Università della Svizzera italiana
Institute of Computing CI

Numerical Computing **2023**

Student: Harkeerat Singh Sawhney

Due date: Wednesday, 6 December 2023, 11:59 PM

Numerical Computing 2023 — Submission Instructions

(Please, notice that following instructions are mandatory: submissions that don't comply with, won't be considered)

- Assignments must be submitted to iCorsi (i.e. in electronic format).
- Provide both executable package and sources (e.g. C/C++ files, MATLAB). If you are using libraries, please add them in the file. Sources must be organized in directories called:
  Project_number_lastname_firstname
  and the file must be called:
  project_number_lastname_firstname.zip
  project_number_lastname_firstname.pdf
- The TAs will grade your project by reviewing your project write-up, and looking at the implementation you attempted, and benchmarking your code's performance.
- You are allowed to discuss all questions with anyone you like; however: (i) your submission must list anyone you discussed problems with and (ii) you must write up your submission independently.

1. General Questions [10 points]

1.1. Size of Matrix A

From the Background Information given for this Project we do know that $A \in \mathbb{R}^{n^2 \times n^2}$ indicates the transformation matrix coming from the repeated application of what is referred to as the "image kernel", which in our case tends to produce the blurring effect. In the other hand B is the transformed blurred image and X is the original square, grayscale image matrix, in which each matrix entry corresponds to one pixel value. Hence the blurring computation can be defined be following equation:

$$Ax = b \tag{1}$$

In the above equation x and b are the vectorized repersentation of X and B respectively.

Listing 1: Computing the size of A

```
%% Load Default Img Data
load('blur_data/B.mat');
B=double(B);
n = size(B,1);
sizeA = n * n;
disp(['Size of A: ', num2str(sizeA)]);
```

From the Code Listing 1 we can see that the size of A is 62500×62500. This is computed through the size of B which is 250×250 and then multiplying it by itself.

1.2. How many diagonal bands does A have?

It is understood that A is a d^2-banded symmetric matrix, where $d << n$. Since we know that the size of the kernel image matrix is 7×7 then we can also compute the amount of diagonal bands that A has. Hence the amount of diagonal bands that A has is 49.

1.3. What is the length of the vectorized blurred image b

In order to compute the length of the vectorized blurred image b, we need to compute the size of B and then multiply it by itself. We know that B is a 250×250 matrix, hence the length of the vectorized blurred image b is 62500.

2. Properties of A [10 points]

2.1. If A is not symmetric, how would this affect $\tilde{A}$?

A is used to compute the Conjugate Gradient method, which is an iterative method to solve the linear system $Ax = b$. If A is symmetric of full rank but not positive-definite we can bypass this issue by solving the augmented System.

$$A^T A x = A^T b \tag{2}$$
$$\tilde{A} x = \tilde{b} \tag{3}$$

In the above equation the pre-multiplication with A^T ensures that the resulting matrix $\tilde{A}$ is symmetric and positive-definite. Hence even if A is not symmetric, we can still compute $\tilde{A}$ because of the properties of A^T.

2.2. Explain why solving $Ax = b$ for x is equivalent to minimizing $\frac{1}{2}x^T Ax - b^T x$ over x, assuming that A is symmetric positive–definite.

We want to show that by minimizing $\frac{1}{2}x^T Ax - b^T x$ over x is equivalent to solving $Ax = b$. We can do this by taking the derivative of $\frac{1}{2}x^T Ax - b^T x$ with respect to x and setting it to zero. Hence we get the following equation:

$$\begin{aligned}
\frac{d}{dx}\left(\frac{1}{2}x^T Ax - b^T x\right) &= 0 && (4)\\
\frac{1}{2}\left(x^T A + x^T A^T\right) - b^T &= 0 && (5)\\
x^T A - b^T &= 0 && (6)\\
x^T A &= b^T && (7)\\
x^T &= b^T A^{-1} && (8)\\
x &= \left(b^T A^{-1}\right)^T && (9)\\
x &= \left(A^{-1}\right)^T b && (10)\\
x &= A^{-1} b && (11)\\
Ax &= b && (12)
\end{aligned}$$

Hence can see at the end that we get the equation $Ax = b$ which is what we wanted to show.

3. Conjugate Gradient [30 points]

3.1. Write a function for the conjugate gradient solver [x,rvec]=myCG(A,b,x0,max itr,tol), where x and rvecare, respectively, the solution value and a vector containing the residual at every iteration.

In this question we are asked to write a function for the conjugate gradient solver. The function is called `myCG` and it takes in the following parameters: `A`, `b`, `x0`, `max_itr` and `tol`. The function returns the solution value `x` and a vector containing the residual at every iteration `rvec`. The function is implemented in the Code Listing **??**.

4. Deblurring problem [35 points]