

AN HYBRID APPROACH TO REAL COMPLEX SYSTEM OPTIMIZATION

Application to satellite constellation design

Enguerran Grandchamp
University of the French West Indies
enguerran.grandchamp@univ-ag.fr

Abstract This paper presents some considerations about hybrid optimization algorithms useful to optimize real complex system. The given indication could be used by readers to conceive hybrid algorithms. These considerations have been deducted from a concrete application case: the satellites constellations design problem. But each of the advanced techniques proposed in this paper are considered in a more general way to solve other problems.

This problem is used to illustrate the techniques along the paper because it is grouping many characteristics (difficulties) of contemporary real complex systems: the size and the characteristics of the search space engendered by a combinatorial problem; The irregularity of the criterions; The mathematical and physical heterogeneity of parameters forbids the use of classical algorithms; The evaluation of a solution, which uses a time consuming simulation; A need of accurate values. More details are available in previous papers (10, 13, 11, 12).

For these reasons, we could learn a lot from this experiment in order to detach hybrid techniques usable for problems having close characteristics. This paper presents the historic leading to the current algorithm, the modeling of the complex system and the sophisticated algorithm proposed to optimize it.

Application cases and ways to built significant tests of hybrid algorithm are also given.

Keywords: Hybrid optimization, metaheuristics, modeling, simulation, constellation design

1. Introduction

This paper explores a hybrid meta-heuristic approach to solve complex systems optimization problems. These methods are dramatically changing our ability to solve problems of practical significance. Faced with the challenge of solving hard optimization problems that abound in the real world, classical methods often encounter great difficulty even when equipped with a theoretical guarantee of finding an optimal solution. So we use such algorithm as classi-

cal resolution techniques found their limits, typically for large combinatorial exploration space.

This paper is more centered on optimization than on space system design. The main goal is to underline and illustrate the power of hybrid optimization and the way to introduce expert knowledge to obtain a more efficient search.

Nowadays, complex systems are defined by a large number of parameters and operational constraints induce the necessity of accurate values.

These two goals are on opposition and exploring the whole searching space with accuracy is not feasible in most cases: using techniques to explore a lot of areas from the searching space doesn't allow to produce accurate values, using a local optimization to refine the values doesn't allow to drive a global optimization.

In such a case, hybrid optimization gives an elegant solution. We have to intelligently manage the two approaches to refine values in the neighborhood of good local optima and to privilege the exploration in less important areas. The orchestration is the key of success and we have to exploit the best of each technique at the appropriate instant.

The paper is organized as follows.

Section 2 introduces the problem to solve in its general context. It explains why an optimization process is required with technical and economical reasons. We also precise the limits of the studied system, the evaluation process of a solution and the research space.

Section 3 presents a frequently adopted approach by the Constellation Designer Community. We will analyze its limits to find a new way to solve the problem. Section 4 presents the new approach foundations. It details the main ideas used to design the algorithm (physical signification of the parameters, guiding the search, accurate optimization).

Section 5 integrates all these remarks to present the whole algorithm. All stages are detailed in this section: the knowledge database, both high and low level stages. A special attention is given to the orchestration level.

Section 6 presents the tests and applications. We will first validate each stage separately before to make a global validation. Both telecommunication and navigation application fields are considered.

Section 7 deals with operational use of the techniques. It presents the developed software and the way to use it. We also give indications to speed up the algorithm.

Section 8 closes the paper with some conclusions and perspectives.

2. Satellite constellation optimization background

The presentation of the problem in its economical (decreasing the costs) and technical context (increasing the performances) is necessary to understand

the challenge of the optimization process and the place of design phase in the whole spatial system.

Halfway between optimization and astronautic the satellites constellations design problems deals with many goals. To find the required number of satellites and to correctly set their position, such are the technical challenges of this study. To minimize the cost and reduce the time such are the economical challenges the space industries are confronted in a daily manner (37).

In this section, we shortly detail the whole space system and the limits of the studied system. Then we detail its parameters to show the complexity of the problem: numerous parameters with different physical and mathematical nature.

Space systems: towards a sequential optimization

Satellite constellations are currently used in many fields for civil and military applications: earth observation, telecommunication, navigation. The implementation of a constellation is a multi step process: mission analysis, design of the constellation, deployment of the satellites, satellite checking, maintenance, replacement and termination. Each of these steps must be optimized to decrease the global cost and increase the global performance.

There are two approaches to conceive such a system: a global approach which integrates every step in a multi-objective optimization; a local approach which optimize each step separately.

The first way to solve the problem is currently not practicable for many reasons: firstly the nature of the problems to solve in each step is too different to regroup them in a general algorithm. As an example the deployment step looks like a Constraint Satisfaction Problem (CSP), where some constraints are satellite launcher availability, payload, cost and accessible altitude. The evaluation function integrates cost and time parameters. The design step is a combinatorial problem. The aim of this step is to minimize the number of satellite (in order to reduce the cost) and to maximize the performances. Secondly the number of parameters to set is too large with integer and real variables. For these reasons, the second approach is preferred (36) and each step is optimized in a different manner. In fact, a step optimization is not totally uncorrelated from the other one.

As if we optimize only one step, some considerations about the previous or next step could influence the current optimization process (as constraints or guide).

Limits of the system

The system we want to optimize is limited to the constellation. We have to set the parameters for each satellite that compose the constellation and also to

determine the number of satellites to use.

A satellite is defined by six orbital parameters ($a, e, i, \omega, \Omega, M$) (precise definition is given in 26), having physical signification and unit.

- a : semi axis of the orbits ellipse (from 400 km to 36 000 km)
- e : eccentricity of the orbits ellipse (0 to 1)
- i : inclination of the orbit plane (0 to 360 degrees)
- ω : argument of the perigee (0 to 360 degrees)
- Ω : longitude of the ascending node (0 to 360 degrees)
- M : mean anomaly (0 to 360 degrees)

A constellation is defined by N satellites.

Without additional constraints there is no relation between satellites and the system is defined by $6N$ independent parameters.

Moreover, the number of satellites (N) is also unknown and the problem has a variable number of parameters.

Solution evaluation

The efficiency of a constellation should be guaranty during its revolution period. The duration of the period depends on the orbital parameters of each satellite and the evaluation is based on an accurate simulation process.

Usually this simulation is time consuming and depends on many operational parameters. In fact, ground sampling could change from 10 km to 100 meters and time sampling from 5 minutes to few seconds. The performances could be awaited for the whole earth surface or just over certain areas (north hemisphere, Europe,

For each time-space sample we evaluate a local performance value. This value could be the number of satellites visible from the sample (telecommunication application), the positioning precision induced by the local configuration of the visible satellites (navigation application) or any other technical value.

From this amount of local evaluations, we compute few values (mean, maximum, minimum) to estimate the constellation performance.

As an example, we used the precision function to optimize a navigation constellation (such as GPS or GNSS) with the simulation parameters set as follows

- The time step is set to 1 minute because it is the maximum acceptable gap between two position values for operational conditions.
- The ground sample step is set to 10 km around the whole earth (satellite localization must be available everywhere). This value has been chosen

because it is sufficient for an accurate computation and acceptable for time consumption. But with such value a second validation is necessary with a ground sample step at least of 500 m after constellation optimization.

- The local criterion is a triangulation compute in order to evaluate the local precision. From satellites positions an iterative algorithm is run to minimize the positioning error (least square method).
- At the end of the simulation process, minimum precision is returned in order to evaluate if the constellation is acceptable, mean precision could also be returned.

The time to compute is about 30 seconds on a Pentium IV 2Ghz. We could already note that there is a lack of information returned by an evaluation compared to the time spends during the simulation.

Research space

The research space is generated by the $6N$ parameters. N is an integer and the $6N$ parameters are real variables.

Theoretically the searching space is infinite. In some optimization contexts we conserve compact intervals. This is the case when defining a parameter as a combination of others. For example, if we want an orbit with a one day period, there is a relation between a and i (26).

In most application cases, a sampling of each interval is made. The problem is converted to a combinatorial problem with a very large searching space. In fact, the sampling step could be of 100 meters for a , 0.01 for eccentricity and 0.1 degrees for each other range.

In such a context the size of the space has about 6.10^{21} elements ($356000 * 100 * 3600^4$) for each satellite.

For a constellation with N satellites the searching space has $(6.10^{21})^N$ elements.

For a constellation with an unknown number of satellites included between 1 and N the searching space has about $(6.10^{21})^{\frac{N(N+1)}{2}} ((6.10^{21})^1 + (6.10^{21})^2 + \dots + (6.10^{21})^N)$. N could be equal to 100.

With such a searching space, an exhaustive search is impossible.

3. The first approach: an instructive failure

Principe

In front of a complex problem with many parameters, a large searching space, an expansive evaluation and without rules to predict the impact of a modification to the value of the solution, the first idea is to use an algorithm

which is reputed to treat any kind of optimization problem with a statistical guaranty to reach the best solution and without any consideration about the nature of the problem.

As if there is no universal algorithm there is some algorithm quiet easy to parameter for any kind of problem. Genetic algorithms are part of them. The main advantage is that the research process is a standalone process using basic operators like selection, crossover or mutation.

With such an algorithm, the resolution is theoretically easy. But when considering the wide search space, the population has to be very large (several thousand) in order to leads to a significant global optimization.

When adding the time consuming criterion, this way looks unfeasible and the first idea is to reduce the complexity of the search space.

Reducing complexity: exploration of regular or symmetric constellations

Usually, engineers reduce the complexity of the search space by linking parameters. We artificially reduce the number of parameters to optimize. For example, fixing a common altitude, inclination and eccentricity reduce the parameters from $6N$ to $3(N+1)$.

Such approach reduces the kind of solution the algorithm could reach. In this way, only symmetrical solutions are explored (same kind of orbit, ...).

This is the case of Walker constellations (57) which are composed of satellites having the same altitude and inclination with a uniform distribution of the orbits around the earth and a uniform distribution of the satellites around the orbits.

With such constraints, the optimization algorithms are skewed and willingly designed for symmetric resolution.

An incomplete panel of solution

Such symmetric solutions have intrinsic properties. We could easily design a constellation that satisfied the required local performances but without any guaranty about the number of satellites (which is directly linked to the cost of the constellation).

Moreover some recent constellations (28,30) are composed of different kind of orbits (tundra, circular, heliosynchronous) and are not accessible from a symmetric point of view.

Designing a symmetric constellation as a starting point for the optimization could appear to be a good idea but the local optimum created by this constellation is so attractive that we must totally break the symmetry to access other kind of solution.

More details on classical approaches are available in the literature (31, 33, 35).

4. The new approach foundations

Born from the limits and drawbacks of the previous method and several preliminary studies, this approach tries to answer to all expectation in a more constructive way than the previous one.

Based on a better use of the simulation and on a simplification of the criterions, the algorithm is composed of several levels and uses different optimization techniques: it integrates a knowledge database on the orbits and a numerical optimization process both orchestrated by a metaheuristic algorithm.

The main idea is to bypass the drawbacks with a decomposition of the system in order to adapt the algorithm to the nature of the sub-problem.

When dealing with a wide, dark and compact search space, we have to find a way to introduce an "exploration map", a "spot light" and a "magnifying glass". Following subsections describe the way to materialize this representation.

Physical signification of the parameters: splitting the search space (the spot light)

For each complex system we advise to make a particular effort to precisely understand the influence of each parameter to the system behavior.

If we consider the physical signification of the parameters we could classify them according to their importance or influence to the solution.

A small variation on a parameter could dramatically change the solution evaluation or on contrary a coarsely variation could have a least effect.

In our application, the physical differences of the parameters previously underlined as being a drawback for a general optimization process without precise consideration will become an advantage in the new modeling.

In fact, for some parameters (a, i, Ω) we could easily analyze their impact on a single orbit. The information given is not extensible to predict the interaction with other orbits because the main problem becomes the synchronization of the satellites.

For example, the parameter a has a direct influence on the area visible from the satellite and the parameter i is linked to the accessible latitude.

So we could divide the parameters ranges into logical subset.

These considerations are at the origin of the expert knowledge introduced in the next section.

The advantage of such an approach is to reduce the complexity (combinatorial) without reducing the panel of solution.

So the better way we found is to organize the search in order to split the space into logical areas. If these areas are intelligently conceived, we could evaluate the ability of each one to resolve the problem and so dramatically reduce the exploration.

This will become an essential actor of the algorithm success and the way to bring light to the search.

An efficient progression: guiding the search to avoid blind optimization (the exploration map)

As if we intelligently organize the search space, there are a lot of areas to explore. We have to manage this exploration in order to guide or constraints it. We use information given by the previous study (parameters) to select the areas, information given by users to constraint the search in a selected way, information returned by previous search. The goal of this part of the algorithm is twice: driving the search among the different areas and coordinates the ratio between local and global search.

Advanced metaheuristics techniques will be used to ensure the efficiency of the progression.

An accurate optimization: local intensification of the search (the magnifying glass)

After the localization of the interesting area, we have to precisely set the value of each parameter. This part of the optimization process deals with fine local optimization.

Specifications are not the same for the algorithm to employ. It must be able to reach a local optimum very quickly. The techniques used are close to classical optimization.

A special reflection must be done in order to integrate the possibility of multiple local optima in the same area (in the sense of previous subsections).

In fact, additionally to previous drawbacks, the criterion used to evaluate a solution usually presents some unfriendly characteristics: many local optima, chaotic surface.

Combining with previous algorithm, we hope being able to explore a wide space and to precisely analyze only interesting one.

Learning from the simulation: converting evaluation time to optimization time.

As previously underlined, the evaluation criterion is time consuming and produces few values. Moreover the simulation doesn't return pertinent information about the good or bad properties of the satellites. These values don't allow differentiating satellites within the constellation in order to modify the orbit of less efficient one.

The idea is to use the simulation process to compute many other values for each satellite in order to evaluate their contribution to the efficiency of the con-

stellation.

This computation induces a fee to the evaluation function but the pertinence of the gain justifies the effort. Moreover the overload is not proportional to the number of values because some of them are computed for each time step instead of for each time-space sample.

For example, a value (per satellite) could be the duration of visibility from the interesting area (telecommunication or earth observation system) or the number of time a satellite is implied in a high precision positioning triangulation (navigation system).

These values will be used during the main algorithm presented later in the paper in order to guide the progression.

5. Modeling: Towards a multi stage optimization

When considering all previous remarks and advises a multistage hybrid algorithm comes naturally. In fact, there is a local and a global approach, an accurate optimization and a wide exploration need. Problems to solve are totally different and must be parsed in order to be treated.

Other hybrid approach are presented in the literature (38). We define three stages for the algorithm: a numerical stage dealing with accurate optimization in a local context; a heuristic stage introducing expert knowledge to organize the search space; a metaheuristic stage managing the progression of the search. Each stage is detailed in the next subsections.

The knowledge interconnection stage: the orbit database

The Orbit Data Base (ODB) is the tool that allows restricting and organizing the search space. It is the way to introduce expert knowledge by the definition of Orbit Classes. According to the characterization of an orbit (Section 0), each class defines a subset of values for each of the six orbital parameters. In fact, a class regroups orbits with common characteristics (altitude, periodicity, inclination, eccentricity, periodicity ...). That is to say, a class is combination of ranges (13, 19) for some orbital parameters, fixed values or relations for others. As an example, if we want to represent orbits which have a revolution period of one day (useful for a daily earth observation) we define a relation between parameters a , e and i . In such a case, the class is not composed of successive orbits.

Now we directly manage classes instead of satellites (a satellite is a sample of a class). This naturally organizes and split the search space into two hierarchical subspaces: subspace of classes and subspace of satellites within a class. The subspace covered by the set of classes is not necessarily the whole search space (search space engendered by the accessible orbit defined by the six orbital parameters). Moreover, classes are not necessarily disjointed.

In fact, in standard classification (16), each sample has to be assigned to exactly one class. The ODB relaxes this requirement by allowing gradual memberships, offering the opportunity to deal with satellites that belong to more than one class at the same time.

This fundamental characteristic of the ODB is extended with fuzzy concepts (17, 18, 19) allowing a flexible frontier between classes.

In our application case, the introduction of fuzzy classes is necessary for many reasons.

First of all, many orbital classes haven't a well defined border. For example, we frequently talk about Low Earth Orbit (LEO) to represent orbits starting with an altitude of 400 km. We also talk about Medium Earth Orbit (MEO) to represent orbits near 12 000 km altitude. But what about an orbit with 6 000 km altitude? Is it a LEO or a MEO one? The transition between LEO and MEO is not so easy to express and the use of fuzzy borders allows a smooth transition between classes.

Secondly, this property appears very useful during the progression of the search because the definition of this intersection between classes allows a natural transition between them.

In fact, when changing a solution, it frequently appends to reach the border of a class. In such a case, we exchange the current class with one that contains the solution (intersection not empty).

Figure 1 illustrates these characteristics. The number and the nature of the

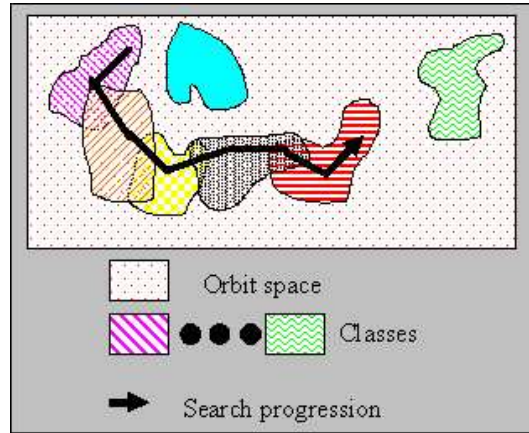


Figure 1. Search space decomposition

classes are determinant for the efficiency of the search and the characteristics of the returned solution.

The parameters that constraint the different stages (number of iterations, thresh-

olds, ...) will also depend on the classes definition.

In fact, as an example, if the ODB is composed with a lot of small classes, an effort will be made for the high level stage to choose the representatives classes and their proportion. On contrary, with less but larger classes, the low level stage will be used to refine the parameters of each satellite.

Now a satellite becomes a handler of a class (the satellite parameters take their value in the fuzzy range defined by the belonging class).

As a constellation is a combination of satellites, a configuration is defined as a combination of ODB class handlers. The fundamental difference between a constellation and a configuration is that one configuration contains a set of constellation (each constellation whose satellites verify the corresponding ODB class). We could compare the configuration space to the constellation space in the same way as in section 2.

Examples of orbits are given in Figures 2-a, 2-b, 2-c, 2-d. in order to illustrate the panel of accessible classes. When designing the ODB, we have to define a

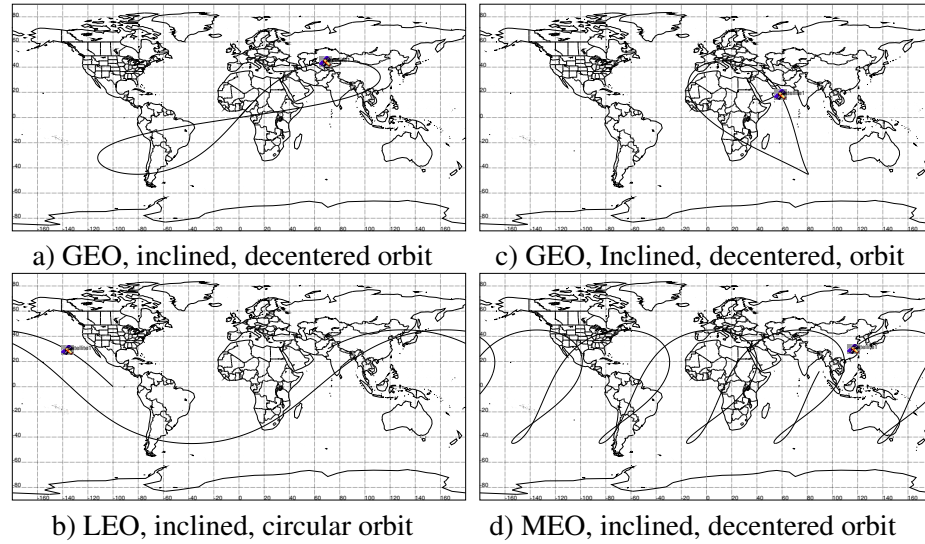


Figure 2. Orbit sample

complete package to manipulate the classes. In a general way, it includes the definition of the elements (accessible orbits), an order to browse them and an operator to compare them. We also have to define a notion of distance between classes, orbits and configurations.

For more details readers are invited to refer to previous works (13).

We note Cl_k a fuzzy orbit class, m the number of classes, $Cl_{k=1..m}$ the ODB and Cfk a configuration.

High level stage: the case of Tabu Search

Tabu foundation This level tries to answer the question "which configuration to explore and how to explore it?". The goal of this level is to manage the search.

The Tabu Search method (3,4,5) has been retained because it is a local search method avoiding heaviness of managing a population of solution and because it integrates the notion of history through memory structures.

This algorithm is very complete and enables learning from previous exploration to predict future progression.

The neighborhood used for the high level stage is composed of configurations. At an iteration of the algorithm, we have to choose the successor of the current configuration between the configurations that composed its neighborhood.

As it is a local search, we define the neighborhood as follow. A neighbor of the configuration Cf_i is any Cf_j configuration obtained by

- Adding a new handler of a class to the current configuration. There is m (number of classes in the ODB) derived moves.
- Removing a handler of a class from the current configuration. There is N (number of satellites in the current constellation) derived moves.
- Replacing the belonging class of a handler in the current configuration. The number of derived moves depends on the composition of the configuration (number of represented classes) and the composition of the ODB (total number of classes).

The authorized moves looks very simple but it ensures a continuity of the search.

The tabu state is set according to the analyze of the memory structures. The selection of the successor is done according to a balanced random choice based on a coarsely evaluation of each non tabu neighbor configuration.

We will explain more precisely the evaluation and selection process in the rest of the section. *Memory management*

The inspiration of Tabu Search to design the high level stage leads us to introduce different kinds of memory structures.

The memory aspect of Tabu Search is essential to drive the search on good conditions.

Short term and long term memory are present in Tabu theory and accompanied by their own special strategies.

The effect of both types of memory may be viewed as modifying the neighborhood of the current solution.

Each memory structure is a kind of history filter which memorize interesting characteristics. We implemented all these structures to have a complete algorithm.

Tabu history analyzing is based on both short term memory (recency-based memory) and long term memory (Quality, Influence, Frequency).

Short term memory forbid to visit recently explored solutions. Long term memory allows to analyze the good or bad effect of certain choice.

The Quality aspect of the memory is used to qualify the contribution of transition to the improvement of a solution.

In our application case, if the addition of a Clk handler often improve the evaluation of the solution (increase performances and/or decrease the costs), the corresponding class will be notify with a high quality value.

The quality value could be affected to each elements of the system: a class, a configuration, a satellite, a transition.

For a configuration there are two steps in the attribution of a quality value. First choosing a configuration among neighbor means evaluating them before any exploration (if we don't want to make a random choice). Secondly, after the exploration of the configuration with the low level algorithm we could precisely evaluate it.

The post evaluation of the configuration is a kind of mean constellation value. The evaluation of a class is a balanced mean of the configuration values where the class is represented.

The pre evaluation of a configuration is based on the evaluation of the classes represented in the configuration.

Then a satellite evaluation is directly linked to the simulation process presented in section 2. We compute separated values for each satellite in order to learn from the simulation to manage the memory.

More details about the expression of the different values are given in (13).

The frequency memory aspect gives information about the number of time we use a move (or a derived move). This indication is used for diversification or intensification process (developed in the continuation).

We could already note that the exploration history take a predominant part in the quality memory management.

The consequence is that the search is starting with a masked neighborhood (random exploration) and values become more and more precise as the search is going on.

High level decisions (exploring a configuration or another) are more and more pertinent and the search is expected to be more and more efficient.

To avoid the blind starting search, we advise to adopt a kind of learning phase in the neighborhood of the starting solution (configuration). This learning phase consist of exploring as much configuration as possible in order to give a first idea of the quality evaluation.

Low level stage: the case of Steepest Descent

The low level stage has been designed to quickly reach a local optimum in a restricted area. This area is limited by the current configuration (authorized values for a satellite within its belonging class).

The goal is not to explore every constellation within the current configuration but to rapidly reach a good solution in order to evaluate the potential of the configuration to solve the problem.

At the origin, this algorithm was a simple Steepest Descent. This algorithm has been chosen as if it could encounter slow convergence because of simplicity of implementation. But the criterion used for our applications is irregular and present many local optima.

So we propose an evolved Steepest Descent algorithm with a restarting strategy in order to bypass the local maxima.

The restarting strategy is a random modification of some orbital parameters. Parameters are selected according to their progression during the steepest descent phase. We select the parameters having changed the least during this phase.

We should note that this restarting strategy is a kind of diversification phase from the low level point of view but stays a local intensification phase from the previous high level stage.

This low level is not the main part of the algorithm because it has a precise and restricted role but its operational implementation is not necessarily easy.

Memory is also present at low level optimization. In fact, during the search it is possible to come back to a configuration already visited. In such a case, we have to restart the search from the last visited solution or from the best encountered solution.

Precisely tuning low and high level optimization parameters: the role of Strategic Oscillations for intensification and diversification ratio.

When analyzing the whole algorithm, we could note the multitude of parameters. The low level stage includes a threshold for the detection of a local optimum, a number of iterations, a progression step The high level stage includes a threshold on class values for the tabu state, a neighborhood size, a range for the number of usable satellites ...

Fixing a value for each parameter is not so easy for many reasons. Firstly, the thresholds and ranges of each parameter strongly depend on the use case. We calibrate the algorithm for each space system application (navigation, earth observation and telecommunication).

Secondly, the optimization needs change during the search. At the beginning, we have to explore several areas in order to locate the most interesting one be-

fore to deepen more accurately few configurations at the end of the search.

Moreover, the parameters depend on the search progression itself. In case of a satisfying search, an intensification phase is preferred in order to exploit the local area. On contrary, if the search is not concluding, a diversification phase starts to reach other areas.

This part of the algorithm is the key of the success. It coordinates the different techniques and algorithm to apply the most adapted one.

We use Strategic Oscillation techniques presented in the Tabu theory (5) to realize such a coordination.

The two phases are sequenced as follow.

The intensification phase (I)

Number of satellites. At the beginning of the phase, the number of satellites included in the constellation belongs to a wide range; As the search progress, the number of required satellites becomes more and more precise and the tolerated number of satellite is centered on a mean value.

Number of used classes. At the beginning of the phase, the number of usable classes is composed of the whole ODB. According to their value (computed from the simulation and exploration process) more and more classes are rejected (tabu state) as the intensification progresses.

The first consequence of this two evolution is a reduction of the neighborhood which facilitates the progression of the search at the high level stage.

Low level optimization At the beginning of the phase, the low level stage is run with a few number of iteration and with low threshold in order to exit the optimization process very quickly. This parameters are regularly increased to make an more accurate low level optimization. With such evolution of the different parameters, we progressively transfer the effort (computes and time) from high level to low level.

The Diversification phase (D)

After a long period of intensification, the search doesn't evolve anymore: the value of the best solution doesn't increase significantly; Visited configuration doesn't change. At this time, a diversification process is engage in order to explore other configuration types. The choice is making according to memory structures (frequency, recency) filled during the intensification phase. In fact, the less used classes are integrated to create a new configuration which will serve as a new departure for the high level stage.

Figure 3 summarizes all the developed notions.

6. Tests and applications

Principe

The validation of a hybrid algorithm is a challenging step. We have to analyze its behavior from different point of view: time computation, suitability

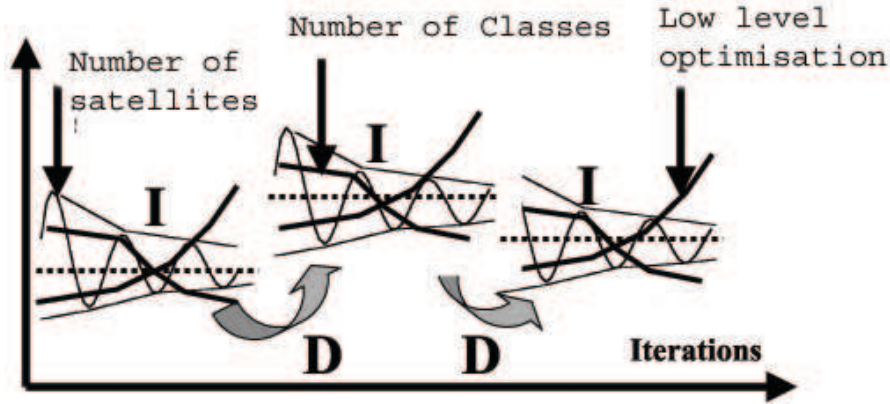


Figure 3. Strategic Oscillations

of returned solutions, pertinence and efficiency of each stage. To precisely analyze the behavior of the algorithm, we have to validate each stage separately before to make a global validation. In this paper, we are overall interested on optimization performances, more considerations about time are given at the end of the paper and in other papers (11,12,10,13).

Another relevant consideration (and difficulty) is that for most of the applications we don't know the best solution or simply a good solution. This remark could be applicable for many other complex systems.

To evaluate the low level optimization, we have to abstract the high level stage. So we set the configuration before to run the optimization process.

To evaluate the ability of this level to reach a good and/or the best solution we choose different kind of configurations: some configurations which contain a solution to the problem and some configurations which don't contain a solution.

The test of the high level stage must confirm the efficiency of the configuration navigation algorithm. We have to abstract the low level optimization. Then the corresponding algorithm must return the best solution within a configuration in order to inhibit the influence of the low level.

The corresponding algorithm (low and high level) is reduced to a combinatorial optimization process. After the separate validation, we have to integrate each stage to make a global validation of the algorithm. The main objective of this part is to finely tune the strategic oscillations to manage the low and high level trade-off during the search. In fact, time and iteration ratio change with the progression of the search.

The tests are also a way to analyze the behavior of the algorithm. So we should run scenarios we can check during their entire development.

Numerical stage validation

In our case, the validation has to be done for different application fields and we choose to apply the algorithm to telecommunication and navigation space system.

We present here a scenario for a continuous coverage of the $[-70, +70]$ latitude band. We know an acceptable solution with 3 Geostationary (GEO) or 3 Medium Earth Orbit (MEO) satellites. So we both test the algorithm with 3 and 4 satellites in order to compare the results. There is no restarting strategy for this first test. Four initial solutions are injected to the algorithm and chosen as follow

- P0 : 3 Low Earth Orbit (LEO) satellites (4 LEO satellites)
- P1 : 3 Medium Earth Orbit (MEO) satellites (4 MEO satellites)
- P2 : 2 LEO + 1 MEO satellites (2 LEO + 2 MEO satellites)
- P3 : 1 LEO + 1 MEO + 1 Geostationary (GEO) (2 LEO + 1 MEO + 1 GEO)

Table 1 present the results for the scenario 1. The stores values are the computation time and the evaluation of the best constellation reached during the search. Vmax indicates the best attended value. The test is made for three sets of relaxed parameters.

Results are satisfying because the best found solution for P1 condition with 3 satellites is acceptable for both relaxed parameters. In fact, the value of the best solution is about 95 percent of the best attended value. We could increase this value with a fine synchronization of the satellites but this is not the goal of the algorithm, we only want to know the configuration and global positioning of the satellites.

With 4 satellites, results are better but not concluding (the maximum is not reached). After a fine analyze, one of the satellites is redundant most of the time during the simulation. The three other satellites are correctly positioned but the forth one doesn't succeed to find a synchronized place.

The first statement concerning this low level algorithm is that during the first iterations the algorithm is able to find a value for the most influence parameters.

In fact, among the six parameters of a satellite, the altitude (a) and the inclination (i) are very determining for the quality of the solution.

This result is very promising because it allows considering a separated optimization which could dramatically reduce the time computation.

A second test has been realized with a restarting strategy, ?Table 2 stores the number of iterations to reach the optimum and the value of the optimum. Results are better than for previous test because the maximum is reached with four satellites and reach 98.5 percents with three satellites.

The restarting strategy seems to efficiently bypass local optimum but the number of iterations is increasing consequently. For operational use, we have to take care of this to prevent exponential time consuming.

Metaheuristiques stage validation

To validate this stage, we create two kinds of ODB to ensure the return of a good constellation after each low level stage. Firstly, we voluntarily choose an aberrant ODB composed of classes containing only one solution. The low level optimization is instantaneous.

Secondly, we built an ODB allowing a configuration containing a good solution. Corresponding classes allow several orbits and we have to precisely know the parameters values for the best solution in order to return it instantaneously. We present now an application for navigation space systems. The goal is to recover the GPS constellation.

The main characteristics of this constellation are the doublet of satellites in each orbital plan and the symmetric distribution of the plane around the earth. The starting configuration is set to a Walker constellation with 18 satellites, in order to keep the symmetry.

The first result concerns the performances of the constellation and its design. Figure 4. presents the constellation design. We remark that a doublet is formed for each orbit plan. This result confirms the efficiency of such configuration for triangulation precision. The main difference between the returned constellation and the GPS constellation is the position of the doublet which is not symmetric. Global performances are quite good because the criterion used to optimize is the availability of the constellation (that is to say the percentage of time the constellation is given a sufficient precision). The value of the returned constellation is about 99.89 percents and the value of the GPS constellation is about 99.999 percents. This values seems very close but the attended percentage should be very close to 100% for operational use.

The second result concerns the configuration evaluation. In fact, we define two ways to evaluate a constellation (a pre evaluation and a post evaluation). Figure 5. presents the evolution of this two values for a selected configuration during the high level search progression. Readers have to remind that the configuration values are linked to the class value which evolves at each new configuration exploration. The expression of these values are given in 13. We remark that the distance between the graphic stay small. The pre evaluation

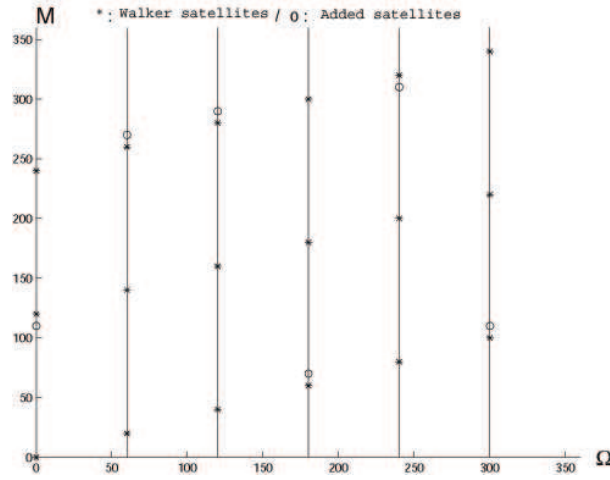


Figure 4. Constellation design

seems to be significant for a coarse estimation of the ability of a configuration to solve the problem.

Another interesting information concerning this test is that the high level

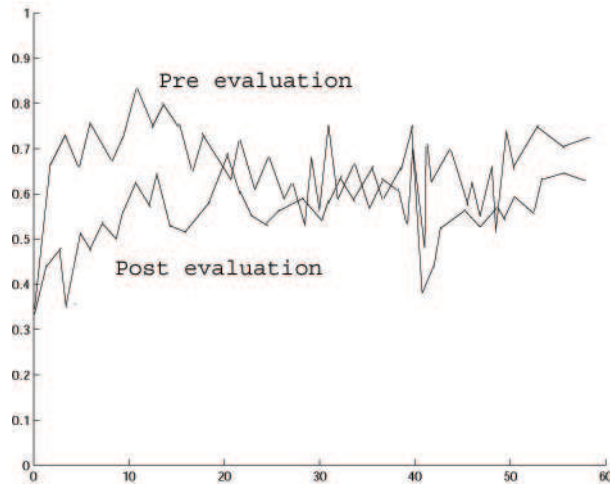


Figure 5. Pre and post configuration

algorithm only explores between 0.5 and 0.7 percents of the configurations. This allows considering a wide complex system optimization but should be decreased to treat the whole search space ($6N$ parameters free).

Global validation

After the independent tests, we are going to evaluate the ability of the whole algorithm to solve the problem.

The application field is the navigation. We try to recover the GPS constellation starting from an 18 satellites Walker constellation (same starting solution as previous test).

The main difference concerns the ODB which is composed of classes with common altitude and inclination but with free other parameters. Moreover, the total number of satellites is not limited.

The total number of satellites is 30, which is more than GPS constellation.

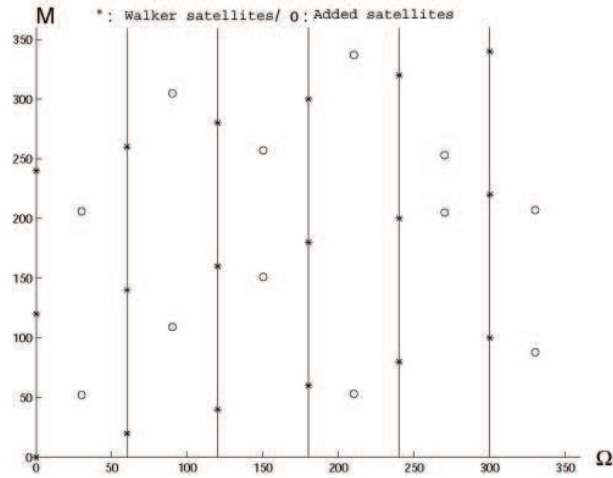


Figure 6. Global optimization results

The repartition is quiet regular but on other orbits than Walker satellites.

The criterion used has an evaluation of 99.999 for GPS constellation and 99.956 for the returned constellation.

This value is inferior but quiet good.

This seems to indicate that there is another good local optimum with a different satellite repartition.

We should also note that the GPS constellation is not necessary the best constellation for the proposed criterion.

On conclusion, the algorithm has been tested for both telecommunication and navigation systems. It presents satisfying characteristics in a directed way (precise orbit classes).

7. Operational use

This tool has to be used by engineers to help those designing constellations. The goal is not to deliver a standalone software but to give the possibility to model a constellation with certain constraints.

Using the tool as an expert system

According to the validation made in the previous section, the tool could be very efficient to find a sub-optimal solution. The efficiency is dependant on the ODB definition which is the way to drive and limit the search. Experts could use the ODB to introduce heterogeneous constraints such as altitude range, number of satellites or orbit inclination.

Experts could rapidly have a solution with a restricting ODB and so explore different kind of constellation. This first exploration allows establishing a global design of the constellation (minimum number of satellite, coarse range of altitude).

After this, an accurate search could be launched to determine precise parameters.

This way seems to be the best way to use the design tool.

Software development

The algorithm has been implemented in a high level computer science environment. A complete modeling of the system has been made with the UML language in order to produce a clean conception.

An object oriented language has been used to implement the algorithm and to develop a user friendly interface allowing to parameter the ODB and the whole algorithm (number of iterations, thresholds ...), to run the resolution and to visualize the results.

Figure 7 and Figure 8 present some snapshot of the main part of the interfaces. The main Window allows to parameter the Strategic Oscillation, the Tabu Search and the Steepest Descent. It displays the memory state, the best solution value, the number of ran iterations, the time spends. It also gives a link to visualize each satellite within the constellation by three ways: parameters values, 2D (Figure 9-a) and 3D (Figure 9-b) shape of the orbit. In a complex system, having different ways to visualize results (numerical, 2D, 3D, criterion) is important in order to analyze them.

How to speed up the search: towards high performance computing

Even if constellation satellite design is not directly dependant on tools performances or real time computing (this step is an upstream work not directly

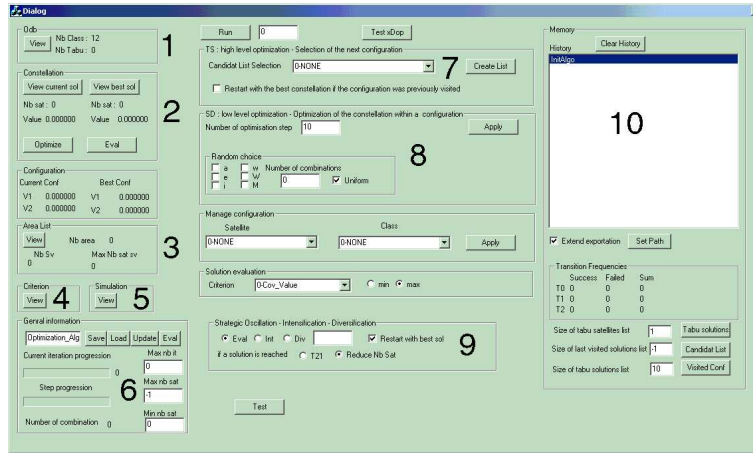
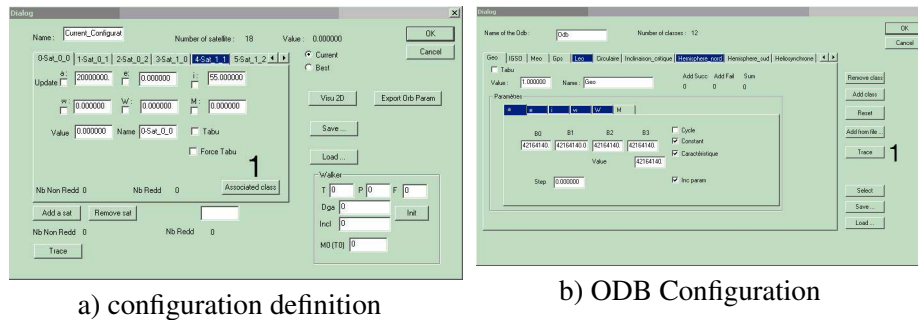


Figure 7. Main windows: algorithm



a) configuration definition

b) ODB Configuration

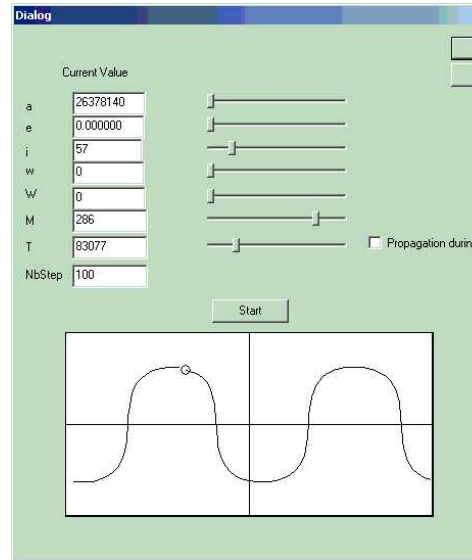
Figure 8. Sub windows

linked to a production context), we have to consider an operational use case. Currently the tool is suitable for a quick solution outline or for a time consuming accurate solution.

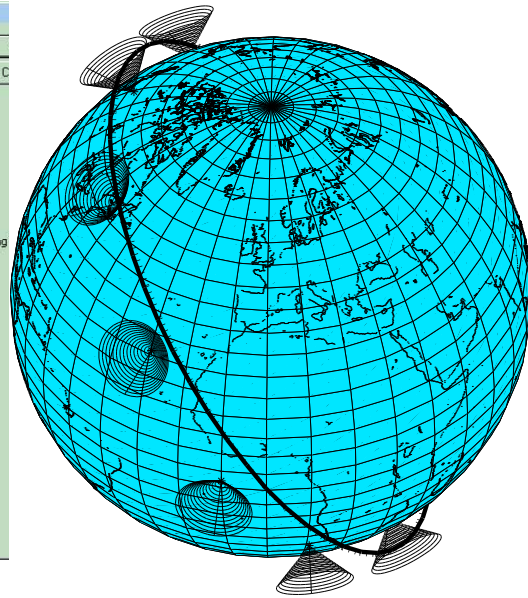
As previously expressed, the main reasons are the time consuming evaluation criterion (sometimes several minutes for a single solution evaluation) due to a simulation process and the combinatorial exploration space engender by the parameters.

Regarding the system, parallelization seems to be the main way to accelerate this time consuming algorithm. In fact, we could consider the evaluation of each configuration of the neighborhood on several processors.

There aren't many parameters to transfer for each configuration to treat compared to the number of iterations to explore it. In fact, if we consider that the



a) 2D orbit visualisation



b) 3D Crirerion visualization

Figure 9. Visualization

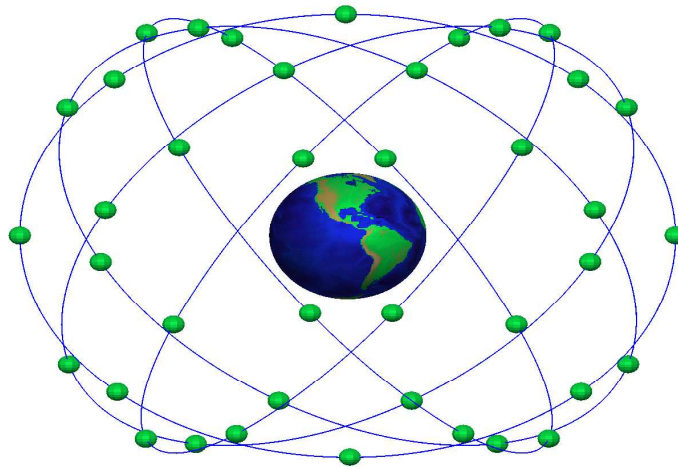


Figure 10. 3D constellation visualization

ODB and the criterion are loaded on each processor, only N parameters are necessary to run a configuration exploration.

If we consider the Strategic Oscillation, parameters concerning the low level algorithm are also necessary to run the exploration (number of iterations, thresholds ...).

All those parameters cumulate, few octets are necessary and a parallel calculation is possible over a computer cluster with a distributed memory. It is not required to invest on a multiprocessor super calculator.

Other considerations, such as pre-calculation, allow considerably reducing time. In fact, when the criterion is evaluated, a simulation is run over a long time period and over a wide surface. When changing the parameters of one satellite, the whole computation is done again. So for a N satellites constellation, $\frac{N-1}{N}$ percent of the simulation is done twice. So during the low level stage which is a local search, we spend a lot of time computing the same values.

With this last consideration, we could divide the computation time with a factor between 10 and 20 (for a classical constellation optimization).

With a parallel approach, the gain could be more important.

8. Conclusion and perspectives

Algorithm performances

At the beginning of the development, we expect a more autonomous tool to design constellations. But operational considerations and accurate performances constraint us to propose a more interactive tool for expert designers.

For localized applications (navigation, telecommunications) both low and high level algorithm present some convincing properties when run separately. The validation of the integrated algorithm is more problematic because of main behavior understanding but engage tests give interesting results.

With this integrated multi stage tool, experts have the possibility to approach a complex problem by different ways.

On conclusion, the policy we adopt to conceive the algorithm was to conserve the possibility to reach any kind of solution. There is no restriction due to the employed algorithm. Nevertheless, we could precisely guide the search in order to reach a preferred kind of solution by introducing expert knowledge using the Orbit Data Bas.

The algorithm has a dynamic full duplex interaction between stages (through memory structures) allowed by the strategic oscillations.

The techniques developed in this paper could be used to any complex system where reducing the searching space is required to reach an interesting solution. But without a good understanding of the dependence between parameters and search space, the progression will be blind and not efficiency.

Extension to other real complex systems

The underlined advantages of this multi-stage algorithm are not specifically adapted to the constellation design problem.

In fact, the algorithm is customizable by many ways.

We do not have the presumption to propose a universal complex system optimization algorithm. Not at all. In this section, we only give indications to select and model problems having close characteristics.

The interaction between the algorithm and the application field (more precisely the problem to solve) is limited to the definition of ODB and evaluation function.

The ODB has been specifically created for the constellation design problem. As express in previous sections, this is an efficient way to drive the search and a specific effort has been made to design this knowledge database.

Such an effort must be done to adapt the ODB to the new challenging problem.

We have to keep in mind, that the adaptation must conserve the characteristics which make the power of the ODB. That is to say, a way to intelligently limit the range of each parameter. In other words, each restriction must be justified with a physical correspondence. An elementary decomposition of the parameters range into uniform interval is not interesting and suitable for the algorithm. In fact, if changing the belonging class means changing the nature of the solution, the metaheuristic level of the algorithm could be able to extract significant structures from the explored solutions.

As example, the distribution of nodes in a network is a quiet similar problem. Likewise, the antenna positioning in cellular networks is close to current problem (24,25).

We could characterize an antenna (a node) with following parameters: cost, range (number of interface), bandwidth ...

Each kind of antenna (node) could be translated in terms of classes having physical signification.

The goal is to find the number of antenna (node), their characteristics (belonging class) and their position.

Each parameters of a class could be fixed or variable: fixed cost, or cost depends on bandwidth ...

Some of the parameters could be interpreted as constraints on localization: maximum altitude for an antenna, environment (high voltage wire proximity ...).

Acknowledgments

This work has been financed by Alcatel Space Industries (ASpI), located in Toulouse France. This industry is a world famous Space Industry. Contracts from many countries are handled each year by the DOS/AS department

directed by Eric Lansard. This work was also due to a collaboration with the LIMA Laboratory of the ENSEIHT, an engineer Computer Sciences School located in Toulouse, and particularly with Vincent Charvillat.

NSD = 1000			PSD	
Time/Val		a,i	a,i, e, w	a,i,e,w,W,M
Scenario 1 3 Satellites Vmax=2016	P0	16-15/280	32-18/315	47-55/362
	P1	16-38/1816	31-58/1756	48-17/1793
	P2	16-15/784	32-22/724	48-31/637
	P3	16-56/1346	32-14/1278	48-29/1460
Scenario 1 4 Satellites Vmax=2016	P0	17-53/324	34-16/290	50-47/428
	P1	18-04/1912	34-22/1953	50-39/1887
	P2	18-11/956	34-19/842	50-16/913
	P3	?	34-11/1314	50-29/1411

Table 1. Scenario 1 [-70, +70] latitude band coverage without restarting strategy

Vmax=2016			PSD	
		a,i	a,i, e, w	a,i,e,w,W,M
3 Satellites 1	P1	1700/1934	2100/1953	3050/1985
4 Satellites	P1	1467/2016	3300/1947	5400/2016

Table 2. Scenario 2 [-70, +70] latitude band coverage with restarting strategy

References

- [1] Fuhan Tian. The Influence Of Crossover Operator On The Diversity In Genetic Algorithms. *IC-AI conference*, 2000.
- [2] F. Glover Genetic Algorithms And Scatter Search : unsuspected Potentials, Statistics And Computing. , 1994.
- [3] F. Glover Tabu Search Part I.. *ORSA Journal On Computing*, 1989
- [4] F. Glover Tabu Search Part II. *ORSA Journal On Computing*, 1990.
- [5] F. Glover and M. Laguna Tabu Search. *Kluwer Academic Publishers*, 1997.
- [6] F. Glover and al. Finite Convergence Of Tabu Search. *MIC conference*, 2001.
- [7] A.E. Smith and al. Genetic Optimization using a penalty function. , 1992.
- [8] J. Grasmeyer Application of genetic algorithm with adaptive penalty functions to airfoil design. *48th International Astronautical Federation Congress*, 1997.
- [9] J.T. Richardson and al. Some Guidelines For Genetic Algorithm With Penalty Functions. , 1993.
- [10] E. Grandchamp and V. Charvillat Metaheuristics to Design Satellite Constellations. *Proceedings of MIC'2001*, 2001.
- [11] E. Grandchamp and V. Charvillat Integrating Orbit Database And Metaheuristics To Design Satellite Constellation. *Proceedings of ICAI'2000*, pages 733-739, 2000
- [12] E. Grandchamp and V. Charvillat Satellite Constellations Optimization With Metaheuristics. *Proceedings of Euro'2000*, pages 139, 2000.
- [13] E. Grandchamp Some contribution to Satellite Constellation Optimization. *PhD Thesis*, 2001.
- [14] D. B. Fogel An Introduction To Simulated Evolutionary Optimization. *IEEE*, 1994.
- [15] D.G. Goldberg Genetic Algorithms In Search, Optimization And Machine Learning, Addison Wesley. , 1989.
- [16] TZay Y. Young and Thomas W. Calvert Classification Estimation and Pattern Recognition. *Elsevier*,
- [17] Zadeh and al. Fuzzy Sets. *Information and Control*, 1965

- [18] Bandler and al. Fuzzy Power Sets and Fuzzy Implications Operators. *Fuzzy Sets and Systems*, 1980
- [19] Dubois and al. A Class of Fuzzy Measures Based on Triangle Inequalities. *Int. J. Gen. Sys.*, 1983
- [20] P.J Rousseeuw and al. Robust regression and outlier detection. *John Wiley and Sons*, 1987
- [21] P.J. Huber Robust Statistics. *John Wiley and Sons*, 1981
- [22] P.T. Boggs et and al. Orthogonal Distance Regression. *Contemporary Mathematics*, 1990
- [23] Åke Björck Numerical Methods for Least Squares Problems. *SIAM*, 1996
- [24] M. Vasquez and J-K. Hao "Logic-Constraint" Knapsack Formulation And A Tabu Algorithm For The Daily Photograph Scheduling Of An Earth Observation Satellite. *Journal Of Computational Optimization And Applications*, 2000
- [25] M. Vasquez and J-K. Hao A Heuristic Approach For Antenna Positioning In Cellular Networks. *Journal Of Heuristics*, 2000
- [26] CNES Techniques Et Technologies Des Véhicules Spatiaux. *Cepadues Editions*, 1989
- [27] F. Zhang and al. Optimization Design Method and Its Application To Satellite Constellation System Design. , 2001
- [28] A. Benhallam and al. Contribution Of Hybridization To The Performance Of Satellite Navigation Systems. , 1999
- [29] P. Jannière and al. The Use Of Genetic Algorithms For The Design Of Future Satellite Navigation Systems. *ASTAIR*, 1994
- [30] K. Kimura and al. Optimum Constellations Using Elliptical Inclined Synchronous Orbits For Mobile And Fixed Satellite Communications. , 1999
- [31] M. Bello Mora and al ORION - A Constellation Mission Analysis Tool. *48th International Astronautical Federation Congress*, 1997
- [32] F. Dufour and al. Constellation Design Optimization With A Dop Based Criterion. *14th Int. Symposium On Space Flight Dynamics*, 1995
- [33] T.A. Ely Satellite Constellation Design For Zonal Coverage Using Genetic Algorithms. *Space Flight Mechanics Meeting*, 1998
- [34] H. Baranger and al. Global Optimization Of GPS Type Satellite Constellations. *42th International Astronautical Federation Congress*, 1991
- [35] T. W. Beech and al. A Study Of Three Satellite Constellation Design Algorithms. , 1999
- [36] C. Brochet and al. A Multiobjective Optimization Approach For The Design Of Walker constellation. *50th International Astronautical Federation Congress*, 1999
- [37] N. Sultan and al. Effect Of Mobile Satellite System Design On Cost / Revenues Of Total Earth and Space Segments. *47th International Astronautical Federation Congress*, 1996

- [38] F. Dufour and al. A Multistage Approach To Design And Optimize A Communication Satellite Constellation. *50th International Astronautical Federation Congress*, 1999
- [39] Greistorfer Peter On The Algorithmic Design In Heuristic Search. *Euro conference*, 2000
- [40] P. Hansen and al. Variable Neighborhood Search: Principles And Applications. *Les cahiers du GERAD*, 1998
- [41] E. Lansard and al. Operational Availability. *48th International Astronautical Congress*, 1997
- [42] Thomas Bäck and al. An Overview Of Evolutionary Algorithms For Parameter Optimization. *Evolutionary Computation*, 1993
- [43] Z. Zhang Parameter Estimation Techniques: A Tutorial With Application To Conic Fitting. *Image And Vision Computing*, 1997
- [44] Hastings and al. The Generalized Analysis Of Distributed Satellite Systems. , 1995
- [45] J. Draim Optimization Of The Ellipso and Ellipso-2G Personal Communication Systems. , 1996
- [46] R. G. Brown A Baseline GPS RAIM Scheme And A Note On The Equivalence Of Three RAIM Methods. *Journal Of The Institute Of Navigation*, 1992
- [47] Bradford W. Parkinson GPS : theory and applications Volume I. , 1993
- [48] D. Diekelman Design Guidelines For Post 2000 Constellations. *48th International Astronautical Federation Congress*, 1997
- [49] M. Romy-Merino and al. Design Of High Performance And Cost Efficient Constellations For GNSS-2. *GNSS*, 1998
- [50] E. Lansard and al. Global Design Of Satellite Constellations : A Multi-Criteria Performance Comparison Of Classical Walker Patterns And New Design Patterns. *47th International Astronautical Federation Congress*, 1996
- [51] E. Frayssinhes Investigating New Satellite Constellation Geometries With Genetic Algorithms. *American Institute Of Aeronautics and Astronautics*, 1996
- [52] F. Zwolska and al. Optimization Of A Navigation Constellation Design Based On Operational Availability Constraint. *GNSS*, 1998
- [53] V. Martinot and al. Deployment And Maintenance Of Satellite Constellations. *49th International Astronautical Federation Congress*, 1998
- [54] S. Abbondanza and al. Optinav : An Efficient Multi-Level Approach For The Design Of Navigation Constellations - Application To Galileo. *GNSS*, 1999
- [55] Draim 9th DARPA Strategic Space Symposium. *GNSS*, 1983
- [56] T. Lang and J.M. Hanson Orbital Constellations Which Minimize Revisit Time. *Astrodynamics Conference*, 1983
- [57] J. G. Walker Circular Orbit Patterns providing whole earth coverage. *Royal Ear Craft Establishment Technical Report*, 1970