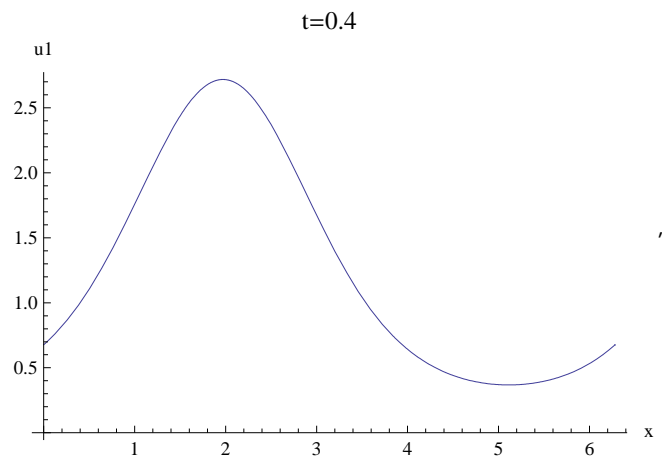
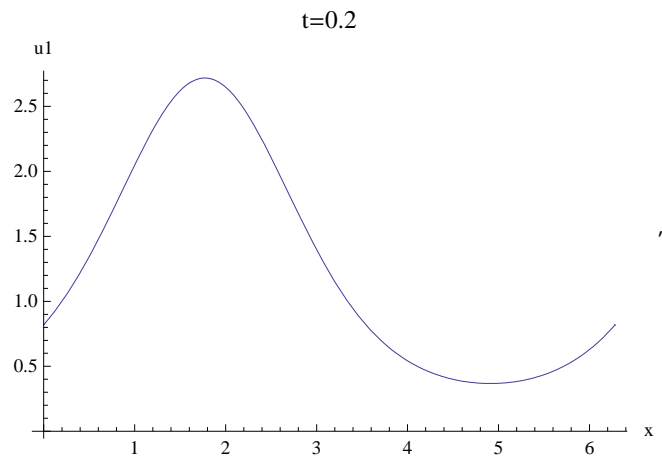
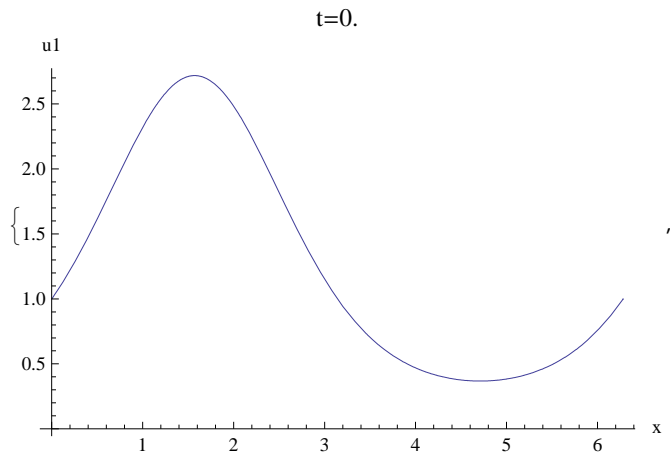
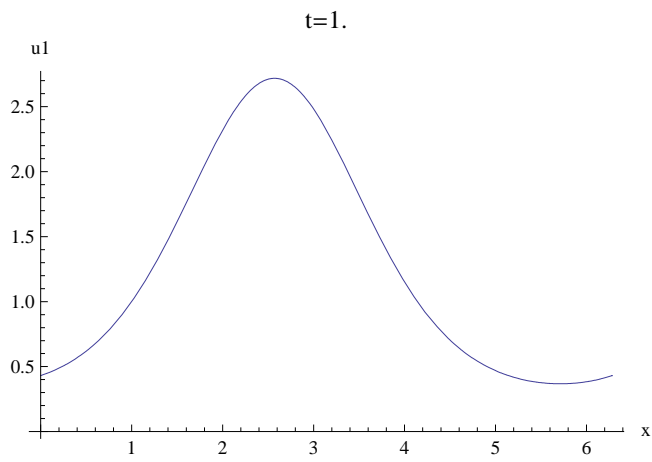
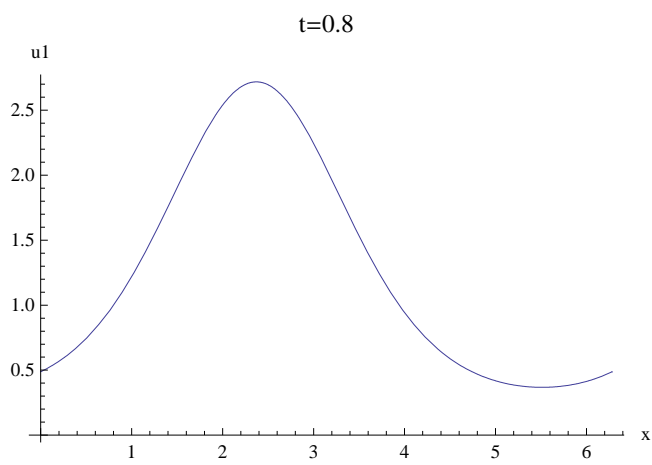
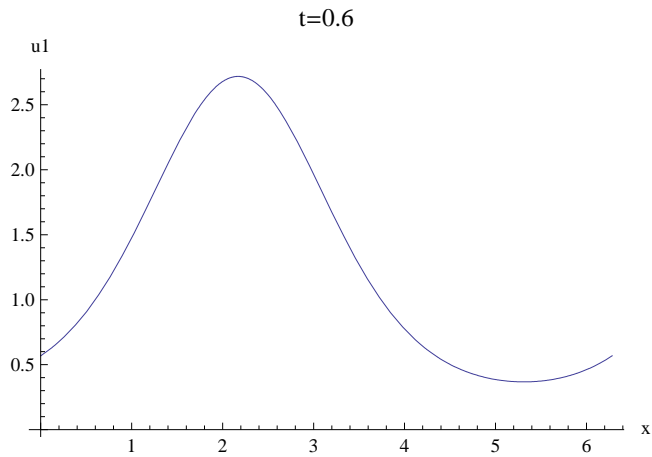
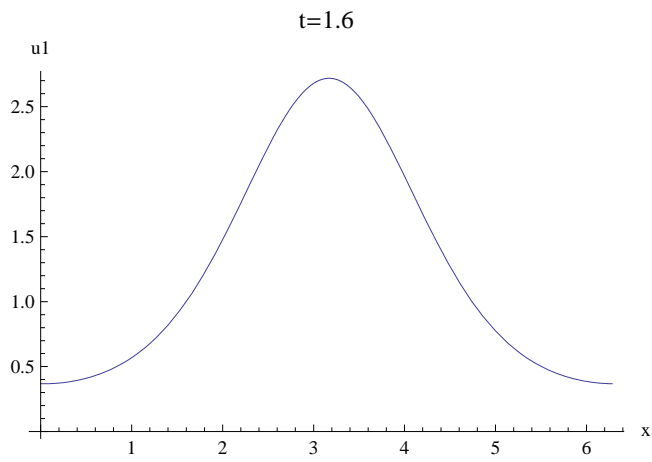
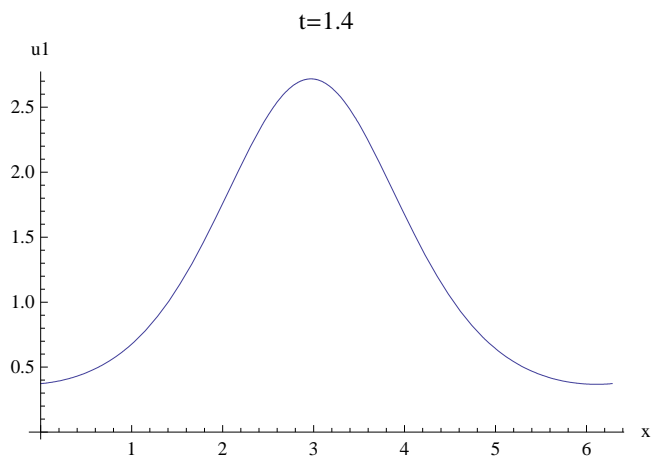
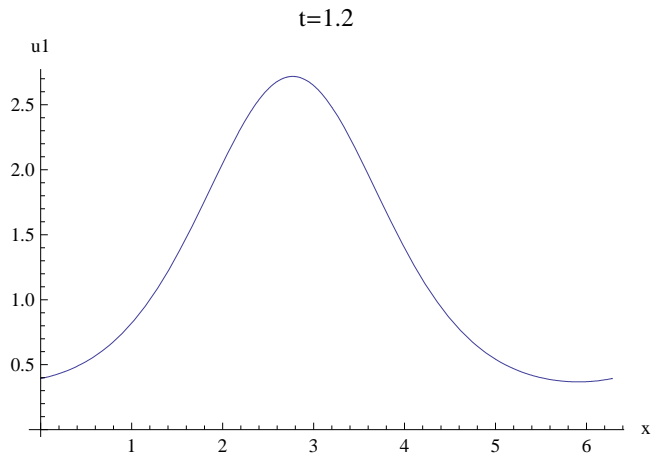

Wave Equations

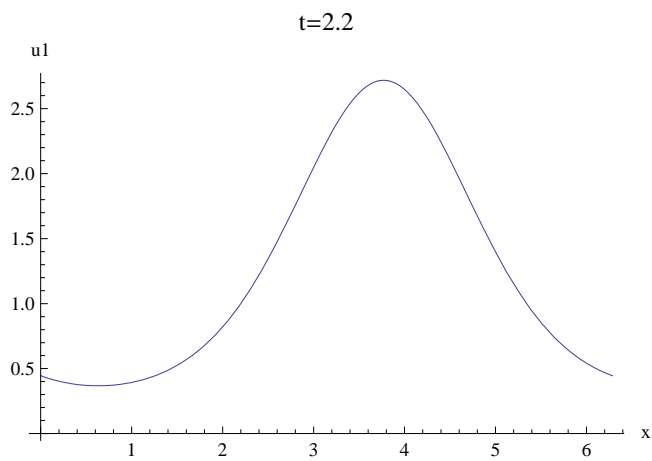
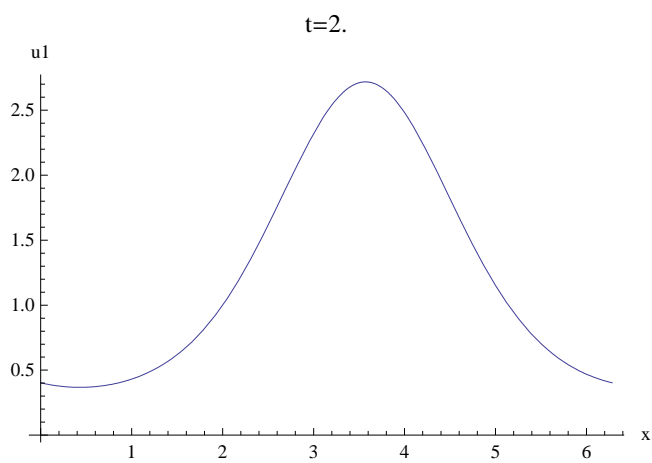
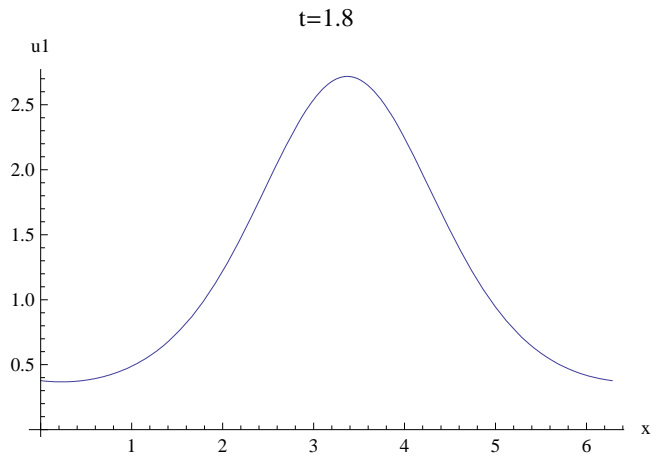
Solutions of the homogeneous Wave Equation with speed one are of the form $f[x+t]+g[x-t]$. This looks fairly simple.

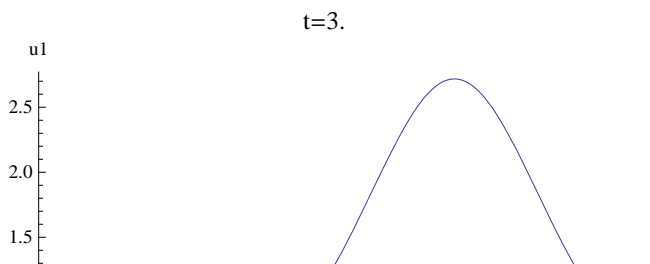
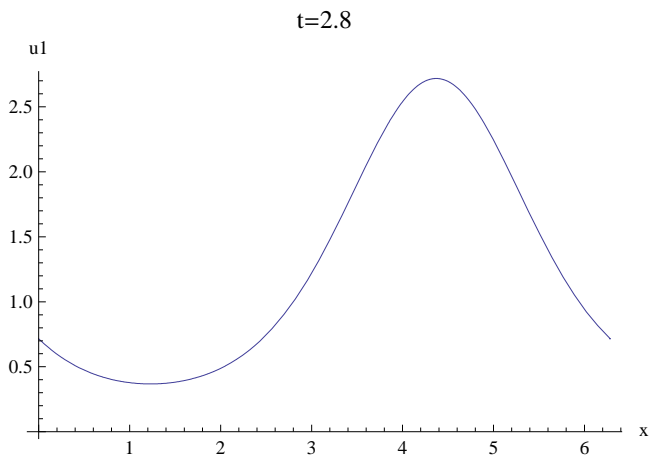
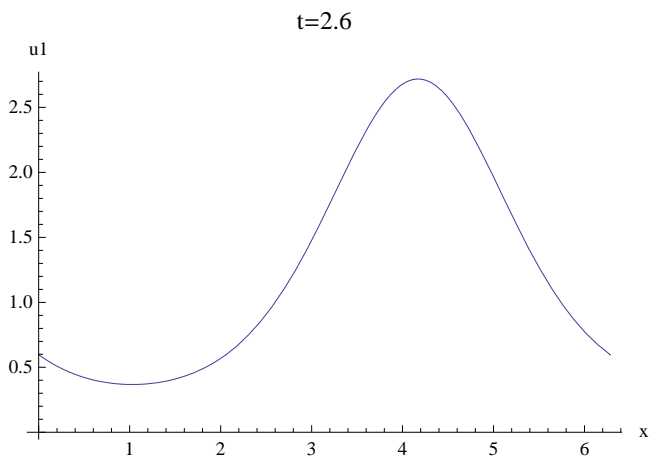
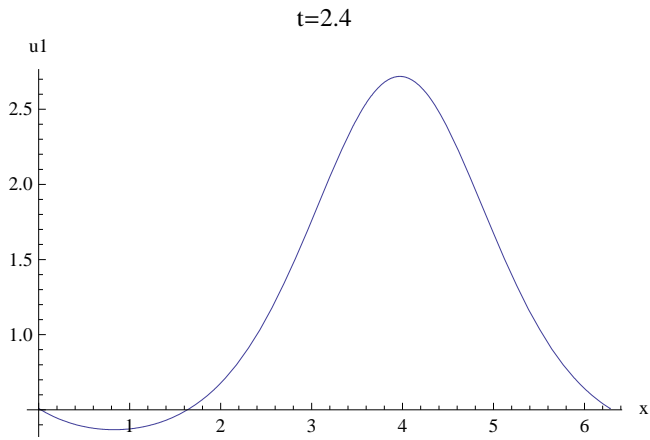
```
u1[x_, t_] = Esin[x-t];
u2[x_, t_] = Cos[2 (x + t)];
Table[
  Plot[u1[x, t], {x, 0, 2 π},
    PlotLabel → StringForm["t=``", t],
    AxesLabel → {"x", "u1"}
  ],
  {t, 0, 3, 0.2}
]
Table[
  Plot[u2[x, t], {x, 0, 2 π},
    PlotLabel → StringForm["t=``", t],
    AxesLabel → {"x", "u2"}
  ],
  {t, 0, 3, 0.2}
]
Table[
  Plot[u1[x, t] + u2[x, t], {x, 0, 2 π},
    PlotLabel → StringForm["t=``", t],
    AxesLabel → {"x", "u1+u2"}
  ],
  {t, 0, 3, 0.2}
]
```

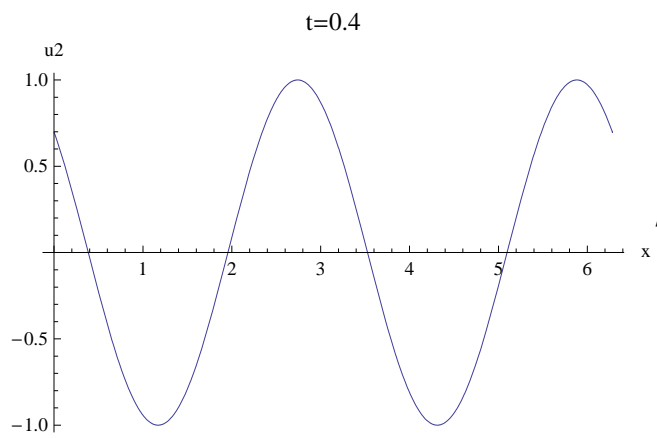
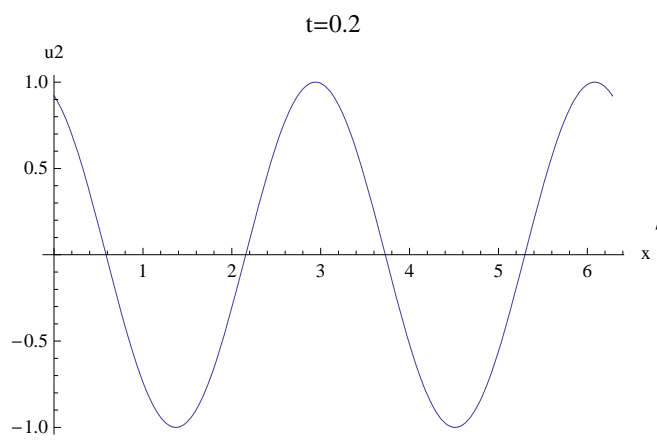
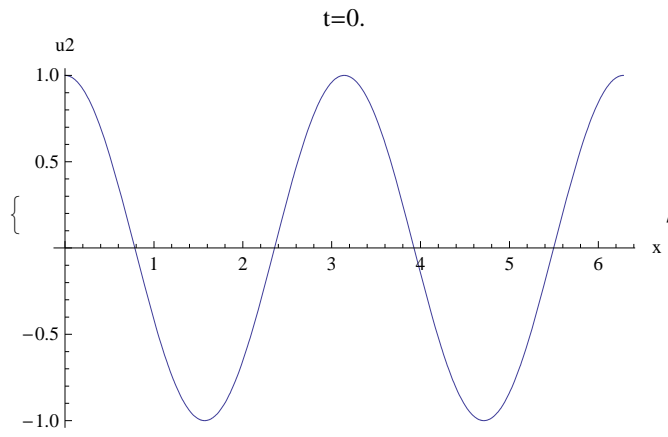


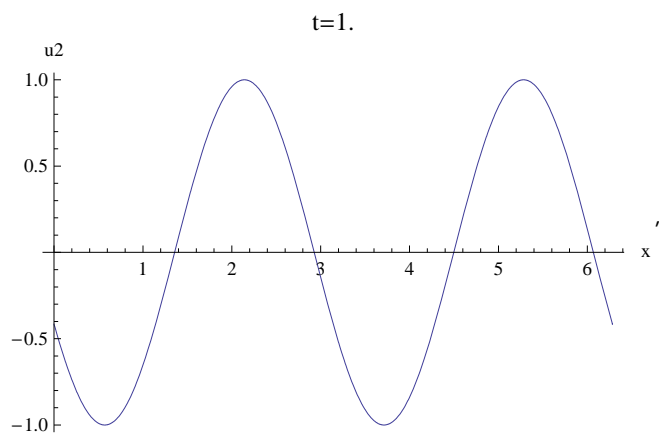
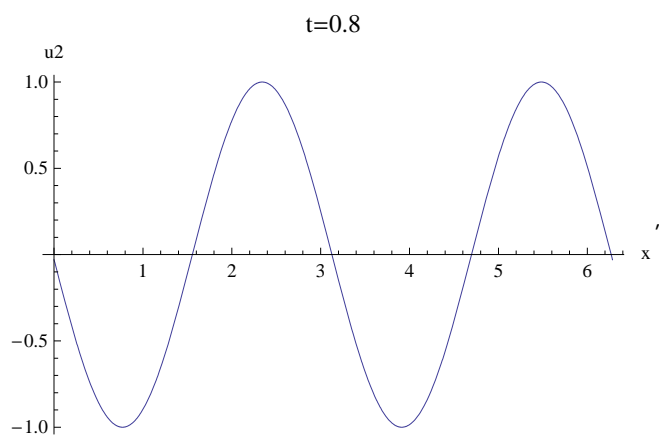
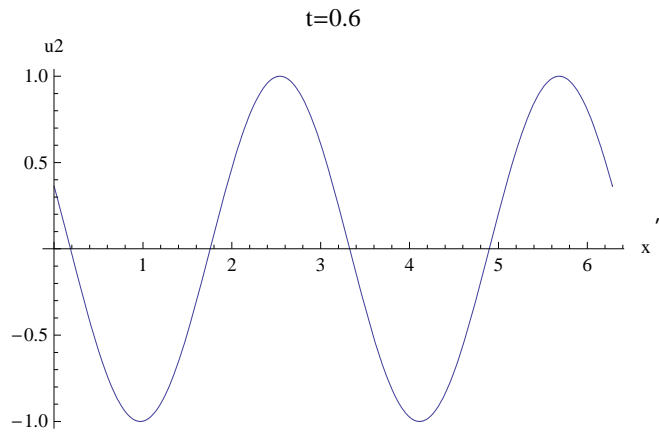


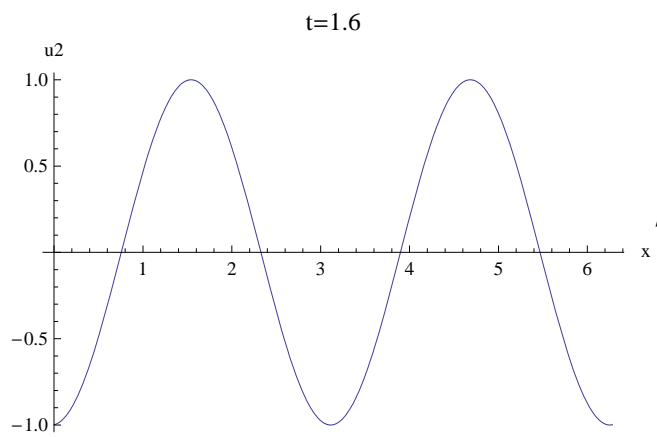
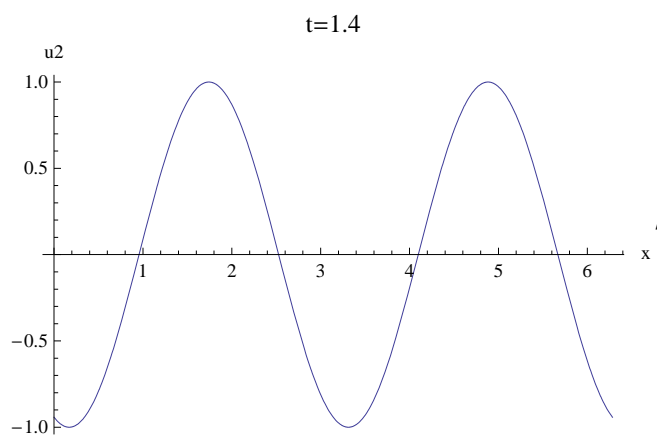
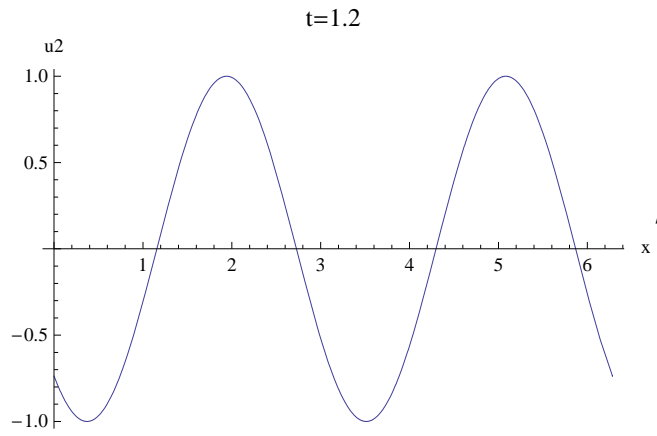


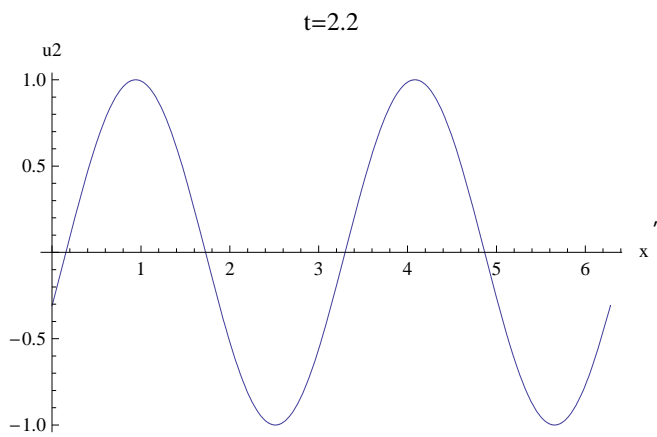
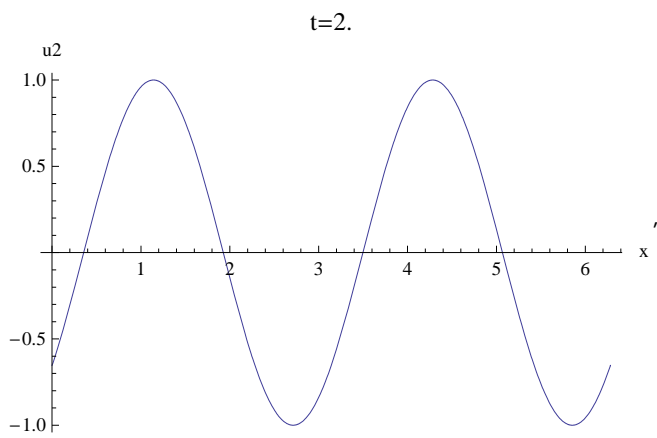
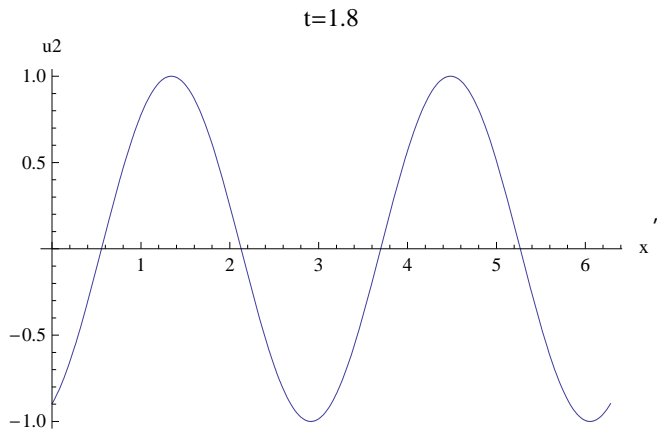


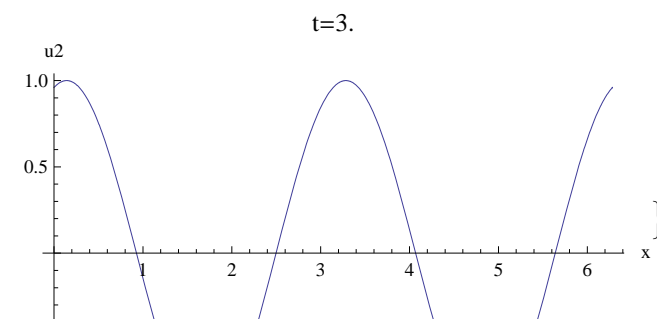
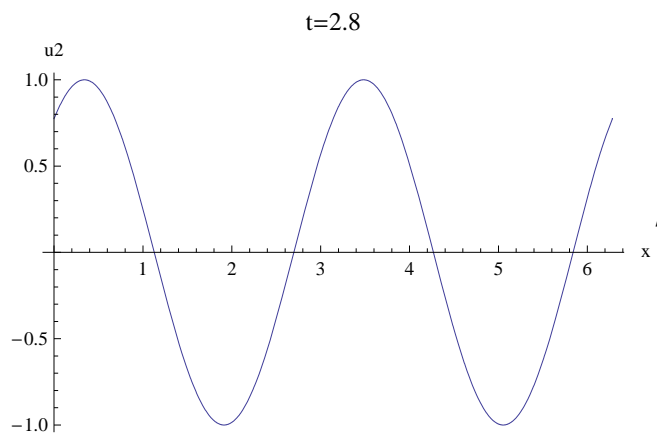
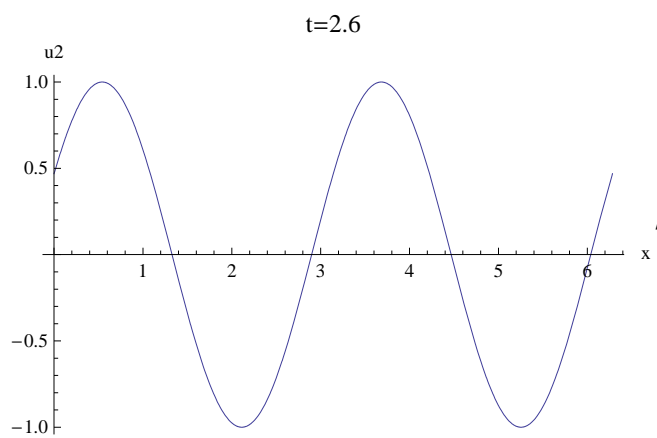
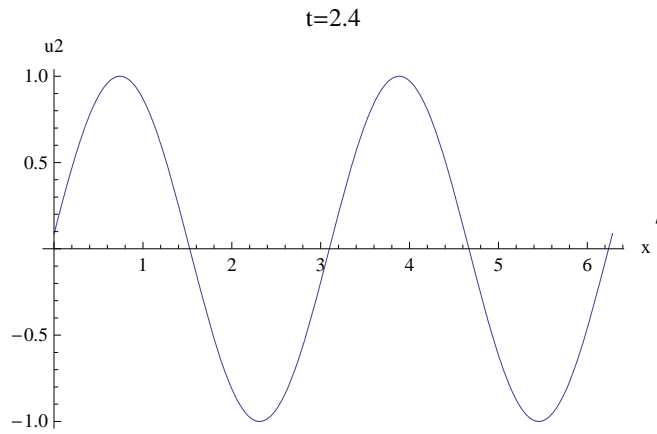


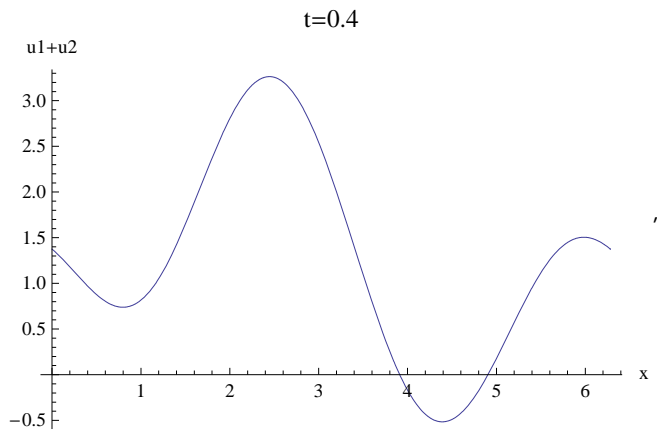
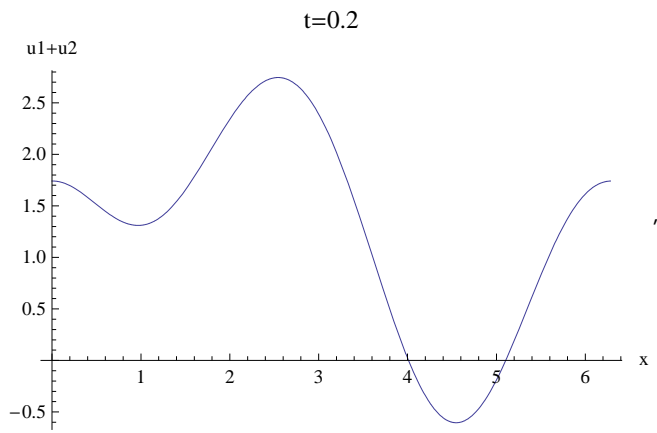
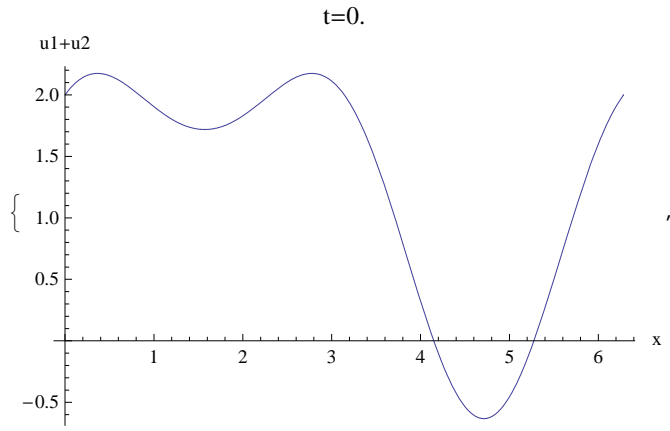


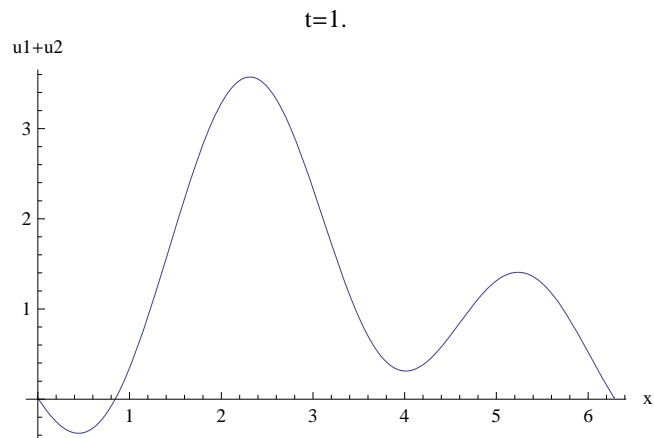
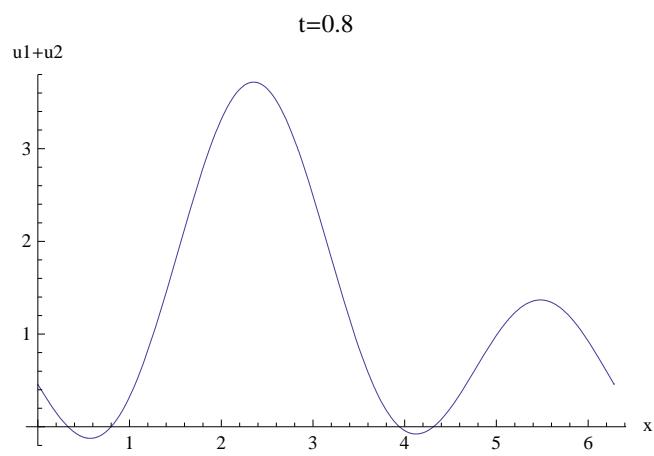
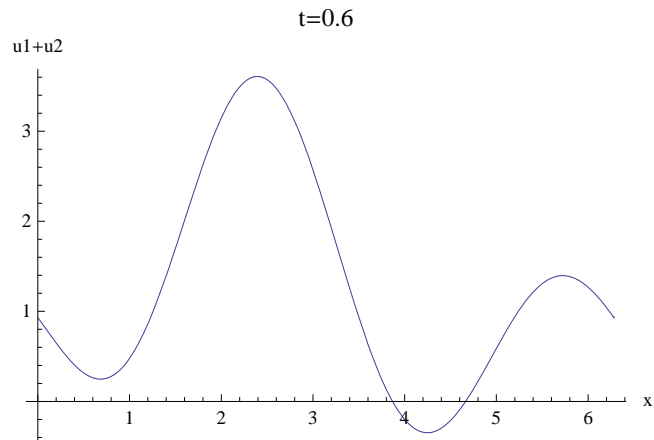


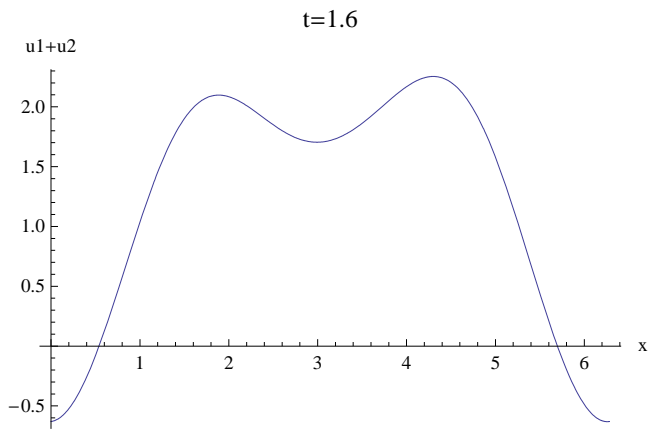
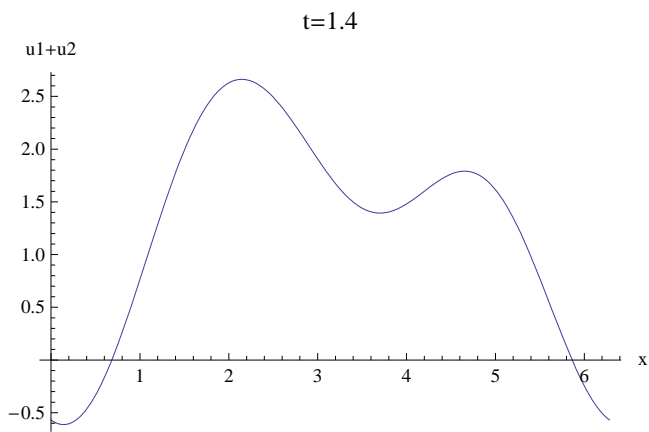
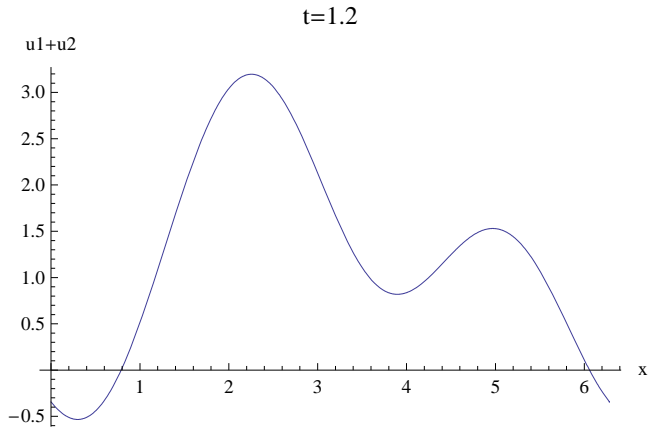


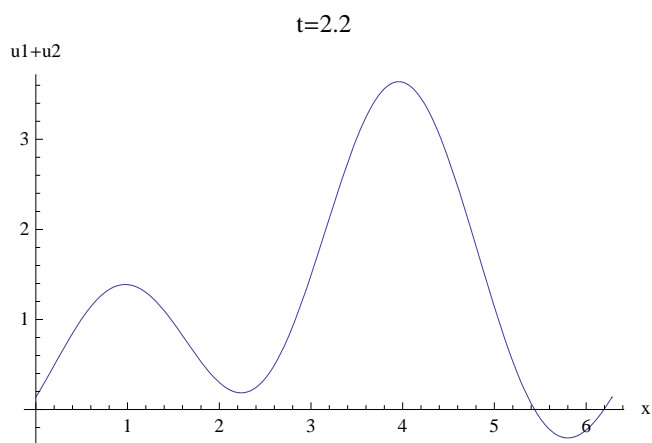
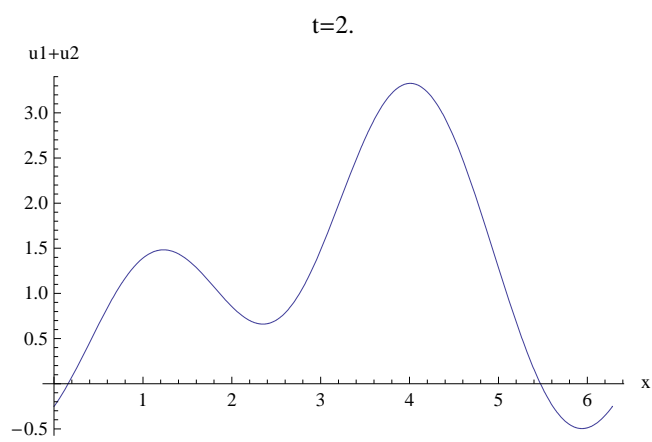
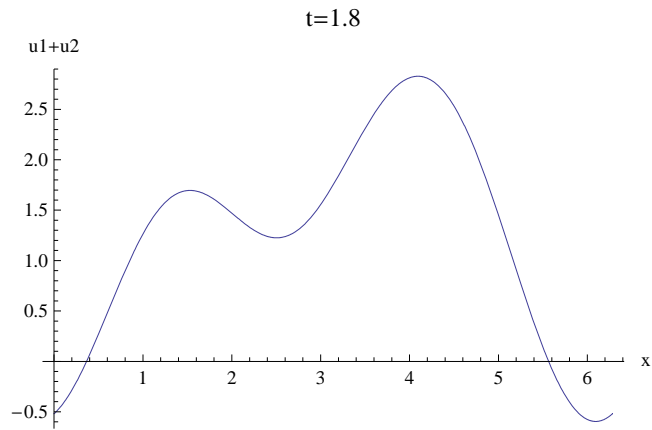


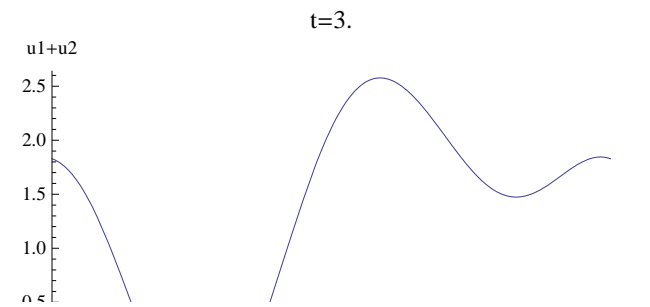
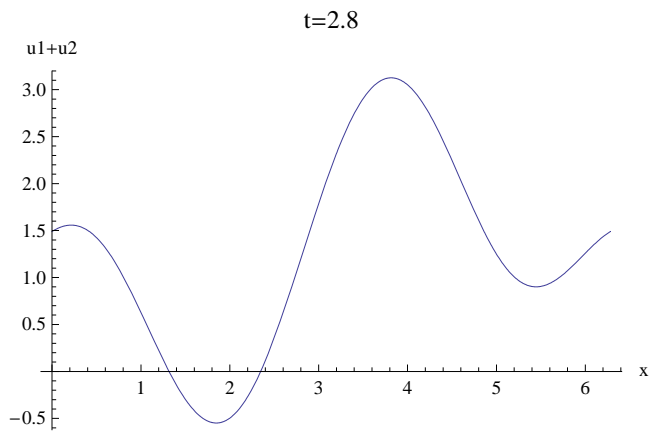
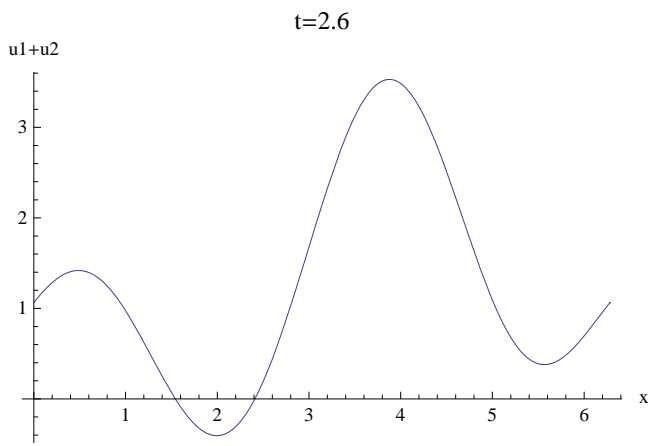
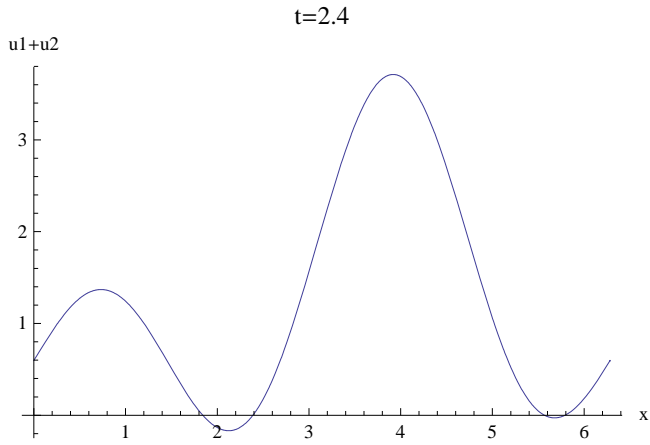








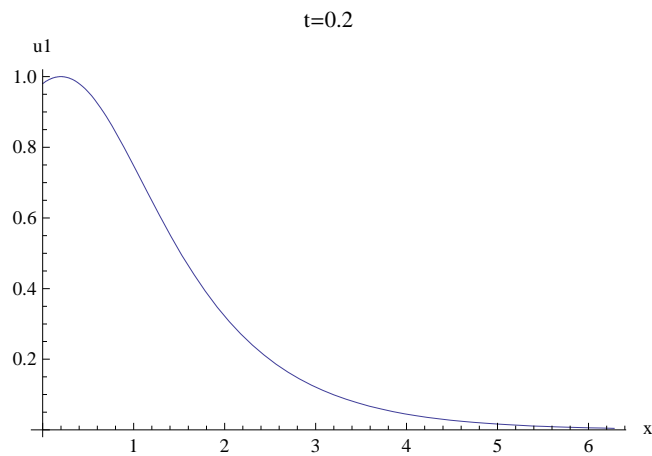
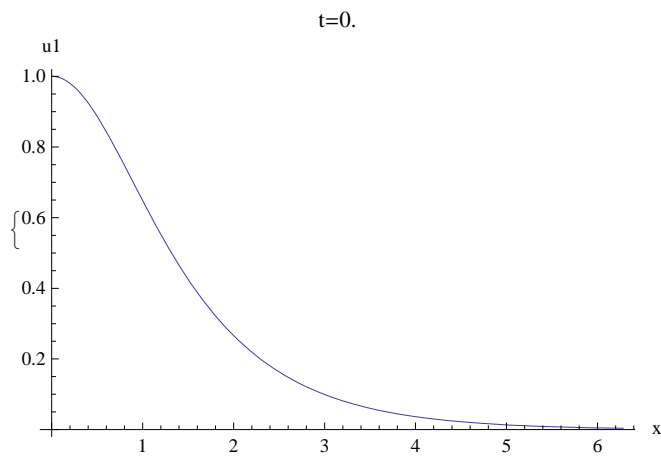


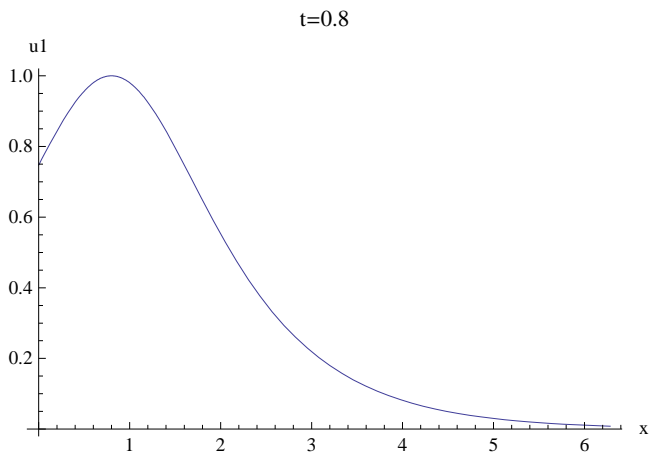
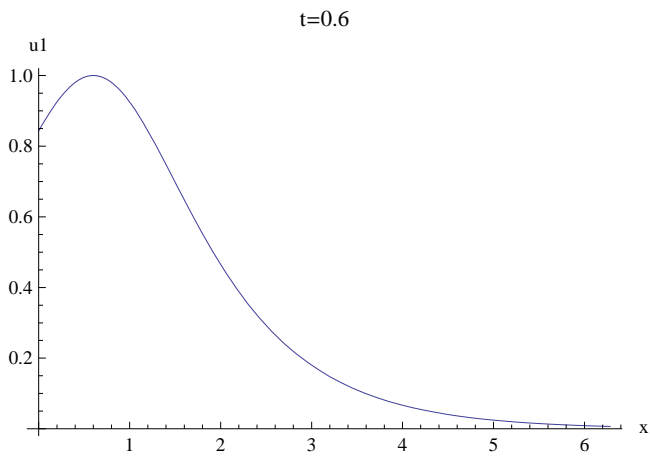
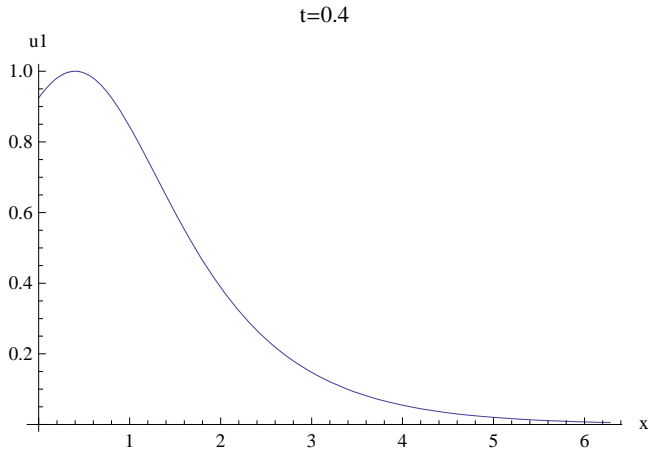


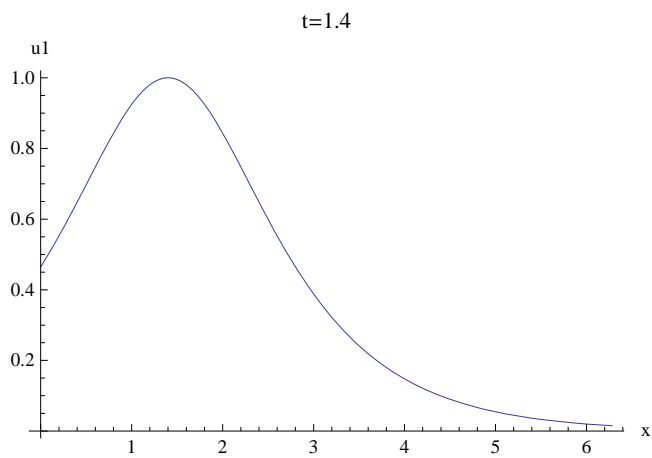
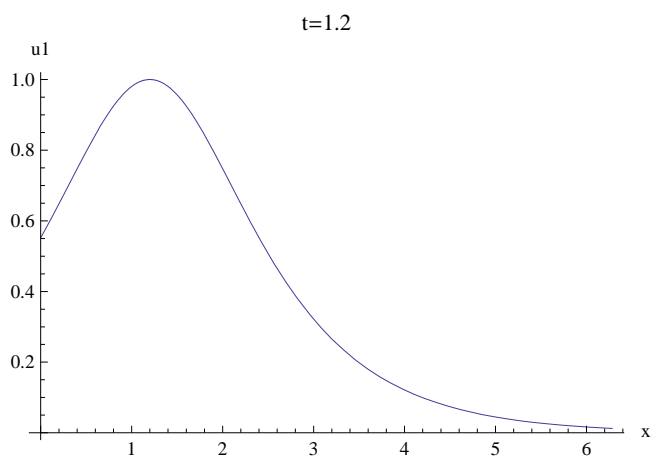
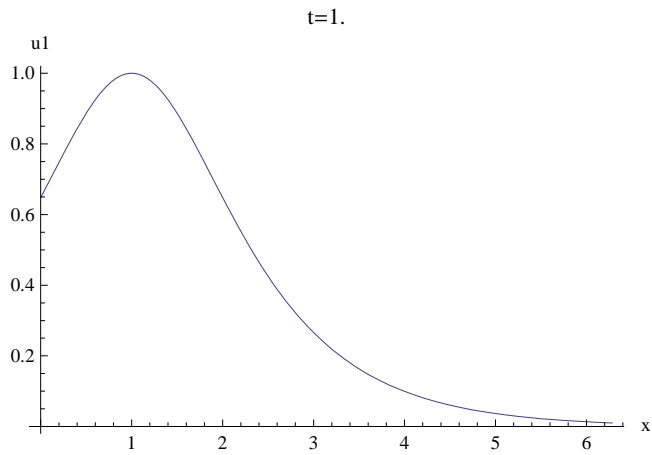
One Directional Wave Equation

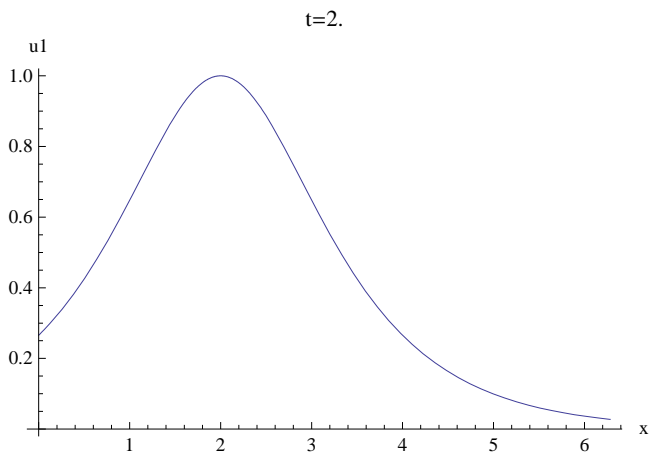
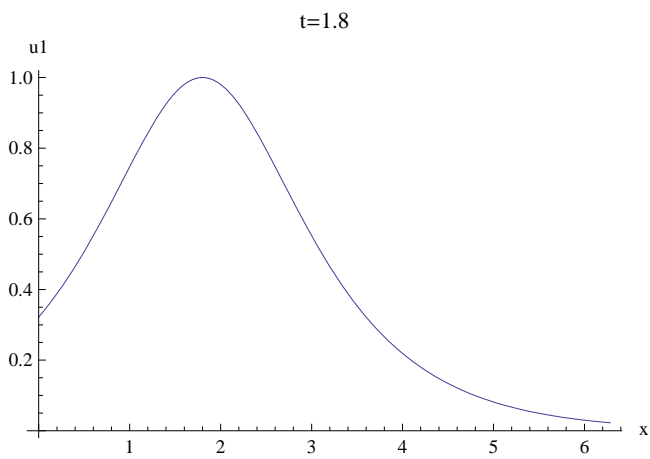
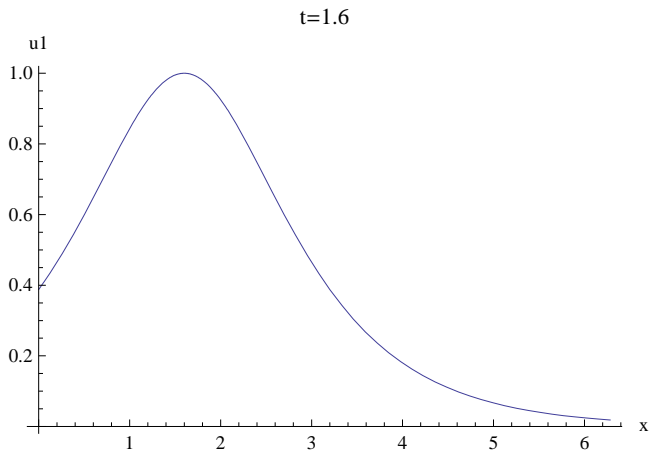
The uni-directional, homogeneous, wave equation with speed one has solutions of the form $f[x-t]$. This represents a wave traveling to the right at speed one.

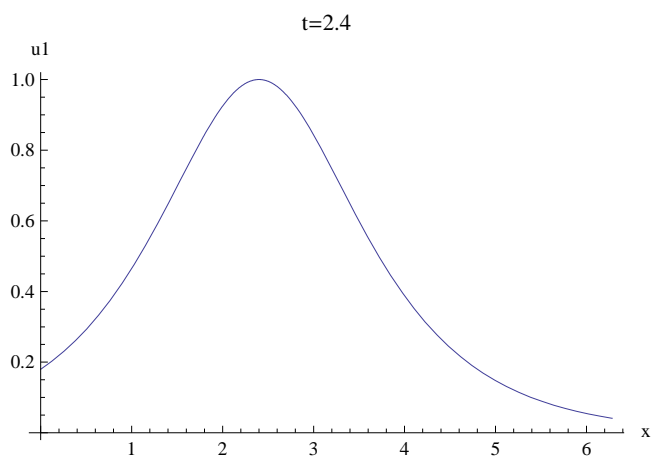
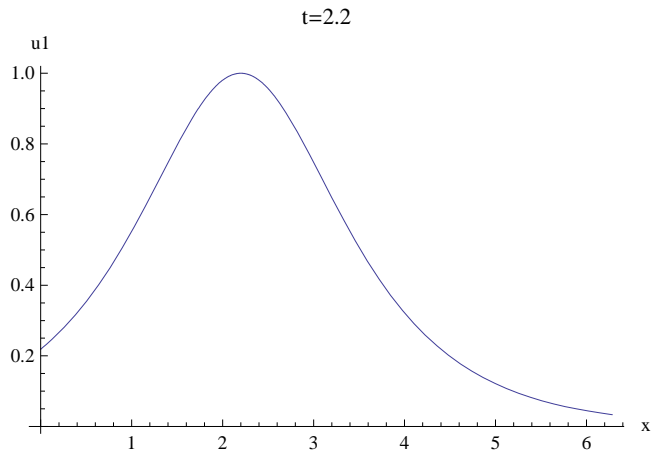
```
u1[x_, t_] = Sech[x - t];  
Table[  
  Plot[u1[x, t], {x, 0, 2  $\pi$ },  
    PlotLabel  $\rightarrow$  StringForm["t=``", t],  
    AxesLabel  $\rightarrow$  {"x", "u1"}  
  ],  
{t, 0, 3, 0.2}
```

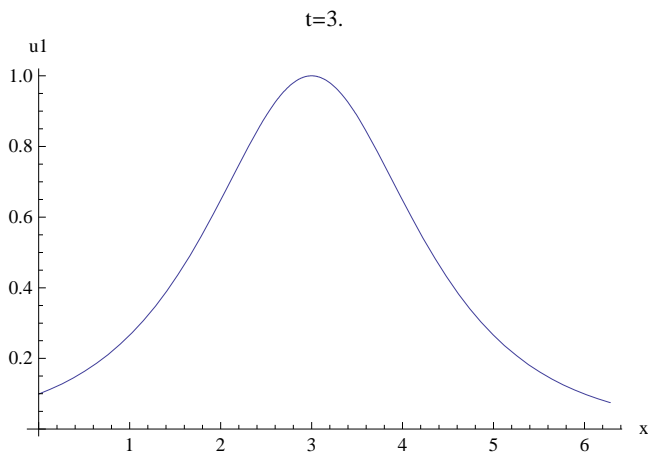
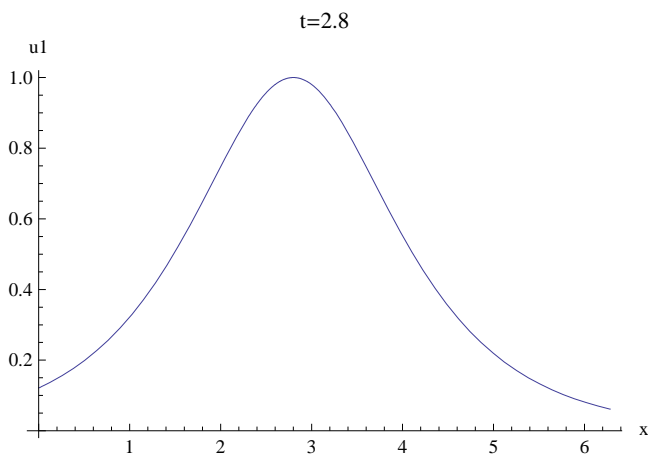
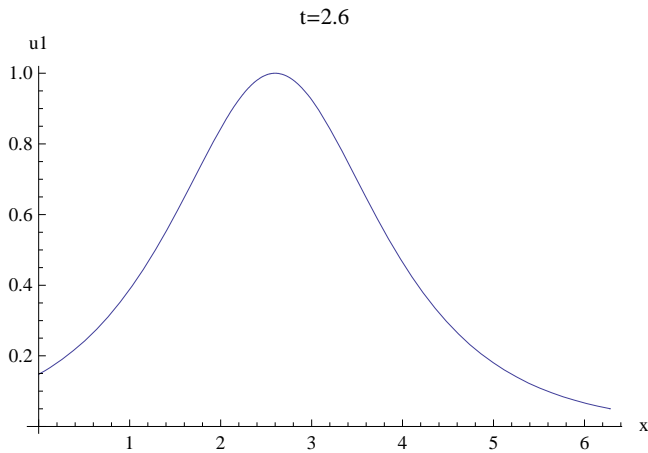












Characteristics

Information in wave equations travels along specific physical families of curves called characteristics. The unidirectional wave equation with speed one has one family of characteristics $x - t = c_1$. The wave equation with speed one has two: $x - t = c_1$ and $x + t = c_2$.

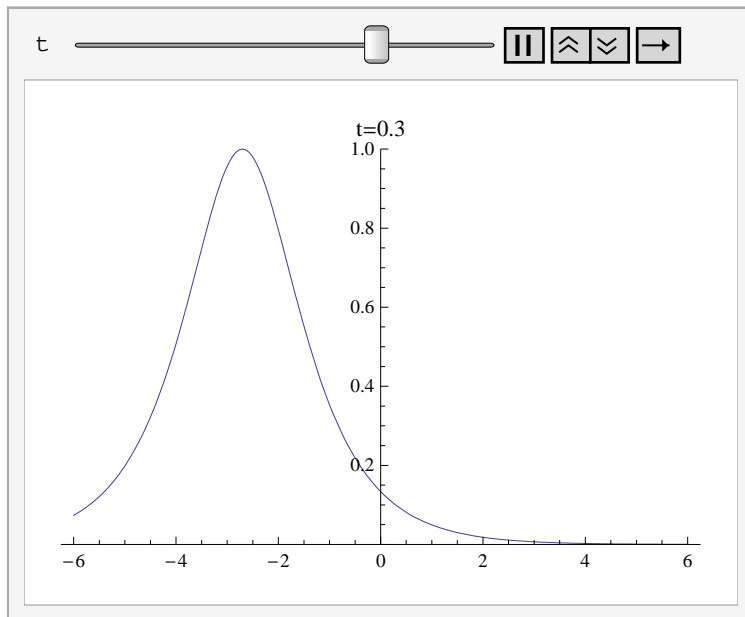
The existence of such curves is what makes the equations hyperbolic and the characteristics have serious physical implications for designing numerical schemes to solve hyperbolic equations. We are going to try to understand this by looking at the model equations above.

The existence of such curves is also what makes the equations easy to find test solutions for. For the unidirectional equation: 1) Pick a function f . 2) A solution of $u_t + \alpha u_x = 0$ is $u = f(x - \alpha t)$. This is easy to check: just differentiate and plug in!

```
In[34]:= f[τ_] = Sech[τ + 3];
α = 1;
u[x_, t_] = f[x - α t];
D[u[x, t], t] + α D[u[x, t], x]
AniPic = Animate[
  Plot[u[x, t], {x, -6, 6},
    PlotRange -> {0, 1},
    PlotLabel -> StringForm["t=``", t]
  ],
  {t, 0, 4, 0.05}]
Export["Wave.gif", AniPic]
```

Out[37]= 0

Out[38]=



Out[5]= Wave.gif

Numerics

$u_t + \alpha u_x = f$ discretizes as follows with positive differences for each derivative.

$$U_j^{n+1} = U_j^n - \alpha \frac{\Delta t}{\Delta x} (U_{j+1}^n - U_j^n) + \Delta t F_j^n$$

The IC sets U_j^0 and then we simply fill in the values using the update formula. The one problem is what to do when the $j + 1$ runs above the appropriate range. Well the periodic boundary conditions say simply to wrap it around to the first value of j . I am going to write this in our matrix form cause it will be easier to do the stability analysis. We write $\eta = \alpha \frac{\Delta t}{\Delta x}$ and have the update formula

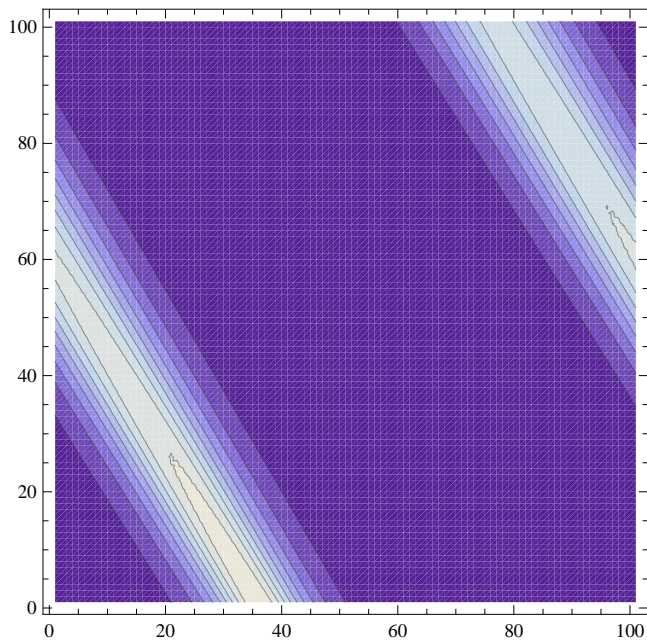
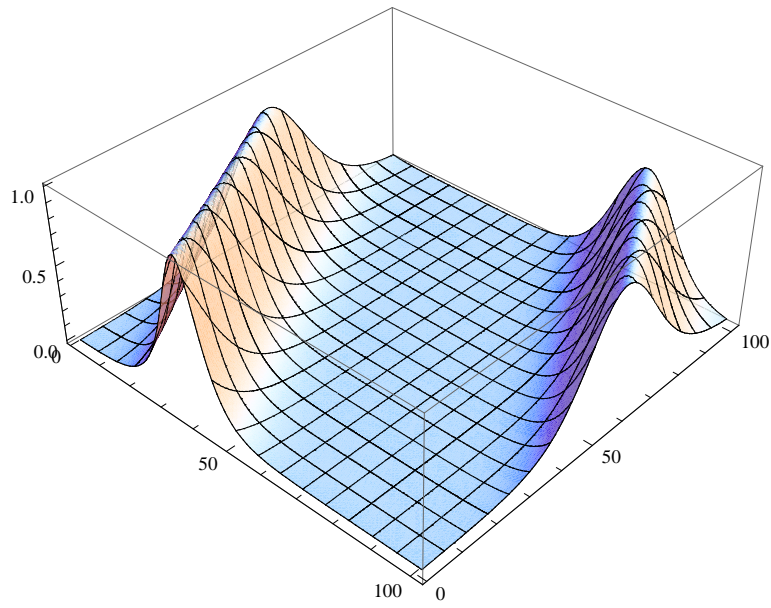
$$\bar{U}^{n+1} = B \cdot \bar{U}^n + \Delta t \bar{F}^n$$

where B is the matrix with $1 - \eta$ 'son the diagonal and η 'son the super diagonal and in the bottom left corner (because of the periodic boundary conditions i.e.

$$B = \begin{pmatrix} 1 - \eta & \eta & 0 & 0 & \cdots & 0 \\ 0 & 1 - \eta & \eta & 0 & \cdots & 0 \\ \vdots & & \ddots & \ddots & & \vdots \\ 0 & \cdots & 0 & 1 - \eta & \eta & 0 \\ 0 & \cdots & 0 & 0 & 1 - \eta & \eta \\ \eta & 0 & 0 & \cdots & 0 & 1 - \eta \end{pmatrix}$$

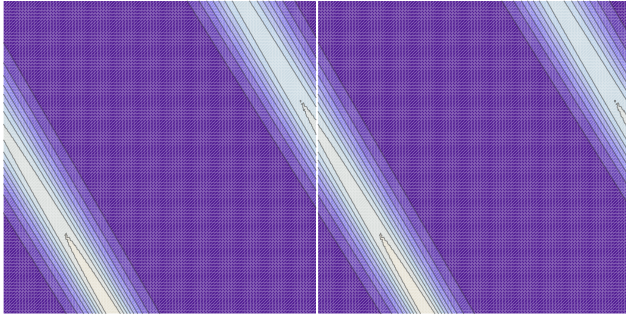
My exact solution (with $f = 0$) is $u[x, t] = \text{Sech}[3 - (x + \alpha t)]$.

```
{a, b} = {-10, 10};
TMax = 12;
{nx, nt} = {100, 100};
{Δx, Δt} = N[{(b - a) / nx, TMax / nt}];
α = 1;
η = α Δt / Δx;
B = SparseArray[
  Join[
    Table[{i, i} → 1 - η, {i, nx + 1}],
    Table[{i, i + 1} → η, {i, nx}],
    {{-1, 1} → η}
  ]
];
g[τ_] = Sec[3 + τ];
uEx[x_, t_] := g[x + α t];
U = Table[0, {nt + 1}, {nx + 1}];
Dimensions[U]
U[[1]] = Table[uEx[x, 0], {x, a, b, Δx}];
Do[
  U[[n]] = B.U[[n - 1]],
  {n, 2, nt + 1}
];
ListPlot3D[U,
  PlotRange → All
]
CPlot = ListContourPlot[U]
{101, 101}
```



The periodicity is pretty obvious if we put to contour plots side by side

```
GraphicsArray[{Show[CPlot, Frame → False], Show[CPlot, Frame → False]},
GraphicsSpacing → -0.05]
```

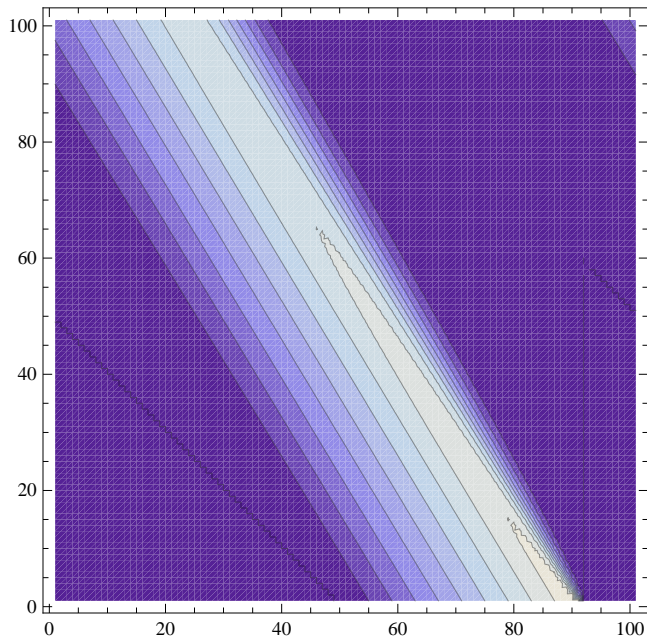
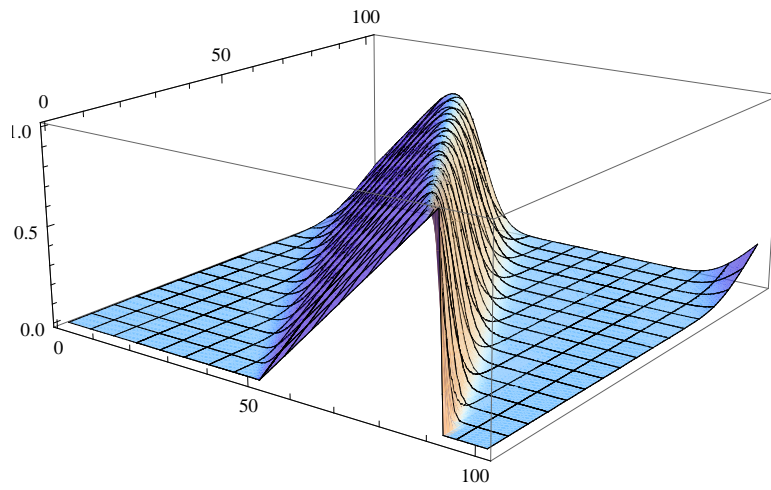


It flat out does not work if we do it with the other sign of α . A simple stability analysis explains this. Before we do that we need to get picky about the quality of our solutions. The peak value should be constant but it drifts downwards in the contour plot. The contour stripes should be constant width but they broaden. The stability analysis will explain this as well but first a harder example with a steeper initial condition!

```
{a, b} = {-10, 10};
TMax = 12;
{nx, nt} = {100, 100};
{Δx, Δt} = N[{(b - a) / nx, TMax / nt}];

α = 1;
η = α Δt / Δx;
B = SparseArray[
  Join[
    Table[{i, i} → 1 - η, {i, nx + 1}],
    Table[{i, i + 1} → η, {i, nx}],
    {{-1, 1} → η}
  ]
];
g[τ_] = Piecewise[{
  {0, τ ≤ 0},
  {τ / 8, 0 < τ ≤ 8},
  {0, τ > 8}
}];
uEx[x_, t_] := g[x + α t];
U = Table[0, {nt + 1}, {nx + 1}];
Dimensions[U]
U[[1]] = Table[uEx[x, 0], {x, a, b, Δx}];
Do[
  U[[n]] = B.U[[n - 1],
    {n, 2, nt + 1}
  ];
ListPlot3D[U,
  PlotRange → All]
CPlot = ListContourPlot[U]

{101, 101}
```



Same thing but now with the cliff facing the other way!

```

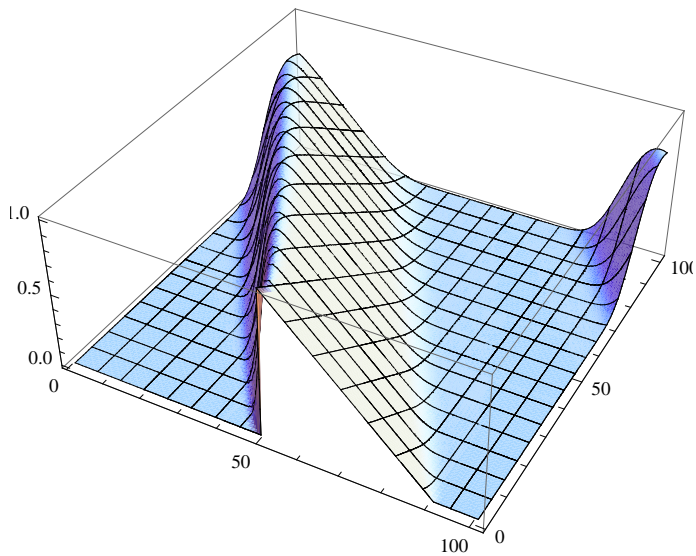
{a, b} = {-10, 10};
TMax = 12;
{nx, nt} = {100, 100};

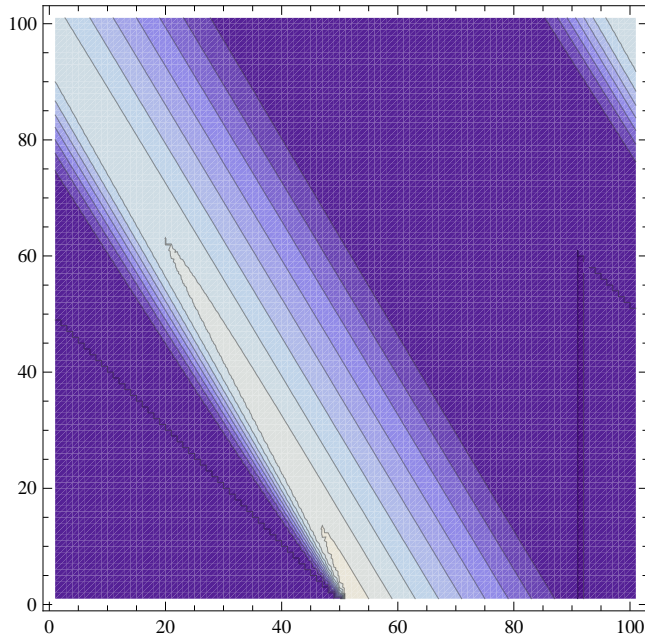
{Δx, Δt} = N[{(b - a) / nx, TMax / nt}];

α = 1;
η = α Δt / Δx;
B = SparseArray[
  Join[
    Table[{i, i} → 1 - η, {i, nx + 1}],
    Table[{i, i + 1} → η, {i, nx}],
    {{-1, 1} → η}
  ]
];
g[τ_] = Piecewise[{
  {0, τ ≤ 0},
  {1 - τ / 8, 0 < τ ≤ 8},
  {0, τ > 8}
}];
uEx[x_, t_] := g[x + α t];
U = Table[0, {nt + 1}, {nx + 1}];
Dimensions[U]
U[[1]] = Table[uEx[x, 0], {x, a, b, Δx}];
Do[
  U[[n]] = B.U[[n - 1]],
  {n, 2, nt + 1}
];
ListPlot3D[U,
  PlotRange → All]
CPlot = ListContourPlot[U]

{101, 101}

```





Same thing but now with a bigger time step. Does not work.

```

In[33]:= {a, b} = {-10, 10};
TMax = 12;
{nx, nt} = {100, 50};
{Δx, Δt} = N[{(b - a) / nx, TMax / nt}];
α = 1;
η = α Δt / Δx;
B = SparseArray[
  Join[
    Table[{i, i} → 1 - η, {i, nx + 1}],
    Table[{i, i + 1} → η, {i, nx}],
    {{-1, 1} → η}
  ]
];
g[τ_] = Piecewise[{
  {0, τ ≤ 0},
  {1 - τ / 8, 0 < τ ≤ 8},
  {0, τ > 8}
}];
uEx[x_, t_] := g[x + α t];
U = Table[0, {nt + 1}, {nx + 1}];
Dimensions[U]
U[[1]] = Table[uEx[x, 0], {x, a, b, Δx}];
Do[
  U[[n]] = B.U[[n - 1]],
  {n, 2, nt + 1}
];
ListPlot3D[U,
  PlotRange → All]
CPlot = ListContourPlot[U]

```

Out[43]= {51, 101}

