

Assignment 2, Report

Maximilian Mayerl, Alexander Schlögl, Benedikt Wimmer

November 29, 2017

1 Implementation

The implementation was carried out in four steps, as suggested by the exercise sheet.

1.1 Computing Partial Derivatives

The first step was to compute the partial derivatives of the linear basis functions. This was implemented in `LinTriElement::ComputeBasisDeriv`. Our approach works as follows:

1. Get the three vectors (x_1, y_1) , (x_2, y_2) and (x_3, y_3) from the `FEModel` via its `GetNodePosition` method.
2. Put the three vectors into a `Matrix3x3` in order to solve the corresponding linear equation.
3. Solve the linear system by inverting the matrix via the `Matrix3x3::Inverse` method.

The partial derivatives were then simply recovered from the inverted matrix.

1.2 Assembling Global Stiffness Matrix

The next step was to assemble the global stiffness matrix. This was implemented in `LinTriElement::AssembleElement`.

For this, we first calculate the derivatives of the basis functions by calling the method implemented in 1.1. After that, we iterate over the elements of the vector of partial derivatives with two nested loops to get every combination of the derivatives. An individual contribution is then calculated as shown in the exercise sheet.

1.3 Determining Right-Hand Side Vector

The third step was to determine the right-hand side vector of the `FEModel`. This was implemented in `FEModel::ComputeRHS`. Our solution for this task works as follows:

1. Clear the right-hand side vector by filling it with zeros.
2. For every node, check for every triangle if it contains that node.
3. If yes, add $A_e \cdot f(x_c, y_c) \cdot N_j(x_c, y_c)$.

1.4 Calculate Error

The last step was to determine the accuracy of our implementation. This was implemented in `FEModel::ComputeError`. For this, we first determine the absolute error vector between the numerical solution and the analytical solution. After that, we calculate the inner product error norm based on this absolute error vector.

2 Results

The visualization produced by the provided framework for our solution looks as shown in Figure 1.

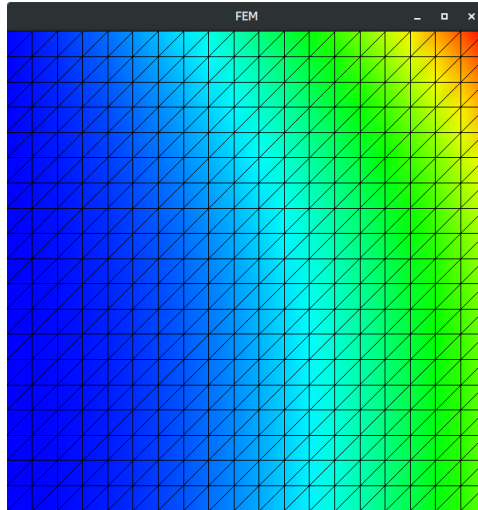


Figure 1: Visualization of our result.

The visualization produced by the provided framework for the error of our solution looks as shown in Figure 2.

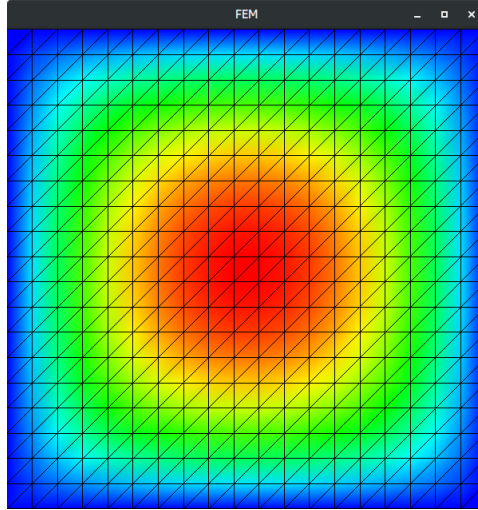


Figure 2: Visualization of our error.

The error norm was calculated as $3.55799 \cdot 10^{-10}$.