Multi-Objective Scheduling for NASA's Future Deep Space Network Array

Mark D. Johnston
Jet Propulsion Laboratory – California Institute of Technology
4800 Oak Grove Dr., Pasadena CA 91109
mark.d.johnston @jpl.nasa.gov

Abstract

We describe a multi-objective approach to the scheduling problem presented by the future NASA Deep Space Array-Based Network (DSAN), the communications system that will support missions from high earth orbit to the outer planets. The DSAN is envisioned to consist of several large arrays of smaller antennas that can be flexibly combined for each spacecraft communication session. This is in contrast to today's Deep Space Network (DSN), which typically relies on a single large antenna to handle each spacecraft contact. Multi-objective techniques for schedule optimization have the attractive advantage of *explicitly* capturing the constraints and preferences of the missions, as well as those based on system-level considerations, thus providing unique insight into trade-offs among competing requirements. We have investigated problem representation issues, objective and constraint formulations, and multi-objective optimization techniques that can be applied to this problem. Evolutionary multi-objective algorithms appear particularly promising. We have developed a testbed for investigating and assessing these algorithms, and describe our initial results on an illustrative sample problem, and on an estimated 2015 mission set with a three site, 300 antenna array.

Introduction

Plans for the evolution of NASA's Deep Space Network (DSN) feature the development of an arraying capability that will link together large collections of less expensive smaller antennas as required to support the communications requirements of future missions. As this development occurs, the current network scheduling process will have to move towards a much greater degree of automation. This paper describes an approach to automated scheduling of the Deep Space Array-Based Network (DSAN) that explicitly recognizes and supports one of its essential characteristics, namely, that it is designed to serve a diverse and competing user community.

In the following we first describe the expected characteristics of the DSAN and how its scheduling will likely be driven by mission and system requirements. We also discuss the kinds of objectives and constraints that are expected to apply to the DSAN. Next we briefly describe the nature of multi-objective optimization in general, and evolutionary algorithms that are proving successful in a variety of other problem domains. We formulate the DSAN scheduling problem as a multi-objective optimization problem, and apply evolutionary algorithms both to a simple example problem, and to a model mission set for 2015. We conclude with a description of preliminary results and plans for further research.

The scheduling problem posed by the DSN as it exists today has been described in (Clement and Johnston 2005; Johnston and Clement 2005). There is a long history of efforts to automate various aspects of DSN operations, based on techniques such as heuristic search, linear programming, and iterative repair (Bell 1992; Kan, Rosas and Vu 1996; Chien, Hill, Govindjee et al. 1997). This effort continues today, and will merge into the development and deployment of the DSAN.

Deep Space Array-Based Network Scheduling

Although the DSAN is years from deployment, certain of its characteristics are known or can be estimated from today's DSN and the state of current array technology (Bagri and Statman 2004; Gatti 2004; Statman and Gatti 2006). Current concepts (Figure 1) call for multiple sites, similar to today's three sites, spaced roughly equally in longitude. The sites may have similar antenna configurations, but there is also the possibility of an unbalanced network with a significant difference in capability from one site to another.

The drivers on the development of the DSAN come from a number of sources. The number of missions is increasing (3x by 2030), as are data rates and volumes (100x or more by 2030) and communications difficulty (due to the combination of higher data rates and greater distance). Manned missions place extremely stringent requirements on availability and reliability. There are more "cluster" (multi-spacecraft) missions planned in the future. There is pressure on the current DSN operations to reduce costs while retaining high levels of service availability 24x7. The current DSN large antenna assets are aging and have high maintenance and replacement costs. One of the goals of a DSAN is to reduce operations and maintenance costs by a significant factor over today's DSN, while providing service to a larger and more demanding mission set. Automation of the DSAN planning and scheduling functions is one of the most important elements of the approach to reducing costs.

An array-based network offers a number of advantages compared to the current DSN. Allocation of antennas can be more granular: anywhere from a single antenna to the entire array can be allocated to a pass, depending on need,

To appear in Proceedings of the International Workshop on Planning and Scheduling in Space (IWPSS 2006), Space Telescope Science Institute, Baltimore MD USA, 22-25 Oct. 2006. JPL Clearance CL#06-2907

where a pass is defined as a specific allocation of antennas to missions over some time interval. Multiple simultaneous passes can be handled by subsets of the array. Antenna allocation can be time-phased, even within a single pass: for example, more antennas can be dedicated to the start and end of a pass when the spacecraft is at a lower elevation. Unused antennas could be switched in to support a pass in case of equipment failure or spacecraft emergency. Thus the array will be much more flexible, but will consequently require an approach to planning and scheduling that can take advantage of this flexibility.

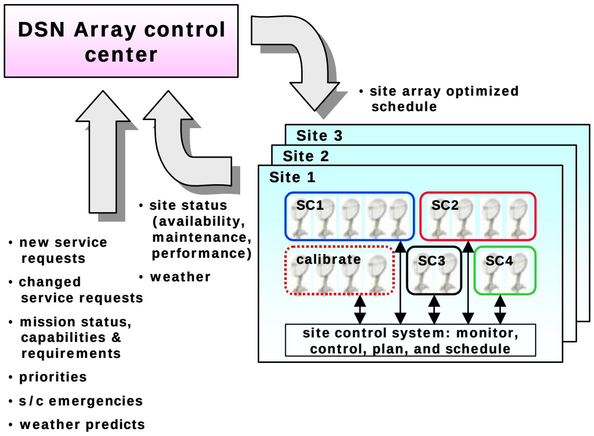
DSN Array control center
• site array optimized schedule
Site 3
Site 2
Site 1
SC1
SC2
calibrate
SC3
SC4
site control system: monitor, control, plan, and schedule
• site status (availability, maintenance, performance)
• weather
• new service requests
• changed service requests
• mission status, capabilities & requirements
• priorities
• s/c emergencies
• weather predicts

Figure 1: Conceptual overview of DSAN operations

While there are many compelling advantages of a DSAN, there are also some major challenges. The shift to higher frequency (Ka band in particular) makes communications more sensitive to weather than before (much less of an issue for X and S bands). There will also be array-specific constraints, for example that a sufficient number of signal combiners be available to support the scheduled number of simultaneous passes. There may also be constraints on the physical subset of antennas that can be allocated to a pass, e.g. based on antenna spatial location within the array.

It is important to note that the DSAN will almost certainly not be developed as a parallel network with a cutover point from the current DSN. Instead, the development is likely to be incremental over a number of years, so that there will exist a heterogeneous system with both array components and large antennas. A viable automation approach must take this likely evolutionary path into account.

Mission Requirements

As with today's DSN, the array will serve a diverse community of mission users. Missions place a variety of constraints and preferences into their requests for communications services (Clement and Johnston 2005). Many requests are for data downlinks, such as science or engineering data collected in the course of normal mission operations. Uplinks are required to send commands and table loads. Other requests are for support for navigation and control, such as precisely locating a spacecraft in order to plan for maneuvers and course corrections. In the profile for many missions are certain critical events, where communications coverage is especially important: examples include engine burns for trajectory correction maneuvers, and orbit insertion around planetary targets. And when anomalies occur, missions often have sudden requirements for additional communications to diagnose and recover the spacecraft. In addition to spacecraft operations support there are other categories of users, such as those using the network assets for radio science and radio astronomy.

Scheduling Objectives: Mission Perspective

To formulate an optimization problem for DSAN schedules it is necessary to quantify the objectives to use in their evaluation. From the mission perspective, these objectives can be categorized based on how they relate to the schedule:

- *pass-based objectives*: these are defined in terms of specific allocations of antennas to a mission over some time interval, designated a "pass". For example, the assignment of 25 antennas at Goldstone to a specific mission during a specified 6-hour visibility period would represent a single pass. A scheduling objective may be defined with respect to specific attributes of the pass, including its duration, timing relative to other passes or to a specified absolute time, etc.
- *service-based objectives*: defined at a higher level than passes, these measure mission needs in terms of how well a service requirement is satisfied. For example, a mission could specify that it needs an 8-hour downlink at a certain data rate, every 3 days over some mission phase. A candidate schedule can be evaluated in terms of how well it satisfies the service requirement.
- *model-based objectives*: at an even higher level, objectives in this category require the scheduler to model some aspect of the mission behavior in order to assess the schedule. For example, the preceding case of an 8 hour pass every three days may actually be derived from a mission's average science data collection rate plus onboard storage capacity. Were the scheduler to have insight into the actual mission science plan and onboard recorder usage, it could schedule passes that more accurately fit with missions needs: more passes in periods of high data collection, fewer when data rates are lower.

In today's DSN scheduling process, requirements are often formulated in pass-based or service-based terms, with the intention to move to a purely service-based specification as the DSAN comes online. While it might make sense in today's DSN for a mission to request the 70m Goldstone antenna for a particular downlink window, in the DSAN era it is not feasible for a mission to request a specific subset of a site's 400 antennas. Model-based objectives are presently not used in DSN scheduling, although it is clear that a better fit to mission needs could be possible if they were, as well as greater flexibility in constructing and modifying schedules in reaction to unexpected events.

Table 1 lists a subset of downlink service-based objectives that we have incorporated in the scheduling model described in this paper. This is by no means a complete list, but does capture many of the factors that relate to the timing of communications service requests.

Objective	Description
contact duration	min and max limits on duration, where a contact is the union of the coverage intervals of overlapping passes
contact gap duration	min and max limits on the sizes of any gaps between contacts
pass duration	min and max limits on individual pass duration
gap duration	min and max limits on the sizes of any gaps between individual passes
coverage fraction	fraction of some specified time interval with scheduled contact coverage (e.g. "1" means continuous coverage)
coverage level	number of distinct passes simultaneously providing coverage (e.g. "2" would mean simultaneous coverage at two different sites)
total gap duration	total gap in coverage over a specified interval
pass time shift	how much a pass has shifted in time from some baseline requested time
objective out of limit	extent to which an objective value exceeds a specified limit

Table 1: Objectives on communications timing, coverage, and gaps that may be relevant to a mission

It is worth noting that the definition of a mission's objectives is often not monolithic: many missions comprise multiple teams (e.g. based on the instrumentation on a mission), and so the detailed mission objectives are frequently the result of an internal evaluation, compromise, and tradeoff process.

The allocation of antennas in today's DSN is typically one by one, with the exception of certain special cases that make use of multiple antennas. In the array era, a more detailed model of downlink margin makes it possible to allocate differing numbers of antennas, depending on the power to noise ratio P_t/N_o of the link. For example, at low elevation, attenuation in the earth's atmosphere means that a larger number of antennas are necessary to adequately receive transmissions sent at a certain power level, but closer to the zenith the same transmission can be received with fewer antennas (e.g. Figure 2). A mission may specify one or more transmitter configurations, each of which will imply a certain minimum number of antennas, as a function of time, during a visibility period. In addition, weather uncertainties might suggest that additional antennas be allocated, or at least held in reserve, should atmospheric conditions have a larger impact than anticipated.

Scheduling Objectives: System Perspective

We can collect all of the non-mission objectives into a single "system" user, reflecting responsibilities for operations of the DSAN from the site level on up. System objectives fall into several categories:

- *satisfy the users of the system*: as the fundamental purpose of the network is to service users, this is clearly a critical objective. However, because we consider explicitly the degree of satisfaction of each mission's objectives, we do not need to separately include them again here.
- *minimize operational costs*: while clearly an objective of the overall system, there is likely to be little that impacts this from the scheduling point of view. On the assumption that scheduling itself will be highly automated, as will operation of the antennas and associated electronics, then the cost of executing one specific schedule will be similar to another.
- *maximize system availability*: this is the core objective of automated scheduling, in that effective scheduling can increase the real capacity of a given system. In this situation, the objective is to maximize the number of users that can be serviced for a given investment in assets and infrastructure. For a fixed number of users, another way to formulate this is to maximize the number of available unallocated antennas, since these can be used to mitigate risk and react to unexpected events, ranging from ground failures, bad weather, opportunistic science, or spacecraft emergencies.

A number of operations policies fit in with the system level objectives. For example, the approach to maintenance could be that antennas are out of service for scheduled maintenance on a pre-determined basis. Alternatively, the scheduling of maintenance can itself be optimized along with other objectives, thus allowing e.g. for adjustments to the maintenance timeline when the result benefits the overall schedule. Another policy issue is that of mission prioritization: whenever there is contention for resources, a scheduling solution can be sought but is not always available. In this case a decision must be made in favor of one or the other of the contending users. This will be discussed further below in the context of multi-objective optimization.

Constraints

Constraints are scheduling factors that must be satisfied in order for a schedule to be feasible. Like objectives, they may be considered from mission and system perspectives.

Mission constraints on the DSN schedule may be formulated in terms similar to objectives, with the significant difference being their importance. For example, during a mission critical event, what might otherwise be a preference for communications timing can be elevated to the highest level of importance.

System constraints include those based on resource availability, for example, the number of antennas available at a site as a function of time, or the availability of required electronics. One constraint expected for the DSAN is that each separate simultaneous pass requires a signal combiner, of which there are a limited number at each site. There may also be constraints on the spatial configuration of antennas that can be used in a single pass.

To the extent that constraints can be formulated as quantitative measures on the schedule that represent degree of violation of the constraint, they may be regarded as similar to objectives: we seek schedules that minimize constraint violations. This is important because a particular scheduling problem may have no feasible solution, so we need to determine which constraint relaxations might be considered. This suggests that a problem formulation that treats objectives and constraints similarly can be very useful.

Multi-Objective Optimization

In the DSAN scheduling problem there are potentially numerous objectives from the missions and other users, as well as from the overall system level. Conventional approaches to schedule optimization would combine these into a single objective value to optimize, as in e.g. (Cheung, Lee and Ho 2005). However, there is a significant drawback to this approach, in that it pre-specifies the tradeoffs among the different objectives in the common and interesting case where there is no solution that simultaneously optimizes all of the separate objectives. That is, for any given value of the combined single objective, the results of optimization will not distinguish between cases where any one mission's objectives are satisfied anywhere from fully to not at all. From the perspective of a mission or user, such an optimization process is at least cause for concern.

An alternative approach is to use multi-objective optimization techniques, which retain the information in separate objectives until the end of the optimization process: see e.g. (Collette and Siarry 2003) and references therein. When there remains contention to be resolved, a multi-objective approach will provide explicit information about the tradeoffs involved. Such visibility is important to users who may be required to compromise on the achievement of their objectives.

Among techniques developed to solve multi-objective optimization problems, evolutionary algorithms (Bagchi 1999; Deb 2001; Madavan 2002; Abraham, Jain and Goldberg 2005; Kukkonen and Lampinen 2005; Robic and Filipic 2005) have become popular for a variety of reasons. They are capable of dealing with objectives that are not mathematically well behaved (e.g. discontinuous, non-differentiable). A solution is *Pareto-optimal* when no improvement can be made to one objective which does not make worse at least one other objective. Because evolutionary methods maintain a *population* of solutions, they can trace out the *Pareto frontier*, an approximation to the entire set of Pareto-optimal solutions. Evolutionary algorithms also lend themselves to parallelization, which can be an advantage for large problems. Two important performance characteristics of a multi-objective evolutionary algorithm are *convergence* to the Pareto frontier, and *diversity* so as to maximally sample the frontier.

In the following we describe evolutionary algorithms for multi-objective optimization in general, then two specific variants that we have explored for applicability to DSAN scheduling.

Evolutionary Algorithms

The problem we consider is minimizing a set of *M* objectives subject to *K* constraints:

$$\text{minimize: } \{f_i(\vec{x})\},\ i = 1\ldots M$$

$$\text{subject to: } (g_j(\vec{x}))^T \le 0,\ j = 1\ldots K$$

Here $\vec{x}$ represents a vector in decision space of dimension *D*. Where necessary below, we refer to the i^{th} member of the population at generation *g* with $\vec{x}_{i,g}$.

An evolutionary algorithm formulation for multi-objective optimization consists of a population of *N* solution candidates. With each step (generation) of the algorithm, we evolve the population according to method-specific rules for crossover and mutation. Crossover combines elements of parent solution candidates to generate a new offspring candidate for evaluation. Mutation introduces randomized variation in the offspring.

Depending on the procedure, following the crossover and mutation steps the size of the population may have increased. If so, it is reduced back to *N* through a selection procedure. The overall process repeats, with a cutoff generally determined by a specified maximum number of generations G^{max}.

NSGA II. This algorithm, developed by (Deb, Pratap, Agrawal et al. 2002) has done well on a variety of test problems and applications. The specifics of this algorithm include:

- *non-dominated sorting* of the population into *ranks*, such that members of rank *n* dominate members of all ranks >*n*. Rank 1 members constitute the non-dominated set, that is, the current approximation to the Pareto frontier.
- *crowding distance* is used as a secondary discriminator on members of the same rank: members in crowded regions are scored lower, so the surviving members after selection have greater diversity. This helps prevent premature convergence.
- members are compared with a *domination* or *constrain-domination* relation – the latter allows for comparisons even when constraints are violated.

The algorithm is *elitist*, in that highest ranked solutions are preserved from one generation to the next.

GDE3. GDE3 is a recent algorithm (Kukkonen and Lampinen 2005) that uses Differential Evolution (DE) (Price, Storn and Lampinen 2005) as the basis for crossover and mutation operations, and non-dominated sorting and crowding distance in a manner similar to NSGA II. DE is an evolutionary algorithm for optimization, initially developed in a single objective context. DE is defined on real-valued decision spaces. It generates offspring through the following procedure:

1. For each parent $\vec{x}_i$, select three distinct population members $\vec{x}_{r_1}, \vec{x}_{r_2}, \vec{x}_{r_3}$, all different and different from parent
2. Calculate a trial vector $\vec{y}_i$ as:
$$\vec{y}_i = \vec{x}_{r_1} + F \cdot \left(\vec{x}_{r_2} - \vec{x}_{r_3} \right)$$
where F is a scaling factor
3. Modify the trial vector by binary crossover with parent with probability *CR*

The result is compared with the parent as follows:

- in the case of infeasible vectors, the trial vector is selected if it weakly dominates the parent vector in constraint violation space, otherwise the parent vector is selected
- in the case of feasible and infeasible vectors, the feasible vector is selected
- if both vectors are feasible, then the trial is selected if it weakly dominates the parent in objective space; if the parent dominates the trial, then the parent is selected, and if neither dominate, then both are selected

The selected vectors may constitute a set of size >*N*, in which case the population size is reduced through the non-dominated sorting and crowding distance mechanism of NSGA II.

GDE3 is appealing for several reasons: it provides a natural treatment of the *K* constraints, while reducing to standard DE when the number of objectives *M*=1. The treatment of constraints makes it straightforward to change constraints into objectives when investigating overconstrained problems. GDE3 performed very well in initial comparisons with NSGA II, and does not introduce any additional control parameters beyond *F* and *CR* from the original formulation of DE.

Multi-Objective Formulation of DSAN Scheduling

In its basic form, DSAN scheduling requires the decision of which antennas and other site resources to allocate to a mission, as a function of time. We consider a network modeled as one or more sites, each with associated resources such as antennas, combiners, etc. Each resource type has a quantity that may vary over time, e.g. as new antennas are brought online, or when electronics is taken out of service for a maintenance period. Resources are non-consumable and, within their category, indistinguishable. So, for example, if a mission requires 25 antennas for an array link, it does not matter for scheduling purposes which 25 are allocated. For simplicity below, we consider only antennas as a resource, but there is no limit on the allocation of multiple resources.

We consider a user collection of *U* users (i.e. missions and other users), over some scheduling time period $[T^s, T^e]$. Associated with each user is a set of view periods, each of which is a time interval during which some variable number of antennas at one site may be allocated. We denote the view periods by $[V^s_{up}, V^e_{up}]$, where $u = 1 \ldots U$ ranges over users, and $p = 1 \ldots P_u$ ranges over the set of view periods for each user. The minimum required time-varying antenna profile is given by $A^{req}_{up}\left(t \in V_{up}\right)$, which may differ from one view period to another. Above the minimum required level, additional antennas might be allocated, e.g. to improve signal strength in the face of uncertain weather: we denote the maximum additional allocation by Δ_{up}.

Decision variables. To encode the problem in a way that preserves neighborhoods, so that a small perturbation in the value of the decision vector will (usually) result in a small change to the scheduled allocation, we define the following correspondence between decision variables and resource allocation (see Figure 2), suppressing the *up* subscripts:

1. For each view period, define a triple of real-valued decision variables $\xi_1, \xi_2, \xi_3 \in [0,1]$
2. Calculate the start and end of the allocated portion of the view period as:
$$t^s = V^s + \xi_1 (V^e - V^s)$$
$$t^e = t^s + \xi_2 (V^e - t^s)$$
3. Calculate the allocated antenna quantity as:
$$A(t) = A^{req}(t) + ceil(\xi_3 \Delta)$$

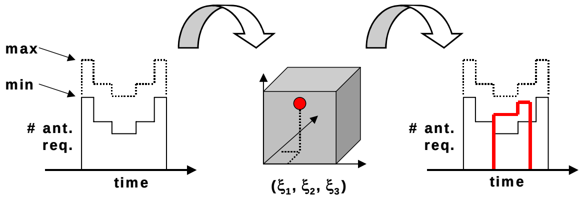

Figure 2. Decision variables for array antenna allocation.

This correspondence leads to the total number of decision variables *D* given by:

$$D = 3 \sum_{u=1}^{U} P_u$$

For concreteness, this corresponds to about 60 decision variables per mission per week. Note that if the antenna allocation quantity is constant, there is no need for a third decision variable ξ_3, and so it can simply be omitted.

We have adopted an approach of quantizing time in the schedule in order to calculate quantities such as resource loading and coverage and gap measures. This quantization can be straightforwardly incorporated in the above calculation, noting that if a view period contains *m* intervals, we quantize to *m+1* so that zero can correspond to the decision to not allocate any resources in the view period.

Objectives. We have taken the approach in our initial experiments that each modeled user specifies a single objective function in the form of a penalty to be minimized. The objective may be made up of an arbitrary number of additive *phases* that delimit a time period over which the component penalty calculation is made. For example, a mission might have one phase that requires relatively infrequent periodic coverage, followed by a phase that requires continuous simultaneous coverage at two sites. Associating a penalty calculation with a phase makes it straightforward to define the correct behavior in each.

Components of the penalty function may correspond to any of the objectives listed in Table 1. The calculation of objectives proceeds as follows:

1. Convert the decision variable values into allocation parameters following the prescription above
2. Calculate a "user perspective" view of the schedule, including coverage levels and gaps, based on resource allocations for that user. This calculation takes into account boundary effects that affect gap and coverage penalties.
3. Loop over each modeled user phase and component, calculating the deviation from full satisfaction, and accumulating the result as the penalty function measure.

Constraints. Constraints are implemented like objectives, i.e. a collection of phase-specific penalty calculations whose values are to be minimized in a multi-objective sense. Modeled users may have zero or more associated constraints. There is also a "system" user representing non-mission constraints. For example, the constraint that antenna allocation must not exceed the available quantity is modeled as a system constraint over a phase corresponding to the entire schedule.

Implementation. We have implemented a testbed for evaluating multi-objective DSAN scheduling, with the following features:

- generic models of networks and resources, and users and their requirements, to facilitate experiments with different scenarios
- pluggable evolver modules, to make it easy to vary algorithms and parameters while running the same test problems
- results recorded every *g* generations for graphical display and restart continuation

Initial results are discussed in the next section.

Results

Example: two missions in contention

We first describe a simple example that highlights the potential of multi-objective optimization in this kind of problem. Consider a 2-mission scenario with the following characteristics:

- two identical missions, user1 and user2, with periodic communications requirements
- each mission requires a constant allocation of 5 antennas for each link (constraint)
- each mission requires contacts that are at least 3 hours in duration (constraint), with a preferred duration of 12 hours (objective)
- gaps between contacts are limited to 18 hours (constraint)
- the scheduling interval is 4 days in duration
- both missions have the same 12 hour view period each day
- a single site with a fixed number of 10 antennas is available at all times *except* day 2, when only 5 are available

Were it not for the shortage of antennas on day 2, this problem would have a trivial solution where each mission could maximally achieve all of its objectives. As it is, there will be contention on day 2 for the available antenna resources. When this problem is encoded as described above, it has the following characteristics:

- two objectives: the penalty functions for user1 and user2
- one constraint for each user, for minimum pass size
- one system level constraint, to enforce antenna allocation levels to not exceed the available quantity
- 16 decision variables, two for each user/viewperiod combination (antenna allocations are constant, so only two decisions variables per viewperiod are required)

Figure 3 shows the evolution of the non-dominated set after various numbers of generations using the GDE3 algorithm. (Results for NSGA II were poorer, in terms of convergence to the Pareto frontier and the uniformity of coverage, and so are not included here.) The parameters in this run were: time resolution = 1 hour, population size *N*=400, *F*=0.5, *CR*=0.1. The initial conditions were uniformly random within the bounds of the decision variable limits.

Initially (Gen#0, upper left), the population is nearly all constraint violated (red "x"), since a random decision variable vector is very likely to exceed either the resources on day 2, or the minimum gap constraint. After 20 generations (upper right) the population has evolved to consist entirely of feasible solutions, and shifts to the lower left as increasingly smaller values of the penalty values are discovered. By generation 140 (lower right) the population has evolved to the Pareto frontier for this problem. Values at the extreme (user1.penalty=30 or user2.penalty=30) represent solutions where the available 5 antennas are allocated entirely to the other mission. The remaining values along the frontier represent partial allocations to each mission. The gaps in the curve reflect the constraint that solutions with pass durations <3 hours are excluded as not feasible.

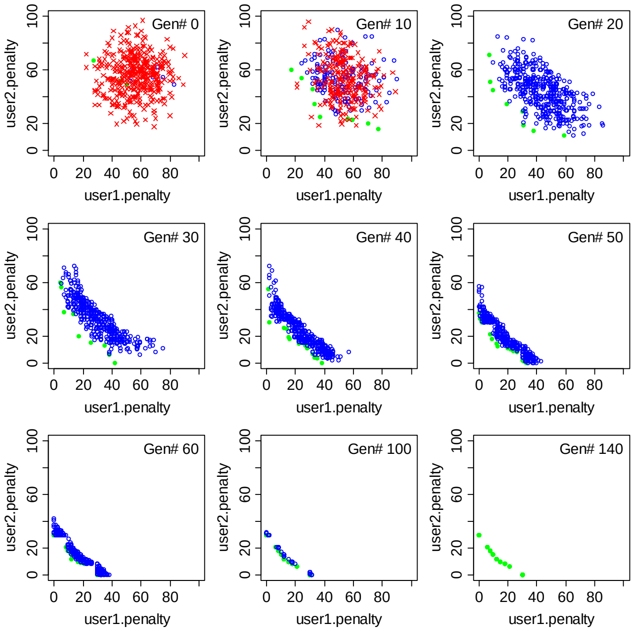

Figure 3: Objective values for the two users, illustrating the evolution of the population towards the Pareto frontier with GDE3. An “x” (red) is an infeasible solution, and a filled circle (green) is in the non-dominated set. Open circles (blue) make up the rest of the population.

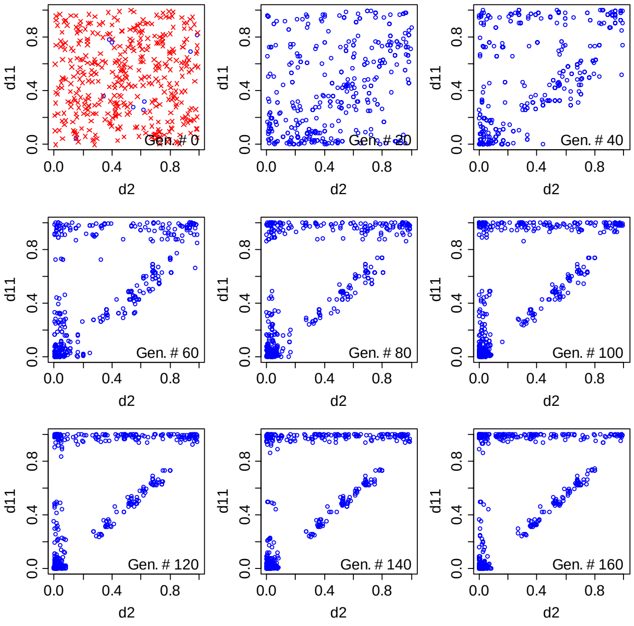

Figure 4: The evolution of two of the decision variable values versus generation number. An “x” (red) indicates an infeasible solution, and an open circle (blue) a feasible one.

In Figure 4 we have plotted the values of two of the 16 decision variables against each other, for a sample of population snapshots. The variable labeled *d2* corresponds to the start of the user1 allocation on day 2, and *d11* to the duration of the user2 allocation. The initial random distribution (generation 0, upper left) evolves into a strong correlation (generation 160, lower right) that shows how the population has captured the relationship of these variables along the Pareto frontier. The bands at *d2*~0 and *d11*~1 correspond to degeneracies near the extremes when time intervals are quantized.

Tradeoffs and Prioritization. This example highlights a strength of multi-objective optimization, that of exposing the detailed nature of the tradeoffs when several objectives cannot be simultaneously optimized. However, it also illustrates the remaining problem of deciding which point on the Pareto frontier to actually select. There are a number of ways to approach this:

- rank order the objectives, then select the optimal points for each in order
- construct a weighted combination of objectives and select the solution that optimizes this value: this is similar to the multi-to-single objective mapping discussed above. It is less extreme than priority-based selection, and if constructed with knowledge of the Pareto frontier, it can be much more informed.
- allow the users who defined the objectives to negotiate among themselves to come up with an acceptable compromise. In contrast to the example here, where one user’s gain is the other’s loss, there are many realistic situations where users can “horse-trade” to advantage.

A 2015 mission set

A more extensive model has been constructed from an extrapolated 2015 mission set using assumed 12m X-band arrayed antennas. The salient characteristics of this problem are:

- 17 missions requiring from 1 to 94 antennas for each pass
- periodic pass requirements ranging from every 8 hours to once per week, with pass durations ranging from 1 to 12 hours, and a 20% flexibility in pass and gap duration
- three sites equally spaced in longitude, each with 100 antennas
- a one-week scheduling interval with one hour resolution

This problem consists of 760 decision variables and 17 objectives (one per mission user). Preliminary runs have been made for different population sizes and flexibility levels, with encouraging results. One measure of the behavior of the algorithm is the summed value of all the objectives for points on the non-dominated front. Since we are minimizing all objectives, the decrease in this sum gives some view into the “condensation” of the population to the Pareto frontier. The results are shown in Figure 5 for a run with a population size of 1,000 and a total of 20,000 generations. When started with a uniform random distribution in decision space, the population starts out entirely infeasible. The first approximately 1000 generations are spent finding a feasible population, which then evolves progressively and rapidly to reduce the penalty values for all users. The rate of progress eventually slows, but continues to show steady progress as the number of generations increases. Since the minimum penalty function value for each user is defined to be zero, the summed value shows that the algorithm has come close to a configuration that optimizes the objectives of all users (the scale is such that a failure to satisfy a duration or gap preference by *n* hours would contribute *n* towards the penalty calculation).

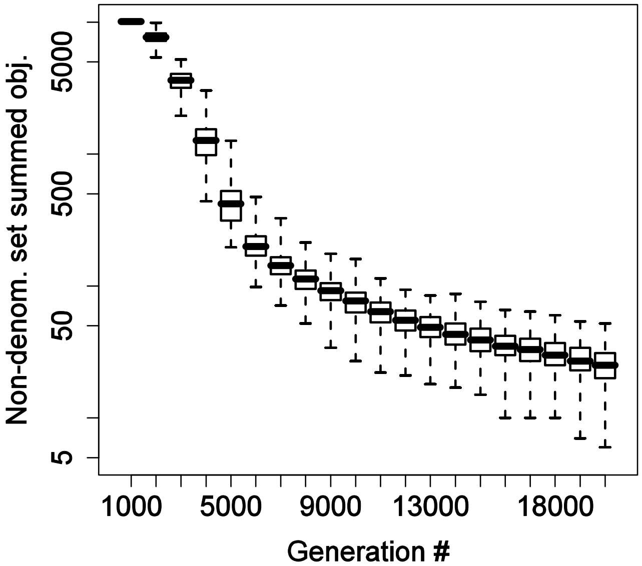

Figure 5: the summed value of objectives for population members in the non-dominated set, for a model 2015 mission set, scheduled for one week.

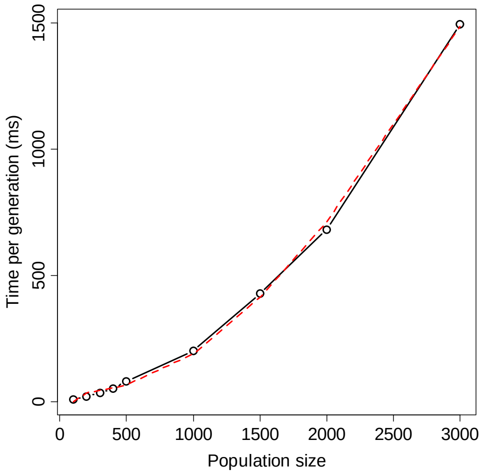

Figure 6: Scaling of runtime versus population size, evaluated for a single mission. The dashed line shows excellent agreement with the expected *NlogN* behavior.

Scaling

The scaling behavior of NSGA II and GDE3 is determined in both cases by the non-dominated sorting used in the selection and pruning step and is of order *MN log(N)* where *M* is the number of objectives and *N* the population size. The results shown in Figure 6 confirm the expected behavior of runtime versus population size.

The major computational effort in assessing each schedule is calculating resource utilization and mission coverage for the quantized timeline, then computing objective and constraint values. This calculation scales as the number of users U (for similar users) times the number of time intervals in the schedule. Figure 7 shows the scaling behavior of the time to assess the schedule as a function of time intervals per week. As expected, the behavior is linear. Depending on the time horizon, it may be adequate to schedule with resolutions of 1 hour for very long range horizons, whereas a resolution of down to 1 minute may be required for the nearest term schedules.

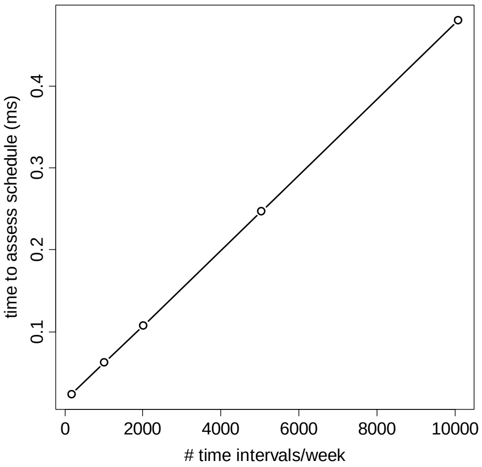

Figure 7: Scaling of schedule assessment (objective and constraint calculation) scales linearly with resolution.

Conclusions

In this paper we have described a new approach to Deep Space Array-Based Network scheduling based on multi-objective optimization techniques. Current evolutionary algorithms such as GDE3 appear to be promising candidates for schedule generation and modification. The strengths of this approach include:

- explicit and separate representation of different missions' objectives, making it easy to consider tradeoffs
- a population of non-dominated solutions that approximates the Pareto frontier, useful both in schedule selection and as a starting point when revising the schedule when changes occur

There are a number of areas that remain to be investigated, and we have developed a testbed to facilitate this. Some of the areas of interest include:

- investigating the impact of highly constrained problems on convergence properties: when feasible solutions are hard to find, the first ones encountered in an evolving population may disproportionately channel the solution to their vicinity, leading to premature convergence
- investigating the effect of different parameter choices (*F* and *CR*) for the Differential Evolution algorithm
- reconsidering the crowding distance calculation for this problem: (Kukkonen and Deb 2006) have developed an improvement to the NSGA II calculation, and have pointed out some problems with the calculation for higher dimensional objective spaces

In addition, we plan to investigate other aspects of the array scheduling problem, including:

- modifying an existing schedule quickly and effectively, e.g. to model last minute resource availability changes, or requests for emergency contacts
- modeling and scheduling multiple spacecraft per link, such as the missions presently at Mars
- scheduling multiple resources, such as combiners or other equipment for which joint allocations must be made
- investigating how to include maintenance scheduling into the framework as another optimization objective
- investigating how to incorporate risk mitigation into the schedule, for example in optimally allocating unused antennas as a hedge against bad weather

The research described in this paper was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

References

Abraham, A., L. Jain and R. Goldberg (2005). Evolutionary Multiobjective Optimization, Springer.

Bagchi, T. P. (1999). Multiobjective Scheduling by Genetic Algorithms. Boston, Kluwer.

Bagri, D. S. and J. Statman. (2004). "Preliminary Concept of Operations for the Deep Space Array-Based Network." The Interplanetary Network Progress Report, 42-157, from http://tmo.jpl.nasa.gov/progress_report/42-157.

Bell, C. (1992). Scheduling Deep Space Network Data Transmissions: A Lagrangian Relaxation Approach, JPL Technical Report.

Cheung, K.-M., C. H. Lee and J. Ho (2005). "Problem Formulation for Optimal Array Modeling and Planning." IEEEAC **438**.

Chien, S. A., R. W. Hill, Jr., A. Govindjee, X. Wang, T. Estlin, M. A. Griesel, R. Lam and K. V. Fayyad (1997). A hierarchical architecture for resource allocation, plan execution, and revision for operation of a network of communications antennas. Proceedings IEEE International Conference on Robotics and Automation Albuquerque, NM.

Clement, B. J. and M. D. Johnston (2005). The Deep Space Network Scheduling Problem. IAAI 2005, Pittsburgh, PA, AAAI Press.

Collette, Y. and P. Siarry (2003). Multiobjective Optimization. Berlin, Springer.

Deb, K. (2001). Multi-Objective Optimization Using Evolutionary Algorithms. New York, John Wiley & Sons.

Deb, K., A. Pratap, S. Agrawal and T. Meyarivan (2002). "A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II." IEEE Transactions on Evolutionary Computation **6**(2): 182-197.

Gatti, M. (2004). "The Deep Space Network Large Array." The Interplanetary Network Progress Report, 42-157, from http://tmo.jpl.nasa.gov/progress_report/42-157.

Johnston, M. D. and B. J. Clement (2005). Automating Deep Space Network Scheduling and Conflict Resolution. ISAIRAS 2005, Munich, Germany.

Kan, E. J., J. Rosas and Q. Vu (1996). Operations Mission Planner - 26M User Guide Modified 1.0, JPL Technical Document D-10092.

Kukkonen, S. and K. Deb (2006). Improved Pruning of Non-Dominated Solutions Based on Crowding Distance for Bi-Objective Optimization Problems. Proceedings of the 2006 Congress on Evolutionary Computation (CEC 2006). Vancouver, BC, Canada.

Kukkonen, S. and J. Lampinen (2005). GDE3: The Third Evolution Step of Generalized Differential Evolution. The 2005 Congress on Evolutionary Computation.

Madavan, N. K. (2002). Multiobjective Optimization Using a Pareto Differential Evolution Approach. 2002 Congress on Evolutionary Computation (CEC 2002). Honolulu, HI.

Price, K., R. Storn and J. Lampinen (2005). Differential Evolution: A Practical Approach to Global Optimization. Berlin, Springer-Verlag.

Robic, T. and B. Filipic (2005). DEMO: Differential Evolution for Multiobjective Optimization. Evolutionary Multi-Criterion Optimization (EMO 2005).

Statman, J. and M. Gatti (2006). Deep Space Network Array - Update. SpaceOps 2006 Conference, Rome, Italy.