Genetic algorithms for satellite scheduling problems

Fatos Xhafa[a,*], Junzi Sun[b], Admir Barolli[c], Alexander Biberaj[d] and Leonard Barolli[e]

[a]*Department of Languages and Informatics Systems, Technical University of Catalonia, Barcelona, Spain*
[b]*Centre de Tecnologia Aeroespacial, Barcelona, Spain*
[c]*Seikei University, Tokyo, Japan*
[d]*Polytechnic University of Tirana, Tirana, Albania*
[e]*Fukuoka Institute of Technology, Fukuoka, Japan*

Abstract. Recently there has been a growing interest in mission operations scheduling problem. The problem, in a variety of formulations, arises in management of satellite/space missions requiring efficient allocation of user requests to make possible the communication between operations teams and spacecraft systems. Not only large space agencies, such as ESA (European Space Agency) and NASA, but also smaller research institutions and universities can establish nowadays their satellite mission, and thus need intelligent systems to automate the allocation of ground station services to space missions. In this paper, we present some relevant formulations of the satellite scheduling viewed as a family of problems and identify various forms of optimization objectives. The main complexities, due highly constrained nature, windows accessibility and visibility, multi-objectives and conflicting objectives are examined. Then, we discuss the resolution of the problem through different heuristic methods. In particular, we focus on the version of ground station scheduling, for which we present computational results obtained with Genetic Algorithms using the STK simulation toolkit.

Keywords: Genetic algorithms, satellite scheduling, ground station, mission operations, multi-objective optimization, STK toolkit

1. Introduction

Mission operations arise by the need to coordinate communications of spacecrafts (extra-planetary crafts including satellites, space stations, etc.) with ground stations. Operation teams request an antenna at a ground station for a specific time window. The number of requests could be large but most importantly different requests maybe conflicting making it very complex to manually compute the time windows for communication of spacecrafts with ground stations.

Clear examples of the real need for automation of the allocation process are ESA (European Space Agency) and NASA. ESA manages several ground stations to support its own missions as well as other missions upon industry costumers request. The number of ground stations [2] is rather limited, actually less than ten, supporting the ESA's missions (Kourou (French Guiana), Maspalomas, Villafranca and Cebreros (Spain), Redu (Belgium), Santa Maria (Portugal), Kiruna (Sweden), Perth and New Norcia

---

*Corresponding author: Fatos Xhafa, Department of Languages and Informatics Systems, Technical University of Catalonia, Jordi Girona 1–3, 08034 Barcelona, Spain. E-mail: fatos@lsi.upc.edu.

(Australia), Malargue (Argentina)). ESA missions are mainly scientific missions. As per the study of the mission planning, it is not needed to know specific characteristics of the missions but their information can be used in simulation phase of scheduling systems. Examples of ESA missions include CLUSTER II,, GIOVE-A/GIOVE-B, INTEGRAL, METEOSAT-6/METEOSAT-7, and XMM-NEWTON [10]. ESA uses the ESA tracking station network (ESTRACK) [8], which is a worldwide system of ground stations providing links between satellites in orbit and Operations Control Center. The scheduling of operations for spacecraft-ground stations communications is currently used but has shown several limitations, the most significant one being the need for human coordination and a labour intensive activity for manual organization of the mission schedules. Certainly, with the increasing number of satellites and the growing number of requests for more observations, manual organization cannot tackle the complexity of the problem.

In the USA, besides NASA whose fleet of Earth Observing Satellites (EOSs) keeps growing to meet demands of assisting scientists in their research activities, other examples where satellite scheduling arises include the Air Force Satellite Control Network (AFSCN). The AFSCN accounts for more than 100 satellites and 16 antennae located at nine ground stations. Interested users request a ground station for use during a preferred time window, while specifying alternative time slots due to the large number of requests received and possible time-window overlapping conflicts among them (typically about a few hundreds requests/a hundred of conflicts per day) [6].

Finally, there have been reported a growing number of other satellite missions from projects in research institutions and universities (e.g. Berkeley mission supported by NASA) that need intelligent systems to automate the allocation of ground station services to space missions. Indeed, with fast advancements in satellite networks [9] the number of applications based on satellite networks is increasing in many domains [18–20].

Despite their particularities, common to all these mission operations projects is the requirement to make maximum usage of ground stations (reducing their idle time) in order to support the largest possible number of space missions. Scheduling problem then arises in a variety of forms, including ground station scheduling (with its variants of one ground station, multiple ground station), satellite range scheduling, AFSCN scheduling, LEO satellite scheduling [3,14,21]. All scheduling variants, in their general formulations are highly constrained problems and have been shown computationally hard [16]. Other scheduling models related to on-air information which uses satellite technology has also been reported in the literature [22].

In this paper, we present and analyse some relevant formulations of the satellite scheduling viewed as a family of problems and identify various forms of optimization objectives. The main complexities, due highly constrained nature, windows accessibility and visibility, multi-objectives and conflicting objectives are examined. Then, we discuss the resolution of the problem through different heuristic methods. In particular, we focus on the version of ground station scheduling, for which we present computational results obtained with Genetic Algorithms using the STK simulation toolkit.

The rest of the paper is organized as follows. In Section 2 we describe some basic terminology and concepts from satellite scheduling domain. Different versions of the problem are presented in Section 3. For each version, we identify the main characteristics and its computational complexity. We consider in more details the case of ground station scheduling as a multi-objective optimization problem. The heuristic resolution methods that can be used for near-optimally solving the problem are given in Section 5. We present a GA for the problem and computational results for its evaluation in Subsection 5.1. We end the paper in Section 6 with some conclusions and remarks for future work.

2. Terminology and preliminary concepts

2.1. Tasks

A task is an object which is scheduled. In satellite scheduling tasks refer to observations, communications, manoeuvres, imaging, taking measurements, uplinks, downlinks, etc. Tasks have specified a starting time and a duration time for their completion. Tasks could have specific requirements on the type of the resource they need to be completed (a resource may not execute all types of tasks). Tasks could be given priorities and, in some cases (e.g. imaging tasks) could be specified location information.

2.2. Resources

In satellite scheduling resources comprise satellites, earth observing satellites, ground stations, instruments, recording devices, transmitters, etc.

The ground station and spacecraft are the main resources in satellite scheduling.

2.3. Ground Station (GS) and Spacecraft (SC)

GSs are terrestrial terminals designed for extra-planetary communications with SCs and in data processing [12]. SCs are extra-planetary crafts, such as satellites, probes, space stations, orbiters, etc. Ground stations communicate with a spacecraft by transmitting and receiving radio waves in high frequency bands (e.g. microwaves). A ground station usually contains more than one satellite dish. Each dish is usually assigned to a specific space mission. With the scheduling from control center, dishes are able to handle and switch among mission spacecrafts.

2.4. Mission operation

These are activities that include payload operations (e.g., using a sensor on the satellite to collect data), bus operations (e.g., maintaining the health and status of the vehicle) and communications operations (e.g., transmitting data between satellites or to the ground and receiving information or commands from a ground station) [14].

2.5. Mission requirements

Time duration is a main requirement of mission operations. Depending on the mission, the time required for links can possibly range from less than an hour to 8 hours per day. Besides the time required, there are also other kinds of mission operation requirements, which should be taken into account in the scheduling process. Following is an example of partial requirements for two missions M1 and M2, for which we assume that they share the same ground station G1. Due to the usage of other SCs, it also involves the use of another ground station G2:

1. Lunar occultation periods have to be avoided
2. The minimum pass duration is 3-hours for M1, selected from within the physical station visibility
3. The minimum pass duration is 3-hours for M2, selected from within the physical station visibility
4. The separation of pass should be 24 hours $\pm$ 30 minute
5. The maximum separation of passes should be 27 hours
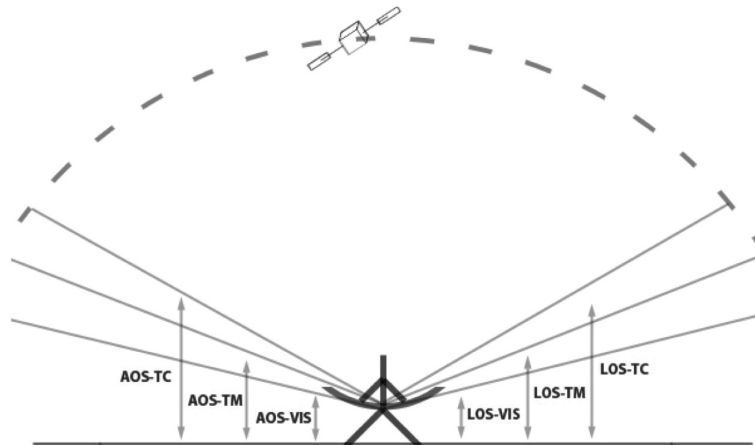6. The minimum pass elevation should be 5°

Fig. 1. Illustration of a GS AOS and LOS angles.

7. M1 and M2 pass should be scheduled within a period of 8 hours. It is expected a ground station reconfigure time between spacecrafts of less than 30 minutes, including the pre-pass test
8. The order M1-M2 or M2-M1 should be retained until a change is requested
9. During some period, due to the load of G1, it may required to support of M1 and M2 spacecraft from G2. The selection of SC support by G2 should be maintained for the full duration of the further analysis.

2.6. Visibility

A ground station can communicate with a SC only when the satellite is within the transmitting angle of the ground station. A spacecraft has three types of *visibility* to a ground station, namely:

– AOS-VIS: Acquisition of Signal, Visible. This indicates the time when the SC appears in the line of sight of the GS.
– AOS-TM: Acquisition of Signal, Telemetry. This is time when GS can start receiving telemetry signals from the SC.
– AOS-TC: Acquisition of Signal, Tele-command. This is time when GS is allowed to send signal to SC.

The visibility angle of each case is defined with the regulations of GS, so that the high amount of transmitting energy will not harm the organisms on the ground. Similarly defined, we also have Loss of Signals, i.e. LOS-TC, LOS-TM, LOS-VIS. In this work we are concerned with scheduling the events ranging from TAOS-TC to TLOS-TC. The scheduling is concerned with planning of the events ranging from TAOS-TC to TLOS-TC. In the simulation process, to get this data, we will set the elevation angle to 10 degrees for all the spacecraft to simplify the process. Figure 1 shows the relations among all those angles.

2.7. Spacecraft visibility clash

The most common constraint in ground station scheduling is the clash of visibility windows caused by multiple spacecrafts to a single ground station. A visibility clash of two spacecraft happens when the AOS time of second spacecraft starts before the LOS time of first one. Figure 2 shows a two-day visibility for 5 spacecraft (the horizontal axis is the time, and vertical are the SCs).
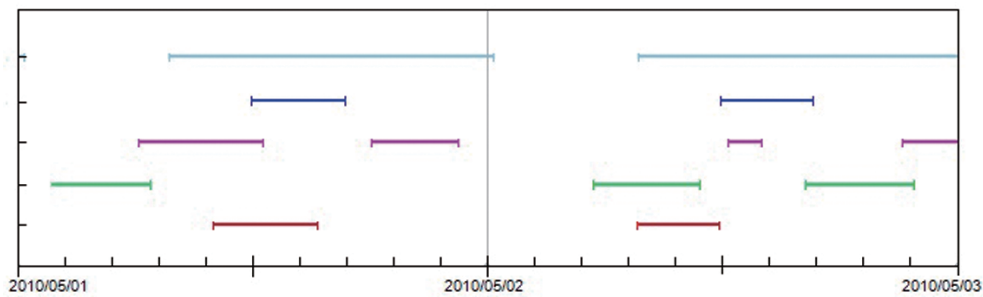
Fig. 2. Visibility clashes of 2 days for 5 spacecrafts.

3. The family of mission operations scheduling problems

Despite the large body of research on scheduling, mission operations scheduling can be seen as a family of problems in its own. We describe next some representative formulations of the problem, identify the complexities and resolution methods proposed in the literature.

3.1. Satellite scheduling

3.1.1. Description and characteristics

Satellite scheduling is the problem of planning tasks to resources. The particularities of this scheduling problem with regard to traditional scheduling problems from manufacturing or distributed systems consists in the fact that tasks cannot be executed at any time, but at specific times, usually specified as time-windows. Additionally, it is an over-constrained problem because not all requests can be scheduled within the specified slot times and thus conflicts should be minimized. One particular case of satellite scheduling arises when satellites are Low Earth Orbit satellites. In this case the time window may vary from a few minutes to only a small fraction of a second.

3.1.2. Constraints

Constraints arise due when and where tasks can be scheduled as well as due to limited resource capacity and availability. In [14], there have been identified three types of satellite scheduling constraints: (a) task constraints, (b) resource constraints and (c) event constraints. Task constraints refer to possible dependencies among tasks (e.g. a task should be scheduled after another task – see Section 2). Resource constraints refer to their processing capabilities (the type of the tasks they can execute). Event constraints refer to time windows in which a task can be executed. This kind of constraint is due spacecrafts have windows visibility to a ground station, and thus can communicate only during concrete window times.

3.1.3. Modelling

Given that the problem is highly constraints, the CSP – Constraint Satisfaction Problem is among first used for modelling and resolution purposes. The resulting instances, nevertheless, cannot be solved to optimality due to the computational hardness of CSP.

3.2. Satellite range scheduling

3.2.1. Description and characteristics

Satellite range scheduling [6] is another flavour of the satellite scheduling problem for a rather concrete scenario: coordinate communications between civilian and military organizations using more

than 100 satellites, 16 antennas located at nine ground stations around the globe. Customers request to reserve an antenna at a ground station for a specified time window to complete two types of task(s): low altitude (usually requiring access to low attitude satellites and having short duration) and high altitude (requiring access to high altitude satellites and having longer window times for completion).

3.2.2. Constraints

As in the case of general satellite scheduling, the constraints are task constraints, resource constraints and time-window (events) constraints. Additionally, some works in literature have considered requests with priorities, fixed start times and durations, that is a kind of strict scheduling requirement. In some other cases, requests are more flexible by specifying earliest start time, latest final time and the minimum and maximum duration.

3.2.3. Objectives

The costumers' requests may cause conflicts due more request can arrive than possible to accommodate. The main objective is then minimize the number of request conflicts when scheduling costumers requests. This objective could equivalently be formulated as maximizing the number of non-conflict requests that can be scheduled. In fact, by maximizing the number of scheduled requests, the usage of resources (satellites) could be maximized as well.

3.3. Task scheduling for satellite based imagery

3.3.1. Description and characteristics

This kind of scheduling (also known as photo-reconnaissance satellite scheduling) arises for a particular type of tasks, namely, when it is needed to schedule tasks to acquire images of the earth surface and scheduling the transmission of the image files to a set of ground stations [11,23]. Two modes of this scheduling can be addressed: static scheduling and dynamic scheduling. In the former, the imaging satellites are to execute tasks according to a schedule *a priori* computed. In the later case, the objective is to increase the resource (satellite) usage dynamically according to the tasks requirements and resource availability.

3.3.2. Constraints

The time windows for task completion in this kind of scheduling are of about one-day horizon. Again resource constraints should be taken into account as well as possible preferences of visibility time windows. Additionally, there are constraints about imaging tasks such as transition time between consecutive downloads, memory capacity, etc.

3.3.3. Objectives

The main objective is to maximize the amount of images acquired and transmitted, maximizing resource usage and minimizing conflicts. Indeed, the problem is over-constrained, in that there are more imaging tasks than can be possibly completed in one pass of the satellite, thus in general only a subset of tasks will be completed. Imaging tasks could be given priorities, in which case the objective would be to to maximize the total priority of the tasks.

3.4. Ground station scheduling

3.4.1. Description and characteristics

Ground station scheduling problem arises in spacecraft operations and aims to allocate ground stations to spacecraft to make possible the communication between operations teams and spacecraft systems. The

Table 1
Time eequirements of spacecrafts

SC	From (min)	To (min)	Require (min)	Meaning
1	1	2880	60	1 hour/2 days
1	2881	5760	60	1 hour/2 days
1	5761	8640	60	1 hour/2 days
1	8641	12960	60	1 hour/2 days
2	1	2880	80	80 mins/2 days
2	2881	5760	80	80 mins/2 days
2	5761	8640	80	80 mins/2 days
2	8641	12960	80	80 mins/2 days
3	1	1440	120	2 hours/day
3	1441	2880	120	2 hours/day
3	2881	4320	120	2 hours/day
3	4321	5760	120	2 hours/day
3	5761	7200	120	2 hours/day
3	7201	8640	120	2 hours/day

problem belongs to the family of satellite scheduling for the specific case of mapping communications to ground stations.

3.4.2. Objectives

Different types of objectives can be formulated, namely, maximizing matching of visibility windows of spacecrafts to communicate with ground stations, minimizing the clashes of different spacecrafts to one ground station, maximizing the communication time of the spacecraft with the ground station, and maximizing the usage of ground stations.

3.4.3. Constraints

The most common constraint in ground station scheduling is the clash of visibility windows caused by multiple spacecrafts to a ground station (see Table 1 for an example of time requirements of three spacecrafts).

3.5. System assumptions

In order to make a practical system environment for a Ground Station (GS) – SpaceCraft (SC) schedule system [8], we make the following system assumptions:

– A maximum amount of communication time is defined for each spacecraft.
– 7 real mission spacecrafts orbit information are used (which are Cluster, Integral, XMM, etc. – see Section 1). The SCs are defined as $\bigcup_{i=1}^{7} SC(i)$.
– 4 real ground station locations are selected. (Cebreros, Kourou, Malargue, and New Norica). These GS are defined as $\bigcup_{g=1}^{4} GS(g)$.
– The relation of GSs and SCs are hypothetical.
– Each SC has one communication with GS each day.
– During the 8-day period, one SC will only use one GS.
– The goal of is to allocate the maximum possible amount of communication time for each SC.

3.6. One vs. multiple ground stations

In general, we may assume that there are multiple ground stations available to communicate with spacecrafts. It might be the case that some spacecraft might have no visibility with some ground stations (see Fig. 3 for an example of four ground stations and their window time to spacecrafts).
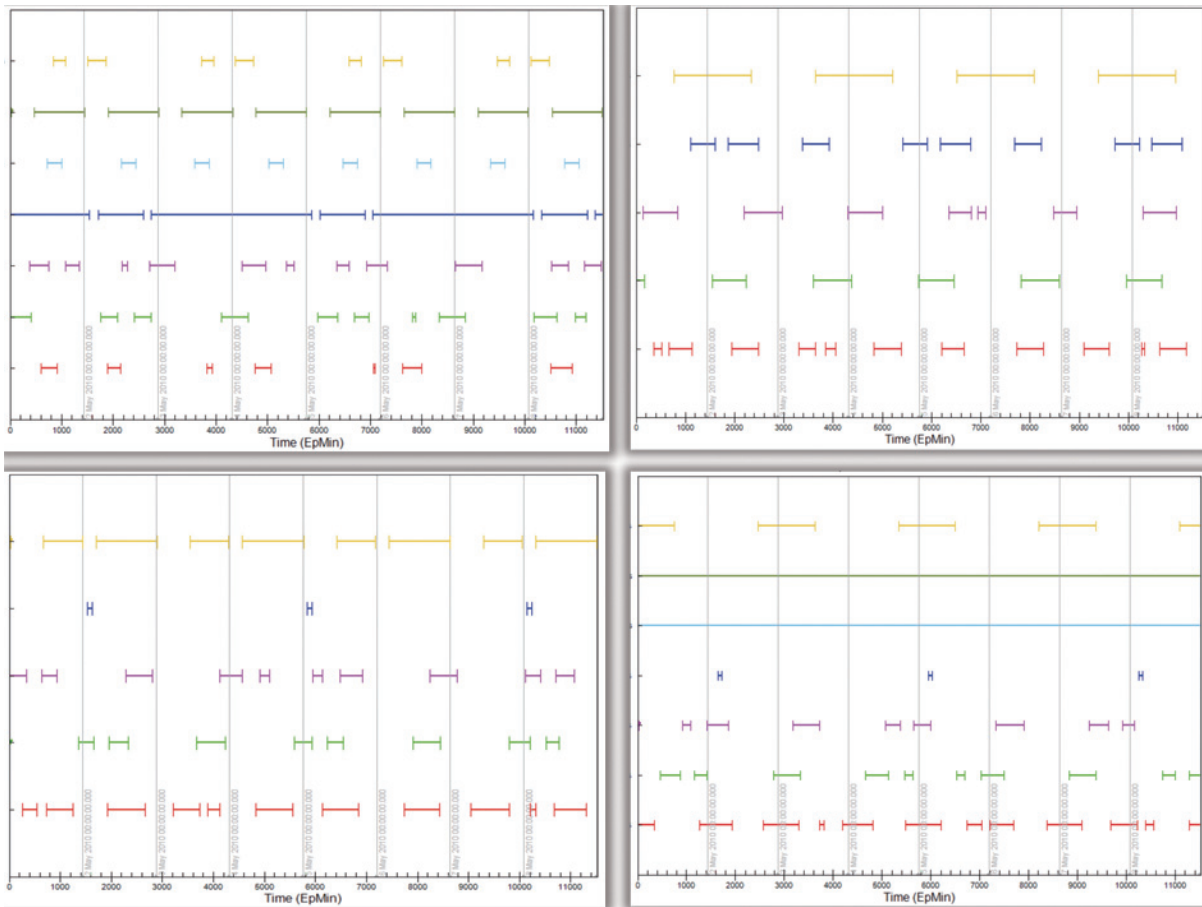
Fig. 3. Four ground stations and their window time to spacecrafts.

4. Scheduling fitness types

4.1. Scheduling fitness

One of the major complexities of the mission operations scheduling comes from the many objectives that can be sought for the problem. These objectives are related to visibility window, communication clashes, communication time, resource usage, among others. The total fitness function, besides being composed of multiple objectives, poses the challenge of how to combine them and in which order to evaluate them. For the combination, one can adopt a hierarchical optimization approach based on the priority of the objectives or a simultaneous optimization approach.

We define next the four objectives that would compose the fitness function and then show possible combinations of them into one fitness function.

4.1.1. Access window fitness

Visibility windows are the time periods when a GS has the possibility to setup a communication link with a SC.
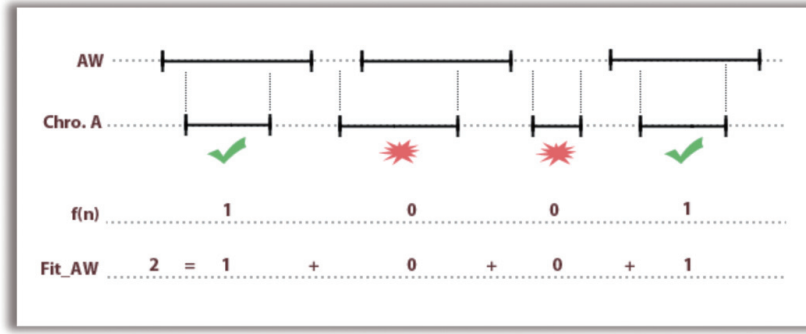
Fig. 4. Access windows fitness.

The objective is that all or the maximum number possible of generated communication links to fall into access windows and thus achieve as many communications as possible. In the following equation, $W_{(g,i)}$ is the Access Window set for Ground Station g and Spacecraft i, $T_{Start}(s)$ and $T_{End}(s)$ are the start and end of each access window.

$$AW(g,i) = \cup_{s=1}^{S}[T_{AOS(g,i)}(s), T_{LOS(g,i)}(s)]$$

Then we define the final Access Window fitness of the scheduling solution (Fit_{AW}) calculated as follows:

$$f(n) = \begin{cases} 1, \text{if } [T_{Start}(n), T_{Start}(n) + T_{Dur}(n)] \subseteq AW(ng, ni), \\ 0 \text{ otherwise.} \end{cases}$$

$$Fit_{AW} = \sum_{n=1}^{N} f(n)$$

where n value corresponds to an event and N is the total number of events of an entire schedule (see Fig. 4).

4.1.2. Communication clashes fitness

Communications clash represents the event when the start of one communication task happens before the end of another one on the same ground station. The objective is to minimize the clashes of different spacecrafts to one ground station. To compute the number of clashes, SCs are sorted by their start time. If as a result of the sorting:

$$T_{Start}(n+1) < T_{Start}(n) + T_{Dur}(n), \quad 1 \leqslant n \leqslant N - 1$$

where n value corresponds to an event and N is the total number of events of an entire schedule, then there is a clash. The fitness will be reduced, and one of the clashed entries has to be removed from the solution. The total fitness of communication clashes is then:

$$f(n) = \begin{cases} -1, \text{if } T_{Start}(n+1) < T_{Start}(n) + T_{Dur}(n), \\ 0 \quad \text{otherwise.} \end{cases}$$

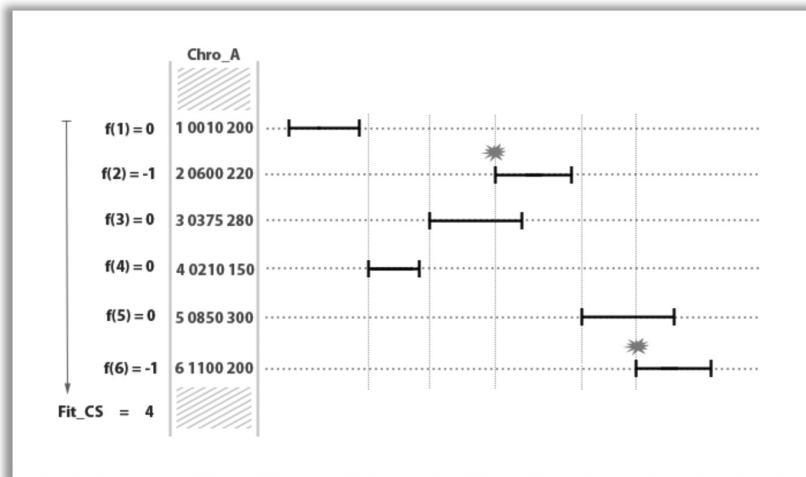$$Fit_{CS} = N + \sum_{n=1}^{N} f(n)$$

Figure 5 shows an example.

Fig. 5. Fitness communication clashes.

$$\begin{array}{llll}
SC & From & To & T_{REQ}[Min] \\
\begin{bmatrix} SC[1] & T_{From}[1]_1 & T_{TO}[1]_1 & T_{REQ}[1]_1 \\
SC[1] & T_{From}[1]_2 & T_{TO}[1]_2 & T_{REQ}[1]_2 \\
\vdots & & & \\
SC[1] & T_{From}[1]_N & T_{TO}[1]_N & T_{REQ}[1]_N \\
SC[2] & T_{From}[2]_1 & T_{TO}[2]_1 & T_{REQ}[2]_1 \\
SC[2] & T_{From}[2]_2 & T_{TO}[2]_2 & T_{REQ}[2]_2 \\
\vdots & & & \\
SC[i] & T_{From}[i]_n & T_{TO}[i]_n & T_{REQ}[i]_n \\
\vdots & & & \\
SC[I] & T_{From}[I]_N & T_{TO}[I]_N & T_{REQ}[I]_N \end{bmatrix}
\end{array}$$

Fig. 6. Matrix representation of periodic tasks.

4.1.3. Fitness – Communication time requirement

The objective is to maximize the communication time of the spacecraft with the ground station so that every spacecraft $SC(i)$ will communicate at least $T_{req}(k)$ time.

A sufficient amount of time should be granted for TTC (Telemetry, Tracking and Command). Depending on the mission type, there are different amount of data needed to be downloaded from the spacecraft. For example, satellites that need to download huge amount of image data require more time for linking with ground station. These communications, especial for data download tasks are usually periodical tasks (e.g. 2 hours communication for SC1 each day, 5 hours data downlink for SC2 every 2 days, etc.) A matrix is used to define those requirements, which is used as the input for the scheduling system (see Fig. 6 for an example.)

The fitness is calculated by summing up all the communication link durations of each spacecraft, and dividing them in the required period to compare if the scheduled time matches requirements, using the
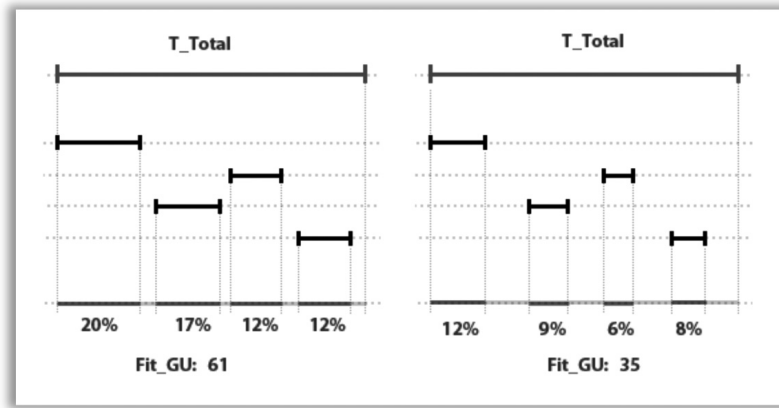
Fig. 7. Ground station usage.

following equations.

$$T_{Start}(m) > T_{From(k)}$$

$$T_{Start}(n) + T_{Dur}(n) < T_{TO}(k)$$

$$T_{Comm}(k) = T_{Dur}(j)$$

$$f(k) = \begin{cases} 1, \text{if } T_{Comm}(k) \geqslant T_{REQ}(k), \\ 0 \text{ otherwise.} \end{cases}$$

$$FIT_{TR} = \sum_{k=1}^{K} f(k)$$

4.1.4. Ground station usage fitness

Given that the number of ground stations is usually much smaller than the number of spacecrafts missions, the objective is to maximize the usage of ground stations, that is, try to reduce the idle time of a ground station. A maximized usage would contribute to provide additional time for SC communications (Fig. 7).

This fitness value is calculated as the percentage of ground stations occupied time by the total amount of the possible communication time. The more a GS is used, the better is the corresponding schedule.

$$Fit_{GU} = \frac{\sum_{n=1}^{N} T_{Dur}(n)}{\sum_{g=1}^{G} T_{Total}(g)} \times 100$$

where N is the number of events of an entire schedule, G is the number of ground stations and $T_{Total(g)}$ is the total available time of a ground station.

4.1.5. Hierarchic optimization model

In this model, the objectives are classified (sorted) according to their priority. Thus, for a problem having k objectives sorted as follows:

$$f_1 \succ f_2 \succ \cdots \succ f_k$$

means that f_1 is the most important objective and f_k is the least important objective. The optimization procedure would first optimize according to f_1 until no further improvements are possible. Then, the algorithm optimizes according to f_2 subject to not worsening the value achieved for f_1, and so on.

This model is useful when for the design or deployment needs some parameters (objectives) are considered of more priority than others. The disadvantage of this model is that the final solution computed by the optimization procedure could be far from optimal for the less priority objectives.

4.1.6. Simultaneous optimization approach

In the simultaneous optimization approach, all objectives are simultaneously optimized. Thus, for a problem having k objectives $f_1, f_2, \ldots, f_k$, the optimization procedure tries to optimize at the same time all the objectives, which actually leads to computing the so called *Pareto front* which contains the optimal solutions. In some cases, it could be possible to apply the *sum model* in which the k objectives are reduced to two objectives:

$$f = \lambda_1 f_1 + \lambda_2 f_2 + \cdots + \lambda_k f_k, \quad \sum_{i=1}^{i=k} \lambda_i = 1, \ \lambda_i > 0.$$

4.1.7. Combination of fitness objectives

The fitness objectives defined above (FIT_{AW}, FIT_{CS}, FIT_{TR}, FIT_{GU}) are conceived as fitness modules so as to facilitate the design phase of the scheduler to easily plug-in other fitness objectives. From the definition of the fitness objectives, we can observe that some of them can be applied in serial fashion (due dependencies, denoted serial-FM), while some others can be applied in parallel (denoted parallel-FM). Thus, in a hierarchical mode, one possible way to arrange fitness checking is that of Fig. 8.

With regard to the simultaneous combination of objectives, one can either consider a proper Pareto-front approach, or combine all the fitness modules into one total fitness function using weights for different fitness module:

$$Fit = \sum_{i=1}^{n} w_i \cdot Fit_S(i) + \sum_{j=1}^{m} w_j Fit_P(j)$$

where w_i, w_j are the weights of fitness modules, $Fit_S(i)$ and $Fit_P(j)$ are the fitness values from Serial-FMs and Parallel-FMs, and n, m are the number of fitness modules, resp.

5. Resolution methods

Most formulations of the satellite scheduling problem are NP-hard, and thus it is unlikely to find polynomial time algorithms for finding optimal solutions. Several heuristics algorithms can be used to search for near-optimal scheduling solutions. Some approaches proposed in the literature include Genitor [4], Branch-and-Bound Algorithm [4], Graph Colouring [21], Tabu Search [15], Hill Climbing [4], Fuzzy techniques [1].

Genetic Algorithms are among most successful algorithms for efficiently tackling with the complexity of computationally hard problems from networking domain [7,13]. We present in the next section Genetic Algorithms for the ground station scheduling and report some computational results for the version with multiple ground stations.
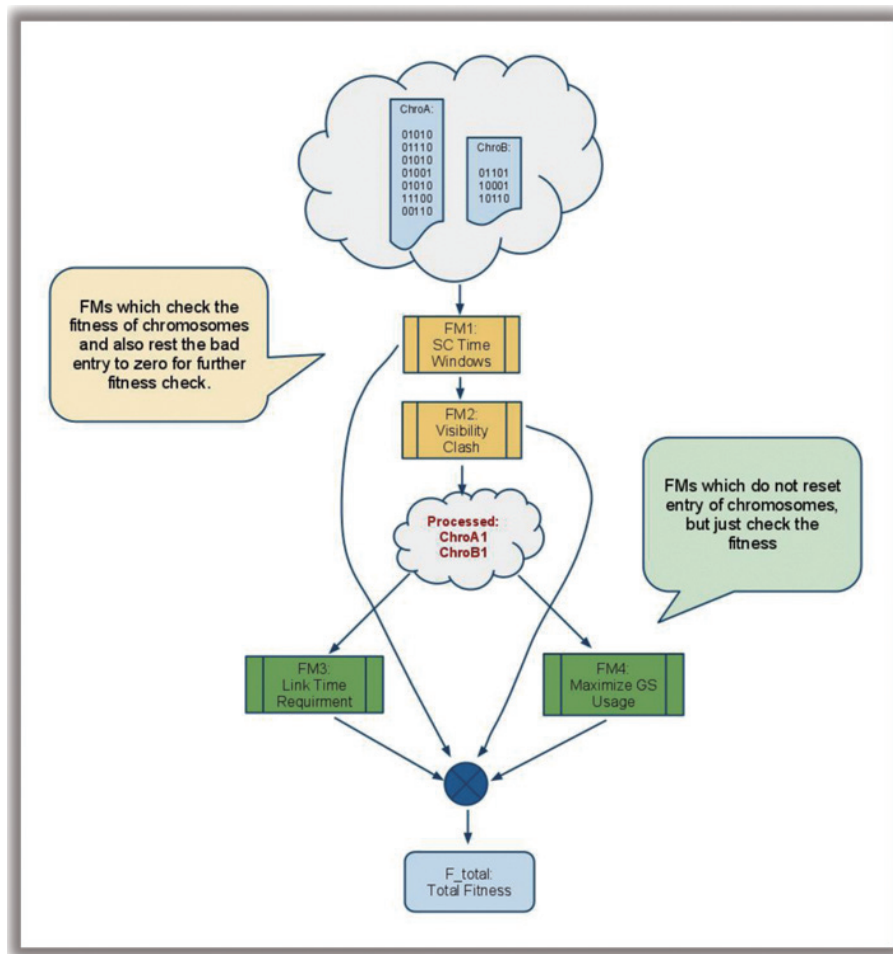
Fig. 8. Combination of fitness modules.

5.1. Genetic algorithm for ground station scheduling

We have used the *template* given in algoritm 1.
We briefly present next the main features of GAs.

Algorithm 1 Genetic algorithm template

Generate the initial population P^0 of size μ;
Evaluate P^0;
while not termination-condition **do**
  Select the parental pool T^t of size λ; $T^t := Select(P^t)$;
  Perform crossover procedure on pairs of individuals in T^t with probability p_c; $P_c^t := Cross(T^t)$;
  Perform mutation procedure on individuals in P_c^t with probability p_m; $P_m^t := Mutate(P_c^t)$;
  Evaluate P_m^t ;
  Create a new population P^{t+1} of size μ from individuals in P^t and/or P_m^t;
  $P^{t+1} := Replace(P^t; P_m^t)$
  $t := t + 1$;
end while
return Best found individual as solution;

Chromosome A :   *Chromosome B* :

$$
\begin{bmatrix}
SC[1], & T_{Start}, & T_{Dur} \\
SC[2], & T_{Start}, & T_{Dur} \\
\vdots \\
SC[i], & T_{Start}, & T_{Dur}
\end{bmatrix}
\quad
\begin{bmatrix}
SC[1], & GS[g_1] \\
SC[2], & GS[g_2] \\
\vdots \\
SC[i], & GS[g_i]
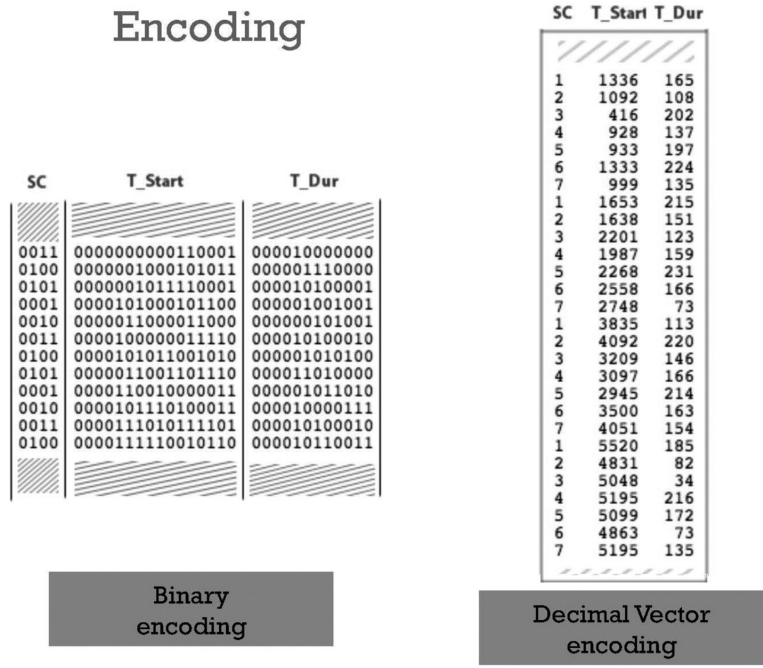\end{bmatrix}
$$

Fig. 9. Chromosome representations.

Fig. 10. Chromosome encoding.

5.1.1. Population of individuals

Unlike local search techniques that construct a path in the solution space jumping from one solution to another one through local perturbations, GAs use a population of individuals giving thus the search a larger scope and chances to find better solutions. This feature is also known as "exploration" process in difference to "exploitation" process of local search methods.

5.1.2. Fitness

The determination of an appropriate fitness function, together with the chromosome encoding are crucial to the performance of GAs. Ideally we would construct objective functions with "certain regularities", i.e. objective functions that verify that for any two individuals which are close in the search space, their respective values in the objective functions are similar.

5.1.3. Selection

The selection of individuals to be crossed is another important aspect in GAs as it impacts on the convergence of the algorithm. Several selection schemes have been proposed in the literature for selection operators trying to cope with premature convergence of GAs.
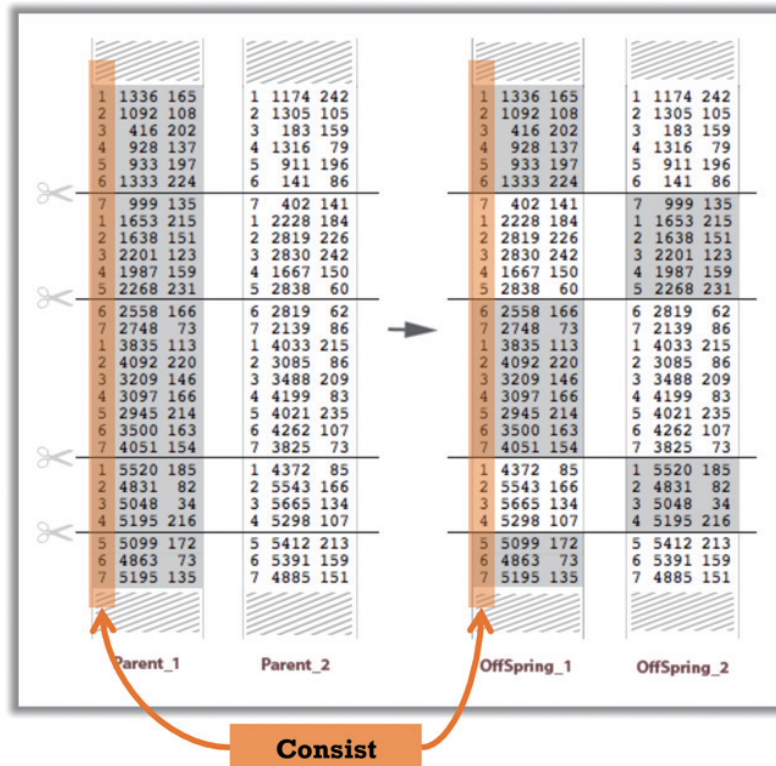
Fig. 11. Crossover operator (consistency information between parent 1 and offspring 1).

5.1.4. Crossover operators

Use of crossover operators is one of the most important characteristics. Crossover operator is the means of GAs to transmit best genetic features of parents to offsprings during generations of the evolution process.

5.1.5. Mutation operators

These operators intend to improve the individuals of a population by small local perturbations. They aim to provide a component of randomness in the neighbourhood of the individuals of the population.

5.1.6. Escaping from local optima

GAs have the ability to avoid falling prematurely into local optima and can eventually escape from them during the search process.

5.1.7. Convergence

The convergence of the algorithm is the mechanism of GAs to reach to good solutions. A premature convergence of the algorithm would cause that all individuals of the population be similar in their genetic features and thus the search would result ineffective and the algorithm getting stuck into local optima. Maintaining the diversity of the population is therefore very important to this family of evolutionary algorithms.
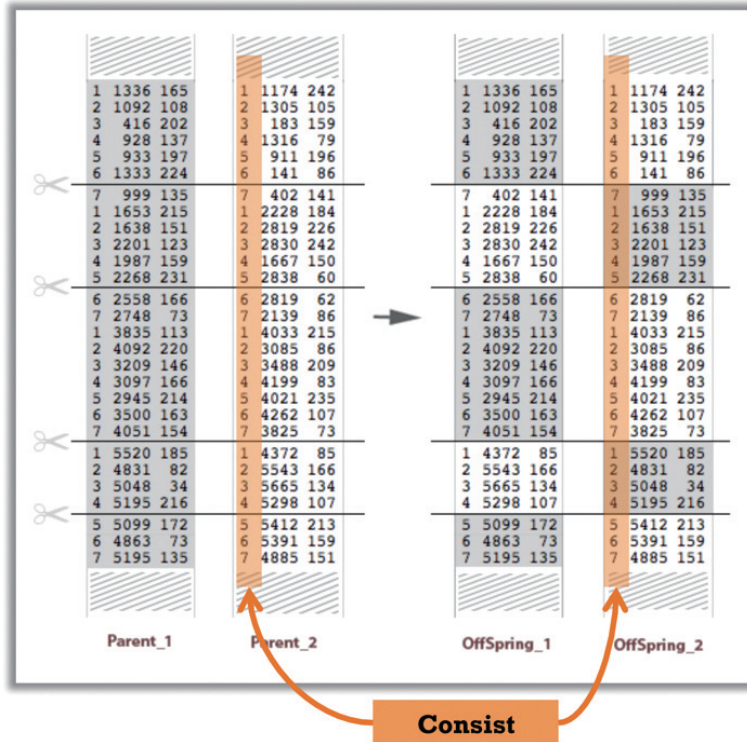
GA operator: Crossover

Fig. 12. Crossover operator (consistency information between parent 2 and offspring 2).

5.2. GA for ground station scheduling

5.2.1. Chromosome encoding

A chromosome (an individual of the population) encodes the scheduling solution, which consists of (a) encoding the spacecrafts timetable and (b) encoding the pairs of GS-SC. One possibility of encoding would be to encode separately into Chromosome A and Chromosome B this information using binary or decimal encoding schemes (see Figs 9 and 10).

5.2.2. Initial population

Initial population is generated randomly.

5.2.3. Crossover operator

Multi-points crossover has been selected as appropriate because of the length of chromosome we are dealing is usually large (e.g. an 8-Day, 7-SC, scheduling can have a vector chromosome size of 56.) We show in Figs 11 and 12 an example of crossover. As can be seen it is important to ensure consistency when passing the information from parents to offsprings.

5.2.4. Mutation operator

Mutation makes some small local perturbations of an individual. Specifically, we only mutate the time of communication (T_{Start} and T_{Dur}); it's worth noting that the bits corresponding to $SC[i]$ cannot be changed. We show in Fig. 13 an example.

Table 2
System input

Parameters	Descriptions
$\bigcup SC(i)$	Spacecrafts that need to be scheduled in the system
$\bigcup GS(g)$	Ground station resources for spacecrafts
$\bigcup N_{days}$	Number of days for one scheduling period
$\bigcup T_{AOS}(g)(i))$	AOS of a SC to GS
$\bigcup T_{LOS}(g)(i))$	LOS of a SC to GS
$\bigcup T_{req}(i))$	Communication time required for each SC

Table 3
System output

| Parameters	Descriptions
$\bigcup_{All} T_{start}(i)$	All start time of communication for SC(i) during the schedule period
$\bigcup_{All} T_{dur}(i)$	All communication duration for SC(i) during the schedule period
$\bigcup SC/GS(i, g)$	Pairs of ground station and spacecraft
Fit_{CC}	Fitness value for minimizing communication clashes
Fit_{VW}	Fitness value for meeting visibility windows of all spacecrafts
Fit_{TR}	Fitness value for fulfilling mission time requirements
Fit_{GU}	Fitness value for maximizing ground station usage
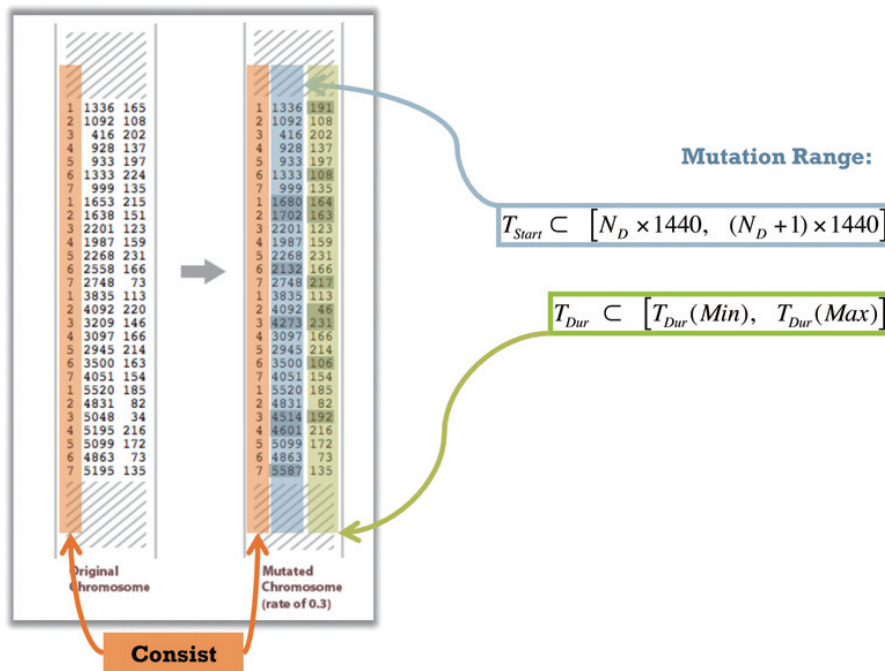
Fig. 13. Mutation operator (feasibility for visibility windows).

5.2.5. Selection operators

The selection method determines which individuals will survive and produce offspring according to their fitness score. The selection methods considered in this work are Fitness Proportionate Selection, Stochastic Universal Sampling, and Tournament Selection.
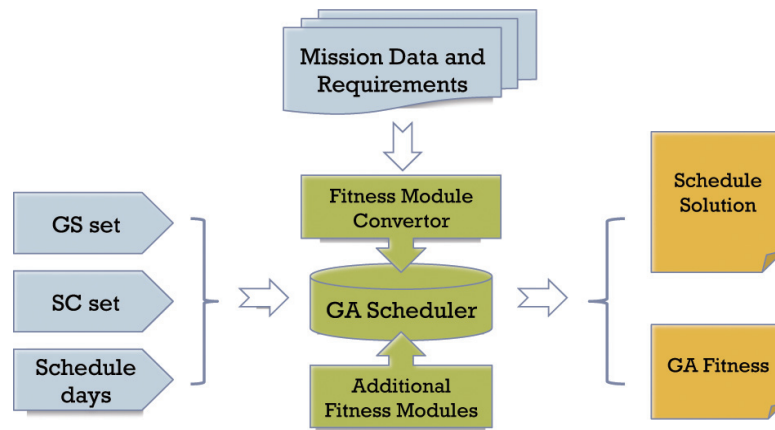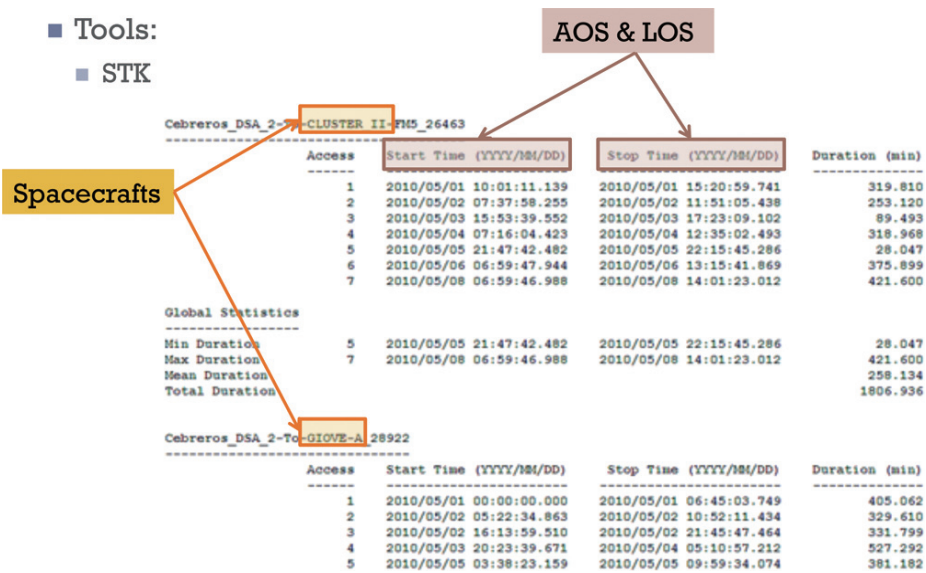
Fig. 14. Scheduling system model.

Fig. 15. Visibility windows.

5.3. Scheduling system model

The scheduling system, using Genetic Algorithms as solver, can be modelled as shown in Fig. 14.

As can be seen from the model, the scheduling system receives in input a set of parameters, which consists of ground stations information, spacecrafts information and information on the schedule days (see Table 2). Additionally, the scheduling system receives other constraints (spacecraft constraints, ground station constraints, and mission data requirements).

Then, the scheduling system, using GA as solver, outputs the pairs of ground stations − spacecrafts, the starting time of communication among ground stations and spacecrafts, the duration time of the communication scheduled (see Table 3). The scheduling system also outputs different fitness values associated with the scheduling such as fitness value that minimizes the communication clashes, fitness value of matching the visibility windows of all spacecrafts and fitness value of maximizing the ground
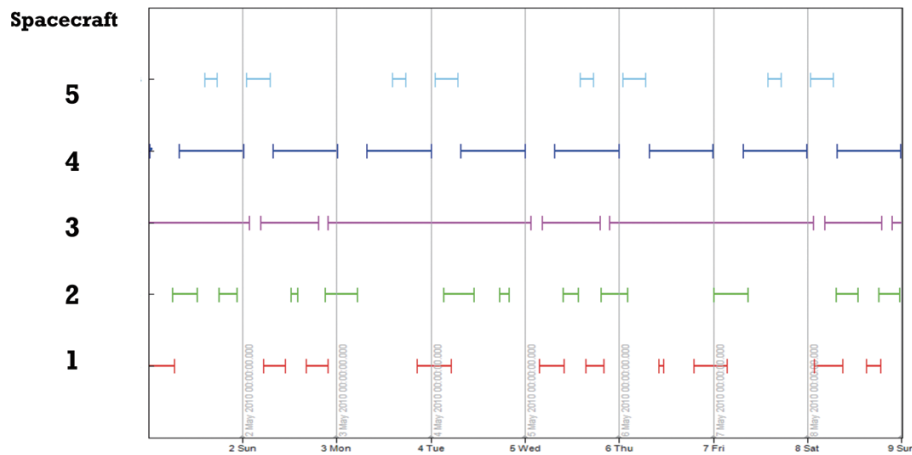
Fig. 16. Ground station and its satellites visibility windows.

$$\begin{array}{ccc} SC & AOS & TOS \\ \hline SC[1], & T_{AOS}[1]_1, & T_{LOS}[1]_1 \\ SC[1], & T_{AOS}[1]_2, & T_{LOS}[1]_2 \\ \vdots & & \\ SC[1], & T_{AOS}[1]_N, & T_{LOS}[1]_N \\ SC[2], & T_{AOS}[2]_1, & T_{LOS}[2]_1 \\ SC[2], & T_{AOS}[2]_2, & T_{LOS}[2]_2 \\ \vdots & & \\ SC[i], & T_{AOS}[i]_n, & T_{LOS}[i]_n \\ \vdots & & \\ SC[I], & T_{AOS}[I]_N, & T_{LOS}[I]_N \end{array}$$

Fig. 17. The extracted information for spacecrafts.

station usage. This last fitness value is important given that the number of ground stations is a limited resource.

5.4. Data simulation model for ground station scheduling

We have used the Satellite Tool Kit (STK) [17] that allows engineers and scientists to design and develop complex dynamic simulations of real-world problems. It is a powerful software tool to solve problems of Earth-orbiting satellites. We used it to simulate a ground station and its satellites. In order to achieve realistic simulation, we are based on ESA ground stations, and create them with exact position on Earth. Then, based on real ESA space mission, orbits of spacecrafts are added to the scenario.

STK generates a report of the time stamp of visibilities of the ground to its entire associated spacecraft (see Fig. 15.)

At the same time, we can also generate the graphic view of the visibility windows in the entire scheduling period. In Fig. 16, from the bottom to the top, the lines represent the visibility windows of

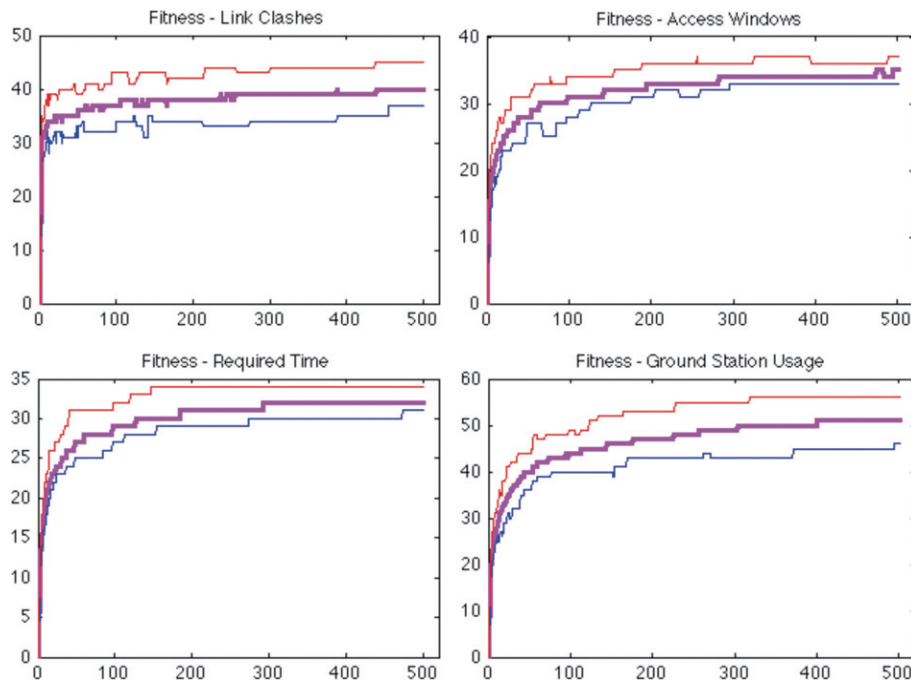| Table 4			Table 5	
GA parameter values			Best fitness results (numeric values)	
GA parameter	Value		Objective	Fitness
Size of population	20		Less link clashed	34
Num. of crossover points	15		Fit to access windows	33
Mutation rate (Start time)	10%		Link time required	24 [4 4 8 4 4]
Mutation rate (Comm. Duration)	10%		Ground station usage	65

Fig. 18. Fitness functions values during 500 generations for a single ground stations and 5 spacecraft.

spacecrafts. From that information we can extract the necessary information for a chromosome (see the corresponding Fig. 17).

5.5. *Computational results*

We present some computational results obtained with GA for the ground stations scheduling for the case of single ground station scenario. Within this scenario, we present results for 5 spacecrafts and 7 spacecrafts. In both cases, a total of 20 independent runs of the GA were performed (under the same parameter configuration) and average results are reported.

5.6. *The case of one ground station and five spacecrafts*

In the case of one ground station and five spacecrafts, for the evaluation of the GA the parameter values used are given in Table 4.

We show in Fig. 18 the graphical representation of different fitness functions. The corresponding schedule is graphically shown in Fig. 19. In the figure, the timeslots of squares are the Access Window
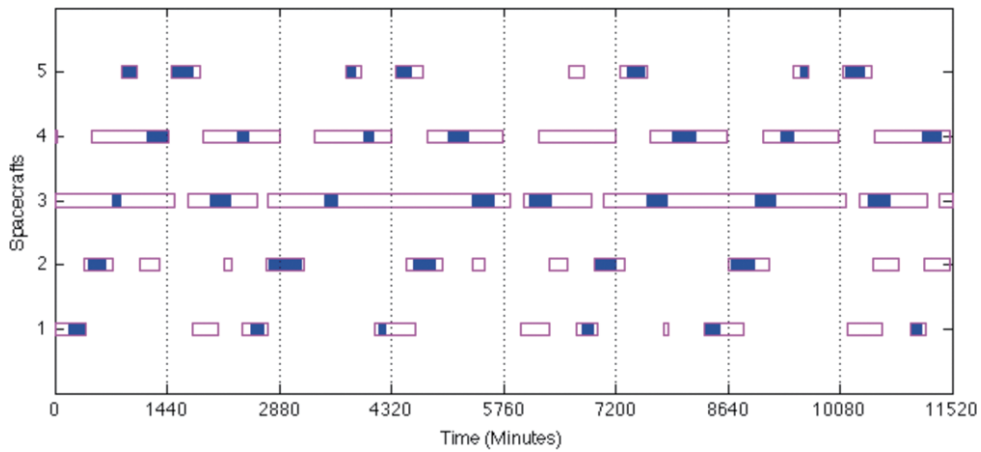
Fig. 19. The schedule obtained from best solution computed by GA for a single ground stations and 5 spacecraft for an eight days period.
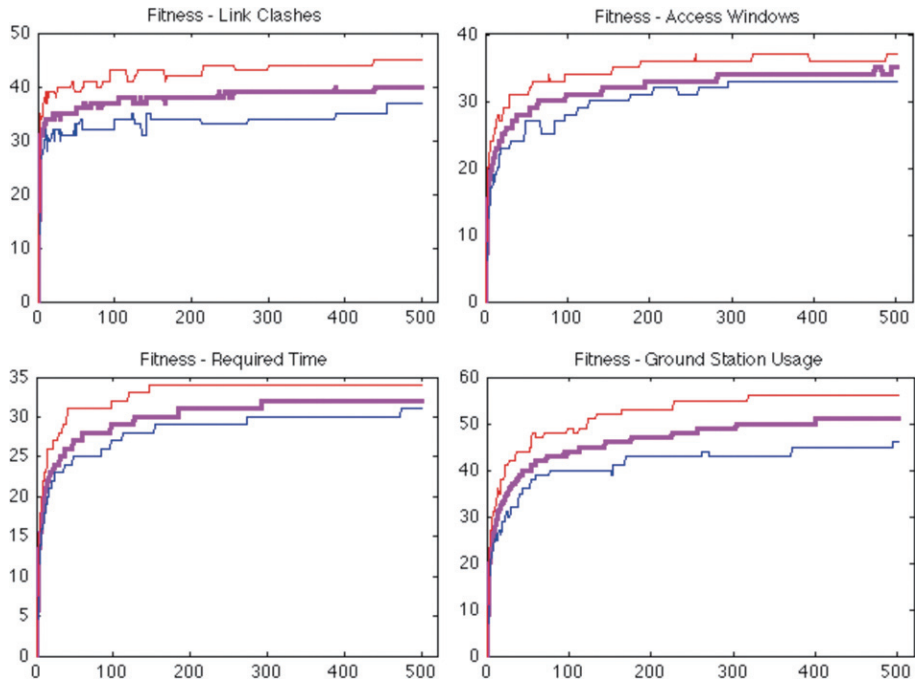

Fig. 20. Fitness functions values during 500 generations for a single ground stations and 7 spacecrafts.

of the ground station to the spacecrafts. While, the solid bars are actually scheduled communication times for each spacecrafts.

The numeric values of the fitness functions are given in Table 5 and the corresponding schedule in Table 6.
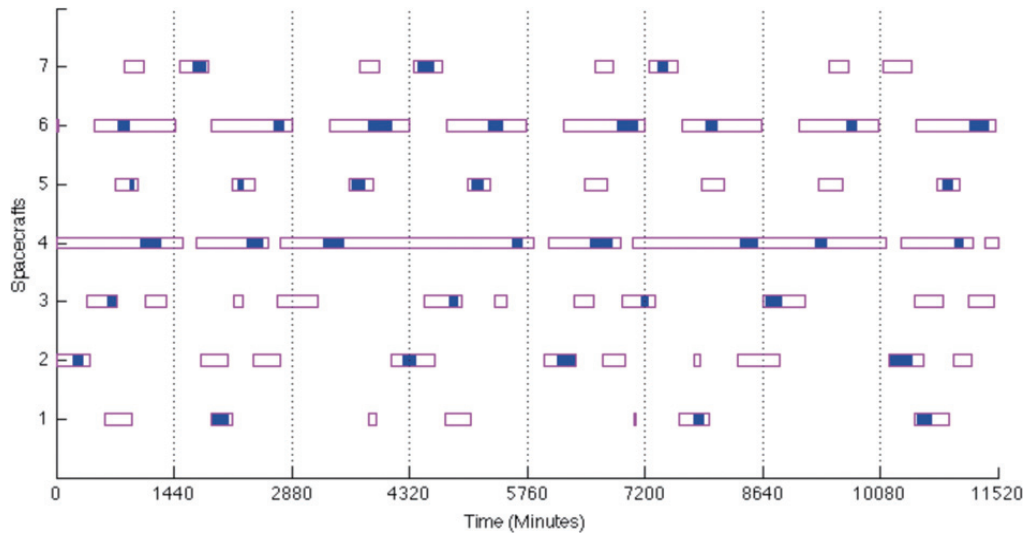
Fig. 21. The schedule obtained from best solution computed by GA for a single ground stations and 7 spacecraft for an eight days period.
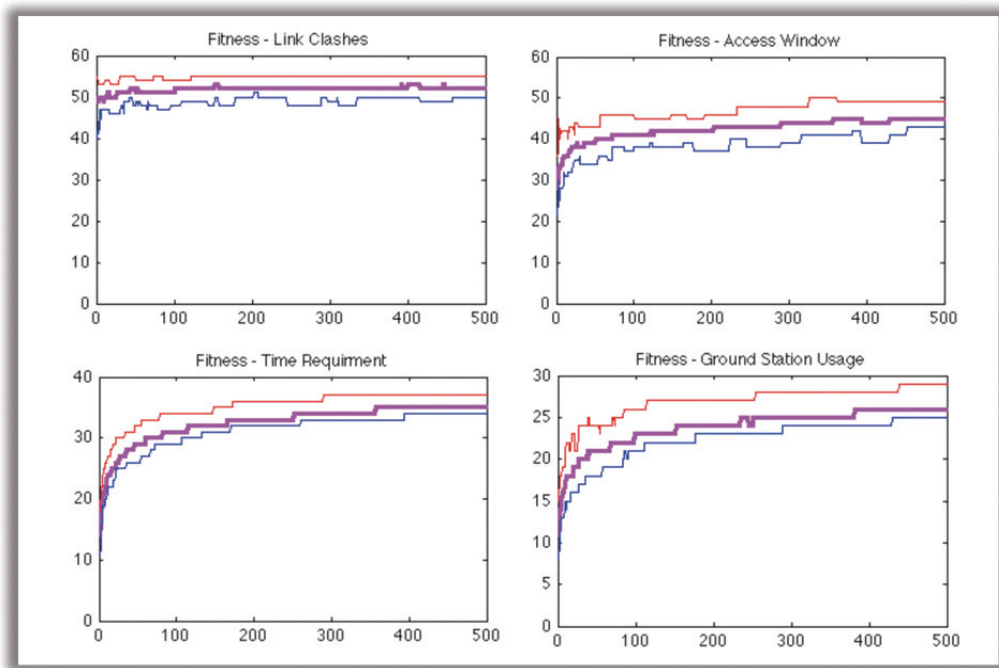

Fig. 22. Fitness functions values during 500 generations for multiple ground stations.

5.7. *The one ground station case*

In this case, for the evaluation of the GA, the parameter values used are given in Table 7.
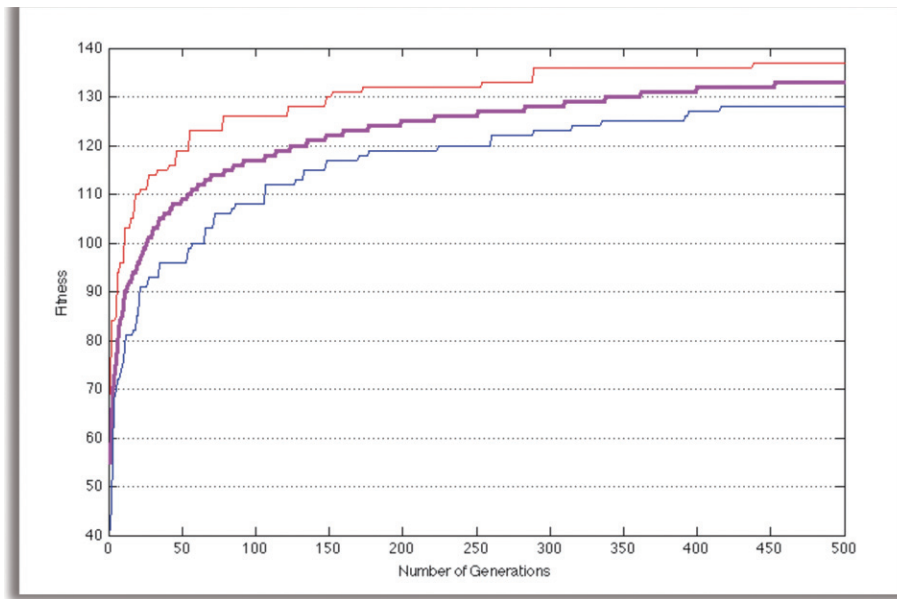We show in Fig. 20 the graphical representation of different fitness functions. The corresponding

Fig. 23. Selection fitness functions values during 500 generations for multiple ground stations.
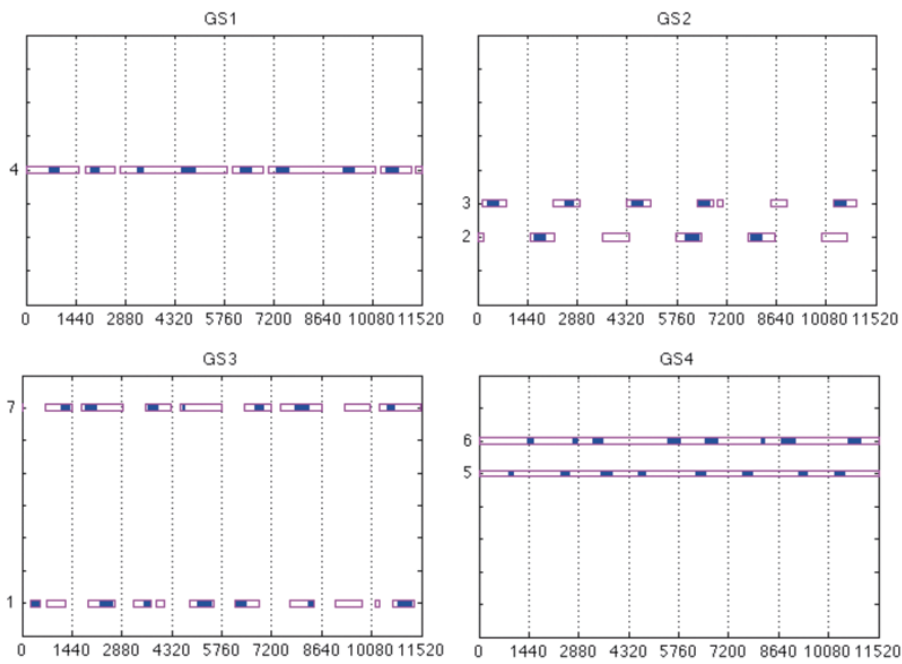

Fig. 24. The schedule obtained from best solution computed by GA for multiple ground stations.

schedule is graphically shown in Fig. 21.

The numeric values of the fitness functions are given in Table 8 and the corresponding schedule in Table 9.

Table 6
The generated schedule for a single GS and 5 SCs

Spacecraft	Link start	Link end
1	183	214
	2525	170
	4165	106
	6769	168
	8350	205
	11000	141
2	437	233
	2760	143
	2910	270
	4612	292
	6943	276
	8694	307
3	735	123
	1996	268
	3472	180
	5361	298
	6089	297
	7601	271
	9001	266
	10447	295
4	1191	261
	2345	152
	3973	141
	5046	287
	7926	309
	9323	184
	11142	255
5	871	185
	1513	283
	3739	141
	4396	186
	7346	243
	9574	95
	10154	259

Table 7
GA parameter values

GA parameter	Value
Size of Population	20
Num. of crossover points	15
Mutation rate	10%

Table 8
Best fitness results (numeric values)

Objective	Fitness
Less link clashed	38
Fit to access windows	35
Link time required	33 [3 4 4 8 3 8 3]
Ground Station Usage	65

Table 9
The generated schedule for a single GS and 5 SCs

Spacecraft	Link start	Link end
1	1915	209
	7806	133
	10540	187
2	205	128
	4255	157
	6127	232
	10192	300
3	637	108
	4821	97
	7159	105
	8692	193
4	1033	256
	2338	196
	3283	256
	5592	125
	6546	274
	8371	227
	9289	143
	11006	97
5	908	62
	2231	73
	3619	178
	5082	154
	10842	137
6	753	152
	2671	118
	3831	286
	5285	187
	6870	254
	7948	153
	9683	124
	11183	231
7	1672	171
	4438	198
	7359	141

Table 10
GA parameter values

GA parameter	Value
Size of population	20
Num. of crossover points (Chromosome A)	10
Num. of crossover points (Chromosome B)	3
Mutation rate (Chromosome A)	10%
Mutation rate (Chromosome B)	40%

5.8. *The multiple ground station case*

We present some computational results obtained with GA for the ground stations scheduling for the case of multiple ground station scenario, namely, 7 spacecraft with 4 ground stations. The Satellite Tool Kit (STK) [17] is used to generate some simulation scenarios. A total of 20 independent runs of the GA were performed (under the same parameter configuration –see Table 10) and average results are reported.

We show in Fig. 22 the graphical representation of different fitness functions and in Fig. 23 for the particular case of selection fitness. The corresponding schedule is graphically shown in Fig. 24. In the figure, the timeslots of squares are the Access Window of the ground station to the spacecrafts while, the solid bars are actually scheduled communication times for each spacecraft.

6. Conclusions

Mission operations scheduling problem arises in management of satellite/space missions. In most real case projects from ESA and NASA, mission planning is still done under human intervention. But, because the problem is very complex, its automation is very important to handle increasing demands for mission planning. We have presented some variants of the problem and have analysed its complexities due the large number of constraints and multi-objective nature. The problem is known to be intractable and therefore meta-heuristics are the *de facto* choice to solve the problem for practical interest. In this paper we evaluate the effectiveness of Genetic Algorithms (GAs) for near-optimally solving the problem. A data simulation model based on STK (Satellite Tool Kit) is used for the experimental study and computational results are presented for the cases of one and multiple ground stations scheduling obtained with Genetic Algorithms using the STK simulation toolkit. Genetic Algorithms showed to efficiently solve the problem as an important resolution technique for the family of satellite scheduling problems.

In our future work we plan to implement other versions of GAs, such as Steady State GA and Struggle GA, and parallel versions of GAs to speed up the GA computations.

References

[1] S. Badaloni, M. Falda and M. Giacomin, Solving temporal over-constrained problems using fuzzy techniques, *Journal of Intelligent and Fuzzy Systems* **18**(3) (2007), 255–265.

[2] L. Barbulescu, A. Howe, J. Watson and L. Whitley, Satellite Range Scheduling: A Comparison of Genetic, Heuristic and Local Search, *Parallel Problem Solving from Nature –PPSN*, VII: 611–620, 2002.

[3] L. Barbulescu, J.-P. Watson, L.D. Whitley and A.E. Scheduling, space-ground communications for the air force satellite control network, *Journal of Scheduling* **7**(1) (2004), 7–34.

[4] L. Barbulescu, A.E. Howe, L.D. Whitley and M. Roberts, Trading places: How to schedule more in a multi-resource oversubscribed scheduling problem. In *Proceedings of the International Conference on Planning and Scheduling*, 2004.

[5] L. Barbulescu, J.P. Watson, L.D. Whitley and A.E. Howe, Scheduling Space-Ground Communications for the Air Force Satellite Control Network, *J of Scheduling* **7**(1) (2004), 7–34.

[6] L. Barbulescu, A.E. Howe and D. Whitley, AFSCN scheduling: How the problem and solution have evolved, *Mathematical and Computer Modelling* **43**(9–10) (2006), 1023–1037.

[7] A. Barolli, E. Spaho, L. Barolli, F. Xhafa and M. Takizawa, QoS routing in ad-hoc networks using GA and multi-objective optimization, *Mobile Information Systems* **7**(3) (2011), 169–188, IOS Press.

[8] S. Damiani, H. Dreihahn, J. Noll, M. Nizette and G.P. Calzolari, A Planning and Scheduling System to Allocate ESA Ground Station Network Services, *The Int'l Conference on Automated Planning and Scheduling*, USA, 2007.

[9] A. Durresi, M. Durresi, L. Barolli and F. Xhafa, MPLS Traffic Engineering for Multimedia on Satellite Networks, *Journal of Mobile Multimedia* **5**(1) (2009), 3–11.

[10] ESA Science and Technology, http://www.esa.int/.
[11] S.A. Harrison, M.E. Price and M.S. Philpott, Task Scheduling for Satellite Based Imagery. In The *Eighteenth Workshop of the UK Planning and Scheduling, Special Interest Group*, University of Salford, UK, 1999, 64–78.
[12] J. Jayaputera and D. Taniar, Data retrieval for location-dependent queries in a multi-cell wireless environment, *Mobile Information Systems* **1**(2) (2005), 91–108.
[13] T. Oda, A. Barolli, F. Xhafa, L. Barolli, M. Ikeda and M. Takizawa, Performance evaluation of WMN-GA for different mutation and crossover rates considering number of covered users parameter, *Mobile Information Systems* **8**(1) (2012), 1–16, IOS Press.
[14] J.C. Pemberton and F. Galiber, A constraint-based approach to satellite scheduling. In *DIMACS workshop on on Constraint programming and large scale discrete optimization*, E.C. Freuder and R.J. Wallace, eds, American Mathematical Society, Boston, MA, USA, 2000, 101–114.
[15] A. Sarkheyli, B.G. Vaghei and A. Bagheri, New tabu search heuristic in scheduling earth observation satellites. In *Proceedings of 2nd International Conference on Software Technology and Engineering* (ICSTE), V2-199–V2-203, 2010.
[16] W.T. Scherer and F. Rotman, Combinatorial optimization techniques for spacecraft scheduling automation, *Annals of Operations Research* **50**(1) (1994), 525–556.
[17] Satellite Tool Kit: http://www.agi.com/products/by-product-type/applications/stk/.
[18] K. Xuan, G. Zhao, D. Taniar and B. Srinivasan, Continuous Range Search Query Processing in Mobile Navigation, *Proceedings of the 14th International Conference on Parallel and Distributed Systems* (ICPADS 2008), IEEE, 2008, pp. 361–368.
[19] K. Xuan, G. Zhao, D. Taniar, M. Safar and B. Srinivasan, Voronoi-based multi-level range search in mobile navigation, *Multimedia Tools Appl* **53**(2) (2011), 459–479.
[20] G. Zhao, K. Xuan, W. Rahayu, D. Taniar, M. Safar, M. Gavrilova and B. Srinivasan, Voronoi-Based Continuous k Nearest Neighbor Search in Mobile Navigation, *IEEE Transactions on Industrial Electronics* **58**(6) (2011), 2247–2257.
[21] N. Zufferey, P. Amstutz and P. Giaccari, Graph Colouring Approaches for a Satellite Range Scheduling Problem, *Journal of Scheduling* **11**(4) (2008), 263–277.
[22] A.B. Waluyo, W. Rahayu, D. Taniar and B. Srinivasan, A Novel Structure and Access Mechanism for Mobile Broadcast Data in Digital Ecosystems, *IEEE Transactions on Industrial Electronics* **58**(6) (2011), 2173–2182.
[23] P. Wang, G. Reinelt, P. Gao and Y. Tan, A model, a heuristic and a decision support system to solve the scheduling problem of an earth observing satellite constellation, *Comput Ind Eng* **61**(2) (2011), 322–335.

Fatos Xhafa holds a PhD in Computer Science from the Department of Languages and Informatics Systems (LSI) of the Technical University of Catalonia (UPC), Barcelona, Spain. He was a Visiting Professor at the Department of Computer Science and Information Systems, Birkbeck, University of London, UK (2009/2010) and a Research Associate at College of Information Science and Technology, Drexel University, Philadelphia, USA (2004/2005). He holds a permanent position of Professor Titular at the Department of LSI, UPC (Spain). His research interests include parallel and distributed algorithms, combinatorial optimization, approximation and meta-heuristics, networking and distributed computing, Grid and P2P computing. He has widely published in peer reviewed international journals, conferences/workshops, book chapters and edited books and proceedings in the field. He is Editor in Chief of the International Journal of Space-based and Situated Computing, and of International Journal of Grid and Utility Computing, Inderscience Publishers. He is an associate/member of Editorial Board of several international peer-reviewed scientific journals. He has also guest co-edited several special issues of international journals. He is actively participating in the organization of several international conferences.

Junzi Sun holds a MSc degree from the Technical University of Catalonia (Barcelona), Spain. He is currently with CTAE-The Aerospace Research and Technology Center, Barcelona, Spain for Information Technology and Aerospace R+D. He is interested in space and computer science. During his time in university, he participated in the Space Study Program (SSP) of the International Space University (ISU) three times as an IT staff member, in Vancouver, Strasbourg and Beijing. And finally he participated as a student in SSP 09 in NASA Ames Research Center. His research field is Information Technology, Netwoking, and Software development.

Admir Barolli holds a PhD from Fukuoka Institute of Technology, Fukuoka, Japan. He has participated in several research projects related to advanced networking systems. He has published in many international conferences and in high quality international journals. From October 2009 to June 2010, he was a Visiting Researcher at Curtin University of Technology, Australia. In 2011, he was a Visiting Researcher at the Department of Computer and Information Science, Seikei University, Japan. His research interests include Optimization, Genetic Algorithms, intelligent systems, wireless mesh networks, ad hoc networks, networking simulation software and P2P systems.

Alexander Biberaj, PhD is an Assistant Professor at the Faculty of Information Technology of the Polytechnic University of Tirana, Tirana, Albania. He has participated in several ICT programs for advances information society development. His research interests include Networking Systems, Resource Management, Management Information Systems, Information Technology and its application to e-Government and e-Society.

Leonard Barolli received BE and PhD degrees from Tirana University and Yamagata University in 1989 and 1997, respectively. From April 1997 to March 1999, he was a JSPS Post Doctor Fellow Researcher at Department of Electrical and Information Engineering, Yamagata University. From April 1999 to March 2002, he worked as a Research Associate at the Department of Public Policy and Social Studies, Yamagata University. From April 2002 to March 2003, he was an Assistant Professor at Department of Computer Science, Saitama Institute of Technology (SIT). From April 2003 to March 2005, he was an Associate Professor and presently is a Full Professor, at Department of Information and Communication Engineering, Fukuoka Institute of Technology (FIT). Dr. Barolli has published more than 250 papers in referred Journals and International Conference proceedings. He has served as a Guest Editor for many International Journals. Dr. Barolli has been a PC Member of many International Conferences and was the PC Chair of IEEE AINA-2004 and IEEE ICPADS-2005. He was General Co-Chair of IEEE AINA-2006 and IEEE AINA-2008, Workshops Chair of iiWAS-2006/MoMM-2006 and iiWAS-2007/MoMM-2007, Workshop Co-Chair of ARES-2007, ARES-2008 and IEEE AINA-2007 and AINA-2009. Presently, he is General Co-Chair of CISIS-2009. Dr. Barolli is the Steering Committee Chair of CISIS International Conference and is serving as Steering Committee Member in many International Conferences. He is organizers of many International Workshops. His research interests include network traffic control, fuzzy control, genetic algorithms, agent-based systems, ad-hoc networks and sensor networks, Web-based applications, distance learning systems and P2P systems. He is a member of SOFT, IPSJ, and IEEE.