Introduction to numerical relativity

Luca Baiotti

梅乙亭　瑠嘉

Graduate School of Science

Osaka University

Plan of the lectures

- My brief overview of general relativity

- Numerical relativity in vacuum and non-vacuum spacetimes

- Hands on codes

My brief overview of general relativity

- Einstein started to think of **the path of an object as a property of spacetime itself**, rather than being related with the specific properties of the object.

- The idea is that gravity is a manifestation of the fact that objects in free fall follow **geodesics**, in curved spacetimes.

- We know in our ordinary experience (flat spacetime) that in the absence of any forces, objects follow straight lines, and we also know that straight lines are the shortest possible paths that connect two points in such conditions.

- The generalisation of the notion of a "straight line" valid also in curved spacetimes is called **geodesic**.

My brief overview of general relativity

Important quantities in general relativity:

- the **metric** (the metric tensor g), which may be regarded as a machinery for measuring distances:

$$dS^2 = g_{\mu\nu} dx^\mu dx^\nu$$

- **curvature**, expressed by the Riemann curvature tensor

$$R^\alpha{}_{\beta\mu\nu} = \Gamma^\alpha{}_{\beta\nu,\mu} - \Gamma^\alpha{}_{\beta\mu,\nu} + \Gamma^\alpha{}_{\sigma\mu}\Gamma^\sigma{}_{\beta\nu} - \Gamma^\alpha{}_{\sigma\nu}\Gamma^\sigma{}_{\beta\mu}$$

(where $\Gamma^\alpha{}_{\beta\mu} = \frac{1}{2} g^{\alpha\sigma}\left(g_{\beta\sigma,\mu} + g_{\sigma\mu,\beta} - g_{\beta\mu,\sigma}\right)$ are the Christoffel symbols)

- the **Ricci tensor** $R_{\mu\nu} = R^\alpha{}_{\mu\alpha\nu}$ and the **curvature (Ricci) scalar** $R = g^{\mu\nu} R_{\mu\nu}$

- the **Einstein tensor**: $$G_{\mu\nu} = R_{\mu\nu} - \frac{1}{2} g_{\mu\nu} R$$

My brief overview of general relativity

John Archibald Wheeler’s famous summary of general relativity:

"Geometry tells matter how to move”

"Matter tells spacetime how to curve"

My brief overview of general relativity

"Geometry tells matter how to move”, namely the geodetic equation:

$$\ddot{x}^{\mu} + \Gamma^{\mu}_{\alpha\beta}\dot{x}^{\alpha}\dot{x}^{\beta} = 0$$

If we know the geometry of spacetime we can compute the paths of matter particles

My brief overview of general relativity

- **"Matter tells spacetime how to curve"**
- The distribution of matter (mass/energy/momentum) determines the spacetime geometry
- Matter distribution is covariantly described through the stress-energy tensor T, which is symmetric and whose (covariant) divergence vanishes (more later)
- Spacetime geometry is described through the metric g and its derivatives up to second order
- Einstein reached a satisfactory form for the equations relating geometry and matter:

Einstein_tensor = constant x T

- The Einstein tensor G is a tensor in **4D** spacetime that has the wanted properties of:
  - containing the metric up to second derivatives
  - being a symmetric tensor (it must because the stress-energy tensor is symmetric)
  - having vanishing (covariant) divergence (it must because the stress-energy tensor has vanishing divergence)
  - the weak-field limit of the Einstein equations gives the Newtonian Poisson equation (from the comparison to which the value of the above constant is found)

The fundamental equations

The Einstein equations:

Einstein tensor (spacetime)

Ricci tensor

Curvature scalar

Matter and other fields

$$G_{\mu\nu} \equiv R_{\mu\nu} - \frac{1}{2} g_{\mu\nu} R = 8\pi T_{\mu\nu}$$

Metric

(measure of spacetime distances)

4-velocity

$$T^{\mu\nu} = \rho(1 + \varepsilon + p/\rho)u^{\mu}u^{\nu} + pg^{\mu\nu}$$

Rest-mass density Internal energy density Pressure

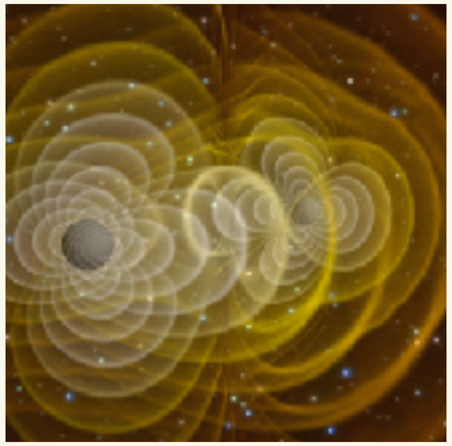
i.e.

- □ six, second-order-in-time, second-order-in-space, coupled, highly-nonlinear, quasi-hyperbolic, partial differential equations (PDEs)
- □ four, second-order-in-space, coupled, highly-nonlinear, elliptic PDEs

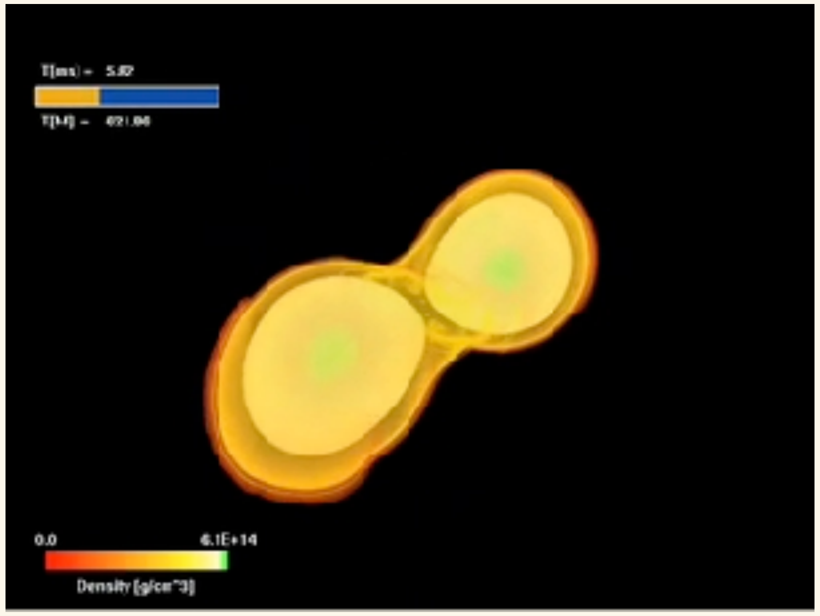
Numerical relativity

Some gravitational-wave sources

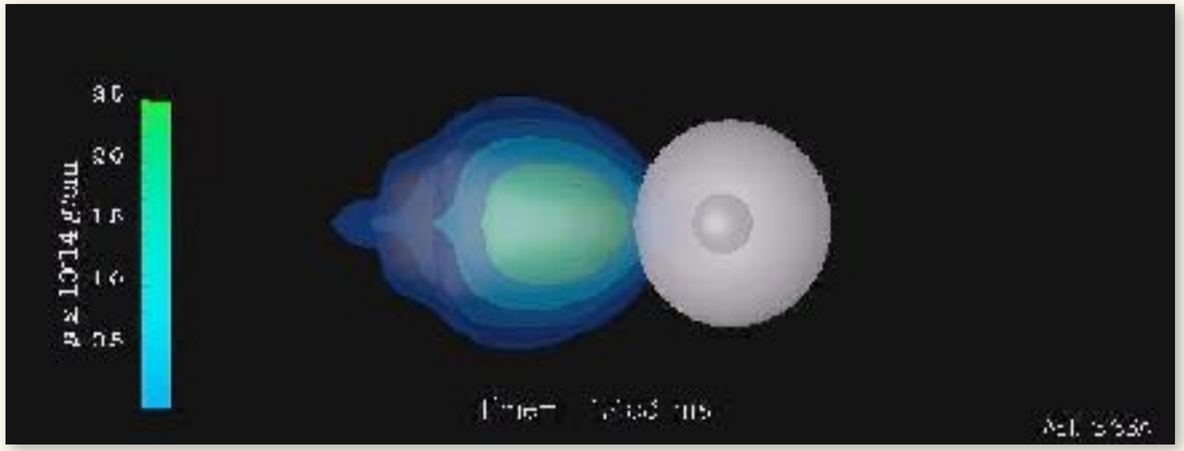
- Binary black holes



- Binary neutron stars

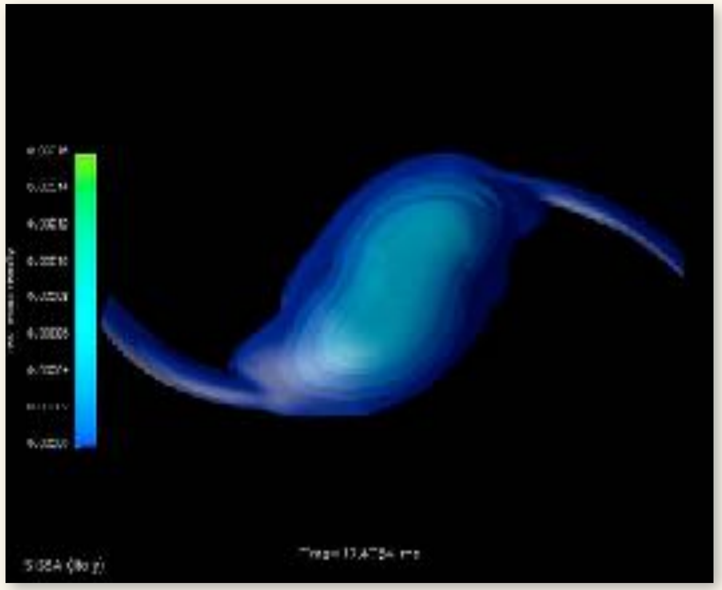


- Mixed binary systems



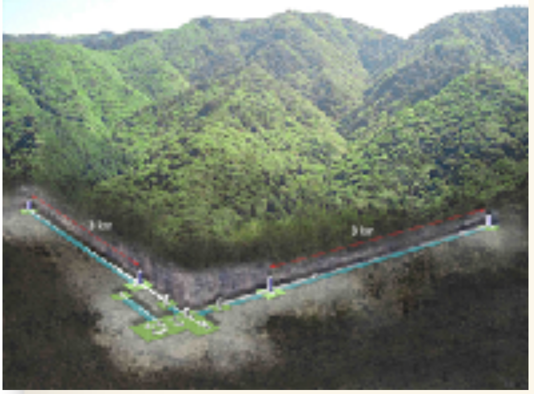
- Gravitational collapse (supernovae, neutron stars)

- Deformed compact stars



Gravitational-wave templates for detectors

Kagra - Kamioka



GEO-Hannover

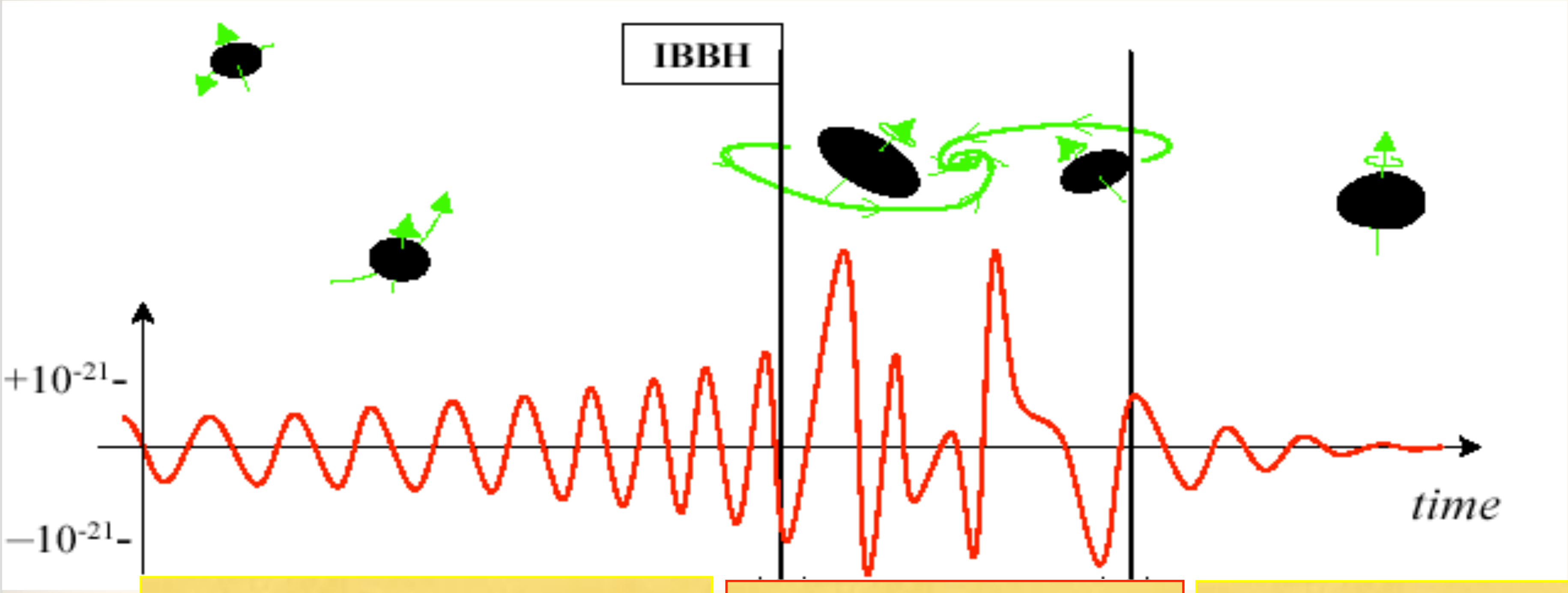
LIGO-Livingston

LIGO-Hanford

VIRGO-Cascina

indigo

Detectors measure $\Delta L/L \sim h < 10^{-21}$ with S/N~1.



post-Newtonian expansions

Numerical relativity

Perturbative analysis

Knowledge of the waveforms can compensate for the very small S/N (matched-filtering) and so enhance detection and improve source-characterisation.

Solving the Einstein equations on computers

General relativity states that our World is a ***4D*** *curved spacetime and the Einstein equations describe its dynamics.*

How to solve the Einstein equations numerically?

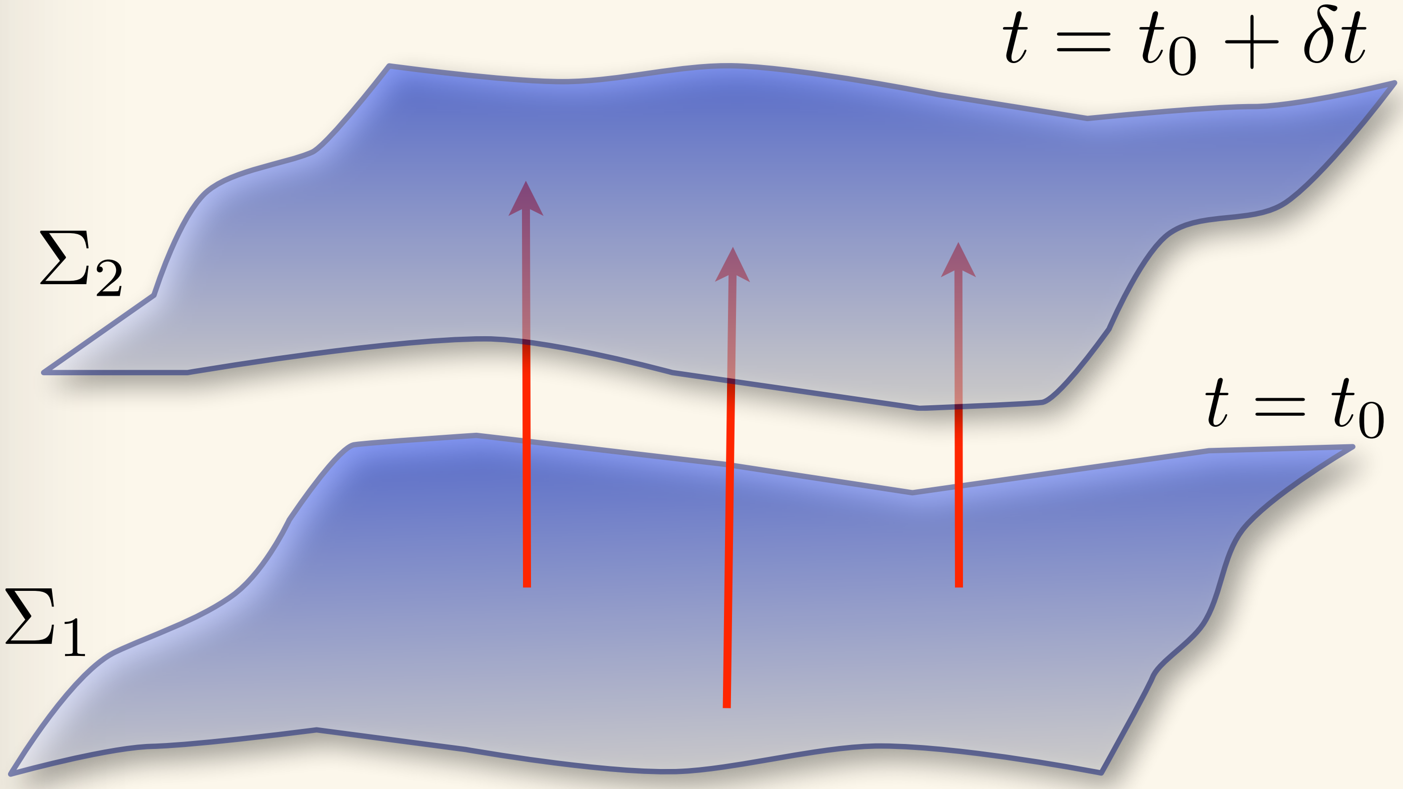
Prominently, there is no a priori concept of "flowing of time" (we are not involved in thermodynamics here), time is just one of the dimensions, and on the same level as space dimensions...

There is a successful recipe, though.

First step: foliate the 4D spacetime

We have the illusion to live in ***3D*** *and it is easier to tell computers to perform simulations (time-)step by (time-)step.*

So, given a manifold $\mathcal{M}$ describing a spacetime with 4-metric $g_{\mu\nu}$ we want to foliate it via space-like, three-dimensional hypersurfaces: $\Sigma_1, \Sigma_2, \ldots$. We label such hypersurfaces with the time coordinate t.



Define therefore:

$$\Omega_\mu \equiv \nabla_\mu t$$

(*"the direction of time"*)

Also assign a normalization such that

$$|\Omega|^2 \equiv g^{\mu\nu}\nabla_\mu t \nabla_\nu t = -\alpha^{-2}$$

(As mostly done in numerical relativity, the signature is here -+++)

The function α is called the "lapse" function and it is strictly positive for spacelike hypersurfaces: $\alpha(t, x^i) > 0$

Let's also define:

i) the unit normal vector to the hypersurface Σ

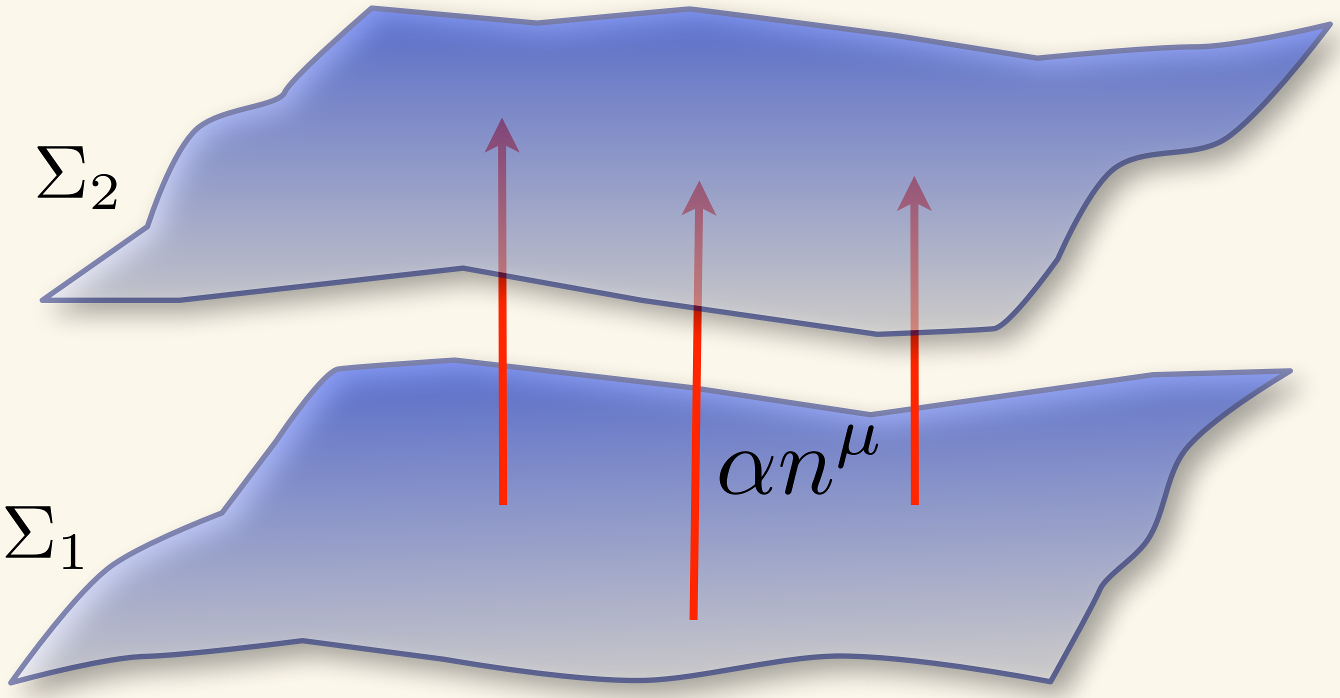
$$n^\mu \equiv -\alpha g^{\mu\nu}\Omega_\nu = -\alpha g^{\mu\nu}\nabla_\nu t$$

so that

$$n^\mu n_\mu = -1$$

ii) and the spatial metric

$$\gamma_{\mu\nu} \equiv g_{\mu\nu} + n_\mu n_\nu$$



Second step: decompose 4D tensors

By using $\boldsymbol{n}$ and $\boldsymbol{\gamma}$ we can decompose any 4D tensor into a purely spatial part (hence in Σ) and a purely timelike part (hence orthogonal to Σ and aligned with $\boldsymbol{n}$).

The spatial part is obtained by contracting with the spatial projection operator, defined as

$$\gamma^{\mu}{}_{\nu} = g^{\mu\alpha}\gamma_{\alpha\nu} = g^{\mu}{}_{\nu} + n^{\mu}n_{\nu} = \delta^{\mu}{}_{\nu} + n^{\mu}n_{\nu}$$

while the timelike part is obtained by contracting with the timelike projection operator:

$$N^{\mu}{}_{\nu} = -n^{\mu}n_{\nu}$$

The two projectors are obviously orthogonal:

$$\gamma^{\nu}{}_{\mu}N^{\mu}{}_{\nu} = 0$$

The 3D covariant derivative of a spatial tensor is then defined as the projection on Σ of all the indices of the the 4D covariant derivative:

$$D_\alpha T^\beta{}_\delta = \gamma^\rho{}_\alpha \gamma^\beta{}_\sigma \gamma^\tau{}_\delta \nabla_\rho T^\sigma{}_\tau$$

All the 4D tensors in the Einstein equations can be projected straightforwardly onto the 3D spatial slice.
In particular, the 3D Christoffel symbols:

$$\Gamma^\alpha{}_{\beta\delta} = \frac{1}{2}\gamma^{\alpha\mu}(\gamma_{\mu\beta,\delta} + \gamma_{\mu\delta,\beta} - \gamma_{\beta\delta\,\mu})$$

the 3D Riemann tensor:

$$R^\alpha{}_{\beta\gamma\delta} = \Gamma^\alpha{}_{\beta\delta,\gamma} - \Gamma^\alpha{}_{\beta\gamma,\delta} + \Gamma^\mu{}_{\beta\delta}\Gamma^\alpha{}_{\mu\gamma} - \Gamma^\mu{}_{\beta\gamma}\Gamma^\alpha{}_{\mu\delta}$$

and the 3D contractions of the 3D Riemann tensor, i.e. the 3D Ricci tensor the 3D Ricci scalar: $R_{\alpha\beta} = R^\delta{}_{\alpha\delta\beta}$ and $R = R^\delta_\delta$.

The information present in the 4D Riemann tensor and absent in the 3D Riemann tensor can be found in another spatial tensor: the extrinsic curvature, which describes how the 3D hypersurface is embedded ("bent") in the 4D manifold.

The extrinsic curvature is defined in terms of the unit normal to Σ as:

$$K_{\alpha\beta} = -\gamma_\alpha{}^\mu \gamma_\beta{}^\nu \nabla_{(\mu} n_{\nu)} \quad \text{where} \quad 2T_{(\alpha\beta)} = T_{\alpha\beta} + T_{\beta\alpha}$$

and this can be shown to be equivalent also to:

$$K_{\alpha\beta} = -\frac{1}{2} \mathcal{L}_{\boldsymbol{n}} \gamma_{\alpha\beta}$$

where $\mathcal{L}_{\boldsymbol{n}}$ is the Lie derivative along $\boldsymbol{n}$.

This also expresses the fact that the extrinsic curvature can be seen as the rate of change of the spatial metric.

Properties of the Lie derivative

Recall that the Lie derivative can be thought of as a geometrical generalization of a directional derivative. It evaluates the change of a tensor field along the flow of a vector field.

For a scalar function ϕ this is given by:

$$\mathcal{L}_{\boldsymbol{X}}\phi = X^\mu D_\mu \phi = x^\mu \partial_\mu \phi$$

For a vector field V^μ, this is given by the commutator:

$$\begin{aligned}\mathcal{L}_{\boldsymbol{X}} V^\nu &= X^\mu D_\mu V^\nu - V^\mu D_\mu X^\nu = [X, V]^\nu \\ &= X^\mu \partial_\mu V^\nu - V^\mu \partial_\mu X^\nu\end{aligned}$$

For a 1-form ω_μ ,this is given by:

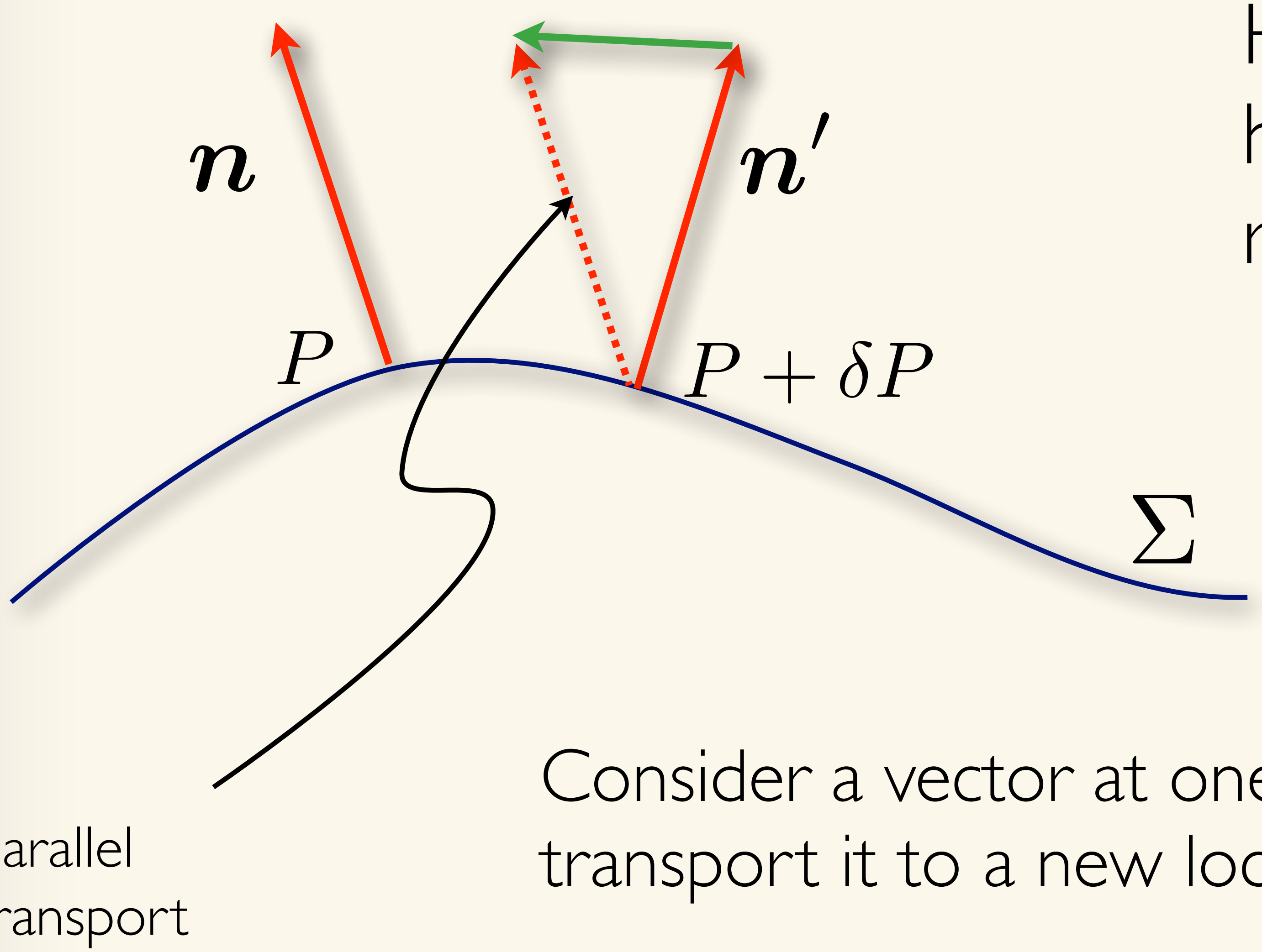
$$\mathcal{L}_{\boldsymbol{X}} \omega_\nu = X^\mu D_\mu \omega_\nu + \omega_\nu D_\mu X^\nu = X^\mu \partial_\mu \omega_\nu + \omega_\mu \partial_\mu X^\nu$$

As a result, for a generic tensor of rank $\binom{1}{1}$ this is given by:

$$\begin{aligned}\mathcal{L}_{\boldsymbol{X}} T^\mu{}_\nu &= X^\alpha D_\alpha T^\mu{}_\nu - T^\alpha{}_\nu D_\alpha X^\mu + T^\mu{}_\alpha D_\nu X^\alpha \\ &= X^\alpha \partial_\alpha T^\mu{}_\nu - T^\alpha{}_\nu \partial_\alpha X^\mu + T^\mu{}_\alpha \partial_\nu X^\alpha\end{aligned}$$

The extrinsic curvature measures the gradients of the normal vectors $\boldsymbol{n}$ and, since these are normalized, they can only differ in direction. Thus the extrinsic curvature provides information on how much the normal direction changes from point to point and so on how the hypersurface is deformed.

$$\delta \boldsymbol{n} = -\boldsymbol{K} \delta P$$



Hence it measures how the 3D hypersurface is “bent” with respect to the 4D spacetime.

Consider a vector at one position P and then parallel-transport it to a new location $P + \delta P$.

The difference in the two vectors is proportional to the extrinsic curvature and this can either be positive or negative.

Third step: decompose the Einstein equations

Next, we need to decompose the Einstein equations in the spatial and timelike parts.

$$ {}^{(4)}G_{\mu\nu} \equiv {}^{(4)}R_{\mu\nu} - \frac{1}{2}{}^{(4)}R g_{\mu\nu} = 8\pi T_{\mu\nu} $$

To this purpose it is useful to derive a few identities.

Gauss equations: decompose the 4D Riemann tensor ${}^{(4)}R_{\alpha\beta\mu\nu}$ projecting all indices:

$$ R_{\alpha\beta\gamma\delta} + K_{\alpha\gamma}K_{\beta\delta} - K_{\alpha\delta}K_{\beta\gamma} = \gamma^{\mu}{}_{\alpha}\gamma^{\nu}{}_{\beta}\gamma^{\rho}{}_{\delta}\gamma^{\sigma}{}_{\gamma}\,{}^{(4)}R_{\mu\nu\rho\sigma} $$

Codazzi equations: take 3 spatial projections and a timelike one:

$$ D_{\alpha}K_{\beta\gamma} - D_{\beta}K_{\alpha\gamma} = \gamma^{\rho}{}_{\beta}\gamma^{\mu}{}_{\alpha}\gamma^{\nu}{}_{\gamma}n^{\sigma}\,{}^{(4)}R_{\rho\mu\nu\sigma} $$

Ricci equations: take 2 spatial projections and 2 timelike ones:

$$\mathcal{L}_{\boldsymbol{n}} K_{\alpha\beta} = n^{\delta} n^{\gamma} \gamma^{\mu}{}_{\alpha} \gamma^{\nu}{}_{\beta} {}^{(4)}R_{\nu\delta\mu\gamma} - \frac{1}{\alpha} D_{\alpha} D_{\beta} \alpha - K^{\gamma}{}_{\beta} K_{\alpha\gamma}$$

Another important identity which will be used in the following is

$$D_{\mu} U^{\nu} = \gamma_{\mu}{}^{\rho} \nabla_{\rho} U^{\nu} + K_{\mu\rho} U^{\rho} n^{\nu}$$

and which holds for any spatial vector $\boldsymbol{U}$.

We must also consider the projections of the stress-energy $T_{\mu\nu}$ tensor (the right-hand-side of the Einstein equations).

Let's define the double timelike projection of the stress-energy tensor as:

$$e = n^{\mu} n^{\nu} T_{\mu\nu}$$

Similarly, the momentum density (i.e. the mass current) will be given by the mixed time and spatial projection:

$$j_{\mu} = -\gamma^{\alpha}_{\mu} n^{\beta} T_{\alpha\beta}$$

And similarly for the space-space projection.

With all these formulas, we can decompose the Einstein equations in the 3+1 splitting.

We get two sets of equations:

1) the “constraint” equations, which are fully defined on each spatial hypersurfaces (and do not involve time derivatives)

2) the “evolution” equations, which instead relate quantities (the spatial metric and the extrinsic curvature) between two adjacent hypersurfaces.

The constraint equations (I)

We first time-project twice the left-hand-side of the Einstein equations to obtain

$$2n^{\mu}n^{\nu\,(4)}G_{\mu\nu} = R + K^2 - K_{\mu\nu}K^{\mu\nu}$$

Doing the same for the right-hand-side, using the Gauss equations contracted twice with the spatial metric and the definition of the energy density we finally reach the form of the equation, which is called Hamiltonian constraint equation

$$\boxed{R + K^2 - K_{\mu\nu}K^{\mu\nu} = 16\pi e}$$

Note that this is a single elliptic equation (not containing time derivatives), which should be satisfied everywhere on the spatial hypersurface Σ.

The constraint equations (II)

Similarly, with a mixed time-space projection of the left-hand-side of the Einstein equations we obtain

$$-\gamma^{\mu}{}_{\alpha} n^{\nu\,(4)}G_{\mu\nu} = -R_{\alpha\nu} n^{\nu} + \frac{1}{2} n_{\alpha} R$$

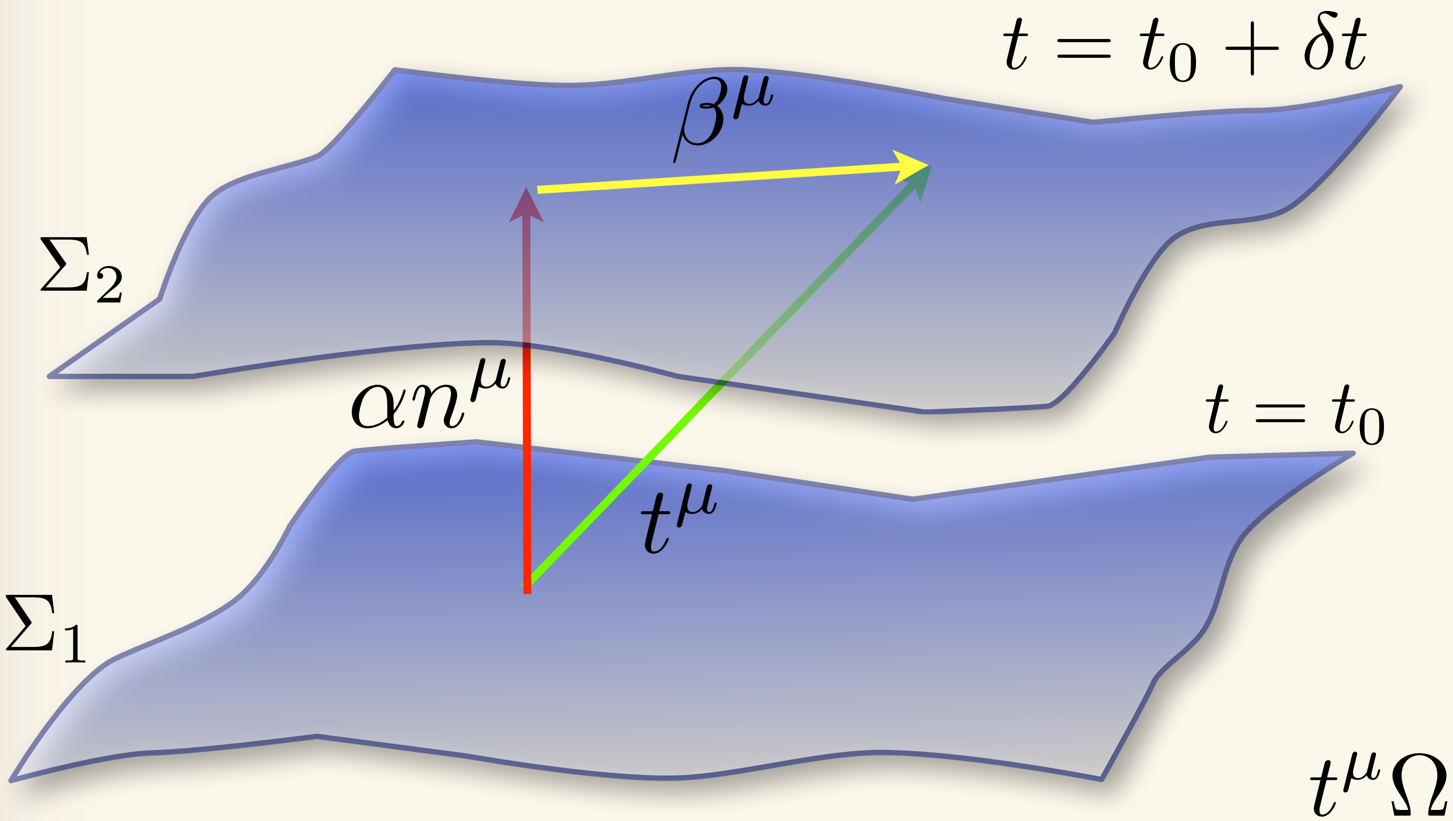
Doing the same for the right-hand-side, using the contracted Codazzi equations and the definition of the momentum density, we reach the equations called the momentum constraint equations

$$\boxed{D_{\nu} K^{\nu}{}_{\mu} - D_{\mu} K = 8\pi j_{\mu}}$$

which are also 3 elliptic equations.
The 4 constraint equations are the necessary and sufficient integrability conditions for the embedding of the spacelike hypersurfaces $(\Sigma, \gamma_{\mu\nu}, K_{\mu\nu})$ in the 4D spacetime $(\mathcal{M}, g_{\mu\nu})$.

Find the direction for evolutions

We must ensure that when going from one hypersurface $\Sigma_1$at time t to another Σ_2 at time $t+\delta t$ all the vectors originating on Σ_1 end up on Σ_2: we must *land* on a single hypersurface.



The most general of such vectors that connect two hypersurfaces is

$$t^\mu \equiv \alpha n^\mu + \beta^\mu$$

where $\boldsymbol{\beta}$ is any spatial "shift" vector. Indeed we see that

$$t^\mu \Omega_\mu = \alpha n^\mu \Omega_\mu + \beta^\mu \Omega_\mu = \alpha/\alpha = 1$$

so that the change in t along t^μ is $\delta t = t^\mu \nabla_\mu t = 1$ and so it is the same for all points, which consequently end up all on the same hypersurface.

The evolution part of the Einstein equations

With this definition we can revise the Lie derivative along the unit normal $\boldsymbol{\mathcal{L}_n}$. Since

$$\alpha\boldsymbol{\mathcal{L}}_{\boldsymbol{n}} = \boldsymbol{\mathcal{L}}_{\boldsymbol{t}} - \boldsymbol{\mathcal{L}}_{\boldsymbol{\beta}}$$

the definition of the extrinsic curvature: $K_{\alpha\beta} = -\frac{1}{2}\boldsymbol{\mathcal{L}}_{\boldsymbol{n}}\gamma_{\alpha\beta}$
can now be rewritten as

$$\boldsymbol{\mathcal{L}}_{\boldsymbol{t}}\gamma_{\mu\nu} = -2\alpha K_{\mu\nu} + \boldsymbol{\mathcal{L}}_{\boldsymbol{\beta}}\gamma_{\mu\nu}$$

The evolution part of the Einstein equations

We can now express the last piece of the 3+1 decomposition and so derive the evolution part of the Einstein equations.

As for the constraints, we need suitable projections of the two sides of the Einstein equations and in particular the two spatial ones:

$$\gamma^{\mu}{}_{\alpha}\gamma^{\nu}{}_{\beta}{}^{(4)}G_{\mu\nu} = 8\pi S_{\alpha\beta} \equiv 8\pi\gamma^{\mu}{}_{\alpha}\gamma^{\nu}{}_{\beta}T_{\mu\nu}$$

Using the Ricci equations one then obtains:

$$\begin{aligned}\mathcal{L}_{\boldsymbol{t}}K_{\mu\nu} = & -D_{\mu}D_{\nu}\alpha + \alpha(R_{\mu\nu} - 2K_{\mu\beta}K^{\beta\nu} + KK_{\mu\nu}) \\ & -8\pi\alpha(R_{\mu\nu} - \frac{1}{2}\gamma_{\mu\nu}(S-e)) + \mathcal{L}_{\boldsymbol{\beta}}K_{\mu\nu}\end{aligned}$$

where $S \equiv S^{\mu}{}_{\mu}$.

Fourth step: select a coordinate basis

So far the treatment has been coordinate independent, but in order to write computer programs we have to specify a coordinate basis. Doing so can also be useful to simplify equations and to highlight the "spatial" nature of $\boldsymbol{\gamma}$ and $\boldsymbol{K}$.

In the spirit of the 3+1 formalism, the natural choice for the coordinate unit vectors $\boldsymbol{e}_j$ is:

i) three purely spatial coordinates with unit vectors:

$$n_\mu(\boldsymbol{e}_j)^\mu = 0 \qquad e.g. \qquad (\boldsymbol{e}_1)^\mu = (0, 1, 0, 0)$$

ii) one coordinate unit vector along the vector $\boldsymbol{t}$:

$$(\boldsymbol{e}_0)^\mu = t^\mu = (1, 0, 0, 0)$$

As a result:

$$\mathcal{L}_{\boldsymbol{t}} = \partial_t$$

i.e. the Lie derivative along $\boldsymbol{t}$ is a simple partial derivative

$$n_j = n_\mu (e_j)^\mu = 0 \quad \text{but} \quad n_0 \neq 0$$

i.e. the space covariant components of a **timelike** vector are zero; only the time component is different from zero

$$n_\mu \beta^\mu = \beta^0 n_0 = 0 \quad \Longrightarrow \quad \beta^0 = 0 \quad \Longrightarrow \quad \beta^\mu = (0, \beta^j)$$

i.e. the zeroth contravariant component of a **spacelike** vector are zero; only the space components are nonzero

Putting things together and bearing in mind that $n_\mu n^\mu = -1$:

$$n^\mu = \frac{1}{\alpha}(1, -\beta^i)\,; \qquad n_\mu = (-\alpha, 0, 0, 0)$$

Recalling that the spatial components of the 4D metric are the components of the 3D metric ($g^{ij} = \gamma^{ij}$) and that $\gamma^{\alpha 0} = 0$ (true in general, for any spatial tensor), the contravariant components of the metric $g^{\mu\nu} = \gamma^{\mu\nu} - n^\mu n^\nu$ in a 3+1 split are

$$g^{\mu\nu} = \begin{pmatrix} -1/\alpha^2 & \beta^i/\alpha^2 \\ \\ \beta^i/\alpha^2 & \gamma^{ij} - \beta^i\beta^j/\alpha^2 \end{pmatrix}$$

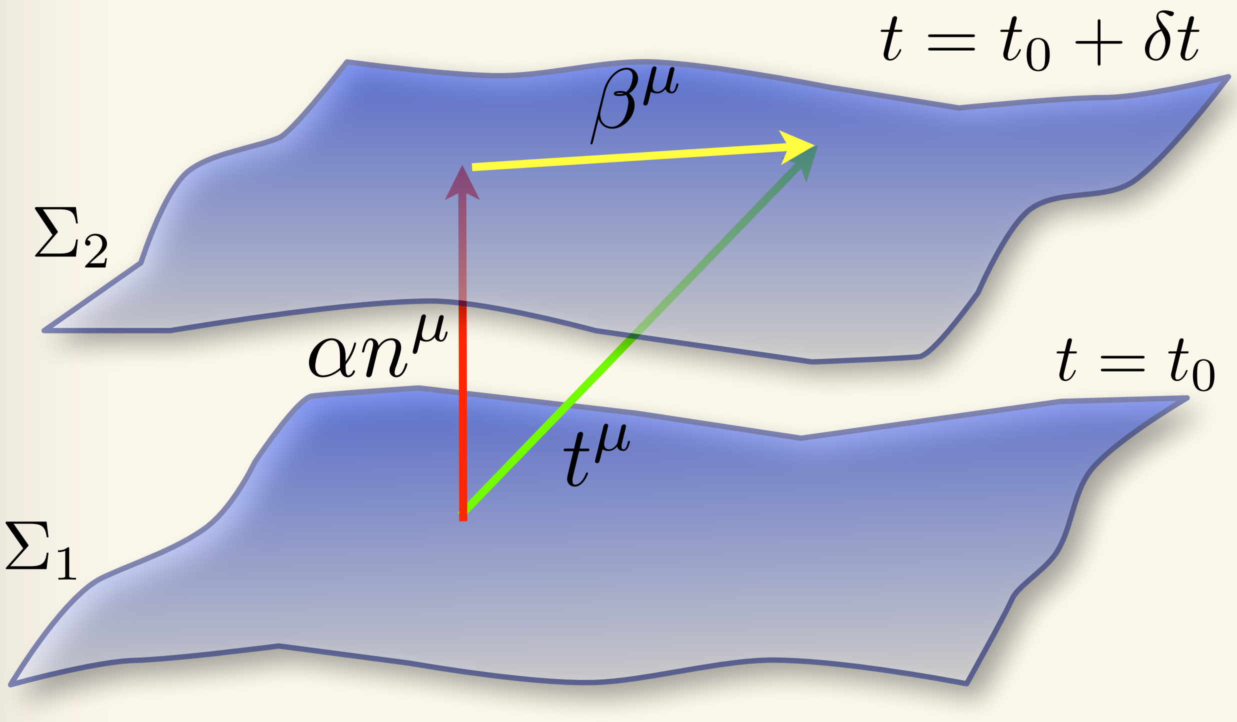
Similarly, since $g_{ij} = \gamma_{ij}$, the covariant components are

$$g_{\mu\nu} = \begin{pmatrix} -\alpha^2 + \beta_i\beta^i & \beta_i \\ \\ \beta_i & \gamma_{ij} \end{pmatrix}$$

Note that $\gamma^{ik}\gamma_{kj} = \delta^i{}_j$ (i.e. γ^{ij}, γ_{ij} are inverses) and thus they can be used to raise/lower the indices of spatial tensors.

We can now have a more intuitive interpretation of the lapse, shift and spatial metric. Using the expression for the 4D covariant metric, the line element is given by

$$\mathrm{d}s^2 = g_{\mu\nu}\mathrm{d}x^\mu \mathrm{d}x^\mu = -(\alpha^2 - \beta^i\beta_i)\mathrm{d}t^2 + 2\beta_i \mathrm{d}x^i \mathrm{d}t + \gamma_{ij}\mathrm{d}x^i \mathrm{d}x^j$$



It is now clearer that:

- the lapse measures proper time between two adjacent hypersurfaces

$$\mathrm{d}\tau^2 = -\alpha^2(t, x^j)\mathrm{d}t^2$$

- the shift relates spatial coordinates between two adjacent hypersurfaces

- the spatial metric measures distances between points on every hypersurface

$$\mathrm{d}l^2 = \gamma_{ij}\mathrm{d}x^i \mathrm{d}x^j$$

Summary of the idea of
the 3+1 or ADM (Arnowitt Deser Misner) formulation

First step: foliate the 4D spacetime in 3D spacelike hypersurfaces leveled by a scalar function: the time coordinate. This determines a normal unit vector to the hypersurfaces.

Second step: decompose 4D spacetime tensors in spatial and timelike parts using the normal vector and the spatial metric.

Third step: rewrite Einstein equations using such decomposed tensors, also selecting two functions, the lapse and the shift, that tell how to relate coordinates between two slices: the lapse measures the proper time, while the shift measures changes in the spatial coordinates.

Fourth step: select a coordinate basis and express all equations in 3+1 form.

Gauge conditions

NOTE: the lapse, and shift are not solutions of the Einstein equations but represent our "gauge freedom", namely the freedom (arbitrariness) in which we choose to foliate the spacetime.

Any prescribed choice for the lapse is usually referred to as a ''slicing condition'', while any choice for the shift is usually referred to as ''spatial gauge condition''.

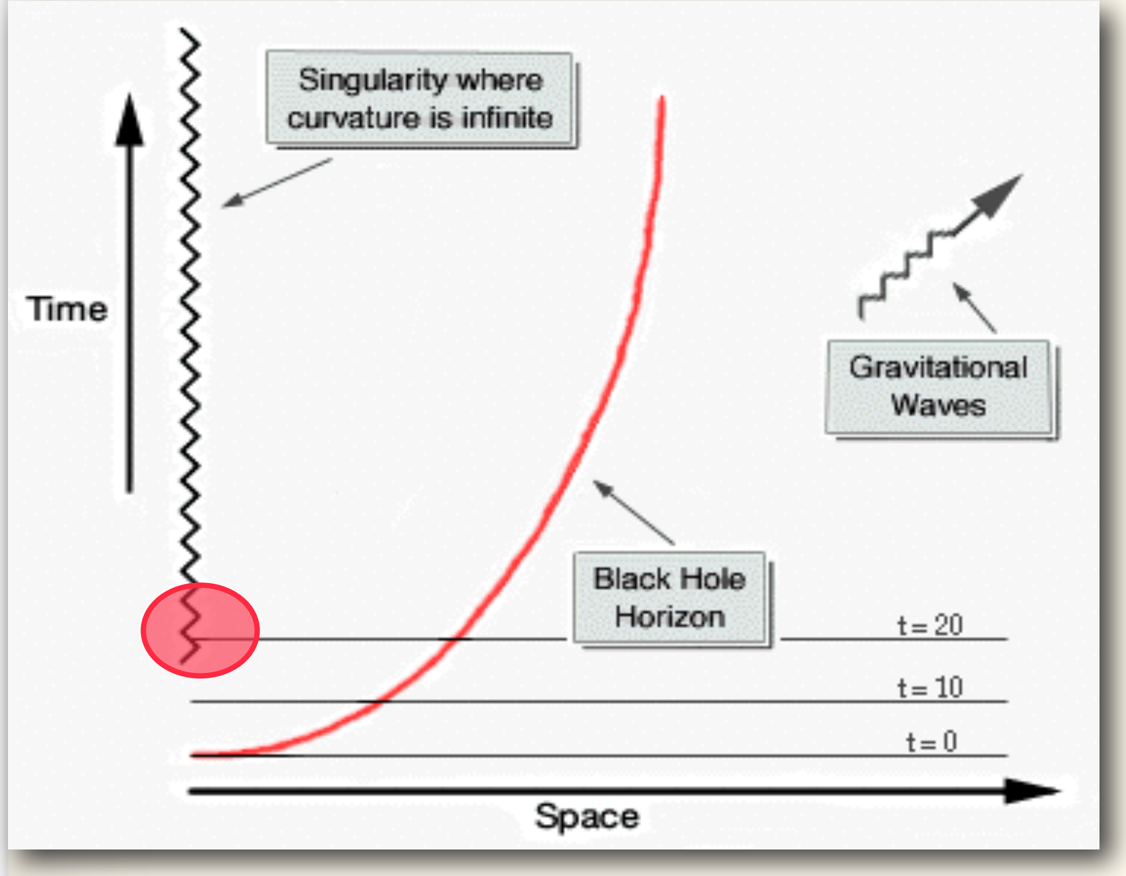
While there are infinite possible choices, not all of them are equally useful to carry out numerical simulations. Indeed, there is a whole branch of numerical relativity that is dedicated to finding suitable gauge conditions.

Different recipes for selecting lapse and shift are possible:

i) make a guess (i.e. prescribe a functional form) for the lapse, and shift: e.g. geodesic slicing $\alpha = 1\,, \;\; \beta^i = 0$

obviously not a good idea

Choosing the right temporal gauge



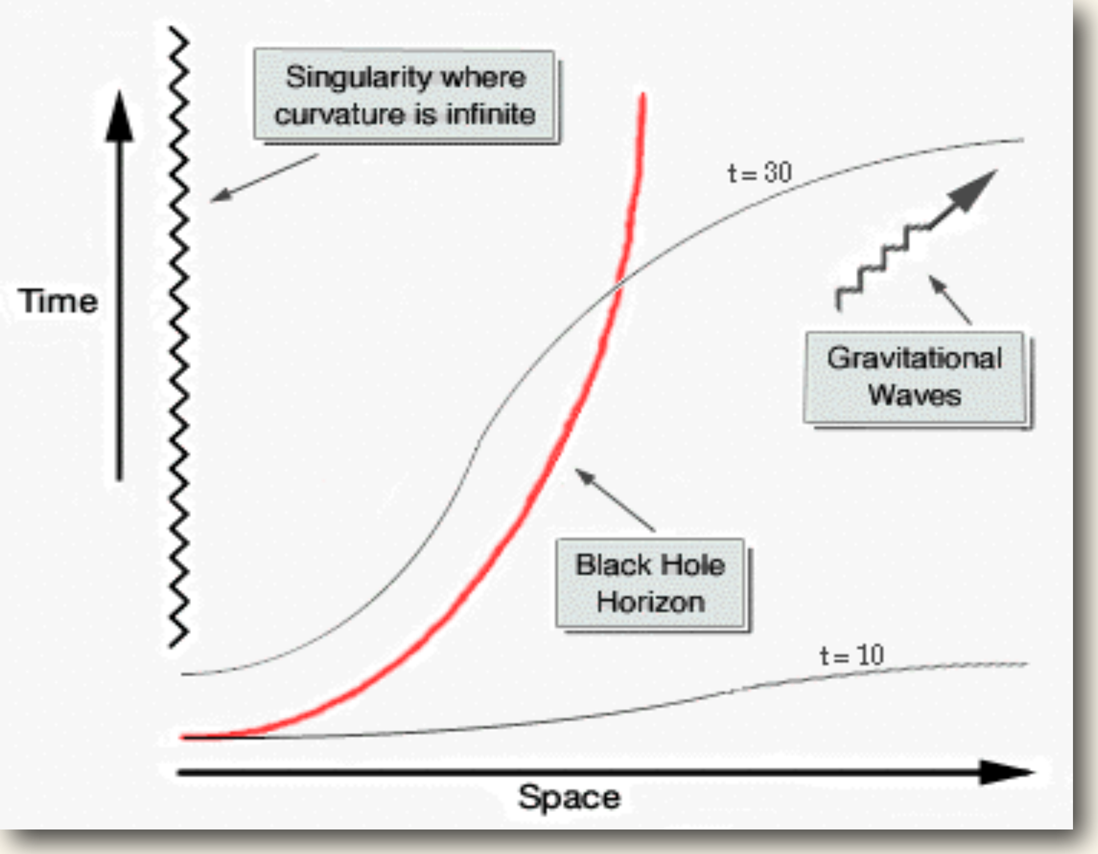
Suppose you want to follow the gravitational collapse to a black hole and assume a simplistic gauge choice (geodesic slicing):

$$\alpha = 1, \quad \beta^i = 0$$

That would lead to a code crash as soon as a singularity forms.

One needs to use smarter temporal gauges. In particular we want time to progress at different rates at different positions in the grid: *"singularity avoiding slicing" (e.g. maximal slicing)*

$$\alpha = \alpha(t, \mathbf{x}), \quad \beta^i = \beta^i(t, \mathbf{x})$$



Different recipes for selecting lapse and shift are possible:

i) make a guess (i.e. prescribe a functional form) for the lapse, and shift: e.g. geodesic slicing $\alpha = 1\,, \;\; \beta^i = 0$

obviously not a good idea

ii) fix the lapse, and shift by requiring they satisfy some condition: e.g. maximal slicing for the lapse

$$\partial_t K = 0 \quad \Longrightarrow \quad D^i D_i \alpha = \alpha [K_{ij} K^{ij} + 4\pi (e + S)]$$

which has the desired "singularity-avoiding" properties.

Good idea mathematically, but unfortunately this leads to elliptic equations which are computationally too expensive to solve at each time.

iii) determine the lapse, and shift dynamically by requiring that they satisfy comparatively simple evolution equations.

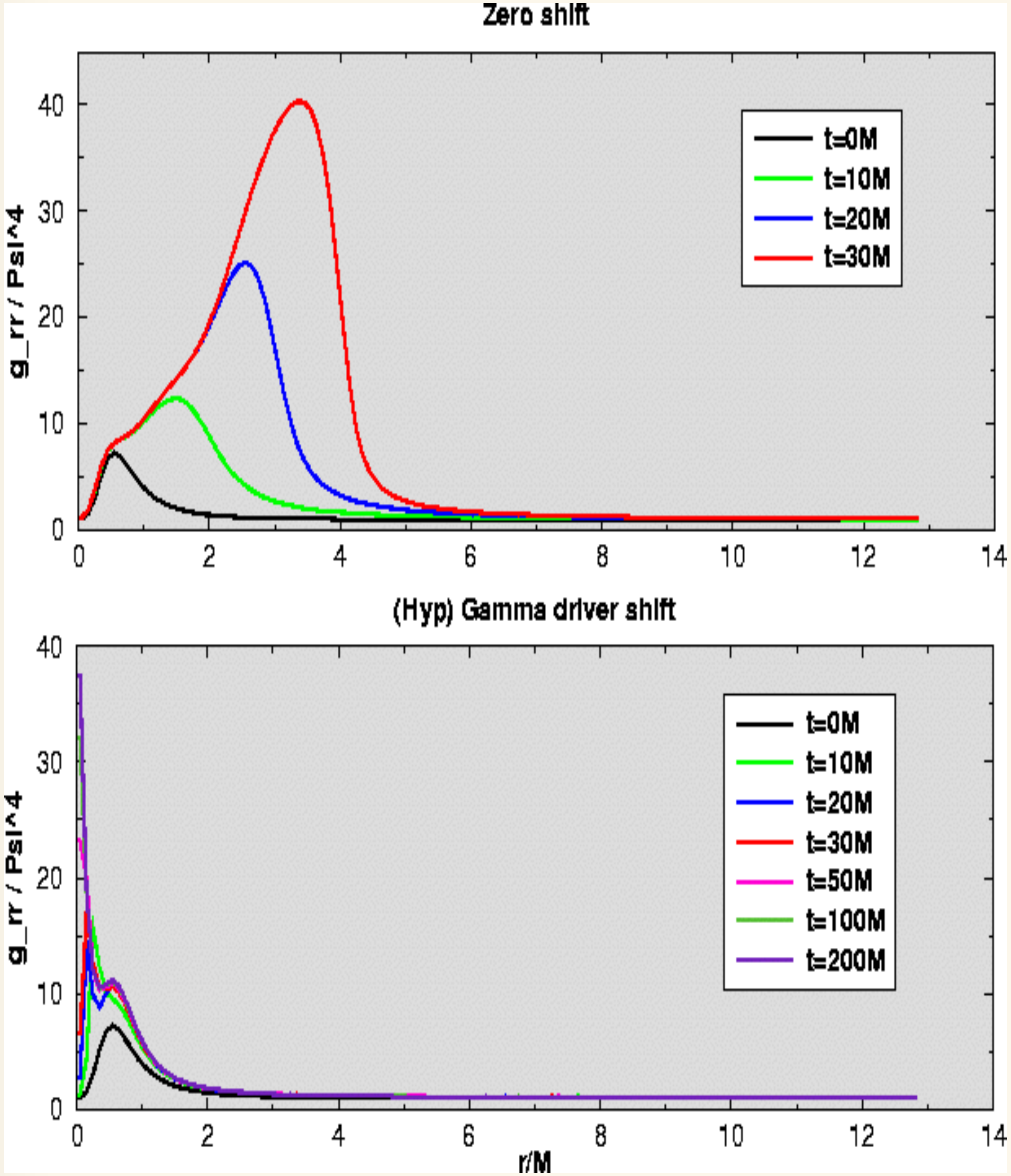
This is the common solution. The advantage is that the equations for the lapse and shift are simple time evolution equations.

A family of slicing conditions that works very well to obtain both a strongly hyperbolic evolution equations and stable numerical evolutions is the Bona-Masso slicing:

$$\partial_t \alpha - \beta^i \partial_i \alpha = -\alpha^2 f(\alpha)(K - K_0)$$

where $K_0 \equiv K(t=0)$ and $f(\alpha)$ is a positive but otherwise arbitrary function.

Choosing the right spatial gauge (i.e. $\beta(x^\mu)$)



A high value of the metric components means that the distance between numerical grid points is actually large and this causes problems.

In addition, large gradients my pose a numerical problem.

Choosing a "bad" shift may even lead to coordinates singularities.

A popular choice for the shift is the hyperbolic "Gamma-driver" condition.

$$\begin{aligned} \partial_t \beta^i - \beta^j \partial_j \beta^i &= \frac{3}{4} \alpha B^i \,, \\ \partial_t B^i - \beta^j \partial_j B^i &= \partial_t \tilde{\Gamma}^i - \beta^j \partial_j \tilde{\Gamma}^i - \eta B^i \,, \end{aligned}$$

where $B^i \equiv \partial_t \beta^i$ and η acts as a restoring force to avoid large oscillations in the shift and the driver tends to keep the Gammas constant.

Overall, the "1+log" slicing condition and the "Gamma-driver" shift condition are the most widely used both in vacuum and non-vacuum spacetimes.

In practice, the ADM equations are essentially never used!

The ADM equations are perfectly all right mathematically but not in a form that is well suited for numerical implementation.

Indeed the system can be shown to be weakly hyperbolic (namely, its set of eigenvectors is not complete) and hence “ill-posed” (namely, as time progresses, some norm of the solution grows more than exponentially in time).

In practice, numerical instabilities rapidly appear that destroy the solution exponentially.

However, the stability properties of numerical implementations can be improved by introducing certain new auxiliary functions and rewriting the ADM equations in terms of these functions.

Let's inspect the 3+1 evolution equations again:

$$\begin{aligned}\partial_t K_{ij} = &-D_i D_j \alpha + \alpha(R_{ij} - 2K_{ik}K^{kj} + KK_{ij}) \\ &-8\pi\alpha(R_{ij} - \frac{1}{2}\gamma_{ij}(S-e)) + \mathcal{L}_{\boldsymbol{\beta}} K_{ij}\end{aligned}$$

$$\partial_t \gamma_{ij} = -2\alpha K_{ij} + \mathcal{L}_{\boldsymbol{\beta}} \gamma_{ij}$$

Think of the above system as a second-order system for γ_{ij}. In this equation there are mixed second derivatives, since R_{ij} contains mixed derivatives in addition to a Laplace operator acting on γ_{ij}.

Instead, we would like to have only terms of the type $\partial^\mu \partial_\mu \gamma_{ij}$ because without the mixed derivatives the 3+1 ADM equations could be written in a such way that they behave like a wave equation for γ_{ij}.

We care to have wave equations, like

$$\Box\phi = \partial_t^2\phi - \partial^i\partial_i\phi = 0$$

because wave equations are manifestly hyperbolic and mathematical theorems guarantee the existence and uniqueness of the solutions (as seen in previous lectures of this school).

We can make the equations manifestly hyperbolic in different ways, including:

i) using a clever specific gauge;

ii) introducing new variables, which follow additional equations (imposed by the original system).

These methods aim at removing the mixed-derivative term.

Generalized Harmonic Coordinate (GHC) formulation

Let choose a clever gauge that makes the ADM equations strongly hyperbolic.

The generalized harmonic formulation is based on a generalization of the harmonic coordinates:

$$\Box x^{\mu} = 0$$

When such condition is enforced in the Einstein equations, the principal part of the equations for each metric element becomes a scalar wave equation, with all nonlinearities and couplings between the equations relegated to lower order terms.

The harmonic condition $\Box x^{\mu} = 0$ is known to suffer from pathologies.
However, alternatives can be found that do not suffer from such pathologies and still have the desired properties of the harmonic formulation. In particular, the generalized harmonic coordinates have the form:

$$\Box x^{\mu} = H^{\mu}$$

where the H^{μ} are a set of source functions.

The H^{μ} and the equations for their evolution must be suitably chosen.

The second method: introducing new variables

New evolution variables are introduced to obtain from the ADM system a set of equations that is strongly hyperbolic. A successful set of new evolution variables is:

$$\phi = \frac{1}{12}\ln(\det(\gamma_{ij})) = \frac{1}{12}\ln(\gamma),$$

ϕ : conformal factor

$$\tilde{\gamma}_{ij} = e^{-4\phi}\gamma_{ij},$$

$\tilde{\gamma}_{ij}$: conformal 3-metric

$$K = \gamma^{ij}K_{ij},$$

K : trace of extrinsic curvature

$$\tilde{A}_{ij} = e^{-4\phi}(K_{ij} - \frac{1}{3}\gamma_{ij}K),$$

$\tilde{A}_{ij}$: trace-free conformal extrinsic curvature

$$\Gamma^i = \gamma^{jk}\Gamma^i_{jk}$$

$$\tilde{\Gamma}^i = \tilde{\gamma}^{jk}\tilde{\Gamma}^i_{jk}$$

$\tilde{\Gamma}^i$:“Gammas”

And the ADM equations are then rewritten as:

$$\mathcal{D}_t \tilde{\gamma}_{ij} = -2\alpha \tilde{A}_{ij} \ , \qquad \text{where } \mathcal{D}_t \equiv \partial_t - \mathcal{L}_\beta$$

$$\mathcal{D}_t \phi = -\frac{1}{6}\alpha K \ ,$$

$$\mathcal{D}_t \tilde{A}_{ij} = e^{-4\phi}\left[-\nabla_i \nabla_j \alpha + \alpha\left(R_{ij} - S_{ij}\right)\right]^{\mathrm{TF}} + \alpha\left(K\tilde{A}_{ij} - 2\tilde{A}_{il}\tilde{A}^l_j\right) \ ,$$

$$\mathcal{D}_t K = -\gamma^{ij}\nabla_i \nabla_j \alpha + \alpha\left[\tilde{A}_{ij}\tilde{A}^{ij} + \frac{1}{3}K^2 + \frac{1}{2}\left(\rho + S\right)\right] \ ,$$

$$\begin{aligned}\mathcal{D}_t \tilde{\Gamma}^i = & -2\tilde{A}^{ij}\partial_j \alpha + 2\alpha\left(\tilde{\Gamma}^i_{jk}\tilde{A}^{kj} - \frac{2}{3}\tilde{\gamma}^{ij}\partial_j K - \tilde{\gamma}^{ij}S_j + 6\tilde{A}^{ij}\partial_j \phi\right) \\ & -\partial_j\left(\beta^l \partial_l \tilde{\gamma}^{ij} - 2\tilde{\gamma}^{m(j}\partial_m \beta^{i)} + \frac{2}{3}\tilde{\gamma}^{ij}\partial_l \beta^l\right) \ .\end{aligned}$$

These equations are also known as the BSSN-NOK equations or as the conformal traceless formulation of the Einstein equations.

Although not self evident, the BSSN-NOK equations are strongly hyperbolic with a structure which is resembling the 1st-order in time, 2nd-order in space formulation

$$\Box\phi = 0 \quad \Longleftrightarrow \begin{cases} \partial_t \phi = \psi \\ \\ \partial_t \psi = \partial^i \partial_i \phi \end{cases}$$

scalar wave equation

$$\begin{cases} \partial_t \tilde{\gamma}_{ij} \propto \tilde{A}_{ij} \\ \\ \partial_t \tilde{A}_{ij} \propto D^i D_i \tilde{\gamma}_{ij} \end{cases}$$

conformal traceless formulation

The BSSN-NOK equations are nowadays the most widely used form of the Einstein equations and have demonstrated to lead to stable and accurate evolution of vacuum (binary--black-holes) and non-vacuum (neutron-stars) spacetimes.

Constraints of BSSN-NOK

In addition to the (6+6+3+1+1=17) hyperbolic evolution equations to be solved from one time slice to the next, there are the usual 3+1=4 elliptic constraint equations:

$$\mathcal{H} \equiv {}^{(3)}\!R + K^2 - K_{ij}K^{ij} = 0\,, \quad \text{(Hamiltonian constraint)}$$

$$\mathcal{M}^i \equiv D_j(K^{ij} - g^{ij}K) = 0\,, \quad \text{(momentum constraints)}$$

NOTE: most often these equations are not solved but only monitored to verify that

$$||\mathcal{H}|| \simeq ||\mathcal{M}^i|| < \varepsilon \sim 10^{-4} - 10^{-2}$$

and 5 additional constraints are introduced by the new variables:

$$\det(\tilde{\gamma}_{ij}) = 1, \qquad \mathrm{tr}(\tilde{A}_{ij}) = 0,$$
$$\tilde{\Gamma}^i = \tilde{\gamma}^{jk}\tilde{\Gamma}^i_{jk}.$$

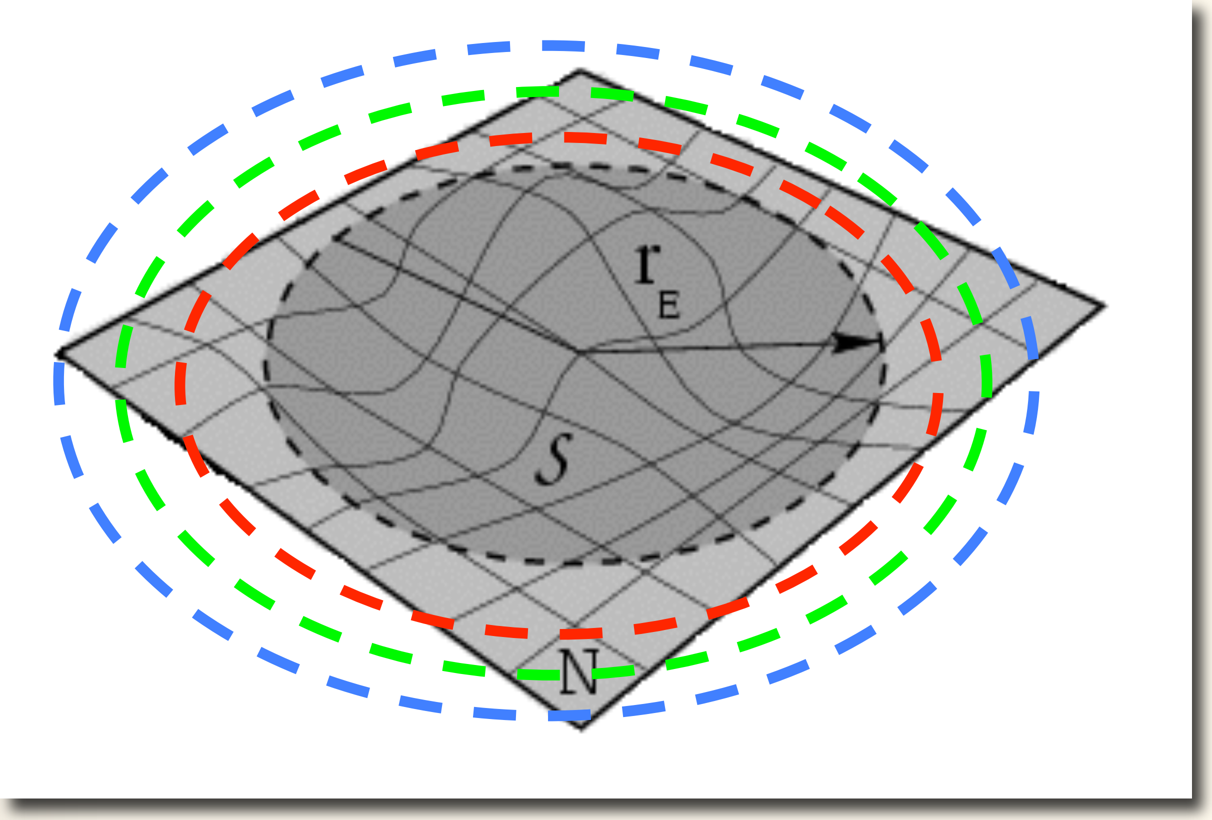
Extraction of gravitational waves

Wave-extraction techniques

Computing the waveforms is the ultimate goal of a large portion of numerical relativity.

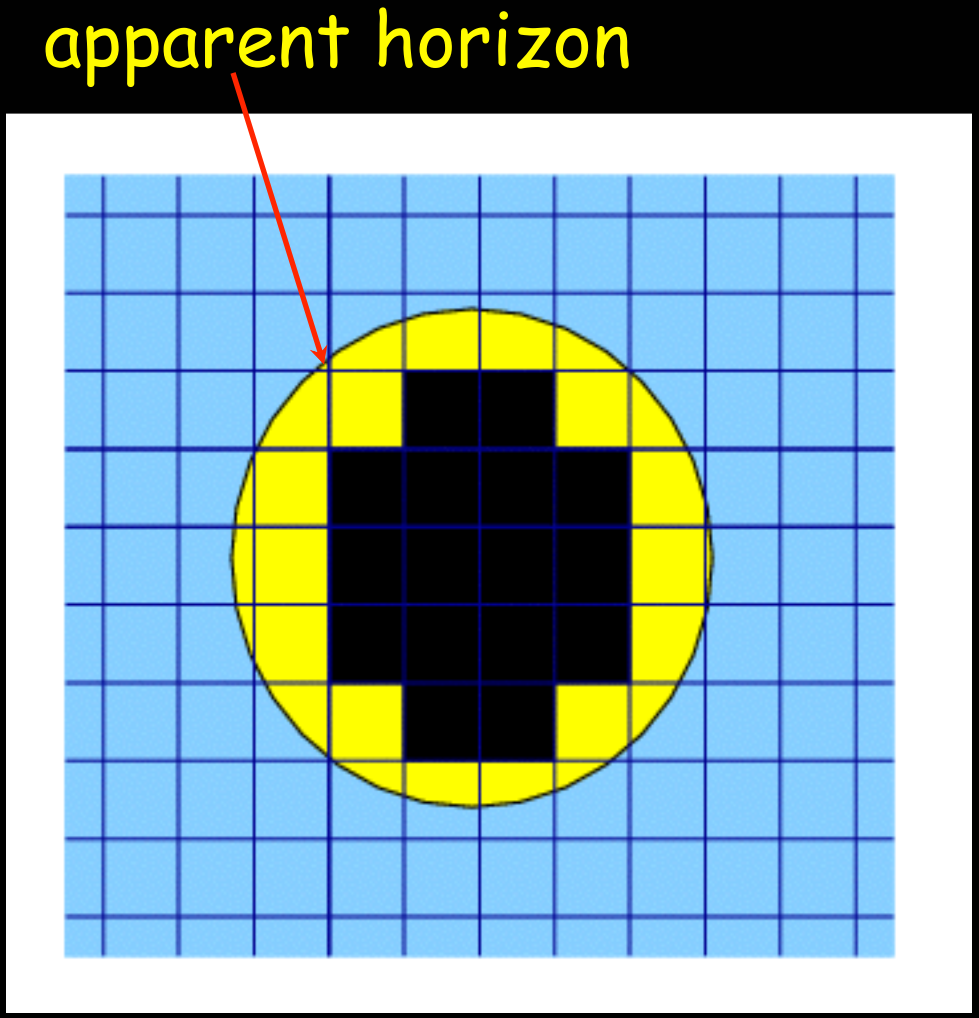
There are several ways of extracting GWs from numerical relativity codes, the most widely used of which are:

- Weyl scalars (a set of five complex scalar quantities, describing the curvature of a 4D spacetime)
- perturbative matching to a Schwarzschild background



Both are based of finding some gauge invariant quantities or the perturbations of some gauge-invariant quantity, and to relate them to the gravitational waveform.

Excising parts of the spacetime with singularities



In practice, the actual excision region is a “legosphere” (black region) and is placed well inside the apparent horizon (which is found at every time step) and is allowed to move on the grid.

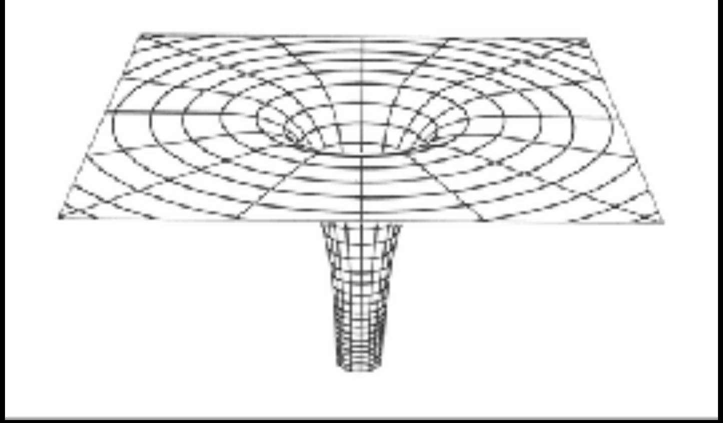
The region of spacetime inside a horizon (yellow region) is causally disconnected from the outside (blue region).

So a region inside a horizon may be excised from the numerical domain.

This is successfully done in pure spacetime evolutions since the work of Nadëzhin Novikov Polnarev (1978).

Baiotti et al. [PRD 71, 104006, (2005)] and other groups [Duez et al., PRD 69, 104016 (2004)] have shown that it can be done also in non-vacuum simulations.

Excising parts of the spacetime with singularities: the moving-puncture method



There is an alternative to explicit excision. Proposed independently by Campanelli et al., PRL96, 111101 (2006) and Baker et al., PRL96, 111102 (2006), it is nowadays a very popular method for moving--black-hole evolution.

It consists in using coordinates that allow the punctures (locations of the singularities) to move through the grid, but do not allow any evolution at the puncture point itself (i.e., the lapse is forced to go to zero at the puncture, though not the shift vector, hence the "frozen" puncture can be advected through the domain). The conditions that have so far proven successful are modifications to the so-called 1+log slicing and Gamma-driver shift conditions.

In practice, such gauges take care that the punctures are never located at a grid point, so that actually no infinity is present on the grid. This method has proven to be stable, convergent, and successful. A few such gauge options are available, with parameters allowed to vary in determined ranges (but no fine tuning is necessary).

This mechanism works because it implements an effective excision ("excision without excision"). It has been shown that in the coordinates implied by the employed gauges the singularity is actually always outside the numerical domain.

A phase transition in numerical simulations

- Efforts to simulate BHs started in the 1970s, but little progress until 2005
- **Pretorius, PRL 95, 121101 (2005), gr-qc/0507014**
  - generalized harmonic coordinates; excision
- **Campanelli, et al., PRL 96, 111101 (2006), gr-qc/0511048**

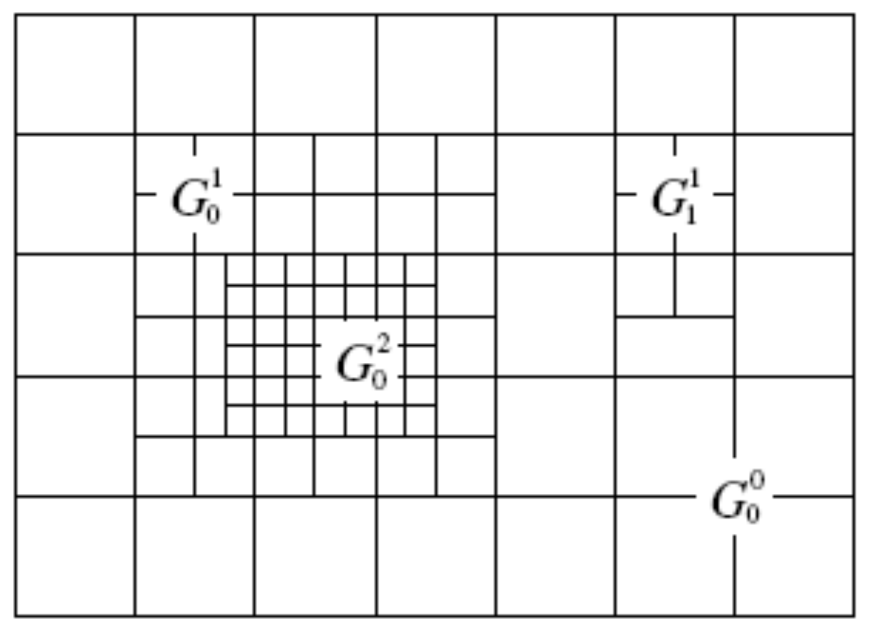
  and

  Baker, et al., PRL 96, 111102 (2006), gr-qc/0511103

  (simultaneous, independent discovery)
  - moving "punctures" across the grid
  - in the "puncture" method the solution is factored into an analytical part, which contains the singularity, and a numerically constructed part, which is then singularity free
  - appropriate gauge conditions make sure that the location of the the singularity is never on a grid point and that gradients near it are not too large
  - Effective resolution is very poor near the singularity locations, but this is fine because these regions are inside horizons

Carpet: a mesh-refinement driver

Carpet (www.carpetcode.org) is a driver mainly developed by E. Schnetter [CQG 21, 1465 (2004)], which has removed the limitation of using uniform 3D grids.



Carpet follows a (simplified) Berger-Oliger [**J. Comput. Phys. 53, 484 (1984)**] approach to mesh refinement, that is:

- refined subdomains consist of a set of cuboid (= rectangular parallelepiped) grids
- refined subdomains have boundaries aligned with the grid lines
- the refinement ratio between refinement levels is constant

While the refined meshes are not automatically moving on the grid, they can be activated and deactivated during the evolution, obtaining a progressive fixed mesh refinement or even a moving-grid mesh refinement.

When the spacetime is not vacuum

When the spacetime is not vacuum

The evolution equations of the matter are given by the conservation of the baryon number and energy-momentum:

$$\begin{aligned} \nabla_\mu T^{\mu\nu} &= 0 \\ \nabla_\mu J^\mu &= 0 \\ J^\mu &\equiv \rho u^\mu \end{aligned}$$

plus an Equation of State P=P(ρ,ε) (microphysics input).

Simple EoSs are:

the ideal-fluid EoS: $P = (\gamma - 1)\varepsilon\rho$

the polytropic EoS: $P = k\rho^\gamma$

Fluids and shocks

Fluids and shocks

Fluids

A fluid is described as a set of fluid elements, or of volumes of fluid with the following properties:

- they are much smaller than the typical size of the macroscopic system they are part of

↓

they can be considered point-like particles for all physical and mathematical purposes

- their linear dimensions are much larger than the collision mean free path of the particles (molecules, atoms, nuclei, electrons, nucleons,…) they contain

↓

such fluid elements can be considered representative of the mean quantities of the particles they contain

Fluids and shocks

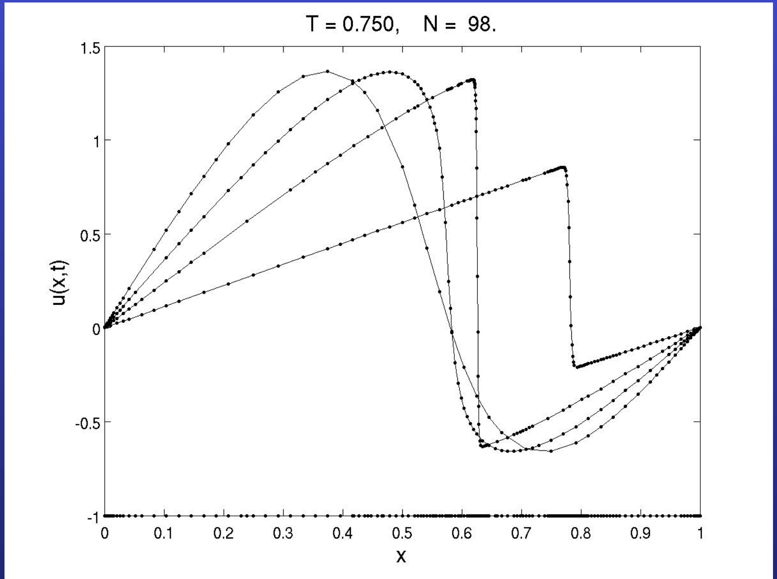
The **hydrodynamics equations** allow for the formation of

arbitrarily large and arbitrarily localized

spatial variations

of the hydrodynamical variables,

which are mathematically treated as **discontinuities** and are often referred to as **shocks**.

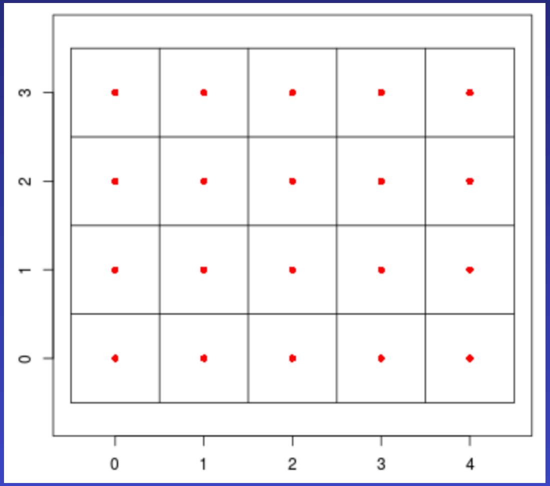


(inviscid) Burger's equation

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = 0$$

Finite-volume methods and conservative systems

Finite-difference
vs
finite-volume
methods



• ***Finite-difference* methods are numerical methods that approximate the solutions to differential equations using finite difference equations to approximate derivatives**

• ***Finite-volume* methods are based on subdividing the spatial domain into intervals ("finite volumes" or *grid cells*) and keeping track of an approximation to the integral:**

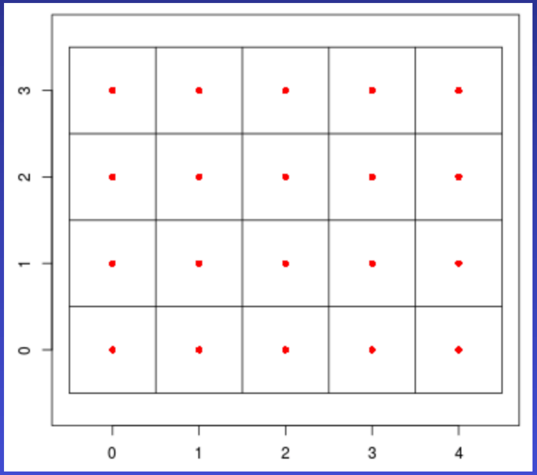
$$\overline{\mathrm{u}}_{\mathrm{i}}^{\mathrm{n}} = \frac{1}{\text{volume of cell}_i} \int_{\text{cell}_i} u(x,t_n)\mathrm{d}x = \frac{1}{\Delta x} \int_{x_{i-1/2}}^{x_{i+1/2}} u(x,t_n)\mathrm{d}x$$

$$\Delta x = x_{i+1/2} - x_{i-1/2}$$

over each of these cells.

Advantages of finite-volume methods

At each time step, the **update** of the grid cells are obtained using (approximations to) the **flux** through the cell boundaries.



Therefore, the integral of *u* over the entire interval *[a,b]*

$$\int_a^b u(x,t_n)\,\mathrm{d}x \approx \sum_{\mathrm{i}=1}^{\mathrm{N}} \overline{\mathrm{u}}_{\mathrm{i}}^{\mathrm{n}} \Delta x$$

this discrete sum will change only due to the fluxes at the domain boundaries *x=a* and *x=b*. So, for example, **the total mass is preserved**, or at least varies correctly, provided that the boundary conditions are properly imposed.

Conservative form

If the homogeneous partial differential equation

$$\partial_t u(x,t) + a[u(x,t)]\partial_x u(x,t) = 0$$

is written in the form

$$\partial_t u(x,t) + \partial_x f[u(x,t)] = 0$$

it is said to be in **conservative form.**

In conservative systems, knowledge of the **state vector** *u* at one point in spacetime allows to determine the **flux** *f* (and so the evolution) for each state variable.

Weak solutions

The partial differential equation

$$\partial_t \mathrm{u} + \partial_x \mathrm{f}(\mathrm{u}) = 0$$

is not valid in the classical sense for solutions containing shocks (discontinuities).

However, its integral form

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{x_{i-1/2}}^{x_{i+1/2}} u(x,t)\mathrm{d}x = f[u(x_{i-1/2},t)] - f[u(x_{i+1/2},t)]$$

holds.

The solution to the integral form of the equation is called **weak solution**.

Conservative form: theorems

Lax Wendroff:

If a method in conservative form converges, then it converges to the weak solution of the conservation laws

Hou LeFloch:

If a method in non-conservative form converges, then in the presence of a shock wave it converges to the wrong solution.

In conclusion, in the presence of shocks, only converging conservative numerical methods converge to the correct weak solution of the problem (which is what we want).

References:

- **Lax Wendroff, *Comm.Pure.Appl.Math.*, 13, p.217**
- **Hou LeFloch, *Math. Of Comp.*, 62, p. 497**

Finite-volume numerical methods

The integral form of the equation: $\partial_t \mathrm{u} + \partial_x \mathrm{f}(\mathrm{u}) = 0$

namely: $$\frac{\mathrm{d}}{\mathrm{d}t}\int_{x_{i-1/2}}^{x_{i+1/2}} u(x,t)\mathrm{d}x = f[u(x_{i-1/2},t)] - f[u(x_{i+1/2},t)]$$

suggests that we should study numerical methods in the form:

$$\frac{1}{\Delta x}\int_{x_{i-1/2}}^{x_{i+1/2}} u(x,t_{n+1})\mathrm{d}x = \frac{1}{\Delta x}\int_{x_{i-1/2}}^{x_{i+1/2}} u(x,t_n)\mathrm{d}x + \frac{1}{\Delta x}\left(\int_{t_n}^{t_{n+1}} f[u(x_{i-1/2},t)]\mathrm{d}t - \int_{t_n}^{t_{n+1}} f[u(x_{i+1/2},t)]\mathrm{d}t\right)$$

or, in more compact form,

$$\overline{u}_{\mathrm{i}}^{\mathrm{n}+1} = \overline{u}_{\mathrm{i}}^{\mathrm{n}} - \frac{\Delta t}{\Delta x}\left(F_{i+1/2} - F_{i-1/2}\right)$$

where

$$F_{i-1/2} \approx \frac{1}{\Delta t}\int_{t_n}^{t_{n+1}} f[u(x_{i-1/2},t)]\mathrm{d}t$$

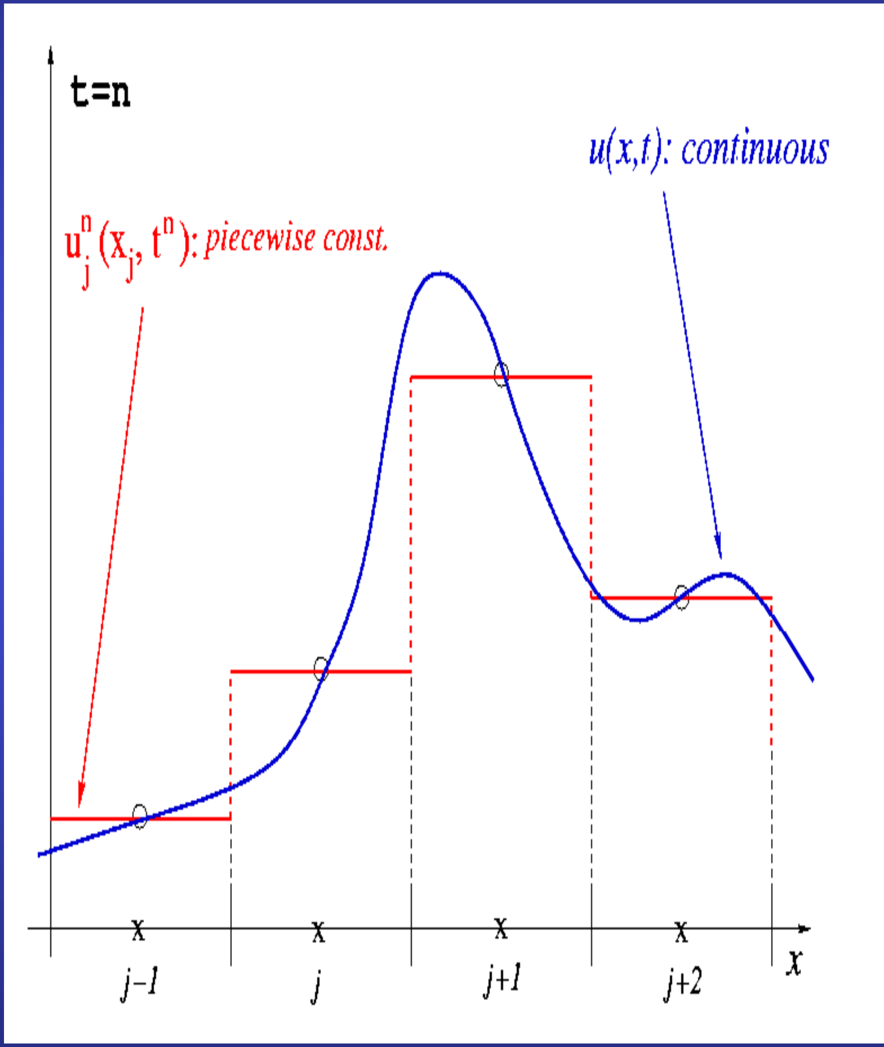
Different numerical schemes differ in the prescription for computing the **flux function** *F*.

Discontinuities and numerical schemes

The problem of discretisation

The nonlinear properties of the hydrodynamical equations for compressible fluids *generically* produce, in a finite time, *nonlinear waves* with discontinuities (i.e. shocks) even from smooth initial data.

In general, difficulties arise when a Cauchy problem described by a set of *continuous* PDEs is solved in a *discretised form:* the numerical solution is, at best, *piecewise constant.*



Discontinuities and numerical schemes

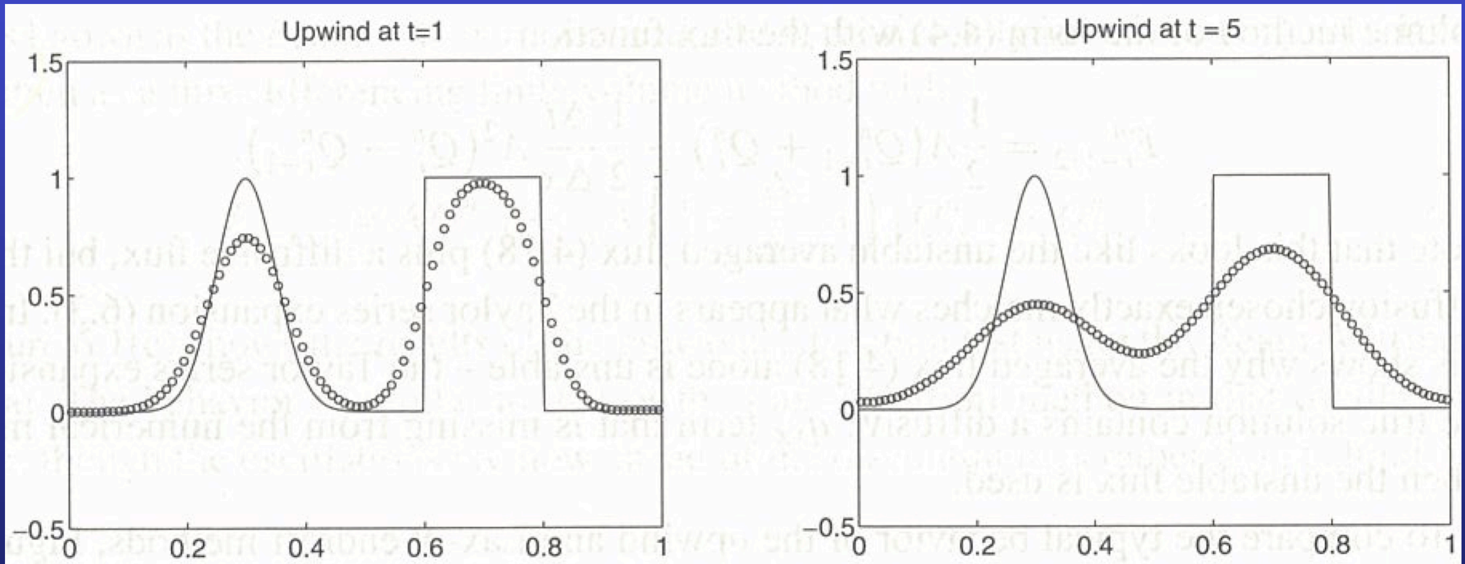
Since the occurrence of discontinuities is a fundamental property of the hydrodynamical equations, any numerical scheme must be able to handle them in a satisfactory way.

Possible solutions to the discontinuity problem:

❖ **1st order accurate schemes**

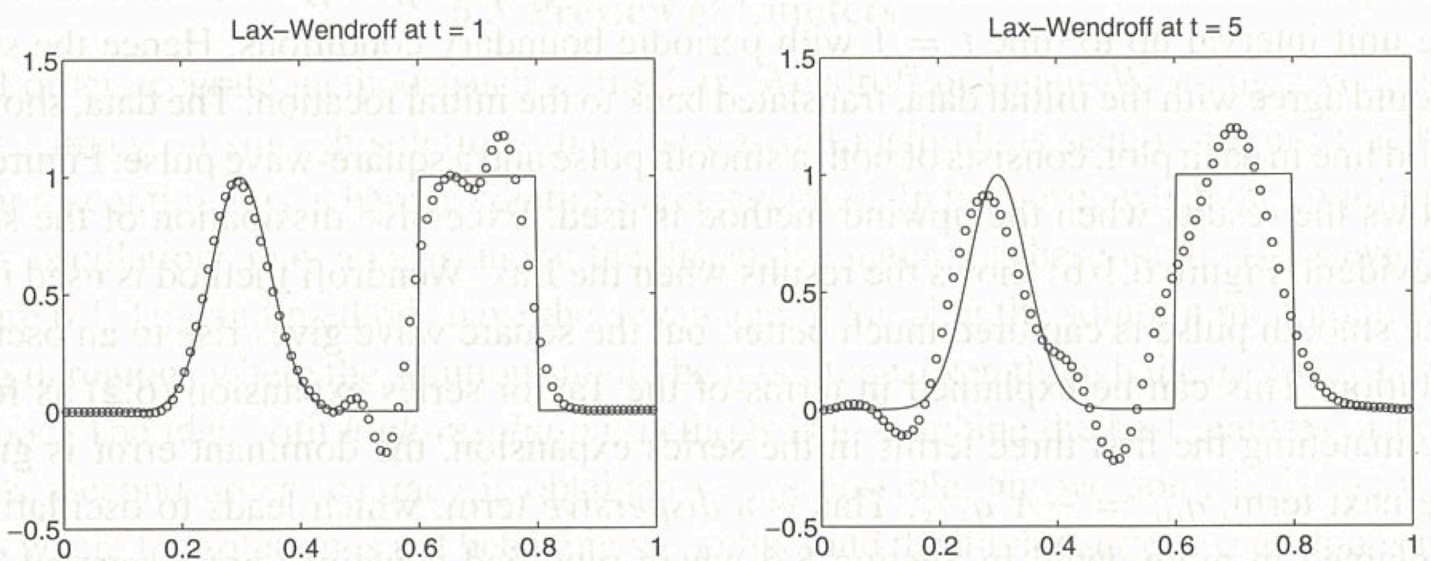
- generally fine, but very inaccurate across discontinuities (excessive diffusion); e.g. Lax-Friedrichs method

$$\overline{u}_{\mathrm{i}}^{\mathrm{n+1}} = \frac{1}{2}(\overline{u}_{\mathrm{i+1}}^{\mathrm{n}} + \overline{u}_{\mathrm{i\text{-}1}}^{\mathrm{n}}) - \frac{\Delta t}{2\Delta x}\left(F_{i+1} - F_{i-1}\right)$$



Discontinuities and numerical schemes

- **2nd order accurate schemes**
  - more accurate, but generally introduce oscillations across discontinuities and are dispersive even on smooth data (especially for steep gradients), causing waves to move with a wrong group velocity (e.g. Lax-Wendroff method)



Discontinuities and numerical schemes

- **2nd order accurate schemes with artificial viscosity**
  - mimic Nature, but problem-dependent and inaccurate for ultrarelativistic flows

- **Godunov methods**
  - discontinuities are not eliminated, rather they are exploited
  - based on the solution of Riemann problems
  - approximately second-order schemes can be derived
  - state of the art in relativistic hydrodynamics

The Riemann problem

Riemann problem

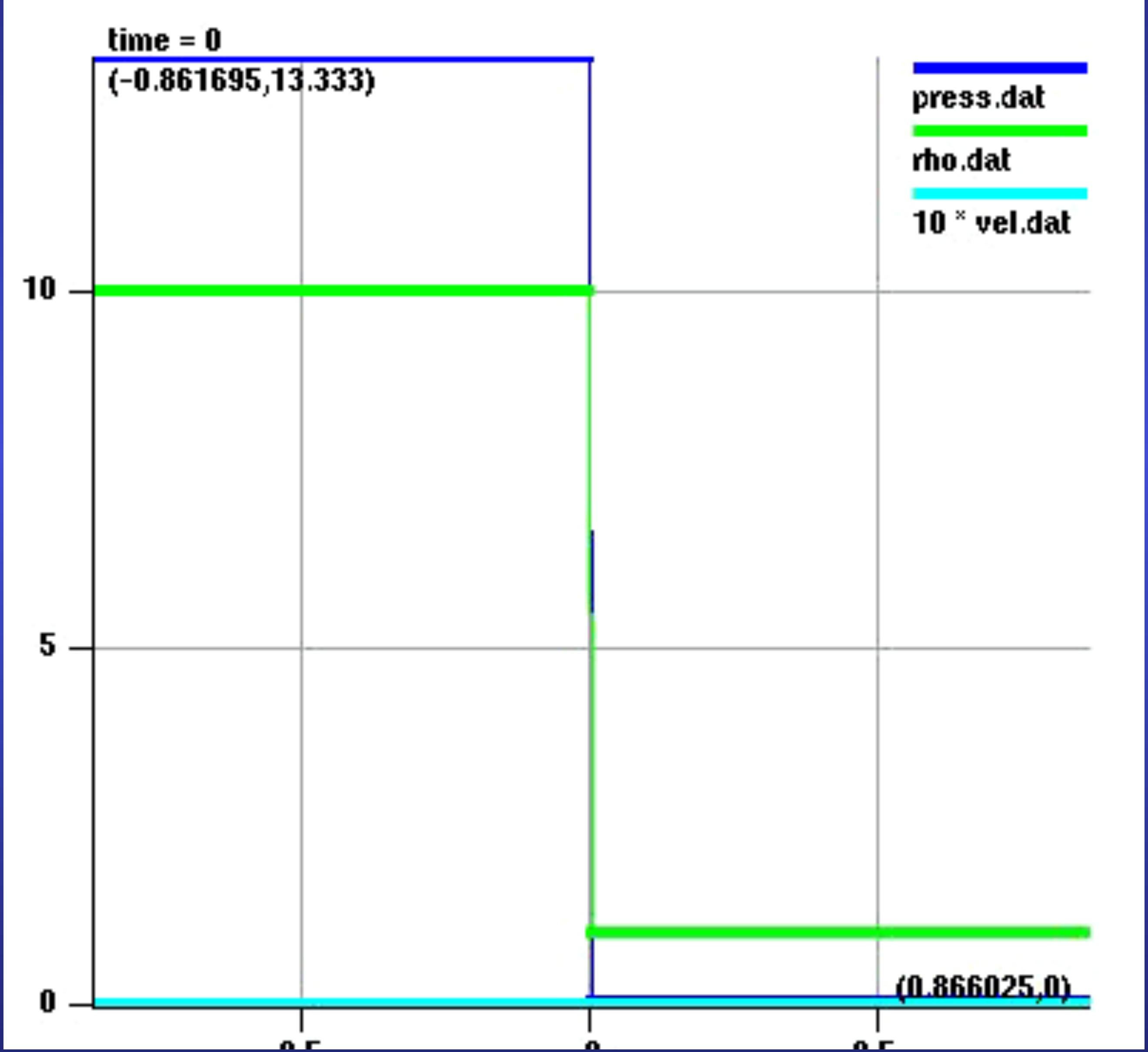
Definition: in general, for a hyperbolic system of equations, a Riemann problem is an initial-value problem with initial condition given by:

$$U(x,0) = \begin{cases} U_L & \text{if} \quad \mathrm{x} < 0 \\ U_R & \text{if} \quad \mathrm{x} > 0 \end{cases}$$

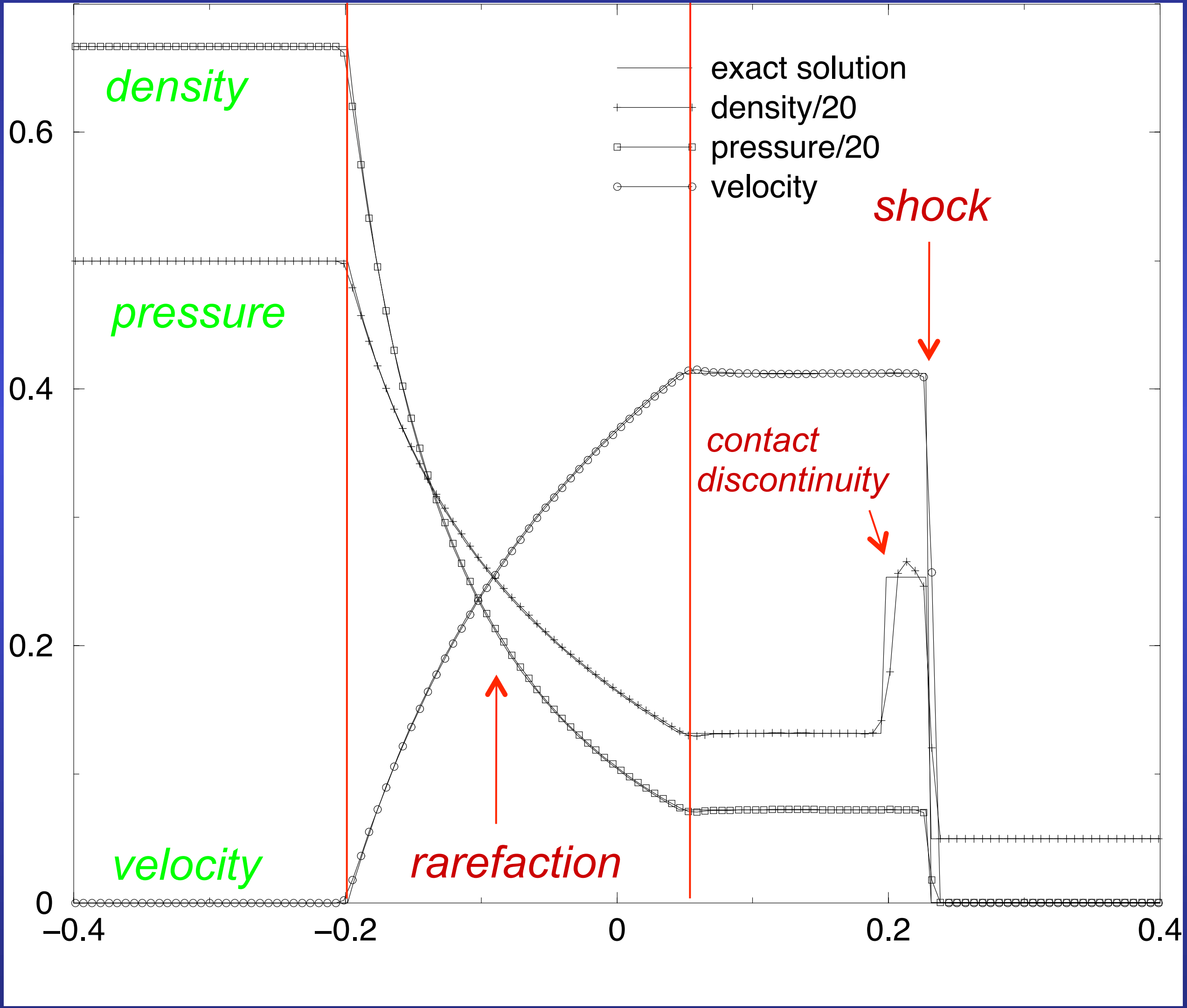
where U_L and U_R are two constant vectors representing the left and right state.

For hydrodynamics, a (physical) Riemann problem is the evolution of a fluid initially composed of two states with different and constant values of velocity, pressure and density.

Riemann problem



Riemann problem

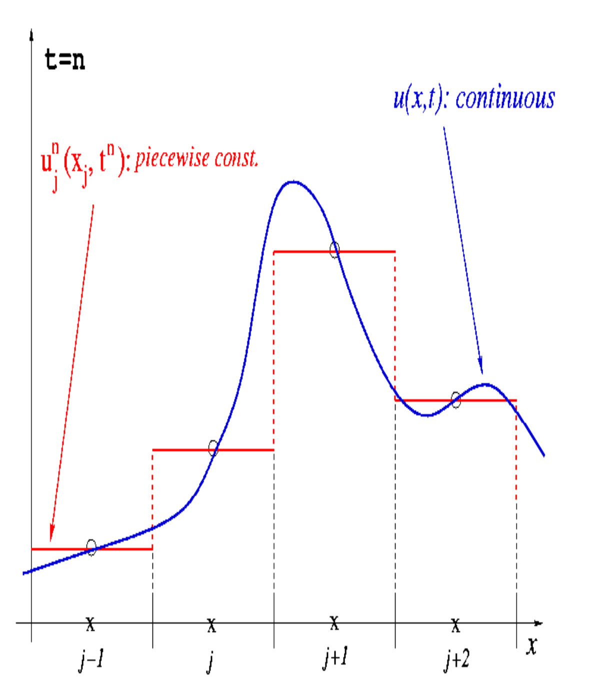


Godunov’s idea

Godunov's idea

Core idea:

a piecewise-constant description of hydrodynamical quantities will produce a collection of local Riemann problems, whose solution can be found exactly (if one wishes).



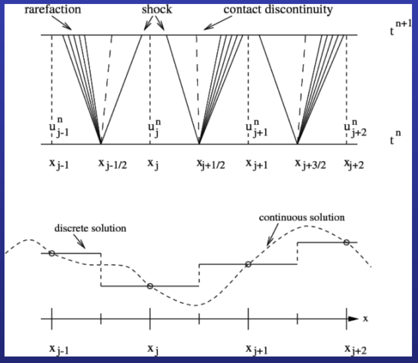
This is an example of how research in basic physics can boost computational methods.

Godunov's idea: notes

- There are powerful methods to solve Riemann problems either exactly or approximately.

- Piecewise constant reconstruction leads naturally to Riemann problems (but gives only a first-order accurate method).

- The solution at time t_{n+1} can be constructed by piecing together the Riemann solutions, provided that the time step is short enough (CFL condition) that the waves from two adjacent Riemann problems have not yet started to interact.



High Resolution Shock Capturing methods

- Godunov’s method with piecewise constant reconstruction is only first order.

- HRSC methods are a compromise between these two options: they are second order where the solution is smooth, but they are only first order near discontinuities, because here the monotonicity preservation is more important.

- Higher order is achieved by improved reconstruction methods.

Total Variation

A useful measure of the oscillations present in the numerical solution is provided by the notion of total variation:

$$\mathrm{TV}(Q) \equiv \sum_{i=-\infty}^{\infty} |Q_i - Q_{i-1}|$$

For this to be a meaningful measure, Q must become constant at infinity. Usually Q has compact support, anyway.

A sufficient condition that ensures that a method does not introduce oscillations is that its total variation does not increase:

$$\mathrm{TV}(Q^{n+1}) \leq \mathrm{TV}(Q^n)$$

Total Variation

Methods that satisfy the condition

$$\mathrm{TV}(Q^{n+1}) \le \mathrm{TV}(Q^n)$$

are called Total Variation Diminishing (TVD) methods

(even if *Total Variation Non Increasing* would be more correct).

It is demonstrated (see Toro) that TVD methods cannot be extended to higher than second order. One must then renounce to strict TVD and allow for

$$\mathrm{TV}(Q^{n+1}) \le \mathrm{TV}(Q^n) + O(\Delta x^k)$$

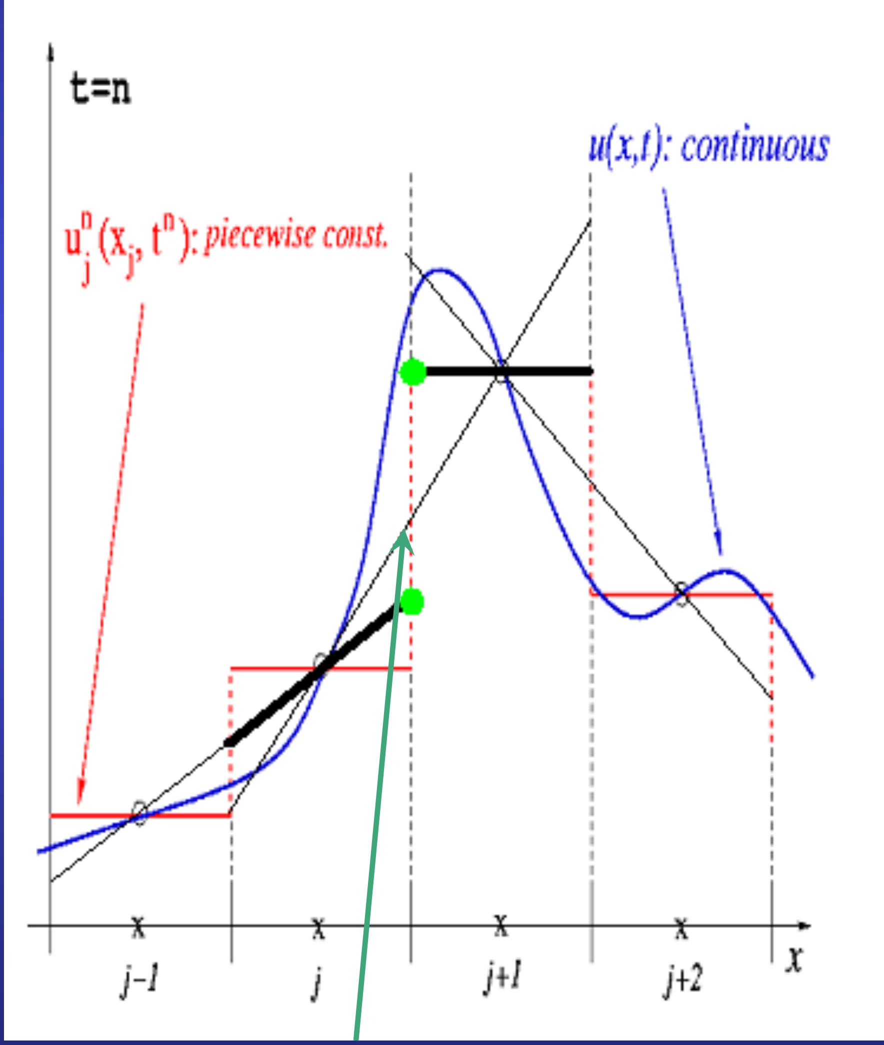
The resulting methods are called Essentially-Non-Oscillatory (ENO) methods.

Reconstruction

In HRSC methods, higher order of accuracy is reached with a better representation of the solution: that is with a *”reconstruction”* of the solution.

Such a reconstruction can be made with different algorithms.

The values of the reconstructed function on the cell boundaries are then used as the initial data for the Riemann problems at the cell boundaries.



“Riemann problems”

TVD Reconstruction

TVD reconstruction methods consist in approximating the solution with a piecewise linear function.

Different TVD reconstruction methods differ in the way in which the slope (linear approximation) is selected. The selected slope is a combination of the upwind S_{up}, downwind S_{down} and central S_c slopes.

$$S_{up} = \frac{x_i - x_{i-1}}{\Delta x}$$

$$S_{down} = \frac{x_{i+1} - x_i}{\Delta x}$$

$$S_c = \frac{x_{i+1} - x_{i-1}}{2\Delta x}$$

TVD Reconstruction: examples

- **minmod (by P. Roe): if *S_{up}* and *S_{down}* have the same sign, selects the slope with the minimum modulus; otherwise (*i.e.* at extrema) selects zero:**

$$\min\mathrm{mod}(a,b) \equiv \begin{cases} a & \text{if } |a|<|b| \quad \text{and} \quad ab>0 \\ b & \text{if } |a|>|b| \quad \text{and} \quad ab>0 \\ 0 & \text{if} \quad ab<0 \end{cases}$$

It is the simplest and most diffusive slopelimiter.

- **superbee (by P. Roe): selects the maximum modulus between** minmod*(S_{up}, $2S_{down}$)* **and** minmod*($2S_{up}$, S_{down})*.
It sharpens discontinuities, but steepens and "squares" also smooth profiles.

- **monotonized central-difference (by B. van Leer):**
selects minmod*($2S_{up}$, $2S_{down}$, S_c)*.
It reduces to the central slope away from discontinuities, so avoiding the problems of superbee. It is a good default limiter for a large class of problems.

ENO Reconstruction

cfr. e.g.: Shu C.W., in T.J. Barth, H. Deconinck (eds.),
High-Order Methods for Computational Physics, Springer (1999)

As said earlier, ENO methods are not TVD, but

$$\mathrm{TV}(Q^{n+1}) \leq \mathrm{TV}(Q^{n}) + O(\Delta x^{k+1})$$

their total variation is bound to grow only *slowly*.

ENO methods are more expensive than TVD ones.

There are numerous variants of ENO methods. The basic idea is to choose a stencil including *s+t<k+1* cells (*s* to he left of the point and *t* to its right), so that the smoothest reconstruction is achieved.

ENO Reconstruction

The smoothness is measured in terms of the Newton divided differences:

$$n[x_{i-1},x_i] \equiv \frac{u_i - u_{i-1}}{x_i - x_{i-1}}$$

$$n[x_{i-s},x_{i+t}] \equiv \frac{u[x_{i-s+1},x_{i+t}] - u[x_{i-s},x_{i+t-1}]}{x_{i+t} - x_{i-s}}$$

The property:

$$n[x_{i-s},x_{i+t}] = \frac{u^{(t+s)}(\xi)}{(t+s)!}$$

where $n^{(t+s)}$ is the (t+s)-th derivative

illustrates how minimising the Newton divided differences provides the smoothest reconstruction.

After the stencil giving the minimum Newton divided differences is found, a *k*-order polynomial interpolation gives the reconstructed value on the *i*-th cell interface.

Piecewise Parabolic Method (PPM)

Colella, Woodward, *J. Comput. Phys*, **54**, 174 (1984)

- **PPM is a rather more complex, composite procedure to achieve, theoretically, third order accuracy.**
- **In practice it is not much above second order, but it is more accurate.**
- **It has several adjustable parameters; which add to its complexity**
- **The basic idea is to construct in each cell an interpolating parabola, such that its integral average coincides with the known solution and that no new extrema appear in the interpolated function.**

Monotonicity-Preserving (MP5) scheme

Suresh & Huynh, J. Comput. Phys., **136**, 83 (1997)

Mignone et al., J. Comput. Phys., **229**, 5896 (2010)

The MP5 scheme is based on a fifth-order reconstruction combined with a flattening procedure designed to avoid the creation of artificial extrema in the function to be reconstructed.

Summary of reconstruction methods

- Reconstruction methods serve the purpose of increasing the order of accuracy of the scheme
- They *reconstruct* the data on the cell boundaries, starting from the data at the cell centres
- They set the initial conditions for the local Riemann problems
- Different type of reconstruction methods are available: TVD, ENO, …
- Some famous examples:
  - slope limiters: linear but reduce the order to 1 at extrema:
    - minmod: the most diffusive
    - superbee: *squares* waves
    - MC: combines the good properties of the above two methods
  - ENO: any order (in theory)
  - PPM: expensive, but accurate

Summary about Riemann solvers

- Riemann solvers are essential to building numerical methods that well describe discontinuities
- exact Riemann solvers are computationally (prohibitively) costly
- several approximate Riemann solvers have been proposed, e.g.
  - HLLE: simple, robust, computationally relatively inexpensive, rather dissipative
  - Roe: more computationally expensive, less dissipative; problems at sonic points
  - Marquina: like Roe, with improvement at sonic points

References

- R. J. Leveque, Finite Volume Methods for Hyperbolic Problems, Cambridge University Press

- E. F. Toro, Riemann Solvers and Numerical Methods for Fluid Dynamics, Springer

- R. J. Leveque, Numerical Methods for Conservation Laws, Birkhauser Verlag

Systems of PDEs in conservative form

- We have seen the importance of using the **conservative form** of the general-relativistic hydrodynamics equations and **High-Resolution Shock-Capturing** (**HRSC**) methods to solve them

- The PDEs of general-relativistic hydrodynamics using natural hydrodynamical quantities like rest-mass density, internal energy pressure, velocities, are not in a flux-conservative form.

Systems of PDEs in conservative form

To hard-wire the conservative nature of the hydrodynamical equations, the so-called ”primitive variables” are replaced by the “conserved variables”:

$$\boldsymbol{U} = \begin{pmatrix} D \\ S_j \\ \tau \end{pmatrix} := \begin{pmatrix} \rho W \\ \rho h W^2 v_j \\ \rho h W^2 - p - D \end{pmatrix}, \quad \boldsymbol{F}^i = \begin{pmatrix} \alpha v^i D - \beta^i D \\ \alpha S^i{}_j - \beta^i S_j \\ \alpha(S^i - D v^i) - \beta^i \tau \end{pmatrix}$$

$$\boldsymbol{S} := \sqrt{\gamma} \begin{pmatrix} 0 \\ \frac{1}{2}\alpha S^{ik}\partial_j \gamma_{ik} + S_i \partial_j \beta^i - E \partial_j \alpha \\ \alpha S^{ij} K_{ij} - S^j \partial_j \alpha \end{pmatrix}$$

$$\frac{\partial \overline{\mathrm{U}}}{\partial t} + \frac{\partial \overline{\mathrm{F}}^i}{\partial x^i} = \overline{\mathrm{S}}$$

Enthalpy: $h = 1 + \epsilon + p/\rho$

Lorentz factor: *W*

This is called the **Valencia formulation** of general-relativistic hydrodynamics (Banyuls Font Ibáñez Martí Miralles, Ap. J., 476,221)

High Resolution Shock Capturing schemes

In summary HRSC schemes consist in

- conversion of primitive variables to conservative variables
- reconstruction on cell boundaries
- computation of fluxes through solutions of Riemann problems
- time update
- conversion of conservative variables to primitive variables

Conversion from conservative to primitive variables

- The evolved variables are the conserved variables, but in order to compute the fluxes, the source terms (and to have physical insight) it is necessary to transform back the updated result to the primitive variables.
- The conversion from primitive to conservative is given analytically, but converting in the other direction is not possible in closed form. A standard procedure for the conversion consists in solving numerically the equation:

$$p - \bar{p}[\rho(\mathbf{U},\mathbf{p}), \varepsilon(\mathbf{U},\mathbf{p})]$$

where p is the pressure to be found and $\bar{p}$ is the pressure expressed through the EoS, in terms of the updated conserved variables and the pressure p itself.

Conversion from conservative to primitive variables

This is done by inverting the equations

$$\begin{aligned} D &= \rho W \\ S_j &= \rho h W^2 v_j \\ \tau &= \rho h W^2 - \rho W - p \end{aligned}$$

in order to express and in terms of the conserved variables and of the pressure p:

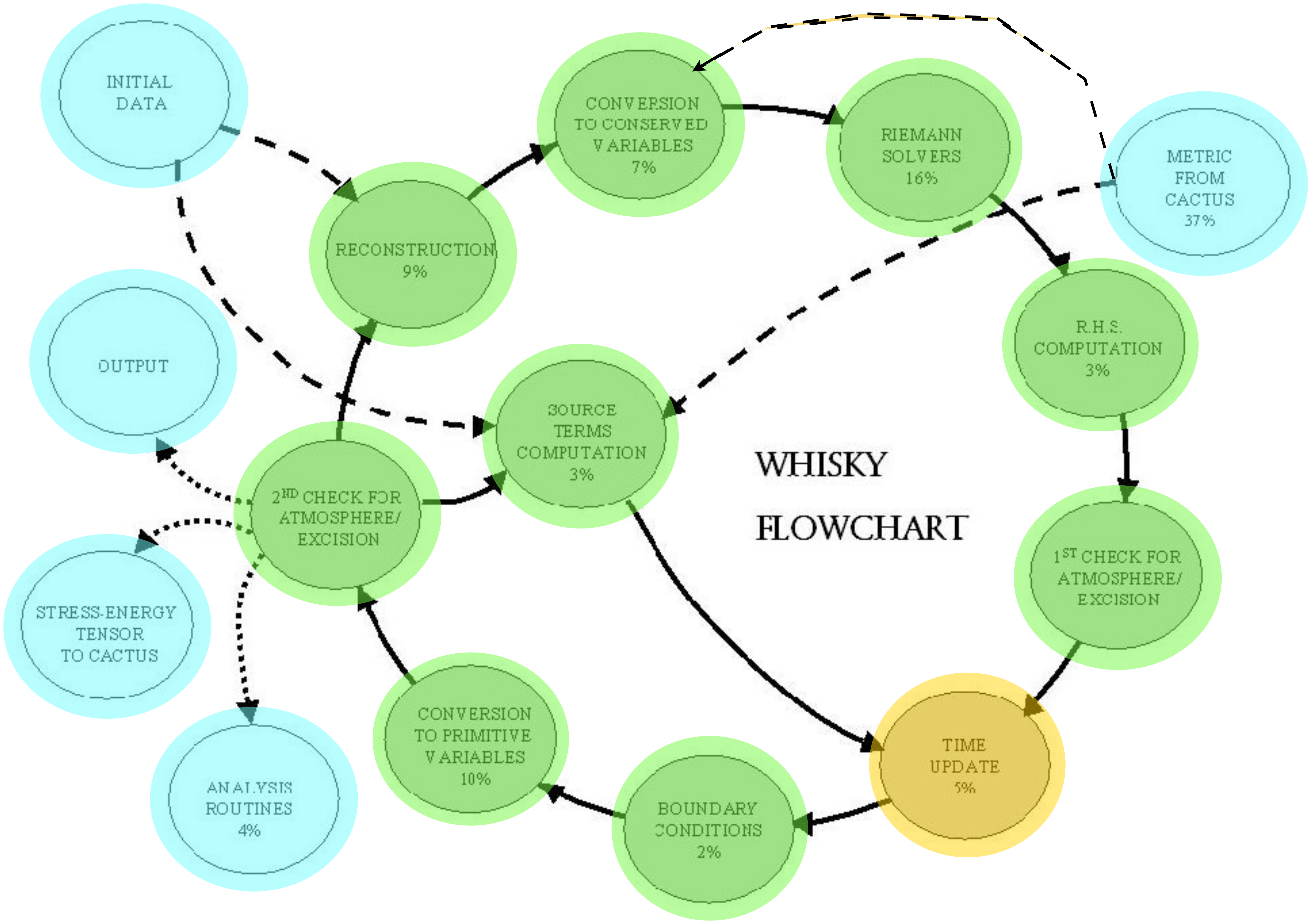
$$\begin{aligned} \rho &= \frac{D}{\tau + p + D}\sqrt{(\tau + p + D)^2 - S^2} \\ \varepsilon &= D^{-1}\left(\sqrt{(\tau + p + D)^2 - S^2} - p\frac{\tau + p + D}{\sqrt{(\tau + p + D)^2 - S^2}} - D\right) \end{aligned}$$

where $S \equiv \gamma^{ij} S_i S_j$

Artificial atmosphere

- In compact-object simulations there are usually parts of the numerical domain where vacuum should be present.
- However, in the vacuum limit the hydrodynamics equations break down, because the speed of sound tends to the speed of light and the Riemann solvers fail. C2P also tends to fail for very low densities.
- To avoid this problem, it is customary to introduce a tenuous atmosphere, i.e. a low-density, low-pressure region surrounding the compact objects.
- The artificial atmosphere is treated as a perfect fluid and its density is set to be several orders of magnitude smaller than the maximum density of the stars. Variations are possible.
- The treatment of the atmosphere is one of the most delicate parts of codes implementing HRSC methods. However, if the control of the atmosphere and the setting of its parameters are carried out with care, the atmosphere usually has negligible influence on the evolution of the compact objects.

A High-Resolution Shock-Capturing scheme


INITIAL DATA
CONVERSION TO CONSERVED VARIABLES 7%
RIEMANN SOLVERS 16%
METRIC FROM CACTUS 37%
RECONSTRUCTION 9%
OUTPUT
R.H.S. COMPUTATION 3%
SOURCE TERMS COMPUTATION 3%
WHISKY
FLOWCHART
2ND CHECK FOR ATMOSPHERE/ EXCISION
1ST CHECK FOR ATMOSPHERE/ EXCISION
STRESS-ENERGY TENSOR TO CACTUS
CONVERSION TO PRIMITIVE VARIABLES 10%
TIME UPDATE 5%
ANALYSIS ROUTINES 4%
BOUNDARY CONDITIONS 2%

For magnetic fields

The Maxwell equations written covariantly using the Farady electromagnetic tensor are

$$\nabla_\nu {}^*F^{\mu\nu} = 0 \qquad {}^*F^{\mu\nu} = \frac{1}{2}\epsilon^{\mu\nu\sigma\rho}F_{\sigma\rho}$$

For a perfect fluid with infinite conductivity (ideal MHD approximation) these become:

the **divergence-free condition**:

$$\nabla \cdot \left(\sqrt{\gamma}\vec{B}\right) = 0$$

and the equations for the **evolution of the magnetic field**:

$$\frac{\partial}{\partial t}\left(\sqrt{\gamma}\vec{B}\right) = \nabla \times \left[\left(\alpha\vec{v} - \vec{\beta}\right) \times \left(\sqrt{\gamma}\vec{B}\right)\right]$$

For magnetic fields

$$T^{\mu\nu} = (\rho + \rho\varepsilon + b^2)u^\mu u^\nu + \left(p + \frac{1}{2}b^2\right) g^{\mu\nu} - b^\mu b^\nu$$

$$b^0 = \frac{WB^i v_i}{\alpha}, b^i = \frac{B^i + \alpha b^0 u^i}{W}, v^i = \frac{u^i}{\alpha u^t} + \frac{\beta^i}{\alpha}$$

where

ρ is the rest-mass density

ε is the specific internal energy

u is the four-velocity

p is the gas pressure

v^i is the Eulerian three-velocity of the fluid (Valencia formulation)

W the Lorentz factor

b the four-vector of the magnetic field

B^i the three-vector of the magnetic field measured by an Eulerian observer

The fundamental equations

$$R_{\mu\nu} - \tfrac{1}{2} g_{\mu\nu} R = 8\pi T_{\mu\nu} \qquad (\text{field eqs}: \ 6+6+3+1 \\ +1+3+1$$

$$\nabla_\mu T^{\mu\nu} = 0\ , \qquad (\text{cons. en./mom.}: \ 3+1)$$

$$\nabla_\mu (\rho u^\mu) = 0\ , \qquad (\text{cons. of baryon no}: \ 1)$$

$$p = p(\rho, \epsilon, \ldots)\ . \qquad (\text{EoS}: \ 1 + \ldots)$$

$$\nabla_\mu {}^*F^{\mu\nu} = 0, \qquad (\text{Maxwell eqs.: induction, zero div.})$$

Additionally, we would like to solve:

- neutrino and photon radiation transport
- nuclear-reaction networks
- multifluids and solids
- high-order, high-accuracy numerical methods

and we would like it fast enough to allow parameter-space exploration!

To know more on numerical relativity and applications

Introduction to 3+1 Numerical Relativity, *Miguel Alcubierre*, Oxford University Press (2008)

Numerical Relativity, *Thomas Baumgarte and Stuart Shapiro*, Cambridge University Press (2010)

Relativistic Hydrodynamics, *Luciano Rezzolla and Olindo Zanotti*, Oxford University Press (2013)

Numerical Relativity, *Masaru Shibata*, World Scientific (2016)