Solutions to the Wave Equation

Max Orton

Sophomore and Junior Year: 2022 through 2023

The wave equation is one of the earliest projects I tried to figure out. I started investigating it during my Freshman year as a stepping stone toward Maxwell's Equations. This write-up describes a model I created that involves breaking an initial condition into components, each defined only along a single axis. It is the simplest general solution I've created so far.

We will be investigating and finding solutions to the wave equation:

$$\Box\, \phi = 0 \quad \text{or} \quad \frac{\partial^2 \phi}{\partial t^2} = c^2 \nabla^2 \phi$$

This problem was a priority of mine, primarily because it was fun, but also because the wave equation turns out to be equivalent to a surprising number of other interesting PDEs. A good example of this is the Klein Gordon equation. Say we have a field ϕ that satisfies the wave equation in N dimensional space (*space* not spacetime).

$$\frac{\partial^2 \phi}{\partial t^2} = \frac{\partial^2 \phi}{\partial x_1^2} + \frac{\partial^2 \phi}{\partial x_2^2} + \cdots + \frac{\partial^2 \phi}{\partial x_N^2}$$

We can invent a new field ψ that depends only on N-1 of the coordinates. It is defined:

$$\phi(x_1, x_2, \cdots x_N) = \psi(x_2, \cdots x_N) e^{imx_1}$$

If we plug this into our equation for ϕ, we can get an expression for ψ.

$$\frac{\partial^2 \psi}{\partial t^2} e^{imx_1} = \left(\frac{\partial^2}{\partial x_1^2} \psi e^{imx_1} \right) + \frac{\partial^2 \psi}{\partial x_2^2} e^{imx_1} + \cdots + \frac{\partial^2 \psi}{\partial x_N^2} e^{imx_1}$$

Expanding this, we find that ψ follows the Klein Gordon Equation.

$$\frac{\partial^2\psi}{\partial t^2} = \frac{\partial^2\psi}{\partial x_2^2} + \cdots + \frac{\partial^2\psi}{\partial x_N^2} - m^2\psi$$

Before we solve the entire wave equation, we can start with the one dimensional version. We see that it is:

$$\left(\frac{\partial^2}{\partial t^2} - c^2\frac{\partial^2}{\partial x^2}\right)\phi = 0$$

We can factor this expression into two terms. It is worth noting that this is not the way I originally solved this equation. When I first solved it, I Fourier Transformed it in space, but not in time, which made it a harmonic oscillator. I solved that equation, then reversed the Fourier Transform. As it is, we will use a simpler method I discovered later.

$$\left(\frac{\partial^2}{\partial t^2} - c^2\frac{\partial^2}{\partial x^2}\right)\phi = \left(\frac{\partial}{\partial t} - c\frac{\partial}{\partial x}\right)\left(\frac{\partial}{\partial t} + c\frac{\partial}{\partial x}\right)\phi = 0$$

Only one of the two factors, when operated on ϕ, needs to yield zero for this equation to be satisfied. This gives us two equations, and their solutions can be written in terms of two arbitrary functions: $f : \mathbb{R} \to \mathbb{R}$ and $g : \mathbb{R} \to \mathbb{R}$.

$$\left(\frac{\partial}{\partial t} - c\frac{\partial}{\partial x}\right)\phi_L \to \phi_L(x,t) = f(x+ct)$$

$$\left(\frac{\partial}{\partial t} + c\frac{\partial}{\partial x}\right)\phi_R \to \phi_R(x,t) = g(x-ct)$$

Because of linearity, ϕ can be written as a sum of these two solutions.

$$\phi(x,t) = f(x+ct) + g(x-ct)$$

Now we can move on to the N dimensional wave equation. The obvious solution is to write it as a Fourier Transform. For a real field, this version of the solution would have the form:

$$\phi(\vec{x},t) = \int \alpha(\vec{k})e^{i(\vec{k}\cdot\vec{x}+t\|\vec{k}\|)} + \overline{\alpha}(\vec{k})e^{-i(\vec{k}\cdot\vec{x}+t\|\vec{k}\|)}d^N k$$

This expression can be very useful. I've used it extensively when working with quantum fields, but it is not the most satisfying answer. After all, it requires N different integrals to evaluate the expression, not to mention at least N more to set the initial conditions.

We are going to notice something else. If a function depends only on one of the x's, it must satisfy the one dimensional wave equation as all the other derivatives would be zero. Likewise, since the wave equation is rotationaly symmetric, a function that depends only on any unit vector dotted with position must follow the one dimensional wave equation with respect to a directional derivative along that unit vector. Any linear combination of these vectors would also need to satisfy the wave equation. This gives us a form for ϕ: basically just a superposition of plane waves f_n moving along a collection of corresponding unit vectors α_n.

$$\phi(\vec{x},t) = \sum_n f_n(\vec{x}\cdot\hat{\alpha}_n - ct)$$

The question now becomes, how do we find the set of f_n given an initial condition? Basically we want an object $\mathfrak{F}$ defined.

$$\text{if} \quad g(\hat{\alpha},r) = \mathfrak{F}[f(\vec{x})](\hat{\alpha},r) \quad \text{then} \quad f(\vec{x}) = \int g(\hat{\alpha},\vec{x}\cdot\hat{\alpha})d^{N-1}\alpha$$

I had a small hint with this one. One of the SoME3 submissions was about "string art" (creating pictures by connecting strings at the edges of a circle), and involved solving a problem similar to this one. The person in the video solved it by just brute-forcing the problem in a way that could not be done symbolically. However, at the very end, he mentioned that he now had a new algorithm involving the Fourier Transform. He did not explain how it worked (honestly, I was glad, as I wanted to figure it out myself), but it gave me a place to start as I knew the Fourier Transform would be involved.

Since we know the solution will somehow involve the Fourier Transform, we can begin by writing f in terms of its Fourier Transform, then manipulate it to try and get it into a form similar to the equation above.

$$f(\vec{x}) = (2\pi)^{-\frac{N}{2}}\int \hat{f}(\vec{k})e^{-i\vec{k}\cdot\vec{x}}d^3k$$

$$f(\vec{x}) = \int (2\pi)^{-\frac{N-1}{2}}\frac{1}{\sqrt{2\pi}}\int \left[\hat{f}(\hat{k},\|k\|)\exp\left(-i\|k\|(\hat{k}\cdot\vec{x})\right)\right]\|k\|^{N-1}d\|k\|d^{N-1}\hat{k}$$

This has the form of an integral over a unit vector $\hat{k}$ of a function that depends on $\vec{x}$ only through $\vec{x} \cdot \hat{k}$. We can use this to write out an expression for $\mathfrak{F}$!

$$\mathfrak{F}[f(\vec{x})] = \mathcal{F}^{-1}_{\|\vec{k}\|}[\mathcal{F}_x[f(\vec{x})](\vec{k})](\hat{k}, r)$$

This means all we have to do to solve the wave equation is evaluate $\mathfrak{F}$ on our initial condition to get the f_n, then add them all back together in the right way to get ϕ. A few moments of thought will show that we actually have the same total number of integrals to perform in this process as in the original Fourier Transform method. That said, this solution still does have an advantage. We've shifted one of our 2N integrals from the evaluation to setting the initial conditions. We will often want to evaluate a solution at multiple points in time to get a feel for how it evolves. This solution saves us one integral for each evaluation, save the first one!

My quest to solve this problem has taught me more than I could ever have imagined at the start. When I began, I barely understood the difference between an ODE and a PDE. Now I have homemade methods for solving both. It sometimes felt like I was approaching this with a strategy that valued quantity over quality. I invented dozens of methods, hoping that one might just apply to the wave equation. This process has left me with tools to solve all kinds of other PDEs. I don't consider this project complete. Though this decomposition method is the simplest I've found so far, I'm hopeful that a simpler solution exists. A lot of my more recent work on the wave equation and other PDEs revolves around creating new number systems in which they can be more easily solved: trying to generalize the fact that an arbitrary function of a complex number reproduces Laplace's Equation in 2D to more equations and corresponding number systems. I've used this to find exact solutions for all 2D PDEs where the total number of derivatives on each term is constant. I'm also exploring ways to extend this method to more complex PDEs and in more dimensions.