Multidisciplinary Design and Control Optimization of a Spherical Robot for Planetary Exploration

Article *in* Collection of Technical Papers - AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference · December 2019

CITATIONS
0

READS
92

2 authors:

Himangshu Kalita
The University of Arizona
53 PUBLICATIONS **167** CITATIONS
SEE PROFILE

Jekan Thangavelautham
The University of Arizona
205 PUBLICATIONS **834** CITATIONS
SEE PROFILE

Multidisciplinary Design and Control Optimization of a Spherical Robot for Planetary Exploration

Himangshu Kalita[1] and Jekan Thangavelautham[2]
Space and Terrestrial Robotic Exploration (SpaceTREx) Laboratory,
Aerospace and Mechanical Engineering Department, University of Arizona, 85721, USA

Missions targeting extreme and rugged environments such as caves, canyons, cliffs and crater rims of the Moon, Mars and icy moons are the next frontiers in solar system exploration. Exploring these sites will help ascertain the range of conditions that can support life and identify planetary processes that are responsible for generating and sustaining habitable worlds. Current landers and rovers are unable to access these areas of high interest due to limitations in precision landing techniques, need for large and sophisticated science instruments and a mission assurance and operations culture where risks are minimized at all costs. This research proposes using multiple spherical robots called SphereX for exploring these extreme environments. The design of SphereX is a complex task that involves a large number of design variables and multiple engineering disciplines. The methodology developed in this work uses Automated Multidisciplinary Design and Control Optimization (AMDCO) techniques to find near optimal design solutions in terms of mass, volume, power and control for SphereX for different mission scenarios. The implementation of AMDCO for SphereX design is a complex process because of complexity of modelling and implementation, discontinuities in the design space, and wide range of time scales and exploration objectives. We address these issues by using machine learning in the form of Evolutionary Algorithms integrated with gradient-based optimization techniques to search through the design space and find pareto optimal solutions for a given mission task. The design space is searched using a GA multi-objective optimizer at the system (global) level to find the Pareto-optimal results while gradient-based techniques are used to search at the discipline (local) level. The modeled disciplines are mobility system, power system, thermal system, shielding, communication system, avionics and shell. Using this technology, it is now possible to perform end to end automated preliminary design of planetary robots for surface exploration.

I. Nomenclature

m	=	mass of the robot
r	=	radius of the robot
P	=	power demand of the robot
$f, \mathbb{F}$	=	objective function
$g, \mathbb{G}$	=	inequality constraints
$h, \mathbb{H}$	=	equality constraints
$d, \mathbb{x}$	=	design variables
λ, μ	=	lagrange multiplier
$\mathcal{J}$	=	cost function

[1] PhD Student, Aerospace and Mechanical Engineering, University of Arizona.
[2] Assistant Professor, Aerospace and Mechanical Engineering, University of Arizona.

II. Introduction

We aspire to send human and robotic explorers to every corner of our solar system to perform orbital, surface and even subsurface exploration in the next few decades. These explorers will pave the way towards identifying the diverse surface environments, physical processes and structure of the planets and small bodies answering fundamental questions about the origins of the solar system, conditions to sustain life and prospects for resource utilization and off-world human settlement. Achieving this major exploration milestone remains technologically daunting but not impossible. An emerging target are the extreme environments of the Moon, Mars and icy moons, including caves, canyons, cliffs, skylights and craters as shown in Fig. 1. These are high-priority targets as outlined in the Planetary Science Decadal survey [1]. These environments are rich targets of origin studies, as the rocks and surfaces carved out from natural processes expose a time-record of ancient events including changing climate and surface composition, violent impacts/collisions events and evidence for organic chemicals. Caves offer natural shelter from radiation, harsh surface processes such as dust storms and are generally insulated by the varying high and low external temperatures. These conditions could harbor isolated, ancient ecosystems.

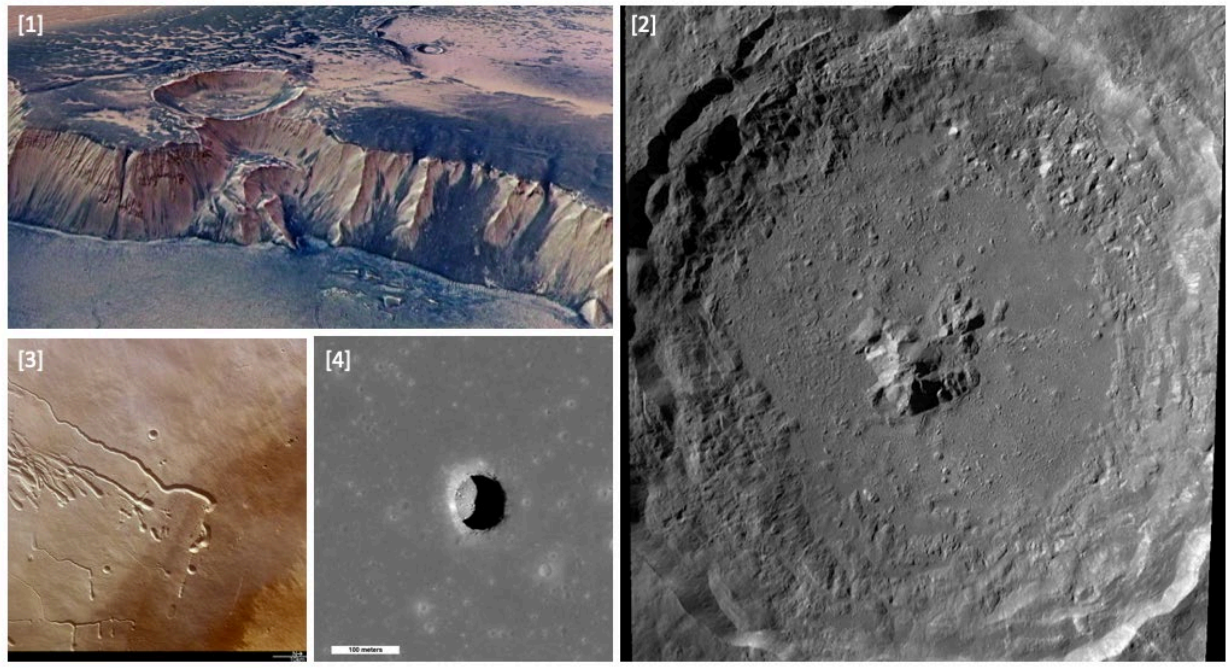

Fig. 1 Extreme environments of the Moon and Mars: (1) High cliffs surrounding Echus Chasma on Mars (nasa.gov), (2) Tycho crater on Moon (NASA/Goddard/Arizona State University), (3) Lava tubes on Pavonis Mons on Mars (ESA), and (4) Mare Tranquilitatis pit on Moon (NASA/GSFC/Arizona State University).

High resolution orbital imagery from the Lunar Reconnaissance Orbiter Camera (LROC) reveal evidence for subsurface voids and mare-pits on the lunar surface [2, 3]. Similar discoveries have been made with the HiRise camera onboard the Mars Reconnaissance Orbiter (MRO) observing the Martian surface. An example pit on the moon is Mare Tranquilitatis pit (MTP; 8.335°N, 33.222°E) which opens into a sub-lunarean void at least 20 meters in extent. The pit diameters range from 86 to 100m with a maximum depth from shadow measures of ~107m. Several large, angular blocks are sparsely distributed across the floor, and likely represent detritus from the pit walls or collapsed roof materials which makes it impassable by conventional wheeled robots. Accessible voids could be used for a future human base because they offer a natural radiation and micrometeorite shield and offer constant habitable temperatures of -20 to -30°C [4].

Exploration of these extreme and rugged environments remains out of reach from current planetary rovers and landers; however, the 2015 NASA Technology Roadmaps prioritizes the need for next-generation robotic and autonomous systems that can explore these extreme and rugged environment [5]. The challenges are three-fold and stem from current landing technology that requires wide-open spaces with no obstacles or landing hazards. A second challenge stems from current planetary vehicle architectures. Planetary rovers and landers ever since Luna 9, the first mission to soft-land on the Moon, have been generally growing in size and capability to house a growing variety of sophisticated science instruments. A third challenge has been the high standards of mission assurance expected. Due

to the high costs and prestige for the nations involved, any form of exploratory risk that may reduce the life of the mission or result in damage to one or more subsystems is avoided. This is despite the potential science rewards from taking these exploratory risks.

A credible solution is to develop an architecture that permits taking high exploratory risks that translates into high reward science but without compromising the rest of the mission. Rapid advancement in electronics, sensors, actuators and power supplies have resulted in ever-shrinking devices and instruments that can be housed in small platforms. This has resulted in the wide adoption of CubeSats for Low Earth Orbit (LEO) missions and technology demonstrations. CubeSats are emerging as platform for performing high-risk, high-reward interplanetary exploration [6]. The technology uses Commercial-Off-The-Shelf (COTS) technology with adaptations to the space environment. Further technological advancement is leading towards radiation hardened version of these components for use in deep-space and planetary environments. However, these technologies still need to be proven in these planetary environments.

We present an architecture of small, low-cost, modular spherical robot called SphereX that is designed for exploring extreme environments like caves, lava tubes, pits and canyons in low-gravity environments like the Moon, Mars, icy moons and asteroids as shown in Fig. 2 [40-42]. It consists of a mobility system to perform optimal exploration of these target environments. It also consists of space-grade electronics like computer board for command and data handling, power board for power management and radio transceiver for communicating among multiple robots. Moreover, it also consists of a power system for power generation/storage, multiple UHF/S-band antennas and accommodates payloads in the rest of the volume. A large rover or lander may carry several of these SphereX robots that can be tactically deployed to explore and access rugged environments inaccessible by it.

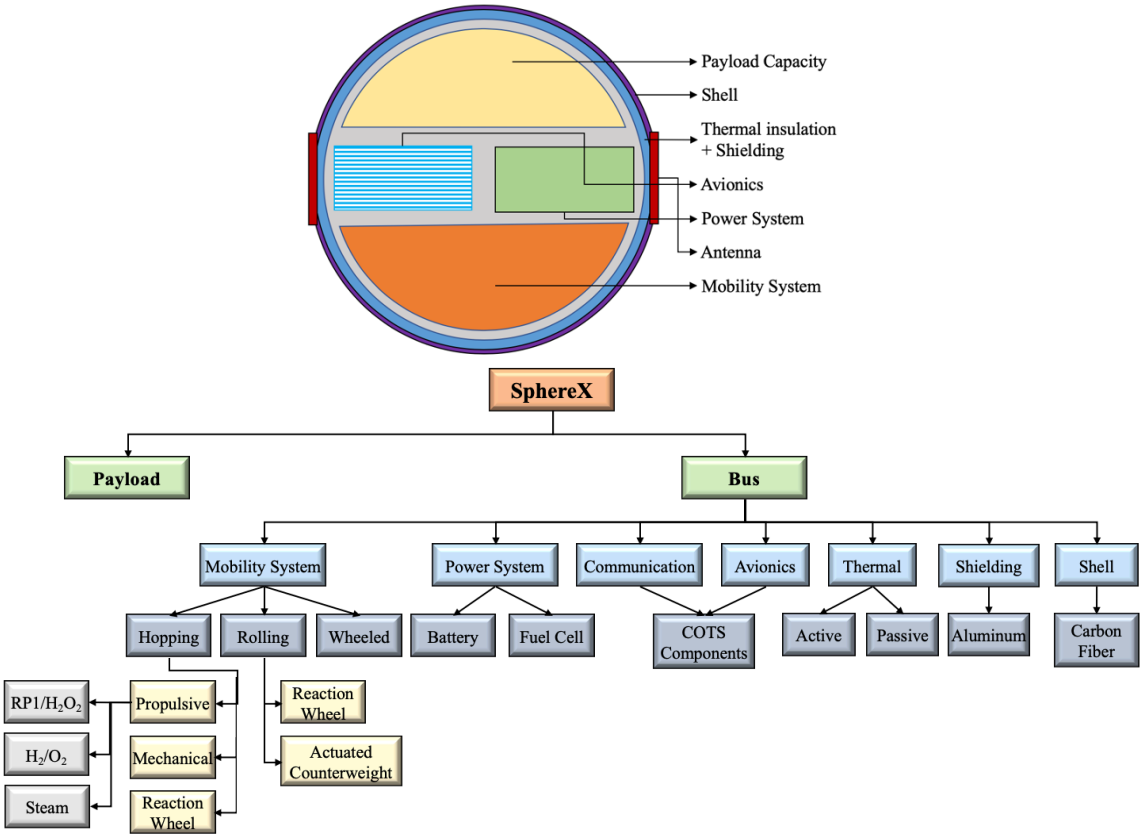

Fig. 2 (Top) SphereX architecture, (Bottom) Available options for each subsystem of SphereX.

However, the design of SphereX is a complex task that involves a large number of variables and multiple engineering disciplines. It is a highly coupled problem between multiple disciplines (Fig. 2(Bottom)), and it must balance payload objectives against its overall size, mass, power and control which affects its cost and operation. Moreover, each subsystem has multiple candidate solutions, e.g. mobility can be achieved through hopping, rolling or wheels, power system can be design through batteries that carries all the required power or can be generated on demand through fuel cells. Similarly, the selection of communication system and the avionics depends on numerous COTS

options available, the thermal system can be designed through active, passive or a combination of both. As such finding optimal design solutions for SphereX to meet a defined mission requirement is of paramount importance. Currently, space systems are optimized manually through evaluation of each discipline independently. With this labor-intensive approach, although feasibility is achieved, there is no guarantee for achieving optimality of the overall system. Thus, space system design could benefit from the application of multidisciplinary design optimization (MDO). However, complexity arises in an MDO approach due to complexity of modeling, complexity and discontinuity in objective cost function and design space, and wide range of time scales and mission requirements. Here, we approach this problem by using a hybrid optimization process where the search of the design space is performed with a GA based multi-objective optimizer at the system level to find the Pareto-optimal results while using gradient-based techniques at the discipline level. The methodology developed in this work uses Automated Multidisciplinary Design and Control Optimization (AMDCO) techniques to find near optimal design solutions in terms of mass, volume, power and control for SphereX for different mission scenarios. Using this technology, it is now possible to perform end to end automated preliminary design of planetary robots for extreme environment exploration.

For implementation perspective, the large number of disciplines of SphereX presents a significant challenge as they are coupled together. Fig. 3 shows the different disciplines of SphereX and how they are coupled together. The mission specifications and environment model affect design decisions of multiple disciplines of SphereX. For e.g. target distance affects the design of the mobility system, target mission time affects the power system and multiple number of robots introduces complexity in the communication system. Moreover, gravitational and surface properties models affect the design of the mobility system, radiation and temperature models affect the design of the thermal and shielding subsystems. Similarly, the mass and volume of each subsystem affect the design of the mobility system, and power requirement of each subsystem affect the design of the power system which in turn increase/decrease the mass and size of the power system affecting the mobility system. Furthermore, to increase payload volume, if we increase the size of the shell of the robot, its mass and inertia increases in the order of $\mathcal{O}(r^2)$, r being the radius of the shell, thus affecting the mobility system. As such finding optimal design for each subsystem taking inter-subsystem dependencies into account is of paramount importance.

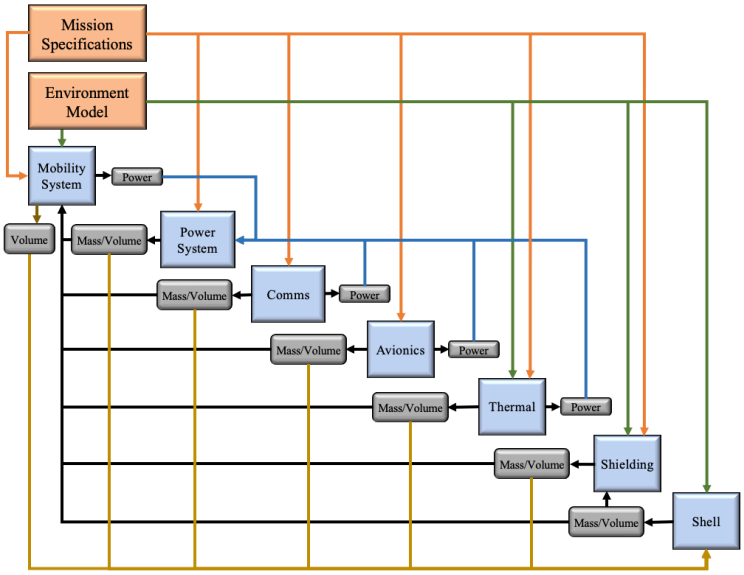

Fig. 3 Design structure matrix of relevant disciplines of SphereX.

III. Background Study

In this section, we review the literature that describes how optimization technology has been applied to the space system design and control process. The related research can be categorized into three groups. The first group approaches the space system design problem from the perspective of a single discipline, the second group employs multidisciplinary approaches and the third group unifies operation and control optimization with the design optimization process.

Single Discipline Optimization (SDO) is the analysis and optimization of an engineering system with focus on a single discipline of a system. Detailed and complex modeling of system discipline often includes discrete or indifferentiable design variables that cannot be handled with gradient-based optimization but can be handled with gradient-free optimization techniques such as Genetic Algorithm (GA) or Particle Swarm Optimization (PSO). Hence, many single optimization researches focus on gradient-free optimization. Zhang et al. optimized the layout of the satellite subsystems with the integrated GA/PSO and Quasi-Principal Component Analysis (QPCA) [7]. Richie et al. used reduced-order, gradient-based solver to optimize the size of a miniature control moment gyroscope for a practical space mission. By analyzing attitude and energy storage requirements of a small satellite, their design problem can be converted to a constrained nonlinear programming problem which is a rare example of gradient-based optimizer

implementation in SDO [8]. Jain et al. optimized the power management of a small satellite using GA, which can be evaluated on Field-Programmable Gate Array (FPGA) in real time [9]. Hull et al. solved the spacecraft radiator design problem with GA [10]. Boudjemai et al. designed small satellite structural topologies with minimum compliance, minimum weight, and optimal compliant mechanisms with Finite Element Analysis (FEA) and GA [11]. The review result confirms that most of SDO literature have successfully implemented GA and other non-gradient based optimization methods.

Multidisciplinary design optimization (MDO) is a rapidly growing field with applications to a variety of aerospace problems and many aerospace system designers have applied MDO approaches to their system design projects. However, most MDO efforts have focused on the design of aircraft structures and space launch vehicles, and very little work has considered the MDO application to space systems with complex constrains such as small satellites and planetary exploration robots. The first identified MDO application to space systems was a launch vehicle design done by Olds et al [12]. Others have applied MDO to launch vehicles as well where the application of MDO was extended to space system engineering [13-16]. Satellite design with MDO was first applied by Matossian to the design of an Earth-observing satellite mission [17]. Spacecraft design-related research was continued by Mosher [18], Riddle [19], and Bearden [20], who focused on the development of space system engineering tools. Increased complexity of the design problem necessitates an organized and structured solving procedure and system engineering tool makes the evaluation of the optimization results easy by including various models and their simulation environments. George et al. [21] developed the Multidisciplinary Integrated Design Assistant for Spacecraft (MIDAS), which is a graphical programming language specially for space mission design. Fukunaga et al. [22] built the Optimization ASsIStant (OASIS), a system tool that can provide the optimized spacecraft using a set of generic, metaheuristic optimization algorithms (e.g., GA, simulated annealing), which are configured for a particular optimization problem by an adaptive problem solver based on artificial intelligence and machine learning techniques. Mosher et al. [23, 24] suggested a tool for conceptual spacecraft design, Spacecraft Concept Optimization and Utility Tool (SCOUT), that uses a set of design-estimating and cost-estimating relationships that are coupled with genetic algorithm optimization. Stump et al. [25] developed the Advanced Trade Space Visualizer (ATSV) that facilitates design by using a shopping paradigm to support trade space exploration. Barnhart et al. [26] implemented a Spacecraft Portal for Integrated Design in Real-time project (SPIDR), a systems-engineering-based framework for satellite design with an artificial-intelligence-based optimization algorithm that incorporates user-defined rules and constraints. Ravanbakhsh et al. [27] introduced a structural design-sizing tool containing the primary structures properties and system level variables. Apart from the above system engineering tool developments, many researchers have developed novel MDO methodology for space systems. Taylor et al. [28] provided an evaluation of optimization techniques which are applied to increasingly complex spacecraft design problems. Jilla et al. [29] developed the constructing process of multi-objective, multidisciplinary design optimization systems. Jafarsalehi et al. [30] focused on the development of an efficient distributed Collaborative Optimization (CO) method for small satellite missions. GA was used at the system level while gradient-based techniques were utilized at the discipline level. Form the MDO literature, we can see that many researchers developed system engineering tools that considers multiple disciplines by applying mathematical optimization techniques to improve space system designs.

Unifying operation and control optimization with the design optimization process is another approach to design space systems. Spangelo et al. [31] developed models and algorithms for solving single-satellite, multi-ground station communication scheduling problems, with the objective of maximizing the total amount of data downloaded from space. The most interesting aspect of their research is that they included the ground station operation over time in the optimization problem, making it more challenging. Hwang et al. [32] applied a new mathematical framework for gradient-based multidisciplinary optimization that automatically computes the coupled derivatives of the multidisciplinary system via a generalized form of the adjoint method to unify the design and the operation of a satellite.

Motivated by these ideas, in this research, the design and control of SphereX is unified in our MDO problem. The solutions of the MDO problem are then used for controller robustness analysis towards uncertainties and unmodeled dynamics and optimal path planning in a target environment.

IV. Approach

Based on the literature review, a multidisciplinary design and control optimization-based approach is proposed to explore the question of how to maximize the payload mass, volume and power budget while minimizing the total mass, volume and power of SphereX. The problem is approached by using a hybrid optimization process where the search of the design space is performed with a multi-objective optimizer at the system level to find the Pareto-optimal

results while using gradient-based techniques at the discipline level as shown is Fig. 4 [37]. At the system level, the multi-objective optimization problem is formulated as Eq. (1).

$$\min \mathbb{F}_k(\mathbb{x}) \quad k = 1,2,\dots,K$$
$$subject\ to \begin{cases} \mathbb{G}_l(\mathbb{x}) \le 0 & l = 1,2,\dots,L \\ \mathbb{H}_m(\mathbb{x}) = 0 & m = 1,2,\dots,M \\ \mathbb{x}_j^{(L)} \le \mathbb{x}_j \le \mathbb{x}_j^{(U)} & j = 1,2,\dots,J \end{cases} \tag{1}$$

The solution $\mathbb{x}$ is a vector of J system level design variables: $\mathbb{x} = [\mathbb{x}_1, \mathbb{x}_2, \dots, \mathbb{x}_J]^T$. There are K objective functions $\mathbb{F} = [\mathbb{F}_1, \mathbb{F}_2, \dots, \mathbb{F}_K]^T$. Associated with the problem are L inequality constraints and M equality constraints. The last set of J constraints are the variable bounds, restricting each decision variable $\mathbb{x}_j$ to take a value within a lower $\mathbb{x}_j^{(L)}$ and an upper $\mathbb{x}_j^{(U)}$ bound.

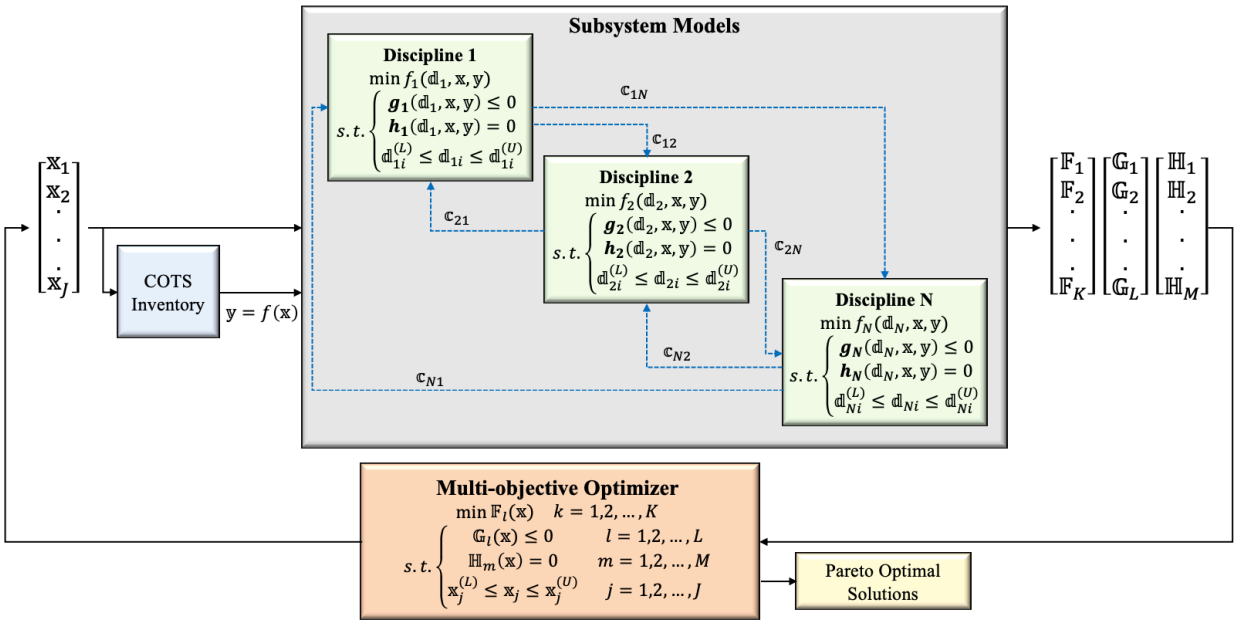

Fig. 4 Hybrid optimization approach for multidisciplinary optimization.

Each of the subsystem discipline are modeled as a single-objective optimization problem and is formulated as Eq. (2).

$$for\ each\ subsystem, i = 1,2,\dots,N \begin{cases} \min f_i(\mathbb{d}_i) \\ subject\ to \begin{cases} g_{ip}(\mathbb{d}_i) \le 0 & p = 1,2,\dots,P_i \\ h_{iq}(\mathbb{d}_i) = 0 & q = 1,2,\dots,Q_i \\ \mathbb{d}_{ir}^{(L)} \le \mathbb{d}_{ir} \le \mathbb{d}_{ir}^{(U)} & r = 1,2,\dots,R_i \end{cases} \end{cases} \tag{2}$$

For each subsystem discipline i, $\mathbb{d}_i$ is a vector of R_i discipline level design variables: $\mathbb{d}_i = [\mathbb{d}_{i1}, \mathbb{d}_{i2}, \dots, \mathbb{d}_{iR_i}]^T$. Each discipline has P_i inequality constraints, Q_i equality constraints, and R_i constraints that restricts the design variables within the upper and lower bounds. The vector $\mathbb{y}$ represents the specifications of the COTS components selected for each subsystem based on the inventory list and the system level design variables $\mathbb{x}$. Moreover, $\mathbb{c}_{ij}, \forall i \in \{1,\dots,N\}, \forall j \in \{1,\dots,N\}, i \ne j$ are the coupling functions calculated by discipline i and input to discipline j.

V. Methods

In order to find optimal design solutions of SphereX, multiple subsystem discipline level single-objective optimization problems, and a system level multi-objective optimization problem are to be solved simultaneously. This section provides the methods used in this research to solve the single-objective and multi-objective optimization problems.

A. Single-objective Optimization

As discussed above, the subsystem discipline models are modeled as single-objective optimization problem with one objective function, P inequality constraints, Q equality constraints and R side constraints. The R side constraints are converted into $2R$ inequality constraints such that there are $S = P + 2R$ inequality constraints, thus the problem is modeled as a nonlinear optimization problem (NLP) of the form shown in Eq. (3).

$$\min_{\mathbb{d}\in\mathbb{R}} f(\mathbb{d}) \quad subject\ to \begin{cases} g(\mathbb{d}) \le 0 \\ h(\mathbb{d}) = 0 \end{cases} \tag{3}$$

Where, $f: \mathbb{R}^R \to \mathbb{R}$ is the objective function, the function $g: \mathbb{R}^R \to \mathbb{R}^S$ and $h: \mathbb{R}^R \to \mathbb{R}^Q$ are the inequality and equality constraints. For scalar-valued function f, the gradient is denoted by $\nabla f(\mathbb{d})$ and the Hessian $Hf(\mathbb{d})$ by the matrix of second partial derivative as Eq. (4) and (5) respectively.

$$\nabla f(\mathbb{d}) = \left(\frac{\partial f(\mathbb{d})}{\partial \mathbb{d}_1}, \frac{\partial f(\mathbb{d})}{\partial \mathbb{d}_2}, \dots, \frac{\partial f(\mathbb{d})}{\partial \mathbb{d}_R}\right)^T \tag{4}$$

$$\left(Hf(\mathbb{d})\right)_{ij} = \frac{\partial^2 f(\mathbb{d})}{\partial \mathbb{d}_i \partial \mathbb{d}_j}, \quad 1 \le i, j \le R \tag{5}$$

For vector-valued functions, ∇ is used to denote the Jacobian of the function as in Eq. (6).

$$\nabla h(\mathbb{d}) = \left(\nabla h_1(\mathbb{d}), \nabla h_2(\mathbb{d}), \dots, \nabla h_Q(\mathbb{d})\right) \tag{6}$$

The scalar-valued function $\mathcal{L}: \mathbb{R}^{R\times S\times Q} \to \mathbb{R}$ defined by Eq. (7)

$$\mathcal{L}(\mathbb{d}, \lambda, \mu) = f(\mathbb{d}) + \lambda^T h(\mathbb{d}) + \mu^T g(\mathbb{d}) \tag{7}$$

is called the Lagrangian function of the NLP. The vectors $\lambda \in \mathbb{R}^Q$ and $\mu \in \mathbb{R}^S$ are the Lagrange multiplier vectors. Given a vector $\mathbb{d}$, the set of active constraints at $\mathbb{d}$ consists of the inequality constraints $g^{\blacksquare}(\mathbb{d})$, if any, satisfied as equalities at $\mathbb{d}$. The index set of active constraints are denoted by, $\mathcal{I}(\mathbb{d}) = \{i: g_i(\mathbb{d}) = 0\}$. Setting $S_{\mathbb{d}} = |\mathcal{I}(\mathbb{d})|$ and assuming $\mathcal{I}(\mathbb{d}) = \{i_1, i_2, \dots, i_{S_{\mathbb{d}}}\}$, the matrix $G(\mathbb{d}) \in \mathbb{R}^{R\times(Q+S_{\mathbb{d}})}$ is made up of the matrix $\nabla h(\mathbb{d})$ along with the columns $\nabla g_i(\mathbb{d}), i \in \mathcal{I}(\mathbb{d})$ as in Eq. (8).

$$G(\mathbb{d}) = \left(\nabla h_1(\mathbb{d}), \nabla h_2(\mathbb{d}), \dots, \nabla h_Q(\mathbb{d}), \nabla g_{i_1}(\mathbb{d}), \dots, \nabla g_{i_{S_{\mathbb{d}}}}(\mathbb{d})\right) \tag{8}$$

For $\mathbb{d}^* \in \mathbb{R}^R$ to be an isolated local minimum of the NLP, the following conditions known as the Karush-Kuhn-Tucker (KKT) conditions should apply [33]:

(A1): the first order necessary conditions hold, i.e., there exist optimal Lagrange multiplier vectors λ^* and $\mu^* \ge 0$ such that

$$\nabla\mathcal{L}(\mathbb{d}^*, \lambda^*, \mu^*) = \nabla f(\mathbb{d}^*) + \nabla h(\mathbb{d}^*)\lambda^* + \nabla g(\mathbb{d}^*)\mu^* = 0$$

(A2): The columns of $G(\mathbb{d}^*)$ are linearly independent

(A3): Strict complementary slackness holds, i.e.,

$$g_i(\mathbb{d}^*)\mu_i^* = 0$$

for $i = 1, \dots, S$ and if $g_i(\mathbb{d}^*) = 0$, then $\mu_i^* > 0$.

(A4): The Hessian of the Lagrangian function with respect to $\mathbb{d}$ is positive definite on the null space of $G(\mathbb{d}^*)^T$; i.e.,

$$a^T H\mathcal{L}(\mathbb{d}^*, \lambda^*, \mu^*)a > 0$$

For all $a \ne 0$ such that $G(\mathbb{d}^*)^T a = 0$

To solve the NLP problem, The Sequential Quadratic Programming (SQP) method is used which is an iterative method in which, at a current iterate $\mathbb{d}^k$, the step to the next iterate is obtained through information generated by solving a quadratic subproblem. The local convergence of the SQP method follows from the application of Newton's method to the nonlinear system given by the KKT conditions as shown in Eq. (9).

$$\Psi(\mathbb{d}, \lambda, \mu) = \begin{bmatrix} \nabla\mathcal{L}(\mathbb{d}, \lambda, \mu) \\ h(\mathbb{d}) \\ g^{\blacksquare}(\mathbb{d}) \end{bmatrix} = 0 \tag{9}$$

The Jacobian of the nonlinear system is given by Eq. (10).

$$J(\mathbb{d}, \lambda, \mu) = \begin{bmatrix} H\mathcal{L}(\mathbb{d}, \lambda, \mu) & \nabla h(\mathbb{d}) & \nabla g^{\blacksquare}(\mathbb{d}) \\ \nabla h(\mathbb{d}) & 0 & 0 \\ \nabla g^{\blacksquare}(\mathbb{d}) & 0 & 0 \end{bmatrix} \tag{10}$$

The conditions (A2) and (A4) imply that the Jacobian is nonsingular at a local solution. Therefore, the Newton iteration is given by Eq. (11).

$$\begin{aligned} \mathbb{d}^{k+1} &= \mathbb{d}^k + s_{\mathbb{d}} \\ \lambda^{k+1} &= \lambda^k + s_\lambda \end{aligned} \tag{11}$$

$$\mu^{k+1} = \mu^k + s_\mu$$

where, $s = (s_{\mathbb{d}}, s_\lambda, s_\mu)$ is the solution of Eq. (12).

$$J(\mathbb{d}^k, \lambda^k, \mu^k)s = -\Psi(\mathbb{d}^k, \lambda^k, \mu^k) \tag{12}$$

B. Multi-objective Optimization

As discussed above, the system model is modeled as a multi-objective optimization problem with K objective functions, L inequality constraints, M equality constraints and J side constraints. The J side constraints are converted into $2J$ inequality constraints such that there are $T = L + 2J$ inequality constraints, thus the problem is modeled as in Eq. (13).

$$\min \mathbb{F}_k(\mathbb{x}) \quad k = 1,2, \dots, K$$
$$subject\ to \begin{cases} \mathbb{G}_t(\mathbb{x}) \le 0 & t = 1,2, \dots, T \\ \mathbb{H}_m(\mathbb{x}) = 0 & m = 1,2, \dots, M \end{cases} \tag{13}$$

The constraints divide the search space into two divisions – feasible and infeasible regions. The constraints are handled by using the Penalty Function approach. For each solution $\mathbb{x}^{(i)}$, the constraint violation for the inequality constraints $\mathbb{G}_t(\mathbb{x}^{(i)})$ for $t = 1,2, \dots, T$ are calculated as in Eq. (14).

$$\mathcal{w}_t(\mathbb{x}^{(i)}) = \begin{cases} |\mathbb{G}_t(\mathbb{x}^{(i)})|, & if\ \mathbb{G}_t(\mathbb{x}^{(i)}) > 0 \\ 0 & otherwise \end{cases} \tag{14}$$

The constraint violation for the equality constraints $\mathbb{H}_m(\mathbb{x})$ for $m = 1,2, \dots, M$ are calculated as in Eq. (15).

$$\mathcal{w}_m(\mathbb{x}^{(i)}) = |\mathbb{H}_m(\mathbb{x}^{(i)})| \tag{15}$$

Thereafter, all constraint violations are added together to get the overall constraint violation as in Eq. (16).

$$\Omega(\mathbb{x}^{(i)}) = \sum_{t=1}^{T} \mathcal{w}_t(\mathbb{x}^{(i)}) + \sum_{m=1}^{M} \mathcal{w}_m(\mathbb{x}^{(i)}) \tag{16}$$

This constraint violation is then multiplied with a penalty parameter $\mathcal{P}_k$ and then the product is added to each of the objective function values as in Eq. (17).

$$\mathcal{J}_k(\mathbb{x}^{(i)}) = \mathbb{F}_k(\mathbb{x}^{(i)}) + \mathcal{P}_k\Omega(\mathbb{x}^{(i)}) \tag{17}$$

The cost function $\mathcal{J}_k$ takes into account the constraint violations. For a feasible solution the corresponding Ω term is zero and $\mathcal{J}_k$ becomes equal to the original objective function $\mathbb{F}_k$. However, for an infeasible solution, $\mathcal{J}_k > \mathbb{F}_k$, thereby adding a penalty corresponding to the total constraint violation. One of the striking differences between single-objective optimization and multi-objective optimization is that in multi-objective optimization the objective(cost) functions constitute a multi-dimensional space, in addition to the usual design variable space $(\mathcal{D})$. This additional space is called the objective space $(\mathcal{Z})$. For each solution $\mathbb{x}$ in the design variable space, there exists a point in the objective space, denoted by $\mathcal{J}(\mathbb{x}) = \mathrm{z} = (z_1, z_2, \dots, z_K)^T$. The mapping takes place between an J-dimensional solution vector and an K-dimensional objective vector. Another difference between single-objective optimization and multi-objective optimization is that unlike single-objective optimization, multi-objective optimization with multiple conflicting objectives cannot have a single optimum solution which simultaneously optimizes all objectives. The resulting outcome is a set of optimal solutions with a varying degree of objective values called the Pareto-optimal solutions. To solve multi-objective optimization problem, there are a few classical methods like 'Weighted Sum Method', 'ε-Constraint Method', 'Weighted Metric Method', 'Benson's Method', 'Value Function Method', and 'Goal Programming Method'. All these classical methods use a single solution update in every iteration and mainly use a deterministic transition rule, however, in case of evolutionary algorithms (EA), a population of solutions is processed in every iteration (or generation). This feature alone gives an EA a tremendous advantage for its use in solving multi-objective optimization problems (MOOPs) [34].

For this research, a real-parameter elitist non-dominated sorting genetic algorithm (NSGA-II) is used to find the pareto optimal solutions. Initially a random parent population of the design variables P_0 is created of size N_P. For each individual, the values of each cost functions are calculated, and the population is sorted based on nondomination and each solution is assigned a rank (r) equal to its nondomination level $(\mathcal{F})$ and a crowding distance (d). Since the problem is formulated as a minimization problem, the vector $\mathbb{x}^{(1)}$ is partially less than another vector $\mathbb{x}^{(2)}$, $(\mathbb{x}^{(1)} \prec \mathbb{x}^{(2)})$, when no value of $\mathbb{x}^{(2)}$ is less than $\mathbb{x}^{(1)}$ and at least one value of $\mathbb{x}^{(2)}$ is strictly greater than $\mathbb{x}^{(1)}$. If $\mathbb{x}^{(1)}$ is partially less than $\mathbb{x}^{(2)}$, the solution $\mathbb{x}^{(1)}$ dominates $\mathbb{x}^{(2)}$ [34]. Any member of such vectors which is not dominated by any other member is said to be nondominated. To get an estimate of the density of solutions surrounding a particular solution in a nondomination level, a quantity called crowding distance that serves as an estimate of the perimeter of the cuboid formed by using the nearest neighbors as vertices is calculated [34]. For computing the crowding distance, first the number of solutions in $\mathcal{F}$ is calculated as $l = |\mathcal{F}|$, and for each i in the set $d_i = 0$ is assigned. Next for each objective

function k, the set is sorted in worst order of $\mathcal{J}_k$ and the sorted indices are stored in a vector: $I^k = sort(\mathcal{J}_k, >)$. Nest for each k, a large distance is assigned to the boundary solutions, $d_{I_1^k} = d_{I_l^k} = \infty$, and for all other solutions $j = 2\ to\ (l-1)$, the distance is shown by Eq. (18).

$$d_{I_j^k} = d_{I_j^k} + \frac{\mathcal{J}_k^{(I_{j+1}^k)} - \mathcal{J}_k^{(I_{j-1}^k)}}{\mathcal{J}_k^{max} - \mathcal{J}_k^{min}} \tag{18}$$

The index I_j denotes the solution index of the j-th member in the sorted list. The calculation is continued with other objective functions and the overall crowding distance value (d) is calculated as the sum of individual distance values corresponding to each objective. With the non-domination rank and crowding distance of each individual determined, the crowded tournament selection operator is used to select individuals for crossover [34]. The selection operator compares two solutions and returns the winner of the tournament. A solution i wins a tournament with another solution j if solution i has a better rank, that is, $r_i < r_j$. If they have the same rank, solution i wins if it has a better crowding distance, that is, $r_i = r_j$ and $d_i > d_j$. Next crossover, and mutation operators are used to create an offspring population Q_0 of size N_Q. For crossover, a blend crossover (BLX-α) operator is used [35]. For two parent solutions $\mathbb{x}_i^{(1,t)}$ and $\mathbb{x}_i^{(2,t)}$ (assuming $\mathbb{x}_i^{(1,t)} < \mathbb{x}_i^{(2,t)}$) in generation (t), the BLX-α randomly picks a solution in the range $[\mathbb{x}_i^{(1,t)} - \alpha\left(\mathbb{x}_i^{(2,t)} - \mathbb{x}_i^{(1,t)}\right), \mathbb{x}_i^{(2,t)} + \alpha\left(\mathbb{x}_i^{(2,t)} - \mathbb{x}_i^{(1,t)}\right)]$. Thus if u_i is a random number between 0 and 1, the following is an offspring:

$$\mathbb{x}_i^{(1,t+1)} = (1-\gamma_i)\mathbb{x}_i^{(1,t)} + \gamma_i \mathbb{x}_i^{(2,t)} \tag{19}$$

where, $\gamma_i = (1+2\alpha)u_i - \alpha$, which is uniformly distributed for a fixed value of α. If $\alpha = 0$, this crossover creates a random solution in the range $(\mathbb{x}_i^{(1,t)}, \mathbb{x}_i^{(2,t)})$. This operator allows the searching of the entire space early on and also allow to maintain a focused search when the population tends to converge in some region in the search space. During mutation, a non-uniform mutation operator is used, where the probability of creating a solution closer to the parent is more than the probability of creating one away from it as illustrated Eq. (20) [36]. However, as the generation (t) proceed, this probability of creating solutions closer to the parent gets higher and higher.

$$\mathbb{x}_i^{(t+1)} = \mathbb{x}_i^{(t)} + \tau\left(\mathbb{x}_i^{(U)} - \mathbb{x}_i^{(L)}\right)\left(1 - u_i^{\left(1-\frac{t}{t_{max}}\right)^b}\right) \tag{20}$$

Here, τ takes a Boolean value, -1 or 1, each with a probability of 0.5. The parameter u_i is a random number between 0 and 1, t_{max} is the maximum number of allowed generations, and b is a user defined parameter. In this way, from early on the above mutation operator acts like a uniform distribution, while in later generations it acts like Dirac's function, thus allowing a focused search. For variables that have integer constraints, it is rounded off to the nearest integer after crossover and mutation. Since elitism is introduced by comparing current population with previously found best nondominated solutions, the procedure is different after the initial generation [34]. For the t^{th} generation, first a combined population $R_t = P_t \cup Q_t$ is formed. The population R_t is of size $N_P + N_Q$. Then, the population is sorted according to nondomination. Since all previous and current population members are included in R_t, elitism is ensured. Now, solutions belonging to the best nondominated set $\mathcal{F}_1$ are emphasized more than any other solution in the combined population. If the size of $\mathcal{F}_1$ is smaller than N_P, we choose all members of the set $\mathcal{F}_1$ for the new population P_{t+1}. The remaining members of the population P_{t+1} are chosen from subsequent nondominated fronts in the order of their ranking. Thus, solutions from the set $\mathcal{F}_2$ are chosen next, followed by solutions from the set $\mathcal{F}_3$, and so on. This procedure is continued until no more sets can be accommodated. When the set $\mathcal{F}_l$ is the last nondominated set beyond which no other set can be accommodated, the count of solutions in all sets from $\mathcal{F}_1$ to $\mathcal{F}_l$ would be larger than the population size. To choose exactly N_P population members, we sort the solutions of the last front $\mathcal{F}_l$ using the crowding distance operator in descending order and choose the best solutions needed to fill all population slots as shown in Fig. 5. The new

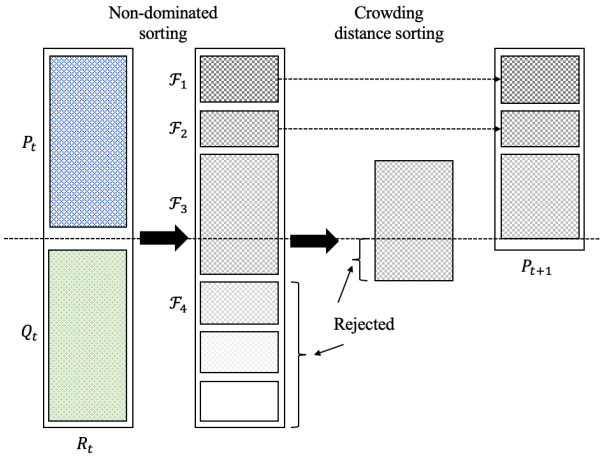

Fig. 5 Schematic of the NSGA-II Procedure.

population P_{t+1} of size N_P is now used for selection, crossover, and mutation to create a new population Q_{t+1} of size N_Q. To measure the performance of the multi-objective optimization, a metric Δ that measures the extent of spread achieved among the obtained pareto solutions is used [34]. First, the Euclidean distance d_i between consecutive solutions obtained in the nondominated set of solutions is calculated. Next the average $\bar{d}$ of these distances is calculated and thereafter Δ is calculated by Eq. (21).

$$\Delta = \frac{\sum_{i=1}^{N-1} |d_i - \bar{d}|}{(N-1)\bar{d}} \tag{21}$$

Where, N is the number of pareto optimal solutions. With N solutions, there are $(N-1)$ consecutive distances, thus the denominator is the value of the numerator for the case when N solutions lie in one solution. When there is a large variance in d_i, the value of the above metric can be greater than one. However, a good distribution would make all distances $d_i = \bar{d}$, making the metric to take a value zero.

VI. Environment and Subsystem Models

The ambient environmental factors present on the lunar and Martian surface pose some of the most difficult challenges for the success of long-term robotic exploration. These factors include dangerous radiation levels and high range of temperatures that can pose a variety of complications like thermal expansion and contraction, bit flips, and electrical leakage. Moreover, the dynamics and efficiency of the robot is dependent on the gravity and surface interaction parameters. As such, the design of the robot should take these factors in account. The environment models include a) Temperature model, b) Radiation model, c) Gravitational model, and d) Surface interaction model for the surface of the Moon and Mars that interacts with the design of SphereX.

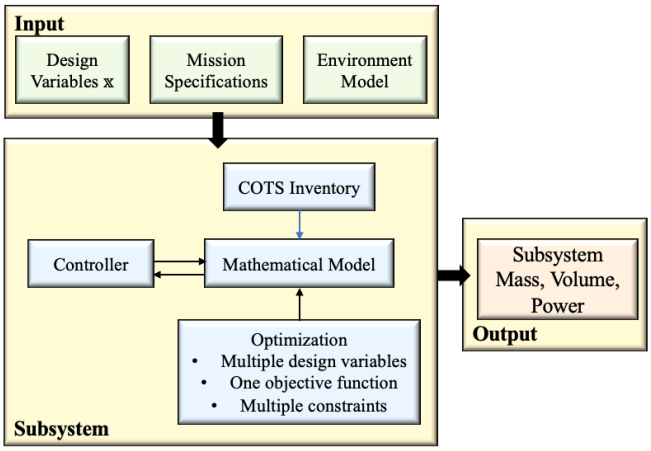

Fig. 6 Block representation of each subsystem models.

Mathematical models for each subsystem disciplines of SphereX are also developed for this research. The modeled subsystems are mobility system, power system, thermal system, shielding, communication system, avionics and shell [38, 39]. Moreover, for the mobility subsystem, multiple controllers are developed that interacts with the mathematical model during each iteration of the optimization process. Each subsystem is defined by multiple design variables, one objective function and multiple equality, inequality and side constraints. The mathematical model is used for iteration to find the optimal design variables and the mass, volume and power requirements for each subsystem are calculated as shown in Fig. 6. The mass, volume and power requirements for each subsystem will then be used for the system level optimization process.

VII. Automated Assembly

With all the dimensions of each subsystem calculated a program was written for automated assembly for all the components and MATLAB VRML was used to visualize the model. Each subsystem was divided into individual components and separate codes with conditional statements were written for each component. For the final assembly, first a sphere of radius r is created and then the insulation and shielding spheres are added inside it. Next, the mobility subsystem is assembled which occupies the lower half of SphereX. On top of the mobility subsystem, the avionics stack and the power system are assembled. If tanks are required for the system, they are assembled next based on available free spaces. The antennas are then assembled on the outer surface of the sphere. For each subsystem a binary assembly index $I_i, i = 1,2,\dots,N$ is assigned based on its feasibility of assembling inside the sphere, where N is the number of subsystems. Finally, the assembly index of the entire system is determined by an AND logical operator as $Index = \prod_{i=1}^{N} I_i$ which will be used as a constraint for the system level multi-objective optimization problem.

VIII. System Level Optimization

With all the subsystem models and their respective optimization models defined, this section defines the system level optimization model. The objective of our MDO approach is to find the optimum mass and radius of the robot

(SphereX) that accommodates the maximum payload in terms of mass, volume and power based on predefined mission specifications. The problem is formulated as a multi-objective optimization problem (MOOP) with 12 design variables $\mathbb{x} = [m, r, P, ms_{ID}, sd_1, sd_2, ps_{ID}, c_{ID}, p_{ID}, b_{ID}, t_{ID}, a_{ID}]$, 4 objective functions and 5+ constraints, where m and r are the mass and radius of the robot, P is the power demand, ms_{ID} defines the type, and sd_1 and sd_2 the subtype of the mobility system, ps_{ID} defines the type of power system and the COTS IDs are defined as main computer (c_{ID}), power management board (p_{ID}), battery (b_{ID}), radio transceiver board (t_{ID}), and attitude controller board (a_{ID}). Based on the values of mass m, radius r, and power demand P with bounds $m^{(b)} = [m^{(L)} \quad m^{(U)}]$, $r^{(b)} = [r^{(L)} \quad r^{(U)}]$, and $P^{(b)} = [P^{(L)} \quad P^{(U)}]$ it is normalized between $[0 \quad 1]$ as shown in Eq. (22).

$$\underline{m} = \frac{m - m^{(L)}}{m^{(U)} - m^{(L)}}, \quad \underline{r} = \frac{r - r^{(L)}}{r^{(U)} - r^{(L)}}, \quad \underline{P} = \frac{P - P^{(L)}}{P^{(U)} - P^{(L)}} \tag{22}$$

The first objective is then defined as $\mathbb{F}_1(\mathbb{x}) = \alpha_1 \underline{m} + \alpha_2 \underline{r}$. Based on the design variable, the mass, volume and power of each subsystem is calculated and the mass and volume of the payload is calculated as $m_{pay} = m - m_{sys}$, $\mathcal{V}_{pay} = \mathcal{V} - \mathcal{V}_{sys}$, and $P_{pay} = P - P_{sys}$, where $\mathcal{V} = 4\pi r^3/3$. The payload mass, volume and power ratio are then calculated as $m_r = m_{pay}/m$, $\mathcal{V}_r = \mathcal{V}_{pay}/\mathcal{V}$, and $P_r = P_{pay}/P$. The second objective function is then defined as $\mathbb{F}_2(\mathbb{x}) = 1 - (\alpha_3 m_r + \alpha_4 \mathcal{V}_r)$. m_{sys}, $\mathcal{V}_{sys}$, and P_{sys} are the total mass, volume and power of all the subsystems described in section VI, and α_1, α_2, α_3, and α_4 are weights. The third and fourth objective functions are defined as $\mathbb{F}_3(\mathbb{x}) = \underline{P}$ and $\mathbb{F}_4(\mathbb{x}) = 1 - P_r$. Three constraints are added to the optimization problem. The first three constraints are $m_r > 0$, $\mathcal{V}_r > 0$ and $P_r > 0$. The fourth constraint is $Index = 1$. The fifth constraint is that the bandwidth of the transceiver selected lies within the resonating frequency of the antenna designed. Finally, other constraints can be added based on other user defined parameters, (e.g. the clock frequency of the computer selected is greater than a user-defined desired clock frequency, storage capacity of the computer selected is greater than a user-defined value etc.). The optimization problem is then mathematically formulated as Eq. (23).

$$\begin{aligned} &\min \mathbb{F}_1(\mathbb{x}) = \alpha_1 \underline{m} + \alpha_2 \underline{r} \\ &\min \mathbb{F}_2(\mathbb{x}) = 1 - (\alpha_3 m_r + \alpha_4 \mathcal{V}_r) \\ &\min \mathbb{F}_3(\mathbb{x}) = \underline{P} \\ &\min \mathbb{F}_4(\mathbb{x}) = 1 - P_r \\ &subject\ to \begin{cases} \mathbb{G}_1(\mathbb{x}) \equiv m_r > 0 \\ \mathbb{G}_2(\mathbb{x}) \equiv \mathcal{V}_r > 0 \\ \mathbb{G}_3(\mathbb{x}) \equiv P_r > 0 \\ \mathbb{G}_4(\mathbb{x}) \equiv Index = 1 \\ \mathbb{G}_5(\mathbb{x}) \equiv f_{trans}^{(L)} + \dfrac{BW}{2} \le f_{r(ant)} \le f_{trans}^{(U)} - \dfrac{BW}{2} \end{cases} \end{aligned} \tag{23}$$

Since our problem is a constrained multi-objective problem, the search space is divided into two regions: feasible and infeasible regions. Hence, all pareto optimal solutions must also lie in the feasible region. The penalty function approach was used to handle the constraints within the objective functions as discussed in Section V. For each solution $\mathbb{x}^{(i)}$, the constraint violation for each constraint are calculated and then added together to get the overall constraint violation $\Omega(\mathbb{x}^{(i)})$. This constraint violation is then multiplied with a penalty parameter $\mathcal{P}$ and the product is added to each of the objective functions. Thus, the constrained multi-objective optimization problem is converted into an unconstrained multi-objective optimization problem with the 4 cost functions defined as Eq. (24).

$$\mathcal{J}_k(\mathbb{x}) = \mathbb{F}_k(\mathbb{x}) + \mathcal{P}\Omega(\mathbb{x}), \quad k = 1,2,3,4 \tag{24}$$

Robot Mass	Robot Radius	Robot Power	Mobility System			Power System	Computer Board	Power Board	Battery	Transceiver	ADCS Board
m	r	P	ms_{ID}	sd_1	sd_2	ps_{ID}	c_{ID}	p_{ID}	b_{ID}	t_{ID}	a_{ID}
Real numbers			Integers								

Fig. 7 System level design variables expressed as a gene for NSGA-II.

With the 4 cost functions defined, an elitist non-dominated sorting genetic algorithm (NSGA-II) is used to find the pareto optimal solutions as discussed in Section 5. For creating the initially random parent population P_0, the values of m, r and P are chosen with a uniform distribution $m = \mathcal{U}(m^{(U)}, m^{(L)})$, $r = \mathcal{U}(r^{(U)}, r^{(L)})$, and $P =$

$\mathcal{U}(P^{(U)}, P^{(L)})$, the integer values of $ms_{ID}, sd_1, sd_2, ps_{ID}$ are chosen at random from the available options, and the integer COTS IDs are chosen at random from the COTS inventory.

IX. Results and Discussion

This section provides the results of simulations performed for different mission scenarios. The simulation results are presented in the form of pareto optimal design solutions for two exploration missions 1) Surface exploration mission on Mare Tranquilitatis, and 2) Subsurface exploration mission of Mare Tranquilitatis pit on the surface of the Moon. The mission specifications were to explore 1000 meters over a mission lifetime of 5 hours and 3000 meters over a mission lifetime of 15 hours respectively. Along with the pareto optimal solutions, the history of selection of the mobility and power system is presented that showed the selection of optimal mobility and power system for different mission scenarios. To better understand the selection probability of the mobility and power system, a comparative analysis is presented for all combinations of propulsive mobility system and power system for varying mission exploration requirements. Finally, the performance of the controllers used in designing the robot is presented through Monte Carlo simulations.

A. Test Scenario-1: Surface exploration on Mare Tranquilitatis

The first simulation was run to perform surface exploration on Mare Tranquilitatis. Mare Tranquilitatis is a lunar mare that sits within the Tranquilitatis basin on the Moon at 8.5°N 31.4°E, which was also the landing site for the first manned landing on the Moon (Apollo-11) on July 20, 1969. The mission target is to explore 1000m around the Apollo-11 landing site in 5 hours. The environmental conditions used for the simulations were gravity $g = 1.62\text{m/s}^2$, ambient temperature $T_a = 340 \pm 40\text{K}$, radiation dose rate $I_0 = 100\text{rad/yr}$, and soil properties of Lunar soil.

In addition to the constraints discussed in Section VIII, 3 additional constraints based on user-defined parameters are added. The clock frequency of the computer should be greater than 500MHz, storage capacity should be greater than 1Gbyte, the power board should have at least 3 output channels ranging from 3-15V. For the communication system, 5 robots were considered to explore in coordination, and the antenna was designed at 1GHz frequency with 3 antennas in the array. The bounds on the design variables were $m^{(b)} = [1 \quad 8]$ kg, $r^{(b)} = [10 \quad 40]$ cm, $P^{(b)} = [5 \quad 30]$ W. 3 different types of mobility systems and 2 different types of power systems were used in the simulation. For the COTS inventory 10 different types of each component were collected from different manufacturers. Fig. 8 shows the scatter plot matrix of the pareto optimal solutions found after 500 generations with 100 individuals in each generation. With 4 objective function, all 6 pairs are plotted.

Fig. 9(Left) shows the number of individuals in each pareto front and total number of pareto fronts over generations. It can be seen that the simulation started with 92 pareto fronts and converged into 1 pareto front after 4 generations. Fig. 9(Right) shows the spread Δ of the pareto solutions over generations that shows the extent of spread achieved. It can be seen that the spread was close to zero which shows a good distribution.

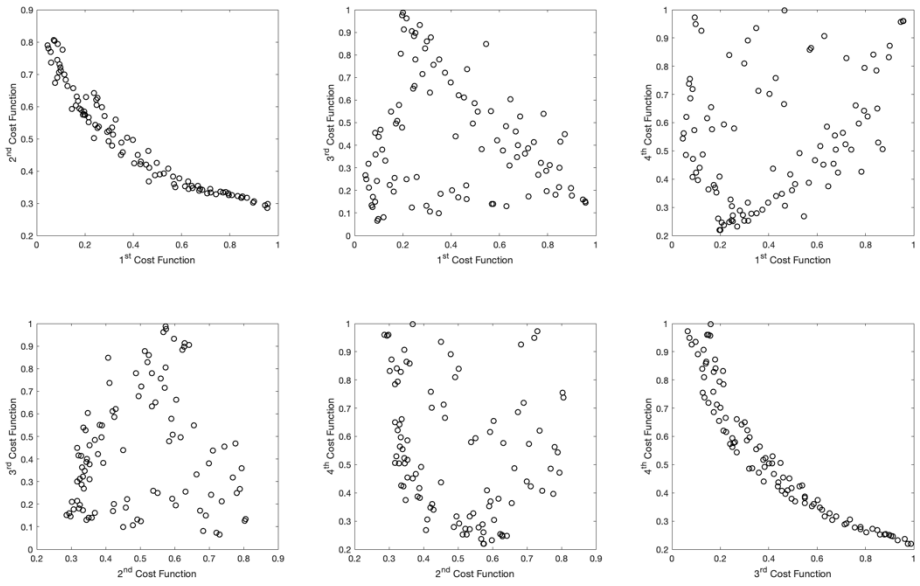

Fig. 8 Scatter plot matrix of the pareto optimal solutions found after 500 generations with 4 objective functions.

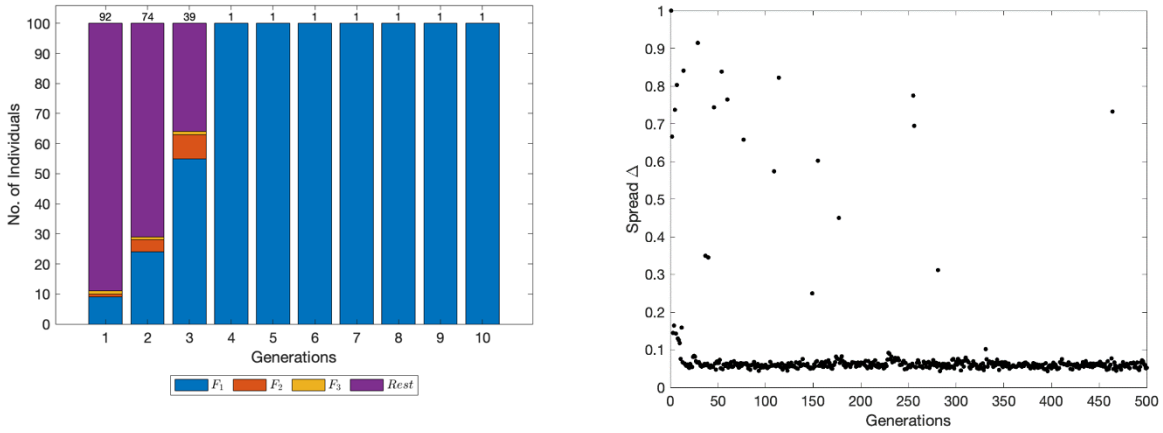

Fig. 9 (Left) Number of individuals in each pareto fronts over generations. The number on top of each bar shows the total number of pareto fronts in each generation. (Right) Performance measure Δ, that shows the extent of spread achieved over 500 generations.

Fig. 10 shows the mass, volume, and power budget of the robot for the pareto-optimal solutions found. It can be seen that the minimum and maximum values of mass, volume and power available for the payload are $m_{pay}^{(L)} = 0.20$, $m_{pay}^{(U)} = 4.56$ kg, $\mathcal{V}_{pay}^{(L)} = 0.00022$, $\mathcal{V}_{pay}^{(U)} = 0.0109$ m^3, and $P_{pay}^{(L)} = 0.001$, $P_{pay}^{(U)} = 22.81$ W. The average values of the mass, volume and power available for the payload are $\bar{m}_{pay} = 2.27$ kg, $\bar{\mathcal{V}}_{pay} = 0.0056$ m^3, and $\bar{P}_{pay} = 9.04$ W over the 100 pareto optimal solutions. Moreover, the average value of the total mass of the robot is 3.9 kg.

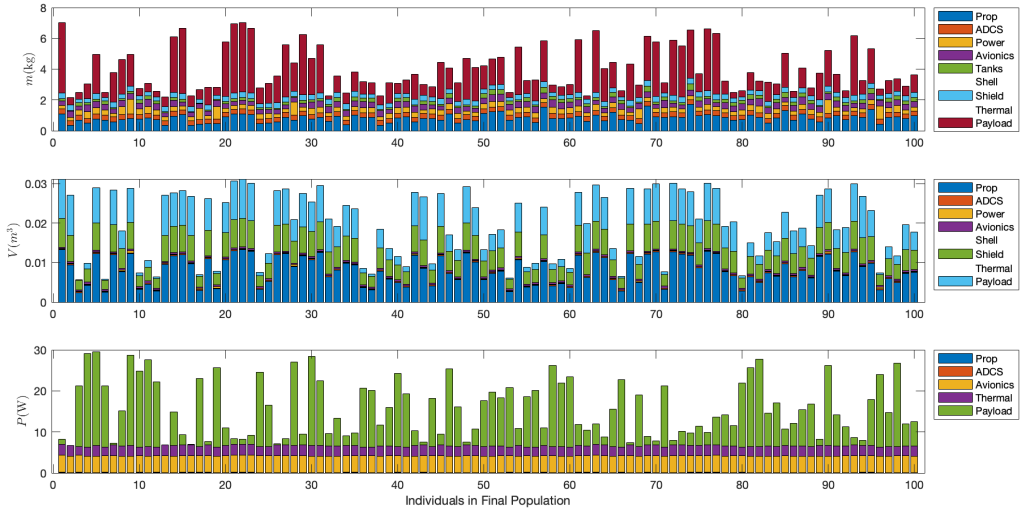

Fig. 10 Mass, volume and power budget of the 100 individuals in the pareto front.

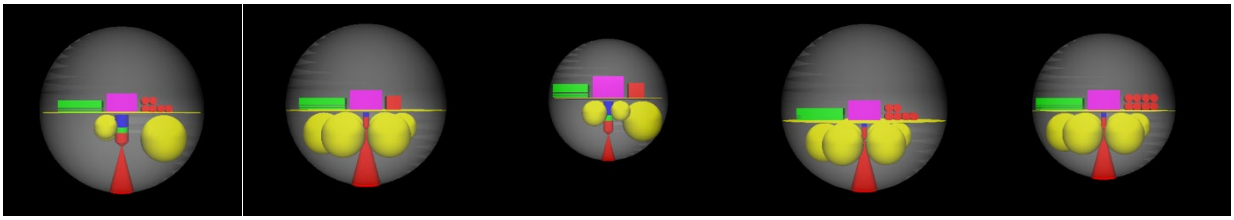

Fig. 11 3D visualization models of the designs in the pareto front. From left to right (1) Design 1, (2) Design 30, (3) Design 71, (4) Design 89, and (5) Design 100.

Fig. 12 shows the number of instances, different modes of mobility and power system is selected over generations. It can be seen from Fig. 12(Top-Left) that among the three modes of mobility (hopping, rolling, and wheeled), hopping is the most efficient one as the other two are rejected within 24 generations. It is also clear from Fig. 12(Top-Right) that among three modes of hopping mobility (propulsive, mechanical and reaction-wheel), propulsive hopping is the most efficient one as the other two are rejected within 14 generations. Also, among the three propellants used for propulsive hopping, steam-propulsion is rejected within 69 generations, while neither RP1/H_2O_2, nor H_2/O_2 propulsion is rejected as shown in Fig. 12(Bottom-Left). This shows that both the options are viable for this mission scenario. Moreover, among the two power systems (battery and fuel cell), neither got rejected making both options viable as shown in Fig. 12(Bottom-Right).

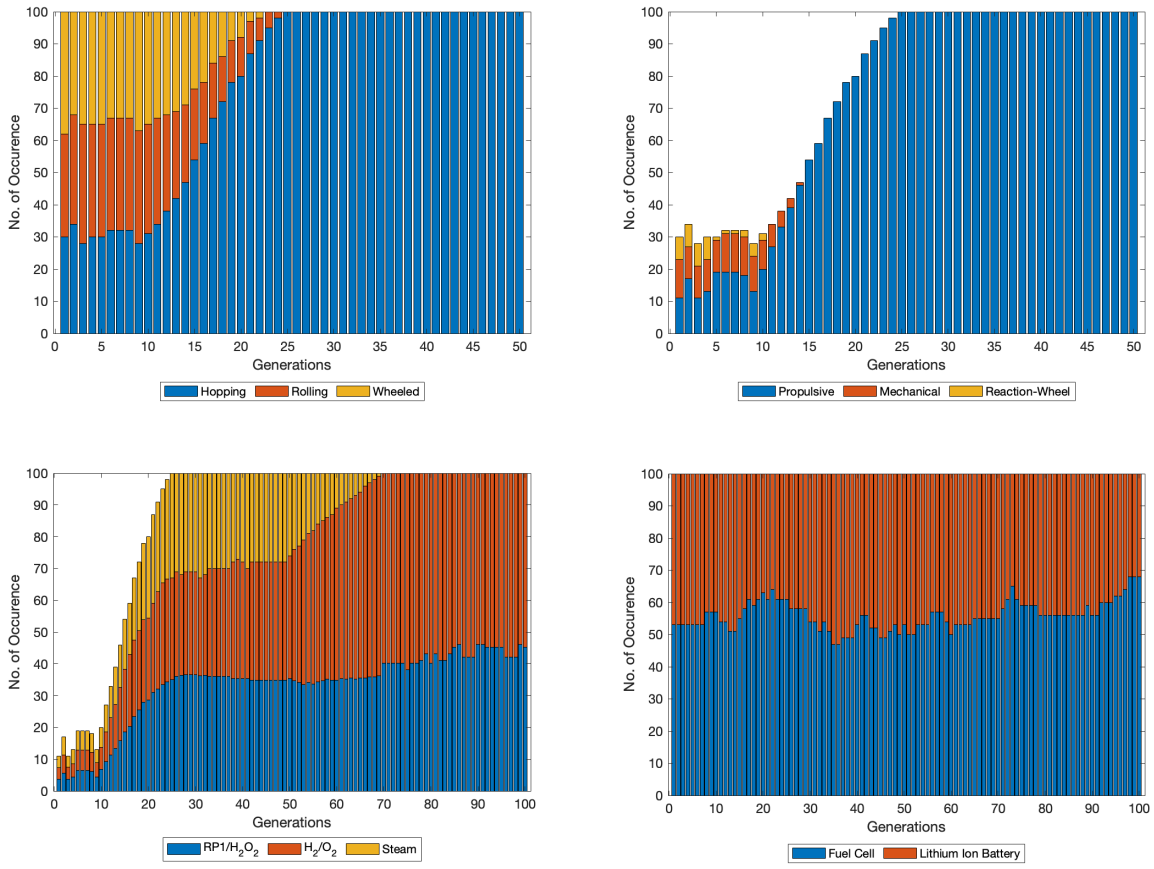

Fig. 12 (Top-Left) Number of instances hopping, rolling and wheeled modes of mobility selected over generations. (Top-Right) Number of instances propulsive, mechanical and reaction-wheel hopping modes of mobility selected over generations. (Bottom-Left) Number of instances H_2/O_2, RP1/H_2O_2 and steam based propulsive hopping modes of mobility selected over generations. (Bottom-Right) Number of instances fuel cells and lithium-ion batteries selected over generations.

B. Test Scenario-2: Sub-Surface exploration of Mare Tranquilitatis pit

The second simulation was run to perform sub-surface exploration of Mare Tranquilitatis Pit at 8.33°N 33.22°E. Lunar Reconnaissance Orbiter Camera (LROC) images reveal that the pit diameter ranges from 86 to 100m with a maximum depth from shadow measures of ~107m and that it opens into a sublunarean void of at least 20meters in extent. However, the sublunarean void might extend to a few kilometers in length and so mission specification is to explore 1000m of the sublunarean void. The con-ops for performing this mission is shown in Fig.13(Left). A lander carrying multiple SphereX robots would descent nearby Mare Tranquilitatis Pit and deploy the robots one by one. Each robot will have three phases 1. Surface operation to approach the pit entrance, 2. Pit entrance maneuver, and 3. Sub-surface operation to explore the pit. The mission target is to explore 2000m on the surface in 10 hours, 50m in 10 minutes to enter the pit and 1000m inside the pit in 5 hours as seen in Fig. 13(Right). The environmental conditions

used for the simulations were gravity $g = 1.62\text{m/s}^2$, ambient temperature $T_a = 340\text{K}$ (surface), $T_a = 250\text{K}$ (sub-surface), radiation dose rate $I_0 = 100\text{rad/yr}$ (surface), $I_0 = 0\text{rad/yr}$ (sub-surface), and soil properties of Lunar soil. The constraints and the bounds on the design variables used were same as discussed in Section IX(A). For entering the pit, the robot needs a soft-landing maneuver, as such the modes of mobility except for propulsive hopping are unfit for this mission scenario. In addition to the three phases discussed for propulsive hopping, the robot has an additional soft-landing phase.

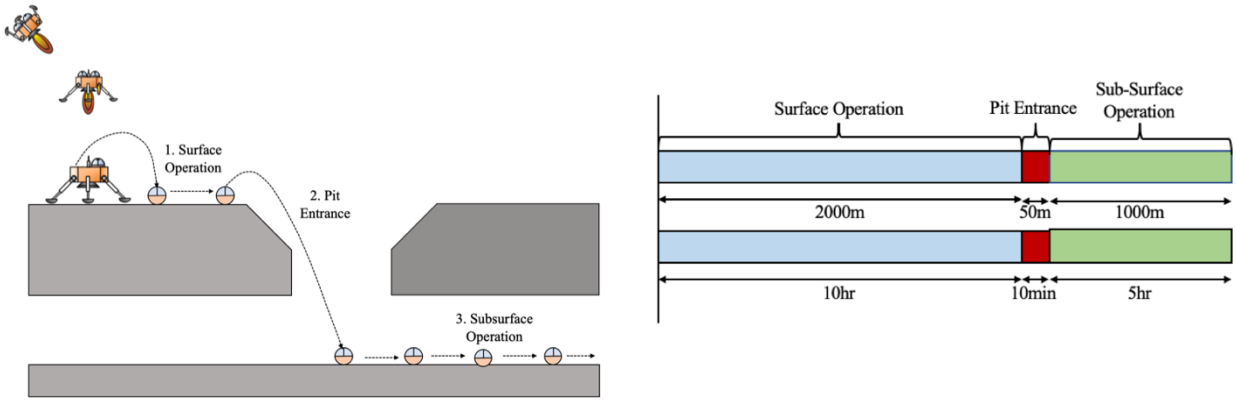

Fig. 13 (Left) Concepts of operation for exploring Lunar pits, (Right) Mission exploration requirements for the robot to enter the pit and explore.

Fig. 14 Mass, volume and power budget of the 100 individuals in the pareto optimal front.

Fig. 14 shows the mass, volume, and power budget of the robot for the pareto-optimal solutions found. It can be seen that the minimum and maximum values of mass, volume and power available for the payload are $m_{pay}^{(L)} = 0.20, m_{pay}^{(U)} = 2.93$ kg, $\mathcal{V}_{pay}^{(L)} = 0.00021, \mathcal{V}_{pay}^{(U)} = 0.0104$ m^3, and $P_{pay}^{(L)} = 0.04, P_{pay}^{(U)} = 22.20$ W. The average values of the mass, volume and power available for the payload are $\bar{m}_{pay} = 1.82$ kg, $\bar{\mathcal{V}}_{pay} = 0.0047$ m^3, and $\bar{P}_{pay} = 8.6$ W over the 100 pareto optimal solutions. Moreover, it can be seen that of average value of the total mass of the robot increased to 5.8 kg from 3.9 kg in test scenario 1. Fig. 15 shows the number of instances, different modes of mobility and power system is selected over generations. Also, among the three propellants used for propulsive hopping, steam-propulsion is rejected within 48 generations, while H_2/O_2 propulsion is rejected within 55 generations. This shows that RP1/H_2O_2 propulsion is the fittest mobility option for this mission scenario. Moreover, among the two power systems, battery system got rejected within 46 generations, thus making fuel cell power system as the fittest option.

From Fig. 12 and 15 it is clear that the selection of mobility and power system depends on the mission exploration and mission time goals.

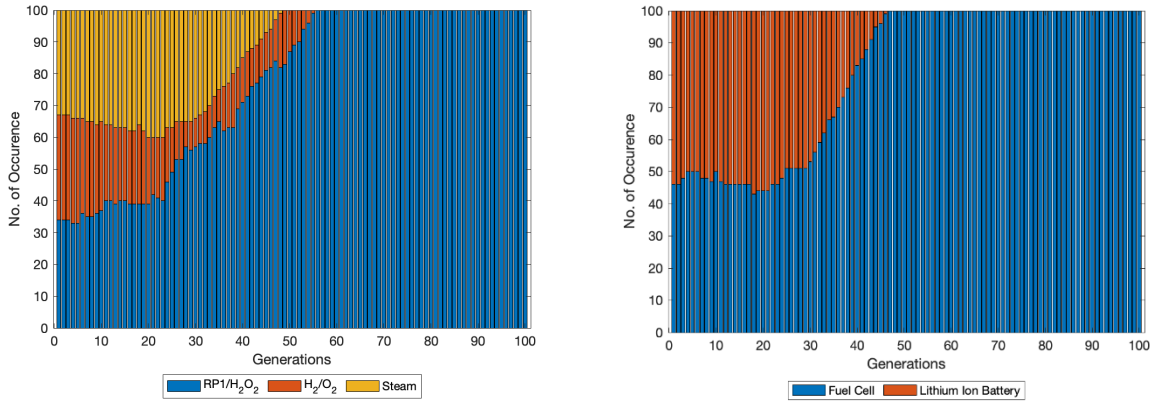

Fig. 15 (Left) Number of instances H_2/O_2, $RP1/H_2O_2$ and steam based propulsive hopping modes of mobility selected over generations. (Right) Number of instances fuel cells and lithium-ion batteries selected over generations.

C. Comparative Analysis

Since, the selection of the propulsive hopping mobility and power system varied across the two mission scenarios presented in Section IX(A) and IX(B), a comparative study of the two systems is done for varying exploration distance and mission time. For the comparative study, the choice of the avionics was fixed, and the available payload mass, volume and power were considered 1kg, 1000cm3, and 10W respectively. As such the problem is expressed as a single-objective optimization problem to minimize the mass of the robot with 2 design variables $\mathbb{x} = [m, r]$. For each design variable, the mass of each subsystem is calculated and then added together to find the total mass of the system m_T. Two constraints were added such that $m = m_T$, and the assembly index $Index = 1$. The optimization problem is mathematically formulated as Eq. (25).

$$\min \mathbb{F}(\mathbb{x}) = m$$
$$subject\ to \begin{cases} \mathbb{G}_1(\mathbb{x}) \equiv (m - m_T)^2 = 0 \\ \mathbb{G}_2(\mathbb{x}) \equiv Index = 1 \end{cases} \tag{25}$$

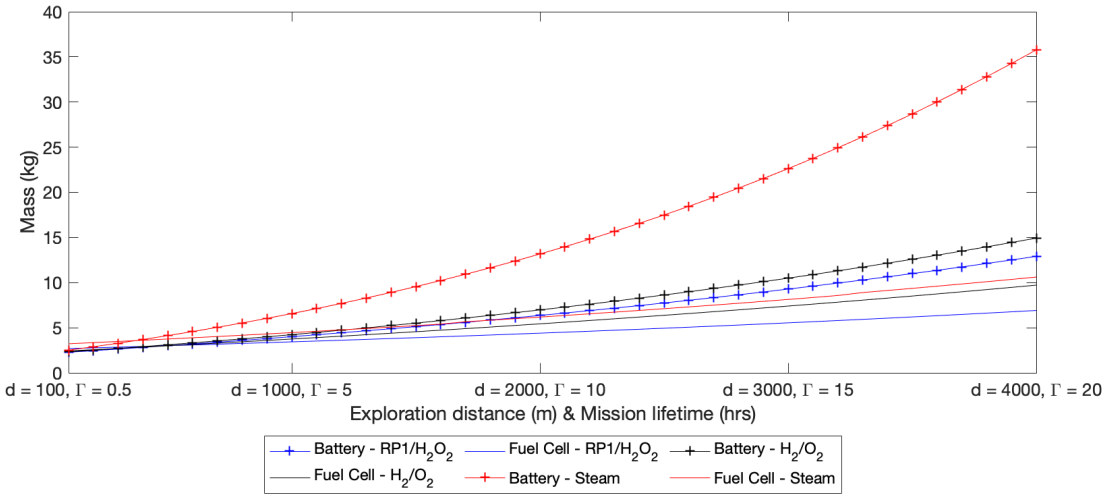

Fig. 16 Mass of the robot for all combinations of propulsive mobility system and power system for varying exploration distance and mission time.

Multiple simulations were performed for each combination of the propulsive hopping mobility and power system to find the optimal mass of the robot for varying exploration distance and mission time on the surface of the Moon. Fig. 16 shows the mass of the robot for each combination. It can be seen that for an exploration objective of $d_{targ} = 100$ m and $\Gamma = 0.5$ hrs, the system with lithium-ion batteries and RP1/H_2O_2 propulsive mobility is the optimal choice with multiple combinations close to each other, however as the exploration objective increases to $d_{targ} = 4000$ m and $\Gamma = 20$ hrs, the system with fuel cells and RP1/H_2O_2 propulsive mobility is the optimal choice. It can also be seen that for each of the propulsive mobility system, the ones with battery system is better than the ones with fuel cells for low exploration objectives, but as the exploration objectives increases the ones with fuel cells are far better.

X. Conclusion

The report formulated and solved a multidisciplinary optimization (MDO) problem for SphereX, which included geometric design along with mobility and temperature control for planetary surface exploration missions. The problem was constructed with seven disciplines: mobility system, power system, communication, avionics, thermal, radiation shielding and shell which interacted with a COTS inventory for electronics, mobility controller for exploration and a thermal controller for maintaining the body temperature of the robot to find optimal design solutions for a specific planetary exploration mission. To solve the problem, the AMDCO framework was implemented that used a genetic algorithm based multi-objective optimizer at the system level to find the Pareto-optimal results while using gradient-based optimization techniques at the subsystem level. We have demonstrated that finding the optimal design variables associated with all the major disciplines of SphereX for a predefined exploration task is feasible through a rigorous multidisciplinary approach. The approach provides a system-level perspective of the problem with sufficient depth to capture high-level trade-offs and reveal insights that are perhaps not obvious at the discipline level. The solution provides a geometric solution that is useful for ground development of SphereX taking into consideration its operational and exploration goals on a target environment. For operational point of view, each of the designs identified by the multidisciplinary optimization process needs further research and development. Future work will involve using these design solutions to perform path-planning with multiple robots to explore a target environment. In addition to that, hardware experimental results will be shown for exploring unknown environments like caves and lava tubes for mapping and localization.

References

[1] National Academies of Sciences, Engineering, and Medicine. 2011. Vision and Voyages for Planetary Science in the Decade 2013-2022. Washington, DC: *The National Academies Press*. doi: 10.17226/13117.

[2] R.V. Wagner and M.S. Robinson, "Distribution, formation mechanisms, and significance of lunar pits," *Icarus*, Vol. 237, pp. 52-60, 2014.

[3] M.S. Robinson et al., "Lunar reconnaissance orbiter camera (LROC) instrument overview," *Space Science Reviews*, Jan. 2010, Vol. 150, No 1-4, pp.81-124.

[4] G. Heiken, D. Vaniman and B. French, "Lunar Sourcebook: A User's Guide to the Moon," *Cambridge Univ. Press*, 1991.

[5] National Academies of Sciences, Engineering, and Medicine. 2016. NASA Space Technology Roadmaps and Priorities Revisited. Washington, DC: *The National Academies Press*. doi: 10.17226/23582.

[6] National Academies of Sciences, Engineering, and Medicine. 2016. Achieving Science with CubeSats: Thinking Inside the Box. Washington, DC: *The National Academies Press*.

[7] B. Zhang, H. Teng, Y. Shi, "Layout optimization of satellite module using soft computing techniques," *Applied Soft Computing*, 8(1):507-521, 2008.

[8] D. J. Richie, V. J. Lappas, P. L. Palmer, "Sizing/optimization of a small satellite energy storage and attitude control system," *Journal of Spacecraft and Rockets*, 44(4):940-952, 2007.

[9] S. Jain, D. Simon, "Genetic algorithm-based charge optimization of lithium-ion batteries in small satellites," *19th AIAA/USU Conference on Small Satellites*, Utah State University, Logan, UT, 2005.

[10] P. V. Hull, M. Tinker, M. SanSoucie, K. Kittredge, "Thermal analysis and shape optimization of an in-space radiator using genetic algorithms," *Space Technology and Applications International Forum – Staif* 2006.

[11] A. Boudjemai, M. H. Bouanane, L. Merad, A. M. Mohammed, "Small satellite structural optimization using genetic algorithm approach," *Recent Advances in Space Technologies*, 2007.

[12] J. Olds, G. Walberg, "Multidisciplinary design of a rocket-based combined cycle SSTO launch vehicle using Taguchi methods," *AIAA/AHS/ASEE Aerospace Design Conference*, Irvine, CA, 1993.

[13] R. D. Braun, R. W. Powell, R. A. Lepsch, D. O. Stanley, I. M. Kroo, "Comparision of two multidisciplinary optimization strategies for launch-vehicle design," *Journal of Spacecraft and Rockets*, 32(3):404-410, 1995.

[14] L. F. Rowell, R. D. Braun, J. R. Olds, R. Unal, "Recent experiences in multidisciplinary conceptual design optimization for launch vehicles," *AIAA, NASA, and ISSMO, Symposium on Multidisciplinary Analysis and Optimization*, Bellevue, WA, 1996.

[15] R. A. Hickman, "Integrated modeling for launch system architecture optimization," *INCOSE International Symposium*, Wiley Online Library, 1997.

[16] M. Kesselmann, "Optimization and evaluation of single-staged transportation systems example for the solution of highly complex technical problems," *INCOSE International Symposium*, Wiley Online Library, 1997.
[17] M. G. Matossian, "Earth observing system mission design - constrained optimization of the EOS constellation configuration design," *46th International Astronautical Congress*, Oslo, Norway, 1995.
[18] T. Mosher, "Applicability of selected multidisciplinary design optimization methods to conceptual spacecraft design," *6th AIAA/NASA/ISSMO, Symposium on Multidisciplinary Analysis and Optimization*, 1996.
[19] E. Riddle, "Use of optimization methods in small satellite systems analysis," *12th AIAA/USU Conference on Small Satellites*, Utah State University, Logan, UT, 1998.
[20] D. A. Bearden, "A methodology for spacecraft technology insertion analysis balancing benefit, cost, and risk," *PhD thesis*, University of Southern California, 1999.
[21] J. George, J. Peterson, S. Southard, "Multidisciplinary integrated design assistant for spacecraft (MIDAS)," *Proceedings of American Institute of Aeronautics and Astronautics (AIAA)*, 1995.
[22] A. S. Fukunaga, S. Chien, D. Mutz, R. L. Sherwood, A. D. Stechert, "Automating the process of optimization in spacecraft design," *IEEE Aerospace Conference*, 1997.
[23] T. Mosher, "Spacecraft design using a genetic algorithm optimization approach," *IEEE Aerospace Conference*, 1998.
[24] T. Mosher, "Conceptual spacecraft design using a genetic algorithm trade selection process," *Journal of Aircraft*, 36(1):200-208, 1999.
[25] G. M. Stump, M. Yukish, T. W. Simpson, J. J. O'Hara, "Trade space exploration of satellite datasets using a design by shopping paradigm," *IEEE Aerospace Conference*, 2004.
[26] D. Barnhart, T. Kichkaylo, L. Hoag, "SPIDR: Integrated systems engineering design-to-simulation software for satellite build," *Conference on Systems Engineering Research*, 2009.
[27] A. Ravanbakhsh, S. Franchini, "Multiobjective optimization applied to structural sizing of low-cost university-class microsatellite projects," *Acta Astronautica*, 79:212-220, 2012.
[28] E. R. Taylor, "Evaluation of multidisciplinary design optimization techniques as applied to spacecraft design," *IEEE Aerospace Conference Proceedings*, 2000.
[29] C. D. Jilla, D. W. Miller, "Multi-objective, multidisciplinary design optimization methodology for distributed satellite systems," *Journal of Spacecraft and Rockets*, 41(1):39-50, 2004.
[30] A. Jafarsalehi, P. M. Zadeh, M. Mirshams, "Collaborative optimization of remote sensing small satellite mission using genetic algorithms," *Iranian Journal of Science and Technology, Transactions of Mechanical Engineering*, 36:117-128, 2012.
[31] S. Spangelo, J. Cutler, K. Gilson, A. Cohn, "Optimization-based scheduling for the single-satellite, multi-ground station communication problem," *Computers & Operations Research*, 57:1-16, 2015.
[32] J. T. Hwang, D. Y. Lee, J. W. Cutler, J. R. R. A. Martins, "Large-Scale Multidisciplinary Optimization of a Small Satellites Design and Operation," *Journal of Spacecraft and Rockets*, 51:1648-1663, 2014.
[33] P. T. Boggs, J. W. Tolle, "Sequential Quadratic Programming," *Acta Numerica*, 4:1-51, 1995.
[34] K. Deb, "Multi-Objective Optimization using Evolutionary Algorithms," *John Wiley & Sons, Inc*., 2010.
[35] L. J. Eshelman, J. D. Schaffer, "Real-coded genetic algorithms and interval-schemata," *In Foundations of Genetic Algorithms 2 (FOGA-2)*, 187-202, 1993.
[36] Z. Michalewicz, "Genetic Algorithms + Data Structures = Evolution Programs," *Springer-Verlag*, Berlin.
[37] H. Kalita, J. Thangavelautham, "Automated Multidisciplinary Design and Control of Hopping Robots for Exploration of Extreme Environments on the Moon and Mars," *70th International Astronautical Congress (IAC),* Washington D.C., USA, 2019, 21-25 October.
[38] H. Kalita, A. S. Gholap, J. Thangavelautham, "Dynamics and Control of a Hopping Robot for Extreme Environment Exploration on the Moon and Mars," *IEEE Aerospace Conference*, Big Sky, USA, 2020, 7-14 March.
[39] H. Kalita, T. M. Jameson, G. Stancu, J. Thangavelautham, "Design and Analysis of a Mechanical Hopping Mechanism Suited for Exploring Low-gravity Environments," *IEEE Aerospace Conference*, Big Sky, USA, 2020.
[40] J. Thangavelautham, M. S. Robinson, A. Taits, et al., "Flying, hopping Pit-Bots for cave and lava tube exploration on the Moon and Mars" *2nd International Workshop on Instrumentation for Planetary Missions*, NASA Goddard, 2014.
[41] H. Kalita, A. Ravindran, S. Morad, J. Thangavelautham, "Path Planning and Navigation Inside Off-World Lava Tubes and Caves," *IEEE/ION PLANS Conference*, 2018.
[42] H. Kalita, J. Thangavelautham, "Multirobot Cliff Climbing on Low-Gravity Environments," *11th NASA/ESA Conference on Adaptive Hardware and Systems*, Pasadena, USA, 2017, 24-27 July.