Industrialization: D-Orbit's experience with the AOCP

Anastasia Osik S. (1), Emanuele Paolini (2), Giovanni Corinaldesi (3), Pietro Bernardini (4)

(1)D-Orbit, Viale Risorgimento, 57 22073 Fino Mornasco (CO) Italy, +34 674437931, anastasia.osik@dorbit.space
(2)D-Orbit, Viale Risorgimento, 57 22073 Fino Mornasco (CO) Italy, +39 3404712801, emapaolini@gmail.com
(3)D-Orbit, Viale Risorgimento, 57 22073 Fino Mornasco (CO) Italy, +39 3277771398, giovanni.corinaldesi@dorbit.space
(4)D-Orbit, Viale Risorgimento, 57 22073 Fino Mornasco (CO) Italy, +39 3421034179, pb474@cornell.edu

ABSTRACT

The growing number of programs and projects managed in parallel by D-Orbit made it necessary to aim towards an industrialized approach of the development and testing of the AOCS software. For this purpose, a software management system has been implemented that enabled flexibility and configurability adapting to existing and upcoming mission needs. Fitting with the growth of the missions an industrialization process had to be created allowing to automatize processes and reduce possible errors. The AOCS SW architecture aims towards a high configurability in line with the adaptation to multiple missions.

1 INTRODUCTION

In the last decade, the space business is maturing from being strongly customized for each specific mission to being more industrial, following what other more mature industrial areas have done in the past, for example automotive. On one side there is the need to reduce costs to access space, so lean philosophies are seen as a good approach; on the other side, such approaches need large numbers to be effective. The need to have standard products with an increasing number of missions and being able to manage them in a proper way is strong, becoming unfeasible and inconvenient to customize each single platform or subsystem when producing a high quantity of ready-to-fly spacecrafts per year.

When numbers are sufficiently high, a more industrialized approach becomes feasible, convenient, and commercially viable. This has been the case in D-Orbit in recent years, with an exponential increase in ION missions to carry CubeSats in space and release them in the appropriate orbits, several other Earth Observation programs and In-Orbit Servicing.

The growing number of programs the AOCS & GNC team was involved in led to the necessity to coordinate better the ongoing AOCS developments establishing common guidance. Thus, the project for a common AOCS baseline architecture started receiving the name of AOCS Platform (AOCP), aiming at achieving a sufficiently flexible and modular architecture, overcoming mission-specific changes.

Thanks to this project, D-Orbit currently does not require much recurrent effort for each mission, also allowing a certain degree of standardization and automatization in the V&V process, ultimately improving the reliability of the AOCS system.

Special attention has been put in defining a proper framework for configuration management, version control and issue defect management data. Indeed, the ability to base every project, with different requirements, on the same SW is done through versioning. Each SW version carries new modifications, bug-fixes, and improvements with respect to the previous one, but also allows flexibility in the avionics that can be used. This way, every one of them can be further developed and maintained in parallel. The chain of features or bugfixes that are implemented creating a new SW version can be posteriorly uploaded in a flying ION mission allowing to demonstrate capabilities in a very fast way and validate in-flight. This grants the possibility of minimizing the time between any change of the SW that is requested and the corresponding in-flight heritage.

Different ION versions carry different SW versions, creating a range of configurable products that adapt to each client's mission, in the same approach as done in the automotive industry. Each product created is a set configuration of the AOCP with fixed customizable avionics: number of actuating devices, sensors, number of star trackers, presence of gyros, etc. This provides the possibility of having more economical products that achieve different performances tailored to each client and adapted to each mission demand: with higher/lower pointing accuracies, associated to higher/lower costs, duration, or payload type, among others.

The flexibility and modularity of the AOCS is such to guarantee long-term industrial benefits, as it has proven to be expendable also on different R&D missions having peculiar technical specifications. The industrialization of the AOCP has proven to be successful to the point that the AOCS team's involvement in the missionization process is negligible and the effective mission-adjustments are carried out by non-specialists of the ION fleet team independently. Such automatic processes also include the calibration of the sensors, carried out before each ION launch automatically.

2 AOCS ARCHITECTURE

The AOCS SW is organized in functions that are divided among several boards in a distributed architecture. The functions can be grouped in:

- Guidance function, which outputs the desired state.
- Navigation function, which outputs the estimated state.
- Control function, which implements the different control strategies.
- Fault Detection, Isolation, and Recovery (FDIR) function, which is in charge of detecting the presence of failures ('Detection'), of identifying the source of this failure ('Isolation') and of performing the actions required to recover from the failure and restore, if possible, the nominal conditions ('Recovery'). One of the main FDIR sub-functions is Data Validation, that checks the data of sensors and actuators and marks them as valid/invalid.

2.1 Hardware

In the next section, the equipment used in the AOCS platform is presented. The sensors and actuators that could be integrated and used to meet customer specifications:

- D-Senses as coarse attitude sensors; up to eight of these can be used. They are manufactured in-house and consist in a fusion of different sensors (sun sensors, magnetometers and gyroscope) which provide a coarse navigation solution.

- Star Trackers as optional fine attitude sensors, it is possible to use up to three STs depending on the objectives of each mission.
- GNSS receivers for orbital determination; two can be interfaced: a coarse one with 100m range accuracy and a much more precise one, with up to 10cm of accuracy.
- Magnetic Torquers; up to four can be mounted.
- Reaction Wheels (RW) with the possibility of having up to four mounted.
- A configurable number of thrusters, depending on the mission demands.
- It is also possible to add a high performing FOG gyroscope for more demanding missions.

The interactions between the functions and the hardware can be viewed in the following figure.

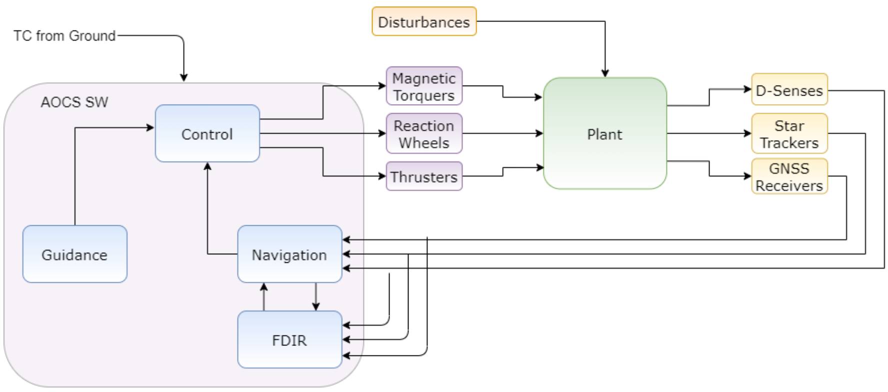

Figure 1 AOCS Architecture

The architecture such that the SW can be easily configured and adapted to handle any mission requirements by simply changing a set of key parameters and flags that allow to use the set of sensors and actuators that provide the needed performance.

<u>D-Sense</u>

As previously explained, the D-Sense is constituted by a combination of sensors that allow computing a coarse Navigation solution. The position of these sensors inside ION is done in a redundant way and maximizing the coverage.

It is a sensor suite based on a rigid-flex PCB whose main purpose is to determine the attitude of the satellite by the combination of sun sensors, magnetometers, and gyroscopes with the possibility of also including a star tracker. It is possible to customize the final product, mounting or dismounting the components:

- Six photodiodes (CSS) installed with different orientation between them.
- A single magnetometer (MGM) is composed of two Sen-XY, a Sen-Z-f sensor coil and a MagI2C ASIC controller that features both continuous and single measurement on the three axis components of the magnetic field.
- A three axial MEMS gyroscope designed for low power, high precision multi-axis application.

2.2 AOCS modes

Each of the available AOCS modes is aimed to perform a specific task with a specific AOCS configuration. The baseline modes are:

- **IDLE:** it is used after launcher separation during the deployment of the solar arrays (when applicable), to let AOCS be passive during the deployment.
- **DSPM:** the goal of this mode is to damp the angular velocities and point the solar panels

towards the Sun as soon as possible.

- **CRU:** this mode foresees a zero-momentum, three-axis attitude control with the use of reaction wheels and MTQ actively avoiding wheels saturation. The purpose of this control mode is to maximize power income pointing the solar array towards the Sun.
- **TPM:** during this mode the AOCS uses the best set of available sensors for fine attitude determination and control. It performs a maneuver to the desired targets, keeps pointing and finally returns to a Sun-pointing attitude.
- **OCM:** during this mode the satellite attitude is maneuvered about 3 axes to point the thrusters to the required direction. It performs accurate control and compensates for the disturbances caused by thrusters firing. It is a three-axis attitude control: depending on the mission and the size of thrusters, it can make active use of the wheels or off-modulate thrusters to aid compensation of parasitic torques.
- **CAL:** This mode allows Ground operators to command reaction wheels and MTQs in open loop so it can only be entered through a safe procedure. The principal aim of this state is to allow a safe environment for Ground checks and tests on actuators, possibly following an equipment malfunction.

The following State Machine configures the AOCS modes by either performing automatic or manual mode transitions.

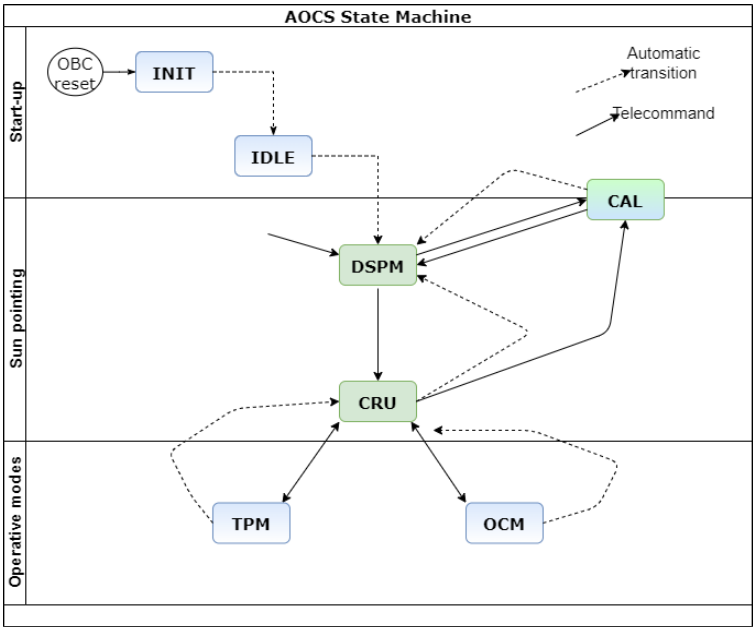

Figure 2 State machine diagram

2.3 Software

In this section, the main functions of the AOCS will be introduced: Navigation, Guidance and Control [5].

Navigation

With the suite of sensors previously described, it is possible to compute, using different strategies, a navigation solution (attitude and rate estimation) that is selected by the Navigation Manager depending on the S/C mode and the available sensor data. Each of the sensors are subjected to a Data Validation, where different checks on the data are performed together by the system software and the AOCS software. This way, it is possible to provide as output the validity of the data so that the Navigation Manager can select the most accurate strategy with the available resources.

In the case of the D-Sense, through the management of flags for each sensor it is possible to remove either a complete D-Sense, a specific sensor, or all sensors of a type from the Navigation loop not including their contribution to the computation of the Navigation solution.

The first selection is either coarse or fine estimation according to the availability of a ST that provides valid data. Attitude estimation can be computed through the following methods:

- **TRIAD algorithm:** this function uses only MGM and CSS available on-board.
- **Fine Attitude Estimation:**
  - The Multiplicative Extended Kalman Filter (MEKF) with two configurations, increasing the flexibility of the AOCS SW [3]:
    - Gyroless.
    - Gyro-aided, with the possibility of adding the optional FOG gyro. So that a more precise and stable rate estimation is achieved, with particular reference to optical missions or those missions focusing on stability and high performance.
  - **Star Tackers Fusion:** a direct quaternion average of the available STs.

For what concerns the rate estimation, there are two coarse methods mainly used during safe modes and when the angular rate of the spacecraft is very high, which are:

- **Coarse rate estimator:** this function makes use of only MGM and CSS available on-board and degrades during eclipses.
- **Gyroscopes raw average:** this function requires GYR to be available. Therefore, due to the bias affecting the measurements, it cannot be considered to be safe when the rate is below a threshold.

Whenever the spacecraft exits the Detumbling and sun pointing mode, a finer algorithm can be applied for the rate bias estimation. In particular, the following two are available on-board:

- **Complementary filter:** this function makes use of the GNSS receiver and strongly degrades during GNSS outages and eclipses (if not fed with ST quaternion).
- **Kalman Filter:** this function makes use of only the sensors available inside the D-Senses, hence strongly recommended when ST is not used.

Different AOCS modes have different Navigation strategies, which also, and mainly, depend on the HW configuration of the mission. Given a HW configuration, it is easy to select the different strategies through a few parameters; these flags will be then eventually automatically over-ridden on board based on availability of data as marked from Data Validation, eventually switching to back-up navigation functions if needed.

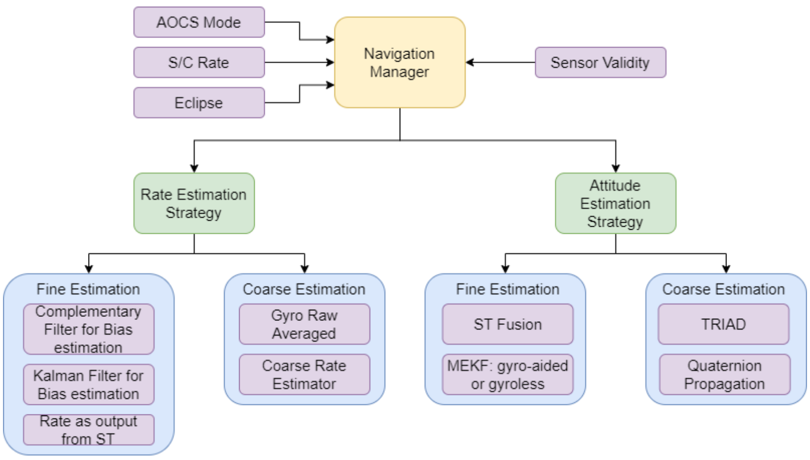

Figure 3 Navigation strategies

Guidance

The interface between the Guidance and Control is fixed and bounded so that it is possible to add new Guidance algorithms without having the need to adapt the output increasing the flexibility.

The Guidance algorithms can be divided into targeting and manoeuvring, the Guidance Manager selects whether it is necessary to enable the manoeuvring functions based on the output of the targeting algorithms. Hence, when the difference between our current state and the targeted one is out of the threshold (that can be modified) the Guidance outputs a manoeuvre to avoid having abrupt changes in the Control input.

The targeting Guidance can be selected from:

- **No Guidance:** null output, used during CAL, IDLE and DSPM modes.
- **TAP:** two-axis pointing. It is used for cruise mode as well as for any other targeting required. Setting four parameters, it is possible to select which axes to point and towards which targets, giving main priority to what is defined as the "target" and secondary priority to what is called "co-target", i.e. the co-target is maximized while the target is pointed. Through this algorithm is possible to also generate a guidance to track a fixed point on Ground., like a Ground Station during a passage for communication purpose.
- **Inertial (QAP):** the target is a quaternion reference.
- **Ground-based:** it is possible to load a quaternion, rate and acceleration profiles to be followed or a profile through Chebyshev coefficients. This provides limitless possibilities to the guidance profiles that can be used during operations, obviously within the physical constraints of the performance of the platform.

The manoeuvring Guidance strategies are either bang-bang or minimum jerk. The bang-bang is a time-optimal profile around the Euler axis, thus covering the shortest angular path from the starting attitude to the desired one. This manoeuvre maintains the maximum available acceleration during the maximum time possible, so it produces a high variation in the acceleration passing through maximum to minimum inducing loads on the S/C structure or any flexible appendages available. To minimize these effects, the minimum jerk function was introduced, bounding the system’s jerk.

It is possible to select the baseline slew profile (bang-bang or minimum-jerk) through a flag, making it easily adaptable to missions with flexible appendages that need to not excite vibration modes, as well as to missions that are more interested in just minimizing the time of the slew [1] [2].

Control

For each of the modes stated hereabove a different control strategy is used, resulting in [5]:

Detumbling and Sun pointing controller.

The principal aim is to dump satellite angular velocity and to point the solar panels towards the Sun. The controller is designed to be as simple and reliable as possible: for this reason, CSS and MGM are the only needed sensors, while MTQs are the only needed actuators. It is composed of two sub-controllers, so that the first one has the goal of reducing the rates and once below a tuneable threshold for a tuneable duration, an autonomous transition to the Sun pointing sub-controller is performed.

The flexibility and configurability of the system is present in this mode through parameters that allow selecting the axis to point towards the Sun (and thus the ones around which to dump angular rates), the presence and amplitude of the spin during Sun pointing for gyroscopic stability, as well as the thresholds to pass from one sub-mode to the other. In wider terms, this mode is a PD controller that can be used with the input of an angular rate, and both the coarse rate estimation and the gyro data could be used.

CRU controller.
It is a 3-axes, near-zero-momentum controller which is aimed at achieving and maintaining a three-axes stabilized Sun pointing attitude. A finer set of sensors and actuators is used in this mode, involving Star Trackers and reaction wheels. Reaction wheel de-saturation is performed continuously and not on-demand, thanks to the action of the Magnetic Torquers. GNSS data is used to compute the needed 3-axial guidance.

The axis to be pointed towards the Sun can be chosen by selecting a parameter, and as mentioned in the Guidance section, a possible co-target can be selected in order to fulfill varied scopes: maximizing contact of an antenna to Ground, phasing a radiator opposite to Earth, etc. Moreover, the controller chain features a filter block after the PID, to be tuned according to the mission needs. Especially useful to shape the overall control frequency response in those missions where flexible appendages are present, and thus, their first resonance frequency should be carefully considered.

TPM controller.
It is a 3-axes, near-zero-momentum controller which is aimed at allowing the spacecraft to perform payload operations in the best possible conditions. The finer set of sensors and actuators is used in this mode, involving Star Trackers in missions where they are available, and reaction wheels. Magnetorquers are not used in order to not decrease the pointing accuracy, thus this mode cannot provide RW de-saturation and can be kept only for a limited time before returning to CRU for momentum management. GNSS data is used to compute the needed 3-axial guidance.

This controller features a flexible configuration with tunable filtering as explained for CRU. The related Guidance allows to follow whatever target is needed during the mission, making use of the TAP guidance and the possibility of loading guidance profiles from Ground. The development of a new guidance function able to point a moving object in another orbit is planned in the framework of IOS-related features, providing even more possibilities.

3 WORKING PRINCIPLES

The AOCS SW has been evolving and adapting to fit the system requirements. The flight heritage of the first flown SW version has provided the AOCS team with a lot of feedback on improvements. In addition, there are new requirements that are inherited from the parallel projects that D-Orbit is working on. This means that all these requirements must be handled and collected in a way that can be categorized and divided into future versions of the SW, developed with an incremental approach to add new capabilities step by step following the timings of the development.

The new developments are based on the previous version, so that each version is compliant with the requirements of the previous ones. Since all versions are flown or will be at that matter, it is crucial to have the ability to maintain all versions.

3.1 Repository structure

All D-Orbit's AOCS SW is handled in one single repository, and it shall be able to handle missions, ION flights (SCV) and the main development of the AOCP. There are two additional repositories, one to handle all related to simulations and the simulation environment (SYS), and a second one to handle common and necessary functions and algorithms (CORE). The main simulation environment is generated using the SYS repository, it allows carrying out simulations and configuring missions, where both the CORE and SW repositories are treated as submodules in Git allowing in this way to use their code.

SW repository

The flight code is maintained and developed in the SW repository. The AOCP project is in charge of carrying out the required developments for the general SW. The repository works considering:

- Each Avionics version has a Master and a develop branch.
- Each development program/product has a develop branch (AOCP, MSRN, DSNS2...).
- Features and issues are tackled in the classic way of branching from the program development that required the feature/issue, and then merge again into the same develop branch.

All the develop branches merge in the AVH develop branch corresponding to the AVH version they are developing for. Programs develop branches never merge directly into AVH Master branches.
The AVH develop is merged into the corresponding AVH Master when a new official release is needed. The program/product Roadmap is appointed as the only source of information regarding which is the AVH version that program/product is developing for.

Each AOCP branch in the repository is intended for the development of new features or bug-fixes and is merged into the correspondent Master branch at the end of each version. When a new development starts for a new version of Avionics, it will have a corresponding Master and develop branch. Each program is linked to a version of the AOCP, if the version is not yet available (but it is in the AOCP plan) the program is linked to the AOCP develop of that version. Instead, if a version with those new features is not yet planned, then it is foreseen as a new AOCP version for the future, in the pipeline. Each of the features and bugfixes are done in branches that are created from the develop and posteriorly merged. If the new development is instead for an existing version of Avionics (therefore a released version), the fixes are carried out from branches of the develop Avionic branch.

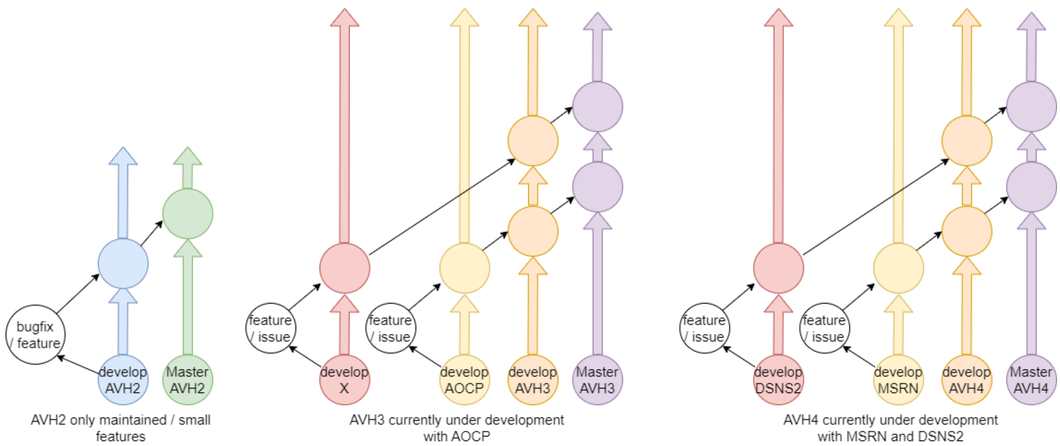

Figure 4 SW repository distribution

When an existing program or project is developed in parallel to the AOCP, the modifications are merged to the corresponding avionics version.

SYS repository

As previously mentioned, the SYS repository creates the necessary simulation environment and the correspondent mission configuration. Each of the SYS repositories contains the necessary mission parameters and configuration. There shall be a Master branch for each of the ongoing AOCS projects linked to one of the versions of the Master SW of an AOCP release. However, the probable scenario is that these projects are going to be developed in parallel to the correspondent AOCP version, so that the Master branch of each project is going to be based on the develop branch instead. On the contrary, each of the branches corresponding to an SCV mission ought to be linked to an already released version of the AOCP SW.

For the development of the AOCP SW a simulation environment is needed for tests and analyses, so there shall be a develop branch in the SYS repository linked to the develop branch of the AOCP version that is in development. Lastly, to be able to quickly simulate and view the performances of each of the versions of the SW release, there shall be a branch associated to each official release of the AOCP with the corresponding set of Avionic Hardware used and nominal mission configuration.

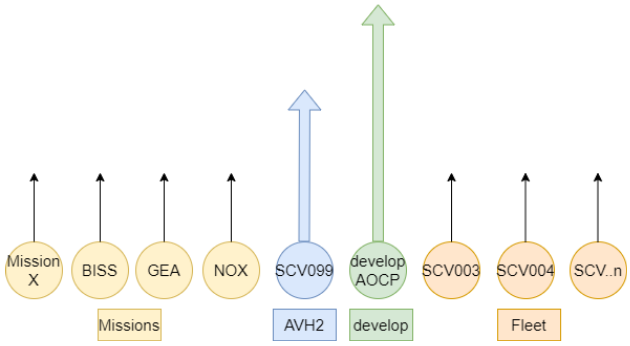

Figure 5 SYS repository distribution

Core repository

This repository works as the SW one, with a few simplifications. For each set of Avionics there's a Master and a develop branch, where the features and fixes are merged directly into the develop branch.

3.2 Interaction with ION missions (SCV)

The development of each of the SW versions, compatible with the corresponding Avionics, is done so that it can be integrated into the SCV missions that are currently in-orbit or will be ready for flight soon. This means that code maintenance for each version shall be needed until there is no ION operative with the corresponding SW version. This system of branching allows exactly this, it is possible to have in parallel different versions that can be modified and tested, with the corresponding V&V processes which will be explained further on and their corresponding industrialization.

Once an AOCP SW version is released, passing all the V&V steps, now it can be used by SW for integration into ION's SW. The initial flexibility of the AOCP SW release means that it needs to be configured and adapted to each specific mission, and this shall be done for each of the code releases. To handle this configuration in a more systematic and secure way a process had to be designed that allowed integrating all the required inputs for the preparation of the code for flight.

Once the AOCS SW version is set, it is necessary to specify the specific parameters of the mission in a SW configuration file, along with the calibration of the sensors and actuators and the inertia measurements. Hence, once it is configured to the needs of the mission, a Monte Carlo campaign is performed so that it is possible to verify the performance and, in case of existence, view the incompatibilities and take the required actions to solve them. Having different SW versions implies differences in the integration and verification tests performed by the AIV team during the launch campaigns.

4 CONFIGURABILITY

The evolution of the AOCP SW has been triggered by in-flight heritage, requirements for new missions and developments done for other projects and programs. This leads to an incremental improvement of the SW capabilities that can be exploited by the SCV missions, because of its configurability and flexibility. It is interesting to view how the repository structure and branching system, described in the previous section, intertwines, and helps in achieving the configurability that allows having so many different missions flying with success.

4.1 SYS configurability

The SYS repository holds the simulation environment, which has been designed with a key structure to allow for flexibility. Overall, it handles the models and functions that create the environment in which the simulation will be held. Furthermore, it is composed of the database of the main configurable data, which are: sensors, actuators, physical and structural characteristics, and inertia data. It holds the mission specific parameters and initial settings of the SW and SYS environments. It can be easily configured to be adapted to different sensor and actuator configurations.

Since the AOCP is used for different missions and projects, it needs to be able to hold different environment options. This leads to having the possibility to adapt and change certain aspects that are, among others, the following:

- Sensors mounted.
- Number of active sensors and actuators.
- Different hardware (HW) options.
- Inertia and mass properties.
- Structural and physical data.

It is possible to modify the number of sensors or actuators that are active and available using different approaches. For example, for star trackers and reaction wheels, the nominal scenario allows up to 3 for the former and 4 for the latter, but this does not mean that this exact number must be boarded. It is possible to use a lower number of each thanks to the presence of algorithms that generate an output with the dimension of the set number of sensors/actuators in the simulation configuration. Also, it is possible that the nominal design accounts for a number of sensors that later it is not possible (due to diverse reasons) to mount. It is feasible to select the sensors and actuators that are mounted but are inactive, using the validity masks which remove from the loop of actuation/control or navigation the actuators or sensors that should be inactive. This could be done from ground, or from SW checks on the sensors/actuators output.

Provided that the interface of the model of the sensor and actuator is fixed, it is possible to mount different hardware using the same model and modifying the respective characteristics, maintaining same avionics. Since the SW is utilized in different missions and projects, it is possible that different HWs are mounted, as an example, the star tracker used in the ION program is the STHS1 ST from Ty Space while the one mounted in NOX project is from another manufacturer, and thus, has a different set of properties. Similarly happens with the thrusters that are mounted in ION, due to the in-house manufacturing of a new model, two different SCV missions with same AOCP SW are going to fly with different thrusters. This is possible since the interface for the model in the simulation environment is fixed and it is only a matter of changing the properties.

The ability of setting simulation specific data and testing different simulation scenarios led to the implementation of variant subsystems, that enable or disable specific parts of the simulator. This way it is possible to generate a larger variety of scenarios and test the SW accordingly. It is both an advantage for the simulation configuration and the implementation of the representative models of the characteristics of the system. Some of the applications of this can be found in enabling/disabling or varying environmental models such as: atmospheric, Earth pressure and albedo; or dynamic additions such as coupling and sloshing models. In terms of simulation configuration, it is possible to enable/disable the logging of data (or modifying the destination), choose different simulation exit options or changing the type of the simulation from a single scenario to multiple sequential ones.

This configuration can be done thanks to the system configuration files, that contain all the configurable parameters. As explained before, the simulation environment contains a database with all the HW options that can be considered for the AOCP SW and the selection is done through the configuration file, where it is possible to select the number of sensors/actuators but also the HW to be used. To summarize, the configuration for each mission can be done at these levels: SW, SYS, and simulation.

4.2 SW configurability

The AOCS SW version is designed for a nominal set of avionics but has a lot of flexibility in terms of Navigation algorithms or control/actuation management. Considering the AOCS architecture, the number of sensors and actuators can be modified with available adaptations in the SW. In a similar way as in the simulation environment, there are parts of the code that can be enabled or disabled to account for a difference in the avionics by means of configuration flags.

The navigation algorithms are implemented in a way that can be subjected to some flexibility; this means that the navigation solution is obtained through different approaches that use a different combination of sensors. Each of the available ADS can be excluded or included in the computation of the navigation solution. There could be scenarios where the gyroscope, or the magnetometer failed so the navigation solution has to be computed with the rest of the sensors by means of a change in the strategy used. This selection is done through the Navigation Manager that can choose the algorithm that has best performances with the available sensors.

In terms of configurability, considering the availability of different sensors such as star trackers (and quantity) or gyroscopes, it is possible to enable a fine attitude estimation that using the solution given by the star trackers through a Kalman filter provides the navigation estimation. The Kalman filter or alternatively the merged solution is dependent on the number of star trackers available and can compute a solution independently on it. The rate can be computed from the gyroscopes or from the merged rate of the star trackers, again considering their number. In this way, when something changes in the availability of both sensors a solution can still be provided either number of sensors are mounted (up to a maximum limit).

From the control/actuation side, through the actuation management system it is possible to update and use the actuators that are available recomputing the torque and thrust sent to each one of them. The matrix used in the rotation to the corresponding actuators is updated each time a failure occurs adapting to the remaining actuation available.

4.3 Adaptability of AOCP to different projects

The AOCP SW can be, and is, used for different programs and projects. To be able to do this, at SYS repository level, a new branch should be available for each of the different uses. So, through this branch (in the simulation environment) the mission can be configured with the options described above, adapting like so the SW parameters to the requirements of this mission/project. The configuration files make this process quite straightforward and provide the possibility of doing analysis and tests with this new setup.

For example, the set-up that is done for NOX (an ongoing project) can be divided into the following categories:

System Configuration

Change in the physical properties to be applied. This includes the use of variants to enable/disable the required blocks for NOX (as previously explained). These properties are:

- Mass and Inertia properties.
- Sloshing parameters.
- Flexible structure parameters.
- Solar panel related parameters.

Mission Configuration

Everything regarding the mission parameters is changed to adapt to the NOX mission. In this way, should be changed:

- Orbital parameters.
- Parameter files: parameters required for the SW functioning.

Simulator Configuration

Setting of the conditions for the simulation:

- Initial Conditions for the start of the simulation.
- Ground commands selected for the simulation scenario.
- Selection of the specific hardware parameters used.

SW Configuration

Adapt the Satellite configuration for NOX, selecting the hardware that should be used in this specific mission, so:

- Number of D-Sense and ST: in ION two ST is the nominal configuration while NOX requires three.
- Number of RW and Thruster.
- Configuration and orientation of Sensors.

5 V&V AND SW AUTOMATION OF PROCESSES

As new iterations of the software are made, new features, generalizations or applications might be added. Some of these might not be compliant to a specific mission, hence particular attention needs to be taken with the code versioning.

Each of the releases are subjected to proper versioning and indexing. The flight code will be generated from the SYS repository. The selected SYS branch is pointing towards an AOCS SW submodule that is going to be auto-coded, for flying purposes only Master AOCS SW version can be used. Therefore, for each mission it is always possible to understand clearly which SW version is under use [4].

Code generation is completely automatized using Jenkins. In Jenkins a "multi-branch pipeline" is created, building a pipeline for every branch present in the repository. Therefore, with an automatic build after change, Jenkins recognizes if a modification has been made in each branch of the repository and generates code, building it for each branch.

5.1 V&V

The process starts with a set of Monte Carlo (MC) simulations where there is a statistical validation of the performance and the compliance with the requirements (Model in the Loop testing). Once the development of a corresponding version is finished in the develop branch, then the SW code is subjected to the following Verification steps:

- SIL (Software In the Loop), to verify correctness of the generated code respect to the original Simulink model.
- PIL (Processor In the loop), to verify computational burden and correct functioning on the actual electronic board.
- SW test on the Engineering Model: to verify correctness of the interfaces and of the SW patch generation. Dry-run of in-orbit test procedure whenever possible.

To carry out the SIL/PIL tests a specific environment has been created, taking into consideration the organization and distribution of all the AOCS code on the boards. Hence, by running specific functions it is possible to obtain the input/outputs of each board and carry out the necessary comparisons. By adapting the input/output corresponding to each board the test can be run easily providing a very flexible environment. Once the V&V is passed the SW version is released and can be used for the requiring missions [6].

5.2 Industrialization

In order to reduce the need for AOCS involvement in the industrialization, several automatizations have been developed for the preparation process for each of the missions. These can be grouped into three categories: SW interface, calibration, and simulation campaigns.

SW Interface

In order to proceed with all the required testing of the industrialization process, it is necessary to create a SYS branch for each mission, pointing at the correct AOCS SW version. There are existing templates for the generation of this mission branch contained within the SYS repository.

There are available several input files that should be tailored to each of the missions and the corresponding variants. These files are easily changed without minimal involvement of the AOCS team, so that the SYS branch is built upon these files. It is possible to group the inputs by parameter tables, binary files, and calibration data.

Since these parameters can be modified by people external to AOCS, there is an input management check in the repository which, among others, includes a check on naming and dimension, so that if the parameters in the three files diverge from the parameters present in the repository, it commits the new ones overwriting the parameters' respective file.

Calibration

This step is responsible for the automatic calibration reports. The raw data from calibration done by Manufacturing has to be loaded into the repository. The calibration process carried out by the Manufacturing team is in itself automatic with the use of a robotic arm.

During the compilation process in Jenkins, it is possible to select through flags the execution of calibration procedures. Once the build is done, the output of the calibration will be present in a PDF file for each of the sensors. This process is totally automated and will be carried out by the team in charge of preparing the mission.

Simulation Campaigns

Once each mission is defined, the team in charge needs to configure all the required input files and complete the simulation campaign in order to re-test the performances. The output is an automatic report that decreases the time for the assessment. There are strict procedures on how to carry out these MC simulations with different scenarios under test, representative of the ones that will happen during orbiting. All the S/C modes are tested with different failure conditions, to ensure the behaviour is the expected.

Each MC can be configured by means of parameters and flags that will remain the same for the complete simulation. The proposed architecture foresees first the definition of baseline orbit and satellite mass configuration, that will remain the same for all the simulations of a given scenario, then the scattering for the relevant parameters, that will vary from simulation to simulation. The MC environment has been adapted to ensure its fast and reliable execution, being in the order of hours.

6 EXPERIENCE BASED DATA

The improvements done to the handling of the AOCS SW in D-Orbit, has clear influence over the SCV missions that have flown over the past years. There is a tendency of providing a usable flight SW in less amount of time.

The AOCS SW has evolved from version zero to the current version in development with is version four, as well as the Avionics have experienced an evolution over time mainly driven by new projects or by needed improvements. Furthermore, this structure makes easy to accept or study the feasibility of any new program/project as it was done with NOX, GEA or IOS. This is thanks to the high flexibility in the configuration of the AOCS SW that is able to handle diverse mission demands: flexible appendages like solar panels and high pointing requirements in the case of NOX, or cutting-edge robust algorithms that can efficiently address close-proximity operations, capture and rendezvous, without relying on expensive hardware, in case of IOS.

As well, the industrialization processes provide highly competitive times to market. The organization and management in the AOCS SW development, allows decreasing the time between one version and another.

With the improvements in the verification and validation process, through the creation of specific environments for the SIL/PIL scenarios easily adapted between versions, reduce highly the time required between end of SW developments and AOCS SW flying. From an industrial point of view, for each of the SCV missions, once the System Module is completed, there is a decrease in the AOCS involvement in the rest of the processes leading to the launch campaign. This is possible thanks to the industrialization automatizations that have been implemented by the AOCS team.

At the end of 2023, it is expected to have up to SCV015 flying with around fifteen different S/C in orbit and three different versions of AOCS SW. The number of SCV missions has increased having very different objectives each one of them. This translated into having each flight SW ready for very different missions both in terms of orbit manoeuvring, pointing capabilities or types of payload and Inertia changes.

At the end of 2024, it is expected to have four versions of AOCS SW in orbit. Furthermore, by the end of 2024 or beginning of 2025 NOX should be deployed and by 2026 GEA should be flying.

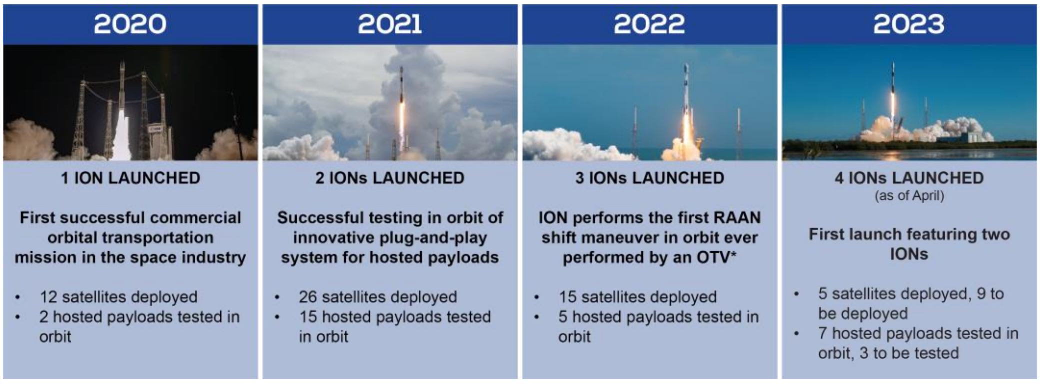

Figure 6 Evolution of ION launches

The effort does not stop here, in the next years, missions such as NOX (2024-2025) and GEA (2026) will be ready for flight, so that the maintenance of the SW will have to encapsulate different versions while developing new ones, which will be challenging for the AOCS team. Nonetheless, the path that is being constructed with the creation of organized processes to build, maintain, and improve SW with addition of the heavily automatized V&V and industrialization, provide defined methods that will allow reducing errors and increasing the performance.

7 REFERENCES

[1] Thompson, A., *Minimum Jerk Attitude Slew Maneuver*, Orbit Book Company, AIAA SciTech Forum, 2020.

[2] Flash, T., and Hogan, N., *The Coordination of Arm Movements: An Experimentally Confirmed Mathematical Model*, The Journal of Neuroscience, Vol. 5, No. 7, July 1985.

[3] Zanetti R and Bishop R.H., *Kalman Filters with Uncompensated Biases*, Journal of Guidance, Control and Dynamics, Vol. 35, 2012.

[4] Bourdon J., *Industrial and optimized auto-coding process for AOCS SW development in CD phase*, 2016.

[5] Markley F.L. and Crassidis J.L., *Fundamentals of Spacecraft Attitude Determination and Control*, Space Technology Library, 2014.

[6] Lafontaine J. et al., *PROBA-2: AOCS Software Validation Process and Critical Results*, 7th International ESA Conference on Guidance Navigation and Control Systems, 2008.