HW #3 Report

Kevin Corcoran

April 24, 2021

1 Problem 1

1.1 Part 1

Derive the stability function $\Phi(z)$ for the following

- Predictor-corrector (Heun's)

$$u_{n+1} = u_n + \frac{h}{2}\left(f(u_n, t_n) + f(u_n + hf(u_n, t_n), t_n)\right).$$

For the model problem, $u'(t) = \gamma u(t)$, $f(u_n, t_n) = \gamma u_n$, and $f(u_n + h\gamma u_n) = \gamma(u_n + h\gamma u_n)$. So we have

$$\begin{aligned} u_{n+1} &= u_n + \frac{h}{2}\left(\gamma u_n + \gamma(u_n + h\gamma u_n)\right) \\ &= \left(1 + h\gamma + \frac{(h\gamma)^2}{2}\right) u_n \end{aligned}$$

Define $z = h\gamma$, then the stability function $\Phi(z)$ is

$$\boxed{\Phi(z) = 1 + z + \frac{z^2}{2}}.$$

- 4-th order Runge-Kutta

For any Runga-Kutta method. Where $\vec{e}$ is a vector of all ones, we can write the general method in vector notation

$$\begin{aligned} \vec{k} &= f(u_n\vec{e} + hA\vec{k}) \\ u_{n+1} &= u_n + h\vec{b}^T\vec{k} \end{aligned}$$

Then for the model problem, we have

$$
\begin{aligned}
\vec{k} &= \gamma u_n \vec{e} + \gamma h A \vec{k} \\
(I - zA)\vec{k} &= \gamma (I - zA)^{-1} u_n \vec{e} \\
\implies \vec{k} &= \gamma (I - zA)^{-1} u_n \vec{e}
\end{aligned}
$$

and then,

$$
\begin{aligned}
u_{n+1} &= u_n + h\vec{b}^T \vec{k} \\
&= u_n + z\vec{b}^T (I - zA)^{-1} u_n \vec{e} \\
&= \left(1 + z\vec{b}^T (I - zA)^{-1} \vec{e}\right) u_n
\end{aligned}
$$

So the stability function $\Phi(z)$ for any RK method is

$$
\boxed{\Phi(z) = 1 + z\vec{b}^T (I - zA)^{-1} \vec{e}}.
$$

1.2 Part 2

Study the zero-stability for each of the two LMMs below

- $$u_{n+2} - 2u_{n+1} + u_n = hf(u_{n+1}, t_{n+1}) - f(u_n, t_n).$$

  This has the following characteristic polynomial $(= 0)$

  $$
  \begin{aligned}
  \rho(\omega) &= \omega^2 - 2\omega + 1 = 0 \\
  &= (\omega - 1)^2 = 0
  \end{aligned}
  $$

  With root $\omega = 1$ of multiplicity 2. Since this is not a simple root, this method is **not** zero stable.

- $$u_{n+2} - u_n = h\left(\frac{1}{3}f(u_{n+2}, t_{n+2}) + \frac{4}{3}f(u_{n+1}, t_{n+1}) + \frac{1}{3}f(u_n, t_n)\right).$$

  This method has characteristic polynomial

  $$\rho(\omega) = \omega^2 - 1.$$

  with roots

  $$\omega = \pm 1.$$

  Since $|w_i| \leq 1$, this method satisfies the root condition, and **is** therefore zero stable.

2 Problem 2

2.1 Part 1

Show that 2s-DIRK is second order for $\alpha = 1 - \frac{1}{\sqrt{2}}$

First order consistency condition

$$\sum_{i=1}^{p=2} b_i = 1$$
$$\implies (1-\alpha) + \alpha = 1$$

Which is satisfied regardless of the value of α.

Second order consistency condition

$$\begin{aligned}\sum_{i=1}^{p=2} b_i c_i &= \frac{1}{2} \\ &= \underbrace{(1-\alpha)}_{b_1}\underbrace{\alpha}_{c_1} + \underbrace{\alpha}_{b_2}\underbrace{(1)}_{c_2} \\ &= \frac{1}{\sqrt{2}}\left(1 - \frac{1}{\sqrt{2}}\right) + 1 - \frac{1}{\sqrt{2}} \\ &= \frac{1}{2}\end{aligned}$$

So for $\alpha = 1 - \frac{1}{\sqrt{2}}$, 2s-DIRK is at least second order.

2.2 Part 2

For the model problem, $u' = \gamma u$, derive the expressions for k_1, k_2, and the stability funciton $\Phi(z)$

$$\begin{aligned}k_1 &= f(u_n + \alpha h k_1, t_n + \alpha h) \\ &= \gamma u_n + \alpha \gamma h k_1 \\ \implies & \boxed{k_1 = \frac{\gamma}{1-\alpha z} u_n}\end{aligned}$$

$$\begin{aligned}k_2 &= f(u_n + h\left((1-\alpha)k_1 + a k_2\right), t_n + h) \\ &= \gamma u_n + z\left((1-\alpha)\underbrace{\frac{\gamma}{1-\alpha z}u_n}_{k_1} + \alpha k_2\right) \\ \implies & \boxed{k_2 = \frac{(1-\alpha z)\gamma + \gamma z(1-\alpha)}{(1-\alpha z)^2} u_n}\end{aligned}$$

Then when h is distributed in $u_{n+1} = u_n + \underbrace{h}((1-\alpha)k_1 + \alpha k_2)$, we get k_1, and k_2 as desired.

The stability function Φ follows from these results. Plugging in k_1 and k_2

$$u_{n+1} = \left(1 + (1-\alpha)\frac{z}{1-\alpha z} + \alpha\left(\frac{(1-\alpha z)z + z^2(1-\alpha)}{(1-\alpha z)^2}\right)\right) u_n.$$

So

$$\Phi(z) = 1 + (1-\alpha)\frac{z}{1-\alpha z} + \alpha\left(\frac{(1-\alpha z)z + z^2(1-\alpha)}{(1-\alpha z)^2}\right).$$

Finding the common denominator and simplifying, we get the result as desired

$$\boxed{\Phi(z) = \frac{1 + (1-2\alpha)z}{(1-\alpha z)^2}}.$$

2.3 Part 3

Suppose 2s-DIRK is A-stable for $\alpha = 1 - \frac{1}{\sqrt{2}}$. Show that it satisfies the second condition of L-stability

We want to show

$$\lim_{z\to\infty} \Phi(z) = 0.$$

Make the change of variable $w = \frac{1}{z}$, then we can equivalently take the limit as $w \to 0$

$$\begin{aligned}
\lim_{w\to 0} \Phi(\frac{1}{w}) &= \lim_{w\to 0} \frac{1 + (1-2\alpha)\frac{1}{w}}{(1-\alpha\frac{1}{w})^2} \\
&= \lim_{w\to 0} \frac{w + (1-2\alpha)}{w\frac{1}{w^2}(w-\alpha)^2} \\
&= \lim_{w\to 0} \frac{w^2 + (1-2\alpha)w}{(w-\alpha)^2} \\
&= 0
\end{aligned}$$

So this method is L-stable.

3 Problem 3

Consider the implicit 2-step method

$$u_{n+2} - u_n = h\left(\frac{1}{3}f(u_{n+2}, t_{n+2}) + \frac{4}{3}f(u_{n+1}, t_{n+1}) + \frac{1}{3}f(u_n, t_n)\right)$$

3.1 Part 1

Show $e_n(h) = O(h^5)$

For the model problem

$$\begin{cases} u'(t) = u(t) \\ u(0) = 1 \end{cases} \implies u(t) = e^t.$$

the local truncation error

$$\begin{aligned} e_n &= \sum_{j=0}^{r} \alpha_j e^{(t_n+jh)} - h \sum_{j=1}^{r} \beta_j e^{(t_n+jh)} \\ &= e^{t_n} \left(\sum \alpha_j e^{jh} - \log(e^{jh}) \sum \beta_j e^{jh} \right) \end{aligned}$$

define $z = e^h$, then the order of the local truncation error can be written in terms of z

$$e_n = O(h^{p+1}) = O(\log(z)^{p+1}) \underbrace{=}_{\text{by Taylor expansion}} O(|z-1|^{p+1}).$$

and more compactly, in terms of characteristic polynomials

$$\begin{aligned} e_n &= \sum \alpha_j z^j - \log(z) \sum \beta_j z^j &(1) \\ &= \rho(z) - \log(z)\sigma(z) &(2) \end{aligned}$$

So now for this problem, we need to check the order of equation (2). The characteristic polynomials are

$$\begin{aligned} \rho(z) &= z^2 - 1 \\ \sigma(z) &= \frac{1}{3}z^2 + \frac{4}{3}z + \frac{1}{3} \end{aligned}$$

let $\zeta = z - 1$, then

$$\begin{aligned} e_n &= \rho(\zeta+1) - \log(\zeta+1)\sigma(\zeta+1) \\ &= (\zeta+1)^2 - 1 - \log(\zeta+1) \left(\frac{1}{3}(\zeta+1)^2 + \frac{4}{3}(\zeta+1) + \frac{1}{3} \right) \end{aligned}$$

Taylor expanding $\log(\zeta+1)$

$$\begin{aligned} e_n &= \zeta^2 + 2\zeta - \left(\left(\zeta - \frac{\zeta^2}{2} + \frac{\zeta^3}{3} - \frac{\zeta^4}{4} + \frac{\zeta^5}{5} + O(\zeta^6) \right) \left(\frac{1}{3}\zeta^2 + 2\zeta + 2 \right) \right) \\ &= \zeta^2 + 2\zeta - \left(2\zeta + \zeta^2(-1+2) + \zeta^3\cancel{(\frac{2}{3} - 1 + \frac{1}{3})} + \zeta^4\cancel{(-\frac{1}{2} + \frac{2}{3} - \frac{1}{6})} + \zeta^5(\frac{2}{5} - \frac{1}{2} + \frac{1}{9}) + O(\zeta^6) \right) \\ &= \frac{1}{90}\zeta^5 + O(\zeta^6) \\ &= \frac{1}{90}(z-1)^5 + O((z-1)^6) \\ &= O(h^5) \end{aligned}$$

This shows that this implicit 2-step method is of order 5.

3.2 Part 2

Find roots of the following for $z = -\epsilon$, $\epsilon > 0$

$$\pi(\xi, z) = (\xi^2 - 1) - z\left(\frac{1}{3}\xi^2 + \frac{4}{3}\xi + \frac{1}{3}\right).$$

Plugging in $z = -\epsilon$ and simplifying

$$\begin{aligned}\xi^2 - 1 + \epsilon\left(\frac{1}{3}\xi^2 + \frac{4}{3}\xi + \frac{1}{3}\right) = 0\\ \xi^2\left(1 + \frac{\epsilon}{3}\right) + \xi\left(\frac{4}{3}\epsilon\right) + \frac{\epsilon}{3} - 1 = 0\end{aligned}$$

Using the quadratic formula and simpifying

$$\begin{aligned}\xi &= \frac{-\frac{4}{3}\epsilon \pm (\sqrt{(\frac{4}{3}\epsilon)^2 - 4(\frac{\epsilon}{3}+1)(\frac{\epsilon}{3}-1)}}{2(1+\frac{\epsilon}{3})}\\ &= -\frac{2}{3}\epsilon\left(\frac{1}{1+\frac{\epsilon}{3}}\right) \pm \frac{\sqrt{\frac{12}{9}\epsilon^2 + 4}}{2(1+\frac{\epsilon}{3})}\\ &= -\frac{2}{3}\epsilon\left(\frac{1}{1+\frac{\epsilon}{3}}\right) \pm \frac{\sqrt{1+\frac{\epsilon}{3}}}{1+\frac{\epsilon}{3}}\\ &= -\frac{2}{3}\epsilon\left(\frac{1}{1+\frac{\epsilon}{3}}\right) \pm \frac{1}{\sqrt{1+\frac{\epsilon}{3}}}\end{aligned}$$

Now taylor expanding,

$$\xi = -\frac{2}{3}\epsilon\left(1 - \frac{\epsilon}{3} + O(\epsilon^2)\right) \pm 1 - \frac{\epsilon}{6} + O(\epsilon^2)$$

Gives us the roots

$$\xi_1(\epsilon) = 1 - \frac{5}{6}\epsilon + O(\epsilon^2), \qquad \xi_2 = -\left(1 + \frac{1}{3}\right) + O(\epsilon^2).$$

Since $|\xi_i| \not\leq 1$, the root condition is not satisfied, and so $z = -\epsilon$ is **not** in the region of absolute stability.

4 Including required packages

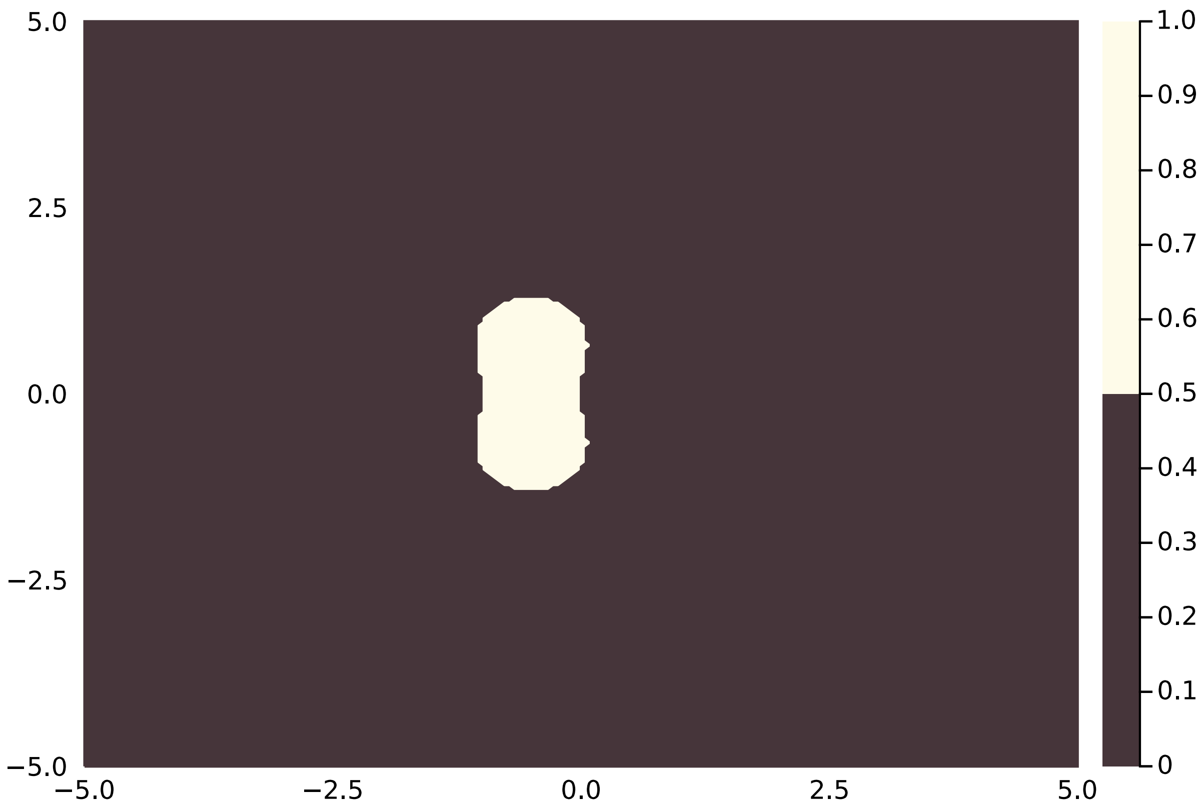
```
using Plots
using LaTeXStrings
theme(:mute)

using Pkg
```

```
Pkg.activate("RAS")
include("code/RAS.jl") # Makes sure the module is run before using it
using .RAS: RAS_stabf, RASrk

Pkg.activate("DiffyQ")
include("code/DiffyQ.jl") # Makes sure the module is run before using it
using .DiffyQ: s2_DIRK, BackwardEuler_n
```

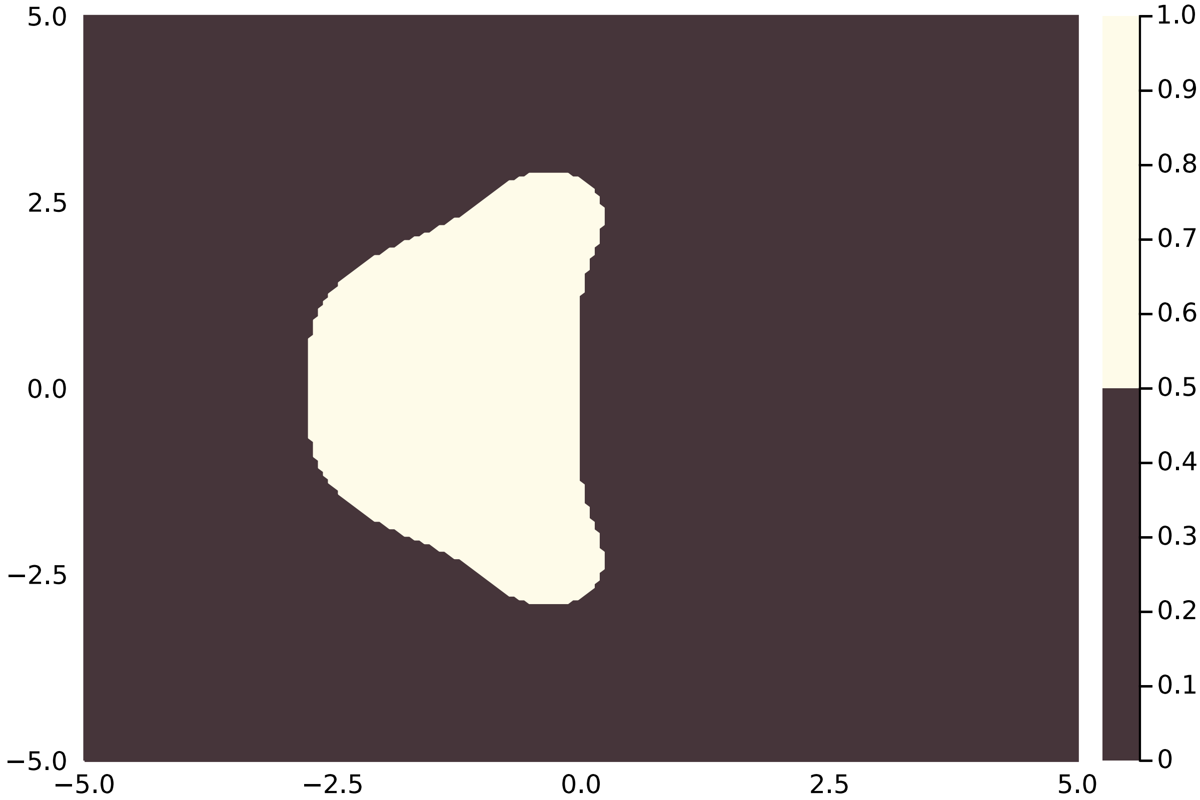
5 Problem 4: Plot Region of Absolute Stability

Note that "light" color is the region of stability

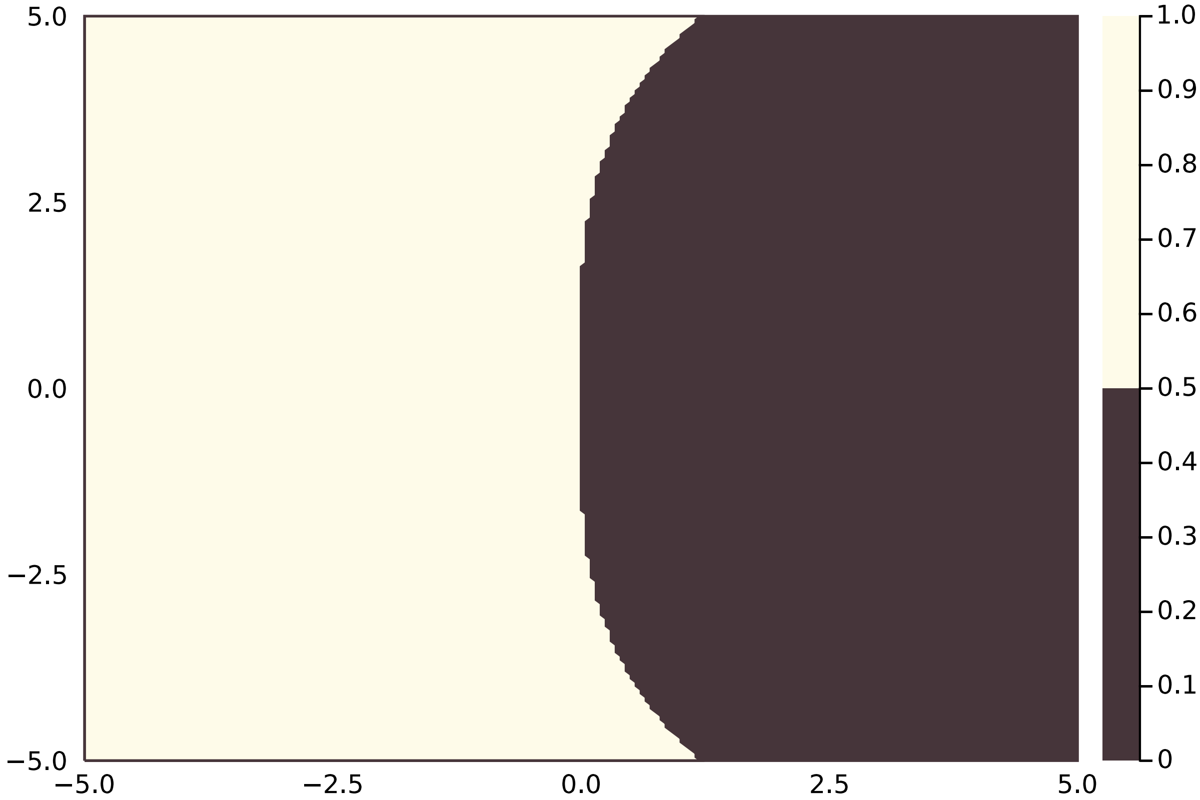
```
# stability function for Heun's
Φ(z) = 1.0 + z + z^2
xs, Z = RAS_stabf(Φ)

contourf(xs, xs, Z, levels = 1)
```



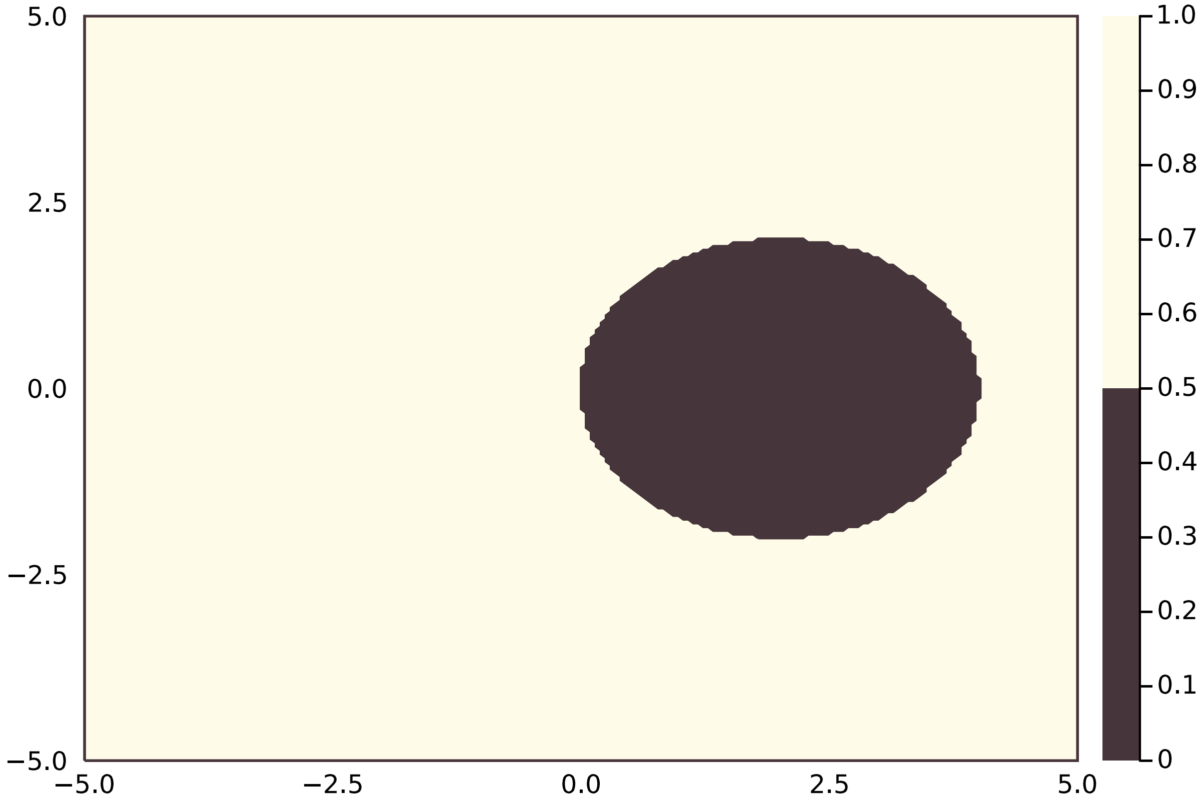

```
# RK4
A = [0 0 0 0
    1/2 0 0 0
    0 1/2 0 0
    0 0 1 0]
b = [1/6, 1/3, 1/3, 1/6]

xs, Z = RASrk(A,b)
# plotly()
contourf(xs,xs,Z, levels = 1)
```




```
# 2s-DIRK
α = 1-1/sqrt(2)
A = [α 0
     1-α α]
b = [1-α, α]

xs, Z = RASrk(A,b)
contourf(xs,xs,Z, levels = 1)
```




```
# 2s-DIRK
α = 0.5
A = [α 0
     1-α α]
b = [1-α, α]

xs, Z = RASrk(A,b)
contourf(xs,xs,Z, levels = 1)
```

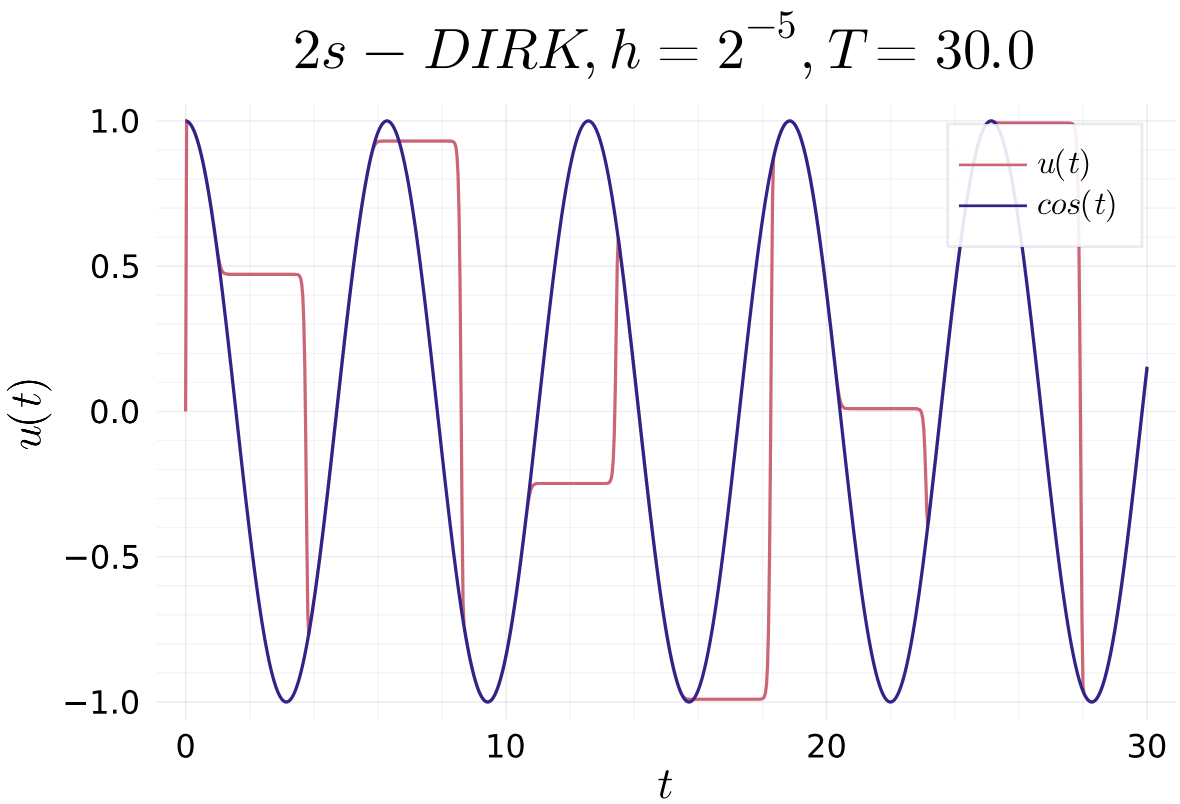


6 Problem 5

```
f(u,t,μ) = -(0.5*exp(20*cos(1.3*t)) * sinh(u-cos(t)));

α = 1 - 1/sqrt(2);
T = 30.0; h = 2.0^(-5); N = Int(T/h);
u0 = 0.0;

u = s2_DIRK(f, N, T, u0, α);
```

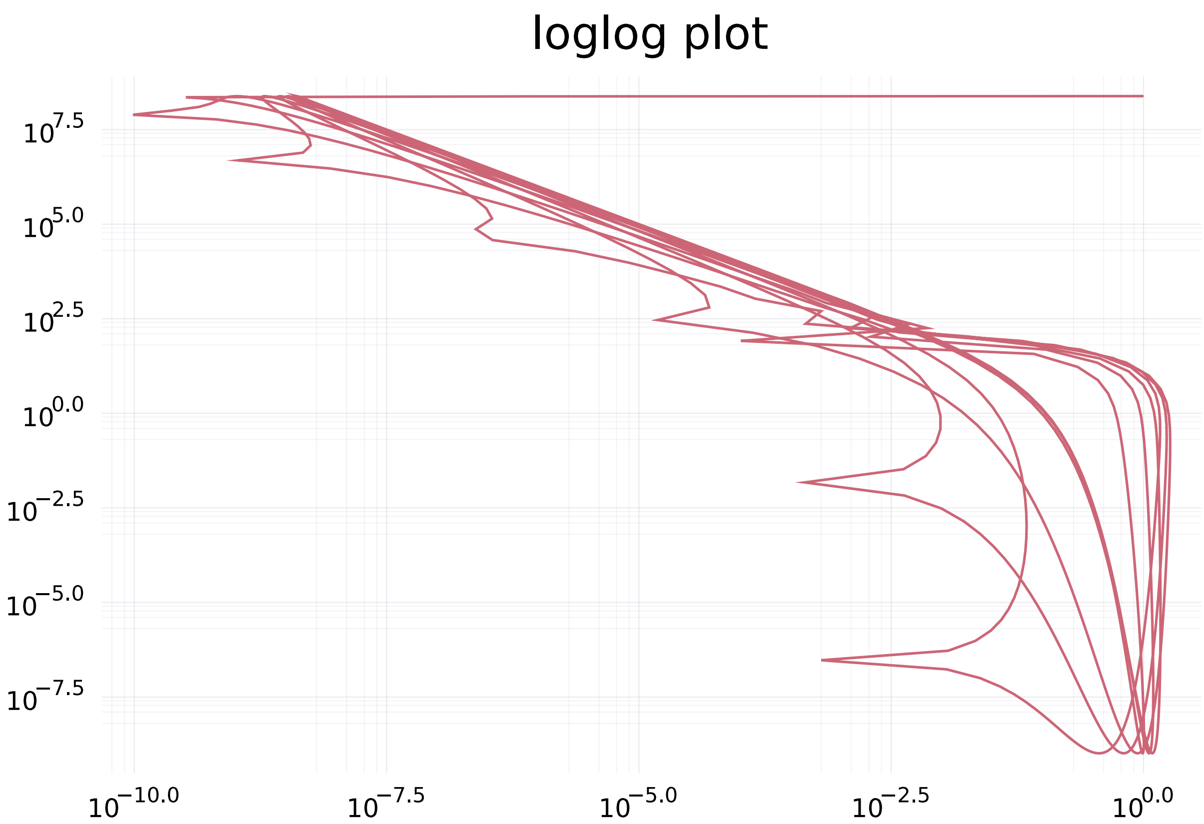
6.1 Part 1

```
tList = collect(0:N)*(T/N)
plot(tList, u, label = L"u(t)", thickness_scaling =1.25)
xlabel!(L"t")
ylabel!(L"u(t)")
title!(latexstring("2s-DIRK,h=2^{-5},T=",T))
plot!(tList, cos.(tList), label = L"cos(t)")
```



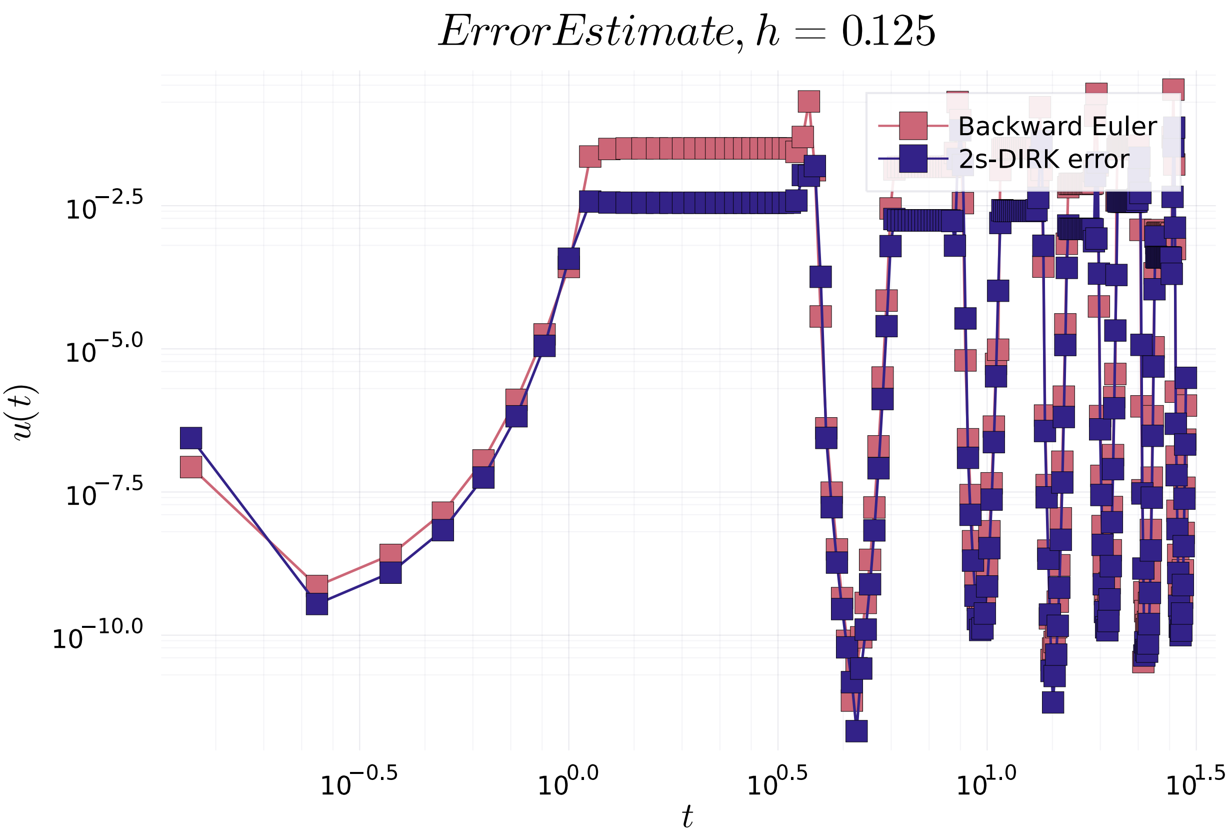
6.2 Part 2

```
g(t) = 0.5*exp(20*cos(1.3*t))
plot(abs.(u - cos.(tList)), g.(tList), xaxis=:log, yaxis=:log, legend = false)
title!("loglog plot")
```

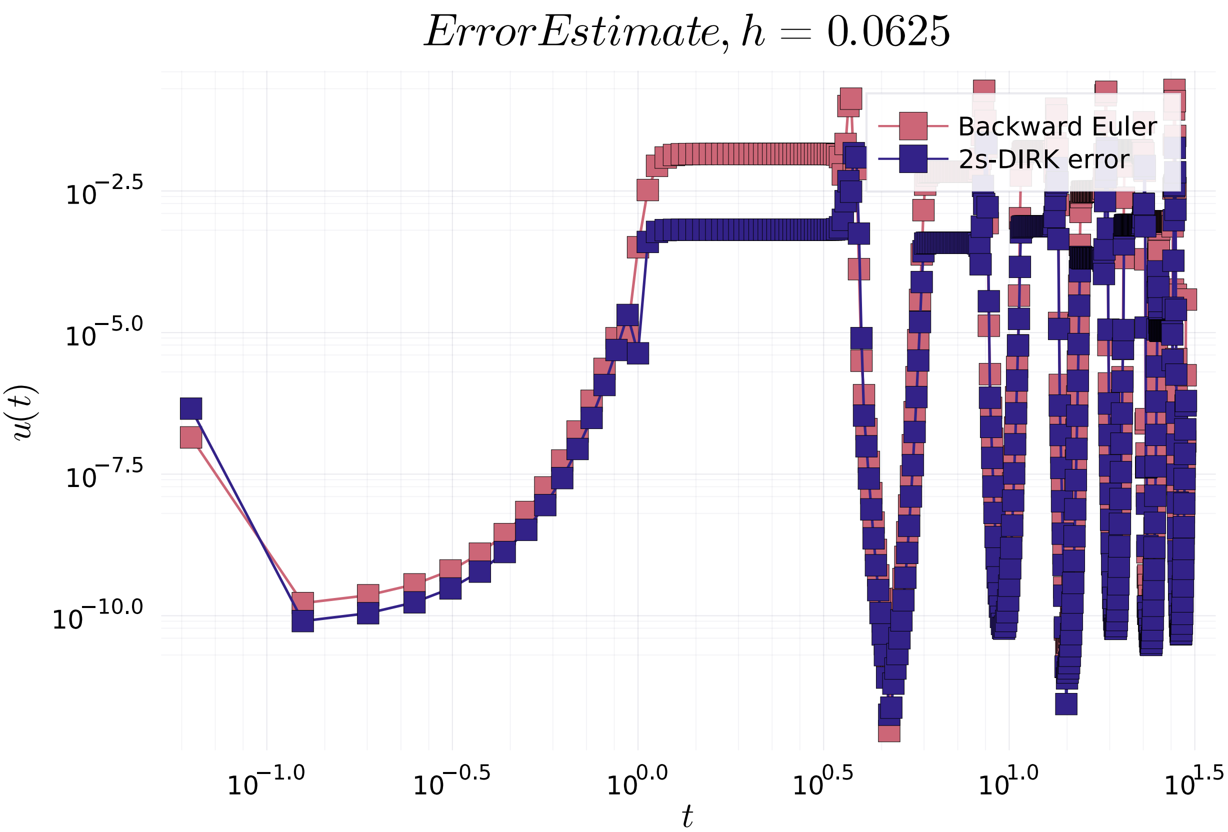


7 Problem 6

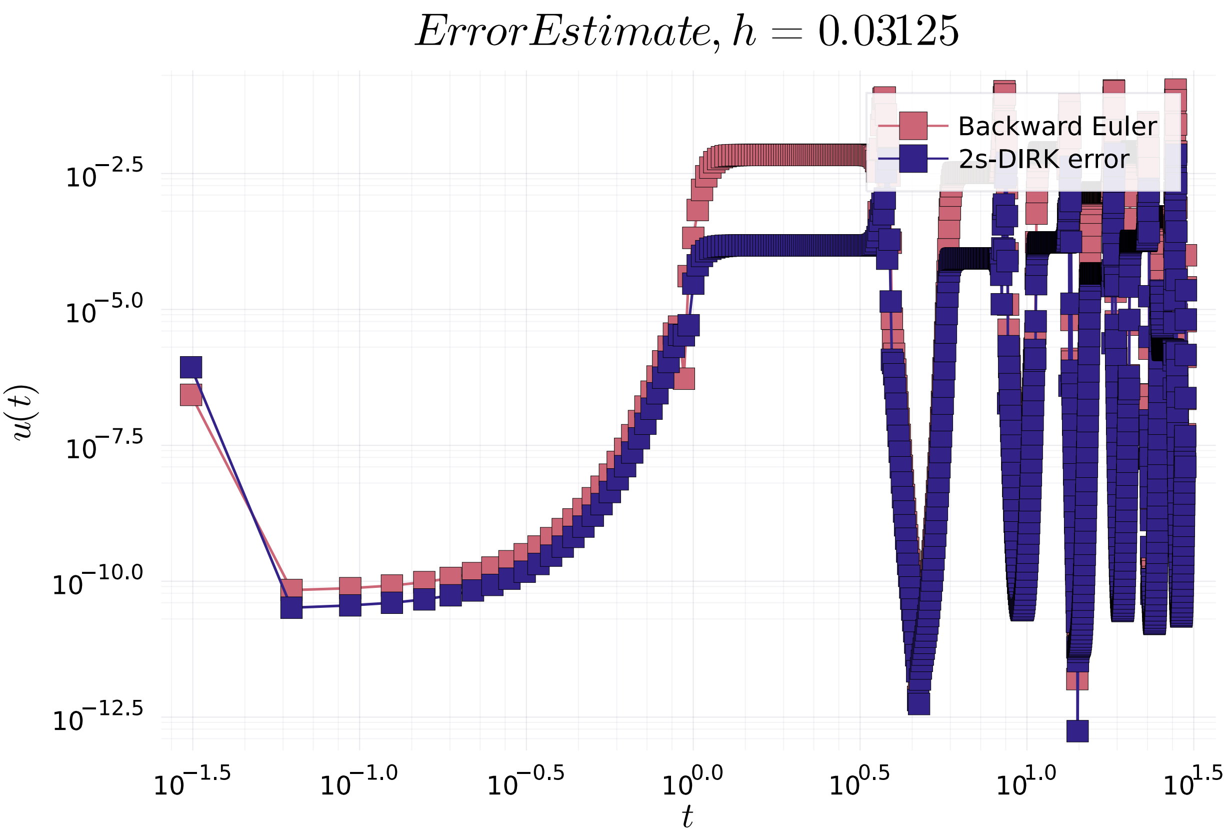
```
hs = 1 ./(2 .^(3:8))
for i = 1 : length(hs)
    N = Int(T/hs[i])
    tList = collect(0:N)*(T/N)

    ## Backward Euler
    u_euler = BackwardEuler_n(f, N, T, u0)
    u_eexact = BackwardEuler_n(f,2*N,T,u0)
    euler_error = abs.(u_euler[1:N] - u_eexact[1:2:2*N])./(1-0.5^1) # first order method
    p1 = plot(tList[2:N], euler_error[2:N], label = "Backward Euler", xaxis=:log,
yaxis=:log, marker = (:square,5))
    xaxis!(L"t")
    yaxis!(L"u(t)")
    title!(latexstring("Error Estimate,h=",hs[i]))

    ## s2-DIRK
    u_s2_DIRK = s2_DIRK(f, N, T, u0, α)
    u_Dexact = s2_DIRK(f, 2*N, T, u0, α)
    DIRK_error = abs.(u_s2_DIRK[1:N] - u_Dexact[1:2:2*N])./(1-0.5^2) # second order
method
    p2 = plot!(tList[2:N], DIRK_error[2:N], label = "2s-DIRK error", xaxis=:log,
yaxis=:log, marker = (:square,5))
    display(p2)
end
```



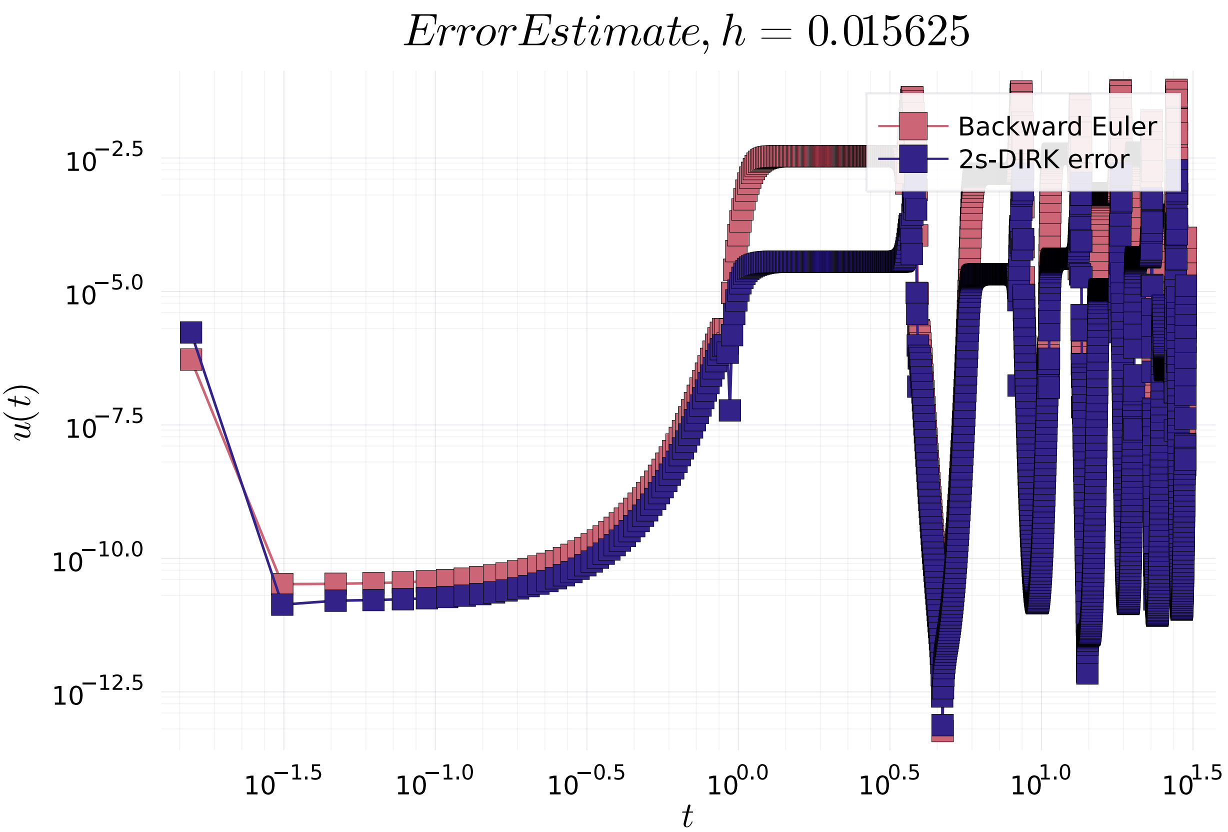
$ErrorEstimate, h = 0.125$
Backward Euler
2s-DIRK error
$10^{-2.5}$
$10^{-5.0}$
$10^{-7.5}$
$10^{-10.0}$
$u(t)$
$10^{-0.5}$
$10^{0.0}$
$10^{0.5}$
$10^{1.0}$
$10^{1.5}$
t



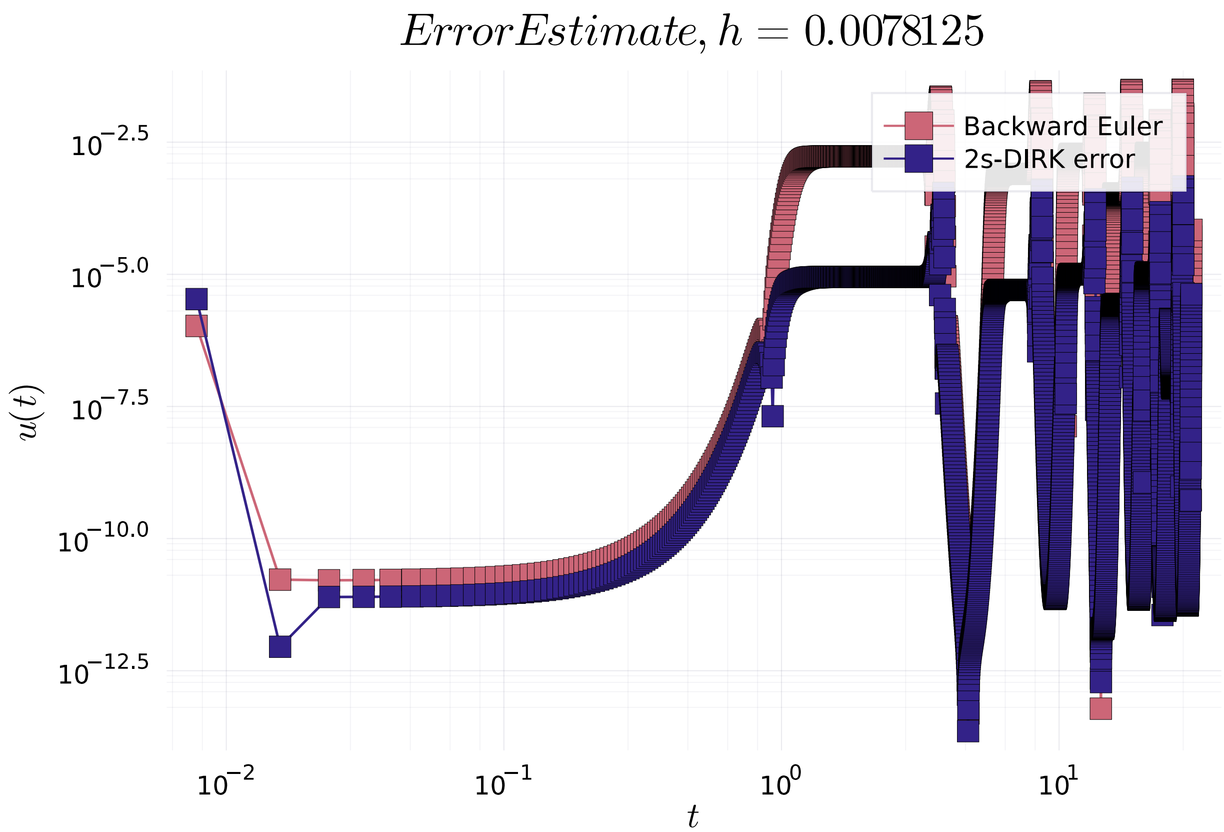
$ErrorEstimate, h = 0.0625$
Backward Euler
2s-DIRK error
$10^{-2.5}$
$10^{-5.0}$
$10^{-7.5}$
$10^{-10.0}$
$u(t)$
$10^{-1.0}$
$10^{-0.5}$
$10^{0.0}$
$10^{0.5}$
$10^{1.0}$
$10^{1.5}$
t

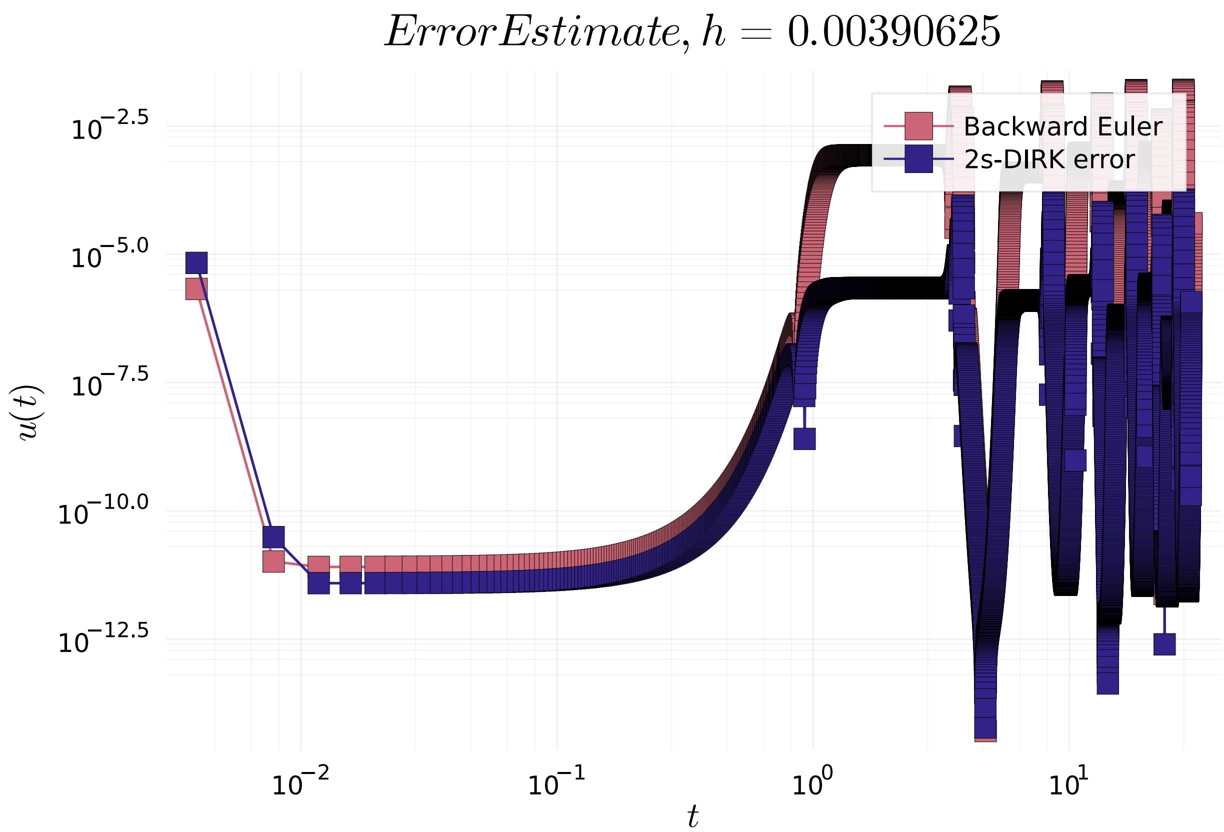


$Error Estimate, h = 0.03125$
Backward Euler
2s-DIRK error
$10^{-2.5}$
$10^{-5.0}$
$10^{-7.5}$
$10^{-10.0}$
$10^{-12.5}$
$u(t)$
$10^{-1.5}$
$10^{-1.0}$
$10^{-0.5}$
$10^{0.0}$
$10^{0.5}$
$10^{1.0}$
$10^{1.5}$
t



$Error Estimate, h = 0.015625$
Backward Euler
2s-DIRK error
$10^{-2.5}$
$10^{-5.0}$
$10^{-7.5}$
$10^{-10.0}$
$10^{-12.5}$
$u(t)$
$10^{-1.5}$
$10^{-1.0}$
$10^{-0.5}$
$10^{0.0}$
$10^{0.5}$
$10^{1.0}$
$10^{1.5}$
t





7.0.1 Part 2

```
h = 2.0^(-7)
N = Int(T/h)
tList = collect(0:N)*(T/N)
```

```
## Backward Euler
u_euler = BackwardEuler_n(f, N, T, u0)
u_eexact = BackwardEuler_n(f,2*N,T,u0)
euler_error = abs.(u_euler[1:N] - u_eexact[1:2:2*N])./(1-0.5^1) # first order method
p1 = plot(tList[2:N], euler_error[2:N], label = "Backward Euler", xaxis=:log,
yaxis=:log, marker = (:square,5))
xaxis!(L"t")
yaxis!(L"u(t)")
title!(latexstring("Error Estimate,h=",h))

## s2-DIRK
u_s2_DIRK = s2_DIRK(f, N, T, u0, α)
u_Dexact = s2_DIRK(f, 2*N, T, u0, α)
DIRK_error = abs.(u_s2_DIRK[1:N] - u_Dexact[1:2:2*N])./(1-0.5^2) # second order method
p2 = plot!(tList[2:N], DIRK_error[2:N], label = "2s-DIRK error", xaxis=:log, yaxis=:log,
marker = (:square,5))
display(p2)
```