Computational Linear Algebra (MATH70024) - Mastery Component

Amaury Francou — CID : 01258326
MSc Applied Mathematics — Imperial College London
amaury.francou16@imperial.ac.uk

January 14, 2022

Abstract

This report has been written as part of the mastery component assessment, required for the completion of the "Computational Linear Algebra" (MATH70024) module, taught in the Department of Mathematics at Imperial College London. It comes along with several python codes (`.py` files), findable in my personal Github Classroom repository (https://github.com/Imperial-MATH96023/clacourse-2021-amauryfra/), in the `mastery` directory. The relevant commit's hash is `f86b10f5604144439b276df7a231c69517987617`. The report goes through 3 sections.

We here discuss the iterative methods devised by [Bai et al., 2003] for solving large sparse non-Hermitian positive definite system of linear equations. The authors provide a method based on a Hermitian and skew-Hermitian splitting of the system-related coefficient matrix.

1 [Bai et al., 2003] HSS and IHSS methods : problem description and main results

We consider the following linear system of equations :

$$Ax = b, \text{ with } A \in \mathbb{C}^{n\times n} \text{ a non-singular matrix, and } x, b \in \mathbb{C}^n \tag{1.1}$$

The authors consider a coefficient matrix A that is non necessarily Hermitian and that is positive definite, i.e. for all non-zero vector $z \in \mathbb{C}^n\backslash\{0\}$, A verifies $z^*Az > 0$. As some iterative solving methods suitable for problem 1.1 use diagonal/off-diagonal or upper/lower triangular splitting techniques for said matrix A, the authors consider here a Hermitian/skew-Hermitian splitting approach. Namely, A is broken as follows :

$$A = H + S \tag{1.2}$$

$$\text{where } H = \frac{1}{2}(A + A^*) \text{ is the Hermitian part } (H^* = H)$$

$$\text{and where } S = \frac{1}{2}(A - A^*) \text{ is the skew-Hermitian part } (S^* = -S) \tag{1.3}$$

1.1 HS splitting method

The first method proposed by [Bai et al., 2003] is called the HSS method (exact HS splitting). The method starts with providing an initial guess $x^{(0)}$ and a coefficient $\alpha > 0$. The solution vector x of problem 1.1 is computed through two-step iterations - until convergence - as follows :

$$\begin{cases} (\alpha I + H)x^{k+\frac{1}{2}} = (\alpha I - S)x^{(k)} + b \\ (\alpha I + S)x^{k+1} = (\alpha I - H)x^{k+\frac{1}{2}} + b \end{cases} \tag{1.4}$$

This method implies obtaining the inverses $(\alpha I + H)^{-1}$ and $(\alpha I + S)^{-1}$. The authors main result states that said method 1.4 converges unconditionally to the unique solution of problem 1.1. We provide the following pseudo-code implementation of the method :

$\langle$ *Input* ϵ $\rangle$ ▷ Provide a convergence tolerance
$H \leftarrow \frac{1}{2}(A + A^*)$
$S \leftarrow \frac{1}{2}(A - A^*)$
$M_1 \leftarrow \alpha I + H$
$M_2 \leftarrow \alpha I + S$
$N_1 \leftarrow \alpha I - S$
$N_2 \leftarrow \alpha I - H$
$\langle$ *Compute* M_1^{-1} $\rangle$
$\langle$ *Compute* M_2^{-1} $\rangle$
$b_1 \leftarrow M_1^{-1}b$ ▷ Compute those matrix-vector products only once
$b_2 \leftarrow M_2^{-1}b$
$C_1 \leftarrow M_1^{-1}N_1$ ▷ Compute those matrix products only once
$C_2 \leftarrow M_2^{-1}N_2$
$x \leftarrow x_0$ ▷ Given initial guess
while $\|b - Ax\| > \epsilon$ **do**
  $x \leftarrow C_1x + b_1$
  $x \leftarrow C_2x + b_2$

They prove this claim using their *Lemma 2.1* result stating that for any two splittings of matrix A - into say $A = M_1 - N_1$ and $A = M_2 - N_2$ - and any two-step iteration process using said splitting, if the iteration matrix defined as $M_2^{-1}N_2M_1^{-1}N_1$ has a spectral radius (i.e. the largest absolute value of its eigenvalues) that is less than 1 ($\rho(M_2^{-1}N_2M_1^{-1}N_1) < 1$), then the iteration process converges to the unique solution of the system 1.1.

In the previously defined HSS method, the two splittings of A are $A = (\alpha I+S)-(\alpha I-H)$ and $A = (\alpha I + H) - (\alpha I - S)$. The iteration matrix - that depends on the coefficient α - is here given by $M(\alpha) = (\alpha I + S)^{-1}(\alpha I - H)(\alpha I + H)^{-1}(\alpha I - S)$. In their *Theorem 2.2* the authors show that the spectral radius of the relevant iteration matrix is bounded by the quantity $\sigma(\alpha) = \max_{\lambda_i \in \Lambda(H)} \left|\frac{\alpha-\lambda_i}{\alpha+\lambda_i}\right|$, where $\Lambda(H)$ is the set of eigenvalues of the Hermitian part H of A. As we directly have that for all $\alpha > 0$ and for all eigenvalues λ_i, $\left|\frac{\alpha-\lambda_i}{\alpha+\lambda_i}\right| < 1$, it follows that the spectral radius of the iteration matrix is unconditionally smaller than 1. Namely, we have $\forall \alpha > 0 \ \rho(M(\alpha)) \leq \sigma(\alpha) < 1$. By *Lemma 2.1*, it follows that the HSS iteration process 1.4 converges to the unique solution of the system 1.1 unconditionally.

Within their *Theorem 2.2*, the authors show that the speed of convergence of the HSS process 1.4 is bounded by $\sigma(\alpha)$, as the spectral radius $\rho(M(\alpha))$ of the iteration matrix governs said speed of convergence. They highlight the fact that the $\sigma(\alpha)$ quantity depends only on the coefficient α and the spectrum of the Hermitian part H of A, in contrast with the methods presented in the previous literature.

In *Corollary 2.3*, the authors compute a minimizer of $\sigma(\alpha)$, given by $\alpha^* = \sqrt{\gamma_{\min}\gamma_{\max}}$, where $\gamma_{\min}$ and $\gamma_{\max}$ are the minimum and the maximum eigenvalues of the matrix H. However, it is pointed out that such α^* minimizes only an upper bound and not the spectral radius itself.

1.2 Inexact HS splitting method

As the (exact) HSS method described above requires inverting two $n \times n$ matrix, and as this may prove to be too costly in real industrial implementations, the authors provide a modified inexact HSS - labeled IHSS - method. The IHSS method uses nested iterative methods to solve the inner iteration step systems $(\alpha I + H)x^{k+\frac{1}{2}} = (\alpha I - S)x^{(k)} + b$ and $(\alpha I + S)x^{k+1} = (\alpha I - H)x^{k+\frac{1}{2}}$ more cost efficiently. Namely, in the IHSS setting the solution of those subproblem is simply approximated by any given iterative solving method.

The authors consider using the conjugate gradient method to solve the subproblem involving the coefficient matrix $(\alpha I + H)$, as it is Hermitian positive definite, and other Krylov subspace methods for the second subproblem involving $(\alpha I + S)$. Note that the previously studied GMRES method can also be used here. Solving 1.1 using the IHSS method translates in the following pseudo-code :

$\langle$ *Input some* $\check{\epsilon}$, ϵ_k *and* η_k $\rangle$ ▷ Provide convergence tolerances
$H \leftarrow \frac{1}{2}(A + A^*)$
$S \leftarrow \frac{1}{2}(A - A^*)$
$M_1 \leftarrow \alpha I + H$
$M_2 \leftarrow \alpha I + S$
$x \leftarrow x_0$ ▷ Given initial guess
$z \leftarrow x_0$
while $\|b - Ax\| > \check{\epsilon}$ **do**
    $r \leftarrow b - Ax$
    $p \leftarrow r - M_1 z$
    while $\|p\| > \epsilon_k \|r\|$ **do**
        $\langle$ *Update* z *by running one more iteration of given* $M_1 z = r$ *solving method* $\rangle$
        $p \leftarrow r - M_1 z$
    $x \leftarrow x + z$
    $r \leftarrow b - Ax$
    $q \leftarrow r - M_2 z$
    while $\|q\| > \eta_k \|r\|$ **do**
        $\langle$ *Update* z *by running one more iteration of given* $M_2 z = r$ *solving method* $\rangle$
        $q \leftarrow r - M_2 z$
    $x \leftarrow x + z$

Note that the authors consider step-dependent tolerance values ϵ_k and η_k, where k represents the current iteration step of the loop.

Analogously to the HSS setting, the authors investigate the convergence conditions of the IHSS method from a theoretical point of view. Considering general two splittings of matrix A in $A = M_1 - N_1$ and $A = M_2 - N_2$. In *Theorem 3.1*, [Bai et al., 2003] obtain that the IHSS method converges to the exact solution of problem 1.1 if $\bar{\sigma} + \bar{\mu}\bar{\theta}\epsilon_{\max} + \bar{\theta}(\bar{\rho} + \bar{\theta}\bar{\nu}\epsilon_{\max})\eta_{\max} < 1$, where $\bar{\sigma} = \|N_2 M_1^{-1} N_1 M_2^{-1}\|$, $\bar{\rho} = \|M_2 M_1^{-1} N_1 M_2^{-1}\|$, $\bar{\mu} = \|N_2 M_1^{-1}\|$, $\bar{\theta} = \|A M_2^{-1}\|$, $\bar{\nu} = \|M_2 M_1^{-1}\|$, $\epsilon_{\max}$ and $\eta_{\max}$ being the maximum values of the ϵ_k and η_k.

Applied to the specific HS splitting, the use of *Theorem 3.1* result gives rise to the convergence condition stated in *Theorem 3.2* : the IHSS method converges to the exact solution of problem 1.1 if $(\sigma(\alpha) + \theta\rho\eta_{\max})(1 + \theta\epsilon_{\max}) < 1$, where $\rho = \|(\alpha I + S)(\alpha I + H)^{-1}\|$ and $\theta = \|A(\alpha I + S)^{-1}\|$.

The authors highlight the fact that the tolerances ϵ_k and η_k do not necessarily need to be chosen positively decreasing to zero as k increases to ensure convergence of the IHSS method. However, given in *Theorem 3.3* and *Theorem 3.4*, choosing appropriate sequences such that $\epsilon_k \to 0$ and $\eta_k \to 0$, allows the IHSS method to retrieve the HSS convergence speed asymptotically.

We recall that the coefficient matrix size is $n \times n$. The authors provide a calculation of the operations count required for computing one step of the IHSS iteration. Namely, by noting a the number of operations required to compute the matrix-vector product Ax, $\chi_k(H)$ and $\chi_k(S)$ the number of operations required to compute the two IHSS subproblems, the IHSS method requires $\mathscr{O}(4n + 2a + \chi_k(H) + \chi_k(S))$ operations.

2 Implementing and testing HSS and IHSS methods

We provide a set of functions for investigating the methods proposed by [Bai et al., 2003].

We consider an application of problem 1.1 in an aeronautical engineering case. Namely, we examine an airfoil profile from the NACA database. It is in the form of an adjacency matrix giving the cross-section shape of an aircraft wing (see figure 1). The American National Advisory Committee for Aeronautics (NACA) has developed an extended collection of such adjacency matrix (see `https://www.nasa.gov/image-feature/langley/100/naca-airfoils` for more information). After linearization, the cross-section related matrix A_0 can be used to calculate structural and aerodynamical properties in the form of problem 1.1 ([Sforza, 2014]).

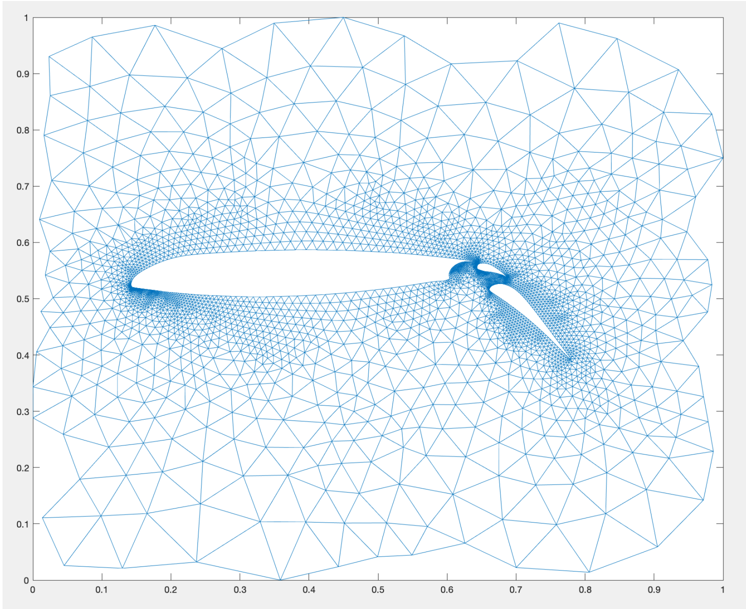

Figure 1: Cross-section plot of the considered adjacency matrix A_0 using `MATLAB gplot`.

The considered airfoil includes two trailing flaps. The matrix data is built-in `MATLAB` and can be accessed using the process available at https://fr.mathworks.com/help/matlab/math/graphical-representation-of-sparse-matrices.html. We transfer the data to our python code using the `scipy.io.loadmat` method. We obtain an 4253×4253 sparse non-symmetric matrix `A0` in the form of an `numpy array`. We implement a function to verify that a provided matrix is positive definite, by verifying all its eigenvalues are strictly positive. By using this function, we check that our given A_0 is well positive definite. We investigate the sparsity structure of A_0 :

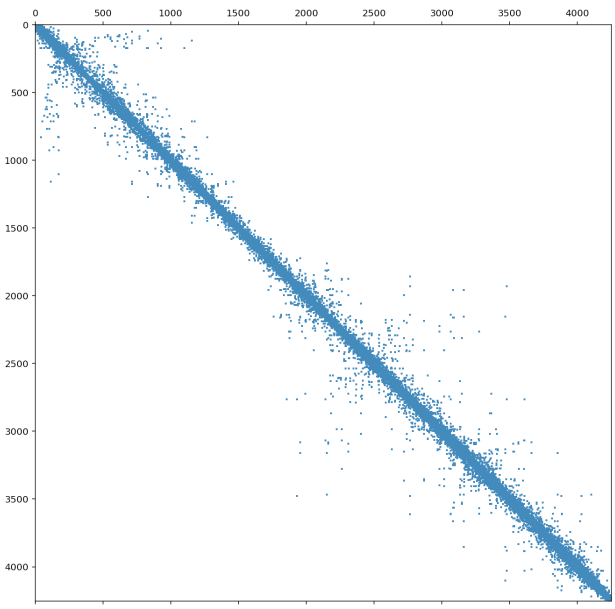

Figure 2: A_0 matrix sparsity structure.

We implement two functions `HSS` and `IHSS` that performs the corresponding methods. The functions take a coefficient matrix `A`, a right-hand side vector `b`, the parameter `alpha`, an initial guess `x0` (defaulted to `b` if not provided), a stopping condition tolerance `tol`, and a maximum number of iterations `maxit` as inputs, and outputs the solution vector `x` and the number of iterations needed `nits` (set to -1 if the algorithm didn't converge to `tol` in `maxit` iterations).

We solve the two inner subsystems of the IHSS method using our pre-implemented `cla_utils.exercise10.GMRES` algorithm. We implement a `theorem32` function that verifies if the given matrix A, and the given parameters α and $\epsilon_{\max} = \eta_{\max}$ are sufficient for the IHSS method to converge, by using the result of *Theorem 3.2*.

We test our functions on the airfoil matrix A_0, as well as on random definite positive matrices. We compute such matrices by performing a change of basis on a random diagonal matrix, where all entries are strictly positive. For the basis change, we use a unitary matrix obtained by computing the QR factorization of a supplementary random matrix.

We provide some computation time assessments for the A_0 matrix, using the `timeit` library. We observe that the IHSS method is relatively faster than the HSS one for solving problem 1.1, with matrix A_0 and a random vector b. Namely, the HSS method needs 7.42 seconds to finish against 5.55 for the IHSS method. We further provide some automatic testing for both methods, using the `pytest` library.

You may find said python implementations in the `mastery.py` file and you may also run the automatic testing of the functions by using `pytest test/test_mastery.py` while in `mastery` directory.

3 Analyzing related literature

We perform a short literature review on the further generalizations that have been provided to [Bai et al., 2003] original methods.

We first consider [Benzi, 2009] setting. The author highlights that the inner sub-system of the IHSS method involving $S + \alpha I$ can be often ill-conditioned and may further present substantial difficulties to solve. To address this issue, he examines a generalized HSS method - labeled GHSS - in which the Hermitian part is itself split as $H = G + K$, having K of *simple form* (for instance diagonal). The coefficient matrix is then broken as follows : $A = H + S = G + (K + S)$. The corresponding two splittings of this newly given method are $A = (G + \alpha I) - (\alpha I - S - K)$ and $A = (S + K + \alpha I) - (\alpha I - G)$. The author shows that this new process is also unconditionally convergent to the solution of 1.1. A main motivation of this new splitting is that the iteration-involved matrix $S + K + \alpha I$ is here *diagonally dominant*, making it better conditioned than the analogous matrix of the IHSS method.

We secondly consider [Cheng et al., 2015] analysis. The authors consider an image denoising and restoration algorithm. The process consists of estimating, and further retrieving through operations, an *ideal* flawless image. Their considered process is mathematically equivalent to solving an augmented linear system, in which the coefficient matrix is of the form :

$\begin{pmatrix} I & A \\ -A^T & \mu^2 I \end{pmatrix}$ where A is a problem-related sub-matrix and μ is a coefficient.

The authors consider solving said augmented system through an iterative HS splitting technique. In their specific setting the matrices H and S are as follows :

$$H = \begin{pmatrix} I & 0 \\ 0 & \mu^2 I \end{pmatrix} \text{ and } S = \begin{pmatrix} 0 & A \\ -A^T & 0 \end{pmatrix}.$$

Considering H is here diagonal and S has a special structure, the authors provide a special HS splitting method - labeled SHSS - that is :

$$\begin{cases} (\alpha I + H)x^{k+\frac{1}{2}} = (\alpha I - S)x^{(k)} + b \\ (I + S)x^{k+1} = (I - H)x^{k+\frac{1}{2}} + b \end{cases}$$

The difference with the original HSS method is that α has been set to 1 in the second iteration step. The authors leverage the fact that H is here diagonal, and moreover that smaller spectral radius induce smaller computation times for the iterative methods. The main motivation is that the matrix $I - H$ has m zeros on the diagonal (where $m \times m$ is the sub-matrix size), which brings that the iteration matrix has at minimum m null eigenvalues. Setting the parameter α only in the first iteration step allows to *control* the remaining m eigenvalues more efficiently.

Finally, we consider [Li and Ma, 2018]'s method. The authors examines here the linear complex problem $Ax = (W + iT)x = b$. They start by multiplying each side of the equation by Euler's formula $e^{i\theta} = \cos(\theta) + i\sin(\theta)$, where $\theta \in [0, 2\pi[$ is a parameter. The considered problem becomes $(\cos(\theta)W + \sin(\theta)T + i(\cos(\theta)T + \sin(\theta)W))x = e^{-i\theta}b$. From this, they devise the one-step Euler-extrapolated HS method - labeled E-HS - that is as follows :

$$(\cos(\theta)W + \sin(\theta)T)x^{(k+1)} = i(\sin(\theta)W - \cos(\theta)T)x^{(k)} + e^{-i\theta}b.$$

The authors provide the conditions parameter θ must verify in order for the method to be convergent. Moreover, they provide a formulation of a such optimal parameter, in terms of speed of convergence.

Some relevant extra references are also provided in the following bibliography.

References

Bahramizadeh, Z., Nazari, M., Zak, M. K. and Yarahmadi, Z. [2020], 'Minimal residual hermitian and skew-hermitian splitting iteration method for the continuous sylvester equation', arXiv preprint arXiv:2012.00310 .

Bai, Z.-Z., Golub, G. H. and Ng, M. K. [2003], 'Hermitian and skew-hermitian splitting methods for non-hermitian positive definite linear systems', SIAM Journal on Matrix Analysis and Applications **24**(3), 603–626.

Bai, Z.-Z., Golub, G. H. and Ng, M. K. [2008], 'On inexact hermitian and skew-hermitian splitting methods for non-hermitian positive definite linear systems', Linear Algebra and Its Applications **428**(2-3), 413–440.

Bai, Z.-Z., Golub, G. H. and Pan, J.-Y. [2004], 'Preconditioned hermitian and skew-hermitian splitting methods for non-hermitian positive semidefinite linear systems', Numerische Mathematik **98**(1), 1–32.

Bai, Z.-Z., Golub, G. and Li, C.-K. [2007], 'Convergence properties of preconditioned hermitian and skew-hermitian splitting methods for non-hermitian positive semidefinite matrices', Mathematics of Computation **76**(257), 287–298.

Benzi, M. [2009], 'A generalization of the hermitian and skew-hermitian splitting iteration', SIAM Journal on Matrix Analysis and Applications **31**(2), 360–374.

Benzi, M., Gander, M. J. and Golub, G. H. [2003], 'Optimization of the hermitian and skew-hermitian splitting iteration for saddle-point problems', BIT Numerical Mathematics **43**(5), 881–900.

Cheng, G.-H., Rao, X. and Lv, X.-G. [2015], 'The comparisons of two special hermitian and skew-hermitian splitting methods for image restoration', Applied Mathematical Modelling **39**(3-4), 1275–1280.

Greif, C. and He, Y. [2021], 'Hss (0): an improved hermitian/skew-hermitian splitting iteration', arXiv preprint arXiv:2109.13327 .

Li, C. and Ma, C. [2018], 'On euler-extrapolated hermitian/skew-hermitian splitting method for complex symmetric linear systems', Applied Mathematics Letters **86**, 42–48.

Li, T., Wang, Q.-W. and Zhang, X.-F. [2021], 'Hermitian and skew-hermitian splitting methods for solving a tensor equation', International Journal of Computer Mathematics **98**(6), 1274–1290.

Pour, H. N. and Goughery, H. S. [2015], 'New hermitian and skew-hermitian splitting methods for non-hermitian positive-definite linear systems', Numerical Algorithms **69**(1), 207–225.

Sforza, P. [2014], Chapter 5 - wing design, in P. Sforza, ed., 'Commercial Airplane Design Principles', Butterworth-Heinemann, Boston, pp. 119–212.
URL: *https://www.sciencedirect.com/science/article/pii/B978012419953800005X*

Simoncini, V. and Benzi, M. [2004], 'Spectral properties of the hermitian and skew-hermitian splitting preconditioner for saddle point problems', SIAM Journal on Matrix Analysis and Applications **26**(2), 377–389.