

Notes on the **bigalgebra** Package

Emmerson, Kane, Lewis

May 8, 2010

1 Introduction

The **bigalgebra** package provides linear algebra operations for “Big Matrices” defined in the **bigmemory** package. The package includes matrix and vector arithmetic operations and several matrix factorizations. Standard infix arithmetic operators are overloaded to use **bigalgebra** functions, keeping syntax familiar and simple.

The **bigalgebra** package may be optionally compiled with high-performance, multithreaded numeric libraries, which may also provide large-index support. Large indices support linear algebra computation on matrices and vectors with up to 2^{51} total entries (the usual R limit constrains objects to less than 2^{31} total entries). The package also supports computations on arrays that are larger than available RAM using **bigmemory**’s file-backed big matrix objects. The **bigalgebra** package presently supports double-precision computation.

2 Installation

The default installation uses available system and R BLAS and LAPACK libraries. The **bigalgebra** package may be optionally compiled with alternate libraries that offer performance and threading optimizations as well as large index support.

Installation assumes the standard R toolchain and GNU compiler suite, and requires setting one or more configuration options:

- `--with-incDir` Extra required include directories and compiler options.
- `--with-blasHeader` Alternate BLAS header file.
- `--with-blas` The BLAS library linker instructions.
- `--with-lapcakHeader` Alternate LAPACK header file.

`--with-lapack` The LAPACK library linker instructions.

`--with-int64` The integer index type (usually `long` or `long long`).

2.1 Installation with Reference BLAS (with large index support)

2.2 Installation with Intel MKL

The Intel® Math Kernel Library (MKL), available from <http://software.intel.com/en-us/intel-mkl/>, is a set of highly optimized and extensively threaded numeric routines for x86, x86-64, and Intel architecture (Itanium) processors. The library is available in standard and large-index versions.

We assume below that the MKL is installed in the `/opt/intel/mkl/10.2.5.035/` directory—substitute your installation path as required. We illustrate the command-line installation procedure below. Installation from the R console with the `install.packages` command follows similarly. We outline dynamic linkage against MKL which requires that the MKL libraries lie in the system library path.

2.2.1 Example: MKL Large-index support on x86-64 systems using GNU Open MP

```
R CMD INSTALL --configure-args='
  --with-incDir="-DMKL_ILP64 -I/opt/intel/mkl/10.2.5.035/include/"
  --with-blasHeader="refblas64longlong.h"
  --with-blas="-L/opt/intel/mkl/10.2.5.035/lib/em64t/
    -lmkl_gf_ilp64 -lmkl_gnu_thread -lgomp -lmkl_core"
  --with-lapack="-L/opt/intel/mkl/10.2.5.035/lib/em64t/
    -lmkl_gf_ilp64 -lmkl_gnu_thread -lgomp -lmkl_core"
  --with-lapackHeader="mkl.h"
  --with-int64="long long" ' bigalgebra
```

Note that we use the “`refblas64longlong.h`” BLAS header file supplied with the `bigalgebra` package. Additional MKL compilation examples can be found in the Intel MKL documentation.

2.3 Installation with AMD ACML

The AMD® Core Math Library (ACML) is available from <http://developer.amd.com/cpu/libraries/acml/page>. The ACML provides a highly-threaded set of numeric libraries optimized for performance on x86 and x86-64 processor architectures. The library is available in standard and large-index versions.

We assume that the ACML is installed in the `/opt/acml4.4.0/` directory.

2.3.1 Example: ACML Large-index support on x86-64 systems

```
R CMD INSTALL --configure-args='
    --with-incDir="-I/opt/acml4.4.0/gfortran64_int64/include/"
    --with-blasHeader="acml.h"
    --with-blas="-L/opt/acml4.4.0/gfortran64_int64/lib -lacml -lacml_mv"
    --with-lapack="-lacml -lacml_mv"
    --with-lapackHeader="acml.h"
    --with-int64=long' bigalgebra
```

2.4 Installation with GOTO BLAS

3 Examples

```
> library('bigalgebra')
XXX
```