
COUPLED SIMULATION OF THE PROPULSION SYSTEM AND VEHICLE USING THE ESPSS SATELLITE LIBRARY

C. R. Koppel¹, F. Di Matteo², J. Moral³, and J. Steelant²

¹KopooS Consulting Ind.
57 rue d'Amsterdam, Paris 75008, France

²ESTEC
1 Keplerlaan, P.O. Box 299, AG Noordwijk 2200, The Netherlands

³Empresarios Agrupados
3 c/Magallanes, Madrid 28015, Spain

The paper documents the implementation and validation of the coupled simulation of the propulsion system and vehicle performed during the 4th development phase of the ESPSS (European Space Propulsion System Simulation) library running on the existing platform EcosimPro[®]. This covers a significant update of the spacecraft propulsion system modeling: the Fluid flow, Tanks and Combustion chamber components are updated to allow coupling to the vehicle's motion, the Archimedes pressure coming from acceleration and rotations given by the vehicle or by any perturbation forces are taken into account, several new features are added to the Satellite library along with new components enabling full attitude control of a platform. A new powerful compact equation is presented for solving elegantly the Archimedes pressure coming from combined acceleration and rotation in the most general case (noncollinear). Eventually, a propulsion system is modeled to check the correct implementation of the new components especially those dealing with the effects of the mission on the propulsion subsystem.

NOMENCLATURE

\vec{a}, γ Acceleration, m/s²
 \vec{F} Nonbody forces, N
 G Gravitation constant, N·m²/kg²

\vec{g}	Acceleration from body forces, m/s^2
g_0	Constant 9.80665 m/s^2
\vec{M}	Moment, $\text{N}\cdot\text{m}$
m	Spacecraft mass, kg
p	Pressure, N/m^2
\vec{P}_{perturb}	Perturbing forces vector from other celestial bodies, interaction forces, N
Q	Quaternion
\vec{r}	Vector from CoM to a point of the spacecraft, m
\vec{r}	Orbital radius vector from focus to spacecraft, m
T_{thrust}	Resultant thrust vector, N
\vec{v}	Spacecraft velocity vector, m/s
$\vec{\nabla}p$	Gradient vector of the scalar p , $(\partial p/\partial x \ \partial p/\partial y \ \partial p/\partial z)^T$
$\vec{\pi}, \vec{\Pi}$	Global vectors
ρ	Density, kg/m^3
$\vec{\Omega}$	Instantaneous rotation vector, rd/s
\times	Vector product
AOCS	Attitude and Orbit Control System
CAD	Computer Aided Design
CoM	Center of mass
DAs	Drag Areas
ECI	Earth Centred Inertial
GG	Gravity gradient
LEO	Low Earth Orbit
MTs	Magnetotorquers
RCT	Reaction Control Thruster
RWs	Reaction Wheels
SAs	Solar Arrays

1 INTRODUCTION

In the frame of the 4th development phase of the ESPSS library [1–3] running on the EcosimPro[®] platform [4], the implementation and validation of the “coupled simulation of the propulsion system and vehicle” is discussed.

EcosimPro[®] is a object-oriented Physical Simulation Modeling tool dedicated for system analyses. Its visual simulation interface is capable of solving various kinds of dynamic systems represented by writing equations and discrete events. It can be used to study both transients and steady states. For example, with the propulsion libraries ESPSS from European Space Agency (ESA), the user draws (and designs at the same time) the propulsion system with components of that specific library consisting of tanks, lines, orifices, thrusters, tees, etc. The user

can enhance the design with components from the thermal library (heaters, thermal conductance, and radiators), the control library (analogue/digital devices), the electrical library, etc.

The present work follows the first step carried out in the previous phase regarding the “evolutionary behavior of components,” where a specific Satellite library was set up for the simulation of the Attitude and Orbit Control System (AOCS) effects on the propulsion subsystem. It included the flight dynamic (orbit and attitude) capabilities for a full spacecraft with orbital and attitude perturbations. The present work focuses on a significant update of this Satellite platform propulsion system modeling and the coupling is performed by updating the Fluid flow, Tanks and Combustion chamber components while taking into account the Archimedes pressure coming from acceleration and angular rates given by the vehicle or by any perturbation forces (and torques).

The paper presents first several new features added to the Satellite library as well as some new components for performing full attitude control of a platform. It is followed by a new compact equation for solving elegantly the Archimedes pressure and finishes with some realistic validation test cases of bipropellant satellites in Low Earth Orbit (LEO).

2 ESPSS BACKGROUND

EcosimPro[®] is a Physical Simulation Modeling tool developed for ESA by Empresarios Agrupados Internacional (Spain) since 1989. EcosimPro[®] was a precursor and now, with its 28 years of careful growing, it belongs to the last generation of the common engineering tools after computer aided design (CAD) and integrated engineering analysis tools available on classical laptops. The kernel of EcosimPro[®] is an expert solver of all the equations set in the different components of a system. Thanks to such expert solver, the tool allows to manipulate components like objects that can be independently further developed with more sophisticated equations. EcosimPro[®] is based on a visual simulation tool for solving simple and complex physical processes that can be expressed in terms of equations (including ordinary differential equations and differential-algebraic equations) and discrete events.

Practically, the modeling of physical components is based on a basic “Ecosim-Pro language” (EL), an object-oriented programming language which is very similar to other conventional programming languages (Basic, C++) but is very powerful to write any equations and differential equations for modeling continuous and discrete processes. EcosimPro[®] employs a set of libraries containing various types of components (mechanical, electrical, pneumatic, hydraulic, etc.) which can be interconnected to model complex multidomain dynamic systems. The ESA ESPSS is a set of EcosimPro[®] libraries written to model all

aspects of a functional propulsion system. As a tool, ESPSS is relying on one-dimensional (1D) flow equations, thermodynamic relationships, and real fluid properties, there is no need for fudge factors; therefore, the results of the simulation could be considered as general as long as the flow is 1D and homogeneous (either as monophasic, or two-phase state or as a mixture).

3 SATELLITE LIBRARY

The satellite library contains the main components needed to build a satellite AOCS with the major interactions and perturbations that occur in flight [5, 6]:

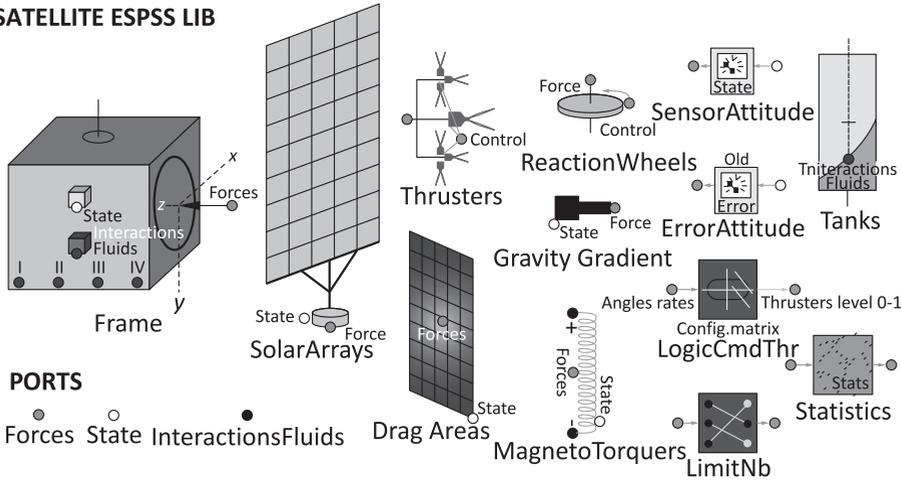
- Moon, Sun, and other planets' perturbations, forcing the implementation and use of thrusters for the North-South station keeping of a GEO (Geostationary Earth Orbit) satellite (the addition of planets allows analyses relative to interplanetary travels, flyby, etc.);
- Earth flatness or so-called “J2” perturbations to be used for the heliosynchronous satellites;
- Sun pressure interaction with the solar arrays (SAs) of the satellite;
- drag forces from upper Earth atmosphere;
- gravity gradient torques; and
- magnetic torques from the interaction of the Satellite magnetic moment with the Earth magnetic field.

In addition, a control of the satellite attitude can be added (using the existing components of the existing CONTROL library) for performing an Earth Pointing with actions on a set of reaction wheels (RWs) or on a set of thrusters. Finally, the components of the library are shown in Fig. 1.

Three ports have been designed for the transfer of information between the components of the library:

- (1) **Force port:** this is a multifunctional port allowing inputs from a set of thrusters, RWs, SAs, Drag Areas (DAs), Magnetotorquers (MTs), and gravity gradient (GG) with the port directions set at IN for the satellite frame and OUT for all other components. The variables of the port are of type SUM in order to automatically account for all mass flow rates, forces, moment, angular momentum, and power coming from all connected components;
- (2) **State port:** this is a multipurpose port allowing the attitude and orbit control, three-dimensional (3D) visualization, as well as the needed inputs

SATELLITE ESPSS LIB



PORTS

Forces State InteractionsFluids

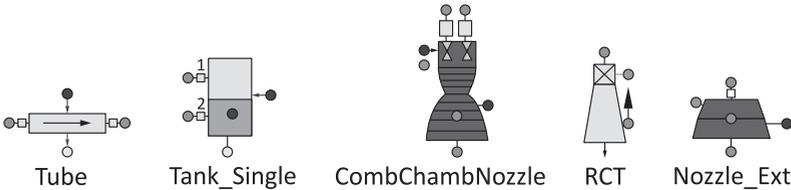
OTHER ESPSS LIB COMPONENTS
WITH ACTIONS FROM SATELLITE LIB

Figure 1 Palette of components for the Satellite library of ESPSS with coupled components in other ESPSS libraries

for the SAs, DAs, GG, and Tanks. For the port directions, there are set at OUT for the satellite frame and IN for all other components; and

- (3) **InteractionsFluids port:** this port allows to communicate interactively the mass, the location of the center of mass (CoM), and the inertia matrix of each liquid fluid in the tanks component.

3.1 Frame Component

The major component of the Satellite library has been improved in the 4th phase by adding the perturbation from the planets (from Mercury to Pluto) in order to get the possibility to enable interplanetary flights.

Its description is recalled here for completeness: the frame component is in charge of solving the flight dynamic with input data of the initial orbit and with nongravity (i. e., nonbody) forces vectors coming from the thrusters (including

reaction control thrusters (RCTs), combustion chamber nozzle, or nozzle extension from other ESPSS library), SA, DA. The perturbation forces coming from Moon, Sun, and all other planets gravity are directly solved inside the component (actually, the complete set of equations is used for computing the perturbation in order to solve accurately any fly-by or orbit around any planet).

The first main equation of the component is based on the momentum equation as follows:

$$\frac{d\vec{\pi}}{dt} = \vec{\Pi}$$

with

$$\vec{\pi} = \begin{pmatrix} \vec{r} \\ \vec{v} \\ m \end{pmatrix}; \quad \vec{\Pi} = \begin{pmatrix} \vec{v} \\ -GM_{\text{focus}} \frac{\vec{r}}{r^3} + \sum \frac{\vec{T}_{\text{thrust}} + \vec{P}_{\text{perturb}}}{m} \\ \sum \text{-thrusters_mass_flow} \end{pmatrix}$$

where \vec{v} is the velocity vector of the satellite frame with respect to the Earth centered inertial (ECI) frame; G is the gravitational constant; M is the mass of the focus body (Earth); \vec{r} is the radius vector from the focus body center; \vec{T} is the resultant thrust vector applied on the satellite CoM, in ECI; \vec{P} is the perturbation and interaction force vector in ECI frame; and m the instantaneous mass of the satellite that is corrected dynamically by the thrusters consumption if any.

The second main equation of the component is based on the dynamic equation for the attitude of satellite around its CoM as follows:

$$\frac{d\vec{H}}{dt} = \vec{\Omega} \wedge \vec{H} = \vec{M}_{\text{Control}} + \vec{M}_{\text{Perturbation}}$$

with

$$\vec{M}_{\text{Control}} = \left\{ M_{\text{thruster}} - \sum_i \frac{d\vec{H}_{\text{RW}i}}{dt} \right\};$$

$$\vec{M}_{\text{Perturbation}} = \left\{ M_{\text{Perturb}} - \sum_i \vec{\Omega} \wedge \vec{H}_{\text{RW}i} - \sum_i \vec{\Omega} \wedge \vec{H}_i - \sum_i \frac{d\vec{H}_i}{dt} \right\}.$$

The source terms come from the torques and perturbations due to the thrusters, MT, GG, and RWs, and from the perturbation torques due to the other mobile parts of the satellite (SA) and with $\vec{\Omega}$ being the instantaneous rotation with respect to the inertial orientation frame of the satellite and the angular momentum with respect to the satellite of the RWs $\vec{H}_{\text{RW}i}$ and of the other mobile parts \vec{H}_i , but with all their coordinates and derivative written in the satellite axis. The angular momentum \vec{H} takes into account the possible “evolutionary behavior” of the fluid in the tanks through the inertia matrix update.

The attitude angles of the satellite must be known in order to properly set the local data like the thrust with respect to the satellite frame into the ECI frame. Those can be given by the Cardan angles (yaw, pitch, roll) of the satellite axis with respect to the orbital frame. The attitude angles of the orbital frame can be given by the Euler angles (precession, nutation, proper rotation) with respect to the ECI frame. But in order to ensure robustness in the solutions, the equations dealing with the attitude angles are based upon the quaternion theory. Thus, the general equation of the quaternion theory is to be solved simultaneously with the other previous equations.

$$\frac{dQ}{dt} = +\frac{1}{2} [Q] \Omega. \quad (1)$$

The integration of Eq. (1) produces at each time t the quaternion $Q(t)$ that enables to retrieve (with suitable conversion matrixes) the attitude angles without any risk of singularities with unit quaternion. As the quaternion of the satellite with respect to the orbital frame is available (given by the state port), it can be used in an external control loop of the attitude of the satellite for an Earth pointing command by setting to zero its imaginary components.

3.2 Solar Arrays Component

For the current release of the library, either fixed SAs on the satellite body or an automatic orientation of the SA with respect to the Sun is considered. The component is linked to the Frame component (state port) in order to get the Satellite to Sun vector and other information.

The classic equations of Sun pressure interaction on the SA are used in this component with input coefficients of reflection, specular reflection, and absorption. During eclipses of Sun by the Earth, those interactions are null.

The output of the SAs are the 3D forces due to the Sun pressure, their moments, and the power produced by the solar cells according to the input data of the efficiency and area of the solar cells.

As for other components, the SA component is written in terms of vector to allow a number of SAs (up to 12, for instance) that are all described in terms of location, orientation of the axis of rotation, and orientation of the canonical normal to the solar cells, size.

3.3 Thrusters Component

When activated, this component outputs the thrust vector with respect to the satellite, the moment induced by the thrusters' forces, and the mass flow rates (for monopropellant as well as for bipropellant). In case of electric propulsion, the electric power consumption is provided as well.

3.4 Reaction Wheels Component

Once activated, this component outputs the angular momentum vector with respect to the satellite as well as the electric power consumption of all RWs considered. This component has been improved in the 4th phase by adding the operational limits in terms of maximum torque and angular rate or rpm limits.

3.5 Drag Array Component

This component is a new component of the 4th phase. It outputs the drag force coming from the dynamic pressure in flight. The Earth atmosphere type is taken from ESA ECSS (European Cooperation for Space Standardization) standard [7] with several cases depending on the Sun activity.

3.6 Gravity Gradient Component

This component is also a new component of the 4th phase. Based on the input from the state port coming from the satellite frame component and the inertia matrix [Inertia] of the whole body (including fluids), this component provides the moment vector with respect to the satellite without any forces (pure torque):

$$\begin{aligned} \vec{M} &\approx \frac{3GM_{\text{Earth}}}{R^3} \int_{\text{body}} (\vec{r} \cdot \vec{e}_{\text{Earth}}) \vec{r} \times \vec{e}_{\text{Earth}} dm \\ &= \frac{3GM_{\text{Earth}}}{R^3} \vec{e}_{\text{Earth}} \times ([\text{Inertia}] \cdot \vec{e}_{\text{Earth}}) \end{aligned}$$

where R is the distance from CoM to the Earth center; \vec{e}_{Earth} is the unit vector from the CoM to the Earth; and \vec{r} is the local vector from the CoM.

3.7 Tanks Component

The classic relation for Archimedes' pressure $dp = -\rho\gamma(z - z_0)$ was already taken into account in the existing Flow1D tank model of the original ESPSS [3] but this was too restrictive. In space, the satellite is rotating around its CoM (and some couples are produced by thrusters and RWs or windmill effect by the solar pressure on the SA). Further particular forces, e. g., originating from thrusters or solar pressure on the SAs, produce acceleration. Therefore, the Archimedes' pressure needs to take into account the effect of the acceleration and the effect of the rotations even when not collinear to the acceleration.

3.7.1 A new equation for the Archimedes' pressure

If \vec{a} represents the acceleration of the CoM in the inertial frame, $m\vec{g}$ all the body forces, and \vec{F} the surface forces like the reaction to the gravity, the thrust forces from the thrusters, the Sun pressure force, etc., one gets to

$$m\vec{a} = m\vec{g} + \vec{F},$$

or

$$\vec{a} - \vec{g} = +\frac{\vec{F}}{m}.$$

The Navier–Stokes equation is the Newton's law of momentum “ $\rho\vec{a}_{\text{abs}} = \rho\vec{g} + \vec{f}$ ” adapted for Newtonian fluids (with a control volume fixed in the noninertial space considered) where \vec{a}_{abs} is the absolute acceleration of a point (located at \vec{r} from the CoM) in an inertial frame and \vec{f} are all the surface forces per unit of volume (those are derived from \vec{F} above). Once we have $\vec{\Omega}$ being the rotation around CoM, \vec{r} and \vec{V}_{rel} being the location and velocity of the fluid particle with respect to the noninertial frame, and μ being the viscosity, one can write the instantaneous momentum equation as

$$\begin{aligned} \vec{a} + 2\vec{\Omega} \times \vec{V}_{\text{rel}} + \vec{\Omega} \times (\vec{\Omega} \times \vec{r}) + \frac{d\vec{\Omega}}{dt} \times \vec{r} + \frac{\partial \vec{V}_{\text{rel}}}{\partial t} + \vec{\nabla} \cdot \frac{1}{2} \vec{V}_{\text{rel}}^2 - \vec{V}_{\text{rel}} \times (\vec{\nabla} \times \vec{V}_{\text{rel}}) \\ = \vec{g} - \frac{\vec{\nabla} p}{\rho} + \frac{\mu}{\rho} \left(\vec{\nabla} \cdot \vec{\nabla} \vec{V}_{\text{rel}} + \frac{1}{3} \vec{\nabla} (\vec{\nabla} \cdot \vec{V}_{\text{rel}}) \right). \end{aligned}$$

Considering a static behavior in the noninertial frame, i. e., $\vec{V}_{\text{rel}} = 0$, it remains for the point \vec{r} :

$$\vec{a} + \vec{\Omega} \times (\vec{\Omega} \times \vec{r}) + \frac{d\vec{\Omega}}{dt} \times \vec{r} = \vec{g} - \frac{\vec{\nabla} p}{\rho},$$

thus, for constant rotation ($\vec{\Omega} = \text{const}$),

$$\vec{\nabla} p = -\rho \left(\vec{a} - \vec{g} + \vec{\Omega} \times (\vec{\Omega} \times \vec{r}) \right).$$

In the literature, it was not possible to find this kind of simple relation; so, no references can be linked to it.

The pressure difference between two points is the integral of the scalar product $\vec{\nabla} p \cdot d\vec{s}$:

$$p_2 - p_1 = \int_{\vec{r}_1}^{\vec{r}_2} dp = \int_{\vec{r}_1}^{\vec{r}_2} \vec{\nabla} p \cdot d\vec{s}.$$

Note that centripetal acceleration (see [8])

$$\vec{\Omega} \times (\vec{\Omega} \times \vec{r}) = -\vec{\nabla} \left(\frac{1}{2} (\vec{\Omega} \times \vec{r})^2 \right).$$

Hence, if the density ρ is constant, one gets the new equation:

$$p_2 - p_1 = \left[-\rho (\vec{a} - \vec{g}) \cdot \vec{r} + \frac{1}{2} \rho (\vec{\Omega} \times \vec{r})^2 \right]_{\vec{r}_1}^{\vec{r}_2}.$$

To the authors' knowledge, this compact formulation for the Archimedes' pressure difference in an accelerating-rotating frame could not be found in the literature.

3.7.2 Case of tanks and components of the Fluid_Flow_1D library

Inside tanks, the free surface should be considered for one of the points; hence, the Archimedes pressure at \vec{z}_b with respect to the point \vec{z}_{free} (a single point belonging to the free surface is enough) is given by

$$p_b - p_{\text{free}} = \left[-\rho \frac{\vec{F}}{m} \cdot \vec{r} + \frac{1}{2} \rho (\vec{\Omega} \times \vec{r})^2 \right]_{\vec{z}_{\text{free}}}^{\vec{z}_b} \quad (2)$$

where \vec{F} is the sum of all nonbody forces acting on the CoM, i.e., surface or contact forces: thrust, perturbation forces due to Sun pressure, drag, etc. and also including the reaction force to the weight for the case of simulation on ground.

The free surface of the tank is given by a 3D discretization of the tank using the equation like (2) for the location of the liquid phase until the volume reaches the current given volume of liquid.

The shape of the free surface is a paraboloid having an axis parallel to $\vec{\Omega}$. Its deviation from $\vec{\Omega}$ depends on the strength of the specific force \vec{F}/m .

For the tubes, a similar equation is used:

$$p_2 - p_1 = \left[-\rho \frac{\vec{F}}{m} \cdot \vec{r} + \frac{1}{2} \rho (\vec{\Omega} \times \vec{r})^2 \right]_{\vec{r}_1}^{\vec{r}_2}.$$

It is to be highlighted that in this equation, \vec{r} is the distance between the point considered \overrightarrow{OM} and the real current CoM.

Because in a dynamic system the three vectors \vec{F}/m , $\vec{\Omega}$, and the vector from the reference origin O to the CoM noted \overrightarrow{OCoM} may change every time, a very powerful technique for fast integration of the system dynamic vs. time is to use additionally the derived variables $\overrightarrow{\text{Accel}}$, $\overrightarrow{\text{Omega}}$, \overrightarrow{OCoM} defined by differential

equations from the three vectors \vec{F}/m , $\vec{\Omega}$, and \overrightarrow{OCoM} . For example, one uses $\overrightarrow{\Omega}$ derived from $\vec{\Omega}$ by

$$\frac{d\overrightarrow{\Omega}}{dt} = \frac{\vec{\Omega} - \overrightarrow{\Omega}}{\tau}$$

where τ is the time constant. Finally, the equation used for the Archimedes pressure is the following:

$$p_2 - p_1 = \left[\rho \overrightarrow{Accel} \cdot (\overrightarrow{OM} - \overrightarrow{OCoM}) + \frac{1}{2} \rho \left(\overrightarrow{\Omega} \times (\overrightarrow{OM} - \overrightarrow{OCoM}) \right)^2 \right] \frac{\overrightarrow{OM}_2}{\overrightarrow{OM}_1}. \quad (3)$$

Those dynamic vectors are set as global for every component of ESPSS, in the existing `Fluid_Flow_1D` library, so that there is no need to add any links between the tubes and the frame for being able to use Eq. (3).

3.8 LogicCmdThr Component

This new component allows to perform the active stabilization with thrusters. Its input data are the force, location, and orientation of the thrusters, and some data regarding the stabilization accuracy (angular rate limit and attitude angles limit).

The logic for the stabilization follows the principle of the Dead-zone control with hysteresis: for each axis, the angular rate drives the evolution of the attitude angle. Once a switching limit (that is, a combination of angle and angular rate) is reached, a torque should be executed in order to control the attitude by reversing the sign of the angular rate. This kind of control is sketched in Fig. 2 in the phase space of one axis and without any perturbations: the limit cycle is the loop *ABCD* indefinitely.

The command of the thrusters is performed with `[matCV]`, a rectangular configuration matrix (its columns are the torque components produced by each thruster activated at 100%) according to

$$\overrightarrow{\text{Torque}} = [\text{matCm}] \overrightarrow{\text{thrusterslevel}}.$$

So, the pseudoinverse of the rectangular configuration matrix allows selecting each thruster's level for a given set of torques. Because the thrusters can only produce thrust in one direction, a vector from the null space of the rectangular matrix `[matCM]` should be added in order to get only null or positive levels for each thruster:

$$\left[\overrightarrow{\text{NullSpaceVectors}} \right] = [\text{Identity}] - [\text{matCM_pseudoInverse}] \cdot [\text{matCM}].$$

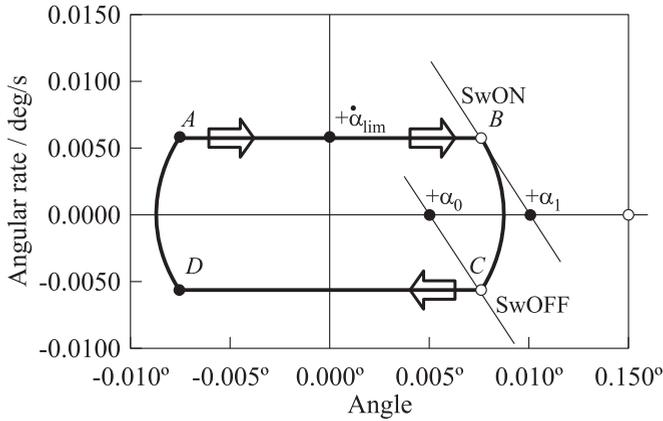


Figure 2 Attitude control Dead-zone with hysteresis (torque = $7.6 \cdot 10^{-6}$ N·m; inertia = 0.0325 kg·m²; and angular rate limit = 0.0057 deg/s)

3.9 LimitNb Component

Linked to the previous component, this component limits automatically the number of thrusters fired simultaneously and, in case of need, sequences the firing between the thrusters. Its input data are the maximum number of thrusters simultaneously fired and the period of sequencing.

3.10 Statistics Component

Linked to the previous component, this component is exclusively used for transparently controlling the thrusters (directly from the input port) while keeping trace of the use of different thrusters.

4 SATELLITE LIBRARY VALIDATION CHECKS

The validation of the library including flight dynamic validation, attitude control validation, and tank Archimedes pressure validation is described in [9].

5 SIMULATION CASES FOR COUPLED PROPULSION SYSTEM AND VEHICLE

In order to highlight the capabilities of the SATELLITE library and, particularly, the coupled Simulation of the Propulsion System and Vehicle, various realistic examples are presented.

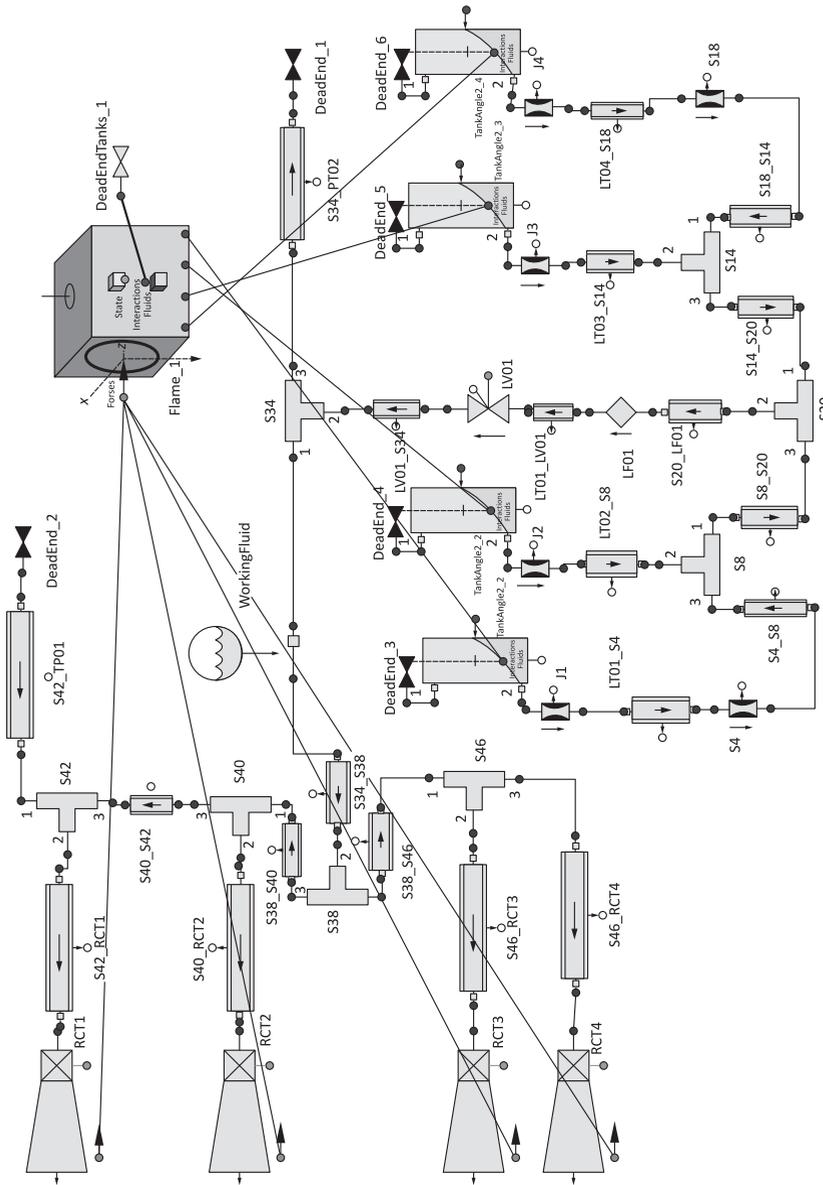


Figure 3 Propulsion system of a satellite with coupling of the propulsion system with acceleration and angular rate

5.1 Propulsion System of a Generic Satellite with Vehicle Coupling by Induced Acceleration and Angular Rate

The model of the propulsion system considered is shown in Fig. 3. It comprises 4 main tanks, several valves, filters, tees and tubes, and 4 RCTs.

The new features of the satellite library allow the connection of the force vector from each RCT to the Frame component, from each tank to the Frame component in order to set the vehicle mass, acceleration, inertia matrixes, and angular rate. Those variables are taken into account for the Archimedes' pressure in the tanks with right location of their free surface and into the whole manifold and, finally, at the RCS (Reaction Control System) injector.

The main results of the simulation are shown in Fig. 4 where a thrust pulse of 3 s is performed.

The derivative of three vectors \vec{F} , $\vec{\Omega}$, and \overrightarrow{OCoM} used for the fluid flow 1D components are almost identical to the original one given by the Frame component of the Satellite library.

During the thrust pulse, the mass decreases according to the tanks depletion (coming from the thrusters consumption through the tubing manifold) and after the pulse, the acceleration cancels while the angular rate around the transverse axis of the vehicle remains constant.

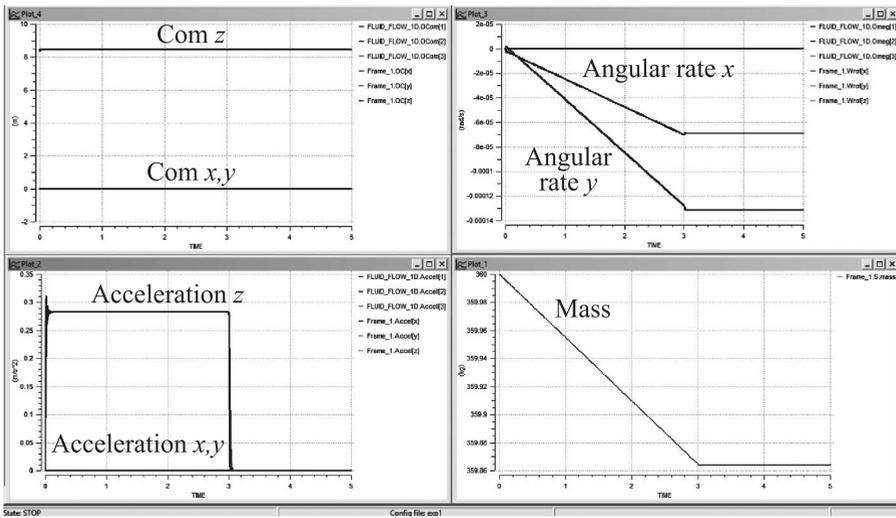


Figure 4 Center of mass, angular rate, acceleration, and mass of the coupled propulsion system with a thrust pulse 3 s

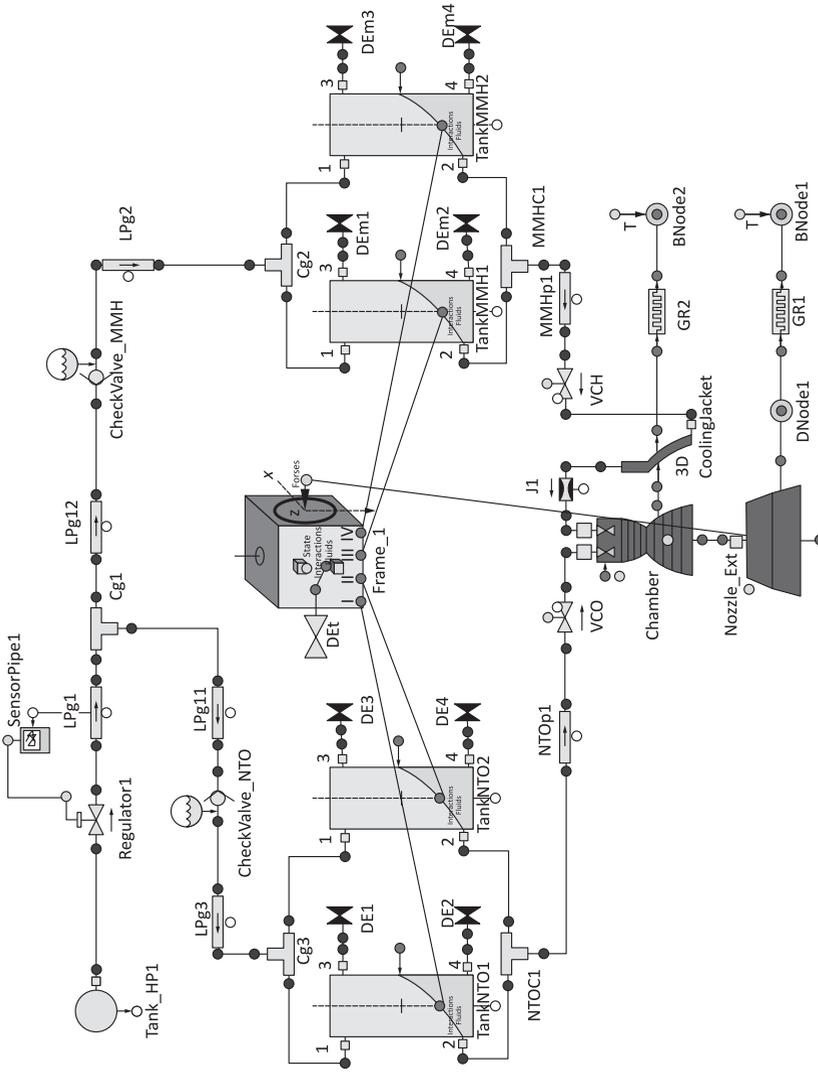


Figure 5 Bipropellant propulsion system of a satellite with coupling of the propulsion system with acceleration and angular rate

MMH and NTO) and after the pulse, the acceleration cancels while the angular rate around one of the transverse axes of the vehicle remains constant at a high rate because the main rocket engine thrust is intentionally deviated along X axis by $+70$ mm relative to the CoM.

Those results show that the new feature regarding the coupling between propulsion and vehicle are well managed and the propulsion system can be simulated with a great realism.

5.3 Active Attitude Control with Thruster

The model of the propulsion system considered is shown in Fig. 7. It comprises 4 thrusters for performing thrust in one main direction and with components along a transverse axis for performing a full three axis attitude control. The logic presented in the previous section is used for the assessment of the system, taking into account the fact that only one thruster can be fired every second and that a delay of 1 s is needed before using a thruster.

The initial orbit is a LEO with all possible perturbations in attitude with the worst case of CoM location (hence, Sun pressure torques, drag torques). This case can represent a Cubesat propulsion system using liquid μ PPT.

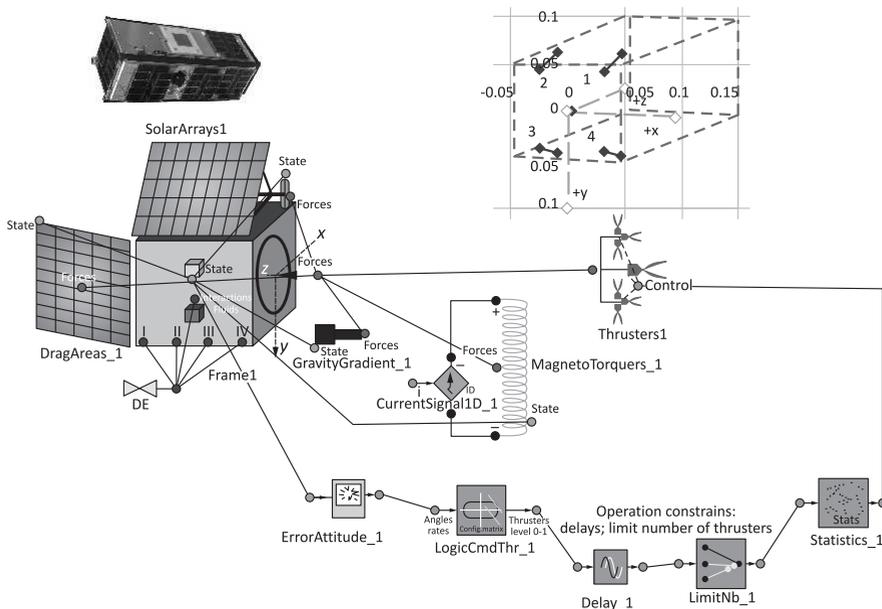


Figure 7 Propulsion system for active AOCS

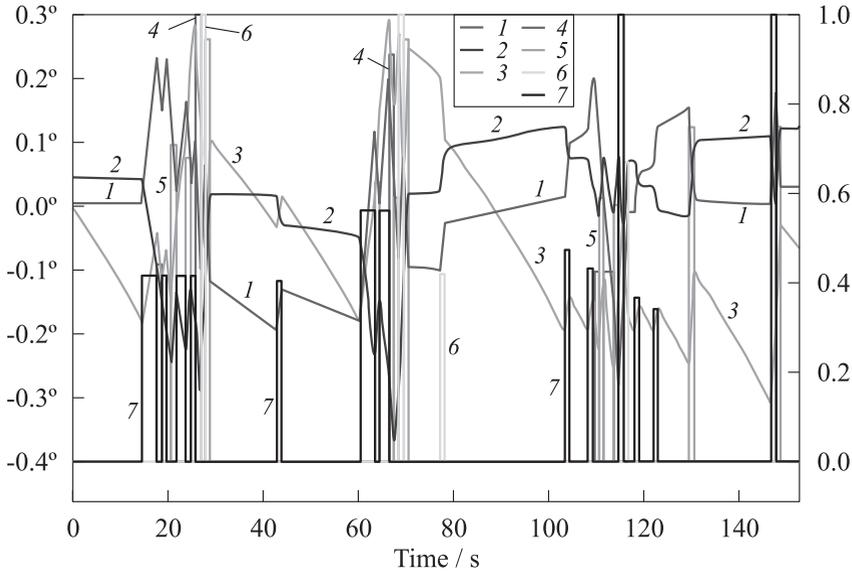


Figure 8 Logic switch function (left axis) and thruster level (right axis) during a stabilization (zoom 150 s): 1 — LogicCmdThr_1.Switch[1]; 2 — LogicCmdThr_1.Switch[2]; 3 — LogicCmdThr_1.Switch[3]; 4 — Thrusters1.OnOff.signal[1]; 5 — Thrusters1.OnOff.signal[2]; 6 — Thrusters1.OnOff.signal[3]; and 7 — Thrusters1.OnOff.signal[4]

The goal of the test case is to stabilize the satellite within $\pm 0.2^\circ$ for all three axes.

Figure 8 shows a zoom (during 150 s) that the constraints imposed to the propulsion system are well respected during the simulation: according to the value of the switch function of the Dead-zone control with hysteresis, none or one of the thrusters are activated (level between 30% and 100%) for a duration of 1 s and, when needed, thrust pulses are successively performed every second. The overall result of the stabilization within $\pm 0.2^\circ$ for the three-axis roll, pitch, yaw is shown in their phase space (Fig. 9b). One can notice that only a few very small overshoots occur for the control of the roll axis, while for pitch and yaw, the attitude goal is fulfilled entirely on this example.

6 CONCLUDING REMARKS

The paper has presented in detail the progress made during the 4th phase of the work performed on the ESPSS library within the existing tool EcosimPro[®]. The significant advance is related to the implementation of the Archimedes pressure

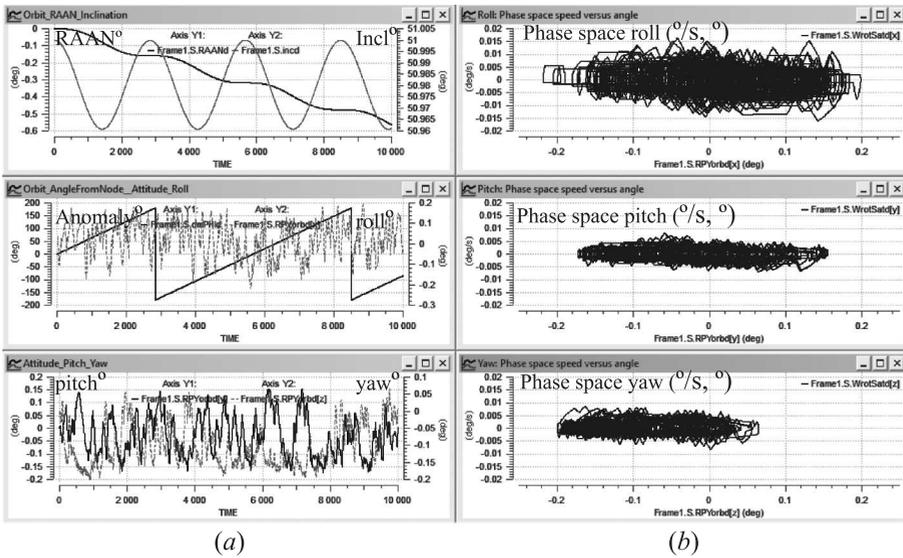


Figure 9 Active stabilization of attitude within 0.2° for two orbits: (a) orbit and attitude angles; and (b) phase spaces

for the most general case of noncollinear acceleration and angular rate for the ESPSS fluid components. Thanks to the use of a new very compact equation, this could be performed in a much easier form than in the previous complex formulations. This has been successfully implemented for the components needed for a propulsion system. Two examples of propulsion systems have been presented with highlights on the coupling between propulsion and vehicle.

Among the progress, new components have been implemented successfully into the Satellite library allowing to simulate with “good” realism typical spacecraft missions, particularly of interest for propulsion applications as the attitude control for a small satellite with thrusters.

ACKNOWLEDGMENTS

The research leading to these results has received funding from ESA contract No. 4000111093/14/NL/PA.

REFERENCES

1. Steelant, J., M. De Rosa, J. Moral, and R. Pérez. 2010. ESPSS simulation platform. *Space Propulsion 2010*. San Sebastian, Spain.

2. Di Matteo, F., and J. Steelant. 2013. Multi-disciplinary propulsion simulations at engineering level by means of the European space propulsion system simulation ESPSS. RTO/AVT/VKI lecture ser. on fluid dynamics associated to launcher developers. St. Genesius-Rode, Belgium: Von Karman Institute. EN-AVT-206-06.
3. ESPSS release 3.1. Available at: <http://www.ecosimpro.com/products/espss/> (accessed August 23, 2017).
4. EcosimPro[®]. Available at: <http://www.ecosimpro.com/products/ecosimpro/> (accessed August 23, 2017).
5. Koppel, C. R., M. De Rosa, J. Moral, and J. Steelant. 2012. Effects of a satellite mission on the propulsion subsystem. *SP2012 2364281 Space Propulsion*. Bordeaux, France.
6. Koppel, C. R. 2013. ESPSS satellite library. *ESPSS Workshop, ESA/ESTEC*. Noordwijk, The Netherlands.
7. ESA/Space Agencies/Industry Associations. November 15, 2008. European Cooperation for Space Standardization. Space engineering. Space environment. Standard ECSS-E-ST-10-04C.
8. Lewalle, J. 2006. *Incompressible fluid dynamics: Phenomenology, concepts and analytical tools*. Lecture notes. Syracuse University.
9. Koppel, C. R., M. De Rosa, F. Rodriguez, and J. Steelant. 2013. The AOCS' effects on the propulsion subsystem using the ESPSS satellite library. *5th EUCASS*. Muenchen, Germany.
10. Koppel, C. R., J. Moral, M. De Rosa, R. P. Vara, J. Steelant, and P. Omaly. 2011. Satellite propulsion modeling with EcosimPro: Comparison between simulation and ground tests. *Progress in propulsion physics*. Eds. L. T. DeLuca, C. Bonnal, O. Haidn, and S. M. Frolov. EUCASS advances in aerospace sciences book ser. TORUS PRESS – EDP Sciences. 2:743–764.