Summary of Numerical Methods

1 Shooting Method

The shooting method is a numerical technique used to solve boundary value problems (BVPs). It transforms the BVP into an initial value problem (IVP) by guessing the initial values (often the initial derivative) and integrating the system of ordinary differential equations (ODEs) until the boundary conditions are satisfied. The key steps include:

1. **Initial Guess**: Make an initial guess for the unknown initial condition.
2. **Integration**: Use a method (like Euler's method) to integrate the ODEs from the initial point to the boundary.
3. **Check Boundary Conditions**: Compare the computed value at the boundary with the desired boundary condition.
4. **Refinement**: Adjust the initial guess based on whether the computed value is too high or too low (using methods like bisection or Newton's method) and repeat until the solution converges.

2 Explicit Euler Method

The explicit Euler method is a simple numerical technique for solving first-order ordinary differential equations (ODEs). It is based on the principle of using the derivative (slope) at the current point to estimate the next value. The key steps include:

1. **Initialize Values**: Start with an initial condition for the function.
2. **Update Rule**: For each time step:

   $$y_{n+1} = y_n + h \cdot f(t_n, y_n)$$

   where h is the step size, and f is the function representing the derivative.
3. **Iterate**: Repeat the update rule until reaching the desired endpoint.

3 Implicit Euler Method

The implicit Euler method is a numerical technique that differs from the explicit Euler method in that it uses the derivative at the next time point rather than the current time point. This makes it more stable for stiff equations. The key steps include:

1. **Initialize Values**: Start with an initial condition for the function.
2. **Update Rule**: For each time step:

   $$y_{n+1} = y_n + h \cdot f(t_{n+1}, y_{n+1})$$

   This often requires solving a system of equations (e.g., using fixed-point iteration or Newton's method).
3. **Iterate**: Repeat the update rule until reaching the desired endpoint.

4 Jacobian

The Jacobian is a matrix of first-order partial derivatives of a vector-valued function. It provides crucial information about the behavior of functions of multiple variables. In the context of numerical methods:

- **Definition**: If $\mathbf{F}(\mathbf{x}) = [F_1(x_1, x_2, \ldots), F_2(x_1, x_2, \ldots), \ldots]$ is a vector function, the Jacobian matrix J is defined as:

  $$J = \begin{bmatrix} \frac{\partial F_1}{\partial x_1} & \frac{\partial F_1}{\partial x_2} & \cdots \\ \frac{\partial F_2}{\partial x_1} & \frac{\partial F_2}{\partial x_2} & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix}$$

- **Use in Numerical Methods**: The Jacobian is essential in analyzing the stability and convergence of numerical methods and is often used in implicit methods to solve systems of nonlinear equations.