

# 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 [including shared memory machines] by making extensive use of Level-3 BLAS. ScaLAPACK provides the same functionalities as 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

ScaLAPACK provides routines for solving:

- Linear Equations [SV] for nonsymmetric, symmetric, symmetric positive definite matrices using respectively LU, LDLT, Cholesky factorizations.
- 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], double complex [z].

## DISTRIBUTION

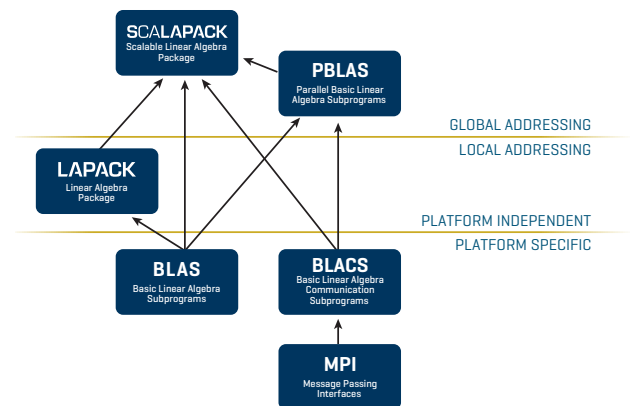
The Sca/LAPACK source code are distributed through <http://www.netlib.org/> and are publicly available. The libraries are regularly tested on numerous machines using multiple computers.

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, SUN, Visual Numeric]. It is also distributed in most Linux distributions [e.g., Fedora, Debian, Cygwin, etc.].

## 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 on all DOE/DOD machines and by many others 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.

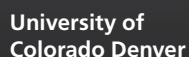
Sca/LAPACK is used for a number of applications of science and engineering in areas such as quantum chemistry and physics, electromechanics, geophysics and seismology, plasma physics, nonlinear mechanics, chemically reactive flows, helicopter flight control, atomic structure calculation, cardio-magnetism, radar cross-sections, and two-dimensional elastodynamics.

The package is used on matrices ranging from size to 2 to 30,000 for LAPACK, and ScaLAPACK is now successfully used on thousands of processors.

**LAPACK** <http://www.netlib.org/lapack/>

**SCALAPACK** <http://www.netlib.org/scalapack/>

A COLLABORATION OF



SPONSORED BY



# 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] 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 algorithms and new functionalities in Sca/LAPACK while maintaining the library. Below is a list of new algorithms and improvements that were added in the latest LAPACK 3.2 version:

- 1 Extra Precise Iterative Refinement: New linear solvers that "guarantee" fully accurate answers [more accurate].
- 2 XBLAS, or portable "extra precise BLAS": New linear solvers in [1] depend on these to perform iterative refinement [more accurate].
- 3 Mixed precision iterative refinement subroutines for exploiting fast single precision hardware. On platforms like the Cell processor that do single precision much faster than double, linear systems can be solved many times faster. Even on commodity processors there is a factor of 2 in speed between single and double precision [faster].
- 4 Non-Negative Diagonals and High Performance on Low-Profile Matrices from Householder QR: New QR routines that guarantee both positive real diagonal entries of R, and automatically reduce the cost of QR from  $O(n^3)$  to  $O(n^2)$  or less for matrices stored in a dense format but with a "narrow profile" [faster].
- 5 High accuracy SVD routine for dense matrices, which can compute tiny singular values to many more correct digits than xGESVD when the matrix has columns differing widely from the norm and usually runs faster than xGESVD too [more accurate and faster].
- 6 Rectangular Full Packed format: The RFP format [SF, HF, PF, TF] enables efficient routines with optimal storage for symmetric, Hermitian or Triangular matrices [faster].
- 7 Pivoted Cholesky: computing the Cholesky factorization with complete pivoting of a symmetric positive, semi-definite matrix [more accurate].
- 8 Added some variants for the one-sided factorization: LU gets Right-Looking, Left-Looking, Crout and Recursive, QR gets Right-Looking and Left-Looking, Cholesky gets Left-Looking, Right-Looking and Top-Looking. These new variants could be faster than the previous ones, depending on the relative speed of the underlying BLAS.

## DID YOU KNOW?

- LINPACK [ancestor of LAPACK] was one of the first libraries made publicly available.
- In 1979, the LINPACK benchmark was initially written for timing references. It has subsequently become the popular benchmark that is now used to rank the TOP500 computer list. A highly efficient implementation of the benchmark is HPL from UTK, which is a tuned version of PDGESV from ScaLAPACK.
- Straightforward parallel programming interfaces to ScaLAPACK are now available [e.g., LFC from ICL at UTK]. These enable scientists to easily use parallel distributed machines.
- LAPACK can solve the symmetric eigenvalue problem in five different ways. One can either use QR [STEQQR], 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 are able to pick the appropriate one according to the user problem.
- LAPACK is written in F77 and uses F90 features but a C version is also available: CLAPACK [via Netlib].
- LAPACK can run up to 100 times slower if it is not calling an optimized BLAS library.
- Matlab uses its own LAPACK library behind the scene. The performance of your Matlab is thus closely related to the performance of LAPACK.
- The BLACS standard interface and the BLACS library were written before the MPI standard and before any MPI library ever existed.
- The BLAS/LAPACK/BLACS/ScaLAPACK test and timing suites provide a convenient and exhaustive way of testing and timing a third party library.
- LAPACK, CLAPACK and ScaLAPACK are now available for Windows.
- A great forum is available for support and discussions at <http://icl.eecs.utk.edu/lapack-forum/>

 **LAPACK** <http://www.netlib.org/lapack/>

 **SCALAPACK** <http://www.netlib.org/scalapack/>

A COLLABORATION OF



University of  
Colorado Denver

SPONSORED BY

