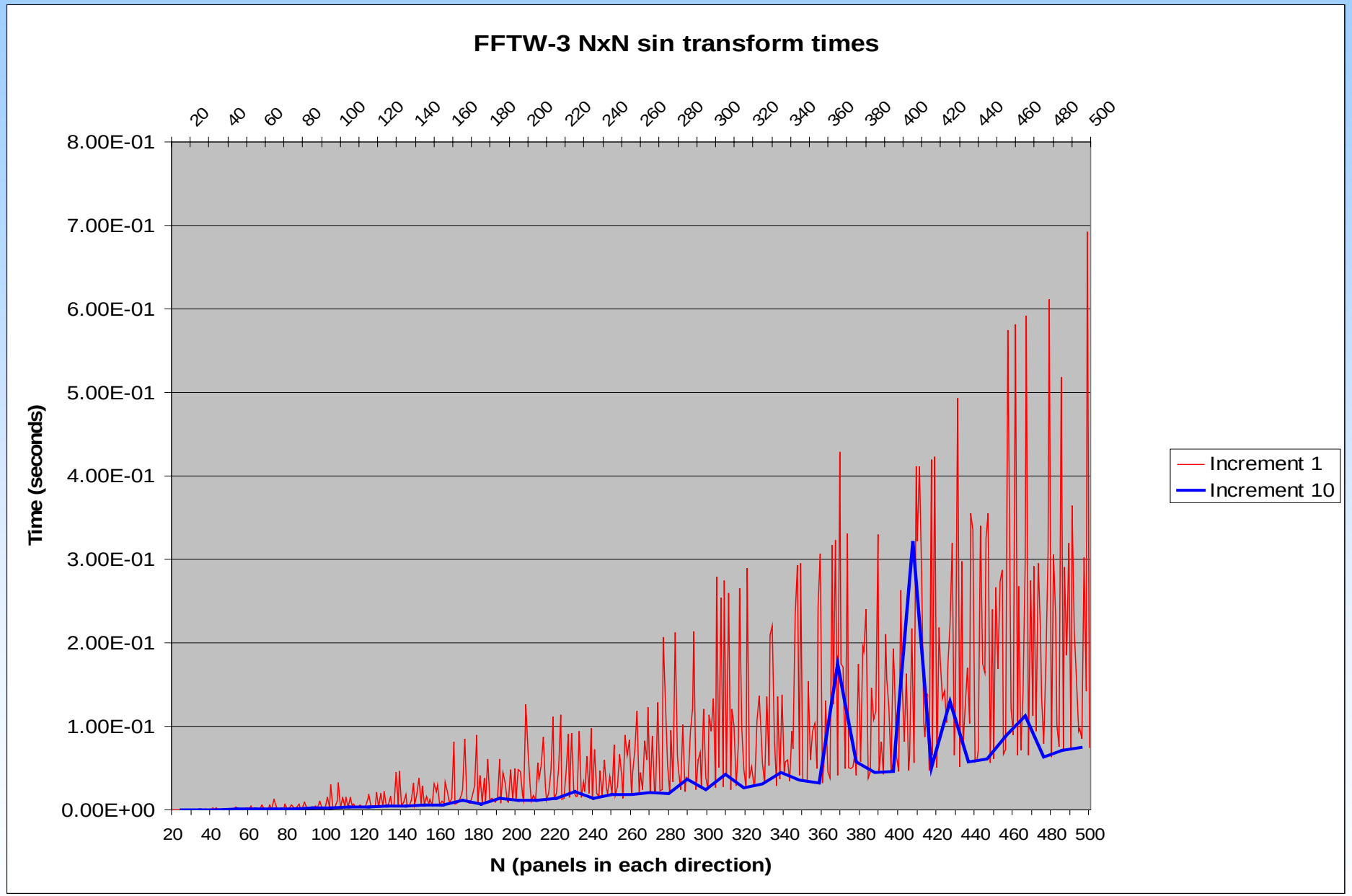
Math 270E: FFTW3 - Usage Information

- Performance dependence on N (panel count)
- FFTWinterface classes
- Extensions

October 27, 2008

FFTW3 - 2D dependence on N (panel count)



Recommendation:

Use a power of 2 or multiple of 10 for the panel count

Avoid 10 x prime --- 370, 410 ,430, 470, etc.

FFTWinterface classes

Desired computation ⇄ **FFTWinterface classes**

- fftw3_sin1d
- fftw3_sin2d

⇄ FFTW-3 routines

$$g_i = \sum_{k=1}^{nx-1} \hat{g}_k \sqrt{2} \sin(k\pi x_i)$$

$$\hat{g}_k = \sum_{i=1}^{nx-1} g_i \sqrt{2} \sin(k\pi x_i) \left(\frac{1}{nx}\right)$$

$$Y_k = 2 \sum_{j=0}^{n-1} X_j \sin(\pi(j+1)\left(\frac{k+1}{n+1}\right))$$

$$Y_k = 2 \sum_{j=0}^{n-1} X_j \sin(\pi(j+1)\left(\frac{k+1}{n+1}\right))$$

What we want/need is different from what FFTW provides.

FFTWinterface classes

Differences to be accommodated :

- scaling/normalization
- ordering of the coefficients
- input array size and structure

Ordering of coefficients

FFTW3/signal processing convention: *only positive k*

4.7.1 The 1d Discrete Fourier Transform (DFT)

The forward (`FFTW_FORWARD`) discrete Fourier transform (DFT) of a 1d complex array X of size n computes an array Y, where:

$$Y_k = \sum_{j=0}^{n-1} X_j e^{-2\pi jk\sqrt{-1}/n} .$$

The backward (`FFTW_BACKWARD`) DFT computes:

$$Y_k = \sum_{j=0}^{n-1} X_j e^{2\pi jk\sqrt{-1}/n} .$$

Math/Numerical solution of differential equations:
positive and negative k

$$\hat{g}_k = \sum_{i=0}^{nx-1} g_i \, e^{2\pi \sqrt{-1}\, k\, x_i} \frac{1}{nx}$$

$$g_i = \sum_{k=-[nx/2]}^{[\frac{nx-1}{2}]} \hat{g}_k \, e^{2\pi \sqrt{-1}\, k\, x_i}$$

A very important difference!!!!

Recommendations

- Read the manual.

- Design internal “consistency checks” and write test codes to verify desired computations are being performed correctly.

FFTWinterface Extensions

- Transforms for other boundary conditions
- Code to take advantage of additional FFTW optimizations (e.g. FFTW `wisdom’)