Exercise summary:

- Practice implementation of, and simulations with a modified, nonlinear Crank-Nicolson method in one spatial dimension. The system of evolution equations for the electric field envelope amplitude, derived in this Section, is as follows:

$$E_j^{n+1} - E_j^n = i\delta(\Delta_{ji}E_i^{n+1} + \Delta_{ji}E_i^n) + \left\{\frac{3}{2}N_j^n - \frac{1}{2}N_j^{n-1}\right\},$$

$$\delta = \frac{\Delta z}{4k_0\Delta x^2} \quad , \quad \Delta_{ji}E_i = E_{j-1} - 2E_j + E_{j+1} \quad , \quad N_j^n \equiv \Delta z\mathcal{N}(E_j^n) = i\Delta z k_0 n_2 |E_j^n|^2 E_j^n \ .$$

- Application to simulation of spatial solitons of different orders. Existence of exact solutions offers a conveniet testing opportunity.

0.0.1 Physical backround for spatial optical solitons

The equation to describe spatial soliton beam propagation is:

$$\frac{\partial \mathcal{E}}{\partial z} = \frac{i}{2k_0}\Delta_\perp \mathcal{E} + \frac{i\omega}{2cn(\omega)}\mathcal{P}/\epsilon_0 = \frac{i}{2k_0}\Delta_\perp \mathcal{E} + ik_0 n_2 I_{unit}|\mathcal{E}|^2\mathcal{E}$$

It represents a beam propagating in one transverse dimension, and subject to self-focusing nonlinearity due to optical Kerr effect. Single transverse dimension is an idealization, of course. In reality, the beam would be confined to a planar waveguide, and $n(\omega)$ would be replaced by the effective index of the fundamental mode — effective index approximation then results in the above propagation equation. This dimensional reduction will be discussed later in the course. For the moment we accept this model as given. Specifically for one transverse dimension, solutions exists in which the self-focusing effects are balanced by diffraction. This balance is perfectly steady for the so-called fundamental soliton, and it is "dynamic" for higher-order (which means higher energy or power in the beam) solutions. Higher order solitons have the same spatial profile with the fundamental one at periodic propagation distances. This is very convenient for our purposes, because it allows us to initialize higher-order nonlinear solutions without explicit use of analytic formulas.

Fundamental solution

Solution for the fundamental soliton can be obtained easily from the following Ansatz

$$\mathcal{E} = A_0 \frac{1}{\cosh(x/w)} \exp\left(i\beta z\right) \ .$$

Insert this into the propagation equation, and cancel some nonzero common terms that occur on both sides, to get

$$\beta = \frac{1}{2w^2k_0} - \frac{1}{k_0w^2}\frac{1}{\cosh^2(x/w)} + k_0 n_2 I_{unit} A_0^2 \frac{1}{\cosh^2(x/w)} \ .$$

Once again, this is nothing but a solvability condition that determines the propagation constant and thus chromatic dispersion properties of the soliton. For this to hold as an identity, both β, and A_0

must be fixed such that the propagation constant equals the first term on the right, and the two x-dependent terms cancel each other. The latter condition relates the intensity to the spatial width of the soliton:

$$k_0^2 n_2 I_{unit} A_0^2 w^2 = 1 \ .$$

Since we have two parameters, A_0 and w, only constrained by one relation, there is a whole family of fundamental solitons. The wider is the beam, the lower intensity is sufficient to hold it together and compensate the diffraction.

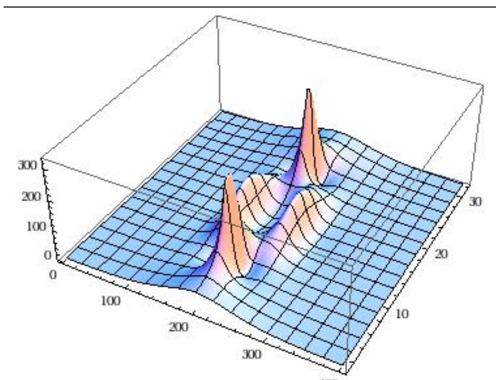
This is a special case of fundamental (spatial) soliton. It provides a convenient way to test numerical implementation of the modified C-N method. One question is whether numerics can "preserve" an initial condition which coincides with this exact solution. Another question can be what happens if the initial beam has a bell shape but is not exactly the same as the fundamental soliton. We will see that the soliton solution is rather robust: It will emerge from the initial condition which will shed the excessive energy (beam power) and adjust its shape to $\mathrm{sech}(x/w)$. For the simulation to be able to handle such a dynamics, transparent boundary conditions must be used. The radiation shed by the to-be soliton will disappear through the computational domain edge, leaving us with a properly formed soliton.

Higher-order solutions

Higher-order solitons can be initialized in a very similar way, because at certain distances, when $betaz = \pi/4$, they assume the same spatial shape as the fundamental one. We can simply start simulation from this special position along the z axis. However, one must adjust the relation between the beam width w and its initial intensity A_0^2 to

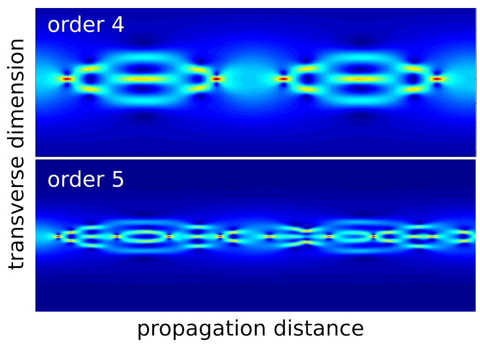
$$k_0^2 n_2 I_{unit} A_0^2 w^2 = N^2 \ .$$

Here N stands for the soliton number. If it happens to be an integer, a periodically repeating solution will appear. This is yet another opportunity to put our nonlinear C-N implementation to a good-strength test: Even without knowing the explicit functional form of the higher-order soliton, one can easily tell if the numerical solution does what it should simply by inspecting if it creates a nice periodic-in-z intensity pattern. The following is an example of the intensity profile in a third-order spatial soliton:



Spatial profile of the intensity in a third-order spatial soliton. With the initial cross-section is identical to that of the fundamental soliton, the evolution along the propagation distance is shown over a single period. The periodicity of a numerical solution offers a convenient test — inaccuracies in the integration of the evolution equation accumulate until a departure from the strict periodicity becomes obvious after a few soliton periods. This trend is more obvious in higher-order solitons.

Of course, higher order soliton solutions are more difficult to simulate. Since they have more complex profiles and richer dynamics, they require both finer spatial resolution and shorter integration step. As a result, the computational effort needed to obtain even a couple of periods of a higher-order soliton with a reasonable quality may be several times bigger than that required for a lower-order soliton. The following figure illustrates these issues:



Numerical simulation of higher-order solitons. The upper panel shows the evolution of the intensity in the fourth-order solution. The result clearly shows the periodic structure, and only has minor deviations from the strict periodicity (visible as a slight asymmetry in the middle of the panel). The lower panel shows an attempt to simulate the soliton of order five. While the numerical solution exhibits the characteristic features with multiple symmetric intensity peaks across the transverse dimension, it is clearly not periodic along z. Higher resolution and a finer integration step would be necessary to correct the problem.

Propagation at angle

If one adds a linear phase-shift to the initial condition spatial profile, the outcome of the evolution is still a soliton, but one that propagates at an angle proportional to its phase-front tilt. This is expressed through the formula

$$\mathcal{E} = A_0 \frac{1}{\cosh\left(\frac{x - z k_x/k_0}{w}\right)} \exp\left(i k_x x\right) \exp\left(i \beta z\right) \quad , \quad \beta = \frac{1}{2k_0 w^2} - \frac{k_x^2}{2k_0} \ ,$$

where the first term describes the envelope moving with a constant transverse "velocity," and the last expression show the change in the propagation constant.

Such solutions can be used to "probe" boundary conditions, as the fundamental soliton provides a non-diffracting beam. This results in an interesting situation, because the Hadley's ABCs rely on "free" propagation of waves in the vicinity of the boundary. Almost by definition, waves that constitute a soliton can not be propagating as free, because the nonlinear interaction is strong enough to cancel diffraction. So it is quite natural to expect that performance of transparent boundary conditions will be negatively affected by the nonlinear interactions.

0.0.2 Numerical simulation of spatial optical solitons

Task 1: Simulator modification

Starting from the Crank-Nicolson method simulator for one transverse dimension with absorbing boundary conditions developed in previous exercises, modify it to include Kerr-effect nonlinearity. Make sure that boundary conditions work properly before proceeding to the next step. This can be done easily with the nonlinear index set to zero and propagating for a sufficiently long distance.

Solution:
Instructor's solution: *CrankNicolson.m, SpatialSoliton.m , Parameters0.m, Parameters1.m,*

The summary of the evolution equations shown at the beginning of this exercise indicates that the nonlinear term evaluation requires to keep in the memory an auxiliary snapshots of the field amplitude from the previous integration step. It is denoted E_{oldold} in the following listing:

Listing 1: Modification of the main loop

```
for m=1:M
    z = z + dz;
    Enew    = CrankNicolson(dx, k0, Eold, Eoldold, NX, dz, nlcoeff);
    Eoldold = Eold;
    Eold    = Enew;
 end
```

where we have modified the call to the Crank-Nicolson routine to make room for the additional, older amplitude array. The modification inside this function is a very simple single line where the nonlinear term is added to the previously prepared right-hand-side, R, of the C-N linear system:

Listing 2: Modification of the solver

```
for row=1:n
  R(row) = R(row) +  nlcoef*3/2*Eold(row)*abs(Eold(row))^2
                  -  nlcoef*1/2*Eoldold(row)*abs(Eoldold(row))^2;
end
```

Task 2: Fundamental soliton demonstration
Set up initial condition (and other parameters) to demonstrate the fundamental spatial soliton. Specifically, show that your simulation preserves the non-diffracting shape of the fundamental soliton. Experiment with various simulation parameters. In particular, find out how big the computational box must be in order to support soliton propagation for a significant distance before artefacts caused by domain boundaries show up.
Solution:
Instructor's solution: *CrankNicolson.m, SpatialSoliton.m, Parameters1.m*

Task 3: Propagation at an angle
A more stringent test in the nonlinear regime requires propagation of a soliton beam at an angle, because such a simulation must reproduce "an evolving solution," and in particular the running phase of the beam. Implement a spatial soliton beam function, using the formula given above, and verify the correct function of you solver against this exact solution.
Solution:
Instructor's solution: *Parameters_angle.m*

Note that parameters set in instructor's solutions *do not* result in very accurate agreement between the final numerical and analytic solutions. It is left to the reader to explore the influence of various settings that allow to achieve better accuracy.

Task 3: Higher-order solitons
Demonstrate simulation of the second- and third-order solitons. Attempt simulation of yet higher orders, paying attention to increasing demands for resolution in both dimensions.
Solution:
Instructor's solution: *Parameters2.m, Parameters3.m*

Task 4: Soliton from an arbitrary initial condition
Starting from a non-soliton initial condition, such as a Gaussian beam, show that if the energy is sufficient for a fundamental soliton, the latter will be formed after excess energy is shed and subsequently absorbed in the transparent boundary.
Solution:
Instructor's solution: *Parameters_GaussIC.m*

Task 5: Instability at long integration steps
Nonlinearly modified C-N method is not unconditionally stable anymore. Set up a simulation to demonstrate that by increasing the integration step instability can be induced.
Solution:
Instructor's solution: *Parameters_G_unstable.m*

Task 6: Boundary conditions and nonlinearity
Test the boundary condition implementation by bouncing a fundamental soliton propagating at an angle from the domain boundary. Note that the reflectivity increases, and that it is not too difficult to find a regime in which the absorbing boundary conditions fail.
Solution:
Instructor's solution: *Parameters_angle.m*