Control System Design and Simulation of Spacecraft Formations via Virtual Formation Approach

Pavan Donepudi

Embry-Riddle Aeronautical University - Daytona Beach

Follow this and additional works at: https://commons.erau.edu/db-theses

Part of the Space Vehicles Commons

Scholarly Commons Citation
https://commons.erau.edu/db-theses/46

This thesis is brought to you for free and open access by Embry-Riddle Aeronautical University – Daytona Beach at ERAU Scholarly Commons. It has been accepted for inclusion in the Theses - Daytona Beach collection by an authorized administrator of ERAU Scholarly Commons. For more information, please contact commons@erau.edu.
CONTROL SYSTEM DESIGN AND SIMULATION OF SPACECRAFT FORMATIONS VIA VIRTUAL FORMATION APPROACH

By
Pavan Donepudi

A thesis submitted to the Physical Sciences Department
In Partial Fulfillment of the Requirements of
Master of Science in Space Science

Embry-Riddle Aeronautical University
Daytona Beach, FL 32117
2007
UMI Number: EP32089

INFORMATION TO USERS

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleed-through, substandard margins, and improper alignment can adversely affect reproduction. In the unlikely event that the author did not send a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

UMI Microform EP32089
Copyright 2011 by ProQuest LLC
All rights reserved. This microform edition is protected against unauthorized copying under Title 17, United States Code.

ProQuest LLC
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106-1346
CONTROL SYSTEM DESIGN AND SIMULATION OF SPACECRAFT FORMATIONS VIA VIRTUAL FORMATION APPROACH

By Pavan Donepudi

This thesis was prepared under the direction of the candidate's thesis committee chair, Dr. Mahmut Reyhanoglu, Department of Physical Sciences, and has been approved by the members of his thesis committee. It was submitted to the Department of Physical Sciences and was accepted in partial fulfillment of the requirements for the

Degree of

Master of Science in Space Sciences

THESIS COMMITTEE:

[Signatures]

Dr. Peter Erdman, MSSPS Graduate Program Coordinator

Dr. John Olivero, Department Chair, Physical Sciences

Dr. Christina Frederick-Recascino, Provost
ACKNOWLEDGEMENTS

I would like to express the deepest appreciation to my advisor Dr. Mahmut Reyhanoglu. Without his guidance and persistent help this project would not have been possible. Also, I would like to thank the Physical Sciences Department at Embry-Riddle Aeronautical University for awarding me both a scholarship and an assistantship, providing me with the financial means for the completion of the project. And finally, I would like to thank the many friends, colleagues, and relatives, who have helped me throughout my graduate education.
ABSTRACT

The problem of formation control of multiple spacecraft using the virtual structure approach is studied. The thesis first summarizes the progress made in the area of spacecraft formation control and then provides an overview of the virtual structure method. Modifying earlier established control design techniques, new feedback laws are constructed to control both the rotational and translational motion of a group of spacecraft. Several computer simulations are performed to illustrate the effectiveness of the modified control laws. The ephemeris and attitude data files generated via Matlab are exported to Satellite Tool Kit (STK) to create 3D animations. Various formation scenarios are modeled and the formation control strategies studied in this thesis are applied for switching from one formation scenario to another.
TABLE OF CONTENTS

TABLE OF CONTENTS ........................................................................................................ vi
LIST OF FIGURES ........................................................................................................... viii
LIST OF TABLES ............................................................................................................... ix

CHAPTER I ....................................................................................................................... 1
1 INTRODUCTION ............................................................................................................. 1
  1.1 Satellite Formation Control .......................................................... 1
     1.1.1 Leader-Follower Approach ................................................. 2
     1.1.2 Behavioral Approach ........................................................... 2
     1.1.3 Virtual Structure Approach .............................................. 3
  1.2 Contribution of Thesis ......................................................................................... 4
  1.3 Organization of Thesis ....................................................................................... 5

CHAPTER II ...................................................................................................................... 6
2 ROTATIONAL MOTION ................................................................................................. 6
  2.1 Reference Frames and Rotations ........................................................................ 6
  2.2 Euler’s Eigenaxis Rotation ................................................................................. 8
  2.3 Quaternions ........................................................................................................ 10
  2.4 Rotational Kinematics ....................................................................................... 12
  2.5 Rotational Dynamics .......................................................................................... 13

CHAPTER III .................................................................................................................... 14
3 DYNAMICS OF VIRTUAL STRUCTURE ..................................................................... 14
  3.1 Problem Statement ............................................................................................ 14
  3.2 Translational Dynamics .................................................................................... 15
  3.3 Rotational Dynamics ........................................................................................ 16
  3.4 Virtual Structure Control Laws without Formation Feedback ....................... 17
  3.5 Formation Feedback ........................................................................................ 19
  3.6 Formation Feedback Control Law .................................................................... 20
  3.7 Expansion/Contraction Maneuvers .................................................................. 21

CHAPTER IV ..................................................................................................................... 24
4 CONTROL SIMULATION ............................................................................................. 24
  4.1 Second Order Closed Loop System ................................................................... 24
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.2</td>
<td>Matlab Simulations</td>
<td>25</td>
</tr>
<tr>
<td>4.3</td>
<td>Matlab Results</td>
<td>28</td>
</tr>
<tr>
<td>4.4</td>
<td>Simulations Using STK (SATELLITE TOOL KIT)</td>
<td>50</td>
</tr>
<tr>
<td>4.4.1</td>
<td>Ephemeris File Format (*.e)</td>
<td>50</td>
</tr>
<tr>
<td>4.4.2</td>
<td>Attitude File Format (*.a)</td>
<td>52</td>
</tr>
<tr>
<td>4.4.3</td>
<td>STK Animations</td>
<td>55</td>
</tr>
<tr>
<td>CHAPTER V</td>
<td></td>
<td>58</td>
</tr>
<tr>
<td>5</td>
<td>CONCLUSION</td>
<td>58</td>
</tr>
<tr>
<td>5.1</td>
<td>Summary</td>
<td>58</td>
</tr>
<tr>
<td>5.2</td>
<td>Recommended Future Work</td>
<td>59</td>
</tr>
<tr>
<td>REFERENCES</td>
<td></td>
<td>60</td>
</tr>
<tr>
<td>APPENDIX A</td>
<td></td>
<td>63</td>
</tr>
<tr>
<td>APPENDIX B</td>
<td></td>
<td>89</td>
</tr>
<tr>
<td>APPENDIX C</td>
<td></td>
<td>92</td>
</tr>
</tbody>
</table>
LIST OF FIGURES

Figure 1: Euler's theorem ................................................................. 8
Figure 2: Virtual structure with three satellites........................................ 14
Figure 3: Block diagram of a virtual structure ........................................ 24
Figure 4: Transient response in formation maneuvers.................................. 25
Figure 5: Final desired formation of virtual structure ................................ 26
Figure 6: Positions of the 1st, 2nd, and 3rd satellites in the virtual frame (Case 1) ........... 33
Figure 7: Positions of the 4th, 5th, and 6th satellites in the virtual frame (Case 1) ........... 34
Figure 8: Quaternions of the satellites with respect to the earth frame (Case 1) ........... 35
Figure 9: Quaternions of the satellites with respect to the virtual frame (Case 1) ........... 36
Figure 10: Quaternions of the virtual frame with respect to the earth frame (Case 1) ....... 37
Figure 11: Angular velocities of the satellites relative to the virtual structure (Case 1) .... 37
Figure 12: Positions of the 1st, 2nd, and 3rd satellites in the virtual frame (Case 2) ........... 38
Figure 13: Positions of the 4th, 5th, and 6th satellites in the virtual frame (Case 2) ........... 39
Figure 14: Quaternions of the virtual frame with respect to the earth frame (Case 2) ....... 40
Figure 15: Quaternions of the satellites with respect to the earth frame (Case 2) ........... 41
Figure 16: Quaternions of the satellites with respect to the virtual frame (Case 2) ........... 42
Figure 17: Angular velocities of the satellites and virtual structure with respect to the earth (Case 2) ................................................................................................. 43
Figure 18: Positions of the 1st, 2nd, and 3rd satellites in the virtual frame (Case 3) ........... 44
Figure 19: Positions of the 4th, 5th, and 6th satellites in the virtual frame (Case 3) ........... 45
Figure 20: Quaternions of the virtual frame with respect to the earth (Case 3) ........... 46
Figure 21: Quaternions of the satellites with respect to the earth frame (Case 3) ........... 47
Figure 22: Angular velocities of the satellites and the virtual frame with respect to the earth (Case 3) ................................................................................................. 49
Figure 23: Hexagonal formation (1st type) ................................................. 55
Figure 24: Hexagonal formation (2nd type) ................................................. 56
Figure 25: Formation switching - from hexagon to straight line ......................... 56
Figure 26: Triangular formation .................................................................... 56
LIST OF TABLES

Table 1: Control gains and physical parameters ................................................................. 28
Table 2: Initial positions and velocities .............................................................................. 29
Table 3: Initial quaternions ................................................................................................. 30
Table 4: Desired positions ($r = 0.5 \text{ km}$) ................................................................. 30
Table 5: Desired quaternions .............................................................................................. 30
Table 6: Ephemeris file keywords ..................................................................................... 51
Table 7: Quaternion file keywords ...................................................................................... 53
CHAPTER I

1 INTRODUCTION

1.1 Satellite Formation Control

With the emergence of pico- and nano-satellites as a viable alternative to large expensive spacecraft, the control and coordination of formation of multiple spacecrafts has received significant attention in recent years. Complex missions are performed utilizing a network of distributed simple satellites. However the coordination of these satellites is not altogether simple. Many different control schemes have been developed and proposed for the formation control of the satellites. The objective of formation flying control design is to command and control individual vehicles as a single system. The primary task of any control scheme includes

1. Formation planning
2. Formation keeping
3. Reconfiguration
4. Collision avoidance
5. Fault detection isolation and recovery

Numerous papers have been published on different types of control schemes, some dealing with translational control only ([4], [5], [10], [13], [14]), or on rotational control only ([2], [6], [7]), and several papers include both translational and rotational control ([1], [3], [8], [9], [12], [15], [16]).
Formation control can in general be classified into three categories:

1. Leader-follower approach
2. Virtual control approach
3. Behavioral approach

1.1.1 Leader-Follower Approach

Leader-follower method is probably the most widely studied approach in formation control of multiple spacecraft. One of the agents is designated as the leader and the rest of the agents are designated as followers. The leader tracks a pre-defined trajectory, and the followers track a transformed version of the leader’s states.

Advantages:

1. Group behavior is directed by specifying the behavior of a single agent labeled as the leader.

Disadvantages:

1. It uses no formation feedback.
2. Leader is a single point of failure in the formation.

1.1.2 Behavioral Approach

In the behavioral approach several desired behaviors are prescribed for each agent. The basic idea is to make the control action of each agent a weighted average of the control for each behavior. Possible behaviors include collision avoidance, obstacle avoidance, goal seeking, and formation keeping.
Advantages:

1. It is natural to derive control strategies when agents have multiple competing objectives.
2. There is an explicit feedback to the formation since each agent reacts according to the position of its neighbors.
3. It lends itself naturally to a decentralized implementation.

Disadvantages:

1. The group behavior cannot be explicitly defined, rather the group behavior is said to “emerge.”
2. It is difficult to analyze the behavioral approach mathematically and guarantee its group stability.

1.1.3 Virtual Structure Approach

Virtual structure, as the name implies, has an imaginary satellite placed at the center of the formation and all the satellites are positioned relative to this satellite. It is analogous to the center of gravity. This way we make sure that there is no possibility of failure of the virtual satellite unlike the leader because it is an imaginary point. In the virtual structure approach, the entire formation is treated as a single structure. The virtual structure can evolve as a rigid body in a given direction with some given orientation and maintain a rigid geometric relationship among multiple vehicles.

Advantages:

1. It is fairly easy to prescribe a coordinated behavior of the group.
2. Formation feedback is possible.
Disadvantage:

1. The convergence speed is limited to the selection of gains used in the control laws.

In the case of the application of synthesizing multiple spacecraft interferometers in deep space, it is desirable to have a constellation of spacecraft act as a single rigid body in order to image stars in deep space. As a result, it is suitable to choose the virtual structure approach to accomplish formation maneuvers.

1.2 Contribution of Thesis

This thesis presents an effective formation feedback control via virtual structure approach and provides a 3-D simulation. Both translational and rotational motions of a group of spacecraft are implemented using established control schemes. The success of the virtual control strategy is tested for the formation to track a wide variety of formation scenarios with reasonable convergence speeds. The thesis first summarizes formation control without formation feedback and then presents a framework for including formation feedback in the control scheme. Toward the end the control laws are modified to include group maneuvers with expansion/contraction of the entire formation. Computer simulations are carried out using Matlab to generate the ephemeris and attitude data file, which are exported to Satellite Tool Kit (STK) to create 3D animations. Explanations as to how Matlab is used to implement the control laws, produce the data files, and generate 3-D animations, are given. Switching between different formation scenarios is also studied.
1.3 Organization of Thesis

The organization of the thesis is as follows. Chapter 2 begins with introduction to reference frames. Then following the development in [17], a number of concepts related to inertial and non-inertial reference frames, Euler's eigenaxis rotation and quaternions are briefly summarized. Finally the rotational kinematics and dynamics equations are presented. Chapter 3 introduces the control laws developed in [11] for the virtual structure. The coupled dynamics of the individual satellite and the virtual structure are explained. The idea of formation feedback is introduced and control laws are modified to accommodate this concept. In Chapter 4, the general architecture of the virtual structure and the importance of the gains used in the virtual structure are discussed. The effectiveness of the modified control laws is illustrated through Matlab simulations and 3D animations using STK. Finally, Chapter 5 concludes this thesis by addressing the future research directions in the area of spacecraft formation control.
CHAPTER II

2 ROTATIONAL MOTION

2.1 Reference Frames and Rotations

The attitude of a rigid spacecraft is most conveniently defined with a set of axes fixed to the spacecraft. This set of axes is called a body coordinate frame. The attitude of the spacecraft is then defined as a coordinate transformation that transforms a set of reference coordinates into the body coordinates of the spacecraft. The basic three-axis attitude transformation is based on direction cosine matrix.

Consider a reference frame A with a dextral orthonormal triad \( \{ \hat{a}_1, \hat{a}_2, \hat{a}_3 \} \) and a frame B with another dextral triad \( \{ \hat{b}_1, \hat{b}_2, \hat{b}_3 \} \). Basis vectors \( \{ \hat{b}_1, \hat{b}_2, \hat{b}_3 \} \) of B are expressed in terms of basis vectors \( \{ \hat{a}_1, \hat{a}_2, \hat{a}_3 \} \) of A as follows:

\[
\begin{bmatrix}
\hat{b}_1 \\
\hat{b}_2 \\
\hat{b}_3 \\
\end{bmatrix} =
\begin{bmatrix}
C_{11} & C_{12} & C_{13} \\
C_{21} & C_{22} & C_{23} \\
C_{31} & C_{32} & C_{33}
\end{bmatrix}
\begin{bmatrix}
\hat{a}_1 \\
\hat{a}_2 \\
\hat{a}_3 \\
\end{bmatrix}
\]

where \( C_{ij} = \hat{b}_i \cdot \hat{a}_j \) is the cosine of the angle between \( \hat{b}_i \) and \( \hat{a}_j \), and \( C_{ij} \) is called the direction cosine. The direction cosine matrix \( \mathbf{C}_{BA} = [C_{ij}] \) is also called the rotation matrix or coordinate transformation matrix from A to B. We often use \( \mathbf{C} \) for \( \mathbf{C}_{BA} \).

The direction cosine matrix is a highly redundant method of describing a relative orientation. The matrix \( \mathbf{C} \) has nine entries, but because of orthogonality condition
there are six redundant parameters. Thus, the minimum number of parameters required to describe a reference frame orientation is three.

The most commonly used sets of attitude parameters are the Euler angles. They describe the attitude of frame B relative to A through three successive rotation angles about the sequentially displaced body-fixed axes. The first rotation is about any axis. The second rotation is about either of the other two axes. The third rotation is then about either of the two axes not used for the second rotation.

The Euler angles provide a compact, three-parameter attitude description whose coordinates are easy to visualize. One major drawback of these angles is their inherent geometric singularity which limits their use in describing large rotations. Also, both the rotation matrix and the kinematics equations involve numerous computations of trigonometric functions. Quaternions provide a four-parameter singularity free representation that does not require the calculation of any trigonometric functions. Therefore, quaternions are well suited for onboard real-time computations. Thus, spacecraft orientation is now commonly described in terms of quaternions.

Quaternions, unlike the Euler angles, use one axis called an “eigenaxis” to rotate between coordinate systems. In the subsequent development, we will first briefly review the attitude kinematics and dynamics formulation used in this thesis to obtain the rotational equations of motion for a group of spacecraft. For full details, the reader is referred to [17].
2.2 Euler’s Eigenaxis Rotation

Euler’s theorem states that any given sequence of rotations can be represented as a single rotation about a single fixed axis called eigenaxis. More specifically, by rotating a rigid body about an axis that is fixed to the body frame B and stationary in the reference frame A, the rigid body’s attitude can be changed from a given orientation to any other orientation.

$$\vec{e} = e_1 \vec{a}_1 + e_2 \vec{a}_2 + e_3 \vec{a}_3 \quad \text{(2.1)}$$

$$= e_1 \vec{b}_1 + e_2 \vec{b}_2 + e_3 \vec{b}_3 \quad \text{(2.2)}$$

i.e. the eigenaxis remains unchanged by the rotation. Consequently, the rotation is characterized by
To parameterize the transformation matrix in terms of $\vec{e}$ and $\theta$, a sequence of Euler's successive rotations is used as follows:

1) Rotate frame $A$ with $R$ to align the $\vec{a}_i$ axis with $\vec{e}$. We will name this new frame after rotation as $A'$ so that

$$C_{A'A} = R = \begin{bmatrix} e_1 & e_2 & e_3 \\ R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix} \quad (2.4)$$

2) Now rotate frame $A'$ into $A''$ about the direction $\vec{e}$ through an angle $\theta$. Then, it is clear that

$$C_{A'A'} = C_1(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & \sin \theta \\ 0 & -\sin \theta & \cos \theta \end{bmatrix} \quad (2.5)$$

3) Now we rotate $A''$ through the inverse matrix $R^{-1} = R^T$ to compensate for the rotation in (1). We do this so that the frame $A''$ now will be aligned with $B$, since these three successive rotations can be combined as

$$C_{BA} = R^T C_1(\theta) R$$

$$= \begin{bmatrix} c\theta + e_1^2 (1-c\theta) & e_2 e_2' (1-c\theta) + e_3 s\theta & e_3 e_3' (1-c\theta) - e_2 s\theta \\ e_2 e_1' (1-c\theta) - e_3 s\theta & c\theta + e_2^2 (1-c\theta) & e_3 e_3' (1-c\theta) + e_2 s\theta \\ e_3 e_1' (1-c\theta) + e_2 s\theta & e_3 e_2' (1-c\theta) - e_2 s\theta & c\theta + e_3^2 (1-c\theta) \end{bmatrix} \quad (2.6)$$

where $c\theta = \cos \theta$, $s\theta = \sin \theta$. This is the parameterization of the transformation matrix $C_{BA}$ in terms of $\vec{e}$ and $\theta$. Introducing the rotation
it can be shown that $C_{BA}$ can be rewritten as

$$C(\vec{e}, \theta) = \vec{e} \vec{e}^T + (I - \vec{e} \vec{e}^T) \cos \theta - \vec{E} \sin \theta$$  \hspace{1cm} (2.8)$$

where $I$ is the identity matrix.

### 2.3 Quaternions

Quaternions or Euler parameters are defined as

$$q_1 = e_1 \sin(\theta / 2)$$ \hspace{1cm} (2.9)

$$q_2 = e_2 \sin(\theta / 2)$$ \hspace{1cm} (2.10)

$$q_3 = e_3 \sin(\theta / 2)$$ \hspace{1cm} (2.11)

$$q_4 = \cos(\theta / 2)$$ \hspace{1cm} (2.12)

where $\theta$ is the rotation angle about the Euler axis $\vec{e}$ vector. In terms of the eigenangle and the eigenaxis vector, the vector part of the quaternion $\vec{q} = (q_1, q_2, q_3)^T$ can be expressed as

$$\vec{q} = \vec{e} \sin(\theta / 2)$$

We will use the following notation in the subsequent sections of this thesis:

$$\mathbf{q} = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} = \begin{bmatrix} \vec{q} \end{bmatrix}$$

The (unit) quaternions are not independent of each other but constrained by the relation

$$\mathbf{q}^T \mathbf{q} = \vec{q}^T \vec{q} + q_4^2 = q_1^2 + q_2^2 + q_3^2 + q_4^2 = 1$$  \hspace{1cm} (2.13)$$
The transformation matrix can be parameterized in terms of quaternions as

\[
C = \begin{bmatrix}
1 - 2(q_2^2 + q_3^2) & 2(q_4q_3 - q_2q_4) & 2(q_4q_1 - q_3q_4) \\
2(q_3q_1 - q_2q_4) & 1 - 2(q_1^2 + q_4^2) & 2(q_2q_3 + q_1q_4) \\
2(q_3q_1 + q_2q_4) & 2(q_1q_2 - q_3q_4) & 1 - 2(q_1^2 + q_2^2)
\end{bmatrix}
\]  \hspace{1cm} (2.14)

From the equation above it can be seen that changing the signs of the quaternions does not change the transformation matrix. Given a certain orientation, there are actually two sets of quaternions that will describe the same orientation. This is due to the non-uniqueness of the principal rotation elements themselves. Define the skew symmetric matrix \(Q\) as

\[
Q = \begin{bmatrix}
0 & -q_3 & q_2 \\
q_3 & 0 & -q_1 \\
-q_2 & q_1 & 0
\end{bmatrix}
\]  \hspace{1cm} (2.15)

Then the transformation matrix can be written as

\[
C = (2q_4^2 - 1)I + 2\tilde{q}\tilde{q}^T + 2q_4Q
\]  \hspace{1cm} (2.16)

The conjugate of a quaternion, which represents a rotation of \(-\theta\) about \(\vec{e}\), is given by

\[
\mathbf{q}^* = \begin{bmatrix}
\tilde{q} \\
q_4
\end{bmatrix}^T = \begin{bmatrix}
-\tilde{q} \\
q_4
\end{bmatrix}
\]  \hspace{1cm} (2.17)

such that \(C(\mathbf{q}^*) = C^T(\mathbf{q})\).

A very important composite rotation property of the quaternion is the manner in which they allow two sequential rotations to be combined into one overall composite rotation.

\[
C(\mathbf{q}) = C(\mathbf{q}')C(\mathbf{q}'')
\]  \hspace{1cm} (2.18)

which can also be written as

\[
\mathbf{q} = \mathbf{q}^* \mathbf{q}'
\]  \hspace{1cm} (2.19)
Using equation (2.14) above and equating corresponding elements leads to the elegant transformation shown below.

\[
\begin{bmatrix}
q_1 \\
q_2 \\
q_3 \\
q_4
\end{bmatrix} =
\begin{bmatrix}
q_1' \\
q_2' \\
q_3' \\
q_4'
\end{bmatrix} =
\begin{bmatrix}
q_4' & q_3' & -q_2' & q_1' \\
-q_3' & q_4' & q_1' & q_2' \\
q_2' & -q_1' & q_3' & q_4' \\
-q_1' & -q_2' & -q_3' & q_4'
\end{bmatrix}
\begin{bmatrix}
q_1 \\
q_2 \\
q_3 \\
q_4
\end{bmatrix}
\] (2.20)

This is known as the quaternion multiplication in matrix form. Note that (2.18), (2.19) and (2.20) are equivalent expressions.

### 2.4 Rotational Kinematics

Let \( \bar{\omega} = (\omega_1, \omega_2, \omega_3)^T \) denote the spacecraft angular velocity in the body frame. Note that, in the subsequent sections, we will use the notation \( \bar{\omega} \) and \( \omega \) interchangeably. It can be shown that

\[
\begin{align*}
\omega_1 &= 2(\dot{q}_1 q_4 + \dot{q}_2 q_3 - \dot{q}_3 q_2 - \dot{q}_4 q_1) \\
\omega_2 &= 2(\dot{q}_2 q_4 + \dot{q}_3 q_1 - \dot{q}_1 q_3 - \dot{q}_4 q_2) \\
\omega_3 &= 2(\dot{q}_3 q_4 + \dot{q}_1 q_2 - \dot{q}_2 q_1 - \dot{q}_3 q_4) \\
0 &= 2(\dot{q}_4 q_1 + \dot{q}_2 q_2 + \dot{q}_3 q_3 + \dot{q}_4 q_4)
\end{align*}
\] (2.21)

which can be written in matrix form as

\[
\begin{bmatrix}
\omega_1 \\
\omega_2 \\
\omega_3 \\
0
\end{bmatrix} = 2
\begin{bmatrix}
q_4 & q_3 & -q_2 & -q_1 \\
-q_3 & q_4 & q_1 & -q_2 \\
q_2 & -q_1 & q_3 & q_4 \\
-q_1 & -q_2 & -q_3 & q_4
\end{bmatrix}
\begin{bmatrix}
\dot{q}_1 \\
\dot{q}_2 \\
\dot{q}_3 \\
\dot{q}_4
\end{bmatrix}
\] (2.22)

This can be rewritten as
The rotational kinematics can also be expressed as

\[
\dot{q} = \frac{1}{2} (q_4 \bar{\omega} - \bar{\omega} \times \bar{q}) \tag{2.24}
\]

\[
\dot{q}_4 = -\frac{1}{2} \bar{\omega}^T \bar{q} \tag{2.25}
\]

### 2.5 Rotational Dynamics

Angular momentum of a satellite or any rigid body is given by \( L = J\omega \), where \( J \) is the inertia tensor given by

\[
J = \begin{bmatrix}
  j_{11} & j_{12} & j_{13} \\
  j_{21} & j_{22} & j_{23} \\
  j_{31} & j_{32} & j_{33}
\end{bmatrix}
\]

We will assume that the body frame \( B \) is chosen as the principal axis frame so that the inertia tensor is diagonal:

\[
J = \begin{bmatrix}
  j_1 & 0 & 0 \\
  0 & j_2 & 0 \\
  0 & 0 & j_3
\end{bmatrix} \tag{2.26}
\]

Let \( T \) denote the net torque acting on the spacecraft. Then the rotational dynamics equations can be written as

\[
T = \left( \frac{dL}{dt} \right)_A = \left( \frac{dL}{dt} \right)_B + \omega \times L = J\omega + \omega \times J\omega \tag{2.27}
\]
CHAPTER III

3  DYNAMICS OF VIRTUAL STRUCTURE

3.1  Problem Statement

We first consider a spacecraft formation scheme that does not involve formation feedback from the virtual structure. Figure 2 shows a virtual structure with 3 satellites. We treat the whole formation as a single rigid body. The place-holders trace out the trajectories of each satellite and the whole virtual structure as the satellites start from arbitrary positions.
and come into the desired final formation. In Figure 2, the reference frames $C_o$ and $C_F$ denote the Newtonian inertial reference frame and the non-inertial virtual formation frame, respectively. The three satellites start from $C_1, C_2, C_3$ and are required to reach the final desired formation $C_1^d, C_2^d, C_3^d$.

### 3.2 Translational Dynamics

We assume that the inertial frame $C_o$ represents the Geocentric Equatorial Frame. In this frame X-axis points toward the first point of Aries (the position of Sun at vernal equinox), Z-axis points toward geographic north, and Y-axis completes the right-handed orthogonal frame. The $i^{th}$ satellite can be identified by $\mathbf{r}_i, \mathbf{v}_i, \mathbf{q}_i, \mathbf{\omega}_i$ representing position, velocity, unit quaternion and angular velocity in the inertial frame or $\mathbf{r}_{iF}, \mathbf{v}_{iF}, \mathbf{q}_{iF}, \mathbf{\omega}_{iF}$ in the virtual formation frame. The vectors $\mathbf{r}_{iF}, \mathbf{v}_{iF}, \mathbf{q}_{iF}, \mathbf{\omega}_{iF}$ represent the desired final parameters of the satellite in the virtual reference frame.

The translational motion of the individual satellites satisfies

$$\left[ \mathbf{r}_i^d(t) \right]_o = \left[ \mathbf{r}_F(t) \right]_o + C_{oF}(t) \left[ \mathbf{r}_{iF}^d \right]_F$$

(3.1)

$$\left[ \mathbf{v}_i^d(t) \right]_o = \left[ \mathbf{v}_F(t) \right]_o + C_{oF}(t) \left[ \mathbf{v}_{iF}^d \right]_F + (\mathbf{\omega}_F)_o \times \left( C_{oF}(t) \left[ \mathbf{r}_{iF}^d \right]_F \right)$$

(3.2)

where $C_{oF}$ is the transformation matrix of $C_o$ with respect to $C_F$. We use the transformation matrix $C_{oF}$ to express all the parameters in the Geocentric Equatorial Frame. It is clear that
\[ C_{oF} = \left[ (2q_F^2 - 1)I + 2\bar{q}_F \bar{q}_F^T + 2q_F Q \right] \] (3.3)

i.e. \( C_{oF} = \begin{bmatrix}
1 - 2(q_{2F}^2 + q_{3F}^2) & 2(q_{1F}q_{2F} + q_{3F}q_{4F}) & 2(q_{1F}q_{3F} - q_{2F}q_{4F}) \\
2(q_{2F}q_{1F} - q_{3F}q_{4F}) & 1 - 2(q_{1F}^2 + q_{3F}^2) & 2(q_{2F}q_{3F} + q_{1F}q_{4F}) \\
2(q_{3F}q_{1F} + q_{2F}q_{4F}) & 2(q_{3F}q_{2F} - q_{1F}q_{4F}) & 1 - 2(q_{1F}^2 + q_{2F}^2)
\end{bmatrix} \) (3.4)

The translational dynamics for the virtual structure as a rigid body is given by

\[
\begin{bmatrix}
\dot{r}_F \\
\dot{\omega}_F
\end{bmatrix}_o = \begin{bmatrix}
v_F \\
\omega_F
\end{bmatrix}_o
\] (3.5)

\[
\begin{bmatrix}
M_F \ddot{r}_F \\
M_F \ddot{\omega}_F
\end{bmatrix}_o = \begin{bmatrix}
f_F \\
\tau_F
\end{bmatrix}_o
\] (3.6)

### 3.3 Rotational Dynamics

The rotational motion of individual satellites satisfies

\[
\begin{bmatrix}
q^d_i(t) \\
\omega^d_i(t)
\end{bmatrix}_o = \begin{bmatrix}
q_F(t) \\
\omega_F(t)
\end{bmatrix}_o + C_{oF}(t) \begin{bmatrix}
q^d_i \\
\omega^d_i
\end{bmatrix}_F
\] (3.7)

Note that equation (3.7) is obtained using the quaternion multiplication rule given by equation (2.19).

The rotational kinematics of the virtual structure are given by

\[
\begin{bmatrix}
\dot{q}_4 \\
\dot{q}_1
\end{bmatrix}_o = \frac{1}{2}(q_{4F} \bar{\omega}_F - \bar{\omega}_F \times q_{4F})
\] (3.9)
which can also be written as

$$
\begin{bmatrix}
\dot{q}_F \\
\end{bmatrix}_o = \frac{1}{2} \Omega(\omega_F) q_F
$$

(3.10)

where

$$
\Omega(\omega) = \begin{bmatrix}
-\omega_x & \omega_y \\
-\omega_y & \omega_z
\end{bmatrix}, \quad \omega^T = \begin{bmatrix}
0 & -\omega_z & \omega_x \\
\omega_z & 0 & -\omega_y \\
-\omega_x & \omega_y & 0
\end{bmatrix}, \quad \omega^T = [\omega_1 \, \omega_2 \, \omega_3]
$$

and

$$
\begin{bmatrix}
J_F \dot{\omega}_F \\
\end{bmatrix}_o = -\omega_F \times J_F \omega_F + T_F
$$

(3.11)

### 3.4 Virtual Structure Control Laws without Formation Feedback

This section starts with combining translational and rotational equations of motion explained above to come up with the virtual structure control laws. The development here follows that in [11]. The control architecture for a virtual structure is derived in 4 steps.

1) The desired dynamics of the virtual structure are defined.

2) The motion of the virtual structure is translated to the satellites making up the virtual structure so that they can track the virtual satellite.

3) The tracking control laws for the satellites are generated to check that they are maintaining the desired formation.

4) Finally we introduce the formation feedback from the spacecraft to the virtual structure.

We know that the rotational motion for the individual satellites is described by
\[ \dot{q} = \frac{1}{2} \Omega(\omega)q \]
\[ J \dot{\omega} = -\omega \times J \omega + T \]

The following torque control laws [11] will be used for the virtual structure and for the individual satellites.

\[
T_F = k_q q_{e_F} - k_\alpha \|q_F - q_{d_F}\|^2 \omega_F
\]
\[
T_i = -\omega_i \times J_i \omega_i + J_i \omega_i^d + \frac{1}{2} \omega_i \times J_i (\omega_i + \omega_i^d) + k_q q_{e_i} - K_{\alpha_i} (\omega_i - \omega_i^d)
\]

where \( \| \cdot \| \) denote the Euclidean metric. The combined closed loop translational and rotational dynamics for the virtual structure are given as follows [11]:

\[
\begin{bmatrix}
\dot{r}_F \\
\dot{v}_F \\
\dot{q}_F \\
J_F \dot{\omega}_F
\end{bmatrix}
= \begin{bmatrix}
\begin{bmatrix}
v_F \\
-K_i (r_F - r_i^d) - k_p \|r_F - r_i^d\|^2 v_F
\end{bmatrix} \\
\frac{1}{2} \Omega(\omega_F) q_F \\
-\omega_F \times J_F \omega_F + k_q q_{e_F} - k_\alpha \|q_F - q_{d_F}\|^2 \omega_F
\end{bmatrix}
\]

Similarly, the closed loop translational and rotational dynamics of the individual satellites are given as [11]:

\[
\begin{bmatrix}
\dot{r}_i \\
\dot{v}_i \\
\dot{q}_i \\
J_i \dot{\omega}_i
\end{bmatrix}
= \begin{bmatrix}
\begin{bmatrix}
v_i \\
\dot{v}_i^d - K_n (r_i - r_i^d) - K_v (v_i - v_i^d)
\end{bmatrix} \\
\frac{1}{2} \Omega(\omega_i) q_i \\
-\omega_i \times J_i \omega_i + J_i \omega_i^d + \frac{1}{2} \omega_i \times J_i (\omega_i + \omega_i^d) + k_q q_{e_i} - K_{\alpha_i} (\omega_i - \omega_i^d)
\end{bmatrix}
\]
Note $K_n, K_v, K_{q_k}$ are the gain matrices which are symmetric positive definite and $k_p, k_q, k_n, k_qi, k_{qi}$ are positive gain constants. The reader is referred to [11] for a Lyapunov-based proof of global asymptotic stability of the closed-loop systems (3.12) and (3.13).

3.5 Formation Feedback

So far we have not included a formation feedback from the satellites to the virtual structure. There is a big drawback when formation feedback is not included in the control laws. Consider a scenario where the virtual structure evolves too fast. We know that, in practice, the control forces and torques are bounded. Thus the control laws could reach a saturation limit due to which the satellite may not be able to track the desired trajectory, thereby leaving the formation. The only way to avoid this problem without including formation feedback would be to slow down the virtual structure so that the satellites can track their trajectory accurately. This slows down the formation maneuver which is not practical, besides there is no probability of getting the desired end formation as it does not take into account perturbations due to internal or external factors.

If one of the satellites were to fail due to some external disturbance or some internal mechanical or electrical malfunction then this satellite will be left behind. The other satellites move on but since the virtual structure act as a rigid body, without feedback from this failed satellite the rest of the system cannot get into the final desired formation. This would lead to the failure of the entire formation. To avoid such a possibility we need to include a group feedback from the satellites making up the formation to the virtual structure so that formation comes to a stop until the failed satellite recovers from its
malfunction or otherwise after some critical time the entire virtual structure is redesigned to accommodate for the failed satellite and move on. The idea of formation feedback was introduced in [1], [6], [9], and [17]. In [6] this idea was applied to control the multi-agent coordinated behavior of a class of robots. The same idea has been extended here with some modifications to include spacecraft dynamics.

3.6 Formation Feedback Control Law

The theoretical aspect behind the spacecraft formation feedback has been first introduced in paper [11]. Before we include formation feedback we need to define a performance measure metric $E(X, X^d)$. This metric in theory has to include two parts. The first part should include spacecraft tracking error. The second part has to include a formation keeping error. However, in our design of control laws we have included only the second part. The reason for this modification was even though theoretically we need both the parts for effective feedback the programming of this architecture turns out to be pretty complicated besides taking considerable runtime. We redesign the performance measure metric $E(X, X^d)$ as follows

$$E(X, X^d) = \left\| X_i(t) - X^d_i(t) \right\|$$

(3.14)

$$\left\| X_i(t) - X^d_i(t) \right\| = \sqrt{\sum_{i=1}^{n-\text{satellites}} (X_i(t) - X^d_i(t))^2}$$

(3.15)

where $X_i = r_i, v_i,$ or $\omega_i$. 
We can now include the formation feedback from individual satellites to virtual structure via gain matrices $\Gamma_v$ and $\Gamma_\omega$, which are chosen as follows [11]:

$$
\Gamma_v = K_v + K_{\omega_v} E_v(X, X^d)^2 \\
\Gamma_\omega = K_\omega + K_{\omega_\omega} E_\omega(X, X^d)^2
$$

(3.16)

where $K_v, K_\omega, K_{\omega_v}$, and $K_{\omega_\omega}$ are positive definite matrices. We now integrate these modifications into our control laws for the virtual structure in order to include formation feedback and obtain the following closed loop dynamics for the virtual structure [11]:

$$
\begin{bmatrix}
\dot{r}_F \\
\dot{v}_F \\
\dot{q}_F \\
J_F \ddot{\omega}_F
\end{bmatrix}_{\omega} =
\begin{bmatrix}
v_F \\
-K_r (r_F - r_F^d) - k_p \|r_F - r_F^d\|^2 v_F - \Gamma_v (X, X^d) v_F \\
\frac{1}{2} \Omega(\omega_F) q_F \\
-\omega_F \times J_F \omega_F + k_q q_{\omega F} - k_q \|q_F - q_{\omega F}^d\|^2 \omega_F - \Gamma_\omega (X, X^d) \omega_F
\end{bmatrix}
$$

(3.17)

The control laws for the individual satellites remain the same and thus the closed loop dynamics for the individual satellites is given by (3.13).

### 3.7 Expansion/Contraction Maneuvers

Group maneuvers with our control laws can be achieved as a succession of elementary formation maneuvers. Our control law so far includes translational and rotational maneuvers. Since the virtual structure acts a single rigid body we would like to include an
expansion/contraction maneuver for the structure as whole. To do this we first include an expansion vector \( \xi_F(t) = [\xi_1(t), \xi_2(t), \xi_3(t)]^T \) and define the following diagonal matrix:

\[
\Xi(t) = \begin{bmatrix}
\xi_1(t) & 0 & 0 \\
0 & \xi_2(t) & 0 \\
0 & 0 & \xi_3(t)
\end{bmatrix}
\] (3.18)

In this thesis we will assume \( \xi_F \) is constant so that the closed loop translational and rotational dynamics for the virtual structure including this parameter can be expressed as [11]:

\[
\begin{bmatrix}
\dot{r}_F \\
\dot{v}_F \\
\dot{q}_F \\
\dot{\omega}_F \\
\dot{\xi}_F
\end{bmatrix} = \begin{bmatrix}
\dot{r}_F \\
-K_r (r_F - r_F^d) - k_p \|r_F - r_F^d\|^2 v_F - \Gamma_v (X, X^d) v_F \\
\frac{1}{2} \Omega (\omega_F) q_F \\
-\omega_F \times J_F \omega_F + k_q q_e F - k_a \|q_F - q^d\|^2 \omega_F - \Gamma_\omega (X, X^d) \omega_F \\
0
\end{bmatrix}
\] (3.19)

Note that the desired position and velocity of each individual satellite is now going to change because of the expansion maneuver. The desired position and velocity have to be calculated at each point in time and then integrated into the control law for the individual satellite as described below

\[
\begin{bmatrix}
r^d_i(t) \\
v^d_i(t)
\end{bmatrix} = \begin{bmatrix}
[r_F(t)]_o + C_{of}(t) \Xi(t) [r^d_F]_F \\
[v_F(t)]_o + C_{of}(t) \Xi(t) [r^d_F]_F + (\omega_F)_o \times \left( C_{of}(t) \Xi(t) [r^d_F]_F \right)
\end{bmatrix}
\] (3.20) (3.21)

The closed loop motion for each satellite would be the same as before [11], i.e.
\[
\begin{bmatrix}
\dot{r}_i \\
\dot{\psi}_i \\
\dot{q}_i \\
J_i \dot{\omega}_i
\end{bmatrix} = \begin{bmatrix}
v_i \\
\dot{v}_i^d - K_n (r_i - r_i^d) - K_n (v_i - v_i^d) \\
\frac{1}{2} \Omega(\omega_i)q_i \\
-\omega_i \times J_i \omega_i + J_i \omega_i^d + \frac{1}{2} \omega_i \times J_i (\omega_i + \omega_i^d) + k_{q}q_{ei} - K_{\omega} (\omega_i - \omega_i^d)
\end{bmatrix}
\]
(3.22)
CHAPTER IV

4 CONTROL SIMULATION

4.1 Second Order Closed Loop System

With the use of formation feedback from the satellites to the virtual structure, our system takes form of the following closed loop system.

Consider a second order characteristic equation of the form \( s^2 + 2\zeta\omega_n s + \omega_n^2 = 0 \), where \( \zeta \) is the damping ratio and \( \omega_n \) is the natural frequency of the oscillation. The gains in our virtual structure are designed such that

\[
\begin{align*}
K_v &= 2\zeta_1 \omega_{n1} I, \quad K_\omega = 2\zeta_2 \omega_{n2} I, \quad K_{v1} = 2\zeta_3 \omega_{n3} I, \quad K_{\omega1} = 2\zeta_4 \omega_{n4} I, \\
K_r &= \omega_{n1}^2 I, \quad K_\xi = \omega_{n2}^2 I, \quad K_{ri} = \omega_{n3}^2 I, \quad k_{qi} = \omega_{n4}^2, \quad k_q = \omega_{n2}^2
\end{align*}
\]  

(4.1)
It is very important for us to select appropriate values for the gains. The convergence of the virtual structure to the final formation depends on the selection of the gains used in the control laws. For a desirable transient response, the damping ratio must be between 0.4 and 0.8. The damping ratio values that are less than 0.4 yield excessive overshoot in the transient response, and a system with damping ratios larger than 0.8 responds sluggishly. It must be noted that a low damping ratio and high natural frequency would mean large oscillations in transient response during the early stages of formation maneuvers. As the satellites are traveling at very high speeds these oscillations could mean a collision between satellites forming the virtual structure, which should be avoided. Improper selection of the gains would mean that the satellites making up the virtual structure never come into the desired formation.

4.2 Matlab Simulations

In our simulations we consider a group of six satellites and one imaginary virtual satellite. The place holder for the virtual satellite is placed at the center of mass of the virtual structure. The virtual structure is required to trace an equatorial orbit around the
Earth. In the simulation the six satellites begin from an arbitrary position close to the virtual structure and then come into formation in the shape of a hexagon. They perform a combination of translational, rotational and expansion/formation maneuvers starting from rest with a set of initial positions and attitudes.

![Figure 5: Final desired formation of virtual structure](image)

Even though the translational and rotational controls are not coupled they are not completely independent of each other. The dynamics of the virtual structure and that of the individual satellites are interlinked with one another. This creates some complications in the programming. To begin with we first define the dynamics of the virtual structure at a particular point in time. This then has to be translated to the required dynamics of each individual satellite. We first derive the tracking control for each individual satellite based on the desired dynamics. We then include a formation feedback based on these tracking controls to virtual structure so that the state of the virtual structure can be computed for the next point in time. Then the whole process is repeated all over again until the desired formation is achieved.
The calculation of the gain values is an important factor as it plays a very important role in achieving the final desired formation. We begin by first using a simplified version of the control laws as explained in (3.5)-(3.9), where formation feedback is not included. We start by using arbitrary gain values. After this we include a formation feedback as described by (3.17) where the updated gain matrices are used. To demonstrate the effectiveness of the control laws in different scenarios we consider the following three simple cases:

1) Formation maneuvers with $\omega_f = 0$

2) Formation maneuvers with $\omega_f = 0.5 \text{ rad/min}$

3) Formation maneuvers with $\omega_f = 0.5 \text{ rad/min}$ and expansion/contraction of

\[ \xi_1 = \xi_2 = \xi_3 = 0.5 \]

The gain matrices used in the simulations are given in Table 1. The physical parameters for the satellites are listed in Table 2.
Table 1: Control gains and physical parameters

<table>
<thead>
<tr>
<th>Gains</th>
<th>Values used in Simulation</th>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$K_r$</td>
<td>3 $I$</td>
<td>$M_f$</td>
<td>1 kg</td>
</tr>
<tr>
<td>$K_v$</td>
<td>0.25 $I$</td>
<td>$M_i$</td>
<td>150 kg</td>
</tr>
<tr>
<td>$K_\omega$</td>
<td>0.32 $I$</td>
<td>$J_f$</td>
<td>$I$ kg m$^2$</td>
</tr>
<tr>
<td>$K_\xi$</td>
<td>5 $I$</td>
<td>$J_i$</td>
<td>25 $I$ kg m$^2$</td>
</tr>
<tr>
<td>$K_r$</td>
<td>2 $I$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$K_v$</td>
<td>1 $I$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$k_q$</td>
<td>3.24</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$k_q$</td>
<td>9.35</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

4.3 Matlab Results

To illustrate the effectiveness of the previously discussed control schemes, we simulate the closed loop response. As discussed earlier the virtual structure is assumed to be in an equatorial orbit. The desired translational motion for the virtual structure is given as

$$r_F^d(t) = \begin{bmatrix} R_F \cos(nt) \\ R_F \sin(nt) \\ 0 \end{bmatrix}$$  \hspace{1cm} (4.2)

where $R_F$ is the orbit radius and $n$ is the orbital rate. The desired quaternion trajectory is given by
where $\omega_F$ is the angular velocity of the virtual frame.

Table 3 shows the initial positions and velocities of the satellites. The initial quaternions are given in Table 4. The desired positions and quaternions of the satellites with respect to the virtual frame are given in Tables 5 and 6, respectively. The desired positions correspond the hexagonal formation described in Figure 5.
### Table 3: Initial quaternions

<table>
<thead>
<tr>
<th>Satellite</th>
<th>Quaternion</th>
</tr>
</thead>
<tbody>
<tr>
<td>Virtual satellite: ( q_{\text{Fi}} )</td>
<td>((0, 0, 0, 1)^T)</td>
</tr>
<tr>
<td>Satellite 1: ( q_{\text{1i}} )</td>
<td>((0, 0, \cos(\pi/6), \sin(\pi/6))^T)</td>
</tr>
<tr>
<td>Satellite 2: ( q_{\text{2i}} )</td>
<td>((0, \cos(\pi/6), 0, \sin(\pi/6))^T)</td>
</tr>
<tr>
<td>Satellite 3: ( q_{\text{3i}} )</td>
<td>((\cos(\pi/6), 0, 0, \sin(\pi/6))^T)</td>
</tr>
<tr>
<td>Satellite 4: ( q_{\text{4i}} )</td>
<td>((0, 0, \sin(\pi/6), \cos(\pi/6))^T)</td>
</tr>
<tr>
<td>Satellite 5: ( q_{\text{5i}} )</td>
<td>((0, \sin(\pi/6), 0, \cos(\pi/6))^T)</td>
</tr>
<tr>
<td>Satellite 6: ( q_{\text{6i}} )</td>
<td>((\sin(\pi/6), 0, 0, \cos(\pi/6))^T)</td>
</tr>
</tbody>
</table>

### Table 4: Desired positions \((r = 0.5 \text{ km})\)

<table>
<thead>
<tr>
<th>Satellite</th>
<th>Desired Position</th>
</tr>
</thead>
<tbody>
<tr>
<td>Satellite 1: ( r_{\text{1F}}^d )</td>
<td>((r, 0, 0)^T \text{ km})</td>
</tr>
<tr>
<td>Satellite 2: ( r_{\text{2F}}^d )</td>
<td>((r \cos(\pi/3), r \sin(\pi/3), 0)^T \text{ km})</td>
</tr>
<tr>
<td>Satellite 3: ( r_{\text{3F}}^d )</td>
<td>((-r \cos(\pi/3), r \sin(\pi/3), 0)^T \text{ km})</td>
</tr>
<tr>
<td>Satellite 4: ( r_{\text{4F}}^d )</td>
<td>((-r, 0, 0)^T \text{ km})</td>
</tr>
<tr>
<td>Satellite 5: ( r_{\text{5F}}^d )</td>
<td>((-r \cos(\pi/3), -r \sin(\pi/3), 0)^T \text{ km})</td>
</tr>
<tr>
<td>Satellite 6: ( r_{\text{6F}}^d )</td>
<td>((r \cos(\pi/3), -r \sin(\pi/3), 0)^T \text{ km})</td>
</tr>
</tbody>
</table>

### Table 5: Desired quaternions

<table>
<thead>
<tr>
<th>Satellite</th>
<th>Desired Quaternion</th>
</tr>
</thead>
<tbody>
<tr>
<td>Satellite 1: ( q_{\text{1i}}^d )</td>
<td>((0, 0, 0, 1)^T)</td>
</tr>
<tr>
<td>Satellite 2: ( q_{\text{2i}}^d )</td>
<td>((0, 0, 0, 1)^T)</td>
</tr>
<tr>
<td>Satellite 3: ( q_{\text{3i}}^d )</td>
<td>((0, 0, 0, 1)^T)</td>
</tr>
<tr>
<td>Satellite 4: ( q_{\text{4i}}^d )</td>
<td>((0, 0, 0, 1)^T)</td>
</tr>
<tr>
<td>Satellite 5: ( q_{\text{5i}}^d )</td>
<td>((0, 0, 0, 1)^T)</td>
</tr>
<tr>
<td>Satellite 6: ( q_{\text{6i}}^d )</td>
<td>((0, 0, 0, 1)^T)</td>
</tr>
</tbody>
</table>
Case 1: A formation maneuver with $\omega_F = 0$:

In this case, the desired quaternion vector for the virtual structure is taken as

$$q^d_F = \begin{bmatrix} \bar{e} \sin(\pi / 4) \\ \cos(\pi / 4) \end{bmatrix}, \quad \bar{e} = \begin{bmatrix} 1 / \sqrt{14} \\ 2 / \sqrt{14} \\ 3 / \sqrt{14} \end{bmatrix}$$

Figures 6-11 show the simulation results. It can be observed that all the satellites converge to the desired positions and attitudes starting from the initial conditions given by Tables 1 and 2. The graphs show that the formation maneuver starts out with some oscillations which die out within 30 minutes.

Case 2: A formation maneuver with $\omega_F = 0.5$ rad/min:

In this case, the desired quaternion trajectory is given by

$$q^d_F = \begin{bmatrix} 0 \\ 0 \\ \sin(0.25r) \\ \cos(0.25r) \end{bmatrix}$$

Figures 12-17 show the simulation results. It can be observed that all the satellites converge to the desired positions and attitudes starting from the initial conditions given by Tables 1 and 2. As expected, the angular velocities of all the satellites converge to 0.5 rad/min.

Case 3: A formation maneuver with expansion/contraction and $\omega_F = 0.5$ rad/min:

Again, in this case, the desired quaternion trajectory is given by
\[ q_F^d = \begin{bmatrix}
0 \\
0 \\
sin(0.25t) \\
cos(0.25t)
\end{bmatrix} \]

and the expansion/contraction matrix is taken as \( \xi_F = [0.5 \ 0.5 \ 0.5]^T \). Figures 18-23 show the simulation results. It can be observed that all the satellites converge to the desired positions and attitudes starting from the initial conditions given by Tables 1 and 2. As expected, the angular velocities of all the satellites converge to 0.5 rad/min. Due to the contraction factor of 0.5, the satellites are driven to locations at 0.25 km from the center of the hexagon.
Figure 6: Positions of the 1st, 2nd, and 3rd satellites in the virtual frame (Case 1)
Figure 7: Positions of the 4th, 5th, and 6th satellites in the virtual frame (Case 1)
Figure 8: Quaternions of the satellites with respect to the earth frame (Case 1)
Figure 9: Quaternions of the satellites with respect to the virtual frame (Case 1)
Figure 10: Quaternions of the virtual frame with respect to the earth frame (Case 1)

Figure 11: Angular velocities of the satellites relative to the virtual structure (Case 1)
Figure 12: Positions of the 1st, 2nd, and 3rd satellites in the virtual frame (Case 2)
Figure 13: Positions of the 4th, 5th, and 6th satellites in the virtual frame (Case 2)
Figure 14: Quaternions of the virtual frame with respect to the earth frame (Case 2)
Figure 15: Quaternions of the satellites with respect to the earth frame (Case 2)
Figure 16: Quaternions of the satellites with respect to the virtual frame (Case 2)
Figure 17: Angular velocities of the satellites and virtual structure with respect to the earth (Case 2)
Figure 18: Positions of the 1st, 2nd, and 3rd satellites in the virtual frame (Case 3)
Figure 19: Positions of the 4th, 5th, and 6th satellites in the virtual frame (Case 3)
Figure 20: Quaternions of the virtual frame with respect to the earth (Case 3)
Figure 21: Quaternions of the satellites with respect to the earth frame (Case 3)
Figure 22: Quaternions of the satellites with respect to the virtual structure (Case 3)
Figure 23: Angular velocities of the satellites and the virtual frame with respect to the earth (Case 3)
4.4 Simulations Using STK (SATELLITE TOOL KIT)

Matlab simulations provide a basis for testing the control laws and show us that our controls work. Even though Matlab simulations give us an insight into the dynamics of the virtual structure they do not give us sufficient information regarding the transient dynamics of the formation other than telling us that the formation converges to the desired states. To study the transient dynamics especially the oscillations during formation maneuvers we need to study in a 3D environment. To do this we use the STK simulation software. We export the data generated by Matlab into STK. To do this we need to first convert the data into a format recognized by STK. We generate two sets of files called ephemeris and attitude files and then import them to STK.

4.4.1 Ephemeris File Format (*.e)

An ephemeris file is an ASCII text file formatted for compatibility with STK that ends in a *.e extension. Ephemeris files are used when you need to provide STK with position and velocity data for a spacecraft to model a scenario. The ephemeris data, organized in a proper format, can be imported into STK using the StkExternal propagator. The data is formatted in the Cartesian X, Y, Z coordinates. The frame of reference used is the Geocentric Equatorial Frame.

Each ephemeris table, regardless of the type of the ephemeris data in it, contains some common elements called keywords. Without these keywords STK does not recognize the data. Some keywords are taken by default while the others need to be given. The table
below explains the function of some of these important keywords. (This information was
provided by AGI which enabled in constructing the Ephemeris file).

Table 6: Ephemeris file keywords

<table>
<thead>
<tr>
<th>S.No</th>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>stk.v.&lt;Version No.&gt;</td>
<td>The version of STK software for which the file is formatted to be used. Files can be created in, and imported to, STK software versions consistent with the version used or higher.</td>
</tr>
<tr>
<td></td>
<td>NOTE:</td>
<td>The every ephemeris file should start with this data. <em>Example: stk.v.7.0</em></td>
</tr>
<tr>
<td>1</td>
<td>BEGIN Ephemeris END Ephemeris</td>
<td>Sets off the beginning and end of the ephemeris table, including all other keyword phrases and data point specification. The data that has to be fed into STK must be between these two keywords. The reference epoch time for the time values of the ephemeris data in the table. Specify the scenario epoch using Gregorian UTC time (dd mmm yyyy hh:mm:ss.s). There is no relationship between the scenario epoch specified in the ephemeris table and the actual scenario epoch in your STK scenario.</td>
</tr>
<tr>
<td>2</td>
<td>ScenarioEpoch</td>
<td><em>Example: ScenarioEpoch 31 Mar 2007 00:00:00.0</em></td>
</tr>
<tr>
<td></td>
<td></td>
<td>The maximum number of ephemeris points to be read.</td>
</tr>
<tr>
<td>4</td>
<td>NumberOfEphemerisPoints</td>
<td><em>Example: NumberOfEphemerisPoints 1000</em></td>
</tr>
<tr>
<td></td>
<td></td>
<td>1000 ephemeris points would be read.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The central body around which the spacecraft orbit. The keyword value that completes the phrase can be the name of any registered central body. Registered central bodies can be found in the STKData/Central Bodies directory. The default value is Earth.</td>
</tr>
<tr>
<td>5</td>
<td>Central Body</td>
<td><em>Example: CentralBody Earth</em></td>
</tr>
</tbody>
</table>

51
There are many different types of Ephemeris format. The format we used in this thesis was EphemerisTimePos which means the time and position of the spacecraft needs to be provided.

Individual data points following the EphemerisTimePosVel keyword look like this:

\(<\text{TimeInSeconds}>\ <X> \ <Y> \ <Z>\) where,

\(<\text{TimeInSeconds}>\)

\(<X> \ <Y> \ <Z>\)

The time value of the point in seconds (in the format \(xxxx.xxxx\)) relative to the epoch as defined by the ScenarioEpoch keyword The vehicle position

4.4.2 Attitude File Format (*.a)

An Attitude file like the ephemeris file is an ASCII text file formatted for compatibility with STK and ends in a *.a extension. Attitude files can be useful when you need to provide STK with data for the rotational kinematics. Spacecraft attitude is a basic property of all space vehicles used in STK. The attitude data in any properly formatted Attitude file can be imported into STK using the pre-computed option or in our case an StkExternal propagator.

Attitude data represents the orientation of the vehicle’s body frame relative to Earth’s Geocentric Equatorial Frame or any other coordinate axes used for that matter. It should be noted that STK interpolates the data in the same coordinate frame that the data is supplied in.
NOTE: Attitude data must represent a transformation from the reference coordinate frame to the vehicle body frame (in our case from the Earth’s inertial frame to the virtual frame). The data in an attitude file can be given as a set of quaternions, Euler angles; yaw, pitch, and roll angles; direction cosine matrices, etc. In this thesis the attitude data are given as quaternions.

The list of keyword used in making up the attitude file is given below.

Table 7: Quaternion file keywords

<table>
<thead>
<tr>
<th>S.No</th>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>stk.v.&lt;Version No.&gt;</td>
<td>The version of STK software for which the file is formatted to be used.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Every attitude file should start with this data.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><em>Example: stk.v.7.0</em></td>
</tr>
<tr>
<td>2</td>
<td>BEGIN Attitude</td>
<td>Sets off the beginning and end of the attitude table.</td>
</tr>
<tr>
<td></td>
<td>END Attitude</td>
<td>Like before The data that has to be feed into STK must be between these two keywords.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The reference epoch time for the time values of the attitude data in the table. Uses Gregorian UTC time (dd mmm yyyy hh:mm:ss.s).</td>
</tr>
<tr>
<td>3</td>
<td>Scenario Epoch</td>
<td><em>Example: ScenarioEpoch 31 Mar 2007 00:00:00.0</em></td>
</tr>
<tr>
<td></td>
<td></td>
<td>The maximum number of attitude points to be read.</td>
</tr>
<tr>
<td>4</td>
<td>NumberOfAttitudePoints</td>
<td><em>Example: NumberOfAttitudePoints 1000</em></td>
</tr>
<tr>
<td></td>
<td></td>
<td>1000 attitude points would be read.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The central body around which the spacecraft orbit.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The default value is Earth.</td>
</tr>
<tr>
<td>5</td>
<td>CentralBody</td>
<td><em>Example: CentralBody Earth</em></td>
</tr>
</tbody>
</table>
The **AttitudeTimeQuaternions** format is designed to import the vehicle attitude data provided as quaternions. The STK convention for quaternions is that the 4th element of the quaternion is the scalar component. The quaternions represent the rotation from the Earth’s coordinate frame to the virtual structure’s coordinate frame.

Individual data points following the **AttitudeTimeQuaternions** keyword look like this:

```
<TimeInSeconds> <q1> <q2> <q3> <q4>
```

- `<TimeInSeconds>`: The time value of the point in seconds (in the format xxxx.xxx) relative to the epoch as defined by the **ScenarioEpoch** keyword.
- `<q1> <q2> <q3> <q4>`: The four elements of the quaternion.

Keywords used in the Ephemeris files or the Attitude files are NOT case-sensitive. The capitalization used is solely for the purpose of readability.

The following conventions must be observed when specifying data points in either of the two formats:

- Each line should contain only one data point.
- The values on each line must be separated by at least one space.
- The lines need to be listed in ascending order in time but do not have to be evenly spaced in time.
- One cannot have multiple points at the same time.
- There must be at least as many points as specified by the keyword.
- If the attitude is very dynamic (e.g., spinning or slewing rapidly), you should specify more points during the period of rapid motion. For a spinning spacecraft, you should have at least 3 or 4 points per revolution in order to create an accurate representation of the spinning motion within the Attitude file.
Once the two files are generated in the format explained above we transfer all the ephemeris and attitude data into STK using StkExternal propagator. A movie is generated using STK’s Soft VTR application.

4.4.3 STK Animations

Once the 3D animation of the translational dynamics is visualized, many issues concerning the formation maneuvers can be observed. The animations show that using the coupled dynamics between the individual satellites and the virtual structure the formation achieves a reasonable convergence speed. However, this convergence speed depends greatly on the gain matrices used in the control laws. Different formations are visualized to study the translational dynamics during formation switching. Some of the formations realized by virtual structure are shown below.
Figure 25: Hexagonal formation (2\textsuperscript{nd} type)

Figure 26: Formation switching - from hexagon to straight line

Figure 27: Triangular formation
Many reshaping scenarios are studied with both zero and nonzero $\omega_f$. It is observed that during the maneuvers the tracking errors for each satellite are driven to zero, which means that the desired formation is realized. One important observation is that a system with formation feedback preserves the formation much better than a system without a formation feedback. However a system that uses formation feedback takes longer time to converge to the final formation.
CHAPTER V

5 CONCLUSION

5.1 Summary

We have first summarized the translational and rotational dynamics of a single spacecraft. We have then introduced the concept of group behavior by making these individual satellites trace the path of an imaginary virtual satellite. This unidirectional information flow from the virtual satellite to the individual satellite leads to the generation of control laws without formation feedback. To include formation feedback we have introduced the concept of tracking error function and formation maintenance error function for the group of satellites making up the formation. We have demonstrated that, by feeding this information back to the virtual satellite, the dynamics of virtual structure and the individual satellites can be coupled so that they behave like a single rigid body. We have then modified the control laws for the virtual structure to perform expansion/contraction maneuvers.

In this thesis we have considered a variety of issues that arise during the coordination of formation maneuvers. We have shown how to select the appropriate control parameters to increase the convergence speed of the formation. We have run several Matlab simulations to test the feedback laws with these control parameters and have generated 3D animations using STK to visualize the transient behavior of the virtual structure.
5.2 Recommended Future Work

We have dealt with a variety of issues related to formation control in this thesis. As our main interest was in coordination and control of multiple spacecraft we have not dealt with issues regarding fuel economy. Most of the complicated formation maneuvers we have simulated would in practice require large fuel expenditure. Since the dynamics of the satellites are coupled with one another there is no stable orbit until the formation is realized. This would mean maintaining the orbit during translational dynamics would be at the cost of fuel expenditure. This is especially true during formation switching. This unfortunately cannot be avoided but can be minimized to a great extent by selecting the control parameters using an optimization scheme. We have also not discussed how to incorporate the actuator dynamics in the design of feedback control laws. The abovementioned issues are subjects of our future research in the area of spacecraft formation control.
REFERENCES


APPENDIX A

MATLAB SIMULATION CODE

% 1) Run this program for two different sets of initial value of quaternions
% 2) Run this program for different values of expansion and contraction
% 3) Run the program with some a certain angular velocity (U will then require
% to change the initial value of quaternion. Remove qfd from f1 if it does not
% have a certain angular velocity

clear
clc
format long
i3=[1 0 0; 0 1 0; 0 0 1];
h=0.1;% (time step 0.1 min)
global Ji Kri Kvi Kwi Kr kq Ketha kp ka ke mu;
Ji=25*i3;
% Kri=0.81*i3;
% Kvi=1.27*i3;
Kri=2*i3;% units:-min^-2;
Kvi=1*i3;% units:-min^-1;

kqi=3.24;% units:-min^-2;
Kwi=6.15*i3;% units:-min^-1;
Jf=i3;
% Kr=0.03*i3;
Kr=3*i3;% units:-min^-1;

% kq=0.05;
kq=9.35;% units:-min^-2;
% Ketha=0.03*i3;
Ketha=5*i3;% units:-min^-1;

kp=0;
ka=0;
ke=0;
global rfd qfd ethafd r1fd r2fd r3fd r4fd r5fd r6fd q1fd q2fd q3fd q4fd q5fd q6fd nn;
nn=0;% rad/min
% The desired position of virtual frame
Rf=11000e3;% m
% Rf=1000;
% orbital parameters
mu=398600e9*3600;
% Virtual structure orbital period
\[ T = \frac{2\pi}{\sqrt{\mu}} R f^{(3/2)}; \]
% Orbital rate
\[ n = \frac{2\pi}{T}; \% \text{units: rad/min}; \]

\[ e = \begin{bmatrix} 1/\sqrt{14} \\ 2/\sqrt{14} \\ 3/\sqrt{14} \end{bmatrix}; \]
\[ q_{fd} = \begin{bmatrix} e \sin(\pi/4) \\ \cos(\pi/4) \end{bmatrix}; \]
% \[ q_{fd} = [0; \]
% \[ 0; \]
% \[ 0; \]
% \[ 1]; \]
\[ \text{ethafd} = [1; \]
\[ 1; \]
\[ 1]; \]
% \[ \text{ethafd} = [2; \]
% \[ 2; \]
% \[ 2]; \]
% \[ \text{ethafd} = [0.5; \]
% \[ 0.5; \]
% \[ 0.5]; \]
\[ r = 100; \% \text{m} \]
\[ r1fd = [r; \]
\[ 0; \]
\[ 0]; \% \text{m} \]
\[ r2fd = [r \cos(\pi/3); \]
\[ r \sin(\pi/3); \]
\[ 0]; \% \text{m} \]
\[ r3fd = [-r \cos(\pi/3); \]
\[ r \sin(\pi/3); \]
\[ 0]; \% \text{m} \]
\[ r4fd = [-r; \]
\[ 0; \]
\[ 0]; \% \text{m} \]
\[ r5fd = [-r \cos(\pi/3); \]
\[ -r \sin(\pi/3); \]
\[ 0]; \% \text{m} \]
\[ r6fd = [r \cos(\pi/3); \]
\[ -r \sin(\pi/3); \]
\[ 0]; \% \text{m} \]

\[ q1fd = [0; \]
\[ 0; \]
\[ 0; \]
\[ 1]; \]
\[ q2fd = [0; \]
\[ 0; \]
q3fd=[0; 0; 0; 1];
q4fd=[0; 0; 0; 1];
q5fd=[0; 0; 0; 1];
q6fd=[0; 0; 0; 1];

% Virtual structure approach without formation feedback
% the call for the function f1 is made here
% the initial conditions for the virtual structure is given below
rfi=[Rf; 0; 0];
vfi=[0; 6.31347764706584e5; 0];% m/min
qfi=[0; 0; 0; 1];

% Wfi has a definite angular velocity. This will change the value of desired
% quaternion
wfi=[0; 0; 0; nn];

% wfi=[0; 0; 0];
% ethafi=[0; 0; 0];
dethafi=[0; 0];
%the initial conditions for the first satellite
r1i=[Rf-r;
  0;
  0];
v1i=[0;
  6.31347764706584e5;
  0];%m/min
q1i=[0;
  0;
  0;
  1];
% q1i=[e*\sin(pi/4);
% \cos(pi/4)];
w1i=[0;
  0;
  0];

%the initial conditions for the second satellite
r2i=[0;
  Rf-r;
  0];
v2i=[0;
  6.31347764706584e5;
  0];%m/min
q2i=[0;
  0;
  0;
  1];
% q2i=[e*\sin(pi/4);
% \cos(pi/4)];
w2i=[0;
  0;
  0];

%the initial conditions for the third satellite
r3i=[0;
  0;
  Rf-r];
v3i=[0;
  6.31347764706584e5;
  0];%m/min
q3i=[0;
  0;
\begin{align*}
\mathbf{r}_3 &= \begin{bmatrix} R_f + r \\ 0 \\ 0 \end{bmatrix}; \\
\mathbf{v}_3 &= \begin{bmatrix} 0 \\ 6.31347764706584 \times 10^5 \\ 0 \end{bmatrix}; \\
\mathbf{q}_3 &= \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}; \\
\mathbf{w}_3 &= \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix};
\end{align*}

and the initial conditions for the fourth satellite
\begin{align*}
\mathbf{r}_4 &= \begin{bmatrix} R_f + r \\ 0 \\ 0 \end{bmatrix}; \\
\mathbf{v}_4 &= \begin{bmatrix} 0 \\ 6.31347764706584 \times 10^5 \\ 0 \end{bmatrix}; \\
\mathbf{q}_4 &= \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}; \\
\mathbf{w}_4 &= \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix};
\end{align*}

and the initial conditions for the fifth satellite
\begin{align*}
\mathbf{r}_5 &= \begin{bmatrix} 0 \\ R_f + r \\ 0 \end{bmatrix}; \\
\mathbf{v}_5 &= \begin{bmatrix} 0 \\ 6.31347764706584 \times 10^5 \\ 0 \end{bmatrix}; \\
\mathbf{q}_5 &= \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}; \\
\mathbf{w}_5 &= \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix};
\end{align*}

and the initial conditions for the sixth satellite
\begin{align*}
\mathbf{r}_6 &= \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix};
\end{align*}
Rf+r];
v6i=[0;
    6.31347764706584e5;
    0];%m/min
q6i=[0;
    0;
    0;
    1];
% q1i=[e*sin(pi/4);
% cos(pi/4)];
w6i=[0;
    0;
    0];

xi=[rfi;
    vfi;
    qfi;
    wfi;
    ethafi;
    dethafi;
    r1i;
    v1i;
    q1i;
    w1i;
    r2i;
    v2i;
    q2i;
    w2i;
    r3i;
    v3i;
    q3i;
    w3i;
    r4i;
    v4i;
    q4i;
    w4i;
    r5i;
    v5i;
    q5i;
    w5i;
    r6i;
    v6i;
    q6i;
    w6i];
z=1;
for t=0:h:100
    [Y]=RK4('f1',t,xi,1);
end
$X(:, z) = Y$;
$xi = Y$;
$z = z + 1$;
end

t = 0:h:100;
X = X';

% [t, X] = ode45('tf1', [0:1:60], xi);
% Error in rf is given by
for i = 1:length(t)
    rfd = [Rf*cos(n*t(i));
          Rf*sin(n*t(i));
          0];
    rfe(i) = norm([(X(i, 1) - rfd(1));
                   (X(i, 2) - rfd(2));
                   (X(i, 3) - rfd(3))]);
end

% The quaternion of the virtual frame w.r.t time
qf(:, 1) = X(:, 7);
qf(:, 2) = X(:, 8);
qf(:, 3) = X(:, 9);
qf(:, 4) = X(:, 10);
figure(1)
plot(t, qf(:, 1), t, qf(:, 2), t, qf(:, 3), t, qf(:, 4))
legend('qf1', 'qf2', 'qf3', 'qf4');
ylabel('qf------------------------>');
xlabel('Time t in seconds---------->');
title('Quaternion of virtual frame w.r.t inertial frame’s time');
grid on
for i = 1:3
    % Defines the position of virtual frame and satellite 1,2,3,4,5,6 w.r.t Inertial % frame
    rf(:, i) = X(:, i);
    r1(:, i) = X(:, 19 + i);
    r2(:, i) = X(:, 32 + i);
    r3(:, i) = X(:, 45 + i);
    r4(:, i) = X(:, 45 + 13 + i);
    r5(:, i) = X(:, 45 + 26 + i);
    r6(:, i) = X(:, 45 + 39 + i);
end

% Below we find the position of the three satellites w.r.t virtual frame
% the Cof matrix is given by
for i = 1:length(X)
    cof = [1 - 2*(qf(i, 2)^2 + qf(i, 3)^2) 2*(qf(i, 1)*qf(i, 2) + qf(i, 3)*qf(i, 4)) 2*(qf(i, 1)*qf(i, 3) - qf(i, 2)*qf(i, 4));
          ];
end
\[2*(qf(i,2)*qf(i,1)-qf(i,3)*qf(i,4)) - 1 - 2*(qf(i,1)^2+qf(i,3)^2)\]
\[2*(qf(i,2)*qf(i,3)+qf(i,1)*qf(i,4));\]
\[2*(qf(i,3)*qf(i,1)+qf(i,2)*qf(i,4)) - 2*(qf(i,3)*qf(i,2)-qf(i,1)*qf(i,4)) - 1 - 2*(qf(i,1)^2+qf(i,2)^2)];\]
\[rlf(i,:) = inv(cof)*([rl(i,1)-rf(i,1);
  rl(i,2)-rf(i,2);
  rl(i,3)-rf(i,3)]);
\]
\[r2f(i,:) = inv(cof)*([r2(i,1)-rf(i,1);
  r2(i,2)-rf(i,2);
  r2(i,3)-rf(i,3)]);
\]
\[r3f(i,:) = inv(cof)*([r3(i,1)-rf(i,1);
  r3(i,2)-rf(i,2);
  r3(i,3)-rf(i,3)]);
\]
\[r4f(i,:) = inv(cof)*([r4(i,1)-rf(i,1);
  r4(i,2)-rf(i,2);
  r4(i,3)-rf(i,3)]);
\]
\[r5f(i,:) = inv(cof)*([r5(i,1)-rf(i,1);
  r5(i,2)-rf(i,2);
  r5(i,3)-rf(i,3)]);
\]
\[r6f(i,:) = inv(cof)*([r6(i,1)-rf(i,1);
  r6(i,2)-rf(i,2);
  r6(i,3)-rf(i,3)]);
\]
end

figure(2)

subplot(3,1,1)
plot(t,rlf(:,1),t,rlf(:,2),t,rlf(:,3))
legend('x component','y component','z component');
ylabel('rlf >');
xlabel('Time t in seconds >');
title('Position of satellite 1 w.r.t virtual frame in time');
axis([0 100 -200 200])
grid on

subplot(3,1,2)
plot(t,r2f(:,1),t,r2f(:,2),t,r2f(:,3))
legend('x component','y component','z component');
ylabel('r2f >');
xlabel('Time t in seconds >');
title('Position of satellite 2 w.r.t virtual frame in time');
axis([0 100 -200 200])
grid on

subplot(3,1,3)
plot(t,r3f(:,1),t,r3f(:,2),t,r3f(:,3))
legend('x component','y component','z component');
ylabel('r3f >');
xlabel('Time t in seconds >');
title('Position of satellite 3 w.r.t virtual frame in time');
axis([0 100 -200 200])

grid on
figure(3)
subplot(3,1,1)
plot(t,r4f(:,1),t,r4f(:,2),t,r4f(:,3))
legend('x component','y component','z component');
ylabel('rf4 >');
xlabel('Time t in seconds---------------->');
title('Position of satellite 4 w.r.t virtual frame in time');
axis([0 100 -200 200])
grid on
subplot(3,1,2)
plot(t,r5f(:,1),t,r5f(:,2),t,r5f(:,3))
legend('x component','y component','z component');
ylabel('rf5 >');
xlabel('Time t in seconds---------------->');
title('Position of satellite 5 w.r.t virtual frame in time');
axis([0 100 -200 200])
grid on
subplot(3,1,3)
plot(t,r6f(:,1),t,r6f(:,2),t,r6f(:,3))
legend('x component','y component','z component');
ylabel('rf6 >');
xlabel('Time t in seconds---------------->');
title('Position of satellite 6 w.r.t virtual frame in time');
axis([0 100 -200 200])

grid on

% Defines the velocity of virtual frame and satellite 1,2,3,4,5,6 w.r.t inertial frame
for i=1:3
    vf(:,i)=X(:,3+i);
    v1(:,i)=X(:,22+i);
    v2(:,i)=X(:,35+i);
    v3(:,i)=X(:,48+i);
    v4(:,i)=X(:,48+13+i);
    v5(:,i)=X(:,48+26+i);
    v6(:,i)=X(:,48+39+i);
end
for i=1:length(X)
    Vf(i)=norm([vf(i,1);vf(i,2);vf(i,3)]);
    Vl(i)=norm([v1(i,1);v1(i,2);v1(i,3)]);
end
v1(i,2); v1(i,3));
V2(i)=norm([v2(i,1); v2(i,2); v2(i,3)]);
V3(i)=norm([v3(i,1); v3(i,2); v3(i,3)]);
V4(i)=norm([v4(i,1); v4(i,2); v4(i,3)]);
V5(i)=norm([v5(i,1); v5(i,2); v5(i,3)]);
V6(i)=norm([v6(i,1); v6(i,2); v6(i,3)]);
end

figure(4)
plot(t,Vf(:),t,Vl(:),t,V2(:),t,V3(:),t,V4(:),t,V5(:),t,V6(:))
legend('Virtual frame','Satellite 1','Satellite 2','Satellite 3','Satellite 4','Satellite 5','Satellite 6');
ylabel('Velocity V >');
xlabel('Time t in seconds >');
title('Velocity w.r.t time');
grid on

%Below the orientation of the three satellites w.r.t to inertial frame are given below
for i=1:4
    q1(:,i)=X(:,25+i);
    q2(:,i)=X(:,38+i);
    q3(:,i)=X(:,51+i);
    q4(:,i)=X(:,51+13+i);
    q5(:,i)=X(:,51+26+i);
    q6(:,i)=X(:,51+39+i);
end

%Below we find the orientation of the SIX satellites w.r.t to virtual frame
for i=1:length(X)
    mf=[qf(i,4) -qf(i,3) qf(i,2) qf(i,1);
        qf(i,3) qf(i,4) -qf(i,1) qf(i,2);
        -qf(i,2) qf(i,1) qf(i,4) qf(i,3);
        -qf(i,1) -qf(i,2) -qf(i,3) qf(i,4)];
    q1f(i,:)=inv(mf)*[q1(i,1);
                        q1(i,2));
ql(i,3);
ql(i,4)];
q2f(i,:)=inv(mf)*[q2(i,1);
q2(i,2);
q2(i,3);
q2(i,4)];
q3f(i,:)=inv(mf)*[q3(i,1);
q3(i,2);
q3(i,3);
q3(i,4)];
q4f(i,:)=inv(mf)*[q4(i,1);
q4(i,2);
q4(i,3);
q4(i,4)];
q5f(i,:)=inv(mf)*[q5(i,1);
q5(i,2);
q5(i,3);
q5(i,4)];
q6f(i,:)=inv(mf)*[q6(i,1);
q6(i,2);
q6(i,3);
q6(i,4)];
end
figure(5)
subplot(3,2,1)
plot(t,q1f(:,1),t,q1f(:,2),t,q1f(:,3),t,q1f(:,4));
legend('q1f(1)' , 'q1f(2)' , 'q1f(3)' , 'q1f(4)' );
ylabel('Quaternion q1------------------');
xlabel('Time t in seconds------------------');
title('quaternion of satellite 1 w.r.t virtual frame vs time');
grid on
subplot(3,2,2)
plot(t,q2f(:,1),t,q2f(:,2),t,q2f(:,3),t,q2f(:,4));
legend('q2f(1)' , 'q2f(2)' , 'q2f(3)' , 'q2f(4)' );
ylabel('Quaternion q2------------------');
xlabel('Time t in seconds------------------');
title('quaternion of satellite 2 w.r.t virtual frame vs time');
grid on
subplot(3,2,3)
plot(t,q3f(:,1),t,q3f(:,2),t,q3f(:,3),t,q3f(:,4));
legend('q3f(1)' , 'q3f(2)' , 'q3f(3)' , 'q3f(4)' );
ylabel('Quaternion q3------------------');
xlabel('Time t in seconds------------------');
title('quaternion of satellite 3 w.r.t virtual frame vs time');
grid on
subplot(3,2,4)
plot(t,q4f(:,1),t,q4f(:,2),t,q4f(:,3),t,q4f(:,4));
legend('q4f(1)','q4f(2)','q4f(3)','q4f(4)');
ylabel('Quaternion q4--------------->');
xlabel('Time t in seconds---------->');
title('Quaternion of satellite 4 w.r.t virtual frame vs time');
grid on
subplot(3,2,5)
plot(t,q5f(:,1),t,q5f(:,2),t,q5f(:,3),t,q5f(:,4));
legend('q5f(1)','q5f(2)','q5f(3)','q5f(4)');
ylabel('Quaternion q5----------------->');
xlabel('Time t in seconds---------->');
title('Quaternion of satellite 5 w.r.t virtual frame vs time');
grid on
subplot(3,2,6)
plot(t,q6f(:,1),t,q6f(:,2),t,q6f(:,3),t,q6f(:,4));
legend('q6f(1)','q6f(2)','q6f(3)','q6f(4)');
ylabel('Quaternion q6----------------->');
xlabel('Time t in seconds---------->');
title('Quaternion of satellite 6 w.r.t virtual frame vs time');
grid on

%Below we plot the orientation of the three satellites w.r.t inertial frame
figure(7)
subplot(3,2,1)
plot(t,q1(:,1),t,q1(:,2),t,q1(:,3),t,q1(:,4));
legend('q1(1)','q1(2)','q1(3)','q1(4)');
ylabel('q1 --------------- >');
xlabel('Time t in seconds---------->');
title('Quaternion of satellite 1 w.r.t inertial frame vs time');
grid on
subplot(3,2,2)
plot(t,q2(:,1),t,q2(:,2),t,q2(:,3),t,q2(:,4));
legend('q2(1)','q2(2)','q2(3)','q2(4)');
ylabel('q2 --------------- >');
xlabel('Time t in seconds---------->');
title('Quaternion of satellite 2 w.r.t inertial frame vs time');
grid on
subplot(3,2,3)
plot(t,q3(:,1),t,q3(:,2),t,q3(:,3),t,q3(:,4));
legend('q3(1)','q3(2)','q3(3)','q3(4)');
ylabel('q3 --------------- >');
xlabel('Time t in seconds---------->');
title('Quaternion of satellite 3 w.r.t inertial frame vs time');
grid on
subplot(3,2,4)
plot(t,q4(:,1),t,q4(:,2),t,q4(:,3),t,q4(:,4));
legend('q4(1)','q4(2)','q4(3)','q4(4)');
ylabel('q4 --------------- >');
xlabel('Time t in seconds---------->');
title('Quaternion of satellite 4 w.r.t inertial frame vs time');
grid on
subplot(3,2,5)
plot(t,q5(:,1),t,q5(:,2),t,q5(:,3),t,q5(:,4))
legend('q5(1)', 'q5(2)', 'q5(3)', 'q5(4)');
ylabel('q5 >');
xlabel('Time t in seconds >');
title('Quaternion of satellite 5 w.r.t inertial frame vs time');
grid on
subplot(3,2,6)
plot(t,q6(:,1),t,q6(:,2),t,q6(:,3),t,q6(:,4))
legend('q6(1)', 'q6(2)', 'q6(3)', 'q6(4)');
ylabel('q6 >');
xlabel('Time t in seconds >');
title('Quaternion of satellite 6 w.r.t inertial frame vs time');
grid on

% The angular velocity of virtual frame and the three satellites are given
% by
for i=1:3
  wf(:,i)=X(:,10+i);
  w1(:,i)=X(:,29+i);
  w2(:,i)=X(:,42+i);
  w3(:,i)=X(:,55+i);
  w4(:,i)=X(:,55+13+i);
  w5(:,i)=X(:,55+26+i);
  w6(:,i)=X(:,55+39+i);
end
for i=1:length(X)
  Wf(i)=norm([wf(i,1);
                wf(i,2);
                wf(i,3)]);
  W1(i)=norm([w1(i,1);
                w1(i,2);
                w1(i,3)]);
  W2(i)=norm([w2(i,1);
                w2(i,2);
                w2(i,3)]);
  W3(i)=norm([w3(i,1);
                w3(i,2);
                w3(i,3)]);
  W4(i)=norm([w4(i,1);
                w4(i,2);
                w4(i,3)]);
  W5(i)=norm([w5(i,1);
                w5(i,2);
                w5(i,3)]);
  W6(i)=norm([w6(i,1)];
\text{w6(i,2);}
\text{w6(i,3));}

end
\text{Wf=Wf;}
\text{W1=W1';}
\text{W2=W2';}
\text{W3=W3';}
\text{W4=W4';}
\text{W5=W5';}
\text{W6=W6';}

\text{figure(8)}
\text{plot(t,Wf(:,1),t,W1(:,1),t,W2(:,1),t,W3(:,1),t,W4(:,1),t,W5(:,1),t,W6(:,1));}
\text{legend('Virtual frame','Satellite 1','Satellite 2','Satellite 3','Satellite 4','Satellite 5','Satellite 6');}
\text{ylabel('Angular Velocity W----------->');}
\text{xlabel('Time t in seconds----------->');}
\text{title('Angular velocity w.r.t time');}
\text{grid on}

\%TRANSFER TO STK
\%
\%Quaternerion 3dimensional array
\%for i=1:4
\text{Xq(:,i,1)=qf(:,i);%Virtual frame}
\text{Xq(:,i,2)=ql(:,i);%follower 1}
\text{Xq(:,i,3)=q2(:,i);%follower 2}
\text{Xq(:,i,4)=q3(:,i);%follower 3}
\text{Xq(:,i,5)=q4(:,i);%follower 4}
\text{Xq(:,i,6)=q5(:,i);%follower 5}
\text{Xq(:,i,7)=q6(:,i);%follower 6}
\%for j=1:7
\text{if j==1

76
att=fopen('virtual.a','w');
end
if j==2
att=fopen('Flwr1.a','w');
end
if j==3
att=fopen('Flwr2.a','w');
end
if j==4
att=fopen('Flwr3.a','w');
end
if j==5
att=fopen('Flwr4.a','w');
end
if j==6
att=fopen('Flwr5.a','w');
end
if j==7
att=fopen('Flwr6.a','w');
end
fprintf(att,'stk.v.5.0

');
fprintf(att,'BEGIN Attitude

');
fprintf(att,'ScenarioEpoch	1 Jun 2005 12:00:00.00
NumberOfAttitudePoints	');
fprintf(att,'%d',length(t));
fprintf(att,'
');
fprintf(att,'BlockingFactor	20
');
fprintf(att,'InterpolationOrder	1
');
fprintf(att,'CentralBody	Earth
');
fprintf(att,'CoordinateAxes	Fixed

');
fprintf(att,'AttitudeTimeQuaternions

');
for h=l:length(t)
 fprintf(att,'
%-.4f %-.3f %-.3f %-.3f %-.3f

',h,Xq(h,1,j),Xq(h,2,j),Xq(h,3,j),Xq(h,4,j));
end
fprintf(att,'END Attitude');
status = fclose(att);
end

% Create ephemeris (time/position) files for transfer to Satellite Tool Kit (STK)
for j=1:7
    if j==1
        eph=fopen('virtual.e','w');
    end
    if j==2
        eph=fopen('Flwr1.e','w');
    end
    if j==3
        eph=fopen('Flwr2.e','w');
    end
    if j==4
        eph=fopen('Flwr3.e','w');
    end
    if j==5
        eph=fopen('Flwr4.e','w');
    end
    if j==6
        eph=fopen('Flwr5.e','w');
    end
    if j==7
        eph=fopen('Flwr6.e','w');
    end
    fprintf(eph,'stk.v.5.0
    BEGIN Ephemeris
    ScenarioEpoch		 1 Jun 2005 12:00:00.00
    CoordinateSystem	J2000
    DistanceUnit		 Meters
    NumberOfEphemerisPoints	%d',length(t))
    CentralBody			Earth
    EphemerisTimePos
    for h=1:length(t)
        fprintf(eph,'
        E(h,1:j),Xt(h,2:j),Xt(h,3:j));
    end
    status = fclose(eph);
end
TRANSLATION & ROTATIONAL CONTROL FUNCTION CODE

% Virtual structure approach without formation feedback
function dx=f1(t,x)
format long
global Ji Kri Kvi kqi Jf Kr kq Ketha kp ka ke;
global rfd qfd ethafd r1fd r2fd r3fd r4fd r5fd r6fd q1fd q2fd q3fd q4fd q5fd q6fd nn;
% The desired position of virtual frame
Rf=11000e3;
% Rf=1000;
% orbital parameters
mu=398600e9;
% Virtual structure orbital period
T=(2*pi/sqrt(mu))*Rf^(3/2);
% Orbital rate
n=2*pi/T;

rfd=[Rf*cos(n*t);
     Rf*sin(n*t);
     0];

qfd=[0;
     0;
     sin(nn*t/2);
     cos(nn*t/2)];

%%%%%%%%%%%%%%%%%%%%
% for virtual frame
%%%%%%%%%%%%%%%%%%%%
rf=[x(1);
    x(2);
    x(3)];
vf=[x(4);
    x(5);
    x(6)];
qf=[x(7);
    x(8);
    x(9);
    x(10)];
% Vector part of qf is
vqf=[qf(1);
     qf(2);
     qf(3)];
wf=[x(11);
    x(12);
    x(13)];
ethaf=[x(14);
x(15);
  x(16));
dethaf=[x(17);
  x(18);
  x(19)];
expansion=[x(14) 0 0;
  0 x(15) 0;
  0 0 x(16)];
dexpansion=[x(17) 0 0;
  0 x(18) 0;
  0 0 x(19)];

%================================
%for the first satellite
%================================
  r1=[x(20);
   x(21);
   x(22)];
  v1=[x(23);
   x(24);
   x(25)];
  q1=[x(26);
   x(27);
   x(28);
   x(29)];
  vq1=[q1(1);
   q1(2);
   q1(3)];
  w1=[x(30);
   x(31);
   x(32)];

%================================
%for the second satellite
%================================
  r2=[x(33);
   x(34);
   x(35)];
  v2=[x(36);
   x(37);
   x(38)];
  q2=[x(39);
   x(40);
   x(41);
   x(42)];
  vq2=[q2(1);
   q2(2);
   q2(3)];
w2=[x(43);
    x(44);
    x(45)];

% for the third satellite
r3=[x(46);
    x(47);
    x(48)];
v3=[x(49);
    x(50);
    x(51)];
q3=[x(52);
    x(53);
    x(54);
    x(55)];

qv3=[q3(1);
    q3(2);
    q3(3)];
w3=[x(56);
    x(57);
    x(58)];

% for the fourth satellite
r4=[x(59);
    x(60);
    x(61)];
v4=[x(62);
    x(63);
    x(64)];
q4=[x(65);
    x(66);
    x(67);
    x(68)];

qv4=[q4(1);
    q4(2);
    q4(3)];
w4=[x(69);
    x(70);
    x(71)];

% for the fifth satellite
r5=[x(72);
    x(73)];
\[ x(74); \]
\[ v5=[x(75); \]
\[ x(76); \]
\[ x(77)]; \]
\[ q5=[x(78); \]
\[ x(79); \]
\[ x(80); \]
\[ x(81)]; \]
\% Vector part of q5 is \[ vq5=[q5(1); \]
\[ q5(2); \]
\[ q5(3)]; \]
\[ w5=[x(82); \]
\[ x(83); \]
\[ x(84)]; \]
\%==========================================
\% for the sixth satellite
\%==========================================
\[ r6=[x(85); \]
\[ x(86); \]
\[ x(87)]; \]
\[ v6=[x(88); \]
\[ x(89); \]
\[ x(90)]; \]
\[ q6=[x(91); \]
\[ x(92); \]
\[ x(93); \]
\[ x(94)]; \]
\% Vector part of q6 is \[ vq6=[q6(1); \]
\[ q6(2); \]
\[ q6(3)]; \]
\[ w6=[x(95); \]
\[ x(96); \]
\[ x(97)]; \]
\%==========================================
\% The Virtual structure dynamics for the virtual frame are given below
\%==========================================
\[ drf=vf; \% position \]
\[ dvf=-Kr*(rf-rfd)-(kp*(norm(rf-rfd)^2)*vf); \% velocity \]
\[ wfx=[0 -wf(3) wf(2); \]
\[ wf(3) 0 -wf(1); \]
\[ -wf(2) wf(1) 0]; \]
\[ dvqf=-0.5*wfx*vqf+0.5*qf(4)*wf\]
\% vector part of quaternion
dqf=-0.5*wf*vqf;%scalar part of quaternion
%the below is the quaternion multiplication of (qf*qfd)
q=[-vqf;
  qf(4)];
mf=[q(4) -q(3) q(2) q(1);
  q(3) q(4) -q(1) q(2);
  -q(2) q(1) q(4) q(3)];
qef=mf*qfd;
dwf=inv(Jf)*((-wfx*Jf*wf)+(kq*qef)-ka*norm(qf-qfd)^2*wf);%angular velocity

dethaf=dethaf;%expansion
ddethetaf=Ketha*(ethaf-ethafd)-ke*norm(ethaf-ethafd)^2*dethaf;
ddexpansion=[ddethetaf(1) 0 0;
  0 ddethetaf(2) 0;
  0 0 ddethetaf(3)];
dwfx=[0 -dwf(3) dwf(2);
  dwf(3) 0 -dwf(1);
  -dwf(2) dwf(1) 0];

%==---------------------------------------------------------------=
%==
%The dynamics for the three satellites in the virtual structure are
%given below
%==---------------------------------------------------------------=
cof=[1-2*(qf(2)^2+qf(3)^2) 2*(qf(1)*qf(2)+qf(3)*qf(4)) 2*(qf(1)*qf(3)-qf(2)*qf(4));
  2*(qf(2)*qf(1)-qf(3)*qf(4)) 1-2*(qf(1)^2+qf(3)^2) 2*(qf(2)*qf(3)+qf(1)*qf(4));
  2*(qf(3)*qf(1)+qf(2)*qf(4)) 2*(qf(3)*qf(2)-qf(1)*qf(4)) 1-2*(qf(1)^2+qf(2)^2)];

%For the first satellite
%==---------------------
  r1d=rf+cof*expansion*r1fd;
  v1d=vf+cof*dexpansion*r1fd+wfx*(cof*expansion*r1fd);
mf=[qf(4) -qf(3) qf(2) qf(1);
  qf(3) qf(4) -qf(1) qf(2);
  -qf(2) qf(1) qf(4) qf(3);
  -qf(1) -qf(2) -qf(3) qf(4)];
q1d=mf*q1fd;
w1d=wf;
%-----------------------------------------------------------------------
  dv1d=dvf+2*wfx*(cof*dexpansion*r1fd)+cof*dexpansion*r1fd+dwx*(cof*expansion *r1fd);
  dw1d=dwf;
%-----------------------------------------------------------------------
  w1x=[0 -w1(3) w1(2);
    w1(3) 0 -w1(1);
    -w1(2) w1(1) 0];
  dl1=v1;%position
dv1=dv1d-Kri*(r1-r1d)-Kvi*(v1-v1d);%velocity
dvq1=-0.5*w1x*vq1+0.5*q1(4)*w1;%vector part of quaternion
dq1=-0.5*w1*vq1;%scalar part of quaternion

q=[-vq1;
    q1(4)];
m=[q(4) -q(3) q(2) q(1);
    q(3) q(4) -q(1) q(2);
    -q(2) q(1) q(4) q(3)];

qe1=m*q1d;%quaternion multiplication of (q1*q1d)
dw1=inv(Ji)*(-wlx*Ji*wl+Ji*dwld+0.5*wlx*Ji*(wl+wld)+kqi*qe1-Kwi*(wl-
wld));%angular velocity

%For the second satellite

r2d=rf+cof*expansion*r2fd;
v2d=vf+cof*dexpansion*r2fd+wfx*(cof*expansion*r2fd);
mf=[qf(4) -qf(3) qf(2) qf(1);
    qf(3) qf(4) -qf(1) qf(2);
    -qf(2) qf(1) qf(4) qf(3);
    -qf(1) -qf(2) -qf(3) qf(4)];
q2d=mf*q2fd;
w2d=wf;

%=================================================================

dv2d=dv2d-Kri*(r2-r2d)-Kvi*(v2-v2d);%velocity
dvq2=-0.5*w2x*vq2+0.5*q2(4)*w2;%vector part of quaternion
dq2=-0.5*w2*vq2;%scalar part of quaternion

q=[-vq2;
    q2(4)];
m=[q(4) -q(3) q(2) q(1);
    q(3) q(4) -q(1) q(2);
    -q(2) q(1) q(4) q(3)];
qe2=m*q2d;%quaternion multiplication of (q2*q2d)
dw2=inv(Ji)*(-w2x*Ji*w2+Ji*dw2d+0.5*w2x*Ji*(w2+w2d)+kqi*qe2-Kwi*(w2-
w2d));%angular velocity

%For the third satellite

r3d=rf+cof*expansion*r3fd;
v3d=vf+cof*dexpansion*r3fd+wfx*(cof*expansion*r3fd);
mf=[qf(4) -qf(3) qf(2) qf(1);
\[ q_f(3)q_f(4) - q_f(1)q_f(2); \]
\[-q_f(2)q_f(1)q_f(4)q_f(3); \]
\[-q_f(1) - q_f(2) - q_f(3)q_f(4); \]
\[ q_{3d} = m_f*q_{3fd}; \]
\[ w_{3d} = w_f; \]
\[
\text{%---------------------------------------------------------}
\]
\[ dv_{3d} = dv_f + 2*w_fx*(cof*expansion*r_{3fd}) + cof*dexpansion*r_{3fd} + dw_fx*(cof*expansion*r_{3fd}); \]
\[ dw_{3d} = dw_f; \]
\[
\text{%---------------------------------------------------------}
\]
\[ w_{3x} = \begin{bmatrix}
0 & -w_{3}(3) & w_{3}(2) \\
-3(3) & 0 & -w_{3}(1) \\
-w_{3}(2) & w_{3}(1) & 0
\end{bmatrix}; \]
\[ dr_{3} = v_{3}; \%	ext{position} \]
\[ dv_{3} = dv_{3d} - K_{ri}*(r_{3}-r_{3d}) - K_{vi}*(v_{3}-v_{3d}); \%	ext{velocity} \]
\[ dvq_{3} = -0.5*w_{3x}*vq_{3} + 0.5*q_{3}(4)*w_{3}; \%	ext{vector part of quaternion} \]
\[ dq_{3} = -0.5*w_{3}*vq_{3}; \%	ext{scalar part of quaternion} \]
\[ q = [-v_{q3}; q_{3}(4)]; \]
\[ m = [q_{f}(4) - q_{f}(3)q_{f}(2) q_{f}(1); q_{f}(3)q_{f}(4) - q_{f}(1)q_{f}(2); -q_{f}(2) q_{f}(1)q_{f}(4)q_{f}(3); -q_{f}(1) - q_{f}(2) - q_{f}(3)q_{f}(4)]; \]
\[ q_{4d} = m_{f}*q_{4fd}; \]
\[ w_{4d} = w_f; \]
\[
\text{%---------------------------------------------------------}
\]
\[ dv_{4d} = dv_f + 2*w_fx*(cof*expansion*r_{4fd}) + cof*dexpansion*r_{4fd} + dw_fx*(cof*expansion*r_{4fd}); \]
\[ dw_{4d} = dw_f; \]
\[
\text{%---------------------------------------------------------}
\]
\[ w_{4x} = \begin{bmatrix}
0 & -w_{4}(3) & w_{4}(2) \\
-3(3) & 0 & -w_{4}(1) \\
-w_{4}(2) & w_{4}(1) & 0
\end{bmatrix}; \]
\[ dr_{4} = v_{4}; \%	ext{position} \]
\[ dv_{4} = dv_{4d} - K_{ri}*(r_{4}-r_{4d}) - K_{vi}*(v_{4}-v_{4d}); \%	ext{velocity} \]
\[ dvq_{4} = -0.5*w_{4x}*vq_{4} + 0.5*q_{4}(4)*w_{4}; \%	ext{vector part of quaternion} \]
\[ dq4 = -0.5 \cdot w4' \cdot vq4; \text{ scalar part of quaternion} \]
\[ q = [-vq4; \\
q4(4)]; \]
\[ m = [q(4) - q(3) q(2) q(1); \\
q(3) q(4) - q(1) q(2); \\
-q(2) q(1) q(4) q(3)]; \]
\[ qe4 = m * q4d; \text{ quaternion multiplication of } (q1 * q1d) \]
\[ dw4 = \text{inv}(Ji) * (-w4x * Ji * w4 + Ji * dw4d + 0.5 * w4x * Ji * (w4 + w4d) + kqi * qe4 - Kwi * (w4 - w4d)); \text{ angular velocity} \]

% For the fifth satellite

%---------------------------------------------------------------
\[ r5d = rf + cof * expansion * r5fd; \]
\[ v5d = vf + cof * dexpansion * r5fd + wfx * (cof * expansion * r5fd); \]
\[ mf = [qf(4) - qf(3) qf(2) qf(1); \\
qf(3) qf(4) - qf(1) qf(2); \\
-qf(2) qf(1) qf(4) qf(3); \\
-qf(1) - qf(2) - qf(3) qf(4)]; \]
\[ q5d = mf * q5fd; \]
\[ w5d = wf; \]
%---------------------------------------------------------------
\[ dv5d = dvf + 2 \cdot wfx * (cof * dexpansion * r5fd) + cof * ddexpansion * r5fd + dwfx * (cof * expansion * r5fd); \]
\[ dw5d = dwf; \]
%---------------------------------------------------------------
\[ w5x = [0 - w5(3) w5(2); \\
-w5(3) 0 - w5(1); \\
-w5(2) w5(1) 0]; \]
\[ dr5 = v5; \text{ position} \]
\[ dv5 = dv5d - Kri * (r5 - r5d) - Kvi * (v5 - v5d); \text{ velocity} \]
\[ dq5 = -0.5 \cdot w5x \cdot vq5 + 0.5 \cdot q5(4) \cdot w5; \text{ scalar part of quaternion} \]
\[ dq5 = -0.5 \cdot w5x \cdot vq5; \text{ scalar part of quaternion} \]
\[ q = [-vq5; \\
q5(4)]; \]
\[ m = [q(4) - q(3) q(2) q(1); \\
q(3) q(4) - q(1) q(2); \\
-q(2) q(1) q(4) q(3)]; \]
\[ qe5 = m * q5d; \text{ quaternion multiplication of } (q1 * q1d) \]
\[ dw5 = \text{inv}(Ji) * (-w5x * Ji * w5 + Ji * dw5d + 0.5 * w5x * Ji * (w5 + w5d) + kqi * qe5 - Kwi * (w5 - w5d)); \text{ angular velocity} \]
% For the sixth satellite

%---------------------------------------------------------------
\[ r6d = rf + cof * expansion * r6fd; \]
\[ v6d = vf + cof * dexpansion * r6fd + wfx * (cof * expansion * r6fd); \]
\[ mf = [qf(4) - qf(3) qf(2) qf(1); \\
---
86
\( q_f(3) q_f(4) - q_f(1) q_f(2); \)
\( -q_f(2) q_f(1) q_f(4) q_f(3); \)
\( -q_f(1) -q_f(2) -q_f(3) q_f(4); \)

\( q_6d = m_f * q_6fd; \)
\( w_6d = w_f; \)

\[
dv_6d = dv_f + 2 * w_f x (c_0_f * dexpansion * r_6fd) + c_0_f * dexpansion * r_6fd + d_w_f x (c_0_f * dexpansion * r_6fd); \\
dw_6d = dw_f; \\
\]

\[
w_6x = [0, -w_6(3), w_6(2)]; \\
\]

\[
dr_6 = v_6; \]
\[
dv_6 = dv_6d - kri * (r_6 - r_6d) - kvi * (v_6 - v_6d); \]
\[
dv_6q = 0.5 * w_6x * v_6q + 0.5 * q_6(4) * w_6; \]
\[
dq_6 = 0.5 * w_6 * v_6q; \]

\[
q = [v_6q; \\
q_6(4)]; \\
m = [q(4) - q(3) q(2) q(1); \\
q(3) q(4) - q(1) q(2); \\
-q(2) q(1) q(4) q(3)]; \\
\]

\[
qe_6 = m * q_6d; \]
\[
dw_6 = inv(J_i) * [-w_6x * J_i * w_6 + J_i * dw_6d + 0.5 * w_6x * J_i * (w_6 + w_6d) + kqi * q_6 - kwi * (w_6 - w_6d)]; \]

\[
dx = [dr_f; \\
dv_f; \\
dv_qf; \\
dq_f; \\
dw_f; \\
dethaf; \\
d_dethaf; \\
dr_1; \\
dv_1; \\
dv_q1; \\
dq_1; \\
dw_1; \\
\]

\[
dr_2; \\
dv_2; \\
dv_q2; \\
dq_2; \\
dw_2; \\
\]

\[
dr_3; \\
\]
function x=rk4(name,t0,x0,h)
  t1=t0+h/2; t2=t0+h;
  f0=feval(name,t0,x0); x1=x0+h*f0/2;
  f1=feval(name,t1,x1); x2=x0+h*f1/2;
  f2=feval(name,t1,x2); x3=x0+h*f2;
  f3=feval(name,t2,x3);
  x=x0+h*(f0+2*f1+2*f2+f3)/6;
APPENDIX B

SAMPLE EPHEMERIS FILE GENERATED FOR SATELLITE TOOL KIT

stk.v.5.0

BEGIN Ephemeris

ScenarioEpoch  1 Jun 2005 12:00:00.00
CoordinateSystem  J2000
DistanceUnit  Meters
NumberOfEphemerisPoints  101
CentralBody  Earth
EphemerisTimePos

1.0000  10999999.588  318683.715  0.0
2.0000  10999995.470  -66126.684  0.0
3.0000  10999986.083  -201782.922  0.0
4.0000  10999974.211  138884.973  0.0
5.0000  10999960.354  169713.169  0.0
6.0000  10999942.257  -86684.390  0.0
7.0000  10999920.098  -34054.508  0.0
8.0000  10999895.563  161225.709  0.0
9.0000  10999868.090  84244.183  0.0
10.0000  10999836.533  -33888.601  0.0
11.0000  10999801.605  66716.794  0.0
12.0000  10999764.007  144144.039  0.0
13.0000  10999723.018  59894.364  0.0
14.0000  10999678.281  33809.645  0.0
15.0000  10999630.438  116967.532  0.0
16.0000  10999579.601  128282.171  0.0
17.0000  10999525.250  73918.698  0.0
18.0000  10999467.431  90979.680  0.0
19.0000  10999406.529  140469.292  0.0
20.0000  10999342.415  127167.331  0.0
21.0000  10999274.824  104735.024  0.0
22.0000  10999203.922  132660.029  0.0
23.0000  10999129.870  154986.050  0.0
24.0000  10999052.501  140156.920  0.0
25.0000  10998971.734  138903.216  0.0
26.0000  10998887.717  162702.480  0.0
27.0000  10998800.474  169844.725  0.0
28.0000  10998709.887  161969.952  0.0
29.0000  10998615.967  170602.364  0.0
30.0000  10998518.801  186605.283  0.0
31.0000  10998418.360  188127.155  0.0
32.0000  10998314.583  187626.136  0.0
33.0000  10998207.509  198717.778  0.0
34.0000  10998097.174  208459.914  0.0
35.0000  10997983.940  209663.675  0.0
36.0000  10997866.589  214035.121  0.0
37.0000  10997746.354  224151.712  0.0
38.0000  10997622.841  230406.626  0.0
39.0000  10997496.024  233228.198  0.0
40.0000  10997365.903  239864.564  0.0
41.0000  10997232.501  248175.741  0.0
<table>
<thead>
<tr>
<th></th>
<th></th>
<th>1097095.809</th>
<th>253147.369</th>
<th>0.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>42.0000</td>
<td>43.0000</td>
<td>10996955.814</td>
<td>257671.457</td>
<td>0.0</td>
</tr>
<tr>
<td>44.0000</td>
<td>45.0000</td>
<td>10996812.525</td>
<td>264864.119</td>
<td>0.0</td>
</tr>
<tr>
<td>46.0000</td>
<td>47.0000</td>
<td>10996665.951</td>
<td>271732.129</td>
<td>0.0</td>
</tr>
<tr>
<td>48.0000</td>
<td>49.0000</td>
<td>10996516.081</td>
<td>276638.014</td>
<td>0.0</td>
</tr>
<tr>
<td>50.0000</td>
<td>51.0000</td>
<td>10996362.914</td>
<td>282282.829</td>
<td>0.0</td>
</tr>
<tr>
<td>52.0000</td>
<td>53.0000</td>
<td>10996206.455</td>
<td>289244.995</td>
<td>0.0</td>
</tr>
<tr>
<td>54.0000</td>
<td>55.0000</td>
<td>10996046.707</td>
<td>295311.910</td>
<td>0.0</td>
</tr>
<tr>
<td>56.0000</td>
<td>57.0000</td>
<td>1099583.664</td>
<td>300593.865</td>
<td>0.0</td>
</tr>
<tr>
<td>58.0000</td>
<td>59.0000</td>
<td>10995717.325</td>
<td>306757.309</td>
<td>0.0</td>
</tr>
<tr>
<td>60.0000</td>
<td>61.0000</td>
<td>1099547.696</td>
<td>313301.183</td>
<td>0.0</td>
</tr>
<tr>
<td>62.0000</td>
<td>63.0000</td>
<td>10995374.775</td>
<td>319074.874</td>
<td>0.0</td>
</tr>
<tr>
<td>64.0000</td>
<td>65.0000</td>
<td>10995198.559</td>
<td>324749.654</td>
<td>0.0</td>
</tr>
<tr>
<td>66.0000</td>
<td>67.0000</td>
<td>10995019.050</td>
<td>331038.641</td>
<td>0.0</td>
</tr>
<tr>
<td>68.0000</td>
<td>69.0000</td>
<td>10994836.250</td>
<td>337249.608</td>
<td>0.0</td>
</tr>
<tr>
<td>70.0000</td>
<td>71.0000</td>
<td>10994650.157</td>
<td>343009.628</td>
<td>0.0</td>
</tr>
<tr>
<td>72.0000</td>
<td>73.0000</td>
<td>10994460.770</td>
<td>348941.914</td>
<td>0.0</td>
</tr>
<tr>
<td>74.0000</td>
<td>75.0000</td>
<td>10994268.091</td>
<td>355176.183</td>
<td>0.0</td>
</tr>
<tr>
<td>76.0000</td>
<td>77.0000</td>
<td>10994072.120</td>
<td>361202.853</td>
<td>0.0</td>
</tr>
<tr>
<td>78.0000</td>
<td>79.0000</td>
<td>10993872.857</td>
<td>367050.004</td>
<td>0.0</td>
</tr>
<tr>
<td>80.0000</td>
<td>81.0000</td>
<td>10993670.300</td>
<td>373100.758</td>
<td>0.0</td>
</tr>
<tr>
<td>82.0000</td>
<td>83.0000</td>
<td>10993464.452</td>
<td>379237.834</td>
<td>0.0</td>
</tr>
<tr>
<td>84.0000</td>
<td>85.0000</td>
<td>10993255.312</td>
<td>385197.220</td>
<td>0.0</td>
</tr>
<tr>
<td>86.0000</td>
<td>87.0000</td>
<td>10993042.879</td>
<td>391134.709</td>
<td>0.0</td>
</tr>
<tr>
<td>88.0000</td>
<td>89.0000</td>
<td>10992827.154</td>
<td>397213.483</td>
<td>0.0</td>
</tr>
<tr>
<td>90.0000</td>
<td>91.0000</td>
<td>10992608.137</td>
<td>403273.465</td>
<td>0.0</td>
</tr>
<tr>
<td>92.0000</td>
<td>93.0000</td>
<td>10992385.828</td>
<td>409229.732</td>
<td>0.0</td>
</tr>
<tr>
<td>94.0000</td>
<td>95.0000</td>
<td>10992160.227</td>
<td>415226.073</td>
<td>0.0</td>
</tr>
<tr>
<td>96.0000</td>
<td>97.0000</td>
<td>10991931.334</td>
<td>421291.554</td>
<td>0.0</td>
</tr>
<tr>
<td>98.0000</td>
<td>99.0000</td>
<td>10991699.150</td>
<td>427308.818</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
100.0000  10983534.209  601666.690  0.0
101.0000  10983203.307  607676.524  0.0

END Ephemeris
APPENDIX C

SAMPLE ATTITUDE FILE GENERATED FOR SATELLITE TOOL KIT

stk.v.5.0

BEGIN Attitude

ScenarioEpoch 1 Jun 2005 12:00:01.00
NumberOfAttitudePoints 101
BlockingFactor 20
InterpolationOrder 1
CentralBody Earth
CoordinateAxes Fixed

AttitudeTimeQuaternions

1.0000 0.0000 0.0000 0.2470 0.9690
2.0000 0.0000 0.0000 0.4790 0.8780
3.0000 0.0000 0.0000 0.6810 0.7320
4.0000 0.0000 0.0000 0.8410 0.5410
5.0000 0.0000 0.0000 0.9490 0.3160
6.0000 0.0000 0.0000 0.9970 0.0710
7.0000 0.0000 0.0000 0.9840 -0.1780
8.0000 0.0000 0.0000 0.9090 -0.4160
9.0000 0.0000 0.0000 0.7780 -0.6280
10.0000 0.0000 0.0000 0.5990 -0.8010
11.0000 0.0000 0.0000 0.3820 -0.9240
12.0000 0.0000 0.0000 0.1420 -0.9900
13.0000 0.0000 0.0000 -0.1080 -0.9940
14.0000 0.0000 0.0000 -0.3500 -0.9370
15.0000 0.0000 0.0000 -0.5710 -0.8210
16.0000 0.0000 0.0000 -0.7560 -0.6540
17.0000 0.0000 0.0000 -0.8950 -0.4460
18.0000 0.0000 0.0000 -0.9770 -0.2110
19.0000 0.0000 0.0000 -0.9990 0.0370
20.0000 0.0000 0.0000 -0.9590 0.2830
21.0000 0.0000 0.0000 -0.8590 0.5120
22.0000 0.0000 0.0000 -0.7060 0.7080
23.0000 0.0000 0.0000 -0.5090 0.8610
24.0000 0.0000 0.0000 -0.2800 0.9600
25.0000 0.0000 0.0000 -0.0340 0.9990
26.0000 0.0000 0.0000 0.2150 0.9770
27.0000 0.0000 0.0000 0.4500 0.8930
28.0000 0.0000 0.0000 0.6570 0.7540
29.0000 0.0000 0.0000 0.8230 0.5680
30.0000 0.0000 0.0000 0.9380 0.3470
31.0000 0.0000 0.0000 0.9950 0.1040
32.0000 0.0000 0.0000 0.9890 -0.1450
33.0000 0.0000 0.0000 0.9230 -0.3850
34.0000 0.0000 0.0000 0.7990 -0.6020
35.0000 0.0000 0.0000 0.6250 -0.7810
36.0000 0.0000 0.0000 0.4130 -0.9110
37.0000 0.0000 0.0000 0.1740 -0.9850
38.0000 0.0000 0.0000 -0.0750 -0.9970
39.0000 0.0000 0.0000 -0.3190 -0.9480
<table>
<thead>
<tr>
<th>40.0000</th>
<th>0.0000</th>
<th>0.0000</th>
<th>-0.544</th>
<th>-0.839</th>
</tr>
</thead>
<tbody>
<tr>
<td>41.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.734</td>
<td>-0.679</td>
</tr>
<tr>
<td>42.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.879</td>
<td>-0.476</td>
</tr>
<tr>
<td>43.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.970</td>
<td>-0.244</td>
</tr>
<tr>
<td>44.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-1.000</td>
<td>0.004</td>
</tr>
<tr>
<td>45.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.968</td>
<td>0.251</td>
</tr>
<tr>
<td>46.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.876</td>
<td>0.483</td>
</tr>
<tr>
<td>47.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.729</td>
<td>0.685</td>
</tr>
<tr>
<td>48.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.537</td>
<td>0.844</td>
</tr>
<tr>
<td>49.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.312</td>
<td>0.950</td>
</tr>
<tr>
<td>50.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.067</td>
<td>0.998</td>
</tr>
<tr>
<td>51.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.182</td>
<td>0.983</td>
</tr>
<tr>
<td>52.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.420</td>
<td>0.908</td>
</tr>
<tr>
<td>53.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.631</td>
<td>0.776</td>
</tr>
<tr>
<td>54.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.803</td>
<td>0.595</td>
</tr>
<tr>
<td>55.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.926</td>
<td>0.378</td>
</tr>
<tr>
<td>56.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.990</td>
<td>0.137</td>
</tr>
<tr>
<td>57.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.994</td>
<td>-0.112</td>
</tr>
<tr>
<td>58.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.935</td>
<td>-0.354</td>
</tr>
<tr>
<td>59.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.835</td>
<td>-0.575</td>
</tr>
<tr>
<td>60.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.651</td>
<td>-0.759</td>
</tr>
<tr>
<td>61.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.442</td>
<td>-0.897</td>
</tr>
<tr>
<td>62.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.207</td>
<td>-0.978</td>
</tr>
<tr>
<td>63.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.042</td>
<td>-0.999</td>
</tr>
<tr>
<td>64.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.287</td>
<td>-0.958</td>
</tr>
<tr>
<td>65.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.515</td>
<td>-0.857</td>
</tr>
<tr>
<td>66.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.711</td>
<td>-0.703</td>
</tr>
<tr>
<td>67.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.863</td>
<td>-0.505</td>
</tr>
<tr>
<td>68.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.961</td>
<td>-0.276</td>
</tr>
<tr>
<td>69.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.999</td>
<td>-0.029</td>
</tr>
<tr>
<td>70.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.976</td>
<td>0.219</td>
</tr>
<tr>
<td>71.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.891</td>
<td>0.454</td>
</tr>
<tr>
<td>72.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.751</td>
<td>0.660</td>
</tr>
<tr>
<td>73.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.565</td>
<td>0.825</td>
</tr>
<tr>
<td>74.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.343</td>
<td>0.939</td>
</tr>
<tr>
<td>75.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.100</td>
<td>0.995</td>
</tr>
<tr>
<td>76.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.149</td>
<td>0.989</td>
</tr>
<tr>
<td>77.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.389</td>
<td>0.921</td>
</tr>
<tr>
<td>78.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.605</td>
<td>0.796</td>
</tr>
<tr>
<td>79.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.783</td>
<td>0.622</td>
</tr>
<tr>
<td>80.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.913</td>
<td>0.408</td>
</tr>
<tr>
<td>81.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.985</td>
<td>0.170</td>
</tr>
<tr>
<td>82.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.997</td>
<td>-0.079</td>
</tr>
<tr>
<td>83.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.946</td>
<td>-0.323</td>
</tr>
<tr>
<td>84.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.837</td>
<td>-0.547</td>
</tr>
<tr>
<td>85.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.675</td>
<td>-0.737</td>
</tr>
<tr>
<td>86.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.472</td>
<td>-0.881</td>
</tr>
<tr>
<td>87.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.239</td>
<td>-0.971</td>
</tr>
<tr>
<td>88.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.008</td>
<td>1.000</td>
</tr>
<tr>
<td>89.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.256</td>
<td>-0.967</td>
</tr>
<tr>
<td>90.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.487</td>
<td>-0.873</td>
</tr>
<tr>
<td>91.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.688</td>
<td>-0.726</td>
</tr>
<tr>
<td>92.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.846</td>
<td>-0.533</td>
</tr>
<tr>
<td>93.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.951</td>
<td>-0.307</td>
</tr>
<tr>
<td>94.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.998</td>
<td>-0.062</td>
</tr>
<tr>
<td>95.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.982</td>
<td>0.186</td>
</tr>
<tr>
<td>96.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.906</td>
<td>0.424</td>
</tr>
<tr>
<td>97.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.773</td>
<td>0.635</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>------</td>
<td>------</td>
<td>-----</td>
<td>------</td>
</tr>
<tr>
<td>98.0000</td>
<td>0.000</td>
<td>0.000</td>
<td>-0.592</td>
<td>0.806</td>
</tr>
<tr>
<td>99.0000</td>
<td>0.000</td>
<td>0.000</td>
<td>-0.374</td>
<td>0.927</td>
</tr>
<tr>
<td>100.0000</td>
<td>0.000</td>
<td>0.000</td>
<td>-0.133</td>
<td>0.991</td>
</tr>
<tr>
<td>101.0000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.117</td>
<td>0.993</td>
</tr>
</tbody>
</table>

END Attitude