1 Solving Large Linear Systems of Equations

The goal of this problem is to solve five systems of equations given in the form $A\mathbf{x} = \mathbf{b}$ using Gauss elimination method and another method from NumPy. We need to time both methods to see which one is faster.

Before solving the systems, we need to inspect the basic properties of matrices A_i $(i = 1, ..., 5)$ to establish whether or not this system can be solved. I used NumPy function *shape* to verify if all matrices are quadratic and NumPy Linalg function *det* to find out if all determinants are non-zero. The first three matrices fulfilled both of these conditions, however the determinants of the last two matrices turn out to be $-inf$. I'm not sure why this is the case, perhaps it is because the matrices are too large for my laptop to handle. So, in the rest of this report I will talk about my results for the first three systems of equations.

I used a Gauss elimination code written by Jarno Elonen[1]. From NumPy solvers I chose to use *linalg.solve*. To estimate which method is faster I used Python function time.time() right before and after calling each solver, and then subtracted those two times to get the duration of each operation. The results are given in the table below.

matrix	shape	determinant	t_{GE} (s)	t_{numpy} (s)
1	10×10	0.010742	0.001403	0.000109
2	100×100	-3.0838×10^{33}	0.954593	0.000712
3	200×200	-1.51714×10^{98}	8.212505	0.004330

Comparing the times for the two methods, I conclude that the NumPy solver is much faster than the method based on Gauss elimination.

[1] I got it from *http://elonen.iki.fi/code/misc-notes/python-gaussj*.