Coding Exercise 1 Solution: Solving the Continuous-Time Consumption Euler Equation

ECON 202A

October 19, 2024

1. Analytical Solution Using the Integrating Factor Method

Step 1: Write the ODE in Standard Form
We rewrite the equation in its standard form:

$$\dot{C}_t - \frac{(r_t - \rho)}{\theta} C_t = 0$$

which corresponds to the form $\dot{y}(t) + p(t)y(t) = 0$, where $p(t) = -\frac{(r_t-\rho)}{\theta}$.

Step 2: Find the Integrating Factor
The integrating factor $\mu(t)$ is:

$$\mu(t) = e^{\int p(t)\,dt} = e^{-\int \frac{(r_t-\rho)}{\theta}\,dt}$$

Step 3: Multiply the ODE by the Integrating Factor
Multiply both sides of the equation by $\mu(t) = e^{-\int \frac{(r_t-\rho)}{\theta}\,dt}$:

$$e^{-\int \frac{(r_t-\rho)}{\theta}\,dt} \cdot \left(\dot{C}_t - \frac{(r_t - \rho)}{\theta} C_t \right) = 0$$

This simplifies the left-hand side to:

$$\frac{d}{dt} \left(C_t e^{-\int \frac{(r_t-\rho)}{\theta}\,dt} \right) = 0$$

Step 4: Integrate Both Sides
Integrate both sides with respect to t:

$$C_t e^{-\int \frac{(r_t-\rho)}{\theta}\,dt} = C$$

where C is the constant of integration.

Step 5: Solve for C_t
Solve for C_t by dividing both sides by $e^{-\int \frac{(r_t-\rho)}{\theta} dt}$:

$$C_t = e^{\int \frac{(r_t-\rho)}{\theta} dt} C$$

Step 6: Determine the Constant of Integration
Use the terminal condition C_T to find C:

$$C_T = e^{\int_0^T \frac{(r_t-\rho)}{\theta} dt} C$$

Thus, $C = e^{-\int_0^T \frac{(r_t-\rho)}{\theta} dt} C_T$.

Final Analytical Solution
The analytical solution to the consumption Euler equation is:

$$C_t = C_T e^{-\int_t^T \frac{(r_s-\rho)}{\theta} ds}$$

where $r_t = r_0 + \alpha t$.

Substituting r_t into the integral, we get:

$$-\int_t^T \frac{(r_s-\rho)}{\theta} ds = \frac{1}{\theta}\left[(r_0-\rho)(t-T) + \frac{\alpha}{2}(t^2-T^2)\right]$$

Therefore, the consumption path is:

$$C_t = C_T e^{\frac{1}{\theta}\left[(r_0-\rho)(t-T)+\frac{\alpha}{2}(t^2-T^2)\right]}$$

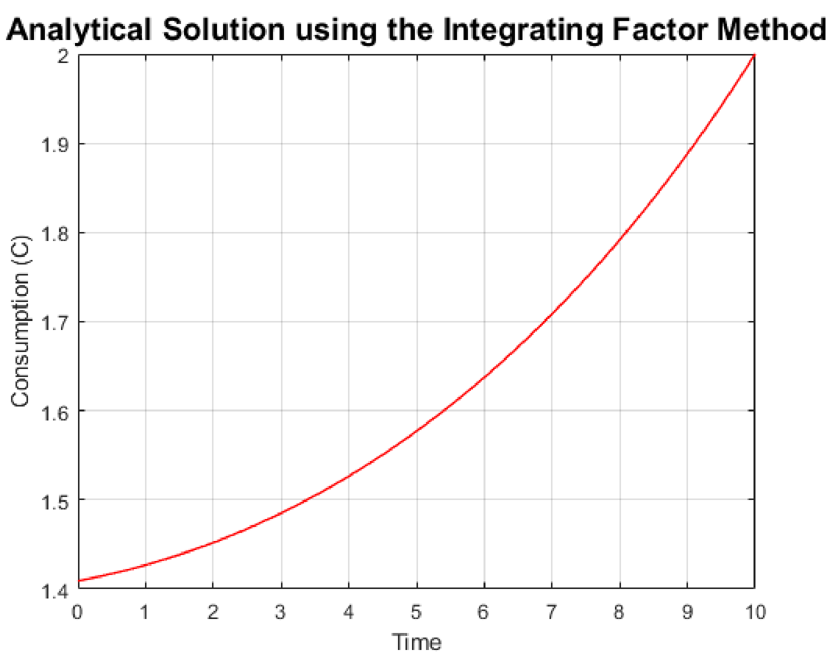

Figure 1: Analytical Solution Using the Integrating Factor Method

2. Numerical Solution Using the Backward Difference Method

Since we are given a terminal boundary condition, we will use the **backward difference method** to solve this equation. The update rule is:

$$C(t - \Delta t) = \frac{C(t)}{1 + \frac{\Delta t}{\theta}\left(r(t) - \rho\right)}$$

where Δt is the time step size. This equation is iterated backward in time, starting from T and progressing to 0.

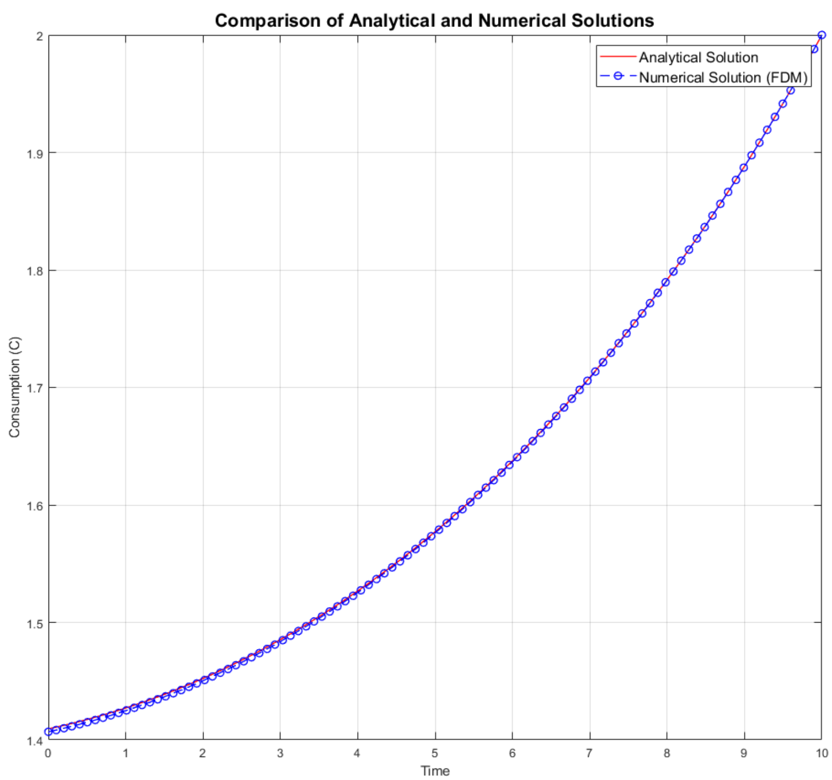

Figure 2: Numerical vs Analytical Solution with 100 Time Steps

3. Comparison Between Numerical and Analytical Solutions

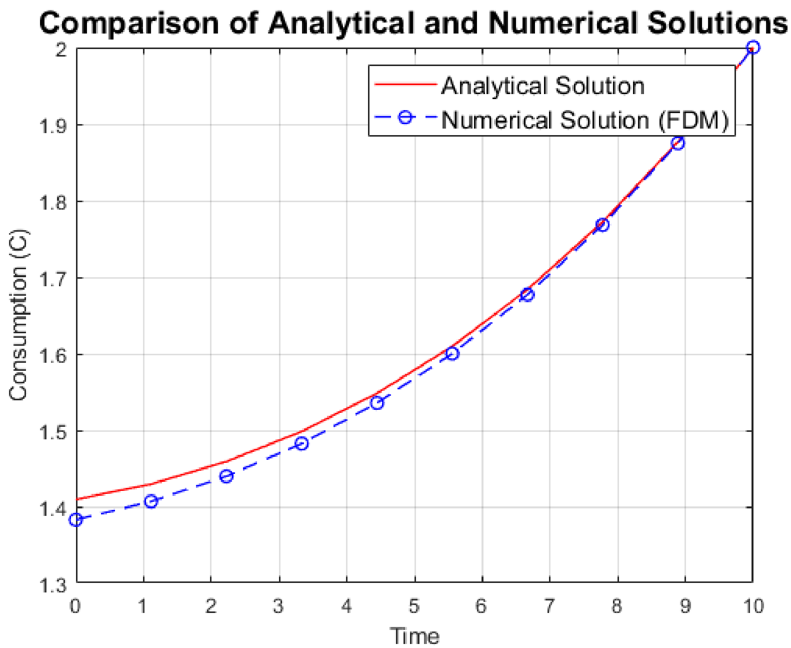

Figure 3: Numerical vs Analytical Solution with 10 Time Steps

4. Discussion on Time Step Size

As the time step size becomes larger (i.e., the grid becomes coarser), the accuracy of the numerical solution decreases. The accuracy is highly dependent on the fineness of the grid. Smaller time steps yield more accurate results, while larger time steps lead to higher numerical errors.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% MATLAB Script: Euler_equation.m
%
% Author: Kiyea Jin
% Date: October 18, 2024
%
% Description:
% This script solves the continuous-time consumption Euler equation:
%   (dC(t)/dt)/C(t) = (r(t)-rho)/theta
% where the interest rate follows r(t) = r(0) + alpha * t,
% and a terminal boundary condition for consumption C(T) is provided.
%
% The solutions are obtained using:
% 1. The integrating factor method for the analytical solution.
% 2. The finite difference method for the numerical solution.
%
% The script also:
% 3. Plots both the analytical and numerical solutions for comparison.
% 4. Explores how the numerical solution varies with grid fineness.
%
% Parameters:
% - Initial interest rate, r(0): 0.05
% - Rate of change in interest rate, alpha: 0.01
% - Intertemporal elasticity of substitution (theta): 2.0
% - Time discount factor (rho): 0.03
% - Terminal time (T): 10
% - Terminal consumption condition, C(T): 2.0
%
% Code Structure:
% 1. DEFINE PARAMETERS
% 2. INITIALIZE GRID POINTS
% 3. ANALYTICAL SOLUTION: INTEGRATING FACTOR METHOD
% 4. NUMERICAL SOLUTION: FINITE DIFFERENCE METHOD
% 5. PLOT
% (6. SOLVE USING ODE45)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

clear all;
close all;
clc;

%% 1. DEFINE PARAMETERS

p = define_parameters();

%% 2. INITIALIZE GRID POINTS

t = linspace(p.tmin, p.tmax, p.I)';
dt = (p.tmax-p.tmin)/(p.I-1);

%% 3. ANALYTICAL SOLUTION: INTEGRATING FACTOR METHOD

% 3-1. Pre-allocate arrays for solutions
```

```
C_analytical = zeros(p.I,1);

% 3-2. Analytical solution:
% c(t) = C(T)*exp(1/theta*((r0-rho)(t-T)+alpha/2*(t^2-T^2)))

for i=1:p.I

    C_analytical(i) = p.CT * exp(1/p.theta*( (p.r0-p.rho)*(t(i)-p.tmax) + p.alpha/2*(t

end

%% 4. NUMERICAL SOLUTION: FINITE DIFFERENCE METHOD

% 4-1. Pre-allocate arrays for solutions
C_numerical = zeros(p.I,1);
C_numerical(end) = p.CT;

% 4-2. Numerical solution: (C(t)-C(t-dt))/dt = C(t-dt)*(r(t)-rho)/theta

% Define interest rates series
r = p.r0 + p.alpha*t;

for i=p.I:-1:2

    C_numerical(i-1) = C_numerical(i) / (1 + dt/p.theta*(r(i)-p.rho));

end

%% 5. PLOT

% Compute the difference between two solutions
error = C_analytical - C_numerical;

% 2. Plot the analytical solution
figure;
plot(t, C_analytical, 'r-', 'Linewidth', 1); hold on;
xlabel('Time', 'FontSize', 11);
ylabel('Consumption (C)', 'FontSize', 11);
title('Analytical Solution using the Integrating Factor Method', 'FontSize', 15);
grid on;

% 3-4. Plot the numerical and analytical solutions together
figure;
plot(t, C_analytical, 'r-', 'LineWidth', 1); hold on;
plot(t, C_numerical, 'bo--', 'LineWidth', 1);
xlabel('Time', 'FontSize', 11);
ylabel('Consumption (C)', 'FontSize', 11);
legend('Analytical Solution', 'Numerical Solution (FDM)', 'FontSize', 12);
title('Comparison of Analytical and Numerical Solutions', 'FontSize', 15);
grid on;
```

```
function p = define_parameters()
```

```
% This function defines the parameters needed for the Euler_equation.m script

%% Economic Parameters

    % Initial interest rate
    p.r0 = 0.05;

    % Rate of change in interest rate
    p.alpha = 0.01;

    % Intertemporal elasticity of substitution
    p.theta = 2.0;

    % Time discount factor
    p.rho = 0.03;

%% Boundary Conditions

    % Terminal consumption
    p.CT = 2.0;

%% Grid Paramters

    p.tmin = 0;
    p.tmax = 10;

    % The number of time steps
    p.I = 10;

end
```