# Operator learning for constructing transparent BC for hyperbolic equations

Rodionov Danila [1]

[1]Lomonosov Moscow State University

## Contructing transparent boundary conditions problem

The scalar wave equation and Maxwell's equations govern problems in such diverse application areas as ultrasonics, seismics, underwater acoustics, antenna design, and microelectronics. In many cases, the governing equations are posed as exterior problems, and the infinite physical domain must be reduced to a finite computational domain through the use of a nonreflecting boundary condition.

## Example: wave propagation in infinite cylinder

As a certain example we will consider wave propagation in infinite circle channel with radius $r = a$, $\tilde{\Omega} = S^1 \times \mathbf{R}$. This process can be desribed as a wave equation

$$\frac{1}{c^2}\frac{\partial^2 w}{\partial t^2} = \Delta w$$

where $c$ - speed of sound in media, $t$ is time, and $\Delta$ - is Laplace operator.

To fix some parameters, we will set

$$w(r, \varphi, z, t) = 0, \text{ while } t \leq 0$$

and

$$\alpha\frac{\partial w}{\partial r} + \beta w = 0, |\alpha| + |\beta| \neq 0$$

Thus, we can conduct, that our problem is to find boundary conditions operator on artificial hyperplanes $z = z_L$ and $z = z_R$, so solutions of differential problems in inner domain $\Omega_{z_L,z_R}$, that is obtained by artificial boundaries from $\tilde{\Omega}$, and outer domain $\tilde{\Omega}\backslash\Omega_{z_L,z_R}$.

Analytical solution for constructing TBC operator $\mathcal{D}$, such that $\mathcal{D}[u] = 0$ can be represented by next algorithm:

1. In outer domain consider Fourier decomposition of our function by eigen functions $\psi_{k,n,m}(r, \varphi)$ of Laplace operator in circle channel:
$$w(r, \varphi, z, t) = \sum_{n=0}^{\infty}\sum_{k=1}^{\infty}\sum_{m=1}^{2}\psi_{k,n,m}(r, \varphi)w_{k,n,m}(z, t)$$
2. Then, after substituion of this form to inital equation, for each basis function we will have:
$$\frac{1}{c^2}\frac{\partial^2 w_{k,m,n}}{\partial t^2} = \frac{\partial^2 w_{k,m,n}}{\partial z^2} - \lambda_{k,n}w_{k,m,n}$$
After some manipulation with coordinates we will obtain
$$\ddot{u} = u'' - u$$
3. For this equation we can apply Laplace transform (definition below):
$$\mathcal{L}[f(t)](s) = \int_0^{\infty} e^{-st}f(t)dt$$
And get ordinary differential equation of the form
$$U'' = (s^2 + 1)U$$
4. Finish? Not yet. This equation we can rewrite as
$$U' = PU\ (*)$$
$P$ is an Poincare-Steklov operator. But why is it an operator? Because this equation is written for ***every*** basis function. Remark : for some problems $P$ has diagonal form.

Actually, for most problems analytical form of $P$ can be calculated. So fair question is - what's next? Problem is literally solved. Yes, but it is solved in the space of Fouirer-Laplace image, and before trying to solve this task in original space we need to evaluate inverse transforms of equation $(*)$. Because of some techinal reasons, before applying inverce Laplace transform we need to decompose operator $P$ by degrees of variable $s$:

$$P(s) = P_1 s + P_0 + K(s)$$

For our example $P(s) = \sqrt{s^2 + 1}$, and decomposition has form $P(s) = s + K(s)$

and asympthotics of $K(s) = \frac{1}{s} + o(1)$.

In the space of Laplace originals $(s)$ must be represented as a sum of exponents, so we will set $K(s) = \frac{P_{n-1}(s)}{Q_n(s)}$, where $P(s), Q(s)$ are polynomials with index corresponding degrees.

## Methodologies for constructing TBC

- Classical numerical method (Pade approximation), based on Taylor decomposition. Implementation can be found in Scipy, Matlab, Maple. Cons - slow and diverges for some cases.
- NeuralPade or vanilla MLP-based method.
- End2end operator learning learning Fourier-Laplace image of Poincare-Steklov convolution kernel operator.
- Generative-based models for wave trajectories sampling.

## NeuralPade

We will construct topology our neural network $F_\theta(s)$ based on classical MLP structure, but with with resulting form as a rational function. Data for training - mesh on $\mathbf{R}$, target - analytical function values at dataset points. Pros - robust for any function, for simple architechures are as fast as classical Pade method.

## Resume

Thus, first of all we want our neural network $F_\theta(s)$ to approximate function $F(s) = \sqrt{s^2 + 1} - s$ meanwhile $F_\theta(s)$ must have form of rational approximation by input. We will validate regression metrics on approximation of $F(s)$ by $F_\theta(s)$ and approximation error between inverce Laplace transforms $\mathcal{L}^{-1}[F(s)](t) = \frac{\mathcal{J}_1(t)}{t}$ and $\mathcal{L}^{-1}[F_\theta(s)](t)$. We will consider Chebyshev norm and MAE metric.

| Number of poles | $C$ norm for Laplace images | $C$ norm for Laplace originals |
|---|---|---|
| 10 | 0.25*1e-6 | 0.012 |
| 15 | 0.16*1e-6 | 0.05 |
| 25 | 0.09*1e-6 | 0.01 |
| 40 | 0.02*1e-7 | 0.0009 |

Table 1. NeuralPade results

## End2end operator learning

End2end operator learning learning Fourier-Laplace image of Poincare-Steklov convolution kernel operator. We will step aside from classical algorithm and learn our function to minimize parametrical operator loss function of the form:

$$\frac{\partial u}{\partial t} - c\frac{\partial u}{\partial z} + B * u = 0$$

where $B$ is convolution kernel, $*$ - is convolution operator.

So, our architechture will have to heads: first will approximate convolution kernel $B$, second will simulate analytical TBC equation by prediction solutions $u_\theta$. So, our loss function will have form:

$$\mathcal{L}_{op} = \lambda_1\mathcal{L}_{TBC} + \lambda_2\mathcal{L}_{Laplace}$$
$$\mathcal{L}_{TBC} = ||\frac{\partial u_\theta}{\partial t} - c\frac{\partial_\theta}{\partial z} + B_\varphi * u_\theta||$$
$$\mathcal{L}_{Laplace} = ||B_\varphi(t) - \tilde{F}(t)||$$

Where, $B_\varphi(s) = \sum_{p=1}^{N} a_p e^{b_p t}$

We will stack our dataset for uniform meshes with affine points $x$, and for approximation target use basis functions from $\mathcal{L}_2$

## Diffusion-based operator generation

Generative-based models for wave trajectories sampling. Consider a neighbourhood of our artificial boudaries. We can consider wave propagation in image domain (motion-based image/spectrogramm). Then via generative model (preferably diffusion) we can learn latent distribution in this area, then convole it to function value (with encoder) in certain point and evaluate function value/values. As a loss function we will use default loss construction for diffusion model and weightned operator learning loss part.

So, our network we be represented by ordinary diffusion decoder sampler, learned by minimization of the loss function below:

$$\mathcal{L} = \mathcal{L}_{DDPM} + \mathcal{L}_{op} = \mathcal{L}_{DDPM} + \lambda_1\mathcal{L}_{TBC} + \lambda_2\mathcal{L}_{Laplace}$$

## Conclusion

In our poster we have presented an idea of transparent boundary conditions (TBC), guided by analytical task from mathematical physics. List of approaches was presented, baseline results were declared. As a next steps preview main tasks were considered.

## References

- N. A. Zaitsev, I. L. Sofronov, "Generation of transparent boundary conditions for modeling wave propagation in anisotropic media", Preprints of Keldysh Institute of Applied Mathematics, 2018, 082, 36 pp.
- Zongyi Li, Hongkai Zheng, Nikola Kovachki, David Jin, Haoxuan Chen, Burigede Liu, Kamyar Azizzadenesheli, Anima Anandkumar "Physics-Informed Neural Operator for Learning Partial Differential Equations"
- Jan-Hendrik Bastek, WaiChing Sun, Dennis M. Kochmann "Physics-Informed Diffusion Models" arXiv:2403.14404v1 [cs.LG]