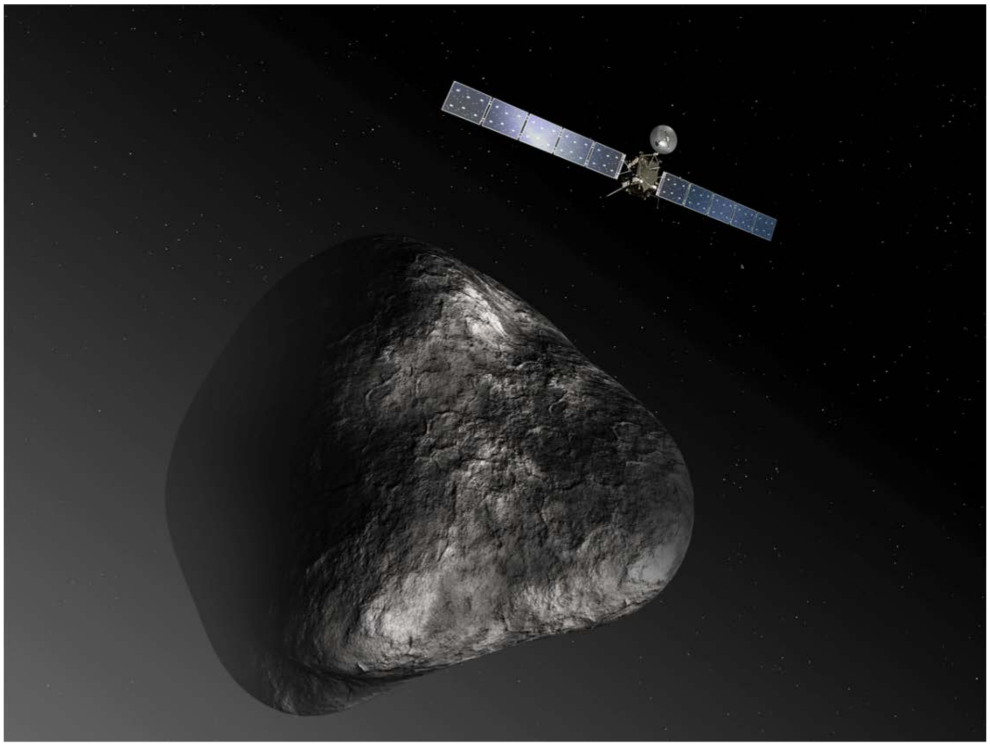
ENGSCI211 MATHEMATICAL MODELLING 2

PROJECT 2014 – PROBING THE UNIVERSE



Rosetta Probe and Comet Churyumov-Gerasimenko (Image Credit: ESA/ATG Medialab https://rosetta.jpl.nasa.gov/sites/default/files/gallery_image/Alexander-1PIA17666.png)

Background

In August this year, the European Space Agency (ESA) probe Rosetta will rendezvous with the comet *Churyumov-Gerasimenko*, hopefully becoming the first man-made object to land on the surface of a comet. The purpose of this mission is to gather and transmit data about the physical properties and chemical make-up of the comet, so that scientists can extend their knowledge and understanding of the Universe. Since no rocket is powerful enough to send probes to the comet directly, a number of *slingshots* around Earth and Mars have been used to propel Rosetta into the comet's path.

In this project you will consider modelling issues relating to space probes reaching their target destinations, as well as understanding of the data subsequently collected by them. Specifically:

In **Part 1**, you will be asked to determine the optimal time to launch a probe (Rosetta2), in order to slingshot around Mars and achieve a distance into the Solar System much greater than otherwise achievable.

In **Part 2**, you will analyse data obtained from previous probes (Voyager 1 and Galileo, which both also used slingshot manoeuvers to reach their destinations). In particular you will investigate differences in the craters of two of Jupiter's moons: Ganymede and Callisto.

File Resources

All files necessary to complete the project are available on Cecil from:

`ENGSCI 211 > Knowledge Map > Project > Resources >`

`ENGSCI 211 > Knowledge Map > Project > Template Files >`

Please save a copy of each of the files therein to a folder that you know will be backed-up regularly e.g. your University home folder (H-drive), your Google Drive folder, or your Dropbox folder etc.

Submissions

Project submission will be handled entirely electronically, to reduce as much as possible both paper and printing. In order to facilitate this, we request the following:

i) For **Part 1, Tasks 1.1a and 1.2a** please save your work in the template files `Task1.1a.docx` and `Task1.2a.docx`. Please save the final version of each of these as a PDF file and upload to Cecil at:

`Knowledge Map > Project > Submission > Task1.1a.pdf`

`Knowledge Map > Project > Submission > Task1.`~~`1b`~~`2a.pdf`

ii) For **Part 1, Tasks 1.1b and 1.2b**, please upload your completed copy of the M-file `MM2ProjectPart1.m` to Cecil at:

`Knowledge Map > Project > Submission > MM2ProjectPart1.m`

Note: You should upload the M-file itself, and **not** the output of code. We will re-run your submitted code, and generated the necessary output. Hence, if your code does not run, few marks for this portion of the project will be available to you.

iii) For **Part 2**, please save your work in the template files `Task2.1.docx`, `Task2.2.docx`, and `Task2.3.docx`. Please save the final version of each of these as a PDF file, and upload to Cecil at:

`Knowledge Map > Project > Submission > Task2.1.pdf`

`Knowledge Map > Project > Submission > Task2.2.pdf`

`Knowledge Map > Project > Submission > Task2.3.pdf`

Note that PDF is one of the file format supported on the *Save As* menu of recent versions of Microsoft Word. If you run into problems, you may scan a printed copy of the original Word document into PDF format.

Submission format

It is not mandatory to employ the original templates - which are provided only for convenience: Files produced by from another source are acceptable (e.g. a scanned copy of a neatly hand-written formulation for Task 1.1a). However, please ensure that the five submissions above (excepting the M-file) are all in PDF format.

Submission deadline

The deadline for all uploads is **1pm on Tuesday 29th April (Week 7)**.

In the interests of equity, transparency, and consistency, please be mindful of the following policy.

Policy on late submissions

Late submissions will be marked as usual, but will be subject to a penalty of 33.3% per day or part thereof i.e.

- 33.3% penalty on a submission made after 1pm on Tue 29/4 and before 1pm on Wed 30/4.
- 66.6% penalty on a submission made after 1pm on Wed 30/4 and before 1pm on Thu 1/5.
- 100% penalty on a submission made after 1pm on Thu 1/5.

If you miss the submission deadline, please email your files to engsci211@esc.auckland.ac.nz.

Please don't let late submission be an issue! Aim to have all files safely uploaded well before the due time.

It is highly recommended that you upload draft copies of all files at least one day in advance of the deadline.

Marking

A breakdown of marks according to project tasks will be published on Cecil.

Annotated copies of marked PDF submissions will be returned via email.

ENGSCI 211 FC Project 2014

Part 1: ODE Model

Background

We consider an Ordinary Differential Equation (ODE) model for the motion of the Rosetta2 probe around the Earth, Mars and the Sun. We will make the assumption that all these bodies move in the same plane, and hence use (x, y) coordinates to indicate their position, and (u, v) for corresponding velocities. We shall take the Sun to be stationary at $(0,0)$, and introduce the following notation:

(x_R, y_R) indicates the position of the Rosetta2 probe

(u_R, v_R) indicates the velocity of the Rosetta2 probe

(x_E, y_E) indicates the position of the Earth

(u_E, v_E) indicates the velocity of the Earth

(x_M, y_M) indicates the position of Mars

(u_M, v_M) indicates the velocity of Mars

$m_R = 2900\ \mathrm{kg}$ is the mass of the Rosetta2 probe

$m_E = 6 \times 10^{24}\ \mathrm{kg}$ is the mass of Earth

$m_M = 6.4185 \times 10^{23}\ \mathrm{kg}$ is the mass of Mars

$m_S = 2 \times 10^{30}\ \mathrm{kg}$ is the mass of the Sun

$r_{SE} = \sqrt{{x_E}^2 + {y_E}^2}$ is the distance between the Earth and the Sun

$r_{SM} = \sqrt{{x_M}^2 + {y_M}^2}$ is the distance between Mars and the Sun

$r_{RS} = \sqrt{x_R^2 + y_R^2}$ is the distance between the Rosetta2 probe and the Sun

$r_{EM} = \sqrt{(x_M - x_E)^2 + (y_M - y_E)^2}$ is the distance between Earth and Mars

$r_{RE} = \sqrt{(x_E - x_R)^2 + (y_E - y_R)^2}$ is the distance between the Rosetta2 probe and Earth

$r_{RM} = \sqrt{(x_M - x_R)^2 + (y_M - y_R)^2}$ is the distance between the Rosetta2 probe and Mars

$\alpha = GT^2 m_M / A_U^3 = 1.272466364267360 \times 10^{-5}$

$\beta = GT^2 m_S / A_U^3 = 39.65$

$\gamma = GT^2 m_E / A_U^3 = 1.1895 \times 10^{-4}$

(NB: $G = 6.67 \times 10^{-11}\ \mathrm{m}^3/\mathrm{kgs}^2$ is the Gravitational Constant).

To avoid excessively large numbers appearing in the calculations, we measure distances in Astronomical Units ($A_U = 1.496 \times 10^{11}$ m), masses in multiples of probe mass (i.e. $2900\ \mathrm{kg}$) and time in years (i.e. $T = 3.1536000 \times 10^7$ seconds).

Equations of Motion

According to Newton's Law of Gravity, the motion of Mars and Earth about the Sun (ignoring the minuscule influence that the Rosetta2 probe has on the orbits of Earth and Mars) is described the following system of Ordinary Differential Equations:

$$\frac{d^2x_E}{dt^2} = -\frac{\alpha}{r_{EM}^3}(x_E - x_M) - \frac{\beta}{r_{SE}^3}x_E$$
$$\frac{d^2y_E}{dt^2} = -\frac{\alpha}{r_{EM}^3}(y_E - y_M) - \frac{\beta}{r_{SE}^3}y_E$$
$$\frac{d^2x_M}{dt^2} = -\frac{\gamma}{r_{EM}^3}(x_M - x_E) - \frac{\beta}{r_{SM}^3}x_M$$
$$\frac{d^2y_M}{dt^2} = -\frac{\gamma}{r_{EM}^3}(y_M - y_E) - \frac{\beta}{r_{SM}^3}y_M$$

Equations 1

Similarly, the motion of the Rosetta2 probe is described by:

$$\frac{d^2x_R}{dt^2} = -\frac{\alpha}{r_{RM}^3}(x_R - x_M) - \frac{\gamma}{r_{RE}^3}(x_R - x_E) - \frac{\beta}{r_{RS}^3}x_R$$
$$\frac{d^2y_R}{dt^2} = -\frac{\alpha}{r_{RM}^3}(y_R - y_M) - \frac{\gamma}{r_{RE}^3}(y_R - y_E) - \frac{\beta}{r_{RS}^3}y_R$$

Equations 2

In early March 2012, both Earth and Mars were (approximately) aligned at their farthest point from the Sun (i.e. their Aphelions). Hence we shall take **1st March 2012** as our reference date, i.e. corresponding to time $t = 0$, when we therefore have the following initial conditions:

$x_E(0) = 1.01671388\ A_U$,
$y_E(0) = 0\ A_U$,
$u_E(0) = 0\ A_U/\text{Year}$,
$v_E(0) = -6.176590574607843\ A_U/Year$

$x_M(0) = 1.666\ A_U\cancel{/\text{Year}}$
$y_M(0) = 0\ A_U\cancel{/\text{Year}}$
$u_M(0) = 0\ A_U/\text{Year}$
$v_M(0) = -4.633\ A_U/\text{Year}$

ENGSCI 211 FC Project 2014

General Part I Instructions

You have been provided with a skeleton Matlab function, `MM2ProjectPart1.m`, which will enable you to complete the following Tasks. Lines of the code containing `YOURCODEHERE('Text')` should be replaced with your own Matlab commands (we advise against replacing or deleting any of the other existing code, which is needed for both functionality and marking purposes).

Additional resources

In addition to the (incomplete) template `MM2ProjectPart1.m`, the following Matlab files are available from the `Project > Resources` node on Cecil:

- `EarthMarsSystemReference.p`, which contains a complete version of the function `EarthMarsSystem` that needs to be completed for Task 1.1b.
- `Rosetta2IC.p`, which contains a function that computes the initial position and velocity of Rosetta2 immediately after its interplanetary engine burn, given positions and velocities for Earth and Mars at the time of engine burn.
- `YOURCODEHERE.p`, which contains a function that simply throws an error whenever it is invoked (perhaps with an optional error message). Ultimately, none of these calls will appear in your final version of `MM2ProjectPart1.m`.

These so-called *P-code* files are content-obscured versions of the corresponding M-files: The function contained in each may be invoked in the usual way, but the function definition cannot be viewed in a text editor. Please save all of these files to your working folder, which should also contain your copy of `MM2ProjectPart1.m`.

Initial conditions for Task 1.2b

A call to `Rosetta2IC` is provided inside the function `Task1_2b`. Please note that the physical interpretations of the components of this array (x_E, y_E, u_E, v_E etc.) are documented in the code comments.

Reference implementations

Testing your code

It is advisable to test your own ODE system function `EarthMarsSystem` by comparing its outputs with those of `EarthMarsSystemReference`.

Note that all calculations involving real numbers are inevitably subject to round-off errors on a digital computer. Hence, *very small* (but nonzero) differences may arise between the values of two mathematically equivalent Matlab expressions whose terms are evaluated in a different order.

A very small (but nonzero) difference between the outputs of your functions and those of the reference implementations does not necessarily mean that you have made an error.

The following code snippet illustrates a sensible test for "equality" of the results of two real-valued calculations on a digital computer:

```
% Example: "Equality test" in finite precision arithmetic
tol = 1e-12; % small but non-zero tolerance
x1 = sin(0.1);
x2 = sin(0.1 + 2*pi);
if abs(x1 - x2)/(abs(x1) + 1) < tol
  disp('the results match')
```

```
else
  disp('the results do not match')
end
```

Note that the choice of a suitable tolerance (`tol`, above) generally depends on the context. In this project we will be happy with answers accurate to 2 decimal places.

IMPORTANT: In order to achieve this order of accuracy, you will need to tighten the default tolerances used by `ode45`. The following code is already included in `MM2ProjectPart1.m`

```
options = odeset('RelTol', 1e-9, 'AbsTol', 1e-9);
```

and must be included in the call to ode45 in the following manner

```
ode45(@odefun, [T0 TF], IC, options),
```

where `odefun` is the function defining your ODE system, `T0` and `TF` are start and end times for the simulation, and `IC` is a vector of initial conditions.

Assisted completion of Part 1

If you are unable to complete the formulations in Task 1.1a or Task 1.2a, or if you are unable to get your implementations of `EarthMarsSystem` to match the output, then you may elect to adopt the reference version in your own code to enable you to complete the analysis in Tasks 1.1b.

This possibility is already highlighted in the comments provided in `MM2ProjectPart1.m`. In particular, the two line of code in the snippet below may be uncommented to guarantee correct functioning of `EarthMarsSystem`:

```
% NB: Please use the following code if you get stuck:
% drdt = EarthMarsSystemReference(t, r)
% return
```

This facility is provided only to enable you to proceed with the rest of the associated task: Please do not use this facility in your final submission unless you have had no success with your own code. Marks will be awarded for correctly functioning implementations of both of the ODE system functions.

For consistency across the ENGSCI211 class, please do make use of `EarthMarsSystemReference` inside of your own `Task1_2b` (even if you are satisfied that `EarthMarsSystem` is implemented correctly).

Task 1.1

Task 1.1a

Write Equations 1 as a system of 8 First Order ODEs, and save your formulation in `Task1.1a.pdf`.

Use the notation $r_1 = x_E,\ r_2 = y_E, r_3 = u_E,\ r_4 = v_E, r_5 = x_M,\ r_6 = y_M, r_7 = u_M,\ r_8 = v_M$.

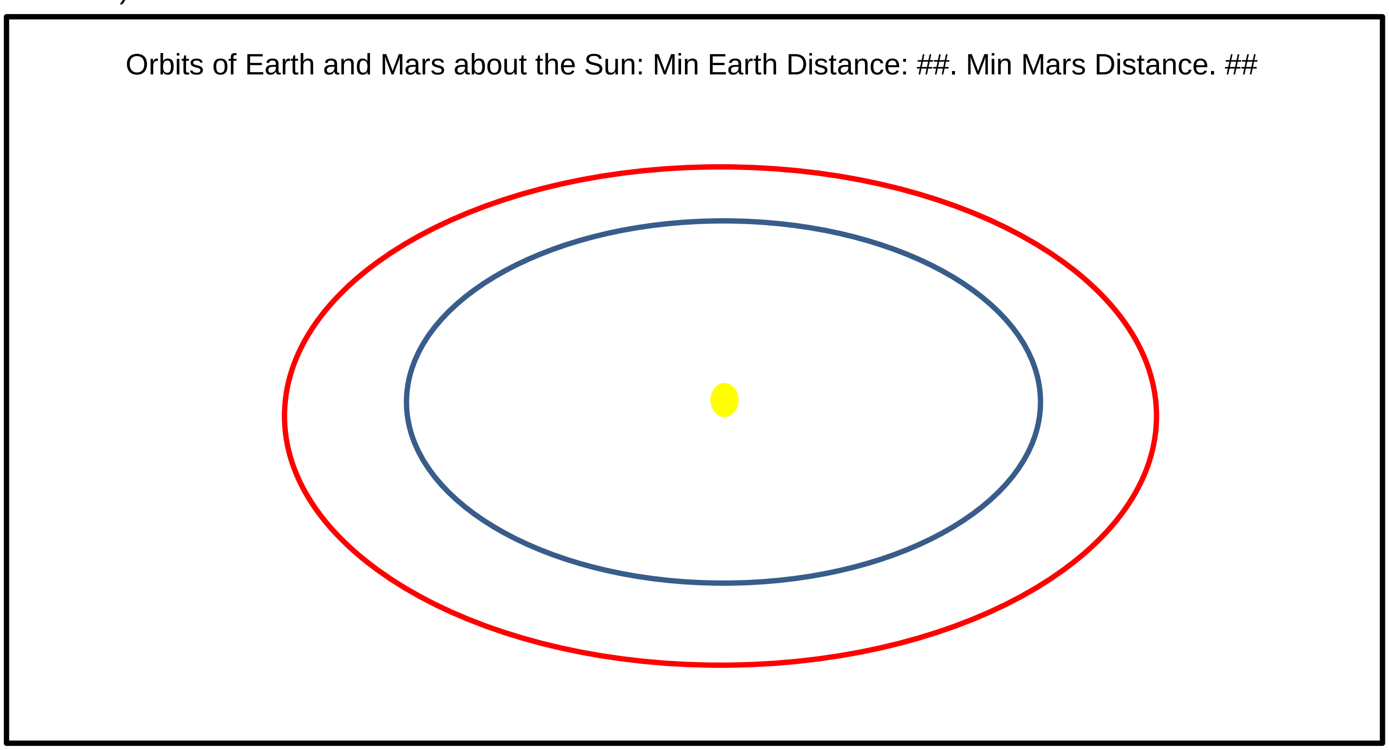
Task 1.1b

i. Complete the function `EarthMarsSystem` within `MM2ProjectPart1.m`, which solves Equations 1 using Matlab's `ode45` routine to compute the motion of Mars and Earth.

ii. Within the function `Task1_1b`, include the Matlab commands which use `ode45` and the function `EarthMarsSystem` to compute the orbits of Earth and Mars between 1st March 2012 and 1st March 2016. Recall that you will need to set `RelTol` and `AbsTol` to both take the value `1e-9` in `ode45`, in order to ensure sufficient numerical accuracy in your solution.

iii. Include some new code in the function `Task1_1b` to compute the closest distances that both Earth and Mars approach the Sun over this time (i.e. their Perihelion), in Astronomical Units, and insert these values into the variables `MinE` and `MinM`, respectively.

If the above tasks have been completed successfully, when `MM2ProjectPart1` is executed, a plot will be displayed that contains a figure similar in form to the sketch below (blue lines represents the orbit of Earth, red the orbit of Mars, and the yellow dot is the Sun. Hashes denote values of `MinE` and `MinM`, as computed by your code).



Task 1.2

The probe will be accelerated to an interplanetary escape velocity of 1.7125 A_U/Year from an orbital altitude of 14,960km. In this task you are asked to determine the correct date to accelerate the Rosetta2 probe to this escape velocity, in order to use Mars' gravity to slingshot the probe beyond the orbit that can be achieved through its own propulsive power alone. In this task you are asked to make the following changes to `MM2ProjectPart1.m`:

Task 1.2a

Write Equations 2 as a system of 4 First Order ODEs, and save your formulation in `Task1.2a.pdf`.

Use the notation of Task 1.1a, together with $r_9 = x_R,\ r_{10} = y_R, r_{11} = u_R, r_{12} = v_R$.

Task 1.2b

i. Complete the function `EarthMarsRosetta2System`, which encodes the system form of Equations 1 and 2 derived in Task 1.1a and Task 1.2a in a way that can be used by `ode45`. The function to encode Equations 1 has already been built in Task 1.1b, and can be called from within `EarthMarsRosetta2System,` and so here you are being asked to include additional code to include the motion of the Rosetta2 probe.

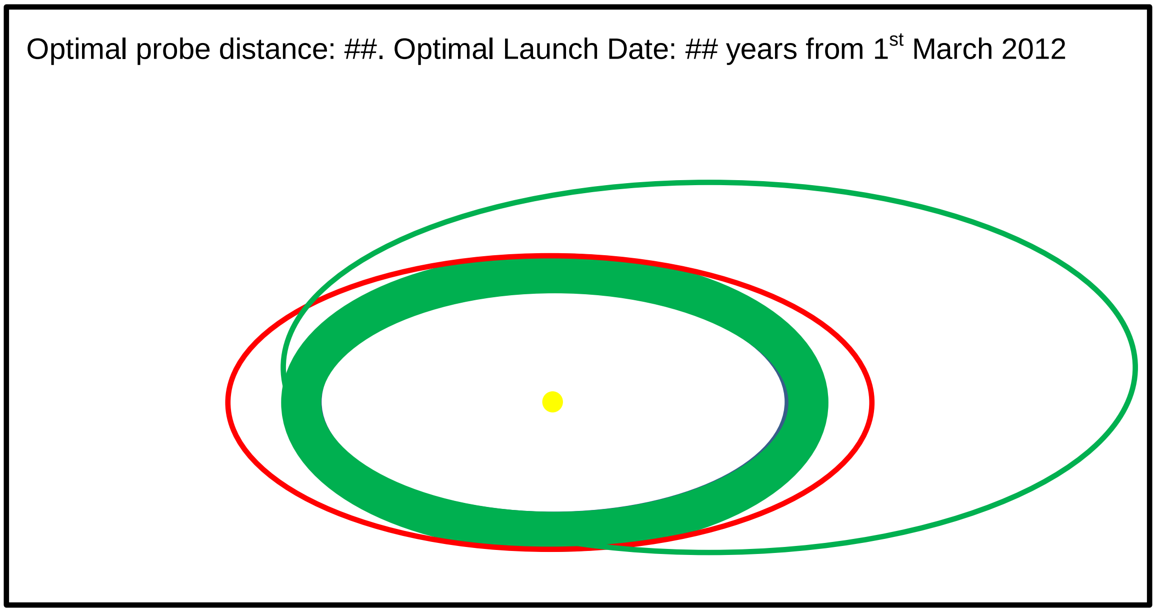
ii. Within the function `Task1_2b`, include commands to use `ode45` and the function `EarthMarsRosetta2System` to compute the orbits of Earth, Mars and the Rosetta2 Probe over a period of five years from a given launch date.

Note: Again, you will need to set `RelTol` and `AbsTol` to both take the value `1e-9` in `ode45`, in order to ensure sufficient numerical accuracy in your solution.

iii. Within the function `Task1_2b`, embed this call to `ode45` in a loop, which considers all possible launch times between the window 1st March 2030 and 1st March 2031, with a possible launch every ~~0.025~~ 0.0025 of a year (approximately one day). For each launch date, save the maximum distance the probe reaches in the variable `Reach`.

iv. Include the Matlab commands in `Task1_2b` to compute the best launch time in number of years after 1st March 2012 to achieve the greatest distance of the Rosetta2 probe from the Sun (for example, 18.23). Save this value in the variable `OptLaunch`, and the corresponding farthest distance in the variable `OptDistance`.

If the above tasks have been completed successfully, when `MM2ProjectPart1` is executed, a plot will be created that contains a figure similar to the following sketch (blue lines represents the orbit of Earth, red the orbit of Mars, green the orbit of the Rosetta2 probe and the yellow dot is the Sun. Hashes denote values of `OptLaunch` and `OptDistance`, as computed by your code).



Part 2: Data Analysis

One of the scientific foci of the Galileo and Voyager 1 missions was the investigation of the moons of Jupiter. In this part of the project we will consider data on the craters of the "Galilean Satellites" – the 4 large moons discovered by Galileo in 1610. No craters have been found on Io, and Europa is only sparsely cratered. However Ganymede and Callisto have a large spectrum of unusual and unique craters (at least 4 crater types are found nowhere else in the solar system[1]).

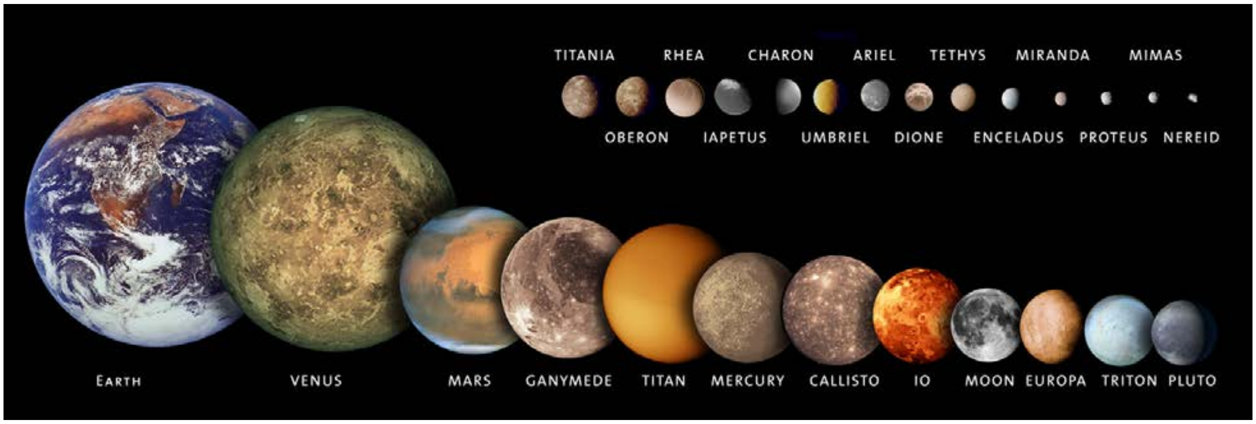
It is of great interest to scientists why Ganymede and Callisto are so different. Prior to the voyage of the Galileo probe it was thought that Ganymede and Callisto were composed of a rocky core surrounded by a large mantle of water or water ice with an ice surface. The Galileo data now suggest that Callisto has a uniform composition while Ganymede is differentiated into a three layer structure: a small molten iron or iron/sulphur core surrounded by a rocky silicate mantle with an icy shell on top. We will investigate whether this difference in composition has resulted in a difference in the distribution of impact craters across the two moons. The statistic we will consider is the diameter of the crater rim. Of the craters detected, the quality of many of the images collected make the calculation of the crater rim's diameter impossible, so we will treat the sample provided as a random independent sample from the population of interest.

The data for this part of the project is contained in the Excel spreadsheet *JupiterMoons.xlsx*, which is downloadable from the `Project > Resources` node on CECIL.

Note: The variable of interest is in column G of the first two spreadsheets. You will need to do some data manipulation before conducting your analysis (such as removing blanks that correspond to missing values and creating a text file for reading into R).

---

[1] http://www.lpi.usra.edu/resources/gc/gcreadme.html



Tasks for Part 2

Task 2.1

Estimate the mean rim diameter for craters on Ganymede.

Task 2.2

Estimate the mean rim diameter for craters on Callisto – is there an overlap with your estimate for Ganymede? (What would this suggest?)

Task 2.3

Determine whether there is a difference in the mean rim diameter for craters on Ganymede and Callisto.

Note: For all tasks you should include diagnostic plots and tests of assumptions as done in class. All R output should be annotated with relevant comments, and all findings should be presented in executive summaries.

References

http://www.dmuller.net/spaceflight/mission.php?mission=rosetta&appear=black&showimg=yes

http://sci.esa.int/rosetta/2279-summary/

http://solarsystem.nasa.gov/news/display.cfm?News_ID=7683