

MODELING AND SIMULATION OF SPACE TETHER SNAPBACK

Kenneth Richard Hammett, Jr.

Certificate of Approval:

---

John Y. Hung  
Associate Professor  
Electrical and Computer Engineering

---

Thomas S. Denney, Chair  
Associate Professor  
Electrical and Computer Engineering

---

George T. Flowers  
Associate Professor  
Mechanical Engineering

---

John F. Pritchett  
Dean, Graduate School

MODELING AND SIMULATION OF SPACE TETHER SNAPBACK

Kenneth Richard Hammett, Jr.

A Thesis

Submitted to

the Graduate Faculty of

Auburn University

in Partial Fulfillment of the

Requirements for the

Degree of

Master of Science

Auburn, Alabama

May 12, 2001

MODELING AND SIMULATION OF SPACE TETHER SNAPBACK

Kenneth Richard Hammett, Jr.

Permission is granted to Auburn University to make copies of this thesis at its discretion, upon the request of individuals or institutions and at their expense. The author reserves all publication rights.

---

Signature of Author

---

Date

Copy sent to:

Name

Date

## VITA

Kenneth Richard (Rich) Hammett, Jr. was born in Mobile, Alabama, to Sue and Kenneth R. Hammett in 1968. They immediately moved to Long Beach, Mississippi to greet Hurricane Camille, and returned to Mobile just in time for Hurricane Frederic. From 1986–1995 he earned a Bachelor of Electrical Engineering degree at Auburn University (*magna cum laude*), pausing only to serve an LDS mission in Finland 1990–1992. He married Lisa Anne (Hunter) in 1995 days before entering the Graduate School, Auburn University. The hurricanes finally caught up to him while he was working on this research, when Hurricane Opal hit Auburn. He will accept donations to keep him from moving to your coastal community.

THESIS ABSTRACT

MODELING AND SIMULATION OF SPACE TETHER SNAPBACK

Kenneth Richard Hammett, Jr.

Master of Science, May 12, 2001  
(B.E.E., Auburn University, 1995)

78 Typed Pages

Directed by Thomas S. Denney

Tethers have many potential uses in space. Launching satellites to a higher orbit, towing the International Space Station, using electrodynamic tethers for orbit maintenance, and building large, lightweight structures in space have all been suggested as possible uses for tethers. When a long tether is deployed, there is a substantial possibility of its breaking or being cut by space debris, electric discharges, or mechanical defect.

This thesis examines the current computer models of tether severers. We propose our own simple mechanical model of a tether and compare simulations using this model to simulations using a similar model. The simulation is also used to replicate actual severed-tether experiments performed in the laboratory. Finally, we use our model to make predictions of severed-tether dynamics in space for two typical tether missions.

## ACKNOWLEDGMENTS

This work was partially funded by NASA/Alabama Space Grant Consortium Fellowship, 1995-1997.

The author also thanks Ken Welzyn and Connie Carrington at NASA-Marshall Space Flight Center for their assistance; his advisor, Dr. Thomas S. Denney, for financial, moral, and technical support; and his wife, Lisa Anne Hammett, who set a good example by completing her thesis.

Style manual or journal used Journal of Approximation Theory (together with the style known as “aums”). Bibliography follows van Leunen’s *A Handbook for Scholars*.

Computer software used The document preparation package T<sub>E</sub>X (specifically L<sup>A</sup>T<sub>E</sub>X) together with the departmental style-file `aums.sty`.

## TABLE OF CONTENTS

LIST OF FIGURES	x
1 REVIEW OF EXISTING LITERATURE AND MODELS	1
1.1 Description of Problem . . . . .	1
2 DERIVATION OF MATHEMATICAL MODEL USING ENERGY METHODS	7
2.1 From d'Alembert's Principle to the Euler-Lagrange Equations . . . . .	7
2.1.1 System With No Damping . . . . .	7
2.1.2 Rayleigh's Dissipation Function . . . . .	9
2.2 A Tether Model with Semi-Linear Springs and Dampers . . . . .	10
2.3 The Equations of Motion . . . . .	11
2.3.1 Kinetic Energy . . . . .	12
2.3.2 Potential Energy . . . . .	12
2.4 Orbital Dynamics . . . . .	18
2.5 Numerical Solution using MATLAB . . . . .	19
3 VALIDATION BY DIGITAL MODEL (TSS TETHER/LABORATORY)	22
3.1 Modifications to Basic Model . . . . .	22
3.2 Validation . . . . .	22
3.2.1 Method of Validation . . . . .	22
3.2.2 Results . . . . .	23
4 VALIDATION BY PHYSICAL EXPERIMENT (TSS TETHER/SEDS TETHER)	30
4.1 Modifications to Model . . . . .	30
4.2 Method and Results of Validation . . . . .	31
4.3 Conclusions . . . . .	35
5 PREDICTIONS OF SNAPBACK DYNAMICS IN ORBIT (SEDS AND TSS TETHER)	37
5.1 Modifications to Basic Model . . . . .	37
5.1.1 Extrapolation of Nonlinear Parameters . . . . .	37
5.2 TSS Simulation . . . . .	39
5.3 SEDS Simulation . . . . .	41
5.4 Conclusions . . . . .	41
6 SUMMARY AND FUTURE WORK	44
6.1 Summary of Results . . . . .	44
6.2 Ideas for Future Work . . . . .	45

BIBLIOGRAPHY	46
APPENDICES	49
A MATLAB PROGRAMS FOR CHAPTER 3	50
A.1 program 1: snap.m . . . . .	50
A.2 program 2: xdot.m . . . . .	51
A.3 program 3: runner.m . . . . .	54
A.4 program 4: runner2.m . . . . .	55
A.5 program 5: runner3.m . . . . .	56
B MATLAB PROGRAMS FOR CHAPTER 4	57
B.1 Program 1: snap.m . . . . .	57
B.2 Program 2: xdot.m . . . . .	58
B.3 Program 3: tss1.m . . . . .	58
B.4 Program 4: sed1.m . . . . .	59
B.5 Program 5: energy.m . . . . .	60
C MATLAB PROGRAMS FOR CHAPTER 5	62
C.1 Program 1: snap.m . . . . .	62
C.2 Program 2: xdot.m . . . . .	64
C.3 Program 3: tss1.m . . . . .	66
C.4 Program 4: sed1.m . . . . .	67

## LIST OF FIGURES

2.1	Tether Model . . . . .	10
2.2	Orbital Configuration . . . . .	18
3.1	1000m Tether Snapback . . . . .	24
3.2	Tether Impact Rate . . . . .	28
5.1	TSS Tether Recoil Velocity in Orbit . . . . .	40
5.2	SEDS Tether Recoil Velocity in Orbit . . . . .	42

## CHAPTER 1

### REVIEW OF EXISTING LITERATURE AND MODELS

#### 1.1 Description of Problem

A space tether is any rope or string-like item used in space. The specific space tethers discussed in this paper are long tethers, typically ten to twenty kilometers or longer. The current definitive space tether reference is the *Tethers In Space Handbook*, Third Edition [1]. It is available in Adobe Portable Document Format by download from <http://cfa-www.harvard.edu/spgroup/handbook.html>. This handbook contains information about previous tether experiments, plans for future tether experiments, and references and contacts for further information.

Two types of tether will be examined in this thesis. One type is called a Tethered Satellite System (TSS) tether, which consists of several copper wires in a central matrix of synthetic fibers, surrounded by various synthetic insulating and strengthening jackets[2]. It is similar in appearance and feel to a very flexible coaxial television cable and is used primarily in plasma electrodynamics experiments. A conducting tether has been proposed as a method of periodically boosting the orbit of the International Space Station (ISS), as described in a recent article in *IEEE Spectrum* [3]. The paper by Vas, et al [4] suggests that this type of reboost could save up to a billion dollars in propellant over ten years for the ISS with only 180 days per year of use.

The other type to be examined is called a Simple Expendable Deployer System (SEDS) tether. This is a 0.75 mm multifilament tether woven from SPECTRA-1000. Similar in appearance to a synthetic kite string [5], it is a multipurpose mechanical

tether. The SEDS experiments used this tether to launch satellites from orbital vehicles. It would allow a manned launch vehicle such as the space shuttle to launch a satellite to a much higher orbit without requiring rockets or even energy-loaded springs. Tethers may also be used to tow large space structures such as the International Space Station into a new orbit (See [6] for a study of this method).

When either type of tether is deployed, there is a substantial possibility of its breaking or being cut. Tethers have been cut by space debris and by electric discharges. The most recent example is the Tethered Satellite System 1 Reflight (TSS-1R). The space shuttle deployed this electrodynamics experiment in 1996. After the tether had been deployed to a length of 19.7 km, a fault in the tether insulation allowed a high-voltage arc between the tether and the deployer boom, melting and thus severing the tether [7]. Several papers in particular have investigated the probability of a tether sever by space debris; Welzyn and Robinson [8] study risks to tethers in general, while Hayashida, Robinson, and Hill [9] study SEDS tethers exclusively. When a tether under tension is severed, the tether remnants rapidly snap back, like a rubber band.

It is important to understand snapback dynamics to insure the safety and comfort of crews of manned space vehicles, and to protect the expensive vehicles and other equipment. For example, if a twenty kilometer SEDS tether were to fail while launching a satellite from the space shuttle, the entire length of the tether could snap back toward the orbiter. Although the SEDS tether does not store enough elastic energy to damage the orbiter or change its orbit, the tether could encircle the orbiter and prevent the cargo bay doors from closing. This would require an astronaut to manually cut and remove the tether from the cargo bay. If, soon after the tether severs, the tether is cut or

released from the shuttle and the shuttle executes an appropriate maneuver, the shuttle can avoid the recoiling tether entirely. Accurate models of these snapback dynamics can help determine how long to wait before cutting the tether free, and the minimum maneuver necessary to move the shuttle from the path of the tether. This gives the pilot leeway to verify that the tether has been severed before releasing the remnant (and thus canceling the experiment).

Beletsky and Levin [10] make a convincing argument that tether snapback dynamics in orbit are dominated by very few forces. Tether internal dynamics, the gravity of earth (linear approximation), the gravity gradient due to earth's gravity, and coriolis forces from orbital dynamics must be accounted for at minimum. When studying steady-state dynamics of large tether structures (spiderwebs, etc.), many other forces must be taken into account: earth gravity harmonics, lunar gravity, solar and jovian gravity. But for the time scale of a tether snap-back, the internal dynamics are the dominant factor.

Many models and simulations of tethers and tethered-satellite systems exist in the literature. Although each model has different intended uses, the models can be broadly categorized as either generic tether models or models of some specific structural component or regime.

Generic tether models and simulations generally model the tether during deployment, stationkeeping, and/or retrieval of entire tethered-satellite systems. Notable among these is the Space Tether Dynamics Simulation by Dr. Levin, developed for use with his textbook [10]. The program can simulate deployment, stationkeeping, retrieval, thrust from either end-body, and several other functions. All parameters are specifiable by the user, and there is a sensible default function and parameter set for

ease of learning. BeadSim by Joseph Carroll is another widely-used generic tether simulation. Both simulations discretize the tether using a bead method similar to the one used in this thesis.

A 1990 paper by Pasca, et al [11], uses a continuum elastic model of the tether. The authors assume a standard tethered-satellite configuration (tether taut between first end-body and second end-body directly overhead) and examine motions of the system as harmonic vibrations from this configuration. They calculate analytic solutions to a simplified problem (assuming constant stress along the length of the tether) and numerically perturb those analytic solutions. Their solution reveals coupling between longitudinal motion and transverse motion exclusively in the orbital plane due to orbital mechanics and mechanics of the tether. Out-of-plane transverse motion of the tether is only weakly coupled to the other modes through tether mechanics. This will be important in the orbital studies in this thesis.

Other more specialized models exist in the literature, such as a nonlinear model of a spinning tether [12], a model consisting of hinged rods to account for bending energy [13], or a group of massive but inextensible rods [14]. Other models attempt to capture the complex mechanics of a braided, multipart tether, such as the group of models in the paper by Fanti [15]. Tether survivability models are very important for economic analyses, and have thus received much attention in recent years [16, 17, 9, 8]. Von Flotow [18] compiled a detailed study of several types of tether motion; in particular he measures and models the non-linear stress/strain relationship in certain tether materials.

All of these models, as diverse as they are, have one thing in common: they model the tether under tension. Some of them allow the tether to go slack, but then simply

ignore the tether and treat the end-bodies as independent orbital bodies. Since we wish to predict the behavior of the tether itself when it is severed from one of the end-bodies, the models listed above are inadequate.

Very few slack tether models exist in the literature. A NASA-sponsored study of slack tether dynamics and contract reports are the chief source of information about slack tether models. Dr. Giuseppe Colombo's groundbreaking work on this contract is summarized in a 1982 report, "Study of Certain Tether Safety Issues" [19]. His work was continued by Dr. Enrico Lorenzini, whose quarterly reports #3 and #5, entitled "The Investigation of Tethered Satellite System Dynamics" [20, 21], contain summaries of his slack tether simulation work.

Ken Welzyn and Jennifer Robinson present several analytic models of tether sever dynamics in a 1995 paper, "Severed Tether Dynamics and Probability." [8] The paper "Effects of Severed Tethers in Space" by Angrilli, et al [22] compares laboratory test results with three slack tether models: a continuum model, a simple energy model, and a discrete element model similar to the one developed in this thesis. The results in that paper will also be used to validate this model.

Our primary model as developed in Chapter 2 is virtually identical to Angrilli's model, a simple model where the mass is divided among discrete beads connected by a semi-linear spring and damper system. The spring and damper system is linear when the tether segment is under tension, and exerts no force when the tether segment is slack. This is according to the principle that you cannot push a string. Although our results match Angrilli's results very well in the validation in Chapter 3, we are not able to match the results of the NASA physical experiments in Chapter 4 with this semi-linear spring

model. For this reason we introduce a term proportional to the cube of tether extension, as suggested by von Flotow in [18]. This will be discussed in more detail in Chapter 4.

An important and ongoing effort to study tether dynamics is the Tether Physics and Survivability Experiment (TiPS) [26]. This is an experiment to observe the long-term (more than one year) survivability of a deployed tether, and the stability of a simple tethered-satellite system over the same time period. The tether used in this experiment is similar to the SEDS tether.

In this thesis we will derive a model of tether snapback and use it to predict snapback behavior in orbit. First, the development of the model equations of motion (of a tether with semi-linear spring elements) using energy methods will be shown. Next, our model will be validated using another model of tether breakage in the laboratory. We then validate it using laboratory test data, and substitute the non-linear spring model. Finally, predictions of orbital tether snapback based on our model using the non-linear spring elements will be presented.

CHAPTER 2  
DERIVATION OF MATHEMATICAL MODEL  
USING ENERGY METHODS

In this chapter, we use energy methods such as the Euler-Lagrange Equations and their corollary, Rayleigh's Dissipation function, to reduce a simple mechanical model of a tether (with internal and external forces) to a system of ordinary differential equations. We then demonstrate how to use MATLAB to calculate a numerical solution to the system of equations for a specified set of initial conditions and time interval.

### 2.1 From d'Alembert's Principle to the Euler-Lagrange Equations

In this section we develop the Euler-Lagrange Equations for a general system. This development will provide the framework for our tether models.

#### 2.1.1 System With No Damping

The starting point of energy methods places inertia in the role of a reactive force. This changes an equation related to Newton's second law of motion ( $F = m \cdot a$ ) to the following seemingly trivial variant, d'Alembert's Principle:

$$F - ma = 0. \tag{2.1}$$

The following development (through Equation (2.10)) is taken from Greenwood [23], pages 156-7.

When applied to a system of  $N$  particles (with mass  $m_i$  and position vector  $r_i$ ) and multiplied by a set of reversible, consistent virtual displacements  $\delta r_i$  this expression becomes:

$$\sum_{i=1}^N (F_i - m_i \ddot{r}_i) \cdot \delta r_i = 0. \quad (2.2)$$

By writing this expression for the variation of the kinetic energy of the system:

$$\delta T = \delta \left( \frac{1}{2} \sum_{i=1}^N m_i \dot{r}_i^2 \right) = \sum_{i=1}^N m_i \dot{r}_i \cdot \delta \dot{r}_i \quad (2.3)$$

and noting that

$$\frac{d}{dt} \left( \sum_{i=1}^N m_i \dot{r}_i \cdot \delta r_i \right) = \sum_{i=1}^N m_i \ddot{r}_i \cdot \delta r_i + \sum_{i=1}^N m_i \dot{r}_i \cdot \delta \dot{r}_i \quad (2.4)$$

we obtain

$$\delta T + \sum_{i=1}^N F_i \cdot \delta r_i = \frac{d}{dt} \left( \sum_{i=1}^N m_i \dot{r}_i \cdot \delta r_i \right). \quad (2.5)$$

Define  $\delta W = \sum_{i=1}^N F_i \cdot \delta r_i$  as the virtual work of the applied forces  $F_i$  and integrate Equation (2.5) from time  $t_0$  to  $t_1$  to reach

$$\int_{t_0}^{t_1} (\delta T + \delta W) dt = \left[ \sum_{i=1}^N m_i \dot{r}_i \cdot \delta r_i \right]. \quad (2.6)$$

By forcing the virtual displacements to vanish at times  $t_0$  and  $t_1$ , we cause the right hand side of Equation (2.6) to disappear, yielding

$$\int_{t_0}^{t_1} (\delta T + \delta W) dt = 0. \quad (2.7)$$

For this portion of the analysis we are assuming that there are no dissipative forces in our system. Thus the virtual work  $\delta W$  is performed by purely conservative forces and can be replaced by the variation of a potential function  $-\delta V$ . By defining the Lagrangian as

$$L = T - V \tag{2.8}$$

and interchanging the operations of variation and integration, we can rewrite Equation (2.7) to the form

$$\delta \int_{t_0}^{t_1} L dt = 0. \tag{2.9}$$

This equation holds true if and only if the left hand side is stationary with respect to generalized coordinates  $q_i$  in the state space. Placing this condition on Equation (2.9) results in the Euler-Lagrange equations, in the standard form of Lagrange's Equation:

$$\frac{d}{dt} \left( \frac{\delta L}{\delta \dot{q}_i} \right) - \frac{\delta L}{\delta q_i} = 0, \quad (i = 1, 2, \dots, N) \tag{2.10}$$

### 2.1.2 Rayleigh's Dissipation Function

If there are external applied forces (non-conservative) in the system, they can be substituted on the right side of Equation (2.10). Damping forces are energy dissipating forces, usually taken to be proportional to the velocities in the system. In general, the forces on each particle  $i$  can be expressed as a function of all of the generalized velocities

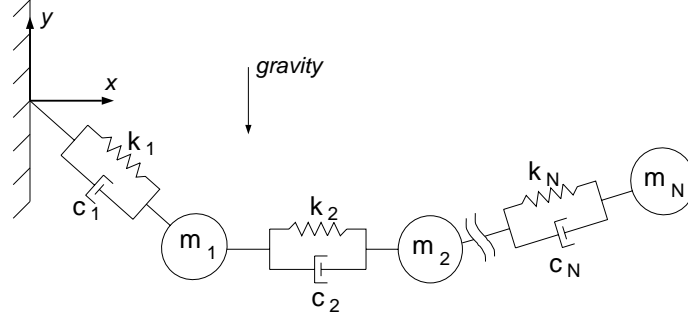


Figure 2.1: Tether Model

of the system [23]:

$$Q'_i = - \sum_{j=1}^N c_{ij}(q, t) \dot{q}_j, \quad (2.11)$$

where  $Q_i$  is the dissipative force on particle  $i$ ,  $c_{ij}$  is a coefficient linking the  $j^{\text{th}}$  velocity with the  $i^{\text{th}}$  particle, and  $q$ ,  $\dot{q}$ , and  $t$  as above. By defining Rayleigh's dissipation function:

$$F = \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N c_{ij} \dot{q}_i \dot{q}_j, \quad (2.12)$$

a term can be added to the Euler-Lagrange equations (Equation (2.10)) by combining Equations (2.10), (2.11), and (2.12) as follows:

$$\frac{d}{dt} \left( \frac{\delta L}{\delta \dot{q}_i} \right) - \frac{\delta L}{\delta q_i} + \frac{\delta F}{\delta \dot{q}_i} = 0, \quad (i = 1, 2, \dots, N) \quad (2.13)$$

## 2.2 A Tether Model with Semi-Linear Springs and Dampers

A mechanical model of the tether is shown in Figure 2.1. The tether is fixed at one end to a restraint which allows free rotation but no translation. The other end is held fixed under tension along the x axis. The tether is broken up into  $N$  segments, with the

mass of each segment  $m_i$  concentrated into a bead away from the free end of the tether. The other mechanical properties of the tether are represented by a parallel spring and damper combination. Each spring originally obeys Hooke's law with a spring constant of  $k_i$ ; similarly, each damper has a damping coefficient of  $c_i$ . Both the spring and damper are nonlinear, however, in that they have no effect when the string is not under tension. Therefore, when a segment length is less than its natural, unstretched length, the spring and damper exert no force.

Each bead has two coordinates,  $x_i$  (horizontal in Figure 2.1, originally along the length of the tether) and  $y_i$  (vertical). Gravity pulls directly in the negative  $y$  direction.

### 2.3 The Equations of Motion

In order to find the equations of motion using the methods described in Section 2.1, the Lagrangian  $L$  and the dissipation function  $F$  must be defined. The Lagrangian is, as defined in Equation (2.8), the kinetic energy minus the potential energy. The kinetic energy of the system as shown in Figure 2.1 is contained entirely in the kinetic energy of the beads  $m_i$ , since the spring and damper combination is massless and does not approach relativistic velocities. The potential energy in the system is stored both in the one-way springs and the gravitational field. The only energy dissipation in the system occurs in the one-way dampers.

### 2.3.1 Kinetic Energy

The kinetic energy  $T$  of the system is the sum of each bead's kinetic energy, as follows:

$$T = \sum_{i=1}^N \frac{1}{2} m_i \sqrt{\dot{x}_i^2 + \dot{y}_i^2}. \quad (2.14)$$

### 2.3.2 Potential Energy

#### Gravitational Potential Energy

For a tether in earth orbit, the force and potential energy of gravity depend on distance from the earth's center of mass. The coordinate framework for the system being analyzed here is illustrated in Figure 2.2. Several simplifying assumptions are used. The mass of the entire orbiting system is assumed to be much less than the mass of the earth. The earth is spherical with radius  $a$ ; the gravitational contributions of all other bodies are ignored. The datum of the gravitational system is taken to be the line  $y = 0$ . The total gravitational potential energy (GPE) is the sum of the GPE of each bead:

$$V_g = \sum_{i=1}^N -m_i \frac{GM}{a + y_i} \quad (2.15)$$

where  $G$  is the gravitational constant,  $M$  is the mass of the earth,  $m_i$  is the mass of the  $i$ th bead,  $a$  is the orbital altitude (from the center of the earth), and  $y_i$  is the  $y$  coordinate of the  $i$ th bead.

### Gravitational Potential Energy in Testbed

Near the surface of the earth the equation can be simplified by substituting the standard acceleration of gravity  $g = \frac{GM}{a^2}$  to reach the following formulation:

$$V_g = \sum_{i=1}^N m_i g y_i. \quad (2.16)$$

### Elastic Energy

The elastic energy is slightly more complicated, because it depends on the distance between beads (and between the first bead and the fixed support). The spring is a one-way spring, so there are two different expressions for the energy: one for when it is extended, one for when it is not. When extended, the energy of the spring can be expressed in a form analogous to the kinetic energy (Equation (2.14)),  $\frac{1}{2}k\Delta^2$ , with  $k$  being the linear spring constant and  $\Delta$  being the distance the spring is stretched beyond its normal length ( $h$ ).

Define  $\Delta$  in terms of bead coordinates  $x_i$  and  $y_i$  and natural length  $h$  as follows:

$$\begin{aligned} \Delta_1 &= \left( \sqrt{x_1^2 + y_1^2} \right) - h \\ \Delta_i &= \left( \sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2} \right) - h, \quad (i = 2, 3, \dots, N) \end{aligned} \quad (2.17)$$

The spring constant  $k$  is the same for each segment. It is derived from a bulk property of the tether, the elastic or Young's Modulus, with symbol  $E$ . For the tether, manufacturers and testers report a quantity  $EA$ , which is the elastic modulus multiplied by the cross-sectional area of the object. In reality, this is a property of the tether that

includes many dynamics other than bulk properties, such as the weaving pattern. A spring constant  $k_{tot}$  can be derived for the entire tether from  $EA$  and the total length of the tether  $l$  by dividing:

$$k_{tot} = \frac{EA}{l} \quad (2.18)$$

The spring constant  $k$  for each segment of tether, then, is  $k_{tot}$  multiplied by the number of equal-length segments,  $N$ . The same expression can be reached by substituting  $\frac{l}{N}$  (the length of each equal tether segment) into Equation (2.18) in place of  $l$ :

$$k = k_i = \frac{(EA)(N)}{l} \quad (2.19)$$

When including the null spring for the unstretched case, the energy for each spring is

$$V_{si} = \begin{cases} \frac{1}{2}k\Delta_i^2 & \Delta > 0 \\ 0 & otherwise \end{cases} \quad (2.20)$$

and the total spring energy for the tether is

$$V_s = \sum_{i=1}^N V_{si}. \quad (2.21)$$

## Damping

He and Powell analyzed the effect of tether internal damping in the paper “Tether Damping in Space [24].” They assume that only the longitudinal vibrational mode possesses intrinsic damping, and that other modes experience damping only through mechanical coupling with the longitudinal mode. They analyze three modes of damping (internal viscous, external viscous, and structural) using a continuum model of tether

mechanics. They compare the results of their analysis on a set of experiments performed on TSS tethers (3.7m – 17.68m long) using small end-masses (.226kg – 6kg) to measure damping in the mass-spring tether vibrational mode.

They find a best-fit combination of damping modes to match the data, and extrapolate this for longer tethers (on the order of 100m) with a 1000kg end mass. Their results indicate that a damping coefficient ( $\xi$ ) of 1.8 % gives a conservative damping model for the TSS tether. Angrilli [22] uses a measured value of 4 %. We will use this slightly less conservative value in all of our simulations with linear viscous damping in the tether segments between the lumped-mass beads, in order to match Angrilli's results.

The damping expression will be developed as a force relationship instead of an energy form. The choice of the bead coordinates complicates the damping expression, since damping is a linear function of the relative bead velocities, and the coordinates are with respect to a fixed coordinate frame. More precisely, damping is a linear function of the range rate of two beads, that is, the rate at which the distance between two beads changes.

A tether segment will be numbered according to the highest numbered bead attached to it (see Figure 2.1. The segment from the fixed end of the tether to the first bead is segment 1. Now, define the relative position vector  $P$  as a vector from the prior bead (closer to fixed end of tether) to the current bead. The first segment of the tether is a special case, as it is attached to the fixed point  $0\hat{x} + 0\hat{y}$ , so the position vector to the first bead is just  $P_1 = x_1\hat{x} + y_1\hat{y}$ . In general, the position vector for the  $i$ th segment of tether is

$$P_i = (x_i - x_{i-1})\hat{x} + (y_i - y_{i-1})\hat{y} \quad (2.22)$$

This vector defines the line of action for the tether forces (spring and damping forces) on the bead.

For the first bead, the range rate of the bead is the velocity of the bead ( $v_1$ ) projected onto the position vector ( $P_1$ ), or

$$rr_1 = v_1 \cdot \frac{P_1}{|P_1|} \quad (2.23)$$

where  $v_1$  is the velocity of the first bead  $v_1 = \dot{x}_1\hat{x} + \dot{y}_1\hat{y}$ . In general  $v_i$  is the velocity of bead  $i$  relative to bead  $i - 1$ ,

$$v_i = (\dot{x}_i - \dot{x}_{i-1})\hat{x} + (\dot{y}_i - \dot{y}_{i-1})\hat{y} \quad (2.24)$$

and the range rate of bead  $i$  can be written as

$$rr_i = v_i \cdot \frac{P_i}{|P_i|}. \quad (2.25)$$

The magnitude of the damping force can now be calculated. The segments are all identical, so each segment's damping constant  $c_i$  will be represented by  $c$ . The magnitude of the damping force is then:  $c \cdot rr_i$ .

As mentioned above, the  $i$  indicates which tether segment is generating the force, not which bead is being acted on. Each bead (except for the terminal bead) has two segments of tether acting directly on it. Thus, the force on the bead is the vector sum of the forces from the two tether segments. Now the direction of the force must be accounted for. Since the vector giving the relative position of the beads has already been calculated, all that remains is to create a unit vector and determine the proper

sign. Because of the method of derivation, the force from the tether segment before the bead (on the side toward the fixed end of the tether) is negated; the other force maintains its calculated sign.

For the  $i$ th bead, the total force is therefore

$$Fd_i = -c \frac{P_i}{|P_i|} + c \frac{P_{i+1}}{|P_{i+1}|}, \quad (2.26)$$

except for the terminal bead, which does not have the right-most term.

To write the total damping in terms of Rayleigh's dissipation function, the entire expression for the force must be expressed as a function of the  $\dot{x}_i$ 's and  $\dot{y}_i$ 's. When expressed in this form, the damping force on  $i$ th bead ( $Fd_i$  from above) becomes:

$$\begin{aligned} Fd_i &= \dot{x}_{i-1} \left( c(x_i - x_{i-1}) \frac{P_i}{|P_i|^2} \right) \\ &+ \dot{x}_i \left( -c(x_i - x_{i-1}) \frac{P_i}{|P_i|^2} - c(x_{i+1} - x_i) \frac{P_{i+1}}{|P_{i+1}|^2} \right) \\ &+ \dot{x}_{i+1} \left( c(x_{i+1} - x_i) \frac{P_{i+1}}{|P_{i+1}|^2} \right) \\ &+ \dot{y}_{i-1} \left( c(y_i - y_{i-1}) \frac{P_i}{|P_i|^2} \right) \\ &+ \dot{y}_i \left( -c(y_i - y_{i-1}) \frac{P_i}{|P_i|^2} - c(y_{i+1} - y_i) \frac{P_{i+1}}{|P_{i+1}|^2} \right) \\ &+ \dot{y}_{i+1} \left( c(y_{i+1} - y_i) \frac{P_{i+1}}{|P_{i+1}|^2} \right) \end{aligned} \quad (2.27)$$

Note that this equation must be slightly modified for the two terminal beads.

$$\begin{aligned} Fd_1 &= \dot{x}_i \left( -c(x_i) \frac{P_i}{|P_i|^2} - c(x_{i+1} - x_i) \frac{P_{i+1}}{|P_{i+1}|^2} \right) \\ &+ \dot{x}_{i+1} \left( c(x_{i+1} - x_i) \frac{P_{i+1}}{|P_{i+1}|^2} \right) \end{aligned}$$

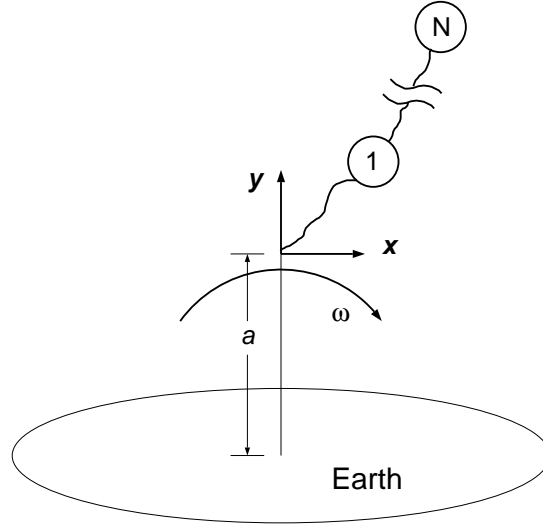


Figure 2.2: Orbital Configuration

$$\begin{aligned}
 & + \dot{y}_i \left( -c(y_i) \frac{P_i}{|P_i|^2} - c(y_{i+1} - y_i) \frac{P_{i+1}}{|P_{i+1}|^2} \right) \\
 & + \dot{y}_{i+1} \left( c(y_{i+1} - y_i) \frac{P_{i+1}}{|P_{i+1}|^2} \right)
 \end{aligned} \tag{2.28}$$

$$\begin{aligned}
 Fd_N & = \dot{x}_{i-1} \left( c(x_i - x_{i-1}) \frac{P_i}{|P_i|^2} \right) \\
 & + \dot{x}_i \left( -c(x_i - x_{i-1}) \frac{P_i}{|P_i|^2} \right) \\
 & + \dot{y}_{i-1} \left( c(y_i - y_{i-1}) \frac{P_i}{|P_i|^2} \right) \\
 & + \dot{y}_i \left( -c(y_i - y_{i-1}) \frac{P_i}{|P_i|^2} \right)
 \end{aligned} \tag{2.29}$$

## 2.4 Orbital Dynamics

It is necessary to include orbital dynamics such as coriolis force and the gravity gradient for the orbital simulations. The gravity gradient and microgravity appear in the equations of motion when the Gravitational Potential Energy expression (Equation (2.16)) for the testbed configuration is replaced with the expression for orbit (Equation (2.15)).

The coriolis force appears with the introduction of a new definition of kinetic energy for the orbital tether. The definition of velocity is a little more complex for this case, illustrated in Figure 2.2. If  $\bar{r}$  represents the position of the particle in a rotating or orbiting frame of reference with an angular velocity of  $\omega$ , then the velocity of the particle in a non-rotating inertial frame of reference is expressed by  $\bar{v} = \dot{\bar{r}} + \bar{\omega} \times \bar{r}$ . As shown in the figure, the x-axis is tangent to the orbit in the orbital direction, the y-axis is the local vertical (gravity in  $-y$  direction), and the z-axis is out of the paper. The frame is orbiting at an angular rate of  $\omega$  in the negative z direction by the right-hand rule. The origin of the coordinate system is orbiting at an distance of  $a$  meters from the center of the earth. For this coordinate system, then, the kinetic energy is

$$T_{orbit} = \sum_{i=1}^N \frac{m_i}{2} \{[\dot{x}_i + \omega(a + y_i)]^2 + [\dot{y}_i - \omega x_i]^2\}. \quad (2.30)$$

## 2.5 Numerical Solution using MATLAB

The  $N$  equations of motion for the simulation are obtained by substituting Equation (2.8) into Equation (2.10), or into Equation (2.13) if there is damping.  $T$  in Equation (2.8) is defined in Equation (2.14).  $V$  is defined as the sum of the gravitational potential energy,  $V_{gpe}$ , and the elastic energy of the tether material  $V_s$ .  $V_{gpe}$  is defined in Equation

(2.15) for the general orbital case, and Equation (2.16) for the earth-surface testbed case. The elastic energy  $V_s$  is defined in Equations (2.21) and (2.20). We solved Equations (2.10) for the second time derivatives of the  $q$ 's ( $\ddot{x}_i$  and  $\ddot{y}_i$ ) to obtain the equations of motion.

The model was implemented using the Mathworks package MATLAB and the associated suite of ordinary differential equation (ODE) integrator routines. We chose the ODE113 solver, described as follows in the accompanying documentation, found in the hypertext file "help/techdoc/ref/ode113.html" delivered with MATLAB:

ode113 is a variable order Adams-Bashforth-Moulton PECE solver. It may be more efficient than ode45 at stringent tolerances and when the ODE file function is particularly expensive to evaluate. ode113 is a multistep solver – it normally needs the solutions at several preceding time points to compute the current solution.

This solver is able to vary from first order to 12th order. It seems particularly appropriate for two reasons: the differences in scale of the problem result in very stringent tolerances, and the ODE file function is relatively complex and expensive to compute. The problem of scale could probably be solved using a renormalization method, but this proved unnecessary when MATLAB was able to generate a solution within specified tolerances.

The integrator routine must be initialized with the initial state of the system, the time span of interest, and some optional parameters. We calculated the initial conditions for the earth's surface simulations by solving for static elastic strain given the static stress. We used the polynomial solver to allow for both the original Hooke's law linear

stress/strain relationship and for the linear-cubic stress/strain relationship we added later.

The only optional parameters we needed to modify from their default values were the minimum absolute and relative error tolerances. The absolute error tolerance defaults to  $10^{-6}$  for each element of the solution vector, and the relative tolerance defaults to  $10^{-3}$ . These tolerances accumulated error so quickly that MATLAB was not able to generate an acceptable solution. By trial and error we settled on  $10^{-8}$  for both values. This allowed MATLAB to integrate the system, and yet stay well above the round-off error of the machine.

The file "xdot.m," listed in Appendix A, contains the equations of motion for the system, derived as explained above. Slight modifications were made in this file for the orbital simulations of Chapter 5; the modified version is listed in Appendix C. The file "snap.m" takes care of setting up tether parameters for the simulation, executing the simulation, and saving or plotting results. Various other files were used in conjunction with these two to sweep through values of tether parameters. "energy.m" calculates the total energy in the tether, which should not change for the simulations executed without damping. This function was used to test the error tolerances discussed above.

CHAPTER 3  
VALIDATION BY DIGITAL MODEL  
(TSS TETHER/LABORATORY)

In this chapter we validate our tether model by replicating the simulation published in Angrilli [22].

### 3.1 Modifications to Basic Model

The test being replicated uses a model nearly identical to the model described in Chapter 2, except that the test is one-dimensional. This can be easily replicated in the current model by setting the gravity parameter to zero. Without gravity, the tether remains on the x-axis. The simulation consists of stretching the tether with a specified tension, then severing the tether instantaneously at one end.

The tether parameters (specifically the stiffness ( $EA$ ) and the density) in this test are drawn from the TSS type tether, a tether similar in structure to co-axial cable with a copper core and multiple shielding layers.

### 3.2 Validation

#### 3.2.1 Method of Validation

The data presented in the Angrilli paper are in graphical form. Visual comparison of graphical data will be the main method of validation. Some of the tests to be conducted consist of tracking bead motion over time, others involve sampling a single output variable as an input parameter is varied. Thus, in many cases, the general shape

Table 3.1: Simulation Physical Parameters

Parameter	Symbol	Value
Stiffness	$EA$	15,000 N
Density	$\mu$	7.5 kg/km
Damping Ratio	$\xi$	4 %
Initial Tension	T	48 N

of the trajectory recorded will be compared, and this will be supplemented by numerical comparisons of curve end-points and slopes.

### 3.2.2 Results

#### Velocity *versus* Tether Length

This test used the values in Table 3.1 for tether physical parameters, for tethers 14m, 50m, 1000m, and 20,000m long.

Figures 3a-3d in [22] contain the results of these simulations using the Angrilli model. Figure 3.1 displays the results of my simulation of the 1000m tether, which I compare to Figure 3c in [22]. The top line shows the velocity of the bead farthest from the sever point (Bead 1). Angrilli's model shows this bead's velocity settling out with a speed of 2.2 m/s at approximately 0.73s after tether sever. Our model shows this bead settling out at the same time with a speed of 2.0 m/s. The twentieth bead (closest to the sever point) settles out to 4.8 m/s in approximately 0.06 s in our simulation, while Angrilli's simulation shows 5.1 m/s in the same time. The results of all four simulations are summarized in Tables 3.2 and 3.3.

In [22], Angrilli concludes that steady-state velocity is independent of tether length. Our simulation is even more consistent in this respect as shown in Table 3.3. There are

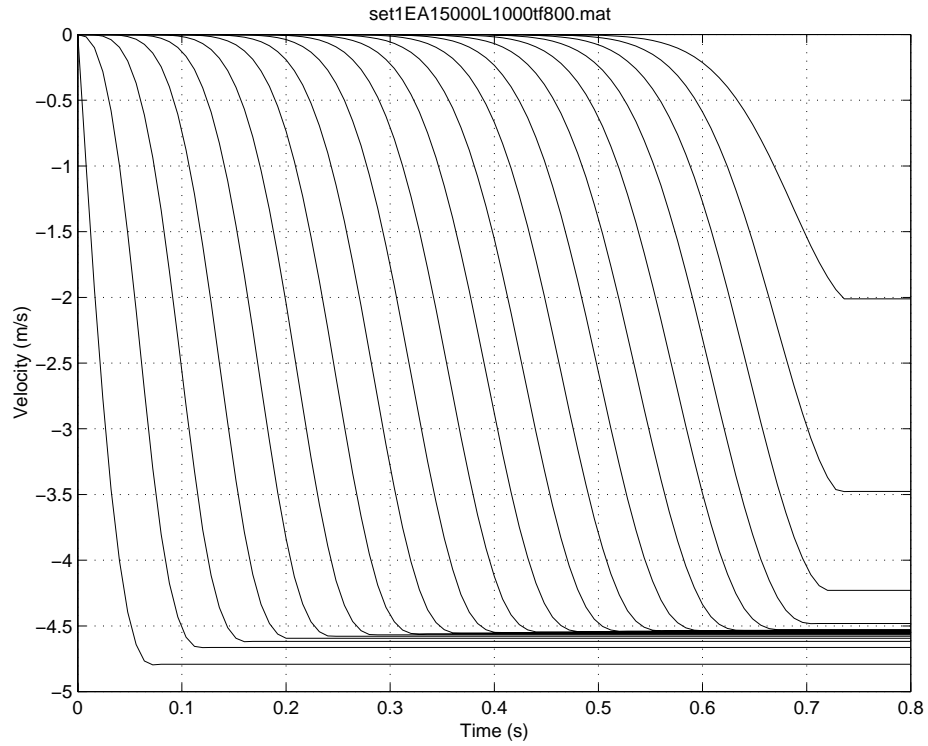


Figure 3.1: 1000m Tether Snapback

Table 3.2: Velocity v. Length Results Comparison: Bead 1

Tether Length (m)	Angrilli		Hammett	
	Velocity (m/s)	Time (s)	Velocity (m/s)	Time (s)
14	2.7	0.011	2.5	0.011
50	2.2	0.037	2.0	0.037
1000	2.2	0.73	2.0	0.73
20000	N/A	N/A	N/A	N/A

Table 3.3: Velocity v. Length Results Comparison: Terminal Bead

Tether Length (m)	Angrilli		Hammett	
	Velocity (m/s)	Time (s)	Velocity (m/s)	Time (s)
14	4.9	0.002	4.8	0.002
50	4.9	0.004	4.8	0.004
1000	5.1	0.06	4.8	0.06
20000	4.8	4.2	4.8	2.8

several possible causes for the small differences between our results and those in [22]. We used MATLAB's ODE Suite to perform the numerical integration, while Angrilli's method is unspecified. The paper also does not specify the method of determining initial conditions. We modeled the tether snapback as strictly one-dimensional (since those were the only data given for comparison in the paper), although the paper itself not clear whether the two-dimensional motion was modeled.

There is no data for the 20km tether length in Table 3.2 because the time axis in [22] was too short to show settling time or velocity for that bead.

### **Tether Stiffness *versus* Velocity**

This experiment shows the effect of changes in tether stiffness on the velocity of the end of the tether after sever. This simulation reproduces the simulation results displayed in [22] Figure 4a. The tether was again under 48 N of tension before the sever. The velocities of the tether end (bead) are summarized in Table 3.4.

Table 3.4: Tether End Velocity v. Stiffness Results Comparison

EA (N)	length (m)	Speed (m/s)	
		Angrilli	Hammett
15000	1.4	2.8	3.48
	2.8	4.1	4.48
	8.0	4.7	4.55
	14.0	4.9	4.79
30000	1.4	1.8	2.46
	2.8	2.9	3.17
	8.0	3.3	3.22
	14.0	3.4	3.39
60000	1.4	1.3	1.74
	2.8	2.0	2.25
	8.0	2.3	2.28
	14.0	2.5	2.40
68000	1.4	1.2	1.64
	2.8	1.9	2.11
	8.0	2.2	2.14
	14.0	2.3	2.26
91000	1.4	1.0	1.42
	2.8	1.6	1.82
	8.0	1.8	1.85
	14.0	2.0	1.95

Table 3.5: Tether Impact Rate Comparison

EA (N)	Bead 1 Time ( $10^3$ s)		Bead 10 Time ( $10^3$ s)	
	Angrilli	Hammett	Angrilli	Hammett
15000	1	0.6	4.1	4.2
60000	2	1.1	8.5	8.4
68000	2	1.1	9	8.8
91000	2	1.4	10	10.1

### Tether Impact *versus* EA

This test simulates a 20000 m tether recoiling towards the orbiter after a sever. This simulation reproduces the simulation results displayed in [22] Figures 7a – 7d. The simulation did not include gravity or orbital dynamics. The data are summarized in Table 3.5, which shows the time after tether sever that the first and the last bead impact the orbiter.

A sample output plot for the  $EA = 15000N$  case is shown in Figure 3.2. Note that after the beads cross the line  $y = 0$  (the "shuttle" location), they reach their natural extension below the "shuttle" and bounce around this position. They are prevented from bouncing back across the  $y = 0$  line by the inertia of the other beads to which they are connected. Angrilli's model seems to handle these dynamics differently, as his beads continue to travel well past the shuttle with no apparent change in velocity in his Figure 7. This could be a clue to some of the differences in the results between Angrilli and our simulation.

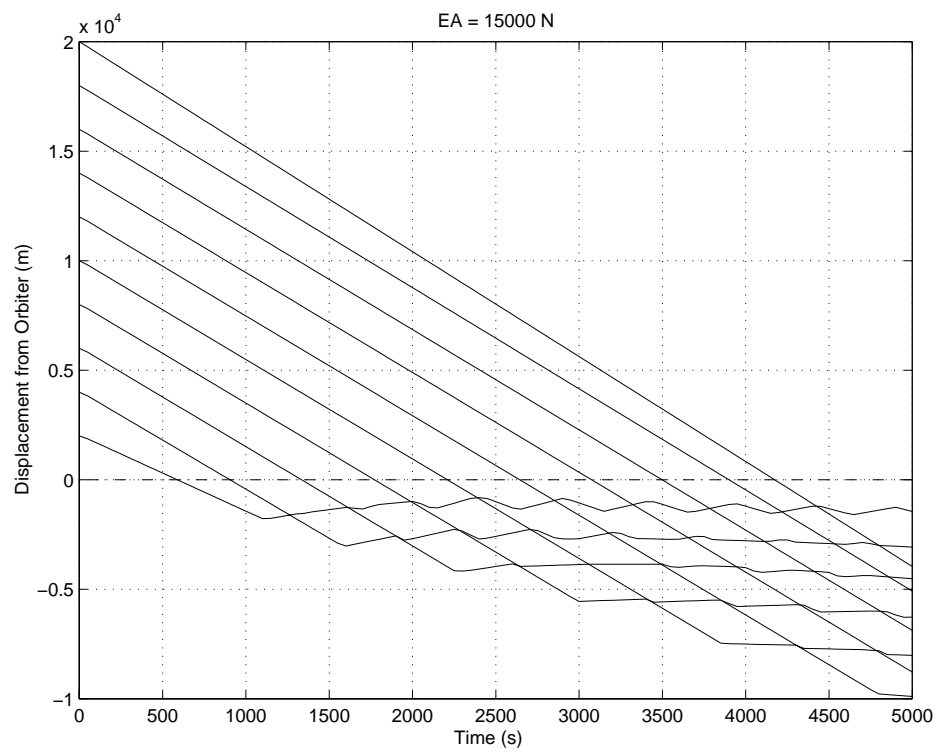


Figure 3.2: Tether Impact Rate

## Conclusions

Our model matches the results of Angrilli's model very well. Angrilli expects and finds that the recoil velocity profile is independent of tether length, and this prediction is supported even more strongly by our data as shown in Figures 3.2 and 3.3.

The second comparison, tether end velocity *versus* stiffness (Figure 3.4) matches the qualitative conclusions of Dr. Angrilli: stiffer tethers recoil more slowly after the same initial tension. Once again, our model shows more clearly the expected independence of velocity profile from tether length.

In the final comparison, tether impact rate *versus* tether stiffness, our results are virtually identical to Angrilli's model. Our results show a stronger relationship between tether stiffness and recoil velocity than Angrilli's, but this is consistent with the results of the previous comparison.

Angrilli, in his Figure 12, shows the results of his discrete model (Figure 3.4) overlaid on the range of values he obtains from analysis using a continuous model. In each case, the values obtained by our model are within the predicted range, and closer to his predicted center value than his own results.

CHAPTER 4  
VALIDATION BY PHYSICAL EXPERIMENT  
(TSS TETHER/SEDS TETHER)

In this chapter we validate our model with experimental data collected in two sets of tether snapback tests conducted at Marshall Space Flight Center in April 1995 [25].

#### 4.1 Modifications to Model

This simulation uses the basic model, discretized with 40 elements, as described in Chapter 2. The simulation includes damping and gravity, but does not need to include orbital dynamics.

The test configuration being simulated is described as follows in the NASA test report [25]:

50 meters (164 ft.) of TSS tether was stretched between two I-beams on mounting points 42" above the floor. One end was fixed, the other was attached to a short length of nylon lacing cord. The lacing cord was attached to a load cell which was mounted on a screw position adjuster on the I-beam.

Three high-speed (100 frames/second) cameras were positioned along the length of the tether. Camera A was at the adjustable end of the tether. Camera B was 20 meters from the fixed end, and Camera C was 10 meters from the fixed end. Background screens with 6 inch graduations were placed behind the tether opposite each camera. Flags were attached to the tether at 10, 20, and 50 meters.

The test procedure consisted of manually applying tension at the adjustable end measured with the load cell. When the required tension was achieved, the cameras were started and the lacing cord was cut.

The test was repeated six times, three at a pre-sever tension of 111N (25 lbs) and three at 55.5N (12.5 lbs). The horizontal position of each flag in each frame was extracted from the images. Using the time stamp of each frame, a horizontal velocity was derived by taking the difference of successive positions. This velocity was plotted versus frame number, and a straight-line curve fit was performed on the velocity after the sever.

The SEDS tether was tested using the same physical setup. The test was performed five times, one test each at .75lbs, 1.5 lbs, 2.0 lbs, 2.5 lbs, and again at 1.5 lbs. The data were reduced in the manner described in the last paragraph for TSS tether test data.

After initial analysis we discovered an obvious anomaly: the three flags' had virtually identical velocities in our simulation, whereas in the NASA experiment the flag near the severed end reached a much higher velocity than the flag at the other end. We substituted an alternate spring model (described in the next section) which increased the difference in these velocities.

## **4.2 Method and Results of Validation**

### **TSS**

We will compare the simulated average velocity for each flag over the first 0.25 s with the measured velocity from the test. Two physical experiments were repeated three times each for the TSS tether. The results of the three repetitions will be averaged to compare to the simulation results. The simulation is deterministic.

The test results are summarized in [25]. We noted that the ends of the tether (Camera A and Camera C) have a large velocity range in the test, but very small range in our simulation with a linear spring model ( $F = kx$ ). We swept over values of  $EA$  from 10,000 N to 100,000 N, achieving the best match for both the 12.5 lb and 115 lb cases at 30,000 N. Those values are shown in the column labeled “linear” in Table 4.1.

We substituted a cubic spring model ( $F = k * (x + \kappa x^3)$ ) and achieved a better match for this velocity spread. After trying various combinations of  $EA$  and  $\kappa$  to become familiar with the problem space, we used MATLAB’s FMINSEARCH function to find an optimal set of parameters. FMINSEARCH can search over an  $N$ -dimensional solution space to minimize a cost function. Due to the time-consuming nature of these simulations, and the complex relationship of the output (tether velocity profile) to the inputs (stiffness parameters), the problem had to be constrained. In particular, we chose starting parameters carefully, basing them on the manual search of the problem space mentioned above.

We used the difference between the steady-state velocity profile in our simulation and the velocity profile measured in the NASA experiments as our cost function. The velocity differences of the outer flags were weighted more heavily to encourage finding a solution which matched the spread in velocities in addition to the average value.

The numbers in Table 4.1 in the column labeled “cubic” were obtained using  $EA = 6040N$  and  $\kappa = 366000$ .

There are several non-linear stress/strain relationships other than the cubic which could possibly achieve even closer matches, such as  $F = k * \tan(x)$  or even changing the

sign on the cubic term above. Damping (in the ranges measured in actual tethers) does not measurably affect tether velocity profiles.

## SEDS

We will compare the simulated average velocity for each flag over the first 0.5 s with the measured velocity from the test. Four different physical experiments were run, with one experiment repeated (1.5 lbs initial tension). Since our simulation is deterministic, the results of the repeated experiment will be averaged for comparison. The raw test results were processed using the same method as for the TSS data in [25]. The comparison is shown in Table 4.2.

One major difference noted with the SEDS tether simulation is that the actual tether velocity was at a maximum soon after the sever, and dropped off rapidly thereafter. In the simulation the velocity after break was nearly constant over the same time period.

Also, as for the TSS tether above, the linear spring model was inadequate to describe the velocities present. The column labeled “linear” in Table 4.2 shows the results using an  $EA$  of 37,000 N, which is substantially higher than the measured  $EA$  for this type tether. The TiPS tether [26] was very similar, and  $EA$  measurements ranged from 100 N to 10,000 N.

The same cubic model is substituted as was used for the TSS tether. The results are in Table 4.2 in the column labeled “cubic,” for  $EA = 14400N$  and  $\kappa = 3.98 \times 10^7$ .

Table 4.1: NASA TSS Test Results Comparison

Camera	Initial Tension (lbs)	Average Flag Velocity (in/sec)		
		NASA	Hammett (linear)	Hammett (cubic)
A	12.5	209	154	192
	25	327	308	310
B	12.5	101	146	156
	25	193	291	241
C	12.5	39	145	86
	25	107	290	139

Table 4.2: NASA SEDS Test Results Comparison

Camera	Initial Tension (lbs)	Average Flag Velocity (in/sec)		
		NASA	Hammett (linear)	Hammett (cubic)
A	0.75	54	43	53
	1.5	82	87	91
	2.0	108	116	112
	2.5	137	145	131
B	0.75	39	40	52
	1.5	55	80	80
	2.0	79	107	93
	2.5	101	134	106
C	0.75	26	40	29
	1.5	37	80	42
	2.0	50	107	50
	2.5	71	134	59

### 4.3 Conclusions

In this chapter we simulated a set of tether snapback experiments performed by NASA. The snapback tests and simulations were performed on both TSS tethers and SEDS tethers. Several tests were performed on each tether type using different initial tensions.

We initially simulated the tests using the same model used in Chapter 3. The results were not satisfactory (even for various values of EA) , so we substituted a linear/cubic model of elasticity as suggested in [18]. A search through values of the two elastic parameters located one set of values for the SEDS tether and another for the TSS tether which yielded acceptable matches for the tests.

There are several possible explanations for the differences in the measured and simulated results. In the measured data the various parts of the tether moved at widely separated velocities. The linear spring gave roughly the same velocity for the entire tether. Substituting the cubic spring gave a larger dispersion in the velocities, but still did not achieve velocity differences as large as those measured. A spring model could probably be found which would match the velocity dispersion better, with a linear component to tune the average velocity.

We also did not account for any compression effects. These could also reduce the velocity near the fixed end of the tether and increase the velocity dispersion. One way this could be modeled would be to greatly decrease the spring constant when compressed, rather than eliminating any force from the spring as we currently do. In combination with this, adding elastic hinges between the beads would add a bending moment which would resist compression on multi-bead scales.

The physical act of severing the tether is modeled as an instantaneous event in our model. It is possible that the sequential severing of lacing cord fibers would change the energy distribution in the tether. The NASA tests were conducted in standard atmosphere, and our model also does not account for aerodynamic effects.

CHAPTER 5  
PREDICTIONS OF SNAPBACK DYNAMICS IN ORBIT  
(SEDS AND TSS TETHER)

In this chapter we use our model to make predictions of tether behavior after a sever in two orbital scenarios. The first scenario corresponds to the TSS-1R reflight, the second is the proposed SEDSAT launch mission. Both of the scenarios are highly simplified: the large endbody is a point mass, and the tether is connected to it by a hinge allowing free angular motion (as the beads are connected to each other). An initial state is calculated by calculating the tension on the deployed tether due to gravity and orbital mechanics, and then uniformly spacing the mass beads between the end-points.

## 5.1 Modifications to Basic Model

This simulation requires the biggest changes to the model, the inclusion of orbital dynamics, discussed in Section 2.4, and the more complex gravitational model, shown in Equation (2.15). We also include the cubic elastic relationship and the specific parameters chosen in the previous chapter.

### 5.1.1 Extrapolation of Nonlinear Parameters

The tether simulations in the previous chapter used a 50-meter tether model discretized into 40 beads. The tether lengths for the orbital simulations are 20,000 meters for the SEDS tether and 22,000 meters for the TSS tether. The linear parameters can be extrapolated with some degree of confidence to different lengths of tether segments,

but the non-linear spring parameter cannot. One solution to this problem would be to use the same tether segment lengths as in the previous chapter, 1.25 meters. This would require around 18,000 beads for each simulation, which proved unworkable when a 1-millisecond simulation did not terminate after one week of execution.

### **TSS Parameter Extrapolation**

For the TSS tether we ran the optimization program from the previous chapter for various numbers of beads. We examined the relationships between the elastic parameters and tether segment length. The linear elastic coefficient  $EA$  was a linear function of the tether segment length, as expected, and could be easily extrapolated. The cubic coefficient  $\kappa$ , however, did not show an obvious trend against the segment length or the cube of the segment length. There was, however, a clear linear relationship when the base-10 logarithm of  $\kappa$  was regressed against the logarithm of the segment length, implying an exponential relationship. The relationship was solid enough to justify extrapolating from the parameters from the longest validated segment length, 25 meters, to 50 meters (440 segments).

### **SEDS Parameter Extrapolation**

We used the same procedure to attempt to extrapolate parameters for the SEDS tether. There was, however, no clear trend in either the  $EA$  or the  $\kappa$  *versus* the segment length. The  $\kappa$  parameter is much larger for the SEDS tether, which probably helped obscure the linear relationship of  $EA$  with segment length, since  $EA$  also multiplies the cubic term in our implementation. The relative effect of the spring is also amplified in this tether, since the individual beads have much less mass but the material is much

Table 5.1: Predictive Simulation Parameters

Parameter	TSS Value	SEDS Value
Density	7.5 kg/km	0.30 kg/km
Length	22000 m	20000 m
Endbody Mass	518 kg	165 kg
Orbital Altitude	230 nm	160 nm
$EA$	889000 N	413281 N
$\kappa$	$8 \times 10^{12}$	$1.48 \times 10^{11}$
Number of Beads	440	2000
Segment Length	50 m	10 m

more stiff than the TSS material. We used the parameters from the longest validated segment length, 10 meters, to run the SEDS simulation. This required 2,000 tether segments for the orbital simulation. The simulation was completed in just under two days of execution time.

## 5.2 TSS Simulation

This simulation reproduces the TSS-1R flight with a tether sever at the endbody after deployment. No electrodynamic effects are considered. Specific parameters used in the simulation are shown in Table 5.1.

The tether accelerates quickly in the first 10 milliseconds until the tether is no longer under tension and each bead is in free motion. The tether segments closest to the shuttle are travelling very slowly, as shown in Figure 5.1. The first kilometer of tether is moving towards the shuttle at velocities ranging from a couple of centimeters a second (50 meters from the shuttle) to 0.5 meters per second (one kilometer from the shuttle). Thus it would take over 2000 seconds before even 1000 meters of the tether

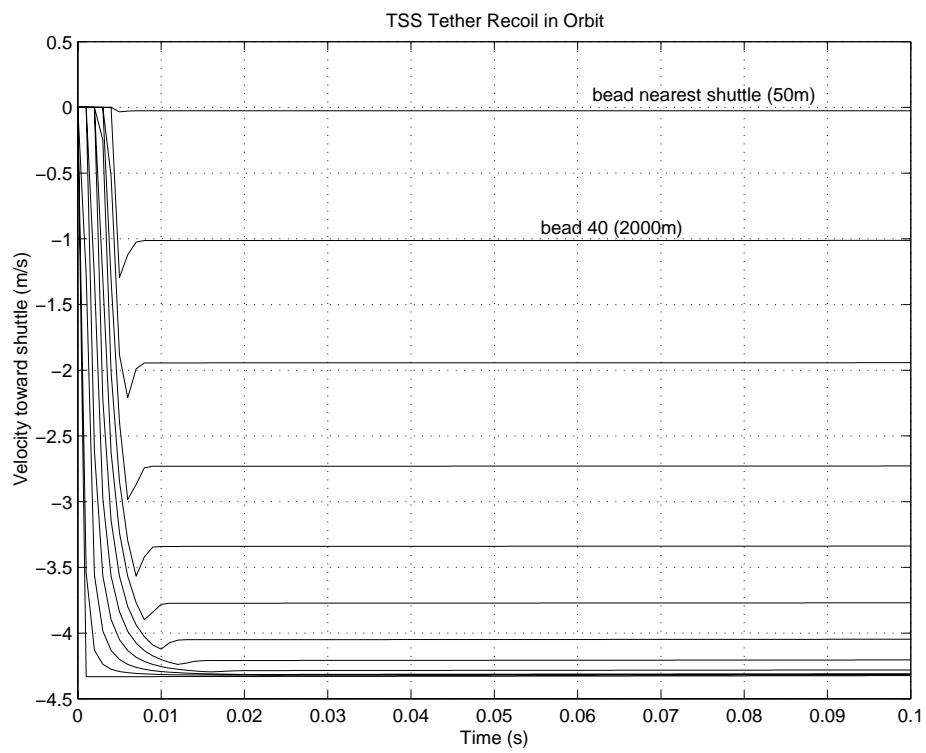


Figure 5.1: TSS Tether Recoil Velocity in Orbit

had recoiled back to the shuttle. The severed end of the tether is moving toward the shuttle at approximately 4.5 meters per second. This implies that it will take at least 80 minutes for the entire length of the tether to reach the plane of the shuttle, by which time coriolis forces will have taken the end well away from the shuttle's original position and the shuttle pilot will have had time to execute any necessary maneuvers.

### 5.3 SEDS Simulation

This simulation reproduces the proposed SEDSAT launch mission, with a tether sever at the endbody after deployment. Specific parameters used in the simulation are given in Table 5.1.

This tether takes 30 to 40 milliseconds to reach free motion velocity. The lower tension of this tether is countered by the low mass, and the net behavior of the tether is quite similar to that of the TSS tether in the previous section. The first kilometer of the tether is moving toward the shuttle at roughly the same velocity as the TSS tether, 0.5 meters per second, and the severed end of the tether is approaching the shuttle at just less than 4 meters per second, as shown in Figure 5.2.

### 5.4 Conclusions

Our analysis is less conservative than that of previous studies, since we assume a much lower velocity of the tether at the orbiter end, as shown by the physical experiments at NASA, and captured in our non-linear cubic spring model. For both the TSS and SEDS tethers, our analysis shows that the near end of the tether travels very slowly toward the orbiter. The first 50 meters of either type of tether would reach the orbiter

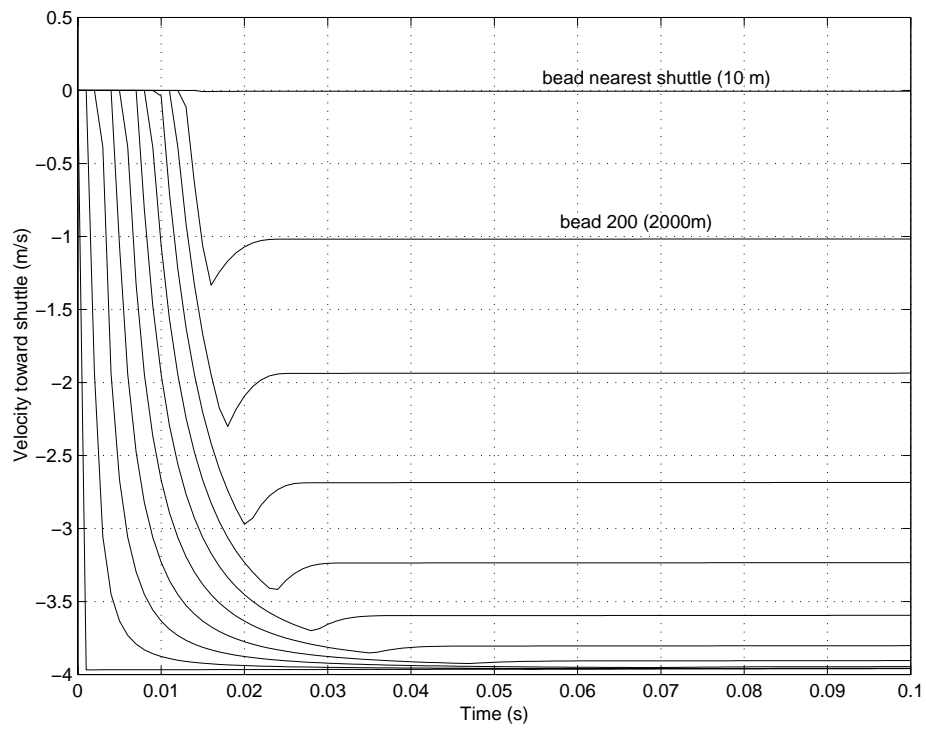


Figure 5.2: SEDS Tether Recoil Velocity in Orbit

in several hundred seconds, giving the pilot at least five minutes to make the decision to end the experiment or deployment by cutting the tether and maneuvering away from the recoiling length. The low velocity of the tether toward the orbiter also allows more time for coriolis forces to act to move the tether away from the orbiter.

## CHAPTER 6

### SUMMARY AND FUTURE WORK

#### 6.1 Summary of Results

The previous models of tether snapback have used simple bead models to present conservative scenarios for worst-case planning. This simulation is a first attempt at modeling more accurately the true behavior of the tether during snapback, in particular the large dispersion of velocities between different segments of the tether which was observed in the NASA experiments.

We developed a discrete element model of a tether, which is appropriate for a slack tether simulation. This model lumped the mass of the tether into beads connected by parallel spring-damper combinations that modeled the elastic behavior of the tether material. The spring and damper were standard linear models when the tether was under tension, but exerted no force when the tether was under compression. This codifies a principle of modeling: “you can’t push a string.”

We validated this model by comparing it to published results of simulations using a similar model in Angrilli [22]. We compared our model against the results of a NASA physical experiment [25]. For this purpose, we used measured physical parameters to generate two versions of our model. One version used the TSS tether physical parameters (density and stiffness), and the other used SEDS tether parameters. After review of this comparison, we substituted a cubic spring model to more closely match the wide velocity dispersion observed in the NASA tests.

These models (one for TSS tether and one for SEDS tether) were used to simulate tether severs in typical orbital mission scenarios: we simulated the TSS-1R experiment using the TSS model, and the proposed deployment of the SEDSAT satellite with the SEDS model. On the basis of these simulations we predicted the reaction time available to the crew between the actual tether sever event and the potential pileup of tether in the orbiter. Accurate prediction of this time is critical, because it gives the crew time to verify that a sever has occurred, rather than jettisoning millions of dollars of equipment at the first, potentially erroneous, indication of sever.

## **6.2 Ideas for Future Work**

Snapback experiments need to be conducted with longer tethers, to determine whether the observed velocity dispersions are an artifact of the test conditions. There are many other models of tether elasticity which could easily be substituted into our tether model to investigate their effects on tether snapback. Among these are variations on the cubic model used in this thesis, a function proportional to the tangent of tether elongation, and even tabular empirical data. Experiments and simulations should be conducted using stronger tethers at higher tensions, to simulate towing the ISS.

## BIBLIOGRAPHY

- [1] M. L. Cosmo and E. C. Lorenzini, *Tethers in Space Handbook*. Marshall Space Flight Center, 3rd edition ed., December 1997.
- [2] E. Scala, “Tethers in space, and micrometeoroids,” in *Third International Conference on Tethers in Space–Toward Flight* [27], pp. 372–378.
- [3] L. Johnson, “The tether solution,” *IEEE Spectrum*, pp. 38–43, July 2000.
- [4] I. E. Vas, T. J. Kelly, and E. Scarl, “Application of an electrodynamic tether system to reboost the international space station,” in Harrison [28], pp. 305–334.
- [5] J. K. Harrison, C. C. Rupp, J. A. Carroll, C. M. Alexander, and E. R. Pulliam, “Small expendable-tether deployment system (SEDS) development status,” in *Third International Conference on Tethers in Space–Toward Flight* [27], pp. 19–26.
- [6] C. Carrington and V. Keller, “Orbiter-towed reboost for ISS,” in Harrison [28], pp. 285–303.
- [7] N. H. Stone, K. H. Wright, J. D. Winningham, K. Papadapolous, T. X. Zhang, K. S. Hwang, S. T. Wu, and U. Samir, “A review of scientific and technological results from the TSS-1R mission,” in Harrison [28], pp. 1–12.
- [8] K. Welzyn and J. Robinson, “Severed tether dynamics and probability,” in *Fourth International Conference on Tethers in Space* [29], pp. 1353–1366.
- [9] K. B. Hayashida, J. H. Robinson, and S. A. Hill, “SEDS tether M/OD damage analysis,” Tech. Rep. NASA/TP-97-206311, Marshall Space Flight Center, November 1997.
- [10] V. V. Beletsky and E. M. Levin, *Dynamics of Space Tether Systems*. American Astronautical Society, 1993.
- [11] M. Pasca, M. Pignataro, and A. Luongo, “Three-dimensional vibrations of tethered satellite systems,” *AIAA Journal of Guidance, Control, and Dynamics*, vol. 14, no. 2, pp. 312–320, 1990.
- [12] B. N. Min and A. K. Misra, “Non-linear free vibration of a spinning tether,” in *Space Flight Mechanics 1997 Part I*, (San Diego, CA), pp. 547–566, AAS Publications, 1997.

- [13] J. Puig-Suari, J. M. Longuski, and S. Tragesser, "A three dimensional hinged-rod model for flexible-elastic aerobraking tethers," in *Astrodynamics 1993 Part I*, pp. 755–774, AAS Publications, 1993.
- [14] E. Netzer and T. R. Kane, "Estimation and control of tethered satellite systems," *AIAA Journal of Guidance, Control, and Dynamics*, vol. 18, pp. 851–858, July-August 1995.
- [15] G. Fanti, "Stress distribution measurements in composite space ropes," in *Fourth International Conference on Tethers in Space* [29], pp. 1181–1194.
- [16] D. D. Tomlin, G. C. Faile, K. B. Hayashida, C. L. Frost, C. Y. Wagner, M. L. Mitchell, J. A. Vaughn, and M. J. Galuska, "Space tethers: Design criteria," NASA Technical Memorandum 108537, Marshall Space Flight Center, July 1997.
- [17] F. M. Kustas, F. J. Jarossy, and L. L. Burgess, "Design considerations for a long-lifetime space tether," in *Fourth International Conference on Tethers in Space* [29], pp. 577–590.
- [18] A. H. von Flotow, "Some approximations for the dynamics of spacecraft tethers," *AIAA Journal of Guidance, Control, and Dynamics*, vol. 11, pp. 357–364, July-August 1998.
- [19] G. Colombo, M. D. Grossi, and D. Arnold, "Study of certain tether safety issues," Contract NAS8-33691 Semiannual Progress Report I, NASA, 1 Sep 1981 – 28 Feb 1982.
- [20] E. Lorenzini, "The investigation of tethered satellite dynamics," Contract NAS8-36160 Quarterly Report 3, NASA, June 1985.
- [21] E. Lorenzini, "The investigation of tethered satellite dynamics," Contract NAS8-36160 Quarterly Report 5, NASA, December 1985.
- [22] F. Angrilli, R. D. Forno, F. Reccanello, and A. Zago, "Effects of severed tethers in space," *Journal of Spacecraft and Rockets*, vol. 34, pp. 239–45, March-April 1997.
- [23] D. T. Greenwood, *Classical Dynamics*. Dover Publications, 1977.
- [24] X. He and J. D. Powell, "Tether damping in space," *AIAA Journal of Guidance, Control, and Dynamics*, vol. 13, no. 1, pp. 104–112, 1989.
- [25] S. Brewster, "Tethered satellite system (TSS) broken tether velocity test," NASA Test Report TCP TSS-DEV-ED95-033, Marshall Space Flight Center, April 1995.
- [26] Naval Center for Space Technology, "Tether physics and survivability experiment." <http://hyperspace.nrl.navy.mil/TiPS/>.

- [27] AIAA/NASA/ASI/ESA, *Third International Conference on Tethers in Space—Toward Flight*, (San Francisco, CA), May 17–19 1989.
- [28] J. K. Harrison, ed., *Tether Technology Interchange Meeting*, September 9–10 1997.
- [29] NASA/SAO/ASI/Lockheed-Martin/AS SPA/SAIC, *Fourth International Conference on Tethers in Space*, (Washington, DC), April 10–14 1995.

## APPENDICES

## APPENDIX A

### MATLAB PROGRAMS FOR CHAPTER 3

The first set of programs is used to replicate the Angrilli paper. `snap.m` sets up the conditions common to all of the simulations. `xdot.m` is the function called by the integrator routine for all of the replications. `runner.m`, `runner2.m`, and `runner3.m` were used to tailor each simulation of this set.

#### A.1 program 1: `snap.m`

```
function [t,xy,E] = snap(EA,tfinal,length,ksiin)

global n g kmi cmi h weights pts kappa

tennewt = 48;
density = 7.5;           %in kg/km, 8.35 nominal
kappa = 0;              %0 means linear springs
g = 0

if (exist('length') == 1) %length of tether in meters
    l = length;
else
    l = 50
end; %if

if (exist('ksiin') == 1) %damping ratio
    ksi = ksiin;
else
    ksi = .04
end; %if

h = l/n;                %length of each bead's segment

ktot = EA/l;
k = ktot*n;             %spring constant for each spring
```

```

mt = density * l/1000;      %total mass of tether
mi = mt/n;                 %mass of each bead

c = 2*ksi * sqrt(k * mi);  %damping constant per damper
                             %(ref Angrilli 1997)

kmi = k/mi;                %normalized spring constant
cmi = c/mi;                %normalized damping constant

t0 = 0;                    %start time

if (exist('tfinal') == 1)
    tf = tfinal;
else
    tf = 10;                %end time
end; %if

pts = 101;                 %number of time steps for simulation output
tstep = (tf - t0)/(pts - 1);
tspan = t0:tstep:tf;

stretch = 1 + tennewt/(ktot*1);

h = h/stretch;

xy0 = [stretch*h*[1:n]' ; zeros(3*n,1)];

xy = zeros(pts,n*4);

options = odeset('RelTol',1e-8,'AbsTol',1e-8);

[t,xy] = ode113('xdot',tspan,xy0,options);

%save lastout t xy h mi g k

E = energy(t,xy,h,mi,g, k);

A.2 program 2: xdot.m

function xydot = xdot(t,xy)

global n g kmi cmi h kappa

```

```

%split incoming state vector up into parts
xydt = xy(((2*n)+1):(4*n));
xdotdot = zeros(n,1);
ydotdot = xdotdot;

%calculate bead distance for first bead in x and y directions
%and from that calculate net distance
dx1 = xy(1);
dx2 = xy(2) - xy(1);
dy1 = xy(n+1);
dy2 = xy(n+2) - xy(n+1);
d1 = sqrt(dx1 ^ 2 + dy1 ^ 2);
d2 = sqrt(dx2 ^ 2 + dy2 ^ 2);

%calculate stretch (distance beyond normal length)
%negative stretch not allowed (string can't push)
stretch1 = d1 - h;
if stretch1 < 0 stretch1 = 0; end;
stretch2 = d2 - h;
if stretch2 < 0 stretch2 = 0; end;

%calculate relative bead velocities.
%and range rate
if (stretch1 > 0)

    vx1 = xydt(1);
    vy1 = xydt(n+1);
    rr1 = ((vx1 * dx1) + (vy1 * dy1))/d1;

else

    rr1 = 0;

end % if stretch1 > 0

if (stretch2 > 0)

    vx2 = xydt(2) - xydt(1);
    vy2 = xydt(n+2) - xydt(n+1);
    rr2 = ((vx2 * dx2) + (vy2 * dy2))/d2;

```

```

else

    rr2 = 0;

end % if stretch2 > 0

%calculate forces on first bead
%this uses cubic stress/strain relationship
%which when kappa = 0 is the same as Hooke's Law
%linear stress/strain
%also includes damping which is linear to range rate
%unless the string is not stretched, in which case it
%is zero.
F1 = kmi * (stretch1 + kappa * stretch1^3) + cmi * rr1;
F2 = kmi * (stretch2 + kappa * stretch2^3) + cmi * rr2;

xdotdot(1) = -F1*dx1/d1 + F2*dx2/d2;
ydotdot(1) = -F1*dy1/d1 + F2*dy2/d2 - g;

for ii = 2:(n-1),

    %reuse distances from previous iteration
    dx1 = dx2;
    dy1 = dy2;
    d1 = d2;
    stretch1 = stretch2;
    F1 = F2;

    %calculate bead distance for each bead in x and y directions
    %and from that calculate net distance
    dx2 = xy(ii+1) - xy(ii);
    dy2 = xy(n+ii+1) - xy(n+ii);
    d2 = sqrt(dx2 ^ 2 + dy2 ^ 2);

    %calculate stretch (distance beyond normal length)
    %negative stretch not allowed (string can't push)
    stretch2 = d2 - h;
    if stretch2 < 0 stretch2 = 0; end;

    %calculate relative bead velocities
    %and range rate
    if (stretch2 > 0)

```

```

    vx2 = xydt(ii+1) - xydt(ii);
    vy2 = xydt(n+ii+1) - xydt(n+ii);
    rr2 = ((vx2 * dx2) + (vy2 * dy2))/d2;

else

    rr2 = 0;

end % if stretch2 > 0

%calculate forces on first bead
%this uses cubic stress/strain relationship
F2 = kmi * (stretch2 + kappa * stretch2^3) + cmi * rr2;

xdotdot(ii) = -F1*dx1/d1 + F2*dx2/d2;
ydotdot(ii) = -F1*dy1/d1 + F2*dy2/d2 - g;

end %for loop

xdotdot(n) = -F2*dx2/d2;
ydotdot(n) = -F2*dy2/d2 - g;

xydot = [xydt;xdotdot;ydotdot];

```

### A.3 program 3: runner.m

```

global n
%This program controls the simulation which
%replicates Figure 3a-d of the Angrilli paper
%set 1
EA = [15000];
lns = [14 50 1000 20000];
tfs = [.015 .04 .8 6];
n = 10;

aa = length(EA);
bb = length(lns);

for ii = 1:aa,
    for jj = 1:bb,

```

```

    [t,xy,E] = snap(EA(ii),tfs(ii,jj),lns(jj),.04);
    savename = ['set1EA' num2str(EA(ii)) 'L' num2str(lns(jj)) ...
               'tf' num2str(1000*tfs(ii,jj))]
    eval(['save ' savename '.mat t xy E n;']);
    clear t xy E savename
    n=20
end
end
clear aa bb EA lns tfs n ii jj

```

#### A.4 program 4: runner2.m

```

global n

%set 2
EA = [15000 30000 60000 68000 91000];
lns = [1.4 2.8 8 14];
tfs = 0.2 * ones(5,4);
n = 20;

aa = length(EA);
bb = length(lns);
speeds = zeros(aa,4);

for ii = 1:aa,
    for jj = 1:bb,
        [t,xy,E] = snap(EA(ii),tfs(ii,jj),lns(jj));
        savename = ['set2EA' num2str(EA(ii)) 'L' num2str(10 * lns(jj)) ...
                   'tf' num2str(1000*tfs(ii,jj))]
        eval(['save ' savename '.mat t xy E n;']);
        speeds(ii,1) = xy(101,2*n+2);
        speeds(ii,2) = xy(101,2*n+4);
        speeds(ii,3) = .5714 * xy(101,2*n+11) + .4286 * xy(101,2*n+12);
        speeds(ii,4) = xy(101,3*n);
        clear t xy E savename
    end
end
clear aa bb n ii jj

plot(EA,-speeds,'*')

```

**A.5 program 5: runner3.m**

```
global n
%set 3
EA = [15000 60000 68000 91000];
lns = [20000];
tfs = [5000 ;10000 ;10000 ;10000];
n = 20;

aa = length(EA);
bb = length(lns);

for ii = 1:aa,
    for jj = 1:bb,
        [t,xy,E] = snap(EA(ii),tfs(ii,jj),lns(jj));
        savename = ['set3EA' num2str(EA(ii)) 'L' num2str(lns(jj)) ...
                    'tf' num2str(tfs(ii,jj))]
        eval(['save ' savename '.mat t xy E n;']);
        clear t xy E savename
    end
end
clear aa bb n ii jj
```

## APPENDIX B

### MATLAB PROGRAMS FOR CHAPTER 4

The next set of programs was used to replicate the NASA tether sever physical experiments.

#### B.1 Program 1: snap.m

```
function [t,xy,E] = snap(EA,tennewt,tfinal,leng,ksiin,density,kappal)

global n g kmi cmi h weights pts kappa

kappa = kappal           %0 means linear springs
g = 9.81;                %gravity

if (exist('leng') == 1)  %length of tether in meters
    l = leng;
else
    l = 50
end; %if

if (exist('ksiin') == 1) %damping ratio
    ksi = ksiin;
else
    ksi = .04
end; %if

h = l/n;                %length of each bead's segment

ktot = EA/l;
k = ktot*n;             %spring constant for each spring
mt = density * l/1000;  %total mass of tether
mi = mt/n;              %mass of each bead

c = 2*ksi * sqrt(k * mi); %damping constant for each damper
                           %(ref Angrilli 1997)
```

```

kmi = k/mi;           %normalized spring constant
cmi = c/mi;          %normalized damping constant

t0 = 0;               %start time

if (exist('tfinal') == 1)
    tf = tfinal;
else
    tf = 10;          %end time
end; %if

pts = 101;            %number of time steps for simulation output
tstep = (tf - t0)/(pts - 1);
tspan = t0:tstep:tf;

lin1_stretch = 1 + tennewt/(ktot*1)
lin2_stretch = 1 + roots([ktot*1 -tennewt])
kappa
lstretch = roots([(ktot*1*kappa) 0 (ktot*1) (-tennewt)])
stretch = 1 + lstretch(length(lstretch))
h = h/stretch;

xy0 = [stretch*h*[1:n]' ; zeros(3*n,1)];

xy = zeros(pts,n*4);

options = odeset('RelTol',1e-8,'AbsTol',1e-8);

[t,xy] = ode113('xdot',tspan,xy0,options);

E = energy(t,xy,h,mi,g, k,kappa);

```

## B.2 Program 2: xdot.m

See Appendix A Program 2.

## B.3 Program 3: tss1.m

```

global n

%TSS runs

```

```

tens = [111 55.5]; %tensions in newtons
lns = [50];
EAS = [10000]
tss_dense = 7.5
tfs = [0.25];
theksi= [0];
kappas= [1e5];
n = 40;

aa = length(tens);
bb = length(EAS);
cc = length(theksi);
dd = length(kappas);

for ii = 1:aa,
    for jj = 1:bb,
        for kk = 1:cc,
            for ll = 1:dd,
                [t,xy,E] = snap(EAS(jj),tens(ii),tfs(1),lns(1),...
                    theksi(kk),tss_dense,kappas(ll));
                savename = ['TSSTen' num2str(10*tens(ii)) 'EA' ...
                    num2str(EAS(jj)) 'ksi' num2str(100*theksi(kk)) ...
                    'k' num2str(100*kappas(ll))]
                eval(['save ' savename '.mat t xy E n;']);
                plotme
                clear t xy E savename
            end
        end
    end
end
clear aa bb EAS lns tfs n ii jj

```

#### B.4 Program 4: sedsl.m

```

global n

%SEDS runs
tens = [.75 1.5 2.0 2.5] * 4.448; %tensions in newtons
lns = [50];
EAS = [15000]
seds_dense = 13 * 0.454 / 20
tfs = [0.25];

```

```

theksi= [0];
kappas= [1e8];
n = 40;

aa = length(tens);
bb = length(EAS);
cc = length(theksi);
dd = length(kappas);

for ii = 1:aa,
    for jj = 1:bb,
        for kk = 1:cc,
            for ll = 1:dd,
                [t,xy,E] = snap(EAS(jj),tens(ii),tfs(1),lns(1),...
                theksi(kk),seds_dense,kappas(ll));
                savename = ['TSSTen' num2str(10*tens(ii)) 'EA' ...
                num2str(EAS(jj)) 'ksi' num2str(100*theksi(kk)) ...
                'k' num2str(100*kappas(ll))]
                plotme
                clear t xy E savename
            end
        end
    end
end
clear aa bb EAS lns tfs n ii jj

```

### B.5 Program 5: energy.m

```

function E = energy(t,xy,h,mi,g, k,kappa)
[r s] = size(xy);
if (r ~= length(t))
    error('Time vector does not correspond to state matrix');
end

n = s/4;

xpos = xy(:,1:n);
ypos = xy(:,(n+1):(2*n));
xvel = xy(:,(2*n+1):(3*n));
yvel = xy(:,(3*n+1):(4*n));
totvel_sqrd = (xvel.^2 + yvel.^2);
xdist = [xpos(:,1) diff(xpos')]';

```

```
ydist = [ypos(:,1) diff(ypos')'];
stretch_tmp = sqrt(xdist.^2 + ydist.^2) - h;
stretch = (stretch_tmp > 0) .* stretch_tmp;
stretch_sqrd = stretch.^2;

%GPE

GPE = mi * g * ypos;

%Spring energy (sounds like ginseng and rice on the buffet)

SE = 0.5 * k * stretch_sqrd + 0.25 * k * kappa * stretch_sqrd.^2;

%Kinetic energy

KE = 0.5 * mi * totvel_sqrd;

%addemup

E = sum((GPE + SE + KE)');
```

## APPENDIX C

### MATLAB PROGRAMS FOR CHAPTER 5

This last set of programs was used to make predictions of tether sever dynamics in orbit.

#### C.1 Program 1: snap.m

```
function [t,xy,E] = snap(EA,orbital_alt,tfinal,leng, ...
    mass_b,ksiin,density,kappa1)

global n kmi cmi h weights pts kappa GM a w w2

kappa = kappa1                %0 means linear springs
GM = 6.67259e-11 * 5.9736e24; %Gravitational parameter of earth
earth_rad = 6.37814e6;        %meters
orbital_alt = 100;           %km above earth's surface
a = orbital_alt + earth_rad

if (exist('leng') == 1)      %length of tether in meters
    l = leng;
else
    l = 50
end; %if

if (exist('ksiin') == 1)    %damping ratio
    ksi = ksiin;
else
    ksi = .04
end; %if

h = l/n;                    %length of each bead's segment

b = a + leng;
w = sqrt(GM/(a))/a;
w2 = w * w;
```

```

ktot = EA/l;
k = ktot*n;           %spring constant for each spring
mt = density * l/1000; %total mass of tether
mi = mt/n;           %mass of each bead

c = 2*ksi * sqrt(k * mi); %damping constant for each damper
    % (ref Angrilli 1997)

kmi = k/mi;          %normalized spring constant
cmi = c/mi;          %normalized damping constant

t0 = 0;              %start time

if (exist('tfinal') == 1)
    tf = tfinal;
else
    tf = 10;          %end time
end; %if

pts = 101;           %number of time steps for simulation output
tstep = (tf - t0)/(pts - 1);
tspan = t0:tstep:tf;

tennewt = mass_b * (w2 * b * b * b - GM)/(b*b)
lin1_stretch = 1 + tennewt/(ktot*l)
lin2_stretch = 1 + roots([ktot*l -tennewt])
kappa
lstretch = roots([(ktot*l*kappa) 0 (ktot*l) (-tennewt)])
stretch = 1 + lstretch(length(lstretch))
%error('here we are')

h = h/stretch;

xy0 = [zeros(n,1) ; stretch*h*[1:n]' ; zeros(2*n,1)];

xy = zeros(pts,n*4);

options = odeset('RelTol',1e-8,'AbsTol',1e-8);

[t,xy] = ode113('xdot',tspan,xy0,options);

```

```
E = 0;
```

## C.2 Program 2: xdot.m

```
function xydot = xdot(t,xy)

global n g kmi cmi h kappa GM a w w2

%split incoming state vector up into parts
xydt = xy(((2*n)+1):(4*n));
xdotdot = zeros(n,1);
ydotdot = xdotdot;

%calculate bead distance for first bead in x and y directions
%and from that calculate net distance
dx1 = xy(1);
dx2 = xy(2) - xy(1);
dy1 = xy(n+1);
dy2 = xy(n+2) - xy(n+1);
d1 = sqrt(dx1 ^ 2 + dy1 ^ 2);
d2 = sqrt(dx2 ^ 2 + dy2 ^ 2);

%calculate stretch (distance beyond normal length)
%negative stretch not allowed (string can't push)
stretch1 = d1 - h;
if stretch1 < 0 stretch1 = 0; end;
stretch2 = d2 - h;
if stretch2 < 0 stretch2 = 0; end;

%calculate relative bead velocities.
%and range rate
if (stretch1 > 0)

    vx1 = xydt(1);
    vy1 = xydt(n+1);
    rr1 = ((vx1 * dx1) + (vy1 * dy1))/d1;

else

    rr1 = 0;
```

```

end % if stretch1 > 0

if (stretch2 > 0)

    vx2 = xydt(2) - xydt(1);
    vy2 = xydt(n+2) - xydt(n+1);
    rr2 = ((vx2 * dx2) + (vy2 * dy2))/d2;

else

    rr2 = 0;

end % if stretch2 > 0

%calculate forces on first bead
%this uses cubic stress/strain relationship
%which when kappa = 0 is the same as Hooke's Law
%linear stress/strain
%also includes damping which is linear to range rate
%unless the string is not stretched, in which case it
%is zero.
F1 = kmi * (stretch1 + kappa * stretch1^3) + cmi * rr1;
F2 = kmi * (stretch2 + kappa * stretch2^3) + cmi * rr2;

xdotdot(1) = -F1*dx1/d1 + F2*dx2/d2 + w2 * xy(1) - 2*w*xy(3*n+1);
ydotdot(1) = -F1*dy1/d1 + F2*dy2/d2 - GM/((a+xy(n+1))^2) ...
            + w2*(a+xy(n+1)) + 2*w*xy(2*n+1);

for ii = 2:(n-1),

    %reuse distances from previous iteration
    dx1 = dx2;
    dy1 = dy2;
    d1 = d2;
    stretch1 = stretch2;
    F1 = F2;

    %calculate bead distance for each bead in x and y directions
    %and from that calculate net distance
    dx2 = xy(ii+1) - xy(ii);
    dy2 = xy(n+ii+1) - xy(n+ii);
    d2 = sqrt(dx2 ^ 2 + dy2 ^ 2);

```

```

%calculate stretch (distance beyond normal length)
%negative stretch not allowed (string can't push)
stretch2 = d2 - h;
if stretch2 < 0 stretch2 = 0; end;

%calculate relative bead velocities
%and range rate
if (stretch2 > 0)

    vx2 = xydt(ii+1) - xydt(ii);
    vy2 = xydt(n+ii+1) - xydt(n+ii);
    rr2 = ((vx2 * dx2) + (vy2 * dy2))/d2;

else

    rr2 = 0;

end % if stretch2 > 0

%calculate forces on first bead
%this uses cubic stress/strain relationship
F2 = kmi * (stretch2 + kappa * stretch2^3) + cmi * rr2;

xdotdot(ii) = -F1*dx1/d1 + F2*dx2/d2 + w2 * xy(ii) - 2*w*xy(3*n+ii);
ydotdot(ii) = -F1*dy1/d1 + F2*dy2/d2 - GM/((a+xy(n+ii))^2) ...
    + w2*(a+xy(n+ii)) + 2*w*xy(2*n+ii);

end %for loop

xdotdot(n) = -F2*dx2/d2 + w2 * xy(n) - 2*w*xy(4*n);
ydotdot(n) = -F2*dy2/d2 - GM/((a+xy(2*n))^2) + ...
    w2*(a+xy(2*n)) + 2*w*xy(3*n);

xydot = [xydt;xdotdot;ydotdot];

```

### C.3 Program 3: tss1.m

```

function errs = tss1(timerin)

global n

```

```

%TSS runs
lms = [22000];
tss_mass = 518 %kg
tss_alt = 230 * 1852; %SEDSAT shuttle orbit 160 nm
EAS = [889000]
tss_dense = 7.5
tfs = timerin;
theksi= [0.04];
kappas= [8e+12];
n = lms/50

bb = length(EAS);
cc = length(theksi);
dd = length(kappas);

    for jj = 1:bb,
        for kk = 1:cc,
            for ll = 1:dd,
                [t,xy,E] = snap(EAS(jj),tss_alt,tfs(1),lms(1),...
                    tss_mass,theksi(kk),tss_dense,kappas(ll));
                save workspace1
            end
        end
    end
errs = 0;

```

#### C.4 Program 4: seds1.m

```

global n

%SEDS runs
lms = [20000];
seds_mass = 75 * 2.2; %SEDSAT mass is 75 lbs
                                %(well, okay, weight at sea level)
seds_alt = 160 * 1852; %SEDSAT shuttle orbit 160 nm
seds_dense = 13 * 0.454 / 20
tfs = [0.1];
theksi= [0.04];
kappas= [1.48e11];
n = 2000;

```

```
bb = length(EAS);
cc = length(theksi);
dd = length(kappas);
for jj = 1:bb,
    for kk = 1:cc,
        for ll = 1:dd,
            [t,xy,E] = snap(EAS(jj),seds_alt,tfs(1),lns(1),...
                            seds_mass,theksi(kk),seds_dense,kappas(ll));
            savename = ['SEDS' 'EA' num2str(EAS(jj)) 'ksi' ...
                        num2str(100*theksi(kk)) 'k' num2str(100*kappas(ll))]
            save workspace1
        end
    end
end
clear aa bb EAS lns tfs n ii jj
```