# Computational Problem Set 1

AAE 440: Spring 2022

2 March 2022

Matthew Mader

Contents

[Computational Problem Set 1 1](#_Toc97119402)

[Relevant Notation 3](#_Toc97119403)

[Rotation Matrices 3](#_Toc97119404)

[Sines and Cosines 3](#_Toc97119405)

[Quaternions: 3](#_Toc97119406)

[Problem 1 3](#_Toc97119407)

[(a) 3](#_Toc97119408)

[Given 3](#_Toc97119409)

[Find 3](#_Toc97119410)

[Analysis 3](#_Toc97119411)

[Solution 5](#_Toc97119412)

[(b) 5](#_Toc97119413)

[Find 5](#_Toc97119414)

[Analysis 5](#_Toc97119415)

[Solution 5](#_Toc97119416)

[(c) 5](#_Toc97119417)

[Given 5](#_Toc97119418)

[Find 6](#_Toc97119419)

[Analysis 6](#_Toc97119420)

[Solution 6](#_Toc97119421)

[(d1) 8](#_Toc97119422)

[Given 8](#_Toc97119423)

[Find 8](#_Toc97119424)

[Solution 8](#_Toc97119425)

[(d2) 9](#_Toc97119426)

[Given 9](#_Toc97119427)

[Find 9](#_Toc97119428)

[Solution 9](#_Toc97119429)

[Problem 2 9](#_Toc97119430)

# Relevant Notation

## Rotation Matrices

## Sines and Cosines

## Quaternions:

# Problem 1

## (a)

### Given

Orientation of frame relative to frame at time t=0 is given by a body (3-2-1) sequence of Euler angles and that is measured as:

Where:

### Find

Determine the value of at time , where choose .

Confirm that the unit-norm constraint is satisfied.

### Analysis

Use the (3-2-1) sequence of Euler angles to compute the DCM at

Where:

Computation done in MATLAB with the function *EA321toDCM* made in CP01

Use the DCM to compute the Euler Parameters at using Sheppard’s method

1. Compute the following
2. Obtain the highest value from step 1
3. Use highest value to compute the following
4. If , then

Computation done in MATLAB using the function *DCMtoEP* made in CP01

Unit-norm computation was also done in MATLAB using the built-in *norm* function

Code:

|  |
| --- |
| % Euler (3-2-1) attitude at t=0  theta1 = -pi/4; % [rad]  theta2 = pi/8; % [rad]  theta3 = pi/5; % [rad]  % DCM attitude at t=0  C = EA321toDCM(theta1,theta2,theta3); % [-]  % Euler Parameter attitude at t=0  epsilon0 = DCMtoEP(C); % [-]  % Confirm unit-norm  disp(norm(epsilon0)) |

### Solution

|  |
| --- |
|  |

|  |
| --- |
| unit-norm is satisfied |

## (b)

### Find

Implement the EP’s kinematic differential equation as a function name *KDE\_EP,* which computes the time derivative of an EP vector given and .

Discuss which vector bases are used to express each of the vector quantities that appear in the function *KDE\_EP*.

### Analysis

Kinematic differential equation for Euler Parameters are as follows

### Solution

Written in MATLAB

|  |
| --- |
| function epsilon\_dot = KDE\_EP(epsilon,omega)  A = [0,omega(3),-omega(2),omega(1)  -omega(3),0,omega(1),omega(2)  omega(2),-omega(1),0,omega(3)  -omega(1),-omega(2),-omega(3),0];  epsilon\_dot = 0.5\*A\*epsilon;  end % function |

|  |
| --- |
| Note that and the vector components of epsilon can be written in either transformation frame or . |

## (c)

### Given

Where:

### Find

Compute the time history of by numerically integrating KDE\_EP over a time span from to seconds with an integration tolerance of .

Show the plots of over time in intervals of 0.5 seconds.

### Analysis

Numerically integrating the equation

With the given and the initial condition of

### Solution

|  |
| --- |
| %% Part a  % Euler (3-2-1) attitude at t=0  theta1 = -pi/4; % [rad]  theta2 = pi/8; % [rad]  theta3 = pi/5; % [rad]  % DCM attitude at t=0  C = EA321toDCM(theta1,theta2,theta3); % [-]  % Euler Parameter attitude at t=0  epsilon0 = DCMtoEP(C); % [-]  % Confirm unit-norm  disp(norm(epsilon0))  %% Part c  % time span  tspan = 0:0.5:100; % [s]  % integration options  opts = odeset("RelTol",1e-10,"AbsTol",1e-10);  % integration  [t,epsilon] = ode45(@odefun,tspan,epsilon0,opts);  % plots  figure(1)  subplot(2,2,1)  plot(t,epsilon(:,1),'b')  grid on  xlabel("Time [s]","Interpreter","latex")  ylabel("Euler Parameter $\epsilon\_1$","Interpreter","latex")  title("$\epsilon\_1(t)$","Interpreter","latex")  subplot(2,2,2)  plot(t,epsilon(:,2),'b')  grid on  xlabel("Time [s]","Interpreter","latex")  ylabel("Euler Parameter $\epsilon\_2$","Interpreter","latex")  title("$\epsilon\_2(t)$","Interpreter","latex")  subplot(2,2,3)  plot(t,epsilon(:,3),'b')  grid on  xlabel("Time [s]","Interpreter","latex")  ylabel("Euler Parameter $\epsilon\_3$","Interpreter","latex")  title("$\epsilon\_3$(t)","Interpreter","latex")  subplot(2,2,4)  plot(t,epsilon(:,4),'b')  grid on  xlabel("Time [s]","Interpreter","latex")  ylabel("Euler Parameter $\epsilon\_4$","Interpreter","latex")  title("$\epsilon\_4$(t)","Interpreter","latex") |

|  |
| --- |
| Chart, line chart  Description automatically generated |

## (d1)

### Given

Error metric

### Find

Compute the time history of .

Show the plots of over time.

Discuss how the values of behaves over time.

### Solution

|  |
| --- |
| % error metric  delta = vecnorm(epsilon,2,2) - 1; % [-]  % plot  figure(2)  plot(t,delta,'r')  grid on  xlabel("Time [s]","Interpreter","latex")  ylabel("Error Metric $\delta$","Interpreter","latex")  title("$\delta(t)$","Interpreter","latex") |

|  |
| --- |
| Chart, line chart  Description automatically generated |

|  |
| --- |
| The error starts of initially very low (in this case close to the integration tolerance of ) and increases in magnitude over time. This is due to the accumulation of errors in the numerical integration process and is resulting in the norm-constraint progressively being violated. |

## (d2)

### Given

Different integration tolerances

### Find

Compute the histories of over the same time span by numerical integration with the given tolerances.

Using the computed , show the plots of over time for each of the tolerances.

Discuss how these three histories of compare against each other

### Analysis

The same method to compute the time history of is used in [part c](#_(c)) and the same method is used to compute and plot the time history of is used in [part d1](#_(d1)).

### Solution

For tolerance of

|  |
| --- |
| %% Part d2 (1e-4)  % new integration options  opts = odeset("RelTol",1e-4,"AbsTol",1e-10);  % integration  [t,epsilon] = ode45(@odefun,tspan,epsilon0,opts);  % new error metric  delta = vecnorm(epsilon,2,2) - 1; % [-]  % new plot  figure(3)  plot(t,delta,'r')  grid on  xlabel("Time [s]","Interpreter","latex")  ylabel("Error Metric $\delta$","Interpreter","latex")  title("$\delta(t)$ with tolerance of $1\times10^{-4}$","Interpreter","latex") |

|  |
| --- |
| Chart, line chart  Description automatically generated |

For

|  |
| --- |
| %% Part d2 (1e-6)  % new integration options  opts = odeset("RelTol",1e-6,"AbsTol",1e-10);  % integration  [t,epsilon] = ode45(@odefun,tspan,epsilon0,opts);  % new error metric  delta = vecnorm(epsilon,2,2) - 1; % [-]  % new plot  figure(4)  plot(t,delta,'r')  grid on  xlabel("Time [s]","Interpreter","latex")  ylabel("Error Metric $\delta$","Interpreter","latex")  title("$\delta(t)$ with tolerance of $1\times10^{-6}$","Interpreter","latex") |

|  |
| --- |
| Chart, line chart  Description automatically generated |

For

|  |
| --- |
| %% Part d3 (1e-8)  % new integration options  opts = odeset("RelTol",1e-8,"AbsTol",1e-10);  % integration  [t,epsilon] = ode45(@odefun,tspan,epsilon0,opts);  % new error metric  delta = vecnorm(epsilon,2,2) - 1; % [-]  % new plot  figure(5)  plot(t,delta,'r')  grid on  xlabel("Time [s]","Interpreter","latex")  ylabel("Error Metric $\delta$","Interpreter","latex")  title("$\delta(t)$ with tolerance of $1\times10^{-8}$","Interpreter","latex") |

|  |
| --- |
| Chart, line chart  Description automatically generated |

|  |
| --- |
| The magnitude of the error metric increases as the integration tolerance increases. The greatest error factor was that of the “loosest” integration tolerance of and the lowest error factor was that of the “tightest” integration tolerance of with the lowest overall error being that of the original integration tolerance of . This coincides with the idea that the numerical inaccuracies increase the violate of the unit-norm constraint over time, with the rate of this violation being proportional to the integration tolerance. |

# Problem 2

## (a)

Chart, radar chart

Description automatically generated

## (b)

Chart

Description automatically generated

## (c)

## (d)

Chart

Description automatically generated

## (e)

A picture containing graphical user interface

Description automatically generated

## (f)