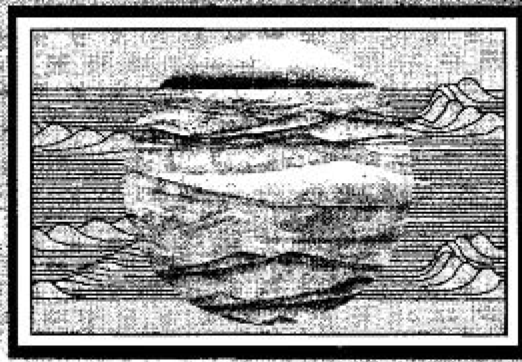
1 Introduction



The rich history and future promise of signal processing derive from a strong synergy between increasingly sophisticated applications, new theoretical developments and constantly emerging new hardware architectures and platforms. Signal processing applications span an immense set of disciplines that include entertainment, communications, space exploration, medicine, archaeology, geophysics, just to name a few. Signal processing algorithms and hardware are prevalent in a wide range of systems, from highly specialized military systems and industrial applications to low-cost, high-volume consumer electronics. Although we routinely take for granted the extraordinary performance of multimedia systems, such as high definition video, high fidelity audio, and interactive games, these systems have always relied heavily on state-of-the-art signal processing. Sophisticated digital signal processors are at the core of all modern cell phones. MPEG audio and video and JPEG[1] image data compression standards rely heavily on many of the signal processing principles and techniques discussed in this text. High-density data storage devices and new solid-state memories rely increasingly on the use of signal processing to provide consistency and robustness to otherwise fragile technologies. As we look to the future, it is clear that the role of signal processing is expanding, driven in part by the convergence of communications, computers, and signal processing in both the consumer arena and in advanced industrial and government applications.

The growing number of applications and demand for increasingly sophisticated algorithms go hand-in-hand with the rapid development of device technology for implementing signal processing systems. By some estimates, even with impending limitations

[1] The acronyms MPEG and JPEG are the terms used in even casual conversation for referring to the standards developed by the "Moving Picture Expert Group (MPEG)" and the "Joint Photographic Expert Group (JPEG)" of the "International Organization for Standardization (ISO)."

on Moore's Law, the processing capability of both special-purpose signal processing microprocessors and personal computers is likely to increase by several orders of magnitude over the next 10 years. Clearly, the importance and role of signal processing will continue to expand at an accelerating rate well into the future.

Signal processing deals with the representation, transformation, and manipulation of signals and the information the signals contain. For example, we may wish to separate two or more signals that have been combined by some operation, such as addition, multiplication, or convolution, or we may want to enhance some signal component or estimate some parameter of a signal model. In communications systems, it is generally necessary to do preprocessing such as modulation, signal conditioning, and compression prior to transmission over a communications channel, and then to carry out postprocessing at the receiver to recover a facsimile of the original signal. Prior to the 1960s, the technology for such signal processing was almost exclusively continuous-time analog technology.[2] A continual and major shift to digital technologies has resulted from the rapid evolution of digital computers and microprocessors and low-cost chips for analog to digital (A/D) and digital to analog (D/A) conversion. These developments in technology have been reinforced by many important theoretical developments, such as the fast Fourier transform (FFT) algorithm, parametric signal modeling, multirate techniques, polyphase filter implementation, and new ways of representing signals, such as with wavelet expansions. As just one example of this shift, analog radio communication systems are evolving into reconfigurable "software radios" that are implemented almost exclusively with digital computation.

Discrete-time signal processing is based on processing of numeric sequences indexed on integer variables rather than functions of a continuous independent variable. In digital signal processing (DSP), signals are represented by sequences of finite-precision numbers, and processing is implemented using digital computation. The more general term *discrete-time signal processing* includes digital signal processing as a special case but also includes the possibility that sequences of samples (sampled data) could be processed with other discrete-time technologies. Often the distinction between the terms discrete-time signal processing and digital signal processing is of minor importance, since both are concerned with discrete-time signals. This is particularly true when high-precision computation is employed. Although there are many examples in which signals to be processed are inherently discrete-time sequences, most applications involve the use of discrete-time technology for processing signals that originate as continuous-time signals. In this case, a continuous-time signal is typically converted into a sequence of samples, i.e., a discrete-time signal. Indeed, one of the most important spurs to widespread application of digital signal processing was the development of low-cost A/D, D/A conversion chips based on differential quantization with noise shaping. After discrete-time processing, the output sequence is converted back to a continuous-time signal. Real-time operation is often required or desirable for such systems. As computer speeds have increased, discrete-time processing of continuous-time signals in real time has become commonplace in communication systems, radar and sonar, speech and video coding and enhancement, biomedical engineering, and many

[2] In a general context, we shall refer to the independent variable as "time," even though in specific contexts, the independent variable may take on any of a broad range of possible dimensions. Consequently, continuous time and discrete time should be thought of as generic terms referring to a continuous independent variable and a discrete independent variable, respectively.

other areas of application. Non-real-time applications are also common. The compact disc player and MP3 player are examples of asymmetric systems in which an input signal is processed only once. The initial processing may occur in real time, slower than real time, or even faster than real time. The processed form of the input is stored (on the compact disc or in a solid state memory), and final processing for reconstructing the audio signal is carried out in real time when the output is played back for listening. The compact disc and MP3 recording and playback systems rely on many of the signal processing concepts that we discuss in this book.

Financial Engineering represents another rapidly emerging field which incorporates many signal processing concepts and techniques. Effective modeling, prediction and filtering of economic data can result in significant gains in economic performance and stability. Portfolio investment managers, for example, are relying increasingly on using sophisticated signal processing since even a very small increase in signal predictability or signal-to-noise ratio (SNR) can result in significant gain in performance.

Another important area of signal processing is *signal interpretation*. In such contexts, the objective of the processing is to obtain a characterization of the input signal. For example, in a speech recognition or understanding system, the objective is to interpret the input signal or extract information from it. Typically, such a system will apply digital pre-processing (filtering, parameter estimation, and so on) followed by a pattern recognition system to produce a symbolic representation, such as a phonemic transcription of the speech. This symbolic output can, in turn, be the input to a symbolic processing system, such as a rules-based expert system, to provide the final signal interpretation.

Still another relatively new category of signal processing involves the symbolic manipulation of signal processing expressions. This type of processing is potentially useful in signal processing workstations and for the computer-aided design of signal processing systems. In this class of processing, signals and systems are represented and manipulated as abstract data objects. Object-oriented programming languages provide a convenient environment for manipulating signals, systems, and signal processing expressions without explicitly evaluating the data sequences. The sophistication of systems designed to do signal expression processing is directly influenced by the incorporation of fundamental signal processing concepts, theorems, and properties, such as those that form the basis for this book. For example, a signal processing environment that incorporates the property that convolution in the time domain corresponds to multiplication in the frequency domain can explore a variety of rearrangements of filtering structures, including those involving the direct use of the discrete Fourier transform (DFT) and the FFT algorithm. Similarly, environments that incorporate the relationship between sampling rate and aliasing can make effective use of decimation and interpolation strategies for filter implementation. Similar ideas are currently being explored for implementing signal processing in network environments. In this type of environment, data can potentially be tagged with a high-level description of the processing to be done, and the details of the implementation can be based dynamically on the resources available on the network.

Many of the concepts and design techniques discussed in this text are now incorporated into the structure of sophisticated software systems such as MATLAB, Simulink, Mathematica, and LabVIEW. In many cases where discrete-time signals are acquired and stored in computers, these tools allow extremely sophisticated signal processing operations to be formed from basic functions. In such cases, it is not generally necessary to know the details of the underlying algorithm that implements the computation of an operation like the FFT, but nevertheless it is essential to understand what is computed and how it should be interpreted. In other words, a good understanding of the concepts considered in this text is essential for intelligent use of the signal processing software tools that are now widely available.

Signal processing problems are not confined, of course, to one-dimensional signals. Although there are some fundamental differences in the theories for one-dimensional and multidimensional signal processing, much of the material that we discuss in this text has a direct counterpart in multidimensional systems. The theory of multidimensional digital signal processing is presented in detail in a variety of references including Dudgeon and Mersereau (1984), Lim (1989), and Bracewell (1994).[3] Many image processing applications require the use of two-dimensional signal processing techniques. This is the case in such areas as video coding, medical imaging, enhancement and analysis of aerial photographs, analysis of satellite weather photos, and enhancement of video transmissions from lunar and deep-space probes. Applications of multidimensional digital signal processing to image processing are discussed, for example, in Macovski (1983), Castleman (1996), Jain (1989), Bovic (ed.) (2005), Woods (2006), Gonzalez and Woods (2007), and Pratt (2007). Seismic data analysis as required in oil exploration, earthquake measurement, and nuclear test monitoring also uses multidimensional signal processing techniques. Seismic applications are discussed in, for example, Robinson and Treitel (1980) and Robinson and Durrani (1985).

Multidimensional signal processing is only one of many advanced and specialized topics that build on the fundamentals covered in this text. Spectral analysis based on the use of the DFT and the use of signal modeling is another particularly rich and important aspect of signal processing. We discuss many facets of this topic in Chapters 10 and 11, which focus on the basic concepts and techniques relating to the use of the DFT and parametric signal modeling. In Chapter 11, we also discuss in some detail high resolution spectrum analysis methods, based on representing the signal to be analyzed as the response of a discrete-time linear time-invariant (LTI) filter to either an impulse or to white noise. Spectral analysis is achieved by estimating the parameters (e.g., the difference equation coefficients) of the system and then evaluating the magnitude squared of the frequency response of the model filter. Detailed discussions of spectrum analysis can be found in the texts by Kay (1988), Marple (1987), Therrien (1992), Hayes (1996) and Stoica and Moses (2005).

Signal modeling also plays an important role in data compression and coding, and here again, the fundamentals of difference equations provide the basis for understanding many of these techniques. For example, one class of signal coding techniques, referred to as linear predictive coding (LPC), exploits the notion that if a signal is the response of a certain class of discrete-time filters, the signal value at any time index is a linear function of (and thus linearly predictable from) previous values. Consequently, efficient signal representations can be obtained by estimating these prediction parameters and using them along with the prediction error to represent the signal. The signal can then be regenerated when needed from the model parameters. This class of signal

[3] Authors names and dates are used throughout the text to refer to books and papers listed in the Bibliography at the end of the book.

coding techniques has been particularly effective in speech coding and is described in considerable detail in Jayant and Noll (1984), Markel and Gray (1976), Rabiner and Schafer (1978) and Quatieri (2002), and is also discussed in some detail in Chapter 11.

Another advanced topic of considerable importance is adaptive signal processing. Adaptive systems represent a particular class of time-varying and, in some sense, nonlinear systems with broad application and with established and effective techniques for their design and analysis. Again, many of these techniques build from the fundamentals of discrete-time signal processing covered in this text. Details of adaptive signal processing are given by Widrow and Stearns (1985), Haykin (2002) and Sayed (2008).

These represent only a few of the many advanced topics that extend from the content covered in this text. Others include advanced and specialized filter design procedures, a variety of specialized algorithms for evaluation of the Fourier transform, specialized filter structures, and various advanced multirate signal processing techniques, including wavelet transforms. (See Burrus, Gopinath, and Guo (1997), Vaidyanathan (1993) and Vetterli and Kovačević (1995) for introductions to these topics.)

It has often been said that the purpose of a fundamental textbook should be to uncover, rather than cover, a subject. In choosing the topics and depth of coverage in this book, we have been guided by this philosophy. The preceding brief discussion and the Bibliography at the end of the book make it abundantly clear that there is a rich variety of both challenging theory and compelling applications to be uncovered by those who diligently prepare themselves with a study of the fundamentals of DSP.

HISTORIC PERSPECTIVE

Discrete-time signal processing has advanced in uneven steps over time. Looking back at the development of the field of discrete-time signal processing provides a valuable perspective on fundamentals that will remain central to the field for a long time to come. Since the invention of calculus in the 17th century, scientists and engineers have developed models to represent physical phenomena in terms of functions of continuous variables and differential equations. However, numeric techniques have been used to solve these equations when analytical solutions are not possible. Indeed, Newton used finite-difference methods that are special cases of some of the discrete-time systems that we present in this text. Mathematicians of the 18th century, such as Euler, Bernoulli, and Lagrange, developed methods for numeric integration and interpolation of functions of a continuous variable. Interesting historic research by Heideman, Johnson, and Burrus (1984) showed that Gauss discovered the fundamental principle of the FFT (discussed in Chapter 9) as early as 1805—even before the publication of Fourier's treatise on harmonic series representation of functions.

Until the early 1950s, signal processing as we have defined it was typically carried out with analog systems implemented with electronic circuits or even with mechanical devices. Even though digital computers were becoming available in business environments and in scientific laboratories, they were expensive and had relatively limited capabilities. About that time, the need for more sophisticated signal processing in some application areas created considerable interest in discrete-time signal processing. One of the first uses of digital computers in DSP was in geophysical exploration, where relatively low frequency seismic signals could be digitized and recorded on magnetic tape for later processing. This type of signal processing could not generally be done in real time; minutes or even hours of computer time were often required to process only seconds of data. Even so, the flexibility of the digital computer and the potential payoffs made this alternative extremely inviting.

Also in the 1950s, the use of digital computers in signal processing arose in a different way. Because of the flexibility of digital computers, it was often useful to simulate a signal processing system on a digital computer before implementing it in analog hardware. In this way, a new signal processing algorithm or system could be studied in a flexible experimental environment before committing economic and engineering resources to constructing it. Typical examples of such simulations were the vocoder simulations carried out at Massachusetts Institute of Technology (MIT) Lincoln Laboratory and Bell Telephone Laboratories. In the implementation of an analog channel vocoder, for example, the filter characteristics affected the perceived quality of the coded speech signal in ways that were difficult to quantify objectively. Through computer simulations, these filter characteristics could be adjusted and the perceived quality of a speech coding system evaluated prior to construction of the analog equipment.

In all of these examples of signal processing using digital computers, the computer offered tremendous advantages in flexibility. However, the processing could not be done in real time. Consequently, the prevalent attitude up to the late 1960s was that the digital computer was being used to *approximate*, or *simulate*, an analog signal processing system. In keeping with that style, early work on digital filtering concentrated on ways in which a filter could be programmed on a digital computer so that with A/D conversion of the signal, followed by digital filtering, followed by D/A conversion, the overall system approximated a good analog filter. The notion that digital systems might, in fact, be practical for the actual real-time implementation of signal processing in speech communication, radar processing, or any of a variety of other applications seemed, even at the most optimistic times, to be highly speculative. Speed, cost, and size were, of course, three of the important factors in favor of the use of analog components.

As signals were being processed on digital computers, researchers had a natural tendency to experiment with increasingly sophisticated signal processing algorithms. Some of these algorithms grew out of the flexibility of the digital computer and had no apparent practical implementation in analog equipment. Thus, many of these algorithms were treated as interesting, but somewhat impractical, ideas. However, the development of such signal processing algorithms made the notion of all-digital implementation of signal processing systems even more tempting. Active work began on the investigation of digital vocoders, digital spectrum analyzers, and other all-digital systems, with the hope that eventually, such systems would become practical.

The evolution of a new point of view toward discrete-time signal processing was further accelerated by the disclosure by Cooley and Tukey (1965) of an efficient class of algorithms for computation of Fourier transforms known collectively as the FFT. The FFT was significant for several reasons. Many signal processing algorithms that had been developed on digital computers required processing times several orders of magnitude greater than real time. Often, this was because spectrum analysis was an important component of the signal processing and no efficient means were available for implementing it. The FFT reduced the computation time of the Fourier transform by orders of magnitude, permitting the implementation of increasingly sophisticated signal processing algorithms with processing times that allowed interactive experimentation with the system. Furthermore, with the realization that the FFT algorithms might, in fact, be implementable with special-purpose digital hardware, many signal processing algorithms that previously had appeared to be impractical began to appear feasible.

Another important implication of the FFT was that it was an inherently discrete-time concept. It was directed toward the computation of the Fourier transform of a discrete-time signal or sequence and involved a set of properties and mathematics that was exact in the discrete-time domain—it was not simply an approximation to a continuous-time Fourier transform. This had the effect of stimulating a reformulation of many signal processing concepts and algorithms in terms of discrete-time mathematics, and these techniques then formed an exact set of relationships in the discrete-time domain. Following this shift away from the notion that signal processing on a digital computer was merely an approximation to analog signal processing techniques, there emerged the current view that discrete-time signal processing is an important field of investigation in its own right.

Another major development in the history of discrete-time signal processing occurred in the field of microelectronics. The invention and subsequent proliferation of the microprocessor paved the way for low-cost implementations of discrete-time signal processing systems. Although the first microprocessors were too slow to implement most discrete-time systems in real time except at very low sampling rates, by the mid-1980s, integrated circuit technology had advanced to a level that permitted the implementation of very fast fixed-point and floating-point microcomputers with architectures specially designed for implementing discrete-time signal processing algorithms. With this technology came, for the first time, the possibility of widespread application of discrete-time signal processing techniques. The rapid pace of development in microelectronics also significantly impacted the development of signal processing algorithms in other ways. For example, in the early days of real-time digital signal processing devices, memory was relatively costly and one of the important metrics in developing signal processing algorithms was the efficient use of memory. Digital memory is now so inexpensive that many algorithms purposely incorporate more memory than is absolutely required so that the power requirements of the processor are reduced. Another area in which technology limitations posed a significant barrier to widespread deployment of DSP was in conversion of signals from analog to discrete-time (digital) form. The first widely available A/D and D/A converters were stand-alone devices costing thousands of dollars. By combining digital signal processing theory with microelectronic technology, oversampled A/D and D/A converters costing a few dollars or less have enabled a myriad of real-time applications.

In a similar way, minimizing the number of arithmetic operations, such as multiplies or floating point additions, is now less essential, since multicore processors often have several multipliers available and it becomes increasingly important to reduce communication between cores, even if it then requires more multiplications. In a multicore environment, for example, direct computation of the DFT (or the use of the Goertzel algorithm) is more "efficient" than the use of an FFT algorithm since, although many more multiplications are required, communication requirements are significantly reduced because the processing can be more efficiently distributed among multiple processors or cores. More broadly, the restructuring of algorithms and the development of new ones to exploit the opportunity for more parallel and distributed processing is becoming a significant new direction in the development of signal processing algorithms.

FUTURE PROMISE

Microelectronics engineers continue to strive for increased circuit densities and production yields, and as a result, the complexity and sophistication of microelectronic systems continually increase. The complexity, speed, and capability of DSP chips have grown exponentially since the early 1980s and show no sign of slowing down. As wafer-scale integration techniques become highly developed, very complex discrete-time signal processing systems will be implemented with low cost, miniature size, and low power consumption. Furthermore, technologies such as microelectronic mechanical systems (MEMS) promise to produce many types of tiny sensors whose outputs will need to be processed using DSP techniques that operate on distributed arrays of sensor inputs. Consequently, the importance of discrete-time signal processing will continue to increase, and the future development of the field promises to be even more dramatic than the course of development that we have just described.

Discrete-time signal processing techniques have already promoted revolutionary advances in some fields of application. A notable example is in the area of telecommunications, where discrete-time signal processing techniques, microelectronic technology, and fiber optic transmission have combined to change the nature of communication systems in truly revolutionary ways. A similar impact can be expected in many other areas. Indeed, signal processing has always been, and will always be, a field that thrives on new applications. The needs of a new field of application can sometimes be filled by knowledge adapted from other applications, but frequently, new application needs stimulate new algorithms and new hardware systems to implement those algorithms. Early on, applications to seismology, radar, and communication provided the context for developing many of the core signal processing techniques that we discuss in this book. Certainly, signal processing will remain at the heart of applications in national defense, entertainment, communication, and medical care and diagnosis. Recently, we have seen applications of signal processing techniques in new areas as disparate as finance and DNA sequence analysis.

Although it is difficult to predict where other new applications will arise, there is no doubt that they will be obvious to those who are prepared to recognize them. The key to being ready to solve new signal processing problems is, and has always been, a thorough grounding in the fundamental mathematics of signals and systems and in the associated design and processing algorithms. While discrete-time signal processing is a dynamic, steadily growing field, its fundamentals are well formulated, and it is extremely valuable to learn them well. Our goal in this book is to uncover the fundamentals of the field by providing a coherent treatment of the theory of discrete-time linear systems, filtering, sampling, discrete-time Fourier analysis, and signal modeling. This text should provide the reader with the knowledge necessary for an appreciation of the wide scope of applications for discrete-time signal processing and a foundation for contributing to future developments in this exciting field.