Project 1

Olav K. Arnesen
(Dated: September 10, 2022)

List a link to your github repository here!

PROBLEM 1

To check whether $u(x) = 1 - (1 - e^{-10})x - e^{-100x}$ is a solution or not we insert it into the Poisson equation:

$$\begin{aligned} f(x) =& -\frac{d^2u(x)}{dx^2} \\ =& -\frac{d^2}{dx^2}\left[1 - \left(1 - e^{-10}\right)x - e^{-10x}\right] \\ =& -\frac{d}{dx}\left[1 - e^{-10} + 10e^{-10x}\right] \\ =& 100e^{-10x} \end{aligned} \tag{1}$$

which is the expected solution.

PROBLEM 2

Using $n = 1000$ steps between $x = 0$ and $x = 1$ we got Fig. 1.

PROBLEM 3

The second derivative of $u(x_i) = u_i$, where $i = 1, 2 \ldots n$, is given numerically by

$$u_i'' = \frac{u_{i+1} - 2u_i + u_{i-1}}{h^2} + O(h^2) \tag{2}$$

where $h = \frac{x_n - x_1}{n}$ and $O(h^2)$ are all terms of order h^2 or higher. Inserting the Poisson equation and removing higher order terms we get the equation

$$-v_{i+1} + 2v_i - v_{i-1} = h^2 f_i \tag{3}$$

where $v_i \approx u_i$ and $f(x_i) = f_i$. We can rename $h^2 f_i$ to g_i to get a discretized Poission equation

$$-v_{i+1} + 2v_i - v_{i-1} = g_i \tag{4}$$

PROBLEM 4

We can write out the discretized Poisson equation for all i:

$$\begin{aligned}
i = 1 :&\; 2v_1 - v_2 \;\; +0 \;\;\; +0 \;\; \cdots + 0 \quad\;\; +0 \;\;\; = g_1 \\
i = 2 :&\; -v_1 + 2v_2 - v_3 \;\; +0 \;\; \cdots + 0 \quad\;\; +0 \;\;\; = g_2 \\
i = 3 :&\;\; 0 \;\; -v_2 \;\; +2v_3 - v_3 \cdots + 0 \quad\;\; +0 \;\;\; = g_3 \\
&\qquad\qquad \vdots \\
i = n :&\;\; 0 \;\; +0 \;\;\; +0 \;\;\; +0 \;\; \cdots - v_{n-1} + 2v_n = g_n
\end{aligned}$$

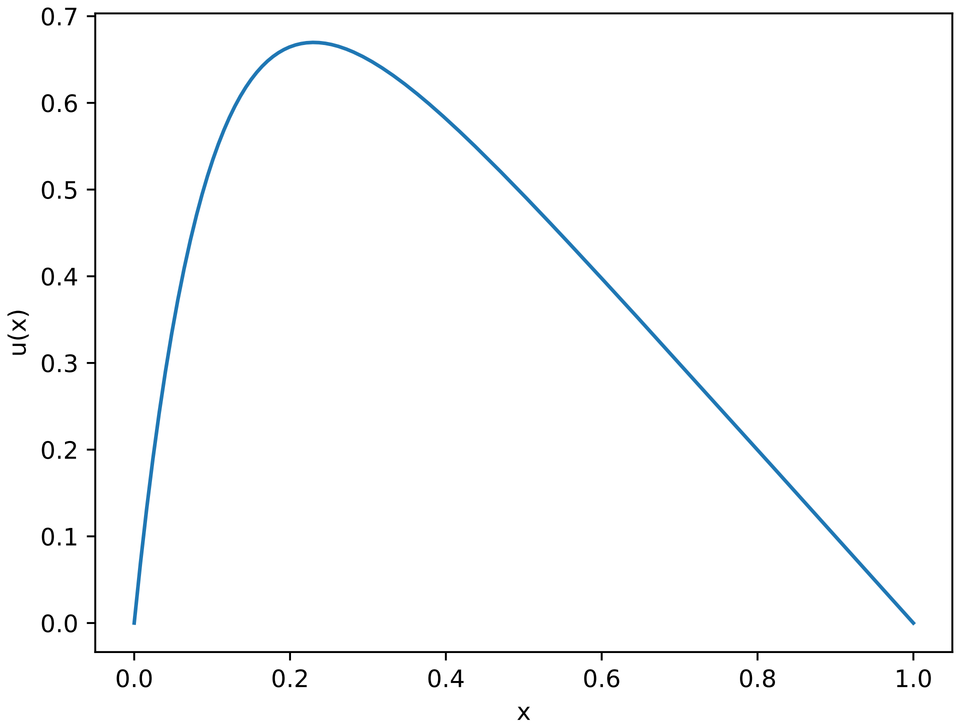

FIG. 1. The function $u(x)$ from $x = 0$ to $x = 1$.

If we define two vectors $\vec{v} = (v_0, v_1, \dots v_n)$ and $\vec{g} = (g_0, g_1, \dots g_n)$ we can clearly see these equations can be represented by

$$\begin{pmatrix} 2 & -1 & 0 & 0 & \dots & 0 \\ -1 & 2 & -1 & 0 & \dots & 0 \\ 0 & -1 & 2 & -1 & \dots & 0 \\ & & \vdots & & & \\ 0 & 0 & 0 & 0 & \dots & 2 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \\ v_3 \\ \vdots \\ v_n \end{pmatrix} = \begin{pmatrix} g_1 \\ g_2 \\ g_3 \\ \vdots \\ g_n \end{pmatrix} \tag{5}$$

where the left-hand matrix is a tri-diagonal $n \times n$-matrix, where all elements in the main diagonal is 2 and the elements in the sub- and superdiagonal are -1. If we call the matrix $\mathbf{A}$ we can write the vector equation as

$$\mathbf{A}\vec{v} = \vec{g} \tag{6}$$

PROBLEM 5

a)

$\vec{v}^*$ being a complete solution means it includes the endpoints $x = 0$ and $x = 1$, while $\vec{v}$ does not (though that was not obvious to me without help from a teacher). This means $m = n + 2$.

b)

When we solve for $\vec{v}$ we find the middle part of $\vec{v}^*$, i.e. not the endpoints. We could write $\vec{v}^* = (0, v_1, v_2, \dots, v_n, 0)$.

PROBLEM 6

a)

Algorithm derived in lecture on 2.09.2022.

Algorithm 1 Gaussian elimination

Subdiagonal: $\vec{a} = (a_2, a_3, \ldots, a_n)$
Main diagonal: $\vec{b} = (b_1, b_2, \ldots, b_n)$
Superdiagonal: $\vec{c} = (c_1, c_2, \ldots, c_{n-1})$
$\tilde{g_1} = g_1$ ▷ Forward substitution
$\tilde{b_1} = b_1$
for $i = 2, \ldots, n$ **do**
  $\tilde{g_i} = g_i - \frac{a_i}{b_{i-1}} g_{i-1}$
  $\tilde{b_i} = b_i - \frac{a_i}{b_{i-1}} c_{i-1}$
$v_n = \frac{\tilde{g_n}}{b_n}$ ▷ Back substitution
for $i = n-1, \ldots, 1$ **do**
  $v_i = \frac{\tilde{g_i} - c_i v_{i+1}}{\tilde{b_i}}$

b)

The number of FLOPs from forward substitution is $(n-1) \cdot 2 \cdot 3$, (3 in $\tilde{g_i}$, 3 in $\tilde{b_i}$, $n-1$ times). From back substitution there are $(n-1) \cdot 3 + 1$, (3 in v_i, $(n-1)$ times, plus v_n). The total number of FLOPs is $(n-1) \cdot 2 \cdot 3 + (n-1) \cdot 3 + 1 = 7n - 6 \approx 7n$ for large n.

PROBLEM 7

a)

Using the algorithm for $n = 1000$ non-endpoints and plotting the results gives us Fig. 2.

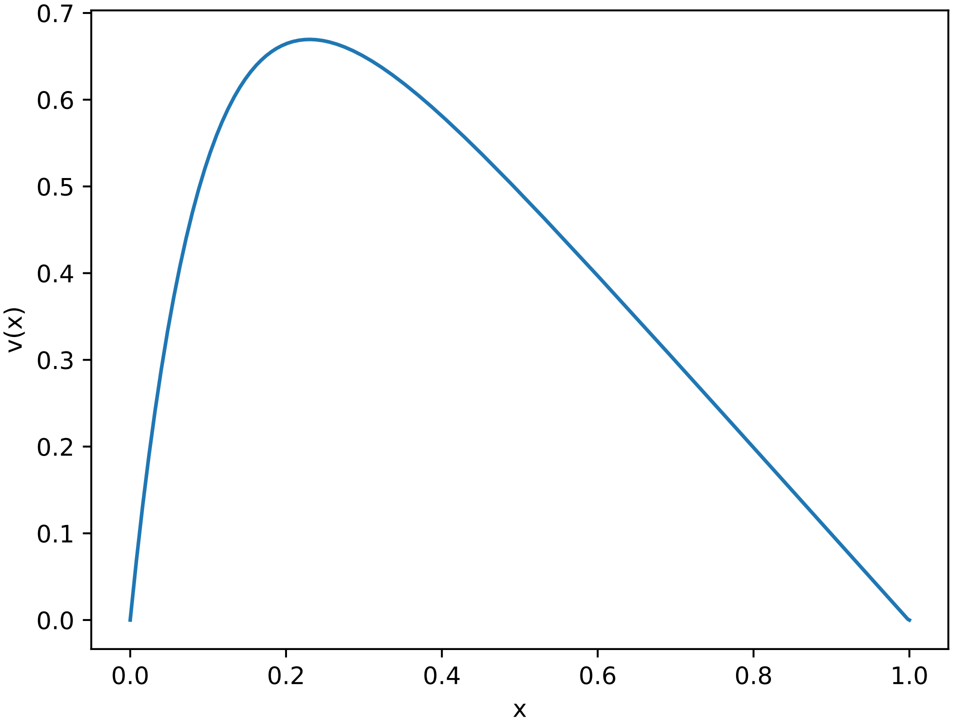

FIG. 2. The function $v(x)$ from $x = 0$ to $x = 1$.