POLITECNICO DI MILANO

Facoltà di Ingegneria Aerospaziale

Corso di Laurea in Ingegneria Aerospaziale

Optimization of the SD2 Memory System

Adrià Pacheco Fuentes & Ricard Alvarez Iglesias

3 July 2012

Anno Accademico 2011/12

ABSTRACT

Optimization of the SD2 Memory System

Adrià Pacheco Fuentes & Ricard Alvarez

Dipartimento di Ingegneria Aerospaziale

Tesi di Laurea

Due to the difficulty of communication and the characteristics of the memory buffer, it is necessary to conduct a study to optimize the data that will be loaded into the system. The strategy is to identify patterns in the sequence to execute. Functions have been developed in Matlab to calculate the needed memory according to the length of the chosen patterns. These functions have been analyzed with the Particle Swarm Optimization method to find optimal solutions.

Politecnico di Milano
Dipartimento di Ingegneria Aerospaziale
Viale La Masa 34, 20156 Milano

Corso di Laurea in Ingegneria Aerospaziale

Autori:
Adrià Pacheco Fuentes
Ricard Àlvarez Iglesias

Precettori:
Dr. Pierluigi di Lizia
Dr. Roberto Armellin

Direttore:
Dr. Franco Bernelli

Acknowledgments

The acknowledgments are for everyone that has made possible this experience.

First of all, I would like to thank Politecnico di Milano and Dr. Franco Bernelli for welcoming me with open arms, showing interest at all times and for giving me the opportunity to work on this project.

Second, thanks for the effort to Dr. Pierluigi di Lizia and Dr. Roberto Armellin. I know too well all the work they have had during these months but they have always been able to find time for my questions and deliveries.

I also thank Ricard. This experience has given me the opportunity to meet with him and, after so many hours in the lab and after lived so many situations together, has become more than a companion, a friend.

Above all, I would like to thank my parents for all they have done for me. In this case, I am proud to thank them for the most valuable skill they have given me: my education.

Along my life, they have given me wings to reach all my objectives, obviously in the economic way, but more importantly: they have trained me as a person.

Last but certainly not least, I would like to thank Mireia, my girlfriend. She has always been there, especially in bad times. I thank her for all the spirits she has given to me and, above all, I thank her for her ability to make me happy.

I'm sure I miss someone so thanks to all who have contributed in one way or another on this project.

Adrià Pacheco

Contents

List of Figures

List of tables

1. Introduction

As far as is evidenced, man has sought the answers to his deepest questions by observing the sky. The findings and studies that have been made in this way have helped us know the planet where we live and our situation in the universe. From Greeks to now, through Galileo, Kepler or the Hubble telescope, look out was key to better know the Earth.

The Rosseta Mission is the best example of this strategy. It is know that there are comets that have similar features with the Earth at its younger phase. So, the goal of the mission is to land on a comet surface and make a study about its composition and properties. In this way it could be increased the knowledge about how was the Earth, and furthermore, what were the conditions that allowed the emergence of life on it.

The objective of this project is work on the optimization of one subsystem of this mission. It is known that the space missions have all their resources optimized in order to reach their goals (the power consumption, the heat, the memory, etc.). So, in that case, it is important to work on the optimization of the memory consumption of the SD2 subsystem.

The objective of this subsystem is to drill the cometary soil and to take some samples. In order to do this, there are three perforations to be performed and these perforations are executed with commands that have to be uploaded to the SD2 commands buffer.

Previous studies have shown that there are patterns on the command sequence that are equal on both sampling procedures. Based on this idea, a deeper study is performed in this project. It is started to look for the repeated patterns without taking into account the previous knowledge. For this, a software was implemented which automatically find repeating patterns and optimize their use on the telecommand buffer.

The reason of looking for patterns is that they will be uploaded to the buffer only once and the software will select them from this when will be necessary. In this way the total number of information that should be uploaded decreases. This is a very big advantage taking into account the big distance (about 3 AU) that separates the orbiter of the Rosetta mission from the ground stations placed on the Earth [1]. This supposes a handicap in the communications procedures and it will be better if the amount of data that will be sent is less.

To better understand how this optimization is done, this document describes the important features of the Rosetta mission and its instruments and subsystems. Then the idea of the patterns is introduced. The commands that will be uploaded and the sampling procedure are also described.

Finally the studies performed are shown and the results obtained are analyzed.

2. Rosetta Mission

The Rosetta Mission is the third of ESA's cornerstone missions within the science program "Horizons 2000". The mission was started at November of 1993 and from then a lot of scientists and engineers around the world have participated in it.

In this chapter, a brief summary of the mission is exposed in order to take into account what are its environment and its features.

2.1 Introduction

The new objective that this mission will perform is the injection into an orbit around a comet and the delivery of the Philae lander that will perform for the first time a detailed in-situ investigation of the comet nucleus physical properties.

Rosetta was launched from Kourou (French Guyana) on 2 March 2004 using an Ariane5G+ rocket. The objective of the mission is the comet 67P/Churyumov-Gerasimenko, which will be reached in 2014 after a complex series of space maneuvers [2].

Before the Philae's landing there is a previous phase of close comet observation and investigation using the remote-sensing instruments of the Rosetta orbiter (Figure 1). This phase includes high-resolution global mapping of the nucleus, the selection of a safe and scientifically important site and a correct parameterization of the descent sequence. After this phase, Philae will land at the comet surface in November 2014 at a distance of 3 astronomical units from the Sun.

Figure 1- Rosetta orbiter with the lander Philae attached.

2.1.1 Main Mission Goals

The measurements to be made in support of the objective of this mission are exposed below [1]:

- Global characterization of the nucleus, determination of dynamics properties, surface morphology and composition.
- Determination of the chemical, mineralogical and isotopic composition of volatiles and refractories in a cometary nucleus.
- Determination of the physical properties and interrelation of volatiles and refractories in a cometary nucleus.
- Study of the development of cometary activity and the processes in the surface layer of the nucleus and the inner coma (dust/gas interaction).

For this there are some in-situ analyses that will be performed by Philae:

- Collecting comet's samples.
- Heating samples at different temperature levels.
- Analyzing the volatiles using scientific instruments.

For doing this it is necessary a subsystem that drills the comet soil and distribute the samples to all the instruments. It is the SD2 subsystem that will be analyzed in the following sections.

This information will be analyzed in order to see if there is a relationship between the materials of the outer layers and those of the inner layers [3].

2.1.2 67P/Churyumov-Gerasimenko

The Rosetta Mission was originally designed for landing at the comet 46P/Wirtanen, but there were problems with the Ariane 5 and the mission was delayed. Consequently the target comet was changed to the 67P/Churyumov-Gerasimenko comet that is also a comet that passes often for the inner part of the solar system.

This comet has a particular and unusual story. To 1840 the distance of his perihelia was 4 AU and it was completely invisible from the Earth.

Nowadays, the gravitational force exerted by Jupiter has approached the comet to the Sun, reducing its perihelia distance to 1.29 AU with an orbital period of 6.57 years. In the Table 1 it can be seen its features:

Comet 67P/Churyumov-Gerasimenko	
Diameter of nucleus - estimated (km)	3x5
Rotation period (hours)	~12
Orbital period (years)	6.57
Perihelion distance from Sun (million km)	194 (1.29 AU)
Aphelion distance from Sun (million km)	858 (5.74 AU)
Orbital eccentricity	0.632
Orbital inclination (degrees)	7.12
Year of discovery	1969
Discoverers	Klim Churyumov & Svetlana Gerasimenko

Table 1 - Main features of the comet 67P/Churyumov-Gerasimenko.

It has been possible to study its behavior when it was close to its perihelia and it has been noticed that the nucleus of the comet is active in this phase. This activity of the nucleus has produced explosions and gas emissions. However the rate of dust production is relatively low.

The images from the Hubble Space Telescope that show the comet on its last pass of the perihelia allowed to see the elliptic shape of the nucleus and to estimate its diameter from 3 to 5 kilometers. [2]

2.2 SD2 Instrument

Inside the lander Philae (Figure 2), the SD2 subsystem has a determining role.

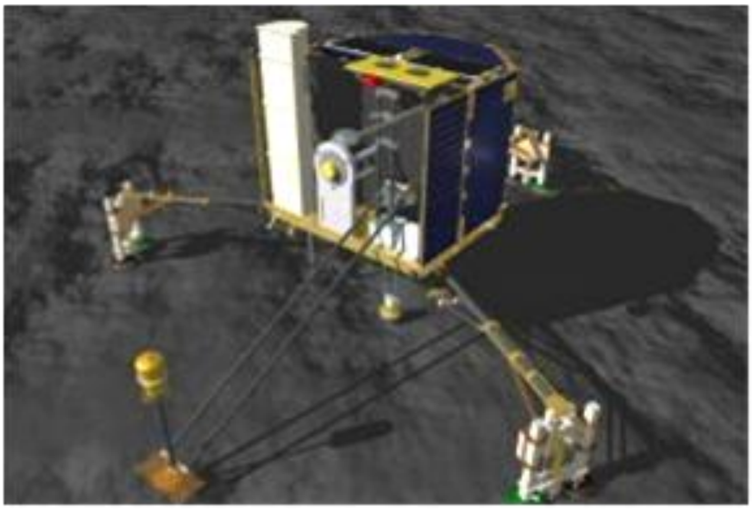


Figure 2 - Lander Philae landed at the comet surface.

Its name comes from Sample, Drill and Distribution and, as it tells, the function of the SD2 is drill the comet soil at different depths. Then the instrument takes the sample and distributes it to the instrument responsible for analyzing the sample. For this purpose there are three scientific instruments that analyze the individual grains [4]:

- COSAC: one of the two evolved gas analyzers onboard Philae, designed to detect and identify complex organic molecules from their elemental and molecular composition [5].
- PTOLEMY: an evolved gas analyzer designed to perform accurate measurements of the isotopic ratios of light elements [6].
- ÇIVA: a visible light microscope coupled to an infrared spectrometer to provide data on the composition, texture and albedo of comet samples [7].

SD2 is mounted on Philae's baseplate and it is equipped with a drill able to collect several samples of 10- 40 mm^3 at different depths (the maximum depth is 230 mm) from the same hole or different ones.

It was designed to operate in critical thermovacuum environment and to meet the demanding mass/power resources limits of Rosetta mission. For this, very innovative technological solutions are adopted [4], [8]:

- Stepper motors.
- Solid and self lubrication.
- Brushless actuation and sensors.
- Low friction/ antijamming approaches.
- Cutting technology (for all range of materials).
- Low power consumption and radiation resistant electronics.
- Special composite material approach.

For the design of the SD2, the parameters that were took into account are the comet soil strength (50 Pa ÷ 50 MPa), the temperature (-140˚ C ÷ +50˚ C) and the pressure along the mission (10^{-5} mbar (space pressure) ÷ 1 bar (Earth's surface pressure)). These are wide in order to assure functioning in presence of high uncertainty [9].

SD2 has a total mass of 5100g and is composed by a mechanical unit (3700g), an electronic unit embedding SD2 software (1000g), and the harness for electrical connection between the mechanical and the electronic units (400g) (Figure 3).

The mechanical unit consists of the Tool Box, the Drill, and the Carousel:

- The Tool box is built in carbon fiber and avoids drill damages due to vibrations and shocks during launch and landing phases.
- The Drill is made of aluminum alloy, and has a radius of 12 mm and a maximum extension of 581.6 mm from the lander balcony; polycrystalline diamonds have been used to reinforce the drill bit for hard soil drilling; position, shape and geometry of the bits have been optimized by theoretical analysis, numerical simulations and experimental tests, in order to maximize the cutting capability with a low vertical trust (100 N) and a low power consumption (20 W is the average during operations). A sampling tube is embedded into the Drill bit and extracted to pick up the sample from the comentary soil. The collection and release of the samples are performed by pressure contact.
- The Carousel is a rotating platform on which some small containers, the ovens, are mounted. The material is picked up and discharged in the ovens. The Carousel task is to position the oven and its sample under the scientific ports of the three analyzers. The ovens provide the interface between the collected sample and the scientific instrument. Two types of ovens were designed in order to agree with the scientific requirements of the mission: there are 10 medium temperature ovens with an optical sapphire prism, suited for the analysis by visible I/R microscopes, before heating up for medium temperature experiment (+180° C), and 16 high temperature ovens for high temperature experiments (+800° C) [10].

The electronic unit is installed into the warm compartment of the lander and incorporates all electronics to control the mechanical unit.

The hardware and software installed provide the interface between the mechanical unit and the lander control system, the Command Data and Management System (CDMS) [11]. SD2 is supplied by Philae's power subsystem with a 28 V line from the lander primary bus, devoted to the mechanical unit, and some auxiliary power lines (±5 V, ±12 V) from the lander secondary converters.

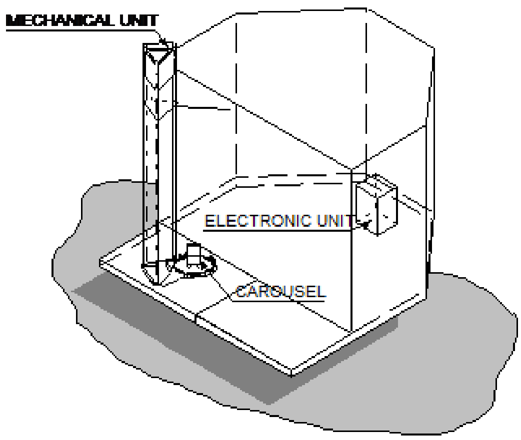


Figure 3 - Mechanical and Electronic Units and Carousel on Philae.

2.3 CDMS

Telecommands dedicated to SD2 are generally laid down in the Stored Telecommand Buffer (STCB) of the Philae Command and Management System (CDMS) and enters SD2 via the software CDMS interface.

The STCB has a capacity of 512 words and different groups of telecommands (also called Mission Plans) can be charged on it as the next Figure illustrates:

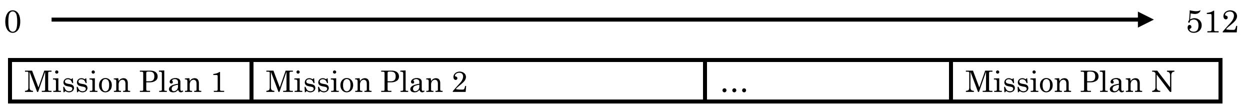


Figure 4 - Telecommand Buffer Charged with Mission Plans.

Once the commands are on the Buffer, the CDMS can execute them as the user programs the sequence. It means, that if the commands are on the buffer, the CDMS is able to call them in the order and the times required.

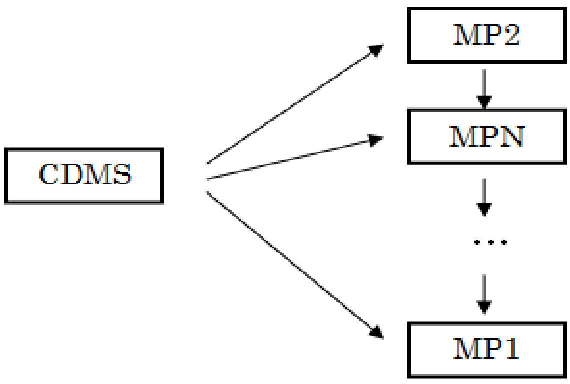


Figure 5 - CDMS tacking Mission Plans from the STCB.

As well as the capacity of the STCB represents a constraint when the Mission Plans are selected, the number of times that the CDMS must contact the buffer (the number of interactions) could also represent a constraint. An amount of time is required each time that the CDMS interact with the buffer to change the MP that is being executed.

3. The First Science Sequence

The First Science Sequence (FSS) contains the set of commands that are performed in order to meet the scientific goals of the mission.

As has been seen in the last section, the SD2 will drill the comet soil and take samples that will be distributed to the scientific instrument for its analysis. It has been defined a baseline to be performed in order to take three samples at different depths and to distribute them. The Table 2 shows the characteristics of each sample:

Sample	Depth	Oven	Instrument
1	440	HTO	PTOLEMY
2	470	HTO	COSAC
3	500	MTO	ÇIVA & PTOLEMY

Table 2 - Main features of the three samples.

As shown in the Table, the three samplings are performed at different depths; obviously each one is deeper than the previous one so that, they could be done at the same hole.

Another important feature is the oven that will be used: for the two first perforations a high temperature oven (HTO) is required and for the last one is required a medium temperature oven (MTO).

Finally, the instrument that analyzes the sample is also important. In the two first cases there is only one instrument involved but in the third case there are two instruments where the sample will be analyzed. Furthermore, as will be seen in this section, ÇIVA requires a previous calibration that will complicate a little the procedure to be done.

3.1 The Commands

For performing this three perforations it is required to know how to tell the software of SD2 the depth and the instrument that are involved, for this purpose there are a lot of commands that the software can send to SD2.

There are commands for downloading the telemetry, for starting and stopping the drill rotation and translation, for changing the acquisition frequency, for aborting action that is executing, etc.

Each commands required a specific number of words to be loaded on the telecommand buffer.

The list of all the available commands for control SD2 is shown in the Annex 1 and the Table 3 below shows the commands used in the FSS:

Name	Code	Words	Description
ONOF	00001	2	Switch On/Off electronic actuation sections.
CAPO	00011	4	Perform carousel rotation to a defined position.
DRTR	00101	3	Perform drill translation to a defined position.
DRGO	00110	3	Perform drill rotation for a defined time.
DRST	00111	2	Perform drill rotation stop.
SARE	01101	2	Release sampling tube.
MHIT	10001	2	Managing HITB acquisition.
STARTOP	10011	2	Notifies that the specified operation is starting. The SD2 current status of BCK_Data is accordingly updated.
STOPOP	10100	2	Notifies that the specified operation has been completed. The SD2 current status of BCK_Data is accordingly updated and the end-of operation is notified to CDMS via SR flag.
DELAY	10101	3	Performs the required delay before to executing next command.
LANDG	10110	10	Loads the data relevant to the landing gear.
DRTC	11000	4	Check drill translation, main or redundant.

Table 3 - List of all the commands and its definition.

3.1.1 Considerations

The sequence that will be analyzed corresponds to the sequence of commands that performs the First Science Sequence. The sequence of all commands to be executed has been modified based on the following considerations.

It is worth highlighting that these considerations will shorten the sequence to be considered but the commands that are removed will be added at the end of the analysis.

The commands that will not be included in the sequence are:

- STARTOP: this command is used to indicate that an operation is starting. Consequently it will be placed in each mission plan so, it is better not to include this command until the final analysis.

- STOPOP: This command has a role similar to the previous one but, in this case, the command is used to indicate the end of an operation.

- LANDG: This command refers to the landing gear. It is sent to the software in order to indicate that the landing gears are not under the drill. This command must be sent only if DRTR appears in the mission plan. So, it will be also considered at the end of the analysis.

3.2 The elementary mission plans

First to start thinking about how the sequence will be optimized it is important to know what will be analyzed and which are the expected results.

For this, the FSS is divided in three parts, one part for each hole. If the procedure to get a sample is observed, it can be seen that it follows a list of actions. This actions or procedures together enable the sampling of the cometary soil and they appear in each sample procedure.

These actions can be separated considering their function. These procedures are called "Elementary Mission Plans" and can be easily recognized observing the list of actions that should be performed for each sample [12]:

Step	Action	Elementary Mission Plan
1	Carousel rotation to put the rearming oven under the drill.	Drill bit rearming
2	Translation of the drill to rearm the drill bit.	
3	Translation to return to the initial position.	
4	Carousel rotation to get back to the initial position.	
5	Roto-translation of the drill to the specific depth and translation 1mm up.	
6	Release of the sampling tube.	Sample retrieval
7	Only rotation for a minute.	
8	Translation to get back to the initial position.	
9	Carousel rotation in order to put the desired oven under the drill.	
10	Translation to discharge in to the oven.	Oven feeding
11	Translation to get back to the 0 position.	
12	Carousel rotation to put the oven on the port of the required instrument.	

Table 4 - Parts of a sampling procedure and EMP identification.

As seen at the Table 4, there are actions that can be aggregated forming the elementary mission plans. Specially, they can be distinguished three that will be repeated at each sample procedure: the drill bit rearming, the sample retrieval and the oven feeding.

Furthermore, there are three actions that are not included in any EMP. The first one refers to the drill (step 5) and it depends on one input: the depth at which the sample should be taken. The other two actions (steps 9 and 12) are referred to the carousel movement and its inputs variables are the type of oven to be used and the scientific instrument that will analyze the sample.

3.2.1 Drill Bit Rearming

The first EMP that is identified is composed by the four first actions. The purpose of this bloc is rearming the drill bit. Along the travel to the comet the drill bit is disarmed, so, it is necessary to start the FSS rearming it. Furthermore, the drill bit is disarmed every time that a sampling is done so, it is essential to perform this process at the beginning of each sampling procedure. The actions that this EMP contains are [13]:

- Carousel rotation to put the rearming oven under the drill.
- Translation of the drill to rearm the drill bit.
- Translation to return to the initial position.
- Carousel rotation to get back to the initial position.

3.2.2 Sample Retrieval

The second EMP is performed after the perforation and it is performed in order to take the sample and put it up at the Drill's zero position. There are three actions inside this EMP [13]:

- Release of the sampling tube.
- Only rotation for a minute.
- Translation to get back to the initial position.

3.2.3 Oven Feeding

Once the sample is in the tube and the drill is above the oven it is necessary to put the sample in the oven. The third EMP has this objective and it is composed by two actions [13]:

- Translation to discharge in to the oven.
- Translation to get back to the 0 position.

3.2.4 Specific Mission Plan

It is important to keep in mind that the previous actions can be repeated exactly with the same commands for each sampling procedure. This fact changes if the steps 5, 9 or 12 are considered. These steps need additional input information, and the parameters used in the commands are different according to this additional information. For this reason they are called Specific Mission Plans.

- Step 5: this step includes the translation to the desired depth (needed input) where the sample will be taken. The DRTC command will include the information about the depth. For that, this step will change in every sampling procedure. After the required depth is reached, it is performed a translation 1 mm up in order to let the Sample Retrieval EMP take the sample.
- Step 9: this step is performed in order to put the desired oven under the drill to discharge the sample in the next EMP. The command involved in this action is the CAPO and it indicates the amplitude of the rotation, so, the oven that will be placed under the drill.

  In the third sampling procedure this step has some variations. The fact is that at this procedure it is required a calibration of the ÇIVA instrument. For doing that, the desired oven is positioned under the instrument and after its calibration the oven is putted under the drill.
- Step 12: this is the last step, and as the step 9, the input required affects the carousel rotation. In this case the objective is to put the oven with the sample under the instrument that will analyze it. As has been seen there are three different instruments and 26 ovens and depending on which is necessary the amplitude of the rotation will be different.

  In the third case it is found again a modification. For the analysis they are used three ports of two different instruments (ÇIVA-MV, ÇIVA-MI and COSAC), so, three rotations are performed and it appears the steps 12.1, 12.2 and 12.3 at the sequence.

It is important take into account that when PTOLEMY, ÇIVA and COSAC are working no SD2 commands can be sent and executed. It means that the same Mission Plan can not include the action before and after PTOLEMY, ÇIVA and COSAC analyses.

4. The Patterns

It has been introduced the fact that there are some groups of commands that are repeated along the first scientific sequence. The strategy that will be followed from this point is to study how these groups can be chosen in order to minimize the amount of data that will be charged on the SD2 Buffer.

First, it is made a study to know more about these groups of repeated commands that from now on will be called patterns. In order to do this a Matlab function is designed.

The main goals of this function are identify where the patterns, how long they are and to identify the strategy to follow. The function and its results are described in the next subsections.

For analyze the FSS it has been written the commands in the order of appearance and then it has been assigned a number to each command. In this way two commands that are equal will have the same number assigned.

4.1 The Pattern Found Function

The function works in the following way:

It takes an integer number that means the length of the patterns that will be searched. The number will start from 2 until a value high enough to ensure that there is not any pattern of that length.

Then the entire vector that contains the sequence of commands is analyzed in order to find the patterns of the selected length (at the first case length is 2). The objective is to have a variable that indicates all the patterns for each length and their positions. So, the algorithm for each length is the next:

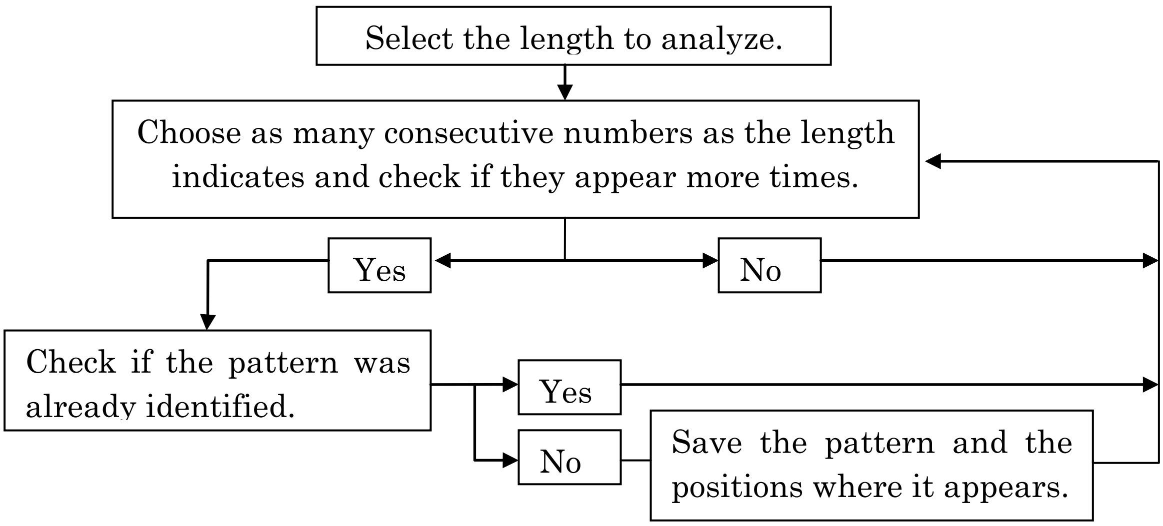


Figure 6 - Algorithm of the Pattern Function.

If the function is launched there will be a variable that contains all the patterns and their positions for all the lengths. The first interesting data to analyze is how many patterns are for each length. The next figure shows the relation within the number of patterns and the length:

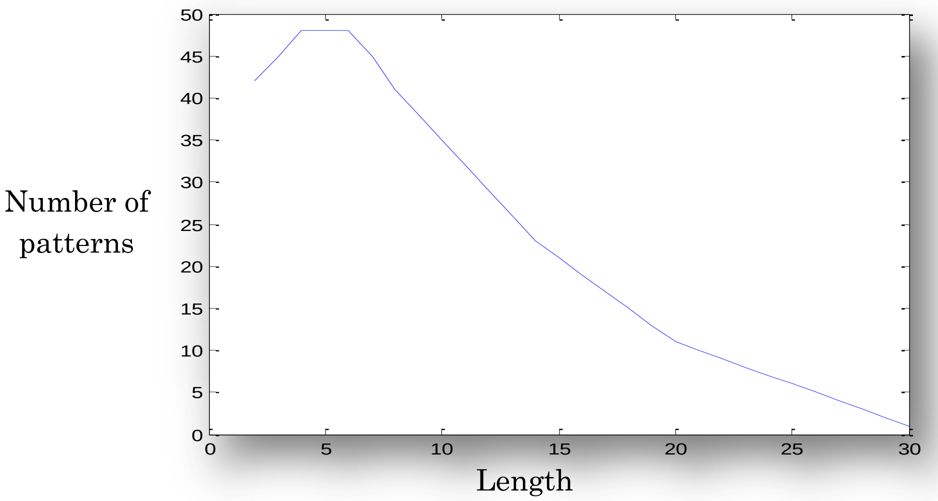


Figure 7 - Number of patterns in function of their lengths.

It is quite unexpected to find fewer patterns of length 2 than 3 but there is a reason. In fact, the small patterns have more probability to be repeated and the program detects these repeated patterns. However, it can be said that there are less patterns of length 2 than patrons of length 3, 4 or 5 but these patterns of length 2 are more repeated.

If it is wanted to know how many repetitions are for each length it is necessary to sum the number of repetitions of all patterns for every length:

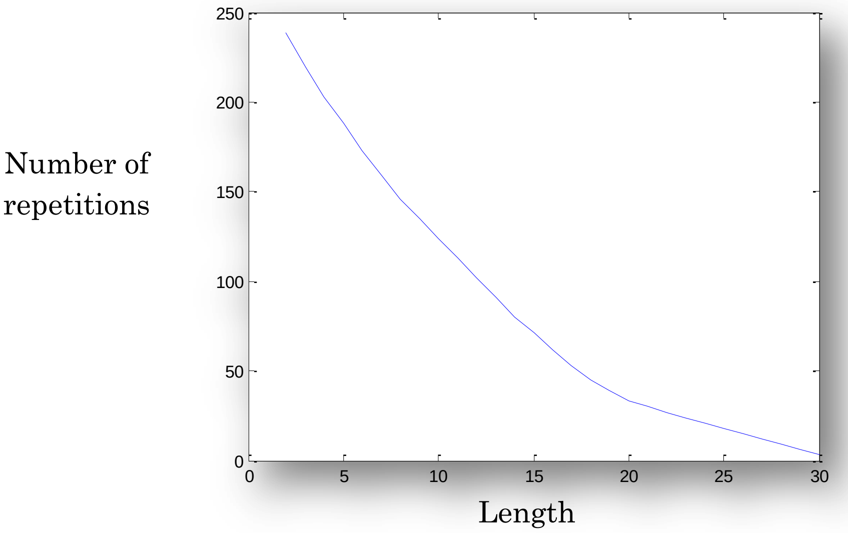


Figure 8 - Number of repetition in function of the length.

And finally, if it is wanted to know how many times a pattern is repeated for each length we have the next Figure:

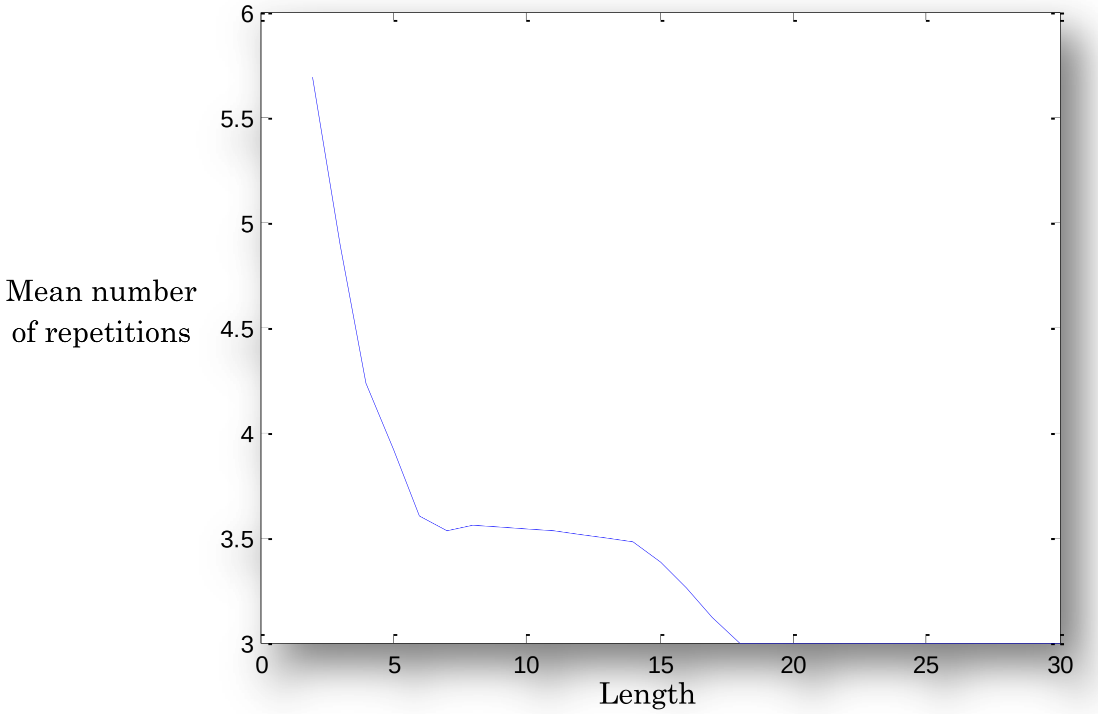


Figure 9 - Mean number of repetitions in function of the length

With this last figure it can be seen that at the first part there are some small patterns that appear a lot of times.

Then there is a phase between lengths 6 to 14 where the number of repetitions is more or less the same (we must keep in mind that the number of patterns will not be the same for these lengths, according to figure 1), and then from length 17 to the end it can be seen that the patterns are repeated three times, due to the three initial part of each hole that should be performed.

In this way, another important characteristic that can be analyzed is where the patterns are.

For this, it can be added the times that a position appears at all the patterns and the results is the Figure 10:

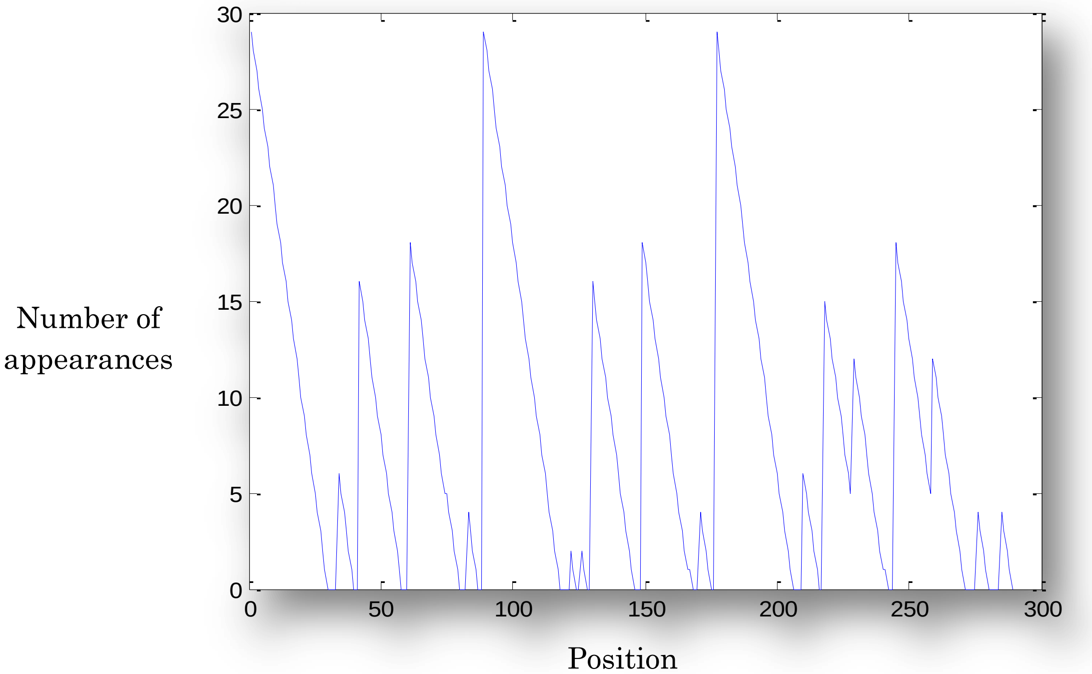


Figure 10 - Number of appearances of each command.

In this figure we can identify the parts of the First Scientific Sequence that more appear in patterns. In this way is possible to deduce that at the beginning of each sampling procedure there is a pattern that appears many times, in this case it corresponds to the first elementary mission plan. Also it can be seen the following two elementary mission plans of each sampling procedure.

It is not difficult to identify the three elementary mission plans that appear: the first form 1 to 30, the second from 45 to 55 and the last from 64 to 87.

It can be extracted more conclusions with this plot: The places where there is 0 repetitions means that this command is not repeated. For example it can be seen that at 57^{th} position there is a non repeated command due to the fact that this command is referred to the rotation of the carrousel and this value changes for all holes. Also there are two holes around the position 45 due to the two translations that should be performed that are also different for each hole.

4.2 The Words

Once it is got this knowledge about the patterns it is possible to continue thinking about how to optimize the memory of the SD2 system.

It is important to keep in mind that the goal is to reduce the data that will be uploaded on the Buffer. The parameter that will define the use of the memory is the number of words.

Not all the commands have the same weight. At Annex 1 it can be seen the number of words that each command has.

The first step is identifying the patterns. Once a pattern is found the number of words that it contains will be added only once to the total number of words, independently of the times that it appears along the sequence. Then the words of the commands that remain will be added. For counting the number of words the Words Function has been implemented. It is described in the next subsection.

4.2.1 The Words Function

The Words Function needs three inputs. First it has to take the sequence of number that identifies the commands. Now it is necessary to put also a sequence with the number of words of each commands, with this the function will be able to count the number of words of each pattern or lonely command. Last but not least the function needs a vector of lengths.

This vector contains the length of the patterns that will be look for. In the case of the Pattern Function the length was increased from 2 to a high number. Now, this vector contains specific lengths ≥ 2. For example, {5, 2, 10, 7, 3, 11}. If the numbers are too high it will be more unlikely to find a pattern with this length and the execution will be stopped leaving many commands lonely.

The next diagram shows the algorithm that the function executes:

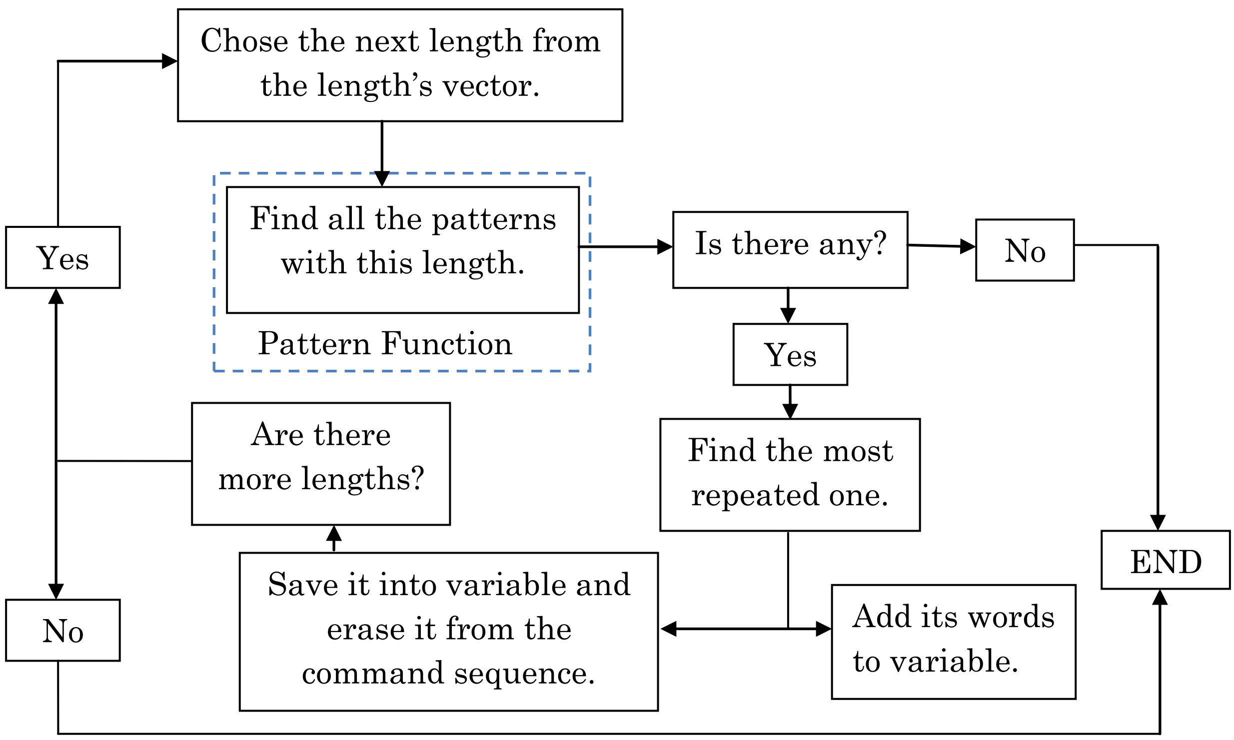


Figure 11 - Algorithm of the Words Function.

After the execution of this algorithm there will be a variable that contains the number of words belonging to the identified patterns. Then, it is necessary to add the words corresponding to the commands do not belong to any pattern.

Now the variable will contain the words of all the sequence. The number of words will vary depending on the combination of lengths that appear in the input vector to be evaluated. The objective of this study is to find the best combination of lengths. At this point the only constrain that will be taken into account is the number of words because that will give the amount of data that must be uploaded on the SD2 system.

In order to do this a PSO (Particle Swarm Optimization) will be performed. The process is based on evaluate the "objective function" (in this case the Words Function) and find a combination of lengths that gives the minimum number of words.

The next chapter describes the PSO algorithm.

5. Particle Swarm Optimization

Particle Swarm Optimization (PSO) is a population-based stochastic approach for solving continuous and discrete optimization problems ([14]).

The PSO method has been introduced for the first time by Kennedy and Eberhart ([15], [16]), and it belongs to the class of swarm intelligence techniques that are used to solve optimization problems. Particle Swarm Optimization was inspired by the movement of swarm of birds and fishes when searching for food and, as swarm of birds or fishes, each individual moves with a combination of his own criteria and the information that comes from the swarm [17]. In these two examples the objective is to find food, at PSO the objective function gives the objective to be minimized.

5.1. Algorithm

The Particle Swarm Optimization algorithm is composed by the following steps:

Initialize the position of each particle or individual inside the search space. This process is random and it is needed to define the search space (upper and lower bounds) and the number of particles that are going to be analyzed:

$$x_i = x_{min} + r_i(x_{max} - x_{min}) \quad for \; i = 1, \dots, N_{par}$$

in which x_i represents the position of the *i-th* individual within the design space, x_{max} and x_{min} are the upper and the lower bound of the search space, N_{par} is the number of particles that appear at the problem and r_i is a vector with random numbers belonging to the interval [0,1].

Evaluate the objective function for each individual:

$$y_i = f(x_i) \;\; i = 1, \dots, N_{par}$$

Update the values of p_{best} and g_{best}. p_{best} represents the best point that the particle has achieved and it will not vary until the new found point is better than it.

$$p_{best_i} = \begin{cases} x_i & if \;\; y_i < y_{p_i} \\ p_{best_i} & in \; the \; other \; cases \end{cases} \qquad for \; i = 1, \dots, N_{par}$$

where y_{p_i} is the value of the objective function at p_{best_i}.

g_{best} represents the best point achieved at all the swarm and it will not vary until the new found point is the best (lowest value) achieved at all the process for all the individuals of the swarm:

$$g_{best} = \begin{cases} x_i & if \quad y_i = \min\{y_g, y_1, y_2, \dots, y_{N_{par}}\} \\ & g_{best} \quad in\ the\ other\ cases \end{cases}$$

where y_g is the value of the objective function at g_{best}.

Compute the new particle position:

$$x_i^{k+1} = x_i^k + v_i^{k+1}\Delta t$$

in which v_i^{k+1} is the velocity of the *i-th* particle at the iteration k+1, and it is obtained by

$$v_i^{k+1} = wv_i^k + c_1 r_1 \frac{x_i^k - p_{best_i}}{\Delta t} + c_2 r_2 \frac{x_i^k - g_{best}}{\Delta t}$$

where w, c_1, r_1, r_2 and c_2 are parameters that will be described in the next section and Δt is the time step that is commonly set to 1.

Repeat the steps 2, 3 and 4 until the best solution is found or the result is not improved in the last N iterations, where N is an optimization parameter.

Due to the large number of iterations, the objective function must be optimized in order to reduce the time needed for the computation (erase unnecessary variables, optimize the code, etc).

In terms of concept, the PSO has common characteristics and features with other population-based models such as Genetic Algorithm (GA) and Evolutionary Algorithms (EA). These models are based on mimic natural processes and have a stochastic component. Furthermore, as the gradient of the objective function is not needed, discontinuous objective function can be dealt with [18].

5.2. Parameters

A Particle Swarm Optimization is characterized by certain parameters and all of these should be adapted according to the problem that will be analyzed.

There are parameters to define the model. N_{par} defines the number of particles involved, usually this parameter is called *population size*, and the bigger this number is the higher are the possibilities to find bests points. The *number of iterations* as the name indicates, is referred to the number of iterations that will be performed. Its value should be high enough to ensure that a better point will not be found.

There are also parameters that define the nature of the optimization. As seen in the algorithm section the PSO is characterized by the only three parameters w, c_1 and c_2 . The choose of their values is strongly problem dependent, therefore it is impossible to find the best values once for all and each one of these three is referred to one feature of the velocity characterization as seen below [17]:

$$v_i^{k+1} = \underbrace{wv_i^k}_{\text{inertia}} + \underbrace{c_1 r_1 \frac{x_i^k - p_{best_i}}{\Delta t}}_{\text{memory}} + \underbrace{c_2 r_2 \frac{x_i^k - g_{best}}{\Delta t}}_{\text{swarm}} \tag{5.2}$$

In equation 5.2 the inertial term depends on the velocity that the particle has at the previous iteration and also on the *inertial parameter w*. A high value of w means that the inertia term is very important and consequently corrections on the trajectory are too difficult. At the early versions of PSO this parameter did not exist or was considered 1 but further studies have proven an increase in the performances when a variable value of *w* in the interval [0.4, 1.4] is considered. The best way to vary the parameter w is to decrease its value along the iterations [18]. In this way we will have a greater value at the first part of the process and this will enable to better explore the search space. Then, as the number of iterations increases the value of *w* decrease and this allows a better analysis of the promising areas, giving more importance to the memory and swarm terms.

Other way is to reduce the value of *w* when a best point is not achieved in the last iterations [19].

The second term of that appears at the Equation 5.2 is related to the individual performance of each particle and it depends on the distance between the actual position and the best position achieved by the particle, for this reason the parameter c_1 is called *self confidence* and its value is from 1.5 to 2.

The last term of the Equation 4.2 is referred to the swarm and it depends on the distance between the actual position of the individual and the best position achieved by the swarm. The parameter c_2 is called *swarm confidence* and its value usually belongs to the interval [2, 2.5]. Obviously the importance of the swarm is bigger than the individual one.

The r_1 *and* r_2 parameters are random numbers between 0 and 1 and they are a good example of the stochastic nature of the method.

At figure 12 is shown graphically how the new velocity is computed taking into account this three terms:

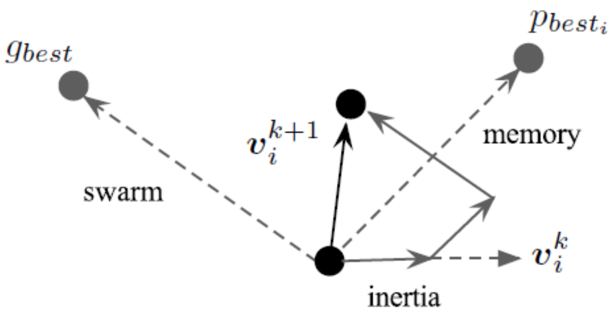


Figure 12 - New velocity obtained with the three contributions (swarm, memory and inertia)

5.3. Results

This section contains a few characteristic of the PSO obtained from the results of its application.

As seen in the last section it is important to choose right the parameters that will be used at the PSO. The two parameters that the user is able to modify in order to improve the results are the population size and the number of generations.

If a high number of generations is used, it can be assumed that a better solution will not be found (i.e. we likely reaches the global optimum). Usually before reaching the number of iterations the PSO's execution is stopped due to the fact that the best solution is not improved in the last 50 generations. With a high number of population size it can be assumed that a better exploration of the search space will be performed but it will be more difficult to influence the swarm due to the high number of individuals (so the mean value is improved more slowly).

In Figure 13 are shown the results for the combination of population size and generations with values from 50 to 300.

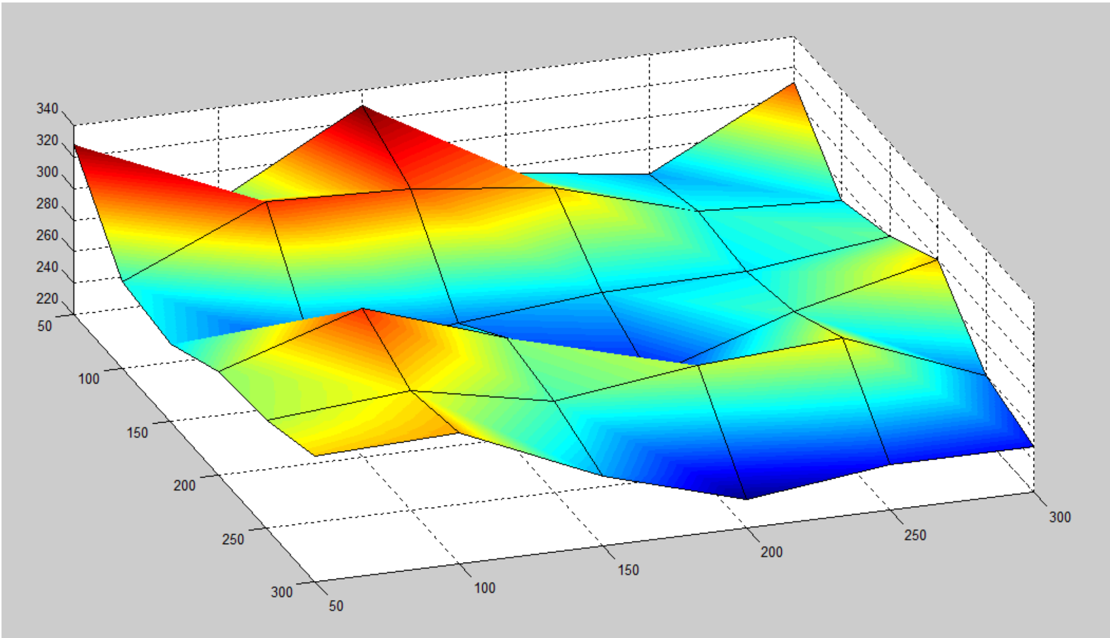


Figure 13 - Results of the PSO algorithm execution.

In this Figure, it can be seen that the better results for the PSO are usually obtained with a combination of a high number of generation and a high number of population size. It is also possible to conclude some rules to use the PSO:

For a small number of generations the best solution is practically random and it depends on the population size. For a high value, the constrain of improving the best value in the last 50 iterations is usually reached. So, it is advisable to use a value high enough for ensure that a better solution won't be found.

Other feature that can be seen while the PSO is executing is the representation of the intermediate results that are obtained. The mean value of the swarm and the value of the best individual are automatically displayed in function of the number of generations. As was expected, with the execution of the PSO, the values of both the mean and the best individual are reduced.

Watching the complete process in Figure 14 is possible to identify three phases. At the beginning of the execution both values vary a lot, like if the swarm was randomly moving looking for the best way to follow.

There is an intermediate phase where it seems that the strategy of the optimization is defined and the results are improved.

The behaviors of this phase look linear and they are followed until a point where the "best score" is not improved.

Finally, as said, it is seen a phase where the value of the best individual is not improved for many generations or it is improved to slowly. In this phase the mean value continues improving, approaching, sometimes, the value of the best individual.

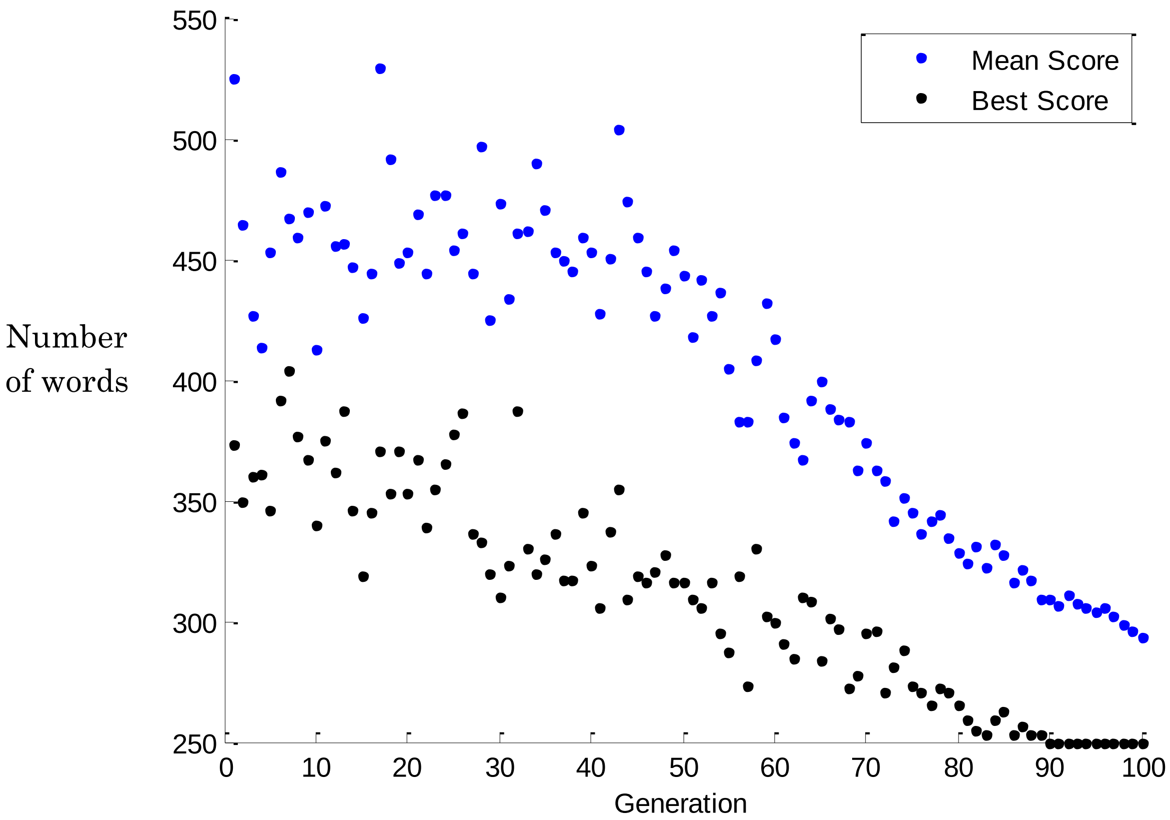


Figure 14 - PSO's evolution with generations.

The last feature of the PSO algorithm that is interesting for the user is the time that the process takes. The Figure 15 shows the time in minutes in function of the number of generations.

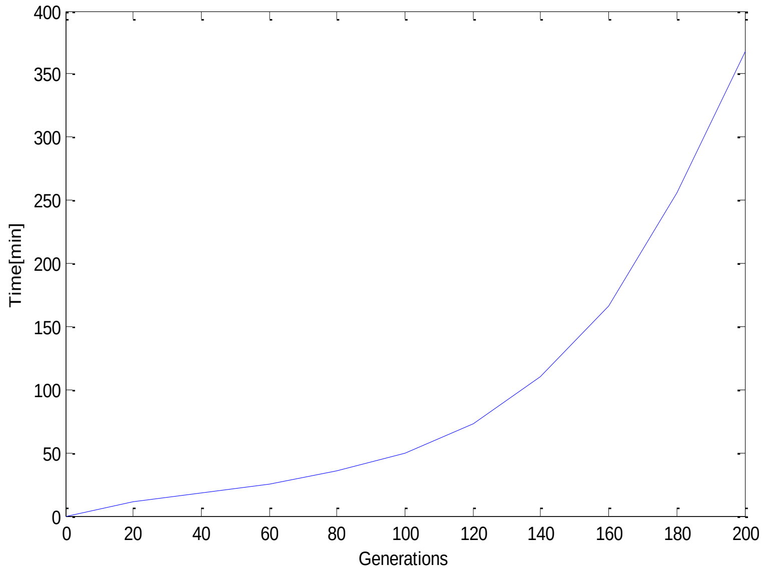


Figure 15 - Time versus generations.

This Figure was obtained with a PSO with 200 generations and a population size of 300. It is obvious that for a lower number of population size the values of time will be also lower but the distribution of the time will not change. It means that at the first phase the process is executed relatively fast and more or less each generations lasts the same time. It is seen at the first 80 generations where the behavior is more or less linear. Nevertheless, as the PSO advances, improving the result is more and more difficult and so the calculations last longer. This can be observed at the Figure 15 where from generation 100 the behavior becomes exponential.

6. Final Results

In this section all the results obtained will be analyzed in order to understand what the best solution is and how it can be reached. It is described the steps that have been made and the modifications that have been applied.

6.1 PSO Solutions

The PSO method finds the combination of lengths that gives the lower number of words. Depending on the evaluated sequence the combinations and the final result change.

At the beginning of the study it was wanted to evaluate a sequence with the considerations of the section 3.1.1 (without the LANDG, STARTOP and STOPOP commands). This sequence contains about 285 commands and a total of 717 words. Changing the parameters of the PSO (population size and number of generations) different solutions were obtained. The most repeated and better ones are shown at Table 5:

Case	Words	Interactions	Percentage
1	250	57	34.9
2	238	69	33.2
3	272	57	38

Table 5 - Results with main considerations

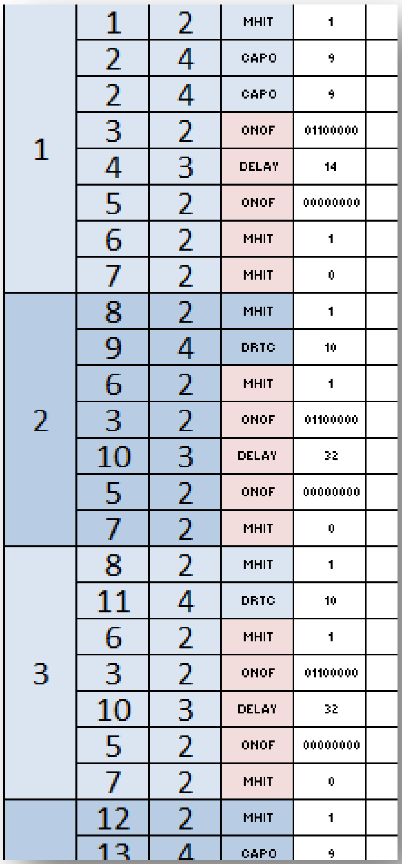
The number of interactions is indicated in order to see the fact that decreasing the number of words means have more patterns and so, more interactions would be performed between the CDMS and the buffer.

It is also important to consider the percentage that this number of words represents to the total one. This will be a helpful variable when a comparison between different sequences will be done.

To see how the PSO method has tried to aggregate into patterns the maximum number of commands it is computed the efficiency. The value obtained in the second case is 85% and it means that the 85% of the commands belong to patterns.

The different results for different values of population size and generations were shown in Figure 13.

Then, it was observed that a little structure was repeated after each operation. This structure contains the commands related to download the telemetry and change the acquisition frequency:



1	1	2	MHIT	1	
	2	4	CAPO	9	
	2	4	CAPO	9	
	3	2	ONOF	01100000	
	4	3	DELAY	14	
	5	2	ONOF	00000000	
	6	2	MHIT	1	
	7	2	MHIT	0	
2	8	2	MHIT	1	
	9	4	DRTC	10	
	6	2	MHIT	1	
	3	2	ONOF	01100000	
	10	3	DELAY	32	
	5	2	ONOF	00000000	
	7	2	MHIT	0	
3	8	2	MHIT	1	
	11	4	DRTC	10	
	6	2	MHIT	1	
	3	2	ONOF	01100000	
	10	3	DELAY	32	
	5	2	ONOF	00000000	
	7	2	MHIT	0	
	12	2	MHIT	1	
	13	4	CAPO	9	

Figure 16 - Highlight of the downloading telemetry and changing frequency commands.

These commands are highlighted in a soft red. They perform the switch on of the resolvers (ONOF), a delay, the switch off of the resolvers (second ONOF), the damping of the information (MHIT 1 1) and the change of the acquisition frequency (MHIT 0 1).

It was decided to remove this structure from the sequence. This must be done because it is necessary to download the data right after the activity. In this way the number of total commands is reduced a lot and a further study can be performed. This is the first of a few modifications that will be performed along the study.

It is important to keep in mind that the structure to download and change the frequency will be considered at the end of the process and added to the result.

If the PSO is launched again for the new sequence, there are obtained two different results as shown in Table 6.

.

Case	Words	Interactions	Percentage
1	171	33	53.3
2	183	27	57

Table 6 - Results without downloading telemetry and changing frequency commands

The difference between the first and the second cases is: +12 words & -6 interactions and it is due to the fact of considering the first EMP together or split it in three parts.

If these results are compared with the results obtained with the first sequence it can be seen that the number of words has decreased. Nevertheless, if a real comparison is wanted, the important parameter is the percentage. In the first case this parameter was around 35% but now this value has increased to 55%. This means that the removed commands were into patterns and the commands that were alone are still alone. On the other hand, this configuration of the sequence allows to easily seeing what the most important patterns are and where they are, as seen in the first column of the Figure 17. In this figure the patterns are shown in different type of blue, from soft to hard in order of appearance.

In this step of the study, it is easy to recognize the EMP of the section *3.2 The elementary mission plans*. It could be seen that, in this case, all the elementary mission plans belong to patterns. Furthermore, there also two blocks included into patterns that do not belong to any EMP: the first is at the fifth action and it includes two commands, the DRST and the MHIT. The second is located at the third sampling procedure and it includes the 9.1 and the 12.1 actions.

These two actions are the same: rotation of the carousel to put the desired oven under the ÇIVA instrument, the first time to calibrate it (9.1) and the second time to analyze the sample (12.1). So, it is expected that this pattern that only belong to the third hole will always appear.

Analyzing the pattern of the fifth action, it has to be noticed that the DRST command affects to the DRGO command, so these two commands have to be put together. In order to do that, the number assigned to the DRST command is changed. In this way it will not be recognized as pattern.

This is the third consideration included whose objective is to put together these two commands. Changing the sequence, the expected result is the same as before but without including this pattern on the fifth action. So, the number of words will increase by 8 ((2words of DRST+2 words of MHIT)*2 sampling procedures) and the number of interactions will decrease by 6.

Case	Words	Interactions	Percentage
1	179	27	55.6
2*	191	21	59.5

Table 7- Results of the DRST modification

The second case was introduced by hand because the PSO did not found it. It is completely logical due to the fact that the PSO method look for the solution with less number of words and the second one has more words. The difference between them is (as at the previous step) grouping the first EMP together or splitting it in three patterns.

At the second column of the Figure 17 this result is illustrated. As was expected, it is the same as that the previous one but without the pattern of the fifth action.

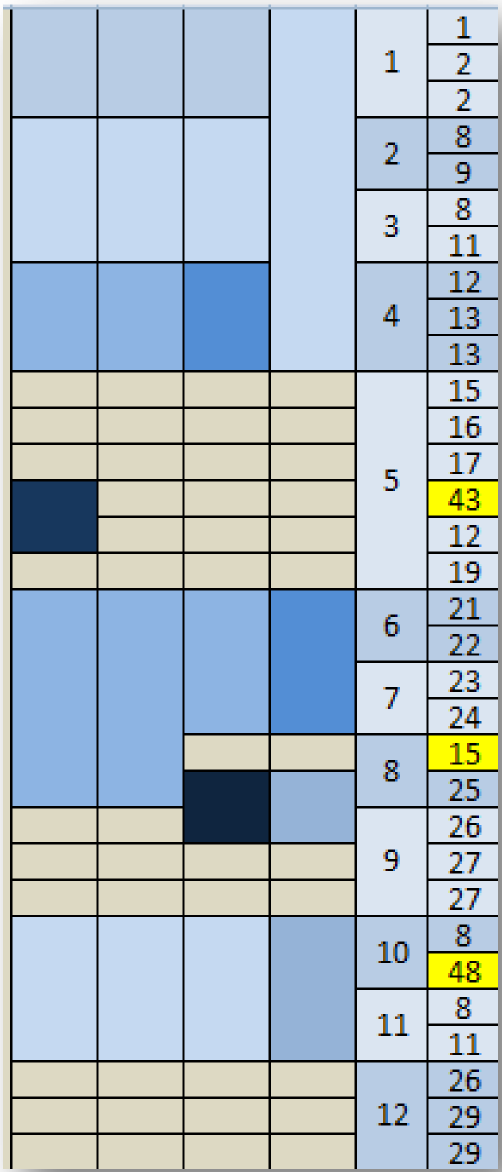


Figure 17 - Representation of patterns with different considerations.

The third constrain that will be introduced in the sequence is related with the MHIT command of the eighth action. This MHIT command has the mission of changing the acquisition frequency.

At the previous studies this frequency was fixed to a constant value but, on second thought, it has been decided to change it at every sampling procedure.

The fact is that the action performed after is to put the drill up to the initial position. It is known that the translation distance to be performed in this step depends on the depth where the drill is. So to have the right telemetry data it is better to change the acquisition frequency.

In order to do that the numbers corresponding to the three MHIT of this position have been changed and in this way the command will not included at any pattern.

The result is shown at third column of the figure. Here it can be seen that the command do not belong to any pattern. Nevertheless, the next two commands have been grouped in a pattern together.

The results of this case are shown in the Table 8. As the previous study, the second case where the first EMP only belongs to a pattern is obtained by hand.

Respect to the previous study there are some changes. The number of interactions is increased by 6. There are two more interactions at each sampling procedure. The number of words is also increased; in this case there are six more words:

Case	Words	Interactions	Percentage
1	185	32	57.6
2*	197	26	61.4

Table 8 - Results whit different acquisition frequencies when the drill moves up.

The last modification that will be done to the sequence is related to the oven feeding procedure. As can be seen all the commands that have been modified are highlighted in yellow. In this case the reason is to change the position of the translation. Now, instead of going to 1500, the drill will go to 1000. This is due to the fact that the rearming procedure needs to put all the drill inside the rearming oven. At the case of the oven feeding the drill will only reach the position 1000, otherwise the oven can be damaged.

The fourth column corresponds to this result. There is visible the difference with respect to the previous study. In this case there is only one solution. This solution is the same as the second one of the previous study:

Case	Words	Interactions	Percentage
1	197	26	61.4

Table 9 - Result of changing the DRTC of the oven feeding procedure.

As was commented previously, this solution corresponds to group the first EMP all together instead of splitting it in three subpatterns. The reason for why now the EMP is not split is the next: at previous studies the commands on the second and third action were the same as the commands on the tenth and eleventh actions. So, one pattern could include all these commands.

However, with the last modification of the sequence, these two actions are not the same so they cannot be grouped into one pattern. In this way, the better solution is to put together the first EMP as is shown in Figure 17.

To summarize all the results and to have a global vision of the changes that have been done it is included the Table 10.

	Case	Words	Interactions	Percentage
Normal (1): only removing STARTOP, STOPOP and LANDG commands.	1	250	57	34.9
	2	238	69	33.2
	3	272	57	38
Without change acquisition frequency and download the telemetry (2)	1	171	33	53.3
	2	183	27	57
(2) + Putting together the DRST and the DRGO commands (3)	1	179	27	55.6
	2*	191	21	59.5
(2) + (3) + Putting a different acquisition frequency when the drill moves up (4)	1	185	32	57.6
	2*	197	26	61.4
(2) + (3) + (4) + Differentiate between the movement of rearming and the oven feeding (5)	1	197	26	61.4

Table 10 - All the results for each consideration.

In conclusion, the final result that includes all the considerations and modifications of the command sequence is the last one. As is shown at Figure 17, the patterns of this solution includes all the elementary mission plans that were described at the third section with a little modification at the eighth action. There is also a pattern that was not expected, this is the pattern that appears at the 9.1 and at the 12.1 actions of the third sampling procedure described before.

Now that an optimal solution is found, it is necessary to predict how many times the Buffer of the SD2 system will be upload in order to carry on all these blocks of commands.

6.2 The uploads

Once the number of words is optimized, it is necessary to make a study in order to know how many uploads will be necessary. In fact, the final aim of the studies is to choose the better way to fill the buffer. The number of words is an intermediate step that is necessary to this purpose.

As was commented at the section

2.3 CDMS, it is know that the Buffer where the commands will be uploaded has space for 512 words. So, let's see how many buffers are necessary to carry on with all the words.

To perform this analysis, it is considered the last solution, namely, the one with all constrains applied. The procedure to fill the telecommand buffer is the following:

- The block with all the patterns is put at the beginning. Once it has been placed, the specific blocks are put in order of appearance.
- If a block is too big and it is not possible to put it into the buffer, a new buffer will be required. This new buffer will be filled with the pattern at the beginning and then the other blocks, starting for the one that was not included in the last buffer.
- The group of commands that was not included in the sequence to analyze must be considered now. The criteria to put this commands is shown below:

➢ LANDG: this command has 10 words. It will be included at every block that has a translation of the drill, namely, at every block that contains the DRTC command.

➢ STARTOP: this command has 2 words. As its name indicates it is used to start an operation. Consequently this command has to be placed at every block in order to start the operation.

➢ STOPOP: As the STARTOP command, this one has 2 words and it should be placed at all the blocks in order to indicate that the operation is finished.

➢ Downloading Data: this group of commands was removed in the first step of the last section due to the fact that it was repeated a lot and could be considered apart. Its aim is to download the telemetry data and change the acquisition frequency to the default one.

The sequence of commands of this procedure is listed below:

Command	Paremeters	Function	Words
ONOF	01100000	Switch on resolvers	2
DELAY	16	Put a delay	3
ONOF	00000000	Switch off resolvers	2
MHIT	0 0	Damp the data	2
MHIT	1 16	Set the default acquisition frequency	2

Table 11- Commands of downloading procedure

As is shown at this table, the total number of words of this group of commands is 11. This command will be added every time that a movement is performed (i. e. carousel rotation, drill translation, etc). Every action of the sampling procedure (Table 4) has a movement except the 6th that has not any and the 5th that has two. So, the total number of actions of every sampling procedure is 12 (expect for the third).

At this point it is possible to compute the total number of words that the FSS contains. At solution that was found in the last section, the one that had 197 words, these commands are added to the solution that was found in the last section (the one with 197 words) and its number of words increases to 584.. This result is closer to 512 than was expected but, it means that will be necessary more than one buffer.

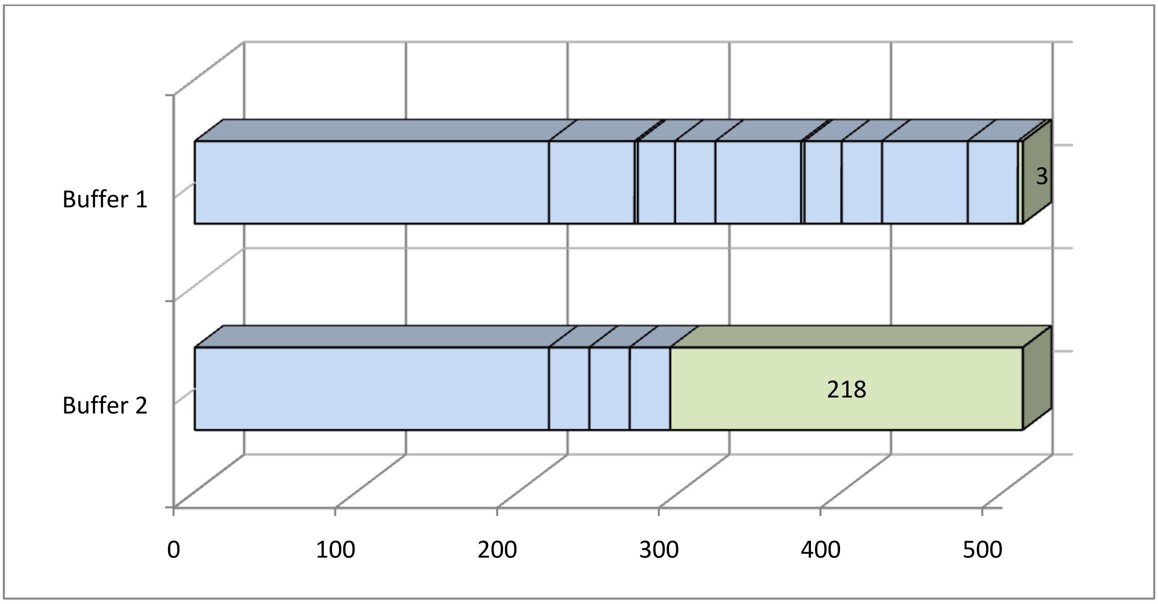


Figure 18 - Buffers for the solution with all considerations (5).

At Figure 18 is shown the distribution of these words into two buffers.

It can be seen that at the beginning of both buffers there is a big block. It contains the words pertaining to the patterns and it is placed on both buffers. After that it is easily observable the buffer is filled with smaller blocks corresponding to the actions that are not included into any pattern.

If the blocks are analyzed it is possible to conclude that the first buffer contains the data for the first and second sapling procedures and part of the third. Specifically, it includes until the 8 action. Consequently the data that the second buffer contains is the patterns words, and the words of the 9.2, 12.2 and 12.3 actions. So, in this case, it is not necessary to load all the words of the patterns, it is only needed to load smaller blocks, so few data.

It is also possible to see that the space remaining at the first buffer is only 3 words and 218 at the second one.

This solution could be compared with the solution that will be obtained if the acquisition frequency is constant for each drill translation to the initial position (case (4) of the previous section). The Figure 19 shows this result:

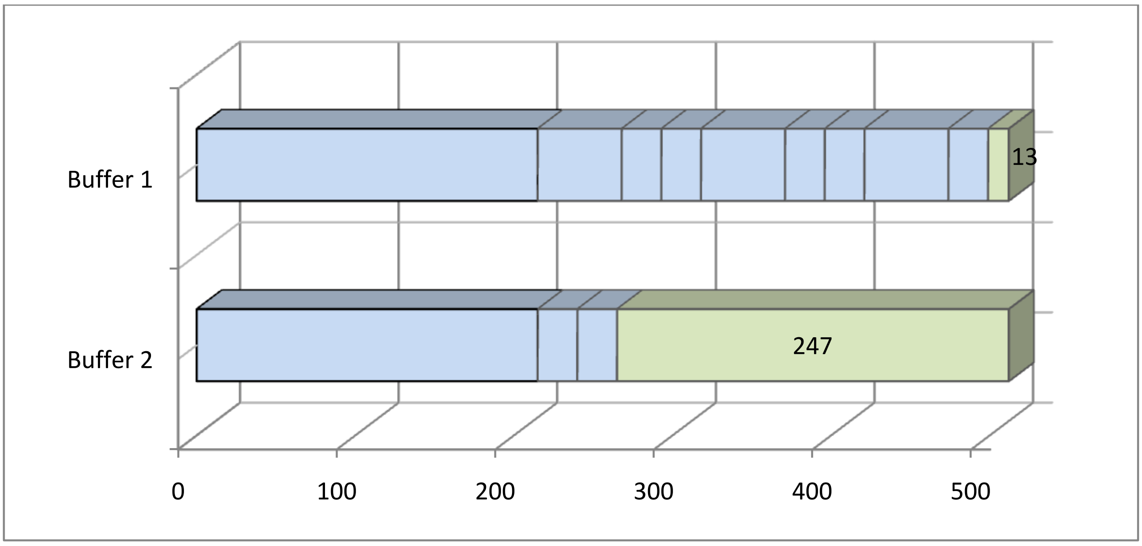


Figure 19 - Buffers for the result where the change of frequency when the drill moves up is not considered (4).

Here it is possible to observe that the amount of words pertaining to the patterns is quite bigger and the number of blocks added is more or less the same in both cases.

In this case, the number of words increases from 197 to 549, only 39 words more than 512. This small difference allows putting the 9.2 action in the first buffer and, then, the second buffer only contains the pattern's words, the 12.2 and the 12.3 actions.

It is important to see that also the free space at the end of the buffers has changed. This seems to have not any importance but the free space at the end of the buffer could let putting useful commands like ABORT, ENEM and this type of commands needed for managing recovery actions for non-nominal SD2 behaviors.

6.3 Number of interactions

Until now, the only objective was to reduce the number of words to put on the Buffer. Having a more general thought it appears the idea of also reduce the number of interactions.

As has been seen, in order to decrease the number of words it is necessary to have more patterns but, having more patterns also implicates that the global sequence is split in more parts and so, there are more interactions.

As was seen in the CDMS subsection, an amount of time is required every time that the CDMS contacts the Buffer to charge a new lock of commands, so the number of interactions also represents a constrain in order to optimize the procedure.

To have a view of what will happen if this parameter is considered it has been defined a new variable called "Interactions Weight" (IW).

6.3.1 The Interactions Weight

The IW factor represents the importance of one interaction respect the importance of one word. For example IW=2 means that reduce one interaction has the same importance than reduce 2 words. Until now, reducing the number of interactions was not an objective, so the IW was 0, it means that the interactions had not importance.

The previous objective function ends in this way: $\boldsymbol{Result} = \boldsymbol{Words}$. Now, the objective function is modified to: $\boldsymbol{Result} = \boldsymbol{Words} + \boldsymbol{IW} * \boldsymbol{Interactions}$.

Launching again the PSO with this modification new results are obtained. It is important to keep in mind that the last sequence of commands was used, so it can be seen that results for IW from 0 to 1.5 is the same as the one found previously.

At Table 12 the results for different IW values are shown. In order to also see how the patterns are grouped it has been made a figure that shows the patterns of the first sampling procedure.

The expected result is to find less patterns if the importance of interactions is increased.

RESULT	INTERACTIONS WEIGHT (IW)	WORDS	INTERACTIONS
1	From 0 to 1.5	197	26
2	From 1.5 to 3.85	203	22
4 (only one pattern)	From 3.85 to 21.3	257	8
5 (without patterns)	From 21.3	321	5

Table 12- Results for different values of Interaction Weight.

Here it can be seen that there are four possible solutions. The first one was the one found previously, as shown at the third column of the Figure 20.

The second is the same as the first but in this case one pattern that group the last command of the eighth action and the first one of the ninth.

The reason of this new sequence is that the pattern is so little. Increasing the interactions weight, the PSO found a result where the smallest pattern is removed.

In this way the number is only increased by 6 and the number of interactions has decreased by 4.

If the IW is increased it is found another solution. This corresponds to the first column and it can be seen that the only pattern that remains is the one that corresponds to the first elementary mission plan.

This result has 60 more words than the first one but the number of interactions has decreased from 26 to only 8.

If the IW is increased more than 21.3 the solution found is the worst case in term of words. It is not taking any pattern and leaving all the commands alone. So, 5 groups are formed, this implies 5 interactions, and the number of words increases to 321. As this solution does not take any pattern it has not been shown at the Figure 18.

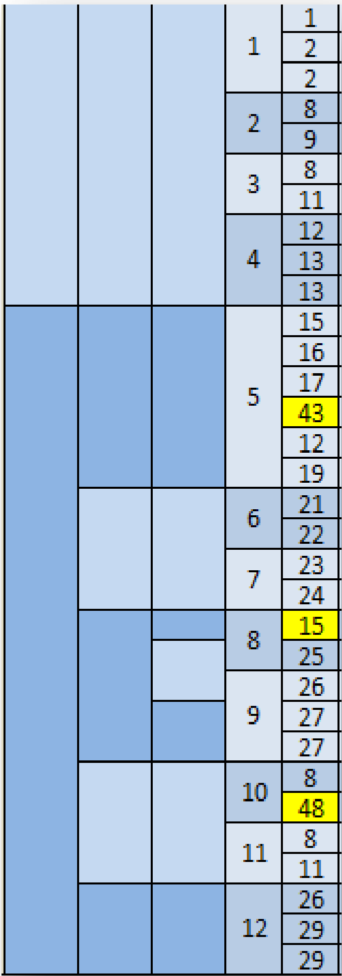


Figure 20 - Results for different values of Interaction Weight

Figure 19 shows the number of words against the number of interactions in order to see the dependence.

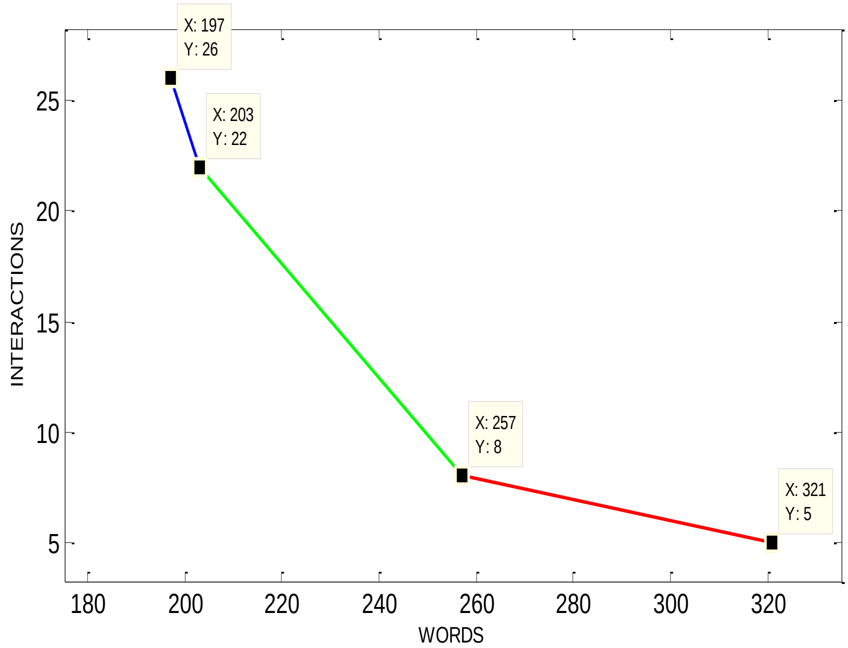


Figure 21 - Number of interactions against number of words.

It is important to keep in mind that it is not possible to choose intermediate solutions: there are only four possibilities and the choice depends on the importance that the interactions have compared to the number of words.

6.3.2 Uploads depending on interactions

Once these solutions are found, it will be interesting to analyze the number of uploads of each one. At this point it is important to consider that the number of words will change when the LANDG, STARTOP, STOPOP and Downloading data commands will be added.

RESULT	INTERACTIONS WEIGHT (IW)	WORDS	**WITH WORDS ADDED**
1	From 0 to 1.5	197	**584**
2	From 1.5 to 3.85	203	**607**
4 (only one pattern)	From 3.85 to 21.3	257	**662**
5 (without patterns)	From 21.3	321	**800**

Table 13 - Comparison with all the words added

As can be seen at Table 13 the differences between the solutions have increased. This difference appears due to the considerations of the new added words, the patterns are designed in order to earn words and, if the total amount of words increases, the earned words will also increase.

As is supposed, the number of interactions is proportional to the number of blocks that are uploaded at the buffer. This means that, for the result 5 of the Table 12, the buffer will have the two first sampling procedures at the first buffer and the third at the second buffer. So, as can be seen at Figure 22 there are big and few blocks.

This solution can be compared with the next one (Result 4) that only considered one pattern. In this case, there is an initial block containing the pattern's words that is placed in both buffers and, then, there are the specific blocks that in this case are smaller. Figure 23 shows this result and allows comparing graphically the two solutions.

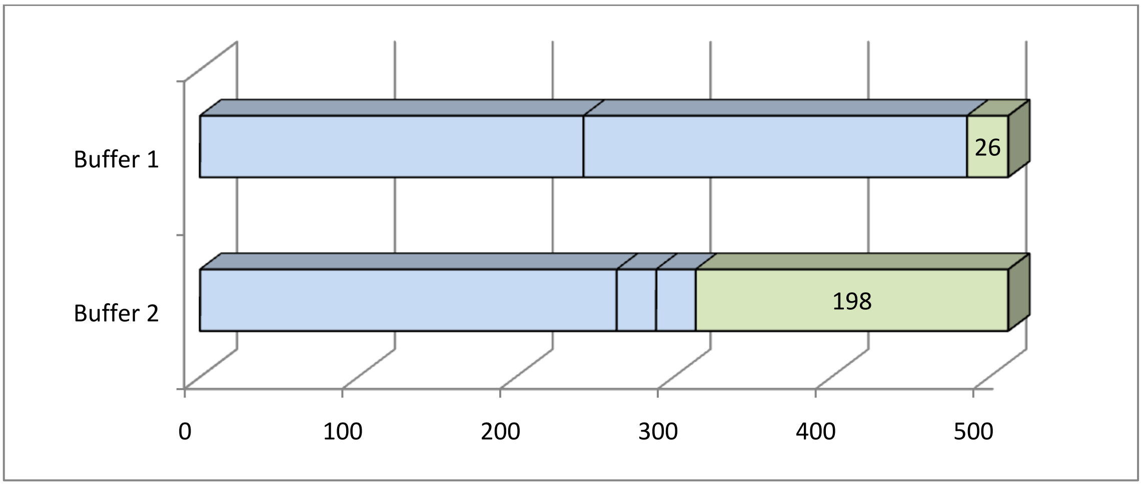


Figure 22 - Buffers if any pattern is considered.

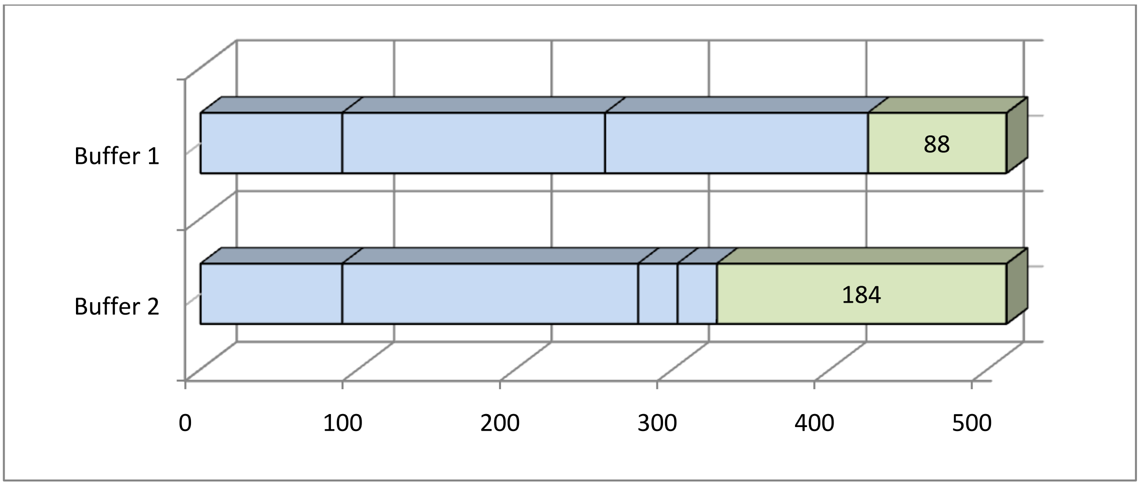


Figure 23 - Buffers if one pattern is considered

The two solutions that were found with a higher number of patterns are also analyzed. In this case is possible to see that the first block of each buffer is bigger than at the previous cases. It is due to the amount of patterns considered. In this case there are more patterns and so, more words related to them. Consequently, the specific blocks are smaller and the buffer can be filled with less words remaining.

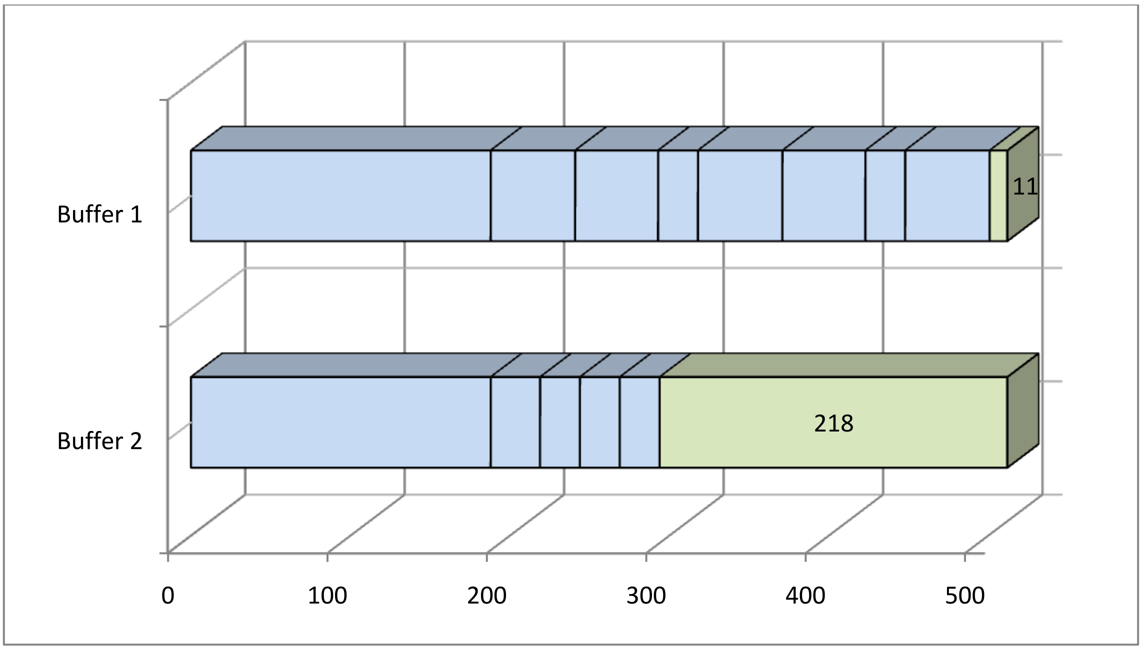


Figure 24 - Buffers for the Result 2 (consider all patterns except one)

At the Figure 24 it can be observed the distribution of the words. Respect the other cases the number of blocks has increased a lot and the remaining space at the end of the buffer has decreased. In this case the first buffer includes until the 8th action of the third sampling procedure. So, the smaller blocks of the second buffer correspond to 8, 9.2, 12.2 and 12.3 actions.

In order to compare it, the Figure 25 is included (it was also included in the previous section: Figure 18) to show the buffers for the Result 1 where all the patterns are considered.

In this case the biggest first block is obtained due to the fact that is the one that includes more patterns. It can be observed that, now, the first buffer includes the commands until the 9.2th procedure.

Finally, it is important to see that there is space only for 3 words at the end of the first buffer.

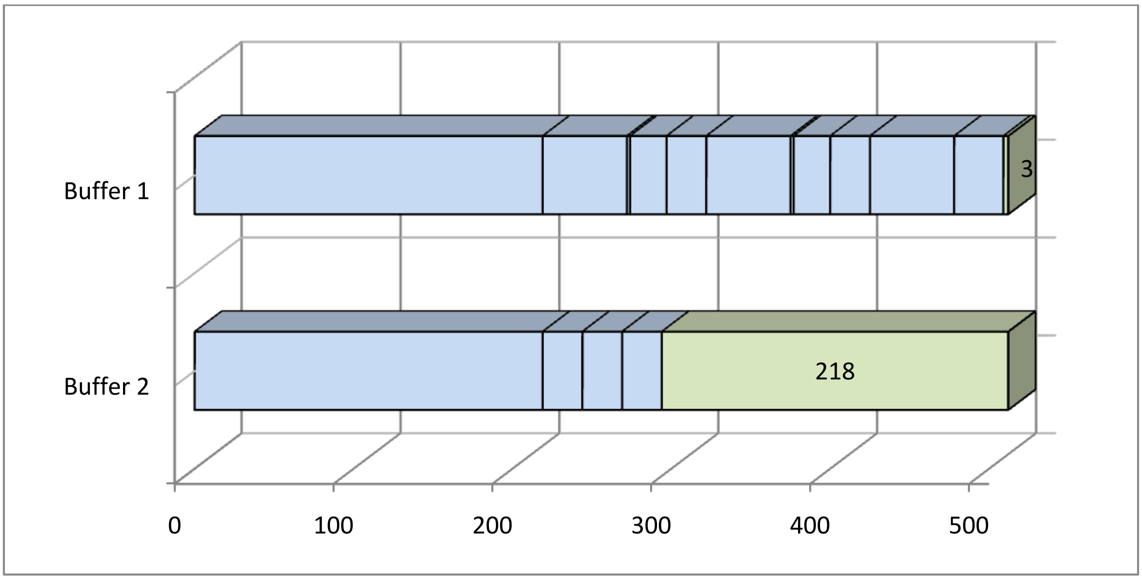


Figure 25 - Buffers for Result 1: considering all the patterns.

Once all these figures and these buffers have been studied, it can be said that, maybe, the best strategy to choose the way of filling the buffers is by observing the buffer's configuration because it gives a global view of the result.

Although it has seen that all solutions (even without considering any pattern) require two buffers, it is important to keep in mind that patterns can be useful in two ways. First, it will be possible use the free space that has been earned for load commands about recovery actions. And secondly, the use of patterns reduces the number of commands to be uploaded for the second buffer. For example, if there are only added the necessary commands in the second buffer for result 1, there will be only 100 words to upload.

7. Conclusions and further developments

In conclusion, it can be said that this study has clarified the way to upload the commands at the Stored Telecommand Buffer. It have been evaluated the results obtained at previous studies and it have been included more strategies. The idea of consider the number of times that the CDMS will contact the STCB has been also included and evaluated and it is shown a graphical model of the STCB in order to illustrate the number of uploads and the memory consumption of the system.

About the number of words, it can be said that it has been found the optimal solution considering all the constrains, this solution contains 584 words. It has been seen that without considering the change of frequency when the drill moves up the total number of words decreases to 549. Consequently, this suggests that, for future developments, it could be interesting to think about if the sequence can be reduced or modified in order to have less than 512. With this, one buffer will be enough to carry all the commands.

In general, the aim of this project is to show all the possibilities that can be considered in order of optimize the memory of the SD2 system. With all this information it may be selected the best configuration.

Below they have been listed some further developments that will be interesting to perform at future studies in order to define the way to upload the commands on the STCB:

- To study the possibility of reduce the commands of the sequence that is analyzed. Try for different acquisition frequencies.

- Modify the objective function. It would be interesting to have an objective function that not only optimize the number of words or interactions but also optimize the number of uploads. This objective function will include the commands that were considered at the end.

- Make a study about what is the best way to fill the STCB. It may be better to have some free space at the end in order to put commands to manage the SD2 performance like ABORT, ENEM, ENHEN, etc.

- Keep the parameter Interactions Weight and investigate what will be the best value in order to have the best solution.

- Test the results in order to prove that they are feasible.

Bibliography

[1] J. M. Sanchez and J. Rodriguez Canabal, "Rosetta: Consolidated report on mission analisys. Churyumov-Gerasimenko 2004", Tech. Rep. RO-ESC-RP-5500, ESOC, 2003.

[2] http://sci.esa.int/science-e/www/object/index.cfm?fobjectid=14615

[3] M. C. De Sanctis, M. T. Capria, and A. Coradini, "67P/Churyumov-Gerasimenko nucleus model: Portrayal of the Rosetta target", Advances in Space Research, vol. 38, pp. 1906-1910, 2006.

[4] Ercoli-Finzi, A., Magnani, P.G.,Re, E.,Espinasse, S., and Olivieri, A., “SD2-How to Sample a Comet”, Space Science Reviews, Vol. 128, 2007, pp. 281-299.

[5] F. Goesmann, H. Rosenbauer, R. Roll, C. Szopa, F. Raulin, R. Sternberg, G. Israel, U. Meierhenrich, W. Thiemann and G. Munoz-Caro, "Cosac, The Cometary Sampling and Composition Experiment on Philae", Space Science Reviews, (2007) 128: 257-280.

[6] I. P. Wright, S. J. Barber, G. H. Morgan, A. D. Morse, S. Sheridan, D. J. Andrews, J. Maynard, D. Yau, S. T. Evans, M. R. Leese, J. C. Zarnecki, B. J. Kent, N. R. Waltham, M. S. Whalley, S Heys, D. L. Drummond, R. L. Edeson, E. C. Sawyer, R. F. Turner and C. T. Pillinger, "Ptolemy - an Instrument to Measure Stable Isotopic Ratios of Key Volatiles on a Cometary Nucleus", Space Science Reviews (2007) 128: 363-381.

[7] J.-P. Bibring, P. Lamy, Y. Langevin, A. Soufflot, M. Berthé, J. Borg, F. Poulet and S. Mottola, "CIVA", Space Science Reviews (2007) 128: 397-412.

[8] J.-P. Bibring, P. Lamy, Y. Langevin, A. Soufflot, M. Berthé, J. Borg, F. Poulet and S. Mottola, "CIVA", Space Science Reviews (2007) 128: 397-412.

[9] P.Francesconi, Pierluigi DiLizia, Fabio Malnati, Franco Bernelli-Zazzer and Amalia Ercoli-Finzi, “Sampler, Drill and Distribution system for in-situ analysis of a cometary nucleus”, in 3rd CEAS Air&Space Conference.

[10] P.G. Magnani, E. Re, A. Ercoli Finzi, R. Mugnuolo and A. Olivieri, "ROSETTA lander - SD2 subsystem"

[11] Crudo, E., and Bologna, P., “CDMS-SD2: Data Interface Control Document”, TECNOSPAZIO S.p.A., 2002.

[12] Pierluigi DiLizia, Fabio Malnati and Amalia Ercoli-Finzi, "SD2 elementary misión plans", SD2 programm: PHILAE-SD-EMP-001, 10 March 2010.

[13] Davide Tonini, Caratterizzazione del sistema SD2 ai fini del suo utilizzo como strumento scientifico, Tesi de laurea, 2008, Politecnico di Milano.

[14] Marco Dorigo et al. (2008) Particle swarm optimization. Scholarpedia, 3(11):1486.

[15] Eberhart, R.C, and Kennedy, J., "A New Optimizer Using Particle Swarm Theory," Proceedings of the Sixth International Symposium on Micro Machine and Human Science, 1995.

[16] Kennedy, J., and Eberhart, R. C., "Particle Swarm Optimization," Proceedings of the IEEE International Conference on Neural Networks, 1995.

[17] Roberto Armellin, Multidisciplinary Optimization of Space Vehicles for Atmospheric Maneuvers, Tesi di Dottorato di Ricerca in Ingegneria Aerospaziale, XIX Ciclo - Marzo 2007, Politecnico di Milano.

[18] Shi, Y., and Eberhart, R., "Parameter Selection in Particle Swarm Optimization," Proceedings of the 7th International Conference on Evolutionary Programming VII, 1998.

[19] Fourie, P.C., and Groenwold, A.A., "The particle swarm optimization algorithm in size and shape optimization," Structural and Multidisciplinary Optimization, 23(4), 2002.

APENDIX A: TABLE OF COMMANDS

Name	Code	Words	Description
ZERO	0000	3	Perform zero condition and switch off all act. Sections.
ONOF	00001	2	Switch On/Off electronic actuation sections
ACRE	00010	2	Acquire values of resolver #1 and resolver #2
CAPO	00011	4	Perform carousel rotation to a defined position
CASI	00100	4	To move an oven to a scientific port
DRTR	00101	3	Perform drill translation to a defined position
DRGO	00110	3	Perform drill rotation for a defined time
DRST	00111	2	Perform drill rotation stop
MVCK	01000	3	Move volume checker
VCAC	01001	4	Activate volume checker
ABRT	01010	2	Abort Command
EMST	01011	2	Emergency Stop
EHEN	01100	2	Enable/disable error handling procedure
SARE	01101	2	Release sampling tube
RDAD	01110	3	Read the contents of specified board address and put it into Scientific Data
WRAD	01111	4	Write the specified word in the specified address
ENEM	10000	2	End of emergency
MHIT	10001	2	Managing HITB acquisition
LDMP	10010	6	Load a SD2 command sequence from CDMS memory to SD2 internal buffer
STARTOP	10011	2	Notifies that the specified operation is starting. The SD2 current status of BCK_Data is accordingly updated
STOPOP	10100	2	Notifies that the specified operation has been completed. The SD2 current status of BCK_Data is accordingly updated and the end-of operation is notified to CDMS via SR flag
DELAY	10101	3	Performs the required delay before to execute next command
LANDG	10110	10	Loads the data relevant to the landing gear
DRTT	10111	4	Perform drill translation with timeout
DRTC	11000	4	Check drill translation, main or redundant