Evolutionary Computation for the Identification of Emergent Behavior in Autonomous Systems

Richard J. Terrile
4800 Oak Grove Drive
Jet Propulsion Lab, California Institue of Technology
Mail Stop 301-330
Pasadena, CA 91109
818-354-6158
rich.terrile@jpl.nasa.gov

Alexandre Guillaume
4800 Oak Grove Drive
Jet Propulsion Lab, California Institue of Technology
Mail Stop 302-205
Pasadena, CA 91109
818-393-6899
alexandre.guillaume@jpl.nasa.gov

Abstract—Over the past several years the Center for Evolutionary Computation and Automated Design at the Jet Propulsion Laboratory has developed a technique based on Evolutionary Computational Methods (ECM) that allows for the automated optimization of complex computationally modeled systems. An important application of this technique is for the identification of emergent behaviors in autonomous systems. Mobility platforms such as rovers or airborne vehicles are now being designed with autonomous mission controllers that can find trajectories over a solution space that is larger than can reasonably be tested. It is critical to identify control behaviors that are not predicted and can have surprising results (both good and bad). These emergent behaviors need to be identified, characterized and either incorporated into or isolated from the acceptable range of control characteristics. We use cluster analysis of automatically retrieved solutions to identify isolated populations of solutions with divergent behaviors. [1] [2]

TABLE OF CONTENTS

1. INTRODUCTION 1
2. BACKGROUND 1
3. EVOLUTIONARY COMPUTATIONAL METHOD 2
4. TECHNICAL APPROACH 3
5. FUTURE WORK 5
6. CONCLUSIONS 5
ACKNOWLEDGEMENTS 5
REFERENCES 6
BIOGRAPHY 6

1. INTRODUCTION

We will describe a methodology for employing evolutionary computational methods (ECM) to address the problem of identifying and characterizing emergent behavior in complex and incompletely specified autonomous systems. These types of systems have many examples in the aerospace community with applications in robotic mobility and path finding. Unmanned land and air vehicles have complex controllers that can respond over a wide range of environmental variables that would otherwise require exhaustive testing and modeling in order to understand their modes of operation and operational safety requirements. There is a need to safely control intelligent, tele-operated or autonomous platforms and to predict autonomous performance in non-deterministic, emergent behavior, and unscripted modes of operation.

Autonomous systems are becoming increasingly capable of independent action in dynamic unpredictable environments, resulting in non-deterministic behaviors that may not have been apparent or anticipated by the tester or developer. Current capabilities and methodologies are insufficient to address testing of such complex systems operating in non-deterministic or unscripted modes. There is a need to predict how modifications may be produced by external stimuli or environmental changes, especially those that yield unacceptable and unpredictable outcomes that can be detrimental to the autonomous systems, humans, or test and evaluation systems. ECM techniques can effectively search the space of possible behaviors to identify and characterize emergent behaviors that are unexpected or detrimental.

2. BACKGROUND

Over the past several years the Center for Evolutionary Computation and Automated Design at the Jet Propulsion Laboratory has developed a technique based on Evolutionary Computational Methods (ECM) that allows for the automated optimization of complex computationally modeled systems [1-4]. An important application of this technique is for robotic mobility such as the optimization of robotic arm path finding or in terrain path finding [1, 5].

Evolutionary computation is a method that operates on a population of existing computational-based engineering models (or simulators) and competes their results using biologically inspired genetic operators on large parallel cluster computers. We have demonstrated that complex engineering and science models can be automatically inverted by incorporating them into evolutionary frameworks and that these inversions have advantages over conventional searches by not requiring expert starting guesses (designs) and by running on large cluster computers with less overall computational time than conventional approaches [1,2,4]. The result is the ability to automatically find design optimizations and trades, and thereby greatly amplify the role of the system engineer.

We have further applied these techniques to the automated retrieval of spectral signatures in data from terrestrial Earth-

1

[1] 978-1-4244-2622-5/09/$25.00 ©2009 IEEE.
[2] IEEEAC paper #1732, Version 5, Updated December 18, 2008

shine [6]. Part of this analysis was to characterize the degenerate behavior of automatically retrieved spectral solutions. We automatically found a population of synthetic spectral solutions that matched the observed data. We then determined how many of the solutions of synthetic spectra matched the real observational parameters associated with the data and if additional solutions exist in the population that match the observational spectrum, but use widely incorrect observational parameters. We used a cluster and principal components analysis to identify the populations of degenerate solutions [6,7].

These same evolutionary computational techniques for finding a population of solutions (in the case of trajectories for autonomous systems) and the clustering technique for characterizing the solution space can be applied to the identification of emergent behaviors. These methods have a proven record of cost effectively discovering the optimum performance characteristics of incompletely specified systems over a wide range of environmental variables that would otherwise require exhaustive testing and modeling in order to understand their modes of operation and operational safety requirements [1,2,4]. This paper describes the architectural methodology of employing evolutionary computational methods for finding the range of likely operational behaviors of a complex autonomous system, and then identifying out-lying behaviors by using principle component clustering methods.

3. EVOLUTIONARY COMPUTATIONAL METHOD

The primary technology, which enables the Evolutionary Computational Method (ECM) to find optimal solutions in complex search spaces, derives from evolutionary algorithms such as the genetic algorithm and differential evolution [8, 9]. These methods take inspiration from biological processes, in particular from genetics, to define an iterative process that evolves a population of parameter sets to the global optimum. We have demonstrated the success of these methods for a wide variety of applications where the range of operational control must be explored, tested and optimized (robotic arm, rover, spacecraft trajectory) [1, 10, 11, 12].

Evolutionary Computational Methods operate on computer models or simulators. An Autonomous Mission Controller (AMC) is one example of this, as is a Command and Control (C2) computer model. Generally, for a given goal these models will generate an outcome (or response) based on several inputs. For an AMC the outcome could be a trajectory or path of an autonomous system (rover), while the inputs would be terrain, obstacles, sensor inputs, etc. The goal might be to cross a field and arrive at a particular destination. ECM creates a metric based on how well the goal is achieved. This could be based on parameters such as minimized time, distance, fuel, visibility, data drop-outs, etc, or maximizing payload, terrain coverage, communication, etc. These goals are weighted according to their importance and make up a metric called fitness. In an optimized case we would be searching for the behavior of an autonomous system that maximizes the fitness. ECM uses biologically inspired techniques that mimic the action of evolution and development in a population of solutions over many generations. We use multiple copies of the AMC and run them on parallel processors. By using a collection of randomly varying inputs we generate a population of initial solutions (e.g. trajectories or paths). These solutions are individually evaluated for their fitness and competed against the entire set. The highest fitness individuals are used to create a next generation of inputs and are again put through the process of random variation and selection. This cycle is repeated until an optimum solution is discovered.

We have demonstrated that the ECM technique is extremely effective at quickly finding not only individual best solutions, but also an entire collection of best solutions for all possible ranges of conditions. The following are demonstrated features of applying ECM to a computer model (like an Autonomous Mission Controller):

- Best solutions are found automatically with very efficient use of computer time.
- No expert initial guesses or knowledge of the environment is required.
- Genetic searches are very opportunistic and will explore non-traditional uses of resources to accomplish a goal (maximize objective fitness).
- Optimum solutions are found for varying conditions, allowing the simple visualization of trade-offs for a range of requirements.
- ECM returned solutions provide a best-case standard that can be used to test against actual AMC performance in computer or field environments.
- Unexpected solutions can be identified and characterized as emergent behaviors.

The metric, or objective function against which we measure the performance of the autonomous system Autonomous Mission Controller, is essentially determined from the optimum path in a given environment and for a given goal requirement. This complex optimization process consists of exploring a vast set of possible paths in parallel to converge toward an optimum. We have successfully demonstrated that the Evolutionary Computational Method automatically performs this task with high efficiency on distributed computing platforms such as multi-core processors currently available on commercial workstations. An example of this path optimization is the application to robotic arm path finding [1]. Here, for the goal of finding the optimal collision-free and joint motion minimization path. We used the JPL FIDO rover control algorithms (both on a computational rover simulation environment and on the actual rover) to test for path fitness. Figure 1 shows an example of the multiple arms paths explored by ECM.

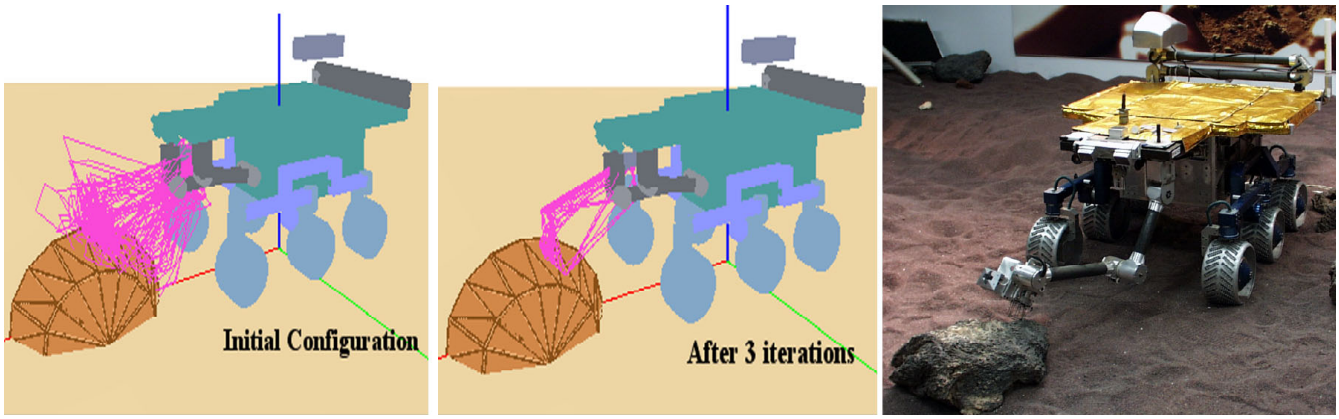

Figure 1: The left two images are from the ECM for robotic arm control. They are taken directly from the simulation environment that is used for testing the FIDO rover algorithms (same ECM object code running on rover as that in the simulation). As can be seen, after three iterations the populations of possible paths are converging to multiple methods. The algorithm takes less than one second to determine an optimal safe path. The right image shows an ECM arm deployment on the FIDO Rover.

4. Technical Approach

We used several of our ECM applications to demonstrate the ability to automatically generate multiple, high-fitness solutions to complex control problems [1]. The FIDO rover arm path finding application has demonstrated optimum path finding in a high degree of freedom environment. However, many of the ECM derived solutions will be high-fitness but, not necessarily, the optimum. To eliminate the sub-optimal solutions a set of non-dominated solutions are sorted that are the best solutions for a given set of input variables. Non-dominated solutions are used to create a set pareto-optimal solutions that define the frontier of best solutions.

For two dimensions of fitness (like propellant and flight time for the propulsion optimization problem), a line can be determined as the pareto-optimal front [11]. Occasionally, discontinuities exist in the front because of changes of control state. An example would be changing the integer number of orbits required to gain a specified velocity, or reducing the propellant mass so that the launch vehicle can be one discrete size smaller. Discretization of certain allowed states can cause discontinuities in the pareto-front. Identification of these discontinuities is difficult when high degrees of dimensions are involved in the fitness. These state changes are also important to identify emergent behaviors. For this reason we use principal component analysis to define solution clustering. The advantage of this technique is that the largest parameter differences are extracted first and deeper parameter searches can be easily performed. Figure 2 shows the principal component clustering of solutions for the spectral retrieval problem [6].

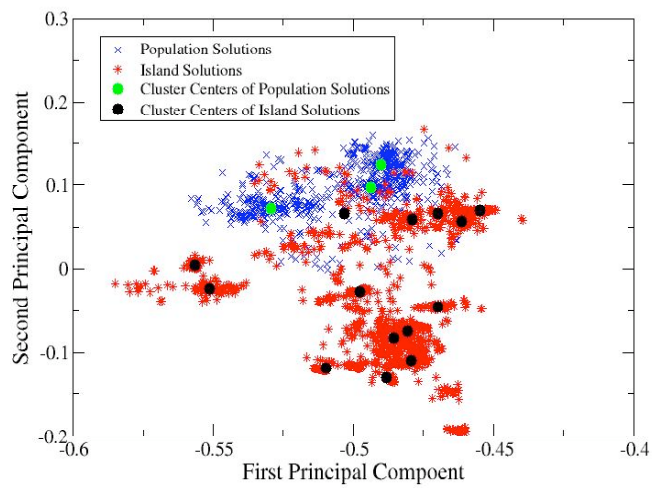

Figure 2: Principal component plots of spectral retrieval solutions satisfying defined fitness requirement. Multiple discrete clusters indicate wide variance of acceptable solutions.

Identification of emergent behaviors can be visualized using principal components analysis of the population of ECM derived solutions. Clustering of solutions indicates that they have similar properties. Outlying clusters indicate different paths to the same goal. Examination of the parameters (or components) of outlying clusters can determine if it is emergent behavior and characterize whether it has advantages or disadvantages to the system. For the application to automated retrieval of spectra, we demonstrated that outlying solution states can be characterized by their principal components.

For analysis and identification of emergent behavior in an autonomous mobility system we incorporate the control model, environmental model and goal requirements into an

evolutionary framework. For a given set of environmental parameters and goal requirements, an Autonomous Mission Controller (AMC) will determine the path or trajectory of an Unmanned Autonomous System (UAS) to the goal (Figure 3).

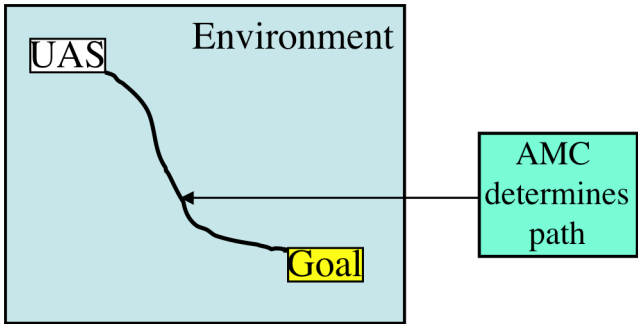

Figure 3: A UAS (Unmanned Autonomous System) path is determined by the performance of an AMC (Automated Mission Controller) within a simulated environment.

Differences in the tuning of the control parameters of the AMC, or changes in the environment or goals will give different paths. In order to test the actual performance of an AMC a metric is required that defines a certain level of anticipated performance based on achievable expectations. ECM optimization of paths, applied to an AMC in a simulated environment, will automatically return the best possible AMC performance for a given goal and environment. This optimum performance can then be used as a standard for testing the actual performance in either a simulated or real field environment. A schematic representation of ECM applied to AMCs is illustrated in Figure 4.

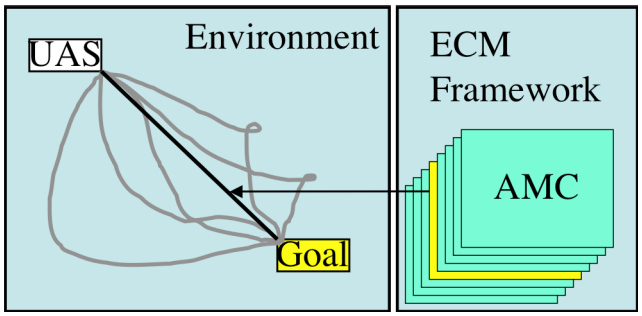

Figure 4: An ECM framework creates a population of AMCs with different tunable parameters.

For each ECM evaluation of an AMC, paths are determined and competed against the fitness of performing the goal. Genetic operators are used to generate a next generation of AMC paths and the process is repeated until a high degree of fitness is achieved. In this way a best path can be determined that can be used as a metric or standard for the best AMC performance. This metric can be used to compare performance of field tests and provides both the magnitude and direction of the deviation from the best expected performance (Figure 5).

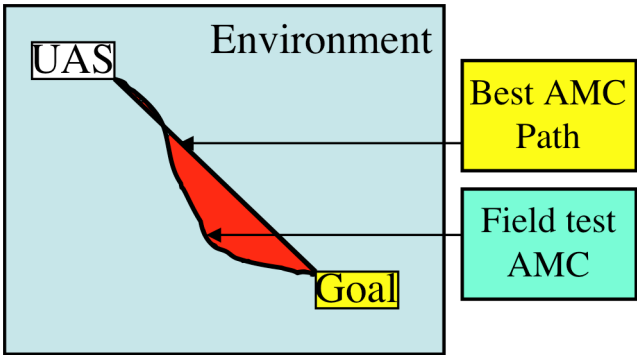

Figure 5: The best EMC determined performance becomes the standard metric for field and simulation testing.

The major advantage afforded by this ECM-based methodology is the ability to perform an automatic, unbiased search for all solutions within the entire parameter space, using criteria that make searching computationally far more economical than in complete-enumeration ("brute force"), Monte Carlo, or random searches. (As used here, "unbiased" characterizes a search that does not depend on initial *ad hoc* guesses by experts.)

For the behavior of a UAS, multiple paths and many solutions for the achievement of a goal are possible (involving multiple paths, different reactions to external stimulus, etc.). Occasionally, complex and unexpected behaviors can result from the unpredicted response to environmental conditions or untested programming states. It is vital to identify these emergent behaviors and to characterize their advantages or disadvantages. Since ECM returns a population of optimized solutions, we will characterize these solutions to identify unexpected behaviors. We have demonstrated techniques [6, 7] that can quickly identify emergent behaviors and characterize their effect on system parameters (see Figure 6).

In the example in Figure 6 we show a cluster of possible optimized paths (A) that utilize a bridge to cross an obstacle. For some ranges of AMC parameters (e.g. high velocity) or for some environmental conditions (e.g. narrow obstacle), there may be possible solution paths that jump the obstacle (B). It is important not only to identify that this emergent behavior is possible, but to also define the range of conditions that creates this behavior. Furthermore, the behavior must also be characterized as to whether it is an allowable advantage or a safety concern that should be avoided. If it is to be avoided, operational, environmental and control limits need to be defined and implemented.

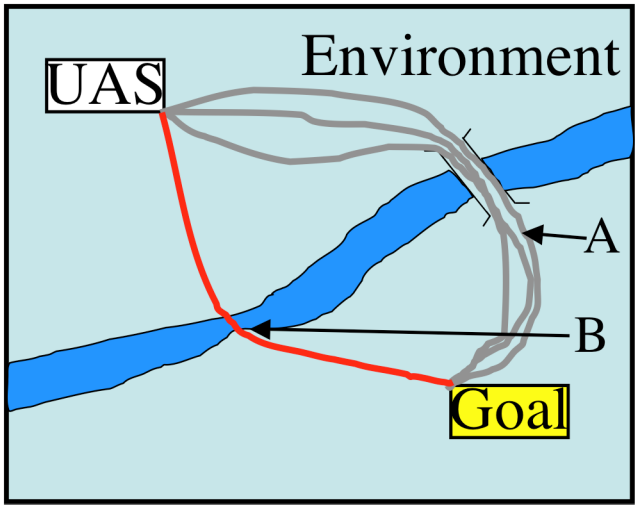

Figure 6: An ECM cluster analysis of populations of ACM created paths can identify outlying emergent behaviors.

5. Future Work

We plan to incorporate our Evolutionary Computational Method into the ROAMS (Rover Analysis, Modeling and Simulation) environment. This is a JPL facility that is an AMC simulation environment developed to test the Mars Exploration Rovers. Our initial work with the FIDO test rover should be directly applicable for the transition to this higher fidelity simulation and test environment. The higher dimensionality of ROAMS will have a larger range of possible emergent behaviors to explore.

6. Conclusions

Autonomous systems are becoming increasingly capable of independent action in dynamic unpredictable environments, resulting in non-deterministic behaviors that may not have been apparent or anticipated by the tester or developer. Current testing and evaluation capabilities and methodologies are insufficient to address testing of such complex systems operating in non-deterministic or unscripted modes. There is a need to predict how modifications may be produced by external stimuli or environmental changes, especially those that yield unacceptable and unpredictable outcomes that can be detrimental to the autonomous system, humans, or test and evaluation systems. ECM techniques can effectively search the space of possible behaviors to identify and characterize emergent behaviors that are unexpected or detrimental.

ECM provides a standard of performance that specifies the best that any AMC can achieve for any given set of requirements or goals. It does this for the range of all possible environmental variables in an efficient way. There are three important advantages of this ECM created standard for testing and the identification of emergent behavior:

- The ECM derived standard of performance can be directly compared to actual performance to provide a test standard of reference.
- Comparison of the ECM best solutions with actual performance will identify where the deviations from optimum occurred leading to localization of problem areas. This includes illuminating deficiencies in simulation capabilities.
- Examination of the full set of ECM derived best performance will enable the identification and characterization of unexpected or emergent behaviors.

Acknowledgements

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

References herein to any specific commercial product, process or service by trade name, trademark, manufacturer, or otherwise does not constitute or imply its endorsement by the United States Government or the Jet Propulsion Laboratory, California Institute of Technology.

REFERENCES

[1] Terrile, R. J., Adami, C., Aghazarian, H., Chau, S. N., Dang, V. T., Ferguson, M. I., Fink, W., Huntsberger, T. L., Klimeck, G., Kordon, M. A., Lee, S., von Allmen, P. A. and Xu, J. (2005) "Evolutionary Computation Technologies for Space Systems" IEEE Aerospace Conference Proceedings, Big Sky, MT, March 2005

[2] Terrile, R. J., Kordon, M., Mandutianu, D., Salcedo, J. and Wood, E. (2006) "Automated Design of Spacecraft Power Sub-Systems." IEEE Aerospace Conference Proceedings, Big Sky, MT, March 2006.

[3] Guillaume, A., Lee, S., Wang, Y-F,. Zheng, H., Hovden, R., Chau, S., Tung, T-W. and Terrile, R. J. (2007) "Deep Space Network Scheduling Using Evolutionary Computational Methods." IEEE Aerospace Conference Proceedings, Big Sky, MT, March 2007.

[4] Terrile, R. J., Kordon, M., Postma, M., Salcedo, J., Hanks, D. and Wood, E. (2007) "Automated Design of Spacecraft Telecommunication Subsystems Using Evolutionary Computational Techniques." IEEE Aerospace Conference Proceedings, Big Sky, MT, March 2007.

[5] Huntsberger, T., Aghazarian, H. and Tunstel, E. (2005) "Onboard Adaptive Learning for Planetary Surface Rover Control in Rough Terrain." Proceedings IEEE International Conference on Robotics and Automation, Barcelona, Spain, April 2005.

[6] Terrile, R. J., Lee, S., Tinetti, G., Fink, W., von Allmen, P. and Huntsberger, T. L. (2008) "Evolutionary Computational Methods for the Design of Spectral Instruments." IEEE Aerospace Conference Proceedings, Big Sky, MT, March 2008.

[7] Guillaume, A., Lee, S., Braverman, A. and Terrile, R. (2008) "Entropy Constrained Clustering Algorithm Guided by Differential Evolution." IEEE Aerospace Conference Proceedings, Big Sky, MT, March 2008.

[8] Holland, J. H. (1975) "Adaptation in Natural and Artificial Systems." The University of Michigan Press, Ann Arbor, Michigan.

[9] Price, K. V., Storn, R. M., Laminen, J. A. (2005) "Differential Evolution: A Practical Approach to Global Optimization." Springer Press, New York.

[10] Lee, S., von Allmen, P., Fink, W., Petropoulos, A. E. and Terrile, R. J. (2005) "Design and Optimization of Low-Thrust Orbit Transfers Using the Q-law and Evolutionary Algorithms." IEEE Aerospace Conference Proceedings, Big Sky, MT, March 2005.

[11] Lee, S., von Allmen, P., Fink, W., Petropoulos, A. E. and Terrile, R. J. (2005) "Comparison of Multi-Objective Genetic Algorithms in Optimizing Q-law Low-Thrust Orbit Transfers." GECCO Conference Proceedings, Washington, DC, June 2005.

[12] Lee, S., Russell, R. P., Fink, W., Terrile, R. J., Petropoulos and von Allmen, P. (2006) "Low-Thrust Mission Trade Studies with Parallel, Evolutionary Computing." IEEE Aerospace Conference Proceedings, Big Sky, MT, March 2006.

BIOGRAPHY

Richard J. Terrile *created and directs the Center for Evolutionary Computation and Automated Design at NASA's Jet Propulsion Laboratory. His group has developed genetic algorithm based tools to improve on human design of space systems and has demonstrated that computer aided design tools can also be used for automated innovation and design of complex systems. He is a planetary astronomer and the co-discoverer of the Beta Pictoris circumstellar disk. Dr. Terrile has B.S. degrees in Physics and Astronomy from the State University of New York at Stony Brook and an M.S. and a Ph.D. in Planetary Science from the California Institute of Technology in 1978.*

Alexandre Guillaume *is a member of the information systems and computer science staff at NASA's Jet Propulsion Laboratory. His recent work includes genetic algorithm application, quantum computing theory, experiments on superconducting devices and materials modeling. His work has been published in peer reviewed journals. He has a Ph.D. in Physics from the Joseph Fourier University in Grenoble (France).*