

# Sca/LAPACK

LAPACK is an open-source subroutine library for solving the most common problems in dense numerical linear algebra. It is designed to run efficiently on modern processors by making extensive use of Level-3 BLAS. ScaLAPACK provides most of the functionalities of LAPACK but for distributed memory parallel systems. The goals of the Sca/LAPACK projects are to provide efficiency, portability, scalability, flexibility, reliability, ease of maintenance, and ease of use software for computational science problems.

## FUNCTIONALITIES

LAPACK provides routines for solving:

- Linear Equations (SV) for nonsymmetric, symmetric, and symmetric positive definite matrices using LU, LDLT, and Cholesky factorizations, respectively.
- Linear Least Squares (LLS)
- Generalized Linear Least Squares (LSE and GLM)
- Standard Eigenvalue and Singular Value Problems
  - Symmetric Eigenproblems (SEP)
  - Nonsymmetric Eigenproblems (NEP)
  - Singular Value Decomposition (SVD)
- Generalized Eigenvalue and Singular Value Problems
  - Generalized Symmetric Definite Eigenproblems (GSEP)
  - Generalized Nonsymmetric Eigenproblems (GNEP)
  - Generalized Singular Value Decomposition (GSVD)

Input matrix layout can be dense, banded, tridiagonal, bidiagonal, or packed (for symmetric or triangular matrices).

For each driver, an expert version is provided, and subroutines are defined in 4 ways: real (s), complex (c), double precision (d), and double complex (z).

## DISTRIBUTION

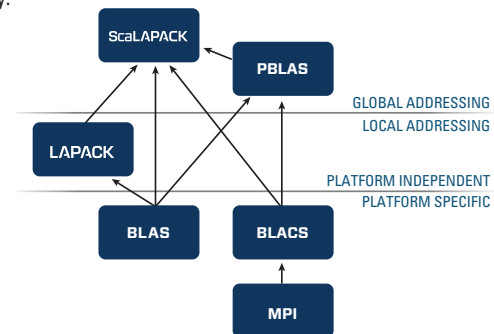
The Sca/LAPACK source code is distributed through <http://www.netlib.org/> under modified BSD license. The libraries are regularly tested on numerous machines using multiple computers. The Sca/LAPACK APIs have been adopted by many vendors, and the Sca/LAPACK public version provides a reference implementation of state-of-the-art algorithms for a wide set of problems. LAPACK has been incorporated into the following commercial packages (often with some shared memory LAPACK implementations and the parallel distributed version of ScaLAPACK): AMD, Apple, Compaq, Fujitsu, Hewlett-Packard, Hitachi, IBM, Intel, MathWorks, NAG, NEC, PGI, Oracle, Rogue Wave. It is also distributed in most Linux distributions (e.g., Fedora, Debian, and Cygwin).

## DID YOU KNOW?

- EISPACK and LINPACK (ancestors of LAPACK) were two of the first libraries made publicly available.
- In 1979, the LINPACK benchmark was initially written for timing references. It has since become the popular benchmark that is used to rank the TOP500 computer list. A highly efficient implementation of the benchmark is HPL from UTK, which is a tuned version of PDGESHV from ScaLAPACK.
- LAPACK can solve the symmetric eigenvalue problem in five different ways. One can either use QR (STEQR), QR only eigenvalues (STERF), Bisection and Inverse Iteration (STEBZ+STEIN), Divide and Conquer (STEDC), or MRRR (STEGR). Each of these methods has its own importance, and the LAPACK drivers enable users to pick the appropriate one according to the problem at hand.
- LAPACK is written in Fortran and has a native C interface (since Nov. 2010).

## ARCHITECTURE DESIGN

LAPACK makes extensive use of BLAS calls. This enables LAPACK to maintain its efficiency when ported from one platform to another. ScaLAPACK software is multi-layered, enabling it to be portable and efficient. Matrices are in the 2D-block cyclic format, an important parameter for scalability and efficiency.



## USERS

Sca/LAPACK is used by most computational simulation codes to provide efficient, easy to use, and reliable numerical dense linear algebra methods. Many users do not even know that they are using Sca/LAPACK, while Matlab and Numeric Python are using LAPACK. Sca/LAPACK is used for a number of applications in science and engineering in areas such as quantum chemistry and physics, electromechanics, geophysics and seismology, plasma physics, nonlinear mechanics, chemically reactive flows, helicopter flight control, atomic structure calculation, cardio-magnetism, radar cross-sections, and two-dimensional elastodynamics. The package is used on matrices ranging in size from 2 to 30,000 for LAPACK, and ScaLAPACK is now successfully used on thousands of processors.

FIND OUT MORE AT



[www.netlib.org/lapack](http://www.netlib.org/lapack)

INNOVATIVE  
COMPUTING LABORATORY  
THE UNIVERSITY of TENNESSEE

IN COLLABORATION WITH

Berkeley  
UNIVERSITY OF CALIFORNIA

University of Colorado  
Denver

SPONSORED BY



FIND OUT MORE AT



[www.netlib.org/scalapack](http://www.netlib.org/scalapack)

# Sca/LAPACK

One of the main strengths of Sca/LAPACK is the widespread support and recognition from the international dense linear algebra community. Researchers, vendors, and individuals all over the world are regularly contributing to the Sca/LAPACK software library. The University of Tennessee's Innovative Computing Laboratory (ICL), the University of Colorado Denver, and the University of California, Berkeley are responsible for the development, integration, and verification of those contributions.

## ON-GOING WORK

Current activity consists of including new functionalities that enrich LAPACK's already impressive capabilities, adding new algorithms that provide faster and more accurate results, maintaining our libraries to guarantee their reliability, providing user support, and increasing ease of use so our users can focus on their research.

For more info please consult the summary of improvements at [www.netlib.org/lapack/improvement.html](http://www.netlib.org/lapack/improvement.html).

## ScaLAPACK 2.0.0

### NEW FUNCTIONALITIES

#### PxHSEQR: Nonsymmetric Eigenvalue Problem

*Contribution by Robert Granat, Bo Kågström, Meiyue Shao (Umeå University and HPC2N), and Daniel Kressner (EPF Lausanne)*

Compute the eigenvalues of a nonsymmetric real matrix. Implement the parallel distributed Hessenberg QR algorithm with the small bulge multi-shift QR algorithm together with aggressive early deflation.

#### PxSYEVR/PxHEEVR: MRRR (Multiple Relatively Robust Representations) algorithm for computing eigenpairs of large real symmetric or complex Hermitian matrices

*Contribution by Christof Voemel*

This parallel algorithm is derived from Parlett and Dhillon's SIAG-LA prize-winning work on sequential MRRR. Compared to other algorithms, parallel MRRR has some striking advantages.

First, for an  $n \times n$  matrix on  $p$  processors, a tridiagonal inverse iteration can require up to  $O(n^3)$  operations and  $O(n^2)$  memory on a single processor to guarantee the correctness of the computed eigenpairs. MRRR is guaranteed to produce the right answer with  $O(n^2/p)$  memory, and it does not need reorthogonalization. Second, MRRR allows the computation of subsets at reduced cost, whereas QR and Divide & Conquer do not. For computing  $k$  eigenpairs, the tridiagonal parallel MRRR requires  $O(nk/p)$  operations per processor.

### EASE OF USE

#### BLACS revamping

With ScaLAPACK 2.0, the (MPI) BLACS is now completely integrated into ScaLAPACK. Linking a ScaLAPACK application now only requires linking with `libscalapack.a`, `liblapack.a`, `libblas.a`, and possibly the MPI libraries.

## LAPACK 3.4.0

### NEW FUNCTIONALITIES

#### xGEQRT: QR factorization (improved interface) *Contribution by Rodney James (UC Denver)*

xGEQRT is analogous to xGEQRF with a modified interface which enables better performance when the blocked reflectors need to be reused. The companion subroutines xGEMQRT apply the reflectors.

#### xGEQRT3: recursive QR factorization *Contribution by Rodney James (UC Denver)*

The recursive QR factorization enables cache-oblivious and enables high performance on tall and skinny matrices.

#### xTPQRT: Communication-Avoiding QR sequential kernels

*Contribution by Rodney James (UC Denver)*

These subroutines are useful for updating a QR factorization and are used in sequential and parallel Communication Avoiding QR. These subroutines support the general case Triangle on top of Pentagon which includes as special cases the so-called triangle on top of triangle and triangle on top of square. This is the right-looking version of the subroutines and the subroutines are blocked. The T matrices and the block size are part of the interface. The companion subroutines xTPMQRT apply the reflectors.

#### xSYSVK: LDL<sup>T</sup> with rook pivoting

*Contribution by Craig Lucas (University of Manchester and NAG) and Sven Hammarling (NAG)*

These subroutines enable better stability than the Bunch-Kaufman pivoting scheme currently used in LAPACK xSYSV solvers; also, the elements of L are bounded, which is important in some applications. The computational time of xSYSVK is slightly higher than the one of xSYSV.

### EASE OF USE

#### CMAKE build system

We are striving to help our users install our libraries seamlessly on their machines. The CMAKE team contributed to our effort to port LAPACK and ScaLAPACK under the CMAKE build system. Building under Windows has never been easier. This also allows us to release dll for Windows, so users no longer need a Fortran compiler to use LAPACK under Windows.

#### Doxygen documentation

LAPACK routine documentation has never been more accessible. See <http://www.netlib.org/lapack/explore-html/>.

#### New website allowing for easier navigation

#### LAPACK - Standard C language APIs for LAPACK

Since LAPACK 3.3.0 and MKL 10.3, LAPACK includes new C interfaces. With the LAPACK 3.4.0 release, LAPACK is directly integrated within the LAPACK library.

FIND OUT MORE AT



[www.netlib.org/lapack](http://www.netlib.org/lapack)

INNOVATIVE  
COMPUTING LABORATORY  
THE UNIVERSITY of TENNESSEE

IN COLLABORATION WITH

Berkeley  
UNIVERSITY OF CALIFORNIA



University of Colorado  
Denver

SPONSORED BY



FIND OUT MORE AT



[www.netlib.org/scalapack](http://www.netlib.org/scalapack)