Solving PDE's with FEniCS

Linear solvers

Chapter 25

Introduction to

Automated Modeling

with FEniCS

by L. Ridgway Scott

Linear Solvers

In large computations, the rate limiting step is often the solution of a linear system.

In some cases, it is sufficient to use Gaussian elimination or its variants (such as Cholesky factorization).

Such algorithms are called **direct methods**.

However, for large, three-dimensional simulations, **iterative methods** are frequently more efficient.

Direct methods

Stationary iterative methods

There are three important classes of iterative methods.

The first of these are known equivalently as **stationary iterative methods** and **relaxation methods**.

Examples include Jacobi, Gauss-Seidel, SOR, SSOR, etc.

algorithm	sufficient conditions on A for convergence
Jacobi	generalized diagonally dominant
Gauss-Seidel	symmetric, positive definite
SOR	symmetric, positive definite

Table 1: Stationary iterative methods for solving $AX = F$ and conditions on A that guarantee convergence.

Current applications of stationary iteration

These basic techniques are still used in certain contexts, and many of the concepts behind them are frequently used in more complicated solvers.

In particular, relaxation methods are frequently used as **smoothers** for multi-grid methods.

Typically, the simpler the iterative method, the easier it is to implement a parallel version.

Stationary iteration theory

Suppose that we are solving a linear system $AX = F$. The general form of a stationary iterative scheme is

$$Nx^{n+1} = PX^n + F,$$

where $A = N - P$ and N is chosen to be an easily invertible matrix, e.g., diagonal (Jacobi) or triangular (Gauss-Seidel, SOR).

The error $E^n = X - X^n$ satisfies

$$x^{n+1} = MX^n,$$

where $M = N^{-1}P$.

Stationary iterative methods

Thus convergence is equivalent to $\rho(M) < 1$ where ρ is the spectral radius.

It is known [22] that

- Jacobi is convergent for generalized diagonally dominant matrices, and
- Gauss-Seidel and SOR are convergent for symmetric, positive definite matrices.

Krylov methods are nonstationary iterative methods.

They naturally adapt to the properties of the solution.

algorithm	matrices for which the method applies
CG	symmetric, positive definite
MINRES	symmetric
GMRES	invertible

Table 2: Krylov subspace based methods for solving $AX = F$ and conditions on A that guarantee convergence.

Krylov subspaces

Krylov[1] methods are a class of techniques based on projecting the solution onto an increasing subspace of vectors that is efficient to create.

Suppose that we are solving a linear system $AX = F$.

Then the Krylov subspace of order k is the linear space spanned by

$$F, AF, \ldots, A^k F.$$

Such vectors are easily created iteratively via $A^i F = A(A^{i-1}F)$, where $A^0 F = F$.

[1] Alexei Nikolaevich Krylov (1863–1945) was very active in the theory and practice of shipbuilding and is commemorated by the Krylov Shipbuilding Research Institute.

Conjugate gradients

The first of the Krylov methods is called **conjugate gradients** (a.k.a. **CG**) and was developed by Hestenes[2] and Stiefel[3].

Conjugate gradients converges for symmetric positive definite matrices, and it has an optimality property [22] that makes it extremely attractive.

[2] Magnus Rudolph Hestenes (1906–1991) obtained a Ph.D. at the University of Chicago with Gilbert Bliss in 1932.

[3] Eduard L. Stiefel (1909–1978) is known both as a pure mathematician (for his work on the Stiefel-Whitney characteristic classes) and as a computational mathematician (he was also an early user and developer of computers [23]). Stiefel was the advisor of Peter Henrici as well as 63 other students over a period of 37 years. Henrici was the advisor of Gilbert Strang, one of the early pioneers of the mathematical theory of the finite element method.

Minimum residual

The algorithm **MINRES** is applicable to symmetric but indefinite matrices [19].

Although CG and MINRES utilize the same Krylov space of vectors, they minimize different quantities.

CG minimizes $\|X - X^k\|_A$, where $\|y\|_A = \sqrt{y^t A y}$, whereas MINRES minimizes $\|F - AX^k\|_{\ell^2}$, where $\|y\|_{\ell^2} = \sqrt{y^t y}$.

It is easy to see that CG requires A to be positive definite, since $\|\cdot\|_A$ is not a norm otherwise.

GMRES

For symmetric, positive definite matrices, MINRES can outperform CG in some cases [12].

The algorithm Generalized Minimum RESidual **GMRES** [20, 14, 10] can be used for general matrices.

The **Arnoldi** algorithm [8, 15] is closely related to GMRES.

Multi-grid

Multi-grid methods apply to problems posed on grids having having coarse grids and fine grids.

Variational, this occurs if we have subspaces $V^i \subset V$ with $V^i \subset V^{i+1}$.

The solutions to the variational problems

$$\text{Find } u^i \in V^i \text{ such that } a(u^i, v) = F(v) \quad \forall v \in V^i \tag{1}$$

are then increasingly accurate approximations to the solution u of the variational problem

$$\text{Find } u \in V \text{ such that } a(u, v) = F(v) \quad \forall v \in V. \tag{2}$$

But more importantly, u^i can be used as an initial guess for an iterative scheme for solving for u^{i+1}.

However, real power of multi-grid is deeper than this.

Suppose that we are trying to solve for $u^i \in V^i$ satisfying (1) and we have an approximate solution $w^i \in V^i$.

Then the residual error $r^i = u^i - w^i$ satisfies

$$a(r^i, v) = a(u^i, v) - a(w^i, v) = F(v) - a(w^i, v) \quad \forall v \in V, \tag{3}$$

which does not involve knowing u^i.

MG magic

But more importantly, u^i can be used as an initial guess for an iterative

The magic of multi-grid is to approximate (3) on a coarser space (V^{i-1}).

Need w^i to be smooth enough that this is an effective strategy, but a variety of iterative methods, both stationary iterative methods and Krylov methods, are smoothers that remove high frequencies.

For details, see [7].

MG variants

It is not strictly required to have $V^i \subset V^{i+1}$.

There are two ways in which $V^i \not\subset V^{i+1}$ occurs.

One is when discontinuous finite element spaces are used [6].

Another is when the underlying grids are not nested [17, 24, 9, 13].

Non-nested grids allow substantial freedom and facilitate mesh coarsening [9].

MG variants

Multi-grid methods were initiated by Bakhvalov[4] [1] in 1966.

Achi Brandt [5] began popularizing and developing the method in 1977.

Bank and Dupont [2, 3] gave one of the first proofs of convergence of the method, in research that initiated at the University of Chicago.

[4] Nikolai Sergeevich Bakhvalov (1934—2005) studied with both Sobolev and Kolmogorov.

Preconditioners

The convergence rate of many iterative methods depends on the condition number of the linear system.

For a symmetric, positive definite linear system, we can take the condition number to be defined as the ratio of the largest eigenvalue divided by the smallest eigenvalue.

We have also seen that round-off is strongly affected by the condition number of a linear system.

Linear systems associated with partial differential operators often have condition numbers that grow inversely with the mesh resolution.

PDE condition numbers

PDE condition numbers grow as mesh size decreases because PDEs have eigenvalues of unbounded size.

For finer meshes, larger eigenvalues can be resolved.

In particular, eigenfunctions often oscillate with a frequency roughly proportional to the eigenvalue.

Thus the finer meshes resolve higher frequencies.

Therefore iterative methods introduce a limit on our ability to resolve solutions based on mesh refinement.

Effect on round-off

We have seen that round-off is strongly affected by the condition number of a linear system.

One approach to this dilemma is to use higher-order approximations.

— But this is limited: higher-order approximation resolves higher frequency eigenfunctions.

— So the condition number is not reduced.

A better approach is to scale the linear system appropriately to control the size of the eigenvalues.

Types of preconditioners

The simplest scaling is diagonal preconditioning.

For a given matrix A, define diag(A) to be the diagonal matrix with the same diagonal entries as A.

Then the diagonal preconditioning of A is the matrix diag$(A)^{-1}A$.

Fortunately, it is simple to compute the inverse of a diagonal matrix $P = \text{diag}(A)^{-1}$.

More complex version: produce preconditioner by using incomplete factorization of the system [16, 4]

Limiting sparsity

Incomplete factorization:

— sparsity condition is enforced autocratically.

— Gaussian elimination (or other direct method) is performed in a way that only creates certain nonzero elements in the resulting factors.

If for example we restrict the factors to be only diagonal, then the algorithm is equivalent to diagonal preconditioning.

On the other hand, we make no restriction on sparsity, and allow arbitrary fill-in, then the algorithm produces the exact inverse (via forward and backward solution algorithms).

Limiting sparsity

One common choice is to limit the sparsity of the factors to be the same as the sparsity pattern of the original matrix.

Such a factorization is obtained by following an elimination algorithms (e.g., Gaussian elimination, Cholesky factorization, etc.), but when the algorithm calls for fill-in to occur, these additions to the original sparse structure of A are ignored.

This yields a matrix P with the same sparsity pattern as A and yet P is in some sense an approximation to A^{-1}.

Benefit of preconditioning

The benefit of preconditioning is that the iterative method performs as if the matrix condition number is that of the preconditioned system PA [11].

Thus significant benefit can occur.

The main objective is to choose P to be as close to A^{-1} as possible.

The discrete Green's function provides a way to solve a system [21], and this can be used to create an efficient, parallel algorithm to implement a preconditioner [21].

Preconditioner theory

A general understanding of preconditioners for linear systems arising in solving PDEs is given in [18].

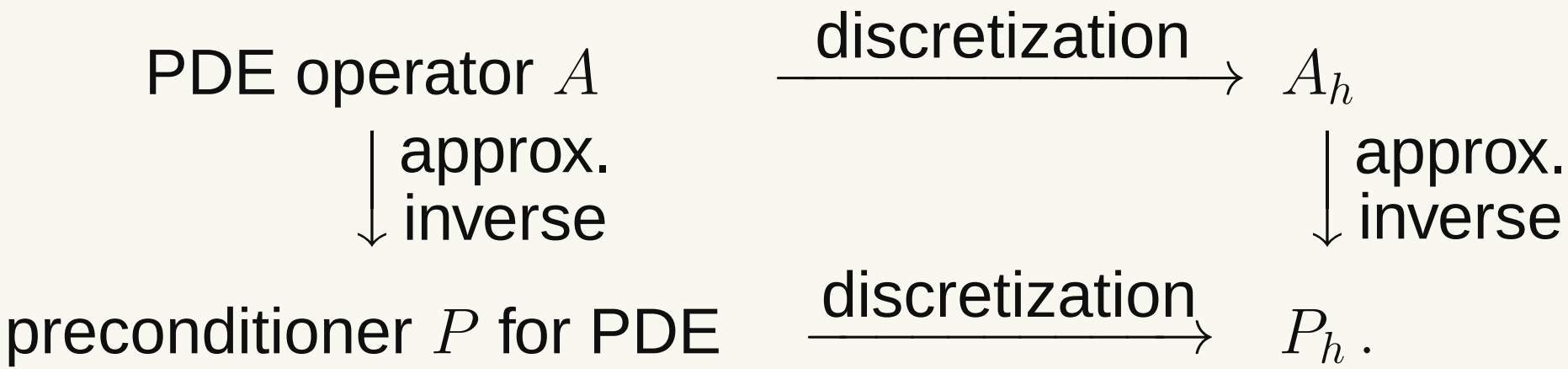
They describe how to extend the concept of preconditioner to the PDE itself.

This in turn, when discretized, provides an effective preconditioner that can be robust with respect to mesh size and parameters in the PDE.

Preconditioner theory

Two ways to construct preconditioners.

[18] advocates the bottom-left of the diagram.



References

[1] Nikolai Sergeevich Bakhvalov. On the convergence of a relaxation method with natural constraints on the elliptic operator. *USSR Computational Mathematics and Mathematical Physics*, 6(5):101–135, 1966.

[2] R. E. Bank and T. Dupont. Analysis of a two-level scheme for solving finite element equations. Research Report CNA-159, Center for Numerical Analysis, Univ. Texas–Austin, 1980.

[3] Randolph E. Bank and Todd Dupont. An optimal order process for solving finite element equations. *Mathematics of Computation*, 36(153):35–51, 1981.

[4] H. Berryman, J. Saltz, W. Gropp, and R. Mirchandaney. Krylov methods preconditioned with incompletely factored matrices on the CM-2. *Journal of Parallel and Distributed Computing*, 8:186–190, 1990.

[5] Achi Brandt. Multi-level adaptive solutions to boundary-value problems. *Mathematics of computation*, 31(138):333–390, 1977.

[6] Susanne C. Brenner. A nonconforming multigrid method for the stationary stokes equations. *Mathematics of Computation*, pages 411–437, 1990.

[7] Susanne C. Brenner and L. Ridgway Scott. *The Mathematical Theory of Finite Element Methods*. Springer-Verlag, third edition, 2008.

[8] Peter N. Brown. A theoretical comparison of the Arnoldi and GMRES algorithms. *SIAM Journal on Scientific and Statistical Computing*, 12(1):58–78, 1991.

[9] Peter R. Brune, Matthew G. Knepley, and L. Ridgway Scott. Unstructured geometric multigrid in two and three dimensions on complex and graded meshes. *SIAM J. Sci. Computing*, 35(1):A173–A191, 2013.

[10] Stephen L. Campbell, Ilse C.F. Ipsen, C. Tim Kelley, and Carl D. Meyer. Gmres and the minimal polynomial. *BIT Numerical Mathematics*, 36(4):664–675, 1996.

[11] S. C. Eisenstat. Efficient implementation of a class of preconditioned conjugate gradient methods. *SIAM J. Scientific and Stat. Comput.*, 2:1–4, 1981.

[12] David Chin-Lung Fong and Michael A. Saunders. CG versus MINRES: An empirical comparison. *SQU Journal for Science*