Contents

MAYBE PUT SOME STUFF HERE ABOUT NUMERICAL ALGORITHMS LIKE RK4 EULAR STEP LEAPFROG AND DO ERROR GROWTH RATES AND COURANT

0.1 Numerical Relativity

0.1.1 Spacetime Foliation

Einstein's equation is a classical field equation which governs the dynamics of physical objects and spacetime curvature.

$$R_{\mu\nu} - \frac{1}{2}Rg_{\mu\nu} = \frac{8\pi G}{c^4}T_{\mu\nu} \tag{0.1.1}$$

The above version is fully covariant, agnostic of the definition of time, and many solutions are known analytically, for instance Black Hole geometries. When the system of interest becomes more complicated, such as the case of orbiting objects which will be discussed later, finding an analytic expression becomes impossible. For low energy dynamics, Newtonian theory, Post-Newtonian theory and perturbation theory can make more progress; however this report will focus on the highly nonlinear regime where Numerical relativity is truly the only hope to solve Einstein's equations. To do this it is common to split spacetime into 3+1 dimensions, evolving a 3 dimensional manifold (maybe with matter) on a computer along the final 4th dimension. To do this we need to define a suitable hypersurface $\Sigma \in \mathcal{M}$. This is usually done by demanding the hypersurface Σ_t be the set of points $p \in \mathcal{M}$ where some scalar function $f : \mathcal{M} \mapsto \mathbb{R}$ satisfies $f(p) = t$. This hypersurface should be a Cauchy surface, intersecting all causal curves only once, or a partial Cauchy surface which intersects all causal curves at most once. Generally we will choose a partial Cauchy surface due to the finite memory of computers, however by picking certain compactified coordinates it is possible to use a Cauchy surface [ref]. A foliation $\mathcal{F}$ is then the union of a set of Σ_t for some range of the parameter t.

$$\mathcal{F} = \cup_t(\Sigma_t) \subseteq \mathcal{M} \tag{0.1.2}$$

This means we should be careful to pick a parameter t such that the foliation is not self intersecting for the parameter range that covers the region of $\mathcal{M}$ that we are interested in simulating. Fortunately the time coordinate in a suitable coordinate system works in all cases covered by this report; it also gives the physical interpretation of Σ_t being an instance of time. Now we should define unit normal vector n to Σ_t.

$$n^\mu = -\frac{\nabla^\mu t}{\sqrt{|g_{\mu\nu}\nabla^\mu t \nabla^\nu t|}} \quad \& \quad n_\mu = -\frac{dt_\mu}{\sqrt{|g_{\mu\nu}\nabla^\mu t \nabla^\nu t|}} \tag{0.1.3}$$

For simplicity we define the lapse function α to be

$$\alpha := \frac{1}{\sqrt{|g_{\mu\nu}\nabla^\mu t \nabla^\nu t|}} \tag{0.1.4}$$

giving us $n_\mu = -\alpha \mathrm{d}t_\mu$ as well as the normal evolution vector $m_\mu = \alpha n_\mu$. Defining two infinitessimaly close points $(p, q) \in (\Sigma_t, \Sigma_{t'})$ where $q^\mu = p^\mu + m^\mu \delta t$ we see

$$t(q) = t(p^\mu + m^\mu \delta t) = t(p) + \frac{\mathrm{d}t}{\mathrm{d}x^\mu} m^\mu \delta t = t(p) + \mathrm{d}t_\mu m^\mu \delta t = t(p) + \delta t \tag{0.1.5}$$

showing that m^μ points between neighbouring hypersurfaces; therefore when creating evolution equations we should care about Lie derivates along m^μ, $\mathcal{L}_m$, rather than $\mathcal{L}_n$.

0.1.2 The 3+1 Decomposition

With the notion of a spacetime foliation we should define how to project tensors onto Σ_t; clearly scalars need no projecting. Splitting a vector $X^\mu e_\mu = X^\mu_\parallel e_\mu + X^\mu_\perp e_\mu$ into components tangent or normal to Σ_t

we define the orthogonal projector $\perp^\mu_\nu$ and parallel projector $-n^\mu n_\nu$.

$$X^\mu_{\|} = \left[\delta^\mu_\nu + n^\mu n_\nu\right] X^\mu = \perp^\mu_\nu X^\nu \tag{0.1.6}$$

$$X^\mu_{\perp} = -n^\mu n_\nu X^\nu \tag{0.1.7}$$

Considering scalars such as $\phi = w_\mu X^\mu$ or $\psi = T^{\mu\nu} w_\mu w_\nu$, and remembering scalars don't vary under projection, it is simple to show that any tensor T can be projected by contracting a projection operator $\perp$ on any free index.

$$T_{\|}{}^{ab\ldots}{}_{cd\ldots} = \mathcal{T}^{ab\ldots}{}_{cd\ldots} = \perp^a_A \perp^b_B \perp^C_c \perp^D_d \cdots T^{AB\ldots}{}_{CD\ldots} \tag{0.1.8}$$

We can find the 3 metric $\gamma_{\mu\nu}$ of Σ_t by projecting $g_{\mu\nu}$

$$\gamma_{ij} = \perp^\mu_i \perp^\nu_j g_{\mu\nu} = g_{\mu\nu} + n_\mu n_\nu \quad \rightarrow \quad \gamma^i_j = \perp^i_j \tag{0.1.9}$$

and we find it is equivalent to the projector $\perp$; this had to be the case as $\perp_{ij} \, \mathrm{d}x^i \mathrm{d}x^j$ gives the line element along Σ_t. With this machinery we can define the extrinsic curvature tensor $\mathcal{K}_{ij}$ representing curvature due to the choice of spacetime foliation; it could be nonzero for certain foliations of Minkowski space. It is not the same as the 3-Ricci tensor $\mathcal{R}_{ij}$ which is due to genuine spacetime curvature of $\mathcal{M}$ regardless of foliation. The extrinsic curvature tensor is defined the following way

$$\mathcal{K}_{ij} = \mathcal{K}_{ji} := - \perp^\mu_i \perp^\nu_j \nabla_\mu n_\nu = - \perp^\mu_i \nabla_\mu n_j = -\nabla_i n_j - n_i a_j \tag{0.1.10}$$

$$\mathcal{K} = \mathcal{K}^i_i = -\nabla \cdot n \tag{0.1.11}$$

where $a_i = n \cdot \nabla n_i$ is called the Eularian acceleration; it should be noted $\mathcal{K}_{ij}$ is symmetric. It can also be shown to take the following form

$$\mathcal{K}_{ij} = -\frac{1}{2}\mathcal{L}_n \gamma_{ij} = -\frac{1}{2\alpha}\mathcal{L}_m \gamma_{ij} \tag{0.1.12}$$

which gives the intuitive explanation of $\mathcal{K}_{ij}$ being the rate of change of the 3-metric γ_{ij} with respect to the foliation. The next object we should discuss is the projected covariant 3-derivative $\mathcal{D}_i$. This is the covariant derivative belonging to Σ_t and hence it's arguments should be tensors belonging to Σ_t. This means we can define it as so

$$\mathcal{T}^{ab\ldots}{}_{cd\ldots} = \perp^a_A \perp^b_B \perp^C_c \perp^D_d \cdots T^{AB\ldots}{}_{CD\ldots}, \tag{0.1.13}$$

$$\mathcal{D}_\mu \mathcal{T}^{ab\ldots}{}_{cd\ldots} : = \perp^\nu_\mu \perp^a_A \perp^b_B \perp^C_c \perp^D_d \nabla_\nu \mathcal{T}^{AB\ldots}{}_{CD\ldots}. \tag{0.1.14}$$

A simple example is the derivative of a vector $X^i e_i \in \mathcal{T}(\Sigma_t)$

$$\mathcal{D}_i X^j = \partial_i X^j + \Upsilon^i{}_{jk} X^k$$

$$\Upsilon^i{}_{jk} = \frac{1}{2}\gamma^{il}\left[\partial_j \gamma_{lk} + \partial_k \gamma_{jk} - \partial_l \gamma_{jk}\right] \tag{0.1.15}$$

where $\Upsilon^i{}_{jk}$ is the Christoffel symbol of Σ_t. Another useful example is a^μ which can be equated to

$$a_\mu = n \cdot \nabla n_\mu = \mathcal{D}_\mu \ln \alpha = \frac{1}{\alpha}\mathcal{D}_\mu \alpha \tag{0.1.16}$$

and allows us to evaluate the Lie derivative of the projector $\perp^i_j$.

$$\mathcal{L}_m \perp^i_j = \alpha n^k \nabla_k \perp^i_j + \perp^i_k \nabla_j \alpha n^k - \perp^k_j \nabla_k \alpha m^i = \alpha n^k \nabla_k \left[n^i n_j\right] + \alpha \nabla_j n^i - \left[\alpha K^i_j + n^i \mathcal{D}_j \alpha\right] = 0$$

The result $\mathcal{L}_m \perp^i_j = 0$ is incredibly important, it tells us that the projector commutes with $\mathcal{L}_m$ and as a result any tensor T which when projected onto Σ_t, written $\mathcal{T}$, satisfies

$$\mathcal{L}_m \mathcal{T}^{ab\ldots}{}_{cd\ldots} = \perp^a_A \perp^b_B \perp^C_c \perp^D_d \mathcal{L}_m \mathcal{T}^{AB\ldots}{}_{CD\ldots}. \tag{0.1.17}$$

or in other words, evolving a projected tensor along integral curves of m leaves the tensor parallel to Σ_t.

0.1.3 Gauss, Codazzi and Ricci Equations

We now have all the results we need to decompose the Riemann and Ricci tensors and scalar. From the definition of the Riemann tensor we know

$$[\mathcal{D}_\alpha \mathcal{D}_\beta - \mathcal{D}_\beta \mathcal{D}_\alpha] v^\gamma = \mathcal{R}^\gamma{}_{\lambda\alpha\beta} v^\lambda,$$
$$[\nabla_\alpha \nabla_\beta - \nabla_\beta \nabla_\alpha] v^\gamma = R^\gamma{}_{\lambda\alpha\beta} v^\lambda,$$

where the vector $v^\lambda = \perp^\lambda_\rho v^\rho$ is tangent to Σ_t. Expanding the $\mathcal{D}$'s into ∇'s gives

$$\mathcal{D}_\alpha \mathcal{D}_\beta v^\gamma = \perp^\mu_\alpha \perp^\sigma_\beta \perp^\gamma_\xi \nabla_\mu (\perp^\nu_\sigma \perp^\xi_\rho \nabla_\nu v^\rho) \tag{0.1.18}$$

and using the following properties; impotence of projections $\perp^\alpha_\mu \perp^\mu_\beta = \perp^\alpha_\beta$, null projection of orthogonal vectors $\perp n = 0$, metric compatibility $\nabla_\mu \perp^\alpha_\beta = n_\beta \nabla_\mu n^\alpha + n^\alpha \nabla_\mu n_\beta$ and equation () for $\mathcal{K}_{ij}$ we obtain the Gauss relation.

$$\perp^\mu_\alpha \perp^\nu_\beta \perp^\gamma_\rho \perp^\sigma_\delta R^\rho{}_{\sigma\mu\nu} = \mathcal{R}^\gamma{}_{\delta\alpha\beta} + \mathcal{K}^\gamma_\alpha \mathcal{K}_{\delta\beta} - \mathcal{K}^\gamma_\beta \mathcal{K}_{\alpha\delta} \tag{0.1.19}$$

Contracting over α, γ above and relabelling indices we get the contracted Gauss relation.

$$\perp^\mu_\alpha \perp^\nu_\beta R_{\mu\nu} + \gamma_{\alpha\mu} n^\nu \perp^\rho_\beta n^\sigma R^\mu{}_{\nu\rho\sigma} = \mathcal{R}_{\alpha\beta} + \mathcal{K}\mathcal{K}_{\alpha\beta} - \mathcal{K}^\mu_\beta \mathcal{K}_{\alpha\mu} \tag{0.1.20}$$

Contracting again and realising $R_{abcd} n^a n^b n^c n^d = 0$ from antisymmetry in indeces 0,1 or 2,3 in the Riemann tensor gives the scalar Gauss equation.

$$R + 2R_{\mu\nu} n^\mu n^\nu = \mathcal{R} + \mathcal{K}^2 - \mathcal{K}_{\mu\nu} \mathcal{K}^{\mu\nu} \tag{0.1.21}$$

The Codazzi relations are derived from a different start point

$$[\nabla_\alpha \nabla_\beta - \nabla_\beta \nabla_\alpha] n^\gamma = R^\gamma{}_{\lambda\alpha\beta} n^\lambda$$

and project down with 3 operators $\perp\perp\perp$. The following relations are used

$$\nabla_\beta n^\gamma = -\mathcal{K}^\gamma{}_\beta - a^\gamma n_\beta,$$
$$\perp^\alpha_\mu \perp^\beta_\nu \perp^\rho_\gamma \nabla_\alpha \nabla_\beta n^\gamma = -\mathcal{D}_\alpha \mathcal{K}^\gamma{}_\beta + a^\gamma \mathcal{K}_{\alpha\beta},$$

which lead immediately to the Codazzi relation

$$\perp^\mu_\alpha \perp^\nu_\beta \perp^\gamma_\rho n^\sigma R^\rho{}_{\sigma\mu\nu} = \mathcal{D}_\beta \mathcal{K}^\gamma{}_\alpha - \mathcal{D}_\alpha \mathcal{K}^\gamma{}_\beta \tag{0.1.22}$$

and the contracted Codazzi relation.

$$\perp^\mu_\alpha n^\nu R_{\mu\nu} = \mathcal{D}_\alpha \mathcal{K} - \mathcal{D}_\mu \mathcal{K}^\mu{}_\alpha \tag{0.1.23}$$

Finally we turn our attention to the Ricci equation. We start with the Riemann term in the contracted Gauss equation,

$$\gamma_{\alpha\mu} \perp^\rho_\beta n^\sigma R^\mu{}_{\nu\rho\sigma} n^\nu = \gamma_{\alpha\mu} \perp^\rho_\beta n^\sigma \left[\nabla_\rho \nabla_\sigma - \nabla_\sigma \nabla_\rho\right] n^\nu \tag{0.1.24}$$

and using the following two formulae with equation (D ln a)

$$\nabla_\sigma n^\mu = -\mathcal{K}^\mu{}_\sigma - n^\mu \mathcal{D}_\sigma \ln(\alpha) \tag{0.1.25}$$

$$\mathcal{L}_m \mathcal{K}_{ij} = \perp^a_i \perp^b_j \mathcal{L}_m \mathcal{K}_{ab} = \perp^a_i \perp^b_j \left[\alpha n \cdot \nabla \mathcal{K}_{ab} + 2\mathcal{K}_{k(b} \nabla_{a)} m^k\right] \tag{0.1.26}$$

some algebra yields

$$\gamma_{\alpha\mu} n^\sigma \gamma^\nu_\beta R^\mu{}_{\rho\nu\sigma} n^\rho = -\mathcal{K}_{\alpha\sigma} \mathcal{K}^\sigma{}_\beta + \frac{1}{\alpha} \mathcal{D}_\beta \mathcal{D}_\alpha \alpha + \gamma^\mu_\alpha \gamma^\nu_\beta n^\sigma \nabla_\sigma \mathcal{K}_{\mu\nu}.$$

From () we know $\mathcal{L}_m \mathcal{K}_{\alpha\beta} \in \Sigma_t$ and the projected Lie derivative is

$$\mathcal{L}_m \mathcal{K}_{\alpha\beta} = \alpha \gamma^\mu_\alpha \gamma^\nu_\beta n^\sigma \nabla_\sigma \mathcal{K}_{\mu\nu} - 2\alpha \mathcal{K}_{\alpha\mu} \mathcal{K}^\mu{}_\beta.$$

This is what we need to derive the Ricci equation.

$$\gamma_{\alpha\mu} n^\sigma \gamma^\nu_\beta R^\mu{}_{\rho\nu\sigma} = \frac{1}{\alpha} \mathcal{L}_m \mathcal{K}_{\alpha\beta} + \frac{1}{\alpha} \mathcal{D}_\beta \mathcal{D}_\alpha \alpha + \mathcal{K}_{\alpha\mu} \mathcal{K}^\mu{}_\beta \tag{0.1.27}$$

0.1.4 Decomposition of Einstein's Equation

To evolve General Relativity numerically we must project the Einstein Equation into 3+1 dimensions. Relations between 3 and 4 dimensional geometric objects have been derived above and will be used to decompose the Einstein tensor $G_{\mu\nu}$ from the left hand side of (). The second component, for simulating non-vacumm spacetimes, is the 3+1 decomposition of the Stress tensor T_{ab}. We contract twice with n, then once with n while projecting onto Σ_t and finally twice projecting onto Σ_t to get an energy, momentum and stress-like split.

$$\begin{aligned} \mathcal{E} &= T(n,n) = T_{ab}n^a n^b && (0.1.28) \\ \mathcal{S}_i &= -T(n,\cdot) = -\perp_i^a n^b T_{ab} && (0.1.29) \\ \mathcal{S}_{ij} &= \perp_i^a \perp_j^b T_{ab} && (0.1.30) \end{aligned}$$

By construction the following is true.

$$T_{ab} = \mathcal{E}n_a n_b + \mathcal{S}_a n_b + \mathcal{S}_b n_a + \mathcal{S}_{ab} \tag{0.1.31}$$

With this and the Gauss-Codazzi equations () we can project the Einstein equation. Lets first look at the scalar equation

$$G_{\mu\nu}n^\mu n^\nu = R_{\mu\nu}n^\mu n^\nu + \frac{1}{2}R = 8\pi\mathcal{E} \tag{0.1.32}$$

and equating the geometric terms to the scalar Gauss equation () we get the Hamiltonian constraint, $\mathcal{H} = 0$.

$$\mathcal{H} = \mathcal{K}_{\mu\nu}\mathcal{K}^{\mu\nu} - \mathcal{K}^2 - \mathcal{R} + 16\pi\mathcal{E} \tag{0.1.33}$$

Now looking at the once projected part we see

$$\perp_i^\mu n^\nu G_{\mu\nu} = \perp_i^\mu n^\nu R_{\mu\nu} = -8\pi\mathcal{S}_i \tag{0.1.34}$$

and substituting the geometric terms for the contracted Codazzi relation we get the Momentum constraint, $\mathcal{P}_i = 0$.

$$\mathcal{P}_i = \mathcal{D}_i\mathcal{K} - \mathcal{D}_j\mathcal{K}_i^j + 8\pi\mathcal{S}_i \tag{0.1.35}$$

Finally, the space-space projection gives the 6 evolution PDE's. This time start with the trace reversed Einstein Equation

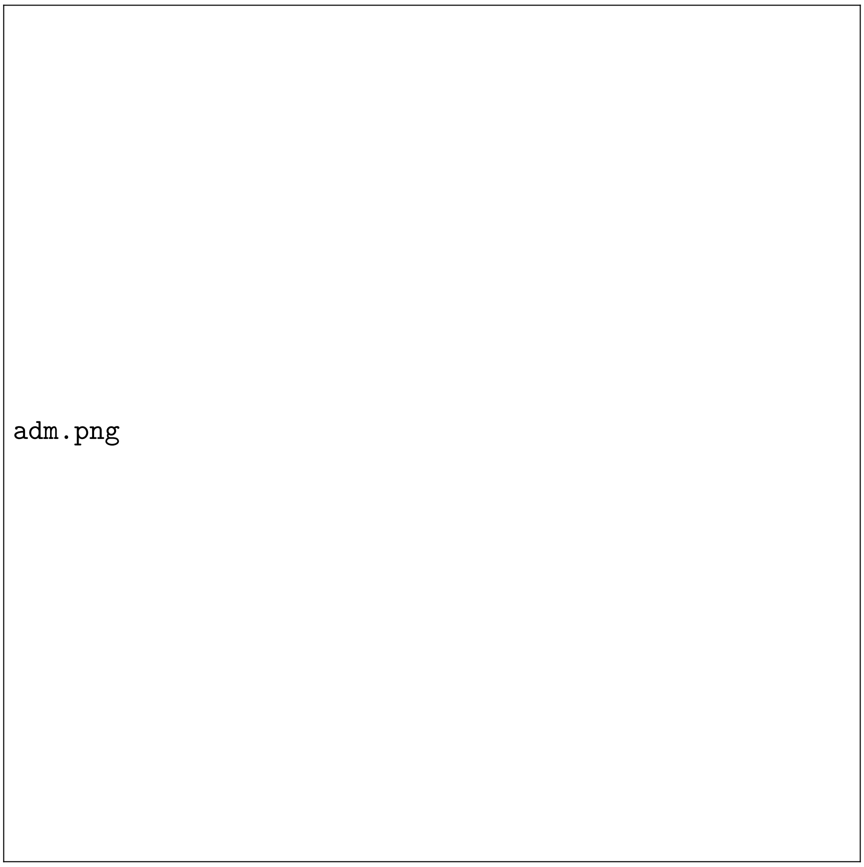
$$\begin{aligned} R_{\mu\nu} &= 8\pi\left[T_{\mu\nu} - \frac{1}{2}Tg_{\mu\nu}\right] \\ \perp_i^\mu \perp_j^\nu R_{\mu\nu} &= 8\pi\left[\mathcal{S}_{ij} - \frac{1}{2}T\gamma_{ij}\right] \\ T &= \left[\gamma^{\mu\nu} - n^\mu n^\nu\right]T_{\mu\nu} = \mathcal{S} - \mathcal{E} \end{aligned}$$

and to project the Ricci tensor, we use the contracted Gauss equation () but replace the term with $R^a{}_{bcd}$ with the Ricci equation (). Rearranging gives a normal evolution for the extrinsic curvature.

$$\mathcal{L}_m\mathcal{K}_{ij} = -\mathcal{D}_j\mathcal{D}_i\alpha + \alpha\left[\mathcal{R}_{ij} + \mathcal{K}\mathcal{K}_{ij} - 2\mathcal{K}_i^k\mathcal{K}_{kj} + 4\pi\left[\gamma_{ij}\left[\mathcal{S} - \mathcal{E}\right] - 2\mathcal{S}_{ij}\right]\right] \tag{0.1.36}$$

Along with the definition of $\mathcal{K}_{ij}$ in equation (), this gives the normal evolution equations for γ_{ij} and $\mathcal{K}_{ij}$. These contain 6 of the second derivatives in Einstein's equation, with the other 4 being the constraint equations to be satisfied along evolution.

0.1.5 Foliation Adapted Coordinates, CHECK ERORR IN ADM METRIC MATRIX



r0.65 Picking coordinates in GR introudces a large gauge freedom, this allows us to use a level set of the time coordinate $x^0 = t$ to define our foliation hypersurfaces Σ_t. The other three coordinates x^i for $i \in [1, 2, 3]$ can be used to span each hypersurface Σ_t however we define. It is conventional to split the normal evolution vector m^μ into time t^μ and space parts β^μ.

$$m^\mu = t^\mu - \beta^\mu = (\partial_0)^\mu - \beta^i(\partial_i)^\mu \tag{0.1.37}$$

$$m^\mu = \left(1, -\beta^1, -\beta^2, -\beta^3\right) \tag{0.1.38}$$

We can view t^μ as the (not necessarily causal) worldline for a simulation gridpoint, hence we would like to evolve our PDE's along t^μ on a computer introducing advection terms like $\mathcal{L}_\beta$. Equations (def of m) specify n^μ and n_μ

$$n^\mu = \frac{1}{\alpha}\left(1, -\beta^1, -\beta^2, -\beta^3\right) \quad \& \quad n_\mu = -\alpha\,(1, 0, 0, 0) \tag{0.1.39}$$

as well as the decomposed metric, using the property that β is tangent to Σ_t and orthogonal to m.

$$g_{00} = g(\partial_0, \partial_0) = g(m + \beta^i\partial_i, m + \beta^j\partial_j) = g(m, m) + \beta^i\beta_j\langle\partial_i, \mathrm{d}x^j\rangle = -\alpha^2 + \beta^i\beta_i$$
$$g_{0i} = \beta_i, \quad g_{ij} = g(\partial_i, \partial_j) = \gamma(\partial_i, \partial_j) = \gamma_{ij}$$

This is commonly called the $3+1$ ADM metric and α, β^i are referred to the lapse and shift vector in this context. The metric is commonly written as follows.

$$\mathrm{d}s^2 = -\alpha^2\mathrm{d}t^2 + \gamma_{ij}\left[\mathrm{d}x^i + \beta^i\mathrm{d}t\right]\left[\mathrm{d}x^j + \beta^j\mathrm{d}t\right] \tag{0.1.40}$$

$$g_{\mu\nu} = \begin{pmatrix} -\alpha^2 + \beta^i\beta_i & \beta_i \\ \beta_j & \gamma_{ij} \end{pmatrix} \quad \leftrightarrow \quad g^{\mu\nu} = \begin{pmatrix} -\alpha^{-2} & -\alpha^{-2}\beta^i \\ -\alpha^{-2}\beta^j & \gamma^{ij} - \alpha^{-2}\beta^i\beta^j \end{pmatrix} \tag{0.1.41}$$

and using Cramers rule for metric determinant

$$g^{00} = \frac{\det \gamma_{ij}}{\det g_{\mu\nu}} \tag{0.1.42}$$

we get the important relationship

$$\sqrt{-g} = \alpha\sqrt{\gamma}. \tag{0.1.43}$$

0.1.6 ADM Equations

Now that we have some coordinates suitable for the spacetime foliation we can find the ADM evolution equations for $\mathcal{K}_{ij}$ and γ_{ij}. First expand the Lie derivative $\mathcal{L}_m$ using

$$t^\mu = (1,0,0,0) \rightarrow \mathcal{L}_t T^{ab\ldots}{}_{cd\ldots} = \partial_t T^{ab\ldots}{}_{cd\ldots} \tag{0.1.44}$$

$$\mathcal{L}_m = \mathcal{L}_t - \mathcal{L}_\beta = \partial_t - \mathcal{L}_\beta \tag{0.1.45}$$

and the ADM equations can be written by expanding the $\mathcal{L}_m$ in equations ().

$$\partial_t \mathcal{K}_{ij} = \mathcal{L}_\beta \mathcal{K}_{ij} - \mathcal{D}_j\mathcal{D}_i\alpha + \alpha\left[\mathcal{R}_{ij} + \mathcal{K}\mathcal{K}_{ij} - 2\mathcal{K}_i^k\mathcal{K}_{kj} + 4\pi\left[\gamma_{ij}\left[\mathcal{S} - \mathcal{E}\right] - 2\mathcal{S}_{ij}\right]\right] \tag{0.1.46}$$

$$\partial_t \gamma_{ij} = \mathcal{L}_\beta \gamma_{ij} - 2\alpha\mathcal{K}_{ij} \tag{0.1.47}$$

Unfortunately, these PDE's turn out to be ill-posed []; the resolution comes in the next section.

0.1.7 BSSN

To tackle the ill-posedness of the ADM equations () the Baumgarte-Shapiro-Shibata-Nakamura (BSSN) formalism is introduced []. Another strongly hyperbolic formalism is the generalised harmonic gauge [?] with $\Box x^\mu = H^\mu$. The first step in BSSN is to decompose the 3-metric into the conformal metric $\tilde{\gamma}_{ij}$ and the conformal factor χ

$$\tilde{\gamma}_{ij} = \chi\gamma_{ij} \quad \rightarrow \quad \det\tilde{\gamma}_{ij} = \tilde{\gamma} = \chi^3\gamma = 1 \tag{0.1.48}$$

with the above being the convention used in GRChombo; other conventions include factors such as

$$\tilde{\gamma}_{ij} = \psi^{-4}\gamma_{ij} \quad \text{or} \quad e^{-\phi}\gamma_{ij}. \tag{0.1.49}$$

Along with this the extrinsic curvature $\mathcal{K}_{ij}$ is modified to be trace free, $\mathcal{A}_{ij}$.

$$\tilde{A}_{ij} = \chi\left[\mathcal{K}_{ij} - \frac{1}{3}\mathcal{K}\gamma_{ij}\right] \quad \rightarrow \quad \tilde{A}_{ij}\gamma^{ij} = 0 \tag{0.1.50}$$

During an evolution the conditions $\tilde{A}_{ij} = 0$ and $\tilde{\gamma} = 1$ are enforced which are observed to improve numerical stability; however it is unclear why beyond heuristic arguments. The definition of $\chi = \gamma^{-1/3}$ is good for black hole simulations where $\gamma \rightarrow \infty$ but $\chi \rightarrow 0$; for example the isotropic schwarzschild metric has

$$\gamma = \left[1 + \frac{M}{2r}\right]^{12} \quad \rightarrow \quad \chi = \left[\frac{r}{\frac{M}{2} + r}\right]^4. \tag{0.1.51}$$

The next step is to introduce the conformal connection functions as auxiliary variables.

$$\tilde{\Upsilon}^i = \tilde{\gamma}^{jk}\tilde{\Upsilon}^i{}_{jk} = -\partial_i\tilde{\gamma}^{ij} \tag{0.1.52}$$

$$\tilde{\Upsilon}^i{}_{jk} = \frac{1}{2}\tilde{\gamma}^{il}\left[\partial_j\tilde{\gamma}_{kl} + \partial_k\tilde{\gamma}_{lj} - \partial_l\tilde{\gamma}_{jk}\right] = \Upsilon^i{}_{jk} + \left[\delta^i_j\partial_k + \delta^i_k\partial_j - \gamma^{il}\gamma_{jk}\partial_l\right]\ln\sqrt{\chi} \tag{0.1.53}$$

Promoting these to evolution variables changes the hyperbolicity of the characteristic matrix and make the system well posed. This reduces the set of vacuum evolution variables to $\{\chi, \tilde{\gamma}_{ij}, \mathcal{K}, \tilde{\mathcal{A}}_{ij}, \tilde{\Upsilon}^i\}$. It is conventional to use $-\partial_i\tilde{\gamma}^{ij}$ to evaluate the conformal connection coefficients when they appear in the RHS of an equation, but $\partial_j\tilde{\Upsilon}^i$ is calculated by differentiating the evolution variable $\tilde{\Upsilon}^i$. One final detail, not included in the CCZ4 formulation discussed later, is to add multiples of the constraints () to the evolution equations to change the characteristic matrix and improve stability.

0.1.8 Z4 Formalism

The Z4 formalism [?] generalises the Einstein equation () to include an unphysical field Z_μ, along with damping terms parameterised by κ_1, κ_2.

$$R_{\mu\nu} + \nabla_\mu Z_\nu + \nabla_\nu Z_\mu - \kappa_1 \left[n_\mu Z_\nu + n_\nu Z_\mu - [1+\kappa_2] g_{\mu\nu} n^\alpha Z_\alpha\right] = 8\pi G \left[T_{\mu\nu} - \frac{1}{2} T g_{\mu\nu}\right] \tag{0.1.54}$$

Of course regular General Relativity is returned setting $Z_\mu = 0$. It is shown in [] that achieving $Z_\mu = 0$ whilst dynamically evolving Z_μ is equivalent to solving the constraints. Z_μ is subjected to a wave equation, transporting constraint violation off the computational domain. It can be shown that the system is driven to $Z_\mu = 0$ for $k_1 > 0$ and $k_2 < -1$. It is much cheaper to evolve the variables Z_μ, driven to zero, than do perform four elliptic solves for the constraints $\{\mathcal{H}, \mathcal{M}^i\}$ on each timestep.

0.1.9 CCZ4

Joining BSSN with Z4 gives the CCZ4 formalism. The additional modifications

$$\Theta = -n \cdot Z = -\alpha Z^0 \tag{0.1.55}$$

$$\hat{\Upsilon}^i = \tilde{\Upsilon}^i + \frac{2\gamma^{ij} Z_j}{\chi} \tag{0.1.56}$$

are made, leaving us with the following set of vacuum evolution variables $\{\chi, \tilde{\gamma}_{ij}, \mathcal{K}, \tilde{\mathcal{A}}_{ij}, \hat{\Upsilon}^i, \Theta\}$. Notably the pair of variables $\mathcal{R}_{ij} + \mathcal{D}_{(i} Z_{j)}$, and traced version $\mathcal{R} + \mathcal{D} \cdot Z$, always appear together; separately they ruin strong hyperbolically but together they do not. The Evolution equations can now be found in the CCZ4 scheme using the following equations (list evolution equations and decomposition of Υ). Note the last two equations () are for the 3+1 Klein-Gordon equation which will be derived later in ().

$$\partial_t \chi = \beta^k \partial_k \chi + \frac{2\chi}{3} \left[\alpha \mathcal{K} - \partial_k \beta^k\right] \tag{0.1.57}$$

$$\partial_t \tilde{\gamma}_{ij} = \beta^k \partial_k \tilde{\gamma}_{ij} + \tilde{\gamma}_{kj} \partial_i \beta^k + \tilde{\gamma}_{ik} \partial_j \beta^k - \frac{2}{3} \tilde{\gamma}_{ij} \partial_k \beta^k - 2\alpha \tilde{\mathcal{A}}_{ij} \tag{0.1.58}$$

$$\begin{aligned}\partial_t \mathcal{K} &= \beta^k \partial_k \mathcal{K} + \alpha \left[\mathcal{R} + 2\mathcal{D} \cdot Z + \mathcal{K}\left[\mathcal{K} - 2\Theta\right]\right] - 3\alpha\kappa_1 \left[1 + \kappa_2\right] \Theta \\ &\quad - \chi \tilde{\gamma}^{kl} \mathcal{D}_k \mathcal{D}_l \alpha + 4\pi G \alpha \left[\mathcal{S} - 3\mathcal{E}\right]\end{aligned} \tag{0.1.59}$$

$$\begin{aligned}\partial_t \tilde{\mathcal{A}}_{ij} &= \beta^k \partial_k \tilde{\mathcal{A}}_{ij} + \chi \left[\alpha \left[\mathcal{R}_{ij} + 2\mathcal{D}_{(i} Z_{j)} - 8\pi G \mathcal{S}_{ij}\right] - \mathcal{D}_i \mathcal{D}_j \alpha\right]^{TF} \\ &\quad + \tilde{\mathcal{A}}_{ij} \left[\alpha\left[\mathcal{K} - 2\Theta\right] - \frac{2}{3}\mathcal{K}^2\right] + 2\tilde{\mathcal{A}}_{k(i} \partial_{j)} \beta^k - 2\alpha \tilde{\gamma}^{kl} \tilde{\mathcal{A}}_{ik} \tilde{\mathcal{A}}_{lj}\end{aligned} \tag{0.1.60}$$

$$\partial_t \Theta = \beta^k \partial_k \Theta + \frac{1}{2}\alpha \left[\mathcal{R} + 2\mathcal{D} \cdot Z - \tilde{\mathcal{A}}_{kl} \tilde{\mathcal{A}}^{kl} + \frac{2}{3}\mathcal{K}^2 - 2\Theta\mathcal{K}\right] - \kappa_1 \alpha \Theta \left[2 + \kappa_2\right] - Z^k \partial_k \alpha - 8\pi G \alpha \mathcal{E} \tag{0.1.61}$$

$$\begin{aligned}\partial_t \hat{\Upsilon}^i &= \beta^k \partial_k \hat{\Upsilon}^i + \frac{2}{3} \left[\partial_k \beta^k \left[\tilde{\Upsilon}^i + 2\kappa_3 \frac{Z^i}{\chi}\right] - 2\alpha \mathcal{K} \frac{Z^j}{\chi}\right] - 2\alpha\kappa_1 \frac{Z^i}{\chi} \\ &\quad + 2\tilde{\gamma}^{ij} \left[\alpha \partial_j \Theta - \Theta \partial_j \alpha\right] - 2\tilde{\mathcal{A}}^{ij} \partial_j \alpha - \alpha \left[\frac{4}{3} \tilde{\gamma}^{ij} \partial_j \mathcal{K} + 3\tilde{\mathcal{A}}^{ij} \frac{\partial_j \chi}{\chi}\right] \\ &\quad - \left[\tilde{\Upsilon}^j + 2\kappa_3 \frac{Z^j}{\chi}\right] \partial_j \beta^i + 2\alpha \tilde{\Upsilon}^i{}_{jk} \tilde{\mathcal{A}}^{jk} + \tilde{\gamma}^{jk} \partial_j \partial_k \beta^i + \frac{1}{3} \tilde{\gamma}^{ij} \partial_k \partial_j \beta^k - 16\pi G \alpha \tilde{\gamma}^{ij} \mathcal{S}_j\end{aligned} \tag{0.1.62}$$

$$\partial_t \varphi = \beta^k \partial_k \varphi - \alpha \Pi \tag{0.1.63}$$

$$\partial_t \Pi = \beta^k \partial_k \Pi - \chi \tilde{\gamma}^{ij} \partial_i \varphi \partial_j \alpha + \alpha \left[\chi \tilde{\Upsilon}^k \partial_k \varphi + \frac{1}{2} \tilde{\gamma}^{lk} \partial_k \chi \partial_l \varphi - \chi \tilde{\gamma}^{ij} \partial_i \partial_j \varphi + \mathcal{K}\Pi + V'\varphi\right] \tag{0.1.64}$$

Also used was the lie derivative for a tensor density $\mathcal{T}$ of weight n.

$$\mathcal{L}_X \mathcal{T} = \tilde{\mathcal{L}}_X \mathcal{T} + n\mathcal{T} \partial_k X^k \tag{0.1.65}$$

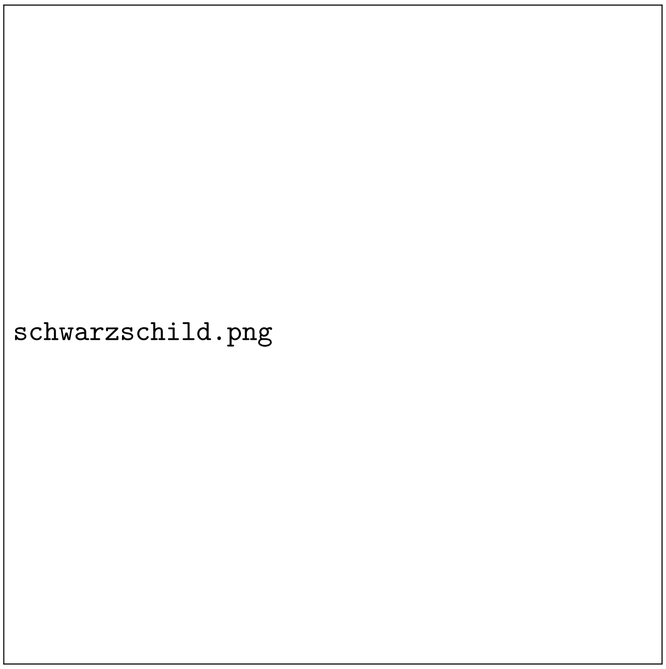
Here $\widetilde{\mathcal{L}}_X$ represents the differential operator that is equivalent to $\mathcal{L}_X$ if it didn't act on a tensor density. Also to be noted, in the CCZ4 equations there is an additional parameter κ_3 premultiplying terms in the evolution of $\hat{\Upsilon}^i$ which experimentally were found to ruin numerical stability for black hole simulations []. Setting $\kappa_3 < 1$ stabilises the simulation but at the cost of covariance. Later on it was realised that setting $\kappa_3 = 1$ and $\alpha\kappa_1 \to \kappa_1$ retains covariance as well as numerical stability [].

The CCZ4 scheme proves useful in my simulations for a few reason. Firstly, any initial data that does not satisfy the constraints will not do so along evolution either when using BSSN. Given that superposition of solutions in GR does not generally give a new solution, but does approximate one for separated compact objects, all the simulated binaries considered in the report will have non constraint satisfying initial data. The boosted stars I use later are an exact solution of GR though, being a coordinate transformation from the exact resting soliton. The use of CCZ4 will help simulations satisfy the constraints even if they initially satisfy them; one reason being that finite resolution imposes some small deviation from the continuum solution. More importantly, the use of adaptive mesh refinement AMR introduces large interpolation errors into the simulation at the boundary of the different grid resolution levels. Finally, the Sommerfeld boundary conditions used are inexact in GR and will introduce small errors at the boundary that ruin constraint satisfaction. In all above cases, the CCZ4 system forces the evolution towards constraint satisfaction, despite the numerical errors and approximations.

0.1.10 Gauge Conditions

In GR, the lapse α and shift β^i are freely specifiable being gauge variables, however they must be chosen carefully along with a suitable initial Cauchy surface Σ_{t_0} for sensible foliations and numerical accuracy. Figure 1 (left) shows how a poor initial Cauchy surface could extend to the singularity for ingoing Eddington-Finkelstein coordinates; likely causing a simulation crash. In my work Σ_{t_0} is chosen to be isotropic as in Figure 1 (right); not only does this allow trivial swapping between spherical polar and Cartesian (used in simulation) coordinates but also provides an initial Cauchy surface that's free of singularities and easy to compute. For a poor choice of lapse function, even a well chosen Σ_{t_0} can advance to the singularity in finite simulation time.

Figure 1: Penrose Diagrams, Σ_{t_0} dashed, Left: Ingoing Eddington-Finkelstein Coordinates, Right: Isotropic Coordinates.



The simplest lapse choice would be to enforce $\alpha = 1$, which is geodesic slicing, with the hypersurface following geodesics; given that geodesics can converge this can lead to hypersurface self-intersection which breaks the definition of a Cauchy surface and the simulation will likely fail. Another problem is that a black hole singularity can be reached in finite simulation time. This can be modified to the maximal slicing condition which keeps the volume element $\sqrt{-g}$ constant along geodesics. This means as $\gamma \to \infty$ nearing a singularity $\alpha \to 0$ using equation (), causing the hypersurface to advance more slowly before a singularity is reached. This property is called singularity avoiding and is crucial for numerical stability (unless using excision?). Maximal slicing can be implemented by forcing $\mathcal{K} = \partial_t \mathcal{K} = 0 \; \forall t$ which requires a slow elliptic solve for α at each timestep. Instead α is promoted to an evolution variable and is evolved along with every other simulation variable. To do this we can pick an algebraic slicing condition of the following type.

$$\mathcal{L}_m \alpha = -\alpha^2 f(\alpha) \mathcal{K} \tag{0.1.66}$$

This is used with $f = 2\alpha^{-1}$ in GRChombo giving

$$\mathcal{L}_m \alpha = -2\alpha \mathcal{K} \tag{0.1.67}$$

and using normal coordinates $\beta^i = 0$ it reduces to

$$\alpha = 1 + \ln \gamma \tag{0.1.68}$$

which is called 1+log slicing; this is very common in Numerical Relativity codes. In practice 1+log slicing is strongly singularity avoiding reaching $\alpha = 0$ before the singularity. This is modified in the CCZ4 scheme to

$$\partial_t \alpha = -2\alpha \left[\mathcal{K} - 2\Theta\right] + \beta^i \partial_i \alpha \tag{0.1.69}$$

which is very similar to before; it should be noted that the advection term is important for black hole simulations.

So far the shift vector β^i has been ignored, the simplest choice for it would be $\beta^i = 0$. Using 1+log slicing causes great stretching and shearing of Σ_t in the neghbourhood of a singularity and choosing $\beta^i = 0$ causes this shear to mean neighbouring gridpoints have large differences in field values leading to inaccurate and unstable evolutions. Another negative side effect is that the computational domain can fall inside an event horizon in black hole simulations. To counteract this we want to pick a shift vector that minimises hypersurface shear σ_{ij} which can be defined as [ref 1977 smarr and york]

$$\sigma_{ij} := \perp_i^\mu \perp_j^\nu \left[\nabla_{(\mu} n_{\nu)} - \frac{1}{3}\gamma^{ab}\nabla_{(a} n_{b)} \gamma_{\mu\nu}\right] \tag{0.1.70}$$

where σ_{ij} is tracefree corresponding to shearing rather than inflation or expansion. Minimising total shear, Σ,

$$\Sigma = \int \sigma_{ij}\sigma^{ij}\sqrt{\gamma}\, \mathrm{d}x^3 \tag{0.1.71}$$

with respect to β^i leads to an elliptic PDE to be solved for each β^i at each time step that minimises shear.

$$\delta\Sigma = 0 \to \mathcal{D}_i \sigma^{ij} = 0 \tag{0.1.72}$$

This is known as the minimal shift condition. As before, promoting β^i to be evolution variables is computationally cheaper. A very common choice is to promote the elliptic PDE for β^i into a hyperbolic equation via introducing a $\partial_t^2 \beta^i$ term and an artificial damping term parameterised by η. This becomes a damped wave equation and is supposed to transport away any part of β^i which does not satisfy $\mathcal{D}_i \sigma^{ij} = 0$. This requires Sommerfeld (outgoing wave) boundary conditions. In GRChombo the standard gamma driver shift condition is used.

$$\partial_t \beta^i = F B^i \tag{0.1.73}$$

$$\partial_t B^i = \partial_t \tilde{\Gamma}^i - \eta B^i \tag{0.1.74}$$

Here $F = 3/4$ and $\eta = 1$ are used in GRChombo.