POLITECNICO DI MILANO

The Space domain: a chance for MAS applications?

M.R.Lavagna

Dipartimento di Scienze e Tecnologie Aerospaziali

Michelle.lavagna@polimi.it

Outlines

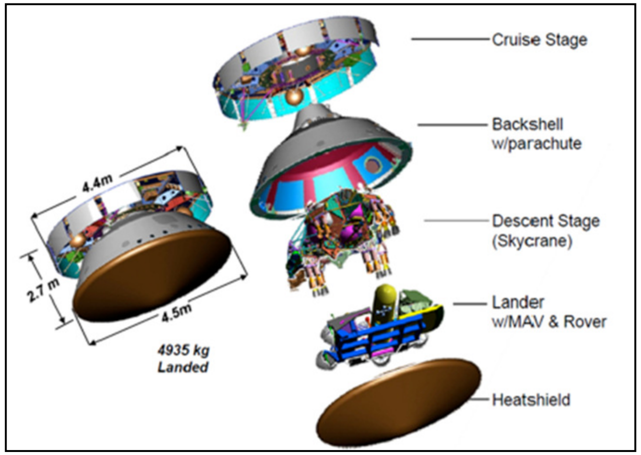
1. Actors in space: the space mission domain building blocks
2. Possible areas for agents' technology: Autonomy & Complexity
3. State of the art: Single Agent examples
4. Distributed scenarios in space
5. Potential MAS technology exploitation scenarios
6. Conclusions

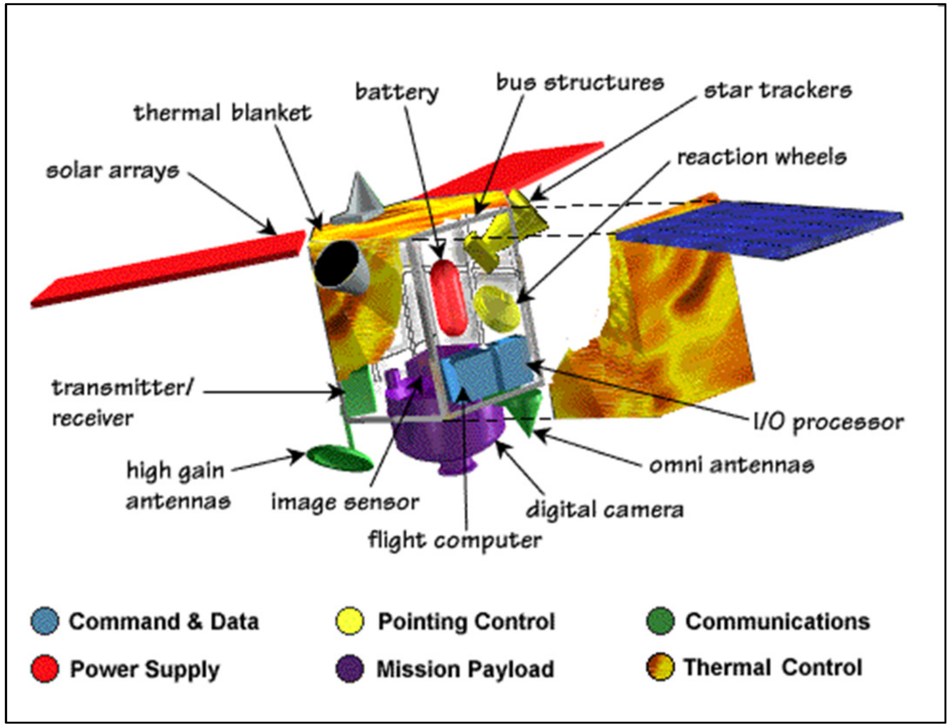
POLITECNICO DI MILANO
Aerospace Dep.

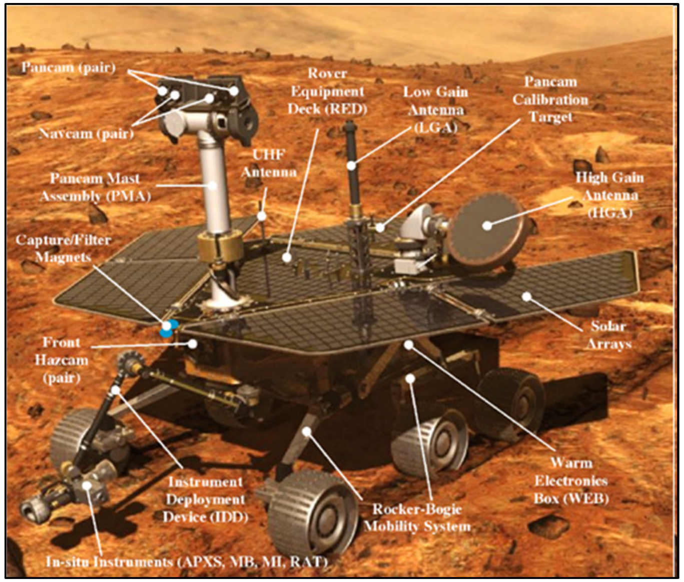
Actors in space: the space mission domain building blocks

The space system

- Complex
- Multi-disciplinary
- Generally mission dependent
- Highly constrained
- Cost function drivers





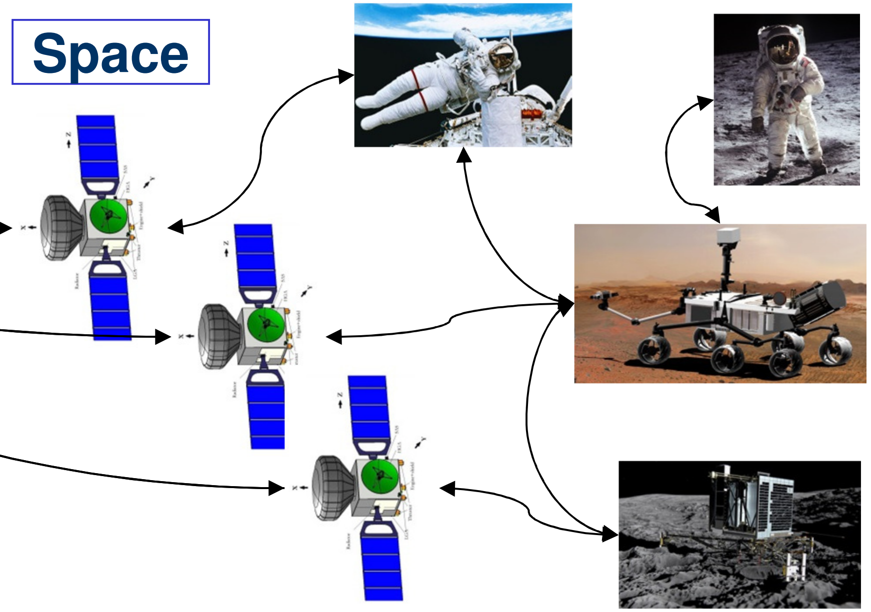


POLITECNICO DI MILANO
Aerospace Dep.

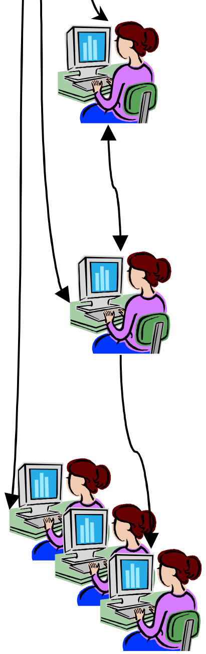
Actors in space: the space mission domain building blocks

Ground

Space



Ground Stations
Data collection/command sending



Mission operation centre
Space segment control timeline definition and setting

Science Operation centre
On board payload activities planning/analysis/distribution

End users
Data buyers/scientific community/etc

The space mission scenario

- Complex
- Multi-disciplinary
- Generally mission dependent
- Highly constrained
- Cost function drivers
- (Fast) dynamics involved
- Humans in the loop
- Different time scales

Reasoning/Decision Making areas

- Design phase: system and mission
- Operational phase: system and mission

Reasoning/Decision Making areas: design phase

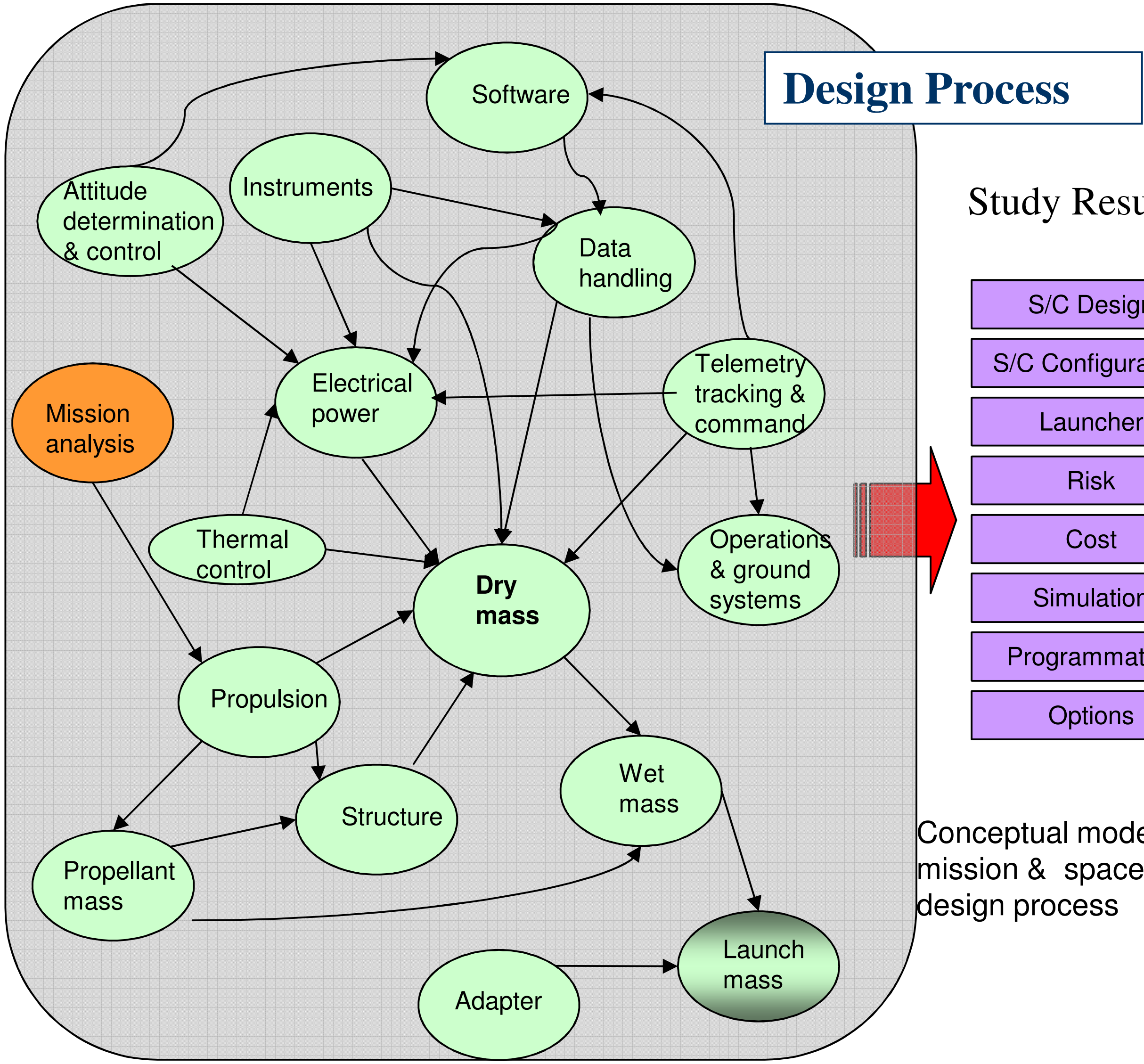
POLITECNICO DI MILANO
Aerospace Dep.

Mission Requirements & Constraints

- Objectives
- Environment
- Lifetime
- Payload
- Reliability
- Schedule
- Technology
- Budget

Study Requirements

- Products
- Study Level
- Planning
- Resources



Study Results

- S/C Design
- S/C Configuration
- Launcher
- Risk
- Cost
- Simulation
- Programmatics
- Options

Conceptual model of mission & spacecraft design process

Courtesy from ESA/CDF

Reasoning/Decision Making areas: design phase

Problem features:

- Large variables domain: subsystem sizing/device selection
- Discrete/continuous domains
- Large net of analytical constraints
- Multiple clashing design drivers
- Numerous disciplinary models/tools
- Multiple experts involved one System Engineer

goal • alternatives space reduction to no more than two consistent solutions in terms of system and mission design

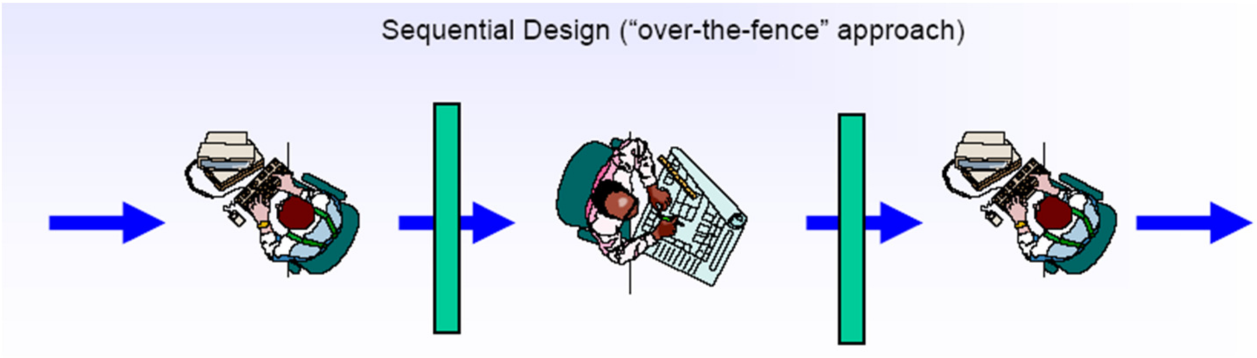
Needs

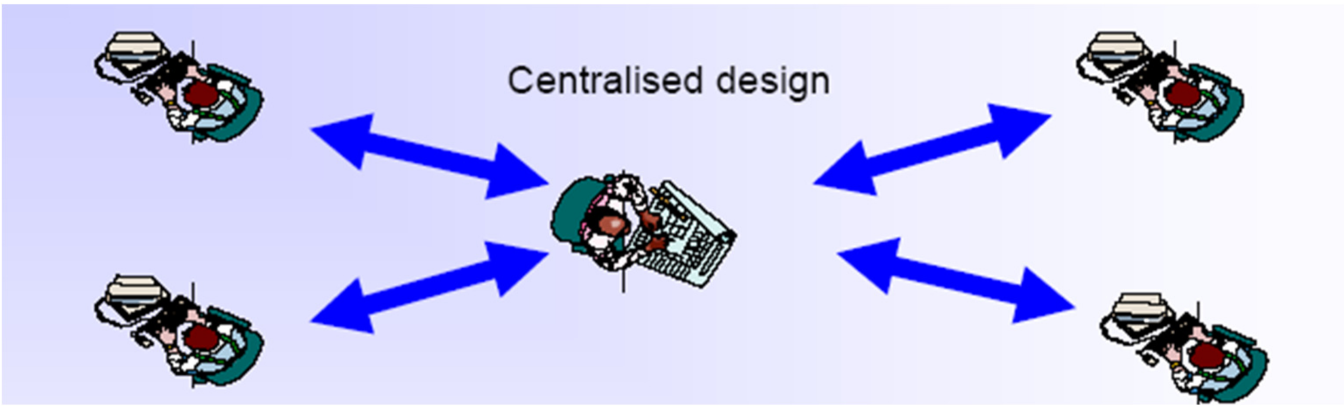
•Automatic tool
•Designers support

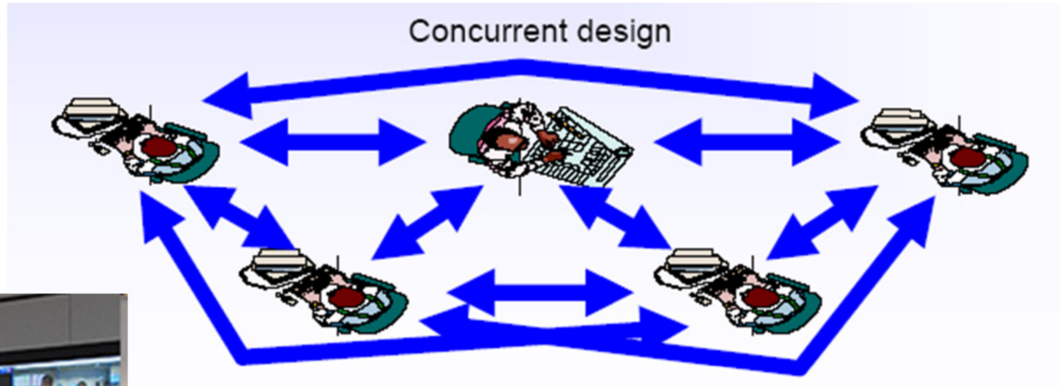
Problem solving tools/approaches

- MCDM
- MADM
- Multi-objective Optimization
- Global optimization
- Multi-Disciplinary Optimization
- Game Theory

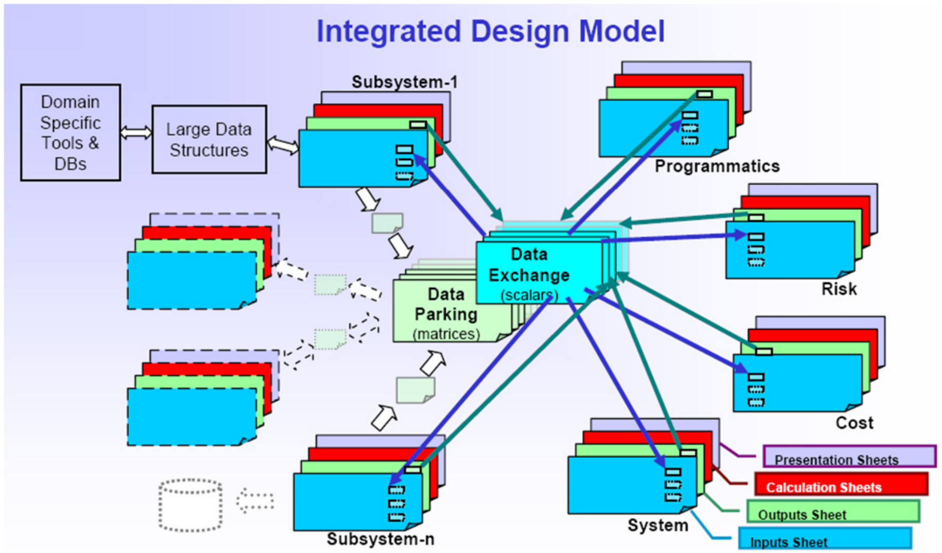
Currently applied solving strategies







Courtesy from ESA/CDF



POLITECNICO DI MILANO
Aerospace Dep.

High level: deliberative skills

Decision making on:

- goal(local/global) → Scheduling/planning/Navigation
- Strategies to get the goal → Diagnosis/Reconfiguration
- Uncertainties/unpredictable events coping

Low level: reactive capabilities

Decision making on:

- control law → Uncertainties management
- Strategies for the basic behaviours coordination → Recovery/Robustness/reactiviness

Problem features:

- Features are segment dependent/mission dependent
- Large variables domain: subsystem/system state vector in time
- Discrete/continuous domains
- Different operational phases differently constrained
- Large net of temporal/logical constraints
- Different and complex resources/resource availability dynamics
- Both hierarchical and peer-to-peer architectures of identical problem to face
- Incomplete environmental and behavioural knowledge

goal
- robust actions sequence/control design over the mission timeline
- opportunistic science

Needs

- Autonomous/automatic solvers
- Operators' DM support

Problem solving tools/approaches

- MCDM
- MADM
- CPS
- COP
- Soft computing/heuristics
- Global optimization

Actor	tasks	Constraints/resources	Variables
Ground Stations	s/c contacts management for data downlink/control sequences uplink	• Pointing capabilities • Frequency bands • # of visible s/c • # of channels • # of antennas • Temporal boundaries • s/c visibility windows • Ground system functional constraints	• Antennas • Frequencies/channels/bands • Visibility windows • #of data packages
Mission Operation centre	• Short/medium/long term planning/scheduling generation for the platform • Long terms p/s platform/p-l harmonization	• On board power/memory/ • electrical /mechanical energy/fuel/devices • On board functional constraints/dynamics/consumption • Visibility windows • GS network	• Devices • Action Instantiation time • System/ sub-system functional modes

Reasoning/Decision Making areas: operational phase

Actor	tasks	Constraints/resources	Variables
Science Operation centre	Long terms planning on instruments generation/ harmonization	• Visibility windows • Science needs • Science instruments • Functional constraints	• Instruments • Activity time instantiation • scientists
End users	Downloaded data exploitation/management		
Space segment	Payload data generation and download/safe survival	• On board power/memory/ • electrical /mechanical energy/fuel/devices/computational capabilities • On board functional constraints/dynamics/consumption • Visibility windows • GS network • Real environment	• Devices • Activities instantiation time • System/ sub-system functional modes • High level goals

POLITECNICO DI MILANO
Aerospace Dep.

Autonomy

system state vector/environment conditions **perception capabilities**

+

Acting skills on the system state vector to:

•Accomplish either partially or globally mission goals

•Assure system safety and survival

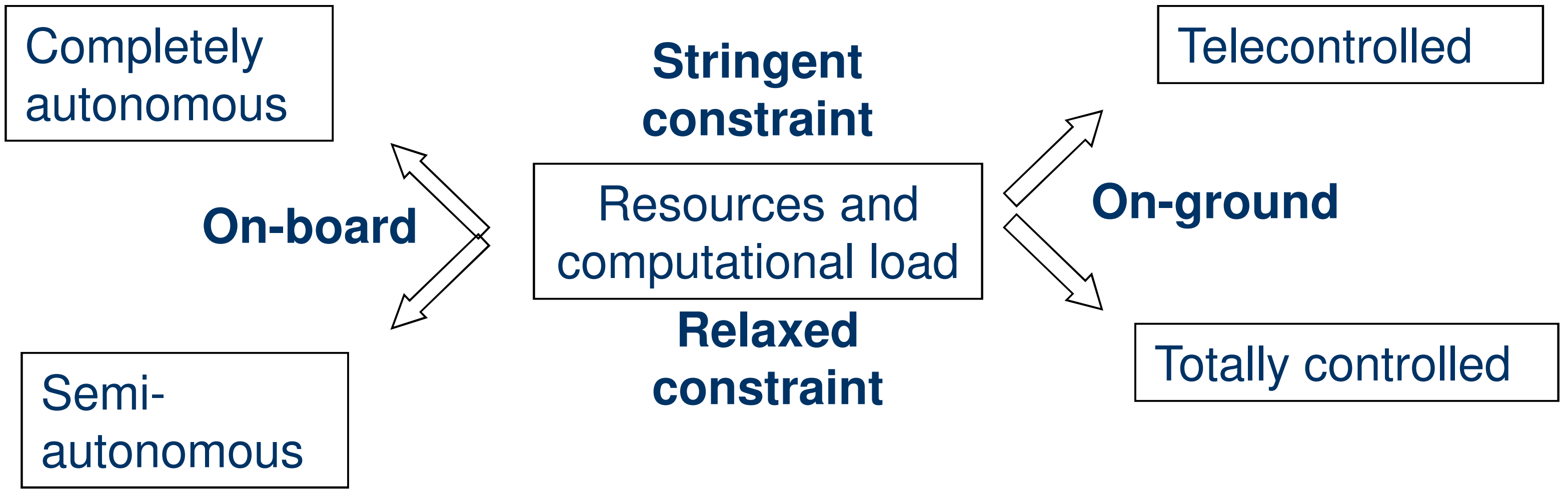
Low level ⇨ **control**

High level ⇨ **decision making problem solving**

POLITECNICO DI MILANO
Aerospace Dep.

Autonomy in space may be:

- On-board
- On-ground
- spread over on-board+on-ground



- confined to specific levels
- limited to specific operational phases

POLITECNICO DI MILANO
Aerospace Dep.

Autonomy represents a powerful tool to

→ **better exploit** robotic systems **performance**
→ **enhance** robotic systems **productivity**

Robots need
automation→ to manage and control repeatable low level operations

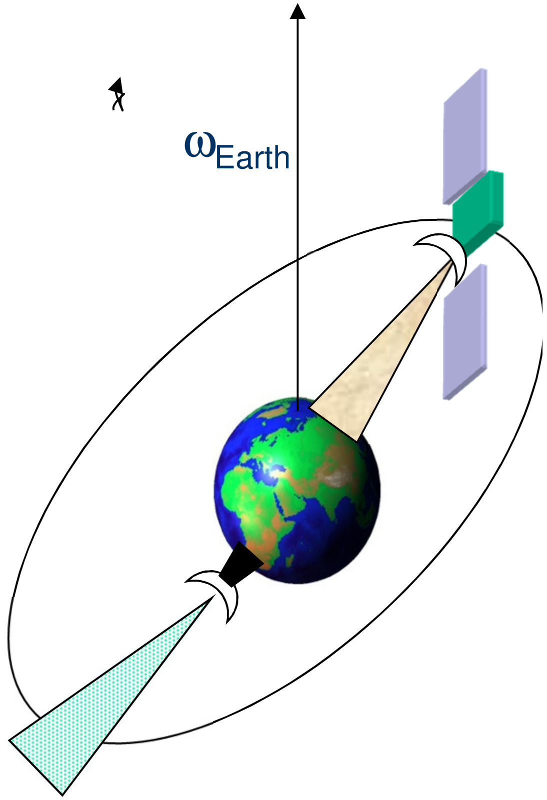
Intelligent robots exploit
autonomy→to enable low/high level operations in changing/unknown environments

Intelligence entails robots to be provided with reasoning/Decision-Making mechanisms which offer:

- **Flexibility**: unpredictable events, not pre-designed situations become manageable; failed systems still produce
- **Timely response**: idleness is avoidable as the system senses and rapidly reacts
- **Adaptation**: better suited behaviour/control according to the actual sensed environment is achievable; limited resources allocated at best to maximise the robot efficiency

Autonomy: Why?

- **Earth Orbiting Missions**:
  - Ground station load ↑↑↑
  - Finite opportunity for contacts
  - Timeliness/reaction ↓↓↓



- **Interplanetary/exploration missions**:
  - Tx/Rx time span O(2R)
  - R→O(10^8Km!) e.g. $2^*T/R_{Mars}$=8.3 min $2^*T/R_{jupiter}$=1.16h!!
  - Unknown environment →reactivity/adaptivity needed
  - System complexity

POLITECNICO DI MILANO
Aerospace Dep.

Autonomy in space would support and cope issues for:

- Complex mission scenarios
  Formation flying (Darwin, Lisa, Copenicus, Swarm, Prisma,Proba 3, Galileo)
  Multimodule missions (ExoMars, Bepicolombo, MSR)
  cooperative heterogeneous entities (Human bases)
- Very far missions → comms delays→ mission return degradations and lack of robustness (Asteroids missions, Laplace, Cassini-Huygens like)
- Complexity in space operations
  Docking/RV, In space Assembling, human bases-Astronauts supports
- Huge scientific data managements: possible multiple p/l, possible timeliness need in elaboration
  Earth monitoring/protection missions (Copenicus, Galileo, Envisat like)
- Mission with harsh and unknown environment
  Exploration missions to NEO, Mars, Moon, Comets
- Limited on-ground resources scenarios
  Mission control centre bottleneck support
- Anomalies & unpredictable events/opportunities management

Autonomy in space: levels definition

Autonomy in space is currently studied, but rarely applied

The implementation of on-board autonomy depends on the specific mission requirements and constraints, and can therefore vary from very low levels of autonomy involving a high level of control from ground to a high level of autonomy, whereby most of the functions are performed on-board.

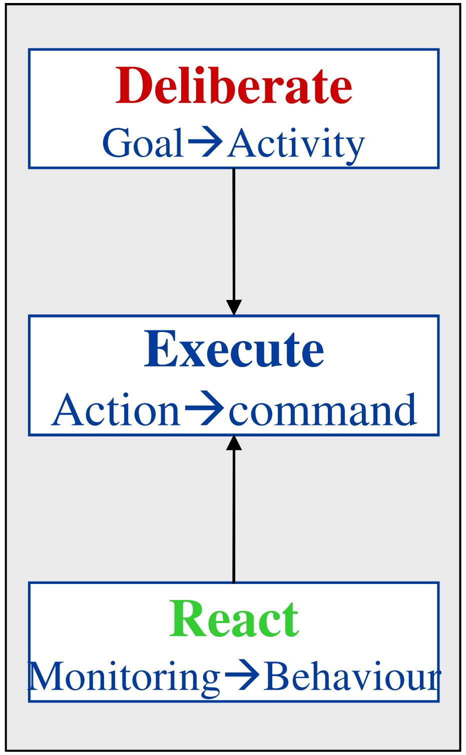
Autonomy levels and criticalities

Level	Description	Functions
E1	Mission execution under ground control; limited onboard capability for safety issues	Real-time control from ground for nominal operations Execution of time-tagged commands for safety issues
E2	Execution of pre-planned, ground-defined, mission operations on-board	Capability to store time-based commands in an on-board scheduler
E3	Execution of adaptive mission operations on-board	Event-based autonomous operations Execution of on-board operations control procedures
E4	Execution of goal-oriented mission operations on-board	Goal-oriented mission re-planning

Agents

Courtesy of ESA-ECSS

The Agent building blocks



DELIBERATIVE LEVEL

- Produces the high level decisions at sytem level (goals);
- Deliberates activities exploited to obtain a long-term control strategy to satisfy high level goals according to the system/environment physical constraints
- States for failures occurrence and recovers

INTERMEDIATE LEVEL

- Turns actions into real commands to the hw coping with possible uncertainties rising from unknown and dynamic environment

REACTIVE LEVEL

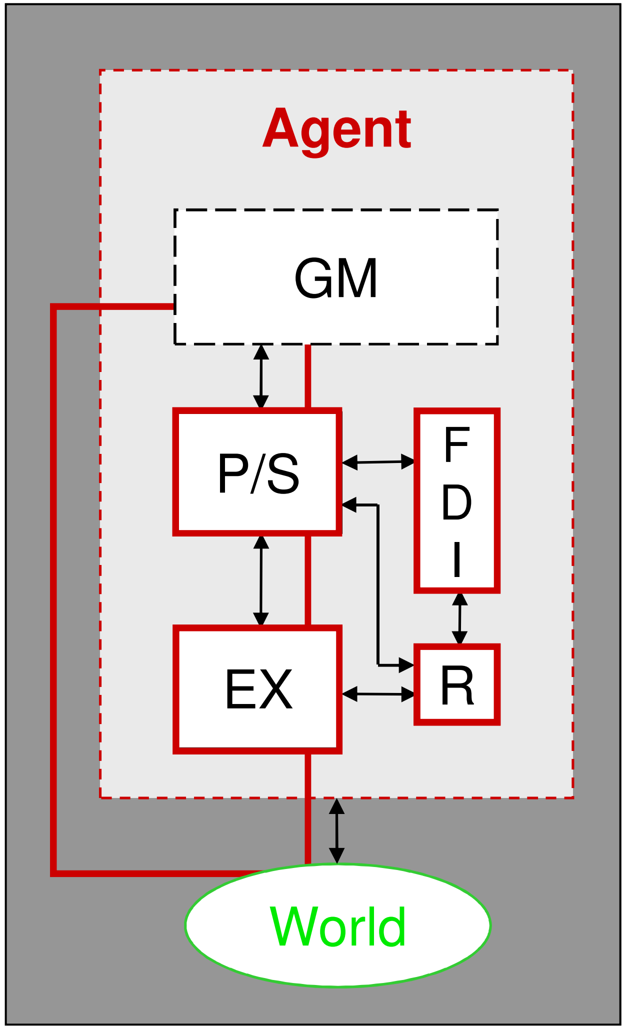
- Monitors the actual system/environment conditions and identifies – within the short incoming time span –the set of commands consistent with them

The Agent building blocks

- Goal Management
- Planning/Scheduling
- Execution
- Failure Detection & Identification/Reconfiguration

Partial architectures can be selected depending on the applicative scenario

- **Deliberative:** based on the symbolic reasoning needs a world representation
- **Reactive:** no knowledge of the world is needed
- **Hybrid:** focused on merging the benefits of both the deliberative and reactive architectures



The Deliberative Agent

Goal generation

- Aim: to identify the best/most convenient particular state of the world to satisfy given criteria
- Control variables: system state/environment
- Constraints: feasibility

Goal generation is a Decision Making/Opt problem on system states/resources

Planning

- Aim: to reach a particular state of the world, starting from the current one
- Control variables: actions to be sequenced (i.e., a plan) to achieve the objectives
- Constraints: pre, post or during conditions among selected activities in the plan

Planning deals with system states, and their changes & logical dependencies

Scheduling

- Aim: To select the feasible plan that satisfies constraints related with time
- Control variables: Timelines of activities
- Constraints: Resource limitation and temporal constraints

Scheduling deals with allocation in time and resources exploitation

Failure Detection/identification

- Aim: To classify deviation as failures/to identify faulty units
- Control variables: Timelines of activities

Constraints: functional dependencies in the system

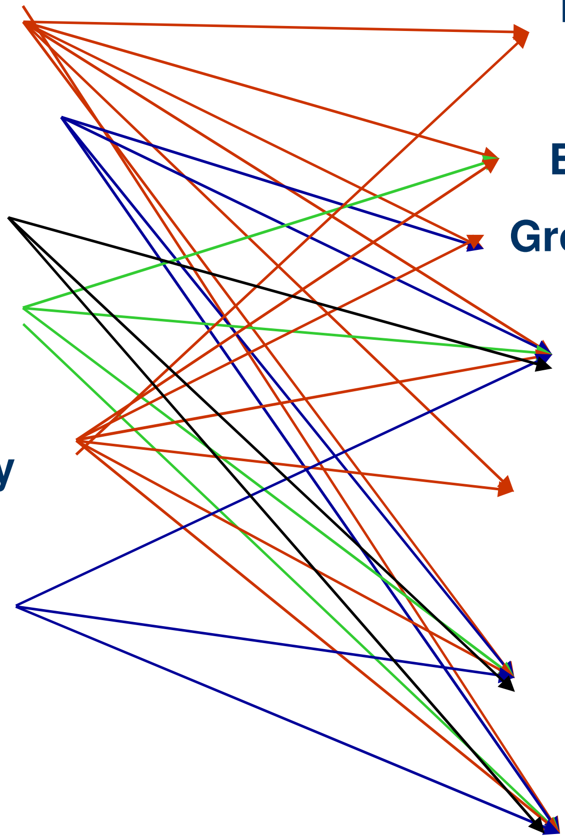
FDI deals with modelling robustness and abductive reasoning

AGENT in space for AUTONOMY

POLITECNICO DI MILANO
Aerospace Dep.

Functionalities

- Science (Re)-planning
- Resource management
- Mission management
- Timeline management
- Failure detection & Identification & Recovery
- Opportunity exploitation\reaction



Scenarios

- Deep Space Navigation & Control
- Entry Descent & Landing
- Ground stations automation
- Planetary Exploration
- Rendez-vous & docking maneuvers
- Coordination of multiple segments (humans included)
- Interplanetary probe\ Earth spacecraft

The Deliberative Agent

Pro-activeness → **Deliberative**

<u>Top-down reasoning</u>

→ large problems/largely constrained managed
→ wide time horizon managed
→ large knowledge to be uploaded (e.g.model needed for resource propagation)

The deliberative paradigm is greatly exploited to get rid of ***decision making***

Pro's

- Forecasting skills
- Decision making supported by a global point of view

Cont's

- Problem solving limited to situations included in the domain of experience
- No reactive behaviour to environment exists

Methods

MCDM/MADM
CSP
COP

Soft computing/heuristics
Global optimization/MDO
Graph theory
Logical reasoning

The Reactive Agent

Reactivity → **Reactive**

<u>Bottom-up reasoning</u> → actions focused on perception
→ very short time horizon managed
→ very limited knowledge required

Pro's

- adaptive to the system current status
- robustness+flexibility+time
- learning from experience capabilities

Cont's

- lack of global vision in time→local point of view
- strategies to be defined for complex reasoning

Methods

MCDM/MADM

Soft computing/heuristics
Global optimization/MDO
Behaviour based reasoning

The Agent's architecture comparison

Architectural aspects\General scenarios	Reactive	Deliberative	Hybrid
Known environment	Useful if no P/S functions are required (or simple tasks/goals have been implemented)	Correct choice in P/S functions are required	Maybe too much complex for this case
Unknown environment	Well suited even it can involve a huge list of behaviours in case of very complex and unknown environment	Not suited because planning is (rather) impossible and not effective	It could be used according to the flexibility and robustness of the agent towards the environmental uncertainties
Timeliness required	Applicable only when the tasks and the actions to be performed, and the goals to be accomplished are extremely simple	Well suited in case of perfect model of the world (extremely high confidence in actions execution by the agent)	Hybrid agent is well suited because guarantees short-term execution and long-term deliberation
Long term vision required	Not applicable	Correct choice	Correct choice if the agent has long-term deliberative faculties
Autonomy level	Low level of autonomy (i.e. directly controlled or semi-controlled robot) or fail-safe functions	High level of autonomy, but with a perfect knowledge of the world (ideal case).	High level of autonomy in a very complex situation and real environment

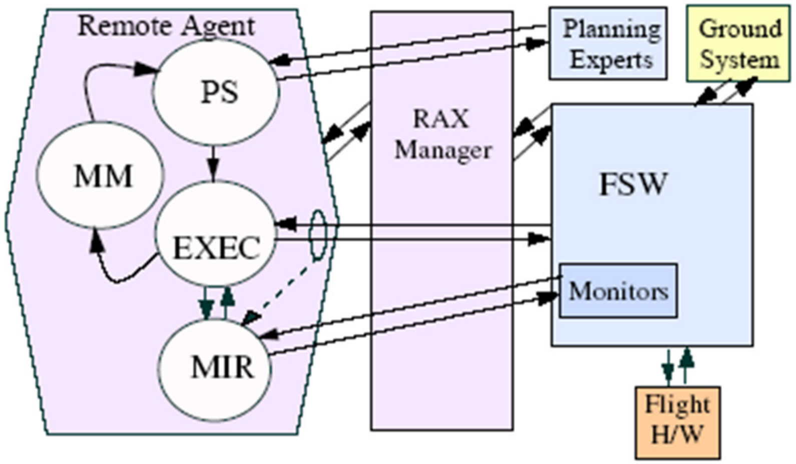
- **Deep Space 1 (DS-1)→** Braille Comet AUTONOMOUS
  - ⇨ NASA mission, technology demonstrator
    - → Launched in1998
      - ➔ On board Autonomy
      - ➔ Electric propulsion
  - ⇨ <u>Autonomous Navigation </u>(AutoNav)
    - → State vector identification→autonom
  - ⇨ <u>Remote Agent</u>
    - → Planner/scheduler
      - ➔ Thrusted maneuvers
      - ➔ Payload management
      - ➔ Reconfiguration
  - ⇨ <u>Livingstone</u>
    - → Failure detection Identification and Recovery (FDIR)→autonomous

Deliberative

POLITECNICO DI MILANO
Aerospace Dep.

Remote Agent-RAX

- Goal-oriented
- Hierarchical decomposition based
- Iterative repair based
- Model based
- Event driven



Remote Agent-Aspen

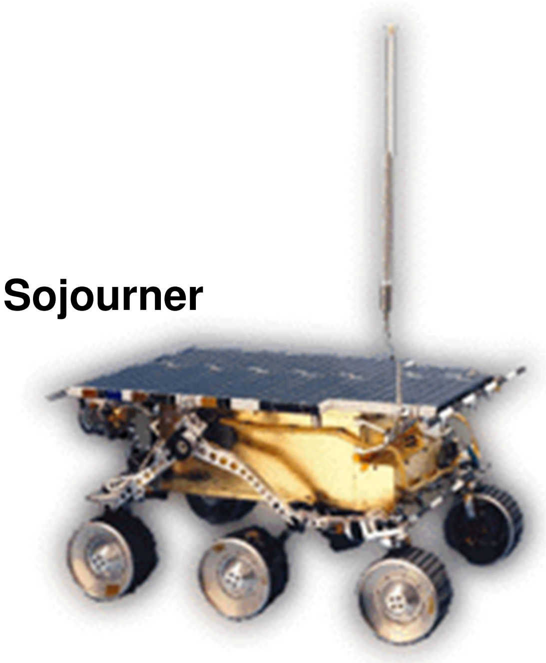
- P/S
- Goal-oriented
- Iterative repair based
- Highly reconfigurable
- Constraints modelling/managing language

POLITECNICO DI MILANO
Aerospace Dep.

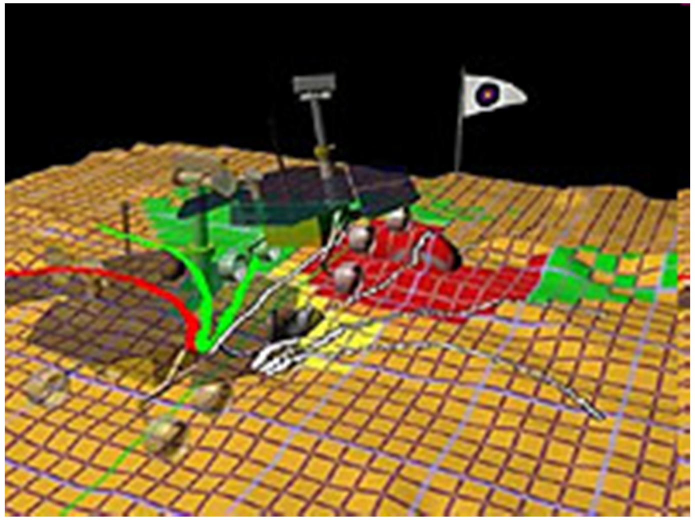
- **Earth-Observing 1** (EO-1) AUTONOMOUS
  - NASA mission, technology demonstrator
    - ⇨ Launched in2000
      - ➔ Multispectral instruments
  - ⇨ Autonomous P/S
    - → Aspen for payload utilisation
  - ⇨ Sensor Web
    - → collaborative scientific data merging/web based

- **Mars Pathfinder**
  - NASA mission for Mars exploration SEMI-AUTONOMOUS
  - ⇨ Launch 1996
    - → First Mars rover
    - → Telecommanded waypoint for navigation
    - → <u>Autonomous obstacle detection</u> and avoidance

Sojourner



- **Mars Exploration Rovers**
  - NASA mission for Mars exploration SEMI-AUTONOMOUS
  - ⇨ Launch 2003
    - → Given goals; Autonomous safe Navigation/obstacle avoidance (Path Planning)
      - → Stereo vision→DEM
      - → Traversability map
      - → Path planning
      - → Drive



MAS in space: distribution why?

Earth & Interplanetary missions

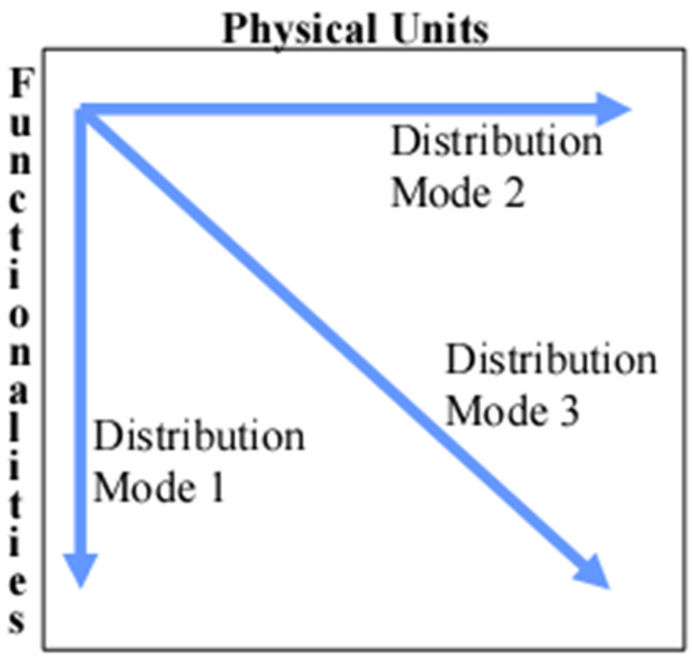
- Reliability/robustness increase
- mission flexibility increase
- mission objective satisfaction
- sometimes easier to be integrated/realized

MAS in space: distribution scenarios

- The **Physical Distribution** implies the deployment of a MAS architecture across multiple physically distinct platforms
- the **Functional Distribution** is such that, a MAS architecture is used to perform the different functions required by a single system, by dedicating an agent to each function or task.

agent technology can be distributed in terms of:

- Intelligence
- capabilities
- communications
- resources



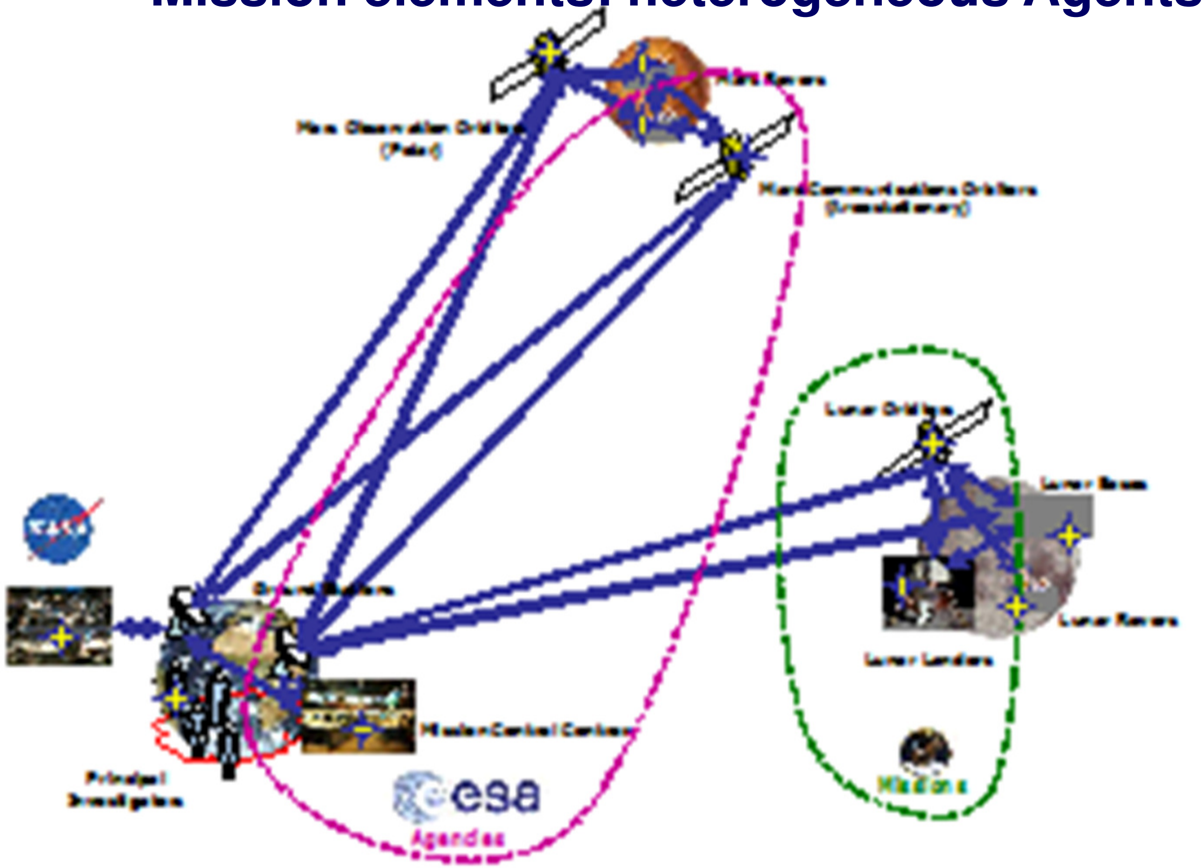
The Multiple Agent's architecture comparison

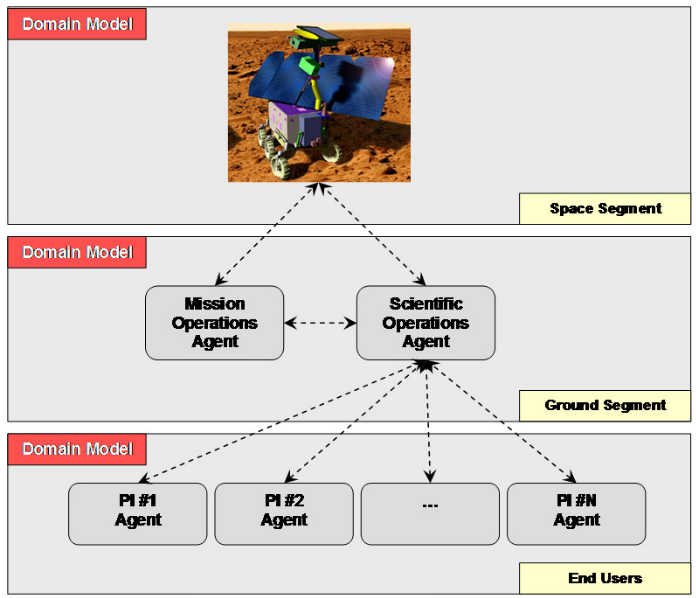
Architectural aspects\General scenarios	Communication	Auction & Negotiation	Organisation
Physical distribution	Extremely important for reaching coordination. Mean of communications to be carefully evaluated (radio-link, internet/ethernet, …).	Negotiation becomes relevant in case of private resources to be used by other agents or by the agency, or in case of instruments to be physically shared, otherwise distribution is not demanding.	The organization pattern is extremely important and could have an added value in this case. Physical distribution helps in choose the pattern (i.e.: mimicking the real-world structure of the agency)
Functional distribution	Communication protocols must be established (in general they are the interfaces among the logical units of the MAS), but they are not demanding because in general there is a unique physical entity.	Coordination is important because different functions (i.e. agents) could be in conflict among each agents, or they could use the same physical resources.	Organisation in general maps the logical relationships among the functions/agents, thus keeping the hierarchies and the dependencies.

The Multiple Agent’s architecture comparison

Architectural aspects\General scenarios	Communication	Auction & Negotiation	Organisation
Limited resources	Extremely relevant in case of a limited number of messages or a limited information flow between the agents; otherwise communication protocols and languages are only functional for reaching the agreement on shared resources.	Extremely relevant issue for the optimization of the limited resources.	The choice of the good MAS pattern could help in optimizing the usage of limited resources or in reaching coordination.
Unlimited resources	Not relevant issue	Negotiation and auction may be useless because resources could be freely exploited; coordination is always relevant when two or more agents execute different actions at the same time or when there are other constraints.	Organization could be extremely important with unlimited resources because could give an added value in the reaching of the common goal (e.g.: could be a pathfinder for the limited resources case).

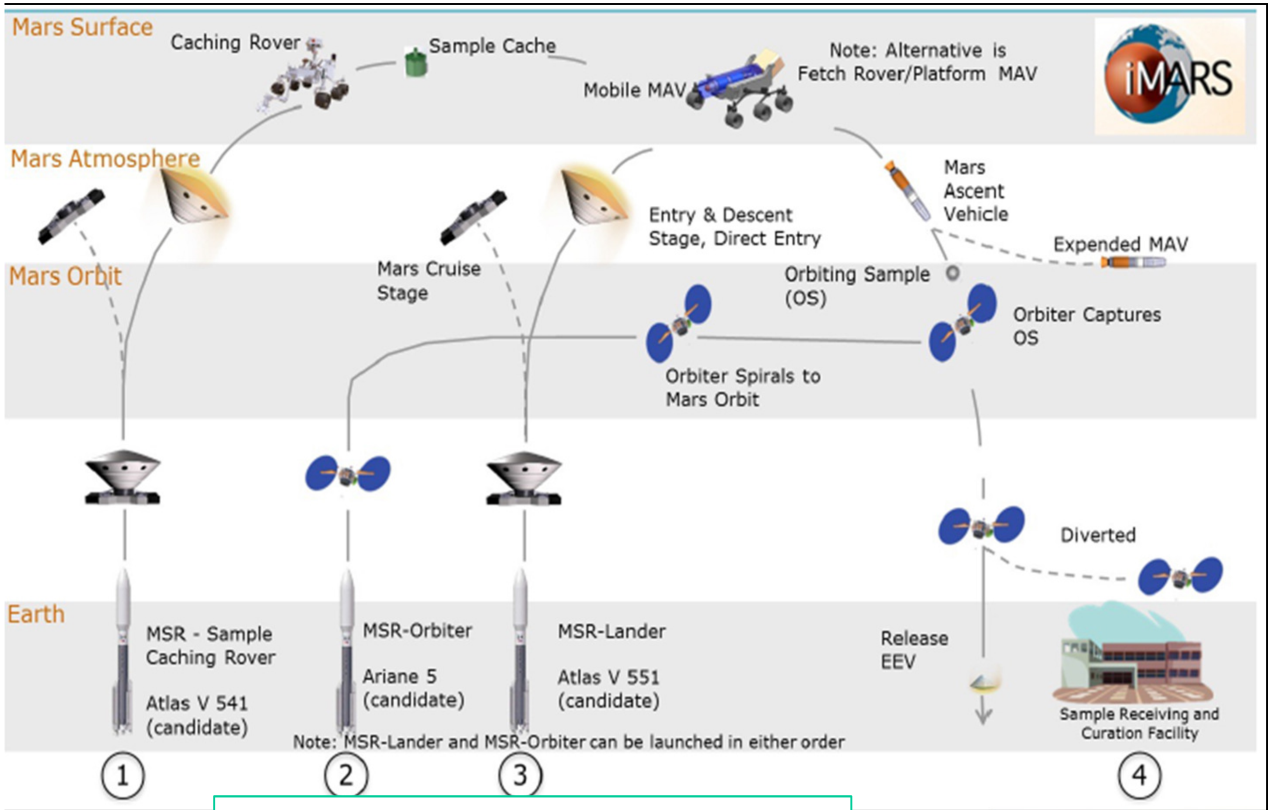
Mission elements: heterogeneous Agents, nested architectures





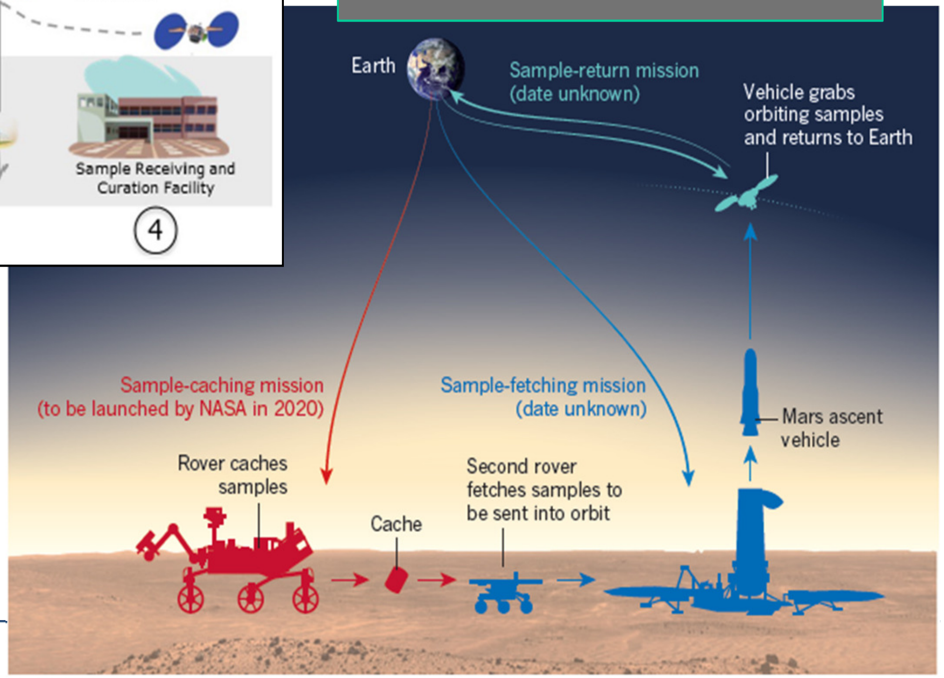
Mars Sample Return
Mission Architecture: example

POLITECNICO DI MILANO
Aerospace Dep.



Overall mission architecture

On surface activities\elements



POLITECNICO DI MILANO
Aerospace Dep.

Formation Flying/Swarms

of units N=[2;100] thight control on relative dynamics→hard constraint → on-board autonomy needed

In orbit missions: EO-1+Landsat-7; GRACE(DLR); LISA (ESA); SMART2 (ESA); DARWIN(ESA); Proba 3;Terrestrial Planet Finder(NASA);ST5 Nanosat(NASA), Swarm (ESA)

Constraints: hard→Formation geometry/relative dynamics

→available local/shared on board resources

→ real-time

→ common high level goals (scientific data collection, maneuvers, etc)

FF autonomous management: centralized (hub)←→distributed

Requisites: flexibility/robustness

Challenges: local ↔local consistency ⇒negotiation, comms strategies, knowledge bases management and distribution, very limited computational resources

Ground station nets

1 Mission Control Centre → N Ground stations devoted to:

Tracking/Ranging orbiting systems

Telemetry/data Rx

Telecommands Tx

ESTRACK: Darmstadt (MOC), 7 Ground stations (Redu 10 Antennas, Villafranca 8)

Constraints: hard→orbiting systems/antennas relative dynamics (visibility)

→shared technical/financial resources

→ high dimension problem

→on-board functional constraints

Autonomous/smart management: centralized/disributed (hetero/homo-geneous nets)

Requisites: flexibility/robustness possible human interaction

POLITECNICO DI MILANO
Aerospace Dep.

Team of Robots

Robots: rover flottillas/UAV from 10 to 100

Scenario: planetary exploration / Location setting for human habitat→autonomy

Elements: heterogeneous/homogeneous

Constraints: →common/coordinated goals/tasks

→ local/shared resources

→ relative dynamics

Autonomous management: centralized←→distributed; hierarchical/peer-to-peer

Requisites: flexibility/robustness/reactivity

Issues: unknown environment; comms management; possible human interaction; learning needed

Single Vehicle

Scenario: planetary surface vehicle exploration/satellite

Agents: cluster of on-board subsystems/cluster of on board functionalities

Goal: robustness/flexibility increase; system product return increase

Constraints: →system functional model

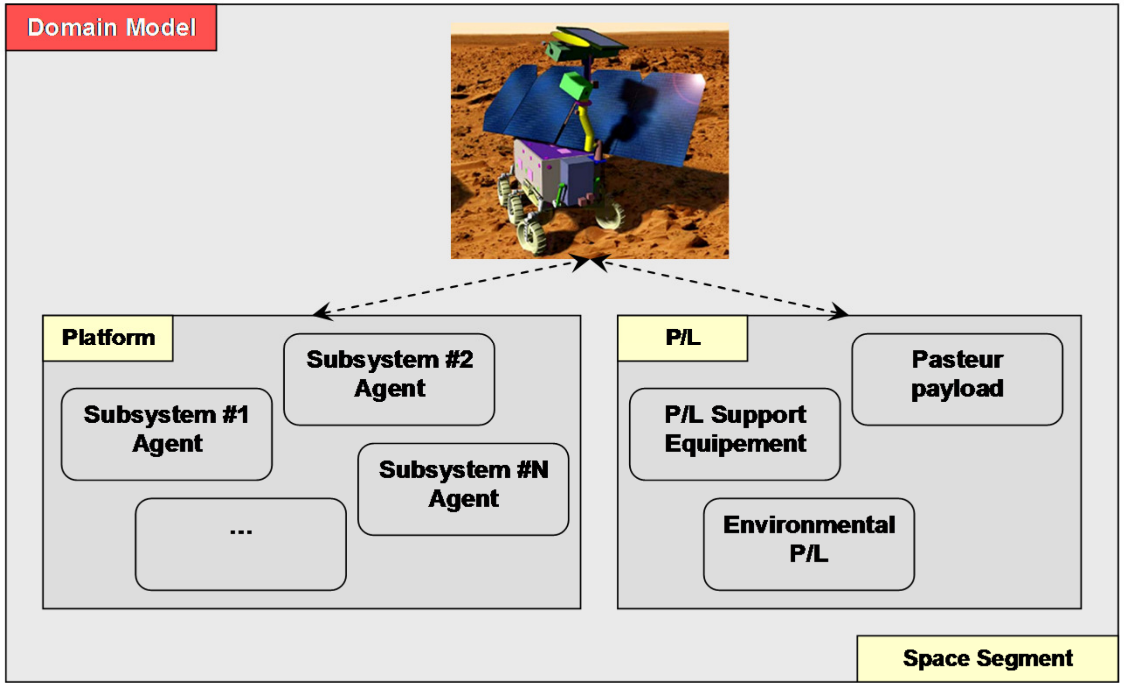
→ shared/local resources

Interaction strategy: competitivw/collaborative

Issues: comms; interfaces; architecture selection

POLITECNICO DI MILANO
Aerospace Dep.

Single Vehicle, Multiple Agents



Design phase

Space System: orbiter/lander/robot etc

Scenario: Concurrent Engineering Process

Agents: Subsystem discipline

Constraints: →design relationships inter/intra-disciplines

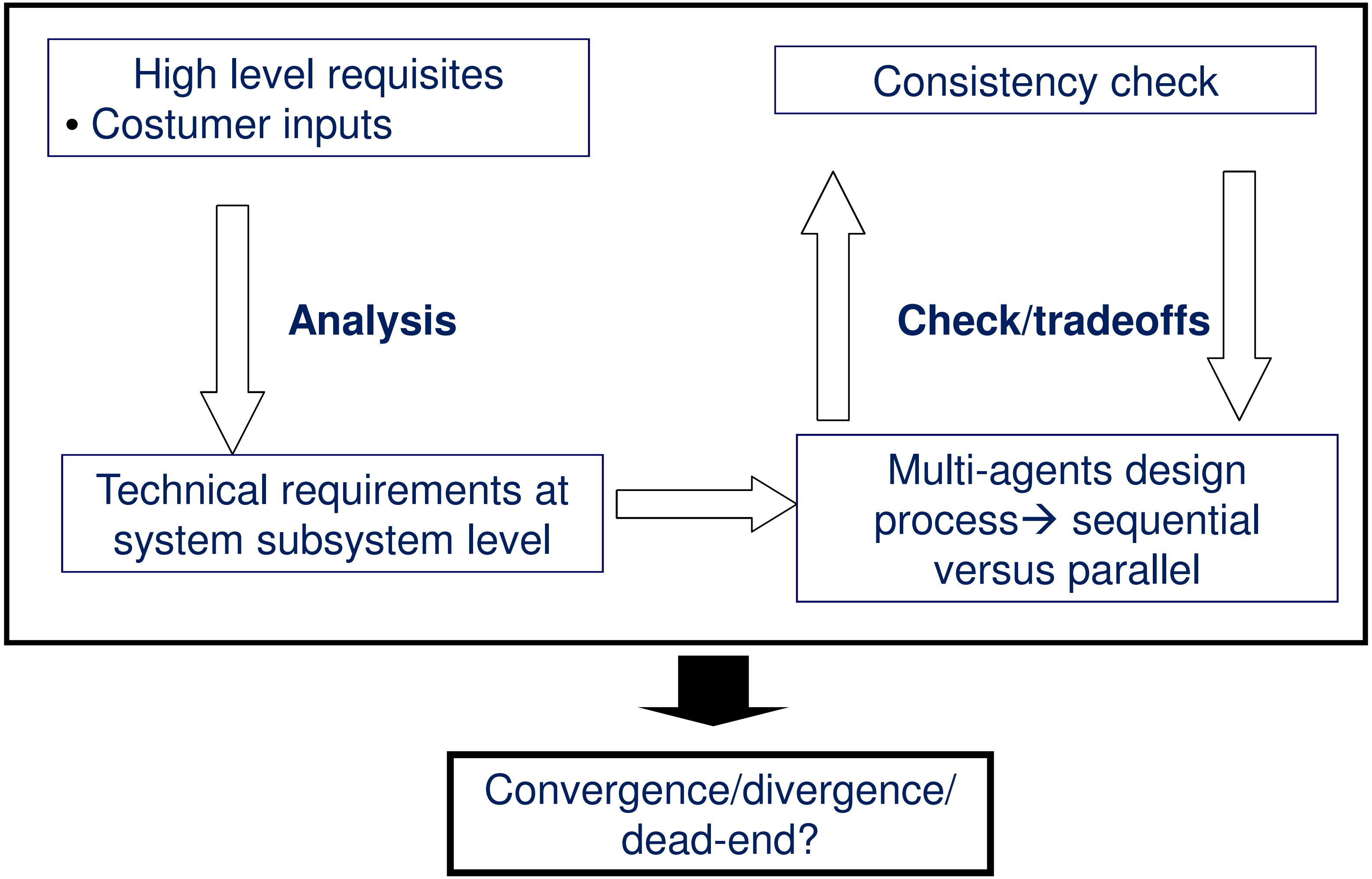
→ temporal/financial/technological resources

Architecture/Interaction:
centralized←→distributed,competitive←→collaborative

Requisites: robustness/reactivity/flexibility

Issues: Strongly coupled design among disciplines; human behaviour in decision making to be possibly modelled

Distribution: where?



POLITECNICO DI MILANO
Aerospace Dep.

Controlled dynamics: control profile

→centralized generation

→strongly coupled with the state vector of any other distributed element in the formation

Functional model: pre-post conditions exist inter-elements too

POLITECNICO DI MILANO
Aerospace Dep.

Limited - Shared – general/scenario dependent

Not depletable:

Electric power →system dynamics dependent

Depletable:

- <u>renewable</u>:

  On board memory

- <u>renewable and history dependent</u>:

  - stored electric energy
  - Angular momentum

  →highly dependent on activity allocation in time

- not <u>renewable:</u>

  fuel, time, financial support

- **Unary:**

  Instruments/devices

Final remarks

•Within the space domain to study and apply autonomy and distribution is an **obliged step to answer mission goals** tighter and more challenging

•**Single Agent** architectures seems to be the well-suited tool especially within the operative phases **to answer flexibility, feasibility and robustness** requirements

•**Multiple Agents Systems** may answer **a wide class of issues and bottlenecks** the space engineering community is facing; this technology, however, posts further challenges in the communication management, the knowledge sharing and the common constraints resource management and need the single agent technology to be firstly exploited and validated in the space activities framework.