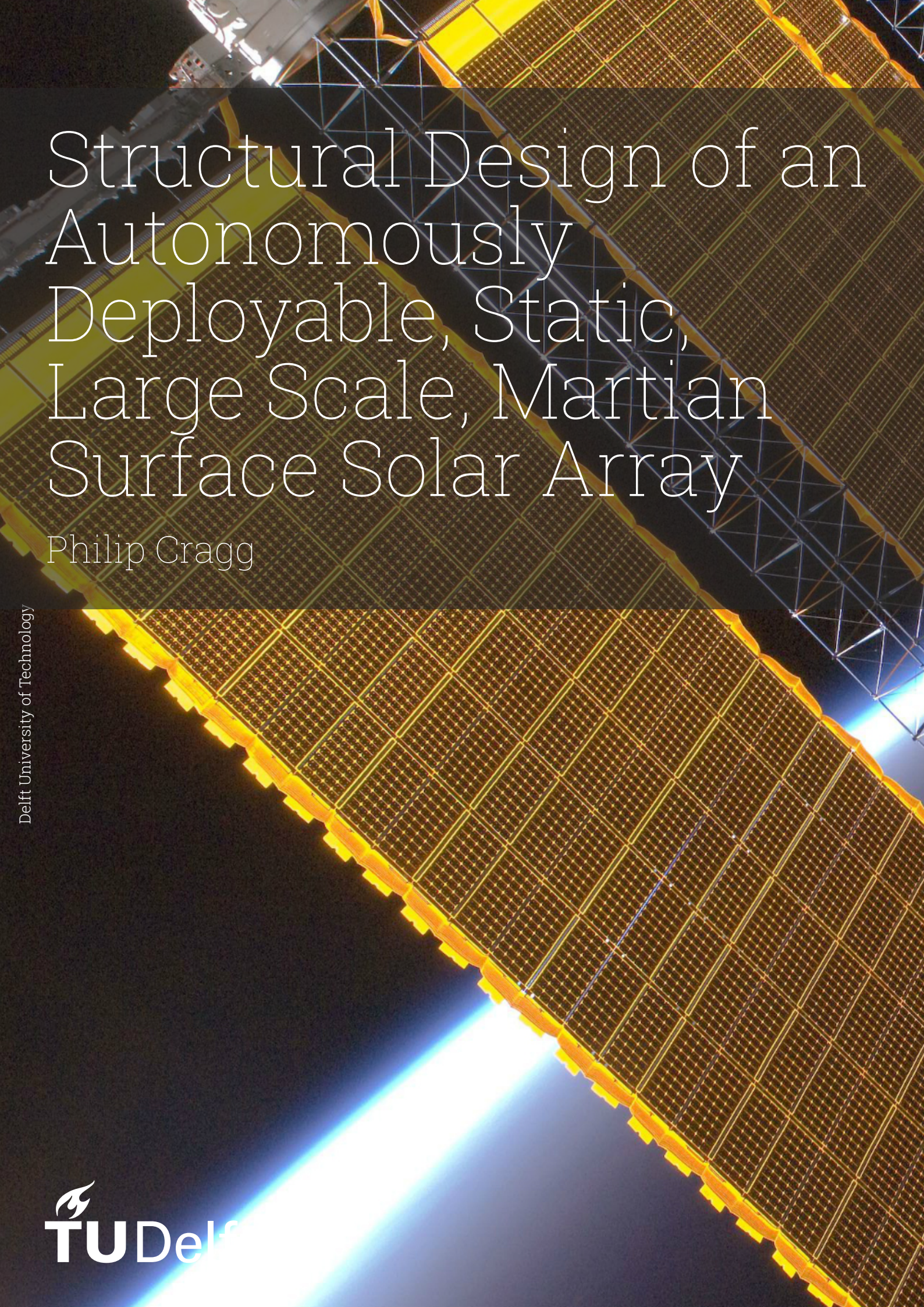


Structural Design of an Autonomously Deployable, Static, Large Scale, Martian Surface Solar Array

Philip Cragg

Delft University of Technology

Structural Design of an Autonomously Deployable, Static, Large Scale, Martian Surface Solar Array

by

Philip Cragg

Student Name	Student Number
Philip Cragg	5063396

Project Duration: Feb, 2022 - Jan, 2023
Faculty: Faculty of Aerospace Engineering, Delft

Cover: International Space Station solar array wing by NASA

TU Delft

Preface

This report details the thesis work carried out as part of my final evaluation for my MSc in Aerospace Engineering. It was a challenging but rewarding process that wouldn't have been possible without the support of a number of people. First and foremost, I would like to thank my advisor Otto Bergsma for his unwavering support and guidance throughout entirety of the thesis process. Despite me choosing an extremely niche topic well outside his scope of research, Otto's insights and analyses were truly invaluable. I would also really like to thank Richard Pappa at NASA Langley Research Center for providing me with excellent resources to kickstart my thesis as well as general knowledge and advice with regard to this challenging topic. Additionally, I would like thank a few other folks at NASA, namely Marc Schultz, Joel Schwartz, and Kauser Imtiaz as well as Curt Larsen at Blue Origin for the valuable discussions that were had over the course my thesis. Here at TU Delft, I would also like to extend my severe gratitude to Daniël Peeters and Dr. Saullo Giovani Pereira Castro for their assistance and insights. From the in-person discussions to lengthy email exchanges, I really do appreciate the support. Also, I'd like to thank my friends and family for supporting me through the ups and downs apparent throughout the duration of this thesis, especially Ro and Diego. Lastly, I'd like to thank Elena in particular for her calming presence and unwavering support during this challenging chapter of my life. I am extremely grateful.

Philip Cragg
Delft, March 2023

Summary

Presently, there exists no “off-the-shelf” option for power generation on the order of magnitude necessary for sustaining a long-term human presence on the surface of Mars. In this thesis work, an investigation was performed to determine the optimal design for a large-scale, static, autonomously deployable, lightweight Martian surface solar array. A principle aim of the design study was to articulate a concept whose stowage volume and mass performance is, at the very least, comparable to that of the Compact Telescoping Surface Array (CTSA) which was developed by the National Aeronautics and Space Administration (NASA) in 2016 to address the same challenge. To narrow the scope of the research, the study only considered solar array applications near the equator of Mars, where sun tracking mechanisms are less advantageous. A rigorous literature review was performed to identify existing applications of large scale solar arrays in space as well as the typical packaging, deployment, and stabilization mechanisms which accompany these systems. A conceptual design phase was pursued thereafter, resulting in the generation of numerous concepts all of which were described in detail on the basis of their characteristic deployable mechanisms, dust mitigation method, and apparent advantages and disadvantages. Derived evaluation metrics in conjunction with the Analytical Hierarchy Process (AHP) were used to evaluate concept performance and down selection. Thereafter, the down-selected concept, dubbed the “Stripped Array”, was subjected to a number of preliminary sensitivity studies. Specifically, system response to changes in total array area, level of pretension, and sizing of various subcomponents of the solar array membrane was evaluated. Furthermore, a preliminary study was performed to understand the effect of solar array membrane pre-tension, level of gravitational loading, supporting architecture cross-sectional sizing, and assumed support conditions on assembly stresses, deflections and natural frequencies.

Contents

List of Figures

List of Tables

Nomenclature

Abbreviations

Abbreviation	Definition
ADEPT	Adaptable Entry and Placement Technology
AHP	Analytical Hierarchy Process
AM0	Air Mass Zero
API	Application Programming Interface
CTA	Compact Telescoping Array
CTSA	Compact Telescoping Surface Array
DDS	Deployable Space Systems
DMTs	Dust Mitigation Technologies
DoE	Department of Energy
DOF	Degree of Freedom
EB	Euler Buckling
EDL	Entry Descent and Landing
GCR	Galactic Cosmic Radiation
HIAD	Hyper-sonic Inflatable Aerodynamic Decelerator
IMBA	Modular photovoltaic Blanket Assembly
IMM	Inverted Metamorphic Multi-junction
ISRU	In-Situ Resource Utilization
ISS	International Space Station
JAXA	Japan Aerospace Exploration Agency
JPL	Jet Propulsion Laboratory
JSC	Adaptable Entry and Placement Technology
KRUSTY	Kilopower Reactor Using Stirling Technology
LB	Local Buckling
LSB	Local Shell Buckling
LWD	Length-Wise Deployment
MAV	Mars Ascent Vehicle
MERs	Mars Exploration Rovers
MMA	Multi-mission Modular Array
NASA	National Aeronautics and Space Administration
PDE	Payload Dynamic Envelope
RCTSA	Revised Compact Telescoping Surface Array
ROSA	Roll-Out Solar Array
RSA	Relocatable Solar Array
SASME	Solar Array Subject Matter Expert group
SAWS	Solar Arrays With Storage
SBIR	Small Business Innovative Research
SLS	Space Launch System
STEM	Storable Extendible Tubular Member
SU	Scissor Unit
TRL	Technological Readiness Level
TS	Telescoping Section
UHMPE	Ultra High Molecular Weight Polyethylene
WWD	Width-Wise Deployment

1

Introduction

In 2010, the National Aeronautics and Space Administration (NASA) received direction from Congress to work towards the long-term goal of human exploration of Mars [1]. Many activities pursued within the agency since then have been in support of this mission, namely the development of the Space Launch System (SLS), a super-heavy lift launch vehicle designed for manned space missions beyond Earth's orbit, and the Artemis Program, NASA's mission to establish a sustained human presence on the Moon. A principal driver of the agency's near-term goal of establishing a permanent human presence on the Moon through the Artemis program is to acquire learning that will support the human exploration of Mars. As the former NASA administrator Jim Bridenstine stated, "…we will use the Moon as the stepping stone for our next greatest leap — human exploration of Mars" [2]. Just a couple years ago, the aforementioned NASA directive was reaffirmed through the NASA Authorization Act of 2020 [3]. Figure 1.1 shows concept art of a manned mission on the surface of Mars.

Figure 1.1: Concept art of a crewed mission on the surface of Mars
Courtesy NASA/JPL-Caltech

Among the myriad of new problems associated with putting an astronaut on another planet, the question of how to sustain and support human life once on the surface is something NASA has been

tackling for some time. One of the key issues of this challenge is power generation. That is, how power will be provided to critical mission hardware like life support systems and scientific equipment. Two of the leading contenders being investigated are fission reactors and large-scale, deployable solar paneled architectures. In 2015, NASA along with partners at the Department of Energy (DoE) began work on the Kilopower Reactor Using Stirling Technology (KRUSTY) which involved the testing of a fission reactor capable of supplying up to 10 kilowatts of continuous electrical power [4]. The experiment, completed in 2018, was a success and prompted the development of a lunar reactor intended for demonstration on the Moon surface in the late 2020's [5]. Concurrently, NASA has dedicated considerable resources to investigating the feasibility of using large solar array structures as well. Once considered volume and mass inefficient on a large scale, advances in lightweight deployable structures beckoned the agency to re-evaluate the feasibility of solar power as an option. In 2016, researchers from NASA Glenn and NASA Langley research centers performed a yearlong "Seedling Study" involving the preliminary design of an "electrical power architecture for human exploration of Mars using deployable solar arrays and regenerative fuel cells with 10kW modules" [6]. Moreover, NASA awarded several Small Business Innovative Research (SBIR) contracts to a handful of US companies in 2017 and 2018 to develop concepts for large deployable solar array structures for use on Mars [7]. Lastly, the agency also solicited the academic community in 2018 through the "Big Idea Competition" to design novel deployable solar array concepts for use on Mars. A trade study between both options was performed by NASA back in 2016 which assessed the relative merits of each based on application in two different conceptual missions. The underlining conclusion of this study was that both options are feasible but are highly dependent on many factors such as chosen landing site, program budget, technology investment strategies, etc..[8]. In short, the motivation to send humans to Mars endures and solar power as the principal power system for supporting this mission remains an option.

Mars presents numerous challenges for power generation, however, many of which are exclusive to photovoltaic systems. Saltating dust on the surface and dust suspended in the atmosphere, for example, alters the mechanisms for which light pervades the surface and can cover solar cells thereby retarding their capability to harvest energy. Furthermore, Mars inherently receives less solar flux than Earth given its station in the solar system. These realities, among others, impose several requirements on Martian solar array systems which creates a uniquely challenging design problem. One of the most prominent of these requirements is sizing of the array, where an enormous surface area of solar cells is likely required to harvest sufficient levels of power to support a crew on the ground. Current estimates place the necessary photovoltaic area somewhere in the range of $100m^2$ to $10,000m^2$ [6]. This is a monumental challenge, especially since heritage surface solar arrays on Mars have not exceeded at most more than approximately $7m^2$. For context, Appendix A provides a brief summary and tabulated list of past and present applications of photovoltaic systems on Mars. In the end, while a variety of solar array designs in past have been formulated that address these challenges, there still exists ample design space for improvement and refinement, particularly as NASA mission objectives evolve and improvements in knowledge about Mars both alter requirements with time.

The foundation of this thesis work is based on the aforementioned seedling study which resulted in the preliminary design of a lightweight, large-scale Martian surface solar array called the Compact Telescoping Surface Array (CTSA) [9]. One of the principal objectives of this study was to develop a surface solar power concept that was as versatile as possible. That is, priority was attributed to a design which could readily be adapted to different landing locations and is scalable as well as re-configurable for sun-tracking purposes. This was motivated by the fact that no firm mission plans existed for a manned mission to Mars. To date, this uncertainty endures as human exploration efforts have been largely concentrated on the Artemis program instead. Contrary to this existing research, the work detailed in this thesis takes a narrower approach to Martian surface solar array design. Specifically, focus was directed towards developing a concept which is optimized for a near equatorial landing site where sun-tracking systems yield less of a benefit. As a consequence, static solar arrays, or ones that do not rotate to maintain optimum orientation in relation to the sun, were the primary focus. Despite this restriction, this research remains valuable as a majority of potential sites for human exploration on Mars are within plus or minus 30 degrees latitude range meaning static solar arrays are very applicable[10]. At higher and higher latitudes, inclination of the array and the magnitude thereof becomes more and more of a requirement in order to have a reasonable photovoltaic efficiency. Lastly, static solar arrays

on or near equatorial latitudes were the preferred subject of this study based on the difficulty of quantifying the efficiency of any particular sun-tracking array concept. As will be seen later, Martian orbital characteristics coupled with a particularly complex environment produces a tough application for solar arrays to operate in. To that end, acquiring power generation estimates for solar arrays on the Martian surface is very difficult, especially without access to existing NASA codes and models built to do this.

With that said, the primary objective of the research detailed henceforth was to devise and characterize a large scale, static, autonomously deployable, lightweight solar array for use on the surface of Mars. The goal here was to come up with a concept which was ,at the very least, comparable to that of the CTSA. Furthermore, an additional goal was to characterize this developed concept further through various sensitivity studies. Doing so would provide meaningful learning about the feasibility of the concept as well as of large scale solar array structures on Mars in general and serve as apt starting point for future detailed study should that be pursued.

To address these goals and objectives, the following research questions for this thesis work were formulated and investigated:

- What is the optimal design for a large-scale, static, autonomously deployable, lightweight Martian surface solar array for use at near equatorial latitudes?

To properly address this question, a number of secondary supporting questions were formulated. For the conceptual design phase:

- Is there a deployable solar array concept that achieves better stowage volume and mass performance than that of the Compact Telescoping Surface Array (CTSA) developed by NASA?
- Should several be identified, how might each compare against one another based on these metrics in addition to concept risk?

For the down-selected concept:

- How does the total array area and applied membrane pre-tension load affect the distribution of pre-stress throughout the membrane sub-assembly?
- How does the pre-tensioning scheme perform in response to changes in sizing of various components of the solar array membrane sub-assembly?
- How does level of pretension of the solar array membrane, magnitude of gravitational loading, and alterations to the supporting architecture's cross-sectional sizing and assumed support conditions affect assembly stresses, deflections, and natural frequencies?

The layout of remainder of this report is as follows. First, a short overview of characteristics of Mars is provided. Next, a thorough review of deployable solar array structures is presented. Thereafter, the conceptual design work performed in this thesis is presented, going into detail with regard to the assumptions and guidelines of the process, the concepts generated, and the down-selected concept. Lastly, additional sensitivity studies are performed on the down-selected design followed by a short section detailing the conclusions of this work and what should be done next.

2

Mars

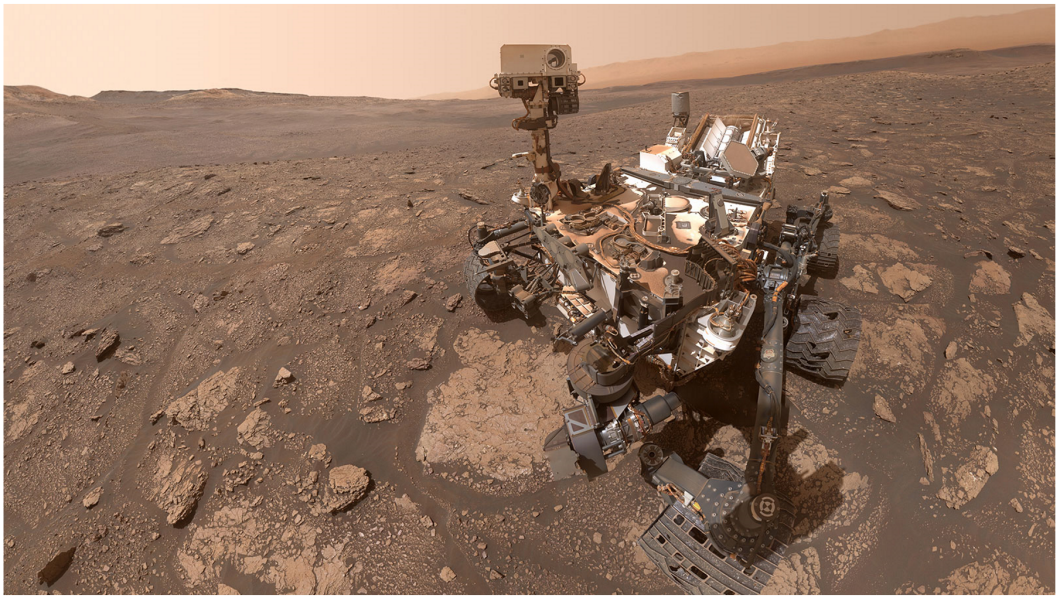


Figure 2.1: NASA's Curiousity Rover in the Glen Torridon region on Mars
Courtesy of NASA/JPL-Caltech/MSSS

Prior to reviewing the current state-of-the-art of deployable solar array technologies, it is important to first examine in detail the characteristics of Mars. Understanding these traits provides context not only for the design process hereafter but also for the discretionary literature research presented in the proceeding chapter. Thus, a brief overview of Mars is provided here that covers the following relevant aspects:

- Orbital Mechanics
- Environment
- Atmosphere and Radiation
- Temperature
- Dust and Wind
- Surface Topology

2.1. Orbital Mechanics

First and foremost, Mars is positioned further away from the Sun than Earth resulting in a significant decrease in solar flux by comparison. The magnitude of this reduction is not constant but rather varies

throughout the Martian year as a consequence of its eccentric orbit around the sun. That said, the average incident solar flux at the top of the Martian atmosphere is approximately $590W/m^2$ which equates to approximately 43% of sunlight Earth receives [11]. The actual intensity varies by plus or minus 19% throughout the Martian year[12]. This unfortunate reality is a requirement driver in Martian solar array design as it directly influences the necessary area coverage to achieve a given level of power generation. It should be reiterated that these values correspond to solar fluxes at the top of the Martian atmosphere, not the surface. The importance of this distinction that will become apparent later on. Mars also has an obliquity, or axial tilt, of approximately 25 degrees with respect to its orbital axis around the Sun meaning the planet has seasons. By comparison, Earth's tilt is approximately 23.4 degrees. Nonetheless, Martian seasons are of increased duration in contrast given the fact that it is further away from the Sun and are uneven in length due to the eccentricity of its orbit. The sun's movement in the Martian sky in reference to a position on the surface, known as solar zenith angle, thus changes with latitudinal position and time of year. As one can imagine, this feature is also very important in establishing solar array operational as well as area requirements. As a final note, the gravitational force on Mars is roughly 3/8th of Earth's gravity.

2.2. Environment

Atmosphere Mars has an atmosphere that is approximately 100 times thinner than of that of Earth's. Atmospheric density at the planet's surface is equivalent to Earth's at an altitude of 30 kilometers. This has many wide-reaching implications, particularly in terms of Entry, Descent, and Landing (EDL) as well as how the planet interacts with meteors and various forms of radiation. Complicating EDL, Mars' thin atmosphere is thick enough to significantly heat an entry vehicle but not enough to substantially reduce its speed [13]. This reality, coupled with restrictions in existing EDL technologies, manifests strict mass and volume allowances for landing anything on the surface of Mars. For context, the car-sized Perseverance rover is the largest and heaviest vehicle to land on the planet to date weighing at 1026 kg.

Mars' atmosphere is fortunately thick enough to provide effective protection from meteors and offers sufficient shielding from solar cosmic rays which are high-energy particles ejected from the Sun [14]. While Galactic Cosmic Radiation (GCR), extragalactic rays, are not attenuated by the Martian atmosphere, the accumulation of dosage in solar cells is low enough to be considered negligible. Risk of degradation in solar cell performance from either of these factors is thus low. Because Mars has no considerable ozone, a larger portion of UV radiation penetrates to the Martian surface than does on Earth albeit the abundance of carbon dioxide in the atmosphere absorbs most wavelengths below 200 nm [15]. That said, for long duration applications on Mars such as is the case for a surface solar array, degradation of material properties is a risk should soft goods be utilized. Proper material selection or protection systems must be in place.

Temperature Similar to that of Earth, the Martian climate is a complex system and has been studied extensively over the past several decades. While reporting all of its intricacies is beyond the scope of this thesis, near-surface temperatures are perhaps the most pertinent when it comes to the preliminary design of a Mars surface solar array. After all, temperature data is important because they influence material selection as well as solar cell performance. Daily mean air temperatures and diurnal amplitudes, which is the difference between maximum and minimum temperature during a day, have been collected across the several Martian years for multiple different landers and rovers. According to the data, the daily mean air temperature changes with solar longitude and ranges between approximately 160K to 235K depending on the latitude, season, and outside influences like global dust storms [16]. Furthermore, the diurnal amplitudes are modest, ranging between approximately 5K and 80K. Failure by thermo-mechanical fatigue of structural materials is consequently as risk.

Dust and Wind One of the most influential aspects of the Martian Environment is the abundance of dust. Per observations by the Spirit and Opportunity rovers, Martian dust can be categorized into three distinct categories: Atmospheric dust (1-2μm radius), Settled dust (>10μm radius), and Saltating dust (>80 μm radius) [17]. Aptly named, atmospheric dust pervades the atmosphere, remains suspended for long durations of time, and ,as demonstrated by measurements taken by the curiosity rover, their size can exceed 4-5 μm in radius during global dust storms [18]. The amount of these aerosols,

measured in terms of optical depth τ (tau), is quantified by the observed optical opacity of the Martian atmosphere, varies depending on the season, latitude, as well as presence of dust storms, and ranges in value between 0.4 and greater than 4 [11, 16, 19]. Optical depth matters because it produces a shift from the Air Mass Zero (AM0) spectrum, which is the spectrum of light present in outer space, as well as reduces the overall intensity of the light reaching the surface [13]. Specifically, airborne dust absorbs and scatters light causing this spectral shift and attenuation. Naturally, these phenomena are important because the shift implies heritage solar cells optimized for AM0 cannot be used on Mars without operating at a deficiency. Likewise, a reduction in incident light at the surface further influences how much solar array area is needed. On top of that, because the optical depth is not constant, the incident spectrum is inherently dynamic thus introducing more complexity into the process of designing an optimal surface solar array system.

Settled dust particles, on the other hand, are those which are raised into the atmosphere by a variety of different mechanisms but ultimately redeposit on the surface given their larger size. Saltating particulate, as the name suggests, move primarily by means of saltation. Together, these types of particles pose a risk to surface solar arrays as they are wont to accumulate and cover surface of the solar cells thereby degrading its current output. Moreover, high speed winds carrying these particles raises concerns of scratching of the solar cell surface lead an increase in surface reflectance and a corresponding decrease in generated power. In effect, these forms of dust influence the operational and structural requirements of the solar cells themselves, where a system mitigating dust collection and resisting abrasion is needed.

To underscore the importance of dust accumulation in the context of surface solar arrays, every lander or rover to have ever touched down on the surface of Mars that utilized solar power has been affected in some way or another[14]. Despite persisting well past its designed mission life, the Opportunity rover ultimately failed due to dust deposition after a large storm [15]. Furthermore, after a loss of mobility caused by a wheel failure, the Spirit rover slowly accumulated dust over the course of several Martian winters till power generation subsequently ceased and the mission lost [20]. The longevity of each of these missions is partially credited to what has been referred to as “cleaning events”, or random gust of wind that blow away accumulated dust on solar panels. More recently, the 2018 Insight Lander was forced into an unplanned hibernation because a lack of cleaning events resulted in too much of a dust build up. Figure 2.2 shows a side-by-side comparison of one of Insight’s Ultraflex arrays shortly after landing in 2018 and later in 2021.

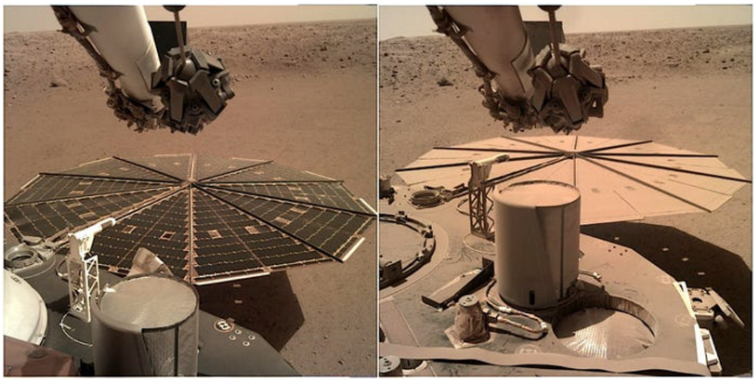


Figure 2.2: One of Insight’s Ultraflex arrays after landing in 2018 (Left) and the same array covered in dust in 2021 (Right) Courtesy NASA/JPL-Caltech

On the subject of wind, speeds on the surface are variable and depend largely on a confluence of factors such as geographic location, season, and the presence of localized (dust devils) and global dust storms (see Figure 2.3). A wealth of wind data has been collected by various rovers and landers in the past but uncertainty remains. Historical data is spread across different geographic positions and taken at different times making it unreasonable to assume this data is globally representative of weather trends. Moreover, some data were collected in bins of 1-5 minute averages making dynamic simulation of wind conditions not possible. Nonetheless, the current understanding is that horizontal winds can reach as high as 100 m/s over short time periods courtesy of dust devils [21]. These meteorological phenomena also have a vertical wind component that is not well characterized. Somewhat frequent

global dust storms bring sustained and elevated winds albeit lower than the larger speeds observed in dust devils. Because the Martian atmosphere is significantly less dense than Earth's, the imparted aerodynamic load on a surface structure is not as big. For context, the load imparted by a 100 m/s peak wind on Mars under max atmospheric density conditions is equivalent to approximately of a 9 m/s wind speed on Earth.

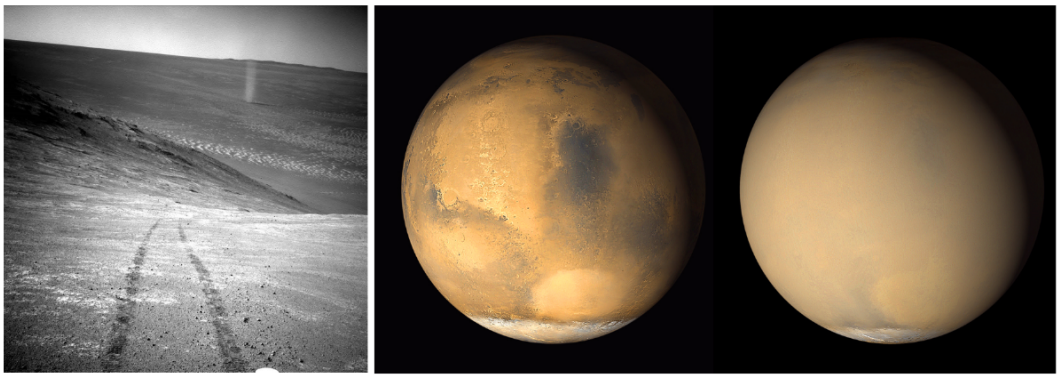


Figure 2.3: A dust devil captured by the Opportunity Rover in 2016 (Left) and back-to-back view of Mars with and without a global dust storm in 2001 captured by the Mars Orbiter (Right)
Courtesy NASA/JPL-Caltech

Surface Topology Mars' diameter is about half the size of Earth's, where its southern hemisphere is marked by highlands and craters whereas the northern hemisphere consists largely of lower elevation plains. Topology of the Martian surface is important to surface solar array design because it influences operational requirements of the system and well as alters the levels of albedo light which has solar cell efficiency implications. Depending on the location, an array may need to navigate rocks, depressions, holes filled with loose sand, and traverse modest slopes during deployment. The landing locations of the Sojourner rover as well as the two Viking landers showed a landscape characterized by fine dust and rocks occupying approximately 8-16 percent of the surrounding area[22].

3

Deployable Solar Array Structures

In this section, a non-comprehensive overview of solar array deployable structures in space applications is presented. First, past and present applications of deployable solar arrays on the surface of Mars as well as the greater outer space are described. These examples extend not only to real applications but also include conceptual designs. More generally, state-of-the-art mechanisms for deployable space structures are then reviewed, followed by an exploration of typical materials used in space and on the surface of Mars.

Prior to reviewing the current state of large-scale deployable solar array structures in space, however, it is important to first acquire perspective in regard to launch architectures. Limitations in launch vehicle payload fairing sizes as well as their restrictive mass budgets have long been a motivating factor for the integration of lightweight deployable systems onboard space bound hardware. Without these systems, large scale architectures in space are simply less feasible. Table 3.1 shows a list of launch vehicles and their corresponding maximum internal fairing diameter available to a given payload. This list is by no means exhaustive but gives context for the necessity of package-able/deployable systems for systems larger than a handful of meters in diameter. In regard to future launch capacity, the rocket with the largest payload fairing currently under development is SpaceX's Starship with an outer diameter and internal usable Payload Dynamic Envelope (PDE) of 9m and 8m, respectively [23]. In comparison, NASA's Space Launch System (SLS), which is also under development, has an outer diameter and PDE of 8.4m and 7.5m, respectively [24]. However, this particular design has a planned upgrade to a 10m fairing in the future which expands the PDE to 9.1m. It is important to note that both of these launch vehicles are planned to support manned and unmanned missions to Mars in the future.

Table 3.1: A list of launch vehicles versus their maximum internal fairing diameter available to a payload [23, 24, 25, 26, 27, 28, 29]

Launch Architecture	Maximum Fairing Payload Internal Diameter (m)
Vega	2.42
Soyuz	3.80
Ariane 5 ECA	4.57
Atlas V-541	4.57
Falcon 9	4.60
Falcon Heavy	4.60
SLS	7.50
Starship	8.00
SLS (10m fairing Variant)	9.10

With the pressing need for deployable systems for large scale structures in space understood, the

field of deployable structures itself is expansive and has experienced an explosion of innovation the past few decades. So much so that researchers have struggled to keep up and articulate an adequate system of categorization [21]. Nonetheless, for the purposes of this report, deployable systems in space applications can be broadly partitioned into three different categories: Rigid, flexible, and hybrid. Rigid deployable systems involve structures in which are inherently stiff, requiring no post-deployment rigidization mechanism for maintaining shape and tolerating loads. An example of this sort of system would be a foldable solar array comprised of discrete, sandwich paneled sections. Flexible deployables are on the other end of the spectrum, involving structures that must be stiffened after deployment by some sort of mechanism in order to hold shape and bear loads. A prime example of this is an inflatable system which relies on pressurization to articulate its form or a ultra-thin membraned solar sail. Unsurprisingly, hybrid deployable structures are a blend of the two systems, incorporating both rigid and flexible elements. As will be presented later, the solar arrays on the International Space Station (ISS) are an excellent example of this.

The primary focus of the remainder of this section will be on hybrid deployable solar array structures. That is, systems that utilize thin solar array membranes in conjunction with rigid supporting elements. This narrowing the field of research was performed due to a variety of reasons. Firstly, rigid deployables typically have much lower packaging efficiencies as their designs are scaled which is problematic in this particular application. Furthermore, flexible deployable structures are more applicable in environments that have less severe loading like in outer space where gravitational forces are low. As was presented earlier, its reasonable to postulate such structures are not applicable to the Martian environment where gravity and wind loads are very consequential. While inflatable deployables have excellent packaging efficiencies and could probably be designed to weather the Martian environment, these systems were omitted as well on the basis that they require constant pressurization in order to function. Mass efficiency and longevity of these design are questionable, considering the parasitic mass associated with the pressurization equipment and feasibility of maintaining pressure for long surface mission operational lives. For some perspective, in the SAWS study, a preliminary design guideline was that any devised solar array design must service have a lifetime of at least 10 years.

3.1. Concepts and Missions

3.1.1. Space

The largest tensioned blanket solar arrays ever deployed are on board the International Space Station (ISS). Numbering eight in total, the solar arrays each have a blanket area of 2512 square meters and have the collective generation capacity of about 256 kW. The single junction solar cells integrated in these Z-folded blankets are 15 percent efficient, made of silicon, and are 8 by 8 centimeters [30]. The current state-of-the-art for large scale deployable tensioned blanket solar arrays for space applications include the Roll-Out Solar Array (ROSA) designed by Deployable Space Systems (DDS), now a part of Redwire Corporation, and the MegaFlex designed by Alliant Techsystems Inc. (ATK), now a part of Northrop Grumman. Another of notoriety is Lockheed Martin's fourth generation flex array, the Multi-Mission Modular Array (MMA).

ROSA, iROSA, and Mega-ROSA The ROSA is a very compact design consisting of a thin Integrated Modular photovoltaic Blanket Assembly (IMBA) flanked by elastic, coilable carbon fiber composite booms on either side [31, 32]. The stowed and deployed configurations are shown in Figure 3.1. The entire assembly is coiled around a central cylindrical member and deployment is driven by the stored elastic energy incurred during packaging in the aforementioned composite booms. Once deployed, the booms themselves provide the necessary structural rigidity to support the blanket solar array as well as maintain tension in the blanket during service. In effect, the booms act as both an actuator for deployment as well as a structural support. This duality provides a significant weight saving as it removes parasitic mass that would otherwise be present when using drive mechanisms like a motor. A spreader bar is attached to the tips of the booms thus joining them together and also serves as a point of connection for the solar cell blanket as well as a means to ensure its spread. The final deployed surface area is approximately 4.5 by 14 meters. In 2017, the ROSA successfully performed its first spaceflight

demonstration aboard the ISS. Complications occurred, however, when attempting to retract the array following the completion of the 7-day mission that ultimately required it to be jettisoned from the ISS for disposal instead of what was originally planned. In June of 2021, two larger versions of the ROSA, dubbed iROSA, were installed on the ISS successfully.

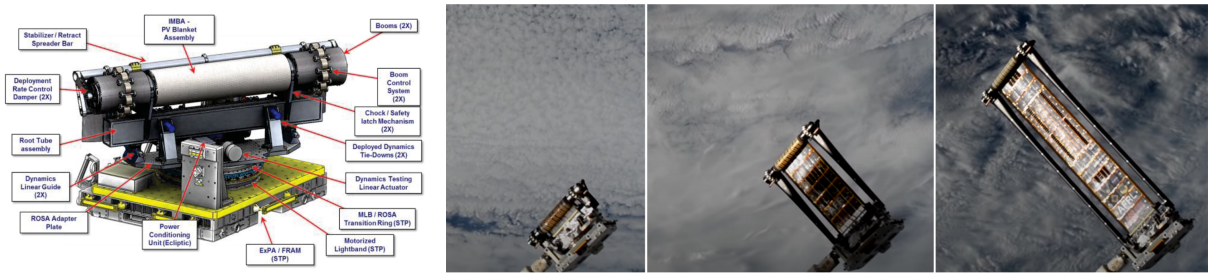
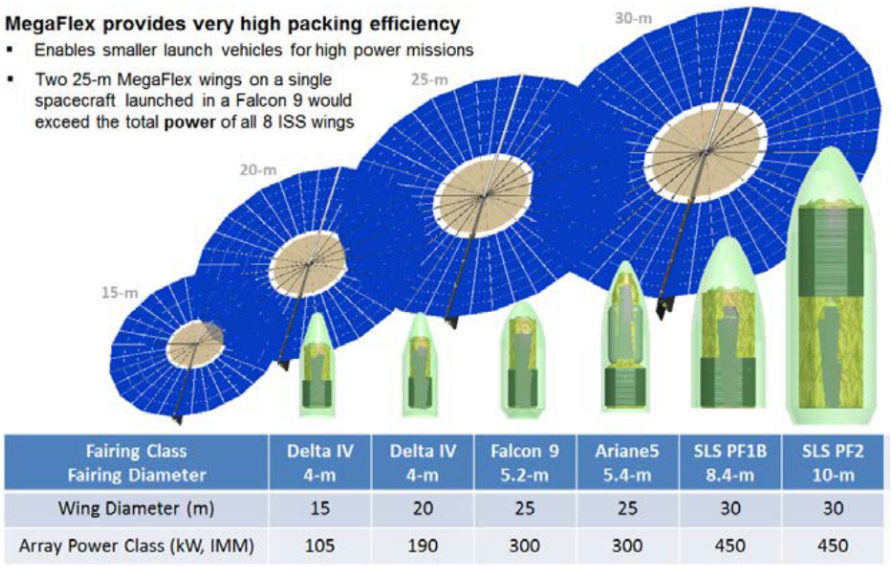


Figure 3.1: The Roll Out Solar Array (ROSA) in its stowed configuration (Left) [31].The ROSA during a deployment test on the ISS (right). Image Courteous of NASA Johnson Spaceflight Center

In terms of power output specifications, the solar array is scalable, allowing for 1kW to 30kW+ of power [33]. Moreover, Redwire corporation is developing a larger modular system that integrates multiple ROSA "winglets" into a mega structure capable of delivering even greater power levels of 20kW to 400kW+ [33]. Aptly named "Mega-ROSA", the structure consists of a deployable arm that is divided into four segments, each the length of a single ROSA winglet. Each arm is comprised to two ROSA winglets that deploy in opposite directions, bringing the assembly to a total of 8 solar array wings. Despite its size, the construction can be packaged compactly with decent form factors due to the fact that the arm segments can nest within each other. Demonstration of the Mega-ROSA concept has been limited to successful ground testing of the deployment mechanisms [34].

Megaflex The Megaflex, on the other hand, is another concept under development which offers a lightweight, high stiffness, and high strength deployable solar array construction akin to a "Shan", a Chinese folding fan [35]. This solar array is an evolution of the Ultraflex solar array technology which has significant flight heritage and possesses power level scalability of up to 450kW. See Figure 3.2. Structurally, the Megaflex array consists of two distinct groupings, the first being what is called the "Platform Assembly" and the other called the "Power Assembly". The platform assembly consists of two honeycomb composite panels as well as radial spars which support the solar array blanket when fully deployed. The power assembly consists of solar array blanket which is divided into segments dubbed "Gores". In reference to the 10m solar array variant, each Gore is further subdivided into 6 "Gorelets" which comprise of multiple strings of Inverted Metamorphic Multi-junction (IMM) solar cells attached to a gossamer woven fabric mesh. Larger versions have more Gorelets. Figure 3.3 provides an excellent visualization of the breakdown. A few key features of the design are the actuation method and compact stowage. In terms of actuation, the entire assembly is powered by the clever integration of a single, motor-driven lanyard which drives the extension of the Megaflex honeycomb extension panel and the unfurling of the gores and spars. This significantly reduces the system mass. The other feature of notoriety is the integration of the extension panel in the design. This extension panel allows for the aforementioned scalability without drastically affecting stowage volume because it can be folded as shown in Figure 3.4. For a more detailed description of the design, see the referenced technical paper [35]. The Megaflex has not been utilized in any mission as of yet but a ground test demonstration of 10m variant in 2014 was successfully performed [35, 36].

MMA Beginning development in 2013 and completed in 2017, the MMA is a flexible 23 meter long solar array assembly whose design is focused primarily on power modularity. That is, to meet the power requirements for a given mission, the MMA can be easily modified from its standard 12.5 kW capability to power levels such as 6kW, 8 kW, 10kW, etc [37]. The entire assembly itself consists of four separate sub-assemblies: Boom Assembly, Deployer Assembly, Blanket Container Assembly,



Fairing Class Fairing Diameter	Delta IV 4-m	Delta IV 4-m	Falcon 9 5.2-m	Ariane5 5.4-m	SLS PF1B 8.4-m	SLS PF2 10-m
Wing Diameter (m)	15	20	25	25	30	30
Array Power Class (kW, IMM)	105	190	300	300	450	450

Figure 3.2: A comparison courtesy of Northrop Grumman showing launch vehicle versus power class capability [35]

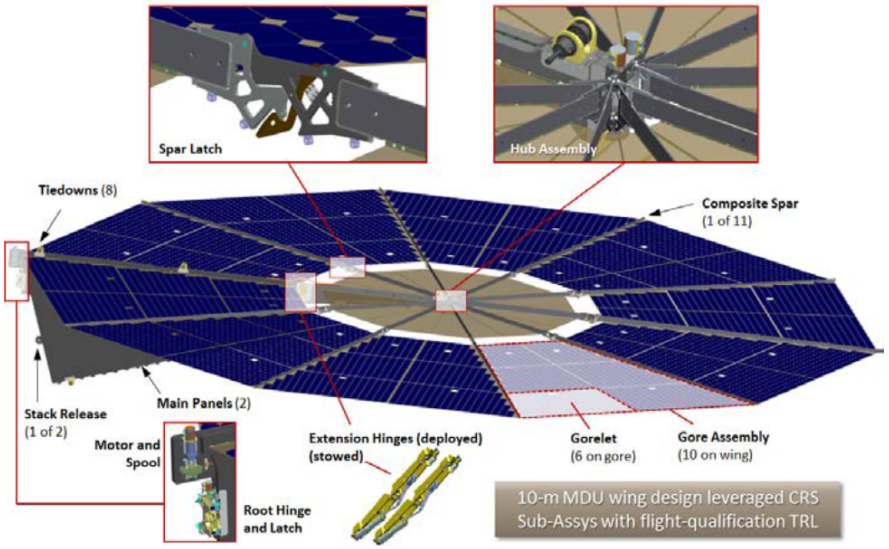


Figure 3.3: Structural breakdown of the Megaflex solar array [35]

and Blanket Assembly. The function of the Boom Assembly is to distance the array away from the spacecraft it is attached to as well as provide a routing path for electrical cabling. It is comprised of several interconnected hollow composite tubes with fittings at their ends made of titanium. The Deployer Assembly serves as a housing for a deployable composite mast, which drives the entire solar array deployment. The purpose of the Blanket Container Assembly is to, “protect the Blanket Assembly during launch, position the Blanket Assembly for deployment once on orbit, and provide tension to the blanket during its on-orbit life”. This sub-assembly includes sandwich panels with aluminum honeycomb and carbon-fiber face sheets as well as the array tensioning mechanisms. Lastly, the Blanket Assembly is packaged in a Z-folded configuration and is comprised of a thin Kapton film with strings of solar cells bonded to it. Figure 3.5 shows the following deployed MMA in a ground test. Figure 3.6 shows the deployment sequence.

A few notable characteristics of this particular large scale flexible solar array is the deployment methodology as well as the tensioning scheme. As mentioned previously, a single coiled composite

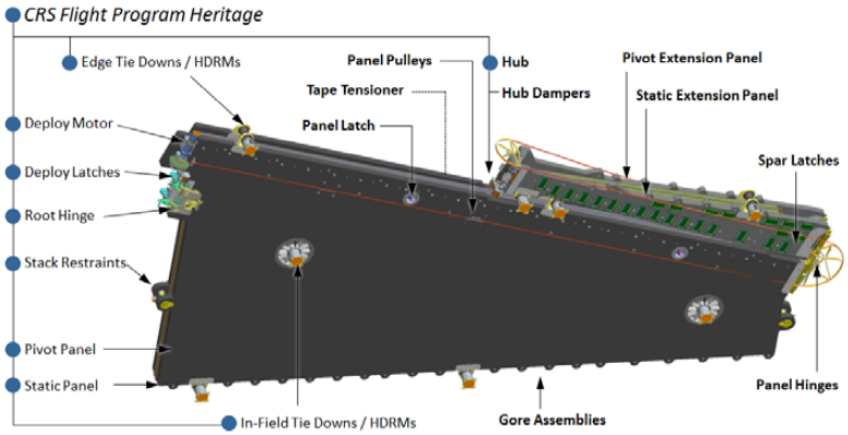


Figure 3.4: Stowed configuration of the Megaflex solar array [35]

Figure 3.5: The Lockheed Martin MMA fully deployed [37].

mast, positioned in the center of the array, is what drives the extension of the Z-folded solar array blanket. Actuating motors, once activated, allow for the mast to unfurl. The mast itself is made of two carbon fiber composite shells bonded together at the edges to form a lenticular cross-section which is advantageous for enduring both bending and torsional loadings [38]. This cross section is flat when it is coiled around a lightweight mandrel, which also allows for compact stowage. Furthermore, at the connections between the two joined shells, there are a plurality of grommets, see right side of Figure 3.7, which interface with the actuating motor during deployment. Specifically, the actuating motors have teeth which engage with composite mast, Figure 3.7, and allow for deployment and retraction.

The tensioning mechanism for this large solar array is also of interest. This mechanism consists of several constant force negator spring devices positioned circumferentially equidistance around a central spool, as shown in Figure 3.8 [37]. The spring devices are connected to the central spool and provide a rotation force on the spool when unraveled. A metallic cable is connected at one end to this spool, wound around it, and has a threaded bolt swaged at the other cable end. This threaded bolt is then connected to the most inboard panel of the Blanket assembly. A consequence of this setup is that when the cable is unwound from the spool, the negator springs are unfurled which results in restoring force that drives to pull the cable back. Before the composite mast nears the end of its deployment, the

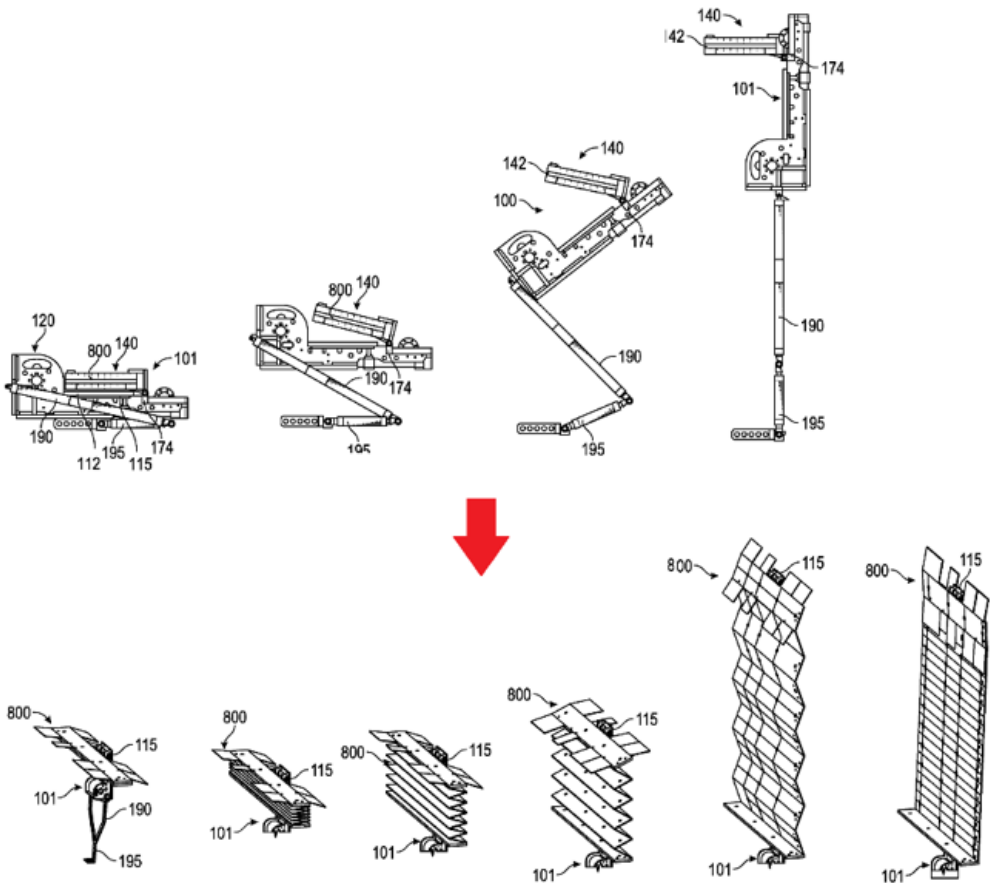


Figure 3.6: The deployment sequence for the MMA. The top is the deployment of the boom assembly and the bottom shows the deployment of the blanket array [38].

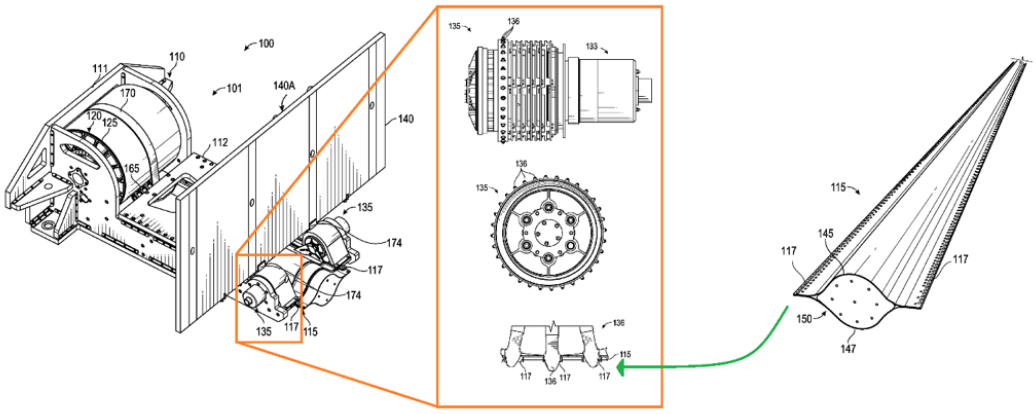


Figure 3.7: : This graphic shows how the actuation motors deploy the central composite mast that unfurls the solar array [38].

tensioning mechanism is engaged. This involves the controlled paying out of this cable which in turn applies the aforementioned constant spring force on the solar array blanket. This effectively tensions the array.

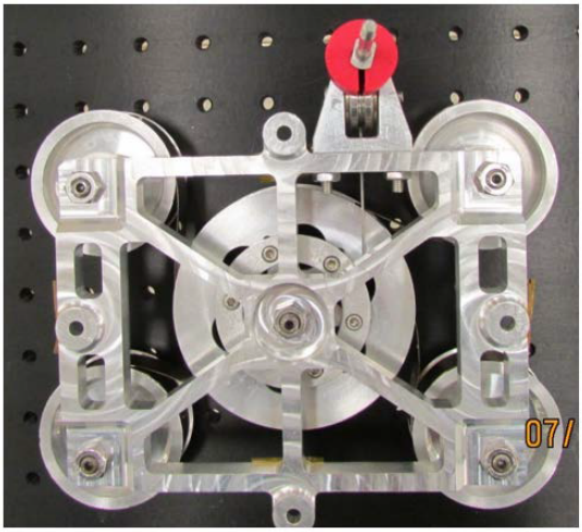


Figure 3.8: The tensioning mechanism of the MMA [37].

3.1.2. Surface

To date, applications of large scale deployable blanketed solar array structures are still limited to powering missions orbiting Earth. On the surface of another planet or another celestial body, like Mars or Earth's Moon respectively, applications are limited and are either small scale or entirely conceptual. Below, NASA's conceptual design of a large-scale deployable solar array structure will first be introduced. Following that, Ultraflex solar arrays will be presented, which constitutes the only known application of hybrid deployable solar array technology on the surface of Mars. Lastly, a few additional conceptual designs derived over the last decades for the Moon and Mars will be introduced.

Compact Telescoping Surface Array (CTSA)

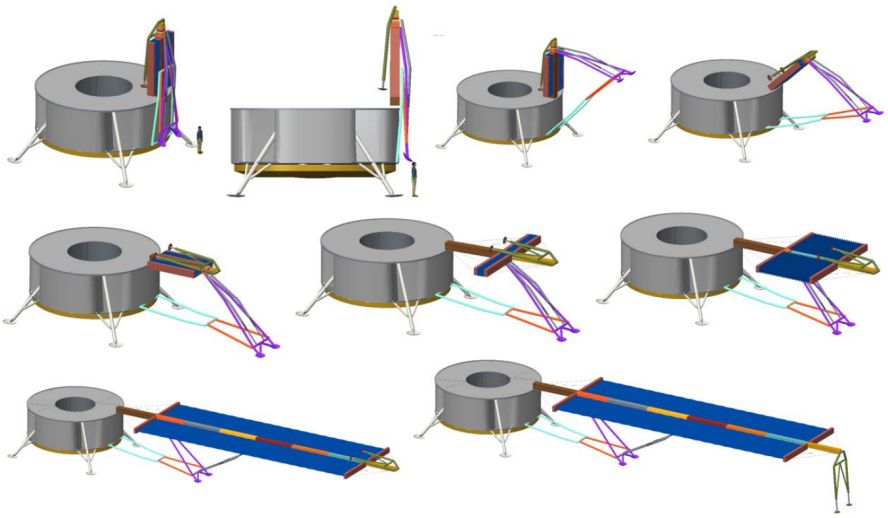


Figure 3.9: Baseline deployment concept for the CTSA. Note, the CTSA is stowed and deploying from a 10m class lander (9.1m diameter) [6]

As introduced earlier, the CTSA is a large-scale, lightweight, horizontally deployable solar array structure that was devised for use on the Martian surface to power an extended crewed mission there [9]. Shown in Figure 3.9, the solar array concept is capable of supplying 50-80kW and 10kW of power during Martian daytime and nighttime, respectively. Of course, nighttime power is courtesy of assumed

energy storage during the day. In terms of mass and volume performance, the combination of six CTSA wings occupies roughly $10m^3$ of space and weighs $1500kg$ for a $1000m^2$ total array area. Again, this preliminary concept is the product of the aforementioned seedling study called “Solar Arrays With Storage” (SAWS) which was performed by NASA at the end of 2016. Additionally, the CTSA design is a modification of a large scale, blanketed solar array concept with excellent mass and volumetric packaging efficiencies that was originally proposed in 2015 for powering vehicles in space called the Compact Telescoping Array (CTA) [39]. Modifications from the CTA were introduced because of the challenges that the Martian environment presents that are absent in space, namely a larger gravitational force, variable wind loading, and dust.

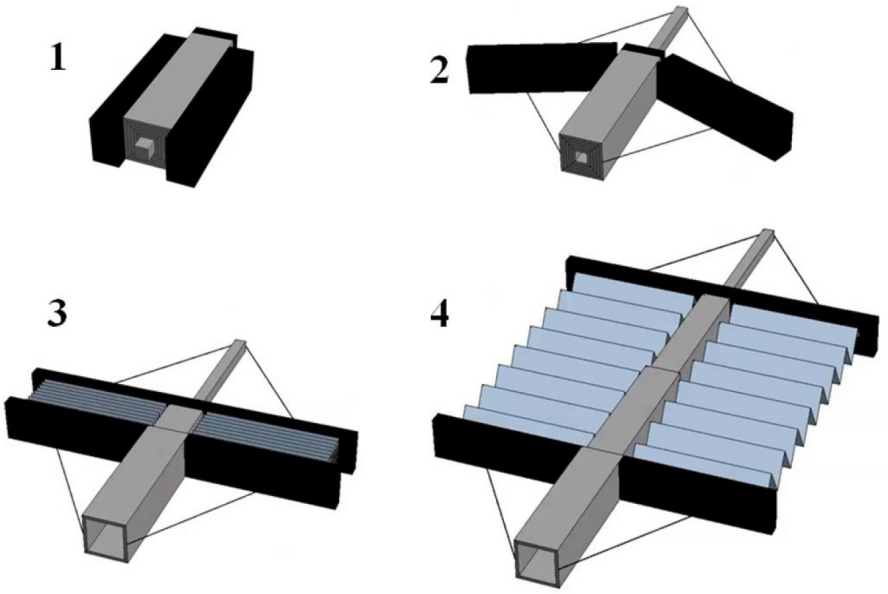


Figure 3.10: Conceptual deployment scheme of the CTA. Actuating systems are not shown. The CTSA uses the same principle of deployment [6].

A signature design feature of the CTSA concept that was carried over from the CTA is that it makes use of a single telescoping, hollow composite central beam, flanked by two expandable Z-folded solar array blankets (Figure 3.10). The characteristics of the telescoping member, namely the number of segments as well as their cross-sectional dimensions, can be easily changed without drastically affecting the overall design. One key design derivation from the CTA is that the CTSA includes a support structure necessary to combat the effects of gravity during and after deployment. Preliminary studies for this design showed that total mass of the entire assembly was sensitive to the support structure system used. Basically, the system mass could be reduced if it included a tip support at the end of the deployed array as shown in Figure 3.9. The central member dimensions could be reduced since the tip support helps reduce incurred bending loads. Another mass study showed that system mass was also sensitive to the assumed packaging volume. A larger packaging volume allowance means that the telescoping central member could have a larger but thinner cross-section. The larger the telescoping segment cross section, the more weight efficient it is in tolerating bending loads. In the event that the packaging volume is kept constant, the only way to improve the bending performance of the structure is to increase its cross-sectional thickness which is not as efficient.

More with regard to the design of the CTSA, different configurations were explored during the SAWS study for rotating the solar array blanket sub-component for the purposes of allowing the shedding of dust, sun tracking, and feathering the structure during high wind conditions. One involved separate actuation of each of the two flanking solar array blankets while the other involved simply rotating the entire horizontally deployable structure about its central telescoping boom.

Other Surface Applications

Figure 3.11: Image of the Insight lander with its two Ultraflex solar arrays. Photo courtesy of NASA/JPL-Caltech/Lockheed Martin

Ultraflex Ultraflex is fan-like hybrid deployable solar array structure, Figure 3.11, that has significant flight heritage in space and has been utilized on Mars to power NASA's Phoenix and Insight landers in the past. In fact, these types of arrays constitute the only known application of hybrid deployable solar array systems used on the surface of Mars to date. As mentioned previously, this solar array technology served as basis for the development of the Megaflex array. Consequently, the general structural characteristics and mechanisms of the Ultraflex array are largely the same as the Megalfex. A principle difference is the lack of the aforementioned extension panel apparent in the Megaflex design. Like the Megaflex, the structure consists of a plurality of triangular gores consisting of open polymer mesh for which solar cells are adhered to. Spars occupy the regions between adjacent gores and carry out-of-plane loads as well as compressive loads apparent in the structure the gores are pre-tensioned. The whole array unfurls from its stowed configuration, as shown in Figure 3.12, through a motor driven shortening of a lanyard tape.

Small Business Innovation Research (SBIR) Contract Studies As mentioned previously, NASA issued a solicitation in 2017 and again in 2018 to small U.S companies that sought "structural and mechanical innovations for solar arrays with at least $1000m^2$ of total area that autonomously deploy from Mars landers" [41]. Figure 3.13 shows concepts developed in 2017 and 2018. For the sake of report brevity, only a handful will be examined below.

The Self-Deploying Tent Array is as the name implies, a simple tent like structure. In the stowed configuration, the tent forms a cylindrical structure and can deploy outwards as shown in Figure 3.14. The design is fully retractable and features a static solar array concept (no sun tracking). Positioning the solar array blankets at an angle limits the decrease in power performance incurred by not tracking and reduces dust build up. In a NASA assessment, tracking systems can offer 5-15 percent power enhancement [42]. This however, would require additional actuation systems that add parasitic mass.

The Articulating Solar Panel Energy System is a derivation of the CTSA mentioned previously that includes interesting features such as dual axis feathering capability and replaceability. In regard to dual axis feathering, the design truncates the 22.5 m long CTSA blanket array into individual "cell blades", each 0.22m in length (Figure 3.15). This truncation improves the structure's resistance to wind loading and also makes solar cell maintenance and replaceability more tenable. In the case of the CTSA, damage to the solar array blanket could mean replacing the entire structure or losing entire strings of

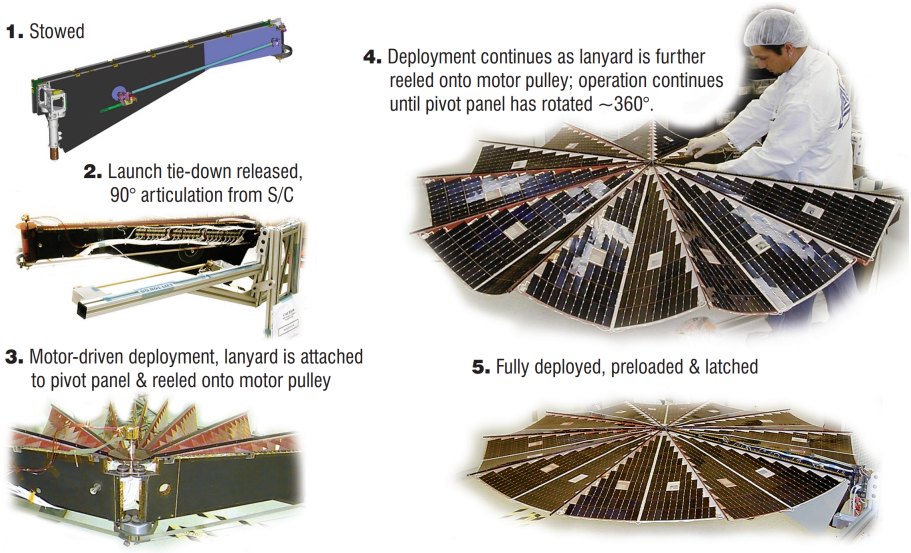


Figure 3.12: Image of the deployment process of the Ultraflex array [40].

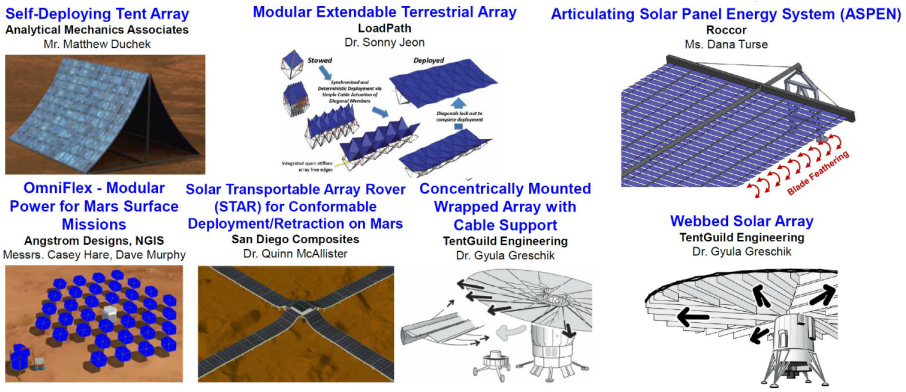


Figure 3.13: 2017 and 2018 SBIR Contract Concepts [41].

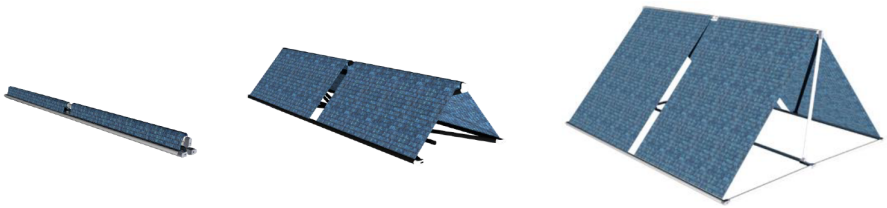


Figure 3.14: Self-Deploying Tent Array deployment scheme [41].

solar cells. The ability for these cells to rotate also in effect provides a built-in dust removal mechanism.

Relocatable Solar Array (RSA) In March 2021, NASA published a paper detailing the conceptual design of a 10kW large scale deployable solar array for use on the surface of the Moon near its lunar South Pole [43]. At this latitude, the Sun is always only a few degrees off the moon horizon meaning the optimal configuration for the solar blanket is to position it in a vertical orientation. That is, have the arrays hang normal to the Moon's surface as shown in Figure 3.16. This maximizes incident light and reduces light deflection. The structure as a whole consists of horizontal cross arm mounted near the top of a comparatively slim vertical mast supported at the base by a tripod configuration. The horizontal

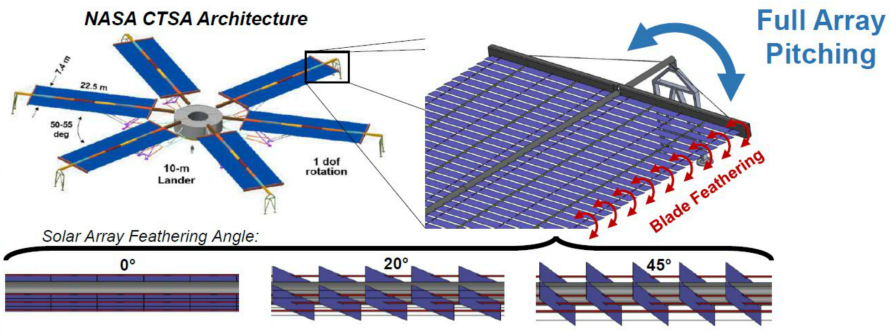


Figure 3.15: The Articulating Solar Panel Energy System [41].

arm is where the deployed flexible arrays hang from. Dimensionally, the fact that the Moon's gravity is a sixth of Earth's allows for significant reduction in part thickness as can be observed with the very slender 0.15 m telescoping mast.

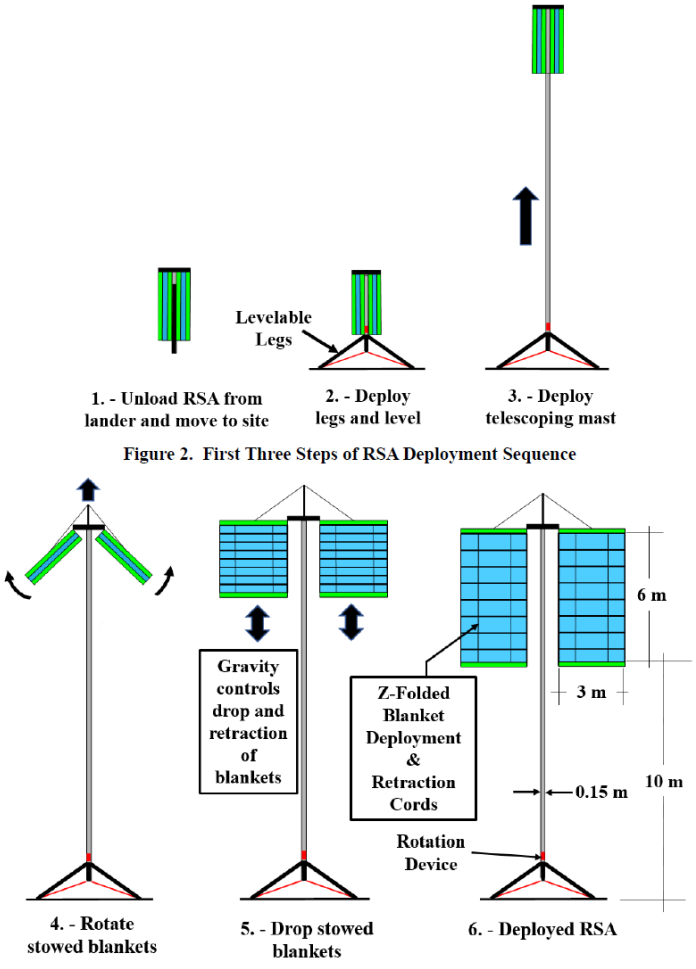


Figure 3.16: Deployment sequence of the RSA [43].

One of the interesting concepts employed in this design is that it uses the Moon's gravity as the method for which the arrays deploy and maintain tension. This greatly reduces the mechanical com-

plexity of the design and allows for a volumetric efficiency increase and mass saving. As the name implies, however, one of the design requirements was that the solar array construction could be repackaged and relocatable. This means that a mechanical system is still required to pull the Z-folded array back into the stowed configuration albeit a much simpler one. An extendable and retractable cord is integrated in the solar array that helps control the gravity assisted deployment as well as permits retraction. It is important to note that the solar array itself is expected to not be allowed to deploy completely, thus the Z-folds are not completely ironed out through tension, in order to ensure that when its time to retract the blanketed array, it folds back properly. No flight demonstration has occurred yet as the premise of this study was to provide a reference concept that concepts in the future could be compared against.

3.2. Mechanisms

3.2.1. Packaging

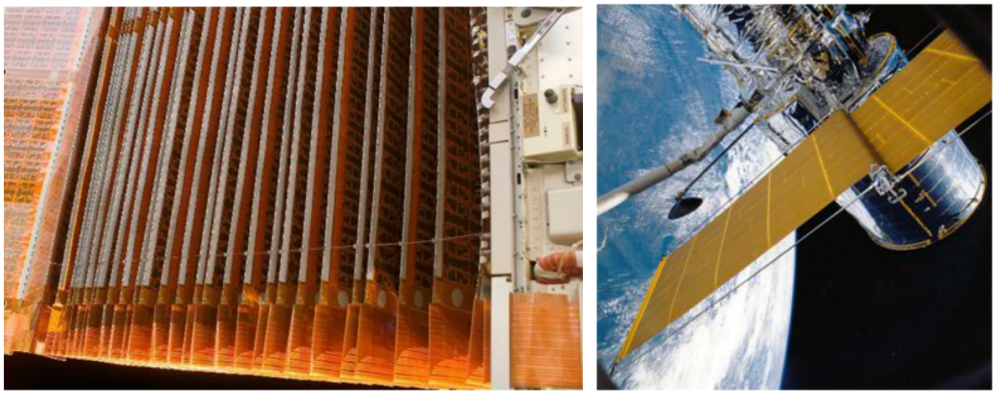


Figure 3.17: Z-folded ISS array (Left) and the rolled Hubble Space Telescope array (Right) [44]

Membranes In terms of large blanketed solar arrays, there are several options available for packaging that are divided into two distinct categories: 1-dimensional and 2-dimensional. As the classifications might suggest, 1-dimensional packaging refers to a decrease in array length along a single axis while 2-dimensional is for two axes. With this in mind, some 1D packaging schemes include simple Z-folding, rolling or wrapping[31, 45], and fan folding [35]. These systems have significant flight heritage and for good reason as they package very efficiently and plastic deformation along crease lines can be avoided by implementing a large enough bend radius. However, restrictions in payload fairing volume may make the scalability of these concepts a problem with regard to large scale deployable arrays. Figure 3.17 shows the ISS Z-folded array and the Hubble Space Telescopes rolled arrays. Ultraflex and Megaflex are previously mentioned examples of fan folding arrays.

There are many 2D packaging schemes as well, namely the double Z-fold, Miura-ori, radial folding [46, 47], and a variety of hybrid systems. Miura-ori is a variation of the double Z-fold that has gotten considerable attention over the past few decades from researchers. The premise of the fold pattern is that instead of having perpendicular fold lines corresponding to the x and y axes, the folds with respect to one dimension are angled as shown in Figure 3.18. The challenge with origami inspired folding patterns like the double Z-fold, Miura-ori, and radial folding is that membrane thickness is unaccounted for [48]. In fold studies, it is general practice to first assume a zero thickness membrane and introducing material thickness can make certain fold patterns impossible. There are strategies to account for the thickness problem, as described by Zirbel et al. [47].

Gdoutos et al. [49] have developed and refined over the past few years a scalable packaging system of up to 60m by 60m that makes use of novel concepts like slip wrapping/slip folding [50] and pressure stabilization [49]. Simply put, the design involves a central hub that houses a solar membrane symmetrically divided amongst four separate symmetrically placed spools. Each solar membrane segment

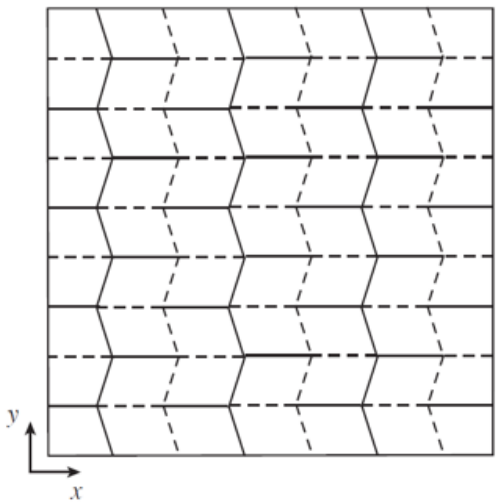


Figure 3.18: Miura-ori folding pattern [47]

is Z-folded and then wrapped. Notably, slip wrapping is a wrapping mechanism that utilizes slip folds, a folding concept where material is removed along the crease line to provide an extensional degree of freedom in the direction of the crease. This newly articulated freedom of movement is important when attempting to wrap a Z-folded membrane around a radius, as shown in Figure 3.19. Without the material removal, the change of radius incurred as the membrane is wrapped around a mandrel would cause the layers to shear causing possible buckling and wrinkling in the inner layers. Lee et al. [51] developed a mathematical methodology for designing a Z-folded membrane that can wrap around cylinders of a chosen size without the integration of slip folds. The process involves the implementation of curved creases rather than straight ones as observed in a regular Z-fold config. The issue with this solution, however, is that the curved creases inhibit the membrane from ever becoming completely flat when stretched which may present solar performance and dust accumulation issues if implemented in solar arrays on Mars.

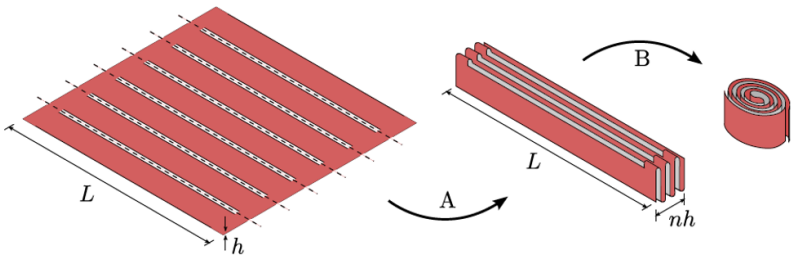


Figure 3.19: Slip wrapped solar membrane that is Z-folded and then wrapped. In this simple case, material is removed along the z-fold crease lines to allow wrapping compatibility [50].

Structural Supports For load bearing structures, a few examples of state-of-the-art storage mechanisms include coiling and telescoping. In regard to coiling, Storable Extendible Tubular Member (STEM) booms and their derivatives have significant flight history [52]. The concept involves the rolling of a circular sheet of steel or another material around a radius thus flattening it and stores elastic energy in the system. If allowed to uncoil, the stored elastic energy drives the deployment, and the material returns to its circular shape. Benefits of this stowage mechanism are that its highly efficient in terms of packaging and easily retractable and typical applications include linear actuators, antennas, and deployment drives. There are many variants of the STEM as can be seen in Figure 3.20. As shown

previously, coilable booms are also very popular in deployable structures as was observed with the MMA and ROSA designs. Specifically, these designs utilize coilable composite booms as seen in Figure 3.21. One problem with these designs, however, is that an additional support structure is required when the boom is deploying because there is a transition region between the flattened boom state and the expanded boom cross-sectional shape seen on the right of Figure 3.21. Consequently, this has negative packing efficiency implications.

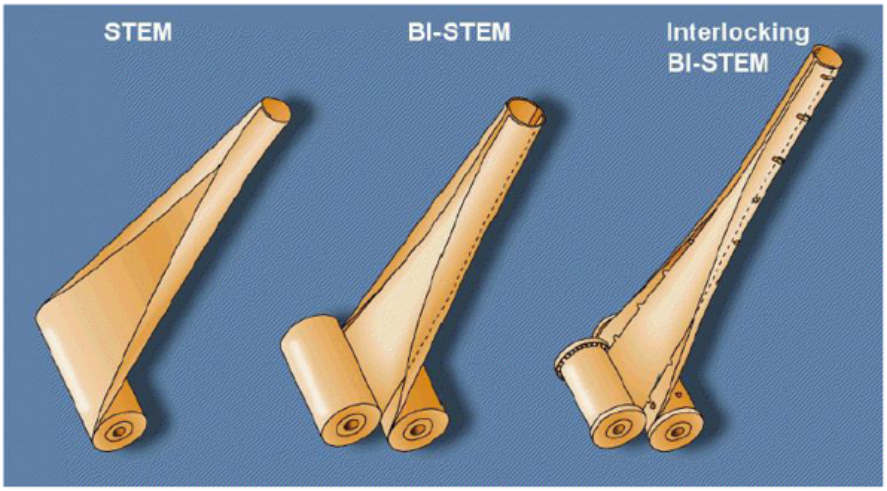


Figure 3.20: Variations of the STEM coilable boom [52]

Figure 3.21: Coiled composite boom (Left) and composite boom designs (Right) [52]

Telescoping is another packaging mechanism that has seen recent development due to its highly efficient packing. As mentioned earlier, the CTSA designed by NASA utilizes the telescoping packaging scheme. Another example which employs the telescoping design was developed by Northrop-Grumman aptly called the “Telescopic Mast” [52]. While highly efficient with regard to packaging, the issue with these designs is typically actuation where deployment and retraction is nontrivial.

3.2.2. Deployment

Pellegrino divides deployment actuation into two separate categories: constraint-driven and energy-driven [48]. Constraint-driven actuation refers methods where a structure’s positional and or dimensional characteristics are constrained to some driving force. For example, varying an angle at a joint through the rotation of an electric motor or lessening the space between two points through the shortening of an active cable. Conversely, energy-driven refers to deployment methods that utilize the minimization of potential energy in a system or the release of stored elastic energy. Constraint-driven and energy-driven methods are typically reversible and non-reversible, respectively.

Uncoiling and unfolding of tape springs is an apt example of energy-driven deployment [53]. Wrapping a tape spring around a radius or folding it stores elastic energy in the structure and typically if left unconstrained, the member will rapidly unfurl or unfold in an uncontrolled and difficult to predict manner. Consequently, most designs involving tape springs typically have control mechanisms that make the deployment highly deterministic. A few of these methods include electric motors [38, 49, 54] and Bi-stability[55]. When considering large deployable solar membrane structures, tape springs and its variants are typically reserved for deployment of structural supports that provide bending and strength performance. This is apparent in the MMA and ROSA designs presented earlier. Moreover, the slip folded design discussed in the previous section utilizes coilable thin shell structures, Figure 3.22, to support the deployed solar membranes. These structures are just two joined tape springs.

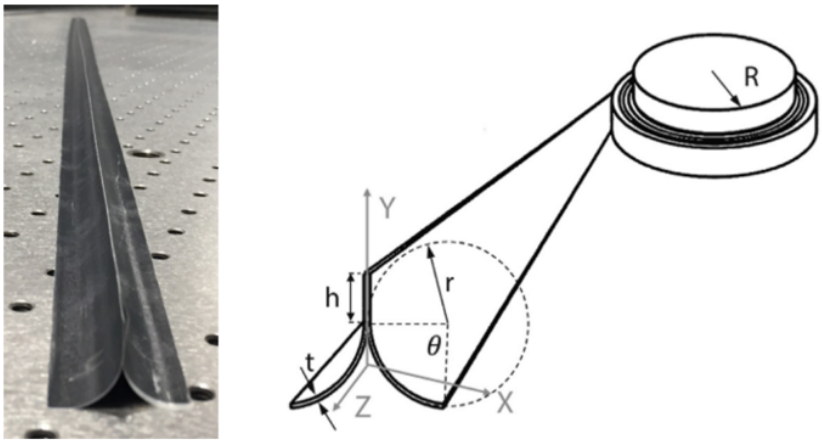


Figure 3.22: Coilable thin shelled structure [56].

Another example of energy-driven deployment is through coilable masts. Structurally, these masts consist of repeating units of three longerons, interconnected through a configuration of pin jointed battens and braced by a network of tensioned cables, as shown in Figure 3.23. In order to be packaged, the longerons bend and curl. If released, stored elastic energy in the longerons in the packaged configuration drives deployment. However, many examples of using coilable masts require controlled deployment meaning the stored elastic energy in the longerons works in conjunction with constraint-driven mechanisms like a motor. Specifically, Hagen Maunch [57] designed a coilable mast where deployment is achieved through a cranking apparatus that rotates a circular baseplate that the packaged coiled mast is attached to. Northrop Grumman [52] developed their own coilable mast with coilable composite carbon fiber longerons.

Lastly, another great example of elastic-driven deployment is the FAST-Mast [52], which is a lightweight deployable mast that is currently flying on the ISS. Specifically, there are eight FAST-Masts support the station's eight large solar arrays. Figure 3.24 shows the packaging configuration as well as its deployed state. Instead of coilable longerons, this design uses buckled battens that store elastic energy which drive deployment. The deployment sequence is controlled by a retaining mechanism in the housing container. A benefit of these mast structure is that they have excellent bending stiffness, are very lightweight, and have components with small cross-sectional areas which could lead to lower wind resistance of the mast when in environments like Mars. The parasitic mass of the housing container after deployment is a reason why in the SAWS, NASA opted out of using these types of structures.

3.2.3. Stabilization/Rigidization

An important design element for any deployable device is stabilization after deployment. Stabilization refers to the process in which a structure "locks" into position after it articulates of its fully deployed configuration, giving it sufficient stiffness to tolerate loading. Some examples include, tape spring snapback, mechanical latching, turning off mechanical actuators, and prestressing. Tape strings are long curved structures that achieve stabilization through snapback [48]. One of the simplest methods for "locking" is through mechanical latching but care needs to be taken so that the locking mechanisms

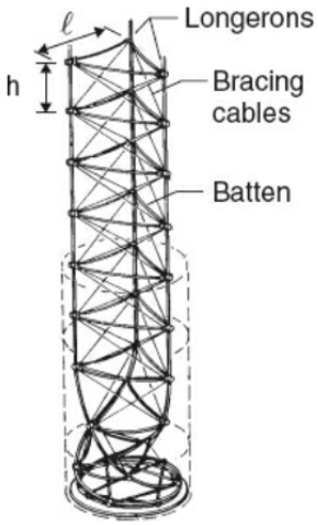


Figure 3.23: A coilable mast configuration [48].

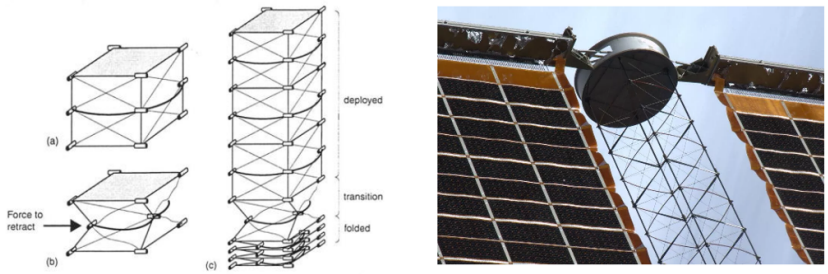


Figure 3.24: The FAST-Mast packaging configuration (Left) and its deployed state (Right) [52].

is volume and weight efficient. If deployment is actuated by a mechanical motor such as in the MMA, turning off the motor is an effective stabilization technique as well.

Pre-stressing and tensioning are common approaches for many deployable structures and are of particular interest for this thesis. In applications where gravity is a factor, it is possible to use this constant force as a locking mechanism for hinges by properly orienting and positioning them as can be observed in Figure 3.25. Through clever routing of a cable, it is possible to also stiffen structural members by applying a tension to this cable, as shown in Figure 3.26. Tensioning is also a common method for stabilizing large thin-walled structures. The CTSA mentioned earlier utilizes guy wires for mass-efficient stiffening of the boxes which house the deployable solar blankets. Furthermore, this method is particularly important for large scale membranes on Mars because there are a number of destabilizing loads like high speed winds and gravity. Gravity can sag the membrane leading to degradation of solar cell performance and wind can cause the membrane to flop around.

Some methods for tensioning solar membranes include negator springs and cable tensioning. In architectural design on Earth, large tensioned membrane roofing structures are very common and their tensioning mechanisms were evaluated [58]. Direct fabric, cable, and mast tensioning are a few that were highlighted. Direct fabric tensioning refers to the used of simple clamped edges that pull on the fabric by adjusting a threaded bolt. The advantage of this system is simplicity but it typically requires close intervals of clamping which may drive up weight of the assembly. Cable tensioning involves applying a pullback force to a mast in which a membrane is attached to through a tieback cable. Lastly, mast tensioning refers to the increasing the length of compression members like a mast or strut through the use of a jack. As the jack moves upwards, the connected fabric membrane is pulled tight. See Figure 3.27.

Figure 3.25: Gravitational pre-stress on joints [48]

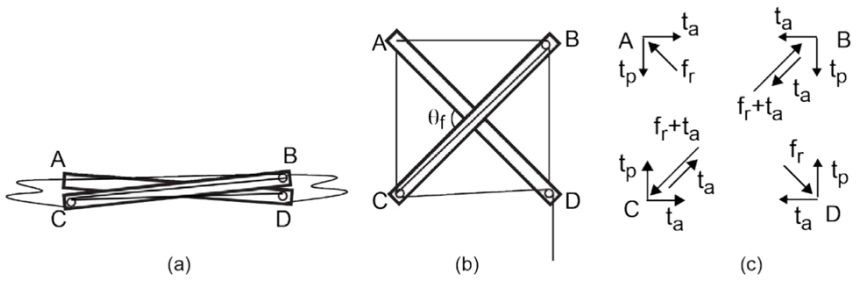


Figure 3.26: Stiffening of a pantograph structure through cable tension [48]

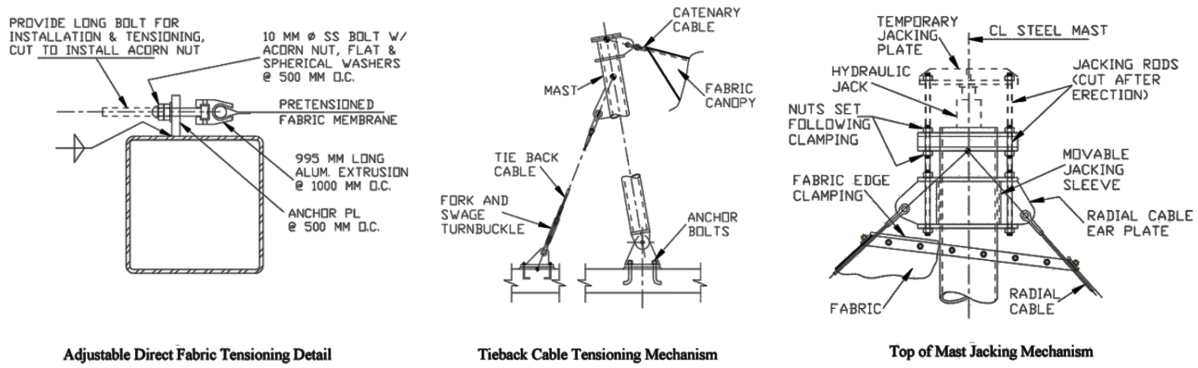


Figure 3.27: Membrane tensioning mechanisms [58]

A prominent example of stabilization mechanisms in thin solar array membrane design can be observed with the Ultraflex solar array shown in Figure 3.28. During service, these arrays articulate stiffness and stability through a combination of static structural elements as well as the use of flat flexure leaf spring elements [59]. As previously described, the Ultraflex solar array is shaped like a folding fan partitioned into several triangular gores. Interstitial spars occupy the boundary between these gores, offering points of connection for the gores as well as supporting them. The spars which are immediately adjacent to the static and pivot sandwich panels (highlighted by the red boxes in Figure 3.28), connect to these sandwich panels with flat flexure leaf spring elements. When the entire array assembly unfurls

during deployment, shown earlier in Figure 3.12, these spring elements engage towards the end of the 360 rotation. This in turn transmits a pre-tension load to each of the triangular gores.

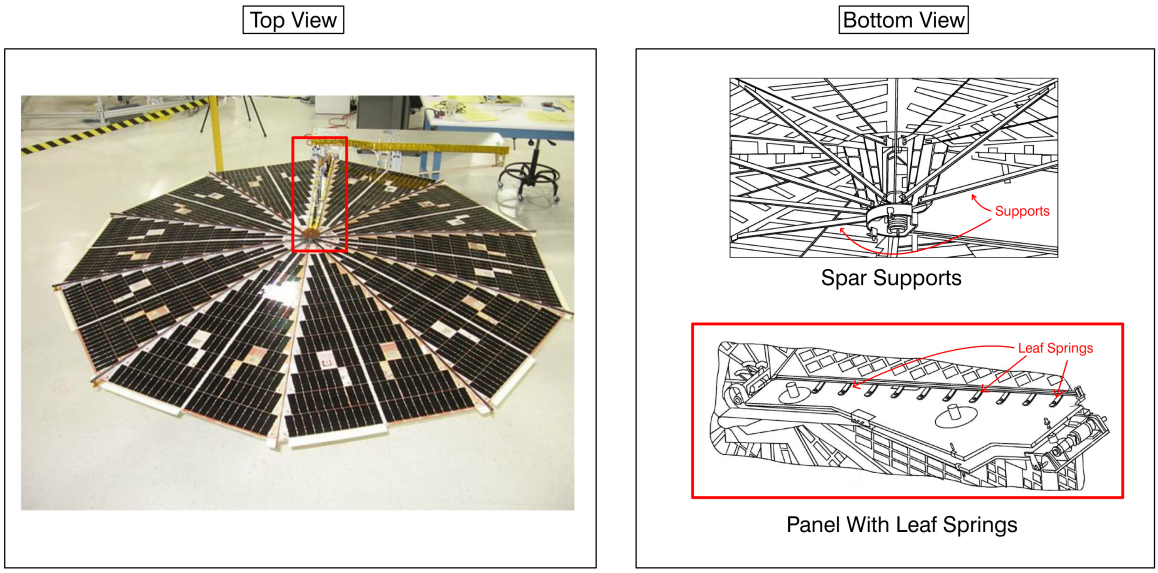


Figure 3.28: Pre-tensioning mechanism for the Ultraflex array and its spar support structure [59]

3.2.4. Dust Mitigation Technologies

Dust Mitigation Technologies (DMTs) can be divided into two separate categories, “active” and “passive”. Passive DMTs refer to methods that aim at reducing the collection rate of dust on a particular surface while active DMTs are methods which utilize some force (mechanical, electrical, magnetic, etc..) to remove dust already deposited on a surface. In effect, active and passive methods can be thought of as removal and abatement processes, respectively.

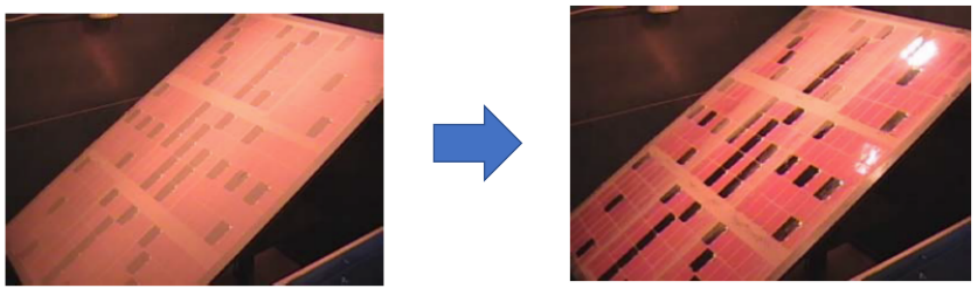


Figure 3.29: Comparison of a solar panel before and after dust removal using piezo-electric actuators [60]

Examples of removal systems include mechanical wipers [61], acoustic waves [62], electrodynamic waves [63], piezoelectric shakers (Figure 3.29) [60], high speed jets [64], Electron beaming [65] and peel-n-discard films [66]. Many of these strategies appear to be effective but their applicability to large scale systems intended for long missions such as in this thesis is still unclear. Moreover, for methods that rely on integration in the solar blanket membrane, as in the case of piezoelectric shakers, it is not understood what impact they might have in regard to packing efficiency. In terms of abatement processes a few include simply tilting the array and work function coatings [66]. For both abatement

and removal systems, a systematic evaluation of the impact of these technologies on large scale array design as well as methods for optimal integration should be performed.

3.3. Materials and Architecture

In this section, attention is concentrated on the flexible, solar cell membrane sub-component characteristic of hybrid deployable solar array structures. Specifically, a brief overview of the typical materials and architectures of the membrane is presented. Following this, a short review of lightweight materials used on Mars is provided.

3.3.1. Space Solar Array Membranes

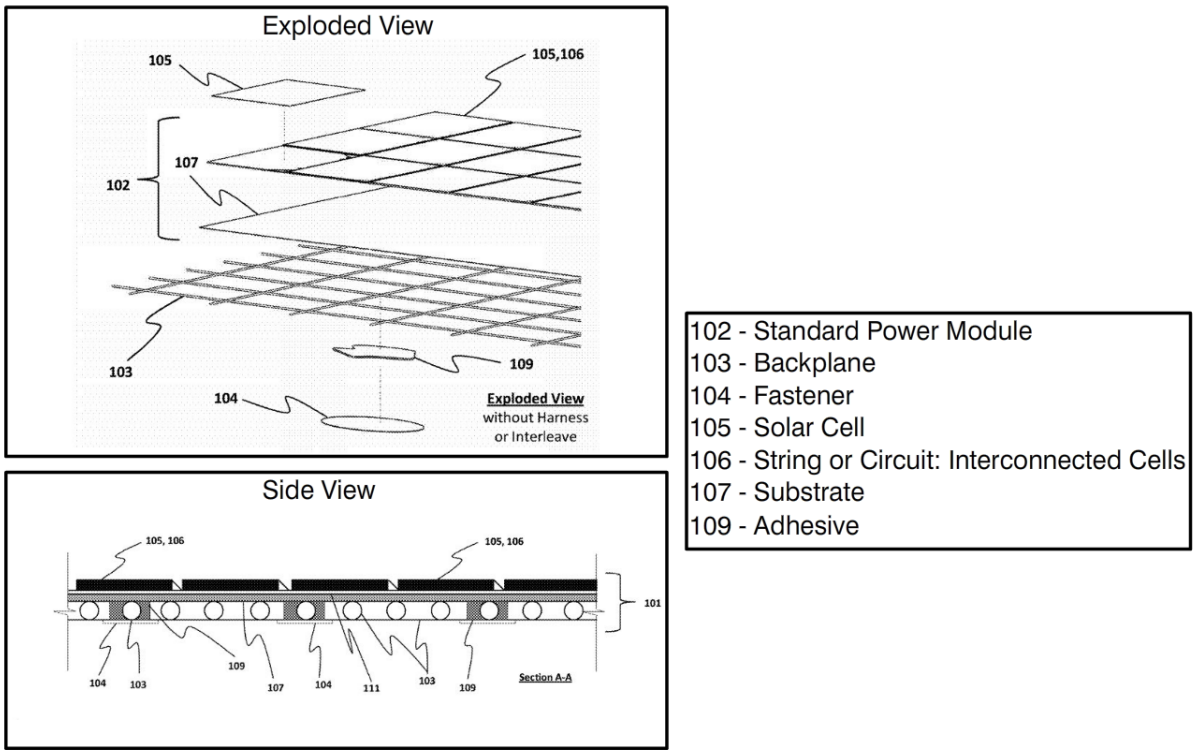


Figure 3.30: Open mesh solar array membrane concept [67]

A non-exhaustive list of heritage blanketed solar arrays in space applications is provided in Table 3.2. Historically, array architectures have usually involved a laminated structure comprised of a thin ceramic or polymer sandwiched between two sheets of Kapton. These structures are typically very thin, on the order of magnitude of less than 1 millimeter in thickness. Presently, flexible solar array blanket design has shifted from this laminated construction to an open mesh design shown in Figure 3.30. Specifically, solar cell strings are adhered to a Kapton or some other kind of polyimide substrate which is then attached to an open weave mesh fiber, cloth, or polymeric material backplane. Adhesion of the solar cells to a substrate together is largely for ease-of-assembly purposes. The backplane is the primary load bearing component of the design, tolerating the tensile loads when the flexible blanket is put in tension to hold shape. Rather than continuously adhering the solar cell-substrate sub-assemblies to the backplane, it is instead performed at discrete points. The dual motivation behind this configuration is that it significantly reduces assembly cost as well as helps isolate the fragile solar cells and their circuitry from experiencing high stresses during deployment. The previously mentioned state-of-the-art Megaflex and ROSA designs employ this setup. An exception to this configuration trend is in the aforementioned Multi-Mission Array whose membrane consists of adjacent flexible, charge dissipative black poly Kapton blanket panels that are connected to one another with carbon fiber hinge pins [38].

In effect, no reinforcing mesh is present and tension is directly applied to the Kapton substrate in which the solar cells are adhered to. As a side note, Kapton is ubiquitous in the space industry for a variety of reasons which include good temperature stability, low solar absorptance, good insulative properties, among others [68]. Atomic oxygen in the Earth's atmosphere heavily degrades the material, however, so Kapton is typically coated in a silicon-based coating for protection if necessitated by the environment.

Table 3.2: Examples of heritage space flexible solar array membrane applications and their respective membrane anatomies [67]

Flexible Blanket Solar Arrays		
Name	Application	Anatomy
Terra (EOS-AM-1)	Earth Observation Orbiter	Kapton-Carbon Fiber Reinforced Laminated Blanket
Solar Array Flight Experiment (SAFE)	Flight Test	Kapton-Glass Fiber Reinforced Laminated Blanket
International Space Station (ISS)	Space Station	Kapton-Glass Fiber Reinforced Laminated Blanket
Milstar	Military Communications Satellite	Kapton-Glass Fiber Reinforced Laminated Blanket
Ultraflex (1st Gen)	Variety of Satellites	Glass Fiber Open Mesh
Hubble Space Telescope (HST)	Telescope	Kapton-Glass Fiber Reinforced Laminated Blanket
Olympus F1	Communications Satellite	Kapton-Glass Fiber Reinforced Laminated Blanket
Roll Out Solar Array (ROSA)	Space Station	Single-Sided Kapton-Glass Fiber Open Mesh

Another key element of flexible solar array design is the inclusion of an interleave material whose function is to protect the fragile solar cells and wiring from damage during launch into space. Simply put, it is a material that is placed between adjacent folds or rolls in the case of a z-folded or rolled configuration, respectively. During stowage, the interleave material is compressed, preloading the stowed structure, and acts as a dampening mechanism to help the photovoltaics tolerate vibratory loads. A polyimide foam is typically used and depending on the applications, special care must be made to ensure it is dimensionally compliant with design. For example, the ROSA has periodically spaced strips of polyimide foam that run parallel with the roll direction. These strips were required to have a certain thickness so that the membrane is compatible with the rolled composite booms that flank each side when rolled. Figure 3.31 shows the ROSA in its stowed state with the exposed, stripped interleave material. It should be noted that encapsulants and various coatings are integrated in solar membrane designs but considering these technologies is beyond the scope of the preliminary design work in this thesis work. Given their low mass and volume implications, these technologies can be suitably ignored for the purposes of this study.

3.3.2. Mars

As shown earlier, two of the most prominent applications of hybrid deployable solar arrays on the surface of Mars were onboard NASA's Phoenix lander in 2008 and more recently NASA's Insight Lander in 2018. Both used the same next-Generation Ultraflex solar arrays which consists of solar cells adhered to an open mesh made of Polyarylate, a thermoplastic with the trade name Vectran. The open mesh allows the design to have low areal mass as well as provides openings for the adhered solar arrays to radiate heat. Vectran has historically been used on Mars in non-solar array applications as well, implemented in deployable airbag systems used for landing the Mars Exploration Rovers (MERs) as well as Pathfinder on Mars.

More generally, other lightweight structural materials that may be of interest to flexible surface solar arrays are currently being tested onboard the Perseverance Rover albeit the focus of this testing is to understand the durability of prospective spacesuit materials to the Marian environment. Limited by the amount of materials that could be brought to Mars, candidate materials for this test were evaluated pre-launch on the basis of loss of tensile strength and elongation when exposed to Mars equivalent UV radiation[69]. Among the materials tested, Spectra (a Polyethylene), Dacron (a Polyester), and Vectran were the three materials that were intended to possible use as a restraint layer in a spacesuit. While Spectra showed the best durability, Dacron and Vectran were ultimately included on the rover for testing.

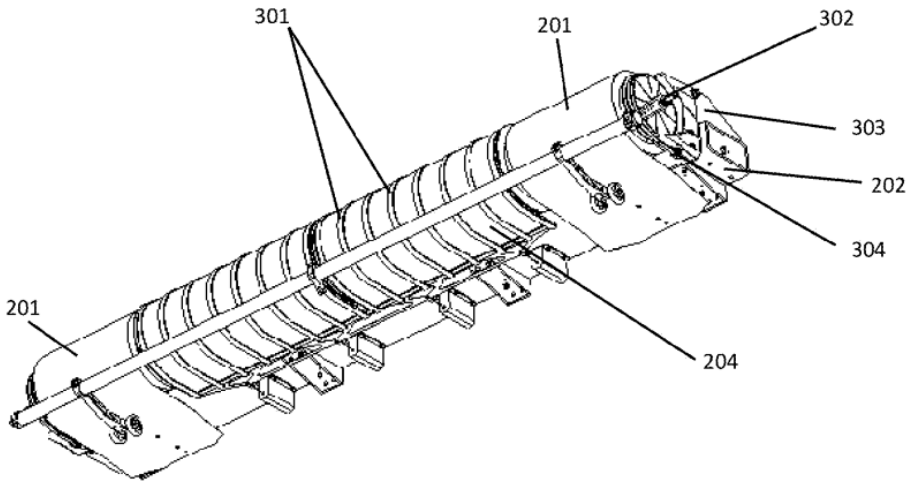


Figure 3.31: The ROSA in its stowed configuration. The interleave strips, pointed out by the number 301, are adhered to the underside of the solar array membrane and run parallel to the roll direction [32].

4

Conceptual Design

In this section, conceptual design of a large-scale, deployable solar array structure for use on the surface of Mars is conducted. The background information presented in the previous section should be more than adequate to understand the inter-workings of the concepts generated herein. First, assumptions and guidelines for the conceptual design phase are presented. A detailed list of the generated concepts are then presented, followed by a detailed down-selection section.

4.1. Assumptions and Guidelines

Prior to undergoing any conceptual design work, establishing a number of working assumptions was paramount. These assumptions not only assist in focusing of the research itself but also dictate the quality and usefulness of the analysis thereof. Such a task is not straightforward, however. Again, it should be noted that NASA does not have any official human Mars program, meaning no decisions have been formulated with regard to specific mission logistics and architecture, time frame, etc. at this time.

4.1.1. Nomenclature

In order to adequately communicate the ideas in the remainder of this report, it is prudent to first provide definitions for the terminology used henceforth. Thus, the nomenclature used to refer to various aspects of the deployable solar array will be as follows. Firstly, the term assembly will be used to refer to the entire solar array structure which includes all the elements which combine together to form the deployable solar array. Logically, these elements will be denoted as components or sub-assemblies, where the solar array assembly consists of two components, a membrane and the supporting architecture. The membrane component refers to all the elements which make up the blanketed array like the solar cells, a substrate material that these cells are adhered to, cell wiring, and whatever structural load-carrying element that integrates with the blanket. Conversely, the supporting architecture refers to every other aspect of the deployable solar array assembly. This includes the structural elements used to tension and suspend the membrane above the ground as well as deployable leg supports (if any) and actuation hardware.

4.1.2. Lander Assumptions

Obtaining a reasonable set of assumptions thus requires a careful review of a collection of different trade studies performed by NASA over the past few decades and an assessment of technological trends. A few design driving factors for deployable, Martian surface solar arrays are associated with the assumed characteristics of the landing vehicle. Namely, the vehicle's diameter, its cargo manifest, and functional capabilities.

Depending on the array design, lander diameter can directly affect its sizing, particularly in the case of radially deployable, planar concepts as shown in Figure 4.1.As shown, adjusting the lander diameter, while maintaining the same offset from the lander perimeter and gap for lander access,results in drastically different dimensions of the deployed array where smaller diameters result in longer and skinnier arrays. Presently, the preferred launch vehicle for a human mission to Mars at NASA appears to be a variant of the SLS, dubbed Block 2B, consisting of a 10 meter fairing diameter, which corresponds to a 9.1 meter diameter lander[70, 71]. Additionally,a study investigating the impact of launch vehicle fairing size on human exploration missions to Mars showed that a 10 meter fairing diameter reduces the performance risk of in-space transportation systems and betters the collective flexibility of the mission design when compared [24]. As a result, assuming a 9.1 meter lander diameter will be used is a reasonable inference.

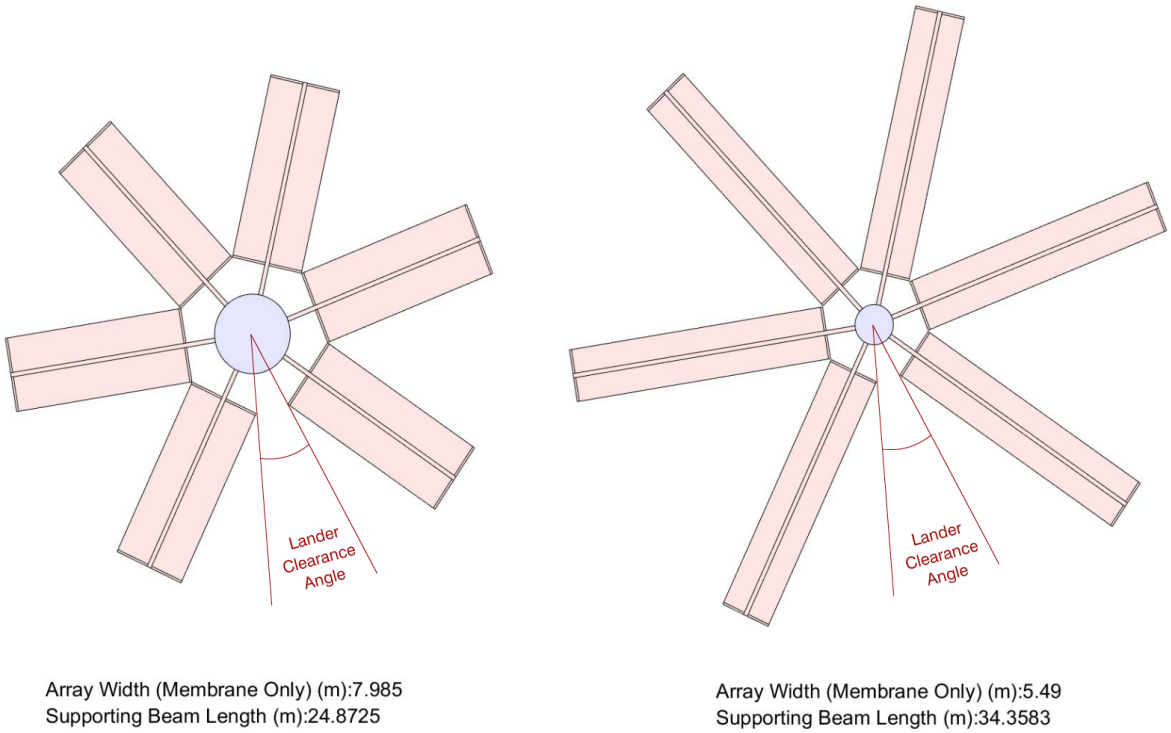


Figure 4.1: Top down view of two different radially, deployable planar arrays. The assumed diameter of the landing vehicle affects the dimensions of planar, radially deployed solar arrays. Given the same total solar array area ($1000m^2$), number of deployable arrays (6),and deployment offset from the lander ($3.5m$), a $9.1m$ class lander (left) allows for shorter and wider arrays in comparison to a $4.6m$ class lander (right) for the same assumed lander clearance angle.

In the baseline SAWS study, one of the principal design conditions was that the devised solar array system be a secondary system onboard the spacecraft. Because of this, the nature and distribution of the rest of the payload is also very influential. That is, it is important to understand what kind of cargo the solar arrays will likely be sharing space with and how this additional cargo may or may not accommodate them. As mentioned previously, there have been several mission architecture trade studies performed by NASA over the past few decades involving sending humans to Mars, each varying in planned duration, surface operations, and overall mission logistics [72, 70, 73]. A common approach among these, however, is to disperse the required cargo to support a surface crewed mission across a hand full of landers over several years prior to human arrival. As described by Polsgrove et al. [70], these cargo elements typically include: a Mars Ascent Vehicle (MAV), some kind of habitation module, surface transportation and mobility systems, power generation infrastructure, In-Situ Resource Utilization (ISRU) equipment (if any), scientific apparatuses, crew consumables, and spares. Preliminary concepts of the MAV, whose primary purpose is to carry astronauts safely from the surface of Mars back into orbit, describe a big crew cabin flanked by several large propellant tanks. This large

assembly imposes dimensional and positional constraints on the deployable solar arrays as well as orientation limitations. Moreover, as pointed out in Pappa et al. [6], rocket plumes from the MAV during orbit reentry could damage arrays attached to the lander thus imposing additional array functional requirements like the ability to retract or be disassembled and relocated. ISRU infrastructure, on the other hand, comprises of equipment meant for producing propellant for MAV. If included in the mission, this equipment may require very large, deployable radiators meant for dissipating heat produced during propellant production [70]. Again, these sizable components can impact packaging and deployment characteristics of the surface solar arrays. That said, a number of recent studies assume that the ISRU equipment and MAV would be coupled together on their own lander and connected after landing to a power grid set up by previous landers [71, 70]. Consequently, for the purposes of this study, this assumption was maintained. In effect, it will be assumed any of the other aforementioned items could be onboard with the solar array assembly.

With regard to functional capabilities of the lander, the chosen method of EDL has particular influence over the deployable solar array design. One of the principle challenges with establishing a crewed mission on Mars is landing all the necessary equipment on the surface. It is estimated that landing payloads 20 times larger than the heaviest robotic payload to date will be required to sustain a human presence on the surface [73]. Larger payloads mean larger landers and even larger EDL systems. Currently, there are a number of competing entry technologies under development within NASA including a Hyper-sonic Inflatable Aerodynamic Decelerator (HIAD), Adaptable Entry and Placement Technology (ADEPT), Mid L/D, and a traditional capsule concept [74]. These technologies are important to surface solar array design because they dictate its packaging characteristics. For example, if the HIAD system is used, this implies the payload would need to be stored on the exposed top deck of the lander with the inflatable system packaged and deployed around the circumference of the vehicle frame, as shown in Figure 4.2. As a result, a deployable solar array concept must be stored on the top deck along with the rest of the cargo without any elements that may overhang and interact with the inflatable structure. Of the aforementioned architectural trade studies, a preference to a HIAD system is noted. Moreover, a recent study involving cost of EDL systems showed that a HIAD system is likely fiscally more viable than the current alternatives [75]. Consequently, this study maintains this assumption.

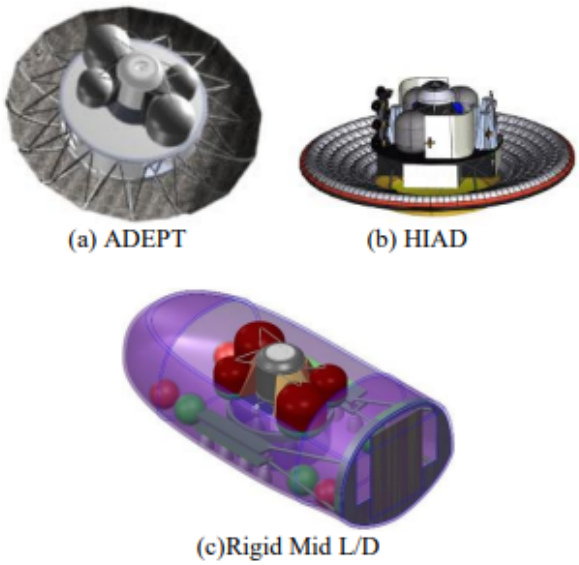


Figure 4.2: Various Mars Entry,Descent,and Landing systems explored by NASA [24]

Lander azimuth control is another design driving element for deployable solar array design on Mars, referring to the lander's ability to rotate around the its central axis once on the surface. This function is important for spacecraft that have deployable solar array elements, particularly in the case of landing at high latitudes. Rotation allows the lander to position its arrays in the optimal position with respect to

the movements of the sun and permits the use of sun tracking systems. Thus, lander azimuth control directly influences the functional requirements of the solar array system itself. In the NASA study used as the baseline for this thesis's investigation, azimuth control of the lander was not assumed, per advice from mission planners at Johnson Spaceflight Center (JSC). This assumption was maintained for this present work.

4.1.3. Solar Array Blanket Assumptions

The assumptions surrounding the solar array blanket sub-assembly can be divided into two categories: photovoltaic cell and blanket. With regard to the cell, its mass, conversion efficiency, and structural composition and configuration all influence various assembly level requirements as well as motivate different aspects of the design decision making process. As an example, cell efficiency directly determines the required total solar array coverage for a given power requirement. This, in turn, influences how one may decide to support or discretize the array. Naturally, assumptions surrounding the cell mass also dictate decisions with respect to the extent and distribution of the supporting architecture as well as the membrane's load carrying element.

Lastly, the assumptions surrounding composition and configuration of the assumed solar cell is also highly influential. Specifically, the cell type is highly deterministic with regard to the overall flexibility of the membrane component. Historically, space-grade triple junction solar cells have significant flight heritage and have been used on the surface of Mars on numerous occasions. By proxy of the cell design, these types of cells contain a relatively thick substrate in which the lattice matched solar cell junctions are grown upon. Because of this, these solar cells are less accommodating to bending and are thus typically mounted to rigid surfaces which possess high flexural stiffness. Conversely, IMM cells are much more flexible given the fact that the manufacturing process allows for the complete removal of the aforementioned thick layer. Clearly, the assumed cell dictates whether or not the deployable solar array assembly consists of a flexible membrane or more of a traditional, rigid panel design. On the topic of cell composition, the assumed cell cover glass thickness also has assembly design implications. Typical cover glass thicknesses in space applications are on the order of magnitude of 0.1 to 0.2 millimeters. While seemingly small, this element of the solar cell accounts for a sizeable portion of the cell's total mass. Again, the total mass of the solar cell directly impacts every aspect of the deployable solar array assembly design. In terms of cell configuration, available size and shapes for which the solar cells can be manufactured also has important design implications. For example, a square or rectangular cell fits less efficiently on a triangular substrate which may influence the decision to explore square membrane structures.

That said, in the reference NASA seedling study, it was assumed that the solar array blankets would have an areal mass of $0.5kg/m^2$. The reasoning behind this choice was based on characteristics of current blanketed solar arrays used in space and positive predictions of how these characteristics may be changed when similar technology is applied in the Martian environment. According to the NASA report, existing blanketed solar arrays like the ISS arrays, MegaFlex and ROSA have an areal mass of approximately $1kg/m^2$. This is a holistic figure that includes the photovoltaic cell, mass of the reinforcement layer, cell cover glass, and cell carrier substrate. The assumed reduction in areal mass was substantiated by the idea that the solar cell cover glass could be reduced by half compared to those used in space. Moreover, it was assumed an IMM cell would be used as opposed to a triple junction cell which provides further mass benefits.

Because of the criticality of this particular design assumption and given it has been some time since the aforementioned study was performed, a small investigation was conducted to determine whether or not this assumption should be revised. Maintaining this idea of using IMM cells for the deployable solar array in this thesis, the approach here was to inspect current applications of these cells, particularly on Mars to estimate the worth of this assumption with the idea that similar performance cells would be used. Per an interview conducted with Joel Schwartz, a supervisor for the Solar Array Subject Matter Experts group (SA SME) at NASA's Jet Propulsion Laboratory (JPL) in California, it was noted that the latest and most advanced use of solar cells used on Mars were onboard the Ingenuity helicopter. These cells were four junction IMM solar cells manufactured by US manufacturer SolAero and are a

derivative of their more general, space-grade IMM cell called IMM-alpha. Reportedly, China's latest rover on Mars also utilizes some variation of IMM cells. Unfortunately, technical information with regard to any of these solar arrays is virtually nonexistent. However, data with regard to the aforementioned IMM-alpha cell shows an areal mass of approximately 49 milligrams per centimeter squared which equates to $0.49kg/m^2$. This value reflects only the cell itself and does not include either the mass of the substrate or of the cover glass. If these features are included in, the mass is approximately $0.71kg/m^2$, assuming a $0.05mm$ cover glass thickness. With this in mind, the $0.5kg/m^2$ figure appears to be overly optimistic, especially since this value is intended to reflect the entire membrane assembly. However, a crewed mission to Mars is likely decades away meaning improvement in technology could be budgeted in. Moreover, the previous comparison is not exact as the IMM-alpha cell is optimized for space applications, not Mars. Thus, for the purposes of this thesis work, the $0.5kg/m^2$ assumption was maintained.

4.1.4. Mission Assumptions

Some of the biggest influencers on the design of a deployable solar array on Mars are the mission architectural parameters like landing location, expected surface operations, and mission duration. Of these, landing location is arguably the most important because it dictates many of the deployable solar array features and functionalities. Moving away from the equator, the effect of solar declination and cosine projection become increasingly problematic for horizontally deployed, static solar arrays. Solar declination refers to the apparent angle between a planet's equatorial plane and its orbital plane formed by its movement around the sun. The degree of this declination changes throughout the year as a consequence of Mars' orbital tilt and is responsible for the seasons much like on Earth. On the other hand, the cosine projection effect refers to an array's reduction in generative area consequential of a relative angle between the source of light and the array. An increase in this angle not only reduces the flux of light incident on the solar array but also increases the surface's reflectance. Collectively, these interrelated factors work together to reduce the efficiency of fixed horizontal solar array systems at high latitudes. Sun-tracking solar array systems offer means to mitigate these problems, providing modest power enhancements at a cost. Specifically, tracking systems are highly dependent on the azimuth control of the lander they are attached to as well as introduce risk, cost, and mass penalties. Per the lander assumptions discussed earlier in this chapter, lander orientation control is not assumed which renders sun-tracking systems less viable. In effect, a focus on higher latitude applications with tracking panels therefore doesn't make sense. Instead, the assumption for this thesis work was that the near-term location of human operations on the surface of Mars would be restricted to near equatorial areas. As mentioned in the introduction, recent work shows a majority of landing zones of interest within plus or minus 30 degree latitudes.

4.2. Concept Generation

The following section provides details with regard to the many conceptual designs articulated during the concept generation phase of this thesis. While a majority of these concepts were ultimately omitted during down selection, it was determined that providing details with regard each design is still important not only from a documentation perspective but also for establishing a design narrative.

4.2.1. Concept 1: Nested Membrane Design

Assembly Overview As shown in Figure 4.3, the anatomy of Concept 1 consists of several folded composite I-beams connected by hinges, where each beam contains nested z-folded solar arrays on either side of the cross-section. Two sets of deployable legs flank the sides of the assembly in the folded configuration and serve as ground supports at the mid and tip locations of the solar array in the deployed state. Similar to the baseline concept, the assembly assumes that several of these systems, referred to as "wings", would be spread equidistant around the edge of the lander's cargo deck.

Packaging With respect to packaging, the design discretizes the total required solar array area, first into multiple separate wings and then within each individual I-beam as shown in Figure 4.4. Height

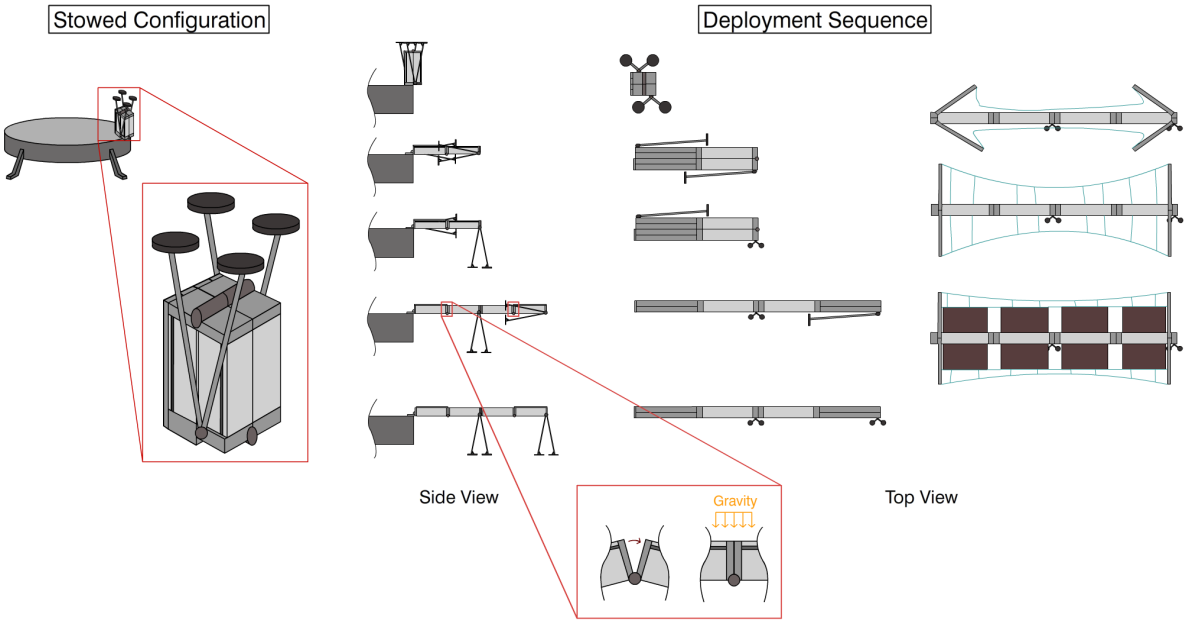


Figure 4.3: The Nested Membrane design includes numerous features including Width-Wise Deployment (WWD) of solar array membrane, gravity stabilized joints, as well as z-folded arrays which nest inside each structural segment.

of the z-folded array is controlled by the depth of the I-beam cross-section. Conversely, the array's packaged width determines the beam's breadth. The purpose of using a z-folded configuration is that it is inherently simple and has widespread industry use, both of which translate to reduced system cost.

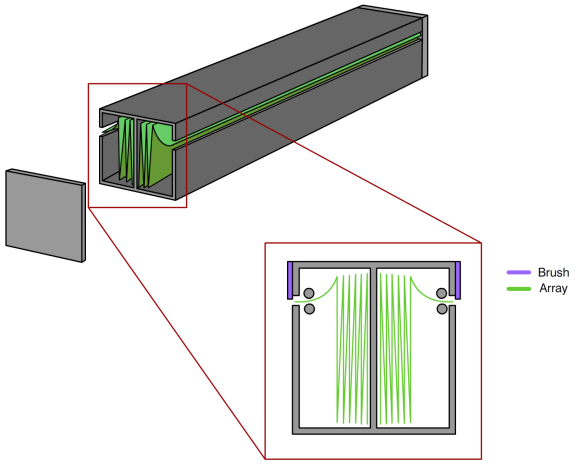


Figure 4.4: Z-folded solar arrays nest within each structural segment as shown, capable of deploying on either side. A static or dynamic brush system could be integrated at the outlet as shown to clean the solar array surface of dust when retracted.

Deployment The deployment process is as follows. Stowed in a vertical orientation as shown in Figure 4.3, the first step involves repositioning the assembly to a horizontal position. Next, the first two I-beam sections unfold and lock into place. Afterwards, the first set of ground support legs are deployed. The remaining two I-beam sections swing into position as shown, allowing the last set of deployable legs to be released into position.The unfolding of the hinged I-beam sections would likely be achieved

through shortening of a low weight, high strength cable routed around each hinge line, a process similar to which is used on the MegaFlex solar array [35]. Controlled deployment of membrane support arms occurs next, unfolding 90 degrees and terminating deployment through a self-locking mechanism. Lastly, the solar array membranes are pulled out from their nested position and tensioned through a polymer based cabling system akin to a suspension bridge. Separate motors drive the deployment for each ”side” of the wing.

Stabilization In the deployed configuration, a number of different strategies are employed to stabilize the structure.The z-folded membranes on each side of the wing are tensioned to provide a stress stiffening effect for flexural resistance against out-of-plane wind loading, gravity, and dust accumulation. The cabling system imparting this stretching force is locked in place at the motor, where the tension can be easily relieved or increased depending on the situation. With regard to the hinged I-beams, the orientations of their hinges are chosen specifically to take advantage of Martian gravity where appropriate. For example, the hinges between supports, identified in red boxes in Figure 4.3, are oriented in a fashion that gravity acts as a stabilizing force in keeping the two sections together. To ensure a solid connection between all the hinged sections, self-locking latches are positioned at each interface and are engaged following deployment completion.

Dust Mitigation A hallmark feature of this concept is the proposed integration of brushes connected on the flanks of each I-beam section. Positioned near the outlet of where the z-folded arrays are pulled out from, the idea is that these brushes would brush any accumulated dust from the surface of the solar array when the array is retracted. It should be noted that any dust that is missed by this mitigation system would not possess any risk to damaging the refolded arrays, provided the accumulation of dust is not excessive or the particulate size is reasonably low. This is due to the fact that solar arrays used in space applications require the inclusion of an interleave material between folds, typically in the form of patches or strips, designed for improving the durability of the stowed array during launch. Thus, in the packaged configuration, the surfaces of each array fold are already offset from one another, leaving adequate space for uncleared areas. The brush system could be static or dynamic, albeit a dynamic brush would severely complicate the design and likely require a significant increase in parasitic mass in the form of additional motors. Intermittent, width-wise slits or gaps are included in the solar membrane design as a consequence of this dust mitigation strategy to allow avenues for brushed dust to be removed.

Advantages and Disadvantages One of the advantages of this design is that it implements what is referred to as ”Width-Wise Deployment” (WWD). The benefit of WWD is that it requires less tension to provide sufficient flexural stiffness to the thin solar array membranes compared to the baseline design which uses ”Length-Wise Deployment” (LWD). Specifically, WWD requires membrane lengths a fraction of the size that LWD does. Another benefit of this design is its dual use functionality. For example, the design is capable of cleaning dust off of the membrane and repackage itself at the same time. Furthermore, the center, load bearing I-beam structure also serves as a housing for the stored arrays. Moreover,the design is less risky from a mission perspective since the assembly consists of several, self contained wings. If one wing fails to deploy or is damaged, power production capability is not completely lost. The integration of slits in the membrane also offer a built in wind load relief mechanism. Conversely, utilization of I-beam cross-sections means torsional stiffness of the structure may be an issue. Furthermore, the design as described in the previous section will require a significant amount of parasitic mass in the form of electric motors in order for the design to function as intended. Additionally, the mechanisms for retracting the membrane and unfolding of the I-beams pose design risks on the basis of system complexity. Lastly, in order of the z-folded array to be removed from the I-beam section cavity without damage, a combination of a system of rollers and internal offset distance, as shown in Figure 4.3 would likely be necessary. This has adverse packaging efficiency implications.

4.2.2. Concept 2: Telescopic Slider Design

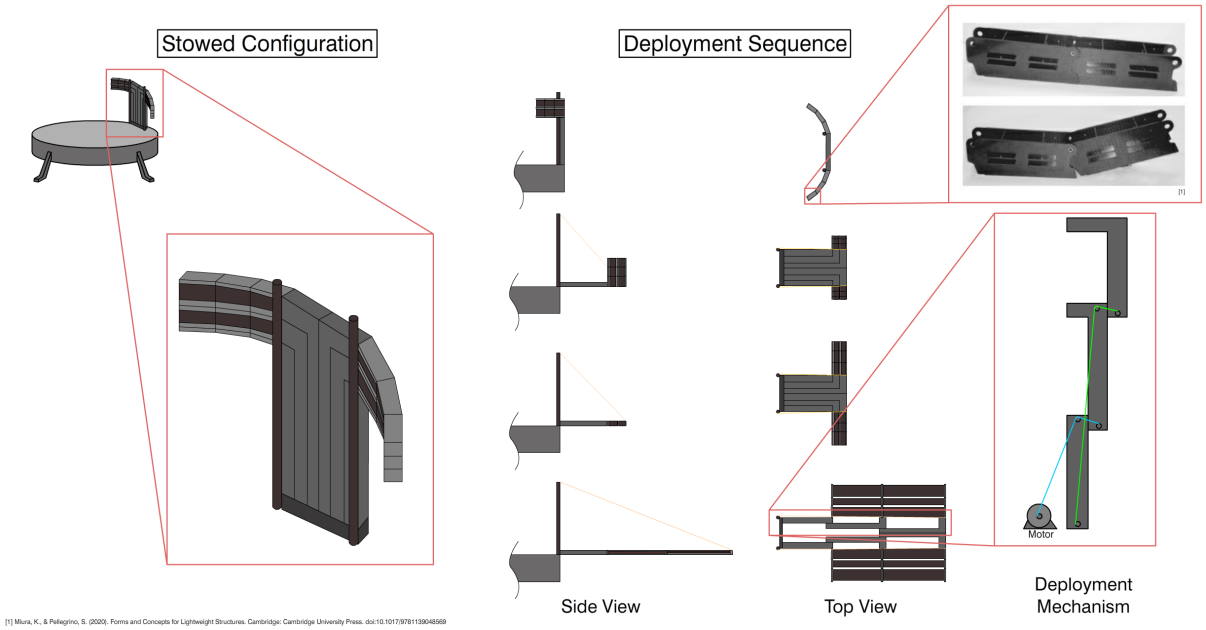


Figure 4.5: The Telescoping Slider Design utilizes a very simple cable actuated deployment system to articulate its structure. Furthermore, the support arms are capable of matching the profile of the lander deck perimeter when stowed and unroll to a flat when deployed as shown. Two cable-staying columns stabilize the deployment process. Lastly, each joint segment in the aforementioned support arms has z-folded arrays. These arrays connect between adjacent support arm segments resulting in the stripped configuration as shown.

Assembly Overview For this design, a series of telescopic sliders work in conjunction with gravity stabilizing, bendable lateral support arms which contain nested z-folded arrays. Two cables connected at the tip of the outer most slider to two static vertical supports on the lander deck helps stabilize the deployment process. Similar to other concepts in this thesis, this design assumes the total solar array area is discretized into numerous wings.

Packaging As mentioned previously, a characteristic of this design is that it utilizes bendable lateral support arms called ”push-chain booms”. This type of boom comprises of several interlocking chain links, as shown Figure 4.5, that permit asymmetric rotational deployment behaviour around one axis. Counterclockwise rotation is unabated, allowing for stowage round some radius. Conversely, clockwise rotation is only viable up until the straight configuration is achieved. The total solar array membrane area is divided among the links, stowed in a z-folded configuration on the flanks of each chain. Each wing assembly has multiple push-chain booms attached on either side of telescopic sliders. The asymmetry in bending behavior allows the assembly to fit to the curvature of the lander deck when in the stowed position.

Deployment To deploy, the wing assembly is first lowered from its vertical stowed orientation till it is parallel with the Martian surface. This process is actuated by an electric motor at the base of the assembly and stabilized by cable stays attached to the assembly tip as shown. The orientation of the push-chain booms is specifically designed so that once in the horizontal position, gravity drives the linkages to unroll into the aforementioned straight configuration. Once in this position, a set of motors shorten an active cable that, through clever cable routing and additional passive cables, allows the system to deploy outboard from the lander. As is done in the baseline configuration, this deployment process serves a secondary function as it also allows the unfolding of the packaged arrays as well. If

a deployable leg tip support is included in the assembly, that last step involves its deployment.

Stabilization As mentioned previously, there are a number of different stabilizing methodologies utilized in this design. Firstly, cable-stays are implemented not only to assist in controlling the rate of rotation of the assembly as it is moved from its vertical stowed position but also to prevent the assembly displacing past the horizontal configuration. The stays maintain this stabilization even as the assembly telescopes outboard. Additionally, gravity is used as a stabilizing force to ”unroll” the push-chain boom and keep the linkages locked. Lastly, at the terminus of each telescoping slider, mechanical locks engaged to lock the structure in position. The cable tension driven by the motor for deployment is thus no longer necessary.

Dust Mitigation With regard to the mitigation of dust from the surface of the solar arrays, the design only really accommodates the introduction of mechanisms like piezoelectric shakers or an EDS integrated in the membrane of each of the array ”strips” . Additional cabling could maybe be routed along the telescopic sliders that could actuate or ”feather’ the solar array strips to a vertical position as shown in Figure 4.6. This operation could be performed in evening hours to prevent dust accumulation.

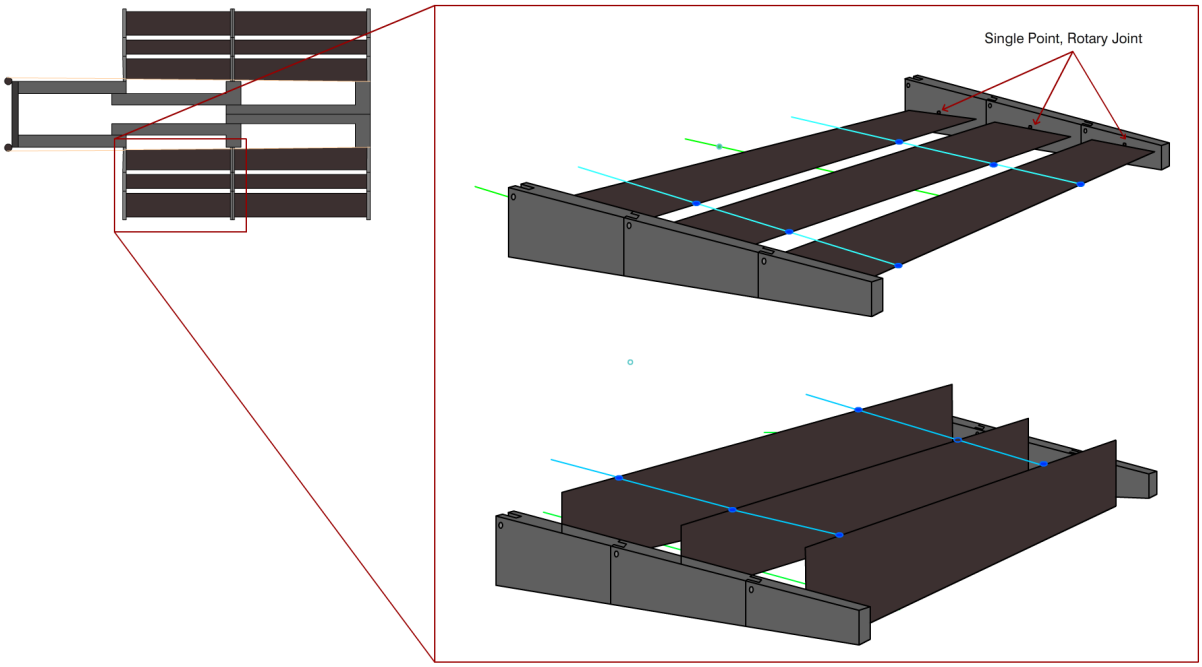


Figure 4.6: Through clever routing of cabling between adjacent solar array strips, single point rotary joints could be included at the solar array connection points to each support arm to allow single degree of freedom rotation as shown. This could be used to relieve incident wind loads as well as prevent dust collection on the surface during evening hours.

Advantages and Disadvantages Advantages in this design are many. Utilizing cable stays likely reduces the necessary structural mass of the telescopic sliders as the tip support improves its bending performance of during deployment (cantilever vs. simple supported). If array feathering is not included in the design, the assembly itself is rather simple and involves a limited amount of motors to completely deploy. Only one motor is required to deploy either side of the the telescopic sliders. Gravity assistance for deployment of the push-chain booms reduces parasitic mass as well. The stowed configuration can be easily made to meet the form factor of the lander thus utilizing the payload space more efficiently. A beneficial consequence of the slider design is that it mandates that the array be split into several sections, which is a benefit not only from a tensioning/support perspective but also in terms of manufacturing and qualification. On the other hand, there are a couple of disadvantages to the design. To

meet the array area needs for a crewed mission, the array area per wing is high which translates into long and slender telescoping sliders. While the stowed configuration can meet the form of the payload deck curvature, it will still have an exceedingly tall form factor which poses possible stowage problems. Furthermore, the push-chain booms will likely be a complex geometry, being difficult to manufacture especially if the component material is a composite.

4.2.3. Concept 3: Cellular Blind Design

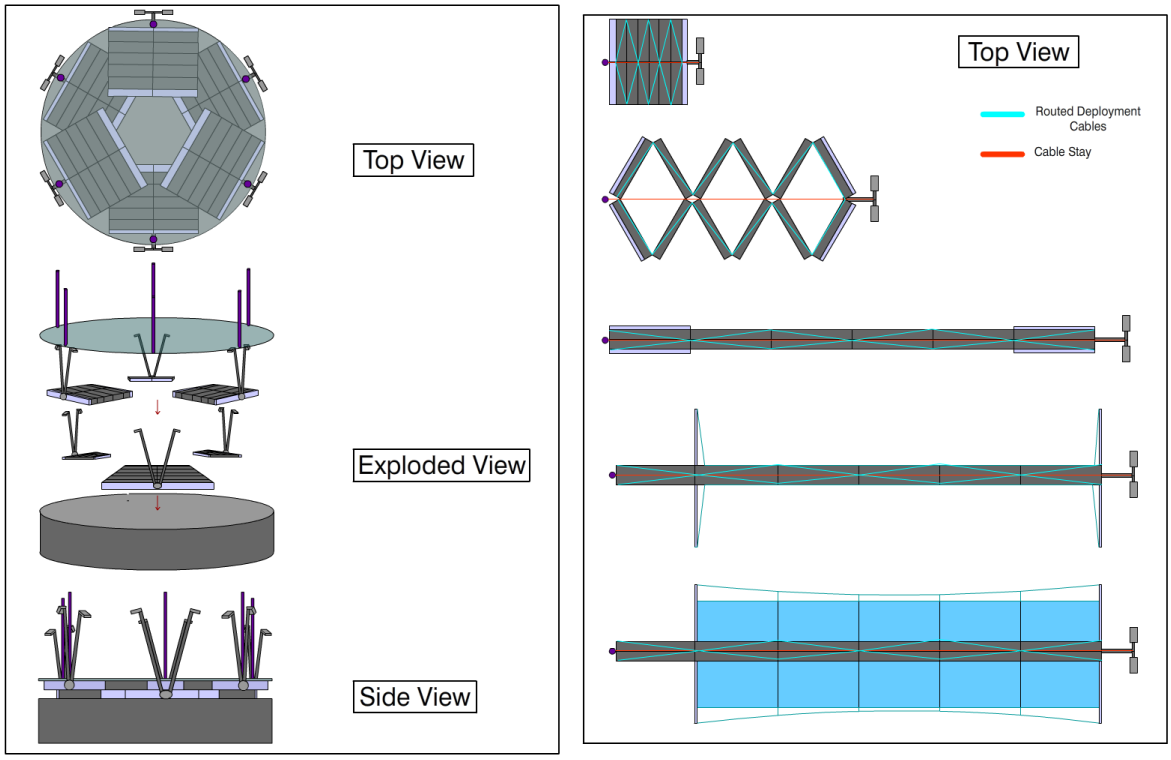


Figure 4.7: A few key features of the Cellular Blind Design are the fact that the array is divided into identical, deployable wings which can stack and stow underneath the lander deck as shown. Cable-staying columns are located at the outlet of each deployable wing and connect to the wing tip to stabilize the deployment process.Cables route around the joints between the array containers. Shortening these cables results in extension of the wing as shown in the ”Top View”.

Assembly Overview The principle elements of the Cellular Blind design are shown in Figure 4.7 and Figure 4.8. As the name suggests, one of the main components of this design is the introduction of a collapsible, cellular solar array membrane. The total array area is segmented into multiple wings and further divided into individual compartments or containers as shown in Figure 4.7. The housing of each array, serves not only as a canister but also a load bearing structure during and after deployment.

Packaging Beginning in the stowed configuration, compaction is achieved through the folding of the array canisters with hinges at their mutual interfaces (Figure 4.7). The arrays package into each canister similar to that of commercially available, cellular window blinds. In the deployed configuration, these array segments deploy width-wise as shown. Another key element of the packaging method for this concept is the fact that the stowed arrays are positioned on separate levels underneath the main lander payload deck and are spaced evenly from one another on each level. The only elements which are on the top deck are the deployable support legs folded upwards and fixed cable-staying pole for each wing.

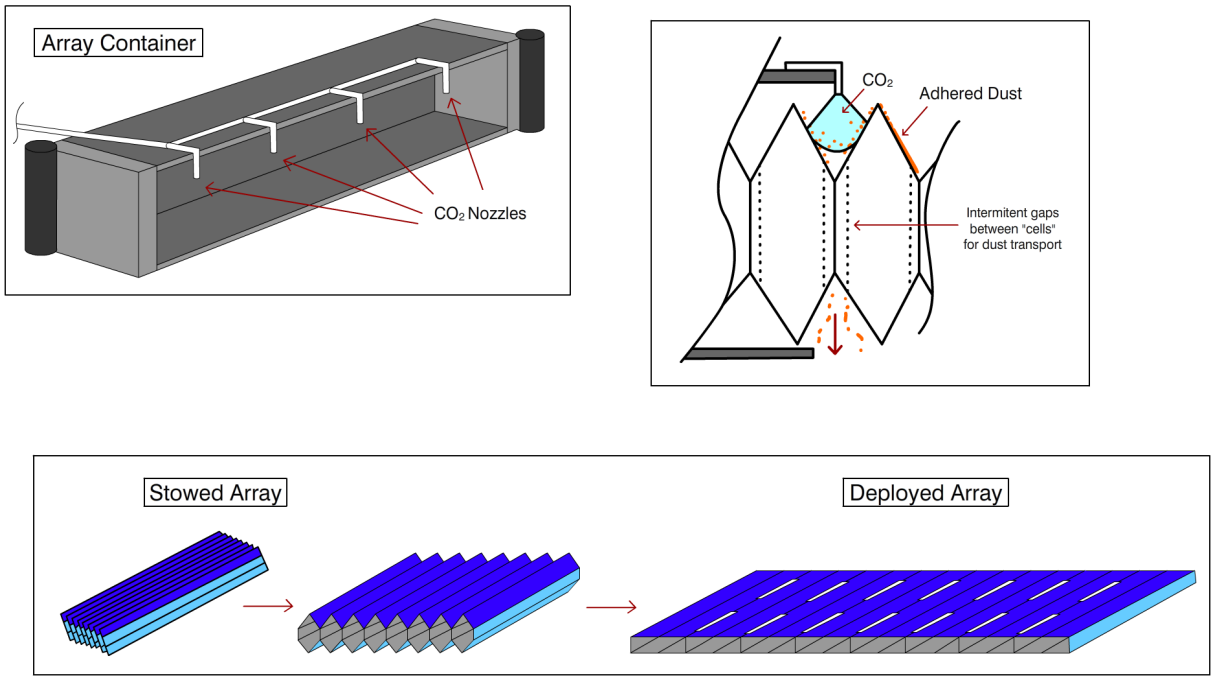


Figure 4.8: The membrane in this application is cellular in design which provides structural depth to improve array flexural stiffness. Each array container has small CO_2 nozzles which activate when retracting the array. Compressed CO_2 blows dust off the solar array membrane surface as shown prior to the membrane folding completely upright as shown. Interstitial gaps between membrane cells offer avenues for the blown dust to be removed.

Deployment The deployment process begins for each of the stowed wings to begin folding outwards as shown in Figure 4.7.Two routed cables through the structure, as shown, are shortened through the use of two separate motors. Routing around each of the joints where array containers meet, this shortening extends the array much in the same way as a pantographic structure. After the structure meets its fully extended state, nested spreader bars unfold at the root and tip of the assembly, offering a structure to deploy the array through suspension cables.

Stabilization To assist in the deployment process, a cable-stay is attached to outer most array canister joint as shown in Figure 4.8 on the right. Once completely deployed, self locking mechanisms in the hinge joints between each canister lock the structure in place. The cellular nature of the individual array segments provides the structure with some degree of structural depth which may assist in improving the flexural rigidity of the structure. This added depth also reduces the amount tension necessary to stiffen the membrane out of plane as well. Similar to the other concepts thus far, tension is applied to each solar membrane segment to bolster their out-of-plane stiffness.

Dust Mitigation The primary dust mitigation strategy of this design is coupled with the array retracting process. The thought process is similar to that of Concept 1 but the design execution is different. Namely, as each fold of the array collapses into its respective housing, strategically placed nozzles blow pressurized gas into the v shaped cavity formed by the folded array, pushing the dust off the array and through small intermittent slits at the base of each cavity (Figure 4.8). This process is performed on each of the array folds as it packages. With regard to the pressurized gas, it is harvested from the surrounding atmosphere through a compressor onboard the lander. The thought process behind integrating this technology is that dust removal by wind has been shown in the past to be an effective cleaning tool, when available. To avoid excessive energy expenditure associated with retracting and deploying the arrays as well as harvesting and releasing the gas, this cleaning system will likely need to be periodic in nature, occurring once or twice every few weeks.

Advantages and Disadvantages The use of a cable-stay in the design is an advantage because it has a load relieving effect on the load bearing array canisters. Instead of being loaded in bending, these segments get loaded in compression. This allows for more efficient use of the structure. A principle advantage of this design is that the design is segmented into separate wings and stow underneath the main payload deck. This permits an unprecedented amount of space for additional cargo onboard the lander. Furthermore, the deployment process is extremely simple and each wing contains several, smaller identical components which has manufacturing benefits. The cellular design of the membrane not only allows for feasible use of pressured gas technology to remove dust, but also offers up another dust abatement strategy. Specifically, each membrane segment can be partially retracted at night to form a zig-zag shape that may prevent dust from setting on the surface. A concern for this design is the additional parasitic mass necessary to articulate the pressurized gas cleaning system. Long lengths of flexible tubing will be required to transport the gas throughout each array structure as well as machinery to collect and compress the gas. Furthermore, the collection system will have to filter out dust suspended in the air in order to properly function, thus adding further complication to the design. Moreover, the stowed form factor will likely be an issue, particularly if the the system is scaled.

4.2.4. Concept 4: Crane Deployment Design

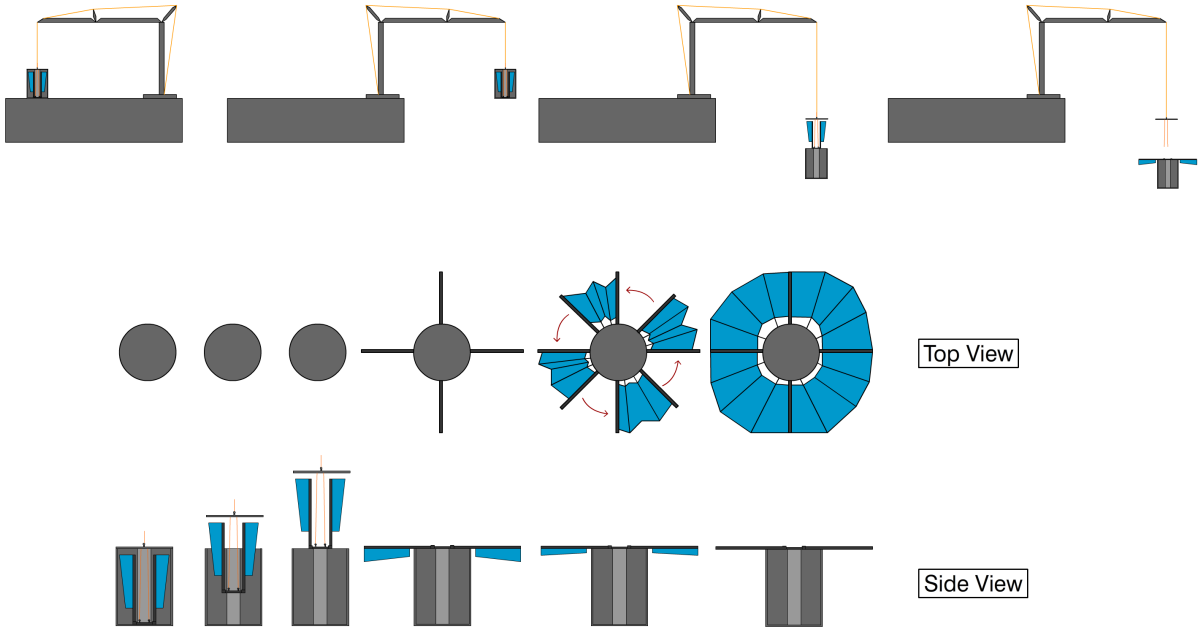


Figure 4.9: The principal feature of this design is the deployable crane system which picks up and deploys self contained, circular arrays as shown. Prior to placing each self-contained array, an internal mechanism is activated to allow the release of the array cover. This cover is interconnected with a ring located at the base of the array container that connects to solar array segments. Thus removal of the array cap, pulls the array out its stowed configuration as shown.A nested motor in the central column rotates the four array segments and lock in with one another to articulate the deployed state.

Assembly Overview Every concept generated thus far has required the solar array assemblies to be stowed equidistant around the exterior of the payload deck. This concepts assumes a deployable crane mechanism for unloading cargo will be present onboard the lander that can be used to pick up and deploy array modules away from the lander. Consequently, greater flexibility is given to placement of the modules on the payload deck. In terms of the array modules, each self contained unit involves a cylindrical casing with the array folded and stowed within as shown Figure 4.9.

Packaging As previously mentioned, the assembly consists of several, self-contained array modules that can be stowed and packaged anywhere on the payload deck that is accessible by crane. The

crane itself is able to be compactly packaged as well and is based off of the lunar crane originally developed during the NASA Constellation program [76, 77]. In terms of the array modules themselves, the exterior is equipped with deployable leg stands that stow relatively flush with the casing and unfold during deployment (Figure 4.10) .The solar array component is segmented into multiple sections and folds as shown with the supporting architecture. The entire array-support sub-assembly collapses into cylindrical casing as shown in the bottom portion of Figure 4.9.

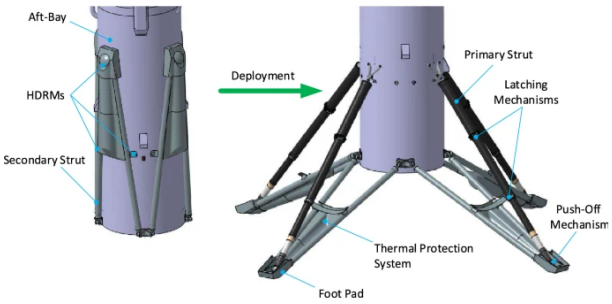


Figure 4.10: The exterior of each array self-contained unit would have deployable legs similar to the one shown here that stow fairly flush with the container shell. [78]

Deployment Deployment consists of a number of steps. Firstly, the heavy-lift crane takes deploys from its stowed state and takes position above the first array module. Robotically, the crane attaches to the top of the canister and lifts it from the lander deck. After the crane has repositioned the array module to the desired location, a release is activated within the array canister that allows for the nested support structure to unfurl as shown. In effect, gravity drives this operation. After the unfurling of the stowed array and supporting structure, the support legs located on the exterior of the array module deploy and the crane gently lowers the assembly to the martian surface. From here, an electric motor at the base of the canister rotates a nested shaft in the center which deploys the arrays.

Stabilization Stabilization in this design comes from two key design features: self-locking mechanisms and coil springs. After the array and supporting structure first deploy from the caster through the clever use of gravity, the structure locks in place with self-locking mechanisms. During the array deployment phase, that is when the electric motor drives the rotation of the central shaft that unfurls the array segments, the arrays are locked into place at the adjacent support arm, as shown. The lock itself is connected to the support arm with coil springs which apply a preload to the solar array membrane. This stress-stiffens the array.

Dust Mitigation Given the architecture of the concept, various active dust mitigation systems could be integrated. Namely, each evening, it may be possible to collapse the arrays in order to prevent dust collection when the arrays are not in use. Moreover, piezoelectric devices or and EDS system could be integrated into the membrane. Lastly, it may be possible to integrate a deployable fan at the center of the deployed array. A small motor could drive the propeller rotation, pushing air down onto the array and shedding dust. Such a system may require more power but this issue could be mitigated through periodic use.

Advantages and Disadvantages Advantages of this design are as follows. The dividing up of the array into self-contained modules improves manufacturability and reduces mission risk. Use of a crane allows greater flexibility for where these modules can be placed on the payload deck as well. Gravity assisted deployment of the nested array and support beams reduces system mass by requiring fewer motors during deployment. A disadvantage to this design is that it is contingent on the inclusion of a crane in the lander payload. Moreover, external connection of the all the modules electrically is

assumed to be done robotically after the arrays are properly deployed. That is, each module would be plugged into the lander via a durable corded connection where the energy storage is located (batteries). If this is not possible, a corded connection must be maintained between each module and the lander during the entire deployment sequence. This adds unwanted complexity and risk to deployment. A prominent downside to this design is also its main feature. Independent robotic assistance from a crane may not be not be feasible and also introduce too much unneeded deployment risk and complexity.

4.2.5. Concept 5: Scissor-Structure Design

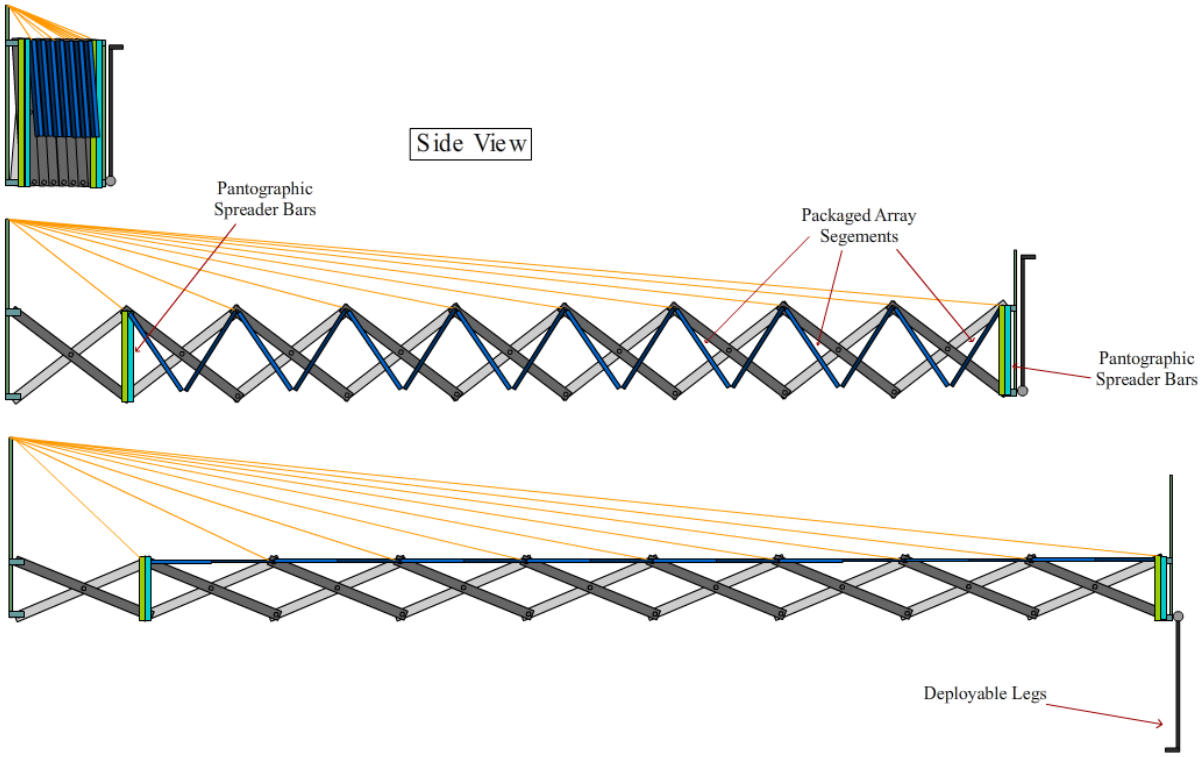


Figure 4.11: The Scissor-Structure Design is a pantographic solar array structure that has several smaller, repeating structural elements that support array containers/segments that fold up as shown. These lock in place and terminate the deployment process. A cable-stay column located at the root of the array connects to each of the scissor joints as shown to support the structure.

Assembly Overview The principle feature of this design is its pantographic structure. This structure is comprised of several repeating units called Scissor Units (SUs), each consisting of a load bearing cross that allows one degree-of-freedom rotation around its center and two collapsible solar array modules attached at the cross tips on either side. Each unit connects to one another at the tips of the cross.This concept assumes the total membrane area is split into several wings/assemblies as is the assumption of the proceeding concepts. Lastly, this concept is a WWD design.

Packaging Stowed vertically and at the edge of the payload deck, the SU structure collapses as shown in Figure 4.11. The packaging efficiency and form factor of this design is determined by the cross-section of the SU structural members, the number of units, and sizing of the array modules. Each array module or segment is similar in design to household window blinds, where the solar array is divided into separate blades, are connected to one another through a network of lightweight cabling, and they all package into a rectangular box shape. Note, each blade contains a string of solar cells arranged in a line, stacked in the long direction of the blade as shown in Figure 4.12. It should be noted

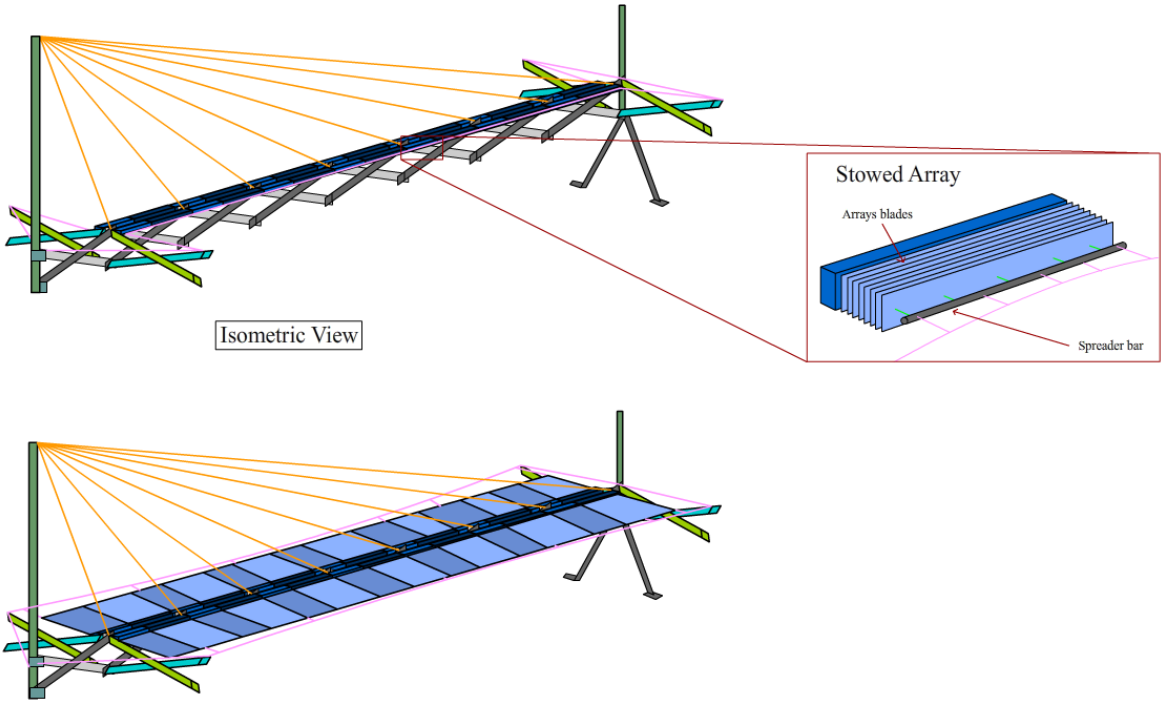


Figure 4.12: WWD of the array is utilized in this design where each array segment consists of several solar array "Blades" that are stretched out by catenary cabling. This cabling pivots around the tip of each of the pantographic spreader bars as shown.

that the spreader bars are also pantographs but differ in dimension from that of the primary, central pantographic structure that the arrays are attached to. This is done so that the spreader bars can optimized for the required deployment lengths of the z-folded arrays.

Deployment The benefit of this design is that the deployment system is synergistic with structural configuration. Starting from the stowed, vertical position, electric motors on either side of the construction apply tension to cables which weave up and down through each SU. This applied tension expands each unit,extending the assembly outboard from the lander but also lowering it. Consequently, the deployment process requires that one edge of the SU at the root of assembly be allowed to slide up and down as shown. Extension of the central pantographic structure also drives the extension of the spreader bars.Cable-stays attached to the tip of the column which permits the aforementioned sliding compatibility, connect to each of the joint interfaces between each SU as shown. This assists in reducing bending loads on the joints during deployment. Bars also comprised of SUs located at the tip and root of the assembly, spread perpendicular to the primary pantographic boom. Deployment is terminated when the solar array modules lock into place with one another as shown. Similar to Concept 1, a suspension cable system is then tightened by two electric motors at the root of the assembly which pulls stowed arrays outwards and tension them.

Stabilization As previously mentioned, cable-stays are used to stabilize the deployment of the central pantographic boom. Self-locking mechanisms between each of the array modules lock the entirety of the structure in place. Lastly, each module is tensioned by a suspension cable system perpendicular to the long direction of the solar array blades. A spreader bar at the tip of the array serves as a static method for maintaining the spread of each blade as shown.

Dust Mitigation The bladed nature of the solar arrays in each array module allows for array feathering. That is, the solar array blades can rotate similar in fashion the household blinds. This rotation is a simple method for reducing dust collection during the evening hours. Lastly, like the previous concept,

piezoelectric devices or an EDS system could be integrated into the membrane of each blade.

Advantages and Disadvantages The repeating nature and geometric simplicity of the load bearing elements within the concept undoubtedly improves its manufacturability. Furthermore, the discretization of solar array area into separate modules and individual blades has a number of benefits. Operational risk is reduced with this concept because if one module fails, there are still several more that may work. Moreover, if a single blade fails, the rest of the module can still function normally. The simple bladed shape allows for standard solar cell geometries to be used, thus driving down manufacturing and assembly costs. Segmentation of the array into modules also provides the possibility for replacement of parts. If a module fails, a replacement module could be installed without having to replace the entire wing. The feathering ability of blades could also be an effective, energy efficient method for dust mitigation and wind load alleviation.There are a number of disadvantages associated with this design, however.First and foremost, depending on how many SUs are utilized in the structure, the required number of mechanical joints can be quite high which adds weight. Also, this can add nonlinear behavior to the deployment process that can adversely affect its determinancy. Furthermore, SU count significantly affects stowed dimensions where excessively tall or wide assemblies can occur which are incompatible with the available lander payload space.

4.2.6. Concept 6: Revised Compact Telescoping Surface Array (RCTSA)

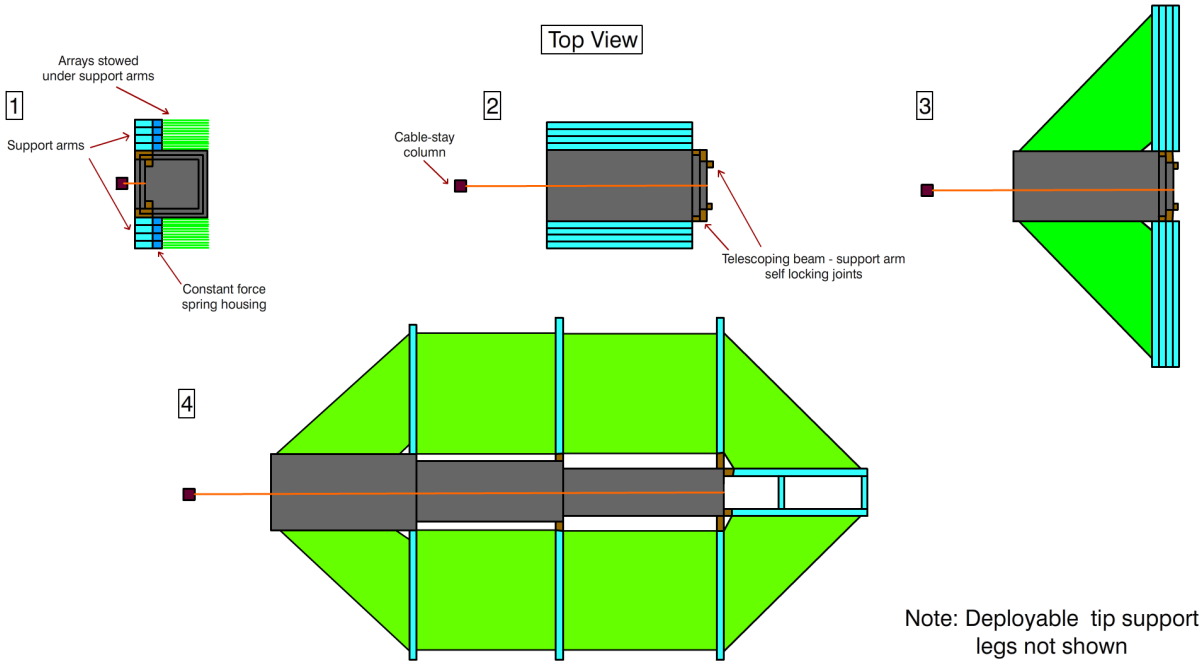


Figure 4.13: The RCTSA is a supposed upgrade to the original CTSA design proposed by NASA. Looking akin to a leaf pedal, the RCTSA makes use of unused space at the root and tip of each array wing in order to decrease the total deployable length of the central telescoping structure. The solar array membrane is divided into segments which stow underneath each support arm as shown. A cable-staying column attached to the tip of the structure stabilizes the deployment.

Assembly Overview The objective of this concept was to see if there were elements of the original concept serving as a baseline for this thesis that can be improved. Features included from the original design are the following: discretization of the total solar array area into several identical wings, use of a central telescoping boom flanked by stretched solar array membranes on either side, and deployable support legs at the tip. Adjustments from the original design are as follows. Firstly, a cable-staying column is included at the root of the telescoping beam to stabilize the deployment process. Secondly,

instead of having one continuous solar array blanket membrane on either side of the central telescoping beam, the membrane is broken up into several smaller segments supported by deployable support arms located at every telescoping section junction, as shown in Figure 4.13. These deployable arms are of equivalent span to that of each telescoping beam section in order to maximize stowed volume efficiency. Thirdly, in an effort to reduce the total telescoping beam deployment length, deployable solar array membranes are included at the root as well as at the tip, as shown.

Packaging The central telescoping beam of each wing nests within itself in the stowed state. Furthermore, the array segments z-fold and stow underneath each support arms as shown in Figure 4.14. These support arms, in turn, fold and stow flush on either flank of the telescoping beam.

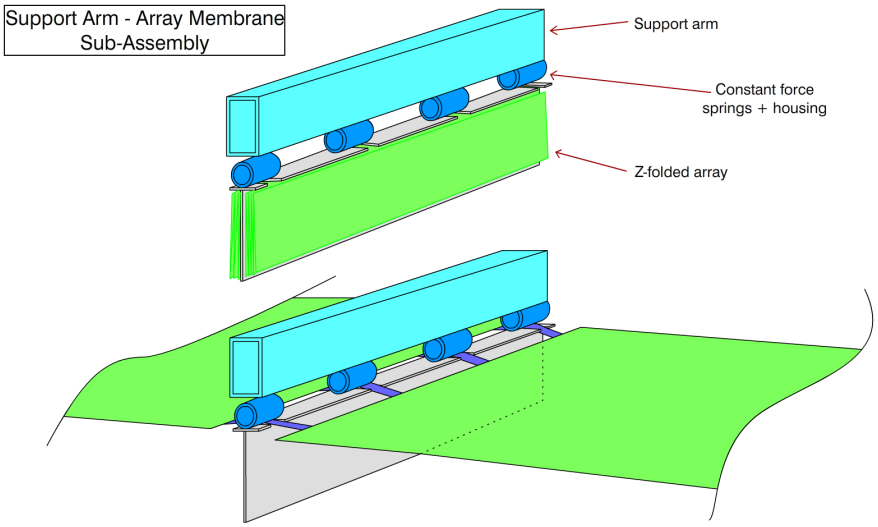


Figure 4.14: The following figure shows a preliminary concept for the support arm sub-assembly. Specifically, the support arm shown in light blue provides connection points the z-folded arrays stowed underneath as well as for spaced spring containers the contain constant force springs. These springs (shown in purple) connect to the edge of the z-folded array and apply a stretch force on the membrane when deployed.

Deployment Starting from a vertical position, parallel with the vertical cable-staying mast, the telescoping assembly is allowed to rotate 90 degrees into a horizontal position. The rate of this maneuver is controlled by the rate in which the staying cable is allowed to lengthen. Again, this cable is connected to the top of the cable-stay column and the tip of the telescoping beam. Next, the support bars located on the flanks of the telescoping assembly rotate 90 degrees to a perpendicular position in reference to the telescoping beam and lock into position. Extension of the telescoping beam occurs next which, in turn, removes the z-folded arrays from their stowed positions and stretches them. After the telescoping beam fully extends, the deployable legs that attach at the tip of the telescoping beam flip down. Thereafter, the final support bars rotate around the tip of the telescoping beam and connect with one another.

Stabilization To stabilize the deployment process, a staying cable is used. Each segment of the telescopic boom includes a self-locking mechanism as well as a small overlap to improve joint stiffness. As with every thin membrane solar array described thus far in this thesis, tension is applied to the membrane in order to improve its out-of-plane stiffness. This tension is applied, however, through the use of constant force springs located underneath each support arm.

Dust Mitigation Dust mitigation technologies integrated in this concept are integrated shaker elements (piezoelectric devices) installed in each array segments.

Advantages and Disadvantages The main advantage of this design is the segmentation of the array within each wing into smaller sections as well as the reduction in overall deployment length of the telescoping structure by means of the additional solar array areas at the tip and root. Segmentation of the array not only improves manufacturability and repairability of the array but also reduces the required amount of tension required to support each segment. This reduces the compression load on the central telescoping beam. Furthermore, a cable-staying column is a weight efficient method to ensure the proper deployment of the telescoping beam.A potential disadvantage of the design is the additional mass and volume associated with the support arms.

4.2.7. Concept 7: Canopy Design

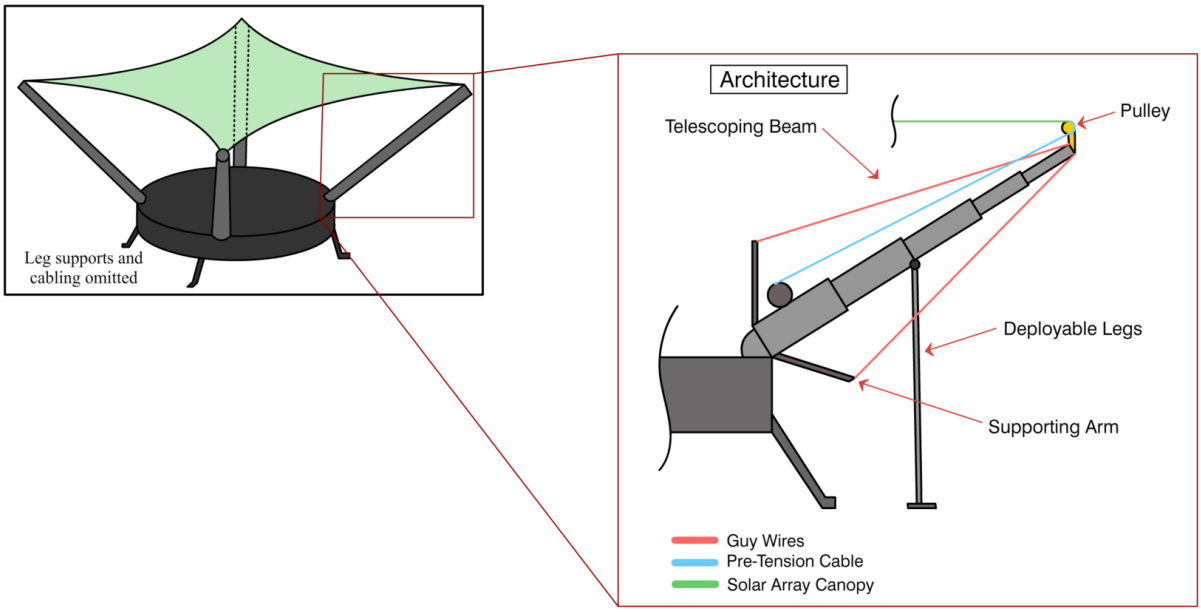


Figure 4.15: As the name suggest, the Canopy Concept involves a star shaped, 4 point connected canopy solar array membrane that is supported by four telescoping support legs. Tension is applied to the membrane at each of the four corners by pulling on the pre-tension cable that connects to the membrane, routes around a pulley at the telescoping support tip, and attaches to at the base of the support as shown.

Assembly Overview The concept centers around the idea of a single, large slip wrapped array canopy which is stowed at the tip of one of four evenly distributed telescoping support members around the circumference of the lander deck. These supports connect to the slip wrapped array through lightweight, high performance polymer cabling and offer connection points for the stowed array to be unfolded and supported its four corners when deployed. Each support can extend, has 1 DOF rotation, and has additional supporting elements as shown in Figure 4.15.

Packaging Structural supports in this concept are divided into telescoping segments and nest within each other to achieve high packaging efficiency. Slip wrapping was used as the primary method of packaging the solar array due to its high packaging efficiency and synergy with the stowage method of the supports. When stowed, the cabling which interconnect the tips of each telescoping support to the slip wrapped array have to be attached at different locations around the extremities of the lander in

order for them to not interfere with the additional payload at the center of the deck (Figure 4.16)

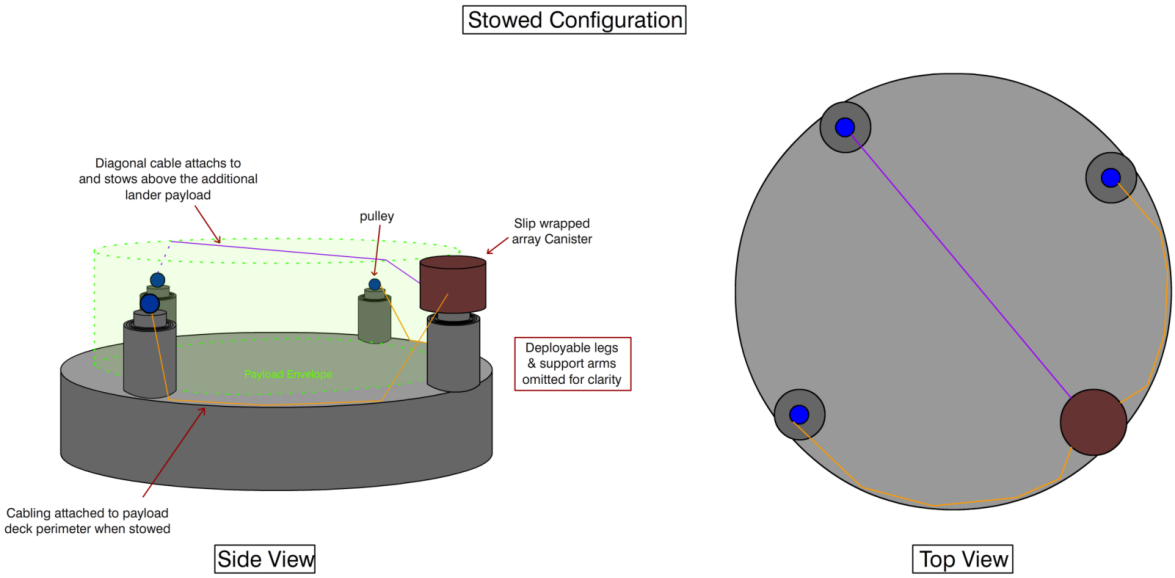


Figure 4.16: In the stowed configuration, the entire canopy membrane is slip wrapped into a small cylindrical container which stows on top of one of the telescoping beams.Because the center of the lander deck has additional payload, precautions are taken to move the cabling imperative to the deployment process out of the way as shown.

Deployment The deployment sequence is as follows and is shown in Figure 4.17. Each support telescopes upwards slightly from its stowed position to a height where the aforementioned stowed cable can be released and not interact with the payload at the center of the lander. Each support than rotates outwards less than 90 degrees as shown to its deployment angle and telescopes outwards. This motion, combined with applied tension on the cabling which attach at the flanks of the stowed array (shown in orange in Figure 4.17), unfurl the array out of its stowed configuraiton. At a certain point, the canister housing the array must detach its inner wall facing inwards towards the lander to allow the spread of the array solar array canopy over the lander. Prior to beginning the process of unfurling the solar array from the canister, the deployable legs as well as the support arms for each telescoping beam deploy and support the telescoping beam as it telescopes outwards.

Stabilization Stabilization is provided through self-locking mechanisms which lock adjacent telescoping sections together in place. Additional overlap is included at these telescoping section interfaces to improve joint stiffness. Tension is provided to the array through a system of cabling connected at each of the four telescoping supports. This tension provides flexural stiffness to the thin solar array membrane. Motors located at the root of each support apply the aforementioned tension to pre-tension cables which route around pulleys at to deployed tip of each telescoping support and attach to each membrane canopy corner. Additionally, deployable legs and support arms with guy-wires assist the long telescoping supports in maintaining their shape in the presence of gravity and large bending loads.

Dust Mitigation Dust mitigation technologies integrated in this concept are integratable piezoelectric shaker elements. Because this concept utilizes slip wrapping, small slits are present along the fold lines which offer avenues for shaken dust to vacate the canopy surface. Increasing the number of folds reduces the transport distance for shaken dust to be removed.

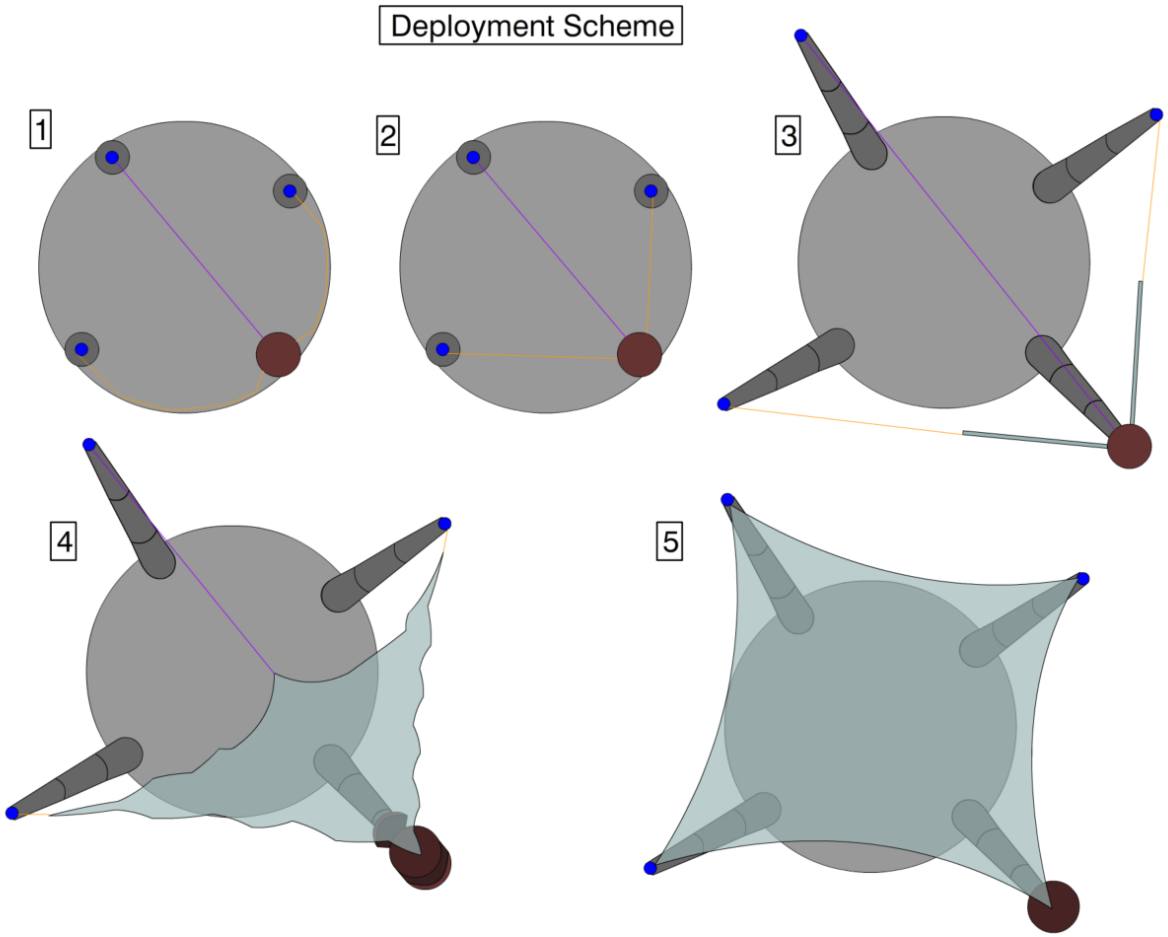


Figure 4.17: The deployment process goes as follows: 1) The array is in the stowed configuration. 2) Each of the telescoping beams telescope upwards from its vertical stowed position to allow clearance for the interconnecting cables which release during this process. 3) Each telescoping support begins to angle outwards as shown. Tension is applied to the 2 orange cables to begin unfurling the array from the housing. 4) Towards the end of the deployment process, the array housing must remove its inner wall to allow the purple cable to pull the last corner of the canopy out as shown. 5) The array is in its fully deployed state. Tension is applied at each corner to pre-stress the membrane to stress stiffen it.

Advantages and Disadvantages A striking advantage of this design is it reduces the number of required telescoping supports to just four. Moreover, since the array deploys above the lander, the actual deployment lengths of these supports are shorter in comparison to previously described concepts. This provides weight savings. The cable tensioning system for giving the array out-of-plane stiffness can be easily adjusted depending on the wind load conditions making the design rather versatile. Furthermore, because the array stows on top of one of the telescoping supports, space on the lander deck is efficiently utilized as every element of the solar array assembly is out of the way of the deck center. A major drawback to this design is that the deployment process in itself is rather complex and involves several steps and moving pieces. This introduces risk into the system. Another potential shortcoming of this design is that the deployment height above the lander may have to be high in order to prevent the membrane from deflecting under high loads and hitting the payload underneath.

4.2.8. Concept 8: Stripped Array

Assembly Overview This concept is an adaptation of the the Canopy concept with a number of important distinctions. Namely, instead of a singular membrane this concept includes a plethora of strips which span four quadrants, each separated by a diagonal cable made of high specific stiffness polymer rope. Each strip connects to a singular point at each end along two of the diagonal cables, as shown in Figure 4.18. Four telescoping supports around the periphery of the lander support the

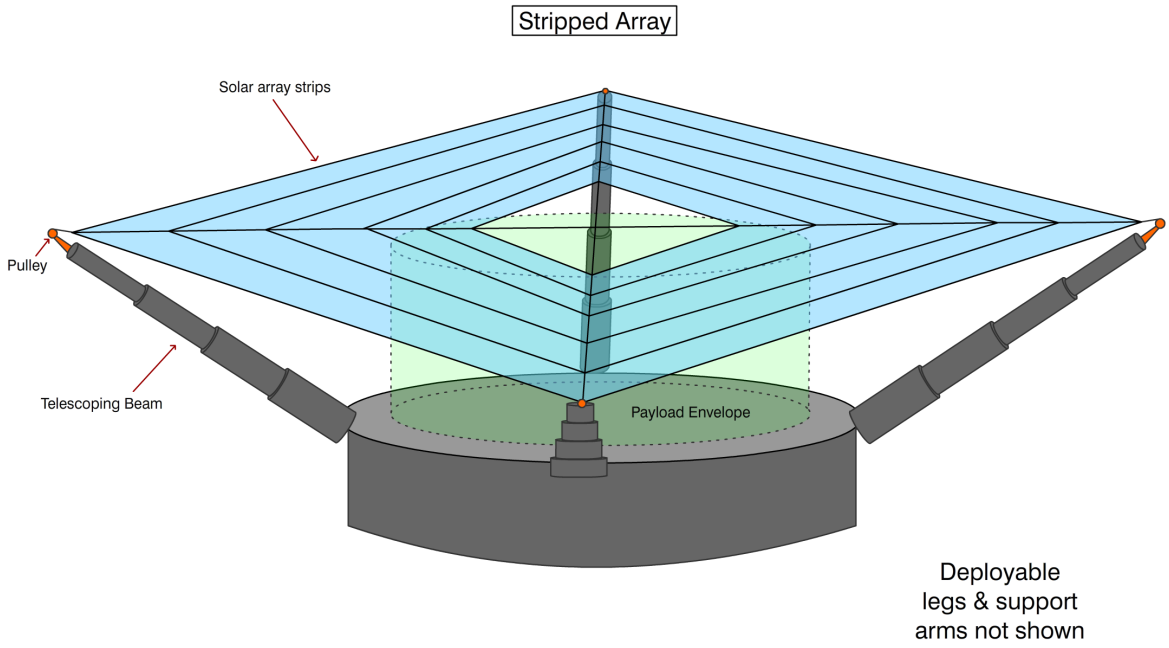


Figure 4.18: The Stripped Array concept deploys above the lander deck as shown and is supported by telescoping support legs with the same configuration as what was shown in Figure 4.15. Importantly, the array membrane here stows at the center of the lander above the payload and consists of four separate quadrants of solar array strips.

stripped membrane above the lander.

Packaging The packaging method for this concept is similar to the that of which is used in the Canopy concept but adapted for slip wrapping a membrane with four arms rather than two [79]. See Figure 4.19. Specifics of this method are explained in detail later. Instead of stowing on one of the telescoping supports, this slip wrapped membrane is intended to either rest on top of the additional payload like a habitation module at the center of the lander or atop of a short column which elevates it above the payload. As with the other concepts which utilized telescoping supports described in this section, the telescoping sections nest within one another in the stowed state.

Deployment The deployment process is straight-forward. First the telescoping supports rotate to their deployment angle and telescope outwards. At the half way point, deployable legs are deployed to support the beam for the remainder of the deployment process. Thereafter, the telescoping beam continues to extend to its final deployed position. Motors at the root of each support apply tension to pre-tension cables which route around pulleys at the tip of each telescoping support and connect to the diagonal cables. This tension unfurls the diagonal cabling and strips from its quad slip wrapped state at the center of the lander. A series of guides and release mechanisms for the rollers shown in Figure 4.19 are required in order for the wrapped array to completely exit its stowed state.

Stabilization Applied tension to the pre-tension cables stretch the diagonal cables slightly which transmits stress to the connected strips. This stress provides a stress stiffening effect to each of the strips. Each strip contains a series of thin reinforcing cables that form an open mesh that follow the length of the strip and carry this aforementioned load. This mesh attaches at either end to a stiff spreader bar, as shown in Figure 4.20, which connects to a single point on the diagonal cable through connector cables.The open mesh reinforcement is also intermittently connected to thin Kapton sheets to which the solar cells are attached too with an adhesive. In terms of stabilization characteristics of the telescoping supports, the telescoping support structure here is largely the same as that which was

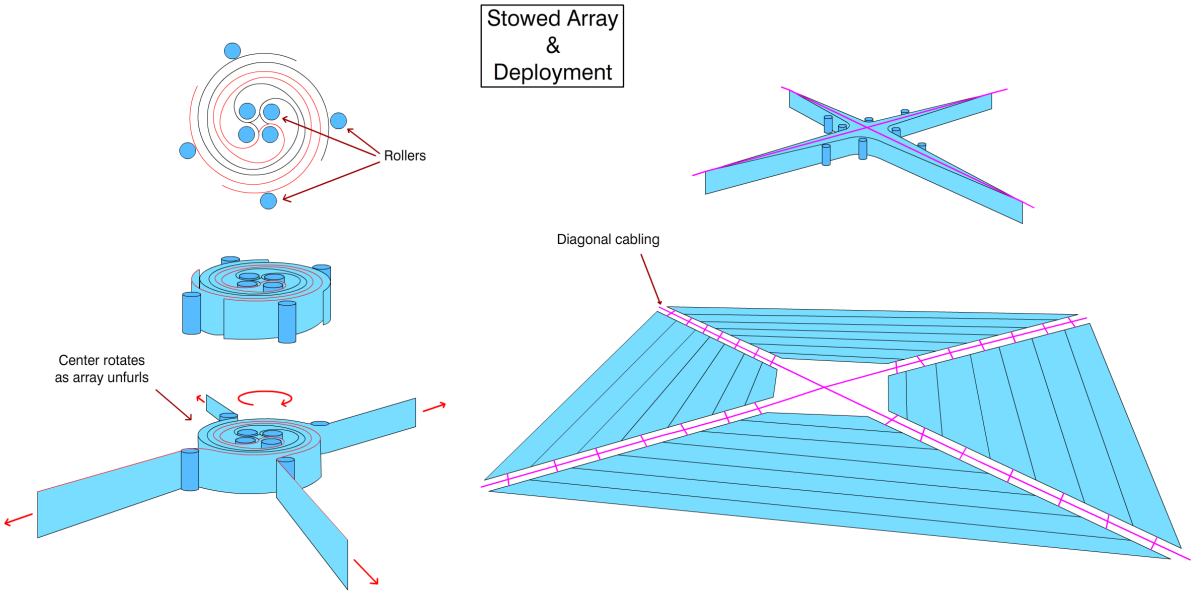


Figure 4.19: The array is stowed in an adapted slip wrapping method as shown. To deploy the array, each of the four ”arms” are pulled out as shown while the central hub unfurls to allow this pulling motion.Before the strips can fully deploy, the rollers at the center and the extremities of the array container must move out of the way. Doing so allows for the full release of the stripped array into the configuration shown.

shown earlier for the Canopy concept in Figure 4.15.

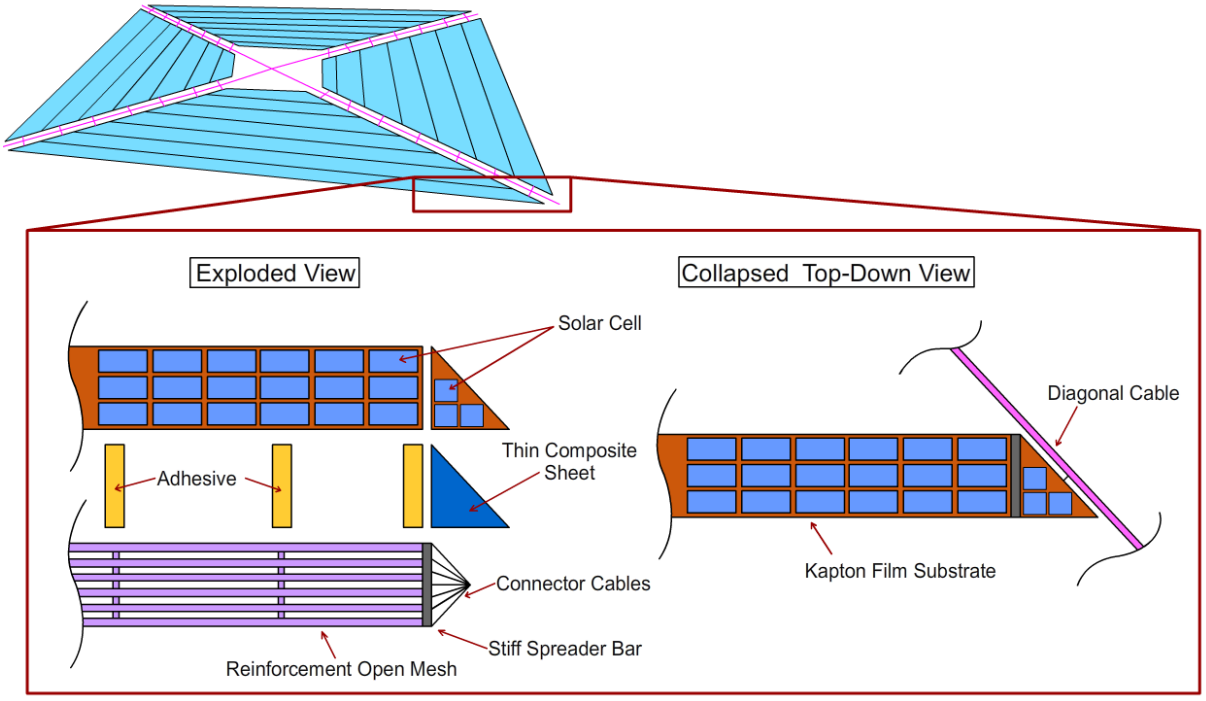


Figure 4.20: A detailed look at the possible construction of each solar array strip.

Dust Mitigation Integratable piezoelectric shaker elements in each of the solar array strips can also be used here to remove dust. Furthermore, because each quadrant consists of a plethora of

strips with single point connections to a diagonal cable at each end, it may be possible to include systems of interconnecting cables between strips to allow them to feather. Feathering could be done in the evening hours or during high wind load conditions to resist dust deposition and high loads, respectively.

Advantages and Disadvantages The main advantage of this design is the discretization of the solar array into independent strips. Manufacturability, testing and qualification, repairability, and others have potential benefits from this design choice. For example, if a strip were to be damaged during service, it could be replaced without replacing the entire solar array system. Furthermore, because the system is discretized into strips, manufacturing, qualification, and assembly can be done in tandem. A prominent potential downside to this concept is the fact that it requires the array be stowed in the center of the lander deck thereby introducing design dependency on the contents of the lander.

4.3. Concept Down Selection

Concept down selection was performed in a simple, two step process. The first step involved initial screening of all the concepts generated to narrow down the pool of options for further, more detailed investigation. This involved challenging the validity of underlining assumptions associated with each concept and assessing at a top level which set of concepts has the most potential for being competitive with the baseline design.The second step involved developing detailed evaluation metrics for the remaining concepts and comparing them using an Analytical Hierarchy Process (AHP) which is a general decision making tool used across engineering disciplines. A detailed explanation of the AHP is presented in Appendix A. The purpose of using this selection format was to improve the time efficiency and quality of the evaluation process. The pre-screening, while subjective in nature, allows for more time to be devoted towards assessing concepts that are believed to have a higher likelihood of competitiveness with the baseline concept. Furthermore, use of AHP and deriving a series of evaluation metrics reduces the overall subjectiveness of the decision making process.

4.3.1. Pre-screening

Of the concepts generated, the Canopy, RCTSA, and Stripped Array Concepts were ultimately chosen for further evaluation. For the Nested Membrane Design, form factor and scalability were both a concern. If the solar array area were to be increased, for example, either the length of the I-beam sections must increase or additional members should be added to the folded hinge pattern. The former results in an increasingly unreasonable form factor in terms of height that may pose problems in terms of launch load survivability and the latter results in a complex hinged structure whose architecture may or may not even be feasible. While an interesting concept, form factor was again a concern for the Telescopic Slider Design. Specifically, stowed width can become untenable as slider count or deployment length increases. Assuming each slider has the same cross-section, the concept width increases by two times the slider cross-sectional width each time an additional slider segment is added. Even if the slider count remains the same and deployed length increases, the required cross-section for adequate structure support will increase which , based on the aforementioned relationship, may result in exceedingly wide stowed arrays. One of the principle innovations of the Cellular Blind Design, the pressurized gas dust cleaning system, was the reason for forgoing this design. Specifically, the concern here is that tubing and systems required for collection, compressing, and transporting gas adds too much complexity to the design. Furthermore, this dust mitigation system may add too much parasitic mass. Moreover, the cellular array design may not be a weight efficient means to support the array in comparison to simply tensioning a single layered membrane. For the Crane Deployment Design, the crane system is an interesting concept because it utilizes a system that could already be on board to unload other cargo components. Consequently, the mass of the crane system is already debited elsewhere meaning this concept could have the potential of being very mass efficient. The concept employs gravity assisted deployment which can further improve the mass efficiency of the design. Nonetheless, since this concept is completely dependent on the inclusion of the crane system, it was determined that it may be better to investigate concepts with better, more general applicability. Lastly, the Scissor-Structure Design was omitted because of stowed volume concerns. In its compact form,

each array wing would likely stand exceedingly tall and protrude too far into the interior of the lander thereby presenting payload packaging limitations.

4.3.2. Evaluation Metrics Definition

In order to compare conceptual designs, evaluation metrics were first defined. Concept mass, stowage volume, dust mitigation capability, and risk were all identified as the most important assessment criteria for concept comparison. In order to reduce subjectivity in the evaluation process, quantitative comparisons were made when possible using simple models or computations. Predictably, mass and stowage volume were included as these are very critical parameters given the extreme weight and size limitations involved with any structure used in space applications. As mentioned previously, Mars's thin atmosphere makes landing heavy spacecraft on its surface extremely difficult which adds additional weight limitations. In terms of specific mass and volume targets, the objective of the concept generation is to come up with concepts which beat the performance of the CTSA for a total array area of $1000m^2$. A grouping of six CTSA wings can articulate this area requirement which together occupy roughly $10m^3$ of space and weighs $1500kg$ in total. Consequently, these values were used as the basis for comparison.

More in regard to the mass metric, mass performance of each concept was evaluated on four categories: max deflection, column buckling, local buckling, and material failure. Max deflection refers to both the membrane component of the array as well as the telescoping architecture. While no specific target was established for this category, the ideal situation is minimum membrane and telescoping support deflections in response to max wind load conditions.

Dust, on the other hand, was highlighted by NASA as one of the biggest threats to the successful application of solar arrays on the surface of Mars. Naturally, it was deemed necessary to include concept performance with respect to dust mitigation as an evaluation metric. However, differences in concept dust mitigation strategies among the down-selected designs are minimal as concepts which integrated technologies like compressed air or brushes were eliminated in pre-selection phase. During the literature review portion of this thesis, it was determined that there are several promising active dust mitigation technologies already well into development that given their nature can easily be integrated into a solar array membrane. Consequently, any of the down-selected designs could feasibility use them leaving little difference with regard to performance. Thus, it was omitted for this evaluation process. Lastly, concept risk was also included in the evaluation criteria despite it typically being excluded during this stage in the design process. Thorough evaluation and quantification of risk is difficult for designs that have not been rigorously defined yet and the quality of the assessment thereof relies heavily on the knowledge and experience of the practitioner doing the work. Regardless of this uncertainty, the earlier that potential sources of risk are identified in the design process, the better the chances of mitigating them without incurring significant unplanned program costs later on.

With regard to the mass metric, it was assumed that mass associated with the thin solar array membrane would be relatively consistent across all concepts and comprise 500 kg of the allotted 1500kg for the entire assembly. Consequently, a concept's performance was evaluated by how weight efficient the supporting architecture was and how many actuators are required. Volume was assessed in a similar fashion, where the stowed dimensions of structural hardware was quantified. It should be noted that in NASA's baseline design, large deployable legs were included but were likely neglected in final accounting of the concept's volume. Consequently, any concept which included deployable support legs did not include them in the volume computations. Dust mitigation performance was measured qualitatively, where designs that included more features or methods for avoiding dust collection received higher ratings. It should be noted that each of the down selected designs assumed integrated piezoelectric shakers or an EDS system in the solar array membrane as active dust mitigation controls. As a result, concept-to-concept abatement performance is the same. The purpose of this metric is to identify if a concept accommodates better preventative controls in the structural design.

Concept risk is a blanket term that was divided into several sub-categories: Requirement, Mission Operation, Design and Integration, and Testing and Qualification Risks. Requirement risk refers to

risks associated with the preliminary requirements utilized in this thesis and how each concept fairs if these requirements were to change. Given the high amount of uncertainty surrounding a human mission to Mars, it is highly probable that the requirements used in this thesis will change moving forward, underscoring the importance of this evaluation metric. Lander clearance angle, array segmentation sensitivity, and concept scalability were all chosen as the factors to assess. While there is no established requirement with regard to the clearance angle or array segmentation, each are still important design variables to understanding concept performance. Lander clearance angle refers to the amount of space required to be left free of deployed solar arrays for access to the lander and its payload deck by astronauts, offloading equipment, transport or robotic vehicles, etc. In terms of scalability, each concept was evaluated on the basis of changes in total array area.

Operational risk is a measure of concept rigor which refers to how reliable a concept is in performing its intended functions. Concept modularity, repair-ability, and retract-ability, for example, are all factors which positively influence a concepts operational risk given the fact that these elements all improve the durability and longevity of a concept. Characteristics of the deployment process like number of steps or complexity of the process contribute to a concepts reliability as well. More deployment steps are generally treated as higher risk given the fact that the system has more opportunities to fail.Moreover, systems with more mechanical hardware or moving parts are assumed to score lower on reliability ratings. Lastly, concepts which consist of several smaller, identical deployable systems pose less operational risk than one large assembly because failure of one of the deployable systems does not equate to total loss of mission power generation capability. In effect, several smaller arrays is less risky than a singular, large array.

Design and integration risk refers to risks associated with the manufacturability of the concept as well as the technological readiness level (TRL) of the systems and hardware inherent to the design. Concepts which can be produced with conventional or established manufacturing practices or consist of simpler geometries pose lower overall risk. Additionally, designs which possess mechanisms and components with high TRL scores also possess lower risk. TRL scoring ranges from one to nine, where one refers to basic principles observed and reported and nine is attributed to hardware or systems with successful in service application [80]. Lastly, testing and qualification risk refers to risks associated with design validation. While the design process in this thesis includes a requirement that each design must be able to survive 1g (Earth gravity) loading for the purposes of easing ground testing, some concepts still have aspects that may make it difficult to test. For example, if a concept has several smaller arrays as opposed to one large array, it is easier to accommodate it in existing laboratory or clean room infrastructure. Furthermore, manufacturing and qualification could potentially be done in tandem for these systems which reduces scheduling risk. That is, while one assembly is being qualified and tested, manufacturing can already commence on the next unit. Ease of inspection is another aspect of this risk measurement.For designs that divide the solar cell membrane sub-assembly into smaller, easy-to-work-with units, it is not hard to understand that such a system is inherently more manageable than a system which does not.

4.3.3. Revised Compact Telescoping Surface Array (RCTSA)

Mass Estimation

Max Deflection As mentioned, deflections of the RCTSA were computed by approximating the concept as a fixed-simply supported beam column subjected to various transverse loads and axial compression (Figure 4.21).Self-weight of the solar array membrane as well as max wind pressure applied normal to its surface are both loads that are transmitted through the wing support arms to the central telescoping structure. These loads, in combination with the weight of support arm themselves, are represented in the beam-column approximation as point loads applied at each of the telescoping beam-to-support arm junctions. With regard to self-weight, Martian gravity is assumed. Furthermore, the weight of the central telescoping beam is represented by a constant distributed load over its entire length. Lastly, an axial compressive load at the beam column tip is included to account for the compression caused by the tensioning of the solar array membrane on each flank of the deployed telescoping beam. Through simple free body analysis of the design, it can be shown that this compression, P, is approximately equal to $2jcos(pi/4)L_{SA}$, where j is the force per unit length tension applied to the

square sections of the membrane and L_{SA} is the length of the support arm.

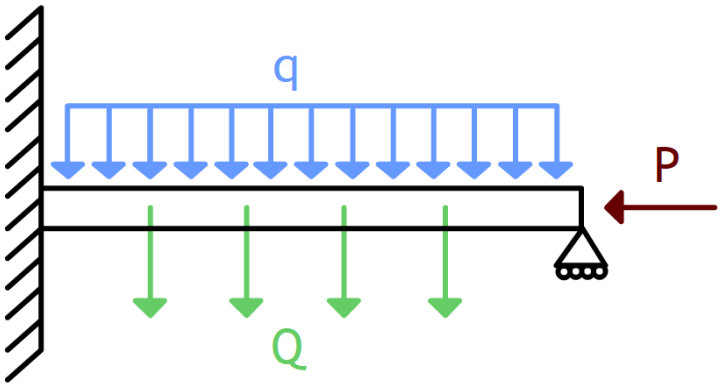


Figure 4.21: A 2-D approximation of the revised compact telescoping array concept with a distributed load q,various point loads represented by Q, and an axial compressive load P. The structure is approximated as a fixed-simply-supported beam column.

Using the method of superposition, this statically indeterminant problem can be solved analytically by dividing the problem into three separate determinant cases: a simply-supported beam-column subjected to a distributed load, a simply-supported beam-column subjected to a moment load around one of the supports, and a simply-supported beam-column subjected to a series of point loads. Taking into account boundary conditions and using solutions to these aforementioned cases, the deflections of the statically indeterminant problem can be solved. The equations and methodology reproduced below to do so are taken from [81].

For a simply supported beam-column subjected to a distributed load, q, its deflection can be represented by the equation:

$$y_{DistributedLoad} = \frac{ql^4}{16EIu^4}\left[\frac{\cos(u - 2ux/l)}{\cos u} - 1\right] - \frac{ql^2}{8EIu^2}x(l-x) \tag{4.1}$$

Where E and I are the material Young's Modulus and cross-section moment of inertia, respectively. Additionally, u is a function of the compressive load on the structure and is equal to $\frac{l}{2}\sqrt{\frac{P}{EI}}$. For a simply supported beam-column subjected to a moment load,M_b, its deflection can be represented by the equation:

$$y_{MomentLoad} = \frac{M_b}{P}\left(\frac{\sin kx}{\sin kl} - \frac{x}{l}\right) \tag{4.2}$$

Where k is $\sqrt{\frac{P}{EI}}$ and M_b is the applied moment. For a simply supported beam-column subjected to a series of point loads Q_n, the displacement curve between two point loads, Q_m and Q_{m+1}, on the beam can be represented by the equation:

$$\begin{aligned} y_{PointLoads} = {} & \frac{\sin kx}{Pk\sin kl}\sum_{i=1}^{i=m} Q_i \sin kc_i - \frac{x}{Pl}\sum_{i=1}^{i-m} Q_i c_i \\ & + \frac{\sin k(l-x)}{Pk\sin kl}\sum_{i=m+1}^{i=n} Q_i \sin k\,(l-c_i) - \frac{l-x}{Pl}\sum_{i=m+1}^{i=n} Q_i\,(l-c_i) \end{aligned} \tag{4.3}$$

Where k is the same relation provided previously, c is the location of the point load on the beam, P is the axial compression load, l is the length of the beam, and Q are the point loads.

The deflection of the indeterminant beam-column in Figure 4.21 is found by summing the deflections from Equation 4.1,Equation 4.2,and Equation 4.3. Prior to doing this however, M_b from Equation 4.2

must be found. This is done by taking the derivative of each of the aforementioned equations with respect to x,substituting the length of the beam-column l for x, summing each derivative together,setting the summation equal to zero, and solving for M_b. Essentially, M_b here represents the reaction moment of the fixed boundary in Figure 4.21. At this connection, the slope of the beam must be zero. Once M_b is found, deflection at any point along the beam can be found.

That said, these deflections can only be solved once the axial compression on the deployable solar array structure is known. As previously mentioned, this compression is a function of the blanket line load tension, j, for the wing solar array membrane. Thus, in order to find a suitable blanket tension a mini-study was first performed.

Since the applied tension controls the amount of sag in the membrane, j should be large enough to reduce this deflection as much as possible. After all, sag adversely affects the generative efficiency of the solar cells on the membrane. At the same time, this tension should also be small enough that it can be realistically articulated by lightweight and compact constant force springs. That said, in the SAWS study it was identified that a blanket tension of $280\frac{N}{m}$ is probably the upper bound in terms of practical spring design [6]. With this in mind, the solar array membrane can be approximated as a cable subjected to a distributed load q as shown in Figure 4.22 assuming the blanket is inextensible, deflections are small, and is supported by a constant axial tension force R_x at each end.

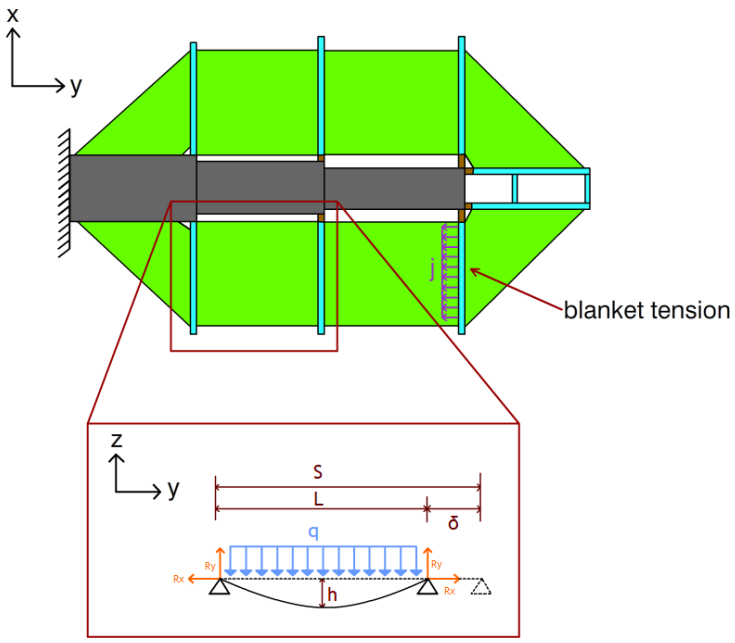


Figure 4.22: A 2D approximation of a segment of the solar array membrane as an inextensible cable, simply supported at both ends. This cable is subjected to an out-of-plane load which is converted to a lineload designated as "q". L is the shortened span of the cable, S is the length of the cable, δ is the axial shortening of the cable, and h is the cable sag.

Using the general cable theorem, the sag in this cable is defined by the following equation:

$$h = \frac{qL^2}{8R_x} \tag{4.4}$$

where L is the span of the sagged cable and h is the maximum sag. Furthermore, R_x is simply the tension per unit width, j, multiplied by the blanket width $w_{blanket}$. In the case of the max wind load applied normal to the solar array blanket, q is equivalent to the pressure imparted on the blanket times the width of the blanket, as shown below:

$$q = \frac{1}{2}\rho_{max} v^2 C_D w_{blanket} \tag{4.5}$$

where ρ_{max} is the max atmospheric density on Mars, v is the max vertical wind speed, and C_D is the coefficient of drag. For this study, these values were $0.023\frac{kg}{m^3}$, $50\frac{m}{s}$, and 1.5 ,respectively. With this in mind, Equation 4.4 can be rewritten as:

$$h = \frac{\frac{1}{2}\rho_{max}v^2C_DL^2}{8j} \tag{4.6}$$

A max wind speed of $50\frac{m}{s}$ was used because in the original NASA study, existing limited wind data suggested that a solar array on the Martian surface would likely experience horizontal and vertical wind speeds up to $100\frac{m}{s}$ and $50\frac{m}{s}$, respectively [6]. Horizontal wind speeds were omitted from analysis due to the fact the array concepts derived herein were all assumed to be static, non-rotating array which deploy parallel to these aforementioned winds. Of course this is a rather significant simplification that most likely warrants revisiting when a better understanding of wind profile on the surface is achieved.

The first subplot in Figure 4.23 shows blanket sag (h in Figure 4.22) versus blanket length at various levels of blanket tension. As observed, $70\frac{N}{m}$ is too low to keep blanket deflections below half a meter where as $280\frac{N}{m}$ is enough for all blanket lengths tested. By comparison, all blanket tension and blanket length combinations when only accounting for blanket self weight have sag below $0.10m$. This sag was computed by replacing q in Equation 4.4 with an assumed areal mass of $0.5\frac{kg}{m^2}$ times Mars gravity ($3.721\frac{m}{s^2}$) times the blanket width $w_{blanket}$.

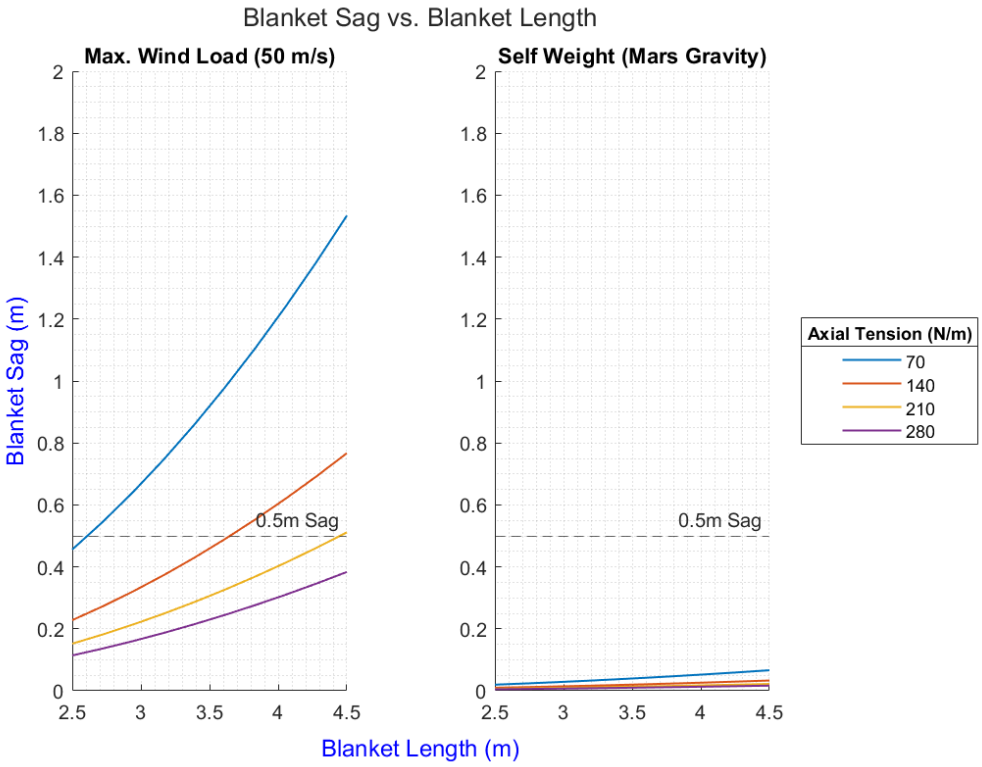


Figure 4.23: Here blanket segment sag versus blanket segment length is shown for various levels of supporting blanket axial tension per unit width j. The left subplot shows sag for blanket segments subjected to the max wind load condition (Equation 4.6). The right subplot shows sag for blanket segments due to self-weight.

While blanket tension as it relates to sag is important, understanding the corresponding amount of axial shortening, denoted as δ in Figure 4.22, as the blanket sags is also paramount. This shortening is equivalent to the required stroke of the constant force springs supporting the blanket. In effect, sag governs the feasibility of the tensioning mechanism where too much axial shortening of the membrane can lead to instability in the springs. That said, axial shortening as a function of sag and sagged cable length is represented by the equation:

$$\delta = S - L = L\left(\left[1 + \frac{2}{3}\left(\frac{h}{(L/2)}\right)^2 - \frac{2}{5}\left(\frac{h}{(L/2)}\right)^4\right] - 1\right) \tag{4.7}$$

where δ, S, h, and L are the axial shortening, the length of the blanket, the blanket sag, and the sagged length of the blanket, respectively. Per Figure 4.24, $0.5m$ of sag for a $2.5m$ and $5m$ long blankets results in a total axial shortening of approximately $0.26m$ and $0.13m$, respectively. For either option, these values are rather large, meaning maintaining a sag below $0.5m$ is likely better.Understanding that $280\frac{N}{m}$ is likely too large for practical constant force spring design and that $0.5m$ of blanket sag is probably too large for the small blanket lengths tested, a blanket tension of $210\frac{N}{m}$ (yellow line in Figure 4.23) was thus assumed for the purposes of this preliminary analysis.

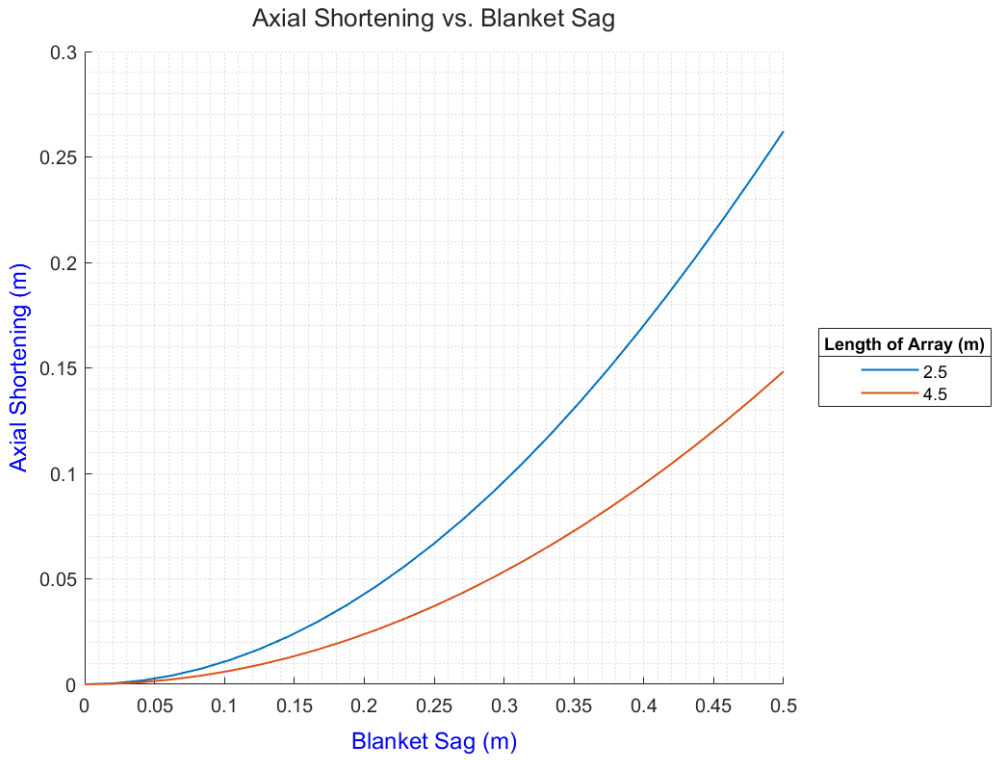


Figure 4.24: Axial shortening as a function of blanket sag is plotted for two different array segment lengths, $2.5m$ and $4.5m$.

Euler Buckling To estimate the column buckling resistance of a RCTSA wing, the central telescoping beam was approximated as a constant cross-section beam in compression with fixed and simply supported end conditions. As stated in the previous section, the compression on the beam during max loading conditions is a function of the constant spring force tension applied to the wing's solar array membrane, j, as shown in the equation:

$$P = 2jcos(pi/4)L_{SA} \tag{4.8}$$

With this compression force known, resistance of central telescoping beam to column buckling can be determined by taking the ratio of this load with respect to the critical column buckling load for a fixed-pinned column as shown in Equation 4.9. L_{Wing},E, and I are the length of the deployed wing, Young's Modulus of the material making up the telescoping beam, and the moment of the inertia. A ratio of over one in this equation implies that the structure buckles. It should be noted that the structure is assumed to buckling in the direction of the transverse loads and is thus only evaluated for buckling in this direction. Furthermore, because the column is subjected to transverse loading, the computed

critical buckling load was divided by a safety factor of 2 to account the column not being entirely straight.

$$\phi_{EulerBuckling} = \frac{P}{P_{crit}} = \frac{2jcos(pi/4)L_{SA}}{\frac{pi^2EI}{(0.699L_{Wing})^2}} \tag{4.9}$$

Local Buckling Local buckling of the central telescoping beam was assessed analytically using the buckling equation for a thin plate undergoing axial compression with unloaded, clamped edges. This equation is the following:

$$\sigma_{\text{crit}} = \frac{k\pi^2 E}{12\left(1-v^2\right)}\left(\frac{t}{b}\right)^2 \tag{4.10}$$

where k,E,v,t, and b are the buckling coefficient, plate material's Young Modulus, plate material's Poisson's ratio, thickness of the plate, and the plate width, respectively. The bottom section of the assumed rectangular cross-section of the telescoping beam incurs the largest compressive loads. This is a consequence of the bending moment in the beam due to the loading conditions shown in Figure 4.21. As a result, b in Equation 4.10 is the width of the bottom section. Furthermore the buckling coefficient given the aforementioned boundary conditions is approximately 7, as shown in Figure 4.25.

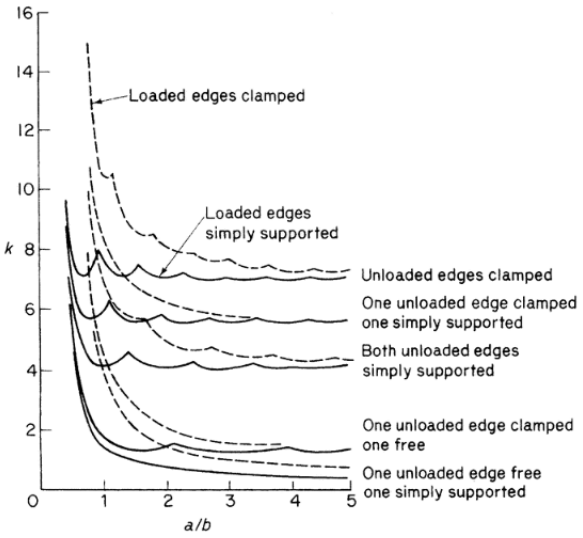


Figure 4.25: Buckling coefficient for thin plates as a function plate length-to-width ratio for various assumed support conditions [82]

As with the column buckling assessment, local buckling of the top plate can be determined by taking a ratio between the max compressive stress in the beam over the critical local buckling stress as shown below:

$$\phi_{LocalBuckling} = \frac{\sigma_{max}}{\sigma_{crit}} = \frac{\frac{M_{max}h}{2I}}{\frac{k\pi^2E}{12(1-v^2)}\left(\frac{t}{b}\right)^2} \tag{4.11}$$

M_{max},h,and I are the max bending moment in the telescoping beam, the cross-sectional height of the telescoping beam, and the moment of the inertia of the telescoping beam, respectively. M_{max} in this case is simply the bending moment at the clamped root of the telescoping beam solved for earlier ,M_b.

Material Yield After computing the maximum stresses used in the local buckling computations, an evaluation of material yield is rather straight forward. A ratio between the maximum stress and the compressive yield strength, as shown below, was computed.

$$\phi_{MaterialYield} = \frac{\sigma_{max}}{\sigma_{YS}} = \frac{\frac{M_{max}h}{2I}}{\sigma_{YS}} \tag{4.12}$$

Volume Estimation

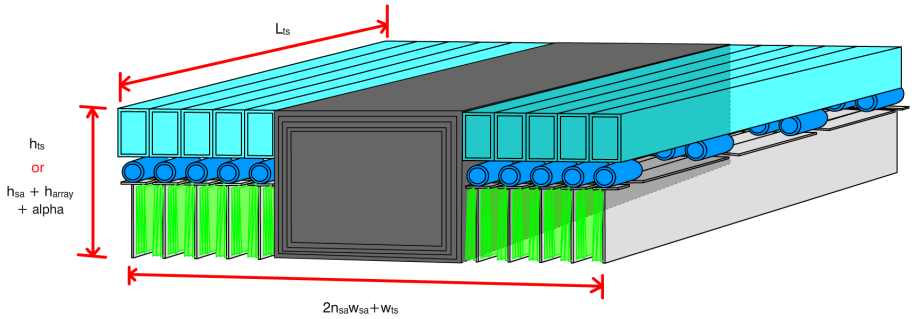


Figure 4.26: Stowed configuration of the RCTSA without the cable-staying column nor the deployable tip support legs. h_ts,h_sa,h_array,α,L_ts,n_sa,w_sa,and w_ts are the height of the central rectangular telescoping beam,the height of the rectangular support arms,the height of the z-folded solar array membrane,the assumed height needed for the membrane tensioning mechanism,the length of the nested telescoping boom, the number of the support arms,the width of the rectangular cross-section support arms, and the width of the stowed rectangular telescoping beam.

Computation of the RCTSA Volume is rather straightforward. A graphical representation of the stowed RCTSA is shown in Figure 4.26. Because the telescoping sections nest within one another and support arms which flank either side of the the root telescoping section remain flush with it, the stowed height is simply the length of one of the telescoping sections, L_{ts}. Ultimately, the assumed lander clearance angle and number of wings influences this parameter. For the purposes of this preliminary analysis, a lander clearance angle of 25 degrees was assumed. Furthermore, the stowed configuration width is simply the the cross-sectional width of the root telescoping section, w_{ts}, combined with the width of one support arm, w_{sa}, times the number of arms which flank either side of the telescoping section in the stowed state, $2n_{sa}$. For this study,w_{sa} was assumed to be $5cm$ Lastly, the depth of the stowed configuration depends on the configuration parameters as this dimension is either driven by the depth of the telescoping section ,h_{ts} or the combined depth of the support arm and the z-folded array and tensioning hardware that nest underneath it. The depth of a stowed support arm sub-assembly is equivalent to $h_{sa} + h_{array} + \alpha$, where h_{sa} is the cross-sectional depth of the support arm, h_{array} is the stowed height of the z-folded array, and α is an assumed constant to account for the additional depth required to house the constant force springs. For this preliminary analysis, α was assumed to be $7.5cm$. h_{sa} is computed for each configuration on the basis that the support arm tip deflects less than a predefined constant ,β, when subjected to self weight and a downward distributed load from the max wind pressure load imparted on the solar array membrane. For this study, the support arm was assumed to be cantilevered under the aforementioned load conditions with a wall thickness of $1.5mm$ and a β of $30cm$. Because the wing is rather geometrically simple, the final wing volume is simply the multiplication of the stowed height, width, and depth. If h_{ts} > $h_{sa} + h_{array} + \alpha$, Equation 4.13 was used. If the opposite was true, Equation 4.14 was used.

$$V = (L_{ts})\,(2n_{sa}w_{sa} + w_{ts})\,(h_{ts}) \tag{4.13}$$

$$V = (L_{ts})\,(2n_{sa}w_{sa} + w_{ts})\,(h_{sa} + h_{array} + \alpha) \tag{4.14}$$

Results

Table 4.1 shows the mass and volume performance the RCTSA sorted by the least total assembly mass for a combined solar array coverage of $1000m^2$. Note, the assembly mass here is the collective mass of every solar array wing combined without accounting for the solar array membrane mass. Furthermore, $TSCount$ refers to the number of telescoping sections, $TSWidth$ and $TSHeight$ are the cross-sectional dimensions of the root telescoping beam section, and $SupportArmWidth$ as well as $SupportArmHeight$ are the cross-sectional dimensions of each support arm. Additionally, $WingWidth$ is the combined length of 2 support arms, $TSLength$ is the length of each telescoping section and by proxy of design the length of each support arm, EB is the column buckling ratio, and LB is the local buckling ratio. That said, hundreds of different configurations were tested and evaluated on the aforementioned criteria. Specifically, telescoping section count ranged between 5 and 10, the number of RCTSA wings was adjusted between 5 and 10, and the cross-sectional height and width of the telescoping sections were adjusted between $0.4m$ to $0.75m$ and $0.3m$ to $0.6m$, respectively. In terms of telescoping section cross-sectional thickness, it was assumed to be constant throughout each section and have a magnitude of $3mm$.

Table 4.1: This table shows a truncated table of the data generated for the RCTSA concept assuming a total solar array area of $1000m^2$.

ID	Number of Wings	TS Count	TS Width (m)	TS Height (m)	Support Arm Width (m)	Support Arm Height (m)	Wing Width (m)	TS Length (m)	Wing Deployment Length (m)	Solar Array Area per Wing (m^2)	Mass per Wing (kg)	Assembly Mass (kg)	Stowage Volme Per Wing (m^3)	Assembly Volume (m^3)	Wing Out-of-plane Deflection (m)	% of YS	EB	LB
445	5	5	0.30	0.40	0.05	0.114	8.94	4.47	22.36	200.00	186.53	932.64	2.20	11.01	-0.13	11.51	0.015	1.803
461	5	6	0.30	0.40	0.05	0.096	8.16	4.08	24.49	200.00	196.57	982.85	2.03	10.16	-0.17	12.81	0.016	2.008
541	6	5	0.30	0.40	0.05	0.096	8.16	4.08	20.41	166.67	165.93	995.58	1.83	10.97	-0.08	8.80	0.011	1.377
477	5	7	0.30	0.40	0.05	0.087	7.56	3.78	26.46	200.00	207.23	1036.17	1.93	9.65	-0.22	13.99	0.017	2.195
449	5	5	0.40	0.40	0.05	0.114	8.94	4.47	22.36	200.00	208.13	1040.65	2.45	12.23	-0.11	9.41	0.012	2.619
557	6	6	0.30	0.40	0.05	0.078	7.45	3.73	22.36	166.67	174.81	1048.89	1.68	10.08	-0.11	9.79	0.012	1.534
446	5	5	0.30	0.52	0.05	0.114	8.94	4.47	22.36	200.00	211.73	1058.65	2.20	11.01	-0.07	8.22	0.008	1.285
493	5	8	0.30	0.40	0.05	0.068	7.07	3.54	28.28	200.00	213.60	1068.02	1.81	9.04	-0.27	15.07	0.019	2.365
465	5	6	0.40	0.40	0.05	0.096	8.16	4.08	24.49	200.00	220.23	1101.16	2.23	11.17	-0.14	10.48	0.013	2.920
573	6	7	0.30	0.40	0.05	0.068	6.90	3.45	24.15	166.67	184.28	1105.68	1.59	9.55	-0.14	10.70	0.013	1.677
545	6	5	0.40	0.40	0.05	0.096	8.16	4.08	20.41	166.67	185.65	1113.89	2.03	12.19	-0.07	7.20	0.009	2.002
462	5	6	0.30	0.52	0.05	0.096	8.16	4.08	24.49	200.00	224.18	1120.88	2.11	10.55	-0.09	9.15	0.009	1.433
509	5	9	0.30	0.40	0.05	0.068	6.67	3.33	30.00	200.00	224.36	1121.80	1.78	8.88	-0.32	16.10	0.020	2.528
542	6	5	0.30	0.52	0.05	0.096	8.16	4.08	20.41	166.67	188.94	1133.61	1.90	11.39	-0.05	6.29	0.006	0.983
453	5	5	0.50	0.40	0.05	0.114	8.94	4.47	22.36	200.00	229.73	1148.65	2.69	13.46	-0.09	7.98	0.010	3.469
589	6	8	0.30	0.40	0.05	0.059	6.45	3.23	25.82	166.67	192.42	1154.50	1.55	9.30	-0.17	11.53	0.014	1.809
525	5	10	0.30	0.40	0.05	0.059	6.32	3.16	31.62	200.00	231.18	1155.92	1.77	8.85	-0.38	17.05	0.021	2.679
481	5	7	0.40	0.40	0.05	0.087	7.56	3.78	26.46	200.00	232.79	1163.96	2.10	10.52	-0.18	11.45	0.014	3.193
450	5	5	0.40	0.52	0.05	0.114	8.94	4.47	22.36	200.00	233.33	1166.65	2.45	12.23	-0.06	6.83	0.007	1.898
637	7	7	0.30	0.40	0.05	0.059	6.39	3.19	22.36	142.86	168.34	1178.40	1.41	9.84	-0.09	8.53	0.010	1.337
561	6	6	0.40	0.40	0.05	0.078	7.45	3.73	22.36	166.67	196.41	1178.49	1.85	11.09	-0.09	8.02	0.010	2.232

As shown, column buckling is not an issue for this concept where as local buckling occurring at the root of the wing is a cause for concern. To mitigate this problem, either additional deployable support legs may need to be included at the center of the deployed wing to reduce the stresses at the root or systems are integrated into each wing that allow for wind load relief such as feathering. Increasing the cross-sectional thickness can help reduce this problem but is not really an option since assembly mass is already exceptionally high where little to no margin exists between the total assembly masses and the requirement of $1000kg$. While only a handful of configurations shown in Table 4.1 do meet the $10m^3$ volume requirement, they all fail the aforementioned mass requirement.

Risk Estimation

Requirement Risk Of the three concepts being evaluated, the RCTSA is the only one where array segmentation (number of wings) and lander clearance angle are relevant parameters. Consequently, a mini sensitivity study was performed for a $1000m^2$ array for various combinations of telescoping section count and number of wings. Again, telescoping section count controls the total width of each wing and ,in conjunction with any given number of wings, controls available clearance to the lander. For this mini study, a telescoping section width was assumed to be $30cm$. That said, Table 4.2 and Table 4.3 shows the affect of telescoping section count and number of wings on the maximum available clearance to the lander in terms of degrees and wing deployment length, respectively. As shown, increasing the number of the wings has the tendency of requiring more telescoping sections in order provide access

to the lander. From Table 4.2, it is evident that wing count has the largest influence on clearance angle. For example, if it is required that at least a quarter of the circumference of the lander is accessible (90 degrees), a 5 wing configuration requires at least 6 telescoping sections which corresponds to a deployment length of approximately $24.5m$. If 6 wings are utilized, at least 10 telescoping sections are required which corresponds to a wing deployment length of approximately $28.9m$ (a 15 % increase).

Table 4.2: The max allowable clearance angle for given combinations of telescoping section count and number wings.

		Max Allowable Clearance Angle (deg)					
Number of Telescoping Sections	5	87.27	43.64	3.64	-	-	-
	6	98.18	58.18	18.18	-	-	-
	7	105.45	69.09	32.73	-	-	-
	8	112.73	76.36	40.00	7.27	-	-
	9	120.00	83.64	50.91	18.18	-	-
	10	127.27	90.91	58.18	29.09	-	-
	11	130.91	98.18	65.45	36.36	7.27	-
	12	134.55	105.45	72.73	43.64	18.18	-
	13	141.82	109.09	80.00	50.91	25.45	-
	14	145.45	112.73	83.64	58.18	32.73	7.27
	15	149.09	120.00	90.91	65.45	40.00	14.55
		5	6	7	8	9	10
		Number of Wings					

Table 4.3: Wing deployment length for given combinations of telescoping section count and number wings assuming a total array area of $1000m^2$. Results are sorted by least mass per wing.

		Wing Deployment Length (m)					
Number of Telescoping Sections	5	22.36	20.41	18.90	-	-	-
	6	24.49	22.36	20.70	-	-	-
	7	26.46	24.15	22.36	-	-	-
	8	28.28	25.82	23.90	22.36	-	-
	9	30.00	27.39	25.35	23.72	-	-
	10	31.62	28.87	26.73	25.00	-	-
	11	33.17	30.28	28.03	26.22	24.72	-
	12	34.64	31.62	29.28	27.39	25.82	-
	13	36.06	32.91	30.47	28.50	26.87	-
	14	37.42	34.16	31.62	29.58	27.89	26.46
	15	38.73	35.36	32.73	30.62	28.87	27.39
		5	6	7	8	9	10
		Number of Wings					

In terms of scalability, the same analysis that was performed in the above sections was repeated herein but for a total solar array area requirement of $1500m^2$. Table 4.4 shows the results from this assessment in the same format as before and sorted by minimum mass. Furthermore, the previous assumptions were maintained here for this assessment as well. Unsurprisingly, an increase in system area to $1500m^2$ requires each wing to have longer deployment lengths. This in turn exacerbates the local buckling issue noted earlier at the root of the structure. Column buckling on the other hand continues to not really be an issue here. In terms of stowage volume, a key element here is how the telescoping section lengths increase in response to the increase of area per wing. The minimum mass configurations tested display an increase in length which as mentioned previously increases the stowed height of the array. The exact stowage height limit is not well understood but it should be noted that while it is not evaluated for in these preliminary computations, every one of these arrays must survive vibrational loads during launch in their stowed form. Excessively tall and slender stowed arrays may pose a risk here.

Table 4.4: This table shows a truncated table of the data generated for the RCTSA concept assuming a total solar array area of $1500m^2$. Results are sorted by least mass per wing.

ID	Number of Wings	TS Count	TS Width (m)	TS Height (m)	Support Arm Width (m)	Support Arm Height (m)	Wing Width (m)	TS Length (m)	Wing Deployment Length (m)	Solar Array Area per Wing (m^2)	Mass per Wing (kg)	Assembly Mass (kg)	Stowage Volme Per Wing (m^3)	Assembly Volume (m^3)	Wing Out-of-plane Deflection (m)	% of YS	EB	LB
793	6	6	0.30	0.40	0.05	0.123	9.13	4.56	27.39	250.00	228.28	1369.65	2.57	15.44	-0.30	0.18	0.022	2.797
805	6	7	0.30	0.40	0.05	0.105	8.45	4.23	29.58	250.00	237.69	1426.16	2.41	14.45	-0.38	0.19	0.024	3.055
869	7	10	0.30	0.40	0.05	0.059	6.55	3.27	32.73	214.29	239.30	1675.09	1.83	12.83	-0.45	0.19	0.023	2.965
821	6	8	0.30	0.40	0.05	0.087	7.91	3.95	31.62	250.00	245.13	1470.78	2.27	13.60	-0.47	0.21	0.026	3.291
697	5	5	0.30	0.40	0.05	0.169	10.95	5.48	27.39	300.00	245.94	1229.71	3.36	16.82	-0.35	0.21	0.027	3.295
837	6	9	0.30	0.40	0.05	0.078	7.45	3.73	33.54	250.00	254.15	1524.88	2.18	13.10	-0.56	0.22	0.027	3.514
797	6	6	0.40	0.40	0.05	0.123	9.13	4.56	27.39	250.00	254.73	1528.38	2.83	16.98	-0.24	0.15	0.018	4.061
713	5	6	0.30	0.40	0.05	0.142	10.00	5.00	30.00	300.00	256.27	1281.36	3.08	15.42	-0.47	0.23	0.029	3.668
794	6	6	0.30	0.52	0.05	0.123	9.13	4.56	27.39	250.00	259.14	1554.84	2.57	15.44	-0.16	0.13	0.012	1.991
853	6	10	0.30	0.40	0.05	0.068	7.07	3.54	35.36	250.00	261.92	1571.54	2.11	12.66	-0.66	0.24	0.029	3.724
809	6	7	0.40	0.40	0.05	0.105	8.45	4.23	29.58	250.00	266.27	1597.61	2.63	15.77	-0.31	0.16	0.020	4.437
729	5	7	0.30	0.40	0.05	0.123	9.26	4.63	32.40	300.00	266.95	1334.75	2.90	14.48	-0.60	0.25	0.032	4.007
806	6	7	0.30	0.52	0.05	0.105	8.45	4.23	29.58	250.00	271.03	1626.18	2.41	14.45	-0.21	0.14	0.013	2.176
701	5	5	0.40	0.40	0.05	0.169	10.95	5.48	27.39	300.00	272.40	1361.99	3.74	18.69	-0.29	0.17	0.022	4.776
745	5	8	0.30	0.40	0.05	0.105	8.66	4.33	34.64	300.00	275.44	1377.20	2.74	13.68	-0.73	0.27	0.034	4.317
825	6	8	0.40	0.40	0.05	0.087	7.91	3.95	31.62	250.00	275.68	1654.06	2.46	14.74	-0.38	0.17	0.021	4.783
870	7	10	0.30	0.52	0.05	0.059	6.55	3.27	32.73	214.29	276.19	1933.32	2.37	16.57	-0.25	0.13	0.013	2.118
698	5	5	0.30	0.52	0.05	0.169	10.95	5.48	27.39	300.00	276.81	1384.04	3.36	16.82	-0.19	0.15	0.015	2.340
822	6	8	0.30	0.52	0.05	0.087	7.91	3.95	31.62	250.00	280.77	1684.61	2.45	14.70	-0.26	0.15	0.014	2.345

Mission Operation Risk With respect to design reliability, the design has a number of advantageous qualities which reduce mission operational risk. Namely , the concept has a couple levels of modularity in the sense the entire assembly is divided into several identical wings where the array component of each wing is segmented into several, identical packaged units. Depending on the extent of damage, a segment of a solar array wing can be easily replaced without sacrificing the entire wing. Furthermore, if one wing fails to deploy, risk of complete power generation loss is lowered given the fact that there are additional solar array wings. An added benefit of this design is the synchronous deployment between the central telescoping structure and array segments which flank it. Specifically, the same system which is used to deploy the telescoping structure is also used to extend the array. This cuts down on number of deployment steps and deployment actuation hardware.

Design and Integration Risk There are several features of this design that improve its manufacturability. Namely, the division of the system into several identical wings, as well as into several smaller identical components like the solar array membrane segments and telescoping sections allows for reduction in custom tooling. Furthermore, telescoping section cross-sectional dimensions as well as lengths are reasonable and are likely feasible to articulate through existing manufacturing methods.

Testing and Qualification Risk Since the design has the array segmented into separate assemblies, and within each assembly, the solar cells are divided into reasonably sized separate sections, testing and qualification can be done efficiently. Thus, testing and qualification risk is expected to be low.

4.3.4. Canopy Solar Array

Mass Estimation

Max Deflection Similar to the RCTSA, evaluation of deflections was separated into two different but interdependent analyses: a solar array blanket deflection analysis and a telescoping support deflection analysis. Prior to evaluating the solar array blanket, formulation of a number of assumptions was first required in addition to characterization of canopy dimensional relations. In order to articulate an isotropic stress state in a square blanketed structure, one method is to modify its edges to include a constant curvature and perimeter cables as shown in Figure 4.27. Achieving an isotropic stress state is important otherwise the thin membrane will undergo zones of compression when pre-tensioned which introduces wrinkling. Ensuring strain compatibility between the thin membrane and perimeter cable is necessary to achieve this ideal state of stress in the membrane. In terms of component sizing, it has been shown that this compatibility allows for the formulation of the following relation between perimeter cable radius and membrane thickness [83]:

$$r_c = \sqrt{\frac{RE_m t_m}{(1-\upsilon_m)E_c\pi}} \tag{4.15}$$

where r_c is the radius of the perimeter cable, R is the radius that the perimeter cable makes with the thin membrane as shown in Figure 4.27, E_m is the Young's Modulus of the membrane material, E_c is the Young's Modulus of the cable material,t_m is the membrane thickness, and υ_m is the Poisson's Ratio of the membrane material. From Equation 4.15, achieving a reasonable radius for the perimeter cable is dependent on reducing the cable-membrane radius, R, as well as the membrane thickness as much as possible. Moreover, the cable material must be stiffer than the membrane material.

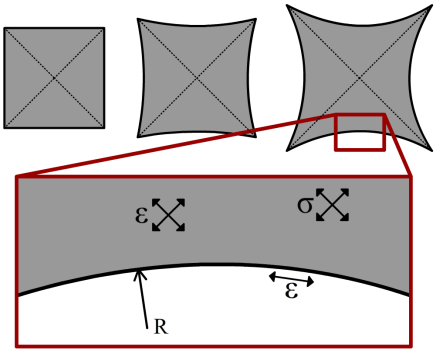


Figure 4.27: In order to articulate an isotropic stress state in a thin square membrane tensioned at its four corners, one method is round its edges as shown and run a stiff cable along the entirety of its perimeter. The radius of each rounded edge, R, as well dimensions and material properties of the membrane and perimeter cable dictate the acquisition of this bi-axial stress state. In short, there needs to be strain compatibility between the perimeter cable and the thin membrane.

R can be found by prescribing the total membrane area, A, and membrane depth, h, (also referred to as "canopy depth" in this report) and solving for θ and R in the following system of equations:

$$\begin{aligned} A &= \left(2R\,\mathrm{Sin}\,\frac{\theta}{2}\right)^2 - 4\left(\frac{1}{2}R^2(\theta - \mathrm{Sin}\,\theta)\right) \\ h &= R\left(1 - \mathrm{Cos}\,\frac{\theta}{2}\right) \end{aligned} \tag{4.16}$$

The parameter θ describes the angle formed between adjacent membrane corners as shown in Figure 4.28.

For the purposes of this preliminary analysis, the membrane thickness was assumed to be $1mm$ and the perimeter cable and membrane materials were assumed to be Spectra 1000 and Vectran, respectively. It should be noted, however, that the reinforcement in the membrane in this concept consists of an open mesh as opposed to continuous sheet of material. Consequently, the aforementioned membrane thickness in combination with an assumed mesh areal mass was used to compute

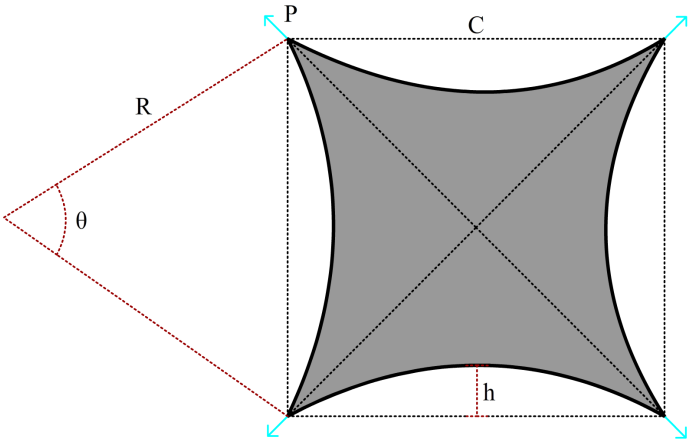


Figure 4.28: Geometric definition of the solar array canopy concept. By prescribing the inscribed area, A, and canopy depth , h, the system of equations shown in Equation 4.16 can be used to define the shape.

an equivalent, continuous membrane thickness. For example, assuming 10 evenly spaced Vectran cables per meter with a $1mm$ in diameter results in an equivalent continuous membrane thickness of approximately, $1.57E-2mm$ for a membrane with an area of $1000m^2$. The equivalent mass per meter squared for Vectran is $21.97\frac{g}{m^2}$. That said, Figure 4.29 shows the relationship between the perimeter cable radius and cable depth, h, for various assumed mesh areal masses. Note, $43.95\frac{g}{m^2}$ and $87.90\frac{g}{m^2}$ areal masses correspond to 20 and 40 evenly spaced Vectran cables per meter, respectively.

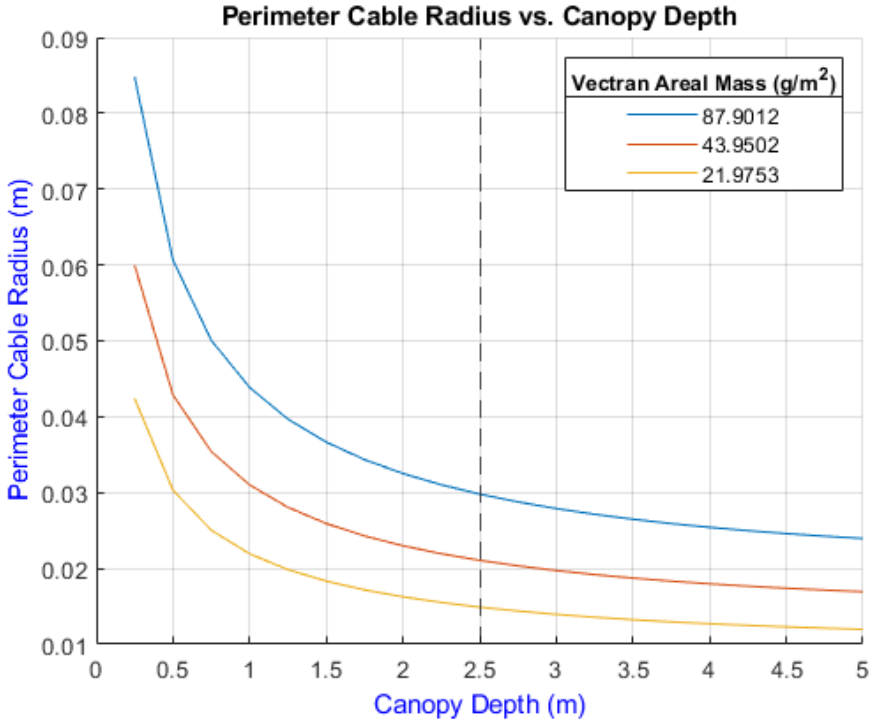


Figure 4.29: Perimeter cable/cord radius as a function of canopy depth, h for various assumed mesh reinforcement areal masses.

For a mesh areal mass of $87.90\frac{g}{m^2}$ and a cable depth of just $0.25m$, the required perimeter cable radius is $0.085m$. This is exceptionally thick and if selected would likely result in an excessive array mass and stowage volume. Conversely, increasing the cable depth by a factor of 10 for the same areal

mass results in a perimeter cable radius of $0.023m$. Decreasing the areal mass of the Vectran mesh results in a thinner equivalent membrane thickness which in turn also results in lower perimeter cable radius, per Equation 4.15. For the proceeding membrane deflection study, a cable depth of $2.5m$ was selected as it results in reasonably low perimeter cable radii for each of the three areal mesh masses shown. Specifically, the perimeter cable radii are $0.015m$, $0.021m$, and $0.030m$ for mesh areal masses $21.97\frac{g}{m^2}$,$43.95\frac{g}{m^2}$ and $87.90\frac{g}{m^2}$, respectively. It should be noted that increasing the cable depth has the benefit of decreasing the required pre-tension load P, shown in Figure 4.28, applied to each corner of the canopy to pre-stress the membrane by the same amount. However, increasing the canopy depth too far presents several issues, namely how efficiently the membrane area can be covered in rectangular solar cells as well as increasing the length of the telescopic support beams connected to the canopy tips. As a final note, a cable depth of $2.5m$ corresponds to a canopy array span, C (Figure 4.28), of approximately $35.15m$ for a $1000m^2$ membrane. For reference, a perfectly square membrane has a span of $31.62m$.

With a cable depth selected for preliminary study, a series of quasi-static finite element analyses were performed in Abaqus using its built in dynamic explicit solver to get an understanding of how and to what degree does the canopy design deflects in response maximum wind load conditions. Specifically, for each of the three aforementioned mesh reinforcement areal masses, a membrane canopy model simply supported at its four corners was built and subjected to various pretension loads at each of its corners through prescribed displacements. After these pretensions are applied, the corners are locked in place. For each level of pretension, a pressure load with the same magnitude as described in subsubsection 4.3.3.1 was applied normal to the canopy. The resultant max deflection as well as the reaction loads at the corners were recorded. An example of one model showing out-of-plane displacement after the pressure load was applied is shown in Figure 4.30. For clarity, the applied pressure is assumed normal to the surface of the canopy and is approximated by using the dynamic pressure equation described earlier, $P_d = \frac{1}{2}\rho_{max}v^2C_D$. ρ_{max} is the max atmospheric density on Mars, v is the max vertical wind speed, and C_D is the coefficient of drag. Again, these values were $0.023\frac{kg}{m^3}$, $50\frac{m}{s}$, and 1.5 ,respectively. The mass of the array is neglected here as it was assumed that the applied pressure load greatly exceeds the loading caused by self-weight.

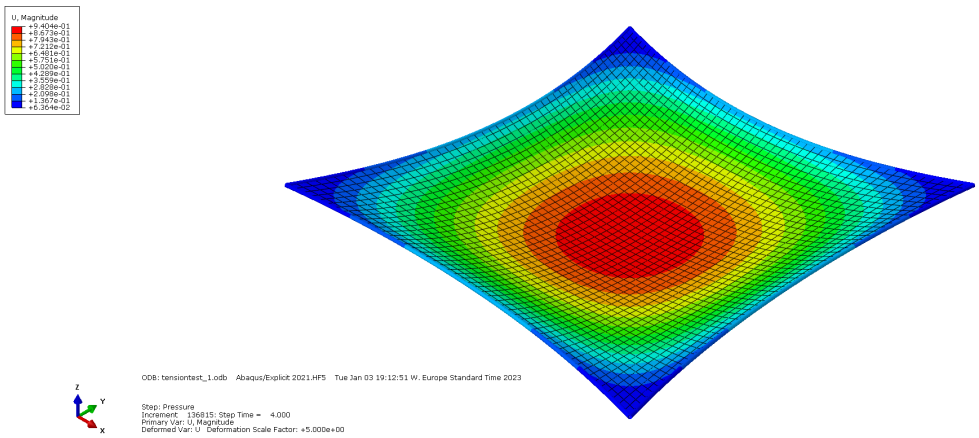


Figure 4.30: Out-of-plane deflections of a quasi-static dynamic Abaqus model of the Canopy concept assuming a max out-of-plane wind load of $50\frac{m}{s^2}$. In terms of canopy dimensions, the inscribed area is $1000m^2$ with a canopy depth of $2.5m$.

Figure 4.31 shows the deflection results of the aforementioned studies. Unsurprisingly, the canopy with the lowest mesh areal mass has the largest amount of out-of-plane deflection for all applied pretension loads. Furthermore, if a limit of $1m$ of out-of-plane deflection is assumed, a $21.97\frac{g}{m^2}$ mesh reinforcement areal mass would require approximately $350kN$ of pretension force applied to each corner. Conversely, a mesh areal mass of $87.90\frac{g}{m^2}$ requires approximately $100kN$. Restricting out-of-plane deflection during max loading conditions is imperative particularly for this design concept to prevent interaction with the stowed payload beneath the deployed array. Relaxing this deflection limit has the

benefit of reducing the required membrane pretension load. As an example, increasing the max. allowable out-of-plane deflection to $1.5m$ for a canopy with a $21.97\frac{g}{m^2}$ mesh reinforcement areal mass, a pre-tension of approximately $100kN$ is required. This reduction, in turn, reduces the incident loading on the telescoping support structure but at the cost of increasing the deployment height of the entire array to, again, reduce the likelihood of interaction between the array and the payload stowed underneath it. Increasing the deployment height increases the deployed length of the telescoping supports which reduces their resistance to buckling and increases assembly mass.

That said, a deflection limit of $1m$ was chosen in combination with a $21.97\frac{g}{m^2}$ mesh reinforcement areal mass for use as input in deflection computations for the telescoping beam supports. That is, the reaction loads at the corners of a canopy with the aforementioned reinforcement areal mass taken at the end of the pressure load step were used to evaluate the telescoping beam supports in the other deflection analysis. While a $350kN$ pre-tension force is required at each corner to resist more than $1m$ of deflection, the reaction force thereof after the wind pressure is applied is approximately $420kN$.

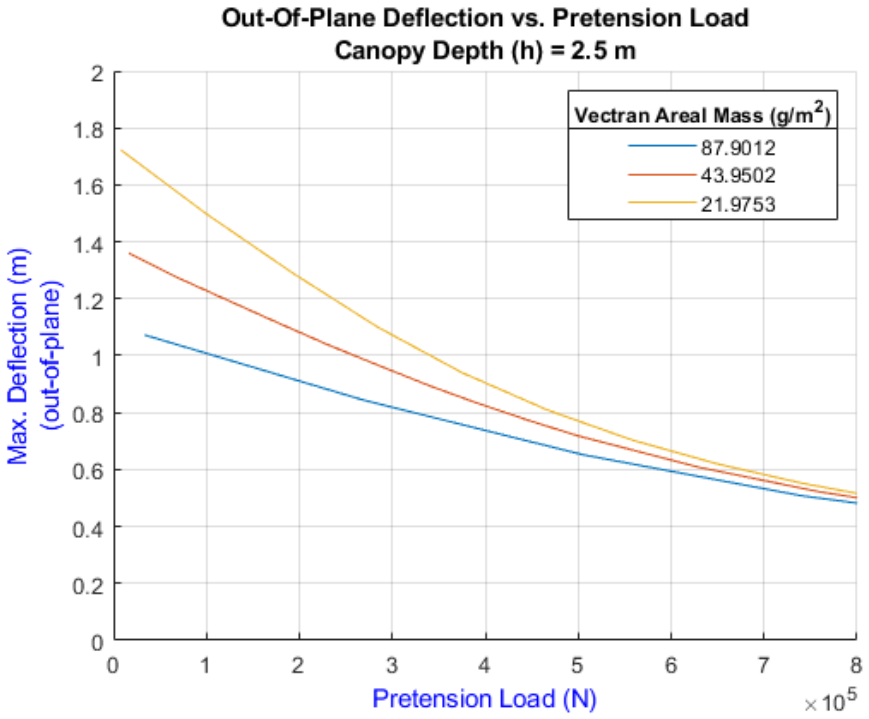


Figure 4.31: Out-of-plane deflection as a function of pre-tension load taken from a quasi-static dynamic Abaqus model of a $1000m^2$ canopy with a canopy depth of $2.5m$ for various mesh reinforcement areal masses

In terms of telescoping beam deflections, a series of simplified static finite element analyses were performed in Abaqus. Specifically, a handful of different cross-sectional radii and thicknesses with regard to the telescoping beam were assessed ranging between $0.3m$ to $0.75m$ and 0.001 to $0.003m$,respectively. As likely already gathered, the assumed telescoping beam cross-section was a cylindrical hollow shell. As shown in Figure 4.32, the telescoping architecture was approximated using a simplified beam model comprised of 2-D Euler-Bernoulli B23 elements where the root of the beam is fixed and the deployable support legs are represented by an angled pinned roller support. The tip of the beam is supported by a pinned roller support as the support guy wires are assumed to restrict transverse displacement of the beam tip. The resultant compressive load , P, on the structure is the combination of several components, namely the axial component of the stabilizing tensions in the guy wires and reaction load derived earlier from the canopy. For these studies, the number of telescoping sections was assumed to be 12 given which was selected on the basis of reducing its stowage volume as will be seen later. Furthermore, the deploy height of the solar array canopy was assumed to be $4m$ to account for a maximum canopy deflection of $1m$ and an assumed spacecraft payload height of $3m$. Lastly, for

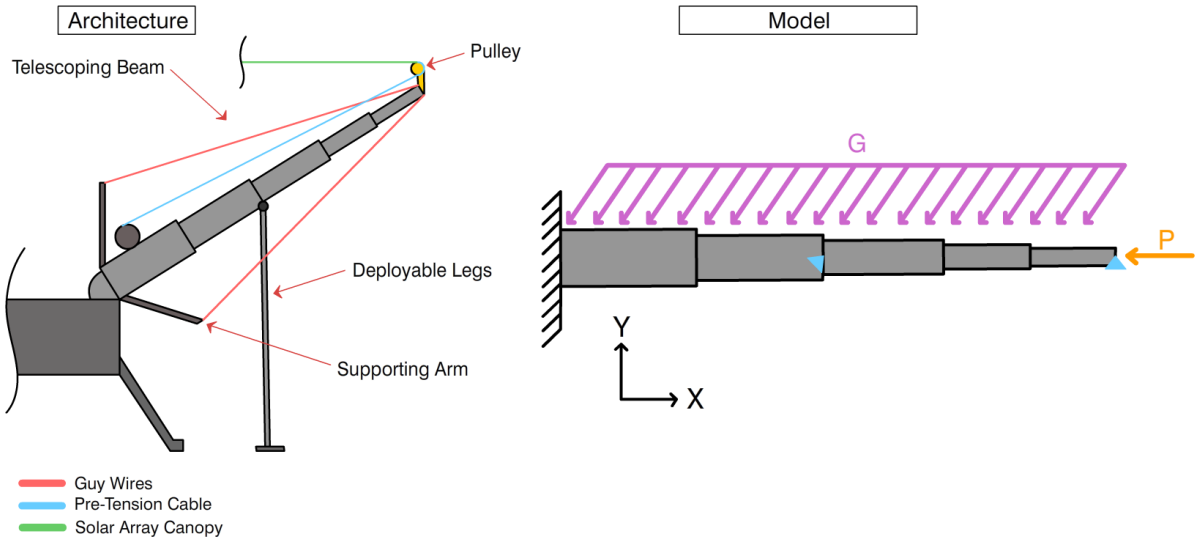


Figure 4.32: Assumed telescoping support architecture, left, and the model approximation used in evaluation, right.

reference, choosing the aforementioned cable depth of $2.5m$ results in a telescoping support length of approximately $20.69m$.

Euler Buckling (EB) Buckling of the telescoping structure was also performed in Abaqus using the same structural model built for the deflection analysis minus the applied loads. Simply put, a single linear perturbation buckling load step was introduced for each of the unique cross-sectional combinations presented earlier in the deflection analysis. Figure 4.33 shows critical buckling loads for various combinations of shell thickness and radii. The dashed line corresponds to the aforementioned resultant compression load, P, during max load conditions. It can be seen that a shell thickness of $1mm$ is only viable for cross-sectional radii larger than approximately $0.525m$. Conversely, a shell thickness of $4mm$ is only viable for cross-sectional radii greater than approximately $0.325m$.

Local Shell Buckling (LSB) LSB was approximated using buckling relations taken from NASA's buckling of thin-cylindrical shells specification, NASA SP-8007 [84]. For a sufficiently long, unstiffened isotropic cylinder subjected to a compressive axial load, the critical buckling stress can be approximated as:

$$\sigma_{cr} = \frac{\gamma_c E}{\sqrt{3(1-v^2)}} \frac{t}{r} \tag{4.17}$$

where E is the material young modulus, r is the radius of the shell, t is the shell thickness, and v is the material Poisson's ratio. Additionally, γ_c is an empirical knockdown factor for pure compression given by the formulation:

$$\gamma_c = 1 - 9.80(1 - e^{\frac{-1}{16}\sqrt{\frac{r}{t}}}) \tag{4.18}$$

Furthermore, the critical bending moment for cross section collapse of an isotropic, sufficiently long, unstiffened cylindrical shell is given by the following formula:

$$M_{cr} = \frac{\gamma_b E \pi r t^2}{\sqrt{3(1-v^2)}} \tag{4.19}$$

where γ_b is an empirical knockdown factor for pure bending given by the formulation:

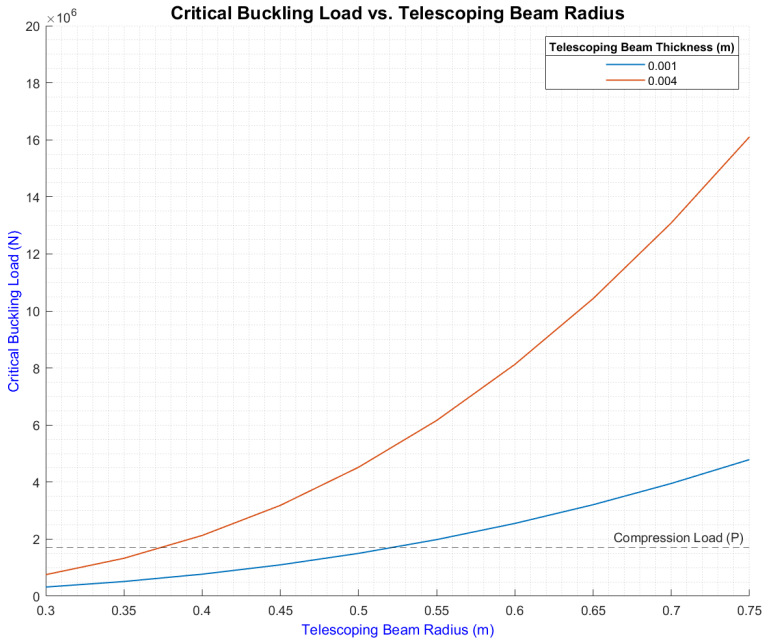


Figure 4.33: The critical buckling load for the telescoping support as a function of the telescoping beam radius for assumed shell thicknesses of $1mm$ and $4mm$. The dashed black line shows the max expected compression load on the support during max wind load conditions.

$$\gamma_b = 1 - 0.731(1 - e^{\frac{-1}{16}\sqrt{\frac{r}{t}}}) \tag{4.20}$$

In this application, however, the telescoping supports are subjected to both axial compression and bending loads. Per the aforementioned specification, in such instances LSB can be approximated through the simple interaction equation:

$$\phi_{LocalBuckling} = R_c + R_b = 1 \tag{4.21}$$

where R_c and R_b are the compressive stress and bending moment ratios given by the following equations

$$R_c = \frac{\sigma}{\sigma_{cr}} \tag{4.22}$$

$$R_b = \frac{M}{M_{cr}} \tag{4.23}$$

where σ and M are max compressive stresses and bending moments taken from the model. LSB occurs when the combination of these ratios exceeds 1.

Material Yield Material yield for this concept was evaluated through comparison of the maximum Von Mises Stress in the model versus the telescoping beam material yield strength as shown below:

$$\phi_{MaterialYield} = 100(\frac{\sigma_{max}}{\sigma_{YS}}) \tag{4.24}$$

Results

Table 4.5 shows how the mass performance of the telescoping support structure with respect to those configurations which meet the four previously described metrics, max transverse deflection, Euler buckling, local shell buckling, and material yield. The columns ”Volume”, ”Support Assembly Volume”, ”Total

Mass", "Max. Def.", "LSB", and "EB" refer to the volume of each individual telescoping support, the volume of all the supports combined, the total mass of one telescoping support, the maximum transverse deflection of the telescoping beam, local shell buckling factor, and Euler buckling factor, respectively. The reported radius column is the radius of the "root" telescoping section in which all the other smaller telescoping sections nest within. Note, 100 different telescoping beam configurations were evaluated but those options which violated any one of the four metrics were removed. Furthermore, the results are sorted from by minimum assembly mass. Configurations whose mass exceeded $250kg$ were also removed since the operating assumption here is that the telescoping support structure is allocated a total of $1000kg$ of the allowed $1500kg$ for the entire solar array assembly.Because this concept involves four telescoping supports, the max allowable weight per support is $250kg$. That said, however, the mass results reported here neglect the mass of the support arms and guy wires as well as the deployable support legs. Thus, the max allowable mass per telescoping support is likely less than $250kg$. Fortunately, there exists a sufficient spread of viable, lower weight options.

Table 4.5: Tabulated performance data for various telescoping support structural configurations assuming a $1000m^2$ array subjected to maximum wind load conditions.

Itr.	Radius (m)	Thickness (mm)	Volume (m^3)	Support Assembly Volume (m^3)	Total Mass (kg)	Max. Def. (m)	Max Von Mises Stress (Pa)	% of YS	LSB	Critical Buckling Load (N)	EB
15	0.50	1.33	1.35	5.42	137.32	0.017	4.37E+08	97.10	0.58	1.94E+06	0.88
16	0.55	1.33	1.64	6.55	151.27	0.015	3.98E+08	88.39	0.66	2.58E+06	0.66
17	0.60	1.33	1.95	7.80	165.23	0.014	3.65E+08	81.17	0.74	3.32E+06	0.52
25	0.50	1.67	1.35	5.42	170.95	0.014	3.54E+08	78.62	0.35	2.36E+06	0.73
18	0.65	1.33	2.29	9.16	179.18	0.013	3.38E+08	75.08	0.82	4.18E+06	0.41
34	0.45	2.00	1.10	4.39	183.37	0.012	3.37E+08	74.82	0.20	1.98E+06	0.86
26	0.55	1.67	1.64	6.55	188.39	0.012	3.21E+08	71.39	0.39	3.13E+06	0.55
19	0.70	1.33	2.65	10.62	193.14	0.012	3.14E+08	69.88	0.90	5.16E+06	0.33
35	0.50	2.00	1.35	5.42	204.30	0.011	3.00E+08	66.61	0.23	2.74E+06	0.62
27	0.60	1.67	1.95	7.80	205.83	0.011	2.95E+08	65.53	0.44	4.05E+06	0.42
20	0.75	1.33	3.05	12.19	207.09	0.011	2.94E+08	65.39	0.98	6.27E+06	0.27
44	0.45	2.33	1.10	4.39	212.95	0.011	2.95E+08	65.47	0.14	2.23E+06	0.77
28	0.65	1.67	2.29	9.16	223.28	0.010	2.73E+08	60.61	0.48	5.11E+06	0.33
36	0.55	2.00	1.64	6.55	225.23	0.010	2.70E+08	60.10	0.25	3.66E+06	0.47
45	0.50	2.33	1.35	5.42	237.37	0.010	2.62E+08	58.17	0.16	3.10E+06	0.55
63	0.40	3.00	0.87	3.47	239.88	0.009	2.75E+08	61.10	0.08	1.82E+06	0.94
29	0.70	1.67	2.65	10.62	240.72	0.010	2.54E+08	56.40	0.53	6.33E+06	0.27
54	0.45	2.67	1.10	4.39	242.26	0.009	2.63E+08	58.51	0.11	2.46E+06	0.69
37	0.60	2.00	1.95	7.80	246.16	0.009	2.48E+08	55.16	0.28	4.74E+06	0.36

Volume Estimation

The stowed volume of the canopy concept can be divided into two categories, the solar array membrane and the telescoping supports. The stowed volumes of different telescoping support configurations was already presented in Table 4.5. However, these values are influenced partially by the assumed number of telescoping sections in each support which in case of the aforementioned results was assumed to be 12. Figure 4.34 shows how this assumption influences the telescoping support stowed volume for various deployment lengths and cross-sectional radii. Note, the assumed shell thickness the presented results was $4mm$. Also, the telescoping sections are assumed to all be equal in length. The global decreasing trend in terms of volume with increased telescoping section count is a consequence of the fact that more telescoping sections reduces the stowed height of the telescoping support. While the stowed radius increases as a result, the magnitude of this change is overshadowed by the decrease in stowed height.For larger cross-sectional radii, the benefit of increasing the number of telescoping sections is more pronounced in comparison to smaller radii.

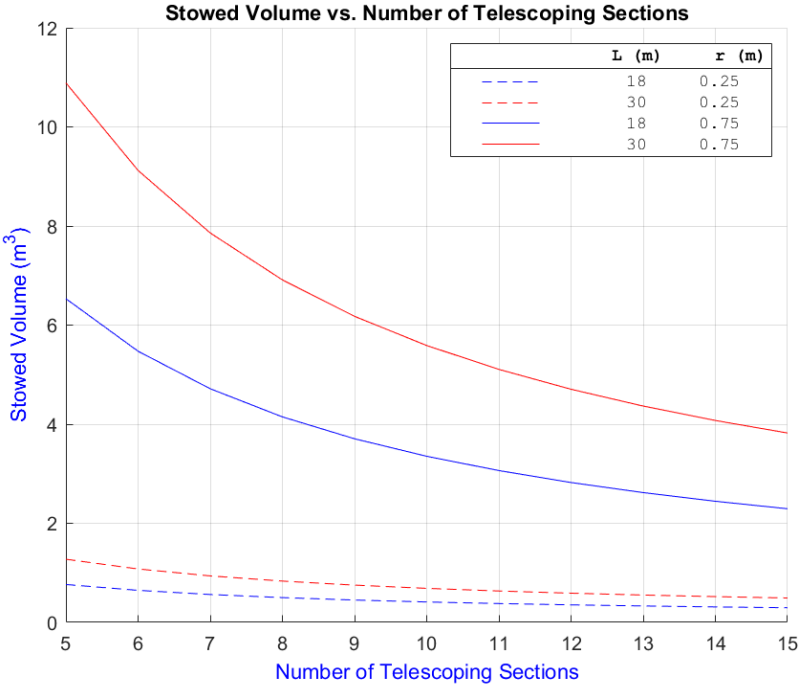


Figure 4.34: Telescoping support stowed volume as a function of number of telescoping sections for various assumed telescoping support deployment lengths and shell radii. Note the reported shell radii is the radius of the telescoping section at the root of the support. Furthermore, the assumed shell thickness was $4mm$.

With respect to the stowed volume of the solar array membrane, it was approximated using slip wrapping equations derived by Arya et al.[50] which are reproduced below. Essentially, the stowed volume is approximated by assuming every strip being wrapped is of equal length and follows the path of a central, composite curve (Equation 4.25) consisting of a semi-circle with a radius R, a line of length f, and an involute of a circle which has a pitch of $2\pi f$ as shown in Figure 4.36. Note, in Equation 4.25, s represents the arc length of the wrapped array. Since the array is symmetrically wrapped at its center, the bounds of v are between 0 and $L/2$, where L is the length of the array. In effect, the slip wrapped array is assumed to have a uniform thickness throughout. In reality, this is not the case since as the array is first z-folded, a non-uniform thickness is apparent resulting in a variable decreasing pitch when wrapping of the array (Figure 4.35). Nonetheless, the volumes computed herein are conservative in nature as a wrapping of a uniform thickness z-folded array will result in a larger stowed diameter.

$$\mathbf{p}(s) = \begin{cases} R\{1 - \cos(s/R), -\sin(s/R)\} & \text{if } s \in [0, \pi R] \\ R\{2, (s/R) - \pi\} & \text{if } s \in (\pi R, \pi R + f) \\ f\{\cos(\alpha - \theta) + \alpha \sin(\alpha - \theta), & \text{if } s \in (\pi R + f, L/2) \\ \sin(\alpha - \theta) - \alpha \cos(\alpha - \theta)\} & \end{cases} \tag{4.25}$$

$$\alpha^2 = \frac{2}{f}(s - \pi R - f) + \left(\frac{2R}{f}\right)^2$$

$$\theta = \frac{2R}{f} - \frac{\pi}{2}$$

The value of R in Equation 4.25 is derived from the equation:

$$R = R_{min} + \phi n h_m / 2 \tag{4.26}$$

where R_{min} is the minimum bend radius of the z-folded canopy,n is the number of strips, and h_m is the thickness of the solar array membrane. In this application, R_{min} was assumed to be the diameter

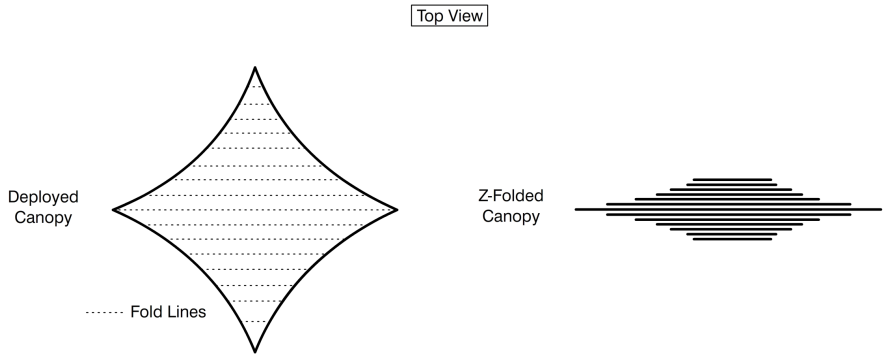


Figure 4.35: Here a top down view of a solar array canopy is shown that is z-folded along the dotted lines as shown. As observed, z-folding in this manner results in a folded canopy with variable thickness. For conservatism, the volume computations performed herein assume a constant thickness for this folded structure prior to wrapping.

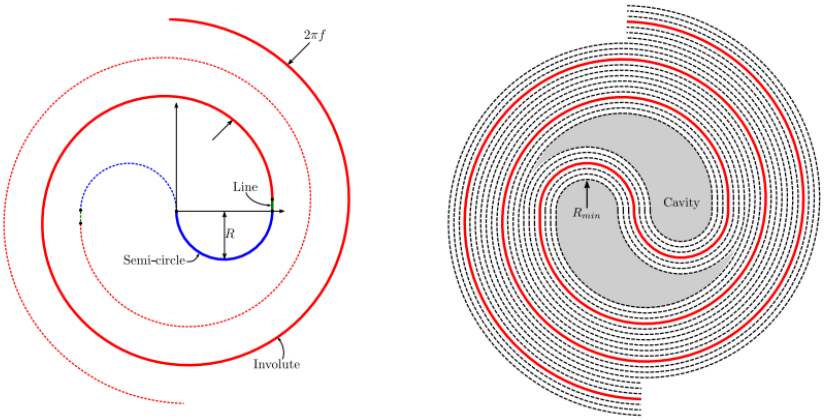


Figure 4.36: A graphical representation of the composite curve defined system of equations presented in Equation 4.25, left. This composite curve represents the path in with the z-folded canopy array follows when slip wrapped. The right image shows the slip wrapped, z-folded array assuming a constant z-folded stack thickness[50].

of the perimeter cable, $2r_c$, times a safety factor of 4. Furthermore, The pitch of the involute in this application is equivalent to:

$$2\pi f = 2\left(4r_c + n\phi h\right) \tag{4.27}$$

where, r_c is the radius of the perimeter cable, ϕ is a packaging inefficiency factor whose value is greater than 1, and h_m is the thickness of the solar array membrane. In effect, the pitch is twice thickness of the z-folded canopy.ϕ here was assumed to be 1.25 whereas h_m was assumed to be $1mm$. Figure 4.37 shows stowed volume versus the perimeter cable radius for various number of strips ,n. Increasing n has the benefit of reducing the stowed height of the array but as can be observed all computed stowage volumes regardless of fold count are excessively high. For example, if the same canopy dimensional and material parameters are used from earlier, the corresponding perimeter cable radius is approximately $0.015m$. Per Figure 4.37, the stowed volumes for z-fold counts of 20,40, and 60 are $20.11m^3$, $11.64m^3$, and $8.82m^3$, respectively.

Until this point, the operating assumption was that the canopy array could be z-folded, then slip wrapped, and stowed on top of one of the telescoping booms. As a reminder, the benefit of this configuration is that it allowed for the entirety of the solar array system be stowed around the circumference of the payload deck, leaving room for other payload in the center. Additionally, it was assumed that the overall stowage volume as well as form factor of this array would be reasonably low and complimentary in size/shape to the telescoping boom it rests upon, respectively. Per Figure 4.37, it is apparent that array stowage volumes are too high given the current assumptions of material choice for the membrane

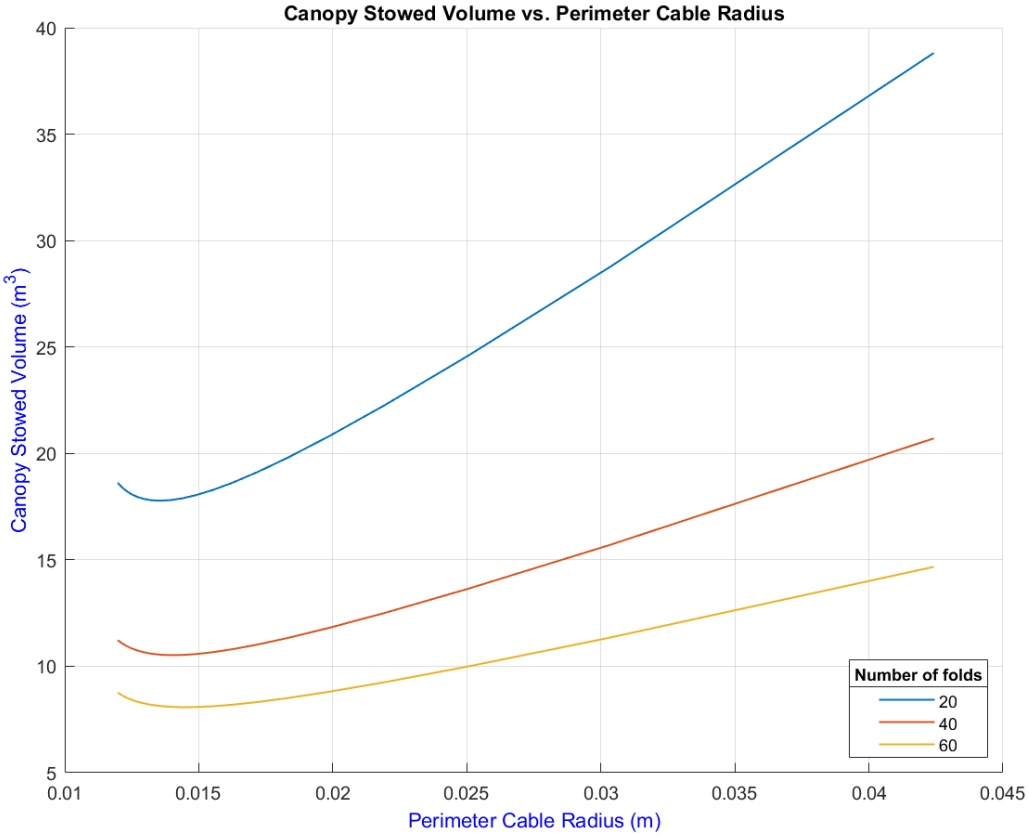


Figure 4.37: Volume of the slip wrapped array as a function of perimeter cable radius, assuming a total array area of $1000m^2$ and membrane thickness of $1mm$. This relation is shown for various number of assumed z-folds in the canopy prior to wrapping.

and perimeter cable as well as the assumed thickness of the membrane. The strain compatibility requirement between the perimeter cable and membrane results in an excessively thick perimeter cable for reasonable cable depths which presents issues in terms of how the canopy can be z-folded. Specifically, in order for the canopy to be compliant with the folding process the perimeter cable would need to bend at radii that may be too small to be feasible, as shown in Figure 4.38. Even so, if for the time being it is assumed that the cable can tolerate these bends, form factor of the stowed array is still a matter of concern. For example, if the total fold count of the canopy is taken to be 60 and the perimeter cable radius is set at $0.015m$ (this corresponds to the volume $8.82m^3$ reported earlier), the radius of the stowed array is approximately $1.88m$ with a stowed height of $0.81m$. This exceptionally large radius means that the stowed array would overhang the stowed telescoping beam it rests on. This is probably not advantageous as the array would protrude from the edge of the payload deck into the interior taking away space from additional payload.

With that said, there are still potential remedies to this situation. First and foremost, a more rigorous study could be conducted to find a more suitable material combination in terms of the perimeter cable and membrane that allows for a decrease in perimeter cable radius. Additionally, the canopy design could be amended to a so-called “Web-Cable Girded Design” which has been shown to reduce the perimeter cable thickness [85].

Risk Estimation

Requirement Risk Given the nature of the canopy Design, lander clearance angle and array segmentation are both irrelevant parameters. However, the purpose of evaluating for the lander clearance

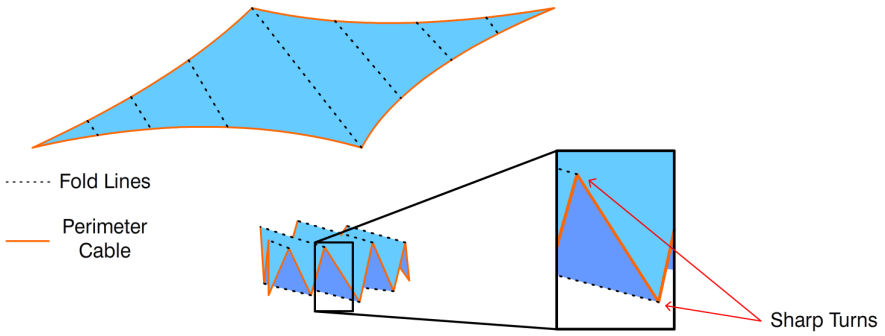


Figure 4.38: Z-folding of the canopy array presents compatibility issues for the perimeter cable in the canopy. As shown, the perimeter cable is required to make sharp turns when the array is folded which may or may not be feasible without damaging the cable. This is especially true for thick cables.

angle in particular is to gain a preliminary assessment of a concept's performance in response to varying levels of required access to the lander. Because the array deploys over the lander in this concept, the question of restriction of access to the additional cargo onboard would only be relevant should crane systems be the primary method of offloading equipment.

With respect to scalability, the evaluation methodology used earlier was repeated for a solar array membrane area of $1500m^2$, the results of which are presented in Table 4.6. Furthermore, a mesh reinforcement areal mass of $21.97\frac{g}{m^2}$, a membrane thickness of $1mm$,and an array deployment height of $4m$ were similarly assumed in this analysis. With this assumed deployment height, a max membrane out-of-plane deflection was also set to $1m$ which, under the same wind load conditions as described earlier, results in a reaction load at each canopy tip of approximately $644kN$. For the purposes of a better comparison, a number of adjustments were made to the telescoping beam design as well as solar array membrane to derive these results. Starting with the membrane, the canopy depth ,h, was increased from $2.5m$ to $5m$ in order to maintain a similar sized perimeter cable radius, r_c, between the $1000m^2$ and $1500m^2$ arrays. After all, it was shown that perimeter cable radius greatly influences the stowage properties of the solar array membrane. With respect to the telescoping beam, the telescoping section count was increased from 12 to 15 to reduce the larger array's telescoping beam support stowage volume. Moreover, the assumed location of the support legs was adjusted so that it remained at the center of the deployed length. The increase in array area in combination with the deeper canopy depth means the $1500m^2$ array's telescoping support deployment length increases from approximately $20.7m$ to $28.3m$ (a 36% increase). The increase in deployment length of the telescoping supports increases the individual telescoping support mass above $250kg$ for all support cross-section combinations tested. Furthermore, the only combinations that were compliant to the max deflection, material yield, and buckling criteria laid out earlier are mostly larger in radius and thickness than those for the $1000m^2$ array which also contributes to this increase in mass. Larger cross-sectional radii is also a factor in driving up assembly stowage volume where ,as shown in Table 4.6, most configurations exceed $10m^3$. Those which are below this requirement have Euler buckling factors which are rather close to failure meaning these options are likely not viable.

Mission Operation Risk From an operational risk perspective, lack of redundancy in the concept is a clear disadvantage. Specifically, there are several steps in the deployment process where if they are not performed successfully, the entire assembly fails. Should any of the telescoping booms fail to deploy or achieve the required deployment angle, for example, the array would not be deployed properly. Furthermore, should crane access to the payload be required, the Canopy array must be able to retract and stow. For this concept, it is not readily apparent how this could be achieved given there are many technical challenges that would need to be addressed like clearing the array of dust prior to rolling, how to efficiently re-z-fold the array, moving cables that span across the lander between the telescoping members when stowed, etc. Because the concept will likely deploy several meters above

Table 4.6: Tabulated performance data for various telescoping support structural configurations assuming a $1500m^2$ array subjected to maximum wind load conditions.

Itr.	Radius (m)	Thickness (mm)	Volume (m^3)	Support Assembly Volume (m^3)	Total Mass (kg)	Max. Def. (m)	Max Von Mises Stress (Pa)	% of YS	LSB	Critical Buckling Load (N)	EB
13	0.65	1.56	2.50	10.01	284.15	0.018	4.41E+08	98.06	0.46	2.90E+06	0.90
14	0.70	1.56	2.90	11.61	306.41	0.017	4.09E+08	90.94	0.51	3.61E+06	0.72
15	0.75	1.56	3.33	13.33	328.67	0.016	3.82E+08	84.93	0.56	4.43E+06	0.59
16	0.80	1.56	3.79	15.17	350.93	0.015	3.59E+08	79.68	0.61	5.36E+06	0.49
22	0.60	2.11	2.13	8.53	352.91	0.014	3.66E+08	81.33	0.25	2.92E+06	0.89
17	0.85	1.56	4.28	17.12	373.18	0.014	3.38E+08	75.08	0.66	6.39E+06	0.41
23	0.65	2.11	2.50	10.01	383.12	0.013	3.35E+08	74.46	0.28	3.74E+06	0.70
18	0.90	1.56	4.80	19.20	395.44	0.013	3.19E+08	70.99	0.71	7.53E+06	0.35
31	0.55	2.67	1.79	7.17	404.45	0.013	3.31E+08	73.58	0.16	2.65E+06	0.99
24	0.70	2.11	2.90	11.61	413.33	0.012	3.09E+08	68.67	0.31	4.69E+06	0.56
19	0.95	1.56	5.35	21.39	417.70	0.012	3.03E+08	67.35	0.77	8.78E+06	0.30
20	1.00	1.56	5.92	23.70	439.96	0.012	2.88E+08	64.08	0.82	1.01E+07	0.26
32	0.60	2.67	2.13	8.53	442.61	0.011	3.00E+08	66.61	0.18	3.49E+06	0.75
25	0.75	2.11	3.33	13.33	443.53	0.012	2.87E+08	63.87	0.34	5.77E+06	0.45
26	0.80	2.11	3.79	15.17	473.74	0.011	2.70E+08	59.92	0.37	6.99E+06	0.37
33	0.65	2.67	2.50	10.01	480.76	0.011	2.74E+08	60.85	0.20	4.49E+06	0.58
41	0.55	3.22	1.79	7.17	484.87	0.010	2.85E+08	63.29	0.13	3.00E+06	0.87
27	0.85	2.11	4.28	17.12	503.95	0.010	2.54E+08	56.45	0.40	8.36E+06	0.31
34	0.70	2.67	2.90	11.61	518.92	0.010	2.52E+08	56.02	0.22	5.65E+06	0.46

the ground, a consequence of the lander deck and cargo profile heights, repairability is also likely to be a risk just by proxy of difficulty of access.

Design and Integration Risk In terms of Manufacturability, the telescopic beams may pose a manufacturing risk given the very small thickness to width ratio of the shell sections which is a consequence of the desire to keep support mass below $250kg$. Adherence of the segments of solar arrays to the canopy membrane may pose a logistical challenge during assembly since the canopy is one contiguous mesh-cable system.

Testing and Qualification Risk While this concept utilizes a single canopy configuration, the solar cell component is divided into several smaller sections which are assembled separately on the canopy. This reduces its testing and qualification risk since manufacturing can be performed in tandem with testing and assembly. The large size of the canopy, however, would likely require a very large clean room for putting together the entire assembly and testing it.

4.3.5. Stripped Array

Mass Estimation

Max Deflection Similar to the Canopy concept, analysis of deflections for the Stripped Array were divided into two separate analyses: one for the solar array membrane sub-assembly and one for the telescoping beam supporting architecture.With regard to the solar array membrane, a series of non-linear static finite element analysis were performed in Abaqus involving different levels of pretension apparent in the structure as well as varying degrees of mesh reinforcement areal mass. The entirety of each model was represented by B33 elements, as shown in Figure 4.39, where the mesh reinforcement in each strip was represented by a single cable whose radius is a function of the assumed reinforcement areal mass. For the purposes of comparison, the mesh reinforcement areal masses used earlier for the Canopy concept , namely $21.97\frac{g}{m^2}$,$43.95\frac{g}{m^2}$ and $87.90\frac{g}{m^2}$, were used here as well. In terms of model setup, the center of the model is pinned. Pretension in the model is introduced in a separate load step through prescribed displacements at each of four corners in the model. Following this displacement, the corners are assumed to be pinned in place. In a final load step, an out-of-plane line-load is then applied to every strip to simulate the maximum wind load condition. The magnitude of this load is computed using the same equations described in the previous section but resolved into a Newton per meter load computed from the strip area. For this preliminary analysis, the diagonal cable radius, r_{dc}, was assumed to be $7.5mm$ where as the total number of solar array strips in each of the four quadrants, n, was assumed to be 22. The motivation behind these assumptions is based on array packaging considerations explained later in the next section.

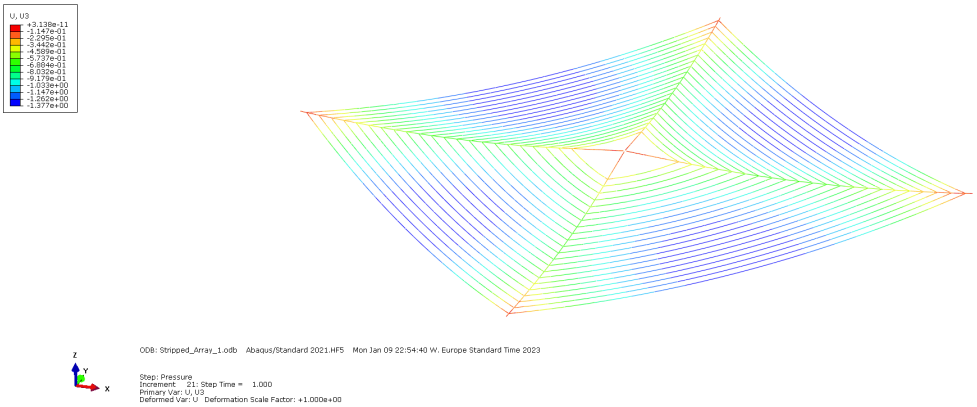


Figure 4.39: Stripped Array static general membrane Abaqus model where the solar array strips are idealized as cables.

The results of the aforementioned analyses are shown in Figure 4.40. By comparison to the Canopy concept, much lower pre-tension loads are required to keep out-of-plane deflections low. This is in part due to the inclusion of a support at the center of the array. If a maximum deflection of $1m$ is kept as before, the approximate required pre-tension loads for each of the mesh reinforcement areal masses tested ($21.97\frac{g}{m^2}$,$43.95\frac{g}{m^2}$ and $87.90\frac{g}{m^2}$) are $112kN$, $47kN$, and $26kN$, respectively. Additionally, if the $21.97\frac{g}{m^2}$ mesh areal mass is selected and the $1m$ max. deflection is maintained, the approximate reaction load at each of the four corners of the stripped array following the application of wind pressure is $140kN$. This load is converted later to a resultant compression load, P, that is used later to evaluate the feasibility of various configurations of the telescoping supporting architecture.

All Other Criteria Since the telescoping support architecture is largely the same as that which is used in the Canopy concept, the same methodology and structural model was used to evaluate the support deflections. Furthermore, the methodology and assumptions utilized earlier to evaluate for EB, LSB, and material yield was repeated here too. Of course, a slight adjustment was made to code to accommodate the difference in deployed length for a $1000m^2$. Specifically, the length of each the telescoping support is dependant on how closely the strips are deployed to the center of the array, as shown in Figure 4.41. When the innermost strip length, L_i, is increased, it has the effect of increasing

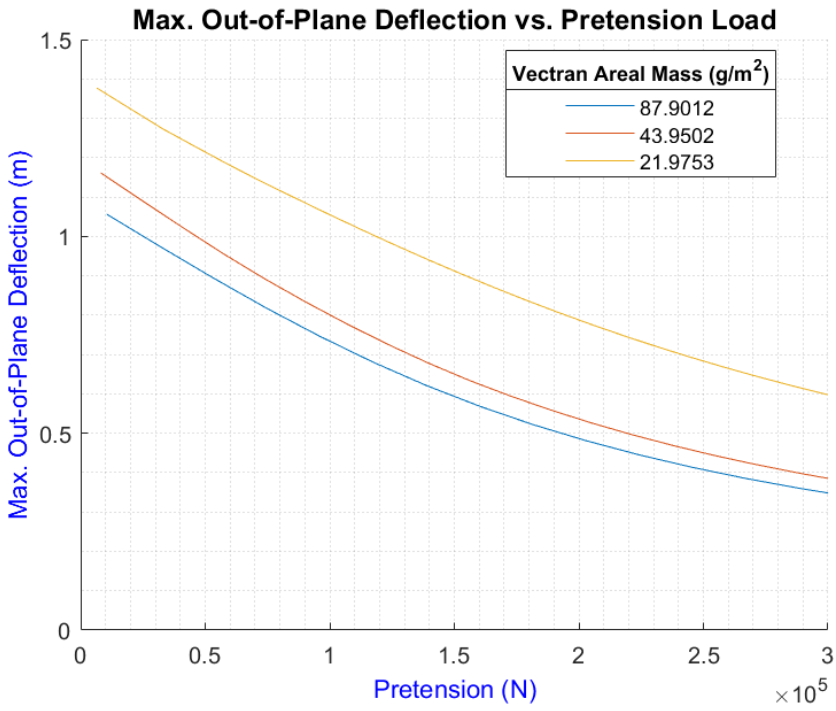


Figure 4.40: Stripped array membrane deflections as a function of the pre-tension load applied to each of the four corners of the array for various assumed mesh reinforcement areal masses.

the distance between each of the four array corners in order to maintain the same total array area. Since the telescoping supports connect at these corners, the consequence here is that the total deployment length of the support increases as well.For clarity, each solar array strip is trapezoidal in shape and the ”inner most strip length” refers to the shorter edge of the inner most strip. For this study, L_i was assumed to be a constant value of $4.5m$. For reference, this corresponds to a total support deployment length of $18.47m$ assuming a $1000m^2$, a lander diameter of $9.1m$, and an array deployment height of $4m$.

Because the telescoping supports in this concept deploy to a shorter length than in the Canopy design, the structure performs better with respect to column buckling as shown in Figure 4.42. Here the red and blue dashed lines represent the telescoping support critical buckling loads in the Canopy design for shell thicknesses of $1mm$ and $4mm$, respectively, should the aforementioned canopy assumptions be maintained. Conversely, the solid red and blue lines are the critical buckling loads for the supports in the Stripped Array concept for shell thicknesses of $1mm$ and $4mm$, respectively. Again, the black dashed line here is the resultant compression load, P, on the support during the assumed max wind load conditions.As observed in Figure 4.42, telescoping beam structures with $1mm$ thick shells tolerate column buckling at cross-sectional radii larger than $0.35m$ whereas every shell radii with a thickness of $4mm$ endures.

Results

Table 4.7 shows the mass performance of the telescoping support structure where structural configurations which have excessive deflection, buckles or yields are filtered out. Again, the columns ”Volume”, ”Support Assembly Volume”, ”Total Mass”, ”Max. Def.”, ”LSB”, and ”EB” refer to the volume of each individual telescoping support, the volume of all the supports combined, the total mass of one telescoping support, the maximum transverse deflection of the telescoping beam, local shell buckling factor, and Euler buckling factor, respectively. Furthermore, while 100 different configurations were tested, only the first few are reported here and are sorted by mass performance. Since this concept utilizes four telescoping supports, the max allowable mass per support is $250kg$ as in the Canopy concept. The reported masses herein neglect the additional mass of the stabilizing guy wires, tensioning cables, and additional support architecture like the short, deployable arms and deployable support legs. For the

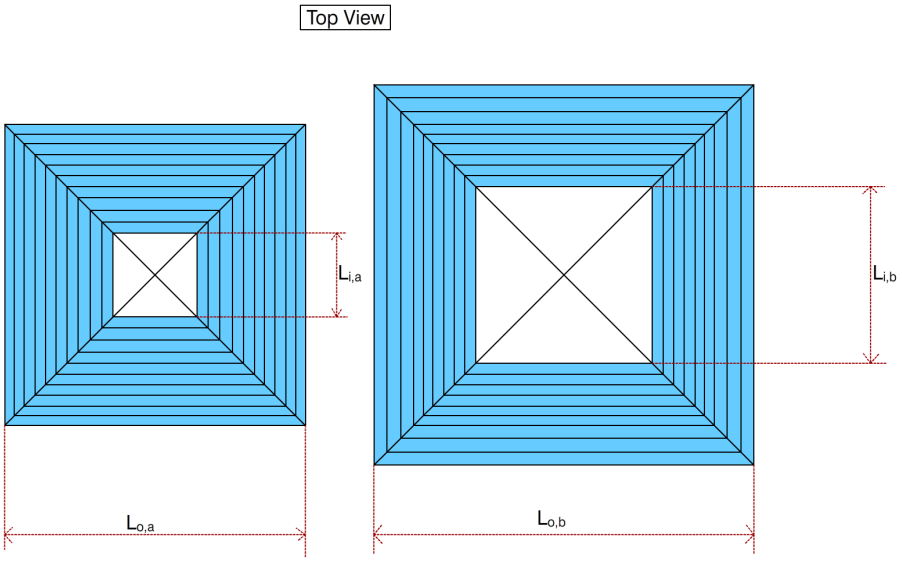


Figure 4.41: This simple geometric figure shows a simplified, top down view of the stripped array and the effect of innermost strip length on the entire array's dimensions. Specifically, if the total array area is kept the same, the exterior length of the array must increase in response to an increase in the innermost strip length.

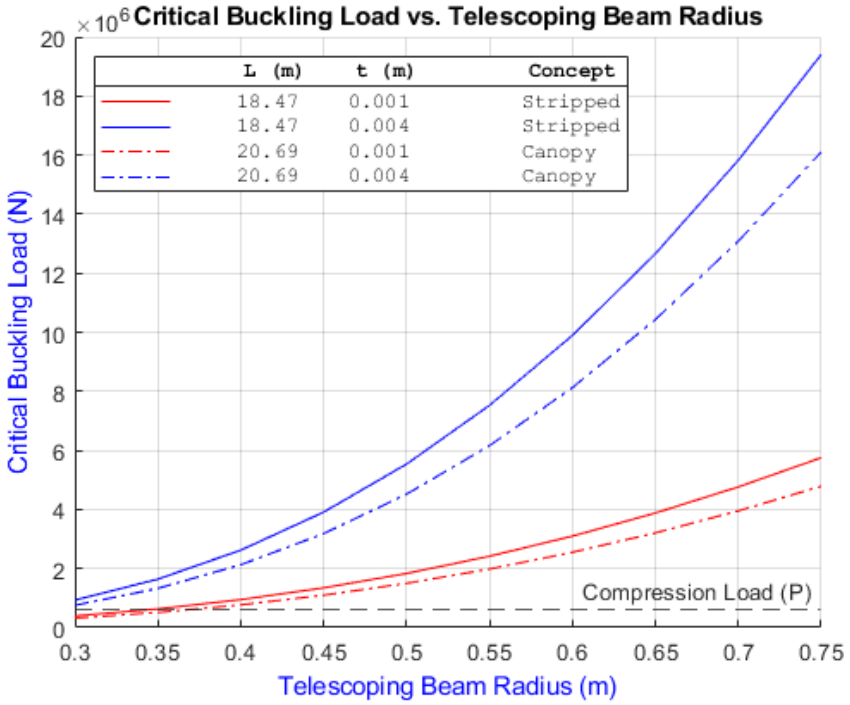


Figure 4.42: This plot shows the derived critical buckling load as a function of the telescoping beam radius for the telescoping supports for different combinations of deployed length and shell thickness, assuming a total array area of $1000m^2$ and deployment height of $4m$. The solid lines show the critical buckling loads of the stripped array's telescoping supports whereas the dashed lines refer to the buckling characteristics of the canopy concept presented earlier in the Figure 4.33. These were included for comparison.

time being, the mass impact of these elements is assumed to be low and thus neglected.

Table 4.7: Tabulated performance data for various telescoping support structural configurations in the stripped array concept assuming a $1000m^2$ array subjected to maximum wind load conditions.

Itr.	Radius (m)	Thickness (mm)	Volume (m^3)	Support Assembly Volume (m^3)	Total Mass (kg)	Max. Def. (m)	Max Von Mises Stress (Pa)	% of YS	LSB	Critical Buckling Load (N)	EB
3	0.40	1.00	0.77	3.10	73.63	0.010	2.53E+08	56.33	0.33	9.48E+05	0.63
12	0.35	1.22	0.59	2.37	78.27	0.009	2.41E+08	53.58	0.19	7.52E+05	0.79
4	0.45	1.00	0.98	3.92	82.98	0.009	2.26E+08	50.17	0.37	1.35E+06	0.44
13	0.40	1.22	0.77	3.10	89.69	0.008	2.09E+08	46.48	0.22	1.13E+06	0.53
22	0.35	1.44	0.59	2.37	92.14	0.008	2.07E+08	46.05	0.13	8.62E+05	0.69
5	0.50	1.00	1.21	4.84	92.32	0.008	2.04E+08	45.27	0.41	1.83E+06	0.33
14	0.45	1.22	0.98	3.92	101.11	0.007	1.86E+08	41.38	0.24	1.61E+06	0.37
6	0.55	1.00	1.46	5.85	101.66	0.007	1.86E+08	41.28	0.45	2.42E+06	0.25
23	0.40	1.44	0.77	3.10	105.64	0.007	1.79E+08	39.78	0.15	1.30E+06	0.46
32	0.35	1.67	0.59	2.37	105.90	0.007	1.83E+08	40.61	0.10	9.66E+05	0.62
7	0.60	1.00	1.74	6.96	111.01	0.007	1.71E+08	37.96	0.49	3.10E+06	0.19
15	0.50	1.22	1.21	4.84	112.53	0.006	1.68E+08	37.33	0.27	2.20E+06	0.27
51	0.30	2.11	0.44	1.74	113.36	0.006	1.79E+08	39.79	0.06	6.96E+05	0.86
24	0.45	1.44	0.98	3.92	119.13	0.006	1.59E+08	35.32	0.17	1.86E+06	0.32
42	0.35	1.89	0.59	2.37	119.55	0.006	1.64E+08	36.48	0.08	1.06E+06	0.56
8	0.65	1.00	2.04	8.17	120.35	0.006	1.58E+08	35.16	0.53	3.88E+06	0.15
33	0.40	1.67	0.77	3.10	121.47	0.006	1.58E+08	35.02	0.11	1.46E+06	0.41
16	0.55	1.22	1.46	5.85	123.95	0.006	1.53E+08	34.03	0.30	2.90E+06	0.21
61	0.30	2.33	0.44	1.74	124.71	0.006	1.66E+08	36.80	0.05	7.41E+05	0.81

Volume Estimation

The volume of the membrane component of this concept was approximated using the general method as described in the previous section but adapted for a membrane in which is ”quad” slip wrapped [79]. Essentially, the Stripped Array concept collapses into a cross formation where each ”arm” is wrapped around each other as shown in Figure 4.43. Again, each arm is assumed to have constant thickness with strips of equal length that collectively follow the path a composite curve containing an involute of a circle. The thickness of each arm can be approximated as $2(nh\phi + r_{dc} + \gamma)$. Here n is the number of solar array strips, h_m is the thickness of each strip, ϕ is a stowage inefficiency factor, r_{dc} is the radius of the diagonal cable, and γ is the thickness of the strip spreader bars. A visual representation of this is shown in Figure 4.44 As before, ϕ was assumed to be 1.25 whereas γ was assumed to be $10mm$. For this concept, the bend radius of each z-folded arm, R, is governed by the equation $R = R_{min} + (n\phi h_m) + \gamma$, where R_{min} was assumed to be the diameter of the diagonal cable times a safety factor of 4.

That said, for a membrane with an assumed total area of $1000m^2$ and a diagonal cable radius of $7.5mm$, the effect of number of solar array strips and strip thickness on the array stowed volume is shown in a contour plot in Figure 4.45. Note these computed volumes do not include the volume of the additional housing and deployment mechanisms that accompany the array in its stowed form. In general, stowed array volumes are exceptionally high at low strip counts due to the fact that decreasing the amount of strips effectively increases the standing height the stowed array. As an example, for an assumed membrane thickness of $1mm$ and strip count of 6, the stowed volume of the array is approximately $7.51m^3$. On the other hand, for the same membrane thickness, stowed volume of the array for strip counts of 20,40, and 60 are $3.79m^3$, $2.99m^3$, and $2.71m^3$, respectively. The maximum stowed height corresponds to the width of a single strip and decays at a rate of 0.5^{x-1} where x is number of strips. The width of these strips cannot be infinitely small as the minimum sizing is dependent on system level considerations such as the size and configuration of the solar cells which populate each strip. Thus, there is certainly an upper limit to the amount of strips that are feasible. Stowage form factor was also a consideration where too high of a strip count would lead to an excessively wide stowed array. Conversely, too low of a count would result in an extremely tall packaging configuration and a large

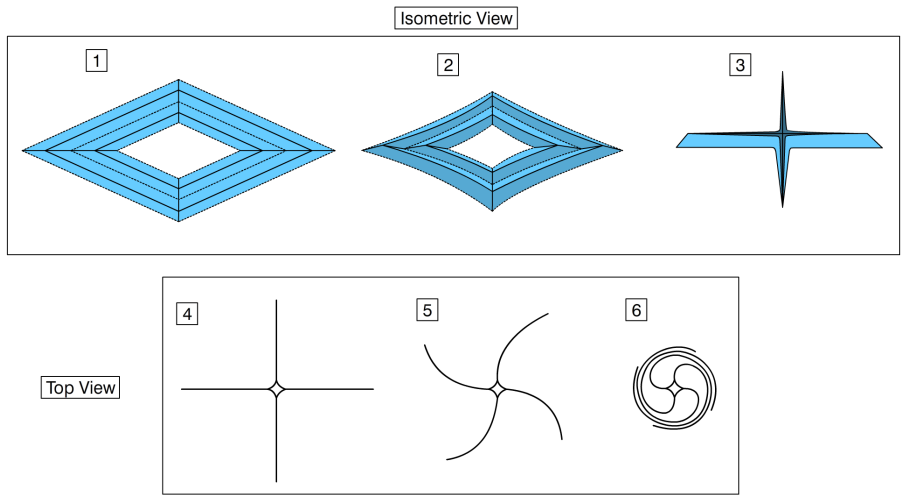


Figure 4.43: The stowage method for the stripped array. The strips in each quadrant collapse into the configuration shown in step 3 where thereafter the entire array is quad slip wrapped as shown.

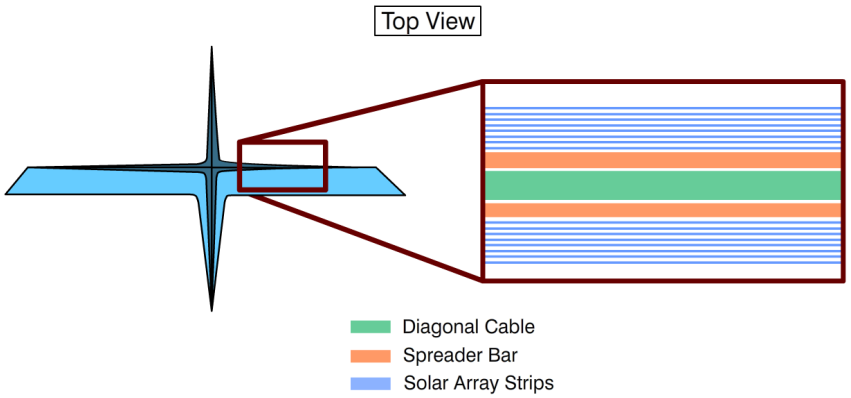


Figure 4.44: The collapsed array shown in step 3 from Figure 4.43 is reproduced here. Essentially, each ”arm” of the collapsed state is assumed to have a constant thickness for the purposes of approximating the stripped array’s stowed volume. The diagonal cable occupies the center of each arm and is flanked by solar array strips on either side. Each of these strips have spreaders bars which sandwich between the strips and the aforementioned diagonal cable as shown.

volume.

Assumed strip thickness is also clearly a driving factor for the array stowed volume. At thicknesses on the order of magnitude of a few millimeters, the stowed volume of the array is already very large. For an assumed strip count of 20, the stowed volume for strips with a mesh reinforcement thickness of $1mm$, $2mm$,and $3mm$ are $3.79m^3$, $5.75m^3$, and $8.14m^3$, respectively. Clearly there is a need to keep this thickness as low as possible where around $1mm$ to $1.5mm$ or less is probably the most viable here. After all, the array stowed volume must be as low as possible since a larger portion of the stowed volume requirement ($10m^3$) will likely be taken up by the telescoping supports. The load that each strip has to carry partially controls the strip thickness, however, but so does manufacturability. Manufacturing capabilities were thereby briefly assessed in determining feasible strip reinforcement mesh thicknesses. Kururay, the primary producer of the high-strength polyarylate fiber Vectran, offers fiber titers that range from 110-1580 Decitext [86]. The equivalent fiber radii are approximately $0.05mm$ to $0.19mm$, respectively. Consequently, selection of a strip thickness on the order of magnitude of a millimeter or two is likely achievable.

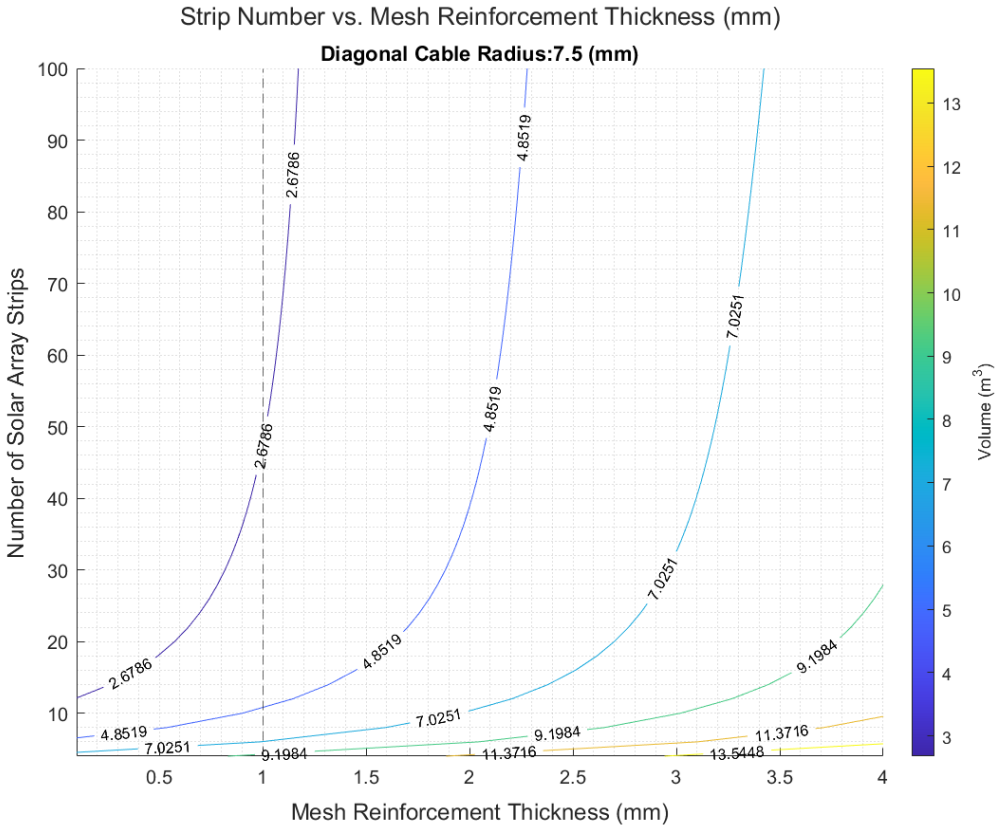


Figure 4.45: Solar array stowed volume as a function of strip count and mesh reinforcement thickness.

Risk Estimation

Requirement Risk As with the Canopy concept, lander clearance angle and array segmentation are neither relevant for this concept since it involves a singular membrane that also deploys above the lander. In terms of scalability, the mass and volume performance of a $1500m^2$ array was evaluated utilizing the same methodology described above for the smaller, $1000m^2$ array.As a consequence of increasing the array area to $1500m^2$,the deployment length of the telescoping supports increases from 18.47 to $23.37m$. For comparison, a mesh reinforcement areal mass of $21.97\frac{g}{m^2}$ was assumed along with a mesh reinforcement membrane thickness of $1mm$. Additionally, the interior strip length, L_i, was kept at $4.5m$. The total strip count, n, was increased from 22 to 28 to maintain a similar strip width between the two array sizes. Also, the radius of the diagonal cable, r_{dc}, was assumed to be $7.5mm$. Lastly, an array deployment height of $4m$ were similarly assumed in this analysis. With that said, a quick membrane deflection analysis was repeated herein and if a maximum out-of-plane displacement of the array is kept at $1m$, the array requires a pretension load of $253kN$ applied at each corner. Following application of the maximum wind load condition explained earlier, the reaction load at the tip of the array is $272kN$. As with the Canopy scalability study, the same adjustments were made to the telescoping support architecture with regard to increasing the telescoping section count from 12 to 15 and adjusting the position of the deployable support legs to the center of the deployed telescoping beam. That said, Table 4.8 shows truncated results from this mini-scalability study where configurations which violated any of the buckling, yield, or deflection criteria were filtered out. Moreover, the table is sorted by mass performance. There are more higher mass configurations which meet requirements but these were left out to avoid reporting an excessively long table. Regardless of the increase in total array area, there are still several configurations which adhere to the aforementioned structural criteria while maintaining reasonably low stowage volumes and mass. While it is not unreasonable to assume that mass and volume requirements for the entire deployable solar array system, $1000kg$ and $10m^3$ respectively, would be relaxed should the array area requirement increase, most configurations shown in Table 4.8 show promise in respecting these original requirements. For example, ”Iteration 16” with a radius of $0.55m$

has a total telescoping support mass and volume of approximately $624kg$ ($156kg$ for each of the four telescoping supports) and $5.92m^3$. A Margin of $376kg$ as well as approximately $4m^3$ of volume is thus allotted for the stowed array, support arms, and cable support systems which is reasonable.

Table 4.8: Tabulated performance data for various telescoping support structural configurations in the stripped array concept assuming a $1500m^2$ array subjected to maximum wind load conditions.

Itr.	Radius (m)	Thickness (mm)	Volume (m^3)	Support Assembly Volume (m^3)	Total Mass (kg)	Max. Def. (m)	Max Von Mises Stress (Pa)	% of YS	LSB	Critical Buckling Load (N)	EB
5	0.50	1.00	1.22	4.89	116.41	0.016	3.81E+08	84.66	0.43	1.28E+06	0.88
6	0.55	1.00	1.48	5.92	128.23	0.015	3.47E+08	77.02	0.48	1.70E+06	0.66
7	0.60	1.00	1.76	7.05	140.04	0.013	3.18E+08	70.68	0.52	2.20E+06	0.51
15	0.50	1.22	1.22	4.89	141.80	0.013	3.15E+08	69.95	0.28	1.53E+06	0.74
24	0.45	1.44	0.99	3.96	149.94	0.012	3.03E+08	67.37	0.19	1.27E+06	0.88
8	0.65	1.00	2.07	8.27	151.86	0.012	2.94E+08	65.34	0.57	2.78E+06	0.40
16	0.55	1.22	1.48	5.92	156.24	0.012	2.86E+08	63.55	0.31	2.03E+06	0.55
9	0.70	1.00	2.40	9.59	163.68	0.011	2.74E+08	60.78	0.61	3.44E+06	0.33
25	0.50	1.44	1.22	4.89	167.01	0.011	2.70E+08	60.03	0.22	1.76E+06	0.64
17	0.60	1.22	1.76	7.05	170.68	0.011	2.62E+08	58.31	0.35	2.63E+06	0.43
34	0.45	1.67	0.99	3.96	172.35	0.011	2.67E+08	59.38	0.16	1.42E+06	0.79
10	0.75	1.00	2.75	11.01	175.50	0.011	2.56E+08	56.84	0.66	4.18E+06	0.27
26	0.55	1.44	1.48	5.92	184.08	0.010	2.44E+08	54.27	0.24	2.35E+06	0.48
18	0.65	1.22	2.07	8.27	185.13	0.010	2.43E+08	53.89	0.38	3.33E+06	0.34
53	0.40	2.11	0.78	3.13	191.69	0.009	2.50E+08	55.67	0.10	1.16E+06	0.97
35	0.50	1.67	1.22	4.89	192.04	0.010	2.38E+08	52.83	0.18	1.98E+06	0.57
44	0.45	1.89	0.99	3.96	194.58	0.009	2.40E+08	53.31	0.13	1.57E+06	0.72
19	0.70	1.22	2.40	9.59	199.57	0.009	2.26E+08	50.13	0.41	4.13E+06	0.27
27	0.60	1.44	1.76	7.05	201.15	0.009	2.24E+08	49.78	0.27	3.05E+06	0.37

Mission Operation Risk With regard to operational risk, the Stripped Array concept shares many of the risks associated with the Canopy concept given the fact that the telescoping supporting architecture is essentially the same as well as many of the deployment characteristics of the membrane like its deployment height and location over the payload. Again, one of the biggest risks is that the system is not single fault tolerant. That is, should one of the four telescoping supports fail to deploy for any reason, the solar array power system would fail. While discretization of the solar array into strips likely is beneficial from a repairability standpoint, the deployment height of several meters above the ground still complicates the process. Similar to the Canopy concept, retract-ability of the design is also not straightforward. Refolding the array properly prior to re-wrapping in particular poses a unique challenge where no easily integratable solution is readily apparent. A considerable risk of this design is the fact that it relies on the assumption that the stowed array at the center of the lander can either be supported by additional payload underneath or there is adequate space on the lander deck for a support column.

Design and Integration Risk A benefit of the Stripped Array concept is its manufacturability. Discretization of the solar array into strips allows for separation of the manufacturing process in to more manageable, smaller steps as opposed to the Canopy concept. Furthermore, preliminary assessment showed that the assumed mesh reinforcement thickness as well as diagonal cable radius is not only viable from a structural standpoint but also is within the realm of existing manufacturing capabilities. In terms of TRL, the telescoping support beam technology has a low readiness level given the fact the technology has not undergone any physical laboratory testing and nor has it been utilized in previous planetary surface missions. With regard to the stripped solar array, this technology also possesses a

low TRL score. However, similar stowage systems have flown on past solar sail missions, namely the IKAROS sail developed by the Japan Aerospace Exploration Agency (JAXA) [87].

Testing and Qualification Risk Discretization of the array into strips allows for manufacturing, testing/qualification, and assembly of the solar array to be done in tandem which in turn reduces the total manufacturing lead time. Moreover,a decrease in overall deployment length of the telescoping support beams compared to the Canopy concept as well as the apparent mass margin shown earlier likely means the thickness of these supports could be increased without violating the mass budget. Consequently, inspection requirements could likely be relaxed for these components as the role of imperfections in the composite shell may be less influential.

4.3.6. AHP Comparison

In light of the above detailed analyses, it is apparent that the Stripped Array concept was likely the ideal candidate on the basis of mass and volume performance. Nonetheless, the AHP method was still implemented for procedural purposes. First, the down selection criteria were ranked against each other, qualitatively, using pairwise comparisons per the method described in Appendix A. The resultant weights from this process are shown in Table 4.9. Risk was attributed the lowest ranking on the basis that it is difficult to properly quantify so early in the design process. Because the risk criteria is further broken down into sub-criteria, weights were also derived for each. With the criteria weights articulated, pairwise comparisons were made between each concept for each criteria and when applicable sub-criteria using the knowledge acquired from the above analyses. The final scoring of each concept is shown in Table 4.10. Note, each column in this table corresponds to a pairwise comparison between every concept for that particular criteria or sub-criteria. Each row is the performance value of a concept per each aforementioned comparison. The total concept performance is just the cumulative sum of each row. As one can see, the clear winner here is the Stripped Array concept.

Table 4.9: Derived criteria and sub-criteria weights using the AHP method.

Criteria	Weights
Mass	0.525
Volume	0.334
Risk	0.142

Risk Sub-criteria	Weights
Requirements	0.0543
Mission Operation	0.0543
Design and Integration	0.0256
Testing of Components	0.0074

Table 4.10: Concept performance derived using the AHP method described in Appendix A

Concept	Mass	Volume	Risk				Performance
			Req.	Mission Ops	Design	Testing	
RCTSA	0.046	0.071	0.005	0.039	0.014	0.005	0.1790
Canopy	0.144	0.028	0.012	0.010	0.005	0.001	0.2001
Stripped Array	0.335	0.234	0.038	0.005	0.007	0.001	0.6209

The justification behind these scorings is briefly summarized below. These justifications are not the only reasons for each decision per say but a few of the main points are highlighted. In terms of mass performance, the Stripped Array received the best score by comparison for a number of reasons. First and foremost, it was shown earlier that the RCTSA telescoping support lacked sufficient means to combat local buckling for configurations with masses below or near the $1000kg$ target. While the Canopy design shares many similarities to that of the Stripped Array, the Stripped Array's better mass performance by comparison stems from a few fundamental features of the design. Namely, the stripped membrane configuration as well as supporting the membrane at five points allows for a decrease in telescoping support deployment length as well as a reduces the incident loading on said supports. This length decrease and load alleviation means less mass is required in the telescoping support thus driving assembly mass down. For a $1000m^2$ array coupled with an assumed lander diameter of $9.1m$,

interior strip length of $4.5m$, and deployment height of $4m$, the Stripped Array concept's four telescoping supports are only required to deploy approximately $18.5m$. In comparison, each of the six CTSA wings have a central telescoping boom that deploys longer than $22.5m$.

In regard to volume performance, the Canopy concept performed the worst due to the realized stowage inefficiencies of the membrane. By proxy of the design in its current form, the perimeter cabling required to properly tension the membrane is simply too thick to articulate a reasonable stowage volume. The RCTSA stows close to the $10m^3$ target but inclusion of the support arms and the equal length constraint between the support arms and telescoping sections makes typical stowage volumes exceed this target. On the other hand, the Stripped Array membrane stows remarkably well. While the telescoping sections stow slightly better than in the Canopy concept for the same telescoping section count due to a general decrease in deployment length, it was shown that the membrane component stows much more efficiently. So much so that assembly stowage volumes less than $10m^3$ are likely feasible.

In terms of concept risk, the Stripped Array concept scored the best with regard to requirement risk. Lack of lander clearance sensitivity coupled with a demonstrated positive performance in terms of array scalability were both the primary motivations here. In fact, given the configurations tested, it may be feasible to stay below existing mass and volume targets if the array area is scaled to $1500m^2$. Scaling the RCTSA total array area requirement only exacerbates the aforementioned local buckling problem with poor mass and volume performance. In terms of mission operational risk, the RCTSA was given the highest score on the basis of reduced deployment complexity and design redundancy. That is, the RCTSA is redundant on the basis that several ”wings” are implemented in order to articulate the required array area. Should one fail, total power generative capability is not lost. This is not the case for either the Stripped Array or Canopy, both of which would result in failure if any element of the design fails to deploy or function. While the Canopy concept shares many of the operational risks apparent in the Stripped Array concept, the Stripped Array concept received a lower score due to the fact that its feasibility relies more heavily in regard to the additional payload onboard the lander. Specifically, the Stripped Array concept may not be feasible if other payload on the lander cannot support the stowed array or accommodate space for a central support column. The RCTSA also scored the best in terms of Design and Integration risk due to design features that improve its manufacturability, namely the segmentation of the array in several wings with simple, identical features and parts. Lastly, the RCTSA was attributed the best testing and qualification risk score primarily due to the advantages associated with segmentation of the array into several smaller wings described earlier.

Admittedly, while the AHP method is a widely used and practical decision making tool, it is apparent that its usefulness in the context of this thesis is rather dubious. The AHP method is most applicable in situations involving many more alternatives with more selection criteria and where the relative merit of each is not as straight forward as was in this case. Simple review of the computations performed above for each concept shows that the Stripped Array concept is clearly to the best which beckons the question of why use the AHP at all? While a fair point, it was determined that using a systematic tool for concept down-selection was necessary for the purposes of maintaining a structured design process. If anything, it provides the designer a general method to consciously reaffirm his or her justifications in the down-selection process.

5

Preliminary Design

5.1. Stripped Array Membrane Sizing Sensitivity Study

5.1.1. Objectives

To kick off the preliminary design, a small sensitivity study was performed with respect to the membrane sub-assembly to understand the effect of diagonal cable radius and mesh reinforcement density on how pre-stress propagates through the structure. While not a principle objective of the research, a simple analysis was also performed to understand the influence of total membrane size on the supporting architecture mass and deployed length.

5.1.2. Assessment Methodology

To study the distribution of pre-stress in the structure, two separate analyses were performed. For the first analysis, the stripped membrane Abaqus model introduced earlier in the conceptual design section was slightly modified and utilized to run a series of different geometric configurations. Specifically, the diagonal cable radii tested were $2.5mm$, $5.0mm$, $7.5mm$, and $10mm$ while the strip mesh reinforcement areal masses tested were $22.5\frac{g}{m^2}$, $65.8\frac{g}{m^2}$, $131.7\frac{g}{m^2}$, $176.7\frac{g}{m^2}$, and $220\frac{g}{m^2}$. Again, these areal masses are computed using an assumed amount of $1mm$ Vectran cables per meter that run the length of each strip and are, in turn, used to compute the equivalent radii of the circular cross-section B33 beam elements that approximate each strip. Combinations of areal mass and diagonal cable radius were evaluated assuming a $1000m^2$ total array area,a total of 20 solar array strips per array quadrant, and an innermost strip length of $7m$. With this assumed array area, interior strip length, and strip count, the aforementioned areal masses correspond to equivalent strip radii of approximately $1.8mm$, $3.1mm$, $4.4mm$, $5mm$, and $5.6mm$, respectively. In terms of material assignments, the same material used earlier were maintained here. A few adjustments were made to the model, however. Instead of using constant prescribed displacement at each of the corners of the stripped membrane, a constant force of $50kN$ was instead applied. Furthermore, the python code used to generate the Stripped Array membrane model was adjusted so that the axial stress in each strip could be collected and stored following application of the pretension force.

The second analysis involved taking the same Abaqus model used earlier but adjusting it so that the distribution of pre-stress in the solar array membrane can be evaluated as the total array area and pre-tension load changes. This was accomplished by assuming a constant strip width of $0.5m$ with an a mesh reinforcement areal mass of $66\frac{g}{m^2}$ and incrementally adding strips to the model to increase the membrane area. Each array area configuration was evaluated at three different pre-tension loads. For clarity, these loads were $10kN$, $50kN$, and $100kN$. To that end, the membrane areas evaluated ranged from $240m^2$ to $3200m^2$ corresponding to a total strip count per quadrant ranging from 10 to 50. As a final comment, a diagonal cable radius of $5mm$ was assumed for each configuration tested. Furthermore, convergence studies were performed for the models used in both analyses to verify the quality of the generated data.

With regard to the telescoping mass and deployment length study, simple computations were performed to provide an insight into the mass and length trends of the telescoping supports as a function of total array area. For simplicity, a constant shell cross-sectional radius of $0.5m$ and a material density of $1610kg/m^3$ was assumed. In effect, these mass estimates are slightly conservative. Furthermore, an array deployment height of $4m$ and a lander diameter of $9.1m$ was also assumed. With regard to the telescoping support mass, masses were computed for shell thicknesses of $1mm$ and $4mm$.

5.1.3. Results and Discussion

Figure 5.1 shows the pre-stress apparent in each strip for all the diagonal cable radius and mesh reinforcement combinations tested. Each of the four subplots shows the pre-stress versus strip number for each of the four diagonal cable radii tested. With respect to the strip count, the lowest strip number,1, refers to strip closest to the center of the array while the largest, 20, refers to the outermost strip.

With regard to trends in the data, there are a number worth mentioning. Beginning with the first subplot, a diagonal cable radius of $2.5mm$ appears to be too small for high strip mesh reinforcement areal masses. Essentially, the diagonal cable lacks enough stiffness to transfer much of the applied pre-tension load to the more inboard strips. As an example, the apparent pre-stress in the innermost strip for the $65.8\frac{g}{m^2}$ strip areal mass configuration, approximately $5e6Pa$, is only reached in the $220\frac{g}{m^2}$ configuration till about the 12th strip. The remaining pre-stresses in the more inboard strips are well below this value which could result in insufficient stress-stiffening to tolerate transverse loads on the strips. As the diagonal cable radius is increased, its stiffness increases which results in the transfer of more of the pre-load to the inner strips. For example, for a diagonal cable radius of $5mm$, the innermost strip for the $65.8\frac{g}{m^2}$ configuration increases in pre-stress from approximately $5e6Pa$ to $2.5e7Pa$. At the same time, this increase in diagonal cable stiffness also means that the cable is wont to displace less under the same applied $50kN$ load. This has the tendency to lower the pre-stresses in the more exterior strips.

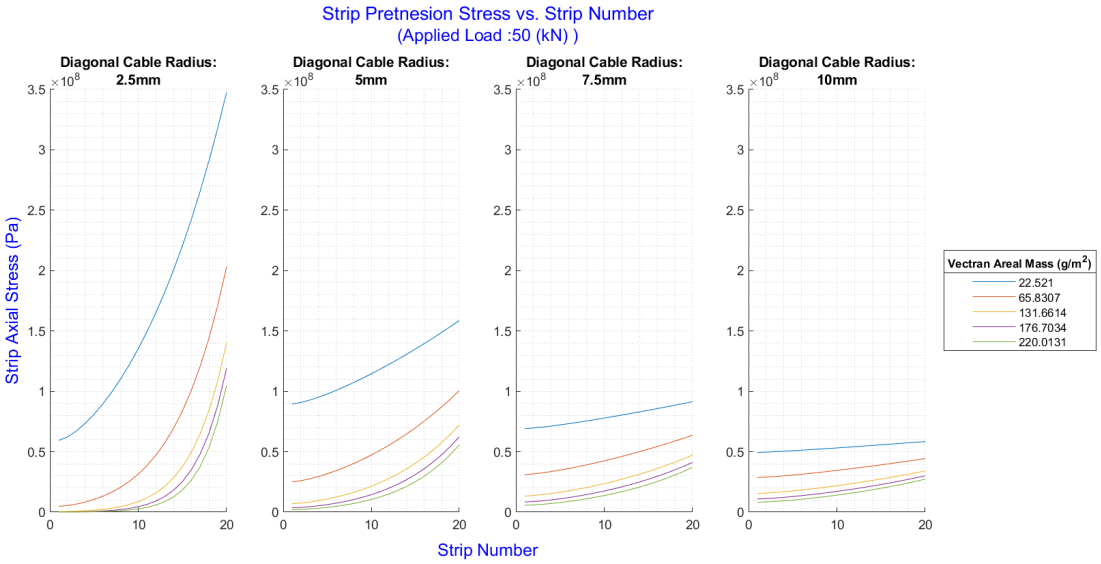


Figure 5.1: Assuming an array size of $1000m^2$,20 solar array strips in total per quadrant,and a pre-tension load of $50kN$, each subplot shows the axial stress in each strip for a solar array membrane with different diagonal cable radii. From left to right, these diagonal cable radii are $2.5mm$, $5.0mm$, $7.5mm$, and $10mm$.

Figure 5.2, on the other hand, shows how the pre-stress varies at various strip locations, under different applied pre-tension loads, as a function of total solar array surface area. Specifically, under each applied pre-tension load, the axial stress of the innermost strip, middle strip, and outer most strip for each total membrane area is reported. Effectively, at each array area, a spread in terms of pre-stress across a membrane quadrant can be discerned. As expected, pre-stresses throughout the membrane

decrease as the total area increases when subjected to the same pre-tension load. Furthermore, the pre-stress spread across the membrane increases with increasing total membrane area for each of the three pre-tension loads evaluated. For a $10kN$ pre-load, the difference between the innermost and outermost strips pre-stress is approximately $1e7Pa$ and $1.5e7$ for total membrane areas of $240m^2$ to $3200m^2$, respectively. For a $50kN$ pre-load, the differences are approximately $5.14e7Pa$ and $7.34e7Pa$. Lastly, for the $100kN$ pre-load, the differences are approximately $1.03e8Pa$ and $1.47e8Pa$. Notably, there is a clear bias in terms of pre-stress towards the outer strips in comparison to the more inboard strips. This is observed given the fact that the ”middle” strip pre-stress for every area and pre-tension combination tested is closer in magnitude to the innermost strip.

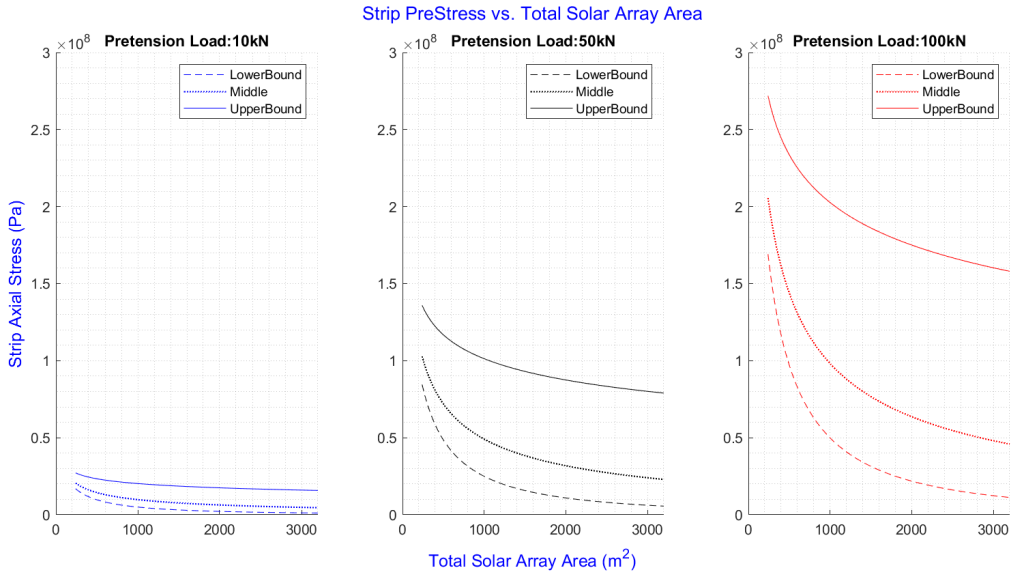


Figure 5.2: Each subplot shows the axial pre-stress in the innermost, middle, and outermost solar array strip per quadrant as a function of total membrane area for different pre-tension loads. These pre-tension loads are $10kN$, $50kN$, and $100kN$. Moreover, the assumed diagonal cable radius and mesh reinforcement areal mass for this study were $5mm$ and $66\frac{g}{m^2}$, respectively.

In the final study, the total mass of all four telescoping supports for telescoping shell thicknesses of $1mm$ and $4mm$ as a function of total membrane area is shown in the first subplot of Figure 5.3. Furthermore, the deployment length of each telescoping support as a function of total membrane area is also shown in the second subplot of Figure 5.3. As the membrane area increases, the telescoping support length increases which also increases the total system mass. In turn, this increases the support's susceptibility to buckling, requiring either more supports along its length or an increase in shell thickness to bolster its durability. As it stands, the $1000kg$ mass limit is a rather formidable requirement and its unlikely that increasing shell thickness is a viable option.

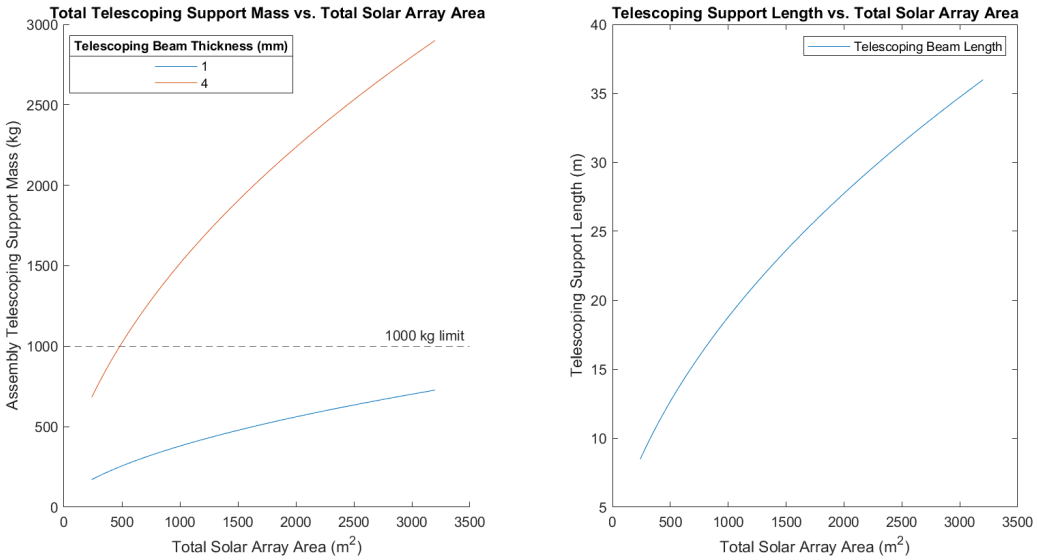


Figure 5.3: The first subplot shows the combined mass of all four telescoping support assuming a constant telescoping shell radius of $0.5m$ for shell thicknesses of $1mm$ and $4mm$. The second plot shows the deployment length of one telescoping support as a function of total membrane area assuming an array deployment height of $4m$ and a lander diameter of $9.1m$.

5.2. Array Assembly Parametric Study

5.2.1. Objectives

The primary objective of this analysis was to understand how changes in level of pretension of the solar array membrane strips in conjunction with alterations to the supporting architecture's telescopic booms and support condition affect assembly stresses, deflections, and natural frequencies. In essence, the goal was to see how the solar array assembly behaves structurally when the originally stipulated telescoping support condition is reduced as shown in Figure 5.4. The idea here is that the pre-tension cable could be oriented in this fashion to reduce bending of the telescoping support tip without the need for additional stabilizing wires and support arms. If the reduced condition shown here is viable, it would severely reduce the complexity of the system and possibly reduce system mass. In addition to this study,system response to variations the level of gravitational loading was also studied.

5.2.2. Assessment Methodology

To meet this objective, the following analysis approach was taken. Taking advantage of Abaqus's Application Programming Interface (API), a highly custimizable parametric python script was developed for building and evaluating finite element models in Abaqus. Specifically, for a number of predefined study variables, this script takes a range of values per parameter, generates combinations based on these values, builds an Abaqus model for every combination, and performs a non-linear, static structural finite element analysis of each. In the case of this overall study, these input parameters were telescoping beam thickness, telescoping beam radius, pulley cable stroke, and magnitude of gravitational loading. For details with regard to the python script developed for this study, see Appendix B. Note, wind loads were omitted from this preliminary study due to the fact that the magnitude and dynamics of wind on Mars is still very loosely understood. Therefore, a rigorous study including this type of loading does not make much sense for the time being.

5.2.3. Model Overview

General Model The generic model developed herein consists of an assembly model divided into two sub-assemblies,membrane strips and the supporting architecture, and is presented in Figure 5.5. Similar to the model developed in the conceptual design phase, the membrane strips are idealized as very slender, solid circular cross-section 3D Euler-Bernoulli beam elements (B33 elements), shown as solid light-blue lines in Figure 5.5. Again, the cross-section of each strip is constant, depends on the chosen mesh reinforcement areal mass and thickness, and does not vary from strip to strip. With

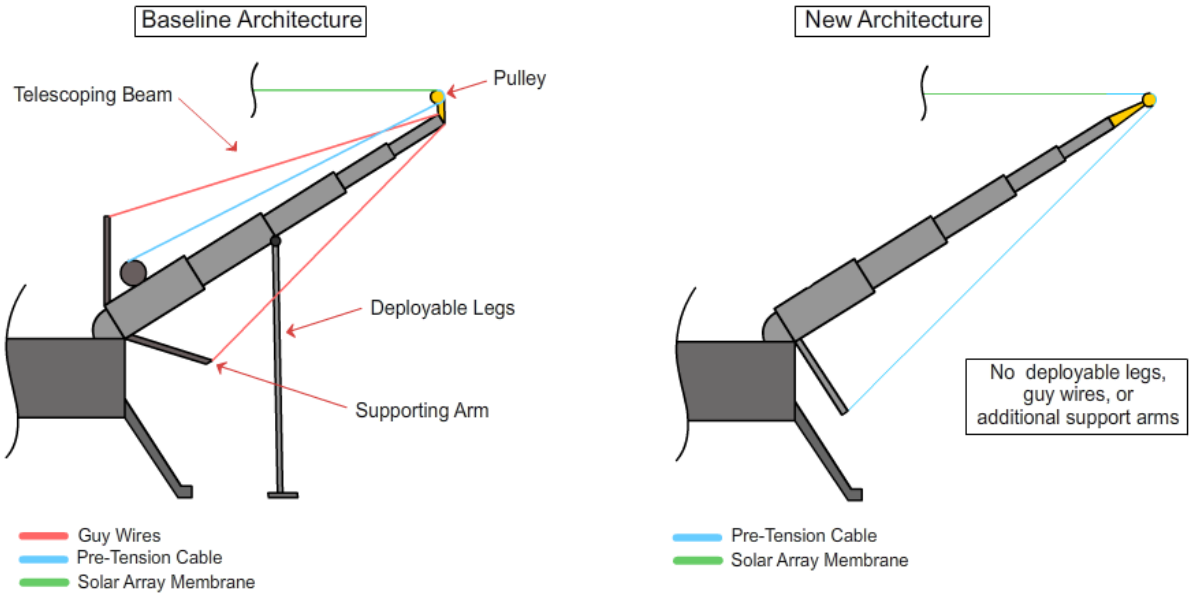


Figure 5.4: On the left is the original telescoping support architecture first introduced in the conceptual design phase. Notably, the tip of the telescoping support is stabilized by guy wires and the pre-tensioning cable that wraps around the pulley follows parallel to the deployed support as shown. The right configuration shows a reduced version of this architecture, where the guy wires are removed and the pre-tension cable is assumed to navigate around the pulley and connect to a support arm offset from the telescoping support root.

regard to material, the default for this component was chosen to be the high performance polyarylate fiber, Vectran. Effectively, the strips in this model only represent the mesh reinforcement meaning the remaining constituents of the membrane are not physically modelled like the solar cells, substrate material, etc. Instead, these non-load-bearing elements are represented in the model through the attachment of non-structural mass to each of the strips. Naturally, the amount of mass applied to each strip depends on their position, where more outboard strips from the array center are longer and thus support more solar cells. Furthermore, each of these simplified strips attach at their tips to the diagonal cables which are shown as solid green lines in-plane with the strips in Figure 5.5. Additional, these diagonally running cables have a thicker, constant circular cross-section in comparison to the strips and are represented by B33 beam elements as well and are comprised of a Ultra High Molecular Weight Polyethylene (UHMPE) material.

On the opposing diagonal cable ends, each connect to pulley cables which navigate around the supporting architecture. These pulley cables are represented in the model by special connector elements from the Abaqus element library called slip rings, shown as the dashed blue lines in Figure 5.5. According to the Abaqus Analysis User's Manual, slip rings are complex connector elements that model material flow and displacement between two nodes and are useful in modelling various types of belt systems like pulleys [88]. In practice, a pulley system can be modelled through the linkage of two adjacent slip ring elements, as shown in Figure 5.6, where the pulley radius that would otherwise be at the connection point of these elements is ignored. Tension on one end of the connected slip rings results in material flow around node "b".It should be noted that these types of connector elements do not impose any kinematic constraints between the two connected nodes.Friction between a cable navigating around a pulley and the pulley can be captured through the specification of a contact angle, α, defined as the angle of intersection between the two slip rings. However, friction effects were omitted for the purposes of this preliminary study. Furthermore, elasticity behavior of these elements were defined as uncoupled, linear elastic where their axial stiffness were attributed using the simple equation $K = \frac{EA}{L}$, where E,A, and L are the pulley cable's Young's modulus, cross-sectional area, and length, respectively.

In total, there are three slip ring connectors in series per telescoping beam that form two pulleys.

As stated previously, the first slip ring connects at the terminus of the diagonal cable and joins with another slip ring connector at the tip of the telescoping beam to form the first pulley. From the tip of the telescoping beam, the second connector navigates downwards and joins with the last slip ring near the base of the telescoping beam to form the second pulley. The position of this junction is governed by the predefined length and angle of deployment of the supporting arm, included as constants in the developed python script. Specifically, in this analysis the support arm was assumed to be 2 meters in length and deployed at a 90 degree angle in reference to the deployed telescoping support. As shown in Figure 5.5, the support arm was not included in the model. Lastly, the third slip ring extends from this junction horizontally to an arbitrarily chosen distance.

With regard to the telescoping beams, these beams were also modeled using B33 elements where each of the telescoping sections starting from the root of the beam were given separate, progressively smaller cross-sections. The number of these sections were assumed to be twelve in total per deployed beam for reasons that will be described later in this chapter. The telescoping beam sections were assigned material properties taken from CES Granta for a quasi-isotropic layup of high strength carbon fiber material. Specific material properties for this component as well as the others in the model are shown in Table 5.1.

Lastly, mesh sizing was determined based on a convergence study performed on an assembly model with a total array area of $1000m^2$. While keeping element count low is advantageous from a simulation time point of view, particularly in this application where hundreds of models were evaluated, ensuring convergence of stresses and deflections is paramount.

Table 5.1: List of materials and their properties used in each model as well as what component they were assigned to.

Material Name	Material Properties				Component
	rho (kg/m^3)	YS (Pa)	E (Pa)	v	
Vectran (Polyarylate fiber)	1400	2.9E+09	5.5E+10	0.35	Strip Reinforcement
Spectra 1000 (UHMWPE Fiber)	980	2.5E+09	1.2E+11	0.41	Diagonal Cables & Pulley Cables
Carbon Fiber Composite (QI-Epoxy Matrix Weave)	1610	4.50E+08	4.40E+10	0.33	Telescoping Beams

Boundary Conditions In terms of boundary conditions, each component is constrained in different ways, some of which are highlighted in Figure 5.7. At the center of the deployed solar array, the root of the diagonal cable is pinned along the three global coordinate axes. Extending outward to the tip of the telescoping beam,the end of this cable is coupled in terms of all Degrees of Freedom (DOFs) to the first node of the first slip ring in the series of slip ring connector elements that form the pulley system. Additionally, a zero material flow constraint is applied at this connection point. As mentioned previously, slip ring elements includes an additional DOF referred to as material flow. At the junction of two slip rings elements, material flows freely between them depending on the magnitude and direction of an applied load or displacement. This relative motion must be restricted at the node of a slip ring element which connects and interfaces with another feature in the model that is not a slip ring. That said, the junction of two slip ring elements at the tip of the telescoping beam is fully constrained along all DOFs to this tip. In effect, this coupling makes it so that as the telescoping beam deflects, the idealized slip ring pulley moves with it. It should be noted that material is still allowed to flow between the two attached slip ring elements regardless of this applied motion constraint.

As shown in Figure 5.7, the root of the telescoping beam is assumed to be fixed as is the junction between the second and third slip ring connector elements. The terminal node of the third slip ring element is fixed along all DOFs except for one where a prescribed displacement is applied towards the center of the array. Note, this prescribed displacement is referred to as ”stroke” for the remainder of this section.Similar to the interface between the diagonal cable and the first slip ring node, a zero material flow constraint is also applied to this aforementioned terminal node.

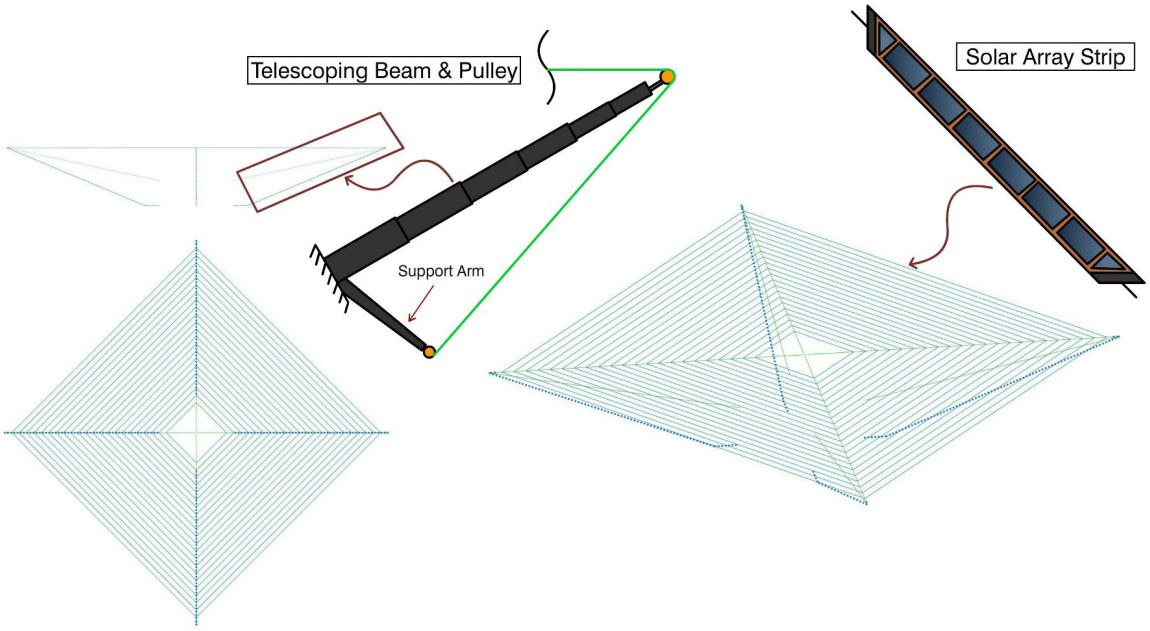


Figure 5.5: Representation of the stripped solar array design in Abaqus. Array top and side views are on the left and an isometric view is on the right.The membrane strips are idealized as very slender, solid circular cross-section B33 elements.The cross-section of each strip is constant, depends on the chosen mesh reinforcement density and thickness, and does not vary from strip to strip.

With regard to the idealized solar array strips, their ends are coupled in terms of displacements only to the diagonal cables in the three global coordinate directions. Under these boundary conditions, torsional constraints were also assigned to every strip to prevent numerical singularity errors during simulation caused by unabated self-rotation. Coupling of displacements only between the strip ends and points along the diagonal cables was applied to capture a critical characteristic of the Stripped Solar Array concept: decoupling of rotations between the two sub-components. Given the symmetric nature of the Stripped Solar Array design, the boundary conditions detailed above are extended to the other three telescoping beams, strip quadrants, and pulley systems.

Load Steps While input geometric and loading parameters vary between models, the analysis methodology is the same. Specifically, each model has an initial load step where the aforementioned boundary conditions are applied. Following this initialization step, the non-linear geometry assessment flag is turned on and the applied displacements at the ends of the pulleys load the assembly in a separate load step, aptly referred to as the ”Pretension” step. After the applied load is adequately distributed through the structure, a gravitational load is applied in a separate load step where the magnitude varies from model to model. Once static equilibrium is reached, a modal analysis is performed in the last load step. Natural frequencies in this load step are derived using the Abaqus default Lanczos eigenvalue extraction method. The reason for this order of operations is because out-of-plane loading cannot be applied to the extremely slender beam elements that model the solar array membrane sub-assembly before they are pre-loaded. Prior to being pre-loaded, these elements have a very small bending stiffness and rely on stress stiffening to tolerate out-of-plane loads. Solution convergence in this static, general model is extremely difficult to achieve should gravity be applied first.

The lack of rotational DOF coupling at the strip-diagonal cable interfaces in combination with the highly asymmetric stiffness of the unloaded elements used to represent the solar array membrane both introduce local instabilities in the model when the assembly is first loaded. To combat this, automatic stabilization is turned on to assist convergence during the pretension load step. Essentially, automatic stabilization introduces an artificial volume proportional dampening into the global equilibrium equations which acts as a pseudo inertia that dissipates the strain energy released by the aforementioned

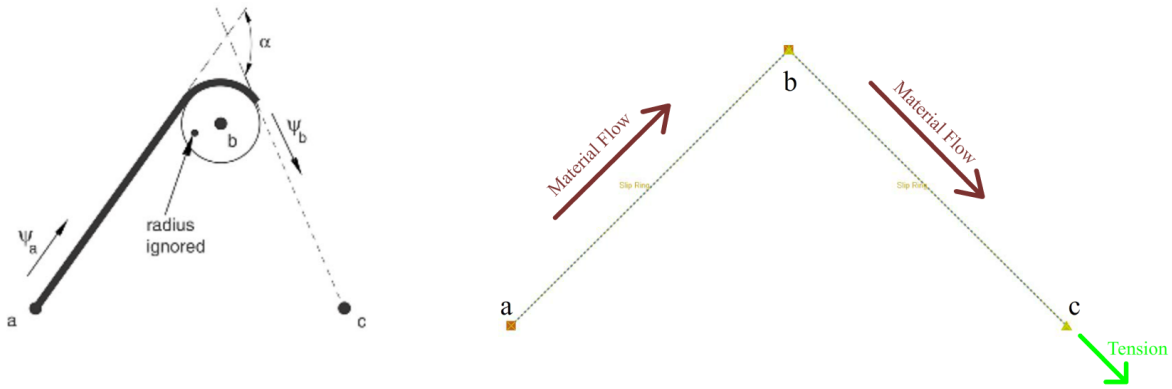


Figure 5.6: Abaqus slip ring connector elements in a configuration to model a pulley.Displacement or loading applied on one connector end results in material flow around node b.

instabilities to assist convergence. This dampening is represented by the inclusion of viscous forces which take the form of:

$$F_v = cMv \tag{5.1}$$

Where c is a damping factor, M is an artificial mass matrix calculated with unity density, and v is a vector of nodal velocities defined by the nodal displacements over the time increment size. The global equilibrium equations are thus:

$$P - I - F_v = 0 \tag{5.2}$$

The use of stabilization should be minimized as much as possible as it artificially removes energy from the system that would otherwise deform the structure. In effect, stabilization adds inaccuracy to the computed solution. For this analysis, stabilization is only required during the beginning stages of the pretension load step after which the elements introducing the instabilities in the model are sufficiently preloaded and develop the desired stress stiffening effect. The default Abaqus damping factor of 0.0002 was used.In order to verify the accuracy of the derived results using stabilization, a comparison between the energy dissipated by viscous damping and the total strain energy in the model after static equilibrium is achieved is required. Per the Abaqus Analysis User's Guide, viscous damping energy should be less than 5 percent of the total strain energy to ensure solution accuracy [88].

Evaluation Criteria For each of the models analyzed, results were stored and their performance were evaluated on the basis of a number criteria. Separated into component and assembly levels, these criteria included are shown in Figure 5.8.

In terms of the individual components, the membrane was evaluated on the basis of stresses and displacements. Max stresses in the solar array strips as well as diagonal cables were compared against their respective yield strengths and care was taken in verifying that compressive stresses were not present.The presence of compressive stresses in the membrane component would mean the model is not accurately capturing the physics of the problem at hand, namely that cables can only tolerate tensile loads. As shown in Figure 5.8, evaluation on the basis of displacements was divided into three important categories. The first was the maximum displacement of the entire membrane sub-assembly. The second evaluation metric relates to the sag of each individual strip. Specifically, the difference between the minimum and maximum point of every strip was taken at the end of the gravity load step and an angle was computed to understand how much each deviates from a horizontal strip (Figure 5.9). Lastly, since each model consists of several strips running parallel to one another in close proximity, the issue of shading could be a problem should there be a large relative difference in sag between strips. Consequently, the absolute value of the maximum deflection was taken for each strip and subtracted

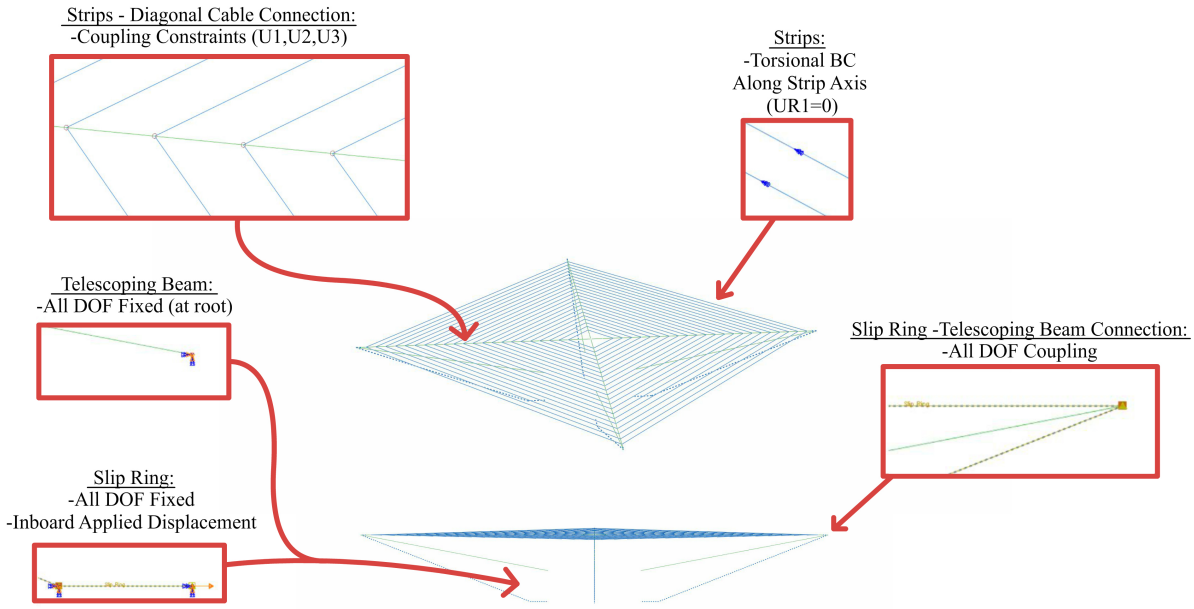


Figure 5.7: Some of the boundary conditions used to model the stripped solar array assembly.

by the absolute value or sag of the inner most strip. For simplicity, a quick sum was taken of these values and compared between models. This summation represents an accumulation of sag, meaning higher values reflect configurations that are less flat whereas lower values represent models with that are more flat.

The reason why sag of each strip was chosen to be reported in terms of an angle was twofold. First, reporting sag as a displacement is problematic when dealing with strips of varying length. Half a meter of strip sag is less of a problem for a strip that is 20 meters long versus one that is 2 meters long. Reporting sag as an angle thus provides a bit better of an insight into general membrane deflections. Secondly, in the SAWS study, $0.5m$ of sag was deemed acceptable for a CTSA wing with $22.5m$ long solar array membranes. Solar array collection efficiency was one of the factors leading to this acceptance. By using an angle to report strip sag, a comparison can be made between the CTSA and solar array strips in a given Stripped Array configuration.

Similar to the membrane, material yield was also assessed for the telescoping beams as well as the magnitude of deflections at the tip of each. Moreover, as shown in Figure 5.8, both Local Shell Buckling (LSB) and Euler Buckling (EB) of the Telescoping Beams were two failure modes checked for. For LSB, computation was performed using same equation equations taken from the NASA's buckling of thin-cylindrical shells specification used in the previous chapter[84]. For EB, a different approach was made. For conservatism, a fixed-free connection was assumed using a continuous cross-section column whose cross-section is made of the smallest telescoping section. Under these end support conditions, the equation for the critical EB load is given by

$$P_{cr,EB} = \frac{\pi^2 EI}{(kL)^2} \tag{5.3}$$

where I is the moment of inertia of the shell, E is the material young's modulus, and L is the length of the column. k is a factor associated with the support conditions which, assuming a fixed-free connection, equates to 2. Additionally, a safety factor of 2 was applied which in effect reduces the critical load by half. Lastly, the slip ring connectors which model the pulleys were evaluated for material yield and, similar to the membrane component, verified to not have any compressive stresses.

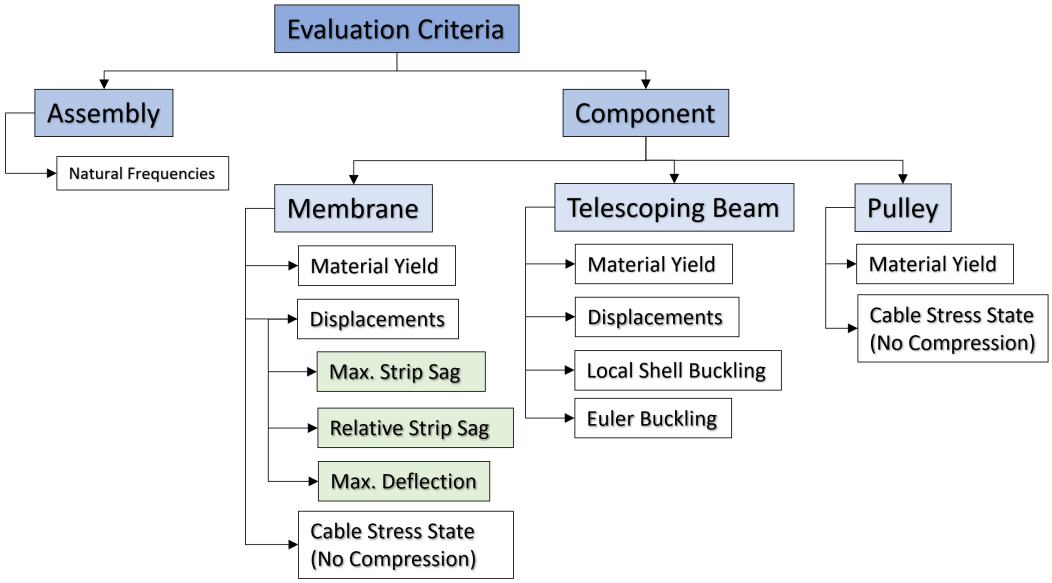


Figure 5.8: Evaluation criteria for each model assessed, subdivided into several categories.

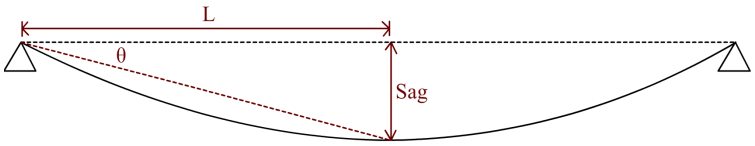


Figure 5.9: Derived angle for each strip on the basis of sag.

Constants Collectively, there are many different aspects of the stripped solar array design that are worth adjusting and studying the concept's response and performance. However, in order to limit the total number of tested configurations, it was imperative that the number of these adjustable variables be kept to a minimum. Consequently, a number of design characteristics of the stripped solar array were judiciously kept constant. A list of these constants and the rationale behind them are presented below. Where appropriate, the variable designation used in the aforementioned python code is reported.

1. Inner Diagonal Length, (L_i)

   The length of the shortest solar array strip located near the center of the solar array was chosen as a constant to mitigate interaction between the deployed strips and the rest of the lander payload. Moreover, there should be ample room for the stripped array deployment mechanisms and housing at the center of the lander. However, reducing the length of the inner most strip is ideal because it reduces the sizing of the rest of the components in the assembly, namely the telescoping beam as well as the length of the longest strip. That said, the inner strip length determines the offset distance of the array from the center through the division of its length by $sqrt(2)$. At $7m$, the offset distance of the array from the center extends mostly beyond the radius of the lander that was assumed for this study. As a result, this value was used. Note, this is a departure from what was assumed in the conceptual design where an L_i of $4.5m$ was selected. However, it was determined prudent to extend this value to $7m$ for the aforementioned reasons.

2. Strip thickness, Diagonal Cable thickness, Number of Telescoping Sections, and Number of Strips, (t_s, r_{dc}, TS, k)

   Each of these parameters were selected on the basis of stowed volume considerations. Using the learning acquired in the conceptual design phase, a strip count of 22 and mesh thickness of $1mm$ was selected in conjunction with a diagonal cable radius of $7.5mm$. For a $1000m^2$ solar

array, the array stows to a volume of around $3.7m^3$. With respect to telescoping section count, six was selected for this preliminary study.

3. Pulley Setup, $(SlipRing_r)$

   The sizing of the slip ring connectors used to represent the pulley cables, namely their radius, was determined primarily by the radius of the Diagonal Cable and the axial stiffnesses of the attached solar array strips. In order for the pulley cables to efficiently stretch the stripped solar array, the cables needs to overcome the combined stiffness of these aforementioned elements. As a first pass, the slip ring connectors were given a radii twice the size of the chosen diagonal cable radius ($0.015m$). No sensitivity study was performed to explore the most optimal sizing of the pulley cables at this time However, this should probably be investigated further in future work.

4. Mesh Reinforcement Areal Mass

   The mesh reinforcement areal mass was partially selected on the basis of reducing the mesh's contribution to the total assembly mass as much as possible. Assuming a $1mm$ thick continuous sheet of Vectran that spans $1000m^2$ and has a material density of $1400kg/m^3$, the mesh mass would exceed the allotted structural mass budget ($1000kg$) for the deployable solar array. Clearly,as mentioned in the previous chapter, a low areal mass, open mesh is required. Learning acquired from the previously described stripped array pre-tensioning sensitivity study motivated the areal mass selection as well. Ultimately, an areal mass of $43.82\frac{g}{m^2}$ was selected. For a solar array of $1000m^2$ and a strip count of 22 and L_i of $7m$, this results in approximately 23, $1mm$ thick Vectran support cables per strip. The equivalent beam element thickness used in the model under these conditions is approximately $5mm$.

5. Cargo and Telescoping Support Height $(CE_{Height}, TS_{Height})$

   As mentioned previously, the final deployment height of the telescoping support beams is dictated by the stowage position of the array. This stowage position, in turn, is dependent on the nature of the additional payload located beneath the array. As with most aspects a manned mission to Mars, the exact nature of this payload with respect to its contents and sizing is not established yet which introduces an element of uncertainty here. In the reference SAWS report, the only aspect of the reference lander used for trade studies and concept generation that was reported was the deck height which was assumed to be 4.8 meters from the Martian surface. For the purposes of this study, it was assumed that the additional cargo would stand no more than 3 meters high and the array would deploy half a meter above it. In effect, the telescoping support beam tips would be at a height of 3.5 meters.

5.2.4. Results and Discussion

In this section, the results gathered for models with solar array areas of $1000m^2$ will be described.

To begin, the minimum membrane deflection versus telescoping beam radius with a cross-sectional thickness of $1mm$, shown in Figure 5.14, is evaluated. As expected, the absolute magnitude of out-of-plane strip deflection increases with increasing gravitational loading at all pretension levels tested. As the level of stroke increases, these displacements are reduced and even reach positive values beginning with a stroke of just $0.15m$. This behavior is a consequence of how the tip of the telescoping beam moves in response to the incident pretension and gravitational loads. Simply put, the telescopic beam tends to deflect downwards towards the Martian surface due to self weight and the absence of significant tensioning of the pulley cable. The degree of this deflection, of course, scales with the amount of gravitational loading. As the stroke is increased, the beam tip begins to move in the opposite direction as a result of the pulley system setup as well as the beam's flexural stiffness. Specifically, the cable which wraps around the pulley located at the tip of the telescoping beam enters and exits the pulley at different angles. As shown in the simplified Figure 5.11,this disparity results in the beam being pulled in the direction of the larger angle. This beam behavior is confirmed in plots Figure 5.20,Figure 5.21,and Figure 5.22 which show the deflections of the telescoping beam tip in response to increases in stroke at various levels of gravity and beam thickness. Ultimately, the degree of bending is dependent on the flexural stiffness of the telescoping beam and the difference in magnitude of the aforementioned angles. In the case of this study which involves a $1000m^2$ array, the telescoping beam extends approximately $19m$ and and the cross-sections analyzed are relatively small by comparison due to packaging volume

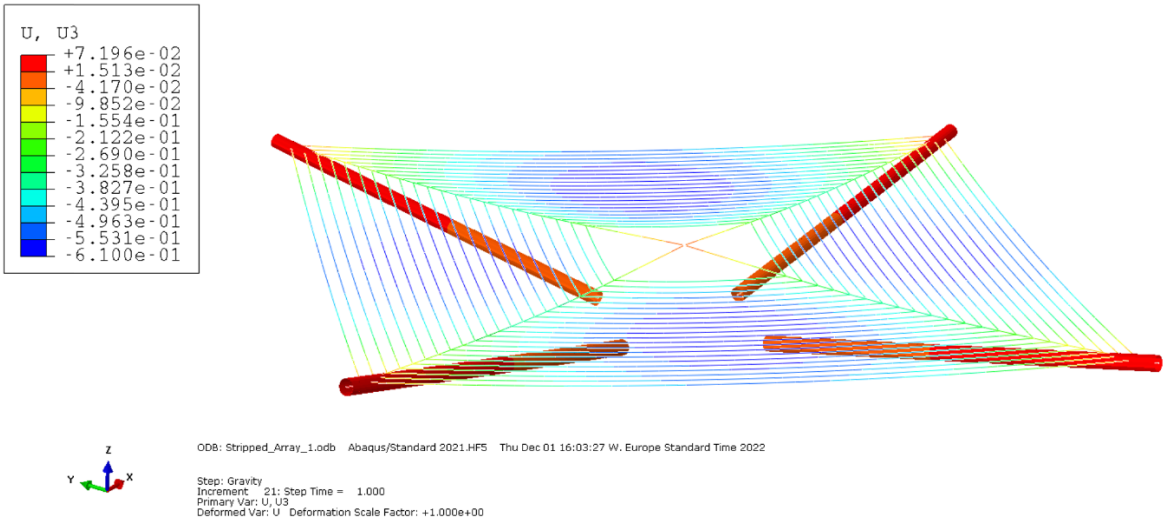


Figure 5.10: An example of deflections at the end of the gravity load step. In this plot, gravity is set at 50 percent of Earth's Gravity, membrane area is $1000m^2$, level of stroke is kept at $0.05m$, and the beam thickness and root radii are $1mm$ and $0.350m$,respectively.

constraints.Flexural stiffness of the beam is thus low. Moreover, the angle of deployment of the supporting arm as well as its limited length make it such that the angle in which the membrane meets the tip of the telescoping beam is larger than that of the angle in which the pulley cable makes between the support arm and telescoping beam. Collectively, these characteristics produce the positive trend seen in Figure 5.14 where the minimum deflection of the membrane increases with increasing stroke. The tendency for the displacement to decrease with increasing radius of the telescoping beam at higher stroke levels is primarily due to improved flexural stiffness as well as an increase in self-weight. As the beam cross-sectional thickness is increased to $2mm$, Figure 5.15, and $3mm$, Figure 5.16, the phenomena previously described becomes less apparent. These increases in thickness effectively bolster the flexural stiffness of the beam thereby preventing less tip deflection.

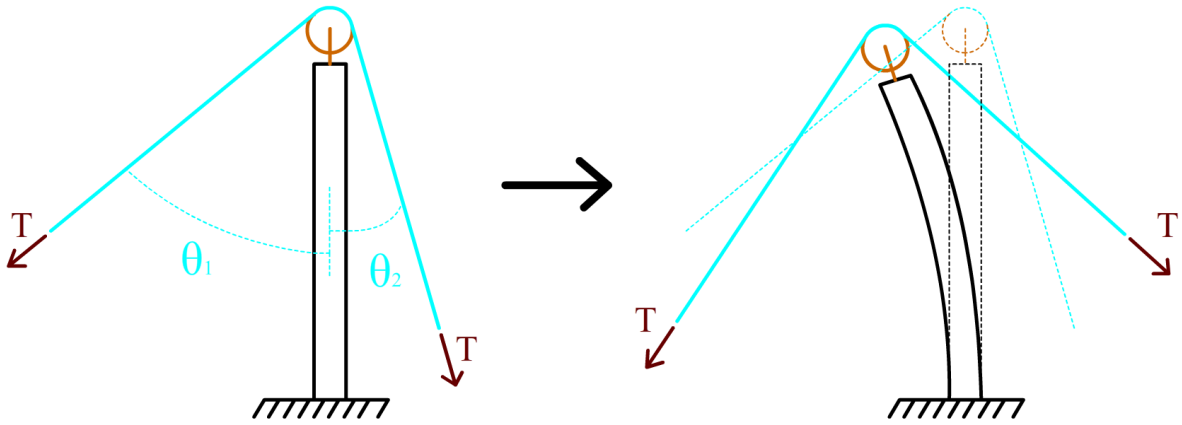


Figure 5.11: This figure demonstrates the concept of beam bending as the result of unequal angles in which a tensioned cable navigates around the pulley. θ_1 is larger than θ_2 causing a larger axial portion of the tension, T, to pull the beam to the left.

Of course, the minimum membrane deflection is only part of the story. As previously mentioned, sag of each strip in the solar array membrane, measured in terms of an angle, was collected for every model and the minimum of each model was then cataloged and reported in Figure 5.17. Consisting of

15 plots in total, this figure shows the minimum strip sag angle versus stroke for all the combinations of beam thickness and gravitational loading. From inspection, beam parameters in general appear to have a minor influence on the minimum strip sag. In fact, the driving factor is for strip out-of-plane deflection is the level of tension provided by the pulley tensioning system. This makes sense given the fact that higher strokes increase the stress-stiffening effect in the strips thus reducing its out-of-plane displacement.At a beam thickness of $1mm$, however, there is a slight variation in strip sag between different telescopic beam radii as stroke and level of gravity increase. At low gravity and high stroke, smaller telescoping beam radii have a tendency to bend upwards more as previously mentioned which has the tendency to relieve stress in the strips needed for flexural stiffening. At higher gravitational loads, the beam does not deflect as much as a result of increased self-weight. To provide additional context to all the plots, the outermost strip length, given a membrane area of $1000m^2$ and the assembly constants provided in the previous section, is approximately 31 meters in length. Per Figure 5.17, the maximum sag angle for any of the configurations tested is about 2 degrees. Thus, the maximum strip sag for all the models tested is about $0.5m$.

The relative flatness of the membrane in each model in response to various degrees of stroke and gravitational loading as well as telescoping beam thickness and radii is shown in Figure 5.18. Unlike with the minimum strip sag angle, beam properties are seen to have an affect on the overall level of flatness of the membrane. At a low beam thickness and gravitational load, all tested beam radii show a tendency to become more flat in response to low to medium levels of stroke. This trend changes after $0.15m$ of stroke, where a linear increase in accumulated deflection is observed first with smaller cross-section beams and extending to larger cross-sectioned beams as the stroke is increased. Again, this behavior is a consequence of the tensioning scheme for the structure as well as the flexural stiffness of the telescoping beams themselves. That is, as the stroke is increased beyond the point in which gravity and the bending stiffness of the beam can resist the tensioning load on the pulley cable, the tip of the beam bends upwards. This results in a membrane shape akin to an inverted pyramid, where the inner most solar array strips are the lowest point of the membrane and the location of each strip thereafter in space linearly increases. This upwards bend phenomena is reduced with increasing gravitational load. Moreover, improvements in beam thickness is shown to reduce the onset of this upwards bending which again is a consequence of improved flexural stiffness and increased self-weight. The maximum sag was taken for every strip in each model and plotted to gain a graphical representation of the final membrane shape at the end of the gravity load step. Instead of reporting hundreds of curves, a membrane shape envelope was instead developed and presented in Figure 5.19. The aforementioned inverted pyramid shape is apparent as the upper bound of this envelope. Lastly, it is observed that in many of the subplots of Figure 5.18 there are noticeable intersections between beam radii curves. For example, for a beam thickness of $1mm$ and a gravitational load of 150 percent of Earth's, this intersection is very apparent. Smaller beam radii bend downwards more in the presence of large gravitational loading and low stroke. As the stroke increases, these beams allow the membrane to trend flatter faster than the beams with larger radii again due to the combination of flexural stiffness and self-weight.

The telescoping beam, given different radii and thicknesses, was evaluated for Euler buckling given multiple different gravitational loads as well as level of pulley stroke. The absolute maximum compressive force in the beam was taken for each model and divided by the computed critical column buckling load as described in the previous section. The result is shown in Figure 5.23, which reports each model's resistance to buckling essentially as a factor where any value over 1 means the configuration would buckle. Quite clearly, the chosen telescoping support structure will buckle under most of structural configurations and incident loads tested. This effectively renders the reduced telescoping support configuration evaluated in this section a failure. It is important to note, however, that the method used herein to compute the column buckling a few layers of conservatism. Since the structure is telescoping in design, it is likely a bit more resistant to global buckling. Furthermore, the end conditions for the telescoping support is most likely a mixture between fixed-free and fixed-pinned given the pulley connection at the beam tip. This would likely further improve the telescoping support's buckling resistance. On the other hand, the column buckling equation used here assumes an initially straight fixed-free beam. Initial curvature in the column has the effect of reducing its resistance to buckling and it was already shown earlier that the telescoping beam's tip does have a tendency to deflect. A safety factor of 2 was included in the computation for this reason as a first pass but it is unclear whether or not this is sufficient.

Unsurprisingly, local shell buckling appears to be another threat to the feasibility of the stripped array design under the reduced telescoping support condition. Similar to the column buckling evaluation, LSB is reported in Figure 5.24 as a buckling ratio, computed using equations outlined in the previous section. As shown in Figure 5.24, designs which have $1mm$ in thickness are largely at risk of failing. Increasing the telescoping beam thickness has the effect of diminishing the LSB factor substantially due to the reduction of bending loads in the cross-section. At the $2mm$ thickness, increasing the stroke too high results in LSB factors close to buckling of the structure. As shown in the top row of subplots, this concern is largely diminished when increasing the shell thickness to $3mm$. For all cases tested, larger gravitational loads have the effect increasing the buckling factor at lower strokes but reducing those at higher strokes. Like everything presented prior to this, beam flexural stiffness and self-weight are the primary reasons for this behavior. As a final note, the equations used to compute these buckling loads are meant for unstiffened isotropic shells where results can be significantly different when actually accounting for the interactions inherent to multilayered composites consisting of several orthotropic layers. Furthermore, the equations used are only experimentally validated for shell length over shell radius ratios of less than 5 [84]. In this study, ratios exceed 12 which adds another element of uncertainty to the reported results. Consequently, the results reported herein are only meant to capture trends. In any case, for this study it was also decided to relegate optimal composite layup studies to the detailed design phase. Without a good understanding of the layup of the beam cross-section, it is not easy to ascertain with more rigor the buckling performance of the structure. That said, based on the data $1mm$ may be too thin.

Based on the nature of the telescoping beams, namely their exceedingly large length-to-radius ratio and thin cross-section, it was expected that failure would occur either through buckling or excessive deflection prior to material yield being a concern. As shown in the first subplot of Figure 5.25, this is indeed the case. Compressive and tensile loads are both well below their respectively yield limits for every configuration tested. In fact, the other components in the the solar array assembly demonstrate a similar character where tensile loads are exceedingly low. Given the low stresses in Diagonal Cables, these cables are likely oversized and could possible be reduced. Additionally, low stresses in the strips implies that either a reduction in size or quantity of the reinforcing mesh could be made as well.

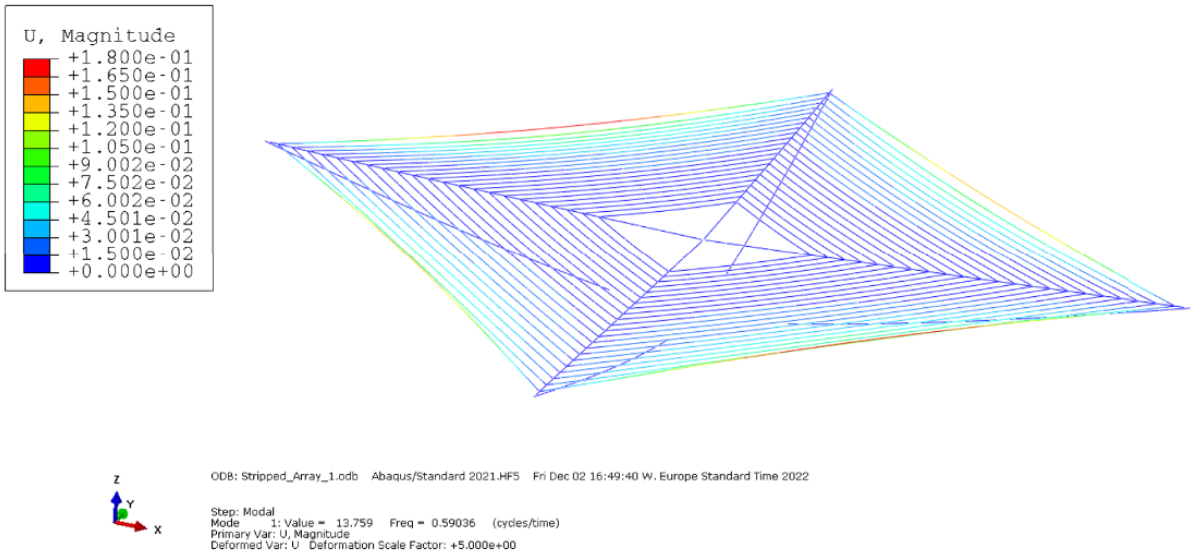


Figure 5.12: First mode shape for a stripped array configuration involving the following: $1mm$ beam thickness, $0.35m$ beam radius,50 percent earth gravitational load, and a stroke of $0.25m$.

The first natural frequency derived for each model, as a function of stroke,beam thickness, and beam radius is shown in Figure 5.26. An example output is shown in Figure 5.12.Moreover, the results herein are only plotted for models which were subjected to a gravitational load of 50 percent of Earth's.

This level of gravity represents, more or less, normal operating conditions on Mars. In general, the order of magnitude of the extracted natural frequencies is unsurprising given the shear size and distribution of mass and structural supports. In comparison to the CTSA design, exhibited similar values albeit lower. Specifically, under a gravitational load of 40 percent of Earth's gravity, the first natural frequency was about $0.43Hz$ [6]. Apart from having fundamental design differences, the disparity between the results here and in the CTSA is in part due to the difference in loading as larger incident gravitational loading has the effect of prestressing the structure more which improves its stiffness. Another notable difference is the finite element modelling strategy performed in the CTSA versus in this thesis work.Namely, the CTSA modelled the solar array blanket using 3D surface elements. The specific element type used, whether they be membrane or shell, was not explicitly stated. The solar array strips modeled here were approximated with 3D beam elements.

Aside from models whose telescoping beams had a thickness of $1mm$, the natural frequency of the solar array structure shows a tendency to increase with increasing stroke. This makes sense as the increase in stroke increases the stress-stiffening effect in the strips. At a beam thickness of $1mm$, deflection of the beam tip upwards occurs at higher strokes, more so for smaller radii beams, which has the tendency to reduce the stress in the outer strips. In turn, this reduces their stiffness, causing a shift in first mode excitation to the extremities of the array. A shift in mode shapes with increasing stroke for various beam radii, as shown in Figure 5.13, supports this idea.

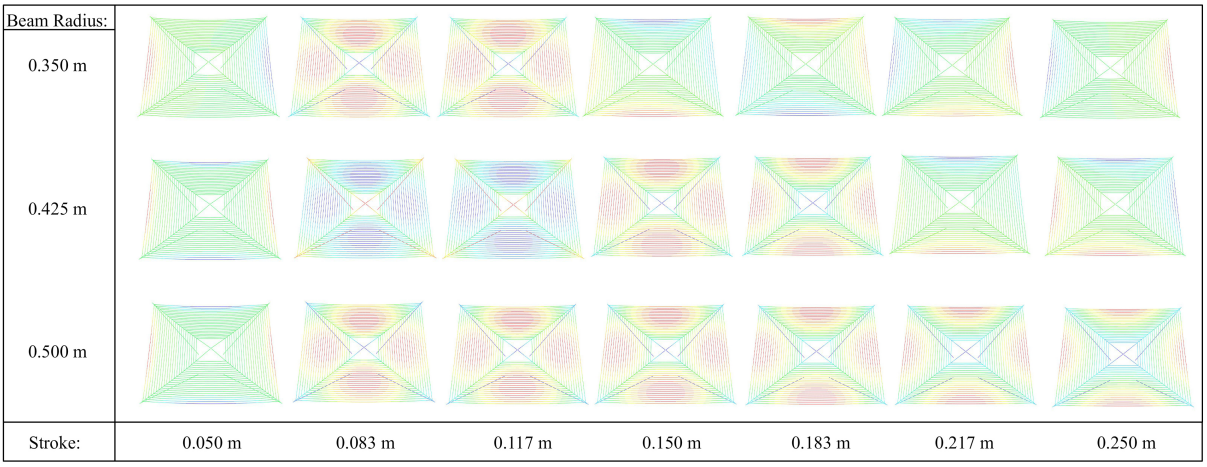


Figure 5.13: The above plot shows a comparison of mode shapes at different stroke levels between different beam radii. Each cross-section has the same thickness: $1mm$. Larger Radii, shown towards the bottom of the figure, experience less stiffness reduction in the membrane sub-assembly at higher strokes than do smaller radii beams. This is evident by the notable lag in mode shift to the outer strips as well as a lack of reduction in natural frequency shown in Figure 5.26. Note,

5.2.5. Figures

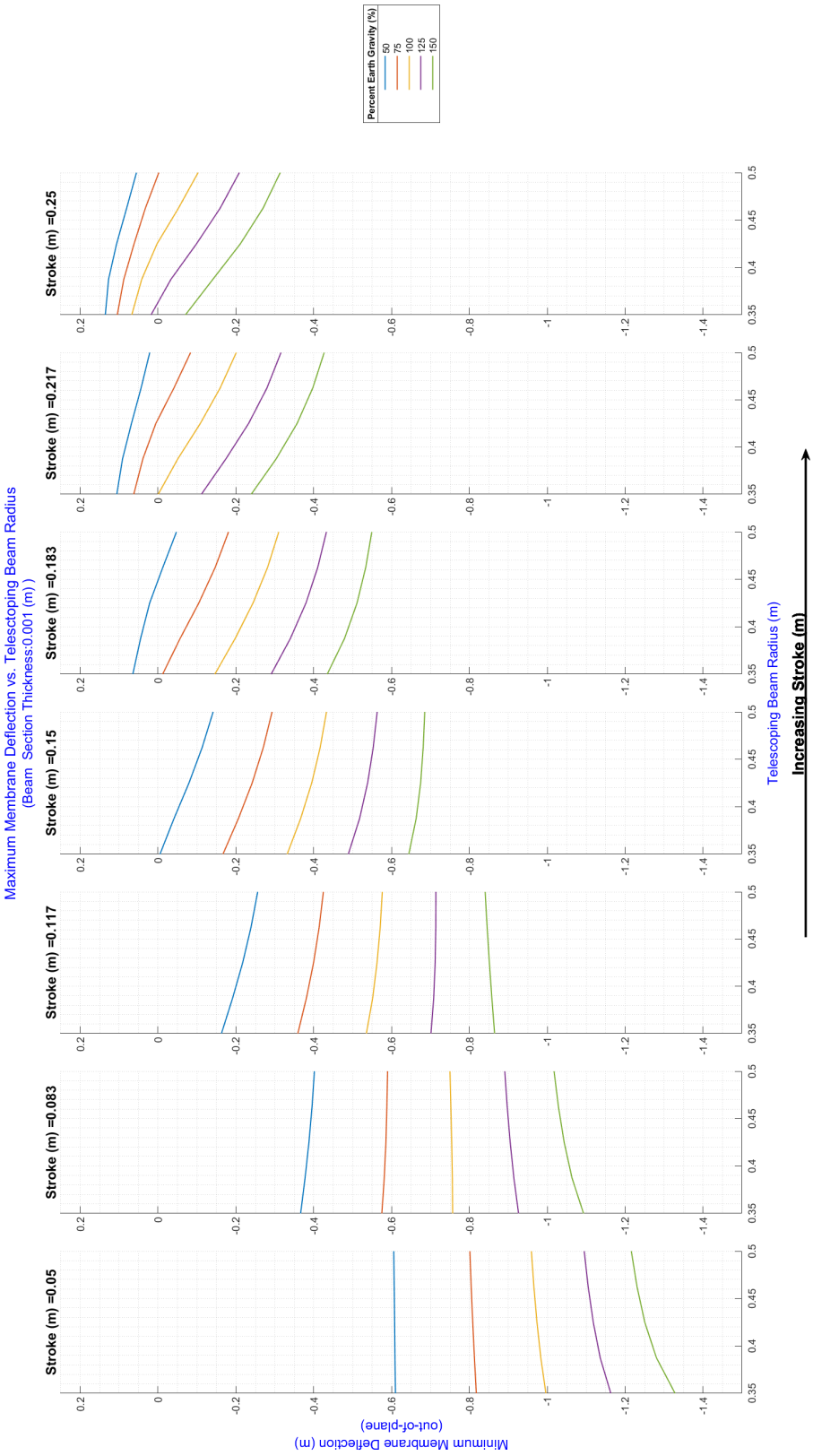


Figure 5.14: This plot shows the minimum membrane deflection for each model as a function of beam radius and level of stroke. Note, the results shown here are associated with beam thicknesses of $1mm$. Each subplot corresponds to a level of stroke where as each point on the subplots corresponds to a model tested.

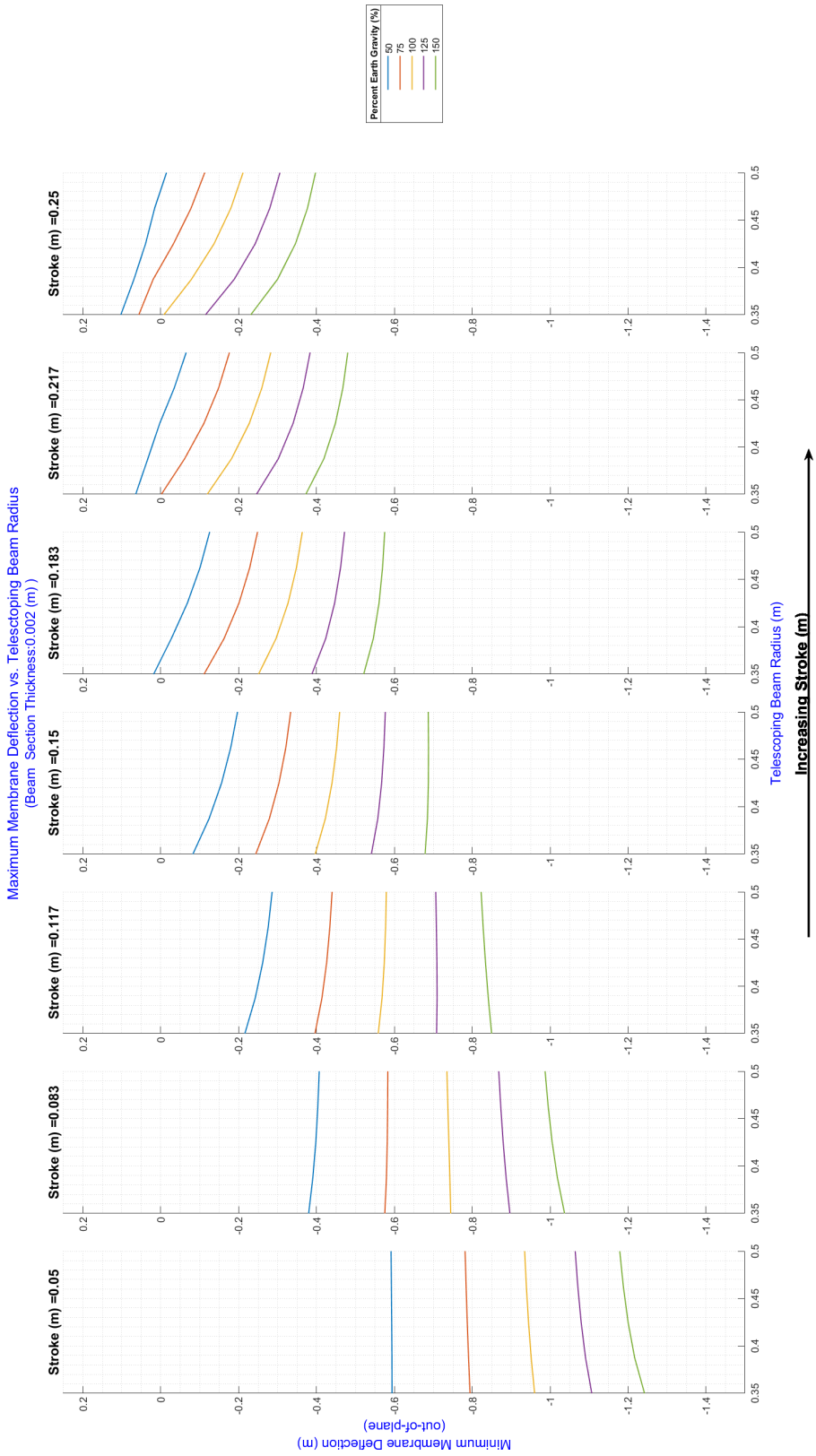


Figure 5.15: This plot shows the minimum membrane deflection for each model as a function of beam radius and level of stroke. Note, the results shown here are associated with beam thicknesses of $2mm$. Each subplot corresponds to a level of stroke where as each point on the subplots corresponds to a model tested.

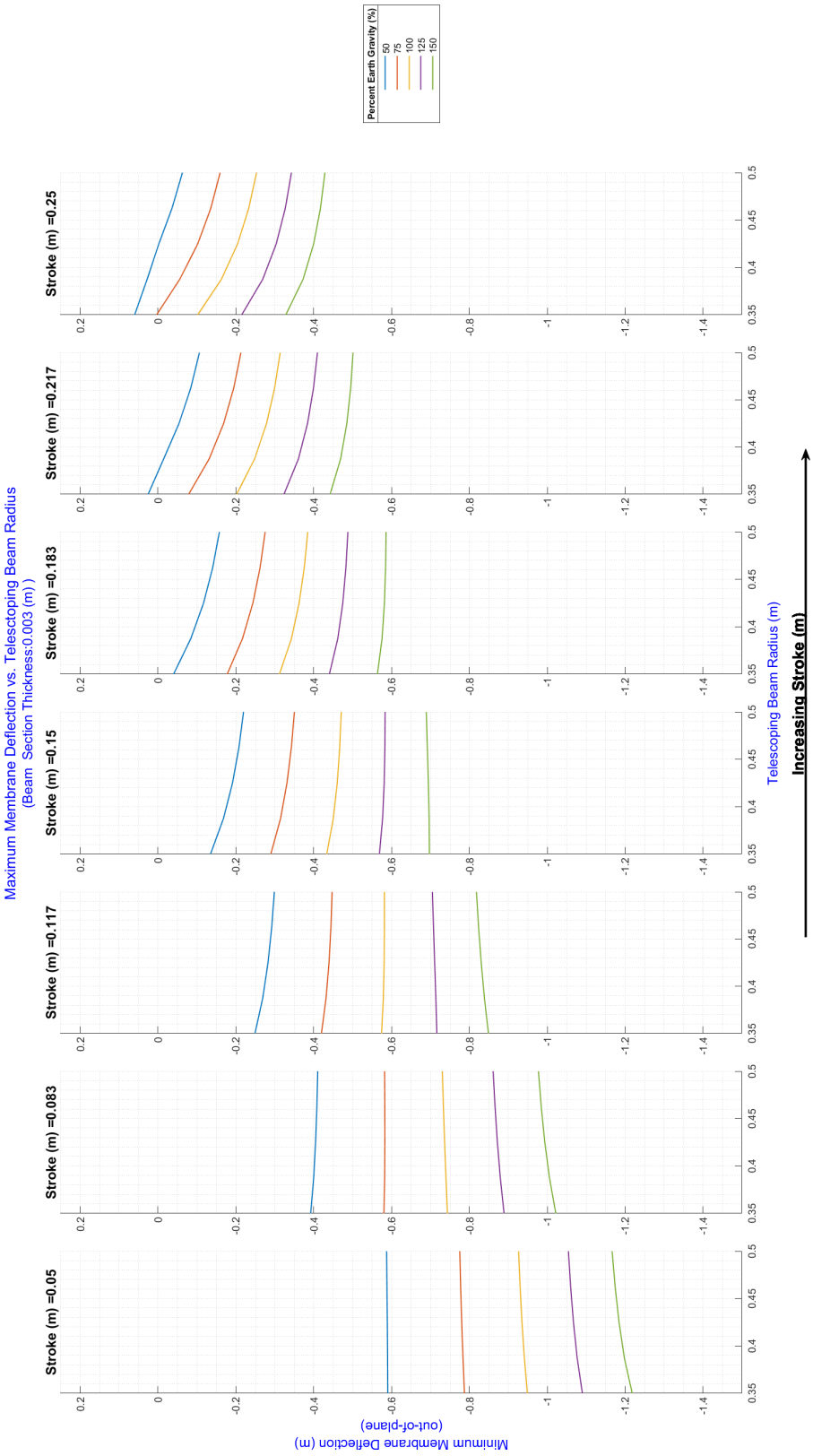


Figure 5.16: This plot shows the minimum membrane deflection for each model as a function of beam radius and level of stroke. Note, the results shown here are associated with beam thicknesses of $3mm$. Each subplot corresponds to a level of stroke where as each point on the subplots corresponds to a model tested.

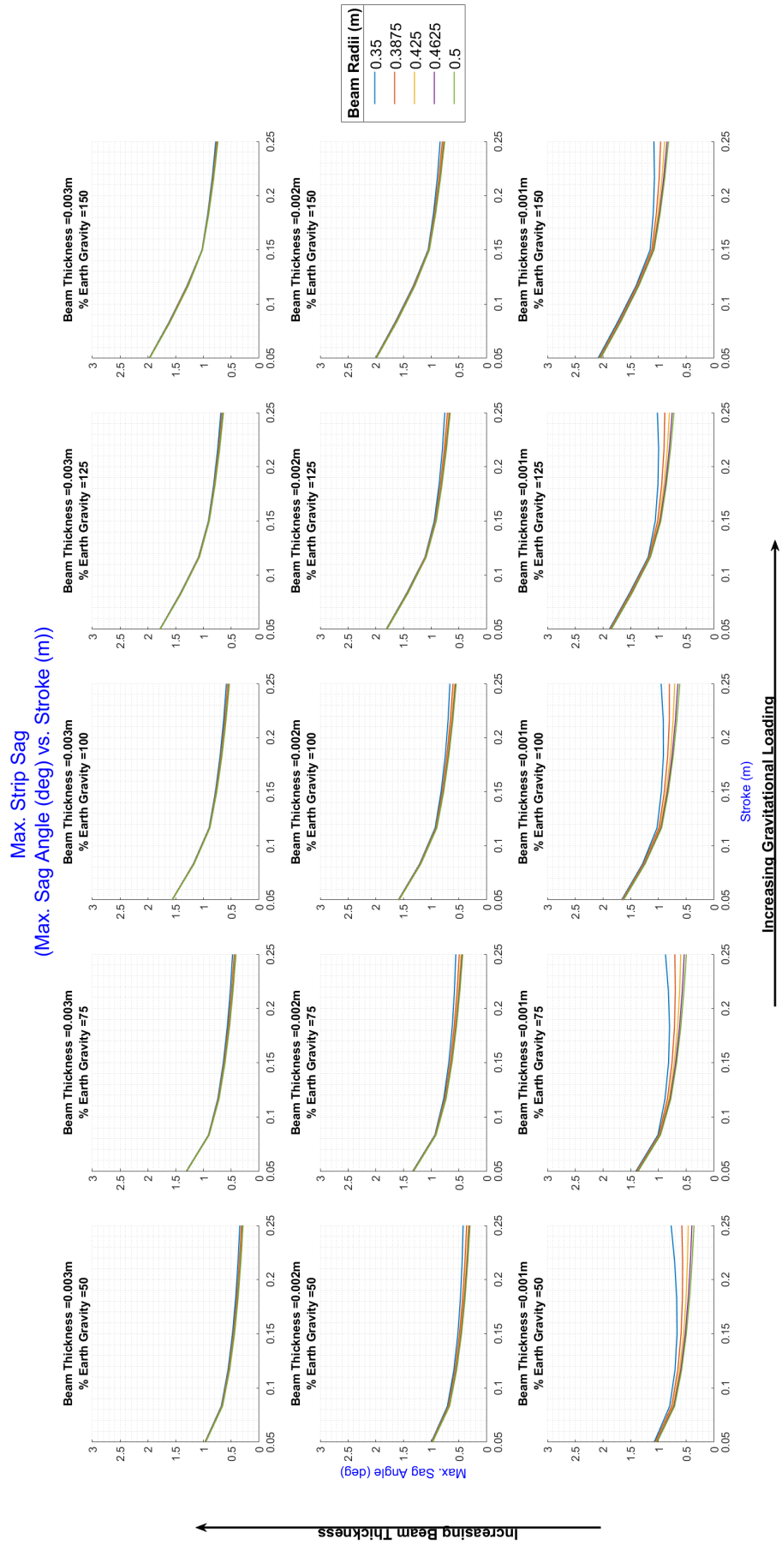


Figure 5.17: The above figure is a collection of subplots where from left to right and bottom to top corresponds to an increase in gravitational loading and beam thickness respectively. Each subplot shows how the telescoping beam radius affects the maximum sag angle in degrees in the membrane

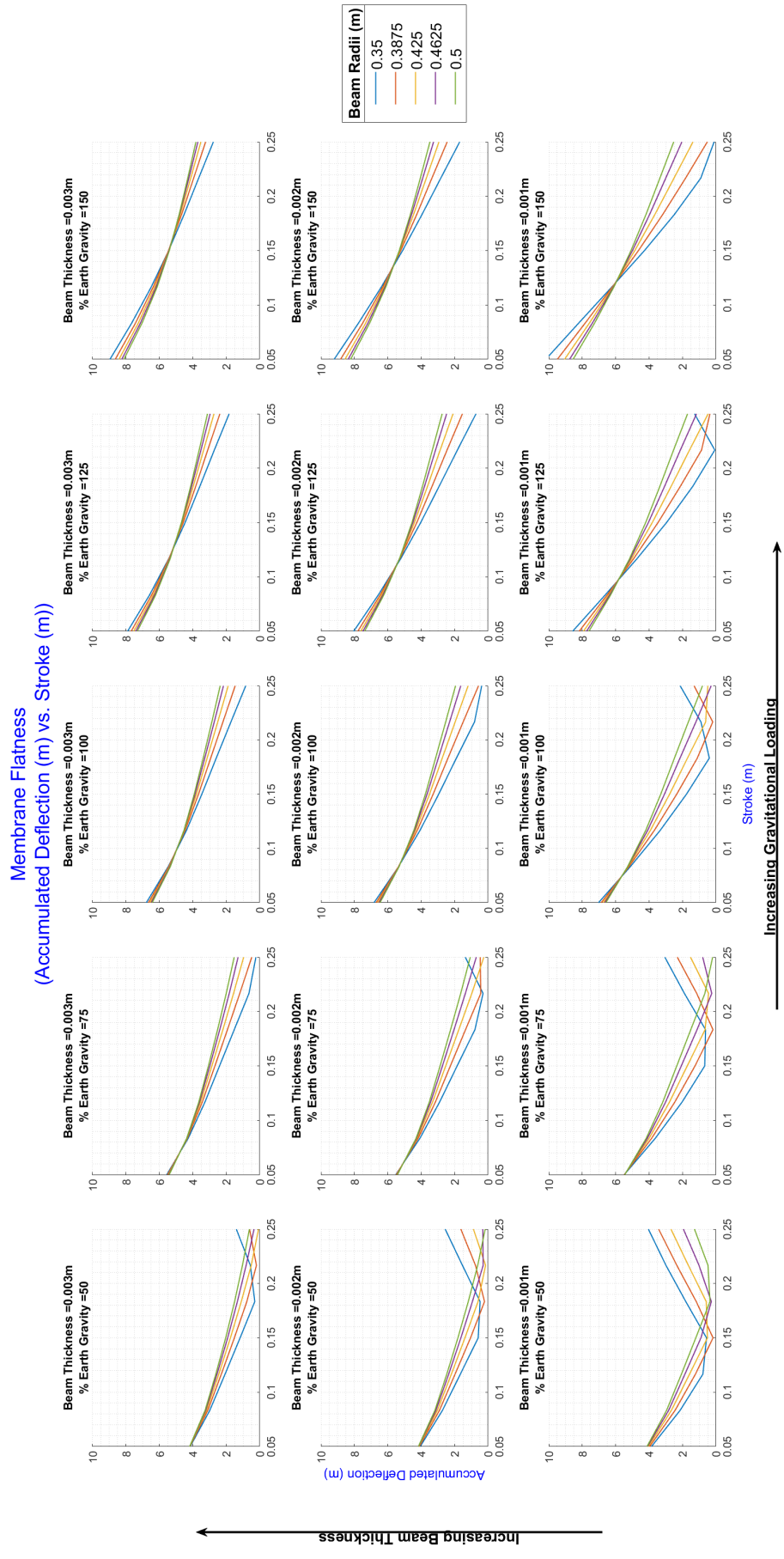


Figure 5.18: The above figure is a collection of subplots where from left to right and bottom to top corresponds to an increase in gravitational loading and beam thickness respectively. Within each subplot, the effect of stroke on the relative flatness of the membrane sub-assembly, measured in accumulated deflection (m), for each beam radius tested is shown.

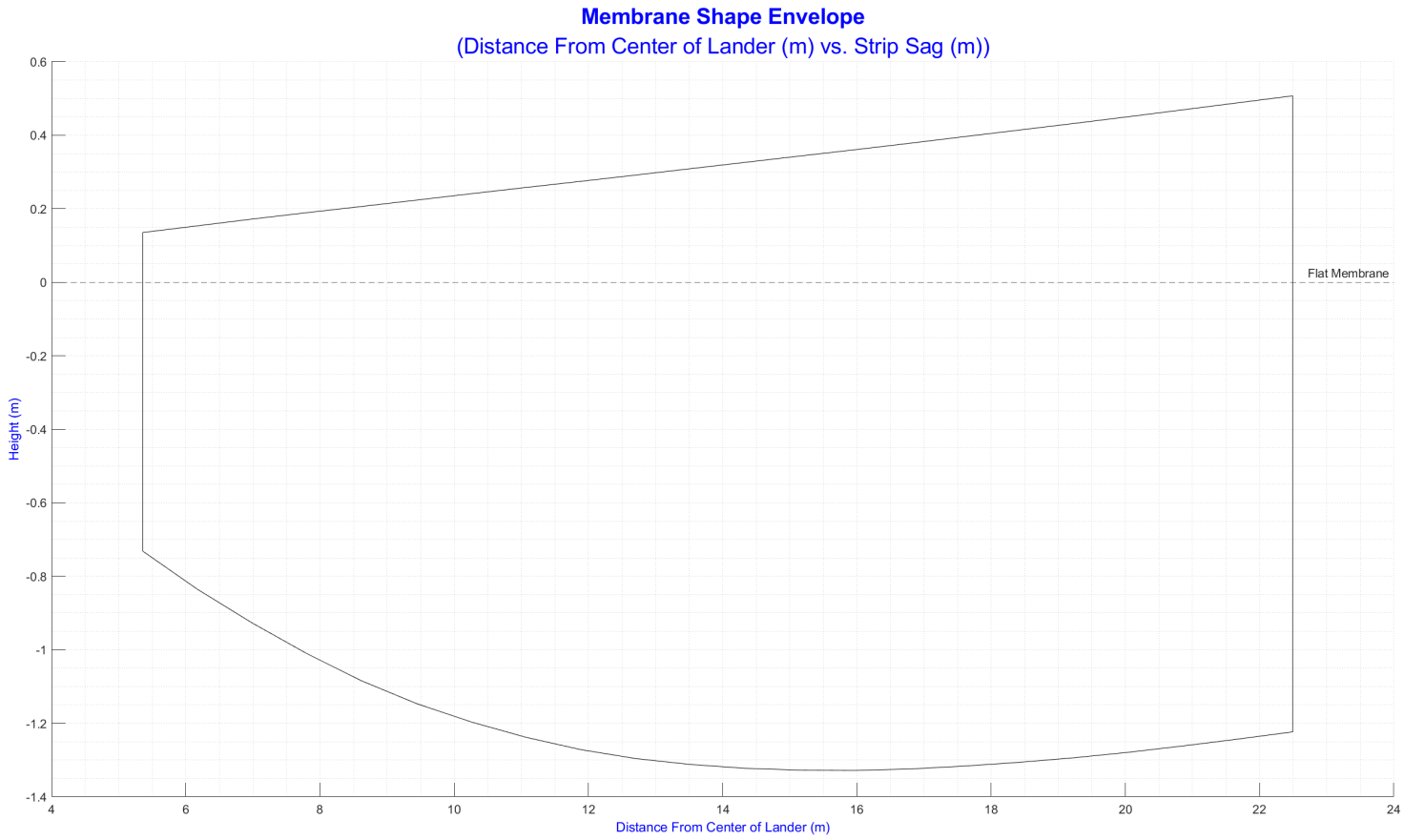


Figure 5.19: This plot shows the out-of-plane shape envelop for every model configuration under every load condition described earlier. The x-axis corresponds to the distance from the center of the lander in terms of meters and the y-axis is the out-of-plane deflection in reference to the initially flat membrane base state. Note, the membrane shapes used to create this envelope were taken from the center of one quadrant of the membrane. That is, the shapes herein represent a slice directly through the center of each strip at the end of the gravity load step.

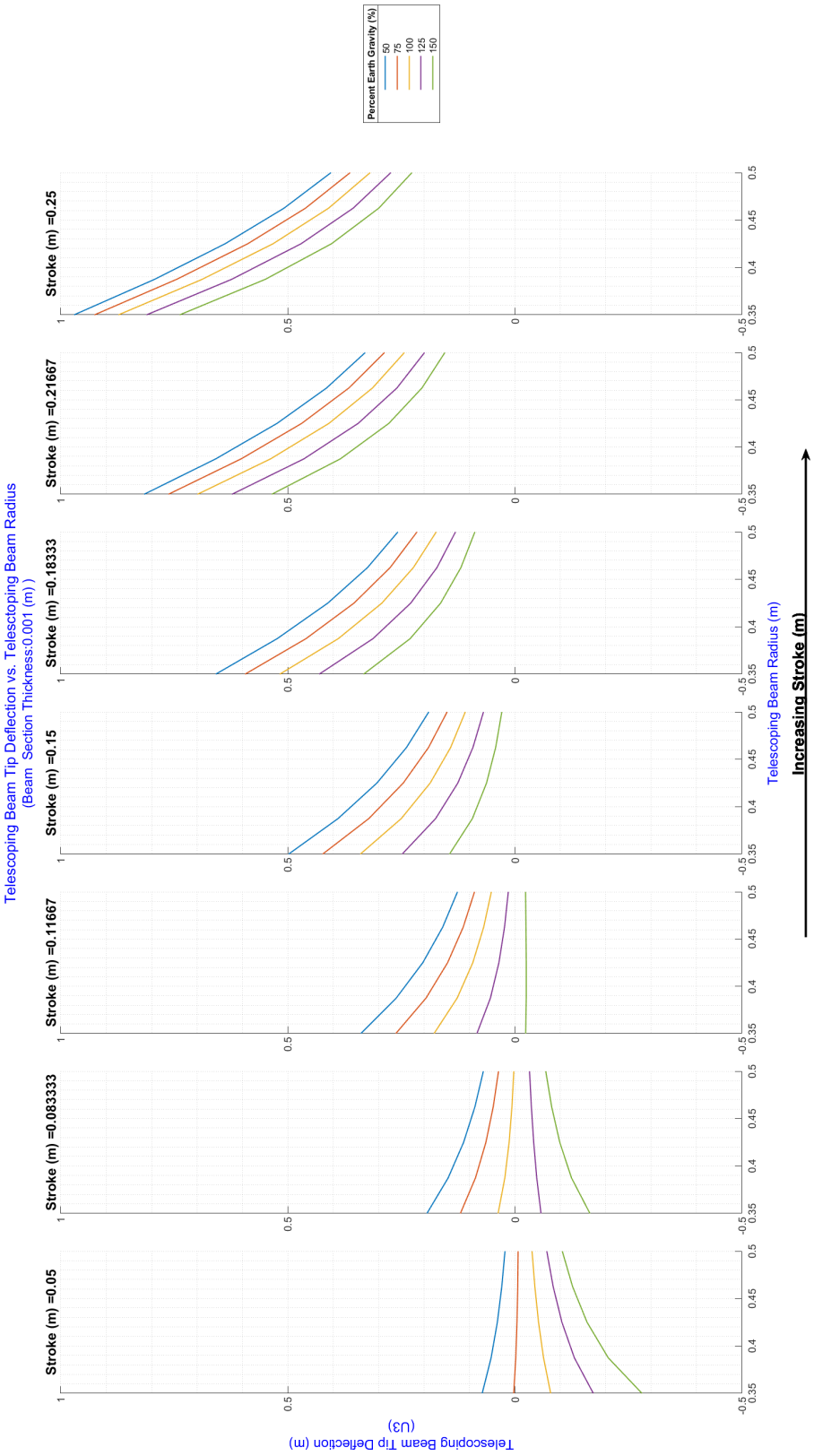


Figure 5.20: This plot shows the tip deflection in meters for different telescoping beam cross-sections at various levels of stroke and gravitational loading. Specifically, each sub-plot shows beam cross-section versus tip deflection at the five different gravitational loads tested. Furthermore, each individual sub-plot is for a different level of stroke, increasing from left to right. Lastly, this plot is for beam thicknesses of just $1mm$.

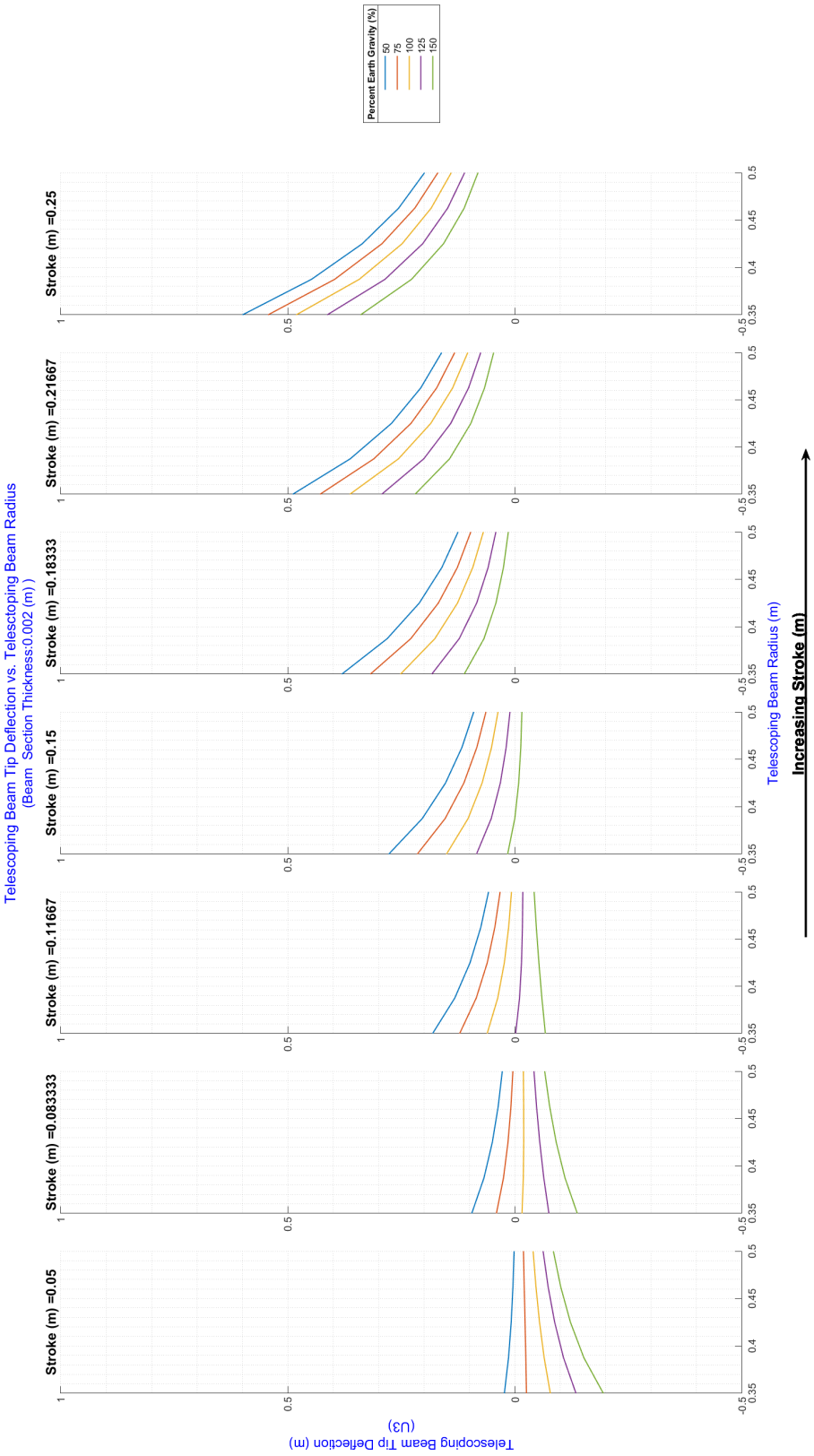


Figure 5.21: This plot shows the tip deflection in meters for different telescoping beam cross-sections at various levels of stroke and gravitational loading. Specifically, each sub-plot shows beam cross-section versus tip deflection at the five different gravitational loads tested. Furthermore, each individual sub-plot is for a different level of stroke, increasing from left to right. Lastly, this plot is for beam thicknesses of just $2mm$.

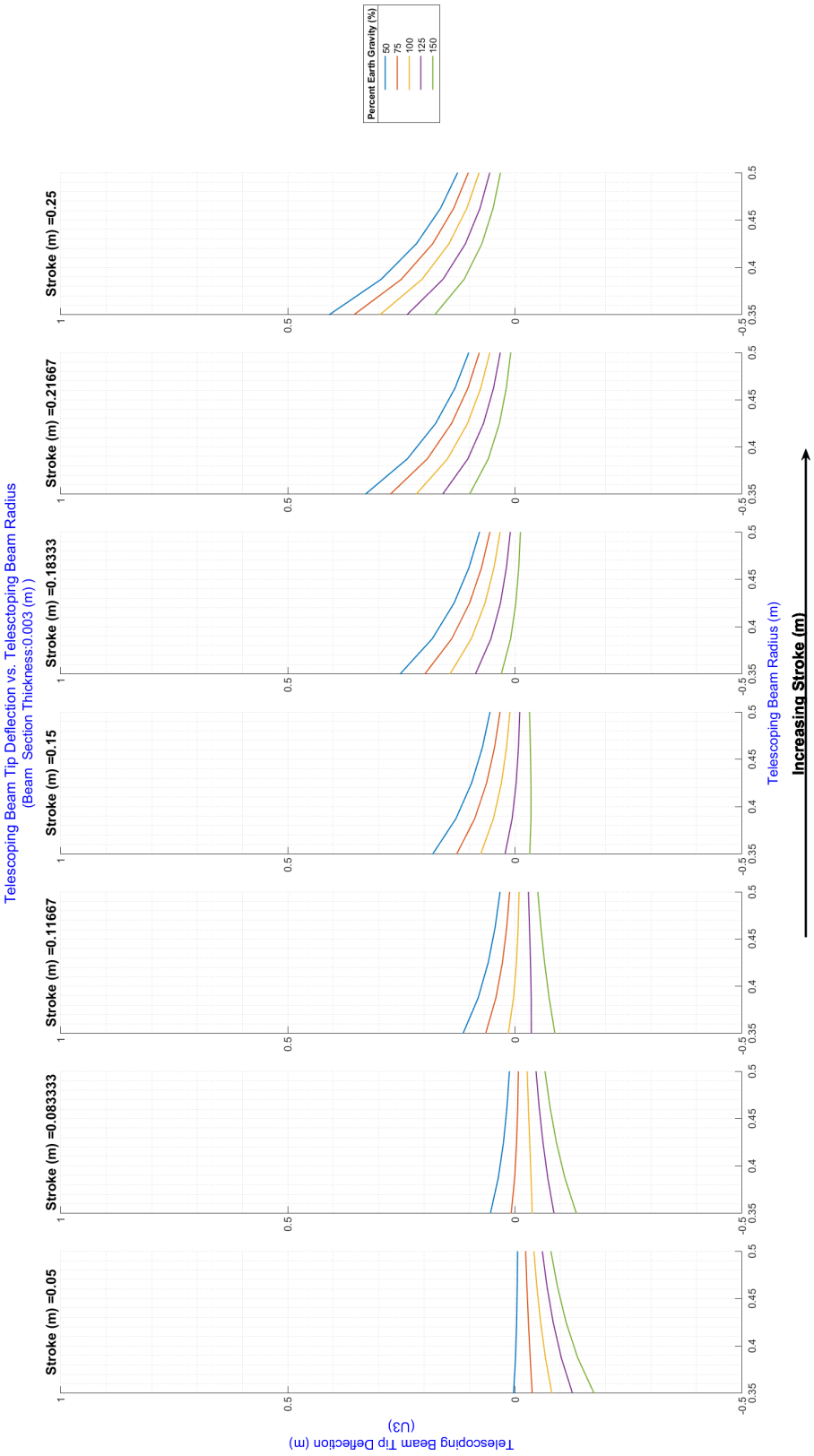


Figure 5.22: This plot shows the tip deflection in meters for different telescoping beam cross-sections at various levels of stroke and gravitational loading. Specifically, each sub-plot shows beam cross-section versus tip deflection at the five different gravitational loads tested. Furthermore, each individual sub-plot is for a different level of stroke, increasing from left to right. Lastly, this plot is for beam thicknesses of just $3mm$.

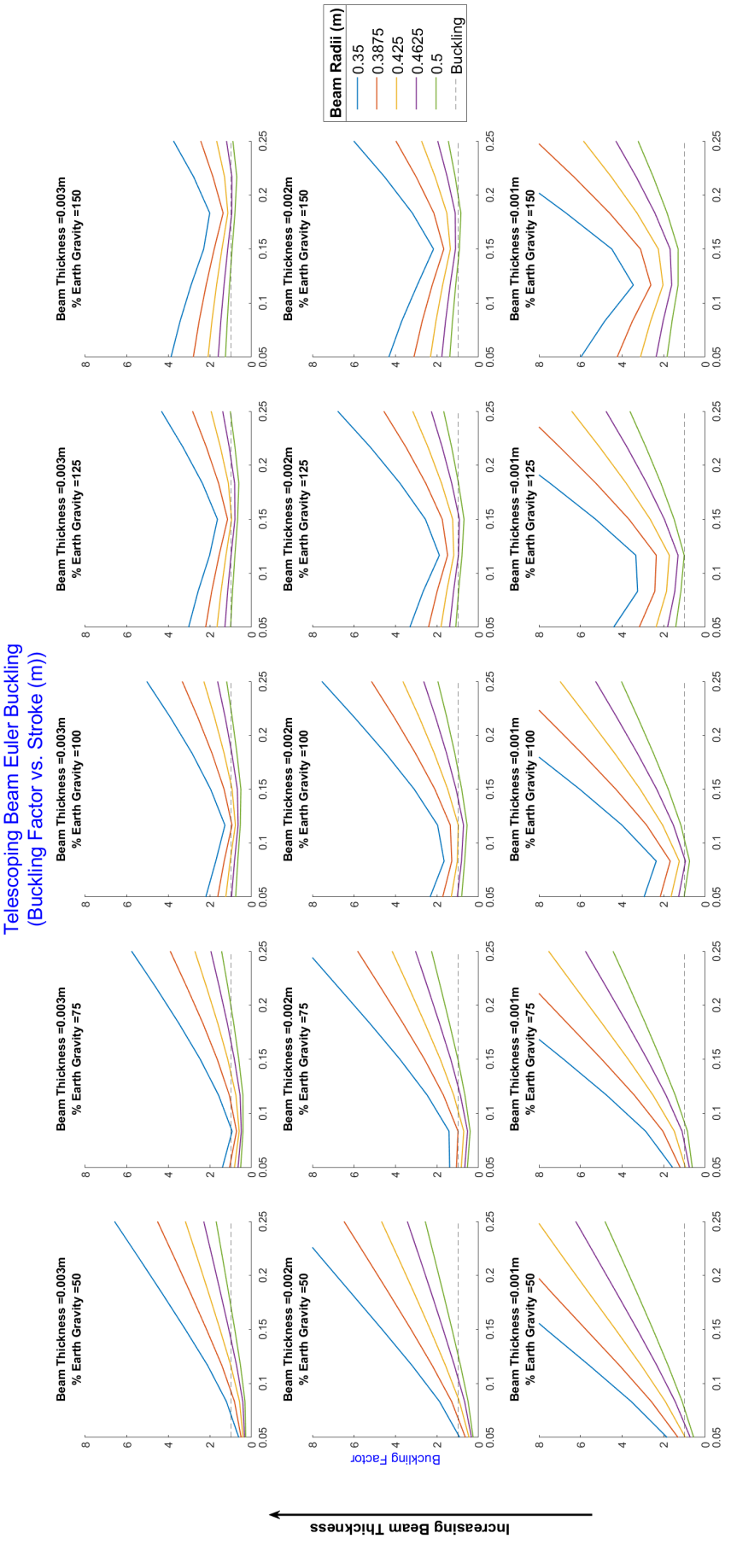


Figure 5.23: This figure is a collection of subplots where from left to right and bottom to top corresponds to an increase in gravitational loading and beam thickness respectively. With regard to contents, each sub-plot shows a model's resistance to Euler buckling as a function of stroke and telescoping beam radius. This resistance is measured as a column buckling factor which is the max beam compressive load divided by the theoretical Euler buckling for the given configuration. A ratio over one indicates failure.

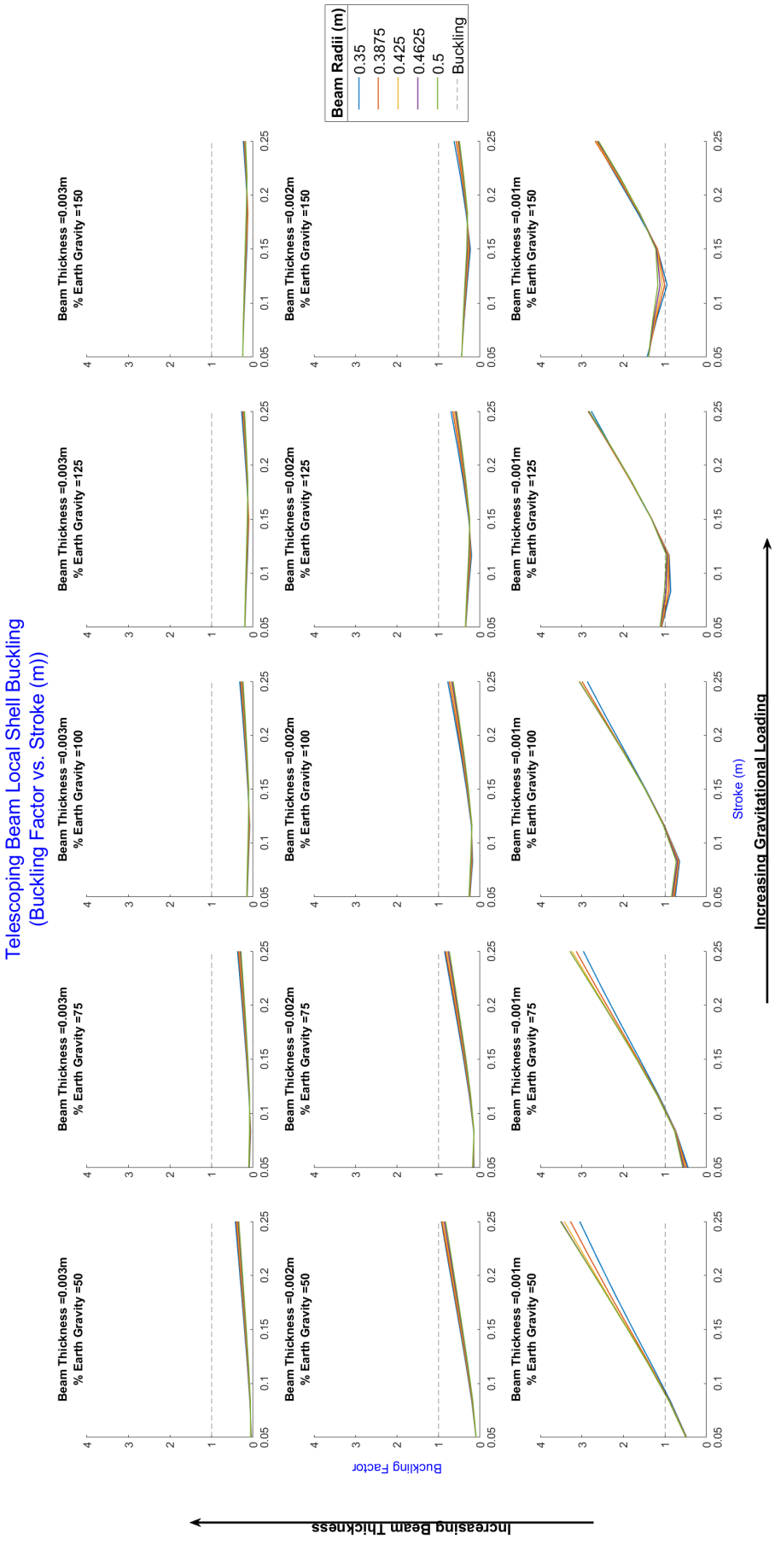


Figure 5.24: This figure is a collection of subplots where from left to right and bottom to top corresponds to an increase in gravitational loading and beam thickness respectively. With regard to contents, each sub-plot shows a model's resistance to Local Shell buckling as a function of stroke and telescoping beam radius. Similar to Figure 5.23, this resistance is measured as a buckling ratio where a ratio over one indicates buckling failure.

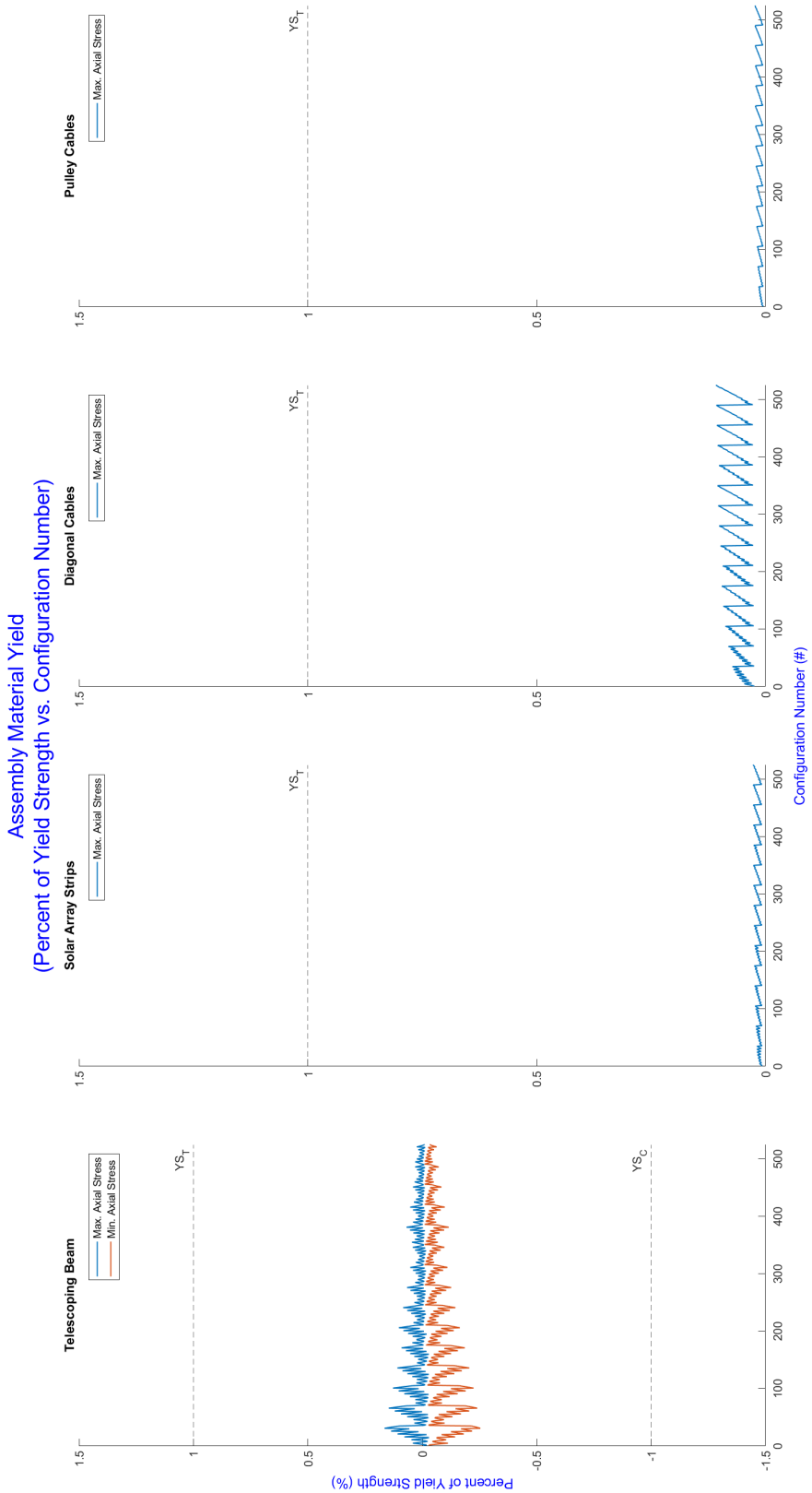


Figure 5.25: Collectively, this plot shows the relevant stress information for each of the important sub-components/sub-assemblies in the stripped solar array model.

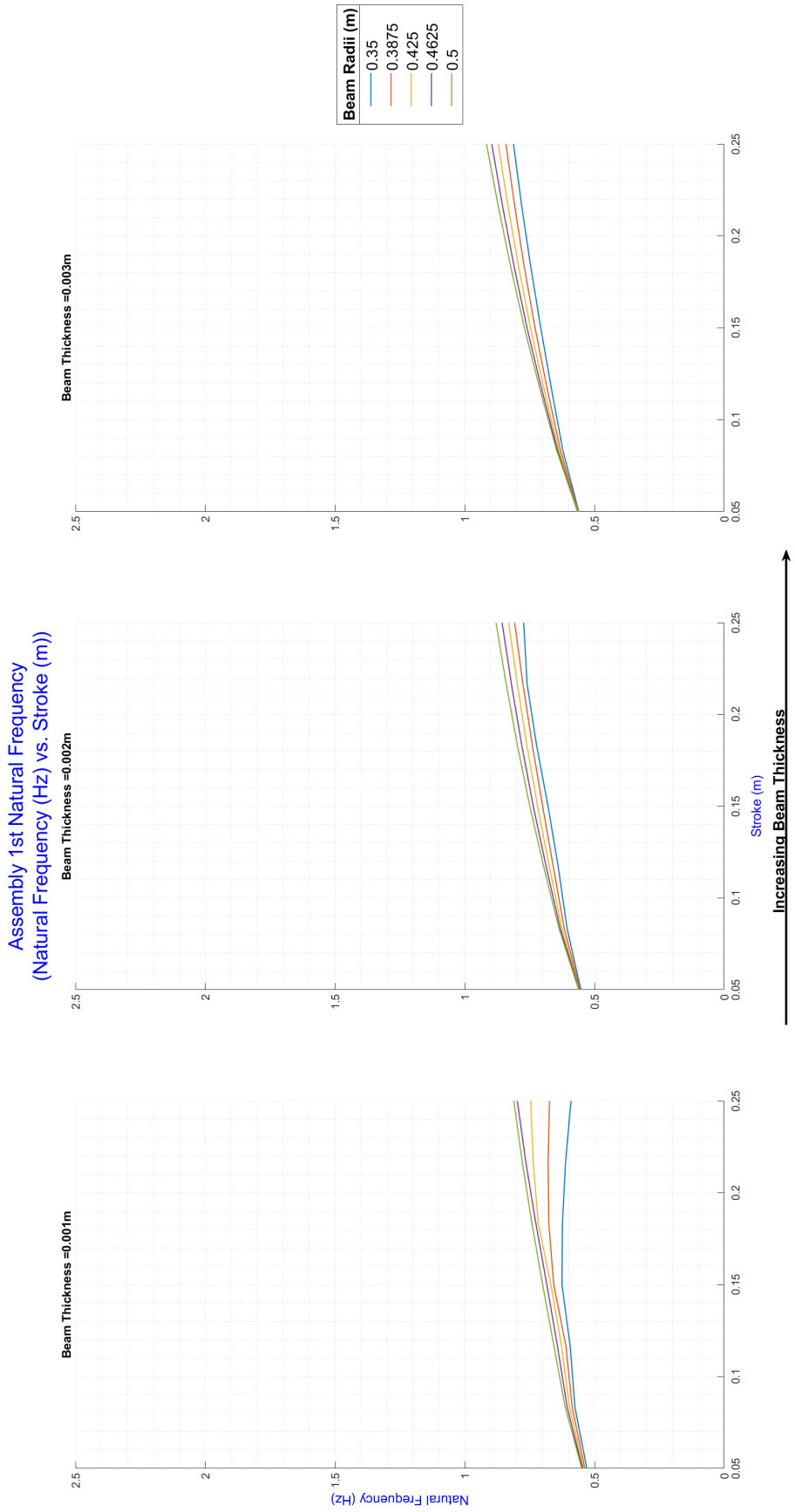


Figure 5.26: This figure is a collection of natural frequency versus level of stroke sub-plots for every beam cross-sectional radius tested. Each subplot corresponds to a different beam thickness. Lastly, these results are for a gravitational load of 50 percent of Earth's.

6

Conclusions and Future Work

6.1. Conclusions

The primary purpose of this research was to devise and characterize a large scale, static, autonomously deployable, lightweight solar array for use on the surface of Mars. Again, the goal was to conceive a competitive design to that of NASA's CTSA as well as perform additional sensitivity studies to further characterize the concept. To that end, the principal research question that was investigated in this thesis work was: What is the optimal design for a large-scale, static, autonomously deployable, lightweight Martian surface solar array for use at near equatorial latitudes? In order to properly address this primary question, a number of sub-questions were formulated and investigated which are reproduced below in conjunction with their respective summarized answers.

To start, the first portion of this thesis aimed to address the following interrelated questions: Is there a deployable solar array concept that achieves better stowage volume and mass performance than that of the CTSA developed by NASA? Should several be identified, how might each compare against one another based on these metrics in addition to concept risk? The main findings of these questions are summarized below:

- Of the handful of generated concepts, only three were believed to have the best chance at outperforming the CTSA and were subsequently selected for detailed analysis as well as cross-comparison. These concepts were the Revised Compact Telescoping Surface Array (RCTSA), the Canopy, and the Stripped Array.
- The Stripped Array concept is likely the only design of those generated in this thesis work which can realistically beat both the $1500kg$ and $10m^3$ assembly mass and stowed volume targets, respectively, set by the CTSA. At a high-level, the Stripped Array is a monolithic stripped solar (the entire solar array area is comprised in a single deployable structure). It is supported at its center by a payload as well as at its corners by four telescoping composite booms which are evenly spaced around the perimeter of said landing vehicle. A main reason for the Stripped Array's improvement in volume and mass performance is based on the reduction in deployment length and in total number of telescoping supports. Furthermore, the stripped membrane configuration in conjunction with the quad slip wrapping stowage method results in low stowed volumes by comparison to the other concepts.
- The Canopy shows promising mass performance but is likely not a better alternative in its current form to the other generated concepts nor to the CTSA mainly due to stowed array considerations. The perimeter cabling sizing severely impacts the stowage efficiency of the design as well as raises questions about the feasibility of the stowage system in general.
- The RCTSA will likely require either additional supporting architecture like deployable support legs or an increase in telescoping support mass in order to bolster its durability against local buckling.
- The Stripped Array shows remarkably good solar array area scalability where impact on the mass and volume performance of the design is minimized. This, in conjunction with its lack of lander

clearance sensitivity, results in a concept which outperforms other concepts in terms of requirement risk.
- A monolithic solar array concept is riskier on the basis of lack of system redundancy and will likely present challenges in regard to testing and qualification problems in comparison to segmented arrays like the RCTSA.

In the latter part of this thesis work, the goal was to investigate the following questions with respect to the down-selected concept: How does the pre-tensioning scheme perform in response to changes in sizing of various components of the solar array membrane sub-assembly? How does the total array area and applied membrane pre-tension load affect the distribution of pre-stress throughout the membrane sub-assembly? Lastly, how does level of pretension of the solar array membrane, magnitude of gravitational loading, and alterations to the supporting architecture's cross-sectional sizing and assumed support conditions affect assembly stresses, deflections, and natural frequencies? Again, the main findings of these questions are summarized below:

- For a $1000m^2$ membrane with 20 strips per quadrant under conditions of the same applied pre-tension load, the distribution of pre-stress throughout the stripped array equalizes and lowers with increasing diagonal cable radius for all assumed strip reinforcement areal masses.To that end, a diagonal cable radius of $2.5mm$ is likely too small to articulate sufficient pre-stress in the inner strips for large strip reinforcement areal masses.
- Assuming a diagonal cable radius of $5mm$ with a strip mesh reinforcement areal mass of $66\frac{g}{m^2}$, the magnitude of pre-stress throughout the stripped membrane parabolically lowers with increasing total array area for all applied pre-tension loads tested. Moreover, the general tendency is for the disparity in pre-stress between the innermost strip and outer strip to widen.
- Unsurprisingly, the maximum out-of-plane membrane deflection increases with increasing assumed level of gravitational loading for all combinations of telescoping support cross-section radius, cross-section thickness, and level of applied pre-tension. One of the principal findings, however, is that reducing the supporting architecture's support conditions, namely the removal of additional guy-wire supports and deployable support legs, results in non-ideal telescoping support behavior. Specifically, the telescoping support tips' displace significantly depending on the magnitude of gravity, applied membrane pre-tension force, and the telescoping support cross-sectional parameters. This movement has the impact of directly affecting the out-of-plane displacements of the stripped membrane. It was shown that this level of this behavior can be retarded slightly through increasing the telescoping support cross-sectional thickness.
- Under the reduced support condition, a telescoping section thickness of $1mm$ is likely too low to prevent local buckling the telescoping supports. Furthermore, while the estimates were conservative, the reduced support condition will likely result in column buckling. This underscores the importance of including the original support configuration
- Individual, relative strip sag appears to be primarily dependent on the applied pre-tension rather than telescoping support characteristics or level of gravitational load. Conversely, relative flatness of the entire stripped array membrane does change based on all of the aforementioned parameters, driven by lack of stability in the telescoping support tips.
- For all combinations of telescoping support cross-section thickness and radii, level of gravity, and level of pre-tension, material failure is not a risk.
- Unsurprisingly, all tested configurations exhibited a rather low first natural frequency, consequential of the shear size and distribution of mass and structural supports. It was shown that telescoping support tip deflection has the effect of reducing the pre-stress in the outer strips of the membrane thus lowering the resultant first natural frequency.

6.2. Future Work

Building upon this thesis work, the main recommendations for future work/trade studies are as follows:

- Construct another Stripped Array assembly finite element model and evaluate the concept utilizing the originally devised telescoping support conditions. That is, include in the model the deployable legs and guy-wire tip supports and repeat the same analysis.

- Investigate how the concept can adapt and perform should the lander be inclined rather than horizontal as a result of a sloped landing surface.
- Evaluate Stripped Array performance with respect to variations in lander diameter.
- Build and evaluate a real life model of the Stripped Array assembly to validate the constructed finite element model.
- Investigate the feasibility of wind load relief on the Stripped Array membrane through integration of feathering.
- Perform wind tunnel testing of the structure to understand its behavior during simulated Mars loads.

References

[1] “National Aeronautics and Space Administration Authorization Act of 2010”. In: 2010. Chap. Pub. L. No. 111-267. URL: https://www.govinfo.gov/app/details/PLAW-111publ267.

[2] *NASA’s Lunar Exploration Program Overview*. Rev. 1. National Aeronautics and Space Administration (NASA). Sept. 2020.

[3] “National Aeronautics and Space Administration Authorization Act of 2020”. In: 2010. Chap. H.R.5666. URL: https://www.congress.gov/bill/116th-congress/house-bill/5666.

[4] Loura Hall. *Kilopower*. Dec. 2017. URL: https://www.nasa.gov/directorates/spacetech/kilopower.

[5] Jennifer Harbaugh. *Fission Surface Power*. May 2021. URL: https://www.nasa.gov/mission_pages/tdm/fission-surface-power/index.html.

[6] Richard Pappa et al. *Solar Arrays with Storage (SAWS) Seedling Study*. Tech. rep. DOE-SLC-6903-1. Hampton,VA: NASA, Sept. 2017.

[7] *Program schedule and selection announcements: nasa sbir and sttr program*. URL: https://sbir.nasa.gov/prg_sched_anncmnt.

[8] Michelle A. Rucker et al. “Solar vs. Fission Surface Power for Mars”. In: *AIAA SPACE 2016*. DOI: 10.2514/6.2016-5452. eprint: https://arc.aiaa.org/doi/pdf/10.2514/6.2016-5452. URL: https://arc.aiaa.org/doi/abs/10.2514/6.2016-5452.

[9] Richard Pappa et al. “Compact Telescoping Surface Array for Mars Solar Power”. In: Jan. 2018. DOI: 10.2514/6.2018-1944.

[10] Ben Bussey and Stephen Hoffman. “Human Mars landing site and impacts on Mars surface operations”. In: Mar. 2016, pp. 1–21. DOI: 10.1109/AERO.2016.7500775.

[11] “Environments, needs and opportunities for future space photovoltaic power generation: A review”. In: *Applied Energy* 290 (2021), p. 116757. ISSN: 0306-2619. DOI: https://doi.org/10.1016/j.apenergy.2021.116757.

[12] “Photovoltaic arrays for Martian surface power”. In: *Acta Astronautica* 30 (1993), pp. 127–142. ISSN: 0094-5765. DOI: https://doi.org/10.1016/0094-5765(93)90105-6.

[13] Geoffrey Landis and Dan Hyatt. “The Solar Spectrum on the Martian Surface and its Effect on Photovoltaic Performance”. In: 2 (Jan. 2006). DOI: 10.1109/WCPEC.2006.279888.

[14] Ralph D. Lorenz et al. “Lander and rover histories of dust accumulation on and removal from solar arrays on Mars”. In: *Planetary and Space Science* 207 (2021), p. 105337. ISSN: 0032-0633. DOI: https://doi.org/10.1016/j.pss.2021.105337. URL: https://www.sciencedirect.com/science/article/pii/S0032063321001768.

[15] John Callas, Matthew P Golombek, and Abigail A. Fraeman. *Mars Exploration Rover Opportunity End of Mission Report*. Tech. rep. JPL Publication 19-10. National Aeronautics and Space Administration: Jet Propulsion Laboratory, Oct. 2019.

[16] Mark T. Lemmon et al. “Dust aerosol, clouds, and the atmospheric optical depth record over 5 Mars years of the Mars Exploration Rover mission”. In: 251 (May 2015), pp. 96–111. DOI: 10.1016/j.icarus.2014.03.029. arXiv: 1403.4234 [astro-ph.EP].

[17] Scott Guzewich et al. “Martian Dust Particle Size During the 2018 Planet-Encircling Dust Storm as Measured by the Curiosity Rover”. In: *EPSC-DPS Joint Meeting 2019*. Vol. 2019. Sept. 2019, EPSC-DPS2019-500, EPSC-DPS2019–500.

[18] Geoffry A. Landis et al. “Dust and Sand Deposition on the MER Solar Arrays as viewed by the Microscopic Imager”. In: *Lunar and Planetary Science* 37 (2006).

[19] German M. Martinez et al. “The Modern Near-Surface Martian Climate: A Review of In-situ Meteorological Data from Viking to Curiosity”. In: *Space Science Reviews* 212 (Oct. 2017). DOI: 10.1007/s11214-017-0360-x.

[20] John Callas. *Mars Exploration Rover Spirit End of Mission Report*. Tech. rep. JPL Publication 16-2. National Aeronautics and Space Administration: Jet Propulsion Laboratory, Dec. 2015.

[21] Giulia Fenci and Neil Currie. “Deployable structures classification: A review”. In: *International Journal of Space Structures* 32 (June 2017), pp. 112–130. DOI: 10.1177/0266351117711290.

[22] M. Alexander. *Mars Transportation Environment Definition Document*. Tech. rep. NASA / TM–2001-210935. National Aeronautics and Space Administration: Marshall Space Flight Center, Mar. 2001.

[23] *Starship Users Guide*. Revision 1.0. SpaceX. Mar. 2020.

[24] Sharon Jefferies et al. “Impacts of launch vehicle fairing size on human exploration architectures”. In: Mar. 2017, pp. 1–16. DOI: 10.1109/AERO.2017.7943833.

[25] Bill Hill and Stephen Creech. “NASA’s Space Launch System: A Revolutionary Capability for Science”. In: NASA NTRS, 2014.

[26] *Ariane 5e*. Revision 2.0. Arianespace. Oct. 2016.

[27] *Falcon’s User’s Guide*. Revision 1.0. SpaceX. Sept. 2021.

[28] *Vega Users Guide*. Revision 0.0. Arianespace. Apr. 2014.

[29] *Soyuz Users Guide*. Revision 0.0. Arianespace. Mar. 2012.

[30] David Hoffman et al. “Concept Design of High Power Solar Electric Propulsion Vehicles for Human Exploration”. In: (Dec. 2011).

[31] Brian Spence et al. “International Space Station (ISS) Roll-Out Solar Array (ROSA) Spaceflight Experiment Mission and Results”. In: June 2018, pp. 3522–3529. DOI: 10.1109/PVSC.2018.8548030.

[32] Brian R. Spence and Stephen F. White. *Directionally controlled elastically deployable roll-out solar array*. US 8,683,755 B1, Apr. 2014.

[33] *Solar Arrays- Roll Out Solar Array (ROSA)*. V6. Redwire Corporation. Nov. 2021.

[34] Rao Surampudi et al. *Solar Power Technologies for Future Planetary Science Missions*. Tech. rep. JPL D-101316. National Aeronautics and Space Administration: Jet Propulsion Laboratory, Dec. 2017.

[35] David M. Murphy et al. “UltraFlex and MegaFlex - Development of highly scalable solar power”. In: *2015 IEEE 42nd Photovoltaic Specialist Conference (PVSC)* (2015), pp. 1–8.

[36] Joel Schwartz et al. “Solar Array Technologies for Planetary Science and Astrobiology Missions”. In: *Bulletin of the AAS* 53 (Mar. 2021). DOI: 10.3847/25c2cfeb.08b22cde.

[37] Thomas Pace. “Lessons Learned During the Development, Qualification, and Production of the MM Solar Array”. In: 2020.

[38] John L. Gibb. *Multi-Mission Modular Array*. US 10,546,967 B2, Jan. 2020.

[39] Martin Mikulas et al. “Telescoping Solar Array Concept for Achieving High Packaging Efficiency”. In: Jan. 2015. DOI: 10.2514/6.2015-1398.

[40] *Ultraflex Solar Array: Enabling Light Weight Low Volume Technology*. V1. Northrop Grumman. 2007.

[41] R. Pappa. “In Mars Surface Solar Array Structures: Recent SBIR Contracts”. In: 25th Photovoltaic Research and Technology Conference, 2018.

[42] Richard Pappa and Tom Kerslake. “Mars Surface Solar Arrays: Part 2 (Power Performance)”. In: NASA NTRS, 2017.

[43] *Relocatable 10 kW Solar Array for Lunar South Pole Missions*. NASA/TM-20210011743. Rev. 1. National Aeronautics and Space Administration (NASA). Mar. 2021.

[44] Marco Straubel, Martin Hillebrandt, and Christian Hühne. “Evaluation of Different Architectural Concepts for Huge Deployable Solar Arrays for Electric Propelled Space Crafts”. In: Sept. 2016.

[45] Carlton L. Foster et al. “Solar-array-induced disturbance of the Hubble Space Telescope pointing system”. In: *Journal of Spacecraft and Rockets* 32.4 (1995), pp. 634–644. DOI: 10.2514/3.26664.

[46] S. Guest and Sergio Pellegrino. “Inextensional wrapping of flat membranes”. In: *Proceedings of International Seminar Structure Morphology* (Jan. 1992).

[47] Spencer Magleby et al. “Accommodating Thickness in Origami-Based Deployable Arrays”. In: *Journal of Mechanical Design* 135 (Nov. 2013). DOI: 10.1115/1.4025372.

[48] Koryo Miura and Sergio Pellegrino. *Forms and Concepts for Lightweight Structures*. Cambridge University Press, 2020. DOI: 10.1017/9781139048569.

[49] Eleftherios Gdoutos et al. “Ultralight Deployable Space Structure Prototype”. In: Jan. 2020. DOI: 10.2514/6.2020-0692.

[50] Manan Arya, Nicolas Lee, and Sergio Pellegrino. “Wrapping Thick Membranes with Slipping Folds”. In: *2nd AIAA Spacecraft Structures Conference*. DOI: 10.2514/6.2015-0682.

[51] Nicolas Lee and Sigrid Close. “Curved pleat folding for smooth wrapping”. In: *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 469 (2013).

[52] W. Belvin et al. “Advanced Deployable Structural Systems for Small Satellites”. In: Sept. 2016.

[53] Omer Soykasap et al. “Tape Spring Large Deployable Antenna”. In: *47th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*. DOI: 10.2514/6.2006-1601. eprint: https://arc.aiaa.org/doi/pdf/10.2514/6.2006-1601. URL: https://arc.aiaa.org/doi/abs/10.2514/6.2006-1601.

[54] Nasir Adeli et al. “Deployment System for the CubeSail nano-Solar Sail Mission”. In: Aug. 2010.

[55] Huina Mao et al. “Deployment of Bistable Self-Deployable Tape Spring Booms Using a Gravity Offloading System”. In: *Journal of Aerospace Engineering* 30 (Jan. 2017), p. 04017007. DOI: 10.1061/(asce)as.1943-5525.0000709.

[56] *Packaging and deployment of solar arrays and antennas - Sergio Pellegrino*. URL: http://www.pellegrino.caltech.edu/packaging-and-deployment-of-solar-arrays-and-antennas.

[57] Hagen R. Mauch. *Deployable Lattice Column*. US 3,486,279, 1967.

[58] Craig G. Huntington. *7.6.2 Cable Tensioning Mechanisms*. 2013. URL: https://app.knovel.com/hotlink/khtml/id:kt00BT6MR9/tensile-fabric-structures/cable-tensioning-mechanisms.

[59] Stephen White et al. *Solar Arrays*. US 9.214,892 B2, Dec. 2015.

[60] Joel Schwartz et al. “Dust Mitigation for Lunar Surface Arrays”. In: (July 2020).

[61] Diego Fernandez, R Cabas, and Luis Moreno. “Dust Wiper Mechanism for Operation in Mars”. In: *Proceedings of SPIE - The International Society for Optical Engineering* 2007 (Aug. 2007). DOI: 10.1117/12.716924.

[62] Di Chen and Jun-ru Wu. “Dislodgement and removal of dust-particles from a surface by a technique combining acoustic standing wave and airflow”. In: *The Journal of the Acoustical Society of America* 127 (Jan. 2010), pp. 45–50. DOI: 10.1121/1.3268507.

[63] Carlos Calle et al. “Electrodynamic Dust Shields on the International Space Station: Exposure to the space environment”. In: *Journal of Electrostatics* 71 (June 2013), pp. 257–259. DOI: 10.1016/j.elstat.2012.10.009.

[64] J Singh. “Dust adhesion on Viking lander camera window”. In: *NASA STI/Recon Technical Report N.* (1978).

[65] Benjamin Farr et al. “Dust mitigation technology for lunar exploration utilizing an electron beam”. In: *Acta Astronautica* 177 (2020), pp. 405–409.

[66] *DUST MITIGATION GAP ASSESSMENT REPORT*. Rev. 1. National Aeronautics and Space Administration (NASA). Feb. 2016.

[67] Brian R. Spence et al. *Integrated Modular Photovoltaic Blanket Assembly For Space Solar Array*. US 9,620,658 B1, Apr. 2017.

[68] Sandra Tomczak et al. “Space Survivability of Main-Chain and Side-Chain POSS-Kapton Polyimides”. In: *AIP Conference Proceedings* 1087 (May 2008), p. 15. DOI: 10.1063/1.3076863.

[69] Kristine Larson. “Ultraviolet Testing of Space Suit Materials for Mars”. In: 2017.

[70] Tara Polsgrove et al. “Human Mars lander design for NASA’s evolvable mars campaign”. In: Mar. 2016, pp. 1–15. DOI: 10.1109/AERO.2016.7500778.

[71] Tara P. Polsgrove et al. “Update to Mars Ascent Vehicle Design for Human Exploration”. In: *2019 IEEE Aerospace Conference*. 2019, pp. 1–15. DOI: 10.1109/AERO.2019.8741709.

[72] Bret G. Drake, Stephen J. Hoffman, and David Beaty. “Human exploration of Mars, Design Reference Architecture 5.0”. In: *2010 IEEE Aerospace Conference* (2010), pp. 1–24.

[73] Michelle Rucker et al. “NASA’s Strategic Analysis Cycle 2021 (SAC21) Human Mars Architecture”. In: Mar. 2022, pp. 1–10.

[74] Polsgrove, Tara and Chapman, Jack and Sutherlin, Steve and Taylor, Brian and Robertson, Ed and Studak, Bill and Vitalpur, Sharada and Fabisinski, Leo and Lee, Allan and Collins, Tim and Cianciolo, Alicia and Samareh, Jamshid and Rakow, Glenn. *Human Mars lander design for NASA’s evolvable mars campaign*. Technical Presentation. 2016.

[75] Paul D. Friz. “Parametric Cost Estimates of Four 20 Ton Payload Mars EDL Vehicle Concepts”. In: *AIAA Scitech 2020 Forum*. DOI: 10.2514/6.2020-1514. URL: https://arc.aiaa.org/doi/abs/10.2514/6.2020-1514.

[76] Lingchong Gao et al. “A Survey of Lunar Cranes and Some Inspiration from the Perspective of Earth Crane Technology”. In: July 2019, pp. 631–636. DOI: 10.1109/CYBER46603.2019.9066753.

[77] Thomas C. Jones. “A Protoflight Lightweight Surface Manipulation System to Enable High Load, Long-Reach Lunar Surface Operations”. In: *ASCEND 2021*. DOI: 10.2514/6.2021-4167. URL: https://arc.aiaa.org/doi/abs/10.2514/6.2021-4167.

[78] Anton Schneider et al. “Deployment dynamics analysis of CALLISTO’s approach and landing system”. In: *CEAS Space Journal* (Dec. 2021). DOI: 10.1007/s12567-021-00411-2.

[79] Manan Arya, Nicolas Lee, and Sergio Pellegrino. “Ultralight Structures for Space Solar Power Satellites”. In: *3rd AIAA Spacecraft Structures Conference*. DOI: 10.2514/6.2016-1950. eprint: https://arc.aiaa.org/doi/pdf/10.2514/6.2016-1950. URL: https://arc.aiaa.org/doi/abs/10.2514/6.2016-1950.

[80] NASA. *Technology readiness level definitions*.

[81] Stephen P. Timoshenko and James M. Gere. Dover Publications, 1989. ISBN: 978-0-486-47207-2.

[82] T. H. G. Megson. *9.2 Inelastic Buckling of Plates*. 2010. URL: https://app.knovel.com/hotlink/khtml/id:kt008U8T1A/introduction-aircraft/inelastic-buckling-plates.

[83] Martin Mikulas and Aaron Adler. “Rapid Structural Assessment Approach for Square Solar Sails Including Edge Support Cords”. In: *44th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*. DOI: 10.2514/6.2003-1447. eprint: https://arc.aiaa.org/doi/pdf/10.2514/6.2003-1447. URL: https://arc.aiaa.org/doi/abs/10.2514/6.2003-1447.

[84] *Buckling of Thin-Walled Circular Cylinders*. NASA/SP-8007-2020. Rev. 2. National Aeronautics and Space Administration (NASA). Nov. 2020.

[85] Hiraku Sakamoto, K.C. Park, and Yasuyuki Miyazaki. “Evaluation of membrane structure designs using boundary web cables for uniform tensioning”. In: *Acta Astronautica* 60.10 (2007), pp. 846–857. ISSN: 0094-5765. DOI: https://doi.org/10.1016/j.actaastro.2006.10.008.

[86] *Vectran: Product Data Sheet*. V1. Kuraray. July 2020.

[87] Yuichi Tsuda et al. “Achievement of IKAROS Japanese deep space solar sail demonstration mission”. In: *Acta Astronautica* 82.2 (2013). 7th IAA Symposium on Realistic Advanced Scientific Space MissionsAosta, Italy, July 2011, pp. 183–188. ISSN: 0094-5765. DOI: https://doi.org/10.1016/j.actaastro.2012.03.032.

[88] *ABAQUS Analysis User's Manual, Version 6.14*. United States: Dassault Systèmes Simulia Corp, 2014.

[89] Thomas L Saaty. "A scaling method for priorities in hierarchical structures". In: *Journal of Mathematical Psychology* 15.3 (1977), pp. 234–281. ISSN: 0022-2496. DOI: https://doi.org/10.1016/0022-2496(77)90033-5. URL: https://www.sciencedirect.com/science/article/pii/0022249677900335.

[90] R.W. Saaty. "The analytic hierarchy process—what it is and how it is used". In: *Mathematical Modelling* 9.3 (1987), pp. 161–176. ISSN: 0270-0255. DOI: https://doi.org/10.1016/0270-0255(87)90473-8. URL: https://www.sciencedirect.com/science/article/pii/0270025587904738.

[91] Thomas L. Saaty. "Analytic Hierarchy Process". In: *Encyclopedia of Biostatistics*. John Wiley and Sons, Ltd, 2005. ISBN: 9780470011812. DOI: https://doi.org/10.1002/0470011815.b2a4a002. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/0470011815.b2a4a002. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/0470011815.b2a4a002.

A

Appendix A

A.1. Mars Solar Power Heritage

The application of solar cell technology on Mars has been on going for the past couple decades, being successfully utilized on both static landers and robotic rovers alike. The first surface application occurred in 1997 through the Pathfinder spacecraft and its accompanying rover, Sojourner. Sojourner, NASA's first wheeled robot to land on the surface of another planet [3], was powered by 0.22 square meters of single junction Gallium Arsenide solar cells mounted on a rigid panel, possessing a modest 18 percent cell efficiency [4][5]. Photovoltaic (PV) technology has evolved considerably since then and its evolution can be observed periodically in the missions sent after Pathfinder. Specifically, The Mars Exploration Rovers (MERs) Spirt and Opportunity in 2004 were the first application of more efficient triple junction solar cells on Mars [14]. More efficient triple junction PV cells followed thereafter with the Phoenix Lander in 2008 and the Insight Lander in 2018. The latest application of better performing solar cells on Mars occurred this year with the Ingenuity helicopter which possesses 0.07 square meters of Inverted Metamorphic (IMM) four function solar cells [15]. A tabulated summary of solar cell applications on Mars is provided in Table A.1 along with the associated PV surface areas and technologies. As an important side note, the Chinese Space Agency (CNSA) landed a solar powered rover on Mars this past year but credible information regarding the onboard PV technology is sparce/nonexistent and thus was omitted from this report.

Table A.1: Mars surface mission utilizing solar power

Vehicle	Year	Solar Cell Technology	Manufacturer	Surface Area (m^2)	Mission Duration (Earth Days)
Sojourner Rover	1997	Single Junction (GaAs/Ge)	Applied Solar Energy Corporation	0.22	83
Mars Exploration Rover (Spirit)	2004	Triple Junction (GaInP2/GaAs/Ge) (ITJ)	Spectrolab	~1.2*	2269 (6 years)
Mars Exploration Rover (Opportunity)	2004	Triple Junction (GaInP2/GaAs/Ge) (ITJ)	Spectrolab	~1.2*	5498 (15 years)
Phoenix Lander	2008	Triple Junction (GaInP2/InGaAs/Ge) (UTJ)	Spectrolab	6.9	162
Insight Lander	2018	Triple Junction (InGaP/InGaAs/Ge) (ZTJ)	SolAero Technologies	7.3	On going
Ingenuity	2021	Inverted Metamorphic Four Junction (IMM4J)	SolAero Technologies	0.07	On going
* solar cell area, not including additional panel area					

A.2. Analytic Hierarchy Process (AHP): Decision Making Tool

Analytic Hierarchy Process (AHP) is a generally applicable tool introduced by Saaty involving the decomposition a decision making problem into a hierarchical structure including a goal, a variety of assessment criteria, and alternatives [89]. Alternatives, referring to choices available to the practitioner, are situated at the base of the hierarchy whereas the goal or desired outcome of the down-selection decision is at the top. Depending on the complexity of the problem at hand, several different interstitial levels exist in this hierarchy of which are composed of criteria and related sub-criteria, if any, that relative comparisons are made from among the alternatives. Moreover, each criteria represents a different branch in the hierarchical structure where additional layers of sub-criteria can also stem from. For every criterion in the hierarchy, pairwise comparisons are performed to establish their relative weights. Each alternative is then ranked against each other through a series of pairwise comparisons for each criterion. Collectively, these comparisons allow for a final selection among the alternatives to be made.

To begin, decision criteria are first articulated. These criteria must be mutually preferentially independent from one another otherwise the Analytic Hierarchy Process is not suitable. Criteria and subsequent levels of sub-criteria, if any, are added till the decision making problem is properly captured in sufficient detail.From here, criteria from each hierarchical level are arranged in a square matrix form, as shown in Table A.2. For the purposes of this demonstration, only one level of criteria is used. Qualitatively, pairwise comparisons are then made among these criteria to establish their relative importance to the practitioner. The scale used in these comparison are derived from Saaty and shown in Table A.3.

Table A.2: Pairwise comparison matrix of criteria

		Criteria		
		A	B	C
Criteria	A	1	3	8
	B	1/3	1	2
	C	1/6	1/2	1

After the criteria scoring is completed, the matrix in Table A.2 is normalized and the average is taken of every row to derive the criteria weights (Table A.3). To determine whether or not these derived weights can be used, the AHP includes a consistency test which is essentially a measure of whether or not the scoring abides by transitive logic. That is, if Criteria A is determined be three times more important than B and six times more important than C, then the scoring should also reflect that Criteria B is double the importance of C. Otherwise, the matrix is inconsistent meaning ranking was done arbitrarily and has little value. It should be noted that a more consistent rating system does not reflect the accuracy of the rating therein but rather only relates how random the scoring was performed. In reality, cardinal consistency is rarely observed meaning some degree of inconsistency is usually present. As one can imagine, the process of logically applying scoring among criteria or alternatives becomes more and more difficult as their numbers and hierarchy layers increase. That said, depending on the number of matrix constituents, AHP allows the use of the weights derived from the pairwise comparison provided the determined inconsistency is below a certain value.

To compute the consistency of a matrix, the following steps are performed. First, each column of the original pairwise comparison matrix is multiplied by the corresponding row of the weight vector.Each row of this new matrix is summed and then divided by the corresponding row of the weight vector. The result is a vector of eigenvalues which are perturbed from that of the eigenvalue of an equivalently sized, consistent reciprocal matrix. See Table A.5. In the ideal case, the pairwise comparison matrix is both reciprocal and consistent in nature meaning it is rank one. It can be shown that only one eigenvalue of this matrix is nonzero and is equal to its trace. This value is referred to as 'n'.

A summation of the perturbed eigenvalues is taken and divided by n resulting in what is referred to as the max eigenvalue. A so called consistency index (CI) is then taken as shown in Equation A.1

Table A.3: AHP pairwise scale [90]

Intensity of importance on an absolute scale	Definition	Explanation
1	Equal importance	Two activities contribute equally to the objective
3	Moderate importance of one over another	Experience and judgement slightly favor one activity over another
5	Essential or strong importance	Experience and judgement strongly favor one activity over another
7	Very strong importance	An activity is strongly favored and its dominance demonstrated in practice
9	Extreme importance	The evidence favoring one activity over another is of the highest possible order of affirmation
2,4,6,8	Intermediate values between two adjacent judgements	When compromise is needed

Table A.4: Normalized criteria pairwise comparison matrix and weights

		Criteria			
		A	B	C	Weights
Criteria	A	2/3	2/3	5/7	0.69
	B	2/9	2/9	1/5	0.21
	C	1/9	1/9	0	0.10

$$CI = \frac{\lambda_{\text{max}} - n}{n - 1} \tag{A.1}$$

The CI is then divided by a predetermined, tabulated random index (RI) (Table A.3) value whose value depends on the order of the original pairwise comparison matrix. This RI number is computed by taking the average of several thousand consistency index values produced from several thousand reciprocal matrices of randomly generated numbers within the boundaries the scaling system used in the pairwise comparisons. The consistency of the pairwise comparison is thus shown in Equation A.2.

$$Consistency = \frac{\lambda_{\text{max}} - n}{(n - 1) * RI} \tag{A.2}$$

For the small example provided, the consistency is computed to be 0.103. Per AHP standards prescribed by Saaty, a matrix can be considered consistent, meaning its derived weights can be used, if the consistency is less than 0.10. For this example, this criteria is not met. Reevaluation of the original pairwise comparison is thus required. After this revision is performed and the consistency is recomputed to ensure compliance with the aforementioned standard, the available alternatives can be assessed against each other for each criterion. The same process used previously to derive the weights of the criteria is now implemented for the alternatives. In the case presented here, the results should be three criteria weights and three weights with respect to each available alternative for each

Table A.5: Matrix for determining the perturbed eigenvalues

		Criteria				
		A	B	C	Row Sums	Perturbed Eigenvalues
Criteria	A	0.686869	0.626263	0.835017	2.148148	3.12745098
	B	0.228956	0.208754	0.208754	0.646465	3.096774194
	C	0.114478	0.104377	0.104377	0.323232	3.096774194

Table A.6: Table of RI values [91]

Number of criteria	1	2	3	4	5	6	7	8	9	10
Random consistency	0	0	0.52	0.89	1.11	1.25	1.35	1.4	1.45	1.49

criteria. In total, twelve weights are computed with three weights assigned to each alternative for each of the assessed criteria. To assess how the alternatives perform against each other in total, each of the weights for an alternative are multiplied by their respective criteria weight and summed. The alternative with the highest score is selected. It should be noted that for each of the pairwise comparisons among alternatives in each criteria, a consistency check should be performed.

B

Code

```
# ----------------------------------------------------
# ----------------- INITIALIZE MODEL
# ----------------------------------------------------
import os
from csv import *
import numpy as np
import os
from collections import OrderedDict
from part import *
from material import *
from section import *
from assembly import *
from step import *
from interaction import *
from load import *
from mesh import *
from optimization import *
from job import *
from sketch import *
from visualization import *
from connectorBehavior import *
import math
import matplotlib.pyplot as plt
import numpy as np
from sympy import *
from abaqusConstants import *
import itertools
from abaqus import *
from timeit import default_timer as timer
import shutil
session.journalOptions.setValues(replayGeometry=COORDINATE, recoverGeometry=COORDINATE)
# ----------------------------------------------------
# ----------------- SETTING GLOBAL WORKING DIRECTORY
# ----------------------------------------------------
WorkingDirectory = "D:/ppcragg Thesis/Abaqus"
os.chdir(WorkingDirectory)
# ----------------------------------------------------
# ----------------- CREATE FOLDER TO STORE PHOTOS
# ----------------------------------------------------
# mode shape pictures
newpath = WorkingDirectory+'/1stModal'
if not os.path.exists(newpath):
    os.makedirs(newpath)

# deflection pictures
newpath = WorkingDirectory+'/U3Deflections'
if not os.path.exists(newpath):
    os.makedirs(newpath)

# strip pictures
```

```
newpath = WorkingDirectory+'/S11'
if not os.path.exists(newpath):
    os.makedirs(newpath)

# ------------------------------------------------
# ---------------- ANALYSIS TYPE
# ------------------------------------------------
Analysis_Flag = False
#         True = Parametric Assessment
#         False = Individual Assessment
Requirements_Flag = False
#         True = restrict analysis to mars req.
#         False = allow all configuratoins
# ------------------------------------------------
# ---------------- VARIABLES
# ------------------------------------------------
Inputs = dict([
    ('Area',np.linspace(1000,3000,3)),                  #(m^2) Total Array Area
    ('k',22),                                           # number of strips
    ('t_ts',np.linspace(0.001,.003,3)),                 #(m) Telescoping Beam Thickness
    ('r_ts',np.linspace(.35,.5,5)),                     #(m) Root Telescoping Beam Radius
    ('Stroke',np.linspace(0.05,.25,7)),                 #(m) Displacement applied to the
        pulley cables
    ('t_s_max',np.linspace(0.001,0.005,5)),             #(m) thickness of the reinforcing
        mesh in each strip
    ('r_dc',0.0075),                                    #(m) radius of the diagonal cables
    ('%_EarthGravity', np.linspace(0.5,1.5,5))          #(%) percent of earth gravity
    ])
Freq_Number = 10                                        #number of frequencies to extract
    during modal analysis
# ------------------------------------------------
# ---------------- MESH CONTROLS
# ------------------------------------------------
Mesh_Controls = dict([
    ('element_size_dc',1),                          #diagonal cable element size
    ('element_size_s',.75),                         #strip element size
    ('node_number_ts',7)                            #telescoping beam node number
    ])
print(Mesh_Controls)
# ------------------------------------------------
# ---------------- CONSTANTS
# ------------------------------------------------
Assembly_Constants = dict([
    ('SF',4),                                       # max unloaded bend radius factor of
        cables
    ('p_tip',0.5),                                  #(m) assumed spacing between diagonal
         cable and tip of the telescoping booms
    ('EarthGravity',9.81),                          #(m/s^2)
    ('Mars_rho',0.023),                             #(kg/m^3) max atmospheric density
    ('Lander_Diameter',9.1),                        #(m) Assumed Lander Diameter of the
        lander
    ('CE_Height',3),                                #(m) Cargo Envelope (CE) height ontop
         of the lander
    ('TS_Height',3.5),                              #(m) Height of the angled telescoping
         support (TS)
    ('TN',6),                                       # Assumed number of telescoping
        sections
    ('Ls',2),                                       # Assumed length of the support arm
    ('L_i',7),                                      #(m) inner square side length
    ('mat','Dyneema'),                              # Diagonals Material Name
    ('E',1.2E11),                                   #(Pa) Young's Modulus
    ('YS',2.5E9),                                   #(Pa) Yield Strength
    ('v',.41),                                      # Poisson's Ratio
    ('rho',980),                                    #(kg/m^3) Density
    ('mat_s','Vectran_Support'),                    # Support Cables Material Name
    ('E_s',5.5E10),                                 #(Pa) Young's Modulus
    ('YS_s',2.9E9),                                 #(Pa) Yield Strength
    ('v_s',.35),                                    # Poisson's Ratio
    ('rho_s',1400),                                 #(kg/m^3) Density
    ('mat_ts','CarbonFiber'),                       # Telescoping Supports Material Name
    ('E_ts',4.42e10),                               #(Pa) Young's Modulus
```

```
    ('YS_ts',4.5e8),                                    #(Pa) Yield Strength
    ('v_ts',.33),                                       # Poisson's Ratio
    ('rho_ts',1610),                                    #(kg/m^3) Density
    ('CablesPerMeter',40),                              #(#) number of lengthwise cables per
        strip, per meter
    ('SlipRing_r',0.015)
    ])
# ------------------------------------------------
# ----------------- FUNCTIONS
# ------------------------------------------------
def GenerateConfigurations(Inputs):
    k,r_dc = [Inputs['k']],[Inputs['r_dc']]
    combinations = list(itertools.product(Inputs['Area'],k,Inputs['t_ts'],Inputs['r_ts'],
        Inputs['Stroke'],Inputs['t_s_max'],r_dc,Inputs['%_EarthGravity']))
    print('Total Number of combinations: ' + str(len(combinations)))
    return combinations

def ExtractResults(path):
    extracted_line=''
    minData,maxData = [],[]
    f = open(path)
    for line in f:
        str1=line
        if 'Maximum' in str1:
            extracted_line = str1
            extracted_list = extracted_line.split()
            extracted_list = extracted_list[1:]
            for item in range(1,len(extracted_list)+1):
                value = float(extracted_list[item-1])
                maxData.append(value)

    f.close()
    f = open(path)
    for line in f:
        str1=line
        if 'Minimum' in str1:
            extracted_line = str1
            extracted_list = extracted_line.split()
            extracted_list = extracted_list[1:]
            for item in range(1,len(extracted_list)+1):
                value = float(extracted_list[item-1])
                minData.append(value)

    f.close()
    return minData,maxData

def WriteResults(path,sortItem,odb,step,frame,outputPosition,variable):
    session.writeFieldReport(fileName=path,append=OFF,
                             sortItem = sortItem, odb = odb,
                             step=step,frame=frame,outputPosition=outputPosition,
                             variable = variable)

def StaticAnalysis(ID,Configuration):
    # CREATE FOLDER FOR ANALYSIS
    newpath = WorkingDirectory+'/Analysis'+str(ID)
    if not os.path.exists(newpath):
        os.makedirs(newpath)
    os.chdir(newpath)
    # PROCEED WITH ANALYSIS
    Inputs = dict([
        ('Area',Configuration[0]),
        ('k',Configuration[1]),
        ('t_ts',Configuration[2]),
        ('r_ts',Configuration[3]),
        ('Stroke',Configuration[4]),
        ('t_s_max',Configuration[5]),
        ('r_dc',Configuration[6]),
        ('%_EarthGravity',Configuration[7])
        ])
    #-------------------------------------------------------------------------
    #              GENERATE GEOMETRIC PARAMETERS
```

```
    #-------------------------------------------------------------------------
    # COMPUTE STRIP WIDTH (GLOBAL AND LOCAL COORD. SYSTEM)
    L_o = math.sqrt(Inputs['Area']+Assembly_Constants['L_i']**2)
    x_i,x_f = Assembly_Constants['L_i']/math.sqrt(2),L_o/math.sqrt(2)
    Spacing = np.linspace(x_i,x_f,Inputs['k']+1)
    Strip_w = Spacing[1]-Spacing[0]                                          #
        strip axial width (global CSYS)
    Strip_h = Strip_w/math.sqrt(2)                                           #
        strip diagonal width (local CSYS)
    Inner_Length = Assembly_Constants['L_i']
    Outer_Length = Assembly_Constants['L_i']+2*Strip_w/math.sqrt(2)
    Strip_Area = (Inner_Length+Outer_Length)/2*Strip_h
    # COMPUTE IDEALIZED STRIP ATTACHMENT POINTS & DIAGONAL CABLE POINTS
    AttachmentPoints_S,AttachmentPoints_DC = [],[]
    for i in range(1,Inputs['k']+1):
        if i == 1:
            AttachmentPoints_S.append(x_i+Strip_w/2)
            AttachmentPoints_DC.append(0.10)
            AttachmentPoints_DC.append(x_i+Strip_w/2)
        else:
            AttachmentPoints_S.append(AttachmentPoints_S[-1]+Strip_w)
            AttachmentPoints_DC.append(AttachmentPoints_DC[-1]+Strip_w)
    AttachmentPoints_DC.append(x_f)
    # COMPUTE STRIP AREAS
    Areas = []
    for i in range(1,len(AttachmentPoints_S)+1):
        innerp,outerp = (AttachmentPoints_S[i-1]-Strip_w/2)*math.sqrt(2),(AttachmentPoints_S[
            i-1]+Strip_w/2)*math.sqrt(2)
        Area_temp = (innerp+outerp)/2*Strip_h
        Areas.append(Area_temp)
    # STRIP LENGTHS AND  EQUIVALENT RADII
    Strip_Lengths,Strip_Radii= [x*math.sqrt(2) for x in AttachmentPoints_S],[]
    Length_Ratio = [x/x for x in Strip_Lengths]
    Cables_PerStrip = round(Strip_h*Assembly_Constants['CablesPerMeter'])
    TotalCableArea_PerStrip = Cables_PerStrip*np.pi*(Inputs['t_s_max']/2)**2
    for i in range(1,Inputs['k']+1):
        rad = float(math.sqrt(TotalCableArea_PerStrip/np.pi))
        Strip_Radii.append(rad)
    # TELESCOPING SUPPORT COMPUTATIONS
    overhang= (x_f+Assembly_Constants['p_tip'])-\
        (Assembly_Constants['Lander_Diameter']/2)
    TS_L = math.sqrt((overhang)**2+(Assembly_Constants['TS_Height'])**2)     #
        (m) Deployment length of the structure (per the Array Arrangement tool)
    theta = math.atan(Assembly_Constants['TS_Height']/(overhang))            #
        (Deg) Deployment Angle
    theta_s = abs(theta-np.pi/2)
    TLength = TS_L/Assembly_Constants['TN']                                  #
        (m) Length of each telescoping section
    TPoints = [[Assembly_Constants['Lander_Diameter']/2,-Assembly_Constants['TS_Height']]]  #
        Points for construction of the Telescoping beam
    TMPoints = []                                                            #
        compute midpoints for making sets later
    for i in range(1,Assembly_Constants['TN']+1):                            #
        compute points for rest of sections
        temp = [TPoints[-1][0]+(np.cos(theta)*TLength),
            TPoints[-1][1]+(np.sin(theta)*TLength)]
        temp2 = [TPoints[-1][0]+(np.cos(theta)*TLength)/2,
            TPoints[-1][1]+(np.sin(theta)*TLength)/2]
        TPoints.append(temp)
        TMPoints.append(temp2)
    # COMPUTE PULLEY LENGTH
    Length_Pulley = math.sqrt(((TPoints[0][0]+Assembly_Constants['Ls']*np.cos(theta_s))\
        -TPoints[-1][0])**2+(TPoints[0][1]+Assembly_Constants['Ls']*np.sin(theta_s))**2)
    #-------------------------------------------------------------------------
    #              INITIALIZE MODEL
    #-------------------------------------------------------------------------
    # CREATE THE MODEL
    mdb.Model(modelType=STANDARD_EXPLICIT, name='Quadrent')
    del mdb.models['Model-1']
    Stripped_Array = mdb.models['Quadrent']
    # CREATE MATERIALS
```

```
    mat =Assembly_Constants['mat']
    Stripped_Array.Material(name=mat)
    Stripped_Array.materials[mat].Elastic(table=((Assembly_Constants['E'],
        Assembly_Constants['v']), ))
    Stripped_Array.materials[mat].Density(table=((Assembly_Constants['rho'], ), ))
    mat_s = Assembly_Constants['mat_s']
    Stripped_Array.Material(name=mat_s)
    Stripped_Array.materials[mat_s].Elastic(table=((Assembly_Constants['E_s'],
        Assembly_Constants['v_s']), ))
    Stripped_Array.materials[mat_s].Density(table=((Assembly_Constants['rho_s'], ), ))
    mat_ts = Assembly_Constants['mat_ts']
    Stripped_Array.Material(name=mat_ts)
    Stripped_Array.materials[mat_ts].Density(table=((Assembly_Constants['rho_ts'], ), ))
    Stripped_Array.materials[mat_ts].Elastic(table=((Assembly_Constants['E_ts'],
        Assembly_Constants['v_ts']), ))
    #-------------------------------------------------------------------------
    #              MEMBRANE SUBASSEMBLY
    #-------------------------------------------------------------------------
    # CREATE THE DIAGONAL CABLE
    Stripped_Array.ConstrainedSketch(name='__profile__', sheetSize=200.0)
    for i in range(1,len(AttachmentPoints_DC)):
        Stripped_Array.sketches['__profile__'].Line(point1=(AttachmentPoints_DC[i-1],0.0),
        point2=(AttachmentPoints_DC[i],0.0))
    Stripped_Array.Part(dimensionality=THREE_D, name='Diagonal_Cable', type=
        DEFORMABLE_BODY)
    Stripped_Array.parts['Diagonal_Cable'].BaseWire(sketch=
        Stripped_Array.sketches['__profile__'])
    # CREATE THE STRIPS
    Stripped_Array.ConstrainedSketch(name='__profile__', sheetSize=200.0)
    for i in range(1,len(AttachmentPoints_S)+1):
        Stripped_Array.sketches['__profile__'].Line(point1=(AttachmentPoints_S[i-1],0),
        point2=(0,AttachmentPoints_S[i-1]))
    Stripped_Array.Part(dimensionality=THREE_D, name='Strips', type=
        DEFORMABLE_BODY)
    Stripped_Array.parts['Strips'].BaseWire(sketch=
        Stripped_Array.sketches['__profile__'])
    # CREATE SECTIONS AND MESH THE PARTS
    # DIAGONAL CABLE
    Stripped_Array.CircularProfile(name='Diagonal_Cable_Profile', r=Inputs['r_dc'])
    Stripped_Array.BeamSection(consistentMassMatrix=False, integration=
        DURING_ANALYSIS, material='Dyneema', name='Beam', poissonRatio=0.0,
        profile='Diagonal_Cable_Profile', temperatureVar=LINEAR)
    Stripped_Array.parts['Diagonal_Cable'].SectionAssignment(offset=0.0,
        offsetField='', offsetType=MIDDLE_SURFACE, region=Region(
        edges=Stripped_Array.parts['Diagonal_Cable'].edges[:]), sectionName='Beam',
        thicknessAssignment=FROM_SECTION)
    mdb.models['Quadrent'].parts['Diagonal_Cable'].seedEdgeBySize(constraint=FINER,
        deviationFactor=0.1, edges= Stripped_Array.parts['Diagonal_Cable'].edges[:], size=
            Mesh_Controls['element_size_dc'])
    Stripped_Array.parts['Diagonal_Cable'].setElementType(elemTypes=(
        ElemType(elemCode=B33, elemLibrary=STANDARD), ), regions=(
        Stripped_Array.parts['Diagonal_Cable'].edges[:], ))
    Stripped_Array.parts['Diagonal_Cable'].generateMesh()
    Stripped_Array.parts['Diagonal_Cable'].assignBeamSectionOrientation(
        method=N1_COSINES, n1=(0.0, 0.0, -1.0), region=Region(
        edges=Stripped_Array.parts['Diagonal_Cable'].edges[:]))
    # STRIPS
    for i in range(1,Inputs['k']+1):
        rad = Strip_Radii[i-1]
        name,name_p,name_set = 'Strip_' +str(i),'Strip_' +str(i) +'_profile','Strip_Set' +str
            (i)
        temp = Stripped_Array.parts['Strips'].edges.findAt(((AttachmentPoints_S[i-1]/2,
            AttachmentPoints_S[i-1]/2,0.0),))
        Stripped_Array.parts['Strips'].Set(edges=temp, name=name_set)
        Stripped_Array.CircularProfile(name=name_p, r=rad)
        Stripped_Array.BeamSection(consistentMassMatrix=False, integration=
            DURING_ANALYSIS, material=mat_s, name=name, poissonRatio=0.0,
            profile=name_p, temperatureVar=LINEAR)
        Stripped_Array.parts['Strips'].SectionAssignment(offset=0.0,
            offsetField='', offsetType=MIDDLE_SURFACE, region=
            Stripped_Array.parts['Strips'].sets[name_set], sectionName=
```

```
            name, thicknessAssignment=FROM_SECTION)
    mdb.models['Quadrent'].parts['Strips'].seedEdgeBySize(constraint=FINER,
        deviationFactor=0.1, edges= Stripped_Array.parts['Strips'].edges[:], size=
            Mesh_Controls['element_size_s'])
    Stripped_Array.parts['Strips'].setElementType(elemTypes=(ElemType(
        elemCode=B33, elemLibrary=STANDARD), ), regions=(
        Stripped_Array.parts['Strips'].edges[:], ))
    Stripped_Array.parts['Strips'].generateMesh()
    Stripped_Array.parts['Strips'].assignBeamSectionOrientation(method=
        N1_COSINES, n1=(0.0, 0.0, -1.0), region=Region(
        edges=Stripped_Array.parts['Strips'].edges[:]))
    #----------------------------------------------------------------------
    #              TELESCOPING BEAMS
    #----------------------------------------------------------------------
    # INITIALIZE TELESCOPING BEAMS
    Stripped_Array.Part(dimensionality=THREE_D, name='Telescoping_Beam',
        type=DEFORMABLE_BODY)
    # BUILD THE TELESCOPING BEAMS
    for i in range(1,len(TPoints)+1): #FIRST BEAM
        #construct datum points
        Stripped_Array.parts['Telescoping_Beam'].DatumPointByCoordinate(
            coords=(TPoints[i-1][0],0,TPoints[i-1][1]))
    #compile points for lines (first beam)
    datumpoints = ()
    for i in range(1,len(TPoints)):
        datumpoints = datumpoints +((Stripped_Array.parts['Telescoping_Beam'].datums[i],
        Stripped_Array.parts['Telescoping_Beam'].datums[i+1]),)
    Stripped_Array.parts['Telescoping_Beam'].WirePolyLine(mergeType=
        IMPRINT, meshable=ON, points=datumpoints)
    #create set
    Stripped_Array.parts['Telescoping_Beam'].Set(edges=
        Stripped_Array.parts['Telescoping_Beam'].edges[:], name='TS_1')
    # DEFINE THE MESH MATERIAL & CROSS SECTION
    TRadii = []
    TThicknesses = []
    # cross section for telescoping beams
    for i in range(1,Assembly_Constants['TN']+1):
        rx,tx = Inputs['r_ts']-(Inputs['t_ts']*(i-1)),Inputs['t_ts']
        TRadii.append(rx)
        TThicknesses.append(tx)
        ProfName,SectName = 'TProf_'+repr(i-1),'TSect_'+repr(i-1)
        Stripped_Array.PipeProfile(name=ProfName,r=rx, t=tx,)
        Stripped_Array.BeamSection(consistentMassMatrix=False, integration=
            DURING_ANALYSIS, material= mat_ts, name=SectName, poissonRatio=0.0,
            profile=ProfName, temperatureVar=LINEAR)
    # DEFINE SET
    for i in range(1,Assembly_Constants['TN']+1):
        temp = Stripped_Array.parts['Telescoping_Beam'].edges.findAt(((TMPoints[i-1][0],0,
        TMPoints[i-1][1]),))
        SetName = 'TSet_'+repr(i-1)
        Stripped_Array.parts['Telescoping_Beam'].Set(edges=temp,
            name=SetName)
    #ASSIGN SECTIONS
    for i in range(1,Assembly_Constants['TN']+1):
        temp = 'TSet_'+repr(i-1)
        temp2 = 'TSect_'+repr(i-1)
        Stripped_Array.parts['Telescoping_Beam'].SectionAssignment(offset=0.0,
            offsetField='', offsetType=MIDDLE_SURFACE, region=
            Stripped_Array.parts['Telescoping_Beam'].sets[temp], sectionName=
            temp2, thicknessAssignment=FROM_SECTION)
    Stripped_Array.parts['Telescoping_Beam'].Set(edges=
        Stripped_Array.parts['Telescoping_Beam'].edges[:], name='ALL')
    Stripped_Array.parts['Telescoping_Beam'].assignBeamSectionOrientation(method=
        N1_COSINES, n1=(0.0, 0.0, -1.0), region=
        Stripped_Array.parts['Telescoping_Beam'].sets['ALL'])
    # DEFINE AND GENERATE MESH
    Stripped_Array.parts['Telescoping_Beam'].seedEdgeByNumber(constraint=FINER,
        edges=Stripped_Array.parts['Telescoping_Beam'].edges[:], number=Mesh_Controls['
            node_number_ts'])
    Stripped_Array.parts['Telescoping_Beam'].setElementType(elemTypes=(ElemType(
        elemCode=B33, elemLibrary=STANDARD), ), regions=(
```

```
        Stripped_Array.parts['Telescoping_Beam'].edges[:], ))
    Stripped_Array.parts['Telescoping_Beam'].generateMesh()
    Stripped_Array.parts['Telescoping_Beam'].assignBeamSectionOrientation(method=
        N1_COSINES, n1=(0.0, 0.0, -1.0), region=Region(
        edges=Stripped_Array.parts['Telescoping_Beam'].edges[:]))
    #-----------------------------------------------------------------------
    #              ASSEMBLE THE MODEL
    #-----------------------------------------------------------------------
    mdb.models['Quadrent'].rootAssembly.DatumCsysByDefault(CARTESIAN)
    # CREATE "MEMBRANE" SUB-ASSEMBLY
    mdb.models['Quadrent'].rootAssembly.Instance(dependent=ON, name=
        'Diagonal_Cable-1', part=mdb.models['Quadrent'].parts['Diagonal_Cable'])
    mdb.models['Quadrent'].rootAssembly.Instance(dependent=ON, name=
        'Strips-1', part=mdb.models['Quadrent'].parts['Strips'])
    mdb.models['Quadrent'].rootAssembly.RadialInstancePattern(axis=(0.0, 0.0, 1.0),
        instanceList=('Diagonal_Cable-1', 'Strips-1'),
        number=4, point=(0.0, 0.0, 0.0), totalAngle=360.0)
    mdb.models['Quadrent'].rootAssembly._previewMergeMeshes(instances=(
        mdb.models['Quadrent'].rootAssembly.instances['Diagonal_Cable-1'],
        mdb.models['Quadrent'].rootAssembly.instances['Diagonal_Cable-1-rad-2'],
        mdb.models['Quadrent'].rootAssembly.instances['Diagonal_Cable-1-rad-3'],
        mdb.models['Quadrent'].rootAssembly.instances['Diagonal_Cable-1-rad-4']),
        mergeBoundaryOnly=False, nodeMergingTolerance=1e-06)
    mdb.models['Quadrent'].rootAssembly.InstanceFromBooleanMerge(domain=MESH,
        instances=(mdb.models['Quadrent'].rootAssembly.instances['Diagonal_Cable-1'],
        mdb.models['Quadrent'].rootAssembly.instances['Diagonal_Cable-1-rad-2'],
        mdb.models['Quadrent'].rootAssembly.instances['Diagonal_Cable-1-rad-3'],
        mdb.models['Quadrent'].rootAssembly.instances['Diagonal_Cable-1-rad-4'],),
        mergeNodes=ALL, name='Diagonals', nodeMergingTolerance=1e-06,
        originalInstances=SUPPRESS)
    mdb.models['Quadrent'].parts['Diagonals'].assignBeamSectionOrientation(method=
        N1_COSINES, n1=(0.0, 0.0, -1.0), region=Region(
        elements=mdb.models['Quadrent'].parts['Diagonals'].elements[:]))
    mdb.models['Quadrent'].rootAssembly.regenerate()
    # CREATE "TELESCOPING BEAM" SUB-ASSEMBLY
    mdb.models['Quadrent'].rootAssembly.Instance(dependent=ON, name=
        'Telescoping_Beam-1', part=
        mdb.models['Quadrent'].parts['Telescoping_Beam'])
    mdb.models['Quadrent'].rootAssembly.RadialInstancePattern(axis=(0.0, 0.0, 1.0),
        instanceList=('Telescoping_Beam-1',),
        number=4, point=(0.0, 0.0, 0.0), totalAngle=360.0)
    mdb.models['Quadrent'].rootAssembly._previewMergeMeshes(instances=(
        mdb.models['Quadrent'].rootAssembly.instances['Telescoping_Beam-1'],
        mdb.models['Quadrent'].rootAssembly.instances['Telescoping_Beam-1-rad-2'],
        mdb.models['Quadrent'].rootAssembly.instances['Telescoping_Beam-1-rad-3'],
        mdb.models['Quadrent'].rootAssembly.instances['Telescoping_Beam-1-rad-4'],),
        mergeBoundaryOnly=False, nodeMergingTolerance=1e-06)
    mdb.models['Quadrent'].rootAssembly.InstanceFromBooleanMerge(domain=MESH,
        instances=(mdb.models['Quadrent'].rootAssembly.instances['Telescoping_Beam-1'],
        mdb.models['Quadrent'].rootAssembly.instances['Telescoping_Beam-1-rad-2'],
        mdb.models['Quadrent'].rootAssembly.instances['Telescoping_Beam-1-rad-3'],
        mdb.models['Quadrent'].rootAssembly.instances['Telescoping_Beam-1-rad-4'],),
        mergeNodes=ALL, name='TelescopingAssembly', nodeMergingTolerance=1e-06,
        originalInstances=SUPPRESS)
    mdb.models['Quadrent'].parts['TelescopingAssembly'].assignBeamSectionOrientation(method=
        N1_COSINES, n1=(0.0, 0.0, -1.0), region=Region(
        elements=mdb.models['Quadrent'].parts['TelescopingAssembly'].elements[:]))
    mdb.models['Quadrent'].rootAssembly.regenerate()
    #-----------------------------------------------------------------------
    #              NODE SETS, ELEMENT SETS, COUPLINGS
    #-----------------------------------------------------------------------
    # (fixed nodes at telescoping beam base)
    n2 = Stripped_Array.rootAssembly.instances['TelescopingAssembly-1'].nodes.getClosest((
        TPoints[0][0],0,TPoints[0][1]),).label
    n3 = Stripped_Array.rootAssembly.instances['TelescopingAssembly-1'].nodes.getClosest((0,
        TPoints[0][0],TPoints[0][1]),).label
    n4 = Stripped_Array.rootAssembly.instances['TelescopingAssembly-1'].nodes.getClosest((-
        TPoints[0][0],0,TPoints[0][1]),).label
    n5 = Stripped_Array.rootAssembly.instances['TelescopingAssembly-1'].nodes.getClosest((0,-
        TPoints[0][0],TPoints[0][1]),).label
    Stripped_Array.rootAssembly.Set(name='FixedNode_TS', nodes=(
```

```
449          Stripped_Array.rootAssembly.instances['TelescopingAssembly-1'].nodes[n2-1:n2]+\
450          Stripped_Array.rootAssembly.instances['TelescopingAssembly-1'].nodes[n3-1:n3]+\
451          Stripped_Array.rootAssembly.instances['TelescopingAssembly-1'].nodes[n4-1:n4]+\
452          Stripped_Array.rootAssembly.instances['TelescopingAssembly-1'].nodes[n5-1:n5]))
453      # (fixed node at center of membrane)
454      n6 = Stripped_Array.rootAssembly.instances['Diagonals-1'].nodes.getClosest((0.1,0,0),).
             label
455      n7 = Stripped_Array.rootAssembly.instances['Diagonals-1'].nodes.getClosest((0,0.1,0),).
             label
456      n8 = Stripped_Array.rootAssembly.instances['Diagonals-1'].nodes.getClosest((-0.1,0,0),).
             label
457      n9 = Stripped_Array.rootAssembly.instances['Diagonals-1'].nodes.getClosest((0,-0.1,0),).
             label
458      Stripped_Array.rootAssembly.Set(name='PinnedNodes_Center', nodes=(
459          Stripped_Array.rootAssembly.instances['Diagonals-1'].nodes[n6-1:n6]+\
460          Stripped_Array.rootAssembly.instances['Diagonals-1'].nodes[n7-1:n7]+\
461          Stripped_Array.rootAssembly.instances['Diagonals-1'].nodes[n8-1:n8]+\
462          Stripped_Array.rootAssembly.instances['Diagonals-1'].nodes[n9-1:n9]))
463      # (strip/diag sets + coupling)
464      counter = 1
465      CouplingOrder = ((1,4,1),(1,2,2),(2,3,3),(3,4,4))
466      instancename = ['Strips-1','Strips-1-rad-2','Strips-1-rad-3','Strips-1-rad-4']
467      for i in range(1,5):
468          for j in range(1,len(AttachmentPoints_S)+1):
469              tuplepoints = ((AttachmentPoints_S[j-1],0,0),(0,AttachmentPoints_S[j-1],0),(-
                     AttachmentPoints_S[j-1],0,0),(0,-AttachmentPoints_S[j-1],0))
470              s1,s2,apoints = instancename[CouplingOrder[i-1][0]-1],instancename[CouplingOrder[
                     i-1][1]-1],CouplingOrder[i-1][2]-1
471              strip1 = Stripped_Array.rootAssembly.instances[s1].nodes.getClosest(tuplepoints[
                     apoints],).label
472              strip2 = Stripped_Array.rootAssembly.instances[s2].nodes.getClosest(tuplepoints[
                     apoints],).label
473              diag = Stripped_Array.rootAssembly.instances['Diagonals-1'].nodes.getClosest(
                     tuplepoints[apoints],).label
474              sname,dname,cname = 'Strips-a-Set'+str(counter),'Diagonals-1-Set'+str(counter),'
                     Couple-a-S_DC-'+str(counter)
475              sname2,cname2 = 'Strips-b-Set'+str(counter),'Couple-b-S_DC-'+str(counter)
476              Stripped_Array.rootAssembly.Set(name=sname, nodes=(Stripped_Array.rootAssembly.
                     instances[s1].nodes[strip1-1:strip1]))
477              Stripped_Array.rootAssembly.Set(name=sname2, nodes=(Stripped_Array.rootAssembly.
                     instances[s2].nodes[strip2-1:strip2]))
478              Stripped_Array.rootAssembly.Set(name=dname, nodes=(Stripped_Array.rootAssembly.
                     instances['Diagonals-1'].nodes[diag-1:diag]))
479              mdb.models['Quadrent'].Coupling(controlPoint=
480                  mdb.models['Quadrent'].rootAssembly.sets[dname], couplingType=KINEMATIC
481                  , influenceRadius=WHOLE_SURFACE, localCsys=None, name=cname, surface=
482                  mdb.models['Quadrent'].rootAssembly.sets[sname], u1=ON, u2=ON,
483                  u3=ON, ur1=OFF, ur2=OFF, ur3=OFF)
484              mdb.models['Quadrent'].Coupling(controlPoint=
485                  mdb.models['Quadrent'].rootAssembly.sets[dname], couplingType=KINEMATIC
486                  , influenceRadius=WHOLE_SURFACE, localCsys=None, name=cname2, surface=
487                  mdb.models['Quadrent'].rootAssembly.sets[sname2], u1=ON, u2=ON,
488                  u3=ON, ur1=OFF, ur2=OFF, ur3=OFF)
489              counter = counter+1
490      # (diagonal nodes)
491      x = mdb.models['Quadrent'].rootAssembly.instances['Diagonals-1'].nodes.getByBoundingBox(
             xMin = -50,yMin = -0.01,zMin = -0.01,xMax = 50,yMax=0.01,zMax = 0.01)
492      y = mdb.models['Quadrent'].rootAssembly.instances['Diagonals-1'].nodes.getByBoundingBox(
             xMin = -0.01,yMin = -50,zMin = -0.01,xMax = 0.01,yMax=50,zMax = 0.01)
493      mdb.models['Quadrent'].rootAssembly.Set(name = 'DiagUR1',nodes=x)
494      mdb.models['Quadrent'].rootAssembly.Set(name = 'DiagUR2',nodes=y)
495      mdb.models['Quadrent'].DisplacementBC(amplitude=UNSET, createStepName='Initial'
496          , distributionType=UNIFORM, fieldName='', localCsys=None, name='D_UR1',
497          region=mdb.models['Quadrent'].rootAssembly.sets['DiagUR1'], u1=UNSET, u2=
498          UNSET, u3=UNSET, ur1=SET, ur2=UNSET, ur3=UNSET)
499      mdb.models['Quadrent'].DisplacementBC(amplitude=UNSET, createStepName='Initial'
500          , distributionType=UNIFORM, fieldName='', localCsys=None, name='D_UR2',
501          region=mdb.models['Quadrent'].rootAssembly.sets['DiagUR2'], u1=UNSET, u2=
502          UNSET, u3=UNSET, ur1=UNSET, ur2=SET, ur3=UNSET)
503      # (membrane node tips)
504      n10 = Stripped_Array.rootAssembly.instances['Diagonals-1'].nodes.getClosest((
```

```
        AttachmentPoints_DC[-1],0,0),).label
    n11 = Stripped_Array.rootAssembly.instances['Diagonals-1'].nodes.getClosest((0,
        AttachmentPoints_DC[-1],0),).label
    n12 = Stripped_Array.rootAssembly.instances['Diagonals-1'].nodes.getClosest((-
        AttachmentPoints_DC[-1],0,0),).label
    n13 = Stripped_Array.rootAssembly.instances['Diagonals-1'].nodes.getClosest((0,-
        AttachmentPoints_DC[-1],0),).label
    Stripped_Array.rootAssembly.Set(name='Membrane_Tip1', nodes=(
         Stripped_Array.rootAssembly.instances['Diagonals-1'].nodes[n10-1:n10]))
    Stripped_Array.rootAssembly.Set(name='Membrane_Tip2', nodes=(
         Stripped_Array.rootAssembly.instances['Diagonals-1'].nodes[n11-1:n11]))
    Stripped_Array.rootAssembly.Set(name='Membrane_Tip3', nodes=(
         Stripped_Array.rootAssembly.instances['Diagonals-1'].nodes[n12-1:n12]))
    Stripped_Array.rootAssembly.Set(name='Membrane_Tip4', nodes=(
         Stripped_Array.rootAssembly.instances['Diagonals-1'].nodes[n13-1:n13]))
    # (telescoping beam tip nodes)
    n14 = Stripped_Array.rootAssembly.instances['TelescopingAssembly-1'].nodes.getClosest((
        TPoints[-1][0],0,TPoints[-1][1]),).label
    n15 = Stripped_Array.rootAssembly.instances['TelescopingAssembly-1'].nodes.getClosest((0,
        TPoints[-1][0],TPoints[-1][1]),).label
    n16 = Stripped_Array.rootAssembly.instances['TelescopingAssembly-1'].nodes.getClosest((-
        TPoints[-1][0],0,TPoints[-1][1]),).label
    n17 = Stripped_Array.rootAssembly.instances['TelescopingAssembly-1'].nodes.getClosest
        ((0,-TPoints[-1][0],TPoints[-1][1]),).label
    Stripped_Array.rootAssembly.Set(name='TS_Tip1', nodes=(
         Stripped_Array.rootAssembly.instances['TelescopingAssembly-1'].nodes[n14-1:n14]))
    Stripped_Array.rootAssembly.Set(name='TS_Tip2', nodes=(
         Stripped_Array.rootAssembly.instances['TelescopingAssembly-1'].nodes[n15-1:n15]))
    Stripped_Array.rootAssembly.Set(name='TS_Tip3', nodes=(
         Stripped_Array.rootAssembly.instances['TelescopingAssembly-1'].nodes[n16-1:n16]))
    Stripped_Array.rootAssembly.Set(name='TS_Tip4', nodes=(
         Stripped_Array.rootAssembly.instances['TelescopingAssembly-1'].nodes[n17-1:n17]))
    # (membrane set)
    Stripped_Array.rootAssembly.Set(name = 'Membrane_Elements',elements=(Stripped_Array.
        rootAssembly.instances['Diagonals-1'].elements[:]+\
        Stripped_Array.rootAssembly.instances['Strips-1'].elements[:]+\
        Stripped_Array.rootAssembly.instances['Strips-1-rad-2'].elements[:]+\
        Stripped_Array.rootAssembly.instances['Strips-1-rad-3'].elements[:]+\
        Stripped_Array.rootAssembly.instances['Strips-1-rad-4'].elements[:]
    ))
    mdb.models['Quadrent'].rootAssembly.Set(edges=mdb.models['Quadrent'].rootAssembly.
        instances['Strips-1'].edges[:]+\
        mdb.models['Quadrent'].rootAssembly.instances['Strips-1-rad-2'].edges[:]+\
        mdb.models['Quadrent'].rootAssembly.instances['Strips-1-rad-3'].edges[:]+\
        mdb.models['Quadrent'].rootAssembly.instances['Strips-1-rad-4'].edges[:]+\
        mdb.models['Quadrent'].rootAssembly.instances['Diagonals-1'].edges[:]
        , name='Membrane_Edges')
    # REFERENCE POINT AT CENTER OF MEMBRANE
    Stripped_Array.rootAssembly.ReferencePoint(point=(0.0, 0.0, 0.0))
    refpointlabel = mdb.models['Quadrent'].rootAssembly.referencePoints.keys()
    # COUPLE NODES AT CENTER TO REFERENCE POINT
    mdb.models['Quadrent'].Coupling(controlPoint=Region(referencePoints=(
    mdb.models['Quadrent'].rootAssembly.referencePoints[refpointlabel[0]], )), couplingType=
    KINEMATIC, influenceRadius=WHOLE_SURFACE, localCsys=None, name='C_Center',
    surface=mdb.models['Quadrent'].rootAssembly.sets['PinnedNodes_Center'], u1=ON, u2=ON
    , u3=ON, ur1=OFF, ur2=OFF, ur3=OFF)
    #-------------------------------------------------------------------------
    #              SET UP BOUNDARY CONDITIONS, LOADS, LOAD STEPS, SLIPRINGS
    #-------------------------------------------------------------------------
    # PIN THE CENTER REFERENCE POINT
    mdb.models['Quadrent'].DisplacementBC(amplitude=UNSET, createStepName='Initial'
        , distributionType=UNIFORM, fieldName='', localCsys=None, name='Pinned',
        region=Region(referencePoints=(
        mdb.models['Quadrent'].rootAssembly.referencePoints[refpointlabel[0]], )), u1=SET, u2
            =SET
        , u3=SET, ur1=UNSET, ur2=UNSET, ur3=UNSET)
    # FIX TELESCOPING BEAM ROOTS
    Stripped_Array.EncastreBC(createStepName='Initial', localCsys=None,
        name='Fixed', region=Stripped_Array.rootAssembly.sets['FixedNode_TS'])
    # ADD NON STRUCTURAL MASS TO IDEALIZED STRIP CABLES, MAKE LOCAL CSYS FOR EACH CABLE,
        RESTRICT ROTATION
```

```
564     counter = 1
565     instancename = ['Strips-1','Strips-1-rad-2','Strips-1-rad-3','Strips-1-rad-4']
566     CSYSOrder = ((1,2),(2,3),(3,4),(4,1))
567     rf = mdb.models['Quadrent'].rootAssembly.referencePoints.keys()
568     for j in range(1,5):
569         for i in range(1,Inputs['k']+1):
570             tuplepoints = ((AttachmentPoints_S[i-1],0,0),(0,AttachmentPoints_S[i-1],0),(-
                    AttachmentPoints_S[i-1],0,0),(0,-AttachmentPoints_S[i-1],0))
571             StripLayer_TotalArea = Areas[i-1]
572             StripLayer_Mass = .5*StripLayer_TotalArea
573             StripLayer_Name = 'Strip_' + str(counter)
574             StripLayer_Set = 'Strip_Set'+str(i)
575             BC_name = 'UR1-BC-' +str(counter)
576             localname = 'LSYS'+str(counter)
577             # ADD NONSTRUCTURAL MASS TO THE MODEL
578             mdb.models['Quadrent'].rootAssembly.engineeringFeatures.NonstructuralMass(
579                 distribution=MASS_PROPORTIONAL, magnitude=StripLayer_Mass, name=
                        StripLayer_Name, region=
580                 mdb.models['Quadrent'].rootAssembly.instances[instancename[j-1]].sets[
                        StripLayer_Set]
581                 , units=TOTAL_MASS)
582             # CREATE LOCAL COORD SYSTEM FOR EACH STRIP
583             mdb.models['Quadrent'].rootAssembly.DatumCsysByThreePoints(coordSysType=
584                 CARTESIAN, name=localname, origin=
585                 mdb.models['Quadrent'].rootAssembly.instances[instancename[CSYSOrder[j
                        -1][0]-1]].vertices.findAt(
586                 tuplepoints[CSYSOrder[j-1][0]-1], ), point1=
587                 mdb.models['Quadrent'].rootAssembly.instances[instancename[CSYSOrder[j
                        -1][1]-1]].vertices.findAt(
588                 tuplepoints[CSYSOrder[j-1][1]-1], ), point2=
589                 mdb.models['Quadrent'].rootAssembly.referencePoints[rf[0]])
590             # RESTRICT UR1 FOR EACH CABLE
591             datumkey = mdb.models['Quadrent'].rootAssembly.datums.keys()
592             mdb.models['Quadrent'].DisplacementBC(amplitude=UNSET, createStepName='Initial'
593                 , distributionType=UNIFORM, fieldName='', localCsys=
594                 mdb.models['Quadrent'].rootAssembly.datums[datumkey[-1]], name=BC_name,
                        region=
595                 mdb.models['Quadrent'].rootAssembly.instances[instancename[j-1]].sets[
                        StripLayer_Set]
596                 , u1=UNSET, u2=UNSET, u3=UNSET, ur1=SET, ur2=UNSET, ur3=UNSET)
597             counter = counter+1
598     # PRETENSION STEP
599     mdb.models['Quadrent'].StaticStep(initialInc=0.001, maxInc=0.1, maxNumInc=1000,
600         name='Pretension', nlgeom=ON, previous='Initial')
601     mdb.models['Quadrent'].steps['Pretension'].setValues(adaptiveDampingRatio=0.05)
602     # CREATE PULLEYS (SLIP RING CONNECTORS)
603     # CREATE REFERENCE POINTS & POLYLINES FOR CONNECTOR CONSTRUCTION
604     # PULLEY 1
605     Stripped_Array.rootAssembly.ReferencePoint(point=(AttachmentPoints_DC[-1],0.0, 0.0))
606     Stripped_Array.rootAssembly.ReferencePoint(point=(TPoints[-1][0],0.0, TPoints[-1][1]))
607     Stripped_Array.rootAssembly.ReferencePoint(point=(TPoints[0][0]+Assembly_Constants['Ls']*
            np.cos(-theta_s),0,TPoints[0][1]+Assembly_Constants['Ls']*np.sin(-theta_s)))
608     rf = mdb.models['Quadrent'].rootAssembly.referencePoints.keys()
609     mdb.models['Quadrent'].rootAssembly.WirePolyLine(mergeType=IMPRINT, meshable=
610         OFF, points=((mdb.models['Quadrent'].rootAssembly.referencePoints[rf[2]],
611         mdb.models['Quadrent'].rootAssembly.referencePoints[rf[1]]), ))
612     mdb.models['Quadrent'].rootAssembly.WirePolyLine(mergeType=IMPRINT, meshable=
613         OFF, points=((mdb.models['Quadrent'].rootAssembly.referencePoints[rf[1]],
614         mdb.models['Quadrent'].rootAssembly.referencePoints[rf[0]]), ))
615     # apply constraints
616     mdb.models['Quadrent'].Coupling(controlPoint=
617         mdb.models['Quadrent'].rootAssembly.sets['Membrane_Tip1'], couplingType=KINEMATIC
618         , influenceRadius=WHOLE_SURFACE, localCsys=None, name='SR_C_1',
619         surface=Region(referencePoints=(
620         mdb.models['Quadrent'].rootAssembly.referencePoints[rf[2]], )), u1=ON, u2=ON,
621         u3=ON, ur1=ON, ur2=ON, ur3=ON)
622     mdb.models['Quadrent'].Coupling(controlPoint=
623         mdb.models['Quadrent'].rootAssembly.sets['TS_Tip1'], couplingType=KINEMATIC
624         , influenceRadius=WHOLE_SURFACE, localCsys=None, name='SR_C_2',
625         surface=Region(referencePoints=(
626         mdb.models['Quadrent'].rootAssembly.referencePoints[rf[1]], )), u1=ON, u2=ON,
```

```
627         u3=ON, ur1=ON, ur2=ON, ur3=ON)
628     # apply material flow bc
629     mdb.models['Quadrent'].MaterialFlowBC(createStepName='Initial',
630     distributionType=UNIFORM, fieldName='', magnitude=0.0, name='MF_1',
631     region=Region(referencePoints=(
632     mdb.models['Quadrent'].rootAssembly.referencePoints[rf[2]], )))
633     # create additional slipring
634     Stripped_Array.rootAssembly.ReferencePoint(point=(TPoints[0][0],0,TPoints[0][1]+
            Assembly_Constants['Ls']*np.sin(-theta_s)))
635     rf = mdb.models['Quadrent'].rootAssembly.referencePoints.keys()
636     mdb.models['Quadrent'].rootAssembly.WirePolyLine(mergeType=IMPRINT, meshable=
637         OFF, points=((mdb.models['Quadrent'].rootAssembly.referencePoints[rf[1]],
638         mdb.models['Quadrent'].rootAssembly.referencePoints[rf[0]]), ))
639     mdb.models['Quadrent'].MaterialFlowBC(createStepName='Initial',
640     distributionType=UNIFORM, fieldName='', magnitude=0.0, name='MF_2',
641     region=Region(referencePoints=(
642     mdb.models['Quadrent'].rootAssembly.referencePoints[rf[0]], )))
643     # create displacement boundary conditions at sliprings
644     mdb.models['Quadrent'].DisplacementBC(amplitude=UNSET, createStepName=
645         'Pretension', distributionType=UNIFORM, fieldName='', fixed=OFF, localCsys=
646         None, name='Stroke_1', region=Region(referencePoints=(
647         mdb.models['Quadrent'].rootAssembly.referencePoints[rf[0]], )), u1=-Inputs['Stroke'],
                u2=
648         0.0, u3=0.0, ur1=0.0, ur2=0.0, ur3=0.0)
649     # PULLEY 2
650     Stripped_Array.rootAssembly.ReferencePoint(point=(0.0,AttachmentPoints_DC[-1], 0.0))
651     Stripped_Array.rootAssembly.ReferencePoint(point=(0.0,TPoints[-1][0], TPoints[-1][1]))
652     Stripped_Array.rootAssembly.ReferencePoint(point=(0,TPoints[0][0]+Assembly_Constants['Ls'
            ]*np.cos(-theta_s),TPoints[0][1]+Assembly_Constants['Ls']*np.sin(-theta_s)))
653     rf = mdb.models['Quadrent'].rootAssembly.referencePoints.keys()
654     mdb.models['Quadrent'].rootAssembly.WirePolyLine(mergeType=IMPRINT, meshable=
655         OFF, points=((mdb.models['Quadrent'].rootAssembly.referencePoints[rf[2]],
656         mdb.models['Quadrent'].rootAssembly.referencePoints[rf[1]]), ))
657     mdb.models['Quadrent'].rootAssembly.WirePolyLine(mergeType=IMPRINT, meshable=
658         OFF, points=((mdb.models['Quadrent'].rootAssembly.referencePoints[rf[1]],
659         mdb.models['Quadrent'].rootAssembly.referencePoints[rf[0]]), ))
660     # apply constraints
661     mdb.models['Quadrent'].Coupling(controlPoint=
662         mdb.models['Quadrent'].rootAssembly.sets['Membrane_Tip2'], couplingType=KINEMATIC
663         , influenceRadius=WHOLE_SURFACE, localCsys=None, name='SR_C_3',
664         surface=Region(referencePoints=(
665         mdb.models['Quadrent'].rootAssembly.referencePoints[rf[2]], )), u1=ON, u2=ON,
666         u3=ON, ur1=ON, ur2=ON, ur3=ON)
667     mdb.models['Quadrent'].Coupling(controlPoint=
668         mdb.models['Quadrent'].rootAssembly.sets['TS_Tip2'], couplingType=KINEMATIC
669         , influenceRadius=WHOLE_SURFACE, localCsys=None, name='SR_C_4',
670         surface=Region(referencePoints=(
671         mdb.models['Quadrent'].rootAssembly.referencePoints[rf[1]], )), u1=ON, u2=ON,
672         u3=ON, ur1=ON, ur2=ON, ur3=ON)
673     # apply material flow bc
674     mdb.models['Quadrent'].MaterialFlowBC(createStepName='Initial',
675     distributionType=UNIFORM, fieldName='', magnitude=0.0, name='MF_3',
676     region=Region(referencePoints=(
677     mdb.models['Quadrent'].rootAssembly.referencePoints[rf[2]], )))
678     # create additional slipring
679     Stripped_Array.rootAssembly.ReferencePoint(point=(0,TPoints[0][0],TPoints[0][1]+
            Assembly_Constants['Ls']*np.sin(-theta_s)))
680     rf = mdb.models['Quadrent'].rootAssembly.referencePoints.keys()
681     mdb.models['Quadrent'].rootAssembly.WirePolyLine(mergeType=IMPRINT, meshable=
682         OFF, points=((mdb.models['Quadrent'].rootAssembly.referencePoints[rf[1]],
683         mdb.models['Quadrent'].rootAssembly.referencePoints[rf[0]]), ))
684     mdb.models['Quadrent'].MaterialFlowBC(createStepName='Initial',
685     distributionType=UNIFORM, fieldName='', magnitude=0.0, name='MF_4',
686     region=Region(referencePoints=(
687     mdb.models['Quadrent'].rootAssembly.referencePoints[rf[0]], )))
688     # create displacement boundary conditions at sliprings
689     mdb.models['Quadrent'].DisplacementBC(amplitude=UNSET, createStepName=
690         'Pretension', distributionType=UNIFORM, fieldName='', fixed=OFF, localCsys=
691         None, name='Stroke_2', region=Region(referencePoints=(
692         mdb.models['Quadrent'].rootAssembly.referencePoints[rf[0]], )), u1=0, u2=
693         -Inputs['Stroke'], u3=0.0, ur1=0.0, ur2=0.0, ur3=0.0)
```

```
    # PULLEY 3
    Stripped_Array.rootAssembly.ReferencePoint(point=(-AttachmentPoints_DC[-1],0.0, 0.0))
    Stripped_Array.rootAssembly.ReferencePoint(point=(-TPoints[-1][0],0.0, TPoints[-1][1]))
    Stripped_Array.rootAssembly.ReferencePoint(point=(-TPoints[0][0]-Assembly_Constants['Ls'
        ]*np.cos(-theta_s),0,TPoints[0][1]+Assembly_Constants['Ls']*np.sin(-theta_s)))
    rf = mdb.models['Quadrent'].rootAssembly.referencePoints.keys()
    mdb.models['Quadrent'].rootAssembly.WirePolyLine(mergeType=IMPRINT, meshable=
        OFF, points=((mdb.models['Quadrent'].rootAssembly.referencePoints[rf[2]],
        mdb.models['Quadrent'].rootAssembly.referencePoints[rf[1]]), ))
    mdb.models['Quadrent'].rootAssembly.WirePolyLine(mergeType=IMPRINT, meshable=
        OFF, points=((mdb.models['Quadrent'].rootAssembly.referencePoints[rf[1]],
        mdb.models['Quadrent'].rootAssembly.referencePoints[rf[0]]), ))
    # apply constraints
    mdb.models['Quadrent'].Coupling(controlPoint=
        mdb.models['Quadrent'].rootAssembly.sets['Membrane_Tip3'], couplingType=KINEMATIC
        , influenceRadius=WHOLE_SURFACE, localCsys=None, name='SR_C_5',
        surface=Region(referencePoints=(
        mdb.models['Quadrent'].rootAssembly.referencePoints[rf[2]], )), u1=ON, u2=ON,
        u3=ON, ur1=ON, ur2=ON, ur3=ON)
    mdb.models['Quadrent'].Coupling(controlPoint=
        mdb.models['Quadrent'].rootAssembly.sets['TS_Tip3'], couplingType=KINEMATIC
        , influenceRadius=WHOLE_SURFACE, localCsys=None, name='SR_C_6',
        surface=Region(referencePoints=(
        mdb.models['Quadrent'].rootAssembly.referencePoints[rf[1]], )), u1=ON, u2=ON,
        u3=ON, ur1=ON, ur2=ON, ur3=ON)
    # apply material flow bc
    mdb.models['Quadrent'].MaterialFlowBC(createStepName='Initial',
    distributionType=UNIFORM, fieldName='', magnitude=0.0, name='MF_5',
    region=Region(referencePoints=(
    mdb.models['Quadrent'].rootAssembly.referencePoints[rf[2]], )))
    # create additional slipring
    Stripped_Array.rootAssembly.ReferencePoint(point=(-TPoints[0][0],0,TPoints[0][1]+
        Assembly_Constants['Ls']*np.sin(-theta_s)))
    rf = mdb.models['Quadrent'].rootAssembly.referencePoints.keys()
    mdb.models['Quadrent'].rootAssembly.WirePolyLine(mergeType=IMPRINT, meshable=
        OFF, points=((mdb.models['Quadrent'].rootAssembly.referencePoints[rf[1]],
        mdb.models['Quadrent'].rootAssembly.referencePoints[rf[0]]), ))
    mdb.models['Quadrent'].MaterialFlowBC(createStepName='Initial',
    distributionType=UNIFORM, fieldName='', magnitude=0.0, name='MF_6',
    region=Region(referencePoints=(
    mdb.models['Quadrent'].rootAssembly.referencePoints[rf[0]], )))
    # create displacement boundary conditions at sliprings
    mdb.models['Quadrent'].DisplacementBC(amplitude=UNSET, createStepName=
        'Pretension', distributionType=UNIFORM, fieldName='', fixed=OFF, localCsys=
        None, name='Stroke_3', region=Region(referencePoints=(
        mdb.models['Quadrent'].rootAssembly.referencePoints[rf[0]], )), u1=Inputs['Stroke'],
            u2=
        0, u3=0.0, ur1=0.0, ur2=0.0, ur3=0.0)
    # PULLEY 4
    Stripped_Array.rootAssembly.ReferencePoint(point=(0.0,-AttachmentPoints_DC[-1], 0.0))
    Stripped_Array.rootAssembly.ReferencePoint(point=(0.0,-TPoints[-1][0], TPoints[-1][1]))
    Stripped_Array.rootAssembly.ReferencePoint(point=(0,-TPoints[0][0]-Assembly_Constants['Ls
        ']*np.cos(-theta_s),TPoints[0][1]+Assembly_Constants['Ls']*np.sin(-theta_s)))
    rf = mdb.models['Quadrent'].rootAssembly.referencePoints.keys()
    mdb.models['Quadrent'].rootAssembly.WirePolyLine(mergeType=IMPRINT, meshable=
        OFF, points=((mdb.models['Quadrent'].rootAssembly.referencePoints[rf[2]],
        mdb.models['Quadrent'].rootAssembly.referencePoints[rf[1]]), ))
    mdb.models['Quadrent'].rootAssembly.WirePolyLine(mergeType=IMPRINT, meshable=
        OFF, points=((mdb.models['Quadrent'].rootAssembly.referencePoints[rf[1]],
        mdb.models['Quadrent'].rootAssembly.referencePoints[rf[0]]), ))
    # apply constraints
    mdb.models['Quadrent'].Coupling(controlPoint=
        mdb.models['Quadrent'].rootAssembly.sets['Membrane_Tip4'], couplingType=KINEMATIC
        , influenceRadius=WHOLE_SURFACE, localCsys=None, name='SR_C_7',
        surface=Region(referencePoints=(
        mdb.models['Quadrent'].rootAssembly.referencePoints[rf[2]], )), u1=ON, u2=ON,
        u3=ON, ur1=ON, ur2=ON, ur3=ON)
    mdb.models['Quadrent'].Coupling(controlPoint=
        mdb.models['Quadrent'].rootAssembly.sets['TS_Tip4'], couplingType=KINEMATIC
        , influenceRadius=WHOLE_SURFACE, localCsys=None, name='SR_C_8',
        surface=Region(referencePoints=(
```

```
761         mdb.models['Quadrent'].rootAssembly.referencePoints[rf[1]], )), u1=ON, u2=ON,
762         u3=ON, ur1=ON, ur2=ON, ur3=ON)
763     # apply material flow bc
764     mdb.models['Quadrent'].MaterialFlowBC(createStepName='Initial',
765     distributionType=UNIFORM, fieldName='', magnitude=0.0, name='MF_7',
766     region=Region(referencePoints=(
767     mdb.models['Quadrent'].rootAssembly.referencePoints[rf[2]], )))
768     # create additional slipring
769     Stripped_Array.rootAssembly.ReferencePoint(point=(0,-TPoints[0][0],TPoints[0][1]+
            Assembly_Constants['Ls']*np.sin(-theta_s)))
770     rf = mdb.models['Quadrent'].rootAssembly.referencePoints.keys()
771     mdb.models['Quadrent'].rootAssembly.WirePolyLine(mergeType=IMPRINT, meshable=
772         OFF, points=((mdb.models['Quadrent'].rootAssembly.referencePoints[rf[1]],
773         mdb.models['Quadrent'].rootAssembly.referencePoints[rf[0]]), ))
774     mdb.models['Quadrent'].MaterialFlowBC(createStepName='Initial',
775     distributionType=UNIFORM, fieldName='', magnitude=0.0, name='MF_8',
776     region=Region(referencePoints=(
777     mdb.models['Quadrent'].rootAssembly.referencePoints[rf[0]], )))
778     # create displacement boundary conditions at sliprings
779     mdb.models['Quadrent'].DisplacementBC(amplitude=UNSET, createStepName=
780         'Pretension', distributionType=UNIFORM, fieldName='', fixed=OFF, localCsys=
781         None, name='Stroke_4', region=Region(referencePoints=(
782         mdb.models['Quadrent'].rootAssembly.referencePoints[rf[0]], )), u1=0, u2=
783         Inputs['Stroke'], u3=0, ur1=0.0, ur2=0.0, ur3=0.0)
784     # DEFINE SLIP RING ELEMENT SECTION
785     SlipRing_Length = Length_Pulley+Assembly_Constants['p_tip']+(Assembly_Constants['Ls']*np.
            cos(-theta_s))
786     SlipRing_D11 = Assembly_Constants['E']* np.pi*Assembly_Constants['SlipRing_r']**2/
            SlipRing_Length
787     SlipRing_ML = np.pi*Assembly_Constants['SlipRing_r']**2*Assembly_Constants['rho']
788     mdb.models['Quadrent'].ConnectorSection(assembledType=SLIPRING, contactAngle = 18,
            massPerLength=
789     SlipRing_ML, name='SlipRingConnector')
790     mdb.models['Quadrent'].sections['SlipRingConnector'].setValues(behaviorOptions=
791         (ConnectorElasticity(table=((SlipRing_D11, ), ), independentComponents=(),
792         components=(1, )), ))
793     mdb.models['Quadrent'].sections['SlipRingConnector'].behaviorOptions[0].ConnectorOptions
            ()
794     # APPLY SLIP RING SECTION TO ALL WIRES
795     mdb.models['Quadrent'].rootAssembly.SectionAssignment(region=Region(
796     edges=mdb.models['Quadrent'].rootAssembly.edges[:]),
797     sectionName='SlipRingConnector')
798     # ADD STABILIZATION TO PRETENSIONING STEP
799     mdb.models['Quadrent'].steps['Pretension'].setValues(adaptiveDampingRatio=None,
800         continueDampingFactors=False, stabilizationMagnitude=0.0002,
801         stabilizationMethod=DAMPING_FACTOR)
802     # CREATE SET FOR CONNECTOR ELEMENTS
803     mdb.models['Quadrent'].rootAssembly.Set(edges=
804     mdb.models['Quadrent'].rootAssembly.edges[:], name=
805     'SlipRings')
806     # APPLY FIXED BOUNDARY CONDITIONS TO RELEVANT SLIPRINGS
807     mdb.models['Quadrent'].rootAssembly.Set(name='Fixed_SlipRings', referencePoints=(
808         mdb.models['Quadrent'].rootAssembly.referencePoints[401],
809         mdb.models['Quadrent'].rootAssembly.referencePoints[413],
810         mdb.models['Quadrent'].rootAssembly.referencePoints[425],
811         mdb.models['Quadrent'].rootAssembly.referencePoints[437]))
812     mdb.models['Quadrent'].EncastreBC(createStepName='Initial', localCsys=None,
813         name='BC-105', region=mdb.models['Quadrent'].rootAssembly.sets['Fixed_SlipRings'])
814     # CREATE GRAVITY LOAD STEP AND APPLY GRAVITY
815     Stripped_Array.StaticStep(initialInc=0.001, maxInc=.1,maxNumInc=1000, name=
816         'Gravity', previous='Pretension',nlgeom=ON)
817     mdb.models['Quadrent'].steps['Gravity'].setValues(adaptiveDampingRatio=0.05)
818     Stripped_Array.Gravity(comp3=(-Assembly_Constants['EarthGravity']*Inputs['%_EarthGravity'
            ]), createStepName='Gravity',
819         distributionType=UNIFORM, field='', name='Gravity')
820     mdb.models['Quadrent'].loads['Gravity'].setValues(distributionType=UNIFORM,
821         field='', region=mdb.models['Quadrent'].rootAssembly.sets['Membrane_Edges'])
822     mdb.models['Quadrent'].Gravity(comp3=-Assembly_Constants['EarthGravity']*Inputs['%
            _EarthGravity'], createStepName='Gravity',
823     distributionType=UNIFORM, field='', name='TSGravity', region=
824     mdb.models['Quadrent'].rootAssembly.instances['TelescopingAssembly-1'].sets['ALL'])
```

```
825     # PRINT THE TOTAL NUMBER OF ELEMENTS IN THE ASSEMBLY
826     instances_keys = mdb.models['Quadrent'].rootAssembly.instances.keys()
827     AssemblyElements = 0
828     for key in instances_keys:
829         temp  = len(mdb.models['Quadrent'].rootAssembly.instances[key].elements[:])
830         AssemblyElements = AssemblyElements +temp
831     print('Model Element Count: ' + str(AssemblyElements))
832     # CREATE FIELD OUTPUT REQUESTS
833     import step
834     # forces and moments for telescoping beams
835     Stripped_Array.FieldOutputRequest(name = 'Forces_Moments', createStepName = 'Gravity',
            region=
836         mdb.models['Quadrent'].rootAssembly.sets['TelescopingAssembly-1.ALL'], sectionPoints=
837         DEFAULT,variables = ('SF',))
838     # slip ring forces and strains
839     mdb.models['Quadrent'].FieldOutputRequest(createStepName='Gravity', name=
840         'Slipring_Forces_Strains', rebar=EXCLUDE, region=
841         mdb.models['Quadrent'].rootAssembly.sets['SlipRings'], sectionPoints=
842         DEFAULT, variables=('CTF', 'CU'))
843     # CREATE MODAL ANALYSIS STEP
844     mdb.models['Quadrent'].FrequencyStep( name='Modal', numEigen=Freq_Number,previous=
845     'Gravity')
846     # CREATE THE JOB
847     import job
848     job_name = 'Stripped_Array_'+ str(ID)
849     mdb.Job(atTime=None, contactPrint=OFF, description='', echoPrint=OFF,
850         explicitPrecision=SINGLE, getMemoryFromAnalysis=True, historyPrint=OFF,
851         memory=90, memoryUnits=PERCENTAGE, model='Quadrent', modelPrint=OFF,
852         multiprocessingMode=DEFAULT, name=job_name, nodalOutputPrecision=SINGLE,
853         numCpus=1, numGPUs=0, queue=None, resultsFormat=ODB, scratch='', type=
854         ANALYSIS, userSubroutine='', waitHours=0, waitMinutes=0)
855     # MONITOR THE JOB
856     warnings,errors = [],[]
857     def onMessage(jobName, messageType, data,userData):
858         if (messageType==WARNING):
859             messageText = getattr(data, 'message')
860             warnings.append(messageText)
861         if((messageType==ERROR)):
862             messageText = getattr(data, 'message')
863             errors.append(messageText)
864     monitorManager.addMessageCallback(jobName=job_name, messageType=ANY_MESSAGE_TYPE,
            callback=onMessage,userData=None)
865     # WAIT TILL JOB IS DONE
866     mdb.jobs[job_name].submit(consistencyChecking=OFF)
867     mdb.jobs[job_name].waitForCompletion()
868     # REMOVE JOB MONITOR
869     monitorManager.removeMessageCallback(jobName=job_name, messageType=ANY_MESSAGE_TYPE,
            callback=onMessage,userData=None)
870     # REMOVE DUPLICATE WARNINGS AND ERRORS
871     warnings = list(set(warnings))
872     errors = list(set(errors))
873     # DETERMINE IF JOB ABORTED OR NOT
874     if len(errors) > 0:
875         errorsflag = 'True'
876         # FLAG JOB FOR WARNINGS
877         if len(warnings) > 2:
878             warningsflag = 'True'
879         else:
880             warningsflag = 'False'
881         #----------------------------
882         # RESULTS (ASSEMBLY)
883         #----------------------------
884         Assembly_Results = dict([
885             ('ID',ID),
886             ('Error',errorsflag),
887             ('Warning',warningsflag),
888             ('Area',Configuration[0]),
889             ('k',Configuration[1]),
890             ('t_ts',Configuration[2]),
891             ('r_ts',Configuration[3]),
892             ('Stroke',Configuration[4]),
```

```
893             ('t_s_max',Configuration[5]),
894             ('r_dc',Configuration[6]),
895             ('%_EarthGravity',Configuration[7]),
896             ('Strip_Radii',Strip_Radii[0])
897         ])
898         # ORDER DICTIONARY
899         Assembly_Results_List = ['ID','Error','Warning','Area','k','t_ts','r_ts','Stroke','
                t_s_max','r_dc','%_EarthGravity','Strip_Radii']
900         from collections import OrderedDict
901         Assembly_Results=OrderedDict(sorted(Assembly_Results.items(), key=lambda pair:
                Assembly_Results_List.index(pair[0])))
902         # REMOVE ODB FROM ABAQUS SESSION
903         Results = session.openOdb(name=job_name +'.odb')
904         Results.close()
905     else:
906         errorsflag = 'False'
907         # FLAG JOB IF SOLUTION WAS ACHIEVED BUT WITH WARNINGS
908         if len(warnings) > 1:
909             warningsflag = 'True'
910         else:
911             warningsflag = 'False'
912         #--------------------------------------------------------------------
913         #              POST PROCESS RESULTS
914         #--------------------------------------------------------------------
915         # IMPORT PACKAGES, BOOK KEEPING, % BRING IN RESULTS
916         import displayGroupOdbToolset
917         import visualization
918         import odbAccess
919         Results = session.openOdb(name=job_name +'.odb')
920         Results_Viewport = session.viewports['Viewport: 1']
921         Results_Viewport.setValues(displayedObject=Results)
922         # get step keys
923         Stepnames = Results.steps.keys()
924         #--------------------------------------------------------------------STRESSES &
                DEFLECTIONS
925         # get increment number and step number
926         lastframe = int(len(Results.steps["Gravity"].frames)-1)
927         Results_Viewport.odbDisplay.setFrame(step = 1,frame = lastframe)
928         Stepnumber = int(Results_Viewport.odbDisplay.fieldFrame[0])
929         frame = int(Results_Viewport.odbDisplay.fieldFrame[1])
930         #----------------------------
931         # RESULTS (TELESCOPING BEAMS)
932         #----------------------------
933         #isolate telescopic supports
934         TB_Display = displayGroupOdbToolset.LeafFromElementSets(elementSets=('
                TELESCOPINGASSEMBLY-1.ALL', ))
935         session.viewports['Viewport: 1'].odbDisplay.displayGroup.intersect(leaf=TB_Display)
936         # STRESS,MOMENTS,AND FORCES
937         report_name_and_path = 'TelescopingSupports_VonMisesStress.rpt'
938         report_name_and_path2 = 'TelescopingSupports_S11.rpt'
939         report_name_and_path3 = 'TelescopingSupports_SF_SF1.rpt'
940         report_name_and_path4 = 'TelescopingSupports_SM_SM2.rpt'
941         report_name_and_path5 = 'TelescopingSupports_U.rpt'
942         WriteResults(report_name_and_path,'S.Mises',Results,Stepnumber,frame,
                INTEGRATION_POINT,(('S', INTEGRATION_POINT, ((INVARIANT, 'Mises'),)), ))
943         WriteResults(report_name_and_path2,'S.S11',Results,Stepnumber,frame,INTEGRATION_POINT
                ,(('S', INTEGRATION_POINT, ((COMPONENT, 'S11'),)), ))
944         WriteResults(report_name_and_path3,'SF.SF1',Results,Stepnumber,frame,NODAL,(('SF',
                INTEGRATION_POINT , ((COMPONENT, 'SF1'),)), ))
945         WriteResults(report_name_and_path4,'SM.SM2',Results,Stepnumber,frame,NODAL,(('SM',
                INTEGRATION_POINT  , ((COMPONENT, 'SM2'),)), ))
946         # DEFLECTIONS
947         #remove elements from displayGroup
948         TB_Display = displayGroupOdbToolset.Leaf(leafType=ALL_ELEMENTS)
949         session.viewports['Viewport: 1'].odbDisplay.displayGroup.remove(leaf=TB_Display)
950         #isolate the telescoping booms again
951         TB_Display = displayGroupOdbToolset.LeafFromNodeSets(nodeSets=('TS_TIP1','TS_TIP2','
                TS_TIP3','TS_TIP4',))
952         session.viewports['Viewport: 1'].odbDisplay.displayGroup.add(leaf=TB_Display)
953         #tip deflections
954         WriteResults(report_name_and_path5,'U',Results,Stepnumber,frame,NODAL,(('U', NODAL,((
```

```
            COMPONENT,'U1'),(COMPONENT,'U2'),(COMPONENT,'U3'),)),))
        #begin extracting data from reports
        #VonMises/%VonMises
        VMmin,VMmax = ExtractResults(report_name_and_path)
        max_stress,min_stress = np.max(VMmax),np.min(VMmin)
        #Axial Stress (Full Telescoping Beam)
        S11min,S11max = ExtractResults(report_name_and_path2)
        min_S11,max_S11 = np.min(S11min),np.max(S11max)
        #Axial Forces (Telescoping Section)
        SF1min,SF1max = ExtractResults(report_name_and_path3)
        min_SF_SF1,max_SF_SF1 = np.min(SF1min),np.max(SF1max)
        #Moment (Telescoping Section)
        #find location of largest moment
        extracted_line=''
        minData,maxData,maxNode,minNode = [],[],[],[]
        collectionFlag = False
        f = open(report_name_and_path4)
        for line in f:
            str1=line
            if 'Maximum' in str1:
                extracted_line = str1
                extracted_list = extracted_line.split()
                extracted_list = extracted_list[1:]
                for item in range(1,len(extracted_list)+1):
                    value = float(extracted_list[item-1])
                    maxData.append(value)
                    collectionFlag = True
            if 'At Node' in str1 and collectionFlag is True:
                extracted_line = str1
                extracted_list = extracted_line.split()
                extracted_list = extracted_list[2:]
                for item in range(1,len(extracted_list)+1):
                    value = int(extracted_list[item-1])
                    maxNode.append(value)
                collectionFlag = False
        maxvalidx1 = maxData.index(min(maxData))
        maxvalidx2 = maxData.index(max(maxData))
        if abs(min(maxData))>abs(max(maxData)):
            maxvalidx = maxvalidx1
        else:
            maxvalidx = maxvalidx2
        f.close()
        extracted_line=''
        collectionFlag = False
        f = open(report_name_and_path4)
        for line in f:
            str1=line
            if 'Minimum' in str1:
                extracted_line = str1
                extracted_list = extracted_line.split()
                extracted_list = extracted_list[1:]
                for item in range(1,len(extracted_list)+1):
                    value = float(extracted_list[item-1])
                    minData.append(value)
                    collectionFlag = True
            if 'At Node' in str1 and collectionFlag is True:
                extracted_line = str1
                extracted_list = extracted_line.split()
                extracted_list = extracted_list[2:]
                for item in range(1,len(extracted_list)+1):
                    value = int(extracted_list[item-1])
                    minNode.append(value)
                collectionFlag = False
        minvalidx1 = minData.index(min(minData))
        minvalidx2 = minData.index(max(minData))
        if abs(min(minData))>abs(max(minData)):
            minvalidx = minvalidx1
        else:
            minvalidx = minvalidx2
        #compare which is larger
        if abs(minData[minvalidx])>abs(maxData[maxvalidx]):
```

```
            MomentValue = minData[minvalidx]
            MomentNode = minNode[minvalidx]
        else:
            MomentValue = maxData[maxvalidx]
            MomentNode = maxNode[maxvalidx]
        f.close()
        # get section of node
        node = mdb.models['Quadrent'].rootAssembly.instances['TelescopingAssembly-1'].nodes[
            MomentNode:MomentNode+1]
        node_coord = abs(node[0].coordinates[0])
        if node_coord == 0:
            node_coord = abs(node[0].coordinates[1])
        count = 0
        for point in TPoints:
            x = point[0]
            if count == 0:
                next
            else:
                if x < node_coord:
                    count += 1
                    next
                else:
                    break
        #report max and min moments
        min_SM_SM2,max_SM_SM2  = minData[minvalidx],maxData[maxvalidx]
        # get S11 at location of max moment
        f = open(report_name_and_path3)
        for line in f:
            extracted_line = line
            extracted_list = extracted_line.split()
            extracted_list = extracted_list[0:]
            if str(MomentNode) in extracted_list:
                s11value = float(extracted_list[1])
                break
        f.close()
        #Displacement (Telescoping Beam Tip Nodes)
        Umin,Umax = ExtractResults(report_name_and_path5)
        # FAILURE MODES
        # Euler Buckling
        I = np.pi/4*(TRadii[-1]**4-(TRadii[-1]-TThicknesses[-1])**4)
        EBLoad = np.pi**2*Assembly_Constants['E_ts']*I/(2*TS_L)**2/(np.pi*(TRadii[-1]**2-(
            TRadii[-1]-TThicknesses[-1])**2))/2 #with safety factor of 2
        # Local Shell Buckling
        r_lsb,t_lsb = TRadii[count],TThicknesses[0] #cross-sectional parameters used for
            local shell buckling of the root telescoping section
        phi_lsb = -1*math.sqrt(r_lsb/t_lsb)/16
        yc = 1-0.901*(1-math.exp(phi_lsb)) #axial compression knockdown factor
        area_lsb = np.pi*(r_lsb**2 - (r_lsb-t_lsb)**2) #area of root telescoping shell
        Pcr_lsb = yc*Assembly_Constants['E_ts']/(math.sqrt(3*(1-Assembly_Constants['v_ts'
            ]**2)))*(t_lsb/r_lsb)*area_lsb #critical axial local buckling load
        yb = float(1-0.731*(1-math.exp(phi_lsb))) #bending knockdown factor
        Mcr_lsb = yb*Assembly_Constants['E_ts']*np.pi*r_lsb*t_lsb**2/(math.sqrt(3*(1-
            Assembly_Constants['v_ts']**2)))
        P_lsb = s11value
        M_lsb = MomentValue
        M_lsb_ratio = abs(M_lsb)/Mcr_lsb
        P_lsb_ratio = abs(P_lsb)/Pcr_lsb
        LocalShellBuckling = M_lsb_ratio+P_lsb_ratio
        #----------------------------
        # RESULTS (STRIPS)
        #----------------------------
        #isolate the strips
        TB_Root_Display = displayGroupOdbToolset.LeafFromElementSets(elementSets=('
            MEMBRANE_ELEMENTS',))
        session.viewports['Viewport: 1'].odbDisplay.displayGroup.replace(leaf=TB_Root_Display
            )
        TB_Root_Display2 = displayGroupOdbToolset.LeafFromOdbElementMaterials(
            elementMaterials=('VECTRAN_SUPPORT',))
        session.viewports['Viewport: 1'].odbDisplay.displayGroup.intersect(leaf=
            TB_Root_Display2)
        # STRESSES AND DEFLECTIONS
```

```
1087        report_name_and_path2 = 'Strips_S11.rpt'
1088        report_name_and_path3 = 'Strips_U.rpt'
1089        report_name_and_path4 = 'Strips_RelativeU.rpt'
1090        report_name_and_path5 = 'Strips_newlength.rpt'
1091        # WRITE REPORTS
1092        WriteResults(report_name_and_path2,'S.S11',Results,Stepnumber,frame,INTEGRATION_POINT
                ,(('S', INTEGRATION_POINT, ((COMPONENT, 'S11'),)), ))
1093        #remove elements from displayGroup
1094        TB_Display = displayGroupOdbToolset.Leaf(leafType=ALL_ELEMENTS)
1095        session.viewports['Viewport: 1'].odbDisplay.displayGroup.remove(leaf=TB_Display)
1096        #isolate the strips again
1097        TB_Root_Display = displayGroupOdbToolset.LeafFromElementSets(elementSets=('
                MEMBRANE_ELEMENTS',))
1098        session.viewports['Viewport: 1'].odbDisplay.displayGroup.replace(leaf=TB_Root_Display
                )
1099        TB_Root_Display3 = displayGroupOdbToolset.LeafFromOdbElementMaterials(
                elementMaterials=('VECTRAN_SUPPORT',))
1100        session.viewports['Viewport: 1'].odbDisplay.displayGroup.intersect(leaf=
                TB_Root_Display3)
1101        #tip deflections
1102        WriteResults(report_name_and_path3,'U',Results,Stepnumber,frame,NODAL,(('U', NODAL,((
                COMPONENT,'U1'),(COMPONENT,'U2'),(COMPONENT,'U3'),)),))
1103        #EXTRACT DATA
1104        #Axial Stress
1105        S11minSB,S11maxSB = ExtractResults(report_name_and_path2)
1106        min_S11SB,max_S11SB = np.min(S11minSB),np.max(S11maxSB)
1107        #Displacement
1108        UminSB,UmaxSB = ExtractResults(report_name_and_path3)
1109        #Get relative displacements per strip
1110        U3_SB = dict([])
1111        U3_SB_List = []
1112        for i in range(1,Inputs['k']+1):
1113            name = 'STRIPS-1.STRIP_SET'+str(i)
1114            lowb,highb,deltab,slength,nlength = 'Strip_'+str(i)+'_low_b','Strip_'+str(i)+'
                    _high_b','Strip_'+str(i)+'_deltab','Strip_'+str(i)+'_Length','Strip_'+str(i)+
                    '_LengthNew'
1115            U3_SB_List.append(lowb)
1116            U3_SB_List.append(highb)
1117            U3_SB_List.append(deltab)
1118            U3_SB_List.append(slength)
1119            TB_Root_Display = displayGroupOdbToolset.LeafFromElementSets(elementSets=(name,))
1120            session.viewports['Viewport: 1'].odbDisplay.displayGroup.replace(leaf=
                    TB_Root_Display)
1121            WriteResults(report_name_and_path4,'U',Results,Stepnumber,frame,NODAL,(('U',
                    NODAL,((COMPONENT,'U3'),)),))
1122            U3minSB,U3maxSB = ExtractResults(report_name_and_path4)
1123            min_U3SB,max_U3SB = np.min(U3minSB),np.max(U3maxSB)
1124            U3_SB[lowb] = min_U3SB
1125            U3_SB[highb] = max_U3SB
1126            U3_SB[deltab] = abs(max_U3SB-min_U3SB)
1127            U3_SB[slength] = Strip_Lengths[i-1]
1128            #sag angle
1129            nodesetame = 'Strips-a-Set'+str(i)
1130            node_org = mdb.models['Quadrent'].rootAssembly.sets[nodesetame].nodes
1131            node_org_label,node_org_coordinates = node_org[0].label,node_org[0].coordinates
1132            finalnodepos=[node_org_coordinates[0],node_org_coordinates[1],
                    node_org_coordinates[2]]
1133            WriteResults(report_name_and_path5,'U',Results,Stepnumber,frame,NODAL,(('U',
                    NODAL,((COMPONENT,'U1'),(COMPONENT,'U2'),(COMPONENT,'U3'),)),))
1134            extracted_line=''
1135            f = open(report_name_and_path5)
1136            Flag= False
1137            values = []
1138            counter = 1
1139            for line in f:
1140                if counter>19:
1141                    str1=line
1142                    if str(node_org_label) in str1:
1143                        extracted_line = str1
1144                        extracted_list = extracted_line.split()
1145                        extracted_list = extracted_list[1:]
```

```
                        for j in range(0,3):
                            finalnodepos[j] = finalnodepos[j]+float("{:.8f}".format(float(
                                extracted_list[j])))
                        break
                counter +=1
            f.close()
            #compute length between strip supports
            x1,y1,z1,x2,y2,z2 = finalnodepos[0],finalnodepos[1],finalnodepos[2],finalnodepos
                [1],finalnodepos[0],finalnodepos[2]
            newlength = math.sqrt((x2-x1)**2 + (y2-y1)**2 + (z2-z1)**2)
            U3_SB_List.append(nlength)
            U3_SB[nlength] = newlength
        #----------------------------
        # RESULTS (DIAGONALS)
        #----------------------------
        #isolate the diagonals
        TB_Root_Display = displayGroupOdbToolset.LeafFromElementSets(elementSets=('
            MEMBRANE_ELEMENTS',))
        session.viewports['Viewport: 1'].odbDisplay.displayGroup.replace(leaf=TB_Root_Display
            )
        TB_Root_Display2 = displayGroupOdbToolset.LeafFromOdbElementMaterials(
            elementMaterials=('DYNEEMA',))
        session.viewports['Viewport: 1'].odbDisplay.displayGroup.intersect(leaf=
            TB_Root_Display2)
        # STRESSES AND DEFLECTIONS
        report_name_and_path2 = 'Diagonal_S11.rpt'
        report_name_and_path3 = 'Diagonal_U.rpt'
        # WRITE REPORTS
        WriteResults(report_name_and_path2,'S.S11',Results,Stepnumber,frame,INTEGRATION_POINT
            ,(('S', INTEGRATION_POINT, ((COMPONENT, 'S11'),)), ))
        #remove elements from displayGroup
        TB_Display = displayGroupOdbToolset.Leaf(leafType=ALL_ELEMENTS)
        session.viewports['Viewport: 1'].odbDisplay.displayGroup.remove(leaf=TB_Display)
        #isolate the support cables
        TB_Root_Display = displayGroupOdbToolset.LeafFromElementSets(elementSets=('
            MEMBRANE_ELEMENTS',))
        session.viewports['Viewport: 1'].odbDisplay.displayGroup.replace(leaf=TB_Root_Display
            )
        TB_Root_Display3 = displayGroupOdbToolset.LeafFromOdbElementMaterials(
            elementMaterials=('DYNEEMA',))
        session.viewports['Viewport: 1'].odbDisplay.displayGroup.intersect(leaf=
            TB_Root_Display3)
        WriteResults(report_name_and_path3,'U',Results,Stepnumber,frame,NODAL,(('U', NODAL,((
            COMPONENT,'U1'),(COMPONENT,'U2'),(COMPONENT,'U3'),)),))
        #EXTRACT DATA
        #Axial Stress
        S11minDC,S11maxDC = ExtractResults(report_name_and_path2)
        min_S11DC,max_S11DC = np.min(S11minDC),np.max(S11maxDC)
        #Displacement
        UminDC,UmaxDC = ExtractResults(report_name_and_path3)
        #----------------------------
        # RESULTS (SLIP RINGS)
        #----------------------------
        #isolate the sliprings
        TB_Root_Display = displayGroupOdbToolset.LeafFromElementSets(elementSets=('SLIPRINGS'
            ,))
        session.viewports['Viewport: 1'].odbDisplay.displayGroup.replace(leaf=TB_Root_Display
            )
        # FORCES AND STRAINS
        report_name_and_path = 'Sliprings_forces.rpt'
        report_name_and_path2 = 'Sliprings_strains.rpt'
        # WRITE REPORTS
        WriteResults(report_name_and_path,'CTF.Magnitude',Results,Stepnumber,frame,
            WHOLE_ELEMENT,(('CTF', WHOLE_ELEMENT, ((INVARIANT,'Magnitude' ),)), ))
        WriteResults(report_name_and_path2,'CU.Magnitude',Results,Stepnumber,frame,
            WHOLE_ELEMENT,(('CU', WHOLE_ELEMENT, ((INVARIANT,'Magnitude' ),)), ))
        # EXTRACT RESULTS
        CTFminSR,CTFmaxSR = ExtractResults(report_name_and_path)
        CUminSR,CUmaxSR = ExtractResults(report_name_and_path2)
        #----------------------------------MODAL FREQUENCIES EXTRACTION
        Frequency_Results = dict([])
```

```
        Frequency_Results_List =[]
        for i in range(1,Freq_Number+1):
            modename = 'Mode '+str(i)
            Frequency_Results_List.append(modename)
            modenamereport = 'Mode '+str(i)+'.rpt'
            Results_Viewport.odbDisplay.setFrame(step = 2,frame = i)
            Stepnumber = int(Results_Viewport.odbDisplay.fieldFrame[0])
            frame = i
            WriteResults(modenamereport,'U',Results,Stepnumber,frame,NODAL,(('U', NODAL,((
                COMPONENT,'U1'),(COMPONENT,'U2'),(COMPONENT,'U3'),)),))
            # EXTRACT THE MODAL FREQUENCY
            extracted_line=''
            f = open(modenamereport)
            for line in f:
                str1=line
                if 'Freq' in str1:
                    extracted_line = str1
                    extracted_list = extracted_line.split()
                    extracted_list = extracted_list[1:]
                    value = float(extracted_list[7])
                    Frequency_Results[modename]=value
                    break
            f.close()
        #----------------------------
        # RESULTS (ASSEMBLY DEFLECTIONS) (PICTURES)
        #----------------------------
        session.viewports['Viewport: 1'].odbDisplay.display.setValues(plotState=(
        CONTOURS_ON_UNDEF, ))
        session.viewports['Viewport: 1'].viewportAnnotationOptions.setValues(legendFont= "-*-
            courier-medium-r-*-*-*-240-*-*-*-*-iso8859-1")
        # OUT OF PLANE DEFLECTION ASSEMBLY PICTURE
        # SET TO LAST FRAME OF GRAVITY LOADSTEP
        lastframe = int(len(Results.steps["Gravity"].frames)-1)
        Results_Viewport.odbDisplay.setFrame(step = 1,frame = lastframe)
        Stepnumber = int(Results_Viewport.odbDisplay.fieldFrame[0])
        frame = int(Results_Viewport.odbDisplay.fieldFrame[1])
        TB_Display = displayGroupOdbToolset.Leaf(leafType=ALL_ELEMENTS)
        session.viewports['Viewport: 1'].odbDisplay.displayGroup.replace(leaf=TB_Display)
        # GET VIEW
        session.viewports['Viewport: 1'].view.setValues(session.views['Iso'])
        session.viewports['Viewport: 1'].view.rotate(xAngle=-90,yAngle=0,zAngle=0)
        session.viewports['Viewport: 1'].view.zoom(zoomFactor=1.25)
        # SET VARIABLE
        session.viewports['Viewport: 1'].odbDisplay.setPrimaryVariable(
        variableLabel='U', outputPosition=NODAL, refinement=(
        COMPONENT, 'U3'), )
        # DEFINE INPUT TEXT FOR ANNOTATIONS ON ALL PICTURES
        import annotationToolset
        namelist = ['Area','k','t_ts','r_ts','Stroke','t_s_max','r_dc','%_EarthGravity']
        count = 1
        startlocation = 150.567
        for i in range(0,8):
            varname = namelist[i]
            varval = str(Configuration[i])
            mdb.Text(name='Text-a-'+str(count), offset=(13.3281, startlocation), text=varname
                )
            mdb.Text(name='Text-b-'+str(count), offset=(60.0000, startlocation), text=varval)
            session.viewports['Viewport: 1'].plotAnnotation(mdb.annotations['Text-a-'+str(
                count)])
            session.viewports['Viewport: 1'].plotAnnotation(mdb.annotations['Text-b-'+str(
                count)])
            count +=1
            startlocation -=5
        mdb.Text(name='Text-17', offset=(98.6281, 244.146), text='ID: '+str(ID))
        # SET PRINT OPTIONS
        session.printOptions.setValues(rendition=COLOR,reduceColors=OFF)
        # SAVE IMAGE
        session.printToFile(fileName=WorkingDirectory+'/U3Deflections'+'/Assembly_Deflection_
            '+str(ID), format=PNG,
        canvasObjects=(session.viewports['Viewport: 1'], ))
        #----------------------------
```

```
1266        # RESULTS (ASSEMBLY 1ST MODAL FREQUENCY) (PICTURES)
1267        #----------------------------
1268        # OUT OF PLANE DEFLECTION ASSEMBLY PICTURE
1269        # SET TO LAST FRAME OF GRAVITY LOADSTEP
1270        Results_Viewport.odbDisplay.setFrame(step = 2,frame = 1)
1271        # GET VIEW
1272        session.viewports['Viewport: 1'].view.setValues(session.views['Iso'])
1273        session.viewports['Viewport: 1'].view.rotate(xAngle=-90,yAngle=0,zAngle=0)
1274        session.viewports['Viewport: 1'].view.zoom(zoomFactor=1.25)
1275        # SET VARIABLE
1276        session.viewports['Viewport: 1'].odbDisplay.setPrimaryVariable(
1277        variableLabel='U', outputPosition=NODAL, refinement=(
1278        COMPONENT, 'U3'), )
1279        session.viewports['Viewport: 1'].odbDisplay.commonOptions.setValues(
                deformationScaling=UNIFORM,uniformScaleFactor=80)
1280        # SET PRINT OPTIONS
1281        session.printOptions.setValues(rendition=COLOR,reduceColors=OFF)
1282        # SAVE IMAGE
1283        session.printToFile(fileName=WorkingDirectory+'/1stModal'+'/Assembly_1st_Modal_'+str(
                ID), format=PNG,
1284        canvasObjects=(session.viewports['Viewport: 1'], ))
1285        #----------------------------
1286        # RESULTS (STRIP S11) (PICTURES)
1287        #----------------------------
1288        # GET VIEW
1289        Results_Viewport.odbDisplay.setFrame(step = 1,frame = lastframe)
1290        TB_Root_Display = displayGroupOdbToolset.LeafFromElementSets(elementSets=('
                MEMBRANE_ELEMENTS',))
1291        session.viewports['Viewport: 1'].odbDisplay.displayGroup.replace(leaf=TB_Root_Display
                )
1292        TB_Root_Display2 = displayGroupOdbToolset.LeafFromOdbElementMaterials(
                elementMaterials=('VECTRAN_SUPPORT',))
1293        session.viewports['Viewport: 1'].odbDisplay.displayGroup.intersect(leaf=
                TB_Root_Display2)
1294        session.viewports['Viewport: 1'].view.setValues(session.views['Iso'])
1295        session.viewports['Viewport: 1'].view.rotate(xAngle=-90,yAngle=0,zAngle=0)
1296        session.viewports['Viewport: 1'].view.zoom(zoomFactor=1.25)
1297        # SET VARIABLE
1298        session.viewports['Viewport: 1'].odbDisplay.setPrimaryVariable(
1299        variableLabel='S', outputPosition=INTEGRATION_POINT, refinement=(
1300        COMPONENT, 'S11'), )
1301        # SET PRINT OPTIONS
1302        session.printOptions.setValues(rendition=COLOR,reduceColors=OFF)
1303        # SAVE IMAGE
1304        session.printToFile(fileName=WorkingDirectory+'/S11'+'/Strip_S11_'+str(ID), format=
                PNG,
1305        canvasObjects=(session.viewports['Viewport: 1'], ))
1306        #DEL ANNOTATION
1307        del mdb.annotations['Text-b-8']
1308        del mdb.annotations['Text-a-8']
1309        del mdb.annotations['Text-b-7']
1310        del mdb.annotations['Text-a-7']
1311        del mdb.annotations['Text-b-6']
1312        del mdb.annotations['Text-a-6']
1313        del mdb.annotations['Text-b-5']
1314        del mdb.annotations['Text-a-5']
1315        del mdb.annotations['Text-b-4']
1316        del mdb.annotations['Text-a-4']
1317        del mdb.annotations['Text-b-3']
1318        del mdb.annotations['Text-a-3']
1319        del mdb.annotations['Text-b-2']
1320        del mdb.annotations['Text-a-2']
1321        del mdb.annotations['Text-b-1']
1322        del mdb.annotations['Text-a-1']
1323        del mdb.annotations['Text-17']
1324        #----------------------------
1325        # RESULTS (ASSEMBLY)
1326        #----------------------------
1327        Assembly_Results = dict([
1328            ('ID',ID),
1329            ('Error',errorsflag),
```

```
            ('Warning',warningsflag),
            ('Area',Configuration[0]),
            ('k',Configuration[1]),
            ('t_ts',Configuration[2]),
            ('r_ts',Configuration[3]),
            ('Stroke',Configuration[4]),
            ('t_s_max',Configuration[5]),
            ('r_dc',Configuration[6]),
            ('%_EarthGravity',Configuration[7]),
            ('Strip_Radii',Strip_Radii[0]),
            ('U_Min_DC_U1', UminDC[0]),
            ('U_Max_DC_U1', UmaxDC[0]),
            ('U_Min_DC_U2', UminDC[1]),
            ('U_Max_DC_U2', UmaxDC[1]),
            ('U_Min_DC_U3', UminDC[2]),
            ('U_Max_DC_U3', UmaxDC[2]),
            ('S_S11_Min_DC', min_S11DC),
            ('S_S11_Max_DC', max_S11DC),
            ('U_Min_SB_U1',UminSB[0]),
            ('U_Max_SB_U1', UmaxSB[0]),
            ('U_Min_SB_U2', UminSB[1]),
            ('U_Max_SB_U2', UmaxSB[1]),
            ('U_Min_SB_U3', UminSB[2]),
            ('U_Max_SB_U3', UmaxSB[2]),
            ('S_S11_Min_SB', min_S11SB),
            ('S_S11_Max_SB', max_S11SB),
            ('U_Min_TS_U1', Umin[0]),
            ('U_Max_TS_U1', Umax[0]),
            ('U_Min_TS_U2', Umin[1]),
            ('U_Max_TS_U2', Umax[1]),
            ('U_Min_TS_U3', Umin[2]),
            ('U_Max_TS_U3', Umax[2]),
            ('S_S11_Min_TS', min_S11),
            ('S_S11_Max_TS', max_S11),
            ('SF_SF1_Min_TS',min_SF_SF1),
            ('SF_SF1_Max_TS',max_SF_SF1),
            ('SM_SM2_Min_TS',min_SM_SM2),
            ('SM_SM2_Max_TS',max_SM_SM2),
            ('Euler_Buckling_TS', EBLoad),
            ('Local_Shell_Buckling_TS', LocalShellBuckling),
            ('CTF_Max_SR',CTFmaxSR[0]),
            ('CTF_Min_SR',CTFminSR[0]),
            ('CU_Max_SR',CUmaxSR[0]),
            ('CU_Min_SR',CUminSR[0])
        ])
        # add in frequency information as well as relative u3 strip displacement information
        Assembly_Results.update(Frequency_Results)
        Assembly_Results.update(U3_SB)
        # order the dictionaries
        Assembly_Results_List = ['ID','Error','Warning','Area','k','t_ts','r_ts','Stroke','
            t_s_max','r_dc','%_EarthGravity','Strip_Radii',
        'U_Min_DC_U1', 'U_Max_DC_U1', 'U_Min_DC_U2', 'U_Max_DC_U2', 'U_Min_DC_U3', '
            U_Max_DC_U3',
        'S_S11_Min_DC', 'S_S11_Max_DC', 'U_Min_SB_U1','U_Max_SB_U1','U_Min_SB_U2','
            U_Max_SB_U2','U_Min_SB_U3','U_Max_SB_U3',
        'S_S11_Min_SB', 'S_S11_Max_SB', 'U_Min_TS_U1', 'U_Max_TS_U1', 'U_Min_TS_U2', '
            U_Max_TS_U2', 'U_Min_TS_U3', 'U_Max_TS_U3',
        'S_S11_Min_TS', 'S_S11_Max_TS', 'SF_SF1_Min_TS','SF_SF1_Max_TS','SM_SM2_Min_TS','
            SM_SM2_Max_TS','Euler_Buckling_TS', 'Local_Shell_Buckling_TS','CTF_Max_SR','
            CTF_Min_SR','CU_Max_SR','CU_Min_SR']
        #add in frequency information to list
        Assembly_Results_List = Assembly_Results_List + Frequency_Results_List + U3_SB_List
        from collections import OrderedDict
        Assembly_Results=OrderedDict(sorted(Assembly_Results.items(), key=lambda pair:
            Assembly_Results_List.index(pair[0])))
        Results.close() # close odb after using it
    mdb.Model(modelType=STANDARD_EXPLICIT, name='Model-1')
    del mdb.models['Quadrent']
    # RETURN TO GLOBAL WORKING DIRECTORY
    os.chdir(WorkingDirectory)
    return Assembly_Results_List,Assembly_Results,warnings,errors
```

```
1394
1395 # ------------------------------------------------
1396 # ----------------- ANALYSIS
1397 # ------------------------------------------------
1398 if Analysis_Flag is True:
1399     # ------------------------------------------------
1400     # ----------------- GENERATE RESULTS
1401     # ------------------------------------------------
1402     # GENERATE CONFIGURATIONS
1403     configurations = GenerateConfigurations(Inputs)
1404     # ITERATE OVER CONFIGURATIONS
1405     for i in range(1,len(configurations)+1):
1406         ID = i
1407         DataName = 'Results.txt'
1408         WarningsName = 'Warnings.txt'
1409         ErrorsName = 'Errors.txt'
1410         config = configurations[i-1]
1411         # PERFORM STATIC ANALYSIS
1412         start = timer()
1413         Assembly_Results_List,Assembly_Results,warnings,errors=StaticAnalysis(ID,config)
1414         end = timer()
1415         # ITERATION TIME
1416         IterationTime = str(end-start)
1417         print('Iteration Time: ' + str(end-start))
1418         PercentComplete = float((ID/1.000/len(configurations))*100)
1419         print('Percent Complete: ' + str(PercentComplete))
1420         # WRITE THE RESULTS
1421         values = []
1422         for key in Assembly_Results:
1423             values.append(Assembly_Results[key])
1424         # CHECK IF DATA SHEETS ALREADY EXIST
1425         if os.path.exists(DataName) == False:
1426             # RESULTS
1427             with open(DataName, 'w') as output:
1428                 for val in Assembly_Results_List:
1429                     output.write(str(val)+',')
1430                 output.write('IterationTime'+',')
1431                 output.write('\n')
1432                 for row in values:
1433                     output.write(str(row)+',')
1434                 output.write(IterationTime+',')
1435                 output.write('\n')
1436             # WARNINGS
1437             with open(WarningsName, 'w') as output:
1438                 output.write(str(ID)+',')
1439                 for val in warnings:
1440                     output.write(str(val)+',')
1441                 output.write('\n')
1442             # ERRORS
1443             with open(ErrorsName, 'w') as output:
1444                 output.write(str(ID)+',')
1445                 for val in errors:
1446                     output.write(str(val)+',')
1447                 output.write('\n')
1448         else:
1449             with open(DataName, 'a') as output:
1450                 for row in values:
1451                     output.write(str(row)+',')
1452                 output.write(IterationTime+',')
1453                 output.write('\n')
1454             # WARNINGS
1455             with open(WarningsName, 'a') as output:
1456                 output.write(str(ID)+',')
1457                 for val in warnings:
1458                     output.write(str(val)+',')
1459                 output.write('\n')
1460             # ERRORS
1461             with open(ErrorsName, 'a') as output:
1462                 output.write(str(ID)+',')
1463                 for val in errors:
1464                     output.write(str(val)+',')
```

```
                output.write('\n')
        #DELETE THE FOLDER WITH ALL THE FILES
        if len(errors)>0:
            next
        else:
            shutil.rmtree('Analysis'+str(ID))
else:
    configurations = GenerateConfigurations(Inputs)
    config = configurations[0]
    ID = 1
    DataName = 'Results.txt'
    # PERFORM STATIC ANALYSIS
    start = timer()
    Assembly_Results_List,Assembly_Results,warnings,errors=StaticAnalysis(ID,config)
    end = timer()
   # ITERATION TIME
    IterationTime = str(end-start)
    # WRITE THE RESULTS
    values = []
    for key in Assembly_Results:
        values.append(Assembly_Results[key])
    # CHECK IF DATA SHEETS ALREADY EXIST
    if os.path.exists(DataName) == False:
        with open(DataName, 'w') as output:
            for val in Assembly_Results_List:
                output.write(str(val)+',')
            output.write('IterationTime'+',')
            output.write('\n')
            for row in values:
                output.write(str(row)+',')
            output.write(IterationTime+',')
            output.write('\n')
    else:
        with open(DataName, 'a') as output:
            for row in values:
                output.write(str(row)+',')
            output.write(IterationTime+',')
            output.write('\n')
    print('Iteration Time: ' + str(end-start))
    #DELETE THE FOLDER WITH ALL THE FILES
    shutil.rmtree('Analysis1')
```