Aerospace GNC

Advanced Simulation and Testing of GNC Algorithms

A.A. 2023-2024

Andrea Colagrossi

POLITECNICO MILANO 1863

Aerospace GNC

GNC is not only an aerospace concept, but it is a human activity since the beginning of human existence on the Earth.

Humans rely on basic tools, or simply on their intuition, to “guide” themselves throughout their habitat, developing instrument to “determine their location” and to “act” accordingly.

This activity is common in our everyday life, while walking out of home and **determining the bast path** to go to the place we want to reach, **controlling the dynamics of our motion** by actuating our legs and upper body parts, and continuing **estimating our location** with our eyes, to check if it is on the right path.

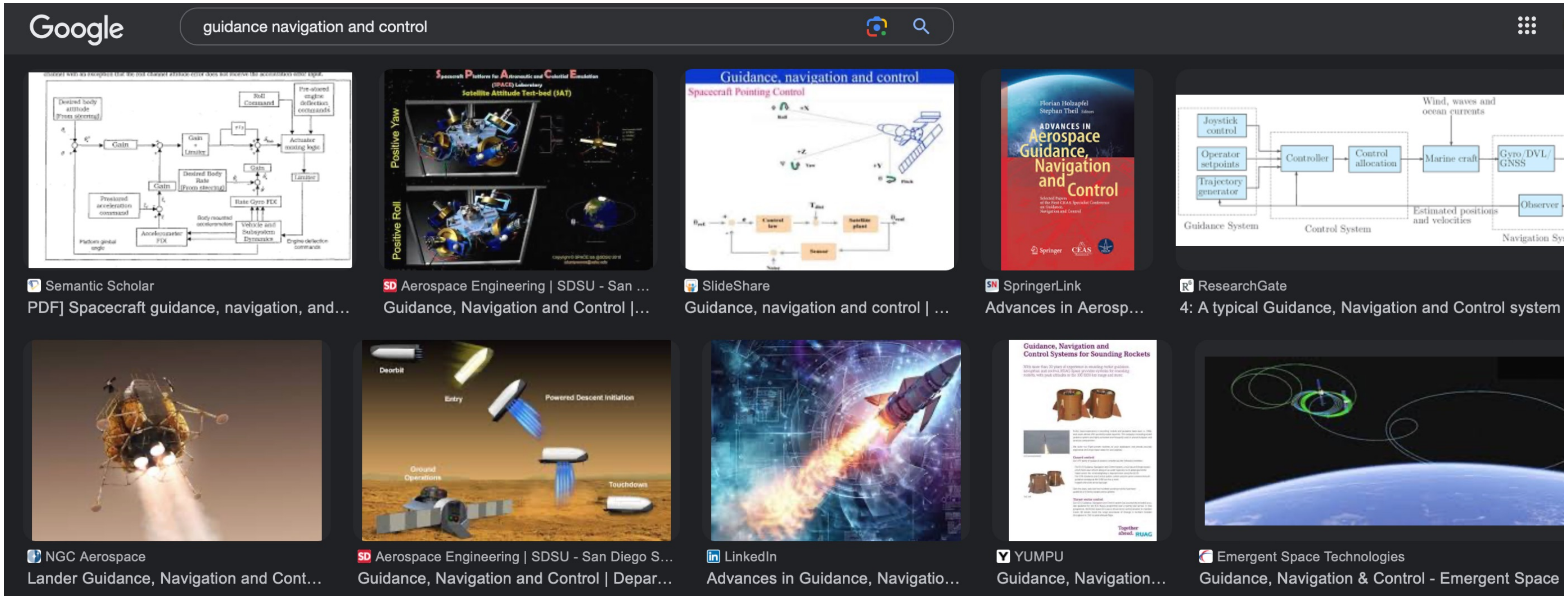
This activity is common and shared in many engineering field, whenever a dynamical object is moving in a prescribed way and has to achieve certain goal.

Aerospace GNC

GNC is therefore related with some generic key elements/blocks:

- Dynamical Objects
- Following a Prescribed Motion
- Sensing the Environment
- Actuating Forces and Torques

GNC is absolutely not an Aerospace thing, but if you search on Google this is what you find



Aerospace GNC

GNC has became primarily an aerospace thing because, humans realized that **GNC** was a **problem** and, thus, deserved a **solution** when a **dynamical object had to move in a vast and unknown space.**

First GNC engineers were ancient mariners who had to explore the world by guiding, navigating, and controlling the motion of an artificial object through a vast environment, still unexplored, without references and landmarks.

The ethimology of the term **GNC**, often colloquially and incorrectly identified by the single word **navigation,** belongs to the sea.

Navigationem was first used in the 1530s, from Latin “navigatus” meaning “to sail, sail over, go by sea, steer a ship,” from:

- *“navis”* (i.e., ship)
- *“agere”* (i.e., to drive)

(note how the word navigation was/is encompassing the entire GNC term!)

Aerospace GNC

The direct evolution of maritime GNC was the aircraft GNC, where airplanes move in vast regions following routes by means of navigation instruments and control systems: the pilots!

The first aerospace GNC was **human in the loop**, both for airplanes and for their descendants, the spacecraft.

"The first human in space was the Soviet cosmonaut Yuri Gagarin, launched with a Vostok 1 on the April 12, 1961. Despite he was a trained air force pilot, carefully selected to accomplish his task, the mission was designed to be automatically controlled or controlled from ground. In fact, the medical doctors were not sure how a human might react to space environment, and therefore it was decided to lock the pilot's manual controls. This primitive spacecraft control system was only partially trusted by the spacecraft engineers themselves, and thus a code to unlock the controls was placed in an on-board envelope to be used in case of emergency. The code was "1-2-5," and ***Gagarin was anyway told about it before launch."***

Aerospace GNC

However, not for long time, aerospace GNC remained human in the loop and soon after became **algorithms in the loop**.

The tremendous conditions of space environment, the complexity of spaceflight dynamics, the increasing velocity of jet airplanes, the inherent instability of air fighters, the complex constraints of drone's formations flying, the stringent requirement and constraints of flying vehicles, the unavoidable need for vehicle autonomy determined **the rise of the complete GNC subsystem in terms of BOTH hardware and software**.

The reliability of this subsystem is critical and fundamental for the accomplishment of any mission goals, for the sustainability of the platform and for the survivability of the system.

Aerospace GNC

Aerospace GNC failure may lead to a disaster and the complete loss of the mission

- ***Explorer 1*** *entered flat spin motion because of energy dissipation*
  - *Wrong dynamical modelling*
- ***Mariner 1*** *unexpected yaw lift because of guidance error*
  - *Left a minus sign out of an equation while entering hand-transcribed code into the GNC software*
- ***Phobos 1*** *entered test mode deactivating actuators*
  - *Missing character in software command*
- ***Hitomi*** *broke apart for excess of spinning*
  - *Error in the software logic*
- ***Ariane 5*** *self-destroyed after a 90 degrees flip*
  - *64-bit floating point data in 16-bit signed integer software*
- ***Boeing 737-Max*** *Max crashes*
  - *Unexpected pitch down maneuvers because of sensor data corruption*
- ***Australian army helicopter*** *crash*
  - *Missing software patch*

…and many more not disclosed software errors in small spacecraft, drones, airplanes or helicopters (not to speak of self driving cars)

Aerospace GNC

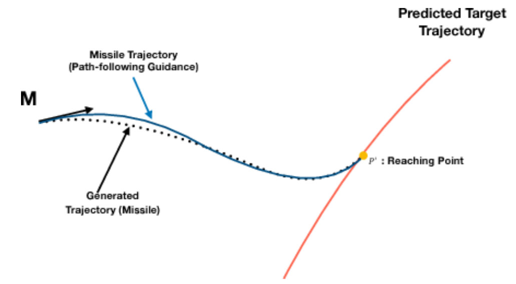
For all the previously described reason, it is of utmost importance to design an aerospace GNC subsystem considering:

- Correct modelling of the dynamics
- Correct selection and modelling of hardware components
  - Sensors
  - Actuators
  - On-board Computers (OBC)
  - Relevant electronic components (communication modules, transducer, memory elements, …)
- Proper model set-up in terms of data interfaces and subsystem structure
- Correct GNC algorithms implementation
- Correct software development process (Algorithm → OBSW)
- Comprehensive validation of models and processes
- Complete verification plan
- Complete testing plan

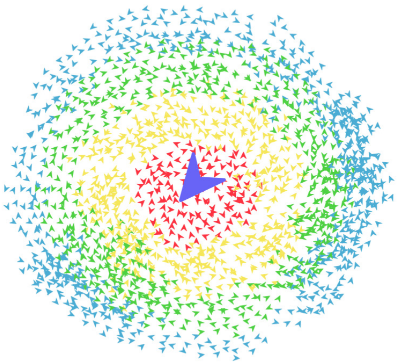
From these points the goal of this course on **(Advanced) Simulation and Testing of (Aerospace) GNC Algorithms**

Main GNC Terminology

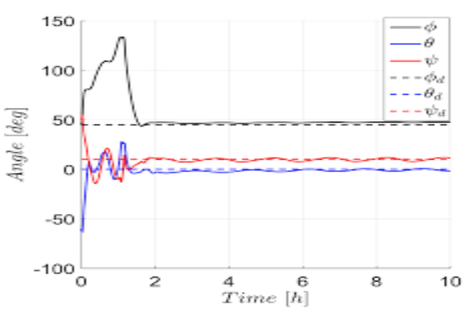
Guidance is referring to the determination of the desired path (i.e., "trajectory") from the vehicle's current location to an assigned target. It also establishes the desired changes in velocity, rotation, and acceleration to follow that course.



Navigation is referring to the determination, at a given instant of time, of the vehicle's location, velocity, rotation, and angular rate (i.e., "state vector").



Control is referring to the manipulation of forces and torques, by means of actuators (i.e., steering device, thrusters, etc.), needed to follow the desired path while maintaining vehicle stability.



Main GNC Terminology

Space engineering refers to GNC sometimes with different and various terminology according to the specific application or the location of GNC subsystem components.

Specifically, the GNC subsystem can be **shared** between the **space segment** and the **ground segment**

Thus, different terms are commonly associated to the GNC:

- GNC (Guidance, Navigation and Control)
- AOCS (Attitude and Orbit Control System)
- OCS (Orbit Control System)
- ADCS (Attitude Determination and Control System)
- …

GNC

GNC is COMMONLY used for the onboard segment, when the satellite position is controlled in closed loop (e.g., rendezvous, formations flying, …)

In the most general terms, the GNC functions are:

- Attitude guidance
- Attitude estimation/determination → Attitude Navigation
- Attitude control
- Orbit guidance (absolute or relative)
- Orbit estimation → Orbit Navigation (or Navigation)
- Orbit control

All the GNC functions are located **on-board** and executed in real-time with a full closed GNC loop.

Trajectory planning (and optimization) is performed on-board → Necessary for autonomy!

AOCS

AOCS is COMMONLY used when the guidance is NOT performed on board (e.g., classic low Earth orbit (LEO), medium Earth orbit, and geostationary orbit missions.)

The classical AOCSs considered here include the following functions:

- Attitude estimation/determination → Attitude Navigation
- Attitude control
- Orbit estimation → Orbit Navigation (or Navigation)
- Orbit control

Path and trajectory are planned (and optimized) on ground and sent to spacecraft in terms of guidance profiles (e.g., polynomials, interpolating functions) or direct commands for the controller.

The full GNC (guidance + navigation + control) system is anyway present, but **distributed and shared between the ground and the space segment**

Ground-based Navigation

Sometimes, orbit navigation is also located and executed on ground (e.g., interplanetary missions, lunar missions, …).

Ground Orbit Navigation is COMMONLY referred to as **Orbit Determination.**

In this case, the AOCS can be NOT equipped on-board in terms of orbital part.

The **ground segment** estimates the orbital state (**Orbit Determination**), computed the best path to reach the desired orbital state (**Orbit Guidance/Path Planning**) and the best control law (**Orbit Control**) to achieve it.

The control law is converted to the actuators command with a ground-based actuation function and the ground segment directly telecommands the actuators on-board.

In this case orbit control is NOT in closed-loop but the control is executed in **open-loop** and discrete trajectory correction maneuvers (**TCM**) or station-keeping maneuvers (**SKM**) are needed.

This GNC system can be referred to as **OCS**

ADCS

Attitude estimation and control functions cannot be located on ground, and it is (almost) always executed on-board in closed-loop.

Ground based attitude control, with direct actuators commanding, can be present in case of hard non-nominal conditions, trying to recover the mission with a safe and simple attitude state (e.g., single or dual spinning, rough detumbling, …)

Attitude GNC is typically referred to as **ADCS**, composed of the following functions:

- Attitude guidance
- Attitude estimation/determination → Attitude Navigation
- Attitude control

Attitude guidance can be fairly simple:

- Reference quaternion
- Solar panels pointing
- Reference spinning state
- …

It is present on-board in terms of predefined reference directions (e.g., ($q_{ref} = [0,0,0,1]$) or look-up tables.

ADCS

Attitude guidance can be more complex and to be executed on-board:

- Lat-Lon ground pointing
  - Requires on-board position estimation
- Target spacecraft pointing
  - Requires on-board relative state estimation
- LVLH relative pointing
  - Requires on-board position estimation
- …

In all these cases attitude guidance computes a single reference direction or a single reference tracking.

A change in reference pointing is commonly joined with a direct control execution, and the path to the reference state is handled by the control law ($u_c = -k\, q_{err}$)

Attitude trajectory ($q_a \rightarrow q_b$) planning can be executed on-board, but it is commonly computed on ground and (tele)commanded to the spacecraft uploading the desired reference attitude path/trajectory in terms of guidance profile (e.g., Chebyshev polynomials).

This commonly referred to as ground-based slew profile maneuvers (**SPM**).

The Murphy's Law in GNC Functions

Murphy's Law belongs to the Aerospace domain and the impossibility to directly access and perform maintenance operations on the spacecraft make the Space Engineer one of most pessimistic professional in the world.

We always need to think to any component that can have a failure, to any event that can bring the system to a dangerous or failure prone state, to any cause whose effect can propagate with a failure on the system.

Despite this bad omen approach, something will for sure fail and we will for sure have problem and to recover the system.

For these reason, the ECSS always lists a specific function in any GNC subsystem, both GNC, AOCS, OCS, ADCS:

Acquisition and maintenance of a safe state in emergency cases and return to nominal mission upon command

Thus, any GNC subsystem shall have a GNC algorithm capable to deal with ideally any unexpected event, bringing and maintaining the spacecraft in a known **safe mode** state to guarantee the survivability of the mission.

Aerospace GNC Algorithms

Aerospace GNC algorithms are the high-level abstract finite sequence of mathematically rigorous instructions used to solve the GNC problem.

Algorithms are organized and subdivided according to the different GNC functions. They are commonly mathematically defined and formulated, to be commonly implemented with a **model-based approach**.

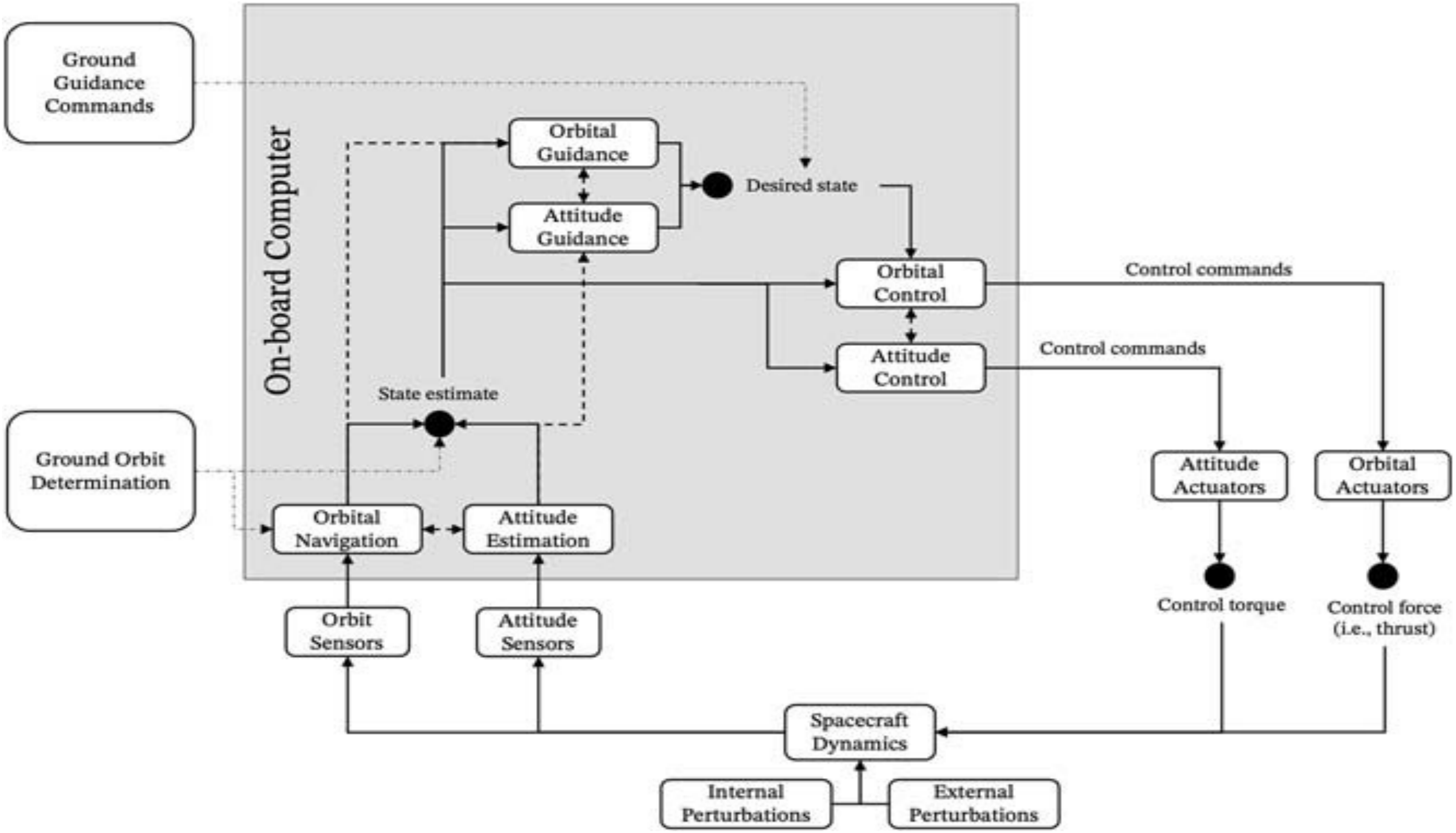
GNC algorithms define the **inner core** of the whole **GNC subsystem** on the **software** side: from the hardware interface with the sensors to the interface with the actuators, passing through the interface with the platform management module.

This PhD course only considers the **on-board GNC part**, with the **complete attitude and orbit sections**.

It focuses on their model-based simulation and verification, with brief hints on their development towards the on-board software and to the hardware testing.

GNC Architecture

The GNC is a complex subsystem that extensively involves hardware and software components. The systematic decomposition of the GNC subsystem in the underlying blocks with the associated interfaces is commonly denoted as **GNC architecture**



GNC Architecture

Guidance block computes desired future plans to fulfill the mission objectives:

- Creates reference trajectories (translational and rotational) to be followed
- Trajectories must be feasible
  - Account for the spacecraft system design
  - Account for constraints and path corridors

Given its higher level of abstraction it descends directly from the mission objectives, and it is characterized by a high level of mathematical content (i.e., optimization loops, interpolants, model-based functions, environment models).

- Computationally demanding
- May require dedicated processing units outside the fast frequency GNC loop
  - Historically demanded to ground control segment

Guidance module generally needs the **output of the navigation** to deliver **inputs to the control** module. Certainly, the guidance module **needs to know the actual state** of the system to adjust future trajectories to the current system status.

GNC Architecture

Navigation block senses the environment to deliver the best estimate of the current state of the spacecraft:

- Computes (translational and rotational) current dynamical states
- It is directly interfaced with sensors
  - Sensors post-processing considers navigation design and implementation

The navigation entails different algorithms, which are highly dependent on the sensor suite available on-board. Navigation fuses measurements from different sensors to deliver the best possible estimates, also based on the environment knowledge.

Navigation module generally needs the **output of the sensor post-processing** to deliver **inputs to the guidance and control** module.

GNC Architecture

Control block manages the reaction of the spacecraft with respect to the dynamical environment to follow the desired trajectory:

- Calculates forces and torques to have the actual and the desired states coincident
- It is directly interfaced with actuators
  - Actuators command processing (actuation block) converts desired forces and torques into specific actuators commands

The control is based on a controller algorithm, which is mathematically designed with high abstraction level. Control functions are designed to comply with system limitations, desired dynamical states and expected environmental disturbances.

Control needs the **outputs from the guidance and navigation modules**, computes the deviation with respect to a reference state and generate **inputs to the actuation block**. Control and actuation blocks can be integrated together to be directly interfaced with the actuators on-board.

GNC Architecture

Sensors is the ensemble of hardware in charge of sensing the environment to deliver a direct or indirect measurement of the spacecraft state, both orbital and attitude one. Sensors must be characterized and calibrated because a proper modeling of their functioning is critical for the navigation algorithms to perform well.

Actuators is the ensemble of hardware in charge of delivering either an external torque or thrust interacting with the environment. Also, actuators must be characterized and calibrated because a proper modeling of their functioning is critical for the GNC algorithms to perform well.

GNC Architecture

GNC Manager is a software block dedicated to handle GNC telemetries and telecommand for the proper operations of the GNC inner algorithmic core.

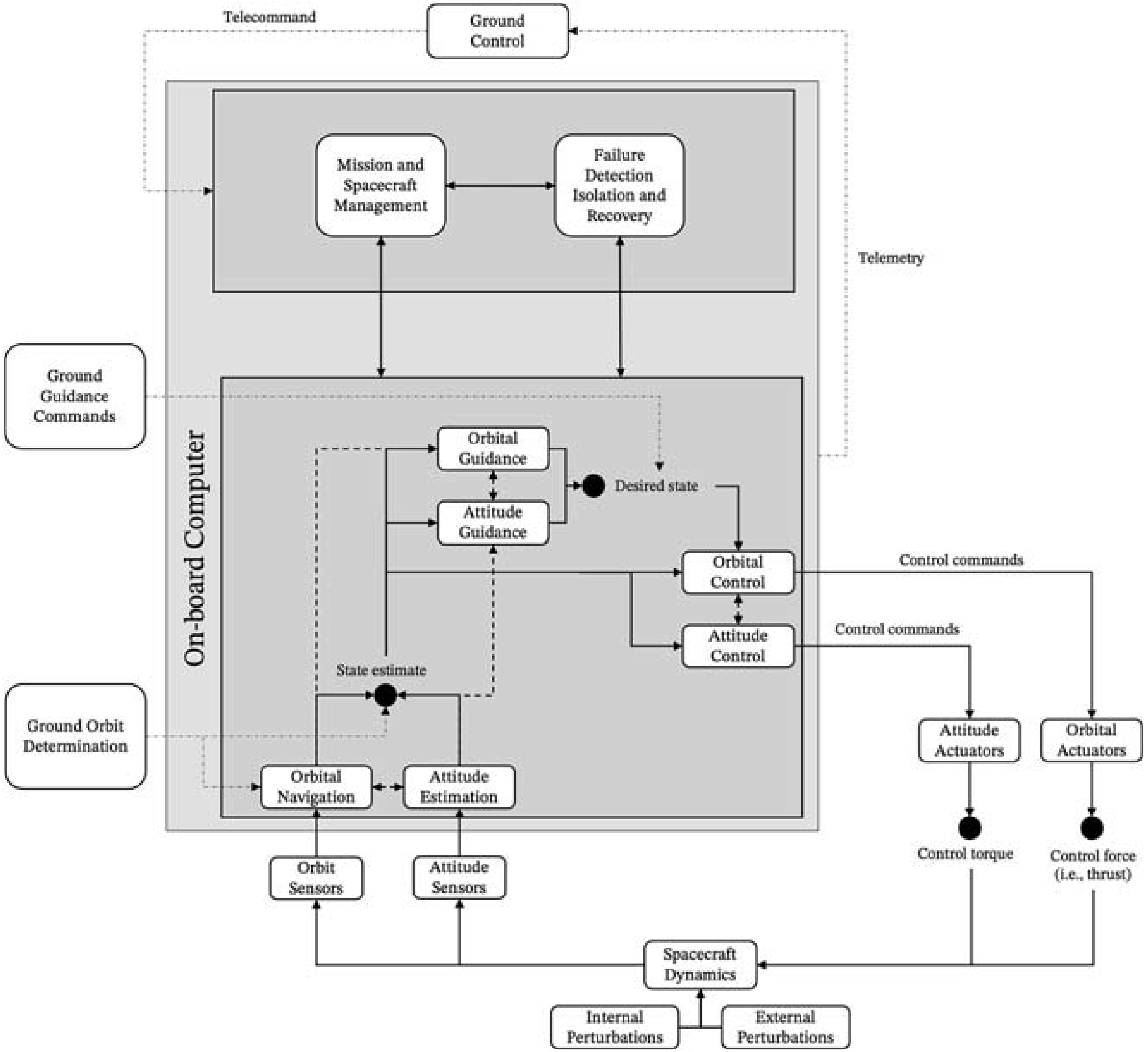
This module is dedicated to:

- Receive commands from the main spacecraft OBC/core and interface them with the GNC blocks
- Send communication to the main spacecraft OBC/core about GNC data and status.
- Enable the proper GNC mode (Mode Manager)
- Collect GNC telemetry data and record unexpected events for monitoring the good operative status of the GNC blocks
- Schedule periodic checks with respect to subsystem hardware components
- Detect faults and anomalies to be then isolated, processed and, possibly, recovered by the FDIR (fault detection, isolation and recovery) functions

This element requires to reason at system level, because spacecraft anomalies can interfere with the GNC and vice versa.

This element is interfaced with the **Mission and Spacecraft Management (MSM)** system, also known as **Mission and Vehicle Management (MVM),** which is in charge to oversee the overall behavior of the spacecraft and its subsystems, including the GNC one, in order to achieve the mission goals, **managing the system phases and modes**.

GNC Architecture



GNC Architecture

One often overlooked component and interface with the GNC blocks is the external **Environment, Spacecraft Dynamics and Kinematics (DKE)**.

This is obviously an external (natural) element, NOT to be designed but to be taken extremely under consideration and to be accurately modelled while handling the GNC simulations for design, verification and testing.

The spacecraft dynamics and environment represents the ensemble of physical laws dominating the surrounding of the spacecraft. They offer a mathematical representation of the natural forces and torques acting on the spacecraft, and of the laws governing the motion of the system.

Combining them together, the DKE has a double role:

- It is a (always partially) known entity that drives the actual system state, whose accurate modeling is critical to validate the developed algorithms.
- It is a modelling instrument at different levels of accuracy when designing the algorithms themselves (e.g., MPC, navigation filters, internal environment models, …).

The first is something we cannot do much about and the spacecraft is simply reacting to it, the second is the **engineering knowledge** that needs to be traded-off in each application.

GNC Design and Requirements

GNC subsystem design is an iterative process, whose first step is the definition of guiding requirements based on mission goals and objectives.

The process can be outlined in a set of design steps

Design step	Description	Inputs	Outputs
1	Subsystem requirements definition and derivation—subsystem criticalities	• Mission requirements • System requirements • Subsystems constraints • Mission Concept of Operations (ConOps) • Mission timeline	GNC subsystem requirements
2	Definition of GNC modes	• GNC subsystem requirements • Subsystems constraints • Mission ConOps • Mission timeline	List of different GNC modes
3	Environment and disturbance assessment	• Spacecraft geometry • Operational orbit • Epoch and mission timeline • Mission ConOps	• Station-keeping needs • Disturbance forces (i.e., internal and external) • Disturbance torques (i.e., internal and external)

GNC Design and Requirements

4	Sizing and selection of GNC subsystem hardware	• Spacecraft geometry • Operational orbit • Epoch and mission timeline • Mission ConOps • Disturbances • Subsystem constraints	• Preliminary definition of the sensors suite • Preliminary definition of actuators suite • Preliminary definition of computational architecture (i.e., on-board software — OBSW — and on-board computer — OBC)
5	Definition of GNC algorithm	• Sensors suite performance and characterization • Actuators suite performance and characterization • Subsystem performance requirements	• Algorithms for GNC subsystem • Mode manager
6	Iterate from 1	Output from 1, 2, 3, 4, 5	Refined mission and GNC subsystem requirements.

GNC Design and Requirements

The GNC design process outlines the final **GNC requirements document,** which is the fundamental basis to set-up the **verification and testing plan,** which is successfully completed when all the requirements are closed and satisfied by:

- **Inspection** → Need a direct visual check on the associated elements
- **Review of Design** (RoD) → Need a set of validated mathematical calculations and abstract models to confirm design results checking design evidences (e.g., reports)
- **Analysis** → Need a set of validated analytical and numerical models and simulation campaign to confirm design results
- **Test** → Need a set of validated HW equivalent models, test benches and HW simulators to confirm design results

The GNC requirements document is part of the Project Requirements Document, which is comprehensive of the entire mission, meaning that **GNC requirements** can be **bidirectionally connected** to other **system/mission requirements**.

Thus, a set of (SW/HW) models, tools, simulations, test benches and procedures, which need to be validated, are the fundamental elements to complete GNC design, verification and testing activities up to the final spacecraft operations.

This phase is completed when no open requirement is left, as stated by the Verification, Control Document (**VCD**).

GNC Design and Requirements

Preliminary Requirements → Design Process → Refined Requirements → Verification and Testing Plan → Closed Requirements → Complete VCD

Design Process

- Abstract (Mathematical/Physical) Models
- Theoretical Calculations
- Single run Simulations
- Reduced Models
- …

Verification and Testing Plan

- Monte Carlo Simulations
- Refined Validated Models
- Testing Activities
- Hardware Characterization
- Subsystem Integration
- …

GNC Design and Requirements

The whole GNC design, verification and testing process shall be conducted in a way that is coherent with the accuracy and complexity level of the mission.

The GNC engineer shall be capable to identify the right level of complexity and performance for the project:

- GNC system complexity and performance
- Model complexity
- Design complexity
- Verification complexity
- Testing complexity
- …

Always having in mind the **golden rule**:

A designer knows that he has achieved perfection not when there is nothing left to add, but when there is nothing left to take away.

Requirements Tailoring

For each mission, it is necessary to adapt the specified requirements through a complete tailoring process:

- To decide if a requirement is necessary, taking into account the specific functionalities required for the mission (cfr. golden design rule). For instance, if a mission requires an on-board navigation function, then the requirements dedicated to this function or to an on-board GNSS receiver are applicable.

- To adapt the numerical values of a requirement, considering the exact performances required for the mission. The designer must reason on the actual needs of the mission, without spending resources for unnecessary performance level.

- To quantify the new hardware and software development necessary for the program, which is a key factor in adapting the verification requirements.

GNC Requirements Properties

GNC requirement need to have the following properties:

- **Correct**: requirements shall describe the functions the software implements.
- **Unambiguous**: requirements should be clear and avoid ambiguities coming from the use of language.
- **Unique**: requirements should not be repeated.
- **Atomic**: requirements shall be addressed separately;
- **Complete**: requirements should contain all the required information or refer to it.
- **Justified**: requirements shall contain a justification and include the person responsible for its development.
- **Traceable**: requirements should be connected with the requirement from which they derive and with the requirements/functions that implement them.
- **Identifiable**: requirements should have a unique identifier.
- **Consistent**: requirements should be consistent with respect to the other requirements and the system specification.
- **Verifiable**: requirements must be verified using a verification method.

Requirements Traceability

Traceability of requirements is fundamental when dealing with GNC verification and testing.

Bi-directional traceability is requested by ECSS, and consists of:

- traceability from system requirements up to algorithms/software, as a way to assess the implementation of functionalities in the software (**forward traceability**)
- traceability from code to system requirements to check that no unintended functionalities are present in the code (**backward traceability**).

During GNC algorithms/software development requirements that are not linked to a higher level one can be originated, these are called derived requirements and need to be directly checked and assessed.

The presence of derived requirements is a metric to determine how well the system assessment and the high-level requirement development processes have been performed. During requirement tailoring, each GNC design defines its own standards about derived requirements, they are usually **tolerated up to 10% of the total number** of requirements.

GNC Requirements

The types of GNC requirements can be divided into:

- **Functional requirements.**
  - Dealing with high level functions of the system (i.e., what the system has to do, and what functions shall contain.
  - Typically paired with FDIR requirements (remind the Murphy’s Law).
- **Operational requirements.**
  - Dealing with GNC operations, telemetry and GNC autonomy.
- **Performance requirements.**
  - Specifying the flight domain to be handled by the GNC.
  - Dealing with the performance of the GNC subsystem in all the flight domain and in different GNC modes.
  - Requesting the calculations of GNC budgets.
- **Verification requirements.**
  - Dealing with verification plan of the GNC at different levels.
  - Specifying requirements for the verification facilities.
  - Specifying requirements along the entire verification plan
    - From design and model performances to in-flight verification
- **Documentation requirements.**
  - Dealing with the documents to produce across the GNC design and verification process

Functional Requirements

High level functions

The GNC shall provide the hardware and software capabilities and performances:

- to perform the attitude measurement, estimation, guidance and control needed for the mission;
- to perform the orbit measurements, estimation, guidance and control manoeuvres, as specified by the mission requirements;
- to ensure a safe state of the spacecraft at any time, including emergency and anomaly situations, according to failure management requirements;
- to ensure the mission availability, as specified at satellite level.

Attitude acquisition and keeping

- The GNC shall provide during all phases of the mission the capability to acquire and keep all attitudes necessary to perform the mission.
- The GNC modes used for initial acquisition shall provide the capability for transition, from the initial attitude and rate after launcher separation to the final mission pointing, in a safe and orderly sequence (e.g., detumbling).

Functional Requirements

Attitude determination

The GNC shall provide, as specified by the mission requirements, the hardware and software means for autonomous on-board attitude determination.

Orbit acquisition and maintenance

The GNC shall provide the capability for achieving orbit control manoeuvres specified by mission analysis

Orbit determination

If a navigation function is necessary for the mission, the GNC shall provide the hardware and software means for autonomous on-board determination of the spacecraft orbital state which includes position, velocity and time. Orbit determination can be implemented on board and/or on ground.

Reference frames

The AOCS shall identify and define unambiguously reference frames needed for:

- attitude and orbit measurement,
- attitude and orbit guidance,
- attitude and orbit navigation,
- attitude and orbit control.

Functional Requirements

Safe mode

In case of major anomaly, the GNC shall provide the autonomous capability to reach and control safe pointing dynamical states (e.g., attitude, rates, velocity, …) to ensure the integrity of the spacecraft vital functions, including power, thermal and communications.

- The return from safe mode shall be commandable by ground TC

GNC mode transitions

The GNC shall define a strategy for the implementation of the mode transitions and describe how this strategy is applied for each AOCS mode transition, including the following items:

- transition conditions and the check performed by the on-board software,
- actions on software and hardware performed autonomously on board,
- actions to be performed on ground.

Transitions between modes shall be triggered by one of the following mechanisms:

- TC: by ground request (Time tagged or not),
- AUTO: autonomously on board, after checking a transition condition,
- FDIR: after failure detection.

Auto and FDIR transition can be inhibited or forced by TC.

GNC Modes

Mission objectives and requirements often imply more than one mode of operating a spacecraft. Indeed, a contextual step to the requirement generation is the **identification of the GNC modes**.

The requirements generally begin with a description of the control modes the GNC is expected to execute to meet those goals. This is because most of the requirements, functional, operational, and performance are dependent on the specific GNC mode involved.

GNC modes shall be defined by evaluating their features against:

- **Flexibility:** It is the capacity to adapt and solve more circumstances, without ambiguities, with simple and effective configuration options.
- **Autonomy:** It is the capacity to be operative without the need of ground intervention.
- **Redundancy:** It is the capacity to be operative in case of the loss of some hardware components.
- **Performance:** It is the capacity to guarantee GNC functionalities with various control authority and computational performances.

In principle, each GNC mode may rely on different sensors, actuators, navigation, guidance, and control functions. In particular, the GNC functions should be as independent and unique as possible.

GNC Modes Example

GNC mode	Tasks	Characteristics
Commissioning	• Commissioning of the GNC system • Sequential start-up of components	• To be used before relevant mission phases • It should include a monitoring of the system in order to be able to detect (autonomously or from ground) anomalies prior to the operative phases • No particular performance requirement is imposed • Refer to Chapter 12 – GNC Verification and Validation
Nominal	• It fulfills the main mission objectives • It may be comprehensive of submodes, such as image acquisition, communication pointing, proximity control	• Best performing hardware is involved • Highest performance requirements
Safe	• Detumble the satellite • Power-saving • Stable pointing, usually toward the Sun	• Reliable hardware is employed • No performance requirement is imposed (i.e., also rough navigation or control are accepted) • Must be autonomous

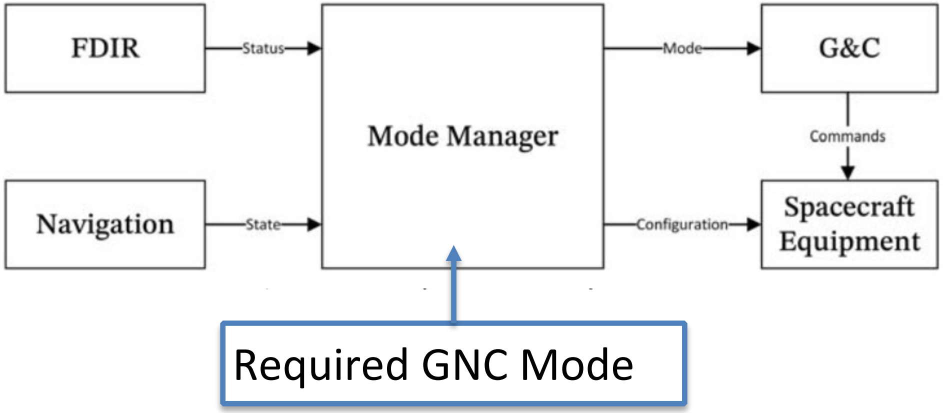
GNC Modes Example

GNC mode	Tasks	Characteristics
Standby	• Power positive pointing • Ground station pointing • Particular pointing required by a set of subsystems (e.g., power and thermal) • Drag drift minimization (e.g., hold point)	• It is typically autonomous • In general, the spacecraft is more operative with respect to safe mode • Different GNC functions with respect to Safe mode
Orbit control	• It performs orbital maneuvers	• It can be split into inertial and relative maneuvers • Propulsion subsystem is involved • Parasitic torques coming from thrust misalignment have to be controlled
Transfer	• It controls the attitude and orbital state during long transfers (e.g., to get to GTO or interplanetary transfer) • Interplanetary navigation	• It shares a lot with orbit control mode • It may have additional system requirements during ballistic arcs
Special	• It fulfills particular extended objectives of the mission	• Various

Modes Management

Despite the mode transition is commanded from ground or autonomous, a modes management (**MM**) software shalle be present and interfaced with the MVM, to actually execute transitions and monitor the GNC algorithms/software.

The role of MM is to directly provide **commands to the GNC algorithms/software** in order to **adapt** with respect to the required GNC mode, according to **system state** and **status**.



Modes transition shall be accurately organized according to a finite state machine (**FSM**) designed, avoiding errors coming from the MM:

- Logical errors (e.g., an event triggering no state).
- Ambiguous states (e.g., an event triggering multiple states).
- Transition errors (e.g., an event triggering the wrong state).
- Unreachable states (e.g., states that cannot be activated by any event).
- Loop states (e.g., states that cannot be excited after their activation).

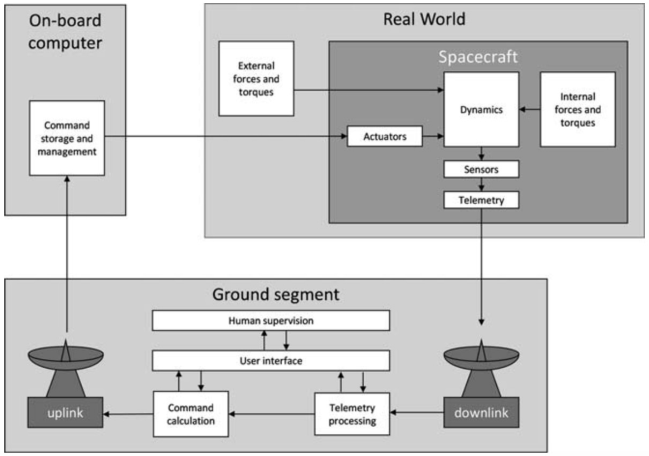
Rely on Ground … or Not

Spacecraft Management and GNC Management is a shared task between ground and space segment, and the input going to the Mode Manager can arrive from different architectural organization.

Primary Ground Based:

More computational power and human control/revision is available on-ground.

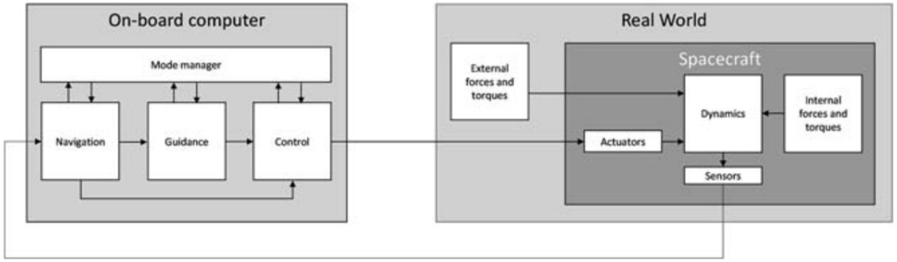
MM handles sequence of telecommands, and provides telemetry to ground.



Primary On-board:

More autonomy is possible, and ground only supervise the GNC loop

MM handles the whole GNC operations paired with the MVM.



Consider the anomalies: FDIR in Functional Requirements

FDIR functions

- The GNC shall contribute to the satellite FDIR definition through identification of GNC failure cases, monitoring means and policy, recovery means and policy, in order to fulfil the objectives for the satellite FDIR.
- The GNC shall provide to satellite FDIR, for the purpose of failure monitoring, parameters observed on GNC units or derived from GNC embedded algorithms.
- The GNC shall implement actions on GNC units and GNC modes required by the satellite FDIR in case of anomaly.
- The selection of the requirements for GNC monitoring and actions, with respect to satellite autonomy, availability, reliability and fault tolerance, shall be justified.
- The GNC software shall provide the capability to:
  - disable or enable, by ground command, any autonomous GNC FDIR monitoring or action function,
  - modify by ground command the parameters of FDIR monitoring and actions.

FDIR functions design is based on a dedicated **top-down Fault Tree Analysis (FTA)** or on an inductive **bottom-up Failure Mode Effects and Criticality Analysis (FMECA).**

Consider the anomalies: Redundancy Functional Requirements

Hardware and software redundancy scheme

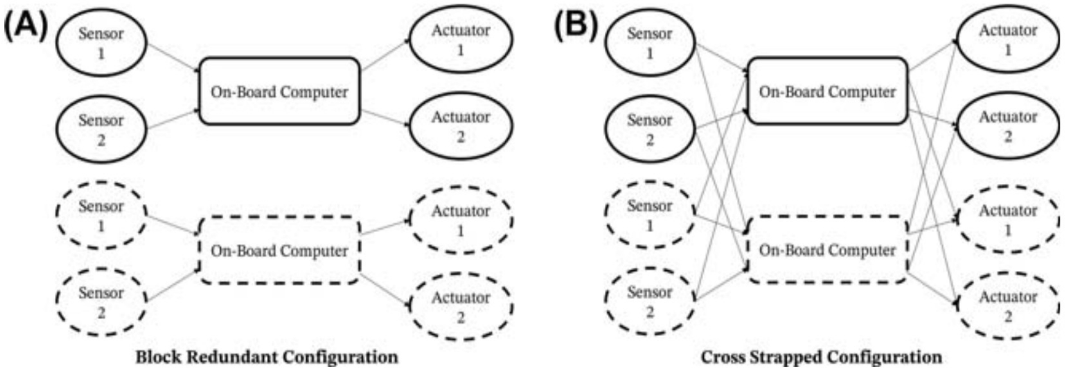
- The GNC shall justify the hardware redundancy implemented against failure tolerance requirements and reliability requirements.
- The GNC shall justify the design of the safe mode against the risk of common design error and common failure with the modes used for the nominal mission.

Redundancy of componente can can be split into cold, hot, and active:

- **Cold redundancy**: The secondary hardware is not operative and normally switched off until a failure on the primary component occurs.
- **Hot redundancy:** All entities are powered on with only one operating.
- **Active redundancy:** All the components are operative and fused/selected by the GNC subsystem.

Hardware and software GNC components can be organized in two alternative configurations:

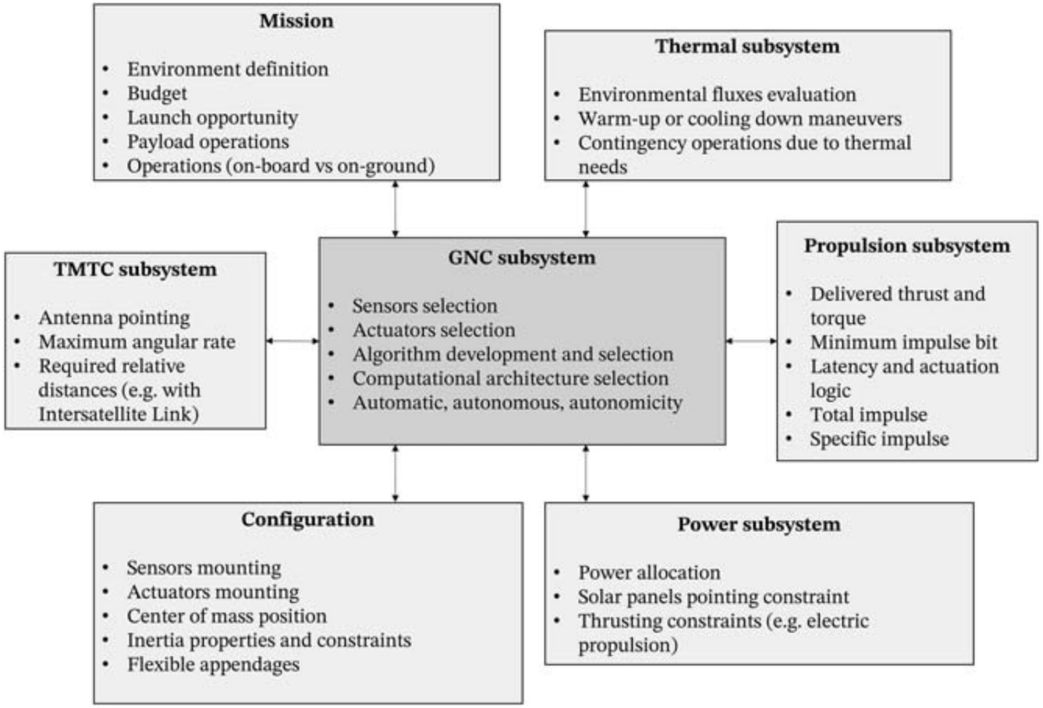
- **block redundant**
- **cross-strapped**



Relation with the Spacecraft System Requirements

System and satellite requirements on GNC

- It shall be demonstrated that the GNC design does not prevent meeting the requirements imposed by the mission and payload constraints.
- At satellite level, it shall be demonstrated that the GNC design is compatible with other functional chains for the attitudes and durations.
- A mapping of GNC modes into satellite modes shall be described, to clarify all the possible configurations in which the satellite can be in relation with the GNC.



Operational Requirements

Operational requirements specifies the operations of the GNC.

- **Requirements for telecommands**
  - Requirements for GNC parameters update
  - Requirements for orbit control manoeuvres
  - Requirements Orbit determination
  - Attitude guidance
- **Requirements for telemetry**
  - AOCS needs for ground monitoring
  - Housekeeping TM
  - Diagnostic and event TM
- **Requirements for autonomous operations**
  - LEOP autonomy
  - Minimum time period of autonomy
  - Minimum time period of autonomy in safe modes
  - …
- **Requirement for calibration operations**

Performance Requirements

Performance requirements charachterize the GNC systems in terms of:

- **Performance errors**:
  - Absolute Attitude Pointing (APE)
  - Absolute Attitude Knowledge (AKE)
  - Absolute Rates (APE)
  - Relative Attitude Pointing (RPE)
  - Absolute Orbit Knowledge (AKE)
  - Orbit Control (APE)
  - …

Absolute knowledge error (AKE)
Absolute performance error (APE)
Relative knowledge error (RKE)
Relative performance error (RPE)

- **Stability**: the ability of a system submitted to bounded external disturbances to remain indefinitely in a bounded domain around an equilibrium position or around an equilibrium trajectory.
- **Stability margins**: quantify maximum parameters excursions preserving stability properties.
- **Agility**: the ability of a system to perform a maneuver in given time interval, including the tranquilization phase.
- **Transient response**: the ability of a system to reach the steady state with given maximum parameters.
- **Robustness**: the ability of a controlled system to maintain some performance or stability characteristics in the presence of plant, sensors, actuators, and/or environmental uncertainties.

Performance Requirements

Performance requirements are commonly associated to a probability and to a statistical interpretation. Moreover, they can be imposed in “real-time” to be achieved on-board or ”a-posteriori” with dedicated processing of telemetries by the ground control centre.

Some example performance requirements are:

- *The GNC shall ensure during the operational mission phase an absolute pointing performance of TBS microradians, at TBS % confidence level, using the TBS (temporal, ensemble or mixed) statistical interpretation.*
- *The GNC contribution to the system level “a posteriori” attitude knowledge shall be lower than TBS microradians, at TBS % confidence level, using the TBS (temporal, ensemble or mixed) statistical interpretation.*
- *The navigation function shall provide the on-board orbit estimation with an accuracy of TBS metres (for position), TBS metres per second (for velocity) and TBS seconds (for time), in a TBS unambiguous space and time reference frame, at TBS % confidence level, using the TBS (temporal, ensemble or mixed) statistical interpretation.*

ECSS specify temporary **outages** to these performance requirements (e.g., LEOP, Safe Modes, Wheel Off-loading, Calibration Modes, Orbit Maneuvers, …) and also impose **time to acquire** these performance through the agility and transient performance requirements.

Verification Requirements

The **GNC cannot be fully verified in real conditions before flight**. The main reason is that the hardware on ground cannot be submitted to the real flight conditions and environment.

During the GNC verification process, a complete and careful step by step verification logic from numerical models to real hardware is carried out in order to validate the behaviour of the GNC.

ECSS outline typical requirements concerning the logic and sometimes the facilities used for GNC verification, for the following main steps of verification at different levels:

- GNC design and performance verification
- GNC hardware/software verification
- Verification at satellite level
- GNC-ground interface verification
- In-flight verification

GNC verification does not complete on ground (!), but it is necessary in orbit during LEOP, Calibration phases, new software functionalities, software patches, …

A bit of Verification Terminology

- **Analytical/Theoretical Investigation**: A set of mathematical, physical, theoretical studies and calculations, executed on abstract models to support design activities or to preliminary verify some fundamental requirements
  - Analytical methods are feasible only if no complex dependencies exist, and requires reduced simplified models (e.g., linearized models, SISO analyses)
  - Provide fast and theoretically exact results to confirm design assumptions and model implementation
  - Can be used to set boundaries to validate simulation results
- **Simulator**: An ensemble of one or more numerical models that are executed together to represent the behaviour of phenomena and/or an artificial system (e.g. spacecraft).
- **Scenario**: A particular initial configuration of a simulator and sequence of events to represent a particular part of a mission (e.g. launcher deployment, eclipse operations, cruise phase.)
- **Simulation**: A run of scenario in a simulator with a simulated start- and end-time. During the simulation events may be injected into the simulation.
  - Can be performed exploiting reduced simiplied models for preliminary verification activities, or functionally validated engineering simulator to verify the GNC design and performances
  - Can be time consuming and need a proper validated simulation set-up to provide valid results
  - Can be used to compare the results of software and hardware verification phases

Theoretical/Analytical methods and Simulation methods support the requirements verification by Analysis (A)

A bit of Verification Terminology

- **Test:** a set of controlled and predefined series of inputs, data, or stimuli applied to representative hardware, equivalent models, or flight hardware, under representative environment, to ensure that the system will produce a predefined output as specified by the requirements (confirming the product characteristics, performance or functions)
  - Can be executed on hardware, software or mixed models
  - It is executed to verify that an existing implemented system works as expected without errors or anomalies
  - It is a verification method, used to verify requirement by test (T)
- **Validation**: a process which demonstrates that the product is able to accomplish its intended use in the intended operational environment
  - It demonstrates that the product is the right product and complies with its initial definition
  - Referring to models, validation process to ensure that the model is representing the real world as much as possible.
- **Verification**: a process which demonstrates through the provision of objective evidence that the product is designed and produced according to its specifications, is free of defects and complies with the requirements
  - It refers to analytical, numerical, simulation, testing, inspection and review activities to check that the system satisfied design requirements
  - Not to be confused with test
  - Not to be confused with validation

GNC Verification

The **GNC design and performance verification step** demonstrates that the GNC definition (e.g. modes, architecture, equipment and tuning) is compliant with the GNC functional requirements (e.g. performance, pointing and delays). It includes both analyses and simulations. Sensitivity and robustness analysis are part of this step.

The **GNC hardware/software verification** is intended to verify the functional GNC behaviour with a configuration representative of hardware/software, interfaces and real-time performances. It concerns the overall functional chain, each part of the AOCS (flight hardware and software) being individually verified with respect to its own specification in a separated process.

The GNC hardware/software verification step can be performed on a dedicated test bench, called **Avionics Test Bench**.

All these steps participate to the complete verification of the GNC functional chain. However, **GNC shall not only be verified alone**, and these verification steps can be also performed at other levels of responsibility or with other functions (such as avionics, satellite and system).

This step-by-step verification logic is fully applicable for a programme including new developments on both hardware and software aspects. A **tailoring of these requirements** has to be done when a lot of **hardware units** and **software functions** are **reused** from a previous development, and when some verification steps performed previously can be considered as applicable to the considered programme. This is especially the case for a family of missions or a family of satellites.

- The GNC design and performance verification includes extensive analyses and simulations for a completely new development, or for a new family of satellites. For a recurring satellite, it can be simplified and adapted in order to focus on the real specificities (e.g., new hardware, new software modules, or new mission parameters.)
- The GNC hardware/software verification for a completely new development relies on benches including usually hardware models functionally representative of flight hardware, while it can involve numerical simulation models of some hardware units for a recurring satellite, or for a satellite of an existing family.

The recurrence level is included in the design justification file and the justification of the selected verification strategy is included in the Verification Plan.

GNC-ground interface shall be verified as well for new development and consistently assessed also for recurring missions.

GNC Verification: MIL to HIL

As seen, the GNC verification is a step-by-step process with different blocks.

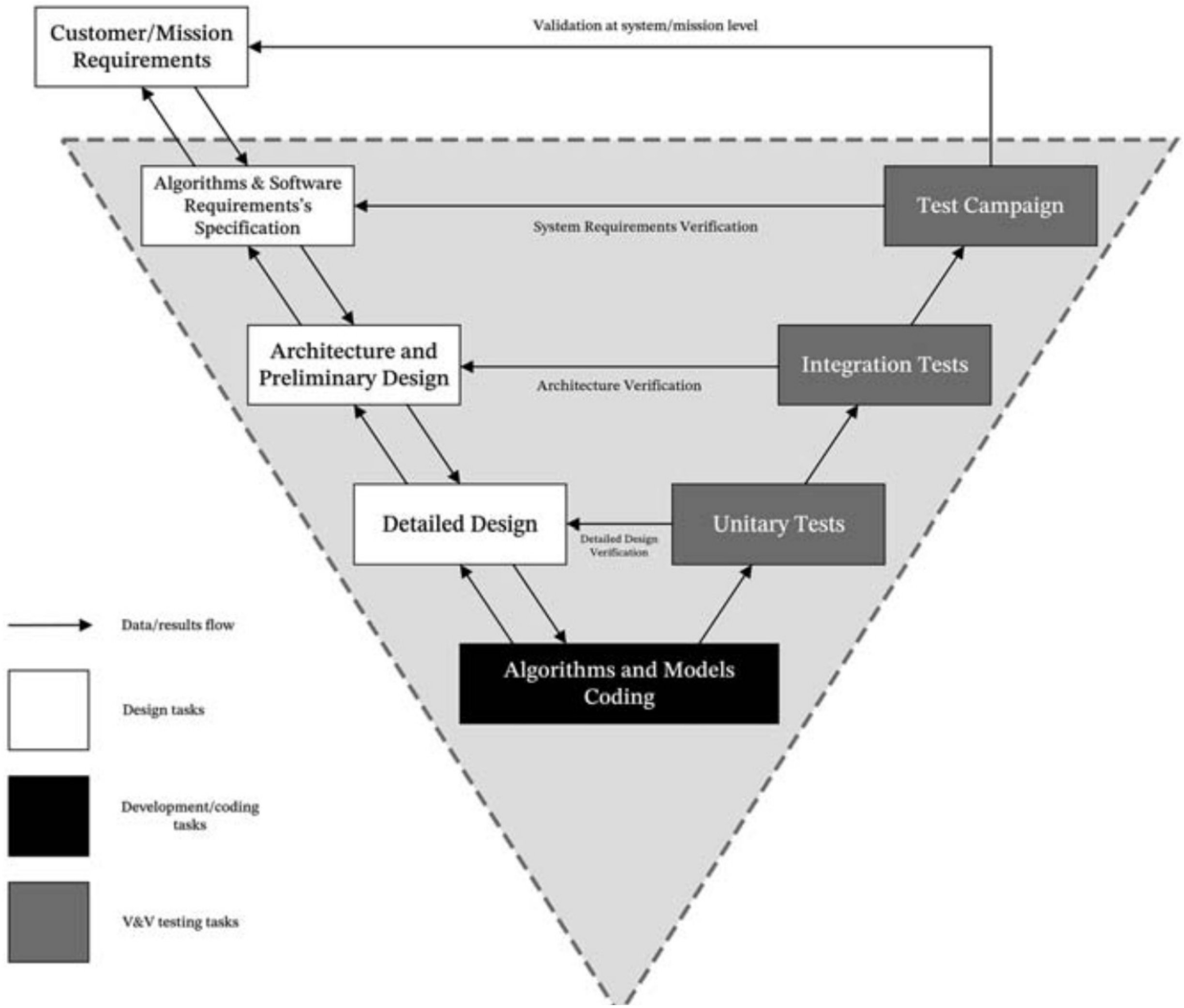
The first two elements (GNC design and performance verification and GNC hardware/software verification) are typically subdivided in the well-known verification phases:

- Model-In-the-Loop (MIL)
- Software-In-the-Loop (SIL)
- Processor-In-the-Loop (PIL)
- Hardware-In-the-Loop (HIL)

The entire process' steps are integrated in an overall iterative process that can be summarized by a classical waterfall scheme. Each elements of this scheme can be iterated multiple times at various levels (e.g., MIL to HIL) until all the GNC requirements are closed and the VCD is complete.

Note that each iteration or step in this process can result in modifications of any part of the requirements and the GNC design. At the same time, early implementations, analyses and advanced testing activities may occur to help in the final definition of the requirements to be closed and satisfied.

GNC Verification: Waterfall Approach



Verification Facilities: Simulators

The **GNC functional simulator** shall be representative of:

- ALL the GNC functions and states,
- the algorithms specified for the on-board software, or directly implemented in hardware,
- the GNC equipment behaviour and performances,
- the satellite dynamics and kinematics,
- the space environment related to the dynamic evolution of the attitude and possibly the position, depending on the mission.
- delays, jitters, discretization, quantization and sampling rates of the GNC loop.

The simulation models of the GNC sensors and actuators implemented respectively in the GNC functional simulator and in the avionics test bench shall be **validated** with respect to the **real hardware** behaviour.

The GNC functional simulators shall give the possibility to **simulate failures and anomalies** to verify FDIR functions.

Verification Facilities: Avionics Test Bench

- The avionics test bench shall include a **hardware model of the on-board computer** functionally representative of the flight model
  - the **numerical precision** of the on-board computer shall be represented and is compared to analysis or simulations performed during the AOCS development process.
- The avionics test bench shall embed the real flight software.
- The avionics test bench shall be possibly interfaced with real flight hardware.
- It shall be possible to introduce a simulation of the forces and torques generated by the GNC actuators in the dynamics model of the avionics test bench.
- The avionics test bench shall be representative of **real hardware interfaces**.
- The avionics test bench shall be representative of the **real-time behaviour**.
- The avionics test bench shall give the possibility to **inject failures and anomalies** during the verification and testing activities.

GNC Design and Performance Verification Requirements: MIL

- The GNC design and performance verification shall be performed through **theoretical analyses** and **numerical simulations** on the GNC functional simulator.
  - It is useful to include the **monitoring of the failures** impacting the GNC functions, with their tuning in the GNC design and performance verification.
- The GNC design and performance verification shall **cover all the GNC modes, functions and mode transitions.**
- The selected approach for the analyses and simulations can use for instance:
  - Monte Carlo method,
  - worst case analysis,
  - sensitivity analysis,

  providing the knowledge of the key parameters and their impacts.
- The GNC design and performance verification shall include a **robustness analysis** covering the **nominal variation range** specified for the physical data and hardware performances, including
  - mass properties,
  - sensor and actuators performances,
  - environmental conditions, orbit uncertainties and drifts.
- The numerical accuracy of the on-board computer shall be analysed in the frame of the GNC design and performance verification.

GNC Hardware/Software Verification Requirements: SIL, PIL, HIL

- The GNC hardware/software verification shall cover **each GNC mode**.
- The GNC hardware/software verification shall test the **mode transitions**.
- The GNC hardware/software verification shall verify functional GNC behaviour with a configuration including the **real GNC flight software** and **representative of flight hardware**, **hardware/software interfaces**, and **real-time performances**.
- The GNC hardware/software verification shall use for **comparison references test cases** coming from GNC design and performances analyses or simulations (**MIL**).
- The GNC sensors shall include a stimulation capability, either physical or electrical, in accordance with the GNC hardware/software verification needs.
- GNC hardware/software verification shall test the GNC equipment in **conditions representative of the mission**.

In-flight Verification Requirements

- Once calibrated, the nominal behaviour of the GNC functional chain shall be verified during the **in-flight commissioning** phase of the satellite.
- A **health check of GNC units after switch ON** shall be performed during the in-flight commissioning phase.
- The in-flight verification activities shall have a duration lower than TBS days, according to mission needs.
- The GNC shall identify in-flight activities contributing to GNC functional chain verification in the verification plan.
  - Some parameters can only be verified in flight.
- In flight verification activities should be (partially) repeated after:
  - Major software update and patches
  - Major GNC anomalies recovery

Aerospace GNC

Advanced Simulation and Testing of GNC Algorithms

A.A. 2023-2024

Andrea Colagrossi

POLITECNICO MILANO 1863