MCS 320 Project Two : Celestial Mechanics a model of the n-body problem

The goal of this project is to use Maple to build a computational model of the n-body problem from celestial mechanics. This document is based on a Maple worksheet, available for downloading from the course web site.

1. The Differential Equations

In this section we define the differential equations for the planar n-body problem and set up an initial value problem suitable to solve it numerically via `dsolve`.

The distance between two points is defined by the function `distance`. The function `isum` is used to sum over `n` elements, excluding the `i`-th entry. The functions `fx` and `fy` will be used to define the right hand sides of the equations for the `x` and `y` coordinates. Then the equations are generated by calls to the functions `eqx` and `eqy`. Finally, with `sys`, we generate the system of differential equations.

```
[> distance := (p,q) -> sqrt((p[1]-q[1])^2+(p[2]-q[2])^2):
[> isum := (i,n,a) -> sum(a(i,'j'),j=1..(i-1)) + sum(a(i,'j'),j=(i+1)..n):
[> fx := (i,j) -> m[i]*m[j]*(x[i](t)-x[j](t))/distance([x[i](t),y[i](t)],[x[j](t),y[j](t)])^3:
[> fy := (i,j) -> m[i]*m[j]*(y[i](t)-y[j](t))/distance([x[i](t),y[i](t)],[x[j](t),y[j](t)])^3:
[> eqx := (i,n) -> m[i]*diff(x[i](t),t$2) = -isum(i,n,fx):
[> eqy := (i,n) -> m[i]*diff(y[i](t),t$2) = -isum(i,n,fy):
[> sys := n -> seq(op([eqx(k,n),eqy(k,n)]),k=1..n):
[> s3 := sys(3):
```

The variable `s3` contains the symbolic equations which govern the motion of a 3-body problem. To solve this problem numerically, we pick values for the masses `m[1]`, `m[2]`, `m[3]`, of the three bodies, specify their initial positions respectively in `ip1`, `ip2`, `ip3`, respectively in `v1`, `v2`, `v3`, which results in initial conditions stored in `ic1`, `ic2`, and `ic3`.

```
[> m[1] := 1.0: m[2] := 0.5: m[3] := 0.2:
[> ip1 := [1, 0]: ip2 := [-.5, .86]: ip3 := [-.5, -.86]:
[> v1 := [-2.0, 0]: v2 := [-2, -2]: v3 := [-2, 2]:
[> ic1 := x[1](0) = ip1[1], y[1](0) = ip1[2], D(x[1])(0) = v1[1], D(y[1])(0) = v1[2]:
[> ic2 := x[2](0) = ip2[1], y[2](0) = ip2[2], D(x[2])(0) = v2[1], D(y[2])(0) = v2[2]:
[> ic3 := x[3](0) = ip3[1], y[3](0) = ip3[2], D(x[3])(0) = v3[1], D(y[3])(0) = v3[2]:
[> ic := ic1,ic2,ic3:
```

Then the initial value problem is defined by the system of differential equations and the initial conditions, grouped in the list `ivp`. The `ivp` will be used as the first argument in `dsolve`. The second argument for dsolve is a list of variables, which consists of all the coordinate functions, collected in the list `var`.

```
[> ivp := [s3,ic]:
[> var := [x[1](t),y[1](t),x[2](t),y[2](t),x[3](t),y[3](t)]:
[> sol := dsolve(ivp, var, numeric):
```

With the call to `sol(0.0)`, we verify the initial conditions. To know the values at `t = 1.0`, we execute `sol(1.0)`.

2. Plotting the Trajectories

To plot the trajectories, we define auxiliary functions which return the values for each point.

```
[> f1px := t -> rhs(sol(t)[2]): f1py := t -> rhs(sol(t)[4]):
[> f2px := t -> rhs(sol(t)[6]): f2py := t -> rhs(sol(t)[8]):
[> f3px := t -> rhs(sol(t)[10]): f3py := t -> rhs(sol(t)[12]):
```

We can now evaluate the coordinate functions and plot the trajectories:

```
[> t1 := plot([f1px,f1py,0..1.0],color=red):
[> t2 := plot([f2px,f2py,0..1.0],color=blue):
[> t3 := plot([f3px,f3py,0..1.0],color=black):
[> plots[display](t1,t2,t3);
```

To animate the trajectories, we make a movie:

```
[> N := 20:     # number of frames
[> T := 0.0:    # initial time
[> step := 0.1: # time step
[> movie := []:
[> for i from 1 to N do
[>    p1 := [f1px(T),f1py(T)]:
[>    p2 := [f2px(T),f2py(T)]:
[>    p3 := [f3px(T),f3py(T)]:
[>    d1 := plottools[disk](p1,0.1,color=red):
[>    d2 := plottools[disk](p2,0.1,color=blue):
[>    d3 := plottools[disk](p3,0.1,color=yellow):
[>    scene := plots[display](d1,d2,d3):
[>    movie := [op(movie),scene]:
[>    T := T + step:
[> end do:
[> plots[display](movie,insequence=true);
```

After clicking on the picture we can animate it using the toolbar.

Assignment One: Generalize to 4 Bodies

The example is set up to work with three bodies. The goal of the first assignment is to generalize the model to four bodies.

Assignment One. Give all Maple commands to generalize the model to four bodies. Choose some good initial values to make a nice plot of the trajectories.

Assignment Two: Make a Spatial Model

Extend the model to three dimensions. To keep the complexity of the computations a bit under control, you may still work with three bodies, i.e.: solve the two assignments independently of each other.

Assignment Two. Give all Maple commands to make a spatial model of the n-body problem. To plot the trajectories, use `tubeplot`. In the animated movie, use `sphere` instead of `disk`.

Assignment Three: a Stable Figure Eight Configuration

Take a look at the online paper [2] and check out the links this paper makes. The goal of this assignment is to make the figure eight configuration which has been used in space missions.

The deadline is Monday 9 April 2007 at 10AM

Bring *your* solution to the project to class. The *your* is emphasized to stress that your solution is the result of an *individual* effort. Collaborations are **not** permitted.

The solution to this project consists in two parts:

1. A print out of the Maple worksheet that you bring to class.

   Deliver a well written document, with grammatically correct and complete sentences, without spelling mistakes, appropriately structured into sections and subsections.

2. The Maple worksheet that you email as an attachment to me.

   The worksheet should run as a computer program, from top to bottom with consistent output and without errors.

   Please remove all output from the worksheet, so as to reduce its size.

If you have questions or difficulties with the assignments, feel free to come to my office for help.

References

[1] Donald G. Saari. *Collisions, Rings, and Other Newtonian N-body Problems.* Number 104 of CBMS Regional Conference Series in Mathematics, A.M.S., 2005.

[2] Ivars Petersen. Strange Orbits. *Science News* 168(7), 2005.
`http://www.sciencenews.org/articles/20050813/mathtrek.asp`