Intelligent Automation and Soft Computing, Vol. 14, No. 3, pp. 367-376, 2008

AN EVOLUTIONARY ALGORITHM APPROACH TO MULTI-OBJECTIVE SCHEDULING OF SPACE NETWORK COMMUNICATIONS

MARK D. JOHNSTON
Jet Propulsion Laboratory
California Institute of Technology
4800 Oak Grove Drive
Pasadena CA 91109
mark.d.johnston@jpl.nasa.gov

ABSTRACT—We describe an evolutionary multi-objective optimization approach to the scheduling problem presented by the NASA's Deep Space Network (DSN). This network is the communications system that supports NASA and other space missions from high earth orbit to the outer planets. Today the DSN consists of a small number of large antennas, but in the future it is expected to incorporate several large arrays of smaller antennas that can be flexibly combined for each spacecraft communication session. Multi-objective techniques for schedule optimization have the attractive advantage of explicitly capturing the constraints and preferences of the missions that use the DSN, as well as those based on system-level considerations, and 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. We describe our initial results using an evolutionary algorithm on an illustrative sample problem (contention for a single antenna), and on a projected 2015 mission set with a three-site, 300 antenna array. The results are very promising, not only for generating initial schedules, but also for resolving conflicts where there is severe resource contention.

Key Words: scheduling; multi-objective optimization; evolutionary algorithms

1. INTRODUCTION

As space technology and missions have progressed in capability, the demands for communications from Earth have likewise grown. Over the next 25 years it is projected that the number of missions will increase by a factor of three, that data rates and volumes will grow by a factor of 100, along with a significant increase in data link difficulty. In addition, plans for human exploration of the Moon, and eventually Mars, will place even greater demands on data rates and on the quality and reliability of communications links. Communications resources capable of supporting this demand are expensive and oversubscribed, and this is likely to continue for some time. Consequently there is a compelling need to efficiently plan, schedule, and control the communications assets that support ongoing missions.

The Deep Space Network (DSN) is NASA's current framework for communications with missions located anywhere between high earth orbit to outside the solar system. It comprises three sites, located in Goldstone, California, USA; Madrid, Spain; and Canberra, Australia, each with a complement of large antennas (26m, 34m, and 70m) and associated radio communications equipment. However, these assets are aging and will be inadequate to support the anticipated growth in capacity over coming decades. Accordingly, there is a plan under investigation to move over time to a new communications architecture, consisting of large arrays of smaller, identical antennas, called the Deep Space Array Network (DSAN)[1]. Such an architecture has many advantages, including greater flexibility, lower hardware costs, and better scalability as capacity needs grow. However, to achieve sufficiently low-cost operations will require a very high degree of automation, which presents a major challenge. In addition, it is clear that the DSAN will not abruptly replace the existing DSN, but rather will phase in over an interval of years to as much as a decade. This means that for some extended period there will be a *heterogeneous* network consisting of today's assets combined with a growing array—thus presenting an even more challenging planning and scheduling problem.

In this paper we report on an investigation of the application of a class of evolutionary algorithms to the DSN/DSAN scheduling problem. Our work was motivated by one of the key features of the network scheduling problem, that of *satisfying its many users* as a primary measure of success. These users have competing objectives and must frequently compromise and tradeoff the satisfaction of their requests among other users as the schedule is developed. Thus the overall optimization problem is thus best characterized as *multi-objective*[2]. Previous work on this problem has concentrated on other aspects of this problem, including modeling[3], iterative repair[4], and systematic search[3, 5]. Related work on oversubscribed scheduling and space communications problems has been reported by Barbulescu et al.[6], Kramer and Smith[7], and Cheung et al.[8], however all of these authors characterize their objectives as single-valued.

In the following we describe the overall scheduling optimization problem we are addressing (Section 2), followed by a brief description of multi-objective optimization and the evolutionary algorithm technique we have investigated (Section 3). We discuss our results in Section 4, first on an illustrative conflict resolution scenario, then on a larger model representing a projected 2015 mission set. Of particular interest is the capability of the multi-objective approach to simultaneously assess alternative solutions based on user-specific objectives, thus immediately giving a view into the problem that is essential for later compromise and negotiation. We summarize our conclusions in Section 5 and note some areas for further investigation.

2. THE DSN/DSAN SCHEDULING PROBLEM

The DSN/DSAN scheduling problem has the following elements (see [2, 3] for a more detailed description):

- Users — generally missions (but not necessarily, e.g. radio astronomy and radio science investigators also use the network) who have an agreed level of access to the network resources. Users formulate their requirements as service requests that specify what they need, when, and with what flexibility.
- Assets — resources to be allocated to meet user requests, including large single antennas as well as (future) arrays of smaller antennas, subsets of which can be separately tasked to meet one user's requirements over some time interval.

Examples of requests include:

- A 12h downlink service for Voyager 1, on any of the DSN's 70m antennas, with no interruption, followed by a 6h uplink service within 18h, also on a 70m antenna, repeated every 72h with no gap longer than 60h • a 9h downlink for Cassini at Saturn, requiring at least 12 dedicated antennas from the DSAN, but preferably 15 to increase signal-to-noise, centered on times when Cassini has maneuvered to point its radio antenna at Earth.

2.1. Objectives

The optimization objectives in this problem most naturally break out on a user-by-user basis. User objectives can be viewed as quantifying a "degree of satisfaction" metric, where examples of factors that might contribute are provided in Table I. From an overall system perspective, optimization is driven by satisfying the maximum number of users, already addressed above. Objectives such as minimizing overall cost play only a minimal role since, once the network is in operation, it costs essentially the same whether it is highly utilized or not. In fact, the most important system-level objective is really to open as many gaps as possible in the schedule, and use them to service additional users. However, given a fixed set of users, a better system-level objective is to minimize risk to the schedule.

The heterogeneous nature of the DSN/DSAN array impacts the specification of user requests. On today's network of large antennas, a request generally specifies a set of mutually exclusive antenna choices. In a pure array architecture, a request would specify how many antennas are required, possibly as a function of time[1]. In a heterogeneous network, *both* of these possibilities must be allowed at the same time.

Table I. Examples of contributors to user-specific objective functions

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 (a single resource allocation over time)
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. a value of "1" indicates continuous converage)
coverage level	Number of distinct passes simultaneously providing coverage (e.g. a value of "2" would mean simultaneous coverage from 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 limiting range

2.2. Constraints

Constraints in the DSN/DSAN scheduling problem come from several sources. Mission constraints may be formulated in terms similar to objectives, the main difference being importance. For example, during a mission-critical event, what might otherwise be a preference for communications coverages may be elevated to the highest level of importance, such that no schedule without coverage will be considered. System level constraints include those based on overall resource availability, for example, reflecting maintenance schedules and the planned introduction of new assets.

It is important to note that constraints and objectives can play a complementary role in a practical scheduling problem, which we exploit in the solution approach described in the next section. For example, consider a problem which is overconstrained such that no solution exists. In this situation it is extremely useful to obtain some insight into what constraints must be relaxed, and by how much, in order to assess feasibility.

[1]The use of Ka band communications in the network architecture will lead to a greater sensitivity to atmospheric moisture; thus the number of required antennas may vary with spacecraft elevation, and also be larger in bad weather.

3. MULTI-OBJECTIVE OPTIMIZATION WITH GENERALIZED DIFFERENTIAL EVOLUTION 3

As discussed in Section 2, the DSN/DSAN scheduling problem is naturally *multi-objective* in that there is no single scalar that characterizes an optimal solution. The traditional approach to problems like this is to *construct* such a single objective, e.g. by taking some function of the individual user objectives. However, it is obvious that important information is lost when this is done. To avoid this, we have adopted a multi-objective optimization perspective (e.g. [9]), in which information about each objective is kept separate and is thus available to assess tradeoffs and sensitivity. Among the best current techniques for solving multi-objective optimization problems are evolutionary algorithms, in which a population of candidate solutions is developed and evolved[10, 11, 12, 13].

3.1. Multi-Objective Optimization

We define a multi-objective optimization problem to minimize M objectives subject to K constraints:

$$\text{minimize: } \{f_1\ (\vec{x})\},\ i = 1 \ldots M$$

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

Here $\vec{x}$ represents a vector in decision space of dimension D.

A solution is called *Pareto optimal*[2] when no improvement can be made to one objective which does not make worse at least one other objective. The set of Pareto optimal solutions is called the *Pareto frontier*. What we seek as a solution to the multi-objective optimization problem is a good approximation to the Pareto frontier. Two important characteristics of a good solution technique are *convergence* to the Pareto frontier, and *diversity* so as to sample the frontier as fully as possible.

3.2. Generalized Differential Evolution 3

Among techniques developed to solve multi-objective optimization problems, evolutionary algorithms have become popular for a variety of reasons. They have been shown effective on a wide range of problems and are capable of dealing with objectives that are not mathematically well-behaved (e.g. discontinuous, non-differentiable). By maintaining a population of solutions they are capable of representing the entire Pareto frontier at any stage. They also lend themselves to parallelization, which is an important performance consideration for large problems.

Here we concentrate on one particular variant called Generalized Differential Evolution 3, or GDE3[14]. This technique is based on Differential Evolution, a single objective evolutionary algorithm for real-valued decision spaces[15]. GDE3 makes use of concepts pioneered in the algorithm NSGA II[16], including:

- *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 of the population are scored lower, so the surviving members after selection have greater diversity. This helps prevent premature convergence of the population to a small portion of the Pareto frontier.

---

[2]See the discussion by Deb[11] of the key concepts and definitions of multi-objective optimization.

- Population members are *compared with a domination or constraint-domination* relation— the latter allows for comparisons even when constraints are violated.

GDE3 operates as follows to evolve the population of size *N* from one generation to the next:

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

The result is compared with the parent as follows:

- in 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; if neither dominates, then both are selected

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

In the work reported here, we implemented both NSGA II and GDE3, and then compared their behavior on several communications network problems modeled as in 3.3. We found that NSGA II was much slower to converge, and for a given number of iterations produced a much less well-sampled Pareto frontier. As a result, we have focused on GDE3 as a preferred algorithm. We plan in future work to better characterize the performance of GDE3 as compared with variants of other evolutionary algorithms on these problems. In addition to high performance, it is also worth noting that one of the strengths of GDE3 is its natural treatment of multiple constraints: it is straightforward to change constraints into objectives when investigating overconstrained problems. This is especially valuable when constraints must be relaxed in order to find any feasible solutions.

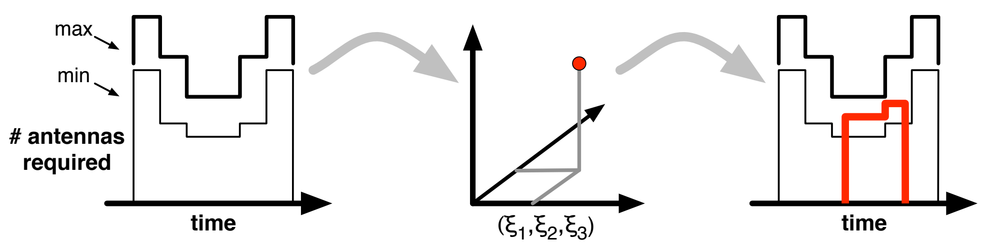

Figure 1. Decision variables for the array antenna allocation: a real-valued triple is sufficient to specify the antenna allocation profile over each possible view period (see Section 3.3)

3.3. Modeling the Network Scheduling Problem

In this section we describe the encoding method we adopted for representing the DSN/DSAN scheduling problem. 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 specific antenna is available for allocation to that mission, or in case of the array, when some number of array antennas at one site may be allocated.

We denote the view periods by [V_{up}^{s}, V_{up}^{e}], where u = 1. . .U ranges over users, and p = 1. . .P_u ranges over the set of view periods for each user. For array allocations, the minimum required time-varying antenna profile is given by $A_{up}^{reg}\left(t \in V_{up}\right)$, which may differ from one view period to another. Above the minimum required level, additional array 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} . For single antenna allocations, the profile function is constant Δ_{up}^{reg} = 1, and Δ_{up} = 0.

For decision variables we selected a mechanism that preserves neighborhoods in general, so that a small perturbation in the value of the decision vector will result in a small change to the scheduled allocation. These are defined as follows (suppressing the *up* subscripts), as illustrated in Figure 1:

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 $t^s = V^s + \xi_1 (V^e - V^s)$ and $t^e = t^s + \xi_2 (V^e - t^s)$, respectively
3. Calculate the allocated antenna quantity (for array allocations) as $A(t) = A^{req}(t) + ceil(\xi_3 \Delta)$

For single antenna allocations only a pair of decision variables is required ξ_1, ξ_2 since the antenna allocated quantity is constant.

4. RESULTS

We have applied the evolutionary algorithm described in Section 3 to a variety of network scheduling scenarios, with very promising results. In the following we first describe a simple but illustrative conflict resolution scenario for the DSN, followed by an DSAN scheduling example for a much larger mission set and time range.

4.1. A Conflict Resolution Scenario

A frequent occurrence in today's DSN scheduling process is that two or more missions find themselves in conflict over the usage of a particular antenna in some time interval. Through a process of analysis, discussion, and negotiation, each such conflict must be resolved, generally by a tradeoff or compromise of one or another mission's requested allocation. To illustrate how our evolutionary multi-objective formulation of the problem can be of use in this situation, consider the following toy problem consisting of three mission users U1, U2, and U3, and three antennas A1, A2, and A3. We consider a one day schedule and use time units of fractional days for simplicity.

- U1 has view periods on A1 and A2 over the interval [0.1,0.6] and is initially scheduled on A1 in [0.2,0.6].
- U2 has a single view period on A1 from [0.4,0.9] and is initially scheduled there in [0.4,0.8].
- U3 has view periods on A2 and A3 over the interval [0.1,0.6] and is initially scheduled on A2 over [0.2,0.6].

All three users have specified a constraint that their allocation during the day have a minimum duration of 0.3. In addition, there is a system constraint that no resource be overloaded. The diagram in Figure 2(a) illustrates the situation: the initial schedule has a conflict between U1 and U2 on A1 in the interval [0.4,0.6], and each user has a preference to retain their original scheduled allocation. The view periods for each user are illustrated in Figure 2(b).

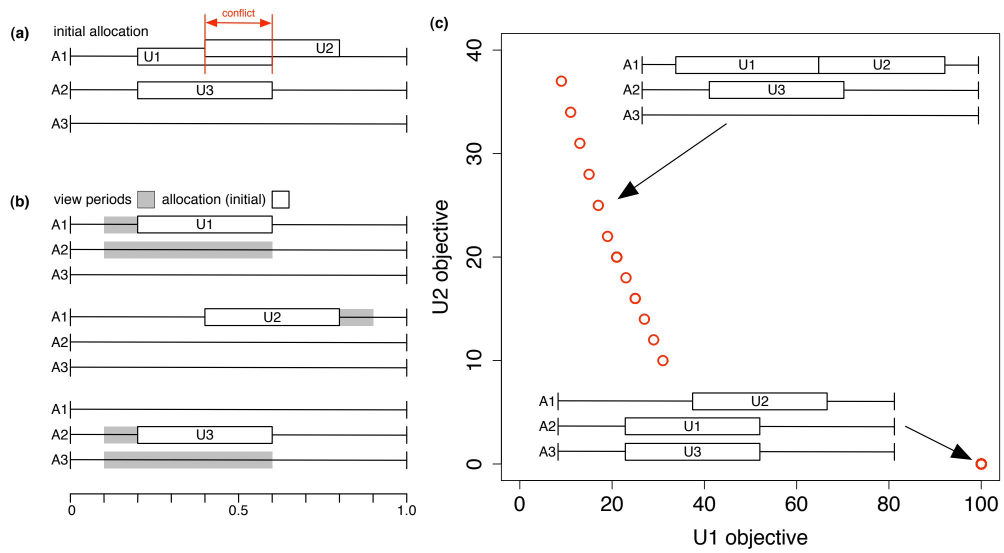

Figure 2. A conflict resolution example: (a) an initial schedule with conflict, (b) allowed antennas and view periods for the three users, and (c) Pareto frontier projected to U1-U2 plane, showing two families **of solution types.**

The most interesting aspect of the evolutionary algorithm solution is that the resulting rank 1 population covers several qualitatively different families of tradeoff possibilities (distinguished by jumps in allocation from one antenna to another), as illustrated in Figure 2(c) where we have plotted the projected U1 and U2 objective values on the Pareto frontier (after 300 generations, population size 300).

- Figure 2(c) upper: points along this curve represent tradeoffs between U1 and U2 (leaving U3 unaffected), such that both remain on the same antenna A1. The tradeoffs are in either the durations or start times of the U1 and U2 allocations, so they both fit within their available view periods.
- Figure 2(c) lower: this point represents a (projected) family of solutions in which U1 and U2 are allocated at their preferred times and durations, but U1 and U3 are shifted to their less preferred antennas A2 and A3, respectively. Variations in U3 all project to this point in this U1-U2 objective view.

This ability to reveal entire families of tradeoff possibilities, consisting of Pareto optimal candidate schedules, is a very powerful feature of this approach. In this particular example, Pareto-optimal candidate schedules form multiple distinct clusters: within each cluster, the schedules are qualitatively the same but differ in some quantitative way (e.g. there are minor shifts in allocation start time or duration). This makes it possible for clusters to be considered as a whole, while at the same time ensuring that each is Pareto optimal and thus superior in one or more objective values. It is worth pointing out that this automatic generation of solution clusters is not possible with conventional single-objective optimization methods.

4.2. A 2015 Projected Mission Set

To assess our approach on a larger scale problem we have applied it to a projected 2015 mission set, modeling a multi-site array of 12m X-band arrayed antennas. The main features of the problem are:

- Seventeen 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). We have made a variety of runs to evaluate the effect of population size and flexibility levels. The results are encouraging, in that the population evolves fairly quickly from infeasible to feasible candidates, then refines the Pareto frontier to develop a wide range of tradeoff alternatives. We have used this larger dataset to evaluate the scaling performance of the GDE3 approach. The results are shown in Figure 3, which shows time per generation scaling as $\alpha N \log N$ with population size N, and linearly with size of the time resolution interval into which the week is divided.

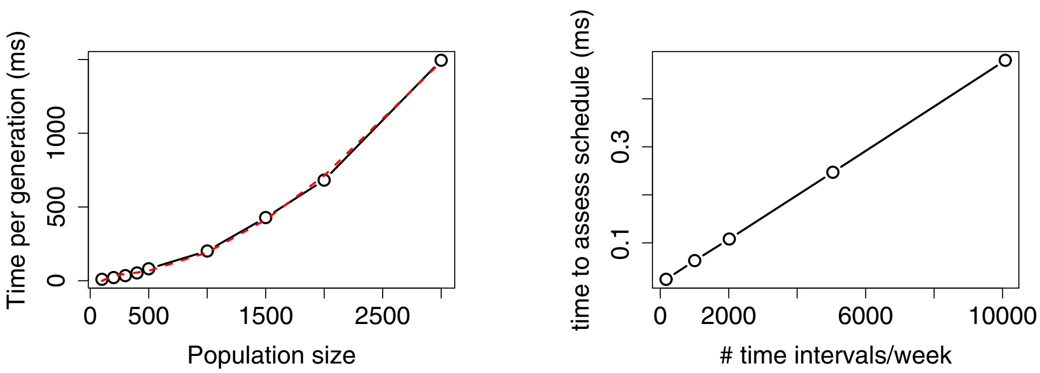

Figure 3. Scaling behavior on a 2015 projected mission set as a function of (left) population size N (α *N*log*N*) and (right) time resolution interval

5. CONCLUSIONS

Our results have shown that an evolutionary algorithm can be effectively applied to the intrinsically multi-objective scheduling problem of large scale space network communications scheduling, and presents several advantages over previous approaches, including:

- Explicit and separate representation of each mission's objectives (and potentially even sub-objectives), making it more straightforward to consider tradeoffs and compromises to resolve conflicts
- A population of Pareto-optimal solutions to use in schedule selection, as well as a starting point when revising the schedule when changes inevitably occur

There remain a number of areas to be investigated in the future:

- Highly constrained problems and their impact 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

- The effect of different parameter choices (*F* and *CR*) for the GDE3 algorithm
- The crowding distance calculation and whether it can be improved to enhance solution diversity: Kukkonen and Deb[17] have developed an improvement to the NSGA II definition, and have pointed out some problems with the calculation for higher dimensional objective spaces

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

1. D. S. Bagri and J. Statman. "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, 2004.
2. M. D. Johnston. "Multi-Objective Scheduling for NASA's Deep Space Network Array." International Workshop on Planning and Scheduling for Space (IWPSS-06). Baltimore, MD, Space Telescope Science Institute, 2006.
3. B. J. Clement and M. D. Johnston. "The Deep Space Network Scheduling Problem." IAAI 2005, Pittsburgh, PA, AAAI Press, 2005.
4. S. Chien, R. Lam, and Q. Vu. "Resource Scheduling for a Network of Communications Antennas." IEEE Aerospace Conference. Aspen, CO, 1997.
5. M. D. Johnston and B. J. Clement. "Automating Deep Space Network Scheduling and Conflict Resolution." ISAIRAS 2005, Munich, Germany, 2005.
6. L. Barbulescu, J.-P.Watson, L. D.Whitley, and A. E. Howe. "Scheduling Space-Ground Communications for the Air Force Satellite Control Network." Journal of Scheduling **7**(1): 7-34, 2004.
7. L. A. Kramer and S. F. Smith. "Maximizing Availability: A Commitment Heuristic for Oversubscribed Scheduling Problems." International Conference on Automated Planning and Scheduling (ICAPS 2005), Monterey, CA, AAAI, 2005.
8. K.-M. Cheung, C. H. Lee, and J. Ho. "Problem Formulation for Optimal Array Modeling and Planning." IEEEAC 438, 2005.
9. Y. Collette and P. Siarry. *Multiobjective Optimization*. Berlin, Springer, 2003.
10. T. P. Bagchi. *Multiobjective Scheduling by Genetic Algorithms*. Boston, Kluwer, 1999.
11. K. Deb. *Multi-Objective Optimization Using Evolutionary Algorithms*. New York, John Wiley & Sons, 2001.
12. N. K. Madavan. "Multiobjective Optimization Using a Pareto Differential Evolution Approach." 2002 Congress on Evolutionary Computation (CEC 2002). Honolulu, HI, 2002.
13. A. Abraham, L. Jain, and R. Goldberg. *Evolutionary Multiobjective Optimization*, Springer, 2005.
14. S. Kukkonen and J. Lampinen. "GDE3: The Third Evolution Step of Generalized Differential Evolution." The 2005 Congress on Evolutionary Computation, 2005.
15. K. Price, R. Storn, and J. Lampinen. *Differential Evolution: A Practical Approach to GlobalOptimization.* Berlin, Springer-Verlag, 2005.
16. K. Deb, A. Pratap, S. Agrawal, and T. Meyarivan. "A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II." IEEE Transactions on Evolutionary Computation 6(2): 182-197, 2002.

17. S. Kukkonen and K. Deb. "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, 2006.

ABOUT THE AUTHOR

M. D. Johnston is a principal scientist in the Artificial Intelligence Group at the Jet Propulsion Laboratory, California Institute of Technology, where he works on new planning and scheduling technologies for a number of space projects sponsored by NASA. His research interests include reactive planning and scheduling, multi-objective optimization, and evolutionary algorithms. He holds an A.B. degree summa cum laude from Princeton University, and a Ph. D. from MIT, both in Physics. He is the originator of the Spike scheduling system, developed for use on Hubble Space Telescope, and used on both the Chandra and Spitzer NASA Great Observatories as well as on various space- and ground-based observatories. His experience includes over 10 years in commercial product development of advanced planning and scheduling systems, especially as applied to semiconductor manufacturing, assembly, and testing.