BLTITLE

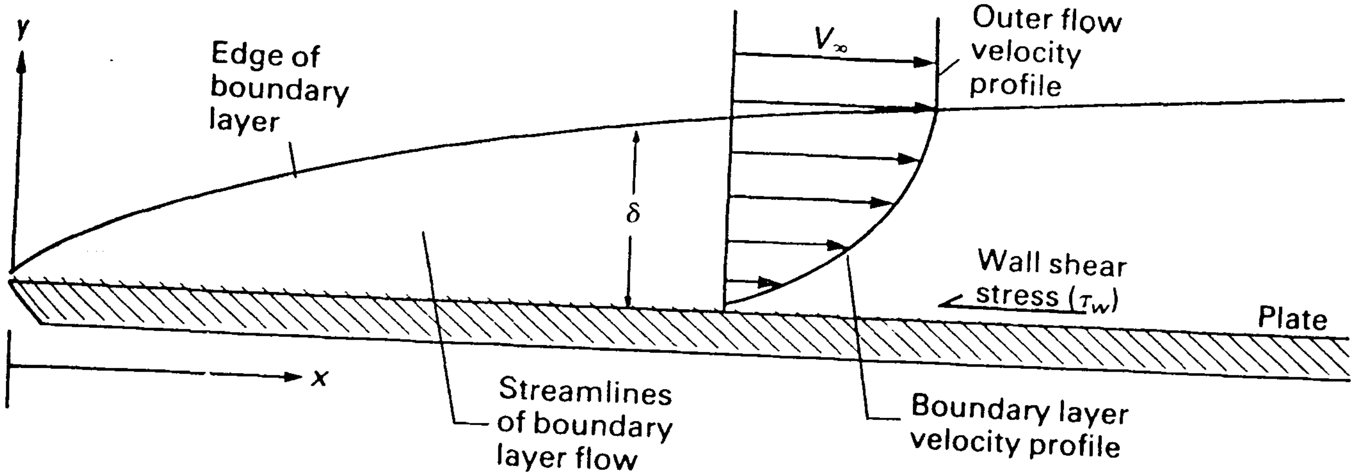
Boundary Layers

Two-Dimensional Steady Boundary Layer Equations

x is horizontal direction along direction of main flow velocity u. Velocity at outer edge of boundary layer is called U_∞ or V_∞ or U_e or V_e.

y is perpendicular to wall and velocity in this direction is v.

The boundary layer begins, say, at $x = 0$ and the boundary layer thickness is δ. $\delta \ll x$. Because the boundary layer is thin, to leading order the pressure is constant through the thickness of the boundary layer, $\frac{\partial P}{\partial y} = 0$. Also, $v \ll u$, and $\frac{\partial u}{\partial x} \ll \frac{\partial u}{\partial y}$.



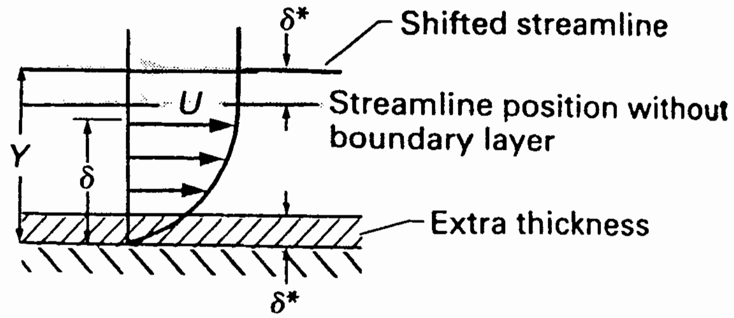
$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0$$

$$u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = -\frac{1}{\rho}\frac{\partial P}{\partial x} + \nu\frac{\partial^2 u}{\partial y^2}$$

$$u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = -\frac{1}{\rho}\frac{\partial P}{\partial x} + \frac{1}{\rho}\frac{\partial \tau}{\partial y}$$

fmbbL2

Boundary Layer Parameters



Thickness of Boundary Layer defined as location where u is 99% of U_e.

$$\delta = y]_{u/U_e=0.99}$$

The wall shear stress τ_w is given by:

$$\tau_w = \mu \left(\frac{\partial u}{\partial y}\right)_{\text{wall}} = \mu \left(\frac{\partial u}{\partial y}\right)_{y=0}$$

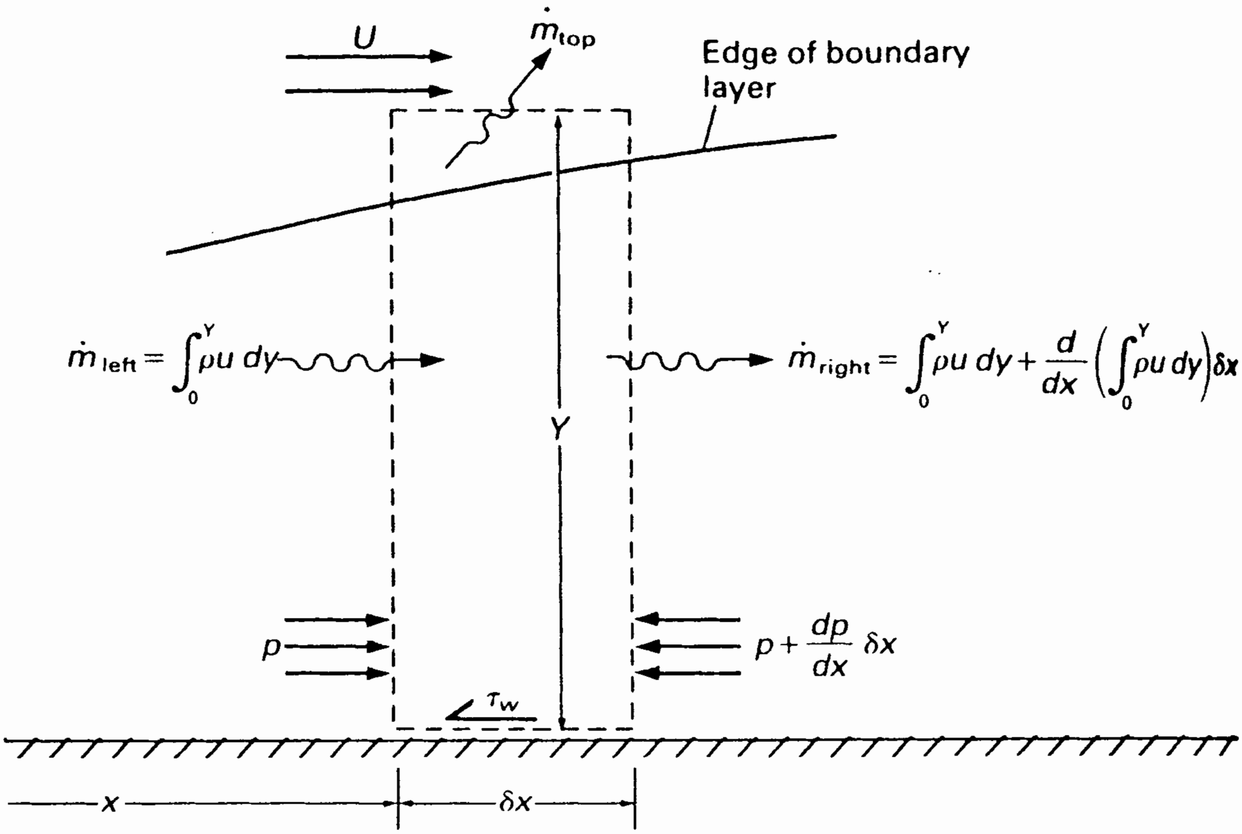
The skin friction coefficient, C_f, is:

$$C_f = \tau_w / \left(\frac{1}{2}\rho U_e^2\right) = \frac{2\tau_w}{\rho U_e^2} = \frac{2\nu}{U_e^2}\left(\frac{\partial u}{\partial y}\right)_{y=0}$$

The displacement thickness, δ^* is the thickness of a flow of speed U_e that carries a flow rate equal to the deficit in the boundary layer because its speed is less than U_e.

$$U_e\,\delta^* = \int_0^\delta (U_e - u)dy \qquad\qquad \delta^* = \int_0^\delta \left(1 - \frac{u}{U_e}\right) dy$$

Mass Fluxes



$$\dot{m}_{\text{left}} = \dot{m}_{\text{right}} + \dot{m}_{\text{top}}$$

$$\dot{m}_{\text{top}} = -\frac{d}{dx}\left(\int_0^Y \rho u dy\right)\delta x$$

Momentum Equation in x direction

$$\dot{M}_{\text{right}} + \dot{M}_{\text{top}} + \dot{M}_{\text{left}} = F_{\text{pressure}} + F_{\text{stress}}$$

$$\dot{M}_{\text{left}} = -\int_0^Y \rho u^2 dy$$

$$\dot{M}_{\text{right}} = \int_0^Y \rho u^2 dy + \frac{d}{dx}\left(\int_0^Y \rho u^2 dy\right)\delta x$$

$$\dot{M}_{\text{top}} = \dot{m}_{\text{top}} U_e = -U_e \frac{d}{dx}\left(\int_0^Y \rho u dy\right)\delta x$$

$$F_{\text{pressure}} = -\frac{dp}{dx} Y\,\delta x = \rho U_e \frac{dU_e}{dx} Y\,\delta x \qquad\qquad F s = -\tau_w \delta x$$

One additional needed equation is:

$$Y = \int_0^Y dy$$

Then all the equations on the last two pages can be combined into:

$$\frac{d}{dx}\int_0^Y u(U_e - u)dy + \frac{dU_e}{dx}\int_0^Y (U_e - u)dy = \frac{\tau_w}{\rho}$$

For $y > \delta$ the integrands are zero so the upper limits can be changed to δ.

$$\frac{d}{dx}\int_0^\delta u(U_e - u)dy + \frac{dU_e}{dx}\int_0^\delta (U_e - u)dy = \frac{\tau_w}{\rho}$$

This is Von Karman's Momentum Integral Equation. It relates the integrals of the velocity profile in the boundary layer to the shear stress and U_e and U_e^2 whose x-derivative is proportional to the pressure gradient.

The momentum thickness Θ is defined as:

$$\Theta = \int_0^\delta \frac{u}{U_e}\left(1 - \frac{u}{U_e}\right) dy$$

With this definition, the momentum integral equation can be written in the following two forms:

$$\frac{d}{dx}\left[U_e^2 \Theta\right] + \delta^* U_e \frac{dU_e}{dx} = \tau_w / \rho$$

$$\frac{d\Theta}{dx} + (2 + H)\frac{\Theta}{U_e}\frac{dU_e}{dx} = \frac{C_f}{2} \qquad\qquad \text{where: } H \equiv \frac{\delta^*}{\Theta}$$

A second boundary layer equation comes from equating the kinetic energy change along x in the boundary layer to the energy input or output from the pressure distribution and the energy dissipation due to shear stresses in the boundary layer.

The kinetic energy thickness, θ^* is defined as:

$$\theta^* = \int_0^\delta \frac{u}{U_e}\left(1 - \frac{u^2}{U_e^*}\right) dy$$

The kinetic energy dissipation coefficient, C_D, is defines ad:

$$C_D = \frac{D}{\rho u_e^3}$$

where D is the dissipation per unit area (along and perpendicular to the surface).

Using these definitions, the kinetic energy equation is:

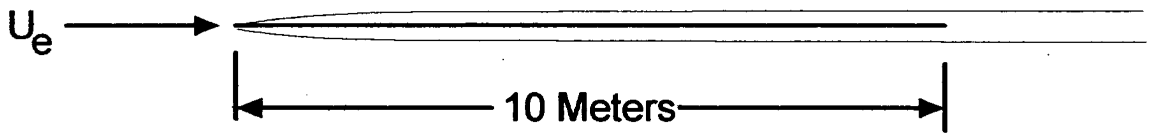
$$\frac{d\theta^*}{dx} + 3\frac{\theta^*}{u_e}\frac{du_e}{dx} = 2C_D$$

The energy thickness ratio, H^* is defined as: $H^* = \dfrac{\theta^*}{\theta}$

It is common to combine the kinetic energy equation and Von Karman's momentum equation to obtain:

$$\frac{\theta}{H^*}\frac{dH^*}{dx} = \frac{2C_D}{H^*} - \frac{C_f}{2} + (H-1)\frac{\theta}{u_e}\frac{du_e}{dx}$$

Example of Solution of Momentum Integral BL Equation



$U_e = 2m/s \qquad \delta(x) = 0.01*(1-e^{-0.1x}) \qquad \frac{u(y)}{U_e} = \left(1 - e^{-k(x)y}\right)^2 \qquad \rho = 1000\text{kg/m}^3$

Problem: Determine the shear stress, τ, at x = 5 meters.

Determination of $k\delta$ from BL thickness:

$$0.99 = \left(1 - e^{-k(x)\delta(x)}\right)^2 \longrightarrow k(x)\delta(x) = 5.3 \qquad k(x) = \frac{5.3}{\delta(x)}$$

At x = 5m, $k = 1347 \ m^{-1}$.

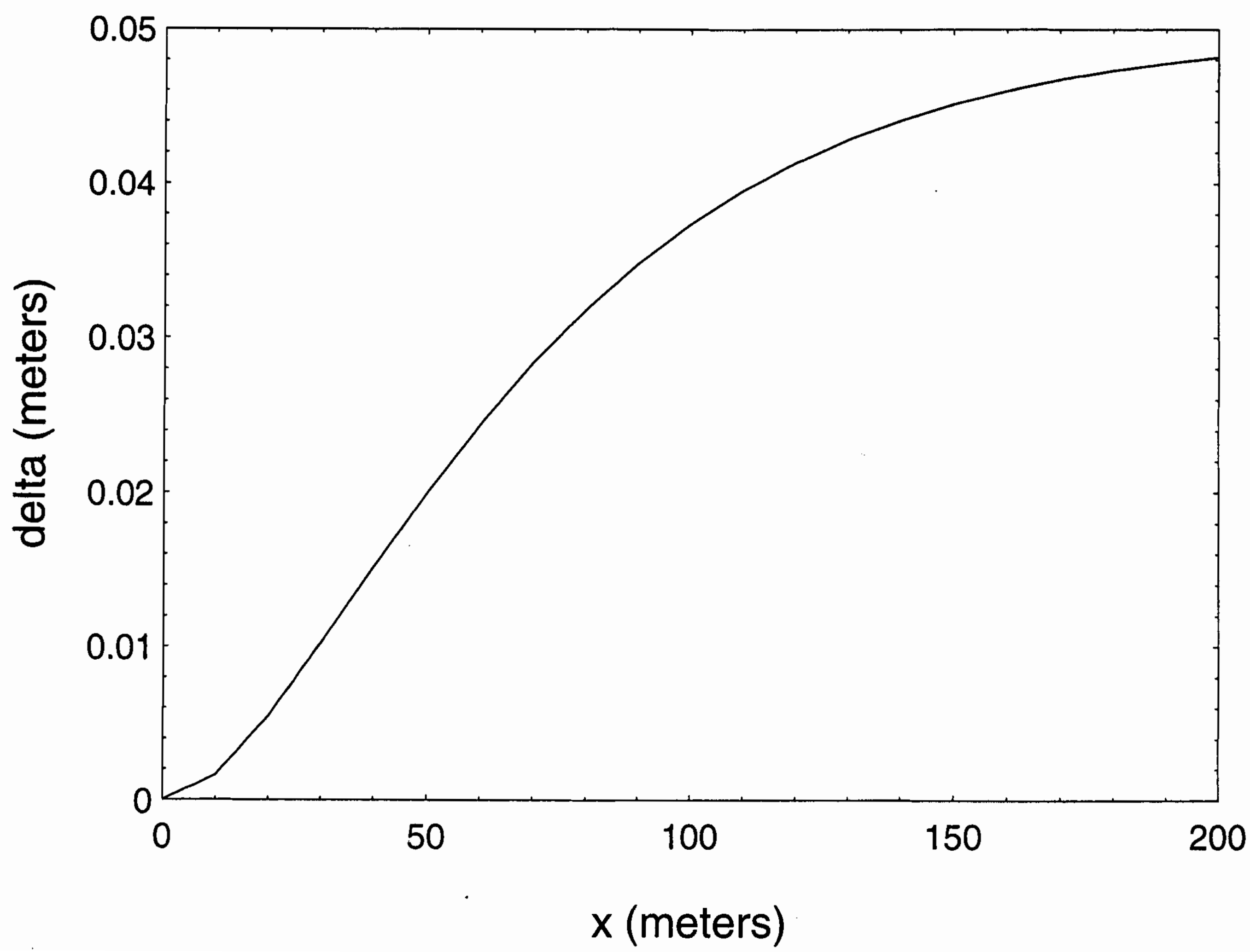
$$\frac{d}{dx}\int_0^{0.01[1-\exp(-0.1x)]} U_e(1-e^{k(x)y})^2\left[U_e - U_e(1-e^{-ky})^2\right]dy + 0 = \frac{\tau}{\rho}$$

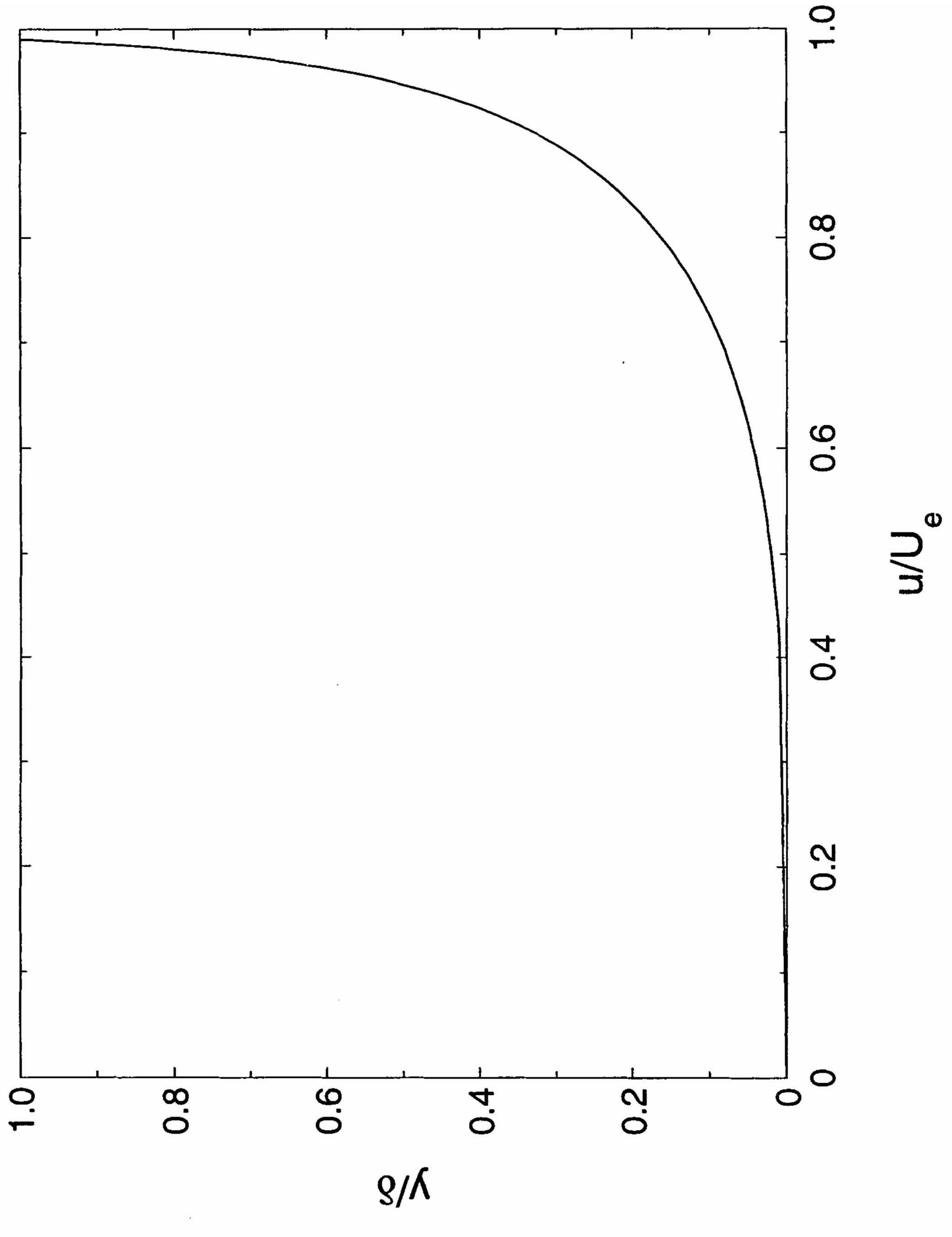
$$0.01(1-e^{-0.1\times 5}) = 0.01(1-e^{-0.5}) = 0.00393$$

$$\begin{aligned}\frac{\tau}{\rho} &= U_e^2(1-e^{-5.3})^2\left[1-(1-e^{-5.3})^2\right]\frac{d}{dx}\left[0.01(1-e^{-0.1x})\right] \\ &\quad + \int_0^{0.00393}\frac{d}{dx}\left\{U_e(1-e^{-k(x)y})^2\left[U_e - U_e(1-e^{-k(x)y})^2\right]\right\}dy \\ &= U_e^2(0.000060 + 0.000100) = 0.00016\,U_e^2\end{aligned}$$

$$\tau = 1000 \times 4 \times 0.00016 = 0.64N/m^2$$

$$c_f = \frac{\tau}{\frac{1}{2}\rho U_e^2} = 0.00032$$


delta (meters)
0.05
0.04
0.03
0.02
0.01
0
0
50
100
150
200
x (meters)



y/δ
u/U_e
0
0.2
0.4
0.6
0.8
1.0

x	delta
0.000000	0.000000
10.000000	0.001643
20.000000	0.005434
30.000000	0.010179
40.000000	0.015162
50.000000	0.019979
60.000000	0.024416
70.000000	0.028381
80.000000	0.031848
90.000000	0.034836
100.000000	0.037382
110.000000	0.039534
120.000000	0.041340
130.000000	0.042848
140.000000	0.044104
150.000000	0.045145
160.000000	0.046007
170.000000	0.046718
180.000000	0.047305
190.000000	0.047788
200.000000	0.048185

udata

y/delta	u/Ue
0.000000	0.000000
0.010000	0.422358
0.020000	0.500261
0.030000	0.551427
0.040000	0.590201
0.050000	0.621605
0.060000	0.648047
0.070000	0.670886
0.080000	0.690975
0.090000	0.708884
0.100000	0.725018
0.110000	0.739674
0.120000	0.753079
0.130000	0.765409
0.140000	0.776804
0.150000	0.787378
0.160000	0.797225
0.170000	0.806423
0.180000	0.815038
0.190000	0.823127
0.200000	0.830737
0.210000	0.837911
0.220000	0.844685
0.230000	0.851092
0.240000	0.857160
0.250000	0.862915
0.260000	0.868379
0.270000	0.873573
0.280000	0.878515
0.290000	0.883221
0.300000	0.887707
0.310000	0.891986
0.320000	0.896072
0.330000	0.899974
0.340000	0.903705
0.350000	0.907274
0.360000	0.910689
0.370000	0.913960
0.380000	0.917093
0.390000	0.920096
0.400000	0.922976
0.410000	0.925739
0.420000	0.928390
0.430000	0.930936
0.440000	0.933381
0.450000	0.935730
0.460000	0.937988
0.470000	0.940158
0.480000	0.942245
0.490000	0.944252
0.500000	0.946184
0.510000	0.948042
0.520000	0.949832
0.530000	0.951554

y/delta	u/Ue
0.540000	0.953213
0.550000	0.954812
0.560000	0.956351
0.570000	0.957835
0.580000	0.959264
0.590000	0.960643
0.600000	0.961971
0.610000	0.963253
0.620000	0.964488
0.630000	0.965680
0.640000	0.966830
0.650000	0.967939
0.660000	0.969009
0.670000	0.970042
0.680000	0.971039
0.690000	0.972001
0.700000	0.972930
0.710000	0.973827
0.720000	0.974693
0.730000	0.975529
0.740000	0.976336
0.750000	0.977116
0.760000	0.977870
0.770000	0.978597
0.780000	0.979300
0.790000	0.979979
0.800000	0.980636
0.810000	0.981270
0.820000	0.981882
0.830000	0.982475
0.840000	0.983047
0.850000	0.983600
0.860000	0.984135
0.870000	0.984651
0.880000	0.985151
0.890000	0.985634
0.900000	0.986101
0.910000	0.986552
0.920000	0.986989
0.930000	0.987411
0.940000	0.987819
0.950000	0.988214
0.960000	0.988595
0.970000	0.988964
0.980000	0.989321
0.990000	0.989666
1.000000	0.990000

FMBBL6A(1)

Calculation of Turbulent Boundary Layer when Pressure Distribution is Known

This result is approximate since the boundary layer thickness will alter the pressure distribution.

The principal unknowns (quantities to be determined) are: $\theta(x)$ and $\delta^*(x)$. An equivalent set of unknowns is $\theta(x)$ and $H(x)$.

There are two fundamental equations:

$$\frac{d\theta}{dx} = -(H+2)\frac{\theta}{U_e}\frac{dU_e}{dx} + \frac{C_f}{2} \tag{1}$$

$$\frac{\theta}{H^*}\frac{dH^*}{dx} = \frac{2C_D}{H^*} - \frac{C_f}{2} + (H-1)\frac{\theta}{u_e}\frac{du_e}{dx} \tag{2}$$

To be able to integrate the unknowns along the boundary layer, the derivatives of each of them are required: $d\theta/dx$ and dH/dx. Equation 1 is in the desired form. To put equation 2 in the desired form, use the chain rule:

$$\frac{dH^*}{dx} = \frac{dH}{dx}\frac{dH^*}{dH} \tag{3}$$

Empirical "closure relations" for $H^*(H)$ and dH^*/dH exist. Therefore we write the energy equation in the desired form as:

$$\frac{dH}{dx} = \frac{H^*}{\theta}\frac{1}{dH^*/dH}\left[\frac{2C_D}{H^*} - \frac{C_f}{2} + (H-1)\frac{\theta}{u_e}\frac{du_e}{dx}\right] \tag{4}$$

To do the integrals numerically, we need a means of determining C_f, C_D, H^* and dH^*/dH in terms of the principal quantities H and R_θ, where $R_\theta = U_e\theta/\nu$. These empirical "closure relations" have been determined by assembling a large amount of experimental data.

Laminar Closure Relations

$$H^* = \begin{cases} 0.76(H-4)^2/H + 1.515, & H < 4.0 \\ 0.015(H-4)^2/H + 1.515, & H \geq 4.0 \end{cases}$$

$$C_f = \begin{cases} \left[0.03954[(7.4-H)^2/(H-1.0)] - 0.134\right]/R_\theta, & H < 7.4 \\ \left[0.044[1.0 - 1.4/(H-6)]^2 - 0.134\right]/R_\theta, & H \geq 7.4 \end{cases}$$

$$\frac{2C_D}{H^*} = \begin{cases} \left[0.00205(4-H)^{5.5} + 0.207\right]/R_\theta, & H < 4.0 \\ \left[-0.003(H-4.0)^2/(1+0.02(H-4)^2) + 0.207\right]/R_\theta, & H \geq 4.0 \end{cases}$$

Turbulent Closure Relations

$$H_o = \begin{cases} 3 + 400/R_\theta, & R_\theta > 400 \\ 4, & R_\theta \leq 400 \end{cases}$$

$$R_{\theta z} = \begin{cases} R_\theta, & R_\theta > 200 \\ 200, & R_\theta \leq 200 \end{cases}$$

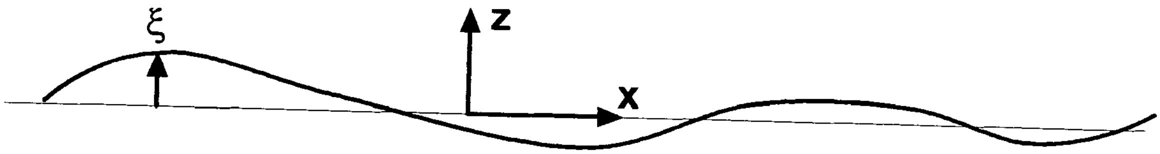
$$H^* = \begin{cases} 1.505 + 4/R_\theta + (0.165 - 1.6/\sqrt{R_\theta})\frac{(H_o-H)^{1.6}}{H}, & H < H_o \\ (H-H_o)^2[0.007\frac{\ln(R_{\theta z})}{[H-H_o+4/\ln(R_{\theta z})]^2} + 0.015/H] + 1.505 + 4.0/R_\theta, & H \geq H_o \end{cases}$$

$$C_f = 0.3e^{-1.33H}\left[\frac{\ln(R_\theta)}{2.3026}\right]^{-(1.74+0.31H)}$$

$$\frac{2C_D}{H^*} = 0.5C_f\frac{4.0/H - 1}{3} + 0.03\left(1 - \frac{1}{H}\right)^3$$

FM813

Sea Waves



Dominated by inviscid irrotational solution ($\nabla^2\phi = 0$)

Boundary Conditions

$$\frac{\partial\phi}{\partial z} = \frac{\partial\zeta}{\partial t} + \left[\frac{\partial\phi}{\partial x}\frac{\partial\zeta}{\partial x} + \frac{\partial\phi}{\partial y}\frac{\partial\zeta}{\partial y}\right]_{z=\zeta} \qquad \text{(kinematic)}$$

$$\left\{\frac{\partial\phi}{\partial t} + \frac{1}{2}\left[\left(\frac{\partial\phi}{\partial x}\right)^2 + \left(\frac{\partial\phi}{\partial x}\right)^2 + \left(\frac{\partial\phi}{\partial y}\right)^2\right]\right\}_{z=\zeta} + g\zeta = \text{constant (0)} \qquad \text{(dynamic)}$$

Linearized Boundary Conditions

Case of onset flow velocity of $-\hat{\imath}U$.
Now ϕ is the perturbation potential and the total potential is $-Ux + \phi$.

$$\left[\frac{\partial\phi}{\partial z}\right]_{z=0} = \frac{\partial\zeta}{\partial t} - U\frac{\partial\zeta}{\partial x} \qquad \left[\frac{\partial\phi}{\partial t} - U\frac{\partial\phi}{\partial x}\right]_{z=0} + g\zeta = 0$$

For steady flow with onset flow:

$$\frac{\partial\phi}{\partial z} = -U\frac{\partial\zeta}{\partial x} \qquad U\frac{\partial\phi}{\partial x} = g\zeta \qquad \frac{\partial\phi}{\partial z} = -\frac{U^2}{g}\frac{\partial^2\phi}{\partial x^2}$$

Case of 2D waves and zero onset flow so ϕ is the total potential.

$$\left[\frac{\partial\phi}{\partial z}\right]_{z=0} = \frac{\partial\zeta}{\partial t} \qquad \left[\frac{\partial\phi}{\partial t}\right]_{z=0} + g\zeta = 0$$

Dispersion Relations for waves of ciruclar frequency $\omega = 2\pi f$ and wavenumber $= k = 2\pi/\lambda$ and zero onset flow.

$$\omega^2 = gk \qquad \text{deep water}$$

$$\omega^2 = gk\tanh kh \qquad \text{water of depth } h$$

FMB13(B)

$$\left.\frac{\partial^2\phi}{\partial t^2}\right]_{z=0} = -g\frac{\partial\zeta}{\partial t} \qquad \left.\frac{\partial\phi}{\partial z}\right]_{z=0} = -\frac{1}{g}\left.\frac{\partial^2\phi}{\partial t^2}\right]_{z=0}$$

$$\zeta = Ae^{i(kx-\omega t)} \quad \text{and} \quad \zeta = -\frac{1}{g}\frac{\partial\phi}{\partial t}$$

Deep Water

$\phi = Be^{kz}e^{i(kx-\omega t)}$ Traveling wave that satisfies Laplace's Equation

$$Bke^{kz}e^{i(kx-\omega t)} = \frac{1}{g}\omega^2 Be^{kz}e^{i(kx-\omega t)} \qquad k = \frac{\omega^2}{g} \qquad \omega^2 = kg$$

$$\zeta = -\frac{1}{g}(-i\omega)B\,e^{i(kx-\omega t)}$$

$$A = \frac{i\omega}{g}B \qquad B = -\frac{ig}{\omega}A = -i\frac{\omega}{k}A$$

Finite Depth

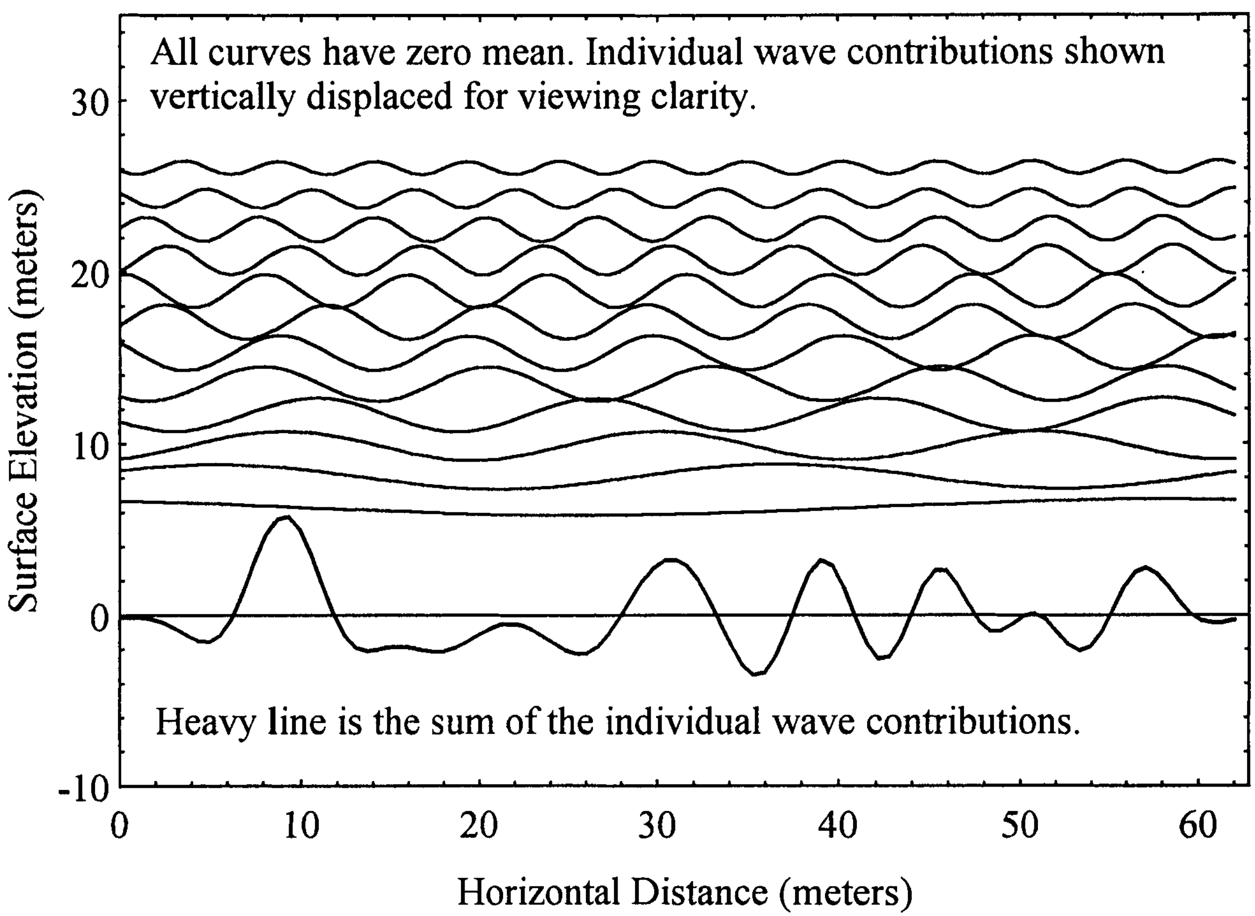
$$\phi = B\cosh\,k(z+h)\,e^{i(kx-\omega t)}$$

$$Bk\sinh\,kh\,e^{i(kx-\omega t)} = B\omega^2\frac{1}{g}\cosh\,kh\,e^{i(kx-\omega t)}$$

$$k\tanh\,kh = \frac{\omega^2}{g} \qquad \omega^2 = gk\,\tanh\,kh$$

$$\zeta = -\frac{1}{g}(-i\omega)B\,e^{i(kx-\omega t)} = \frac{i\omega}{g}B\cosh(kh)\,e^{i(kx-\omega t)} \qquad A = \frac{i\omega}{g}\cosh(kh)\,B$$

Generation of Random Wave Form From Sinusoidal Components



wavedem.ch

Example of Simulation

Suppose a two dimensional (long crested) wave is generated with a wavemaker in a wave tank with an elevation at a specified location given by $z(t)$, where:

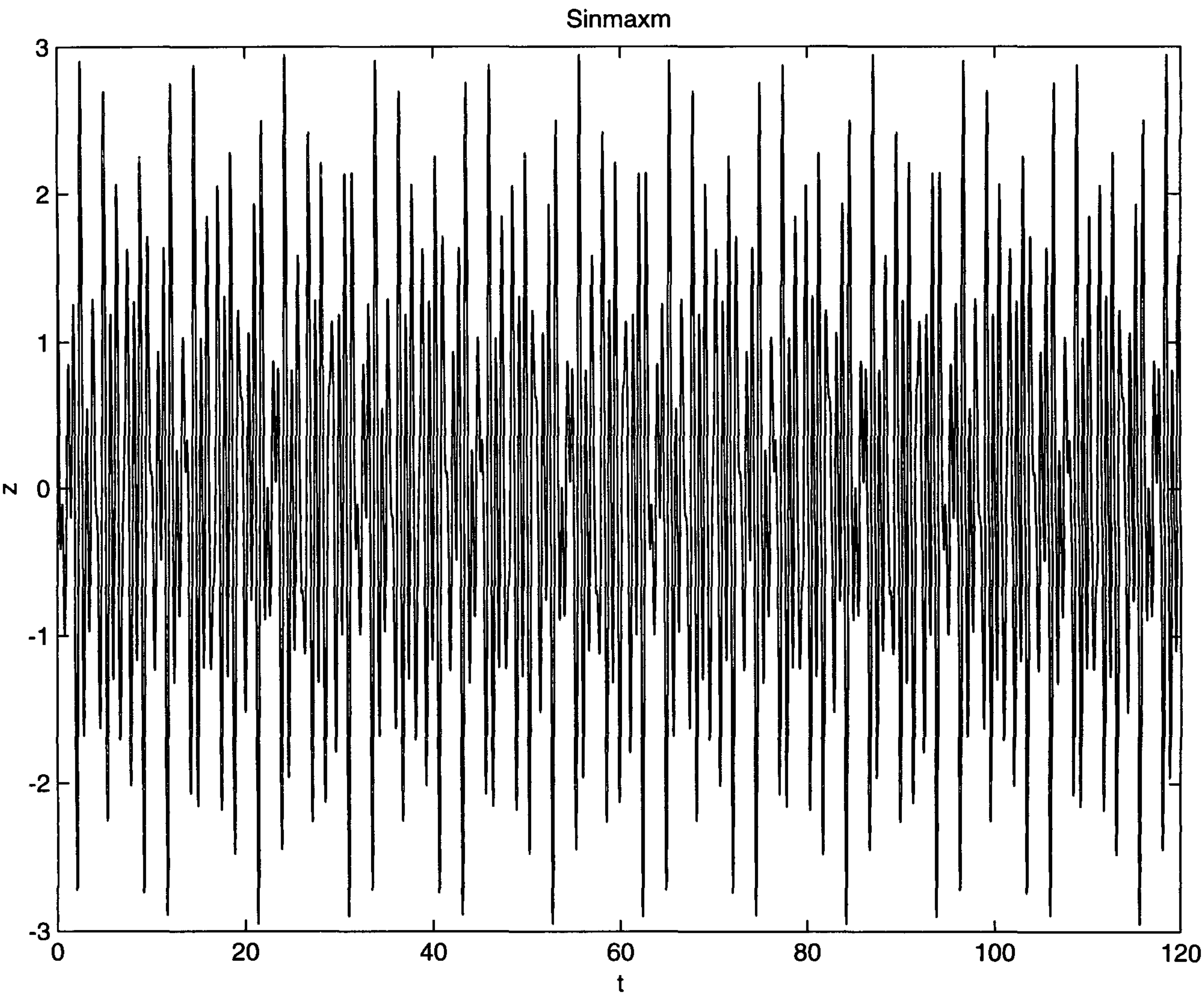
$$z(t) = 0.97\sin(5.2t + 0.82) + 0.99\sin(7.8t + 1.24) + 1.08\sin(9.8t + 2.72)$$

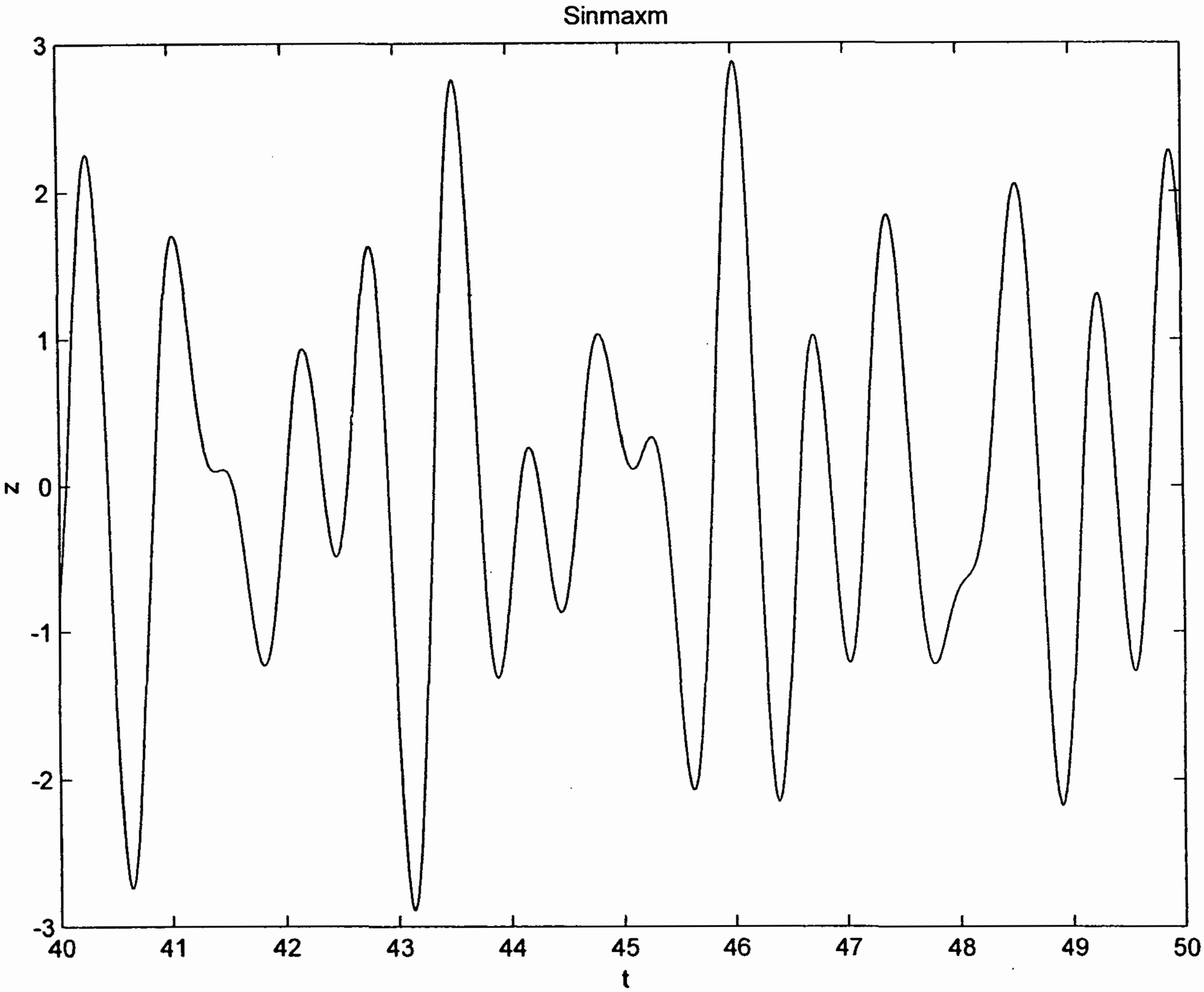
What is the maximum elevation that occurs in the time interval of 0 to 120 seconds (2 minutes). The usual way of finding maxima of analytic functions by setting the derivative to zero is not practical here because there are a great many maxima and the largest of these must be determined. However, because of the great computational speed of common computers, this can be done numerically without much effort.

```
% MATLAB Version of program sinmax
t = 0:0.01:120;
z = 0.97*sin(5.2*t + 0.82) ...
    + 0.99*sin(7.8*t +1.24) + 1.08*sin(9.8*t + 2.72);
zmax = max(z);
mmax= find(z == zmax);
tmax =(t(mmax));
fprintf (1,'tmax = %7.3f     zmax = %8.4f\n',tmax,zmax);
q = [t;z];
fid = fopen('zmaxm.dat','w');
fprintf(fid,'%f7.2f  %8.4f\n',q);
plot(t,z);
xlabel('t');
ylabel('z');
title('Sinmaxm');
```

```
>> sinmaxm
tmax = 118.490     zmax =    2.9447
>>
```

NUmet 8





Sinmaxm
3
2
1
0
-1
-2
-3
z
40
41
42
43
44
45
46
47
48
49
50
t

Sea Spectra

We consider wave fields whose statistics are both stationary and homogeneous in the horizontal plane.

A sea spectrum function $S_T(k, \omega, \theta)$ is a partial description of the statistics of the wave field defined such that $S_T(k, \omega, \theta)\delta k\, \delta\omega\, \delta\theta$ is the contribution to the *average wave energy per unit surface area*, E, in the wavenumber, wave circular frequency and propagation angle bands; $\delta k\, \delta\omega\, \delta\theta$.

For surface elevation $\zeta(\mathbf{x}, t)$ the average wave energy is defined as:

$$E = <\zeta^2>$$

where $< >$ signifies the statistical, temporal or spatial average.

$$\text{Thus:} \quad <\zeta^2> = \int_0^{2\pi} \int_0^{\infty} \int_0^{\infty} S_T(k, \omega, \theta)\delta k\, \delta\omega\, \delta\theta$$

Similar definitions apply when frequency, f, is used instead of circular frequency, ω, and/or when spatial frequency, $\frac{1}{\lambda}$, is used instead of wavenumber, k.

For the frequently encountered case of linear, deep water gravity waves the circular frequency and the wavenumber are related to each other through the *dispersion relation*

$$\omega^2 = gk$$

so that ω and k are not independent of each other. Then the spectrum is a function of only one or the other of these variables and can be written as: $S_t(\omega, \theta)$ or $S_x(k, \theta)$. These functions are related by:

$$S_x(k, \theta) = \frac{g}{2\omega} S_t(\omega, \theta)$$

$$\text{Hence:} \quad <\zeta^2> = \int_0^{2\pi} \int_0^{\infty} S_x(k, \theta) dk d\theta = \int_0^{2\pi} \int_0^{\infty} S_x(\omega, \theta) d\omega d\theta$$

For unidirectional (long crested) seas, all the waves are in a single direction and the spectra are described by $S_t(\omega)$ or $S_x(k)$.

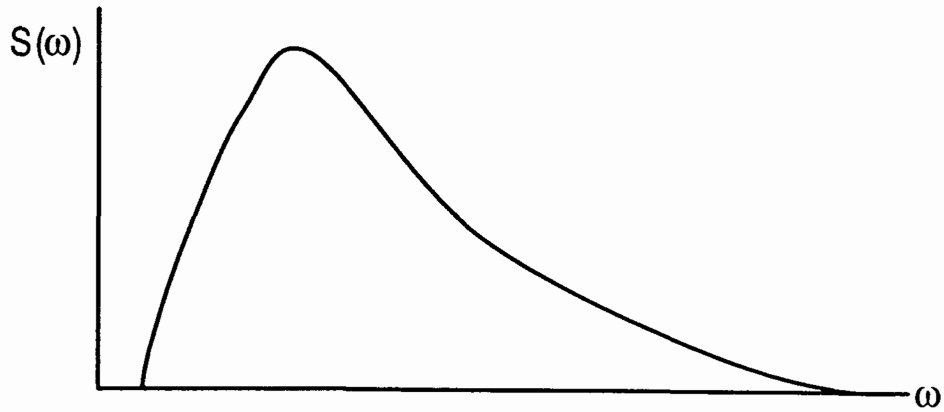
$$<\zeta^2> = \int_0^{\infty} S_t(\omega) d\omega = \int_0^{\infty} S_x(k) dk$$

The fundamental linearized plane progressive wave is:

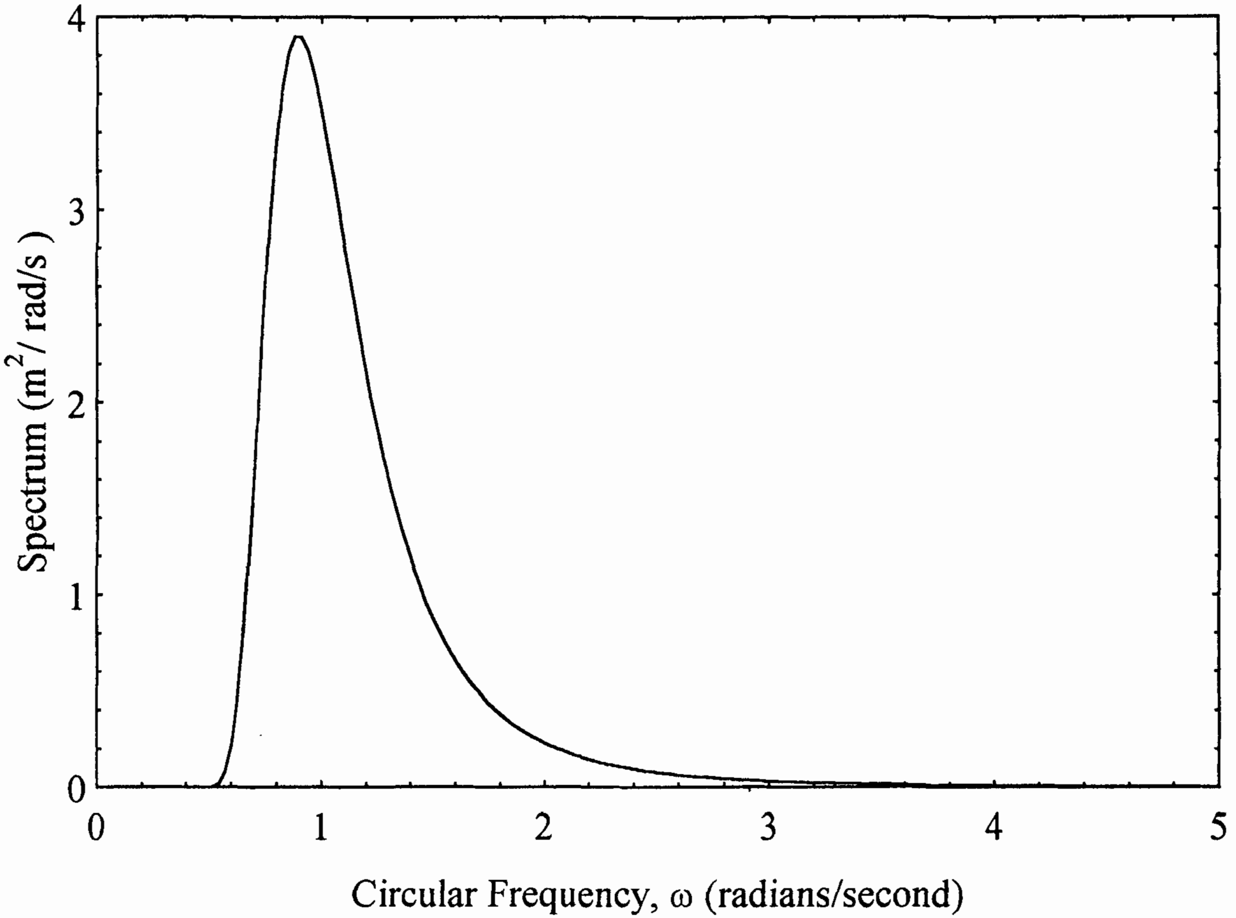
$$\zeta = Ae^{i(kx-\omega t)}$$

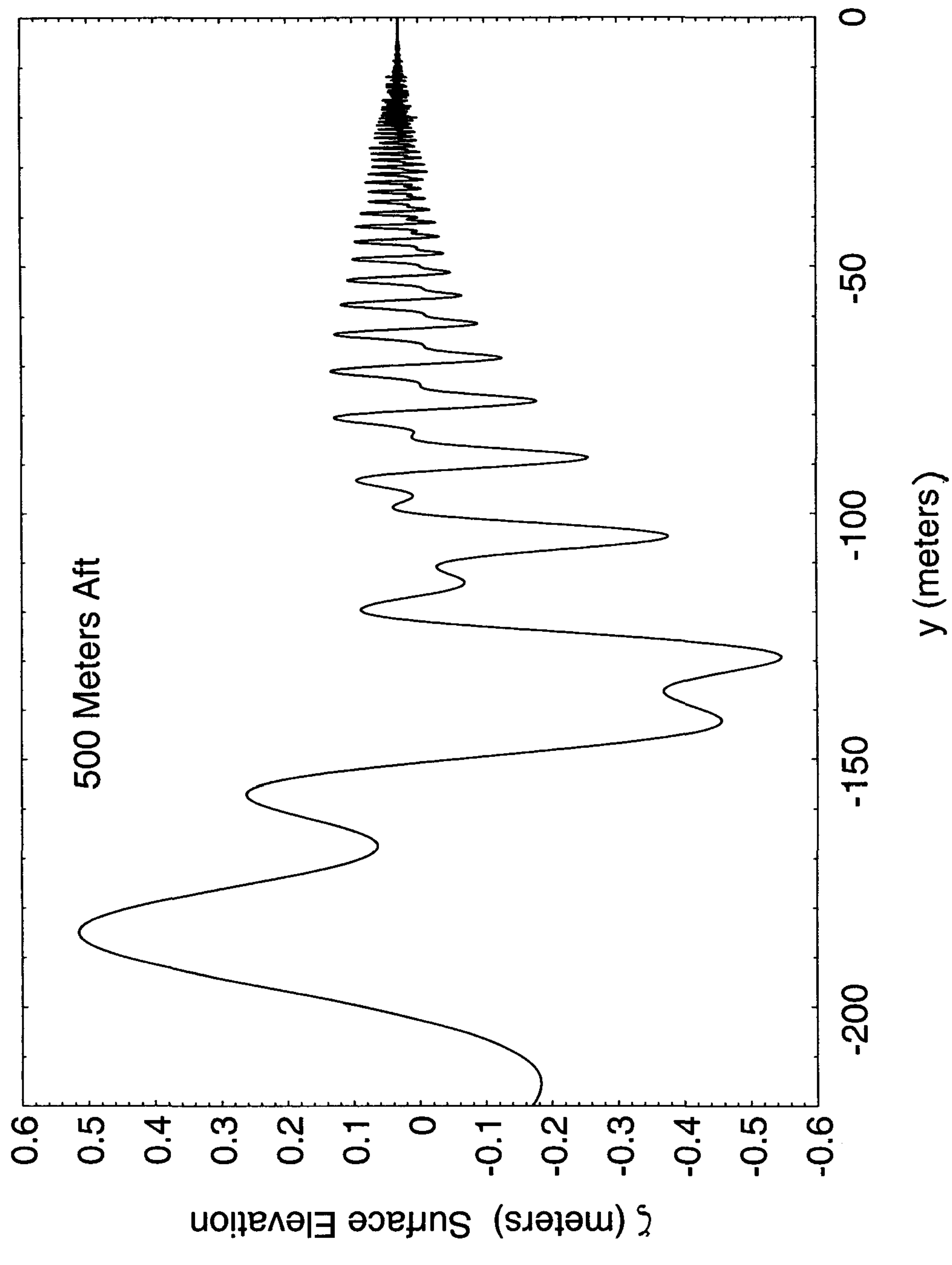
$$\phi = -\frac{i\omega A}{k} e^{kz} e^{i(kx-\omega t)}$$

Random sea waves have spectrum $S(\omega, \theta)$.
For the 2D case the spectrum is $S(\omega)$.

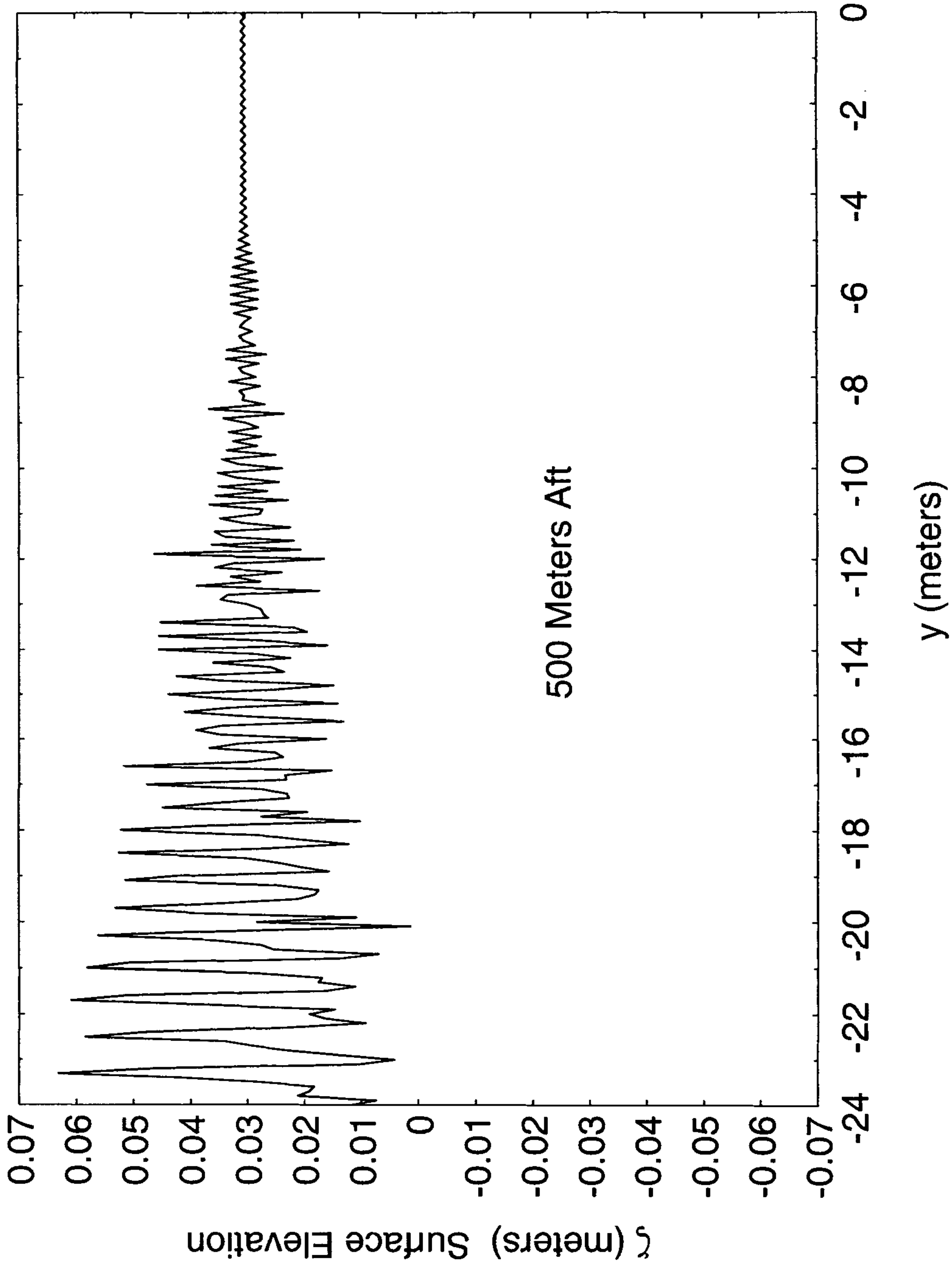


$\int_{\omega_1}^{\omega_2} S(\omega)\, d\omega$ is the contribution to $\overline{\zeta^2}$ of waves with circular frequencies between ω_1 and ω_2.

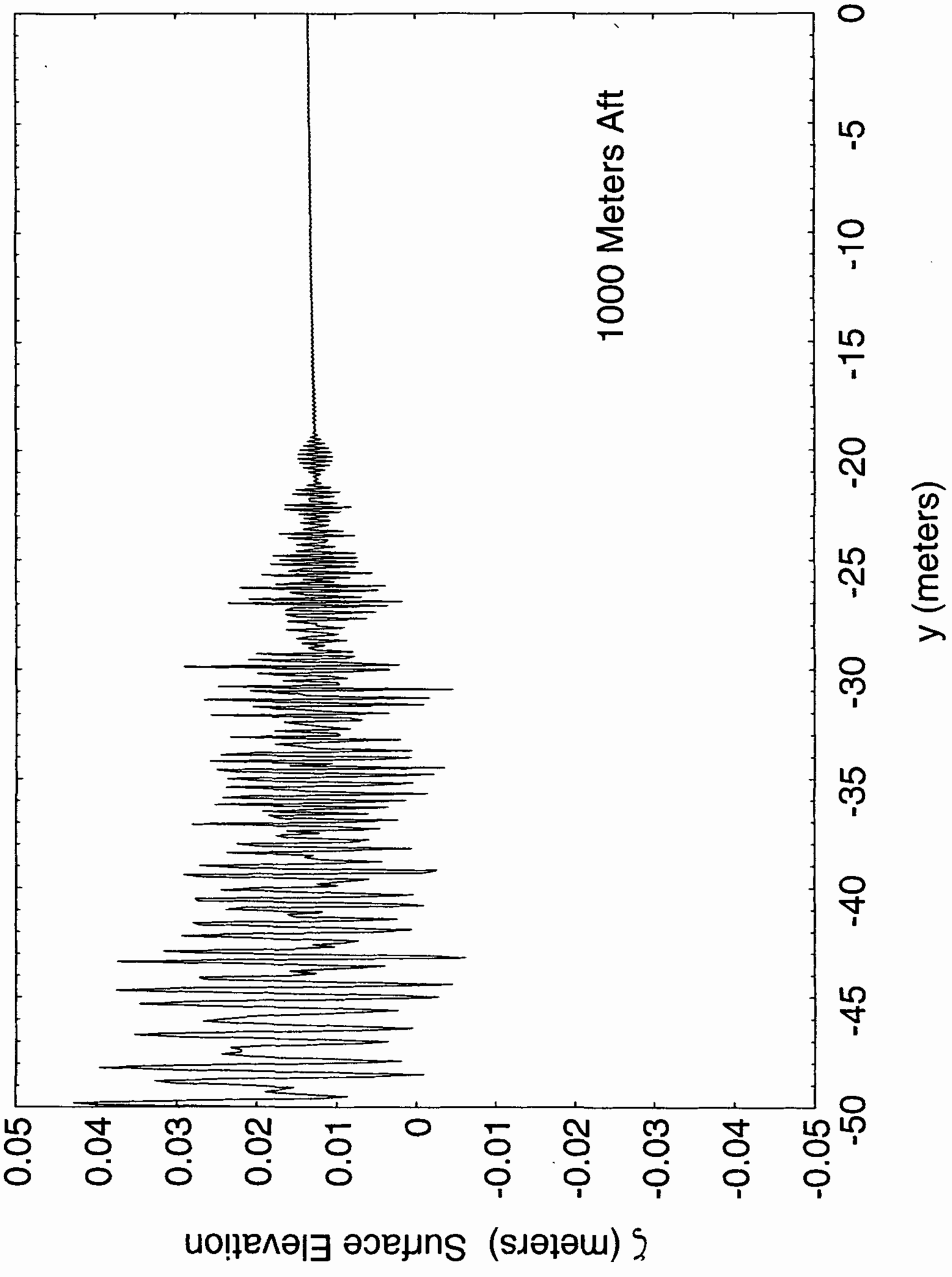

Spectrum (m²/ rad/s)
0
1
2
3
4
0
1
2
3
4
5
Circular Frequency, ω (radians/second)



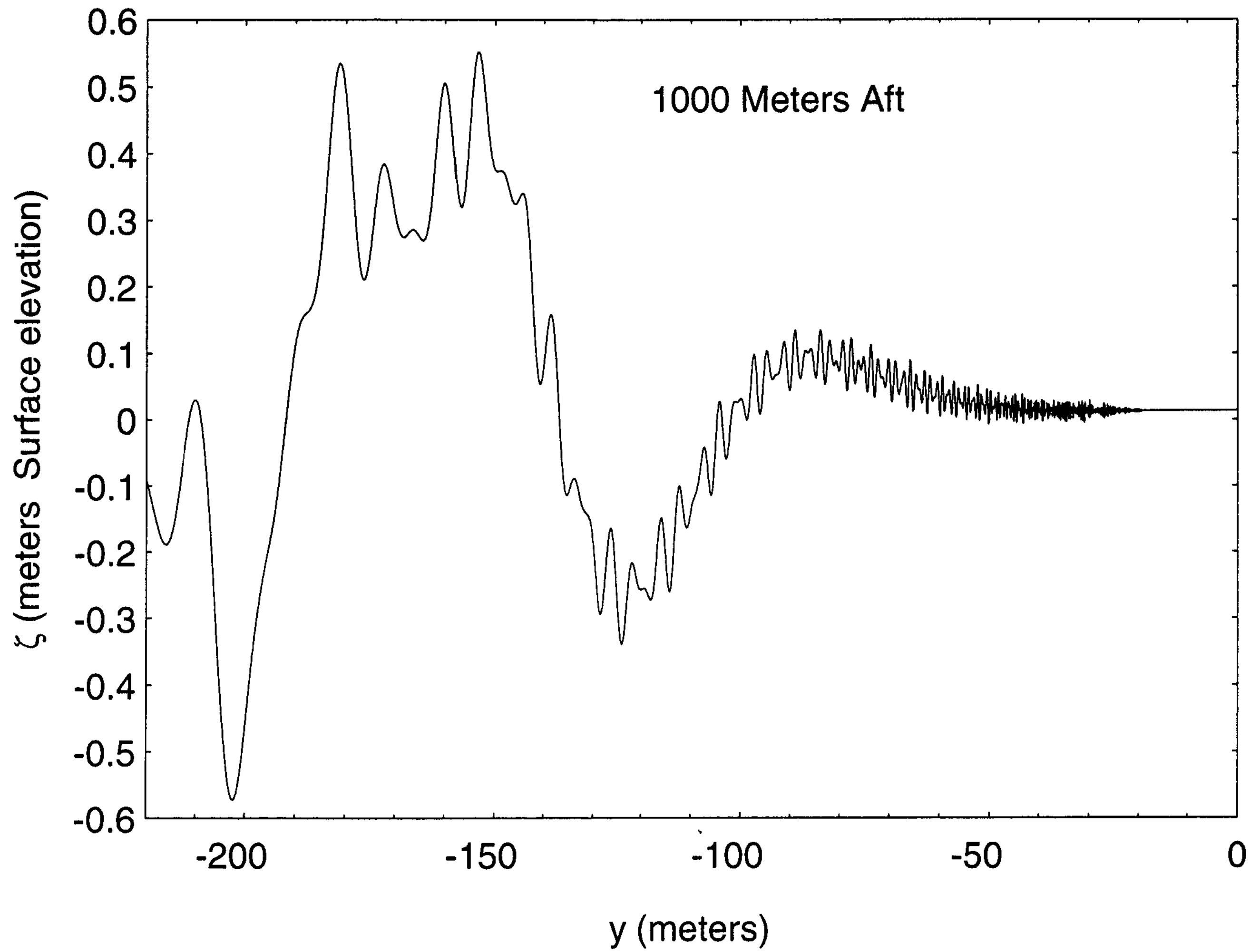
500 Meters Aft
y (meters)
ζ (meters) Surface Elevation
0.6
0.5
0.4
0.3
0.2
0.1
0
-0.1
-0.2
-0.3
-0.4
-0.5
-0.6
-200
-150
-100
-50
0



500 Meters Aft
y (meters)
ζ (meters) Surface Elevation
0.07
0.06
0.05
0.04
0.03
0.02
0.01
0
-0.01
-0.02
-0.03
-0.04
-0.05
-0.06
-0.07
-24
-22
-20
-18
-16
-14
-12
-10
-8
-6
-4
-2
0



1000 Meters Aft
y (meters)
ζ (meters) Surface Elevation
0.05
0.04
0.03
0.02
0.01
0
-0.01
-0.02
-0.03
-0.04
-0.05
-50
-45
-40
-35
-30
-25
-20
-15
-10
-5
0


1000 Meters Aft
ζ (meters Surface elevation)
y (meters)
0.6
0.5
0.4
0.3
0.2
0.1
0
-0.1
-0.2
-0.3
-0.4
-0.5
-0.6
-200
-150
-100
-50
0

Fourier Transforms

Fourier Transforms are valuable tools in numerical hydrodynamics because a number of problems can be described in the form of Fourier transforms and they can be computed very quickly by the Fast Fourier Transform (FFT) method. Two of these problems are solving a certain class of differential equations, and in simulating sea waves.

$$X(f) = \mathcal{F}x(t) = \int_{-\infty}^{\infty} x(t)e^{-i2\pi ft}dt$$

$$x(t) = \mathcal{F}^{-1}X(f) = \int_{-\infty}^{\infty} X(f)e^{i2\pi ft}df$$

As an example, consider a differential equation with constant coefficients of the form:

$$A_n\frac{d^ny}{dx^n} + A_{n-1}\frac{d^{n-1}y}{dx^{n-1}} + ... + A_0y(x) = g(x)$$

Consider Fourier Transforms from x to f where:

$$\mathcal{F}[y(x)] \equiv Y(f) \qquad \text{and} \qquad \mathcal{F}[g(x)] \equiv G(f)$$

Take the Fourier transform of the differential equation to get:

$$(i2\pi f)^nA_nY(f) + (i2\pi f)^{n-1}A_{n-1}Y(f) + ... + A_0Y(f) = G(f)$$

This is an algebraic equation which can be numerically solved for $Y(f)$:

$$Y(f) = \frac{G(f)}{(i2\pi f)^nA_n + (i2\pi f)^{n-1}A_{n-1} + A_0}$$

$y(x)$ can be determined by inverse Fourier transformation. Not only is this less computationally intensive than solving the differential equation by direct numerical methods, but the error in the integration rule is avoided.

Fourier Transforms (continued)

$$X(f) = \mathcal{F}x(t) = \int_{-\infty}^{\infty} x(t)e^{-i2\pi ft}dt$$

$$x(t) = \mathcal{F}^{-1}X(f) = \int_{-\infty}^{\infty} X(f)e^{i2\pi ft}df$$

Suppose $x(t) \approx 0$ for $t < 0$ and $t > T$. Then:

$$X(f) \approx \int_{0}^{T} x(t)e^{-i2\pi ft}dt$$

Also, suppose x(t) is *band limited* such that: $X(f) = 0$ for $|f| \geq F_{max}$. Then:

$$x(t) = \mathcal{F}^{-1}X(f) = \int_{-Fmax}^{Fmax} X(f)e^{i2\pi ft}df$$

Now, consider a periodic function having period t that is identical to $x(t)$ for $0 \leq t \leq T$. This function has a Fourier series given by:

$$x(t) = \sum_{n=-\infty}^{\infty} A_n e^{i2\pi nt/T}, \qquad A_n = \frac{1}{T}\int_{0}^{T} x(t)e^{-i2\pi nt/T}dt$$

The expression for A_n is identical to $\frac{1}{T}$ times the Fourier Transform evaluated at $f = \frac{n}{T}$. These Fourier coefficients, $A_n = \frac{1}{T}X\left(\frac{n}{T}\right)$ can be numerically evaluated very quickly by an algorithm called the *Fast Fourier Transform* (FFT).

From the A_n's, the function $x(t)$ can be constructed over the t-range $0 < t < T$. Outside this range the reconstruction is periodic whereas the real value of $x(t) \approx 0$.

Evaluate A_n by the following rectangular rule integration:

$$\delta t = \frac{1}{2F_{max}} \qquad t = j\delta t \qquad j_{max} \equiv N \qquad T = N\delta t \qquad x_j \equiv x(j\delta t)$$

$$A_n = \frac{1}{N\delta t}\sum_{j=0}^{N-1} x(j\delta t)\exp\left[-\frac{i2\pi nj\delta t}{N\delta t}\right]\delta t = \frac{1}{N}\sum_{j=0}^{N-1} x_j e^{-i2\pi nj/N}$$

$$f_t = \frac{n_t}{T} \qquad n_{max} = M$$

$$F_{max} = \frac{M}{T} \qquad M = T F_{max}$$

$$\delta t = \frac{1}{2 F_{max}} \quad \text{; Sampling Theorem}$$

$$F_{max} = \frac{1}{2\delta t} \qquad M = \frac{T}{2\delta t} \qquad \boxed{\delta t = \frac{1}{2F_{max}}}$$

$$F_{max} = M\,\delta f \qquad \frac{1}{2\delta t} = \frac{T}{2\delta t}\,\delta f$$

$$\boxed{\delta f = \frac{1}{T}}$$

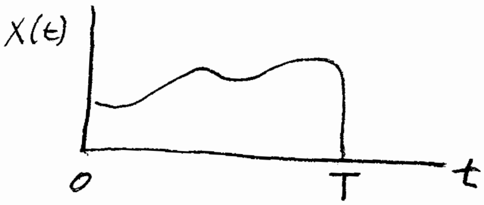
$$t = j\,\delta t \qquad T = j_{max}\,\delta t$$

$$M = \frac{j_{max}\,\delta t}{2\,\delta t} \qquad j_{max} = 2M$$

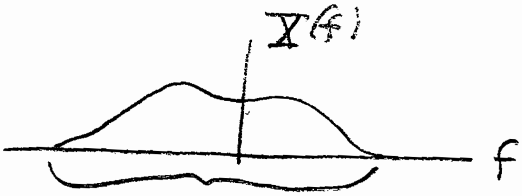
Let $2M = N$ $\qquad j_{max} = N$

$$\boxed{\delta t = \frac{T}{2M} = \frac{T}{N}}$$

$$\delta f = \frac{1}{T} = \frac{1}{M/F_{MAX}} = \frac{F_{max}}{M} = \frac{2F_{max}}{N}$$



$T = N\,\delta t$



$2F_{max} = N\,\delta f$

$$-\frac{i2\pi nt}{T} = -\frac{i2\pi nj\delta t}{N\delta t} = -\frac{i2\pi nj}{N}$$

$$F_{MAX} = M\delta f = M\frac{1}{T} = M\frac{1}{N\delta t} = \frac{1}{2\delta t}$$

$$f = n\delta f = \frac{n}{T}$$

$$n_{max} = TF_{max} = T\frac{1}{2\delta t} = \frac{N}{2}$$

$$\delta t = \frac{T}{N} \qquad df = \frac{1}{T} \qquad dt\,df = \frac{1}{N}$$

Fourier Transforms (continued)

$$X(f) = 0 \text{ for } f \geq F_{max} = \frac{1}{2\delta t} \text{ and } f = \frac{n}{T}, \text{ so } \delta f = \frac{1}{T} \text{ and } n_{max} = T f_{max} = \frac{N}{2}$$

$$\delta t = \frac{1}{2F_{max}}, \qquad \delta f \delta t = \frac{1}{2F_{max}T} = \frac{1}{N}$$

$$x(j\delta t) = x_j = \sum_{n=-N/2}^{N/2} T A_n \exp\left[\frac{i2\pi nj\delta t}{N\delta t}\right]\frac{1}{T} = \sum_{n=-N/2}^{N/2} A_n e^{i2\pi nj/N}$$

Fast Computing

Computing speed is minimized by minimizing the number of complex exponentials that must be computed.

Let: $q_1 = e^{-i2\pi/N}$ $\qquad q_2 = e^{i2\pi/N}$

$$e^{-i2\pi nj/N} = e^{-i2\pi(n-1)j/N}\, q_1^j = e^{-i2\pi n(j-1)/N}\, q_1^n$$

$$e^{i2\pi nj/N} = e^{i2\pi(n-1)j/N}\, q_2^j = e^{i2\pi n(j-1)/N}\, q_2^n$$

Even the powers of q can be avoided:

$$e^{-i2\pi(0)(0)/N} = 1$$

$$e^{-i2\pi(1)(1)/N} = e^{-i2\pi(0)(0)/N}\, q_1$$

$$e^{-i2\pi(1)(2)/N} = e^{-i2\pi(1)(1)/N}\, q_1$$

$$e^{-i2\pi(2)(1)/N} = e^{-i2\pi(1)(1)/N}\, q_1$$

$$e^{-i2\pi(1)(3)/N} = e^{-i2\pi(1)(2)/N}\, q_1$$

$$e^{-i2\pi(2)(2)/N} = e^{-i2\pi(1)(3)/N}\, q_1$$

etc.

Fourier Transforms (continued)

Periodicity

The actual integral transforms are of limited extent.

$$x(t) = 0 \text{ except for } 0 \le t \le T$$

$$X(f) = 0 \text{ except for } -F_{max} \le f \le F_{max}$$

However, the mathematical constructions, while consistent with the integral transforms for : $0 \le t \le T$, and $-Fmax \le f \le Fmax$, are periodic outside these ranges.

$$A_{n+N} = \frac{1}{N}\sum_{j=0}^{N-1} x_j e^{-i2\pi(n+N)j/N} = \frac{1}{N}\sum_{j=0}^{N-1} x_j e^{-i2\pi nj/N} e^{-i2\pi j} = \frac{1}{N}\sum_{j=0}^{N-1} x_j e^{-i2\pi nj/N} = A_n$$

$$x_{j+N} = \sum_{n=-N/2}^{N/2} A_n e^{i2\pi n(j+N)/N} = \sum_{n=-N/2}^{N/2} A_n e^{i2\pi nj/N} e^{i2\pi n} = \sum_{n=-N/2}^{N/2} A_n e^{i2\pi nj/N} = x_j$$

$$\text{Therefore: } x_j = \sum_{n=0}^{N-1} A_n e^{i2\pi nj/N}$$

Computational FFT and IFFT of REAL Numbers

$$A_n = \frac{1}{N}\sum_{j=0}^{N-1} x_j e^{-i2\pi nj/N} \quad ; \quad x_j = \sum_{n=-\frac{N}{2}}^{N/2} A_n e^{i2\pi nj/N}$$

$$A_{\frac{N}{2}} = \frac{1}{N}\sum_{j=0}^{N-1} x_j e^{i2\pi j \frac{N}{2}\frac{1}{N}} = \frac{1}{N}\sum_{j=0} x_j e^{i\pi j}$$

If x_j's are real $\quad A_{\frac{N}{2}}$ is real

$A_{-1} = A_{N-1}$

$A_{-2} = A_{N-2} \Rightarrow A_{-K} = A_{N-K}$

$A_{-\frac{N}{2}} = A_{\frac{N}{2}}$

also, if the x_j are real,

$$A_{-n} = \frac{1}{N}\sum_{j=0}^{N-1} x_j e^{i2\pi(-n)j/N} = A_n^*$$

Since $e^{i2\pi(-n)j/N} = e^{i2\pi(N-n)j/N}$

$$A_{-n} e^{i2\pi(-n)j/N} = A_n^* e^{i2\pi(N-j)/N}$$

$$A_{-\frac{N}{2}} e^{i2\pi(+\frac{N}{2})j/N} = A_{\frac{N}{2}} e^{i2\pi(N-\frac{N}{2})j/N}$$
$$= A_{N/2} e^{i2\pi N/2\, j/N}$$

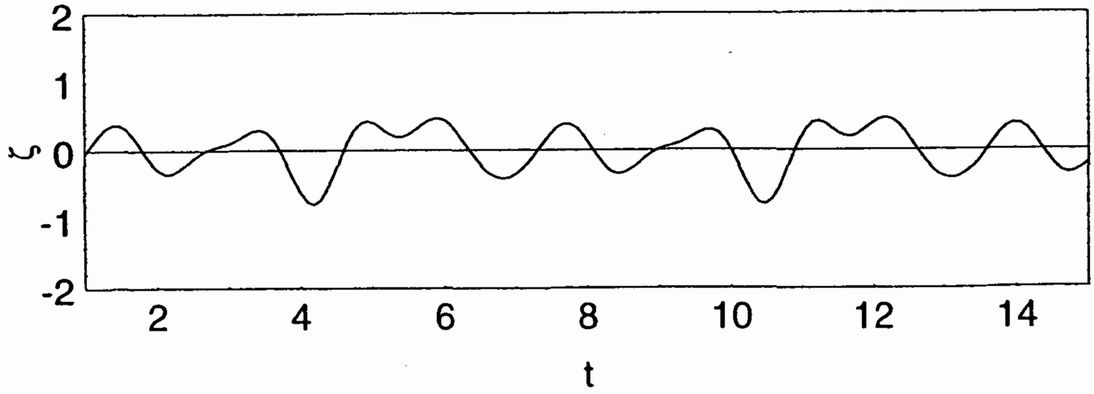
Therefore, $x_j = \sum_{n=0}^{N-1} A_n' e^{i2\pi nj/N}$

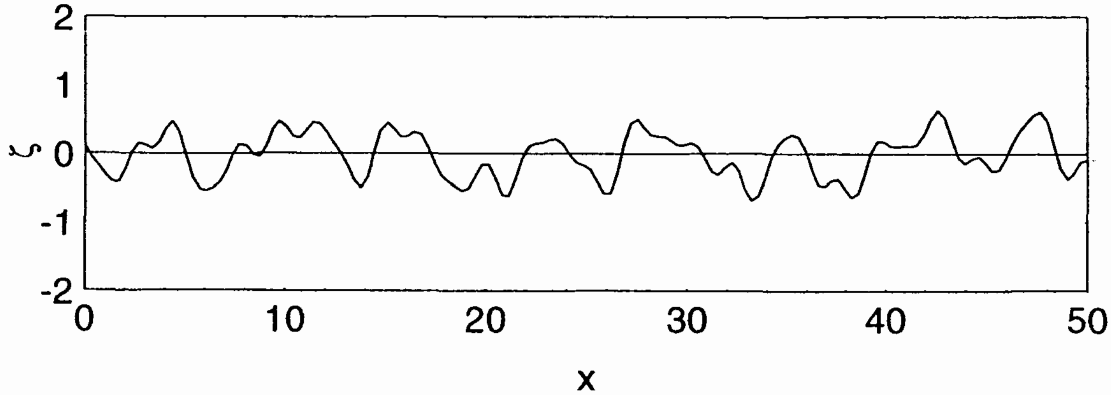
where $A_n' = \begin{cases} A_n, & 0 \le n' \le N/2 - 1 \\ 2A_n, & n' = N/2 \\ A_{N-n'}^*, & \frac{N}{2} < n' < N \end{cases}$

ws1

Simulation of Random Waves

Here we consider two-dimensional (long crested) waves. The waves are approximated as hydrodynamically linear in the sense that wave breaking and other nonlinear effects are neglected.





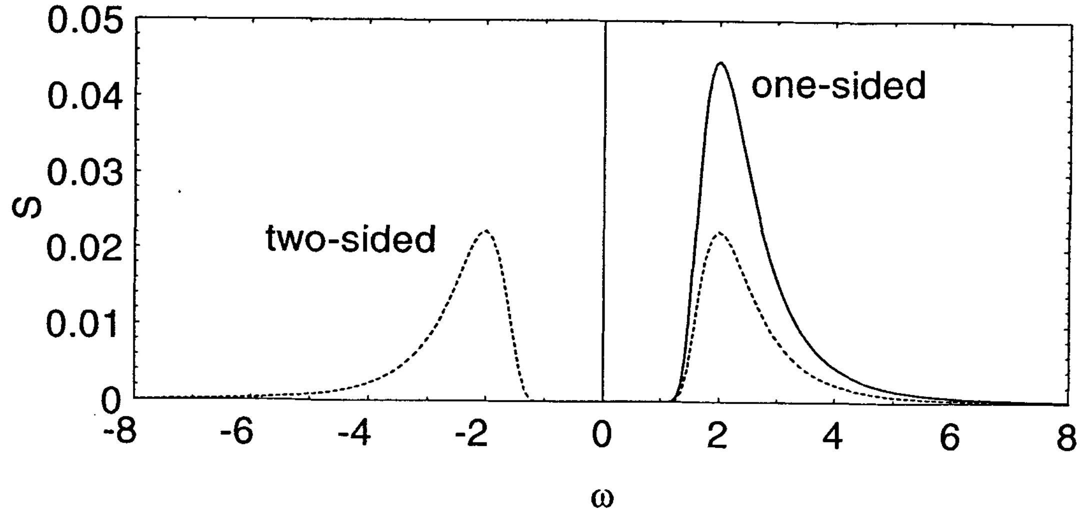
$$\zeta(x,t) = \sum_{n=0}^{\infty} Z_n \cos\left(-\frac{\omega_n^2}{g}x + \omega_n t + \alpha_n\right)$$

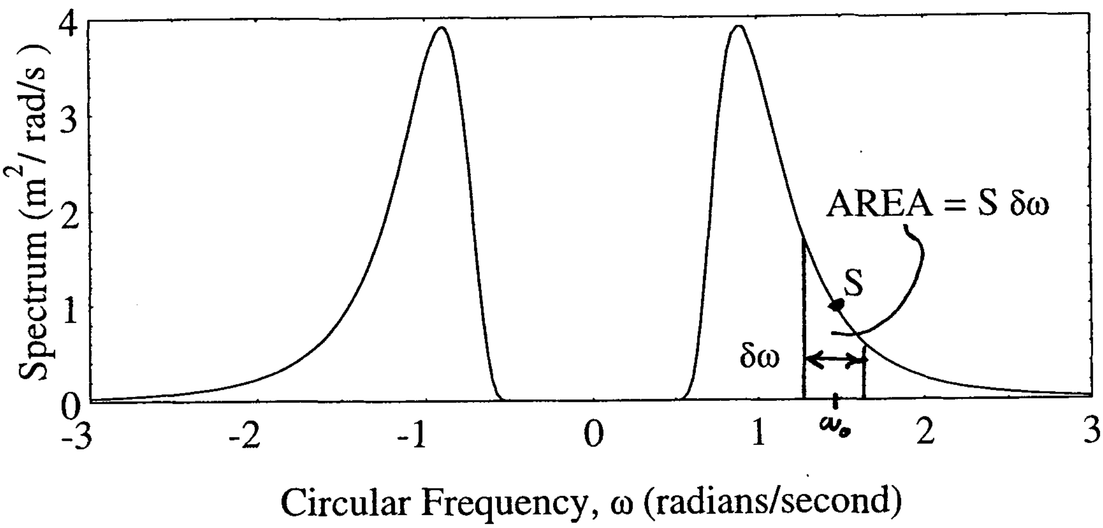
where the Z_n's are chosen to provide the desired wave spectrum and the α_n's are random numbers uniformly distributed on $0 \le \alpha < 2\pi$.

An alternate expression is: $\zeta(x,t) = \sum_{n=-\infty}^{\infty} Z_n \exp\left[i\left(-\frac{\omega_n^2}{g}x + \omega_n t + \alpha_n\right)\right]$

Combining $e^{i\alpha_n}$ into Z_n, the surface elevation vs time at $x = 0$ is:

$$\zeta(t) = \sum_{n=-\infty}^{\infty} Z_n e^{i\omega_n t}$$





The region in the "almost trapezoid" is represented by a sinusoidal wave having frequency ω_o and the same energy, E, of this region of the spectrum. The sinusoidal wave $Ae^{i\omega_o t}$ has energy $|A^2|$. Thus,

$$|A^2| = S(\omega_o)\delta\omega$$

The waves are random processes and can be represented in two different ways. One way is to have stochastic waves and a stochastic spectrum whose expectation is equal to the spectrum being simulated (Type 1). The other way has stochastic waves and a deterministic spectrum equal to the spectrum being simulated (type 2).

Similarly, at $t = 0$ the surface elevation vs x is:

$$\zeta(x) = \sum_{n=-\infty}^{\infty} Z'_n e^{-ik_n x} = \sum_{n=-\infty}^{\infty} Z_n e^{ik_n x} \quad \text{where } k_n = \frac{\omega_n^2}{g}$$

With $\omega_n = 2\pi n\delta f$, $k_n = 2\pi n\delta b$, $(b = 1/\lambda)$, $t = j\delta t$, $x = j\delta x$, and n limited to $-\frac{N}{2} \leq n \leq \frac{N}{2}$ with $\delta f\,\delta t$ or $\delta b\,\delta x$ equal to $1/N$, the expressions for ζ have the form of an inverse discrete Fourier transform. Hence, by first choosing the Z_n's so they are consistent with the wave spectrum, the surface elevation for all values of t or for all values of x can be computed very rapidly by using an FFT program.

Either set $Z_{-n} = Z_n^*$ or use non-negative n and take the real (or imaginary) part.

We will use the method in which $Z_{-n} = Z_n^*$.

This corresponds to a two-sided spectrum whose levels are half the levels of the corresponding 1-sided spectrum.

WS3

Type 1

At a fixed value of x, the sea elevation is $\zeta(t)$ which is a sample function of a random process having a 2-sided power density function, $S_w(\omega)$. The associated 1-sided spectrum is $S_W(\omega) = 2S_w(\omega)$ for $\omega = 0$. The fourier transform of $\zeta(t)$ is $Z(\omega)$. The spectrum and the Fourier transform of $\zeta(t)$ are truncated at $|\omega| = \omega_c = 2\pi f_c$.

$\zeta(t)$ is discretized with the time interval $\delta t = \pi/\omega_c$ to satisfy the sampling theorem. Thus, $\zeta(t)$ is specified at the discrete times $\zeta_j = \zeta(j\delta t),\ j = 0, 1, 2, ..., N$.

The Fourier coefficient Z_j corresponds to the circular frequency $\omega_j = j\delta\omega$, $j = -\frac{N}{2} + 1, ..., 0, ...\frac{N}{2}$, where $\delta\omega = \frac{2\pi}{N\delta t}$. N is usually chosen as a power of 2 for computational efficiency.

For the Type 1 approach, each Fourier coefficient is separated into its real and imaginary parts and each of these is an uncorrelated Gaussian variate.

$$Z_j = Z_{r_j} + iZ_{i_j}$$

Z_{r_j} and Z_{i_j} are identically distributed with the probability density function:

$$p(Z_{r_j}) = \frac{1}{\sigma_j\sqrt{2\pi}} \exp\left(-\frac{Z_{r_j}^2}{2\sigma_j^2}\right)$$

From the physics of the modeling, where here E means "Expectation":

$$E\left[|Z_j^2|\right] = S_w(\omega_j)\delta\omega$$

$$E\left[Z_{r_j}^2\right] = E\left[Z_{i_j}^2\right] = \frac{1}{2}S_w(\omega_j)\delta\omega$$

From the mathematics of the Gaussian pdf: $\sigma_j^2 = E\left[Z_{r_j}^2\right]$

$$\sigma_j = \sqrt{\frac{1}{2}S_w(\omega_j)\delta\omega} = \sqrt{\frac{1}{4}S_W(|\omega_j|)\delta\omega}$$

There are computer programs which give Gaussian distributed random numbers for which the user specifies σ_j.

Type 2 $\quad Z_j = e^{i\alpha_j}\sqrt{S_w(\omega_j)\delta\omega} = e^{i\alpha_j}\sqrt{\frac{1}{2}S_W(\omega_j)\delta\omega}\,, \qquad \omega_j \geq 0$

α_j is uniformly distributed on $0 \leq \alpha_j < 2\pi$ and can be obtained from a random number computer program.

We truncate the spectrum at frequencies $\pm N/2\delta\omega$.

Thus the expression for a simulated two-dimensional (long-crested) random wave elevation at a point on the ocean surface is:

$$\zeta(t) = \sum_{-N/2}^{N/2} e^{i\alpha_n} \sqrt{\frac{1}{2} S_W(|n\delta\omega|)\delta\omega}\; e^{i(n\delta\omega)t}$$

where α^n is a random number, $\leq \alpha_n < 2\pi$, and $\alpha_n = -\alpha_{-n}$.

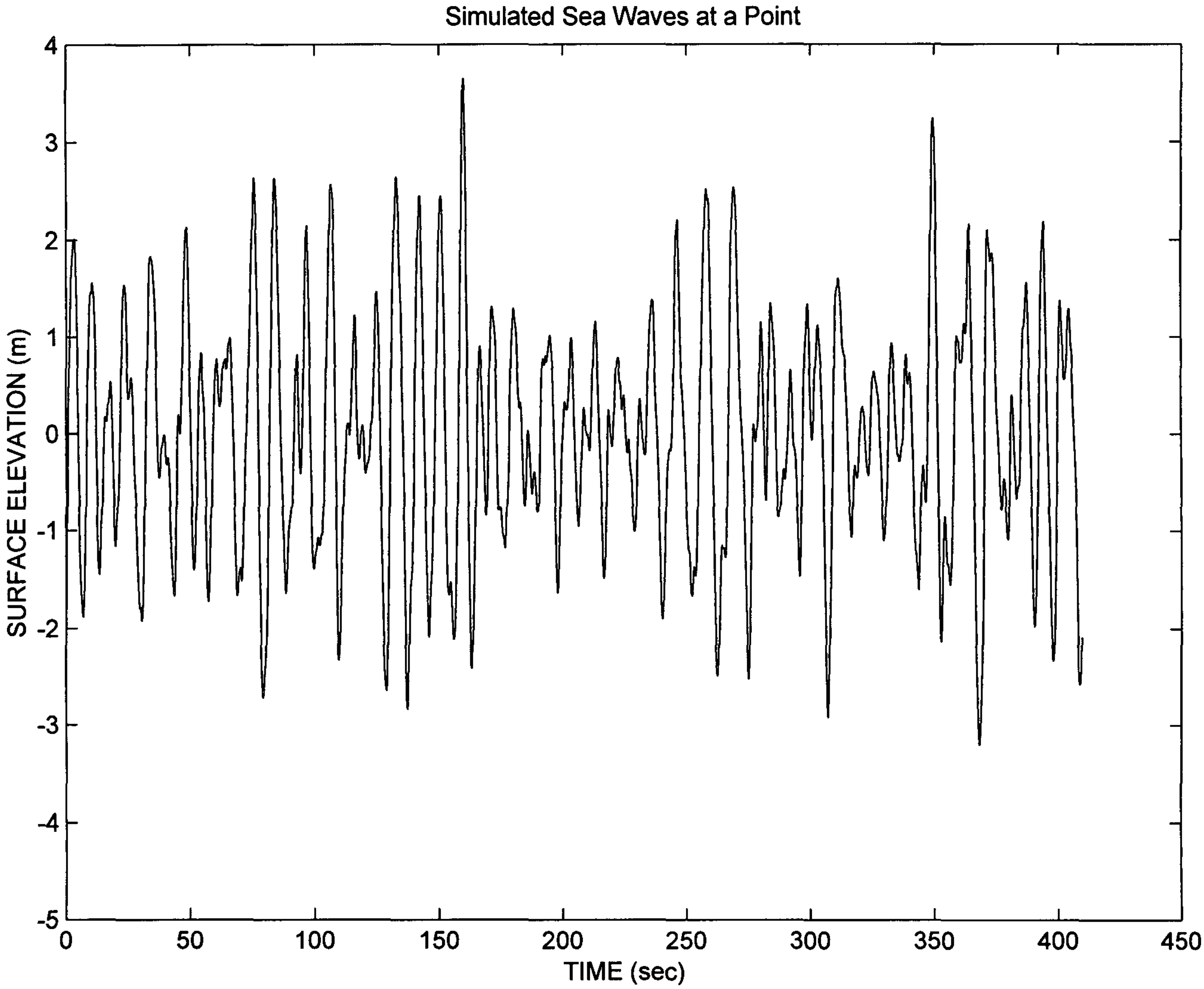
This can be extended to a long-crested wave *field*, dependent on x and t as:

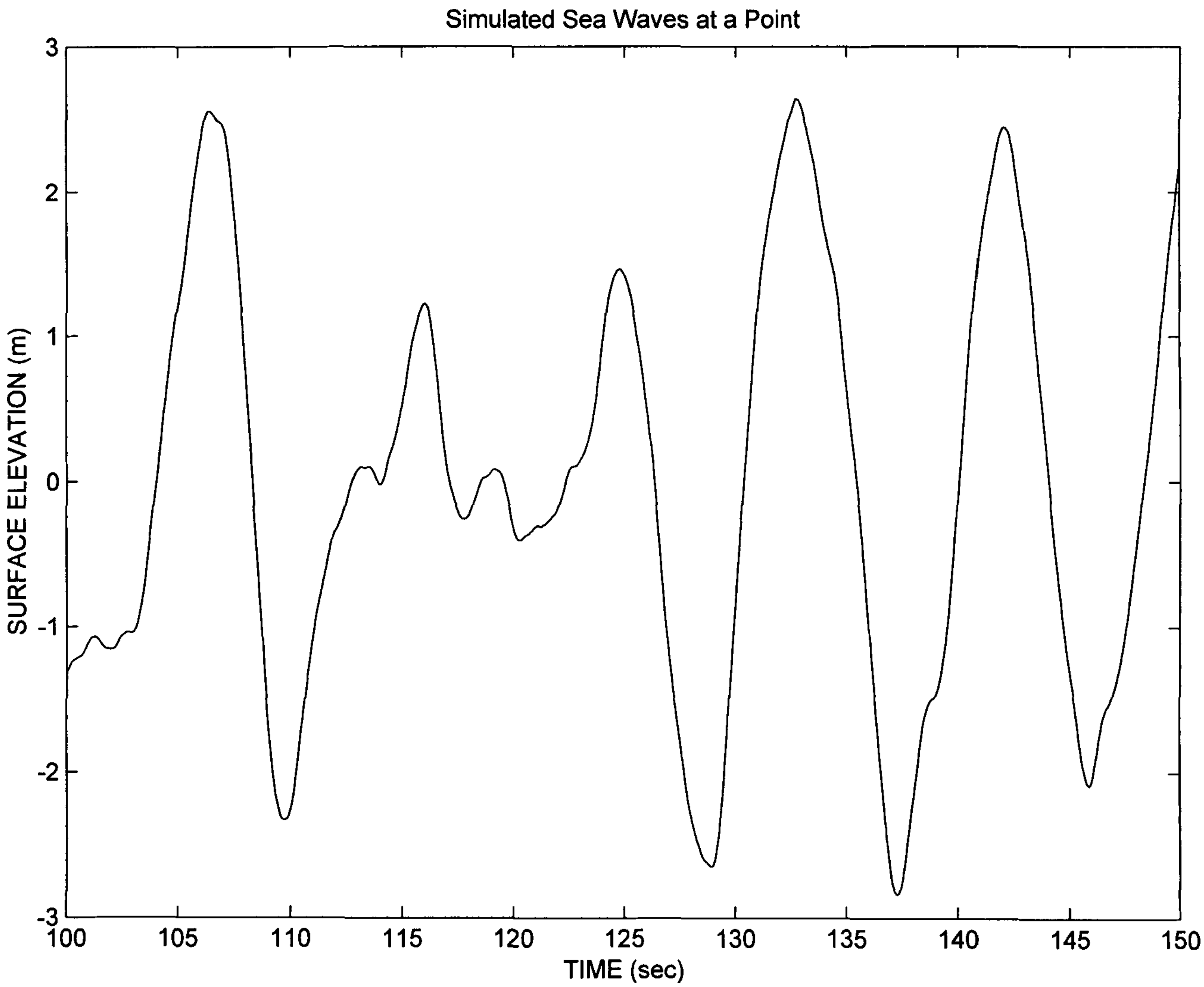
$$\zeta(x,t) = \sum_{-N/2}^{N/2} e^{i\alpha_n} \sqrt{\frac{1}{2} S_W(|n\delta\omega|)\delta\omega}\; e^{i[(n\delta\omega)t-(n\delta\omega)|n\delta\omega|x/g]}$$

This is because $|k| = \omega^2/g$.

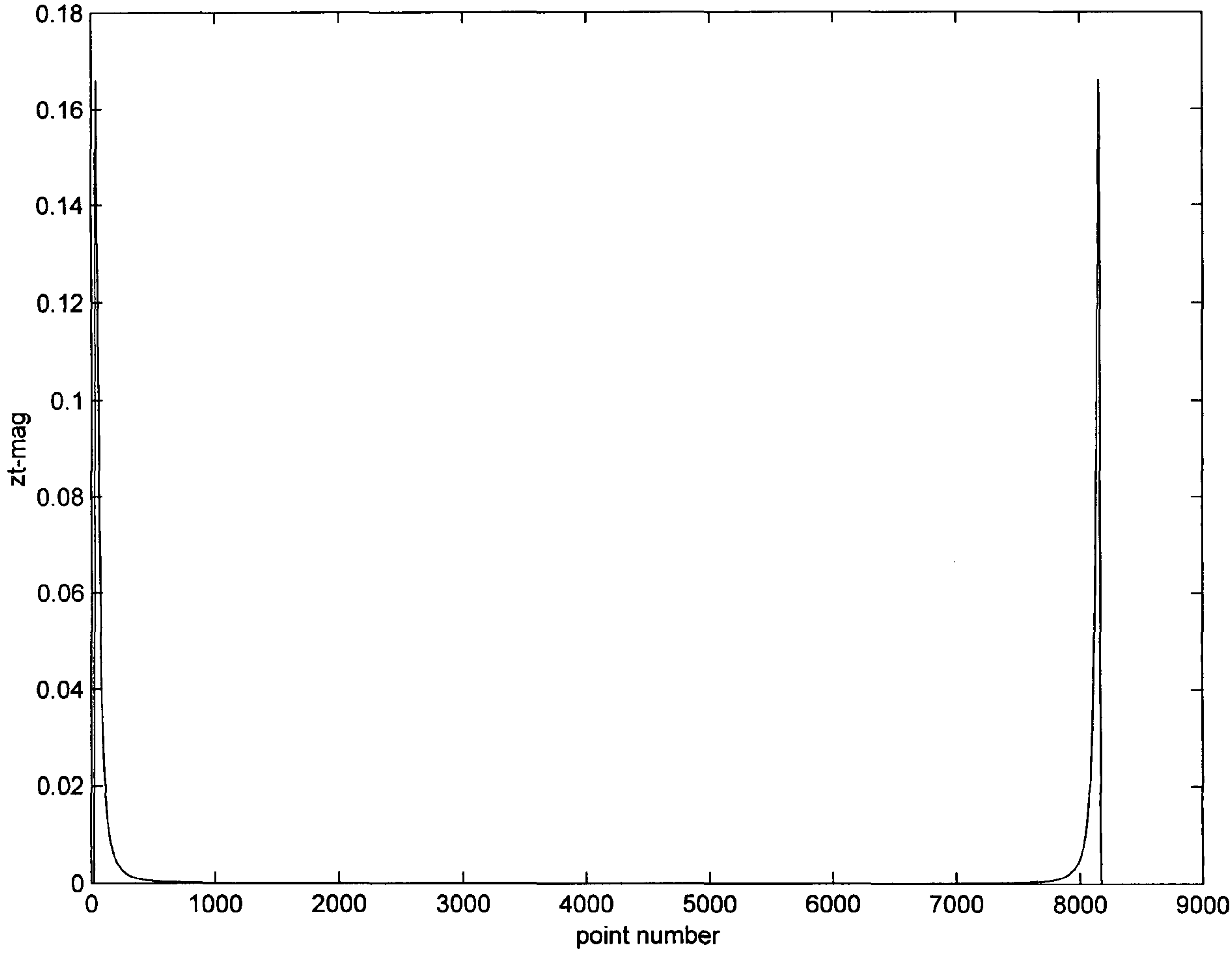
wavesims.m

```
% Wavesims
dt = 0.05;
npts = 8192;
nptso2 = npts/2.0;
tr = dt* npts; %dt * 8192 for 8192 total points
t = 0:dt:(tr-dt);
tp4 =  (2.0*pi) .^ 4;
g = 9.81;
v = 15.0;
df = 1.0/tr;
ffold = df * nptso2; %df * 4096 for 8192 total points
f = 0:df:ffold;
f = f+eps;
fac1 = 0.0081 *g*g/tp4;
fac2 = 0.74 *(g/v)^4/tp4;
s = 0.5*fac1 ./ f.^5 .* exp( -fac2 ./f .^ 4);
rand ('state',sum(100*clock));
p = 2.0 * pi * rand(1,nptso2);
p(nptso2+1) = 0.0;     %4097 for 8192 total points
z = exp(i*p) .* sqrt(s*df);
zt = [ z  conj(fliplr(z(2:4096)))];
zeta = real(fft(zt));
%The above gives same result as zeta = npts*real(ifft(zt))
plot (t,zeta);
xlabel('TIME (sec)')
ylabel('SURFACE ELEVATION (m)');
title('Simulated Sea Waves at a Point');
```

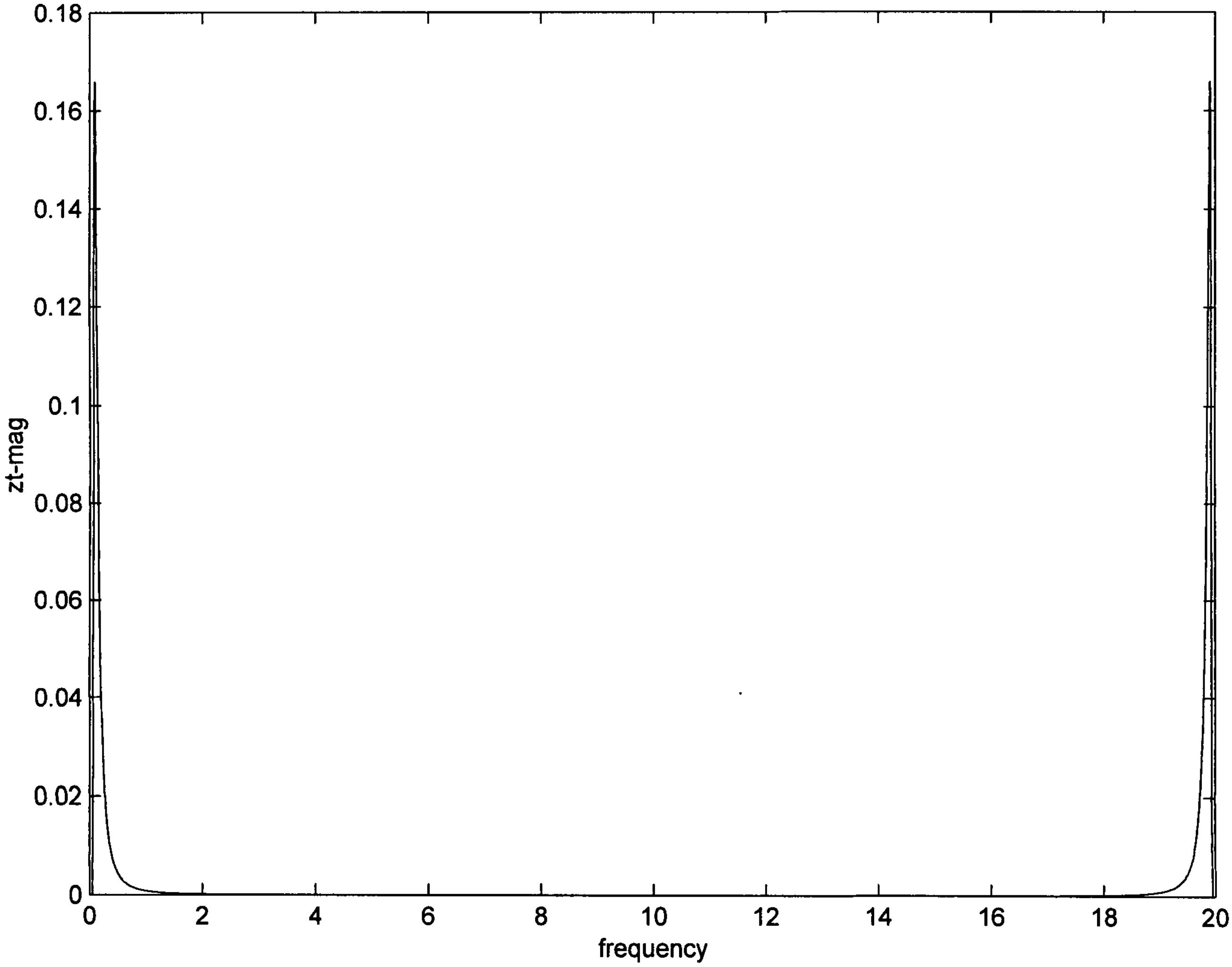

Simulated Sea Waves at a Point
SURFACE ELEVATION (m)
TIME (sec)
4
3
2
1
0
-1
-2
-3
-4
-5
0
50
100
150
200
250
300
350
400
450

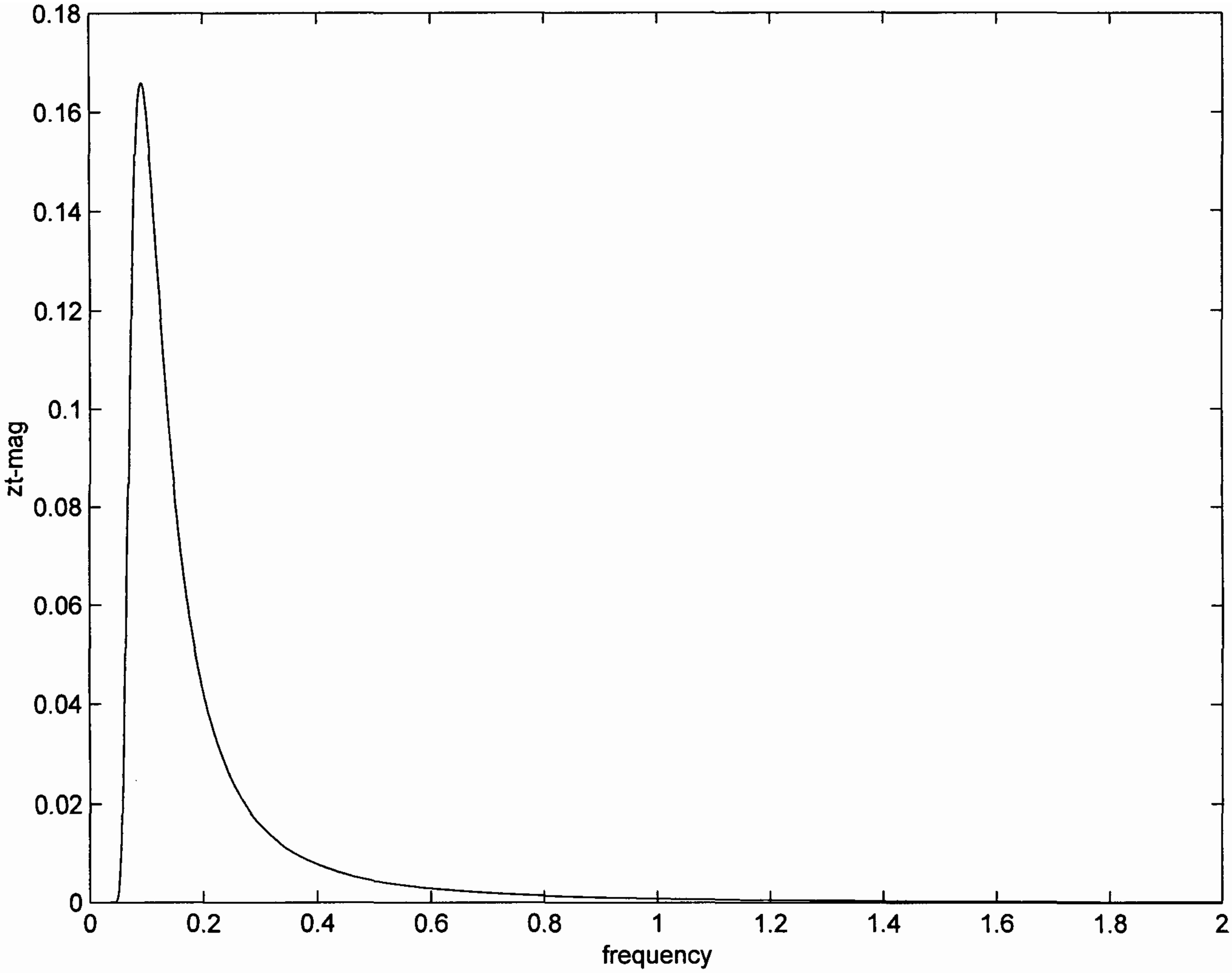


Simulated Sea Waves at a Point
SURFACE ELEVATION (m)
3
2
1
0
-1
-2
-3
100
105
110
115
120
125
130
135
140
145
150
TIME (sec)



0.18
0.16
0.14
0.12
0.1
0.08
0.06
0.04
0.02
0
zt-mag
0
1000
2000
3000
4000
5000
6000
7000
8000
9000
point number


0.18
0.16
0.14
0.12
0.1
0.08
0.06
0.04
0.02
0
zt-mag
0
2
4
6
8
10
12
14
16
18
20
frequency


0.18
0.16
0.14
0.12
0.1
0.08
0.06
0.04
0.02
0
zt-mag
0
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6
1.8
2
frequency

Review of Fourier Transforms, Inverse Fourier Transforms, FFT'S, IFFT'S and Wave Simulation

$$X(f) = \int_{-\infty}^{\infty} x(t) e^{-i2\pi f t} dt, \quad x(t) = \int_{-\infty}^{\infty} X(f) e^{i2\pi f t} df$$

Consider functions of the form:

$x(t) = 0$ except for $0 \le t \le T$

$X(f) = 0$ except for $-F_{max} \le f \le F_{max}$

Then:

$$X(f) = \int_0^T x(t) e^{-i2\pi f t} dt, \quad x(t) = \int_{-F_{max}}^{F_{max}} X(f) e^{i2\pi f t} df$$

Construct a periodic function, $x_p(t)$ of period T

$x_p(t) = x(t)$, for $0 \le t < T$

$x_p(t)$ Has a Fourier Series Representation

$$x_p(t) = \sum_{n=-\infty}^{\infty} A_n e^{i2\pi f t}$$

where: $A_n = \frac{1}{T} \int_0^T x(t) e^{-i2\pi \frac{n}{T} t} dt$

Note that $X(\frac{n}{T}) = T A_n$

If T is very large, values of $\frac{n}{T}$ $n = 0, 1, 2, \ldots$ are dense so $X(f)$ can be determined from the Fourier Coefficients at closely spaced frequencies.

Note that $A_n = \frac{1}{T} X(\frac{n}{T})$

Since $f = \frac{n}{T}$, $A_n = 0$ for $|n| > T f_{max}$, also $\boxed{\delta f = \frac{1}{T}}$

In the range $0 \le t \le T$, $x(t) = \sum_{n=-Tf_{max}}^{Tf_{max}} A_n e^{i2\pi f t}$

Let $T f_{max} = M$ $\quad x(t) = \sum_{n=-M}^{M} A_n e^{i2\pi f t}$

$2M = N$

Set $\boxed{dt = \frac{1}{2 f_{max}}}$, $t = j\,\delta t$, $x(j\,\delta t) \equiv x_j$, $\boxed{dt = \frac{T}{N}}$

We need to Evaluate $i2\pi f t = i2\pi n \delta f \, j \frac{T}{N}$

$= i2\pi n \frac{1}{T} j \frac{T}{N} = i2\pi \frac{nj}{N}$

Discretized approximation of $A_n = \frac{1}{T} X(\frac{n}{T})$

$A_n \cong \frac{1}{N \delta T} \sum_{n=0}^{N-1} x_j e^{-i2\pi nj/N}$

$x(j\,\delta t) = x_j = \sum_{n=-N/2}^{N/2} A_n e^{i2\pi nj/N} = \sum_{n=0}^{N-1} A'_n e^{i2\pi nj/N}$

In $x(t)$ is real,

$$A'_n = \begin{cases} A_n, & 0 \le n \le \frac{N}{2} - 1 \\ 2A_n, & n = N/2 \\ A^*_{N-n}, & \frac{N}{2} < n < N \end{cases}$$

To simulate waves having a one-sided frequency spectrum $S_F(f)$ whose equivalent two-sided spectrum is $S_f(f) = \frac{1}{2} S_F(|f|)$;

the elevation $\zeta(t)$ at a point is

$$\zeta(j\delta t) = \sum_{n=0}^{N-1} A_n' e^{i 2\pi n j/N} \quad , \; \delta t = \frac{T}{N}$$

where:

$$A_n' = e^{i\alpha_n} \sqrt{S_f(n\delta f)\, \delta f}$$

$$= e^{i\alpha_n} \sqrt{\tfrac{1}{2} S_F(n\delta f)\, \delta f}$$

where: α_n is a random number in the range $0 \leq \alpha_n < 2\pi$ and the rules for A_n' are as given on the previous page for real ζ.

This is precisely the form of an inverse Fast Fourier transform

Generating Gaussian Random Numbers

This note is about the topic of generating **Gaussian** pseudo-random numbers given a source of **uniform** pseudo-random numbers. This topic comes up more frequently than I would have expected, so I decided to write this up on *one* of the best ways to do this. At the end of this note there is a list of references in the literature that are relevant to this topic. You can see some code examples that implement the technique, and a step-by-step example for generating **Weibull** distributed random numbers.

There are many ways of solving this problem (see for example **Rubinstein, 1981**, for an extensive discussion of this topic) but we will only go into one important method here. If we have an equation that describes our desired distribution function, then it is possible to use some mathematical trickery based upon the *fundamental transformation law of probabilities* to obtain a transformation function for the distributions. This transformation takes random variables from one distribution as inputs and outputs random variables in a new distribution function. Probably the most important of these transformation functions is known as the **Box-Muller** (1958) transformation. It allows us to transform uniformly distributed random variables, to a new set of random variables with a Gaussian (or Normal) distribution.

The most basic form of the transformation looks like:

```
y1 = sqrt( - 2 ln(x1) ) cos( 2 pi x2 )
y2 = sqrt( - 2 ln(x1) ) sin( 2 pi x2 )
```

We start with *two* independent random numbers, x1 and x2, which come from a uniform distribution (in the range from 0 to 1). Then apply the above transformations to get two new independent random numbers which have a Gaussian distribution with zero mean and a standard deviation of one.

This particular form of the transformation has two problems with it,

1. It is slow because of many calls to the math library.
2. It can have numerical stability problems when x1 is very close to zero.

These are serious problems if you are doing stochastic modelling and generating millions of numbers.

The **polar form** of the Box-Muller transformation is both faster and more robust numerically. The algorithmic description of it is:

```
float x1, x2, w, y1, y2;

do {
        x1 = 2.0 * ranf() - 1.0;
        x2 = 2.0 * ranf() - 1.0;
        w = x1 * x1 + x2 * x2;
} while ( w >= 1.0 );

w = sqrt( (-2.0 * log( w ) ) / w );
y1 = x1 * w;
y2 = x2 * w;
```

where **ranf()** is the routine to obtain a random number uniformly distributed in [0,1]. The polar form is faster because it does the equivalent of the sine and cosine geometrically without a call to the trigonometric function library. But because of the possiblity of many calls to **ranf()**, the uniform wss

random number generator should be fast (I generally recommend R250 for most applications).

Probability transformations for Non Gaussian distributions

Finding transformations like the Box-Muller is a tedious process, and in the case of empirical distributions it is not possible. When this happens, other (often approximate) methods must be resorted to. See the reference list below (in particular **Rubinstein, 1981**) for more information.

There are other very useful distributions for which these probability transforms *have* been worked out. Transformations for such distributions as the **Erlang**, **exponential**, **hyperexponential**, and the Weibull distribution can be found in the literature (see for example,**MacDougall, 1987**).

Useful References

Box, G.E.P, M.E. Muller 1958; **A note on the generation of random normal deviates**, Annals Math. Stat, V. 29, pp. 610-611

Carter, E.F, 1994; The Generation and Application of Random Numbers , Forth Dimensions Vol XVI Nos 1 & 2, Forth Interest Group, Oakland California

Knuth, D.E., 1981; **The Art of Computer Programming, Volume 2 Seminumerical Algorithms**, Addison-Wesley, Reading Mass., 688 pages, ISBN 0-201-03822-6

MacDougall,M.H., 1987; **Simulating Computer Systems**, M.I.T. Press, Cambridge, Ma., 292 pages, ISBN 0-262-13229-X

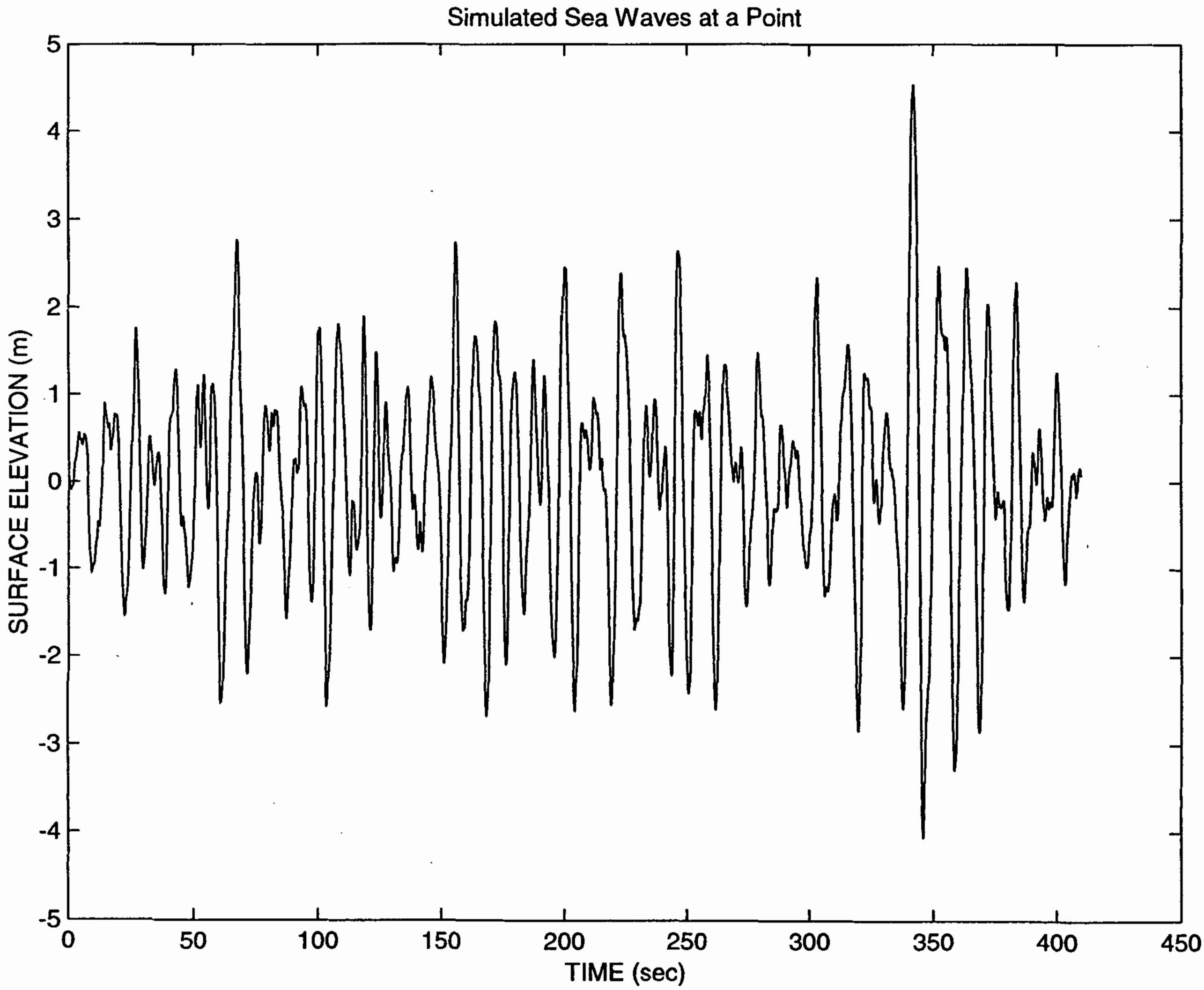
Press, W.H., B.P. Flannery, S.A. Teukolsky, W.T. Vetterling, 1986; **Numerical Recipes, The Art of Scientific Computing**, Cambridge University Press, Cambridge, 818 pages, ISBN 0-512-30811-9

Rubinstein, R.Y., 1981; **Simulation and the Monte Carlo method**, John Wiley & Sons, ISBN 0-471-08917-6

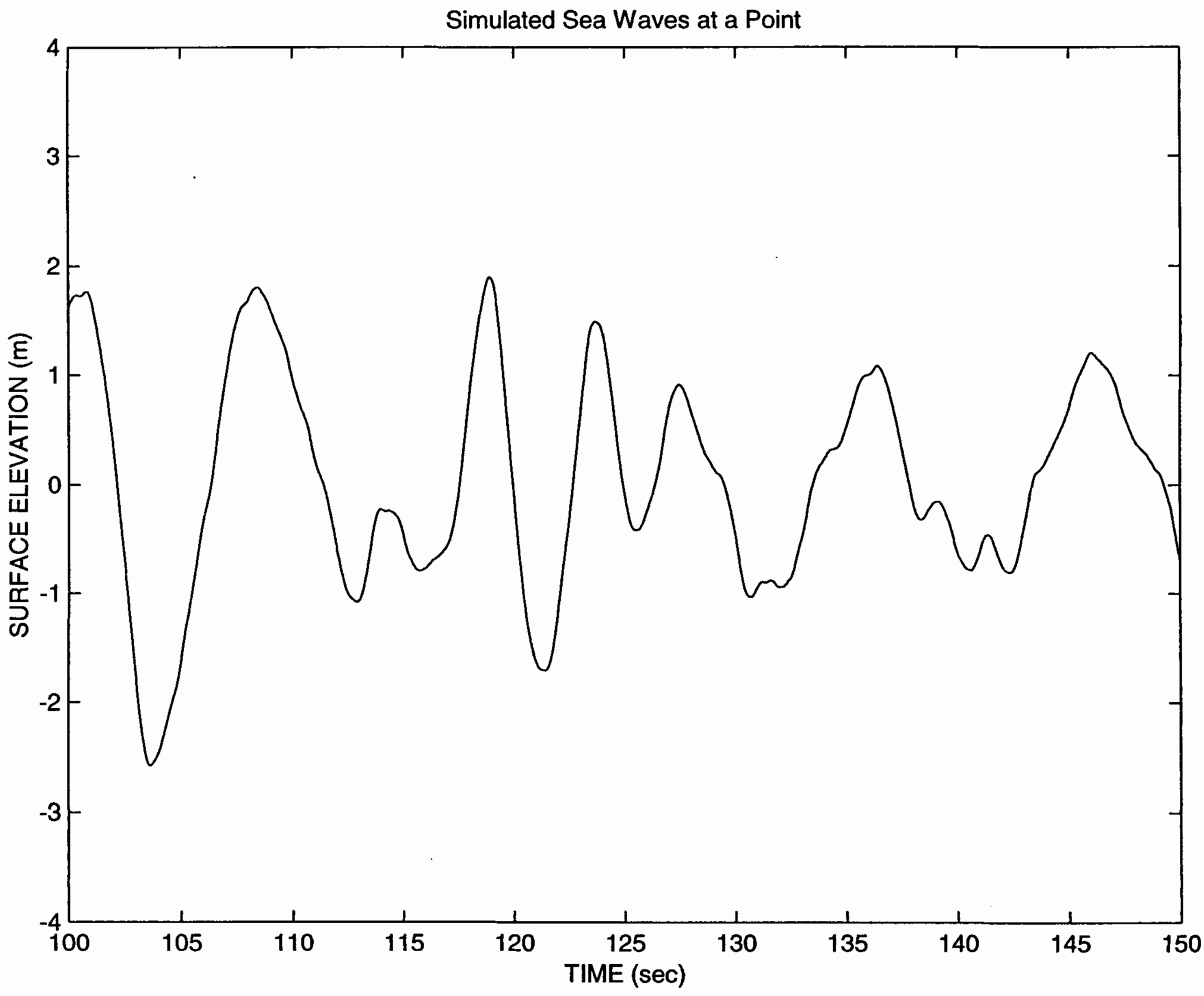
See Also: A Reference list of papers on Random Number Generation.

```
Everett (Skip) Carter
Taygeta Scientific Inc.
                        UUCP:     ...!uunet!taygeta!skip
                        WWW: http://www.taygeta.com/
```

```
/* boxmuller.c           Implements the Polar form of the Box-Muller
                         Transformation

                       (c) Copyright 1994, Everett F. Carter Jr.
                           Permission is granted by the author to use
                     this software for any application provided this
                     copyright notice is preserved.

*/

#include <math.h>


extern float ranf();         /* ranf() is uniform in 0..1 */


float box_muller(float m, float s)  /* normal random variate generator */
{                                   /* mean m, standard deviation s */
      float x1, x2, w, y1;
      static float y2;
      static int use_last = 0;

      if (use_last)                  /* use value from previous call */
      {
            y1 = y2;
            use_last = 0;
      }
      else
      {
            do {
                  x1 = 2.0 * ranf() - 1.0;
                  x2 = 2.0 * ranf() - 1.0;
                  w = x1 * x1 + x2 * x2;
            } while ( w >= 1.0 );

            w = sqrt( (-2.0 * log( w ) ) / w );
            y1 = x1 * w;
            y2 = x2 * w;
            use_last = 1;
      }

      return( m + y1 * s );
}
```



Simulated Sea Waves at a Point
SURFACE ELEVATION (m)
5
4
3
2
1
0
-1
-2
-3
-4
-5
0
50
100
150
200
250
300
350
400
450
TIME (sec)


Simulated Sea Waves at a Point
SURFACE ELEVATION (m)
4
3
2
1
0
-1
-2
-3
-4
100
105
110
115
120
125
130
135
140
145
150
TIME (sec)

WS 7

Wave Statistics

One way to calculate wave statistics is directly from long-term simulations.

Example What is the expected value of the largest wave elevation in a day?

Solution by simulation from a known wave spectrum.

1. Simulate waves for many days.
2. List the largest elevation in each day.
3. Calculatethe average of the values in the list.

Another Example What is the probability that the largest wave elevation in one day is less than the value V. Solution by simulation.

1. Simulate waves for many days.
2. Determine the fraction of days that the elevation does not exceed V.
3. This fraction is an estimate of the desired probability.

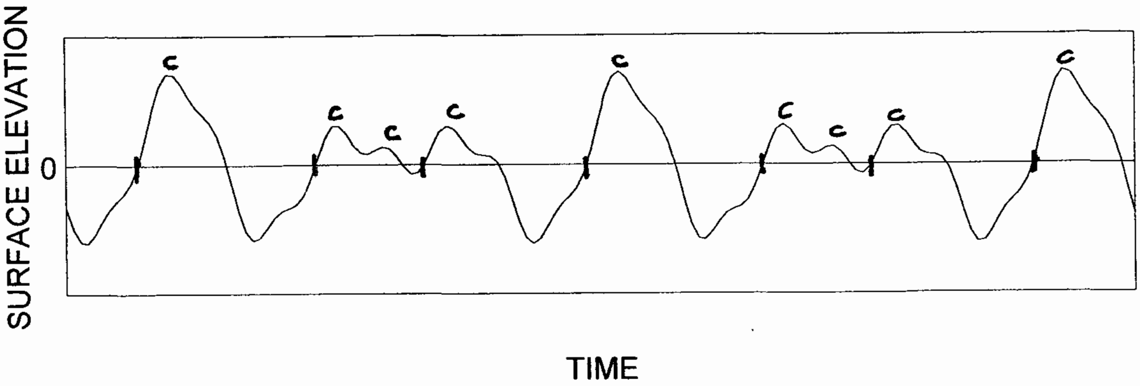
The above direct approach is cumbersome and computationally intensive. Many wave statistics have been theoretically determined in terms of the wave specturm. The associated formulae can be determined using numerical integration.

Results from Theory

The spectral moments, m_n, are defined in terms of the one-sided spectrum, $S_W(\omega)$, as:

$$m_n = \int_0^\infty S_W(\omega) d\omega$$

The following results apply when the surface elevation is a gaussian random process.



Number of Waves per Unit Time

The average number of times the wave elevation, ζ, crosses the mean sea level ($\zeta = 0$) per unit time while increasing is called f_o and given by:

$$f_o = \frac{1}{2\pi}\sqrt{\frac{m_2}{m_0}}$$

The average number of wave crests per unit time is called f_c and is given by:

$$f_c = \frac{1}{2\pi}\sqrt{\frac{m_4}{m_2}}$$

The bandwidth, ϵ, is given by:

$$\epsilon = \sqrt{1 - f_0^2 / f_c^2}$$

Definition of a gaussian random process For any number of variables, the joint probability density (pdf) of all the variables is a joint gaussian random variable at each time for a gaussian random process. This probability density function is given by:

$$p(x_1, x_2, ..., x_n) = \frac{1}{\sqrt{(2\pi)^n|\Delta|}} \exp\left\{-\frac{1}{2}[X]^T[\Delta^{-1}][X]\right\}$$

$[X]$ is the column vector of the variables. Δ is the n-by-n covariance matrix whose elements are given by:

$$\Delta_{ij} = E[x_i x_j]$$

For most wave statistics of interest, the doubly joint pdf between surface elevation, ζ and vertical surface velocity, $\dot{\zeta}$, and the triply joint pdf where the surface acceleration, $\ddot{\zeta}$, is included are all that are needed.

$$p(\zeta, \dot{\zeta}) = \frac{1}{2\pi\sqrt{m_0 m_2}} \exp\left[-\frac{m_2\zeta^2 + m_0\dot{\zeta}^2}{2m_0 m_2}\right]$$

$$p(\zeta, \dot{\zeta}, \ddot{\zeta}) = \frac{1}{(2\pi)^{3/2}\sqrt{m_2(m_0 m_4 - m_2^2)}} \exp\left[-\frac{m_2 m_4\zeta^2 + (m_0 m_4 - m_2^2)\dot{\zeta}^2 + m_0 m_2\ddot{\zeta}^2 + 2m_2^2\zeta\ddot{\zeta}}{2m_2(m_0 m_4 - m_2^2)}\right]$$

The normalized Gaussian probability distribution function (pdf), $\Psi(x)$, is:

$$\Psi(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{-z^2/2} dz$$

Call the crest height ξ.

The normalized crest height, η, is defined by: $\eta = \dfrac{\xi}{\sqrt{m_0}}$

The probability distribution function for η is:

$$P(\eta) = \Psi\left(\frac{\eta}{\epsilon}\right) - \sqrt{1-\epsilon^2}\, e^{-\eta^2/2}\, \Psi\left(\frac{\sqrt{1-\epsilon^2}}{\epsilon}\eta\right)$$

and the pdf for η is:

$$p(\eta) = \frac{\epsilon}{2\pi} \exp\left[-\frac{\eta^2}{2\epsilon^2}\right] + \sqrt{1-\epsilon^2}\, \eta\, e^{-\eta^2/2}\, \Psi\left(\frac{\sqrt{1-\epsilon^2}}{\epsilon}\eta\right)$$

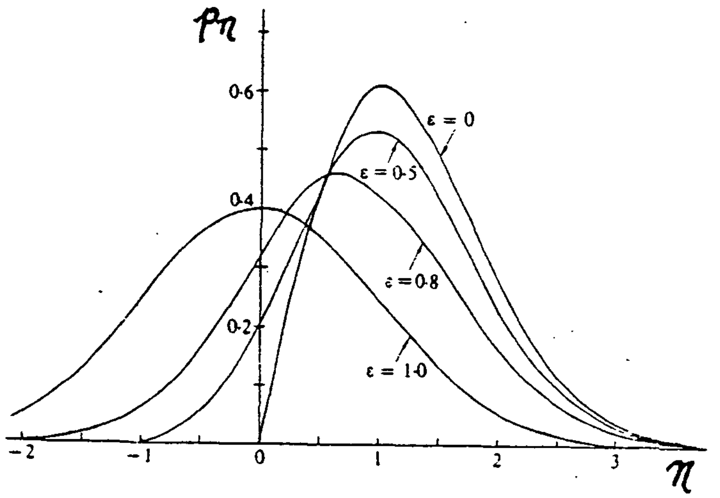


Figure 7.5 Probability density function of η for various values of the band width ϵ.

Typically, $\epsilon \approx 0.6$.

For engineering purposes we are interested in large seas ($\eta >> 1$). This corresponds to the tail of the pdf for η. In this region:

$$p(\eta) = \sqrt{1-\epsilon^2}\, \eta\, e^{-\eta^2/2} \qquad P(\eta) = 1 - \sqrt{1-\epsilon^2}\, e^{-\eta^2/2}$$

Average Amplitude of the 1/n'th Highest waves

Call the smallest normalized wave amplitude in the 1/n'th highest Waves $\eta_{1/n}$.

$$\frac{1}{n} = 1 - P(\eta_{1/n})$$

Example: n = 10.
1 - (probability that a wave amplitude is less than the smallest of the 10% largest waves) is 1/10.
This is because the probability that a (random) wave is smaller than 10% is 90%.

For $n >> 1$, use the approximate P.

$$\frac{1}{n} = \sqrt{1-\epsilon^2}\exp\left[-\frac{1}{2}\eta_{1/n}^2\right]$$

$$\eta_{1/n} = \sqrt{2\ln(n\sqrt{1-\epsilon^2})}$$

Amongst the 1/n'th highest waves, the conditional pdf is:

$$p_{\eta>\eta_{1/n}}(\eta) = np(\eta) = n\sqrt{1-\epsilon^2}\,\eta\,\exp(-\eta^2/2), \quad \eta_{1/n} < \eta < \infty$$

The expectation of these amplitudes is the average of the 1/n'th highest waves.

$$\overline{\eta_{1/n}} = n\sqrt{1-\epsilon^2}\int_{\eta_{1/n}}^{\infty} \eta^2 e^{-\eta^2/2} d\eta$$

Let $n' = \sqrt{1-\epsilon^2}\, n$. Then, n' is the number of zero up-crossings in a record with n crests. The result of the integration is:

$$\overline{\eta_{1/n}} = n'\left\{\frac{\sqrt{2\ln n'}}{n'} + \sqrt{2\pi}\left[1 - \Psi(\sqrt{2\ln n'})\right]\right\}$$

Extreme Waves

Consider n non-dimensional random wave Amplitudes. Each has same pdf.

What are the probabilities of the largest waves in the set?

Aproach
Order the waves from smallest to largest.
ϕ_1 is the smallest and ϕ_n is the largest wave amplitude. Now, each of the ϕ's has a different pdf.
We want to find the pdf for ϕ_n.

Probability that ϕ_n is less than a particular value ϕ_{n_o} is equal to the probability that all the waves are smaller than ϕ_{n_o}.

$$P_{\phi_n}(\phi_{n_o}) = [P_\eta(\phi_{n_o}]^n$$

The amplitude that has a probability, α, of being exceeded by ϕ_n is called ${}_\alpha\phi_n$.

$$P_{\phi_n}({}_\alpha\phi_n) = [P_\eta({}_\alpha\phi_n)]^n = 1 - \alpha$$

Meaning of the Nomenclature
Suppose $\alpha = 0.01$. Then the amplitude whose probability of being exceeded by ϕ_n is 0.01 is named ${}_{0.01}\phi_n$.
The probability that ϕ_n is less than ${}_{0.01}\phi_n$ is 0.99.

$$P_\eta({}_\alpha\phi_n) = (1-\alpha)^{1/n}$$

$$\Psi\left(\frac{{}_\alpha\phi_n}{\epsilon}\right) - \sqrt{1-\epsilon^2}\exp\left[-\frac{1}{2}\,{}_\alpha\phi_n^2\right]\Psi\left(\frac{\sqrt{1-\epsilon^2}}{\epsilon}\,{}_\alpha\phi_n\right) = (1-\alpha)^{1/n}$$

Since we are interested large waves, we can use the expressions for the tails of the probability functions:

$$P(\eta) = 1 - \sqrt{1-\epsilon^2}\, e^{-n^2/2}$$

$$\text{Then,} \quad 1 - \sqrt{1-\epsilon^2}\, \exp\left[-\frac{1}{2}\, {}_{\alpha}\phi_n^2\right] = (1-\alpha)^{1/n}$$

Solve for ${}_{\alpha}\phi_n$:

$${}_{\alpha}\phi_n = \sqrt{2\ln\left(\frac{\sqrt{1-\epsilon^2}}{1-(1-\alpha)^{1/n}}\right)}$$

Note: The value of n for a given period of time T can be obtained from:

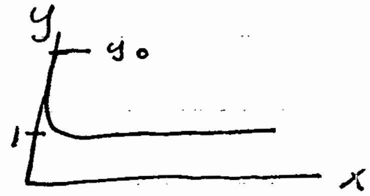
$$f_c = \frac{1}{2\pi}\sqrt{\frac{m_4}{m_2}}$$

$$n = f_c T = \frac{T}{2\pi}\sqrt{\frac{m_4}{m_2}}$$

STiff Equations

$$\frac{dy}{dx} = -100y + 100 \quad ; \text{ initial Condition, } y(0) = y_0$$

Exact Solution $y(x) = (y_0 - 1)e^{-100x} + 1$



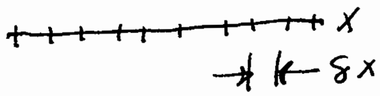
This is stable in sence that small change in initial condition causes small change in solution.

Example, if $y(0) = y_0 + \epsilon$,

$$y(x) = (y_0 + \epsilon - 1)e^{-100x} + 1$$

Change in Solution, δ, is ϵe^{-100x}

Solution by the forward Euler method



$$y_{n+1} = y_n + (-100y_n + 100)\delta x = (1 - 100\delta x)y_n + 100\,\delta x$$

This difference equation has an exact Solution

$$y_n = (y_0 - 1)(1 - 100\delta x)^n + 1$$

For example, if $y_0 = 2$, $y(x) = e^{-100x} + 1$

$$y_n = (1 - 100\delta x)^n + 1$$

Note; if $\delta x > 0.02$, the solution (numerical) diverges

$(1 - 100\delta x)^n$ is an approximation to e^{-100x}. It is a poor approximation unless δx is very small even though e^{-100x} hardly contributes to the solution for $x > .01$

This problem is often overcome by implicit methods. One is the backward Euler method

$$y_{n+1} = y_n + f(x_{n+1}, y_{n+1})\,\delta x$$

For our example $f(x_{n+1}, y_{n+1}) = (-100\,y_{n+1} + 100)$

$$y_{n+1} = y_n + (-100\,y_{n+1} + 100)\,\delta x$$

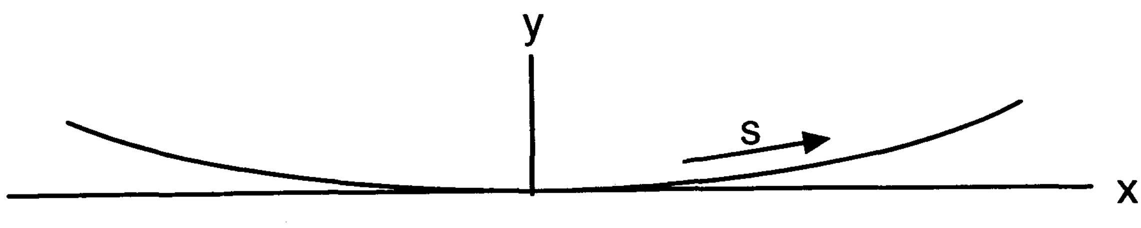
$$y_{n+1}(1 + 100\,\delta x) = y_n + 100\,\delta x \; ; \; y_{n+1} = \frac{y_n + 100\,\delta x}{1 + 100\,\delta x}$$

The exact solution to this is:

$$y_n = \frac{1}{(1 + 100\,\delta x)^n} + 1$$

This is not unstable for any δx

Dynamics of Horizontal Shallow Sag Cables in Water



Static Solution

H is the horizontal component of the Tension.
w is the weight in water/unit length.
T is the tension.
L is the static length.

$$y = \frac{H}{w} \cosh \frac{w}{H} x - \frac{H}{w} \qquad\qquad T = H \cosh \frac{w}{H} x$$

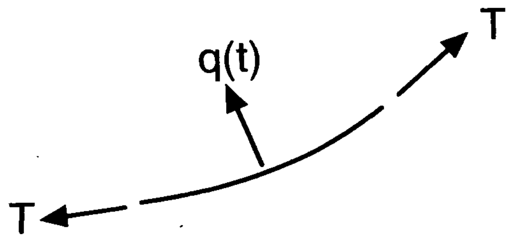
For $T \gg wL$

$$y = \frac{T_o}{w} \left[1 + \frac{w}{2T_o} x^2 + ... \right] - \frac{T_o}{w} \qquad\qquad T \equiv T_o \cong H$$

$$\frac{dy}{dx} = \frac{wx}{T_o}$$

$$\frac{d^2y}{dx^2} = \frac{w}{T_o} \equiv \alpha \qquad\qquad \text{static curvature } = \alpha$$

Dynamics



$$\text{vertical mechanical force/unit length } = \left(T_o + \widetilde{T}\right)\left(\alpha + \frac{\partial^2 q}{\partial t^2}\right)$$

q is the displacement normal to the cable towards the inside of the static curvature.

$$\text{dynamic vertical mechanical force/unit length } = \left(T_o + \widetilde{T}\right)\left(\alpha + \frac{\partial^2 q}{\partial s^2}\right) - T_o\alpha$$

$$\text{hydrodynamic vertical force/unit length } = -b\frac{dq}{dt}\left|\frac{dq}{dt}\right|$$

where: $b = \frac{1}{2}\rho C_d D$, ρ is the density of water, C_d is the drag coefficient and D is the diameter of the cable.

Equation of Motion

$$m\frac{\partial^2 q}{\partial t^2} = (T_o + \widetilde{T})\left(\alpha + \frac{\partial^2 q}{\partial s^2}\right) - b\frac{dq}{dt}\left|\frac{dq}{dt}\right| - T_o\alpha$$

Strain Compatibility

$$\text{Tension increase due to } q \ = \text{increased length } \times \frac{EA}{L}$$

E is the elastic modulus and A is the cross sectional Area of the cable.

$$\widetilde{T} = \frac{EA}{L}\left[p_o - \alpha\int_0^L q\,ds + \frac{1}{2}\int_0^L \left(\frac{\partial q}{\partial s}\right)^2 ds\right]$$

where: p_o is the sum of the tangential extensions of the ends of the cable.