



Free Molecular Monte Carlo Model for Lunar Contaminant Transport and Deposition in Permanently Shadowed Regions

JVSRP Final Presentation

Raphael Alves Hailer^{1,2*}

JPL Mentors:

Dr. William A. Hoey ^{2**}

Dr. John M. Alred ²

Dr. John R. Anderson ²

¹ Universidade Estadual de Campinas
(UNICAMP), Cidade Universitária
Zeferino Vaz - Barão Geraldo, Campinas
- SP, 13083-970, Brazil

² Jet Propulsion Laboratory, California
Institute of Technology, 4800 Oak Grove
Dr. M/S 125-109, Pasadena, CA 91109

* < r223852@dac.unicamp.br >

** < william.a.hoey@jpl.nasa.gov >

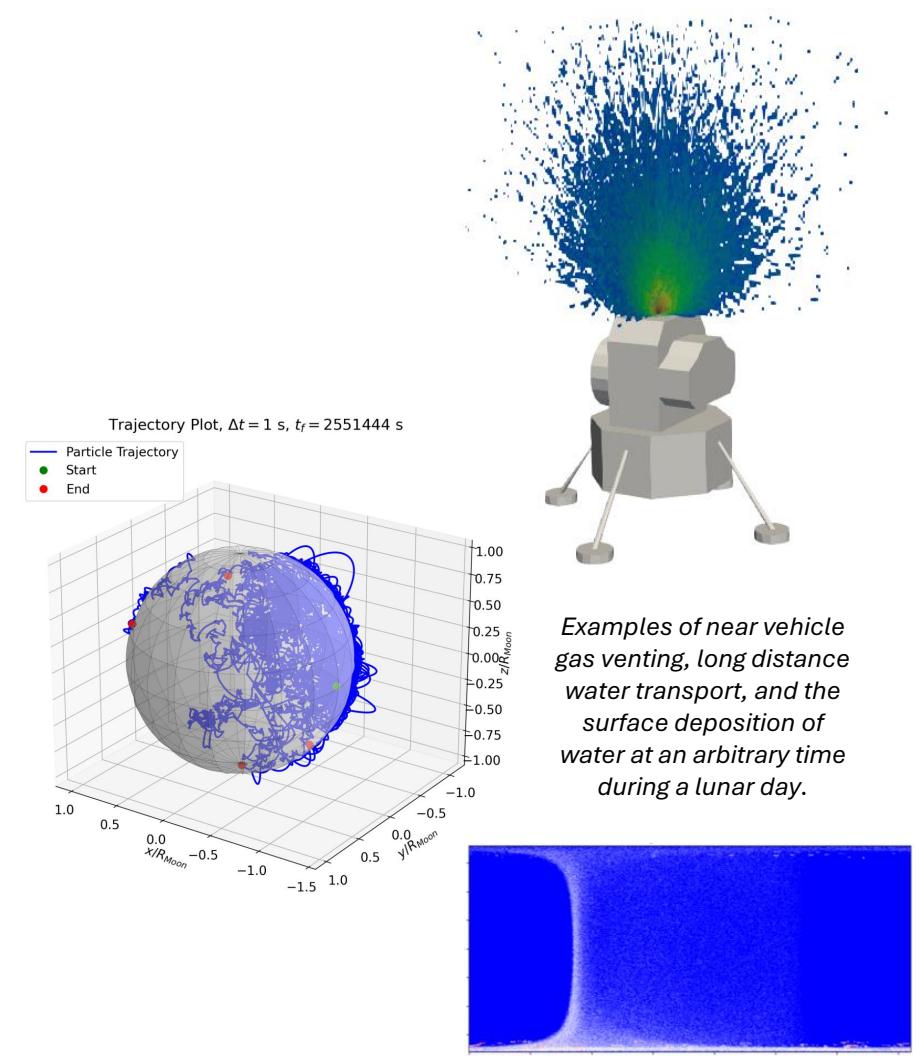


Jet Propulsion Laboratory
California Institute of Technology

This document has been reviewed and determined
not to contain export controlled technical data.
© 2025. All rights reserved.

Table of Contents

- Introduction
- Free Molecular Flow
- Mathematical Foundation
- Validation Against Analytical Solutions
- Deposition and Re-emission
- Permanent Loss Mechanisms
 - Permanently Shadowed Regions
 - Another Form of Permanent Particle Loss
- Results and Discussion
 - Molecular ‘Hopping’
 - Surface Interactions
 - Problem of Spherical Coordinates
 - RK4 Integrator with Cartesian Coordinates
 - Obtaining Inputs from SPARTA Outputs
 - Example of Long-Distance Lunar Transport
- Conclusion and Perspectives for Future Work



Who Am I?

- Mechanical Engineering undergraduate from the **State University of Campinas - Unicamp** (Brazil)
- Concluded the Cycle d'ingénieur at **ENSTA Paris** with advanced studies in space science and technology at École Polytechnique, France
 - Recognized as Bachelor's/Master's (MS/MS) degree
- Research interests:
 - Numerical Modeling and Analysis
 - Optimization
 - Fluid Mechanics/CFD
 - Planetary Science
- Contact info:
 - r223852@dac.unicamp.br (Unicamp)
 - raphael.alves@ensta-paris.fr (ENSTA Paris)



2024 JPL Visiting Student Research Program (JVSRP) fellow with the Contamination Control Engineering group (JPL 353D).

Introduction

- NASA / JPL scientists and engineers are interested in characterizing and quantifying contamination, including organic contamination, to spacecraft, instruments, scientific sites and solar system bodies.
 - *Contamination* is unwanted material, whether molecular or particulate. Contamination can degrade instrument performance, e.g., by depositing a film on a lens, or can degrade mission science, e.g. by mixing spacecraft materials with samples.
 - Sensors in satellites are increasingly sensitive, thus even the smallest levels of contamination must be studied
 - Contamination deposited onto scientific sites can compromise the scientific returns of current and future missions
- Sources of organic contamination for landed missions include:
 - Materials outgassing
 - Descent and ascent propulsion systems
 - Venting processes, particularly for crewed missions
 - Crew environmental and life support processes (ECLSS)
- In conditions of low gravity and vacuum as on the Moon, contaminants can travel vast distances and accumulate in cold regions, including Permanently Shadowed Regions (PSRs) of scientific interest.
 - Artemis future missions to return to the Moon → highly interested in landing near the PSRs at the lunar south pole
- This JVSRP project develops:
 - A framework for simulating free molecular transport, and deposition of molecular contaminant, around an arbitrary airless body accounting for gravity (and escape) and surface adsorption / desorption with a transient temperature map
 - The framework can assume a simple source or can take input from SPARTA DSMC simulations of certain landing vehicle venting processes.
 - Example simulations of free molecular water transport around the Moon including into specific PSRs

Examples of JPL Contamination Control Applications



Jet Propulsion Laboratory
California Institute of Technology

Mars 2020 Perseverance



Alred, J., Martin, M., Hoey, W., et al., 2020. Proc. v. 11489, SPIE Systems Contamination; 1148904. [DOI](#)

Slide credit: John Alred, William Hoey, 2023

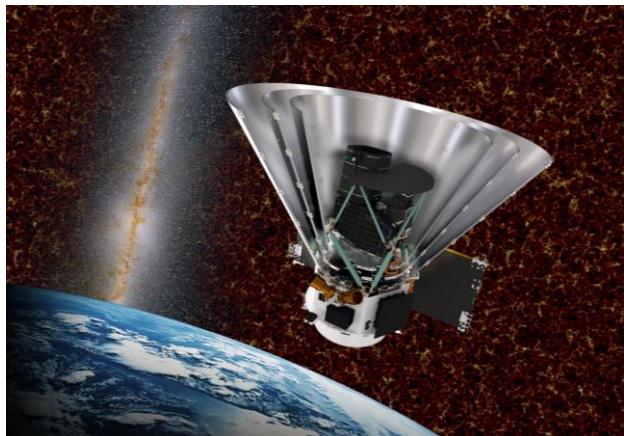
Mission: Sampling Mars with objectives to detect potential organics or biosignatures

Instrument: Returned samples, multiple instruments

Threat: Outgassing contaminants condensing within sample tubes could jeopardize detection

Science missions require control and characterization of contaminants to protect instrument performance and unambiguous detections of, e.g., organics or biosignatures.

SPHEREx



Alred, J. et al., 2021. Proc. IEEE Aerospace Conference. [DOI](#)

Mission: Scientific objectives to image biogenic molecules in the universe

Instrument: Optical characterization

Threat: Water and other biogenic molecules outgassing condensing in telescope causes severe attenuation of throughput

Mission: Objectives to sample Europa's atmosphere, including for organics / biosignatures

Instrument: Mass spectrometer

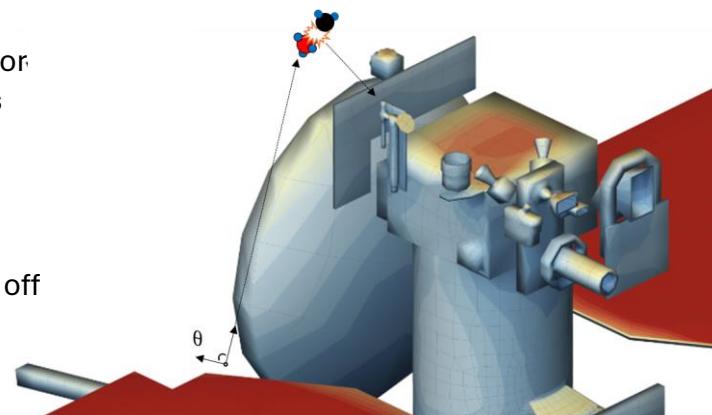
Threat: Outgassing contaminants reflected off atmosphere induce spurious mass spectra

Mars Sample Return Planning



Alred, J., Martin, M., Hoey, W., Soares, C., 2024. NASA Office of Planetary Protection - Planetary Protection Organic Inventory Workshop. [NASA Link](#)

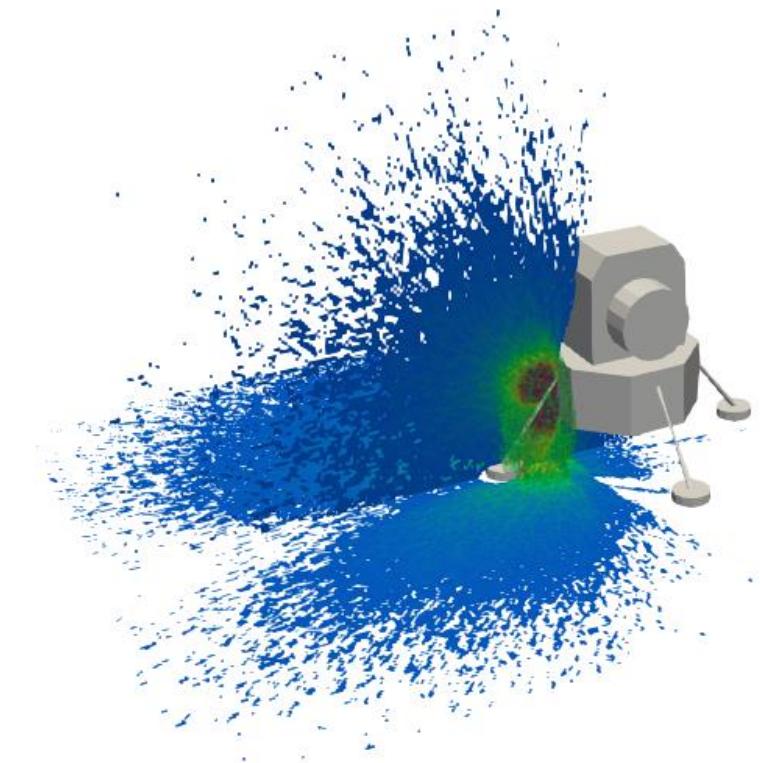
Europa Clipper



Soares, C., Hoey, W., Anderson, J., Ferraro, N., 2019. Proc. 70th Int'l Astronautical Cong. IAC-19-D5.3.12.x50567. [DOI](#)

Introduction

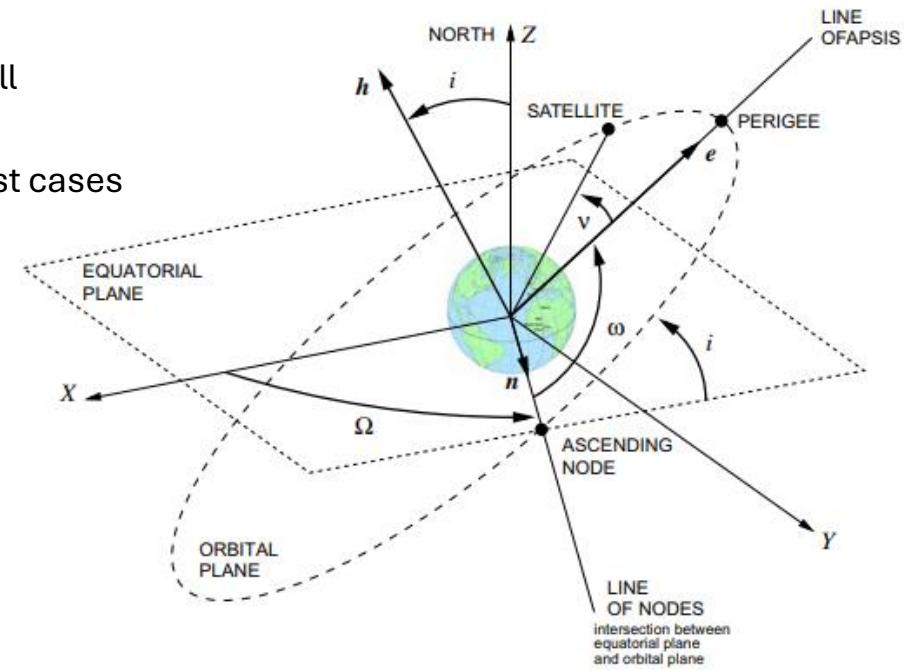
- Present study is a direct follow up to prior studies of near vehicle contamination by venting processes using SPARTA DSMC.
- SPARTA simulation domains resolve near vehicles, are not large enough to study long range contamination → computational burden
 - Additionally, these simulation domains are cubic, neglecting surface curvature
- Here, we consider molecules to be in a collisionless (free molecular) regime ($K_n \gg 100$) at the outer boundaries of such simulations.
- Where are such molecules deposited across the Moon as they continue traveling away from the vehicle?
- How can we couple SPARTA simulations with a new, free-molecular solver to develop a long-distance contamination analysis?



Example of gas venting near a landed vehicle, simulated with SPARTA (JVSRP fellow Paula Gutierrez-Cascales).

Free Molecular Flow

- Objective: determine trajectories for many distinct simulator particles as they travel around the Moon and ‘hop’ along its surface
- In free molecular flow, trajectories of particles follow ballistic paths
- Analytical solution → Keplerian mechanics
 - Solution obtained analytically by truncating elliptical expansions (full mathematical description is disclosed in the final JVSRP report)
 - An algorithm was developed for the solver and remains useful for test cases
- Difficulties with an analytic integration approach:
 - Computationally expensive
 - Involves truncation of series expansions
 - Difficult to scale with millions of particles
- Solution: Implement a numerical integrator
 - Selected method: 3/8 Runge-Kutta 4



Mathematical Foundation

- Consider the coordinate system presented:
 - Spherical coordinates: $(\rho, \theta, \phi) \rightarrow$ Radial distance, Longitude and Latitude
- Final equations governing the motion can be rearranged as a system of Ordinary Differential Equations (ODEs), forming an Initial Value Problem (IVP):
 - Let

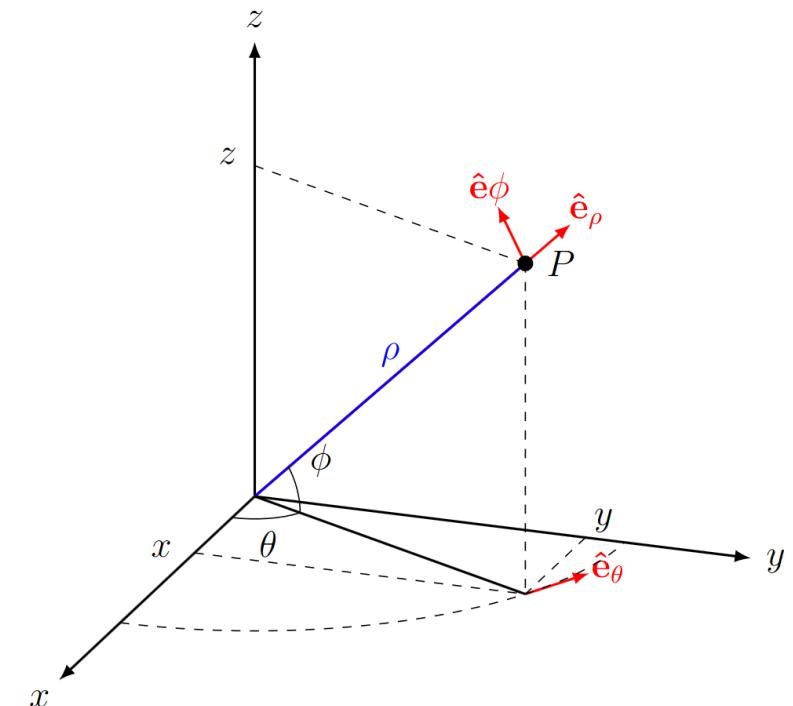
$$u = \begin{pmatrix} \rho \\ \theta \\ \phi \\ \dot{\rho} \\ \dot{\theta} \\ \dot{\phi} \end{pmatrix} \text{ and } f = \begin{pmatrix} \dot{\rho} \\ \dot{\theta} \\ \dot{\phi} \\ \rho\dot{\theta}^2 \cos^2(\phi) + \rho\dot{\phi}^2 - \frac{GM_{Moon}}{\rho^2} \\ \frac{2\dot{\theta}}{\rho \cos(\phi)} [\rho\dot{\phi} \sin(\phi) - \dot{\rho} \cos(\phi)] \\ -\dot{\theta}^2 \sin(\phi) \cos(\phi) - 2\left(\frac{\dot{\rho}}{\rho}\right)\dot{\phi} \end{pmatrix}$$

- Thus, we define the IVP as follows:

$$\begin{cases} \frac{du}{dt} = f \\ u(t=0) = u_0 \end{cases}$$

- Where:

$$u_0 = \{\rho_0, \theta_0, \phi_0, \dot{\rho}_0, \dot{\theta}_0, \dot{\phi}_0\}^T$$



Mathematical Foundation

- 3/8 RK4 method is then implemented:

$$\Delta u_1 = \Delta t f(u^k)$$

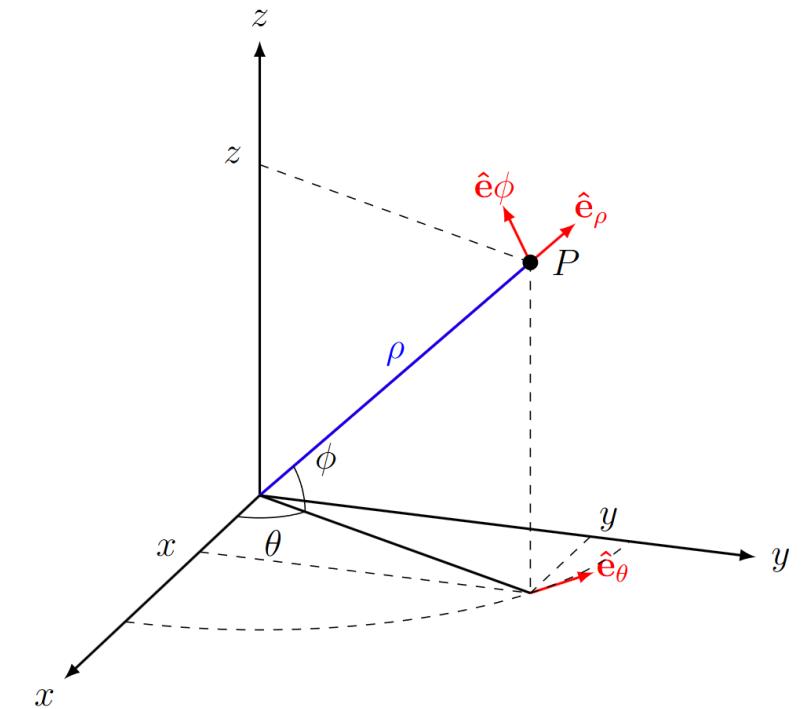
$$\Delta u_2 = \Delta t f\left(u^k + \frac{1}{3}\Delta u_1\right)$$

$$\Delta u_3 = \Delta t f\left(u^k - \frac{1}{3}\Delta u_1 + \Delta u_2\right)$$

$$\Delta u_4 = \Delta t f\left(u^k + \Delta u_1 - \Delta u_2 + \Delta u_3\right)$$

$$u^{k+1} = u^k + \frac{1}{8}(\Delta u_1 + 3\Delta u_2 + 3\Delta u_3 + \Delta u_4)$$

- Order of error: $\mathcal{O}(\Delta t^4)$



Validation Against Analytical Solutions

- The framework is generalizable for airless solar system bodies
- The example simulations shown have been developed for the Moon
 - Approximated as a spherical body
- Data required for simulations:
 - $R_{Moon} = 1738.1 \text{ km}$ (Equatorial radius)
 - $M_{Moon} = 7.346 \times 10^{22} \text{ kg}$
 - $G = 6.6743 \times 10^{-11} \text{ Nm}^2/\text{kg}^2$
- Reference for data:
 - <https://nssdc.gsfc.nasa.gov/planetary/factsheet/moonfact.html>

Moon Fact Sheet



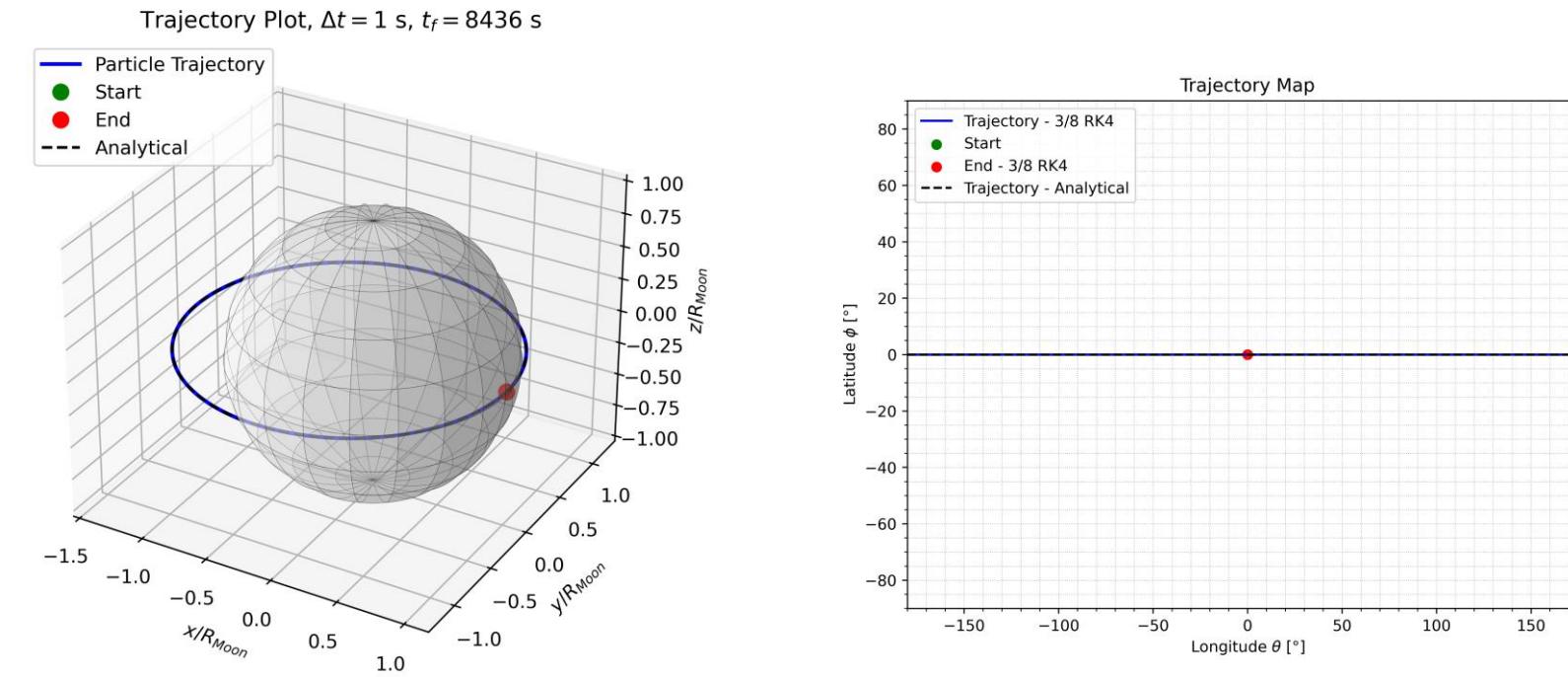
Moon/Earth Comparison

Bulk parameters

	Moon	Earth	Ratio (Moon/Earth)
Mass (10^{24} kg)	0.07346	5.9724	0.0123
Volume (10^{10} km^3)	2.1968	108.321	0.0203
Equatorial radius (km)	1738.1	6378.1	0.2725
Polar radius (km)	1736.0	6356.8	0.2731
Volumetric mean radius (km)	1737.4	6371.0	0.2727
Ellipticity (Flattening)	0.0012	0.00335	0.36
Mean density (kg/m^3)	3344	5514	0.606
Surface gravity (m/s^2)	1.62	9.80	0.165

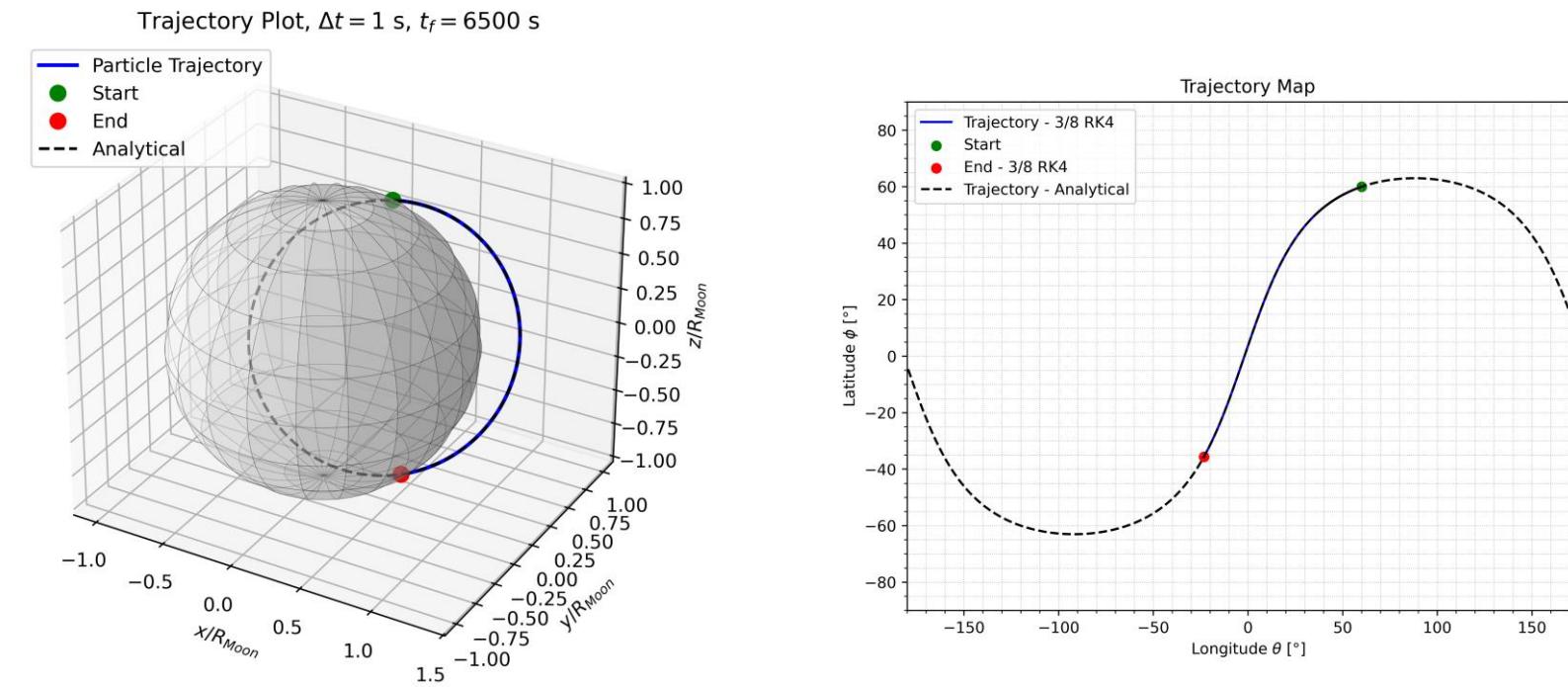
Validation Against Analytical Solutions

- Comparison of trajectories: numerical integration vs. analytical solution.



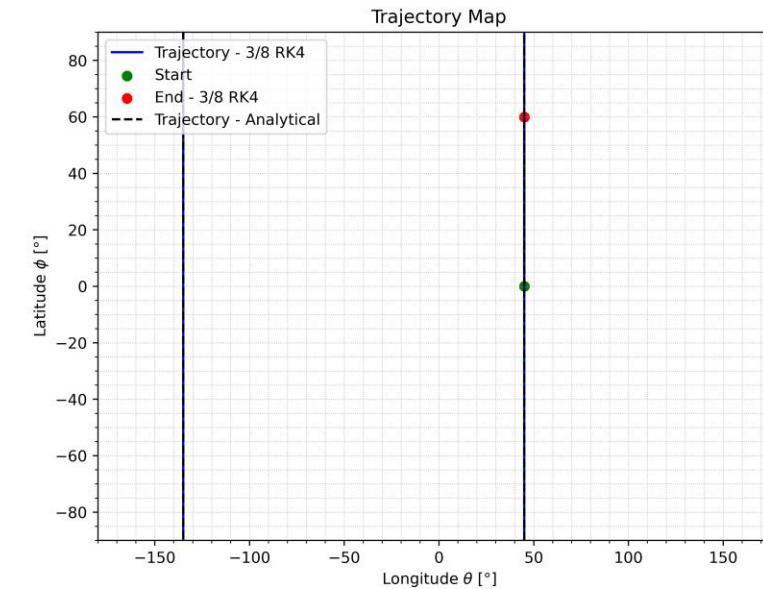
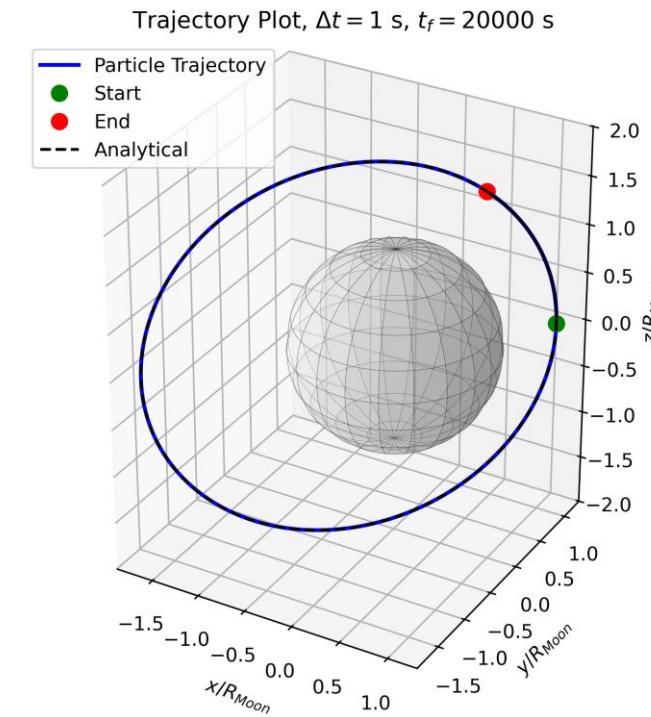
Validation Against Analytical Solutions

- Comparison of trajectories: numerical integration vs. analytical solution.



Validation Against Analytical Solutions

- Comparison of trajectories: numerical integration vs. analytical solution.

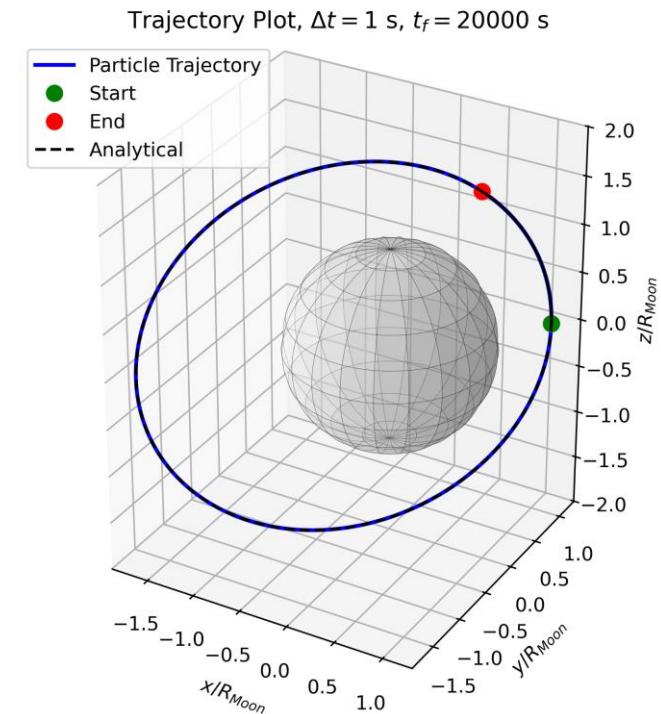
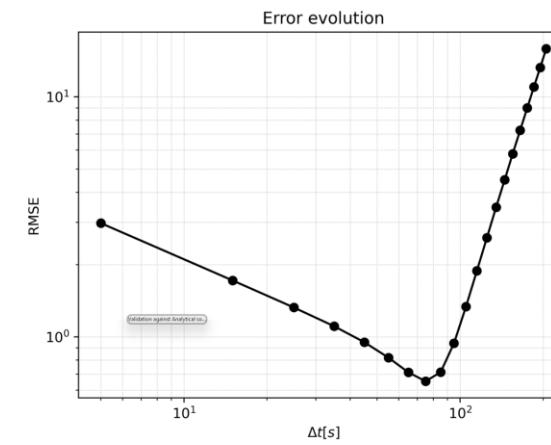


Validation Against Analytical Solutions

- Comparison of trajectories:
 - Simple analysis on RMSE over all points of the orbit trajectory

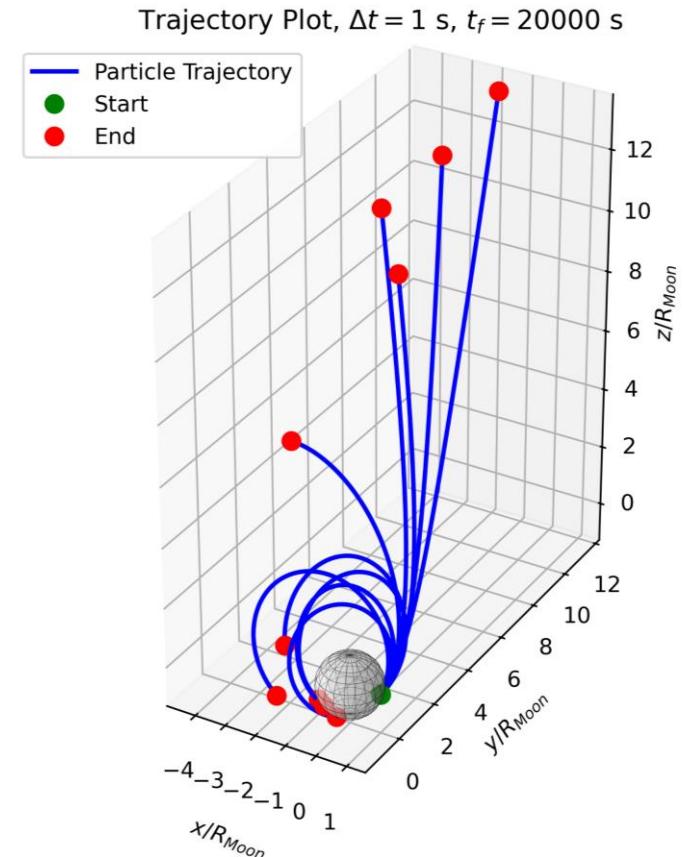
$$\text{RMSE} = \sqrt{\frac{1}{n_t + 1} \sum_{k=0}^{n_t} |\vec{r}_{\text{analytical},k} - \vec{r}_{\text{RK4},k}|^2}$$

- “V-shaped” error evolution, common to RK4
- Mainly occurs due to error accumulation



Working With Multiple Particles

- There are scaling challenges for simulating many particles
 - Next, we focused on increasing the efficiency of calculations
- Selected approach → Vectorized calculations
 - We compute the next time step of all particles simultaneously using vectorization in Python
 - Proved to be more efficient than building ThreadPool or with similar approaches to parallelization (future work may need to investigate)
- Biggest challenge: implement features related to deposition and re-emission of particles simultaneously across a very high number of particles ($>10^6$)



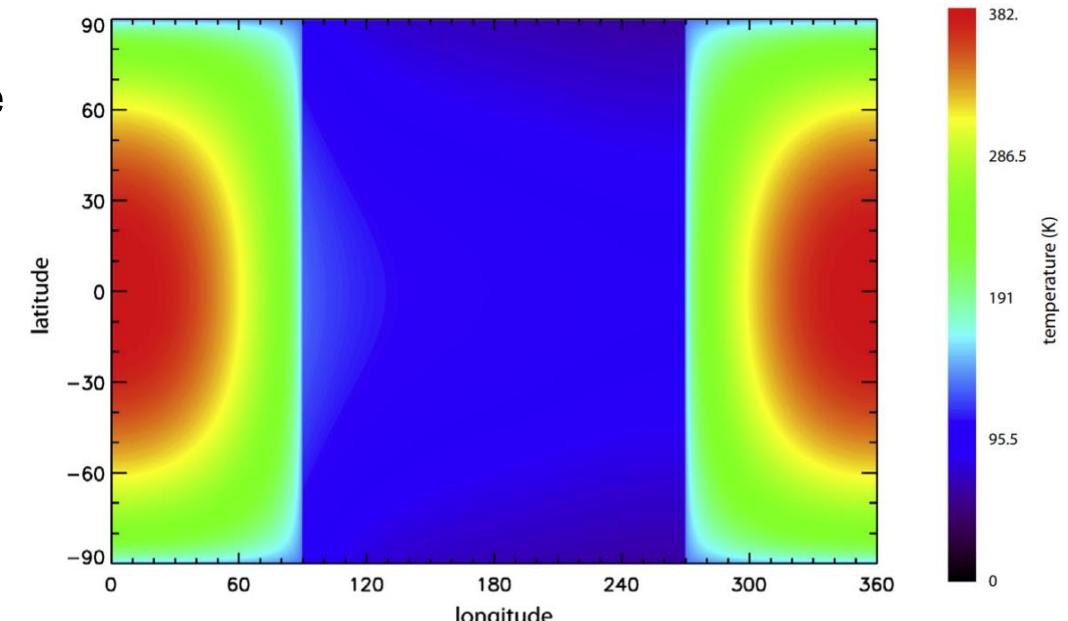
Deposition and Re-emission

- Particle deposition and re-emission (adsorption and desorption) is related to the surface temperature of the landing coordinate for the particle
- We follow an analytical approximation for surface temperature from Hurley et al. (2015), given by:

$$T(\psi) = \max\left(392 \cos(\psi)^{\frac{1}{4}}, 130\right) K \quad (\psi < 90^\circ)$$

$$T = \sum_{i=0}^5 (a_i \theta^i) + 35(\sin(\varphi) - 1)K \quad (\psi > 90^\circ)$$

- Where ψ and φ are the solar zenith angle and the colatitude, respectively
- Consider subsolar point always at the equator
- Revolution period of Moon: 29.5306 Earth days



Hurley, Dana M., et al. "An analytic function of lunar surface temperature for exospheric modeling." *Icarus* 255 (2015): 159-163.

Deposition and Re-emission

- After determining the position a particle landed, we compute the residence time of the particle following Parvathy Prem's PhD dissertation (2017)*:

$$t_{res} = \frac{1}{\nu_0} e^{\left(\frac{E_a}{k_B T}\right)}$$

- Where ν_0 and E_a are the lattice vibrational frequency and the binding energy of the molecule, respectively. For H₂O, we have $\nu_0 = 2 \times 10^{12} \text{ s}^{-1}$ and $E_a = 6.65 \times 10^{-20} \text{ J}$.
- In addition, $k_B = 1.380649 \times 10^{-23} \text{ J/K}$ is the Boltzmann constant
- Using the residence time, we must decide whether a particle is deposited or re-emitted, and if re-emitted then when should we re-emit it
- The present study only considers water molecules; however, the framework has been built in a generalized way for future multispecies studies

*Prem, Parvathy. *DSMC simulations of volatile transport in a transient lunar atmosphere and ice deposition in cold traps after a comet impact*. PhD dissertation, 2017.

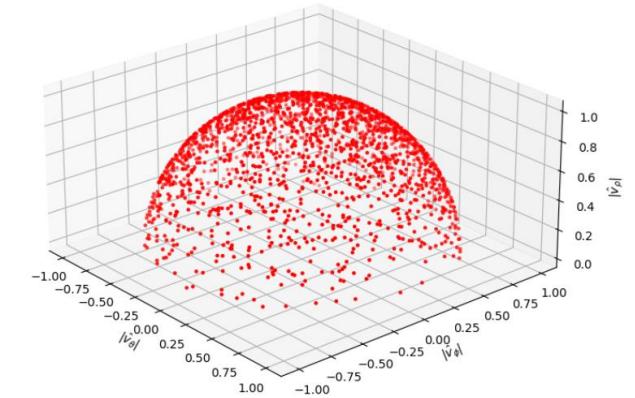
Deposition and Re-emission

- Consider at time step t_k that a particle has radial distance $\rho \leq R_{Moon}$. We perform a linear interpolation to approximate the landing location and time $t_{landing}$, with $t_{k-1} < t_{landing} < t_k$.
- Then, we compute the residence time considering the temperature at the landing coordinate $(\theta_{landing}, \phi_{landing})$. If $t_{k-1} + t_{landing} < t_k$, then the particle can be re-emitted in the current time step
- Otherwise, the particle is considered to have been deposited, and its index is separated from all other particles.
- For all particles being re-emitted, be it now or later, two important steps are necessary to assess its emission velocity
 - Use the cosine angular distribution to sample the direction
 - Use the velocity distribution in thermal equilibrium to sample the magnitude

Deposition and Re-emission

- To sample the direction of the particle, we make use of the cosine angular distribution
- Consider \mathcal{R}_1 and \mathcal{R}_2 to be two random numbers sampled from a uniform Probability Density Function (PDF) with range $[0,1)$. Thus, we sample the unit vectors for all velocity components using the procedure on the right:^{*}

*Greenwood, John: The correct and incorrect generation of a cosine distribution of scattered particles for monte-carlo modelling of vacuum systems. Vacuum, 67(2):217– 222, 2002,
ISSN 0042-207X. <https://www.sciencedirect.com/science/article/pii/S0042207X02001732>.



$$\sin(\psi_1) = \sqrt{\mathcal{R}_1}$$

$$\cos(\psi_1) = \sqrt{1 - \sin^2(\psi_1)}$$

$$\psi_2 = 2\pi\mathcal{R}_2$$

$$\hat{v}_\rho = \cos \psi_1 \hat{e}_\rho$$

$$\hat{v}_\theta = \sin \psi_1 \cos \psi_2 \hat{e}_\theta$$

$$\hat{v}_\phi = \sin \psi_1 \sin \psi_2 \hat{e}_\phi$$

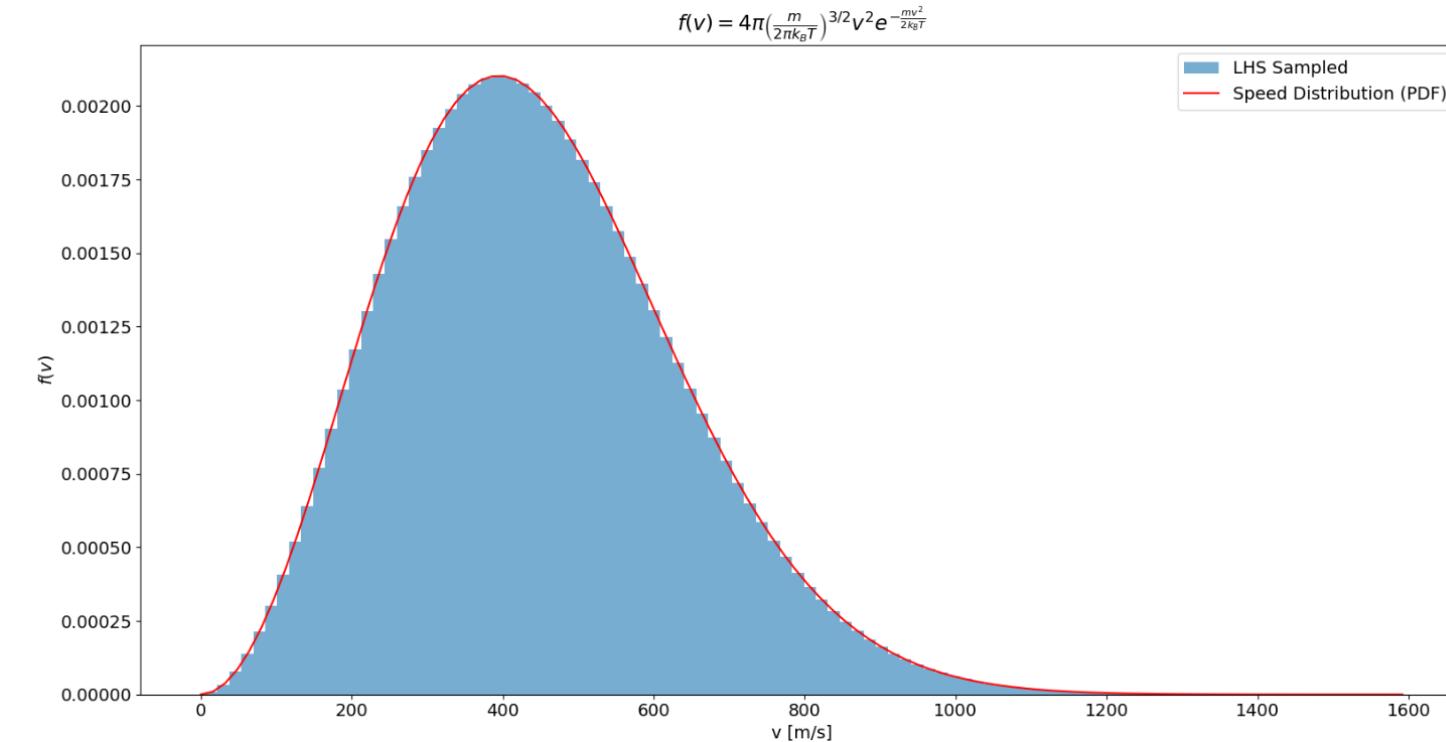
Deposition and Re-emission

- Regarding magnitude, we assume that the surface is in thermal equilibrium, meaning the velocity must be sampled from a Maxwell-Boltzmann velocity distribution $f(v)$, given by

$$f(v) = 4\pi \left(\frac{m}{2\pi k_B T} \right)^{3/2} v^2 e^{-\frac{mv^2}{2k_B T}}$$

- We sample from this distribution for $v \in [0, 6v_{mp}]$, where $v_{mp} = \sqrt{\frac{2k_B T}{m}}$ is the most probable speed (Thermal speed)
 - Maximum speed to sample from is selected this way to include the Moon's escape velocity for water molecules sampling
- Sampling process uses Latin Hypercube Sampling (LHS) technique

Deposition and Re-emission



$$f(v) = 4\pi \left(\frac{m}{2\pi k_B T}\right)^{3/2} v^2 e^{-\frac{mv^2}{2k_B T}}$$

Illustration of the LHS sampling method with the speed distribution function, considering $T = 300$ K and $m = 5.312 \times 10^{-26}$ kg, which is the mass of an O₂ molecule.

Deposition and Re-emission

- For particles that were deposited, we consider another approach. We create a longitude-latitude mesh of the Moon and we compute the sublimation rate only for cells that have particles deposited. The resolution of this grid is set to $\Delta\theta = \Delta\phi = 1^\circ$
- Sublimation rate equation → Hertz-Knudsen (Knudsen-Langmuir) equation

$$S_0 = \alpha \left(\frac{\mathcal{M}}{2\pi \bar{R}T} \right)^{1/2} (p_{sat} - p)$$

- Where \mathcal{M} , p_{sat} , p and \bar{R} are the molecular weight, saturation vapor pressure, back pressure and the universal gas constant, respectively. Since particles are traveling in vacuum, we consider $\alpha = 1$ and $p = 0$
- Expression for p_{sat} : Murphy and Koop (2005)* expression (Good for $T \in [110, 273.15]$ K):

$$p_{sat} = \exp \left(9.550426 - \frac{5723.265}{T} + 3.53068 \ln(T) - 0.00728332T \right)$$

*Murphy, Daniel M and Thomas Koop: Review of the vapour pressures of ice and supercooled water for atmospheric applications. Quarterly Journal of the Royal Meteorological Society: A journal of the atmospheric sciences, applied meteorology and physical oceanography, 131(608):1539–1565, 2005.

Deposition and Re-emission

- To compute the necessary number of particles inside each cell to re-emit, we approximate the area of the cell:

$$A_{cell} \approx R_{Moon}^2 \cos(\phi_{center}) \Delta\theta \Delta\phi$$

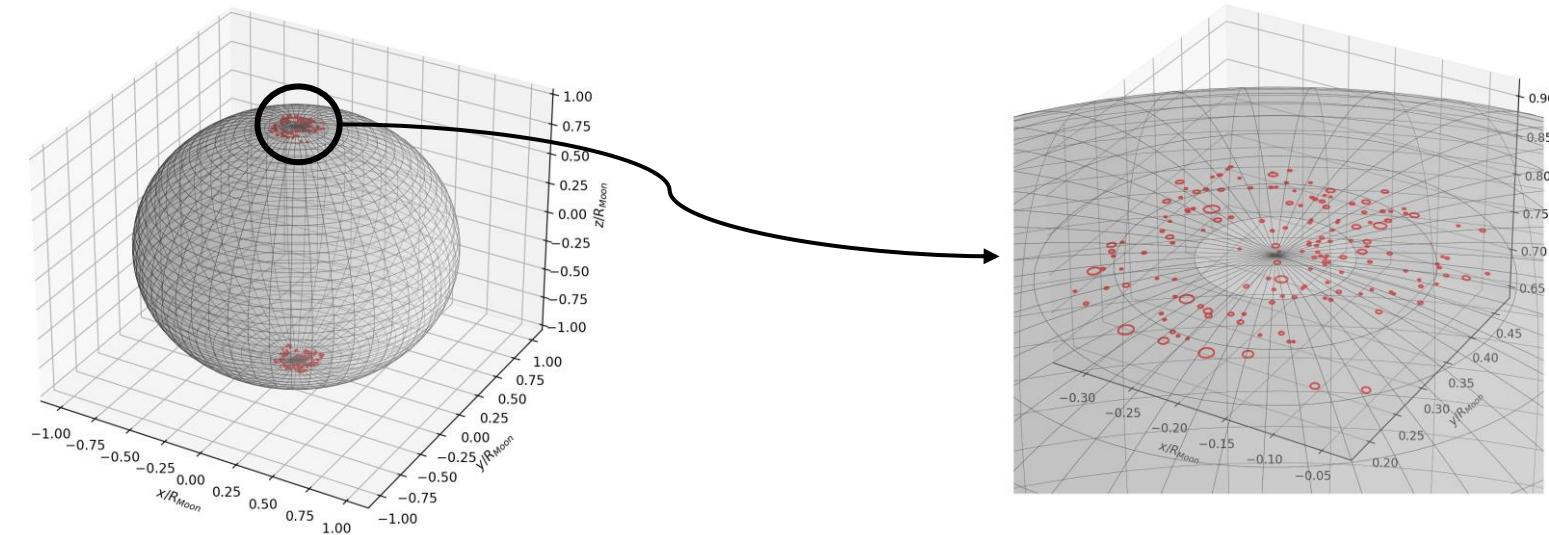
- With ϕ_{center} being the latitude at the center of the cell (Not the barycenter latitude, but rather $\phi_{center} = (\phi_j + \phi_{j+1})/2$)
- Number of particles is computed considering the ratio between real particles and virtual particles, denoted as F_{num}

$$n_p = \frac{A_{cell} S_0 \Delta t}{M F_{num}}$$

- Since n_p is a float result, then we sample an integer value using a Poisson distribution, with mean value equal to n_p

Permanently Shadowed Regions

- To enhance the Model even further, Permanently Shadowed Regions (PSR) were added to the code
- The regions are considered to be circular craters in the north and south poles of the Moon, mapped by the Lunar Reconnaissance Orbiter (LRO)* mission (<https://www.lroc.asu.edu/atlasses/psr/list>)



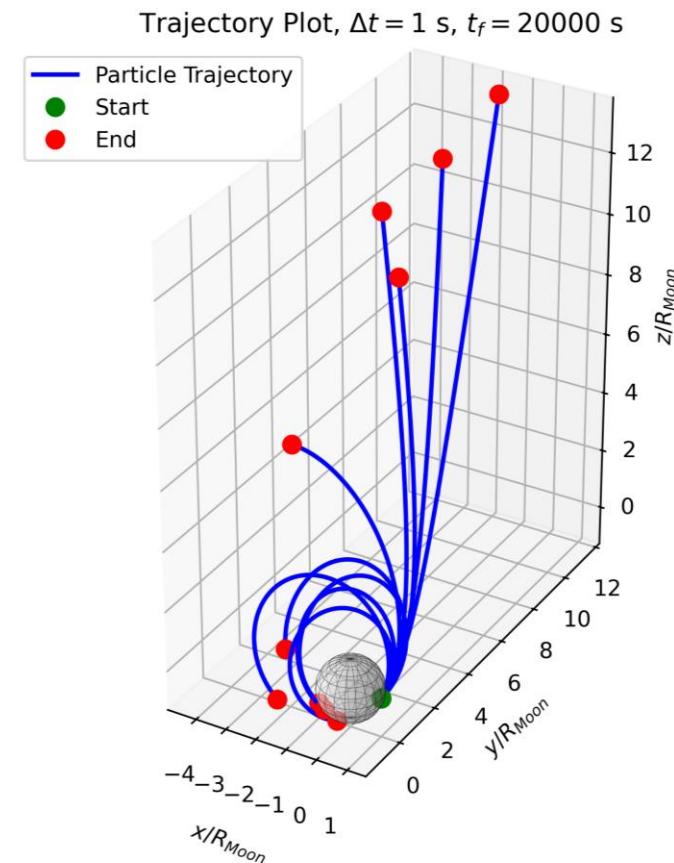
*Robinson, M.S., et. al. (2010). Lunar Reconnaissance Orbiter Camera (LROC) Instrument Overview, *Space Science Reviews*, Vol 150, pp. 81-124.

Permanently Shadowed Regions

- If any particle lands in a PSR, it gets removed from any further simulations, as it is considered permanently trapped.
- Angular radius of each crater from this model is computed considering that the area of the crater measured from the LRO mission is the area corresponding to a spherical cap on the Moon's surface
- These approximations were made to simplify the implementation in the solver
- Area of spherical cap: $A_{cap} = 2\pi R_{Moon}^2 (1 - \cos(\alpha_R))$
- Where α_R is the angular radius of the crater
- Given $(\theta_{landing}, \phi_{landing})$, we compare either with the PSRs in the northern or the southern hemispheres
 - If the measured angular radius are smaller or equal than α_R , then the particle is considered to be in a PSR

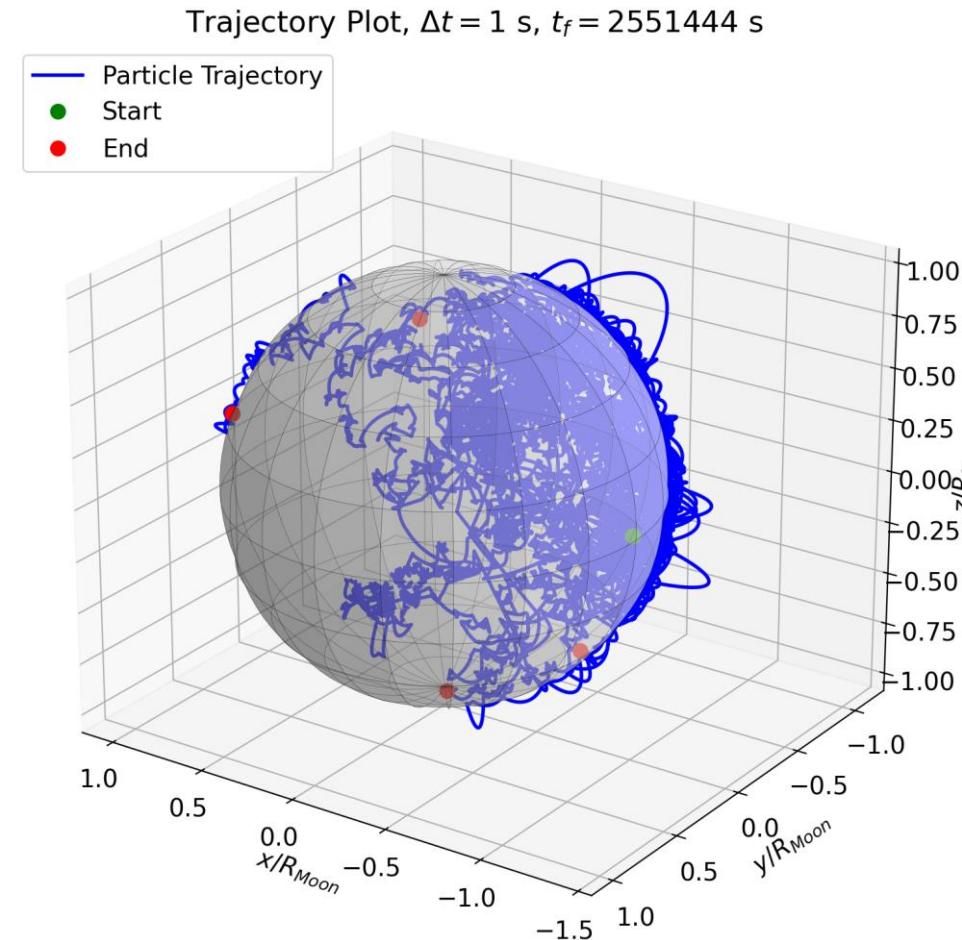
Another Form of Permanent Particle Loss

- Currently, the solver also computes the escape velocity of the celestial body considered
- If in the re-emission process the sampled velocity is greater or equal to the escape velocity, we artificially introduce “NaN” results, to omit them from any plots
- This prevents the solver to compute unnecessary trajectories



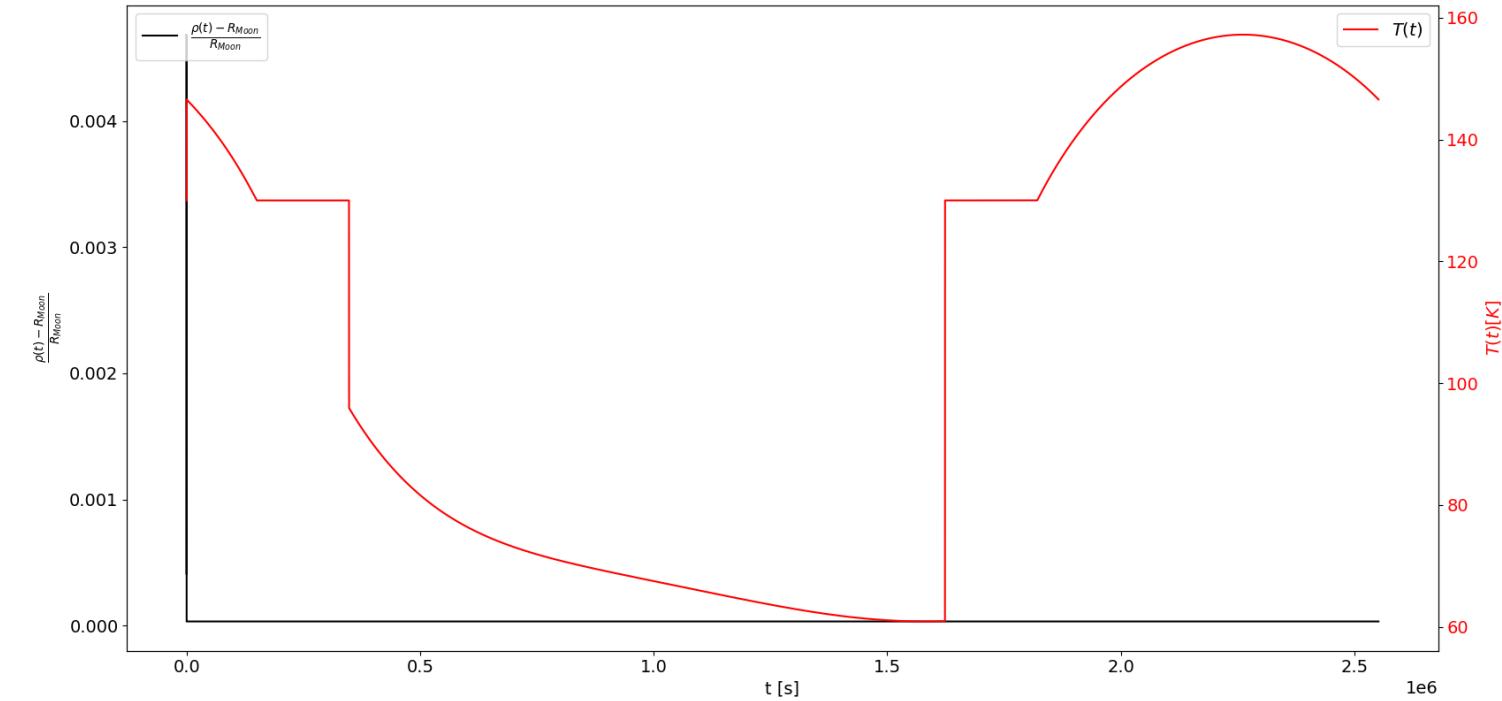
Results and Discussion: Molecular ‘Hopping’

- Initially, it was possible to compute for a whole lunar day the trajectory of few particles, and to observe the temperature of the particle coordinates through time
- Sample result considering 4 particles (each particle is a water molecule)



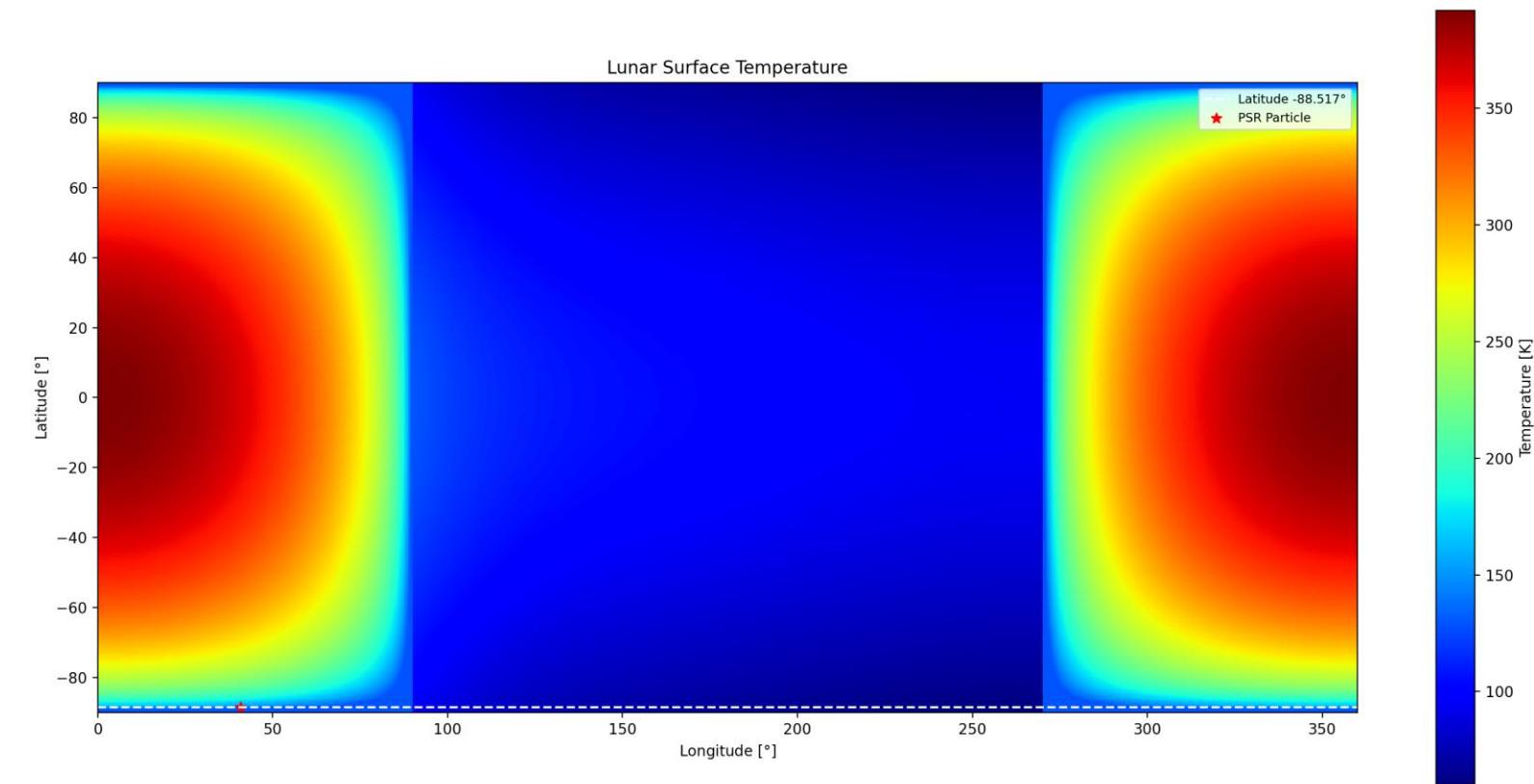
Results and Discussion: Surface Interactions

- Example of particle landing in a PSR → It's removed from the simulation entirely!
 - $(\theta_{landing}, \phi_{landing}) = (40.935^\circ, -88.517^\circ)$



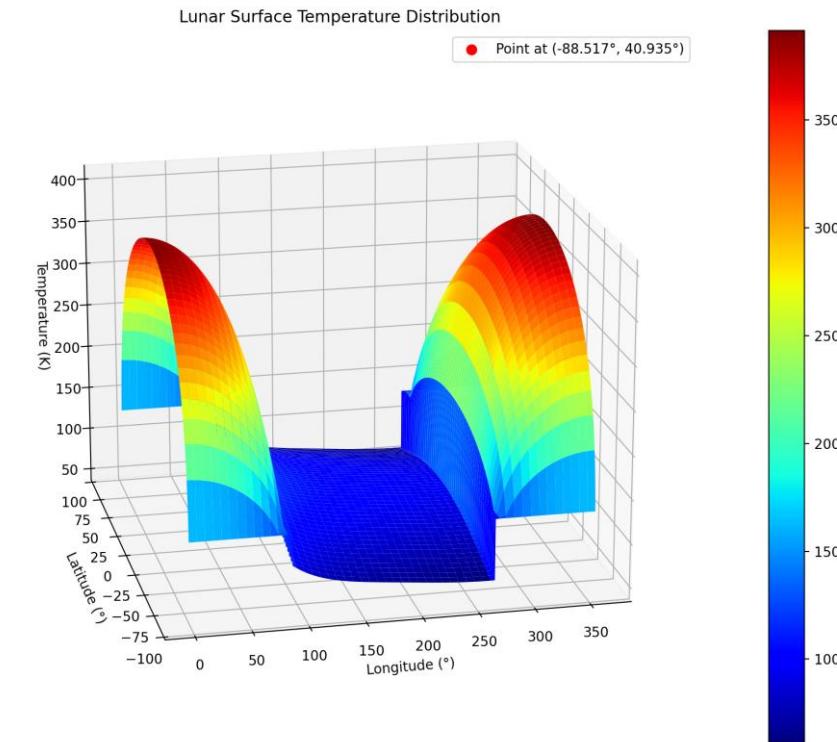
Results and Discussion: Surface Interactions

- Example of particle landing in a PSR → It's removed from the simulation entirely!
 - $(\theta_{landing}, \phi_{landing}) = (40.935^\circ, -88.517^\circ)$



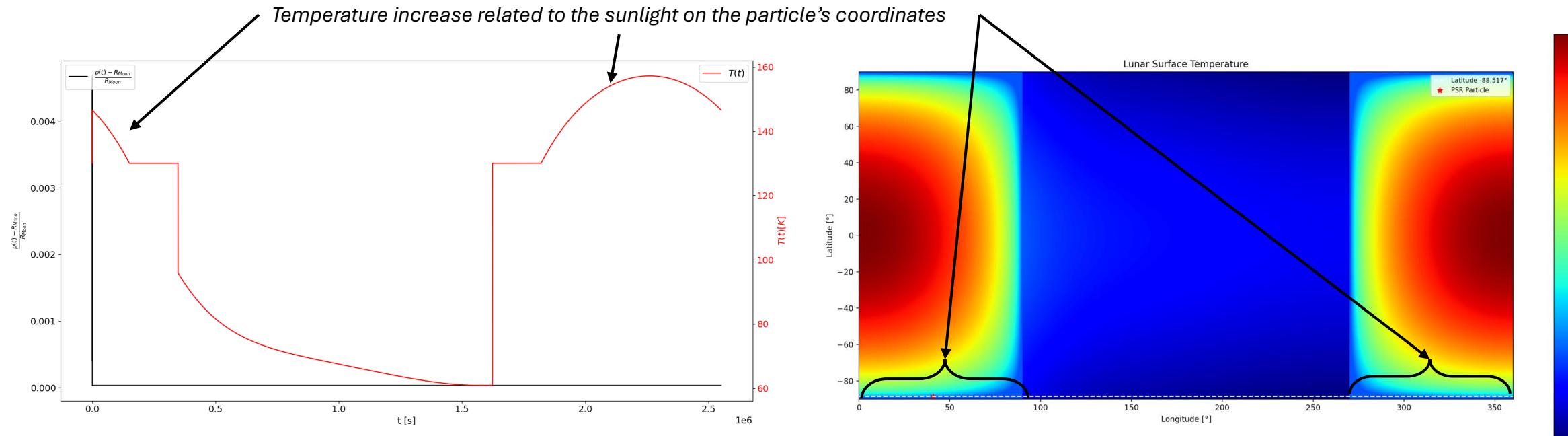
Results and Discussion: Surface Interactions

- Example of particle landing in a PSR → It's removed from the simulation entirely!
 - $(\theta_{landing}, \phi_{landing}) = (40.935^\circ, -88.517^\circ)$



Results and Discussion: Surface Interactions

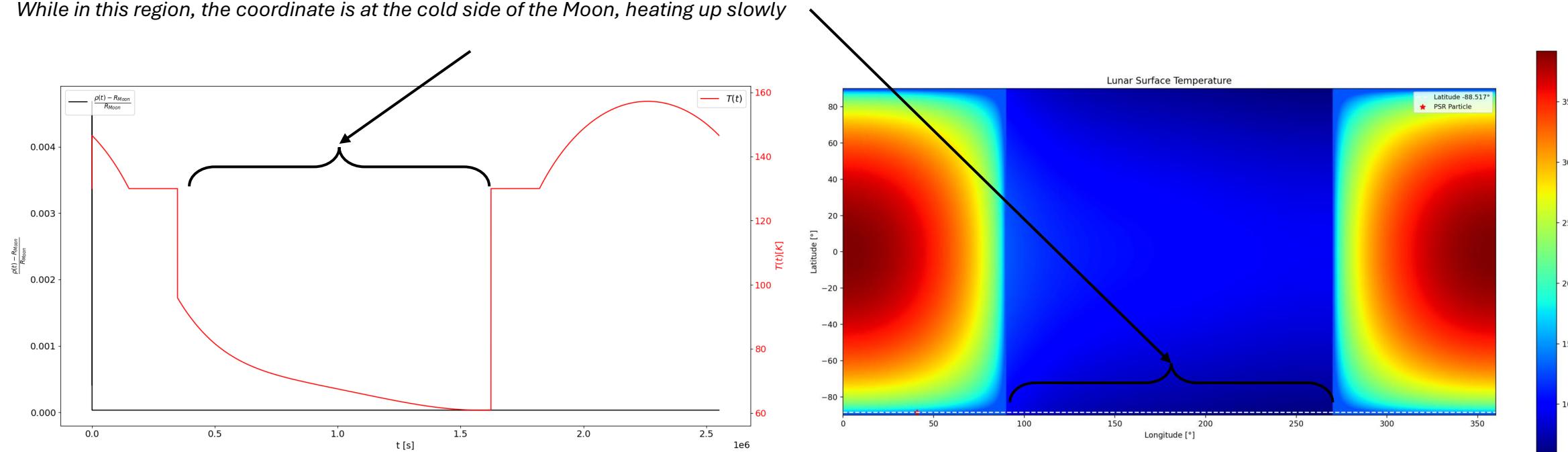
- Example of particle landing in a PSR → It's removed from the simulation entirely!



Results and Discussion: Surface Interactions

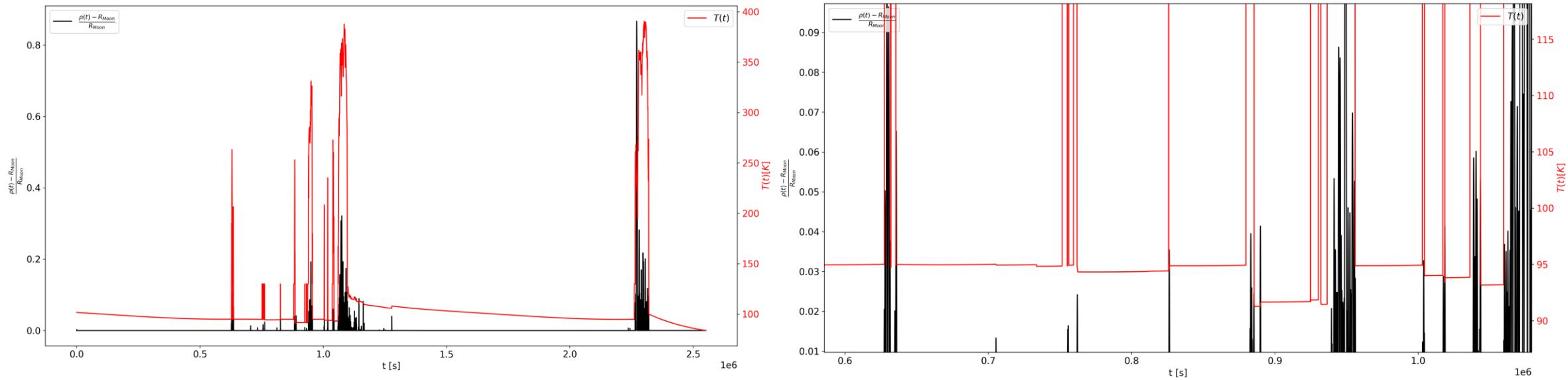
- Example of particle landing in a PSR → It's removed from the simulation entirely!

While in this region, the coordinate is at the cold side of the Moon, heating up slowly



Results and Discussion: Surface Interactions

- Temperature threshold for particles to be re-emitted → between 90 and 95 K approximately
- This indicates that both models (residence time and sublimation model) agree with each other



Problem of Spherical Coordinates

- After tests with the spherical coordinates RK4 integrator, a singularity was observed near the poles
- Whenever a new velocity is generated, we have new values of v_ρ , v_θ and v_ϕ
- In order to compute $\dot{\rho}$, $\dot{\theta}$ and $\dot{\phi}$, we would need to use these velocities

$$\vec{v} = v_\rho \hat{e}_\rho + v_\theta \hat{e}_\theta + v_\phi \hat{e}_\phi \longleftrightarrow \vec{v} = \dot{\rho} \hat{e}_\rho + \rho \dot{\theta} \cos(\phi) \hat{e}_\theta + \rho \dot{\phi} \hat{e}_\phi$$

- Problem arises when near the poles $\rightarrow \cos(\phi)$ approaches zero
- Thus, many “NaN” results were being generated, and particle data was lost
 - Major problem, since the PSRs are located near the poles
- **Solution:** build a time integrator in cartesian coordinates every time we need to update position

RK4 Integrator with Cartesian Coordinates

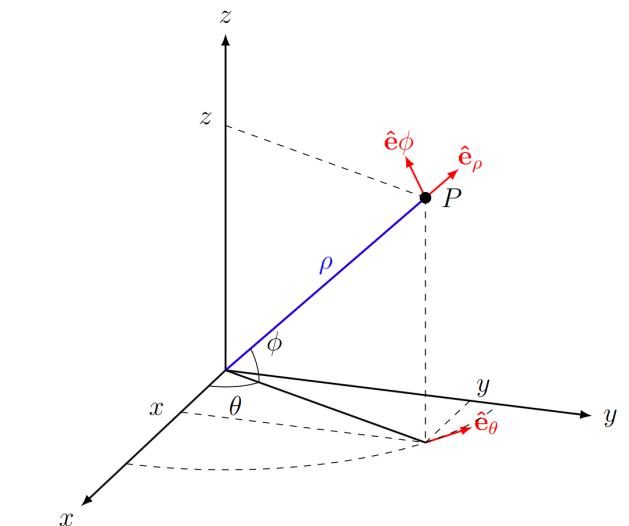
- New system of differential equations to solve:

$$\begin{cases} \frac{du}{dt} = f \\ u(t=0) = u_0 \end{cases}, \quad u = \begin{pmatrix} x \\ y \\ z \\ \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix} \text{ and } f = \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ -\frac{GM_{Moon}}{\rho^2} \cos(\phi) \cos(\theta) \\ -\frac{GM_{Moon}}{\rho^2} \cos(\phi) \sin(\theta) \\ -\frac{GM_{Moon}}{\rho^2} \sin(\phi) \end{pmatrix}$$

- Here we still make use of spherical coordinates, to simplify the formulation, as well as to implement all features of the code other than the moving function

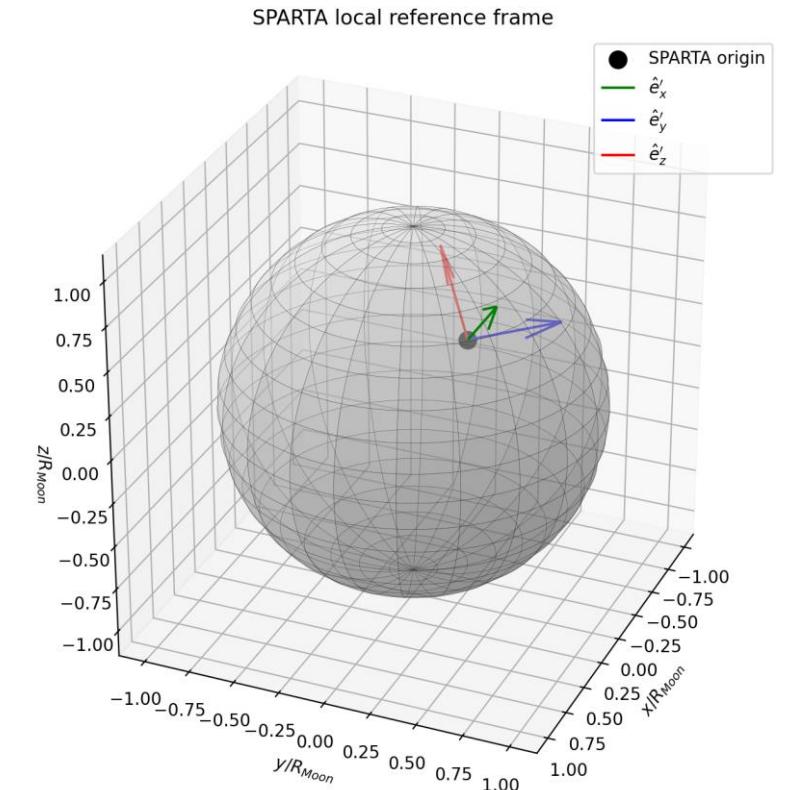
$$\begin{cases} \rho = \sqrt{x^2 + y^2 + z^2} \\ \theta = \text{atan2}(y, x) \\ \phi = \text{atan} \frac{z}{\sqrt{x^2 + y^2}} \end{cases}$$

$$\begin{cases} x = \rho \cos(\phi) \cos(\theta) \\ y = \rho \cos(\phi) \sin(\theta) \\ z = \rho \sin(\phi) \end{cases}$$



Obtaining Inputs from SPARTA Outputs

- Assume we insert particles in the free molecular flow solver by inserting local coordinates extracted from a SPARTA simulation
 - Fixed reference frame: $(\mathcal{O}, (\hat{e}_x, \hat{e}_y, \hat{e}_z))$
 - Local coordinate system: $(\mathcal{O}', (\hat{e}'_x, \hat{e}'_y, \hat{e}'_z))$
- We transform the coordinates from the local to the fixed reference frame
 - Assume \hat{e}'_x and \hat{e}_ρ colinear, as well as \hat{e}'_y and \hat{e}_θ
- By doing these considerations, we can just provide the initial coordinate of the SPARTA reference frame in the fixed reference frame and translate/rotate the coordinates from the local reference frame (obtained from simulation) back to the fixed reference frame
- For the local velocity vectors, the appropriate rotation matrix is computed to obtain the velocities in the fixed reference frame



Example of Long-Distance Lunar Transport

- **Test case:**

- 200,000 simulator particles released near the lunar south pole
- Local reference frame position → $R_{ref} = (x_{ref}, y_{ref}, z_{ref}) = R_{Moon} \left(\cos\left(\frac{0.9\pi}{2}\right), 0, -\sin\left(\frac{0.9\pi}{2}\right) \right)$
- Local velocities and positions → random values in specific range
 - $(x, y, z) \in [0, 1000 \text{ m}]^3$
 - $(v_x, v_y, v_z) \in [100 \text{ m/s}, 200 \text{ m/s}]^3$
- Global $\Delta t \rightarrow 20 \text{ s}$
- Final simulation time → one lunar day, t_{day}
- Results are saved at $t = 0.5t_{day}$ and $t = t_{day}$, where $t_{day} = 2551440 \text{ s}$
- Only water molecules considered in this example
- $F_{num} = 1 \times 10^{15}$ (ratio between real and virtual particles)

$$(\theta, \phi) = (0, -81^\circ)$$

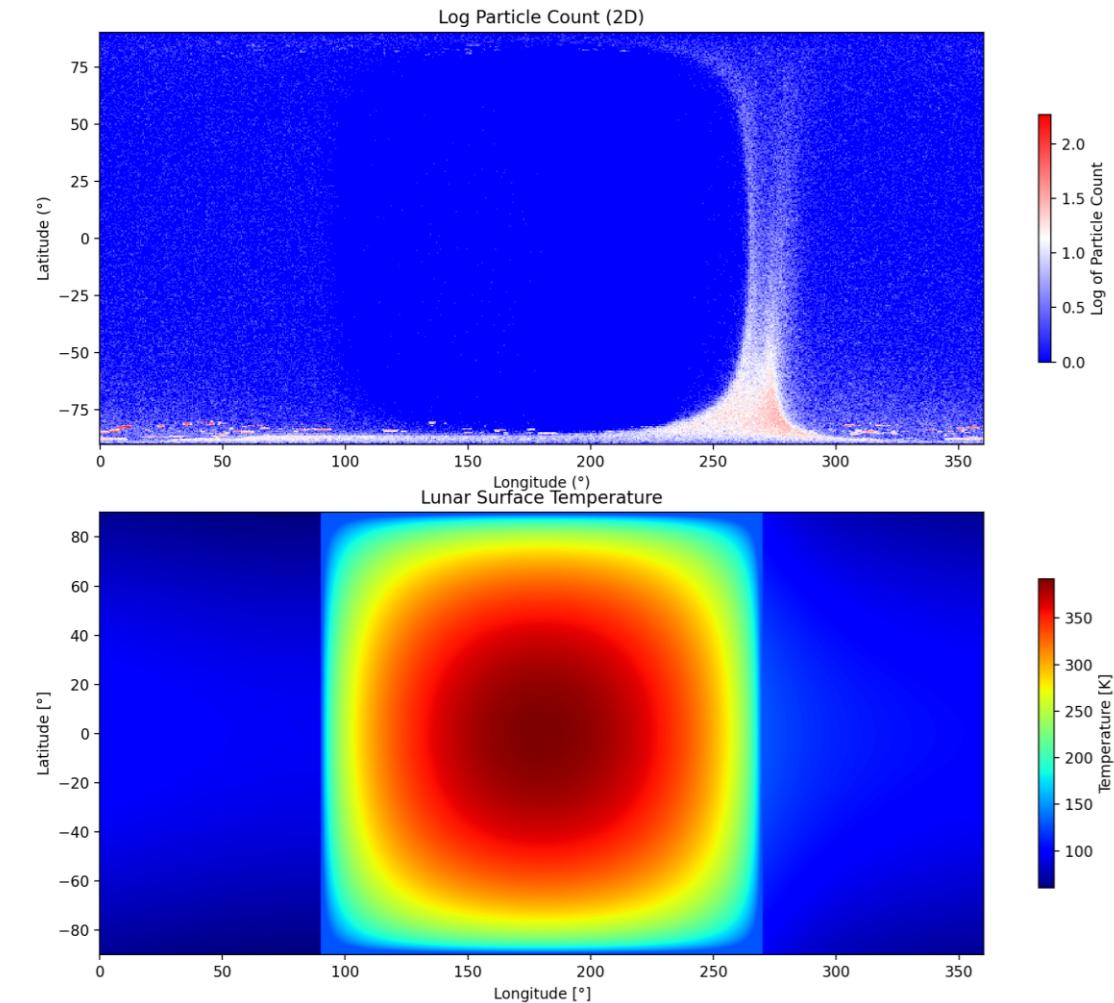
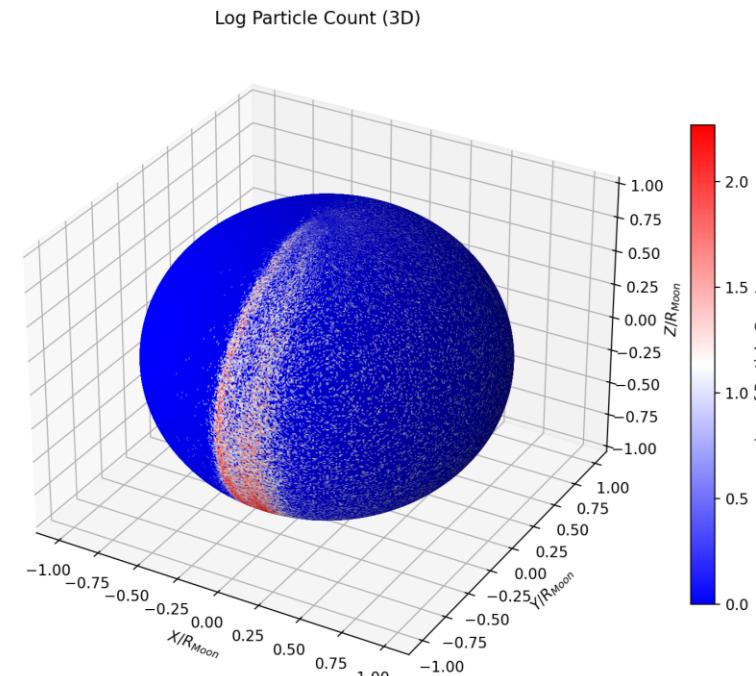

Example of Long-Distance Lunar Transport

- Performance:
 - Use of array jobs: 200 particles per job instance, with total of 1000 instances → 200,000 particles
 - Each instance uses one single CPU
 - Average time per simulation: 4336.83s
 - Maximum at 256 instances (CPUs) allowed to be used at a time
- Supposing that each 256 job instances start only when all others end (conservative estimate), then the total time of the simulation is roughly 17347.32 s, or 4 hours, 49 minutes and 7 seconds.
- Example of job efficiency for a single instance:

```
[rhailer@gattaca2-hn1 Output_files]$ seff 1540050-1
Job ID: 1540050
Array Job ID: 1540050_1000
Cluster: slurm
User/Group: rhailer/rhailer
State: COMPLETED (exit code 0)
Cores: 1
CPU Utilized: 01:12:19
CPU Efficiency: 99.25% of 01:12:52 core-walltime
Job Wall-clock time: 01:12:52
Memory Utilized: 165.98 MB
Memory Efficiency: 2.03% of 8.00 GB
```

Example of Long-Distance Lunar Transport

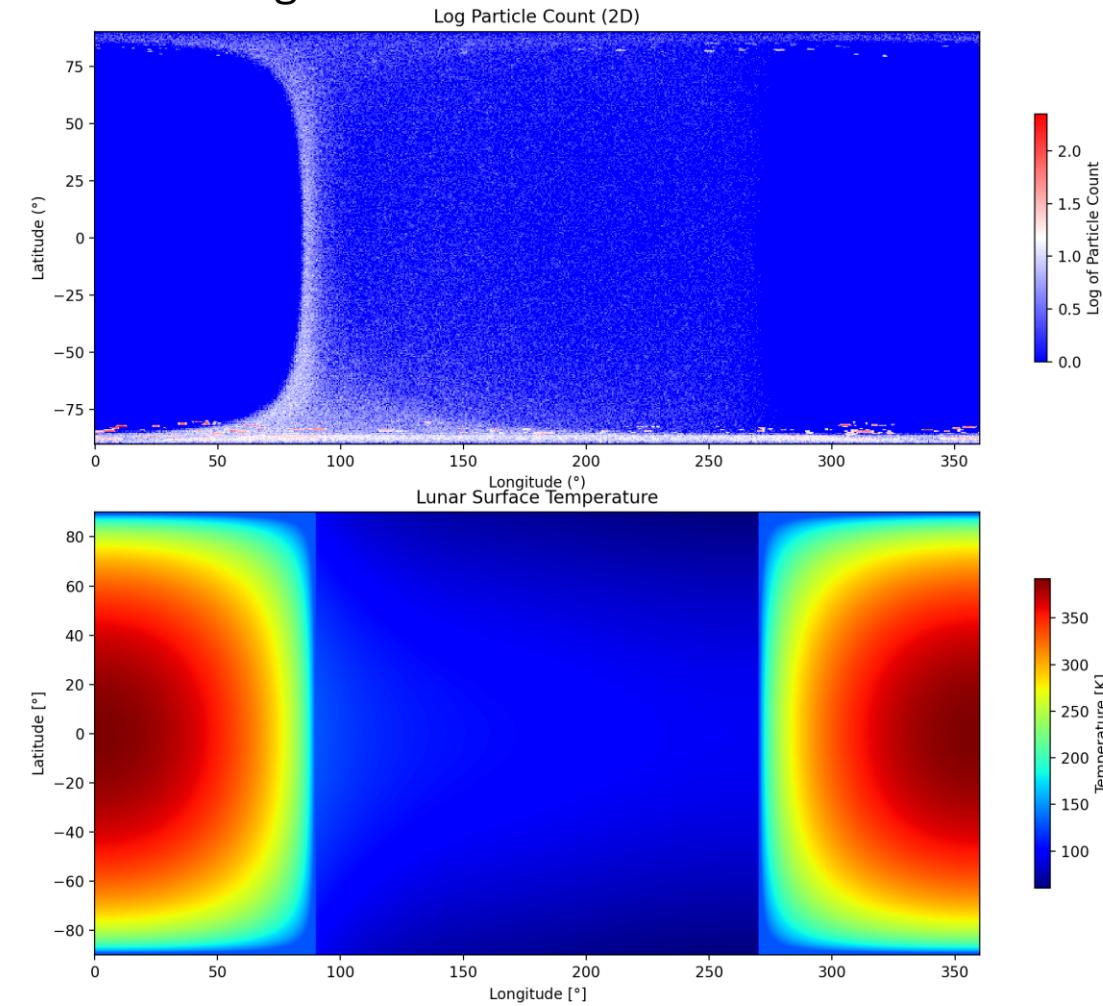
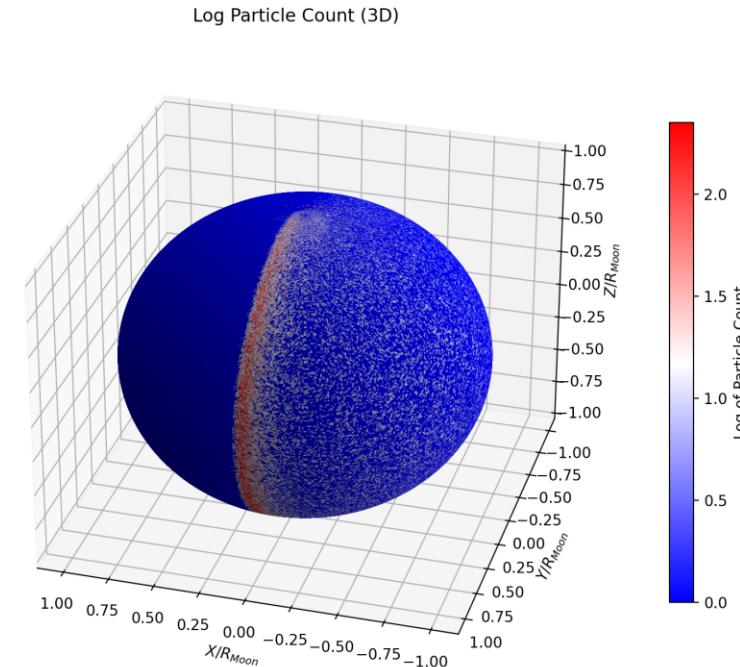
- **Final results***: particle deposition in a longitude-latitude grid
 - Resolution $\rightarrow \Delta\theta = \Delta\phi = 0.5^\circ$



*See note on slide 46 about fix to temperature advancement convention

Example of Long-Distance Lunar Transport

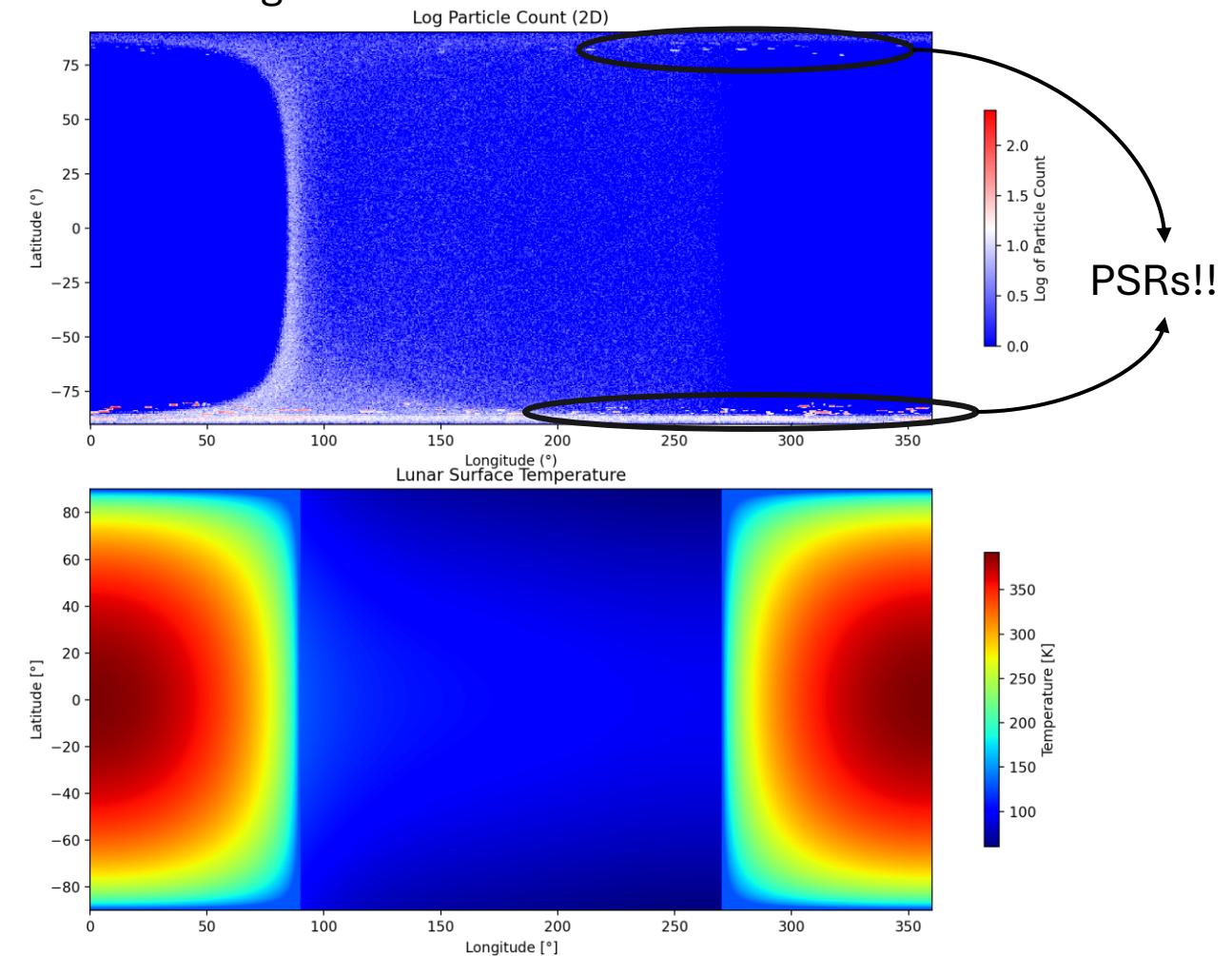
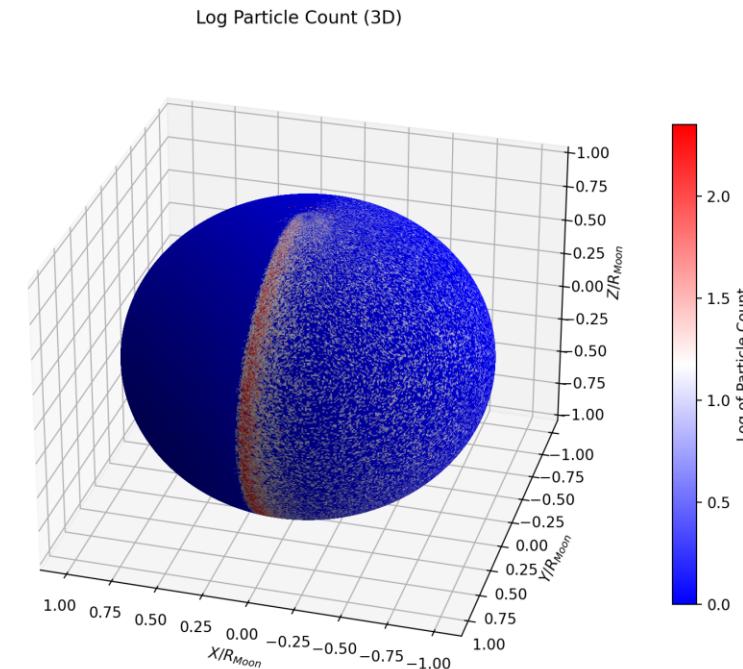
- **Final results***: particle deposition in a longitude-latitude grid
 - Resolution $\rightarrow \Delta\theta = \Delta\phi = 0.5^\circ$



*See note on slide 46 about fix to
temperature advancement convention

Example of Long-Distance Lunar Transport

- **Final results***: particle deposition in a longitude-latitude grid
 - Resolution $\rightarrow \Delta\theta = \Delta\phi = 0.5^\circ$

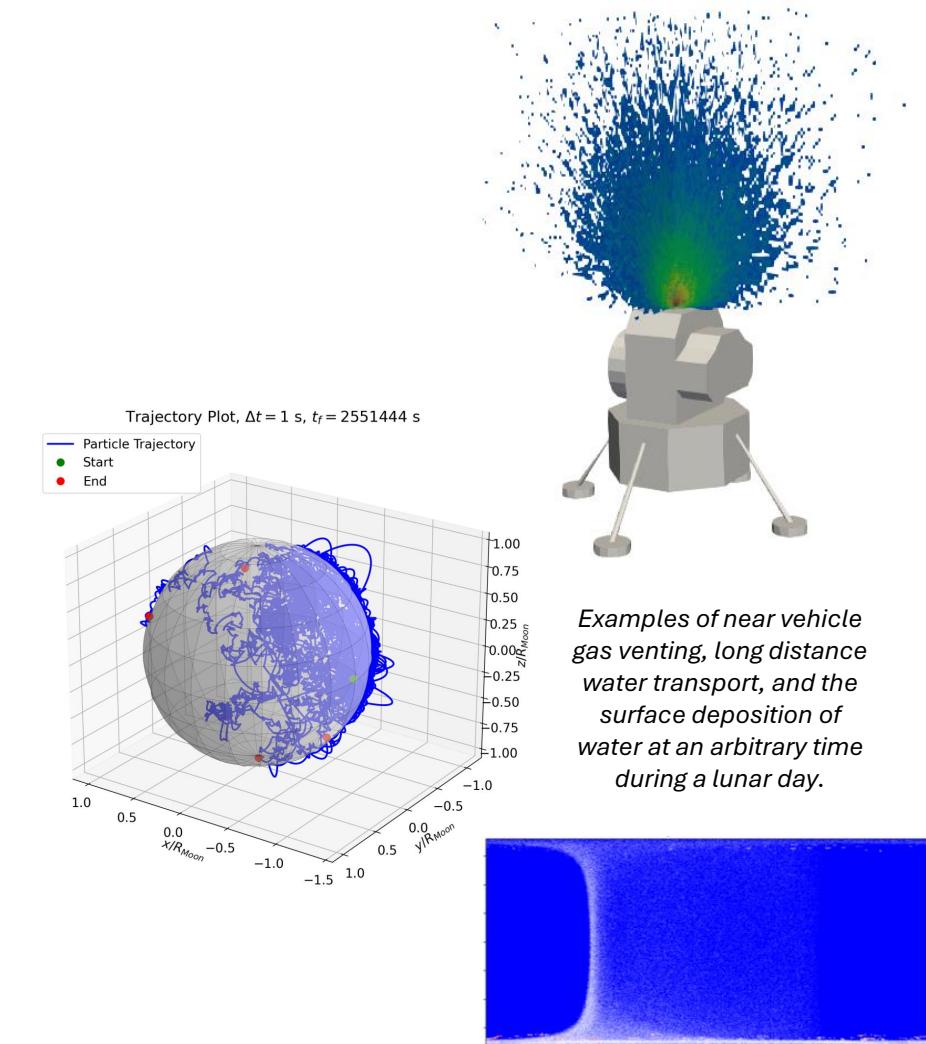


*See note on slide 46 about fix to
temperature advancement convention

Conclusion and Perspectives for Future Work

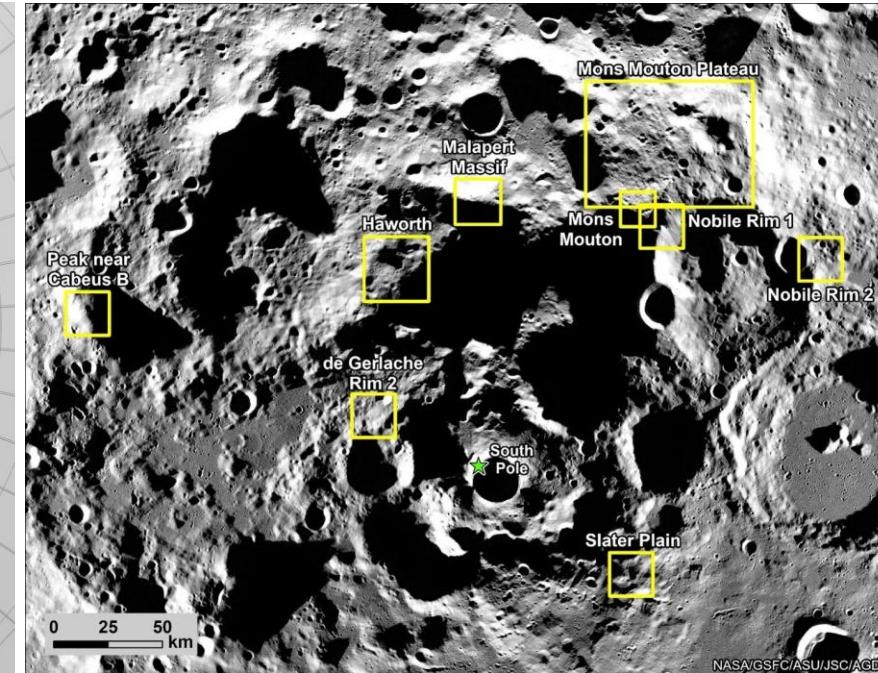
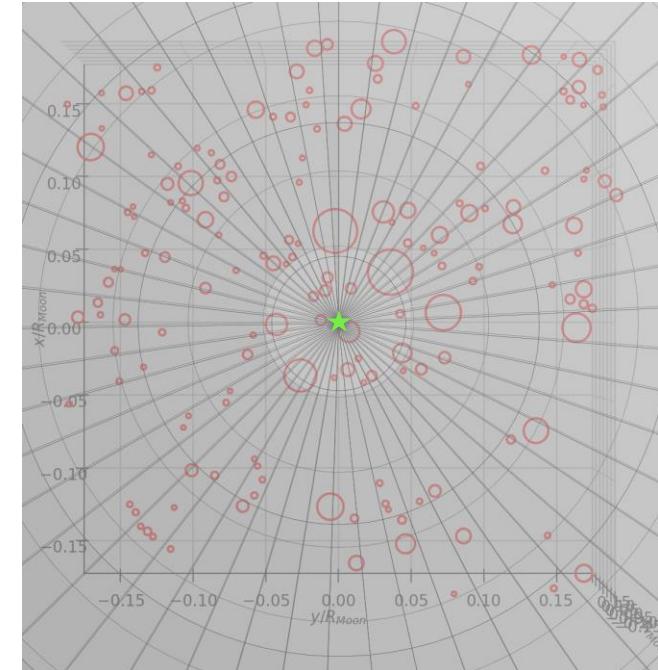
This JVSRP project developed a Python-based free molecular Monte Carlo solver for contamination transport on airless bodies

- There is room for code optimization, which could benefit sampling statistics / reduced computational expense for future simulations.
- Although the presented results have shown randomly-generated molecules released from the lunar surface, a framework for using SPARTA outputs has been built and is ready for future studies.
- Future work can be done to implement other physical processes
 - Photodissociation and ionization
 - Chemical reactions with the ground
 - Improved lunar surface temperature maps
 - (...)



Conclusion and Perspectives for Future Work

- Future work could focus in improving the efficiency of the solver and to consider real scenario inputs including simulated outgassing, venting, and pluming data for a lander
- Example: Artemis III mission → Crewed mission to Moon's south pole



Map of south polar PSRs as modeled in these simulations (left) vs. image of lunar south pole (right).

Thank you for your attention!

This research was carried out at the Jet Propulsion Laboratory, California Institute of Technology, and was sponsored by JVSRP and the National Aeronautics and Space Administration (80NM0018D0004).



Free Molecular Monte Carlo Model for Lunar Contaminant Transport and Deposition in Permanently Shadowed Regions

JVSRP Final Presentation

Raphael Alves Hailer^{1,2*}

JPL Mentors:

Dr. William A. Hoey ^{2**}

Dr. John M. Alred ²

Dr. John R. Anderson ²

¹ Universidade Estadual de Campinas (UNICAMP), Cidade Universitária Zeferino Vaz - Barão Geraldo, Campinas - SP, 13083-970, Brazil

² Jet Propulsion Laboratory, California Institute of Technology, 4800 Oak Grove Dr. M/S 125-109, Pasadena, CA 91109

* < r223852@dac.unicamp.br >

** < william.a.hoey@jpl.nasa.gov >

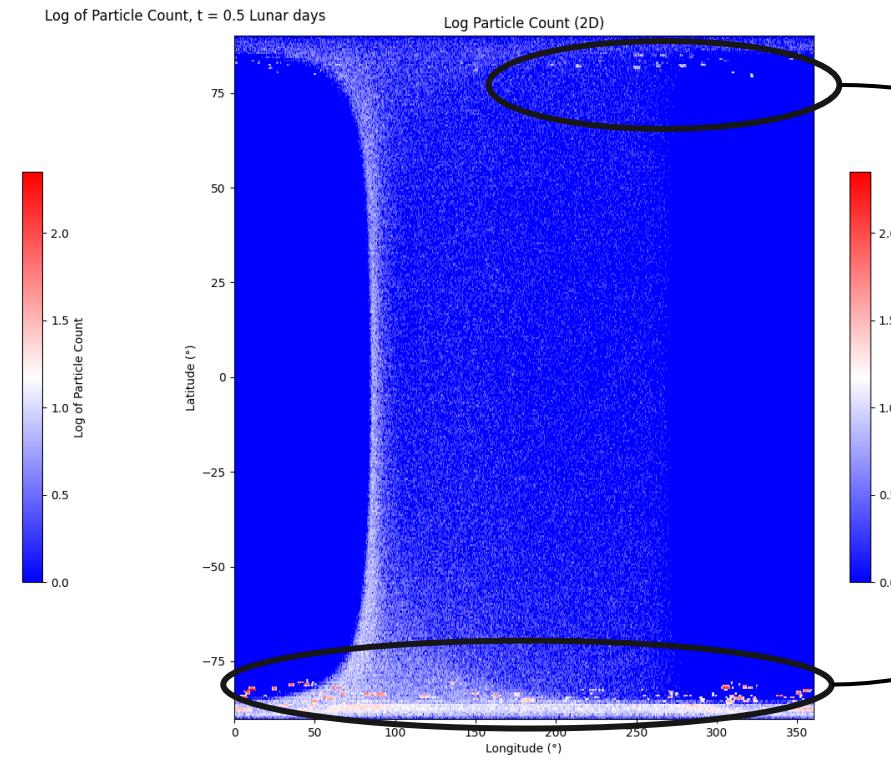
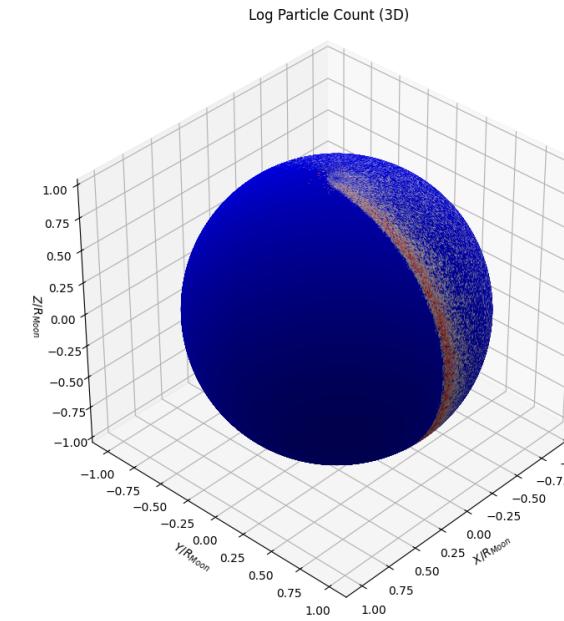


Jet Propulsion Laboratory
California Institute of Technology

This document has been reviewed and determined
not to contain export controlled technical data.
© 2025. All rights reserved.

Testing with inputs from a SPARTA simulation

- Final results: particle deposition in a longitude-latitude grid
 - Resolution $\rightarrow \Delta\theta = \Delta\phi = 0.5^\circ$



PSRs!!

Last moment issue found in the solver

- A blunder that was found in the Python code, which has already been corrected for generation of new results, is that the **convention for advancement of the subsolar point was reversed**
- This does not change the global physics of the solver, nor it will affect the prior results much as in nighttime the H₂O particles are deposited in all regions of the cold side of the Moon
- This error is observed in the last result (slides 39 – 41); these images were not updated for this final presentation but will be in any future publications

