1989

NASA/ASEE SUMMER FACULTY FELLOWSHIP PROGRAM

MARSHALL SPACE FLIGHT CENTER
THE UNIVERSITY OF ALABAMA IN HUNTSVILLE

A VECTORIZED ALGORITHM FOR 3D DYNAMICS OF A TETHERED SATELLITE

Prepared by:	Howard B. Wilson
Academic Rank:	Professor
University and Department:	The University of Alabama Department of Engineering Mechanics Tuscaloosa, Alabama
NASA/MSFC Laboratory:	Structures and Dynamics
Division:	Structural Analysis
MSFC Colleague:	Mario Rheinfurth
Date:	August 18, 1989
Contract No.:	The University of Alabama in Huntsville NGT 01-008-021

A VECTORIZED ALGORITHM FOR 3D DYNAMICS OF A TETHERED SATELLITE

by
Howard Wilson
Professor of Engineering Mechanics
The University of Alabama
Tuscaloosa, Alabama

ABSTRACT

Equations of motion characterizing the three dimensional motion of a tethered satellite during the retrieval phase are studied. The mathematical model involves an arbitrary number of point masses connected by weightless cords. Motion occurs in a gravity gradient field. The formulation presented accounts for general functions describing support point motion, rate of tether retrieval, and arbitrary forces applied to the point masses. The matrix oriented program language MATLAB is used to produce an efficient vectorized formulation for a) computing natural frequencies and mode shapes for small oscillations about the static equilibrium configuration; b) for integrating the nonlinear differential equations governing large amplitude motions. An example of time response pertaining to the skip rope effect is investigated.

ACKNOWLEDGEMENTS

I wish to express sincere appreciation to members of the Structures and Dynamics Laboratory Center for allowing me to have the opportunity during the current summer of working in the exciting and intellectually stimulating environment which prevails at Marshall Space Flight Center. I am particularly grateful to Bob Ryan, Mario Rheinfurth and John Admire. The technical guidance, interesting discussions and friendships I have enjoyed with these gentlemen is especially rewarding. Much credit is also deserved by Frank Six, Gerald Karr, and Mike Freeman for so successfully informing participants in the summer fellowship program about the pioneering leadership which NASA continues to foster and about the opportunities existing for involvement in their interesting research activities.

INTRODUCTION AND OBJECTIVES

Tethered satellites employed in space vehicles have many important practical applications. The time response occurring during the retrieval phase is of special interest since this motion can sometimes become unstable as the tether length is reduced to zero. One type of motion currently being studied is known as the skiprope mode. In this configuration a satellite is initially positioned on the line from the orbiter to the earth center (z-axis) and the tether is spinning about this axis in a sinusoidal curve resembling the deflection pattern of a skyrope. Some disagreement currently exists among analysts regarding the dynamics which occurs during retrieval. The author's efforts were directed toward developing a concise comрuter formulation to facilitate a response analysis in this problem.

NONLINEAR EQUATIONS FOR LARGE AMPLITUDE MOTIONS

Consider a tether model described as a series of point masses connected by weightless cords assumed to remain in tension throughout the motion. The geometry is shown in Figure 1. The vector from mass m_{j+1} to m_j is denoted by r_j. Thus

$$r_j = l_j \hat{e}_j \left(\theta_j, \varphi_j\right) \tag{1}$$

where l_j denotes the vector length and (θ_j, φ_j) are spherical coordinate angles for unit vectors $\hat{e}_j$ which specify the member directions. We will assume that l_j are known functions of time. This allows for treatment of the tether retrieval problem during an interval over which l_n is reduced to zero at a specified rate. Once a particular link has been completely retrieved, the problem dimensionality car be reduced by one and the retrieval process can be continued.

The equations of motion can be integrated numerically when $\ddot{\theta}$ and $\ddot{\varphi}$ values are known for each link. The numerical procedure e below is as follows.

1. For known $\theta, \varphi, \dot{\theta}$ and $\dot{\varphi}$ values, link tensions are computed and these are used to compute global accelerations
2. The global accelerations are used to compute $\ddot{r}$ for each link.
3. The $\ddot{r}$ values are used to compute $\ddot{\theta}$ and $\ddot{\varphi}$ values.

For convenient reference let us call the tether section between m_{i+1} and m_i as link i. The tension force in link i is representable as

$$T_i = \alpha_i \, r_i \tag{2}$$

where α_i is a scalar multiplier. A free body diagram of m_i is shown in Figure 2. The force F_i describes the effects of external loads such as gravity, aerodynamic drag, and electromagnetic phenomena. The position of mass i in a rotating reference frame tangent to the circular orbit is

$$R_i = \sum_{j=i}^{n} r_j + R_{n+1} \; , \quad i \leq n \tag{3}$$

The length of link i can be computed from

$$l_i^2 = r_i \cdot r_i = (R_{i+1} - R_i) \cdot (R_{i+1} - R_i) \tag{4}$$

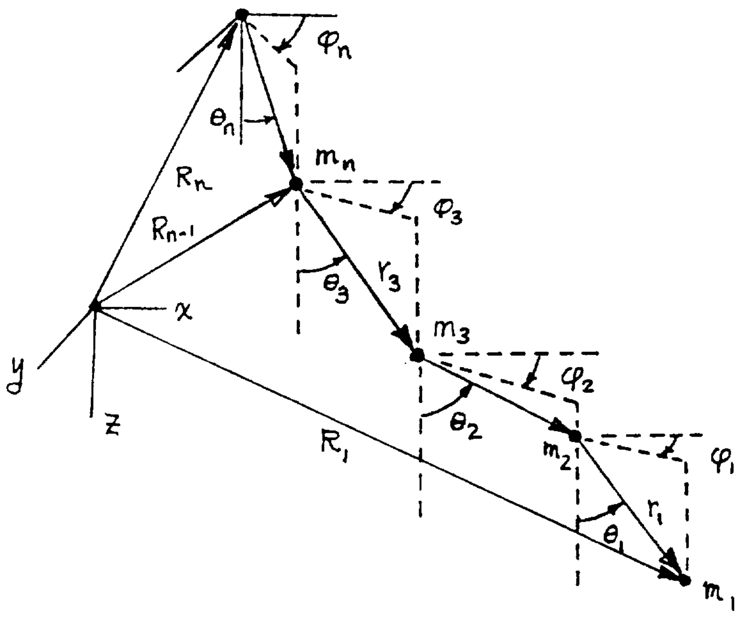

Figure 1: General Tether Configuration

Differentiation of this constraint gives

$$l_i\dot{l}_i = (R_{i+1} - R_i)\cdot(\dot{R}_{i+1} - \dot{R}_i) \tag{5}$$

and

$$\dot{l}_i^2 + l_i\ddot{l}_i = (R_{i+1} - R_i)\cdot(\ddot{R}_{i+1} - \ddot{R}_i) + (\dot{R}_{i+1} - \dot{R}_i)\cdot(\dot{R}_{i+1} - \dot{R}_i) \tag{6}$$

which is equivalent to

$$\dot{l}_i^2 + l_i\ddot{l}_i = -r_i\cdot(\ddot{R}_{i+1} - \ddot{R}_i) + \dot{r}_i\cdot\dot{r}_i \tag{7}$$

When a link has constant length then the constraint simplifies to

$$r_i\cdot(\ddot{R}_{i+1} - \ddot{R}_i) = \dot{r}_i\cdot\dot{r}_i \tag{8}$$

We write

$$r_i = l_i\ \hat{e}_i(\theta_i, \varphi_i) \tag{9}$$

so

$$\dot{r}_i = \dot{l}_i\ \hat{e}_i + l_i\left(\frac{\partial\hat{e}_i}{\partial\theta_i}\ \dot{\theta}_i + \frac{\partial\hat{e}_i}{\partial\varphi_i}\ \dot{\varphi}_i\right) = \dot{l}_i\ \hat{e}_i + l_i(\hat{e}_{i\theta}\dot{\theta}_i + \hat{e}_{i\varphi}\ \dot{\varphi}_i) \tag{10}$$

For the coordinate system of interest $\hat{e}_i, \hat{e}_{i\varphi}$ and $\hat{e}_{i\theta}$ form an orthogonal (but not orthonormal triad). Then

$$\dot{r}_i \cdot \dot{r}_i = |\dot{l}_i \, \hat{e}_i + l_i(\hat{e}_{i\theta} \, \dot{\theta}_i + \hat{e}_{i\varphi} \, \dot{\varphi}_i)|^2 \tag{11}$$

$$= \dot{l}_i^2 + l_i^2(|\hat{e}_{i\theta}|^2 \, \dot{\theta}_i^2 + |\hat{e}_{i\varphi}|^2 \, \dot{\varphi}_i^2) \tag{12}$$

So

$$l_i \ddot{l}_i = l_i^2 \left(|\hat{e}_{i\theta}|^2 \, \dot{\theta}_i^2 + |\hat{e}_{i\varphi}|^2 \, \dot{\varphi}_i^2 \right) - r_i \cdot \left(\ddot{R}_{i+1} - \ddot{R}_i \right) \tag{13}$$

$$- r_i \cdot (\ddot{R}_{i+1} - \ddot{R}_i) = l_i \ddot{l}_i - l_i^2(| \, \hat{e}_{i\theta}|^2 \, \dot{\theta}_i^2 + |\hat{e}_{i\varphi}|^2 \, \dot{\varphi}_i^2) = p_i \tag{14}$$

The Newtonian equation of motion for a typical mass implies

$$\ddot{R}_i = \frac{1}{m_i}[F_i + \alpha_{i-1} \, r_{i-1} - \alpha_i r_i] \tag{15}$$

so the constraint equation leads to the following relation among link tensions

$$r_i \cdot (\frac{1}{m_{i+1}} [F_{i+1} + \alpha_i \, r_i - \alpha_{i+1} r_{i+1}] - \frac{1}{m_i} [F_i + \alpha_{i-1} \, r_{i-1} - \alpha_i r_i]) = -p_i \tag{16}$$

$$\begin{aligned} -\frac{1}{m_i}(r_{i-1} \cdot r_i)\alpha_{i-1} + (\frac{1}{m_i} + \frac{1}{m_{i+1}})l_i^2 \alpha_i - \frac{1}{m_{i+1}}(r_i \cdot r_{i+1})\alpha_{i+1} = p_i \\ + r_i \cdot [F_i/m_i - F_{i+1}/m_{i+1}] \end{aligned} \tag{17}$$

Solution of this tridiagonal system determines the member tensions which ca.ı be used to compute global accelerations $\ddot{R}_i$. These global accelerations allow computation of

$$\ddot{r}_i = -(\ddot{R}_{i+1} - \ddot{R}_i) \tag{18}$$

Since $r_i = l_i \hat{e}_i$, we have

$$\dot{r}_i = \dot{l}_i \, \hat{e}_i + l_i(\hat{e}_{i\theta} \, \dot{\theta}_i + \hat{e}_{i\varphi} \, \dot{\varphi}_i) \tag{19}$$

$$\begin{aligned} \ddot{r}_i = \ddot{l}_i \, \hat{e}_i + 2\dot{l}_i(\hat{e}_{i\theta} \, \dot{\theta}_i + \hat{e}_{i\varphi} \, \dot{\varphi}_i) \\ + l_i(\hat{e}_{i\theta} \, \ddot{\theta}_i + \hat{e}_{i\varphi} \, \ddot{\varphi}_i + \hat{e}_{i\theta\theta} \, \dot{\theta}_i^2 + \hat{e}_{i\varphi\varphi} \, \dot{\varphi}_i^2 + 2\hat{e}_{i\theta\varphi} \, \dot{\varphi}_i \, \dot{\theta}_i) \end{aligned} \tag{20}$$

The last equation can be rearranged as

$$\ddot{r}_i = \ddot{l}_i \, \hat{e}_i + l_i \, \hat{e}_{i\theta} \, \ddot{\theta}_i + l_i \, \hat{e}_{i\varphi} \, \ddot{\varphi}_i + q_i \tag{21}$$

Since the coordinate system of interest satisfies the orthogonality condition

$$\hat{e}_i \cdot \hat{e}_{i\theta} = \hat{e}_i \cdot \hat{e}_{i\varphi} = \hat{e}_{i\theta} \cdot \hat{e}_{i\varphi} = 0 \tag{22}$$

then

$$(\ddot{r}_i - q_i) \cdot \hat{e}_{i\theta} = \ddot{\theta}_i \; l_i |\hat{e}_{i\theta}|^2 \tag{23}$$

implies

$$\ddot{\theta}_i = (\ddot{r}_i - q_i) \cdot \hat{e}_{i\theta} / (l_i |\hat{e}_{i\theta}|^2). \tag{24}$$

Similarly

$$(\ddot{r}_i - q_i) \cdot \hat{e}_{i\varphi} = \ddot{\varphi}_i \; l_i |\hat{e}_{i\theta}|^2 \tag{25}$$

implies

$$\ddot{\varphi}_i = (\ddot{r}_i - q_i) \cdot \hat{e}_{i\varphi} / (l_i |\hat{e}_{i\varphi}|^2). \tag{26}$$

These relations form the basis for assembling the equations of motion. The coordinate reference frame used in the present study assumes a coordinate transformation of the form

$$r = (x, y, z) = l \; (\sin(\theta) \; \cos(\varphi), \; \sin(\theta) \; \sin(\varphi), \; \cos(\theta))$$

which leads to $|\hat{e}_{i\theta}| = 1$ and $|\hat{e}_{i\varphi}| = \sin^2(\theta)$. A set of $(l_i, \theta_i, \varphi_i)$ values defines the configuration of link i. The values of l_i are constant except for $i = n$, in which case, the time dependence of ln, $\dot{l}n$ and $\ddot{l}n$ are specified by the chosen retrieval rate equation.

FORCE EFFECTS IN THE GRAVITY GRADIENT FIELD

We are interested in the dynamics of a tether attached to an orbiter moving about the earth with constant angular speed ω_0 in a circular orbit of radius ρ_0. A balance of gravitational and centrifugal force requires an orbital speed $\omega_0 = \sqrt{g\rho_E^2/\rho_0^3}$ where ρ_E denotes the radius of the earth. The motion of objects moving relative to the satellite are appropriately referred to a frame which rotates with constant angular velocity about the earth center. The coordinate frame is positioned such that $\hat\imath$ is pointed tangent to the orbit in the direction of motion, $\hat k$ is directed toward the center of the earth, and $\hat\jmath$ is defined as $\hat k \times \hat\imath$ to give a right-handed system. Then the angular velocity of the satellite frame is $-\hat k\omega_0$ and the position of a generic point on the tether is

$$r = \hat\imath\, x_0 + \hat\jmath\, y_0 + \hat k\, z_0 \tag{27}$$

The total acceleration of a point referred to the rotating axes is

$$a = \ddot R_0 + \dot\omega \times r + \omega \times (\omega \times r) + 2\omega \times \dot r + \ddot r \tag{28}$$

where R_0 is a vector from the earth center to the origin of the rotating frame, and $\dot r$ and $\ddot r$ are velocity and acceleration vectors measured in the rotating frame. The total acceleration of point (x_0, y_0, z) is therefore

$$a = (\hat\imath\, \ddot x_0 + \hat\jmath\, \ddot y_0 + \hat k\, \ddot z_0) + \hat\imath(\omega_0^2\, x_0 - 2\omega_0\, \dot z_0) + \hat k(\omega_0^2(\rho_0 - z_0) + 2\omega_0\, \dot x_0) \tag{29}$$

The gravitational force on a particle of mass m is $F_g = mg\ \rho_E^2\ R/|R|^3$ where $R = -\hat\imath\, x_0 - \hat\jmath\, y_0 + \hat k\, (\rho_0 - z_0)$. It is helpful to express F_g in a power series and neglect terms of order ρ_0^{-2}. The binomial theorem gives

$$F_g = m\omega_0^2 \left[-\hat\imath\, x_0\ - \hat\jmath\, y_0 + \hat k(\rho_0 - z_0)\right] \left[(1 + \frac{3z_0}{\rho_0}) + 0\ (\rho_0^{-2})\right] \tag{30}$$

When $0(\rho_0^{-2})$ is neglected in the last equation, it is found that a particle subjected to gravity loading and an additional force $\hat\imath\, F_x + \hat\jmath\, F_y + \hat k\, F_z$ satisfies

$$m\left[\hat\imath\, \ddot x_0 + \hat\jmath\, \ddot y_0 + \hat k\, \ddot z_0\right] = \hat\imath\, F_x + \hat\jmath\, F_y + \hat k\, F_z + m\, \hat\imath(2\omega_0\dot z_0)$$
$$+ m\hat\jmath(-\omega_0^2 y_0) + m\hat k(3\omega_0^2\, z_0 - 2\omega_0\dot x_0) \tag{31}$$

This equation provides useful information regarding feasible two-dimensional states of motion in either the orbit plane $(\hat\imath, \hat k)$ or the plane transverse to the orbit plane $(\hat\jmath, \hat k)$. It is possible to have $y_0 = 0, F_y = 0$, and all other quantities dependent only on the x_0 and z_0 coordinates. However, if we is assume that $F_x = 0$ and $x_0 = 0$, then the gyroscopic term $\hat{i}\omega_0\dot{z}_0$ produces an acceleration in the x_0 direction. Consequently, pure motion in the y, z plane is not possible unless $\dot{z}_0$ is negligible. This circumstance can occur when a tether, initially in static equilibrium along the z axis, is given a small perturbation in the yz plane. Then the motion component for the axial direction is of second order magnitude in comparison to the transverse components. The linearized problem for small transverse motions from static equilibrium is worth further discussion.

TETHER MOTION FOR SMALL PERTURBATION ABOUT THE Z-AXIS

The problem for motion of a tether experiencing small oscillations about the z-direction in the orbital plane were previously investigated by M. Rheinfurth and Z. Galaboff [2]. This paper showed that the static tension for extension along the z-axis is

$$T = \frac{3}{2}\, M_T\, \omega_0^2\, \ell(\alpha + 1 - (\frac{z}{\ell})^2) \tag{32}$$

where M_T is the mass of a tether having length ℓ and a mass M_s attached at the free end. The parameter α equals $2M_s/M_T$. For small transverse oscillations, the static tension will not change substantially and motion in the z-direction wi'l be negligible. It is useful to employ dimensionless coordinates according to $\xi = z/\ell$ and $\tau = \omega_0 t$. Following the analysis of Rheinfurth and Galaboff, the linearized equations of motion accounting for both in-plane and out-of-plane motion are

$$1.5\, \frac{\partial}{\partial \xi}\left[(\alpha + 1 - \xi^2)\frac{\partial x}{\partial \xi}\right] = \frac{\partial^2 x}{\partial \tau^2} \tag{33}$$

$$1.5\, \frac{\partial}{\partial \xi}\left[(\alpha + 1 - \xi^2)\frac{\partial y}{\partial \xi}\right] - y = \frac{\partial^2 y}{\partial \tau^2} \tag{34}$$

The boundary conditions at $\xi = 0$ are $x = y = 0$. Force balance conditions on the end mass M_s gives

$$-3\,\frac{\partial x}{\partial \xi} = \frac{\partial^2 x}{\partial \tau^2} \tag{35}$$

$$-3\,\frac{\partial y}{\partial \xi} - y = \frac{\partial^2 x}{\partial \tau^2} \tag{36}$$

Assuming modal forms $X = X(\xi)\sin(\omega\tau)$ and $y = Y(\xi)\sin(\omega\tau)$ leads to

$$1.5\, \frac{d}{d\xi}\left[(\alpha + 1 - \xi^2)\frac{dX}{d\xi}\right] + \omega^2 X = 0 \tag{37}$$

$$1.5\, \frac{d}{d\xi}\left[(\alpha + 1 - \xi^2)\frac{dY}{d\xi}\right] + (\omega^2 - 1)Y = 0 \tag{38}$$

with corresponding boundary conditions of

$$X(0) = 0, \quad 3X'(1) + \omega^2 X(1) = 0 \tag{39}$$

$$Y(0) = 0, \quad 3Y'(1) + (\omega^2 - 1)Y(1) = 0 \tag{40}$$

It shows from the above equations that the modal functions for in-plane and out-of-plane motion are identical and a frequency ω_x for the in-plane motion also corresponds to an out-of-plane frequency

$$\omega_y = \sqrt{\omega_x^2 + 1} \tag{41}$$

Reference [2] treated the problem of in-plane motion in considerable detail and gave an exact solution for the case where $M_s = 0$, $(\alpha = 0)$. In that case, the exact dimensionless frequencies and modal functions are

$$\omega_{kx} = \sqrt{6k(k - .5)}, \qquad k = 1, 2, 3, \ldots \tag{42}$$

$$X_k(\xi) = P_{2k-1}(\xi), \;\; 0 \;\leq\; \xi \;\leq\; 1 \tag{43}$$

where ω_{kx} means the k'th in-plane frequency and P_{2k-1} is the Legendre polynomial of degree $2k - 1$. Rheinfurth and Galaboff showed that the first frequency equals $\sqrt{3}$ and this modal function is a straight line even for the case of a variable mass distribution. The general Rayleigh quotient formula from [2] expressed in dimensionless form is

$$\omega_{kx}^2 = \left[3 \int_0^1 (\alpha + 1 - \xi^2) \left[X_k'(\xi)\right]^2 \, d\xi\right] \Big/ \left[\alpha X_k^2(1) + 2 \int_0^1 X_k^2(\xi) \, d\xi\right] \tag{44}$$

when $M_s/M_T > 1$ and $n \geq 3$, the mode shapes are closely approximated by

$$X_n(\xi) \approx \sin\left[\pi(n - 1)\xi\right] \tag{45}$$

so the Rayleigh quotient gives

$$\omega_{kx} \approx \pi(n - 1)\sqrt{1 + 3M_s/M_T} \tag{46}$$

This formula is accurate within about 4% for $n \geq 3$ and $M_s/M_T > 1$.

A concise algorithm to compute frequencies and mode shapes for the general case can be obtained by employing PC-MATLAB [3] and finite differences as a formulation equivalent to that used in [1]. We want to solve

$$\frac{d}{d\xi}\left[(1 + \alpha - \xi^2)X'(\xi)\right] = -\lambda X(\xi) \quad \text{for } \alpha = 2M_s/M_T \text{ and } 0 \;\leq\; \xi \;\leq\; 1 \tag{47}$$

The boundary conditions are

$$X(0) = 0 \quad \text{and} \quad 2X'(1) = \lambda X(1) \tag{48}$$

This corresponds to a modal solution of the form

$$X(\xi,t) = X(\xi) \;\; \sin \;\; (\omega_0 \sqrt{1.5\lambda} \;\; t) \tag{49}$$

To express this problem in difference form we take

$$D = 1/N \;\; , \;\; z_j = j \; D \;\; , \;\; 0 \leq j \leq N \tag{50}$$

The boundary conditions are specified as

$$X_0 = 0, \;\; (X_{N-2} - 4X_{N-1} + 3X_N)D^{-1} = \lambda X_N \tag{51}$$

and the differential equation becomes

$$\left[j^2 - (1+\alpha)D^{-2}\right] [X_{j-1} - 2X_j + X_{j+1}] + j\,[X_{j+1} - X_j] = \lambda X_j, \;\; 2 \leq j \leq N-1. \tag{52}$$

This gives an eigenvalue problem $AX = \lambda X$ which is solvable using linear algebra operators in MATLAB. Figure 2 shows a function *tetfrg* which determines frequencies and mode shapes. The figure also lists dimensionless in-plane frequencies for several combinations of mass ratio. Computations were made using $npts = 50$. The first four modal vectors for $M_s/M_T = 0.5$ are also shown in Figure 3.

MATLAB FUNCTION TO COMPUTE TETHER INPLANE FREQUENCIES

```
function [frqs,modevecs] = tetfrq(npts,msatovmtet)
% [frqs,modevecs] = tetfrq(npts,msatovmtet)
% Frequencies and modal vectors for a tether with an
% end mass attached in a gravity gradient field.
%   npts          number of segments into which the tether
%                 length is divided for finite difference
%                 approximation
%   msatomtet     sub-satellite mass divided by the
%                 tether mass
%   frqs          frequencies for transverse planar motion
%                 divided by the  circular orbit frequency
%

% Technical reference: Mario H. Rheinfurth and Zachary
% J. Galaboff "Modal Analysis of a Nonuniform String
% With End Mass and Variable Tension", NASA Technical
% Paper 2198, August 1983
%
%   written by Howard Wilson, July 1989
%
a = zeros(npts,npts); d = 1/npts; b = (2*msatovmtet+1)/d^2;
a(1,1) = 2*(b-1); a(1,2) = -b; a(npts,npts-2) = 1/d;
a(npts,npts-1) = -4/d; a(npts,npts) = 3/d;
for j = 2:npts-1,  a(j,j-1) = j*(j-1)-b;
  a(j,j) = 2*(b-j*j);  a(j,j+1) = j*(j+1)-b;
end
[modevecs,frqs] = eig(a);
[frqs,id] = sort(sqrt(1.5*diag(frqs)));
modevecs = modevecs(:,id);
modevecs = [zeros(1,npts);modevecs];
```

TETHER INPLANE CIRCULAR FREQUENCY DIVIDED BY ORBIT ANGULAR SPEED

	Ratio of (Subsatellite Mass)/(Tether Mass)						
freq nmbr	0	0.5000	1.0000	1.5000	2.0000	2.5000	3.0000
1	1.7320	1.7320	1.7320	1.7320	1.7320	1.7320	1.7320
2	4.2425	5.5159	6.7081	7.7274	8.6294	9.4466	10.1990
3	6.7066	10.1300	12.7360	14.8847	16.7576	18.4400	19.9807
4	9.1551	14.9072	18.8904	22.1464	24.9748	27.5111	29.8311
5	11.5785	19.7213	25.0646	29.4180	33.1951	36.5799	39.6749
6	13.9503	24.5373	31.2296	36.6735	41.3939	45.6228	49.4888
7	16.2319	29.3394	37.3715	43.8992	49.5574	54.6256	59.2585
8	18.4152	34.1181	43.4807	51.0851	57.6751	63.5773	68.9722
9	20.5863	38.8659	49.5493	58.2226	65.7377	72.4680	78.6194
10	22.8428	43.5767	55.5703	65.3038	73.7366	81.2882	88.1900

Figure 2: Natural Frequency Analysis

ORIGINAL PAGE IS
OF POOR QUALITY

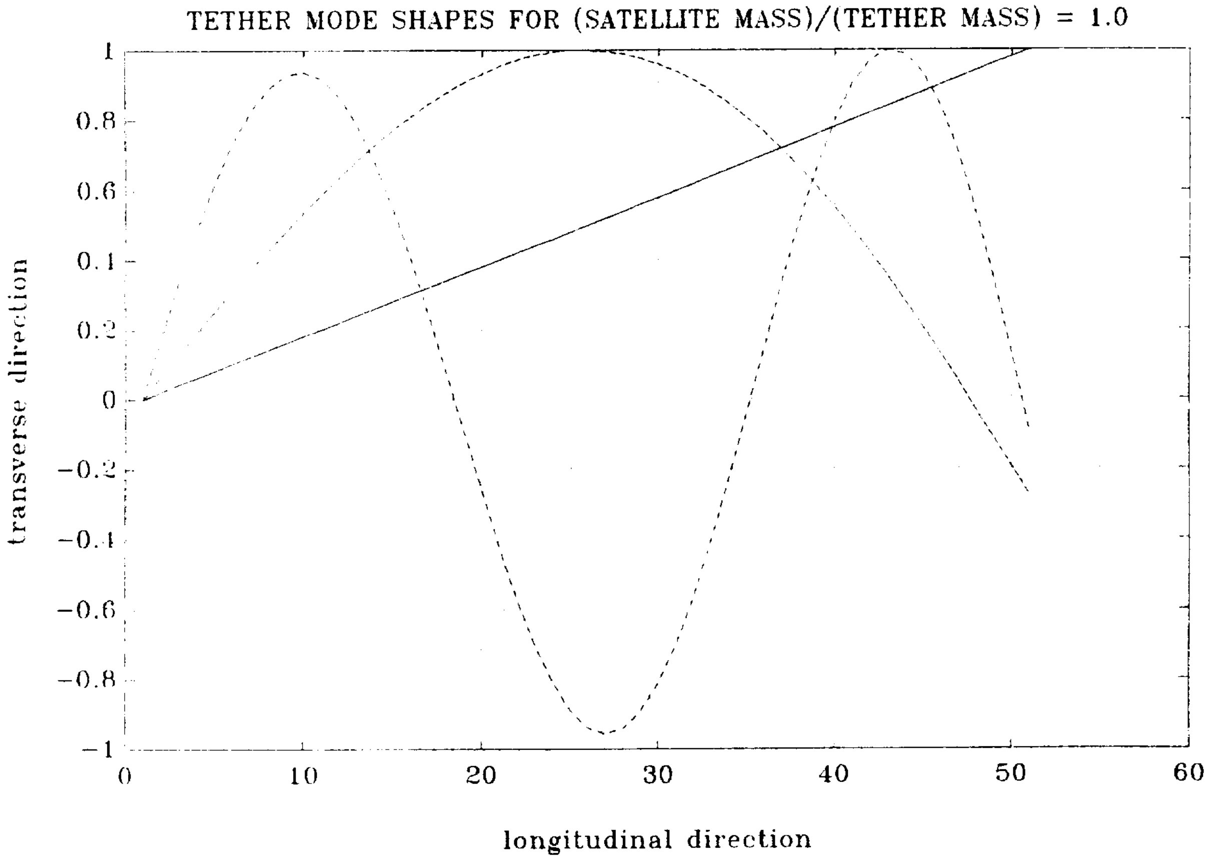

Figure 3: Tether Modal Shapes

COMPUTER IMPLEMENTATION

The 3D tether dynamics algorithm developed above was implemented in a MATLAB program having the following modules.

module	function
tetrun	Create data, call the integrator and print results
tetdat	Generate data for a tether moving in skip rope mode
tettim	Compute time when various lumped masses have been retrieved to the orbiter
tetlen	Compute the time dependent tether length and corresponding derivatives
tetforce	Compute gravity and Coriolis forces on the tether
tettop	Compute position, velocity and acceleration of the point to which the tether is retrieved
tetdef	Form the nonlinear equations of motion of the tether
odeven	Routine to compute solutions at even time increments
ode78	Seventh and eighth order Runge Kutta integrator based on Fehlberg formulas

At present, the program has not been checked fully although initial tension values computed for the skip rope are reasonable, angular values obtained are large. This is apparently caused by the necessity of dividing linear acceleration quantities by $\rho \sin \theta$ to get angular accelerations. Figures [4] and [5] show the angular displacements obtained for θ and φ. The motion rapidly becomes confined to the orbital plane and amplitudes of θ angles become large. The validity of these results seem questionable.

ORIGINAL PAGE IS
OF POOR QUALITY

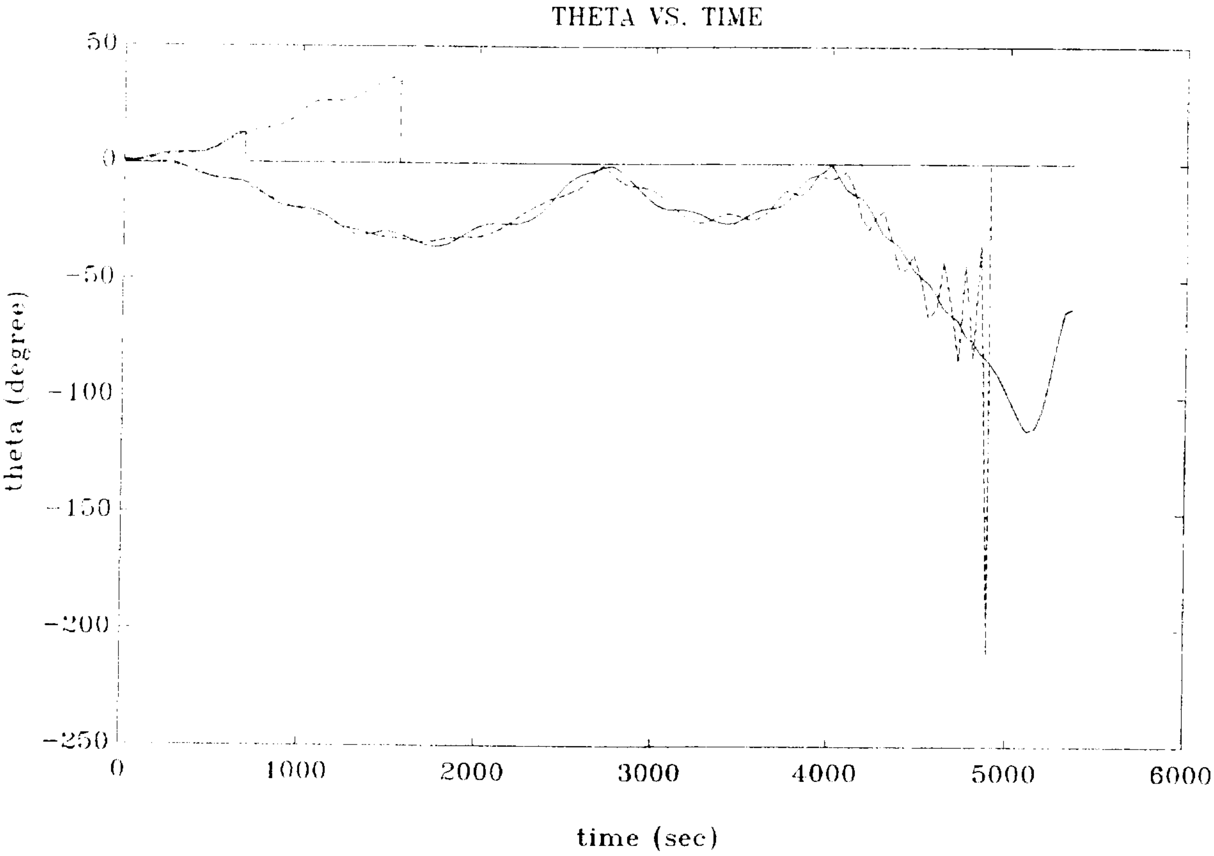

Figure 4

ORIGINAL PAGE IS
OF POOR QUALITY

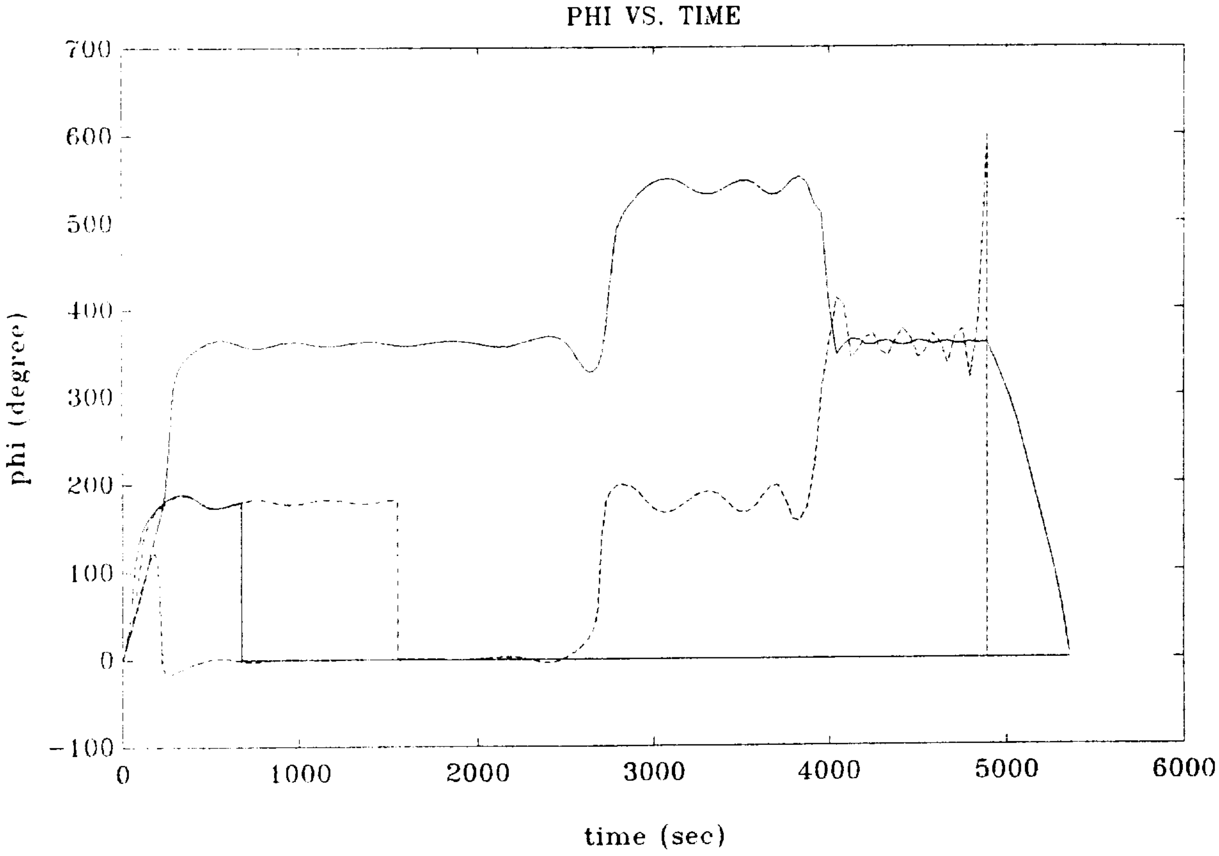

Figure 5

```
%      Program for 3D Dynamic Simulation of a Tether in a
%                  Gravity Gradient Field.
%                            by
%                Howard B. Wilson, Summer 1989

% The chosen data corresponds to a system which was analyzed by
% David B. Weaver in "Analysis of Transverse Wire Oscillations
% During The Tethered Satellite System Electromagnetic Mission",
% November 24,1987


% global variables for the tether analysis program

global nlink0_ lvec0_ lsum0_  cofexp_  gravity_
global omegac_ masvec0_ tretrv0_
global nlknow_ lvecnow_ masvcnow_


omegac_=1.1847e-3; % orbit angular speed in rad/sec
                   % The orbit period is 1.473 hours
gravity_=32.2;     % gravity constant in ft/sec^2
omegaskip=.01134;  % skip rope spin rate (rad/sec). This value
                   % is 11.31 times the orbit angular speed.
nlink0_=5;         % number of links used in the model
mtet=11.43;        % total tether mass in slugs
msat=37.7;         % satellite mass
z1=65617;          % satellite initial z coordinate in feet.
                   % This equals 20 kilometers
x1=10;             % Satellite is initially on the z axis
xsin=286.315;      % Amplitude of initial deflection curve
                   % which is half of a sine curve
omeg=1;            % Parameter defining the spatial period
                   % length of the initial defelction curve
ph0=0;             % Start initial motion in the x-z plane
phd0=omegaskip;    % Initial spin about z axis
thd1=0;            % Initial time derivatives of theta are zero.
thdn=0;
tfrac=7000;        % Time at which frac*(total_tether_length)
                   % has been retrieved
frac=.9;           % The solution is generated until
                   % frac*(total_tether_length) has been
                   % been retrieved

% Generate starting data

mt=mtet/nlink0_;  masvec0_=[mt+msat;mt*ones(nlink0_-1,1)];

[lvec0_,thvec0,phvec0,thdvec0,phdvec0,z0,x0]=...
tetdat(nlink0_,thd1,thdn,ph0,phd0,z1,x1,xsin,omeg);
% Compute times when successive tether masses reach the
% orbiter

[tretrv0_,lsum0_,cofexp_]=tettim(frac,lvec0_,tfrac);

yinit=[thvec0(:);phvec0(:);thdvec0(:);phdvec0(:)];

for k=1:nlink0_,
   nlknow_=nlink0_-k+1;
```

```
%  tmin=tretrv0_(k); tmax=.98*tretrv0_(k+1); h=(tmax-tmin)/50;
   tmin=tretrv0_(k); tmax=tretrv0_(k+1)-10; h=(tmax-tmin)/50;
   tol=1.e-12; trace=1;
   [tout,yout]=odeven('tetdif',tmin,tmax,h,yinit,tol,trace);

   ntp=2; nt=length(tout);
   yinit=lagtrp(tretrv0_(k+1),tout(nt-ntp+1:nt),yout(nt-ntp+1:nt,:))';
   tim=[tim;tout;tretrv0_(k+1)]; yout=[yout;yinit'];
   ytep=[];
   for j=4:-1:1,
      yinit(j*nlknow_)=[];
      yyyy=[yout(:,(j-1)*nlknow_+1:j*nlknow_),zeros(nt+1,k-1)];
      ytep=[yyyy,ytep];
   end
   zout=[zout;ytep];
end
save tet2

=====================================================

function [lng,th,ph,thd,phd,z,x]=tetdat(n,thd1,thdn,ph0,phd0,zn,...
                                 x0,xsin,omeg)
% [lng,th,ph,thd,phd,z,x]=tetdat(n,thd1,thdn,ph0,phd0,zn,...
%                           x0,xsin,omeg)

% Compute data to define an initial planar tether configuration

% n             Number of links
% thd1,thdn     Initial values of the time derivative of theta.
%               Theta_dot varies linearly from thd1 at the first
%               link to thdn at the last link.
% ph0,phd0      Inital values of phi and the time derivative of
%               phi. All links have the same inital values for
%               these variable.
% zn            The projection of the tether length on the z axis
% xsin          Amplitude of a sine wave which is added to a
%               linear taper to create the initial radial deflect-
%               ion in the x direction. This deflection has the
%               form x = x0*(1 - z/zn) + xsin*sin(pi*omeg*z/zn).
% x0            Transverse deflection value at the free end of the
%               tether.

% lng           Vector of lengths of the individual tether links.
%               These lengths are defined by cords on the chosen
%               deflection curve.
% th,ph         Initial theta and phi values.
% thd,phd       Initial time derivatives of theta and phi.
% z,x           Initial z and x coordinates viewed normal to the
%               plane of the system.

nn=(0:n)'/n;
z=zn*nn;
x=x0*(1-nn)+xsin*sin(pi*omeg*nn);
zdif=z(2:n+1)-z(1:n);
xdif=x(2:n+1)-x(1:n);
ph=ph0*ones(n,1);
phd=phd0*ones(n,1);
th=-atan2(xdif,zdif);
thd=thd1+(thdn-thd1)*nn(2:n+1);
lng=sqrt(xdif.*xdif+zdif.*zdif);
```

```
==========================================================

function [tvtrv,ltotl,cof]=tettim(frac,leng,tfinl)
% [tvtrv,ltotl,cof]=tettim(frac,leng,tfinl)

% This function computes the vector of times at which
% successive links in the tether reach the orbiter during
% retrieval according to an exponential length reduction rate.
% leng        The vector of link lengths. The links are
%             indexed with the bottom link as the first
%             and the base link as the last.
% frac        The fraction of the total length to be
%             retrieved by time tfinl.
% tfinl       The time at which frac*(total tether length)
%             has been taken into the orbiter according to
%             the exponential rate law.
% tvtrv       The vector of times at which link(n),...,
%             link(1) reach the orbiter.
% ltotl       The total tether length leng(1)+...+leng(1)
% cof         The exponent in the tether equation
%             Tether_length = ltotl*exp(cof*t)
n=length(leng); lngsum=flipx(cumsum( flipx(leng(:))));
ltotl=sum(leng); cof=log(1-frac)/tfinl;
lngsum=[lngsum;(1-frac)*ltotl];
tvtrv=log(lngsum/ltotl)/cof;

==========================================================

function [lvec,lvecd,lvecdd]=tetlen(t)
% [len,lend,lendd]=tetlen(t)
% Compute the vector of lengths and their time derivatives.
% Only the last link changes during the current time interval.

% global variables for the tether analysis program

% global nlink0_  lvec0_  lsum0_  cofexp_
% global omegac_  masvec0_  tretrv0_
% global nlknow_  lvecnow_  masvcnow_

%lvec=lvec0_; lvecd=zeros(lvec); lvecdd=zeros(lvec);
%break

% remaining statements are presently inactive. the tether
% has constant length

len=lsum0_*exp(cofexp_*t);
lend=cofexp_*len;
lendd=cofexp_*lend;
len=len-sum(lvec0_(1:nlknow_-1));
lvec=len; lvecd=lend; lvecdd=lendd;
if nlknow_ > 1
  lvec=[lvec0_(1:nlknow_-1);lvec];
  lvecd=[zeros(nlknow_-1,1);lvecd];
  lvecdd=[zeros(nlknow_-1,1);lvecdd];
end

==========================================================

function forc=tetforce(t,r,v)
```

```
% Force effects due to gavitational attraction and
% rotation in a gravity gradient field

% global variables for the tether analysis program

% global nlink0_ lvec0_ lsum0_  cofexp_  gravity_
% global omegac_ masvec0_ tretrv0_
% global nlknow_ lvecnow_ masvcnow_

wsq=omegac_*omegac_; w2=2*omegac_;
fx=w2*v(:,3).*masvec0_(1:nlknow_);
fy=-wsq*r(:,2).*masvec0_(1:nlknow_);
fz=( (3*wsq)*r(:,3)-w2*v(:,1) ).*masvec0_(1:nlknow_);
forc=[fx,fy,fz];
% 'in tetforce, (omegac_,masvec_,forc)'
% omegac_,masvec_,forc
% pause
%forc=[zeros(nlknow_,2),(gravity_*masvec0_(1:nlknow_)).*ones(nlknow_,1)];

===================================================

function [rtop,rtopd,rtopdd]=tettop(t)
% [rtop,rtopd,rtopdd]=tettop(t)
% Coordinates of the tether top and related derivatives.
% The top is fixed at present.
rtop  =[0; 0; 0];
rtopd =[0; 0; 0];
rtopdd=[0; 0; 0];

===================================================

function yt=lagtrp(t,tdat,ydat)
% yt=lagtrp(t,tdat,ydat)
% Lagrange interpolation of a vector function yt which
% is a function of scalar time. Interpolation is
% performed such that at tdat(i) the value of yt is
% ydat(i,:). Thus, yt is a polynomial of degree n-1
% where n is the number of components in tdat
[mrow,mcol]=size(ydat);
yt=zeros(1,mcol);
for k=1:mrow
  yt=yt+ydat(k,:)*plag(k,tdat,t);
end

===================================================

function y=plag(j,tdat,t)
%  y=plag(j,tdat,t)
% generates a Lagrange interpolating polynomial
td=tdat(:)'; td(j)=[];
y=prod(t-td)./prod(tdat(j)-td);

===================================================

function zp = tetdif(t,z)
% zp = tetdif(t,z)
% Dynamical equations for a lumped mass 3D tether model

%    written by Howard B. Wilson, Summer 1989
% global variables for the tether analysis program

% global nlink0_ lvec0_ lsum0_  cofexp_
% global omegac_ masvec0_ tretrv0_
% global nlknow_ lvecnow_ masvcnow_

% t, z
% nlnow_=5;

n4 = length(z); n = round(n4/4); w3 = ones(1,3);
n1 = n+1; n2 = n1+n; n3 = n2+n; zn = zeros(n,1);

% Obtain local angular coordinates and their derivatives

th = z(1:n); ph = z(n1:n2-1); thd = z(n2:n3-1); phd = z(n3:n4);
ct = cos(th); st = sin(th); cp = cos(ph); sp = sin(ph);
zp = zeros(z); zp(1:n2-1) = z(n2:n4);

% Lengths of tether links and corresponding derivatives defined
% by an external function

[len,lend,lendd] = tetlen(t);

% Reciprocals of mass elements needed later

masinv = 1 ./masvec0_; miv3 = masinv*w3;

% Compute unit base vectors in spherical coordinates
% along with spatial derivatives of those vectors

ur = [st.*cp,st.*sp,ct];
ut = [ct.*cp,ct.*sp,-st];
% size(ur)
% size(zn)
up = [-ur(:,2),ur(:,1),zn];
utt = -ur;
utp = [-ut(:,2),ut(:,1),zn];
upp = [-ur(:,1:2),zn];
len3 = len*w3;
thd3 = thd*w3;
phd3 = phd*w3;

% Local Cartesian components of position vectors and
% velocity components

r = len3.*ur;
% [size(len3);size(ut);size(thd3);size(up);size(phd3)]

v = len3.*(ut.*thd3+up.*phd3)+lend*w3.*ur;

% Compute base point motion defined by an external function

[RT,RTD,RTDD] = tettop(t);
RT = RT(:)'; RTD = RTD(:)'; RTDD = RTDD(:)';


% global coordinates and velocity vectors

R = ones(n,1)*RT+flipx(cumsum(flipx(r)));
V = ones(n,1)*RTD+flipx(cumsum(flipx(v)));
```

```
% Nodal forces defined by an external function depending on
% time, position and velocity

FRC = tetforce(t,R,V);

% Form the tridiagonal equations to solve for member tensions

if nlknow_ > 1,

   lsq = len.*len;
   b = lsq.*([masinv(2:n);0]+masinv(1:n));
   c = r(1:n-1,:).*r(2:n,:);
   c = -masinv(2:n).*sum(c')';
   fovm = [FRC.*miv3(1:n,:);RTDD];

   e = sum(   (r.*( fovm(1:n,:)-fovm(2:n+1,:) ) )'   )';
   e = e+sum((v.*v)')'-lend.*lend-len.*lendd;

% Solve for tension multipliers

   alp = trisol([0;c],b,c,e);

% Compute member forces

   tenfrc = [[0 0 0];(alp*w3).*r];

% Compute global accelerations of point masses

   RDD = (FRC+tenfrc(1:n,:)-tenfrc(2:n+1,:)).*miv3(1:n,:);
   RDD = [RDD;RTDD];

% Second derivatives of link radii

   rdd = RDD(1:n,:)-RDD(2:n+1,:);

else

   lsq = len.*len;
   b=lsq*masinv(1);
   fovm=FRC.*miv3(1,:)-RTDD;
   e=sum( (r.*fovm)');
   e=e+sum( (v.*v)' )-lend.*lend-len.*lendd;
   alp=e/b;
   tenfrc=(alp*w3).*r;
   RDD=(FRC-tenfrc).*miv3(1,:);
   rdd=RDD-RTDD;

end

% Compute local velocity dependent acceleration terms

a = (utt.*thd3+utp.*phd3).*thd3+(upp.*phd3+utp.*thd3).*phd3;
a = a.*len3;

% Compute the second derivatives of the spherical coordinate
% angles to complete formation of the equations of motion.

rdda = rdd-a;
thdd = sum((rdda.*ut)')'-2*lend.*thd;
thdd = thdd./len;
phdd = sum((rdda.*up)')'./(st.*st)-2*lend.*phd;
phdd = phdd./len;
zp(n2:n4) = [thdd;phdd];

% End

=========================================================

function [tout, yout] = odeven(F, t0, tfinal, h, y0, tol, trace)
%
% INPUT:
% F      - String containing name of user-supplied problem description.
%          Call: yprime = fun(t,y) where F = 'fun'.
%          t      - Time (scalar).
%          y      - Solution column-vector.
%          yprime - Returned derivative column-vector; yprime(i) = dy(i)/dt
% t0     - Initial value of t.
% tfinal- Final value of t.
% y0     - Initial value column-vector.
% h      - Time increment between which solution values are output
% tol    - The desired accuracy. (Default: tol = 1.e-3).
% trace - If nonzero, each step is printed. (Default: trace = 0).
%
% OUTPUT:
% tout   - Returned integration time points in increments of h
%          (kolumn-vector).
% yout   - Returned solution, one solution column-vector per tout-value.
%

tout=(t0:h:tfinal)'; ntims=length(tout); ncols=length(y0);
yout=zeros(ntims,ncols); yout(1,:)=y0(:)';
for j=1:ntims-1
  [ttemp,ytemp]=ode78(F, tout(j), tout(j+1), yout(j,:)', tol, trace);
  [nr,nc]=size(ytemp);
  yout(j+1,:)=ytemp(nr,:);
end


=========================================================

function [tout, yout] = ode78(F, t0, tfinal, y0, tol, trace)
% ODE78  Integrates a system of ordinary differential equations using
% 7th order formulas.
%
% [tout, yout] = ode78(F, t0, tfinal, y0, tol, trace)
%
% INPUT:
% F      - String containing name of user-supplied problem description.
%          Call: yprime = fun(t,y) where F = 'fun'.
%          t      - Time (scalar).
%          y      - Solution column-vector.
%          yprime - Returned derivative column-vector; yprime(i) = dy(i)/dt
% t0     - Initial value of t.
% tfinal- Final value of t.
% y0     - Initial value column-vector.
% tol    - The desired accuracy. (Default: tol = 1.e-6).
% trace - If nonzero, each step is printed. (Default: trace = 0).
%
% OUTPUT:
```

```
% tout  - Returned integration time points (row-vector).
% yout  - Returned solution, one solution column-vector per tout-value.
%
% The result can be displayed by: plot(tout, yout).

%   Daljeet Singh
%   Dept. Of Electrical Engg., The University of Alabama.
%   11-24-1988.

% The Fehlberg coefficients:
alpha = [ 2./27. 1/9 1/6 5/12 .5 5/6 1/6 2/3 1/3 1 0 1 ]';
beta = [ [  2/27  0  0   0   0  0  0  0  0  0  0   0  0  ]
[  1/36 1/12  0  0  0  0  0  0   0  0  0  0  0  ]
[  1/24  0  1/8  0  0  0  0  0  0  0  0  0  0 ]
[  5/12  0  -25/16  25/16  0  0  0  0  0  0   0  0  0  ]
[ .05   0  0  .25  .2  0  0  0  0  0  0  0  0 ]
[ -25/108  0  0  125/108  -65/27  125/54  0  0  0  0  0  0   0  ]
[ 31/300  0  0  0  61/225  -2/9  13/900  0  0  0   0  0  0  ]
[ 2  0  0  -53/6  704/45  -107/9  67/90  3  0  0  0  0  0  ]
[ -91/108  0  0  23/108  -976/135  311/54  -19/60  17/6  -1/12  0  0  0  0
[2383/4100 0 0 -341/164 4496/1025 -301/82 2133/4100 45/82 45/164 18/41 0 0
[ 3/205  0   0  0   0    -6/41  -3/205   -3/41     3/41   6/41   0   0  0
[-1777/4100 0 0 -341/164 4496/1025 -289/82 2193/4100 51/82 33/164 12/41 0
 chi = [ 0 0 0 0 0 34/105 9/35 9/35 9/280 9/280 0 41/840 41/840]';
 psi = [1  0  0  0  0  0  0  0  0  0  1 -1  -1 ]';
pow = 1/8;
if nargin < 6, trace = 0; end
if nargin < 5, tol = 1.e-6; end

% Initialization
t = t0;
% the following step parameters are used in ODE45
% hmax = (tfinal - t)/5;
% hmin = (tfinal - t)/20000;
% h = (tfinal - t)/100;
% The following parameters were taken because the integrator has
% higher order than ODE45. This choice is somewhat subjective.
hmax = (tfinal - t)/2.5;
hmin = (tfinal - t)/10000;
h = (tfinal - t)/50;
y = y0(:);
f = y*zeros(1,13);
tout = t;
yout = y.';
tau = tol * max(norm(y, 'inf'), 1);

if trace
%  clc, t, h, y
   clc, t, h
end
% The main loop
   while (t < tfinal) & (h >= hmin)
      if t + h > tfinal, h = tfinal - t; end

      % Compute the slopes
      f(:,1) = feval(F,t,y);
      for j = 1: 12
         f(:,j+1) = feval(F, t+alpha(j)*h, y+h*f*beta(:,j));
      end
      % Truncation error term
      gamma1 = h*41/840*f*psi;

      % Estimate the error and the acceptable error
      delta = norm(gamma1,'inf');
      tau = tol*max(norm(y,'inf'),1.0);

      % Update the solution only if the error is acceptable
      if delta <= tau
         t = t + h;
         y = y + h*f*chi;
         tout = [tout; t];
         yout = [yout; y.'];
      end
      if trace
%        home, t, h, y
         home, t, h
      end

      % Update the step size
      if delta ~= 0.0
         h = min(hmax, 0.8*h*(tau/delta)^pow);
      end
   end;

   if (t < tfinal)
      disp('SINGULARITY LIKELY.')
      t
   end
```

CONCLUSIONS AND RECOMMENDATIONS

The computer implementation of the 3D version tether dynamics program should be studied further to resolve numerical difficulties encountered when one or more of the θ-angles are small. An alternative formulation which may be appropriate for this case is to develop the equations of motion based on a different set of Euler angles than those presently used in the program. This work will be performed during the coming months.

REFERENCES

1. Baker, W. P. et al., "Tethered Satellite Study," NASA Technical Memorandum TM X-73314, Marshall Space Flight Center, Huntsville, Alabama, March 1976.

2. Rheinfurth, M. H. and Galaboff, Z. J. "Modal Analysis of a Nonuniform String With End Mass and Variable Tension, NASA Technical Paper 2198, Marshall Space Flight Center, Huntsville, Alabama, 1983.

3. MATLAB for MS-DOS Personal Computers, User's Guide, The MathWorks, Inc., 21 Eliot Street, South Natick, MA 01760.

4. Greenwood, D. T., Principles of Dynamics, 2nd Ed., Prentice-Hall, Englewood Cliffs, 1988.