Math 5604 Homework 2

Jacob Hauck

February 13, 2024

Problem 1.

Consider the IVP

$$y' = f(t, y), \qquad y(0) = a. \tag{1}$$

Let $k > 0$ be the time step for a numerical scheme to approximate y'. Assume that f is L-Lipschitz in y for all t.

1. Consider the scheme

$$y^{n+1} = y^n + kf\big(t_{n+1}, y^{n+1}\big), \quad n = 0, 1, 2, \dots, \qquad y^0 = a. \tag{2}$$

Suppose that $y(t_n) = y^n$. Using the Taylor expansion of y about t_{n+1},

$$y(t_n) = y(t_{n+1}) - ky'(t_{n+1}) + \tau(k),$$

where the remainder $\tau(k) = \mathcal{O}(k^2)$ as $k \to 0$. Using the assumption that $y(t_n) = y^n$ and the definition of the scheme, we have

$$\begin{aligned} y(t_{n+1}) &= y(t_n) + ky'(t_{n+1}) + \tau(k) \\ &= y^n + kf\big(t_{n+1}, y^{n+1}\big) + k\left[f(t_{n+1}, y(t_{n+1})) - f\big(t_{n+1}, y^{n+1}\big)\right] + \tau(k) \\ &= y^{n+1} + k\left[f(t_{n+1}, y(t_{n+1})) - f\big(t_{n+1}, y^{n+1}\big)\right] + \tau(k). \end{aligned}$$

Thus,

$$\text{LTE} = \left|y(t_{n+1}) - y^{n+1}\right| = \left|k\big[f(t_{n+1}, y(t_{n+1})) - f\big(t_{n+1}, y^{n+1}\big)\big] + \tau(k)\right|.$$

We can easily show that LTE $\to 0$ as $k \to 0$, that is, that the scheme is consistent.

By the Lipschitz condition on f,

$$\begin{aligned} \text{LTE} = \left|y(t_{n+1}) - y^{n+1}\right| &\leq k\left|f(t_{n+1}, y(t_{n+1})) - f\big(t_{n+1}, y^{n+1}\big)\right| + |\tau(k)| \\ &\leq kL\big|y(t_{n+1}) - y^{n+1}\big| + |\tau(k)|. \end{aligned}$$

For all $k < \frac{1}{L}$, we have $1 - kL > 0$, so

$$\text{LTE} \leq \frac{|\tau(k)|}{1 - kL}, \qquad k < \frac{1}{L}.$$

This implies that

$$0 \leq \lim_{k \to 0} \text{LTE} \leq \lim_{k \to 0} \frac{|\tau(k)|}{1 - kL} = 0$$

because $\tau(k) \to 0$ as $k \to 0$, and $1 - kL \to 1$ as $k \to 0$. That is, LTE $\to 0$ as $k \to 0$, and the scheme is consistent.

2. Consider the scheme

$$y^{n+1} = y^{n-1} + 2kf(t_n, y_n), \quad n = 0, 1, 2, \ldots, \qquad y^0 = a. \tag{3}$$

Suppose that $y(t_{n-1}) = y^{n-1}$, and $y(t_n) = y^n$. Using the Taylor expansion of y about t_n to the left and to the right, we have

$$\begin{aligned} y(t_{n+1}) &= y(t_n) + ky'(t_n) + \tau_1(k) \\ y(t_{n-1}) &= y(t_n) - ky'(t_n) + \tau_2(k), \end{aligned}$$

where the remainders $\tau_1(k)$ and $\tau_2(k)$ satisfy $\tau_1(k) = \mathcal{O}(k^2)$ and $\tau_2(k) = \mathcal{O}(k^2)$ as $k \to 0$.
By the ODE and the assumptions that $y(t_{n-1}) = y^{n-1}$ and $y(t_n) = y^n$, this implies that

$$\begin{aligned} y(t_{n+1}) - y^{n-1} &= y(t_{n+1}) - y(t_{n-1}) \\ &= 2ky'(t_n) + \tau_1(k) - \tau_2(k) \\ &= 2kf(t_n, y(t_n)) + \tau_1(k) - \tau_2(k) \\ &= 2kf(t_n, y^n) + \tau_1(k) - \tau_2(k). \end{aligned}$$

Therefore, the LTE is given by

$$\text{LTE} = \left|y^{n+1} - y(t_{n+1})\right| = |\tau_1(k) - \tau_2(k)|.$$

Since both $\tau_1(k) \to 0$ and $\tau_2(k) \to 0$ as $k \to 0$, it follows that LTE $\to 0$ as $k \to 0$. That is, the scheme is consistent.

3. Let $\theta \in [0, 1]$, and consider the scheme

$$y^{n+1} = y^n + kf\left(t^n + (1-\theta)k, \theta y^n + (1-\theta)y^{n+1}\right), \quad n = 0, 1, 2, \ldots, \qquad y^0 = a. \tag{4}$$

Suppose that $y(t_n) = y^n$. Using the Taylor expansion of y about $t_n + (1-\theta)k$, we have

$$y(t_n) = y(t_n + (1-\theta)k) - (1-\theta)ky'(t_n + (1-\theta)k) + \tau_1(k), \tag{5}$$

where $\tau_1(k) = \mathcal{O}(k^2)$ as $k \to 0$ (because $\theta \in [0, 1]$). Similarly,

$$y(t_{n+1}) = y(t_n + (1-\theta)k) + \theta ky'(t_n + (1-\theta)k) + \tau_2(k), \tag{6}$$

where $\tau_2(k) = \mathcal{O}(k^2)$ as $k \to 0$. Therefore,

$$\begin{aligned} y(t_{n+1}) &= y(t_n) + (1-\theta)ky'(t_n + (1-\theta)k) - \tau_1(k) + \theta ky'(t_n + (1-\theta)k) + \tau_2(k) \\ &= y(t_n) + ky'(t_n + (1-\theta)k) - \tau_1(k) + \tau_2(k) \\ &= y^n + kf(t_n + (1-\theta)k, y(t_n + (1-\theta)k)) - \tau_1(k) + \tau_2(k). \end{aligned}$$

Then the local truncation error is given by

$$\begin{aligned} \text{LTE} &= \left|y(t_{n+1}) - y^{n+1}\right| \\ &= \left|k\left[f(t_n + (1-\theta)k, y(t_n + (1-\theta)k)) - f\big(t_n + (1-\theta)k, \theta y^n + (1-\theta)y^{n+1}\big)\right] - \tau_1(k) + \tau_2(k)\right|. \end{aligned}$$

By the Lipschitz property of f, we have

$$\text{LTE} \leq kL\left|y(t_n + (1-\theta)k) - \theta y^n - (1-\theta)y^{n+1}\right| + |\tau_2(k) - \tau_1(k)|.$$

Multiplying (5) by θ and (6) by $1-\theta$ and adding the results, we see that

$$y(t_n + (1-\theta)k) = \theta y(t_n) + (1-\theta)y(t_{n+1}) + \theta\tau_1(k) + (1-\theta)\tau_2(k).$$

Since $y(t_n) = y^n$ by hypothesis, we have

$$\text{LTE} \leq kL(1-\theta)\left|y(t_{n+1}) - y^{n+1}\right| + \tau(k),$$

where $\tau(k) = |\theta\tau_1(k) + (1-\theta)\tau_2(k)| + |\tau_2(k) - \tau_1(k)|$. If $\theta = 1$, then clearly LTE $\to 0$ as $k \to 0$. Otherwise, for all $k < \frac{1}{L(1-\theta)}$, we have $1 - kL(1-\theta) > 0$, so

$$\text{LTE} \leq \frac{\tau(k)}{1 - kL(1-\theta)}, \qquad k < \frac{1}{1 - kL(1-\theta)}.$$

Hence,

$$0 \leq \lim_{k\to 0} \text{LTE} \leq \lim_{k\to 0} \frac{\tau(k)}{1 - kL(1-\theta)} = 0$$

because $\tau(k) \to 0$ and $1 - kL(1-\theta) \to 1$ as $k \to 0$. Therefore, LTE $\to 0$ as $k \to 0$ for any $\theta \in [0,1]$, and the scheme is consistent.

Problem 2.

Consider the IVP

$$y'(t) = \frac{1}{1+t^2} - 2y^2, \quad t > 0; \qquad y(0) = 0. \tag{7}$$

We will discretize this problem by using scheme 3 from Problem 1 on the interval $[0, 2]$. Note that this scheme is implicit, so the implementation of it is a straightforward generalization of the implementation of the backward Euler method. The main difference is the construction of the implicit function f_n such that $f_n\left(y^{n+1}\right) = 0$.

In the case of IVP (7), we have

$$f(t,y) = \frac{1}{1+t^2} - 2y^2, \qquad a = 0.$$

Rewriting the equation for y^{n+1} in the definition of the scheme, we get

$$y^{n+1} - y^n - kf\left(t_n + (1-\theta)k, \theta y^n + (1-\theta)y^{n+1}\right) = 0, \quad n = 0, 1, \ldots,$$

so we can find y^{n+1} by finding a root of

$$f_n(x) = x - y^n - k\left[\frac{1}{1 + (t_n + (1-\theta)k)^2} - 2(\theta y^n + (1-\theta)x)^2\right].$$

We find this root numerically using Newton's method, which means we need to calculate f_n':

$$f_n'(x) = 1 + 4k(1-\theta)(\theta y^n + (1-\theta)x).$$

If $\{x_j\}$ is the sequence of Newton's method approximations of the root, then we use the stopping criterion $|x_j - x_{j-1}| < 10^{-8}$, where x_j is the returned approximation.

The code for running the scheme with given values of the parameters k and θ is given in `problem2.m`. Note that this refers to `newton.m`, which is the same implementation of Newton's method from the previous homework.

Listing 1: `problem2.m`, which solves IVP (7) using scheme 3

```
function [t, y] = problem2(k, theta)
% Problem 2.
% Implementation of Problem 1 Method 3 for
```

```
%       y' = 1 / (1 + t^2) - 2y^2, t > 0;  y(0) = 0
% on the interval [0, 2].
%
% Parameters
% ----------
%   k: Step size. n = ceil((2 - 0) / k), enough steps to cover [0, 2]
%   theta: Parameter of Method 3 scheme
%
% Return
% ------
%   [t, y]: t is vector of times {t_i}, y is vector
%           of numerical solution values {y^i}.

% initialization
n = ceil(2 / k);
t = linspace(0, 2, n + 1);
y = zeros(1, n + 1);

% initial condition
y(1) = 0;

% Method 3 iteration, solving each step using Newton's method with eps=1e-8
eps = 1e-8;
for i = 1:n
    f_i = @(x) x - y(i) ...
        - k*(1 / (1 + (t(i) + (1-theta)*k)^2) - 2*(theta*y(i) + (1-theta)*x)^2);
    f_i_prime = @(x) 1 + 4*k*(1-theta)*(theta*y(i) + (1-theta)*x);

    y(i + 1) = newton(f_i, f_i_prime, y(i), 100, eps, 0, 0);
end
```

1. Consider the case $\theta = 1$.

   (a) To create a plot of the numerical solution on the interval $[0, 2]$, we need to choose a small enough k value. We choose $k = \frac{1}{2048}$ for consistency with the value used in the reference solution in subsequent parts. The resulting plot is given in Figure 1. Additionally, the numerical value of $y(2)$ is given in `problem2_output.txt` as 0.400024. These results can be obtained by running the following excerpt from `problem2_calculations.m`.

Listing 2: Problem 2.1 (a)

```
%% 2.1 (a)
% What is the numerical value for y(2) (using theta = 1)
fprintf("Running problem 2.1 (a)\n");

% make sure theta = 1
theta = 1;

% Use k = 1/2048 for consistency with the reference solution used later
[t, y] = problem2(1/2048, theta);

% Create plot
fig = figure();
plot(t, y);
xlabel("t");
ylabel("y");
```

```
saveas(fig, "p2_1_plot.eps", "epsc");
```

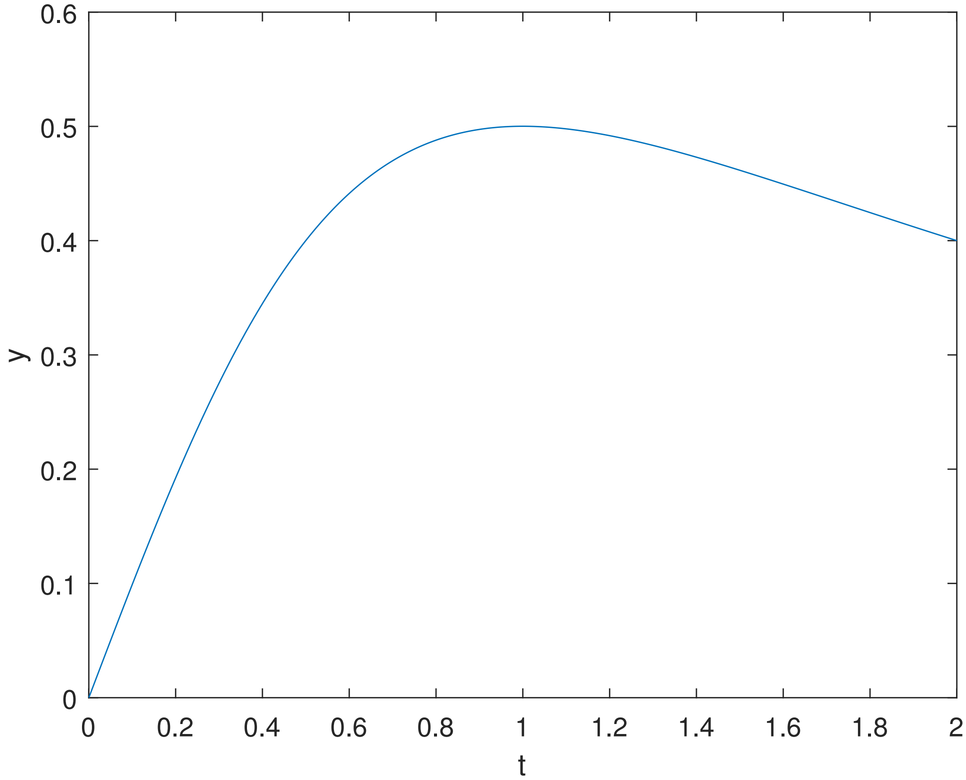

Figure 1: The numerical solution of (7) on $[0, 2]$ with $k = \frac{1}{2048}$ and $\theta = 1$

(b) The following excerpt (Listing 3) from `problem2_calculations.m` computes a reference solution with $k = \frac{1}{2048}$ then calculates the errors at $t = 2$ between the numerical solutions with various step sizes and the reference solution.

The table of values that is printed is given in `p2_output.txt` and copied here for convenience (Table 1).

k	Error at $t = 2$
1/16	0.002761
1/32	0.001436
1/64	0.000722
1/128	0.000353
1/256	0.000166
1/512	0.000071

Table 1: Numerical errors at $t = 2$ when $\theta = 1$

Listing 3: Problem 2.1 (b)

```
%% 2.1 (b)
% Numerical errors at t = 2 for a range of time steps
fprintf("Running problem 2.1 (b)\n");
```

```

% make sure theta = 1
theta = 1;

% Get reference solution (k = 1/2048)
[t_ref, y_ref] = problem2(1/2048, theta);

% Get numerical solutions at t = 2 for range of time steps
k = (1/2).^(4:9);
y_at_2 = zeros(1, length(k));

for i_k = 1:length(k)
    [t, y] = problem2(k(i_k), theta);
    y_at_2(i_k) = y(end);
end

% Calculate errors
errors = abs(y_at_2 - y_ref(end));

% Display table
fprintf("Time step\tError at t = 2\n");
fprintf("-------------------------\n");
for i_k = 1:length(k)
    fprintf("1/%d    \t%f\n", round(1/k(i_k)), errors(i_k));
end
```

(c) We can estimate the convergence rate of the scheme by using a table. Recall that if e_1 and e_2 are the errors with $k = k_1$ and $k = k_2$, then, assuming that error $= Ck^\alpha$, where α is the order, we have

$$\alpha = \frac{\log\left(\frac{e_1}{e_2}\right)}{\log\left(\frac{k_1}{k_2}\right)}.$$

Computing the order this way between consecutive errors in Table 1, we see that the order appears to be 1 (see Table 2). This is expected, considering that scheme 3 is actually just the forward Euler method when $\theta = 1$. The excerpt from `problem2_calculations.m` used to generate this table is given below, and the table itself is copied from `p2_output.txt`.

Listing 4: Problem 2.1 (c)

```
%% 2.1 (c)
% Find the order of convergence based on the results of (b).
fprintf("Running problem 2.1 (c)\n");

% Display convergence rate table
fprintf("Time step\tError at t = 2\tOrder\n");
fprintf("-----------------------------------\n");
fprintf("1/%d    \t%f    \t-  \n", round(1/k(1)), errors(1));
for i_k = 2:length(k)
    fprintf( ...
        "1/%d    \t%f    \t%f\n", ...
        round(1/k(i_k)), ...
        errors(i_k), ...
        log(errors(i_k) / errors(i_k - 1)) / log(k(i_k)/k(i_k - 1)) ...
    );
end
```

k	Error at $t = 2$	Order
1/16	0.002761	-
1/32	0.001436	0.943780
1/64	0.000722	0.991354
1/128	0.000353	1.031922
1/256	0.000166	1.091960
1/512	0.000071	1.218633

Table 2: Order of numerical errors at $t = 2$ when $\theta = 1$

2. Consider the case $\theta = \frac{1}{2}$.

   (a) To create a plot of the numerical solution on the interval $[0, 2]$, we need to choose a small enough k value. We choose $k = \frac{1}{2048}$ for consistency with the value used in the reference solution in subsequent parts. The resulting plot is given in Figure 2. Additionally, the numerical value of $y(2)$ is given in `problem2_output.txt` as 0.400000. I will omit the code for these parts, as it is virtually identical to the code from the previous parts.

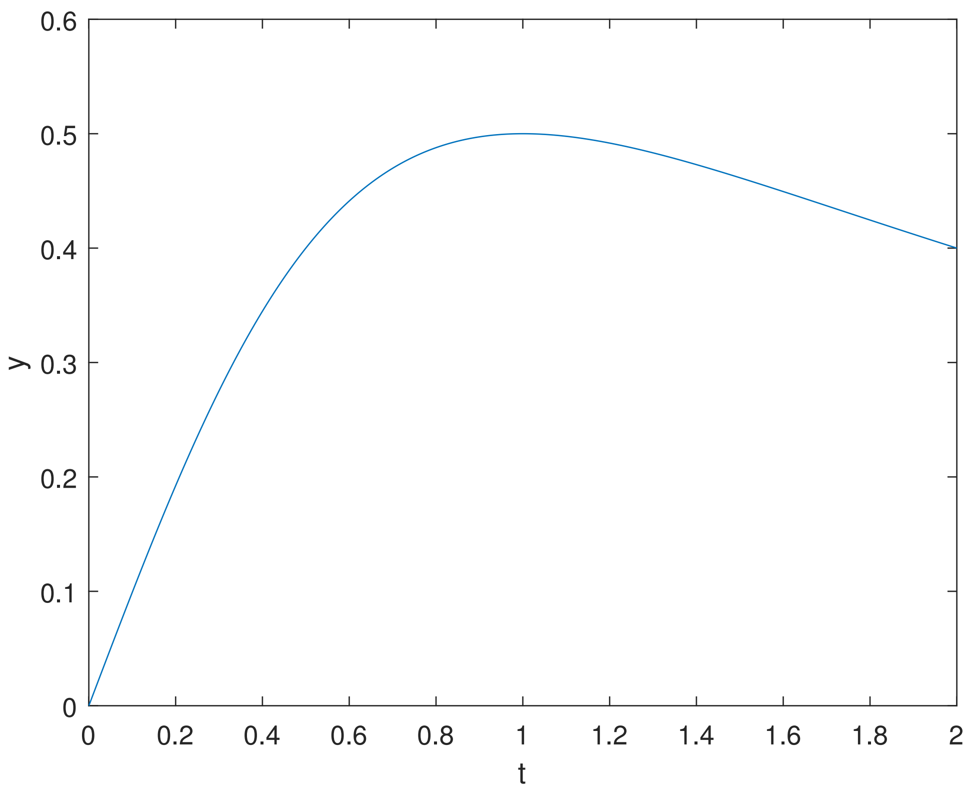

Figure 2: The numerical solution of (7) on $[0, 2]$ with $k = \frac{1}{2048}$ and $\theta = \frac{1}{2}$ – I promise it's a different figure!

   (b) The table of errors computed by `problem2_calculations.m` is given in `p2_output.txt` and copied here for convenience (Table 3). It works the same way as in part 1.

   (c) We can estimate the convergence rate of the scheme by using a table. Recall that if e_1 and e_2 are the errors with $k = k_1$ and $k = k_2$, then, assuming that error $= Ck^\alpha$, where α is the order, we

k	Error at $t = 2$
1/16	4.905053e-05
1/32	1.227079e-05
1/64	3.066097e-06
1/128	7.643168e-07
1/256	1.888337e-07
1/512	4.496056e-08

Table 3: Numerical errors at $t = 2$ when $\theta = \frac{1}{2}$

have

$$\alpha = \frac{\log\left(\frac{e_1}{e_2}\right)}{\log\left(\frac{k_1}{k_2}\right)}.$$

Computing the order this way between consecutive errors in Table 3, we see that the order appears to be 2 (see Table 4, which is copied from `p2_output.txt`).

k	Error at $t = 2$	Order
1/16	4.905053e-05	-
1/32	1.227079e-05	1.999041
1/64	3.066097e-06	2.000752
1/128	7.643168e-07	2.004161
1/256	1.888337e-07	2.017054
1/512	4.496056e-08	2.070385

Table 4: Order of numerical errors at $t = 2$ when $\theta = \frac{1}{2}$