MSAS – Final project

Alejandro de Miguel, Carmen Salas, Rafael Félix

1 The real system

Gravity Field and Steady-State Ocean Circulation Explorer (GOCE) was a science satellite designed and operated by European Space Agency, developed to measure Earth´s gravitational field with high accuracy. In order to achieve its purpose, it employed the concept of gradiometry, being equipped with six high-sensitivity accelerometers arranged along the three axes of the spacecraft.

Furthermore, to be able to acquire high-resolution measurements it had to travel at a low orbital altitude, leading to significant atmospheric drag effects to be compensated with continuous ion propulsion. In order to achieve these requirements and compensate for the variable drag, it was equipped with a control system able to detect the drag deceleration thanks to the capacitive accelerometer aligned with the direction of the velocity, and regulate the ion propulsive system in consequence. More precisely, the input of the control system is the distance the charged inertial mass (which at the beginning is electrostatically suspended in equilibrium in vacuum between two electrodes) moves away from the electrodes due to the drag. This measure is then used to produce a control voltage able to return the mass to the equilibrium position (and so compensate the drag perturbation). The control voltage is also related to the intensity of an inductor, which transmits a force onto the spool that controls the valve (and so, the mass flow rate and consequently the thrust) of the xenon motor. This way, the drag deceleration is compensated by the thrust.

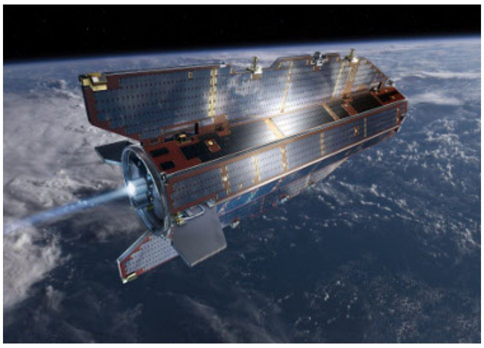


Figure 1: Artist's impression of GOCE

2 The physical model

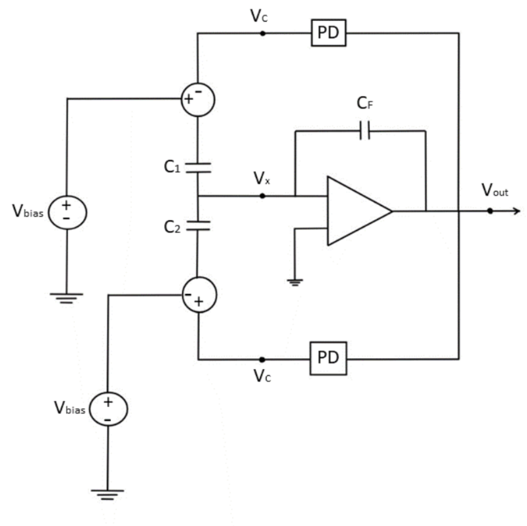
The main components to be considered in order to model the drag free and attitude control system of GOCE are the Electrostatic Gravity Gradiometer to measure the perturbations, the valve that controls the mass flow rate and the ion propulsion system providing the thrust. Furthermore, the motion and the perturbations suffered by the spacecraft have to be modeled (via orbital mechanics) to extract the deceleration value that the system needs to compensate.

To model the orbital mechanics some simplifications have been done. First of all the shape of the Earth has been considered to be an ellipsoid with semi-major axis equal to 6378.16 km and semi-minor axis equal to 6356.778 km. The satellite is considered to be a punctual mass in an orbit with 254.9 km altitude, 0.0045 eccentricity and 90º inclination. Furthermore, only J_2 and atmospheric drag perturbing effects have been taken into account due to the fact that these are the most relevant ones for the current case. Earth's oblateness is dominant above the other gravitational perturbations, and aerodynamic perturbation has to be considered because the orbit is too low and close to the atmosphere to neglect it.

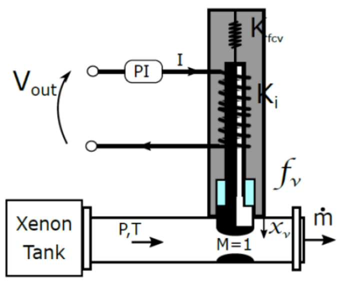
The real Electrostatic Gravity Gradiometer consists of 3 pairs of accelerometers. However in this model it has been simplified to just one accelerometer in the main direction of movement of the spacecraft, assuming that in the other directions the perturbations are compensated. The accelerometer is capacitive, with an electrostatically suspended platinum-rhodium inertial mass which position variation is detected and compensated by the electrodes. When the spacecraft is perturbed by the drag deceleration, the mass moves away from the equilibrium position a measurable distance. This measure is then used to produce a control voltage able to return the mass to the equilibrium position (and so compensate the drag perturbation).

The voltage obtained as an output from the accelerometer is processed by a control system to compute the intensity that the solenoid valve needs in order to open the valve just enough for the motor to compensate the drag. The valve controls the mass flow rate of the motor, and so, its thrust. This system is modeled like a chocked nozzle whose throat area is controlled by the intensity passing through the solenoid.

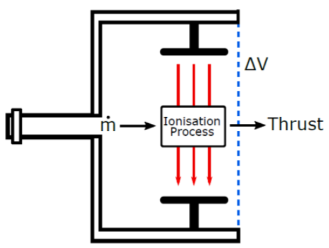
Lastly, the ion thruster ionizes xenon using a magnetic field. These ions are moved towards an acceleration grid where they are accelerated by the Coulomb force along a strong electric field to be then expelled out, thus producing the thrust. The complete ion thruster system has been model as a one lump system that takes the mass flow rate as an input and returns the consequent thrust as an output.



a: Accelerometer physical model



b: Solenoid valve physical model



c: Thruster physical model

Figure 2: Physical model of the main components

3 The mathematical model

Based on the physical model described before, the set of differential equations defining the orbit of the spacecraft around Earth is written by means of the Gauss planetary equations. These equations describe the variation of each of the keplerian elements over time. The advantages of this set of equations is that we can easily introduce all the orbit perturbations that the spacecraft is subjected to, such as the J2 effect, drag, and low-thrust, and that we can easily observe how these perturbations affect each of the orbital elements. For example, in our case, due to the air drag perturbation we expect a diminution in the semi-major axis and the eccentricity. The Gauss planetary equations are:

$$\dot{a} = \frac{2a^2v}{\mu}a_t \tag{1}$$

$$\dot{e} = \frac{1}{v}\left(2(e+\cos\theta)a_t - \frac{r}{a}\sin\theta a_n\right) \tag{2}$$

$$\dot{i} = \frac{r\cos\theta^*}{h}a_h \tag{3}$$

$$\dot{\Omega} = \frac{r\sin\theta^*}{h\sin i}a_h \tag{4}$$

$$\dot{\omega} = \frac{1}{ev}\left(2\sin\theta a_t + \left(2e + \frac{r}{a}\cos\theta\right)a_n\right) - \frac{r\sin\theta^*\cos i}{h\sin i}a_h \tag{5}$$

$$\dot{\theta} = \frac{h}{r^2} - \frac{1}{ev}\left(2\sin\theta a_t + \left(2e + \frac{r}{a}\cos\theta\right)a_n\right) \tag{6}$$

with $v = \sqrt{\frac{2\mu}{r} - \frac{\mu}{a}}$, $r = \frac{p}{1+e\cos\theta}$, $p = b^2/a$, $n = \sqrt{\mu/a^3}$, $h = nab$, $\theta^* = \theta + \omega$, $b = a\sqrt{1-e^2}$. Note also that a_t, a_n, a_h are the components of the perturbing acceleration $\mathbf{a}_p$, and can be obtained from $\mathbf{s}_{car} = \{\mathbf{r}, \mathbf{v}\}$ with a rotation matrix $\mathbf{A} = [\hat{\mathbf{t}}, \hat{\mathbf{n}}, \hat{\mathbf{h}}]$.

With this model, all the perturbations can be simply added up as $\mathbf{a}_p = \mathbf{a}_{J2} + \mathbf{a}_{drag} + \mathbf{a}_{thrust}$ where the different perturbations are modeled as:

$$\mathbf{a}_{J2_{car}} = \frac{3}{2}\frac{J2\mu R_e^2}{r^4}\left[\frac{x}{r}\left(5\frac{z^2}{r^2} - 1\right)\hat{\mathbf{i}} + \frac{y}{r}\left(5\frac{z^2}{r^2} - 1\right)\hat{\mathbf{j}} + \frac{z}{r}\left(5\frac{z^2}{r^2} - 3\right)\hat{\mathbf{k}}\right] \tag{7}$$

$$\mathbf{a}_{drag} = -\frac{1}{2}\frac{AC_D}{m}\rho(h,t)v_{rel}^2\frac{\mathbf{v}_{rel}}{||\mathbf{v}_{rel}||} \quad \text{with} \quad \mathbf{v}_{rel} = \frac{d\mathbf{r}}{dt} - \omega_{Earth} \times \mathbf{r} \tag{8}$$

$$\mathbf{a}_{thrust} = \frac{T}{m}\frac{\mathbf{v}_{rel}}{||\mathbf{v}_{rel}||} \tag{9}$$

Note that $\mathbf{a}_{J2_{car}}$ must be rotated to the $[\hat{\mathbf{t}}, \hat{\mathbf{n}}, \hat{\mathbf{h}}]$ system, $\mathbf{a}_{drag}$ has to be projected onto the spacecraft velocity direction and $\mathbf{a}_{thrust}$ is assumed to be aligned with the spacecraft velocity direction.

In view of the physical model previously represented of the accelerometer, the mathematical model can be obtained solving the electrical circuit as follows:

$$C_1 = \epsilon\frac{A}{g - x_A}; \quad C_2 = \epsilon\frac{A}{g + x_A}$$

Both capacitors C_1 and C_2 can be expressed together as an equivalent one

$$C_{eq} = \frac{C_1C_2}{C_1 + C_2}$$

Considering everything constant and aplying Ohm's law:

$$V_x = \frac{2V_{bias}C_{eq}}{C_2} - V_{bias} = \frac{C_1 - C_2}{C_1 + C_2}V_{bias} = \frac{x}{g}V_{bias}$$

V_C is obtained from a proportional derivative (PD) controller as:

$$V_C = K_{pa}V_{out} + K_{da}\dot{V}_{out}$$

For the line containing the capacitor C_F:

$$i_c = i_F = \frac{dQ}{dt} = \frac{d(C_F V_{bias})}{dt} = V_{bias}\frac{\partial C_F}{\partial x_a}\frac{dx_a}{dt} C_F = C_1 - C_2 V_{out} = -V_F i_F = C_F\dot{V}_F = -C_F\dot{V}_{out} \quad ,$$

so that $\dot{V}_{out}$ can be expressed as:

$$\dot{V}_{out} = -\frac{1}{c_f}2\epsilon A_a\frac{g^2 + x_a^2}{(g^2 + x_a^2)^2}v_a V_{bias} \tag{10}$$

Finally, applying equilibrium of forces to the suspended mass:

$$m\ddot{x}_a = F_{ext} - F_{e1} + F_{e2} \quad ,$$

where F_{ext} are the external forces (drag and thrust for this case) and $F_{ei} = \frac{1}{2}\epsilon A\frac{\Delta V_i^2}{\delta_i^2}$ is the electric force. Finally it results to be

$$m\ddot{x}_a = -D + T - \frac{1}{2}\epsilon A\frac{(V_{bias} - V_C - \frac{1}{2}V_x)^2}{(g - x_a)^2} + \frac{1}{2}\epsilon A\frac{(V_{bias} + V_C + \frac{1}{2}V_x)^2}{(g + x_a)^2} \tag{11}$$

Once the output of the accelerometer is know, the mathematical model of the valve and ion thruster can be obtained. Firstly the intensity through the solenoid is obtained from a proportional integral controller:

$$I = K_{pv}V_{out} + K_{iv}\int_0^t V_{out}dx \quad ,$$

that can also be expressed as a differential equation as:

$$\dot{I} = K_{pv}\dot{V}_{out} + K_{iv}V_{out} \tag{12}$$

Applying now equilibrium of forces to the valve:

$$m_{fcv}\ddot{x}_v = k_{fcv}(10A_0 - x_v) - K_i I - cv_v \tag{13}$$

The mass flow rate is defined as:

$$\dot{m} = \frac{dm}{dt} = \rho v A$$

Assuming chocked flow ($M_t = 1$) and compressible gas in every moment, it can be rewritten as:

$$\dot{m} = C_d A_v\sqrt{\frac{2}{RT_2}}\sqrt{\frac{k}{k-1}}p_2\left[\left(\frac{2}{k+1}\right)^{\frac{2k}{k(k-1)}} - \left(\frac{2}{k+1}\right)^{\frac{k(k+1)}{k(k-1)}}\right]^{\frac{1}{2}} \quad , \tag{14}$$

where C_d is the discharge coefficient that has been considered to be 0.61. T_2 and p_2 are the temperature and pressure of xenon before the nozzle and A_v is the variable area of the nozzle's throat, simulated like the one of a cylindrical pipe that closes due to a perpendicular piston.

$$A_v = A_0\frac{(\alpha - sin(\alpha))}{2\pi} \quad ;\alpha = 2\pi - 2acos(1 - 2\frac{x_v}{10A_0})$$

For the ion thruster, the velocity of a charge accelerated by an electric field due to a potential difference ΔV can be obtained from a energy balance between kinetic and electric energy:

$$\frac{1}{2}m_p v^2 = q\Delta V \quad ,$$

where m_p is the particle mass (xenon mass in this case) and q is the electron charge. Knowing than $T = v\dot{m}$ the thrust resuts to be:

$$T = \dot{m}\sqrt{\frac{2q\Delta V}{m_p}} \tag{15}$$

After rearranging all the equations the mathematical model to be integrated results to be the following system added to equations Eq. (1)-Eq. (6).

$$\begin{cases} \dot{x_a} = v_a \\ \dot{v_a} = \dfrac{1}{m}\left(-D + T - \dfrac{1}{2}\epsilon A\dfrac{(V_{bias} - V_C - \frac{1}{2}V_x)^2}{(g - x_a)^2} + \dfrac{1}{2}\epsilon A\dfrac{(V_{bias} + V_C + \frac{1}{2}V_x)^2}{(g + x_a)^2}\right) \\ \dot{V_{out}} = -\dfrac{1}{c_f}2\epsilon A_a\dfrac{g^2 + x_a^2}{(g^2 + x_a^2)^2}v_a V_{bias} \\ \dot{x_v} = v_v \\ \dot{v_v} = \dfrac{1}{m_{fcv}}[k_{fcv}(10A_0 - x_v) - K_i I - cv_v] \\ \dot{I} = K_{pv}\dot{V_{out}} + K_{iv}V_{out} \end{cases} \tag{16}$$

4 Numerical integration

To select an adequate integration scheme a previous analysis of the mathematical model has to be done, to detect for example if we are facing a stiff or a non-stiff problem. A first approach to this analysis is computing the eigenvalues of the system to check how far apart they are from each other. If they differ by one or more orders of magnitude, it means that the problem faced is stiff, and consequently the integrator used will be an implicit one.

In order to compute the eigenvalues of the system, first there is the need to linearize the set of equations using a Taylor expansion up to first order: $\mathbf{f}(\mathbf{x}) = \mathbf{f}(\mathbf{x}_{eq}) + \mathbf{f}'(\mathbf{x}_{eq})(\mathbf{x} - \mathbf{x}_{eq})$, where $\mathbf{f}'$ is the Jacobian of the system. As $\mathbf{f}(\mathbf{x})$ is the $\dot{\mathbf{x}}$ of the system, it turns zero in the equilibrium point, yielding: $\mathbf{f}(\mathbf{x}) = \mathbf{f}'(\mathbf{x}_{eq})(\mathbf{x} - \mathbf{x}_{eq})$, which is the equivalent of $\dot{\mathbf{x}} = \boldsymbol{A}\mathbf{x}$, and so, the eigenvalues of $\boldsymbol{A}$ can be computed as the eigenvalues of the Jacobian by solving $det(\lambda\boldsymbol{I} - \boldsymbol{f}') = 0$. Note that, as the orbital equations (Eq. (1)-Eq. (6)) do not have any equilibrium point due to the true anomaly θ, being always increasing, they have to be discarded from the analysis. Therefore, it will be just considered an ideal circular orbit with constant altitude (and so constant ρ, D, T...). The remaining 6 equations have to be linearized around an equilibrium point, where the derivatives of the variables are obviously zero and the thrust is equal to the drag. Therefore, substituting the equilibrium point in the Jacobian, the eigenvalues of the system result to be:

$$\begin{aligned} \lambda_1 &= -100232.796 \\ \lambda_2 &= -19.989 \\ \lambda_3 &= -0.040 \\ \lambda_4 &= -1.780 \times 10^{-39} \\ \lambda_5 &= -74.987 - 171.385i \\ \lambda_6 &= -74.987 + 171.385i \end{aligned} \tag{17}$$

From a simple analysis of the eigenvalues two main conclusions can be obtained concerning the present problem.

- The system is stable as all the eigenvalues have negative real part.
- The problem to be solved is stiff as the eigenvalues differ various orders of magnitude.

It has been decided to utilize Matlab integrated ODEs solvers to compute the solution. Therefore once the problem is known to be stiff, different implicit solvers available for this kind of systems have been compared using various tolerances.

Table 1: ODE solvers comparison.

ODE solver	Tolerance	Number of points	CPU time
ode23s	10^{-6}	5451	7.871
ode23s	10^{-9}	40318	44.306
ode15s	10^{-6}	5418	0.772
ode15s	10^{-9}	15795	1.063
ode15s	10^{-12}	15795	5.663
ode23tb	10^{-6}	6138	1.445
ode23tb	10^{-9}	46484	10.559

For every integration, the initial conditions are considered as an equilibrium point, with the initial keplerian elements and $T = D$. The remaining variables are derived from the mathematical model.

In view of the results obtained, ode15s has been chosen setting the options of the integration with a tolerance of 10^{-9} allowing to have an adequate error for a preliminary analysis without a noticeable increment in the computational time required. This is a variable-order solver based on the numerical differentation formulas, it is a multi-step ODE solver used to solve stiff problems.

5 The simulation framework

Once the mathematical and physical model have been described it is time to analyze the simulation framework. Firstly, using the orbital mechanics equations the keplerian parameters are obtained, to this aim, perturbations due to earth oblateness and drag are considered. As so, a drag value will be calculated and computed as an input in the accelerometer block. This block computes the movement of the suspended mass in the accelerometer and gives as an output a voltage value, that is next converted in intensity using a proportional integral controller. The next block is the solenoid control valve, that uses the intensity to move the valve as an input to calculate the displacement of it, and so the mass of xenon that goes into the thruster (following block) can be extracted. Finally, the thrust is obtained and compared to the drag in order to restart the process.

Fig. 3 flow chart helps to better understand the above mentioned.

Limit situations inside accelerometer and valve are also modeled. In order to avoid loss of continuity, limit situations are not corrected inside the integration. Instead, an event is introduced to detect whether accelerometer or valve have surpassed or reached their physical limits. Should the event act, integration is halted, and a consequent simulation starts from the same point, only with initial variables corrected. In this way, no integration variables are modified inside the ODE.

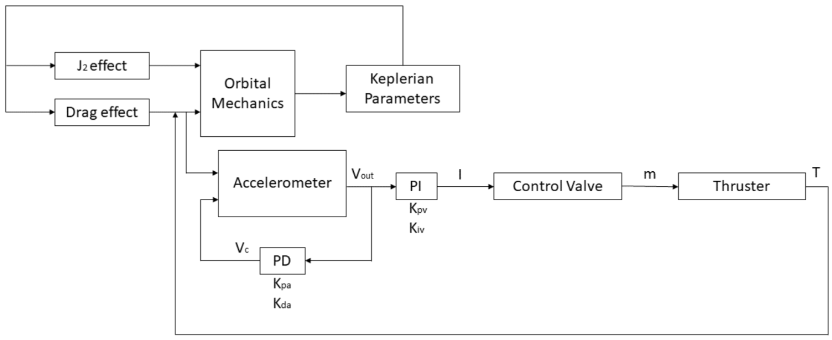
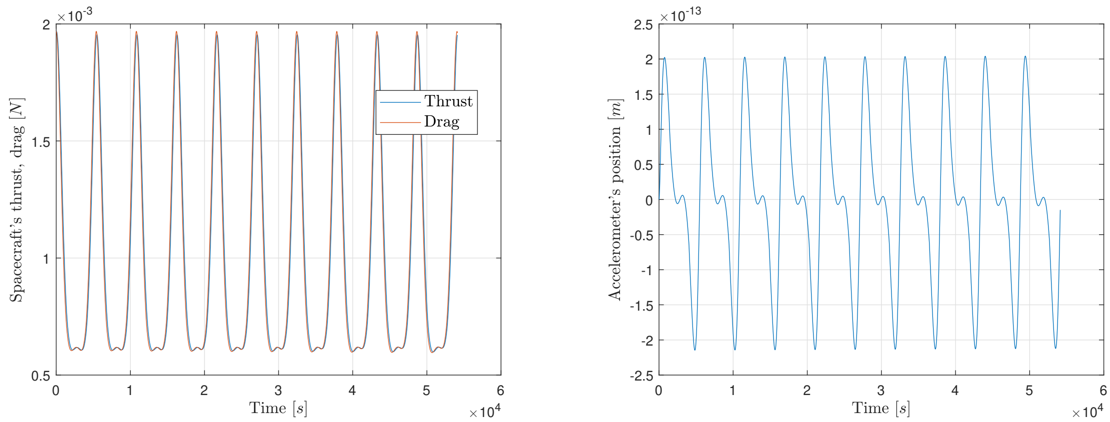


Figure 3: Flow diagram of the simulation

6 System response

In this section we will determine the system response. In addition, we will add two off-nominal conditions to observe how the control system reacts to these particular failures.

NOMINAL CONDITIONS: At this point, we are ready to observe the system response by simply plotting the results of the integration over a number of orbital periods. Before doing so, an analysis about the orbital mechanics implemented has been made by observing if the effects of the perturbations affecting the spacecraft correspond with the expected behaviour. The air drag affects a and e diminishing them. On the other hand, the J2 perturbation causes the well known regression of the nodes and advance of the perigee effects. Indeed, the simulation showed these behaviours, concluding that the orbital mechanic implementation is reasonable. Now we can focus completely in the response of the rest of our system variables. As figure 4a shows, we can conclude that the control system effectively counteracts the deceleration of the spacecraft. Also, figure 4b depicts that the accelerometer mass oscillates around the equilibrium position, thus confirming the effectiveness of the control system.



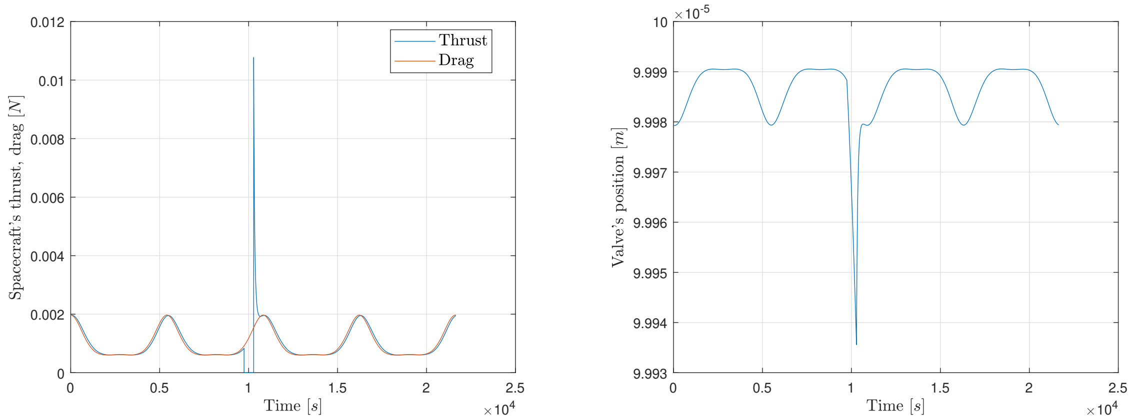
(a) Thrust and drag over time.

(b) Accelerometer's position over time.

Figure 4: System under nominal conditions.

OFF-NOMINAL: To test how the system behaves under off-nominal conditions we have introduced two different cases, a failure in the engine and a failure in the valve system that does not allow the valve to move. When the failures happen we assume that they can be fixed after a certain time, and we study how the system recovers from it.

In the first case, an engine failure, we simply introduce the condition $T = 0$ in our system. At failure time, the engine is not able to provide thrust anymore so the air drag decelerates the spacecraft. To try to compensate it, the system opens a little bit more the valve each time (to increase the $\dot{m}$ and therefore, the thrust). At recovery time, the system can provide thrust again. At this time, the valve is opened enough to compensate all the deceleration suffered with the necessary amount of thrust, so it is able to recover what he had previously lost. This is depicted in figure 5. In figure 5a we can see that the thrust drops to zero when the failure occurs. Note that in figure 5b at failure time the valve starts opening more each time, although no thrust can be provided. Because the valve has been acting during the failure, when the engine starts working again, the resulting thrust is huge. Once the spacecraft reaches again nominal conditions, it stabilizes again.

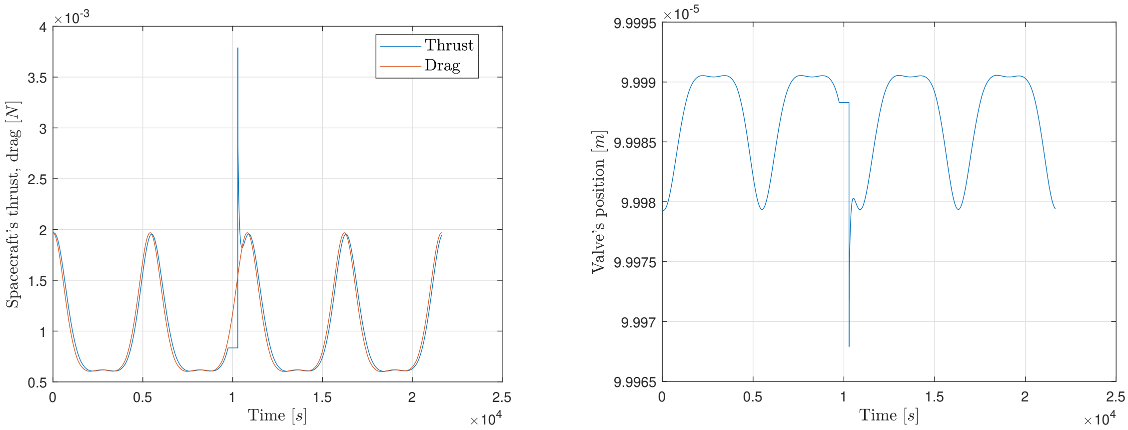


(a) Thrust and drag over time.

(b) Valve's position over time.

Figure 5: System during an engine failure.

For the second case, a valve failure, we have introduced the condition $\frac{dx_v}{dt} = \frac{d^2x_v}{dt^2} = 0$ as the valve is not able to move and remains fixed in position. This means that we are stuck with a constant thrust as long as the failure lasts. Depending on when the failure happens the thrust will have a high or low value, based on its requirements on that moment, so it may be higher or lower than the drag force. If due to the failure the spacecraft has been gaining too much thrust, at recovery point the control system will close the valve as soon as possible until the system recovers the original orbit, and it will open the valve as much as needed if it has been gaining too little thrust compared with the drag suffered. In figure 6 we have represented a case when the valve failure happens in a low air drag resistance. As we can see in figure 6a during the failure the thrust remains constant, as the valve, and therefore $\dot{m}$ are constant (see figure 6b and how in failure time the valve position is constant). During that period, the drag has been increasing each time, so the thrust has not been able to compensate this deceleration. Consequently, the spacecraft losses altitude due to the insufficient thrust. When the failure ends, the control system notices this altitude loss and opens the valve rapidly to increase the thrust and recover the nominal conditions. As soon as it reaches again the original orbit, the valve starts working as usual and the spacecraft stabilizes again. It is worth saying that the immediate control system response is to close more than physically possible (piercing the nozzle) in order to come back to the original orbit even faster, which obviously is not possible. To fix this behaviour, we have introduced an event function in the integrator that stops whenever the valve reaches maximum or minimum stroke, and starts the integration again with the proper boundary conditions.



(a) Thrust and drag over time. **(b)** Valve's position over time.

Figure 6: System during a valve failure.

7 Sensitivity analysis and system optimization

To obtain a more desirable behaviour for the aircraft, an optimization of the system's variables is carried out. Firstly, a sensitivity analysis is performed to understand which of the optimizable variables affect more severely the response of the system and how. Said variables are: K_{p_a}, K_{d_a}, m_{fcv}, K_i, K_{p_v}, K_{i_v}. For each of them, a variation of -50% to $+50\%$ is imposed. The response is then computed for every value - with the rest of the variables kept constant and equal to the initial ones - and a value of the error is obtained. The error is determined as the norm of the difference between thrust and drag. The results are depicted in Fig. 7.

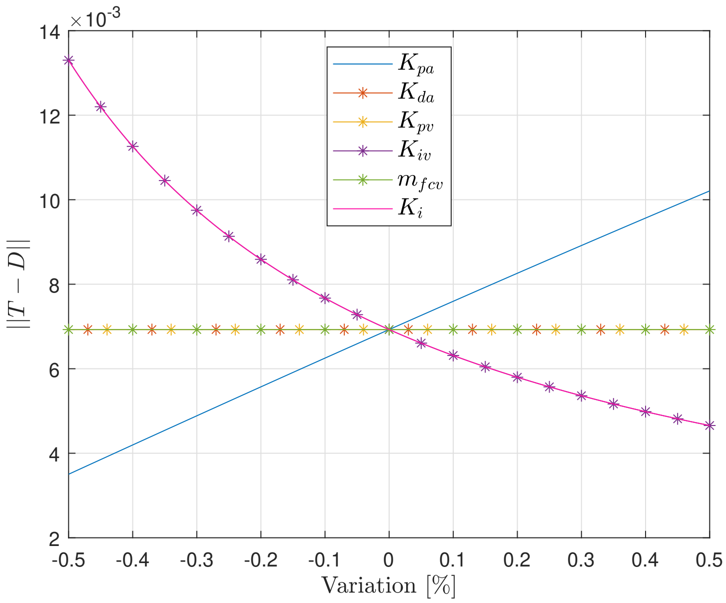


Figure 7: Sensitivity analysis

It can be gathered from the sensitivity analysis that only K_{p_a}, K_i and K_{i_v} significantly modify the response of the system. Therefore, these three variables are the ones to be optimized. The objective function utilised is similar to the one employed for the sensitivity analysis, and is defined in Eq. (18), where $||T-D||_i$ and $||T-D||_0$ denote the norm of the difference between thrust and drag at iteration i and for initial values, respectively.

$$f_{opt} = \frac{||T - D||_i}{||T - D||_0} \tag{18}$$

The optimization is performed via Matlab's `fmincon`, the only constraint being non-negative values. Five periods are simulated. Tolerances for constraint, optimality and step are equal to those employed for integration, that is, 10^{-9}. Two different algorithms are utilised: `fmincon`'s default one ('`interior-point`'), and '`active-set`'. At each iteration, the system is simulated for the modified values, and the objective function determined for the data obtained. The results of the optimization are presented in Table 2.

Table 2: Optimization results.

	K_{p_a}	K_i	K_{i_v}	$\|\|T-D\|\|$
Initial	100000	0.2	3	$2.1903955053\times10^{-3}$
`Interior-point`	999999.99999	2.7297518689	5.0106513122	$9.7539645748\times10^{-5}$
`Active-set`	1000000.0999	2540.3950451	387.07568293	$9.1034899520\times10^{-7}$

Both algorithms succeed in significantly reducing the objective function, but '`active-set`' reaches a value for the norm much smaller than '`interior-point`' accomplishes. However, in order to attain such low orders, values of K_i and K_{i_v} differ from the initial ones much more than those obtained via '`interior-point`'. Fig. 8 represents the aircraft's optimized results for thrust and drag for the '`active-set`' algorithm.

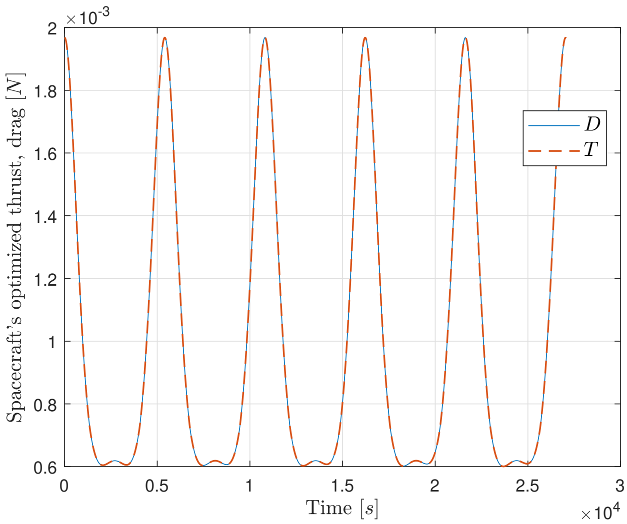


Figure 8: Optimized system's thrust and drag

References

[1] H.D.Curtis. *Orbital Mechanics for Engineering Students*. 2010.

[2] F. Topputo. *Modeling and Simulation of Aerospace Systems* [*Lecture notes*]. 2019.

[3] Canuto, Enrico, et al."Drag free control for the European satellite GOCE. Part I: modelling." *Decision and Control, 2002, Proceedings of the 41st IEEE Conference on.* Vol. 2. IEEE, 2002.

[4] Canuto, Enrico, and Luca Massotti."All-propulsion design of the drag-free and attitude control of the European satellite GOCE." *Acta Astronautica 64.2 (2009):*325-344.

[5] Vaughan, N. D., and J. B. Gamble."The modeling and simulation of a proportional solenoid valve." *Transactions-American Society of Mechanical Engineers Journal of Dynamic Systems Measurement And Control 118 (1996)* : 120-125.

Weights ↓	Fail (<18)	Poor (18-21)	Fair (22-25)	Good (26-28)	Excellent (29-30)
Report (50%)	• Major modelling errors (wrong assumptions, missing domains) • Simulation contains several wrong concepts • Report awfully written	• Some modelling errors (assumptions not complete, missing domains) • Simulation part contains several conceptual mistakes • English is poor	• Fair modelling (good assumptions, physics caught) • Simulation not appropriate (wrong integration scheme) • Report not clear, minor issue (typos)	• Detailed modelling (assumptions detailed and justified, all physics considered) • Minor problems with simulation (numerical issues) • Good English	• Accurate modelling (all physics considered, second order effects quantified/modeled) • Good simulation techniques (integrator, numerical issues) • Good English
Code (30%)	• Code does not run • Major algorithmic errors • Code not complete	• Minor algorithmic errors • Code is not documented • Code takes unnecessary long to run (inefficient)	• Code runs smoothly, fairly documented • Computational efficiency improvable	• Code runs smoothly, well documented • Care is taken to computational efficiency • Add-ons are produced	• Code runs smoothly, well documented • Care is taken to computational efficiency • Valuable add-ons are produced
Presentation (20%)	• Major errors in the presentation • Questions are not answered	• Poor presentation • Poor time management • Poor answers to questions	• Fair presentation • Poor time management • Weak answers to questions	• Good presentation • Good management of time • Answers not exhaustive	• Excellent presentation • Excellent time management • Satisfactory answers given

Figure 9: Rubric used for grading.