University of Edinburgh, EOS-MLS, Clouds, and ARTS

Dr. Cory Davis

cory@met.ed.ac.uk

University of Edinburgh

- Institute for Meteorology

University of Edinburgh

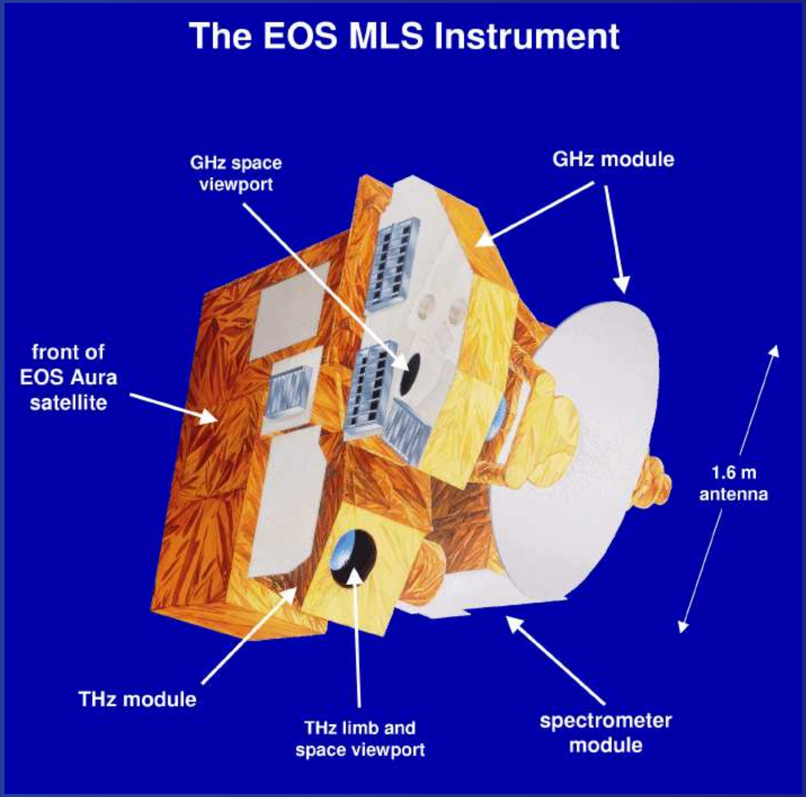
- ~~Institute for Meteorology~~ Institute of Atmospheric and Environmental Science, School of GeoSciences

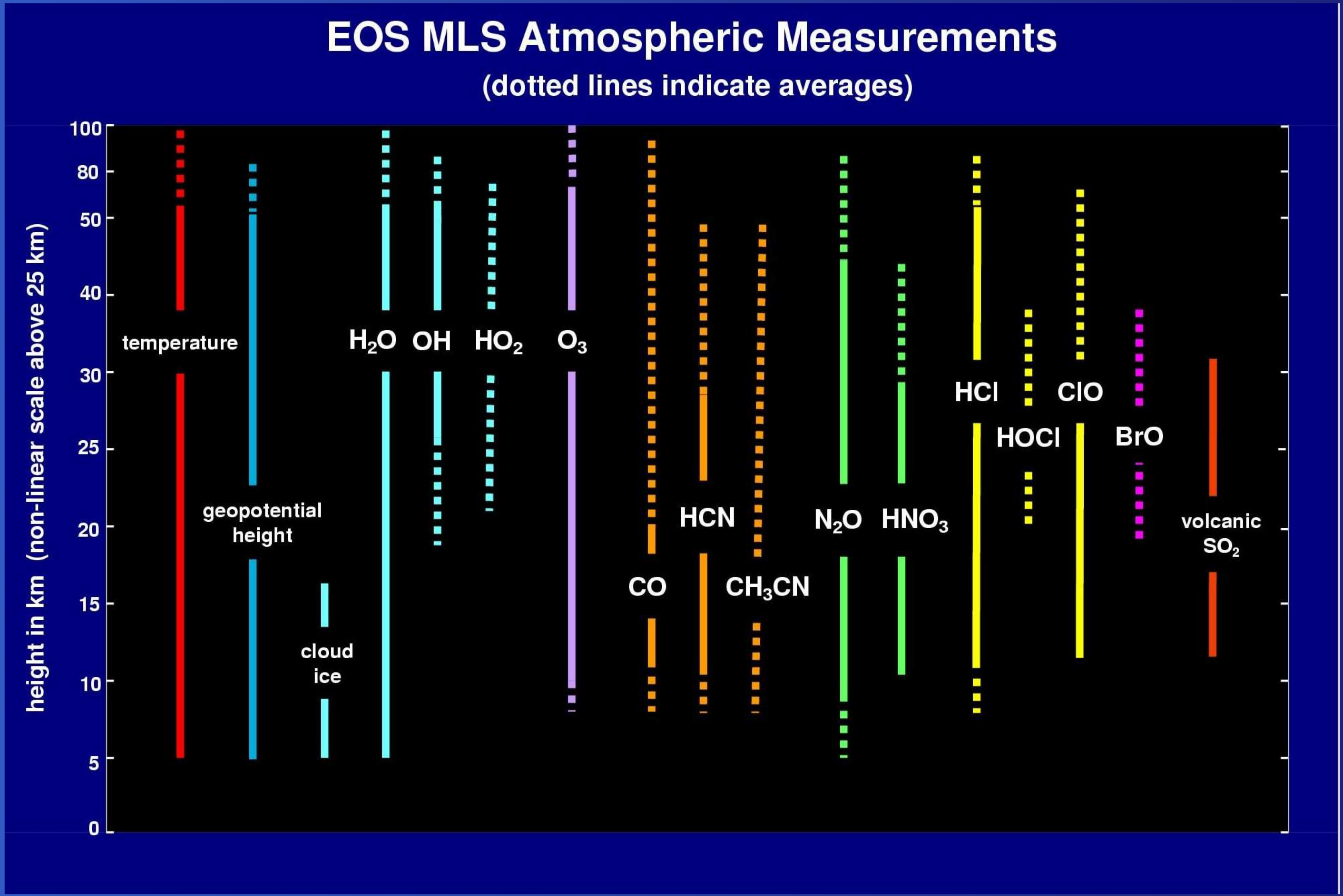
University of Edinburgh

- ~~Institute for Meteorology~~ Institute of Atmospheric and Environmental Science, School of GeoSciences
- Research: Oceanography, Global Dynamics and Chemistry, Remote Sensing, Surface Processes, Pollution

EOS-MLS

- Co-PIs: Joe Waters(JPL), Bob Harwood(EU)
- Sequel to UARS-MLS
- on AURA satellite - due for launch January 2004
- mm and sub-mm wave-length heterodyne radiometers in 5 broad bands – 118 GHz, 190 GHz, 240 GHz, 640 GHz, 2.5 THz




EOS MLS Atmospheric Measurements
(dotted lines indicate averages)
height in km (non-linear scale above 25 km)
100
80
50
40
30
25
20
15
10
5
0
temperature
geopotential height
cloud ice
H_2O
OH
HO_2
O_3
CO
HCN
CH_3CN
N_2O
HNO_3
HCl
HOCl
ClO
BrO
volcanic SO_2

Clouds and EOS-MLS

- Proposed measurements well below the tropopause – particularly H_20 and ozone – can be influenced by cirrus.
- In some cases, where the cirrus is optically thin, IWC can be retrieved. In optically thick cases greenhouse gas retrievals can become impossible.
- These conditions need to be defined. $\longrightarrow$ my project.

Radiative Transfer Model

- Existing EOSMLS RT models (including operational JPL cloudy-sky model) have been 1D and used Mie theory.
- There is a concern that we might be missing important effects due to finite horizontal extent, inhomogeneity, and non-spherical hydrometeors.
- Require 3D polarised radiative transfer model.

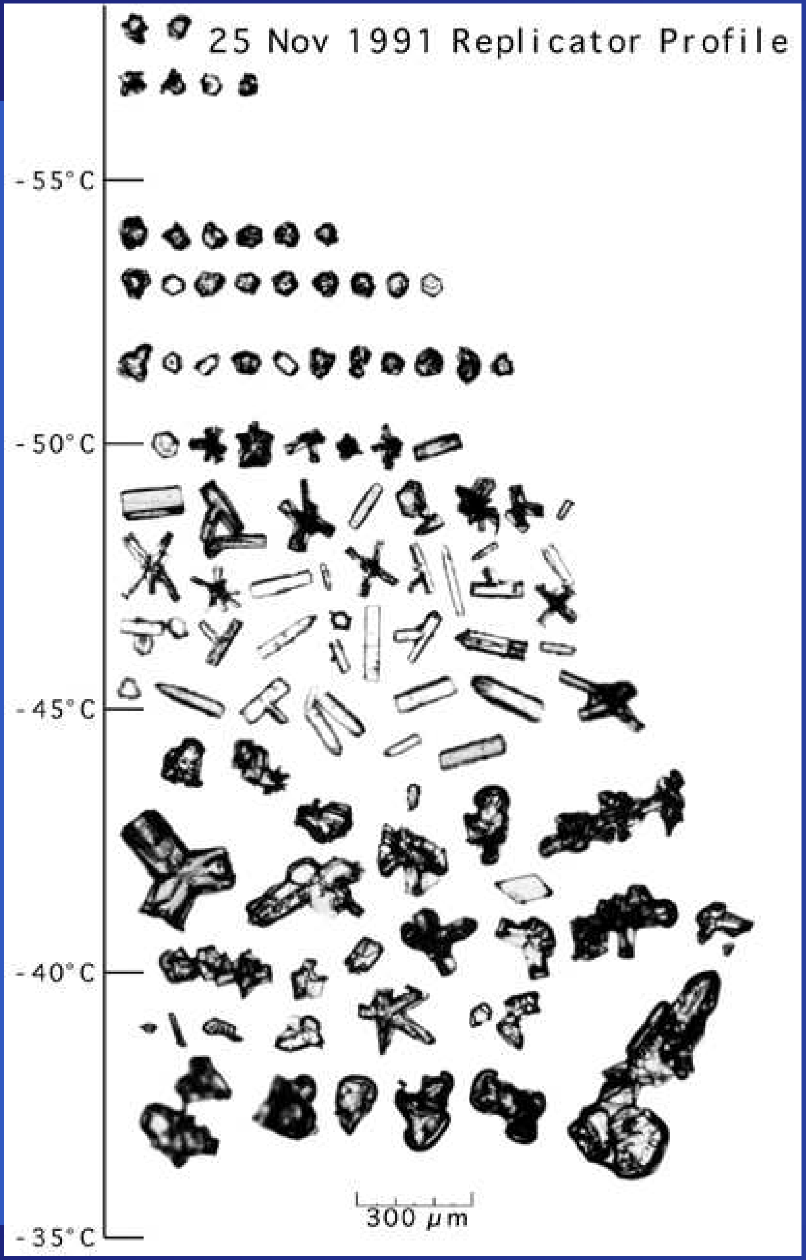
Single Scattering Properties

$$\frac{d\mathbf{I}(\mathbf{n})}{ds} = -n_0\langle\mathbf{K}(\mathbf{n})\rangle\mathbf{I}(\mathbf{n}) + \mathbf{K_a}(\mathbf{n})I_b(T) + n_0\int_{4\pi}\langle\mathbf{Z}(\mathbf{n},\mathbf{n}')\rangle\mathbf{I}(\mathbf{n}')d\mathbf{n}'$$

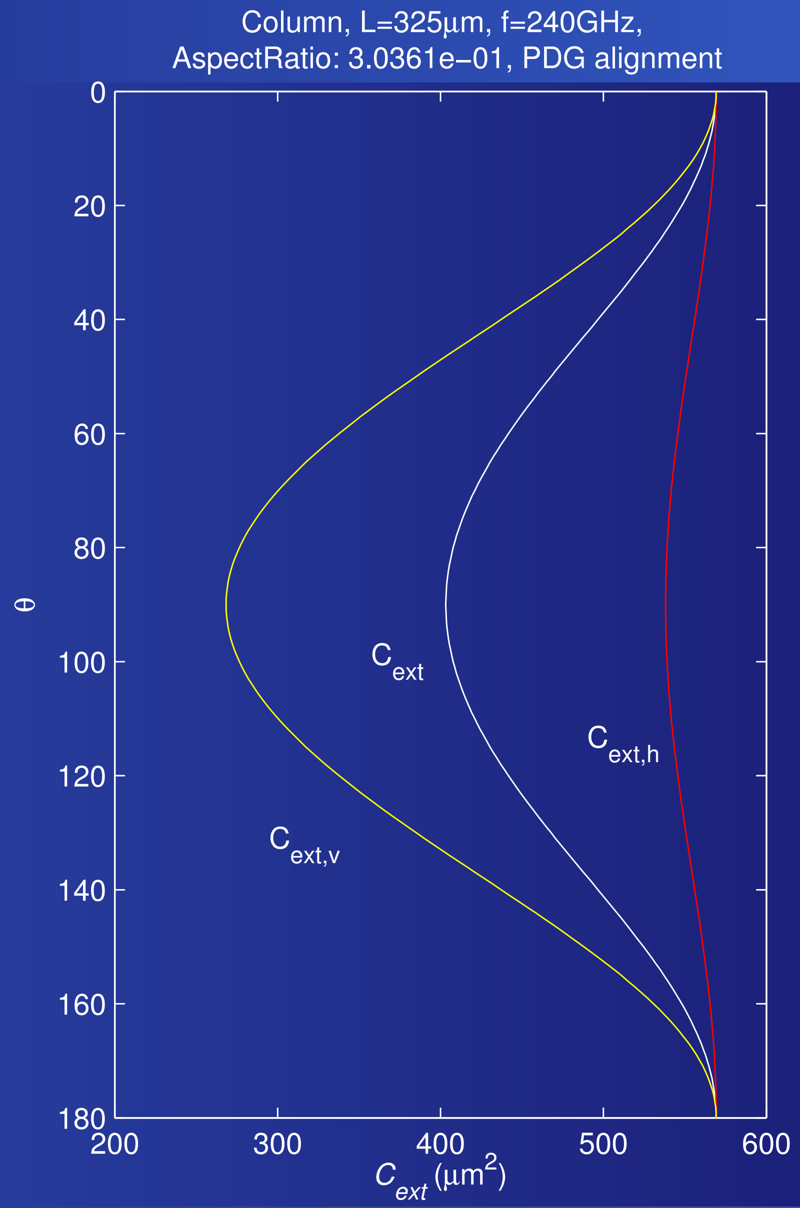
- First we need a way of calculating extinction matrix, absorption coefficient vector, and phase matrix.
- Also useful to know whether or not $\langle\mathbf{K}(\mathbf{n})\rangle$ is non-diagonal in practical cases.
- T-matrix code of Mishchenko. Some limitations.
  - size parameter - aspect ratio limits
  - spheroids and circular cylinders (and chebyshev particles) only
- seems reasonable to approximate column and plate cirrus crystals by circular cylinders.

Single Scattering Properties

- modified and extended Mishchenko's fortran code to allow
  - Analytic orientation averaging of the T-matrix to give $\langle \mathbf{K}(\mathbf{n}) \rangle$.
  - Numerical orientation averaging of $\mathbf{Z}(\mathbf{n}, \mathbf{n}')$.
  - $\mathbf{K}(\mathbf{n})$ and $\mathbf{Z}(\mathbf{n}, \mathbf{n}')$ for several incidence and scattering directions with a single calculation of the T-matrix.
- Top level Python code to generate XML data files for later use in RT calculations, for given
  - particle type and size (dimensional relations from cloud literature)
  - orientation distribution (random or horizontal)
  - frequency, temperature



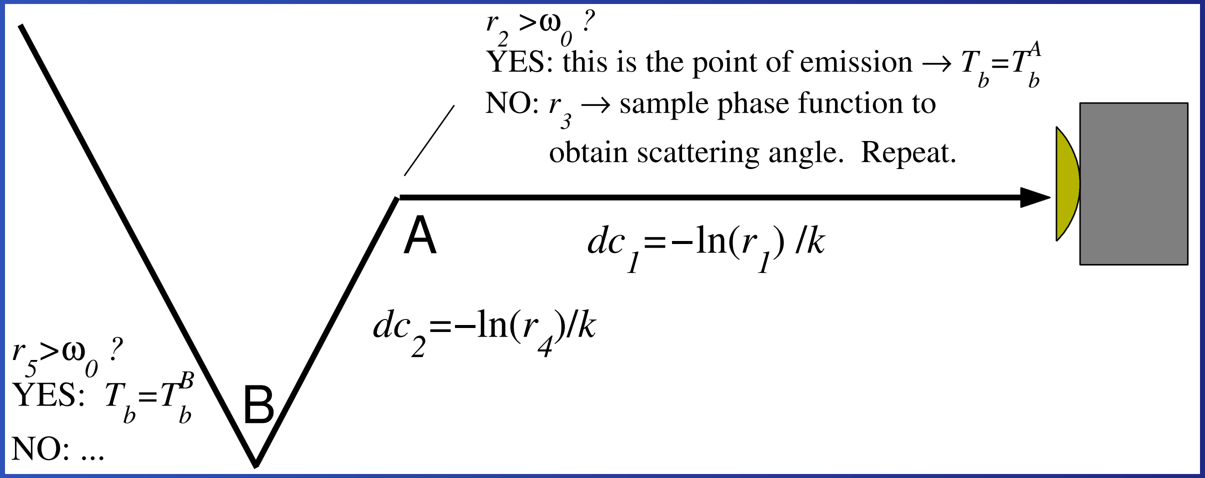
25 Nov 1991 Replicator Profile
-55°C
-50°C
-45°C
-40°C
-35°C
300 μm



Column, L=325µm, f=240GHz,
AspectRatio: 3.0361e−01, PDG alignment
θ
C_ext
C_ext,h
C_ext,v
C_ext (µm²)
0
20
40
60
80
100
120
140
160
180
200
300
400
500
600

Backward Monte Carlo RT

- The traditional (unpolarized) picture



- How do we deal with matrix extinction?

- Start from scratch, focus on the VRTE , and think about Monte Carlo Integration.

$$\int_V f dV \approx V \left(\langle f \rangle \pm \sqrt{\frac{\langle f^2 \rangle - \langle f \rangle^2}{N}} \right)$$

- Put VRTE in integral form

$$\mathbf{I}(\mathbf{n}, \mathbf{s_0}) = \mathbf{T}(\mathbf{u_0}, \mathbf{s_0})\mathbf{I}(\mathbf{n}, \mathbf{u_0}) +$$
$$\int_{u_0}^{s_0} \mathbf{T}(\mathbf{s}', \mathbf{s}) \left(\mathbf{K_a}(\mathbf{n}) I_b(T) + n_0 \int_{4\pi} \langle \mathbf{Z}(\mathbf{n}, \mathbf{n}') \rangle \mathbf{I}(\mathbf{n}') d\mathbf{n}' \right) ds'$$

- apply Monte Carlo integration to 2nd term. Many ways to do this.

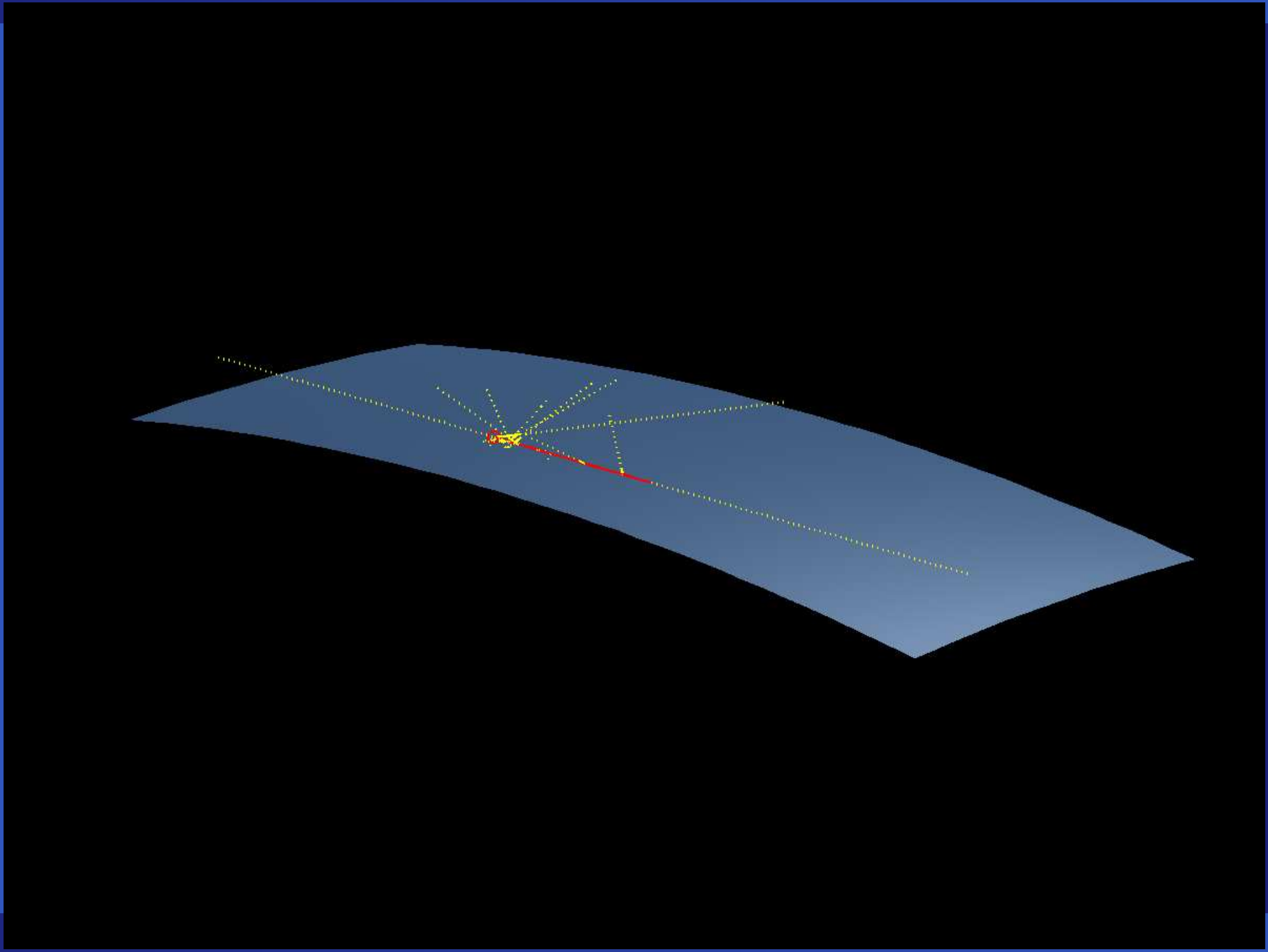
ScatteringMonteCarlo

- Initial Monte Carlo models written in MATLAB, investigated several sampling regimes. Accuracy verified (with severe limitations) by SHDOM (Evans).
- To advance this work I needed a more realistic atmosphere - spherical geometry, spectroscopy, clear-sky RT, framework for representing inhomogeneous cloudy atmosphere. All these already dealt with in ARTS.
- Workspace method ScatteringMonteCarlo. NOT the most efficient algorithm from MATLAB tests, but easiest to program.

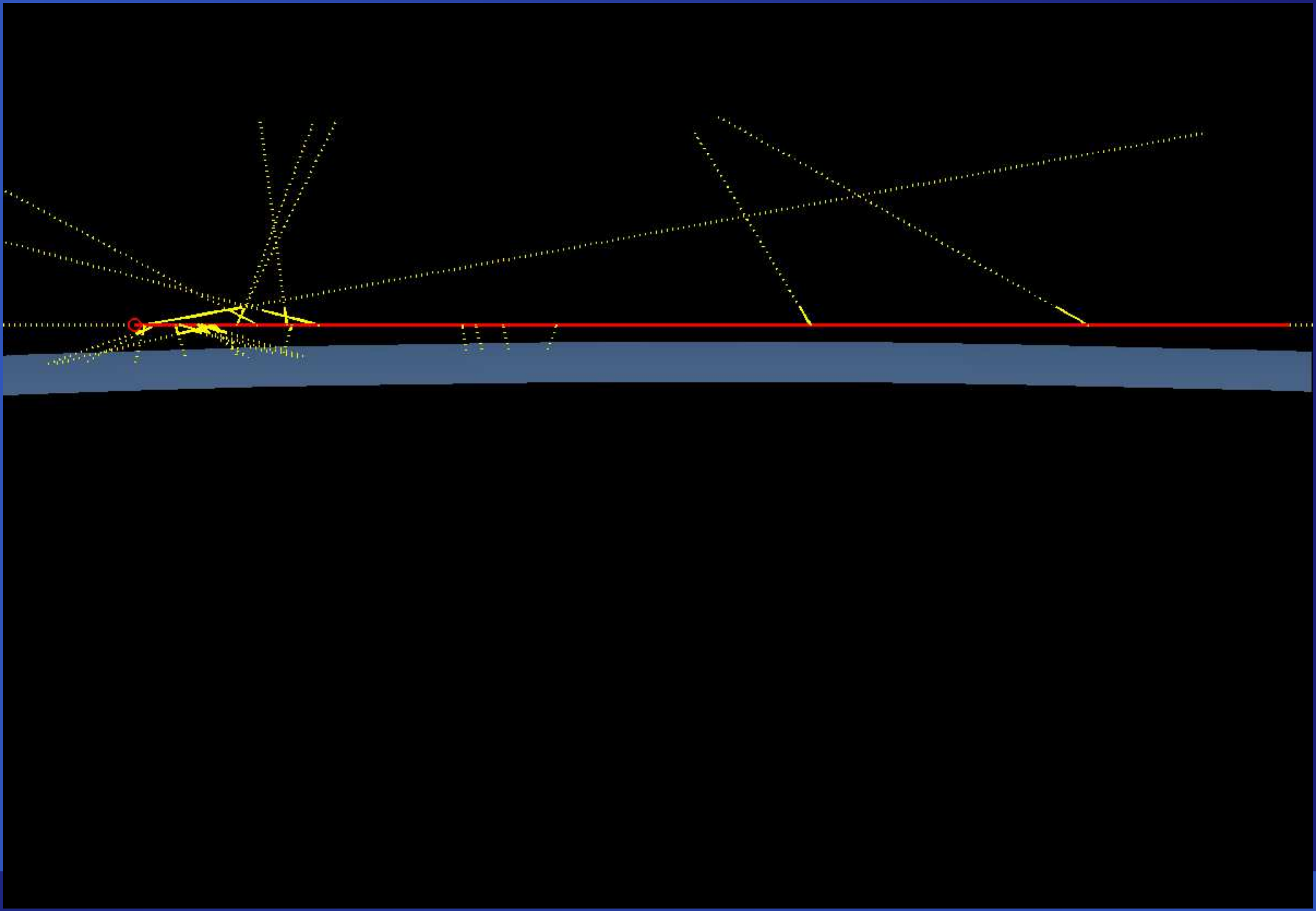
ScatteringMonteCarlo: Performance

- only tested with spheres, `stokes_dim`$= 2$ (to check that $Q = 0$)
- to get an estimated error of 1K need `max_iter`$=10^5$. On one of my processors this takes about 2 hours.
- yet to test accuracy.

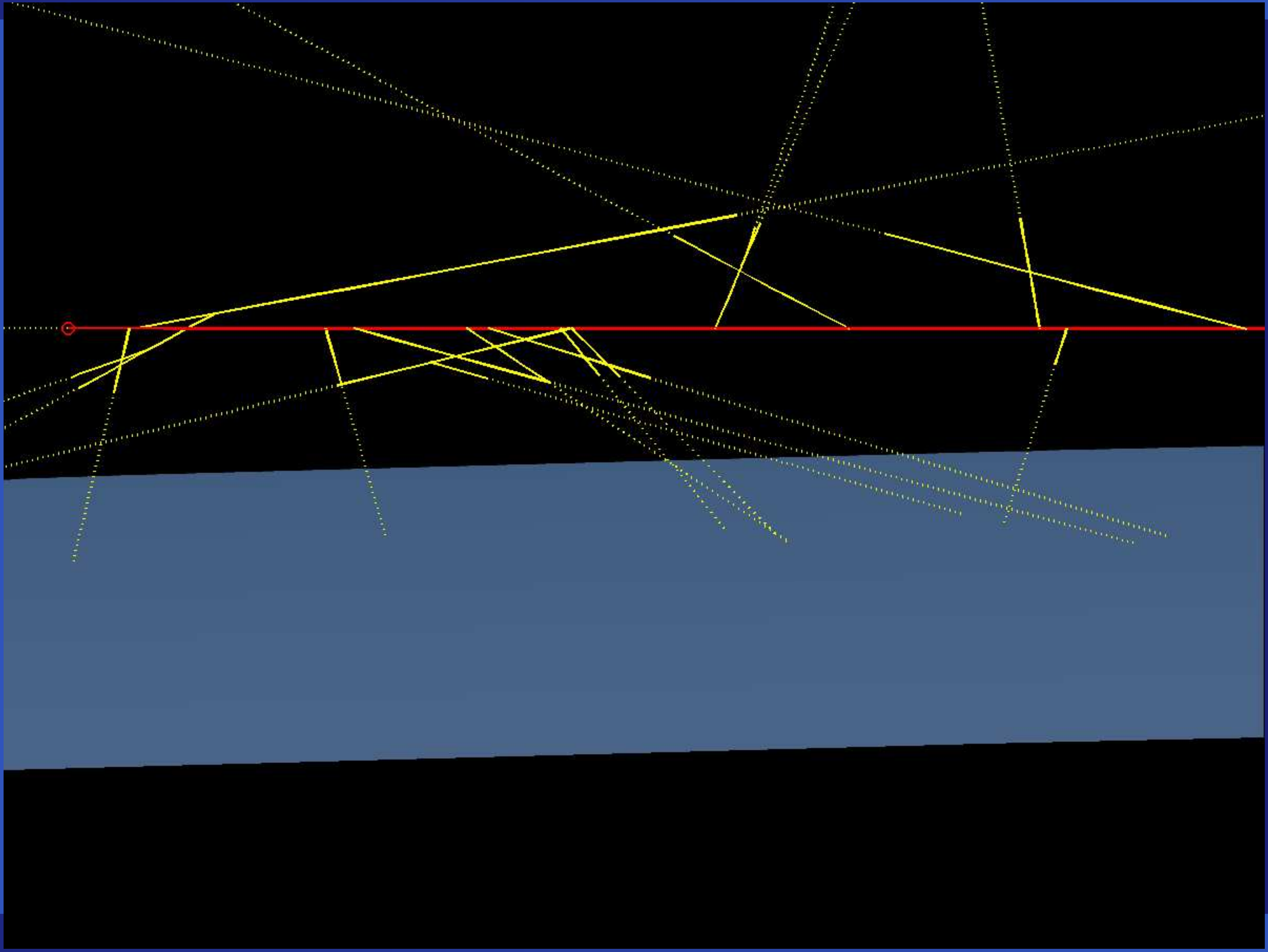
an example



an example



an example



Future plans

- Validate against other ARTS scattering functions
- Address efficiency issues
  - initial look with gprof - looking up single scattering properties by far the most time-consuming
  - more advanced algorithm that accounts for the large variance in incoming radiance at the cloudbox boundary (stratified sampling) combined with sampling incident direction using a PDF $\propto Z_{11} sin\theta_{in}$.
  - parallel computing: MC makes this trivial.
- Use ARTS to investigate possible shortfalls in existing EOS-MLS cloudy-sky model, with the long-term goal of improving cloud and greenhouse gas retrievals in cloudy cases.

The End.