UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Scalable Library for Eigenvalue Problem Computations SLEPc

SLEPc Technical Report **STR-6**
Available at http://slepc.upv.es

A Survey of Software for Sparse Eigenvalue Problems

V. Hernández
J. E. Román
A. Tomás
V. Vidal

Last update: February, 2009 (SLEPc 3.0.0)
Previous updates: SLEPc 2.3.0, SLEPc 2.3.2, SLEPc 2.3.3

About SLEPc Technical Reports: These reports are part of the documentation of SLEPc, the *Scalable Library for Eigenvalue Problem Computations*. They are intended to complement the Users Guide by providing technical details that normal users typically do not need to know but may be of interest for more advanced users.

Contents

1 Summary

This document is a survey of freely available software tools for the numerical solution of large sparse eigenvalue problems. It includes a list of libraries, programs or subroutines, describing each of them very briefly. Only software that can be obtained freely via the Internet is considered, thus excluding proprietary software. Also, the scope of this report is limited to software aimed at sparse problems; software for dense matrices is not considered, even if the implemented methods are of iterative nature. On the other hand, no comparisons of the different methods/solvers are provided; for that, the reader is referred to [Lehoucq and Scott, 1996], [Bergamaschi and Putti, 2002], or [Arbenz *et al.*, 2005].

The aim of the survey is to provide SLEPc users with a broad view of the current eigensolver scenario, in order to help them assess how well SLEPc fits their needs. The survey is most useful as long as it is complete and up-to-date. Please communicate inaccuracies and additions of new software via the SLEPc contact email, slepc-maint@upv.es.

The software is classified in two groups: "current" and "legacy" software.

Current software is listed alphabetically in Table 1. This software is either actively maintained, or has less than ten years.

Legacy software is listed alphabetically in Table 2. Subroutines or libraries included in this list are quite old, with more than ten years since the last update, or are not available anymore. This software is rather old-styled in its design and generally forces the user to interface with it via a fixed storage format or through a user-supplied matrix-vector product routine with a fixed prototype. This software will probably be of little use for application programmers who want to solve real problems, but may be of interest for researchers on eigensolvers. The only exception is maybe ARPACK, which provides a user-friendly reverse communication interface and is still widely used.

Note that some of the packages can be used directly from SLEPc provided they are activated during installation, in particular PRIMME, ARPACK, BLZPACK, TRLAN, and BLOPEX.

Name	Description	Version	Date	Language	Par
ANASAZI	Block Krylov-Schur, block Davidson, LOBPCG	9.0	2008	C++	M
BLKLAN	Block Lanczos [P]	-	2003	C/Matlab	-
BLOPEX	LOBPCG	-	2004	C/Matlab	M
BLZPACK	Block Lanczos [P+S]	04/00	2000	F77	M
EIGIFP	Inverse-free Krylov subspace method	2.1.1	2004	Matlab	-
IETL	Power, RQI, Lanczos [N]	2.2	2006	C++	-
INSYLAN	Indefinite Symmetric Lanczos	1.0	2000	Matlab	-
IRBLEIGS	Block Lanczos (implicit restart)	1.0	2002	Matlab	-
JADAMILU	Jacobi-Davidson (symmetric)	2.0	2009	F77	-
JDCG	Jacobi-Davidson (symmetric)	-	2000	Matlab	-
MPB	Conjugate Gradient, Davidson	1.4.2	2003	C	M
PDACG	Deflation-accelerated Conjugate Gradient	-	2000	F77	M
PRIMME	Block Davidson, JDQMR, JDQR, LOBPCG	1.1	2006	C/F77	M
PROPACK	SVD via Lanczos [P]	2.1/1.1	2005	F77/Matlab	O
PySPARSE	Jacobi-Davidson (symmetric)	1.0.1	2007	Python	-
SLEPc	Krylov-Schur, Arnoldi, Lanczos, RQI, Subspace	3.0.0	2009	C/F77	M
SPAM	Davidson with SPAM	-	2001	F90	-
TRLAN	Lanczos (dynamic thick-restart)	-	2006	F90	M

Table 1: List of software for the solution of sparse eigenvalue problems.

Name	Description	Version	Date	Language	Par
ARNCHEB	Arnoldi-Chebyshev	-	1997	F77	-
ARPACK	Arnoldi/Lanczos (implicit restart)	2.1	1995	F77	M
ARPACK++	Arnoldi/Lanczos (implicit restart)	1.1	1998	C++	-
DVDSON	Davidson	1.0	1995	F77	-
LANCZOS	Lanczos [N]	-	1992	F77	-
LANZ	Lanczos [P]	1.0	1991	F77	-
LASO	Lanczos [S]	2	1983	F77	-
LOPSI	Subspace Iteration	1	1981	F77	-
JDQR/JDQZ	Jacobi-Davidson	-	1998	F77/Matlab	-
NA18	Block Davidson	-	1999	F77	-
NAPACK	Power, Lanczos [N]	-	1987	F77	-
QMRPACK	Nonsymmetric Lanczos (lookahead)	-	1996	F77	-
SRRIT	Subspace Iteration	1	1997	F77	-
SVDPACK	SVD via Lanczos [P], Ritzit & Trace Minimization	-	1992	C/F77	-
Underwood	Block Lanczos [F]	-	1975	F77	-

Table 2: List of legacy software.

In the tables, the name of each package contains a link to the Internet address where the software can be found. The description simply gives a summary of the method(s) provided by each library. In the case of Lanczos methods, the description indicates which type of reorthogonalization is implemented: full (F), selective (S), partial (P) or none (N). A slightly more informative description is provided in section 2. The version and date information is included to give an idea of whether a package is currently being actively developed or not. The language column indicates from which programming language(s) the software can be used (C, C++, Fortran77, Fortran90, Matlab, Java or Python). The last column marks those packages which are prepared for running in parallel, either with a message-passing paradigm via MPI (M), or with a shared-memory paradigm via OpenMP (O).

2 Software Listed by Method Class

This section is organized in several classes of methods, starting with the simplest algorithms (single-vector iterations) and increasing complexity towards methods that have attracted more interest recently. Note that software implementing state-of-the-art methods may be rather experimental, with little documentation and other drawbacks, whereas older methods tend to have more reliable and usable implementations.

Currently, SLEPc provides methods of the first four categories.

2.1 Single and Multiple Vector Iteration Methods

Single-vector iteration methods are available in the following packages:

- NAPACK provides a version of the power iteration that can handle complex conjugate eigenvalues. (NAPACK also includes a simple implementation of Lanczos, see 2.3.)
- IETL implements the power iteration, inverse iteration and Rayleigh quotient iteration. (IETL also includes a Lanczos solver, see 2.3.)

A version of subspace iteration for symmetric matrices is Rutishauser's Ritzit procedure, which was translated to Fortran in SVDPACK (see 2.4).

LOPSI [Stewart and Jennings, 1981] implements the subspace iteration method for non-symmetric matrices. The algorithm computes eigenvectors directly, so it is less robust than other implementations.

SRRIT [Bai and Stewart, 1997] is another implementation of subspace iteration for non-symmetric matrices, but based on the Schur decomposition. It combines subspace iteration with Rayleigh-Ritz projection and locking of converged eigenvectors. SLEPc provides a re-implementation of this method, see SLEPc Technical Report STR-3, "Subspace Iteration in SLEPc".

2.2 Arnoldi Methods

ARNCHEB [Braconnier, 1993] is a Fortran software that implements the Arnoldi method with explicit restart, combined with Chebyshev polynomial acceleration.

ARPACK [Lehoucq *et al.*, 1998] provides a Fortran implementation of the Implicitly Restarted Arnoldi method, for both real and complex arithmetic. It can be used for both standard and generalized problems and for both symmetric and non-symmetric problems. In the symmetric case, Lanczos with full reorthogonalization is used instead of Arnoldi. ARPACK is one of the most popular eigensolvers, due to its efficiency and robustness. ARPACK++ is a C++ interface to ARPACK.

Another implementation of Arnoldi is also available in IRBLEIGS (described in 2.3), which can solve standard or generalized eigenproblems.

A generalization of block Arnoldi with implicit restart is the block Krylov-Schur algorithm, which is available in the ANASAZI eigensolver package. ANASAZI is part of TRILINOS, a parallel object-oriented software framework for large-scale multi-physics scientific applications. Other eigensolvers available in ANASAZI are block Davidson (see 2.5) and LOBPCG (see 2.6).

2.3 Lanczos Methods

Implementation of Lanczos methods abound, most of them oriented to real generalized symmetric eigenvalue problems. Packages specific for the computation of the singular value decomposition (SVD) are described in section 2.4.

Basic reorthogonalization strategies The Lanczos method with full reorthogonalization can be found in the ARPACK package, which incorporates an implicit restart technique (see 2.2). An implementation of the block Lanczos method that also employs full reorthogonalization is available in the UNDERWOOD subroutines [Golub and Underwood, 1977].

The LANCZOS software implements the strategy proposed by Cullum and Willoughby [1985] in which no reorthogonalization is carried out and the resulting tridiagonal matrix is post-processed in order to eliminate spurious eigenvalues. This is also the strategy of the Lanczos solver found in IETL (described in 2.1). On the other hand, NAPACK (see 2.1) provides a straightforward Lanczos implementation with no reorthogonalization and no post-processing.

The LASO package implements a block Lanczos method with selective reorthogonalization [Parlett and Scott, 1979].

Partial reorthogonalization The partial reorthogonalization idea is present in a number of packages: LANZ, BLZPACK, PROPACK, and BLKLAN.

LANZ [Jones and Patrick, 1993] is a shared-memory parallel Lanczos with partial reorthogonalization that is also intended for real generalized problems. It incorporates subroutines for the spectral transformation and computation of inertia, and allows to specify a computational interval in which the solutions are to be sought.

BLZPACK [Marques, 1995] is an MPI-based parallel implementation of Lanczos, also for real generalized eigenproblems. The algorithm that it provides is a block method combining both partial and selective reorthogonalization. Computational intervals are also allowed.

PROPACK and BLKLAN are packages for the computation of the SVD that also employ partial reorthogonalization, see 2.4.

Restarting schemes Some packages provide a restarting mechanism in order to limit the required amount of memory. ARPACK incorporates implicit restart, as mentioned above.

IRBLEIGS [Baglama *et al.*, 2003] is a Matlab program that implements an implicitly restarted block Lanczos method, that allows the computation of extreme eigenvalues of symmetric matrices or symmetric positive-definite pencils. Also, interior eigenvalues can be found without requiring a factorization.

TRLAN is based on a different restarting scheme called thick restart [Wu and Simon, 2000]. TRLAN is a parallel software written in Fortran 90 that can be used to address standard real symmetric problems.

Non-SPD problems Except for the SVD packages described in next section, all the above Lanczos solvers are intended for symmetric positive-definite problems, i.e., real symmetric or complex Hermitian matrices and symmetric positive-definite matrix pairs. The following packages address other kind of problems.

INSYLAN is a prototype Matlab implementation of the symmetric indefinite Lanczos method for symmetric indefinite matrix pairs [Bai *et al.*, 2000].

QMRPACK [Freund and Nachtigal, 1996] contains subroutines that implement the two-sided Lanczos algorithm with look-ahead, that can be used to compute eigenvalue approximations of non-Hermitian matrices.

2.4 Singular Value Decomposition

The software described in this subsection is specific for the computation of the partial singular value decomposition (SVD). The methods implemented in these packages are usually iterative eigensolvers such as Lanczos, that have been modified in such a way that compute the singular triplets of matrix A via an equivalent symmetric eigenvalue problem defined by matrix A^TA (or AA^T) or the 2-cyclic matrix $\left[\begin{smallmatrix} \alpha I & A \\ A^T & \alpha I \end{smallmatrix}\right]$.

SVDPACK [Berry, 1992] provides four alternative solvers, all of them for the equivalent eigensystem with either matrix A^TA or the 2-cyclic matrix. The solvers are the following:

- Single-vector Lanczos, in particular, SVDPACK incorporates the LANSO library (see 2.3).
- Hybrid Block Lanczos with full reorthogonalization.
- Subspace Iteration (Rutishauser's Ritzit procedure).
- Trace minimization.

PROPACK [Larsen, 1998] is based on the Lanczos bidiagonalization algorithm with partial reorthogonalization. It works directly on matrix A without forming the equivalent symmetric eigensystem, leading to a more efficient algorithm. The Fortran version of PROPACK incorporates implicit restart, thus reducing the storage requirements. PROPACK can be used with either real or complex matrices.

BLKLAN [Liu *et al.*, 2004] is a version of block Lanczos specifically oriented to the computation of the Takagi factorization of complex symmetric matrices. It is implemented in C and Matlab. There is also a non-blocked version available only in Matlab.

Finally, a Matlab subroutine for the SVD is also available in IRBLEIGS (described in 2.3).

2.5 Davidson and Jacobi-Davidson Methods

The package DVDSON [Stathopoulos and Fischer, 1994] is a block implementation of the Davidson method with several extensions such as reorthogonalization. It is intended for real symmetric matrices.

The package NA18 [Sadkane and Sidje, 1999] is a Fortran-77 software package which implements a deflated and variable-block version of the Davidson method for computing a few of the extreme (i.e., leftmost or rightmost) eigenpairs of large sparse symmetric matrices.

A parallel block Davidson for symmetric problems is included in ANASAZI (described in 2.2).

Another implementation of Davidson's method can be found in MPB (described in 2.6).

JDQR is a Matlab implementation of the Jacobi-Davidson method for the computation of eigenpairs of non-symmetric matrices, as described in [Fokkema *et al.*, 1998]. The corresponding algorithm for matrix pairs is implemented in another package called JDQZ, which is also available in Fortran with complex arithmetic.

JDCG is a modified version of JDQR for symmetric problems, based on [Notay, 2002].

PYSPARSE is a Python toolkit that provides a module JDSYM that implements the Jacobi-Davidson method for symmetric generalized eigenproblems.

PRIMME [Stathopoulos, 2007] is a C library for finding a number of eigenvalues and their corresponding eigenvectors of a real symmetric (or complex Hermitian) matrix. This library provides a multimethod eigensolver, based on Davidson/Jacobi-Davidson. Particular methods include GD+1, JDQMR, and LOBPCG. It supports preconditioning as well as the computation of interior eigenvalues.

JADAMILU [Bollhöfer and Notay, 2007] is an implementation of the Jacobi-Davidson method for computing smallest or interior eigenvalues of symmetric matrices. It provides a built-in preconditioner based on ILUPACK.

2.6 Optimization and Preconditioned Methods

Several packages provide implementations of methods based on some optimization strategy. One of such packages is SVDPACK, that contains a trace minimization eigensolver as mentioned in 2.4.

MPB [Johnson and Joannopoulos, 2001] is a package specific for electromagnetics simulation that includes two eigensolvers: preconditioned conjugate-gradient Rayleigh-quotient minimization and Davidson's method.

PDACG is a parallel implementation of the Deflation-Accelerated Conjugate Gradient (DACG) method [Gambolati *et al.*, 1992], to compute the smallest eigenpairs of a symmetric positive-definite matrix pair.

BLOPEX provides the Locally Optimal Block Preconditioned Conjugate Gradient (LOBPCG) method [Knyazev, 2001]. This package comes in several flavors: a Matlab function, a serial C library and a parallel C library to be combined with either Hypre or PETSc. The LOBPCG method is also available in ANASAZI (described in 2.2).

EIGIFP is a Matlab program that computes the smallest or largest eigenpairs of a symmetric matrix or a symmetric positive-definite matrix pair. It is based on the inverse free preconditioned Krylov subspace method [Golub and Ye, 2003].

SPAM implements the Subspace Projected Approximating Matrices technique for extending Davidson's method, as described in [Shepard *et al.*, 2001]. This method is applicable to symmetric eigenproblems.

3 Classification by Technical Features

This section provides a more vertical view of the libraries, classifying them with respect to several technical features. This section mostly covers current software listed in table 1.

Programming language According to the programming language in which the software is implemented, the following cases can be distinguished:

- Pure Matlab implementations: EIGIFP, INSYLAN, IRBLEIGS, and JDCG. These are typically prototype implementations of novel methods.
- Matlab functions with alternative C or Fortran implementation: BLKLAN, BLOPEX, JDQZ, and PROPACK. These allow fast testing of the method in Matlab before carrying out the C or Fortran coding.
- Fortran77 libraries: ARPACK, BLZPACK, PDACG, JADAMILU, QMRPACK, NA18 and SRRIT. Note that these libraries can usually be called from C/C++ code, if appropriate calling conventions are used.
- C libraries: SLEPc, PRIMME, JDBSYM and MPB. These can be called from C++ code without problems. SLEPc and PRIMME also provide a Fortran77 interface.
- Fortran90 libraries: SPAM and TRLAN. These can be called from C/C++ and Fortran77.
- C++ libraries: ANASAZI and IETL.
- Python packages: PYSPARSE.

Interface to user data An important factor of usability is the way in which the application programmer interfaces with the software, particularly how the matrices of the problem are to be represented. In this respect, the packages offer different mechanisms with varying degree of flexibility.

Most Matlab implementations allow the user to pass a Matlab matrix as well as to specify a Matlab function for the matrix-vector product.

Some C or Fortran libraries force the user to define a matrix-vector subroutine with a fixed prototype. This is the case of BLOPEX, JDBSYM, JDQZ, MPB, NA18, PRIMME, PROPACK, SPAM, SRRIT, and TRLAN.

Reverse communication is generally a good solution for implementing a package that is independent of the matrix representation. It is very flexible although the resulting code might be quite obscure. This approach is followed by ARPACK, BLZPACK, and QMRPACK.

SLEPc provides considerable flexibility with a data-structure neutral implementation, that allows different matrix formats as well as matrix-free computations.

The C++ libraries, IETL and ANASAZI, are object-oriented and thus the user has to work with C++ objects and classes that define a set of interfaces. The philosophy is similar in the Python and Java packages.

The least flexible interfaces correspond to BLKLAN, which requires a dense matrix, and PDACG, where the matrices are input through a file. JADAMILU requires the matrix to be provided in compressed sparse row format.

Parallel computing The following software supports MPI parallelization: SLEPc, ANASAZI, ARPACK, BLOPEX, BLZPACK, MPB, PDACG, PRIMME, and TRLAN. Note that most of them require that the user provides an MPI-parallel implementation of certain operations such as matrix-vector products.

PROPACK has been parallelized with the OpenMP shared-memory paradigm and has been tested in several parallel platforms.

Supported scalar types and precision The following libraries support both real and complex scalar types: SLEPc, ANASAZI, ARPACK, IETL, JADAMILU, MPB, PROPACK, PRIMME, and QMRPACK.

The following libraries work only with real scalars: BLZPACK, JDBSYM, NA18, PDACG, SPAM, SRRIT, and TRLAN.

The following libraries work only with complex scalars: BLKLAN and JDQZ.

With respect to precision, the following packages provide implementations for both single and double precision: SLEPc, ARPACK, BLZPACK, JADAMILU, MPB, PROPACK, QMRPACK, SPAM, and SRRIT.

References

Arbenz, P., U. L. Hetmaniuk, R. B. Lehoucq, and R. S. Tuminaro (2005). A Comparison of Eigensolvers for Large-scale 3D Modal Analysis using AMG-Preconditioned Iterative Methods. *Internat. J. Numer. Methods Engrg.*, 64(2):204–236.

Baglama, J., D. Calvetti, and L. Reichel (2003). Algorithm 827: `irbleigs`: A MATLAB Program for Computing a Few Eigenpairs of a Large Sparse Hermitian Matrix. *ACM Trans. Math. Softw.*, 29(3):337–348.

Bai, Z., T. Ericsson, and T. Kowalski (2000). Symmetric Indefinite Lanczos Method. In *Templates for the Solution of Algebraic Eigenvalue Problems: A Practical Guide* (edited by Z. Bai, J. Demmel, J. Dongarra, A. Ruhe, and H. van der Vorst), pp. 249–260. Society for Industrial and Applied Mathematics, Philadelphia.

Bai, Z. and G. W. Stewart (1997). Algorithm 776: SRRIT: A FORTRAN Subroutine to Calculate the Dominant Invariant Subspace of a Nonsymmetric Matrix. *ACM Trans. Math. Softw.*, 23(4):494–513.

Bergamaschi, L. and M. Putti (2002). Numerical Comparison of Iterative Eigensolvers for Large Sparse Symmetric Positive Definite Matrices. *Comput. Methods Appl. Mech. Engrg.*, 191(45):5233–5247.

Berry, M. W. (1992). SVDPACK: A Fortran-77 Software Library for the Sparse Singular Value Decomposition. Technical Report UT-CS-92-159, Department of Computer Science, University of Tennessee.

Bollhöfer, M. and Y. Notay (2007). JADAMILU: a Software Code for Computing Selected Eigenvalues of Large Sparse Symmetric Matrices. *Comput. Phys. Commun.*, 177(12):951–964.

Braconnier, T. (1993). The Arnoldi-Tchebycheff Algorithm for Solving Large Nonsymmetric Eigenproblems. Technical Report TR/PA/93/25, CERFACS, Toulouse, France.

Cullum, J. K. and R. A. Willoughby (1985). *Lanczos Algorithms for Large Symmetric Eigenvalue Computations. Vol. 1: Theory*. Birkhaüser, Boston, MA. Reissued by SIAM, Philadelphia, 2002.

Fokkema, D. R., G. L. G. Sleijpen, and H. A. van der Vorst (1998). Jacobi–Davidson Style QR and QZ Algorithms for the Reduction of Matrix Pencils. *SIAM J. Sci. Comput.*, 20(1):94–125.

Freund, R. W. and N. M. Nachtigal (1996). QMRPACK: a Package of QMR Algorithms. *ACM Trans. Math. Softw.*, 22(1):46–77.

Gambolati, G., F. Sartoretto, and P. Florian (1992). An Orthogonal Accelerated Deflation Technique For Large Symmetric Eigenproblems. *Comput. Methods Appl. Mech. Engrg.*, 94(1):13–23.

Golub, G. H. and R. Underwood (1977). The Block Lanczos Method for Computing Eigenvalues. In *Mathematical Software III* (edited by J. Rice), pp. 364–377. Academic Press, New York, NY, USA.

Golub, G. H. and Q. Ye (2003). An Inverse Free Preconditioned Krylov Subspace Method for Symmetric Generalized Eigenvalue Problems. *SIAM J. Sci. Comput.*, 24(1):312–334.

Johnson, S. G. and J. D. Joannopoulos (2001). Block-Iterative Frequency-Domain Methods for Maxwell's Equations in a Planewave Basis. *Opt. Express*, 8(3):173–190.

Jones, M. T. and M. L. Patrick (1993). The Lanczos Algorithm for the Generalized Symmetric Eigenproblem on Shared-Memory Architectures. *App. Numer. Math.*, 12(5):377–289.

Knyazev, A. V. (2001). Toward the Optimal Preconditioned Eigensolver: Locally Optimal Block Preconditioned Conjugate Gradient Method. *SIAM J. Sci. Comput.*, 23(2):517–541.

Larsen, R. M. (1998). Lanczos Bidiagonalization with Partial Reorthogonalization. Technical Report PB-537, Department of Computer Science, University of Aarhus, Aarhus, Denmark. Available at http://www.daimi.au.dk/PB/537.

Lehoucq, R. B. and J. A. Scott (1996). An Evaluation of Software for Computing Eigenvalues of Sparse Nonsymmetric Matrices. Preprint MCS-P547-1195, Argonne National Laboratory.

Lehoucq, R. B., D. C. Sorensen, and C. Yang (1998). *ARPACK Users' Guide, Solution of Large-Scale Eigenvalue Problems by Implicitly Restarted Arnoldi Methods*. Society for Industrial and Applied Mathematics, Philadelphia, PA.

Liu, G., W. Xu, and S. Qiao (2004). Block Lanczos Tridiagonalization of Complex Symmetric Matrices. Technical Report CAS-04-07-SQ, Department of Computing and Software, McMaster University, Hamilton, Ontario, Canada.

Marques, O. A. (1995). BLZPACK: Description and User's Guide. Technical Report TR/PA/95/30, CERFACS, Toulouse, France.

Notay, Y. (2002). Combination of Jacobi-Davidson and Conjugate Gradients for the Partial Symmetric Eigenproblem. *Numer. Linear Algebra Appl.*, 9(1):21–44.

Parlett, B. N. and D. S. Scott (1979). The Lanczos Algorithm with Selective Orthogonalization. *Math. Comp.*, 33:217–238.

Sadkane, M. and R. B. Sidje (1999). Implementation of a Variable Block Davidson Method with Deflation for Solving Large Sparse Eigenproblems. *Numer. Algorithms*, 20(2–3):217–240.

Shepard, R., A. F. Wagner, J. L. Tilson, and M. Minkoff (2001). The Subspace Projected Approximate Matrix (SPAM) Modification of the Davidson Method. *J. Comput. Phys.*, 172(2):472–514.

Stathopoulos, A. (2007). Nearly Optimal Preconditioned Methods for Hermitian Eigenproblems under Limited Memory. Part I: Seeking One Eigenvalue. *SIAM J. Sci. Comput.*, 29(2):481–514.

Stathopoulos, A. and C. F. Fischer (1994). A Davidson Program for Finding a Few Selected Extreme Eigenpairs of a Large, Sparse, Real, Symmetric Matrix. *Comput. Phys. Commun.*, 79:268.

Stewart, W. J. and A. Jennings (1981). Algorithm 570: LOPSI: A Simultaneous Iteration Method for Real Matrices [F2]. *ACM Trans. Math. Softw.*, 7(2):230–232.

Wu, K. and H. Simon (2000). Thick-Restart Lanczos Method for Large Symmetric Eigenvalue Problems. *SIAM J. Matrix Anal. Appl.*, 22(2):602–616.