F.7 Numerical Solution of Satellite Equations of Motion

The equations of motion are stated in (F.5.8) which we replicate here.

$$1 \quad \dot{v}_\theta + \sin\theta\cos\theta(3\omega^2 - \dot{\varphi}^2) + \omega\sin\theta\cos\varphi\,(\omega\cos\theta\cos\varphi - 2\sin\theta\,\dot{\varphi}) = 0 \qquad // \hat{\boldsymbol{\varphi}}$$

$$2 \quad \dot{v}_\varphi + 2\cot\theta\,\dot{\theta}\,\dot{\varphi} - \omega\cos\varphi\,(\omega\sin\varphi - 2\dot{\theta}) = 0 \qquad // \hat{\boldsymbol{\theta}} \qquad \text{(F.5.7)}$$

$$3 \quad \dot{\theta} = v_\theta$$

$$4 \quad \dot{\varphi} = v_\varphi \qquad \text{(F.5.8)}$$

To keep our Maple code compact, we make these name changes:

θ	$\leftrightarrow$	T	Theta
φ	$\leftrightarrow$	P	Phi
v_θ	$\leftrightarrow$	vT	$v_{\mathbf{Theta}}$
v_φ	$\leftrightarrow$	vP	$v_{\mathbf{Phi}}$
ω	$\leftrightarrow$	w	

The first step is to enter the four equations and other setup data including $\omega = 1$:

```
restart; with(plots):
eq1 := diff(vT(t),t)+sin(T(t))*cos(T(t))*(3*w^2 -vP(t)^2) +
w*sin(T(t))*cos(P(t))*(w*cos(T(t))*cos(P(t))-2*sin(T(t))*vP(t))=0;
```

$$eq1 := \left(\frac{\partial}{\partial t}\mathrm{vT}(t)\right) + \sin(\mathrm{T}(t))\cos(\mathrm{T}(t))\,(3w^2 - \mathrm{vP}(t)^2) + w\sin(\mathrm{T}(t))\cos(\mathrm{P}(t))\,(w\cos(\mathrm{T}(t))\cos(\mathrm{P}(t)) - 2\sin(\mathrm{T}(t))\,\mathrm{vP}(t)) = 0$$

```
eq2 := diff(vP(t),t)+2*cot(T(t))*vT(t)*vP(t)-w*cos(P(t))*(w*sin(P(t))-2*vT(t))=0;
```

$$eq2 := \left(\frac{\partial}{\partial t}\mathrm{vP}(t)\right) + 2\cot(\mathrm{T}(t))\,\mathrm{vT}(t)\,\mathrm{vP}(t) - w\cos(\mathrm{P}(t))\,(w\sin(\mathrm{P}(t)) - 2\,\mathrm{vT}(t)) = 0$$

```
eq3 := vT(t)=Diff(T(t),t);
```

$$eq3 := \mathrm{vT}(t) = \frac{\partial}{\partial t}\mathrm{T}(t)$$

```
eq4 := vP(t)=Diff(P(t),t);
```

$$eq4 := \mathrm{vP}(t) = \frac{\partial}{\partial t}\mathrm{P}(t)$$

```
w := 1:
eqs := {eq1,eq2,eq3,eq4}:
funcs := {T(t),P(t),vT(t),vP(t)}:
```

For each solution run we must select some initial conditions. Notice that we have set

$$\omega = 1 \qquad \Rightarrow \qquad T = 2\pi = 6.28 \text{ seconds}$$

<u>Verification of in-plane libration</u>

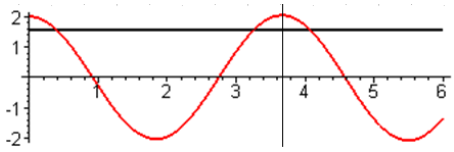
The initial conditions are taken to be

$\theta = 0.2$ $\dot{\theta} = 0$ $\varphi = \pi/2$ $\dot{\varphi} = 0$.

Here is a plot of the result where θ(t) is magnified by a factor of 10:

```
Demonstration of in-plane oscillation
inits := {T(0)=0.2,vT(0)=0,P(0)= Pi/2,vP(0)= 0 }:
f := dsolve(eqs union inits,funcs,type=numeric,method=classical,output=listprocedure):
s1 := odeplot(f,[t,10*T(t)],0..6,numpoints=1500,color=red,thickness=2):
s2 := odeplot(f,[t,P(t)],0..6,numpoints=1500,color=black,thickness=2):

display(s1,s2);
```



The azimuth φ(t) holds at π/2 while θ(t) oscillates. From (F.5.10) for in-plane libration one predicts,

$$T_{ocs1} = T/\sqrt{3} = 6.28/1.73 = 3.63 \text{ sec}$$

which is verified by the vertical line in the figure.

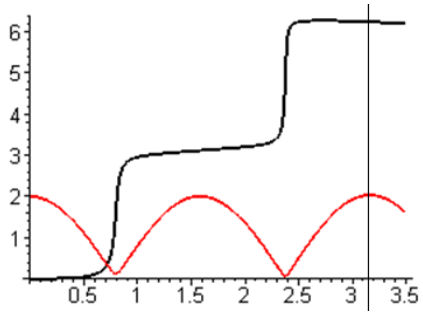
Verification of out-of-plane libration

The initial conditions are taken to be

$\theta = 0.2$ $\dot{\theta} = 0$ $\varphi = 0$ $\dot{\varphi} = 0$.

Here is a plot of the result where again θ(t) is magnified by a factor of 10:

```
Demonstration of out-of-plane oscillation
inits := {T(0)=0.2,vT(0)=0,P(0)= 0,vP(0)= 0 }:
```



Maple has chosen to show the oscillation by keeping θ(t) positive and having φ(t) jump by π at each zero crossing. This is just another way to represent a cosine function. As suggested below (F.5.13), one cannot maintain φ(t) = 0 for the out-of-plane case and one sees φ(t) creeping up from 0 around t = 0.5 sec.

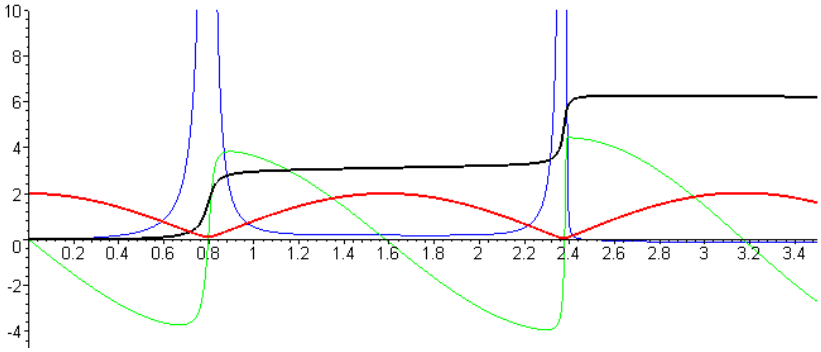
From (F.5.13) for out-of-plane libration one predicts,

$$T_{ocs2} = T/2 = 6.28/2 = 3.14$$

which is verified by the vertical line in the figure.

Maple can also plot $\dot{\theta}(t)$ and $\dot{\varphi}(t)$ and here is a plot showing all four curves $\theta,\varphi,\dot{\theta},\dot{\varphi}$ = red,black,green,blue:

```
s3 := odeplot(f,[t,10*vT(t)],0..3.5,numpoints=1000,color=green):
s4 := odeplot(f,[t,vP(t)],0..3.5,numpoints=1000,color=blue):
display(s1,s2,s3,s4,view = [0..3.5,-5..10]);
```



This plot hints at a technical problem with the ODE system of equations as entered above. The second equation contains a factor cot(θ) which blows up at θ = 0,

$$\dot{v}_{\varphi} + 2\cot\theta\,\dot{\theta}\,\dot{\varphi} - \omega\cos\varphi\,(\omega\sin\varphi - 2\dot{\theta}) = 0,$$

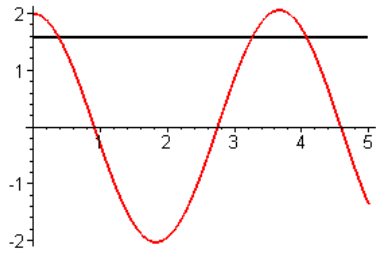
so whenever the red θ(t) curve approaches 0, one sees $\dot{v}_{\varphi}$ getting very large (the slope of the blue curve $\dot{\varphi} = v_{\varphi}$). Unless $\dot{\varphi} = 0$, the solution cannot pass through the θ = 0 value. The problem is that θ = 0 is a singular point in spherical coordinates where azimuth φ is undefined. One solution to this problem might be to convert the differential equations from spherical coordinates θ and φ to Cartesian coordinates x,y,z where

$$x = r\sin\theta\cos\varphi$$
$$y = r\sin\theta\sin\varphi$$
$$z = r\cos\theta \qquad (E.2.5)$$

A more general solution

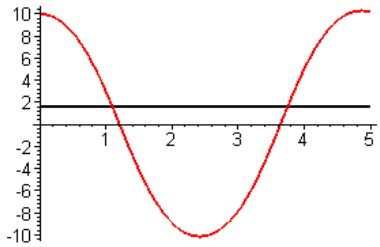
so θ is a small angle. Here then is the solution run and a plot ofθ(t) in red and φ(t) in black (θ is magnified by a factor of 10),

```
inits := {T(0)=0.2,vT(0)=0,P(0)= Pi/2,vP(0)= 0 }:
f := dsolve(eqs union inits,funcs,type=numeric,method=classical,output=listprocedure):
s1 := odeplot(f,[t,10*T(t)],0..5,numpoints=1000,color=red,thickness=2):
s2 := odeplot(f,[t,P(t)],0..5,numpoints=1000,color=black,thickness=2):
display(s1,s2);
```



One can see that the period of the small-θ oscillation is about 3.7 seconds so $\omega_{osc} = 2\pi/3.7 = 1.7$. This is in agreement with (F.5.10) which says $\omega_{osc1} = \sqrt{3}\ \omega = 1.7 * 1 = 1{,}7$ for in-plane libration. If θ = 1 radian, so the polar angle is no longer small, the period gets longer :

```
inits := {T(0)=1,vT(0)=0,P(0)= Pi/2,vP(0)= 0 }:
```



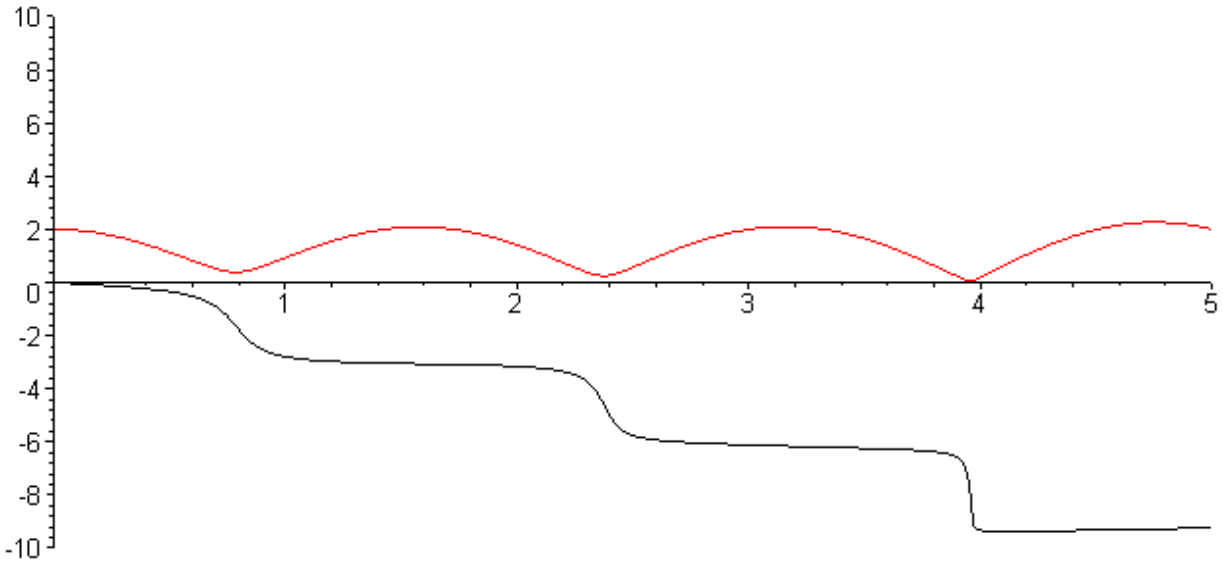
as happens in a spherical pendulum.

```
w := 1:
eqs := {eq1,eq2,eq3,eq4}:
inits := {T(0)=0.2,vT(0)=0,P(0)= 0,vP(0)= -1 }:
funcs := {T(t),P(t),vT(t),vP(t)}:
f := dsolve(eqs union inits,funcs,type=numeric,method=classical,output=listprocedure);
```

$$f := [t = (\mathbf{proc}(t) \ \ldots\ \mathbf{end}), P(t) = (\mathbf{proc}(t) \ \ldots\ \mathbf{end}), vP(t) = (\mathbf{proc}(t) \ \ldots\ \mathbf{end}), vT(t) = (\mathbf{proc}(t) \ \ldots\ \mathbf{end}), T(t) = (\mathbf{proc}(t) \ \ldots\ \mathbf{end})]$$

Here we plot 30*θ(t) and φ(t) on the same graph (we magnify θ(t) in order to see both plots),

```
s1 := odeplot(f,[t,30*T(t)],0..5,numpoints=1000,color=red):
s2 := odeplot(f,[t,P(t)],0..5,numpoints=1000,color=black):
display(s1,s2,view = [0..5,-10..10]);
```



The motion of this example is quite complicated: the polar angle θ(t) of the dumbbell oscillates back and forth without reaching 0 while the dumbbell rotates around the vertical axis in azimuth φ since we started it with $\dot{\varphi} = -1$, so the φ just winds around in the negative direction. Another way to plot this motion is as a trajectory in θ-φ space,

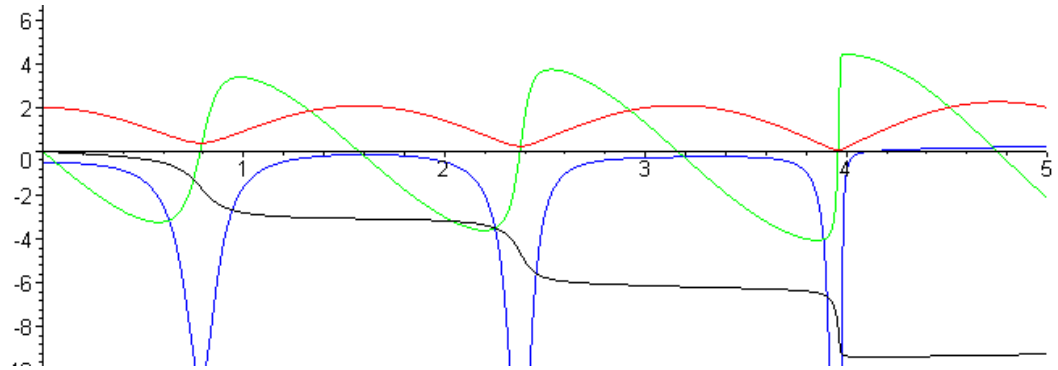
```
odeplot(f,[10*T(t),P(t)],0..5,numpoints=1000,color=red);
```



In this graph, mass m_1 starts at the top right and then moves along the curve. Each descent of π takes the satellite halfway around the vertical axis.

The azimuthal velocity is quite uneven as energy shifts between the two degrees of freedom. Here we add the velocity $\dot{\theta}$ in green and $\dot{\varphi}$ in blue to our original plot ** to get

```
s3 := odeplot(f,[t,10*vT(t)],0..5,numpoints=1000,color=green):
s4 := odeplot(f,[t,vP(t)],0..5,numpoints=1000,color=blue):
display(s1,s2,s3,s4,view = [0..5,-10..10]);
```



As the red curve gets close to $\theta = 0$, the $\cot\theta$ in equation #2 gets large, forcing $\dot{v}_\varphi$ to be large as indicated by the large negative blue peaks. The simulation is very non-robust and dies if the point $\theta = 0$ is hit since then $\cot\theta = \infty$. This is unfortunate, since we would like to plot the in-plane and out-of-plane oscillations which should pass through $\theta = 0$.

The problem of course is that $\theta = 0$ is a singular point of the spherical coordinate system where the value of φ is undefined, so our simple simulation not surprisingly has problems at $\theta = 0$. One could fix this problem by changing from θ,φ to x,y.z coordinates, but we leave that task to the reader.

$$x = r_1 \sin\theta\cos\varphi$$
$$y = r_1 \sin\theta\sin\varphi$$
$$z = r_1 \cos\theta$$

$$\dot{x} = r_1 \cos\theta\dot{\theta}\cos\varphi - r_1 \sin\theta \sin\varphi \dot{\varphi}$$
$$\dot{y} = r_1 \cos\theta\dot{\theta}\sin\varphi + r_1 \sin\theta \cos\varphi \dot{\varphi}$$
$$\dot{z} = -r_1 \sin\theta\dot{\theta}$$

The motion is a bit difficult to visualize. From t = 0 to t = 1 $\theta(t)$ drops down from its initial 0.2 to some small $\theta_{\mathbf{min}}$ while φ moves from 0 to $-\pi$, so the dumbbell has gone half way around the vertical axis.

The lower dumbbell mass moves somewhat like a spherical pendulum (while the upper mass as an upside down spherical pendulum). The angle $\theta(t)$ shows cosine-like behavior between some $\theta_{\mathbf{max}}$ and $\theta_{\mathbf{min}}$, while the azimuthal coordinate φ just winds around the vertical axis roughly once per oscillation of θ. One can see that the period of the oscillation is about 1.6 seconds so $\omega_{\mathbf{osc}} \approx 2\pi/1.6 \approx 4$.

For this example the dumbbell initial conditions are:

$\theta = 0.2 \qquad \dot{\theta} = 0$

$\varphi = 0 \qquad \dot{\varphi} = -1$

The dumbbell oscillates in θ while it rotates in azimuth in the $-\hat{\varphi}$ direction.

Next we group the four equations into set "eqs", and put initial conditions into "init", and set ω = 1:

```
eqs := {eq1,eq2,eq3,eq4}:
init := {T(0)=0.2,vT(0)=0,P(0)= 0,vP(0)= -1 }:
```

The call to solve the system of equations is then,

```
f := dsolve(eqs union
init,{T(t),P(t),vT(t),vP(t)},type=numeric,method=classical,output=listprocedure);
```

$$f := [t = (\mathbf{proc}(t) \ \ldots \ \mathbf{end}), P(t) = (\mathbf{proc}(t) \ \ldots \ \mathbf{end}), vP(t) = (\mathbf{proc}(t) \ \ldots \ \mathbf{end}), vT(t) = (\mathbf{proc}(t) \ \ldots \ \mathbf{end}), T(t) = (\mathbf{proc}(t) \ \ldots \ \mathbf{end})]$$

We then plot T = θ and P = φ for t =

```
odeplot(f,[[t,30*T(t)],[t,P(t)]],0..5,numpoints=1000);
```

