

Maastricht University

Department of Data Science and Knowledge Engineering

Titan Expedition

Project 1-2 - Report

Group 16

Authors:

Jan-Felix De Man (i6201411)

Selim Gilon (i6192074)

Gyumin Oh ()

Pietro Piccini (i6188338)

Gianluca Vico (i6183186)

Supervisors: ...

Examiners: ...

Block 1.6

Bouillonstraat 8-10, 6211 LH Maastricht, Netherlands

27 July 2019

Preface

Abstract

Bla bla bla with footnote ¹

¹Bla bla bla footnote

Contents

1	List	4
1.1	Abbreviations	4
1.2	Tables	4
1.3	Figures	4
2	Introduction	4
2.1	Problem description	4
2.2	Research questions	4
2.3	Report structure	4
3	Models	4
3.1	Solar System	4
3.1.1	Physical approach and alternatives	4
3.1.2	Equation solvers	5
3.1.2.1	Euler	5
3.1.2.2	Adams-Bashforth	5
3.1.2.3	3 point difference	5
3.1.2.4	Error analysis	6
3.2	Spaceship	6
3.2.1	Shape	6
3.2.2	Engines and uses	6
3.2.3	Rotation	6
4	Travel	7
4.1	Launch	7
4.1.1	From Earth	7
4.1.2	From Titan	7
4.2	Trajectory	7
4.2.1	Algorithm	7
4.2.2	To Titan	7
4.2.3	To Earth	7
5	Landing	7
5.1	On Titan	7
5.1.1	Given tolerances	7
5.1.2	Stochastic wind model	8
5.1.3	Feedback controller	8
5.1.4	Open loop controller	8
5.2	On Earth	8
6	Optimizations	8
6.1	Fuel efficiency	8
6.2	Quickest route to Titan	8
6.3	Math optimizations	8
7	GUI	8
8	Experiments	9

9 Results and discussion	9
10 Conclusion	10

1 List

1.1 Abbreviations

Bla bla bla

1.2 Tables

Bla bla bla

1.3 Figures

Bla bla bla

2 Introduction

2.1 Problem description

2.2 Research questions

2.3 Report structure

3 Models

3.1 Solar System

3.1.1 Physical approach and alternatives

To digitally simulate our solar systems we used Newton's gravitational theorem. The model is based on equation $\vec{F} = \frac{\vec{r}}{|\vec{r}|} G \frac{m_1 m_2}{|\vec{r}|^2}$ ², this means that the gravitational force F is equal to the product of the masses of both objects, divided by the distance squared, multiplied by the gravitational constant G . Or in other words: the bigger the mass and the closer the distance, the bigger the force. To implement this model, it is necessary to pick a certain point in time and to know where the planets were located at that time. Once all the celestial bodies are loaded in their original place, the gravity will do its work, which results in the planets orbiting around the Sun and the moons around their respective planets.

Another possibility was to simulate our solar system using an orbital model. This approach is simpler: draw the sun and surround it with ellipses that represent the size of the orbits of the planets.

Newton's model has been chosen instead of the orbital model, because more physically accurate and it can show the evolution of the Solar System. Moreover it only needs the initial conditions of each body and the gravitational formula. On the other hand calculating the position of a body at a given time

²(2019). Retrieved from <https://www3.nd.edu/~z xu2/acms40390F15/Lec-5.6.pdf>

could be tricky and the complexity of the algorithm increases exponentially if the number of bodies in the simulation increases.

3.1.2 Equation solvers

The whole simulated solar system works on just equation: Newton's gravitational theorem. In order to implement this and actually make the celestial bodies orbit, you have to solve this equation. There are multiple ways to do it, represented are the ones used in the making of this report.

3.1.2.1 Euler

Euler's method is a numerical method that solves first order first degree differential equations. It is one of the most simple ODE-solvers, yet still relatively accurate.

$$\dot{y}(t) = f(t, y) \quad (1)$$

$$y_i \approx w_i = w_{i-1} + hf(t_{i-1}, w_{i-1}), \text{ with } i > 0 \quad (2)$$

This methods as a global error $O(h)$, require only one function evaluation at each iteration and it is easy to implements.

3.1.2.2 Adams-Bashforth

We decided to use Adams-Bashforth explicit 4 steps method to solve the different equations. This method has been implemented and replaced the Euler's method to have a higher accuracy in the calculus of the simulation. As mentioned earlier, Euler's method is first order while Adams-Bashforth is a 4th order equation solver. Adams-Bashforth's method is more accurate than Euler's, that is why it is recommended to work with the Adams-Bashforth method. This method is defined as

$$\dot{y}(t) = f(t, y) \quad (3)$$

$$y_{i+1} \approx w_{i+1} = w_i + \frac{h}{24}[55f(t_i, w_i) - 59f(t_{i-1}, w_{i-1}) + 37f(t_{i-2}, w_{i-2}) - 9f(t_{i-3}, w_{i-3})], \text{ with } i > 3 \quad (4)$$

To initialize this method, data from Nasa are used if available. The acceleration is calculated by using 3 points difference formula and data for the spaceship are assumed to be linear.

The main advantage of the approach is that it reduces the error (from $O(h)$ for Euler's to $O(h^4)$ for Adams-Bashforth's) without increasing the computational cost, because both of the require only a function evaluation at each iteration.

3.1.2.3 3 point difference

It was not possible to obtain information about the acceleration of the celestial bodies from Nasa, so it is necessary to calculate it. The acceleration is defined as $\vec{a} = \frac{d\vec{v}}{dt}$. So the values of the acceleration can be calculated as following:

$$\text{Forward difference: } \vec{a}_i = \frac{-3\vec{v}_i + 4\vec{v}_{i+1} - \vec{v}_{i+2}}{2h} \quad (5)$$

$$\text{Centered difference: } \vec{a}_i = \frac{\vec{v}_{i+1} - \vec{v}_{i-1}}{2h} \quad (6)$$

$$\text{Backward difference: } \vec{a}_i = \frac{3\vec{v}_i - 4\vec{v}_{i-1} + \vec{v}_{i-2}}{2h} \quad (7)$$

3.1.2.4 Error analysis

3.2 Spaceship

3.2.1 Shape

Our rocket is designed to be a spherical shell that is hollow inside. This makes it easier to calculate forces caused by external factors, such as wind and pressure, that affect the rocket. The effects of the wind is determined by the surface area: that more area that can be blown by the wind, the bigger the force. The choice of this shape, although not the most aerodynamic, which leads to increased costs, does help to control the rotation, increases the shuttle's stability and efficiency.

3.2.2 Engines and uses

Our rocket is uses 7 engines which control velocity and rotation. The engines are divided into 2 categories: 1 main engine and 6 sub engines. Main engine allows us to control the rocket's velocity and acceleration. The sub engines change the rotation of our vehicle for the different axis x, y, z. For every axis, or dimension, we have 2 sub engines or thrusters. This helps us establish better control over our rocket. This is very important in the case of heavy winds. Main engine acceleration / deceleration; thrusters to balance out wind.

3.2.3 Rotation

Spaceship's rotations are calculated from torque momentum formula $\vec{L} = I\vec{\omega}$, where r is the distance between the application point of the force and the center of the spaceship, and the angular momentum formula $\vec{L} = I\vec{\omega}$. Because of the particular shape of the spaceship inertia momentum is $I = \frac{2}{5}m \frac{r_2^5 - r_1^5}{r_2^3 - r_1^3}$. Then

the angular velocity could be approximated as $\vec{\omega} = \frac{\sum_{i=0}^n (\vec{r}_i \times \vec{F}_i) \Delta t}{I}$ and the rotation is $\vec{\theta} = \frac{d\vec{\omega}}{dt}$, which is calculated with Adams-Bashforth's method.

To calculate the orientation of the spaceship three perpendicular axis are used: x and y are horizontal and z is the vertical one. Orientation is computed by using axis-angle rotations: given the unit axis of rotation \vec{e} and the angle θ , the axis of the spaceship are updated to

$$\vec{v}^* = (\cos \theta) \vec{v} + (\sin \theta) (\vec{e} \times \vec{v}) + (1 - \cos \theta) (\vec{e} \cdot \vec{v}) \vec{e} \quad (8)$$

4 Travel

4.1 Launch

4.1.1 From Earth

For the launch on Earth we use the same point in time as we initialize our digital Solar System, which is the 22nd of March 2019 at midnight. We launch from really close to the equator (latitude -0.0016°) facing Titan

4.1.2 From Titan

4.2 Trajectory

The trajectory is determined by the starting velocity of the rocket and the gravity influenced on the rocket by the celestial bodies in the solar system, according to Newton's gravitational theorem (see 3.1.1.).

4.2.1 Algorithm

To find a trajectory we developed an algorithm that finds the initial vector velocity of the rocket such that the rocket reaches Titan's atmosphere. The algorithm works by launching the rocket towards Titan with a generic initial velocity. After a determined amount of time the rocket stops and the vector distance between the rocket and Titan is stored and used to correct the rocket's initial velocity, then it is launched again with the improved initial velocity. This process is repeated and the correction made to the initial velocity is proportional to the distance between the rocket and Titan at the end of the launch. This is useful to avoid over-correcting and guarantees that the rocket will find a trajectory after a finite number of iterations. Moreover the correction factor is changed when the rocket can go close to Titan, in this way it oscillates less around the optimal solution and converges faster.

4.2.2 To Titan

4.2.3 To Earth

5 Landing

When the shuttle managed to orbit around the target planet, it will start an automated landing. The landing is coordinated by two built-in controllers. These algorithms help to land our shuttle safely on the surface of the planet.

5.1 On Titan

5.1.1 Given tolerances

In order to assure a safe landing, it is required to respect multiple parameters. A landing is considered safe when the velocity is less than 0.1, the angular velocity less than 0.01, the angle between the vertical axis of the spaceship and the ground is less than 0.02 and the distance to the ground less than 0.1.

5.1.2 Stochastic wind model

5.1.3 Feedback controller

5.1.4 Open loop controller

The computation for the velocity during the landing is done by using an open loop controller: at 50'000 km the velocity is reduced to 500 km/h (with respect to Titan) with the main engine and with the same direction of the velocity of Titan. Then at 600km, when the spaceship enters the atmosphere, it starts braking to have the same velocity of the planet. During this phase the feedback controller for the rotation is still working and a three dimensional space is used.

5.2 On Earth

6 Optimizations

6.1 Fuel efficiency

Every time that the spaceship uses its engines it consume fuel, so it reduces its mass. The total mass at the launch is 20'000 kg, 15'000 kg are fuel. When it lands on Titan it has consumed 5529.55 kg of fuel. The cost of the fuel is assumed to be the same of kerosene (1.68 \$ / gallon ³, about 0.55 \$ / kg), then the cost is about \$ 3067.58. The estimate of the cost when the rocket start the landing at 50'000 km from Titan is \$ 1181.00. The estimation assumes that there are no rotation nor atmosphere and it calculates the constant acceleration needed to land with a null velocity and a distance of zero meters from the ground. From constant acceleration motion formula it is possible to derive that $\vec{a} = \frac{|\vec{v}|^2}{2h}$, where h is the distance from the ground. Then the mass M of fuel need is $M = \frac{ma}{F_{engine}} m_{engine}$ and the cost is calculated by using the mass M .

6.2 Quickest route to Titan

6.3 Math optimizations

7 GUI

The program described in this report calculates everything in the three dimensions. The solar system it simulates is visualized in 3D. There are added functionalities to make this program more user-friendly: use the buttons A and D to rotate the camera view, press 0 to reset it to the standard rotation. The buttons W and S can be used respectively to zoom in and out. If you press E, the camera will be focused on Earth, T will focus on Titan, U will set the view to Saturn. Furthermore you can use the arrow keys to roam around freely in space.

³Jet Fuel - Monthly Price Charts, Data, and News - IndexMundi. Retrieved from <https://www.indexmundi.com/commodities/?commodity=jet-fuel&months=12¤cy=eur>

As soon as we initialize our landing, the GUI switches to a 2D visualization of our rocket landing on the celestial body. All the calculations it uses are still made in three dimensions. You can see the spherical-shaped rocket getting closer to the planet's surface, while it gets affected by the wind. You see the rocket correcting its rotation and speed before it hits the surface.

8 Experiments

Since our rocket travels with a speed of approximately 56'118 km/h, we experimented with lower speeds to make it more realistic. To do so, the initial velocity with which the rocket is launched has been reduced. When changing the velocity to realistic speeds, the shuttle ended up orbiting around the Sun, or sometimes even worse, Earth itself. But if the speed is realistic, then why does not it work in this simulation? In this case, the rocket is shot into space from our launching station, and does not use its engines until it reaches the orbit of the desired celestial body. Therefore it will get stuck in the orbit of either the Sun or Earth. In real life, a rocket uses its own engine to leave Earth orbit and to steer itself to its desired location. Simulating this process would mean that you would need to rethink your whole launching and trajectory approaches, thus would be very time consuming. We see this as a challenge we would like to work on later.

Since our wind model rightfully predicts a low wind on the surface, we wanted to test the controller for stormier weather as well. We did find data about the wind on Titan and Earth, but we do not want to be surprised by having our rocket crash after all the effort we put into getting it to the planet.

9 Results and discussion

It is possible to test how robust is the feedback controller by increasing the effect of the rotations generated by the wind.

Rotation influence	x1	x10	x100	x1000
Angular velocity (rad/s)	0.0012	0.0099	0.0755	0.8471
Angle (rad)	0.0093	0.0093	0.0093	0.0092

Then the values of the angular velocity and the angle of the spaceship are within the tolerance range, even with rotations 1000 times stronger.

It is possible to reduce the initial velocity of the spaceship from Earth to test if it can land.

Velocity compared to optimal	Result - cost (€)
99%	Fail
99.9%	Fail (in Titan gravitational field)
100%	3067.58
100.1%	Fail (in Titan gravitational field)
101%	Fail

Then trajectory is really sensitive to the initial conditions and the spaceship completely miss Titan if there are small changes at the beginning. The velocities used have the same direction of the optimal one.

10 Conclusion

We are planning to search for a more realistic launching velocity by using other celestial bodies to improve the trajectory (in particular Jupiter seems promising) and by organizing a longer travel, for example a 8-year-long travel similar to Cassini mission. Then it is possible to improve the orbit around Titan: at the moment the spaceship can complete a few orbits around the satellite, but because of the speed and the Saturn gravitational force it cannot stay longer. By reducing the velocity it will be possible to orbit longer. A feedback controller that tries to reduce the distance between Titan and the spaceship can used, this way it is possible to reduce the landing velocity more and to have a more realistic landing. Finally it is possible to explore other numerical methods to improve the accuracy of the whole simulation.

Bibliography

1. Rincon, P. (2005). BBC NEWS — Science/Nature — Huygens sends first Titan images. Retrieved from <http://news.bbc.co.uk/2/hi/science/nature/4175099.stm> [Accessed 6 May 2019].
2. Bortman, H. (2010). Life Without Water And The Habitable Zone. Retrieved from http://www.spacedaily.com/reports/Life_Without_Water_And_The_Habitable_Zone_999.html [Accessed 4 May 2019].
3. Bartels, M. (2019). NASA May Decide This Year to Land a Drone on Saturn’s Moon Titan. Retrieved from <https://www.space.com/43010-dragonfly-mission-would-put-a-drone-on-titan.html> [Accessed 17 June 2019].
4. Retrieved from <https://www3.nd.edu/~zxu2/acms40390F15/Lec-5.6.pdf> [Accessed 1 May 2019].
5. Hendrix, A. and Yung, Y. (2017). Energy Options for Future Humans on Titan. [online] Available at: <https://arxiv.org/ftp/arxiv/papers/1707/1707.00365.pdf> [Accessed 2 May 2019].
6. Carey, B. (2005). New View of Titan: Strong Winds, Soft Ground and Lightning. Retrieved from <https://www.space.com/1825-view-titan-strong-winds-soft-ground-lightning.html> [Accessed 3 May 2019]

7. Jet Fuel - Monthly Price (Euro per Gallon) - Commodity Prices - Price Charts, Data, and News - IndexMundi. Retrieved from <https://www.indexmundi.com/commodities/?commodity=jet-fuel> [Accessed 18 June 2019].

Appendix

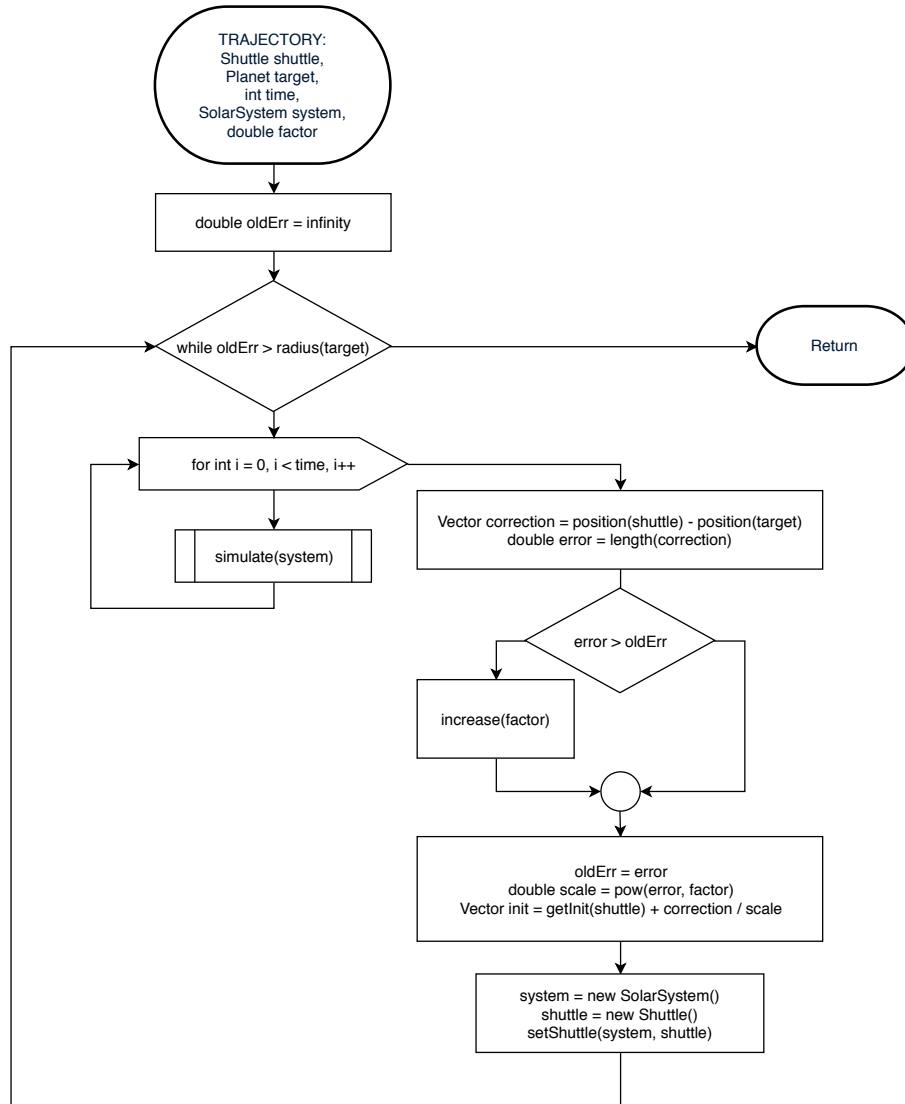


Figure 1: Algorithm to calculate the trajectory to reach a planet.