TECHNISCHE UNIVERSITÄT MÜNCHEN
Lehrstuhl für Raumfahrttechnik

Interactive Acquisition Scheduling for Low Earth Orbiting Satellites

Robert Axmann

Vollständiger Abdruck der von der Fakultät für Maschinenwesen der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktor-Ingenieurs (Dr.-Ing.)

genehmigten Dissertation.

Vorsitzender: Univ.-Prof. Dr.-Ing. F. Holzapfel

Prüfer der Disseration:
1. Univ.-Prof. Dr. rer. nat. U. Walter
2. apl. Prof. Dr.-Ing. R. Schmucker

Die Dissertation wurde am 19.11.2009 bei der Technischen Universität München eingereicht und durch die Fakultät für Maschinenwesen am 04.05.2010 angenommen.

Danksagung

Diese Arbeit entstand während meiner Tätigkeit im Raumfahrtkontrollzentrum des Deutschen Zentrums für Luft- & Raumfahrt parallel zur Arbeit an verschiedenen Satellitenmissionen und Studien.

Mein besonderer Dank gilt meinem Doktorvater Prof. Dr. Ulrich Walter für die intensive Diskussion und kritische Begleitung während der Erstellung. Er hat mir mit vielen Hinweisen und Ratschlägen geholfen und war stets ansprechbar.

Weiterhin danke ich Dr. Martin Wickler für die Möglichkeit, die Arbeit im DLR anzufertigen, die stete Unterstützung und hilfreiche Diskussion.

Viele meiner Kollegen haben mich mit ihrer Erfahrung unterstützt. Auch dafür sage ich Dank.

Nicht zuletzt danke ich meiner Freundin, meiner Familie und meinen Freunden.

Per aspera ad astra

Contents

List of Figures

List of Tables

Acronyms

AAR	Area Access Rate
C&DH	Command and Data Handling
CCSDS	Consultative Committee for Space Data Systems
D/L	Down-Link
DT	DataTake
DWD	Deutscher Wetter Dienst
ECMWF	European Centre for Medium-Range Weather Forecasts
ECSS	European Cooperation on Space Standardization
EDF	Event Definition File
EnMAP	Environmental Mapping and Analysis Program
EO	Earth Observation
FOV	Field of View
GOES	Geostationary Operational Environmental Satellite
GRIB	GRIdded Binary
GS	Ground Station
GSOC	German Space Operations Center
LEOP	Launch and Early Orbit Phase
MDF	Mission Definition File
MPS	Mission Planning System
NOAA	National Oceanic and Atmospheric Administration
OS	Operating System
RTOS	Real-Time Operating System
SBC	Spacecraft Board Computer
SEPPL	SElf Pointing Processor Library
SSO	Sun-Synchronous Orbit

STK	Satellite Tool Kit
SYNOP	SYNOPtic
TC	TeleCommand
TLE	Two Line Element
TM	TeleMetry
TT&C	Telemetry Tracking and Command
U/L	Up-Link
WMO	World Meteorological Organisation

Symbols, Units and Indices

Symbols & Units

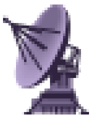
α	central angle
a	semi-major axis
A	area
b	arc length
cr	compression ratio
d	distance
e	great circle distance between two points (radians)
f	function (of something)
G	gravitational constant
λ	longitude
μ	gravitational parameter
$m_\oplus$	mass of the Earth
M	capacity of mass memory
n	mean motion
p	cloud free fraction
P	power
π	Pi
ϕ	latitude
r	satellite distance from mass center of the Earth
$R_\oplus$	radius of the Earth (6378137 m)
R	datarate
Σ	sum
t	time
T	temperature
v	velocity
x	sensor raw data
y	reconstructed sensor data

Indices

c	compressed
cc	cloud covered
$circ$	circular (orbit)
DL	down-link
MM	mass memory
n	point index
OU	orbit usage
P	payload
S	sensor
UL	up-link

Graphical Symbols

Control Center



Ground Station (providing communication between control center and satellite)

Satellite

Abstract

Scheduling of earth observation (EO) satellite sensor data acquisitions is usually executed in a ground control center as a part of the daily the mission operations and mission planning. The resulting sequence of telecommands is send to the satellite and executed subsequently. The satellite is without contact to the ground control center during execution of scheduled activities due to the EO mission design (orbit altitude, location of ground station).

If scheduling on ground does not know all scheduling relevant parameters it is advantageous to move this process on-board the satellite. Certain application areas have been identified for this. One important area is scheduling for optical sensors affected by cloud coverage in images. The other is scheduling of data acquisitions if loss-less data compression is used. For the on-board scheduling under uncertainty of loss-less data compression a patent has been submitted.

The different configurations and operations relevant aspects of an on-board scheduling system are discussed in the subsequent chapter. Scheduling operations relevant equations are derived that can be used for estimation of on-board scheduling reasonability. A greedy scheduling algorithm is selected for simulation. With this algorithm on-board scheduling under cloud coverage is simulated as a case study. The example mission parameters are selected similar to the EnMAP EO satellite. Cloud coverage data is taken from an ECMWF global weather model. Three different operations scenarios are selected for simulation. On-board scheduling with a greedy algorithm showed that 6% - 44% additional cloud free area could be acquired compared to the ground based scheduling solution.

Finally the economical relevance of on-board scheduling is discussed based on the selected example mission. Hints for integration of on-board scheduling into a satellite mission are given, based on the project experience at GSOC.

Zusammenfassung

Scheduling von Erdbeobachtungssatelliten (EO) wird normalerweise in einem Bodenkontrollzentrum als Teil des Satellitenbetriebs und der Missionsplanung gemacht. Die resultierende Telekommandosequenz wird zum Satelliten geschickt und sequentiell ausgeführt. Durch das Missionsdesign (Orbithöhe, Ort der Bodenstation) hat der Satellit während der Ausführung der geplanten Aktivitäten keinen Kontakt zum Bodenkontrollzentrum.

Wenn der Schedulingprozess am Boden nicht alle für ihn relevanten Parameter kennt, ist es von Vorteil, diesen Prozess stattdessen im Satelliten auszuführen. Verschiedene Anwendungsgebiete wurden dafür identifiziert. Ein wichtiges Gebiet ist die Planung für optische Sensoren welche von Wolkenbedeckung in den Bildern betroffen sind. Ein Weiteres ist die Planung von Aufnahmen wenn verlustfreie Datenkompression benutzt wird. Für die On-Board Planung mit der Unsicherheit von verlustfreier Datenkompression wurde ein Patent beantragt.

Im nachfolgenden Kapitel werden die möglichen verschiedenen Konfigurationen und betriebsrelevanten Aspekte der On-Board Planung besprochen. Planungsbestimmende Gleichungen werden abgeleitet welche für die Analyse der Machbarkeit von On-Board Planung benutzt werden können. Ein Greedy-Algorithmus wurde für die Planung ausgewählt. Mit diesem Algorithmus wird anschließend eine Fallstudie für On-Board Planung durchgeführt. Die Missionsparameter werden zu diesem Zweck ähnlich zur Satellitenmission EnMAP gewählt. Wolkendaten des globalen ECMWF Wettermodells werden für die Simulation benutzt und drei verschiedene Betriebsszenarien simuliert. Es wurde gezeigt, dass On-Board Planung mit einem Greedy-Algorithmus im Vergleich mit der bodenbasierten Planung zur Aufnahme von 6% - 44% zusätzlicher wolkenfreier Fläche führt.

Schließlich wird die ökonomische Bedeutung von On-Board Planung anhand der simulierten Beispielmission diskutiert. Basierend auf der Projekterfahrung am GSOC werden Hinweise zur Integration von On-Board Schedulingsystemen in eine Satellitenmission gegeben.

1. Overview

1.1. Introduction

Space agencies as well as industry operate earth observation satellites. Central element for these daily mission operations is a mission operations center connected with a ground station (GS) used to send telecommands (TC) and receive telemetry (TM) and payload data for the satellite. The mission control center receives inputs from satellite data users on the requested type of data (e.g. target area and time). Requests are processed together with information about available satellite resources and current orbit by the mission planning system which is a part of the overall mission operations process.

As a result of the mission planning process a timeline is generated placing the necessary satellite activities in time taking into account resources and constraints. Based on the generated timeline a time-tagged satellite command sequence is prepared for up-link and executed by the satellite. The command sequence is usually immutable and executed by the satellite without taking any change in planning relevant boundary conditions (e.g. available on-board power or cloud coverage in the target area) into account. This reduces the maximum possible utilization of earth observing satellite resources, as boundary conditions may have changed between timeline scheduling on-ground and command execution on-board the satellite.

Additional satellite autonomy through on-board scheduling of user requests may increase the return of satellite imagery significantly. This work studies this problem therefore under the aspect of problem identification and simulation of a possible problem solution by on-board scheduling.

1.2. Scope of Work

Rather than scheduling all activities in advance on ground, satellites can execute this functionalities on-board (a good example is given in [GNT04]). This increases on-board autonomy allowing for a reaction of the satellite operations to changed scheduling relevant boundary conditions. [Alg03] states a possible increase in useful satellite data return by 100% through on-board scheduling. Applying such a way of operations requires an extension of the mission scheduling functionality from on-ground only to preparation on-ground and scheduling on-board the satellite. Scheduling of user requests can be executed on-board the satellite instead on-ground.

Goal of this work is analysis of conventional mission operations and comparing simulation of conventional scheduling and on-board scheduling. For simulation an application area is selected with an expected advantage for on-board scheduling.

For this, boundary constraints of mission operations and the conventional mission planning process are depicted. Potential application areas for on-board scheduling are to be identified. Finally a simulation of an interactive on-board scheduling system improving the utilization of an earth observing satellite with an optical sensor under changing cloud coverage is conducted as an example case. Results are compared to conventional mission operations, economic and technical impacts are analyzed.

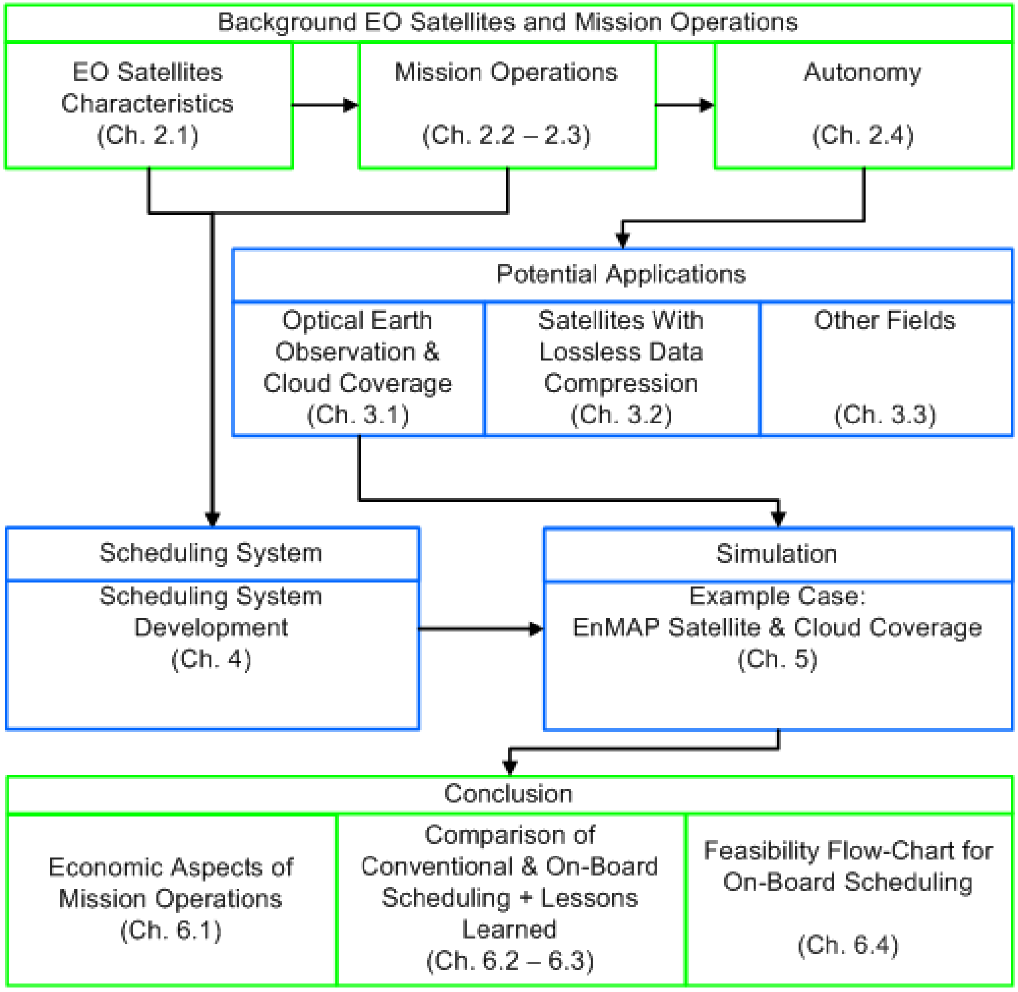


Figure 1.1.: Dependency of the different chapters

An overview of the chapters and relation between chapters can be found in Figure 1.1. The starting point for this work is the existing ground-based mission planning system as currently used in the DLR mission planning group. This mission planning system is operationally used in different earth observation missions at the German Space Operations Center (GSOC) [MGB$^+$08]. On-board scheduling may be a routine part of future satellite missions and is therefore of high interest for a space operations center.

Roadmap

In a first step the functionality of mission operations for satellites is discussed (Chapter 2) with the mission planning system as a subsystem and the driving boundary conditions of daily operations. This includes TC uplink (U/L) and payload data downlinks (D/L) as well as satellite resources. Based on the tasks of the common ground based mission scheduling process functionalities that can be moved on-board a spacecraft are identified.

Based on this task:

- different on-board scheduling application areas are identified - an important application is scheduling for optical earth sensors (imaging area occlusion due to cloud coverage constitutes a significant problem for these EO satellites reducing the return of useful images) (Chapter 3)
- an on-board scheduling system framework with a scheduling algorithm is implemented for simulation, augmenting the traditional ground-based mission scheduling process to interactive scheduling and execution during flight (Chapter 4)
- appropriate cloud data is identified and processed from meteorological GRIB (GRIdded Binary) files for simulation purposes (Chapter 5) and a standard EO mission using an optical sensor is simulated under the influence of cloud coverage (Chapter 5) in order to evaluate the different operation approaches

Finally the on-board scheduling solution is compared to the conventional ground-based solution in Chapter 6 (economical and technical impact). This shows advantages and disadvantages of both solutions. Chapter 7 concludes the results and an outlook for the next possible steps is given; which can be a starting point for further work.

2. Mission Operations of Earth Observing Satellites

Interactive operations scheduling is a very interesting field for many satellite missions. For this work the focus is placed on image acquisition scheduling for earth observing (EO) satellites as they are the relevant mission types operated by GSOC. Relevant parameters for meaningful on-board scheduling depend on the overall space mission architecture, therefore the mission operations relevant background of earth observation is given in the following chapters. First, operations relevant mission characteristics are analyzed and technical parameter of some example EO missions are collected. Second, mission operations with a focus on the mission planning and scheduling functionality as used by GSOC are discussed. In the last chapter of this section the current state of satellite autonomy is analyzed with respect to the idea of a further improvement by on-board scheduling.

2.1. Earth Observing Satellites Mission Characteristics

An EO satellite mission consists of the space segment (the satellite) and the ground segment (ground control center and ground station). Figure 2.1 shows the different mission elements. The ground control center controls the satellite operations by TC, commands are send with a ground station over a radio-frequency link (U/L). Ground station and control center are connected over data transmission lines and can be at different locations. Acquisition of a certain target is usually done by the satellite without contact to a ground station at the same time. The acquired data is therefore stored till next possibility for D/L.

Key satellite mission parameters from operations and scheduling point of view are

- the satellite acquisition capabilities:
    - the sensor type and its operations constraints
    - pointing capability of the satellite bus
    - immediate data storage capacity

- the selected orbit
- the ground station contacts and communication
- satellite command distribution

Satellite Acquisition Capabilities

Satellite sensors for earth observation may be active or a passive. The difference is the source of the radiation used. *Passive* sensors rely on an external radiation source, solar radiation for example. *Active* sensors illuminate the object under study by e.g. radar or laser, so they don't rely on an external source. The derived operations constraint is therefore that sensors using sun light for target illumination can not operate **if target illumination is not sufficient**.

A further constraint is the usage of resources during sensor operations. The most important one is sensor power consumption as well as immediate storage of raw data in an on-board mass memory. These parameters are mission specific and depend on the satellite bus capabilities. Sensor specific requirements like calibration measurement or cooling of detectors may also apply.

Finally the orbit together with the instrument footprint and the satellites target pointing capability drive the revisit time for a certain target.

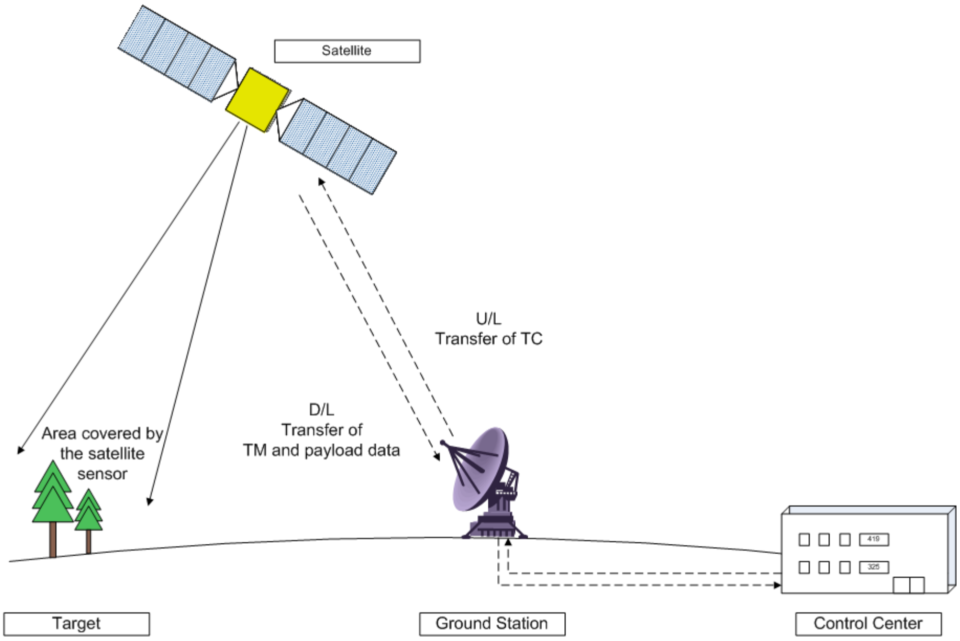


Figure 2.1.: Overview of the EO satellite mission elements

Table 2.1[1] shows the main technical parameter of some satellites of the 4^{th} generation.

Table 2.1.: Mission parameters of typical earth observing satellites

			LANDSAT 7	IKONOS	SPOT 5	EnMAP
Launch			1999	1999	2002	2012
Orbit						
	Type		SSO	SSO	SSO	SSO
	Altitude	[km]	705	709	833	643
	Inclination	[°]	98.2	98.1	98.7	97.96
Instrument			ETM+	OSA	HRG	HSI
	Spectral range	[μm]	0.45 - 2.35	0.45 - 0.9	0.43 - 0.89	0.42-1.0
			10.4 - 12.5	n/a	1.58 - 1.75	0.9-2.45
	Datarate	[Mbit/s]	150	n/a	n/a	822
	Power	[W]	590	350	n/a	n/a
	Swath	[km]	185	n/a	60	30
Mass Memory		[Gbit]	378	64	n/a	512
Data Compression Method			n/a	ADPCM	DCT	n/a
Payloaddata DL (X-Band)		[Mbit/s]	2 x 75	320	150	320

Orbit Characteristics

The selected orbit is mission operations relevant as it distinguishes the mission with respect to

- velocity of the targets on the earths surfaces as seen from the satellite
- illumination conditions for the target as well as satellite solar generator
- ground station contact times

It is usually a trade-off between coverage of the Earths surface, orbit decay and sensor needs. Most of satellite missions are using orbits around the Earth with eccentricities close to 0 and altitudes between 300-1500 km [Val01, p. 768-769][MG05, p. 2].

Especially for remote sensing purposes a complete coverage of the Earth is required with short revisit times. For this reason the sun-synchronous orbit (SSO) is often used by EO satellites. Advantages are uniform illumination of earths surface (target) and of the solar arrays [Val01] [MG05] [Pea94] and continuous coverage of most of the Earths surface [LW99] [Val01]. A SSO keeps a nearly constant orientation towards the Sun during the year as the nodal regression rate of the right ascension of ascending node is kept at a rate similar to the rotation of the Earth around the Sun (0.98564736 deg/day) [Val01].

---

[1] Data about the satellite missions from [Kra02] and [SKH$^+$07]

The satellite velocity (v) in orbit is relevant for the speed of the ground targets coming in sight of the sensor and is given by the following equation [MG05]:

$$v = \sqrt{\frac{2\mu}{r} - \frac{\mu}{a}} \tag{2.1}$$

with the gravitational parameter for an earth orbiting satellite

$$\mu \approx Gm_{\oplus} = 3.98600441\underline{5} \times 10^{14}\,\frac{\mathrm{m}^3}{\mathrm{s}^2} \tag{2.2}$$

Satellite velocity depends only on semi-major axis (a) of the orbit and actual distance from the center of mass (r). For circular orbits is $r = a$ and the velocity therefore constant. Velocity is than given by:

$$v_{circ} = \sqrt{\frac{\mu}{a}} \tag{2.3}$$

Orbit data for satellite are often provided in the format of two-line elements (TLE). Semi-major axis is not provided by the TLE but can be calculated based on the mean motion n using:

$$a = \sqrt[3]{\frac{\mu}{n^2}} \tag{2.4}$$

For EO satellites in circular or near circular orbit, the speed relative to the Earth is nearly constant. Targets relevant for on-board scheduling enter the reasoning horizon at a continuous rate (AAR is nearly constant for EO satellites). The selected orbit is only one operations aspect relevant for communication with the satellite. This is subject of the next paragraph.

Ground Station Contacts and Communication

Communication links between satellite and control center are used for command and control (TC and TM) of the spacecraft and reception of payload data. Use of one ground station for one satellite is considered as the usual case for an earth observing satellite. The two interesting operations relevant aspects are the ground station contact time and the amount of data that can be transferred during this contact.

Satellites in SSO have only limited contacts to a ground station as they are changing their position relative to the Earths surface. Depending on the position of the ground station (especially the latitude) on the Earth, contact frequency may be once per orbit or only some orbits per day. This depends on the given orbit geometry and ground station location. Minimum time without ground station contact is therefore one orbit for each EO satellite in a SSO with only one dedicated ground station. During this time the satellite is not in contact with the ground station and no command and control is possible. Duration of the ground station contact depends on orbit geometry, the type of station used and does usually not exceed some minutes - which is only a small fraction of one orbit. Figure 2.2 depicts the EnMAP satellite ground track with contact to Weilheim ground station in Germany (long. 11.078°, lat. 47.880°, assumed minimum elevation 8°). For a certain satellite mission/ground station combination contacts and durations must be analyzed by software simulation.

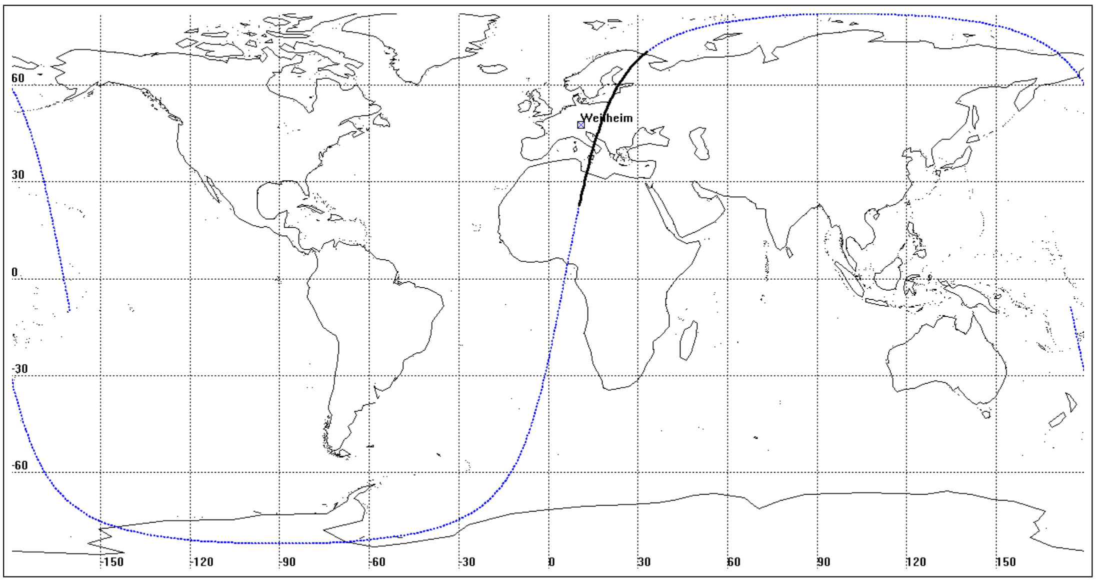


Figure 2.2.: Satellite ground track of earth observing satellite

The link itself is usually using radio frequency (RF) for information transport. Optical communication using laser technology is a further option with higher data rates but still far away from being a standard. Furthermore optical communication with laser has potential problems with visibility of the ground in case of clouds.

The following data is transferred:

- TC via U/L from the control center to the satellite
- TM via D/L from the satellite to the control center
- sensor data via D/L from the satellite via control center to the user

For earth observing satellites the amount of sensor / payload data usually exceeds the amount of TM and TC by orders of magnitude. Therefore the amount of transferred payload data is relevant for mission operations.

Sensor data D/L rates depend on the frequency band, the used hardware and coding mechanism. Required transmitter power and antenna size are positively coupled to the data rate

[LW99, p. 542]. Furthermore it is necessary to differentiate between gross data rate and net data rate due to the used error coding. Example data rates of satellite sensors and the corresponding D/L data rates can be found in Table 2.1. A trend is the increasing amount of data produced per second by the sensor while the D/L datarate is increasing much slower. An example is the development of data rates for the Landsat 1 - 7 EO satellite[2] and EnMAP, the relation between sensor data rate and D/L data rates becomes:

- ≈ 0.18 for Landsat 1
- ≈ 1.0 for Landsat 5 - 7
- ≈ 2.6 for EnMAP

This leads to the conclusion that D/L of payload data is increasingly the bottle neck of EO missions, as increasing amounts of data are produced per second by modern sensors. On-board scheduling coupled with deletion of unneeded sensor data may be a potential solution.

Satellite Command Reception and Distribution

On-board planning requires control of image acquisitions by software running on-board the satellite. Usually telecommands are used to control satellite activities from ground. TC are received by the telemetry tracking and command (TT&C) subsystem of the satellite which is sometimes also called the communication subsystem. Commands are usually formatted due to standards, today mostly following CCSDS (consultative committee for space data systems) recommendations [FS95] [LW99] [fSDSC]. The TT&C subsystem is the responsible for the communication between satellite and ground station.

Received commands are forwarded from TT&C to the command and data handling (C&DH) system. C&DH is responsible for command arbitration, command validation and command decoding. Depending on the complexity of the satellite and the mission the function of TT&C and C&DH may be integrated into one system [Gar96]. Usually bigger satellite missions divide functionality into two separate subsystems due to overall system complexity.

Command sources for C&DH can be the U/L from a ground station (via TT&C subsystem) or an on-board computer [LW99]. The up-link from ground is the nominal way commands are loaded into the satellite. The on-board computer may be a command source e.g. in case of low power conditions. An overview of command sources and distribution can be found in Figure 2.3. Immediate commands are sent by C&DH to the required subsystem. Time-tagged commands are stored till the execution time is reached and than forwarded to the subsystem.

C&DH also provides a time reference which is needed for execution of time-tagged TC. C&DH also gives the granularity of time which is the smallest increment of time used on-board the satellite. It is driven by the requirements for data time-tagging and commanding accuracy.

---

[2] Data about Landsat satellites as published by Kramer in [Kra02].

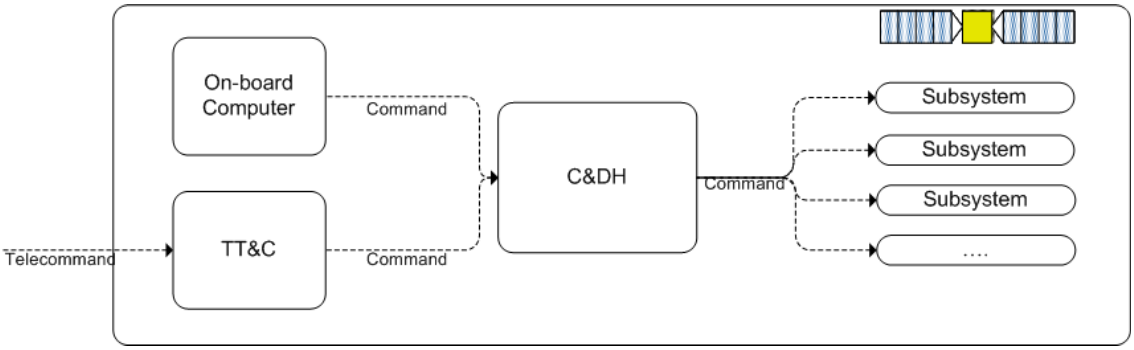


Figure 2.3.: Command distribution within a spacecraft

Conclusion

This chapter showed acquisition scheduling relevant aspects of EO satellite mission operation. Those satellites use usually a SSO and are not in continuous contact with the control center. The data source is usually a sensor generating high amounts of data. An on-board mass memory stores the data in the time between data acquisition and next possible D/L. During ground station contact as much as possible data is down-linked. As the sensors are producing increasingly more data than can be down-linked in the same time interval the D/L is usually the bottle-neck of data transfer for modern EO missions. A solution may be on-board scheduling which allows down-linking of only the interesting data.

2.2. Mission Operations Principles for Earth Observing Satellites

The mission control center is the central instance for the operations controlling the satellite [LB96].

> **Mission operations** can roughly be defined as the process of controlling satellites to fulfill a given set of mission requirements.

The current way of mission operation is preparation of all satellite TC on ground. On-board planning and scheduling would move some of this functionality to on-board the satellite away from the control center. This chapter therefore gives a short overview on the way of conventional mission operations for an EO satellite with command and monitoring of the spacecraft achieved by a bidirectional system using TC and TM for communication [Gar96].

Figure 2.4 depicts the mission operations cycle and the relevant systems (see also [WL91]). The mission is separated into different segments. These segments are distinguished functionally with interfaces connecting them. It can be separated between the *ground segment* (sometimes also referred as earth segment), the *launch segment* and the *space segment* ([fSDSC][ECSS-E00A], [WL91], [Gar96]). For routine mission operation the launch segment can be neglected because it is only relevant for mission preparation and during LEOP (launch and early orbit phase - the phase of satellite activation and testing right after launch). This leaves ground and space segment as the main segments for the phase of routine mission operations. Both segments consist of different elements: The space segment of the spacecraft (satellite bus and sensor), the ground segment of the mission control center, ground stations, communication infrastructure and the payload data user.

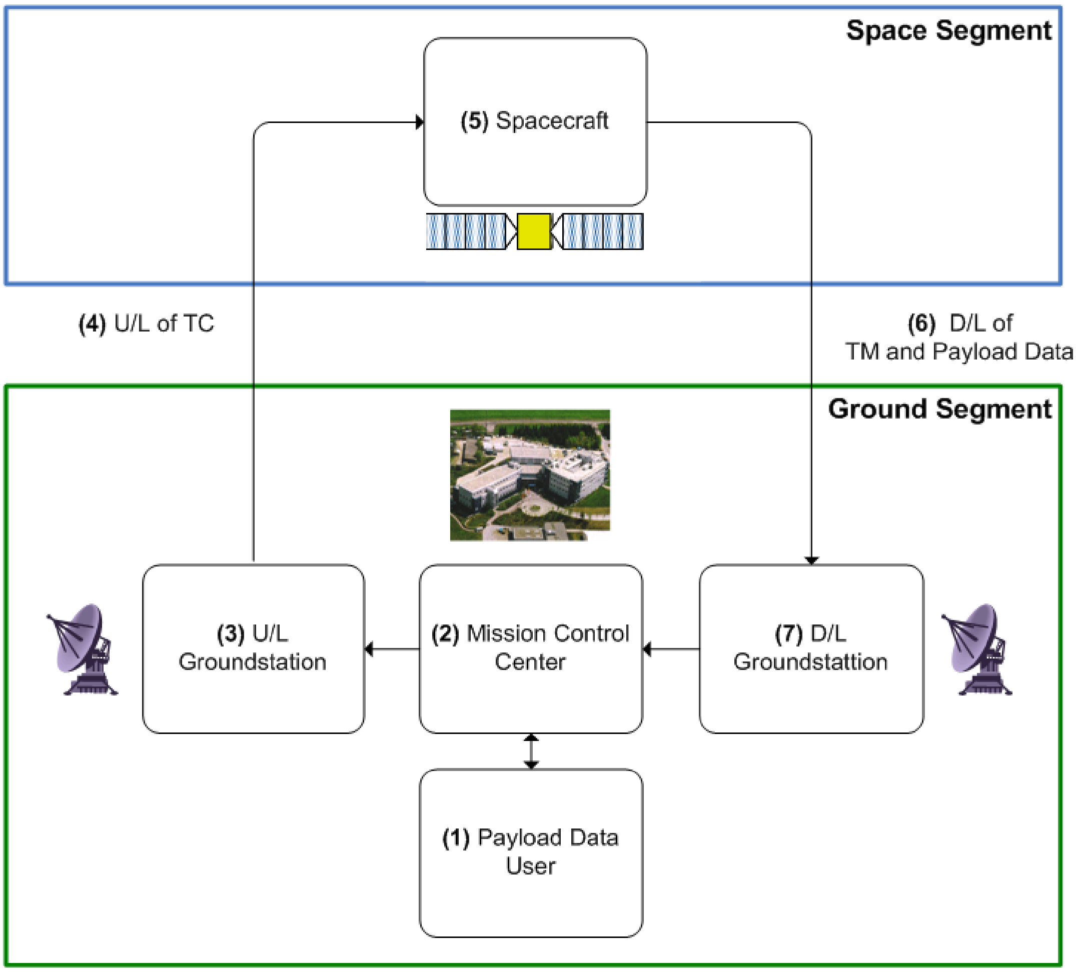


Figure 2.4.: Mission operations elements overview

The interested user prepares request for data of a certain target on the Earth (1). The mission control center (2) is responsible for preparation and up-link (3) of TC (4) to the spacecraft (5) based on the requirements. After execution of the TC the corresponding TM and payload data (6) is down-linked during ground station contact (7). Data is than provided to the user (1).

All software required for commanding the spacecraft and analysis of telemetry is also provided by the mission control center. Ground stations provide the physical up-link and down-link capability to the satellite. They are connected by dedicated data links to the mission control center and can therefore be at the different locations on earth - thus allowing a flexible design of the ground station network.

Due to the limited communication between EO satellite and the responsible control center mission operations is executed in a cyclic manner. Figure 2.4 shows the interaction of the two different mission operations activities ([WL91]): First operations preparation and command up-link to the spacecraft. And second, TM and data reception including data processing. U/L of telecommands and D/L of telemetry are usually synchronous processes during contact between satellite and ground station.

Figure 2.5 depicts three different datatake (DT) cycles:

- at **moment A**, when the satellite **is in contact** with the control center via ground station
  - data and TM of datatake cycle n-1 is down-linked from the satellite to the ground
  - the control center up-links new TC of datatake cycle n for time-tagged execution by the spacecraft
- at **moment B**, when the satellite **is without contact** with the control center
  - down-linked data and TM of datatake cycle n-1 is processed by the control center
  - time-tagged TC of datatake cycle n are executed by the spacecraft controlling image acquisitions
  - TC for datatake cycle n+1 are prepared by the control center for up-link during next contact
- at **moment C**, when the satellite **is in contact** with the control center via ground station
  - data and TM of datatake cycle n is down-linked from the satellite to the ground
  - the control center up-links new TC of datatake cycle n+1 for time-tagged execution by the spacecraft

Preparation activities consist of reception and processing of user requests, e.g. the planned area for image acquisition. This information is combined with the orbit and attitude information together with the information on satellite resources envelopes. Those envelopes describe for example the amount of power and memory available for data takes. Based on those limitations an activity schedule is generated and translated into a command sequence which can be up-linked and received by the satellite and processed by the C&DH subsystem (see Chapter 2.1.6). The corresponding telemetry is received during subsequent passes (see Figure 2.5). All telemetry is than processed and information relating resource envelopes, user requests, orbit and attitude is updated.

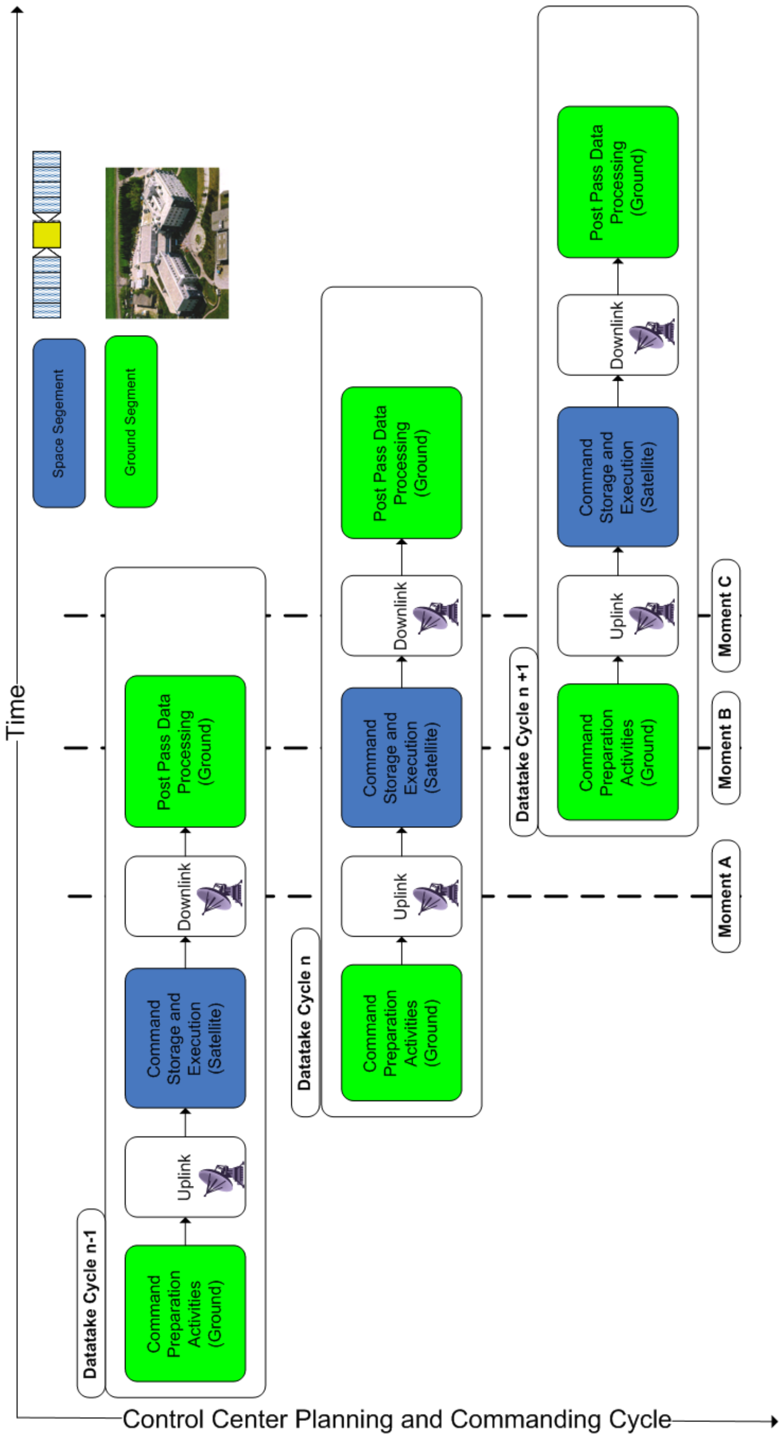


Figure 2.5.: Mission operations dataflow

Moment B in the diagram: While the results (TM and payload data) of cycle n-1 are processed and evaluated, commands of cycle n are executed on-board the satellite and the datatake commands of cycle n+1 are prepared in the ground control center.

The complete process is running in a cycling manner (see Figure 2.5), new commands are uploaded to the satellite while the results of the preceding commanding (telemetry and payload data) are down-linked and processed. Therefore it can be seen as an open loop control. Results (telemetry and payload data) of the commands prepared, uploaded and executed during cycle n-1 are available after generation and up-link of the commands for cycle n. Any reaction to the results of a telecommand is possible at cycle n+2 at the earliest.

2.3. Mission Planning as a Subsystem of Mission Operations

The idea of on-board planning and scheduling is an extension of parts of the conventional process from on-ground to on-board the spacecraft. This chapter analyzes the on-ground mission planning process as applied by GSOC together with the relevant data flow. Furthermore, as there is no commonly used definition of the mission planning function, this chapter defines the term and shows the differentiation to other possible meanings. Furthermore the conventional mission planning process is explained with its different steps as currently used by German Space Operations Center.

In the GSOC the mission planning system (MPS) is a subsystem of the overall satellite mission operation system [LWH07, p. 453]. As there is no common definition the function of an MPS it is necessary to lock at the definitions used in relevant literature and to define the term as used by GSOC. [ZF94, p. 169] defines planning as ”how to achieve a given goal by selecting a certain set of activities whereas scheduling means assigning times and resources to these activities”. [LB96, p. 37] uses the word *mission plan* to describe the way the mission will be flown, expresses objectives in operational terms, and sets in place major activities. [LW99, p. 601–603] uses *mission planning* in conjunction with *activity planning* as a two-stage process of defining a command sequence for execution in the framework of mission operations. Mission planning defines a rough activity timeline across mission phases whereas activity planning produces a sequence of executable commands, which are constraint checked and conflict free.

As shown before, mission planning and scheduling are not a clearly defined term in space operations, so each company or agency defines the functionality of this system more or less on their own. Only the function core is usually the same. For the purpose of this work the definition for the mission planning and scheduling system is as follows:

> **Mission planning** is processing of user requests for the generation of a spacecraft TC sequence for a given set of goals under constraints imposed by the spacecraft mission.

Mission planning in this context is therefore not considered to be a mission preparation process with calculations for best satellite design and necessary ground system development (this should be meant by the expression mission design and analysis). In this case it is the process of every day activity preparation for the satellite.

In the framework of low earth and especially earth observation satellite missions the planning and scheduling subsystem defines execution of orbit maneuvers, satellite modes (e.g. sun-pointing or nadir pointing), commanding of data takes and other activities. The functions planning and scheduling must be distinguished as defined by [ZF94, p. 169]. For operation of earth observing satellites it is usually clear which activities (e.g. activation of sensor or downlink of data) must be executed to reach a certain goal. These activities are provided by the satellite or payload designer to the control center as commands, command sequences or procedures describing the execution of an activity. The job of mission planning and scheduling is to schedule the activities by assigning execution times and to check satellite resource usage by the scheduled activities. This is especially necessary if several hundred activities are to be planned and many constraints have to be checked.

Figure 2.6 shows the flow of a user request from entering the control center up to delivery of data. The steps shown in the figure do not cover all processing steps but the planning relevant ones. A user request is the requirement of certain data delivered by a satellite sensor for example an optical image of a lake in Germany. The request is usually defined by:

Table 2.2.: Elements of a user request for satellite data

Request Information	Comment
Target Area	Determination of the geographical target
Time Slot	Time for acquisition
Priority	Depending on the user different priorities may be assigned
Sensor Configuration	Some sensors need certain configurations for acquisitions
Acquisition Constraints	e.g. max. off nadir angle, max. cloud coverage

Once the user request enters the mission planning process the possibilities [1] must be calculated to acquire data from the defined target. If the target is greater than the sensor swath the area must be split into more than one acquisition. This requires data about the orbit, usually delivered to the calculation software as a TLE (two-line element)[2]. After calculation of opportunities and splitting into single acquisitions the data is prepared for the planning process [3]. The planning process processes prepared acquisitions (usually from a database) and generates a schedule. Beside the user requested data acquisitions other activities like D/L of data or orbit keeping maneuvers are also included by [3]. Schedule generation requires knowledge of the satellite resources [4] the constraints [5] set and engineering requests [6]. Constraints to mission planning process are given by the design of the mission (spacecraft margins, orbit geometry, mass memory, power consumption etc.) and by project management. The result is a timeline [7] which is translated into TC's [8] and up-linked to a satellite.

After reception of the telecommands and immediate storage [9], time-tagged commands are executed [10] by the satellite and generated raw sensor data is stored [11]. Immediate storage of TC's and sensor raw data is driven by orbit geometry and used ground station network. After D/L of raw data, different processing stages are usually passed before the requested data is delivered to the user. The complete process shown runs in a cyclic manner with usually no interaction (from a planning perspective) on-board the spacecraft [PG02].

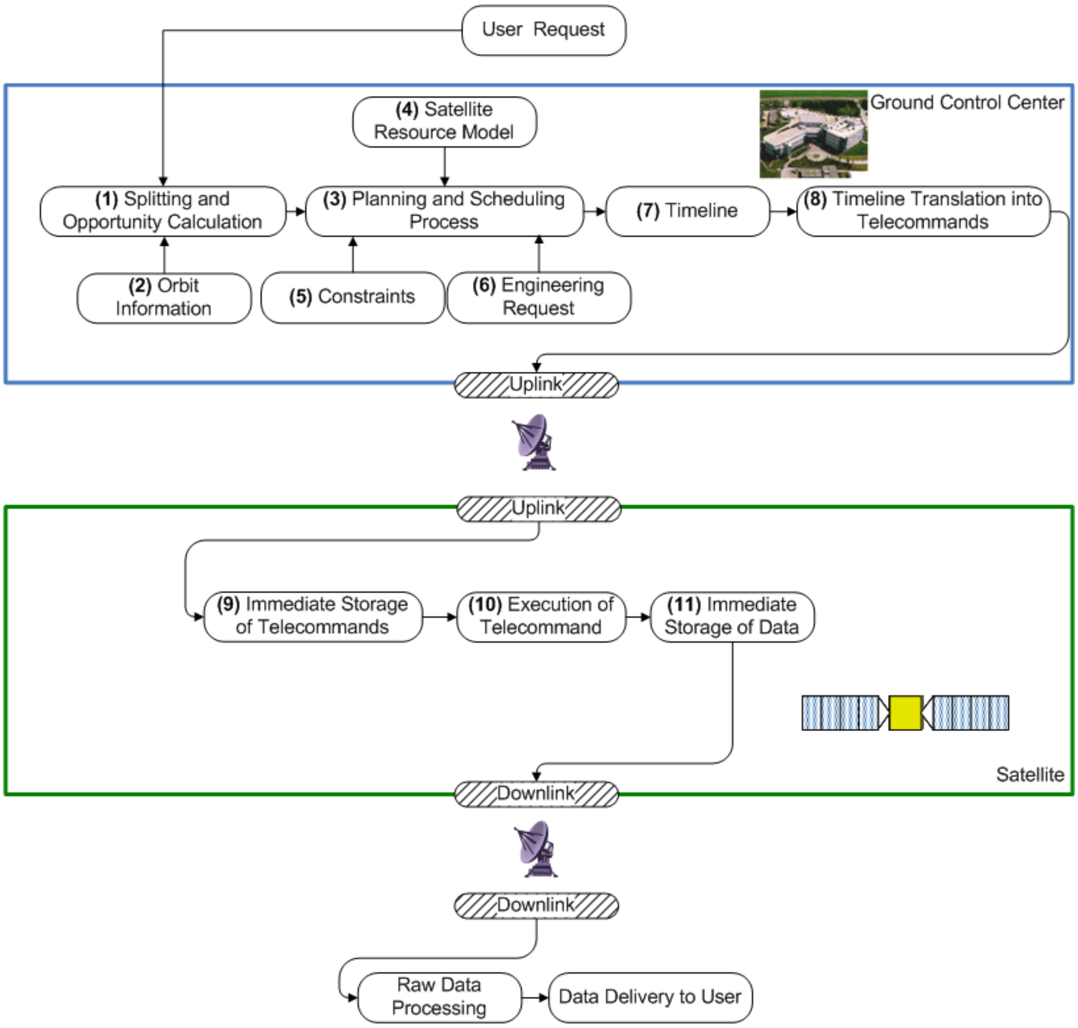


Figure 2.6.: Flow of a user request through mission operations

The system used for planning and scheduling in GSOC is based on different core software components [LWH07, p. 461–462]. Those components are adapted to planning and scheduling needs of each mission. Additionally, mission specific software is developed and integrated if necessary. All tools are developed in high-level programming languages as IDL, C++ and .NET languages for use in the ground control center environment. The productive environment is therefore based on standard PC's with Microsoft Windows operating systems (OS). Table 2.3 lists the tools used by ground based mission planning in GSOC. In the following the function of the different mission planning core tools are explained. The basic idea of this work has been to select the core mission planning components and to use them with some modifications on a satellite platform to perform scheduling functionality. This would be a modification and adaption of existing components.

Table 2.3.: Generic planning and scheduling software tools used at GSOC

Tool	Purpose
SEPPL	Calculation of planning events
PLATON	Automated timeline Generation
ATLAS	Event visualization
PINTA	Visualization and manipulation of timeline
TimOnWeb	Visualization of timeline

The Planning Tool (PLATON) is used for timeline generation. With this function it is the core element of the GSOC mission planning system. For the TerraSAR-X radar satellite mission planning system PLATON needs roughly a system equipped with 1 GB RAM, a 1.7 GHz processor and several MB of hard disk space.

SEPPL (Self Pointing Processor Library) is a tool used for event generation. It is developed and maintained by the GSOC mission planning group in C++ running on a standard PC. Extensions and changes to this software are made for each project which depends on specific needs.

PINTA, TimOnWeb and ATLAS are tools used for visualization. The Program for Interactive Timeline Analysis (PINTA) is used for timeline creation and interactive timeline changing. The tool visualizes resources, activities and constraints. TimOnWeb is a web-based server application with a user front end similar to the GUI of PINTA. It allows the timeline to be visualized over the Internet, mainly used in missions with a widely separated user community to make planning information available via the Internet. ATLAS stands for Advanced Trajectory Lookahead and Analysis System and is software used for analysis of satellite orbits and visualization of planning relevant events.

The different used high level programming languages have already precluded to use a simple copy of the existing tools on-board a satellite. Also the hardware demanding performance needs make the use or adaption of the existing tools not feasible for embedded systems in an useful way.

In this chapter the term mission planning has been clarified as used and understood by GSOC MPS group. The process from a user requesting sensor data up to the delivery of final data has been shown from the perspective of conventional mission planning. This process can only be improved if parts of this process are moved to on-board mission planning. The tools used in GSOC for the conventional planning process have been shown. With respect to the heterogeneous use of different programming languages and high performance needs it becomes clear that an adaption of the existing tools to on-board planning is not possible.

2.4. Current Spacecraft Missions and Autonomy

This chapter discusses the current state of satellite autonomy with respect to daily mission operations and on-board scheduling as this may be already state of the art and nothing new. Emphasis is placed autonomy and the area of planning and scheduling with of survey of example missions.

> **Autonomy** is usually considered to be the capability of a system to act on it's own within certain limits.

Each modern satellite mission uses autonomy functions up to a certain level of complexity. In the context of this work it is necessary to classify autonomy and identify the degree of autonomy that is currently used by satellite missions and therefore considered to be state of the art. Satellite autonomy enables limited reaction of the spacecraft to changed boundary conditions or system states. This can trigger reactions of the satellite to increase mission security (e.g. by shutting down systems), enhance the possible ways of reaction or reduce mission operations cost. Development and implementation requires additional financial effort ([LB96, p.42], [LW99, p.890]) and must be seen in relation to potential savings.

Satellite autonomy levels can be defined according to [LW99, p.617] in four steps which are shown in Table 2.4. The lowest level of autonomy uses on-board control loops whereas the fourth level includes on-board adaptive planning.

Table 2.4.: Levels of satellite autonomy

Level	Technology Used
1	On-board control loops
2	Usage of time-tagged command sequences
3	Execute fault response rules
4	Adaptive planning and resource management

Level 1 is used be all satellites. Examples for on-board control loops are subsystems switching heating on or off based on predefined temperature limits. These are features realized by hardware and or software on a very low level. *Level 2* is the usage of predefined command sequences which are stored and executed at a certain point in time (TT-TC). Those commands are ground generated, but high level commands are sometimes expanded to a set of low level commands. *Level 3* autonomy is equal to FDIR (Failure, Detection, Isolation and Recovery) - a mechanism takes control in case of unforeseen system states and executes predefined command sequences. These commands shall secure the satellite till next contact to the ground control center. *Level 4* is adaptive planning and resource management. This has not yet been demonstrated by any spacecraft although future generations of earth observation satellites are expected to be more autonomous than the current generation [ZK02].

Two different tendencies for future development in the area of level 4 autonomy can be identified by literature review:

> ***The first one*** **is the development of autonomy for scientific purposes, for technology demonstration or special mission purposes.**
>
> Example missions are Deep Space 1 (DS-1) or Proba 1. DS-1 is a NASA new millennium program spacecraft launched on October 24, 1998. Main goal was testing of new technologies during a flight to comet Borelly. One of the technologies on-board DS-1 relevant to autonomy was Autonomous Remote Agent (RA) system [GNT04, p. 451]. RA was used for experimental automated planning for DS-1 between May 17 and May 21, 1999. Proba-1 (Project for On-Board Autonomy - 1) is a minisatellite launched on 22 October 2001 with the Indian PSLV launcher. The satellite is an initiative of the European Space Agency where the mission goal was the technology demonstration of autonomous guidance ([Ber00]), navigation, control, on-board scheduling and payload resources management. Mission planning and scheduling is done by the OBMM (On-board mission manager). In the complete autonomous mode, the operations requests are checked on ground by the mission planning tool and then up-linked to the satellite with the OBMM scheduling the requested activity in the most efficient way [ZK02].
>
> ***The second tendency*** **is development into the direction of more autonomous operations of EO satellites.**
>
> Earth Observing - 1 (EO-1) is an example satellite, launched on November 21, 2000. Three advanced imaging payloads are flown on-board, the Advanced Land Imager (ALI), Hyperion and the Atmospheric Corrector (AC). The satellite is equipped with two Mongoose M5 processors running at 12 MHz (approx. 8 MIPS) with 256 MB RAM each. Autonomous mission planning is done by the CASPER (Continuous Activity Scheduling Planning Execution and Replanning) software [SCT$^+$07]. The software has been uploaded to the spacecraft for the flight experiment in mid-2003 after launch in November 2000. One satellite payload, the Hyperion instrument, is used for activity scheduling by the autonomous mission planning software in conjunction with on-board science event detection, e.g. on-board cloud detection [MMGB03]. Different events detected by the satellite are the starting point for the planning process. First experiments focused on the detection of thermal anomalies and cloud coverage in sensor data. These events cause rescheduling by CASPER using a local search approach to develop operations plans. CASPER used a ground-based orbit analysis tool for the over-flight opportunities calculations. Additionally, the pointing requirements have been uploaded to the satellite as a table and the initial management of the momentum wheels is done by the ground operations team.

The different missions mentioned showed different ways of autonomy for mission operations concerning planning, scheduling and resource management. Those operations have been on experimental basis and not for regular mission operations. It can be concluded that the current way is satellite operation by command and control from ground as described in the preceding chapters. The development of extended autonomy concerning satellite operations is demonstrated by some experimental missions as described above. Those missions show the potential mission autonomy that can be used for future EO satellite generations. Around 30 new EO missions are expected for the next decade following the news, see also [Fer08b]. On-board acquisition scheduling may extend the autonomy and usability of those satellites.

The potential of more autonomous EO satellites operations by on-board scheduling will be analyzed in this work in more detail.

3. Reasons for On-Board Scheduling

The preceding chapter showed the framework where mission planning is executed in daily mission operations and introduced the idea of executing this functionality on-board the spacecraft. But there must be a reason to move this functionality. This chapter discusses possible reasons for on-board scheduling and the application areas. The focus is identification of these areas and there potential importance for today's EO satellite with respect to better use of satellite resources.

So how can a potential application area be identified? Based on the design of a ground based mission planning process as shown in Chapter 2.3 the following definition for a potential application area for on-board scheduling can be used:

> **On-board scheduling** is beneficial to all spacecraft with non continuous control center contact and subject to a changing environment which can not be accounted for during ground based mission planning process.

The attribute non continuous control center contact maps to EO missions as well as to deep space missions. EO satellites in SSO have only limited contact to any ground station on the earth due to the used orbit. Most earth observing satellites use sun-synchronous orbits. Depending on the used ground station only a small period of time per day can be used for communication with the a ground control center.

Also the second attribute of a changing scheduling relevant environment applies to EO missions. Cloud coverage is a problem for optical sensors but also data compression as shown in the following chapters. Applications are also thinkable for deep space missions, like the NASA mars rovers, but are not discussed here.

3.1. Earth Observation and Cloud Coverage

Cloud covered areas on the Earth can not be acquired by optical EO satellites. The effect can be seen in Figure 3.1[1], it shows clouds and cloud shadows blocking the view on a target. The effect of clouds is twofold: First the direct blocking of target areas and second the shadowing (as shown in the picture). The effects of the shadowing can be corrected by post processing

---

[1] This example image has been taken from GoogleEarth

of the images; direct blocking can not be corrected. Basis for development of clouds in the atmosphere is evaporation of water from the oceans which increases moisture in the atmosphere. Clouds develop from under-cooled water or ice crystals in the atmosphere depending on the actual temperature of the atmosphere [Mal94, p.90]. The troposphere (0 - 18 km height) is the part of the atmosphere where most of the clouds occur. Higher clouds for example in the stratosphere are rare. Within the troposphere clouds are classified by type and height (low, medium and high).

Development of clouds over target areas can not be accounted for during the ground based mission planning process. So it may be potentially interesting for on-board scheduling. But is it a significant impact on the return of EO satellite image data? What is the maximum cloud coverage in imagery accepted by the customer?

Figure 3.1.: Cloud coverage over Munich airport

[vdLRdC$^+$99] gives a maximum value of 15% of the image area covered by clouds as the threshold in practice for selling satellite imagery. This threshold has to be seen in relation with the fact that on average 58.4% [RS99] of the Earths land masses are cloud covered. Regional cloud coverage varies due to location and season. Example amounts of scene cloud coverage for European regions are given by [vdLRdC$^+$99], namely the Netherlands, Spain and Greece. Investigation covered images taken by the following satellites: SPOT, Landsat and IRS-1C. The percentage of useless classified images due to cloud coverage can be found in Table 3.1. Useless classified images have a cloud coverage of $> 15\%$. For all satellites an investigation period between 1 - 2 years has been used, therefore seasonal effects are covered by this study. It can be concluded that more than half of the images are not useable and cloud coverage is a significant problem.

Some EO satellite missions handle this problem by taking cloud coverage forecasts into account during ground based scheduling process ([AGG01] and [GC00]). Examples are given in [WZ96] for MOMS-2P or in [AGG01] for LandSat 7.

Table 3.1.: Percentage of useless images - different mission examples [vdLRdC$^+$99]

Satellite	Netherlands	Spain	Greece
SPOT	86%	59%	57%
Landsat	95%	63%	62%
IRS-1C	74%	38%	69%
Total	88%	53%	63%

This forecast is valid at the time of scheduling of acquisitions on ground and conditions at execution of the datatake may deviate significantly. As a result mass memory is filled with a certain amount of cloud covered data which is finally down-linked and useless for further processing. On-board scheduling can potentially reduce the amount of useless data by avoiding or deleting it before acquisition and downlink and better satellite resource utilization. Relevance of cloud coverage is also shown by different patents related to this problem and the quality of satellite data for EO satellites, like Borg and Fichtelmann in [BF04] and [BF05] or Weiner in [Wei05].

3.2. Lossless Data Compression

Lossless compression of sensor data is a further potential application area for on-board scheduling. EO satellite sensors often produce high amounts of data during operations. As shown in Chapter 2.1 EO sensor data is stored in satellite mass memory till the next ground station contact and D/L.

Two aspects of EO satellite mission operations are potential limitations for this temporal storage. First, the size of the satellite mass memory is a limiting factor for the temporal on-board data storage. Second, down-link data rate and average down-link time may also be a limiting factor for the amount of data that can be transferred to the ground during daily operations. Both factors, down-link time and on-board mass memory storage, are cost driver for the whole mission and therefore kept at the absolute minimum.

A possible solution to is the usage of compression for sensor raw data ([Say06, p. 2], [fSDSC, CCSDS 121.0-B-1]). Data compression is transformation of data into a format that requires less space. It basically consists of the two algorithms (see Figure 3.2) compression and decompression (or reconstruction). Compression takes the raw input data (x) to construct a representation requiring fewer bits for presentation (x_c). Decompression takes the compressed representation of data to generate a reconstruction of original raw data (y).

Table 3.2.: Compression ratio used by some space missions

Satellite	Launch	Compression Ratio
IRS-P5	May 2005	3.2:1
MTI	March 2000	2.5:1
IKONOS-2	September 1999	4.25:1
SPOT-5	May 2002	3:1

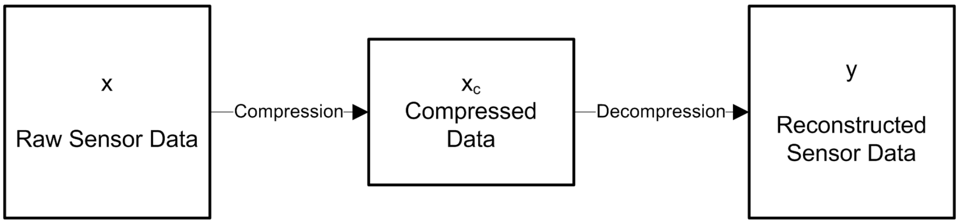


Figure 3.2.: Generic compression process

Two basically different compression schemes are available and can be classified in the categories *lossless compression* and *lossy compression* [Say06, p.4] [LW99]. The classification is based on reconstruction requirements of information after decompression compared to the original information.

> **Lossy Compression** ($x \approx y$): In case of lossy compression some data is lost during the compression and decompression process at a certain level that is accepted. Information of raw data differs from information of reconstructed data after decompression.
>
> **Lossless Compression** ($x = y$): Lossless compression is used if data after decompression is required to be identical with the original data before compression.

The most important performance measure for both compression schemes is the compression ratio cr:

$$cr = \frac{x}{x_c} \tag{3.1}$$

Lossy compression techniques achieve usually higher compression ratios than lossless compression techniques. Ratios can achieve values up to 80:1 for lossy compression and 5:1 for lossless compression according to [LW99]. An overview of space missions using lossy or lossless compression methods can be found in Table 3.2 based on information given in [Kra02].

The maximum achievable compression ratio depends on a quantitative measure can be defined called self-information of a data source, according to Shannon.The self information can be used to estimate the entropy of a data source. For lossless compression schemes the entropy limits the maximum reachable compression ratio possible [Say06, p.16]. In case of lossy compression this limit is ignored at the price of a difference between original and reconstructed data called distortion (loss of original information).

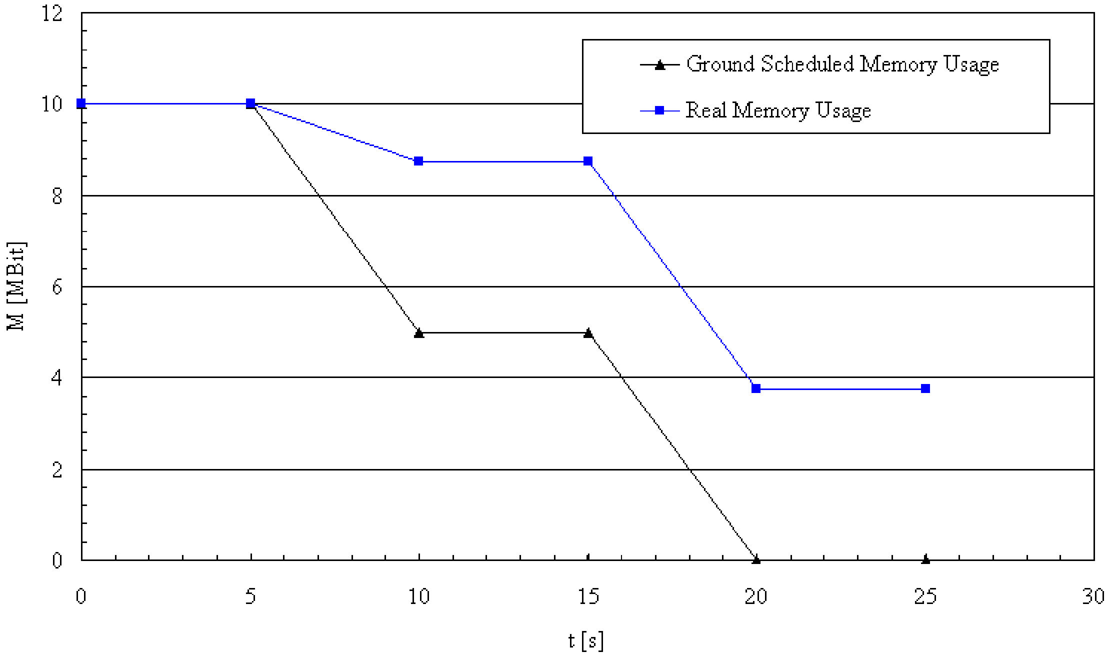


Figure 3.3.: Example of the lossless raw data compression problem

For satellite sensors the entropy of the data to be compressed can be assumed to be different for each data set (e.g. different images). So the achievable lossless compression ratio varies for each image stored in satellite mass memory. It follows that the achievable compression ratio can not be known without knowing the data itself if lossless compression techniques are applied.

If the compression ratio is not known during ground based acquisition scheduling an unknown filling state of satellite mass memory during operations execution is the result. Memory may either be filled more than expected or more empty than expected. First case prohibits additional datatakes whereas the second case allows additional datatakes. Figure 3.3 is an example illustration of the problem:

- 2 images with a push-broom scanner are too be executed, each 5 s long
- available satellite mass memory of 10 Mbit

- sensor raw data rate of $2\,\mathrm{Mbit/s}$
- ground based scheduling assumption for compression ratio for all images: $\frac{2}{1}$
- real compression ratio for 1^{st} image: $\frac{4}{1}$
- real compression ratio for 2^{nd} image: $\frac{2}{1}$

This problem can potentially be solved by on-board scheduling taking into account the state of the satellites mass memory after each additional image acquisition. This ideas has been detailed as a patent written by the author [AW09]. The patent has been granted.

3.3. Other Reasons

In the two previous chapters potential application areas for on-board scheduling have been identified. Cloud coverage and lossless data compression are common problems of current EO missions but of course further applications of on-board planning systems exist.

A possible further application area for on-board scheduling may be ”targets of opportunity” [PG02]. Examples are forest fires, volcanic eruptions, ice shelf erosion or military applications as identification of training sites and movements on airfields [SO02]. In all cases it is necessary to identify the interesting target characteristics on-board the spacecraft as they are not known during ground based scheduling.

A use case may be observation of forest fires. The satellite get’s information about 10 possible sites of forest fires world wide, but the system is limited to maximum storages and D/L of 4 images. In this case the on-board scheduler schedules the first 4 observations. The observation are than qualified on-board the satellite. This may be feasible by using e.g. neural networks or detection of intensity thresholds. If only in two images fires are detected another 2 images of the remaining 6 sites can be scheduled. This allows a more light-weight satellite (less hardware necessary for e.g. power system or mass memory) for the observation of 10 targets as usually only some of them are interesting.

The general assumption of the ground based scheduling process is the knowledge of the satellite resources available during operation. This is valid as long nothing unplanned happens, e.g. subsystem failure etc. An example may be a satellite permanent or temporal attitude control problems. As the satellites solar panels are not continuously pointed towards the sun it is impossible to know exactly the batteries state of charge. Thus an on-board scheduler can command datatakes if energy level and nadir pointing constraints are sufficient.

4. Development of an On-Board Scheduling System

The basic idea of on-board scheduling is scheduling of the datatake sequence on-board instead of on-ground and advance.

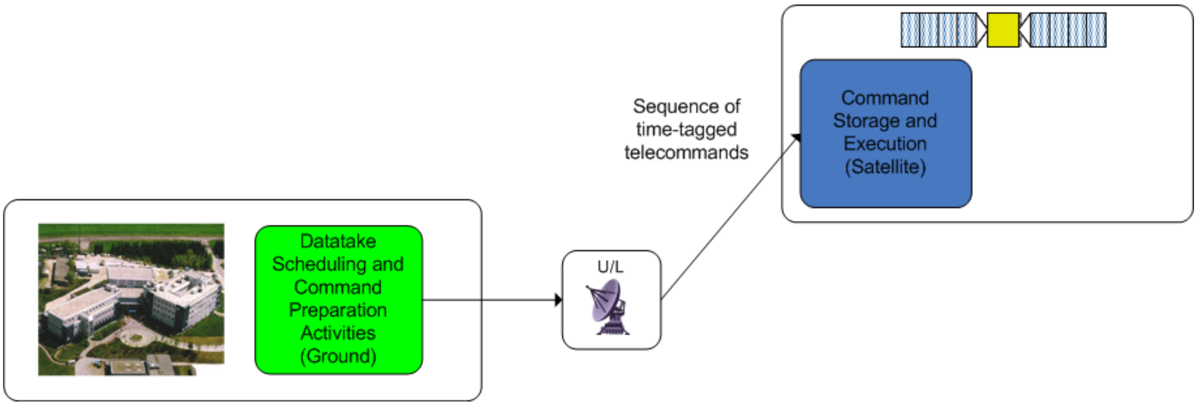


Figure 4.1.: Information flow when using on-*ground* scheduling

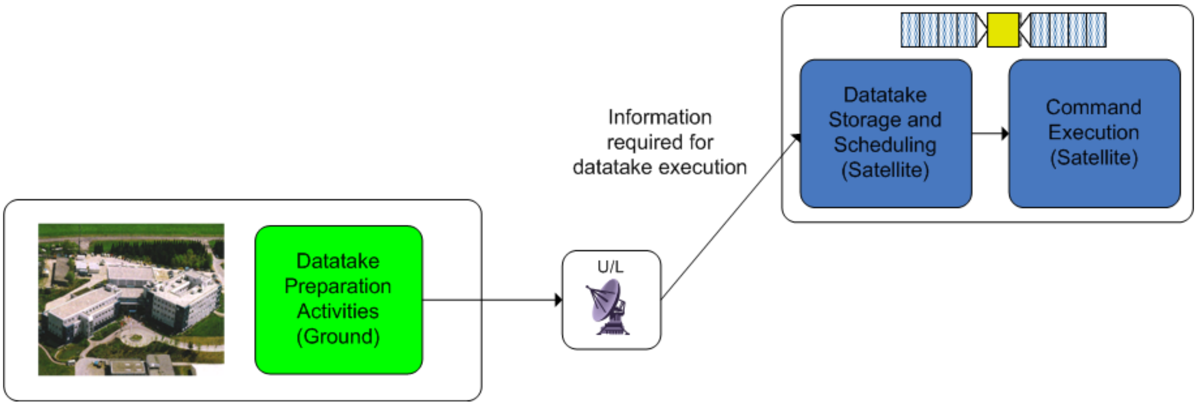


Figure 4.2.: Information flow when using on-*board* scheduling

Figure 4.1 shows the conventional ground-based approach, information for the datatake is collected in the control center and scheduled. A TC sequence is than generated out of the schedule. This sequence is up-linked and subsequently executed by the satellite. The process of datatake scheduling and command generation can be moved on-board the satellite as shown in Figure 4.2. This requires on-ground only preparation of all information required for datatake execution but no scheduling.

With this idea in mind the following questions are to be answered:

1. What possible configurations of satellite and sensor for on-board scheduling exist?
2. Which operational constraints must be considered by the scheduler?
3. What scheduling algorithm can be used?

Subsequent chapters discuss these questions not only with the focus on simulation but also with respect to questions of a real implementation.

4.1. Scheduling System Configurations

On-board scheduling relevant information must be made available to the scheduling software. Two basically different combinations of sensor(s)/satellite(s) are principle feasible and should be discussed in this chapter.

For the next thoughts the data acquisition phase of the sensor should be considered as ”a pass” over a certain target area. A ”pre-pass” phase is the time before image acquisition by the main sensor starts including preparation of commands, instrument and bus. The ”post-pass” phase is the time after acquisition end by the main sensor. Different constellations of on-board/on-ground scheduling as well as satellites and sensors on the satellites are possible and should be classified as constellation A, B, C and D.

The simple ”post-pass” case is shown in Figure 4.3. The satellite passes over the interesting target and acquires data by its sensor. Interesting target properties (e.g. clouds) can be identified after down-link on ground (constellation A, one satellite with one instrument; on-ground scheduling) or directly after acquisition in the data on-board the satellite (constellation B, one satellite with one instrument; on-board scheduling). A reaction of an on-board scheduler is possible ”post-pass” at the earliest in case of constellation B.

For ”pre-pass” target identification it is possible to use ”look-ahead” instruments as a real-time sensor - depicted in Figure 4.4 and named constellation C (one satellite with one look-ahead sensor and one main sensor). This instrument looks under a certain angle along-track into the direction of flight observing the target area to identify the interesting information (e.g. cloud coverage). Masaru describes this technology in [Mas94] as a patent. This technology is only useful for small targets (e.g. point). Long targets can’t be acquired at once by the look-ahead instrument. A further possibility is installation of this instrument on another satellite flying ahead of the imaging satellite - depicted in Figure 4.5 [Alg03] and named constellation D (two satellites with one sensor each).

The next question is: How can this be applied to the problems identified in Chapter 3.1 and Chapter 3.2? Can similarities be identified?

Clouds in images can be identified using classification algorithms as used for satellite image processing. Principle feasibility has already been shown on the satellite EO-1 [MMGB03].

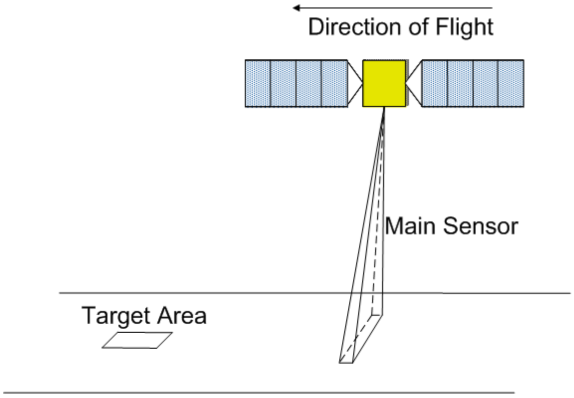


Figure 4.3.: Standard satellite/sensor configuration

*The satellite passes over the target and executes the scheduled acquisition. Classification of interesting target properties (e.g. clouds in the image, forest fires etc.) can only be done **after** the acquisition is finished.*

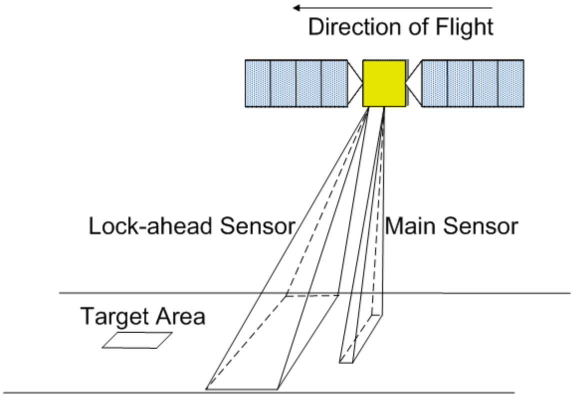


Figure 4.4.: Look-ahead sensor configuration for on-board scheduling

*The satellite passes over the target and executes the scheduled acquisition if it is relevant. Classification of relevant target properties (e.g. clouds in the image, forest fires etc.) can be done **before** the acquisition is finished.*

This technology can be used to classify clouds in the target area either pre-pass (data from look-ahead sensor) or post-pass (main sensor data is analyzed and deleted if cloud coverage threshold is exceeded). In case lossless data compression is applied any analysis of ”pre-pass”' sensor data won't work. To calculate the compression ratio only the main sensor data can be used as data from other sensors (low resolution look-ahead sensor or other satellite sensor) show other compression ratios.

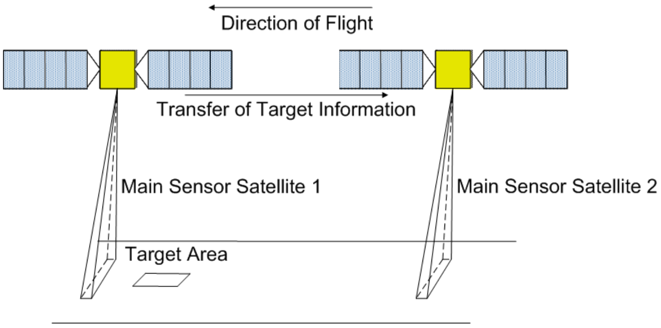


Figure 4.5.: Configuration for on-board scheduling with an additional satellite

*The first satellite passes over the target and identifies relevant target properties. Information is transfered to the second satellite equipped with the main sensor. Classification of relevant target properties can be done **before** the main acquisition is started.*

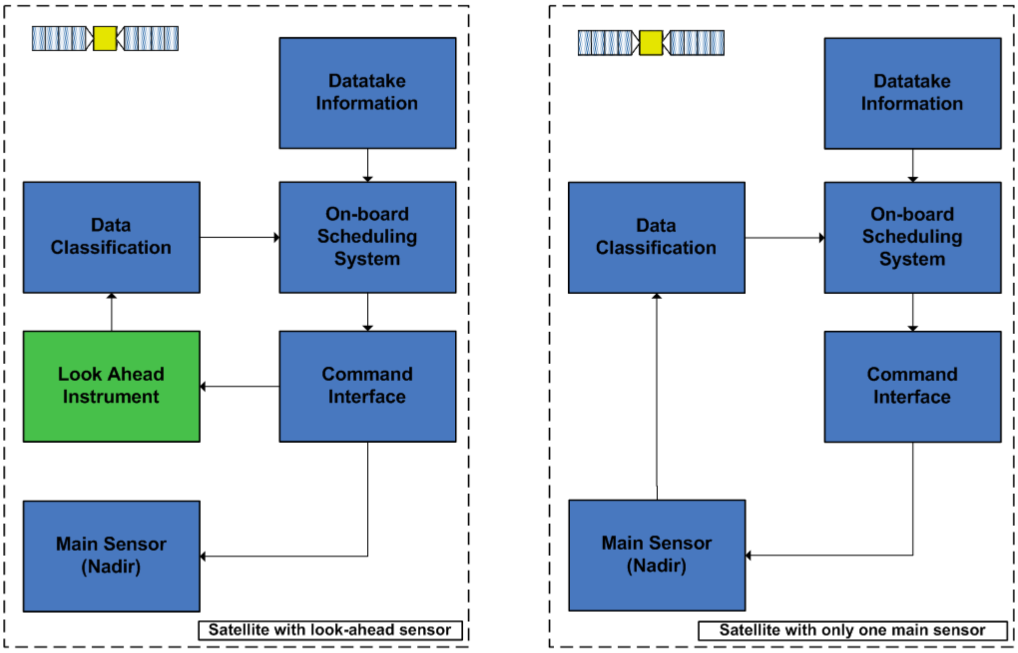


Figure 4.6.: On-board control chains with and without look-ahead sensor

> It can be concluded that all possible solutions are only different concerning the point in time when the information is available to the scheduling process. This may be either before execution of image acquisition (information from other satellites or look-ahead sensor) or after image acquisition (information from acquired data). *This drives the point in time when the first reaction to changed scheduling constraints is possible for an on-board scheduler.*

4.2. Boundary Conditions for Satellite Operations

After analysis of the possible sensor/satellite constellations emphasis is now placed on the scheduling relevant constraints that must be obeyed by the on-board scheduler.

Satellite operations are limited by different constraints which are implied by the satellite sub-systems and the network for communication. These boundary constraints must also be obeyed when developing an on-board scheduling system. The communication network limits the available data transfer time whereas the satellite itself limits the maximum payload operation time due to limited resources. According to [ZF94, p. 629] a resource may be a unit, tools, material or personnel which is used or consumed during the production process. Concerning the scheduling problem for satellite data takes resources are primarily power and memory. Other constraints may exist and should also briefly be discussed.

Memory - M

Satellite on-board mass memory is used to store data which is produced by the payload. Data storage is limited and must therefore be considered as a resource. This resource is consumed or produced (d) by two satellite activities. During a datatake by the payload mass memory resource is used for immediate data storage. Payload data is produced at a constant rate, given in e.g. Gbit/s (this assumption is verified by EO missions like e.g. LANDSAT or IRS with the sensor characteristic listed in [Kra02]). The used memory resource can be restored on two ways - either by deleting data or by dumping data to a ground station. Data may be deleted if considered not useful for dumping (e.g. due to cloud coverage).

Power - P

Power is consumed by all satellite subsystems, provided and produced by the electrical power subsystem. Power is therefore also a resource necessary for payload operations. Consumption depends on the activated subsystems. Active subsystems on the other hand depend on the current satellite mode. The consumption during a datatake is higher as the additional instrument operations consume additional power. Power generation depends on the size of the solar generator and the attitude towards the sun. The process of generation and consumption is therefore coupled by the necessary attitude towards the sun or towards the target.

Other Constraints

Beside power and memory other limitations for satellite payload operations may exist. Attitude sub-system limitations can limit the maximum number of roll or point maneuvers due to a given maximum slew rate or system desaturation requirements. The thermal sub-system can reduce maximum operation times due to heating of hardware parts. Furthermore complicated mechanism of mass memory management may also exist and set limitation. The discussion should not be detailed anymore as these limitation depend strongly on the selected satellite.

Orbit Usage - t_{OU}

The orbit usage is a value accumulating the different constraints like power, attitude system limitations or thermal sensor constraints to a single value that can easily be used by scheduling.

This value is provided as a maximum sensor operations time per orbit or maximum kilometer that can be acquired. It is based on simulations of the satellite manufacturer. This allows easier handling of constraints avoiding extensive modeling and calculation (e.g. thermal sensor behavior).

Derived Operations Equations

During one operations cycle an equilibrium valid between data generation $t_{OU}R_S$, immediate storage M and down-link $t_{DL}R_{DL}$ must exist. With the resource limitations in mind the requirement for meaningful scheduling of systems with completely known scheduling constraints can be than written as (valid for on-ground scheduling and on-board scheduling if planning relevant information is available before data acquisition):

$$t_{OU}R_S \leq M = t_{DL}R_{DL} \tag{4.1}$$

with the maximum operations time (orbit usage) depending on the satellite and the available down-link time:

$$t_{OU} = f(P, T, ...) \qquad t_{DL} = f(orbit, groundstation) \tag{4.2}$$

Equation 4.1 is not valid anymore if a satellite with post-pass on-board scheduling is assumed (constellation B). For the EO satellite operations relevant problems we must consider cloud coverage or lossless data compression in this equation. For the cloud coverage only problem we add p indicating the useful fraction of the acquired data:

$$t_{OU}R_{S}p \leq M = t_{DL}R_{DL} \quad (0 \leq p \leq 1) \tag{4.3}$$

In case only lossless data compression (see Ch. 3.2) is used a division by the achieved compression ratio cr of all stored data is required:

$$\frac{t_{OU}R_{S}}{cr} \leq M = t_{DL}R_{DL} \quad (1 \leq cr) \tag{4.4}$$

Size of mass memory is a known value for each EO satellite. Available average data D/L can be calculated for a given ground station location, data transmission rate and orbit. Payload source data rates are constant and depend only on the operations mode. The maximum possible operations time of the payload depends on the sensor. Values p and cr are unknown and can only be handled by an on-board scheduling process. Furthermore it allows for increased sensor operations time without an increase in M and $t_{DL}R_{DL}$ subsequently.

Table 4.1.: Main characteristics of different scheduling constellation

Constellation	(see Ch. 4.1)	A	B	C	D
Scheduling		On-Ground	On-Board	On-Board	On-Board
Handling of	Cloud Coverage		x	(x)	x
Handling of	Lossless Compression		x		
Requirement	$t_{OU}R_{S}$	$= M$	$\geq M$	$= M$	$= M$

Table 4.1 lists the different possible constellations mentioned. An x in the table marks the EO scheduling problem that can in principle be handled by on-board scheduling. The (x) means: possible in case that all interesting area is captured by the lock-ahead sensor before target comes into main sensors field of view (usually not valid for push-broom scanner). $t_{OU}R_{S}$ is $\geq M$ for constellation B as the unknown values of either p or cr require it to be greater accordingly.

4.3. Integration of an On-Board Scheduling System

This section contains practical considerations for integration of an on-board scheduling system. Integration on-board the satellite is discussed as well as required separation into different software processes and types of data transfered. Details are satellite dependent; the discussion in this section is therefore kept on a more general level, valid for all satellites.

Integration of the Software

Satellite software is separated into the OS and application software [LW99]. The OS is usually a Real Time Operating System (RTOS) according to the software execution requirements in the satellite domain. The RTOS manages on-board computer resources and execution of application software.

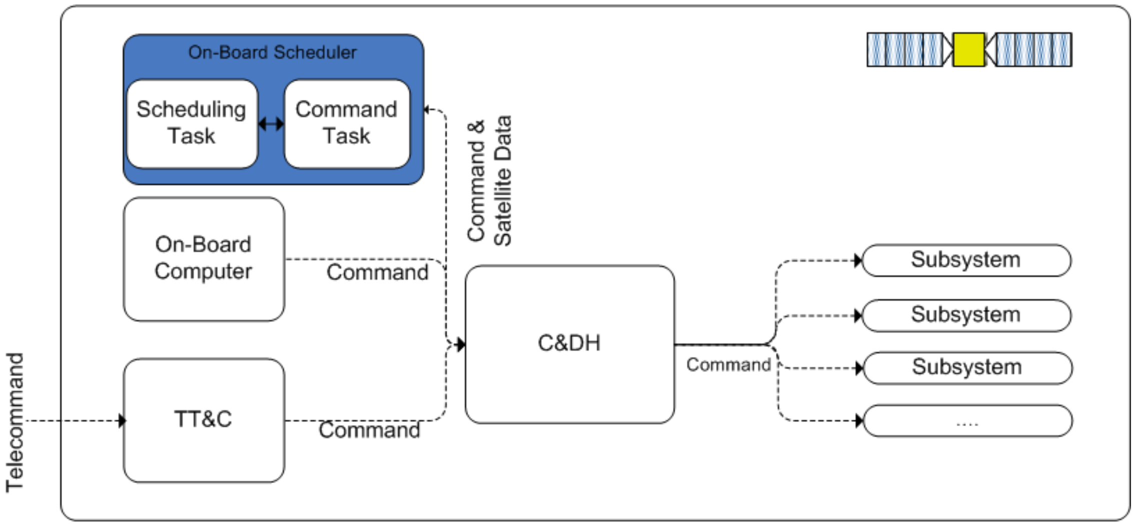


Figure 4.7.: Spacecraft command distribution with integrated on-board scheduler

The RTOS basic block of software is a task [Sim99]. Each software task is assigned a status and a priority by the OS scheduler. The scheduler of the OS schedules the software tasks in a certain order depending on status and priority. Task status can be running, ready or blocked. Together with the task priority the RTOS schedules the applications so that only one task is running at a time and hereby defining the processing time available to each task.

The basic idea may be the design and integration of the on-board scheduling software as a single software task. This software task would require a very high execution priority in order to ensure in time execution of time-tagged commands which are issued by the scheduler. Furthermore any scheduling activity of this task may block in time command execution. This leads to a recommended design using a least two different tasks with different priority (a high priority task taking care of command execution and a low priority task taking care of the scheduling). With the interface and task requirements in mind, Figure 2.3 can be extended to Figure 4.7 depicting the additional two tasks for the scheduler.

This task separation allows one task keeping all the commands issued by the scheduler, while the other task generates new schedules and prepares commands. If a newer schedule is finished the commands kept by the commanding task can be deleted and new commands can be send from the scheduling task.

Software Interfaces and Data Transfer

Which information must be transferred between the scheduler and the satellite?

1. Commands from ground to control the scheduler
2. Scheduling and execution relevant information for each datatake
3. Information of the satellite state (e.g. quality of the acquired data)

(1) Commands to control the scheduler are similar to other on-board software control commands. Those commands are used to start, stop and configure the software.

(2) All DT scheduling relevant information must be provided to the on-board system. The required information for an optical sensor is listed in Table 4.2. The position of the satellite is known by orbit prediction as long there are no orbit maneuvers. Datatakes can therefore be described by start time and end time (position of the satellite) and a necessary roll angle within the field of view. For a differentiation between different important DT a priority is associated with each DT.

Table 4.2.: Additional data-take information required for on-board scheduling

Datatake Information	Description
Datatake Start	Start time of the datatake
Datatake End	End time of the datatake
Target Information	E.g. pointing information or roll angle information
Priority	Priority of the datatake

Execution relevant information is the command information needed by the scheduler to issue the right command and configuration setting for the satellite sensor operation. As an example: the DLR BIRD satellite mission used atomic single commands like

```
(TT;20:06:08:26:10:29:50;PDH_M_PSTART_MEA,6,0,0,0,0)
```

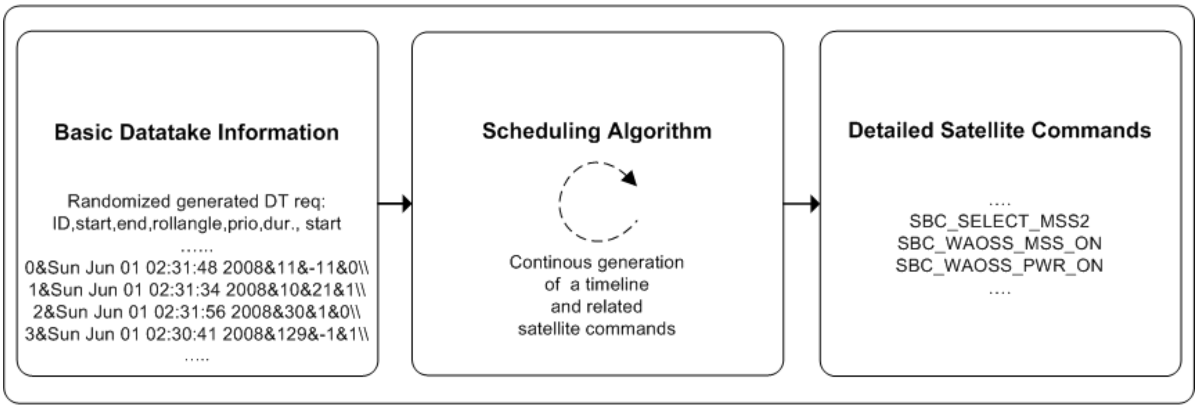
switching the instrument configuration and starting the instrument for measurement - in this example already together with the execution time stamp. Execution and scheduling information is stored together by in the scheduler.

(3) The scheduling relevant information gathered during DT execution (like state of the satellite, results of acquisitions, data from a look-ahead instrument...) must be made available to the scheduler by a dedicated interface.

4.4. Development of a Scheduling Algorithm

Overview

This chapter deals with the problem of selecting and implementing an on-board scheduling algorithm. The algorithm is responsible for scheduling of datatakes based on the information provided from the ground control center (the basic DT information). The result of the scheduling process is a DT sequence (timeline) which is translated into detailed commands and sent to the satellite (see Figure 4.8) for execution.



```
......
0&Sun Jun 01 02:31:48 2008&11&-11&0\\
1&Sun Jun 01 02:31:34 2008&10&21&1\\
2&Sun Jun 01 02:31:56 2008&30&1&0\\
3&Sun Jun 01 02:30:41 2008&129&-1&1\\
.....
```

```
....
SBC_SELECT_MSS2
SBC_WAOSS_MSS_ON
SBC_WAOSS_PWR_ON
....
```

Figure 4.8.: Basic input and output of the on-board scheduling algorithm

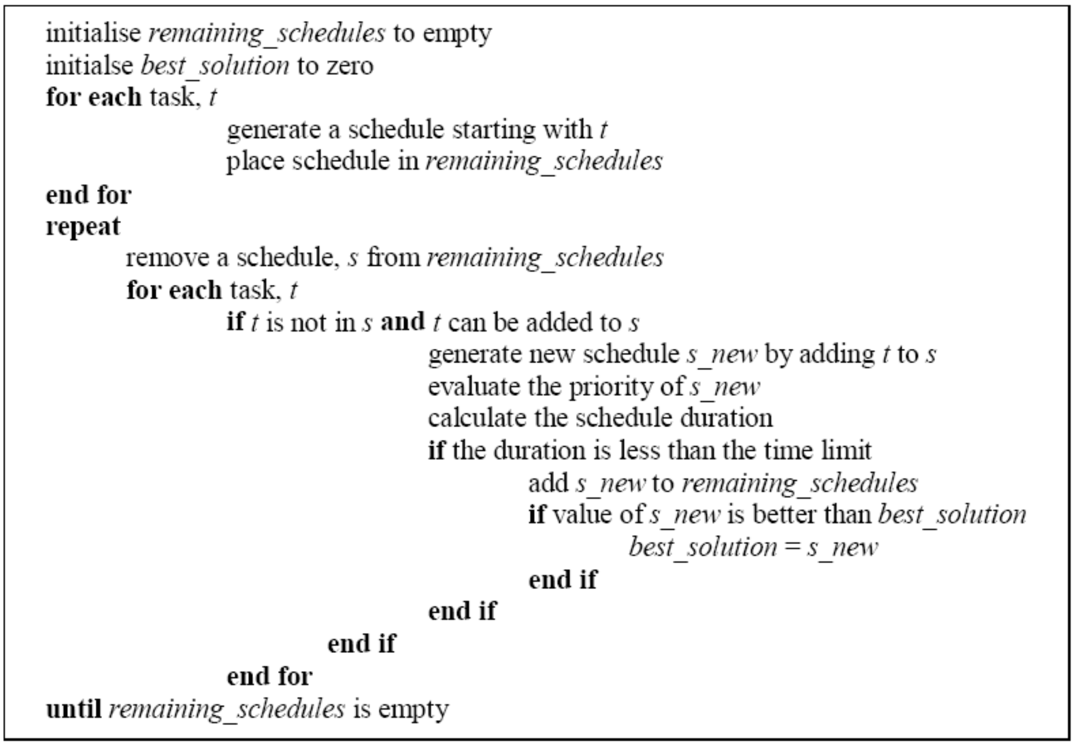
In literature an algorithm is defined to be a set of computational steps that transforms input into output [LRS01] or an algorithm is a set of well-defined steps required to accomplish some task [HR06]. The required properties of computer implementable algorithms vary in literature, but common are:

- the description of the algorithm is finite (finiteness)
- the next step is defined at each time (definiteness)
- with the same input the algorithm produces the same output (determinism)
- each step can be executed (executable)
- the memory space required during execution is finite (dynamic finiteness)

The properties finiteness, dynamic finiteness and executability are required for each algorithm. Finiteness and dynamic finiteness are especially relevant for on-board scheduling application where the requirement is performance with small overall program (algorithm) size and memory consumption. Definiteness is a property implicitly given by programming the scheduling algorithm as spacecraft board computer executable code. If no random values are used by the algorithm for the decision about the next step for execution, determinism is given. In this case the same input produces the same output (nevertheless, if the input varies by chance the output is also different).

Satellite Scheduling Algorithms in Literature

For the development of an on-board scheduling algorithm it is interesting which types of algorithms have been used for scheduling by other EO satellite missions or have been studied for this purpose. Two algorithms should be depicted as an example description. This is done by use of pseudo-code as mostly used in scientific literature (although the use of natural language or programming language would also be possible). Such a description is independent of a particular (computer) implementation language.



```
initialise remaining_schedules to empty
initialse best_solution to zero
for each task, t
        generate a schedule starting with t
        place schedule in remaining_schedules
end for
repeat
    remove a schedule, s from remaining_schedules
    for each task, t
        if t is not in s and t can be added to s
                generate new schedule s_new by adding t to s
                evaluate the priority of s_new
                calculate the schedule duration
                if the duration is less than the time limit
                    add s_new to remaining_schedules
                    if value of s_new is better than best_solution
                        best_solution = s_new
                    end if
                end if
            end if
        end for
until remaining_schedules is empty
```

Figure 4.9.: Partial enumeration algorithm as described by [HP99]

The class of greedy algorithms is often used in the satellite scheduling domain. [PG98] uses a greedy scheduler for LANDSAT 7 acquisition scheduling. The GSOC ground-based Mission Planning System also uses a greedy approach for acquisition scheduling (e.g. TerraSAR or EnMAP in the future). A modified greedy algorithm has been used by [FJM01] for EO satellite scheduling in general. An example algorithm description can be found in Figure 4.10.

Greedy algorithms go through the computation steps by making the locally optimal choice (greedy choice property) with the expectation of making a globally optimal choice [LRS01].

This selection strategy is a called heuristic. Resulting global solutions are not guaranteed to be optimal - but mostly solutions close to the optimum are produced.

For the SPOT-5 satellite branch-and-bound, russian doll search, greedy and tabu search algorithms have been analyzed in [BVA$^+$96]. Greedy and tabu search algorithm are approximate as they don't produce necessarily optimal solutions whereas brand-and-bound and russian doll search are exact methods.

Partial enumeration methods as depicted in Figure 4.9 have been discussed in [HP99] and genetic algorithms have been analyzed in [GCLP02]. For image acquisition scheduling of fleets of EO satellite also greedy algorithms are proposed by [J$^+$02].

A variety of different other algorithms is discussed in literature for satellite scheduling (e.g. in [GNT04]). The ones mentioned can be classified by the on-board scheduling relevant characteristics:

- they are working iterative
- they produce exact or approximate solutions
- results are deterministic

```
procedure HBSS($Obs$)
  $P = \emptyset$
  while $Obs \neq \emptyset$
    $o$ = SelectObs($Obs$)
    $t$ = SelectTime($o$)
    $P = P \cup o$ starting at time $t$
    $Obs = Obs - o$
    $P$ = Propagate($P$)
    if no plan found return $\emptyset$
  end while
  FindPlan($P$)
  return $P$
end
```

Figure 4.10.: The steps of the HBSS Algorithm (taken from [FJM01])

Acquisition scheduling for existing EO satellites is often done by the use of greedy algorithms based on heuristic selection strategies. Other algorithms are mostly investigated in literature to see the theoretic potential but are not used in operations due to their time consuming search for a solution.

Algorithm Selection and Implementation

Any algorithm suggested for on-board planning is bound to the limits of implementation and execution on a spacecraft board computer, which are:

- significant limited on-board processing power (compared to usual office computers)
- limited mass memory for storage of the algorithm and the functionality around (ROM)
- limited temporal memory during program execution (RAM)

Limited processing power together with an on-board scheduling system in a configuration as discussed in Chapter 4.1 require an algorithm producing the scheduling result as fast as possible.

Otherwise the next acquisition area may come in sight for the satellite instrument without the scheduler being finished with timeline generation. Limited ROM require the description of the algorithm to be finite whereas limited RAM forbids any algorithm working recursive (the algorithm should not be allowed to make a self-reference, this makes the memory consumption unpredictable).

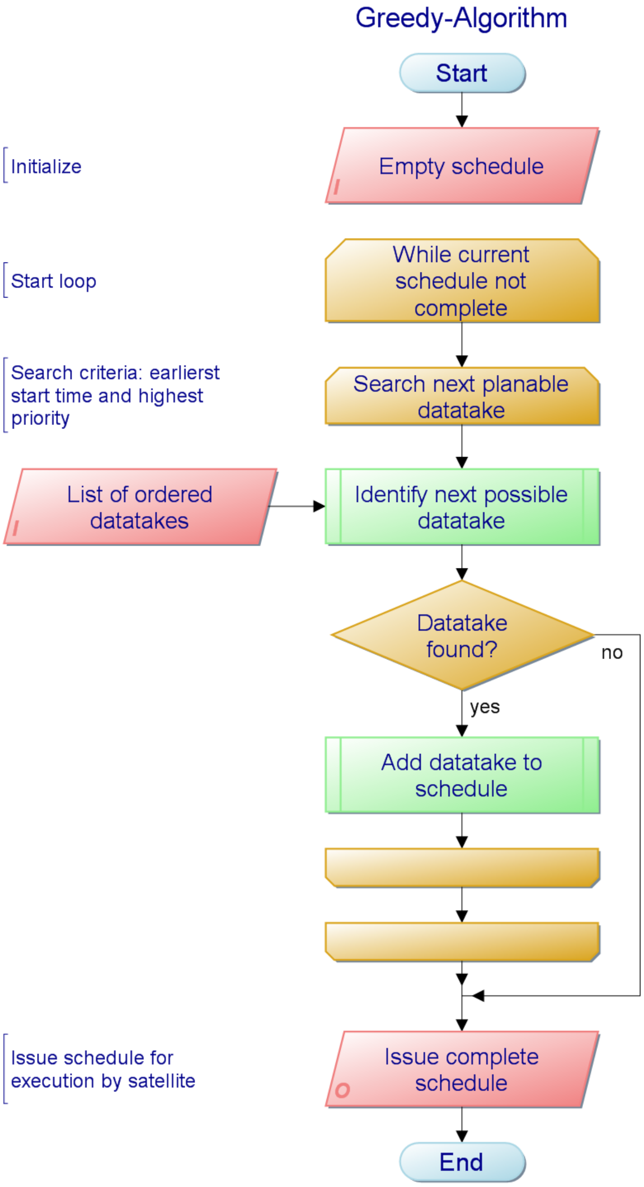


Figure 4.11.: Flowchart of the on-board scheduling algorithm according to [fNeV83]

In [BVA+96] different algorithms are tested on identical problems and the required CPU usage as well as the quality of the scheduling solution are calculated. The clear statement is that the greedy scheduler is the fastest algorithm (up to two orders of magnitude compared to other), producing mostly good solutions. Also [K^+03] concludes that a greedy scheduler is the best solution for an on-board scheduling system.

For the above implementation and research reasons a greedy algorithm has been selected for on-board scheduling investigation. The algorithm response is fast and the required memory consumption (RAM and ROM) is already now during implementation.

The implementation is be based on the existing GSOC MPS scheduling algorithm with an adapted greedy choice property in step 2. The greedy scheduling algorithm consists of the following basic steps:

1. initialization of an empty schedule
2. identification of the next possible DT for scheduling using the ”greedy choice property”:
    - 1. priority
    - 2. starting time
3. check if the found DT can be added to the schedule (based on available resources)
4. stops if the schedule is full or no more resources are available, indicated by the *orbit usage*

The algorithm uses a heuristic to select the next possible DT with the highest priority and the first start time after the end of the current timeline, an approach that is usually selected for EO missions [FJM01].

Algorithm description is done in a flow chart in Figure 4.11 according to norm [fNeV83]. The final schedule generated by this algorithm is feasible if no two jobs overlap, operations time is not exceeded and mass memory only used to the maximum limit. The detailed implementation of the used algorithm can be found as C programming code in Annex A.2.

If implemented in a satellite SBC the scheduler works as shown in Figure 4.6 on the right hand side: During a ground station contact the prepared imaging requests are up-linked to the satellite and stored. With this information the scheduler generates a timeline. This timeline is executed and subsequently send as executable commands to the SBC. An interface between the spacecraft board computer and the scheduling process sends information on the cloud coverage of the acquired images and initiates new scheduling runs if necessary.

Greedy Algorithm Discussion

Is the selected greedy algorithm appropriate for scheduling of image acquisitions? What are advantages and drawbacks of the selected implementation? With a look to the implementation it is clear that the required algorithm properties (static and dynamic finite, executable, deterministic, defined) are given. Furthermore it can be classified as an algorithm working iterative, generating approximate solutions based on a heuristic (greedy choice property). The heuristic is led by selection of acquisitions following the rule highest priority and early start. So implementation and use on a satellite is principle feasible.

Advantages
The greedy scheduler has the lowest computation requirements of all algorithm discussed, shown by [BVA$^+$96] in comparison to tabu-search, branch-and-bound and russian-doll-search. The fast response time for schedule generation is the biggest advantage. Time needed for schedule generation by the greedy scheduler grows only linear with the number of selectable datatakes n. Other algorithms which are searching the whole number of potential schedules (combinations of the different possible image acquisitions / DT into a timeline) suffer from the complexity grow by $n!$ theoretical different schedules.

Remark: The exact computation effort and time required for timeline generation by the implemented algorithm can only be quantified if implemented on satellite specific SBC hardware. The comparison of the required processing power is therefore relative left open in the framework of this discussion.

Drawbacks
A clear drawback is that not in each case the optimal schedule is produced. This argument could be countered by stating that it is better to have a new schedule ready for execution before the satellite misses the next potential target. A genetic algorithm for example could generate an optimal solution if no time limit for schedule generation is set.

Another drawback is that this on-board scheduler works only for a certain satellite type (one satellite with an slew-able optical instrument). For other satellite or instrument types modifications of the algorithm may be necessary. Nevertheless this drawback applies also to all of the discussed algorithms.

No Use of Scheduling Algorithm?
Finally the question should be if it is better not to use any scheduling at all. An option would be to simply select the next possible DT and execute it. But even an implementation of this logic is already an algorithm with a simple selection heuristic. Furthermore this would no allow to use a priorization of the different DT as required for satellite operations (see Chapter 5.3). Another problem that the overall schedule needs to be feasible (no overlaps between DT, no overbooking of satellite resources), a property that is not checked if ”no scheduling algorithm” is used.

4.5. Resource Usage Optimum

A schedule is optimal if it minimizes or maximizes a certain criteria [Bru07]. As mentioned in the chapter above, greedy algorithms produce not in every case the optimal schedule solution. The scheduling goal for the algorithm is a maximum return on (useful) imaging data. In case of cloud covered images a minimum of cloud coverage in down-linked data is the goal.

The optimum is 100% of cloud free down-linked images while using the satellite resources to the maximum extend possible. Satellite resources are used in every scenario in Chapter 5 up to 100%, as the limit for EO satellite operations is usually the satellite but not number of potential imaging areas. The resulting efficiency is calculated as the relation between cloud free and cloud covered area in the image (value A_{cc} & A).

5. Simulation of Mission Scheduling With Cloud Coverage

As a case study on-board scheduling is simulated for an example EO satellite mission. The selected problem is an optical earth observation satellite with one optical sensor and the on-board scheduling relevant problem of cloud coverage. It will be shown that on-board scheduling is advantageous compared to the conventional ground based scheduling.

First the scheduling relevant parameters of the mission are characterized based on a real mission. This is followed by selection of the required data for cloud coverage calculation. In the subsequent chapters a user request scenario is generated and used for simulation purposes. Finally the setup of simulation software and results are shown.

5.1. EO Mission Simulation Relevant Parameters

For simulation purposes example EO satellite mission parameters are used. All required parameters are selected similar to the EnMAP (planned launch 2012) EO mission, see [SKH^{+}07] and [Fer08a], with the aim of a selection based on a real mission. The general mission parameters and the planning relevant parameters are known, as GSOC is responsible for preparation and operation of this mission. A sun synchronous orbit is used for this mission and characterized by the following TLE [1]:

1	99999U	99999ZZZ	11	1.95833333	0.00000000	00000-0	000000+0	0	02
2	99999	97.9606	86.2414	0000001	0.0000	0.0000	14.74527579		01

EnMAP is equipped with one optical nadir looking instrument. This is a constellation like shown in Figure 4.3 and classified in Table 4.1 to be of type B. Assumption for the simulation: the satellite is equipped with on-board scheduling based on the greedy algorithm of the preceding chapter and the possibility for cloud coverage classification of the acquired data.

For commanding and reception of telemetry the Weilheim ground station (located at 47°52‘ N; 11°05‘ E) in southern Germany is used. The satellite can point the instrument by rolling

---

[1] Explanation of the Two Line Element orbit description can be found in [VCHK06]

the whole bus together with the instrument within ± 30°. The opening angle of the optical instrument is 2.5°. Therefore all points within a swath of ± 31.25° can potentially be covered by the satellite sensor.

Due to it's passive optical sensor the satellite is only able to acquire images during daylight time. All images are stored in an on-board mass memory of 512 Gbit. Relevant satellite characteristics are concluded in Table 5.1.

Table 5.1.: Mission parameters used for simulation

Parameter		Value
Satellite		
	max. swath (roll)	± 31.25°
Instrument		
	instrument opening angle	2.5°
	instrument source datarate R_S	0.822 Gbit/s
Memory		
	on-board mass memory M	512 Gbit
Downlink		
	downlink datarate R_{DL}	300 Gbit/s

The mission planning relevant parameters are given in Table 5.2. Minimum and maximum possible datatake length are given in km (although a value in seconds would usually be expected). Using the mean motion that is given by the used TLE

$$n[rad/s] = n[rev/d] \cdot \frac{\pi}{43200} \tag{5.1}$$

and

$$\alpha = \frac{180 \cdot b}{R_{\oplus} \pi} \tag{5.2}$$

the mean motion can be calculated to be ≈ 0.061439 °/s. The resulting min. and max. operation times per datatake based on the set requirements can be calculated using Equation 5.2 and are listed in Table 5.2. While min. and max. values are required for the generation of the user scenario the accumulated maximum values per orbit or day are required for the scheduling. This is the upper limit of DT that can be scheduled and executed.

Figure 5.1 shows the satellite orbits over a duration of 24 hours with bold lines during ground station contacts. During several orbits no ground station contact is possible (the contact gap may be up to 9 hours or even more).

Table 5.2.: Planning specific parameters used for simulation

Parameter		Value	
Single datatake length			
	min.	30 km	$\approx 5\,s$
	max.	1000 km	$\approx 146\,s$
Accumulated maximum values			
	max. in one orbit	1000 km	$\approx 120\,GBit$
	max. per day	5000 km	$\approx 600\,GBit$
t_{OU}			
	max.	146 s/orbit	
AOCS positioning time			
	max.	20 s	

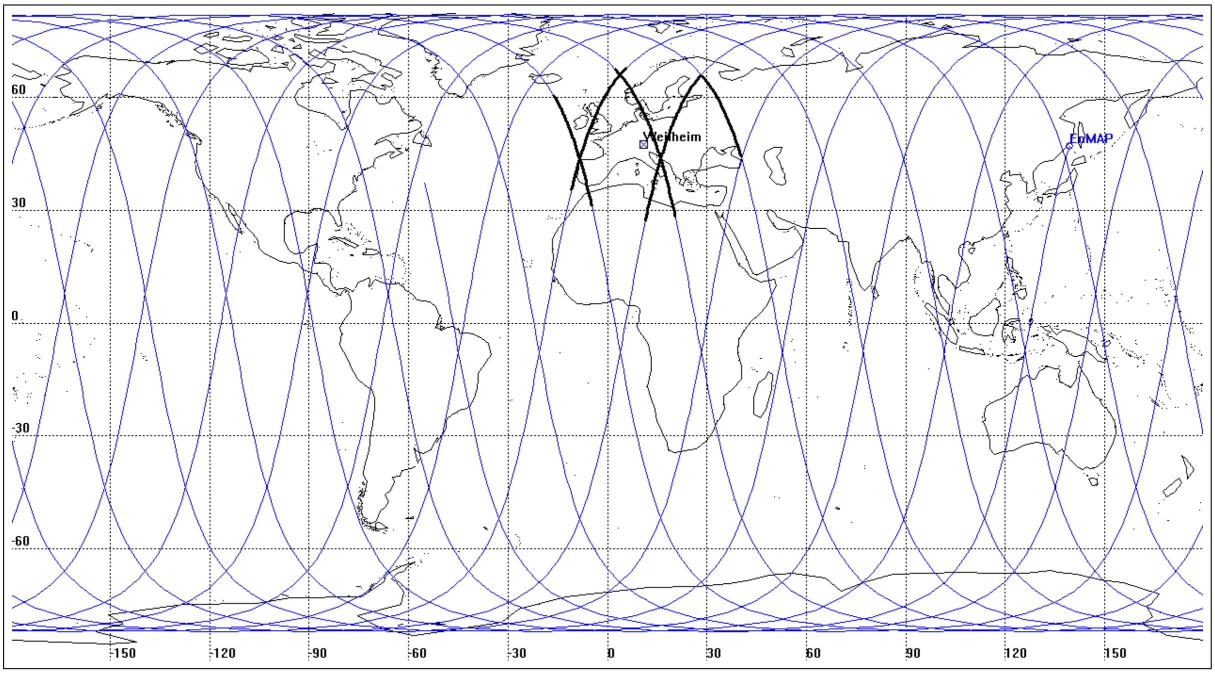


Figure 5.1.: 24 hour orbit of EnMAP satellite

5.2. Cloud Coverage Data for Simulation

Cloud Data Sources

For the simulation of cloud coverage on optical earth observation images sufficient data must be used by the simulation system. Different sources exist for this but differ in usability for simulation purposes. Cloud coverage data can principally be derived from numerical calculation or observation and the following sources are accessed:

- weather satellite images
- SYNOP observations
- numerical models

Simulation data usability depends on the aspects:

- continuous temporal availability
- spatial coverage of the area that can be covered by the simulated satellite
- good spatial resolution of data and numerical access to data by simulation software

The first obvious data source are images of weather satellites as for example provided by NOAA or GOES. Different orbits are used by those satellites, usually polar orbits for a good coverage of the whole Earth and sometimes geostationary orbits. Satellites in polar orbits have no possibilities for a continuous coverage of the same area on ground. Satellites in geostationary orbits cover only the same part of the surface (a disc - see example in Figure 5.2[2]). Within regular intervals images of the Earth are made and send to the ground.

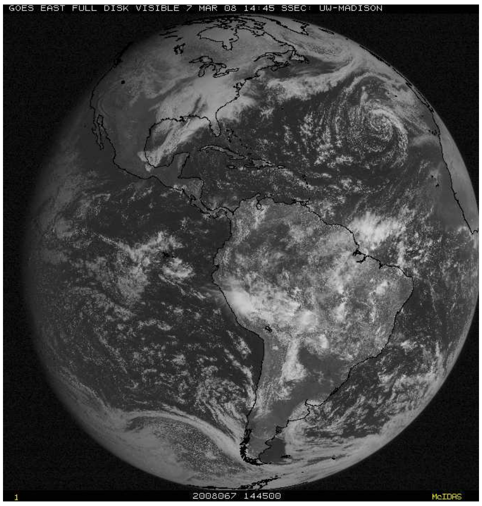


Figure 5.2.: GOES image

Depending on the wavelength of the weather satellite payload clouds can be identified in imagery by visual inspection. A problem is to deviate snow on ground from clouds in the atmosphere. This requires automatic processing with snow/cloud discrimination or manual analysis. Furthermore a geometric correction must be made as most of the images are not nadir images. Also a matching of pixels to coordinates on ground is necessary.

---

[2] Source: http://www.ssec.wisc.edu/data/geo/

A second source for cloud data are observations from ground as made by SYNOP[3] stations. Stations are manned or automatic and generate weather information on a continuous basis. But station distribution on the Earth is irregular. European areas are covered by a dense station grid while on other continents only a loose grid of stations is installed. Furthermore data is generated on irregular intervals.

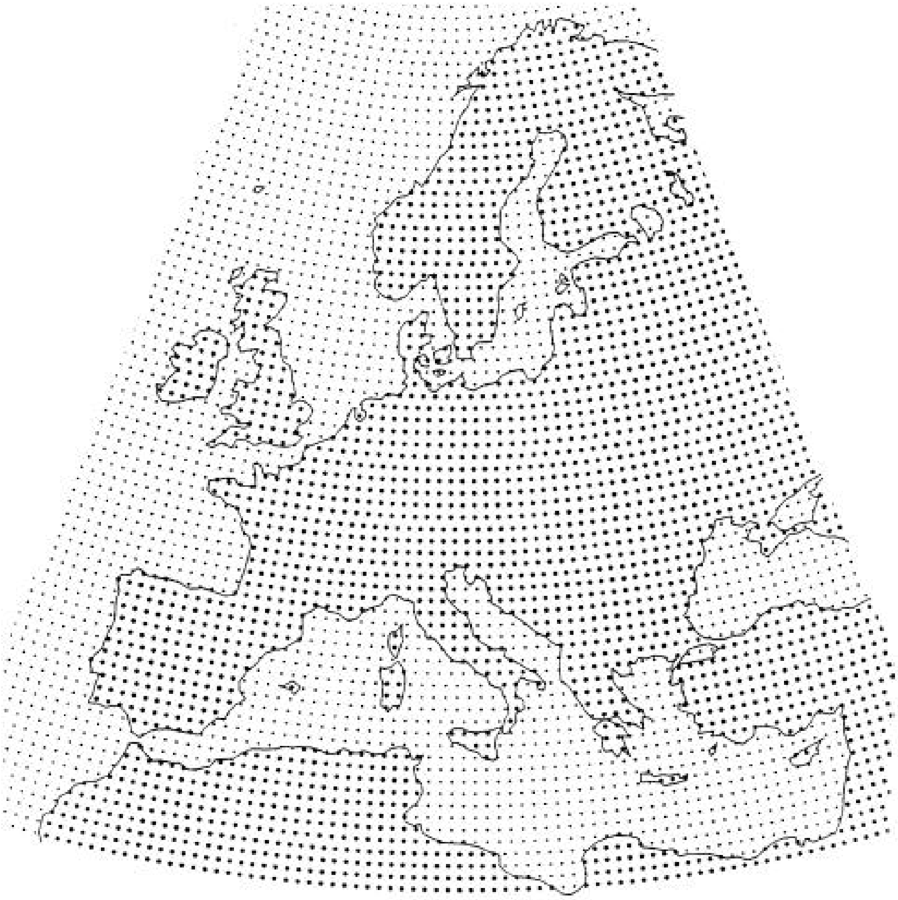


Figure 5.3.: Germany with 0.5 grid overlay as used by global cloud coverage models, e.g. the ECMWF model

The third possible source are numerical weather models. Those models take the current weather situation as observed and calculations of the last model run to generate a new model. The models cover the whole earth with a grid of model-depended density and different forecasts durations. Example providers in Europe are DWD[4] or ECMWF[5]. Data of those models is provided in a numerical GRIB coded form (packed) and can be decoded with specific software to its original values.

---

[3] SYNOP - SYNOPtic Observation

[4] Deutscher Wetter Dienst, http://www.dwd.de

[5] European Centre for Medium-Range Weather Forecasts, http://www.ecmwf.int

Figure 5.4.: ECMWF model - cloud coverage world wide

Based on the fact of a temporal and spatial continuous grid provided and the relatively easy integration into simulation software (no geometric correction or assimilation of different data sources necessary) it can be concluded that numerical weather models are the best source of cloud coverage data for an EO mission simulation (see Figure 5.3). Figure 5.4 is an example image of the ECMWF global model[6].

Cloud Coverage Calculation

The area acquired by the sensor and the cloud coverage must by calculated during simulation. Intersection of the sensor swath with the surface of the Earth gives a nearly rectangular shape marked by four corners points (see Figure 5.5). Position and distance of the corner points depends on the roll angle of the satellite, the duration of acquisition and the sensor opening angle. For the given corner points area and fraction of cloud coverage are to be calculated for simulation. Figure 5.5 shows the covered area of 10 seconds nadir data acquisition by a satellite within a sun-synchronous orbit (black line depicts satellite ground track).

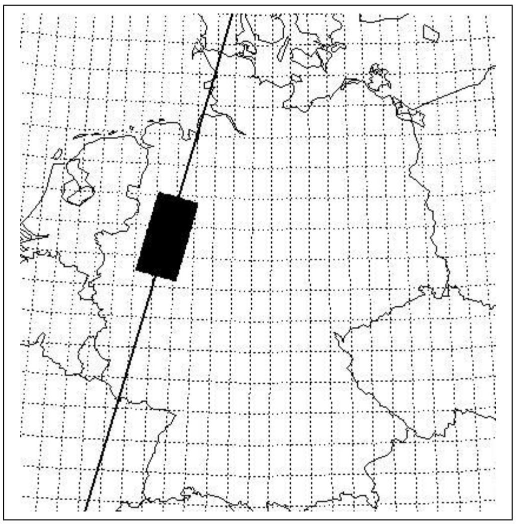


Figure 5.5.: Example data acquisition over Germany

The area can be calculated by splitting the datatake area during each simulation time step in single areas (based on the sensor FOV) as shown by Figure 5.6. The area is defined by the four corner points each with latitude (ϕ) and longitude (λ).

Covered area of each simulation step can be calculated by summation of small squares (symbolized by the dots, left side of Figure 5.6) based on a fine grid using Equation 5.5. The area of each square is calculated using Equation 5.3 (great-circle distance between two points (P1 and P2) on the earth, given by [Bar98]) and Equations 5.4 for the edge length and the derived area.

$$cos\ e = sin\phi_1\ sin\phi_2\ +\ cos\phi_1\ cos\phi_2\ cos\Delta\lambda \tag{5.3}$$

$$A_{(\phi,\lambda)_n} = d^2 \qquad\qquad d = R_{\oplus} \cdot e \tag{5.4}$$

---

[6] Example image generated from the ECMWF cloud forecast data: Data from January 1, 1992 00h. Image rotated anti-clockwise by 90°. Blurring results from data mapping to a grid with 0.5° steps in longitude and latitude directions (grid covers a smaller area at the poles compared to equatorial regions).

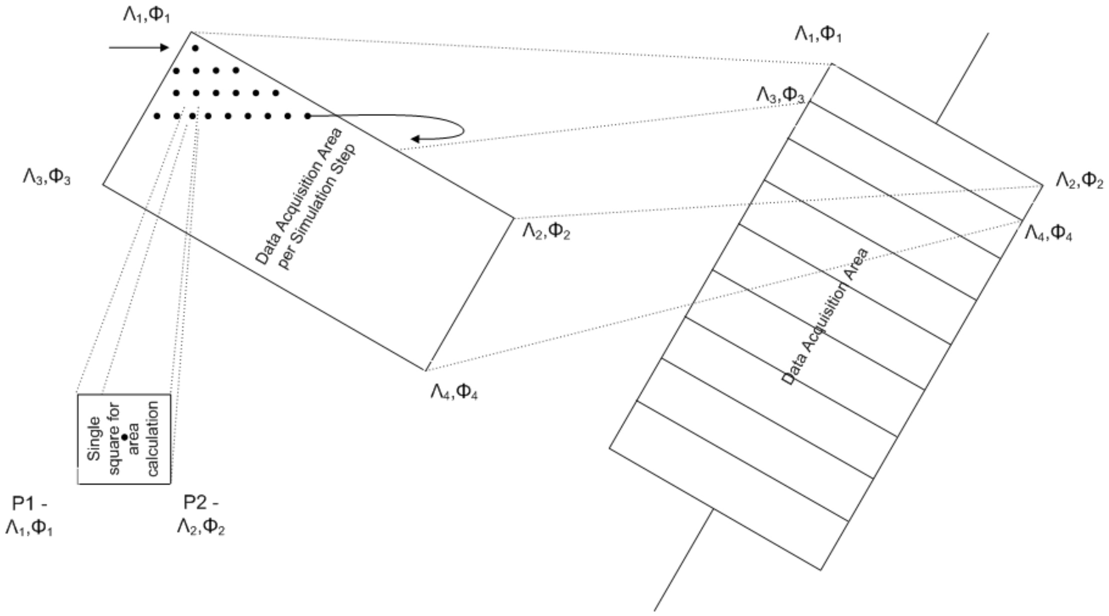


Figure 5.6.: Integration of area - each datatake can be split into smaller rectangular sub-areas based on the sensor FOV per simulation step

$$A = \sum_{1}^{n} A_{(\phi,\lambda)_n} \qquad A_{CC} = \sum_{1}^{n} A_{(\phi,\lambda)_n} \cdot p_{(\phi,\lambda)_n} \tag{5.5}$$

For each rectangular area cloud coverage is assumed to be constant. With the data of the numerical cloud model a bi-linear interpolation (see Equation 5.6) of cloud coverage is done for each grid point using the four neighboring points of the cloud model. First, the interpolation is done with the each pair of points along a parallel of latitude and the results are interpolated along a parallel of longitude. If the required data is not available at that point in time, an additional linear interpolation between the two neighboring points in time is made.

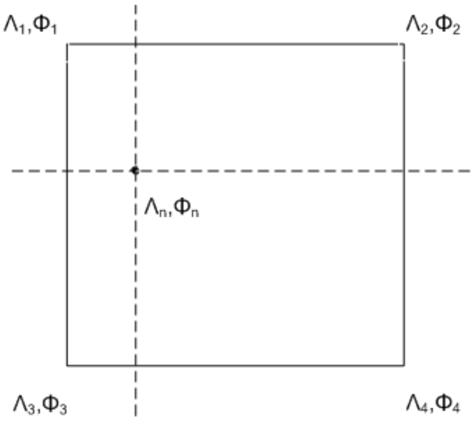


Figure 5.7.: Bi-linear interpolation of cloud coverage

ECMWF provided cloud coverage data for the years 1992 - 1996 ([fMRWF08]) with global coverage has been used for simulation. This data is freely available and coded in FM-92 VIII GRIB Ed. 1. The globe is covered by a grid of 0.5 degree. For each grid point four values are stored per day, averaged from data of the observation period (hour: 0, 6, 12, 18).

The GRIB format has been developed by the World Meteorological Organisation (WMO) as a standard to exchange meteorological data. It is used to exchange historical and forecast data in a standard format. Advantages are packing of data and fast transmission of big volumes of data. Detailed information can be found on the WMO website and in the WMO Manual of Code [Org94].

$$p_{(\phi,\lambda)_n} = p_{(\phi,\lambda)_1} + \frac{p_{(\phi,\lambda)_2} - p_{(\phi,\lambda)_1}}{\lambda_2 - \lambda_1}(\lambda_n - \lambda_1) \tag{5.6}$$

Three different versions of the GRIB exist. GRIB 0 is the initial version and no longer in use. Version 1 is still in use but the definition of the format has been frozen and no more changes will occur. The most current version is GRIB 2.

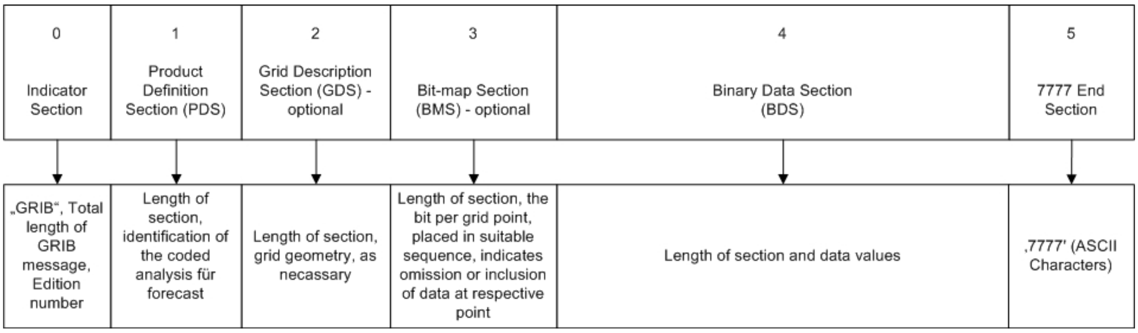


Figure 5.8.: Structure of a GRIB-1 file

The structure of a GRIB 1 file is shown in Figure 5.8. This binary structure is not directly readable in the packed format. It is therefore necessary to decode the raw data according to the settings of the GRIB message (see Appendix A.5). Decoding can be done using the basic Equation 5.7 provided by WMO Manual of Code [Org94] for GRIB-1 messages and the information of the ECMWF GRIB message header (R is not the datarate in this equation but a constant that is provided by the GRIB dataset).

$$Y \times 10^D = R + (X \times 2^E) \tag{5.7}$$

With the given values (see values in 5.8) Equation 5.7 simplifies to Equation 5.9. This formula can be used for direct reading of data from ECMWF dataset. Y is the original cloud coverage value and X given as a two octet number in the dataset.

$$D = 0 \quad E = -15 \quad R = 0 \tag{5.8}$$

$$Y = X \times 2^{-15} \tag{5.9}$$

Based on the sensor coverage of the Earth and the given cloud coverage data a calculation of the cloud coverage impact can be made. For a test area the integrations results have been compared between the selected solution and the commercial STK (satellite tool kit) tool.

Table 5.3.: Area used for comparison (corner points)

Point	Latitude	Longitude
1	66.7133	171.5548
2	66.5987	172.8413
3	66.5238	173.3510
4	66.6419	172.0753

The test area defined in Table 5.3 corresponds to the segment of a datatake covered during the first second of a datatake. The results of the simulation between the DLR simulation tool and STK can be found in Table 5.4.

Table 5.4.: Comparison of results

	SEPPL	STK	Difference[%]
Selected Grid	0.01	0.01	
Area Calculated	168.25	163.89	2.6
Selected Grid	0.001	0.001	
Area Calculated	166.22	165.21	0.006

5.3. Generation of a User Scenario

Because ENMAP is a future satellite mission no real user requests for data of this satellite are available yet. User requests are therefore generated randomly (as also done by [GCLP02]). Request are generated for time intervals in between ground station contacts as only during those times on-board scheduling is of interest. Furthermore users request are only generated for times of sun-illuminated land coverage.

Based on the constellation of the selected satellite orbit and ground station location (see Chapter 5.1) three different scenarios for simulation have been selected: Two scenarios (called Scenario 1 & 2 - see Figures 5.10, 5.11) covering a period of approximately one orbit and one scenario (called Scenario 3 - see Figure 5.11) covering several orbits with land coverage and including eclipse phases (lasting 475 minutes). All scenarios are without GS contact for the satellite. No information exchange with the ground control center is therefore possible. Main scenario characteristics listed in Table 5.5 for start and end at 1. Jan 2008.

Table 5.5.: Main characteristics of user scenarios

Scenario	Covered Area	Start	End	Min	Possible DT
Scenario 1	Asia, Australia	02:22	04:00	98	52
Scenario 2	Africa	08:55	10:25	90	45
Scenario 3	Asia, Australia, Africa	02:30	10:25	475	206

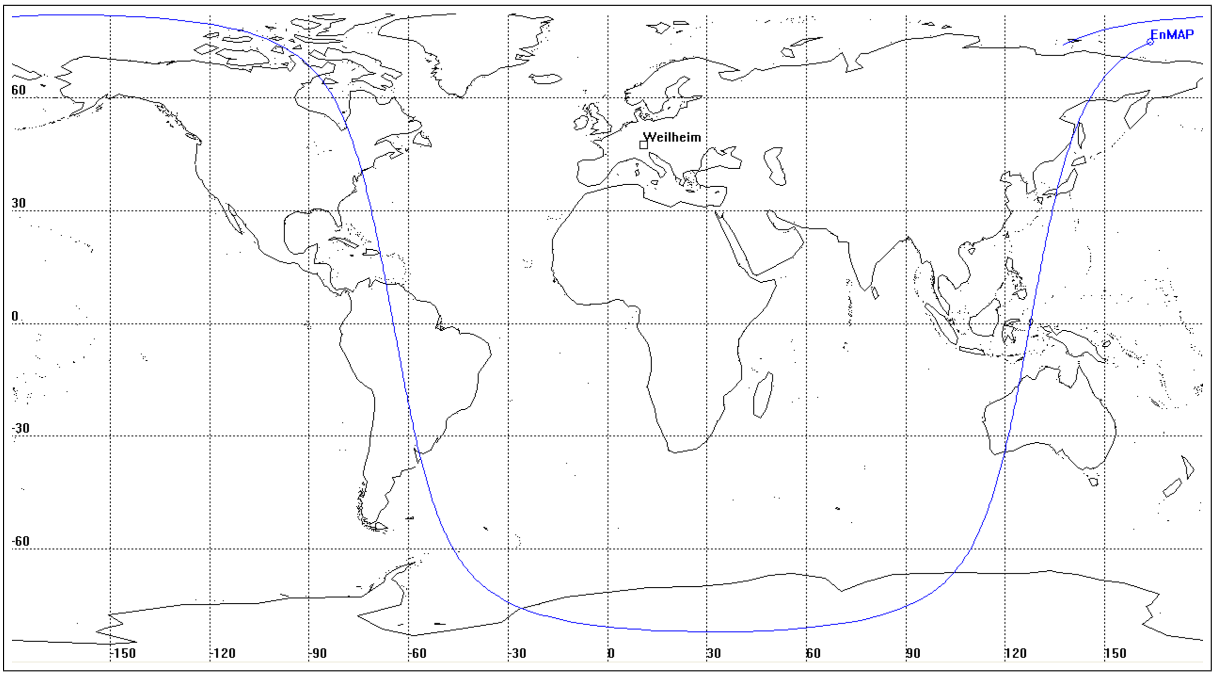


Figure 5.9.: Ground track of scenario 1 (98 min)

The source of user requests for the satellite sensor is a user community interested in the data. This community usually consists of different scientific and/or commercial users with different requirements. Requests cover a certain area (land, water or both) and may have additional constraints like a time window or maximum allowed roll angle. As the user request can cover an area that may be not acquired with only one instrument activation a splitting is done into different executable single datatakes by a ground-based preparation process. The single datatake completely describes the information needed for on-board execution and can be seen as a single job description for the instrument.

All information associated with a DT is described in Table 4.2 and must be provided for a complete DT. For each DT start and stop times are generated. The start and stop times of

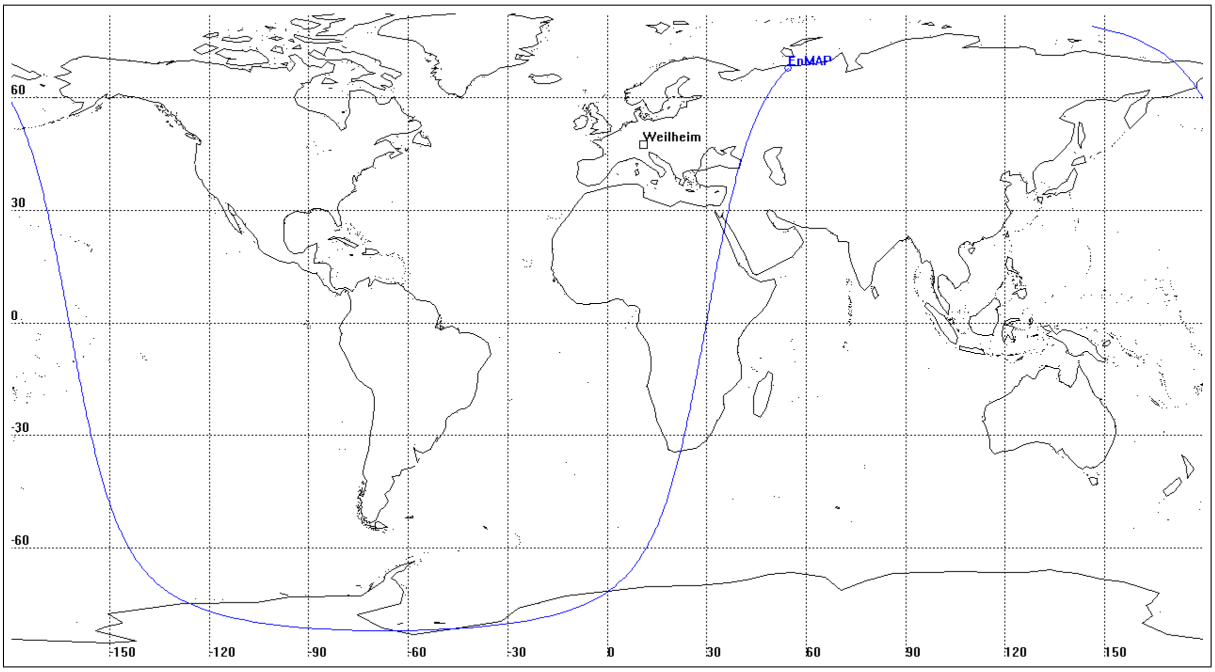


Figure 5.10.: Ground track of scenario 2 (90 min)

the DTs are generated within the given start and stop limits of the user scenario (min and max duration for a single DTs) and under the condition of coverage of illuminated land. Different distribution functions can be used for random generation of data. A rectangular distribution is used for start time, stop time and duration. Start and stop times of the DTs map to the satellite position in orbit. Together with a defined roll angle the covered area of the instrument is given. The roll angle is generated using a normal distribution function (see Annex A.2) as it is assumed that most of the DTs will be nadir or close to nadir [SKH$^+$07]. Importance of each DT request is defined by associating a priority being either 0 or 1. Usage of priorities is common in satellite image scheduling problems (see e.g. [HP99] or [PG98]) but is limited to the two different values for this simulation. It is assumed that at least a so called background mission is active. This allows filling gaps if no user is interested in data. The satellite can than acquire a target of the background mission. As the background mission is not the primary goal of the mission DT - priority is set to lower values (e.g. value 0). The priority of each DT is generated within the given range using a rectangular distribution. A detailed description of all DTs of the user scenarios can be found in Annex A.3.

The user scenario and the acquisition capabilities of the system influence the possible number of DT. Two basic scenarios are imaginable:

- the satellite can acquire infinite small sub-areas in the target area
- the satellite can acquire the target area at once

In the first case, all DTs shrink from areas to points. The number of DTs is than infinite and the

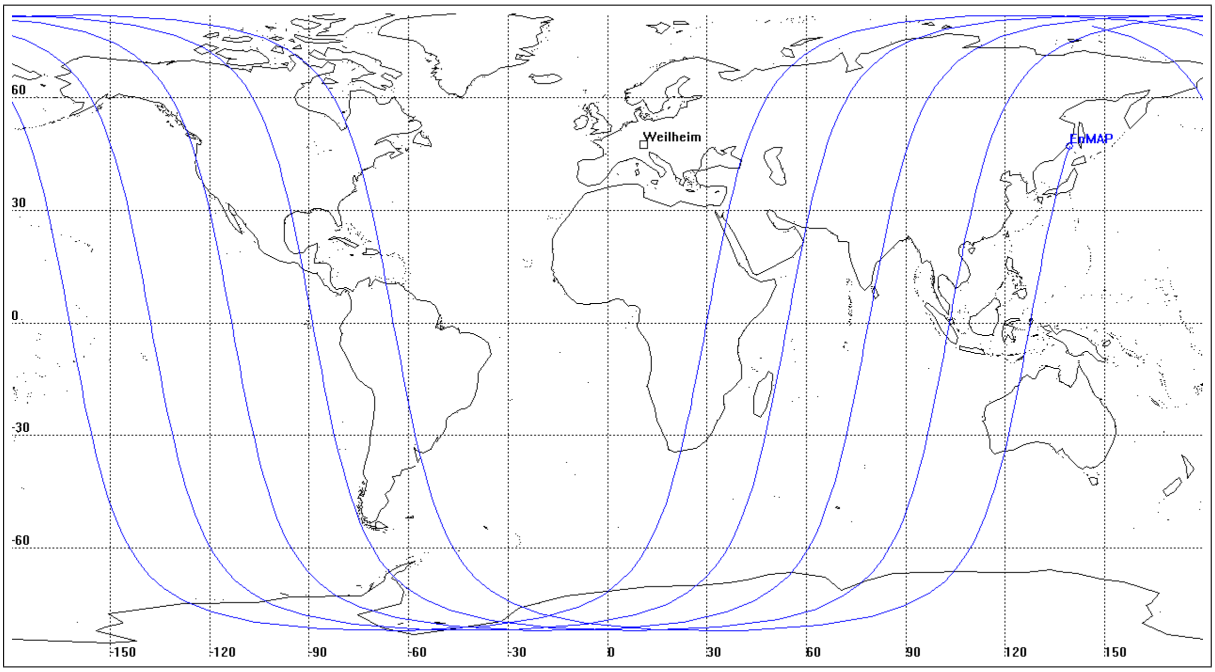


Figure 5.11.: Ground track of scenario 3 (475 min)

classification can be either 0% or 100% cloud covered (see Figure 5.12). In the second case only one DT is made, with the average cloud coverage of that area (see Figure 5.12). Both cases can not be realized as infinite small areas as well as whole areas can not be acquired by sensors (the satellite can acquire smaller areas by rolling the sensor along the axis). The result is a distribution of DTs in different groups as shown in Figure 5.14. Simulation values are expected to be similiar to this distribution.

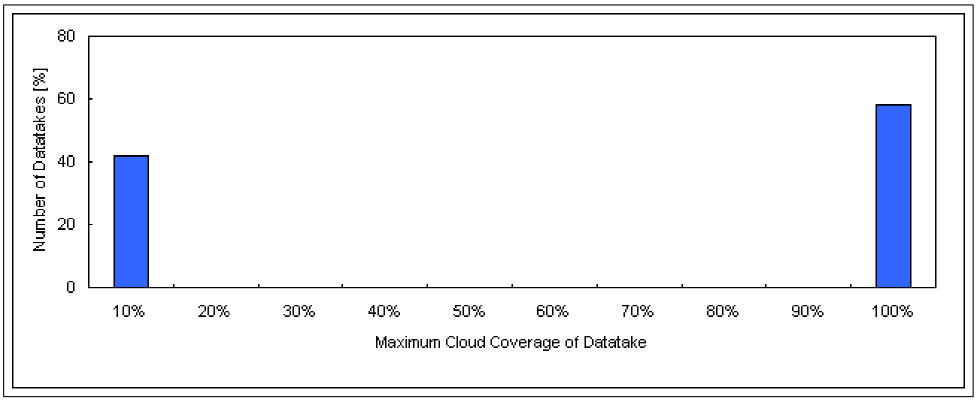


Figure 5.12.: Cloud coverage histogram for assumed infinite small DTs in the target area

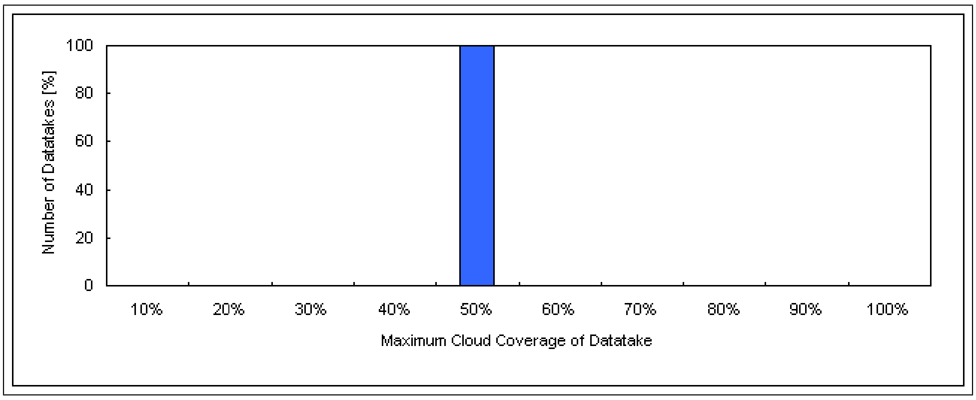


Figure 5.13.: Cloud coverage histogram for one assumed DT of the target area

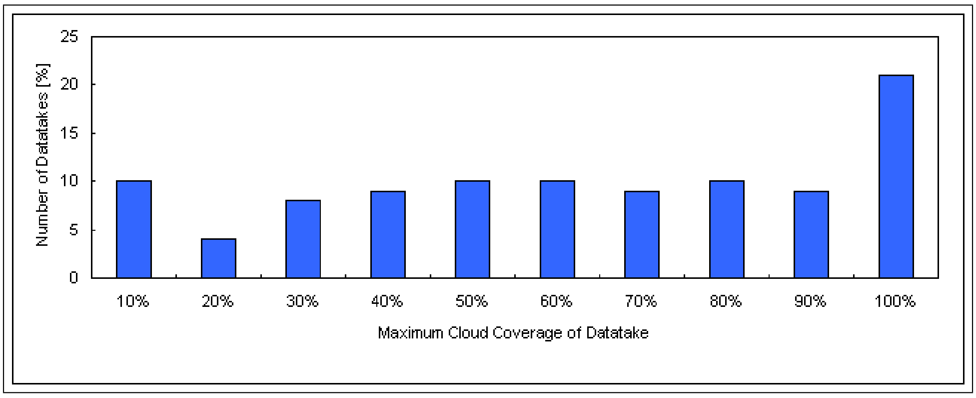


Figure 5.14.: Cloud coverage histogram for 100 assumed DT of the target area

5.4. Setup of Simulation Environment

Existing software of the GSOC mission planning group at has been used for simulation after some necessary extension and modification. The SEPPL software is used (see also Table 2.3) for calculation of planning relevant events. Figure 5.15 shows the original and modified version used for on-board scheduling under cloud coverage simulation. It was necessary to add software modules to allow simulation of on-board scheduling and calculate the resulting cloud coverage in the datatakes. Added functionality contains:

- reading of the user scenarios and generation of the initial schedule
- computation of the covered area using the provided ECMWF cloud data
- after each DT assessment of the image cloud coverage and new scheduling if necessary

The software first reads data from a ”Mission Definition File” (MDF) containing the data about start and stop times of the simulation run, instrument configuration, simulation time step and orbital elements. The ”Event Definition File” (EDF) contains information about the event that should be processed by the software (computation of the covered target on ground together with the cloud coverage). After reading all relevant mission information the trajectory (satellite position and time) is computed within the requested time frame with the given time step and stored in memory.

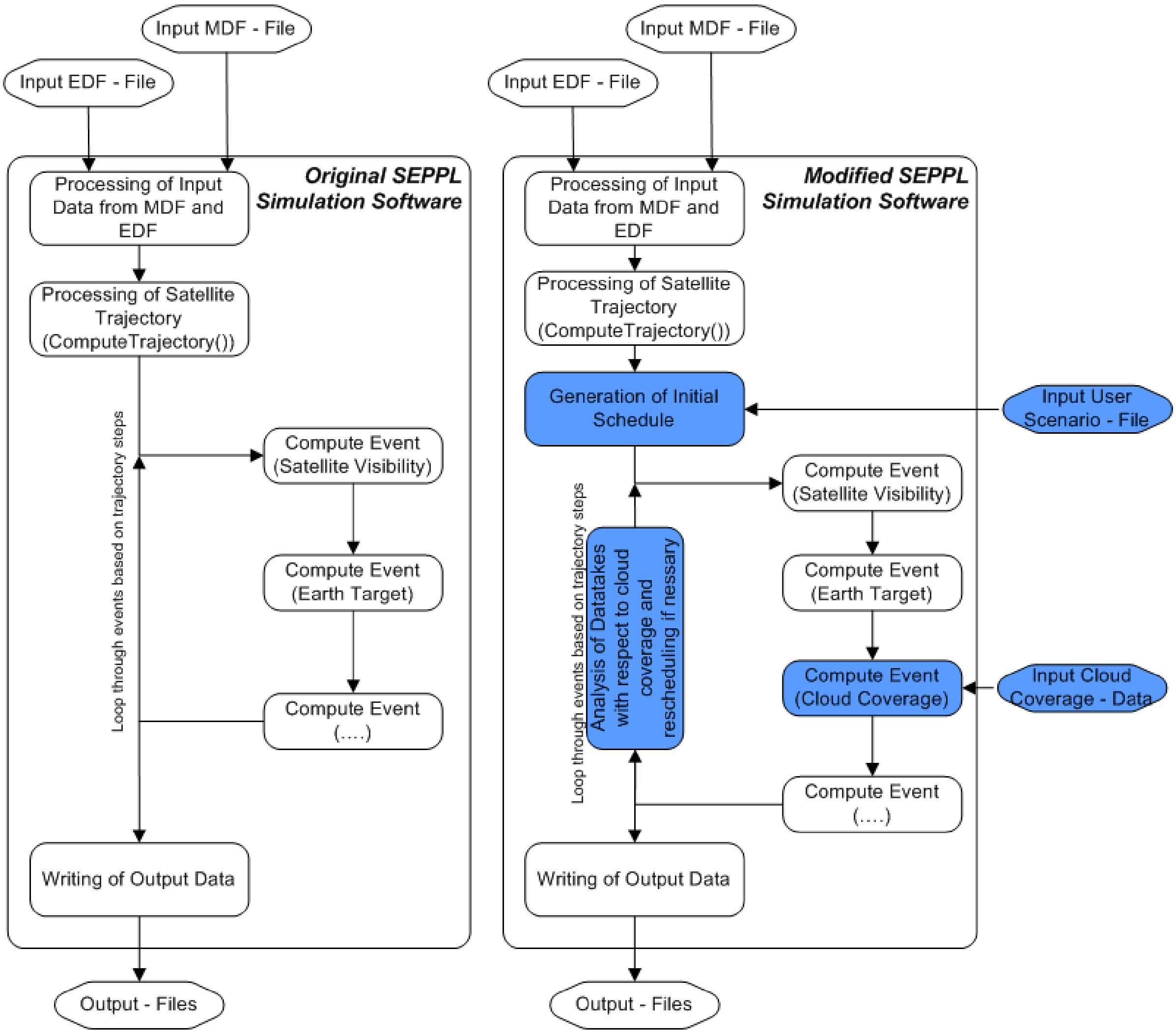


Figure 5.15.: Simulation approach (original software on the left, modified version on the right)

The additional functionality in the blue boxes of Figure 5.15 are:

Generation of Initial Schedule & Input of User Scenario File

The user scenario is loaded and an initial schedule is constructed based on the selected greedy algorithm. This step is done before any DT is executed.

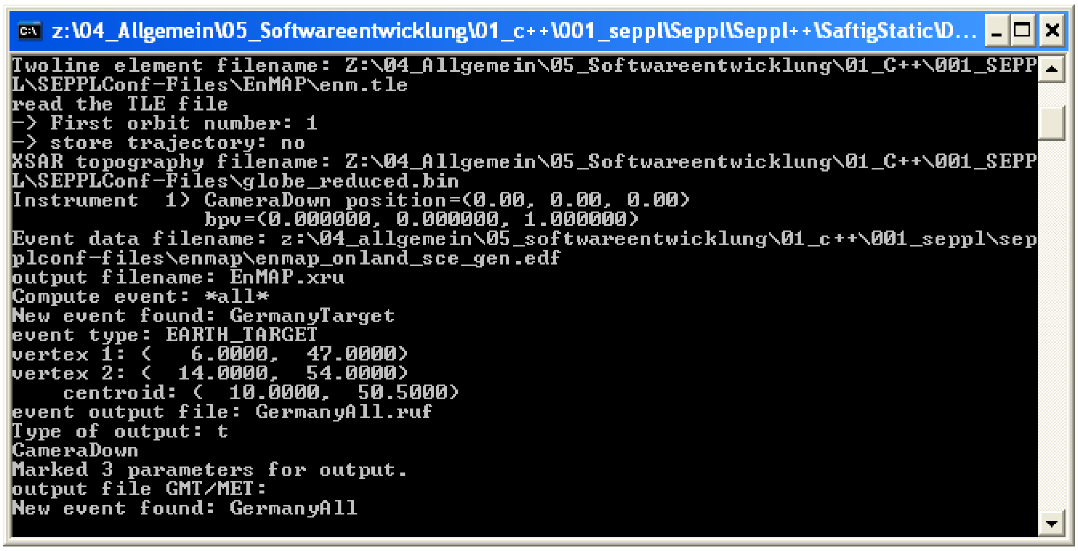
Compute Event (Cloud Coverage) & Input Cloud Coverage File

The software now calculates the event for computation of cloud coverage, which is called in every simulation step. If a DT is scheduled for execution during the computed time step covered area and cloud coverage is calculated. Cloud coverage data is taken from the ECMWF global model.

Analysis of Datatake with respect to cloud coverage and rescheduling of necessary

If cloud coverage of a completed datatake is higher than accepted the schedule is re-scheduled after each DT. Data of that DT is deleted from satellite mass memory. This implies use of an on-board classificator, a software for analysis of raw data identifying amount of cloud coverage in the DT. Principle feasibility of on-board cloud detection has been shown e.g. on the satellite EO-1 [MMGB03]. For this work it is assumed that the result of the classification is available at the end of the acquisition (realtime).

State of the satellite mass memory and the acquired amount of data and its cloud coverage is continuously written into output files. Steps are executed in a loop for each step of the simulation.



```
Twoline element filename: Z:\04_Allgemein\05_Softwareentwicklung\01_C++\001_SEPP
L\SEPPLConf-Files\EnMAP\enm.tle
read the TLE file
-> First orbit number: 1
-> store trajectory: no
XSAR topography filename: Z:\04_Allgemein\05_Softwareentwicklung\01_C++\001_SEPP
L\SEPPLConf-Files\globe_reduced.bin
Instrument  1) CameraDown position=(0.00, 0.00, 0.00)
               bpv=(0.000000, 0.000000, 1.000000)
Event data filename: z:\04_allgemein\05_softwareentwicklung\01_c++\001_seppl\sep
plconf-files\enmap\enmap_onland_sce_gen.edf
output filename: EnMAP.xru
Compute event: *all*
New event found: GermanyTarget
event type: EARTH_TARGET
vertex 1: (   6.0000,   47.0000)
vertex 2: (  14.0000,   54.0000)
    centroid: (  10.0000,   50.5000)
event output file: GermanyAll.ruf
Type of output: t
CameraDown
Marked 3 parameters for output.
output file GMT/MET:
New event found: GermanyAll
```

Figure 5.16.: SEPPL simulation software during computation

5.5. Simulation Results and Discussion

For subsequent diagrams and tables the following units and symbols are used:

- **Area** - Area that is covered by the sensor during one simulation step (A)

- **Area Overall** - Accumulated area that is covered by the sensor since simulation start ($\sum A$)
- **Area Cloud Covered** - Area that is covered by the sensor during one simulation step and blocked by clouds (A_{CC})
- **Area Cloud Covered Overall** - Accumulated area that is covered by the sensor since simulation start and blocked by clouds ($\sum A_{CC}$)
- **Area Stored in Mass Memory** - Accumulated area that is covered in satellite mass memory since simulation start (A_{MM})
- **Area Cloud Covered Overall Stored in Mass Memory** - Accumulated area that is covered in satellite mass memory since simulation start and blocked by clouds (A_{ccMM})
- **Free Mass Memory** - Remaining free mass memory after storing already acquired DT's (MM_{free})

Simulations in the two subsequent chapters concentrate in the first chapter on the general impact of cloud coverage and in the second chapter on the impact if a user scenario is used by the on-board scheduler to schedule acquisitions. Simulations are executed for the year 2008 with underlying data from ECMWF GRIB files based on the year 1992.

5.5.1. General Impact of Cloud Coverage

With the aim to identify the typical relation between cloud covered and uncovered scenes different flight durations of an EnMAP-like EO satellite mission covering the Earth are simulated in which cloud coverage is analyzed. Four different simulations are conducted: durations are selected for two simulations to be approximately one orbit (100 minutes), 24 h to see the development over several orbits and 1 year to cover also possible long-term effects.

The simulation is conducted using the modified simulation software described in the preceding chapter and the cloud coverage data from ECMWF together with the EO mission parameters as described in Chapter 5.1. Simulated values are the covered area of the sensor per simulation step, the cloud covered fraction of the area and the accumulated values since simulation start. This does not take into account any order scenario or satellite limitations as done during more detailed simulations later on, only theoretical possible values are computed.

Figure 5.18 (simulation 1) and 5.19 (simulation 2) show results of the first two simulations. It can be found (as expected) that the additional constraint of land coverage splits the possible imaging time slots into shorter periods compared to the ”sun illumination-only” constraint. The development over 24 hours in simulation 3 is similar to simulation 1 and 2 (see Figure 5.20), the covered area increases while a part of the area is cloud covered. Figure 5.21 shows a long duration simulation (simulation 4) over 1 year. This should eliminate short term effects that may influence results. Cloud data of the year 1992 is used for simulation.

Numerical results are collected in Table 5.6. **The clear tendency is that more than half of the area that can potentially be acquired by an EO satellite is covered by clouds, simulation results vary from 57% - 67%.** This must be compared to the information given in Chapter 3.1. According to Rossow and Schiffer in [RS99] 58.4% of the land masses are cloud covered. Simulation results are close to this value, differences may be due to other observation time frames of the underlying cloud data ([RS99] used an observation period from July 1983 - June 1994). It can generally be concluded that more than half of the imaged area is not cloud free. Simulation results confirm the problem significance for EO satellites.

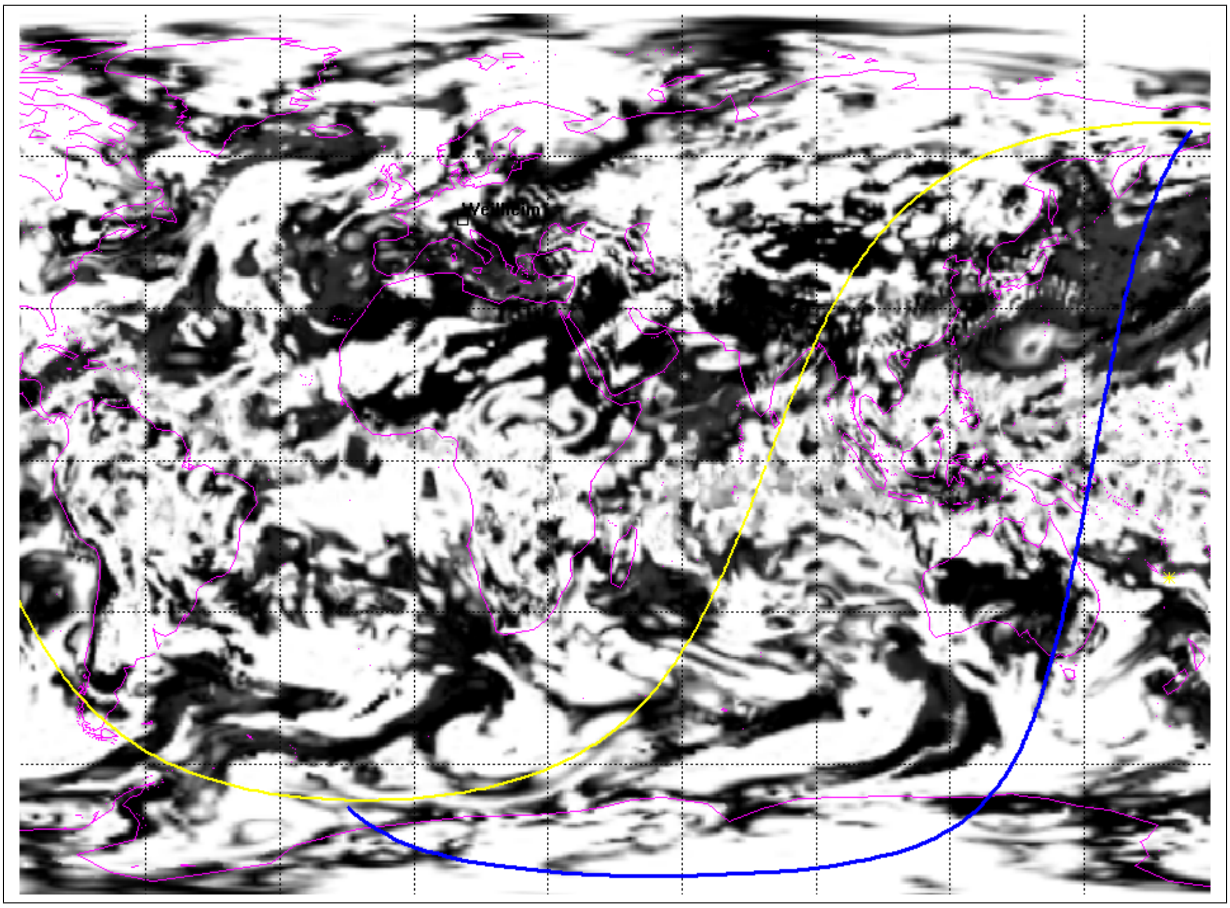


Figure 5.17.: Graphic of Simulation 1

The cloud coverage is displayed on a gray scale (white = complete coverage, black = no coverage). Red lines depict coastal lines and islands. Yellow line shows the day/night delimiter. Blue line is the satellite ground track.

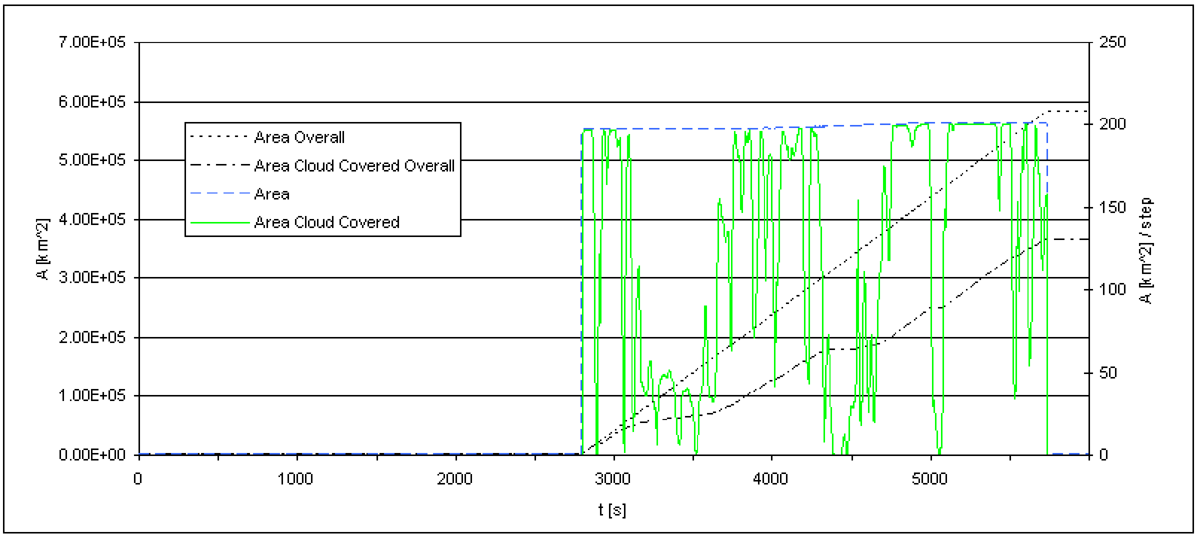


Figure 5.18.: Simulation of 1 orbit (1)

Simulation 1, duration: 100 min, start: 01-Jan-2008:00:00:00, end: 01-Jan-2008:01:40:00, simulation step: 1 s, grid: 0.001°, constraint: sun illumination

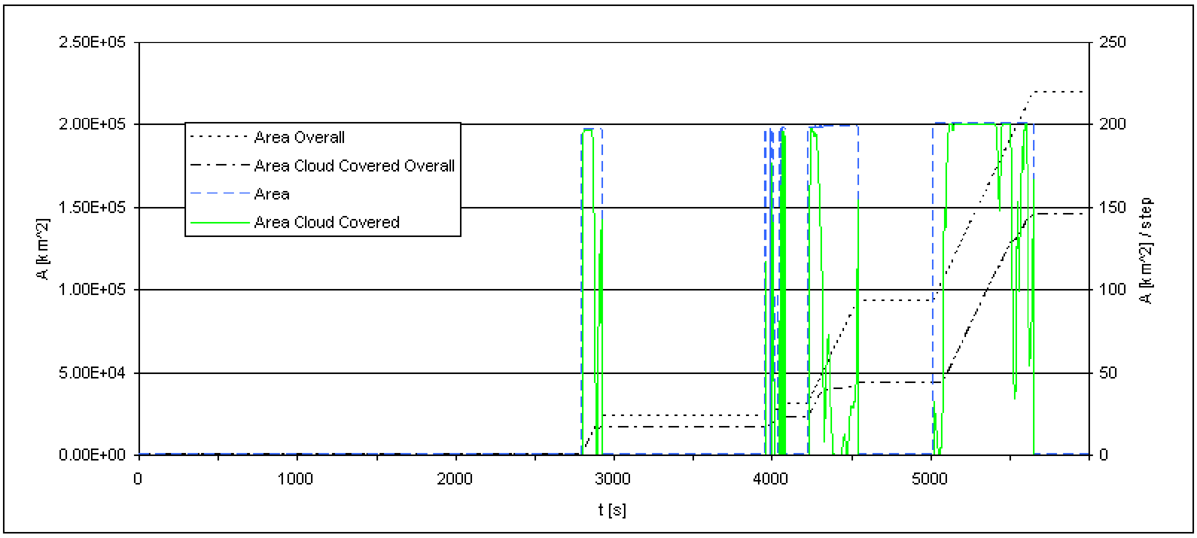


Figure 5.19.: Simulation of 1 orbit (2)

Simulation 2, duration: 100 min, start: 01-Jan-2008:00:00:00, end: 01-Jan-2008:01:40:00, simulation step: 1 s, grid: 0.001°, constraint: coverage of sun illuminated land

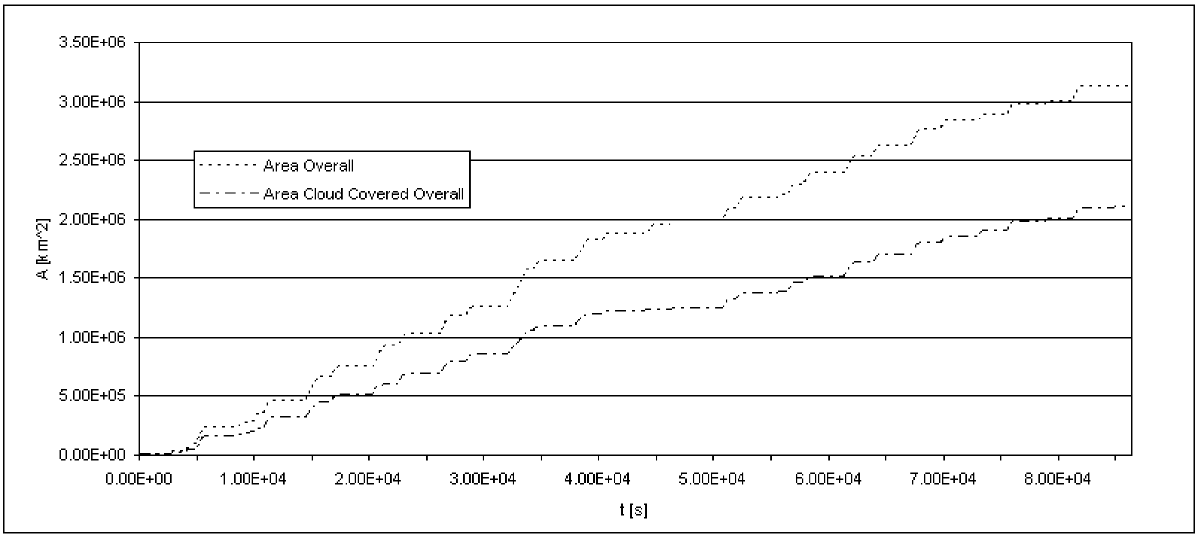


Figure 5.20.: Simulation of 24 h

Simulation 3, duration: 100 min, start: 01-Jan-2008:00:00:00, end: 01-Jan-2008:23:59:00, grid: 0.005°, constraint: coverage of sun illuminated land

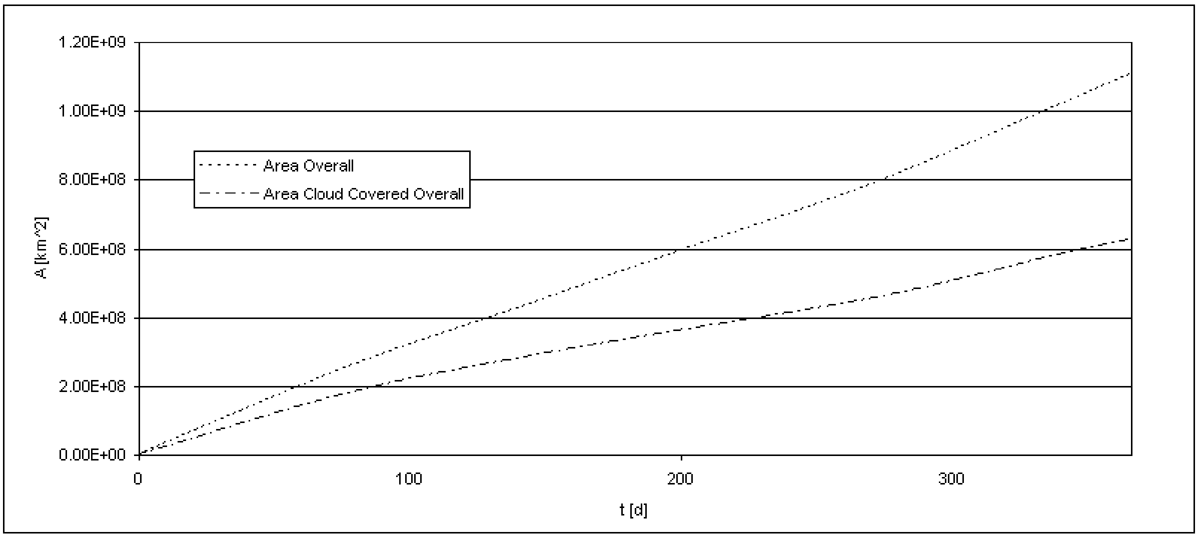


Figure 5.21.: Simulation of 1 year

Simulation 4, duration: 1 year, start: 01-Jan-2008:00:00:00, end: 31-Dec-2008:23:59:59, grid: 0.02°, constraint: coverage of sun illuminated land

Table 5.6.: Simulation results - generic cloud coverage impact

Simulation		1	2	3	4
$\sum A$	$[km^2]$	581×10^3	219×10^3	3127×10^3	1106×10^6
$\sum A_{cc}$	$[km^2]$	363×10^3	145×10^3	2095×10^3	625×10^6
$\sum A_{cc} / \sum A$	$[-]$	0.63	0.66	0.67	0.57

5.5.2. Simulation of Datatake Scheduling

The generic simulation of the preceding chapter is now extended taking the satellite resources (mass-memory and operations limits) into account (see Table 5.1 & 5.2) and user scenarios for the simulation as described in Table 5.5. The scenarios are simulated using different quality criteria (expressed by A_{cc}/A) for the decision to delete or store and down-link a DT.

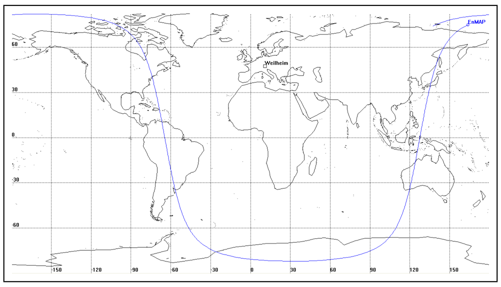
Table 5.7.: Main characteristics of simulations

Scenario	Duration $[s]$	t_{OU} $[s]$	M $[Gbit]$
Scenario 1	5880	146	120
Scenario 2	5400	146	120
Scenario 3	28500	146	360

Table 5.7 shows duration, orbit usage and available mass memory according to following ground station contacts for each simulation. These values are used for the simulation software settings.

Simulation of Scenario 1

For scenario 1 the quality criteria given by A_{cc}/A is decreased in steps of 0.1 from 1.0 to 0.4. Criterion $A_{cc}/A \leq 1$ is equal to static scheduling, the schedule is generated ones and all DT are accepted (maximum allowed cloud coverage 100%), from completely cloud free to completely covered. This simulates the ground based scheduling, the schedule is generated once and executed without taking imaging quality into account. For values $A_{cc}/A < 1$ the schedule is newly generated each time a DT is above selected limit and deleted after acquisition.



For $A_{cc}/A \leq 1$ the initially generated schedule is executed and none of the DT are deleted from satellite mass memory after acquisition. Figure 5.22 shows the diagram of the acquired

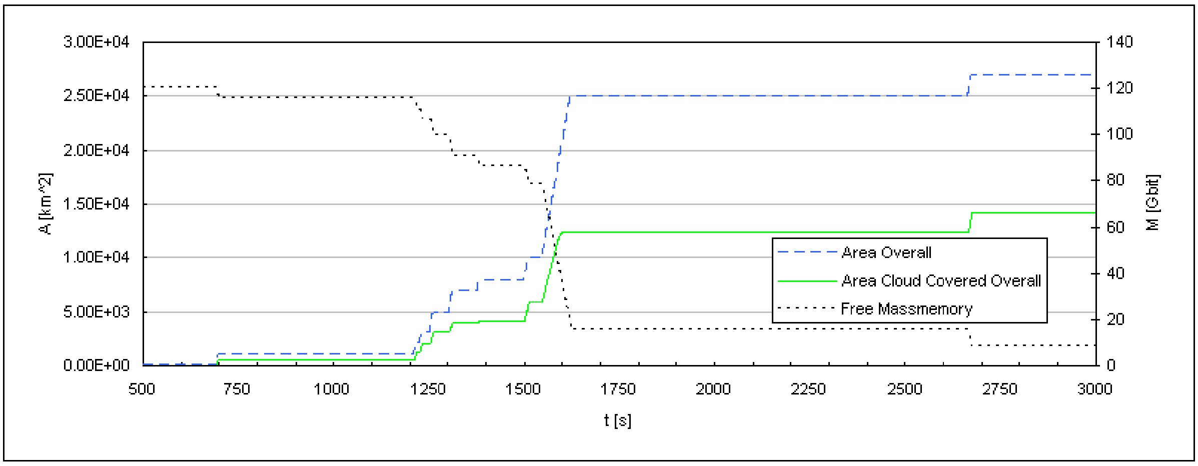


Figure 5.22.: $A_{cc}/A \leq 1.0$ (scenario 1, grid: 0.001°)

area, the cloud covered fraction and the state of mass memory. Tabular results of the executed DT's are given for all simulation runs in Table 5.8. 10 DT's are executed right at the beginning of the simulation time frame, no more DT's are possible due to the available satellite operations time (orbit usage). At the end of simulation time frame an area of $26.8 \times 10^3\,km^2$ with $14.2 \times 10^3\,km^2$ cloud covered has been acquired. 53% of the data is unusable (see Table 5.9).

If $A_{cc}/A < 0.5$ or $A_{cc}/A < 0.4$ is set (Figure 5.23 & 5.24) some of the DT's are deleted after acquisition and scheduling is executed again. Deletion can be found in the diagrams where the state of free mass memory is increased. The lower (more strict) the A_{cc}/A quality criteria the more DT's must be deleted.

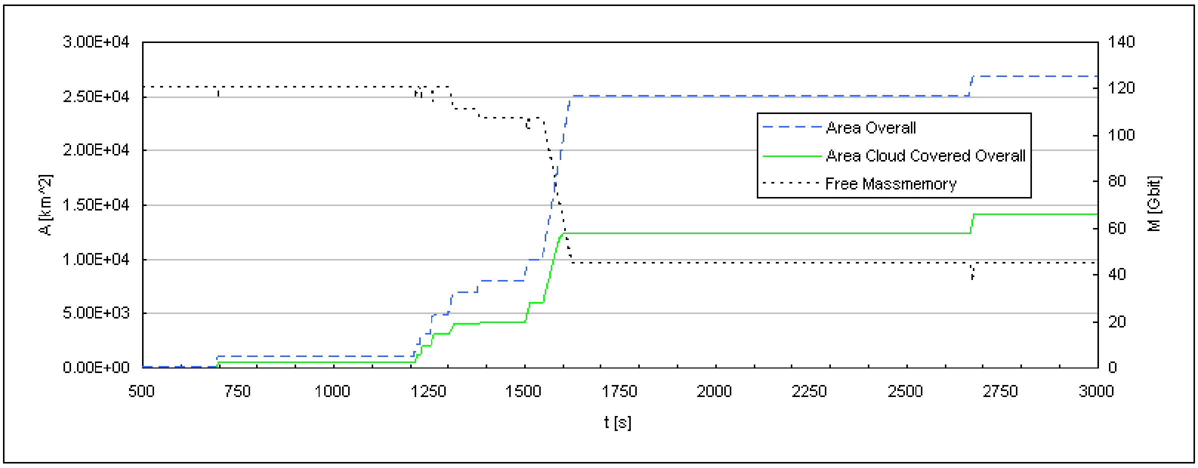


Figure 5.23.: $A_{cc}/A < 0.5$ (scenario 1, grid: 0.001°)

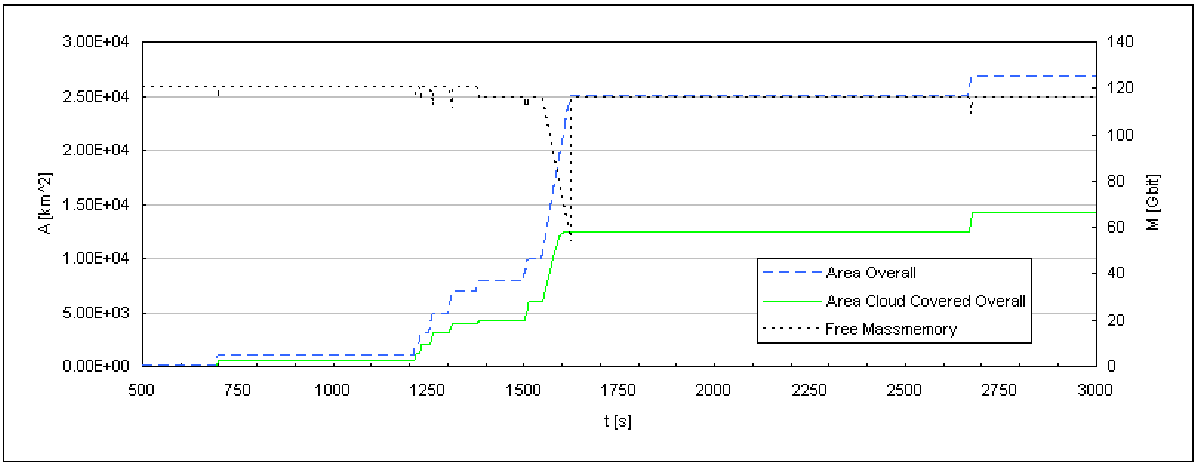


Figure 5.24.: $A_{cc}/A < 0.4$ (scenario 1, grid: 0.001°)

Table 5.9.: Simulation results - scenario 1

A_{cc}/A		≤ 0.4	≤ 0.5	≤ 0.8	≤ 0.9	≤ 1.0
A	$[km^2]$	26.8×10^3	26.8×10^3	26.8×10^3	26.8×10^3	26.8×10^3
A_{cc}	$[km^2]$	14.2×10^3	14.2×10^3	14.2×10^3	14.2×10^3	14.2×10^3
A_{cc}/A	$[-]$	0.53	0.53	0.53	0.53	0.53
A_M	$[km^2]$	996	18.2×10^3	23.0×10^3	24.0×10^3	26.8×10^3
A_{ccM}	$[km^2]$	158	7.5×10^3	10.6×10^3	11.5×10^3	14.2×10^3
A_{ccM}/A_M	$[-]$	0.16	0.41	0.46	0.48	0.53
M_{free}	$[Gbit]$	115.89	44.38	23.83	19.72	8.21

Simulation results of selected simulations of scenario 1 can be found in Table 5.8, Table 5.9 and Figure 5.25. Rescheduling the imaging sequence has no effect on the return of the mission, except that nearly all DTs are deleted after acquisition as they don't meet the A_{cc}/A quality criterion. For $A_{cc}/A \leq 0.4$ only one (number 23) of the DTs remains in mass memory, with $A_{cc}/A \leq 0.5$ only three DTs remain for downlink. Only if $A_{cc}/A \leq 1$ mass memory is used completely. The limiting criterion in this scenario is the orbit usage of 146 s maximum, which equals 120 GBit of data. The satellite is not able to acquire more images and store only the better ones in mass memory.

Executed DT - $A_{cc}/A \leq 1$

DT-ID	$A[km^2]$	$A_{cc}[km^2]$	$[\%_{cc}]$	$t_{DT}[s]$	Prio	Deleted
3	984.00	609.69	0.62	5.00	1	No
4	1031.82	636.09	0.62	5.00	1	No
9	1025.59	777.72	0.76	6.00	1	No
16	1798.25	1152.10	0.64	9.00	1	No
20	2021.36	858.32	0.42	11.00	0	No
23	995.88	158.35	0.16	5.00	0	No
28	1001.49	836.73	0.84	5.00	0	No
30	1010.10	943.77	0.93	5.00	0	No
41	15104.21	6456.05	0.43	76.00	1	No
43	1804.60	1804.60	1.00	9.00	0	No

Executed DT - $A_{cc}/A \leq 0.5$

DT-ID	$A[km^2]$	$A_{cc}[km^2]$	$[\%_{cc}]$	$t_{DT}[s]$	Prio	Deleted
3	984.00	609.69	0.62	5.00	1	Yes
4	1031.82	636.09	0.62	5.00	1	Yes
9	1025.59	777.72	0.76	6.00	1	Yes
16	1798.25	1152.10	0.64	9.00	1	Yes
20	2021.36	858.32	0.42	11.00	0	No
23	995.88	158.35	0.16	5.00	0	No
28	1001.49	836.73	0.84	5.00	0	Yes
30	1010.10	943.77	0.93	5.00	0	Yes
41	15104.21	6456.05	0.43	76.00	1	No
43	1804.60	1804.60	1.00	9.00	0	Yes

Executed DT - $A_{cc}/A \leq 0.4$

DT-ID	$A[km^2]$	$A_{cc}[km^2]$	$[\%_{cc}]$	$t_{DT}[s]$	Prio	Deleted
3	984.00	609.69	0.62	5.00	1	Yes
4	1031.82	636.09	0.62	5.00	1	Yes
9	1025.59	777.72	0.76	6.00	1	Yes
16	1798.25	1152.10	0.64	9.00	1	Yes
20	2021.36	858.32	0.42	11.00	0	Yes
23	995.88	158.35	0.16	5.00	0	No
28	1001.49	836.73	0.84	5.00	0	Yes
30	1010.10	943.77	0.93	5.00	0	Yes
41	15104.21	6456.05	0.43	76.00	1	Yes
43	1804.60	1804.60	1.00	9.00	0	Yes

Table 5.8.: Simulation details - scenario 1

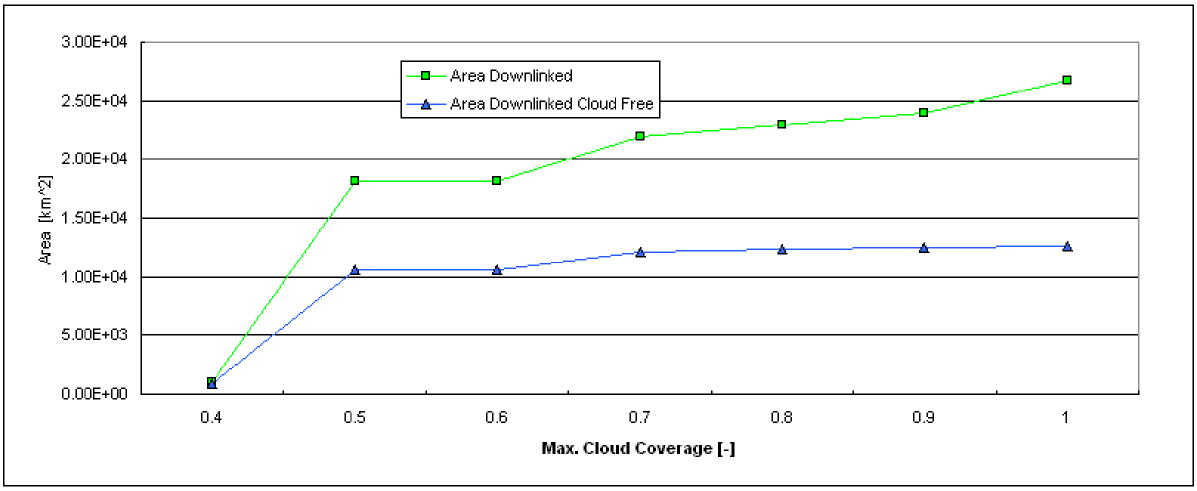
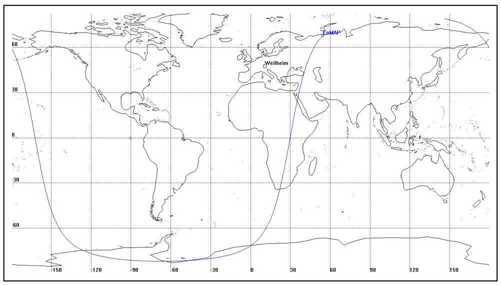


Figure 5.25.: Scenario 1 - simulation results

Simulation of Scenario 2

Scenario 2 is similar to scenario 1 with the difference of longer land coverage. Scenario 1 consists of many short DT's due to the many small islands in the target area. For this scenario longer DT's are expected (coverage of land masses over africa) with A_{cc}/A values closer to the global 58,9% value. For $A_{cc}/A \leq 1$ the initially generated schedule is executed and none of the DT are deleted from satellite mass memory after acquisition. Figure 5.26 shows the diagram of the acquired area, the cloud covered fraction and the state of mass memory. Tabular results of the executed DT's are given for selected simulation runs in Table 5.11.



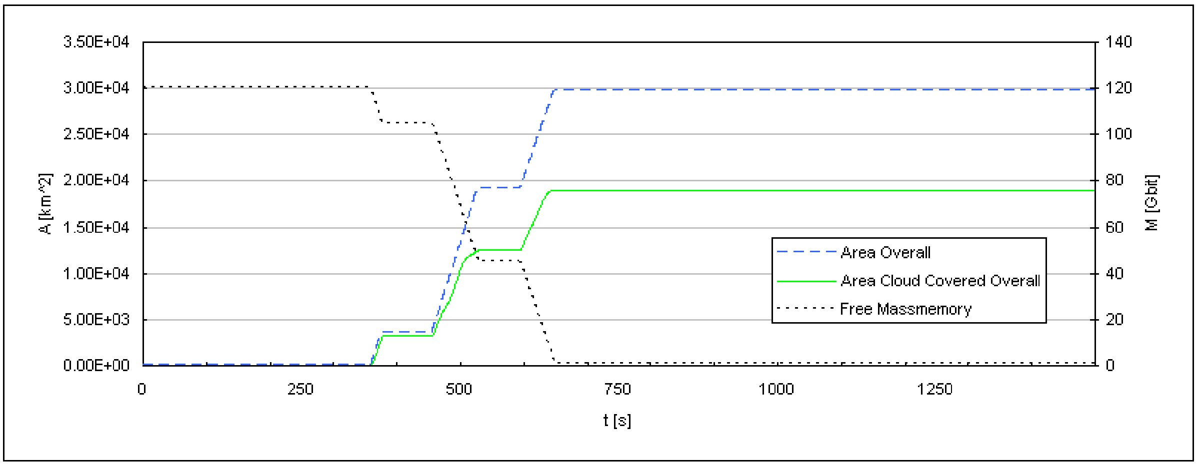


Figure 5.26.: $A_{cc}/A \leq 1.0$ (scenario 2, grid: 0.001°)

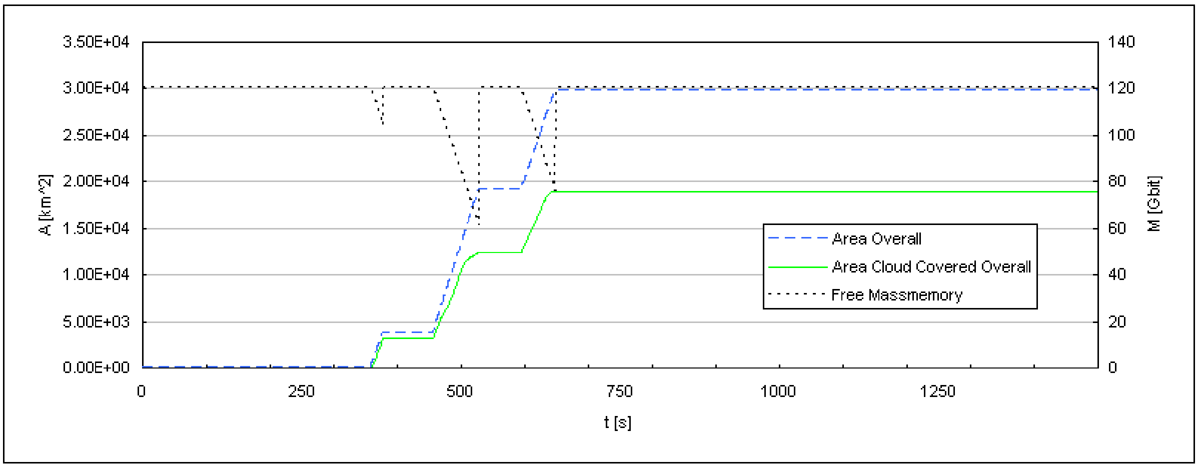


Figure 5.27.: $A_{cc}/A \leq 0.6$ (scenario 2, grid: 0.001°)

3 DT's are executed right at the beginning of the simulation time frame, no more DT's are possible due to the avaible memory. At the end of simulation time frame an area of $29.8 \times 10^3\,km^2$ with $19.0 \times 10^3\,km^2$ cloud covered has been acquired. 0.64 of the data are unusable (see Table 5.10).

If $A_{cc}/A \leq 0.5$ or $A_{cc}/A \leq 0.4$ is set (Figure 5.28 & 5.29) some of the DT's are deleted after acquisition and scheduling is executed again. Deletion can be found in the diagrams where the state of mass memory is increased. The lower the A_{cc}/A quality criteria the more DT's must be executed. Scheduling results can be found in Table 5.11.

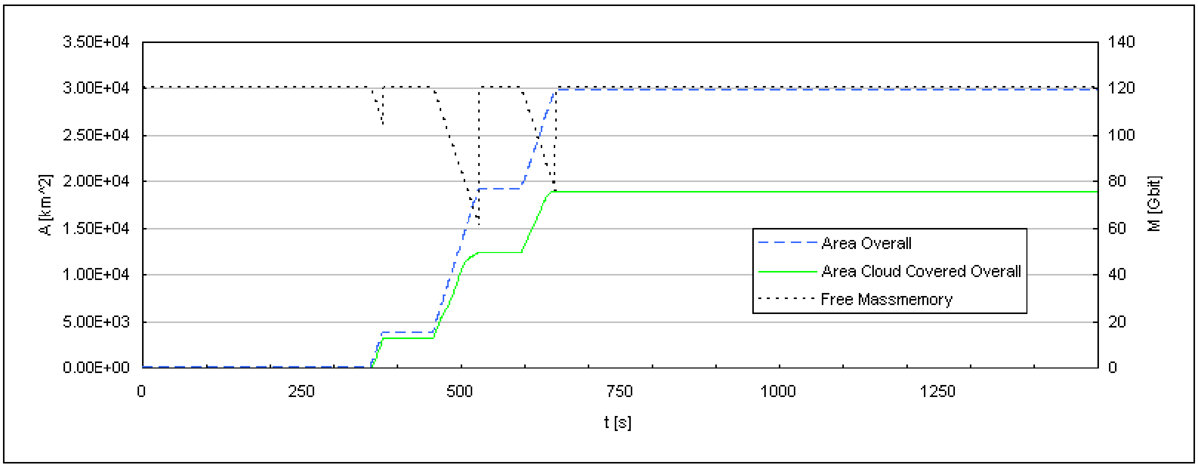


Figure 5.28.: $A_{cc}/A \leq 0.5$ (scenario 2, grid: 0.001°)

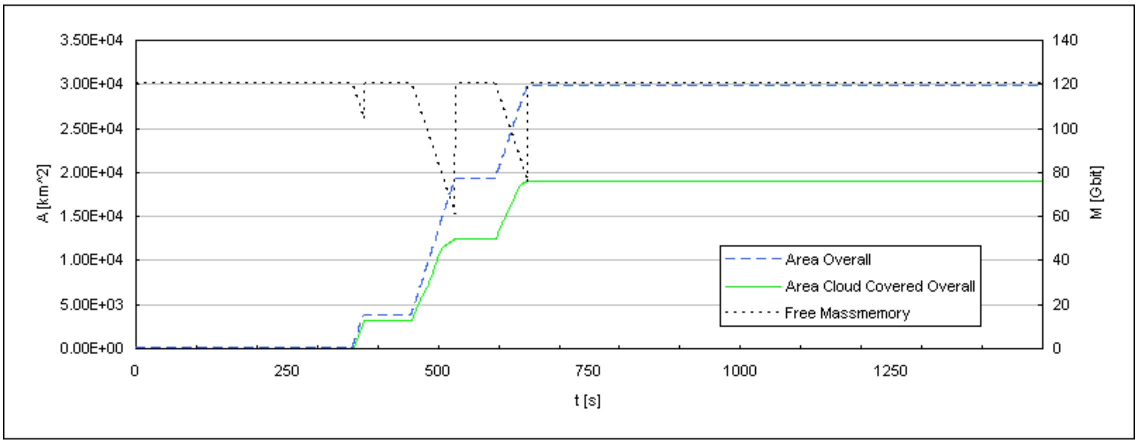


Figure 5.29.: $A_{cc}/A \leq 0.4$ (scenario 2, grid: 0.001°)

Table 5.10.: Simulation results - scenario 2

A_{cc}/A		≤ 0.6	≤ 0.7	≤ 0.8	≤ 0.9	≤ 1.0
A	$[km^2]$	29.8×10^3	29.8×10^3	29.8×10^3	29.8×10^3	29.8×10^3
A_{cc}	$[km^2]$	19.0×10^3	19.0×10^3	19.0×10^3	19.0×10^3	19.0×10^3
A_{cc}/A	$[-]$	0.64	0.64	0.64	0.64	0.64
A_M	$[km^2]$	0	26.1×10^3	26.1×10^3	29.8×10^3	29.8×10^3
A_{ccM}	$[km^2]$	0	15.8×10^3	15.8×10^3	19.0×10^3	19.0×10^3
A_{ccM}/A_M	$[-]$	–	0.60	0.60	0.64	0.64
M_{free}	$[Gbit]$	120	16.43	16.43	0.81	0.81

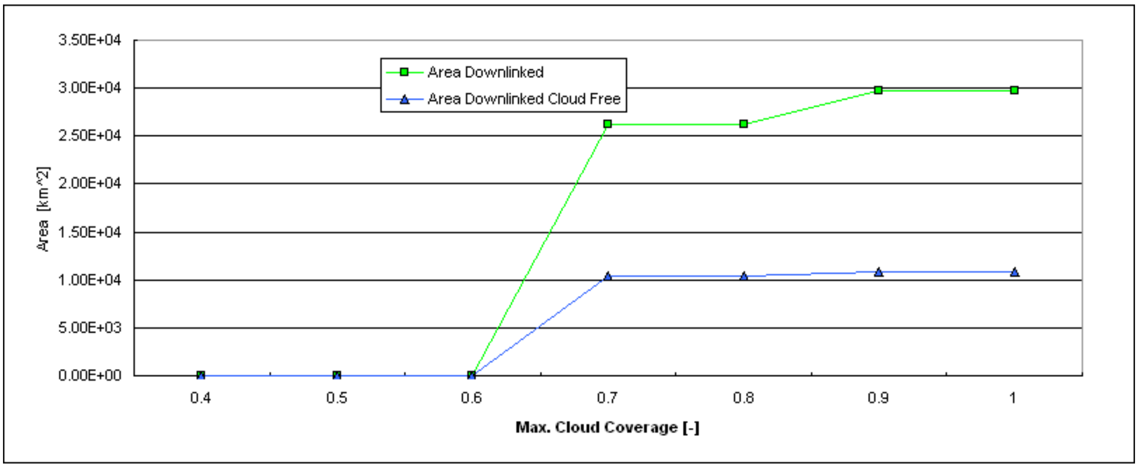


Figure 5.30.: Scenario 2 - simulation results

Executed DT - $A_{cc}/A \leq 1$

DT-ID	$A[km^2]$	$A_{cc}[km^2]$	$[\%_{cc}]$	$t_{DT}[s]$	Prio	Deleted
0	3610.72	3161.13	0.88	19.00	1	No
4	15489.32	9326.82	0.60	72.00	1	No
5	10656.88	6480.12	0.61	54.00	1	No

Executed DT - $A_{cc}/A \leq 0.9$

DT-ID	$A[km^2]$	$A_{cc}[km^2]$	$[\%_{cc}]$	$t_{DT}[s]$	Prio	Deleted
0	3610.72	3161.13	0.88	19.00	1	No
4	15489.32	9326.82	0.60	72.00	1	No
5	10656.88	6480.12	0.61	54.00	1	No

Executed DT - $A_{cc}/A \leq 0.8$

DT-ID	$A[km^2]$	$A_{cc}[km^2]$	$[\%_{cc}]$	$t_{DT}[s]$	Prio	Deleted
0	3610.72	3161.13	0.88	19.00	1	Yes
4	15489.32	9326.82	0.60	72.00	1	No
5	10656.88	6480.12	0.61	54.00	1	No

Executed DT - $A_{cc}/A \leq 0.7$

DT-ID	$A[km^2]$	$A_{cc}[km^2]$	$[\%_{cc}]$	$t_{DT}[s]$	Prio	Deleted
0	3610.72	3161.13	0.88	19.00	1	Yes
4	15489.32	9326.82	0.60	72.00	1	No
5	10656.88	6480.12	0.61	54.00	1	No

Executed DT - $A_{cc}/A \leq 0.6$

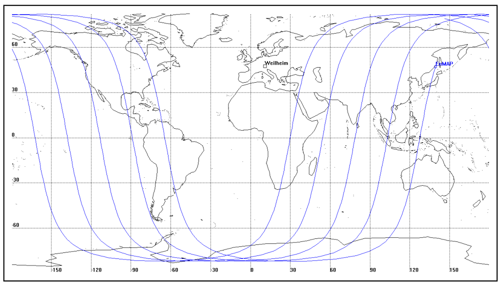
DT-ID	$A[km^2]$	$A_{cc}[km^2]$	$[\%_{cc}]$	$t_{DT}[s]$	Prio	Deleted
0	3610.72	3161.13	0.88	19.00	1	Yes
4	15489.32	9326.82	0.60	72.00	1	Yes
5	10656.88	6480.12	0.61	54.00	1	Yes

Table 5.11.: Simulation details - scenario 2

Table 5.10 and Figure 5.30 show an overview of the simulation results. Like for scenario 1 the limiting constraint is the maximum orbit usage. The maximum usage of mass memory is only given for $A_{cc}/A \leq 1$. Lower values lead to an incomplete usage of mass memory at higher image quality.

Simulation of Scenario 3

Scenario is 475 minutes long, comprises several orbits[7] and is followed by two subsequent ground station contacts over Weilheim GS with an overall duration of 1200 s (at 5° minimum elevation). With the assumed downlink datarate 360 Gbit of payload data can be downlinked during both contacts. The possible instrument operations time (orbit usage) is higher than the amount of data that can be downlinked. For this scenario an improvement of the data returned is expected. The simulation results are listed in Table 5.12 for values of A_{cc}/A between 1 and 0.4.



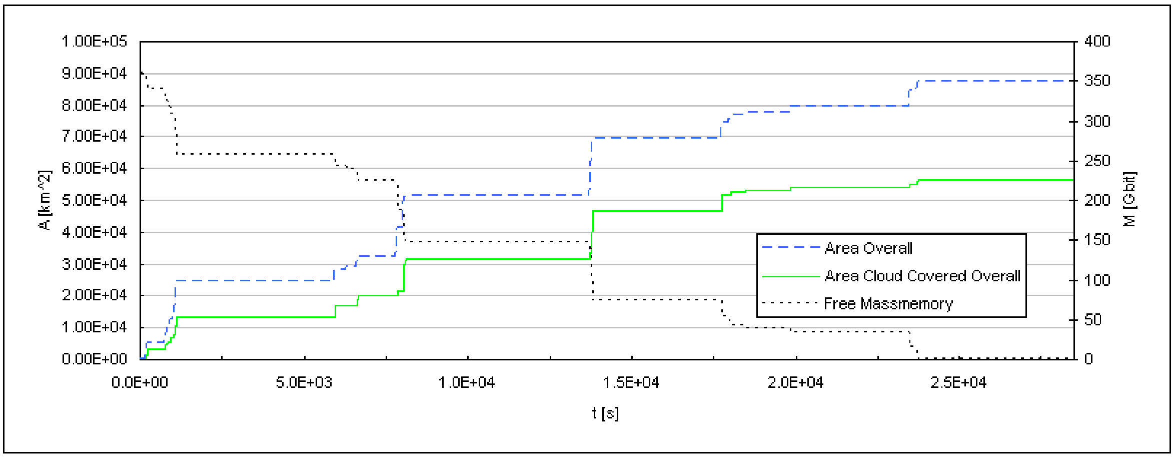


Figure 5.31.: $A_{cc}/A \leq 1.0$ (scenario 3, grid: 0.001°)

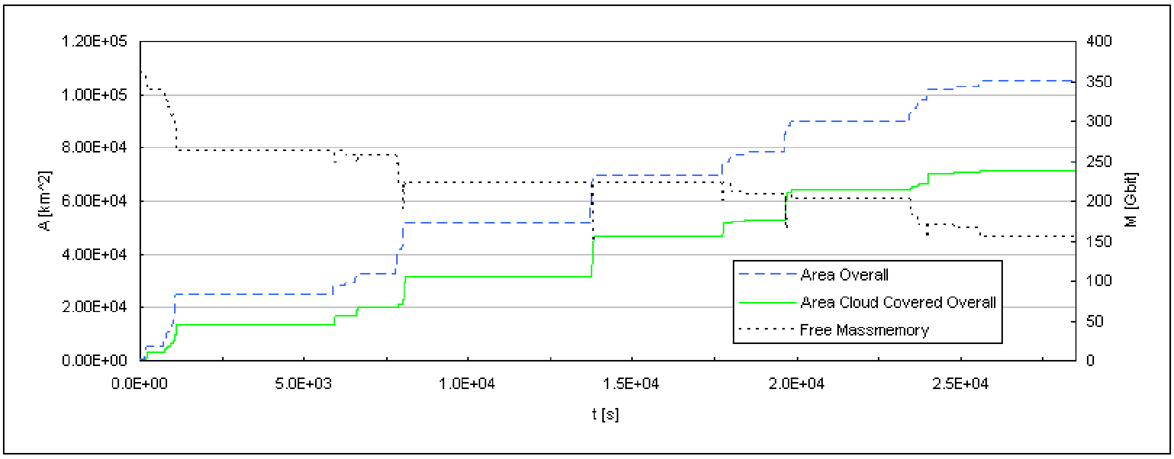


Figure 5.32.: $A_{cc}/A \leq 0.7$ (scenario 3, grid: 0.001°)

---

[7] t_{OU} overall is therefore higher as this values indicates maximum sensor operations time **per orbit**

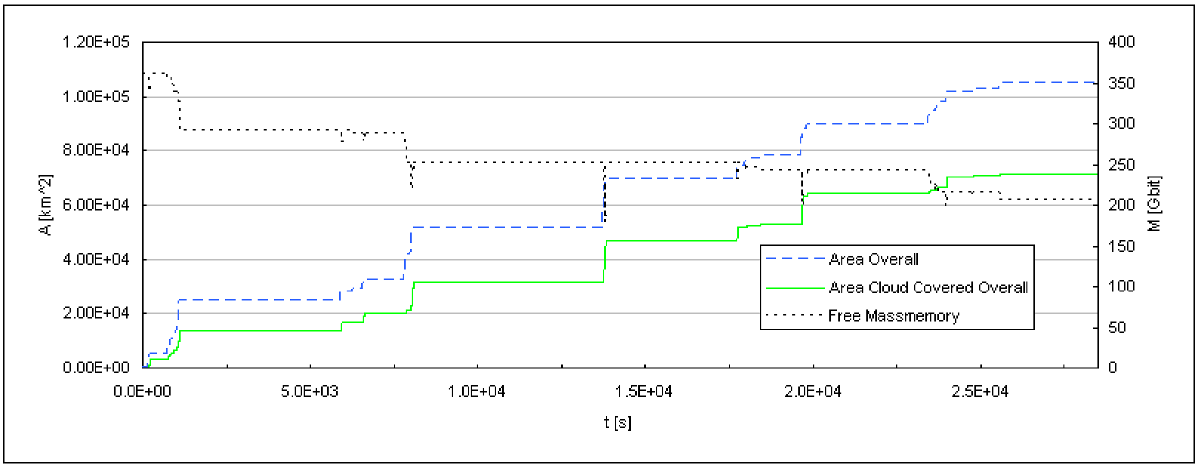


Figure 5.33.: $A_{cc}/A \leq 0.6$ (scenario 3, grid: 0.001°)

Table 5.12.: Simulation results - scenario 3

A_{cc}/A		≤ 0.4	≤ 0.5	≤ 0.6	≤ 0.7
A	$[km^2]$	105.0×10^3	105.0×10^3	105.0×10^3	105.0×10^3
A_{cc}	$[km^2]$	71.7×10^3	71.7×10^3	71.7×10^3	71.7×10^3
A_{cc}/A	$[-]$	0.68	0.68	0.68	0.68
A_M	$[km^2]$	21.5×10^3	25.9×10^3	37.8×10^3	49.7×10^3
A_{ccM}	$[km^2]$	4.2×10^3	6.0×10^3	12.4×10^3	$20. \times 10^3$
A_{ccM}/A_M	$[-]$	0.20	0.23	0.33	0.40
M_{free}	$[Gbit]$	272.87	254.78	204.64	154.50

A_{cc}/A		≤ 0.8	≤ 0.9	≤ 1.0
A	$[km^2]$	105.0×10^3	105.0×10^3	87.2×10^3
A_{cc}	$[km^2]$	71.7×10^3	71.7×10^3	56.6×10^3
A_{cc}/A	$[-]$	0.68	0.68	0.65
A_M	$[km^2]$	49.7×10^3	72.8×10^3	87.2×10^3
A_{ccM}	$[km^2]$	20.0×10^3	40.1×10^3	56.6×10^3
A_{ccM}/A_M	$[-]$	0.40	0.55	0.65
M_{free}	$[Gbit]$	154.50	59.15	0.79

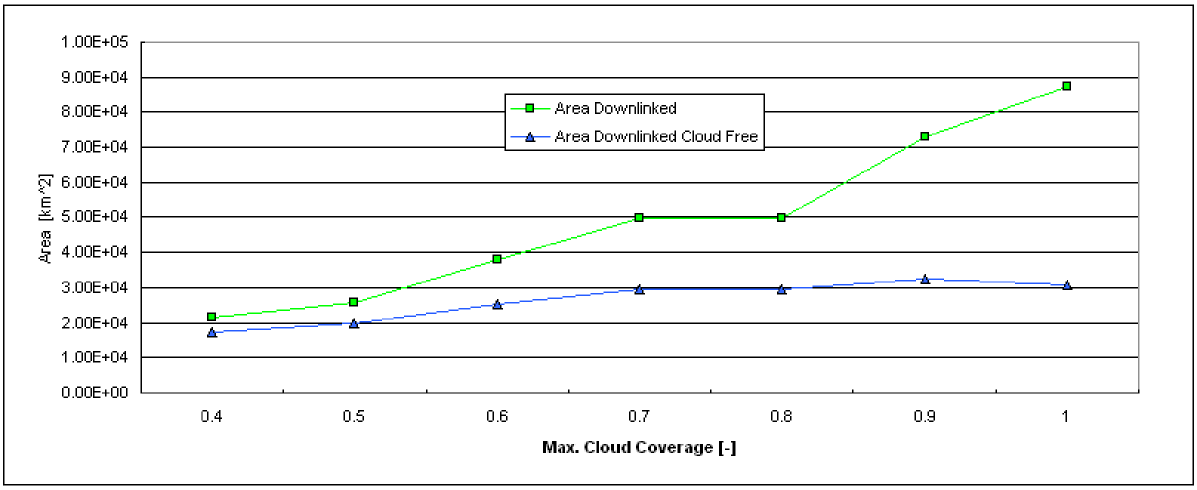


Figure 5.34.: Scenario 3 - simulation results

Results show an increase of cloud free delivered area by 6.8% ($32.7 \times 10^3\,\text{km}^2$) for scenario 3 at $A_{cc}/A \leq 0.9$. For $A_{cc}/A \leq 1.0$ a cloud free area of only $30.6 \times 10^3\,\text{km}^2$ is down-linked by the satellite.

Simulation 3 has been executed again as a 475 minute scenario with identical start and stop times but moved from January to June. Results are shown in Figure 5.35. Highest amount of cloud free area is reached at $A_{cc}/A \leq 0.8$ with $49.8 \times 10^3\,\text{km}^2$ compared $A_{cc}/A \leq 1.0$ with $34.5 \times 10^3\,\text{km}^2$ which is additional 44%. The difference is based on the fact that the different datatakes are relatively long. So some big cloud free datatakes in June increase the values significantly.

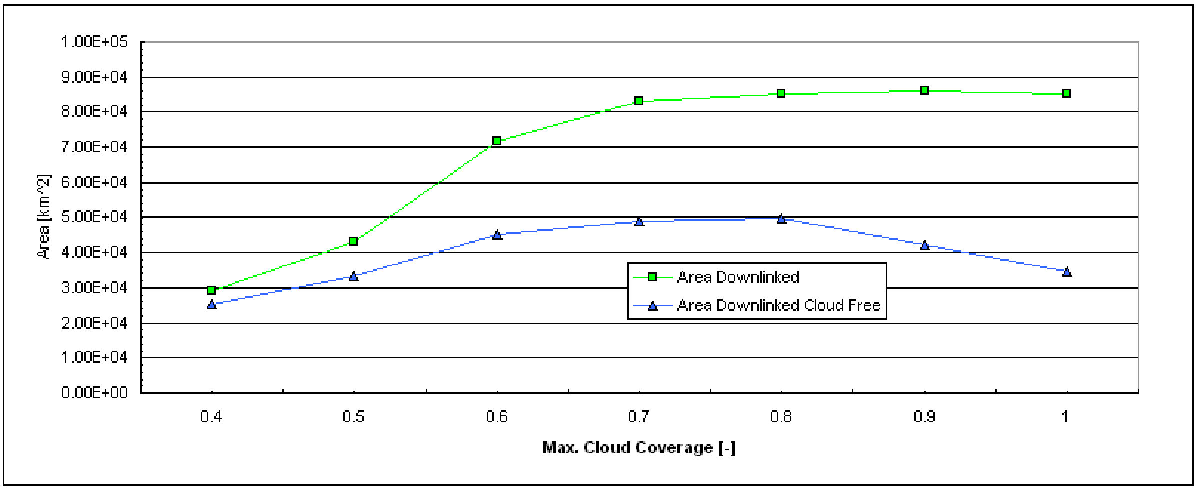


Figure 5.35.: Scenario 4 - simulation results

5.5.3. Discussion of Simulation Results

General: In Chapter 5.5.1 the general impact of cloud coverage on the EnMAP satellite mission has been simulated. Accumulated values for cloud coverage are in the range from 57 % - 67 %. These values match with the value of 58.4% as calculated by Rossow and Schiffer in [RS99].

Simulation of Operations: The given EnMAP satellite system as planned for launch in 2012 has been simulated with an assumed classification of the cloud coverage after acquisition with subsequent re-scheduling activities (based on the greedy scheduling) if the A_{cc}/A relation is exceeded. This constellation is a typical post-pass re-scheduling system, assumed to be the most likely for future EO satellites. The EnMAP system, as an example EO-mission, showed no advantage compared to on-ground scheduling in single orbit scenarios 1 & 2 with on-board scheduling. ***For scenario 3 and its modification (January and June) a significant improvement between 6 % - 44 % of the satellite image return could be achieved.*** Advantages in scenario 3 simulations are based on the fact that the satellite is able to executed more image acquisitions than can be stored and downlinked during the following ground station contact: 360 Gbit can be downlinked but 583 Gbit can be acquired (based on the orbit usage of 146 s/orbit $\approx$ 710 s imaging time). These statements are valid for the existing satellite augmented by on-board classification and on-board scheduling. So also the current design of EnMAP can use on-board scheduling in a beneficial way (satellite design is based on the assumption of on-ground scheduling)

For scenarios 1 & 2 the Equation 4.3 becomes:

$$t_{OU} R_S = M \tag{5.10}$$

and for scenario 3 it becomes:

$$t_{OU} R_S > M \tag{5.11}$$

Extension of Simulation: If a modification of the satellite is assumed concerning an increased operations time of the optical sensor t_{OU}, ***it is expected that also scenarios 1 & 2 could perform better concerning the return of on-board scheduling***. Therefore the orbit usage (maximum sensors operations time per orbit) is doubled to 292 s while the available memory is left at 120 Gbit. Figure 5.36 shows the simulation results of scenario 2 with $t_{OU} = 292\ s$ and $M = 120\ Gbit$. The maximum cloud free area delivered is reached at $A_{cc}/A \leq 0.7$ with $A = 16.9 \times 10^3\ km^2$.

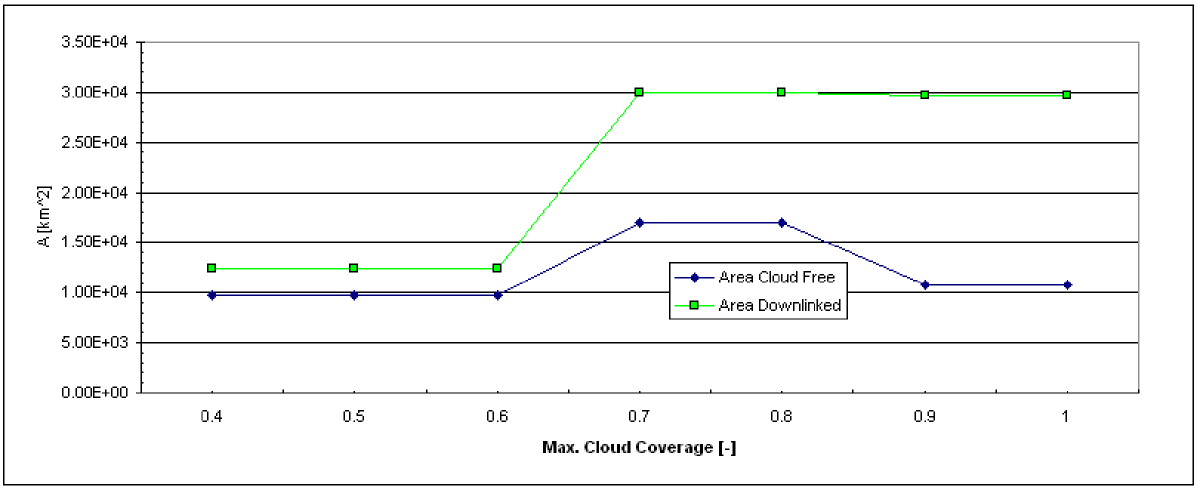


Figure 5.36.: Scenario 2 simulation results with $t_{OU} = 292\ s$

Design changes for an increased t_{OU} are necessary at only some parts of the satellite for a satellite of this type (one nadir looking instrument).

Additional Changes: Modifications are necessary to provide additional energy for instrument operations (powering of instrument, possibly heating or cooling). This requires a changed power system design (more powerful solar generator and more power storage). Furthermore software for on-board scheduling and classification must be executed. **Aspects Unchanged:** No additional mass memory, down-link time, increased data rate for down-link or additional ground stations are required.

For scenario 3, EnMAP is already in it's basic design able to benefit from on-board scheduling. For other scenarios of this satellite or other satellites of this type changes like the one mentioned above can be applied for an increase in overall image return. This can be achieved without significantly increasing satellite hardware requirements and therefore costs.

6. Autonomous Mission Planning versus Conventional Solutions

6.1. Economical Aspects of Earth Observation Operations

Optimization of satellite operations may affect technical as well as economical aspects of the mission. Technical improvement in satellite operations can increase the return of data or reduce the mission operation costs. It can be assumed that increasing satellite autonomy is the future tendency for operations as many operations aspects (scheduling, check of satellite health or payload data) can be automated.

With a look to the preceding chapters the question is: Is the advantage of on-board scheduling significant for overall mission operations and the related operations costs?

Optimization of satellite resource usage is usually a trade-off between additional cost of implementation for the technology and the benefit for example in the form of additional usable imagery. The problem of cloud coverage as simulated for EnMAP in Chapter 5 may serve as an example.

Additional cost of on-board scheduling implementation is driven mainly by the development and integration of the software. Exact numbers for the labor expenses may be difficult. It is assumed that the overall effort would not be greater than 1 year of labor for one experienced team member with software and operations knowledge.

For estimation of the on-board scheduling benefit it is useful to include a list containing the range of satellite imagery prices, giving no exact values but a range showing the order of magnitude. Satellite imagery is priced based on different quality attributes as:

- area of acquisition
- post processing stage of the image
- spectral range and geometric resolution
- price policy of the supplier

- acquisiton time (age of information)

[Kay03] gives a list of prices for the year 2000. Current prices (which are at similar levels) are collected in Table 6.1[1].

Table 6.1.: Current prices for earth observation imagery

Sensor/Satellite	Scene	Price	Supplier
SPOT 2/4	60x60 km	1900 US$	ScanEx
ETM+ (LandSat-7)	185x185 km	1100 US$	ACRES
LISS-3 (IRS-1C/1D)	140x140 km	2400 US$	Euromap

Example prices range from 0.03 US$ - 0.53 US$ per km^2. Achievable prices for EnMAP can only be estimated as such hyperspectral sensor data is currently not available on the market. It should be assumed that a price of 0.30 US$ per km^2 can be achieved. This does not take into account that the same data may be sold several times to different customers (which would increase the image value).

Simulation showed additional cloud free image area for the scenarios with longer gaps in GS coverage (Scenario 3 over Asia before contact over Neustrelitz GS). This additional area may vary depending on the weather and the different orders from the user community. It should be assumed that the average daily benefit is 6% (simulation showed values between 6% and 44%). This value is the minimum of the simulated scenarios and may not directly apply to other EO missions. It's expected that this increased amount of additional imagery (or higher values) can also be achieved by other missions which can use on-board scheduling (see Chapter 6.4).

If 6% are assumed, the regular raise is an additional cloud free area of $2 \times 10^3\,\text{km}^2$ per day. This would add up to an area of $3650 \times 10^3\,\text{km}^2$ over the whole planned mission life time (5 years). In this case benefits ($\approx$ 1 Mio. US $) of on-board planning would exceed the expenses ten times compared to implementation costs ($\approx$ 0.1 Mio. US $).

6.2. Earth Observation and On-Board Scheduling: Lessons Learned

During this work it has been suggested to develop and install an on-board scheduling system on a satellite mission. Three different missions have been evaluated for real-usage: BayernSat, BIRD and EnMAP. For different reasons, none of the missions could be used for deployment

---

[1] More information can be found on the websites of the supplier in [Eur09],[Sca09] and [ACR09]

of an on-board scheduling system and experiments. Experience gained should be concluded as ”lessons learned” from these three projects.

BayernSat is a satellite project under the leadership development of Technische Universität München. The satellite is equipped with an optical camera and a PowerPC, an ideal platform for on-board scheduling experiments. The project is currently delayed due to financial problems. Advantageous concerning the implementation of on-board scheduling was that all the required information would have been available in a relatively small university team. Furthermore the satellite would have been dedicated to technology demonstration which means reduced limitations for the use of new technologies.

The second project was BIRD. BIRD is a satellite build an operated solely by DLR. The satellite has been developed as an experimental infrared imaging satellite. Launched in 2001 the satellite is still operational. But aging hardware led to different failures. Only reduced operations are conducted during the last years, therefore BIRD would have been an interesting satellite as there was no use for this satellite anymore. Difficulties occurred as the original satellite development team is not available anymore including detailed technical information on the hard- and software of the satellite which is required for implementation of on-board scheduling.

The EnMAP project has been used as a reference satellite throughout this work. It has been recommended to use on-board scheduling for this project. A drawback from the standpoint of on-board scheduling has been the fact that EnMAP is designed as a regular EO mission, not as a technology demonstration mission. Furthermore EnMAP has been already in project phase B (preliminary design phase of a satellite mission according to ECSS standard) when the integration of an on-board planning system has been suggested. Thsi led to the decision not to change the existing design in such a late project phase.

Nevertheless on-board scheduling can also be ”upgraded” for an existing satellite mission by up-load of new software to the satellite computers. Sufficient memory and computation power is required as well as interfaces between the different software modules.

Based on the work in different satellite projects it can be concluded that for the application of on-board scheduling:

- a detailed analysis (and simulation) of the possible advantage should be made
- discussion and integration should start as early as possible, during design and development of space segment as well as ground segment (phase A of the project)
- the satellite software should be designed in such a way that conventional and on-board scheduling operations can be used as necessary
- the satellite RTOS should not be affected by on-board scheduling failures except reduced performance, installation on a separate processor is for security and performance reasons strongly preferred

6.3. Advantages of Autonomous Mission Scheduling

Both solutions, the on-board scheduling and the on-ground scheduling solution, have advantages and disadvantages. This chapter should be used to list and compare both. The biggest advantage of the on-board solution is the possibility of an immediate reaction to the current state of the satellite during payload operations. This can increase the amount of useful payload data returned by the satellite. Furthermore the scheduling process on ground is not necessary if everything works as expected on-board. This can potentially reduce operations effort in the ground control center, as manual labor is reduced.

Interactive on-board scheduling has also some disadvantages with the most important one being the loss of control about some satellite activities by the control center. Execution of datatakes and commanding is at least partly done on-board driven by unforeseeable boundary conditions. During each GS contact the satellite will be found in a not exactly anticipated state by ground. If life-limited items are used during payload operations a tracking of the operations time is only possible after datatake execution. A further drawback is the additional effort for software development and qualification for the satellite. Furthermore safety mechanisms must be developed and tested to assure that on-board commanding does not harm the satellite in any way.

Conclusion: The overall driver for use of on-board scheduling will be the additional return of payload data - either used by the scientific community or as an commercial return during selling of data. If the value of additional data exceeds the implementation effort significantly this technology will be used increasingly in future satellite missions.

6.4. Feasibility Flow Chart for On-Board Scheduling

The results of the preceding chapters can be used to development a flow chart (see Figure 6.4). It is used for estimation of of on-board scheduling the feasibility and based on the questions as discussed in this work.

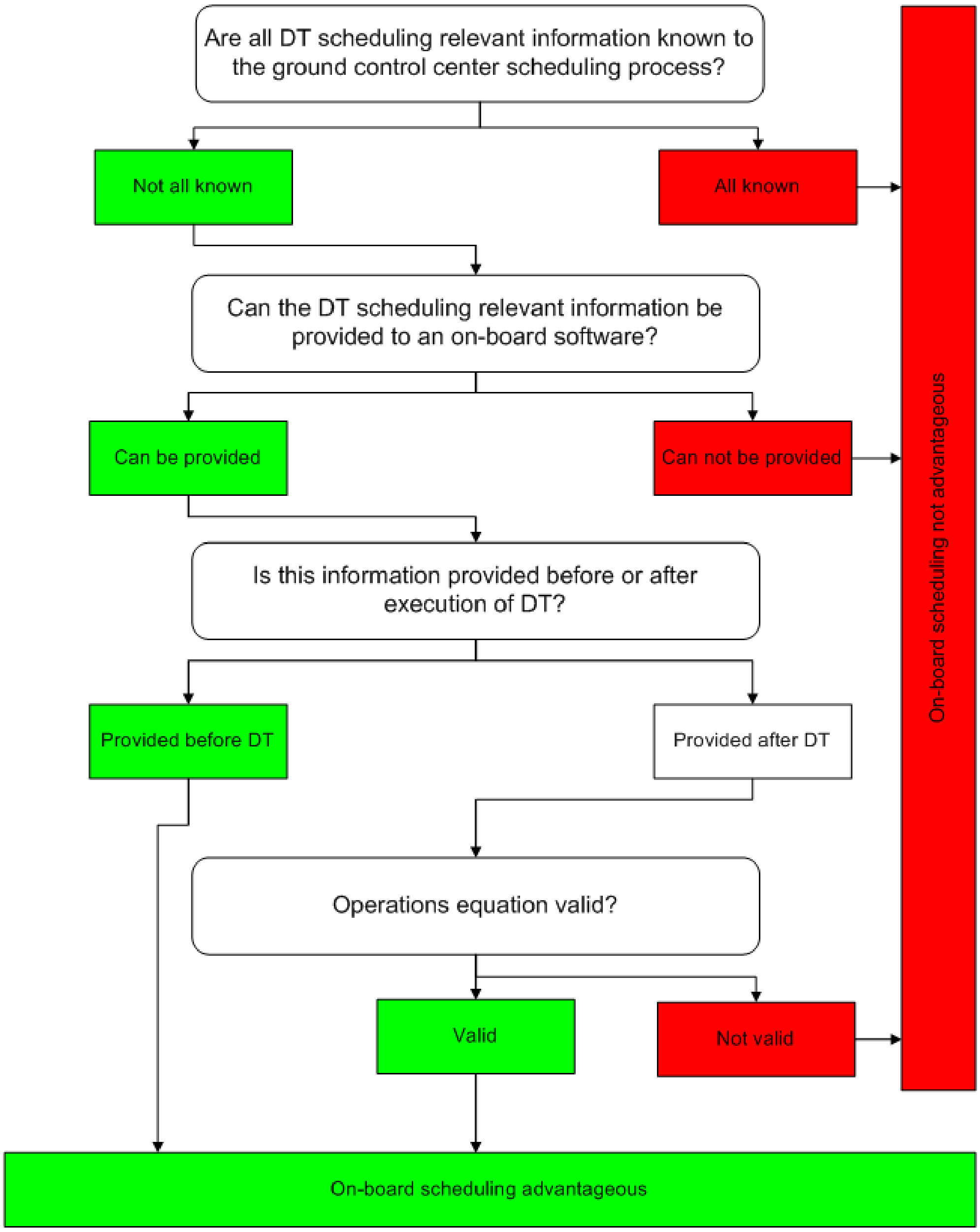


Figure 6.1.: Estimation of on-board scheduling feasibility

The start is at the top of the figure and subsequently the different scheduling relevant aspects are checked if they are fulfilled or not. If the flow ends on the green path it can be advantageous to use on-board scheduling for the mission. This does not imply that it may be economic reasonable. This can only be estimated after simulation of the mission. The operations relevant equations can be found in Chapter 4.2.

7. Conclusion

7.1. Conclusion of the Results

Acquisition scheduling for EO satellites is done in advance (several days or hours before acquisition) and on ground, usually in a ground control center. As a result a sequence of telecommands is send to the satellite, stored and executed based on timestamps.

This work focused on on-board scheduling in conjunction with the common acquisition scheduling problem for earth observing satellites. The basic assumption is that the return of useful imaging data can be increased by on-board scheduling. Chapter 2 describes the conventional way of mission operations and mission planning for EO satellites. Operations limitations result from

- decoupled scheduling and execution of the acquisitions
- relatively short contacts times over ground stations for satellites in a low earth orbit
- the down-link data rate is low compared to the source date rate of optical sensors

Chapter 3 identified potential EO acquisition problems where on-board scheduling can be used in a potentially beneficial way. The first identified important problem is cloud coverage in satellite images. Rossow and Schiffer state in [RS99] an average cloud coverage of the landmasses by 58.4%. Values for certain target areas and seasons vary but generally more than half of the earths surface is blocked for acquisition with optical sensors. These acquisitions are potentially unusable data for the satellite operator. A second application area are satellites using lossless data compression. Compressed size of images depends on the amount of information contained in the raw data. For lossless data compression it is therefore not know how much of the mass memory unit is used by the stored images. This may lead to unused mass memory. For this on-board scheduling appplication a patent has been filed [AW09]. Further potential application areas exist but are of minor importance for daily operations and are therefore not discussed in more detail.

For the simulation of on-board scheduling a greedy algorithm has been developed for usage with an existing DLR orbit simulation program. This algorithm is based on the one used already by the mission planning group of GSOC, scheduling the acquisitions based on a first come first serve basis starting with DT of highest priority and the earliest start time.

The problem of cloud coverage has been addressed in Chapter 5 for simulation of an EO satellite mission based on the system characteristics of the EnMAP satellite. Potential cloud data sources have been identified an evaluated for simulation use. It was found that global numerical cloud coverage data as provided by weather service providers would fit best for simulations, providing continuous temporal and spatial data. Raw data provided by ECMWF has been used for the simulation. Furthermore an artificial user scenario is generated to simulate the acquisition orders. Randomized values based on a rectangular distribution are used for start time, stop time and duration of orders. A normal distribution is used for the generation of a random roll angle.

Existing software of the GSOC Mission Planning group has been used for simulation of on-board scheduling. Some additional software modules have been added for this purpose, including processing of cloud data and the greedy scheduling algorithm.

In a first step the general impact of cloud coverage on the EnMAP mission is simulated for selected durations. Simulation showed cloud coverage between 57 % - 67 % for the seleted mission setup which matches quite well the statements in scientific literature.

In a second step, different user scenarios have been used to simulate an operational EO satellite mission. User scenarios have been loaded and scheduled, acquired DT have been selected by the A_{cc}/A quality criteria. It has been shown that for a post-pass constellation (constellation B in Chapter 4.1 with technical setup like EnMAP) on-board scheduling is beneficial. An additional cloud free area could be acquired in the range from 6% and 44% for selected scenarios.

With the prices for commercial satellite imagery in mind it can be assumed that the value of additional imagery can easily sum up to more than a million $ per satellite mission. It is expected that the commercial advantages together with increasing on-board processing capabilities pave the way for this type of operations in future EO missions.

7.2. Outlook for Future Work

Future work may be related to the fast on-board classification of image quality criteria. This requires the simulation of instrument raw data together with the on-board identification of the interesting quality information. Processing must be fast enough to allow scheduling activities in between planned acquisitions. In a next step a full system consisting of a satellite simulator with its on-board computers and optical sensor can be simulated. This should be used for development of a flight experiment for a technology demonstration mission. A principle advantage of this new way of satellite operations is the possibility to execute this experiment, on-board scheduling and data classification, in parallel to conventional operations. So the experiment can be executed for phases of only several orbits or days in order collect information about the system and potential error sources during operations.

Bibliography

[ACR09] National Earth Observation Group ACRES. *http://www.ga.gov.au/remote-sensing/get-satellite-imagery-data/ordering/pricing/index.jsp*, August 2009.

[AGG01] T. Arvidson, J. Gasch, and S.N. Goward. *Landsat 7's Long-term Acquisition Plan - an Innovative Approach to Building a Global Imagery Archive*. pages 13–26, 2001.

[Alg03] T. Algra. *Real-time Cloud Sensing for Efficiency Improvement of Optical High-resolution Satellite Remote Sensing*. In *Geoscience and Remote Sensing Symposium*, San Francisco, 2003. IGARSS.

[AW09] R. Axmann and M. Wickler. *Verfahren sowie Vorrichtung zum Aufnehmen und Speichern von Daten in einem Raumfahrzeug*. Patent, 2009. Patent Number: 10 2007 059 511, Deutsches Zentrum für Luft- und Raumfahrt e.V.,51147 Köln.

[Bar98] Hans-Jochen Bartsch. *Taschenbuch mathematischer Formeln*. Fachbuchverlag Leipzig im Carl-Hanser-Verlag, 18th edition, 1998. ISBN 3-446-19396-0.

[Ber00] J. Bermyn. *PROBA - PRoject for On-Board Autonomy*. *Air and Space Europe*, 2:70–76, 2000. issue 1.

[BF04] E. Borg and B. Fichtelmann. *Verfahren und Vorrichtung zum Feststellen einer Nutzbarkeit von Fernerkundungsdaten*. Patent, 2004. 10 2004 024 595.9, Deutsches Zentrum für Luft- und Raumfahrt.

[BF05] E. Borg and B. Fichtelmann. *Verfahren und Anordnung zum Gewinnen von Fernerkundungsdaten*. Patent, 2005. 10 2005 010 155.0, Deutsches Zentrum für Luft- und Raumfahrt.

[Bru07] Peter Brucker. *Scheduling Algorithms*. Springer, 5th edition, 2007. ISBN 978-3540695158.

[BVA$^+$96] E. Bensana, G. Verfaillie, J. Agnese, N. Bataille, and D. Blumstein. *Exact and Inexact Methods for the Daily Management of an Earth Observation Satellite*. In *In Proc. 4 th Intl. Symposium on Space Mission Operations and Ground Data Systems*, Munich, Germany, 1996.

[Eur09] Euromap. Homepage: *http://www.euromap.de/products/prod_001.html*, August 2009.

[Fer08a] W. Ferster. Kayser threde to build mapping satellite for dlr. *Space News*, 19(45), 2008.

[Fer08b] W. Ferster. *Launch of 29 EO Satellites Expected in Next 10 Years*, 2008.

[FJM01] J. Frank, A. Jonsson, and R. Morris. *Planning and Scheduling for Fleets of Earth Observing Satellites.* In *In Proc. 6th Int. Symposium on AI, Robotics and Automation for Space*, 2001.

[fMRWF08] European Centre for Medium-Range Weather Forecasts. DVD, May 2008. World-wide Cloud Coverage Forecast Model, 0.5° Grid,1992-1996,FM-92 VIII GRIB Ed.1.

[fNeV83] Deutsches Institut für Normung e. V. *Information Processing; Graphical Symbols and Their Application.* Norm, 1983. DIN 66001.

[FS95] Peter Fortescue and John Stark. *Spacecraft Systems Engineering.* Wiley, 2nd edition, 1995. ISBN 0471619515.

[fSDSC] Consultative Committee for Space Data Systems (CCSDS). Homepage: *http://www.ccsds.org*. Last access: February 27th 2007.

[Gar96] J.T. Garner, editor. *Satellite Control - A Comprehensive Approach.* Wiley, Chichester, 2nd edition, 1996. ISBN 0-471-96254-6.

[GC00] J. R. Gasch and K. A. Campana. *Cloud Cover Avoidance In Space-Based Remote Sensing Acquisitions.* In *Algorithms for Multispectral, Hyperspectral and Ultraspectral Imagery VI.*, volume SPIE Vol. 4049, pages 336–347, 2000.

[GCLP02] Al Globus, James Crawford, Jason Lohn, and Anna Pryor. *Scheduling Earth Observing Satellites Using Evolutionary Algorithms: Problem Description and Approach*, 2002.

[GNT04] M. Ghallab, D. Nau, and P. Traverso, editors. *Automated Planning - Theory and Practice.* Elsevier, San Francisco, 2004. ISBN 1-55860-856-7.

[HP99] S. Harrison and M. Price. *Task Scheduling for Satellite Based Imagery.* In *Proceedings of the Eighteenth Workshop of the UK Planning and Scheduling Special Interest Group*, pages pages 64–78, University of Salford, UK, December 1999.

[HR06] S. Harris and J. Ross, editors. *Beginning Algorithm.* Wiley Publishing, Inc., Indianapolis, Indiana, 2006. ISBN 978-0-7645-9674-2.

[J$^+$02] A. Jonsson et al. *Advances in Planning and Scheduling of Remote Sensing Instruments For Fleets of Earth Observing Satellites.* In *Integrated Remote Sensing at the Global, Regional and Local Scale*, Denver, CO USA, 10-15 November 2002.

[K+03] L. Khatib et al. *Interleaved Observation Execution and Rescheduling on Earth Observing Systems*, 2003. ICAPS-03 Workshop on Plan Execution, Trento, Italy.

[Kay03] Hakan Kayal. *Auslegung und Betrieb einer dedizierten low-cost Bodenstation für Fernerkundungsaufgaben*. PhD thesis, Technische Universität Berlin, 2003.

[Kra02] Herbert J. Kramer. *Observation of the Earth and its Environment. Survey of Missions and Sensors*. Springer-Verlag GmbH, 2002. ISBN-3540423885.

[LB96] W.J. Larson and D.G. Boden, editors. *Cost-Effective Space Mission Operations*. McGraw-Hill, New York, 1996. ISBN 0-07-006382-6.

[LRS01] C. E. Leiserson, R. L. Rivest, and C. Stein, editors. *Introduction to Algorithms*. MIT Press, 2nd edition, 2001. ISBN 978-0262032933.

[LW99] W.J. Larson and J.R. Wertz, editors. *Space Mission Analysis And Design*. Microcosm Press / Springer, El Segundo/New York, 3rd edition, 1999. ISBN 1-881883-10-8.

[LWH07] Wilfried Ley, Klaus Wittmann, and Willi Hallmann. *Handbuch der Raumfahrttechnik*. Hanser Fachbuch, 2007. ISBN 3446411852.

[Mal94] H. Malberg. *Meteorologie und Klimatologie - Eine Einführung*. Springer-Verlag, Berlin/Heidelberg/New York, 2nd edition, 1994. ISBN 3-540-57178-7.

[Mas94] H. Masaru. *Image Pickup System with an Image Pickup Device for Control*. Patent, 1994. US 5,353,055, NEC Corporation, Tokyo, Japan.

[MG05] Oliver Montenbruck and Eberhard Gill. *Satellite Orbits: Models, Methods and Applications*. Springer, 2005. ISBN 9783540672807.

[MGB+08] E. Maurer, M. Geyer, A. Braun, F. Mrowka, C. Lenzen, Y. Wasser, and A. Köhler. *TerraSAR-X Short Notice Planning*. In *Proceeding Space Ops 2008*, Heidelberg, 2008. AIAA.

[MMGB03] D. Mandl, J. Miller, M. Griffin, and H. Burke. *Validation of Onboard Cloud Cover Assessment Using EO-1*. Technical report, NASA, Goddard Space Flight Center, June 2003.

[Org94] World Meteorological Organisation. *Manual on Codes*, 1994. http://www.wmo.ch/pages/prog/www/WMOCodes/GRIB.html.

[Pea94] C. B. Pease. *Satellite Imaging Instruments: Principles, Technologies and Operational Systems (Space Science)*. John Wiley and Sons, 1994. ISBN 0471954330.

[PG98] W. Potter and J. Gasch. *A Photo Album of Earth: Scheduling LANDSAT 7 Mission Daily Activities*. In *Proceedings of the International Symposium Space Mission Operations and Ground Data Systems*, 1998.

[PG02] J.C. Pemberton and L.G. Greenwald. *On the Need For Dynamic Scheduling of Imaging Satellites*. In *Integrated Remote Sensing at the Global, Regional and Local Scale*, Denver, CO USA, 10-15 November 2002.

[RS99] W. B. Rossow and R. A. Schiffer. *Advances in Understanding Clouds from ISCCP*. *Bulletin of the American Meteorological Society*, 80 Issue 11:2261–2287, 1999.

[Say06] Khalid Sayood. *Introduction to Data Compression (The Morgan Kaufmann Series in Multimedia Information and Systems)*. Morgan Kaufmann, 3rd edition, 2006. ISBN 978-0-12-620862-7.

[Sca09] ScanEx. *http://catalog.scanex.ru*, August 2009.

[SCT$^+$07] R. Sherwood, S. Chien, D. Tran, B. Cichy, R. Castano, A. Davies, and G. Rabideau. *The EO-1 Autonomous Sciencecraft*. In *21st Annual Conference on Small Satellites*, Logan, UT USA, 13-16 August 2007.

[Sim99] D.E. Simon. *An Embedded Software Primer*. Addison-Wesley, Boston, 1999. ISBN 0-201-61569-X.

[SKH$^+$07] T. Stuffler, C. Kaufmann, S. Hofer, K.P. Forster, G. Schreier, A. Mueller, A. Eckardt, H. Bach, B. Penne, U. Benz, and R. Haydn. *The EnMAP Hyperspectral Imager An Advanced Optical Payload for Future Applications in Earth Observation Programmes*. In *Acta AstronauticaVolume 61, Issues 1-6*, June-August 2007.

[SO02] M. Spigai and G. Oller. *On-Board Change Detection With Neural Networks*. In *Integrated Remote Sensing at the Global, Regional and Local Scale*, Denver, CO USA, 10-15 November 2002.

[Val01] D.A. Vallado. *Fundamentals of Astrodynamics and Applications*. Microcosm Press, El Segundo, 2001. ISBN 1-881883-12-4.

[VCHK06] David A. Vallado, Paul Crawford, Richard Hujsak, and T. S. Kelso. *Revisiting Spacetrack Report No. 3*, 2006.

[vdLRdC$^+$99] N. van der Linden, H. Roozekrans, W. Som de Cerff, S. van der Veen, and R. Beck. *Cloud Forecasting for Earth Observation Satellites*, 1999. BCRS Report USP-2 98-20,ISBN 90- 5411-261-1.

[Wei05] A. Weiner. *Onboard Satellite Processing to Mitigate Data Recorder Limits*. Patent, 2005. US 2005/0071054 A1, Sacco & Associates, PA, Florida, US.

[WL91] S. D. Wall and K.W. Ledbetter. *Design of Mission Operations Systems for Scientific Remote Sensing*. Taylor Francis, London/Washington DC, 1991. ISBN 0-85066-860-3.

[WZ96] M. Wickler and G. Zoeschinger. *Mission Planning for MOMS-2P*. In *SpaceOps 1996*, page SO96.3.04, September 1996.

[ZF94] M. Zweben and M.S. Fox. *Intelligent Scheduling*. Morgan Kaufmann Publishers, San Francisco, 1994. ISBN 1-55860-260-7.

[ZK02] G. Zhou and M. Kafatos. *Future Intelligent Earth Observing Satellites*. In *Integrated Remote Sensing at the Global, Regional and Local Scale*, Denver, CO USA, 10-15 November 2002.

A. Annex

A.1. Patent DE 10 2007 059 511

(19)
Bundesrepublik Deutschland
Deutsches Patent- und Markenamt

(10) **DE 10 2007 059 511 B4** 2009.09.17

(12) **Patentschrift**

(21) Aktenzeichen: **10 2007 059 511.7**
(22) Anmeldetag: **11.12.2007**
(43) Offenlegungstag: **02.07.2009**
(45) Veröffentlichungstag
der Patenterteilung: **17.09.2009**

(51) Int Cl.^{8}: ***H04L 12/06*** (2006.01)
B64G 1/66 (2006.01)

Innerhalb von drei Monaten nach Veröffentlichung der Patenterteilung kann nach § 59 Patentgesetz gegen das Patent Einspruch erhoben werden. Der Einspruch ist schriftlich zu erklären und zu begründen. Innerhalb der Einspruchsfrist ist eine Einspruchsgebühr in Höhe von 200 Euro zu entrichten(§ 6 Patentkostengesetz in Verbindung mit der Anlage zu § 2 Abs. 1 Patentkostengesetz).

(73) Patentinhaber:
Deutsches Zentrum für Luft- und Raumfahrt e.V., 51147 Köln, DE

(74) Vertreter:
Patentanwälte von Kreisler, Selting, Werner et col., 50667 Köln

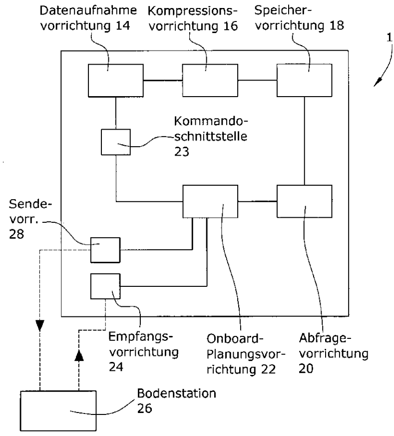
(72) Erfinder:
Axmann, Robert, 81373 München, DE; Wickler, Martin, Dr., 80636 München, DE

(56) Für die Beurteilung der Patentfähigkeit in Betracht gezogene Druckschriften:

US	**55 96 494**	**A**
US	**2005/00 71 054**	**A1**

(54) Bezeichnung: **Verfahren sowie Vorrichtung zum Aufnehmen und Speichern von Daten in einem Raumfahrzeug**

(57) Hauptanspruch: Verfahren zum Aufnehmen und Speichern von Daten in einem Raumfahrzeug (10) mit den Schritten:
a) Abfragen der freien Speicherkapazität einer Speichervorrichtung (18) des Raumfahrzeugs (10) durch eine Onboard-Planungsvorrichtung (22) im Raumfahrzeug (10) und
b) Ausgeben eines Befehls durch die Onboard-Planungsvorrichtung (22) zum Aufnehmen von Daten durch eine Datenaufnahmevorrichtung (14), sofern die Abfrage der Speichervorrichtung (18) durch die Onboard-Planungsvorrichtung (22) ergeben hat, dass genügend freie Speicherkapazität für die Aufnahme von Daten zur Verfügung steht
gekennzeichnet durch die anschließenden Schritte:
c) Aufnehmen von Daten durch die Datenaufnahmevorrichtung (14)
d) Komprimieren der aufgenommenen Daten und
e) Speichern der komprimierten Daten in der Speichervorrichtung (18) des Raumfahrzeugs (10)



A.2. Listing of Used C++ Source Code

This listing describes the used scheduling as implemented for simulation, Chapter 4.4 describes this in pidgin code while here the use C++ code is listed.

Greedy Scheduling Algorithm

Listing A.1: Greedy Scheduling Algorithm

```cpp
void Scheduler::startScheduling(Container* CurCont,Schedule* CurSched,
    SatelliteModel* CurSat,int nScheduleIndex, double dScheduleStart,
    double dScheduleEnd)
{
        int nPrio = 1;
        int nCounter = 0;
        int nNewContInd = -1;
        int nNewSchedInd = -1;
        double dStartAfter =0;
        bool bExceedOrbRes = false;
        bool bExceedAbsRes = false;
        time_t tStart;
        tStart = dScheduleStart;

        nCounter=nScheduleIndex;          // Initialize scheduler index
            should and start time
        dStartAfter = dScheduleStart;
        CurCont->unschedAllDT();          // Mark all DT as unscheduled
        CurSched->emptySchedule(nScheduleIndex); // Delete remaining
            schedule before new scheduler run

        while(CurSched->nNumElements < MAX_NUM_ELEMENTS) // Fill schedule
        {
                nNewContInd = -1;
                nNewSchedInd = -1;
                while((nNewContInd == -1) && (nPrio >= 0)) // Find next
                    possible DT
                {
                        nNewContInd = Scheduler::findNextDT(CurCont,
                            dStartAfter,nPrio);
                        if (nNewContInd == -1)
                        {
                                nPrio--;
                                                        // Select next lower DT
                                    priority
                                dStartAfter = dScheduleStart;    // Set
                                    search time to schedule start
                        }
                }
                if ((nNewContInd == -1) && (nPrio <= 0)) break; // No more
                    DT to schedule because lowest prio reached for search
```

```
                // Try to place DT in schedule
                if ((Scheduler::findOverlay(CurSat,CurSched,CurCont,
                    nNewContInd)) != true)
                {
                        nNewSchedInd = Scheduler::findDTPosition(CurSched,
                            CurCont,nNewContInd);
                }
                if ((nNewSchedInd != -1) && (nNewContInd !=-1))
                {
                        Scheduler::addDT(CurSat,CurSched,CurCont,
                            nNewSchedInd,nNewContInd);
                        CurSched->nNumElements++;
                        bExceedOrbRes = Scheduler::checkOrbitUsage(CurSat,
                            CurSched,CurCont,nNewContInd);
                        bExceedAbsRes = Scheduler::checkResUsage(CurSat,
                            CurSched,CurCont,nScheduleIndex);

                        if ((!bExceedOrbRes) && (!bExceedAbsRes))
                        {
                                dStartAfter = CurCont->requestArray[
                                    nNewContInd].dEnd + CurSat->
                                    nPointingDelay;
                                CurCont->requestArray[nNewContInd].
                                    bScheduled = true;
                        }
                        else
                        {
                                Scheduler::removeDT(CurSat,CurSched,
                                    CurCont,nNewSchedInd,nNewContInd);
                                CurSched->nNumElements--;
                        }
                }
                if (dStartAfter >= dScheduleEnd) break;
        }
        printf(”Finisch␣Scheduler\n”);
};
```

Following programming code is used for the roll-angle value generation of the artificial user scenarios. A normal distribution function is used.

Rollangle Generation with Normal Distribution

Listing A.2: Rollangle Generation with Normal Distribution

```
double Mission::NormalDistribution(double mean,double sigma,int iMaxRoll)
{
        double ran_equal, temp;
        const double PI = 3.1416593;
        double dTempResult;
```

```
        int iResult;
        ran_equal = rand();
        temp = rand();
        ran_equal /= 0x7fff;
        temp /= 0x7fff;
        if (ran_equal < 1E-200) ran_equal = 1E-200;
        dTempResult =  cos(temp * 2 * PI) * sigma * sqrt(-2 * log(
            ran_equal)) + mean;
        if (dTempResult < -3) dTempResult = -3;
        if (dTempResult > 3) dTempResult = 3;
        dTempResult /= 3;
        iResult = int(dTempResult*iMaxRoll);
        return iResult;
};
```

Calculation of Cloud Coverage

Cloud coverage is calculated using the four neighboring points of the source data. Coverage values are than interpolated bi-linear for the requested point.

Listing A.3: Cloud Coverage Calculation

```
bool CloudCoverage::calculateCoverage(time_t timestamp, double grid,double
    * tmplat, double* tmplon,FILE* cloudfile,double* area, double* areacc)
{
        bool bNextPoint = true;
        double tmp_area = 0;
        double tmp_areacc = 0;
        double tmp_area_segment = 0;
        double tmp_area_segment_cc = 0;
        double curlat = 0;
        double curlon = 0;
        double dCloudCov = 0;

        // Define two arrays keeping latitude and longitude values
            immediate
        double locallat[4];
        double locallon[4];

        // Copy values
        locallat[0] = tmplat[0];
        locallat[1] = tmplat[1];
        locallat[2] = tmplat[2];
        locallat[3] = tmplat[3];
        locallon[0] = tmplon[0];
        locallon[1] = tmplon[1];
        locallon[2] = tmplon[2];
        locallon[3] = tmplon[3];

        bool atSeam = false;
```

```
        int nAdjustDeg = 0;
        atSeam = checkSeam(locallon);
        if (atSeam) nAdjustDeg = adjustLong(locallon);

        // Sort given 4 points in the expected precedence
        sortPoints(locallat,locallon);

        // Initialize first point
        curlat = locallat[0];
        curlon = locallon[0];

        int counter = 0;
        while(bNextPoint)
        {
                if (atSeam) curlon = calculateLong(curlon,(nAdjustDeg *
                    -1));
                tmp_area_segment = calculateArea(curlat,curlon,grid);
                tmp_area_segment_cc = tmp_area_segment *
                    averageCloudCoverage(curlat,curlon, timestamp,cloudfile
                    );
                tmp_area += tmp_area_segment;
                tmp_areacc += tmp_area_segment_cc;
                if (atSeam) curlon = calculateLong(curlon,(nAdjustDeg));
                bNextPoint = getNextPoint(&curlat,&curlon,grid,locallat,
                    locallon);
                counter++;
        }
        *area = tmp_area;
        *areacc = tmp_areacc;
        return true;
};
```

Listing A.4: Area Calculation

```
double CloudCoverage::calculateArea(double dTmpLatDeg,double dTmpLonDeg,
    double dGridDeg)
{
        double v_length;
        double h_length;
        double dLat1Rad,dLat2Rad,dLon1Rad,dLon2Rad;

        // Horizontal lenght of box
        dLat1Rad = dTmpLatDeg * PI/180;
        dLat2Rad = dTmpLatDeg * PI/180;

        dLon1Rad = (dTmpLonDeg - dGridDeg / 2) * PI/180;
        dLon2Rad = (dTmpLonDeg + dGridDeg / 2) * PI/180;

        h_length = acos(sin(dLat1Rad)*sin(dLat2Rad) + cos(dLat1Rad)*cos(
            dLat2Rad)*cos(dLon2Rad-dLon1Rad));
        h_length = Earthsphere*h_length;

        // Vertical length of box
```

```
        dLon1Rad = dTmpLonDeg * PI/180;
        dLon2Rad = dTmpLonDeg * PI/180;

        dLat1Rad = (dTmpLatDeg - dGridDeg / 2) * PI/180;
        dLat2Rad = (dTmpLatDeg + dGridDeg / 2) * PI/180;

        v_length = acos(sin(dLat1Rad)*sin(dLat2Rad) + cos(dLat1Rad)*cos(
            dLat2Rad)*cos(dLon2Rad-dLon1Rad));
        v_length = Earthsphere*v_length;

        return (h_length*v_length);
}
```

A.3. Userscenarios

ID	Start	t[s]	α	Prio
0	Jan 01 02:33:12 2008	23	0	0
1	Jan 01 02:33:18 2008	16	3	0
2	Jan 01 02:33:17 2008	22	5	0
3	Jan 01 02:33:34 2008	5	-2	1
4	Jan 01 02:42:14 2008	5	-12	1
5	Jan 01 02:42:14 2008	5	-13	1
6	Jan 01 02:42:19 2008	5	-5	1
7	Jan 01 02:42:19 2008	5	-9	0
8	Jan 01 02:42:25 2008	9	11	1
9	Jan 01 02:42:27 2008	6	10	1
10	Jan 01 02:42:27 2008	7	-2	1
11	Jan 01 02:42:24 2008	10	4	0
12	Jan 01 02:42:41 2008	9	-7	1
13	Jan 01 02:42:40 2008	8	-4	1
14	Jan 01 02:42:42 2008	6	10	0
15	Jan 01 02:42:56 2008	12	-4	1
16	Jan 01 02:42:55 2008	8	-6	1
17	Jan 01 02:42:56 2008	6	-10	0
18	Jan 01 02:43:12 2008	5	-7	0
19	Jan 01 02:43:12 2008	5	7	0
20	Jan 01 02:43:44 2008	11	8	0
21	Jan 01 02:43:44 2008	8	1	1
22	Jan 01 02:43:44 2008	11	5	1
23	Jan 01 02:44:59 2008	5	5	0
24	Jan 01 02:44:59 2008	5	-6	0
25	Jan 01 02:45:14 2008	7	8	1
26	Jan 01 02:45:11 2008	11	0	0
27	Jan 01 02:45:11 2008	9	7	1
28	Jan 01 02:47:02 2008	5	-6	0
29	Jan 01 02:47:02 2008	5	0	0
30	Jan 01 02:47:09 2008	5	-8	0
31	Jan 01 02:47:09 2008	5	13	1
32	Jan 01 02:48:45 2008	135	-6	1
33	Jan 01 02:48:03 2008	27	-5	1
34	Jan 01 02:50:36 2008	29	-9	1
35	Jan 01 02:47:42 2008	136	7	0
36	Jan 01 02:48:17 2008	6	-7	0
37	Jan 01 02:48:41 2008	139	18	1
38	Jan 01 02:49:04 2008	50	-9	1
39	Jan 01 02:49:39 2008	52	-3	0
40	Jan 01 02:48:01 2008	5	-12	1
41	Jan 01 02:47:49 2008	75	7	1
42	Jan 01 03:06:09 2008	27	-15	1
43	Jan 01 03:06:27 2008	9	0	0
44	Jan 01 03:05:41 2008	95	5	0
45	Jan 01 03:04:15 2008	136	4	0
46	Jan 01 03:04:17 2008	94	-3	1
47	Jan 01 03:01:10 2008	140	5	0
48	Jan 01 03:06:40 2008	31	18	0
49	Jan 01 03:03:37 2008	117	-22	1
50	Jan 01 03:01:51 2008	44	-18	1
51	Jan 01 03:02:49 2008	117	8	0

Table A.1.: Scenario 1 - DT details

DTs for scenario 2 listed in a detailed form.

ID	Start	t[s]	α	Prio
0	Jan 01 09:01:00 2008	19	7	1
1	Jan 01 09:00:58 2008	35	-3	0
2	Jan 01 09:01:10 2008	40	-8	0
3	Jan 01 09:01:16 2008	38	-19	1
4	Jan 01 09:02:38 2008	72	-16	1
5	Jan 01 09:04:56 2008	54	-8	1
6	Jan 01 09:03:17 2008	142	-1	1
7	Jan 01 09:04:28 2008	60	14	1
8	Jan 01 09:03:04 2008	18	0	1
9	Jan 01 09:04:07 2008	73	-14	0
10	Jan 01 09:05:25 2008	22	0	1
11	Jan 01 09:03:57 2008	124	-13	1
12	Jan 01 09:14:55 2008	117	1	1
13	Jan 01 09:11:00 2008	101	-5	1
14	Jan 01 09:08:03 2008	73	3	1
15	Jan 01 09:14:21 2008	13	1	0
16	Jan 01 09:14:16 2008	118	-13	1
17	Jan 01 09:20:20 2008	77	4	1
18	Jan 01 09:14:59 2008	46	-4	0
19	Jan 01 09:08:38 2008	114	10	1
20	Jan 01 09:18:55 2008	133	8	1

continued on next page

continued from previous page

21	Jan 01 09:15:23 2008	22	2	1
22	Jan 01 09:10:58 2008	70	-13	0
23	Jan 01 09:18:33 2008	125	-7	1
24	Jan 01 09:10:29 2008	89	-3	0
25	Jan 01 09:14:14 2008	124	2	0
26	Jan 01 09:15:59 2008	143	-3	0
27	Jan 01 09:10:27 2008	103	-9	0
28	Jan 01 09:17:29 2008	107	6	0
29	Jan 01 09:12:05 2008	144	1	1
30	Jan 01 09:12:20 2008	64	-15	1
31	Jan 01 09:07:13 2008	26	-12	0
32	Jan 01 09:11:21 2008	88	19	1
33	Jan 01 09:21:18 2008	64	0	0
34	Jan 01 09:23:06 2008	5	-1	0
35	Jan 01 09:23:06 2008	5	18	0
36	Jan 01 09:34:41 2008	118	-7	1
37	Jan 01 09:33:41 2008	73	-1	0
38	Jan 01 09:35:06 2008	62	-7	1
39	Jan 01 09:34:24 2008	100	-8	0
40	Jan 01 09:33:44 2008	137	-5	0
41	Jan 01 09:38:13 2008	15	-22	0
42	Jan 01 09:38:13 2008	14	0	0
43	Jan 01 09:38:16 2008	5	8	1
44	Jan 01 09:38:22 2008	9	-2	0

Table A.2.: Scenario 2 - DT details

DTs for scenario 3 listed in a detailed form.

ID	Start	t[s]	α	Prio
0	Jan 01 02:33:13 2008	23	-7	0
1	Jan 01 02:33:13 2008	13	-4	1
2	Jan 01 02:33:22 2008	7	-9	0
3	Jan 01 02:33:32 2008	5	12	0
4	Jan 01 02:33:10 2008	22	0	1
5	Jan 01 02:33:15 2008	8	-6	1
6	Jan 01 02:33:23 2008	13	-5	0
7	Jan 01 02:33:13 2008	19	-4	1
8	Jan 01 02:33:10 2008	28	13	1
9	Jan 01 02:33:10 2008	28	12	1
10	Jan 01 02:42:14 2008	3	-7	0
11	Jan 01 02:42:14 2008	4	-4	1
12	Jan 01 02:42:14 2008	5	-9	0
13	Jan 01 02:42:14 2008	5	12	0
14	Jan 01 02:42:14 2008	3	0	1
15	Jan 01 02:42:14 2008	5	-6	1
16	Jan 01 02:42:14 2008	4	-5	0
17	Jan 01 02:42:14 2008	3	-4	1
18	Jan 01 02:42:14 2008	2	13	1
19	Jan 01 02:42:14 2008	2	12	1
20	Jan 01 02:42:20 2008	1	-7	0
21	Jan 01 02:42:19 2008	4	-4	1
22	Jan 01 02:42:19 2008	5	-9	0
23	Jan 01 02:42:19 2008	5	12	0
24	Jan 01 02:42:19 2008	2	0	1
25	Jan 01 02:42:19 2008	4	-6	1
26	Jan 01 02:42:20 2008	4	-5	0
27	Jan 01 02:42:19 2008	2	-4	1
28	Jan 01 02:42:20 2008	1	13	1
29	Jan 01 02:42:19 2008	1	12	1
30	Jan 01 02:42:24 2008	9	-7	0
31	Jan 01 02:42:24 2008	7	-4	1
32	Jan 01 02:42:26 2008	5	-9	0
33	Jan 01 02:42:29 2008	5	12	0
34	Jan 01 02:42:24 2008	9	0	1
35	Jan 01 02:42:25 2008	6	-6	1
36	Jan 01 02:42:27 2008	7	-5	0
37	Jan 01 02:42:24 2008	8	-4	1
38	Jan 01 02:42:24 2008	10	13	1
39	Jan 01 02:42:24 2008	10	12	1
40	Jan 01 02:42:40 2008	9	-7	0
41	Jan 01 02:42:40 2008	6	-4	1
42	Jan 01 02:42:42 2008	5	-9	0
43	Jan 01 02:42:44 2008	5	12	0
44	Jan 01 02:42:40 2008	8	0	1
45	Jan 01 02:42:41 2008	6	-6	1
46	Jan 01 02:42:42 2008	6	-5	0
47	Jan 01 02:42:40 2008	8	-4	1
48	Jan 01 02:42:40 2008	9	13	1
49	Jan 01 02:42:40 2008	9	12	1
50	Jan 01 02:42:55 2008	11	-7	0
51	Jan 01 02:42:55 2008	7	-4	1
52	Jan 01 02:42:58 2008	6	-9	0

continued on next page

continued from previous page

53	Jan 01 02:43:01 2008	5	12	0
54	Jan 01 02:42:55 2008	10	0	1
55	Jan 01 02:42:56 2008	6	-6	1
56	Jan 01 02:42:59 2008	7	-5	0
57	Jan 01 02:42:56 2008	9	-4	1
58	Jan 01 02:42:55 2008	12	13	1
59	Jan 01 02:42:55 2008	12	12	1
60	Jan 01 02:43:12 2008	5	-7	0
61	Jan 01 02:43:12 2008	5	-4	1
62	Jan 01 02:43:12 2008	5	-9	0
63	Jan 01 02:43:12 2008	5	12	0
64	Jan 01 02:43:12 2008	5	0	1
65	Jan 01 02:43:12 2008	5	-6	1
66	Jan 01 02:43:12 2008	5	-5	0
67	Jan 01 02:43:12 2008	5	-4	1
68	Jan 01 02:43:12 2008	5	13	1
69	Jan 01 02:43:12 2008	5	12	1
70	Jan 01 02:43:44 2008	9	-7	0
71	Jan 01 02:43:44 2008	7	-4	1
72	Jan 01 02:43:46 2008	5	-9	0
73	Jan 01 02:43:49 2008	5	12	0
74	Jan 01 02:43:44 2008	9	0	1
75	Jan 01 02:43:45 2008	6	-6	1
76	Jan 01 02:43:47 2008	7	-5	0
77	Jan 01 02:43:44 2008	8	-4	1
78	Jan 01 02:43:44 2008	10	13	1
79	Jan 01 02:43:44 2008	10	12	1
80	Jan 01 02:44:59 2008	5	-7	0
81	Jan 01 02:44:59 2008	5	-4	1
82	Jan 01 02:44:59 2008	5	-9	0
83	Jan 01 02:44:59 2008	5	12	0
84	Jan 01 02:44:59 2008	5	0	1
85	Jan 01 02:44:59 2008	5	-6	1
86	Jan 01 02:44:59 2008	5	-5	0
87	Jan 01 02:44:59 2008	5	-4	1
88	Jan 01 02:44:59 2008	5	13	1
89	Jan 01 02:44:59 2008	5	12	1
90	Jan 01 02:45:11 2008	11	-7	0
91	Jan 01 02:45:11 2008	7	-4	1
92	Jan 01 02:45:14 2008	6	-9	0
93	Jan 01 02:45:17 2008	5	12	0

continued on next page

continued from previous page

94	Jan 01 02:45:11 2008	10	0	1
95	Jan 01 02:45:12 2008	6	-6	1
96	Jan 01 02:45:15 2008	7	-5	0
97	Jan 01 02:45:12 2008	9	-4	1
98	Jan 01 02:45:11 2008	12	13	1
99	Jan 01 02:45:11 2008	12	12	1
100	Jan 01 02:47:02 2008	4	-7	0
101	Jan 01 02:47:02 2008	4	-4	1
102	Jan 01 02:47:02 2008	5	-9	0
103	Jan 01 02:47:02 2008	5	12	0
104	Jan 01 02:47:02 2008	4	0	1
105	Jan 01 02:47:02 2008	5	-6	1
106	Jan 01 02:47:02 2008	4	-5	0
107	Jan 01 02:47:02 2008	4	-4	1
108	Jan 01 02:47:02 2008	3	13	1
109	Jan 01 02:47:02 2008	3	12	1
110	Jan 01 02:47:10 2008	1	-7	0
111	Jan 01 02:47:09 2008	4	-4	1
112	Jan 01 02:47:09 2008	5	-9	0
113	Jan 01 02:47:09 2008	5	12	0
114	Jan 01 02:47:09 2008	2	0	1
115	Jan 01 02:47:09 2008	4	-6	1
116	Jan 01 02:47:10 2008	4	-5	0
117	Jan 01 02:47:09 2008	2	-4	1
118	Jan 01 02:47:10 2008	1	13	1
119	Jan 01 02:47:09 2008	1	12	1
120	Jan 01 02:48:34 2008	105	-7	0
121	Jan 01 02:47:59 2008	47	-4	1
122	Jan 01 02:49:46 2008	15	-9	0
123	Jan 01 02:51:27 2008	7	12	0
124	Jan 01 02:47:32 2008	100	0	1
125	Jan 01 02:48:22 2008	24	-6	1
126	Jan 01 02:50:14 2008	46	-5	0
127	Jan 01 02:48:23 2008	82	-4	1
128	Jan 01 02:48:12 2008	130	13	1
129	Jan 01 02:47:49 2008	131	12	1
130	Jan 01 03:04:39 2008	105	12	0
131	Jan 01 03:02:35 2008	64	4	0
132	Jan 01 03:04:31 2008	15	14	0
133	Jan 01 03:04:43 2008	81	30	1
134	Jan 01 03:04:37 2008	8	-19	0

continued on next page

continued from previous page

135	Jan 01 03:05:53 2008	71	3	0
136	Jan 01 03:03:09 2008	59	4	1
137	Jan 01 03:03:06 2008	76	-9	0
138	Jan 01 03:04:54 2008	126	-8	0
139	Jan 01 03:02:42 2008	63	-1	0
140	Jan 01 04:11:41 2008	105	12	0
141	Jan 01 04:09:21 2008	64	4	0
142	Jan 01 04:11:26 2008	15	14	0
143	Jan 01 04:11:43 2008	81	30	1
144	Jan 01 04:11:32 2008	8	-19	0
145	Jan 01 04:12:59 2008	71	3	0
146	Jan 01 04:09:58 2008	59	4	1
147	Jan 01 04:09:56 2008	76	-9	0
148	Jan 01 04:11:59 2008	126	-8	0
149	Jan 01 04:09:29 2008	63	-1	0
150	Jan 01 04:14:53 2008	3	12	0
151	Jan 01 04:14:52 2008	4	4	0
152	Jan 01 04:14:52 2008	5	14	0
153	Jan 01 04:14:53 2008	3	30	1
154	Jan 01 04:14:52 2008	5	-19	0
155	Jan 01 04:14:53 2008	4	3	0
156	Jan 01 04:14:52 2008	4	4	1
157	Jan 01 04:14:52 2008	3	-9	0
158	Jan 01 04:14:54 2008	2	-8	0
159	Jan 01 04:14:52 2008	4	-1	0
160	Jan 01 04:16:23 2008	103	12	0
161	Jan 01 04:16:15 2008	62	4	0
162	Jan 01 04:17:03 2008	14	14	0
163	Jan 01 04:16:37 2008	80	30	1
164	Jan 01 04:17:07 2008	8	-19	0
165	Jan 01 04:16:59 2008	70	3	0
166	Jan 01 04:16:24 2008	58	4	1
167	Jan 01 04:16:19 2008	75	-9	0
168	Jan 01 04:16:10 2008	124	-8	0
169	Jan 01 04:16:17 2008	62	-1	0
170	Jan 01 04:19:55 2008	18	12	0
171	Jan 01 04:19:54 2008	12	4	0
172	Jan 01 04:20:01 2008	6	14	0
173	Jan 01 04:19:57 2008	15	30	1
174	Jan 01 04:20:01 2008	5	-19	0
175	Jan 01 04:20:00 2008	13	3	0
176	Jan 01 04:19:55 2008	12	4	1
177	Jan 01 04:19:55 2008	14	-9	0
178	Jan 01 04:19:54 2008	20	-8	0
179	Jan 01 04:19:55 2008	12	-1	0
180	Jan 01 04:20:36 2008	62	12	0
181	Jan 01 04:20:31 2008	38	4	0
182	Jan 01 04:20:59 2008	10	14	0
183	Jan 01 04:20:44 2008	48	30	1
184	Jan 01 04:21:01 2008	7	-19	0
185	Jan 01 04:20:56 2008	43	3	0
186	Jan 01 04:20:36 2008	36	4	1
187	Jan 01 04:20:33 2008	46	-9	0
188	Jan 01 04:20:28 2008	74	-8	0
189	Jan 01 04:20:32 2008	38	-1	0
190	Jan 01 04:22:05 2008	2	12	0
191	Jan 01 04:22:03 2008	3	4	0
192	Jan 01 04:22:03 2008	5	14	0
193	Jan 01 04:22:04 2008	3	30	1
194	Jan 01 04:22:03 2008	5	-19	0
195	Jan 01 04:22:04 2008	3	3	0
196	Jan 01 04:22:03 2008	3	4	1
197	Jan 01 04:22:03 2008	3	-9	0
198	Jan 01 04:22:05 2008	2	-8	0
199	Jan 01 04:22:03 2008	3	-1	0
200	Jan 01 04:40:01 2008	105	5	1
201	Jan 01 04:41:18 2008	80	-4	0
202	Jan 01 04:42:40 2008	14	-3	1
203	Jan 01 04:42:30 2008	15	9	1
204	Jan 01 04:39:51 2008	57	-4	1
205	Jan 01 04:41:44 2008	119	5	1
206	Jan 01 04:39:48 2008	72	12	0
207	Jan 01 04:41:42 2008	71	1	0
208	Jan 01 04:41:05 2008	123	0	1
209	Jan 01 04:41:01 2008	135	-2	1
210	Jan 01 05:45:55 2008	105	5	1
211	Jan 01 05:47:59 2008	80	-4	0
212	Jan 01 05:49:51 2008	14	-3	1
213	Jan 01 05:49:37 2008	15	9	1
214	Jan 01 05:45:37 2008	57	-4	1
215	Jan 01 05:48:55 2008	119	5	1
216	Jan 01 05:45:33 2008	72	12	0

continued on next page

continued from previous page

217	Jan 01 05:48:36 2008	71	1	0
218	Jan 01 05:47:48 2008	123	0	1
219	Jan 01 05:47:46 2008	135	-2	1
220	Jan 01 05:55:46 2008	38	5	1
221	Jan 01 05:55:53 2008	30	-4	0
222	Jan 01 05:56:09 2008	8	-3	1
223	Jan 01 05:56:08 2008	8	9	1
224	Jan 01 05:55:46 2008	22	-4	1
225	Jan 01 05:55:50 2008	43	5	1
226	Jan 01 05:55:46 2008	27	12	0
227	Jan 01 05:55:56 2008	27	1	0
228	Jan 01 05:55:48 2008	44	0	1
229	Jan 01 05:55:47 2008	48	-2	1
230	Jan 01 06:16:54 2008	105	5	1
231	Jan 01 06:18:26 2008	80	-4	0
232	Jan 01 06:19:58 2008	14	-3	1
233	Jan 01 06:19:47 2008	15	9	1
234	Jan 01 06:16:40 2008	57	-4	1
235	Jan 01 06:19:02 2008	119	5	1
236	Jan 01 06:16:37 2008	72	12	0
237	Jan 01 06:18:54 2008	71	1	0
238	Jan 01 06:18:14 2008	123	0	1
239	Jan 01 06:18:10 2008	135	-2	1
240	Jan 01 07:24:51 2008	105	-13	1
241	Jan 01 07:25:09 2008	97	4	0
242	Jan 01 07:26:17 2008	13	8	0
243	Jan 01 07:24:40 2008	90	14	0
244	Jan 01 07:25:14 2008	106	19	0
245	Jan 01 07:24:16 2008	25	-11	0
246	Jan 01 07:26:08 2008	85	-3	1
247	Jan 01 07:25:37 2008	65	16	1
248	Jan 01 07:26:41 2008	119	6	0
249	Jan 01 07:25:26 2008	67	17	1
250	Jan 01 07:29:53 2008	24	-13	1
251	Jan 01 07:29:54 2008	23	4	0
252	Jan 01 07:30:03 2008	7	8	0
253	Jan 01 07:29:53 2008	21	14	0
254	Jan 01 07:29:53 2008	24	19	0
255	Jan 01 07:29:54 2008	9	-11	0
256	Jan 01 07:29:57 2008	20	-3	1
257	Jan 01 07:29:57 2008	16	16	1
258	Jan 01 07:29:54 2008	27	6	0
259	Jan 01 07:29:56 2008	17	17	1
260	Jan 01 07:30:26 2008	1	-13	1
261	Jan 01 07:30:26 2008	2	4	0
262	Jan 01 07:30:25 2008	5	8	0
263	Jan 01 07:30:25 2008	2	14	0
264	Jan 01 07:30:26 2008	1	19	0
265	Jan 01 07:30:25 2008	4	-11	0
266	Jan 01 07:30:26 2008	2	-3	1
267	Jan 01 07:30:26 2008	3	16	1
268	Jan 01 07:30:28 2008	1	6	0
269	Jan 01 07:30:26 2008	3	17	1
270	Jan 01 07:30:28 2008	4	-13	1
271	Jan 01 07:30:28 2008	4	4	0
272	Jan 01 07:30:28 2008	5	8	0
273	Jan 01 07:30:28 2008	4	14	0
274	Jan 01 07:30:28 2008	4	19	0
275	Jan 01 07:30:28 2008	5	-11	0
276	Jan 01 07:30:28 2008	4	-3	1
277	Jan 01 07:30:28 2008	5	16	1
278	Jan 01 07:30:28 2008	4	6	0
279	Jan 01 07:30:28 2008	5	17	1
280	Jan 01 07:37:27 2008	1	-13	1
281	Jan 01 07:37:27 2008	2	4	0
282	Jan 01 07:37:26 2008	5	8	0
283	Jan 01 07:37:26 2008	2	14	0
284	Jan 01 07:37:27 2008	1	19	0
285	Jan 01 07:37:26 2008	4	-11	0
286	Jan 01 07:37:27 2008	2	-3	1
287	Jan 01 07:37:27 2008	3	16	1
288	Jan 01 07:37:29 2008	1	6	0
289	Jan 01 07:37:27 2008	3	17	1
290	Jan 01 07:37:45 2008	1	-13	1
291	Jan 01 07:37:45 2008	2	4	0
292	Jan 01 07:37:44 2008	5	8	0
293	Jan 01 07:37:44 2008	2	14	0
294	Jan 01 07:37:45 2008	1	19	0
295	Jan 01 07:37:44 2008	4	-11	0
296	Jan 01 07:37:45 2008	2	-3	1
297	Jan 01 07:37:45 2008	3	16	1
298	Jan 01 07:37:47 2008	1	6	0

continued on next page

continued from previous page

299	Jan 01 07:37:45 2008	3	17	1
300	Jan 01 07:56:56 2008	105	-13	1
301	Jan 01 07:57:10 2008	97	4	0
302	Jan 01 07:58:13 2008	13	8	0
303	Jan 01 07:56:50 2008	90	14	0
304	Jan 01 07:57:13 2008	106	19	0
305	Jan 01 07:56:34 2008	25	-11	0
306	Jan 01 07:57:57 2008	85	-3	1
307	Jan 01 07:57:35 2008	65	16	1
308	Jan 01 07:58:14 2008	119	6	0
309	Jan 01 07:57:27 2008	67	17	1
310	Jan 01 08:00:40 2008	8	-13	1
311	Jan 01 08:00:40 2008	8	4	0
312	Jan 01 08:00:42 2008	5	8	0
313	Jan 01 08:00:40 2008	7	14	0
314	Jan 01 08:00:40 2008	8	19	0
315	Jan 01 08:00:40 2008	6	-11	0
316	Jan 01 08:00:41 2008	7	-3	1
317	Jan 01 08:00:41 2008	7	16	1
318	Jan 01 08:00:40 2008	8	6	0
319	Jan 01 08:00:41 2008	7	17	1
320	Jan 01 09:01:08 2008	45	-3	0
321	Jan 01 09:01:04 2008	49	-5	1
322	Jan 01 09:01:26 2008	8	-13	0
323	Jan 01 09:01:04 2008	12	3	0
324	Jan 01 09:01:48 2008	9	9	1
325	Jan 01 09:01:21 2008	32	-2	0
326	Jan 01 09:01:02 2008	43	-19	0
327	Jan 01 09:01:16 2008	27	9	1
328	Jan 01 09:01:01 2008	50	-7	0
329	Jan 01 09:00:58 2008	59	-9	0
330	Jan 01 09:04:04 2008	105	-3	0
331	Jan 01 09:03:37 2008	114	-5	1
332	Jan 01 09:04:31 2008	13	-13	0
333	Jan 01 09:03:03 2008	23	3	0
334	Jan 01 09:06:01 2008	14	9	1
335	Jan 01 09:04:46 2008	73	-2	0
336	Jan 01 09:03:12 2008	98	-19	0
337	Jan 01 09:04:09 2008	59	9	1
338	Jan 01 09:03:19 2008	116	-7	0
339	Jan 01 09:03:24 2008	139	-9	0
340	Jan 01 09:17:18 2008	105	-3	0
341	Jan 01 09:14:45 2008	114	-5	1
342	Jan 01 09:15:36 2008	13	-13	0
343	Jan 01 09:09:20 2008	23	3	0
344	Jan 01 09:22:11 2008	14	9	1
345	Jan 01 09:19:28 2008	73	-2	0
346	Jan 01 09:11:17 2008	98	-19	0
347	Jan 01 09:15:27 2008	59	9	1
348	Jan 01 09:12:42 2008	116	-7	0
349	Jan 01 09:14:36 2008	139	-9	0
350	Jan 01 09:23:08 2008	1	-3	0
351	Jan 01 09:23:08 2008	1	-5	1
352	Jan 01 09:23:06 2008	5	-13	0
353	Jan 01 09:23:06 2008	4	3	0
354	Jan 01 09:23:06 2008	5	9	1
355	Jan 01 09:23:08 2008	3	-2	0
356	Jan 01 09:23:06 2008	2	-19	0
357	Jan 01 09:23:07 2008	3	9	1
358	Jan 01 09:23:07 2008	1	-7	0
359	Jan 01 09:23:08 2008	0	-9	0
360	Jan 01 09:34:56 2008	105	-3	0
361	Jan 01 09:34:30 2008	114	-5	1
362	Jan 01 09:35:25 2008	13	-13	0
363	Jan 01 09:34:01 2008	23	3	0
364	Jan 01 09:36:51 2008	14	9	1
365	Jan 01 09:35:37 2008	73	-2	0
366	Jan 01 09:34:08 2008	98	-19	0
367	Jan 01 09:35:03 2008	59	9	1
368	Jan 01 09:34:14 2008	116	-7	0
369	Jan 01 09:34:17 2008	139	-9	0
370	Jan 01 09:38:15 2008	15	-3	0
371	Jan 01 09:38:14 2008	16	-5	1
372	Jan 01 09:38:20 2008	6	-13	0
373	Jan 01 09:38:14 2008	7	3	0
374	Jan 01 09:38:25 2008	6	9	1
375	Jan 01 09:38:19 2008	12	-2	0
376	Jan 01 09:38:14 2008	14	-19	0
377	Jan 01 09:38:17 2008	10	9	1
378	Jan 01 09:38:14 2008	16	-7	0
379	Jan 01 09:38:13 2008	18	-9	0

Table A.3.: Scenario 3 - DT details

A.4. Simulation Results

Executed DT of the different scenario 3 simulations.

Executed DT - $A_{cc}/A \leq 1$						
DT-ID	$A[km^2]$	$A_{cc}[km^2]$	$[\%_{cc}]$	$t_{DT}[s]$	Prio	Deleted
3	4331.30	2100.65	0.48	26.00	1	No
4	830.41	785.97	0.95	5.00	1	No
12	840.59	747.09	0.89	6.00	1	No
19	831.05	618.31	0.74	5.00	1	No
20	1684.38	1304.20	0.77	10.00	1	No
26	2025.64	1413.92	0.70	12.00	1	No
31	879.25	717.63	0.82	5.00	1	No
35	9153.33	8008.68	0.87	56.00	1	No
60	2998.96	1868.67	0.62	18.00	1	No
68	662.95	341.06	0.51	5.00	1	No
74	1697.10	1369.98	0.81	11.00	1	No
78	1161.96	898.54	0.77	7.00	1	No
84	7540.90	4726.75	0.63	44.00	1	No
91	6968.31	4499.54	0.65	40.00	1	No
96	1581.38	1025.99	0.65	9.00	1	No
116	14866.19	8624.27	0.58	89.00	1	No
128	4297.45	2984.91	0.69	26.00	1	No
136	1212.38	773.35	0.64	7.00	1	No
142	895.82	650.13	0.73	5.00	1	No
143	858.93	720.18	0.84	5.00	1	No
155	1180.31	826.78	0.70	7.00	1	No
159	4250.98	2291.49	0.54	23.00	1	No
165	1001.87	263.48	0.26	7.00	1	No
166	1505.20	717.15	0.48	9.00	1	No

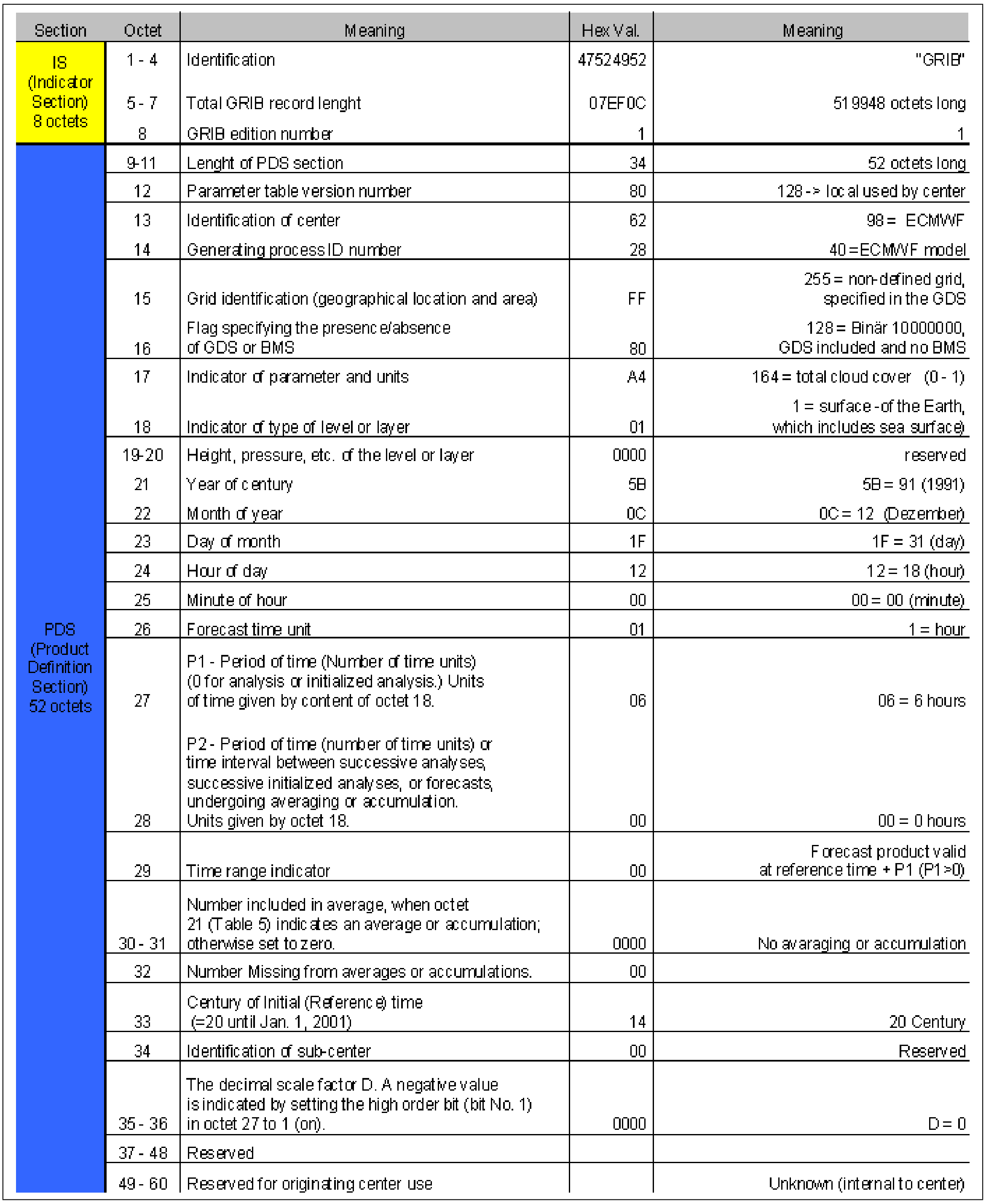
Executed DT - $A_{cc}/A < 0.7$						
DT-ID	$A[km^2]$	$A_{cc}[km^2]$	$[\%_{cc}]$	$t_{DT}[s]$	Prio	Deleted
3	4331.30	2100.65	0.48	26.00	1	No
4	830.41	785.97	0.95	5.00	1	Yes
12	840.59	747.09	0.89	6.00	1	Yes
19	831.05	618.31	0.74	5.00	1	Yes
20	1684.38	1304.20	0.77	10.00	1	Yes
26	2025.64	1413.92	0.70	12.00	1	No
31	879.25	717.63	0.82	5.00	1	Yes
35	9153.33	8008.68	0.87	56.00	1	Yes
60	2998.96	1868.67	0.62	18.00	1	No
68	662.95	341.06	0.51	5.00	1	No
74	1697.10	1369.98	0.81	11.00	1	Yes
78	1161.96	898.54	0.77	7.00	1	Yes
84	7540.90	4726.75	0.63	44.00	1	No
91	6968.31	4499.54	0.65	40.00	1	No

96	1581.38	1025.99	0.65	9.00	1	No
116	14866.19	8624.27	0.58	89.00	1	No
128	4297.45	2984.91	0.69	26.00	1	No
136	1212.38	773.35	0.64	7.00	1	No
142	895.82	650.13	0.73	5.00	1	Yes
143	858.93	720.18	0.84	5.00	1	Yes
147	8590.01	4240.59	0.49	52.00	1	No
155	1179.67	811.04	0.69	7.00	1	No
159	4250.98	2291.49	0.54	23.00	1	No
165	1001.87	263.48	0.26	7.00	1	No
166	1505.20	717.15	0.48	9.00	1	No
190	3399.00	2598.96	0.76	20.00	1	Yes
195	892.28	736.75	0.83	5.00	0	Yes
197	2036.19	882.96	0.43	13.00	1	No

Executed DT - $A_{cc}/A < 0.6$						
DT-ID	$A[km^2]$	$A_{cc}[km^2]$	$[\%_{cc}]$	$t_{DT}[s]$	Prio	Deleted
3	4331.30	2100.65	0.48	26.00	1	No
4	830.41	785.97	0.95	5.00	1	Yes
12	840.59	747.09	0.89	6.00	1	Yes
19	831.05	618.31	0.74	5.00	1	Yes
20	1684.38	1304.20	0.77	10.00	1	Yes
26	2025.64	1413.92	0.70	12.00	1	Yes
31	879.25	717.63	0.82	5.00	1	Yes
35	9153.33	8008.68	0.87	56.00	1	Yes
60	2998.96	1868.67	0.62	18.00	1	Yes
68	662.95	341.06	0.51	5.00	1	No
74	1697.10	1369.98	0.81	11.00	1	Yes
78	1161.96	898.54	0.77	7.00	1	Yes
84	7540.90	4726.75	0.63	44.00	1	Yes
91	6968.31	4499.54	0.65	40.00	1	Yes
96	1581.38	1025.99	0.65	9.00	1	Yes
116	14866.19	8624.27	0.58	89.00	1	No
128	4297.45	2984.91	0.69	26.00	1	Yes
136	1212.38	773.35	0.64	7.00	1	Yes
142	895.82	650.13	0.73	5.00	1	Yes
143	858.93	720.18	0.84	5.00	1	Yes
147	8590.01	4240.59	0.49	52.00	1	No
155	1179.67	811.04	0.69	7.00	1	Yes
159	4250.98	2291.49	0.54	23.00	1	No
165	1001.87	263.48	0.26	7.00	1	No
166	1505.20	717.15	0.48	9.00	1	No
190	3399.00	2598.96	0.76	20.00	1	Yes
195	892.28	736.75	0.83	5.00	0	Yes
197	2036.19	882.96	0.43	13.00	1	No

Table A.4.: Simulation details - scenario 3

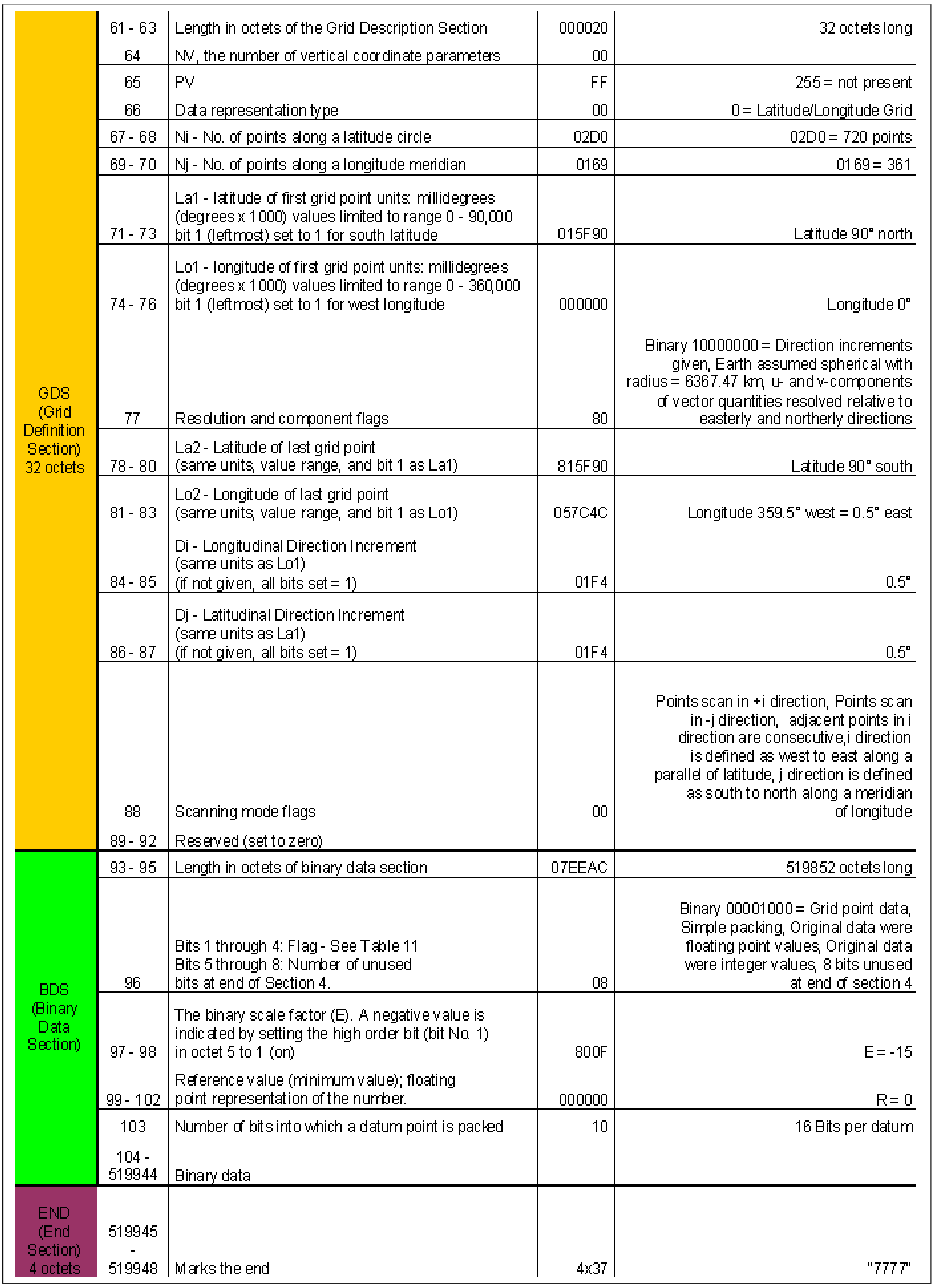
A.5. GRIB Header



Section	Octet	Meaning	Hex Val.	Meaning
IS (Indicator Section) 8 octets	1 - 4	Identification	47524952	"GRIB"
	5 - 7	Total GRIB record lenght	07EF0C	519948 octets long
	8	GRIB edition number	1	1
PDS (Product Definition Section) 52 octets	9-11	Lenght of PDS section	34	52 octets long
	12	Parameter table version number	80	128 -> local used by center
	13	Identification of center	62	98 = ECMWF
	14	Generating process ID number	28	40 =ECMWF model
	15	Grid identification (geographical location and area)	FF	255 = non-defined grid, specified in the GDS
	16	Flag specifying the presence/absence of GDS or BMS	80	128 = Binär 10000000, GDS included and no BMS
	17	Indicator of parameter and units	A4	164 = total cloud cover (0 - 1)
	18	Indicator of type of level or layer	01	1 = surface -of the Earth, which includes sea surface)
	19-20	Height, pressure, etc. of the level or layer	0000	reserved
	21	Year of century	5B	5B = 91 (1991)
	22	Month of year	0C	0C = 12 (Dezember)
	23	Day of month	1F	1F = 31 (day)
	24	Hour of day	12	12 = 18 (hour)
	25	Minute of hour	00	00 = 00 (minute)
	26	Forecast time unit	01	1 = hour
	27	P1 - Period of time (Number of time units) (0 for analysis or initialized analysis.) Units of time given by content of octet 18.	06	06 = 6 hours
	28	P2 - Period of time (number of time units) or time interval between successive analyses, successive initialized analyses, or forecasts, undergoing averaging or accumulation. Units given by octet 18.	00	00 = 0 hours
	29	Time range indicator	00	Forecast product valid at reference time + P1 (P1>0)
	30 - 31	Number included in average, when octet 21 (Table 5) indicates an average or accumulation; otherwise set to zero.	0000	No avaraging or accumulation
	32	Number Missing from averages or accumulations.	00	
	33	Century of Initial (Reference) time (=20 until Jan. 1, 2001)	14	20 Century
	34	Identification of sub-center	00	Reserved
	35 - 36	The decimal scale factor D. A negative value is indicated by setting the high order bit (bit No. 1) in octet 27 to 1 (on).	0000	D = 0
	37 - 48	Reserved		
	49 - 60	Reserved for originating center use		Unknown (internal to center)

Figure A.1.: Structure of a GRIB message (part 1)

Header of the first GRIB message of the ECMWF provided dataset. Sections of the message are identified together with its values and the meaning, showing the used grid and packing type of raw data. Subsequent messages are identical except date and time.



GDS (Grid Definition Section) 32 octets	61 - 63	Length in octets of the Grid Description Section	000020	32 octets long
	64	NV, the number of vertical coordinate parameters	00	
	65	PV	FF	255 = not present
	66	Data representation type	00	0 = Latitude/Longitude Grid
	67 - 68	Ni - No. of points along a latitude circle	02D0	02D0 = 720 points
	69 - 70	Nj - No. of points along a longitude meridian	0169	0169 = 361
	71 - 73	La1 - latitude of first grid point units: millidegrees (degrees x 1000) values limited to range 0 - 90,000 bit 1 (leftmost) set to 1 for south latitude	015F90	Latitude 90° north
	74 - 76	Lo1 - longitude of first grid point units: millidegrees (degrees x 1000) values limited to range 0 - 360,000 bit 1 (leftmost) set to 1 for west longitude	000000	Longitude 0°
	77	Resolution and component flags	80	Binary 10000000 = Direction increments given, Earth assumed spherical with radius = 6367.47 km, u- and v-components of vector quantities resolved relative to easterly and northerly directions
	78 - 80	La2 - Latitude of last grid point (same units, value range, and bit 1 as La1)	815F90	Latitude 90° south
	81 - 83	Lo2 - Longitude of last grid point (same units, value range, and bit 1 as Lo1)	057C4C	Longitude 359.5° west = 0.5° east
	84 - 85	Di - Longitudinal Direction Increment (same units as Lo1) (if not given, all bits set = 1)	01F4	0.5°
	86 - 87	Dj - Latitudinal Direction Increment (same units as La1) (if not given, all bits set = 1)	01F4	0.5°
	88	Scanning mode flags	00	Points scan in +i direction, Points scan in -j direction, adjacent points in i direction are consecutive,i direction is defined as west to east along a parallel of latitude, j direction is defined as south to north along a meridian of longitude
	89 - 92	Reserved (set to zero)		
BDS (Binary Data Section)	93 - 95	Length in octets of binary data section	07EEAC	519852 octets long
	96	Bits 1 through 4: Flag - See Table 11 Bits 5 through 8: Number of unused bits at end of Section 4.	08	Binary 00001000 = Grid point data, Simple packing, Original data were floating point values, Original data were integer values, 8 bits unused at end of section 4
	97 - 98	The binary scale factor (E). A negative value is indicated by setting the high order bit (bit No. 1) in octet 5 to 1 (on)	800F	E = -15
	99 - 102	Reference value (minimum value); floating point representation of the number.	000000	R = 0
	103	Number of bits into which a datum point is packed	10	16 Bits per datum
	104 - 519944	Binary data		
END (End Section) 4 octets	519945 - 519948	Marks the end	4x37	"7777"

Figure A.2.: Structure of a GRIB message (part 2)

Index