

# Parallel & Distributed Computing: Lecture 33

Alberto Paoluzzi

December 18, 2019

1 Julia's LinearAlgebra.jl

2 GraphBLAS Forum

# Julia's LinearAlgebra.jl

# Standard Library

<https://docs.julialang.org/en/v1/stdlib/LinearAlgebra/>

# Low-level matrix operations

<https://docs.julialang.org/en/stable/stdlib/linalg/#Low-level-matrix-operations-1>

# BLAS functions

## How to Check If Julia Is Using OpenBLAS or Intel MKL

### BLAS Functions

LinearAlgebra.BLAS provides wrappers for some of the BLAS functions. Those BLAS functions that overwrite one of the input arrays have names ending in '!'. Usually, a BLAS function has four methods defined, for Float64, Float32, ComplexF64, and ComplexF32 arrays.

# LAPACK functions

## LAPACK Functions

LinearAlgebra.LAPACK provides wrappers for some of the LAPACK functions for linear algebra. Those functions that overwrite one of the input arrays have names ending in '!'.  
Usually a function has 4 methods defined, one each for Float64, Float32, ComplexF64 and ComplexF32 arrays.

## Intel MKL linear algebra in Julia.

MKL.jl is a package that makes Julia's linear algebra use Intel MKL BLAS and LAPACK instead of OpenBLAS. The build step of the package will automatically download Intel MKL and rebuild Julia's system image against Intel MKL

Intel MKL linear algebra in Julia.



# GraphBLAS Forum

# GraphBLAS Forum

The **GraphBLAS Forum** is an open effort to define **standard building blocks** for **graph algorithms** in the language of **linear algebra**.

We believe that the state of the art in constructing a large collection of graph algorithms in terms of **linear algebraic operations** is mature enough to support the emergence of a **standard set of primitive building blocks**.

We believe that it is critical to move quickly and define such a standard, thereby freeing up researchers to **innovate** and **diversify** at the level of **higher level algorithms** and graph analytics applications.

This effort **was inspired by** the **Basic Linear Algebra Subprograms** (BLAS) of **dense Linear Algebra**, and hence our working name for this standard is “the GraphBLAS”.

A key insight behind this work is that when a graph is represented by a sparse incidence or adjacency matrix, **sparse matrix-vector multiplication** is a step of **breadth first search**.

# SUITESPARSE : A Suite of Sparse Matrix Software

Suitesparse : A Suite of Sparse Matrix Software

SuiteSparse:GraphBLAS

# SUITESPARSE : Tim Davis News

People of the ACM, Jan 29, 2019

In an abstract for GraphBLAS, a project you're currently working on, you indicated that the project has the potential for creating a transformative shift in how graph algorithms are expressed. Will you explain this?

GraphBLAS is a community effort, including industry, academics, and government labs, that is working to design a library that can implement graph algorithms based on sparse linear algebra over semirings. There are lots of great graph libraries out there that don't exploit the linear algebraic abstraction, but most of what they do can be viewed as matrix operations on adjacency matrices. GraphBLAS makes the connection to linear algebra explicit.

# GraphBLAS Forum

- 1 Goals and premises
- 2 More information
- 3 Application Program Interface (API)
- 4 Reference Implementations
- 5 Projects developing implementations of the GraphBLAS
- 6 Graph analysis systems that integrate GraphBLAS
- 7 Workshops and conferences featuring the GraphBLAS (reverse chronological)
- 8 Blogs (and other interesting discussions) on GraphBLAS
- 9 GraphBLAS mailing list
- 10 Steering Committee