A case of numerical relativity

Conghuan Luo
Computational Physics

December 20, 2018

1 Introduction

General relativity, accompanied by quantum mechanics, are the foundations of modern physics. The equations of general relativity are given by Einstein's field equations. As a series of nonlinear equations, they can only be solved in highly-symmetric cases or in weak-field, post Newtonian limits. As a theory ruling highly macroscopic and near-speed-of-light phenomena, it is extremely difficult to do experiments in laboratories. This is why we need numerical relativity.

Using numerical techniques to solve Einstein's field equations can help us study many cases that cannot be easily studied analytically, like the gravitational collapse of black holes and the inspiral and coalescence of binary black hole systems. It can be treated as a laboratories for classical regime of different gravitational theories. In recent decades, due to the developments of computers and algorithms, numerical relativity becomes a popular areas of research.

As a numerical problem, it is similar to numerical hydrodynamics, but it also shows its own particular difficulties. People would like to use numerical relativity to study black holes which is extremely hard to study numerically. But for black holes, there are singularities that we need to take care of. One technique is called black hole excision.

In my project, my present goal is to write a code that can simulate the evolution of spacetime coupled with some matter in Anti de-Sitter (AdS) background. Asymptotically AdS spacetime has its own features that makes numerical computation much more difficult than that in asymptotically flat

spacetime. AdS spacetime is interesting in its own right, due to the famous Anti de-Sitter/Conformal Field Theory correspondence firstly proposed by Juan Maldacena [5]. This correspondence gives us profound insight about how quantum gravity should look like. Yet I am not sure if numerical relativity can provide us some hints about AdS/CFT correspondence. As for numerical part, I only succeed in simulating weak-field limit configurations. For strong-field cases, the codes I use will generate instabilities during evolution, so that I cannot study the gravitational collapses. This should also be the future directions of my project.

In section 2, I will introduce the basic knowledge we need to translate the physics problem into numerical languages. I will show how to generally solve the Einstein's field equations numerically and what the difference is for AdS spacetime. In section 3, I will introduce the numerical schemes I use to overcome the instabilities that generate for a coupled series of nonlinear partial differential equations. There are some techniques specifically for AdS spacetime. In section 4, I will provide some computational results given by my own codes. In section 5, I will conclude my work and point out some possible future directions.

2 Physics background for numerical interpretation

Einstein's field equations are written as tensorial form:

$$R_{\mu\nu} - \frac{1}{2}Rg_{\mu\nu} = T_{\mu\nu} \tag{1}$$

the metric tensor $g_{\mu\nu}$ is the dynamical variable of this equation. But from ADM formalism [1], we can construct a Hamilton's formalism for general relativity that sorts out the real dynamical components of metric tensor and gives us the real dynamical equations of them. Let me sketch the basic ideas here.

General relativity is invariant under coordinate transformation, so it is actually in parametrized form with four reparametrization symmetry. Moreover, the 0μ components of Einstein's field equation are constraint equations, which contains at most 1st order time derivative. They are the relations on each time slice. So eliminating these 8 degrees of freedom, there are actually only two dynamical degrees of freedom, corresponding to transverse traceless

part of g_{ij}. In other words, we only need to use two equations to solve the time evolution.

One thing we need to do is to choose a convenient coordinate system for our study. In this project, I study the propagation of massless scalar wave in asymptotically AdS_3 spacetime. AdS spacetime is the solution of Einstein's field equation with an extra cosmological constant term. In general relativity, we usually use Penrose diagrams to visualize the spacetime configuration, because Penrose diagrams preserve the causal structure of spacetime while compactifying the space and time coordinate into a finite region. So we can see that light cones are along 45°straight lines, which are also the geodesics that massless fields propagates along. From the Penrose diagram of AdS_3 spacetime we find that there is no way to compactify the time direction while preserving causal structure. That's because in asymptotically AdS_3 spacetime a static observer can send a light signal to null infinity and reflect back in a finite proper time. In asymptotically flat spacetime, to simulate the time evolution, we do not need to worry about spacelike infinity and null infinity because nothing will reach there in finite proper time. What we need to do is to impose out-going boundary condition for fluxes. But in AdS_3 spacetime, we need to impose physical boundary conditions that conserve the fluxes during "reflection", and hence conserve the total matter.

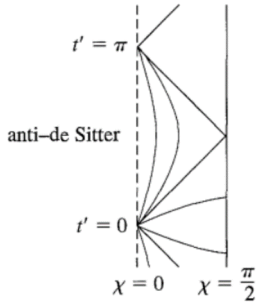


Figure 1: The Penrose diagram of AdS_3 spacetime

In this project I use the same coordinate system in [6], which actually generates Penrose diagram. The metric of asymptotically AdS_3 is:

$$ds^2 = \frac{e^{2A(r,t)}}{\cos^2(r/\ell)}\left(dr^2 - dt^2\right) + \ell^2 \tan^2(r/\ell) e^{2B(r,t)} d\theta^2 \tag{2}$$

The advantage of this coordinate system is that we only need to consider

finite region of space and the light-cones are along 45°straight lines. After we rescale the AdS radius ℓ, we have $r \in [0, 1]$. When $A = 1$ and $B = 1$, we recover AdS vacuum. Here the real dynamical parts of metric are represented by A and B, which is determined by the matter field.

We consider massless scalar field satisfying Klein-Gordon equation:

$$\nabla_\mu \nabla^\mu \phi = 0 \tag{3}$$

Here, to lower the time derivative to 1st order, we denote:

$$\Phi(r,t) \equiv \partial_r \phi(r,t), \quad \Pi(r,t) \equiv \partial_t \phi(r,t) \tag{4}$$

$$\partial_t A(r,t) \equiv A_t(r,t), \quad \partial_t B(r,t) \equiv B_t(r,t) \tag{5}$$

Write out all the components of Einstein's field equations as well as Klein-Gordon equation:

$$\partial_t A_t = \partial_r^2 A + \big(\frac{\pi}{2}\big)^2 \frac{1 - e^{2A}}{cos^2(\pi r/2)} + 2\pi(\Phi^2 - \Pi^2) \tag{6}$$

$$\partial_t B_t = \partial_r^2 B + \partial_r B(\partial_r B + \frac{2\pi}{sin(\pi r)}) - B_t^2 + \frac{\pi^2}{2}\frac{1 - e^{2A}}{cos^2(\pi r/2)} \tag{7}$$

$$\partial_r^2 B + \partial_r B(\partial_r B - \partial_r A + \frac{\pi(1 + cos^2(\pi r/2))}{sin(\pi r)}) - \frac{\pi \partial_r A}{sin(\pi r)} - A_t B_t + \frac{\pi^2(1 - e^{2A})}{4cos^2(\pi r/2)} + 2\pi(\Phi^2 + \Pi^2) = 0 \tag{8}$$

$$\partial_r B_t + B_t(\partial_r B - \partial_r A + \frac{\pi}{2tan(\pi r/2)}) - A_t(\partial_r B + \frac{\pi}{sin(\pi r/2)}) + 4\pi\Phi\Pi = 0 \tag{9}$$

$$\partial_t \Phi = \partial_r \Pi \tag{10}$$

$$\partial_t \Pi = \Phi \partial_r B + \partial_r \Phi + \frac{\pi \Phi}{sin(\pi r)} - B_t \Pi \tag{11}$$

Equations 8 and 9 are constraint equations. I use them to determine initial conditions and monitor the error of time evolution. Other four equations are used to compute time evolution.

We also need to set either Dirichlet of Neumann boundary conditions at $r = 0$ and $r = 1$. At $r = 0$, we want it to be regular, so the partial derivative with respect to r must vanish for all physical quantities. And at $r = 1$ we need the spacetime to be asymptotically AdS. So we choose the boundary conditions as follows:

$$\partial_r A(0,t) = 0, \quad \partial_r B(0,t) = 0, \quad \Phi(0,t) = 0, \quad \partial_r \Pi(0,t) = 0 \tag{12}$$

$$A(1,t) = 0, \quad \partial_r B(1,t) = 0, \quad \Phi(1,t) = 0, \quad \Pi(1,t) = 0 \tag{13}$$

Actually there are more physical boundary conditions than that we need. The extra ones can be used to check the consistency of evolution.

Note that here I don't apply the conservation of fluxes at the boundary, but it can be proved that Dirichlet boundary conditions for Φ and Π are equivalent to conservation of fluxes. One thing worth mentioning is that we find that dynamical equations diverge at the boundary as $\frac{1}{sin(\pi r)}$ or something, which somehow imply the discontinuity given by the reflection of fluxes there. This is also a feature of asymptotically AdS spacetime. And the main trouble I will encounter comes from the unstable modes given by the divergent parts. That's what I am trying to solve in the next section.

3 Numerical schemes

Constraint equations can be considered as 1st order ordinary differential equations on the spatial dimension, which can be easily solved using 4th order Runge-Kutta method (RK4) if we impose proper boundary conditions. So I am not gonna discuss them here. The main problem remains is how to solve the dynamical equations, which are hyperbolic or parabolic partial differential equations.

3.1 Spatial discretization

Generally, partial differential equations with 1st order time derivative can be written as:

$$\partial_t u = F(u, x, t, \partial_x u, \partial_x^2 u, \dots) \tag{14}$$

The most popular way is to discretize the spatial dimension. We can represent $u(x,t)$ as $u_i(t)$, where i is the index of grid points. If we Taylor expand $u(x,t)$

at the grid points, we can represent spatial derivatives by several points. The representation often used is 2nd order:

$$(\partial_x f)_i = \frac{f_{i+1} - f_{i-1}}{2\Delta x} + \mathcal{O}\left(\Delta x^2\right) \tag{15}$$

$$\left(\partial_x^2 f\right)_i = \frac{f_{i+1} - 2f_i + f_{i-1}}{(\Delta x)^2} + \mathcal{O}\left(\Delta x^2\right) \tag{16}$$

However, for AdS case, due to the divergence at the boundary previously mentioned, we need to go to higher order scheme so as to smear the divergence. And this is really important for the stability of my algorithm. Here I use 4th order representation:

$$\begin{array}{c}(\partial_x f)_i = \frac{1}{12\Delta x}\left(f_{i-2} - 8f_{i-1} + 8f_{i+1} - f_{i+2}\right) \\ (\partial_x^2 f)_i = \frac{1}{12(\Delta x)^2}\left(-f_{i-2} + 16f_{i-1} - 30f_i + 16f_{i+1} - f_{i+2}\right)\end{array} \tag{17}$$

After discretizing the spatial dimension, the time evolution for each i can be regarded as 1st order ordinary differential equations, which can also be solved using RK4. But in the next sub-section, I will introduce a new scheme for non-linear differential equations.

I impose my boundary conditions by adding ghost cells at the boundary. For example, for Neumann boundary condition $\partial_r A = 0$, I will add $A[-1] = A[1]$, $A[-2] = A[2]$, etc., so that I can represent derivatives at the boundary using central representation.

3.2 Iterative Crank-Nicolson scheme

The simplest way to perform time evolution is forward Euler method:

$$u_i^{n+1} = u_i^n + \Delta t * F \tag{18}$$

But it is 1st order precise and has instability problem. If we consider advection problem, we find that forward Euler method is unconditionally unstable. A more precise and stable method is called Crank-Nicolson scheme. It is the combination of forward Euler method and backward Euler method and represent the time derivative at the midpoint:

$$u_j^{n+1} = u_j^n + \frac{1}{2}\Delta t(F_i^n(u^{n+1}, x, t) + F_i^n(u^n, x, t)) \tag{19}$$

It is unconditionally stable for advection problem and also stabler for many other partial differential equations.

But the shortcoming of it is that it is implicit, which means we cannot compute the $n+1$ terms from the n terms. Instead, we need to solve a coupled equations, which is extremely difficult for nonlinear partial differential equations. The alternative usually used in numerical relativity is iterative Crank-Nicolson scheme.

The method can be described as follows:

$$\begin{aligned} {}^{(1)}u_j^{n+1} &= u_j^n + \Delta t F_i^n(u^n, x, t) \\ {}^{(2)}u_j^{n+1} &= u_j^n + \frac{1}{2}\Delta t(F_i^n({}^{(1)}u^{n+1}, x, t) + F_i^n(u^n, x, t)) \\ {}^{(3)}u_j^{n+1} &= u_j^n + \frac{1}{2}\Delta t(F_i^n({}^{(2)}u^{n+1}, x, t) + F_i^n(u^n, x, t)) \\ &\cdots \end{aligned} \tag{20}$$

We firstly use forward Euler method to approximate the value at the next step and plug the result back into Crank-Nicolson scheme. By iterating in this way, it can be proved that it will converge to implicit Crank-Nicolson scheme. However, now it becomes explicit and can be applied to nonlinear equations easily. It is studied that for the scheme to be stable and precise, we should iterative for exactly two times [7].

Except for iterative Crank-Nicolson scheme, I also use non-adaptive RK4 to do the time evolution. It happens that these two schemes give almost the same results and do not help for the stability problem in strong-field limit. That's why I infer that the main instability comes from the divergent parts (this actually is shown in the plots), which has nothing to do with the instabilities during time evolution.

3.3 Kreiss-Oliger dissipation

A clever way to stabilize the solutions of partial differential equations with time evolution is given by Kreiss and Oliger [4]. When studying advection problem, another way to stabilize the FTCS method is called Lax method. It is equivalent to adding a nonphysical diffusive term on the right-hand side. Similar ideas can be used in numerical relativity.

Kreiss-Oliger dissipation is a numerical diffusive term with high order derivatives and small amplitude. Such a diffusive term can damp the high

frequency mode and stabilize the evolution. The normally used form can be found in Appendix.C of [2]. Here I use a 3rd order version of Kreiss-Oliger dissipation:

$$\partial_t u_m \to \partial_t u_m + \frac{\sigma}{64\Delta x}(u_{m+3} - 6u_{m+2} + 15u_{m+1} - 20u_m + 15u_{m-1} - 6u_{m-2} + u_{m-3}) \quad (21)$$

Higher order dissipation can also be applied. It turns out that this technique stabilizes the algorithm considerably.

4 Results

I use some specific wave packets to visualize the evolution of spacetime and matter field. The first one is Gaussian wave:

$$\phi(r,0) = Pe^{((r-r_0)/\sigma)^{2n}} \quad (22)$$

Here I set $n = 1$, $r_0 = 0.2$ and $\sigma = 0.05$. P can be varied and the algorithm is only stable for small P.

The second one is harmonic wave:

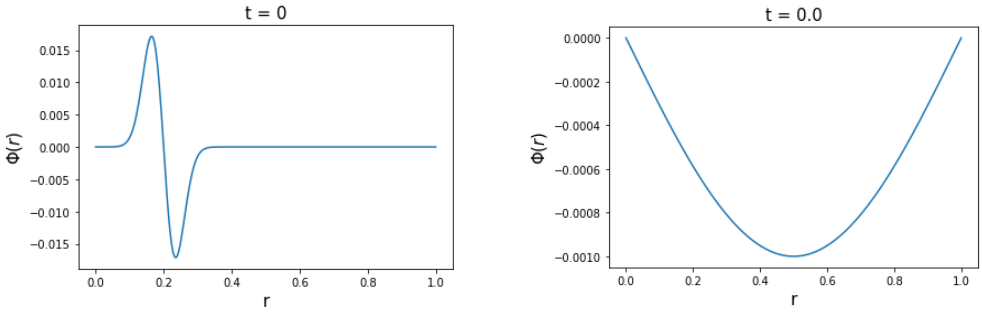
$$\phi(r,0) = P\cos^2(\pi r n/2) \quad (23)$$

Here I set $n = 1$.

We can study three cases: in-going wave $\Phi(r,0) = \Pi(r,0)$, out-going wave $\Phi(r,0) = -\Pi(r,0)$ and static wave $\Pi(r,0) = 0$.

For the convenience, we set $B(r,0) = 0$, $B_t(r,0) = 0$. They only determine how we slice and seam our time slices and don't affect the dynamical information. With Φ and Π, we can obtain $A(r,0)$ and $A_t(r,0)$ from equations 8 and 9. For those two wave packets mentioned, $\Phi(r,0)$ look like figures 2 for $P = 0.005$:

The calculated dynamical metric component $A(r,0)$ is depicted as in figure 3. Note that in weak field limit, the matter field acts as a small perturbation on the metric field, so for all different wave packets the metric looks pretty much the same. Yet they are actually different because Einstein's field equations are coupled with matter field. Note that to compute $A(r,0)$ and



(a) gradient of Gaussian wave (b) gradient of harmonic wave

Figure 2: initial wave packets

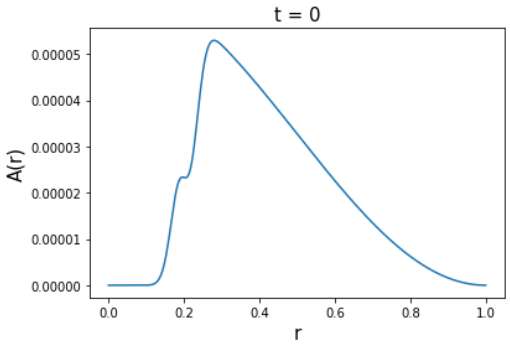


Figure 3: A(r, 0)

$A_t(r, 0)$ we assume that $A(0, 0) = 0$, which means that there is no finite mass point particle at the origin.

With all the initial conditions set, I compute the time evolution of both wave packets. The Gaussian wave in AdS_3 spacetime propagates as in figure 4. We find that it propagates along null geodesics as expected. If we look carefully, we find that there is a weak out-going propagating wave, too. It may come from the back-reaction of the metric field. It is not interesting to plot out-going wave and static wave of Gaussian form because we can still only see the light-cones.

The harmonic wave in AdS_3 spacetime propagates as in figure 5. The static wave is more interesting with the interference pattern.

To visualize the back-reaction on metric, we can also plot the Ricci scalar, computed as follows. From the expression, we find that in weak-field limit,

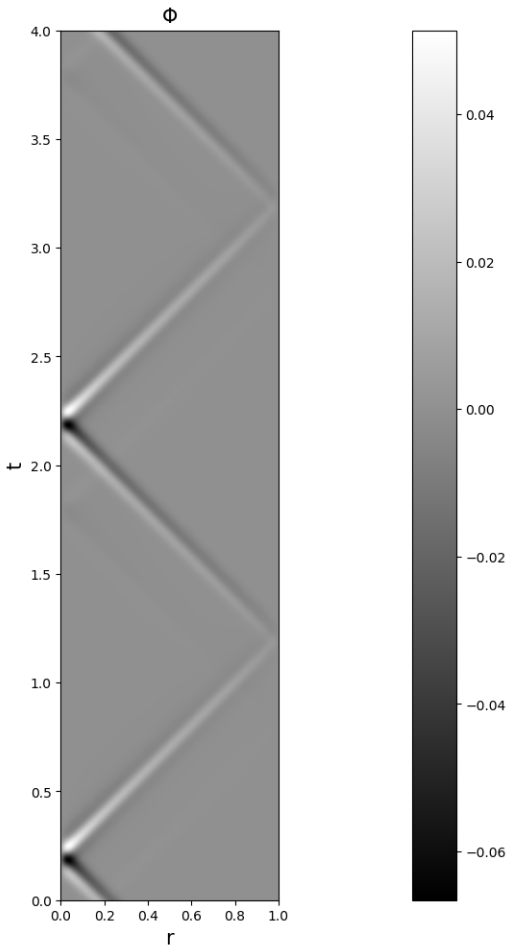


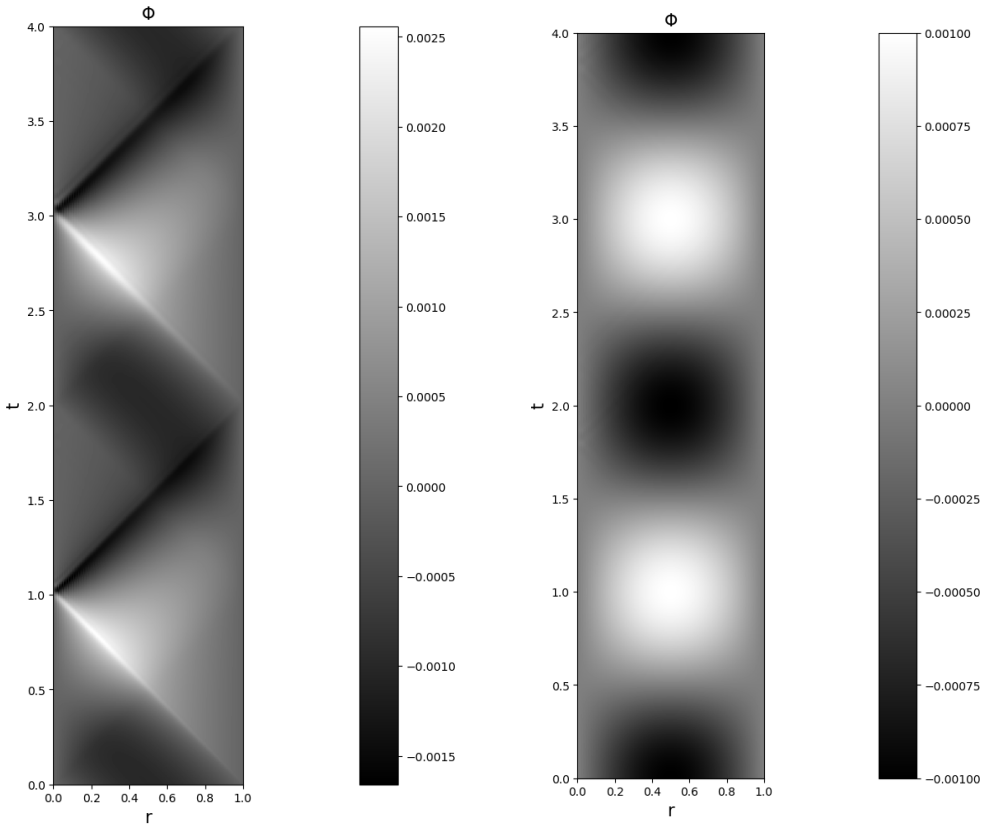
Figure 4: $\Phi(r,t)$ for in-going Gaussian wave

i.e. $P \ll \frac{1}{\ell} = \frac{\pi}{2}$, matter field do act as a small perturbation on Ricci scalar. The leading part is contributed by negative cosmological constant.

$$R = \frac{\pi^3 \cos(\pi r/2)^2}{e^{2A}} \left(\Phi^2 - \Pi^2\right) - \frac{3\pi^2}{2} \tag{24}$$

The diagram is as in figure 6.

Because $r \in [0,1]$ and massless field propagates along 45°straight lines, wave propagation in AdS_3 spacetime has a period $T = 2$. The cross-section of wave at $t = 4.0$ is depicted in figure 7. We find that there are already unphysical modes in the figure, as a proof that my algorithm is not stable



(a) $\Phi(r,t)$ for in-going harmonic wave (b) $\Phi(r,t)$ for static harmonic wave

Figure 5: Propagation of harmonic wave

enough, even for weak-field limit. But the general picture is correct.

4.1 A study of stability

Given that there are still manifest instabilities during the evolution, I hope to analyze how the parameters I use affect the stability of the algorithms, so that hopefully I can find out an optimal set of parameters. Here there are four parameters that can be monitored: CFL coefficient (i.e. the ratio of time-step and spatial grid spacing), Kreiss-Oliger dissipation amplitude, number of spatial grid points N_x and scalar wave amplitude.

By varying these parameters, I computed corresponding errors of the evolution over time. Here I use equation 9 to represent the error. But note

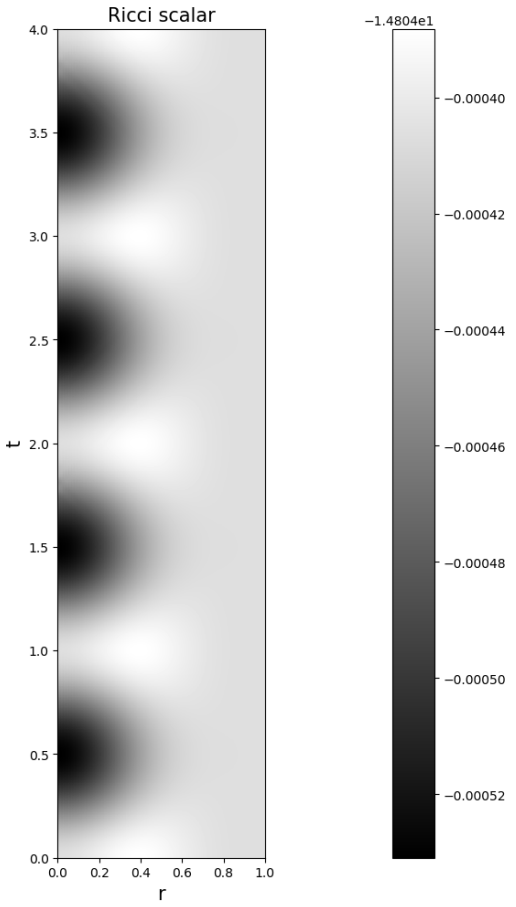
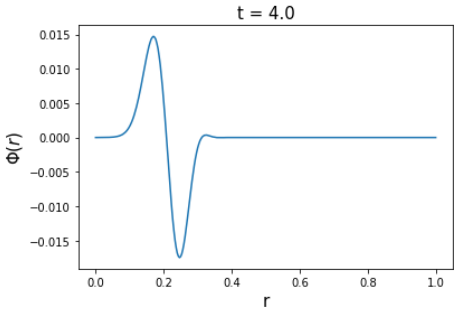


Figure 6: Ricci scalar for static harmonic wave

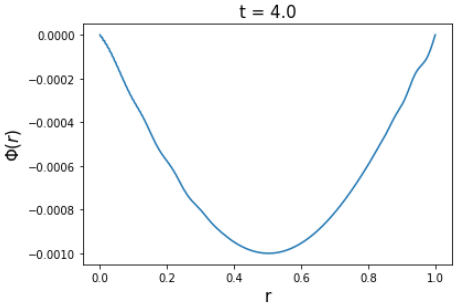
that we already see the instabilities in the diagrams, so the error is actually not small.

I find that there are safe windows for those parameters, while outside the windows the algorithm will just fail. Inside the windows the error is approximately at the same order so optimizing the algorithm by monitoring the parameters seems in vain. Here I still depict the relation between error and parameters in figure 8. For the pictures depicted above, the parameter I use is CFL coefficient 0.8, dissipation amplitude 0.08, $N_x = 300$ and $P = 0.005$.

I compute the error for $t = 4.0$. For CFl coefficient, I find that the algorithm works only for CFL coefficient around 0.8, not when larger or

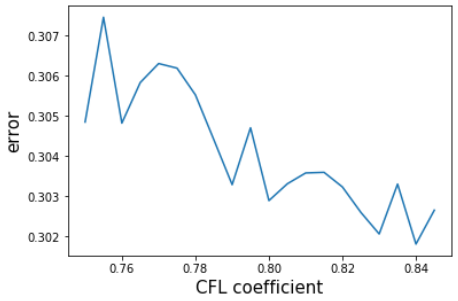


(a) $\Phi(r, 4)$ for in-going Gaussian wave

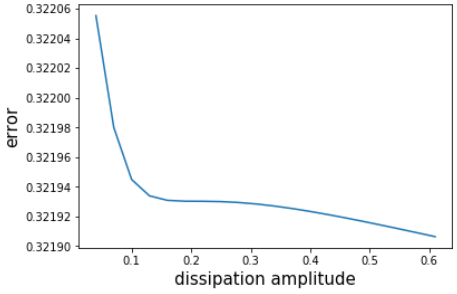


(b) $\Phi(r, 4)$ for in-going harmonic wave

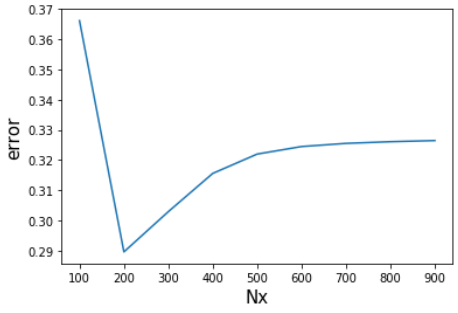
Figure 7: $t = 4.0$ cross-section



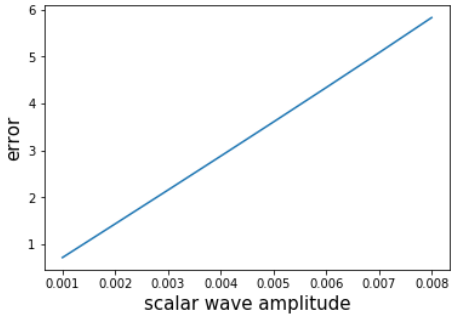
(a)



(b)



(c)



(d)

Figure 8: The relation between error and monitored parameters

smaller. That's pretty counter-intuitive, because normally we expect it to be stabler for smaller CFL coefficient. I guess if we choose larger CFl coefficient, we then need more time-steps, which may increase the instabilities.

For dissipation amplitude, larger ones will smear the physical modes and smaller ones are not enough to control unphysical modes. While inside the safe windows, the error is almost the same. So it doesn't help much by using larger dissipation amplitude.

The relation between error and number of spatial grid points is also interesting. Commonly when we use larger N_x, we should get preciser result. But here the algorithm will fail for large N_x, because the instability mainly grow at the boundary due to divergent terms, which will become severer for small lattice spacing.

For scalar wave amplitude, there is a safe threshold, larger than which the algorithm will fail. That's how I determine the weak-field limit in this case.

In conclusion, the obtain a stabler algorithm, I have to modify it, either by using higher order schemes, or using new techniques.

5 Future directions

In this project, I successfully simulate the evolution of spacetime and matter contents in AdS_3 spacetime for weak-field limit. This can be served as a test of my code, but that's not enough for research project. Although numerical relativity may not be the area of research I plan to do during my PhD. I am happy to point out some possible directions to continue in the future.

One exciting direction of numerical relativity is to study gravitational collapse of black holes. Choptuik [3] found out that there is critical behavior at the threshold of formation of black holes with scalar field collapse. Namely, if we use one parameter to describe initial data, like scalar wave amplitude P, black hole masses have critical behavior:

$$M_{BH} = C|p - p^*|^\gamma \tag{25}$$

Moreover, the solutions of Einstein's field equation around the critical point tend to the same solution and exhibit self-similarity. This is true in both flat spacetime and AdS spacetime.

But to study gravitational collapse in AdS_3, I need an algorithm that is stable for strong field. One attempt is to use one-sided derivative representation at the boundary, and impose Neumann boundary condition using the equation given by one-sided representation, but it only makes things worse.

There are several ideas I can try in the future. Because the instability mainly come from the discretization of spatial dimension, I can try using even higher order representation of derivatives. I can also try higher order Kreiss-Oliger dissipation term. I can also modify the time evolution scheme, although I think it is not the main source of instabilities.

Other thing interesting to do is that I can apply my algorithm to asymptotically flat spacetime, which is easier to do. Maybe I can study gravitational collapse there.

One algorithm that is popular in numerical relativity is called adaptive mesh refinement (AMR), but I don't know much about it.

References

[1] Richard Arnowitt, Stanley Deser, and Charles W Misner. "Dynamical structure and definition of energy in general relativity". In: *Physical Review* 116.5 (1959), p. 1322.

[2] Maria C Babiuc et al. "Implementation of standard testbeds for numerical relativity". In: *Classical and Quantum Gravity* 25.12 (2008), p. 125012.

[3] Matthew W Choptuik. "Universality and scaling in gravitational collapse of a massless scalar field". In: *Physical Review Letters* 70.1 (1993), p. 9.

[4] Heinz Kreiss, Heinz-Otto Kreiss, and Joseph Oliger. *Methods for the approximate solution of time dependent problems*. 10. International Council of Scientific Unions, World Meteorological Organization, 1973.

[5] Juan Maldacena. "The large-N limit of superconformal field theories and supergravity". In: *International journal of theoretical physics* 38.4 (1999), pp. 1113–1133.

[6] Frans Pretorius and Matthew W Choptuik. "Gravitational collapse in 2+ 1 dimensional AdS spacetime". In: *Physical Review D* 62.12 (2000), p. 124012.

[7] SA Teukolsky. "On the stability of the iterated Crank-Nicholson method in numerical relativity, 1999". In: *arXiv preprint gr-qc/9909026* ().