Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Programming Techniques for Scientific Simulations Exercise 12

HS 12
Prof. M. Troyer

Problem 12.1 Matrix multiplication with libraries (no block assignment)

BLAS/LAPACK Installation In case you don't have these libraries on your system yet here are some hints:
If you work on your laptop, install BLAS and LAPACK (e.g. ATLAS, which you can obtain from `http://www.netlib.org/atlas/`). If you work on one of the computers in the exercise room, BLAS and LAPACK should already be installed. If you work with Mac OS X, you can either install ATLAS or use the pre-installed `vecLib` framework by linking with `-framework vecLib` instead of the standard BLAS/LAPACK linker options.

To use the BLAS routines, you have to link your program with some additional libraries (`g++ -lblas`). As the BLAS routines are FORTRAN routines, you might have to link your program also with some additional FORTRAN runtime libraries (flag `-lgfortran` may help).

Matrix-Matrix Multiplication Compare the time required by your implementation of the optimized matrix-matrix multiplication of Problem 10.2 with the same operation performed by an highly optimized library (i.e. BLAS).
To use it from C++, you need to declare DGEMM as follows:

```
extern "C" {
void dgemm_(const char& TRANSA , const char& TRANSB,
            const int& M, const int& N, const int& K,
            const double& alpha , double* A, const int& LDA,
            double* B, const int& LDB,
            const double& beta , double* C, const int& LDC);
}
```

This performs the following multiplication:

$$C = \alpha\, \mathrm{TRANSA}(A)\, \mathrm{TRANSB}(B) + \beta\, C, \tag{1}$$

where

- TRANSA and TRANSB can be **T**ranspose (indicated by 'T') or **N**oTranspose (indicated by 'N');
- doubles α and β are scalars;
- TRANSA(A) is a $M \times K$, TRANSB(B) a $K \times N$ and C a $M \times N$ matrix of doubles in column-major storage, which means that to access the element in row r and column c of matrix A, you need the element $A[r + M * c]$
- LDA, LDB, LDC parameters define the first dimension (=number of rows) of the corresponding matrix.

For a detailed description look at `http://www.netlib.org/blas/dgemm.f`.

Problem 12.2 Linear algebra with libraries (no block assignment)

Anharmonic oscillator In this exercise we will consider a quantum mechanical problem: we will calculate properties of the anharmonic oscillator. The quantum mechanical description is based on an eigenvalue problem (the stationary Schrödinger equation),

$$H|\Psi\rangle = E|\Psi\rangle \tag{2}$$

where

- $|\Psi\rangle \in \mathcal{H}$ is a vector in some Hilbert space $\mathcal{H}$. It is the *wave function* that describes the properties of a quantum mechanical state;
- H is the Hamilton operator which acts on vectors in $\mathcal{H}$;
- E is the corresponding energy eigenvalue.

Further explanations will be given in the exercise class. To solve this problem, we will set up the eigenvalue problem numerically and find the eigenvalues using a LAPACK routine. The Hamiltonian of the anharmonic oscillator is given by

$$\begin{aligned} H &= H_{\text{harmonic}} + H_{\text{anharmonic}} && (3)\\ &= \frac{p^2}{2m} + \frac{1}{2}m\omega^2x^2 + Kx^4, && (4) \end{aligned}$$

where x and p are operators that generally do not commute, $xp - px \neq 0$.
The harmonic part of this Hamiltonian can be written as

$$H_{\text{harmonic}} = \hbar\omega(a^\dagger a + \frac{1}{2}) \tag{5}$$

with the operators a and $a^\dagger$ defined by

$$a = \sqrt{\frac{m\omega}{2\hbar}}x + \frac{ip}{\sqrt{2m\hbar\omega}} \tag{6}$$

$$a^\dagger = \sqrt{\frac{m\omega}{2\hbar}}x - \frac{ip}{\sqrt{2m\hbar\omega}} \tag{7}$$

The eigenstates $|n\rangle$ of the count operator $N = a^\dagger a$ build a natural set of basis states for the harmonic oscillator. Their energy eigenvalues are given by $\langle n|H_{harm}|n\rangle = \hbar\omega(n + \frac{1}{2})$. Using the definitions of a and $a^\dagger$ one can write the anharmonic part of the hamiltonian as

$$Kx^4 = \frac{K\hbar^2}{4m^2\omega^2}(a + a^\dagger)^4 \tag{8}$$

Using the commutation relation $[a, a^\dagger] = 1$ one can obtain the nonzero matrix elements of H_{anharm}

$$\begin{aligned} \langle n+4|(a+a^\dagger)^4|n\rangle &= \sqrt{(n+1)(n+2)(n+3)(n+4)} && (9)\\ \langle n+2|(a+a^\dagger)^4|n\rangle &= (4n+6)\sqrt{(n+1)(n+2)} && (10)\\ \langle n|(a+a^\dagger)^4|n\rangle &= 3[n^2 + (n+1)^2] && (11) \end{aligned}$$

Numerical solution Store the matrix representation M_{ij} of $H = H_{harm} + H_{anharm}$ in a matrix and diagonalize it with LAPACK routine DSYEV (see the documentation in `http://www.netlib.org/lapack/double/dsyev.f`). Of course, you will need to choose a finite cutoff n_{MAX} in order to make the matrix finite. It is also probably a good idea to keep n_{MAX} relatively small – do you understand why?

To use the LAPACK routine, you have to link your program with some additional libraries (g++ -lblas -llapack). As the LAPACK routines are FORTRAN routines, you might have to link your program also with some additional FORTRAN runtime libraries (flag `-lgfortran` may help).