Appendix A: The Fourier Transform

In Chap. 2 it was shown that the Fourier transform plays an important role in the analysis of LTI systems. This appendix summarizes some of the important properties of Fourier transforms and their numerical calculation.

A.1 Properties of the Fourier Transform

A function \( f(t) \) and its Fourier transform, \( F(\omega) \), are related by the Fourier transform pair [1, 3]:

\[
F(\omega) = \int_{-\infty}^{+\infty} f(t)e^{i\omega t}dt, \quad (A.1)
\]

\[
f(t) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} F(\omega)e^{-i\omega t}d\omega. \quad (A.2)
\]

We denote the relationship between these functions symbolically as

\[
f(t) \leftrightarrow F(\omega). \quad (A.3)
\]

Through the use of the definition of the Fourier transform one can show that the following relationships also exist

1. Differentiation

\[
f^{(n)}(t) \leftrightarrow (-i\omega)^n F(\omega)
\]

provided \( f^{(m)}(t) \to 0 \) as \( |t| \to \infty \), for \( m = 0, \ldots, n - 1 \), where

\[
f^{(n)}(t) = \partial^n f / \partial t^n.
\]
2. Shifting

\[ f(t \pm t_0) \leftrightarrow F(\omega)\exp(\mp i\omega t_0) \]

3. Scaling

\[ f(at) \leftrightarrow F(\omega/a)/a \quad \text{for} \quad a > 0 \]

4. Integration

\[
\int_{-\infty}^{t} f(t)dt \leftrightarrow \pi F(0)\delta(\omega) - \frac{F(\omega)}{i\omega}
\]

\[ \pi f(0)\delta(t) + \frac{f(t)}{it} \leftrightarrow \int_{-\infty}^{\infty} F(\omega)d\omega \]

5. Conjugation

\[ f^*(t) \leftrightarrow F^*(-\omega) \]

\[ f^*(-t) \leftrightarrow F^*(\omega) \quad \text{where} \quad (a + ib)^* = (a - ib) \]

6. Convolution

\[
\int_{-\infty}^{+\infty} f_1(\tau)f_2(t - \tau)d\tau \leftrightarrow F_1(\omega)F_2(\omega)
\]

\[ f_1(t)f_2(t) \leftrightarrow 1/2\pi \int_{-\infty}^{+\infty} F_1(\xi)F_2(\omega - \xi)d\xi \]

7. Correlation

\[
\int_{-\infty}^{+\infty} f_1^*(\tau)f_2(t + \tau)d\tau \leftrightarrow F_1^*(\omega)F_2(\omega)
\]

\[ f_1^*(t)f_2(t) \leftrightarrow 1/2\pi \int_{-\infty}^{+\infty} F_1^*(\xi)F_2(\omega + \xi)d\xi \]
8. Parsevals’ theorem

\[
\int_{-\infty}^{+\infty} f_1^*(\tau)f_2(\tau)d\tau = \frac{1}{2\pi} \int_{-\infty}^{+\infty} F_1^*(\omega)F_2(\omega)d\omega
\]

and when \( f_1 = f_2 = f \)

\[
\int_{-\infty}^{+\infty} |f(t)|^2 dt = \frac{1}{2\pi} \int_{-\infty}^{+\infty} |F(\omega)|^2 d\omega
\]

A.2 Some Fourier Transform Pairs

\[
f(t) \quad F(\omega)
\]

1. Dirac delta function

\[
\delta(t) \quad 1
\]

2. Heaviside step function

\[
H(t) = \begin{cases} 
1 & t > 0 \\
1/2 & t = 0 \\
0 & t < 0 
\end{cases} \quad \pi\delta(\omega) + \frac{i}{\omega}
\]

3. Sign function

\[
\text{sgn}(t) = \begin{cases} 
1 & t > 0 \\
-1 & t < 0 
\end{cases} \quad \frac{2i}{\omega}
\]

4. \(|t|\) \quad -\frac{2}{\omega^2}

5. \(\frac{1}{t}\) \quad i\pi \text{sgn}(\omega)

6. \(\exp(-at)H(t)\) \quad \frac{a + i\omega}{a^2 + \omega^2}

7. Laplace function

\[
\exp(-a|t|) \quad \frac{2a}{a^2 + \omega^2}
\]
8. Gauss function

\[ \exp(-at^2) \quad \sqrt{\frac{\pi}{a}} \exp\left(-\frac{\omega^2}{4a}\right) \]

9. Rectangular function

\[ \Pi(t) = \begin{cases} 1 & |t| < 1/2 \\ 0 & |t| > 1/2 \end{cases} \quad \frac{\sin(\omega/2)}{\omega/2} = \text{sinc}(\omega/2\pi) \]

10. Triangular function

\[ \Lambda(t) = \begin{cases} 1 - |t| & |t| \leq 1 \\ 0 & |t| > 1 \end{cases} \quad \text{sinc}^2(\omega/2\pi) \]

A.3 The Discrete Fourier Transform

The typical output of an ultrasonic system is an analog voltage versus time waveform. Most modern NDE measurement systems digitize such waveforms so that the sampled values can be manipulated by computer. In this section we will examine the consequences of dealing with a sampled signal and consider the way in which the Fourier transform can be computed from these samples in an efficient manner. To begin, we rewrite the Fourier transform (and its inverse) in terms of the frequency \( f \) (in Hertz) instead of the circular frequency \( \omega \):

\[
X(f) = \int_{-\infty}^{+\infty} x(t) \exp(2\pi if t) \, dt \\
x(t) = \int_{-\infty}^{+\infty} X(f) \exp(-2\pi if t) \, df. \tag{A.4}
\]

Now we consider the discrete (sampled) values of \( x(t) \) at the times \( t_j = j\Delta t \) where \( j = 0, \pm 1, \pm 2, \ldots \) and we let \( f_s = 1/\Delta t \). Then these sampled values are given by

\[
x(t_j) = \int_{-\infty}^{+\infty} X(f) \exp(-2\pi jf / f_s) \, df, \tag{A.5}
\]

\[ \]
which can be written equivalently as an infinite sum over a set of finite intervals as

$$x(t_j) = \sum_{k=-\infty}^{\infty} \int_{k f_s}^{(k+1)f_s} X(f) \exp(-2\pi j f / f_s) df.$$  \hfill (A.6)

Since the exponential function in Eq. (A.6) is periodic in $f$ with period $f_s$, the expression in Eq. (A.6) is unchanged if we replace $\exp(-2\pi j f / f_s)$ by $\exp(-2\pi j (f - kf_s) / f_s)$. However,

$$\int_{k f_s}^{(k+1)f_s} X(f) \exp(-2\pi j f / f_s) df = \int_{0}^{f_s} X(u + kf_s) \exp(-2\pi j u / f_s) du,$$

which follows directly by letting $u = f - kf_s$, so

$$x(t_j) = \sum_{k=-\infty}^{\infty} \int_{0}^{f_s} X(u + kf_s) \exp(-2\pi j u / f_s) du.$$  \hfill (A.8)

Now, let $u = f$ and define

$$X_p(f) = \sum_{k=-\infty}^{\infty} X(f + kf_s).$$ \hfill (A.9)

Then

$$x(t_j) = \int_{0}^{f_s} X_p(f) \exp(-2\pi j f / f_s) df.$$  \hfill (A.10)

Since $X_p(f)$ is a periodic function in $f$ with period $f_s$, it can be expanded in a Fourier series as

$$X_p(f) = \sum_{j=-\infty}^{\infty} C_j \exp(2\pi j f / f_s),$$ \hfill (A.11)

where the Fourier coefficients are given by

$$C_j = \frac{1}{f_s} \int_{0}^{f_s} X_p(f) \exp(-2\pi j f / f_s) df.$$ \hfill (A.12)
Recognizing these Fourier coefficients as just $x(t_j)/f_s$ from Eq. (A.11), we have

$$X_p(f) = \frac{1}{f_s} \sum_{j=-\infty}^{\infty} x(t_j) \exp(2\pi i j f / f_s). \quad (A.13)$$

Thus, knowing all the sampled values of $x(t)$, Eq. (A.13) shows that we recover $X_p(f)$ rather than $X(f)$ itself. The function $X_p(f)$ differs from our original $X(f)$ by the sum of $X(f)$’s displaced at multiples of $f_s$ (Fig. A.1). This difference is referred to as aliasing. If $X(f)$ is band limited, i.e. $X(f) = 0$ for $|f| > f_{\text{max}}$ where $f_{\text{max}}$ is the maximum frequency in the signal, then we see that $X_p(f) = X(f)$ for $|f| < f_{\text{max}}$ if $f_s \geq 2f_{\text{max}}$ (Fig. A.2). Even when $X(f)$ is not band limited, if $f_{\text{max}}$ represents the

---

**Fig. A.1** (a) A Fourier transform spectra, $X(f)$, and (b) its infinitely repeated periodic replica, $X_p(f)$

**Fig. A.2** (a) A Fourier transform spectra, $X(f)$, of a band limited function and (b) its infinitely repeated replica, $X_p(f)$, if $f_s > 2f_{\text{max}}$
maximum significant frequency contained in the signal we will have \(X_p(f) = X(f)\) approximately for \(|f| < f_{\text{max}}\), again as long as

\[
f_s \geq 2f_{\text{max}}, \quad (A.14)
\]

which is the well-known Nyquist sampling criterion. This criterion says that in order to recover a true replica of the frequency spectra of our original signal in the frequency range \(|f| < f_{\text{max}}\), we should sample our signal at least twice the value of the maximum significant frequency in that signal. Since the Nyquist criterion only gives a lower bound for the sampling frequency, in practice we usually sample a signal at a much higher rate such as \(5f_{\text{max}} \rightarrow 10f_{\text{max}}\) to be conservative.

Now, suppose that we sample \(X_p(f)\) in Eq. (A.13) at \(f = n\Delta f\), \(n = 0, \pm 1, \pm 2, \ldots\) where \(\Delta f = 1/T\) and \(T\) is the total time interval sampled. Then

\[
X_p(f_n) = 1/f_s \sum_{j=-\infty}^{\infty} x(t_j)\exp(2\pi ijn/Tf_s). \quad (A.15)
\]

However, if we take \(Tf_s = N\) where \(N\) is an integer and use the fact that \(\exp(2\pi ijn/N)\) is periodic in \(j\) with period \(N\), we can rewrite \(X_p(f_n)\) as

\[
X_p(f_n) = T/N \sum_{j=0}^{N-1} x_p(t_j)\exp(2\pi ijn/N), \quad (A.16)
\]

where

\[
x_p(t_j) = \sum_{k=-\infty}^{\infty} x(t_j + kT) \quad (A.17)
\]

are the sampled values of a periodic function \(x_p(t)\) given by

\[
x_p(t) = \sum_{k=-\infty}^{\infty} x(t + kT), \quad (A.18)
\]

which is formed from \(x(t)\) in the same manner as \(X_p(f)\) is formed from \(X(f)\). Eq. (A.16) is in the form of an inverse discrete Fourier transform [2] that relates the sampled values of \(X_p(f)\) to the sampled values of \(x_p(t)\). If aliasing is negligible, then Eq. (A.16) is also a relation between the sampled values of our original wave form \(x(t)\) and its frequency components \(X(f)\).

Given the sampled values \(X_p(f)\), we can also recover \(x_p(t)\) via an inverse relationship similar to Eq. (A.16). To see this consider the sum

\[
\sum_{n=0}^{N-1} X_p(f_n)\exp(-2\pi ikn/N) = T/N \sum_{n=0}^{N-1} \sum_{j=0}^{N-1} x_p(t_j)\exp(-2\pi i (k - j)n/N). \quad (A.19)
\]
However,

\[ \frac{1}{N} \sum_{n=0}^{N-1} \exp(-2\pi i (k - j)n/N) = \Delta(j - k), \tag{A.20} \]

where \( \Delta(j - k) \) is the discrete delta function, having a sampling property similar to the Dirac delta function (see problem A.1), i.e.

\[ \sum_{j=0}^{N-1} c_j \Delta(j - k) = c_k \quad k = 0, 1, 2, \ldots N - 1 \tag{A.21} \]

Thus, we find

\[ Tx_p(t_k) = \sum_{n=0}^{N-1} X_p(f_n) \exp(-2\pi i kn/N), \tag{A.22} \]

which is the desired inversion formula. In the absence of aliasing, the discrete forward Fourier transform pair (Eq. (A.22) and Eq. (A.16)) give us an explicit way to calculate sampled values of the Fourier transform \( X(f) \) directly from the sampled values of the waveform \( x(t) \) and vice versa. However, for large \( N \), this may still involve a considerable amount of computation. Fortunately, there is a way to reduce the computation required as discussed in the next section.

### A.4 The Fast Fourier Transform

A direct evaluation of the discrete Fourier transforms in Eqs. (A.16) and (A.22) requires on the order of \((N - 1)^2\) multiplications, a value which grows rapidly as \(N\) increases (see Table A.1). In 1965, however, Cooley and Tukey [5] published an algorithm that can reduce the number of multiplications required to an order of \(N/2 \log_2 N\), a value that grows much slower (Table A.1). This algorithm, now known as the fast Fourier transform (FFT) [2], has made the numerical calculation of Fourier integrals and their inverses practical.

Many software packages, such as MATLAB, MathCad, and Mathematica, for example, compute the discrete Fourier transforms with an FFT algorithm. One must be aware of some subtle differences between those various

| Table A.1 Number of calculations needed for direct evaluation of the discrete Fourier transform vs the FFT |
|---|---|---|
| \(N\) | \((N-1)^2\) | \(N/2 \log_2 N\) |
| 256 | 65,025 | 1,024 |
| 1,024 | 1,046,529 | 5,120 |
| 4,096 | 16,769,025 | 24,576 |
implementations. In this book we have defined the forward and inverse Fourier transforms by the relations in Eqs. (A.1) and (A.2). However, we can also define these pair of equations in the forms

\[
\tilde{F}(\omega) = n_1 \int_{-\infty}^{+\infty} f(t) \exp(\pm i\omega t) dt
\]

\[
f(t) = n_2 \int_{-\infty}^{+\infty} \tilde{F}(\omega) \exp(\mp i\omega t) d\omega
\]

as long as the constants \((n_1, n_2)\) satisfy \(n_1 n_2 = 1/2\pi\). Different choices will obviously lead to different values for the Fourier transform which is why we have placed a tilde over the transform in Eq. (A.23). In analytical discussions of wave propagation, one must be especially aware of the signs chosen in Eq. (A.23), since this can alter the physical meaning of the result. In Chap. 4, for example, we expressed a 1-D plane wave traveling in the + \(x\)-direction by its Fourier transform (see Eq. (4.5)) as

\[
f(t - x/c) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} F(\omega) \exp[i\omega(x/c - t)]d\omega,
\]

where

\[
F(\omega) = \int_{-\infty}^{+\infty} f(t) \exp(i\omega t)dt.
\]

Equation (A.24a) can be interpreted as representing the traveling pulse as a superposition of harmonic waves of the form \(F(\omega)\exp(ikx - i\omega t)\). This corresponds to choosing the minus sign in exponential term for the inverse Fourier transform as done in Eq. (A.2). If we had chosen the plus sign instead then we would have instead of Eq. (A.24a) the result:

\[
f(t - x/c) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} \tilde{F}(\omega) \exp[i\omega(-x/c + t)]d\omega,
\]

where the Fourier transform is now the complex conjugate of the Fourier transform appearing in Eq. (A.24a) since

\[
\tilde{F}(\omega) = \int_{-\infty}^{+\infty} f(t) \exp(-i\omega t)dt = F^*(\omega).
\]
In discussing harmonic waves some authors will implicitly assume one of these sign choices without stating them explicitly. For example, a plane wave traveling in the $+x$-direction may be written as $A(\omega)\exp(-ikx)$. This means that the author is implicitly using the form of Eq. (A.25) and the complex amplitude, $A(\omega)$, will be the complex conjugate of the same amplitude obtained when using our definitions of Eqs. (A.1) and (A.2). Differences due to different definitions can also arise in the case of the discrete Fourier transform and its implementation. In our case, we derived the discrete Fourier transform and its inverse directly from Eqs. (A.1) and (A.2), obtaining Eqs. (A.16) and (A.22), which we rewrite here as:

$$X_p(f_n) = \Delta t \sum_{j=0}^{N-1} x(t_j) \exp(2\pi j n/N)$$  \hspace{1cm} (A.27)

$$x_p(t_k) = \frac{1}{N\Delta t} \sum_{n=0}^{N-1} X_p(f_n) \exp(-2\pi i k n/N).$$

Other equivalent forms are

$$\tilde{X}_p(f_n) = m_1 \sum_{j=0}^{N-1} x(t_j) \exp(\pm 2\pi j n/N)$$  \hspace{1cm} (A.28)

$$x_p(t_k) = m_2 \sum_{n=0}^{N-1} \tilde{X}_p(f_n) \exp(\mp 2\pi i k n/N)$$

as long as the constants ($m_1, m_2$) satisfy $m_1 m_2 = 1/N$. In this book, we will exclusively use Eqs. (A.1) and (A.2) for the Fourier transform pair and Eq. (A.27) for the discrete Fourier transform pair. Numerical evaluation of the discrete Fourier transform pair will be performed in MATLAB, but the built-in MATLAB functions fft and ifft do not implement Eq. (A.27). Instead these functions are defined as

$$\text{fft}(x) \rightarrow X(n) = \sum_{j=1}^{N} x(j) \exp[-2\pi i (j - 1)(n - 1)/N]$$  \hspace{1cm} (A.29)

$$\text{ifft}(X) \rightarrow x(k) = \frac{1}{N} \sum_{n=1}^{N} X(n) \exp[2\pi i (n - 1)(k - 1)/N].$$

We see that the signs have been changed on the exponentials from our definitions and that the sampling interval, $\Delta t$, is missing in the coefficients. Also, the indexing is shifted by one so the sum ranges from one to $N$ rather than zero to $N-1$. To perform the Fourier transform pair of relations in MATLAB using our definitions (but keeping the same indexing as MATLAB), we have therefore defined two new MATLAB functions, FourierT(x, $\Delta t$) and IFourierT(X, $\Delta t$) where x is a vector.
or matrix of \(N\) sampled time domain values and \(X\) is a vector or matrix of \(N\) sampled frequency domain values. As with the built-in MATLAB functions, if the inputs are matrices, then the sampled data is assumed to be arranged in columns and the Fourier transforms or their inverse are performed on those columns. More details on these functions can be found in [4]. These MATLAB functions are:

\[
\text{function } y = \text{FourierT}(x, dt)
\]

% FourierT(x,dt) computes forward FFT of x with sampling time interval dt
% FourierT approximates the Fourier transform where the integrand of the % transform is \(x \times \exp(2\pi i f t)\)
% For NDE applications the sampled frequencies are normally in MHz, so % dt is in microseconds

\[
[\text{nr, nc}] = \text{size}(x);
\]

if nr == 1
    \(N = \text{nc};\)
else
    \(N = nr;\)
end
\(y = N \times dt \times \text{ifft}(x);\)

\[
\text{function } y = \text{IFourierT}(x, dt)
\]

% IFourierT(x,dt) computes the inverse FFT of x, for a sampling time % interval dt
% IFourierT assumes the integrand of the inverse transform is given by % \(x \times \exp(-2\pi i f t)\)
% The first half of the sampled values of x are the spectral components for % positive frequencies ranging from 0 to the Nyquist frequency \(1/(2 \times dt)\)
% The second half of the sampled values are the spectral components for % the corresponding negative frequencies. If these negative frequency % values are set equal to zero then to recover the inverse FFT of x we must % replace \(x(1)\) by \(x(1)/2\) and then compute \(2 \times \text{real(IFourierT(x,dt))}\)

\[
[\text{nr, nc}] = \text{size}(x);
\]

if nr == 1
    \(N = \text{nc};\)
else
    \(N = nr;\)
end
\(y = (1/(N \times dt)) \times \text{fft}(x);\)

In performing FFTs on sampled functions in MATLAB we must also be aware of a sampling issue. This is illustrated in Fig. A.3 where sampled time domain and frequency domain functions are shown. In the time domain, the \(N\) sampled points go from time \(t = 0\) to time \(t = (N-1) \Delta t\) over a time window of length \(T = N \Delta t\), while in the frequency domain the samples from frequency \(f = 0\) to frequency
\[ f = \frac{(N - 1)}{\Delta t} \] over a frequency window of length \( f_s = 1/\Delta t \). If we use the MATLAB function \( e = \text{linspace}(0, E, N) \) to generate \( N \) sampled values \( e \) (representing times or frequencies) over the interval from zero to \( E \) we will not obtain the proper sampled values shown in Fig. A.3 and this application of \text{linspace} will produce an incorrect sampling interval given by \( \Delta e = E/(N - 1) \). Thus, we need to have a function that does produce the proper sampled values. The function \text{s_space} listed below is a small modification of \text{linspace} that fills that need and should be used in place of \text{linspace} when generating sampled time or frequency values for performing Fourier analysis with FFTs:

```matlab
function y = s_space(xstart, xend, num)
% s_space(xstart,xend, num) generates num evenly spaced sampled
% values from xstart to (xend - dx), where dx = (xend - xstart)/num is
% the sample spacing. This is useful in FFT analysis where we generate
% sampled periodic functions. Example: generate 1000
% sampled frequencies from 0 to 100MHz via f = s_space(0,100,1000);
% In this case the last value of f will be 99.9 MHz and the
% sampling interval will be 100/1000 = 0.1 MHz.

ye = linspace(xstart, xend, num+1);
y = ye(1:num);
```

Using \text{s_space} in conjunction with \text{FourierT} and \text{IFourierT} gives you the tools necessary to perform time and frequency domain evaluations consistent with the Fourier transform pair definitions used throughout this book. For additional information on Fourier analysis see [4].
A.5 Problems

A.1. Define the discrete delta function, $\Delta$, as

$$\Delta(k - n) = \begin{cases} 1 & k = n \\ 0 & k \neq n \end{cases}.$$ 

From this definition, the sampling property of the discrete delta function (Eq. (A.21)) follows directly. Show that

$$\Delta(k - n) = \frac{1}{N} \sum_{m=0}^{N-1} \exp(2\pi im(k - n)/N)$$

i.e. prove that Eq. (A.20) is valid. (Hint: show that the series can be summed explicitly to the value

$$\left(\frac{w^{N(k-n)} - 1}{w^{(k-n)} - 1}\right),$$

where $w = \exp(2\pi i/N)$.

A.2. Consider the wave form given by

$$x(t) = \begin{cases} e^{-t} & t > 0 \\ 0 & t < 0 \end{cases},$$

where $t$ is measured in $\mu$sec.

(a) What is the analytical Fourier transform, $X(f)$, of this function? Plot the real and imaginary parts of $X(f)$ from $f = 0$ to $f = 4$ MHz. Also plot the magnitude of $X(f)$ over this same range.

(b) Suppose we sample $x(t)$ from $t = 0$ to $t = 8 \mu$sec with $N = 16$, i.e. $\Delta t = 0.5 \mu$sec. Note that there is a discontinuity in $x(t)$ at $t = 0$. Take the value to be 0.5 at this sample point. Why do we do this? Plot the sampled function and compare with the original function. Is the sampling adequate to represent this function?

(c) Compute the FFT of this sampled wave form. What is $f_s$ here? Plot the sampled values of the frequency components from $f = 0$ to $f = 2$ MHz. Compare with your results from part (a). Is aliasing a problem here?

(d) Repeat parts (b) and (c) changing only $N$ to $N = 32$. How do your results change in parts (b) and (c)?

(e) Repeat parts (b) and (c), changing $T$ to $T = 16 \mu$sec and $N$ to $N = 32$. How do your results change from parts (b) and (c)?
References

Appendix B: The Dirac Delta Function

In Chap. 2 we saw that the delta function also plays an important role in determining the response of LTI systems since it acts as an ideal input function whose Fourier transform contains all frequencies equally weighted. Strictly speaking the delta function is not an ordinary function but is a distribution \([1, 2]\). However, we will continue to speak of \(\delta(t)\) as an ordinary function, remembering that \(\delta\) only has a meaning in terms of its sifting property, i.e.

\[
\int_{a}^{b} \delta(t-x)f(x)dx = f(t) \quad a < t < b
\]

\[
= 0 \quad t < a \quad \text{or} \quad t > b
\]

\[
= f(t)/2 \quad t = a \quad \text{or} \quad t = b.
\]

The delta function also plays an important role in dealing with integral equations and boundary value problems, as discussed in Chap. 5 \([3]\).

B.1 Properties of the Delta Function

Several of the more useful properties of the delta function are summarized below, given the real constants \(a, b, c,\) and \(A\) and the function \(f(t)\):

1. Scaling properties

\[
\delta\left(\frac{t-c}{b}\right) = |b|\delta(t-c)
\]

\[
\delta(-t) = \delta(t)
\]

\[
\delta[f(t)] = \sum_{k} \frac{\delta(t-t_k)}{|f'(t_k)|} \quad f(t_k) = 0, \quad f' = df/dt
\]
2. Properties in product form

\[ f(t)\delta(t - c) = f(c)\delta(t - c) \]
\[ \delta(t)\delta(t - c) = 0 \quad c \neq 0 \]
\[ \delta(t - c)\delta(t - c) \text{ is not defined} \]

3. Integrals

\[ \int_{-\infty}^{+\infty} A\delta(t - c)dt = A \]
\[ \int_{-\infty}^{+\infty} \delta(t - c) = 1 \]
\[ \int_{-\infty}^{t} \delta(u - c)du = H(t - c) \text{ where } H(t) \text{ is the unit step function:} \]
\[ H(t) = \begin{cases} 
1 & t > 0 \\
1/2 & t = 0 \\
0 & t < 0 
\end{cases} \]

4. Derivatives

\[ \int_{a}^{b} f(t)\delta^{(n)}(t - c) \, dt = (-1)^n f^{(n)}(c) \quad a < c < b, \quad f^{(n)} = \frac{d^nf}{dt^n} \]

5. Fourier Transform pair

\[ 1 = \int_{-\infty}^{+\infty} \delta(t)\exp(i\omega t)dt \]
\[ \delta(t) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} \exp(-i\omega t)d\omega \]
References

Appendix C: Basic Notations and Concepts

This Appendix will give a brief discussion of indicial and vector notation and review some of the basic concepts of motion, mass, and stress/strain needed for our consideration of the equations governing elastic media. Continuum mechanics texts [1–6] are good sources for a more complete description of these topics.

C.1 Indicial Notation

The equations involved in elasticity problems are often algebraically quite complex. The use of Cartesian tensor (index) notation significantly simplifies the expression and manipulation of such equations.

In index notation, the Cartesian coordinates \((x, y, z)\) will be denoted by the subscripted variables \((x_1, x_2, x_3)\) (Fig. C.1) and the corresponding unit base vectors by the vectors \((e_1, e_2, e_3)\). An arbitrary three-dimensional vector \(u\) with Cartesian components \((u_1, u_2, u_3)\) can thus be written as

\[
u = \sum_{i=1}^{3} u_i e_i.
\]

If we adopt the Einstein summation convention, where the appearance twice of the same index symbol implies summation over the range of that index, Eq. (C.1) can be written simply as

\[
u = u_i e_i.
\]

In Eq. (C.2), the repeated index \(i\) is merely a “dummy” index since it is “summed out”. We can replace any such dummy index symbol by any other symbol without
changing the meaning of the equation. Thus, for example, the following expressions are all equally valid:

\[ f = a_p b_p = a_q b_q \]  \hspace{1cm} (C.3a)

\[ \mathbf{u} = u_1 \mathbf{e}_1 = u_k \mathbf{e}_k \]  \hspace{1cm} (C.3b)

\[ w_i = a_{ij} x_j = a_{ik} x_k \hspace{0.5cm} (i = 1, 2, 3) \]  \hspace{1cm} (C.3c)

\[ \sigma_{ij} = C_{ijkl} e_{kl} = C_{ijlm} e_{lm}. \]  \hspace{1cm} (C.3d)

In Eq. (C.3c), the \( j \) and \( k \) indices are dummy indices that assume all values in the range (1, 2, 3) while the \( i \) index is a “free” index which assumes only any one of these values at a time. In some cases, as in Eq. (C.3c), we will show the range of the free indices explicitly. Otherwise, the range will be assumed to be implicitly given by (1, 2, 3). In Eq. (C.3d), for example, both free indices \( (i, j) \) can take on any one of the three values (1, 2, 3).

Quantities with no indices (see \( f \) in Eq. (C.3a)) are scalars, or tensors of order zero, while quantities with only one free index are normally components of vectors, or tensors of order one (see \( a_p, b_p, w_1 \) in Eqs. (C.3b, c)). Similarly, quantities with \( n \) free indices are often components of tensors of order \( n \). For example, \( \sigma_{ij} \) and \( e_{kl} \) are components of tensors of order two, while \( C_{ijkl} \) is a tensor of order four. Whether or not a quantity is a tensor of a particular order, of course, depends on its behavior under coordinate transformation [1].

Several special quantities appear frequently in indicial notation. One of these is the Kronecker delta tensor, \( \delta_{ij} \), defined as

\[ \delta_{ij} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}. \]  \hspace{1cm} (C.4)
Another common quantity is the alternating tensor, $\varepsilon_{ijk}$, defined by

$$
\varepsilon_{ijk} = \begin{cases} 
  +1 & \text{if } (i,j,k) \text{ an even permutation of } (1,2,3) \\
  -1 & \text{if } (i,j,k) \text{ an odd permutation of } (1,2,3) \\
  0 & \text{if any two } (i,j,k) \text{ indices are equal}
\end{cases}
$$

(C.5)

The Kronecker delta tensor and the alternating tensor are related through a useful expression called the $\varepsilon - \delta$ identity given by

$$
\varepsilon_{ijk} \varepsilon_{irs} = \delta_j^p \delta_{ls} - \delta_j^s \delta_{lr},
$$

which can be verified by direct evaluation.

Also, we note that in index notation partial differentiation with respect to an indexed variable is indicated by a comma followed by the index symbol such as, for example,

$$
f_{,k} = \frac{\partial f}{\partial x_k} \quad w_{i,j} = \frac{\partial w_i}{\partial x_j}.
$$

Since the variable name is dropped by this convention, where derivatives of more than one variable occur in the same expression, it is sometimes convenient to indicate which variable is being differentiated by using distinct index symbols. To illustrate, consider the function $f(x_i y_j) = f(x_1, x_2, x_3; y_1, y_2, y_3)$. We write

$$
f_{,k} = \frac{\partial f}{\partial x_k} \quad f_{,K} = \frac{\partial f}{\partial y_K}
$$

so

$$
f_{,iK} = \frac{\partial^2 f}{\partial x_i \partial y_K}
$$

$$
f_{,KK} = \frac{\partial^2 f}{\partial y_K \partial y_K}
$$

$$
f_{,kk} = \frac{\partial^2 f}{\partial x_k \partial x_k}
$$

etc.

Using indicial notation, many of the standard vector operations are easily written:

1. Dot product of two vectors, $\mathbf{u}$ and $\mathbf{v}$

$$
\mathbf{u} \cdot \mathbf{v} = u_i v_i
$$

2. The vector cross product

$$
\text{if } \mathbf{c} = \mathbf{a} \times \mathbf{b} \text{ then } c_i = \varepsilon_{ijk} a_j b_k
$$
3. The vector operator del

\[ \nabla = e_1 \partial ( )/\partial x_1 + e_2 \partial ( )/\partial x_2 + e_3 \partial ( )/\partial x_3 \]

\[ = e_i \partial ( )/\partial x_i = e_i( )_i \]

4. The gradient of a scalar field, \( f \)

\[ \nabla f = f_{,p} e_p \]

5. The divergence of a vector field, \( \mathbf{f} \)

\[ \nabla \cdot \mathbf{f} = f_{i,i} \]

6. The curl of a vector field, \( \mathbf{u} \)

\[
\text{if } q = \nabla \times \mathbf{u} \text{ then } q_i = \epsilon_{ijk} u_{k,j}
\]

7. The Laplacian of a scalar field, \( f \)

\[ \nabla^2 f = \nabla \cdot (\nabla f) = f_{,kk} \]

8. The Laplacian of a vector field, \( \mathbf{u} \)

\[ \nabla^2 \mathbf{u} = u_{k,j} e_k \]

C.2 Integral Theorems

C.2.1 Gauss’ Theorem

We will frequently make use of a fundamental integral theorem, called Gauss’ theorem, which transforms a volume integral into an integral over the surface of that volume. In its most general form, Gauss’ theorem states that for a volume \( V \) bounded by a surface \( S \) and for a differentiable tensor field, \( a_{ij...n} \), of any order we have

\[ \int_V a_{ij...n,k} \ dV = \int_S n_k a_{ij...n} \ dS, \quad (C.7) \]

where \( n_k \) are components of the unit normal vector of the surface \( S \) which points outward from \( V \). Some important special cases of Gauss’ theorem arise from the following specific choices for the general tensor \( a \):
1. \( a = \phi \), a scalar

\[
\int_{V} \phi_{,k} \; dV = \int_{S} \phi \; n_{k} \; dS \tag{C.8}
\]

or, in vector notation

\[
\int_{V} \nabla \phi \; dV = \int_{S} \phi \; n \; dS
\]

2. \( a = v_{i} \), a vector

\[
\int_{V} v_{i,k} \; dV = \int_{S} v_{i} \; n_{k} \; dS \tag{C.9}
\]

3. \( a = v_{i} \), a vector

\[
\int_{V} v_{i,i} \; dV = \int_{S} v_{i} \; n_{i} \; dS \tag{C.10}
\]

or, in vector notation

\[
\int_{V} \nabla \cdot v \; dS = \int_{S} v \cdot n \; dS
\]

(This form of the theorem is also referred to as Green’s theorem, Ostrogradskiy’s theorem, or the divergence theorem)

4. \( a = \tau_{ij} \), a second order tensor

\[
\int_{V} \tau_{ij,k} \; dV = \int_{S} \tau_{ij} \; n_{k} \; dS \tag{C.11}
\]

### C.2.2 Stokes’ Theorem

Another fundamental integral theorem that we will make use of is Stokes’ theorem. This theorem relates the integral over an open surface \( S \) to a corresponding closed contour (line) integral over the edge \( C \) of \( S \) where, if \( n \) is the outward normal to \( S \), the contour is obtained from \( n \) according to the right hand rule. There are also various forms of this theorem. The form which we will use in this book for transducer modeling with boundary diffraction waves (Chap. 8) is
\[ \int_S \left( \nabla \times \mathbf{f} \right) \cdot \mathbf{n} dS = \oint_C \mathbf{f} \cdot d\mathbf{c}. \]  
(C.12)

Several other forms of Stokes’ theorem that are particularly useful in regularizing the hypersingular integral equations appearing in crack problems (see Chap. 5) are

1. For a vector \( f_k \)

\[ \int_S \left[ \left( \partial f_k / \partial x_r \right) n_k - \left( \partial f_k / \partial x_k \right) n_r \right] dS = \oint_C \mathbf{\varepsilon}_{krq} f_k dx_q. \]  
(C.13)

2. For a tensor \( f_{km} \)

\[ \int_S \left[ \left( \partial f_{km} / \partial x_r \right) n_k - \left( \partial f_{km} / \partial x_k \right) n_r \right] dS = \oint_C \mathbf{\varepsilon}_{krq} f_{km} dx_q. \]  
(C.14)

### C.3 Strain and Deformation

Consider the deformation of a body, which occupies a volume \( V \) in an undeformed state, to a new deformed volume \( V' \) (Fig. C.2). If we let \( X \) denote the position vector of an arbitrary point \( P \) in volume \( V \) which moves to point \( P' \) in \( V' \), whose position vector is \( x \), then the displacement vector of the body from \( X \) to \( x \) is given by

\[ \mathbf{u} = \mathbf{x} - \mathbf{X}. \]  
(C.15)

Now, consider another two points \( Q \) and \( Q' \) which are located in the neighborhood of points \( P \) and \( P' \), respectively, as shown. The differential length between these two pairs of neighboring points is

---

**Fig. C.2** Motion of a deformable body
\[ dS = (dX_idX_i)^{1/2} \] (C.16)

in volume \( V \), and

\[ ds = (dx_idx_i)^{1/2} \] (C.17)

in volume \( V' \). Any distortion that occurs between these points during the deformation is given by

\[ ds^2 - dS^2 = dx_idx_i - dX_idX_i = (\delta_{ij} - X_{k,i}X_{k,j})dx_idx_j. \] (C.18)

Thus, this local distortion, or strain, is described by the second order tensor, \( \varepsilon_{ij} \), given by

\[ \varepsilon_{ij} = \frac{1}{2}(\delta_{ij} - X_{k,i}X_{k,j}), \] (C.19)

where the factor of 1/2 is introduced merely for convenience. The quantity \( \varepsilon_{ij} \) is known as the Almanski strain tensor. From Eq. (C.15), the Almanski strain tensor can be rewritten in terms of the displacement components, \( u_i \), as

\[ \varepsilon_{ij} = \frac{1}{2}(u_{i,j} + u_{j,i} - u_{k,i}u_{k,j}). \] (C.20)

In all the applications considered in this book, the displacement gradients will be everywhere small, so to first order we can write \( \varepsilon_{ij} = e_{ij} \) where

\[ e_{ij} = \frac{1}{2}(u_{i,j} + u_{j,i}) \] (C.21)

is the Cauchy strain tensor. Since to first order we also have a change in displacement, \( du_i \), given by

\[ du_i = u_{i,j}dx_j = \frac{1}{2}(u_{i,j} + u_{j,i})dx_j + \frac{1}{2}(u_{i,j} - u_{j,i})dx_j = e_{ij}dx_j + \omega_{ij}dx_j, \] (C.22)

where \( \omega_{ij} = \frac{1}{2}(u_{i,j} - u_{j,i}) \) is the rotation tensor, we see that locally the kinematics of motion can be decomposed into two parts. The first part, \( e_{ij} = \frac{1}{2}(u_{i,j} + u_{j,i}) \) is due to the distortion (strain) occurring while the second part is due to a local rotation, \( \omega_{ij} \).

To see that \( \omega_{ij} \) does indeed represent a local rotation, we note that we may write

\[ \omega_{ij} = -\frac{1}{2}\varepsilon_{ijk}\omega_k \] (C.23)
in terms of the components of a rotation vector, \( \omega_k \), given by
\[
\omega = \nabla \times \mathbf{u}
\]  
(C.24)
so that the total change in displacement \( d\mathbf{u} \) can be written in vector notation, from Eq. (C.22) as
\[
d\mathbf{u} = e \cdot dx + \frac{1}{2} \omega \times dx.
\]  
(C.25)
Since we will assume the displacement (and velocity) gradients of the motion are everywhere small, these gradients will also be neglected when computing the velocity, \( \mathbf{v} \), and the acceleration, \( \mathbf{a} \). For the velocity \( \mathbf{v} \), for example, we have
\[
\mathbf{v}(x,t) = D\mathbf{u}(x,t)/Dt,
\]  
(C.26)
where
\[
D/Dt = \partial / \partial t + \mathbf{v} \cdot \nabla
\]  
(C.27)
is a total material time derivative. Thus for small displacement gradients we have
\[
\mathbf{v}_i(x,t) = \partial \mathbf{u}_i(x,t)/\partial t.
\]  
(C.28)
Similarly, for the acceleration
\[
\mathbf{a}(x,t) = D\mathbf{v}(x,t)/Dt = \partial \mathbf{v}(x,t)/\partial t + [\mathbf{v}(x,t) \cdot \nabla] \mathbf{v}(x,t),
\]  
(C.29)
which for small velocity gradients becomes, in component form
\[
a_i(x,t) = \partial \mathbf{v}_i(x,t)/\partial t = \partial^2 u_i(x,t)/\partial t^2.
\]  
(C.30)

### C.4 Conservation of Mass

If we let \( \rho_0(X) \) be the mass density of the material in some original volume \( V \) at a fixed time \( t = 0 \) (Fig. C.2) and \( \rho(x,t) \) the corresponding density in volume \( V' \) at time \( t \), then by conservation of mass we have
\[
\int_{V'} \rho(x,t)dx_1dx_2dx_3 = \int_V \rho_0(X)dx_1dx_2dx_3
\]
or
\[
\int_{V'} [\rho(x(X),t)J - \rho_0(X)]dx_1dx_2dx_3,
\]  
(C.31)
where \( J = \frac{\partial (x_1, x_2, x_3)}{\partial (X_1, X_2, X_3)} \) is a Jacobian \([1]\) of the transformation from \( V \) to \( V' \). Since volume \( V' \) is arbitrary, we must have

\[
\rho J = \rho_0. \tag{C.32}
\]

For small displacement gradients, however, it can be shown that the Jacobian is approximately

\[
J = (1 + u_{k,k}) \tag{C.33}
\]

so that to first order

\[
\rho = \rho_0 (1 - u_{k,k}) \cong \rho_0. \tag{C.34}
\]

In most of our discussions we will take the original density, \( \rho_0 \), to be independent of \( X \), so that Eq. (C.34) then shows that \( \rho \) is also, to first order, merely a constant.

### C.5 Stress

#### C.5.1 The Traction Vector

Consider a volume \( V \) of material which is acted upon by a set of forces as shown in Fig. C.3a. If we imagine passing a cutting plane with unit normal vector, \( n \), through an arbitrary point \( P \) in this body, then we can separate the body into two parts as shown in Fig. C.3b. For part \( V_- \), on a small area \( \Delta A \) at point \( P \) (on the planar cut exposed by the cutting plane) there will be a net force \( \Delta F \) acting on \( V_- \) due to the
internal forces exerted across the plane by $V_+$. We define the traction vector, $t(n)$, acting at $P$ as

$$ t(n) = \lim_{\Delta A \to 0} \frac{\Delta F}{\Delta A}. \quad (C.35) $$

Note that since an equal and opposite force, $-\Delta F$, acts on an identical area $\Delta A$ for $V_+$ with unit normal $-n$, it follows that

$$ t(-n) = -t(n), \quad (C.36) $$

which is just a statement of Newton’s third law.

### C.5.2 Concept of Stress

Now, consider in particular the traction vectors, $t_k \ (k = 1, 2, 3)$, acting on the faces of the element shown in Fig. C.4 whose sides are parallel to the Cartesian coordinate axes. We can decompose these traction vectors into components $\tau_{kl}$ along the coordinate axes, i.e. $t_k = \tau_{kl} e_l$ where $\tau_{kl}$ are called stresses. For example, for $t_1$ we have the components $(\tau_{11}, \tau_{12}, \tau_{13})$ as shown in Fig. C.5. Physically, the stress $\tau_{kl}$ represents the force/unit area in the $l$th direction acting on a plane with unit normal $e_k$.

### C.5.3 Tractions and Stresses

Suppose at any point $P$ in a body we remove a small tetrahedron-shaped volume $dV$ having three sides of areas $dA_k$ parallel to a set of Cartesian axes and a front face of area $dA$ and unit normal $n$ (Fig. C.6). If we let $\rho$ be the density of the material and $f$ the force/unit volume acting on this element, then from Newton’s second law we have

---

**Fig. C.4** Traction vectors on the faces of a small element whose sides are parallel to the Cartesian axes (negative tractions acting on the back faces are not shown)
Dividing by $dA$ and noting that the components, $n_k$ of the unit vector normal to $dA$, $\mathbf{n} = n_k \mathbf{e}_k$, are given by $n_k = dA_k / dA$ we find

$$\mathbf{t}(\mathbf{n}) dA - t_k dA_k + f dV = \rho \mathbf{a} dV. \quad (C.37)$$

As $dA$ and $dV \to 0$, however, $dV/dA \to 0$ so that in the limit we have

$$\mathbf{t}(\mathbf{n}) = t_k n_k \quad (C.38)$$

or

$$\mathbf{t}(\mathbf{n}) = \tau_{kl} n_k \mathbf{e}_l. \quad (C.39)$$
Equation (C.39) shows that a knowledge of the stress tensor, $\tau_{kl}$, at any point $P$ determines the traction vector at $P$ for any cutting plane through $P$. For this reason, we say that the stress components $\tau_{kl}$ completely determine the state of stress at point $P$.

References

Appendix D: The Hilbert Transform

The Hilbert transform of a function often occurs in signal processing applications and in reflection and refraction problems when critical angles are exceeded (see Chap. 6). It also appears frequently in integral transform theory [1]. Here, we define the Hilbert transform of the function $f(t)$ as:

$$f_H(u) = \mathcal{H}[f(t); u] = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{f(t)dt}{t-u}.$$  \hspace{1cm} (D.1)

where the integral is interpreted in the principal value sense, i.e.

$$\lim_{\varepsilon \to 0} \left( \int_{u-\varepsilon}^{u+\varepsilon} + \int_{-\infty}^{u-\varepsilon} \right) \frac{f(t)dt}{t-u}.$$  \hspace{1cm} (D.2)

As with the Fourier transform, there is an inverse Hilbert transform that can recover $f(t)$ given by

$$f(t) = -\frac{1}{\pi} \int_{-\infty}^{\infty} \frac{f_H(u)du}{u-t}.$$  \hspace{1cm} (D.3)

D.1 Properties of the Hilbert Transform

Some of the important properties of the Hilbert transform are given by

1. $\mathcal{H}[f(a + t); u] = \mathcal{H}[f(t); u + a]$  \hspace{1cm} (D.4a)
Hilbert transforms for some commonly occurring functions \( f(t) \):

\[
\begin{align*}
\delta(t) & : \\ H(t - t_0) & : -\frac{1}{\pi u} \\
H(t - b) - H(t - a) & : (1/\pi)\ln\frac{u - b}{u - a} \\
P_v^{-1}H(t) & : 0 < \text{Re} \nu < 1 \\
\sin(a\sqrt{t})H(t) & : a > 0 \\
\cos(\omega t) & : \omega > 0 \\
\sin(\omega t) & : \omega > 0 \\
\frac{t}{t^2 + a^2} & : \text{Re}a > 0 \\
\frac{a}{t^2 + a^2} & : \text{Re}a > 0
\end{align*}
\]

References

Appendix E: The Method of Stationary Phase

In wave propagation and scattering models, obtaining explicit analytical results in many cases is made difficult by the complex interactions present. Usually, expressions can only be reduced to the form of integrals which then must be evaluated numerically or approximately. This Appendix will consider a method of approximation that is particularly useful called the method of stationary phase \([4, 5]\). Basically the method relies on the fact that at high frequencies, where the phase term in an integrand can be rapidly varying, only certain regions contribute significantly to the integral. Assuming high frequencies is a relatively mild assumption in many NDE problems since the characteristic lengths involved in most applications are many wavelengths long. Thus, the method of stationary phase can often be applied without placing severe restrictions on the validity of the results.

E.1 Single Integral Forms

Consider the case of a one-dimensional integral of the form

\[
I = \int_a^b f(x) \exp[i k \phi(x)] \, dx,
\]

(E.1)

where the wave number, \(k\), is assumed to be large and \(f(x)\) is a smoothly varying function. Since the phase of the complex exponential will vary rapidly in this case, with each half period oscillation nearly canceling the adjacent half period oscillation of opposite sign, we expect that \(I \to 0\) as \(k \to \infty\). Under such circumstances the major contribution of the integral will come from points where the phase term \(\phi(x)\) is stationary, i.e. \(\phi'(x) = d\phi(x)/dx = 0\). To see this, consider the following integral:
\[ I_p = \int_0^4 (4-x) \exp \left[ ik \left( x - \frac{x^2}{2} \right)^2 \right] dx. \]

Figures E.1(a),(b) show the behavior of the real part of this integrand for \( k = 20 \) and \( k = 40 \), respectively. One can see from the behavior of the integrand that most oscillations are canceling except near \( x = 2 \) which is a stationary point for the phase term \( \phi(x) = (x - 2)^2 \).

Assume now, that the integral in Eq. (E.1) has one stationary point within the open interval \((a, b)\) at the point \( x_s \) where \( \phi'(x_s) = 0 \). Then, expanding the phase \( \phi(x) \) in a Taylor series about this point gives

\[ \phi(x) \approx \phi(x_s) + \phi''(x_s) s^2 / 2, \quad (E.2) \]

where \( \phi''(x_s) = d^2 \phi / dx^2 \) and \( s = x - x_s \) and where we have assumed \( \phi''(x_s) \neq 0 \). Since \( f \) is assumed to be slowly varying, it is nearly a constant and can be taken out of the integrand, together with the constant phase part of the exponential, and the major contribution of the integrand remaining comes from the region \((x_s - \epsilon, x_s + \epsilon)\) about the stationary phase point. Thus, we have approximately

\[ I = f(x_s) \exp \left[ ik \phi(x_s) \right] \int_{s=-\epsilon}^{s=+\epsilon} \exp \left[ ik \phi''(x_s) s^2 / 2 \right] ds. \quad (E.3) \]

However, because of the highly oscillatory, constant amplitude nature of the integrand in Eq. (E.3), we can replace the upper and lower limits by \(+\infty\) and \(-\infty\) without introducing significant error to obtain

\[ I = f(x_s) \exp \left[ ik \phi(x_s) \right] \int_{-\infty}^{+\infty} \exp \left[ ik \phi''(x_s) s^2 / 2 \right] ds. \quad (E.4) \]
Now, if we let $k|\phi''(x_s)|s^2/2 = r^2$, it follows that $ds = \left[2/k|\phi''(x_s)|\right]^{1/2}dt$ and the stationary phase approximation of the integral, $I_s$, is

$$I_s = f(x_s)\left[2/k|\phi''(x_s)|\right]^{1/2}\exp\left[ik\phi(x_s)\right]\int_{-\infty}^{+\infty} \exp\left[i\frac{r^2}{2}\text{sgn}\left\{\phi''(x_s)\right\}\right] dt.$$  \hspace{1cm} (E.5)

The integral in Eq. (E.5) can be performed exactly since we have [1]

$$\int_{-\infty}^{+\infty} \cos \left(\frac{r^2}{2}\right) dt = \sqrt{\pi}/2$$  \hspace{1cm} (E.6)

and

$$\int_{-\infty}^{+\infty} \sin \left(\frac{r^2}{2}\right) dt = \sqrt{\pi}/2,$$

so that

$$\int_{-\infty}^{+\infty} \exp\left[i\frac{r^2}{2}\text{sgn}\left\{\phi''(x_s)\right\}\right] dt = \left[1 + is\text{gn}\left\{\phi''(x_s)\right\}\right]\sqrt{\pi}/2$$

$$\hspace{0.5cm} = \sqrt{\pi}\exp\left[i\text{sgn}\left\{\phi''(x_s)\right\}/4\right]$$  \hspace{1cm} (E.7)

and the integral becomes, finally

$$I_s = \left[2\pi/k\phi''(x_s)\right]^{1/2} f(x_s)\exp\left[ik\phi(x_s) + i\pi\text{sgn}\left\{\phi''(x_s)\right\}/4\right].$$  \hspace{1cm} (E.8)

Equation (E.8) gives an explicit expression for the stationary phase evaluation of the original integral when a single isolated stationary phase point exists in the open interval $(a, b)$. Special cases when multiple stationary phase points exist, or when the stationary phase point(s) are at the interval end points, and cases where $\phi'' = 0$ at the stationary point can also be considered (see [1] for further details). However, for the specific problems covered in this book, Eq. (E.8) will generally suffice.

The only special case not covered by Eq. (E.8) we would like to consider is when there is no stationary phase point at all in the closed interval $[a, b]$. Then we can write $I$ in the form

$$I = \int_{a}^{b} \left[\frac{f(x)}{ik\phi'(x)}\right]\exp\{ik\phi(x)\}ik\phi'(x)dx,$$  \hspace{1cm} (E.9)

where the second term in square brackets in Eq. (E.9) is a perfect differential so that an integration by parts gives
\[ I = \frac{f(x)\exp[ik\phi(x)]}{ik\phi'(x)} \bigg|_a^b - \frac{1}{ik} \left[ f(x) / \phi'(x) \right]' \exp[ik\phi(x)] \, dx. \tag{E.10} \]

Since \( \phi'(x) \neq 0 \) in \([a, b]\) we can again write the integral in Eq. (E.10) as the product of two terms, one being a perfect differential, and perform the integration by parts again. Thus, keeping only the first terms, for \( k \) large we have

\[ I = \frac{f(b)\exp(ikb)}{ik\phi'(b)} - \frac{f(a)\exp(ika)}{ik\phi'(a)} + O\left(\frac{1}{k^2}\right). \tag{E.11} \]

### E.2 Double Integral Forms

In some problems we will encounter two-dimensional integrals similar to Eq. (E.1), e.g.

\[ I = \int_{a_1}^{b_1} \int_{a_2}^{b_2} f(x, y)\exp[ik\phi(x, y)] \, dx \, dy, \tag{E.12} \]

where again \( k \) is large and the function \( f \) is assumed to be smoothly varying. We will assume a stationary phase point exists at the point \( \mathbf{x}_s = (x_s, y_s) \) inside the limits of the integrals. Then \( \partial \phi(x_s) / \partial x = \partial \phi(x_s) / \partial y = 0 \) and we can expand \( \phi \) in a Taylor series to second order as

\[ \phi(x, y) = \phi(x_s, y_s) + \frac{1}{2} \left\{ \phi_{xx}^s (x - x_s)^2 + 2\phi_{xy}^s (x - x_s)(y - y_s) + \phi_{yy}^s (y - y_s)^2 \right\}, \tag{E.13} \]

where \( \phi_{xy}^s = \partial^2 \phi(x_s) / \partial x \partial y \), etc. If we change to new coordinates \((u, v)\), where

\[\begin{align*}
u &= (y - y_s) + \phi_{xy}^s (x - x_s) / \phi_{yy}^s \\
u &= (y - y_s) + \phi_{xy}^s (x - x_s) / \phi_{yy}^s \end{align*} \tag{E.14}\]

then \( dx \, dy = du \, dv \) since the Jacobian, \( J \), of the transformation is given by

\[ J = \begin{vmatrix}
\frac{\partial u}{\partial x} & \frac{\partial u}{\partial y} \\
\frac{\partial v}{\partial x} & \frac{\partial v}{\partial y}
\end{vmatrix} = \begin{vmatrix}
1 & 0 \\
\phi_{xy}^s / \phi_{yy}^s & 1
\end{vmatrix}. \tag{E.15} \]
Also, as can be verified, via the chain rule,

\[ \phi_s(x - x_s)^2 + 2\phi_{,xy}(x - x_s)(y - y_s) + \phi_s(y - y_s)^2 = \phi_{,uu}u^2 + 2\phi_{,uv}uv + \phi_{,vv}v^2 \]  

(E.16)

and

\[ \phi_{,vv} = \phi_{,yy} \]
\[ \phi_{,uv} = 0 \]
\[ \phi_{,u} = \phi_{,xx} - \left(\phi_{,xy}\right)^2 / \phi_{,yy} \]  

(E.17)

so that the original integral, in the \( u, v \) coordinates can be approximated, as in the one-dimensional case, as

\[ I_s = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f(x_s, y_s) \exp\left[ik\phi(x_s, y_s)\right] \exp[\pmik(\phi_{,uu}u^2 + \phi_{,vv}v^2)] \, du \, dv. \]  

(E.18)

Since the \( u \) and \( v \) integrals in Eq. (E.18) are identical to the forms we evaluated previously in the one-dimensional case (see Eq. (E.4)), we can obtain immediately

\[ I_s = \frac{2\pi}{\left[k|\phi_{,uu}|\right]^{1/2} \left[k|\phi_{,vv}|\right]^{1/2}} f(x_s, y_s) \exp\left[\pmik\phi(x_s, y_s)\right] + i\pi(\text{sgn}\{\phi_{,uu}\} + \text{sgn}\{\phi_{,vv}\}) / 4, \]  

(E.19)

which can be written in terms of the original \( x \) and \( y \) coordinate values by defining

\[ H = \phi_{,xx}\phi_{,yy} = \phi_{,xx}\phi_{,yy} - \left(\phi_{,xy}\right)^2 \]  

(E.20)

and noting that

\[ \sigma = \begin{cases} 0 & \text{if } H < 0 \\ 2 & \text{if } H > 0, \phi_{,yy} > 0 \\ -2 & \text{if } H > 0, \phi_{,yy} < 0 \end{cases} \]  

(E.21)

so that we have, finally

\[ I_s = \frac{2\pi}{k} \frac{f(x_s, y_s)}{\sqrt{|\phi_{,xx}\phi_{,yy} - (\phi_{,xy})^2|}} \exp[\pmik\phi(x_s, y_s) + i\pi\sigma / 4]. \]  

(E.22)
E.3 Curved Surface Integral

The last case that we will consider by the method of stationary phase is the evaluation of a particular integral over a curved surface given by

\[
I = \int_S f(\mathbf{x}) \exp[ik(\mathbf{a} \cdot \mathbf{x})]dS(\mathbf{x}),
\]

where the surface \(S\) will be assumed to be smooth, but otherwise arbitrary, and the vector \(\mathbf{a}\), is a constant. Again, we wish to evaluate this integral at high frequencies, i.e. when \(k\) is large. In this case, we have taken the phase term to be of the particular form \(\phi(x) = \mathbf{a} \cdot \mathbf{x}\) which will occur in a number of applications.

As in the previous cases, we will assume the major contribution to this integral comes from the evaluation near a stationary phase point \(x_s\) on the surface. We can parameterize \(S\) in terms of two surface coordinates \((t_1, t_2)\) so that on \(S\) we can write \(x = x(t_1, t_2)\). Then in this case the stationary phase conditions become

\[
\frac{\partial \phi}{\partial t_1} = \frac{\partial \phi}{\partial t_2} = 0
\]

or, explicitly

\[
\mathbf{a} \cdot \frac{\partial \mathbf{x}}{\partial t_1} = 0
\]

\[
\mathbf{a} \cdot \frac{\partial \mathbf{x}}{\partial t_2} = 0.
\]

Since the vectors \(\frac{\partial \mathbf{x}}{\partial t_{\alpha}} (\alpha = 1, 2)\) are tangent to the surface, it follows from stationary phase that at the stationary phase point \(x_s = (b_1, b_2)\), \(\mathbf{a}\) is parallel to the unit normal, \(\mathbf{n}\), which is taken here to be the inward normal. Then, expanding \(\phi\) in a Taylor series about the stationary phase point as before, we obtain in this case

\[
\phi = \mathbf{a} \cdot \mathbf{x}(t_1, t_2) = \mathbf{a} \cdot \mathbf{x}_s + \frac{1}{2} \{ (t_\alpha - b_\alpha)(t_\beta - b_\beta) \mathbf{a} \cdot \mathbf{x}_{s,\alpha\beta} \},
\]

where \(\mathbf{x}_{s,\alpha\beta} = \frac{\partial^2 \mathbf{x}}{\partial t_{\alpha}\partial t_{\beta}}\). Then, the integral can be written approximately as

\[
I = f(\mathbf{x}_s) \exp[ik(\mathbf{a} \cdot \mathbf{x}_s)] J(\mathbf{x}_s) \int_{b_2+\epsilon_2}^{b_2} \int_{b_1+\epsilon_1}^{b_1} \exp \left[ \frac{1}{2} k \{ (t_\alpha - b_\alpha)(t_\beta - b_\beta) \mathbf{a} \cdot \mathbf{x}_{s,\alpha\beta} \} \right] dt_1 dt_2,
\]

where \(dS = Jdt_1 dt_2\) is the area element in the \((t_1, t_2)\) coordinates and \(J\) is a Jacobian. From differential geometry [3] we have

\[
\mathbf{x}_{s,\alpha\beta} = I^{\alpha\beta}_{\alpha\beta}(\mathbf{x}_s) \mathbf{x}_{s,\alpha}^{\alpha} + h_{\alpha\beta}(\mathbf{x}_s) \mathbf{n}^\alpha(\mathbf{x}_s).
\]
where $\Gamma^\alpha_{\beta\gamma}$ are Christoffel’s symbols of the second kind, $h_{\alpha\beta}$ is the curvature tensor, and $x^\gamma = \partial x(b_1, b_2)/\partial t_\gamma$. The dot product of $a$ with these second order partial derivatives of the position vector can be expressed in general as

$$a \cdot x^\gamma_{,\alpha\beta} = \Gamma^\gamma_{,\alpha\beta}(x) \left( a \cdot x^\gamma_{,\gamma} \right) + h_{\alpha\beta}(x) \left( a \cdot n'(x) \right). \quad (E.28)$$

However, from the stationary phase conditions the first term on the right side of Eq. (E.27) vanishes, leaving

$$a \cdot x^\gamma_{,\alpha\beta} = h_{\alpha\beta}(x) \left( a \cdot n'(x) \right). \quad (E.29)$$

If $t_1$ and $t_2$ are chosen in particular to be the arc length parameters taken along the principal directions of the surface at $x_s$, then

$$h_{12}(x_s) = h_{21}(x_s) = 0$$
$$h_{11}(x_s) = \kappa_1$$
$$h_{22}(x_s) = \kappa_2$$

and $J(x_s) = 1$ where $\kappa_1$ and $\kappa_2$ are the principal curvatures of the surface at $x_s$ and the stationary phase contribution to the integral becomes (expanding the limits of integration, as before to infinity):

$$I_s = f(x_s) \exp[i k a \cdot x_s] \times \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \exp \left[ \frac{1}{2} k a \cdot n'(x_s) \left\{ \kappa_1 (t_1 - b_1)^2 + \kappa_2 (t_2 - b_2)^2 \right\} \right] dt_1 dt_2. \quad (E.31)$$

With this choice of coordinates the integrals appearing in Eq. (E.31) can be independently evaluated since they are in the same general form as found in the one- and two-dimensional integral cases considered previously. Thus, we find

$$I_s = \left[ \frac{2\pi}{k|\kappa_1 a \cdot n'(x_s)|} \right]^{1/2} \left[ \frac{2\pi}{k|\kappa_2 a \cdot n'(x_s)|} \right]^{1/2} f(x_s) \exp[i k a \cdot x_s + i\pi\sigma/4], \quad (E.32)$$

where

$$\sigma = \left[ \text{sgn}\{\kappa_1 a \cdot n'(x_s)\} + \text{sgn}\{\kappa_2 a \cdot n'(x_s)\} \right]. \quad (E.33)$$

If we let

$$|\kappa_1| = 1/R_1$$
$$|\kappa_2| = 1/R_2,$$  

(E.34)
where $R_1$ and $R_2$ are the magnitudes of the principal radii of curvature of $S$ at $x_s$, we have

\[ I_s = \frac{2\pi \sqrt{R_1 R_2}}{k|\mathbf{a} \cdot \mathbf{n}(x_s)|} f(x_s) \exp[i k \mathbf{a} \cdot \mathbf{x}_s + i\pi\sigma/4]. \] (E.35)

References

Appendix F: Properties of Ellipsoids

F.1 Geometry of an Ellipsoid

In Chap. 10, it was shown that the far field scattering response of ellipsoidal inclusions and elliptical cracks can be obtained explicitly through the use of the Born and Kirchhoff approximations. These same approximations were used in Chap. 15 to develop a number of equivalent flaw sizing algorithms. In this Appendix, we will derive some of the geometrical properties of ellipsoids that are useful in such applications. The details discussed here can also be typically found in texts on tensor analysis and differential geometry [1–3].

One geometrical quantity that appears frequently in Chaps. 10 and 15 is the effective radius of an ellipsoid, \( r_e \), which was defined to be the perpendicular distance from the center of an ellipsoid to a plane \( P \) which is tangent to its surface at some point, \( x^0 \), as shown in Fig. F.1. If we write the equation of the ellipsoid as

\[
g(x_1, x_2, x_3) = \frac{x_1^2}{a_1^2} + \frac{x_2^2}{a_2^2} + \frac{x_3^2}{a_3^2} - 1 = 0, \tag{F.1}
\]

then since the unit outward normal, \( n \), is given by \( n = \nabla g / |\nabla g| \) its components are

\[
n_1 = \frac{x_1}{Ha_1^2}, \quad n_2 = \frac{x_2}{Ha_2^2}, \quad n_3 = \frac{x_3}{Ha_3^2} \tag{F.2}
\]

with

\[
H = \sqrt{\frac{x_1^2}{a_1^4} + \frac{x_2^2}{a_2^4} + \frac{x_3^2}{a_3^4}} \tag{F.3}
\]

If a plane \( P \), whose unit normal is \( e \), touches and is tangent to the ellipsoid at the point \( x^0 = (x_1^0, x_2^0, x_3^0) \), then \( e \) coincides with the unit normal at this point, i.e. \( e = n(x_1^0, x_2^0, x_3^0) = n^0 \) and for any point \( X \) in this plane we may write
which is the general equation for a plane whose unit normal is \( \mathbf{n}^0 = (n_{10}, n_{20}, n_{30}) \) and whose perpendicular distance from the origin is the “effective radius”, \( r_e \). Since point \( \mathbf{x}^0 = (x_{10}, x_{20}, x_{30}) \) also lies in the plane \( P \), for this point Eq. (F.4) gives

\[
n_{10}^0 x_{10} + n_{20}^0 x_{20} + n_{30}^0 x_{30} = r_e \quad \text{(F.5)}
\]

or, from Eqs. (F.1) and (F.2)

\[
\frac{(x_{10})^2}{a_1^2} + \frac{(x_{20})^2}{a_2^2} + \frac{(x_{30})^2}{a_3^2} = H^0 r_e = 1 \quad \text{(F.6)}
\]

so that \( r_e \) is given by

\[
r_e = \frac{1}{H^0} = \sqrt{\frac{(x_{10})^2}{a_1^2} + \frac{(x_{20})^2}{a_2^2} + \frac{(x_{30})^2}{a_3^2}}. \quad \text{(F.7)}
\]

To obtain a more useful form for \( r_e \), consider \( a_1^2(n_{10}^0)^2 + a_2^2(n_{20}^0)^2 + a_3^2(n_{30}^0)^2 \). From Eqs. (F.1), (F.2), and (F.7)

\[
a_1^2(n_{10}^0)^2 + a_2^2(n_{20}^0)^2 + a_3^2(n_{30}^0)^2 = \frac{(x_{10})^2}{a_1^2} + \frac{(x_{20})^2}{a_2^2} + \frac{(x_{30})^2}{a_3^2} = \frac{1}{(H^0)^2} \quad \text{(F.8)}
\]

so that we have, finally

\[
r_e = \sqrt{a_1^2(n_{10}^0)^2 + a_2^2(n_{20}^0)^2 + a_3^2(n_{30}^0)^2}
= \sqrt{a_1^2(e_i \cdot u_1)^2 + a_2^2(e_i \cdot u_2)^2 + a_3^2(e_i \cdot u_3)^2}, \quad \text{(F.9)}
\]
where \((\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3)\) are unit vectors along the axes of the ellipsoid (Fig. F.1). Thus, if we are given the normal to the plane, \(\mathbf{e}_i\), and the size and orientation of the ellipsoid through the parameters \((a_1, a_2, a_3)\) and \((\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3)\), respectively, Eq. (F.9) determines the effective radius. Also, the point where the plane touches the ellipsoid can then be obtained, since

\[
\begin{align*}
    x_1^0 &= a_1^2 n_1^0 / r_e \\
    x_2^0 &= a_2^2 n_2^0 / r_e \\
    x_3^0 &= a_3^2 n_3^0 / r_e.
\end{align*}
\]

Finally, it is interesting to note that Eqs. (F.2), (F.4), and (F.6) also imply that an equivalent form for the plane \(P\) is given by

\[
\frac{x_1^0 X_1}{a_1^2} + \frac{x_2^0 X_2}{a_2^2} + \frac{x_3^0 X_3}{a_3^2} = 1.
\]

which is a result commonly derived in differential geometry texts.

As shown in Chap. 15, measurements of an effective radius of an unknown flaw from different directions allows one to use Eq. (F.9) to obtain the equivalent ellipsoidal size and orientation parameters for that flaw. Thus, Eq. (F.9) is the key geometrical relationship for performing equivalent flaw sizing.

Another geometrical parameter that appeared in the leading edge response of ellipsoids in Chap. 10 was the Gaussian curvature of the ellipsoid. To obtain an expression for this curvature, we parameterize the surface of the ellipsoid by two parameters \((t_1, t_2)\) as follows:

\[
\begin{align*}
    x_1 &= a_1 \cos t_1 \cos t_2 \\
    x_2 &= a_2 \cos t_1 \sin t_2 \\
    x_3 &= a_3 \sin t_1.
\end{align*}
\]

Then, any point \(\mathbf{x}\) on the ellipsoid is given by

\[
\mathbf{x} = a_1 \cos t_1 \cos t_2 \ \mathbf{u}_1 + a_2 \cos t_1 \sin t_2 \ \mathbf{u}_2 + a_3 \sin t_1 \ \mathbf{u}_3.
\]

From differential geometry the mean and Gaussian curvatures, \(M\) and \(K\), respectively, of a surface are given by [1]

\[
\begin{align*}
    M &= \frac{g_{11} h_{22} + g_{22} h_{11} - 2 g_{12} h_{12}}{2|\mathbf{x}_1 \times \mathbf{x}_2|^2} \\
    K &= \frac{h_{11} h_{22} - h_{12}^2}{|\mathbf{x}_1 \times \mathbf{x}_2|^2},
\end{align*}
\]
where \( g_{\alpha\beta} \) and \( h_{\alpha\beta} \) are the first and second fundamental forms

\[
g_{\alpha\beta} = \mathbf{x}_\alpha \cdot \mathbf{x}_\beta \\
h_{\alpha\beta} = \mathbf{n}' \cdot \mathbf{x}_\alpha 
\]  

(F.15)

and the comma denotes partial differentiation with respect to the \( t \) parameters, i.e. \( \mathbf{x}_\alpha = \frac{\partial \mathbf{x}}{\partial t_\alpha} \), etc. The vector \( \mathbf{n}' \) is the inward unit normal to the ellipsoid which is given parametrically from Eqs. (F.2), (F.7), and (F.12) as

\[
\mathbf{n}' = \frac{r_e}{a_1} \cos t_1 \cos t_2 \; \mathbf{u}_1 - \frac{r_e}{a_2} \cos t_1 \sin t_2 \; \mathbf{u}_2 - \frac{r_e}{a_3} \sin t_1 \; \mathbf{u}_3. 
\]  

(F.16)

Placing the \( \mathbf{x} \) and \( \mathbf{n}' \) expressions into Eq. (F.15), we obtain explicitly

\[
g_{11} = \mathbf{x}_1 \cdot \mathbf{x}_1 = a_1^2 \sin^2 t_1 \cos^2 t_2 + a_2^2 \sin^2 t_1 \sin^2 t_2 + a_3^2 \cos^2 t_1 \\
g_{12} = \mathbf{x}_1 \cdot \mathbf{x}_2 = (a_1^2 - a_2^2) \sin t_1 \cos t_1 \sin t_2 \cos t_2 \\
g_{22} = \mathbf{x}_2 \cdot \mathbf{x}_2 = (a_1^2 \sin^2 t_2 + a_2^2 \cos^2 t_2) \cos^2 t_1 \\
h_{11} = \mathbf{n}' \cdot \mathbf{x}_{11} = r_e \\
h_{12} = \mathbf{n}' \cdot \mathbf{x}_{12} = 0 \\
h_{22} = \mathbf{n}' \cdot \mathbf{x}_{22} = r_e \cos^2 t_1 
\]  

(F.17)

from which the mean and Gaussian curvatures can be computed at \( \mathbf{x}^0 \) from Eq. (F.14) as

\[
M = \frac{\kappa_1 + \kappa_2}{2} = \frac{r_e}{2a_1 a_2 a_3} \left\{ \frac{a_1^2 (a_2^2 + a_3^2) (\mathbf{e}_1 \cdot \mathbf{u}_1)^2}{2} + \frac{a_2^2 (a_1^2 + a_3^2) (\mathbf{e}_1 \cdot \mathbf{u}_1)^2}{2} + \frac{a_3^2 (a_1^2 + a_2^2) (\mathbf{e}_1 \cdot \mathbf{u}_1)^2}{2} \right\} 
\]  

(F.18)

and

\[
K = \kappa_1 \kappa_2 = \frac{1}{R_1 R_2} = \frac{r_e^4}{a_1^2 a_2^2 a_3^2}, 
\]  

(F.19)

where \( \kappa_1 \) and \( \kappa_2 \) are the principal curvatures of the ellipsoid and \( R_1 \) and \( R_2 \) are the corresponding principal radii of curvature. Just as the Gaussian curvature appears in the first order leading edge response of volumetric flaws, Chen [4] has shown that the mean curvature also appears in a higher order expansion of this leading edge expression.
References

Appendix G: Matlab Functions and Scripts

In this second edition, MATLAB® functions and scripts have been included to implement some of the ultrasonic models we have considered. This Appendix briefly describes those functions and scripts. Additional MATLAB models for a variety of ultrasonic NDE problems are also available in two related books: L.W. Schmerr and S.J. Song, Ultrasonic Nondestructive Evaluation Systems – Models and Measurements, Springer, 2007, and L.W. Schmerr, Fundamentals of Ultrasonic Phased Arrays, Springer, Springer, 2015. The MATLAB code listings for this book are available on the web from the publisher or they can be obtained directly by sending an email with the subject title “Ultrasonic NDE Codes” to the author at lschmerr@iastate.edu.

G.1 Transmission and Reflection of Plane Waves

\texttt{fluid\_fluid}

\begin{verbatim}
>> [R, T] = fluid\_fluid(iangd, d1, d2, c1, c2). A function which returns (R, T) the reflection and transmission coefficients (based on velocity ratios) for a plane wave incident on a plane interface between two fluids at an angle iangd (in degrees). The parameters (d1, c1) and (d2, c2) are the densities and wave speeds in the first and second fluids. The units of these quantities are arbitrary but must be consistent.
\end{verbatim}

\texttt{fluid\_solid}

\begin{verbatim}
>> [tpp, tps] = fluid\_solid(iangd, d1, d2, cp1, cp2, cs2). A function which returns (tpp, tps) the plane wave transmission coefficients for a P-wave and S-wave, respectively, produced in a solid at a fluid-solid interface due to a P-wave incident in the fluid at an angle iangd (in degrees). The coefficients are both based on velocity ratios. The parameters (d1, cp1) are the density and wave speed of the fluid and (d2,
cp2, cs2) are the density, P-wave speed, and S-wave speed, respectively. The units of these quantities are arbitrary but must be consistent.

**solid_f_solid**

\[
[tpp, tps] = \text{solid}_f_{\text{solid}}(\text{iangd}, d_1, d_2, c_{p1}, c_{s1}, c_{p2}, c_{s2}).
\]

A function which returns the P-P(tpp) and P-S(tps) transmission coefficients, based on velocity ratios, for a P-wave incident on a plane interface between two solids that are in smooth contact through an intermediate fluid layer of zero thickness. The parameters \(d_1, c_{p1}, c_{s1}\) are the density, P-wave speed, and S-wave speed for the first solid while \(d_2, c_{p2}, c_{s2}\) are the corresponding parameters for the second solid. The units of these parameters are arbitrary but must be consistent.

**solid_solid**

\[
[tp, ts, rp, rs] = \text{solid}_s_{\text{solid}}(\text{iangd}, d_1, d_2, c_{p1}, c_{s1}, c_{p2}, c_{s2}, \text{type}).
\]

A function which returns the transmitted P-wave (tp), transmitted SV-wave (ts), reflected P-wave (rp), and reflected SV-wave (rs) transmission/reflection coefficients (based on velocity ratios) for two solids in welded contact. The inputs are the incident angle (s), iangd, (in degrees), \((d_1, d_2)\), the densities of the two media, \((c_{p1}, c_{s1})\), the compressional and shear wave speeds of the first medium, and \((c_{p2}, c_{s2})\) the compressional and shear wave speeds of the second medium. The parameter type is a string, \(\{'P'\ or \'S'\}\), which indicates the type of incident wave in medium one. If \(c_{s1} = 0\) and \(\text{type} = 'P'\) the function returns the coefficients for a fluid-solid interface with \(rs = 0\). The wave speed \(c_{s2}\) cannot be set equal to zero.

**stress_freeP**

\[
[Rp, Rs] = \text{stress}_f_{\text{reeP}}(\text{ang}, c_p, c_s).
\]

A function which returns \((Rp, Rs)\), the reflection coefficients for the reflected P-waves and S-waves, respectively, (based on velocity ratios) for a plane P-wave incident on a plane stress-free surface of an elastic solid. The parameter \(\text{ang}\) is the incident angle (in degrees), \(c_p, c_s\) are the P- and S-wave speeds, respectively, and. The units of the wave speeds are arbitrary but must be consistent. Note that there are no critical angles for this case.

**snells_law**

\[
[\text{ang}_\text{in}, \text{ang}_\text{out}] = \text{snells}_\text{law}(\text{ang}, c_1, c_2, \text{type}).
\]

A function which returns \((\text{ang}_\text{in}, \text{ang}_\text{out})\) the incident and refracted angle at a plane interface, respectively, (in degrees) that satisfy Snell’s law. The first input, \(\text{ang}\), is a given incident or refracted angle, \(\text{ang}\), (in degrees) that a travelling wave makes with respect to the normal of a plane interface. The wave speed in the first medium is \(c_1\) and the wave speed in the second medium is \(c_2\). The input parameter, \(\text{type}\), is either the string ‘f’ or the string ‘r’, indicating a forward or reverse,
problem, respectively. For a forward problem \( \text{ang} \) is taken as the incident angle and the refracted angle is calculated. For a reverse problem \( \text{ang} \) is taken to be the refracted angle and the incident angle is calculated. The angle argument, \( \text{ang} \), must be in the range \( 0 < \text{ang} < 90 \) degrees.

G.2 Surface and Plate Waves

**Rayleigh_speed**

\[
\text{>> } \text{cr} = \text{Rayleigh\_speed} (\text{cp}, \text{cs}) .
\]

A function which returns the Rayleigh wave speed, \( \text{cr} \), at a free surface of a solid where \( \text{(cp, cs)} \) are the P-wave and S-wave speeds, respectively. The units of the wave speeds are arbitrary but must be consistent.

**dispersion\_plots**

\[
\text{>> } \text{dispersion\_plots}.
\]

A script which creates a 2-D set of normalized frequency and wave speed values at which the Rayleigh-Lamb dispersion function for symmetrical or anti-symmetrical plate waves is evaluated. The MATLAB function contour is then used to plot the dispersion curves for this 2-D region. Note that the fundamental anti-symmetrical mode plot is unreliable at very small frequencies and must be replaced by the analytical results for this mode at those values.

**Rayleigh\_LambM**

\[
\text{>> } \text{y} = \text{Rayleigh\_LambM} (\text{c}, \text{cp}, \text{cs}, \text{fh}, \text{type}) .
\]

A function which returns the Rayleigh-Lamb function values for symmetrical or unsymmetrical plate waves. The input parameter, \( \text{c} \), is the wave speed of the plate wave, while \( \text{(cp, cs)} \) are the compressional and shear wave speeds of the plate, respectively. The wave speed values are arbitrary but must be consistent. The parameter \( \text{fh} \) is the frequency times the half width of the plate. The input parameter, \( \text{type} \), is a string (‘s’ or ‘a’) for symmetrical or anti-symmetrical modes, respectively.

**dispersion\_curves**

\[
\text{>> } \text{dispersion\_curves}.
\]

A script, which like the script dispersion\_plots, generates a 2-D set of non-dimensional frequency and wave speed values at which the plate wave Rayleigh-Lamb dispersion function is evaluated for symmetrical or anti-symmetrical plate waves. The dispersion curves are then plotted with the MATLAB function contour. Individual dispersion curves are extracted and ordered in frequency. The fundamental flexural mode values for small frequencies where the contour values are unreliable are modified with analytical results. The specific dispersion curve specified by the user is then plotted.
G.3 Ultrasonic Beam Models

**bdw_fluid**

\[ p = \text{bdw\_fluid}(x, y, z, a, c, f, N) \]

A function which returns the normalized pressure, \( p \), for a planar circular piston transducer of radius, \( a \) (in mm), radiating into a fluid of wave speed, \( c \), (in m/sec), at a frequency, \( f \), (in MHz). The pressure is calculated at the points \( (x, y, z) \) (in mm), where \( (x, y, z) \) can be scalars, vectors, or matrices. \( N \), is an optional input parameter that specifies the number of line segments used to approximate the edge integral in a boundary diffraction wave model. If \( N \) is not specified, \( N \) is determined automatically so that the line segment length is no larger than one tenth of a wavelength. This function calls the supporting function \( \text{bdw\_model} \) which implements the boundary diffraction wave model of the transducer.

**bdw_model**

\[ p = \text{bdw\_model}(\rho, z, a, c, f, N) \]

A function which returns the normalized pressure, \( p \), for a circular transducer of radius, \( a \) (in mm), radiating into a fluid of wave speed, \( c \), (in m/sec), at a frequency, \( f \), (in MHz). The pressure is calculated at the radial distance \( \rho \) (in mm) and axial distance, \( z \), (in mm) in the fluid. \( N \) is the number of line segments used to approximate the edge integral in a boundary diffraction wave model. All input parameters must be scalars. This is a supporting function which called multiple times by the function \( \text{bdw\_fluid} \).

**onaxis_foc**

\[ p = \text{onaxis\_foc}(z, f, a, R, c) \]

A function which calculates the on-axis normalized pressure, \( p \), at the locations \( z \) (in mm) of a spherically focused transducer of radius \( a \) (in mm) and focal length \( R \) (in mm) radiating into a fluid with wave speed \( c \) (in m/sec), using the O’Neil model. If \( R = \infty \), the function returns the on-axis pressure of a planar transducer of radius \( a \).

**parameters2**

\[ \text{parameters2} \]

A script which is used to define the input parameters needed to model a planar or focused circular piston transducer radiating through a curved interface. These parameters are then placed in a MATLAB structure called setup.

**gauss_c15**

\[ [A, B] = \text{gauss\_c15}() \]

**MG beam2**

\[ v = MG\_beam2(setup) \]

A function which returns the normalized velocity amplitude, \( v \), due to a planar or spherically focused piston transducer radiating obliquely through a curved fluid-solid or a smooth solid-solid interface where the plane of incidence must be aligned with one of the principal axes of the curved surface. The input parameter setup is a MATLAB structure that is generated by the MATLAB script parameters2 and contains all of the input parameters needed to define a given inspection. This is a multi-Gaussian beam model that uses the 15 optimized coefficients of Wen and Breazeale to calculate the wave field. Those coefficients are returned by the MATLAB function gauss_c15.

**onaxis_interface**

\[ [v, vp] = onaxis\_interface(z1, z2, f, a, d1, d2, cp1, cp2, cs2) \]

A function which computes the on-axis velocity of a circular piston transducer of radius \( a \) (in mm) radiating at a frequency \( f \) (in MHz) through a planar fluid-solid interface at normal incidence. The distances \( z1, z2 \) (in mm) are in the fluid and solid, respectively, whose densities are \( d1, d2 \). The compressional wave speed of the fluid is \( cp1 \) and the compressional and shear wave speeds of the solid are \( cp2, cs2 \). All the wave speeds are measured in m/sec. The units of the densities are arbitrary but must be consistent. The function returns both the normalized on-axis velocity, \( v \), and the corresponding paraxial approximation, \( vp \), for the on-axis velocity of the transmitted P-wave. The distance \( z1 \) in the fluid must be a scalar.

**angle_beam2**

\[ \text{angle\_beam2} \]

A script which generates an image of the wave field of a 45 degree angle beam shear wave transducer in the plane of incidence. The script uses the multi-Gaussian beam model MG\_beam2.

**pulse2**

\[ \text{pulse2} \]

A script which generates a wave form at an on-axis point in water produced by a circular planar transducer. It uses the multi-Gaussian beam model MG\_beam2 and the function spectrum to generate the pulse.

### G.4 Flaw Scattering Models

**A crack pe**

\[ A = A\_\text{crack\_pe}(f, b, c, \text{angd}) \]

A function which returns the pulse-echo scattering amplitude, \( A \), (in mm) for a circular crack of radius \( b \) (in mm) at a frequency, \( f \), (in MHz) in a material with a wave speed, \( c \), (in m/sec) and at an angle \( \text{angd} \) (in degrees) using the Kirchhoff approximation.
A \_void\_pe

\[ A = A\_\text{void\_pe}(f, b, c). \] A function which returns the pulse-echo scattering amplitude, $A$, (in mm) of a spherical void of radius $b$ (in mm) at frequency $f$ (in MHz) in a solid or fluid with wave speed $c$ (in m/sec) using the Kirchhoff approximation.

A \_incl\_pe

\[ A = A\_\text{incl\_pe}(f, b, c_1, d_1, c_0, d_0). \] A function which calculates the pulse-echo scattering amplitude, $A$, (in mm) for a spherical inclusion of radius $b$ (in mm) where $(c_1, d_1)$ and $(c_0, d_0)$ are the wave speeds and densities of the flaw and host materials, respectively. Wave speeds are in m/sec and densities are in arbitrary but consistent units. The function uses the modified Born approximation where the wave travels in the flaw at the flaw wave speed and the amplitude of the front and back responses are in terms of the reflection coefficient without making the weak scattering approximation. A phase correction puts the front and back responses at the correct times.

sphere\_rigid\_pe

\[ A = \text{sphere\_rigid\_pe}(f, b, c). \] A function which calculates the pulse-echo scattering amplitude, $A$, (in mm) for a rigid sphere of radius $b$ (in mm) in a fluid with wave speed, $c$, (in m/sec). This function uses the method of separation of variables.

G.5 Effective Parameters and Flaw Sizing

eff\_parameter

\[ [\text{aeff}, \text{Reff}] = \text{eff\_parameters}(c, f, z_{\text{min}}, z_{\text{max}}). \] A function which computes the effective radius ($\text{aeff}$) and effective focal length ($\text{Reff}$) of a spherically focused transducer from measurements of $z_{\text{min}}$ and $z_{\text{max}}$ (both in mm), the on-axis min and max locations in the frequency domain response of the transducer, at a frequency $f$ (in MHz) where the wave speed $c$ is in meters/sec. If $z_{\text{max}}$ is not specified, then $\text{Reff} = \infty$ (planar transducer) is returned and only $\text{aeff}$ is calculated. This function calls a supporting function, transeq.

transeq

\[ y = \text{transeq}(x, z_{\text{max}}, z_{\text{min}}, k). \] A function which returns the values $y$ of the equation $y = f(x, z_{\text{min}}, z_{\text{max}}, k)$, where $(z_{\text{min}}, z_{\text{max}})$ are measured values (in mm) of the on-axis min and max response of a focused transducer and $k$ is the wave number (in rad/mm). The function $f$ is a supporting function needed to determine the effective parameters of a spherically focused transducer.
**ellipse_data**

```
>> [theta, phi, dt] = ellipse_data(). A function which returns experimental sizing data for a 2.5 x 0.6 mm elliptical crack in titanium (c = 6100 m/sec) as measured with a phased array where (theta, phi) are the spherical angles (in degrees) at which dt is measured, where dt is the time between flash points (in microseconds). These measured values are corrected for finite bandwidth errors.
```

**K_sizing**

```
>> [pds, a] = K_sizing(c, theta, phi, dt). A function which returns matrices containing the direction cosines, pds, and lengths of the semi-major axes, a, (in mm) of an equivalent flat elliptical crack in a solid with wave speed, c, (in mm/microsec), based on measured times between flashpoints, dt, (in microsec) at the spherical angles angles (theta, phi), measured in degrees.
```

### G.6 Miscellaneous Functions

**FourierT**

```
>> y = FourierT(x, dt). A function which returns y, the forward FFT of the sampled time domain values, x, where dt is the sampling time interval between points. FourierT approximates the Fourier transform where the integrand of the transform is x*exp(2*pi*i*f*t). This is consistent with the definition of the forward Fourier transform used in this book. For NDE applications the frequency components, f, are normally in MHz, and the sampling time, dt, in microseconds. The values, y, are complex while the values of x are real.
```

**IFourierT**

```
>> y = IFourierT(x, dt). A function which computes the inverse FFT of the sampled frequency domain values, x, for a sampling time interval dt. IFourierT assumes the integrand of the inverse transform is given by x*exp(-2*pi*i*f*t). This is consistent with the definition of the inverse Fourier transform used in this book. The first half of the sampled values of x are the spectral components for positive frequencies ranging from 0 to the Nyquist frequency 1/(2*dt). The second half of the sampled values are the spectral components for the corresponding negative frequencies. The values, y, should be real but may contain small imaginary parts from numerical round off, which can be eliminated by taking the real part, i.e. real(y). If the negative frequency values are set equal to zero then to recover the inverse FFT of x we must replace x (1) by x(1)/2 and then compute 2*real(IFourierT(x,dt)).
```
s_space

\[ y = s\_space(x\_start, x\_end, num) \]

A function which generates num evenly spaced sampled values from \( x\_start \) to \((x\_end - dx)\), where \( dx \) is the sample spacing. This is useful in FFT analysis where we generate sampled periodic functions. Example: generate 1000 sampled frequencies from 0 to 100MHz via \( f = s\_space(0,100,1000) \); in this case the last value of \( f \) will be 99.9 MHz and the sampling interval will be \( 100/1000 = 0.1 \) MHz.

\[ y = c\_shift(x, n) \]

A function which moves the last \( n \) components of the vector \( x \) into the first \( n \) component places and shifts the remaining components of \( x \) to follow those \( n \) components, i.e. this is a circular shift. It can be used to prevent a periodic function, as found in FFT analyses, from being split into the first and last parts of the window in which it is displayed. Note: \( x \) must be a row vector.

t_shift

\[ y = t\_shift(x, n) \]

A function which is used in conjunction with the \( c\_shift \) function to change the time axis values appropriately so that the time axis is shifted along with the function. Example use: plot \((t\_shift(t, 100), c\_shift(fun, 100))\)

spectrum

\[ V = spectrum(f, B, fc, bw) \]

A function which returns a Gaussian-shaped spectrum, \( V \), for a pulse whose time domain waveform is given by \( v = B\cos(2\pi fc t)\exp(-t^2/4a^2) \), where \( fc \) is the center frequency (in MHz when \( t \) is in microseconds), and \( bw \) is the -6dB bandwidth (in MHz). Note that this is a function defined only for positive frequencies so to recover the real time domain waveform, \( v \), we must let \( V(1) = V(1)/2 \), and take twice the real part of the inverse FFT.

lp_filter

\[ y = lp\_filter(f, f\_start, fend) \]

A function which returns low-pass filter, \( y \), at the frequencies contained in the vector \( f \). The filter has values of 1.0 below the frequency value \( f\_start \) and tapers to zero at frequencies above the value \( fend \) with a cosine function. An error is returned if \( fend \) is outside the range of values contained in the vector \( f \) or if \( fend \) is less than \( f\_start \).

display_struct

\[ display\_struct(setup) \]

A function which returns all the parameters contained in the structure, \( setup \), where \( setup \) is generated by the script \( parameters2 \).

sphH

\[ y = sphH(n, x) \]

A function which returns the spherical Hankel function of order \( n \) at \( x \), where \( x \) is a non-dimensional argument. It is used in separation of variables solutions.
sphJ

>> y = sphJ(n, x). A function which returns the spherical Bessel function of order \( n \) at \( x \), where \( x \) is a non-dimensional argument. It is used in separation of variables solutions.
Index

A
Absorption losses, 386
Acoustic/elastic transfer function, 23–25, 33, 392–394, 397, 413, 559, 567, 574, 616
Acoustic impedance, specific, 46, 115, 194
Acoustic intensity, 116–119
Acoustic lens, 251, 270–272
Acoustic radiation impedance, 391, 393, 397, 558
ALOK, 670
Alternating tensor, 40, 717
Angle beam transducer, 155, 156, 199, 352–358
Angular spectrum, 67, 68, 247–251, 272, 273, 278, 284, 291, 344, 398, 528, 532, 534, 602
Anti-plane strain, 50, 110, 163
Area function. See Flaw area function
A-scan, 10, 12, 24, 580, 634, 643, 651, 659, 669
Associated Legendre functions, 495
Auld measurement model. See Reciprocity-based models

B
Blocked force, 393, 394, 397, 558
Born approximation
fluid, 448–457, 477, 483, 520
solid, 477–482
Boundary diffraction waves, 219, 239, 247–251, 282, 325, 326, 332, 367, 369, 375, 601, 604, 719, 746
Boundary elements, 97, 104, 108, 515
B-scan, 10–12, 643
Bulk modulus, 35, 47, 500
Bulk waves, 12, 45, 47, 55–87, 201, 204, 208, 219, 344, 345, 350, 532

C
Caustics, 186, 309, 316–317, 322
Christoffel symbol, 303, 735
Confocal parameters, 81
 Constitutive equations
fluid, 34, 35
solid, 40, 41
Contact transducer, 7
Convolution, 16, 19–21, 27, 29, 30
698, 710
Crack width function
fluid, 444, 445, 517
solid, 471–475
C-scan, 11, 12, 643
Cylindrical scatterer, 519

D
Damage accumulation function, 691, 692
Decibel (dB), 378, 379, 387
Deconvolution, 21, 22, 24, 407–413, 544, 610, 616, 637–639, 657, 666, 672
Diffraction correction integral, 398–406, 411, 413, 415, 416
Diffraction corrections
  on reception, 525, 527, 530, 535, 536, 542, 543, 545–551, 603, 604
Digitizer, 3, 8–9, 12
Dilatation, 35, 46–47
Dirac delta function, 63, 699, 704, 711–714
Direct waves, 223, 234, 248, 249, 281, 282, 285, 292, 293, 324, 326, 588, 603, 623–625
Discrete delta function, 704, 709
Dispersion
  geometrical, 389
  material, 389
Divergence theorem, 34, 90, 430, 478, 719

E
  Edge elements, 309, 325–338, 341, 357–360, 446, 604
  Effective focal length, 614, 618, 748
  Effective radius
    flaw, 653, 659, 663, 666, 739
    transducer, 611–612, 615, 622, 623, 748
  Eigenvalue sizing. See Linear least squares/eigenvalue sizing
  Eikonal equation, 170
  Elastic constant tensor, 99, 319, 321, 424, 463, 477
  Electromechanical reciprocity, 89, 104–107, 539, 570–573
  Ellipsoidal flaw, 452, 666, 668, 677
  Elliptical crack, 439–445, 468–471, 653, 671–673, 737, 749
  Energy density function, 40, 41
  Energy flux
    fluid, 118, 126
    solid, 148
  Equations of motion
    fluid, 33
    solid, 38–50
  Equivalent flaw sizing, 475, 651–652, 654, 656, 666–671, 737, 739
  Equivalent radius. See Effective radius
  Equivalent time sampling, 9
  Extensional waves, 209–211
  Far field scattering amplitude integrals
    cracks, 421
    volumetric flaws, 419–421
  Fast Fourier transform (FFT), 19, 26, 205, 612, 616, 619, 634, 704–708
  Figure of merit. See Noise figure of merit
  Flash points, 231, 442, 445, 448, 475, 652–654, 656, 657, 659, 667, 669, 673, 674, 676, 678, 749
  Flat-bottom hole models, 623–636, 644, 646
  Flaw area function
    fluid, 435, 456
    solid, 482
  Flexural waves. See Plate waves
  Focused transducer models. See Transducer radiation models (focused)
  Fraunhofer approximation, 335
  Free surface velocity, 397, 567
  Frequency, 2, 16, 57, 113, 199, 223, 386, 421, 534, 542, 583, 610, 657
  Fundamental solution
    far field, 68–69, 73–75, 318
    fluid, 63, 86
    integral forms, 66–68
    solid, 69–73, 89
  Gauss-Hermite beam model, 640
  Gaussian curvature, 739, 740
  Gauss’ theorem, 39, 65, 98, 106, 450, 718–719
  Geometrical theory of diffraction (GTD), 644
  Governing equations, elastic solid
displacement formulation, 47–48
potential formulation, 47–49
Grain scattering, 57, 386, 387, 390, 574, 640, 642, 685, 688
Group velocity, 207, 216
Hankel function, 86, 87, 493, 494, 497, 519
Heaviside step function, 699
Helmholtz decomposition, 44, 48, 49, 101
Helmholtz equation, 66, 89, 90, 92, 101, 487, 488, 492, 502
Hilbert transform, 129, 131, 473, 727–728
Hypersingular integrals, 104, 720

I
Immersion testing, 6, 10, 22, 113, 122, 132, 133, 146, 219, 272, 279, 375, 398, 539, 540, 545, 552–563, 577–579, 611, 624, 634, 635, 667
Impedance. See Acoustic impedance
Indicial notation, 715–718
Inhomogeneous waves, 113, 122–125, 127–132, 145, 155, 187, 197, 200, 201, 250
Integral equations
fluid, 89, 96–98, 108
solid, 89, 103–104, 108
Integral representation theorem
fluid, 91, 92, 111, 220
solid, 99, 100
Interface/boundary conditions
fluid, 36–38
solid, 42–44

J
Jacobian, 723, 732, 734

K
Kirchhoff approximation
    curved interface, 167, 168, 301, 339
    fluid (cracks), 438–448
    fluid (volumetric flaws), 427–438
    solid (cracks), 466–477
    solid (volumetric flaws), 457–466
Kirchhoff flaw sizing, 169, 449, 475, 652, 666, 670, 737
Kronecker delta function, 716, 717

L
Lamb waves. See Plate waves
Leading edge response
    fluid, 436–438
    solid, 463–466, 605
Legendre polynomials, 590
Lens. See Acoustic lens
Linear least squares/eigenvalue sizing, 654–657, 663, 672, 675
Love waves, 215
Low frequency scattering, 448, 499, 500, 515, 521
LTI model
    fluid medium, 540–544
    general, 390–398
    immersion testing, 545–552, 579
Mean curvature, 740
Measurement models
    angle beam shear wave (contact), 563–570
    direct derivation, 539–552
    near field (see Near field models)
    reciprocity-based derivation, 552–570
Mode conversion, 8, 141, 146, 194, 352, 606
Model-assisted POD (MAPOD), 689, 694
Modified Born approximation (MBA), 482–486, 510, 514, 748

N
Near field distances, 225, 226, 372, 374, 381, 600, 601
Near field models
    pressure-pressure, 597–598
    pressure-velocity, 598
    velocity-pressure, 583–596
    velocity-velocity, 596
Noise figure of merit (FOM), 688, 689
Nonlinear least squares sizing, 654
Numerical beam modeling
    boundary diffraction wave model, 238–243
    focused transducer (on-axis), 258
    multi-Gaussian beam model, 361–374
    planar transducer (on-axis) radiating through a plane interface, 367–368
Numerical flaw modeling
    crack, 510, 513
    inclusion, 510
    rigid sphere, 510, 514
    void, 510–512
Nyquist criterion, 703
**O**

O’Neil model, 251–254, 260, 270, 272, 329, 364, 746

**P**


Paraxial wave equation, 77–80, 84, 87, 183

Paris-Erdogan flaw growth law, 691, 693

Parseval’s theorem, 27, 699

Phased array transducers, 7, 8, 11

Physical optics far field inverse scattering (POFFIS), 475, 670

Piezoelectric crystal, 1, 4, 194

Piezoelectric medium reciprocal theorem.

See Electromechanical reciprocity

Pitch-catch, 10, 22, 411, 438, 457, 501–502, 519, 520, 525, 526, 532, 540, 541, 543, 545, 552, 570, 583, 604, 638, 639, 652

Planar aperture model-focused transducer, 260

Planar piston transducer in a fluid boundary diffraction waves, 247–251
diffraction correction, 227, 228, 231–235
far field response, 226, 227
impulse response, 244, 245, 247, 248
off-axis response, 227–231
on-axis response, 222–226

Planar transducer models. See Transducer radiation models (planar)


Plane strain, 49–50, 110, 197, 208

Plane waves

fluid (1-D), 55–59
fluid (3-D), 58–59
solid (1-D), 59–60
solid (3-D), 60–63

Plate waves

extensional, 208–211
flexural, 211–214
$SH$, 206–208


Potentials, elastic, 44, 45, 47, 75, 78, 141, 197, 502

Potentials, plane wave amplitudes, 169

Principal curvatures, 176, 305, 315, 361, 366, 735, 740

Principal value integral, 96, 103, 240, 263, 279

Probability of detection (POD), 544, 643, 644, 685–694

Pulse distortion, 113, 127–132, 146, 471

Pulse-echo, definition, 10

Pulser-receiver, 3–4

**Q**

Quarter wave plate, 191, 192

**R**

Radiation impedance, 567

Ray theory, 170, 306, 308, 309, 614

Rayleigh distance, 270, 362

Rayleigh-Sommerfeld equation, 249

Rayleigh wave, 8, 141, 197–201, 204, 215, 216, 645, 745

Reception process model fluid, 525–527, 532, 534, 536
fluid-fluid, 527–531, 536, 537
fluid-solid, 531–536

Reciprocal theorem

electromechanical (see Electromechanical reciprocity)
fluid, 89–90
solid, 98–101

Reciprocity-based models
angle beam (contact) model, 563–570
Auld measurement model, 539, 552–559, 562, 563, 568, 570, 573–576, 639–641

Reciprocity, scattering amplitude, 486–491

Reflection coefficients

normal incidence, 113–120, 166
oblique incidence, 120–153

Reliability index. See Safety index
Reliability modeling, 689–693
Rodrigues’ formula, 495
Rotation, local, 47

S
Safety index, 690
Satellite pulse method, 669, 670
Scattering amplitude expressions for
crack (fluid), 421
crack (solid), 425–426
ellipsoid (fluid), 419–421
ellipsoid (solid), 422–426
Scattering amplitude integrals. See Far field
scattering amplitude integrals
Separation of variables
spherical fluid inclusion in a solid, 507
spherical fluid inclusion in a solid, 502–509
spherical scatterer in a fluid, 492–502
spherical void, 507
SH-waves, 86, 158, 163–164, 185, 354, 459,
460, 464–466
Simulators, ultrasonic
CIVA, 643–644
EFIT, 643
GB and GPSS, 641
UTDefect and simSUNDT, 642–643
UTSIM, 640
Sizing. See Flaw sizing
Slowness vector, 168–170
Snell’s law, 113, 121, 124, 136–142, 145, 154,
159, 169, 177, 180, 181, 186–188, 195,
200, 275, 279, 280, 284, 286, 289, 300,
303, 304, 310, 318, 321, 336, 340, 342,
357, 368, 370, 399, 461, 464, 465, 561,
667, 744
Snell’s law and stationary phase, 138–141, 188
Sommerfeld radiation conditions
fluid, 109, 317, 419, 488, 494
solid, 317
Spherical Bessel functions, 484, 493, 494,
590, 751
Spherical Hankel functions, 493, 494, 497,
590, 750
Spherical waves
fluid, 63–69
solid, 69–75
Spherically focused transducer in a fluid
diffraction correction, 259, 260, 263–264
off-axis response, 261–269
on-axis response, 254, 257–258
Stationary phase, method of
double integral, 732–733
single integral, 729–732
surface integral, 734–736
Stokes’ relations, 132–134, 151–153, 188,
531, 549
Stokes’ theorem, 250, 328, 330, 446, 602,
719–720
Strain tensor, 40, 41, 721, 726
Surface wave. See Rayleigh wave
SV-waves, 61, 144, 147, 151, 157–162,
164–166, 184, 185, 208, 355, 365, 370,
459, 464, 470, 471, 521, 532, 563, 626,
652, 744
Synthetic aperture focusing technique (SAFT),
8, 475, 670
System efficiency factor. See Efficiency factor
System function, 24, 33, 385–409, 411–416,
543, 546, 548, 559, 567, 573, 574, 576,
588, 610, 616–617, 644
T
Thompson-Gray measurement model.
See Reciprocity based models
Through-transmission, 10
Time of flight diffraction sizing (TOFD), 475,
669, 670
Time of flight equivalent sizing (TOFE),
666–670, 674, 678–681
Total focusing method (TFM), 475, 670
Traction vector, 39, 43, 44, 148, 553, 555,
723–725
Transducer characterization, 610–622
Transducer radiation models (focused), 291
curved interface, 298–317, 299
plane fluid-fluid interface, 291–294
plane fluid-solid interface, 295–298
single fluid (see Spherically focused
transducer in a fluid)
Transducer radiation models (planar)
curved interface, 298–318, 320–323
plane fluid-fluid interface, 272–279,
284–290
plane fluid-solid interface, 279–284,
290–291
single fluid (see Planar piston transducer
in a fluid)
Transducers
angle beam, 7, 8, 153, 155, 200,
352, 359
contact, 5–7, 344–352, 356, 359, 396,
406, 409, 413, 578
immersion, 6, 194, 344, 392, 405, 406,
601, 605, 622, 623, 643, 646
Transmission coefficients
normal incidence, 114–116
oblique incidence, 120–122, 141–147
Transport equation, 170
U
Ultrasonic simulators. See Simulators, ultrasonic

V
Velocity. See Wave speed
Velocity, group. See Group velocity

W
Wave equations
  fluid, 36, 44, 47, 55, 78, 79, 553
  solid, 45, 75, 78, 84, 115, 183
Wave speed
  in a fluid, 33, 55, 86, 141, 150, 511, 517
  of P-waves, 45, 57, 157, 162, 166, 280, 318, 366, 483, 600, 744, 747
  of Rayleigh wave, 201, 216
  of SH plate waves, 203
  of S-waves, 45, 47, 157, 280, 366, 483, 743, 744
  table, 46
Wen and Breazeale coefficients, 361, 365
Width function. See Crack width function
Wiener filter, 22, 24, 407, 409, 413, 414, 616, 617, 637

Z
Zero-of-time problem, 665, 666, 670