CS 510 - extra problems - not to be turned in

I. Consider the linear algebraic system

$$\begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 2 \\ 1 \end{bmatrix}.$$

(i) Write down the Jacobi, Gauss-Seidel, and SOR iterations in component form and in matrix form. If $x^{(0)} = \left(\begin{smallmatrix} 0 \\ 0 \end{smallmatrix}\right)$, what is $x^{(1)}$ for each of these iterations? (Your SOR result will contain ω as a parameter.)
(ii) How many Jacobi iterations are required to reduce the initial error by a factor 10^{-6}?

II. A generic way of deriving an iterative method for $Ax = b$ is to introduce a "splitting" $A = M + N$, and define $x^{(k+1)}$ as follows:

$$Mx^{(k+1)} = -Nx^{(k)} + b.$$

For this iteration to make sense, M should approximate A (perhaps crudely), and it should be easier to solve a linear system with M as coefficient matrix than with A.
Let L, D, U denote the lower triangular, diagonal, and upper triangular parts of A. The Jacobi iteration corresponds to the splitting $M = D$, $N = L + U$. Give analogous "splitting" interpretations for the Gauss-Seidel and SOR iterations.

III. Consider the following iteration for $Ax = b$:

$$x^{(k+1)} = x^{(k)} - \alpha \left(Ax^{(k)} - b \right), \quad \alpha \text{ a scalar.}$$

(i) What is the iteration matrix G for this iteration?
(ii) Assuming A is strictly diagonally dominant and that its rows have been scaled so that $a_{ii} = 1$ for all i, determine the value of α for which $\|G\|_\infty$ is minimized.

IV. Gaussian elimination applied to the tridiagonal system

$$\begin{bmatrix} d_1 & u_1 & & \\ l_2 & d_2 & u_2 & \\ & & \ddots & \\ & & l_n & d_n \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix} \tag{1}$$

is a highly sequential process, not well-suited for parallel computers. In this setting, a more attractive alternative is the following *odd-even* reduction procedure, which we describe assuming $n = 2^p - 1$. Note that if, for all even i, we eliminate x_{i-1} and x_{i+1} from equation

Typeset by $\mathcal{AMS}$-TEX

i using equations $i-1$ and $i+1$, respectively, the result is a tridiagonal system of size $2^{p-1}-1$ for the even x_i's:

$$
(2) \qquad \begin{bmatrix} d_2' & u_2' & & \\ l_4' & d_4' & u_4' & \\ & & \ddots & \\ & & l_{n-1}' & d_{n-1}' \end{bmatrix} \begin{bmatrix} x_2 \\ x_4 \\ \vdots \\ x_{n-1} \end{bmatrix} = \begin{bmatrix} b_2' \\ b_4' \\ \vdots \\ b_{n-1}' \end{bmatrix}
$$

Moreover, once the solution of the reduced system (2) is known, the odd-indexed equations in (1) can be used to 'backsolve' for the odd x_i's. This procedure, consisting of reduction and backsolving, can be applied recursively until a 1 by 1 system results.

Solve the following tridiagonal system via this procedure:

$$
\begin{bmatrix} 2 & -1 & & & & & \\ -1 & 2 & -1 & & & & \\ & -1 & 2 & -1 & & & \\ & & -1 & 2 & -1 & & \\ & & & -1 & 2 & -1 & \\ & & & & -1 & 2 & -1 \\ & & & & & -1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}.
$$

For a system of size $n = 2^p - 1$, what is the operation count (number of multiplies and divides) for this algorithm? Ideally, how would its parallel implementation time vary with n? Note: Odd-even reduction can be interpreted as Gaussian elimination without pivoting as applied to a suitably reordered system. For example, in the case of the above 7×7 problem, the reordered system is:

$$
\begin{bmatrix} 2 & & & & -1 & & \\ & 2 & & & -1 & & -1 \\ & & 2 & & & -1 & -1 \\ & & & 1 & & -1 & \\ -1 & -1 & & & 2 & & \\ & & -1 & -1 & & 2 & \\ & -1 & -1 & & & & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_3 \\ x_5 \\ x_7 \\ x_2 \\ x_6 \\ x_4 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}.
$$

V. Consider the following Laplace-type equation problem:

$$
-\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2}\right) + u = 0
$$

in the unit cube

$$
0 < x < 1,\ 0 < y < 1,\ 0 < z < 1,
$$

with u given on the boundary of the domain. To approximate the solution $u(x, y, z)$ numerically, we take $h = \frac{1}{n+1}$, lay down a uniform mesh

$$(x_i, y_j, z_k) = (ih, jh, kh), \qquad 0 \le i \le n+1,\ 0 \le j \le n+1,\ 0 \le k \le n+1,$$

and replace the 2^{nd} derivatives by the standard 3-point approximation. The resulting numerical solution $u_{i,j,k} \approx u(x_i, y_j, z_k)$ satisfies

$$(1) \qquad -u_{i,j,k-1} - u_{i,j-1,k} - u_{i-1,j,k} + (6 + h^2)u_{i,j,k} - u_{i+1,j,k} - u_{i,j+1,k} - u_{i,j,k+1} = 0$$

$$1 \le i \le n,\ 1 \le j \le n,\ 1 \le k \le n.$$

(i) Suppose the system (1) is solved by Gaussian elimination with the solution vector x ordered as follows:

$$u(i, j, k) \leftrightarrow x(i + n * j + n^2 * k).$$

(This is the obvious extension to 3-D of the row ordering we considered in class for the 2-D Laplace equation system.)

Which form of Gaussian elimination would you apply to the system (1)? What is the resulting operation count? Express your answer in the form $O(n^p)$ for appropriate p.

(ii) Next consider the Jabobi iteration as applied to the system (1), starting with initial iterate $u^{(0)} = 0$. For what m can you guarantee that the corresponding iterate $u^{(m)}$ will have relative error $< \epsilon$ in the ∞-norm (ϵ a fixed number)? What is the overall operation count to solve the system (1) in this way? Give your answers in the form $O(n^p)$ for appropriate p.

Notes:

- Gaussian elimination as applied to an $n \times n$ matrix with bandwidth w takes $O(nw^2)$ operations.
- You might find it useful to take a logarithm at some point.
- $\ln(1 + x) \cong x \quad$ for $|x| << 1$.