Attempts to Improve Parabolic Equation Solver

Leo Deer †

Abstract

This article is about one project I worked on with the Radar Echo Telescope collaboration, an ultra-high energy neutrino research, while at OSU. My job was to improve the mathematical foundations of a numeric solver used to simulate an experiment. Here I go over approaches to improve the solver and how I implemented and tested if they worked.

1 Intro to research

The Radar Echo Telescope (RET) is a collaboration with the aim of detecting ultra-high energy neutrinos using the radar echo method. More information on the collaboration may be found here. The collaboration needs to run many simulations of antennas sending and receiving signals in Antarctica in order to plan and test large-scale experiments before deployment in Antarctica. To aid these simulations they developed ParaProp, a parabolic equation[1] EM solver which is used to simulate the electric field transmitted from antennas. ParaProp uses the split-step method to solve the parabolic equation because it is computationally fast but introduces error into the solution making it less accurate than other numeric solutions to electric fields, for example, finite difference time domain[2]. So, my job was to improve the accuracy of ParaProp by improving the approximations used when implementing the split-step method. My work is based on the paper [PSA+21], authored by the head of the collaboration and my research advisor, which derives how ParaProp uses the split-step method when solving the parabolic equation. This was a theory-based project, so it revolves around lots of math, most of which builds upon the work in this paper [PSA+21]. So, instead of rewriting what is already in that paper, I will cite it and the relevant equations when necessary.

2 Changing order of operations

ParaProp uses the "split-step" method to solve the parabolic equation which solves for the fields. It is done by continually taking forward and backward Fourier Transforms. Here is a Wikipedia that explains more about it if you're curious. Appendix A of [PSA+21] contained an error when implementing the split step method to the parabolic approximation which was my first idea on how to improve ParaProp. I will explain it and how I corrected it.

The beginning of the derivation sets up and manipulates the parabolic equation into a form we can use. I'll skip over that and use Eq. A8 from Appendix A in [PSA+21] which shows the factored parabolic equation as a starting place.

$$\partial_x u_\pm = -ik_0(1-\mathbf{Q})u_\pm \tag{A8}$$

$u_\pm$ is the ansatz that represents the forward and backward propagating reduced fields and $Q = \sqrt{\partial_{zz}/k_0^2 + n^2}$ is a pseudo-differential operator and n is the index of refraction of the ice the radio signal will propagate through. Eq. A9 from Appendix A gives the formal solution of the forward propagating field.

$$u(x) = e^{-ik_0(1-\mathbf{Q})} \tag{A9}$$

†leo99deer@gmail.com

[1] The parabolic equation is an approximation to the wave equation.

[2] Throughout my research I used Meep a finite difference time domain solver available as a Python Library. More information on it may be found here.

From this,

$$u(x+\Delta x) = e^{-ik_0\Delta x(1-\mathbf{Q})}u(x) \tag{A10}$$

This only relies on the previous solution $u(x)$ which means the solution may be marched along in code. But because we do not know how to evaluate the $\mathbf{Q}$ operator this doesn't solve the problem. We will come back to this expression but for now, we will continue the derivation in [PSA+21]. Consider again the expression in Eq. A8 except we will expand the $\mathbf{Q}$ operator to lowest order by Taylor expanding in powers of the quantity in parenthesis $\mathbf{Q} = \sqrt{1 + (\partial_{zz}/k_0^2 + n^2 - 1)}$. This leads to

$$\partial_{zz}u + 2ik_0\partial_x u + k_0^2(n^2-1)u = 0 \tag{A11}$$

The paper implements the split-step method here but incorrectly takes the z space Fourier Transform of this expression[3] It neglects the $z-$dependence on n.

$$\partial_x U = \frac{ik_0}{2}\left[(n^2-1) - \frac{k_z^2}{k_0^2}\right]U \tag{INCORRECT A14}$$

The expression should be

$$\partial_x U = -\frac{ik_0}{2}\left[1 + \frac{k_z^2}{k_0^2}\right]U + \frac{ik_0}{2}\mathcal{F}(n^2 u) \tag{CORRECT A14}$$

But because we are trying to solve for u, it is unknown and we cannot evaluate this expression anymore.

In my solution, I went to the formal solution of Eq. A10. I split the $\mathbf{Q}$ operator into a refractive part $\mathbf{A}$ and a diffractive part $\mathbf{B}$. For now, we use the same first-order Taylor expansion of $\mathbf{Q}$ as we did in Eq. A11. meaning $\mathbf{A} = \frac{1}{2}(1-n^2)$ and $\mathbf{B} = -\frac{\partial_{zz}}{2k_0^2}$. Now Eq. A10 becomes[4]

$$u(x+\Delta x) = e^{-ik_0\Delta x(\mathbf{A}+\mathbf{B})}u(x) = e^{-ik_0\Delta x\mathbf{A}}e^{-ik_0\Delta x\mathbf{B}}u(x) \tag{1}$$

Now, this is an expression suitable to solve using the split-step method. We will apply a forward Fourier Transform

$$U(x+\Delta x, k_z) = e^{-ik_0\Delta x\tilde{\mathbf{B}}}\mathcal{F}\left(e^{-ik_0\Delta x\mathbf{A}}u(x)\right) \tag{2}$$

And we now apply a backward Fourier Transform

$$u(x+\Delta x) = \mathcal{F}^{-1}\left(e^{-ik_0\Delta x\tilde{\mathbf{B}}}\mathcal{F}\left(e^{-ik_0\Delta x\mathbf{A}}u(x)\right)\right) \tag{3}$$

and for our specific $\mathbf{A}$ and $\mathbf{B}$ we get

$$u(x+\Delta x) = \mathcal{F}^{-1}\left(\exp\left[-\frac{ik_0\Delta x k_z^2}{2k_0^2}\right]\mathcal{F}\left(\exp\left[\frac{ik_0\Delta x}{2}(n^2-1)\right]u(x)\right)\right) \tag{4}$$

This is similar to Eq. A17 in the appendix of [PSA+21], except in Eq. A17 the $\exp\left[\frac{ik_0\Delta x}{2}(n^2-1)\right]$ term is outside the Fourier Transforms. So, the error was in the order of operations of the split-step method. I also used a suitable $\mathbf{A}$ and $\mathbf{B}$ from the $\mathbf{Q}_{ice}$ expression in Eq. A13 to get an analogous expression to Eq. A18

$$u(x+\Delta x) = \mathcal{F}^{-1}\left(\exp\left[ik_0\Delta x\left(n\sqrt{1+\frac{1}{n_0^2}} - \sqrt{1+\frac{n^2}{n_0^2}}\right)\right]\mathcal{F}\left(\exp\left[-ik_0\Delta x\sqrt{1-\frac{k_z^2}{k_0^2}}+1\right]u(x)\right)\right) \tag{5}$$

Next, I modified the ParaProp code. This new version is called `paraPropPythonNew.py`. Using `DipoleComparison.ipynb` I compared the results of the a new ParaProp with the corrected order of operations with the old ParaProp. I found that the differences in simulated electric fields were negligible compared to the difference with a simulation using finite difference time domain software see Figure 1.

[3] This step hinges on the Fourier identities $\mathcal{F}(\partial_{zz}u) = -k_z^2\mathcal{F}(u)$ and $\mathcal{F}(\partial_x u) = \partial_x\mathcal{F}(u)$

[4] These are only actually only approximately equal because $\mathbf{A}$ and $\mathbf{B}$ do not commute

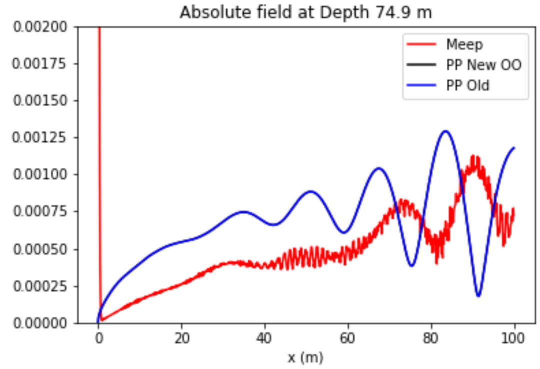


Figure 1: This compares the E-fields from simulated transmission dipoles in the Antarctic ice using ParaProp with the old and new order of operations and Meep, a finite difference time domain software. The field is measured at a depth of 74.9m and plotted along the radial direction (x).

Thus, I conclude that the order of operations error is not the dominant error of ParaProp.

3 Improving the approximation on Q

Next, I turned my focus to the pseudo-differential operator $\mathbf{Q}$. As shown above, it is neccessary to split $\mathbf{Q}$ into a refractive part $\mathbf{A}$ and differential part $\mathbf{B}$. In [PSA+21] an improved approximation to $\mathbf{Q}$ is given in Eq. A13. I used the same methodology as the author to try and find a better approximation of $\mathbf{Q}$. The derivation for the better $\mathbf{Q}$ operator is not in [PSA+21], so I will go over the background in the appendix. Here I will start here with this expression.

$$\mathbf{Q'} = \sqrt{1 + \frac{\partial_{zz}}{k_0^2}} + \sqrt{n^2 - 1 + \mathbf{Y}^2} - \mathbf{Y} \tag{6}$$

This is split into a refractive and differential part and leaves $\mathbf{Y}$ as an unknown function of z which is suppose to match the action of $\sqrt{1 + \partial_{zz}/k_0^2}$ on our reduced function u. The way we did this was to make a guess for u and evaluate $\partial_{zz}(u)/k_0^2$ based on that guess. Then solve for $\mathbf{y}$ in $\partial_{zz}(u)/k_0^2 = \mathbf{y}u$ which we use to get $\mathbf{Y} = \sqrt{1+\mathbf{y}}$. [PSA+21] guessed $u = \exp[ik(z)z]$ and got $\mathbf{Y} = \sqrt{1 + n(z)^2/n_0^2}$ where n_0 is a reference index of refraction of the ice taken at a reference depth[5]. There are a few mistakes with this expression based on the method I described above.

3.1 Recalculating $\partial_{zz}(u)/k_0^2$

The original expression used for $\mathbf{Y}$ seems to neglect the $z-$dependence on $k(z)$. One should actually find

$$\mathbf{Y} = \sqrt{1 + \left(-[n + \partial_z(n)z]^2 + i\left[2\frac{\partial_z(n)}{k_0} + \frac{\partial_{zz}(n)z}{k_0}\right]\right)} \tag{7}$$

When I implemented both this $\mathbf{Y}$ operator and that used in [PSA+21] and compared[6] both with a finite difference time domain simulation, I found that Eq. 7 reduced the error in E-field in certain regions, but did not provide an overall better approximation than that used in [PSA+21].

[5]For now it is set to the depth of the detectors receiving the signal of our transmitting antenna.

[6]The analysis for this work was done on the Ohio Supercomputer which I had access to as a student researcher at OSU. I'm making these write-ups after my graduation based on the material I saved locally on my machine. Most of it is here, although certain plots, like the ones for this, were not saved. I am unable to recreate those plots because re-running the simulations requires more computing power than my machine can handle.

3.2 New guess for u

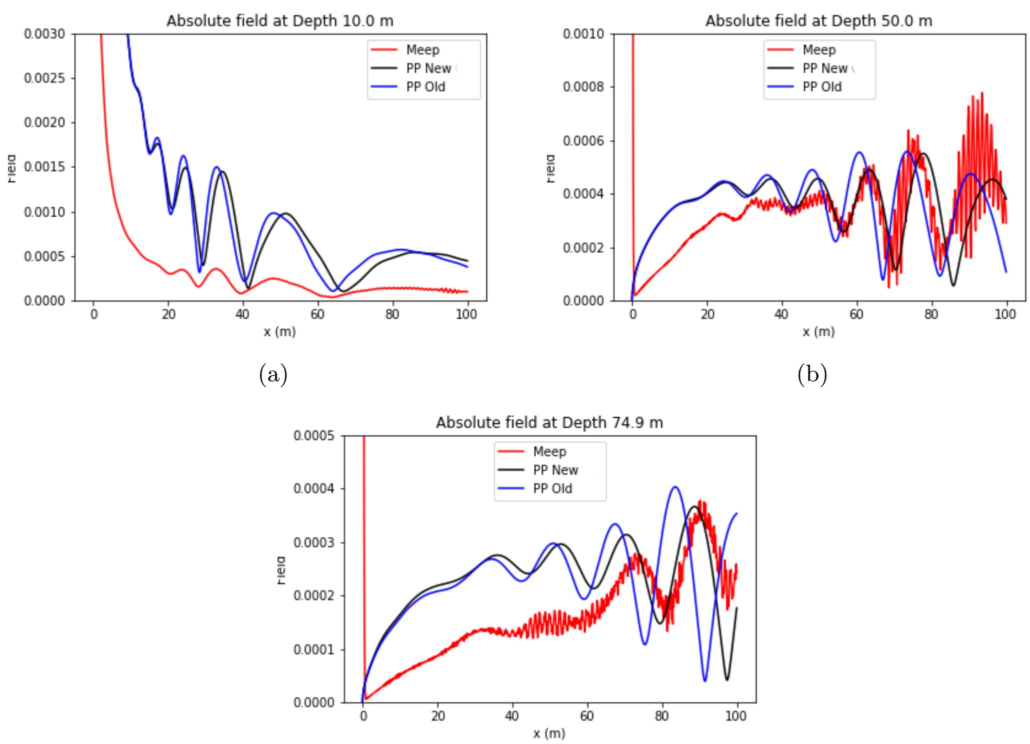
Next, I tried a guessing a gaussian, $u = \exp\left[\frac{-k(z)^2(z-z_0)^2}{2}\right]$ where z_0 is the reference depth used in n_0. I found

$$\mathbf{Y} = \sqrt{1 + \Big([k(z)(z-z_0)(\partial_z(n)(z-z_0)+n)]^2 - [(\partial_{zz}(n)(z-z_0))^2 + n\partial_{zz}(n)(z-z_0)^2 + 4n\partial_z(n)(z-z_0) + n^2] \Big)} \tag{8}$$

But as with Eq. 7 this new $\mathbf{Y}$ reduced the error of the E-field only in certain regions but did not improve the overall better approximation.

3.3 Investigating the purpose of n_0

It seems that the two expressions above did not provide a significant advantage which led me to return to the expression for $\mathbf{Y}$ used in [PSA+21]. Although, in the derivation of Eq. 7 and Eq. 8, n_0 did not arise from the math. It was not clear in the notes of the derivation of $\mathbf{Q}_{ice}$ from [PSA+21] whether n_0 was put there on purpose or by mathematical error, so I investigated if it provides an advantage. I simulated the E-fields from ParaProp using $\mathbf{Q}$ operators with and without n_0 and compared those with E-fields found using finite difference time domain software see Figure 2.



(a)

(b)

(c)

Figure 2: This compares the E-fields from simulated transmission dipoles in the Antarctic ice using the old paraProp with n_0 and new one without n_0 and Meep a finite difference time domain software. The field is measured at 3 depths of (a) 10m (b) 50m (c) 74.9m and plotted along the radial direction (x).

As we've seen before, the improvements are only in certain regions. In Figure 2b the lack of n_0 seems to improve the approximation around $x = 60m$ and likewise in Figure 2c around $x = 80m$ but in most other places the errors seem comparable to the ParaProp that uses n_0. Thus, I cannot recommend removing n_0 because it doesn't seem to offer a significant advantage.

4 Conclusion

I attempted to improve the parabolic equation solver ParaProp by fixing its implementation of the split step method and improving the approximation of the $\mathbf{Q}$, though neither seemed to significantly improve the error in paraProp when compared with finite difference time domain solutions. Moving forward I would suggest continuing to look at improvements in the $\mathbf{Q}$ operator, though it seems an empirical approach would be better. Even more, I would suggest writing a script to optimize the $\mathbf{Q}$ by incorporating more free variables in it. To start with, one could optimize the best value of n_0 in Eq. A13 of [PSA+21]. The other suggestion is reevaluating whether the split-step method is practical. The conditions of the RET are different from what the parabolic equation and split-step algorithm were originally intended for. The error in our method may be dominated by the fact that the split-step is not meant for this system.

References

[PSA+21] S. Prohira, C. Sbrocco, P. Allison, J. Beatty, D. Besson, A. Connolly, P. Dasgupta, C. Deaconu, K. D. de Vries, S. De Kockere, D. Frikken, C. Hast, E. Huesca Santiago, C.-Y. Kuo, U. A. Latif, V. Lukic, T. Meures, K. Mulrey, J. Nam, A. Nozdrina, J. P. Ralston, R. S. Stanley, J. Torres, S. Toscano, D. Van den Broeck, N. van Eijndhoven, and S. Wissel and. Modeling in-ice radio propagation with parabolic equation methods. *Physical Review D*, 103(10), may 2021.

A Derivation of $\mathbf{Q'}$

Fair warning, almost none of these manipulations are rigorous. We will start by writing $\mathbf{Q}$ as

$$\mathbf{Q} = \sqrt{1 + \boldsymbol{\alpha} + \boldsymbol{\beta}} \tag{1}$$

with $\boldsymbol{\alpha} = \partial_{zz}/k_0^2$ and $\boldsymbol{\beta} = n^2 - 1$. We will then split $\mathbf{Q}$

$$\mathbf{Q'} = \sqrt{1 + \boldsymbol{\alpha}} + \mathbf{x} \tag{2}$$

Where $\mathbf{x}$ holds the refractive part. To find $\mathbf{x}$ we demand $\mathbf{Q'}^2 = \mathbf{Q}^2$ and obtain

$$1 + \boldsymbol{\alpha} + \boldsymbol{\beta} = 1 + \boldsymbol{\alpha} + \mathbf{x}^2 + 2\mathbf{x}\sqrt{1 + \boldsymbol{\alpha}}$$

$$\boldsymbol{\beta} = \mathbf{x}^2 + 2\mathbf{x}\sqrt{1 + \boldsymbol{\alpha}}$$

but we will make a substitution of $\mathbf{Y}$ in for $\sqrt{1 + \boldsymbol{\alpha}}$. We define $\mathbf{Y}$ as a function of z, so it is a part of the refractive component of $\mathbf{Q}$, but will match the action $\sqrt{1 + \boldsymbol{\alpha}}$ when acting on u. Solving for $\mathbf{x}$ in terms of $\mathbf{Y}$ and $\boldsymbol{\beta}$

$$\mathbf{x} = \pm\sqrt{\boldsymbol{\beta} + \mathbf{Y}^2} - \mathbf{Y} \tag{3}$$

To find which sign to use, consider the case when $1 + \boldsymbol{\alpha} \approx 1$ then $\mathbf{Y} = 1$ and $\mathbf{Q'}$ reduces to

$$\mathbf{Q'} = \sqrt{1 + \frac{\partial_{zz}}{k_0^2}} \pm n - 1 \tag{4}$$

we compare this to the Feit and Fleck approximation to $\mathbf{Q}$

$$\mathbf{Q_{ff}} = \sqrt{1 + \frac{\partial_{zz}}{k_0^2}} + n - 1 \tag{5}$$

it is common in parabolic equation literature. So, for our $\mathbf{Q'}$ to properly reduce to the Feit and Fleck approximation, we will choose the (+) solution. Thus our general form of our improved $\mathbf{Q}$ approximation is

$$\mathbf{Q'} = \sqrt{1 + \frac{\partial_{zz}}{k_0^2}} + \sqrt{n^2 - 1 + \mathbf{Y}^2} - \mathbf{Y} \tag{6}$$

where $\mathbf{Y}$ is left to be found based on the specific problem.