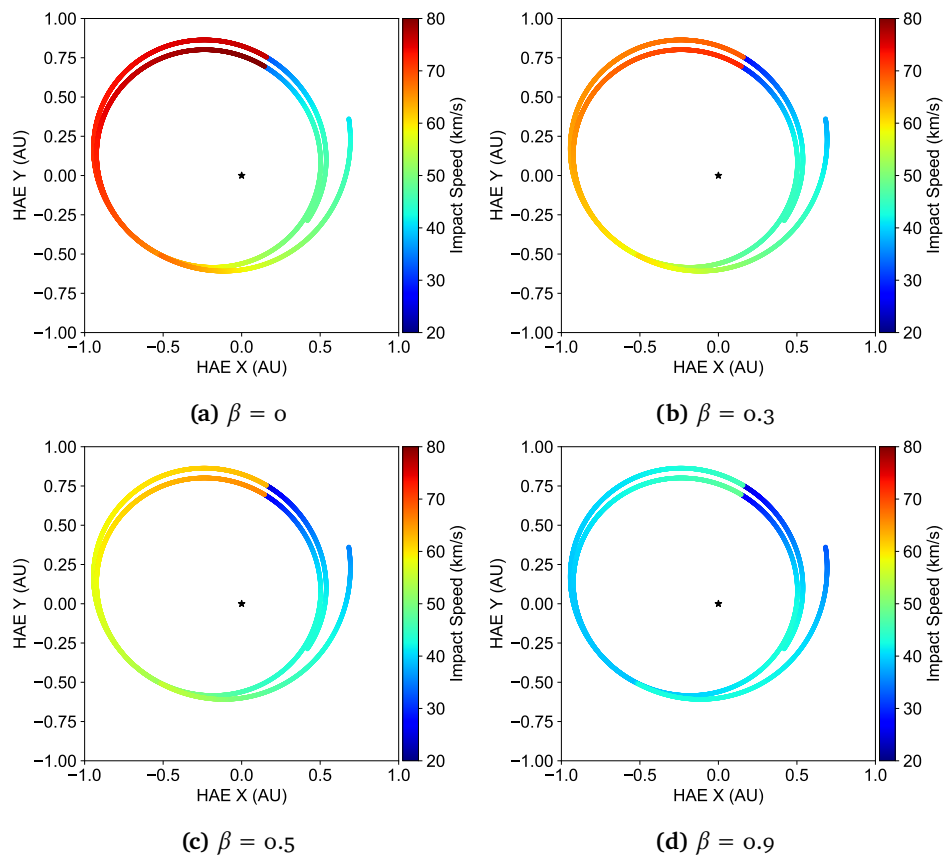
is later shown in Fig 6.3a where we see what we would expect from Keplers laws, that the highest velocity is when the spacecraft is closest to the sun, and lowest furthest away.



**Figure 5.1:** The speed of Solar Orbiter is shown in the left image with the color indicating the orbital velocity of the spacecraft, on the right impact vectors on Solar Orbiter from ISD is shown. Red arrows indicate the vectors from ISD and white arrows represent vectors from Solar Orbiter

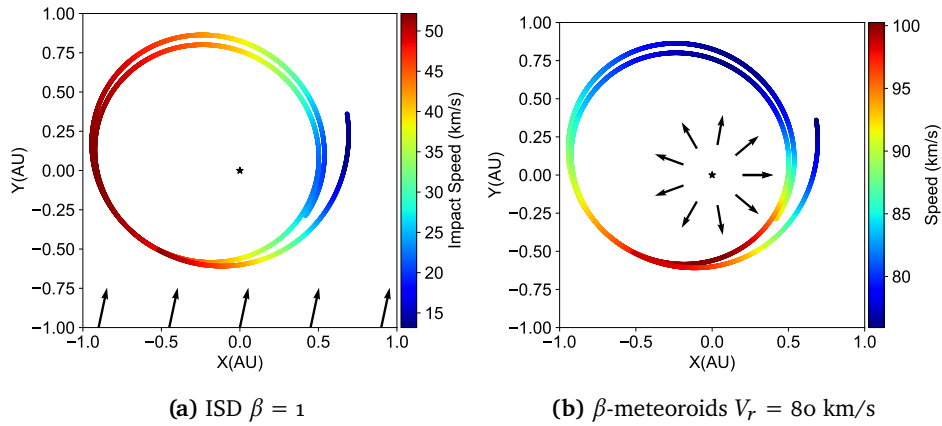
From the figures 5.2 showing the impact velocity from ISD with different  $\beta$  values on Solar Orbiter, we can see that the focusing region in the downstream direction has a very high impact velocity, which continues when the spacecraft is travelling parallel to the upstream direction all the way down to when the spacecraft is perpendicular to the upstream direction and starts travelling in the same direction as the ISD. Lower  $\beta$  value ISD reach higher impact velocities due to the higher acceleration they feel from gravity, but even for  $\beta = 0.9$  the impact velocity can get as high as 40-50 km/s.



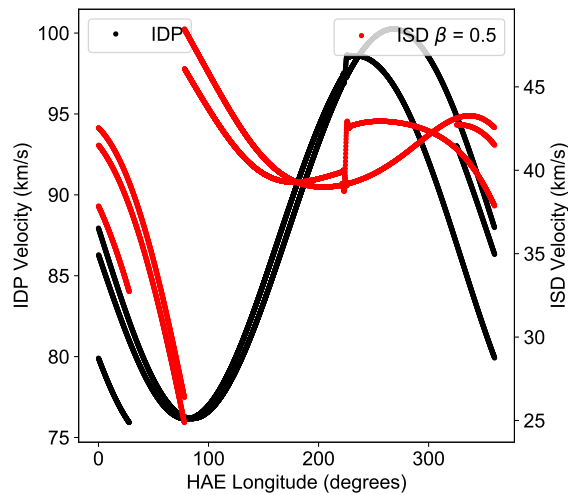


**Figure 5.2:** ISD impact velocities analytically calculated for different  $\beta$  values. The Impact velocity has only been calculated for incoming particles, leaving a big edge in the focused region due to the separation of two different directions the ISD is travelling and which one the impact velocity has been calculated for.

Comparing the impact velocity from ISD to IDP we can see that IDPs mostly contribute in a lower region of the orbit between  $260^\circ$  and  $300^\circ$  longitude (Fig 5.3b), while ISD have a higher flux from roughly  $90^\circ$  to  $270^\circ$  longitude (Fig 5.3a).



**Figure 5.3:** Impact Velocities for ISD with  $\beta = 1$  is shown on the left and IDPs are shown on the right with a constant radial velocity of 80 km/s



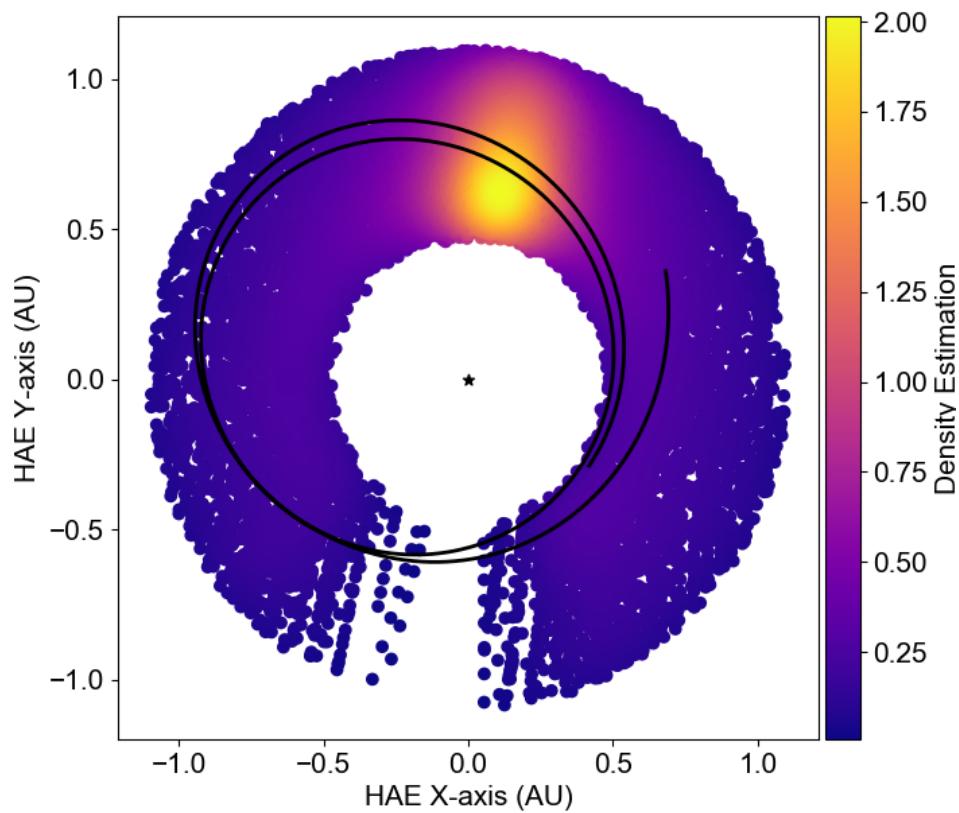
**Figure 5.4:** Impact Velocity of ISD and IDPs compared on the same orbit. Red line represent ISD impact velocity for  $\beta = 0.5$ , black line represent IDPs with  $V_r = 80$  km/s

## 5.2 Information of ISD in the inner heliosphere

Using the alternative method with an IVP solver (Initial Value Problem) as shown in section 4.3, I was able to illustrate other parameters in the region of space we find Solar Orbiter. I used a kernel density estimation (KDE) to make a probability density of where ISD can be found in the inner heliosphere, in Figure 5.5 the intensity of the color represents the probability density for a

population of ISD. As one might easily deduce, the focusing in the downstream direction is the cause for an increase of population of ISD. This region also coincides with high impact velocities shown in the figures 5.2.

The KDE can be very useful for determining where the highest likelihood of finding ISD can be, and because the focusing region does not have a high impact velocity contributed by IDPs (Fig 5.3b) it will be easier to separate ISD from IDP.

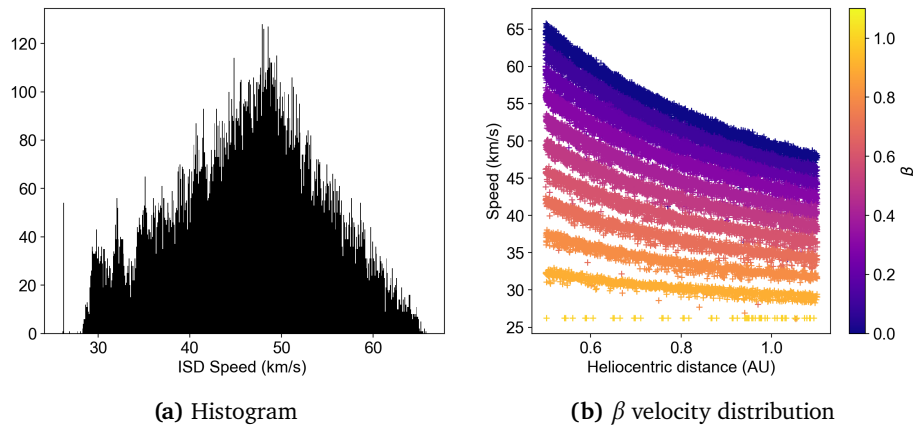


**Figure 5.5:** Gaussian kernel Density Estimation of ISD population with  $\beta$  values between 0.3 and 0.9 inside a region from 0.5 AU to 1.1 AU in the Ecliptic plane. Colors indicate hotspots where ISD trajectories overlap.

For instruments like the RPW to be able to detect dust impacts from ISD, the ISD needs sufficient collisional velocity to produce charge. The thesis has already shown some analytical calculated impact velocities for the orbit of the Solar Orbiter spacecraft, but in the later stages of the mission it will have a

different orbit and it could be very useful to know the general speed of ISD in the region, also for future spacecrafts. Through numerical calculations I made a histogram of all the velocities of the ISD that made up the Gaussian KDE image in 5.5, the histogram 5.6a show the speed distribution for all  $\beta$  values (0 to 1.1) combined.

To better see how the ISD speed changes with heliocentric distance, refer to the Figure 5.6b. All of the ISD with  $\beta < 1$  are accelerated above their initial entrance velocity, and a common speed is seen between 40-50 km/s for multiple  $\beta$  values. As the heliocentric distance decreases the difference in velocity for each  $\beta$  value increases. Vice Versa at far distances from the sun we can then expect a more similar velocity for ISD even with different  $\beta$  values and only small variations due to the solar pressure radiation force.



**Figure 5.6:** The histogram on the left has 500 bins of ISD velocities in the inner heliosphere covering 0.5 AU to 1.1 AU in the ecliptic plane, The right image shows the speed of ISD particles as a function of the heliocentric distance,  $\beta$  ratios are separated by color with the values shown in the colorbar on the right

### 5.3 Discussion

Our simulations show that potential dust impact velocities can reach up to 80 km/s for  $\beta = 0$  and in the 50 km/s range for  $\beta = 1$ , as shown in the figures 5.2. The regions we expect the highest flux and impact velocities are when the spacecraft is travelling in the upstream direction close to the perihelion, due to higher spacecraft velocities and impact velocities.

In the inner heliosphere we can expect much more impacts from  $\beta$ -meteroids

(Mann, 2010) as the production increases with the inverse square distance of the sun. Because of this, the spacecraft will be able to detect a higher flux of dust impacts as it gets increasingly closer to the sun. When the radial component is a much stronger variable in the dust-flux detection it is hard to distinguish any changes that happen longitudinally without a stable orbit.



# /6

## A Brief Look at the RPW Dust Impact Data

### 6.1 Information on the data

Data for detected dust impacts per day on board of Solar Orbiter can be downloaded from the TDS archive ([Link](#)). This thesis also considers daily dust counts from Kvammen et al. (2022) using a CNN on RPW data (presented in section 2.2.3). 497 days are recorded from June 15th 2020 to November 25th 2021 with each day providing two integers for number of dust impacts measured with each flagging algorithm. Gaps in the data exist when the RPW was configured in dipole mode, and the data used only contains measurements from when the RPW was in a monopole configuration (explained in sec 2.2), this can be seen in the data as multiple days where 0 dust impacts per day has been recorded. Full data is given in appendix.

From the previous models on impact velocity from ISD on Solar Orbiter (Figure 5.2) it was concluded that the region of the orbit with the highest impact velocity and likely highest flux for dust impacts by ISD were from  $90^\circ$  to  $200^\circ$  HAE longitude, therefore I want to focus our attention on this region when looking at dust impacts. IDPs were dominant from  $200^\circ$  to  $340^\circ$  HAE longitude and will likely crowd data with dust impacts between those longitudes.

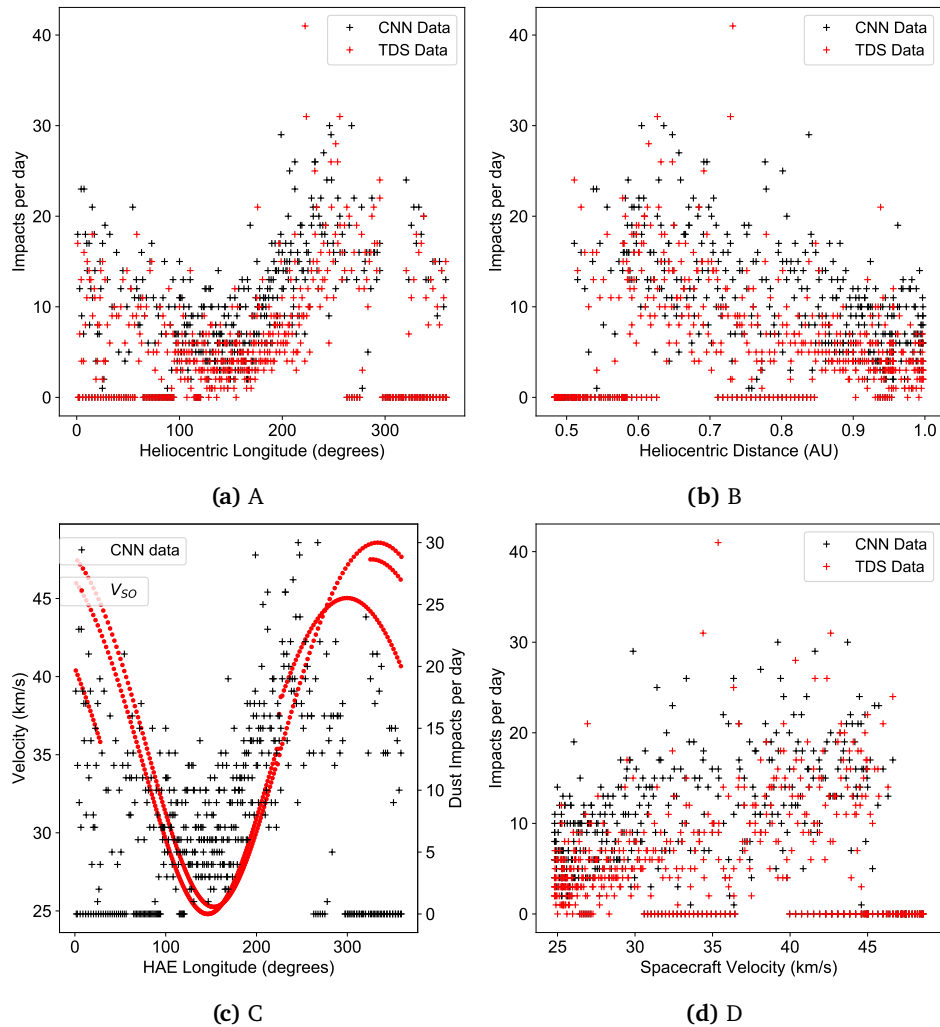
Positional data provided for the Solar Orbiter spacecraft from the ephemeris

needed to be downscaled to the same length as the data, after which the data could be displayed as a function of a variety of spacecraft parameters as they were each day.

## 6.2 Dust Impacts per day

If IDPs provide most of the dust impacts recorded, we should expect the highest flux closest to the sun where production of  $\beta$ -meteoroids is increased. Plotting the data as a function of heliocentric distance (Fig 6.1b) shows higher flux closer to the sun. If we look closer at the spacecraft velocity we see that the CNN flagging algorithm finds more dust impacts when Solar Orbiter has a higher velocity (Fig 6.1c). Assume that the number of dust impacts per day from IDPs is related to the orbital velocity of the spacecraft, dividing by the spacecraft velocity removes some of the variation contributed by the velocity.

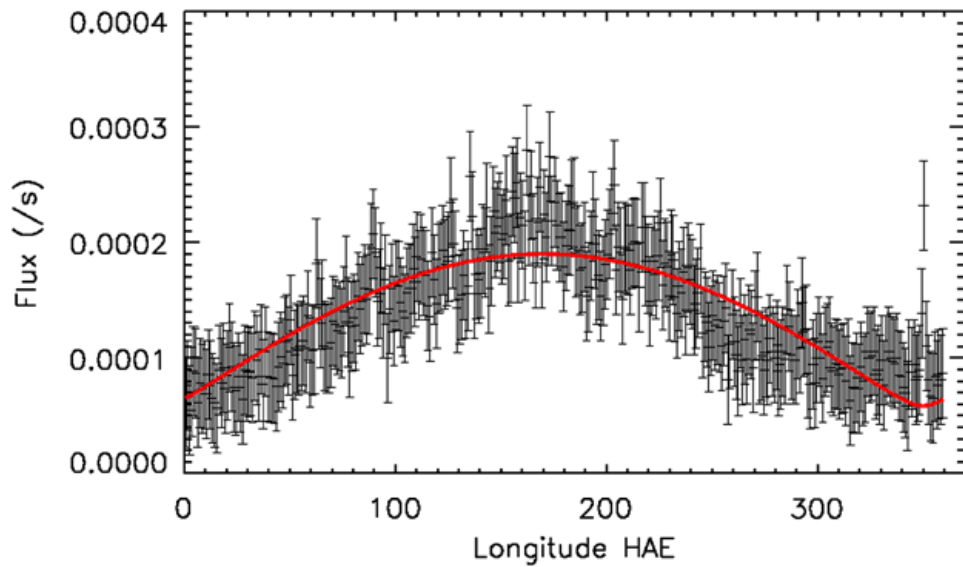




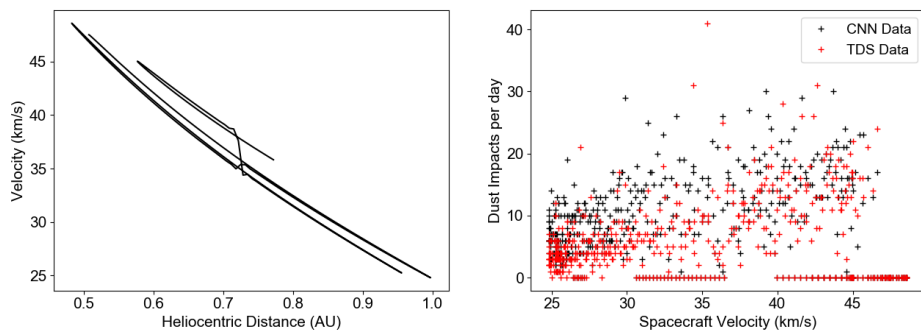
**Figure 6.1:** The data has measured dust impacts per day using two different flagging algorithms. The CNN classification detects about 26% more dust than TDS flagging (Kvammen et al., 2022). All figures show the dust flux as a function of a spacecraft parameter, Fig 6.1a is a function of the heliocentric longitude in HAE, Fig 6.1b as heliocentric distance, 6.1c heliocentric longitude with spacecraft velocity on a separate Y axis, and 6.1d as a function of the spacecraft velocity

Variable ecliptic orbits like the Solar Orbiter spacecraft has around the sun create challenges for detecting the upstream direction of ISD, since the spacecraft is travelling radially for a large part of the orbit the contribution of IDPs can not be assumed to be constant. STEREO was a spacecraft in a circular orbit around the sun at  $1 AU$  that used flux variation in dust impacts over heliocentric longitude to measure the upstream direction of ISD (Belheouane

et al., 2012). Because STEREO only travelled azimuthal to the sun it allowed for the assumption that  $\beta$ -meteoroids would impact radially from the sun and not vary in longitudes, data was gathered for over a 5 year period between 2007 and 2011 (Fig 6.2) to show a flux increase when the spacecraft was travelling in the upstream direction. During the STEREO mission the solar cycle was in a focusing phase, when ISD with a large enough charge to mass ratio could be driven into the ecliptic plane through Lorentz Forces, which might have contributed to the flux measurements.



**Figure 6.2:** Averaged dust flux over five years as function of solar ecliptic longitude. Red curve is a fitted model. From (Belheouane et al., 2012)



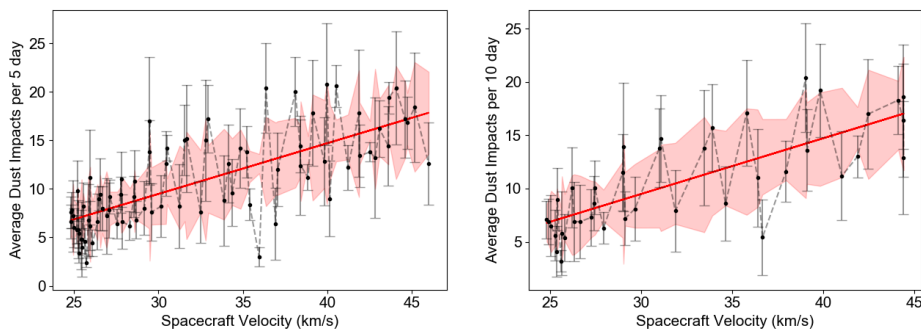
**(a)** Velocity of Solar Orbiter over the heliocentric distance **(b)** Measured Dust impacts per day as a function of spacecraft velocity

**Figure 6.3:** As the spacecraft got closer to the sun, its velocity is increased and so would the measured dust impacts.

I tried finding a direct relation between the number of dust impacts per day and a spacecraft parameter like orbital velocity or heliocentric distance, this could allow for some way to account for  $\beta$ -meteoroids in the data and if the contribution is proportional to a spacecraft parameter we can divide by the same parameter to see a reduced flux variation.

When a spacecraft has a higher velocity, the velocity needed from a dust particle (ISD or ISD) to reach sufficient impact velocity to be detected is lower, it can thus be deduced that a high spacecraft velocity can lead to increased dust flux measurements, compared to a constant orbital velocity (like in a circular orbit) where the velocity of the dust particle is a more dominant variable in equating a high enough impact velocity for dust impact detection.

Data shown in Fig 6.1c confirm that more dust impacts are recorded per day when Solar Orbiter has a higher orbital velocity, and the velocity of Solar Orbiter increases with decreasing heliocentric distance (Fig 6.3a). While it actually can be described by the vis-viva equation 4.8 and is inversely proportional to the square root of the distance to the sun.



(a) 5 day averaged numbered of dust impacts (b) 10 day averaged numbered of dust impacts

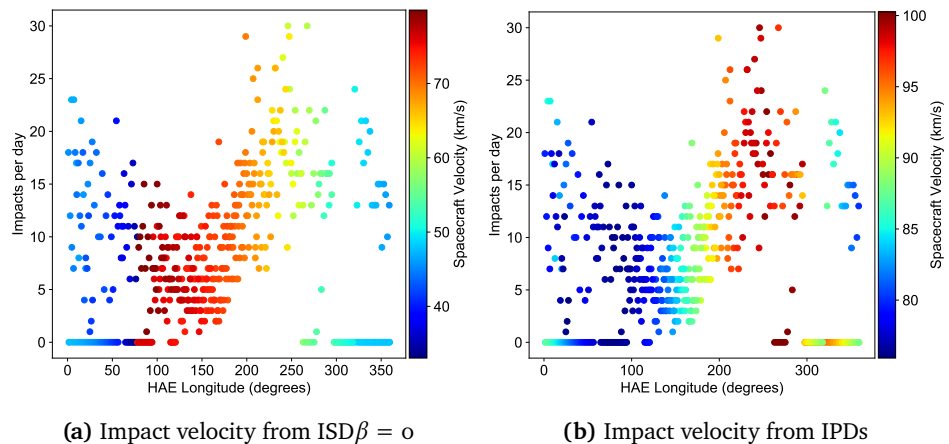
**Figure 6.4:** Dust impacts averaged over multiple days, as a function of spacecraft velocity

Simply plotting the dust impacts as a function of Spacecraft velocity (Fig 6.3b) makes it very difficult to extract any useful information and the data points are very cluttered and spread out. To help make the data more readable further work included removing days when no dust impacts were recorded what so ever, and then taking the number of dust impacts measured per day and averaging them over multiple days. This removes some of the clutter of almost 400 data points, and the figures maintain error bars to represent the standard deviation for each average. Two instances was created, one for a five day average and another for a 10 day average, both seen in the figures 6.4. A linear fitted line

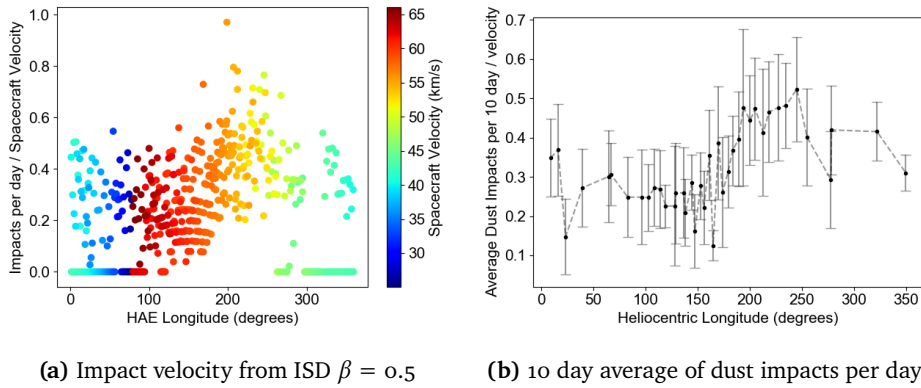
by least mean squares method is seen as the red line, and the filled area is marked as one standard deviation away from the linear fitted line, as we can see most data points fit inside the area. Already from the 5 day average it is clear that the impact rate increases with the spacecraft velocity, which should not come as a surprise, but the linearity in the Figure 6.4a must be taken simply at face value as the data points are gathered over multiple orbits and are in fact not linearly dependent on the velocity.

### 6.3 Relation to Impact Velocity

I divided the Dust impacts per day with the velocity of the spacecraft and plotted over the heliocentric longitude (Fig 6.1d). The idea was that the number of impacts per day would be greater when the velocity is higher, and lower with lower velocity, as inferred earlier. If we keep the assumption that all  $\beta$ -meteoroids are radially ejected from the sun, with a constant velocity then the spacecraft velocity is the determining factor in the impact velocity and dust impacts. As shown in the figures 6.3a and 6.4 there is a connection between the distance of sun/speed of Solar Orbiter and the number of recorded dust impacts per day, with some slight variation. The data has been aggregated together so data from multiple orbits overlap and can make it difficult to see the actual relation in one orbit.



**Figure 6.5:** Dust impacts per day as a function of heliocentric longitude. Color is represented as the velocity at each recorded day, on the left is the impact velocity from ISD with  $\beta = 0.5$  and on the right the color is the impact velocity produced from IPDs



**Figure 6.6:** The ratio of dust impacts per day over the spacecraft velocity, shown as a scatter plot with the impact velocity from ISD  $\beta = 0.5$  on the left and a 10 day averaged impacts on the right

Note that in the Figure 6.5a, the dark blue on the left side is misleading, as previously discussed and shown in Figure 5.1b, outgoing ISD will still impact Solar Orbiter in the opposite direction and produce much higher impact velocities. Seen in Fig 6.5b there is a clear indication that the measured dust flux is heavily contributed by  $\beta$ -meteoroids in the inner heliosphere between 200 and 300 degrees longitude.

Attempting again to compensate for the orbital velocity of the spacecraft, if we divide the number of dust impacts per day with the velocity of Solar orbiter and display the impact velocity from ISD in color we achieve the Figure 6.6a and with a 10 day average we see in Figure 6.6b an increased dust flux from  $150^\circ$  to  $300^\circ$  longitude.





## Conclusion

In this thesis I wanted to model impact velocities from ISD onto the Solar Orbiter spacecraft and see if it could be used to measure ISD fluxes. Using the equations from section 4.1 to calculate the impact velocity on Solar Orbiter from ISD with different  $\beta$ -ratios, it was found that for ISD with  $\beta = 0$  impact velocities range from 50 km/s upwards to 80 km/s between the HAE longitudes of  $80^\circ$  and  $270^\circ$  (Fig 5.2a). In the same longitudes ( $80^\circ$  to  $270^\circ$ ) the impact velocity is highest for other  $\beta$  values calculated as well, with  $\beta = 0.9$  having the lowest impact velocity of roughly 40-50 km/s (Fig 5.2d). Lorentz forces were not relevant for this Master Thesis, so all model calculations have been done without them.

For ISD with  $\beta = 1$  the impact velocity is above 40 km/s from  $90^\circ$  to  $280^\circ$  HAE longitude reaching a maximum of above 60 km/s (Fig 5.3a). The impact velocity for  $\beta = 1$  is calculated as higher than for  $\beta = 0.9$  as a result of more optimal impact trajectory, where  $\beta = 0.9$  has the vectors from Fig 5.1b but  $\beta = 1$  is calculated as a straight line trajectory.

Simulating ISD flux with the IVP solver as discussed in section 4.3 it was found that, for ISD with a uniform distribution of  $\beta$  ratios between 0 and 1.1, the highest population density is in the gravitational focusing region behind the sun (Fig 5.5) where ISD trajectories from all directions (From the left, right, above and below) gather towards the same line. The ISD velocity distribution (Fig 5.6a) across the entire region for all  $\beta$  ratios show a peak around 50 km/s.

This Master Thesis has not been able to prove that ISD was detected in the data provided on the number of dust impacts per day, but as discussed in section 6.3 the elliptical orbit of Solar Orbiter amplifies the contribution from  $\beta$ -meteoroids in the data, compared to a circular orbit like STEREO with no radial variation. With STEREO (Belheouane et al., 2012) it was much easier to see a flux increase in the ISD upstream direction (Fig 6.2) compared to Solar Orbiter (Fig 6.1a).

$\beta$ -meteoroid impact velocity is modeled in Figure 5.3b assuming only radial collisions. In Figure 6.5b the impact velocity is displayed over the dust flux, and the spike in dust fluxes seen between  $200^\circ$  and  $300^\circ$  coincide with an increase in impact velocity from  $\beta$ -meteoroids. This implies that the increase in measured dust impacts comes from a high production of  $\beta$ -meteoroids near the sun.

Solar Orbiter proves very useful for measuring the flux of  $\beta$ -meteoroids in the inner heliosphere, but has yet to prove itself capable of measuring dust impacts from ISD or measuring the ISD upstream direction. A region that might prove itself useful is the gravitational focusing region in the downstream direction, where my models show that the flux has an enhanced population of ISD particles (Fig 5.5). The direction of ISD will however be heavily scattered in this region, Fig 5.4 show that  $\beta$ -meteoroids have the lowest impact velocity where ISD has its highest.

Future work could include a more detailed analysis of the charges produced by dust impacts close to this region. In this way one could calculate if the charge production is higher than possible charges from IDP impacts. A similar approach was followed by others and Altobelli et al. (2006) was who studied the ISD flux at the Helios spacecraft.



# Bibliography

Rpw data archive. [https://rpw.lesia.obspm.fr/roc/data/pub/solo/rpw/data/L2/tds\\_wf\\_e/](https://rpw.lesia.obspm.fr/roc/data/pub/solo/rpw/data/L2/tds_wf_e/). Accessed: 2022-05-29.

N Altobelli, Eberhard Grün, and M Landgraf. A new look into the helios dust experiment data: presence of interstellar dust inside the earth's orbit. *Astronomy & Astrophysics*, 448(1):243–252, 2006.

Nicolas Altobelli. *Monitoring of the interstellar dust stream in the inner solar system using data of different spacecraft*. PhD thesis, Dissertation, Ruprecht-Karls Universität Heidelberg, 2004.

Roger R Bate, Donald D Mueller, Jerry E White, and William W Saylor. *Fundamentals of astrodynamics*. Courier Dover Publications, 2020.

S Belheouane, A Zaslavsky, N Meyer-Vernet, K Issautier, I Mann, and M Maksimovic. Detection of interstellar dust with stereo/waves at 1 au. *Solar Physics*, 281(1):501–506, 2012.

Priscilla C Frisch, Johann M Dorschner, Johannes Geiss, J Mayo Greenberg, Eberhard Grün, Markus Landgraf, Peter Hoppe, Anthony P Jones, Wolfgang Krätschmer, Timur J Linde, et al. Dust in the local interstellar wind. *The Astrophysical Journal*, 525(1):492, 1999.

Eberhard Grün, Markus Landgraf, Mihaly Horányi, Jochen Kissel, Harald Krüger, Ralf Srama, Håkan Svedhem, and Peter Withnell. Techniques for galactic dust measurements in the heliosphere. *Journal of Geophysical Research: Space Physics*, 105(A5):10403–10410, 2000.

Emil Gorseth Henriksen. Interplanetary dust fluxes observed with parker solar probe. Master's thesis, UiT Norges arktiske universitet, 2020.

Harald Krüger, Peter Strub, Nicolas Altobelli, Veerle J Sterken, Ralf Srama, and Eberhard Grün. Interstellar dust in the solar system: model versus in situ spacecraft data. *Astronomy & Astrophysics*, 626:A37, 2019.

- Andreas Kvammen, Kristoffer Wickstrøm, Samuel Kociscak, Jakub Vaverka, Libor Nouzak, Arnaud Zaslavsky, Kristina Rackovic, Audun Theodorsen, Amalie Gjelsvik, David Pisa, Jan Soucek, and Ingrid Mann. In *Machine Learning Classification of Dust Impact Signals Observed by The Solar Orbiter [Unpublished Manuscript]*, 2022.
- M Landgraf. Modeling the motion and distribution of interstellar dust inside the heliosphere. *Journal of Geophysical Research: Space Physics*, 105(A5): 10303–10316, 2000.
- Kenneth R Lang. *The Sun from space*, volume 1. Springer, 2000.
- M Maksimovic, SD Bale, Thomas Chust, Y Khotyaintsev, V Krasnoselskikh, M Kretzschmar, D Plettemeier, HO Rucker, J Souček, M Steller, et al. The solar orbiter radio and plasma waves (rpw) instrument. *Astronomy & Astrophysics*, 642:A12, 2020.
- Ingrid Mann. Interstellar dust in the solar system. *Annual Review of Astronomy and Astrophysics*, 48:173–203, 2010.
- Ingrid Mann, Libor Nouzák, Jakub Vaverka, Tarjei Antonsen, Åshild Fredriksen, Karine Issautier, David Malaspina, Nicole Meyer-Vernet, Jiří Pavl, Zoltan Sternovsky, et al. Dust observations with antenna measurements and its prospects for observations with parker solar probe and solar orbiter. In *Annales Geophysicae*, volume 37, pages 1121–1140. Copernicus GmbH, 2019.
- Daniel Mueller, Richard George Marsden, OC St Cyr, and Holly Robin Gilbert. Solar orbiter. *Solar Physics*, 285(1):25–70, 2013.
- J Soucek, D Píša, I Kolmasova, L Uhlir, R Lan, O Santolík, V Krupar, O Kruparova, J Baše, M Maksimovic, et al. Solar orbiter radio and plasma waves–time domain sampler: In-flight performance and first results. *Astronomy & Astrophysics*, 656:A26, 2021.
- Veerle Jasmin Sterken, N Altobelli, S Kempf, G Schwehm, R Srama, and E Grün. The flow of interstellar dust into the solar system. *Astronomy & Astrophysics*, 538:A102, 2012.
- BN Suresh and K Sivan. *Integrated design for space transportation system*. Springer, 2015.
- Zhiguang Wang, Weizhong Yan, and Tim Oates. Time series classification from scratch with deep neural networks: A strong baseline. In *2017 International joint conference on neural networks (IJCNN)*, pages 1578–1585. IEEE, 2017.

- A Zaslavsky. Floating potential perturbations due to micrometeoroid impacts: Theory and application to s/waves data. *Journal of Geophysical Research: Space Physics*, 120(2):855–867, 2015.





# appendix

## A.1 Special Curriculum

### A.1.1 Abstract from Project Paper A. Henriksen 2020

The spacecraft Solar Orbiter orbits the inner heliosphere between  $1 AU$  and  $0.3 AU$ , previously interstellar dust has only been detected as close as  $1 AU$  with the STEREO A/B spacecraft. Using assumptions gathered from previous data and calculated parameters, can the Solar Orbiter spacecraft detect interstellar dust? Short answer: Yes

## A.2 Extra equations

In Fig A.1 I have taken the equation 4.16 and inserted every variable until it is only a function of the spacecraft coordinates. The equation is only valid for an ISD-fixed coordinate system hence transformation matrices can also be added to the very big equation, and this has to be done for both velocity components.



## A.3 Code

Codes written by the author was created in Spyder using Python 3.7. The conversion scripts provided were also written in Spyder using Python but the version is not known, however it does run on Python 3.7. Codes are segmented to be run cell by cell, but to call functions from other files the entire full needs to be compiled.

### A.3.1 Solar Orbiter Ephemeris

The code for converting the ephemeris was given to me by Samuel Kociisák, and the credit goes to him for converting different coordinate systems into HAE

```

1 import numpy as np
2 import time as t
3 import datetime as dt
4 import pycdf as pycdf
5
6 def zrot(th): #in degrees
7     thrad = np.radians(th)
8     c, s = np.cos(thrad), np.sin(thrad)
9     return np.array(((c, -s, 0), (s, c, 0), (0,0,1)))
10
11 def gse2hae(gse,julian): #expecting 3-item vector or list in km
12     , julian date
13     gse=np.array(gse)
14     if len(gse)!=3:
15         raise ValueError("GSE is a 3D coordinate system, vector
16         has to be 3 items long!")
17     mjd = julian-2400000.5 #modified julian date
18     #print(round(mjd))
19     r0 = 1.495985e8 #in km
20     T0 = (round(mjd)-51544.5)/(36525.0)
21     UT = (julian%1)*24 #UTC in hours
22     M = 357.528 + 35999.05*T0 + 0.04107*UT
23     L = 280.46 + 36000.772*T0 + 0.04107*UT
24     l = L + (1.915 - 0.0048*T0)*np.sin(np.radians(M)) + 0.02*
25     np.sin(np.radians(2*M))
26     e = 0.016709 - 0.0000418*T0
27     w = 282.94 + 1.72*T0
28     nu = l - w
29     R = (r0*(1-e**2))/(1+e*np.cos(np.radians(nu)))
30     hee = np.array([R,0,0]) + np.inner(zrot(180),gse)
31     hae = np.inner(zrot(+1+180),hee)
32     return hae
33
34 def gse2gae(gse,julian): #expecting 3-item vector or list in km
35     , julian date
36     gse=np.array(gse)

```

```

33     if len(gse)!=3:
34         raise ValueError("GSE is a 3D coordinate system, vector
           has to be 3 items long!")
35     mjd = julian-2400000.5 #modified julian date
36     #print(round(mjd))
37     #r0 = 1.495985e8 #in km
38     TO = (round(mjd)-51544.5)/(36525.0)
39     UT = (julian%1)*24 #UTC in hours
40     M = 357.528 + 35999.05*TO + 0.04107*UT
41     L = 280.46 + 36000.772*TO + 0.04107*UT
42     l = L + (1.915 - 0.0048*TO)*np.sin(np.radians(M)) + 0.02*
           np.sin(np.radians(2*M))
43     #e = 0.016709 - 0.0000418*TO
44     #w = 282.94 + 1.72*TO
45     #nu = l - w
46     #R = (r0*(1-e**2))/(1+e*np.cos(np.radians(nu)))
47     gae = np.inner(zrot(180),gse)
48     gae = np.inner(zrot(+l+180),gae)
49     return [gae[0],gae[1],gae[2]]
50
51 def tt2000_to_date(epoch): #some adapted C routine
52     x=pycdf.Library()
53     epoch = round(epoch)
54     x=x.v_tt2000_to_datetime(epoch)
55     return x
56
57 def date_to_tt2000(date): #some adapted C routine
58     x=pycdf.Library()
59     x=x.v_datetime_to_tt2000(date)
60     return x
61
62 def date2unixtime(date):
63     return t.mktime(date.timetuple())
64
65 def unix2date(unixsec):
66     return dt.datetime.utcfromtimestamp(unixsec)
67
68 def unix2jd(unixsec):
69     return ( unixsec / 86400.0 ) + 2440587.5
70
71 def jd2unix(jd):
72     return (jd - 2440587.5) * 86400.0
73
74 def tt2000_to_unix(epoch):
75     return epoch/1e9 + 946724335
76
77 def unix_to_tt2000(unixsec):
78     return (unixsec - 946724335)*1e9
79
80 def jd2date(jd):
81     return dt.datetime.utcfromtimestamp((jd - 2440587.5) *
           86400.0)
82
83 def jd_to_tt2000(jd):

```



```

84     return unix_to_tt2000(jd2unix(jd))
85
86 def tt2000_to_jd(epoch):
87     return unix2jd(tt2000_to_unix(epoch))

```

Listing A.1: conversions.py

### Takes the ephemeris data for Solar Orbiter and produces HAE coordinates over time as python arrays

```

1  import csv
2  import numpy as np
3  import time as t
4  import datetime as dt
5  from conversions import tt2000_to_jd
6  from scipy import interpolate
7
8  #read ephemeris
9  time = np.zeros(0)
10 hae_r = np.zeros(0)
11 hae_v = np.zeros(0)
12 ephemeris = "ephemeris_noheader.txt"
13
14 with open(ephemeris) as file:
15     reader = csv.reader(file, delimiter=',')
16     for row in reader:
17         time = np.append(time, float(row[0]))
18         hae_r = np.append(hae_r, [float(row[2]), float(row[3]),
19 float(row[4])])
20         hae_v = np.append(hae_v, [float(row[5]), float(row[6]),
21 float(row[7])])
22     hae_r = np.reshape(hae_r, ((len(hae_r)//3,3)))
23     hae_v = np.reshape(hae_v, ((len(hae_v)//3,3)))
24
25 #construct functions for location and velocity
26 hae_r_x = interpolate.interp1d(time, hae_r[:,0], fill_value="
27 extrapolate", kind=3)
28 hae_r_y = interpolate.interp1d(time, hae_r[:,1], fill_value="
29 extrapolate", kind=3)
30 hae_r_z = interpolate.interp1d(time, hae_r[:,2], fill_value="
31 extrapolate", kind=3)
32 hae_v_x = interpolate.interp1d(time, hae_v[:,0], fill_value="
33 extrapolate", kind=3)
34 hae_v_y = interpolate.interp1d(time, hae_v[:,1], fill_value="
35 extrapolate", kind=3)
36 hae_v_z = interpolate.interp1d(time, hae_v[:,2], fill_value="
37 extrapolate", kind=3)
38
39 def f_hae_r(epoch):
40     jd = tt2000_to_jd(epoch)
41     return np.array([ hae_r_x(jd) , hae_r_y(jd) , hae_r_z(jd)
42 ])

```

```

34
35 def f_hae_v(epoch):
36     jd = tt2000_to_jd(epoch)
37     return np.array([ hae_v_x(jd) , hae_v_y(jd) , hae_v_z(jd)
38                     ])
39
40 def date2unixtime(date):
41     return t.mktime(date.timetuple())
42
43 def unix_to_tt2000(unixsec):
44     return (unixsec - 946724335)*1e9
45
46
47 epoch = lambda Y,M,D: unix_to_tt2000(date2unixtime(dt.date(Y,M,
48                                     D)))
49 epoch2 = lambda Y,M,D: unix_to_tt2000(date2unixtime(dt.date(Y,M,
50                                     D)))
51 #print(epoch)
52 AU = 149597871 # in km
53 vector = lambda Y1,M1,D1,Y2,M2,D2: f_hae_r(np.linspace(epoch(Y1,
54                                     M1,D1), epoch2(Y2,M2,D2), 365*24))/AU # AU in km
55
56
57 def Sol0_vectors3d(Y1,M1,D1,Y2,M2,D2):
58     vector = f_hae_r(np.linspace(epoch(Y1,M1,D1), epoch2(Y2,M2,
59                                     D2), 365*24))/AU # AU in km
60     return vector
61
62
63 def Sol0_vel(Y1,M1,D1,Y2,M2,D2):
64     return f_hae_v(np.linspace(epoch(Y1,M1,D1), epoch2(Y2,M2,D2),
65                                     365*24)) # AU in km
66
67
68 #import matplotlib.pyplot as plt
69 #from mpl_toolkits.mplot3d import Axes3D
70 #fig = plt.figure(figsize=(10,10))
71 #ax = fig.add_subplot(111, projection="3d")
72 #ax.plot(vector[0], vector[1], vector[2])
73 #ax.set_xlim(-1,1)
74 #ax.set_ylim(-1,1)
75 #ax.set_zlim(-1,1)
76 #plt.show()
77
78 #plt.plot(vector[0], vector[1], color="red")
79 #plt.scatter(vector[0,0], vector[1,0])
80 #plt.show()

```

Listing A.2: solo\_ephemeris.py

## A.3.2 Analytical Impact Velocity

### Calculate Impact Velocity on Solar Orbiter

Following code was created by the author Aleksander Henriksen. Running the script will produce the following plots:

1. Speed of Solar Orbiter around the orbit
2. Impact Velocity from  $\beta$ -meteoroids
3. Impact Velocity from ISD  $\beta = 1$
4. Coordinate system rotating to ISD fixed
5. Impact Velocity from ISD  $\beta = 0$  (default)
6. Graph comparing impact velocity between ISD and IDP
7. Dust Impacts measured colored with velocity from IDPs

```

1 import numpy as np
2 import matplotlib.pyplot as plt
3 import matplotlib
4 import sympy as sym
5 from solo_ephemeris import Sol0_vectors3d, Sol0_vel
6 from mpl_toolkits.axes_grid1 import make_axes_locatable
7 import pandas as pd
8 import scipy.interpolate as sci
9
10 v_inf = 26 # km/s ISD speed
11 G = 6.67408e-11 # Gravitational constant
12 M = 1.989e30 # Sun mass in kg
13 AU = 149597871000
14 Sol0 = Sol0_vectors3d(2020,6,15,2021,10,25)
15 Sol0_velocity = Sol0_vel(2020,6,15,2021,10,25)
16 theta = np.deg2rad(258)
17 kappa = 3*np.pi - theta # Angle to rotate ISD to X axis
18 bravo = theta - np.pi # Angle to rotate from X axis to upstream
    direction
19 mpl_config = {
20     # Use LaTeX to write all text
21     "text.usetex": False,
22     "font.family": "Arial",
23     # Use 10pt font in plots, to match 10pt font in document
24     "axes.labelsize": 15,
25     "font.size": 15,
26     # Make the legend/label fonts a little smaller
27     "legend.fontsize": 15,

```

```

28     "xtick.labelsize": 15,
29     "ytick.labelsize": 15
30 }
31 plt.rcParams.update(mpl_config)
32
33 """
34 def colorbar(mappable, label):
35     import matplotlib.pyplot as plt
36     last_axes = plt.gca()
37     ax = mappable.axes
38     fig = ax.figure
39     divider = make_axes_locatable(ax)
40     cax = divider.append_axes("right", size="5%", pad=0.05)
41     cbar = fig.colorbar(mappable, cax=cax, label=label)
42     plt.sca(last_axes)
43     return cbar
44
45 """
46 X = Sol0[0]
47 Y = Sol0[1]
48 VX = Sol0_velocity[0]
49 VY = Sol0_velocity[1]
50 SO_V = np.sqrt(VX**2 + VY**2)
51
52
53 """
54
55
56 def IDP(fs):
57     return 80*np.cos(fs), 80*np.sin(fs)
58 fs = np.arctan2(Y,X)
59 IDP_V = np.array([IDP(fs)[0], IDP(fs)[1]])
60 IDP_impact = np.array([VX, VY]) - IDP_V
61 IDP_impact_speed = np.sqrt(IDP_impact[0]**2 + IDP_impact[1]**2)
62
63 """
64 Solar Orbiter speed
65 """
66 fig, ax = plt.subplots(figsize=(6,6), dpi=200)
67 scatter = ax.scatter(X, Y, c=SO_V, cmap="jet", marker=".")
68 ax.scatter(0, 0, marker="*", color="black")
69 colorbar(scatter, "Speed (km/s)")
70 plt.xlabel("X (AU)")
71 plt.ylabel("Y (AU)")
72 ax.set_aspect(1)
73 plt.xlim(-1, 1)
74 plt.ylim(-1, 1)
75 plt.tight_layout()
76
77 fig, ax = plt.subplots(figsize=(6,6), dpi=200)
78 scatter = ax.scatter(X, Y, c=IDP_impact_speed, cmap="jet", marker="
79     .")
80 ax.scatter(0, 0, marker="*", color="black")
81 colorbar(scatter, "Speed (km/s)")

```

```

81 axis_rotation = np.linspace(0,2*np.pi,10)
82 for x in axis_rotation:
83     ax.quiver(0.2*np.cos(x),0.2*np.sin(x),np.cos(x),np.sin(x),
84             angles="xy",zorder=4,color="black",scale=10)
85 plt.xlabel("X(AU)")
86 plt.ylabel("Y(AU)")
87 ax.set_aspect(1)
88 plt.xlim(-1,1)
89 plt.ylim(-1,1)
90 plt.tight_layout()
91 #plt.savefig("S0_speed.pdf",format="pdf",bbox_inches="tight
92 #%%
93 """
94 Impact velocity for beta = 1
95 """
96
97 ISD_VX = [-v_inf*np.cos(theta)]*len(VX)
98 ISD_VY = [-v_inf*np.sin(theta)]*len(VX)
99 ISD_V = np.array([ISD_VX,ISD_VY])
100 impact_velocity_vector = np.array([VX,VY]) - ISD_V
101 Impact_velocity = np.sqrt(impact_velocity_vector[0]**2 +
102     impact_velocity_vector[1]**2)
103
104 long = np.arctan2(Y,X) # Longitudes in HAE-frame
105 long_degree = np.rad2deg(long)
106 long_360 = np.where(long_degree < 0, long_degree + 360,
107     long_degree)
108
109 fig, ax = plt.subplots(figsize=(6,6),dpi=200)
110 ax.plot(long_360,Impact_velocity,".",color="black",label=r"$\
111     \beta$ = 1")
112 plt.axhline(y=35,color="red",label="35 km/s")
113 plt.legend()
114 plt.xlabel("Heliocentric Longitude (degrees)")
115 plt.ylabel("Impact Velocity (km/s)")
116 plt.tight_layout()
117 #%%
118 fig, ax = plt.subplots(figsize=(6,6),dpi=200)
119 scatter = ax.scatter(X,Y,c=Impact_velocity,cmap="jet",marker="."
120     ")
121 colorbar(scatter,"Impact Speed (km/s)")
122 axis_x = np.linspace(-0.9,0.9,5)
123 for x in axis_x:
124     ax.quiver(x,-1,ISD_VX[0],ISD_VY[0],angles="xy",zorder=4,
125         color="black",units="xy",scale=100,edgecolor="black")
126 plt.xlabel("X(AU)")
127 plt.ylabel("Y(AU)")
128 ax.set_aspect(1)
129 plt.xlim(-1,1)
130 plt.ylim(-1,1)
131 ax.scatter(0,0,marker="*",color="black")
132 plt.tight_layout()

```

```

129 #plt.savefig("ISD_beta_1_vel.pdf",format="pdf",bbox_inches="
      tight")
130 #%%
131 """
132 Rotate the coordinate system
133 """
134 R = [-X*np.cos(kappa)+Y*np.sin(kappa), X*np.sin(kappa) + Y*np.
      cos(kappa)] # rotating to an ISD locked on X-axis frame and
      flipping around the Y-axis
135 fig, ax = plt.subplots(figsize=(5,5),dpi=200)
136
137 ax.plot(X,Y,color="black")
138 ax.plot(R[0],R[1],color="red")
139 plt.xlabel("X(AU)")
140 plt.ylabel("Y(AU)")
141 ax.set_aspect(1)
142 plt.xlim(-1,1)
143 plt.ylim(-1,1)
144 ax.scatter(0,0,marker="*",color="black")
145 for x in axis_x:
146     ax.quiver(x,-1,ISD_VX[0],ISD_VY[0],angles="xy",zorder=4,
147             color="black",units="xy",scale=100,edgecolor="black")
148 axis_Y = np.linspace(-0.9,0.9,5)
149 for y in axis_Y:
150     ax.quiver(1,y,-1,0,angles="xy",zorder=4,color="red",units="
151             xy",scale=5,edgecolor="red")
152 plt.tight_layout()
153 #plt.savefig("system_rotation",format="pdf",bbox_inches="tight
154             ")
155 #%%
156 """
157 Calculate impact velocity on Solar Orbiter from ISD
158 choose beta value to change beta ratio for ISD
159 """
160 beta = 0
161 mu = G*M*(1-beta)
162
163 f = np.arctan2(R[1],R[0]) # Longitudes in ISD-frame
164 f_degree = np.rad2deg(f)
165 f_new = np.deg2rad(np.where(f_degree < 0, f_degree + 360,
166                             f_degree))
167 Radial_R = np.sqrt(R[0]**2 + R[1]**2) # Heliocentric distance
168 in ISD-frame
169
170 def impact_parameter(r,f):
171     b1 = (r*np.sin(f))/2*(1+np.sqrt(1+(4*mu)/(v_inf**2*r*(1+np.
172     cos(f))))))
173     b2 = (r*np.sin(f))/2*(1-np.sqrt(1+(4*mu)/(v_inf**2*r*(1+np.
174     cos(f))))))
175     return b1,b2

```

```

173 index = np.linspace(0, len(f)-1, len(f), dtype=int)
174 #fig, ax = plt.subplots(figsize=(6,6), dpi=200)
175 Relative_vel = []
176 for i in index:
177     # fs = f[i]
178     fs = f_new[i]
179     rs = Radial_R[i]
180     b1, b2 = impact_parameter(rs*AU, fs) # Impact parameter
181     # needed for an ISD to reach polar coordinate (rs, fs)
182     if R[1][i] > 0:
183         b = b1
184     if R[1][i] < 0:
185         b = b2
186     p = b**2*v_inf**2/mu
187     e = np.sqrt(1+ b**2*v_inf**4/mu**2)
188     ISD_r = p/(1+e*np.cos(fs-np.arcsin(1/e) - np.pi/2))
189     ISD_X = ISD_r*np.cos(fs)/AU
190     ISD_Y = ISD_r*np.sin(fs)/AU
191
192     a = b/(np.sqrt(e**2-1))
193     v = np.sqrt(mu*(2/ISD_r - 1/a))
194     theta_max = np.arccos(1/(-e))
195     delta = 2*np.arcsin(1/e)
196     dY = e*np.sin(fs + theta_max - delta)
197     dX = 1 + e*np.cos(fs + theta_max - delta)
198     phi = np.arctan2(dY, dX)
199     phi = np.pi + phi
200     ISD_VX = v*np.cos(phi)
201     ISD_VY = v*np.sin(phi)
202
203     """
204     Flip to X-axis
205     """
206
207     ISD_X = -ISD_X
208     ISD_VX = -ISD_VX
209
210     """
211     Rotating back to HAE frame
212     """
213
214     X_ISD = [ISD_X*np.cos(bravo) - ISD_Y*np.sin(bravo)]
215     Y_ISD = [ISD_X*np.sin(bravo) + ISD_Y*np.cos(bravo)]
216     VX_ISD = ISD_VX*np.cos(bravo)-ISD_VY*np.sin(bravo)
217     VY_ISD = ISD_VX*np.sin(bravo)-ISD_VY*np.cos(bravo)
218
219     V_S0 = np.array([VX[i], VY[i]])
220     V_ISD = np.array([VX_ISD/1e3, VY_ISD/1e3])
221     Relative_velocity = V_ISD - V_S0
222     Impact_speed = np.sqrt(Relative_velocity[0]**2 +
223     Relative_velocity[1]**2)
224     Relative_vel.append(Impact_speed)
225 #

```

```

225     """
226     Use the plots inside the loop to plot vector arrows on the
227     orbit.
228     Recommended to reduce the number of loops in the index
229     array.
230     change index = np.linspace(0,len(f)-1,len(f),dtype=int)
231     to index = np.linspace(0,len(f)-1,N,dtype=int) for N loops
232     """
233 #     plt.plot(X,Y,color="black")
234 #     plt.quiver(X_ISD,Y_ISD,VX_ISD,VY_ISD,angles="xy",zorder=4,
235 #               color="red")
236 #     plt.quiver(X[i],Y[i],VX[i],VY[i],angles="xy",zorder=3,
237 #               color="white",linewidth=1,edgecolor="black")
238 #
239 V_impact = np.asarray(Relative_vel).flatten()
240
241 """
242 Display impact velocity as color on the orbit
243 c=V_impact decides the color function.
244 """
245
246 fig, ax = plt.subplots(figsize=(6,6),dpi=200)
247 scatter = ax.scatter(X,Y,c=V_impact,cmap="jet",marker=".",vmin
248                    =20,vmax=80)
249 colorbar(scatter,"Impact Speed (km/s)")
250 ax.scatter(0,0,marker="*",color="black")
251 ax.set_aspect(1)
252 plt.xlabel("HAE X (AU)")
253 plt.ylabel("HAE Y (AU)")
254 plt.xlim(-1,1)
255 plt.ylim(-1,1)
256
257 """
258 Create arrows pointing in the ISD downstream direction
259 """
260 axis_x = np.linspace(-0.9,0.9,5)
261 isd_vX = [-v_inf*np.cos(theta)]
262 isd_vY = [-v_inf*np.sin(theta)]
263 #for x in axis_x:
264 #     ax.quiver(x,-1,isd_vX,isd_vY,angles="xy",zorder=4,color="
265 #               black",units="xy",scale=100,edgecolor="black")
266 plt.tight_layout()
267 #plt.savefig("Vectors.pdf",format="pdf",bbox_inches="tight")
268 #%%
269 """
270 Velocity plots of ISD
271 """
272 fig, ax = plt.subplots(figsize=(6,6),dpi=200)
273 ax.plot(long_360,V_impact,".",color="black",label=r"$\beta$ = 1
274 ")
275 plt.axhline(y=40,color="red",label="40 km/s")
276 plt.legend()

```



```

272 plt.xlabel("Heliocentric Longitude (degrees)")
273 plt.ylabel("Impact Velocity (km/s)")
274
275 #%%
276 """
277 Compare impact velocity from IDP with ISD
278
279 The Y axis are separate to illustrate change in normalized
    magnitude,
280 since IDP has a much higher velocity than ISD.
281 """
282
283 fig, ax1 = plt.subplots(figsize=(6,6),dpi=200)
284 longitude = np.rad2deg(np.arctan2(Y,X))
285 longitude_360 = np.where(longitude < 0, longitude + 360,
    longitude)
286 ax1.plot(longitude_360, IDP_impact_speed, ".", color="black", label
    ="IDP")
287 plt.legend()
288 ax1.set_ylabel("IDP Velocity (km/s)")
289 ax1.set_xlabel("HAE Longitude (degrees)")
290 ax2 = ax1.twinx()
291 ax2.set_ylabel("ISD Velocity (km/s)")
292 ax2.plot(longitude_360, V_impact, ".", color="red", label=str(r"ISD
    $\beta$ = " + str(beta)))
293 ax.set_aspect(1)
294 plt.legend()
295 #plt.savefig("ISD_vs_IDP", format="pdf", bbox_inches="tight")
296 #plt.tight_layout()
297 #%%
298
299 """
300 Load dust impact data.
301
302 Interpolate the data we have previously calculated to fit the
    size of dust data,
303 then you are free to plot them against each other.
304 """
305
306 data = pd.read_csv("DustData.csv", names=["days", "CNN", "TDS"],
    delimiter=",", header=1)
307
308
309 CNN = data.CNN.values
310 TDS = data.TDS.values
311
312 Sp_speed = np.sqrt(Sol0_velocity[0]**2 + Sol0_velocity[1]**2)
313
314 Sol0_longitude = np.rad2deg(np.arctan2(Sol0[1], Sol0[0]))
315 Sol0_length = np.linspace(0,1, len(Sol0[0]))
316 data_length = np.linspace(0,1, len(CNN))
317 interpolate_data = sci.interp1d(data_length, CNN, kind="linear")
318 CNN_interpolated = interpolate_data(Sol0_length)
319 old_degrees = np.linspace(-180,180, len(Sol0[0]))

```

```

320 new_degrees = np.linspace(-180,180,len(CNN))
321 temp_deg = sci.interp1d(old_degrees,Sol0_longitude)
322 longitude = temp_deg(new_degrees)
323 old_speed_range = np.linspace(0,1,len(Sol0[0]))
324 new_speed_range = np.linspace(0,1,len(CNN))
325 temp_speed = sci.interp1d(old_speed_range, Sp_speed)
326 temp_V_speed = sci.interp1d(old_speed_range, V_impact)
327 V_speed = temp_V_speed(new_speed_range)
328 temp_IDP_speed = sci.interp1d(old_speed_range, IDP_impact_speed)
329 IDP_speed = temp_IDP_speed(new_speed_range)
330
331 Speed = temp_speed(new_speed_range)
332 longitude_360 = np.where(longitude < 0, longitude + 360,
333                          longitude)
334
335 Rad = np.sqrt(Sol0[0]**2 + Sol0[1]**2)
336 old_radial = np.linspace(np.min(Rad),np.max(Rad),len(Rad))
337 new_radial = np.linspace(np.min(Rad),np.max(Rad),len(CNN))
338 temp_R = sci.interp1d(old_radial,Rad)
339 R_interp = temp_R(new_radial)
340 ###
341 fig, ax = plt.subplots(figsize=(7,6),dpi=200)
342
343 scatter = ax.scatter(longitude_360,CNN,c=IDP_speed,cmap="jet")
344 colorbar(scatter,"Spacecraft Velocity (km/s)")
345 plt.xlabel("HAE Longitude (degrees)")
346 plt.ylabel("Impacts per day")
347 plt.tight_layout()
348 #plt.savefig("dust_IDP.pdf",format="pdf",bbox_inches="tight")

```

Listing A.3: Script for calculating impact velocity

### A.3.3 Numerical calculations

Big script that numerically calculates ISD in the environment around solar orbiter, from 0.5 AU to 1.1 AU in radius around the sun. The Kernel Density Estimations are made from the numerical calculations

```

1 import numpy as np
2 import scipy.integrate as sc
3 import matplotlib.pyplot as plt
4 import matplotlib
5 from scipy import interpolate
6 from mpl_toolkits.mplot3d import Axes3D
7 from solo_ephemeris import Sol0_vectors3d, Sol0_vel
8 import random
9
10 AU = 149597871000
11 Sol0 = Sol0_vectors3d(2020,6,15,2021,10,25)
12 Sol0_velocity = Sol0_vel(2020,6,15,2021,10,25)
13

```

```

14
15 v0 = 26e3
16 ###
17 mpl_config = {
18     # Use LaTeX to write all text
19     "text.usetex": False,
20     "font.family": "Arial",
21     # Use 10pt font in plots, to match 10pt font in document
22     "axes.labelsize": 15,
23     "font.size": 15,
24     # Make the legend/label fonts a little smaller
25     "legend.fontsize": 15,
26     "xtick.labelsize": 15,
27     "ytick.labelsize": 15
28 }
29 plt.rcParams.update(mpl_config)
30 ###
31
32
33 beta = [0.5]
34 ISD_u_d = np.deg2rad(260) # Interstellar Dust Upstream
    Direction
35 ISD_latitude = np.deg2rad(8) # ISD upstream inclination
36 t0 = 0
37 t_bound = 10*365*24*60*60 # 9 years in seconds
38 t_span = (t0,t_bound)
39 time_steps = t_bound/2
40
41 ###
42 """
43 the solver(ip,beta) function is the main function for the
    numerical calculation
44 it takes in initial values in position and velocity as 'ip' and
    the beta ratio.
45
46 position and velocity has 3 dimensions so ip has 6 elements.
47 it uses a forward eulers method, it calculates the change in
    position based on
48 velocity, and a change in velocity from the acceleration.
49
50 the function then gives solution for the tuple t_span, for
    time_steps amount of steps.
51
52 Returned are positional and velocity values.
53 """
54 def solver(ip,beta):
55     G = 6.67408e-11 # Gravitational constant
56     M = 1.989e30
57     def func(t,p):
58         r = p[:3] #X,Y,Z
59         v = p[3:] #Vx,Vy,Vz
60         norm_r = np.linalg.norm(r)
61         a = -G*M*(1-beta)*r/norm_r**3
62         drdt = v

```

```

63     dydt = np.concatenate((drdt, a))
64     return dydt
65
66     sol = sc.solve_ivp(func, t_span, ip, method="RK45", max_step=
time_steps)
67     R = sol.y[:3,:]
68     V = sol.y[3:,:]
69
70     return R, V
71
72 %%
73
74 """
75
76
77 """
78
79
80 #fig = plt.figure(figsize=(10,10))
81 #ax = fig.add_subplot(111, projection="3d")
82
83 grid = np.arange(-2,2,0.11)
84 beta = np.arange(0,1.1,0.1)
85 #beta = [0.5]
86 empty_list = []
87 cmap = matplotlib.cm.get_cmap('plasma')
88 Speeds = []
89 V_dict = {}
90 P_dict = {}
91 for k in range(len(beta)):
92     b = beta[k]
93     colors = cmap(k/len(beta)) # Gives an individual color for
each beta value
94     grid_list_V = []
95     grid_list_P = []
96 #     label = "beta: " + str("%.2f" % b)
97     for i in grid:
98         for j in grid:
99             # init_pos is rotated to start 50 AU in the
upstream direction
100             # init_v is also rotated to stream towards the sun
from the upstream direction
101             # ip gathers both the arrays into a single array
injected into the solver function
102
103             init_pos = np.array([50*AU*np.cos(ISD_u_d)+i*AU*np.
sin(ISD_u_d), 50*AU*np.sin(ISD_u_d)+i*AU*np.cos(ISD_u_d), j
*AU+7*AU])
104             init_v = np.array([v0*np.cos(ISD_u_d-np.pi), v0*np.
sin(ISD_u_d-np.pi), -v0*np.sin(ISD_latitude)])
105             ip = np.concatenate((init_pos, init_v))
106             R, V = solver(ip, b) # The numerically calculated
positional and velocity values
107

```

```

108     t = [np.linspace(0,t_bound,len(R[i])) for i in
109     range(3)] # time array for each positional axis
110
111     """
112     The first restriction is the time variable, that
113     reduces from 13/20ths to 20/20ths of t_bound.
114     Then an interpolation increases the steps valid
115     inside the time variable range.
116
117     New positional and velocity values are interpolated
118     .
119     """
120     time = np.arange(13*t_bound/20,t_bound,int(t_bound
121     /200))
122
123     f = [interpolate.interp1d(t[i],R[i],kind="quadratic
124     ") for i in range(3)]
125     f_v = [interpolate.interp1d(t[i],V[i],kind="
126     quadratic") for i in range(3)]
127
128     R_new = [i(time) for i in f]
129     V_new = [i(time) for i in f_v]
130     x,y,z = R_new
131     vx,vy,vz = V_new
132
133     #
134     speed = np.sqrt(vx**2 + vy**2 + vz**2)
135     #
136     ax.scatter(x,y,z,c=speed) # Scatter plot the
137     position of ISD with a color based on the velocity value
138
139     radial = np.asarray(np.sqrt(x**2+y**2+z**2))
140     index = np.argwhere((0.5*AU < radial) & (radial <
141     1.1*AU)) # only give data between 0.5 and 1.1 AU
142
143     X = [float(i/AU) for i in x[index]]
144     Y = [float(i/AU) for i in y[index]]
145     Z = [float(i/AU) for i in z[index]]
146     V_X = [float(i) for i in vx[index]]
147     V_Y = [float(i) for i in vy[index]]
148     V_Z = [float(i) for i in vz[index]]
149
150     index_1 = np.argwhere((-0.1 < np.asarray(Z)) & (np.
151     asarray(Z) < 0.2)) # Only give data close to ecliptic plane
152
153     X = np.asarray(X)[index_1].ravel()
154     Y = np.asarray(Y)[index_1].ravel()
155     Z = np.asarray(Z)[index_1].ravel()
156     VX = np.asarray(V_X)[index_1].ravel()
157     VY = np.asarray(V_Y)[index_1].ravel()
158     VZ = np.asarray(V_Z)[index_1].ravel()
159
160     R1 = np.array([X,Y,Z])
161     R2 = [So10[0],So10[1],So10[2]]
162     if R1.size == 0:

```

```

152         None
153     else:
154         Velocity_vector = [VX,VY,VZ]
155         Position_vector = [X,Y,Z]
156         grid_list_V.append(Velocity_vector)
157         grid_list_P.append(Position_vector)
158         empty_list.append(b)
159
160         speed = np.sqrt(VX**2 + VY**2 + VZ**2)
161         longitude = np.rad2deg(np.arctan2(Y,X)+np.pi)
162         radius = np.sqrt(X**2 + Y**2 + Z**2)
163
164         """
165         New arrays containing interpolated and restricted ISD
166         trajectories close to Solar Orbiter
167         """
168         array_like_V = np.asarray(grid_list_V)
169         array_like_P = np.asarray(grid_list_P)
170         V_dict[k] = array_like_V
171         P_dict[k] = array_like_P
172
173     """
174     """
175     """
176     """
177     """
178     """
179     """
180     """
181     """
182     """
183     """
184     """
185     """
186     """
187     """
188     """
189     """
190     """
191     """
192     """
193     """
194     """
195     """
196     """
197     """
198     """
199     """
200     """
201     """
202     """
203     """

```

```

204 ###
205 def colorbar(mappable, label):
206     from mpl_toolkits.axes_grid1 import make_axes_locatable
207     import matplotlib.pyplot as plt
208     last_axes = plt.gca()
209     ax = mappable.axes
210     fig = ax.figure
211     divider = make_axes_locatable(ax)
212     cax = divider.append_axes("right", size="5%", pad=0.05)
213     cbar = fig.colorbar(mappable, cax=cax, label=label)
214     plt.sca(last_axes)
215     return cbar
216
217     ###
218
219 x = np.asarray(x).ravel()
220 y = np.asarray(y).ravel()
221 Vx = np.asarray(Vx).ravel()
222 Vy = np.asarray(Vy).ravel()
223 Vz = np.asarray(Vz).ravel()
224
225 xy = np.vstack([x,y])
226 z = gaussian_kde(xy)(xy)
227 idx = z.argsort()
228 x, y, z = x[idx], y[idx], z[idx]
229
230 fig = plt.figure(figsize=(7,7))
231 scatter = plt.scatter(x,y,c=z,cmap="plasma")
232 colorbar(scatter, "Density Estimation")
233 plt.plot(Sol0[0], Sol0[1], color="black", linewidth=2)
234 plt.scatter(0,0,marker="*", color="black")
235 plt.xlabel("HAE X-axis (AU)")
236 plt.ylabel("HAE Y-axis (AU)")
237 #plt.tight_layout()
238 plt.show()
239 ###
240 Speed = np.sqrt(Vx**2 + Vy**2 + Vz**2)
241 fig, ax = plt.subplots(figsize=(6,6), dpi=200)
242 ax.hist(Speed/1e3, bins=500, histtype="barstacked", color="black")
243 plt.xlabel("ISD Speed (km/s) ")
244 plt.tight_layout()
245 plt.show()
246
247 #plt.plot(np.sqrt(x**2+y**2+z**2), Speed, "+")
248
249 ###
250 fig, ax = plt.subplots(figsize=(6,6), dpi=200)
251
252 VY_170 = []
253 for v,p in zip(V_dict, P_dict):
254     colors = cmap(v/len(V_dict))
255     for i,j in zip(V_dict[v], P_dict[p]):
256         VX,VY,VZ = i
257         X,Y,Z = j

```

```

258     vect2 = [VX,VY,VZ]
259     longitude = np.rad2deg(np.arctan2(Y,X))
260     longitude = np.where(longitude < 0, longitude + 360,
        longitude)
261
262     radius = np.sqrt(X**2 + Y**2 + Z**2)
263     speed = np.sqrt(VX**2 + VY**2 + VZ**2)
264     ax.plot(radius ,speed/1e3,"+",color=colors)
265 #ax.set_aspect(1)
266 norm = matplotlib.colors.Normalize(vmin=0, vmax=1.1)
267 #colorbar(norm,r'$ \beta $ ')
268 fig.colorbar(matplotlib.cm.ScalarMappable(norm=norm, cmap=cmap)
        , label=r'$ \beta $ ')
269 plt.xlabel("Heliocentric distance (AU)")
270 plt.ylabel("Speed (km/s)")
271 plt.tight_layout()
272 plt.show()
273
274 #%%
275 beta_distribution = np.asarray(empty_list).ravel()
276 beta_speeds = np.asarray(Beta_speed).ravel()
277
278 plt.plot(beta_distribution ,Speed)
279 #%%
280
281 plt.figure(1)
282 plt.hist(beta_distribution ,bins=len(beta),density=True,histtype
        ="barstacked",color="green")
283 plt.xlabel(r"ISD $ \beta $ ratio")
284 plt.ylabel("Commonality")
285 plt.title(r"Random $ \beta $ distribution along Solar Orbiter
        trajectory ")
286 plt.show()

```

Listing A.4: ISD environment

Uses Runge Kutta 45 to plot ISD trajectories in 3D.

```

1 import numpy as np
2 import scipy.integrate as sc
3 import matplotlib.pyplot as plt
4 import matplotlib
5 from scipy import interpolate
6 from mpl_toolkits.mplot3d import Axes3D
7
8
9 AU = 149597871000 # 1 Astronomical Unit in meters
10 v0 = 26e3 # Initial velocity at infinity
11 G = 6.67408e-11 # Gravitational constant
12 M = 1.989e30 # Mass of sun
13
14 beta = 0
15 ISD_u_d = np.deg2rad(258)
16 ISD_latitude = np.deg2rad(8)

```



```

17 t0 = 0
18 t_bound = 12*365*24*60*60 # 9 years in seconds
19 t_span = (t0,t_bound)
20 time_steps = t_bound/200
21
22
23 def solver(ip):
24     def func(t,p):
25         r = p[:3] #X,Y,Z
26         v = p[3:] #Vx,Vy,Vz
27         norm_r = np.linalg.norm(r)
28         a = -G*M*(1-beta)*r/norm_r**3
29         drdt = v
30         dydt = np.concatenate((drdt, a))
31         return dydt
32     sol = sc.solve_ivp(func,t_span,ip,method="RK45",max_step=
time_steps)
33     R = sol.y[:3,:]
34     V = sol.y[3:,:]
35     return R,V
36
37 %%
38 fig = plt.figure(figsize=(6,6))
39 ax = fig.add_subplot(111, projection="3d")
40 mpl_config = {
41     # Use LaTeX to write all text
42     "text.usetex": False,
43     "font.family": "Arial",
44     # Use 10pt font in plots, to match 10pt font in document
45     "axes.labelsize": 20,
46     "font.size": 20,
47     # Make the legend/label fonts a little smaller
48     "legend.fontsize": 20,
49     "xtick.labelsize": 20,
50     "ytick.labelsize": 20
51 }
52 plt.rcParams.update(mpl_config)
53
54 grid = np.arange(-2,2,0.31)
55 ax.set_xlim(-2,2)
56 ax.set_ylim(-2,2)
57 ax.set_zlim(-2,2)
58 ax.set_xlabel("X(AU)")
59 ax.set_ylabel("Y(AU)")
60 ax.set_zlabel("Z(AU)")
61 import matplotlib.cm as cm
62 av_max_V = []
63 for i in grid:
64     for j in grid:
65         init_pos = np.array([-50*AU,i*AU, j*AU])
66         init_v = np.array([v0, 0, 0])
67         ip = np.concatenate((init_pos, init_v))
68         R,V = solver(ip)
69         t = [np.linspace(0,t_bound,len(R[i])) for i in range(3)]

```

```

]
70
71     time = np.arange(13*t_bound/20,t_bound,int(t_bound
72     /1000))
73     f = [interpolate.interp1d(t[i],R[i],kind="quadratic")
74     for i in range(3)]
75     f_v = [interpolate.interp1d(t[i],V[i],kind="quadratic")
76     for i in range(3)]
77     R_new = [i(time)/AU for i in f]
78     V_new = [i(time) for i in f_v]
79     x,y,z = R_new
80     vx,vy,vz = V_new
81     # x,y,z = R/AU
82     # vx,vy,vz = V
83     speed = np.sqrt(vx**2 + vy**2 + vz**2)
84     av_max_V.append(np.max(speed))
85     # ax.plot(x,y,z,"--",color="black",alpha=0.5)
86     ax.scatter(x,y,z,c=speed/1e3,cmap="jet",alpha=0.4,vmin
87     =10,vmax=60)
88 av_V_max = np.mean(av_max_V)
89 norm = matplotlib.colors.Normalize(vmin=10, vmax=60)
90 fig.colorbar(cm.ScalarMappable(norm=norm,cmap="jet"), ax=ax,
91 label="ISD Velocity (km/s)")

```

**Listing A.5:** Simple script for 3D flow illustration

The same numerical methods as the others, but only in the XY plane. Produces plots for ISD focusing around Solar Orbiter.

```

1 import numpy as np
2 import scipy.integrate as sc
3 import matplotlib.pyplot as plt
4 import matplotlib
5 import matplotlib.cm as cm
6 from scipy import interpolate
7 from mpl_toolkits.mplot3d import Axes3D
8 from solo_ephemeris import Sol0_vectors3d,Sol0_vel
9 from mpl_toolkits.axes_grid1 import make_axes_locatable
10 import pandas as pd
11 import scipy.interpolate as sci
12
13 AU = 149597871000 # 1 Astronomical Unit in meters
14 v0 = 26e3 # Initial velocity at infinity
15 G = 6.67408e-11 # Gravitational constant
16 M = 1.989e30 # Mass of sun
17
18 Sol0 = Sol0_vectors3d(2020,6,15,2021,10,25)
19 Sol0_velocity = Sol0_vel(2020,6,15,2021,10,25)
20
21 beta = 0
22 ISD_u_d = np.deg2rad(258)
23 ISD_latitude = np.deg2rad(8)

```

```

24 bravo = ISD_u_d - np.pi
25 t0 = 0
26 t_bound = 9*365*24*60*60 # 9 years in seconds
27 t_span = (t0,t_bound)
28 time_steps = t_bound/5000
29
30
31 def solver(ip):
32     def func(t,p):
33         r = p[:2] #X,Y
34         v = p[2:] #Vx,Vy
35         norm_r = np.linalg.norm(r)
36         a = -G*M*(1-beta)*r/norm_r**3
37         drdt = v
38         dydt = np.concatenate((drdt, a))
39         return dydt
40     sol = sc.solve_ivp(func,t_span,ip,method="RK45",max_step=
time_steps)
41     R = sol.y[:2,:]
42     V = sol.y[2:,:]
43     return R,V
44
45 %%
46 fig, ax = plt.subplots(figsize=(6,6),dpi=200)
47
48 mpl_config = {
49     # Use LaTeX to write all text
50     "text.usetex": False,
51     "font.family": "Arial",
52     # Use 10pt font in plots, to match 10pt font in document
53     "axes.labelsize": 15,
54     "font.size": 15,
55     # Make the legend/label fonts a little smaller
56     "legend.fontsize": 15,
57     "xtick.labelsize": 15,
58     "ytick.labelsize": 15
59 }
60 plt.rcParams.update(mpl_config)
61
62 impact_parameters = np.linspace(-2,2,10)
63
64
65 av_max_V = []
66 for b in impact_parameters:
67     init_pos = np.array([-30*AU,b*AU])
68     init_v = np.array([v0, 0])
69     ip = np.concatenate((init_pos, init_v))
70     R,V = solver(ip)
71     t = [np.linspace(0,t_bound,len(R[i])) for i in range(2)]
72     X = R[0]*np.cos(bravo) - R[1]*np.sin(bravo)
73     Y = R[0]*np.sin(bravo) + R[1]*np.cos(bravo)
74     VX = V[0]*np.cos(bravo) - V[1]*np.sin(bravo)
75     VY = V[0]*np.sin(bravo) + V[1]*np.cos(bravo)
76     speed = np.sqrt(VX**2 + VY**2)

```

```
77     ax.plot(X/AU, Y/AU)
78 #     ax.scatter(X/AU, Y/AU, c=speed/1e3, marker=".", cmap="jet",
79               vmin=26, vmax=100)
80 ax.plot(Sol0[0], Sol0[1], color="black")
81 ax.scatter(0, 0, color="black", marker="*")
82 plt.xlim(-2, 2)
83 plt.ylim(-2, 2)
84 plt.xlabel("X (AU)")
85 plt.ylabel("Y (AU)")
86 ax.set_aspect(1)
87 plt.tight_layout()
88 plt.savefig("ISD_focusing.pdf", format="pdf", bbox_inches="tight"
89           )
89 #av_V_max = np.mean(av_max_V)
90 #norm = matplotlib.colors.Normalize(vmin=10, vmax=60)
91 #fig.colorbar(cm.ScalarMappable(norm=norm, cmap="jet"), ax=ax,
92             label="ISD Velocity (km/s)")
```

Listing A.6: 2D illustrations

## A.4 Dust Impact Data

Below is the data for dust impacts per day flagged with the CNN and TDS respectively.

Days	CNN	TDS	Days	CNN	TDS
2020/06/15	0	0	2020/07/24	8	10
2020/06/16	0	0	2020/07/25	8	7
2020/06/17	0	0	2020/07/26	16	6
2020/06/18	0	0	2020/07/27	0	0
2020/06/19	0	0	2020/07/28	0	0
2020/06/20	0	0	2020/07/29	0	0
2020/06/21	0	0	2020/07/30	0	0
2020/06/22	0	0	2020/07/31	0	0
2020/06/23	0	0	2020/08/01	0	0
2020/06/24	0	0	2020/08/02	0	0
2020/06/25	0	0	2020/08/03	0	0
2020/06/26	0	0	2020/08/04	0	0
2020/06/27	0	0	2020/08/05	0	0
2020/06/28	0	0	2020/08/06	0	0
2020/06/29	23	13	2020/08/07	0	0
2020/06/30	23	16	2020/08/08	0	0
2020/07/01	17	11	2020/08/09	0	0
2020/07/02	8	13	2020/08/10	0	0
2020/07/03	21	18	2020/08/11	0	0
2020/07/04	14	11	2020/08/12	0	0
2020/07/05	7	4	2020/08/13	0	0
2020/07/06	15	11	2020/08/14	0	0
2020/07/07	17	15	2020/08/15	0	0
2020/07/08	19	13	2020/08/16	0	0
2020/07/09	13	8	2020/08/17	0	0
2020/07/10	18	6	2020/08/18	0	0
2020/07/11	10	9	2020/08/19	10	5
2020/07/12	10	10	2020/08/20	6	3
2020/07/13	16	9	2020/08/21	4	3
2020/07/14	5	9	2020/08/22	7	5
2020/07/15	12	10	2020/08/23	11	4
2020/07/16	14	13	2020/08/24	11	5
2020/07/17	4	6	2020/08/25	5	4
2020/07/18	10	14	2020/08/26	6	3
2020/07/19	5	8	2020/08/27	10	5
2020/07/20	12	12	2020/08/28	7	2
2020/07/21	21	9	2020/08/29	7	4
2020/07/22	11	8	2020/08/30	4	0
2020/07/23	13	9	2020/08/31	8	2

Days	CNN	TDS
2020/09/01	10	5
2020/09/02	8	4
2020/09/03	2	1
2020/09/04	5	6
2020/09/05	9	6
2020/09/06	4	0
2020/09/07	0	0
2020/09/08	0	0
2020/09/09	0	0
2020/09/10	0	0
2020/09/11	0	0
2020/09/12	0	0
2020/09/13	0	0
2020/09/14	10	3
2020/09/15	7	3
2020/09/16	10	7
2020/09/17	10	9
2020/09/18	4	2
2020/09/19	12	5
2020/09/20	2	1
2020/09/21	6	3
2020/09/22	2	0
2020/09/23	4	1
2020/09/24	12	8
2020/09/25	1	1
2020/09/26	5	1
2020/09/27	7	5
2020/09/28	7	3
2020/09/29	4	1
2020/09/30	4	2
2020/10/01	7	3
2020/10/02	6	3
2020/10/03	14	6
2020/10/04	7	6
2020/10/05	6	2
2020/10/06	3	1
2020/10/07	7	2
2020/10/08	6	4
2020/10/09	6	3

Days	CNN	TDS
2020/10/10	9	4
2020/10/11	10	6
2020/10/12	8	3
2020/10/13	6	2
2020/10/14	4	3
2020/10/15	9	2
2020/10/16	6	2
2020/10/17	11	7
2020/10/18	6	5
2020/10/19	8	3
2020/10/20	6	4
2020/10/21	8	4
2020/10/22	5	6
2020/10/23	3	2
2020/10/24	6	0
2020/10/25	9	4
2020/10/26	7	3
2020/10/27	12	7
2020/10/28	10	6
2020/10/29	10	3
2020/10/30	4	4
2020/10/31	13	4
2020/11/01	6	4
2020/11/02	12	4
2020/11/03	6	2
2020/11/04	10	4
2020/11/05	7	1
2020/11/06	13	5
2020/11/07	12	5
2020/11/08	6	3
2020/11/09	19	7
2020/11/10	6	1
2020/11/11	9	3
2020/11/12	9	5
2020/11/13	7	6
2020/11/14	9	4
2020/11/15	11	3
2020/11/16	7	11
2020/11/17	10	21

Days	CNN	TDS	Days	CNN	TDS
2020/11/18	6	10	2020/12/27	13	41
2020/11/19	12	10	2020/12/28	16	9
2020/11/20	4	4	2020/12/29	16	10
2020/11/21	6	6	2020/12/30	15	10
2020/11/22	10	13	2020/12/31	22	14
2020/11/23	11	7	2021/01/01	21	13
2020/11/24	9	5	2021/01/02	26	25
2020/11/25	11	4	2021/01/03	21	21
2020/11/26	9	6	2021/01/04	12	12
2020/11/27	12	5	2021/01/05	19	10
2020/11/28	11	7	2021/01/06	18	12
2020/11/29	8	5	2021/01/07	27	17
2020/11/30	6	4	2021/01/08	19	19
2020/12/01	17	3	2021/01/09	17	14
2020/12/02	11	7	2021/01/10	30	16
2020/12/03	15	9	2021/01/11	24	21
2020/12/04	17	8	2021/01/12	16	11
2020/12/05	9	5	2021/01/13	12	28
2020/12/06	11	8	2021/01/14	22	17
2020/12/07	14	7	2021/01/15	16	19
2020/12/08	8	5	2021/01/16	18	16
2020/12/09	18	6	2021/01/17	16	14
2020/12/10	12	11	2021/01/18	0	0
2020/12/11	14	6	2021/01/19	0	0
2020/12/12	9	7	2021/01/20	0	0
2020/12/13	17	6	2021/01/21	0	0
2020/12/14	15	4	2021/01/22	0	0
2020/12/15	20	8	2021/01/23	0	0
2020/12/16	15	8	2021/01/24	1	3
2020/12/17	16	13	2021/01/25	15	16
2020/12/18	8	4	2021/01/26	5	10
2020/12/19	10	5	2021/01/27	12	16
2020/12/20	26	15	2021/01/28	16	21
2020/12/21	7	3	2021/01/29	13	14
2020/12/22	15	5	2021/01/30	17	24
2020/12/23	9	8	2021/01/31	0	0
2020/12/24	18	6	2021/02/01	0	0
2020/12/25	14	11	2021/02/02	0	0
2020/12/26	9	9	2021/02/03	0	0

Days	CNN	TDS
2021/02/04	0	0
2021/02/05	0	0
2021/02/06	0	0
2021/02/07	0	0
2021/02/08	0	0
2021/02/09	0	0
2021/02/10	0	0
2021/02/11	0	0
2021/02/12	0	0
2021/02/13	0	0
2021/02/14	0	0
2021/02/15	0	0
2021/02/16	0	0
2021/02/17	0	0
2021/02/18	0	0
2021/02/19	0	0
2021/02/20	0	0
2021/02/21	0	0
2021/02/22	0	0
2021/02/23	0	0
2021/02/24	0	0
2021/02/25	0	0
2021/02/26	0	0
2021/02/27	0	0
2021/02/28	0	0
2021/03/01	0	0
2021/03/02	0	0
2021/03/03	0	0
2021/03/04	0	0
2021/03/05	0	0
2021/03/06	0	0
2021/03/07	0	0
2021/03/08	0	0
2021/03/09	0	0
2021/03/10	0	0
2021/03/11	0	0
2021/03/12	0	0
2021/03/13	0	0
2021/03/14	12	18

Days	CNN	TDS
2021/03/15	11	4
2021/03/16	11	10
2021/03/17	13	11
2021/03/18	9	9
2021/03/19	11	10
2021/03/20	6	5
2021/03/21	15	14
2021/03/22	17	15
2021/03/23	11	5
2021/03/24	6	3
2021/03/25	10	5
2021/03/26	9	6
2021/03/27	10	10
2021/03/28	7	2
2021/03/29	11	9
2021/03/30	15	8
2021/03/31	7	7
2021/04/01	1	5
2021/04/02	10	12
2021/04/03	8	8
2021/04/04	3	1
2021/04/05	13	7
2021/04/06	7	4
2021/04/07	7	5
2021/04/08	7	6
2021/04/09	5	6
2021/04/10	15	10
2021/04/11	3	2
2021/04/12	11	7
2021/04/13	13	9
2021/04/14	8	5
2021/04/15	9	6
2021/04/16	8	5
2021/04/17	3	2
2021/04/18	6	4
2021/04/19	13	14
2021/04/20	9	5
2021/04/21	6	7
2021/04/22	6	4



Days	CNN	TDS
2021/04/23	6	4
2021/04/24	5	3
2021/04/25	6	4
2021/04/26	9	6
2021/04/27	5	3
2021/04/28	7	3
2021/04/29	5	2
2021/04/30	9	6
2021/05/01	5	4
2021/05/02	6	3
2021/05/03	12	6
2021/05/04	4	7
2021/05/05	4	2
2021/05/06	4	2
2021/05/07	12	7
2021/05/08	5	4
2021/05/09	10	3
2021/05/10	10	8
2021/05/11	5	3
2021/05/12	3	3
2021/05/13	4	3
2021/05/14	8	5
2021/05/15	3	0
2021/05/16	6	4
2021/05/17	10	5
2021/05/18	3	2
2021/05/19	7	8
2021/05/20	4	4
2021/05/21	5	4
2021/05/22	4	3
2021/05/23	3	1
2021/05/24	10	6
2021/05/25	4	3
2021/05/26	6	6
2021/05/27	1	1
2021/05/28	3	3
2021/05/29	3	4
2021/05/30	2	3
2021/05/31	4	4

Days	CNN	TDS
2021/06/01	5	10
2021/06/02	4	4
2021/06/03	2	1
2021/06/04	11	12
2021/06/05	6	6
2021/06/06	6	6
2021/06/07	5	10
2021/06/08	3	4
2021/06/09	7	5
2021/06/10	5	3
2021/06/11	7	5
2021/06/12	3	4
2021/06/13	5	5
2021/06/14	4	3
2021/06/15	4	3
2021/06/16	4	4
2021/06/17	2	3
2021/06/18	2	3
2021/06/19	3	2
2021/06/20	3	3
2021/06/21	2	2
2021/06/22	4	4
2021/06/23	4	9
2021/06/24	4	6
2021/06/25	7	6
2021/06/26	3	2
2021/06/27	4	6
2021/06/28	15	9
2021/06/29	10	3
2021/06/30	7	4
2021/07/01	11	5
2021/07/02	9	4
2021/07/03	10	6
2021/07/04	10	4
2021/07/05	11	3
2021/07/06	6	4
2021/07/07	10	6
2021/07/08	6	4
2021/07/09	14	4

Days	CNN	TDS
2021/07/10	13	10
2021/07/11	12	7
2021/07/12	13	9
2021/07/13	6	6
2021/07/14	13	4
2021/07/15	8	4
2021/07/16	14	8
2021/07/17	14	9
2021/07/18	9	6
2021/07/19	16	17
2021/07/20	17	9
2021/07/21	29	15
2021/07/22	16	8
2021/07/23	12	6
2021/07/24	15	5
2021/07/25	14	7
2021/07/26	14	6
2021/07/27	13	7
2021/07/28	25	14
2021/07/29	17	9
2021/07/30	9	7
2021/07/31	12	11
2021/08/01	23	18
2021/08/02	13	5
2021/08/03	16	9
2021/08/04	15	10
2021/08/05	19	12
2021/08/06	19	9
2021/08/07	14	9
2021/08/08	7	4
2021/08/09	17	31
2021/08/10	14	12
2021/08/11	10	5
2021/08/12	19	14
2021/08/13	20	15
2021/08/14	26	18
2021/08/15	14	15
2021/08/16	20	14
2021/08/17	22	16

Days	CNN	TDS
2021/08/18	19	13
2021/08/19	18	13
2021/08/20	24	11
2021/08/21	9	6
2021/08/22	29	26
2021/08/23	17	15
2021/08/24	15	18
2021/08/25	19	26
2021/08/26	20	31
2021/08/27	13	13
2021/08/28	16	19
2021/08/29	13	21
2021/08/30	19	20
2021/08/31	30	14
2021/09/01	22	16
2021/09/02	13	19
2021/09/03	16	12
2021/09/04	21	18
2021/09/05	14	12
2021/09/06	16	13
2021/09/07	16	15
2021/09/08	22	20
2021/09/09	16	17
2021/09/10	16	17
2021/09/11	14	22
2021/09/12	0	0
2021/09/13	0	0
2021/09/14	0	0
2021/09/15	0	0
2021/09/16	0	0
2021/09/17	0	0
2021/09/18	0	0
2021/09/19	0	0
2021/09/20	0	0
2021/09/21	24	14
2021/09/22	13	7
2021/09/23	19	11
2021/09/24	21	13
2021/09/25	17	18

Days	CNN	TDS
2021/09/26	21	16
2021/09/27	18	17
2021/09/28	20	20
2021/09/29	13	11
2021/09/30	13	8
2021/10/01	13	13
2021/10/02	15	12
2021/10/03	15	15
2021/10/04	9	10
2021/10/05	15	8
2021/10/06	14	14
2021/10/07	10	11
2021/10/08	13	15
2021/10/09	18	17
2021/10/10	12	7
2021/10/11	9	4
2021/10/12	7	4
2021/10/13	18	8
2021/10/14	14	15
2021/10/15	17	14
2021/10/16	12	13
2021/10/17	11	9
2021/10/18	18	12
2021/10/19	4	2
2021/10/20	8	8
2021/10/21	7	8
2021/10/22	12	14
2021/10/23	1	2
2021/10/24	2	4
2021/10/25	4	4





