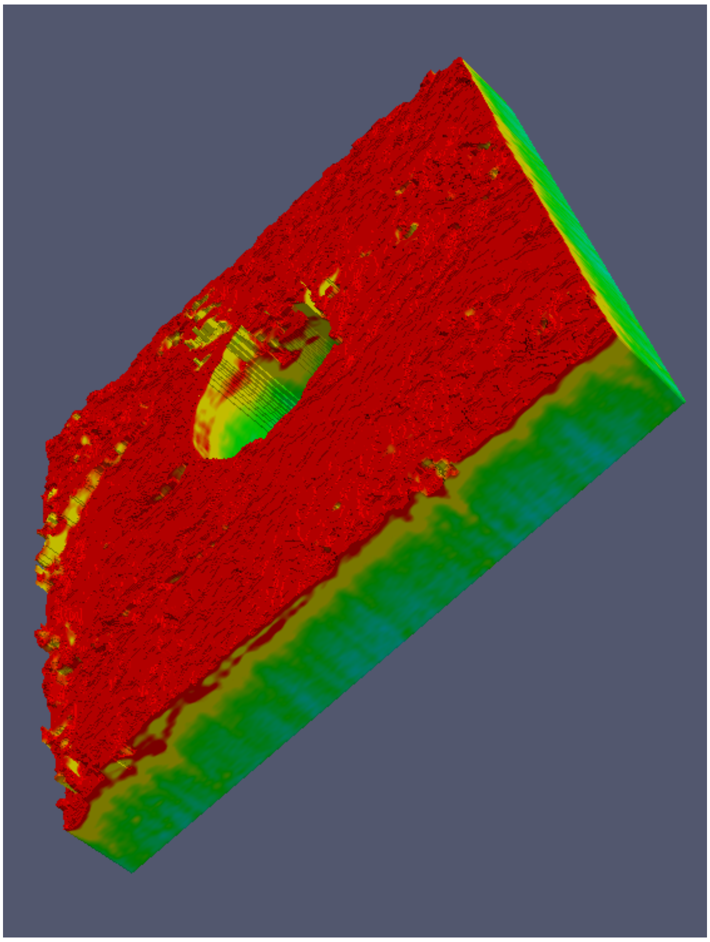
Exploring Mars Surface Using 3D Visualisation of Post Processed MARSIS Data

Visualising Mars Pole



Leon Kocjančič

MARSIS is a radar carried by spacecraft Mars Express, and probing the interior of Mars to look for ice and water. Using radargrams that represent electromagnetic wave scattering, the 3D surface and subsurface layers of Mars south pole were visualised.

The reconstruction of the Mars south pole surface and subsurface layers using MARSIS data was divided in multiple consecutive steps. All of them are described below in the correct order. The most computationally demanding was interpolation of the missing datapoints in predefined matrix.

Where the data comes from

Mars Express is a spacecraft produced by the European Space Agency. Its journey to the Red Planet began on 2 June 2003 with the lunch on a Soyuz-Fregat Rocket. The satellite consists of many components for essential working and seven instruments to study all aspects of Mars.

One of the instruments is called MARSIS which is a low frequency, nadir-looking pulse limited radar sounder and altimeter. It operates at the altitudes of up to 800 km above the Martian surface for subsurface sounding and up to 1200 km for ionospheric sounding. The instrument is working in 1 MHz wide frequency bands centred at 1.8, 3.0, 4.0 and 5.0 MHz. It is transmitting a linear frequency modulated chirp signal with its dipole antenna. The returned signal is received on both the dipole and secondary monopole antenna. The secondary monopole antenna is oriented towards surface and it is used to reduce signal clutter from the ground.

Both signals from monopole and dipole antenna are converted to range offset video signals before they are being processed by analogue to digital converter. Then the data is formatted by the on-board digital processor and passed to telecommunication module for transmission to Earth. The receiver electronics has local oscillator, which generates chirp signal, and dual channel receiver that down converts the received echoes. Each receiver channel has a bandpass filter, a mixer, an amplifier chain, low-pass filtering and an analogue to digital converter. The receiver and digital electronics are housed together within a spacecraft. The transmitter electronics is housed in a separate box within satellite.

The main transmit and receive antenna is a deployable dipole which consists of two 20 m elements. The extra gain towards the Martian surface is achieved through on-board processing.

Second monopole antenna is 7 m long an it is arranged in that way, that its gain null is in direction of Martial surface. Both antennas are of novel design and consist of folding composite tube with two wires for signal transmission. The antennas were deployed by pyrotechnic release mechanism.

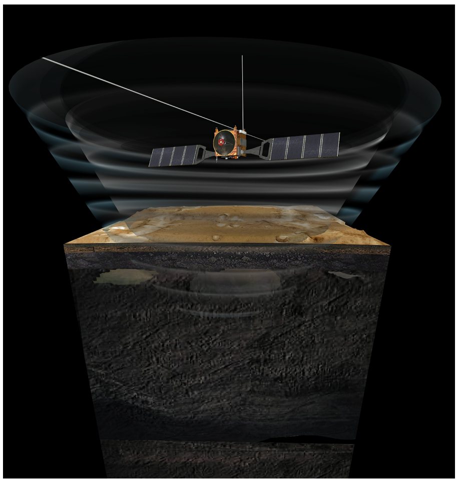


Figure 1: Image of Mars Express probing the Martian terrain.

MARSIS performed subsurface sounding in the highly eccentric orbit selected for Mars Express, which corresponds to about 26 minutes of operation. This is equal to about 100 degrees of arc on the Martian surface for each satellite pass. During the mission extensive coverage at all latitudes was possible, because MARSIS supports dayside and nightside operations. Performance of the ground penetrating radar is best during the night, that is when solar zenith angle is above 80 degrees. In this case ionosphere plasma frequency drops significantly and lower frequency modes of operation can be used, which can penetrate deeper in the ground. If lower frequencies are obstructed by ionosphere, higher frequency mode of operation are used and on-board computer is constantly switching between them.

The primary objective of MARSIS is to map the distribution of water, both liquid and solid, in the upper portions of the crust of Mars. The discoveries will be very important for hydrological, geologic, climatic and possible biologic research. The experiment has also three secondary objectives, which are subsurface geologic probing, surface characterisations, and ionosphere sounding.

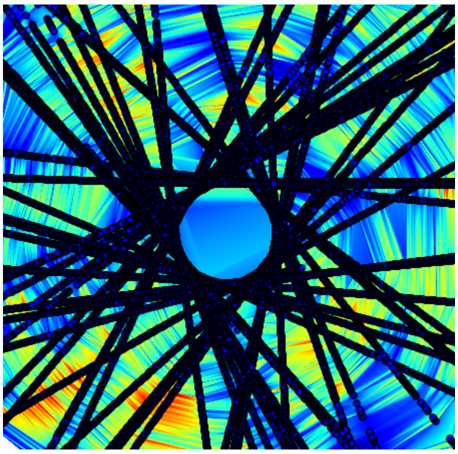


Figure 2: First 50 satellite tracks that were used for surface reconstruction.

Reading satellite data

Our first step incorporated reading the data that radar module on Mars Express generated during its operation. Each satellite orbit around Mars is being tracked and overall 38 parameters describe the various states of MARSIS module. All the parameters are stored in well organised binary file. Our collection of data consisted of around 3000 satellite tracks that were passing over Mars south pole.

The parameters that we were interested in were recordings of reflected radar pulses as well as longitude and latitude at which recordings were done. Radar is constantly scanning at two different frequencies and with three different filtered views at certain location. For our visualisation we have chosen first frequency, which is also the highest, in order to minimise distortion effects of ionosphere to the propagating signal. Additionally, we have used only the middle filtered view that contained visually the best samples of data.

During the importing routine of the data to our algorithm, we first checked if the gathered data lies south of the latitude of -75°. For all the coordinates that were in our interested field of view we applied the coordinate system transformation from spherical to Cartesian coordinate system. In this way we ensured simpler representation of all the data in the matrix. Additionally, it turned out that also interpolation was easier to implement in comparison to the case with polar coordinate system. At the end the gathered data covered a square surface of 1700 km by 1700 km over the Mars south pole. Data was actually structured from radar echoes along satellite path, that were combined to radargram. However, the region closest to the pole is not covered since satellite orbits do not pass this area.

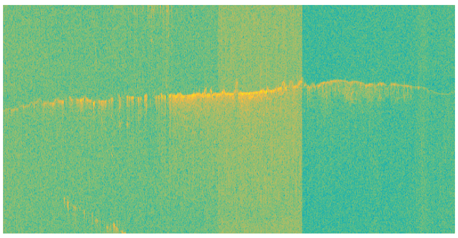


Figure 3: Example of a radargram that combines echoes along satellite path.

Right at the beginning we had to overcome first major technical difficulty. We predicted that our sequential code for data reading would run for more than 30 hours, which was more than our session allowed. At this stage we decided to parallelise the code to use all the available processing power at one computing node, which consisted of 20 processing cores. Since all our code was implemented in Python we decided to use multiprocessing library to parallelise reading routine. In this case we defined a pool of workers and each of them received only a part of the data to process. The order of calculation was not important that is why no synchronisation between processors was needed. The parallelised reading routine finished 10 times faster than sequential one.

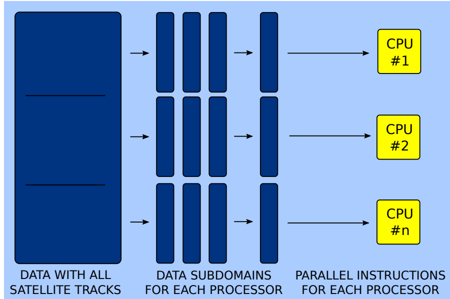


Figure 4: Schematics showing initial data decomposition and parallel instructions for multiple processors.

Interpolating missing data

Once all the data was imported we had to take care of all the missing values that were left in the matrix. The appropriate interpolation method was needed at this point. For testing purposes we used only 50 satellite tracks out of 3000 and immediately found out that the Scipy's barycentric interpolation method does not fulfil our needs.

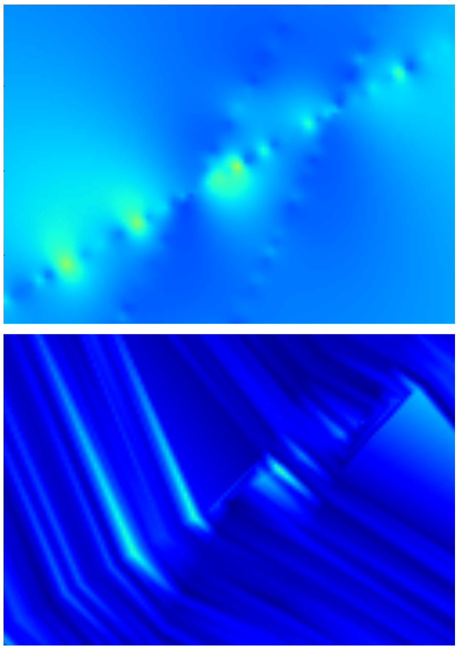
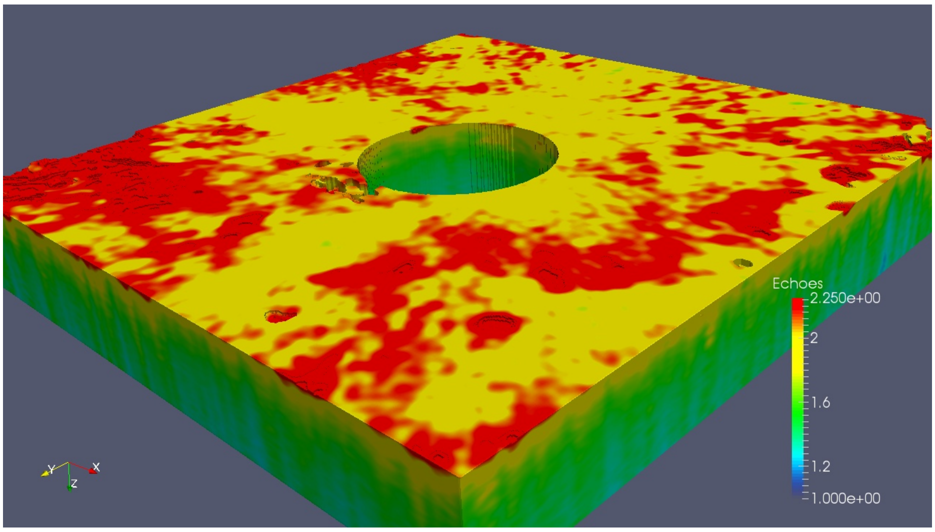


Figure 5: Comparison between barycentric interpolation method on top and RBF interpolation below.

One can see in the figure above that this interpolation method does not produce a smooth surface, which is expected to be seen after the process.



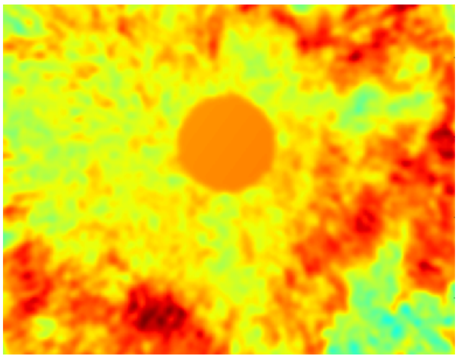
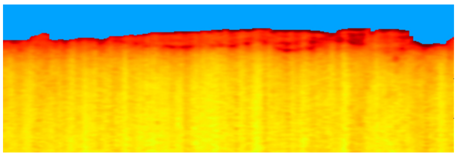



Figure on the left is xy-plane cut of the 3D model, in upper right corner is the result of 3D Gaussian filtering and below is the xz-plane cut where subsurface layers can be seen.

Since the final results were not as expected we had to find better interpolation method. Fortunately, there exists more suitable method which is frequently used in topographical applications and is called radial basis function interpolation or RBF interpolation.

Using this method the interpolated result is actually the sum of the radial basis functions that are real-valued and depend only on the distance from the origin point. With the choice of appropriate function we can affect the smoothness of the end result. That is why the Gaussian function was used. During implementation of interpolating algorithm we found that the Numpy's RBF interpolation function was too slow so we have decided to use the algorithm in Alglib library.

The results were far better since this interpolation technique was also acting like a smoothing filter. With fine tuning of interpolation parameters that affect interpolation radius of each function we were able to achieve good results. For more time efficient calculation we used the same parallelisation technique as in case of data importing. This time the data was divided among z-planes and each process was interpolating its own plane. The interpolation lasted for 3 hours and the final result was 1.5 GB big matrix with surface data.

Data filtering

At this stage the interpolated data was not in the final form because the original data, gathered by the satellite, was quite noisy. It is quite an art to apply numerous digital filtering and still obtain nice realistic pictures of the recorded terrain. First filter that we implemented was cancelling negative values, which resulted at some specific points during the interpolation. When all values that represented reflected power in data were positive we applied a filter for discarding data points with values below the value of average noise level. The filtering was done in the frequency domain using fast Fourier transform.

Our data points obtained from the satellite tracks were not evenly distributed and not perfectly matched along z-axis. This is why we had to introduce further filtering in order to smooth the image and better visualise the characteristics of the terrain. We constructed three dimensional Gaussian filtering algorithm and applied it to the whole three dimensional data set. The filter had the range of 12 pixels in x and y direction and 5 pixels in z dimension. In this way we were able to produce good surface smoothing without loosing the details of the subsurface layers in z-direction.

Final step

With all the data processed and filtered we could start to import it into ParaView which is used for scientific visualisation. In this environment we were able to use some additional filters for cropping our data and removing central part of the region, which contained no real data. Then colouring of all the echo values contained in a matrix was done. We tried to select such colour scale that exposed also subsurface layers that were responsible for electromagnetic wave reflections. Finally fly-by movies and live cross sections were produced using ParaView's animation tools.

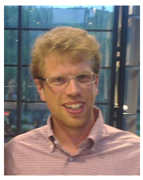
Conclusion

At the end all algorithms were successfully tuned in order to produce expected results. The library with used functions was created and will be used for further investigations of data produced by MARSIS module on Mars Express satellite.

PRACE SoHPCProject Title
Exploring Mars Surface – 3D Visualization of post processed MARSIS data

PRACE SoHPCSite
CINECA, Italy

PRACE SoHPCAuthors
Leon Kocjančič, University of Ljubljana, Slovenia


Leon Kocjančič photo

PRACE SoHPCMentors
Massimiliano Guarrasi, CINECA, Italy
Roberto Orosei, INAF, Italy

PRACE SoHPCContact
Massimiliano Guarrasi, CINECA
Phone: +39 051 6171 560
E-mail: m.guarrasi@cineca.it

PRACE SoHPCSoftware applied
Python, ParaView

PRACE SoHPCMore Information
sci.esa.int/mars-express

PRACE SoHPCAcknowledgement
Author would like to thank both mentors and all supporting staff at CINECA.

PRACE SoHPCProject ID
1506