Multiphysics Acausal Modeling and Simulation of Satellites Using Modelica Library

Salvatore Borgia[1] Francesco Topputo[1]

[1]Department of Aerospace Science and Technology, Politecnico di Milano, Italy, salvatore.borgia@polimi.it

Abstract

The multiphysics modeling has a great importance when a complex space system (as a satellite) is considered. Indeed, it is necessary to analyse how the system's behavior is affected by the space environment or by on board failures. In this paper, the *Modelica Library* is used to hierarchically build and connect the main subsystems that can be found in a traditional satellite. Specifically, the modeling and simulation of the entire system is carried out in the *Dymola*[1] environment. Finally, the FMI is applied to simulate in Dymola some specific satellite models/logics created with higher fidelity in the Matlab/Simulink[2] domain.

Keywords: Multiphysics modeling, Space system, Modelica library, Dymola, FMI tool

1 Introduction

A space system is generally composed of several subsystems belonging to different physical domains. A malicious entity could compromise even one of them to produce escalation effects involving the whole system. In this situation, the modeling task requires a holistic approach in order to simulate multiphysics interactions that occur inside the system. For this reason, the *Dymola* environment has been used to build a hybrid-complex system from the basic physical elements of the *Modelica library* exploiting the acausal modeling technique (Tiller 2001). Moreover, thanks to Modelica text coding and FMI (Functional Mock-up Interface) tool, ad hoc functions have been created to connect physical variables (simulating their mathematical relation), and to import from other tools (as *Matlab/Simulink*) the acausal model of system's subparts.

1.1 Satellite system

In this paper, a small satellite is considered as an example of space system. It is worth noting that each satellite has its own architecture and on board equipment depending on the specific mission to be accomplished. However, for the purpose of modeling the multyphysics interaction and creating a general simulation platform, a breakdown architecture (Figure 1) has been considered in order to catch the main physical domains characterizing almost any satellite: mechanical, thermal, fluid and electrical. Specifically, the considered subsystems are:

- ADCS (Attitude and Determination Control Subsystem): it allows to change/keep the satellite orientation in space through actuators (as reaction wheels (RWs)), or estimate it using sensors (as gyroscope (Gyro)).
- THR (Thermal subsystem): it allows to monitor the satellite instrumentation temperature or surface temperature keeping it within certain nominal bounds.
- PROP (Propulsion subsystem): it is responsible of delivering thrust to perform/support ADCS attitude maneuver or to counteract disturbance torques (as solar radiation pressure (SRP)).
- EPS (Electrical Power Subsystem): it allows generating power on board, using for example photovoltaic (PV) solar panel, and consequently the Power Management Unit (PMU) manages and distributes it to all subsystems (exploiting Maximum Power Point Tracking (MPPT), battery element, and DC-DC converter (Czarkowski 2011)).

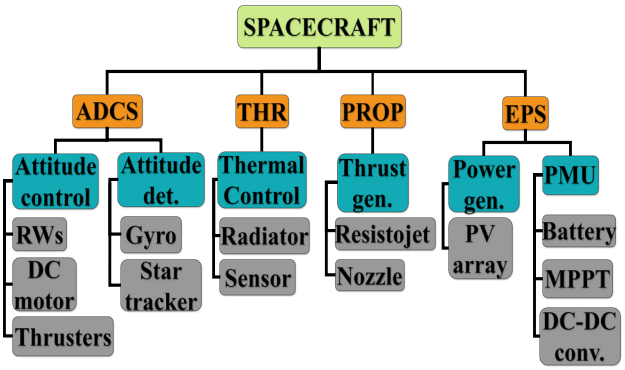


Figure 1. Satellite breakdown structure considered for Dymola modeling.

As it can be noticed in Figure 1: the subsystems of which the satellite is composed are at the first level (orange boxes), the subsystems tasks are reported at the second level (blue boxes), and the physical components (such

[1]https://www.3ds.com/products/catia/dymola

[2]https://it.mathworks.com/products/matlab.html

as actuators or sensors) to perform the tasks are depicted at the third level (grey boxes). The proposed architecture is more likely to be found in CubeSats (Song and al. 2018) due to the choice of resistojet as propulsion solution (Tummala and Dutta 2017). Apart from that, the structure can be then easily adaptable for other cases like large spacecraft. In the following sections, the modeling design of each subsystem and its principle of operation will be presented.

2 Modeling design

The main modeling steps of a general real system can be summarized as (Umez-Eronini-Eronini 1999):

1. Extract a physical model from reality: this process requires engineering judgment to isolate only the physical (state) variables which play a dominant role for the systems behaviour.

2. Extract the mathematical model from the physical one: at this stage, the identified physics phenomena shall be translated into mathematical expressions through the constitutive law equations of the specific physical component.

3. Simulate the mathematical model: the mathematical model of the system then has to be resolved trough the use of integrator scheme that returns the evolution of the state variables in time.

4. Perform sensitivity analysis: from the previous stage, it is possible then to compare the simulation results with the real response matching the behaviour of the digital world with the real one (state identification problem). This procedure allows closing the modeling loop and eventually obtaining a "digital twin" (Singh, Fuenmayor, and al. 2021).

In Figure 2, an example of the modeling process (above described) for the case of a typical electrical direct current (DC) motor which drives a shaft. It can be noticed how the initial complex real system is reduced to a physical representation governed by a set of simple first order differential equations (Cannon 1967).

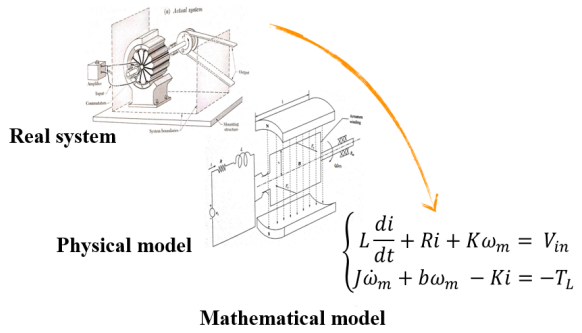


Figure 2. Modeling phases for a DC motor (Cannon 1967).

2.1 Satellite general architecture

After the definition of the high-level structure, the subsystems correlation flow scheme has to be derived to understand how the different physical variables interacts each other within the satellite system. Figure 3 reports the general satellite configuration considered for the modeling part and Figure 4 the relative actuators configuration. Without loss of generality, to simplify the simulation part, the PV arrays are rigidly connected with the main body and the satellite center of mass (CM) is assumed to be located at the origin of the Dymola world frame {**xyz**} where the attitude dynamics equations are expressed. Regarding the thrusters, for modeling simplification, they are assumed to be 6 and, according to which nozzle is activated, three-axis control torques are generated. Specifically, a water resistojet propulsion system is considered for this work. Regarding the star trackers, there are 2 looking at inertial fixed stars respectively along **x** and **y** direction. The satellite characteristic dimensions considered in terms of main body and actuators size, together with the main physical parameters of the system, are similar to the one of a 16U CubeSat (as it will be ramarked in section 5).

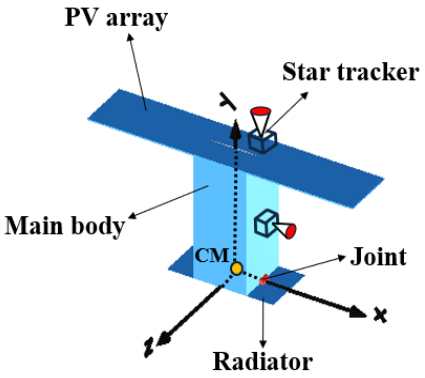


Figure 3. The satellite external configuration.

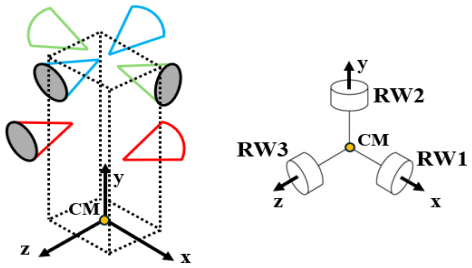


Figure 4. The satellite actuators configuration.

2.1.1 Satellite subsystems interaction

According to the satellite architecture and relative configuration previously presented, the subsystems mutual interaction is derived in Figure 5. From the diagram we can notice that:

 10.3384/ECP207155 DOI

- the EPS bus voltage $\mathbf{V}_{Bus}$, given by the DC-DC converter, or the battery voltage $\mathbf{V}_{Batt}$ feed RWs actuators of the ADCS and the resistojet of the PROP subsystem.

- The attitude matrix $\hat{\mathbf{R}}_{123}$, expressed as euler sequence "123", determines the relative orientation between Sun/Earth and Dymola body-frame. This affects which surfaces are receiving radiation heat or not as boolean vector ν_S. The latter vector is computed by the shadow model knowing the inertial Earth and Sun position ($\hat{\mathbf{r}}_{Earth}$ and $\hat{\mathbf{r}}_{Sun}$, respectively).

- The amount of power generated by the PV array is affected by the PV-Sun incidence angle $\hat{\alpha}_{SA}$.

- The steam obtained after a water mass flow rate $\dot{\mathbf{m}}_{H20}$ is heated up from the electro-thermal circuit of the resistojet, expands through the nozzle generating a level of thrust $\hat{\mathbf{F}}_p$ function of the steam pressure $\mathbf{P}_{st}$ and temperature $\mathbf{T}_{st}$.

- The fuzzy logic regulates the firing time and the thrust direction of each nozzle. In this case, the final output is the relative torque induced on the satellite $\hat{\mathbf{T}}_p$.

- The battery temperature $\mathbf{T}_{Batt}$ and PV array temperature $\mathbf{T}_{SA}$ are inputs for the EPS affecting the level of power generated on board and the battery charge-discharge profile.

- The SRP torque block is able to determine the disturbance torques $\hat{\mathbf{T}}_{srp}$ acting on the satellite knowing both the Sun inertial position and the satellite attitude.

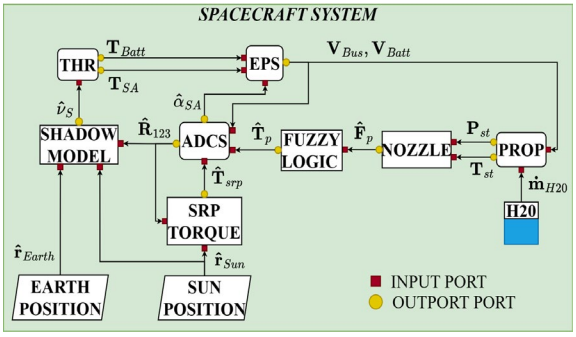


Figure 5. Satellite physical cross-interactions.

The subsystems interaction diagram just described is an example of basic working logic that can be found in many satellite (specifically in CubeSat). Again, each satellite can implement its own architecture slightly changing how the outputs of a subsystem affect the other ones. However, for the purpose of this work, the main multiphysics connections are considered and they shall be taken into account in the modeling part of the satellite physical components. Moreover, during the modeling process, some control logics have been assumed to link variables from a physical world to another. In this way, the space system simulated assumes the characteristics of a typical cyber-physical system.

2.1.2 ADCS subsystem

In this section, the ADCS subsystem is analysed more in details focusing on its working logic flow and on the corresponding modeling translation into Dymola. Figure 6 shows the specific subsystem logic, in particular from left to right we have:

1. The error between the desired satellite attitude angles $\hat{\alpha}^{(123)}_{target}$ (as Euler sequence "123") and the on bard estimation from sensors $\hat{\alpha}^{(123)}_{sat}$ is an input for the PID (Knospe 2006) control block.

2. The PID block returns the continuous signal voltage $\tilde{\mathbf{V}}_m$ to be supplied on the virtual DC motor in order to match the ideal control torque with the one generated by the virtual reaction wheel $\tilde{\mathbf{T}}_{RW}$.

3. According to the virtual RWs angular velocity ω_{RW}, another PID block calculates the duty cycle $\mathbf{D}_c$ of the H-bridge circuit to generate the needed square voltage signal $\mathbf{V}_m$ of amplitude given by the EPS bus voltage $\mathbf{V}_m$.

4. The square signal $\mathbf{V}_m$ supplies the real DC motor block. According to the motor angular rate ω_M, the real reaction wheel follows an angular velocity profile similar to the virtual one. The final effect is the torque released on satellite by the real RWs $\hat{\mathbf{T}}_{RW}$ due to the principle of "action-reaction".

5. Besides the RWs, the SRP torques and the PROP torques drive the attitude dynamics of the satellite assumed to be a simple rigid-body. This latter simplification avoids us to model flexibility which is dominant when large impulsive maneuvers occur on satellite having long solar arrays (Wei, Cao, and al. 2017). Indeed, for a small system, as the one considered in this paper (16U CubeSat), the rigid-body assumption makes more sense because of the short solar panel length and dominant main-body inertia.

As explained in section 2, the mathematical model has to be extracted from the physical world to simulate the behavior along the time. For the examined ADCS subsystem (Figure 6), the main governing equations are:

$$J\dot{\omega}_{sat} = -\omega_{sat} \times (J\omega_{sat}) + \hat{\mathbf{T}}_{srp} + \hat{\mathbf{T}}_p + \hat{\mathbf{T}}_{RW} \quad (1)$$

$$L_a \frac{di}{dt} = -R_a i - K_m \omega_M + \mathbf{V}_m \quad (2)$$

$$J_m \dot{\omega}_M = -b\omega_M + K_m i + \hat{\mathbf{T}}_{RW} \quad (3)$$

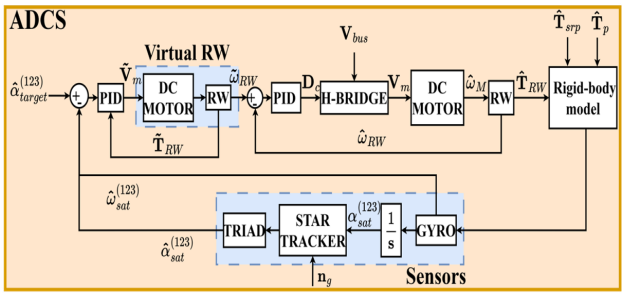


Figure 6. ADCS subsystem modeling scheme.

where in Equation 1: J is the satellite inertia matrix, and ω_{sat} is the angular velocity vector expressed in body-frame axes ({**xyz**} in Figure 3). In Equation 2: L_a and R_a are the inductor and the armature resistance of the DC motor circuit, respectively; i is the current flowing in the circuit; and K_m is the motor constant that allows the coupling of the electric domain with the mechanical one. In fact, in Equation 3, the torque produced by the motor ($K_m i$), together with friction losses ($b\omega_M$) and load torque given by the RW, determines the angular speed of the DC motor with inertia J_m. In this case, the motor shaft is directly connected with the load (RW) without gear-box in between ($\omega_M = \hat{\omega}_{RW}$). In Table 1 it is reported the *Modelica library* components used for modeling the ADCS subsystem in Dymola. The external satellite configuration (Figure 3) has been replicated with the basic rigid body element and then, through links, PV arrays have been added to the main body. The spherical joint is needed to simulate only the three degree of freedom of the satellite. The world frame in Dymola has been then connected to the spherical joint in correspondence of the satellite CM, and the gravity field option has been imposed to zero to replicate the deep space condition. The radiator mechanism is allowed to rotate only along the **z**-axis of Dymola frame thanks to the revolute joint element. Assuming an ideal gyro, it is simple an angular velocity sensor in Dymola. Regarding the DC motor speed controlled by the H-bridge electric circuit (Priyanka and Mariyammal 2018), the homonymous components already existing in the *Modelica library* have been exploited instead, for the RW, it has been modeled as inertia load element attached to the DC motor shaft. The Pulse Width Modulation (PWM) block in Modelica allows to generate the real voltage signal with discrete values: $[-\mathbf{V}_{Bus}, +\mathbf{V}_{Bus}]$ and a variable duty cycle given by a limited [0 to 1] PID. The fixed rotation/translation element, listed in the table, permits to link the main body with the PV arrays or, eventually, to arrange the satellite rigid body elements into another configuration.

In section 3, the Modelica text coding (within Dymola) will be discussed in order to simulate: the TRIAD algorithm, for estimating the attitude matrix exploiting the two measured vectors by star trackers, the Gaussian noise $\mathbf{n}_g$ model of the star tracker, and the SRP torque computation in Dymola body-fixed frame axes.

Table 1. Modelica structure of the ADCS subsystem

Physical element	*Modelica component*
Rigid-body	
World frame joint	
World torque	
Fixed rotation	
Radiators joint	
Voltage source	
DC motor	
RW	
H-Bridge	
PWM	
GYRO	
PID (limited)	

2.1.3 THR subsystem

The thermal model has been tackled using the lumped approach (Cannon 1967). According to this method, the satellite surfaces or internal instruments (like the battery) are modeled as nodes with a certain thermal capacity (depending on the material). The thermal nodes are capable to exchange heat each other, with the Sun/Earth, and with the deep space. In Figure 7 it is reported the lumped scheme of the satellite system presented in 2.1. In fact, the lumped nodes (3-4-5-6-7-8) represent the surfaces of the main body, the PV array surfaces correspond to the nodes (1-2-9-10) instead the radiators surfaces correspond to the nodes (11-12-13-14). As it can be noticed, the dominant heat exchange ways considered are conduction and radiation (the convection is not present or negligible in deep space due to lack of air). The deep space acts as a sink

10.3384/ECP207155 DOI

(average temperature of 3 [K]) with which each node exchanges heat by radiation. The PV array surfaces have been split in four nodes to model the different thermal properties between the top side (higher emissivity) and the bottom one (higher absorptivity). Similarly for the radiators with the difference of having higher emissivity at nodes (12-14) and higher absorptivity at nodes (11-13).

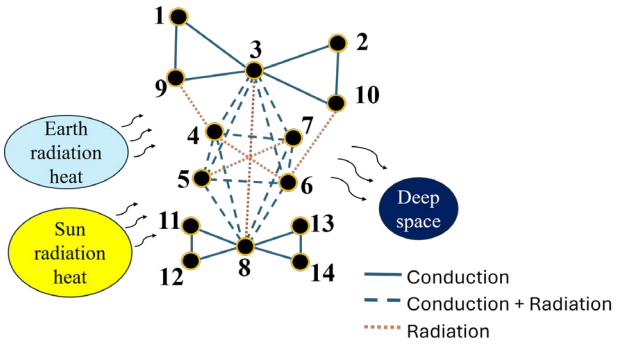


Figure 7. Satellite thermal lumped model.

The THR subsystem logic flow scheme is shown in Figure 8. Specifically, we can observe that:

1. The subsystem receives as external inputs: the satellite attitude matrix from ADCS, the sun position vector with respect to an inertial frame, and the solar irradiance $\mathbf{E}_{ir}$ with intensity depending on the satellite to Sun distance.

2. The shadow model returns the boolean vector ν_S with dimension equals to the number of thermal nodes (14 in this case). If the value is 1 means that the corresponding node is receiving the Sun or Earth radiation.

3. From the resulting satellite temperature distribution, hysteresis logic can be adopted to maintain the average nodes temperature between a desired range. The hysteresis logic will then open/close the radiators surface through the control torque $\hat{\mathbf{T}}_c$.

4. The opening of radiator will affect the heat transfer coefficients of the thermal model both for the conduction $\mathbf{G}_c$ and radiation $\mathbf{G}_r$.

5. Finally, from the radiator rotation angle $\hat{\alpha}_{radiator}$ along $\mathbf{z}$-axis of the body frame (Figure 3), the shadow condition for nodes (11-12-13-14) can be established.

Regarding the mathematical model, the main governing equations of the THR subsystem can be summarized as (Lienhard 2024):

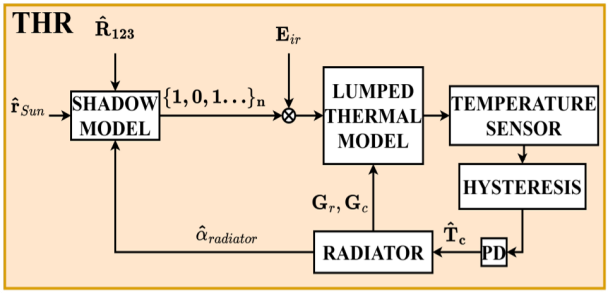


Figure 8. THR subsystem modeling scheme.

$$C_i \frac{dT_i}{dt} = \dot{Q}^i_{Sun} + \dot{Q}^i_{Earth} + \sum_{ij} \dot{Q}^{ij}_c + \sum_{ij} \dot{Q}^{ij}_r \tag{4}$$

$$\dot{Q}^i_{Sun} = \alpha_i \dot{q}_{Sun} A_i \tag{5}$$

$$\dot{Q}^i_{Earth} = \alpha_{al} \dot{q}_{al} A_i + \alpha_{ir} \dot{q}_{ir} A_i \tag{6}$$

$$\dot{Q}^{ij}_c = G^{ij}_c (T_i - T_j) \tag{7}$$

$$\dot{Q}^{ij}_r = G^{ij}_r (T_i - T_j) \tag{8}$$

$$G^{ij}_c = \frac{\lambda^{ij}_c A^{ij}_c}{s^{ij}_c} \tag{9}$$

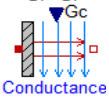
$$G^{ij}_r = \frac{\sigma_B (T_i + T_j)(T_i^2 + T_j^2)}{\frac{1-\varepsilon_i}{\varepsilon_i A_i} + \frac{1}{A_i F_{ij}} + \frac{1-\varepsilon_j}{\varepsilon_j A_j}} \tag{10}$$

where in Equation 4: T_i is the temperature of the $i-th$ node, C_i is the thermal mass capacity of the $i-th$ node, $\dot{Q}^i_{Sun}$ and $\dot{Q}^i_{Earth}$ are the Sun and Earth radiation energy falling into $i-th$ node, respectively. The Sun energy is computed through Equation 5 knowing the node absorptivity α_i, the Sun heat flux $\dot{q}_{Sun}$, and the node area A_i; instead the Earth energy is calculated using Equation 6 knowing the albedo absorptivity α_{al}, the infrared absorptivity α_{ir}, the albedo irradiance $\dot{q}_{al}$, and the infrared irradiance $\dot{q}_{ir}$ perceived on s/c. In Equation 4: $\dot{Q}^{ij}_c$ is the conduction energy exchange between the $i-th$ and $j-th$ node, and $\dot{Q}^{ij}_r$ is the radiation energy exchange between the $i-th$ and $j-th$ node. The conduction energy is governed by Equation 7 in which the thermal conductance G^{ij}_c has to be determined assuming a contact area A^{ij}_c between the nodes, a wall thickness s^{ij}_c, and the material thermal conductivity λ^{ij}_c (Equation 9). Regarding the radiation energy, it is calculated using Equation 8 knowing the radiation conductance G^{ij}_r. This latter term can be estimated through Equation 10 where: σ_B is the Boltzmann constant, $\varepsilon_{i,j}$ is the node emissivity, and F_{ij} is the radiative view factor between the $i-th$ and $j-th$ node (which can be obtained analytically applying the geometrical equations in (Martínez 2015)).

In Table 2, the Dymola structure of the just described THR subsystem is reported. Particularly, it can be noticed how the main physical variables of the mathematical model and logic in Figure 8 are acausally modeled by

a specific *Modelica library* component. In section 3, the shadow self-built function will be discussed in details.

Table 2. Modelica structure of the THR subsystem

Physical element	*Modelica component*
Thermal node	
Thermal conductor	
Radiation heat transfer	
Heat transfer coefficient	
Sun (Earth) radiation	
Hysteresis control	
Temperature sensor	
Deep space sink	

2.1.4 PROP subsystem

The PROP subsystem logic, for the considered satellite architecture in Figure 1, is shown in Figure 9. From left to right we have:

1. the target error, in terms of angle θ and angular rate error $\dot{\theta}$, are inputs for the fuzzy logic block which returns the firing time of each satellite nozzle (Figure 4).

2. The EPS bus voltage is decreased to a suitable value (~ 0.5 [V]) to be applied at the extremes of the tungsten rod. The voltage conversion is done thanks to a buck circuit (Czarkowski 2011). The final heat flow $\mathbf{Q}_{flow}$ released by tungsten, as result of Joule effect, feeds the boiler's furnace.

3. Inside the vaporizer, the liquid water flow $\dot{\mathbf{m}}_{H2O}$ turns into vapour phase expanding towards the convergent-divergent nozzle. The amount of liquid phase in the boiler is controlled by a PID that maintains the furnace on.

4. From the nozzle moment arm $\mathbf{L}_{arm}$ with respect to CM, the propulsive torques $\hat{\mathbf{T}}prop$, due to the exhausted steam mass flow, are calculated.

5. The valve element allows the steam generation and consequently the thrust delivery on satellite. It can be opened whenever an attitude maneuver with the PROP subsystem shall be performed.

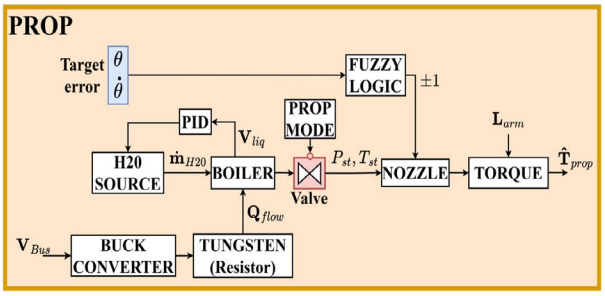


Figure 9. PROP subsystem modeling scheme.

In this case, the boiler is an in-built acausal Modelica component which implements the drum-boiler dynamics (Åström and Bell 2000). The other main equations of the mathematical model involve the nozzle, and the fuzzy logic block. The first block computes the thrust according to the rocket equations (Sutton and Biblarz 2017) (assuming the steam as an ideal gas):

$$\hat{\mathbf{F}}_p = \dot{m}_{st} V_e + p_e A_e \tag{11}$$

$$V_e = M_e \sqrt{\gamma R T_e} \tag{12}$$

$$\frac{p_e}{P_{st}} = \left(1 + \frac{\gamma - 1}{2} M_e^2\right)^{-1} \tag{13}$$

$$\frac{T_e}{T_{st}} = \left(1 + \frac{\gamma - 1}{2} M_e^2\right)^{-\frac{\gamma}{\gamma - 1}} \tag{14}$$

$$\frac{A_e}{A^*} = \left(\frac{\gamma + 1}{2}\right)^{-\frac{\gamma+1}{2(\gamma-1)}} \left[\frac{\left(1 + \frac{\gamma-1}{2} M_e^2\right)^{\frac{\gamma+1}{2(\gamma-1)}}}{M_e}\right] \tag{15}$$

where in Equation 11: $\dot{m}_{st}$ is the steam mass flow rate produced in the boiler, V_e is the steam exhaust velocity, and p_e is the pressure at the exit area A_e of the nozzle. Equation 12 allows to estimate the exhaust velocity from the exit Mach number M_e, the heat capacity ratio γ, the steam gas constant R, and the temperature condition T_e at exit nozzle area. The Mach number can be iteratively calculated, applying for example the Newton method (Galántai 2000), once the ratio between the nozzle throat area A^* and the exit one is fixed (Equation 15). The missing quantities of steam pressure and temperature conditions at the exit can be determined respectively with Equation 13 and Equation 14. The fuzzy logic has been designed using the Mamdani GUI interface in Matlab (as it will be explained in section 4). For the fuzzy rules, they have been chosen following the procedure in (Nagi, Ahmed, and al. 2009). Figure 10 shows the membership functions considered for the design of the fuzzy bang-bang controller.

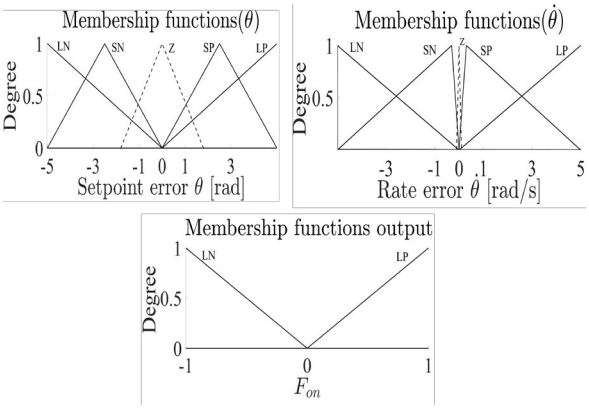


Figure 10. Fuzzy memebership functions used in Mamdani model scheme.

In Table 3, the PROP acausal version in Dymola is shown. Notice how the resistor component has the thermal port enabled to simulate the dissipation heat $\mathbf{Q}_{flow}$ released by the tungsten mass. Moreover, the heat transfer component is added to restore the nominal boiler conditions (tungsten temperature ~ 298 [K]) just after the PROP is turned off.

Finally, the nozzle I/O (Input/Output) block in Dymola will be presented in section 3 where the Modelica text coding has been applied to connect this element with the others Modelica PROP components listed in Table 3.

2.1.5 EPS subsysyem

The EPS subsystem is now analysed to understand the logic architecture assumed for this work. In Figure 11 it is reported the subsystem operation:

1. The external inputs of: the Sun heat flux over the PV array, the Sun rays incidence angle on the PV surface, and the solar panel temperature are used to estimate the photocurrent flowing in the PV array circuit.

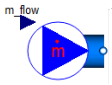
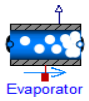
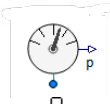
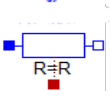
2. The MPPT algorithm regulates the PV voltage to operate at the maximum power possible. Then, the PV voltage is increased or decreased, using a two-switch buck-boost converter (Kim and al. 2022), to obtain the operative bus voltage (for 16U CubeSat assumed to be ~ 13 [V]).

3. The battery is fed nominally on bus voltage and, according to its state-of-charge (SOC) or eclipse condition ($V_{Bus} \sim 0$ [V]), the BMS (Battery Management System) unit regulates the current i_{Batt} flowing in it.

4. When an eclipse occurs or, in general, the bus voltage is too low, the switch is closed such that the battery can supply power to the loads.

5. The charge-discharge battery profile is affected by the temperature of the node associated to the battery T_{Batt}.

Table 3. Modelica structure of the PROP subsystem

Physical element	*Modelica component*
Water source	
Steam drum boiler	
Steam sink boundary	
Valve	
Steam pressure sensor	
Steam temperature sensor	
Voltage source	
Thermal resistor	
Tungsten mass	
Heat transfer	

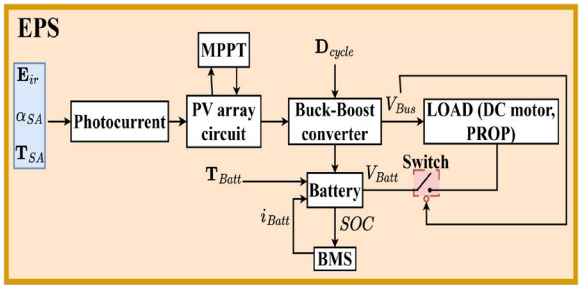


Figure 11. EPS subsystem modeling scheme.

Regarding the MPPT algorithm, Figure 12 shows the logic scheme of the applied method to make PV array operate at the maximum extractable power for different photocurrent values. Instead, the BMS applies the typical Constant Current-Constant Voltage (CC-CV) algorithm (Mostacciuolo, Iannelli, and al. 2018) during the battery charging-discharging process.

In Figure 13, the Buck-Boost converter physical model, considered for the Dymola simulation, is reported. It is worth noticing that, in the context of acausal modeling approach, only the physical representation of the real system

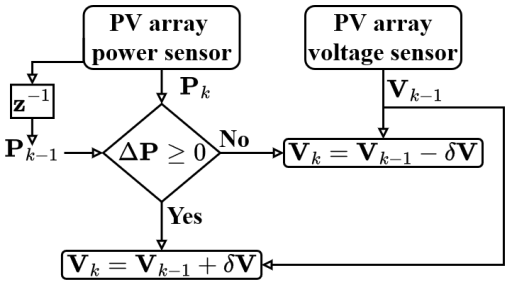


Figure 12. MPPT algorithm scheme.

has to be extracted, while the constitutive equations are already embedded in each singular component of the *Modelica library*. In this way, the modeling focus is moved more on the physical concept rather than on the solving procedure of the equations (Kulhánek and al. 2015). The mathematical model of the EPS subsystem can be summarized by the following main constitutive equations:

$$\mathbf{I}_{diode} = \mathbf{I}_{ds}\left[\exp\left(\frac{v_d}{N v_t}\right) - 1\right] \tag{16}$$

$$v_t = \frac{\sigma_B T_d}{q} \tag{17}$$

$$\mathbf{V}_{out} = \frac{d_1}{1 - d_2}\mathbf{V}_{in} \tag{18}$$

where in Equation 16: $\mathbf{I}_{diode}$ is the current flowing into a single diode, $\mathbf{I}_{ds}$ is the diode saturation current, v_d is the diode voltage drop, N is the diode emission coefficient, and v_t is the diode thermal voltage. This latter quantity can be retrieved using Equation 17 knowing the Boltzman constant, the diode (or solar array) temperature T_d, and the electron charge q.

In Equation 18, the voltage conversion of the two switch buck-boost converter as function of: the duty cycles d_1 and d_2 of the two switches (S_1 and S_2, respectively), and the input circuit voltage $\mathbf{V}_{in}$.

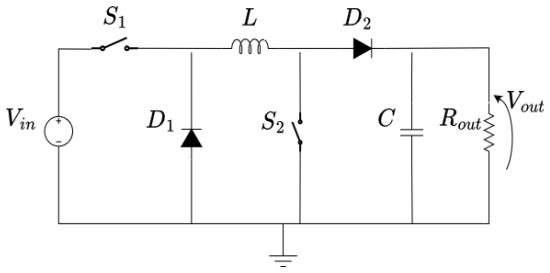
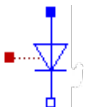
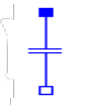


Figure 13. Buck-Boost converter electric circuit.

As it has been done with the previous satellite subsystems, the Dymola EPS translation is listed in Table 4. Specifically, the series/parallel connection of a singular thermal diode element in Modelica allows to build the entire PV array circuit with temperature as inputs. For the MPPT logic, the `Logical` library has been exploited. Regarding the BMS unit, in section 4 it will be discussed the modeling exportation from the Matlab/Simulink to Dymola environment using the *FMI* interface.

Table 4. Modelica structure of the EPS subsystem

Physical element	*Modelica component*
PV array diode	
Photo-current	
Resistor	
Capacitor	
Inductor	
Switch (BB)	
Diode (BB)	
Power sensor	
Ground	

3 Modelica coding

In this section, the generation method of the ad hoc functions presented so far is discussed. For the analyzed satellite system, they can be summarized in:

- SRP torque
- ADCS sensor
- Shadow model
- Nozzle
- Photocurrent

These functions have dual objectives: modeling all that specific algorithms or components to which a corresponding acausal translation can not be found in the *Modelica Standard Library*, and connecting a s/c subsystem to another or to the space environment (like SRP torque function). The functions generation has been performed using

the `Text` option in Dymola starting from the basic layout in the `Blocks` library (*Modelica® - A Unified Object-Oriented Language for Systems Modeling* 2014). This option allows to design the block (I/O) ports and to write inside of it the algorithm which relates the input with the output. Specifically, the main algorithms/models used are:

1. The SRP torque generated by each satellite surface, with respect to CM (Figure 3), depending on its absorption, diffuse reflection, and specular reflection coefficients (Wertz 1978).
2. The perturbing matrix as Euler sequence "123", with a rotation angle function of the gaussian noise $\mathbf{n}_g$, to determine the vectors measured by the star trackers in body-frame axes.
3. The TRIAD algorithm (Markley 1999) to estimate the s/c attitude angles as euler sequence "123" (alphaX_est, alphaY_est, and alphaZ_est).
4. The determination of the shadow condition for each lumped node (Figure 7) checking the value of the scalar product between the Sun direction $\hat{r}_{Sun}$ and the normal of each s/c surface.
5. The converging-diverging isentropic flow expansion (Sutton and Biblarz 2017).
6. The PV photocurrent mathematical model (X. H. Nguyen and M. P. Nguyen 2015).

Figure 14 shows the corresponding Dymola layout of the functions above described (highlighting the variables discussed in the previous sections). In Listing 1 it is reported an example of the Modelica text structure for the *Photocurrent block*.

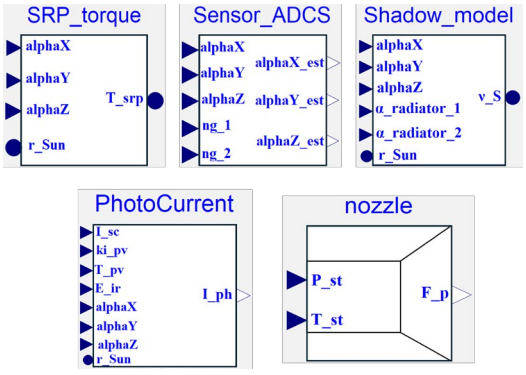


Figure 14. Dymola block functions architecture.

Listing 1. Modelica text of Photocurrent block function

```
block PhotoCurrent
  extends Modelica.Blocks.Icons.Block;
  Modelica.Blocks.Interfaces.RealInput I_sc
      ;
  Modelica.Blocks.Interfaces.RealInput
      ki_pv;
  Modelica.Blocks.Interfaces.RealInput T_SA
      ;
  Modelica.Blocks.Interfaces.RealInput E_ir
      ;
  Modelica.Blocks.Interfaces.
      RealVectorInput y_body[3];
  Modelica.Blocks.Interfaces.
      RealVectorInput r_Sun[3];
  Modelica.Blocks.Interfaces.RealOutput
      I_ph;
protected
    Real cosine_alpha_SA;
    Real nu_el;
    Real cosine_alpha_SA;
    Real F_sensitivity;
    Real I_ir_SA;

algorithm

cosine_alpha_SA := y_body[1]*r_Sun[1] +
    y_body[2]*r_Sun[2] + y_body[3]*r_Sun
    [3];

if cosine_alpha_SA < 0 then
  cosine_alpha_SA :=0;
end if;

I_ir_SA :=E_ir*nu_el*F_sensitivity*
    cosine_alpha_SA;

I_ph :=(I_sc + ki_pv*(T_SA - 298.15))*(
    I_ir_SA/1000);
end PhotoCurrent;
```

4 FMI interface

The Functional Mock-up Interface (FMI) is a powerful tool when the modeling of complex hybrid system (as a space system) is requested. Indeed, this interface allows: the interaction between different programming language (making the model more versatile), and the use of the potentialities coming from each modeling platforms. For this work, the FMI has been used for:

- The Fuzzy bang-bang logic (Figure 15).
- The Battery-BMS module (Figure 16).

Each of them has been acausally modeled exploiting the `Simscape library` within the Matlab/Simulink environment. The choice of changing the modeling language has been made for two reasons: speeding up the implementation using the fuzzy logic designer app in Matlab/Simulink, and taking advantage of the medium-high fidelity model of the battery element in *Simscape* (especially for the temperature dependency effects).

5 Simulation results

Once the hierarchical model of the satellite system is done, with the relative differential equations implicitly defined for each physical component, the next step is to simulate the system (as described in section 2). The tab

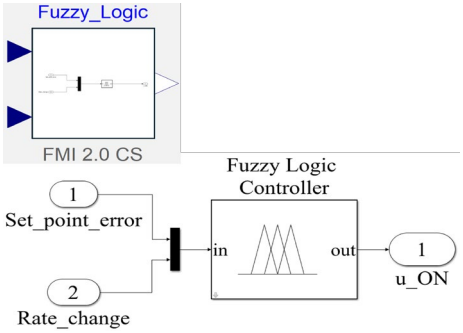


Figure 15. The Dymola FMI block (top) and the relative Simulink model (bottom) of the Fuzzy logic.

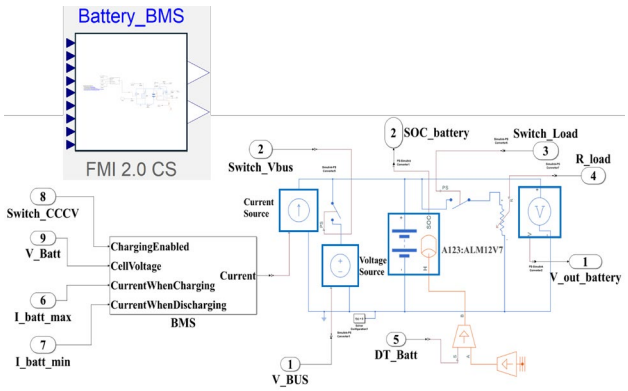


Figure 16. The Dymola FMI block (top) and the relative Simulink model (bottom) of the Battery-BMS module.

`Simulation` in Dymola allows to integrate the entire satellite mathematical model selecting a proper integration scheme. Below are the main assumptions and parameters selected to perform the simulation:

- The satellite is a 16U CubeSat with dimension: $0.2 \times 0.2 \times 0.4$ [m].
- The PV array dimension are: $0.2 \times 0.4 \times 0.01$ [m].
- The DC motor nominal voltage is ~ 12 [V].
- The RWs inertia is ~ 0.0005 [kg m^2].
- The peak power generated on board is ~ 180 [W].
- The maximum battery voltage is ~ 15 [V].
- The star tracker covariance is $1\sigma \sim 10^{-4}$ [rad].
- The Mach number M_e at each nozzle exit area is ~ 10.
- The satellite is in a heliocentric orbit receiving a constant Sun heat flux of 1370 [W/m^2].
- Only the Sun is considered as heat source for the thermal model (no Earth contribution).
- The simulation time is ~ 100 [s] in which the satellite displacement is neglected considering the inertial Sun direction fixed.
- No orbit propagation but only satellite attitude evolution in time.
- The aluminium is assumed as the satellite main body material and silicon one for the PV surfaces.
- The wall thickness considered for the thermal conduction is ~ 0.01 [m].

In the following sections, the results of a singular satellite subsystem and a scenario embedding more subsystems will be analysed. Due to the high number of physical variables involved, in section 6 it has been reported the satellite animation (obtained in Dymola) together with the evolution of the main subsystems variables. Regarding the integrator, to solve the mathematical model, the `Dassl` scheme has been used for all the simulations except for the last one (section 5) where the stiff scheme `Sdirk34hw` has been proved to be more performing.

5.1 ADCS-Sensors simulation

The ADCS subsystem is now simulated in Dymola according to the scheme presented in Figure 6. The simulation assumes to have the Sun direction fixed at $[0,\ 0,\ 1]$ and the target ramp profiles shown in Figure 23**b** to be tracked by the satellite (or spacecraft (s/c)). In Figure 17 it can be visualized the comparison between the ideal gyro measurement integration and the TRIAD algorithm output along **x**-axis body frame. Obviously, the estimated attitude (black) differs from the real one (red) unless of the Gaussian noise $\mathbf{n}_g$. The peak overshoot attitude error (Figure 23**e**) obtained during the maneuver is $\sim 0.34°$ (which can be modified tuning the PID parameters). The real voltage square signal given to the DC motor of the RW1 is depicted in Figure 23**f**. In this simulation, the PWM block switching frequency is set at 1000 [Hz]. In Figure 23**a** the satellite attitude animation, during the maneuver, is extracted thanks to the `Animation` tool in Dymola.

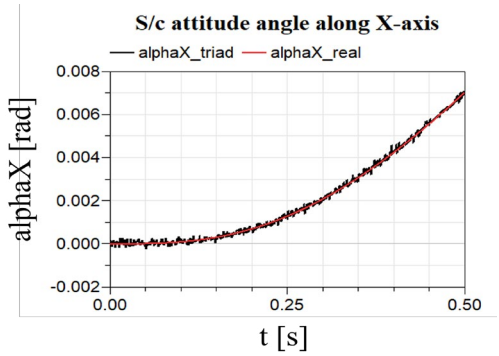


Figure 17. Satellite attitude angle comparison between the real and the estimated one by the *Sensor_ADCS* block.

10.3384/ECP207155 DOI

5.2 Fuzzy logic (FMI simulation)

The PROP architecture described in Figure 9 is now tested in Dymola. As we noticed from the previous ADCS simulation (Figure 17), the TRIAD estimation doesn't affect the PID control which is able to bring anyway the satellite at the desired attitude. So, from now on the ideal attitude angles from gyro will be considered in place of star trackers measurements. Also in this case, the Sun direction is fixed at $[0,\ 0,\ 1]$ and the final desired attitude sequence "123" is set to $\hat{\alpha}_{target} = [0.3,\ 0.8,\ 0.1]$ [rad]. Figure 18 shows the simulation results. It can be noticed how the PROP system is able to maneuver the satellite reaching a final attitude error (Figure 18**d**) of $\sim 0.06°$.

The Fuzzy FMI block outputs are reported in Figure 19 in terms of propulsive torques, expressed in Dymola *body fixed-frame*, and the corresponding firing time/direction associated to each satellite nozzle (Figure 4).

As expected from the exit vaporizer conditions in Figure 20, the resistojet gives a torque intensity in the order of 10 [mNm] with a nominal steam pressure $P_{st} = 13$ [bar] and a water mass flow rate, to maintain the vaporizer on, of $\dot{\mathbf{m}}_{H2O} \sim 0.13$ [kg/s]. It is worth highlighting that the mass of water storable inside the satellite is constrained by the dimensions of the system itself so, according to the mission design, the PROP subsystem parameters have to be refined accordingly.

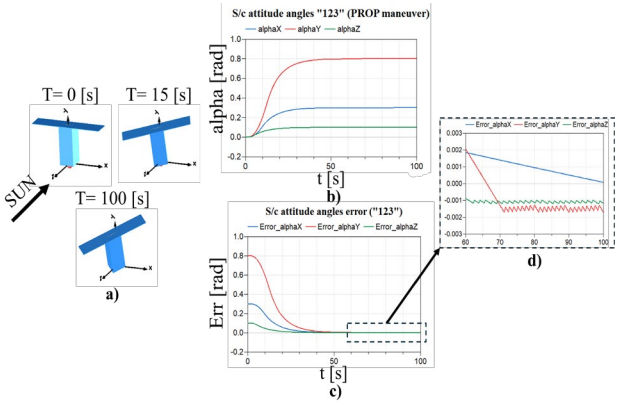


Figure 18. Fuzzy FMI block outcome: **a)** Satellite attitude animation during the propulsive maneuver; **b)** Satellite attitude Euler angles $\hat{\alpha}_{sat}^{(123)}$; **c)** Error profile between the target attitude angles and the real ones; **d)** Zoom in of the attitude error in the time interval [60-100] [s].

5.3 Battery-BMS (FMI simulation)

In this section, the Battery-BMS FMI block presented in section 4 is simulated separately to visualize its behavior in time. For this simulation, the simulink model has been augmented of a load resistance of 0.1 [Ω] to replicate the power absorption event when the battery feeds the loads. Regarding the satellite bus voltage signal, it has been imposed to be squared to simulate periodical eclipse effects. Figure 21 shows the overall response of the battery. Specifically, when $\mathbf{V}_{Bus}$ is zero, the load switch is activated with the effect of a drop battery voltage and consequently discharge. Then, a sensitivity analysis has been done for three different battery temperature: 283 [K], 293 [K], and 313 [K]. In Figure 22**a** it can be noticed how the higher battery temperature accelerated the discharge time (red SOC curve) against the case of a lower temperature (green SOC curve). For the battery voltage instead, it can be caught the effects of an increment of the battery charging voltage and higher loaded drop voltage at lower temperature (Figure 22**b**).

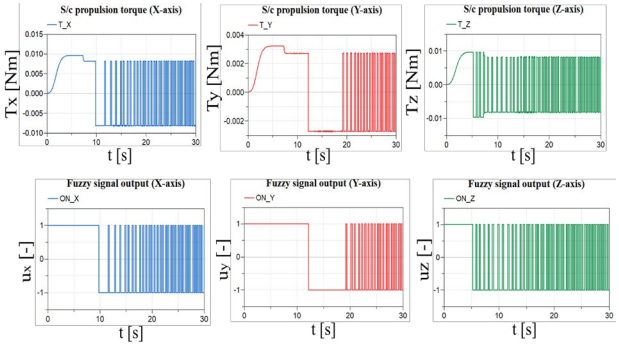


Figure 19. Fuzzy logic FMI outputs: at the top the propulsive torques profile $\hat{\mathbf{T}}_p$ in the Dymola *body fixed-frame*, and at the bottom the firing time/direction for each nozzle in the time interval [0-30] [s].

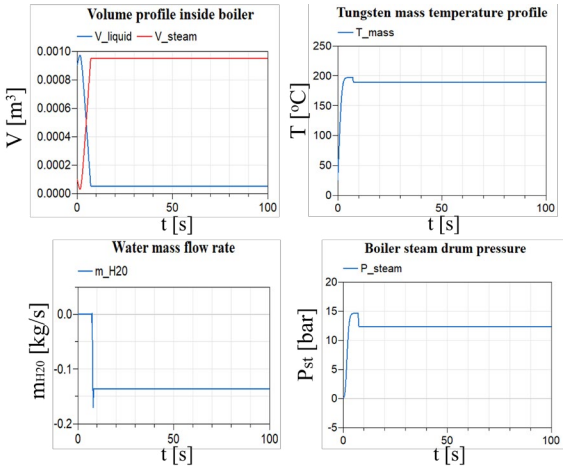


Figure 20. Vaporizer state variables evolution with time during the PROP maneuver.

5.4 THR-Radiator simulation

The THR subsystem, with the radiators active control architecture shown in Figure 8, is simulated here in Dymola considering: the inertial Sun direction at $[0,\ -1,\ 0]$ and then, after ~ 6.9 [h], it changes at $[0,\ 0,\ 1]$; the control is based on checking the temperature of node 8 and trying to maintain it below 70 [°C]. Moreover, when the radiators are closed, they don't exchange heat at all keeping constant their temperature. The simulation results are re-

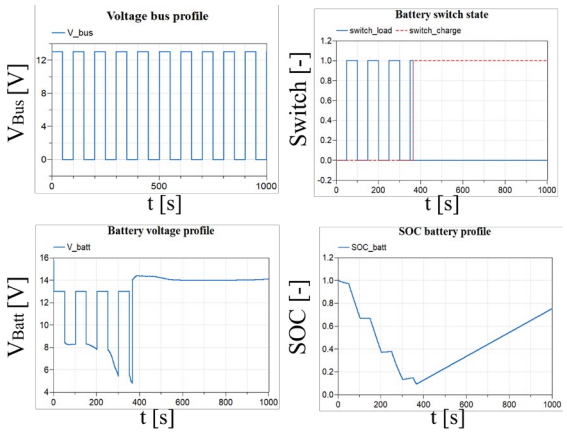


Figure 21. Battery-BMS simulation outputs.

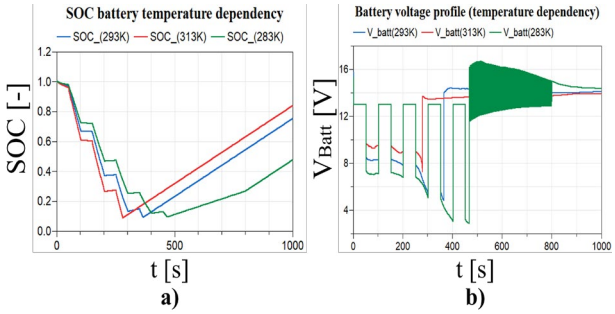


Figure 22. Battery response for different temperature conditions.

ported in Figure 24. Specifically, it can be noticed how the radiators are opened when T_8 reaches 60 [°C](Figure 24**c**), and they are closed again when T_8 goes below 40 [°C] to avoid overcooling of the system. It is worth remarking that due to the small satellite dimensions (16U CubeSat), the symmetry of the thermal scheme (Figure 7), and the homogeneity of the material used, the temperature difference among the nodes is small (as it can be seen in Figure 24**d** and in Figure 24**g**). Finally, Figure 24**e** shows the increment of the heat radiation exchange to deep space when high emissivity nodes (8-12-14) are active.

5.5 Multyphysics scenario

The last simulation involves all the satellite subsystems except for PROP (the attitude maneuver is performed only by RWs). The analysed multiphysics scenario is the following: the satellite (or s/c) shall perform a rotation of 180° around the fixed Dymola **x**-axis; the Sun direction is fixed at $[0,\ 1,\ 0]$. In Figure 25 the overall system response is reported. In particular, it can be noticed how the power generation is higher at the beginning of the maneuver (Figure 25**b**) and then zero when the nodes (1-2) are in shadow (Figure 26**a**). In this latter case, the battery feeds the RWs by discharging (Figure 25**d**) and delivering them a voltage ~ 13 [V] (Figure 25**g**). For this simulation, the PWM signal frequency has been set to 100 [Hz].

In Figure 26**b**, the MPPT logics allows to make the main bus operate at the nominal value ~ 13 [V] despite the PV array is working at higher voltage. Finally, Figure 26**c** shows the real-time satellite attitude angle trajectory controlled by the PID, and the corresponding angular velocity of the RW1 in Figure 26**d**.

6 Conclusions

The main objectives of this work were: modeling and simulating an example of complex space system using the *Modelica* tools; creating a platform where different tests and failure analysis could be carried out. The future work will be to expand the satellite model including also the structure subsystem to study the flexibility effects (especially for large satellites). Lastly, the FMI option will help on exporting the models into others programs to perform parametric system identification (Gupta and al. 2018) or nonlinear system identification in frequency domain (Pintelon and Schoukens 2012), using the real system telemetry, and to obtain higher fidelity models.

References

Åström, K.J. and R.D. Bell (2000). "Drum-boiler dynamics". In: *Automatica* 36.3, pp. 363–378. DOI: 10.1016/S0005-1098(99)00171-5.

Cannon, Robert H. (1967). *Dynamics of physical systems*. McGraw-Hill. ISBN: 0486428656.

Czarkowski, Dariusz (2011). *Power Electronics Handbook (Third Edition)*. Butterworth-Heinemann, pp. 249–263. DOI: 10.1016/B978-0-12-382036-5.00013-6.

Galántai, A. (2000). "The theory of Newton's method". In: *Journal of Computational and Applied Mathematics* 124.1-2, pp. 25–44. DOI: 10.1016/S0377-0427(00)00435-0.

Gupta, Sapna and et al. (2018). "Parametric system identification and robust controller design for liquid–liquid heat exchanger system". In: *IET Control Theory & Applications* 12.10, pp. 1474–1482. DOI: 10.1049/iet-cta.2017.1128.

Kim, Sunghwan and et al. (2022). "Modified Design of Two-Switch Buck-Boost Converter to Improve Power Efficiency Using Fewer Conduction Components". In: *Applied Sciences* 13.1, p. 343. DOI: 10.3390/app13010343.

Knospe, C. (2006). "PID control". In: *IEEE Control Systems Magazine* 26.1, pp. 30–31. DOI: 10.1109/MCS.2006.1580151.

Kulhánek, Tomáš and et al. (2015). "Experiences in teaching of modeling and simulation with emphasize on equation-based and acausal modeling techniques". In: *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 3683–3686. DOI: 10.1109/EMBC.2015.7319192.

Lienhard, John (2024). *A Heat Transfer Textbook*. 6th ed. Phlogiston Press.

Markley, F. Landis (1999). *Attitude Determination Using Two Vector Measurements*. URL: https://www.researchgate.net/publication/4706531_Attitude_Determination_Using_Two_Vector_Measurements/link/0c960525b7927e11c5000000/download.

Martínez, Isidoro (2015). *Radiative view factors*. URL: http://imartinez.etsiae.upm.es/~isidoro/tc3/Radiation%20View%20factors.pdf.

Modelica® - A Unified Object-Oriented Language for Systems Modeling (2014). URL: https://modelica.org/documents/ModelicaSpec33Revision1.pdf.

Mostacciuolo, E., L. Iannelli, and et al. (2018). "Modeling and power management of a LEO small satellite electrical power system". In: *2018 European Control Conference (ECC)*, pp. 2738–2743. DOI: 10.23919/ECC.2018.8550095.

Nagi, Farrukh, S.K. Ahmed, and et al. (2009). "Fuzzy bang-bang relay controller for satellite attitude control system". In: *Fuzzy Sets and Systems* 161.15, pp. 2104–2125. DOI: 10.1016/j.fss.2009.12.004.

Nguyen, Xuan Hieu and Minh Phuong Nguyen (2015). "Mathematical modeling of photovoltaic cell/module/arrays with tags in Matlab/Simulink". In: *Environmental Systems Research* 4.24. DOI: 10.1186/s40068-015-0047-9.

Pintelon, Rick and Johan Schoukens (2012). *System Identification: A Frequency Domain Approach*. 2nd ed. John Wiley & Sons Inc.

Priyanka, K. and A. Mariyammal (2018). "DC Motor Speed Control Using PWM". In: *International Journal of Innovative Science and Research Technology* 3.2.

Singh, Maulshree, Evert Fuenmayor, and et al. (2021). "Digital Twin: Origin to Future". In: *Applied System Innovation* 4.2, p. 36. DOI: 10.3390/asi4020036.

Song, Sua and et al. (2018). "Design and Implementation of 3U CubeSat Platform Architecture". In: *International Journal of Aerospace Engineering* 2018. DOI: 10.1155/2018/2079219.

Sutton, George P. and Oscar Biblarz (2017). *Rocket Propulsion Elements*. John Wiley & Sons Inc. ISBN: 1118753658.

Tiller, Michael (2001). *Introduction to Physical Modeling with Modelica*. Vol. 615. Springer Science & Business Media.

Tummala, Akshay Reddy and Atri Dutta (2017). "An Overview of Cube-Satellite Propulsion Technologies and Trends". In: *Aerospace* 4.4, p. 58. DOI: 10.3390/aerospace4040058.

Umez-Eronini-Eronini (1999). *System dynamics and control*. Pacific Grove : PWS Publishing Company. ISBN: 0534944515.

Wei, Jin, Dengqing Cao, and et al. (2017). "Dynamic modeling and simulation for flexible spacecraft with flexible jointed solar panels". In: *International Journal of Mechanical Sciences* 130, pp. 558–570. DOI: 10.1016/j.ijmecsci.2017.06.037.

Wertz, James R. (1978). *Spacecraft Attitude Determination and Control*. D. Reidel. ISBN: 9027712042.

Appendix: Simulation Graphs

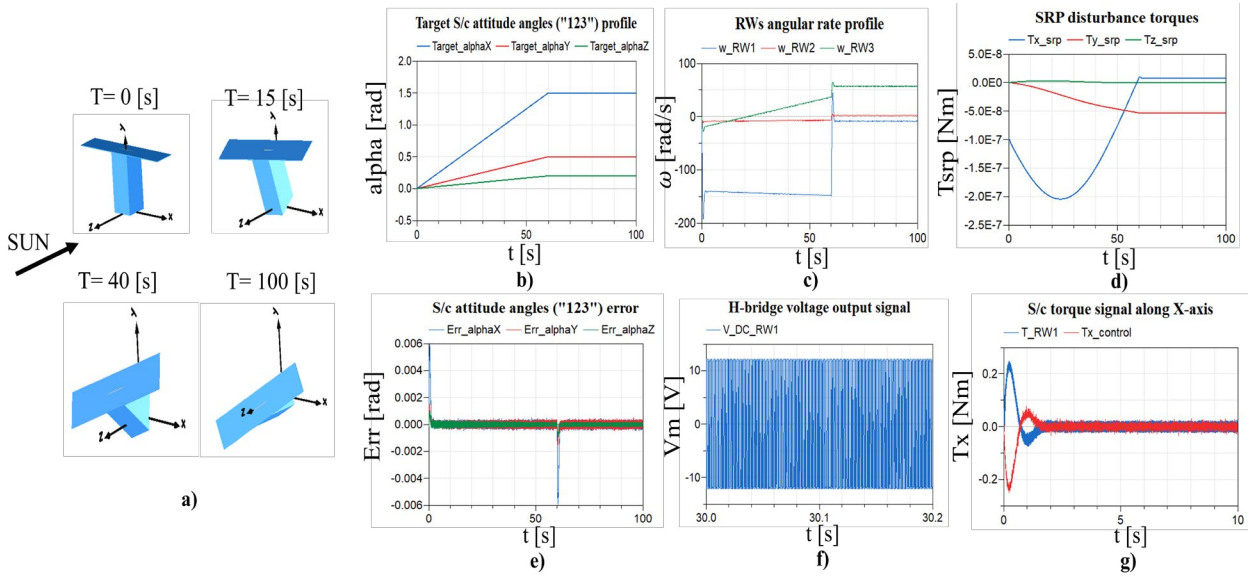


Figure 23. ADCS subsystem simulation outcome: **a)** Satellite attitude animation at four different times; **b)** Target angles profile $\hat{\alpha}_{target}^{(123)}$; **c)** RWs angular rate profile ω_M; **d)** SRP torques profile in the Dymola inertial frame {**xyz**}; **e)** Error profile between the desired attitude and the estimated one with TRIAD algorithm; **f)** The real voltage profile $\mathbf{V}_m$ applied to the DC motor of the RW1 in the time interval [30 - 30.2] [s]; **g)** Control torque profile from PID (red) and RW1 torque released along *body* **x**-axis (blue) in the time interval [0 - 10] [s].

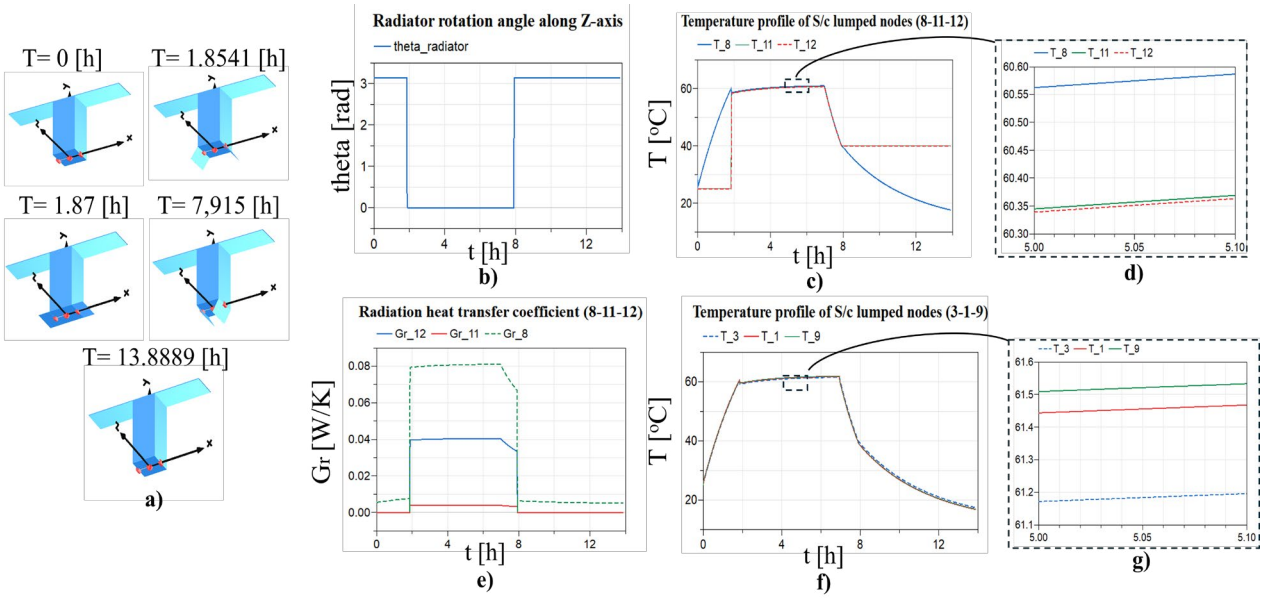


Figure 24. THR subsystem simulation outcome: **a)** Satellite radiators opening/closing animation; **b)** The evolution of the radiator angle $\hat{\alpha}_{radiator}$ with time; **c)** The lumped temperature profile of nodes (8-11-12) as shown in Figure 7; **d)** Zoom in of the temperature difference between nodes (8-11-12) in the time interval [5 - 5.1] [h]; **e)** The radiation heat transfer coefficient variation towards deep space for nodes (8-11-12); **f)** The lumped temperature profile of nodes (1-3-9) as shown in Figure 7; **g)** Zoom in of the temperature difference between nodes (1-3-9) in the time interval [5 - 5.1] [h].

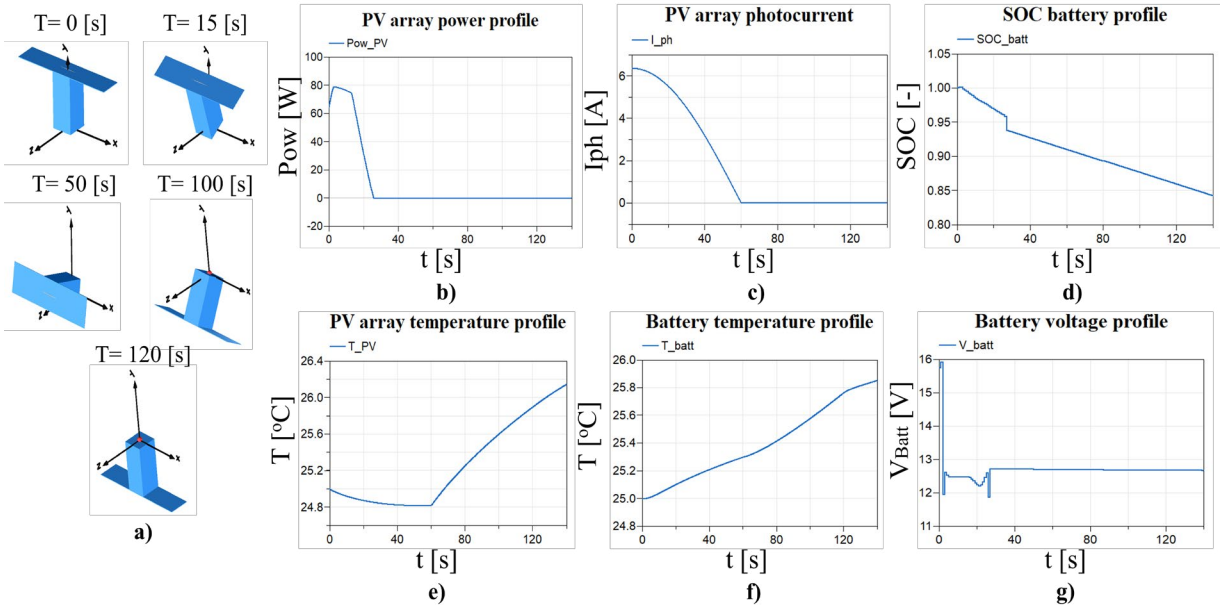


Figure 25. Satellite multiphysics simulation outcome: **a)** Satellite animation during the attitude maneuver; **b)** Power generated by a singular PV array surface; **c)** Photocurrent flowing into a singular PV circuit with time; **d)** SOC battery evolution with time; **e)** PV array temperature evolution with time; **f)** Battery temperature profile with time; **g)** Battery voltage response with time.

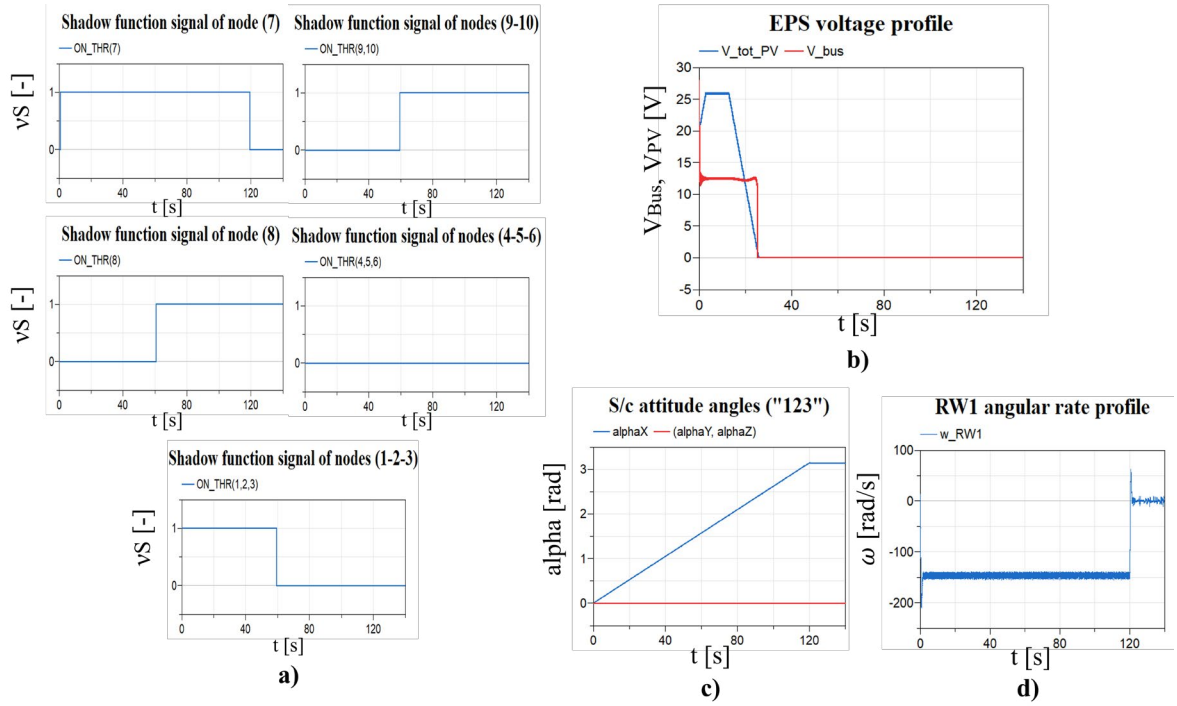


Figure 26. Satellite multiphysics simulation outcome: **a)** The shadow conditions of the satellite lumped nodes coming from the *Shadow_Model* block; **b)** The EPS voltage signals controlled by the MPPT algorithm; **c)** The satellite attitude Euler angles ("123") evolution with time ; **d)** RW1 angular rate during the maneuver