Compute Element Matrix Array

- Next we compute the element matrices and fill them into the array
- for(unsigned int t = 0; t < T; ++t)
- {
    - Es[t].m_blocks[0][0] = … compute E_ii….;
    - Es[t].m_blocks[0][1] = … compute E_ij….
    - …
    - Es[t].m_blocks[2][2] = … compute E_kk….
- }

Do Matrix Assembly

- For getting easy started a matrix data type is provided in util_coo_matrix.h header file and for working with GLUE the header file glue_matrix_assembly.h makes it easy to transform a element matrix array into a global matrix
  - bool interlaced = true;
  - util::COOMatrix<double> A = glue::matrix_assembly<double>(omega, Es,interlaced)
- The last argument controls how the memory layout is generated. That is how x and y coordinates are mapped to global matrix indices.