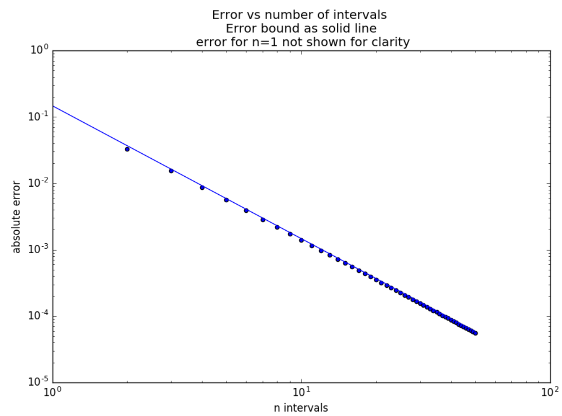
APMTH 205 HW 3
Louis Baum
October 20, 2016

Problem 1

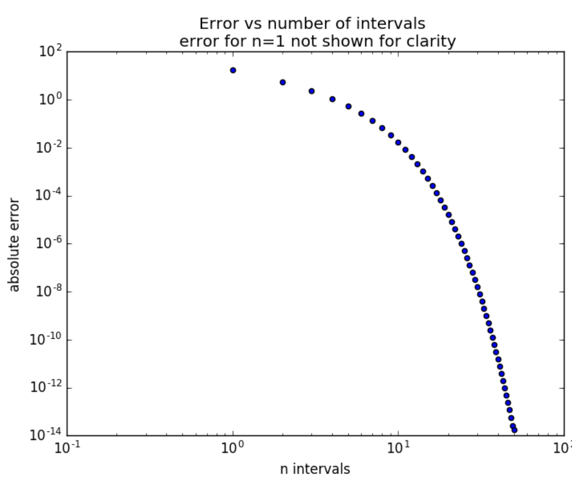
a)

The program "prob1a()" is located in the attached python code. As you can see from the figure, the absolute error is close to but below the limit derived in class. Please note that I do not plot the error when n = 1 because the absolute error is 0 and log(0) is undefined.



b)

The program "prob1b()" is located in the attached python code. As you can see from the plot the error is not linear in n on a log-log plot. Therefor the error does not scale with h^m for any value of m



Problem 2

a)

We want to show that the following is exact if f(x) is a polynomial of degree 3 :

$$\int_{-1}^{1} f(x)dx = \sum_{i=0}^{n} w_i f(x_i)$$

Let us define $q(x) = a + bx + cx^2 + dx^3$ a polynomial of degree 3.
if

$$\int_{-1}^{1} q(x)x^k dx = 0$$

for k = 0,1,2. The roots of q(x) are our values of x_i

$$\int_{-1}^{1} q(x)dx = \int_{-1}^{1} q(x)xdx = \int_{-1}^{1} q(x)x^2 dx = 0$$

if we do these integrals we obtain the following relations
$2a + \frac{2}{3}c = \frac{2}{3}b + \frac{2}{5}d = \frac{2}{3}a + \frac{2}{5}c = 0$
which when solved. we obtain
$q(x) = P_3(x) = \frac{1}{2}x(5x^2 - 3)$
The zeros of P_3 are our values of x_i. These occur at x = $-\sqrt{\frac{3}{5}}$, 0, $\sqrt{\frac{3}{5}}$

We now will find the weights of these points w_i
We know that the expression below is exact for f(x) = $1, x, x^2, x^3, x^3, x^5$

$$\int_{-1}^{1} f(x)dx = w_1 f(-\sqrt{\frac{3}{5}}) + w_2 f(0) + w_3 f(\sqrt{\frac{3}{5}})$$

to determine w_1, w_2, w_3 we evaluate the expression such that f(x)= 1, f(x) =x, f(x) = x^2 to get three equations to solve for 3 unknowns.

$$\int_{-1}^{1} 1dx = 2 = w_1 + w_2 + w_3$$

$$\int_{-1}^{1} xdx = 0 = -\sqrt{\frac{3}{5}}w_1 + \sqrt{\frac{3}{5}}w_3$$

$$\int_{-1}^{1} x^2 dx = \frac{2}{3} = \frac{3}{5}w_1 + \frac{3}{5}w_3$$

this system of equations is satisfied when
$w_1 = \frac{5}{9}$, $w_2 = \frac{8}{9}$, $w_3 = \frac{5}{9}$
To demonstrate that this is in fact exact for polynomials of degree 5 we will solve the following expression when f(x) is a general polynomial of degree 5:

$$\int_{-1}^{1} f(x)dx = \sum_{i=0}^{n} w_i f(x_i)$$

$$\begin{aligned}
\int_{-1}^{1} a + bx + cx^2 + dx^3 + ex^4 + fx^5 dx &= \frac{10}{9}(a + c\frac{3}{5} + e\frac{9}{25}) + \frac{8}{9}a \\
2a + \frac{2c}{3} + \frac{2e}{5} &= \frac{10}{9}(a + c\frac{3}{5} + e\frac{9}{25}) + \frac{8}{9}a \\
2a + \frac{2c}{3} + \frac{2e}{5} &= \frac{18}{9}a + \frac{6}{9}c + \frac{2}{5}e \\
2a + \frac{2c}{3} + \frac{2e}{5} &= 2a + \frac{2c}{3} + \frac{2e}{5}
\end{aligned}$$

Thus the quadrature rule integrates polynomials of degree 5 exactly.

b)

For this problem we need to convert the integral into the form that we can use the Gaussian quadriture rule:

$$\int_{a}^{b} f(x)dx = \frac{b-a}{2}\int_{-1}^{1} f(u)du = \frac{b-a}{2}\sum_{i=0}^{n} w_i f(u_i)$$

Since we know the values of u to evaluate the function via the Gaussian quadrature (zeros of P_3), we can back out the values of x to evaluate f(x) to get f(u). with f(u) and w_i (part a) we can determine the integral.

$$integrate(a,b) = \int_{a}^{b} f(x)dx$$

I then implemented "adaptive(a,b,tol)"" which evaluates integral(a,b) and compares it to the sum of integral(a,c) and integral(c,b). If the difference is larger than the tolerance defined in the problem statement adaptive(a,b,tol) recursively calls itself after splitting the integral in half.
I then calculate the integral for the following values of m to obtain the following table.

m	integral	error	intervals
4	1.5235026	0	1
5	1.13903809	0	1
6	1.43796787	$6.74\ 10^{-8}$	8
7	1.16355584	$2.08\ 10^{-7}$	8
8	1.39549466	$8.44\ 10^{-7}$	7

c)

For this Section I have modified F(x) directly in the code and used the same functions I implemented in problem 2b.(I will have a commented out version in the code if you care to look at it.)

$$\int_{-1}^{1} |x| dx = 1; \text{ error } = 0 \text{ with 2 intervals}$$

$$\int_{-1}^{2} |x| dx = 2.5; \text{ error } = 7.2\ 10^{-11} \text{ with 16 intervals}$$

$$\int_{-1}^{1} (500x^6 - 700x^4 + 245x^2 - 3)\sin^2(2\pi x) dx = 3.08\ 10^{-14}$$

$$\text{error} = 2.4\ 10^{-14} \text{ with 1 interval}$$

$$\text{(somthing is fishy)}$$

$$\int_{0}^{1} x^{3/4} \sin\frac{1}{x} dx = .407; \text{error} = 2.13\ 10^{-7} \text{ with 194326 intervals}$$

d)

In part c this integral is incorrect.

$$\int_{-1}^{1} (500x^6 - 700x^4 + 245x^2 - 3)\sin^2(2\pi x) dx \neq 3.08 * 10^{-14}$$

As it turns out the function we are attempting to integrate is approximately zero at the points that we are sampling. Not only that but once we split the interval in half, We find that the function is also close to zero for the second batch of sample points. Part of the reason that this occurs is that the function is even. Since the function is even we only sample 6 different values of f(x). we sample 3 for the original interval. Once we split the interval we sample 3 in each of the two intervals because the function is even these intervals are identical. So by checking to see if the function is even and forcing another split into 4 sub-intervals we sample the function enough to know that we are significantly off.
In short, the way we defined error worked against us because (by design I presume) the standardized sample points were not good representations of the function.

We could have also fixed this by splitting into two uneven intervals 1/3 2/3 rds or a random split - anything to break the symmetry of the integral would have worked for this function. I check to see if the two sub-intervals are reflections of one another and if they are I force another split.
After implementing my change I calculated

$$\int_{-1}^{1} (500x^6 - 700x^4 + 245x^2 - 3)\sin^2(2\pi x)dx = 7.366726$$

with an error of 6.74 10^{-10} in 184 intervals

Problem 3

a)

If we consider a function such that:

$$y(t_{k+1}) = y(t_k) + \int_{t_k}^{t_{k+1}} f(t,(y(t))dt$$

The trapezoid rule yields the approximation:

$$y_{k+1} = y_k + hf(t_{k+1/2},(y_k + y_{k+1/2})/2)$$

We can expand

$$f(t,y(t)) = f(t_k,y(t_k)) + (t-t_k)f'(t_k,y(t_k)) + \frac{1}{2}(t-t_k)^2 f''(t_k,y(t_k)) + O((t-t_k)^3)$$

we then drop the $O((t-t_k)^3)$ terms as the error of this method can be described by the remaining terms.
making the substitution

$$s = \frac{t-t_k}{t_{k+1}-t_k} = \frac{t-t_k}{h}$$

We obtain:

$$\begin{aligned}
\int_{t_k}^{t_{k+1}} f(t,(y(t))dt &= h\int_0^1 f_k + (s)f'_k + \frac{1}{2}(s)^2 f''_k \\
&= hsf_k + \frac{1}{2}s^2 f'_k + \frac{1}{6}s^3 f''_k \Big|_0^1 \\
&= hf_k + \frac{1}{2}h^2 f'_k + \frac{1}{6}h^3 f''_k
\end{aligned}$$

We then need to approximate f'

$$f'_k = \frac{f_{k+1}-f_k}{h} - \frac{h}{2}f''_k$$

Substituting that in we now have We obtain:

$$\begin{aligned}
\int_{t_k}^{t_{k+1}} f(t,(y(t))dt &= hf_k + \frac{1}{2}h^2 f'_k + \frac{1}{6}h^3 f''_k \\
&= hf_k + \frac{1}{2}(\frac{f_{k+1}-f_k}{h} - \frac{h}{2}f''_k) + \frac{1}{6}f''_k \\
&= h\frac{f_k + f_{k+1}}{2} - \frac{1}{12}h^2 f''_k
\end{aligned}$$

The final row we see is simply the Trapezoidal rule with an error term O(h^2)
Thus we have shown that the order of accuracy of the Trapezoidal rule is 2.

b)

For this section we will determine the stability of $y' = \lambda y$

$$y_{k+1} = yk + h\lambda y_k$$

Following lecture 12 slide 23 we find our amplification factor

$$(1 + h\lambda)$$

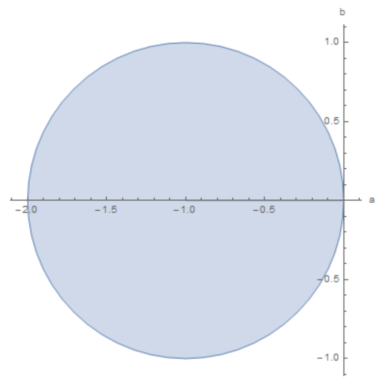
and we require $|1 + \tilde{h}| \leq 1$ for stability

so letting $\tilde{h} = a + bi$

$$|1 + a + bi| \leq 1$$

$$(1 + a)^2 + b^2 \leq 1$$

this describes the following area on a graph:



So this method is conditionally stable when $h\lambda = \tilde{h} = a + bi$ is in the shaded region shown above.

Problem 5

a)

Given:

$$J(x,y,u,v) = (x-\mu)^2 + y^2 + \frac{2(1-\mu)}{\sqrt{x^2+y^2}} + \frac{2\mu}{\sqrt{(x-1)^2+y^2}} - u^2 - v^2$$

$$\begin{aligned} x' &= -\frac{1}{2}\frac{\partial J}{\partial u}, & y' &= -\frac{1}{2}\frac{\partial J}{\partial v}, \\ u' &= v + \frac{1}{2}\frac{\partial J}{\partial x}, & v' &= -u + \frac{1}{2}\frac{\partial J}{\partial y}. \end{aligned}$$

We can establish the following system of ODEs to solve.

$$\begin{aligned} x' &= u \\ y' &= v, \\ u' &= v + \frac{x(1-\mu)}{(x^2+y^2)^{3/2}} + (x-\mu) - \frac{(x-1)\mu}{((x-1)^2+y^2)^{3/2}}, \\ v' &= -u + y - \frac{y(1-\mu)}{(x^2+y^2)^{3/2}} - \frac{y\mu}{((x-1)^2+y^2)^{3/2}} \end{aligned}$$

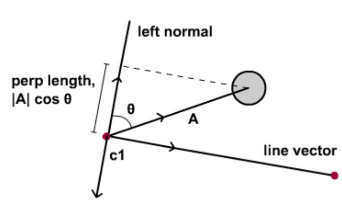
b)

I have implemented "intersect(x0,y0,x1,y1,px,py,r)" in the attached python code. Intersect() determines if a line from (x0,y0) to (x1,y1) intersects a circle centered at (px,py) with radius r. As it turns out the perpendicular distance between a circle and a line can be calculated as a dot product of two vectors :

1 - the vector from an endpoint to the center of the circle.

2 - the unit vector perpendicular to the line segment.

Once we have the perpendicular distance we can obtain the parallel distance. with both perpendicular distance and parrallel distance we can determine if the line intersected the circle.



c)

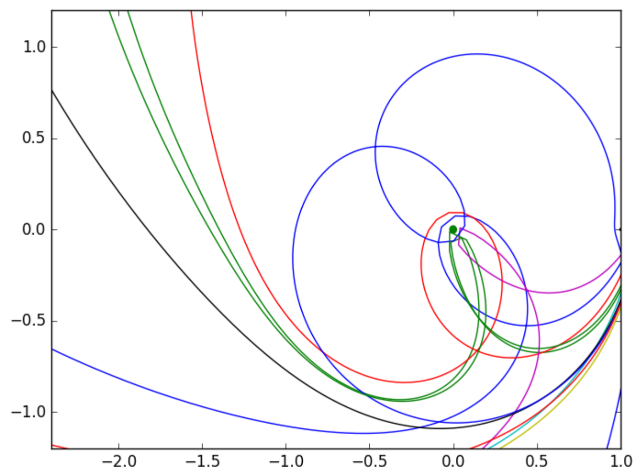
Given

$$X_{obs}(0) = (1.0798, 0) \qquad X_{obs}(.02) = (1.0802, -0.0189)$$

and there is error associated with the observation:
$\mathrm{X(t)} = \mathrm{X}_{obs}(t) + (E_x, E_y)$
then:

$$V(0) = \frac{X(.02) - X(0)}{0.02}$$

After implementing the ode solver and simulating 10 tragectories we obtain:



d)

To determine what happens for a given trajectory I run a script that calls my intersect() function for each leg of the predicted trajectory. It does this for the earth, if it does intersect I record the trajectory number and move on to another trajectory. If false it check for a collision with the moon.

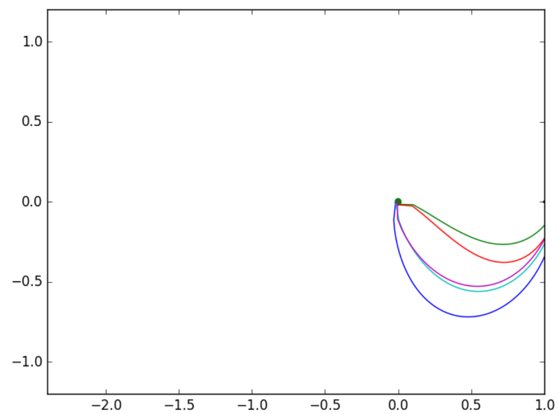


Figure 1: samples of trajectories that hit the earth

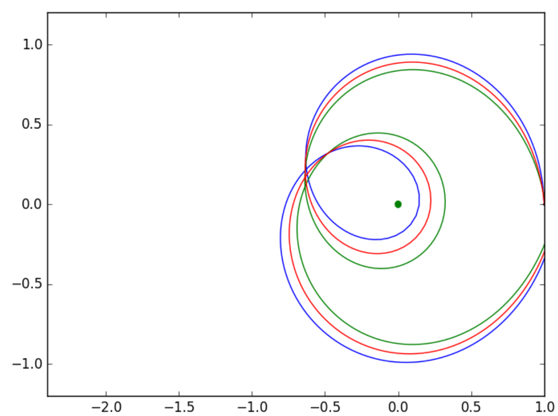


Figure 2: sample trajectories that hit the moon

Of the 2500 sample trajectories calculated 157 hit the earth first and 3 hit the moon first and 2340 did not collide. Based on my simulation there is 6.3% change the asteroid will hit the earth first and a 0.1% chance it will hit the moon