

DEPARTMENT OF PHYSICS
INDIAN INSTITUTE OF TECHNOLOGY, MADRAS

PH5720 Num. Methods
Time: 2:00 pm - 5:00 pm

Session 08

3 April 2018
[Total: 10 points]

Goal of this session:

1. Write a program for solving linear equations.
2. Compare the LU decomposition algorithm and the tridiagonal solver.
3. Please upload your plots (.pdf file) on moodle and submit this lab sheet by Monday 9 April 2018, 5:00 pm.

1 Problems:

1. Let us solve the Poisson's Equation in 1D using linear algebra techniques. The 1D equation we will solve is:

$$\frac{d^2\phi}{dx^2} = -\frac{\rho(x)}{\epsilon_0}$$

As a first step we will cast this equation in the following form:

$$-u''(x) = f(x)$$

where we assume that all variables are dimensionless. Let $f(x) = (3x + x^2)e^x$. This can be solved analytically to give:

$$u(x) = x(1 - x)e^x$$

We will solve the Poisson Equation in terms of simultaneous linear equation.

- (a) Using the numerical second derivative and the following definition for $h = 1/(N+1)$ and choosing $x \in [0, 1]$ with the boundary conditions $u(0) = u(1) = 0$, cast the Poisson's Equation (dimensionless) in the form of a matrix equation $AX = B$ and show that

$$A = \begin{pmatrix} 2 & -1 & 0 & 0 & 0 \cdots \\ -1 & 2 & -1 & 0 & 0 \cdots \\ 0 & -1 & 2 & -1 & 0 \cdots \\ \vdots & & & & \end{pmatrix} \quad B = \begin{pmatrix} h^2 f(x_1) \\ h^2 f(x_2) \\ \vdots \end{pmatrix}$$

- (b) Obtain $u(x)$ numerically using the gsl library functions `gsl_linalg_LU_decomp` and `gsl_linalg_LU_solve`. Compare the relative error between the analytical and the numerical solutions as a function of x for different N values by plotting the relative error as for $N = 10, 100, 1000$ as a function of x .
- (c) Using the C clock function `clock()` determine the time taken for solving the linear equation as a function of N . You will need to include the header file `time.h`. The clock function can be used as follows:

```
clock_t start, stop; /*Declaring variables start and stop of type clock_t,
which is defined in time.h*/

start = clock();

    - Body of the code: for example calls to the linear algebra solver -

stop = clock();

printf("time in seconds: %e \n", (double)(stop - start)/(CLOCKS_PER_SEC));
/*CLOCKS_PER_SEC is the number of ticks in a second and the clock() function
measures time in units of ticks.*/
```

Plot the time taken in seconds for the Linear Equation solver that uses LU decomposition followed by solving the factorized system as a function of N and find the slope.

- (d) Instead of the LU decomposition, use the tridiagonal solver in the GSL library and time your algorithm. Plot the time taken in seconds for the tridiagonal solver as a function of N and find the slope.

- (e) Compare the LU decomposition and the solver with the tridiagonal solver.