**Solve Parabolic PDE Function Documentation**

**Description**

The **solveParabolicPDE** function numerically solves the one-dimensional parabolic partial differential equation (PDE), commonly known as the heat equation, using an explicit finite difference method. It discretizes both the spatial domain and the time domain and iteratively computes the solution over time.

**Input Arguments**

* **L**: Length of the spatial domain (1D) over which the PDE is solved.
* **T**: Total time for which the solution is computed.
* **Nx**: Number of spatial grid points (nodes) excluding the boundaries.
* **Nt**: Number of time steps.
* **alpha**: Thermal diffusivity coefficient.
* **u\_left**: Boundary condition at the left boundary.
* **u\_right**: Boundary condition at the right boundary.
* **u0**: Initial condition function handle.

**Output Argument**

* **u**: Matrix representing the computed solution of the PDE. Each column corresponds to a time step, and each row corresponds to a spatial grid point.

**Method Explanation**

1. **Initialization**: Initialize parameters such as spatial and temporal discretization steps, spatial grid points, and construct the grid in the spatial domain.
2. **Construct Coefficient Matrix**: Call the **constructParabolicMatrix** function to construct the coefficient matrix 𝐴*A* representing the discretized form of the parabolic PDE.
3. **Initial Condition**: Compute the initial condition 𝑢0(𝑥)*u*0​(*x*) using the provided function handle.
4. **Time Stepping**: Iterate overtime steps and update the solution using the explicit finite difference method. At each time step, boundary conditions are applied explicitly.
5. **Visualization**: Visualize the computed solution using a surface plot, where the x-axis represents spatial coordinates, the y-axis represents time, and the z-axis represents the solution value 𝑢*u*.