1.1

Ordinary Differential Equations

Contents

1 Introduction

This project is programmed in **MATLAB R2014a**. Consult section A for program documentation, listings and information on the structure of the programming for the project. This report is written in LaTeX 2_ε.

2 Equation I

Q1. If $|E_n|$ is eventually proportional to $e^{\gamma x}$, then as $n \to \infty$,

$$\gamma_n := \frac{1}{h} \ln \left| \frac{E_{n+1}}{E_n} \right| \to \frac{\gamma h(n+1-n)}{h} = \gamma$$

γ is henceforth defined as this limit, where it exists. The Euler method, applied to equation I on $[0, 6]$ with $h = 0.6$, yields the following results:

Table 1: Euler method with $h = 0.6$.

x_n	Y_n	y_n	E_n	γ_n
0.00	0.00000e+00	0.00000e+00	0.00000e+00	-
0.60	4.80000e+00	2.98715e-01	4.50128e+00	Inf
1.20	-2.25543e+01	9.07118e-02	-2.26450e+01	2.69263
1.80	1.13207e+02	2.73237e-02	1.13179e+02	2.68173
2.40	-5.65903e+02	8.22975e-03	-5.65911e+02	2.68244
3.00	2.82955e+03	2.47875e-03	2.82955e+03	2.68239
3.60	-1.41478e+04	7.46586e-04	-1.41478e+04	2.68240
4.20	7.07388e+04	2.24867e-04	7.07388e+04	2.68240
4.80	-3.53694e+05	6.77287e-05	-3.53694e+05	2.68240
5.40	1.76847e+06	2.03995e-05	1.76847e+06	2.68240
6.00	-8.84235e+06	6.14421e-06	-8.84235e+06	2.68240

This gives an estimate of the growth rate of the error magnitude:

$$\gamma(h) \approx 2.68240$$

Varying h yields the following output:

Table 2: Euler method with $h = 0.4$.

x_n	Y_n	y_n	E_n	γ_n
0.00	0.00000e+00	0.00000e+00	0.00000e+00	-
1.20	2.51325e+01	9.07118e-02	2.50418e+01	2.74161
2.40	-6.76298e+02	8.22975e-03	-6.76306e+02	2.74655
3.60	1.82602e+04	7.46586e-04	1.82602e+04	2.74653
4.80	-4.93027e+05	6.77287e-05	-4.93027e+05	2.74653
5.20	1.47908e+06	3.04325e-05	1.47908e+06	2.74653
5.60	-4.43724e+06	1.36742e-05	-4.43724e+06	2.74653
6.00	1.33117e+07	6.14421e-06	1.33117e+07	2.74653

Table 3: Euler method with $h = 0.25$.

x_n	Y_n	y_n	E_n	γ_n
0.00	0.00000e+00	0.00000e+00	0.00000e+00	-
1.00	-4.67799e+00	1.35290e-01	-4.81328e+00	1.64093
2.00	-2.43154e+01	1.83156e-02	-2.43337e+01	1.62239
3.00	-1.23183e+02	2.47875e-03	-1.23185e+02	1.62187
4.00	-6.23623e+02	3.35463e-04	-6.23623e+02	1.62186
5.00	-3.15709e+03	4.53999e-05	-3.15709e+03	1.62186
5.25	4.73564e+03	2.75364e-05	4.73564e+03	1.62186
5.50	-7.10346e+03	1.67017e-05	-7.10346e+03	1.62186
5.75	1.06552e+04	1.01301e-05	1.06552e+04	1.62186
6.00	-1.59828e+04	6.14421e-06	-1.59828e+04	1.62186

Table 4: Euler method with $h = 0.2$.

x_n	Y_n	y_n	E_n	γ_n
0.00	0.00000e+00	0.00000e+00	0.00000e+00	-
1.00	1.08754e+00	1.35290e-01	9.52248e-01	-0.07203
2.00	-9.40356e-01	1.83156e-02	-9.58671e-01	0.01003
3.00	9.60275e-01	2.47875e-03	9.57796e-01	-0.00136
4.00	-9.57579e-01	3.35463e-04	-9.57914e-01	0.00018
5.00	9.57944e-01	4.53999e-05	9.57898e-01	-0.00002
5.40	9.57920e-01	2.03995e-05	9.57899e-01	-0.00001
5.60	-9.57887e-01	1.36742e-05	-9.57901e-01	0.00001
5.80	9.57909e-01	9.16609e-06	9.57900e-01	-0.00001
6.00	-9.57894e-01	6.14421e-06	-9.57901e-01	0.00000

Table 5: Euler method with $h = 0.1$.

x_n	Y_n	y_n	E_n	γ_n
0.00	0.00000e+00	0.00000e+00	0.00000e+00	-
1.00	1.32239e-01	1.35290e-01	-3.05077e-03	-1.81594
2.00	1.78966e-02	1.83156e-02	-4.19019e-04	-1.99994
3.00	2.42204e-03	2.47875e-03	-5.67084e-05	-2.00000
4.00	3.27788e-04	3.35463e-04	-7.67464e-06	-2.00000
5.00	4.43613e-05	4.53999e-05	-1.03865e-06	-2.00000
5.70	1.09394e-05	1.11955e-05	-2.56128e-07	-2.00000
5.80	8.95639e-06	9.16609e-06	-2.09700e-07	-2.00000
5.90	7.33287e-06	7.50456e-06	-1.71688e-07	-2.00000
6.00	6.00365e-06	6.14421e-06	-1.40566e-07	-2.00000

Table 6: Euler method with $h = 0.01$.

x_n	Y_n	y_n	E_n	γ_n
0.00	0.00000e+00	0.00000e+00	0.00000e+00	-
1.00	1.34974e-01	1.35290e-01	-3.16357e-04	-1.55006
2.00	1.82703e-02	1.83156e-02	-4.53713e-05	-1.99976
3.00	2.47261e-03	2.47875e-03	-6.14052e-06	-2.00000
4.00	3.34632e-04	3.35463e-04	-8.31030e-07	-2.00000
5.00	4.52875e-05	4.53999e-05	-1.12468e-07	-2.00000
5.97	6.50799e-06	6.52415e-06	-1.61620e-08	-2.00000
5.98	6.37912e-06	6.39496e-06	-1.58420e-08	-2.00000
5.99	6.25281e-06	6.26833e-06	-1.55283e-08	-2.00000
6.00	6.12899e-06	6.14421e-06	-1.52208e-08	-2.00000

Thus, $h = 0.2$ presents a critical value, above which the error grows exponentially and below which the error decays exponentially (and indeed, the error converges to a non-zero constant for $h = 0.2$). There appears to be another critical value, ξ, between 0.1 and 0.2, below which $\gamma = -2$ exactly.

In fact, accurate estimates of γ can be obtained for all h, by iterating an appropriate number of times (sufficiently for convergence – which is slower near the singularity in the curve –but not enough to introduce errors due to small numbers – which become more prevalent as the curve decays as $x \to \infty$). This approach yields figure 1.[1]

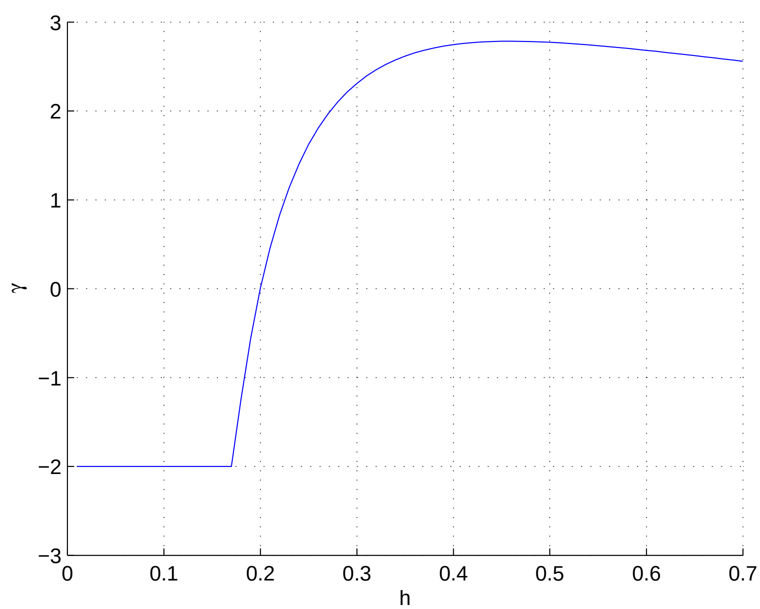


Figure 1: A numerical plot of $\gamma(h)$.

As will be proven in the following question, the critical point is precisely $h = 0.2$, and above it, $\gamma = \frac{1}{h}\ln(10h - 1)$. Apparently, ξ is the (unique) positive solution of the equation $e^{-2h} = 10h - 1$.

[1] See the listing for the function `hg` in section A.3 for the exact values of n used; these were determined heuristically.

Q2.

Explicit Euler Method Formula. It is possible to express the iterates given by the Euler method in a non-recursive, closed form. Fixing $n \in \mathbb{N}_0$, and noting that $x_n = hn$ under the scheme,

$$Y_{n+1} = Y_n + h\left(-10Y_n + 8e^{-2x_n}\right) = Y_n + h\left(-10Y_n + 8e^{-2hn}\right)$$

This is a first-order forced linear difference equation. The kernel of the homogeneous difference operator is $\langle (1-10h)^n \rangle$. It has a (particular) solution of the form $Y_n = A\left(e^{-2h}\right)^n$, $A \in \mathbb{R}$.

$Y_{n+1} - (1-10h)\,Y_n = Ae^{-2hn}\left(e^{-2h} - 1 + 10h\right) = 8h\left(e^{-2h}\right)^n$, so $A = \frac{8h}{e^{-2h}-1+10h}$.

$Y_0 = B(1-10h)^0 + \frac{8h}{e^{-2h}-1+10h}$, $B \in \mathbb{R}$, whence $B = -\frac{8h}{\left(e^{-2h}-1+10h\right)}$. Therefore,

$$Y_n = \frac{8h}{e^{-2h}+10h-1}\left(e^{-2hn} - (1-10h)^n\right)$$

Instability. Fixing h, Y_n diverges iff $|1-10h| \geq 1$, iff $h \geq 0.2$. In this case, γ can easily be directly computed.

Let $h \geq 0.2$. $\forall n \in \mathbb{N} \quad (1-10h)^n \neq 0$; by dividing by this quantity and rearranging, $\frac{E_{n+1}}{E_n}$ can be expressed as follows:

$$\frac{E_{n+1}}{E_n} = \frac{H\left(e^{-2h(n+1)} - (1-10h)^{n+1}\right) - y(hn)}{H\left(e^{-2hn} - (1-10h)^n\right) - y\left(h(n+1)\right)} = \frac{\left(e^{-2h(n+1)} - \frac{1}{H}y(hn)\right)(1-10h)^{-n} - (1-10h)}{\left(e^{-2hn} - \frac{1}{H}y\left(h(n+1)\right)\right)(1-10h)^{-n} - 1}$$

where $H = \frac{8h}{e^{-2h}+10h-1} \neq 0$. The left-hand term of both the denominator and numerator is a product of a bounded term $(1-10h)^{-n}$ and a null-convergent term, thereby converging to 0. Thus, $\lim_{n\to\infty} \frac{E_{n+1}}{E_n} = 1-10h$, and so, by continuity,

$$\gamma = \lim_{n\to\infty} \frac{1}{h} \ln\left|\frac{E_{n+1}}{E_n}\right| = \frac{1}{h}\ln(10h-1)$$

Convergence at $x = nh$. Fix $x = nh$. In the limit $h \to 0$, $n \to \infty$, by de l'Hôpital's rule, $\lim_{h\to 0} \frac{8h}{e^{-2h}+10h-1} = \lim_{h\to 0} \frac{8}{-2e^{-2h}+10} = 1$, and $e^{-2hn} = e^{-2x} \to e^{-2x}$, and $(1-10h)^n = \left(1 - \frac{10x}{n}\right)^n \to e^{-10x}$, so

$$Y_n \to e^{-2x} + e^{-10x}$$

The global error is in fact asymptotically linear. $\frac{E_n}{h}$ can be expressed as:

$$\frac{Y_n - y_n}{h} = \frac{1}{h}\left(\frac{8h}{e^{-2h}+10h-1} - 1\right)\left(e^{-2x} - e^{-10x}\right) + \frac{8}{e^{-2h}+10h-1}\left(e^{-10x} - (1-10h)^{\frac{x}{h}}\right)$$

Applying de l'Hôpital's rule twice on the left term (without its constant factor),

$$\lim_{h\to 0} \frac{1}{h}\left(\frac{8h}{e^{-2h}+10h-1} - 1\right) = -\lim_{h\to 0} \frac{4e^{-2h}}{4he^{-2h} + -4e^{-2h} + 20} = -\frac{1}{4}$$

The limit of the right term is the product of $\lim_{h\to 0} \frac{8h}{e^{-2h}+10h-1} = 1$ and $\lim_{h\to 0} \frac{1}{h}\left(e^{-10x} - (1-10h)^{\frac{x}{h}}\right)$. The latter can be computed using Taylor series:

$$\frac{1}{h}\left(e^{-10x} - (1-10h)^{\frac{x}{h}}\right) = \frac{e^{-10x}}{h}\left(1 - e^{\frac{x}{h}\ln(1-10h)+10x}\right)$$

$\frac{x}{h} \ln(1 - 10h) + 10x = -50xh + O(h^2)$, so $\forall n \geq 2 \quad \left(\frac{x}{h} \ln(1-10h) + 10x\right)^n = O(h^2)$, so

$$\frac{1}{h}\left(e^{-10x} - (1-10h)^{\frac{x}{h}}\right) = \frac{e^{-10x}}{h}\left(1 - \left(1 - \left(50xh + O\left(h^2\right)\right) + O\left(h^2\right)\right)\right) = 50xe^{-10x} + O(h)$$

Therefore,

$$\frac{E_n}{h} \to \chi(x) := -\frac{1}{4}\left(e^{-2x} - e^{-10x}\right) + 50xe^{-10x} = -\frac{1}{4}e^{-2x} + \left(50x + \frac{1}{4}\right)e^{-10x}$$

Q3. The Euler method is less accurate than the RK4 method for step size $h = 0.05$.

Table 7: Comparison of the two methods for $h = 0.05$.

x_n	Y_n (Euler)	Y_n (RK4)	E_n (Euler)	E_n (RK4)	$E_n/h\chi(x_n)$
0.00	0.00000e+00	0.00000e+00	0.00000e+00	0.00000e+00	-
0.05	4.00000e-01	2.98079e-01	1.01693e-01	-2.28134e-04	1.41069
0.10	5.61935e-01	4.50578e-01	1.11084e-01	-2.73203e-04	1.28667
0.15	6.08460e-01	5.17444e-01	9.07717e-02	-2.44269e-04	1.17576
0.20	6.00557e-01	5.34792e-01	6.55724e-02	-1.92886e-04	1.07530
0.25	5.68407e-01	5.24304e-01	4.39609e-02	-1.41471e-04	0.98242
0.50	3.62519e-01	3.61129e-01	1.37725e-03	-1.21418e-05	0.35240
0.75	2.20434e-01	2.22583e-01	-2.14326e-03	5.69625e-06	1.22810
1.00	1.33717e-01	1.35295e-01	-1.57267e-03	5.10391e-06	0.99686
1.25	8.11041e-02	8.20846e-02	-9.77140e-04	3.27848e-06	0.96330
1.50	4.91922e-02	4.97888e-02	-5.94603e-04	2.00740e-06	0.95720
1.75	2.98366e-02	3.01986e-02	-3.60805e-04	1.21943e-06	0.95614
2.00	1.80968e-02	1.83164e-02	-2.18852e-04	7.39799e-07	0.95596
2.25	1.09763e-02	1.11094e-02	-1.32742e-04	4.48728e-07	0.95593
2.50	6.65744e-03	6.73822e-03	-8.05120e-05	2.72169e-07	0.95592
2.75	4.03794e-03	4.08694e-03	-4.88330e-05	1.65079e-07	0.95592
2.80	3.65368e-03	3.69801e-03	-4.41859e-05	1.49369e-07	0.95592
2.85	3.30598e-03	3.34610e-03	-3.99811e-05	1.35155e-07	0.95592
2.90	2.99138e-03	3.02768e-03	-3.61764e-05	1.22293e-07	0.95592
2.95	2.70671e-03	2.73956e-03	-3.27337e-05	1.10656e-07	0.95592
3.00	2.44913e-03	2.47885e-03	-2.96187e-05	1.00125e-07	0.95592

The last column of 7 gives a set of values that pointwise converge to 1 as h is reduced ($\chi(x)$ is the asymptote predicted by Q2). Plotting various values of h, it is demonstrated in figure 3 that convergence is in fact uniform.

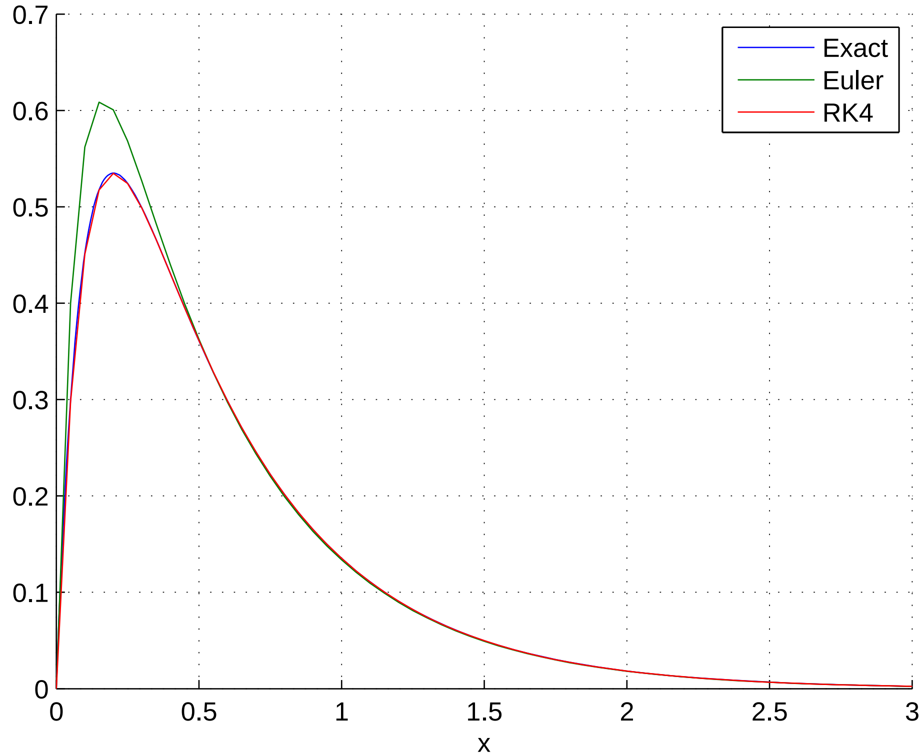


Figure 2: Comparison of the two methods for $h = 0.05$. Note the exact and RK4 curves are barely distinguishable.

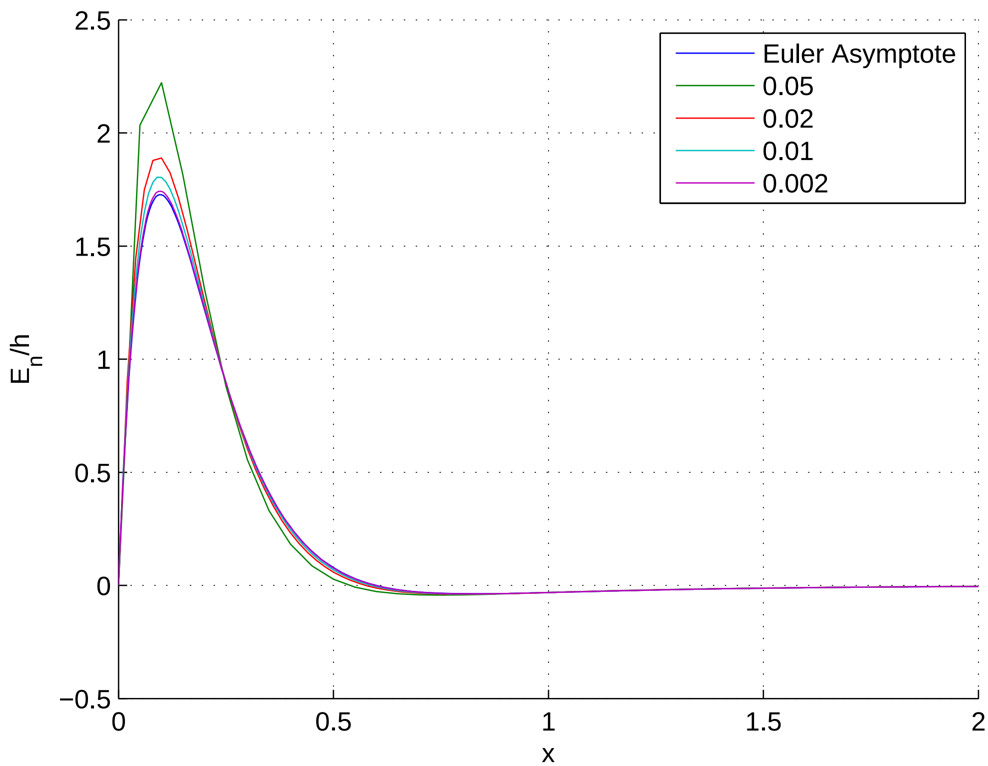


Figure 3: $\frac{E_n}{h}$ and its limit.

Q4. This plot yields two approximately-linear regressions between $\ln|E_n|$ and $\ln(h)$:

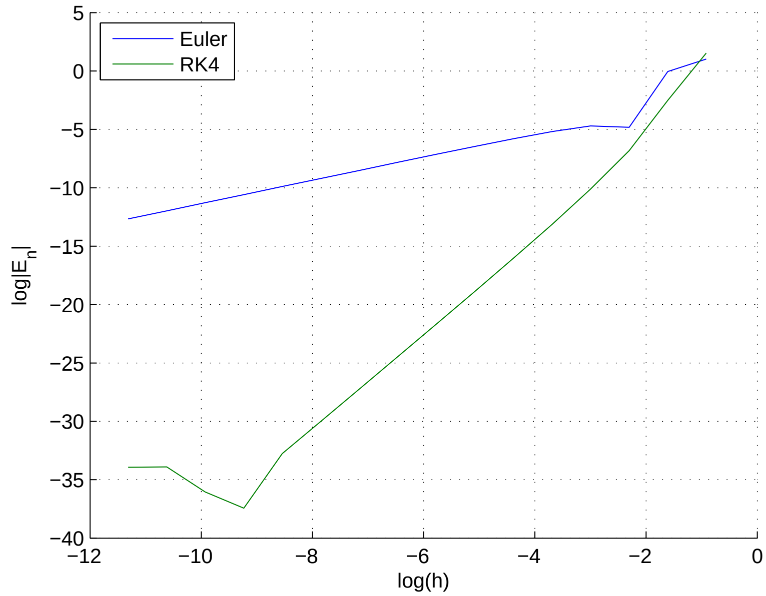


Figure 4: The order of the global errors of the two methods.

The gradient of a putative linear correlation can be estimated from the output (discarding the anomalies for extreme h, which are due to there being too few increments for stability to emerge for large h, and rounding error[2] as $E_n \to 0$ for small h), by looking for a modal value of the gradients m between successive points (represented in the the latter two columns of table 8). Indeed, denoting successive errors E_1,E_2 (the higher index corresponding to the higher value of h,

$$m = \frac{\log|E_2| - \log|E_1|}{\Delta \log(h)} = \frac{\log\left|\frac{E_2}{E_1}\right|}{\log(2)}$$

Hence, both relationships are approximately represented by $\log(E_n) = a\log(h) + b$ (with aforementioned gradient a), with $a = 1$ for the Euler method and $a = 4$ for RK4. This implies that $|E_n| = e^b h^a$, whereupon $E_n = O(h^a)$.

This agrees with theoretical predictions, as it is expected that $O(E_n) = O\left(\frac{e_n}{h}\right)$. Intuitively, fixing $x = hn$, $n \propto \frac{1}{h}$, and as each preceding step satisfies $e_k \propto h^{a+1}$, E_n is approximately proportional to nh^{a+1}, that is, h^a. The good behaviour of the differential equation means that $\sum_{k=1}^{n} e_k$ approximates E_n.

[2] Note that $\log(\epsilon) \approx -36$, where $\epsilon \approx 2 \times 10^{-16}$ is MATLAB's double (floating point) precision.

Table 8: E_n vs. h for both methods, with logarithmic gradient m.

h	E_n (Euler)	E_n (RK4)	m (Euler)	m (RK4)
0.400000	2.76899e+00	-4.62802e+00	-Inf	-Inf
0.200000	-9.58501e-01	-8.09975e-02	1.53051	5.83637
0.100000	8.03598e-03	-1.09271e-03	6.89816	6.21189
0.050000	9.08702e-03	-4.06936e-05	-0.17733	4.74697
0.025000	5.60907e-03	-1.99111e-06	0.69604	4.35316
0.012500	3.03065e-03	-1.10176e-07	0.88814	4.17569
0.006250	1.56714e-03	-6.47942e-09	0.95149	4.08780
0.003125	7.95966e-04	-3.92827e-10	0.97736	4.04390
0.001563	4.01013e-04	-2.41809e-11	0.98906	4.02195
0.000781	2.01256e-04	-1.49974e-12	0.99462	4.01108
0.000391	1.00814e-04	-9.30922e-14	0.99733	4.00991
0.000195	5.04535e-05	-5.88418e-15	0.99867	3.98375
0.000098	2.52384e-05	-5.55112e-17	0.99934	6.72792
0.000049	1.26221e-05	2.22045e-16	0.99967	-2.00000
0.000024	6.31176e-06	1.88738e-15	0.99983	-3.08746
0.000012	3.15606e-06	-1.83187e-15	0.99992	0.04307

3 Equation II

Q5.

Solution. The equation is a second-order forced linear differential equation with constant coefficients. The kernel of the homogeneous operator is $\langle e^{-\frac{\gamma}{2}t}\sin(\upsilon t), e^{-\frac{\gamma}{2}t}\cos(\upsilon t)\rangle$, where

$$\upsilon = \sqrt{\Omega^2 - \left(\frac{\gamma}{2}\right)^2}$$

A (particular) solution has the form $y_p = C\sin(\omega t) + D\cos(\omega t)$. Assuming such a solution and comparing components in the basis $\{\sin(\omega t), \cos(\omega t)\}$, the differential equation gives the following relations:

$$\sigma C - \tau D = a \quad \wedge \quad \tau C + \sigma D = 0$$

where $\sigma = \Omega^2 - \omega^2$, $\tau = \gamma\omega$ so that $\tau > 0$. This operator has determinant $\sigma^2 + \tau^2 > 0$, so there is a (unique) solution for C, D, and so a (unique) solution of the required form:

$$y_p(t) = \frac{a}{\sigma^2 + \tau^2}\left(\sigma\sin(\omega t) - \tau\cos(\omega t)\right)$$

Therefore, in the original parameters, the general solution is

$$y(t) = e^{-\frac{\gamma}{2}t}\left(A\sin\left(\sqrt{\Omega^2 - \left(\frac{\gamma}{2}\right)^2}\,t\right) + B\cos\left(\sqrt{\Omega^2 - \left(\frac{\gamma}{2}\right)^2}\,t\right)\right) + \frac{a\left((\Omega^2-\omega^2)\sin(\omega t) - \gamma\omega\cos(\omega t)\right)}{\left(\Omega^2-\omega^2\right)^2 + (\gamma\omega)^2}$$

for $A, B \in \mathbb{R}$.

Convergence. $\forall A, B \in \mathbb{R}$,

$$y(t) - y_p(t) = e^{-\frac{\gamma}{2}t}\left(A\sin\left(\sqrt{\Omega^2 - \left(\frac{\gamma}{2}\right)^2}\,t\right) + B\cos\left(\sqrt{\Omega^2 - \left(\frac{\gamma}{2}\right)^2}\,t\right)\right) \le e^{-\frac{\gamma}{2}t}\left(|A| + |B|\right) \to 0$$

so y_p is a sinusoidal function satisfying $\forall A, B \in \mathbb{R} \quad y(t) - y_p(t) \to 0$. Expressing it in the form $M\sin(\omega t - \phi)$, using a double-angle formula and comparing components in the basis $\{\sin(\omega t), \cos(\omega t)\}$,

$$M\cos(\phi) = \frac{a\sigma}{\sigma^2 + \tau^2} \quad \wedge \quad M\sin(\phi) = \frac{a\tau}{\sigma^2+\tau^2}$$

whence

$$M = \frac{a}{\sqrt{\sigma^2+\tau^2}} \quad \wedge \quad \phi_s \equiv_\pi \tan^{-1}\left(\frac{\beta}{\alpha}\right)$$

the latter assuming $\sigma \neq 0$ and giving two potential solutions for ϕ. Noting that $y_p(0) = \frac{-a\tau}{\sigma^2+\tau^2} < 0$ (since $\tau > 0$), and $y_p(0) = M\sin\left(-\tan^{-1}\left(\frac{\tau}{\sigma}\right)\right), M\sin\left(-\tan^{-1}\left(\frac{\tau}{\sigma}\right) + \pi\right)$, the following cases arise:

$$\sigma \gtrless 0 \implies \tan^{-1}\left(\frac{\tau}{\sigma}\right) \gtrless 0 \implies \sin\left(-\tan^{-1}\left(\frac{\tau}{\sigma}\right)\right) \lesseqgtr 0 \,\wedge\, \sin\left(-\tan^{-1}\left(\frac{\tau}{\sigma}\right) + \pi\right) \gtrless 0$$

The case $\sigma = 0$ can be deduced directly from the original equation by a phase shift. Therefore, in the original notation,

$$y_p(t) = M\sin(\omega t - \phi)\,; \quad M = \frac{a}{\sqrt{\left(\Omega^2-\omega^2\right)^2 + (\gamma\omega)^2}}, \quad \phi = \begin{cases} \tan^{-1}\left(\frac{\gamma\omega}{\Omega^2-\omega^2}\right) & \text{if } \Omega^2 > \omega^2 \\ \pi + \tan^{-1}\left(\frac{\gamma\omega}{\Omega^2-\omega^2}\right) & \text{if } \Omega^2 < \omega^2 \\ \frac{\pi}{2} & \text{if } \Omega^2 = \omega^2 \end{cases}$$

Q6. Under the specified conditions, the equation takes the form:

$$\frac{e^{-\frac{\gamma}{2}t}\left(\left(\frac{\omega\left(\frac{\gamma^2}{2}-4+\omega^2\right)}{\sqrt{4-\left(\frac{\gamma}{2}\right)^2}}\right)\sin\left(\sqrt{4-\left(\frac{\gamma}{2}\right)^2}t\right)+\gamma\omega\cos\left(\sqrt{4-\left(\frac{\gamma}{2}\right)^2}t\right)\right)+(4-\omega^2)\sin(\omega t)-\gamma\omega\cos(\omega t)}{(4-\omega^2)^2+(\gamma\omega)^2}$$

x_n	Y_n	y_n	E_n
0.8	6.7394e-02	1.2827e-01	-6.0873e-02
1.2	2.3825e-01	2.2913e-01	9.1206e-03
1.6	3.7312e-01	1.7419e-01	1.9893e-01
2.0	2.6699e-01	-5.2280e-02	3.1927e-01
2.4	-1.2632e-01	-2.7053e-01	1.4421e-01
8.0	1.2303e+00	-2.7341e-02	1.2577e+00
8.4	1.7944e+00	1.9973e-01	1.5947e+00
8.8	1.5561e+00	2.2178e-01	1.3343e+00
9.2	3.5992e-01	1.5529e-02	3.4439e-01
9.6	-1.3834e+00	-2.0821e-01	-1.1752e+00
10.0	-2.7702e+00	-2.2017e-01	-2.5501e+00

Figure 5: $h = 0.4$

x_n	Y_n	y_n	E_n
0.8	1.0436e-01	1.2827e-01	-2.3904e-02
1.2	2.5553e-01	2.2913e-01	2.6398e-02
1.6	2.8291e-01	1.7419e-01	1.0872e-01
2.0	5.9306e-02	-5.2280e-02	1.1159e-01
2.4	-3.0009e-01	-2.7053e-01	-2.9558e-02
8.0	-1.6862e-01	-2.7341e-02	-1.4128e-01
8.4	-4.1281e-02	1.9973e-01	-2.4101e-01
8.8	1.0182e-01	2.2178e-01	-1.1996e-01
9.2	7.0239e-02	1.5529e-02	5.4710e-02
9.6	-1.0636e-01	-2.0821e-01	1.0185e-01
10.0	-2.1004e-01	-2.2017e-01	1.0123e-02

Figure 6: $h = 0.2$

x_n	Y_n	y_n	E_n
0.8	1.1884e-01	1.2827e-01	-9.4265e-03
1.2	2.4637e-01	2.2913e-01	1.7236e-02
1.6	2.2534e-01	1.7419e-01	5.1150e-02
2.0	-1.1781e-02	-5.2280e-02	4.0499e-02
2.4	-3.0086e-01	-2.7053e-01	-3.0328e-02
8.0	-1.4257e-01	-2.7341e-02	-1.1523e-01
8.4	1.2685e-01	1.9973e-01	-7.2875e-02
8.8	2.5742e-01	2.2178e-01	3.5637e-02
9.2	1.1172e-01	1.5529e-02	9.6191e-02
9.6	-1.5776e-01	-2.0821e-01	5.0449e-02
10.0	-2.6794e-01	-2.2017e-01	-4.7778e-02

Figure 7: $h = 0.1$

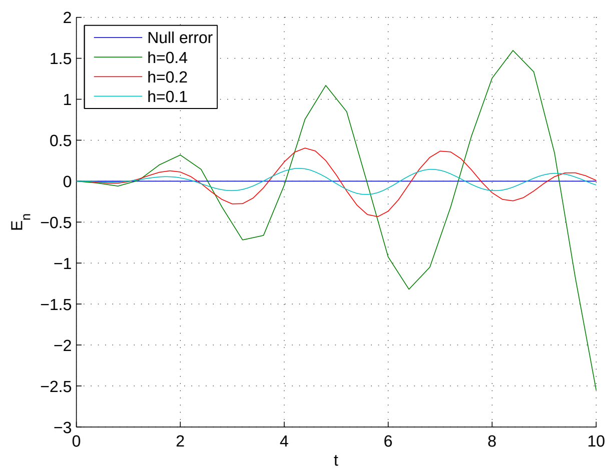
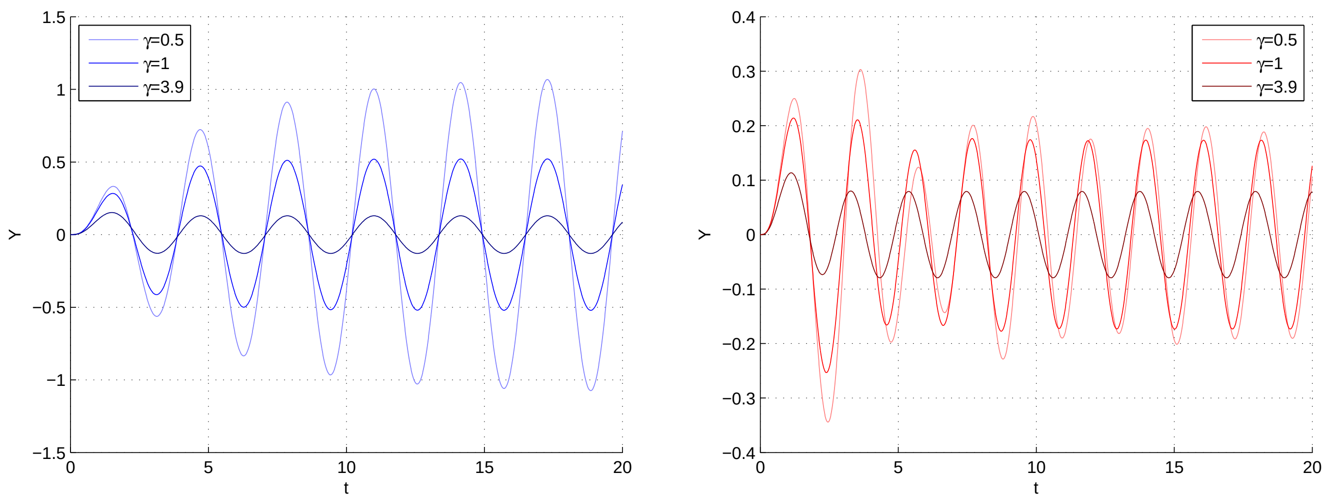


Figure 8: Comparison of errors.

In this case, the global error in the method oscillates and its rate of growth decays as $h \to 0$. The error itself appears to diverge (as $n \to \infty$) for $h = 0.4$.

Q7. The behaviour of the equation with $\omega = 2$ and $\omega = 3$ is displayed below.



(a) $\omega = 2$ (b) $\omega = 3$

Figure 9: Linearly-damped harmonic motion at different forcing frequencies.

In each case, the most salient feature of the graphs is that **the amplitude of the wave is negatively correlated with** γ; this is to be expected as, interpreting the equation as an equation of motion (with y'' representing acceleration, y' velocity and y position), higher values of γ imply a greater acceleration in the direction opposite to motion, yielding an equilibrium with lesser amplitude.

The exact relationship between the steady-state amplitude and γ follows from the formula of Q6:

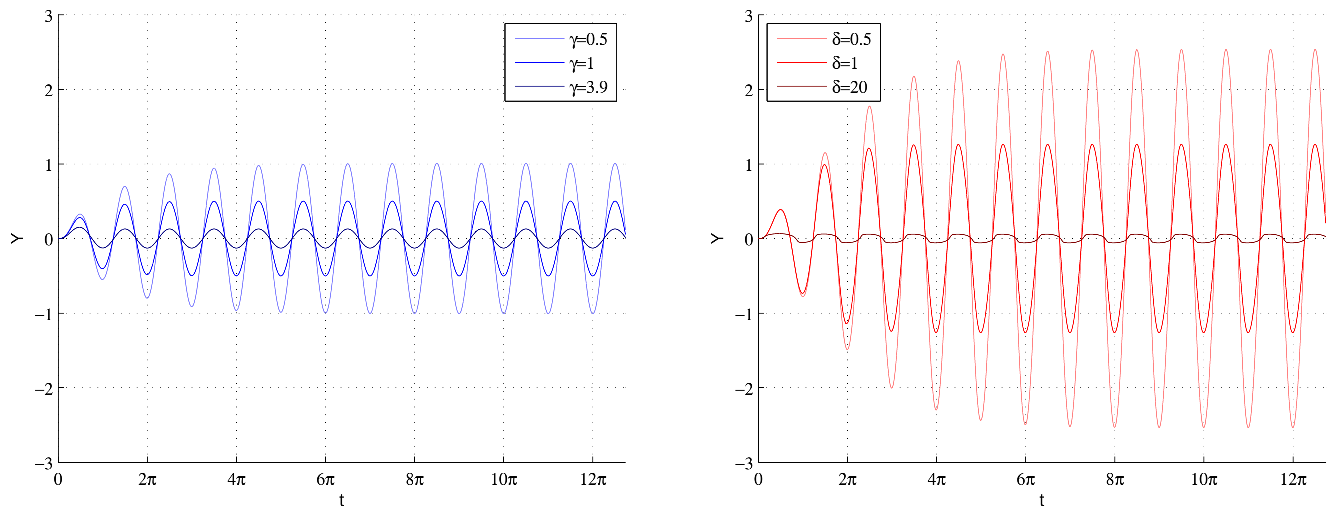
$$M = \frac{1}{\sqrt{(\omega^2 - 4) + \gamma^2\omega^2}}$$

This demonstrates the key difference; the condition for **resonance** of a simple harmonic oscillator (degeneracy in the equation) is for the forcing term to be within the span of the kernel of the differential operator. In this context, the only way to achieve this linear dependency relation is for $\gamma = 0$ (cancelling the exponential) and for the natural frequency of the oscillator to agree with the forcing frequency – i.e., for $\omega = \sqrt{\Omega^2 - \left(\frac{\gamma}{2}^2\right)} = 2$. These two conditions give an unbounded wave, rather than a steady-state, and indeed in the limit $\gamma \to 0$, $M = \frac{1}{2\gamma} \to \infty$. By contrast, it is not possible for resonance to occur with $\omega = 3$, and $M = \frac{1}{\sqrt{25+9\gamma^2}} \to \frac{1}{5}$ as $\gamma \to 0$, which explains the similar amplitudes of the $\gamma = 0.5$ and $\gamma = 1$ waves in this case. Physically, the resonance conditions mean that the oscillating system is able to efficiently absorb energy from the driving force (being synchronised in frequency) and has to oscillate with very large amplitudes for damping to counteract it and bring about an equilibrium.

Both cases also demonstrate a **transient period**, for low t, before their waves converge to steady states. If there is some damping, there is no resonance, so the particular integral is sinusoidal, and the complementary function decays (being bounded by a decaying exponential), so convergence occurs. The rate of convergence depends on the exponential envelope, $e^{-\frac{\gamma}{2}t}$, and is thus negatively correlated with γ; this can be seen by comparing the amplitudes of the waves with time.

Aside from amplitude, the other key difference exhibited is the **phase shift due to damping**. In the case $\Omega = \omega = 2$, the formula for the steady-state phase gives the phase as $\frac{\pi}{2}$; thus, the waves are eventually in phase with each other, regardless of the size of damping. The case $\omega = 3$ yields a shift of $\phi = \pi - tan^{-1}\left(\frac{3\gamma}{5}\right)$, which decreases as γ falls, approaching $\frac{\pi}{2}$. Despite this, each wave maintains an eventual **frequency independent of its damping**, equal to ω; thus, the $\omega = 3$ waves oscillate 1.5 times faster than the $\omega = 2$ waves.

Q8. Fixing $\omega = 2$, linear and non-linear damping are compared below, with different values of γ and δ, each non-zero in isolation. The interval $h = 0.001$ was chosen by plotting each equation with different, exponentially-decreasing values of h simultaneously until values were found that rendered each curve almost indistinguishable.



(a) Linear damping (γ)

(b) Non-linear damping (δ)

Figure 10: Comparison of damping.

The amplitude of the non-linearly–damped (*non-LD*) waves is also negatively correlated with the non-LD coefficient δ, for the same physical reason as the linear case. These waves have approximately the phase of the LD waves. However, the most striking difference the non-LD waves exhibit from the LD waves is that their steady state is not sinusoidal in shape. This is demonstrated clearly in the case $\delta = 20$, which is contrasted in figure 11 with an equivalent LD ($\gamma = 8.7$) wave (of similar amplitude and phase, coloured as above). The non-LD oscillator accelerates faster during increasing displacement, reaching maximum displacement sooner than the equivalent LD oscillator, and returning to equilibrium more slowly thereafter. Nonetheless, as $\delta \to 0$, the $A\delta^{-1}\cos(2t)$ term in the given formulation dominates, so the non-LD waves appear to converge to a sinusoidal wave (compare the positions of the peaks in figure 10(b)).

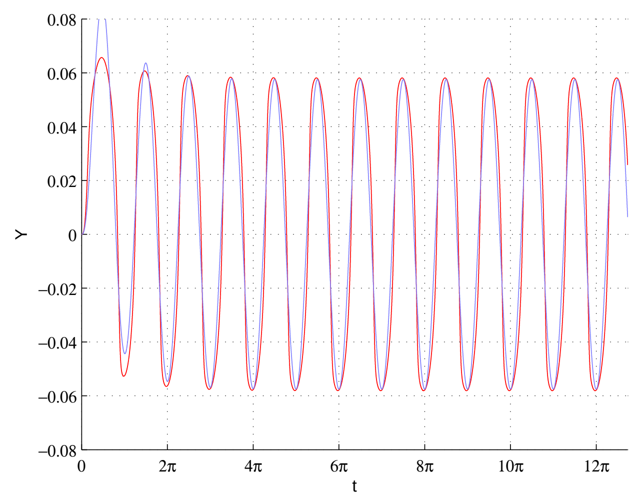


Figure 11: The case $\delta = 20$.

A Programs

The programs take the form of two modules, `E*.m`, which provide methods for each equation, and six scripts, `Q*.m`, which use these methods to generate output for each question. These depend on two auxiliary, unlisted scripts that process output and are based on scripts from the internet:

`tabler(L,d,f)`: Appends the table at global variable T in LaTeX-format to the file L.pdf with heading d and formatting f, specified as a cell array of statements in MATLAB fprintf format.

`grapher(P)`: Writes the figure at global variable G in pdf-format to the file P.pdf.

A.1 E1

```
classdef E1                                      % EQUATION I
methods(Static)
    function s = f(x,y)                          % f(x,y)
        s = -10*y + 8*exp(-2*x) ;
    end
    function s = e(x)                            % Exact
        s = exp(-2*x) - exp(-10*x) ;
    end
    function s = a(h,n)                          % Euler
        x = 0:h:h*n; y = zeros(1,n+1);
        for r = 1:n
            y(r+1) = y(r) + h*E1.f(x(r),y(r)) ;
        end
        s = y;
    end
    function s = b(h,n)                          % RK4
        x = 0:h:h*n; y = zeros(1,n+1);
        for r = 1:n
           a = h*E1.f(x(r), y(r)) ;
           b = h*E1.f(x(r)+0.5*h, y(r)+0.5*a) ;
           c = h*E1.f(x(r)+0.5*h, y(r)+0.5*b) ;
           d = h*E1.f(x(r)+h, y(r)+c) ;
           y(r+1) = y(r)+(1/6)*(a+2*b+2*c+d) ;
        end
        s = y;
    end
    function o = aE(h,n)                         % Euler Error
        o = E1.a(h,n) - arrayfun(@E1.e,0:h:h*n);
    end
    function o = bE(h,n)                         % RK4 Error
        o = E1.b(h,n) - arrayfun(@E1.e,0:h:h*n);
    end
    function o = asy(x)                          % Euler Asymptote
        o = -0.25*exp(-2*x) + (50*x+0.25)*exp(-10*x);
```

```
    end
end end
```

A.2 E2

```
classdef E2                                    % EQUATION II (ICs)
methods(Static)
    function s = y(t, g,O,a,o)                 % Exact (d = 0)
        if O <= g/2
            disp('No.')
            return
        end
        u = sqrt(O^2 - (g/2)^2);
        m = O^2 - o^2;
        n = g*o;
        r = a/(m^2 + n^2);
        c = exp(-g*t/2)*(((g*n/2-m*o)/u)*sin(u*t)+n*cos(u*t));
        p = m*sin(o*t) - n*cos(o*t);
        s = r*(c+p);
    end
    function s = Y(h,n, g,d,O,a,o)             % RK4
        f1 = @(t,y1,y2) y2;
        f2 = @(t,y1,y2) -g*y2 - (d^3)*(y1^2)*y2 - (O^2)*y1 + a*
           sin(o*t);
        t = 0:h:h*n;
        Y1 = zeros(1,n+1);
        Y2 = zeros(1,n+1);
        for r = 1:n
            i = h*f1(t(r), Y1(r), Y2(r)) ;
            j = h*f1(t(r)+0.5*h, Y1(r)+0.5*i, Y2(r)) ;
            k = h*f1(t(r)+0.5*h, Y1(r)+0.5*j, Y2(r)) ;
            l = h*f1(t(r)+h, Y1(r)+k, Y2(r)) ;
                Y1(r+1) = Y1(r)+(1/6)*(i+2*j+2*k+l) ;
            i = h*f2(t(r), Y1(r), Y2(r)) ;
            j = h*f2(t(r)+0.5*h, Y1(r), Y2(r)+0.5*i) ;
            k = h*f2(t(r)+0.5*h, Y1(r), Y2(r)+0.5*j) ;
            l = h*f2(t(r)+h, Y1(r), Y2(r)+k) ;
                Y2(r+1) = Y2(r)+(1/6)*(i+2*j+2*k+l) ;
        end
        s=Y1;
    end
end end
```

A.3 Q1

```
function Q1
    for i = [0.6,0.5,0.4,0.25,0.2,0.1,0.01]
        go(i)
    end
    hg()
end

function go(h)                                          % Table
    fprintf(num2str(h))
    n = 6/h;                    x = (0:h:h*n).';
    Y = (E1.a(h,n)).';          y = arrayfun(@E1.e,x);
    E = Y - y;                  Er = [0;E(1:end-1)];
    g = (1/h)*log(abs(E./Er));
    T = table(x,Y,y,E,g) %#ok<NOPRT>
    assignin('base', 'T', T);
    tabler('L1',num2str(h),{'%3.2f',1,'%7.5e',3,'%6.5f',1});
end

function hg                                             % Graph (h -- gamma)
    s = zeros(1,70);
    for r = 1:70
    % The conditional statement adapts different resolutions to
       different parts of the graph.
        if r < 20
            s(r) = g(r/100,900);
        else
            s(r) = g(r/100,350);
        end
    end
    assignin('base', 'G', figure); clf; hold all; grid on
        plot(0.01:0.01:0.7,s)
        ylabel('\gamma'); xlabel('h')
    grapher('P1')
end

function o = g(h,n)                                     % Gamma Estimator
    E = E1.aE(h,n); E1 = E(2:end); E2 = E(1:end-1);
    O = (1/h)*log(abs(E1./E2));
    o = O(n-1);
end
```

A.4 Q3

```
function Q3
    h = 0.05;
    T = solve(h,3/h) %#ok<NOPRT>
    t = (0:h:3);
    assignin('base', 'T', T);
    tabler('L3','',{'%3.2f',1,'%7.5e',4,'%6.5f',1})
    T = table2array(T);

    assignin('base', 'G', figure); clf; hold all; grid on
        fplot(@E1.e,[0,3])                          % Graph (a)
        plot(t,T(:,2))
        plot(t,T(:,3))
        legend('Exact','Euler','RK4'); xlabel('x')
    grapher('P3a')

    assignin('base', 'G', figure); clf; hold all; grid on
        fplot(@E1.asy,[0,2])                        % Graph (b)
        l = {'Euler␣Asymptote'};
        for i = [0.05,0.02,0.01,0.002]
            t = (0:i:2);
            T = table2array(solve(i,2/i));
            plot(t,T(:,4)/i)
            l = [l, num2str(i)];
        end
        legend(l); xlabel('x'); ylabel('E_n/h')
    grapher('P3b')
end

function T = solve(h,n)
    t = (0:h:h*n).';                                % Table
    e = arrayfun(@E1.e,t);
    a = E1.a(h,n).';
    b = E1.b(h,n).';
    Ea = a - e;
    Eb = b - e;
    asy = Ea ./ (h .* arrayfun(@E1.asy,t));
    T = table(t,a,b,Ea,Eb,asy);
end
```

A.5 Q4

```
k = (0:15).'; h = 0.4./(2.^k); l = log(h);
s = zeros(16,1);                                % Euler Data
for i = k.'
    E = E1.aE(0.4/(2^i),2^i);
    s(i+1) = E((2^i)+1);
end
sr = [0;s(1:end-1)];
ga = log(abs(sr./s))/log(2);

t = zeros(16,1);                                % RK4 Data
for i = k.'
    E = E1.bE(0.4/(2^i),2^i);
    t(i+1) = E((2^i)+1);
end
tr = [0;t(1:end-1)];
gb = log(abs(tr./t))/log(2);

T = table(h,s,t,ga,gb) %#ok<NOPTS>              % Table
tabler('L4','',{'%7.6f',1,'%7.5e',2,'%6.5f',2})

G = figure; clf; hold all; grid on; axis on % Graph
    plot(l,log(abs(s)))
    plot(l,log(abs(t)))
    legend('Euler','RK4','Location','northwest')
    ylabel('log|E_n|'); xlabel('log(h)')
grapher('P4')
```

A.6 Q6

```
function Q6
    assignin('base', 'G', figure); clf; hold all; grid on
        z = @(t) 0;
        fplot(z,[0,10])
        l = {'Null error'};
        for i = [0.4,0.2,0.1]
            go(i)
            l = [l, strcat('h=',num2str(i))];
        end
        legend(l,'Location','northwest')
        ylabel('E_n'); xlabel('t')
    grapher('P6')
end

function go(h)
    n = 10/h;
    t = (0:h:h*n).';
    Y = E2.Y(h,n, 1,0,2,1,sqrt(7)).';
    f = @E2.y;
    e = @(t) f(t, 1,2,1,sqrt(7));
    y = arrayfun(e,t);
    E = Y - y;
    T = table(t,Y,y,E) %#ok<NOPRT>
    assignin('base', 'T', T);
    tabler('L6',num2str(h),{'%2.1f',1,'%6.4e',3});
    plot(t,E)
end
```

A.7 Q7

```
h = 0.01; n = 20/h; t = (0:h:h*n).';
Y1 = E2.Y(h,n,0.5,0,2,1,2).'; Y4 = E2.Y(h,n,0.5,0,2,1,3).';
Y2 = E2.Y(h,n,1  ,0,2,1,2).'; Y5 = E2.Y(h,n,1  ,0,2,1,3).';
Y3 = E2.Y(h,n,3.9,0,2,1,2).'; Y6 = E2.Y(h,n,3.9,0,2,1,3).';

G = figure; clf; hold all; grid on
    plot(t,Y1,'Color',[0.5 0.5 1]);
    plot(t,Y2,'Color',[0 0 1]);
    plot(t,Y3,'Color',[0 0 0.5]);
    legend('\gamma=0.5','\gamma=1','\gamma=3.9','Location','
       northwest');
    ylabel('Y'); xlabel('t')
grapher('P7a')
G = figure; clf; hold all; grid on
    plot(t,Y4,'Color',[1 0.5 0.5]);
    plot(t,Y5,'Color',[1 0 0]);
    plot(t,Y6,'Color',[0.5 0 0]);
    legend('\gamma=0.5','\gamma=1','\gamma=3.9')
    ylabel('Y'); xlabel('t')
grapher('P7b')
```

A.8 Q8

```
function Q8
    h = 0.001; n = 40/h; t = (0:h:h*n).';
    Y1 = E2.Y(h,n,0.5,0,2,1,2).'; Y4 = E2.Y(h,n,0,0.5,2,1,2).';
    Y2 = E2.Y(h,n,1  ,0,2,1,2).'; Y5 = E2.Y(h,n,0,1  ,2,1,2).';
    Y3 = E2.Y(h,n,3.9,0,2,1,2).'; Y6 = E2.Y(h,n,0,20 ,2,1,2).';

    assignin('base', 'G', figure); clf; hold all; grid on
        plot(t,Y1,'Color',[0.5 0.5 1]); plot(t,Y2,'Color',[0 0
           1]);
        plot(t,Y3,'Color',[0 0 0.5]);
        legend('\gamma=0.5','\gamma=1','\gamma=3.9')
        ylim([-3,3]); ylabel('Y'); xlabel('t'); piaxis()
    grapher('P8a')
    assignin('base', 'G', figure); clf; hold all; grid on
        plot(t,Y4,'Color',[1 0.5 0.5]); plot(t,Y5,'Color',[1 0
           0]);
        plot(t,Y6,'Color',[0.5 0 0]);
        legend('\delta=0.5','\delta=1','\delta=20','Location','
           northwest')
        ylim([-3,3]); ylabel('Y'); xlabel('t'); piaxis()
    grapher('P8b')
    assignin('base', 'G', figure); clf; hold all; grid on
        t = (0:h:40);
        plot(t,Y6,'Color',[1 0 0])
        plot(t,E2.Y(h,n,8.7,0,2,1,2).','Color',[0.5 0.5 1])
        ylim([-0.08,0.08]); ylabel('Y'); xlabel('t'); piaxis()
        grapher('P8c')
end

function piaxis
    set(gca,'xtick',[0:2*pi():12*pi()],...
        'xticklabel',{'0' '2p' '4p' '6p' '8p' '10p' '12p'},...
        'fontname','symbol');
end
```