

COUPLED MISSION AND GNC ANALYSIS FOR SPACE ROBOTIC MISSIONS

Sven Weikert

Astos Solutions GmbH, Germany, Sven.Weikert@astos.de

Andreas Wiegand^{*}, Prof. Walter Fichter[†], Rainer Saage[‡]

Advance space missions, like orbital servicing missions, comprise a high degree of dependencies between subsystems and analysis procedures. Those dependencies are normally not very well represented by today's analysis tools as used for standard space missions. However, advanced space missions already required a high degree of confidence in early mission phases to keep the overall costs low. An efficient approach to achieve such confidence are highly flexible simulation environments, which allow coupled analysis of mission and GNC aspects at the same time already in Phase A of a project. A space robotics simulator will be presented, which combines mission analysis, GNC design and analysis, as well as realistic visualization of the scenario with OpenGL including detailed computation of environmental disturbances utilizing the geometrical knowledge within OpenGL.

The ASTOS software provides the highly flexible mission analysis component and exports the real world representation to the GNC simulator, which is implemented in Simulink. The realistic visualization is done by VESTA. It computes the differential forces and moments due to solar pressure and air drag in high accuracy and faster than in real-time and feeds back this information to the GNC simulation. Moreover VESTA mimics the scenario as realistic as possible, which allows to verify the performance of visual sensors. Such sensors are essential for any rendezvous or planetary and lunar landing maneuvers and are normally difficult to test for complete space scenarios.

Finally, an interface for a manipulator arm completes the list of feature. The combination of Simulink - the most widely used software for GNC design and analysis - with the COTS software ASTOS and the open source software VESTA provides an extremely flexible and powerful solution to investigate critical flight phases of advanced mission scenarios. The Space Robotics Simulator has been applied to the German Orbital Servicing Mission and has proven its added value for future space projects. Especially for close approach maneuvers, like berthing maneuvers, the coupled analysis is able to give full confidence in proposed strategies or to expose its weak points. The development of the Space Robotics Simulator has been co-funded by a grant of DLR/BMWi.

^{*} Astos Solutions GmbH, Germany, Andreas.Wiegand@astos.de

[†] Universität Stuttgart, Germany, Walter.Fichter@ifr.uni-stuttgart.de

[‡] Universität Stuttgart, Germany, Rainer.Saage@ifr.uni-stuttgart.de

I. BACKGROUND

The simulation of multi-vehicle scenarios like rendezvous and docking poses an extraordinary challenge for the simulation tools in use. Due to the short distances between the vehicles even minor propagation errors could become critical. Interaction between the vehicles is not negligible, e.g. a vehicle can be shielded by the other vehicle from the braking residual atmosphere or solar radiation. Such shielding effects obviously might also become relevant in the communication with ground stations or relay satellites.

A further reason why rendezvous scenarios are demanding is the multitude of (relative) navigation sensors in use that have to be simulated. Furthermore the motion of robotic arms induces counteraction of the satellites body which affects the AOCS system.

II. ASTOS OVERVIEW

The ASTOS software is an industrial grade COTS optimization, mission and system analysis software. It combines a highly flexible scenario definition based on a graphical user interface and an extensive object oriented model library with tools for trajectory and vehicle design optimization using large scale direct optimization and random search techniques, short and long time propagation based on numerical and semi-analytical methods, and interfaces to various tools like Simulink for detailed GNC/AOCS design and analysis.

ASTOS is suited to model and analyze endo- and exo-atmospheric, orbital and interplanetary missions like launch, (re-)entry, observation, satellite communication, rendezvous, formation flying, constellations, aero-assisted missions.

Originally established as optimization software, ASTOS has been extended to a powerful mission analysis software providing a virtual spacecraft designer including sensor specification (Fig. 1) and analysis and a GNC design and analysis software interfacing with Simulink. The ASTOS-GNC version provides the capability to specify the space scenario in the known ASTOS user environment and provides the dynamics and environment simulation as s-function to Simulink, where the sensor and actuator equipment can be modeled in detail and where all onboard algorithms can be implemented for later export to real-time platforms. Moreover a ground station and simulation station completes the functionality. A Simulink library with equipment models and algorithms reduces the configuration time at least for a first analysis step. Finally, a 3D visualization and animation can be performed with the VESTA software (Virtual Environment for Space and Terrestrial Applications).

ASTOS focuses on the trajectory and GNC related aspects of a mission and includes all relevant subsystems and disciplines like GNC/AOCS, power, thermal, structure, aerodynamics, to provide a complete analysis of loads and budgets during for all analysis processes.

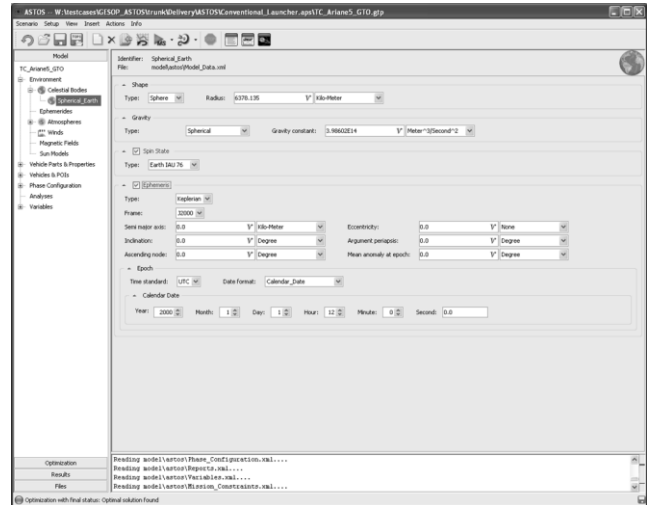


Fig. 1: Celestial body definition in ASTOS

ASTOS is the basis for the Launcher GNC Simulation Sizing Tool [4] and it has been applied as Space Robotics Simulator to the German Orbital Servicing Mission (DEOS) [1], where it has proven its added value for future space projects. Especially for close approach maneuvers, like berthing maneuvers, the coupled analysis is able to give full confidence in proposed strategies or to expose its weak points.

Virtual mission design is derived from the trajectory analysis including related simulations and subsystems. Such are primarily trajectory design and analysis, mission analysis, vehicle design, GNC and related subsystems and disciplines like propulsion, aerodynamics, structure, power, thermal, communication, and data handling. Such subsystems are considered in a relevant maturity to answer questions concerning the feasibility of a concept or mission analyzing loads and budgets.

Coupled GNC and Mission Analysis

Typically mission analysis and GNC analysis are separate tasks performed even in different project phases. In consequence it is typically not possible to consider the AOCS system during the mission analysis in phase A. Instead rough estimates are applied e.g. for the fuel consumption of the attitude control system.

With ASTOS 8 it is possible to perform a coupled GNC and mission analysis already in phase A, whereas ASTOS is shipped with a variety of representative example models and algorithms that can be used as long the AOCS system is not yet defined.

As the assessment of several projects showed (see Fig. 2) stronger investments in early design phases reduce the risk of cost overruns. This can be applied also and in particular to the mission analysis. The more detailed it will be performed, the lower the risk of an infeasible design will be.

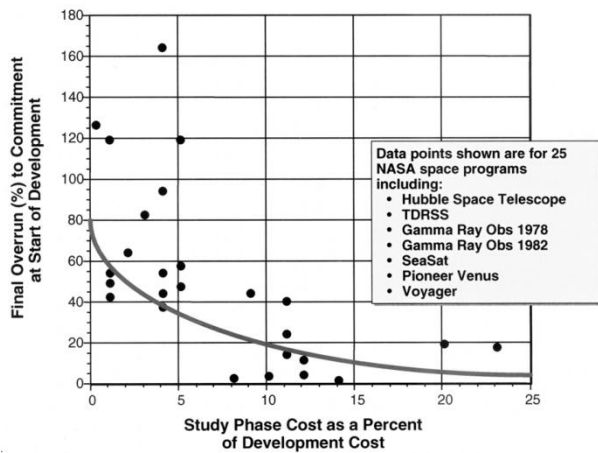


Fig. 2: Cost overrun as function of early phase investments (source NASA)

A few more examples shall illustrate that a detailed mission analysis is crucial:

- Due to limited accuracy in thrust vector alignment and thrust value, attitude control thruster firings will produce not only moments but also a residual force that changes the orbit (critical in rendezvous scenarios).
- The operation of a robotic arm causes complementary attitude and position changes of the core satellite. Depending on the operational mode those perturbations may not be compensated and in consequence attached sensors might lose their target which can result in a abort of the rendezvous.
- Eclipses or shadows from the other satellites can reduce the power budget of the mission phase.
- Sun light can blind sensors depending on the materials in use also indirect blinding due to

reflections might occur, which can be revealed only if the vehicle is accurately modeled.

For such detailed analyses ASTOS has been extended by several sensor models and models for the power and thermal system. Furthermore user-defined models for actuators and sensors as well as the AOCS system can be implemented in MATLAB/Simulink and linked to ASTOS.

In this new linked mode, developed for the explained coupled GNC and mission analysis, Simulink becomes the main tool that controls the simulation process.

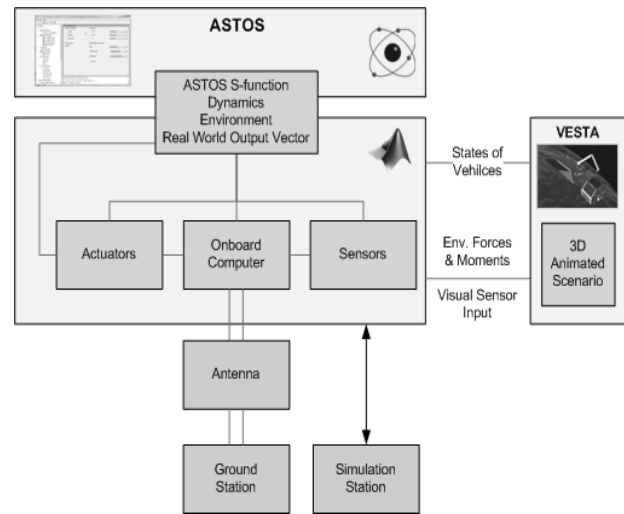


Fig. 3: Components of a coupled GNC/mission analysis using ASTOS and MATLAB/Simulink

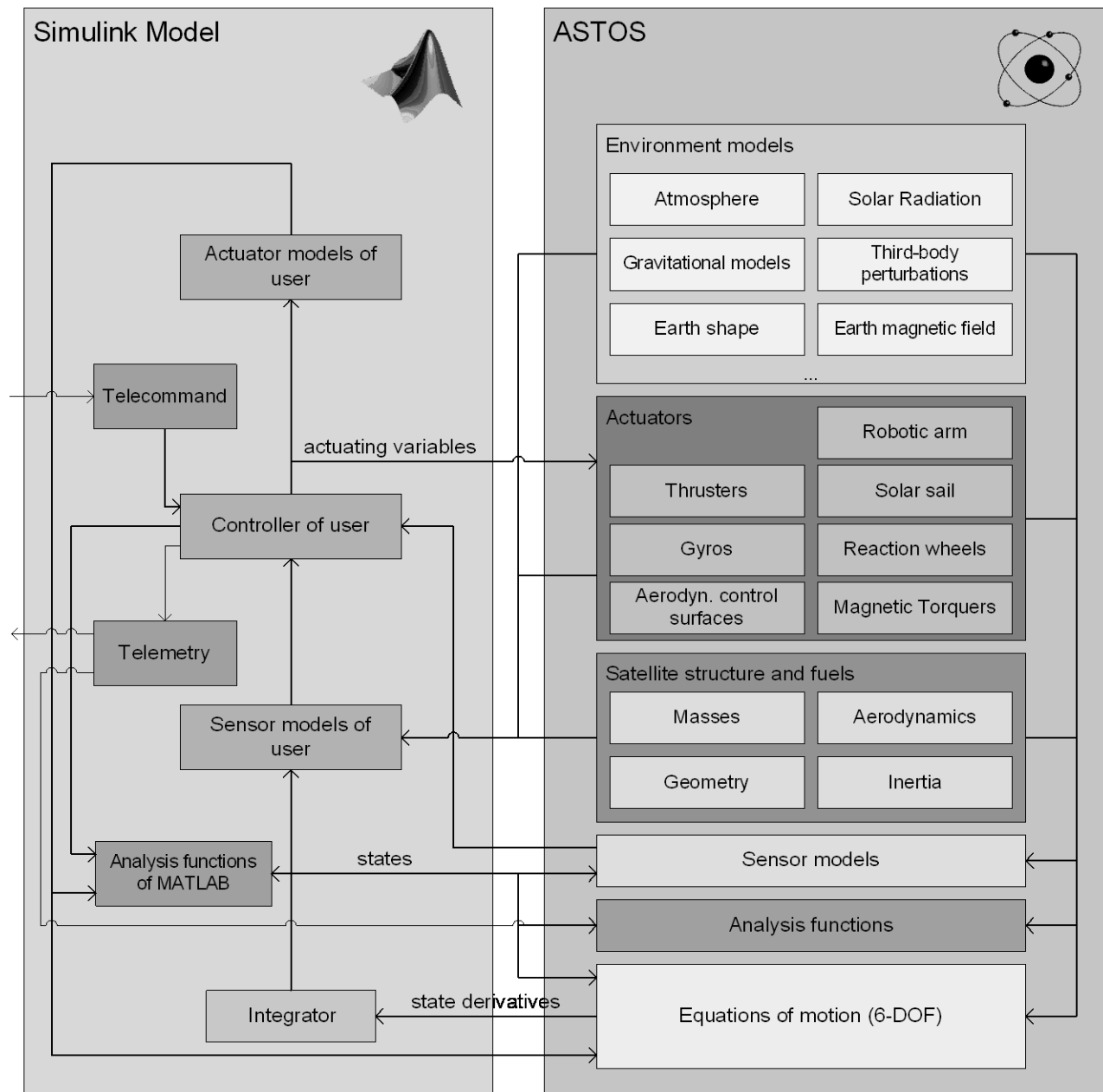


Fig. 4: Architecture and data flow of the coupled GNC/mission analysis

All parts of the ASTOS model library like equations of motion, sensor, actuator and environment models can be used. Alternatively ASTOS comes with a MATLAB toolbox (see Fig. 5) that was developed during the DLR funded SRSIM (Space Robotics Simulator) project. It comprises standard models for typical sensors (e.g. Lidar, star tracker) and actuators (e.g. thrusters, gyros, magnetorquers), control algorithms (e.g. spin control, target pointing, quaternion feedback), utility blocks (e.g. coordinate transformations) as well as blocks to realize a telecommand/telemetry console and also a simulation console (via it is possible e.g. to simulate errors).

In Fig. 6 an example with the pulsed cold gas thruster model of ASTOS/SRSIM is shown. In the coupled simulation mode where Simulink serves as the main software tool all analysis and plotting functions of ASTOS are still available. Fig. 4 shows the architecture of the coupled simulation: On the left all components implemented in Simulink are shown, while on the right all available ASTOS models are presented. ASTOS is linked to Simulink via an S-function interface in such a way that all output functions of ASTOS are accessible from Simulink.

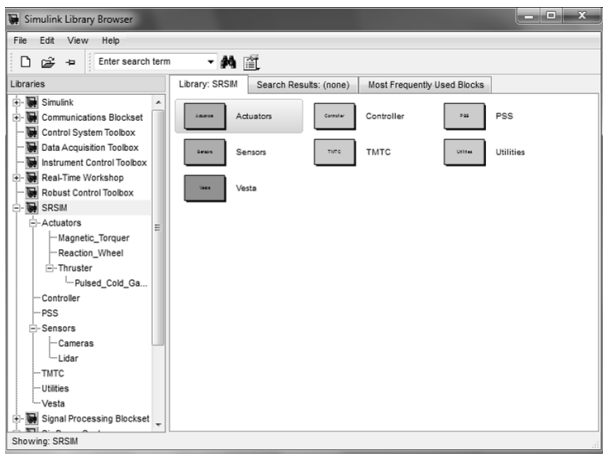


Fig. 5: Simulink toolbox provided by ASTOS/SRSIM

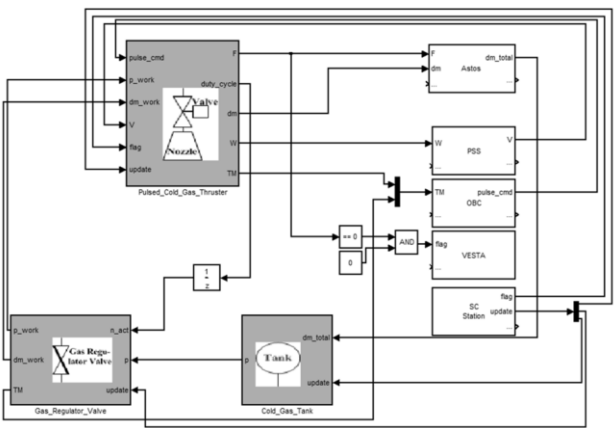


Fig. 6: Pulsed cold gas thruster model using blocks from the SRSIM database

III. SIMULATION OF IMAGING SENSOR OUTPUT

Within the new coupled Simulink/ASTOS architecture it is possible to simulate the raw output of imaging sensors like cameras. For this purpose the ASTOS camera simulator is available. This simulator has a TCP/IP interface and can therefore run on a different computer. A corresponding Simulink block that serves as interface to ASTOS and the Simulink model is shipped with ASTOS.

One design goal for the camera simulator was its real-time capability, which shall allow an integration of the camera simulator into a processor-in-the-loop simulation. For that reason a pure ray-tracing approach was not viable. Therefore the simulator is partly relying on OpenGL techniques and benefits from the performance of the graphics hardware. Nonetheless ray-tracing techniques are implemented on top of OpenGL wherever required and beneficial.

The camera simulator provides all realism-related characteristics of the underlying VESTA graphics engine [2] like shadowing, reflections, atmospheric scattering or the visualization of exhausts using particle systems. Furthermore sensor- and optics-specific characteristics are considered. These are:

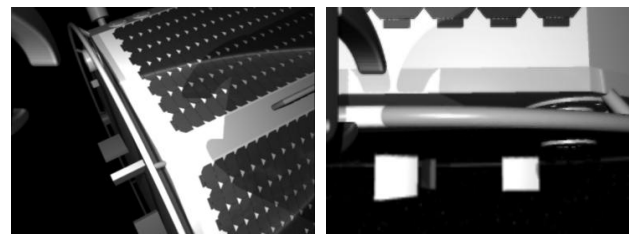


Fig. 7: Example images produced by the camera simulator

- the field-of-view (x/y)
- sensor resolution (x/y)
- focal length
- sensor technology
- sensor spectrum

Fig. 7 shows example images produced with the camera simulator for the DEOS [1] project.

The pure software solution has been coupled with a CameraLink interface card. This setup can replace a real industrial camera and makes it possible to connect the camera simulator to a framegrabber and image processing unit. The derived navigation information can be then fed back into the GNC simulation.

The interface supports single base, medium and full mode with 1 to 8 taps (PCIe version; PCI version max. 4 taps) and dual base mode with both channels running at the same frequency, which supports monochrome and color cameras. The pixel clock can be set between 20 and 85 MHz with resolutions up to 2048x2048 pixels.

The emulation of stereoscopic cameras is also supported. This can be realized by either feeding side-by-side frames or by transmitting two single mode frames in parallel).

The camera simulator cannot replace the real camera but in early design phases, especially when the mechanical design is not yet fixed and satellite mockups would be too expensive or mockup manufacturing would be too slow, it can serve as a verifications means for the sensor setup and can supports the selection of imaging sensors since it can identify sensor blinding, inadequate field-of-views, insufficient field-of-depth when the distance ranges or the minimum frame rate to be achieved.

Within the simulation environment it is furthermore possible to adjust camera parameters and to observe the changes in the resulting image stream on the fly. In such a way design errors can be identified in early project phases helping to reduce cost and project risks.

For the simulation of scanning sensors like Lidar further aspects have to be considered. During a single scan objects can move within the field-of-view. This results in a distortion of the scanned object surface, which is important to be considered since it is critical for the evaluation and accuracy of the navigation solution. The distance information as such is already part of the rendering process and can be provided analogue to the usual color information applied to a pixel but due scanning mode and relative motion every “pixel” has to be

calculated at a different simulation time (a similar effect occurs using a sensor with a rolling shutter). Even though not the whole “frame” has to be produced at every simulation time it is quite impossible to apply small integration steps for each individual pixel without losing real-time capability. For line scan Lidars therefore the frame is split into blocks. The beginning of a block corresponds to a certain simulation time as well as the end. This time interval is integrated as whole, thus the number of integration intervals is significantly reduced allowing a real-time simulation.

As far the distance information for a certain pixel at the beginning of a block interval and at the end of a block interval is not infinity the distance is simply interpolated. If at the end and the beginning both snapshots contain an infinite pixel distance, the pixel distance value will be set to infinity. In all other cases the information about the distance to the edge of the vehicle (the next infinity or non-infinity-pixel) is used to identify if the pixel should be associated to infinity distance or not (by a spatial interpolation). In case of non-infinity the distance information is then interpolated using the distance information of the identified edge pixel (instead of the infinity value). Using adequate block sizes this technique is suitable to maintain real-time capability. The size is either user-defined or can be automatically adapted according to the desired frame rate.

Other scan geometries (e.g. spiral scans) can be considered in the same way; only the block geometry has to be adapted.

IV. MODELLING OF DIFFERENTIAL FORCES AND TORQUES

For the calculation of forces and torques due to solar radiation and residual atmosphere respectively, the same graphical engine as presented in the previous sections is used. This approach has several advantages. First of all there is already an interface that deals with a 3D surface model of the spacecrafts and second the graphics hardware can deal with many surface triangles in parallel.

The same technique that is used to calculate shadows can be used to identify regions of a spacecraft that are shielded. For the calculation of shadows the scene is rendered from the perspective of the light source. All visible parts are then not inside the shadow, non-visible parts of the vehicle are inside the shadow area.

Here instead, the vehicle is rendered from the direction of the perturbing source, which is the vector from the sun to the vehicle in case of solar radiation pressure, and the negative aerodynamic velocity vector in case of the atmospheric drag. Furthermore the scene is rendered with an orthographic instead of a perspective projection and in such a way that the vehicle fits into the field of view (see Fig. 8).

At each pixel the surface normal is known. This surface normal is essential for the calculation of the local

force which is integrated over all pixels to get the overall force and the moment with respect to the reference point.

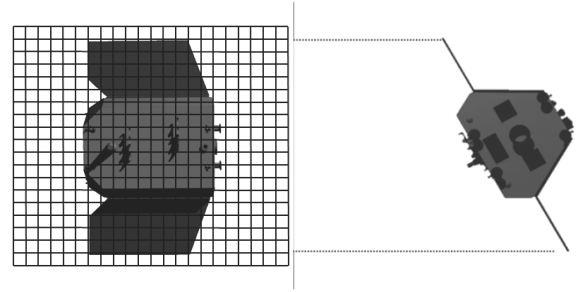


Fig. 8: Rendering technique used to calculate forces and moments

The following methodology is used to calculate the force due to solar radiation on a pixel tile: The specific impulse (force) that can be transferred to the tile if full absorption is considered can be defined as

$$\frac{\Delta p}{\Delta t} = F = \frac{S}{c} A \quad [1]$$

whereas S is the solar constant, c is the speed of light and A is the projected area exposed to the sun light.

Since the sun light is not fully absorbed three components have to be distinguished:

- absorbed light
- specularly reflected light
- diffuse reflected light

Fig. 9 illustrates these components. The shares of these components are defined by the absorption coefficient α , the specular reflection coefficient ρ and the diffuse reflection coefficient δ whereas the sum of these coefficients must be 1.

The force due to absorption on the tile is then defined as

$$\vec{F}_\alpha = -\alpha \frac{S}{c} A \cos \theta \vec{s}, \quad [2]$$

the force due to specular reflection as

$$\vec{F}_\rho = -\rho \frac{S}{c} 2A (\cos \theta)^2 \vec{N} \quad [3]$$

and the force on the tile due to diffuse reflection as

$$\vec{F}_\delta = -\delta \frac{S}{c} A \cos \theta \left(\vec{s} + \frac{2}{3} \vec{N} \right), \quad [4]$$

whereas \vec{N} is the normal direction of the surface as shown in the diagram, \vec{s} is the vector towards (!) the sun and A is the area of the tile (not the projected one). From the equations above follows that the total force on the tile is:

$$\vec{F} = -\frac{S}{c} A \cos \theta \left\{ (1 - \rho) \vec{s} + 2 \left(\frac{\delta}{3} + \cos \theta \right) \vec{N} \right\} \quad [5]$$

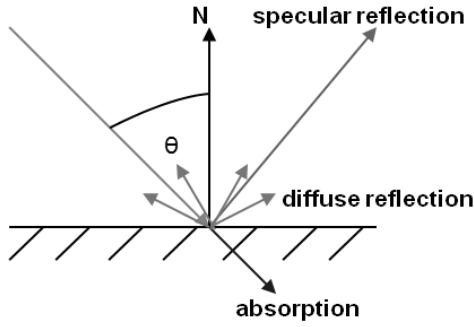


Fig. 9: Light reflection and absorption at the surface of the vehicle

Since materials typically have different reflection characteristics depending on the wavelength, the solar flux is split into three components: one component for 1) red light and longer wavelengths, 2) shorter than red and longer than blue wavelengths and 3) blue light and shorter wavelengths. Corresponding coefficients α , ρ and δ are provided in the material file of the 3D vehicle model for each of these channels and for each material.

The calculation of the atmospheric drag force can be considered as analogue to the calculation of the force due to specularly reflected sun light (elastic impacts):

$$\vec{F}_d = -2\rho v^2 A (\cos \theta)^2 \vec{N} \quad [6]$$

This assumption is only valid in the free molecular flow regime why it cannot be used for re-entry scenarios.

Fig. 10 shows a visualization of differential forces and torques due to air drag and solar radiation as it is produced by the visualization module of ASTOS. The picture shows the DEOS client and servicer.

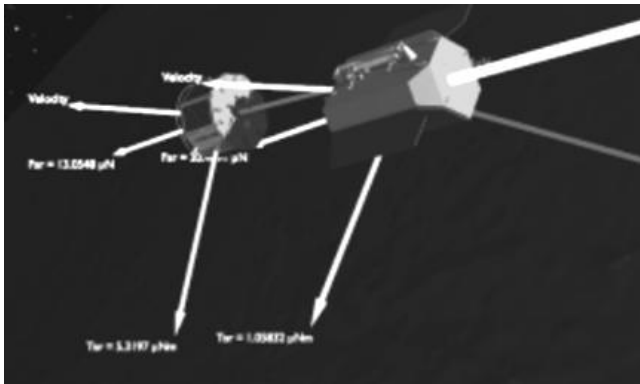


Fig. 10: Differential torques and forces in the DEOS scenario

V. ROBOTIC ARM INTERFACE

For rendezvous scenarios where contact is made by berthing and not docking, the simulation of a robotic arm can become essential. This additional component requires the propagation of multi-body dynamics considering the

control of the robotic arm. This control can be either simple inverse kinematics or a more sophisticated algorithm implemented e.g. in MATLAB/Simulink. For such a model ASTOS comprises an external interface.

As simplification, instead of the hinge moments the hinge angular velocities are controlled and it is assumed that the hinge motor control can follow these given angular velocities. This simplification is mandatory for the definition of a generic robotic arm interface. The definition of the robotic arm then consists only of the following parameters:

- robotic arm geometry (number of hinges/elements, hinge axes orientation, length of arm elements)
- mass of each robotic arm element
- inertia matrix of each robotic arm element

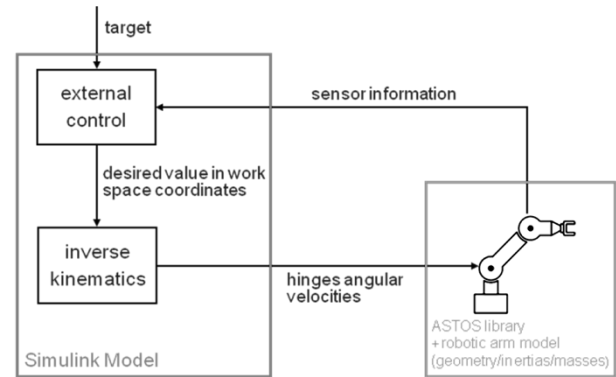


Fig. 11: Architecture of the robotic arm simulation

With the given information the propagation of the multi-body dynamics can be solely performed within ASTOS and need not to be part of the user-supplied model.

Via the user interface (ASTOS GUI) it is also possible to specify 3D model files for each robotic arm component (as well as for the satellite core). Thus animations that show the motion of the arm in a realistic manner can be provided to the user.

Also the camera simulation can make use of the robotic arm model and can visualize its motion. On the other hand it is also possible to place the camera or light sources onto the robotic arm and to incorporate both camera simulation and robotic arm controller model into a closed-loop simulation.

VI. LINK BUDGET ANALYSIS

ASTOS comprises a link budget analysis module that can determine the signal to noise ratio as well as the link budget between a satellite and a ground station or two satellites. It considers the antenna characteristics (e.g. geometry, polarization, bandwidth, beam width) of receiver and transmitter as well as atmospheric effects e.g. due to water vapor, rain or clouds. A pointing algorithm for the antennas can be optionally applied.

Fig. 12 shows a typical output of the link budget analysis.

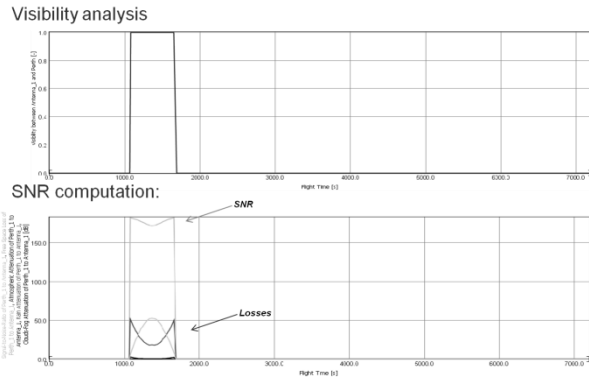


Fig. 12: Typical output of the link budget analysis

VII. RESULT PRESENTATION AND VISUALISATION

ASTOS comes with a variety of post-analysis capabilities that allow the presentation and exploitation of the obtained results without the need of other tools. Those capabilities are described below.

Animations

Besides the camera simulation ASTOS provides an animation window that allows realistic presentation of the scenario (Fig. 15). This realistic scenario can be overlaid with mission analysis data like for example

- trajectories
- sensor field-of-view
- uncertainty ellipses
- direction, velocity, force and momentum vectors
- points and areas of interest
- alternative textures for celestial bodies (e.g. population density)

Likewise, the ASTOS animation window is based on the VESTA graphics engine [2] and therefore the animation provides the same qualitative characteristics as the camera simulator:

- rendering of shadows
- rendering of reflections
- consideration of (artificial) light sources
- realistic atmospheric scattering
- realistic surface characteristics

If the user does not specify a 3D visualization model for the satellite(s), ASTOS is able to create a simplified model based on the user input for the mission scenario.

It is also possible to combine automatically created 3D models with user-defined 3D models on assembly and component level.

The animation window provides controls for the observer point, the field-of-view and the animation speed. It provides further the possibility to export the animation into video using the installed video codec.

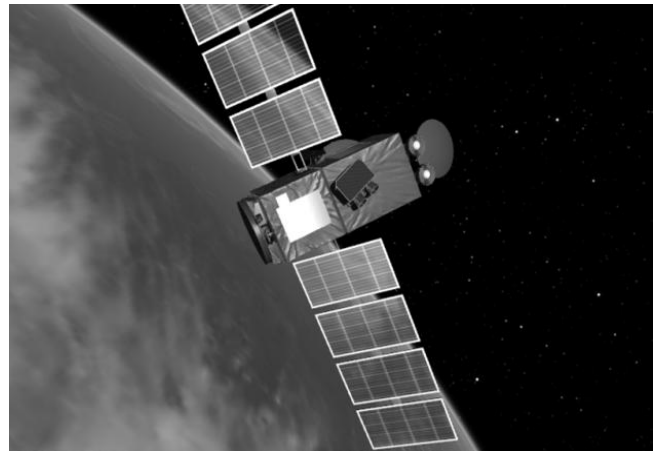


Fig. 13: Reflection of the Earth in the solar panels

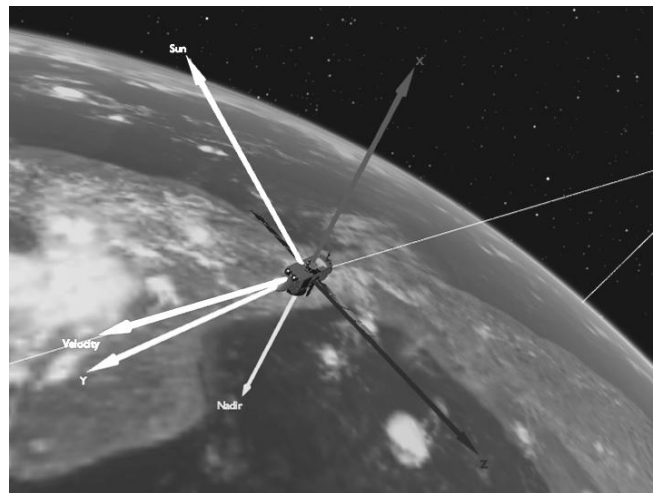


Fig. 14: Display of vectors and coordinate frames

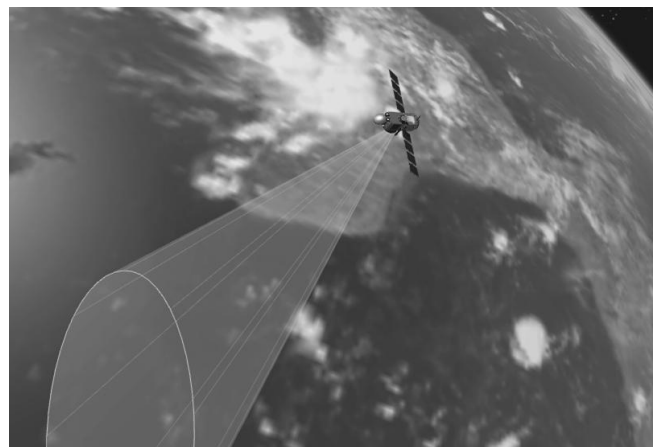


Fig. 15: Displayed lobe of a simulated sensor

Plotting

ASTOS 8 provides internal plotting capabilities with a variety of plot types. These plots can be used to plot simulation and analysis results produced by ASTOS and it's also possible to import data produced by other

software. The following plot types are available (see also Fig. 16):

- 2D plots
- 3D plots
- 2D map plots
- 3D map plots
- Condition plots
- perspective 3D view

The map plots provide an interface for GeoTIFF and ArcGIS shape files such that GIS layers can be displayed. The user can select between several cylindrical and spherical projections.

With the schedule plots it is possible to evaluate certain conditions along the simulation time. Using a build-in editor the user specifies a Boolean expression that may contain time-dependent function values (that are calculated during the simulation), arithmetic operators, logical operators and constants. From that expression a bar graph is produced that is colored red whenever the expression is false and green when it is true (see Fig. 17). Several of these Boolean expressions can be combined in one condition plot.

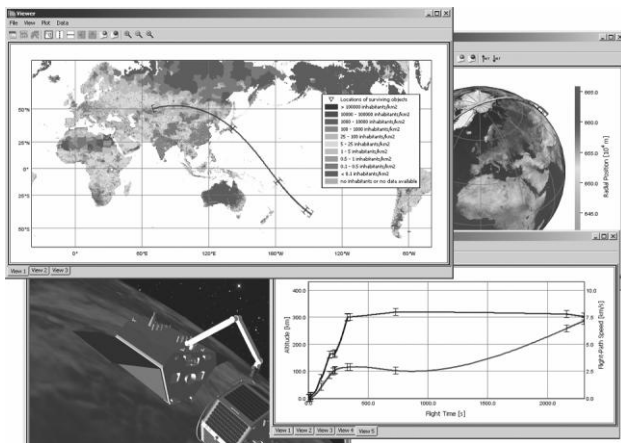


Fig. 16: ASTOS plotting windows

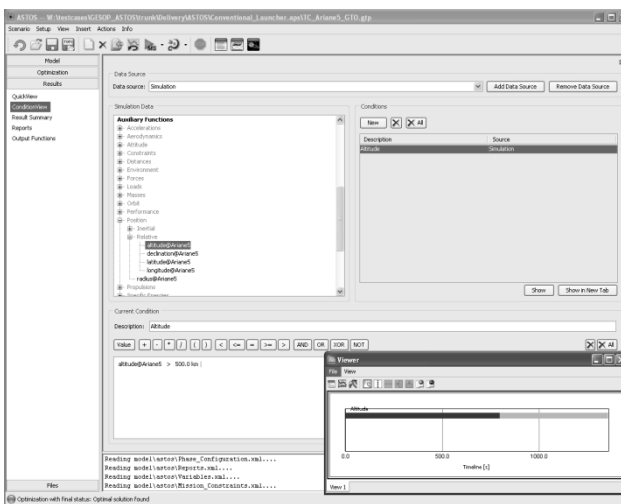


Fig. 17: ASTOS condition plot and condition editor

Reporting

ASTOS 8 comprises a user interface for automatic report generation. Via a template-based architecture the user can create flexible reports that will comply with design and structural requirements defined by applicable standards or company/agency guidelines.

The user can insert simulation and analyses results as scalar values, tabular data or plots (see Fig. 18). The user can define if the displayed data shall be automatically updated as soon a simulation finished or if this shall be done only on user action. Generic fields for e.g. for simulation date and time or the names of the underlying simulation file are also available and version control is supported as well. The structure of a report consists of header content and footer. Header and footer can be defined on installation level, on user level or case-wise.

The content can be fully user-defined but on the other hand every analysis module of ASTOS offers a predefined reporting content that is automatically adapted to the users CI by means of the header/footer template technique.

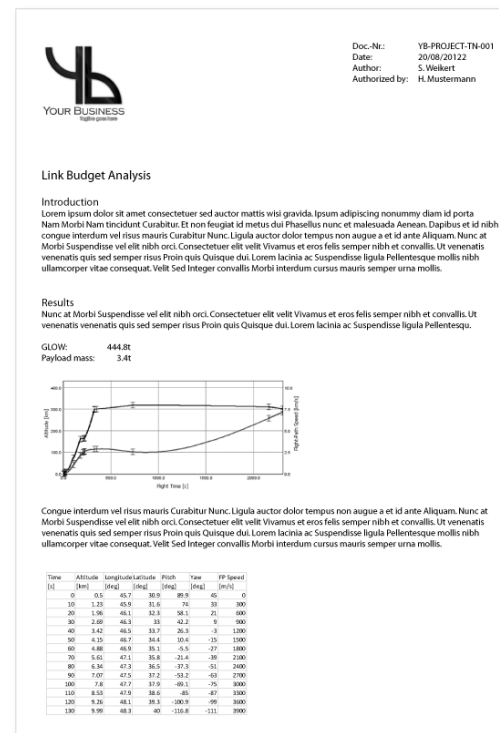


Fig. 18: Example of an automatically created report with customized header, plot and tabular data

VIII. CONCLUSIONS

The paper showed the new functionalities of ASTOS 8. These new features make ASTOS suitable for the analysis of rendezvous and robotic scenarios. Especially such multi-vehicle scenarios can benefit from the coupled GNC/mission analysis increasing confidence in the analysis results and reducing the risk of cost overrun.

The flexible post-processing capabilities (plotting, animations, automatic report generation and export functionalities) reduce the work load of the user and make him independent from other post-processing tools.

IX. REFERENCES

- [1] D. Reintsema, B.Sommer, J. Thaeter, W. Naumann, J. Sommer, A. Rathke, P. Rank, "DEOS – German's Robotic Agent Concept to Service, Secure and De-Orbit Malfunctioned Satellites From Orbit", 62nd International Astronautical Congress 2011, IAC-11-D1.1.2-x10954
- [2] S. Weikert, C. Laurel, "VESTA - Virtual Environment for Space and Terrestrial Applications, an open source 3D graphics engine", 4th International Conference on Astrodynamics Tools and Techniques, ESA/ESAC, Spain, May 2010
- [3] A. Wiegand, S. Weikert, "Concept and Performance Simulation with ASTOS", 12th International Workshop on Simulation & EGSE facilities for Space Programmes (SESP 2012), ESTEC, Sep 2012
- [4] A. Wiegand, M. Sudars, M. Dumontel, A. Helmersson, "Definition of the LGSST framework", ASTOS-LGSST-TN-001, 2012