ENGR 6101: Computational Engineering

Solution Set 4 *(due on Friday, 10/9, in my mailbox)*

Questions:

1. Consider the following ODE:

$$\begin{aligned} x' &= x(1-x) \\ y(0) &= 0.1 \end{aligned}$$

   (a) (3 pts.) Find the analytic solution. Note that you can do this with separation of variables.
   SOLUTION.

$$\begin{aligned} \frac{dx}{dt} &= x(1-x) \\ \int \frac{dx}{x(1-x)} &= \int dt \\ \int \frac{dx}{x} + \int \frac{dx}{1-x} &= t + C \\ \ln(x) - \ln(1-x) &= t + C \\ \frac{x}{1-x} &= Ce^t \\ x(t) &= \frac{1}{1+Ce^{-t}} \end{aligned}$$

   Plugging the initial condition $x(0) = 0.1$, we get the exact analytic solution

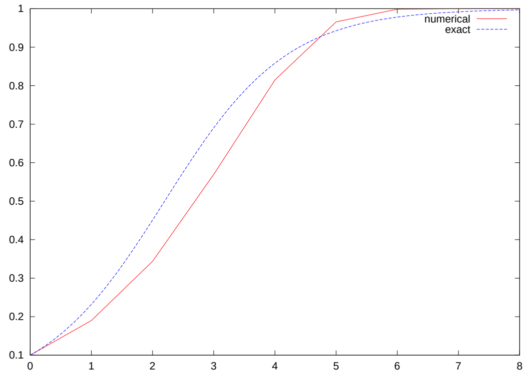
$$= y(t) = \frac{1}{1+9e^{-t}}$$

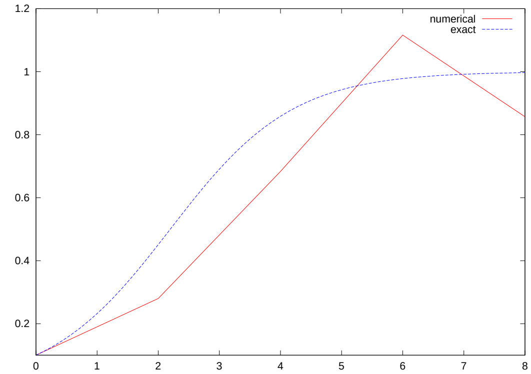
   (b) (3 pts.) Write a Matlab code that solves this equation on the interval $[0, 8]$ using Euler method. Your code should plot both the numerical and the analytic solution on the same figure. Run your code with $h = 1$ and $h = 2$. How does the accuracy change? Name your code `ode_euler.m` and submit soft copy to `caner@uga.edu`. Include a hard copy of your code and the two graphs (for $h = 1$ and $h = 2$) along with your HW solutions.

   SOLUTION.

```
function euler
t(1) = 0;
x(1) = .1;
h = 2;
it = 8/h;
for i=1:it
  x(i+1) = x(i) + h*x(i)*(1-x(i));
  t(i+1) = t(i) + h;
end
plot(t,x,'-r')
hold on;
```

```
fplot('1/(1+9*exp(-x))',[0,8]);
legend('numerical','exact')
hold off
```

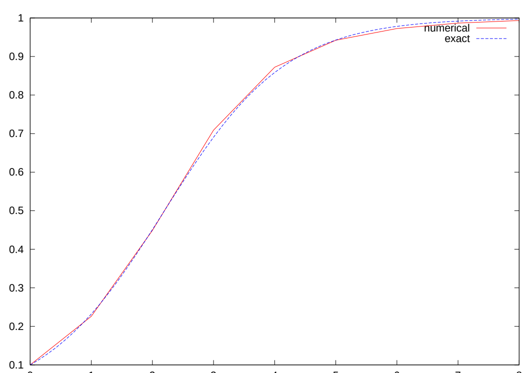


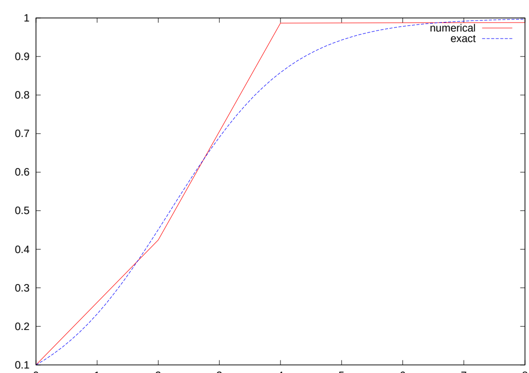


(c) (3 pts.) Repeat part (b) using second order Taylor-series method. In this case, name your code `ode_taylor2.m`. Again, include both hard and soft copies of your code, and a hard copy of the two graphs.

SOLUTION.

```
function taylor2
t(1) = 0;
x(1) = .1;
h = 2;
it = 8/h;
for i=1:it
  x(i+1) = x(i) + h*x(i)*(1-x(i)) + h^2/2*(1-2*x(i))*x(i)*(1-x(i));
  t(i+1) = t(i) + h;
end
plot(t,x,'-r')
hold on;
fplot('1/(1+9*exp(-x))',[0,8]);
legend('numerical','exact')
hold off
```

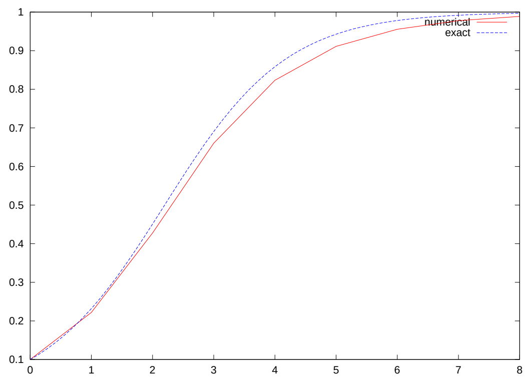


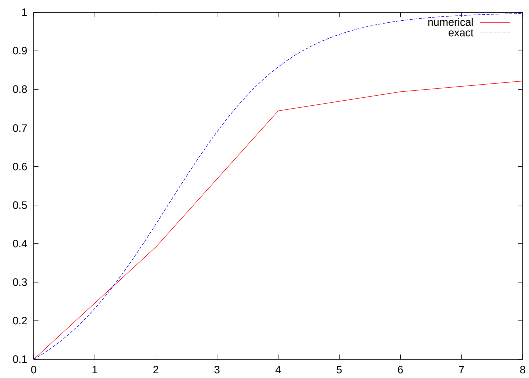


(d) (3 pts.) Repeat part (b) using second order Runge-Kutta method. In this case, name your code `ode_rk2.m`. Again, include both hard and soft copies of your code, and a hard copy of the two graphs.

SOLUTION.

```
function rk2
t(1) = 0;
x(1) = .1;
h = 2;
it = 8/h;
for i=1:it
  k1 = h*x(i)*(1-x(i));
  k2 = h*(x(i)+k1)*(1-(x(i)+k1));
  x(i+1) = x(i) + (k1+k2)/2;
  t(i+1) = t(i) + h;
end
plot(t,x,'-r')
hold on;
fplot('1/(1+9*exp(-x))',[0,8]);
legend('numerical','exact')
hold off
```



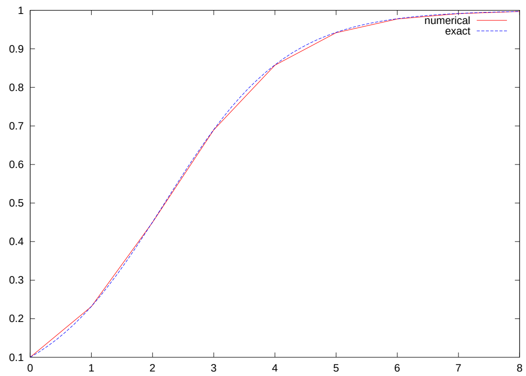


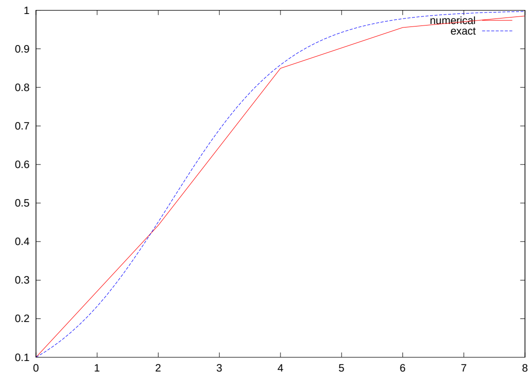
(e) (3 pts.) Repeat part (b) using fourth order Runge-Kutta method. In this case, name your code `ode_rk4.m`. Again, include both hard and soft copies of your code, and a hard copy of the two graphs.

SOLUTION.

```
function rk4
t(1) = 0;
x(1) = .1;
h = 2;
it = 8/h;
for i=1:it
  k1 = h*x(i)*(1-x(i));
  k2 = h*(x(i)+k1/2)*(1-(x(i)+k1/2));
  k3 = h*(x(i)+k2/2)*(1-(x(i)+k2/2));
  k4 = h*(x(i)+k3)*(1-(x(i)+k3));
  x(i+1) = x(i) + (k1+2*k2+2*k3+k4)/6;
  t(i+1) = t(i) + h;
end
```

```
plot(t,x,'-r')
hold on;
fplot('1/(1+9*exp(-x))',[0,8]);
legend('numerical','exact')
hold off
```





(f) (3 pts.) Sort the accuracy of all four numerical methods that you've tried. Second order Taylor and second order Runge-Kutta are both second order methods. What could be the mathematical reason for the additional accuracy of the better method?

SOLUTION. Accuracy increases as follows:

$$\text{Euler} \;\rightarrow\; \text{RK2} \;\rightarrow\; 2^{nd} \text{ order Taylor} \;\rightarrow\; \text{RK4}$$

Second order Taylor method is more accurate than second order Runge-Kutta method because the derivation of the latter method includes one more Taylor's expansion compared to the former.

2. (3 pts.) Prove that when the fourth-order Runge-Kutta method is applied to the problem $x' = \lambda x$, the formula for advancing this solution will be

$$x(t+h) = \left[1 + h\lambda + \frac{1}{2}h^2\lambda^2 + \frac{1}{6}h^3\lambda^3 + \frac{1}{24}h^4\lambda^4\right] x(t)$$

SOLUTION.

$$\begin{aligned}
K_1 &= h\lambda x \\
K_2 &= h\lambda(x + \frac{K_1}{2}) = h\lambda x(1 + \frac{h\lambda}{2}) \\
K_3 &= h\lambda(x + \frac{K_2}{2}) = h\lambda[x + \frac{h\lambda x}{2}(1 + \frac{h\lambda}{2})] = h\lambda x(1 + \frac{h\lambda}{2} + \frac{h^2\lambda^2}{4}) \\
K_4 &= h\lambda(x + \frac{K_3}{2}) = h\lambda[x + h\lambda x(1 + \frac{h\lambda}{2} + \frac{h^2\lambda^2}{4})] = h\lambda x(1 + h\lambda + \frac{h^2\lambda^2}{2} + \frac{h^3\lambda^3}{4})
\end{aligned}$$

$$\begin{aligned}
x(t+h) &= x(t) + \frac{K_1 + 2k_1 + 2K_3 + K_4}{6} \\
&= x + \frac{1}{6}\left[h\lambda x + 2h\lambda x(1 + \frac{h\lambda}{2}) + 2h\lambda x(1 + \frac{h\lambda}{2} + \frac{h^2\lambda^2}{4}) + h\lambda x(1 + h\lambda + \frac{h^2\lambda^2}{2} + \frac{h^3\lambda^3}{4})\right]
\end{aligned}$$

$$
\begin{aligned}
&= x + \frac{x}{6}\left[6h\lambda + 3h^2\lambda^2 + h^3\lambda^3 + \frac{1}{4}h^4\lambda^4\right] \\
&= x(t)\left[1 + h\lambda + \frac{1}{2}h^2\lambda^2 + \frac{1}{6}h^2\lambda^2 + \frac{1}{24}h^4\lambda^4\right]
\end{aligned}
$$