Dynamic Modeling of a Large Solar Array in a Deep Space Orbit

a project presented to
The Faculty of the Department of Aerospace Engineering San
José State University

in partial fulfillment of the requirements for the degree
Master of Science in Aerospace Engineering

by

Richard J. Morrison

May 2019

approved by

Dr. Sean Swei
Faculty Advisor

San José State
U N I V E R S I T Y

Symbol	Definition	Units
I_{xx}	Measurement of the resistance to accelerate a mass about an axis	lb in^2 (kg m^2)
I_{yy}	Measurement of the resistance to accelerate a mass about an axis	lb in^2 (kg m^2)
I_{zz}	Measurement of the resistance to accelerate a mass about an axis	lb in^2 (kg m^2)
M	Moment	Nm
P	Solar pressure	N
B	Magnetic pressure	N
G	Gravitational force	N
T	Thermal pressure	N
Q	Uniform load on the beam	lb/in (kg/m^2)
L	Length of the beam	in (m)
E	Modulus of elasticity	psi (pa)
I	Area moment of inertia	in^4 (m^4)
∋	Elastic deflection	in (m)
∅	Angle of deflection	rad (m)
B	End of the uniformly loaded cantilever beam	
B'	Point on the wall of the uniformly loaded cantilever beam	
ρ	Density of the beam	kg/m^3
A	Area of the beam cross section	

k_v	Stiffness of the spring	kg/m
m_v	Mass of the sprung mass	kg
w	Transverse deflection of the beam	m
u	Vertical displacement of the mass	m
q	Modal coordinate	
t	Time	sec
γ, β, α	Newmark integration parameters	
p_v	Iteration sprung mass parameter	
P_B	Iteration beam parameter	
η	Relaxation coefficient	
a	Iteration coefficient	
ψ	Mode shape	
T	Time offset	sec
λ	Substitution variable	
n	Number of modes	
i, j	Integer between 1 and n	
l	Arbitrary length	
x	Arbitrary position	m
X	NASTRAN axis	
X'	Rotated axes	
Y	NASTRAN axis	
Y'	Rotated axes	
Z	NASTRAN axis	
Z'	Rotated axes	
b	Iterative beam subscript	
δ	Dirac delta function	

M	Mass matrix	
K	Stiffness matrix	
s_v	Newmark sprung mass sum	
S_B	Newmark beam sum	
g	gravity	m/s^2
r	Moment arm	m
ω	Angular rates	Rad/s
F	Arbitrary force	N
ϵ	Convergence criteria	

Dynamic Modeling of a Large Solar Array in a Deep Space Orbit

Chapter 1 – Introduction

1.1 Motivation

The problem at hand is the lack of a Large Space Structure (LSS) that can adequately power something as massive as a space habitat, or otherwise large spacecraft in interplanetary orbits. No such structure has been designed that is strong enough to survive the maneuvers required to keep itself in interplanetary orbits. Without research into how to hold an LSS together, and without an in-depth understanding of their dynamics, these structures cannot exist. During the present time, this issue hasn't come to practical concern. LSSs have only been theorized and haven't been put into practice. The closest thing that comes to an LSS would be the International Space Station (ISS), and it isn't as large as what would be considered in this paper's analysis. The size of the LSS in the paper will be on the order of a kilometer long. LSS will be necessary for the spacecraft of the future to support large payloads in interplanetary orbits. So how is an LSS modeled? It's incredibly difficult to model a spacecraft on the order of a kilometer in size. It may require a super computer, depending on exactly how complicated the structure is. This paper will focus on the solar array surface, and to be as efficient as possible with whatever Finite Element Analysis (FEA) software is used.

1.2 Literature Review

It is important to note that the scope of modeling a Large Solar Array (LSA) is not in the same scope as most fundamental research in control theory or flexbody dynamics. Most other analysis on the stability of spacecraft are focused on the design of a satellite in an Earth orbit. It is very common to find papers on rigid bodies with flexible attachments. An example of a rigid body

with a flexible attachment would be the main body of a communications spacecraft with attached reflectors and solar arrays. Creating a model for an LSA generally assumes one massive flexible structure. The focus of modeling an LSA as a massive flexible structure is to find a way to simplify the flexible structure into smaller, computable parts. The idea of a massive flexible structure is kept in mind during this review.

To set a good baseline of appropriate assumptions of external influences and forces on LSS and how to control LSS, Gasbarri, Monti, Angelis, and Sabatini (2014), is considered. Robustness of the design once the control law has been developed is going to be critical to future LSAs. If the system is not robust (neglects disturbances), such a large structure would not be useful. There is a tremendous amount of mass in an LSA, and that gives the structure more dynamic uncertainty. Elastic characteristics are also discussed in Gasbarri, Monti, Angelis, and Sabatini (2014) and these elastic effects would greatly affect the dynamics of an LSA. When the dimensions are on the order of a kilometer in size, a spacecraft will stretch and compress a considerable distance. Thus, the elasticity will certainly affect the dynamics on any LSS. How the inertia and elasticity relate to the robustness of an LSA are going to be critical design points in developing the model of this paper. Gasbarri, Monti, Angelis, and Sabatini (2014) also defines the kinematic equations of motion, applies Euler and quaternion rotations, defines a very large spacecraft transfer orbit and attitude delta, considers gravity influences, elastic influences, applies a PD controller, and uses NASTRAN for the Finite Element Analysis (FEA). This is a good front to back analysis of the flexible modes of a solar array including an impulse torque, however it only extends to a PD controller. A PD controller is very likely not powerful enough of a controller to consider in the design of an LSA. Although the focus of this paper is not the analysis of the controller, however, a modern controller should be considered so that the constraints of the model developed in the paper are useful. The

application of NASTRAN provides a good example of FEA and the derivation of equations of motion for a three-dimensional spacecraft body will directly aide the model of the LSA.

In contrast, Wang (2017) describes the basics of the equations of motion of a rigid body spacecraft with flexible appendages. The LSA is one massive flexible body, so it is hard to imagine an attached rigid body that is large enough to dominate the dynamics of the model. With that being said; coupling dynamics will still exist on the LSA because the solar array is so large that one part of the solar array will couple with other parts, or attachments, of the solar array in unique ways. Part of the motive behind doing an analysis on an LSA is to discover an efficient way to scale a model of an LSS, knowing that it could couple with itself. Although the LSA is not a rigid body, it will still be useful to have the first order time derivate matrices required to transform one part of a spacecraft to another. FEA was used to model the flex dynamics of the solar array on the spacecraft, but all that was mentioned for analysis was MATLAB. Wang (2017) is a good paper, which could be useful as an example to set up equations of motion.

And in Wallrapp (2002), there is a detailed description of the kinematics of a generic flexible body. Although it should be noted that the simulation case given was the deployment of a solar array. In the model of an LSA, deployment will not be considered. It is assumed that the solar array was assembled in space, and not folded and deployed. Kinematic analysis is good to be mindful of with respect to orbital mechanics. However, the use of kinematics in this paper with respect to LSS would only be useful if an orbital analysis was necessary to determine the external forces exerted on the LSA. It is also another reminder that the assumption of a multibody system will likely be necessary to set up the model for the analysis on the LSA. It will also be necessary to use FEA software to find a solution to the LSA system. Making a scalable model that can be applied to multiple FEA softwares will prove the model has flexibility to be used on any computer.

With the application of linear algebra, Wallrapp (2002) had a lot of very good examples describing how to mathematically set up solar array flexibility models. Wallrapp (2002) concludes with saying that its investigation could use more research to determine what the best set up of the problem is i.e. how to model something that only has a certain number of flexible bodies.

Once equations of motions are established, it is necessary to know whether a system is controllable. Controllability and observability are described in Williams (2002). The impression of Williams (2002) is a general overview of controllability and observability on flexible structures in certain vibrational modes. A big focus were the modes that occur at natural frequencies. Natural frequencies will be important to identify for the LSA for any simulations that are run, or any analysis that is done on the stability of the solar array. Knowing ahead of time if different modes natural frequencies of the LSA are controllable and observable will indicate if the model is practical. If the chosen design of the LSA cannot be controlled and predicted through its modes, then the model is not useful. Williams (2002) does makes some conclusions about the magnitude of the damping ratio relating to the controllability and observability.

Again, controllability and observability will be critical parts of developing the model. And to describe all the real systems of a spacecraft, Bainum (1984), is a good start. It covers: open- and closed-loop beam dynamics, solar radiation torques, and thermal effects. The discussion of Bainum (1984) pertains to plants, and noise and such involved with large flexible structures. Although Bainum (1984) discusses the solar arrays and antenna of the spacecraft as lumped masses, it still provides analysis for a closed-loop system of the spacecraft with time delayed actuators, thermal disturbances, and solar radiation pressures. A lumped mass assumption will not be used on the LSA, however solar radiation and thermal disturbances will be considered. It is seen in the analysis in Bainum (1984), that there is a breakdown of exactly what all the vectors and forces are being

applied to the structure (bending, solar, and thermal effects). The dynamic models are boiler plate, along with equations describing solar radiation forces. Solar radiation torque will likely be a large contributing factor to the model of the LSA given the shear surface area exposed to the sun. An equivalent observation about the solar radiation can be applied to the thermal forces from diurnal effects of the orbit of the LSA.

More classical control theory is described in Thau (1985). Four different methods were used to study adaptive learning control. Those methods were: a modal model, a least-squares identification scheme, a separable nonlinear least-squares approach, and a dual-adaptive control strategy. Although a lot of the content in Thau (1985) is not modern, it does breakdown how to model large structures and identify modes, find the function of all the models, and estimate pole-zero parameters. Classical methods for discussing and setting up the model of a control system are discussed such as; rank, linear dependence, and eigenvectors. A simulation software was used to generate a solution to their model, however the software was not modern.

Maghami (1995) used simulations to obtain its results. However, it also developed a way to design a large spacecraft (in this case a geostationary spacecraft) while doing the structures analysis, and the controls analysis at the same time. This method would streamline the design process of an LSA because the size of the structure and its stability are highly coupled. A massive solar array must maintain pointing to be useful, and every bit of mass on LSSs matters. There were two control system designs presented in the paper, the first was an analytical solution, and the second of the two design approaches used CSI-DESIGN. CSI-DESIGN was what was used as the primary solution in the paper. The first design is an Earth Pointing System, and the second is a test bed at Langley. In both design processes, the goal was to achieve a certain amount of mass/stiffness

in the spacecraft that would be adequately controllable, and still reject disturbances. The spacecraft structures between the two problems were similar, but not the same.

In Mu (2018) it was found that with an increase in size of the spacecraft, and if the structural fundamental frequency and attitude frequency are close, that the orbit would decrease in altitude. System instability was also found to be very closely related to structural vibrations. The discussion of how structural excitation relates to stability is critical in developing an LSA model. An uninformed reader that knew little or nothing about the orbital mechanics and structural dynamics of an LSS would find Mu (2018) very informative. The thesis was classic, and it broke down the equations of motion for an LSS in a GEO orbit in a way that could be understood. Mu (2018) was modeling flexible body dynamics to some degree, but it was in the form of beam bending. The analysis done was purely two dimensional, however still applicable to LSS. The paper discusses the adverse effect of delta-v on an LSS as it orbits around a body. At the beginning of constructing the model of an LSA, it will be necessary to build equations of motion from solar pressure forces, gravity, and some thermal effects.

For a modern look at how to design controllers specifically for LSS, we have Hyland (1993). The scope of this paper was broad and covered several different kinds of control. Active control of LSS is stated to require the use of a multiple-input-multiple-output (MIMO) systems. Utilization of a MIMO state-space equation will be required to model an LSA. A single-input-single-output (SISO) system will not suffice given the large amount of disturbances and inputs to an LSA. LGQ controllers and their limitations were the most emphasized analytic control method in Hyland (1993). The rest of this literature seemed to discuss LSS as a whole; including things like in–orbit-test, and hardware application. However weak in analysis, this paper has many helpful references and suggestions.

Stability is a big concern in the integrity of a solar array, because the modes at which it is stable aren't necessarily the same modes that won't cause the solar array to break. Bloch (1992) discusses a method of expressing the stability of a system with velocity and momentum. The torques on the spacecraft are applied by gas jets and a multi-rotor system using a quadratic feedback control law. A Hamiltonian is then applied to a Lie-Poisson bracket. The overall topic was focused on the analysis of internal rotor control on a rigid body. Bloch (1992) utilized things like Hamilton Structures, Lie-Poisson, and Energy-Casimir. Bloch (1992) mathematically describes systems in terms of velocity and momentum for the dynamic behavior of a rigid body controlled internally by wheels of different kinds. It discusses three- and single-wheel situations, as well as phase shifts. And to see more about stabilization Bloch (1990) is discussed. Bloch (1990) has a method of using angular momentum applied to a rigid body to provide stability. Bloch (1990), however, is unique because it takes the case of only one applied torque. Normally the stability of a spacecraft needs at least two torques to stabilize a spacecraft about its principal axis. From the point of stability, using one torque may be useful to an LSA. Stability will depend on the shape, size, and trajectory of the LSA. Knowing the analysis required to stabilize the solar array, use of one torque could simplify the solar array structure. The method of Energy-Casimir was applied to single axis the torques.

Kaplia (1999) contains a discussion of the Ricatti equation and how it is applied to stabilization. A direct application of the Ricatti equation can be seen in Gasbarri, Monti, and Sabatini (2014). However, in Kaplia (1999), it was realized that open-loop dynamics applied at the wrong time can cause undesirable dynamics. By closing the loop, the input is shaped so the flexbody vibrations change and better the performance of the output. There is a trade in adding impulses to the system between speed and robustness. Kaplia (1999) seems to contain much more

modern control theory and is closer to what is taught in academics today compared to other papers applying classical control theory. The application of time-delay control, however, may not be applicable in an LSA. Kaplia (1999) is a healthy discussion of modern control theory techniques like the use of LQR controllers, poles and zeros, and Lyapunov stability. Stability being applied directly to rigid and flexible bodies is of interest to an LSA model. Krishnaprasad (1985) is a good place to start for determining the stability of a rigid or flexible body. A model is developed using Poisson manifolds on a rigid body with attached flexible beams (a spacecraft main body with attachments like solar arrays). Energy-Casimir was also applied to determine if a model had stability. Having reviewed rigid body models and Hamiltonian methods, Krishnaprasad (1985) also assumes that the reader has had some exposure to Possion manifolds. There is a brief review of Lie-Poisson brackets in Krishnaprasad (1985). Lie-Poisson represent flexible structures in the coordinates of the body. Rigid bodies with flexible attachments have smooth, nonlinear solutions. The Energy-Casimir method provides an algorithm for determining sufficient conditions for the nonlinear stability of equilibria for systems whose underlying Hamiltonian structure has a rich collection of Casimir functions. All the proceeding methods are applied at the end of Krishnaprasad (1985) to find the stability of the equilibrium solution.

Bathe (2014) is a textbook about FEA methods. Bathe (2014) will be useful when the techniques are applied an LSA. Slicing the LSA will require FEA to determine if the methods are feasible to use on computers with different computational levels.

There are several methods to approach constructing a model of an LSA. Most published research pertains to specific methods with respect to either stability, constructing a model, structure design, mode analysis, or a combination of all of them. Not all angles of analysis will be required

to construct a model for an LSA. However, mindfulness of all the tools available through published research to create a model is helpful.

1.3 Project Proposal

The objective of this project is to model the flex dynamics of a very large solar array in a deep space orbit. The solar array in reality would have the dimensions on the order of kilometers long, although the model in this paper will be 1 meter. The solar technology itself is not of interest in this project. The focus of this paper will be on how to most efficiently model the dynamics of a solar array that is very large. The efficiency of the model is driven by the desire to be able to scale this analysis to any workstation. The need for a very large solar array has not become necessary yet, however because one day a very large solar array will be required for things like space habitats, a dynamic model will be developed and analyzed here in this paper.

1.4 Methodology

The methodology to completing this scalable mathematical model will be approached with classic equations of motion of the dynamic system. Once a proper dynamic system has been established, a computer with finite element analysis software will be used to compare different vibrational mode shapes of the large solar array. The modes will be components that can be stitched together to represent the full dynamics of the very large solar array. Having the first mode shapes of the LSA allows for further analysis to iterate and predict all mode shapes of the LSA.

The model of the very large solar array will be used to discover the most efficient way of slicing the structure into scalable components to find a solution to the dynamics.

Chapter 2 – Purpose

2.1 The Model

The goal of the FEA software (NASTRAN) model's initial setup is to create a set of slices that can appropriately model an LSA as a uniformly loaded cantilever beam. Computation time and accuracy shall be the parameters used to qualify success. The least computation time for the most amount of accuracy is desired. Requirements of each slice much fall within a list of assumed condition that keep the simulations within the realm of usefulness.

Nikos Mourtos
2019-05-29 00:28:59

PURPOSE OF WHAT?

Having an accurate model of the cantilever beam enables the ability to predict the flex dynamics of the LSA. However, developing an accurate model for an LSA would be extraordinarily difficult. With that in mind, a model will not be developed in this paper. NASTRAN will allow for a computational analysis on how an LSA would flex.

2.2 Flexible beam

The use of a large flexible beam instead of solar array is for a few reasons. The first reason is to perform a proof of concept on something simple. The slicing method must be feasible on a simple object like a bending fixed beam to apply it to an LSA. The flexible beam solution will yield details about how something very large bends over a large distance given a bunch of scaled slices.

There is no total defined area that will be used for the first FEA models. The simulation needs to be accurate enough answer when scaled to the total area of whatever dimensions that a fix beam has. The flexible modes that result from the computations can be compared to an analytical solution from theory. The total area of the flexible body should be built from a certain number of elements (number of slices). The number of elements could approach infinity until the

displacement of the beam becomes impractical to imagine. An LSA is very large, and thus the scaled computation could be applied to a spacecraft that is multiple kilometers in magnitude. Each slice does not need to be rectangular, but it is likely that a massive rectangular will be the shape of an LSA of the future.

Accuracy of the simulation is going to depend on the software chosen and how sparse nodes in the mesh are. A software that has sufficient fidelity and speed should be chosen to complete the simulation. The shape of a real LSA is likely going to be vastly different than in these simulations, however properly constructing a mesh of elements allows these simulations to be applicable to a high variety of possible LSA shapes. Gravity gradients, temperature, and solar torque are only considered instantaneously in time. No diurnal analysis of forces will be included in this simulation. A picture of the LSA is given in Figure 1.
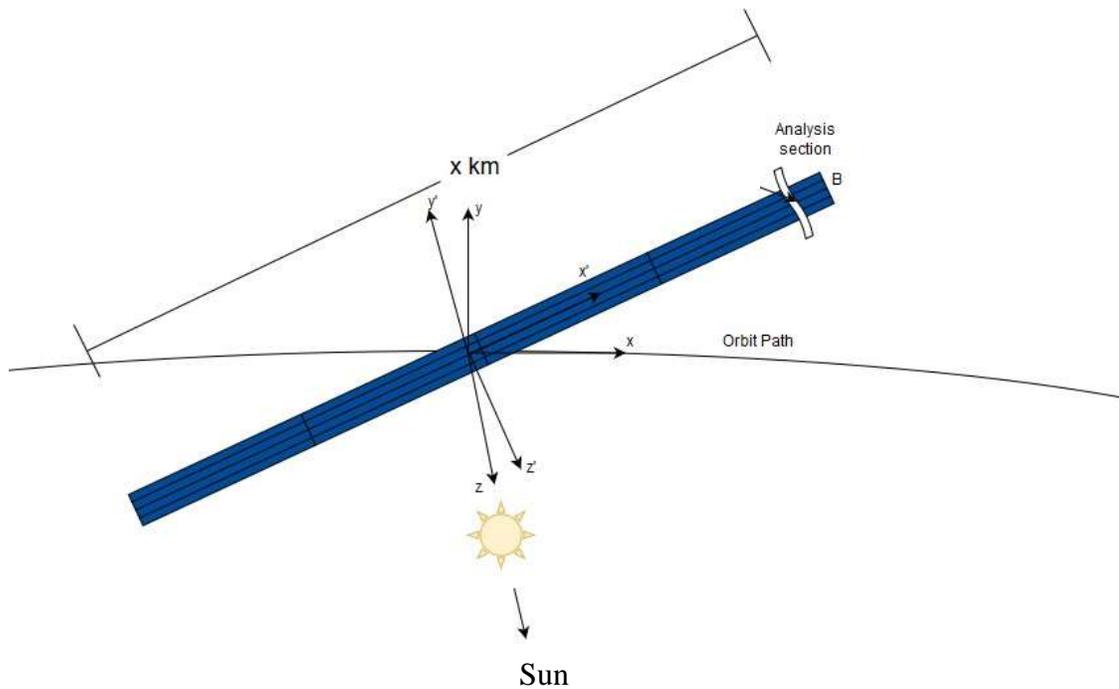
Figure 1: An LSA On Orbit

The first model of the beam is going to be something very simple. It will be a small, symmetric, slice of what should be scaled to become the whole solar array.

2.3 Assumptions

A list of assumptions has been made to simplify the LSA into a problem that's more manageable for the proof of concept in this paper.

2.3.1 The solar array is in deep space.

Having a deep space spacecraft comes with a slew of different conditions compared to a spacecraft that is near earth. The most obvious implication of having a deep space LSA is the distance from the sun. Because the LSA is at a much further distance from the sun than the earth

is, the solar pressure, and available solar flux are much lower. Solar pressure and solar flux are affected by the inverse square law, meaning that the solar pressure and solar flux are decreasing at the inverse squared distance from the suns center. Depending on where the Solar array is in its orbit, the LSA could have a solar pressure and a solar flux like earths, or near zero.

The second most obvious implication of having a deep space solar array is the varying degrees of gravitational acceleration. The gravity gradient that is being imposed on the LSA is going to completely depend on the nearest, and largest objects to the LSA. For the purposes of the paper, the LSA is going to be assumed to be near mars and circularized around the sun.

Given that the LSA is in orbit around the sun and not another, closer, planetary body, eclipse conditions can be neglected.

2.3.2 The solar array is a homogenous material

Solar arrays are almost always made from several layers of materials, attached to various electronics, and structurally reinforced so that they are rigid (or can bend a certain amount). Given the complication and scope of doing a structural analysis on several layers of a solar array, the assumption of a homogenous material is sensible. With a homogenous material, one elastic modulus can be used, and it will be much easier to benchmark the accuracy of solution to the LSA bending dynamics without corrupting the proof of concept. The proof of concept only needs to show that it is possible to model the mode shapes and normal modes of the LSA, and to find a displacement solution. To find the solution of a structurally complicated LSAs bending dynamics is outside the scope of the paper.

2.3.3 The deflections of the solar array are small

If deflections are too large, bending gets to the point where the structure may begin to deform. A plastic structure is not useful to model because if an LSA were to be built, it would need to be flexible enough to retain its strength and shape. This papers proof of concept is focused on the bending dynamics and not a structural failure of the LSA. The small deflections assumption also simplifies the analysis.

2.3.4 The solar array is a beam shape

There are many kinds of shapes that a solar array can be. The shape of a future LSA is not only unpredictable, but arbitrary. No matter the shape of the future LSA, there is still quite a bit of practicality in a beam shaped solar array. Solar arrays are built as sheets and a long, thin sheet it effectively a beam. The equations of motion are very well defined for a fixed beam, and a beam is a plausible macro shape for an LSA. It then makes sense to use a fixed beam in the simulation of an LSAs bending dynamics.

2.3.5 The solar array (beam) is originally straight

An LSA in an actual orbit is likely never going to be perfectly straight. However, to validate the analysis in this paper, it is necessary to start from a predictable equilibrium. The bending from equilibrium (a straight beam) is valid given that the range of motion of the beam is reasonable.

2.3.6 The summed forces of gravity, solar pressure, thermal expansion/contraction, and magnetic field act uniformly on the beam

Different spacecrafts may experience many different forces on orbit. Exercising different mechanisms to apply a load will not provide a more accurate simulation to the reality of a flexible LSA. It is however, noteworthy to understand the variety of forces that are acting on the solar array. A force distribution is being used to simplify the analysis with the equations of motion. The

forces are acting in a downward direction (like gravity on earth), however in this paper they represent the summed deep space forces.

2.4 Equations of Motion

To better understand the dynamics involved with a flex body, some equations of motion are given. The equation at the core of all motion is Newton's Second Law.

$$\sum \vec{F} = \sum m \, \vec{x} \qquad\qquad 2.1$$

Newton's Second Law tells us that the LSA must react in response to a sum of all forces acting on it. In this case; gravity, solar pressure, thermal expansion/contraction, and magnetic field are all the external forces that are acting on the solar array. Given is a symbolic representation of all the forces acting on the LSA:

$$\sum \vec{F} = \vec{G} + \vec{P} + \vec{T} + \vec{B} \qquad\qquad 2.2$$

The sum of forces F shall be represented for a force distribution on a cantilever beam. The uniformly loaded cantilever beam in this case is representing the LSA under all the forces of F. See Figure 2 for a representation of F on the cantilever beam. F is described per unit length of the beam as f.
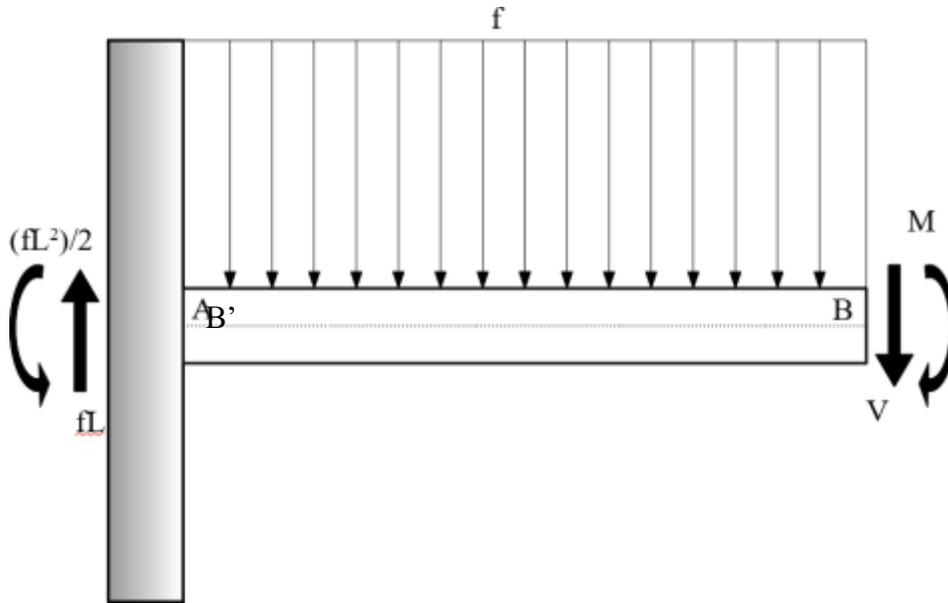
Figure 2: A Uniformly Loaded Cantilever Beam

Nikos Mourtos
2019-05-29 00:29:51

Formatting guidelines.

2.3

Following is Newton's Second Law in the form of moments.

$$\sum \vec{\vec{M}} = \sum F \vec{r}$$

2.3

Moments can be used to describe beam bending, so Newton's Second Law in this form will be more useful than Equation 2.1. A sum of moments can be used to derive equations for beam bending. To find the point of maximum deflection, the following equations are expressed in common literature as Equations 2.4 and 2.5.

$$\ni_B = \frac{QL^4}{8EI}$$

2.4

$$\emptyset_B = \frac{QL^3}{6EI}$$

2.5

The subscript B is indicating the end of the beam. If the conditions for breakage of the cantilever beam are known, maximum deflection with a certain amount of safety factor can be used as a certain amount of allowed bending in the design.

Equations 2.6 and 2.7 describe the deflection of a uniform load on a cantilever beam at a point x along the line connecting points B' and B. These equations will be useful by providing a function to plot the deflection. Plotting the deflection of the cantilever beam will show the critical point of stress, transitioning into plastic regions of the solar array.

$$\exists_x = \frac{Qx^2}{24EI}(6L^2 - 4Lx + x^2) \qquad\qquad 2.6$$

$$\emptyset_x = \frac{Qx}{6EI}(3L^2 - 3Lx + x^2) \qquad\qquad 2.7$$

Equations 2.6 and 2.7 provide the deflection at the end of a cantilever beam with a uniform load. These equations will be useful by providing a sense of how far a piece of a solar array will deflect along its length. The station along the beam is indicated by the subscript x.
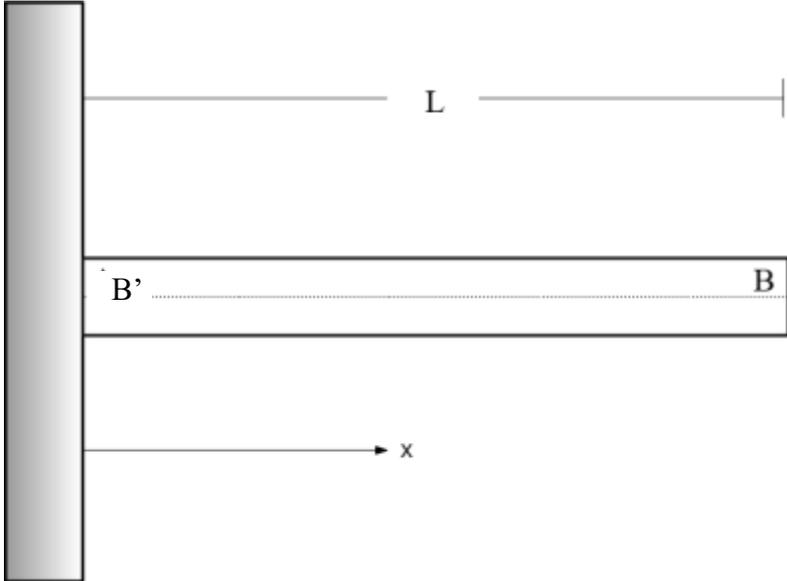
Figure 3: Dimensions of the Cantilever Beam
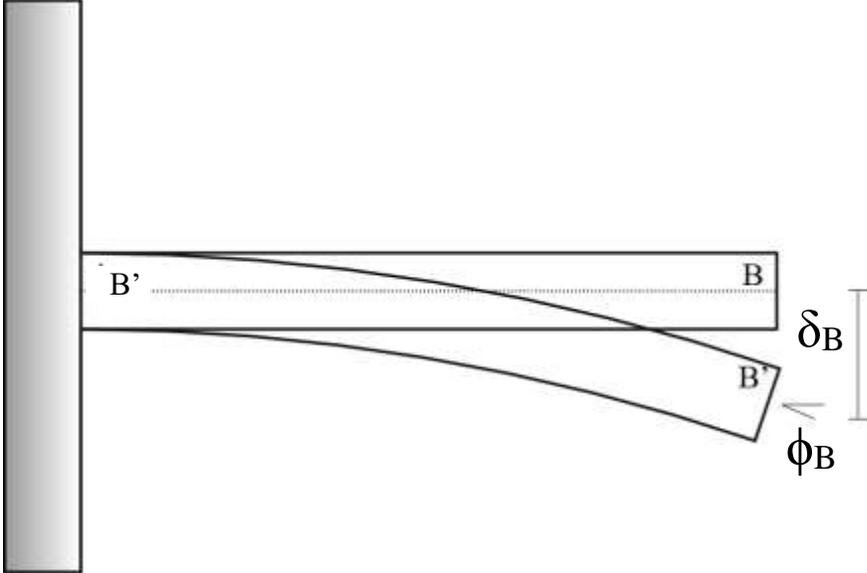
Figure 4: Displacement of the Cantilever Beam

Euler's equations of rotational motion will be useful while describing the rigid body dynamics involved in this LSA problem. These equations (2.8 – 2.10) can be applied to the LSA problem by describing how the LSA is rotating with respect to its external forces.

$$M_{netx} = I_{xx}\dot{\omega}_x + (I_{zz} - I_{yy})\omega_y\omega_z$$

$$M_{nety} = I_{yy}\dot{\omega}_y + (I_{xx} - I_{zz})\omega_z\omega_x \qquad\qquad 2.8\text{ - }10$$

$$M_{netz} = I_{zz}\dot{\omega}_z + (I_{yy} - I_{xx})\omega_x\omega_y$$

2.5 FEA in AutoCAD

How the slices are comprised in the FEA is going to depend on the software. For the simulation of the LSA, AutoCAD NASTRAN In-CAD has been selected. NASTRAN In-CAD is easy to use and has proven reliability from its use in industry.

The solution from the computation in NASTRAN will show how the normal modes (in this case caused by gravity, or solar radiation) will bend and flex the solar array. A theoretical solution should be compared to the computational solution to confirm the validity of the solution before it is iterated.

A flexible beam FEA analysis will be benchmarked against a traditional analysis.

Chapter 3 – FEA

3.1 Parts

A certain number of different shapes have been chosen as example

shapes have been chosen first to compare results. First a rectangle has been chosen because it is

the easiest to imagine as a scalable shape. Most modern solar arrays are rectangular, so it is

reasonable to extend that an LSA of the future could also be rectangular. Other shapes to be

considered are the hexagon and the triangle. A hexagonal shape could be scalable to solar arrays

that are circular, or at the very least, not perfectly rectangular. A triangle on the other hand can be

applied to a very high variety of different shapes. For instance, two triangles can be rectangular,

or many may be fitted together to assume any shape. Meshes in FEA are commonly comprised of

triangles, and thus is designed to handle many different shapes and sizes of models. If triangles

can be used to build a high variety of shapes in FEA, it is likely that triangles can also be used to

build any slice of an LSA.

All the parts in this FEA analysis were drawn in AutoCAD Inventor. Each part is a three-

dimensional solid material. Once all the parts were drawn in Inventor, they were moved into the

AutoCAD NASTRAN In-CAD environment. In NASTRAN the shapes had similar loads,

constraints, and meshes applied to them. Loads are the forces that each part experiences. All the

loads are distributed forces on one face of each part. For NASTRAN to come up with a solution

to the loaded part, the part must be constrained. Constraints are sections of each part that cannot

move or bend. The meshes are grids made up of triangles that are the points NASTRAN will

calculate solutions for. Generally, the more points, the more accurate the solution.

Figure 5 represents a slice of the solar array in the form of a rectangular prism. Simulating

the summed forces acting on the slice is a distributed force on the top surface in the negative Y

direction. The slice is also constrained along its face parallel to the XY plane i.e. the XY face is rigidly attached to a wall. There is also a fine mesh of triangles that has been distributed over the slices surface.
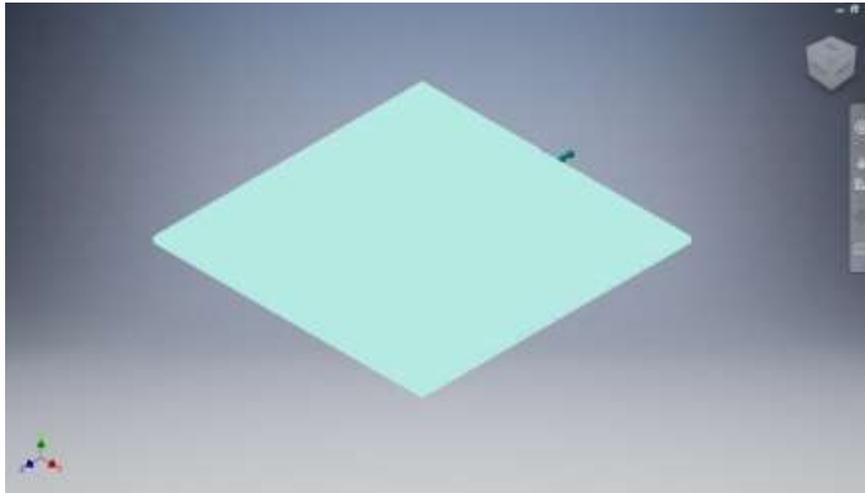

Figure 5: Rectangular Slice

Figure 6 shows a triangular slice and it is close in shape and size to the rectangular slice. The triangular slice is constrained on the face that is parallel to the YZ plane. The distributed load is applied in the negative Z direction on the XY face of the slice. The triangle slice has an equivalent fine mesh to the rectangular slice.
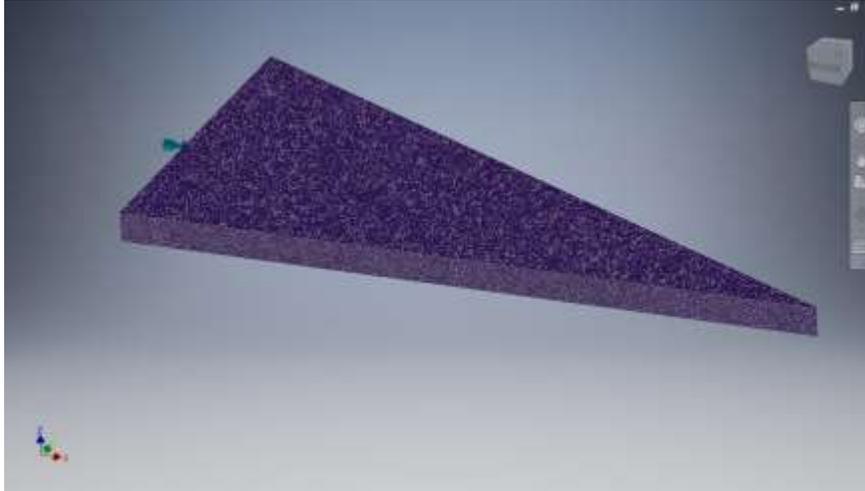
Figure 6: Triangular Slice

Lastly is the hexagonal slice. The hexagon is similar in size to the rectangle and the triangular slices. The hexagon is constrained in the XZ plane. The distributed load is applied to the hexagon on the XY face of the slice. The mesh is also similar in size and shape to the rectangular slice and the triangular slice.
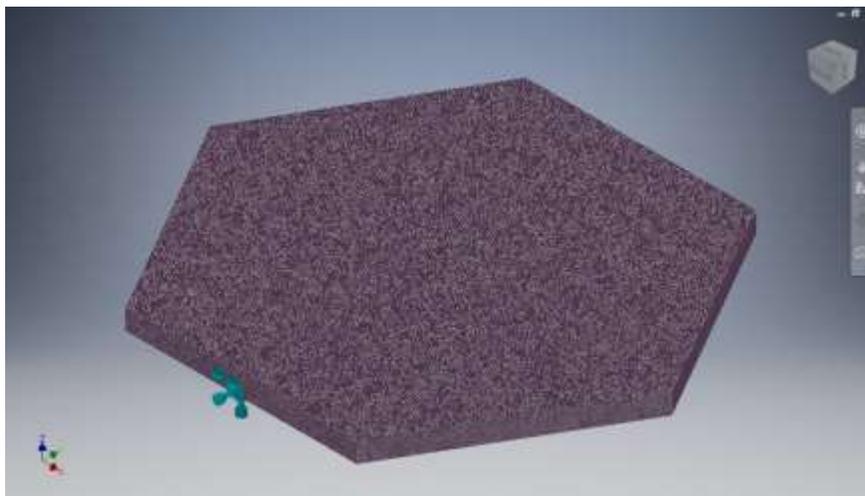
Figure 7: Hexagonal Slice

3.2 Simulations

NASTRAN was used to find the solutions of a uniformly loaded cantilever beam. Each slice is equivalent to a beam. All solutions are three dimensional. Solutions are represented in the following figures:
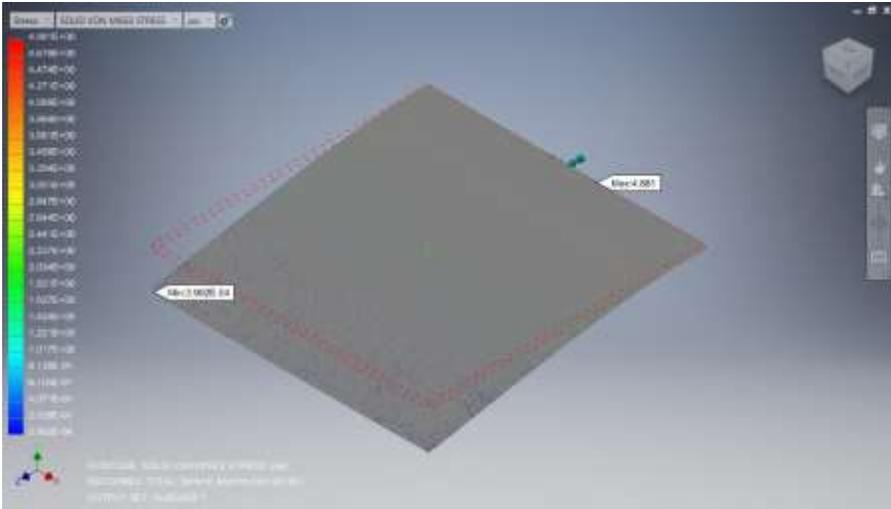
Figure 8: Bending of the Rectangular Slice
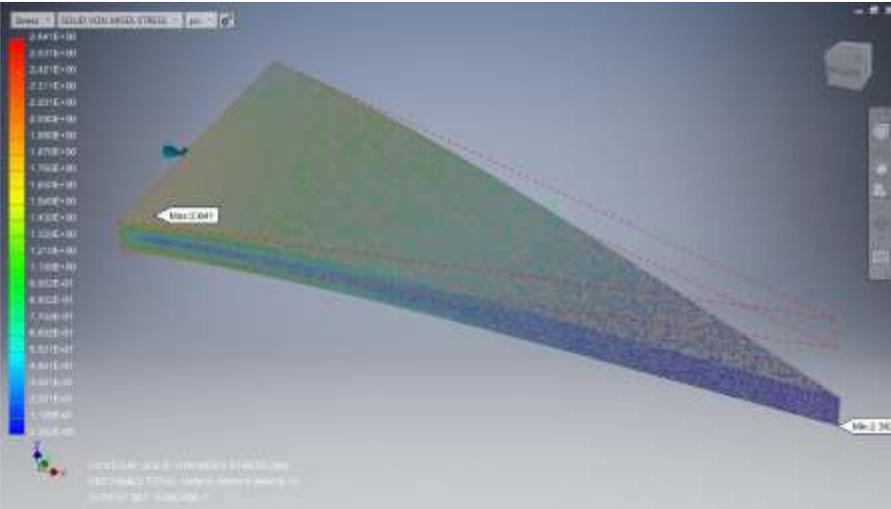
Figure 9: Bending of the Triangular Slice
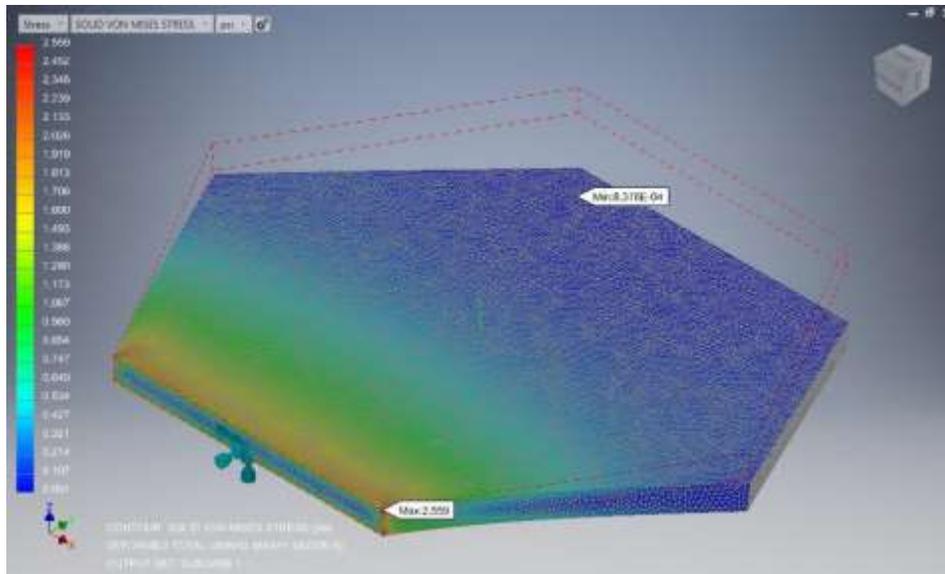
Figure 10: Bending of the Hexagonal Slice

Displacement and stress for all results were believable. Displacement was on the order of 1×10^{-4} inches and stress was around 4 psi for all solutions. The material for all slices was aluminum and each of the slices were 12 inches long.

The solutions of all three slices provide proof that NASTRAN is viable software to solve the problem of what slicing method is best. The processing power, and accuracy of the solutions from NASTRAN are enough to exercise various slices. Computation time appears to be heavily dependent on the size of the mesh. Finding a balance between computational requirement and solution accuracy are relevant to doing a practical analysis. The finer the mesh, the more accurate the solution and the more difficult it is to process.

To get the solutions from NASTRAN closer to what an LSA would experience, a different constraint and loading should be used. From these first three simulations, the slices do not bend properly. The bending angle is much greater towards the wall than it is towards the free end. With many conjoined slices, each slice would experience wall stress from all directions -unless the slice happened to be at the edge of the LSA.

Chapter 4 – Normal Modes and NASTRAN

4.1 Simulation Dynamics

Modeling an LSA can be done with NASTRAN. To add to the question of how an LSA behave in a deep space orbit, it is easy to imagine a spacecraft used to repair the LSA. Something that walks across the surface of the LSA may be used to do maintenance. The vehicle that walks on the LSA will be referred to as the Robot in this report.

The Robot is assumed to apply a normal force to the LSA at two points, and it is assumed to walk along the length of the LSA from the wall. The Robot is a pendulum type walker with two legs, one moving after the other. A sample picture can be in Figure 11.
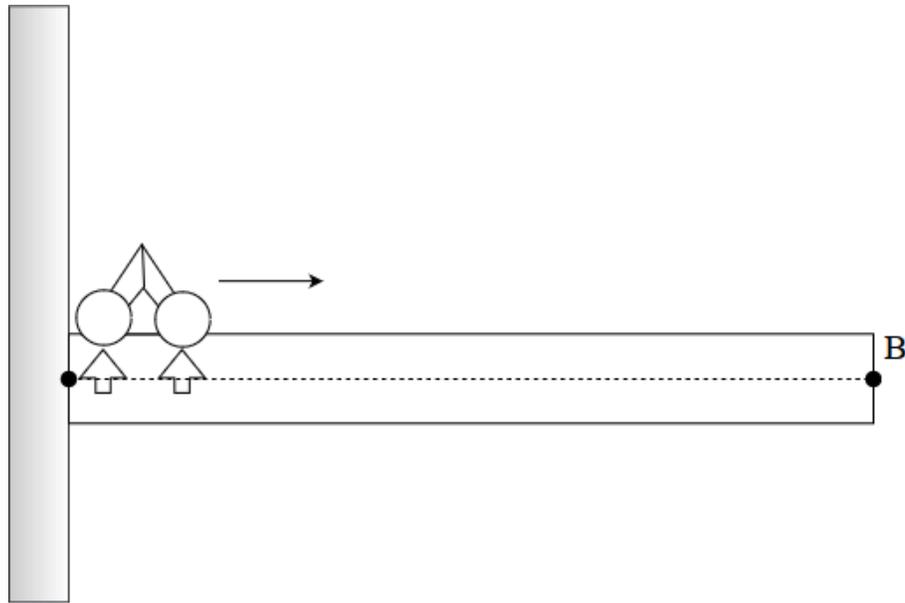
Figure 11: The Robot Walker on the LSA

Simulating a robot walking across the solar array is going to be different than simulating the bending dynamics of an LSA only experiencing forces from its orbit. It can be assumed that

the Robot walking on the surface of the LSA is the only significant force. The next closest force in magnitude to the normal force of the Robot is gravity.

The goal is to simulate a walker, or a pendulum type robot, moving across the surface of the solar panel. This is very complicated thing to try and simulated because the dynamics will not only vary with time, but with the distance that the robot has walked. Because the LSA is being assumed to have a fixed wall on one side of the length, the LSA will bend and flex more and more as the Robot walks away from the wall. With NASTRAN In-CAD being the software of choice for this simulation, there are limited options to modeling this robot. There is a way to simplify modeling distance and time in the same simulation, and that would be to run several simulations incrementing the location of the robot's force along the length of the solar array, only varying time for each increment. However, that would not model the real dynamic response of the LSA, because every step the robot takes is likely not going to happen before the solar array becomes at rest or reaches an equilibrium.

In Figure 2 is the LSA in AutoCAD Inventor. It is 1m long, 2.5mm thick, and 5mm wide. The LSA is this sized only to scale of what a real LSA might be. The exact dimensions are somewhat arbitrary.

Figure 12: The LSA in AutoCAD
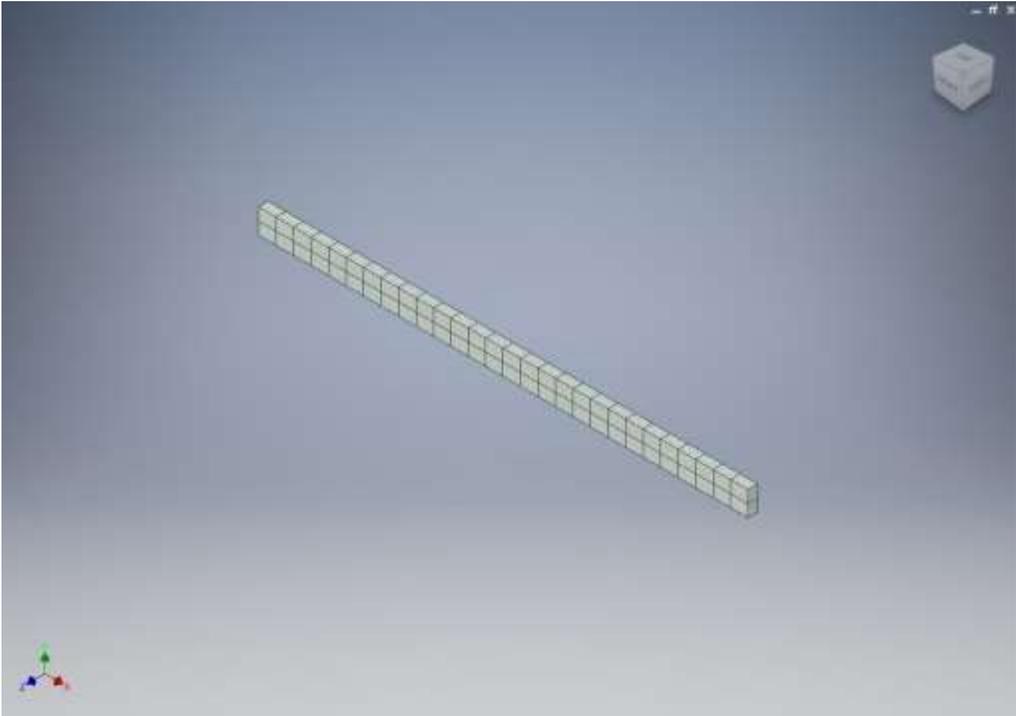
Figure 13: The LSA in AutoCAD with mesh

Figure 13 the same LSA in Figure 12, however there is a mesh applied in Figure 13. It also includes the set up for NASTRAN to run an analysis with its "Normal Modes" setting. The Normal Modes finds a user entered first number of modes frequencies and mode shapes that the model has. These mode shapes and frequencies are part of the output of the results.

The mesh took almost no time to generate at all (4 seconds). To compare, running a normal modes simulation on a beam that was 1 km long took 4 days computation time. The workstation used to generate this mesh took about a day to generate, and if it generated any slower, the progress of this project would be severely hindered. The mesh applied to the 1m LSA will be used to find the normal modes of the LSA. The mesh is used by NASTRAN to calculate the bending dynamics at all elements of the mesh. A coarseness of 172 elements and 518 nodes was chosen to the ease of calculation by the workstation being used to run NASTRAN. The mesh is bound to all surface areas of the LSA volume, and none of the mesh is spread through the internal space of the LSA model. The green color on the LSA in Figure 13 is only to mark space that is covered by mesh elements.

Some other constraints on the LSA in NASTRAN are as follows:

- Mode outputs are restricted to the first 10 flexbody modes between 0.01 and 10 hz
- NASTRAN is set to analyze the normal modes

Once the constraints of the LSA were set, NASTRAN was set to run an analysis to identify the normal mode frequencies and shapes. NASTRAN was left to run on its work station for about 4 days (the exact time can be seen in the output file) during a 1km simulation. However, the 1m simulation given in figure 14 took 4 seconds. The output file has all the details of the analysis by

NASTRAN. The first 10 modes were determined at the end of the simulation shown in Figure 14.

Frequency Versus Mode Number
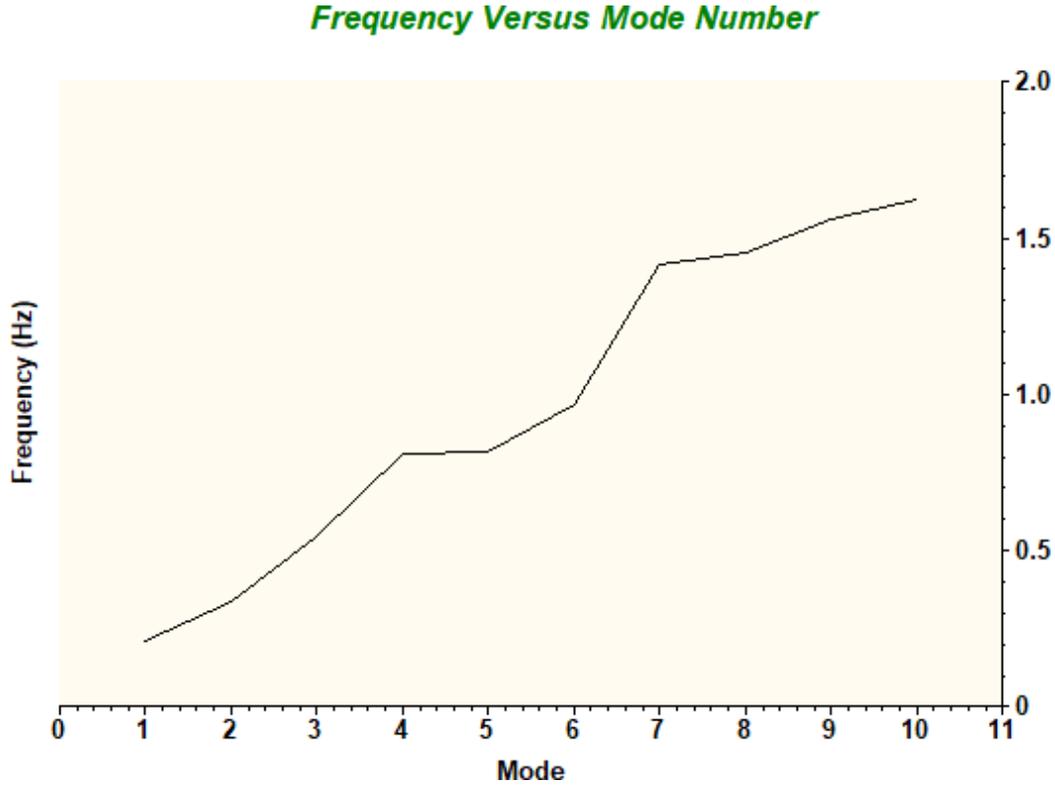
Figure 14: Frequency Versus Mode Number

Modes are certain frequencies that cause a specific object (in this can an LSA) to vibrate sinusoidally with a certain pattern that are going to be the same as other natural frequencies. Different natural frequencies that produce similar sinusoids are parts of an object's normal modes.

The first 10 modes found by NASTRAN are the frequencies at which the LSA naturally wants to bend. Each frequency has its own mode shape that is sinusoidal. Not all the sinusoidal modes of the NASTRAN output are useful for the purposes of this paper. Some of the mode frequencies are about axes that aren't of interest. Some of the mode shapes are torsional, axial, or about other axes not analyzed in this paper. However, some mode shapes (1,3,6, and 8) observed from the

output of NASTRAN appear to be about a useful axis (+Y). A few of the output mode shapes from

NASTRAN have consistent sinusoidal shapes. Figure 15 is an example of a believable mode shape,

and Figure 16 is an example of a mode shape that is not useful.
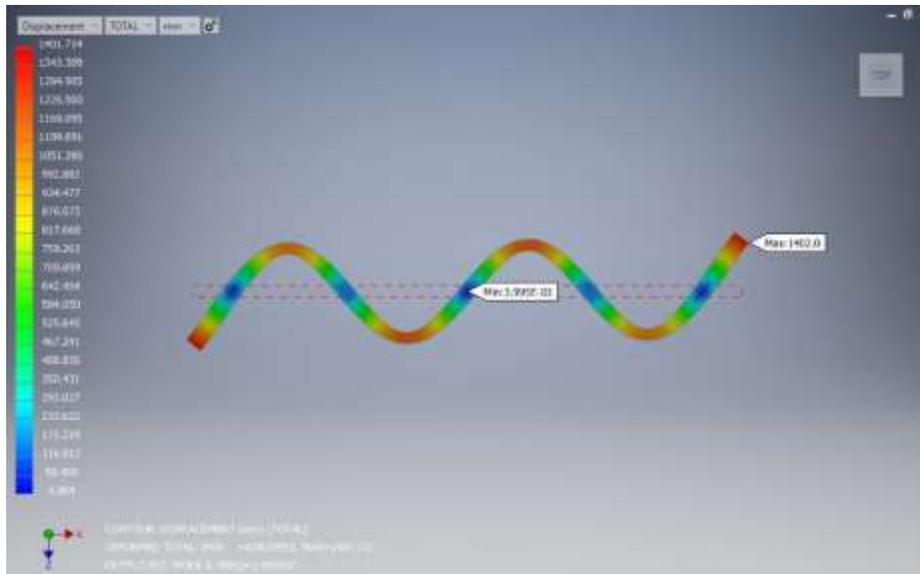
Figure 15: Mode Shape 8 from NASTRAN
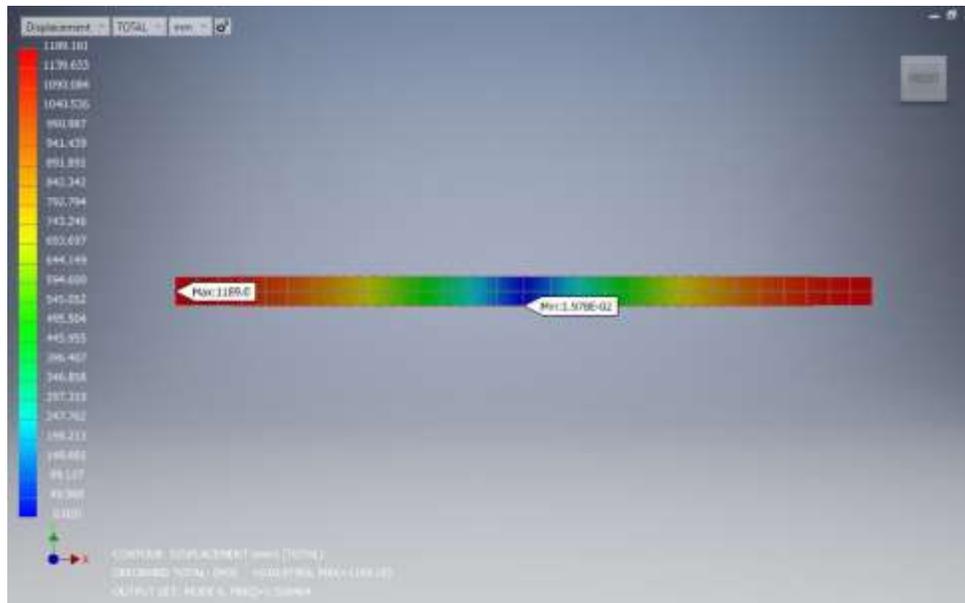
Figure 16: Mode Shape 9 from NASTRAN

Mode shape 9 appears to have axial bending, which is not of interest. None the less, normal modes have been identified, although some of the mode shapes need to be disregarded. The next step after obtaining the modes is to use the normal mode frequencies to construct plots for the

deflection of the LSA. An iterative method from Yang (2015) shall be applied. MATLAB is used to apply the iterative method. The remainder of the mode shapes can be seen in the Appendix.

Chapter 5 – MATLAB and Iterative Analysis

Yang, Ouyang, and Stacioiu (2015) is the primary paper to base the dynamic analysis of a robot walking on a large solar array, on. Like the solar array problem, is a train going over a bridge (a very common problem). A train going over a bridge is effectively a sprung mass moving across a fixed beam. The train car being the mass, the suspension being the spring, and the beam being the bridge itself. This can be compared to a mass (the robot) and the beam (the solar array). The spring may not be necessary in the solar array analysis because a walking robot does not have a suspension.

There are some critical differences between a train and a spacecraft, however. For instance, a deep space environment is going to have different thermal conditions than a train would. Comparatively, a train is going to be in a rather controlled environment, given the atmosphere of the earth compared to deep space. A solar array is going to have dramatic thermal differences on its own body in space. This will affect how the solar array vibrates, and how stiff it is. The solar array may not have an even stiffness because of thermal asymmetries across its body due to the space environment. The second biggest difference is the magnitude of gravity. A solar array in space is not exposed to the same gravity as a train on earth. It is also assumed that the dynamics are such that the robot walker can apply a force to the solar array as it walks across it. A train has a very consistent pull of gravity from the earth, and this should be kept in mind when comparing the analysis of this paper, and Yang (2015).

A two-step process is used to find the dynamic solution of the solar array (beam bending) problem as seen in Yang (2015).

The first step is Modal Superposition (MS) with numerical modes. MS will allow for a subset of Eigenmodes to be revealed. Space and time can be looked at separately, and the solution is equivalent to the total existing number of modes, using this method. The simplified solar array and train problems both include a mass and a beam, and therefore share similar equations of motion which can be found below from Yang (2015):

$$\rho A \frac{\partial^2 w(x,t)}{\partial t^2} + EI \frac{\partial^4 w(x,t)}{\partial x^2} = k_v \ (u - w(vt, t)) \cdot \delta(x - vt) \qquad 5.1$$

$$-m_v g - k_v(u(t) - w(vt, t)) = m_v \ \frac{\partial^2 u(t)}{\partial t^2} \qquad 5.2$$

These equations (5.1 and 5.2) are found in Yang (2015), and are for the mass, spring, and beam problem. Equations 5.1 and 5.2 will be different for the robot and solar array problem given the lack of spring, and the fact that the robot "walks". A walking robot would apply a force on the surface of the solar array with discontinuous steps -in stark contrast to the continuous force applied by a train. Defined in Yang (2015) are the following variables:

- E = Young's modulus of the beam

- ρ = density of the beam

- A = area of the beam cross section

- I = moment of inertia of beam's cross section

- k_v = stiffness of the spring

- m_v = mass of the sprung mass

- w = transverse deflection of the beam

- u = vertical displacement of the mass
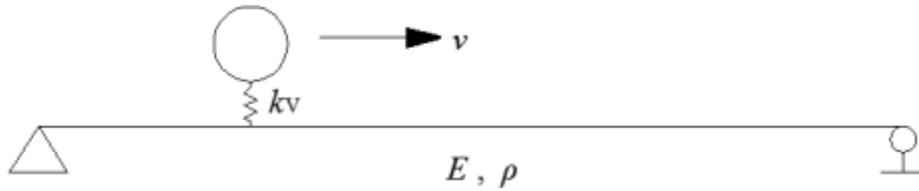
Figure 117: Mass spring model taken directly from Yang (2015)

Start of the Robots
path on the LSA

End of the Robots
path on the LSA
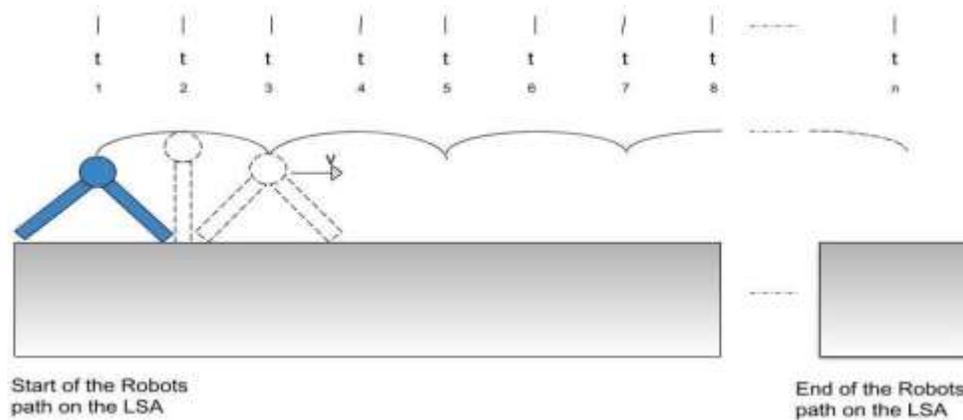
Figure 18: Arc Path of the Robot Walker Mass

Figure 18 describes how the quantity vt in all equations of motion used for the Robot/LSA problem are described in the arcs of semi circles. The vertical velocity of the mass of the robot (the plumb of the inverted pendulum) changes with respect to where it is in the circular arc. At stations t_n, the mass can be described are different extreme velocities. This change in velocity must be considered when solving the equations of motion of the Robot/LSA problem.

Equations 5.1 and 5.2 need to be transformed into their vectors of analytical modes before they can be used in MATLAB. Once in MATLAB, an iterative method adopted from Yang (2015) can be used to describe the dynamic response of the beam. The iterative method in Yang (2015) is

taken from Yang and Fonder (1996). ABAQUS is used in Yang (2015) to find the initial numerical

modes. NASTRAN was used in place of ABAQUS like in Yang (2015) to yield similar results in

this paper. The ten modes are tabulated in the Appendix and plotted in Figure 14.

To solve both equations of motion (with inputs from NASTRAN), an iterative method must

be used. The iterative scheme used in Yang (2015) and Fonder (1996) is a combination of a

convergence criteria, and the Newmark integration method. The convergence criteria is:

$$\frac{Norm(^{\Delta t}q' - ^{\Delta t}q^0)}{Norm(^{\Delta t}q' - ^0q\ )} \leq \epsilon \qquad\qquad 5.3$$

ϵ is recommended to fall between 1.0 x 10^{-5} and 1.0 x 10^{-8} in Fonder (1996). q in the

convergence equation represents a certain modal coordinate.

The equations of motion of the LSA are like the equations of motion of a mass spring

system moving across a bridge. The proof of which can be is derived in the Appendix.

Newmark integration method on the other hand uses the following equations:

$$\dot{u}_{n+1} = \dot{u}_n + \Delta t \ddot{u}_\gamma \qquad\qquad 5.4$$

$$\ddot{u}_\gamma = (1 - \gamma)\ddot{u}_n + \gamma \ddot{u}_{n+1},\ 0 \leq \gamma \leq 1 \qquad\qquad 5.5$$

In the Newmark integration method; u is the velocity and γ is a scalar. Equations 5.3, 5.4,

and 5.5 are combined to make the iterative method described in Yang (2015) and Fonder (1996).

The steps of the iterative method are repeated until the mass travels from one end of the beam, to

the other.

If there were a program that could find the dynamic solutions of the beam bending and the

numerical modes (FEA software on its own does not), then that would have been the preferred

path. However, because of the numerical limitations of FEA software at the time of writing, a combination of MATLAB and NASTRAN will be used to demonstrate the modes of a robot walking across a solar array.

This next section of the report is taken directly from Yang (2015). It should be noted that this iterative process below is what is being used in MATLAB to ingest the output from NASTRAN:

The time increments for this iterative process are time t_0 to time $t_0 + \Delta t$, and the steps are below:

1. Calculate $^0\mathbf{S}_b = \mathbf{M}(a_1\,{^0\mathbf{q}} + a_3\,{^0\dot{\mathbf{q}}} + a_4\,{^0\ddot{\mathbf{q}}})$ and $^0s_v = m_v(a_1\,{^0u} + a_3\,{^0\dot{u}} + a_4\,{^0\ddot{u}})$, where $a_1 = 1/\alpha\Delta t^2$, $a^3 = 1/\alpha\Delta t$, $a_4 = 1/(2\alpha - 1)$ and α is a Newmark integration parameter;

2. Assume the initial $\mathbf{q}$ at time $t + \Delta t$ as $^{\Delta t}\mathbf{q}^{(0)} = {^0\mathbf{q}}$;

3. Calculate $^{\Delta t}\dot{\mathbf{q}}^0 = a_2\,(^{\Delta t}\mathbf{q}^0 - {^0\mathbf{q}}) - a_5\,{^0\dot{\mathbf{q}}} - a_6\,{^0\ddot{\mathbf{q}}}$ and $^{\Delta t}\ddot{\mathbf{q}}^0 = a_1\,(^{\Delta t}\mathbf{q}^0 - {^0\mathbf{q}}) - a_3\,{^0\dot{\mathbf{q}}}$ $- a_4\,{^0\ddot{\mathbf{q}}}$, where $a_2 = \beta/2\Delta t$, $a_5 = \beta/\alpha - 1$, $a_6 = (\beta/(2\alpha - 1))\,\Delta t$, α and β are Newmark integration parameters;

4. Calculate $^{\Delta t}p^0_v = -m\,g_v + k\,{^{\Delta t}\boldsymbol{\varphi}}^T\,{^{\Delta t}\mathbf{q}^0}$;

5. The equation of motion of the mass after Newmark integration can be obtained as $(k_v + a_1 m_v)\,{^{\Delta t}u^0} = {^{\Delta t}p_v^0} + {^0s_v}$, so $^{\Delta t}u^0 = ({^{\Delta t}p_v^0} + {^0s_v})/(k_v + a_1 m_v)$;

6. Calculate $^{\Delta t}\dot{u}^0 = a_2\,(^{\Delta t}u^0 - {^0u}) - a_5\,{^0\dot{u}} - a_6\,{^0\ddot{u}}$ and $^{\Delta t}\ddot{u}^0 = a_1\,(^{\Delta t}u^0 - {^0u}) - a_3$ $^0\dot{u} - a_4\,{^0\ddot{u}}$;

7. Calculate $^{\Delta t}\mathbf{P}_b^0 = k_v\,(^{\Delta t}u^0 - {^{\Delta t}\boldsymbol{\varphi}}^T\,{^{\Delta t}\mathbf{q}^0})\,{^{\Delta t}\boldsymbol{\varphi}}$;

8. The equation of motion of the beam after Newmark integration can be obtained as

$$(\mathbf{K} + a_1\mathbf{M})\,{}^{\Delta t}\mathbf{q}' = {}^{\Delta t}\mathbf{P}_b{}^0 + {}^0\mathbf{S}_b, \text{ so } {}^{\Delta t}\mathbf{q}' = (\mathbf{K} + a_1\mathbf{M})^{-1} \cdot ({}^{\Delta t}\mathbf{P}_b^0 + {}^0\mathbf{S}_b);$$

9. The convergence criteria is as follows: $\dfrac{Norm({}^{\Delta t}q' - {}^{\Delta t}q^0)}{Norm({}^{\Delta t}q' - {}^0q)} \leq \epsilon$

where ϵ is suggested to be between 1.0×10^{-5} and 1.0×10^{-8}.

Once the time increment has completed all 9 steps, the next time step starts back on step 1. Each increment calculates ${}^{\Delta t}\mathbf{q}^1$ by ${}^{\Delta t}q^1 = {}^{\Delta t}q^0 + \eta({}^{\Delta t}q' - {}^{\Delta t}q^0)$, where η is a relaxation coefficient between 0 and 1. Step 1 through 9 are repeated until the time that the walker robot is not longer in contact with the LSA.

Using this iterative process will produce plots of displacement of the LSA.

Chapter 6 – Results and Recommendations for Future Work

It is worthy to note that if the NASTRAN portion of this analysis was done again, it would be desirable to seek out more flexible mode shapes. The more mode shapes from NASTRAN available, the more data there is to input into MATLAB. Some of the NASTRAN solutions given in this paper are not useful. The normal modes from NASTRAN take a significant amount of time to be produced for a 1km LSA. Because the results of a 1km LSA are produced in days, more time is needed to solve that case in NASTRAN. More accurate results from NASTRAN are required to produce accurate beam displacements in the iterative step of MATLAB. A 1m solution is easier to produce and analyze. It is recommended that any continuation of this papers work, generate the flexbody modes in increments of LSA sizes starting from 1m (1, 5, 10 … 100 meters). Incrementing the LSA sizes can show at what size the solutions become incomparable to a full scale model.

At the time of writing, the displacement plots for the LSA are inconclusive. The MATLAB code of the iterative portion of the solution still need to be worked out and debugged in order to produce displacement plots. All the inputs required to do the final analysis are available to the MATLAB script, however they have not been solved yet. Given more time to develop the script further, it would be possible to produce displacement plots. Once the MATLAB script successfully produced displacement plots, the data could be used to judge the vibrational modes of an LSA. Judging the vibrational modes of the LSA would be critical to designing large space habitats, or other massive space structures.

The results of this paper could be used to build other engineering analyses, such as; material choice of large space structures, orbital analyses on large space structures, sensor and actuator analyses, control law analysis, and many other engineering problems involved with building a massive orbital structure. All these aspects of engineering require some understanding of how a large space structure would tend to flex. The normal modes of a large space structure could answer, for example, where certain actuators should be placed, or where points of great stress could be.

In summary, the dynamic analysis of an LSA needs to further develop its iterative process, and to increase the accuracy of its NASTRAN output. Given more development, the displacements of the LSA would produce useful results, that could be applied to several other dynamic design problems centered around the construction and dynamics of large space structures.

References

Bainum, P. M., Reddy, A. S. S. R., Krishna, R., Diarra, C. M., and Anathakrishnan, S. (1984).

   The dynamics and control of large flexible structures (Report No. NASA-CR-173781).

   United States

Bathe, K. (2014). *Finite element procedures*. Watertown, MA: Prentice Hall, Person Education,

   Inc.

Bloch, A. M., and Marsden, J. E. (1990). Stabilization of rigid body dynamics by the energy-

   casimir method. *Systems & Controls Letters*. Vol 14(No. 4) pp. 341-346

Bloch, A. M., Krishnaprasad, P.S., Marsden, J. E., Sánchez de Alvarez, G. (1992) Stabilization

   of rigid body dynamics by internal and external torques. *Automatica*, Vol 28(No 4) pp.

   745-756

Gasbarri, P., Monti, R., Angelis, C. D., and Sabatini, M. (2014). Effects of uncertainties and

   flexible dynamic contributions on the control of a spacecraft full-coupled model. *Acta

   Astronautica*. Vol. 94(No. 1), pp. 515-526

Gasbarri, Monti, & Sabatini. (2014). Very large space structures: Non-linear control and

   robustness to structural uncertainties. *Acta Astronautica, 93*, 252-265.

Hyland, D. C., Junkins, J. L., and Longman, R. W. (1993). Active control technology for large

   space structures. *Journal of Guidance, Control, and Dynamics, Vol. 16(No. 7),* pp. 801-

   821

Kaplia, V., Tzes, A, and Yan, Q. (1999). *Closed-loop input shaping for flexible structures using time-delay control*. Proceedings of the 38th Conference on Decision & Control. Phoenix, Arizona USA, 1999. Phoenix, Arizona: IEEE

Krishnaprasad, P. S., and Marsden, J. E. (1985). Hamiltonian structures and stability for rigid bodies with flexible attachments. *Archive for Rotational Mechanics and Analysis*. Vol. 98(No. 1), pp. 71-93

Maghami, P. G., Joshi, S. M., and Price, D. B. (1995). Integrated controls-structures design methodology for flexible spacecraft. *Journal of Spacecraft and Rockets*. Vol. 32(No. 5), pp. 839-844

Mu, Tan, Wu, & Qi. (2018). Coupling dynamics of super large space structures in the presence of environmental disturbances. *Acta Astronautica, 148*, 385-395.

Thau, F. E. (1985) *An adaptive learning control system for large flexible structures* (Report No. NASA-CR-176422). New York, New York: NASA.

Wallrapp, O., and Wiedemann, S. (2002). Simulation of deployment of a flexible solar array. *Multibody Systems Dynamics.* Vol. 7(No. 1), pp 101-125

Wang, Li, & Jiang. (2017). First order coupled dynamic model of flexible space structures with time-varying configurations. *Acta Astronautica, 132*, 117-123.

Williams, T., and Cheng, X. (1999). Degrees of controllability and observability for close modes of flexible space structures. *IEEE Transactions on Automatic Control*. Vol. 44(No. 9), pp 1791-1795

Yang, J., Ouyang, H., & Stancioiu, D. (2015). An approach of solving moving load problems by

ABAQUS and ... Retrieved from

https://www.researchgate.net/publication/282655788_An_approach_of_solving_moving

_load_problems_by_ABAQUS_and_MATLAB_using_numerical_modes

Yang, F., & Fonder, G. A. (1998, December 04). An Iterative Solution Method for Dynamic

Response of Bridge–Vehicles Systems. Retrieved from

https://onlinelibrary.wiley.com/doi/abs/10.1002/(SICI)1096-9845(199602)25:23.0.CO;2-

R

Appendix

Derivation of the Equations of Motion of a fixed cantilever beam:

The equation of motion of a fixed beam are known from Y[Nikos Mourtos 2019-05-29 00:35:18]

is

$$\rho A \frac{\partial^2 w(x,t)}{\partial t^2} + EI \frac{\partial^4 w(x,t)}{\partial x^2} = k_v(u - w(vt,t)) \cdot \delta(x - vt)$$

The left side of the equation can be solved first, by setting it equal to zero. The left side

of the equation is a classical description of a cantilever beams bending dynamics, and is well

published in many statics textbooks that include flexible bodies. The free bending solution will

have a similar solution to the forced bending solution due to the principle of super position…

$$\rho A \frac{\partial^2 w(x,t)}{\partial t^2} + EI \frac{\partial^4 w(x,t)}{\partial x^2} = 0$$

$$\rho A \frac{\partial^2 w(x,t)}{\partial t^2} = -EI \frac{\partial^4 w(x,t)}{\partial x^2}$$

Where; $w(x,t) = \sum_{i=1}^{n} \psi_i(x) q_i(t)$

$$\rho A \frac{\partial^2 \psi_i(x) q_i(t)}{\partial t^2} = -EI \frac{\partial^4 \psi_i(x) q_i(t)}{\partial x^2}$$

$$\rho A \psi_i(x) \frac{\partial^2 q_i(t)}{\partial t^2} = -EI q_i(t) \frac{\partial^4 \psi_i(x)}{\partial x^2}$$

$$\rho A \frac{1}{q_i(t)} \frac{\partial^2 q_i(t)}{\partial t^2} = -EI \frac{1}{\psi_i(x)} \frac{\partial^4 \psi_i(x)}{\partial x^2}$$

$$\frac{\rho A}{-EI} \frac{1}{q_i(t)} \frac{\partial^2 q_i(t)}{\partial t^2} = \frac{1}{\psi^i(x)} \frac{\partial^4 \psi_i(x)}{\partial x^2}$$

$$\frac{\rho A}{-EI}\frac{1}{q_i(t)}\frac{\partial^2 q_i(t)}{\partial t^2} = \frac{1}{\psi_i(x)}\frac{\partial^4 \psi_i(x)}{\partial x^2} = +\lambda$$

By separation of variables…

$$\frac{1}{\psi_i(x)}\frac{\partial^4 \psi_i(x)}{\partial x^2} = +\lambda \qquad\qquad \frac{\rho A}{-EI}\frac{1}{q_i(t)}\frac{\partial^2 q_i(t)}{\partial t^2} = +\lambda$$

$$\frac{\partial^4 \psi_i(x)}{\partial x^2} = +\lambda \psi_i(x) \qquad\qquad \frac{\partial^2 q_i(t)}{\partial t^2} = +\lambda q_i(t)\frac{-EI}{\rho A}$$

$$\frac{\partial^4 \psi_i(x)}{\partial x^2} - \lambda\psi_i(x) = 0 \qquad\qquad \frac{\partial^2 q_i(t)}{\partial t^2} - \lambda\frac{-EI}{\rho A}q_i(t) = 0$$

The right side of the equation of motion of the cantilever beam can be found by the

following algebraic steps…

$$= k_v(u(t) - w(vt, t))\delta(x - vt)\int_0^l \psi_i(x)\,dx$$

By property of the Dirac delta function…

$$\int_{-\infty}^{\infty} f(t)\delta(t - T)\,dt = f(T)$$

And therefore, apply the property above to the previous step of the derivation yields…

$$= k_v[u(t) - \sum_{j=1}^{n} \psi_j(vt)q_j(t)]\psi_i(vt)$$

Combining the new forms of the left and right side of the equation of motion for forced

beam bending becomes…

$$M\ddot{q} + Kq = k_v(u(t) - \psi^T q)\psi$$

<u>Derivation of the equations of motion from a moving mass robot:</u>

The equations of motion for the moving mass robot can be found by starting with

Newton's Second Law…

$$\sum F = m\ddot{x}$$

Which, for the vertical motion of the robot, can be written as…

$$-m_v g = m_v \frac{d^2 u(t)}{dt^2}$$

Given the way that the robot walker moves, the vertical acceleration is going to change

depending on where the mass is on its arc (indicated by vt).

Tabulated Normal Mode Frequencies:

| Normal Mode Number	Frequency in Hertz
1	0.211791
2	0.3394779
3	0.5423266
4	0.8077536
5	0.8192063
6	0.9703132
7	1.413780
8	1.455097
9	1.558484
10	1.621266

Mode Shapes of the LSA:

Mode 1
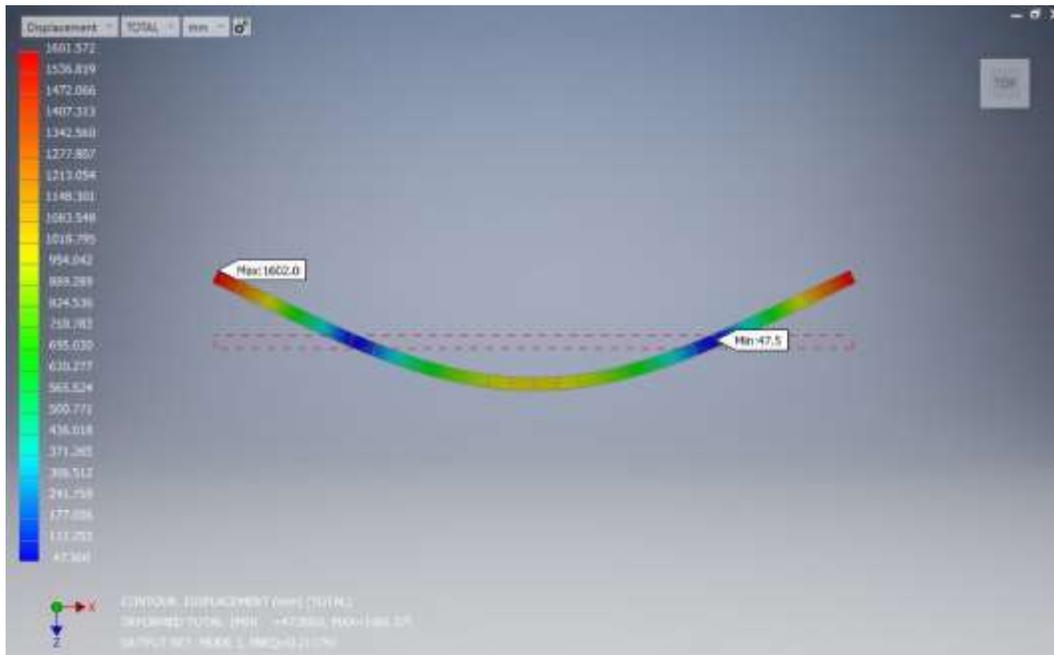
Mode 2
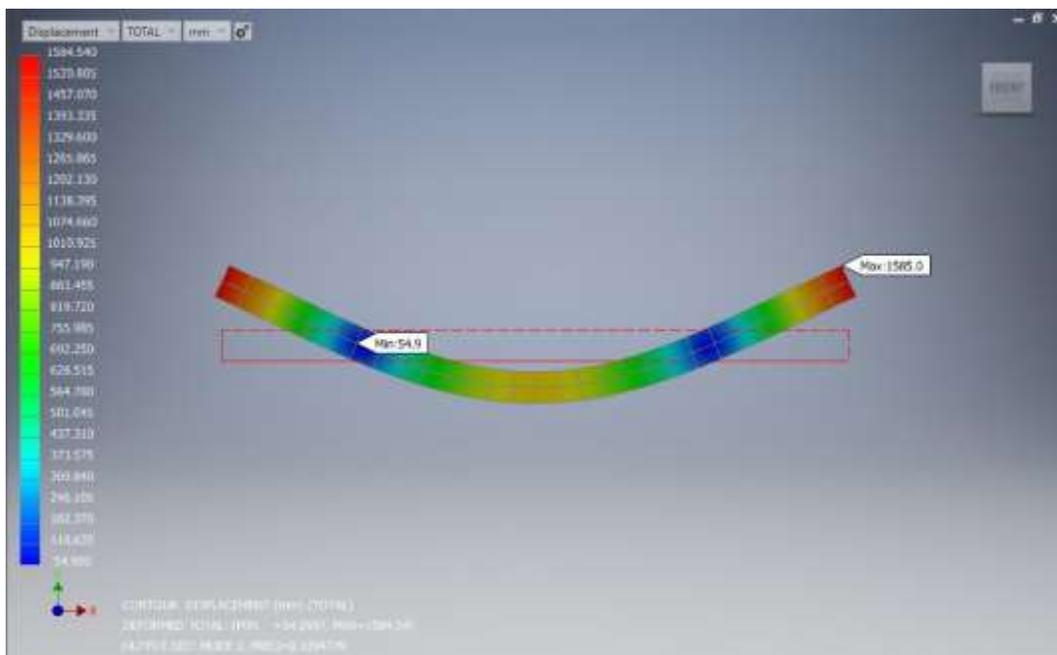
Mode 3
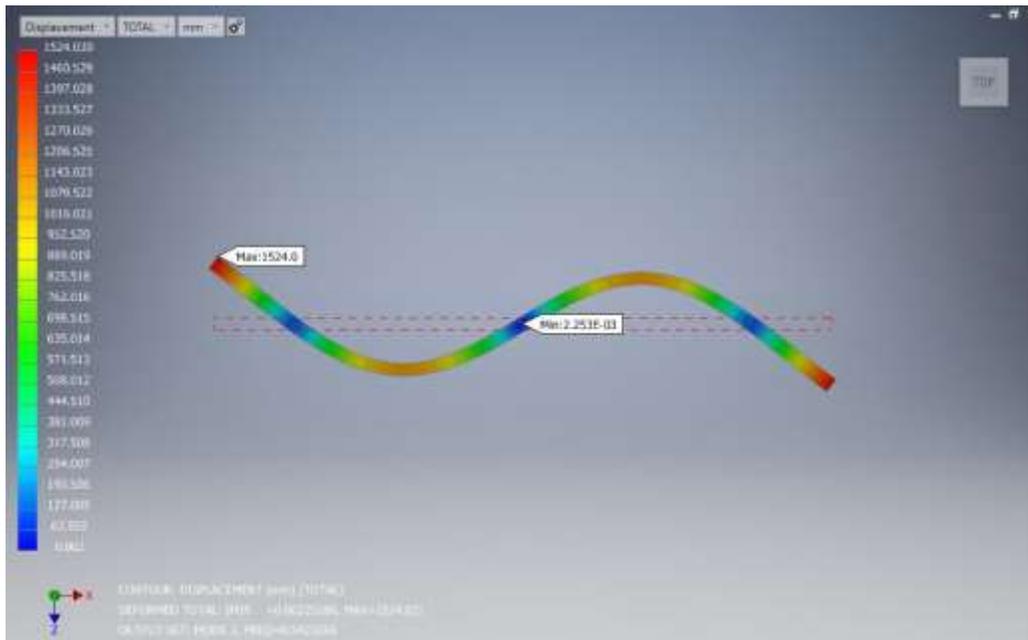
Mode 4
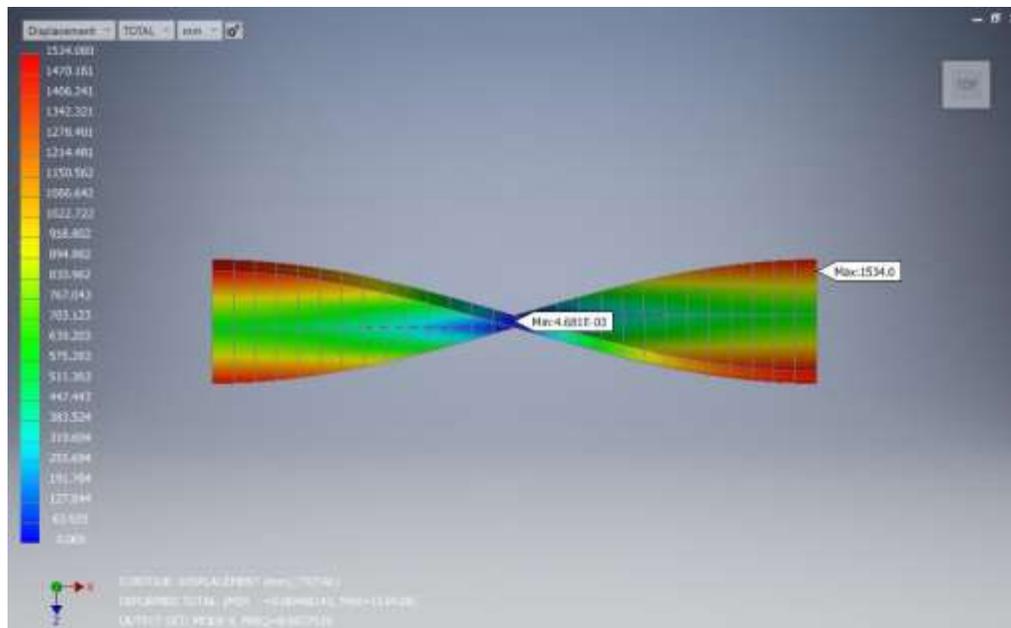
Mode 5
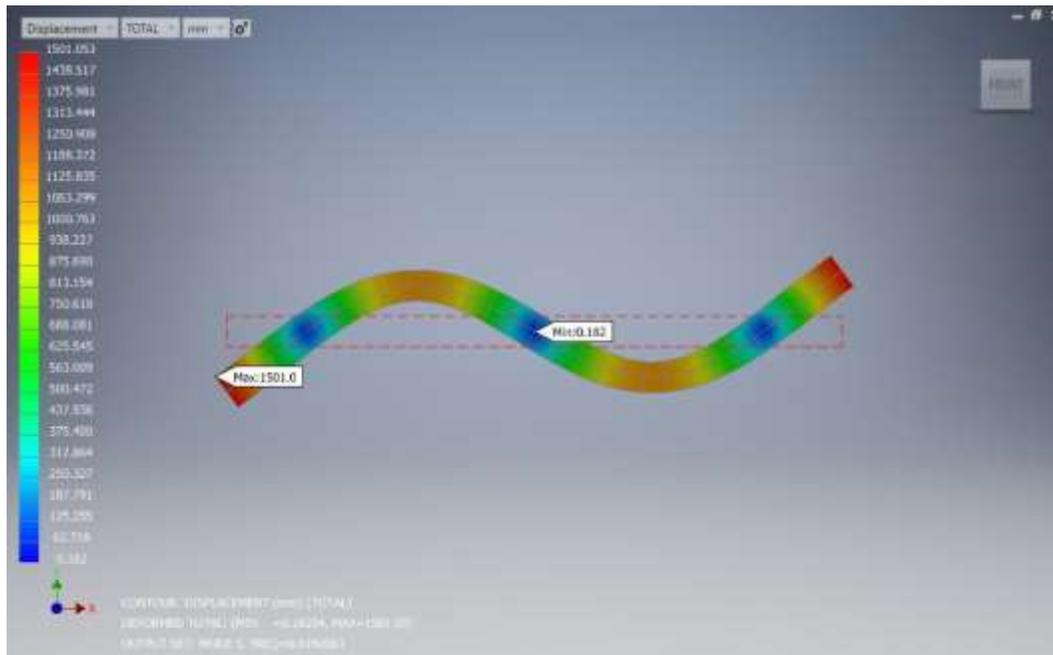
Mode 6
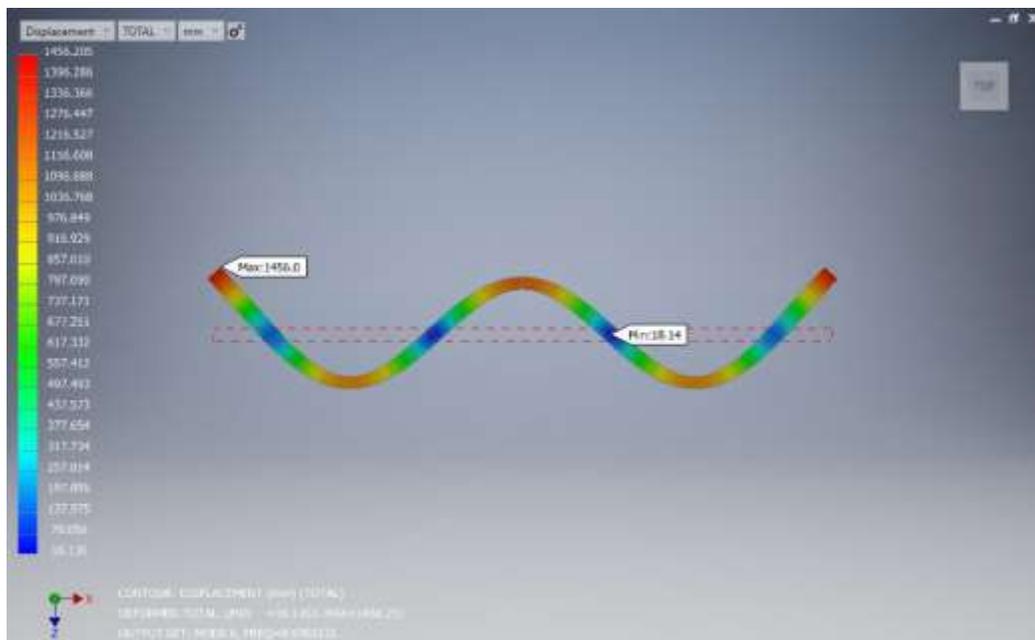
Mode 7
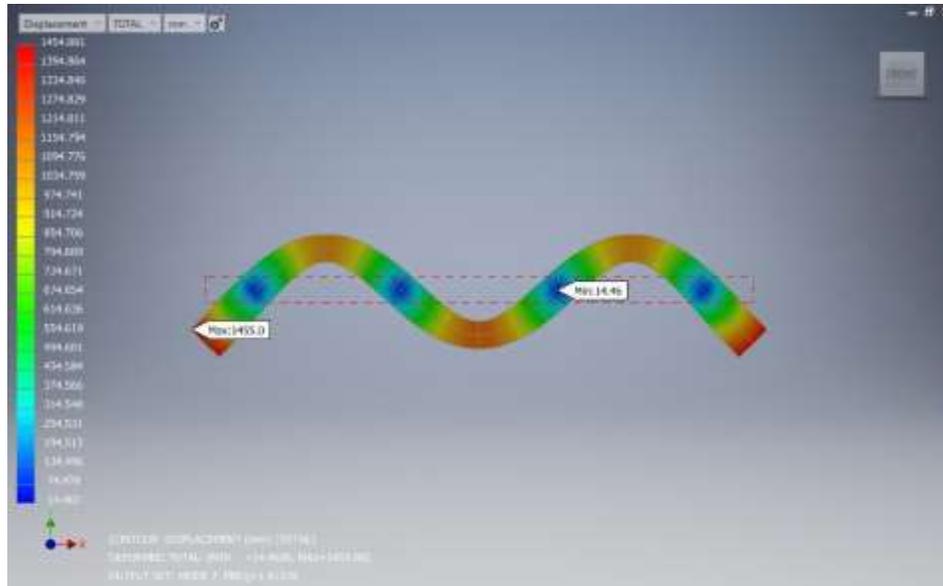
Mode 8
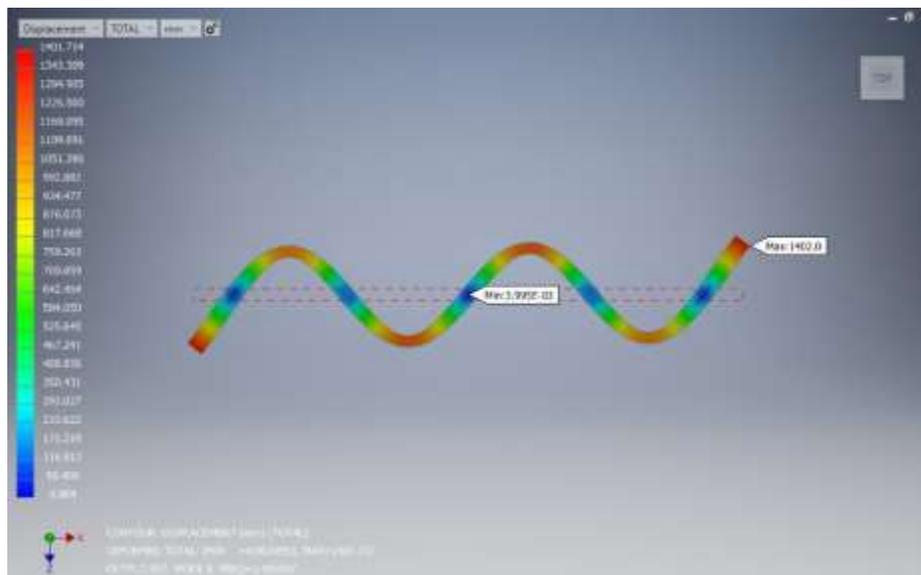
Mode 9
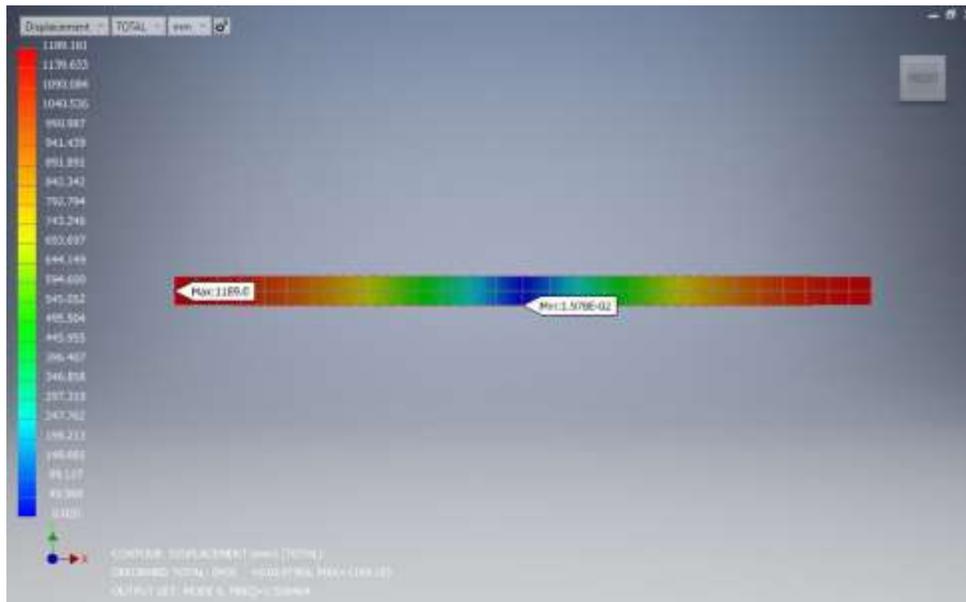
Mode 10
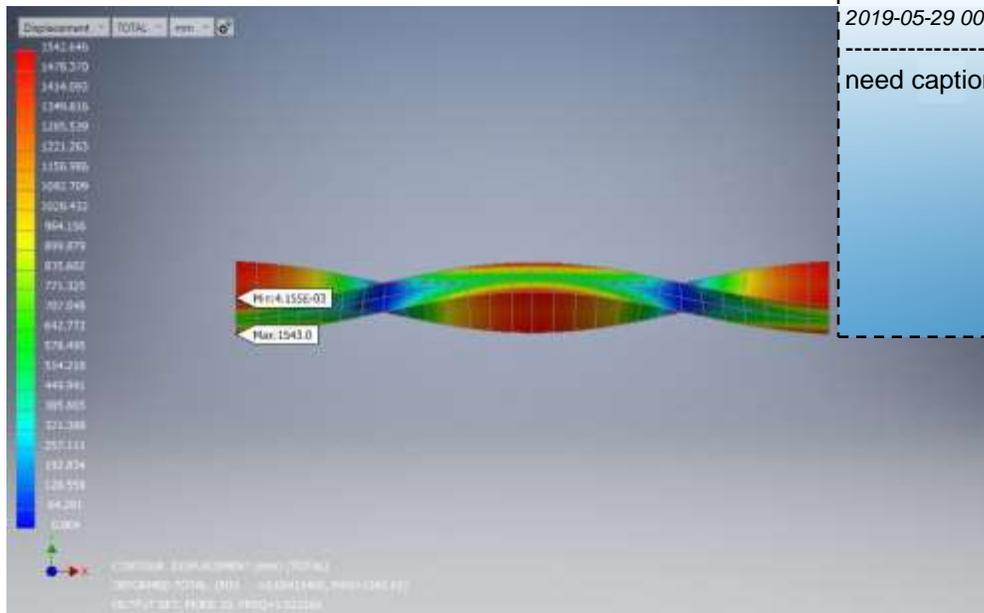
Nikos Mourtos
2019-05-29 00:39:25
----
need captions for all these figures

<p style="text-align:center;"><u>MATLAB Code</u></p>

Contents

```
%%Displacement of an LSA

%Need the normal modes from NASTRAN for the iterative scheme
```

Vectors and Variables

n is equivalent to n, q is q, u is u, p is phi, and d is delta.

```
syms n q u p pv_tstep p_tstep S_b esp;
clear all, clc, close all;

% Inputs
E = 0.0001;
I = 0.390625;
rho = 2710;
A = 0.125;
l = 30;
m_v = 1;
g = -9.81;
k_v = 1.06*10^6;

% Normal Modes
load('normal_modes.mat');

d = eye(57);
w = sqrt((E*I)/(rho*A))*(((pi())/l)^4);
M = d;
K = (w^2)*d;
```

Iterative scheme

Iterative parameters

```
alph = 0.5;


beta = 0.5;




% Convergence value
esp = 0;

% Initial conditions
p = unnamed(:,2);
q = unnamed(:,1);
qd = 0;
```

```matlab
qdd = 0;
u = 0;
ud = 0;
udd = 0;
t = 0;
del_t = 1;
net = 0.25;

% Time derivatives
%qd = diff(q)/diff(t)
%qdd = diff(q,2)/diff(t,2)
%ud = diff(u)/diff(t)
%udd = diff(u,2)/diff(t,2)

% Step 2 Initial conditions
q_tstep_n = q;
p_tstep = p;

while t<200
    if (1.0*10^-8 > esp) | (esp > 1.0*10^-5)
    % Step 1 Calculate S_b s_v
    a_1 = 1/(alph*(del_t^2));
    a_3 = 1/(alph*del_t);
    a_4 = (1/(2*alph))-1;
    S_b_init = M*(a_1*q + a_3*qd + a_4*qdd);
    s_v = m_v*(a_1*u + a_3*ud + a_4*udd);

    % Step 2 Outside the loop

    % Step 3 Calculate the rates of the q
    a_2 = beta/(2*del_t);
    a_5 = (beta/alph)-1;
    a_6 = (beta/(2*alph)-1)*del_t;
    qd_tstep = a_2*(q_tstep_n - q) - a_5*qd - a_6*qdd;
    qdd_tstep = a_1*(q_tstep_n - q) - a_3*qd - a_4*qdd;

    % Step 4 p_v calculation
    pv_tstep = -m_v*g + k_v*(p_tstep.')*q_tstep_n;

    % Step 5 Equations of motion of the mass after the Newmark integration
    % parameters have been applied
    u_tstep = (pv_tstep + s_v)/(k_v + a_1*m_v);

    % Step 6 Calculating rates
    ud_tstep = a_2*(u_tstep - u) - a_5*ud + a_6*udd;
    udd_tstep = a_1*(u_tstep - u) - a_3*ud - a_4*udd;

    % Step 7 Calculate P
    P_tstep = k_v*(u_tstep - (p_tstep.')*q_tstep_n)*p_tstep;

    % Step 8 Equations of motion of the beam after Newmark integration
    % parameters have been obtained
    q_tstepa = ((K + a_1*M)^-1)*(P_tstep + S_b_init);
```

```matlab
    % Step 9 Check the convergence criteria
    esp = norm(q_tstepa - q_tstep_n)/norm(q_tstepa - q);

    % Time step
    t = t + del_t;

    else
    % Increment for q (mode coordinate)
    q_tstep_n = q_tstep_n + net*(q_tstepa - q_tstep_n);

    % Time step
    t = t + del_t;
    end
end

disp('Hello world!')
```

```
Hello world!
```