SUOMALAINEN TIEDEAKATEMIA
FINNISH ACADEMY OF SCIENCE AND LETTERS
ACADEMIA SCIENTIARUM FENNICA

UNIVERSITY OF OULU

Development of a Modern Open Source Magnetospheric Computation Tool

NICHOLAS LARSEN (1), ALEXANDAR MISHEV (1,2), ILYA USOSKIN(1,2)

1) SODANKYLÄ GEOPHYSICAL OBSERVATORY, UNIVERSITY OF OULU, FINLAND

2) SPACE PHYSICS AND ASTRONOMY RESEARCH UNIT, UNIVERSITY OF OULU, FINLAND

QUASARE Project

- QUAntification of Spectral and Angular characteristics of extreme Solar eRuptive Events
- Part the analysis for these events involves determining the asymptotic viewing cones and cutoff rigidity at neutron monitor stations.
- These values are complex to compute and are determined via numerical integration.
- A program is needed to preform these calculations.

MAGNETOCOSMICS

Developed by Dr. Laurent Desorgher whilst at the University of Bern, Switzerland

Functions:

- Computes asymptotic direction of incidence
- Computes cutoff rigidity based on inputs of time, direction, and position
- Visualises particle trajectories and magnetic field lines

More info at: http://cosray.unibe.ch/~laurent/magnetocosmics/

MAGNETOCOSMICS

- Issues:
- MAGNETOCOSMICS was last updated in 2006 to be compatible with Geant4 8.1
- Current Geant4 release is 11.0
- Oldest Geant4 release available on the official website is version 9.6
- (https://geant4.web.cern.ch/support/download)

- Very difficult to get MAGNETOCOSMICS to work on new software!
- SPENVIS (https://www.spenvis.oma.be/)

OTSO - Open source geomagneToSphere prOpagation tool

Aim:

- Create an alternate tool to MAGNETOCOSMICS that fulfills the same purpose.
- To provide a solid, functional foundation for a community driven tool.
- All computations can be done under various magnetospheric conditions.

Functions:

- Model the trajectory of a CR in the magnetosphere
- Calculate the cutoff rigidity at any specified location on the globe at any time
- Determine the asymptotic cones of acceptance for any given location
- (Currently doesn't visualise magnetic field lines)

OTSO Languages

PYTHON

Simple language

Used to call OTSO's functions

Used to define input parameters

Easy multicore processing

F2PY

FORTRAN

IRBEM Library

Tsyganenko models (Geopack)

Intensive Computations

Downloading anaconda and a suitable Fortran compiler (gfortran) is all you need to use the tool.

Magnetic Field Models

Internal

- IGRF13
- Tilted Dipole

To be added in the future:
Geomagnetic excursion models

External (Tsyganenko Models)

Fastest

- No External Field
- TSY87S
- TSY87L
- TSY89
- TSY96
- TSY01
- TSY01 - Storm

Slowest

All models can be found from Tsyganenko’s website
https://geo.phys.spbu.ru/~tsyganenko/modeling.html

How OTSO Works

- Numerical integration of equations of motion using the 4th order Runge-Kutta method used.
- Key difference between MAGNETOCOSMICS and OTSO is the integration method.
- Trajectory computed backwards from the Earth's surface to the magnetopause boundary.

$$K_1 = hf(x_n, y_n)$$
$$K_2 = hf(x_n + \frac{h}{2}, y_n + \frac{k_1}{2})$$
$$K_3 = hf(x_n + \frac{h}{2}, y_n + \frac{k_2}{2})$$
$$K_4 = hf(x_n + h, y_n + k_3)$$
$$y_{n+1} = y_n + k_1/6 + k_2/3 + k_3/3 + k_4/6 + O(h^5)$$

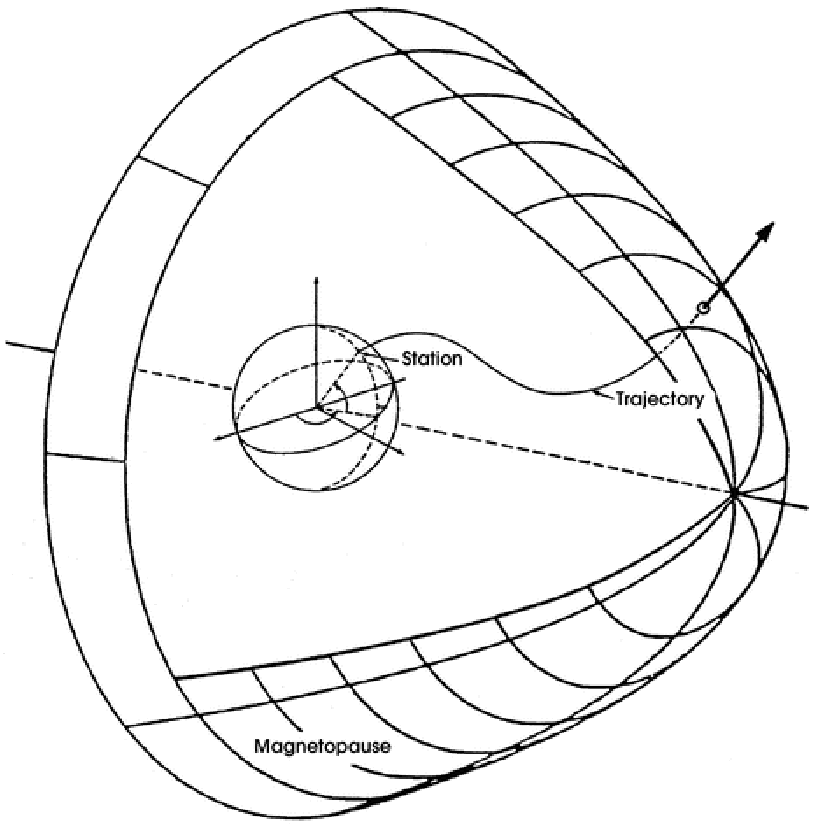


Image: *Bütikofer, R. (2018). Cosmic Ray Particle Transport in the Earth's Magnetosphere*

Example Script – Cone Function

```python
# Picking the stations to be tested.
# Additional stations can be added via the .AddLocation("Name",Latitude,Longitude) function

List = ["Oulu"
"Kiel",
"Moskva"]
Alt = 20.0
Zenith = 0
Azimuth = 0
CreateStations = Stations(List, Alt, Zenith, Azimuth)
#UsedStations.AddLocation("New", 30, 30)

UsedStations = CreateStations.GetStations()

temp = list(UsedStations)
UsedStations = random.shuffle(temp)
UsedStations = temp

###########################################################################################################
# Solar Wind Conditions
Vx = -1003 #[km/s]
Vy = 0.0 #[km/s]
Vz = 0.0 #[km/s]
By = 16.369 #[nT]
Bz = -19.679 #[nT]
Density = 1.9 #[cm^-2]
Dst = -253 #[nT]

# G1 and G2 are only needed for TSY01 use (use external TSY01_Constants tool to calculate them)
# or enter the average solar wind speed, By, and Bz over the 1 hour period prior to event
# G3 Used only in TSY01(Storm)
G1 = 208.309
G2 = 98.69018
G3 = 18.7511


WindCreate = SolarWind(Vx, Vy, Vz, By, Bz, Density, Dst, G1, G2, G3)
WindArray = WindCreate.GetWind()
###########################################################################################################
# IOPT is Picked depending on the Kp index at the time picked
# Take IOPT = kp + 1
# Exception if Kp>=6 IOPT = 7
IOPT = 7
```

```python
###########################################################################################################
# Choose the atomic number of particle you want to test e.g 0 = electron, 1 = proton, 2 = helium
# Choose if you want to reverse the charge 0 = particle, 1 = anti-particle
AtomicNum = 1
AntiCheck = 1
###########################################################################################################
# Enter date to be investigated as a datetime under EventDate
EventDate = datetime(2003, 10, 29, 21, 00, 00)
DateCreate = Date(EventDate)
DateArray = DateCreate.GetDate()
###########################################################################################################
# Pick the magnetosphere models that you want to use.
# Internal: 1 = IGRF, 2 = Dipole (First Number in model array)
# External: 0 = No External Field 1 = TSY87(short), 2 = TSY87(long), 3 = TSY89, 4 = TSY96, 5 = TSY01 , 6 = TSY01 (Planetocosmics), 7 = TSY01(Storm) (Second Number in model Array)
model = np.array([1,3])
###########################################################################################################
# Pick the start and end rigidity for the computation, as well as the step
StartRigidity = 20
EndRigidity = 0
RigidityStep = 0.001
RigidityArray = [StartRigidity, EndRigidity, RigidityStep]
###########################################################################################################
# Choose model magnetopause
# 0 = 25Re Sphere, 1 = Aberrated Formisano, 2 = Sibeck, 3 = Shue et al
Magnetopause = 4
###########################################################################################################
# Choose name of folder that output files will be sent to. Folder created in current directory
FolderName = "GLE66"
FileName = "_TSY89"
current_directory = os.getcwd()
final_directory = os.path.join(current_directory, FolderName)
if not os.path.exists(final_directory):
    os.makedirs(final_directory)

FileDescriptors = [FileName, FolderName, final_directory]
###########################################################################################################
# Select number of cores for multicore processing
CoreNum = 7
UsedCores = Cores(UsedStations, CoreNum)
CoreList = UsedCores.getCoreList()
Positionlists = UsedCores.getPositions()
ChildProcesses = []
###########################################################################################################
```

- NM location data stored in external .csv file
- Unique locations can be checked as well

Example Script – Cone Function

```
OTSOLib.cone(Position, StartRigidity, EndRigidity, RigidityStep, DateArray, model, AtomicNum, AntiCheck, IOPT, WindArray, Magnetopause, FileName, CoordinateSystem)
```

```
Rigidity(GV),Filter,Latitude,Longitude,Xgeo,Ygeo,Zgeo
20.0000,1,40.4249,62.6722,11.7894,20.1165,21.3501
19.9990,1,40.4238,62.6718,11.7891,20.1157,21.3486
19.9980,1,40.4227,62.6713,11.7889,20.1150,21.3471
19.9970,1,40.4216,62.6709,11.7887,20.1143,21.3457
19.9960,1,40.4204,62.6705,11.7885,20.1135,21.3442
19.9950,1,40.4193,62.6701,11.7883,20.1128,21.3427
19.9940,1,40.4182,62.6697,11.7880,20.1120,21.3412
```

```
0.700000E-2,-1,-51.2010,24.0719,0.376843,0.193008,0.907566
0.600000E-2,-1,-51.2498,23.1965,0.381060,0.186118,0.907232
0.500000E-2,-1,38.7509,32.8497,0.292235,0.486600,-0.825619
0.400000E-2,-1,38.7802,32.4941,0.295012,0.483836,-0.826260
0.300000E-2,-1,38.8542,32.5447,0.293872,0.484401,-0.826343
0.200000E-2,-1,38.7804,32.1998,0.297177,0.481256,-0.826942
Ru:0.170000,   Rc:0.170000,   Rl:0.170000
```

- Output designed to be as similar to MAGNETOCOSMICS as possible

Example Script – Cone (README)

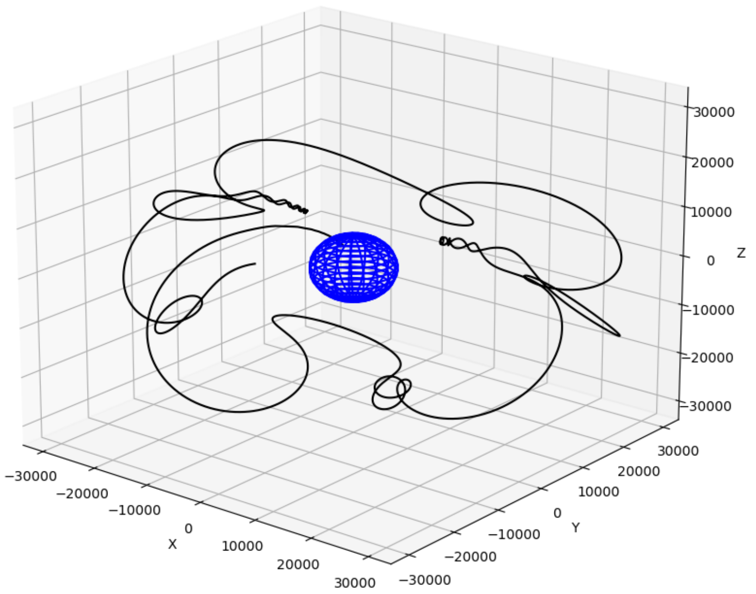
- README file is produced storing the information related to the computation

```
Date of OTSO computation: 2022-06-13
Total computation time: 12541.837 seconds

Input Variables:

Simulation Date: 29/10/2003, 21:00:00

Start Altitude = 20.0km
Zenith = 0
Azimuth = 0

IOPT = 7

Solar Wind Speed [km/s]:
Vx = -1003.0
Vy = 0.0
Vz = 0.0

IMF [nT]:
By = 16.369
Bz = -19.679

Density = 1.9 cm^-2

Dst = -253.0 nT

G1 = 208.3093
G2 = 98.69018
G3 = 0.0

Atomic Number = 1

Particle Type = anti-particle
```

```
Magnetic Field Models:
Internal Model = IGRF
External Model = Tsyganenko 89

Magnetopause Model = Kobel Model

Rigidity
Start = 20 [GV]
End = 0 [GV]
Step = 0.001 [GV]

Stations:
Alma-Ata B, Latitude: 43.14, Longitude: 76.6
Apatity, Latitude: 67.55, Longitude: 33.33
Athens, Latitude: 37.58, Longitude: 23.47
Baksan, Latitude: 43.28, Longitude: 42.69
Barentsburg, Latitude: 78.06, Longitude: 14.22
Bern, Latitude: 46.95, Longitude: 7.45
Calgary, Latitude: 51.08, Longitude: -114.13
Erevan, Latitude: 40.5, Longitude: 44.17
Erevan3, Latitude: 40.5, Longitude: 44.17
FortSmith, Latitude: 60.02, Longitude: 248.07
Hermanus, Latitude: -34.43, Longitude: 19.23
Inuvik, Latitude: 68.35, Longitude: 226.28
```

Trajectory

- Individual trajectories can be computed
- Standard output in geocentric coordinates
- Coordinate system can be changed



GLE71

Start Position: Oulu

Rigidity: 0.50GV

Result: Forbidden

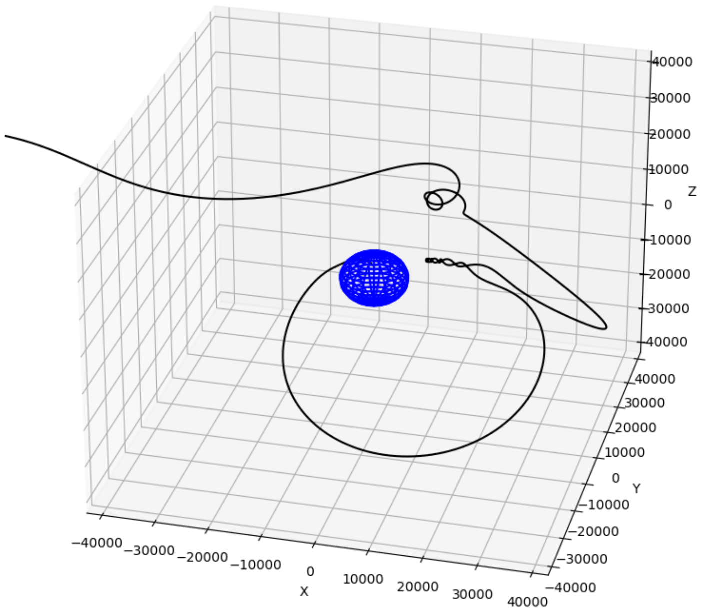
Reason: Exceeded travel distance without escape

GLE71

Start Position: Oulu

Rigidity: 0.60GV

Result: Allowed



Cutoff Comparison – GLE70

UNIVERSITY OF OULU

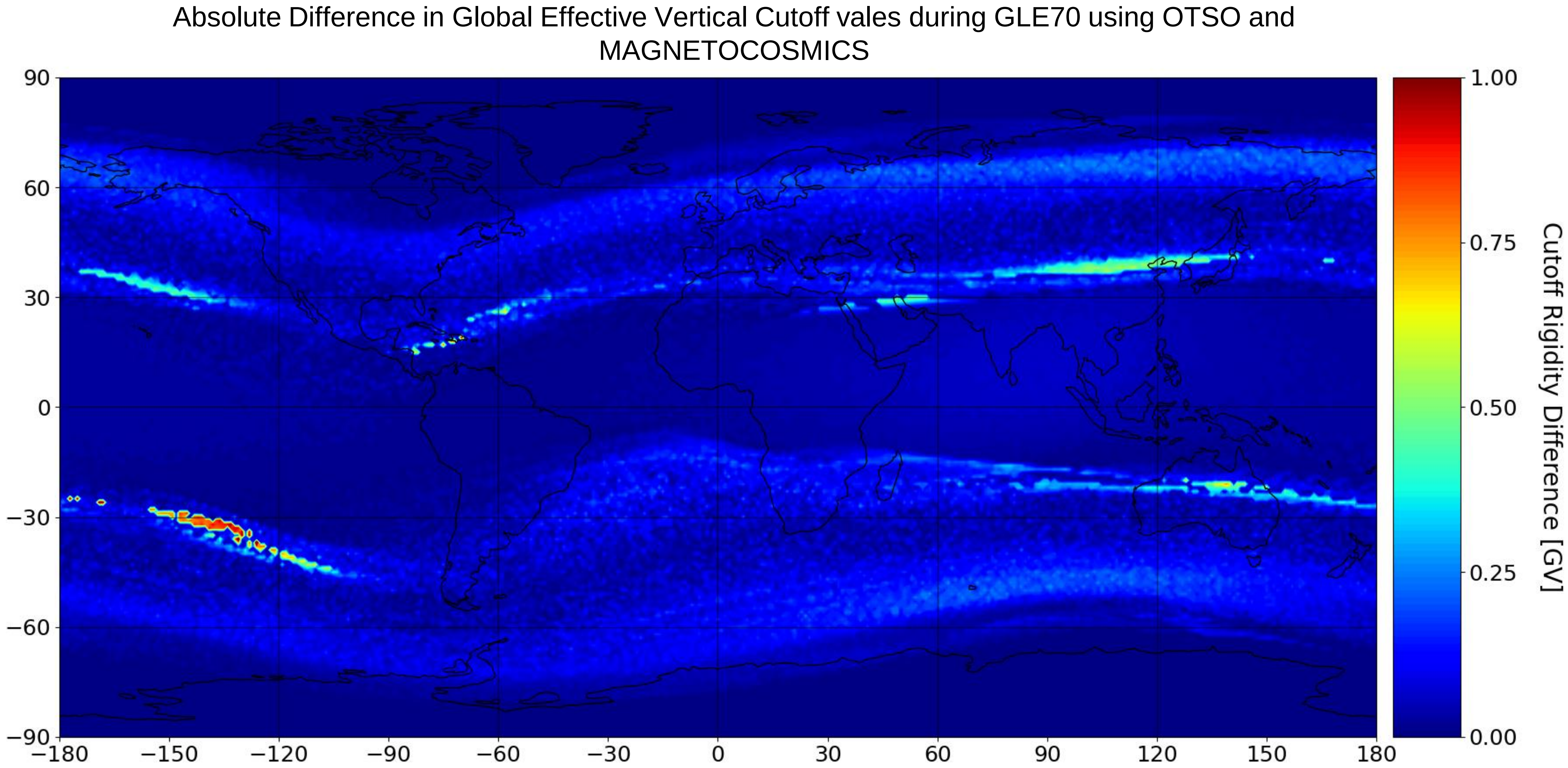
$$R_c = R_U - \int_{R_U}^{R_L} \Delta R(allowed)$$

Station	Vertical Cutoff Rigidity [GV] MAGNETOCOSMICS	OTSO	Percentage Difference [%]
Apatity	0.516	0.527	2.109
Calgary	0.92	0.924	0.434
Cape Schmidt	0.368	0.377	2.416
Fort Smith	0.158	0.167	5.538
Kerguelen	0.933	0.947	1.490
Kingston	1.725	1.738	0.751
LS	3.633	3.644	0.302
McMurdo	0.000	0.000	0.000
Oulu	0.622	0.647	3.940
Rome	6.091	6.089	0.033
TA	0.000	0.000	0.000
Tixie Bay	0.416	0.441	5.834

Cutoff Comparison – GLE66

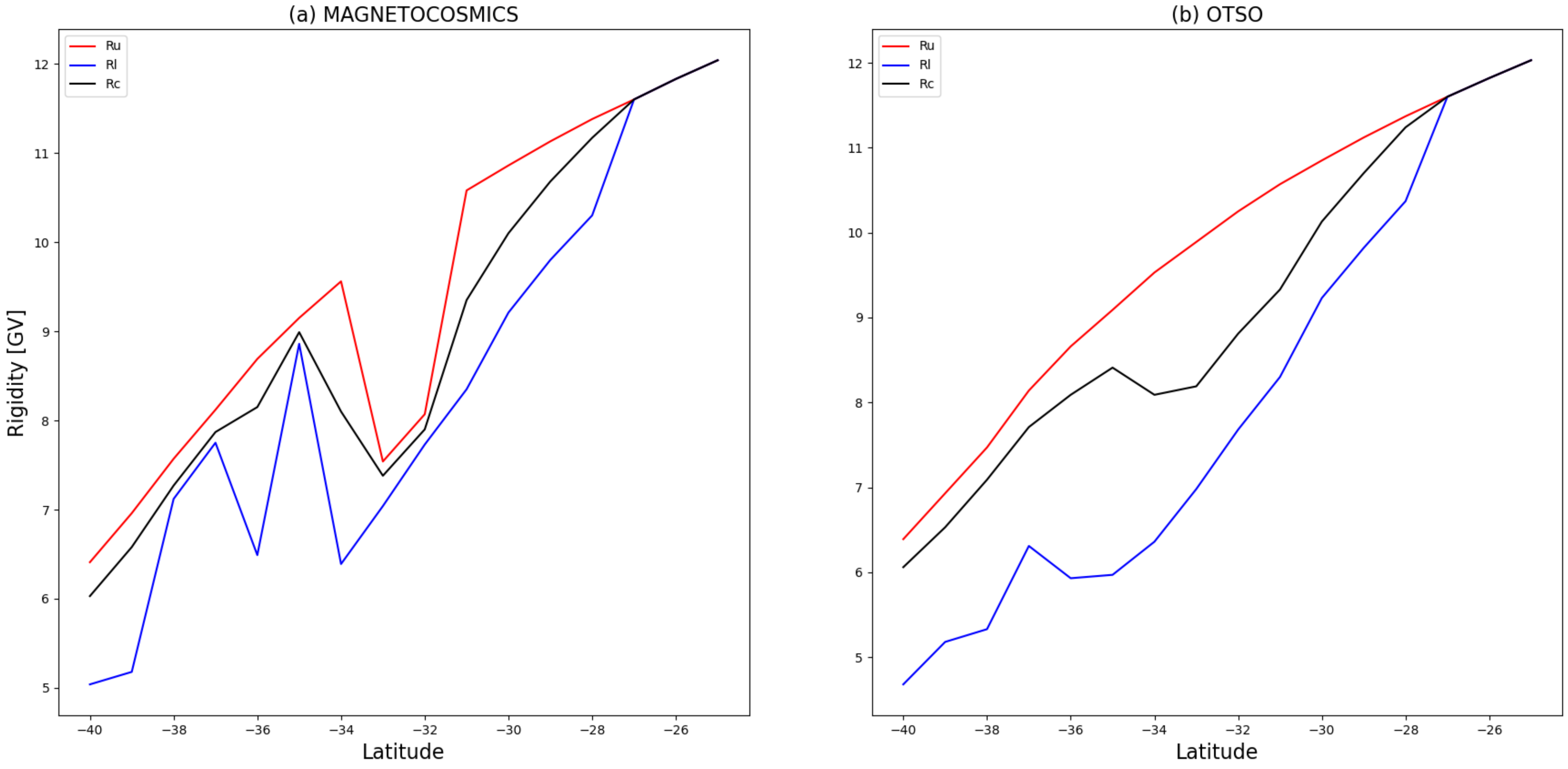
	OTSO Rc [GV]		MAGNETOCOSMICS Rc [GV]	
Station	**TSY96**	**TSY01**	**TSY96**	**TSY01**
Apatity	0.113	0.517	0.1	0.512
Athens	7.813	7.95	7.812	7.968
Baksan	4.866	5.069	4.87	5.032
Fort Smith	0	0	0	0
Inuvik	0	0	0	0
Jungfraujoch	3.582	3.773	3.562	3.782
Kerguelen	0.204	0.843	0.176	0.831
Kiel	0.788	1.167	0.769	1.179
Kingston	0.509	0.361	0.769	0.353
LomnickiStit	2.608	2.812	2.626	2.81

Global Vertical Cutoffs

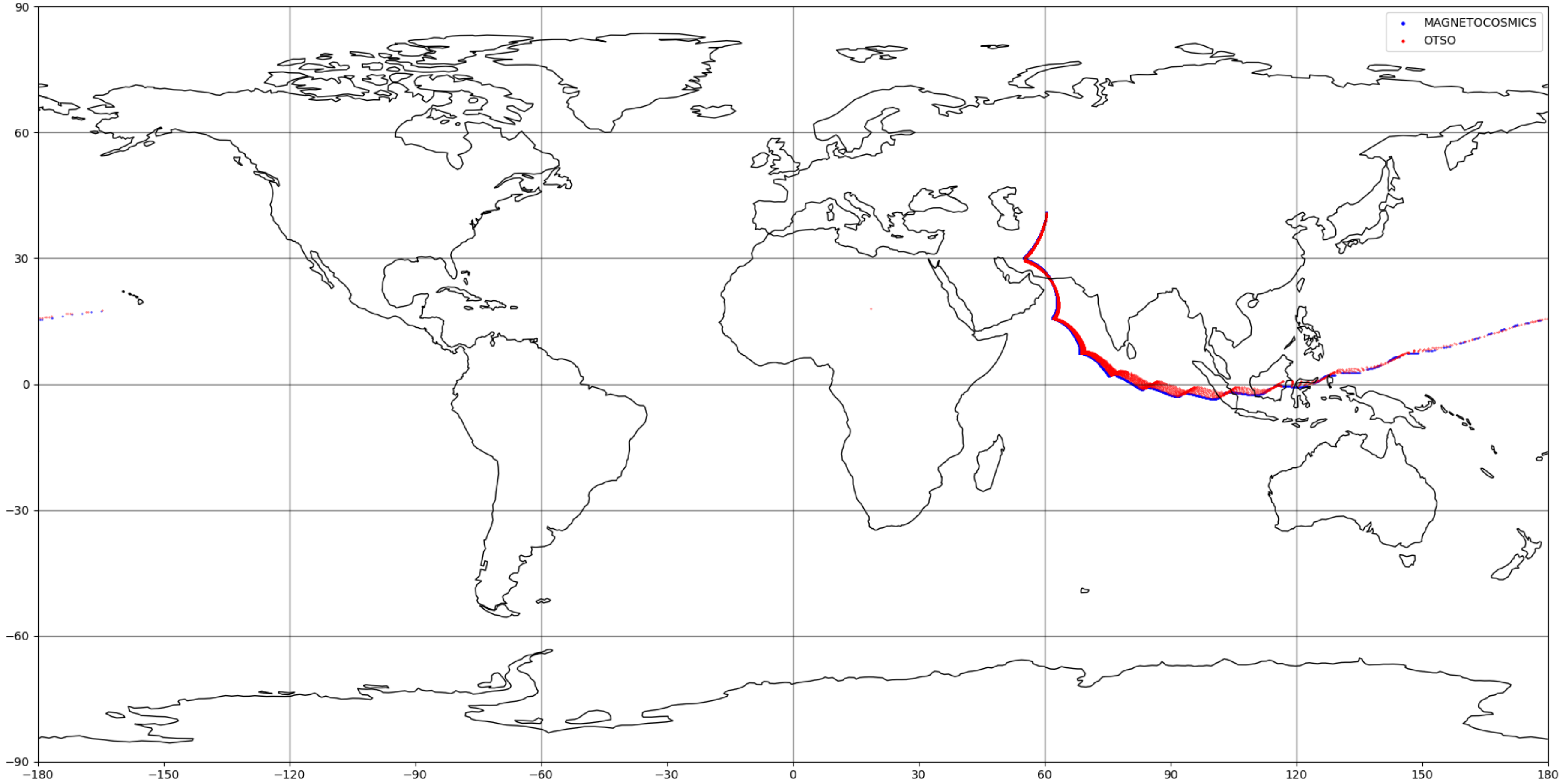


Global Vertical Cutoffs Anomaly

Side by side comparison of the South Pacific anomaly shown previously.
Data taken from longitude of -140°



Asymptotic Cone GLE70 - Oulu



Summary

- OTSO shows promise for being a useful tool that can be developed by the community
- Very simple and easy to get working on any computer
- Easy to edit and customise
- Differences in results between OTSO and MAGNETOCOSMICS is a result of differing integration methods

Thank you for Listening

Otso = “King of the Forest” = 

Contact:
Email: nicholas.larsen@oulu.fi
Phone: +358414973008