



C7 Constrained Pendulum – Maple 8

Symbolic/Numerical Approach

Simulator: Maple is a computer algebra system which is mainly used for symbolic calculation but it also includes numerical features. It provides a vast library of built-in functions and operations and allows arbitrary high accuracy by making it possible to change the numbers of digits carried in floats.

Model: In this solution we test the state event detection mechanism in the numerical differential equation solvers in Maple. Not all of the implemented algorithms provide a state event finder. For nonstiff problems the Runge-Kutta-Fehlberg method of order 4(5) and for stiff problems a Rosenbrock method give the possibility to formulate stop conditions. These conditions can be used in two different ways. The first is to give expressions as stop condition, where the integration halts once the expression evaluates to zero. Second, one can provide range conditions where the integration once the value of an expression steps outside a given range. The second possibility is useful when one has to start a new integration beginning from the last event to avoid timestepping in the last event.

Task a: Simulation in the time domain: For the simulation of the motion of the pendulum we define two differential equations. One for the pendulum with length l and one for length ls . As stop condition we define an expression which evaluates to zero when the angle is equal to ϕ_p . When a state event occurs we switch between the model equations for the long and short pendulum.

```
dglsys_long:={diff(phi1(t),t)=phi2(t),
diff(phi2(t),t)=-g/l*sin(phi1(t))-d/m*phi2(t)};
dglsys_short:={diff(phi1(t),t)=phi2(t),
diff(phi2(t),t)=..};
stopcond:={phi1(t)=Pi/12};
sol_long:=dsolve(dglsys_long union initcond,
...,type=numeric,stop_cond=stopcond);
sol_short:=dsolve(dglsys_long union initcond,
...,type=numeric,stop_cond=stopcond);
```

Above code is packed into a loop with a function giving the solution at timepoint t gives the result (fig. 1):

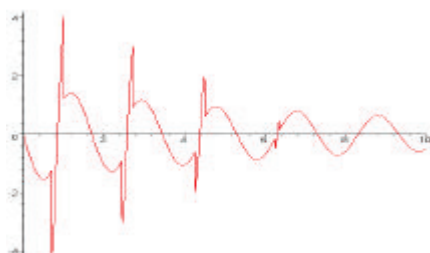


Fig. 1: Angle velocity with events, task a)

Task b: Comparison of linear and nonlinear model. The treatment of the linear model is a straightforward task in Maple. The linear system can be solved explicitly.

```
linsys_long:={diff(phi1(t),t)=phi2(t),
diff(phi2(t),t)=-g/l*phi1(t)-d/m*phi2(t)};
sol_long:=dsolve(linsys_long union
{phi1(t0)=phi10,
phi2(t0)=phi20},{phi1(t),phi2(t)});
```

The same can be done for the short pendulum. The absolute error between the linear equations and nonlinear results are shown in the picture below.

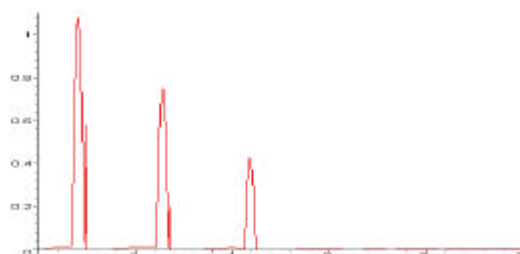


Fig. 2: Differences of nonlinear and linear model

In the following table we show the timepoints of the state events. Right column gives the timepoints calculated with the nonlinear model, left column the timepoints in the linear case.

Nonlinear, [s]	Linear, [s]
0.69337931	0.69202334
1.127779	1.12054473
2.5472503	2.540860458
2.9404404	2.931799150
4.4837766	4.464755483
4.8095060	4.790786113
6.5506487	6.532057557
6.6794029	6.652995455

Task c: Boundary value problem. We transform the problem into an initial value problem with reverse time.

The initial conditions are then $\phi = -\pi/2$ and angular velocity equal to zero. With help of the Newton procedure we calculate an initial angular velocity of -2.184701221 Units.

```
phiinit:=proc(t)evalf(reverse(t)-Pi/6);
Solve(phiinit,t=0.5)
```

Michael Wibmer
mwibmer@osiris.tuiwen.ac.at

