MAT401: Applied Mathematics 4
Coursework: free rotation of a torus

Dr Craig Stark

Term 2, 2017

MAT401: Applied Mathematics 4
Coursework: free rotation of a torus

Dr Craig Stark

Term 2, 2017

Overview

Module Code: MAT401

Unit of Assessment: Coursework (30% of Module Grade)

Submission Date: 21st March 2017

Marked and feedback given by: 4th April 2017

Submission requirements: An electronic copy of your source code, plots and movie of the rotating torus, in a zip file should be submitted as per the instructions on Blackboard. (Please don't use WinRAR).

1 Introduction

You are required to develop software to simulate the rigid body dynamics of a rotating torus (a ring) for a computer game application. In this project you will numerically solve Euler's equations for a torus in the special case where there are no external torques acting on the body and whose rotational motion takes place about its centre of mass.

Consider a torus with major radius c and minor radius a, that has uniform density ρ, mass M and participates in rotational motion about an axis passing through its centre of mass (i.e. its geometric centre) with angular velocity $\boldsymbol{\omega} = (\omega_x, \omega_y, \omega_z)$. The inertia tensor of a torus with major radius c, minor radius a and for a coordinate system centred at the centre of mass is given by

$$\mathsf{I} = \begin{bmatrix} M\left(\frac{5}{8}a^2 + \frac{1}{2}c^2\right) & 0 & 0 \\ 0 & M\left(\frac{5}{8}a^2 + \frac{1}{2}c^2\right) & 0 \\ 0 & 0 & M\left(\frac{3}{4}a^2 + c^2\right) \end{bmatrix}$$

Let's consider the special case where there are no external torques acting on the torus, $\sum \boldsymbol{\tau} = 0$. Therefore, using principal axes with an origin at the centre of mass, Euler's equations (lecture 7 of the course notes) become

$$\begin{aligned} \dot{\omega}_x &= -\gamma_1 \omega_y \omega_z, \\ \dot{\omega}_y &= -\gamma_2 \omega_x \omega_z, \\ \dot{\omega}_z &= -\gamma_3 \omega_x \omega_y, \end{aligned}$$

where $\gamma_1 = (I_3 - I_2)/I_1$, $\gamma_2 = (I_1 - I_3)/I_2$, $\gamma_3 = (I_2 - I_1)/I_3$ and I_i are the principal moments of inertia. These are a coupled set of ordinary differential equations that describe how the angular speed of rotation evolves with time, t, under no external torques. If you are unsure please check lecture 7 of the lecture notes on Euler's equations, their derivation and their underlying assumptions. Knowing the principal moments of inertia I_i for a torus and initial conditions for the angular speed, $\boldsymbol{\omega}_0$, the Runge-Kutta method can be used to solve these equations numerically in a way similar to how the coupled set of equations describing simple harmonic motion can be solved i.e. $v = \dot{x}$ and $\dot{v} = -kx/m$. The difference here being that we have three equations instead of just two, but the principle is the same.

2 Deliverables

For the assessment you should:

1. For a torus of mass $M = 3.14$ kg, major radius $c = 4$ m and minor radius $a = 1$ m, and as initial conditions $\boldsymbol{\omega}_0 = (1,1,1)$ rad s^{-1}, numerically solve Euler's equations using your own bespoke fourth-order Runge-Kutta algorithm. Solve Euler's equations for the case when there are no external torques acting on the torus, whose rotational motion takes place about an axis passing through its centre of mass and for a time period of $t \in [0,16]$ s. Note that you can use any suitable programming language you prefer, such as MATLAB, C++, etc.

2. From your solutions, produce plots of $\omega_x(t)$, $\omega_y(t)$ and $\omega_z(t)$ as a function time, t, showing the periodic evolution of the angular speed for the toroidal system.

3. Produce a 3D plot of ω_x versus ω_y versus ω_z for all times t, such that your plot will show a 3D trace of $\boldsymbol{\omega}$ in time.

4. Recall that the position of a general point P in our rigid body is given by $\mathbf{r} = \Lambda \mathbf{r}'_0$, where $\mathbf{r}'_0$ is the initial position vector of P relative to the centre of mass; and Λ is the standard rotation transformation matrix (for details see lecture 8 on general motion of a rigid body). Using this expression for $\mathbf{r}$ and the $\boldsymbol{\omega}$ solution obtained from your simulations, extend your code to produce a 3D animation of the rotating torus. For the initial position of the points on the surface of the torus, $\mathbf{r}'_0$, you may use the parametric equations of a torus with major radius c and minor radius a given by

$$\begin{aligned} x &= (c + a\cos v)\cos u \\ y &= (c + a\cos v)\sin u \\ z &= a\sin v \end{aligned}$$

   for $u,\ v \in [0, 2\pi)$. See Figure 1.

For an example of the expected form of your output, see below for sample results (Figures 2 and 3) for the case of a rotating ellipsoid with semi-axes $a = 3$ m, $b = 2$ m and $c = 1$ m, mass $M = 3.14$ kg; and for initial conditions $\boldsymbol{\omega}_0 = (1,1,1)$ rad s^{-1}.

Submission requirements: An electronic copy of your source code, plots and movie of the rotating torus, in a zip file should be submitted as per the instructions on Blackboard. (Please don't use WinRAR). The deadline for submission is Tuesday the 21st of March 2017 and the assessment will be marked and feedback returned by Tuesday 4th of April 2017.

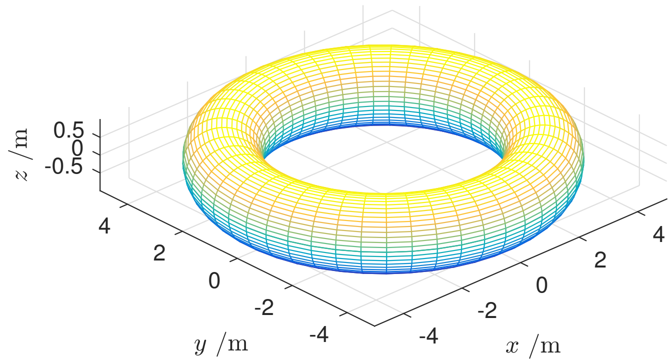

Figure 1: A torus of mass $M = 3.14$ kg, major radius $c = 4$ m and minor radius $a = 1$ m. This plot is generated using the parametric equations of a torus.

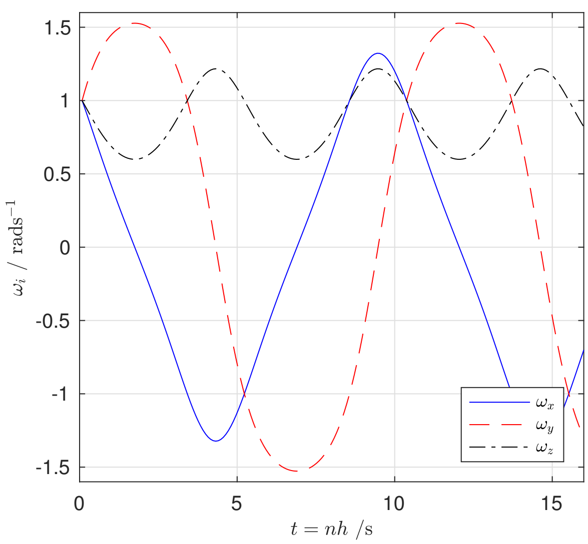

Figure 2: $\omega_x(t)$ (blue solid line), $\omega_y(t)$ (red dashed line) and $\omega_z(t)$ (black dot-dash line) as a function time, t, showing the periodic evolution of the angular speed for the ellipsoidal system.

3 Grading

Your bespoke numerical code (for solving Euler's equations and producing your animation) should be coded efficiently and properly commented. You may be required to demonstrate your code at a later date. *The numerical code will account for* 60% *of your grade.* The results produced from your code (plots & movie) should be clear, correctly labelled and with appropriate units on the axes where required. *Your results will account for* 40% *of your grade.* Your grade for the coursework will count as 30% towards the overall grade for the module.

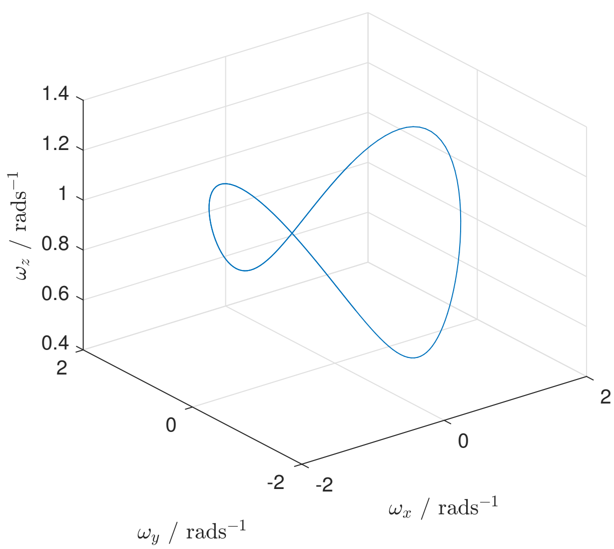

Figure 3: ω_x versus ω_y versus ω_z for all times t for the ellipsoidal system.

MAT401 – Coursework Grading Criteria

	A	B	C	D	MF	F	NS
RK implementation & code structure (60%)	Excellent overall. Demonstrates an excellent understanding of the RK method applied to coupled ODEs. Exhibits excellent appreciation of the general motion of a rigid body and its numerical simulation. A very efficiently written code, well commented and easy to follow.	Very good overall. Demonstrates a very good understanding of the RK method applied to coupled ODEs. Exhibits very good appreciation of the general motion of a rigid body and its numerical simulation. An efficiently written code, well commented and easy to follow, with only minor flaws.	Good overall. Demonstrates a good understanding of the RK method applied to coupled ODEs. Exhibits good appreciation of the general motion of a rigid body and its numerical simulation. A reasonably efficient code, with comments and moderately easy to follow, with some flaws and omissions.	Satisfactory overall. Demonstrates a satisfactorily understanding of the RK method applied to coupled ODEs. Exhibits satisfactory appreciation of the general motion of a rigid body and its numerical simulation. An adequately efficient code, with some comments but hard to follow ,with several flaws and significant omissions.	Marginal Fail. Some understanding of the RK method applied to coupled ODEs. Exhibits some appreciation of the general motion of a rigid body and its numerical simulation. Poorly written code, inefficient, little commenting and very hard to follow.	Fail. Poor understanding of the RK method applied to coupled ODEs. Exhibits poor appreciation of the general motion of a rigid body and its numerical simulation. Unacceptably written code, inefficient, no commenting and impossible to follow.	Nothing submitted by student.
Results & presentation (40%)	Excellent results. Correct, very clearly plotted, correctly labelled axes with units and legend, where appropriate.	Very good results. Correct, clearly plotted, correctly labelled axes with units and legend, where appropriate. Some minor flaws or omissions	Good results. Reasonable plots, labelled axes, units and legend. Some flaws and omissions	Satisfactory results. Satisfactory plots, labelled axes, units and legend. Several flaws, significant omissions.	Poor results. A few results with poor plots, little labelling and a disregard for units.	Fail. Inadequate results, incorrectly plotted , with no labels, units or legends.	Nothing submitted by student.

Please note that the percentage weightings for each criterion are approximate.