Chapter 25

Spacecraft Autonomous Reaction Capabilities, Control Approaches and Self-Aware Computing

Klaus Schilling, Jürgen Walter, Samuel Kounev

Abstract Space exploration missions require very challenging autonomous reaction capabilities, as spacecraft have to react appropriately to the partially unknown environment in time critical situations. Here, direct human interaction is often impossible due to significant signal propagation delays related to the huge distances. We discuss existing solution strategies for autonomy in space and exemplified by the missions CASSINI-HUYGEN (landing on the Saturnian moon) and ROSETTA (the accompanying and landing on a comet), and the NetSat project (low Earth orbit formations). Based on the state-of-the-art, we outline how self-aware computing may improve autonomy in future space missions.

25.1 Introduction

In the last 60 years, exploration of our home planet's environment raised challenging technical tasks. While physics provided the basis for initial models, the specific parameters and relevant perturbations still had to be determined by experience with engineering such complex systems. Thus, today our Earth's environment is reasonably well known, while our knowledge about even the other bodies in our solar system is still very limited and therefore such bodies are target of challenging space exploration tasks which are for their part subject to specific requirements. In this

---

Klaus Schilling
University of Würzburg, Department of Computer Science, Am Hubland, D-97074 Würzburg, Germany e-mail: `schi@informatik.uni-wuerzburg.de`

Jürgen Walter
University of Würzburg, Department of Computer Science, Am Hubland, D-97074 Würzburg, Germany e-mail: `juergen.walter@uni-wuerzburg.de`

Samuel Kounev
University of Würzburg, Department of Computer Science, Am Hubland, D-97074 Würzburg, Germany e-mail: `samuel.kounev@uni-wuerzburg.de`

context, the on-board data handling systems have to act autonomously in order to adapt to unforeseen conditions. In this contribution potential future approaches to improve autonomy by adding self-awareness-capabilities are suggested. These capabilities are aimed at improving the spacecraft ability to continuously

- characterize its status by its sensors as well as by the space environment status (self-reflective),
- assess from the known dynamics, what changes will occur in the near future (self-predictive), and
- react appropriately in time such that the operations necessary for the planned mission are initiated (self-adaptive).

The above three —already partially realized— properties (i.e., self-reflective, self-predictive, and self-adaptive) are in line with our notion of self-aware computing, defined in Chapter 1, which stresses model *learning* and *reasoning* as ongoing activities enabling ”informed” actions in order to meet higher-level goals. Existing solutions in space missions are based on physical models and control theoretical solution approaches. This chapter summarizes them and discusses possible advances using self-aware methodologies.

The remainder of this chapter is organized as follows: Section 25.2 discusses the specific requirements in space, while Section 25.3 discusses self-awareness and other solution approaches. Section 25.4 depicts two classes of example missions. First, we emphasize two exemplary ESA missions to explore our solar system. These are HUYGENS (which landed 2005 on the largest Saturnian moon Titan) and ROSETTA (which accompanied 2015 the comet 67P/Churyumov-Gerasimenko during its closest approach to the sun, called perihel passage) [17]. Besides such traditional single multi-functional big spacecraft missions, there is a trend towards a distributed combination of multiple small spacecrafts. Therefore, secondly, Section 25.4 also addresses formations of cooperating satellites in low Earth orbits using the NetSat project as an example. Finally, Section 25.5 presents some concluding remarks.

25.2 Requirements in Space

Space system operations have to address challenges, such as higher levels of noise and huge distances causing significant signal propagation delays. Hence, unlike the often theoretical discussions of the benefits of autonomy for terrestrial robotic applications (e.g., in [4]), the autonomous reaction capabilities of space vehicles are needed to survive until the situation can be analyzed remotely by human teleoperators and ground control can intervene by appropriate reactions [16], [9]. A typical definition to characterize the required reaction capabilities for spacecraft is:

Definition 25.1. *Autonomy* defines the capability of a vehicle

- to meet mission performance requirements for a specified period of time without external support,
- to optimize the mission science products, e.g., the scientific measurements, within the given constraints.

Here in particular interplanetary space probes encounter specific challenges due to:

- extreme working environments (radiation, temperature, pressure, gravity)
- huge distances (leading to significant signal propagation delays, teleoperations autonomy needs, no human interaction capabilities in time critical situations)
- major uncertainties (limited sensors to characterize the spacecraft's environment, poorly modeled working environments, limited capabilities to verify and test)

Although there are well-known mathematical models of the physical environment, the crucial values for specific parameters are still to be determined on site. For example, the satellite dynamics are determined by gravity, where well-known generic mathematical models are available. Nevertheless, the mass distribution and inhomogenities of a specific target planet still needs to be determined and to be represented in the coefficients of the detailed power series expansion of the gravity field.

25.3 Solution Approaches

Compared to other contexts where autonomy is applied, space exploration missions have to survive in time critical situations and learned aspects can only improve the next mission. Every experiment or exploration is unique. All these aspects are different compared to continuous service provisioning (e.g., in data centers). In space exploration, advanced classical control approaches are often applied in order to handle related uncertainties. In this section, solution approaches are reviewed and the potential for future use of model-driven algorithms and architectures from self-aware computing is outlined.

25.3.1 Model-Based Adaptive Control

The adaptive control approach has been applied in the context of highly reactive systems. One of the greatest challenges for modern aerospace applications is the ability to react in real-time to changing environmental conditions and to adapt the related responses [3]. Thus, supersonic aircraft are often aerodynamically unstable and need continuously active control in order not to crash. Figure 25.1 depicts the basic adaptive control principle. While the design of a conventional feedback control

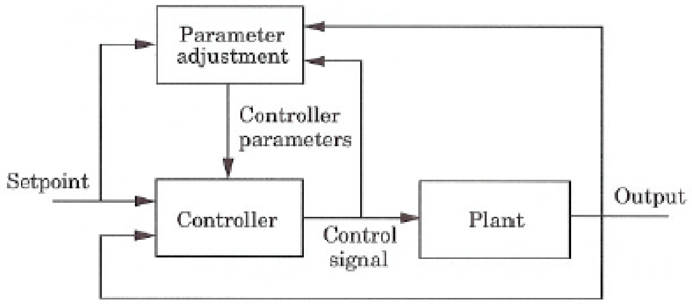

Fig. 25.1: Adaptive control principle.

system firstly targets the elimination of the effect of disturbances upon the controlled variables, the design of adaptive control systems firstly targets the elimination of the effect of parameter disturbances upon the performance of the control system. [8]

Adaptive control addresses the update of parameters in models, but not adaptations of the whole solution strategy, like in self-aware computing. A new aspect of self-aware computing, compared to model-based control, is the use of semantic models. These can be transformed via model-to-model transformations which enables an easy swapping of solution strategies.

25.3.2 Supervisory Control

In contrast to automatic control, supervisory control considers a human in the control loop and provides a framework to assign tasks. While real-time reactions should be realized autonomously, the human operator contributes to higher level superimposed control loops (like at the planning level) with less stringent time constraints. An integrated human-machine control system can be described as a set of embedded control loops working at different time scales, as illustrated in Fig. 25.2 [22], with high frequency feedback in the center and more long-term “learning” schemes at the outer loop. The `plan` step includes attaining awareness of the environment situation and the system to control, as well as the setting of achievable goals or related intermediate steps The `teach` step is about to decide control actions. Sensor and model-based supervision of the current state of the spacecraft are done within `monitor` step. The `intervene` box depicts human intervention to modify the control algorithm. The `learn` step is about to record experience and updating models. The teach, monitor, and intervene functions are done iteratively, and therefore are depicted within an inner and online loop [22]. The implementation challenges relate to avoiding conflicts between these nested control loops. While the real-time features are to be realized on-board the spacecraft, the planning levels are usually done by the tele-operators in the ground control centers.

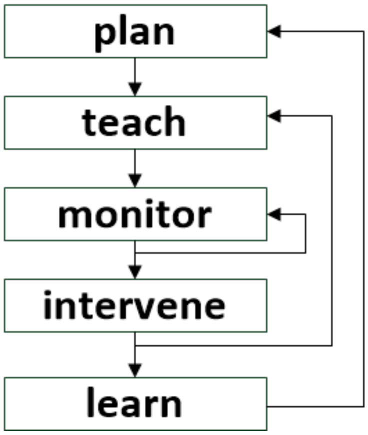

Fig. 25.2: The five generic supervisory control functions with nested related control loops.

We complete the description of supervisory control discussing the relation to self-aware computing. Both try to reduce the dependency on human intervention by integrating the human at the highest levels of abstraction, while maintaining lower-level functions within the machine. However, self-aware systems may interact with each other and do not require human intervention. Regarding the differences, supervisory control sets the focus on interaction of human operator and machine. Instead, self-aware computing sets the focus on how the programs work internally (e.g., use of model-based approaches differing descriptive, prescriptive, and predictive models) and includes a concrete model-based solution strategy. Due to the different focus, we see a broad area of space applications were both paradigms form a good complement to each other.

25.3.3 Distributed Networked Control

Distributed networked control addresses the control via communication links. In space applications, satellites form nodes that are connected and coordinated throughout a network. Linking technologies include e.g., dedicated space protocols, Internet Protocols (IP), delay tolerant networks (DTNs) and mobile ad-hoc networks (MANets) adapted to the space environment (e.g. high noise levels, link interruptions, interferences, radiation, inaccuracies in pointing). The research on Distributed Networked Control [7] has a multidisciplinary nature, blending the areas of communication networks, computer science and control. The properties of the telecommunication links together with the control characteristics are to be combined to an integrated system [5].

One application scenario that attained much attention in recent times is the provision of Internet via satellites (current projects are discussed in Section 25.4.3). For this purpose existing approaches have to be extended and new strategies have to be found. In particular, the case of coordinating several spacecraft in a formation is challenging. In addition to the interaction between ground control and satellite, also the exchange of information between vehicles regarding their status and plans for future actions via inter-satellite links is to be analyzed. The challenge consists in a reliable coordination of the distributed computers and decision making resources in order to achieve the mission objectives in a consistent and robust way. Similar to the evolution in computing, where the traditional mainframe computers of the 70ies have been replaced by the Internet connected laptops or smart phones, also in spacecraft system design the established multi-functional large spacecraft are expected to become in specific application areas complemented or substituted by networked small satellites [14].

In contrast to distributed networked control, self-aware computing describes a solution strategy using descriptive models that is independent of the problem domain. Due to the profile of the addressed problems in distributed networked control, there is interesting application potential for future self-aware computing approaches. Self-aware computing can be applied to provide distributed networked control.

25.3.4 System Health Management

Reliable spacecraft operations require system health management. Extreme radiation environments dramatically increase failure risks for all electric components. Radiation may cause a change of an electronic state due to one single ionizing particle (ions, electrons, photons...) striking a sensitive node in a micro-electronic device, such as in a microprocessor, semiconductor memory, or power transistors. The state change is a result of the free charge created by ionization in or close to an important node of a logic element (e.g. memory ”bit”).

Due to the high likelihood of errors, the system health has to be continuously monitored. In classical space engineering approaches health management handles redundancy switching. Ideally, system health management should detect, resolve and predict failures. NASA researchers promote statistical approaches of health management with Bayesian networks [20] This solution approach is popular in the academic context but has not been applied in space missions so far. A major drawback are the computationally intensive reasoning algorithms [21]. Therefore, the use of special hardware, Field Programmable Gate Arrays (FPGAs), has been proposed in [21]. Bayesian networks are not the only solution to ensure system health. Furthermore, data mining techniques can be used for detection, diagnostics, and prognostics [23]. Advanced approaches to health monitoring often rely on model based fault detection, isolation and recovery (FDIR) methods. This way deviations from expected status are detected and corrected. Most often this is realized by taking advantage of redundant systems.

Compared to self-aware computing, system health management describes the problem domain without specifying a concrete solution strategy beyond redundancy switching and disconnecting faulty components. Self-aware computing, instead, describes a solution strategy, using descriptive models, that is independent of the problem domain. According to Schuhmann et al. [21], the trend should go towards real-time, on-board, sensor and software health management. Our proposed realization is that aerial systems get self-aware concerning their health status. For future applications, the concept of self-aware computing can help to incorporate different solution strategies like Bayesian networks or data mining approaches into one combined view on system health management. This would enable a self-aware change of the system health insurance strategy that considers for example a trade off between cost and accuracy of approaches.

25.3.5 Self-Aware Computing

Compared to the previously presented approaches, the idea of self-aware computing, applied in the context of space applications, translates into combining model-based learning and reasoning as on-going processes built into the spacecraft design to support autonomous reaction and control mechanisms. Space missions are becoming more and more complex and challenging. We see a need for further automation to reach new goals. We argue that inspiration from self-aware computing can help to advance the field. Despite sharing crucial aspects with classical adaptive control, self-aware computing introduces complementary new aspects:

- in addition to collecting observations and monitoring data during operation, self-aware computing emphasizes the learning of formal models capturing knowledge in an abstract and compact manner and supporting reasoning with respect to the system goals,
- model learning processes are first-class entities in the system design that drive the spacecraft decisions; they integrate knowledge provided by the system designer with observations obtained during operation,
- the learned models support complex reasoning and predictive analytics that go beyond applying simple rules or heuristics explicitly programmed at system design-time,
- both the learning and reasoning processes are assumed to be running on an ongoing basis during operation; thus, models are expected to evolve as time progresses leading to improved reasoning and more reliable decisions
- self-aware computing leverages models of different types in an integrated manner: i) *descriptive models* describe selected aspects of the system and its environment in an abstract manner enabling formal analysis and reasoning, ii) *prescriptive models* typically define behaviors to be applied in different situations, e.g., adaptation processes, iii) *predictive models* support more complex reasoning, e.g., predicting the system behavior under given conditions or predicting the impact of a considered possible adaptation action,

- by leveraging model-to-model transformations, flexibility in trading-off between model accuracy and analysis overhead is provided. A suitable model combined with a tailored solution strategy can be selected depending on the specific reasoning scenario (urgency of the situation, criticality of the decision to be made, required accuracy, etc.)

25.4 Example Missions and Projects

The research on autonomy in space is driven by uncertainties of the space environment, where reliable reactions are required and will be evaluated in reality. In the following, we will describe several projects that faced challenging autonomous operation tasks. These are the landing on the Saturnian moon presented in Section 25.4.1, the accompanying and landing on a comet presented in Section 25.4.2 and the satellite formations in low Earth orbits presented in Section 25.4.3.

25.4.1 HUYGENS – Landing on the Saturnian Moon Titan

While NASA's VOYAGER 2 spacecraft approached, in November 1980, the largest Saturnian moon Titan at a close distance of 5000 km, the instruments could not penetrate the unexpectedly dense atmosphere. Nevertheless, during this flyby Hydrocarbon molecules were detected and justified a return for more detailed investigations of the exotic organic chemistry in this atmosphere. In the resulting joint mission, NASA contributed the CASSINI spacecraft for long-term remote sensing observations by orbiting the Saturnian system, while the European Space Agency (ESA) contributed the HUYGENS probe to descend to the surface of Titan. The CASSINI-HUYGENS-mission was launched on the 6th of October 1997 and arrived at the Saturnian system in June 2004 [9, 13]. The significant distance led to a signal propagation delay of 68 minutes. Therefore, ground control interaction during the entry and descent, lasting 2.5 hours, was not feasible, and autonomous adaptation and decision making on-board was unavoidable [10]. In the following, we will focus on the control approaches to pass through the poorly known Titan atmosphere by entry and parachute descent manoeuvres in order to safely land on Titan's surface. The descent, illustrated in Fig. 25.3, had to meet the following requirements:

- minimum period for measurements in the different atmospheric layers
- coordination of instrument activities for efficient energy consumption
- monitored landing on the surface
- limited descent duration caused by the transmission geometry towards the CASSINI-spacecraft, which acts as data relay towards Earth during its fly-by

A desirable control approach in that context would react adaptively to the incoming information from instrument measurements in order to update the atmospheric

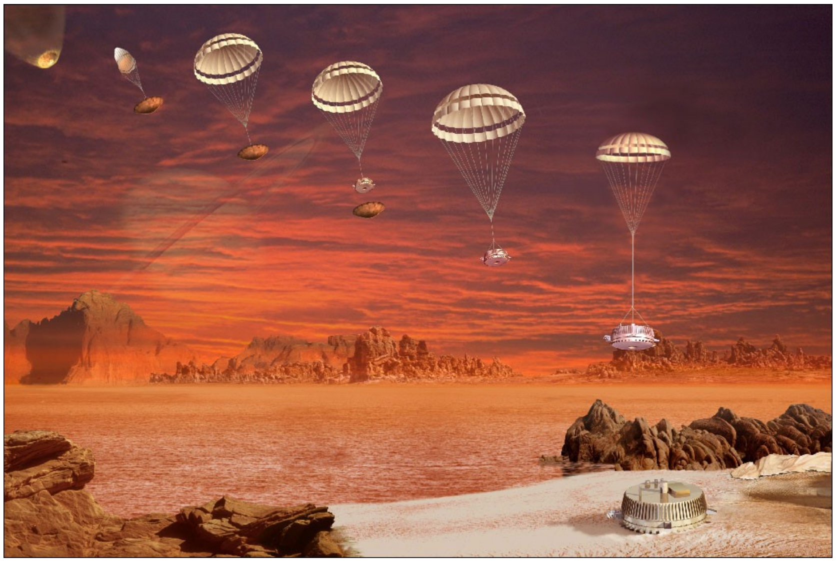

Fig. 25.3: The HUYGENS entry and descent scenario for exploration of Titans atmosphere and surface (image courtesy of ESA). Timing, speed adjustment, and the establishing and retaining of the communication link posed many challenges.

models and the predicted descent models. However, the means to realize such a control approach were rather limited and related to the timing of

- parachute deployment after significant deceleration in the atmospheric entry phase,
- separation from decelerator heat shield for mass reduction, thus increasing the descent duration,
- change from the 8m diameter first parachute towards the smaller 3m parachute, thus accelerating the descent duration.

The timing of the parachute deployment had to select the right moment:

- not too early, as otherwise the high velocity would cause significant drag forces just destroying the parachute
- not to late, as otherwise at lower velocities the atmospheric particles would not be able to inflate the parachute.

Due to the high thermal flux at the time of parachute deployment (temperatures about 1000 °C at the outside of the heat shield), only inertial acceleration sensors could be used to determine the critical velocity to open the parachute. Another challenge in the Titan atmosphere was to cope with an uncertain atmospheric density model. The simulations results depicted in Fig. 25.4 exhibit the different potential atmospheric profiles for the velocity / acceleration evolution. Fortunately, the graphs of the atmospheric profiles `nominal`, `minimal` and `maximal` converge at the planned deployment velocity of about Mach 1.5 (about 400 m/s). Thus acceleration profiles could be used as timing criterion for the main parachutes deployment despite uncertainties in atmospheric density profiles.

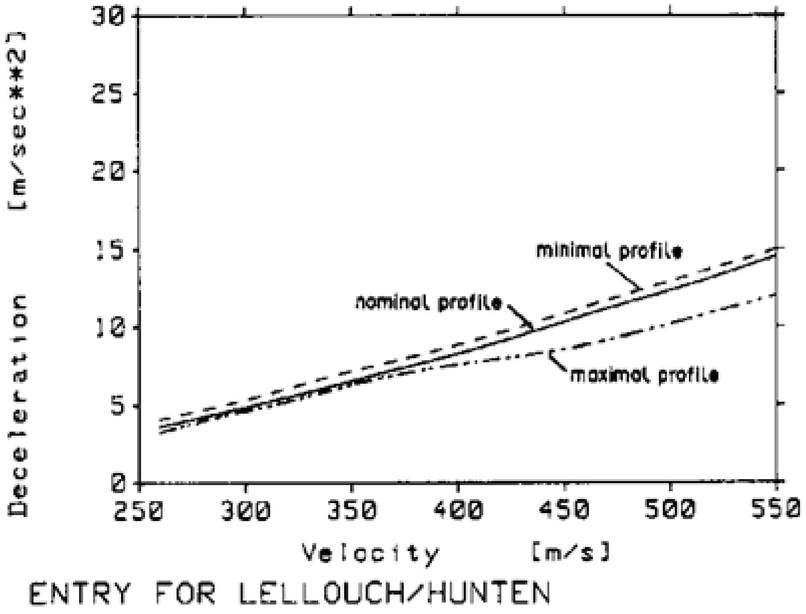

Fig. 25.4: The deceleration as a function of the velocity for different atmospheric models (according to Lellouch-Hunten). The three profiles converge at a velocity of about 400 m/s which is the speed the parachute opening was triggered.

During and after the decent, communication had to be ensured. Therefore, HUYGENS dropped an additional spacecraft named CASSINI to act as a relay link for transferring the measurement data to Earth. The descent of HUYGENS had to be coordinated with CASSINI to no lose connection. HUYGENS had to respect the time constraints related to the flyby geometry of the CASSINI spacecraft. In case of delays on decent, CASSINI could have passed the horizon to keep the connection established. As a result, the interesting on-surface measurement could not have been transfered back to Earth when HUYGENS finally landed on the surface.

The decent was adaptively controlled, as depicted in 25.5. The height profile $h(t)$, including in particular the time for the surface impact, is predicted from the equations of motion depending on the gravity forces F_G of Titan (well known since the flyby of VOYAGER 2) and the drag forces F_D (poorly known as the VOYAGER 2 instruments could only measure upper atmospheric layers). So in the beginning $h(t)$ will only be a rough estimate, but after parachute deployment, the atmospheric density ρ will be measured in addition to the deceleration a. The parameters inserted into the model of drag forces F_D are very poor during the initial period of the mis-

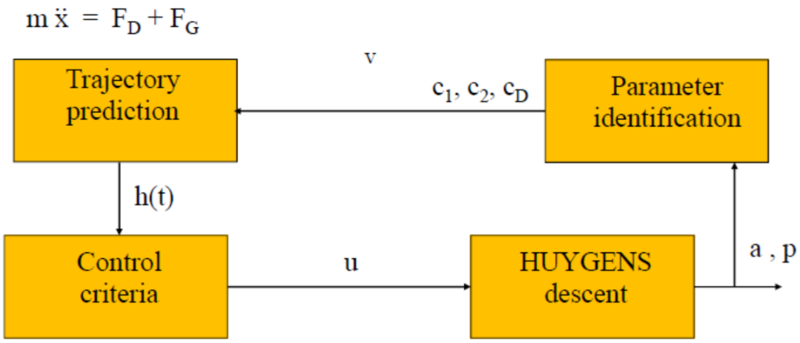

Fig. 25.5: Adaptive descent control scheme for HUYGENS landing on Titan. It includes many sources of uncertainty at the parameter identification that increase confidence intervals for decent time prediction. The parameters to be identified are the atmospheric desisity profile which introduced an uncertainty of ± 7.5 min and the HUYGENS drag coefficient which increased the confidence interval by ± 6.3 min. Finally, the uncertainty concerning the Titan surface topography additionally added ± 7.2 min.

sion. However, over the mission progress, more measurements become available, which leads to continuous and significant improvements. On this basis, descent profile control has continuously been optimized during the descent and landing.

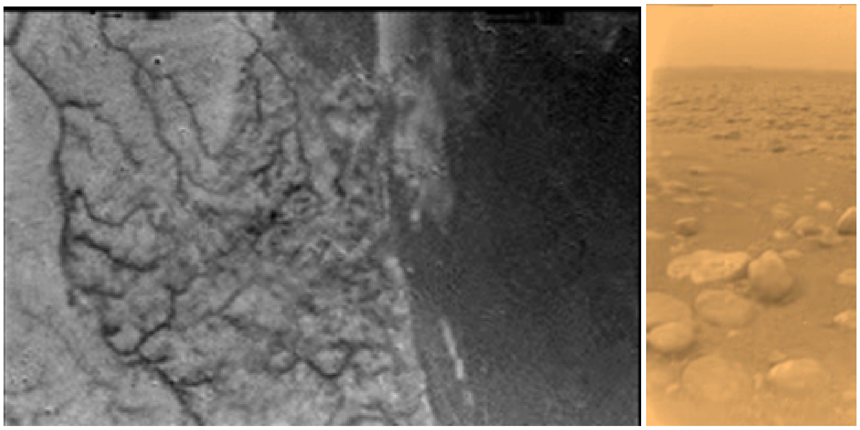

Fig. 25.6: Images from Titan: The picture on the left shows a swampy area with river features (dark streams) and the colorized image on the rihgt shows the landing site in the dry bed of the lake (both images are original mission pictures and courtesy of ESA).

On January 14th 2005, HUYGENS landed successfully on Titan with a deviation of less than 5 min from the pre-planned values. Very impressive images and material composition data were acquired and transferred to Earth. More information on the mission can be found at http://sci.esa.int/cassini-huygens/http://sci.esa.fnt/cassini-huygens/.

Until now, we described the HUYGENS mission in retrospect. Questions targeting prospective missions are “How could self-awareness be applied?”, and “What are the benefits?”. The HUYGENS mission was choreographed for a certain scenario, realized using a hard wiring of solution strategies. In contrast, an increased flexibility would be desirable for future missions. This could be achieved if future system creators are inspired by the self-awareness idea. Further, exchange and reuse of solution strategies would increase. The techniques applied in the CASSINI/HUYGENS mission enable to setup self-awareness. The mission design employed by example discrete models for payloads operations as well as thruster (trust on or off) and continuous models for physical phenomena. The data learned during operation for model parameterization was about position, temperature, and pressure. Complemented by a few rule based approaches, the reasoning was dominated by measurement, control, and regulation technology. Besides HUYGENS specific aspects, typical mission constraints have to anticipate limited resources related to energy availability and consumption, but also to fulfill objectives (e.g. to instrument pointing by attitude control). Mission objectives are to be compromised with resources available in time critical situations. This is one example where self-aware computing incorporates easily: Predictive models support decision making, leading to a self-awareness concerning energy availability and consumption costs.

25.4.2 Rosetta – Accompanying and Landing on a Comet

The ESA-mission ROSETTA had the objective of a detailed comet exploration. In 2014 a rendezvous manoeuvre injected the spacecraft into orbit around comet 67P/Churyumov-Gerasimenko, which enabled long-term observation during the evolution of the comets tail during perihel passage on August 13th, 2015. On November 12th, 2014 the PHILAE lander probe was deployed to the surface. Accidentally but fortunately, all devices attaching it to the surface (one cold gas thruster, two harpoons, and three ice screws) failed. Thus, it finally settled after 3 times bouncing in a scientifically more interesting riff. In the following, the intended adaptive drilling in the poorly characterized soil will be described [12], [11]. Due to the unestablished anchoring, drilling could not be applied, nevertheless the challenge of dealing with uncertain environments is of generic interest also for future missions. The distance to Earth during the landing was about 500 million km leading to a latency of 28 minutes. Thus, the control commands arrive at the earliest after 56 minutes after the input measurements occurred. So for almost one hour all situations in this very uncertain work environment had to be handled autonomously by the on-board data processing system before any reactions from ground control could arrive. The core problem was the coordination of the drilling device, the flight attitude control system, and the anchoring system for a safe and energy efficient sample acquisition. For this purpose, the targeted optimization goals i)maximization of spacecraft attitude stability and ii)minimization of drilling duration provided the design requirements. For mission design simulations, the following model components had to be taken into account:

- drilling equipment
- cold gas thruster
- mechanical soil properties of the comet surface
- anchoring by harpoons
- force and torque transfer in the structure of the probe
- force sensors, gyros, energy consumption monitoring

The soil parameters to be identified during the drilling process included: i)Young's modulus and ii) adhesive friction. These were expected to vary according to drilling depth. Thus while drilling progresses, the related parameters are identified for the specific depth level based on encountered forces.

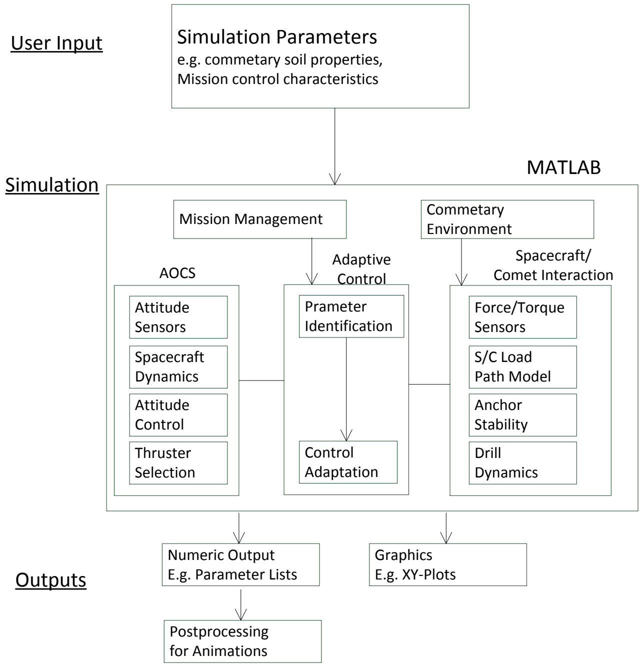

Fig. 25.7: Structure of the simulation tool for core drilling for the ROSETTA mission.

Fig. 25.7 depicts the simulation tool for predicting the expected performance properties. As self-aware computing was not introduced, the simulations were important to manually tailor the controller for the Rosetta mission. Fig. 25.8 summarizes the adaptive controller design to anticipate all potential situations for robust performance of drilling, considering multiple inherent uncertainties. Thus, the full control chain, from sensor modeling, the data acquisition and control reactions, to the impact of the actuators on the spacecraft, are anticipated. The on-comet operations were considered high risk already during the initial mission planning phases in the early 1990s. Malfunctions of single components and the appropriate reactions had been included in the control strategy of the spacecraft. For example, the drilling assisted by the thrusters was only planned in case harpoons and screws fail. Detailed variations of unknown soil parameters were assessed in simulations of typical mission scenarios in order to generate the most promising control strategy.

The obtained simulation results predicted significant performance improvements compared to an adaptive control scheme. The coordination of all capabilities onboard the spacecraft enabled a reduction of maximum forces and torques on the anchors of about 30% (cf. Fig. 25.9). Despite the simulation strategy provided appropriate results, more flexible reactions, possibly enabled by self-aware approaches, would be desirable.

Surface science activities — except the drilling — had been performed by PHILAE with the energy provided from the batteries charged before launch and the acquired data had been relied by ROSETTA to ground control. ROSETTA continues the journey as companion of the comet and just passed on the 13th of August 2015 the closest approach towards Sun with significant increase of material sublimation activities. Further details can be found at http://sci.esa.int/rosetta/.

The encountered surface operations well exceeded earlier anticipated situations, as all devices to attach the spacecraft to the cometary surface failed. Nevertheless PHILAE was somehow fixed in gap, but modeling and assessment of this unforeseen situation was not possible. Self-aware computing offers methodology to better cope with unforeseen situations. However, it is hard evaluate whether additional operations could have been performed and measurements could In general, ROSETTA and HUYGENS had to face many challenges (sumarized in table 25.1) where one could ask whether the self-aware idea would have improved mission outcome.

25.4.3 NetSat - A Satellite Formation in Low Earth Orbit

In Earth orbits, several multi-satellite systems have been established for applications in communications (e.g., IRIDIUM, Globalstar, TDRSS, Orbcomm), in navigation (e.g., GPS, Glonass, Galileo, BeiDou), in Earth observation (e.g., Rapid Eye, Dove), and in science (e.g., Cluster, Swarm). All these systems are realized as satellite *constellations*, where each satellite is individually controlled from the ground. Future —more advanced multi-satellite— systems are expected to be formations based on relative distances between spacecrafts. Appropriate topologies will be maintained

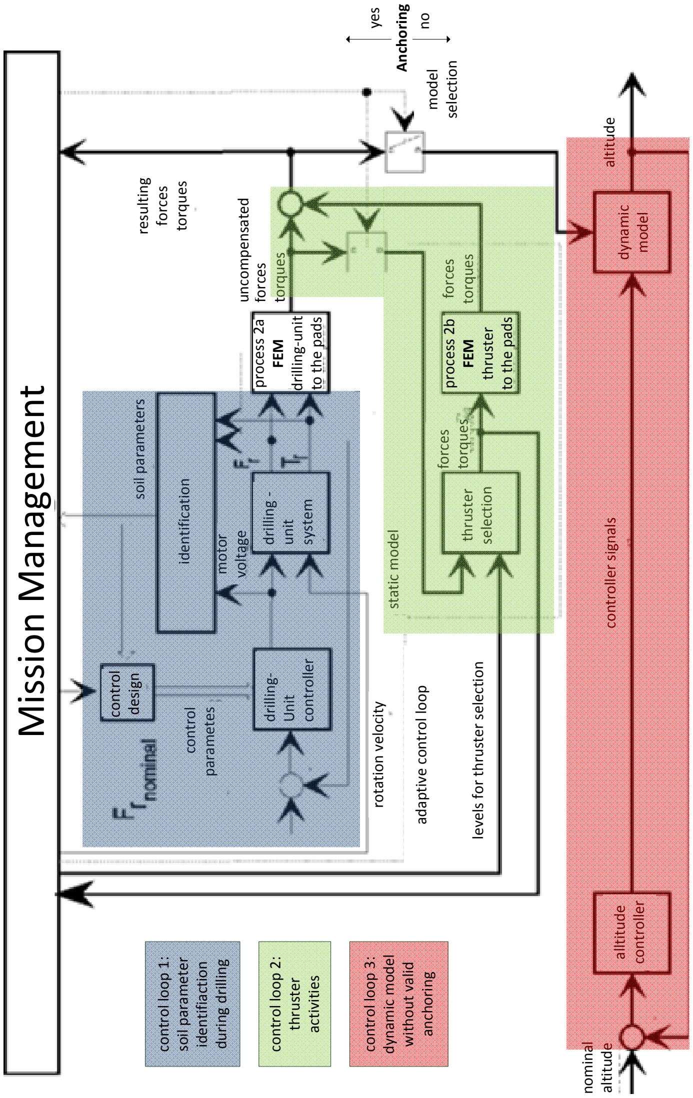

Fig. 25.8: Block diagram of the adaptive control strategy including three main control loops. These are focused on the adaptation to soil consistency (**control loop 1**), the controlling of thrusters **control loop 2**, and the altitude control (**control loop 3**).

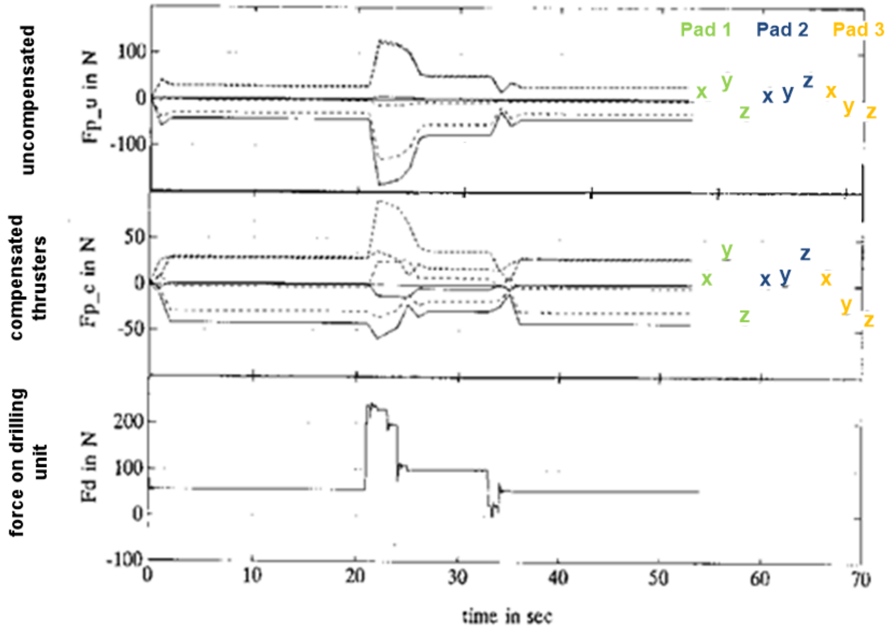

Fig. 25.9: The first two rows show the forces acting at the pads during the drilling process at a typical soil layer profile: F_p u = uncompensated, F_p c=compensated by cold gas thrusting. The graphs depict the force per direction (**x, y, z**) for each of the three pads (**pad1**, **pad2**, **pad3**). It can be seen that the thrusters reduced the forces on the pads. Beneath, in the third row, the reaction to the drill pushing force F_d is depicted.

Table 25.1: Summary of challenges for autonomous reactions in the mission scenario for the interplanetary missions HUYGENS and ROSETTA

Mission	Huygens	Rosetta	
Phase	atmospheric descent	descent and landing on the comet	soil sampling
Objectives	to achieve atmospheric descent profile suitable for scientific measurements	safe landing near specified location	safe acquisition of subsurface samples
Main environmental uncertainty	atmospheric density profile	dynamic/kinematic properties, topography of landing site	mechanical soil properties for anchoring and drilling
Controlled states	descent profile	spacecraft attitude, orbit parameters	flight attitude, drill pushing force, rotation rate
Available actuators	timing of main parachute deployment	hydrazine/cold gas thrusters	drill motors, cold gas thrusters

using data exchange between the self-organizing spacecrafts This is necessary, as

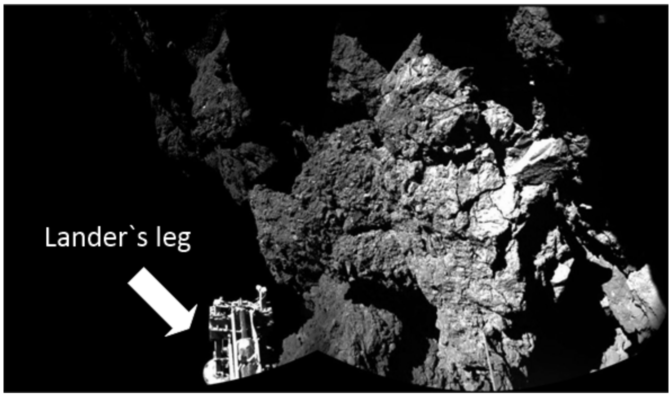

Fig. 25.10: PHILAE finally landed on the comet: in front the Lander's leg is visible in front of the surprisingly hard cometary surface rocks (image courtesy of ESA, picture).

for low Earth orbits (LEO) one ground station has less than 10 % of time access to the satellite. For several orbits no contact at all occurs. In particular currently planned mega-constellations in low earth orbits are foreseen to provide a worldwide infrastructure for Internet access (e.g., OneWeb, SpaceX). These constellations will require more advanced methods for efficient operations. Beyond these telecommunication applications, also in Earth observation and Space Weather characterization, commercial multi-satellite missions (e.g. by Planet Labs, Spire, PlanetIQ) have been recently placed in orbit and are expected to be further expanded [6]. These distributed networked multi-satellite systems provide data with high temporal and spatial resolution, and thus enable innovative environment monitoring. Huge additional numbers of satellite in such LEO orbits, significantly increase the risks for collisions, especially in the regions near the poles. As satellite densities in the polar regions are expected to significantly increase due to orbit dynamics properties, range detection and collision avoidance might become requirements for the future in order to avoid significant increase of space debris. There are significant similarities to networked automobiles as well as to networked industrial production methods. In this context, *formations* of multi-satellite systems have to be self-organizing, in order to provide appropriate position and orientation of the satellites for observations or for communication links. This requires an inter-satellite communication link to close the control loop in orbit. Cooperation and exchange of information will be based on relative distance and attitude measurements, as well as on telecommunication links [1, 2, 15].

At *U*niversity *W*ürzburg's *E*xperimental satellite (UWE) program a longterm roadmap was established to realize the relevant technologies for formations at pico-satellite level (at a mass of just a few kilograms). The first German pico-satellite UWE-1 (launched 2015 by a COSMOS-3M) addressed the scientific aspects of "internet in space" as basis for the inter-satellite network in orbit. While UWE-2

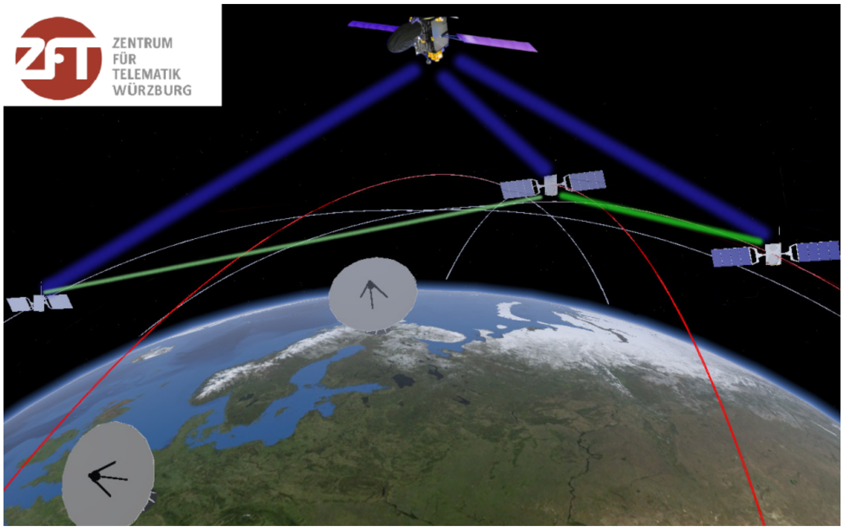

Fig. 25.11: NetSat: Networks of a ground stations and satellites formations promise potential for applications in future Earth observation and telecommunication services.

(launched 2009 by PSLV) had emphasis on attitude determination, the UWE-3 mission (launched 2013 by Dnepr) continued with attitude control. This technology base will be complemented by the currently prepared UWE-4 to demonstrate orbit control capabilities. On this basis as next step the NetSat mission employing 4 satellites is implemented to analyze the control of 3-dimensional topologies in orbit (planned launch in 2018). This will enable innovative photogrammetric Earth observation approaches. Scientific challenges of NetSat address model-based orbit predictions and autonomous adaptive corrections of deviations with respect to telecommunication, data processing and control [18, 19]. The objective of NetSat is the realization of a distributed, cooperating multi-satellite systems using autonomous formation control for optimization of observation periods. Relative distances at begin of mission will be between 50 - 10 km for safe operations. The first subgoal is to autonomously maintain the formation configuration. After having acquired sufficient experiences and derived appropriate models more risky, near proximity formations at distances between 20 - 40 meters are foreseen. Technology challenges to be addressed by NetSat include:

- formation control
    - model reference based adaptive control for attitude and orbit control by reaction wheels, magnetic torquers, electric propulsion
    - relative attitude and position determination within the formation, based on data exchange and data fusion

- autonomous, networked satellite control
  - Reliable data exchange between the satellites by mobile DTNs and ad-hoc networks to adapt to changing communication topologies and interruptions
  - Networked control of the satellite formation, combination of supervisory control from ground with autonomous reactions
- small satellite in-orbit demonstration
  - Implementation of a demonstrator mission based on 4 pico-satellites
  - navigation sensor system, in particular for relative distances & orientations

While the first phase of NetSat applies adaptive and supervisory control approaches in a conservative way, the NetSat mission offers the capabilities to upload new operational software. Thus it could serve as a testbed for self-aware approaches the next phase. Candidate application fields might address health management at component and subsystem level. Other interesting features are collision avoidance and minimum fuel consumption detour manoeuvres in a cluttered environment with space debris. With respect to planning also suitable strategies for de-orbiting at the end of mission lifetime can be considered. Thus direct comparisons of self-aware performance advantages with traditional techniques in orbit could be implemented.

25.5 Conclusions

Space missions raise challenging tasks for providing autonomous reaction capabilities in the context of interactions with poorly known environments being the target of explorations. Earth-based control for time-critical situations is often impossible due to significant signal propagation delay and link occultation periods. Additionally, we expect that more logic will move from the supervisor into the spacecraft. Space missions are becoming more and more complex and challenging. We see a need for further automation to reach new goals. We argue that inspiration from self-aware computing can help to advance the field.

In this chapter, we presented general solution approaches that have been applied for autonomy in space and explain their relation to self-aware computing. Further, we discussed the potential of self-aware computing for the application in space exemplified by two interplanetary missions HUYGENS and ROSETTA, and the satellite constellations of the NetSat project. Candidate application fields of self-aware computing might address health management, collision avoidance, minimum fuel consumption at detour manoeuvres, and de-orbiting at the end of mission lifetime. Compared to the previously applied approaches, the idea of self-aware computing translates into combining model-based learning and reasoning as on-going processes built into the spacecraft design to support autonomous reaction and control mechanisms. Despite sharing crucial aspects with classical adaptive control, self-aware computing introduces complementary new aspects. At self-aware computing, model learning processes are first-class entities. Formal models capture knowledge in an abstract and compact manner and support reasoning with respect to the system goals. Both the learning and reasoning processes are assumed to be running on an ongoing basis during operation; thus, models are expected to evolve as time progresses leading to improved reasoning and more reliable decisions The learned models support complex reasoning and predictive analytics that go beyond applying simple rules or heuristics explicitly programmed at system design-time. Self-aware computing leverages *descriptive*, *prescriptive*, and *predictive* in an integrated manner. Model-to-model transformations enable flexibility in trading-off between model accuracy and analysis overhead. A suitable model combined with a tailored solution strategy can be selected depending on the specific reasoning scenario. To move forward into unknown areas, these ideas of self-aware computing may support advanced autonomous behavior of spacecrafts and ground control in the future.

References

1. Kyle Alfriend, Srinivas Rao Vadali, Pini Gurfil, Jonathan How, and Louis Breger. *Spacecraft Formation Flying: Dynamics, Control and Navigation*. Elsevier Astrodynamics Series. Elsevier, 2010.
2. F. Ankersen, editor. *3rd International Symposium on Formation Flying, Missions and Technologies*. ESA SP-654, 2008.
3. Karl Johan Astrom and Bjorn Wittenmark. *Adaptive Control*. Prentice Hall, 1994. [3]Astrom, K., B. Wittenmark, Adaptive Control, Addison-Wesley, 1989, 2d ed. 1994.
4. George A Bekey. *Autonomous robots: from biological inspiration to implementation and control*. MIT press, 2005.
5. Francesco Bullo, Jorge Cortés, and Sonia Martinez. *Distributed Control of Robotic Networks: A Mathematical Approach to Motion Coordination Algorithms: A Mathematical Approach to Motion Coordination Algorithms*. Princeton University Press, 2009.
6. Marco D'Errico. *Distributed Space Missions for Earth System Monitoring*, volume 31. Springer Science & Business Media, 2012.
7. Xiaohua Ge, Fuwen Yang, and Qing-Long Han. Distributed networked control systems: A brief overview. *Information Sciences*, pages –, 2015.
8. Ioan Doré Landau, Rogelio Lozano, Mohammed M'Saad, and Alireza Karimi. *Adaptive control*, volume 51. Springer Berlin, 1998.
9. K Schilling. Control aspects of interplanetary spacecrafts: An introduction to the CASSINI/HUYGENS mission. *Control Engineering Practice*, 3(11):1599–1601, 1995.
10. K Schilling and W Flury. Autonomy and on-board mission management aspects for the CASSINI Titan probe. *Acta Astronautica*, 21(1):55–68, 1990.
11. K Schilling, H Roth, and B Theobold. Autonomous on-comet operations aspects of the RoSETTA mission. *Control Engineering Practice*, 2(3):499–507, 1994.
12. Klaus Schilling. Simulation of ROSETTA on-comet operations. In *Annales Geophysicae*, volume 10, pages 141–144, 1992.
13. Klaus Schilling. The HUYGENS Mission to Explore the Saturnian Moon Titan. 2005.
14. Klaus Schilling. Networked control of cooperating distributed pico-satellites. In *19th International Federation of Automatic Control World Congress*, pages 7960–7964, 2014.
15. Klaus Schilling. Kleinstsatelliten - winzlinge im orbit. *Spektrum der Wissenschaft*, pages 48–51, May 2015.
16. Klaus Schilling, J De Lafontaine, and Hubert Roth. Autonomy capabilities of European deep space probes. *Autonomous Robots*, 3(1):19–30, 1996.
17. KLAUS Schilling and MICHAEL Eiden. Huygens und Rosetta. Die geplanten planetaren Missionen der Europäer. *Sterne und Weltraum*, 29:428–437, 1990.

18. Schmidt M. Schilling, K. Communication in distributed satellite systems. In *DErrico, M. (ed.), Distributed Missions for Earth System Monitoring*. Springer, 2012.
19. Schmidt M. Busch S. Schilling, K. Crucial technologies for distributed systems of pico-satellites. In . IAC-12-D1.2.4, editor, *Proceedings 63rd International Astronautical Congress*, Naples, Italy, 2012.
20. J. Schumann, T. Mbaya, O. J. Mengshoel, K. Pipatsrisawat, A. Srivastava, A. Choi, and A. Darwiche. Software health management with bayesian networks. *Innov. Syst. Softw. Eng.*, 9(4):271–292, December 2013.
21. Johann Schumann, Kristin Y. Rozier, Thomas Reinbacher, Ole J. Mengshoel, Timmy Mbaya, and Corey Ippolito. Towards real-time, on-board, hardware-supported sensor and software health management for unmanned aerial systems. *International Journal of Prognostics and Health Management (IJPHM)*, 6(1):1–27, June 2015.
22. Thomas B Sheridan. *Telerobotics, automation, and human supervisory control*. MIT press, 1992.
23. A. Srivastava and J. Han, editors. *Data Mining in Systems Health Management: Detection, Diagnostics, and Prognostics*. Chapman and Hall/CRC Press, 2011.