INTRODUCTION

In this chapter we describe briefly and tutorially the concept of transform analysis. The Fourier transform is related to this basic concept and is examined with respect to its basic analysis properties. A survey of the scientific fields which utilize the Fourier transform as a principal analysis tool is included. The requirement for discrete Fourier transforms and the historical development of the fast Fourier transform (FFT) are presented.

1-1 TRANSFORM ANALYSIS

Every reader has at one time or another used transform analysis techniques to simplify a problem solution.

The reader may question the validity of this statement because the term transform is not a familiar analysis description. However, recall that the logarithm is in fact a transform which we have all used.

To more clearly relate the logarithm to transform analysis consider Fig. 1-1. We show a flow diagram which demonstrates the general relationship between conventional and transform analysis procedures. In addition, we illustrate on the diagram a simplified transform example, the logarithm transform. We will use this example as a mechanism for solidifying the meaning of the term transform analysis.

From Fig. 1-1 the example problem is to determine the quotient \( Y = X/Z \). Let us assume that extremely good accuracy is desired and a computer is not available. Conventional analysis implies that we must determine \( Y \) by long-hand division. If we must perform the computation of \( Y \) repeatedly,
then conventional analysis (long-hand division) represents a time-consuming process.

The right-hand side of Fig. 1-1 illustrates the basic steps of transform analysis. As shown, the first step is to convert or transform the problem statement. For the example problem, we choose the logarithm to transform division to a subtraction operation.

Because of this simplification, transform analysis then requires only a table look-up of \( \log(X) \) and \( \log(Z) \), and a subtraction operation to determine \( \log(Y) \). From Fig. 1-1, we next find the inverse transform (anti-logarithm) of \( \log(Y) \) by table look-up and complete the problem solution. We note that
by using transform analysis techniques we have reduced the complexity of the example problem.

In general, transforms often result in simplified problem solving analysis. One such transform analysis technique is the Fourier transform. This transform has been found to be especially useful for problem simplification in many fields of scientific endeavor. The Fourier transform is of fundamental concern in this book.

1-2 BASIC FOURIER TRANSFORM ANALYSIS

The logarithm transform considered previously is easily understood because of its single dimensionality; that is, the logarithm function transforms a single value $X$ into the single value $\log(X)$. The Fourier transform is not as easily interpreted because we must now consider functions defined from $-\infty$ to $+\infty$. Hence, contrasted to the logarithm function we must now transform a function of a variable defined from $-\infty$ to $+\infty$ to the function of another variable also defined from $-\infty$ to $+\infty$.

A straightforward interpretation of the Fourier transform is illustrated in Fig. 1-2. As shown, the essence of the Fourier transform of a waveform is to decompose or separate the waveform into a sum of sinusoids of different frequencies. If these sinusoids sum to the original waveform then we have determined the Fourier transform of the waveform. The pictorial representation of the Fourier transform is a diagram which displays the amplitude and frequency of each of the determined sinusoids.

Figure 1-2 also illustrates an example of the Fourier transform of a simple waveform. The Fourier transform of the example waveform is the two sinusoids which add to yield the waveform. As shown, the Fourier transform diagram displays both the amplitude and frequency of each of the sinusoids. We have followed the usual convention and displayed both positive and negative frequency sinusoids for each frequency; the amplitude has been halved accordingly. The Fourier transform then decomposes the example waveform into its two individual sinusoidal components.

The Fourier transform identifies or distinguishes the different frequency sinusoids (and their respective amplitudes) which combine to form an arbitrary waveform. Mathematically, this relationship is stated as

$$S(f) = \int_{-\infty}^{\infty} s(t)e^{-j2\pi ft} dt \quad (1-1)$$

where $s(t)$ is the waveform to be decomposed into a sum of sinusoids, $S(f)$ is the Fourier transform of $s(t)$, and $j = \sqrt{-1}$. An example of the Fourier transform of a square wave function is illustrated in Fig. 1-3(a). An intuitive justification that a square waveform can be decomposed into the set of sinusoids determined by the Fourier transform is shown in Fig. 1-3(b).
Figure 1-2. Interpretation of the Fourier transform.
Figure 1-3. Fourier transform of a square wave function.
We normally associate the analysis of periodic functions such as a square wave with Fourier series rather than Fourier transforms. However, as we will show in Chapter 5, the Fourier series is a special case of the Fourier transform.

If the waveform $s(t)$ is not periodic then the Fourier transform will be a continuous function of frequency; that is, $s(t)$ is represented by the summation of sinusoids of all frequencies. For illustration, consider the pulse waveform and its Fourier transform as shown in Fig. 1-4. In this example

the Fourier transform indicates that one sinusoid frequency becomes indistinguishable from the next and, as a result, all frequencies must be considered.

The Fourier transform is then a frequency domain representation of a function. As illustrated in both Figs. 1-3(a) and 1-4, the Fourier transform frequency domain contains exactly the same information as that of the original function; they differ only in the manner of presentation of the information. Fourier analysis allows one to examine a function from another point of view, the transform domain. As we will see in the discussions to
follow, the method of Fourier transform analysis, employed as illustrated in
Fig. 1-1, is often the key to problem solving success.

1.3 THE UBIQUITOUS FOURIER TRANSFORM

The term ubiquitous means to be everywhere at the same time. Because
of the great variety of seemingly unrelated topics which can be effectivly
dealt with using the Fourier transform, the modifier ubiquitous is certainly
appropriate. One can easily carry over the Fourier analysis techniques de vel oped
in one field to many diverse areas. Typical application areas of the Fourier
transform include:

*Linear Systems.* The Fourier transform of the output of a linear system
is given by the product of the system transfer function and the Fourier
transform of the input signal [1].

*Antennas.* The field pattern of an antenna is given by the Fourier trans-
form of the antenna current illumination [2].

*Optics.* Optical systems have the property that a Fourier transform rela-
tion exists between the light amplitude distribution at the front and back
focal planes of a converging lens [3].

*Random Process.* The power density spectrum of a random process is
given by the Fourier transform of the auto-correlation function of the pro-
cess [4].

*Probability.* The characteristic function of a random variable is defined
as the Fourier transform of the probability density function of the random
variable [5].

*Quantum Physics.* The uncertainty principle in quantum theory is funda-
mentally associated with the Fourier transform since particle momentum
and position are essentially related through the Fourier transform [6].

*Boundary-Value Problems.* The solution of partial differential equations
can be obtained by means of the Fourier transform [7].

Although these application areas are extremely diverse, they are united
by the common entity, the Fourier transform. In an age where it is impossible
to stay abreast with technology across the spectrum, it is stimulating to find a
theory and technique which enables one to invade an unfamiliar field with
familiar tools.

1.4 DIGITAL COMPUTER FOURIER ANALYSIS

Because of the wide range of problems which are susceptible to attack
by the Fourier transform, we would expect the logical extension of Fourier
transform analysis to the digital computer. Numerical integration of Eq. (1-1) implies the relationship

\[ S(f_k) = \sum_{n=0}^{N-1} s(t_n)e^{-2\pi if_k t_n}(t_{n+1} - t_n) \quad k = 0, 1, \ldots, N - 1 \quad (1-2) \]

For those problems which do not yield to a closed-form Fourier transform solution, the discrete Fourier transform (1-2) offers a potential method of attack. However, careful inspection of (1-2) reveals that if there are \( N \) data points of the function \( s(t_n) \) and if we desire to determine the amplitude of \( N \) separate sinusoids, then computation time is proportional to \( N^2 \), the number of multiplications. Even with high speed computers, computation of the discrete Fourier transform requires excessive machine time for large \( N \).

An obvious requirement existed for the development of techniques to reduce the computing time of the discrete Fourier transform; however, the scientific community met with little success. Then in 1965 Cooley and Tukey published their mathematical algorithm [8] which has become known as the “fast Fourier transform.” The fast Fourier transform (FFT) is a computational algorithm which reduces the computing time of Eq. (1-2) 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.

**1-5 HISTORICAL SUMMARY OF THE FAST FOURIER TRANSFORM**

During a meeting of the President’s Scientific Advisory Committee, Richard L. Garwin noted that John W. Tukey was writing Fourier transforms [9]. Garwin, who in his own research was in desperate need of a fast means to compute the Fourier transform, questioned Tukey as to his knowledge of techniques to compute the Fourier transform. Tukey outlined to Garwin essentially what has led to the famous Cooley-Tukey algorithm.

Garwin went to the computing center at IBM Research in Yorktown Heights to have the technique programmed. James W. Cooley was a relatively new member of the staff at IBM Research and by his own admission was given the problem to work on because he was the only one with nothing important to do [9]. At Garwin’s insistence, Cooley quickly worked out a computer program and returned to his own work with the expectation that this project was over and could be forgotten. However, requests for copies of the program and a writeup began accumulating, and Cooley was asked to write a paper on the algorithm. In 1965 Cooley and Tukey published the now famous “An Algorithm for the Machine Calculation of Complex Fourier Series” in the *Mathematics of Computation* [8].
Without the tenacity of Garwin, it is possible that the fast Fourier transform would still be relatively unknown today. The term relative is used because after Cooley and Tukey published their findings, reports of other people using similar techniques began to become known [10]. P. Rudnick [11] of Oceanic Institution in La Jolla, California reported that he was using a similar technique and that he had gotten his idea from a paper published in 1942 by Danielson and Lanczos [12]. This paper in turn referenced Runge [13] and [14] for the source of their methods. These two papers, together with the lecture notes of Runge and König [15], describe essentially the computational advantages of the FFT algorithm as we know it today.

L. H. Thomas of IBM Watson Laboratory also was using a technique [16] very similar to that published by Cooley and Tukey. He implied that he simply had gone to the library and looked up a method to do Fourier series calculations, a book by Stumpff [17]. Thomas generalized the concepts presented in Stumpff and derived a similar technique to what is now known as the fast Fourier transform.

Another line of development also led to an algorithm equivalent to that of Thomas. In 1937, Yates [18] developed an algorithm to compute the interaction of 2n factorial experiments. Good [19] extended this avenue of approach and outlined a procedure for the computation of N-point Fourier transforms which was essentially equivalent to that of Thomas.

The fast Fourier transform algorithm has had a long and interesting history. Unfortunately, not until recently did the contributions of those involved in its early history become known.

REFERENCES


