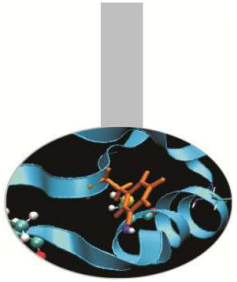
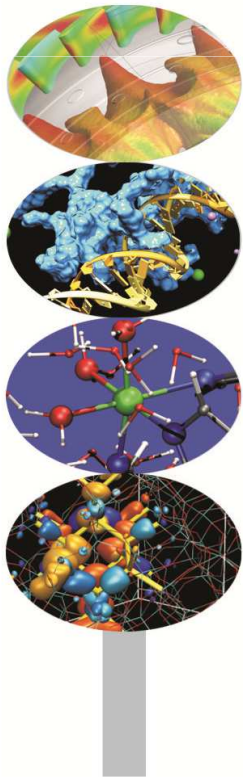
CINECA SCAI
SuperComputing Applications and Innovation



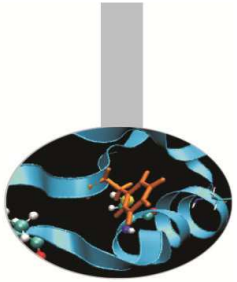
Introduction to Scientific Libraries



M. Guarrasi,
M. Cremonesi, F. Affinito
- *CINECA*

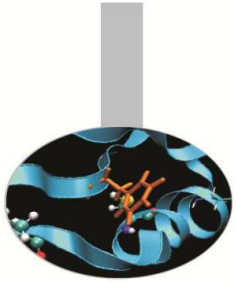
2015/10/27

CINECA



Numerical Libraries

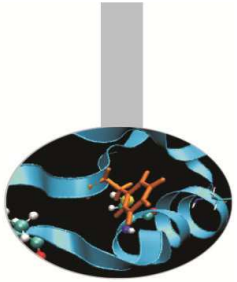
- Groups of functions or subroutines
- They implement various numerical algorithms, e.g:
  - Simple arithmetic operations;
  - Linear algebra;
  - FFT;
  - Solver for minimization problems;
  - Random generators;



Numerical Libraries

Why we should use NLs?

- Improve modularity
- Standardization
- Portability
- Efficency
- Ready to use



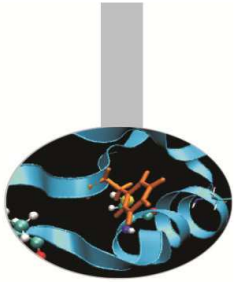
Numerical Libraries

Disadvantages:

- Hidden details
- Unknown Algorithms
- Too many confidence in the implementation:
  - E.g.: the algorithm works on only in some specific cases

CINECA SCAI
SuperComputing Applications and Innovation

Numerical Libraries



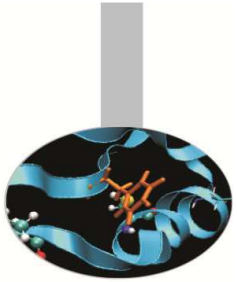
Agenda:

- Linear Algebra:
  - BLAS/PBLAS
  - LAPACK/SCALAPAK
  - ARPAK/P_ARPACK
  - PETSc
  - ….
- AMR:
  - METHIS/ParaMETHIS
  - PARAMESH
  - CHOMBO
- FFT:
  - FFTPACK
  - FFTW
  - 2DECOMP&FFT
  - P3DFFT
- I/O:
  - HDF5
  - NETCDF

CINECA

CINECA SCAI
SuperComputing Applications and Innovation

Numerical Libraries (Linear Algebra)



- BLAS
- LAPACK

Serial Linear Algebra Packages

essl (IBM AIX)
mkl (Intel)
acml (AMD)
plasma (ICL – Univ. Tennessee)
magma (ICL – Univ. Tennessee) → hybrid

threaded

- PBLAS
- ScaLAPACK

Parallel (distributed) Linear Algebra Packages (for dense matrices)

dplasma (ICL – Univ. Tennessee)

- ARPACK
- P_ARPACK

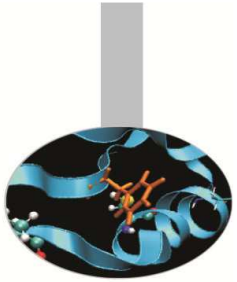
Eigenvalues Problems (for sparse matrices)

- PETSc

→ Sparse (non-)Linear Systems

CINECA

Linear Algebra Libraries



BLAS/CBLAS

- La *Basic Linear Algebra Subprograms* is one of the first libraries (written in 1979).
- It includes simple operation between vectors and matrixes (e.g. scalar product, scalar operations, transposition,..);
- It is currently used by other numerical libraries;
- Several distributions was created.
- Available on several architectures.

Language: FORTRAN, C

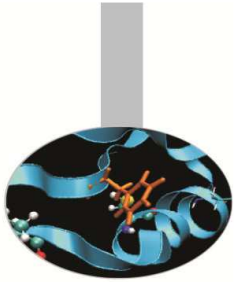
Availability: public domain

Developers: Jack Dongarra, ORNL and Eric Grosse, Bell Labs

Distributors: NETLIB

Ref.: The University of Tennessee at Knoxville and Bell Laboratories

Linear Algebra Libraries



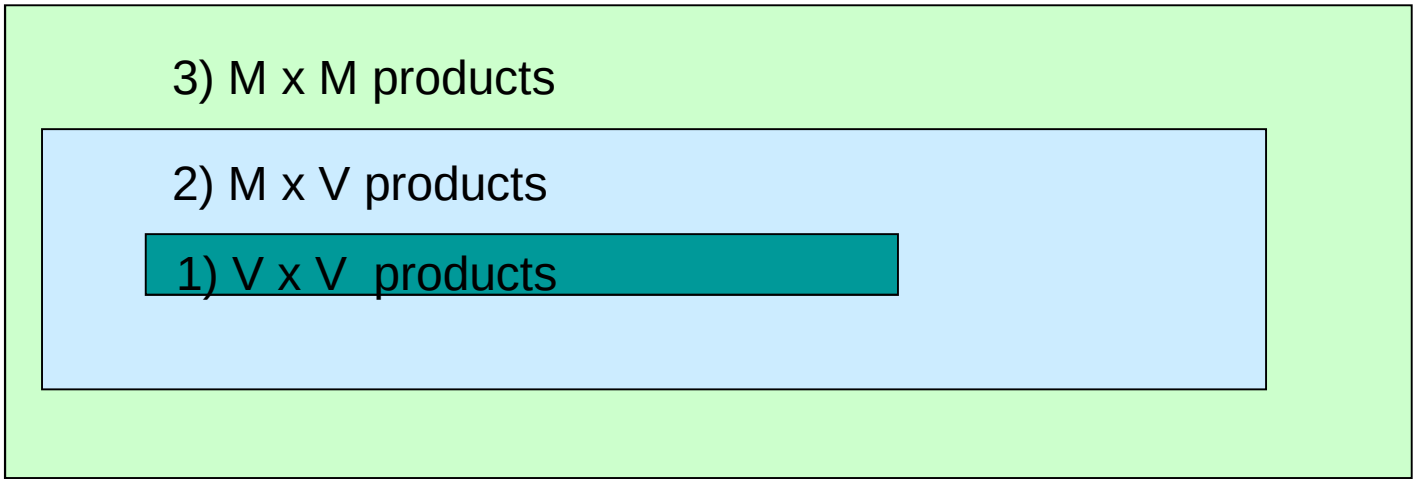
BLAS/CBLAS

BLAS lev. 1: Fortran subroutine for simple algebra operations (scalar-vector op.s). Developed in 1977.

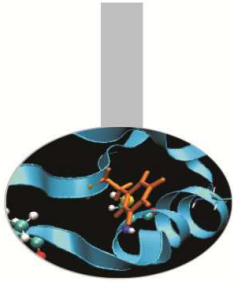
BLAS lev. 2: Vectors-matrices operations. Written in Fortran between 1984 and1986.

BLAS lev. 3: Fortran subroutine for matrices-matrices operations. Written in1988.

Blas lev. 2 and 3 are well suited for modern HPC cluster. Used is several libraries (e.g.: LAPACK).



Linear Algebra Libraries



BLAS/CBLAS

BLAS subroutines are usable both for real and complex data, both in sigle and double precision.

Vector-scalar operations(O(n))

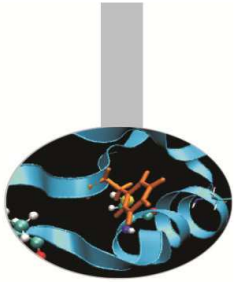
- *SWAP – vector swap
- *COPY – vector copy
- *SCAL – change scale factor
- *NRM2 – norm of a vector
- *AXPY – sum: Y + A*X

Where * is the type of variables used in the operations:

S –Single precision, **D** – Double precision,

C – Complex, **Z** – Complex Double precision

Linear Algebra Libraries



BLAS/CBLAS

Vector operations ($O(n^2)$)

*_GE_MV – product of vectors (all matrices)

*_HE_MV – product of vectors (Hermitian matrices)

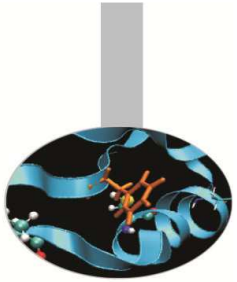
*_SY_MV – product of vectors (Symmetric matrices)

Where * is the type of variables used in the operations:

S –Single precision, **D** – Double precision,

C – Complex, **Z** – Complex Double precision

Linear Algebra Libraries



BLAS/CBLAS

Matrix-Matrix Operations ($O(n^3)$)

*_GE_MM – matrix multiplication (all matrices)

*_HE_MM – matrix multiplication (Hermitian matrices)

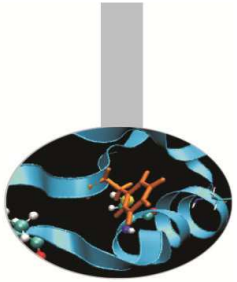
*_SY_MM – matrix multiplication (Symmetric matrices)

Where * is the type of variables used in the operations:

S –Single precision, **D** – Double precision,

C – Complex, **Z** – Complex Double precision

Syntax for BLAS/CBLAS



Schema for routine sintax : XYYZZZ

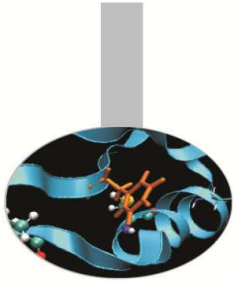
X Data tyte → S = REAL
D = DOUBLE PRECISION
C = COMPLEX
Z = DOUBLE COMPLEX

YY Type of matrix → (GE = general, SY = symmetric, HE = hermitian)

ZZZ Algorithm for computation (e.g.: MM, MV)

NOTE: This is true for almost all the routines, **but not for all**!

Linear Algebra Libraries



BLACS

The *Basic Linear Algebra Communication Subprograms* is a low level library that distributes elementary operation between vectors and matrices using distributed memory clusters. It was ported on several architectures, and it uses a lot of different communication interfaces (e.g. MPI). It is currently used by a lot of high level numerical (e.g.: scaLAPACK).

Language: C, FORTRAN

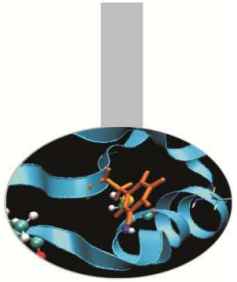
Availability: public domain

Developers: Jack J. Dongarra and R. Clint Whaley

Distributors: NETLIB

Ref.: The University of Tennessee at Knoxville and Bell Laboratories

Linear Algebra Libraries



LINPACK

Fortran Library created for solving linear equations systems and for Least Squares Fitting operations. It contains some subroutines devoted to solve many types of matrix problems (e.g.: generic, band, triangular Hermitian matrices).

Language: FORTRAN

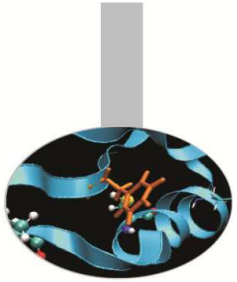
Availability: public domain

Developers: Jack Dongarra, Jim Bunch, Cleve Moler and Pete Stewart

Distributors: NETLIB

Ref.: The University of Tennessee at Knoxville and Bell Laboratories

Linear Algebra Libraries



LAPACK/LAPACKE

It contains a some FORTRAN Subroutines and C functions for solving linear algebra problems (e.g.: linear equations systems, Least Squares Fitting operations, eigenvalues problems, …). It can be considered very similar to LINPACK and EISPACK, but with increased performances.

Language: FORTRAN, C

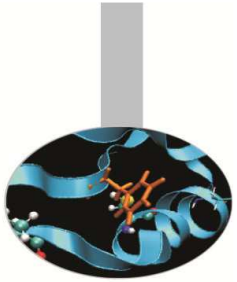
Availability: public domain

Developers: Jack Dongarra, ORNLand Eric Grosse, Bell Labs

Distributors: NETLIB

Ref.: The University of Tennessee at Knoxville and Bell Laboratories

Linear Algebra Libraries



LAPACK/LAPACKE

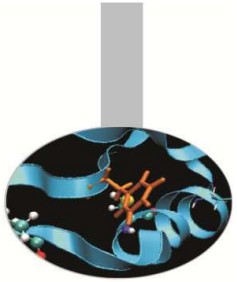
It uses the low level BLAS library (especially Level 3) to increase the performance of its subroutines.

For this reason in order to install this library it is mandatory to install also an optimized version of BLAS library.

You can also install the ATLAS library instead of BLAS, since it already contains an optimized version of BLAS

Warning: Standard version of BLAS probably is not optimized for your cluster, thus it could not be installed!

Linear Algebra Libraries



ATLAS

The tool Automatically Tuned Linear Algebra Software contains an optimized version of the most important Linear Algebra Libraries including also BLAS and LAPACK

Language: FORTRAN, C

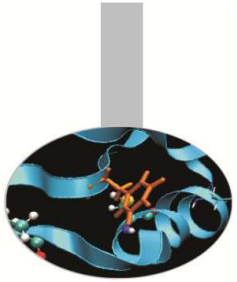
Availability: public domain / open source

Developers: R. Clint Whaley, Antoine Petitet, Jack Dongarra

Distributors: sourceforge.net

Ref.: http://math-atlas.sourceforge.net/

Linear Algebra Libraries



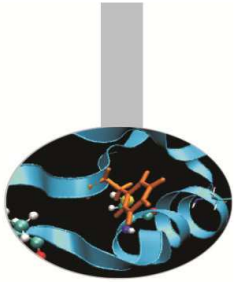
GotoBLAS

In scientific computing, GotoBLAS and GotoBLAS2 are open source implementations of the BLAS (Basic Linear Algebra Subprograms) API with many hand-crafted optimizations (modifying by hand the assembler code of BLAS) for specific processor types. GotoBLAS was developed by Kazushige Goto at the Texas Advanced Computing Center. As of 2003, it was used in seven of the world's ten fastest supercomputers.

Currently the work started with GotoBLAS continue in the framework of OpenBLAS project.

Ref.: http://www.tacc.utexas.edu/tacc-projects/gotoblas2
http://xianyi.github.com/OpenBLAS/

Linear Algebra Libraries



PLASMA

(Parallel Linear Algebra Software for Multi-core Architectures)

It was made to work on multi-core processors. Written in FORTRAN and C (only wrapper for FORTRAN routines), it solves linear equation systems. Similar to LAPACK but with less functionalities (e.g.: it doesn't contain functions to solve eigenvalues problems, it doesn't support band and sparse matrices optimized solvers).

Language: FORTRAN, C

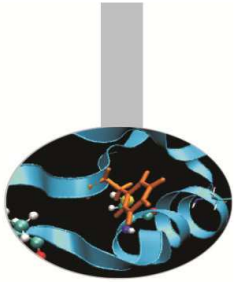
Availability: public domain

Developers: Various

Distributors: University of Tennessee

Ref.: Dep. Electrical Engineering and Computer Science, University of Tennessee at Knoxville

Linear Algebra Libraries



LAPACK++

Similar to LAPACK, but written ic C++. It solves linear equation systems and eigenvalues problems.

It contains some specific functions to solve problems that uses different types of matrices (e.g.: symmetric, triangular, tridiagonal, …).

Language: C++

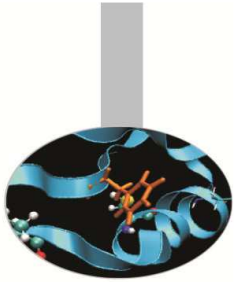
Availability: public domain

Developers: Roldan Pozo, Mathematical and Computational Sciences Division, NIST

Distributors: NETLIB

Ref.: The University of Tennessee at Knoxville and Bell Laboratories

Linear Algebra Libraries



SCALAPACK

Library for solving linear algebra problems (e.g matrix multiplication, linear equation systems, eigenvalues problems) on distributed memory architectures.

It is based on MPI and PVM libraries.

It contains also specific functions for dense, tridiagonal, and band matrices.

Languages: FORTRAN, C

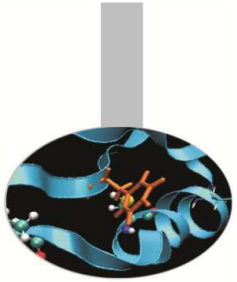
Availability: public domain

Developers: various

Distributors: NETLIB

Ref.: The University of Tennessee at Knoxville and Bell Laboratories

Linear Algebra Libraries



SCALAPACK

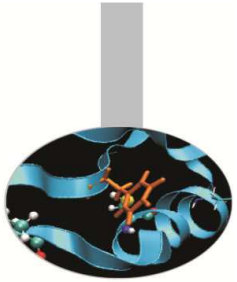
The ScaLAPACK library is (or should be…):

- efficient
- Scalable (when the number of cores and the size of the problem increase)
- Reliable (the results are the same also changing the number of CPU)
- Portable (results and (in some cases) parallel efficiency is the same also changing the system)

ScaLAPACK is written Fortran (with few C functions).

Based on BLAS, LAPACK and BLACS.

Linear Algebra Libraries



AZTEC

Parallel library for iterative solution of linear algebra problems. It includes several solvers, and preconditioners. Particularly well suited for distributed memory clusters. It contains also some tools for data conversion and query for searching inside a matrix.

Language: FORTRAN, C

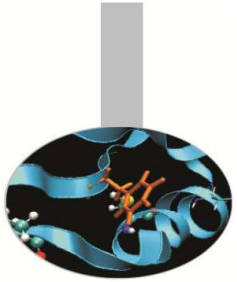
Availability: free

Developers: Ray S. Tuminaro, John N. Shadid, Mike Heroux

Distributors: Sandia National Laboratories

Ref.: Sandia National Laboratories

Linear Algebra Libraries



EISPACK

FORTRAN Libraries for solving eigenvalues and eigenvectors problems.
Can be considered as an extension of some LAPACK subroutines.

Language: FORTRAN

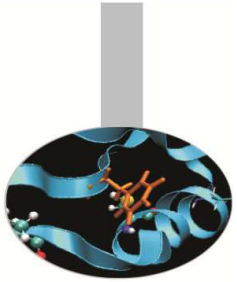
Availability: public domain

Developers: B.T. Smith, J.M. Boyle, B.S. Garbow, Y. Ikebe, V.C. Klema, C.B. Moler, and J.J. Dongarra

Distributors: NETLIB

Ref.: The University of Tennessee at Knoxville and Bell Laboratories

Linear Algebra Libraries



ARPACK/PARPACK

(Arnoldi package)

FORTRAN library similar to EISPACK. Particularly well suited for very large eigenvalues problems. PARPACK is the parallel version of ARPACK, and it uses BLACS and MPI libraries for parallelization.

Language: FORTRAN

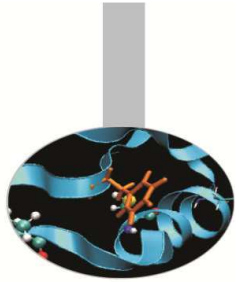
Availability: public domain

Developers: Rich Lehoucq, Kristi Maschhoff, Danny Sorensen, Chao Yang

Distributors: Rice University

Ref.: Computational & Applied Mathematics, Rice University, Houston

Linear Algebra Libraries



SLATEC Common Mathematical Library

SLATEC Common Mathematical Library is a FORTRAN 77 library of over 1400 general purpose mathematical and statistical routines. The code was developed at US Government research laboratories and is therefore in the public domain.

"SLATEC" is an acronym for the Sandia, Los Alamos, Air Force Weapons Laboratory Technical Exchange Committee.

It includes some other important libraries such as: BLAS, LINPACK, EISPACK

Language: FORTRAN

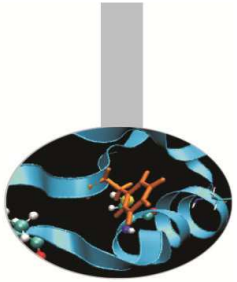
Availability: public domain

Developers: Sandia, Los Alamos, Air Force Weapons Laboratory

Distributors: NETLIB

Ref.: National Institute of Standards and Technology (NIST), Gaithersburg, Maryland.

Linear Algebra Libraries



UMFPACK

Group of C Functions for solving sparse and non-symmetrical linear equation systems.

Language: C

Availability: GNU GPL license.

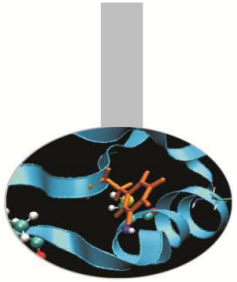
Developers: various

Distributors: Computer & Information Science & Engineering, University of Florida

Ref.: University of Florida

CINECA SCAI
SuperComputing Applications and Innovation

Linear Algebra Libraries



PETSC

The Portable, Extensible Toolkit for Scientific Computation (PETSc, pronounced PET-see; the S is silent), is a suite of data structures and routines developed by Argonne National Laboratory for the scalable (parallel) solution of scientific applications modeled by partial differential equations. It employs the Message Passing Interface (MPI) standard for all message-passing communication. PETSc includes a large suite of parallel linear and nonlinear equation solvers that are easily used in application codes written in C, C++, Fortran and now Python.

Particularly well suited for Sparse Linear Algebra.

Language: C, C++, Fortran, Python

Availability: open source

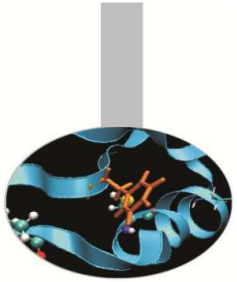
Developers: various

Distributors: Mathematics and Computer Science Division, Argonne National Laboratory

Ref.: Argonne National Laboratory

CINECA

Linear Algebra Libraries



GSL (GNU Scientific Library)

Multi-purpose library, written in C/C++, for solving several kinds of numerical problems (e.g.: linear equations systems, eigenvalues, FFT, statistic functions, random numbers generators, PDE, integrals, …).

Language: C/C++

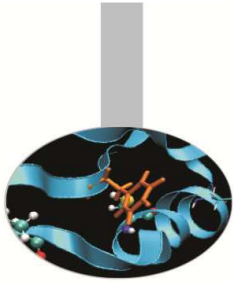
Availability: GNU license

Developers: various

Distributors: GNU

Ref.: GNU

Linear Algebra Libraries



MKL

Intel Math Kernel Library is a library of optimized math routines for science, engineering, and financial applications. Core math functions include BLAS, LAPACK, ScaLAPACK, sparse solvers, fast Fourier transforms, and vector math. The routines in MKL are hand-optimized specifically for Intel processors.

Language: Fortran, C, C++

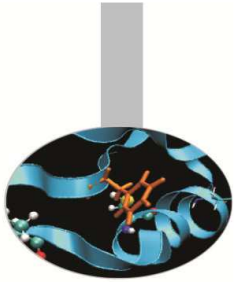
Availability: under license

Developers: Intel

Distributors: Intel

Ref.: Intel

Linear Algebra Libraries



ACML

(AMD Core Math Library)

AMD Core Math Library is a software development library released by AMD. This library provides mathematical routines optimized for AMD processors, and it also includes BLAS, LAPACK, ScaLAPACK, FFT.

Language: Fortran, C, C++

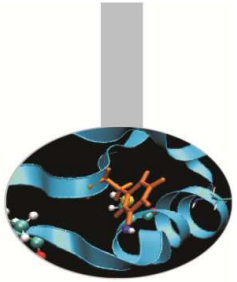
Availability: under license

Developers: AMD

Distributors: AMD

Ref.: AMD

Linear Algebra Libraries



ESSL/PESSL

(Engineering and Scientific Subroutine library)

IBM ESSL and Parallel ESSL (PESSL) are collections of state-of-the-art mathematical subroutines specifically designed to improve the performance of engineering and scientific applications on the IBM POWER processor-based clusters. They include BLAS, LAPACK, ScaLAPACK, sparse solvers, FFT...

PESSL uses MPI for parallelization.

Language: Fortran, C, C++

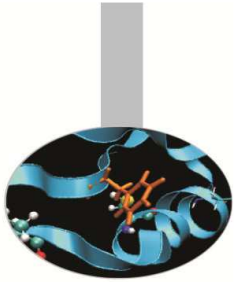
Availability: under license

Developers: IBM

Distributors: IBM

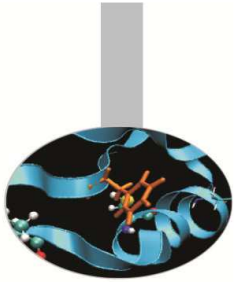
Ref.: IBM

Libraries for Adaptive Mesh Refinement (AMR)



- METIS
- ParaMETIS
- Paramesh
- Chombo

AMR libraries



METIS

METIS is a software package for graph partitioning that implements various multilevel algorithms. It was written in C, and it can generate, manipulate and partition, graphs grids and matrices. Particularly well suited for partitioning regular grids on a parallel distributed memory system.

Language: C

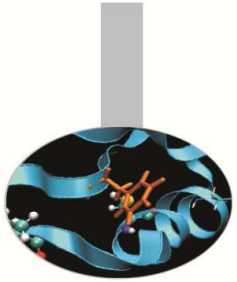
Availability: free for academy and research

Developers: George Karypis Lab

Distributors: George Karypis Lab

Ref.: Department of Computer Science & Engineering, University of Minnesota

AMR libraries



ParMETIS

ParMETIS is an MPI-based parallel library that implements a variety of algorithms for partitioning unstructured graphs, meshes, and for computing fill-reducing orderings of sparse matrices. ParMETIS extends the functionality provided by METIS and includes routines that are especially suited for parallel AMR computations and large scale numerical simulations

Language: C

Availability: free for academy and research

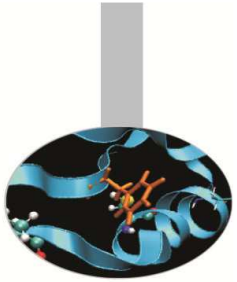
Developers: George Karypis Lab

Distributors: George Karypis Lab

Ref.: Department of Computer Science & Engineering, University of Minnesota

http://glaros.dtc.umn.edu/gkhome/views/metis

AMR libraries



PARAMESH

PARAMESH is a package of Fortran 90 subroutines designed to provide an application developer with an easy route to extend an existing serial code which uses a logically Cartesian structured mesh into a parallel code with adaptive mesh refinement(AMR).

Language: FORTRAN

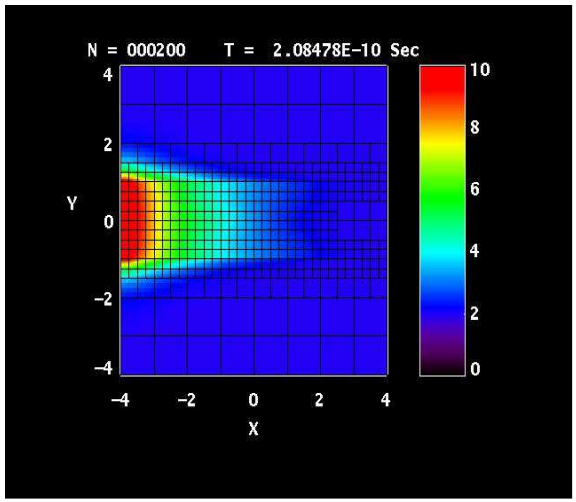
Availability: free for academy and research

Developers: Peter MacNeice, Kevin M. Olson et al.

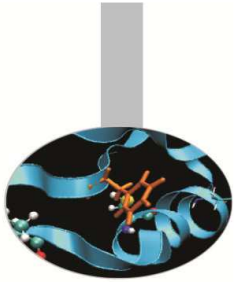
Distributors: Kevin M. Olson

Ref.: NASA e Drexel University

http://sourceforge.net/projects/paramesh/



AMR libraries



CHOMBO

Chombo provides a set of tools for implementing finite difference and finite volume methods for the solution of partial differential equations on block-structured adaptively refined rectangular grids.

Language: C

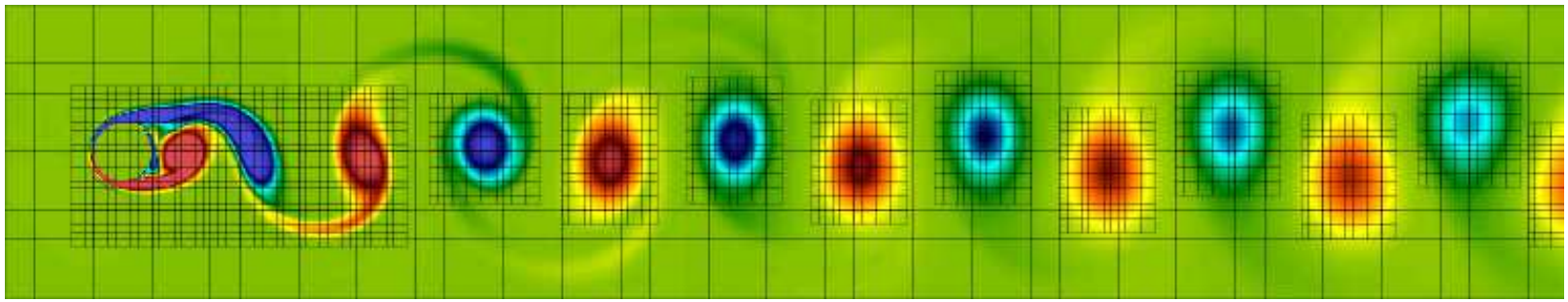
Availability: free for academy and research

Developers: P. Colella, D. T. Graves, J. N. Johnson et al.

Distributors: D. T. Graves

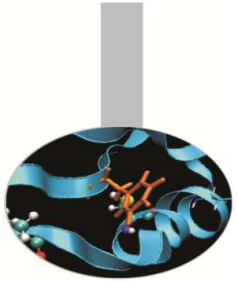
Ref.: Applied Numerical Algorithms Group - Computational Research Division - Lawrence Berkeley National Laboratory- Berkeley, CA, USA

https://commons.lbl.gov/display/chombo/



CINECA SCAI
SuperComputing Applications and Innovation

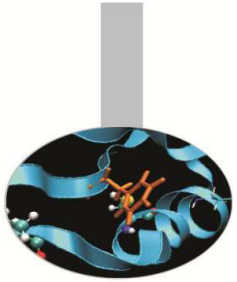
Fast Fourier Transform (FFT)



- FFTPACK
- FFTW
- 2Decomp&FFT
- P3DFFT

CINECA

Fast Fourier Transform (FFT)



FFTPACK

FFTPACK is a package of Fortran subroutines for the fast Fourier transform. It includes complex, real, sine, cosine, and quarter-wave transforms. It is included in the general-purpose mathematical library SLATEC.

Language: FORTRAN

Availability: public domain

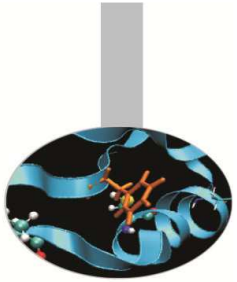
Developers: Paul N. Swarztrauber, National Center for Atmospheric Research, Boulder, CO

Distributors: NETLIB

Ref.: The University of Tennessee at Knoxville and Bell Laboratories

CINECA SCAI
SuperComputing Applications and Innovation

Fast Fourier Transform (FFT)



FFTW

(Fastest Fourier Transform in the West)

FFTW is a C subroutine library for computing the discrete Fourier transform (DFT) in one or more dimensions, of arbitrary input size, and of both real and complex data (as well as of even/odd data, i.e. the discrete cosine/sine transforms or DCT/DST).

The currently most used free library to compute DFTs.

It uses both MPI and OpenMP.

Language: C

Availability: public domain

Developers: Matteo Frigo and Steven G. Johnson.

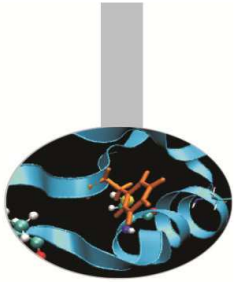
Distributors: FFTW

Ref.: MIT

CINECA

CINECA SCAI
SuperComputing Applications and Innovation

Fast Fourier Transform (FFT)



FFTW

(Fastest Fourier Transform in the West)

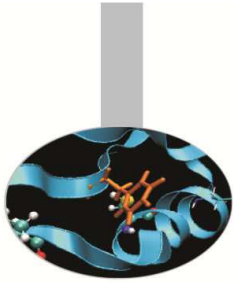
- Written in C (FORTRAN 77/90/2003 wrappers are included)
- FFTW adapt itself to your system and your problem: It check both total memory, size of your array, cache size register, CPU type, …
- … and it chose the best algorithm for you!
  - 2 setps is required:
    - Plan creation
    - Execution
- It supports FFTs for any different array size (< total memory)

CINECA

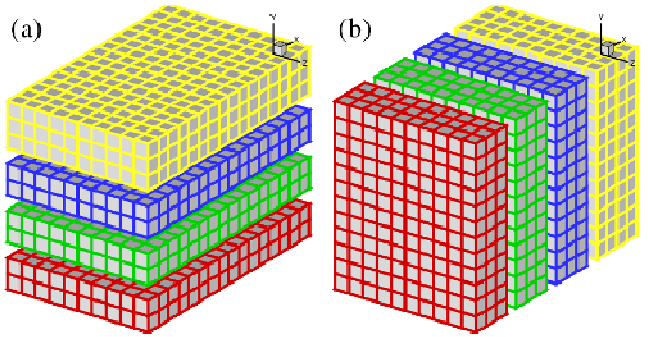
CINECA SCAI
SuperComputing Applications and Innovation

Fast Fourier Transform (FFT)

2Decomp&FFT



The 2DECOMP&FFT library is a software framework in Fortran to build large-scale parallel applications. It is designed for applications using three-dimensional structured mesh and spatially implicit numerical algorithms. At the foundation it implements a general-purpose 2D pencil decomposition for data distribution on distributed-memory platforms. On top it provides a highly scalable and efficient interface to perform three-dimensional distributed FFTs. The library is optimised for supercomputers and scales well to hundreds of thousands of cores. It relies on MPI but provides a user-friendly programming interface that hides communication details from application developers.



Language: FORTRAN

Availability: public domain

Developers: Ning LI.

Distributors: 2decomp.org

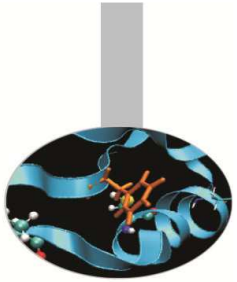
Ref.: HECToR Distributed Computational Science and Engineering (CSE) Service operated by NAG Ltd.

http://www.2decomp.org/

CINECA

CINECA SCAI
SuperComputing Applications and Innovation

Fast Fourier Transform (FFT)



P3DFFT

3DFFT is a scalable software library implementing three-dimensional spectral transforms. It has been used in a variety of codes from many areas of computational science. It has been tested and used on many high-end computational system. It uses two-dimensional domain decomposition in order to overcome a scaling bottleneck of one-dimensional decomposition. This allows the programs with this library to scale well on a large number of cores, consistent with bisection bandwidth scaling of interconnect of the underlying hardware system.

Language: C

Availability: public domain

Developers: Dmitry Pekurovsky.

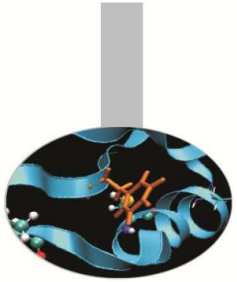
Distributors: 2decomp.org

Ref.: San Diego Supercomputer Center (SDSC) at UC San Diego.

https://code.google.com/p/p3dfft/

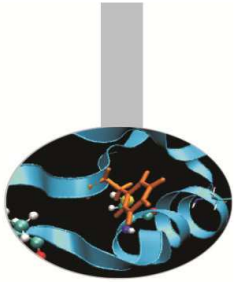
CINECA

I/O libraries



- HDF5
- NetCDF

I/O libraries



HDF5

HDF5 is a data model, library, and file format for storing and managing data. It supports an unlimited variety of datatypes, and is designed for flexible and efficient I/O and for high volume and complex data. HDF5 is portable and is extensible, allowing applications to evolve in their use of HDF5. The HDF5 Technology suite includes tools and applications for managing, manipulating, viewing, and analyzing data in the HDF5 format.

Language: Fortran/C/C++/Java

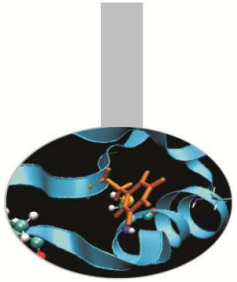
Availability: open source.

Developers: NCSA at the University of Illinois

Distributors: HDF Group

Ref.: http://www.hdfgroup.org

I/O libraries



NetCDF

NetCDF is a set of software libraries and self-describing, machine-independent data formats that support the creation, access, and sharing of array-oriented scientific data.

The NetCDF source-code is hosted at GitHub, and may be found directly at http://github.com/Unidata/netcdf-c.

NetCDF-4 (and pNetCDF) work with MPI version 2 (MPI-2).

Language: Fortran/C/C++/Java

Availability: open source.

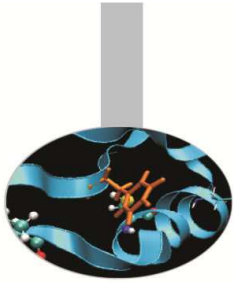
Developers: Unidata

Distributors: Unidata

Ref.: http://www.unidata.ucar.edu/netcdf

CINECA SCAI
SuperComputing Applications and Innovation

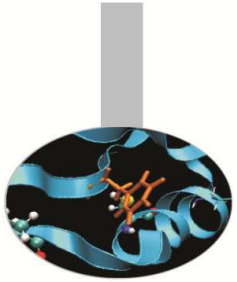
Numerical libraries (Some Suggestions)



- Names not resolved
- Static and Dynamic Libraries
- Compilation & Linking

CINECA

Numerical libraries (Names not resolved)



Names not resolved and name mangling

In some cases compilers (especially FORTRAN add an _ in front or at the end of the name of the variable. This can be a problem if you compile a code written using both FORTAN and C. If in the linking phase some not resolved name appears, and you are confident about the library that you linked, you can check the name of the subroutine simply by typing on yous PC:

nm <name_of_the_library>

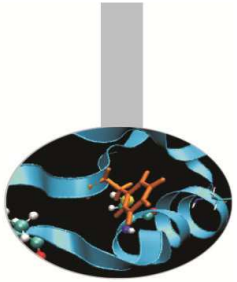
strings < name_of_the_library >

In Fortran	In C
CALL DROTG(…)	drotg_(…)
CALL DROTG(…)	_drotg(…)

Some compiler flag allow us to add or remove underscores, or if you prefer:

```
SUBROUTINE f() BIND(C,NAME="f")
END SUBROUTINE
```

Numerical libraries



Static and Dynamic Libraries

Static: the library linked in your code became part of your executable (*library*.a)

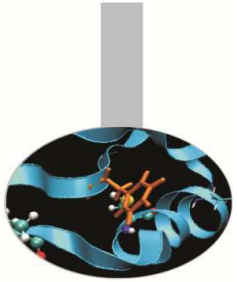
Dynamic the library linked in your code is load only during execution and only if it is called from a function/subroutine. (*library*.so)

Static Linking: Better performances, and more portability

Dynamic Linking: Small size executables

CINECA SCAI
SuperComputing Applications and Innovation

Numerical libraries (Compilation & Linking)



•In your code you must use the correct syntax to call functions included in the library.

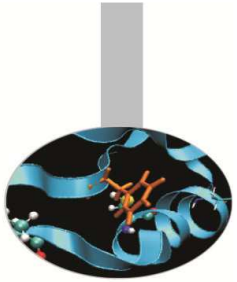
•You have also tell to the compiler the right position of the library and (if necessary of the header file.

•Sometimes, especially for some private libraries, it is necessary to use a particular compilation procedure.

•In the next slides we will show some examples.

CINECA

Numerical libraries (Compilation & Linking)



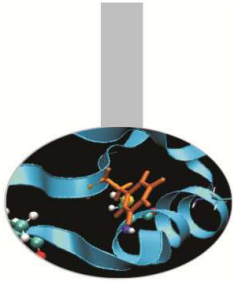
BLAS/CBLAS

```
Intel: ifort <programma> -L$MKLROOT/lib/intel64 \
   -lguide -lpthread –lmkl

PGI: pgf77 <programma> -L$PGI_ROOT/lib –lacml

GNU: gfortran <programma> -L$BLAS_ROOT/lib -lblas
```

Numerical libraries (Compilation & Linking)



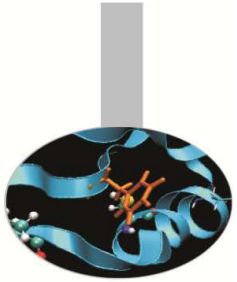
LINPACK

```
Intel: ifort <programma> -L$MKLROOT/lib/intel64 \
   -lguide -lpthread –lmkl

PGI: pgf77 <programma> -llapack –lblas

GNU: gfortran <programma> -L$LIB_ROOT/lib -llinpack
```

Numerical libraries (Compilation & Linking)



ScaLAPACK

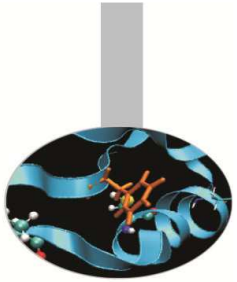
```
Intel: mpif77 <programma> -L$MKLROOT/lib/intel64 \
   -lmkl_scalapack_lp64  -lmkl_blacs_openmpi \
   -lmkl_intel_lp64 -lmkl_intel_thread -lmkl_core \
   -liomp5 –lpthread

PGI: pgf77 <programma> –Mmpi=mpich –Mscalapack

GNU: gfortran -L$LIB_ROOT/lib -lscalapack
```

CINECA SCAI
SuperComputing Applications and Innovation

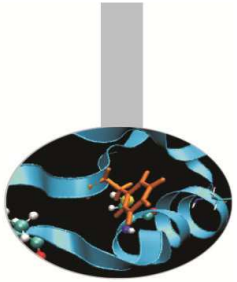
Numerical libraries (Compilation & Linking)



ESSL

```
module load bgq-xl
module load essl
bgxlf90_r -qarch=qp -qtune=qp <programma> \
   -L$ESSL_LIB –lesslb
runjob --np <mpi-procs> --ranks-per-node 1 \
   --envs OMP_NUM_THREADS=<threads>  \
   --exe  MatrMult-f-xl
```

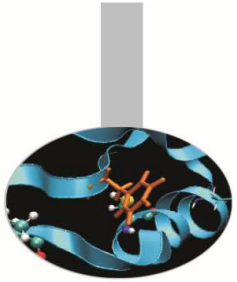
CINECA



Link Utili

- HPC Numerical Libraries:
  - 2 or 3 Days
  - Usually on March
  - http://www.hpc.cineca.it/content/hpc-numerical-and-domain-specific-libraries
  - Useful Links:
  - http://www.cineca.it/it/content/elenco-corsi-hpc-2015
- Online Material:
  - https://hpc-forge.cineca.it/files/CoursesDev/public/2015/HPC_Numerical_and_domain_specific_Libraries/Bologna/
  - http://www.hpc.cineca.it/content/numerical-libraries

CINECA SCAI
SuperComputing Applications and Innovation



Thank You

For futher informations please send me a mail: m.guarrasi@cineca.it

CINECA