Preconditioners.jl: A Flexible and Extensible Framework for Preconditioning in Iterative Solvers

Mohamed Tarek[1,2], ChatGPT[3]

[1]Pumas-AI Inc., USA
[2]University of Sydney Business School, Australia
[3]OpenAI, USA

February 25, 2023

Iterative linear solvers are widely used to solve large-scale linear systems and eigenvalue problems arising from various scientific and engineering applications, such as computational fluid dynamics, structural mechanics, and electromagnetics. These solvers typically involve iterative methods that generate a sequence of approximate solutions to the original system, with the hope that the sequence converges to the exact solution.

One of the key challenges in iterative solvers is that they can converge slowly or even fail to converge altogether, especially for large and ill-conditioned systems. Preconditioning is a powerful technique for improving the convergence rate and robustness of iterative solvers by transforming the original system into an equivalent one that is easier to solve.

The idea behind preconditioning is to (lazily) multiply the original system by a suitable matrix, called a preconditioner, before applying an iterative solver. The preconditioner is chosen to reduce the condition number of the system, which measures the sensitivity of the solution to small changes in the input data. By reducing the condition number, the preconditioner can make the iterative solver converge faster and with fewer iterations, thus reducing the computational cost.

There are various types of preconditioners, ranging from simple diagonal and incomplete factorizations to more sophisticated algebraic multigrid and block preconditioners. The choice of preconditioner depends on the properties of the system and the characteristics of the iterative solver. In general, a good preconditioner should be easy to compute, cheap to apply, and effective in reducing the condition number of the system.

Preconditioners.jl provides a wide range of preconditioners that can be used with iterative solvers in Julia, typically by wrapping other packages which implement said preconditioners. By leveraging the power of Julia's high-level abstractions, Preconditioners.jl together with IterativeSolvers.jl offer a convenient and efficient framework for iteratively solving linear systems and eigenvalue problems with preconditioning.