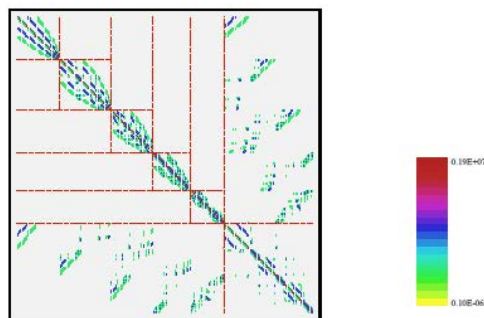
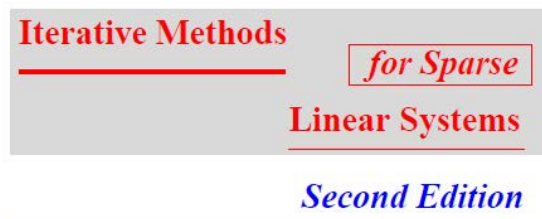


# Master Vietnam-France in HCMC

## High Performance computing

### TP 01 : Iterative Solving



Yousef Saad

Copyright ©2003 by the Society for Industrial and Applied Mathematics

FIGURE 1 – A good book

([http ://www-users.cs.umn.edu/~saad/IterMethBook\\_2ndEd.pdf](http://www-users.cs.umn.edu/~saad/IterMethBook_2ndEd.pdf))

The aim of this session is to recall and compare several iterative solvers

## Provided

A matlab conjugate gradient code and a main program prog.m to fill in.

## To do : Solve $Ax = b$ iteratively

1. Write a Jacobi solver with the arguments mentioned in prog.m
2. Write a Gauss-Seidel solver with the arguments mentioned in prog.m.
3. Write a Richardson solver with the arguments mentioned in prog.m. Check numerically that the optimal parameter in the Richardson's scheme is  $\rho = \lambda_1 + \lambda_N$  where  $\lambda_1$  (resp.  $\lambda_N$ ) is the minimal (resp. maximum) eigenvalue of A (Use Matlab "eig" function to compute eigenvalues.)
4. Write a SOR solver. Compute numerically an optimal parameter  $\omega$ . How does it compare with the one given in the lecture.
5. Write a Steepest Descent iterative solver
6. Plot convergence curves for all these methods and compare them with the provided conjugate gradient.
7. In prog.m , change the value of N to N=60. What happens ? Why ?
8. Replace the backslash operator in FE\_Poisson\_reduit.m by all these iterative algorithms and plot convergence curves.
9. Comment costs in time and required memory for all these methods.

## Send us your scripts

Create a directory with your names XX-YY and "-Iter" :  
for example mkdir XX-YY-Iter  
In this directory, you will put all the developed scripts .  
Zip this directory and send it by mail to  
halpern@math.univ-paris13.fr  
ryan@math.univ-paris13.fr