Partial differential equations

This chapter is an introduction to PDE with physical examples that allow straightforward numerical solution with *Mathematica*. Methods of solution of PDEs that require more analytical work may be will be considered in subsequent chapters.

General facts about PDE

Partial differential equations (PDE) are equations for functions of several variables that contain partial derivatives. Typical PDEs are Laplace equation

$$\Delta\phi[x, y, \ldots] == 0$$

(Δ is the Laplace operator), Poisson equation (Laplace equation with a source)

$$\Delta\phi[x, y, \ldots] == f[x, y, \ldots],$$

wave equation

$$\partial_t^2 \phi[t, x, y, \ldots] - c^2 \Delta\phi[t, x, y, \ldots] == 0,$$

heat conduction / diffusion equation

$$\partial_t \phi[t, x, y, \ldots] - \kappa\Delta\phi[t, x, y, \ldots] == 0,$$

Schrödinger equation

$$i\, \partial_t \phi[t, x, y, \ldots] + (a\Delta + b f[x, y, \ldots])\, \phi[t, x, y, \ldots] == 0,$$

etc. There are systems of PDEs and nonlinear PDEs.

Solution of PDEs have more freedom than those of ODEs because integration "constants" are in fact functions. For instance, the general solution of the second-order PDE

$$\partial_{x,y} f[x, y] == 0$$

is

$$f[x, y] = F[x] + G[y],$$

where $F[x]$ and $G[y]$ are arbitrary functions. The solution of the first-order PDE

$$\partial_t f[t, x] - v\, \partial_x f[t, x] == 0$$

is

$$f[t, x] = g[x - vt]$$

that describes a front of arbitrary shape moving in the positive direction if $v > 0$.

General analytical solutions of PDEs are available only in the simplest cases, and because of this freedom, they do not yet solve the problem. The actual form of the solution is defined by the symmetry of the problem (if it exists) and boundary conditions. If one of the variables is time, one usually speaks of initial conditions set at the initial time and of the boundary conditions for spatial variables.

If there are initial conditions but no final conditions, the problem is evolutionary and it can be solved numerically starting from the initial conditions and increasing time. The most efficient method to do it is the so-called "method of lines" used by *Mathematica*. First, the problem is discretized in spacial variables and spatial derivatives are approximated by differences. This reduces the PDE to a system of ODEs in time. Then the resulting system of ODEs is solved by one of high-performance ODE solvers. In *Mathematica*, PDEs, as well as ODEs, are solved by NDSolve.

However, currently *Mathematica* can only solve problems with a rectangular spatial region. For a typical PDE, in these cases one also can find an analytical solution.

Mathematically, there is no difference between the time and other variables. If for a particular spacial variable a boundary condition is set only at one end of the interval, one can treat this variable as time and the problem is evolutionary. *Mathematica* recognizes this situation and finds the solution. If boundary conditions are set at all ends of the interval (or infinity) NDSolve does not find the solution and other methods have to be used. Most time-independent problems are like that.

Partial differential equations in physics

In physics, PDEs describe continua such as fluids, elastic solids, temperature and concentration distributions, electromagnetic fields, and quantum-mechanical probabilities. Below we review these equations.

Continuity equation

Any conserved quantity described by the density ρ and the corresponding current $\mathbf{j}$ satisfies the continuity equation

$$\frac{\partial \rho}{\partial t} + \mathrm{div}\mathbf{j} == 0. \tag{1}$$

This is the continuity equation in the differential form. If dV is an infinitesimal volume, $\partial_t \rho d$V is the rate of change of the amount (corresponding to ρ) within this volume, whereas div$\mathbf{j}$ dV is the flux out of this volume. If the flux is positive, the amount decreases. The integral form of the continuity equation can be obtained by integrating over a finite volume V

$$\frac{\partial}{\partial t}\int \rho\, dV == -\int \mathrm{div}\mathbf{j}\, dV.$$

At the right, the integral over the volume can be expressed via a surface integral, the flux of $\mathbf{j}$, according to the Gauss theorem

$$\frac{\partial}{\partial t}\int \rho\, dV == -\int \mathbf{j}\cdot d\mathbf{S}.$$

As there are many types of densities in physics (mass density, energy density, charge density, etc.) there are also many types of continuity equations.

Fluid dynamics

Motion of a fluid satisfies the continuity equation (1), there ρ is the mass density and the mass current $\mathbf{j}$ density is given by

$$\mathbf{j} = \rho \mathbf{v}.$$

Liquids, unlike gases, are practically uncompressible, ρ = const, thus the continuity equation simplifies to

$$\mathrm{div}\mathbf{v} == 0.$$

Dynamics of a non-viscous fluid is described by the Euler equation

$$\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v}\,\nabla)\,\mathbf{v} == -\frac{\nabla P}{\rho} + \mathbf{g},$$

where P is pressure and $\mathbf{g}$ is the gravity acceleration. Minus in front of the pressure gradient shows that the liquid accelerates in the direction of a smaller pressure. The kinematic (streaming) term $(\mathbf{v}\,\nabla)\,\mathbf{v}$ accounts for the change of the velocity at a given point as a result of the media motion and makes the Euler equation nonlinear.

Flow of a viscous fluid obeys the Navier-Stokes equation. For an incompressible fluid it has the form

$$\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v}\,\nabla)\,\mathbf{v} == -\frac{\nabla P}{\rho} + \mathbf{g} + \frac{\eta}{\rho}\,\Delta\mathbf{v},$$

where η is the viscosity coefficient.

Euler equation has a limited applicability because its solutions (for instance, for an object in a flow) have discontinuities of the velocity in the stream. In these regions the viscosity becomes important and at high enough speeds the flow becomes turbulent. Still, Euler equation generates Bernoilli equation that explains the lifting force on airplanes as well as the mechanism of sound creation by voice bands.

Navier-Stokes equation is more realistic and it can be applied, for instance, to the problem of a drag force acting on a object moving in a liquid. For small boby speeds u the quadratic terms can be dropped, equation becomes linear and can be solved analytically for simple shapes such as sphere. The drag forces on a solid sphere and air bubble of radius R have the form

$$F = \begin{cases} 6\,\pi\eta R u, & \text{solid sphere} \\ 4\,\pi\eta R u, & \text{air bubble.} \end{cases}$$

The difference in the coefficient is due to different bounary conditions: The liquid sticks to the solid sphere but glides along the surface of the air bubble that reduces the resistance.

■ Waves

Waves exist in compressible media such as fluids and solids. The main prerequisite for the waves' existence is a restoring force. In fluids, there is only a restoring force with respect to change of the volume, thus the waves are waves of compression / expansion. In these waves the media displacement is collinear with the direction of the wave propagation, so that these waves are called longitudinal waves. In solids, there is additionally a restoring force with respect to change of the shape of a media element. The corresponding additional waves are transverse waves. Below we will consider small-amplitude longitudinal waves in fluids.

In small-amplitude waves pressure and density only slightly deviate from their equilibrium values, so that one can define

$$P = P_0 + \delta P, \qquad \rho = \rho_0 + \delta\rho$$

and linearize equations in small δP and $\delta\rho$, neglecting quadratic velocity terms because the velocity is small. The continuity equation becomes

$$\frac{\partial}{\partial t}\delta\rho + \rho_0\,\mathrm{div}\mathbf{v} == 0, \qquad (2)$$

whereas in the absence of viscosity from the Euler equation one obtains

$$\frac{\partial \mathbf{v}}{\partial t} == -\frac{\nabla\delta P}{\rho_0}. \qquad (3)$$

The small changes of pressure and density are related by the adiabatic compressibility,

$$\delta P = \left(\frac{\partial P}{\partial \rho}\right)_S \delta\rho.$$

One can obtain a single equation describing waves by differentiating the continuity equation over time and then using the Euler equation. This yields the wave equation

$$\partial_t^2\,\delta\rho - c^2\,\Delta\delta\rho, \qquad c \equiv \sqrt{\left(\frac{\partial P}{\partial \rho}\right)_S},$$

where c is the speed of sound. Note that the speed of sound (that can be large) has no relation to the velocity of the media (that is small). Since $\delta P \sim \delta\rho$, it satisfies the same equation,

$$\partial_t^2 \delta P - c^2 \Delta \delta P. \tag{4}$$

Let us obtain the equation for the velocity. Taking rotor of the linearized Euler equation, one obtains

$$\frac{\partial}{\partial t} \nabla \times \mathbf{v} == 0,$$

thus velocity is a potential field, $\nabla \times \mathbf{v} = 0$, and it can be searched in the form

$$\mathbf{v} = -\nabla \phi.$$

Inserting it into the linearized Euler equation (3) and integrating it by removing the gradient one obtains

$$\frac{\partial \phi}{\partial t} == \frac{\delta P}{\rho_0} = \frac{c^2}{\rho_0} \delta \rho. \tag{5}$$

Differentiating this over time and using the continuity equation, one obtains the same wave equation for the velocity potential,

$$\partial_t^2 \phi - c^2 \Delta \phi.$$

Let us now seek for the solution of the wave equation in the form of a plane wave

$$\phi = A\mathrm{Cos}[\omega t - \mathbf{k} \cdot \mathbf{r} + \phi_0],$$

where A is the amplitude, ω is the frequency and $\mathbf{k}$ is the wave vector that defines the direction of the wave propagation. After substitution, differentiation, and cancellation one obtains the dispersion relation

$$\omega^2 - c^2 k^2 == 0.$$

The period T and wave length λ are related to ω and k by

$$\omega = \frac{2\pi}{T}, \qquad k = \frac{2\pi}{\lambda}.$$

One can see that ω can be different while k adjusts to ω as $ck = \omega$.

Let us now find the media velocity in a plane wave:

$$\mathbf{v} = -\nabla \phi = -A\,\mathbf{k}\,\mathrm{Sin}[\omega t - \mathbf{k} \cdot \mathbf{r} + \phi_0].$$

One can see that $\mathbf{v}$ is collinear to $\mathbf{k}$, this is why such waves are called longitudinal waves.

Since the wave equation is linear, a superposition of plane waves moving in different directions with different frequencies and wave lengths is also its solution. This illustrates the fact that there is much freedom in the solution of PDEs. The selection of the actual solution is done by the initial and boundary conditions.

In the important case of monochromatic waves whose time dependence is describe by a single frequency,

$$\phi[\mathbf{r}, t] = \phi[\mathbf{r}]\,\mathrm{Sin}[\omega t],$$

or similar with Sin[ωt] or Exp[$i\omega$t], the wave equation simplifies to the time-independent stationary wave equation

$$\Delta \phi + k^2 \phi == 0, \qquad k = \frac{\omega}{c}.$$

In one dimension it becomes an ODE.

■ Thermal conductivity and diffusion

Temperature in an isolated body tends to equilibrate with time because the energy flows from warmer regions to colder regions. Since the energy is conserved, its flow obeys the continuity equation

$$\frac{\partial e}{\partial t} + \mathrm{div}\, \mathbf{q} == 0,$$

where e is the density of the internal energy and $\mathbf{q} \equiv \mathbf{j}_e$ is its current. If the body is incompressible, the change of the internal energy according to thermodynamics is due to the heat received, $de = \delta Q$. On the other hand, the heat received is related to the temperature change via heat capacity c_V (per unit volume), so that $de = c_V\, dT$. On the other hand, experimentally it is known that the energy (heat) current density is proportional to the temperature gradient. For isotropic materials this relation has the simplest form

$$\mathbf{q} = -k\, \nabla T,$$

where k is thermal conductivity depending on the material. Putting all together and neglecting the temperature dependence of k (that is justified if changes ot the temperature are small) one obtains the thermal diffusion (or heat conduction) equation

$$\frac{\partial T}{\partial t} == \kappa \Delta T, \quad \kappa \equiv \frac{k}{c_V},$$

where κ is thermal diffusivity having the unit m^2/s. In the stationary case this equation becomes a Laplace equation.

This equation is valid in the absence of motion, otherwise one has to add the convective term $\mathbf{v}\cdot\nabla T$ on the left, and then the equation for the temperature becomes coupled to the equation for the fluid dynamics above.

If the thermal energy is injected into the system (e.g., as a result of burning), one has to add a corresponding source term on the right side of the equation .

Physics of diffusion is similar, and the concentration of particles c satisfies the equation

$$\frac{\partial c}{\partial t} == \kappa \Delta c,$$

where κ is diffusivity. If particles can be carried by the media motion, the convective term $\mathbf{v}\cdot\nabla c$, similarly to the heat conduction equation. The corresponding generalization is the Smoluchowski equation.

- **Heat flow between two walls**

As an illistration, consider the region between the y-z planes (walls) at $x = 0$ and $x = a > 0$. The temperature at the left boundary is T_0 while the temperature at the right boundary is T_1. If one is interested in the stationary solution of the thermal diffusion equation, one can drop the time derivative, search for the solution in the form $T = T[x]$ and obtain the ordinary differential equation

$$\frac{d^2 T}{dx^2} == 0.$$

The solution of this equation is a linear function of x

$$T = C_1 x + C_2 = T_0 + (T_1 - T_0)\, \frac{x}{a},$$

where the constants have been eliminated with the help of the boundary conditions. This solution corresponds to the heat flux density

$$q = -k\, \frac{dT}{dx} = -k\, \frac{T_1 - T_0}{a}$$

flowing from the hot wall to the cold wall. With these boundary conditions, the stationary solution found above will be achieved from any initial condition after a sufficiently long relaxation time.

One could reformulate this problem setting the heat flux density q entering the system, say, from the left and one of the temperatures, say, on the right. In this case the boundary conditions are

$$\frac{dT}{dx}[0] == -\frac{q}{k}, \qquad T[a] = T_1.$$

The resulting solution has the form

$$T = T_1 + \frac{q}{k}(a - x).$$

- **Heat flow from a heated sphere**

Another example of a stationary heat flow is a sphere of radius R at temperature T_0 immersed in a heat-conductive media having the temperature T_∞ at infinity. Here the symmetry allows to bypass the complete solution of the heat conduction equation by using the energy conservation argument saying that in the stationary case the total outwards heat flux through any spherical surface of radius r is the same

$$Q = 4\pi r^2 q = \text{const}$$

Thus one finds

$$q = \frac{C}{r^2},$$

similarly to the electric field of a point charge found from the Gauss theorem. Now the temperature can be found integrating the equation

$$\frac{dT}{dr} = -\frac{q}{k} = \frac{C}{kr^2},$$

that yields

$$T[r] = C_1 + \frac{C_2}{r}.$$

Finding the constants from the boundary conditions one finally obtains

$$T[r] = T_\infty + (T_0 - T_\infty)\frac{R}{r},$$

similarly to the electric potential of a point charge. This solution corresponds to the total heat flux

$$Q = -4\pi r^2 k \frac{dT}{dr} = 4\pi kR(T_0 - T_\infty).$$

One can see that the smaller is the body, the smaller energy input is needed to support it at a temperature exceeding the temperature of the environment (a tip for building houses, to reduce heating bills).

Maxwell equations

Electromagnetic field satisfies the system of Maxwell equations. These equations become more complicated if the media strongly interacts with the electromagnetic field by being segnetoelectric (electrically polarized), ferromagnetic, or superconducting. For simplicity we will consider electromagnetic field in vacuum.

There are four Maxwell equations that describe generation of two different parts of the electric field and of the magnetic field by electric charges, electric currents, and temporal change of the electromagnetic field. In general, any vector field **F**[**r**] can be represented as the sum of the potential part and solenoidal part,

$$\mathbf{F}[\mathbf{r}] = -\nabla\phi + \nabla\times\mathbf{A},$$

where ϕ and $\mathbf{A}$ are scalar and vector potentials. The potential part has zero curl, $\nabla \times \nabla \phi = 0$, while the solenoidal part has zero divergence, $\nabla \cdot \nabla \times \mathbf{A} = 0$. The first Maxwell equation describes creation of the potential part of the electric field $\mathbf{E}$ by electric charges,

$$\nabla \cdot \mathbf{E} == \frac{\rho}{\epsilon_0},$$

where ρ is the electric charge density and ϵ_0 is related to units. The solenoidal part of $\mathbf{E}$ is created by the magnetic field $\mathbf{B}$ changing in time,

$$\nabla \times \mathbf{E} == -\frac{\partial \mathbf{B}}{\partial t}. \tag{6}$$

The next Maxwell equation states that there is no potential part in $\mathbf{B}$,

$$\nabla \cdot \mathbf{B} == 0,$$

because there are no magnetic charges. The last Maxwell equation describes creation of the solenoidal part of $\mathbf{B}$ by the electric current and electric field $\mathbf{E}$ changing with time,

$$\nabla \times \mathbf{B} == \mu_0\, \mathbf{j} + \mu_0\, \epsilon_0 \frac{\partial \mathbf{E}}{\partial t},$$

where $\mathbf{j}$ is the electric current density and μ_0 is related to units.

Representing the magnetic field in the form $\mathbf{B} = -\nabla \psi + \nabla \times \mathbf{A}$, from the third Maxwell equation one obtains $\Delta \psi = 0$. In the absence of the source in this equation, its solution is zero, $\psi = 0$, as said above. So that one finally has

$$\mathbf{B} = \nabla \times \mathbf{A}. \tag{7}$$

Substituting this into the last Maxwell equation, one obtains the equation for the vector potential $\mathbf{A}$

$$\nabla \times \nabla \times \mathbf{A} = \nabla (\nabla \cdot \mathbf{A}) - \Delta \mathbf{A} == \mu_0\, \mathbf{j} + \mu_0\, \epsilon_0 \frac{\partial \mathbf{E}}{\partial t}. \tag{8}$$

As there is a freedom in the definition of $\mathbf{A}$, one can additionally require $\nabla \cdot \mathbf{A} = 0$ (the so-called calibration). After that only the vector Laplacian remains on the left side of this equation.

Let us now consider equations for the electric field. The second equation becomes

$$\nabla \times \mathbf{E} == -\frac{\partial}{\partial t} \nabla \times \mathbf{A}$$

and can be integrated (by removing $\nabla\times$) to give

$$\mathbf{E} == -\nabla \phi - \frac{\partial \mathbf{A}}{\partial t}. \tag{9}$$

Here the first term arises as an integration "constant" and is the potential part of the electric field. Now substituting this into the first Maxwell equation and using $\nabla \cdot A = 0$, one obtains the Poisson equation for the electric potential

$$\Delta \phi == -\frac{\rho}{\epsilon_0}. \tag{10}$$

Substituting Eq. (9) into Eq. (8) to eliminate $\mathbf{E}$, one obtains the closed equation for the vector potential

$$\frac{1}{c^2} \frac{\partial^2 \mathbf{A}}{\partial t^2} - \Delta \mathbf{A} == \mu_0\, \mathbf{j} - \frac{1}{c^2} \frac{\partial \nabla \phi}{\partial t}, \qquad c = \frac{1}{\sqrt{\mu_0\, \epsilon_0}}. \tag{11}$$

In the static case time derivatives vanishes and this equation describes generation of the vector potential **A** and thus of the magnetic field **B** by the electric current. In general, this is a wave equation for electromagnetic waves with c being the speed of light. There are two sources on the right side of this equation that excite electromagnetic waves. One source is electric current and another is time-dependent potential part of the electirc field. The latter is created by moving electric charges according to Eq. (10).

As above, one can seek for the solution of this wave equation in the form of a plane wave. It turns out that in this plane wave **E** and **B** are perpendicular to each other and to the wave vector k. This means that electromagnetic waves are transverse waves (Excercise).

Since the electric current is due to the motion of electric changes in space, one can write the continuity equation (1) with the electric current density $\mathbf{j} = \rho \mathbf{v}$.

■ Schrödinger equation

In quantum mechanics, dynamics of an isolated system is described by the Schrödinger equation for the complex wave function $\Psi[\mathbf{r}, t]$ that has the form

$$i\hbar \frac{\partial \Psi}{\partial t} == \hat{H}\,\Psi, \tag{12}$$

where $\hbar$ is the Plank constant and $\hat{H}$ is the so-called Hamilton operator, the energy of the system expressed via positions and momenta, whereas the momenta are replaced by first-order spatial differential operators. For instance, for a particle with a potential energy $U[\mathbf{r}]$ the Hamiltonian function (the energy) is given by

$$\mathcal{H} = \frac{\mathbf{p}^2}{2\,m} + U[\mathbf{r}]\,.$$

Replacing **p** by the quantum-mechanical operator

$$\hat{\mathbf{p}} = -i\hbar\,\nabla,$$

one obtains the Hamilton operator

$$\hat{H} = -\frac{\hbar^2\,\Delta}{2\,m} + U[\mathbf{r}]$$

that acts on the wave function in the Schrödinger equation. This equation resembles the heat conduction and diffusion equations, only it has an imaginary factor at the time derivative. In spite of this similarity, there is little in common between the two equations.Whereas the heat conduction equation is relaxational, Schrödinger equation describes a conservative dynamics.

The interpretation of the wave function is that $|\Psi[\mathbf{r},\ t]|^2 \equiv \Psi[\mathbf{r},\ t]\,\Psi[\mathbf{r},\ t]^*$ is the probability density for a particle to be at the position r. Practically this is not of the primary importance, however. What mostly matters in quantum mechanics is stationary states that depend of time monochromatically,

$$\Psi[\mathbf{r}, t] = \psi[\mathbf{r}]\,\mathrm{Exp}\left[\frac{i\mathrm{E}}{\hbar}\,t\right],$$

E being the energy of the particle, and satisfy the stationary Schrödinger equation

$$\hat{H}\,\psi == \mathrm{E}\psi. \tag{13}$$

These states corresponds to quantum energy levels (next chapter). In fact, Schrödinger equation has more in common with the wave equation and it also has a solution in the form of a plane wave if $U[r] = U = \mathrm{const}$

$$\Psi[\mathbf{r},\ t] = A\,\mathrm{Exp}[-i\,(\omega t - \mathbf{k}\cdot\mathbf{r})]\,. \tag{14}$$

Substituting this into Schrödinger equation one obtains the relation

$$\hbar\omega = \frac{\hbar^2 k^2}{2 m} + U \qquad (15)$$

between the frequency ω and the wave vector k. Frequency ω is related to the energy E and the wave vector to the de Broglie wave length as

$$\hbar\omega = E, \quad k = 2\pi/\lambda. \qquad (16)$$

Solving Eq. (15) for k one obtains

$$k = \sqrt{2 m (E - U)} \Big/ \hbar. \qquad (17)$$

$E > U$ is the regular case in which the kinetic energy $E - U$ is positive and k is real. The case $E < U$ corresponds to the negative kinetic energy and describes quantum motion in the regions where classical motion is impossible (e.g., under a potential barrier). In this case k is imaginary,

$$k = \pm i\kappa, \qquad \kappa = \sqrt{2 m (U - E)} \Big/ \hbar, \qquad (18)$$

and the plane wave is an exponential imaginary in time but real in space:

$$\Psi[\mathbf{r}, t] = A \operatorname{Exp}[-i\omega t + \boldsymbol{\kappa} \cdot \mathbf{r}]. \qquad (19)$$

There are a lot of different forms of the Schrödinger equation. For instance, in the case of rotational motion the angular momentum becomes a differential operator. For a system of interacting particles the wave function depends on all positions and there are Laplacians for each particle. This multidimensional PDE cannot be solved directly without drastic approximations, analytically and numerically.

Quantum mechanics of relativistic particles is described by Dirac equation that contains Schrödinger equation as its non-relativistic limit.

Numerical solution of PDE

■ Heat conduction and Laplace equations

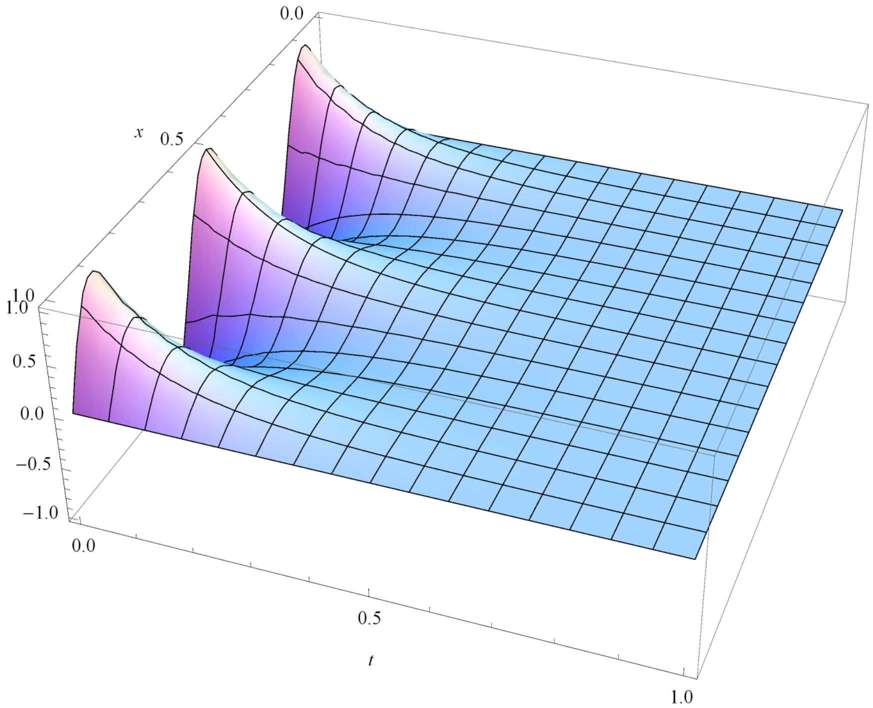
■ 1d: Relaxation of temperature from an initial state

It is important that the initial condition does not contradict the boundary conditions, otherwise some conditions will be discarded.

```
κ = 0.03;   L = 1;   tMax = 1;
Equation = ∂_t T[x, t] - κ ∂_{x,x} T[x, t] == 0
BConds = {T[0, t] == 0, T[L, t] == 0};
ICond = T[x, 0] == Sin[5 π x / L];

Sol = NDSolve[Join[{Equation}, BConds, {ICond}], T, {x, 0, L}, {t, 0, tMax}]
Txt[x_, t_] := T[x, t] /. Sol[[1]]
```

```
T^(0,1)[x, t] - 0.03 T^(2,0)[x, t] == 0

{{T → InterpolatingFunction[{{0., 1.}, {0., 1.}}, <>]}}
```

```
Plot3D[Txt[x, t], {x, 0, L}, {t, 0, tMax}, PlotRange → All, AxesLabel → Automatic]
```



The greater the thermal diffusivity, the faster the temperature approaches its equilibrium value.

The fact that the temperature relaxes to a time-independent stationary state may be used to solve the Laplace equation and similar non-evolutionary equations by the method of relaxation.

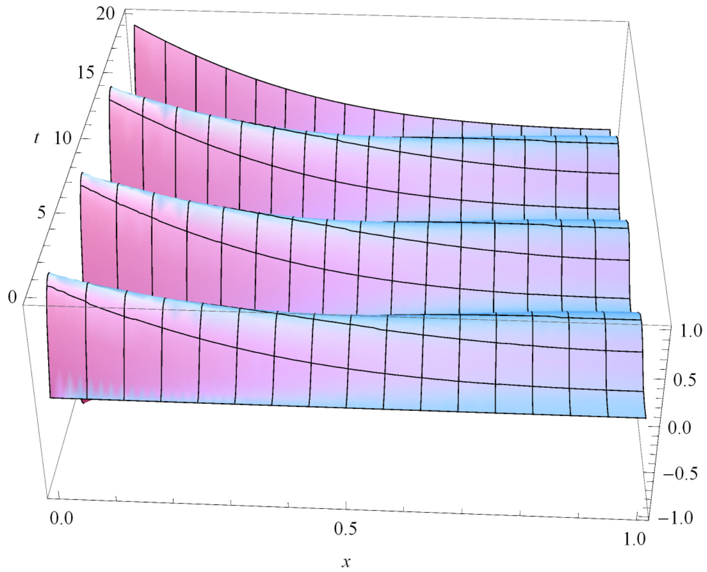
- **1d: Temperature driven by a time-dependent boundary condition**

On the left side the temperature is oscillating in time and on the right side the system is thermally insulated

```
κ = 0.1;   L = 1;   tMax = 20;   ω = 1;
Equation = ∂t T[x, t] - κ ∂x,x T[x, t] == 0
BConds = {T[0, t] == Sin[ω t], (∂x T[x, t] /. x → L) == 0}
ICond = T[x, 0] == 0;

Sol = NDSolve[Join[{Equation}, BConds, {ICond}], T, {x, 0, L}, {t, 0, tMax}]
Txt[x_, t_] := T[x, t] /. Sol[[1]]
```

```
T^(0,1)[x, t] - 0.1 T^(2,0)[x, t] == 0

{T[0, t] == Sin[t], T^(1,0)[1, t] == 0}

{{T → InterpolatingFunction[{{0., 1.}, {0., 20.}}, <>]}}
```

```
Plot3D[Txt[x, t], {x, 0, L}, {t, 0, tMax},
 PlotRange → All, AxesLabel → Automatic, PlotPoints → 30]
```



The greater the thermal diffusivity, the further to the right goes the disturbance.

- **2d: Solution of the Laplace equation by the relaxation method**

The problem is to solve the Laplace equation

$$\Delta T == 0$$

in the rectangular 2d region $0 \le x \le L_x$, $0 \le y \le L_y$ with the boundary conditions $T[0, y] = T_0$, $T[L_x, y] = T_1$, and $T[x, 0] = T[x, L_y] = 0$. The function T can be temperature or concentration, or electric potential, or anything else that is described by the Laplace equation. The idea of the relaxation method is to solve the equation

$$\frac{\partial T}{\partial t} == \kappa \Delta T$$

with some initial condition. After some time the system will come to the time independent state in which $\partial_t T = 0$ and thus $\Delta T = 0$. Setting the initial condition, one has to keep in mind that it has to be compatible with the boundary conditions, otherwise *Mathematica* might be unable to solve the problem. One of such initial conditions is

$$T[x, y, 0] == T_0 \frac{(L_x - x)\, y\, \left(L_y - y\right)}{x + (L_x - x)\, y\, \left(L_y - y\right)} + T_1 \frac{x\, y\, \left(L_y - y\right)}{(L_x - x) + x\, y\, \left(L_y - y\right)}$$

The parameters κ and tMax can be determined experimentally. In fact, one can set $\kappa = 1$ and choose an appropriate tMax because these two parameters in fact enter as the product κtMax.

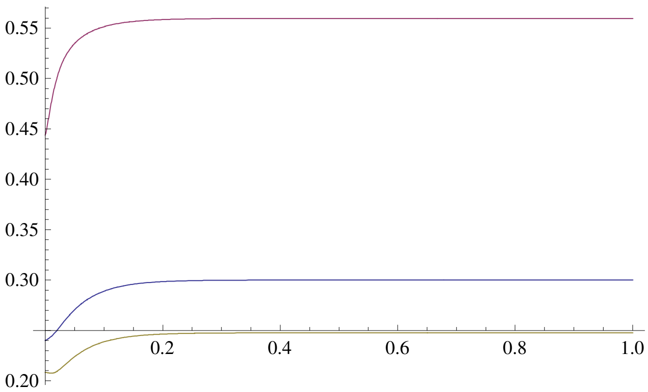
The number of spatial discretization grid points can be controlled by the MaxSteps option. As the initial condition is singular, *Mathematica* uses by default 100 grid points. However, it complains anyway: Because of the singularity, any number of grid points is formally insufficient (although practically 100 grid points is OK).

Apart of this, the initial condition contains division by zero and works only with an added small number in the denominator. After that initial and boudary conditions contradict each other. However, this does not really lead to a problem because the inconsistency is very small. The resulting solution is good.

```
κ = 1;   Lx = 1; Ly = 1;  tMax = 1;  T0 = 1; T1 = 0.2;
Equation = ∂t T[x, y, t] - κ (∂x,x T[x, y, t] + ∂y,y T[x, y, t]) == 0;
BConds = {  T[0, y, t] == T0,
            T[Lx, y, t] == T0,
            T[x, 0, t] == 0,
            T[x, Ly, t] == 0    };
ICond = T[x, y, 0] == T0 (Lx - x) y (Ly - y) / (x + (Lx - x) y (Ly - y) + 10^-15 (* avoid division by 0 *)) +
     T1 x y (Ly - y) / ((Lx - x) + x y (Ly - y) + 10^-15 (* avoid division by 0 *));

Sol = NDSolve[Join[{Equation}, BConds, {ICond}], T,
   {x, 0, Lx}, {y, 0, Ly}, {t, 0, tMax}, MaxSteps → {110, 110, Infinity}];
Txyt[x_, y_, t_] := T[x, y, t] /. Sol[[1]]
```

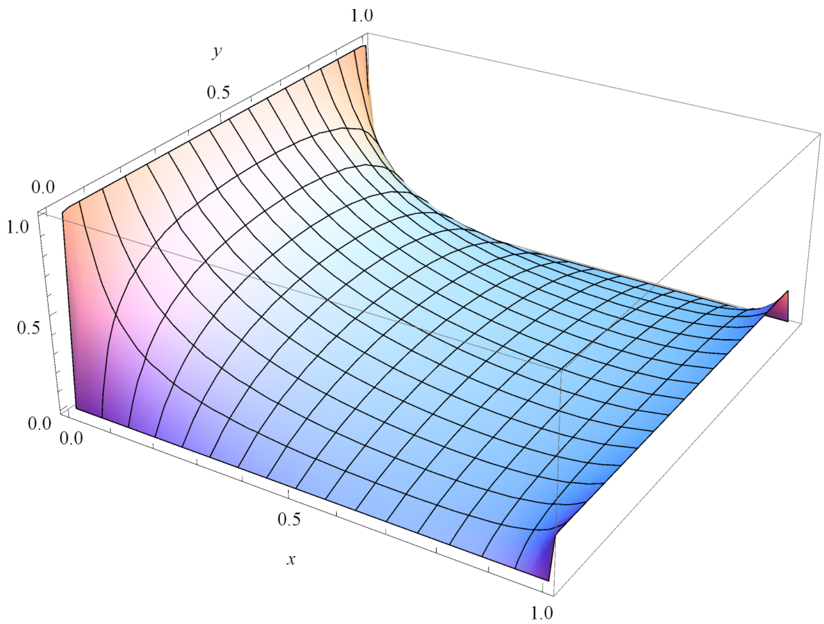
Now check that tMax is sufficient for relaxation by plotting T at some points vs. t.

```
Plot[{Txyt[0.5, 0.5, t], Txyt[0.25, 0.5, t], Txyt[0.5, 0.3, t]},
 {t, 0, tMax}, PlotRange → All]
```



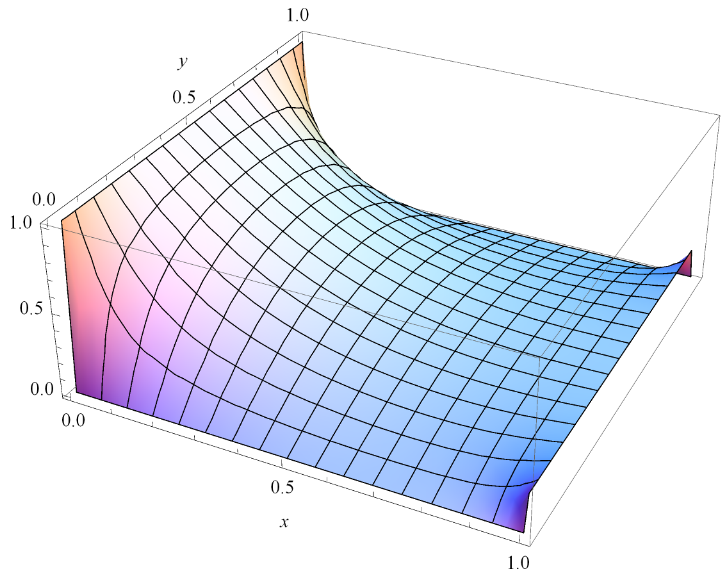
The relaxation is complete. One can see that the initial condition was a good approximation because the changes are not large. Now plot the initial condition

```
tt = 0;
Plot3D[Txyt[x, y, tt], {x, 0, Lx}, {y, 0, Ly}, PlotRange → All, AxesLabel → Automatic]
```



Now plot the final solution

```
tt = tMax;
Plot3D[Txyt[x, y, tt], {x, 0, Lx}, {y, 0, Ly}, PlotRange → All, AxesLabel → Automatic]
```



One can see that the final solution does not differ much from the initial condition.

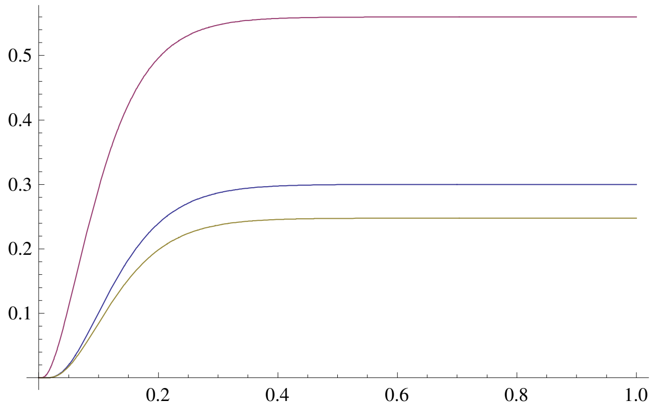
There is a more elegant version of the method of relaxation that avoids the thorny problem of constructing an initial condition that satisfies the boundary conditions. The idea is to start with trivial boundary and initial conditions and then transform the boundary conditions into true ones, as is implemented in the code below.

There is a problem, however, that *Mathematica* does not detect from the initial and bondary conditions that there is a singularity in the problem and chooses a grid with a too small number of points. To correct it, one has to find right options in the *Mathematica* help. As a result, *Mathematica* complains because of the singularity but the solution is good.

```
κ = 1;   Lx = 1; Ly = 1;  tMax = 1;  T0 = 1;  T1 = 0.2;
Equation = ∂t T[x, y, t] - κ (∂x,x T[x, y, t] + ∂y,y T[x, y, t]) == 0;
BConds = {  T[0, y, t] == T0 Tanh[10 t],
            T[Lx, y, t] == T1 Tanh[10 t],
            T[x, 0, t] == 0,
            T[x, Ly, t] == 0    };
ICond = T[x, y, 0] == 0;

Sol = NDSolve[Join[{Equation}, BConds, {ICond}], T,
   {x, 0, Lx}, {y, 0, Ly}, {t, 0, tMax}, Method → {"MethodOfLines",
     "SpatialDiscretization" → {"TensorProductGrid", "MinPoints" → 100}}];
Txyt[x_, y_, t_] := T[x, y, t] /. Sol[[1]]
```

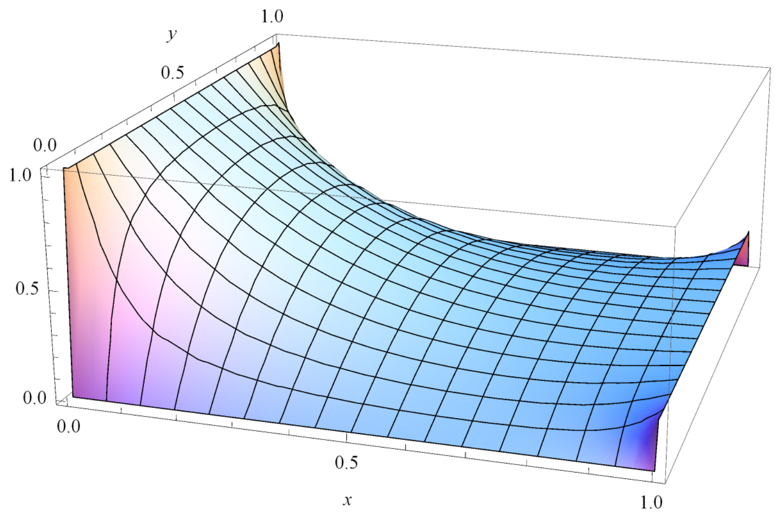
There are less complaints than in the main version of the relaxation method. Now check that tMax is sufficient for relaxation by plotting T at some points vs. t.

```
Plot[{Txyt[0.5, 0.5, t], Txyt[0.25, 0.5, t], Txyt[0.5, 0.3, t]},
 {t, 0, tMax}, PlotRange → All]
```



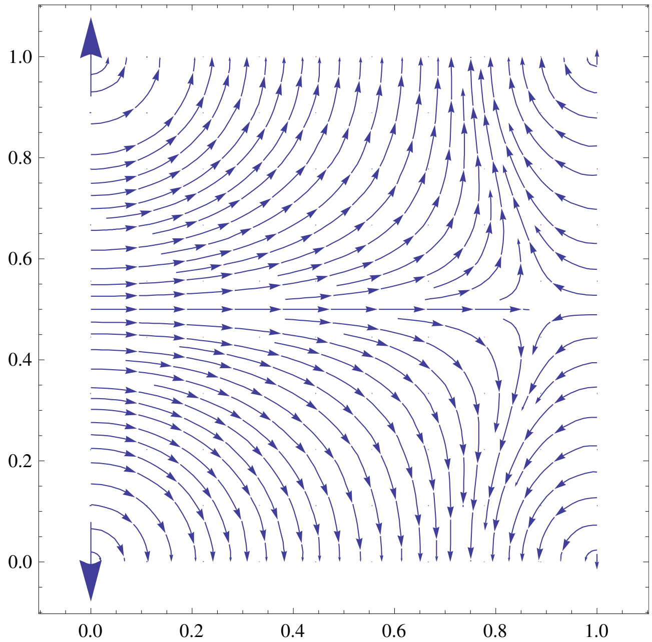
Relaxation is complete.

```
tt = tMax;
Plot3D[Txyt[x, y, tt], {x, 0, Lx}, {y, 0, Ly}, PlotRange → All, AxesLabel → Automatic]
```

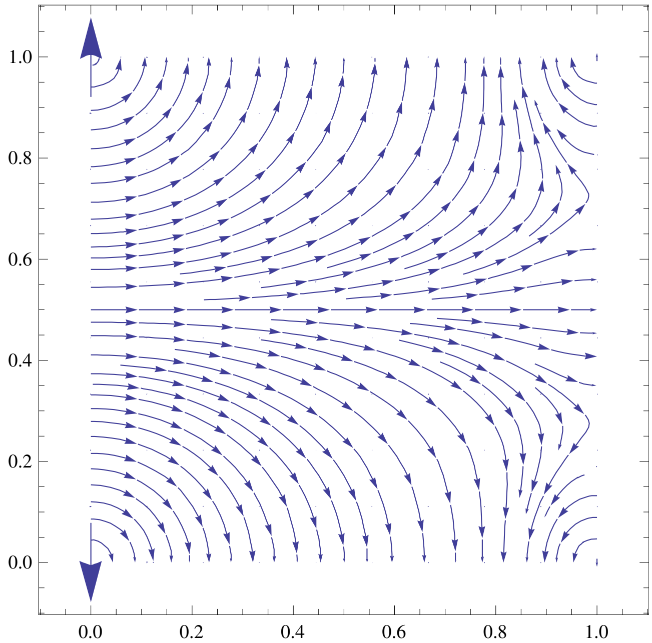


Let us now plot the heat flux (or the electric field in the case of the electric potential)

```
Grad[f_] := {∂x f, ∂y f}
Q[x_, y_] = -Grad[Txyt[x, y, tMax]]; (* Heat flux *)
StreamPlot[{Q[x, y][[1]], Q[x, y][[2]]}, {x, 0, Lx}, {y, 0, Ly}, VectorPoints → 10]
```



The two heat fluxes collide but the flux from the left is stronger (`T0 = 1;  T1 = 0.2`). If we reduce the temperature of the right wall to $T_1 = 0.1$, the flux from the left wins totally in the middle, whereas there is still a peripheric flux from the right.



- **1d: Deflagration (burning)**

Above we have considered the thermal runaway (or explosive instability) by assuming that the process is uniform in space. In reality it is not completely true because the heat exchange with the environment is not uniform and occurs at the boundary of the region with explosives. Thus the thermal instability process should begin in the center and then spread to the periphery. This spreading can happen in the form of propagating fronts of thermal runaway. The usual slow burning fronts that we observe igniting a sheet of paper is also described by this mechanism, it does not have to be a violent process such as explosion. In physics burning that occurs via moving fronts is called deflagration.

Deflagration consists of two processes: (i) Decay of a flammable substance (fuel, explosive) with a rate strongly increasing with temperature and (ii) Spreading of the released heat from hot (burned) to cold (yet unburned) areas via heat conduction. The mathematical model of deflagration uses two equations: (i) Rate (or relaxational) equation for the mass of fuel *m* and (ii) heat conduction equation with a source

$$\frac{dm}{dt} == -\Gamma m, \qquad \Gamma[T] = \Gamma_0 \, \mathrm{Exp}\left[-\frac{\Delta U}{k_B T}\right]$$

$$\frac{\partial T}{\partial t} == \kappa \Delta T - \frac{E_m}{C} \frac{dm}{dt}.$$

Here E_m is the energy released by burning of one unit mass of the fuel and C is the heat capacity. The expression for the burning rate of the fuel $\Gamma[T]$ is taken to be activational (Arrhenius) with ΔU being the activation energy (the potential barrier to overcome for the decay). The first deflagration equation is ODE while the second one is a PDE.

We will consider $1\,d$ problem with the fuel localized in the region $0 \le x \le L$ and we can set the temperature at the ends, as well as the initial temperature, to the environmental temperature T_0. Clearly $T_0 \ll \Delta U$ has to be satisfied, otherwise burning occurs immediately. Then with time the temperature in the depth of the fuel region will increase because of the field decay. If the cooling via heat conduction is sufficient, the system will reach a stationary state with the maximal temperature in the center, $x = L/2$, that only slightly exceeds T_0. If cooling is insufficient (κ too small or L too large) the temperature will continue to increase and thermal runaway occurs. In this case burning occurs throughout the whole region practically at the same time. This is qualitatively similar to the $0\,d$ Semenov model considered above but more involved because of the PDE. The expression for the threshold of thermal runaway in this case was obtained by Frank-Kamenetskii.

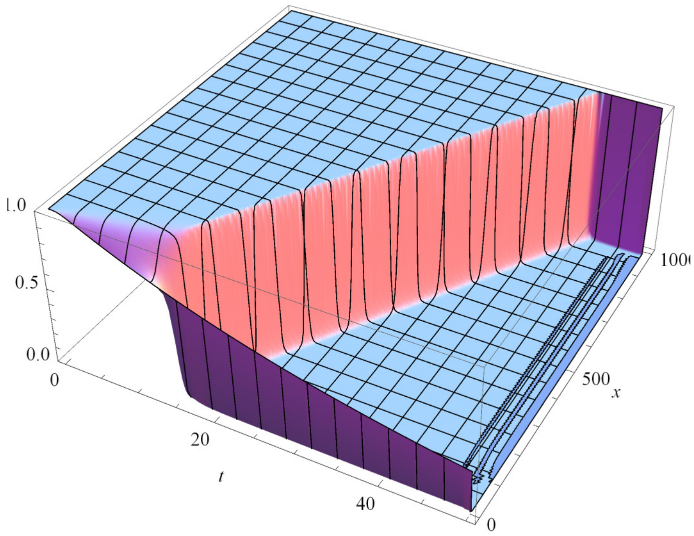
If the temperature is increased or heat is injected locally, burning starts at this point and then deflagration front propagates throughout the sample. This is the most interesting case that will be modeled below. To initiate the deflagration front, we will raise the temperature at the left end of the sample, $x = 0$, during a short time.

In[52]:=
```
L = 1000;  (* Length of the region *)
tMax = 100;  (* Time of calculation *)
kB = 1;
Em = 3000;  (* Energy released per unit mass of burned fuel *)
c = 1;    (* Heat capacity *)
m0 = 1;   (* Initial mass of fuel *)
κ = 200;  (* Thermal diffusivity *)

(* Relaxation rate *)
Γ0 = 20;    ΔU = 4000;
Γ[T_] := Γ0 ⅇ^(-ΔU/T);

T0 = 300;  (* Temperature of the environment *)

(* Ignition by raising temperature on the left end *)
TIgnition = 600;    tIgnition = 1;
TLeft[t_] = T0 + (TIgnition - T0) Tanh[t/tIgnition];

(* Equations and solving *)
EqT = ∂t T[x, t] + Em/c ∂t m[x, t] == κ ∂x,x T[x, t];
Eqm = ∂t m[x, t] == -Γ[T[x, t]] m[x, t];
IniConds = {T[x, 0] == T0, m[x, 0] == 1};
BConds = {T[0, t] == TLeft[t], T[L, t] == T0};
solution = NDSolve[Join[{EqT, Eqm}, IniConds, BConds],
  {T, m}, {x, 0, L}, {t, 0, tMax}, Method → {"MethodOfLines",
    "SpatialDiscretization" → {"TensorProductGrid", "MinPoints" → 200}}]
mxt[x_, t_] := m[x, t] /. First[solution]
Txt[x_, t_] := T[x, t] /. First[solution]
```

Out[68]=
```
{{T → InterpolatingFunction[{{0., 1000.}, {0., 100.}}, <>],
  m → InterpolatingFunction[{{0., 1000.}, {0., 100.}}, <>]}}
```

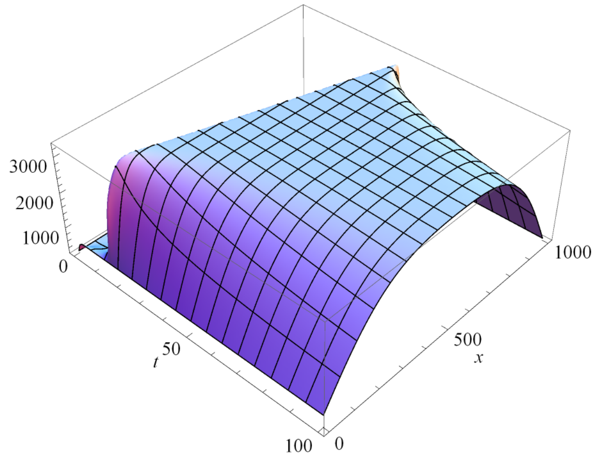
In[71]:=
```
Plot3D[mxt[x, t], {t, 0, 0.5 tMax}, {x, 0, L},
 PlotRange → {0, 1}, PlotPoints → 100, AxesLabel → Automatic]
```

Out[71]=



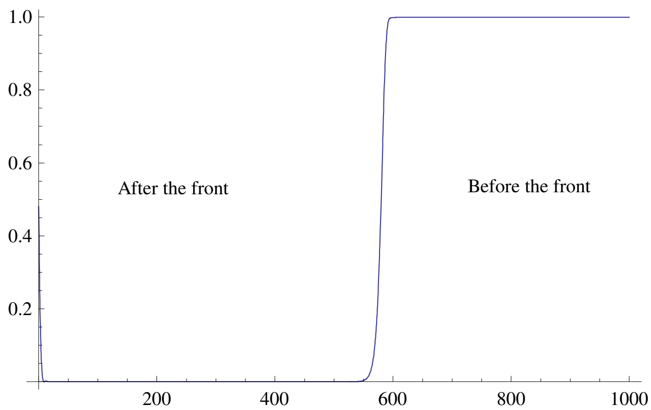
One can see a deflagration front propagating with a constant speed from the left to right end of the region.

```
Plot3D[Txt[x, t], {t, 0, tMax}, {x, 0, L},
 PlotRange → All, PlotPoints → 100, AxesLabel → Automatic]
```

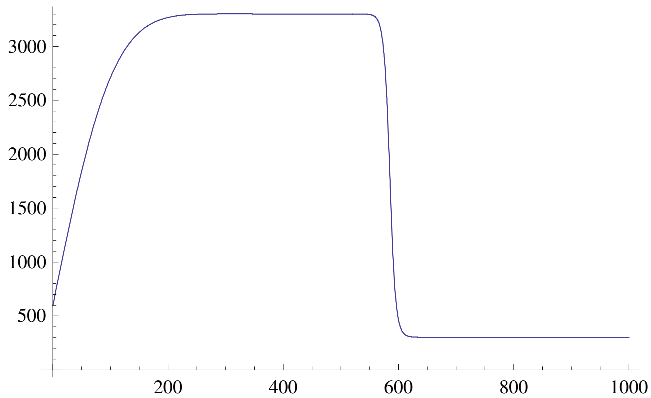


Temperature sharply increases as a result of burning but then begins to decrease because of the energy loss to the environment, especially at the ends. Let us plot spatial profiles of the amount of fuel and temperature at a particular time.

```
tt = 30;
Plot[mxt[x, tt], {x, 0, L}]
```




```
tt = 30;
Plot[Txt[x, tt], {x, 0, L}]
```



Wave equation

Stationary wave equation for monochromatic waves $\Delta\phi + k^2\,\phi == 0$ cannot be solved directly by *Mathematica*'s NDSolve and, unlike the Laplace equation, it cannot be solved by the handy relaxation method, also using NDSolve. Solution of the stationary wave equation requires special methods such as spatial discretization by the user that reduces it to a system of linear algebraic equations that can be then solved numerically by *Mathematica*. Below we will consider only the full time-dependent wave equation with initial conditions (an evolution equation) that can be solved by NDSolve.

1d: Excitation of standing waves in a closed region

Consider a closed $1d$ region between two walls, $0 \le x \le L$. The left wall makes a small-amplitude sinusoidal motion with a frequency ω that excites waves in the system. These waves reflect from the right wall that leads to formation of standing waves. The displacement of the left wall is assumed to be much smaller than L and the wave length $\lambda = 2\,\pi c/\omega$, where c is the speed of the wave. Thus this displacement can be ignored and we set a boundary condition at $x = 0$. On the other hand, motion of the left wall generates motion of the media, so that the boundary condition at $x = 0$ reads $v[0,\ t] = v_{\text{wall}}[t]$. To avoid the problem of inconsistency of initial and boundary conditions, we switch on oscillations of the left wall gradually from zero. We solve the wave equation for the media velocity potential ϕ and them find the media velocity as $v[x,\ t] = -\partial_x \phi[x,\ t]$. This model describes standing waves in a pipe with both ends closed, whereas the influence of the side walls is neglected.

The results show that the wave amplitude unlimitedly increases with time if the wave length coincides with the fundamental wave length

$$\lambda_1 = 2\,L$$

or those of the overtones,

$$\lambda_n = \frac{\lambda_1}{n}, \quad n = 1, 2, 3, \ldots$$

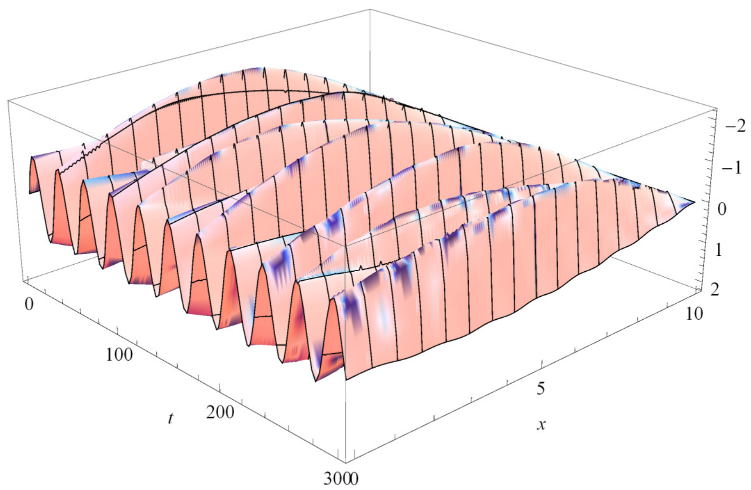
This is a resonance behavior.

```
L = 10;
λ = 2^0 L; (* Wave length.  λ1=2L is the fundamental wave length in this case *)
c = 1;   (* Wave speed *)
ω = 2 π c / λ;  (* Frequency *)
tMax = 30 L/c;  (* Maximal time of the calculation. L/c
 is the time for the wave to cross the region *)

Equation = c^2 ∂x,x ϕ[x, t] - ∂t,t ϕ[x, t] == 0;
(* 1d wave equation for the velocity potential *)
IConds = {ϕ[x, 0] == 0, (∂t ϕ[x, t] /. t → 0) == 0};  (* All zero at initial time *)
BConds = { (∂x ϕ[x, t] /. x → 0) == Tanh[t] Sin[ ω t], (* The excitation begins
    gradually to avoid inconsistency of initial and boundary conditions *)
           (∂x ϕ[x, t] /. x → L) == 0   (* Closed on the right end *)
      };

Timing[Sol = NDSolve[Join[{Equation}, BConds, IConds],
    ϕ, {x, 0, L}, {t, 0, tMax}, Method → {"MethodOfLines",
      "SpatialDiscretization" → {"TensorProductGrid", "MinPoints" → 100}}];]
ϕxt[x_, t_] := ϕ[x, t] /. Sol[[1]]
vxt[x_, t_] := -∂xx ϕxt[xx, t] /. xx → x
(* Velocity of the media in the wave is negative gradient of its potential *)
```

```
{7.188, Null}
```

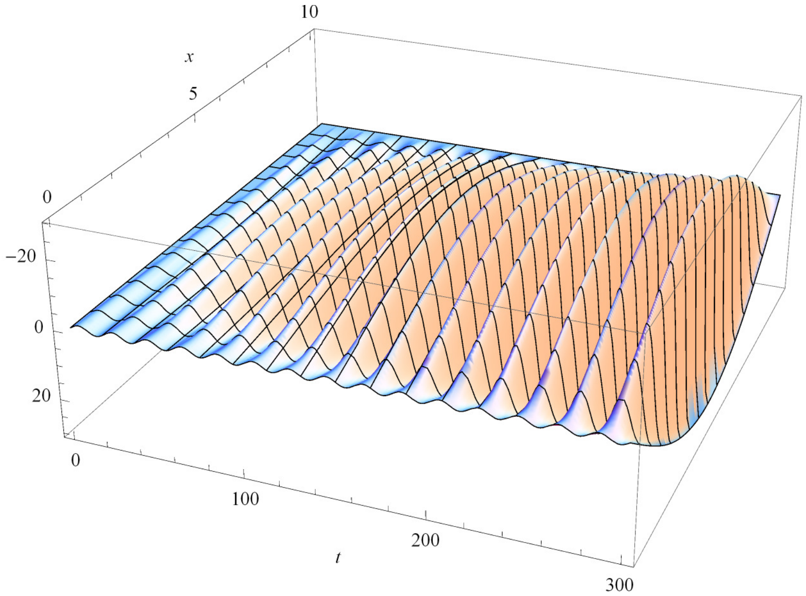
Calculation with $\lambda = 2^{3/2}$ L - longer than the fundamental wave length, nonresonant case. The wave pattern is irregular and the wave amplitude does not increase with time, remaining of the order of the excitation amplitude at the left end.

```
Plot3D[vxt[x, t], {x, 0, L}, {t, 0, tMax}, PlotPoints → 100, AxesLabel → Automatic]
```



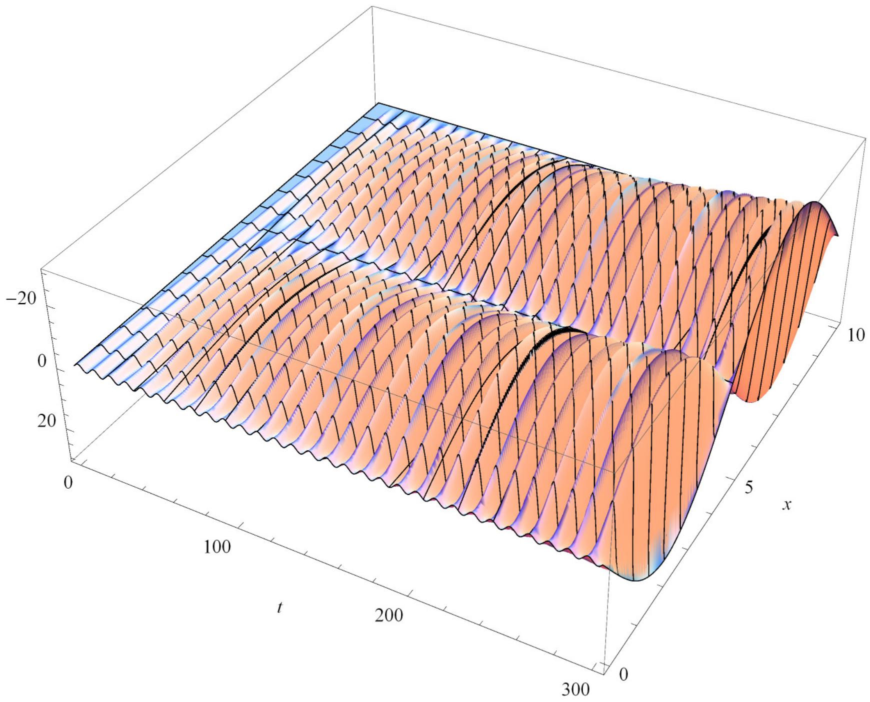
Calculation with $\lambda = 2$ L - the fundamental wave length, resonance. The wave pattern is regular with nodes at both boundaries, and the wave amplitude unlimitedly increases with time, as always the case for resonance in linear undamped systems.

```
Plot3D[vxt[x, t], {x, 0, L}, {t, 0, tMax}, PlotPoints → 100, AxesLabel → Automatic]
```



Calculation with λ = L - the second overtone. The wave pattern is regular with nodes at both boundaries and in the middle, and the wave amplitude unlimitedly increases with time.

```
Plot3D[vxt[x, t], {x, 0, L}, {t, 0, tMax}, PlotPoints → 100, AxesLabel → Automatic]
```



Let us now consider excitation of standing waves in a pipe with the left end exciting waves as before and the right end open. As before, we consider the problem as 1d neglecting the influence of side walls. The boundary condition at the right end changes to $\phi[L, t] == 0$. To see it, use Eq. (5), $\partial_t \phi \propto \delta P$. At the open end, the pressure with a good accuracy (although not absolutely) merges with the constant atmospheric pressure, thus $\delta P[L, t] == 0$. Thus also $\partial_t \phi[L, t] == 0$. Integrating over time, one obtains $\phi[L, t] ==$ const. However, the potential is defined up to an arbitrary constant, so that this constant can be set to zero.

The results show that the wave amplitude unlimitedly increases with time if the wave length coincides with the fundamental wave length

$\lambda_1 = 4\ L$

or those of the overtones,

$$\lambda_n = \frac{\lambda_1}{n}, \quad n = 1, 3, 5, \ldots \quad \text{(even overtones absent)}$$

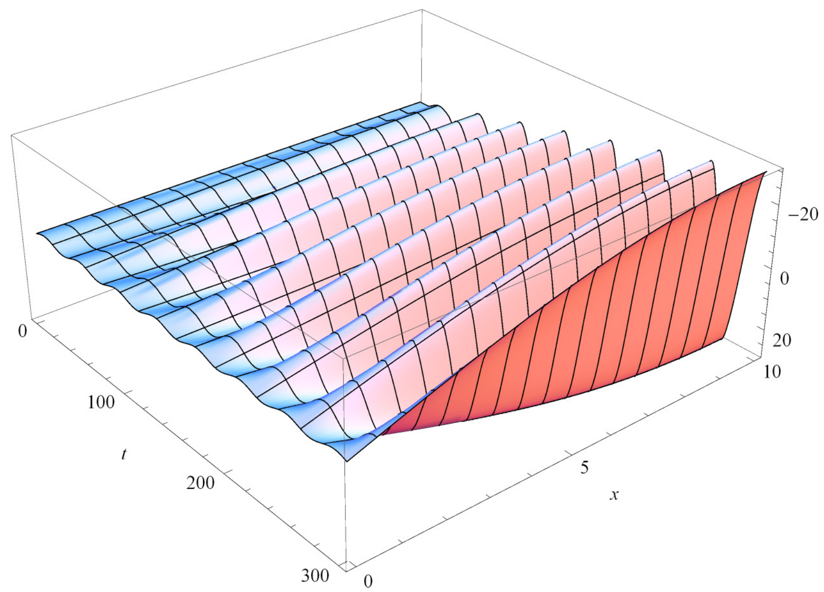
This is a resonance behavior.

```
L = 10;
λ = 4 L; (* Wave length.  λ1=4L is the fundamental wave length in this case *)
c = 1;   (* Wave speed *)
ω = 2 π c / λ;  (* Frequency *)
tMax = 30 L/c;  (* Maximal time of the calculation. L/c
 is the time for the wave to cross the region *)

Equation = c^2 ∂x,x ϕ[x, t] - ∂t,t ϕ[x, t] == 0;
(* 1d wave equation for the velocity potential *)
IConds = {ϕ[x, 0] == 0, (∂t ϕ[x, t] /. t → 0) == 0};  (* All zero at initial time *)
BConds = { (∂x ϕ[x, t] /. x → 0) == Tanh[t] Sin[ ω t], (* The excitation begins
    gradually to avoid inconsistency of initial and boundary conditions *)
          ϕ[L, t] == 0   (* Closed on the right end *)
      };

Timing[Sol = NDSolve[Join[{Equation}, BConds, IConds],
    ϕ, {x, 0, L}, {t, 0, tMax}, Method → {"MethodOfLines",
      "SpatialDiscretization" → {"TensorProductGrid", "MinPoints" → 100}}];]
ϕxt[x_, t_] := ϕ[x, t] /. Sol[[1]]
vxt[x_, t_] := -∂xx ϕxt[xx, t] /. xx → x
(* Velocity of the media in the wave is negative gradient of its potential *)
```

```
{2.61, Null}
```

Calculation with $\lambda = 4\ L$ - the fundamental. The wave pattern is regular with a node at the left and an anti-node at the right ends of the pipe, and the wave amplitude unlimitedly increases with time.

```
Plot3D[vxt[x, t], {x, 0, L}, {t, 0, tMax}, PlotPoints → 100, AxesLabel → Automatic]
```



Further one can check that there is no resonance at $\lambda = 2L$ that would be the second overtone etc. Finding the intensity of the sound emitted from the open end is a more complicated numerical problem because the region external to the pipe is not rectangular. However, there is an analytical solution to this problem.

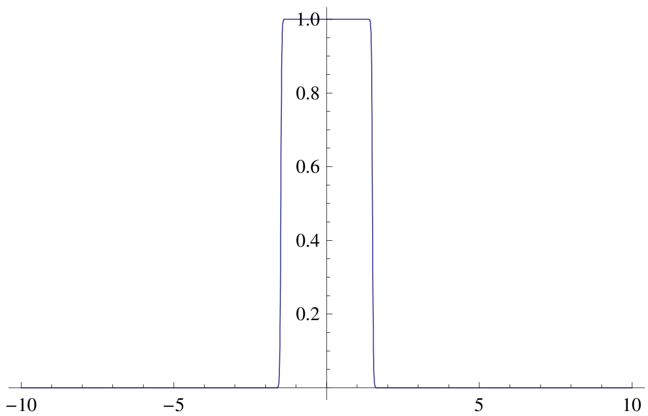
- **2d: One-slit diffraction in time**

One slit diffraction results in the minima of the wave intensity for the directions specified by the angle θ that satisfy

$$d_0 \, \mathrm{Sin}[\theta] = m \, \lambda, \quad m = \pm 1, \pm 2, \ldots \tag{20}$$

Here d_0 is the width of the slit and λ is the wave length. We solve the wave equation in a rectangular region with the slip in the middle of the bottom line within the time interval between opening the slit and the wave hitting the far (top) wall. (Stationary wave equation cannot be solved by *Mathematica* at the moment.)

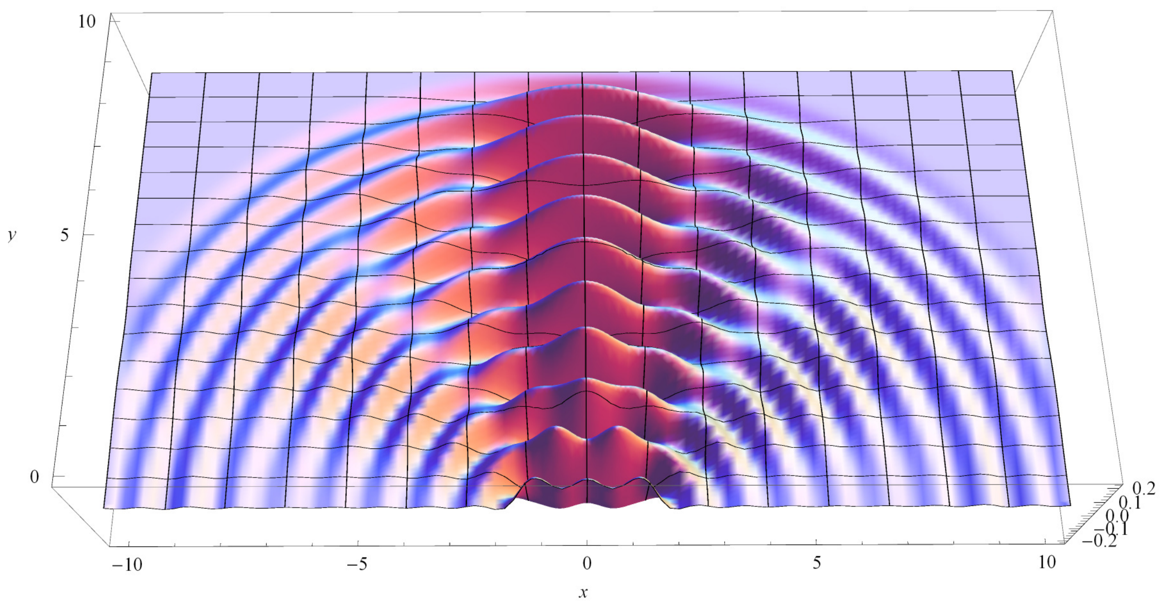
```
Lx = 10;  Ly = 10;
d0 = 3;  (* Width of the slit *)
λ = 1;  (* Wave length *)
c = 1;  (* Wave speed *)
ω = 2 π c / λ;  (* Frequency *)

tMax = 10;  (* Has to be chosen so that
 the first wave front is about to hit the far wall,
so that there is no reflection yet. Reflections will complicate the wave pattern *)

(* Modeling the slit *)
PowSlit = 50;
FSlit[x_] = 1 / (1 + (2 x / d0)^(2 PowSlit));
Plot[FSlit[x], {x, -Lx, Lx}, PlotRange → All]

(* Creating and solving the equations for the velocity potential *)
(* Boundary conditions: perpendicular component of the velocity on boundaries is zero,
except for the slit and is a given periodic function on the slit *)
Equation = ∂x,x ϕ[x, y, t] + ∂y,y ϕ[x, y, t] - c^-2 ∂t,t ϕ[x, y, t] == 0;
IConds = {ϕ[x, y, 0] == 0, (∂t ϕ[x, y, t] /. t → 0) == 0};  (* All zero at initial time *)
BConds = { (∂y ϕ[x, y, t] /. y → 0) == FSlit[x] Tanh[t] Sin[ ω t],  (* The slit opens
    gradually to avoid inconsistency of initial and boundary conditions *)
        (∂y ϕ[x, y, t] /. y → Ly) == 0, (* This BC can be removed *)
        (∂x ϕ[x, y, t] /. x → Lx) == 0,
        (∂x ϕ[x, y, t] /. x → -Lx) == 0   };
Timing[Sol = NDSolve[Join[{Equation}, BConds, IConds],
    ϕ, {x, -Lx, Lx}, {y, 0, Ly}, {t, 0, tMax}, Method → {"MethodOfLines",
      "SpatialDiscretization" → {"TensorProductGrid", "MinPoints" → 100}}];]
ϕxyt[x_, y_, t_] := ϕ[x, y, t] /. Sol[[1]]
```



NDSolve::eerr:
Warning: Scaled local spatial error estimate of 83.86174822863588` at t = 10.` in the direction of independent variable x is much greater than prescribed error tolerance. Grid spacing with 100 points may be too large to achieve the desired accuracy or precision. A singularity may have formed or you may want to specify a smaller grid spacing using the MaxStepSize or MinPoints method options. ≫

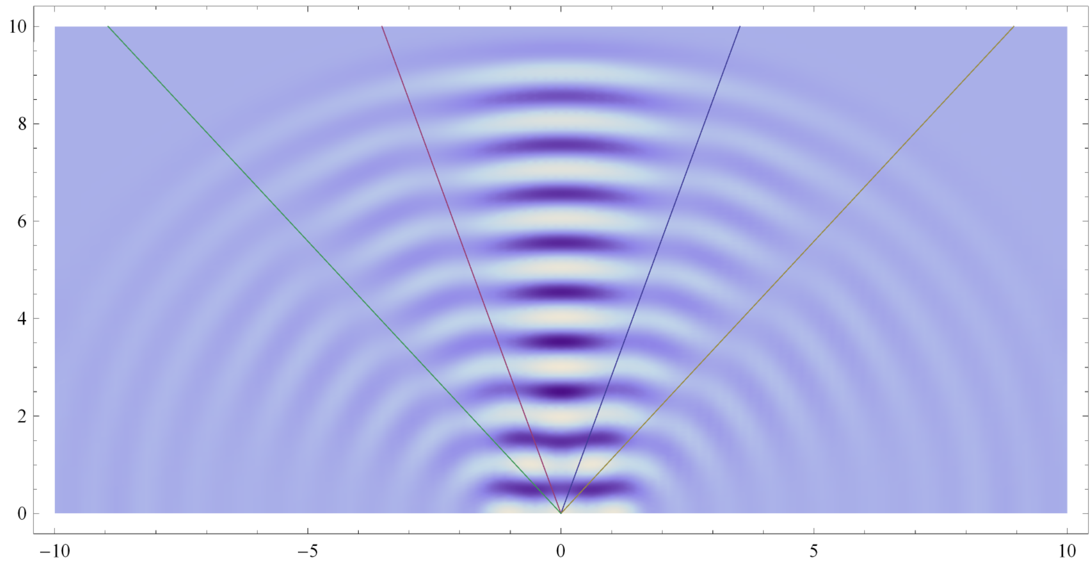
```
{85.344, Null}
```

```
tt = tMax;
Plot3D[φxyt[x, y, tt], {x, -Lx, Lx}, {y, 0, Ly}, PlotRange → All,
  AxesLabel → Automatic, PlotPoints → 100, AspectRatio → Ly/(2 Lx)]
```



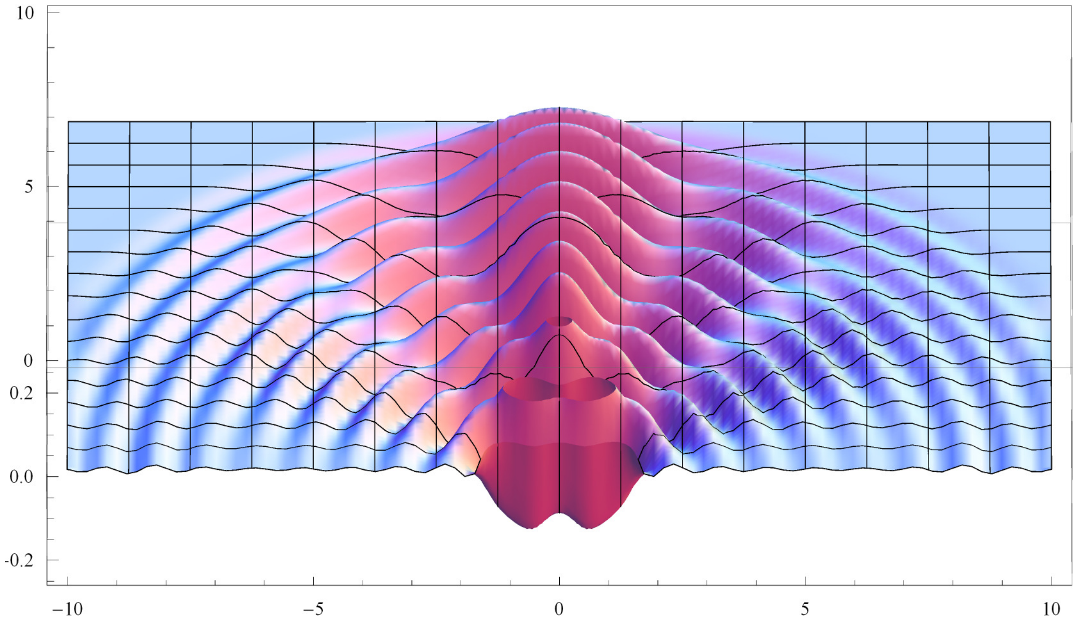
Plotting the wave pattern together with the theoretical result for the directions of the minima

```
tt = 10;
Show[
 DensityPlot[ϕxyt[x, y, tt], {x, -Lx, Lx}, {y, 0, Ly},
  PlotRange → All, PlotPoints → 100, AspectRatio → Ly/(2 Lx)],
 Plot[{Cot[ArcSin[λ/d0]] x, -Cot[ArcSin[λ/d0]] x, Cot[ArcSin[(2 λ)/d0]] x, -Cot[ArcSin[(2 λ)/d0]] x},
  {x, -10, 10}, PlotRange → {{-Ly, Ly}, {0, Lx}}, AspectRatio → Lx/(2 Ly)]
]
```



Let us create an animated GIF file to put it onto a web site.

```
(* Preparing to export into animated GIF *)
(* Set PlotRange (to avoid jitter), Viewpoint, BoxRatios, and ImageSize *)
tt = tMax;
Plot3D[ϕxyt[x, y, tt], {x, -Lx, Lx}, {y, 0, Ly}, PlotRange → {-0.26, 0.26},
 PlotPoints → 100, BoxRatios → {1, Ly/(2 Lx), 0.3}, ViewPoint → {0, -100, 100}, ImageSize → 500]
```




```
(* Export into animated GIF *)
(* Create a list of frames, Plot3Ds at different times *)

WaveFrame[t_] := Plot3D[ϕxyt[x, y, t], {x, -Lx, Lx}, {y, 0, Ly},
  PlotRange → {-0.26, 0.26}, PlotPoints → 100, BoxRatios → {1, Ly/(2 Lx), 0.3},
  ViewPoint → {0, -100, 100}, Axes → None, ImageSize → 500]
NFrames = 100;
tFrame[k_] := tMax k/NFrames;
WaveFrames = Table[WaveFrame[tFrame[k]], {k, 1, NFrames}];
Export["D:\\_Lehman\\2010 Fall\\PHY 307\\Notes\Diffraction-one-slit.gif",
  WaveFrames(*,"TransparentColor"→White*)] ;
```

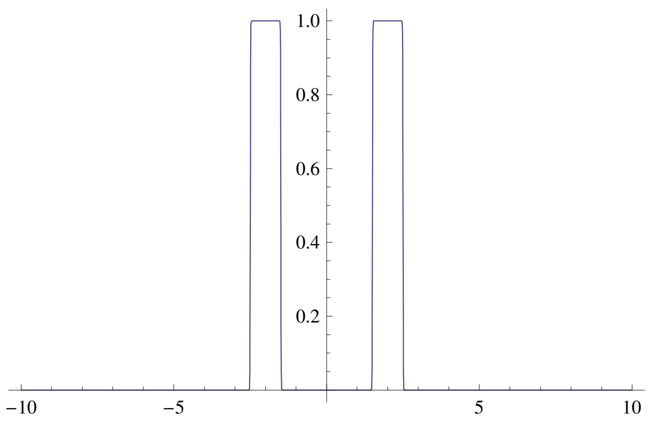
- **2d: Two-slit diffraction in time**

Two-slit diffraction results in the maxima of the wave intensity for the directions specified by the angle θ that satisfy

$$d \, \mathrm{Sin}[\theta] = m \lambda, \quad m = 0, \pm 1, \pm 2, \ldots$$

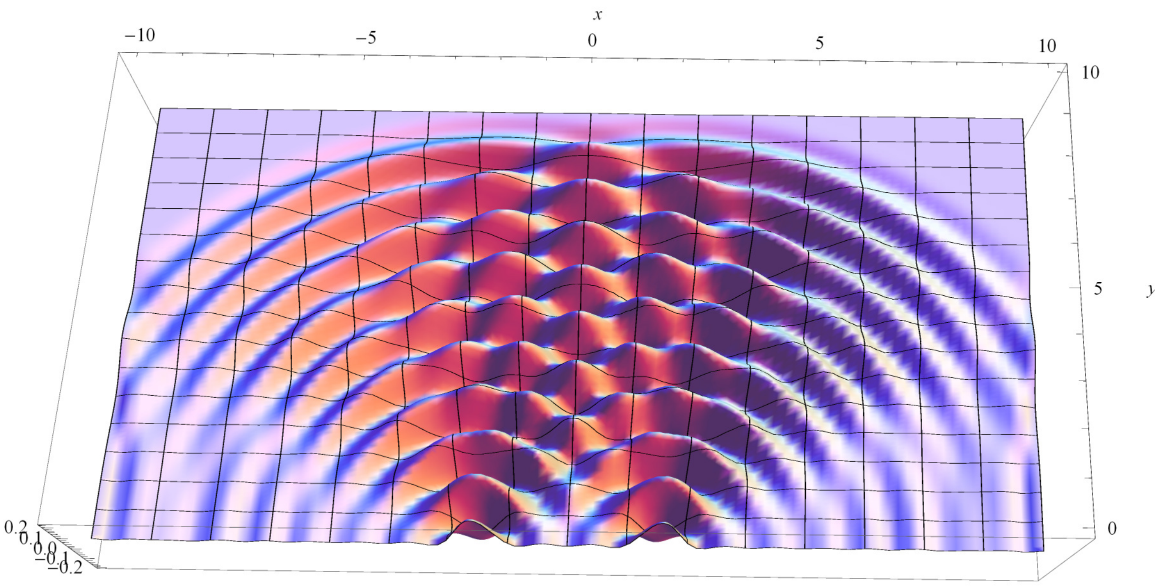
Here d is the distance between the slits and λ is the wave length. We solve the wave equation in a rectangular region with the slips in the middle of the bottom line within the time interval between opening the slits and the wave hitting the far (top) wall. (Stationary wave equation cannot be solved by *Mathematica* at the moment.)

```
Lx = 10;  Ly = 10;
d0 = 1;  (* Width of a slit *)
d = 4;   (* Distance between slits *)
λ = 1;  (* Wave length *)
c = 1;   (* Wave speed *)
ω = 2 π c / λ;  (* Frequency *)

tMax = 10;  (* Has to be chosen so that
 the first wave front is about to hit the far wall,
so that there is no reflection yet. Reflections will complicate the wave pattern *)

(* Modeling the slits *)
PowSlit = 50;
FSlit[x_] = 1 / (1 + ((x+d/2)/(d0/2))^(2 PowSlit)) + 1 / (1 + ((x-d/2)/(d0/2))^(2 PowSlit));
Plot[FSlit[x], {x, -Lx, Lx}, PlotRange → All]

(* Creating and solving the equations for the velocity potential *)
(* Boundary conditions: perpendicular component of the velocity on boundaries is zero,
except for slits and is a given periodic function on slits *)
Equation = ∂x,x ϕ[x, y, t] + ∂y,y ϕ[x, y, t] - c^-2 ∂t,t ϕ[x, y, t] == 0;
IConds = {ϕ[x, y, 0] == 0, (∂t ϕ[x, y, t] /. t → 0) == 0}; (* All zero at initial time *)
BConds = { (∂y ϕ[x, y, t] /. y → 0) == FSlit[x] Tanh[t] Sin[ ω t],  (* Slits open
    gradually to avoid inconsistency of initial and boundary conditions *)
          (∂y ϕ[x, y, t] /. y → Ly) == 0,  (* This BC can be removed *)
          (∂x ϕ[x, y, t] /. x → Lx) == 0,
          (∂x ϕ[x, y, t] /. x → -Lx) == 0    };
Timing[Sol = NDSolve[Join[{Equation}, BConds, IConds],
    ϕ, {x, -Lx, Lx}, {y, 0, Ly}, {t, 0, tMax}, Method → {"MethodOfLines",
      "SpatialDiscretization" → {"TensorProductGrid", "MinPoints" → 100}}];]
ϕxyt[x_, y_, t_] := ϕ[x, y, t] /. Sol[[1]]
```



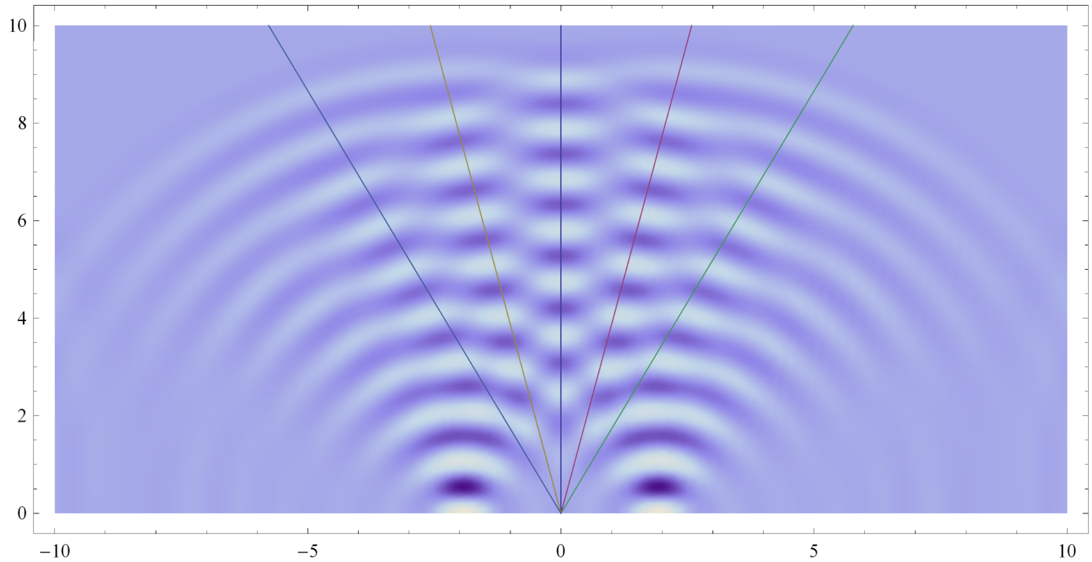

```
{141.469, Null}
```

```
tt = 10;
Plot3D[ϕxyt[x, y, tt], {x, -Lx, Lx}, {y, 0, Ly}, PlotRange → All,
  AxesLabel → Automatic, PlotPoints → 100, AspectRatio → Ly/(2 Lx)]
```



Plotting the wave pattern together with the theoretical result for the directions of the maxima

```
tt = 10;
Show[
 DensityPlot[ϕxyt[x, y, tt], {x, -Lx, Lx}, {y, 0, Ly},
  PlotRange → All, PlotPoints → 100, AspectRatio → Ly/(2 Lx)],
 Plot[{10^10 x, Cot[ArcSin[λ/d]] x, -Cot[ArcSin[λ/d]] x,
   Cot[ArcSin[2 λ/d]] x, -Cot[ArcSin[2 λ/d]] x}, {x, -10, 10},
  PlotRange → {{-Ly, Ly}, {0, Lx}}, AspectRatio → Lx/(2 Ly)]
]
```



■ Schrödinger equation

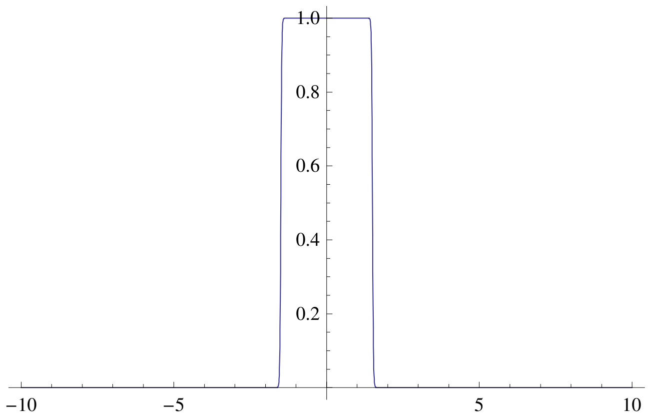
■ 2d: One-slit diffraction in time

Since the original complex Schrödinger equation cannot be solved by NDSolve, its real and complex parts have to be separated that leads to doubling the equation and boundary and initial conditions. We consider a plane wave, Eq. (14) falling on a slit from $y < 0$ and solve the time-dependent Schrödinger equation, Eq. (12) on the other side of the slit in the region $0 \le y \le L_y$ and $-L_x \le x \le L_x$. We assume that the region is bounded by rigid walls, so that the boundary condition is $\Psi == 0$ everywhere except the slit of width d_0 in the bottom wall, where is Ψ is the same as in the incident plane wave, $\Psi \sim \text{Exp}[-i\omega t]$. It is convenient to express the frequency ω through the particle's de Broglie wave length λ as

$$\omega = \frac{\hbar}{2\,m}\left(\frac{2\,\pi}{\lambda}\right)^2,$$

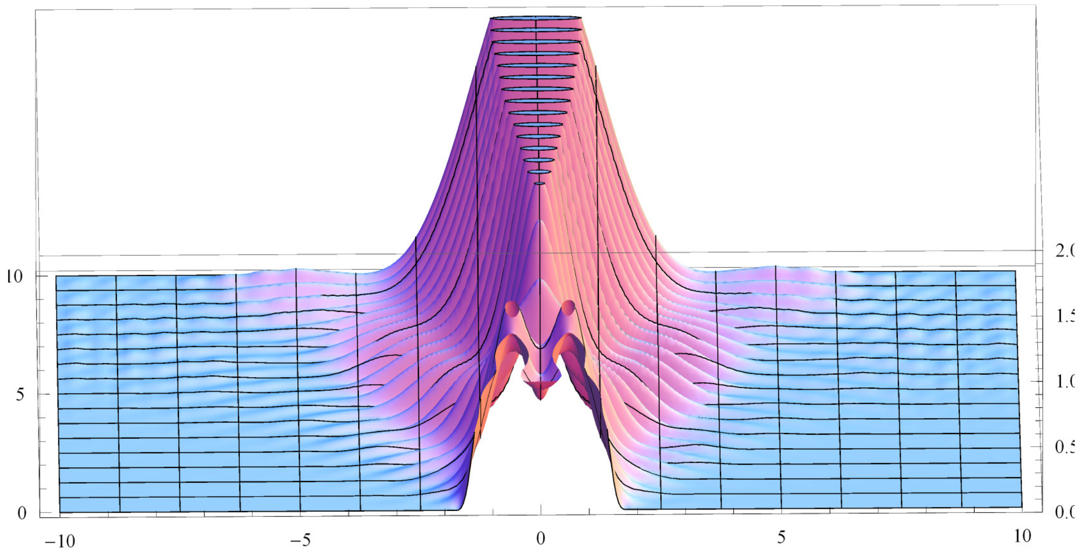
see Eqs. (15) and (16). Having λ as a parameter in calculations, we can relate numerical result to the one-slit diffraction formula, Eq. (20). We gradually open the slit starting at $t = 0$, to avoid inconsistency of the initial and boundary conditions. As the initial condition we take $\Psi[x,\ y,\ 0] == 0$, no particles in the box.

```
Lx = 10;   Ly = 10;   (* Sizes of the rectangular spatial region *)
ℏ = 1;   (* Planck's constant *)
m = 1;   (* Mass of the particle *)
d0 = 3;  (* Width of the slit *)
λ = 1;  (* De Broglie wave length of the particle *)

ω = ℏ/(2 m) (2 π/λ)^2;  (* Frequency corresponding to de Broglie wave length *)

tMax = 3;  (* Time of the calculation *)

(* Modeling the slit *)
PowSlit = 50;
FSlit[x_] = 1/(1 + (2 x / d0)^(2 PowSlit));
Plot[FSlit[x], {x, -Lx, Lx}, PlotRange → All]
tOpenSlit = 0.1;  (* Time to fully open the slit *)

(* Creating and solving the equations for the velocity potential *)
(* Boundary conditions: perpendicular component of the velocity on boundaries is zero,
except for the slit and is a given periodic function on the slit *)
Equations = { ℏ ∂t ΨRe[x, y, t] + ℏ^2/(2 m) (∂x,x ΨIm[x, y, t] + ∂y,y ΨIm[x, y, t]) == 0,
    -ℏ ∂t ΨIm[x, y, t] + ℏ^2/(2 m) (∂x,x ΨRe[x, y, t] + ∂y,y ΨRe[x, y, t]) == 0
   };
IConds = {ΨRe[x, y, 0] == 0, ΨIm[x, y, 0] == 0};  (* All zero at initial time *)
BConds = { ΨRe[x, 0, t] == FSlit[x] Tanh[t/tOpenSlit] Cos[ω t],
    ΨIm[x, 0, t] == -FSlit[x] Tanh[t/tOpenSlit] Sin[ω t], (* The slit opens
  gradually to avoid inconsistency of initial and boundary conditions *)
      ΨRe[x, Ly, t] == 0,   ΨIm[x, Ly, t] == 0,
      ΨRe[Lx, y, t] == 0,   ΨIm[Lx, y, t] == 0,
      ΨRe[-Lx, y, t] == 0,   ΨIm[-Lx, y, t] == 0 };
Timing[Sol = NDSolve[Join[Equations, BConds, IConds], {ΨRe, ΨIm},
   {x, -Lx, Lx}, {y, 0, Ly}, {t, 0, tMax}, Method → {"MethodOfLines",
    "SpatialDiscretization" → {"TensorProductGrid", "MinPoints" → 80}}];]
ΨRexyt[x_, y_, t_] := ΨRe[x, y, t] /. Sol[[1]]
ΨImxyt[x_, y_, t_] := ΨIm[x, y, t] /. Sol[[1]]
Ψ2xyt[x_, y_, t_] := ΨRexyt[x, y, t]^2 + ΨImxyt[x, y, t]^2
```




```
{289.36, Null}
```

```
tt = tMax;
Plot3D[Ψ2xyt[x, y, tt], {x, -Lx, Lx}, {y, 0, Ly}, PlotRange → {0, 2}, PlotPoints → 100,
 BoxRatios → {1, Ly/(2 Lx), 0.3}, ViewPoint → {0, -100, 100}, ImageSize → 500]
```



The results show that quantum particles entering the box via the slit undergo diffraction with the directions of diffraction minima in accordance with the elementary wave theory. Before the quantum wave hits the far wall, the plotted probability density $|\Psi[r,\ t]|^2$ remains smooth. However, after the beam of particles hits the far wall, because of the interference of the incident and reflected wave the wave pattern becomes oscillating in space. The total probability for a particle to be in the box $\int |\Psi[\mathbf{r},\ t]|^2\, d\mathbf{r}$ steadily increases as more and more particles are entering the box via the slit.

Let us create an animated GIF file to put it onto a web site.

```
(* Preparing to export into animated GIF *)
(* Set PlotRange (to avoid jitter), Viewpoint, BoxRatios, and ImageSize *)
WaveFrame[t_] :=
 Plot3D[Ψ2xyt[x, y, t], {x, -Lx, Lx}, {y, 0, Ly}, PlotRange → {0, 2}, PlotPoints → 100,
  BoxRatios → {1, Ly/(2 Lx), 0.3}, ViewPoint → {0, -100, 100}, Axes → None, ImageSize → 500]
(* Export into animated GIF *)
(* Create a list of frames, Plot3Ds at different times *)
NFrames = 100;    tFrame[k_] := tMax k/NFrames;
WaveFrames = Table[WaveFrame[tFrame[k]], {k, 1, NFrames}];
Export["D:\\_Lehman\\2010 Fall\\PHY 307\\Notes\Diffraction_of_electron_one-slit.gif",
  WaveFrames(*,"TransparentColor"→White*)] ;
```