

The fast Fourier transform applied to estimate wave energy spectral density in random sea state

M. Rahman¹, D. Riordan¹, A. Susilo² & S. H. Mousavizadegan³

¹*Faculty of Computer Science, Dalhousie University, Halifax, Canada*

²*Steubenring 8, 53175 Bonn, Germany*

³*Faculty of Marine Tech., Amirkabir University of Tech., Tehran, Iran*

Abstract

This paper deals with an important area of real life problems. It is concerned with the application of the fast Fourier transform to estimate wave energy spectral density in a random sea state. Graphical illustrations are manifested in a clear and distinct manner. The fast Fourier transform is a mathematical procedure which can be thought of as transforming a function from the time domain to the frequency domain. The application of a Fourier transform is analogous to the splitting up of a light beam by a prism to form the optical *spectrum* of the light source. An optical spectrum consists of lines or beams of colors corresponding to the various wavelengths and hence different frequencies of light wave emitted by the source. The spectrum of a signal in digital signal processing refers to the way energy in the signal is distributed over its various frequency components.

Keywords: Discrete Fourier transforms, fast Fourier transforms, spectrum, frequency, wavelength, energy density, random sea state.

1 Introduction

This paper brings some ideas and concepts in the way the Fourier transform plays a vital role in our daily life. We investigate here from theoretical and mathematical view points an important topic called *the fast Fourier transform* (FFT). The fast Fourier transform is a mathematical algorithm which is an extremely important and widely used method that is used to extract information from sampled signals. In



this twenty first century, electronics dominate our life style tremendously. We can say that the Fourier transform is a mathematical procedure which can be thought of as transforming a function from the time domain to the frequency domain. The application of the Fourier transform to a signal is analogous to the splitting up of a light beam by a prism to form the optical *spectrum* of the light source. An optical spectrum consists of lines or bands of color corresponding to the various wavelengths and hence different frequencies of light wave emitted by the source. The spectrum of a signal in a digital signal processing refers to the way energy in the signal is distributed over its various frequency components.

Digital signal processing involves *discrete* signals that means signals which are sampled at regular intervals of time rather than continuous signals. A modified form of the Fourier transform, known as the *discrete Fourier transform* or DFT, is used in the case of sampled (discrete) signals. To compute the DFT of a signal comprising 1000 samples, say, would entail of the order of one million (1000^2) calculations. The DFT is therefore an extremely numerically intensive procedure. With this procedure we get extremely accurate information about the frequency components of a signal with the huge computational effort. With the development of the digital computer it is not a problem to perform numerical calculation rapidly and accurately. These calculations were performed by using DFT procedure until the 1960s when Cooley and Tukey [3], discovered a numerical algorithm which allows the DFT to be evaluated with a significant reduction in the amount of calculation required. This algorithm, called the fast Fourier transform, or FFT, allows the discrete Fourier transform of a sampled signal to be obtained rapidly and efficiently. Nowadays, the FFT is used in many areas of applied problems because of its rapidity, accuracy and efficiency. Thus the FFT is nothing but the DFT which is very popular among scientists and engineers because of the reason cited above.

2 Some preliminaries leading to the fast Fourier transforms

The formulas for the Fourier transform and its inverse are given by (see [5,6])

$$\mathcal{F}\{f(x)\} = \int_{-\infty}^{\infty} f(x)e^{-2\pi ixy} dx = g(y) \quad (1)$$

$$f(x) = \int_{-\infty}^{\infty} g(y)e^{2\pi ixy} dy = \mathcal{F}^{-1}\{g(y)\} \quad (2)$$

where x is treated as the time variable and y the frequency. We use these symbols to make them universally valid because the Fourier transform can be regarded as *ubiquitous*. The Fourier transform has long been a principle analytical tool in such diverse fields as linear system, probability theory, quantum physics, antennas, distribution theory and signal processing. We know that the impulse function $\delta(x)$ is defined as

$$\int_{-\infty}^{\infty} \delta(x - x_0)f(x)dx = f(x_0) \quad (3)$$



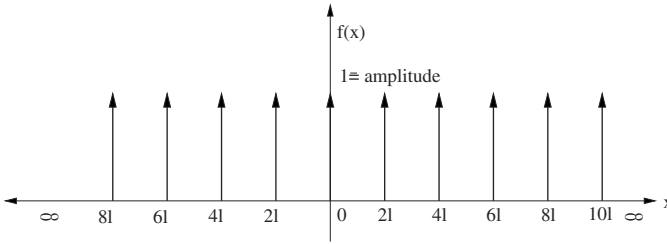


Figure 1: A sequence of periodic delta functions $f(x) = \sum_{n=-\infty}^{\infty} \delta(x - 2n\ell)$.

where $f(x)$ is an arbitrary function continuous at $x = x_0$. Application of the definition (3) yields Fourier transform of many important functions very easily. A ‘unitary function’ $U(x)$ can be found, which is a good function vanishing for $|x| \geq 1$ and such that (see [5])

$$\sum_{n=-\infty}^{\infty} U(x + n) = 1 \tag{4}$$

for all x . The Fourier transform $V(y)$ of any such function has $V(0) = 1$, but $V(m) = 0$ if m is an integer other than zero. The Fourier transform of a sequence of equal distant impulse functions $\delta(x)$ is another sequence of equal distant impulses: Mathematically, if 2ℓ is the period of the sequence of functions, then we need to prove that

$$f(x) = \sum_{n=-\infty}^{\infty} \delta(x - 2\ell n)$$

$$\mathcal{F}\{f(x)\} = g(y) = \frac{1}{2\ell} \sum_{n=-\infty}^{\infty} \delta\left(y - \frac{n}{2\ell}\right).$$

The Fourier series can be written as [7]

$$f(x) = \frac{1}{2\ell} \sum_{n=-\infty}^{\infty} e^{in\pi x/\ell} = \sum_{n=-\infty}^{\infty} \delta(x - 2n\ell) = \frac{1}{2\ell} + \frac{1}{\ell} \sum_{n=-\infty}^{\infty} \cos\left(\frac{n\pi x}{\ell}\right).$$

Thus through this investigation we have found an important relationship that

$$2\ell \sum_{n=-\infty}^{\infty} \delta(x - 2n\ell) = 1 + 2 \sum_{n=-\infty}^{\infty} \cos\left(\frac{n\pi x}{\ell}\right).$$

Figure 1 depicts the periodic impulse of the delta function whereas Figure 2 depicts its transform as a row of periodic delta functions.

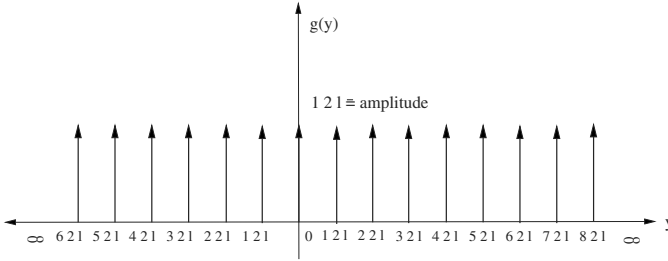


Figure 2: A sequence of the Fourier transform of periodic impulse functions $g(y) = \frac{1}{2l} \sum_{n=-\infty}^{\infty} \delta(y - \frac{n}{2l})$.

3 The discrete Fourier transform

We develop the discrete function as follows. When $f(x)$, a continuous function, is sampled at regular interval of period $2l$ the usual Fourier transform technique is modified. A diagrammatic form of a simple sample together with its associated input-output waveforms is depicted in the following Figure 3.

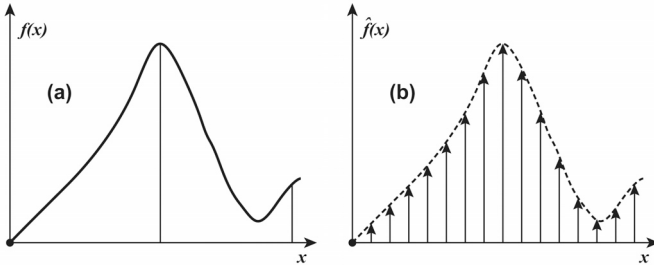


Figure 3: (a) A sketch of the continuous function $f(x)$; (b) A sketch of the discrete function $\hat{f}(x)$.

Let us consider the sampling frequency to be $y_s = \frac{1}{2l}$ which is the cycle per second and is called Hertz. Here $f(x)$ is the continuous function and $\hat{f}(x)$ is the discrete version of $f(x)$. Defining the set of impulse functions $\hat{\delta}(x)$ by

$$\hat{\delta}(x) = \sum_{n=-\infty}^{\infty} \delta(x - 2nl),$$

the input-output relationship of the sample becomes

$$\begin{aligned} \hat{f}(x) &= f(x)\hat{\delta}(x) \\ &= \sum_{n=-\infty}^{\infty} f(x)\delta(x - 2n\ell) \\ &= \sum_{n=-\infty}^{\infty} f(2n\ell)\delta(x - 2n\ell) \end{aligned}$$

where for a given $f(x)$ and (2ℓ) , the $\hat{f}(x)$ is unique, but the converse is not true. The above infinite series can be truncated to finite number of terms if we use the following rectangular function of amplitude unity.

$$u(x) = \begin{cases} 1 & -\ell < x < \lambda - \ell \\ 0 & \text{otherwise} \end{cases}$$

where λ is the total length of time such that there are N samples each of length 2ℓ and hence $2\ell = \frac{\lambda}{N}$. Thus the above series can be rewritten as follows:

$$\hat{f}(x)u(x) = \sum_{n=0}^{N-1} f(2n\ell)\delta(x - 2n\ell),$$

where it has been assumed that there are N equidistant impulse functions lying within the truncation interval, i.e., $N = \frac{\lambda}{2\ell}$. The sampled truncation waveform and its Fourier transform are suitable because truncation in time domain results in *rippling* in the frequency domain.

To get our original transform pair from a discrete Fourier transform pair, we need to modify to sample the Fourier transform of the above equation. In the time domain this product is equivalent to convolving the sample truncated waveform of the above and the time function

$$\hat{\delta}_1(x) = \lambda \sum_{r=-\infty}^{\infty} \delta(x - r\lambda) = \sum_{r=-\infty}^{\infty} e^{2\pi irx/\lambda}.$$

Because $\hat{\delta}_1(x)$ function is periodic with period λ .

Referring to Brigham [1], the desired relationship can be written as follows:

$$\begin{aligned} \tilde{f}(x) &= f(x)\hat{\delta}(x)u(x) * \hat{\delta}_1(x) \\ &= \lambda \sum_{r=-\infty}^{\infty} \left\{ \sum_{n=0}^{N-1} f(2n\ell)\delta(x - 2n\ell - r\lambda) \right\} \end{aligned} \tag{5}$$

where $\tilde{f}(x)$ is the approximation to the function $f(x)$. This function is a periodic function with period λ and hence we can expand it as a Fourier series expansion and this series is given as follows:



$$\tilde{f}(x) = \sum_{n=-\infty}^{\infty} C_n e^{2\pi i n x / \lambda} \tag{6}$$

where C_n is the Fourier coefficient and actually it is defined as the discrete Fourier transform. Thus the problem reduces to determine this Fourier coefficient and our goal will be achieved. Using the definition to find the Fourier coefficient C_n yields

$$\begin{aligned} C_n &= \frac{1}{\lambda} \int_{-\ell}^{\lambda-\ell} \tilde{f} x e^{-2\pi i n x / \lambda} dx \\ &= \frac{1}{\lambda} \int_{-\ell}^{\lambda-\ell} \lambda \sum_{r=-\infty}^{\infty} \left\{ \sum_{k=0}^{N-1} f(2k\ell) \delta(x - 2k\ell - r\lambda) \right\} e^{-2\pi i n x / \lambda} dx \\ &= \int_{-\ell}^{\lambda-\ell} \left\{ \sum_{k=0}^{N-1} f(2k\ell) \delta(x - 2k\ell) \right\} e^{-2\pi i n x / \lambda} dx \\ &= \sum_{k=0}^{N-1} f(2k\ell) e^{-2\pi i n k / N} \quad n = 0, \pm 1, \pm 2, \dots \end{aligned} \tag{7}$$

In evaluating this integral we have used the fact that the integration is done only over one period putting $r = 0$, and that $2\ell = \frac{\lambda}{N}$.

Thus we can write the Fourier transform of the approximate periodic function $\tilde{f}(x)$ as

$$\tilde{g}(y) = \sum_{k=0}^{N-1} f(2k\ell) e^{-2\pi i n k / N} \quad n = 0, 1, 2, \dots, N - 1. \tag{8}$$

If we define $y = n\Delta y$ such that $\Delta y(2\ell) = \frac{1}{N}$, then the discrete Fourier transform will take the following familiar form

$$\tilde{g}\left(\frac{n}{2\ell N}\right) = \sum_{k=0}^{N-1} f(2k\ell) e^{-2\pi i n k / N} \quad n = 0, 1, 2, \dots, N - 1. \tag{9}$$

Equation (9) is the desired discrete Fourier transform ; the expression relates N samples of time and N samples of frequency by means of the continuous Fourier transform. The discrete Fourier transform is then a special case of the continuous Fourier transform.

A graphical development of the discrete Fourier transform pairs has been depicted in Figures 4 and 5. Figure 4 is sketched using analytical treatment of the continuous Fourier transform whereas Figure 5 is depicted due to the discrete Fourier transform pair just developed. The discrete Fourier transform pair of Figure 5 is acceptable for the purpose of the digital machine computation since both the time and frequency domain are repeated by discrete values. In this figure, the original time function Figure 4 is approximated by N samples; the original



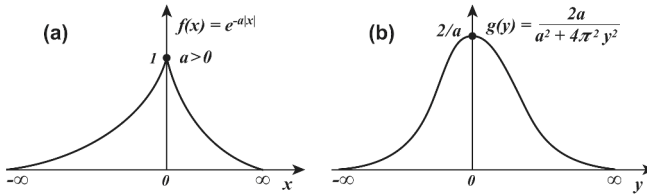


Figure 4: (a) The graph of a continuous function $f(x)$; (b) The graph of the Fourier transform of the continuous function $f(x)$, i.e., $g(y) = \frac{2a}{a^2 + 4\pi^2 y^2}$.

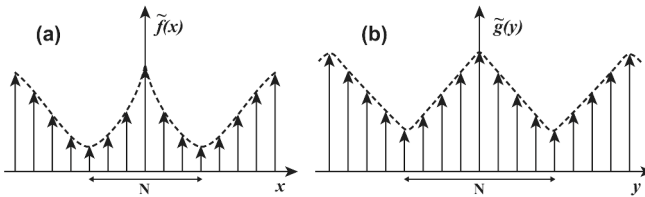


Figure 5: Graphical development of the discrete Fourier transform [1].

Fourier transform $g(y)$ is also approximated by N samples. These N samples define the discrete Fourier transform pair and approximate the original Fourier transform pair. It is easily noted from these figures that the sampling of the time and the frequency functions display that they are periodic functions which should be the case because of discrete behavior of the functions. Thus this algorithm seems suitable for the numerical computation. It is to be noted that these forms are predicted by earlier workers in this important field of research.

4 Mathematical aspects of FFT

In this section we shall briefly describe the mathematical development leading to the discrete Fourier transform (DFT) and then to the FFT. To do this we need to bring the Fourier series pair and the Fourier transform pair which are already developed in the previous chapters for ready reference. The Fourier series pair and the Fourier transform pair are respectively given by the following equations. The Fourier series pair in complex variable form is:

$$f(x) = \sum_{n=-\infty}^{\infty} C_n e^{in\pi x/\ell}$$

$$C_n = \frac{1}{2\ell} \int_{-\ell}^{\ell} f(x) e^{-in\pi x/\ell} dx. \tag{10}$$

The Fourier transforms can be defined in two ways. The first definition is:

$$\begin{aligned}\mathcal{F}\{f(x)\} &= \int_{-\infty}^{\infty} f(x)e^{-2\pi ixy} dx = g(y) \\ f(x) &= \int_{-\infty}^{\infty} g(y)e^{2\pi ixy} dy = \mathcal{F}^{-1}\{g(y)\}.\end{aligned}\quad (11)$$

The second definition (conventional) is:

$$\begin{aligned}\mathcal{F}\{f(x)\} &= \int_{-\infty}^{\infty} f(x)e^{-ikx} dx = g(k) \\ f(x) &= \frac{1}{2\pi} \int_{-\infty}^{\infty} g(k)e^{ikx} dk = \mathcal{F}^{-1}\{g(k)\}.\end{aligned}\quad (12)$$

The first equation in equations (11) is the Fourier transform and the second equation is its inverse. Similarly, the first equation (12) is the Fourier transform and the second equation is its inverse. They are equivalent. It is obvious that $2\pi y = k$ such that $dy = \frac{1}{2\pi} dk$. It is to be noted here that the fast Fourier transform (FFT) is a discrete Fourier transform algorithm which reduces the number of computations needed for N points from N^2 to $N \log_2 N$, where \log_2 is the base-2 logarithm. FFTs were first discussed by Cooley and Tukey [3], although Gauss [4] had actually described the critical factorization step as early as 1805. The DFT version of Fourier transforms can be derived by discretizing the equation (11) in the following manner. In the signal processing literature, it is usual practice to write the DFT and its inverse in the more pure form as given below

$$g(p) = \sum_{n=0}^{N-1} f(n\Delta t) \exp(-i2\pi pn\Delta t/N), \quad p = 0, 1, 2, \dots, N-1 \quad (13)$$

$$f(n\Delta t) = \sum_{p=0}^{N-1} g(p) \exp(i2\pi pn\Delta t/N), \quad n = 0, 1, 2, \dots, N-1 \quad (14)$$

where $f(n\Delta t)$ denotes the input signal at time (sample) $n\Delta t$, and $g(p)$ denotes the p th spectral sample. This form is the simplest mathematically.

5 The fast Fourier transform algorithm

We have already developed the relationship between the continuous Fourier transform and the discrete Fourier transform in the last section. We illustrate the application of the discrete Fourier transform **DFT** and then how this transform plays a vital role in the development of the fast Fourier transform (**FFT**). One does not need any special expertise to formulate the FFT algorithm. The FFT is simply an algorithm. It is just a particular method of performing a series of machine calculation which can compute the discrete Fourier transform much more



rapidly than other available algorithms. We will discuss in this section very briefly the computational aspect of the algorithm.

With reference to the equation (9), we rewrite the equation in a simple form so that a non-expert in this field can very easily understand the algorithm. To simplify the matter, we replace $2\ell k$ by k and $\frac{n}{2\ell N}$ by n so that our new equation will look like as follows:

$$g(n) = \sum_{k=0}^{N-1} f(k)e^{-2\pi ink/N}, \quad n = 0, 1, 2, 3, \dots, N-1. \quad (15)$$

This equation implies that it is a $N \times N$ matrix equation that means it is an algebraic equation of n unknowns. Let us define $e^{-2\pi i/N} = \omega$ so that the equation (15) can be rewritten in compact form as

$$\mathbf{g}(\mathbf{n}) = \omega^{\mathbf{nk}}\mathbf{f}(\mathbf{k}). \quad (16)$$

This is a matrix equation displayed in bold faces. It is to be noted that this equation is a complex equation because ω and $\mathbf{f}(\mathbf{k})$ complex and so there are N^2 complex multiplication and $N(N-1)$ complex additions that are needed to perform the required matrix computation. The FFT owes its success to the fact that the algorithm reduces the number of multiplications and additions required in computation of (16). Thus we can see that FFT is nothing but the DFT. Only difference is that FFT works much faster than DFT in terms of computer time and efficiency. For $N = 2^m$ the FFT algorithm is then simply a procedure for factoring an $N \times N$ matrix into m matrices (each $N \times N$) such that each of these factored matrices has the special property of minimizing the number of complex multiplications and additions. If we consider $N = 4 = 2^2$, such that $m = 2$, we note that the FFT requires $N \times m/2 = 4$ complex multiplications and $N \times m = 8$ complex addition, whereas the direct method requires $N^2 = 16$ complex multiplications and $N(N-1) = 4 \times 3 = 12$ complex additions. If we assume that the computing time is proportional to the number of multiplications, the approximate ratio of the direct to FFT computing time is given by

$$\frac{N^2}{N \times m/2} = \frac{2N}{m} = 4.$$

If $N = 1024 = 2^{10} = 2^m$, then $m = 10$ and the ratio = $\frac{2N}{m} = \frac{2 \times 1024}{10} = 204.8$. That means the computational reduction of more than 200 to 1. This is a fantastic saving of time by the FFT algorithm.

The fast Fourier transform (FFT) is computational algorithm which reduces the computational time of the discrete Fourier transform to a time proportional to $N \log_2 N$. This increase in computing speed has completely revolutionized many facets of scientific analysis. A historical review of the discovery of the FFT illustrates that this important development was almost ignored. As for example if $N = 1024$, then DFT computing time will be $N^2 = 1024^2 = 1048576$ whereas the FFT computing time will be $N \log_2 N = N \left\{ \frac{\ln N}{\ln 2} \right\} = 1024 \times \frac{\ln 1024}{\ln 2} =$



$1024 \times 10 = 10240$. This is really a fantastic saving in computational time by FFT algorithm.

6 Cooley–Tukey algorithms

So far we have found that the most common FFT is the Cooley–Tukey algorithm. This is a divide and conquer algorithm that recursively breaks down a DFT of any composite size $N = N_1 N_2$ into many smaller DFTs of sizes N_1 and N_2 , along with $O(N)$ multiplications by complex roots of unity traditionally called twiddle factors. The most well-known use of the Cooley–Tukey algorithm is to divide the transform into two pieces of size $N/2$ at each step, and is therefore limited to power- of - two sizes, but any factorization can be used in general (as was known to both Gauss and Cooley and Tukey). These are called the **radix-2** and **mixed-radix** cases, respectively. Although the basic idea is recursive, most traditional implementations rearrange the algorithm to avoid explicit recursion. Since the Cooley–Tukey algorithm breaks the DFT into smaller DFTs, it can be combined arbitrarily with any other algorithm for the DFT.

7 An application of FFT to wave energy spectral density

We shall not go deep into the algorithm of the FFT rather in the following we will demonstrate an application of wave energy spectral density as described by Chakrabarti [6] (see also [2]). In random sea state on a short term basis maintains certain identifiable statistical properties and is best represented by its energy density spectrum. The total energy of a wave E (per unit surface area) in the wave record between infinite time limits is given by the integral

$$E = \frac{1}{2} \rho g \int_{-\infty}^{\infty} |\eta(t)|^2 dt \quad (17)$$

where $\eta(t)$ is the wave elevation, ρ is the density and g is the acceleration due to gravity. Chakrabarti [2] has defined that the energy spectral density by FFT can be obtained as

$$S(\omega) = \frac{1}{T_s} \left| \sum_{n=1}^N \eta(n\Delta t) \exp(i2\pi f(n\Delta t)) \Delta t \right|^2 \quad (18)$$

Usually in the FFT calculation, the total data length, T_s , is divided into a number of smaller segments, M , each one having an equal number of data points, N at a constant time increment, Δt . The final result then is averaged over the M sections. The advantage of this method has already mentioned above. We now discuss what parameters are involved in the computation by FFT. The variables that have to be selected before an energy spectrum of a wave record can be obtained by the FFT algorithm are: Number of sections, M ; Number of data points in each section, N (a power of 2); Time increment or sampling rate, Δt ; Frequency increment



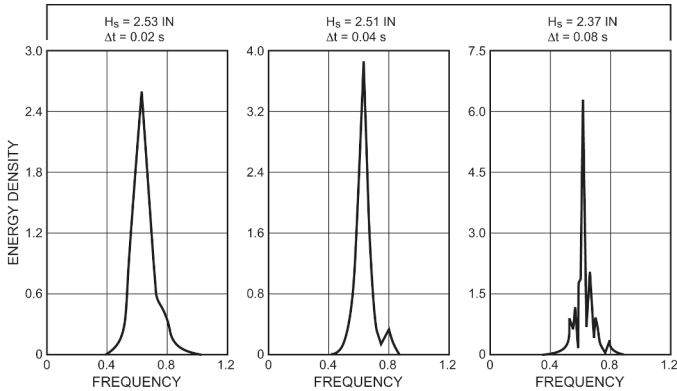


Figure 6: Variation of energy density spectral shape of wave record (fixed T_s) with variation of N (or alternatively, Δt for a fixed M). From Chakrabarti, Computational Mechanics Publications, Southampton Boston (1987).

or resolution, Δf ; and Frequency range, or so-called Nyquist frequency, f_N . The first three of these quantities have to be independently selected. The length of the record, T_s is dependent on M , N , and Δt , i.e., $T_s = MN\Delta t$. For a given record, T_s and Δt are fixed, so that the total number of data points can be obtained from these values. Thus, the only choice that has to be made is the number of sections, M . If we know the first three variables, the last two can be calculated as follows:

$$\Delta f = \frac{1}{N(\Delta t)} \quad (19)$$

$$f_N = \frac{1}{2(\Delta t)}. \quad (20)$$

Note that the length of wave record is always finite. This requires limiting the Fourier transform in the evaluation of the energy spectrum to a finite Fourier transform. An example of the effect of varying N is shown in Fig. 6. Note that the energy density spectrum is composed of finite number of frequencies and higher values of N reveals the individual peaks and reduce the confidence in the ordinary values. The values of H_s (significant wave height), however, is relatively unchanged. In particular, the value of M is usually taken as $M \geq 8$, while the value of N is normally between 512 and 2048.

Acknowledgements

We are thankful to Professor Carlos Brebbia, Director of Wessex Institute of Technology and Professor Michael Shepherd, Dean of Faculty of Computer Science, Dalhousie University for their kind interest in the contents of the paper.



References

- [1] Brigham, E. O., *The fast Fourier transform*, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1974.
- [2] Chakrabarti, S. K., *Hydrodynamics of offshore structures*, Computational Mechanics Publications, Southampton Boston, 1987.
- [3] Cooley, James W. and Tukey, John W., An algorithm for the machine calculation of complex Fourier series, *Math. Comput.*, **19**, 297-301, 1965.
- [4] Gauss, C. F., Nachlass: Theoria interpolationis methodo nova tractata, *Werke band*, **3**, 265-327, 1866. Göttingen: Königliche Gesellschaft der Wissenschaften.
- [5] Lighthill, M. J., *Introduction to Fourier analysis and generalized functions*, Cambridge University Press, Cambridge, 1964.
- [6] Rahman, M., *Water waves: Relating modern theory to advanced engineering applications*, Oxford University Press, Oxford, 1995.
- [7] Rahman, M., *Applications of Fourier transforms to generalized functions*, WIT Press, Southampton Boston, 2011.

