

Peter Junglas

Transparent boundary conditions for simulation programs

The Problem

A lot of simulation programs are available today as invaluable tools for teaching physics covering many different application areas ([2],[5]). With the tremendous increase of available computer power on the students' desks even the solution of partial differential equations for realtime animations is now possible, at least for one- and twodimensional systems. Important examples are ripple tank simulations solving the wave equation [5] or quantum mechanical applications solving the Schrödinger equation [6].

All these programs have to cope with the problem of boundary conditions: The simple Dirichlet condition (i.e. fixing the wave function) creates hard walls which reflect all incoming waves back into the simulated region. They interfere with the outgoing waves and create complicated patterns, which are difficult to interpret (cf. fig. 0.1). Using von Neumann conditions (i.e. fixing the derivative of the wave function) doesn't help at all, it simply creates reflections at a free end.

What one needs are boundaries that are "transparent" for outgoing waves so that the simple simulated systems will not be disturbed by incoming reflected waves. Surrounding the system by a large empty boundary region is usually not an option, since generally the simulation already needs all available computing power. The creation of transparent boundary conditions is an amazingly complex task that has been under active research for long and many different solutions have been proposed [1]. In the following we will show how one of them (the "perfectly matched layer") can be incorporated into standard simulation programs.

Wave equation - Poor man's solution

A simple, physically motivated idea to get rid of boundary reflections is to surround the area of interest with a layer of absorbing material. A straightforward implementation could be to change the wave equation

$$\partial_t^2 u = \partial_x^2 u$$

by adding a laminar friction term

$$\partial_t^2 u + \tau \partial_t u = \partial_x^2 u$$

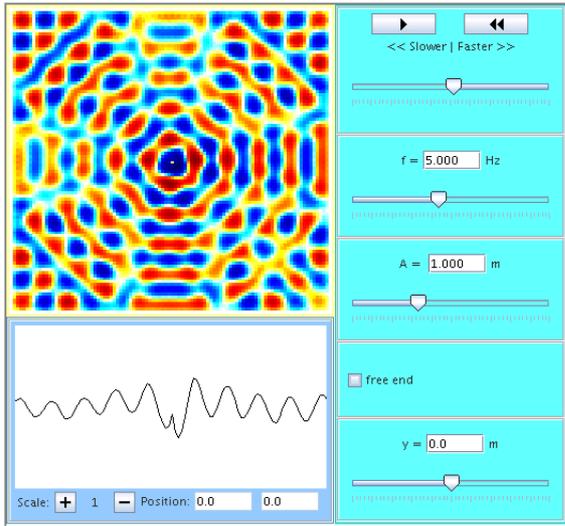


Figure 0.1: Reflected waves

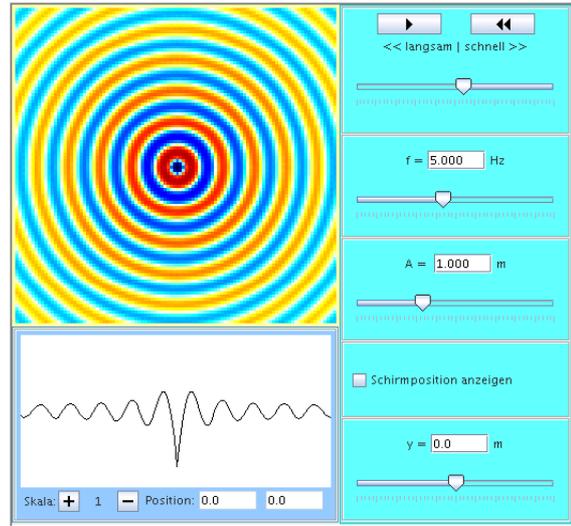


Figure 0.2: Waves with transparent boundaries

where the damping $\tau(x)$ is 0 inside the region and increasing in a boundary layer. Unfortunately this layer still creates reflecting waves, but by a careful choice of $\tau(x)$ and a sufficiently large layer they can be made small enough, at least for teaching purposes.

Incorporating this idea in standard ripple tank simulations leads to impressive results, e.g. the simple circle wave simulation, which doesn't show visible reflections over a wide range of parameters (fig. 0.2). More than twenty applets have been constructed that use the ripple tank with absorbing boundaries to illustrate different wave phenomena. On the other hand the strength of the reflections depends on the incidence angle and increases with the wave length, which can lead to unphysical results (fig 0.3). Therefore a better solution is needed in such cases.

Schrödinger equation - The Perfectly Matched Layer

The Schrödinger equation (in suitably chosen units) is given by

$$i\partial_t\psi = \left(-\frac{1}{2}\partial_x^2 + V(x)\right)\psi$$

where $V(x)$ defines the potential energy function. It is assumed to be 0 in the following although a non-vanishing potential can be easily incorporated as long as it is constant outside the simulated region. The free Schrödinger equation admits plane wave solutions

$$\psi = e^{ikx - i\omega t}$$

for arbitrary $\omega > 0$ and

$$k = \sqrt{2\omega}$$

A simple absorbing layer is not sufficient here to reduce the reflections to a reasonably low level. Instead the idea of the “perfectly matched layer” (PML), first introduced by Berenger for the Maxwell equation [3], is to design a special material with unphysical properties which absorbs incoming waves without creating any reflections. Alternatively it can be formulated as a coordinate transformation into the complex plane [4], an approach that will be adopted here.

For simplicity we will assume that the boundary is at $x = 0$, the interior region being $x < 0$. Then we introduce a new coordinate \tilde{x} by

$$\tilde{x} = \begin{cases} x & | \quad x \leq 0 \\ x(1 + i\sigma) & | \quad x > 0 \end{cases}$$

and define the wave function for this coordinate as

$$\tilde{\psi}(x) := \psi(\tilde{x})$$

The Schrödinger equation then transforms into

$$i\partial_t \tilde{\psi} = -\frac{1}{2} \frac{1}{(1 + i\sigma)^2} \partial_x^2 \tilde{\psi}$$

This equation has plane wave solutions with

$$k = (1 + i\sigma)\sqrt{2\omega}$$

which decrease exponentially in the boundary $x > 0$.

Unfortunately the boundary leads to reflections unless $\sigma = 0$. But this can be easily handled by making σ a function $\sigma(x)$ with $\sigma(0) = 0$, increasing strictly with x . The new coordinate is now defined by

$$\tilde{x} = \begin{cases} x & | \quad x \leq 0 \\ x + i \int_0^x \sigma(x') dx' & | \quad x > 0 \end{cases}$$

and the Schrödinger equation becomes

$$i\partial_t \tilde{\psi} = -\frac{1}{2} \frac{1}{1 + i\sigma} \partial_x \left(\frac{1}{1 + i\sigma} \partial_x \tilde{\psi} \right)$$

Now we get exponentially decreasing waves without any reflections at the boundary.

The same idea has been used for many different equations, among them the wave equation. Since it is second order in time, the details there are a bit more complicated.

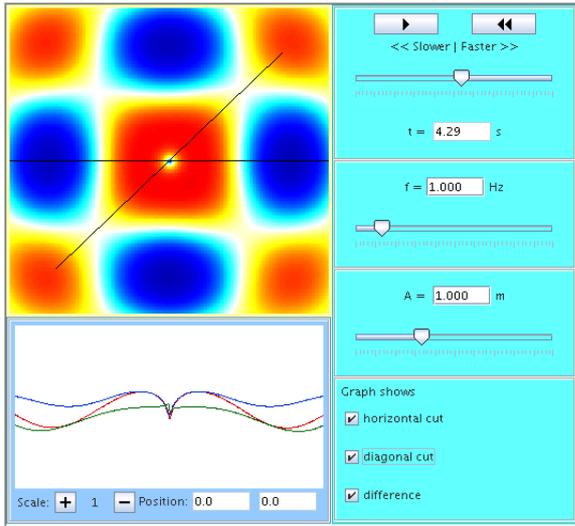


Figure 0.3: Residual reflections

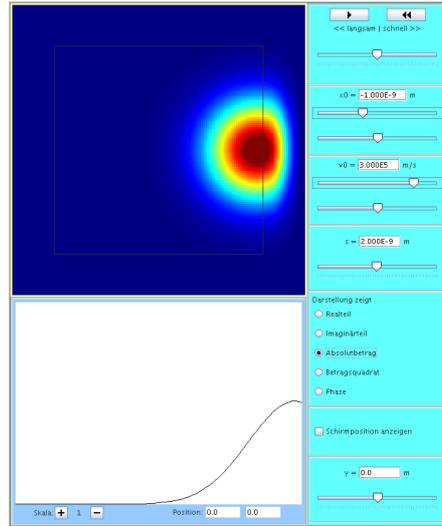


Figure 0.4: Schrödinger wave packet

Implementation - Nothing is simple

The numerical solution of the Schrödinger equation used here is based on the method from [7]. It starts with a discretisation of ∂_x^2 to the 4th order given by

$$f''(x) = -\frac{1}{12(\Delta x)^2} (f(x + 2\Delta x) - 16f(x + \Delta x) + 30f(x) - 16f(x - \Delta x) + f(x - 2\Delta x))$$

For the PML-enhanced Schrödinger equation one needs instead a 4th order discretisation of

$$(a(x)f'(x))' \quad \text{with} \quad a(x) := \frac{1}{1 + i\sigma}$$

Using a 4th order formula for f' twice leads to terms $f(x + 4\Delta x)$ which makes the algorithm very unwieldy and slow, so one needs a 4th order discretisation which extends only to $x + 2\Delta x$.

To find such a formula a general ansatz containing 25 terms has been reduced to 13 terms by requiring symmetry under $\Delta x \rightarrow -\Delta x$. Using a computer algebra system this has been expanded into a Taylor series and compared to the desired result, which leads to a singular linear system for the coefficients. A systematical scan of its solution space with Matlab

resulted in 15 shortest formulae with 12 terms each, the one used here is

$$\begin{aligned} \partial_x(a\partial_x f) = & -\frac{1}{48(\Delta x)^2} \left\{ (3a(x+2\Delta x) + a(x-2\Delta x)) f(x+2\Delta x) \right. \\ & - 16(3a(x+\Delta x) + a(x-\Delta x)) f(x+\Delta x) \\ & - 4(a(x+2\Delta x) - 16a(x+\Delta x) - 16a(x-\Delta x) + a(x-2\Delta x)) f(x) \\ & - 16(a(x+\Delta x) + 3a(x-\Delta x)) f(x-\Delta x) \\ & \left. + (a(x+2\Delta x) + 3a(x-2\Delta x)) f(x-2\Delta x) \right\} + O(\Delta x^5) \end{aligned}$$

This complicated formula has an obvious impact on the performance of the simulation program. Even worse is the fact that the coefficients $a(x+n\Delta x)$ are depending on x , which complicates subsequent computations in the DeRaedt algorithm. And finally the number of grid points has to be enlarged considerably (for a 2d simulation generally by a factor of 2) to make room for boundary layers that sufficiently suppress all reflections.

Consequently a first implementation was slower by a factor of 22 than the corresponding program without PML, even though caching of precalculated values had been used. After extensive measures to increase the performance, among them introducing a fast library for submatrix computations and optimisation of memory accesses, the PML version is slower by a factor of 6.5, where a factor of 2 is due already to the larger grid. The computing power of a current standard PC allows for animations with 4 - 5 frames per second using a 140x140 grid (inner region: 100x100), which is at the lower edge of a movie perception, but still makes it a useful tool for virtual experiments.

The simulation program `FreeParticle` uses the PML-enhanced solver to show the time evolution of a Gaussian wave packet (fig 0.4). With default parameters the packet leaves the simulation area without visible reflections, they can be seen only by largely increasing the scale of the plot. A wave function with lower velocity has a larger wavelength, which leads to stronger reflections. When the wave length exceeds the width of the boundary layer (e.g. for $v_0 = 1.0 \cdot 10^5$), the reflections are directly visible in the wave display.

Conclusions

The PML-enhanced Schrödinger solver presented here is part of the open source PhysBeans library [5] and has been used in several simulations, e.g.

a free particle, a particle running against a slit or double slit, a particle in a potential box, a potential barrier or a two-dimensional Coulomb potential. It could easily be extended to include electric or magnetic fields, particle spin or 1d two-particle systems. This would allow to show such interesting phenomena as the Aharonov-Bohm effect, quantum computing or the EPR paradoxon.

Limiting factor at the moment is the speed of the resulting simulation programs, which is now at the lower edge and will get worse by including more degrees of freedom or external fields. Though a lot of work has been done to increase the performance, there is still potential for further improvement by parallelisation to use more CPUs, faster algorithms or general advance of CPU performance.

Bibliography

- [1] **Antoine, X. et. al.:** *A Review of Transparent and Artificial Boundary Conditions Techniques for Linear and Nonlinear Schrödinger Equations.* Commun.Comput. Phys. **4**, 729–796 (2008).
- [2] **Belloni, M.; Christian, W.; Cox, A. J.:** *Physlet Quantum Physics.* Pearson Prentice Hall, Upper Saddle River (2006).
- [3] **Berenger, J.-P.:** *A Perfectly Matched Layer for the Absorption of Electromagnetic Waves.* J. Comput. Phys. **114**, 185–200 (1994).
- [4] **Chew, W. C. and Weedon, W. H.:** *A 3D perfectly matched medium from modified maxwell's equations with stretched coordinates.* Micro. Opt. Lett. **7** (13), 599–604 (1994).
- [5] **Junglas, P.:** *cliXX PhysBeans.* Verlag Harri Deutsch, Frankfurt (2008).
- [6] **Junglas, P.:** *Interaktive Simulationsprogramme zur Lösung der Schrödingergleichung.* Wismarer Frege-Reihe **3**, 13–18, (2009).
- [7] **De Raedt, H.; Michielsen, K.:** *Algorithm to solve the time-dependent schrödinger equation for a charged particle in an inhomogeneous magnetic field: Application to the aharonov-bohm effect.* Comp. Phys. **8**, 600-607 (1994).

Autor

Prof. Dr. rer. nat. Peter Junglas

Private Fachhochschule für Wirtschaft und Technik Vechta/Diepholz/Oldenburg

Schlesierstraße 13a

D-49356 Diepholz

E-Mail: peter@peter-junglas.de