Journal of Computational Mathematics
Vol.37, No.6, 2019, 889–915.

http://www.global-sci.org/jcm
doi:10.4208/jcm.1910-m2019-0034

BLOCK ALGORITHMS WITH AUGMENTED RAYLEIGH-RITZ PROJECTIONS FOR LARGE-SCALE EIGENPAIR COMPUTATION*

Haoyang Liu and Zaiwen Wen
Beijing International Center for Mathematical Research, Peking University, Beijing 100871, China
Email: liuhaoyang@pku.edu.cn, wenzw@pku.edu.cn
Chao Yang
Computational Research Division, Lawrence Berkeley National Laboratory, Berkeley, CA, USA
Email: cyang@lbl.gov
Yin Zhang
Department of Computational and Applied Mathematics, Rice University, Houston, USA
Email: yzhang@rice.edu

Abstract

Most iterative algorithms for eigenpair computation consist of two main steps: a subspace update (SU) step that generates bases for approximate eigenspaces, followed by a Rayleigh-Ritz (RR) projection step that extracts approximate eigenpairs. So far the predominant methodology for the SU step is based on Krylov subspaces that builds orthonormal bases piece by piece in a sequential manner. In this work, we investigate block methods in the SU step that allow a higher level of concurrency than what is reachable by Krylov subspace methods. To achieve a competitive speed, we propose an augmented Rayleigh-Ritz (ARR) procedure. Combining this ARR procedure with a set of polynomial accelerators, as well as utilizing a few other techniques such as continuation and deflation, we construct a block algorithm designed to reduce the number of RR steps and elevate concurrency in the SU steps. Extensive computational experiments are conducted in C on a representative set of test problems to evaluate the performance of two variants of our algorithm. Numerical results, obtained on a many-core computer without explicit code parallelization, show that when computing a relatively large number of eigenpairs, the performance of our algorithms is competitive with that of several state-of-the-art eigensolvers.

Mathematics subject classification: 65F15, 90C06.
Key words: Extreme eigenpairs, Augmented Rayleigh-Ritz projection.

1. Introduction

In this paper, we consider to compute $k \ll n$ eigenpairs corresponding to k largest or smallest eigenvalues of a given large-scale real symmetric matrix $A \in \mathbb{R}^{n\times n}$. Let $\lambda_1, \lambda_2, \cdots, \lambda_n$ be the eigenvalues of A sorted in a descending order: $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$, and $q_1, \ldots, q_n \in \mathbb{R}^n$ be the corresponding eigenvectors such that $Aq_i = \lambda_i q_i$, $\|q_i\|_2 = 1$, $i = 1, \ldots, n$ and $q_i^T q_j = 0$ for $i \neq j$. The eigenvalue decomposition of A is defined as $A = Q_n \Lambda_n Q_n^T$, where, for any integer $i \in [1, n]$,

$$Q_i = [q_1,\ q_2,\ \ldots, q_i] \in \mathbb{R}^{n\times i}, \qquad \Lambda_i = \mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_i) \in \mathbb{R}^{i\times i}, \tag{1.1}$$

* Received February 10, 2019 / Revised version received May 15, 2019 / Accepted October 9, 2019 / Published online November 18, 2019 /

where diag$(\cdot)$ denotes a diagonal matrix with its arguments on the diagonal. For simplicity, we also write $A = Q\Lambda Q^T$ where $Q = Q_n$ and $\Lambda = \Lambda_n$.

Most algorithms for computing a subset of eigenpairs of large matrices are iterative in which each iteration consists of two main steps: a subspace update (SU) step and a projection step. The subspace update step varies from method to method but with a common goal in finding a matrix $X \in \mathbb{R}^{n\times k}$ so that its column space is a good approximation to the k-dimensional eigenspace spanned by k desired eigenvectors. At present, the predominant methodology for subspace updating is still Krylov subspace methods, as represented by Lanczos type methods [12, 15] for real symmetric matrices. These methods generate an orthonormal matrix X one (or a few) column at a time in a sequential mode. Along the way, each column is multiplied by the matrix A and made orthogonal to all the previous columns. In contrast to Krylov subspace methods, block methods, as represented by the classic simultaneous subspace iteration method [22], carry out the multiplications of A to all columns of X at the same time in a batch mode. As such, block methods generally demand a lower level of communication intensity. Once X is obtained and orthonormalized, the projection step aims to extract from X a set of approximate eigenpairs (see more details in Section 2) that are optimal in a sense. For the projection step often the method of choice is the Rayleigh-Ritz (RR) procedure. More complete treatments of iterative algorithms for computing subsets of eigenpairs can be found, for example, in the books [1, 3, 19, 25, 30].

The purpose of this work is to construct and test a framework for block algorithms that can efficiently, reliably and accurately compute a relatively large number of exterior eigenpairs of large-scale matrices. The algorithm framework is constructed to take advantages of multi/many-core or parallel computers, although a study of parallel scalability itself will be left as a future topic. It appears widely accepted that a key property hindering the competitiveness of block methods is that their convergence can become intolerably slow when decay rates in relevant eigenvalues are excessively flat. A central task of our algorithm construction is to rectify this issue of slow convergence.

Our framework starts with an outer iteration loop that features an enhanced RR step called the augmented Rayleigh-Ritz (ARR) projection which can provably accelerate convergence under mild conditions. For the SU step, we consider two block iteration schemes whose computational cost is dominated by block SpMVs (the sparse matrix A multiplying a vector): (i) the classic power method applied to multiple vectors without periodic orthogonalization, and (ii) a recently proposed Gauss-Newton method. For further acceleration, we apply our block SU schemes to a set of polynomial accelerators, say $\rho(A)$, aiming to suppress the magnitudes of $\rho(\lambda_j)$ where λ_j's are the unwanted eigenvalue of A for $j > k$. In addition, a deflation scheme is utilized to enhance the algorithm's efficiency. Some of these techniques have been studied in the literature over the years (e.g. [24, 34] on polynomial filters), and are relatively well understood. In practice, however, it is still a nontrivial task to integrate all the aforementioned components into an efficient and robust eigensolver. For example, an effective use of a set of polynomial filters involves the choice of polynomial types and degrees, and the estimations of intervals in which eigenvalues are to be promoted or suppressed. There are quite a number of choices to be made and parameters to be chosen that can significantly impact algorithm performance.

Specifically, our main contributions are summarized as follows.

1. The augmented Rayleigh-Ritz (ARR) procedure proposed in [33] is carefully implemented and extensively evaluated in numerical experiments. Our ARR provably speeds up convergence without increasing the block size of the iterate matrix X in the SU step (thus