Numerical Methods, project A, Number 31

Krzysztof Rudnicki
Student number: 307585
Advisor: dr Adam Krzemieniowski

October 27, 2021

Contents

Chapter 1

Problem 1 - Finding machine epsilion

1.1 Problem

Write a program finding macheps in the MATLAB environment

1.2 Theoretical Introduction

1.2.1 Definition of machine epsilion

Machine epsilion is the maximal possible relative error of the floating-point representation. (Tatjewski, p.14) Machine epsilion is equal to 2^{-t} where t is number of bits in the mantissa. In our case when we use IEEE Standard 754, mantissa is 53 bits long with first bit omitted as it is always equal to '1', so we technicaly work with 52 bits mantissa which makes the machine epsilion equal to: $2^{-52} = 2.220446\text{e}{-16}$

1.2.2 Practical applications of machine epsilion

Since macheps is connected to IEEE754 standard it is always equal to the same number, which means that we can safely compare results from different machines without worrying about their individual errors.

Macheps is also essential when we calculate cumulation of errors of given mathematical operation.

1.3 Solution

1.3.1 Matlab code

```
macheps = 1;
while 1.0 + macheps / 2 > 1.0
    macheps = macheps/2;
end
```

Code above shifts macheps one bit to the right each iteration (by dividing by 2), it ends when we run out of mantissa bits which renders us unable to save smaller number. Due to underflow the value of macheps becomes 0 and therefore 1.0 > (macheps / 2) > 1.0 will become false.

1.4 Discussion of the result

```
format long
disp(Display calculated macheps:)
disp(macheps);
disp(Display actual eps:)
disp(eps);
disp(Display 2^-52)
disp(2^—52)
disp(Display difference between calculated macheps and actual eps:)
disp(macheps — eps)
disp(Display difference between 2^-52 and actual eps:)
disp(2^—52 — eps) \
disp(Display difference between calculated macheps and 2^-52:)
disp(macheps — 2^—52)
```

Display calculated macheps:

$$2.220446049250313\text{e}{-16}$$

Display actual eps:

$$2.220446049250313\text{e}{-16}$$

Display 2^{-52}:

$$2.220446049250313\text{e}{-16}$$

Display difference between calculated macheps and actual eps:

$$0$$

Display difference between 2^{-52} and actual eps:

$$0$$

Display difference between calculated macheps and 2^{-52}:

$$0$$

As expected they are all equal to eachother. It means that our method of calculating macheps was correct.

Chapter 2

Problem 2 - Solving a system of n linear equations - indicated method

2.1 Problem

Write a program solving a system of n linear equations Ax = b using the indicated method (Gaussian elimination with partial pivoting).

2.2 Theoretical Introduction

Gaussian elimination with partial pivoting consists of 3 main steps:

2.2.1 Transform system of equation into an upper-triangular matrix

We start with the system of linear equations looking like this:

$$
\begin{aligned}
a_{11}^{(1)}x_1 + a_{12}^{(1)}x_2 + \cdots + a_{1n}^{(1)}x_n &= b_1^{(1)}, \\
a_{21}^{(1)}x_1 + a_{22}^{(1)}x_2 + \cdots + a_{2n}^{(1)}x_n &= b_2^{(1)}, \\
&\vdots \\
a_{n1}^{(1)}x_1 + a_{n2}^{(1)}x_2 + \cdots + a_{nn}^{(1)}x_n &= b_n^{(1)},
\end{aligned}
$$

2.2.2 Backward substitution

2.2.3 Partial Pivoting

2.3 Solution

2.4 Discussion of the result

Chapter 3

Problem 3 - Solving a system of n linear equations - iterative algorithm

3.1 Problem

3.2 Theoretical introduction

3.3 Solution

3.4 Discussion of the result

Chapter 4

Problem 4 - QR method of finding eigenvalues

4.1 Problem

4.2 Theoretical introduction

4.3 Solution

4.4 Discussion of the result

Bibliography

[1] Piotr Tatjewski (2014) *Numerical Methods*, Oficyna Wydawnicza Politechniki Warszawskiej