Water wave equations 09-25-16

N. T. Gladd

Initialization: Be sure the files *NTGStylesheet2.nb* and *NTGUtilityFunctions.m* is are in the same directory as that from which this notebook was loaded. Then execute the cell immediately below by mousing left on the cell bar to the right of that cell and then typing “shift” + “enter”. Respond “Yes” in response to the query to evaluate initialization cells.

In[12]:=
```
SetDirectory[NotebookDirectory[]];
(* set directory where source files are located *)
SetOptions[EvaluationNotebook[], (* load the StyleSheet *)
 StyleDefinitions → Get["NTGStylesheet2.nb"]];
Get["NTGUtilityFunctions.m"]; (* Load utilities package *)
```

For several years I have lived near the ocean and watched the behavior of waves on beaches. Being a physicist by training (although not a specialist in hydrodynamics), I have enjoyed exploring the simpler mathematical models for wave and beach physics. The purpose of this notebook is to illustrate how *Mathematica* can be used to facilitate the derivation of the equation describing the propagation of water waves when the depth of the water is shallow. The shallow water wave equation is a starting point for many analyses of wave-beach physics.

There are many derivations of water wave equations in textbooks and the literature. I follow the basic procedure used by R. S. Johnson in *A Modern Introduction to the Mathematical Theory of Water Waves*.

This work is a revision and extension of earlier work contained in the notebook *Water Waves Equations - Dimensionless Forms 12-23-06.nb*. I have performed other calculations involving water waves that, as time permits, I will incorporate on this website.

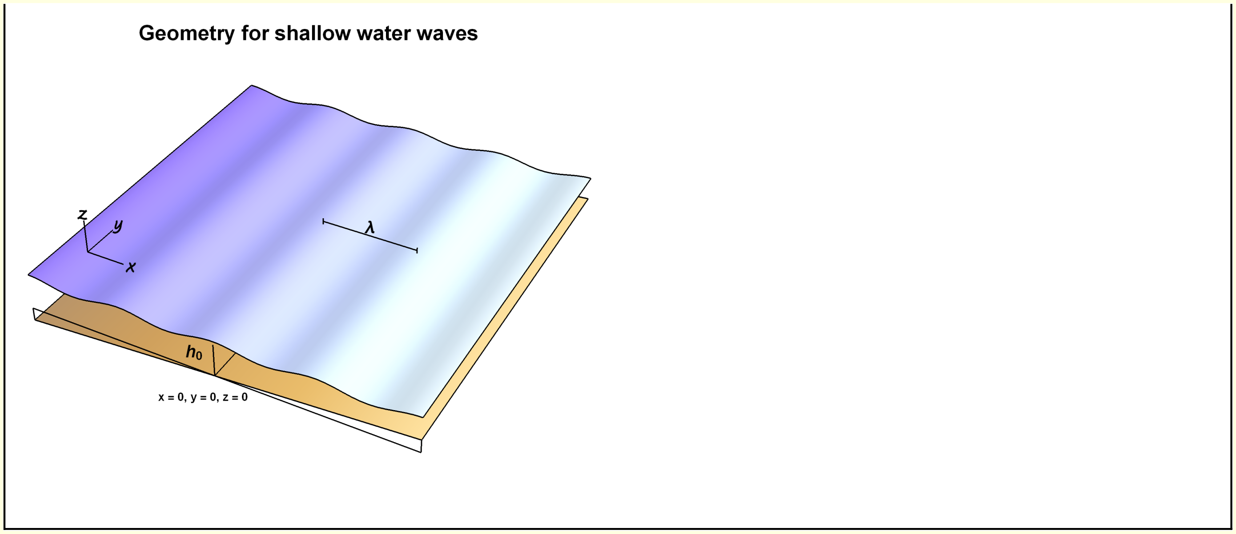
Overview
- Background
1 Dimensionless forms for the basic equations
2 Derivation of the shallow water wave equation

In a related notebook the results of section 1 are used to derive the dispersion relation for water waves.

Background

I derive the partial differential equation for waves in shallow water from first principles. The following figure illustrates the situation of interest, a body of water with nonuniform depth in the x direction.

Waves propagating in the x-redirection are considered but, since the geometry in uniform in the y-direction, it is trivial to also incorporate a component of wave propagation in the y-direction. There are two characteristic scale lengths - the nominal depth of the water, h(x = 0) = h_0, and the wave length of the waves. The shallow water approximation means that the wavelength is much longer than the depth of the water: formally, $\frac{\lambda}{h_0} \to 0$.



Basic equations

The first physical equation is the equation of continuity for a fluid element

$$\frac{\partial \rho}{\partial t} + \nabla.(\rho \boldsymbol{v}) = 0 \tag{1}$$

where ρ is the mass density of the water and **v** represents the velocity vector of a fluid element. To a good approximation, water is incompressible — meaning ρ is a constant. In that case,

$$\nabla.\boldsymbol{v} = 0 \tag{2}$$

A second equation is the force law for a fluid element

$$\rho \frac{d\boldsymbol{v}}{\mathrm{dt}} = \rho \frac{\partial \boldsymbol{v}}{\partial t} + \rho(\boldsymbol{v} \cdot \nabla)\boldsymbol{v} = -\nabla p + \boldsymbol{F} \tag{3}$$

For the problem I consider the only external force is the downward force of gravity, F = ρ (- $\hat{\boldsymbol{z}}$). I deviate from Johnson here who also considers surface tension and can thus treat both gravity waves and capillary waves.

At the bottom of the water it is assumed that the water particles are confined to a specified surface

$$z = b(x) \tag{4}$$

and thus there is a kinematic condition on the vertical velocity

$$\frac{\mathrm{dz}}{\mathrm{dt}} = v_z(z = b(x)) = \frac{\partial b(x)}{\partial t} + (\boldsymbol{v} \cdot \nabla)\, b(x) = (\boldsymbol{v} \cdot \nabla)\, b(x) \tag{5}$$

At the top surface of the water, there are two conditions, the kinematic condition that the fluid particles be confined within the top surface.

$$z = h(x, t) \tag{6}$$

or

$$\frac{dz}{dt} = v_z(z = h(x, t)) = \frac{\partial h(x, t)}{\partial t} + (\boldsymbol{v} \cdot \nabla)\, h(x, t) \tag{7}$$

and a dynamical condition

$$p(z, t) = p_{\text{atmosphere}} + \rho\, g\,(h(x, t) - z) \tag{8}$$

where the first term on the right hand side represents the atmospheric pressure and the second term is the hydroscopic pressure.

Objective of derivation

The primary objective of this derivation is to obtain a partial differential equation (the wave equation) for the perturbed height of the surface, $h_1(x, z, t)$, where h(z, t) = $h_0(z) + h_1$(z) and $|\, h_1(z)| << h_0(z)$. In the limiting case of shallow water this equation takes the form

$$\frac{\partial^2 h_1(x, t)}{\partial t^2} = \frac{\partial}{\partial x}\left((1 - b(x))\,\frac{\partial h_1(x, t)}{\partial x}\right) \tag{9}$$

Besides having relevance to the physics of water waves, this derivation illustrates various techniques by which partial differential equations characteristics of physics can be symbolically manipulated into a desired form. For this simple example, a pen and paper derivation is much quicker. However, it is not hard to generalize the problem to the extent that the number of required manual operations becomes onerous.

The first step is to transform the basic equations into dimensionless forms that make it easier to reason about the physics of the problem. In physics, the analytical process often consists of casting the relevant equations into dimensionless forms that involve key dependent and independent variables, together with a few dimensionless parameters that characterize the system being examined.

With an appropriate set of dimensional equations, the imposition of the shallow water condition is easily accomplished. The next step consists of perturbing the equations about an equilibrium state. This provides a different set of pdes for the perturbed surface height, the velocity components and the pressure.

Finally, dependent variables are eliminated until the desired equation for the perturbed surface height is obtained.

1 Dimensionless forms for the basic equations

When analyzing physical problems, it is useful to introduce dimensionless variables. In the vertical direction z, the natural scale length for distance is the depth of the water h_0 at the origin of the coordinate system. In the transverse direction there is no natural scale but, since the focus of interest will be on waves, the wavelength λ is a reasonable scaling parameter. The parameter τ is chosen as a time scale that will be specified later.

I will need an implementation of the convective derivative $\frac{d}{dt} = \frac{\partial}{\partial t} + v\cdot\nabla$. This is accomplished with the following functions

In[14]:=
```
Clear[DDt, vDel];
vDel[{vx_, vy_, vz_}, f_] :=
  vx D[f, x] + vy D[f, y] + vz D[f, z];
vDel[{vx_, vy_, vz_}, {fx_, fy_, fz_}] :=
  {vx D[fx, x], vy D[fy, y], vz D[fz, z]};
DDt[{vx_, vy_, vz_}, {fx_, fy_, fz_}] :=
  D[{fx, fy, fz}, t] + vDel[{vx, vy, vz}, {fx, fy, fz}];
DDt[{vx_, vy_, vz_}, f_] := D[f, t] + vDel[{vx, vy, vz}, f];
```

1-1 Dimensionless form for the continuity equation

In[19]:=
```
w11["start"] =
 Div[{v[𝓍][x, y, z, t], v[𝓎][x, y, z, t], v[z][x, y, z, t]}, {x, y , z} ] == 0
```

Out[19]=
$$v[z]^{(0,0,1,0)}[x, y, z, t] + v[\mathcal{y}]^{(0,1,0,0)}[x, y, z, t] + v[\mathcal{x}]^{(1,0,0,0)}[x, y, z, t] == 0$$

***Mathematica* note:** Although it is tempting to use subscripted variables for the components of the velocity vector, I shall resist. One has to go through special contortions to avoid distracting technical difficulties when performing *Mathematica* manipulations involving subscripted variables. For years, I did use subscripts for many of my calculations. After all, notation is very important in physics — consider Einstein, Dirac and Feynman. However, I relented when I found that even the wizards who answer questions at *http://mathematica.stackexchange.com* recommend against subscripts. In the notebook *Water Waves Equations - Dimensionless Forms 12-23-06.nb,* in which I first performed this derivation, I did use subscripted variables. Hopefully, developing the ability to *readily* use standard physical and mathematical notations will someday rise to to top of Wolfram’s technology stack. In fairness, the standard notations often used in the physics and mathematics literature are ambiguous for the purpose of performing computational symbolic manipulations.

I choose the notation v[𝓍][x, y, z, t] to represent v_x(x, y, z, t). The brackets [] attach so-called “DownValues” to the symbol v. I use “script x” — 𝓍 (escape scx escape) — to represent the direction of the vector and x to represent the independent variable x. This avoids problems with differentiation with respect to x.

I first transform the independent variables by applying rewrite rules

In[20]:=
```
w11[2] = w11["start"] /. v[a_] → ((v[a][#1 / λ, #2 / λ, #3 / h0, #4 / τ]) &) /.
   {x / λ → X, y / λ → Y, z / h0 → Z, t / τ → T}
```

Out[20]=
$$\frac{v[z]^{(0,0,1,0)}[X, Y, Z, T]}{h0} + \frac{v[y]^{(0,1,0,0)}[X, Y, Z, T]}{\lambda} + \frac{v[x]^{(1,0,0,0)}[X, Y, Z, T]}{\lambda} == 0$$

Next I introduce dimensionless forms for the dependent variables. For the velocities I use the scale factors - h/τ for the vertical velocity and λ/τ for the transverse velocity components. These are consistent with the scaling of the independent variables.

In[21]:=
```
w11[3] = w11[2] /. Derivative[a_, b_, c_, d_][v[lab_]][X, Y, Z, T] :→
    If[MemberQ[{x, y}, lab], λ / τ, h0 / τ] Derivative[a, b, c, d][V[lab]][X, Y, Z, T];
(*clean up*)
w11["final"] = (# τ) & /@ w11[3] // Expand
```

Out[22]=
$$V[z]^{(0,0,1,0)}[X, Y, Z, T] + V[y]^{(0,1,0,0)}[X, Y, Z, T] + V[x]^{(1,0,0,0)}[X, Y, Z, T] == 0$$

1-2 Equations of motion (Euler)

The equations of motion (which for temporary convenience, I write as expressions rather than equations) are

In[23]:=
```
w12["start"] = ρ DDt[{v[x][x, y, z, t], v[y][x, y, z, t], v[z][x, y, z, t]},
    {v[x][x, y, z, t], v[y][x, y, z, t], v[z][x, y, z, t]} ] +
  Grad[p[x, y, z, t], {x, y, z}] - {F[x], F[y], F[z]}
```

Out[23]=
$$\{-F[x] + p^{(1,0,0,0)}[x, y, z, t] + \rho\,(v[x]^{(0,0,0,1)}[x, y, z, t] + v[x][x, y, z, t]\, v[x]^{(1,0,0,0)}[x, y, z, t]),$$
$$-F[y] + p^{(0,1,0,0)}[x, y, z, t] + \rho\,(v[y]^{(0,0,0,1)}[x, y, z, t] + v[y][x, y, z, t]\, v[y]^{(0,1,0,0)}[x, y, z, t]),$$
$$-F[z] + p^{(0,0,1,0)}[x, y, z, t] + \rho\,(v[z]^{(0,0,0,1)}[x, y, z, t] + v[z][x, y, z, t]\, v[z]^{(0,0,1,0)}[x, y, z, t])\}$$

The only external force for this problem is the downward force of gravity

In[24]:=
```
w12[2] = w12["start"] /. {F[x] → 0, F[y] → 0, F[z] → -ρ g}
```

Out[24]=
$$\{p^{(1,0,0,0)}[x, y, z, t] + \rho\,(v[x]^{(0,0,0,1)}[x, y, z, t] + v[x][x, y, z, t]\, v[x]^{(1,0,0,0)}[x, y, z, t]),$$
$$p^{(0,1,0,0)}[x, y, z, t] + \rho\,(v[y]^{(0,0,0,1)}[x, y, z, t] + v[y][x, y, z, t]\, v[y]^{(0,1,0,0)}[x, y, z, t]),\ g\,\rho + p^{(0,0,1,0)}[x, y, z, t] + \rho\,(v[z]^{(0,0,0,1)}[x, y, z, t] + v[z][x, y, z, t]\, v[z]^{(0,0,1,0)}[x, y, z, t])\}$$

Introduce dimensionless independent variables

In[25]:=
```
w12[3] = w12[2] /. v[a_] → ((v[a][#1 / λ, #2 / λ, #3 / h0, #4 / τ]) &) /.
    p → ((p[#1 / λ, #2 / λ, #3 / h0, #4 / τ]) &) /.
   {x / λ → X, y / λ → Y, z / h0 → Z, t / τ → T} // Expand
```

Out[25]=

$$\left\{\frac{\rho\, v[x]^{(0,0,0,1)}[X,Y,Z,T]}{\tau}+\frac{p^{(1,0,0,0)}[X,Y,Z,T]}{\lambda}+\frac{1}{\lambda}\rho\, v[x][X,Y,Z,T]\, v[x]^{(1,0,0,0)}[X,Y,Z,T],\ \frac{\rho\, v[y]^{(0,0,0,1)}[X,Y,Z,T]}{\tau}+\frac{p^{(0,1,0,0)}[X,Y,Z,T]}{\lambda}+\frac{1}{\lambda}\rho\, v[y][X,Y,Z,T]\, v[y]^{(0,1,0,0)}[X,Y,Z,T],\ g\rho+\frac{\rho\, v[z]^{(0,0,0,1)}[X,Y,Z,T]}{\tau}+\frac{p^{(0,0,1,0)}[X,Y,Z,T]}{h0}+\frac{1}{h0}\rho\, v[z][X,Y,Z,T]\, v[z]^{(0,0,1,0)}[X,Y,Z,T]\right\}$$

Introduce dimensionless dependent variables

In[26]:=
```
w12[4] = w12[3] /.
    Derivative[a_, b_, c_, d_][v[lab_]][X, Y, Z, T] :> If[MemberQ[{x, y}, lab],
       λ / τ, h0 / τ] Derivative[a, b, c, d][V[lab]][X, Y, Z, T] /.
   v[lab_][X, Y, Z, T] :> If[MemberQ[{x, y}, lab], λ / τ, h0 / τ] V[lab][X, Y, Z, T] /.
  Derivative[a_, b_, c_, d_][p][X, Y, Z, T] :> p0 Derivative[a, b, c, d][P][X, Y, Z, T]
```

Out[26]=

$$\left\{\frac{\lambda\rho\, V[x]^{(0,0,0,1)}[X,Y,Z,T]}{\tau^2}+\frac{p0\, P^{(1,0,0,0)}[X,Y,Z,T]}{\lambda}+\frac{1}{\tau^2}\lambda\rho\, V[x][X,Y,Z,T]\, V[x]^{(1,0,0,0)}[X,Y,Z,T],\ \frac{\lambda\rho\, V[y]^{(0,0,0,1)}[X,Y,Z,T]}{\tau^2}+\frac{p0\, P^{(0,1,0,0)}[X,Y,Z,T]}{\lambda}+\frac{1}{\tau^2}\lambda\rho\, V[y][X,Y,Z,T]\, V[y]^{(0,1,0,0)}[X,Y,Z,T],\ g\rho+\frac{h0\,\rho\, V[z]^{(0,0,0,1)}[X,Y,Z,T]}{\tau^2}+\frac{p0\, P^{(0,0,1,0)}[X,Y,Z,T]}{h0}+\frac{1}{\tau^2}h0\,\rho\, V[z][X,Y,Z,T]\, V[z]^{(0,0,1,0)}[X,Y,Z,T]\right\}$$

Simplify

In[27]:=
```
w12[5] = (# τ^2 / (ρ λ)) & /@ w12[4] // Expand
```

Out[27]=

$$\left\{V[x]^{(0,0,0,1)}[X,Y,Z,T]+\frac{p0\,\tau^2 P^{(1,0,0,0)}[X,Y,Z,T]}{\lambda^2\rho}+V[x][X,Y,Z,T]\, V[x]^{(1,0,0,0)}[X,Y,Z,T],\ V[y]^{(0,0,0,1)}[X,Y,Z,T]+\frac{p0\,\tau^2 P^{(0,1,0,0)}[X,Y,Z,T]}{\lambda^2\rho}+V[y][X,Y,Z,T]\, V[y]^{(0,1,0,0)}[X,Y,Z,T],\ \frac{g\tau^2}{\lambda}+\frac{h0\, V[z]^{(0,0,0,1)}[X,Y,Z,T]}{\lambda}+\frac{p0\,\tau^2 P^{(0,0,1,0)}[X,Y,Z,T]}{h0\,\lambda\,\rho}+\frac{1}{\lambda}h0\, V[z][X,Y,Z,T]\, V[z]^{(0,0,1,0)}[X,Y,Z,T]\right\}$$

Eliminate the time scale in terms of a velocity scale

In[28]:=
```
w12[6] = w12[5] /. τ → λ/v0
```

Out[28]=

$$\left\{V[x]^{(0,0,0,1)}[X,Y,Z,T]+\frac{p0\,P^{(1,0,0,0)}[X,Y,Z,T]}{v0^2\,\rho}+V[x][X,Y,Z,T]\,V[x]^{(1,0,0,0)}[X,Y,Z,T],\right.$$
$$V[y]^{(0,0,0,1)}[X,Y,Z,T]+\frac{p0\,P^{(0,1,0,0)}[X,Y,Z,T]}{v0^2\,\rho}+V[y][X,Y,Z,T]\,V[y]^{(0,1,0,0)}[X,Y,Z,T],$$
$$\frac{g\,\lambda}{v0^2}+\frac{h0\,V[z]^{(0,0,0,1)}[X,Y,Z,T]}{\lambda}+\frac{p0\,\lambda\,P^{(0,0,1,0)}[X,Y,Z,T]}{h0\,v0^2\,\rho}+$$
$$\left.\frac{1}{\lambda}h0\,V[z][X,Y,Z,T]\,V[z]^{(0,0,1,0)}[X,Y,Z,T]\right\}$$

The pressure scaling parameter p0 can be chosen freely. A natural choice that simplifies the equations is p0 = ρ v0^2

In[29]:=
```
w12[7] = w12[6] /. p0 → ρ v0^2
```

Out[29]=

$$\left\{V[x]^{(0,0,0,1)}[X,Y,Z,T]+P^{(1,0,0,0)}[X,Y,Z,T]+V[x][X,Y,Z,T]\,V[x]^{(1,0,0,0)}[X,Y,Z,T],\right.$$
$$V[y]^{(0,0,0,1)}[X,Y,Z,T]+P^{(0,1,0,0)}[X,Y,Z,T]+V[y][X,Y,Z,T]\,V[y]^{(0,1,0,0)}[X,Y,Z,T],$$
$$\frac{g\,\lambda}{v0^2}+\frac{h0\,V[z]^{(0,0,0,1)}[X,Y,Z,T]}{\lambda}+\frac{\lambda\,P^{(0,0,1,0)}[X,Y,Z,T]}{h0}+$$
$$\left.\frac{1}{\lambda}h0\,V[z][X,Y,Z,T]\,V[z]^{(0,0,1,0)}[X,Y,Z,T]\right\}$$

At this point I introduce a parameter that characterizes the relationship between the vertical and transverse velocity scales

In[30]:=
```
def[δ] = δ == h0/λ;
```

In[32]:=
```
w12[8] = w12[7] /. Solve[def[δ], h0][[1, 1]]
```

Out[32]=

$$\left\{V[x]^{(0,0,0,1)}[X,Y,Z,T]+P^{(1,0,0,0)}[X,Y,Z,T]+V[x][X,Y,Z,T]\,V[x]^{(1,0,0,0)}[X,Y,Z,T],\right.$$
$$V[y]^{(0,0,0,1)}[X,Y,Z,T]+P^{(0,1,0,0)}[X,Y,Z,T]+V[y][X,Y,Z,T]\,V[y]^{(0,1,0,0)}[X,Y,Z,T],$$
$$\frac{g\,\lambda}{v0^2}+\delta\,V[z]^{(0,0,0,1)}[X,Y,Z,T]+\frac{P^{(0,0,1,0)}[X,Y,Z,T]}{\delta}+$$
$$\left.\delta\,V[z][X,Y,Z,T]\,V[z]^{(0,0,1,0)}[X,Y,Z,T]\right\}$$

Consider the term $g\,\lambda/\mathrm{v0}^2$, which I can write as $g\,\mathrm{h0}/\mathrm{v0}^2\;\frac{\lambda}{\mathrm{h0}} = g\,\mathrm{h0}/\mathrm{v0}^2\;\frac{1}{\delta}$

Choosing the velocity scale to be v0 = $\sqrt{g\,\mathrm{h0}}$, results in a set of equations for dimensionless variables with the single parameter δ

In[67]:=
```
w12[9] = w12[8] /. v0 → √(g h0) /. Solve[def[δ], h0][[1, 1]]
```

Out[67]=

$$\left\{V[x]^{(0,0,0,1)}[X, Y, Z, T] + P^{(1,0,0,0)}[X, Y, Z, T] + V[x][X, Y, Z, T]\, V[x]^{(1,0,0,0)}[X, Y, Z, T],\; V[y]^{(0,0,0,1)}[X, Y, Z, T] + P^{(0,1,0,0)}[X, Y, Z, T] + V[y][X, Y, Z, T]\, V[y]^{(0,1,0,0)}[X, Y, Z, T],\; \frac{1}{\delta} + \delta\, V[z]^{(0,0,0,1)}[X, Y, Z, T] + \frac{P^{(0,0,1,0)}[X, Y, Z, T]}{\delta} + \delta\, V[z][X, Y, Z, T]\, V[z]^{(0,0,1,0)}[X, Y, Z, T]\right\}$$

For future convenience, I ensure that no factor of δ appears in a denominator

In[71]:=
```
w12[10] = ReplacePart[w12[9], 3 → Expand[δ w12[9][[3]]] ]
```

Out[71]=

$$\left\{V[x]^{(0,0,0,1)}[X, Y, Z, T] + P^{(1,0,0,0)}[X, Y, Z, T] + V[x][X, Y, Z, T]\, V[x]^{(1,0,0,0)}[X, Y, Z, T],\; V[y]^{(0,0,0,1)}[X, Y, Z, T] + P^{(0,1,0,0)}[X, Y, Z, T] + V[y][X, Y, Z, T]\, V[y]^{(0,1,0,0)}[X, Y, Z, T],\; 1 + \delta^2 V[z]^{(0,0,0,1)}[X, Y, Z, T] + P^{(0,0,1,0)}[X, Y, Z, T] + \delta^2 V[z][X, Y, Z, T]\, V[z]^{(0,0,1,0)}[X, Y, Z, T]\right\}$$

I now convert these expressions to equations

In[72]:=
```
w12["final"] = (# == 0) & /@ w12[10]
```

Out[72]=

$$\left\{V[x]^{(0,0,0,1)}[X, Y, Z, T] + P^{(1,0,0,0)}[X, Y, Z, T] + V[x][X, Y, Z, T]\, V[x]^{(1,0,0,0)}[X, Y, Z, T] == 0,\; V[y]^{(0,0,0,1)}[X, Y, Z, T] + P^{(0,1,0,0)}[X, Y, Z, T] + V[y][X, Y, Z, T]\, V[y]^{(0,1,0,0)}[X, Y, Z, T] == 0,\; 1 + \delta^2 V[z]^{(0,0,0,1)}[X, Y, Z, T] + P^{(0,0,1,0)}[X, Y, Z, T] + \delta^2 V[z][X, Y, Z, T]\, V[z]^{(0,0,1,0)}[X, Y, Z, T] == 0\right\}$$

These are the dimensionless equations for the bulk of the water. It remains to obtain the particular forms of the equations that hold at the surface and bottom.

1-3 Boundary condition at the bottom

The bottom of the water is defined by the boundary condition z = b(x).

In[38]:=
```
w13["start"] = z == b[x]
```

Out[38]=
```
z == b[x]
```

Applying the convective time derivative to this kinematic equation

In[39]:=
```
w13[2] =
 DDt[{v[x][x, y, z, t], v[y][x, y, z, t], v[z][x, y, z, t]}, #] & /@ w13["start"]
```

Out[39]=

$$v[z][x, y, z, t] == v[x][x, y, z, t]\, b'[x]$$

Make the variables dimensionless

In[40]:=
```
w13[3] = w13[2] /.
    v[lab_][x, y, z, t] :→ If[MemberQ[{x, y}, lab], v0 , δ v0] V[lab][X, Y, Z, T] /.
   b → (b[#1/λ] &) /.
  x/λ → X
```

Out[40]=

$$v0\,\delta\,V[z][X, Y, Z, T] == \frac{v0\,V[x][X, Y, Z, T]\,b'[X]}{\lambda}$$

In[43]:=
```
w13[4] = w13[3] /. Derivative[1][b][X] → h0 Derivative[1][B][X] /.
   Solve[def[δ], h0][[1, 1]];
(* clean up *)
w13[4] = (#/(δ v0)) & /@ w13[4]
```

Out[44]=

$$V[z][X, Y, Z, T] == V[x][X, Y, Z, T]\,B'[X]$$

This equation is true at the bottom (Z = 0)

In[45]:=
```
w13["final"] = w13[4] /. Z → B[X]
```

Out[45]=

$$V[z][X, Y, B[X], T] == V[x][X, Y, B[X], T]\,B'[X]$$

1-4 Kinematic boundary condition at the surface

The kinematic constraint at the surface is z = h(x, y, t)

In[46]:=
```
w14["start"] = z == h[x, y, t]
```

Out[46]=

$$z == h[x, y, t]$$

In[47]:=
```
w14[2] =
 DDt[{v[x][x, y, z, t], v[y][x, y, z, t], v[z][x, y, z, t]}, #] & /@ w14["start"]
```

Out[47]=

$$v[z][x, y, z, t] == h^{(0,0,1)}[x, y, t] + v[y][x, y, z, t]\,h^{(0,1,0)}[x, y, t] + v[x][x, y, z, t]\,h^{(1,0,0)}[x, y, t]$$

Make the variables dimensionless

In[48]:=
```
w14[3] = w14[2] /. h → ((h[#1/λ, #2/λ, #3/τ]) &) /. {x/λ → X, y/λ → Y, t/τ → T}
```

Out[48]=

$$v[z][x, y, z, t] == \frac{h^{(0,0,1)}[X, Y, T]}{\tau} + \frac{1}{\lambda}v[y][x, y, z, t]\,h^{(0,1,0)}[X, Y, T] + \frac{1}{\lambda}v[x][x, y, z, t]\,h^{(1,0,0)}[X, Y, T]$$

In[49]:=
```
w14[4] = w14[3] /.
  v[lab_][x_, y_, z_, t_] :> If[MemberQ[{x, y}, lab], v0, δ v0] V[lab][X, Y, Z, T]
```

Out[49]=

$$v0\,\delta\,V[z][X, Y, Z, T] == \frac{h^{(0,0,1)}[X, Y, T]}{\tau} + \frac{1}{\lambda} v0\,V[y][X, Y, Z, T]\,h^{(0,1,0)}[X, Y, T] + \frac{1}{\lambda} v0\,V[x][X, Y, Z, T]\,h^{(1,0,0)}[X, Y, T]$$

In[50]:=
```
w14[5] = w14[4] /.
   Derivative[a_, b_, c_][h][X, Y, T] → h0 Derivative[a, b, c][H][X, Y, T];
(* rearrangement *)
w14[5] = (#/(v0 δ)) & /@ w14[5] // Expand
```

Out[51]=

$$V[z][X, Y, Z, T] == \frac{h0\,H^{(0,0,1)}[X, Y, T]}{v0\,\delta\,\tau} + \frac{1}{\delta\,\lambda} h0\,V[y][X, Y, Z, T]\,H^{(0,1,0)}[X, Y, T] + \frac{1}{\delta\,\lambda} h0\,V[x][X, Y, Z, T]\,H^{(1,0,0)}[X, Y, T]$$

In[52]:=
```
w14[6] = w14[5] /. v0 → λ/τ /. Solve[def[δ], h0][[1, 1]]
```

Out[52]=

$$V[z][X, Y, Z, T] == H^{(0,0,1)}[X, Y, T] + V[y][X, Y, Z, T]\,H^{(0,1,0)}[X, Y, T] + V[x][X, Y, Z, T]\,H^{(1,0,0)}[X, Y, T]$$

This equation is true at the surface z = h0 or Z = 1

In[53]:=
```
w14["final"] = w14[6] /. Z → 1
```

Out[53]=

$$V[z][X, Y, 1, T] == H^{(0,0,1)}[X, Y, T] + V[y][X, Y, 1, T]\,H^{(0,1,0)}[X, Y, T] + V[x][X, Y, 1, T]\,H^{(1,0,0)}[X, Y, T]$$

1-5 Dynamic boundary condition at the surface

Vertical pressure balance requires

In[54]:=
```
w15["start"] = p[x, y, z, t] == Pa + ρ g (h[x, y, t] - z)
```

Out[54]=

$$p[x, y, z, t] == g\,\rho\,(-z + h[x, y, t]) + P_a$$

In[55]:=
```
w15[2] = w15["start"] /. p[x, y, z, t] → p0 P[X, Y, Z, T] /. z → h0 Z /.
     h[x, y, t] → h0 H[X, Y, T] // Expand;
(* rearrangement *)
w15[2] = (#/p0) & /@ w15[2] // Expand
```

Out[56]=

$$P[X, Y, Z, T] == -\frac{g\,h0\,Z\,\rho}{p0} + \frac{g\,h0\,\rho\,H[X, Y, T]}{p0} + \frac{P_a}{p0}$$

Set the fluid pressure equal to the atmospheric pressure

In[57]:=
```
w15[3] = w15[2] /. Pa → p0 /. p0 → g h0 ρ
```
Out[57]=
```
P[X, Y, Z, T] == 1 - Z + H[X, Y, T]
```

Then, at the surface Z = 1

In[58]:=
```
w15["final"] = w15[3] /. Z → 1
```
Out[58]=
```
P[X, Y, 1, T] == H[X, Y, T]
```

1-6 Collected equations and imposition of the shallow water approximation

In[73]:=
```
w16["final"] =
 {w11["final"], w12["final"], w13["final"], w14["final"], w15["final"]} // Flatten
```
Out[73]=
```
{V[z]^(0,0,1,0)[X, Y, Z, T] + V[y]^(0,1,0,0)[X, Y, Z, T] + V[x]^(1,0,0,0)[X, Y, Z, T] == 0,
 V[x]^(0,0,0,1)[X, Y, Z, T] + P^(1,0,0,0)[X, Y, Z, T] + V[x][X, Y, Z, T] V[x]^(1,0,0,0)[X, Y, Z, T] ==
  0, V[y]^(0,0,0,1)[X, Y, Z, T] + P^(0,1,0,0)[X, Y, Z, T] +
   V[y][X, Y, Z, T] V[y]^(0,1,0,0)[X, Y, Z, T] == 0, 1 + δ^2 V[z]^(0,0,0,1)[X, Y, Z, T] +
   P^(0,0,1,0)[X, Y, Z, T] + δ^2 V[z][X, Y, Z, T] V[z]^(0,0,1,0)[X, Y, Z, T] == 0,
 V[z][X, Y, B[X], T] == V[x][X, Y, B[X], T] B′[X], V[z][X, Y, 1, T] ==
  H^(0,0,1)[X, Y, T] + V[y][X, Y, 1, T] H^(0,1,0)[X, Y, T] + V[x][X, Y, 1, T] H^(1,0,0)[X, Y, T],
 P[X, Y, 1, T] == H[X, Y, T]}
```

A more traditional display can be achieved with a special function (see section *Display of partial derivatives* below)

In[74]:=
```
(w16["final"] // ColumnForm) // PhysicsForm
```
Out[74]//TraditionalForm=

$$\frac{\partial V(x)(X,Y,Z,T)}{\partial X} + \frac{\partial V(y)(X,Y,Z,T)}{\partial Y} + \frac{\partial V(z)(X,Y,Z,T)}{\partial Z} = 0$$
$$\frac{\partial P(X,Y,Z,T)}{\partial X} + \frac{\partial V(x)(X,Y,Z,T)}{\partial T} + \frac{\partial V(x)(X,Y,Z,T)}{\partial X}\, V(x)(X,\ Y,\ Z,\ T) = 0$$
$$\frac{\partial P(X,Y,Z,T)}{\partial Y} + \frac{\partial V(y)(X,Y,Z,T)}{\partial T} + \frac{\partial V(y)(X,Y,Z,T)}{\partial Y}\, V(y)(X,\ Y,\ Z,\ T) = 0$$
$$\frac{\partial V(z)(X,Y,Z,T)}{\partial T}\,\delta^2 + \frac{\partial V(z)(X,Y,Z,T)}{\partial Z}\, V(z)(X,\ Y,\ Z,\ T)\,\delta^2 + \frac{\partial P(X,Y,Z,T)}{\partial Z} + 1 = 0$$
$$V(z)(X,\ Y,\ B(X),\ T) = \frac{\partial B(X)}{\partial X}\, V(x)(X,\ Y,\ B(X),\ T)$$
$$V(z)(X,\ Y,\ 1,\ T) = \frac{\partial H(X,Y,T)}{\partial T} + \frac{\partial H(X,Y,T)}{\partial X}\, V(x)(X,\ Y,\ 1,\ T) + \frac{\partial H(X,Y,T)}{\partial Y}\, V(y)(X,\ Y,\ 1,\ T)$$
$$P(X,\ Y,\ 1,\ T) = H(X,\ Y,\ T)$$

With this set of equations, I derived the dispersion relation for water waves in another notebook — *Water wave dispersion - 09-25-16.*

2 Derivation of the shallow water wave equation

2.1 Shallow water

Imposing the shallow water approximation consists of setting $\delta = \frac{h0}{\lambda} \to 0$. This corresponds to the wavelengths being much longer than the depth of the water.

In[75]:=
```
w21[1] = w16["final"] /. δ → 0
```

Out[75]=
```
{V[z]^(0,0,1,0)[X, Y, Z, T] + V[y]^(0,1,0,0)[X, Y, Z, T] + V[x]^(1,0,0,0)[X, Y, Z, T] == 0,
 V[x]^(0,0,0,1)[X, Y, Z, T] + P^(1,0,0,0)[X, Y, Z, T] + V[x][X, Y, Z, T] V[x]^(1,0,0,0)[X, Y, Z, T] ==
  0, V[y]^(0,0,0,1)[X, Y, Z, T] + P^(0,1,0,0)[X, Y, Z, T] +
   V[y][X, Y, Z, T] V[y]^(0,1,0,0)[X, Y, Z, T] == 0,
 1 + P^(0,0,1,0)[X, Y, Z, T] == 0, V[z][X, Y, B[X], T] == V[x][X, Y, B[X], T] B′[X],
 V[z][X, Y, 1, T] == H^(0,0,1)[X, Y, T] + V[y][X, Y, 1, T] H^(0,1,0)[X, Y, T] +
   V[x][X, Y, 1, T] H^(1,0,0)[X, Y, T], P[X, Y, 1, T] == H[X, Y, T]}
```

2.2 Linearizing the equations

Consider perturbations about an equilibrium state. Notice that I take the opportunity to write the replacement rules so as to eliminate the ignorable Y dependence.

$$H = \frac{h}{h_0} = 1 + \epsilon\,\mathrm{H1} \qquad \textbf{(surface is perturbed)}$$

$$P = \mathrm{P0} + \epsilon\,\mathrm{P1} \qquad \textbf{(pressure is perturbed)}$$

$$V = \mathrm{V0} + \epsilon\,\mathrm{V1} = 0 + \epsilon\,\mathrm{V1} \qquad \textbf{(no equilibrium flows)}$$

In[76]:=
```
w22[1] = w21[1] /. {V[a_] → ((ϵ V1[a][#1, #3, #4]) &),
    P → ((P0[#3] + ϵ P1[#1, #3, #4]) &),
    H → ((1 + ϵ H1[#1, #3]) &)}
```

Out[76]=
```
{ϵ V1[z]^(0,1,0)[X, Z, T] + ϵ V1[x]^(1,0,0)[X, Z, T] == 0,
 ϵ V1[x]^(0,0,1)[X, Z, T] + ϵ P1^(1,0,0)[X, Z, T] + ϵ^2 V1[x][X, Z, T] V1[x]^(1,0,0)[X, Z, T] == 0,
 ϵ V1[y]^(0,0,1)[X, Z, T] == 0, 1 + P0′[Z] + ϵ P1^(0,1,0)[X, Z, T] == 0,
 ϵ V1[z][X, B[X], T] == ϵ V1[x][X, B[X], T] B′[X],
 ϵ V1[z][X, 1, T] == ϵ H1^(0,1)[X, T] + ϵ^2 V1[x][X, 1, T] H1^(1,0)[X, T],
 P0[1] + ϵ P1[X, 1, T] == 1 + ϵ H1[X, T]}
```

To order ϵ^0

In[77]:=
```
w22[2] = Expand@w22[1] /. ϵ → 0
```
Out[77]=
```
{True, True, True, 1 + P0′[Z] == 0, True, True, P0[1] == 1}
```

The equilibrium conditions are that the pressure at the surface is equal to the atmospheric pressures and that the vertical force on a fluid element is balanced.

To order ϵ^1 with equilibrium conditions applied

In[78]:=
```
w22[3] = Expand@w22[1] /. ϵ^n_ /; n≥2 → 0 /. P0[1] → 1 /. P0'[Z] → -1
```
Out[78]=
```
{ϵ V1[z]^(0,1,0)[X, Z, T] + ϵ V1[x]^(1,0,0)[X, Z, T] == 0,
 ϵ V1[x]^(0,0,1)[X, Z, T] + ϵ P1^(1,0,0)[X, Z, T] == 0, ϵ V1[y]^(0,0,1)[X, Z, T] == 0,
 ϵ P1^(0,1,0)[X, Z, T] == 0, ϵ V1[z][X, B[X], T] == ϵ V1[x][X, B[X], T] B′[X],
 ϵ V1[z][X, 1, T] == ϵ H1^(0,1)[X, T], 1 + ϵ P1[X, 1, T] == 1 + ϵ H1[X, T]}
```

The equation involving V1[*y*] can be removed.

In[79]:=
```
w22[4] = Drop[w22[3], {3}]
```
Out[79]=
```
{ϵ V1[z]^(0,1,0)[X, Z, T] + ϵ V1[x]^(1,0,0)[X, Z, T] == 0,
 ϵ V1[x]^(0,0,1)[X, Z, T] + ϵ P1^(1,0,0)[X, Z, T] == 0, ϵ P1^(0,1,0)[X, Z, T] == 0,
 ϵ V1[z][X, B[X], T] == ϵ V1[x][X, B[X], T] B′[X],
 ϵ V1[z][X, 1, T] == ϵ H1^(0,1)[X, T], 1 + ϵ P1[X, 1, T] == 1 + ϵ H1[X, T]}
```

In[80]:=
```
w22[5] = w22[4] /. ϵ → 1 // Simplify
```
Out[80]=
```
{V1[z]^(0,1,0)[X, Z, T] + V1[x]^(1,0,0)[X, Z, T] == 0, V1[x]^(0,0,1)[X, Z, T] + P1^(1,0,0)[X, Z, T] == 0,
 P1^(0,1,0)[X, Z, T] == 0, V1[z][X, B[X], T] == V1[x][X, B[X], T] B′[X],
 V1[z][X, 1, T] == H1^(0,1)[X, T], H1[X, T] == P1[X, 1, T]}
```

2-3 Solving the linearized equations

The objective will be to manipulate these partial differential equations so as to eliminate the perturbed pressure and velocity components and finally obtain a single partial differential equation for the surface perturbation H_1.

I will be manipulating the set of fluid equations and boundary equations. For convenience, I develop a function for displaying the equations with a corresponding reference number.

In[81]:=
```
DisplayEqns[w22[5], {V1 → V₁, P1 → P₁, H1 → H₁}]
```

Out[81]//TraditionalForm=

$$\begin{pmatrix} 1 & \frac{\partial V_1(x)(X,Z,T)}{\partial X} + \frac{\partial V_1(z)(X,Z,T)}{\partial Z} = 0 \\ 2 & \frac{\partial P_1(X,Z,T)}{\partial X} + \frac{\partial V_1(x)(X,Z,T)}{\partial T} = 0 \\ 3 & \frac{\partial P_1(X,Z,T)}{\partial Z} = 0 \\ 4 & V_1(z)(X, B(X), T) = \frac{\partial B(X)}{\partial X} V_1(x)(X, B(X), T) \\ 5 & V_1(z)(X, 1, T) = \frac{\partial H_1(X,T)}{\partial T} \\ 6 & H_1(X, T) = P_1(X, 1, T) \end{pmatrix}$$

Note that the equation 3 implies that pressure perturbations P_1 are independent of the coordinate Z. I impose this condition with the following rule.

In[82]:=
```
w23[1] = w22[5] /. P1 → (P1[#1, #3] &)
```

Out[82]=
```
{V1[z]^(0,1,0)[X, Z, T] + V1[x]^(1,0,0)[X, Z, T] == 0,
 P1^(1,0)[X, T] + V1[x]^(0,0,1)[X, Z, T] == 0, True, V1[z][X, B[X], T] == V1[x][X, B[X], T] B′[X],
 V1[z][X, 1, T] == H1^(0,1)[X, T], H1[X, T] == P1[X, T]}
```

In[83]:=
```
DisplayEqns[w23[1], {V1 → V₁, P1 → P₁, H1 → H₁}]
```

Out[83]//TraditionalForm=

$$\begin{pmatrix} 1 & \frac{\partial V_1(x)(X,Z,T)}{\partial X} + \frac{\partial V_1(z)(X,Z,T)}{\partial Z} = 0 \\ 2 & \frac{\partial P_1(X,T)}{\partial X} + \frac{\partial V_1(x)(X,Z,T)}{\partial T} = 0 \\ 3 & \text{True} \\ 4 & V_1(z)(X, B(X), T) = \frac{\partial B(X)}{\partial X} V_1(x)(X, B(X), T) \\ 5 & V_1(z)(X, 1, T) = \frac{\partial H_1(X,T)}{\partial T} \\ 6 & H_1(X, T) = P_1(X, T) \end{pmatrix}$$

Further, equation 2 implies that $V_1(x)$ cannot depend on Z either.

In[84]:=
```
w23[2] = Drop[w23[1], {3}] /. V1[x] → (V1[x][#1, #3] &)
```

Out[84]=
```
{V1[x]^(1,0)[X, T] + V1[z]^(0,1,0)[X, Z, T] == 0, V1[x]^(0,1)[X, T] + P1^(1,0)[X, T] == 0,
 V1[z][X, B[X], T] == V1[x][X, T] B′[X], V1[z][X, 1, T] == H1^(0,1)[X, T], H1[X, T] == P1[X, T]}
```

In[85]:=
```
DisplayEqns[w23[2], {V1 → V₁, P1 → P₁, H1 → H₁}]
```

Out[85]//TraditionalForm=

$$\begin{pmatrix} 1 & \frac{\partial V_1(x)(X,T)}{\partial X} + \frac{\partial V_1(z)(X,Z,T)}{\partial Z} = 0 \\ 2 & \frac{\partial P_1(X,T)}{\partial X} + \frac{\partial V_1(x)(X,T)}{\partial T} = 0 \\ 3 & V_1(z)(X,\, B(X),\, T) = \frac{\partial B(X)}{\partial X}\, V_1(x)(X,\, T) \\ 4 & V_1(z)(X,\, 1,\, T) = \frac{\partial H_1(X,T)}{\partial T} \\ 5 & H_1(X,\, T) = P_1(X,\, T) \end{pmatrix}$$

Equation 5 can be used to eliminate P1 from the system

In[86]:=
```
P1Rule = (D[#, X] & /@ Solve[w23[2][[5]], P1[X, T]] [[1, 1]])
```

Out[86]=
```
P1^(1,0)[X, T] → H1^(1,0)[X, T]
```

In[87]:=
```
w23[3] = w23[2][[1 ;; 4]] /. P1Rule
```

Out[87]=
```
{V1[x]^(1,0)[X, T] + V1[z]^(0,1,0)[X, Z, T] == 0, V1[x]^(0,1)[X, T] + H1^(1,0)[X, T] == 0,
 V1[z][X, B[X], T] == V1[x][X, T] B′[X], V1[z][X, 1, T] == H1^(0,1)[X, T]}
```

In[88]:=
```
DisplayEqns[w23[3], {V1 → V₁, P1 → P₁, H1 → H₁}]
```

Out[88]//TraditionalForm=

$$\begin{pmatrix} 1 & \frac{\partial V_1(x)(X,T)}{\partial X} + \frac{\partial V_1(z)(X,Z,T)}{\partial Z} = 0 \\ 2 & \frac{\partial H_1(X,T)}{\partial X} + \frac{\partial V_1(x)(X,T)}{\partial T} = 0 \\ 3 & V_1(z)(X,\, B(X),\, T) = \frac{\partial B(X)}{\partial X}\, V_1(x)(X,\, T) \\ 4 & V_1(z)(X,\, 1,\, T) = \frac{\partial H_1(X,T)}{\partial T} \end{pmatrix}$$

Notice that equation 1 can be integrated with respect to Z from B[X] to 1 and that equations 3 and 4 provide the boundary values.

In[89]:=
```
w23[4] = Solve[w23[2][[1]], V1[z]^(0,1,0)[X, Z, T]][[1, 1]] /. Rule → Equal
```

Out[89]=
```
V1[z]^(0,1,0)[X, Z, T] == -V1[x]^(1,0)[X, T]
```

Integrate with respect to Z, from the bottom Z = B[x] to the surface Z = 1

In[90]:=
```
w23[5] = (Integrate[#, {Z, B[X], 1}] &) /@ w23[4]
```

Out[90]=

$$\int_{B[X]}^{1} V1[z]^{(0,1,0)}[X, Z, T]\, dZ == -\left(1 - B[X]\right) V1[x]^{(1,0)}[X, T]$$

In[91]:=
```
w23[6] = w23[5] /. ∫_{B[X]}^1 V1[z]^(0,1,0)[X, Z, T] ⅆZ → V1[z][X, 1, T] - V1[z][X, B[X], T]
```
Out[91]=
$$V1[z][X, 1, T] - V1[z][X, B[X], T] == -(1 - B[X])\, V1[x]^{(1,0)}[X, T]$$

In[92]:=
```
boundaryRules = {w23[3][[3]], w23[3][[4]]} // ER
```
Out[92]=
$$\{V1[z][X, B[X], T] \to V1[x][X, T]\, B'[X],\ V1[z][X, 1, T] \to H1^{(0,1)}[X, T]\}$$

In[93]:=
```
w23[7] = w23[6] /. boundaryRules
```
Out[93]=
$$-V1[x][X, T]\, B'[X] + H1^{(0,1)}[X, T] == -(1 - B[X])\, V1[x]^{(1,0)}[X, T]$$

The remaining equations are

In[94]:=
```
w23[8] = {w23[7], w23[3][[2]]}
```
Out[94]=
$$\{-V1[x][X, T]\, B'[X] + H1^{(0,1)}[X, T] == -(1 - B[X])\, V1[x]^{(1,0)}[X, T],$$
$$V1[x]^{(0,1)}[X, T] + H1^{(1,0)}[X, T] == 0\}$$

In[95]:=
```
DisplayEqns[w23[8], {V1 → V₁, P1 → P₁, H1 → H₁}]
```
Out[95]//TraditionalForm=
$$\begin{pmatrix} 1 & \frac{\partial H_1(X,T)}{\partial T} - \frac{\partial B(X)}{\partial X} V_1(x)(X,T) = -(1 - B(X)) \frac{\partial V_1(x)(X,T)}{\partial X} \\ 2 & \frac{\partial H_1(X,T)}{\partial X} + \frac{\partial V_1(x)(X,T)}{\partial T} = 0 \end{pmatrix}$$

Differentiate equation 1 with respect to T and use equation 2 to eliminate $V_1[x]$

In[96]:=
```
w23[9] = MapEqn[D[#, T] &, w23[8][[1]] ]
```
Out[96]=
$$-B'[X]\, V1[x]^{(0,1)}[X, T] + H1^{(0,2)}[X, T] == -(1 - B[X])\, V1[x]^{(1,1)}[X, T]$$

From equation (2)

In[97]:=
```
VXrule = Solve[w23[8][[2]], V1[x]^(0,1)[X, T]][[1, 1]]
```
Out[97]=
$$V1[x]^{(0,1)}[X, T] \to -H1^{(1,0)}[X, T]$$

In[98]:=
```
VXrule = {VXrule, D[#, X] & /@ VXrule}
```
Out[98]=
$$\{V1[x]^{(0,1)}[X, T] \to -H1^{(1,0)}[X, T],\ V1[x]^{(1,1)}[X, T] \to -H1^{(2,0)}[X, T]\}$$

In[99]:=
```
w23[10] = w23[9] /. VXrule
```
Out[99]=
$$H1^{(0,2)}[X, T] + B'[X]\, H1^{(1,0)}[X, T] == (1 - B[X])\, H1^{(2,0)}[X, T]$$

I finally have a single equation for the surface perturbation H_1.

In[100]:=
```
DisplayEqns[{w23[10]}, {H1 → H₁}]
```

Out[100]//TraditionalForm=

$$\left(1 \quad \frac{\partial B(X)}{\partial X}\frac{\partial H_1(X,T)}{\partial X}+\frac{\partial^2 H_1(X,T)}{\partial T^2}=(1-B(X))\frac{\partial^2 H_1(X,T)}{\partial X^2}\right)$$

which is the sought after wave equation for water waves in shallow water.

The rhs of this equation can be written in a more compact form by introducing a “depth function” defined by

In[101]:=
```
def[𝒟] = 𝒟[X] == 1 - B[x]
```

Out[101]=
```
𝒟[X] == 1 - B[x]
```

In[102]:=
```
w23[11] = Solve[def[𝒟], B[x]][[1, 1]]
```

Out[102]=
```
B[x] → 1 - 𝒟[X]
```

In[103]:=
```
w23[12] = w23[10] /. B → ((1 - 𝒟[#]) &) // Expand
```

Out[103]=
```
H1^(0,2)[X, T] - 𝒟'[X] H1^(1,0)[X, T] == 𝒟[X] H1^(2,0)[X, T]
```

which can be rewritten as

In[104]:=
```
w23[13] = H1^(0,2)[X, T] - HoldForm[D[𝒟[X] H1^(1,0)[X, T], X]] == 0
```

Out[104]=
```
-∂X(𝒟[X] H1^(1,0)[X, T]) + H1^(0,2)[X, T] == 0
```

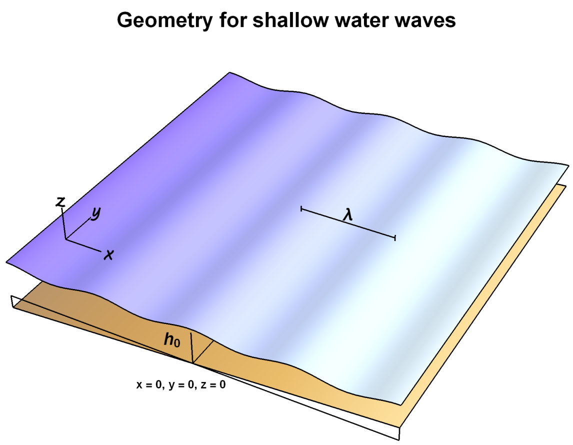
which, finally, is the shallow water wave equation.

Appendix - Generation of figures

In[105]:=

```
Module[{λ = 40, h0 = 1.0, xMax = 80, yMax = 40, δ = 0.1, L = 200, d = 2,
  zMin = -2, zMax = 2.5, Surface, Bottom, refLines, axes, labels, Seq, g},
 Surface[x_, y_, λ_] := 1 + δ Cos[2 π x/λ];
 Bottom[x_, y_, L_] := x/L;
 Seq[colorScheme_] := Sequence[
   ColorFunction → (ColorData[colorScheme][Rescale[#, {0, 1}, {0.25, 0.75}]] &),
   Mesh → False, Axes → False, Boxed → False, PlotPoints → 50,
   PlotRange → {{-xMax, xMax}, {-yMax, yMax}, {zMin, zMax}}, ImageSize → 300];

 g[1] = Plot3D[Surface[x, y, λ],
   {x, -xMax, xMax}, {y, -yMax, yMax}, Evaluate@Seq["LakeColors"],
   PlotLabel → Stl["Geometry for shallow water waves"]];

 g[2] = Plot3D[Bottom[x, y, L], {x, -xMax, xMax},
   {y, -yMax, yMax}, Evaluate@Seq["CoffeeTones"]];

 refLines =
  Module[{ℒ = Line, x = xMax, y = yMax},
   {{Black, ℒ[{{0, -y, 0}, {0, y, 0}}],
     ℒ[{{0, -y, 0}, {0, -y, 1}}], ℒ[{{-x, -y, 0}, {x, -y, 0}}],
     ℒ[{{-x, -y, 0}, {-x, -y, Bottom[-x, -y, L]}}],
     ℒ[{{x, -y, 0}, {x, -y, Bottom[x, -y, L]}}]}}];

 axes =
  Module[{O = {-0.8 xMax, -0.8 yMax, 1.5}, xS = xMax / 5, yS = yMax / 5, zS = 1},
   {Black, Arrowheads[Tiny], Arrow[{O, O + {xS, 0, 0}}],
    Arrow[{O, O + {0, yS, 0}}], Arrow[{O, O + {0, 0, zS}}],
    Text[Stl["x"], {O + {1.2 xS, 0, 0}}],
    Text[Stl["y"], {O + {0, 1.2 yS, 0}}],
    Text[Stl["z"], {O + {0, 0, 1.2 zS}}]}];

 labels =
  Module[{pλ1, pλ2, off},
   pλ1 = {0, 0, 0.1 + Surface[0, 0, λ]};
   pλ2 = {λ, 0, 0.1 + Surface[λ, 0, λ]};
   off = {0, 0, 0.1};
   {Black, Line[{pλ1, pλ2}],
    Line[{pλ1, pλ1 - off}], Line[{pλ1, pλ1 + off}],
    Line[{pλ2, pλ2 - off}], Line[{pλ2, pλ2 + off}],
    Text[Stl["λ"], {(pλ1 + pλ2)/2 + 2.5 off}],
    Text[Stl["h0"], {-(xMax / 10), -yMax, 0.5}],
    Text[Style["x = 0, y = 0, z = 0", Bold, Tiny], {0, -1.1 yMax, -0.25}]}];

 g[3] = Graphics3D[{refLines, axes, labels},
   PlotRange → {{-xMax, xMax}, {-yMax, yMax}, {-2, 2}}];
 Show[{g[1], g[2], g[3]} ]]
```

Out[105]=



Display of partial derivatives

The following magical function comes from Vitaly Kaurov
http://blog.wolfram.com/2011/12/15/mathematica-qa-series-converting-to-conventional-mathematical-typesetting/

In[106]:=

```
Clear[pdConv];
pdConv[f_] :=
 TraditionalForm[
  f /. Derivative[inds__][g_][vars__] :>
    Apply[Defer[D[g[vars], ##]] &,
     Transpose[{{vars}, {inds}}] /.
      {{var_, 0} :> Sequence[], {var_, 1} :> {var}}]]
```

I generate a convenience function for displaying a numbered list of equations

In[108]:=

```
Clear[DisplayEqns];
DisplayEqns[eqns_, subs_] :=
 Module[{neqns},
  neqns = Transpose[{Range[1, Length[eqns]], eqns}] /. subs // PhysicsForm]
```

In[110]:=

```
Clear[MyDisplayForm];
MyDisplayForm[nums_, eqns_] :=
 Module[{neqns},
  neqns = Transpose[{nums, eqns}] /. {V1 → V1, P1 → P1, H1 → H1} // pdConv]
```