Numerical Methods

Assignment 05

Georgios Christos Chouliaras (2592496)

May 25, 2017

Exercise 1

The function

The function `[T,U] = Chouliaras_assignment5_exercise1(f, S, u0, theta, h)` solves the initial value problem for a system of differential equations using the θ-method.
The inputs of the function are a function handle `f` which denotes the right-hand side $f(t, u)$ of the differential equation, a vector `S` which includes the requested points in time, a column vector `u0` with the initial conditions, the parameter `theta`, where $\theta \in [0, 1]$ and the step-size `h`. If the vector `S` consists of exactly two elements, the function gives output for all the calculated points in time including the times in S, however if it has more than two elements, it returns the solution only for those points in time.
The output of the function is a column vector of times `T` and a matrix `U` which has in every row the solution at the time corresponding to that row in T.
As an example, in order to solve $u' = -u$ in $[0, 20]$ with initial condition $u_0 = 7$, $\theta = 0.5$ (Crank-Nicolson scheme) and step-size $h = 0.1$, one should type:
`[T,U] = Chouliaras_assignment5_exercise1(@(t,u) -u, [0 20], 7, 0.5, 0.1)`

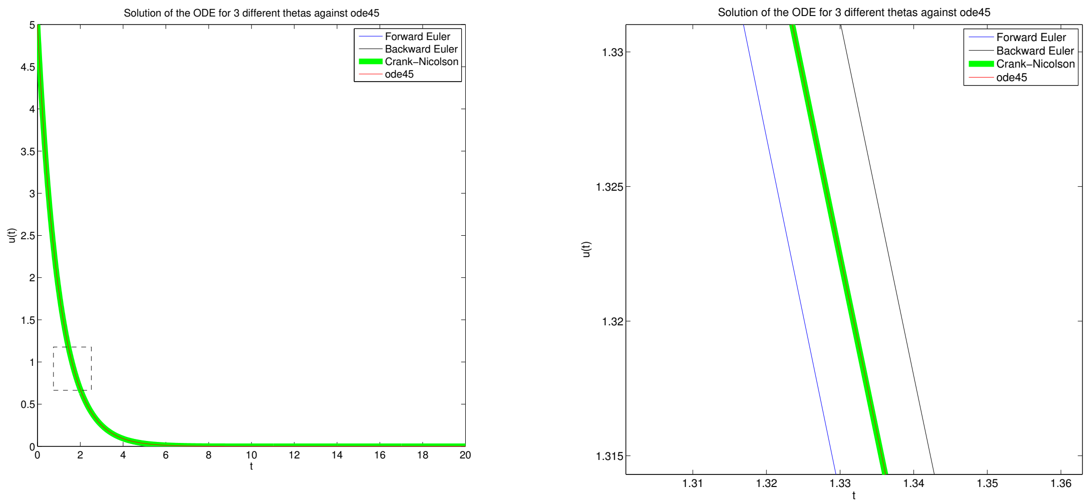
The method

For the differential equation $u' = f(t, u)$ the θ-method with step-size h is given by $u_{n+1} = u_n + h[(1 - \theta)f(t_n, u_n) + \theta f(t_{n+1}, u_{n+1})]$. The function $f : \mathbb{R} \times \mathbb{R}^k \to \mathbb{R}^k$, with $k \in \mathbb{N}$ defines the system (in case $k > 1$) of differential equations. For $\theta > 0$ the scheme of the θ-method is implicit, which means that we cannot find u_n directly from u_{n+1} but we need to solve non-linear equations. Due to this, we rewrite the scheme as $G(u_{n+1}) = u_{n+1} - u_n - h[(1 - \theta)f(t_n, u_n) + \theta f(t_{n+1}, u_{n+1})] = \mathbf{0}$ and we use Newton with tolerance 10^{-4} and maximum 150 steps to solve this non-linear problem. In case that the vector S is given as a time interval of the form $[S_{start}, S_{end}]$ the function returns the solution from the starting point S_{start} with all calculated points in time up until and including the final point S_{end}. If the vector S includes many points, i.e. is of the form $S = [S_1, S_2, \ldots, S_{end}]$, we must provide only those points in time. However, these points are not precisely the instants where we computed the solution, thus we interpolate at the requested points using splines.

Discussion & Illustration

After experimenting with the function it became clear that the parameter θ as well as the step size h play an important role in the accuracy of the solution. In order to evaluate the function, its solution was compared with the built-in `ode45` solver. In general a larger step size resulted to worse performance, meaning that the solution diverged from that of `ode45`. As an example a step size of $h = 0.01$ produces a solution that is much more similar to the one of `ode45`, while $h = 0.1$ results to a solution that diverges in a greater extent. In order to evaluate the algorithm we first solve the simple ODE $u' = -u$ with initial condition $u_0 = 5$, for which we know by paper that the solution is $u(t) = 5e^{-t}$, hence we expect the solution to have an exponential form. In figure 1a we plot the solution of `ode45` along with our function in $S \in [0, 20]$ with $h = 0.01$ and for 3 different $\theta = 0, 1, 0.5$ corresponding to forward Euler, backward Euler

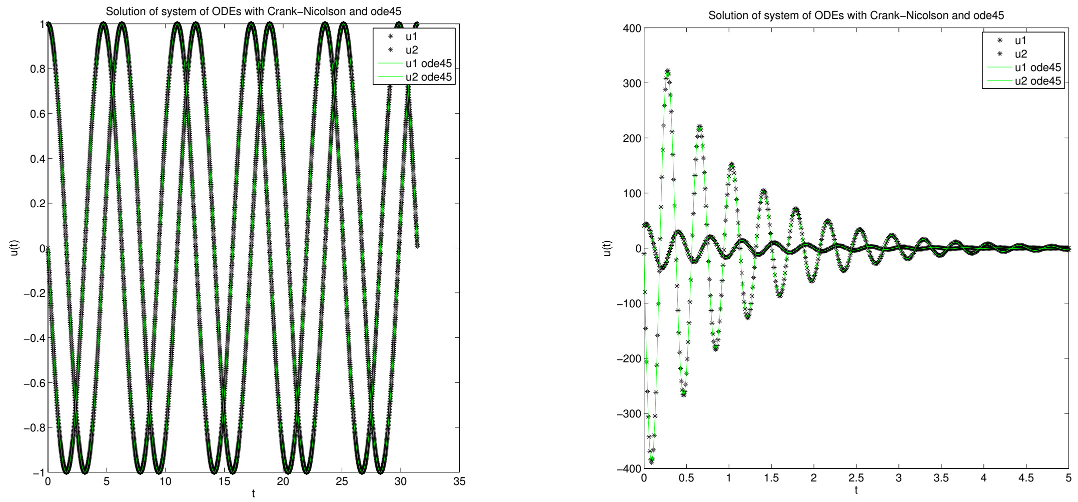
and Crank-Nicolson respectively. Since from this plot it is difficult to conduct a comparison between the different results, we zoom-in to the dashed rectangle shown in the plot. The zoom can be seen in figure 1b. As it can be seen, the result of Crank-Nicolson is identical with the one of `ode45` while forward and backward Euler diverge at a greater extent. Hence, we can verify the fact we know from theory that the Crank - Nicolson method outperforms both forward and backward Euler method. It must be noted that the fact that the results are almost identical with `ode45` does not mean that we found the exact solution, since `ode45` also approximates the solution.
Another example is the system of ODE's $u_1' = u_2$, $u_2' = -u_1$ for $S \in [0, 10\pi]$, initials conditions $u_0 =$ [0,1], $\theta = 0.5$ and $h = 0.01$. We know by paper that the solutions are linear combinations of sine and cosine something that can be verified by the plot in figure 2a. Again, a comparison with `ode45` solver shows that the solutions are almost identical something that verifies the efficiency of the function. In figure 2b can be seen a picture of the solution curves for the system $u_1' = 2u_2 + 8u_1$, $u_2' = -10u_2 - 180u_1$ for $S \in [0, 5]$, $u_0 = [40, -10]$, $\theta = 0.5$, $h = 0.01$. The solutions of this system represent dampened oscillations and by comparing the solutions of our function (with Crank-Nicolson scheme) to those of `ode45`, the accuracy of the function is verified once more.



(a) Solution of u' = -u for $S \in [0, 20]$. The dashed rectangle indicates the zoom area.

(b) Zoom of the solution curve to make comparison clear.

Figure 1: (a) Solution of u' = -u for time interval [0,20], $u_0 = 5$, step-size 0.01 and $\theta = 0, 1, 0.5$. Comparison with built in `ode45` solver. (b) Zoom of the solution to make the comparison clear. Crank-Nicolson produces identical result with `ode45` while the forward and backward Euler diverge at some extent.



(a) Solution of system of ODE's for $S \in [0, 10\pi]$.

(b) Solution of system of ODE's for $S \in [0, 5]$

Figure 2: (a) Solution of system $u_1' = u_2$, $u_2' = -u_1$ *for* $S \in [0, 10\pi]$ *,*u_0 *= [0,1],* $\theta = 0.5$, $h = 0.01$. *Comparison with* `ode45` *solver shows almost identical results. (b) Solution of system* $u_1' = 2u_2 + 8u_1$, $u_2' = -10u_2 - 180u_1$ *for* $S \in [0, 5]$, $u_0 = [40, -10]$, $\theta = 0.5$, $h = 0.01$. *Again, comparison with* `ode45` *shows identical results.*

Exercise 2

The function

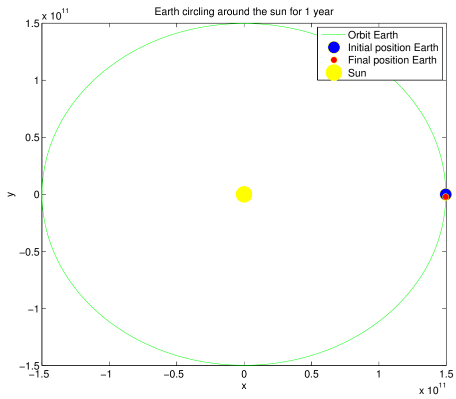
The function `[T, posvel] = Chouliaras_assignment5_exercise2(m,x0,v0,Tin)` shows the movement of planets in the solar system. Given a set of planet masses, their initial positions and velocities as well as the length of the time to compute the movement, the function plots the orbits of the planets and returns a vector of points in time and a matrix with the corresponding positions and velocities of the planets.
The inputs of the function are a vector `m` with the masses of the planets of length p (p stands for the number of planets), a $n \times p$ (n is the number of dimensions) matrix of their initial positions `x0`, a $n \times p$ matrix of their initial velocities `v0` and the length of time `Tin` over which the movement of the planets is computed. It is recommended for the input parameters to be in SI units.
The output parameters are a vector of points in time `T` and a matrix `posvel` which includes the corresponding positions in the first half of the columns and the corresponding velocities in the second half.
As an example, in order to plot the movement of the Earth circling around the Sun for one year in 2 dimensions, we first write the inputs in an appropriate way: `m = [1.9885e30;5.9724e24]` (kg), `x0 = [0, 1.496e11;0,0]` (m), `v0 = [0,0;0,29800]` (m/s), `Tin = 365*24*60*60`(s). Then we call the function as follows:
`[T,posvel] = Chouliaras_assignment5_exercise2(m,x0,v0,Tin)`

The method

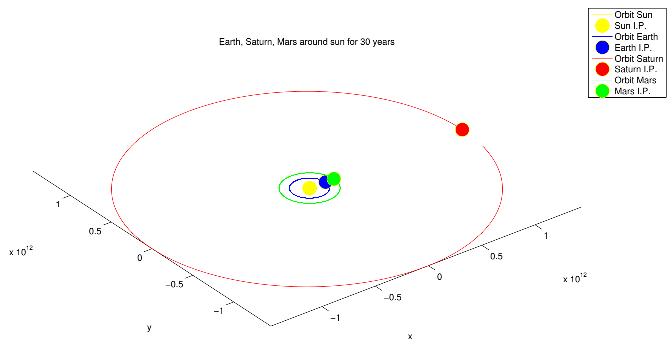
From the matrices `x0` and `v0`, we create a vector of initial positions of the form (for n,p=2): $initvector = [x_1\ y_1\ x_2\ y_2\ v_{x_1}\ v_{y_1}\ v_{x_2}\ v_{y_2}]$. This vector is created in this form in order to insert it as initial position vector in the `ode45` solver. In order to create the right-hand side for the solver, we build a sub-function `f_handle` which computes the total forces on the planets and by using $F = m\alpha$ it computes their accelerations and returns a vector which `ode45` uses as rhs of the system of ODE's. The force acting between two planets i and j has size $\frac{Gm_im_j}{r^2}$, where $r = |x_i - x_j|$ (Euclidean norm) is the distance between the planets and G is the gravitational constant, and direction $\frac{x_j - x_i}{|x_j - x_i|}$, where x_i stands for the vector of positions of planet i. Hence, the vector of the force between the planets is $F_{i,j} = \frac{Gm_im_j}{r^3} \cdot (x_j - x_i)$ for $i \neq j$ and 0 otherwise. Then, using Newton's law $\alpha = F/m$ we compute the accelerations and we store them in a vector of the form $[\alpha_{x_1}\ \alpha_{y_1}\ \alpha_{x_2}\ \alpha_{y_2}]$. Since the acceleration is the second derivative of the position we have to solve a system of $n \times p$ 2^{nd} order ODE's, but since `ode45` solves 1^{st} order ODE's we convert our system to a $2 \times n \times p$ 1^{st} order ODE's. The first $n \times p$ equations refer to the velocities ($x' = v$) while the rest refer to the accelerations ($v' = \alpha$). Finally, the vector that is passed to the `ode45` is of the form $[v_{x_1}\ v_{y_1}\ v_{x_2}\ v_{y_2}\ \alpha_{x_1}\ \alpha_{y_1}\ \alpha_{x_2}\ \alpha_{y_2}]$. After solving this system of ODE's we take the first $n \times p$ columns of the solution array of `ode45` which indicate the positions of the planets and we plot them in 2 or 3 dimensions depending on n.

Discussion & Illustration

In order to test the efficiency of our function, we make several plots in which we used facts about the planets provided by NASA [1]. It is noted that the orbital velocities were used in the `v0` vector. Firstly, we plot in figure 3a the movement of the Earth around the Sun for 1 year. In this figure can be seen the position of the sun, the elliptic orbit of the Earth around the sun, as well as the initial and the final positions of the Earth. It is observed that the final position of the Earth is very close to the initial one, something that verifies the efficiency of the function, since it is known that it takes one year for the Earth to make a full circle around the sun. Next, in figure 3b we plot Earth, Saturn and Mars around the sun for a time period of 30 years. It is known that Saturn needs almost 30 years for a full circle around the sun, something that can be verified by the plot. Furthermore, figure 4a depicts a Sun - Earth - Moon system in 2 dimensions for 1 year. The initial position of the Earth is not visible in the picture since the distance from the Moon is very small compared to the distance from the Sun. In order to observe the orbits closer, we zoom in the picture as it is shown in figure 4b. It is observed that the moon travels around the Earth as it was expected. It is noted that the moon orbits the Earth once every 27 days, hence in the plot this happens almost 13 times, since we plotted for 365 days. Additionally, in figure 5a are shown the orbits of two identical planets, in 2 dimensions, with masses equal to the Sun's mass and orbital velocities of 29700 m/s and -29700 m/s respectively. The movement is plotted for one year, and it is observed that the planets do not "escape" to infinity. In figure 5b can be seen the movement of again 2 identical planets of the same masses, but this time the first planet (red orbit) has velocity -29700 m/s in the y-axis, while the second planet (blue orbit) has the same velocity but only in the z - axis. Hence, the two planets make a movement in 3 dimensions which again is bounded, since they do not escape in infinity even for T equal to 1 year.



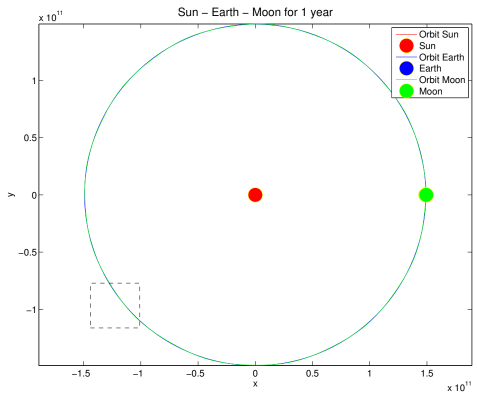
(a) Earth circling around the sun for 1 year.



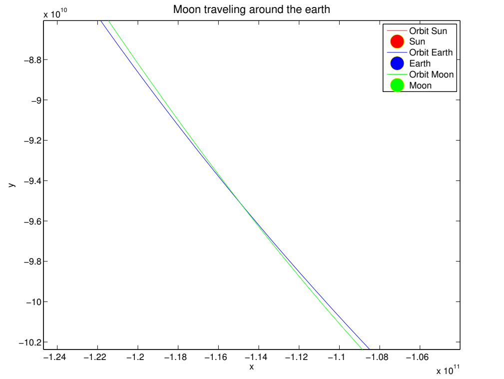
(b) Earth, Saturn, Mars around the sun for 30 years.

Figure 3: (a) Earth circling around the sun for 1 year in 2 dimensions. The green line indicates the orbit of the earth, the blue dot is the Earth's initial position, while the red dot is the Earth's position after 1 year. (b) Earth, Saturn and Mars around the sun for 30 years along with the orbits and the initial positions (I.P.) of the planets. Saturn does 1 full circle around the sun in almost 30 years, something that can be observed in the plot.

[1] https://nssdc.gsfc.nasa.gov/planetary/factsheet/

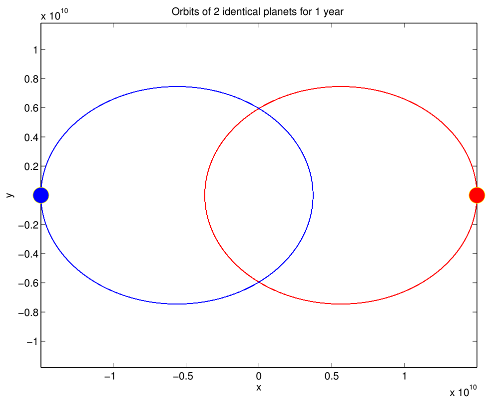


(a) Earth and Moon around the sun for 1 year.

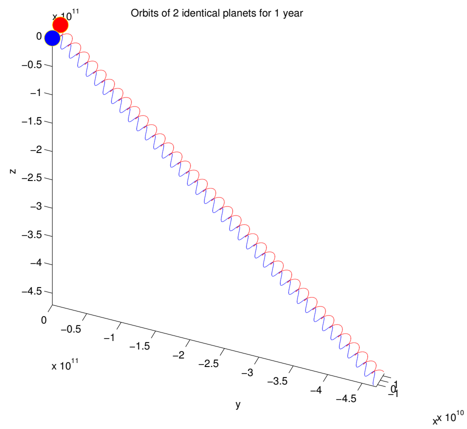


(b) Zoom in the orbits.

Figure 4: (a) Sun-Earth-Moon system in 2D for 1 year. The orbs indicate the initial positions of the planets. The initial positions of Earth and Moon overlap, this is why only the moon is visible. (b) Zoom in the picture to make distinguishible the orbits of Earth and the Moon. It is observed that the Moon goes around the Earth (once every 27 days).



(a) 2 identical planets' orbit for 1 year in 2D.



(b) 2 identical planets' orbit for 1 year.

Figure 5: (a) Orbit in 2 dimensions of 2 identical planets of mass equal to the sun. Velocities in y-axis are 29700 m/s and -29700 m/s for each planet respectively. Distance between each planet $1.496 \cdot 10^{10} m$. *(b) The same identical planets in 3D but now the second planet has 0 velocity in y-axis and -29700 m/s in z-axis. We see that the two planets do not escape in infinity for a period of 1 year.*

Exercise 3

The function

The function `[tvector,dvector,matrix] = Chouliaras_assignment5_exercise3(u0,L,T,S,N,s)` solves the Korteweg-de Vries (KdV) equation which is a non-linear partial differential equation that describes moderately high waves in a swallow channel.
The input parameters are a function handle `u0` with the (almost) periodic initial conditions, the length of the interval `L`, the number of time steps `S` and optionally, the number of spatial discretization points `N` (by default $N = 100$), and an input parameter `s` which controls the speed of the animation that the function produces (by default $s = 1/40$).
The output of the function consists of a vector of times `tvector`, a vector of discretization points `dvector` and a `matrix` which contains the solution at those points in time and space. Additionally the function

produces two pictures, a three-dimensional representation of the solution with respect to time and space and a two-dimensional animation which shows the solution in space through the passing of time.
As an example, in order to solve the KdV with initial conditions $u_0(x) = \mu sin(\pi x/L)$, for $\mu = 2$, $L = 300$, 50 time steps and for length of time $T = 3$ one should type:

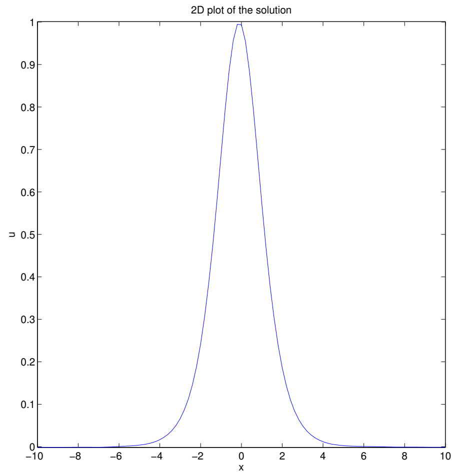
```
[tvector,dvector,matrix] = Chouliaras_assignment5_exercise3(@(x) 2*sin(pi*x/300),300,3,50)
```

The method

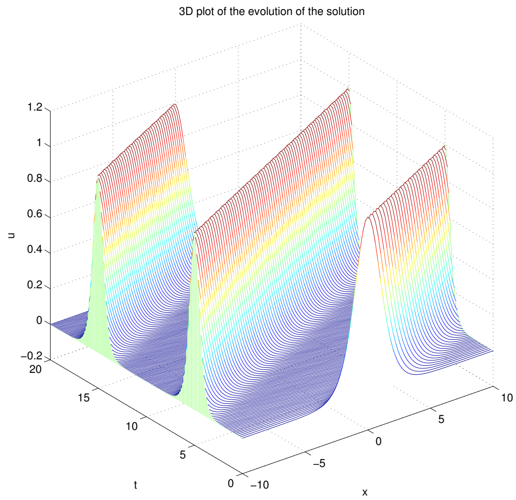
The KdV equation after rescaling is given by $\frac{\partial u}{\partial t} = -\frac{\partial}{\partial x}\left(\frac{\partial^2 u}{\partial x^2} + 3u^2\right)$ where u is the height of the wave above the mean water level and is non-linear. We consider KdV on an interval $x \in [-L, L]$ with periodic boundary conditions which means that $u(t, x + L) = u(t, x - L)$. We first discretize the spatial variable $x = \{x_0 + i\Delta x\}_{i=0}^{N}$ with $x_0 = -L$ and $\Delta x = \frac{2L}{N}$ and we define $u^i = u(i\Delta x)$. Since we consider periodic boundary conditions we know that it holds $u^0 = u^N$ and hence $u^1 = u^{N+1}$. We use centered differences to estimate the partial derivatives as follows: $\frac{\partial u}{\partial t} = \frac{u^{i+1} - u^{i-1}}{2\Delta x}$ and $\frac{\partial^2 u}{\partial t^2} = \frac{u^{i+1} - 2u + u^{i-1}}{(\Delta x)^2}$. Then we define $y^i = \frac{\partial^2 u^i}{\partial x^2} + 3(u^i)^2$ and by taking centered differences we have for $y^i = \frac{u^{i+1} - 2u^i + u^{i-1}}{(\Delta x)^2} + 3(u^i)^2$ $i = 2, 3, \dots, N-1$. For y^1 and y^N we use the fact that $u^0 = u^N$ and $u^1 = u^{N+1}$. Next, we define $f(u) = -\frac{\partial y}{\partial x}$ and hence, the problem of solving this PDE becomes $u' = f(u)$. Using again centered differences we have that $f^i = -\frac{y^{i+1} - y^{i-1}}{2\Delta x}$ for $i = 2, 3, \dots, N-1$. For f^1 and f^N we use the fact that $y^0 = y^N$ and $y^1 = y^{N+1}$, since y is again periodic. Finally, we use `ode45` to solve a system of N ODE's for all the time points $\{i\frac{T}{S}\}_{i=0}^{S}$, where S, T are given as input parameters.

Discussion & Illustration

Concerning the error estimate, we do a discretization in space and we use finite differences to approximate the actual derivatives, for example since we use centered differences for the approximation of the first order derivative we know that the error as a function of the discretization size is $O(h^2)$. In order to check if the function performs correctly, we use several different initial conditions. In figure 6 can be seen the plots produced by the function for initial condition $u_0(x) = \frac{\lambda}{2}\frac{1}{cosh^2(\sqrt{\lambda}x/2)}$ for $\lambda = 2$, $L = 10$, $N = 100$, $S = 50$ and $T = 20$. In figure 6a we see the height of the wave with respect to space. After experimenting with different λ's it was found that a larger λ leads to a larger height for the wave and also to a larger velocity. In figure 6b we can see a three-dimensional plot of the solution with respect to time and space, which represents waves propagating at constant velocity. In case that λ is increased, in order to get again this form of propagation in the 3D picture, the length of time T has to be decreased, since the velocity of the wave is larger.
Figure 7a depicts the solution of the KdV for initial condition $u_0(x) = f(x - \frac{L}{2}; \lambda_1) + f(x + \frac{L}{2}; \lambda_2)$ for $\lambda_1 = 14$, $\lambda_2 = 7$, $L = 5$ and $f(x; \lambda) = \frac{\lambda}{2}\frac{1}{cosh^2(\sqrt{\lambda}x/2)}$. In this case we add two waves (where each one propagates with constant velocity as in figure 6) with different velocities, determined by the λ's and we can see the interaction of the two waves in figure 7a. Figure 7b shows that if we add two traveling waves, then we do not get again a traveling wave, but the interaction between these waves. It is also clear from figure 7b that the second wave has twice the height than the first, since $\lambda_2 = 2\lambda_1$.
In figure 8 can be seen the pictures for initial condition $u_0(x) = \mu sin(\pi x/L)$ for $\mu = 5$, $T = 3$, $S = 200$, $N = 100$ and fairly large interval $L = 100$. Figure 8a shows the solution in space at the time point that the breaking of the wave has started. In this plot we can observe a recurrent behavior. The breaking of the wave becomes more clear in figure 8b where can be seen the evolution of the wave for this time interval. It was observed, that for higher values of μ, u takes higher values too and also the wave breaks sooner. As an example, if we take $\mu = 10$, in order to get a plot similar to figure 8b, we must decrease the time length to $T = 1$. We can see that the observations of these pictures coincide with what is known in paper about these initial conditions, hence the function works appropriately. A last remark is that in order to get an accurate solution, the discretization sizes must be appropriate.

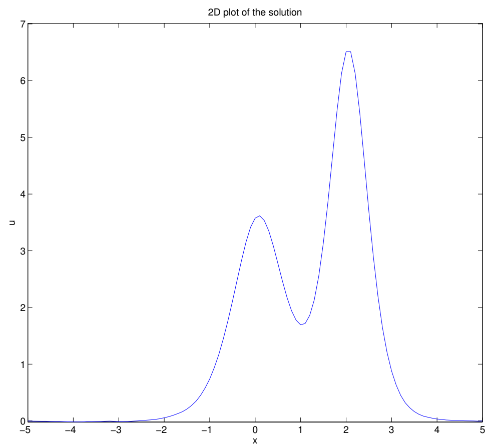


(a) 2D solution in space.

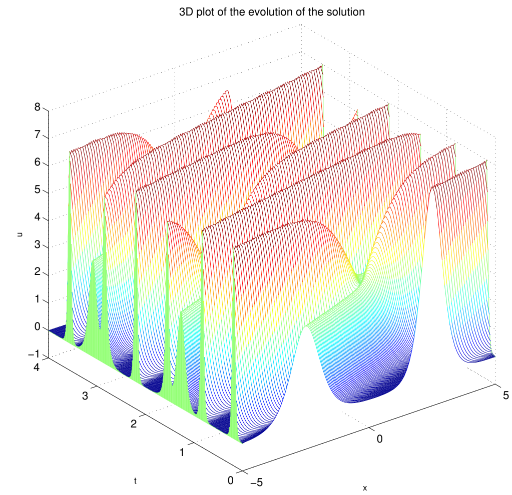


(b) 3D solution in space and time.

Figure 6: (a) 2D solution in space for initial condition $u_0(x) = \frac{\lambda}{2} \frac{1}{cosh^2(\sqrt{\lambda}x/2)}$ *for* $\lambda = 2$, $L = 10$, $T = 20$, $N = 100$, $S = 150$. *(b) 3D solution with respect to space and time which shows waves that propagate at constant velocity.*

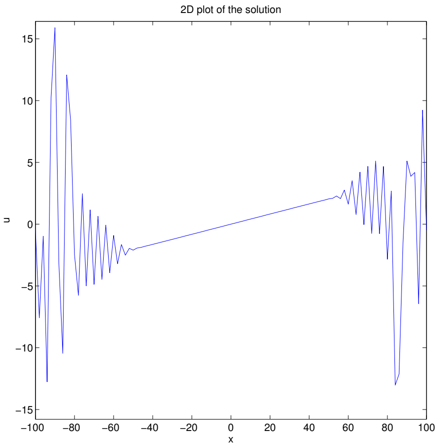


(a) 2D solution in space.

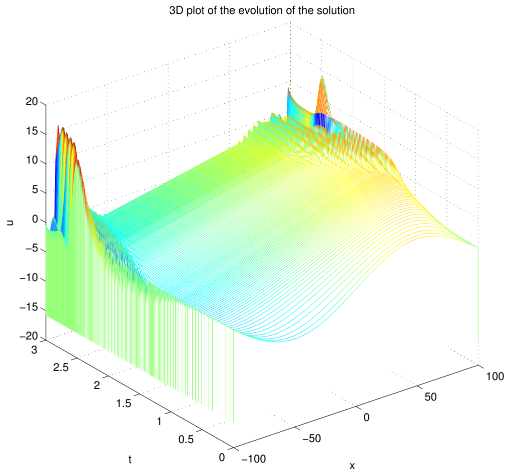


(b) 3D solution in space and time.

Figure 7: (a) 2D solution in space for initial condition $u_0(x) = f(x - \frac{L}{2}; \lambda_1) + f(x + \frac{L}{2}; \lambda_2)$ *for* $\lambda_1 = 14$, $\lambda_2 = 7$, $T = 4$, *on interval* $x \in [-5, 5]$ *and* $f(x; \lambda) = \frac{\lambda}{2} \frac{1}{cosh^2(\sqrt{\lambda}x/2)}$. *(b) 3D solution with respect to space and time which shows the interaction of waves with different velocities.*



(a) 2D solution in space.



(b) 3D solution in space and time.

Figure 8: (a) 2D solution in space for initial condition $u_0(x) = \mu sin(\pi x/L)$ *for* $\mu = 5$, $L = 100$, $N = 100$, $S = 200$ *and* $T = 3$. *In the plot can be seen a recurrent behavior. (b) 3D solution with respect to space and time which shows the breaking of the waves.*