

Implementation

Group 5

November 2023

1 Problem Presentation

The FEM method is used to modelize and resolve a real physics problem, our goal is to find the deflection vector $u(x)$. We have modeled the problem and it consists into a linear problem of the form: $A \cdot \vec{u} = \vec{b}$.

1. Our job is to implement in Julia programming language a function that solves the linear system. Julia provides an easy solution to that problem: the "`\`" operator. So we implemented a Type safe function that takes the matrix A and the vector \vec{b} with floating values, and returns the vector \vec{u} as a float. To control the type-safety of the function we used the macro *@code_warntype*.

2. Our second job was to compare the time complexity required to solve linear problem in which the A matrix changed its form using the `@elapsed` macro: firstly we have considered a matrix stored in memory as a sparse matrix and then as a full matrix. To build the Matrixes we used the function `sprand` that randomize the position of the numbers given a density and the dimension of the matrix. To generate the dense function, which has the zero in place of points, we used the function: `Matrix` which takes the sparse matrix and densifies it. We saw that the `"\"` operator is faster with the full one:

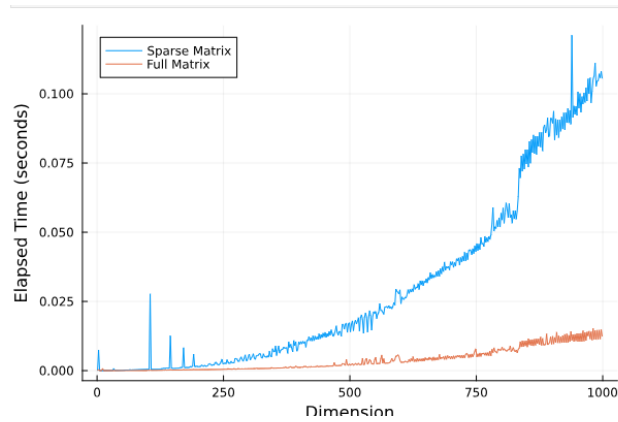
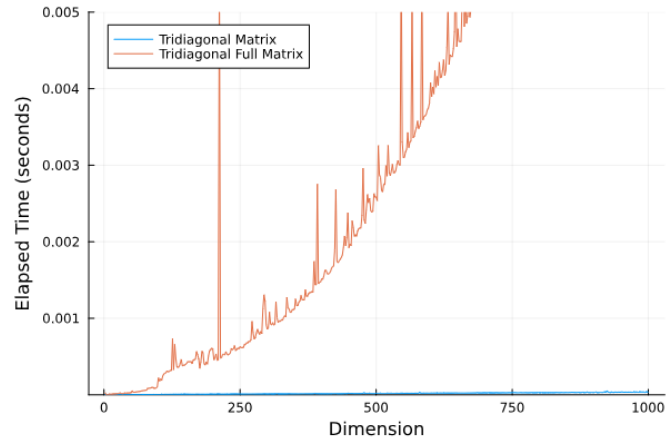
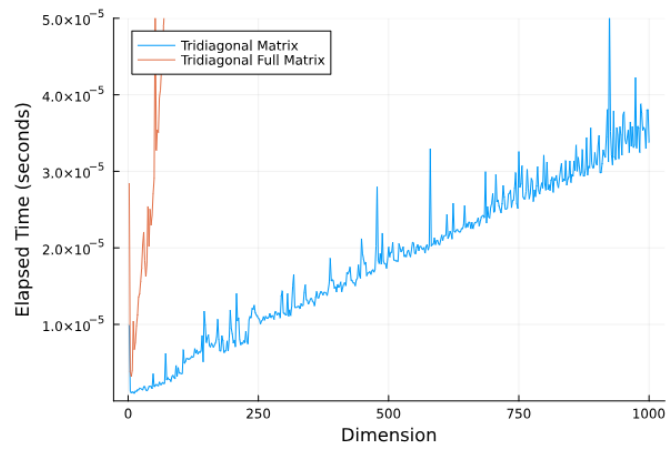


Figure 1: This is a graphic of elapsed time of solving the problem in base of the dimension of the matrix

3. Considering the second point conclusion, we have generated a tridiagonal matrix in Julia and compared how the operator worked on tridiagonal matrixes and tridiagonal full matrixes. We came to the conclusion that is way better to store the matrix as a tridiagonal because storing zeros leads to overhead:



(a) Zoomed out



(b) Zoomed in

Figure 2: Two different perspectives of the elapsed time for the tridiagonal matrixes