PiTP 2009: Computational Astrophysics

Computational Methods for Numerical Relativity

Lecture 1: Basics

Frans Pretorius
Princeton University

Outline

- General Relativity
  - motivation : why solve the classical field equations?
  - mathematical structure of the field equations
    - general covariance and the character of the field equations
  - properties of typical solutions
    - smooth solutions, though large range of dynamical length scales
    - singularities inside of black holes
  - computational techniques
- overview of elementary finite difference methods and principles
- project: wave propagation on a black hole background
  - basics of black hole physics
  - solving hyperbolic PDEs with RNPL

Motivation

- why solve the classical field equations of general relativity, in particular using numerical methods?
  - **understanding gravity**
    - gravity is one of the fundamental forces of nature, and barring questions about our understanding of dark matter and dark energy, general relativity is the only theory of gravity consistent with all existing tests and observations of the universe where gravity plays an significant role
    - thus, it is important (and interesting!) to understand the full consequences of the theory, which means *solving* the equations to describe situations of interest
    - however, the field equations are quite complicated and non-linear, and analytic solutions only exist in a few special cases
    - global methods have been extremely successful in uncovering general properties of spacetime, though do not give any details on specific scenarios
    - perturbative techniques work in weak-field, slow-motion scenarios; in the dynamical, strong field regime, numerical methods are required

Motivation

- why solve the classical field equations of general relativity, in particular using numerical methods?
  - **gravitational wave astronomy**
    - a new generation of gravitational wave detectors, including LIGO, GEO, VIRGO and the planned space mission LISA, and inferences from pulsar timing and CMB polarization measurements, hold promise to open up the universe to observation in the gravitational wave spectrum
    - however, the weak nature of gravitational waves makes "direct" observation essentially impossible with this generation of ground-based detectors; the only way one could see gravitational wave sources is through some form of matched filtering, where template waveforms are convolved with the signal
    - thus, understanding the detailed nature of gravitational waves from expected sources is crucial to realize the full potential of gravitational wave astronomy
    - the final stages of compact object mergers (black holes, neutron stars) occur in the regime where full numerical solution is required

General Relativity

- In general relativity there is no gravitational "force"; rather, the postulate is we live in a 4-dimensional curved spacetime, and it's curvature of the spacetime that we feel as the gravitational force.
- it is convenient to describe the geometry by a metric tensor g_{ab} , defined via the line element (a generalization of Pythagoras' thm.):

$$ds^2 = g_{\alpha\beta} dx^\alpha dx^\beta$$

  - note that g_{ab} has Lorentzian signature (-1,1,1,1), and distances can be positive (spacelike), negative (timelike) or zero (null).
- The Einstein equations tell us what class of geometries are physically viable given the stress energy tensor of matter in the spacetime, and suitable boundary conditions (the rest of the lectures will use *geometric units*, in which $G=c=1$)

$$G_{\alpha\beta} = \frac{8\pi G}{c^4} T_{\alpha\beta}$$

General Relativity

- The Einstein tensor G_{ab} is constructed from the metric tensor g_{ab} as follows:

$$\Gamma^a_{bc} = \frac{1}{2} g^{ae} \left(g_{be,c} + g_{ce,b} - g_{bc,e} \right)$$

$$R_{ab} = \Gamma^d_{ab,d} - \Gamma^d_{db,a} + \Gamma^e_{ab}\Gamma^d_{ed} - \Gamma^e_{db}\Gamma^d_{ea}$$

$$G_{ab} = R_{ab} - \frac{1}{2} R \cdot g_{ab}$$

- The result then is that the Einstein equations, written in terms of the metric tensor, form a system of 10 coupled, second order, quasi-linear partial differential equations for the 10 unique components of the metric
  - these are the equations that must be solved to deduce the geometry of a scenario of interest

Mathematical Structure of the field Equations

- Written as just described, the equations have ***no*** definite mathematical character (i.e. , elliptic, hyperbolic or parabolic). This is largely due to the *general covariance* of the theory, namely that a given physical geometry can be represented in infinitely many forms, differing from each another via coordinate transformations
  - *every* representation of a given spacetime satisfies the *same* field equations
    - implies that without "fixing" the coordinates in some manner the equations are ill-posed (no uniqueness)

Mathematical Structure

- two main decompositions of the field equations
  - 3+1 (space+time)
    - typically get a coupled system of 4 elliptic (*constraint*) equations, 6 hyperbolic (*evolution*) equations, plus 4 freely specifiable gauge "equations"
    - must specify a gauge; effectively then have 10 equations for 6 remaining degrees of freedom
    - traditionally, two methods of solving this system as an initial boundary value problem
      - free evolution — solve the constraints at the initial time only, then evolve the gauge and evolution equations forwards in time
      - constrained evolution — solve the constraints at every time step, together with "2" of the evolution equations

Mathematical Structure

- two main decompositions of the field equations
  - 3+1 (space+time)
    - with either choice (free or constrained), at the analytical level (depending on the details of the particular formalism), one can show, using various identities, that a solution to the particular sub-system of equations will also solve the full system of equations
    - numerically, the above property only holds to within truncation error — this can lead to problems, in particular with free evolution schemes, that often exhibit exponential growth of constraints in generic (3D) scenarios
      - the non-linearity of the field equations prevent the constrained-transport techniques developed for the similar problem in Maxwell's from working in GR

Mathematical Structure

- two main decompositions of the field equations
  - 3+1 (space+time)
    - to date, only 2 free evolution schemes (of ~10's to 100's proposed) known to not suffer from exponential constraint growth in "generic" scenarios — *BSSN* (Baumgarte-Shapiro-Shibata-Nakamura), and *generalized harmonic*
      - in *lecture 2* will look at the ADM (Arnowitt-Deser-Misner) formalism in detail, which is the starting point of BSSN; in *lecture 4* we will briefly overview the generalized harmonic and BSSN formalisms
      - ADM, though problematic in general, works well in symmetry reduced spacetimes … will look at a spherically symmetric example

Mathematical Structure

- two main decompositions of the field equations
  - 3+1 (space+time)
  - a null, or characteristic decomposition
    - 1 or 2 of the coordinates are chosen to be to null, the rest spacelike
    - can be formulated so that there are no constraints, though the main problem is that to date there have been no serious proposals on how to deal with caustics that will generically form along the null directions
    - *Cauchy-characteristic matching* proposes to use a Cauchy (3+1) code for interior, strong-field evolution, and a characteristic code for far-field evolution

Some basic properties of solutions influencing choice of numerical solution methods

- Barring "pathological" coordinates, the geometry of most spacetimes of interest will be continuous and free of shocks, turbulence, discontinuities, etc., *even when coupled to matter that does have these features*
  - can intuitively see why from the field equations, as second derivatives of the metric couple to the stress/energy … even a delta function in the stress/energy tensor with get "smoothed out" to a C^0 function.
- Geometric singularities are inevitable in gravitational collapse, though if *cosmic censorship* holds (and no indications to-date it does not for "reasonable" matter), the singularities will always be hidden inside of black holes, and thus, via causality, cannot influence the outside universe
  - use *excision* or *moving punctures* to deal with these singularities numerically

Some basic properties of solutions influencing choice of numerical solution methods

- Perhaps the most interesting astrophysical scenarios where solving for the structure of spacetime is important involves compact objects — black holes and neutron stars — and their interactions. This typically introduces at least 3 orders of magnitude of spatio-temporal length scales that need to be resolved
  - smallest --- compact object radius
  - medium --- volume in which interesting GR interaction unfolds … eg. orbital radius for a binary
  - largest --- the far-field regime where observable quantities, such as gravitational wave emission, can be measures
    - it is difficult to unambiguously measure physically relevant properties of a geometry in the region of strong interaction … need to be far away to observe what happens

Some basic properties of solutions influencing choice of numerical solution methods

- A comment about the "right hand side"
  - "normal" matter only couples to the *local* geometry
    - mathematically, the coupling is to lower order terms in the equations, and so will not affect properties nor solution methods of the PDEs
    - physically, this is a manifestation of the equivalence principle; i.e. on sufficiently small scales the metric becomes arbitrarily close to Minkowski
  - this implies if a computational method for a matter field works in a special relativistic setting, no additional complications *should* arise incorporating it into a general relativity code

Summary of properties → numerics

- many non-linear equations with lots of terms
  - need lots of flops → parallel computing *[lectures 3 & 4]*
- principle parts of equations in a good formulation are simple wave operators and/or Laplacians, and we expect smooth solutions
  - simple basic numerical algorithms will suffice → finite differencing with Newton-Gauss-Seidel iteration
    - is generally a good smoother of the residual, so can form the back-bone of a multigrid solution of the elliptics
    - converges very rapidly to the solution given a good initial guess; for hyperbolic equations the solution at the previous time step provides this if the time step is not too large (rule of thumb: use the CFL condition even for a fully implicit scheme)
  - finite differencing with method-of-lines (Runge-Kutta) is often used for free evolution schemes; some groups also use pseudo-spectral methods

Summary of properties → numerics

- given the length scales that need to be resolved, and that often *a priori* lack of knowledge of where the small length scale features are or will evolve to, need *adaptive mesh refinement* (AMR)
  - because of the smoothness of solutions, and for situations where the small length scale features are not volume filling (eg. compact objects), a simple variant of Berger and Oliger (B&O) AMR is ideal *[lecture 3]*
  - solving coupled elliptics/hyperbolics are a bit more complicated though, in particular if the recursive time stepping is kept
    - difficulty boils down to the non-linear nature of the equations

Basics of finite difference solution of PDEs

- Denote a general differential system

$$\mathcal{L}u=f$$

where $\mathcal{L}$ is a differential operator acting on a set of unknown variables u, and f are a set of pre-specified "source" functions

- Finite-difference discretization of this system involves sampling all functions on a mesh, with characteristic mesh spacing h, and replacing *derivative* operators with *difference* operators

$$u \rightarrow u^h, \quad f \rightarrow f^h, \quad \mathcal{L} \rightarrow \mathcal{L}^h$$

Basics of finite difference solution of PDEs

- Some common second order accurate difference operators:

$$\frac{\partial u(x,t)}{\partial x} = \frac{u_{i+1}^n - u_{i-1}^n}{h} + O(h^2)$$

$$\frac{\partial u(x,t)}{\partial t} = \frac{u_i^{n+1} - u_i^{n-1}}{\lambda h} + O(h^2)$$

$$\frac{\partial u^2(x,t)}{\partial x^2} = \frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{h^2} + O(h^2)$$

- here, the lower (upper) index labels space (time), $h=\Delta x$ and $\lambda=\Delta x/\Delta t$ is the CFL (Courant-Fredrichs-Lewy) factor

Basics of finite difference solution of PDEs

- From the form of the difference operators, one can see that converting a system of PDEs to difference equations results in a system of (possibly non-linear) algebraic equations, one set of equations at *each* mesh point in the computational domain; represent this system of equations via

$$\mathcal{L}^h u^h = f^h$$

- One step of a numerical algorithm thus involves solving this system of equations for the unknowns u_i^{n+1} at the advanced timed level n, given "initial data" at past time levels n, n-1, … etc.

Important Concepts/Definitions in FD

- The solution error is

$$e^h = u - u^h$$

- Given a solution u to the continuum differential equations, the truncation error is

$$\tau^h = \mathcal{L}^h u - f^h$$

- The discrete solution converges to the continuum solution if and only if

$u^h \rightarrow u$ in the limit $h \rightarrow 0$

Important Concepts/Definitions in FD

- A necessary condition for convergence is for the particular FD approximation to be consistent

$$\tau^h \to 0 \text{ in the limit } h \to 0$$

- The order p of a consistent FD scheme is the rate at which the truncation error scales to zero with decreasing mesh spacing

$$\tau^h = O(h^p) \text{ in the limit } h \to 0$$

Important Concepts/Definitions in FD

- In many cases one can not, or it is too expensive, to solve the difference equations exactly, and some iterative method gives an approximate solution $\underline{u}^h$. The residual $\mathcal{R}^h$ is then

$$\mathcal{R}^h = \mathcal{L}^h \underline{u}^h - f^h$$

- A FD scheme is stable if the norm of the solution is bounded in time by *some* exponential

$$|u^h| < e^{ct}$$

with c a constant *independent* of h. A *consistent* FD scheme is stable if and if only if it converges. Note for differential equations that develop singular solutions, the notion of stability is still valid over some time interval prior to the formation of the singularity

Important Concepts/Definitions in FD

- In situations of interest one will not know what the solution error nor truncation error is. In practice then, for a stable, consistent pth order numerical scheme one *assumes* the solution admits a Richardson expansion:

$$u^h = u + e_1 h^p + e_2 h^{2p} + \ldots$$

Here, the e_1, e_2, ... are error functions (i.e. of space and time), but are independent of mesh spacing

- For simple equations, difference schemes and boundary conditions, one can prove that a Richardson expansion exists; for more complication equations (like the Einstein equations) it empirically does if the finite difference approximates are consistent, and the solutions are smooth

Important Concepts/Definitions in FD

- The Richardson expansion allows for two of the most important tools-of-the-trade in computational physics; convergence testing and error estimation. These are performed by comparing solutions obtained at different resolutions

- A standard convergence test uses 3 resolutions: h, $2h$ and $4h$:

$$\left\| \frac{u^{4h} - u^{2h}}{u^{2h} - u^h} \right\| = 2^p + O(h^p)$$

Important Concepts/Definitions in FD

- If you see convergence, you have (1) a rather non-trivial consistency check that the assumed Richardson does hold, and (2) are adequately resolving the problem so that the solution you obtain is a decent approximation to the continuum solution; specifically then, an estimate of the error in the finest resolution result is

$$e^h = \frac{u^{2h} - u^h}{2^p - 1} + O(h^p)$$

- Similar error estimates can be derived for properties extracted from the solution; e.g. gravitational waves, total energy, etc.

Black Holes

- Black holes are one of the more important and interesting consequences of general relativity. To get some experience in how to numerically deal with black holes in a simulation, we will look at wave propagation on a Schwarzschild background
- The common form the Schwarzschild metric is usually presented in is

$$ds^2 = -\left(1 - \frac{2m}{r}\right)dt^2 + \left(1 - \frac{2m}{r}\right)^{-1} dr^2 + r^2\left(d\theta^2 + \sin^2\theta d\phi^2\right)$$

the ("true") geometric singularity is at r=0; though this form of the metric also has a coordinate singularity at r=2m, which can be problematic.

Black Holes

- There are many forms of the Schwarzschild metric that are regular on the horizon; we will use the metric in *ingoing-Eddington-Finkelstein* coordinates, written in terms of ADM variables (defined next lecture)

$$ds^2 = -\left(\alpha^2 + a^2\beta^2\right)dt^2 + 2a^2\beta drdt + a^2dr^2 + r^2\left(d\theta^2 + \sin^2\theta d\phi^2\right)$$

$$\alpha = \sqrt{\frac{r}{r+2m}};\ \beta = \frac{2m}{r+2m};\ a = \frac{1}{\alpha}$$

- It is easy to check that the metric is non-singular at $r=2m$

Scalar wave propagation on a curved background

- The covariant wave equation for a massles scalar field ϕ on a general background metric with metric g is

$$\nabla^\alpha \nabla_\alpha \phi = \frac{1}{\sqrt{-g}}\partial_\alpha\left(\sqrt{-g}\, g^{\alpha\beta}\partial_\beta\phi\right) = 0$$

- For wave propagation on a Schwarzschild background we will restrict attention to a spherically symmetric scalar fields $\phi=\phi(r,t)$. It is also convenient to reduce the wave equation to first order form by introducing the "conjugate" variables Φ and Π

$$\Phi(r,t) \equiv \partial_r\phi$$

$$\Pi(r,t) \equiv \frac{a}{\alpha}\left(\partial_t\phi - \beta\partial_r\phi\right)$$

Scalar wave propagation on a curved background

- In terms of Φ and Π the wave equation becomes

$$\partial_t \Pi = \frac{1}{r^2}\partial_r\left(r^2\left[\beta\Pi + \frac{\alpha}{a}\Phi\right]\right)$$

and from the definitions of Π, Φ one can derive an evolution equation for Φ

$$\partial_t \Phi = \partial_r\left(\beta\Phi + \frac{\alpha}{a}\Pi\right)$$

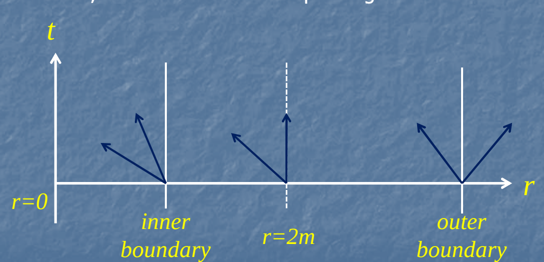
Complication with black holes

- The wave equation on a general background metric with Lorentz signature is hyperbolic, and can "easily" be solved with standard techniques, in particular in a black hole spacetime if the domain excludes the geometric singularity at r=0.
- However, the one complication arising with the black hole is the event horizon, due to its nature as a one way boundary to propagation of causal signals, such as waves
- This property of the spacetime manifests itself in the *characteristics* of the wave equation; here there are two radial modes with velocities

$$v_{\pm} = -\beta \pm \frac{\alpha}{a} = \frac{-2m \pm r}{r+2m}$$

Characteristic structure

- I.e., as the event horizon is crossed the light cone "tips over", with all characteristics pointing into the black hole



Excision

- When solving the initial boundary value problem for hyperbolic equations, one can *only* place boundary conditions on degrees of freedom that are propagating *into* the computational domain
- Inside the event horizon, there are *no* such modes, and hence one *cannot* place boundary conditions there. Instead, one must solve the equations at the inner boundary, with central spatial difference operators replaced with sideways operators, .e.g:

$$\frac{u^n_{i+1} - u^n_{i-1}}{h} \rightarrow \frac{-u^n_{i+2} + 4u^n_{i+1} - 3u^n_i}{2h}$$

- This is the idea behind *black hole excision*, and since the interior is causally disconnected from the exterior spacetime, not solving for the interior structure of the fields will have no effect on the exterior solution

Comment on dissipation

- often unwanted high-frequency solution components ("noise") arise with certain FD operators, at refinement boundaries, the excision surface, near the axis in a spherical or axisymmetric code, etc.
  - at best could cause unphysical reflections, at worst be unstable
- Kreiss-Oliger (KO) style dissipation is very effective at reducing these high frequency parts of the solution. The stencil for this high-pass filter (which is used to *subtract* the high-frequency components from a field) is

$$\frac{\varepsilon}{16}\left(f_{i+2}-4f_{i+1}+6f_i-4f_{i-1}+f_{i-2}\right)$$

  - this is the undivided, centered, second order accurate 4^{th} derivative operator multiplied by a parameter ε (<1 for stability). It's frequency response is $\varepsilon\sin^4(\xi/2)$ for wave number ξ, and so completely filters out waves at the Nyquist limit when $\varepsilon=1$.

Comments on KO dissipation

- NOTE: KO dissipation is not artificial viscosity … it only alters the solution at the level of the truncation error (I.e., it's effects "converge away")
- Also, typical 2^{nd} order finite difference schemes get the phase and amplitude evolution of such high frequency components of the solution completely wrong in any case, and so does not hurt to get rid of them
- practically, the cost is one might need up to a factor of two more resolution (if large values of ε are required) to obtain equivalent solution error compared to stable a non-dissipative scheme

Solving Hyperbolic PDEs with RNPL

- RNPL (Rapid Numerical Prototyping Language), written by Robert Marsa and Matthew Choptuik, is a tool designed to aid in the construction of programs to solve hyperbolic PDEs
- Advantages
  - provides simple and natural notation to define difference operators, and to write the difference equations in a form close to the analytic expressions
  - takes care of much of the program infrastructure unrelated to the numerics, in particular file I/O, checkpointing and reading run-time parameters
  - Implements a 1-step Newton-Gauss-Seidel (NGS) relaxation algorithm, which is powerful enough to solve a large class of non-linear hyperbolic difference equations

Solving Hyperbolic PDEs with RNPL

- Disadvantanges
  - though designed with GR applications in mind, does not provide native support for excision; notation makes it "messy" to implement excision in 2 & 3D
  - does not provide easy mechanism to implement alternative time-steppers, such as Runge Kutta
  - would greatly benefit from more features, including support for elliptic PDEs, AMR, parallel execution (though does provide a mechanism to interface with external routines to extend capabilities)
  - for class of PDEs that it works well for, temptation to use as a "black box", which can be "dangerous"

One step Newton-Gauss-Seidel iteration

- Perhaps the most important thing to understand about RNPL's inner workings is the NGS relaxation scheme
- Given a specification of a uniform mesh, a set of differences operators, and a set of PDEs and boundary conditions in the form

  $$\mathcal{L}^h u^h = f^h$$

  RNPL constructs a residual for each equation (interior or boundary) at each grid point:

  $$\mathcal{R}^h = \mathcal{L}^h \underline{u}^h - f^h$$

- Again, at each grid point we have a set of non-linear algebraic equations (equal to the number of unknown functions), that could in general depend on discretized values of all variables at all grid points and all time levels

One step Newton-Gauss-Seidel iteration

- Any single step of a numerical solution scheme can be thought of as an algorithm that finds values of the unknowns so that residual is zero, given known values of the grid functions at past time levels
- 1 step NGS attempts to drive the residual to zero by the following iteration
  - (i) linearize the equations about the current guess $\underline{u}$ for the unknowns (dropping the "h" superscript denoting the mesh spacing, and using a subscript to label equation/variable as appopriate)

    $$\mathcal{R}_i(\underline{u} + \delta\underline{u}) = \mathcal{R}_i(\underline{u}) + \partial_j \mathcal{R}_i(\underline{u})\, \delta\underline{u}_j + O((\delta\underline{u})^2)$$

  - (ii) obtain an approximation to the correction $\delta\underline{u}$ performing 1 step of a Gauss-Seidel iterative solution of the resulting linear system
    - approximates the Jacobian matrix $\partial_j \mathcal{R}_i$ with its the diagonal (i.e., ignores couplings between equations and between adjacent mesh points)
    - in the inner loop of the iteration, immediately replaces the unknown with its next guess before moving on to the next variable

One step Newton-Gauss-Seidel iteration

- This scheme converges quite rapidly for hyperbolic equations, given a good initial guess
  - since hyperbolic equations have finite propagation speeds, using the solution from the previous time step is a good initial guess
  - will converge in 1 step for an explicit, linear equation
  - typically can drive the residual to below truncation error with several iterations for the Einstein equations
- Easiest to introduce RNPL notation by looking at a sample code … see:
- http://physics.princeton.edu/~fpretori/group_resources/index.html