

FFTW on HPC

What is FFTW?

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). We believe that FFTW, which is free software, should become the FFT library of choice for most applications. [\[Source\]](#)

Links:

[Official Website](#)

[Documentation](#)

Versions Available:

The following versions are available on the cluster:

- fftw2
- fftw3

How to load FFTW?

To load FFTW, use the following commands:

```
#Load the FFTW module
module load fftw3/gcc5.4.0
# Load GCC compiler
module load compiler/gcc/11.1.0
```

To verify if the module and dependencies are loaded correctly, use the following command.

```
#Show all the modules loaded
module list
```

This should list all the FFTW and dependencies that are loaded. In this case, it is fftw3 and gcc.

How to use FFTW?

To demonstrate the usage of this module, use the following C code to do a Discrete Fourier transformation on real numbers generated randomly,

```
#include <complex.h>
#include <fftw3.h>
#include <stdlib.h>
#include <time.h>

const double PI = 3.1415926535;

int main(int argc, char** argv){

    // Check command line argc
    if (argc != 2){
        printf("Not enough args");
        exit(-1);
    }

    // Var decleration
    int N = atoi(argv[1]);
    double *in;
    fftw_complex *out;
    fftw_plan p;
```

```

// Set seed
srand(time(NULL));

// Allocate input
in = (double*) malloc(sizeof(double) * N);
printf("FFT input: ");
for (int i=0 ; i < N ; i++){
in[i] = (rand() % 200) / PI;
printf("%.2lf " , in[i]);
}
printf("\n\n");

// Allocate out
out = (fftw_complex*) fftw_malloc(sizeof(fftw_complex) * N);

// Generate a plan
p = fftw_plan_dft_r2c_1d(N, in, out,FFTW_PRESERVE_INPUT);

// Execute a plan
fftw_execute(p);

// Print output
printf("FFT output: ");
for (int i=0 ; i < N ; i++){
printf("%.2lf ",out[i]);
}
printf("\n");

// Free allocated mem
fftw_destroy_plan(p);
fftw_free(in); free(out);

return 0;
}

```

The headers includes the fftw3 library which allows an interface with FFTW library. For understanding the code, please refer to the documentation of the fftw3 library.

To compile it, users must link the library while compiling,

```
# Compile the program
# Users must load gcc 5.4.0 or above
gcc test.c -lfftw3 -lm
```

Execute the compiled program, the output should be something like this,

```
# Execute the sub routine
./a.out 12
```

The output should be something like this,

```
# Output of FFT transform
FFT input: 54.11 10.50 63.34 40.74 9.23 24.83 11.14 7.64 63.34 51.88
45.84 9.55
FFT output: 392.16 50.52 -91.99 6.37 40.74 72.03 101.86 0.00 0.00 0.00
0.00 0.00
```

Where to find help?

If you are stuck on some part or need help at any point, please contact OIT at the following address.

<https://ua-app01.ua.edu/researchComputingPortal/public/oitHelp>

