Numerical Relativity and scalar fields

Dr Katy Clough
Ernest Rutherford Fellow
Queen Mary University of London

engrenage

aka Baby GRChombo

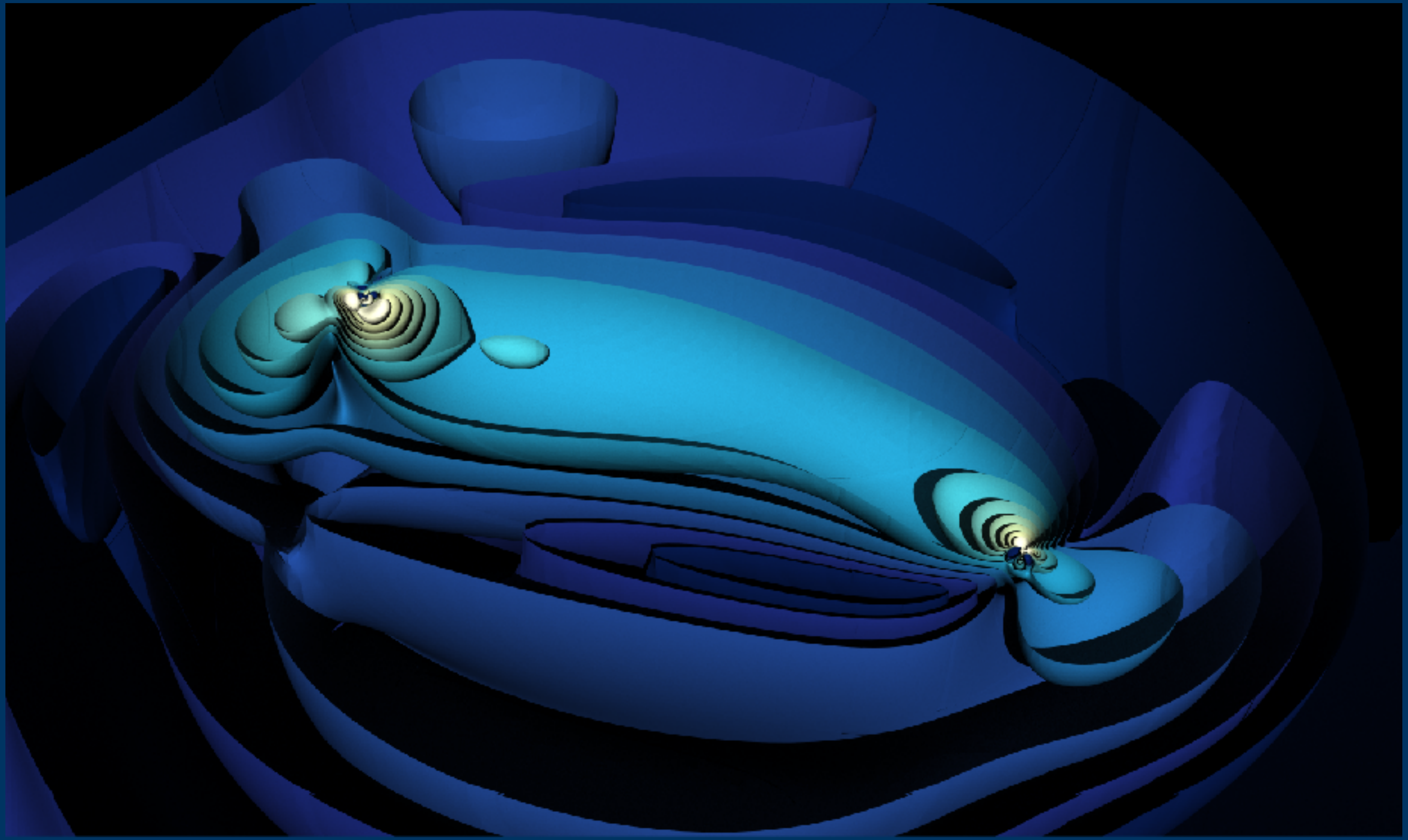


Image credit: Bamber/KC/Cielo

Numerical Relativity and scalar fields

Dr Katy Clough
Ernest Rutherford Fellow
Queen Mary University of London

engrenagem

aka Baby GRChombo

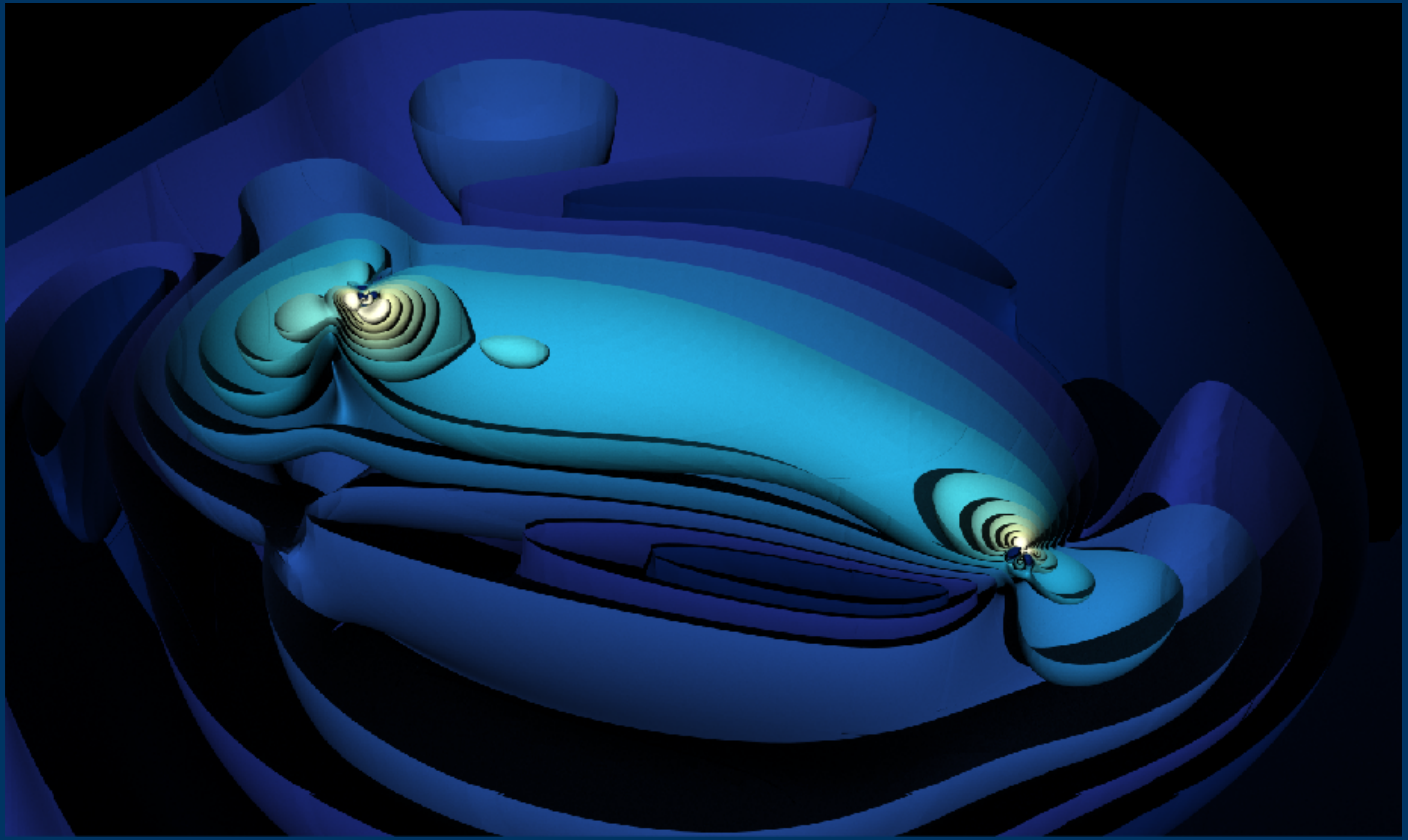
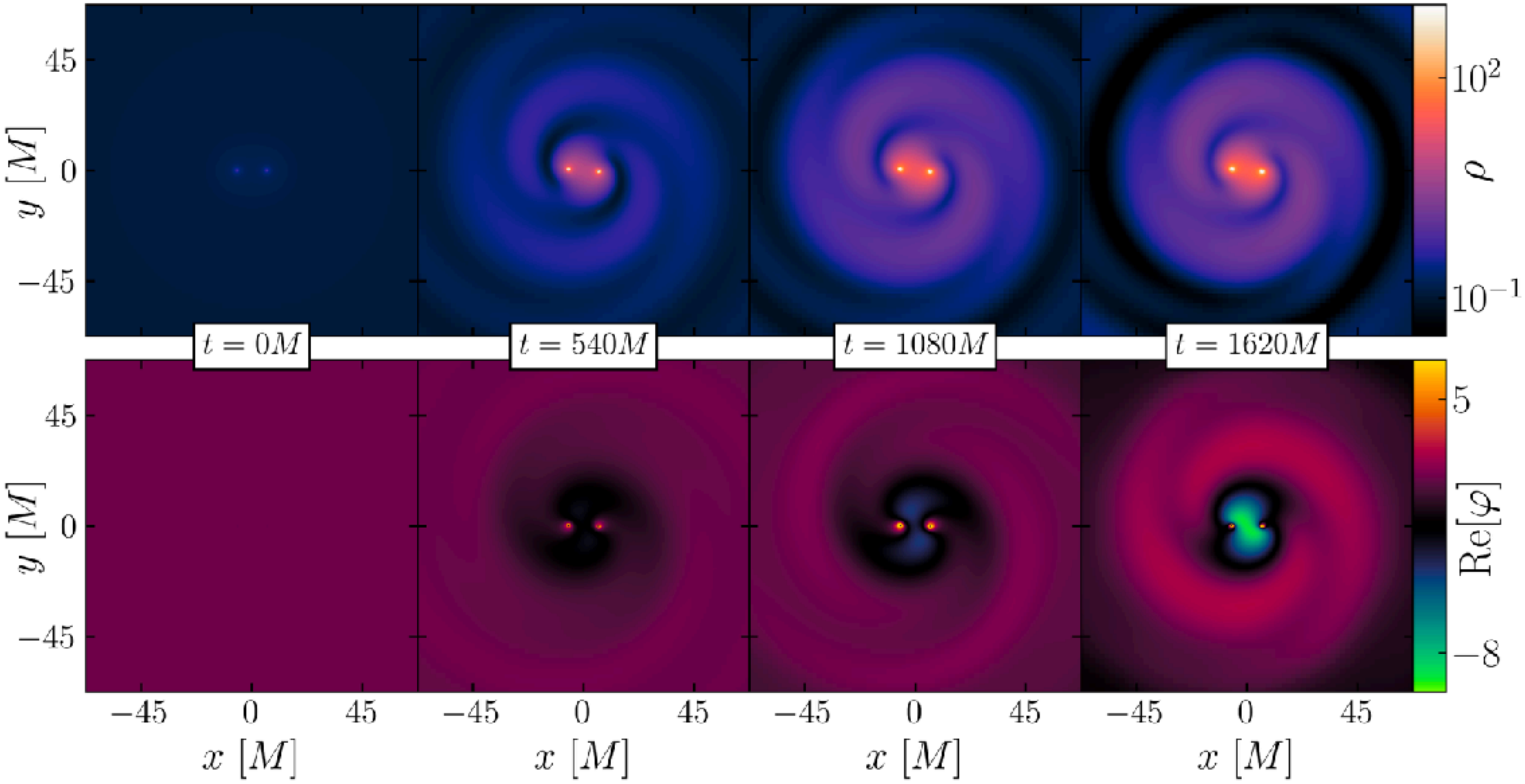


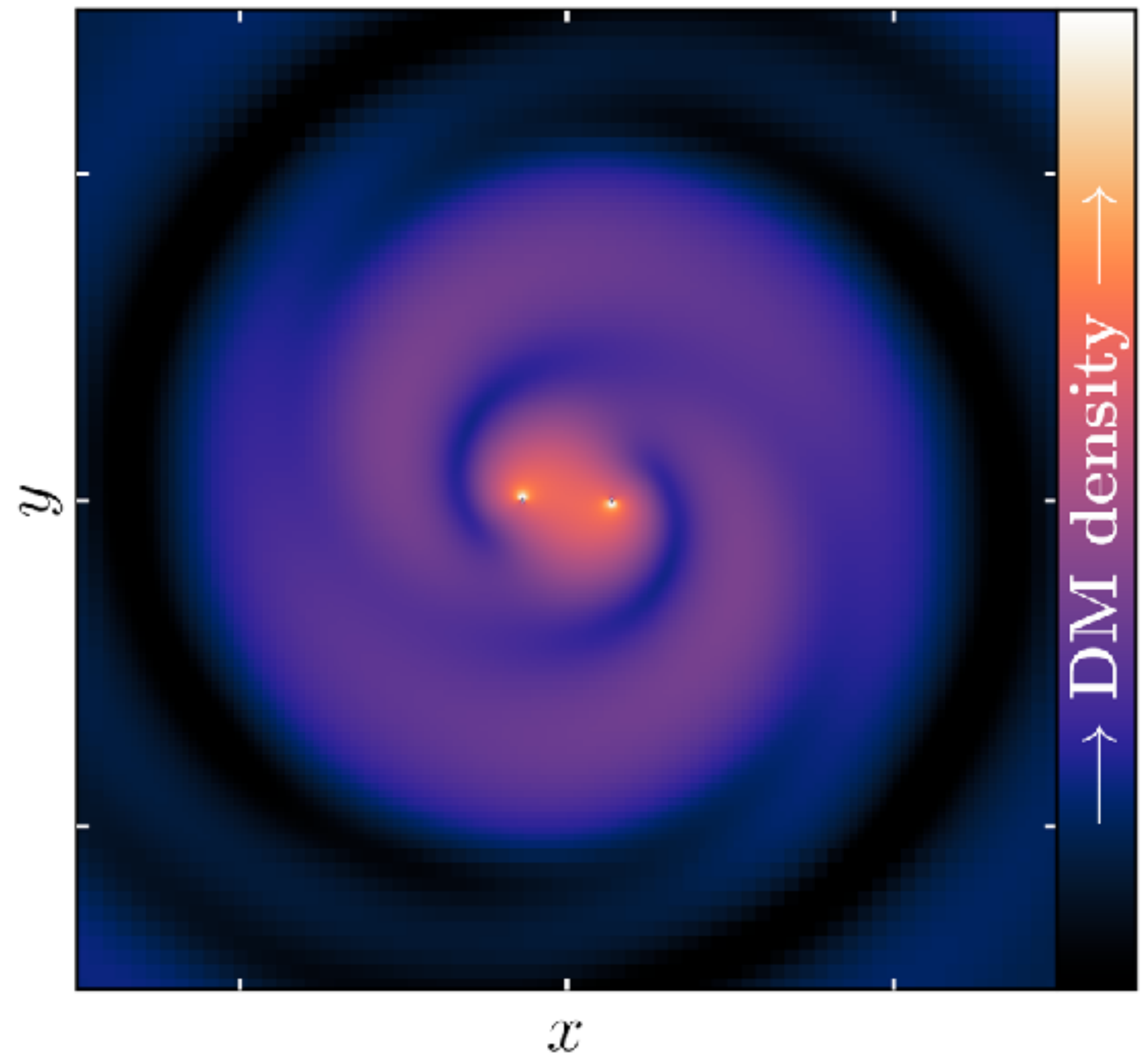
Image credit: Bamber/KC/Cielo

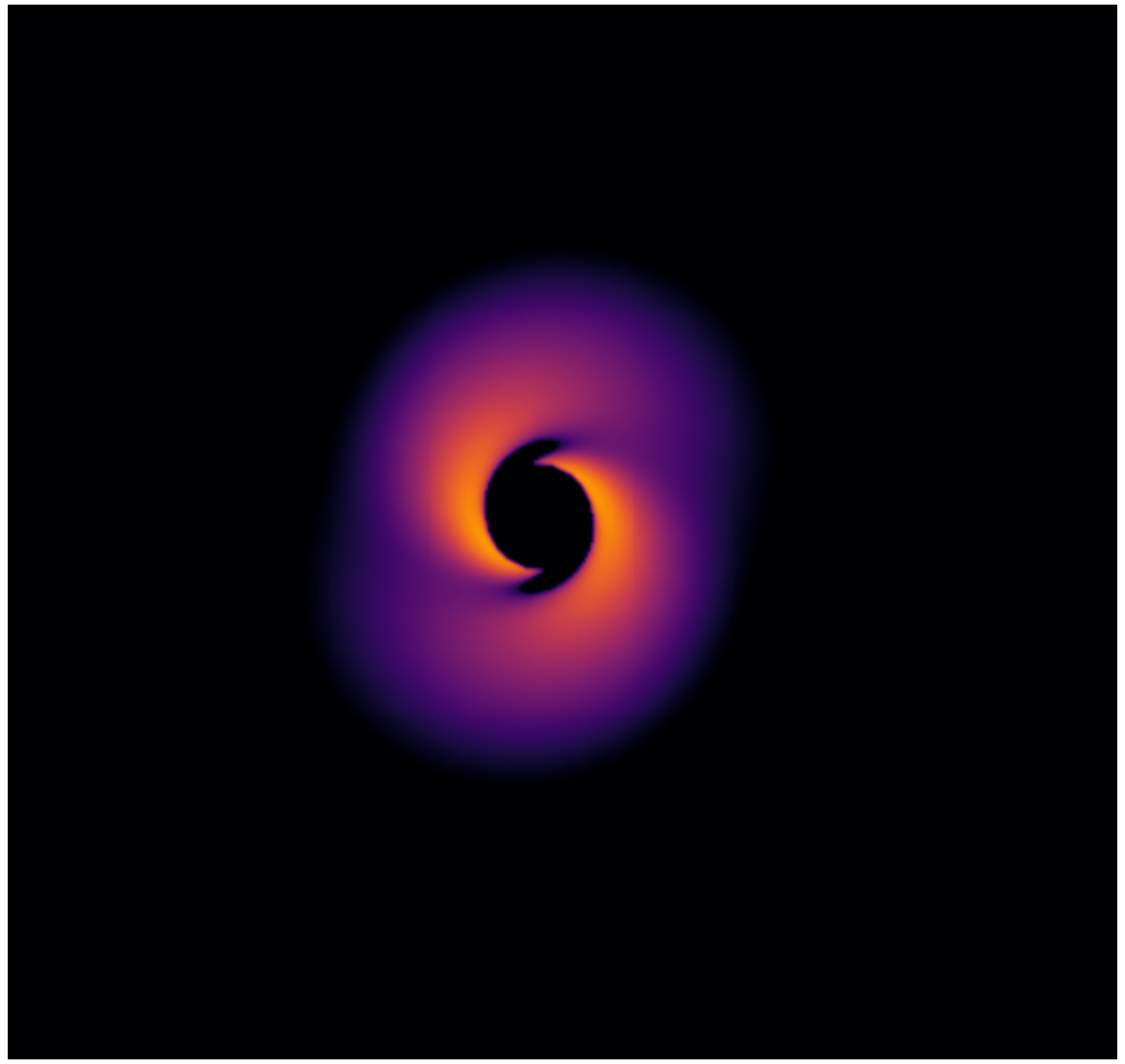
(In principle) GWs can inform us about new fields in BH environments, but dynamics is complex

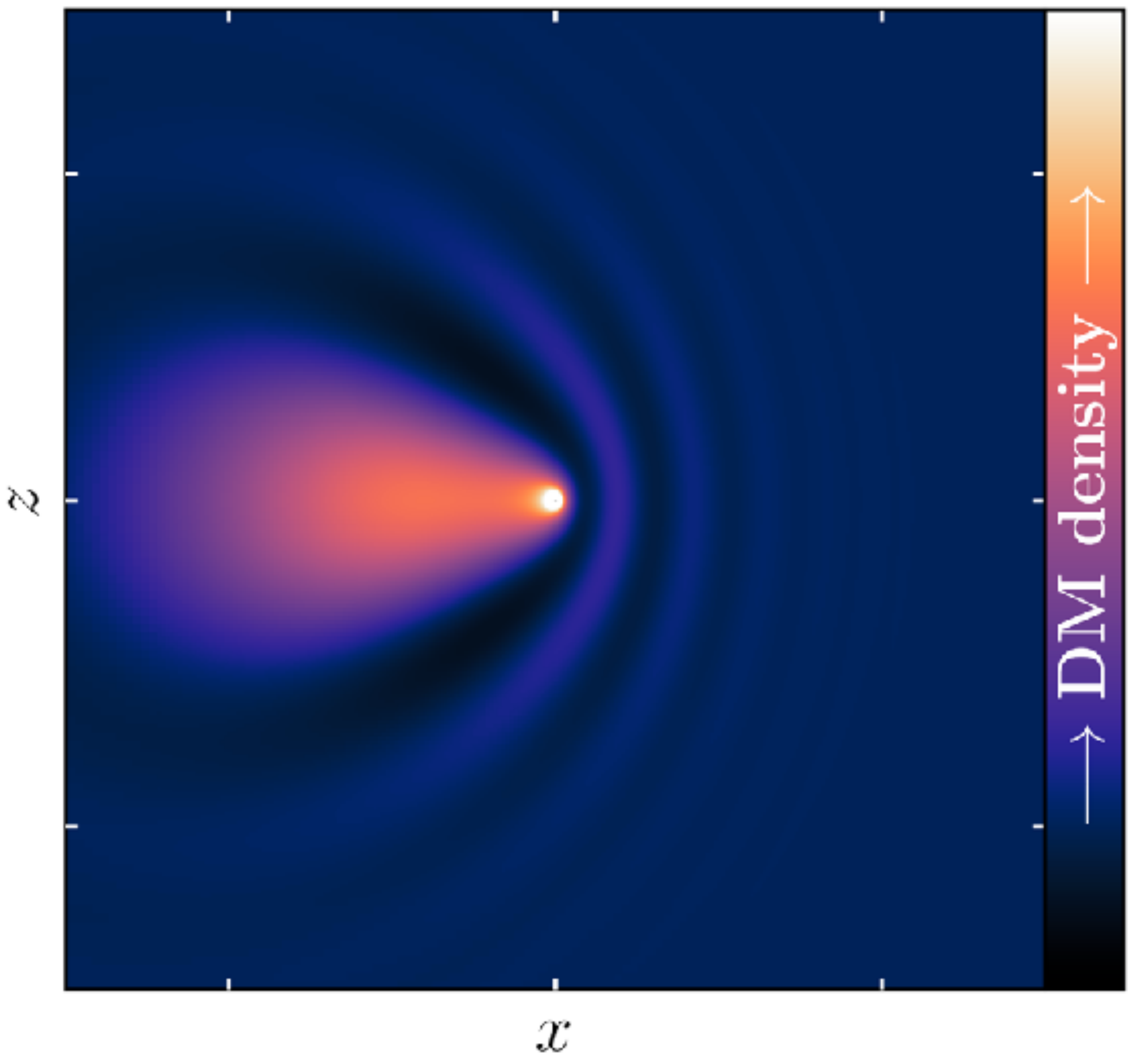


Black hole merger simulations in wave dark matter environments
Jamie Bamber, Josu C. Aurrekoetxea, Katy Clough, Pedro G. Ferreira
Phys.Rev.D 107 (2023) 2, 024035

(In principle) GWs can inform us about new fields in BH environments, but dynamics is complex



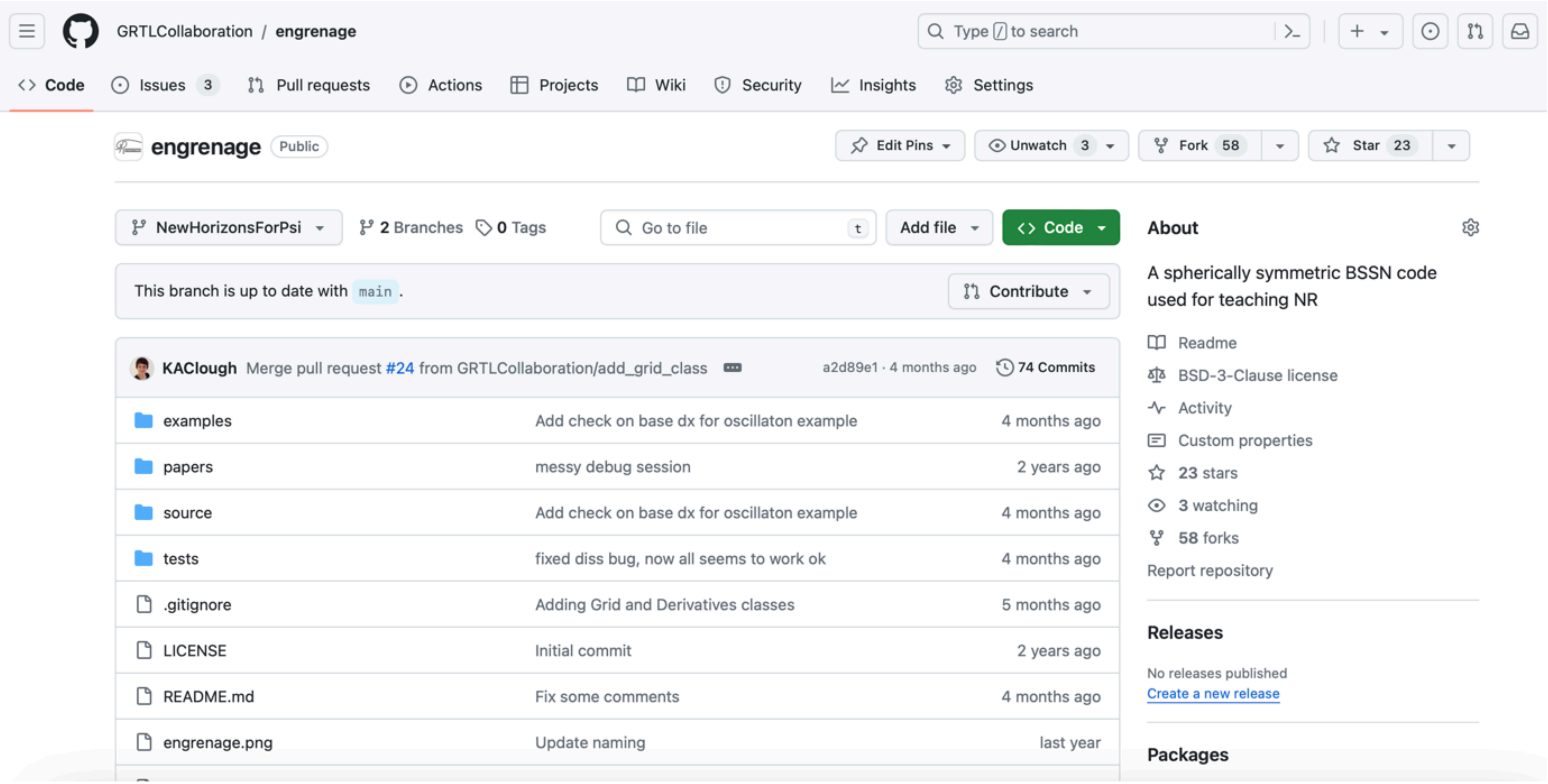




Accretion, superradiance, dynamical friction, scalar-tensor theories, bosonic stars…

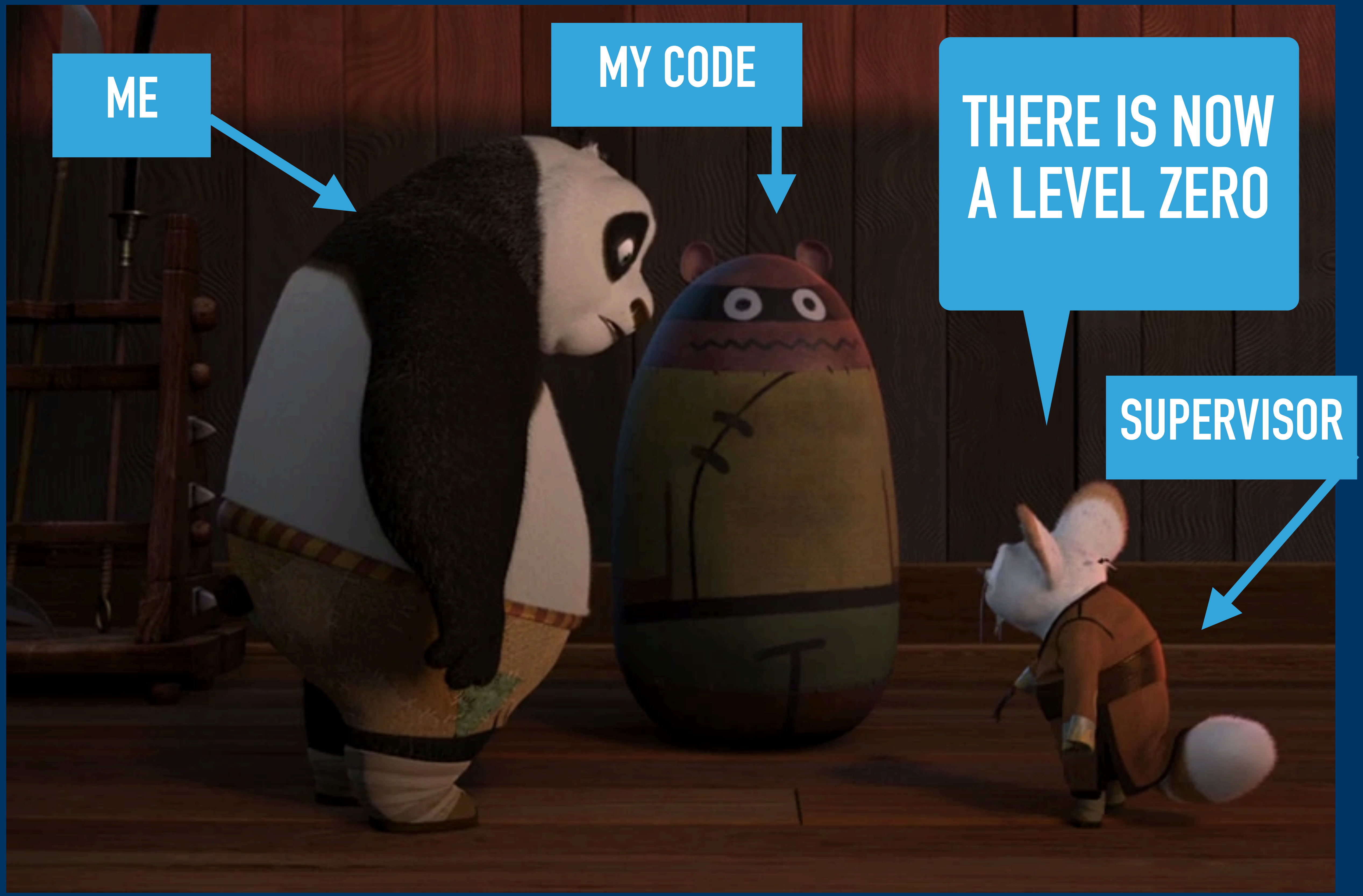
Supercomputers can help

engrenage aka Baby GRChombo is born



https://github.com/GRTLCollaboration/engrenage/tree/NewHorizonsForPsi

We start where we start…



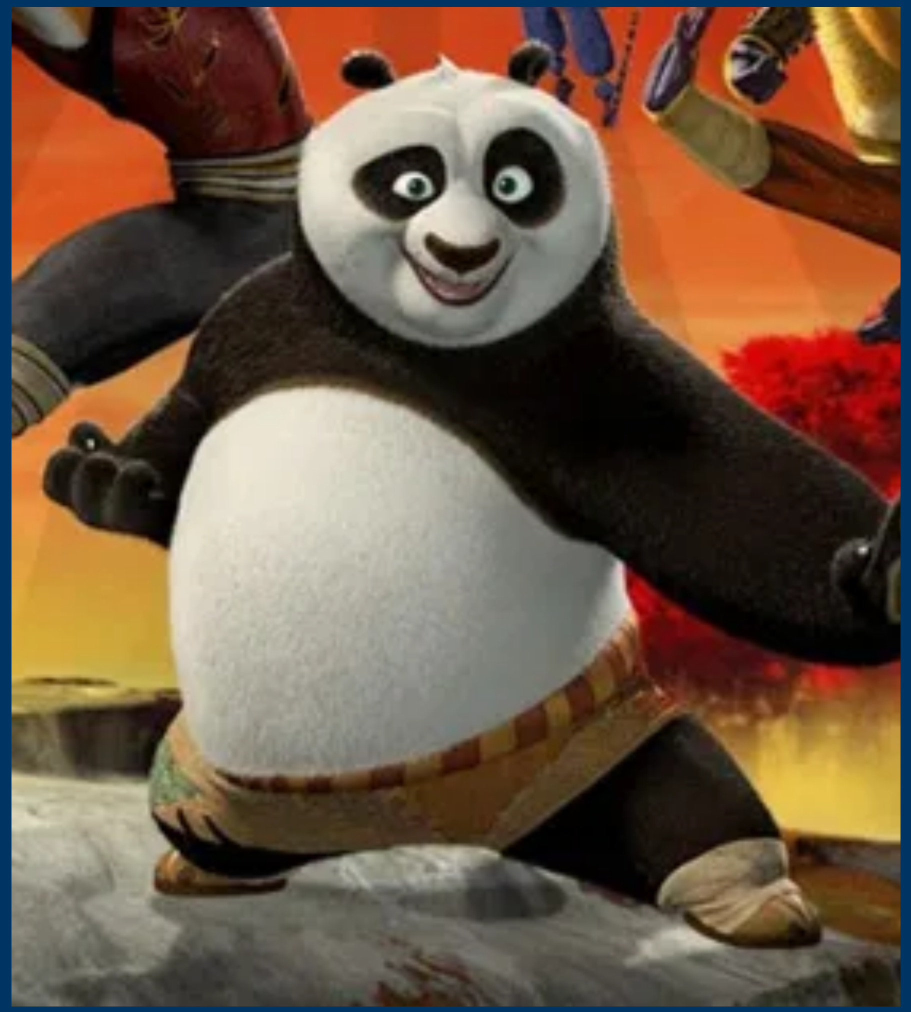
London, England, October 2013…

We start where we start…

Lecture 1: Level zero - theoretical background

- How to solve PDEs on a computer
- Overview of numerical relativity
- The variables of the engrenage code

We start where we start…



Lecture 2: Level one - 4 practical exercises

- Initial conditions - adding the scalar field to a BH
- Modifying equations of motion for the scalar
- Modifying the dynamical gauge for the metric
- Diagnostics - measuring scalar energy fluxes

Lecture 1: Level zero

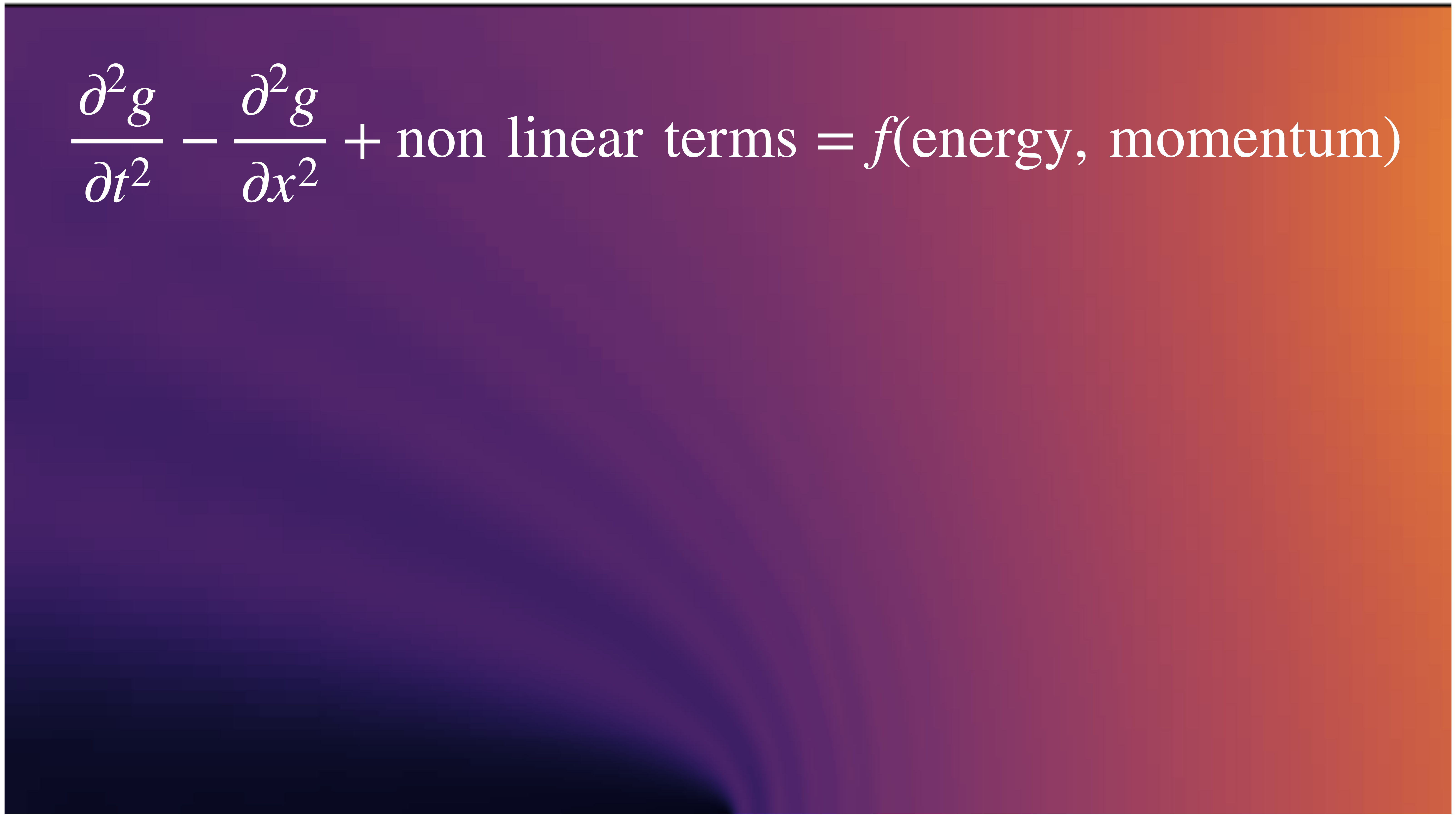
Lecture 1: theoretical background

- How to solve PDEs on a computer
- Overview of numerical relativity
- The variables of the engrenage code

“[Nature] does not care about our mathematical difficulties; [it] integrates [numeri]cally.”

- Albert Einstein (roughly said this)

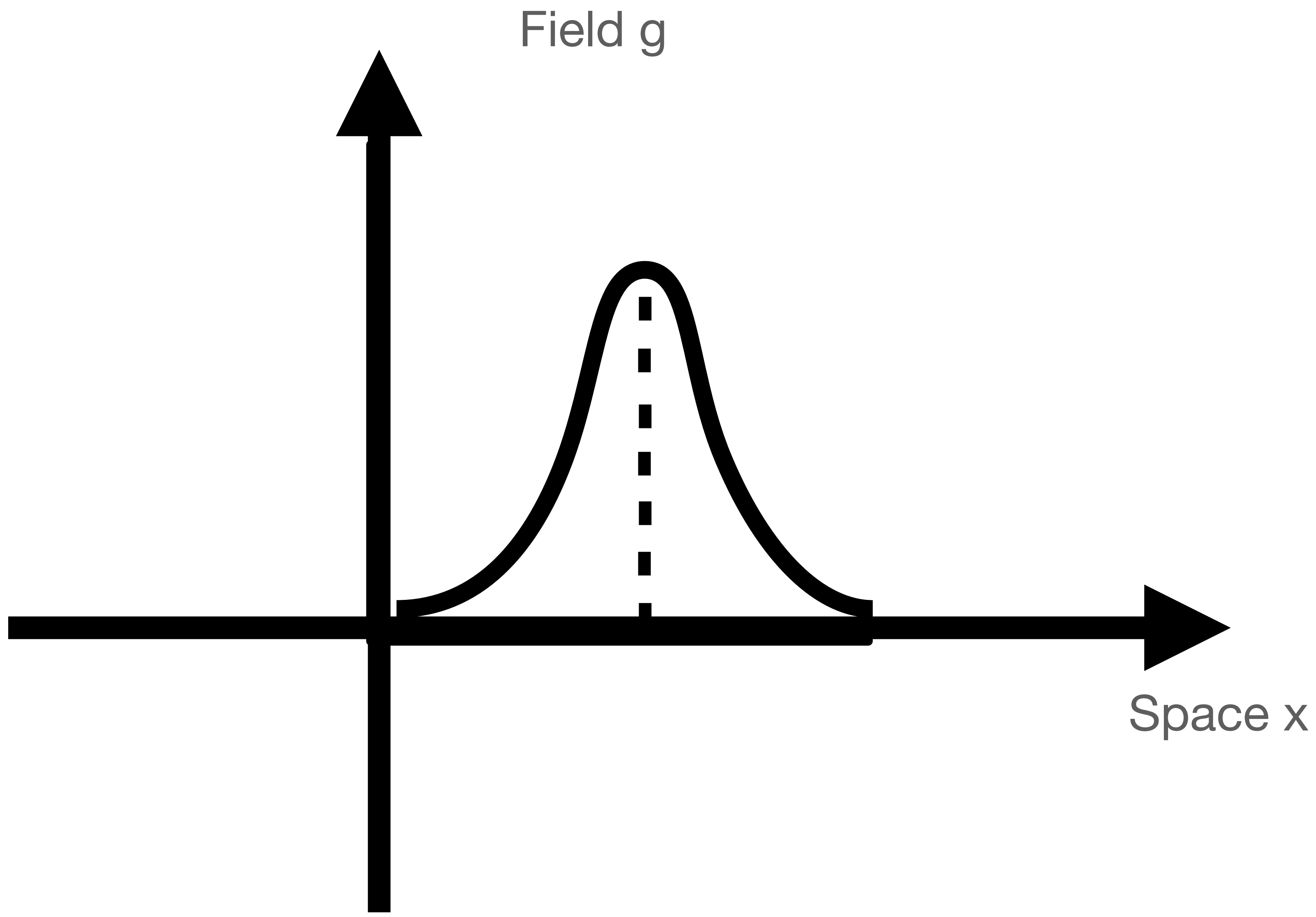
How would Nature solve the wave equation?

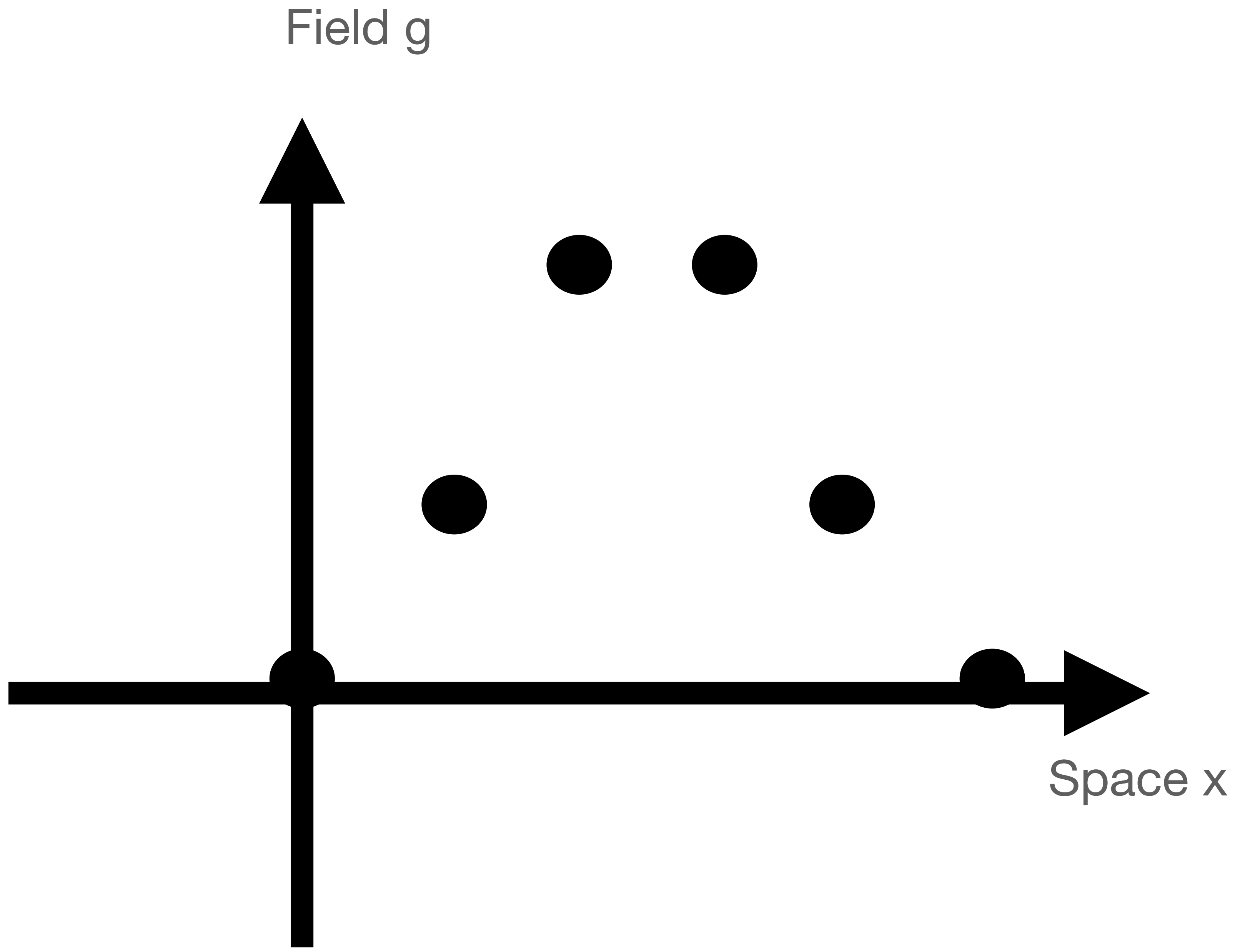


Each event in space “feels” the points around it, and evolves forward in time in response

How do I represent a continuous function on a computer?

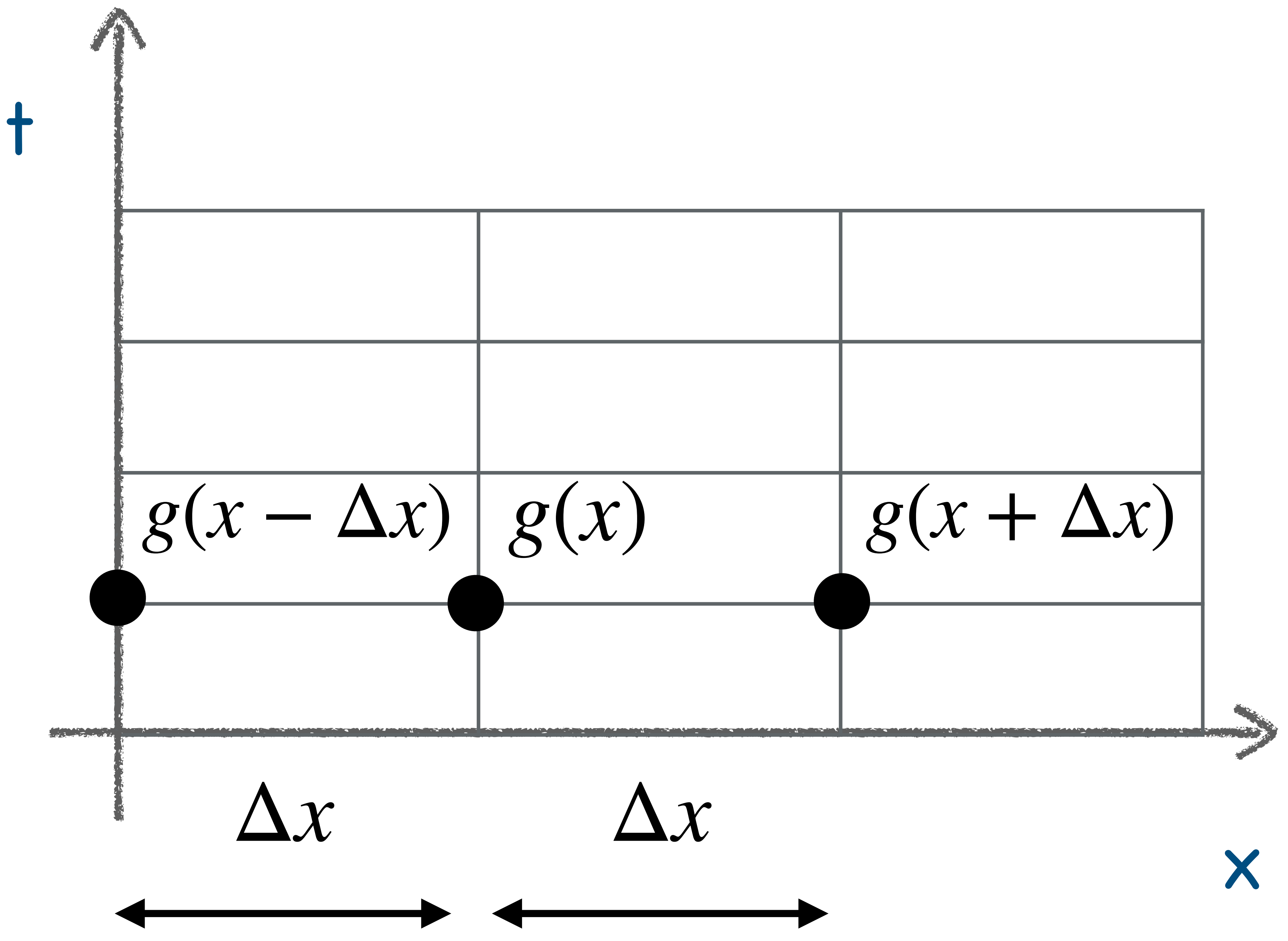
Position x	0	0.5	1	1.5	2	2.5
Field g	0	1	3	3	1	0





How do I find spatial derivatives numerically?

$$\frac{\partial^2 g}{\partial t^2} - \frac{\partial^2 g}{\partial x^2} = \text{Source}$$



$$\frac{\partial^2 g}{\partial x^2} \approx \frac{\frac{g(x+\Delta x) - g(x)}{\Delta x} - \frac{g(x) - g(x-\Delta x)}{\Delta x}}{\Delta x}$$

This approach is called finite differencing

We can see it as the convolution of ***a stencil*** with the ***current state vector***.

$\Delta x = 0.5$

Position x	0	0.5	1	1.5	2	2.5
Field g	0	1	3	2	1	0

↑

-1	0	1

First derivative stencil

$$\frac{\partial g}{\partial x} \approx \frac{g(x+\Delta x) - g(x-\Delta x)}{2\Delta x}$$

dg/dx						

This approach is called finite differencing

We can see it as the convolution of ***a stencil*** with the ***current state vector***.

Position x	0	0.5	1	1.5	2	2.5
Field g	0	1	3	2	1	0

↑

-1	0	1

First derivative stencil

$$\frac{\partial g}{\partial x} \approx \frac{g(x+\Delta x) - g(x-\Delta x)}{2\Delta x}$$

dg/dx			**1**			

This approach is called finite differencing

We can see it as the convolution of ***a stencil*** with the ***current state vector***.

Position x	0	0.5	1	1.5	2	2.5
Field g	0	1	3	2	1	0

↑

-1	0	1

First derivative stencil

$$\frac{\partial g}{\partial x} \approx \frac{g(x + \Delta x) - g(x - \Delta x)}{2\Delta x}$$

dg/dx			**1**			

This approach is called finite differencing

We can see it as the convolution of ***a stencil*** with the ***current state vector***.

Position x	0	0.5	1	1.5	2	2.5
Field g	0	1	3	2	1	0

↑

-1	0	1

First derivative stencil

$$\frac{\partial g}{\partial x} \approx \frac{g(x+\Delta x) - g(x-\Delta x)}{2\Delta x}$$

dg/dx		**3**	**1**			

This approach is called finite differencing

We can see it as the convolution of ***a stencil*** with the ***current state vector***.

Position x	0	0.5	1	1.5	2	2.5
Field g	0	1	3	2	1	0

-1	0	1

dg/dx		**3**	**1**	**-2**		

This approach is called finite differencing

We can see it as the convolution of ***a stencil*** with the ***current state vector***.

Position x	0	0.5	1	1.5	2	2.5
Field g	0	1	3	2	1	0

-1	0	1

dg/dx		**3**	**1**	**-2**	**-2**	

What about the end points?

This approach is called finite differencing

We can see it as the convolution of ***a stencil*** with the ***current state vector***.

Position x	0	0.5	1	1.5	2	2.5
Field g	0	1	3	2	1	0

Use one sided stencil - doesn't have to be centralised

-2	2

dg/dx		3	1	-2	-2	-2

This approach is called finite differencing

We can see it as the convolution of ***a stencil*** with the ***current state vector***.

Position x	0	0.5	1	1.5	2	2.5
Field g	0	1	3	2	1	0

*OR use a **boundary condition** -*
some knowledge about the function
- e.g. maybe its derivative goes to zero here

↓

dg/dx		**3**	**1**	**-2**	**-2**	**0**

Finite differencing - matrix representation

We can also represent this convolution in matrix form:

Position x	0	0.5	1	1.5	2	2.5
Field g	0	1	3	2	1	0

dg/dx = ***Matrix D*** ***g***

dg/dx:

2
3
1
-2
-2
-2

=

Matrix D:

-2	2				
-1	0	1			
	-1	0	1		
		-1	0	1	
			-1	0	1
				-2	2

•

g:

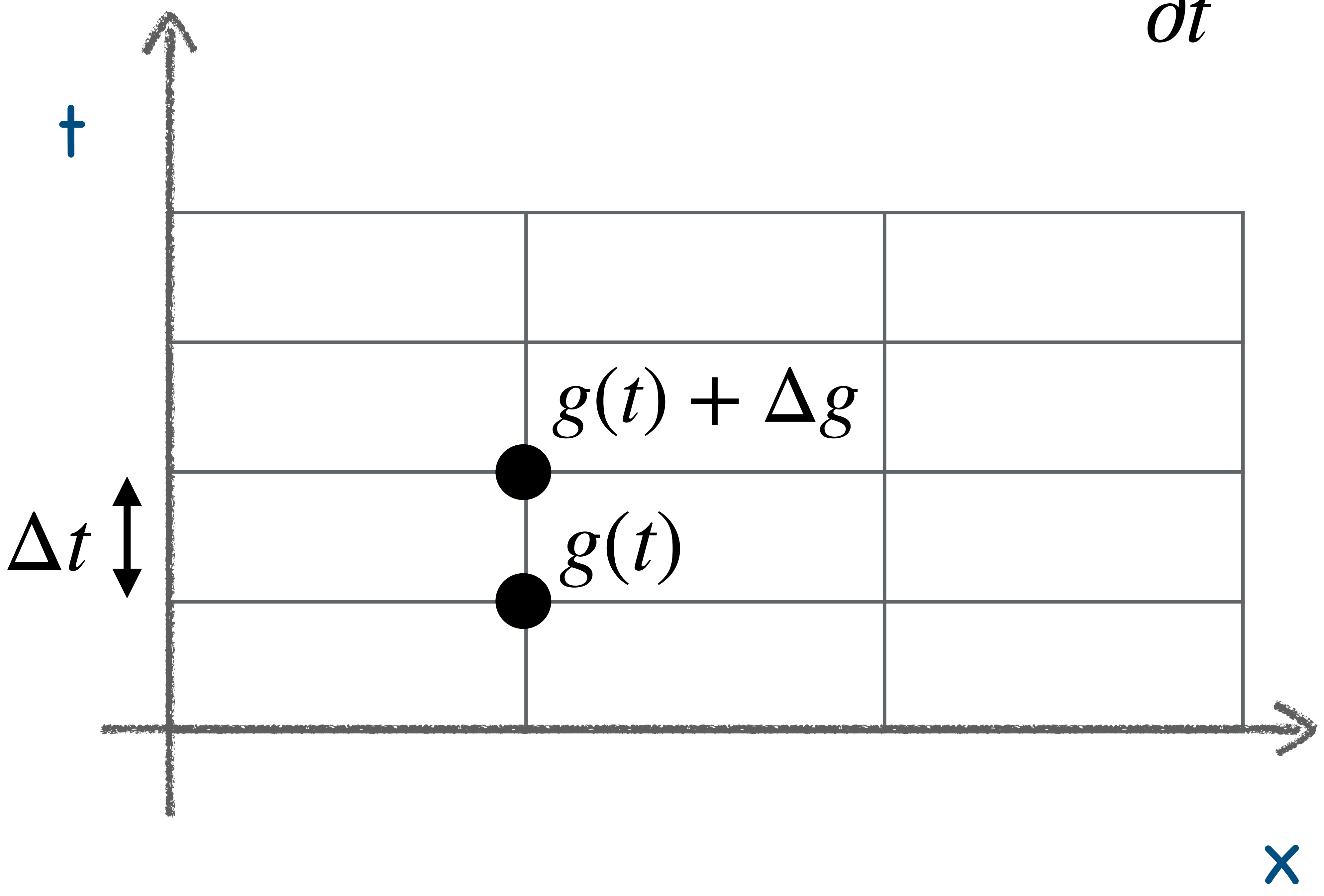
0
1
3
2
1
0

All blank entries zero

How do I integrate time derivatives numerically?

$$\frac{\partial^2 g}{\partial t^2} - \frac{\partial^2 g}{\partial x^2} = \text{Source}$$

$$\begin{cases} \dfrac{\partial K}{\partial t} = \dfrac{\partial^2 g}{\partial x^2} + \text{Source} \\ \dfrac{\partial g}{\partial t} = K \end{cases}$$



$$\Delta K = \Delta t \left(\frac{\partial^2 g}{\partial x^2} + \text{Source} \right)$$

$$\Delta g = K \, \Delta t$$

Matrix implementation of time evolution

Position x	0	0.5	1	1.5	2	2.5
Field g	0	1	3	2	1	0
Field K	0	2	1	1	1	0

dKdt* = *Matrix D^2* • *g* + *Source

$$\begin{pmatrix} 2 \\ 3 \\ 1 \\ -2 \\ -2 \\ -2 \end{pmatrix} = \begin{pmatrix} X & X & & & & \\ X & X & X & & & \\ & X & X & X & & \\ & & X & X & X & \\ & & & X & X & X \\ & & & & X & X \end{pmatrix} \bullet \begin{pmatrix} 0 \\ 1 \\ 3 \\ 2 \\ 1 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 3 \\ 4 \\ 5 \\ 7 \\ 0 \end{pmatrix}$$

Matrix implementation of time evolution

Position x	0	0.5	1	1.5	2	2.5
Field g	0	1	3	2	1	0
Field K	0	2	1	1	1	0

dgdt **=** ***K***

dgdt
0
2
1
1
1
0

=

K
0
2
1
1
1
0

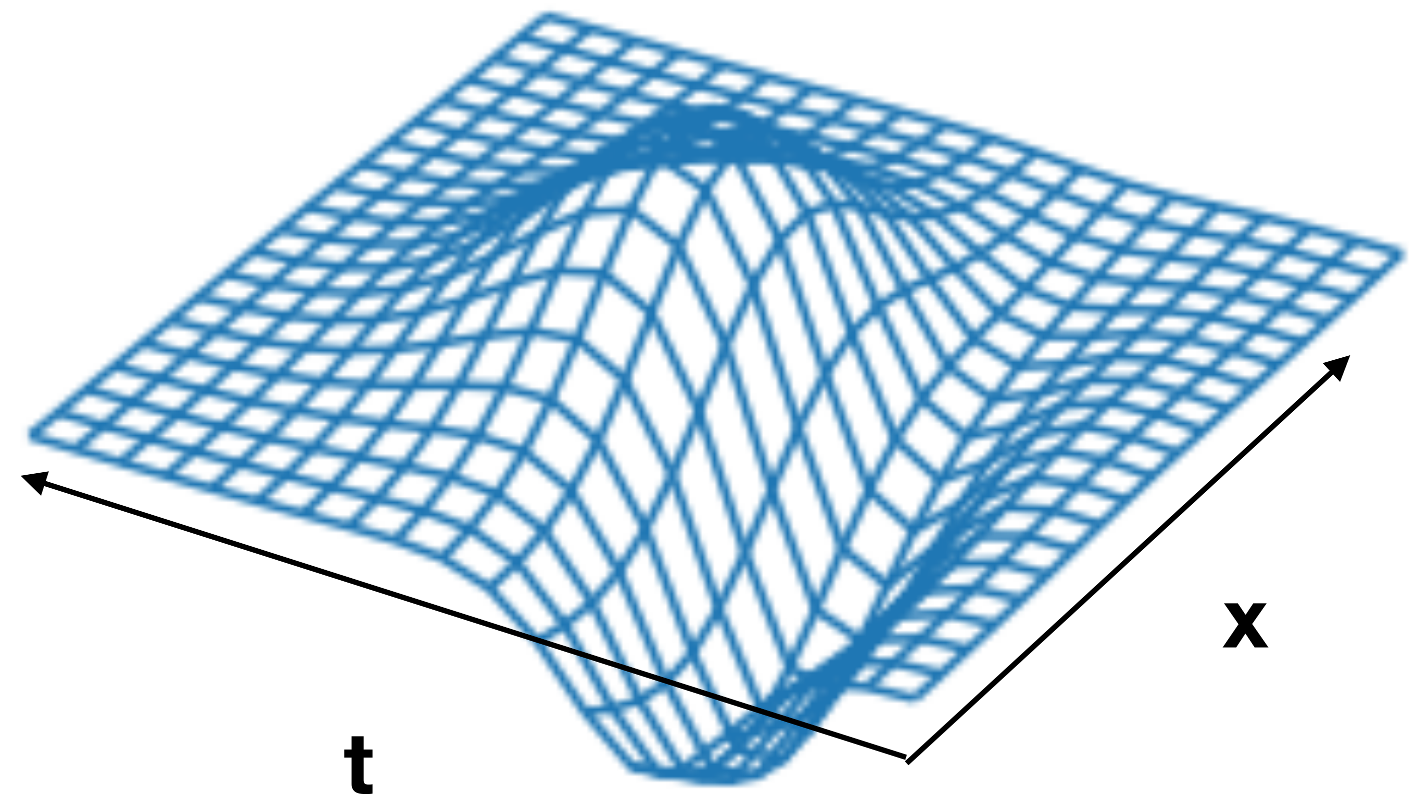
GR & NR 101

$$R_{ab} - R/2\, g_{ab} = 8\pi\, T_{ab}$$

Curved spacetime

$$ds^2 = f(x,t)\ dt^2 + g(x,t)\ dx^2 +$$

$$2\ h(x,t)\ dt\ dx$$



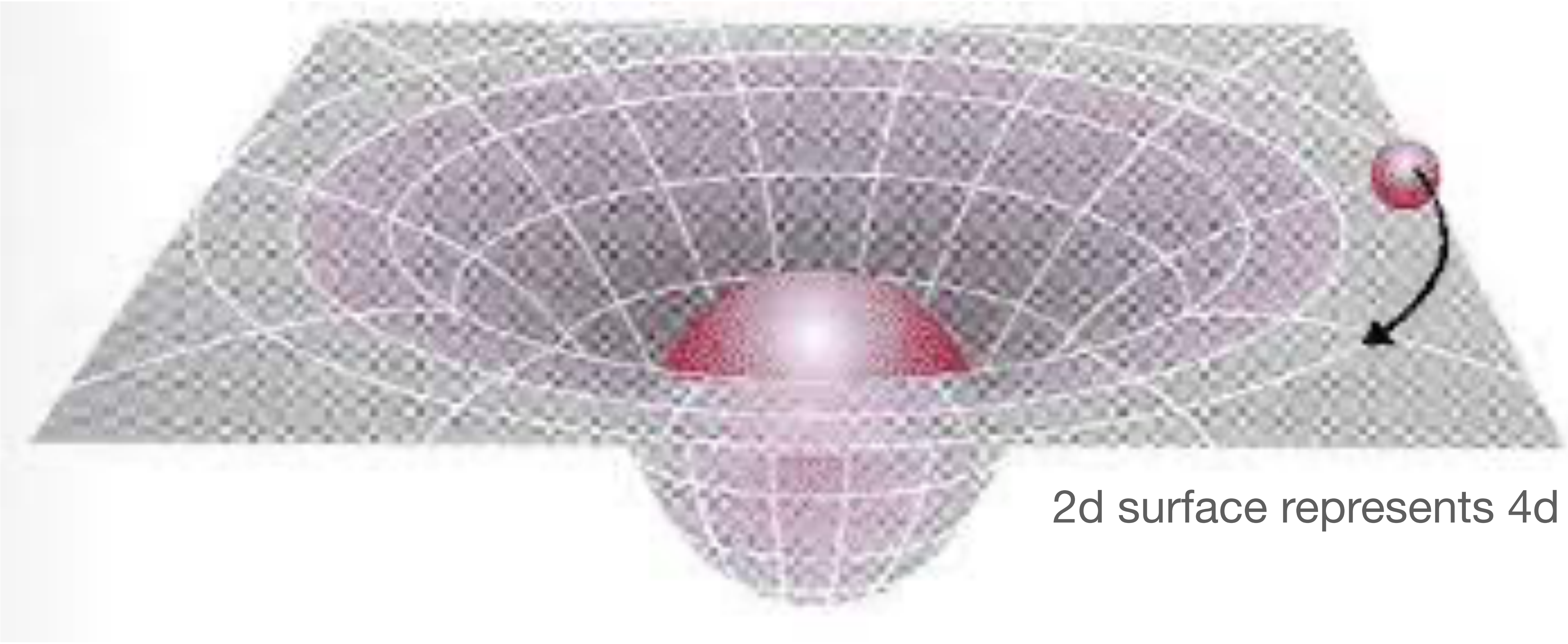
Curved spacetime

$$ds^2 = \begin{pmatrix} dt & dx & dy & dz \end{pmatrix} \begin{pmatrix} g_{00} & g_{01} & g_{02} & g_{03} \\ g_{10} & g_{11} & g_{12} & g_{13} \\ g_{20} & g_{21} & g_{22} & g_{23} \\ g_{30} & g_{31} & g_{32} & g_{33} \end{pmatrix} \begin{pmatrix} dt \\ dx \\ dy \\ dz \end{pmatrix}$$

"The spacetime metric"

$$g_{ab}(t, \vec{x})$$

The Einstein equation tells us how the metric should look, given some energy/matter distribution

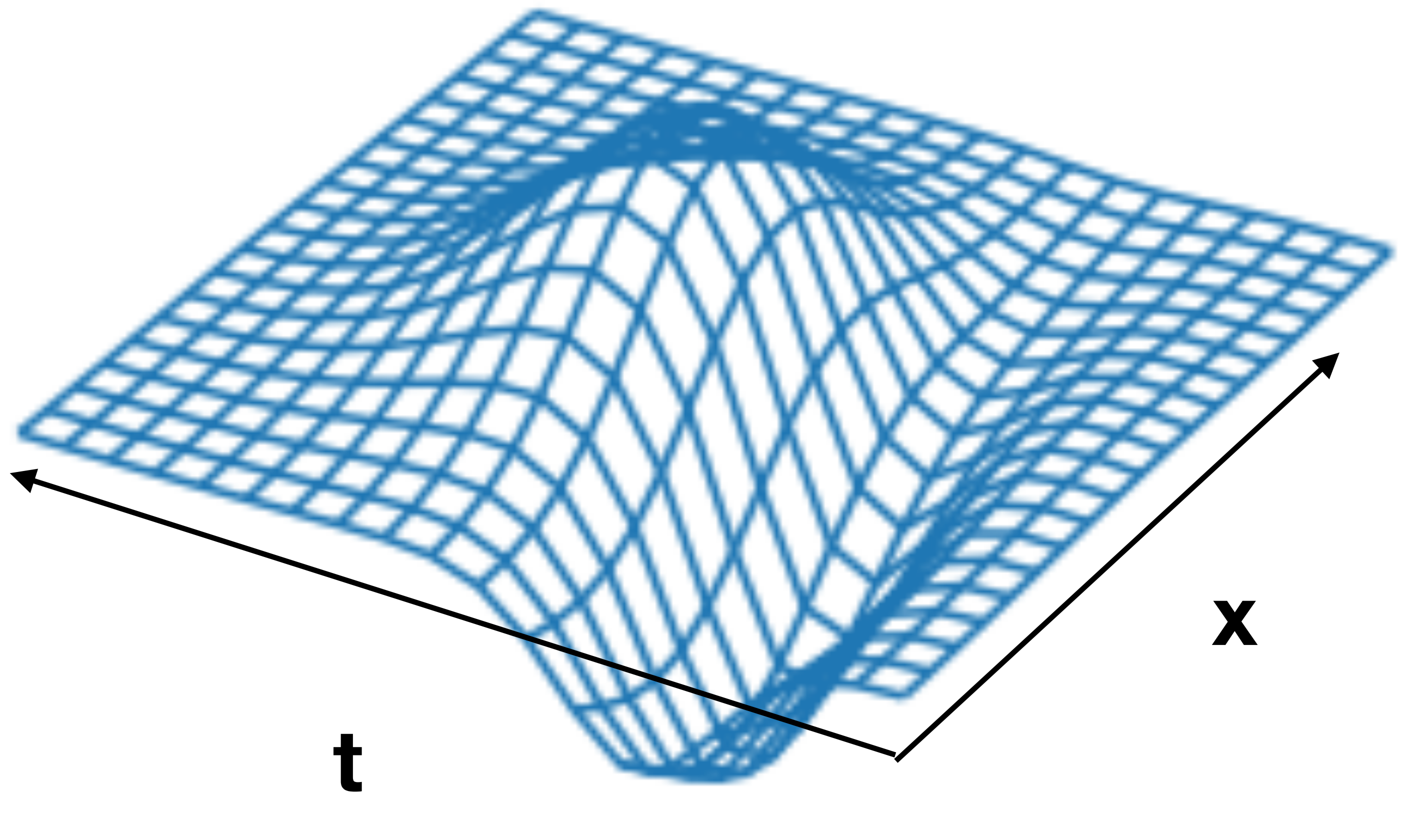


2d surface represents 4d spacetime

$$R_{ab} - R/2\, g_{ab} = 8\pi\, T_{ab}$$

"Matter tells spacetime how to curve…"

The Einstein equation tells us how the metric should look, given some energy/matter distribution



$R_{ab} - R/2\, g_{ab} = 8\pi\, T_{ab}$

4 constraint equations for any time slice - non linear elliptic/Poisson equation

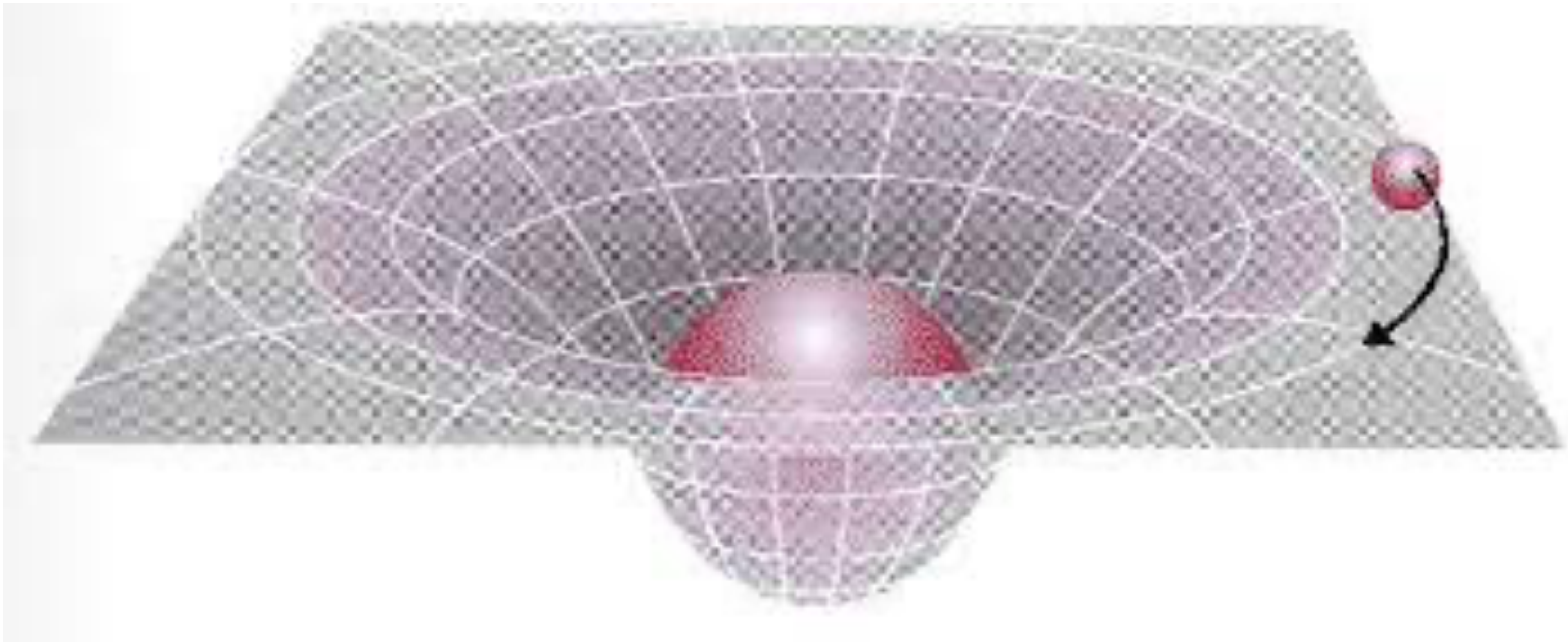
$$\frac{\partial^2 g}{\partial x^2} + \text{non linear terms} = f(\text{energy, momentum})$$

An evolution equation for all time - non linear hyperbolic/wave equation

$$\frac{\partial^2 g}{\partial t^2} - \frac{\partial^2 g}{\partial x^2} + \text{non linear terms} = f(\text{energy, momentum})$$

"Matter tells spacetime how to curve..."

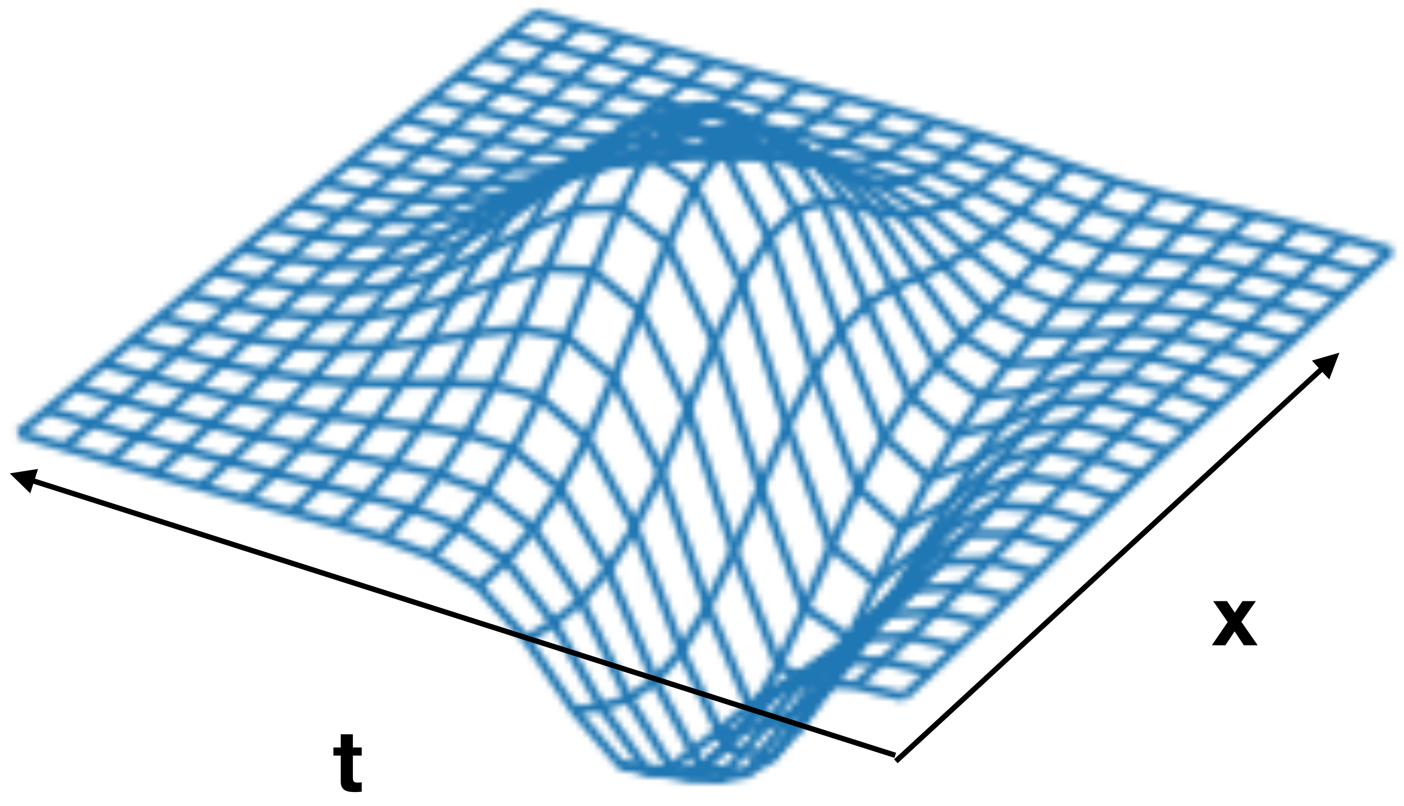
The metric determines the motion of matter



$$R_{ab} - R/2\ g_{ab} = 8\pi\ T_{ab}$$

"...spacetime tells matter how to move."

The metric determines the motion of matter



$R_{ab} - R/2\ g_{ab} = 8\pi\ T_{ab}$

Klein Gordon equation for the scalar field u

$$g^{\mu\nu} \nabla_\mu \nabla_\nu u = \frac{dV}{du}$$

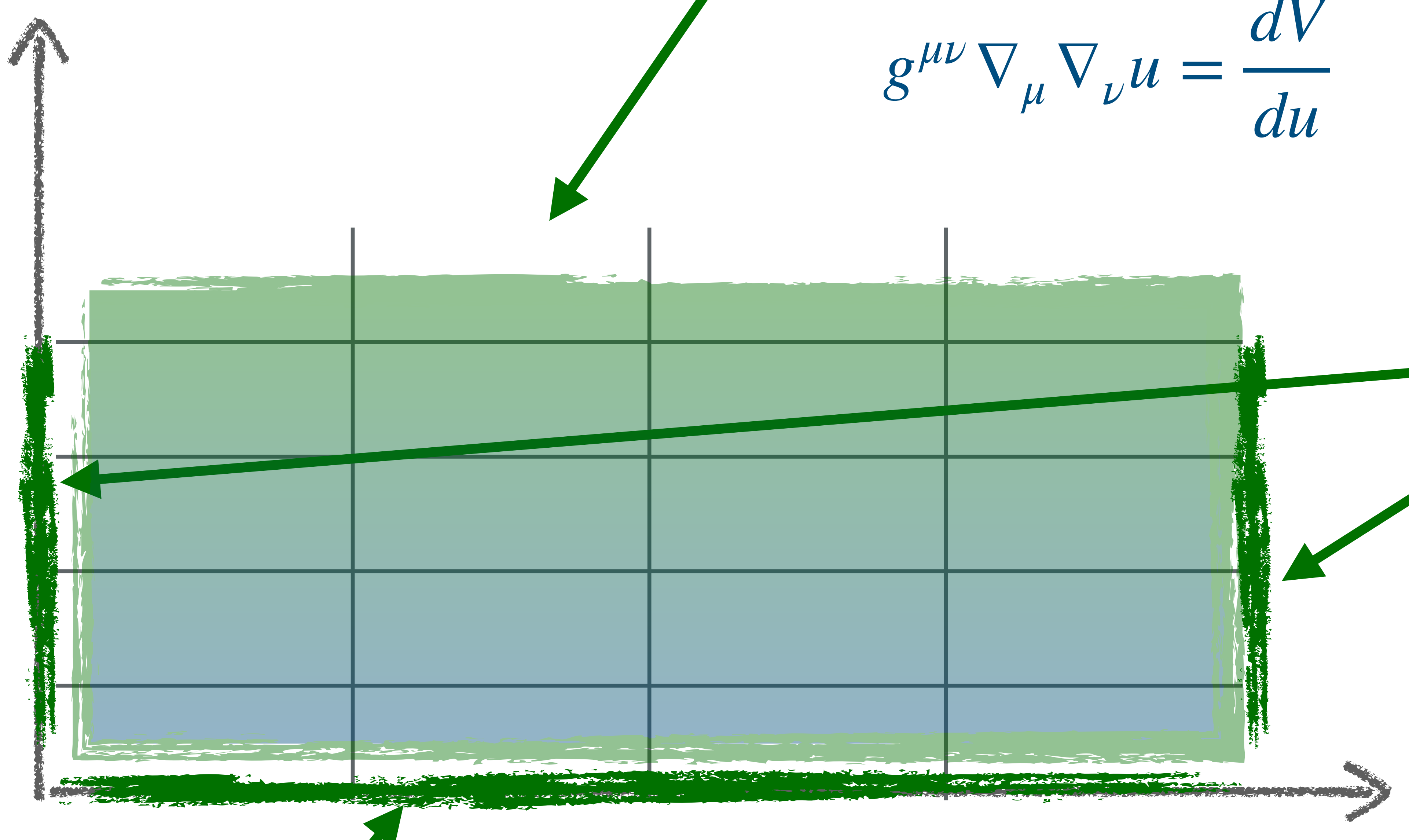
"...spacetime tells matter how to move."

Numerical relativity

Fill using Einstein equation and continuity for matter

$$\frac{\partial^2 g}{\partial t^2} - \frac{\partial^2 g}{\partial x^2} + \text{non linear terms} = f(\text{energy, momentum})$$

$$g^{\mu\nu} \nabla_\mu \nabla_\nu u = \frac{dV}{du}$$



boundary conditions $(\partial_{xx} g_{ab}, \partial_x g_{ab},\ g_{ab},\ T_{ab})$

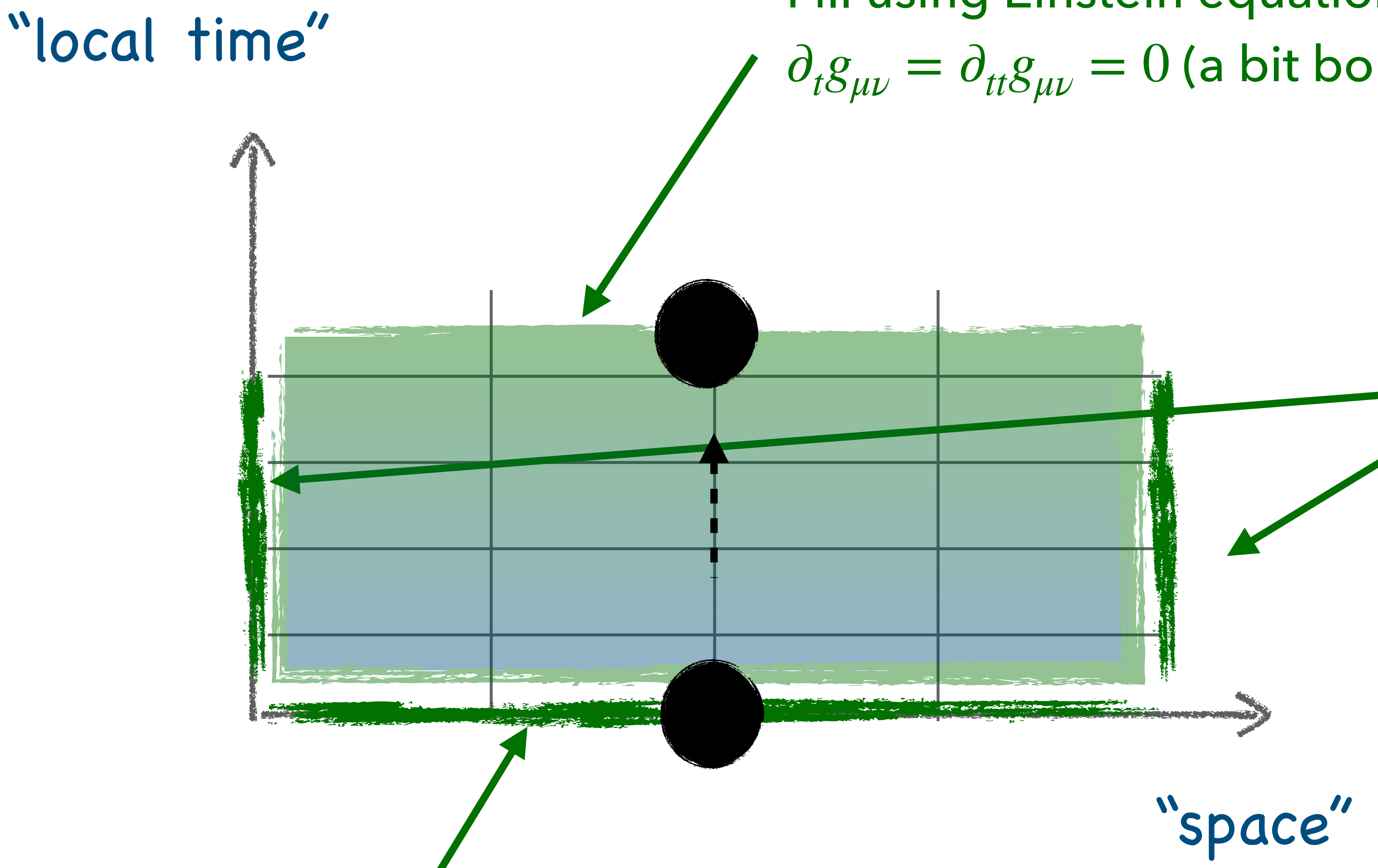
initial data $(\partial_t g_{ab},\ g_{ab},\ T_{ab})$ satisfying

$$\frac{\partial^2 g}{\partial x^2} + \text{non linear terms} = f(\text{energy, momentum})$$

Numerical relativity

Fill using Einstein equation (classical black holes are stable)

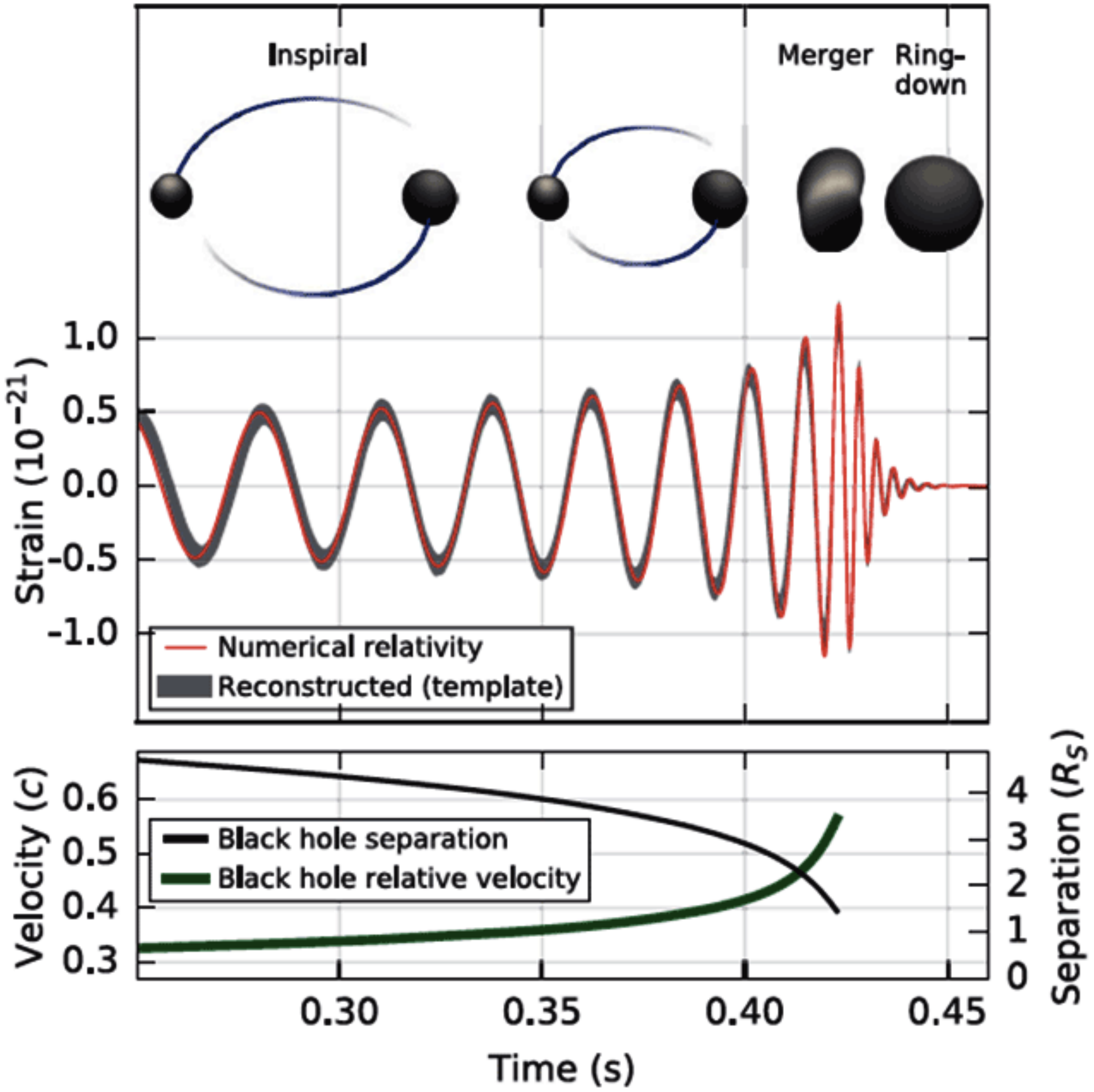
$\partial_t g_{\mu\nu} = \partial_{tt} g_{\mu\nu} = 0$ (a bit boring!)



boundary conditions

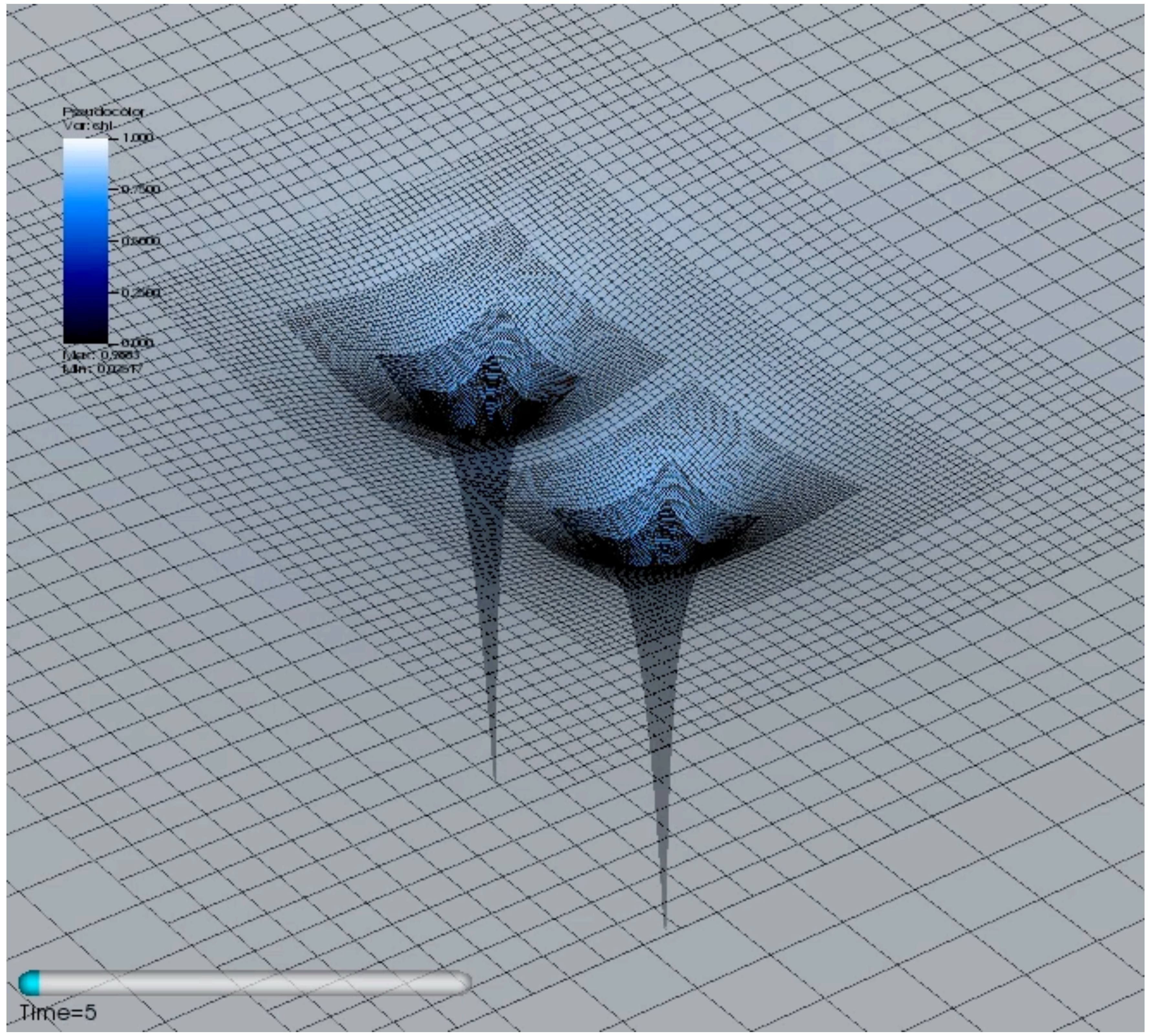
= asymptotically flat space

initial data = a black hole , no other matter



GW150914

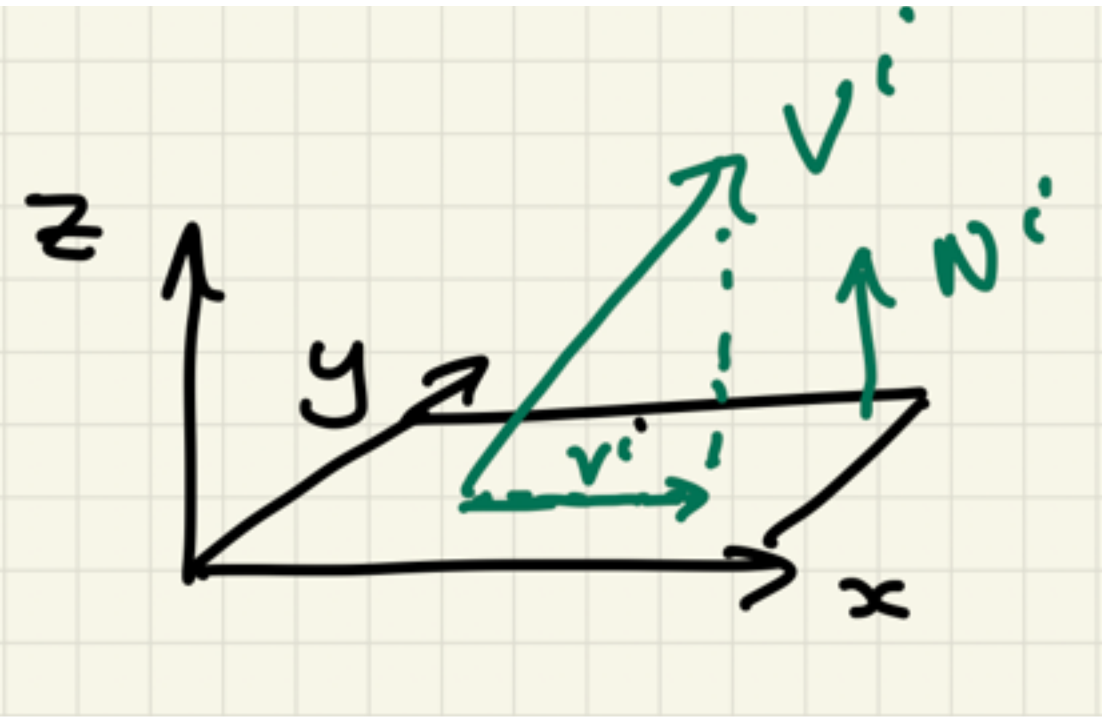
t=14 September 2015, x = LIGO, Earth



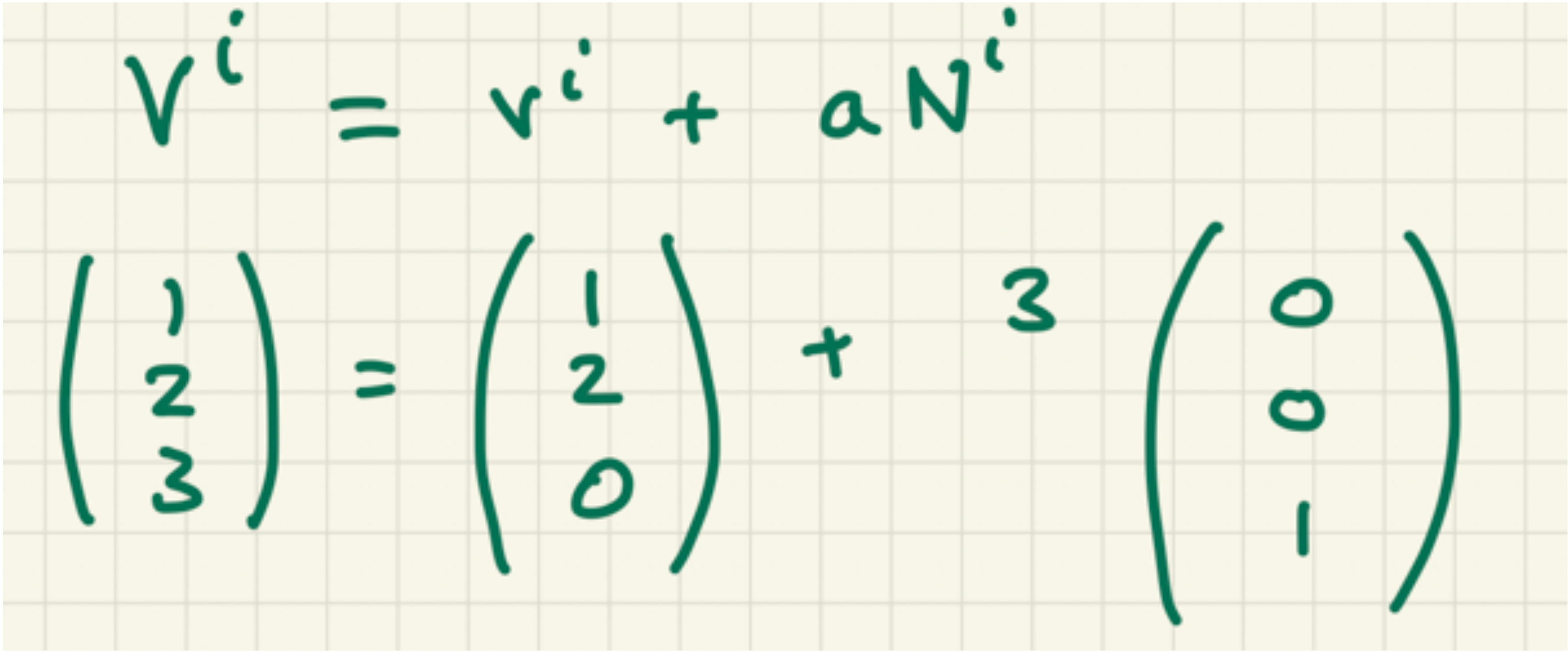
(Roughly) $\dfrac{1}{det(g_{ab})}$

ADM decomposition, in theory and in practise

What is the ADM decomposition?

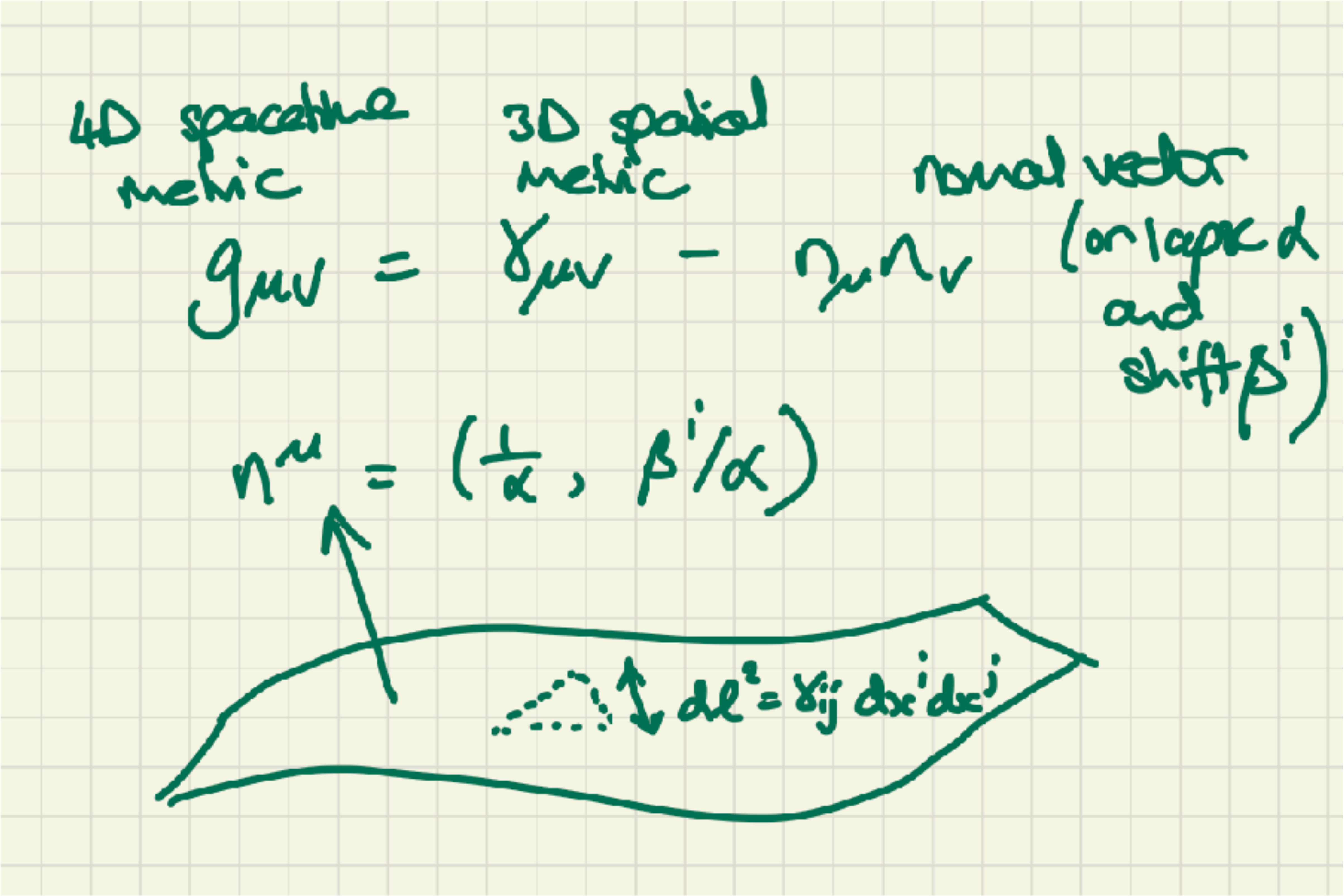


We can decompose a vector into the part that lies in a surface and a part normal to the surface



What is the ADM decomposition?

We can decompose the 4D spacetime metric into the part that lies in a 3D spatial hypersurface and a part normal to the 3D spatial hypersurface



We can also decompose the Einstein equations themselves into the part that lies in the surface and the part normal to the surface

$$n^\mu n^\nu (G_{\mu\nu} - 8\pi T_{\mu\nu}) \implies \qquad \mathscr{H} \equiv {}^{(3)}R + K^2 + K_{ij}K^{ij} - 16\pi\rho = 0$$

$$P^\mu_i n^\mu (G_{\mu\nu} - 8\pi T_{\mu\nu}) \implies \qquad \mathscr{M}_i \equiv D_j K^j{}_i - D_i K - 8\pi S_i = 0$$

$$P^\mu_i P^\nu_j (G_{\mu\nu} - 8\pi T_{\mu\nu}) \implies \qquad \partial_t K_{ij} = f(\alpha, \beta^i, \gamma^{ij}, K_{ij}, \partial_i(\text{variables}), \text{matter})$$

Where we defined $\partial_t \gamma_{ij} = -2\alpha K_{ij} + D_i\beta_j + D_j\beta_i$

What is the ADM decomposition?

If we know the metric, we can read off the quantities from the line element in the adapted coordinates

In adapted coordinates, the line element can be written as

$$ds^2 = -(\alpha^2 - \beta^i \beta_i)\, dt^2 + 2\beta_i\, dt\, dx^i + \gamma_{ij}\, dx^i\, dx^j$$

What is the ADM decomposition?

e.g. Schwarzschild

$$ds^2 = -\left(1 - \frac{2M}{r}\right) dt^2 + \left(1 - \frac{2M}{r}\right)^{-1} dr^2 + r^2 d\Omega^2$$

What is ...

$\alpha =$

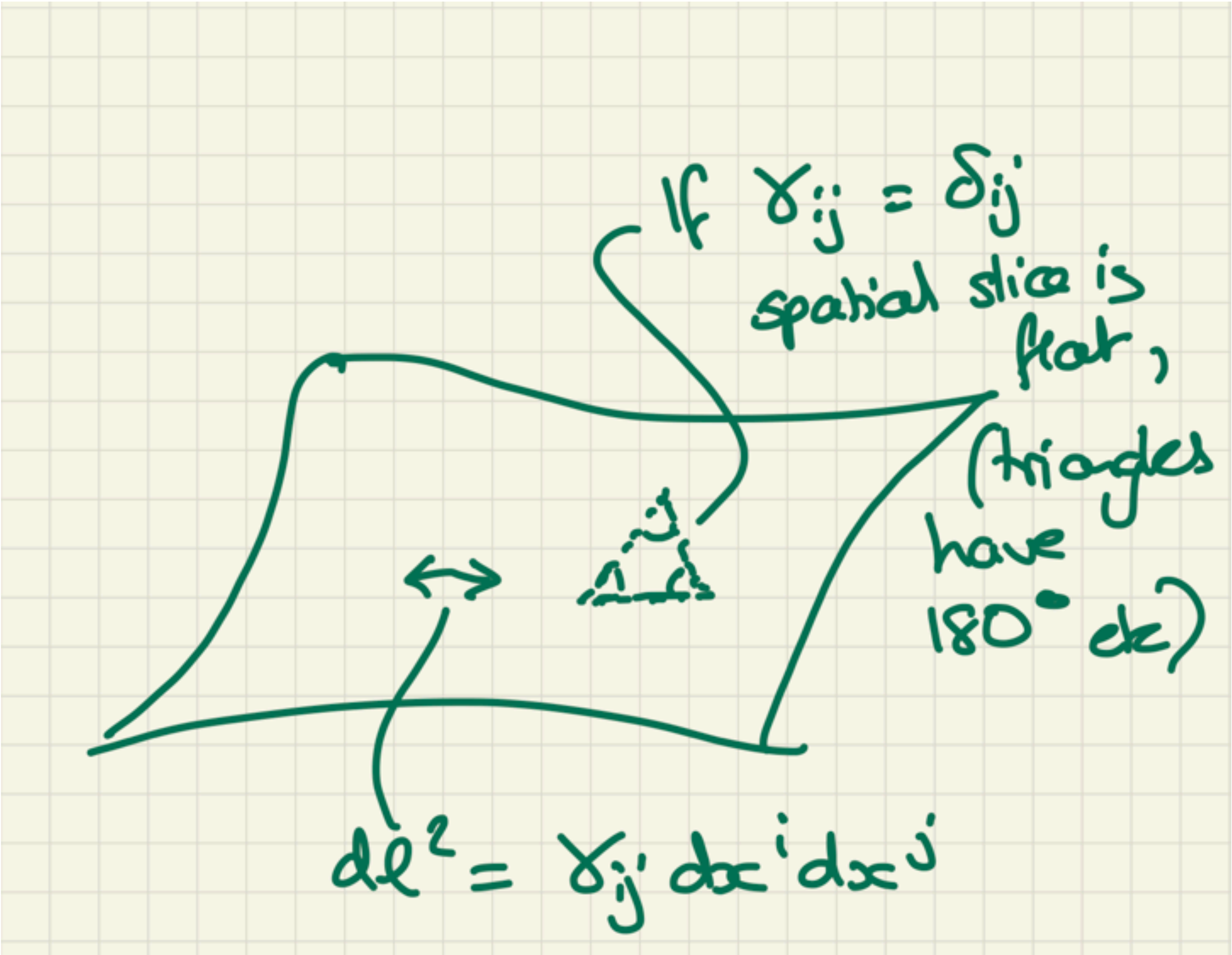
$\beta^i =$

$\gamma_{ij} =$

?

What (physically) is the spatial metric γ_{ij}?

The spatial metric tells us about proper distances on the spacelike hypersurface, which can be flat or curved



BSSN decomposition of the intrinsic curvature/spatial metric

We perform a conformal decomposition of the spatial metric into a conformal part and an overall conformal factor

$$\gamma_{ij} = e^{4\phi}\ \bar{\gamma}_{ij}$$

The rough motivation is to "factor out" any large overall stretching of spacetime (ie, around singularities) into the conformal factor
(at this point we haven't defined how exactly to make the split)

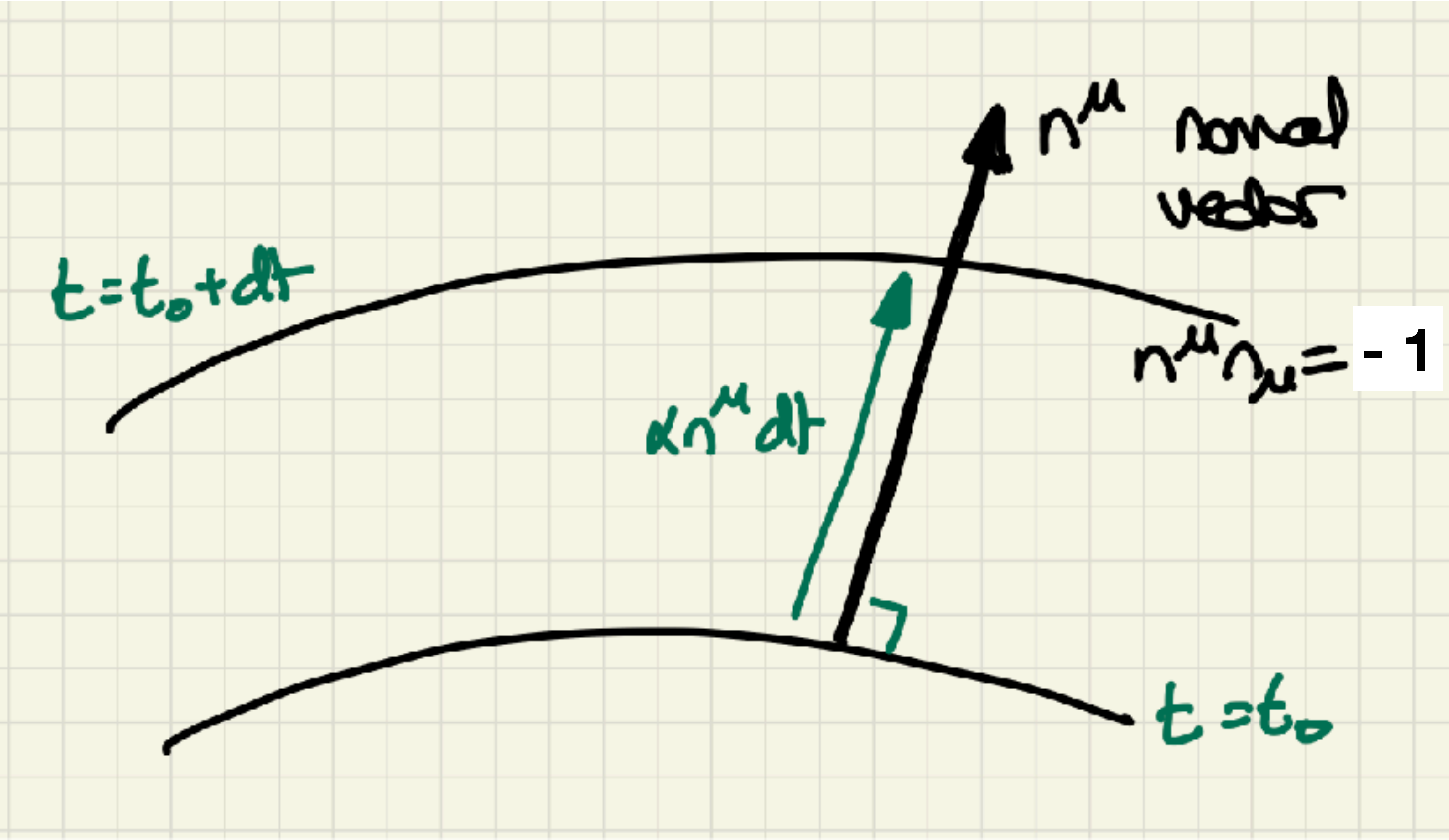
What relates to the spatial metric γ_{ij} in engrenage?

```python
#uservariables.py

# hard code number of ghosts to 3 here
num_ghosts = 3

# This file provides the list of (rescaled) variables to be evolved and
# assigns each one an index and its parity
# For description of the data structure see https://github.com/GRChombo/engrenage/wiki/Useful-code-background

idx_u       = 0    # scalar field
idx_v       = 1    # scalar field conjugate momentum (roughly the time derivative of u)
idx_phi     = 2    # conformal factor of metric, \gamma_ij = e^{4 \phi} \bar \gamma_ij
idx_hrr     = 3    # rescaled \epsilon_rr -> h_rr - deviation of rr component of the metric from flat
idx_htt     = 4    # rescaled \epsilon_tt -> h_tt - deviation of tt component of the metric from flat
idx_hpp     = 5    # rescaled \epsilon_pp -> h_pp - deviation of pp component of the metric from flat
idx_K       = 6    # mean curvature K
idx_arr     = 7    # rescaled \tilde A_rr -> a_rr - (roughly) time derivative of hrr
idx_att     = 8    # rescaled \tilde A_tt -> a_tt - (roughly) time derivative of htt
idx_app     = 9    # rescaled \tilde A_pp -> a_pp - (roughly) time derivative of hpp
idx_lambdar = 10   # rescaled \bar\Lambda -> lambda^r
idx_shiftr  = 11   # rescaled \beta^r -> radial shift - gauge variable for relabelling spatial points
idx_br      = 12   # rescaled B^r -> b^r - time derivative of shift
idx_lapse   = 13   # lapse - gauge variable for time slicing

```

What (physically) is the lapse α?

The lapse is related to how much **proper time** passes for an observer going to the next slice

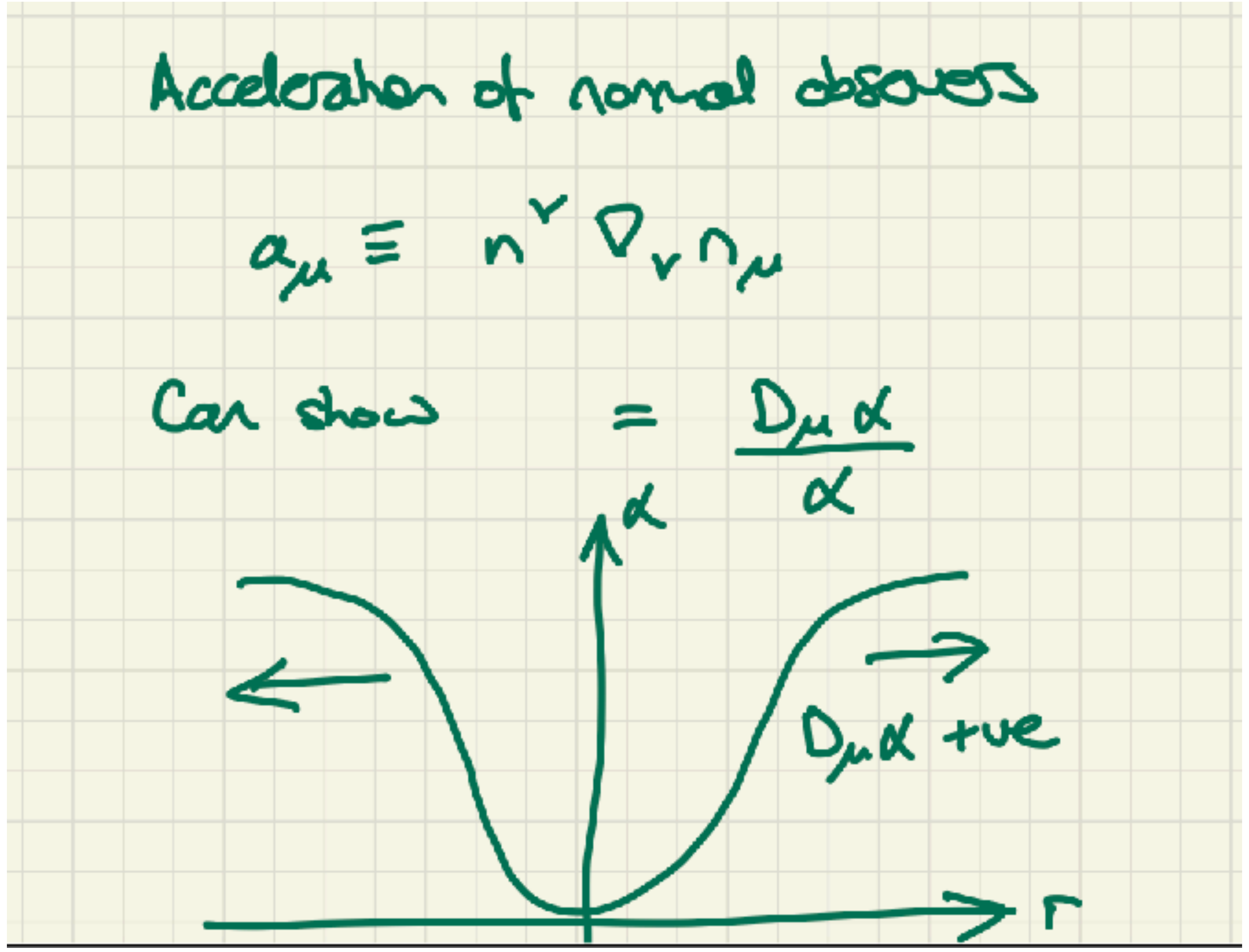
(not the full story - see *the shift* coming up soon)



What (physically) is the lapse α?

A spatially varying lapse indicates that the normal observers are **accelerated**

This is important for the stability of the puncture gauge in NR, where we will observe the "**collapse of the lapse**".

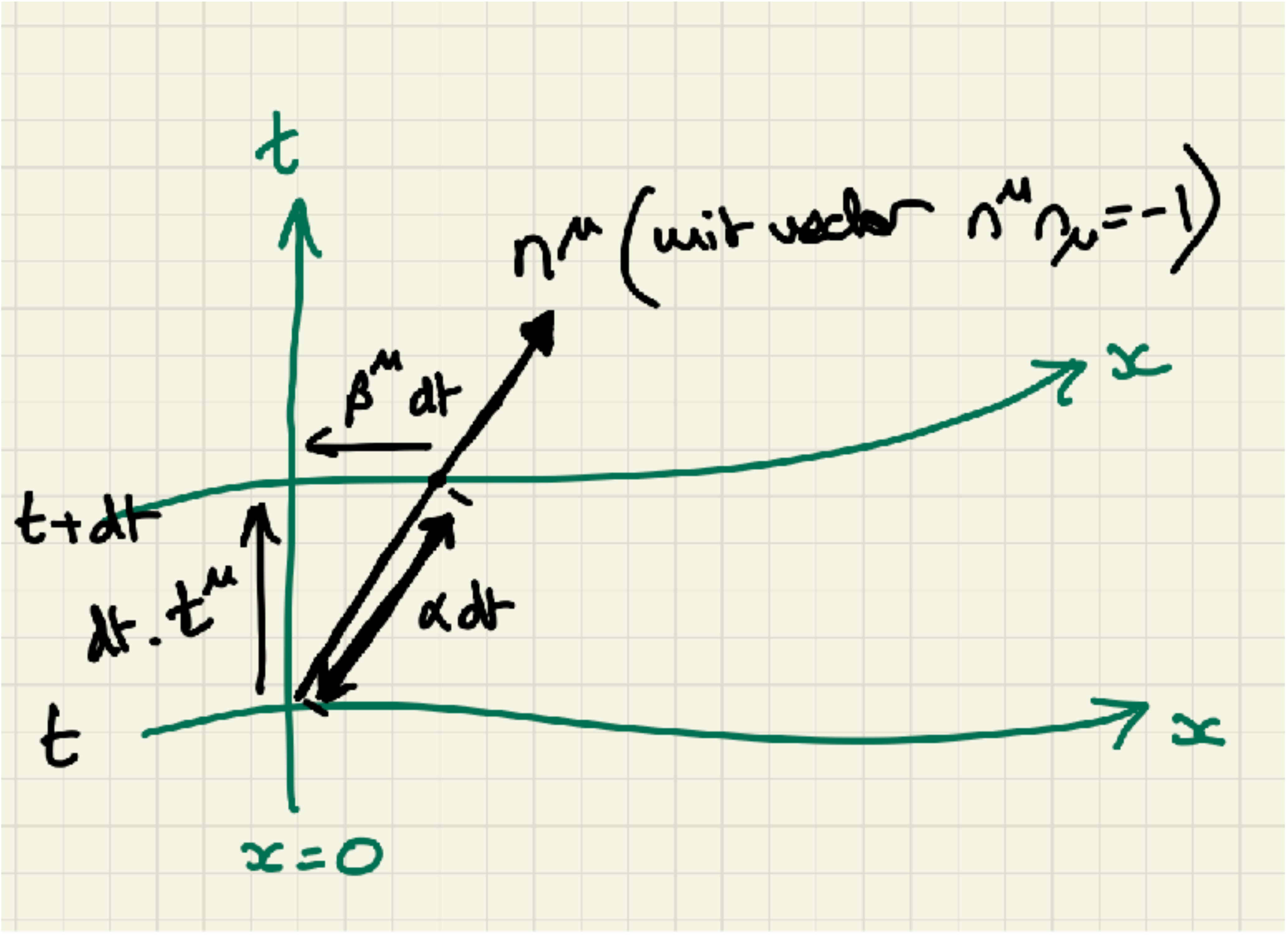


What relates to the lapse in engrenage?

```
#uservariables.py

# hard code number of ghosts to 3 here
num_ghosts = 3

# This file provides the list of (rescaled) variables to be evolved and
# assigns each one an index and its parity
# For description of the data structure see https://github.com/GRChombo/engrenage/wiki/Useful-code-background

idx_u       = 0    # scalar field
idx_v       = 1    # scalar field conjugate momentum (roughly the time derivative of u)
idx_phi     = 2    # conformal factor of metric, \gamma_ij = e^{4 \phi} \bar \gamma_ij
idx_hrr     = 3    # rescaled \epsilon_rr -> h_rr - deviation of rr component of the metric from flat
idx_htt     = 4    # rescaled \epsilon_tt -> h_tt - deviation of tt component of the metric from flat
idx_hpp     = 5    # rescaled \epsilon_pp -> h_pp - deviation of pp component of the metric from flat
idx_K       = 6    # mean curvature K
idx_arr     = 7    # rescaled \tilde A_rr -> a_rr - (roughly) time derivative of hrr
idx_att     = 8    # rescaled \tilde A_tt -> a_tt - (roughly) time derivative of htt
idx_app     = 9    # rescaled \tilde A_pp -> a_pp - (roughly) time derivative of hpp
idx_lambdar = 10   # rescaled \bar\Lambda -> lambda^r
idx_shiftr  = 11   # rescaled \beta^r -> radial shift - gauge variable for relabelling spatial points
idx_br      = 12   # rescaled B^r -> b^r - time derivative of shift
idx_lapse   = 13   # lapse - gauge variable for time slicing

```

What (physically) is the shift β^i?

The shift tells us about how we **relabel coordinates** from one slice to the next. I like to think of it as the amount the normal observers have to "jump" to get back to the coordinate they were on at the last time slice



Dynamical gauge

1. Lapse aims (roughly) to minimise K

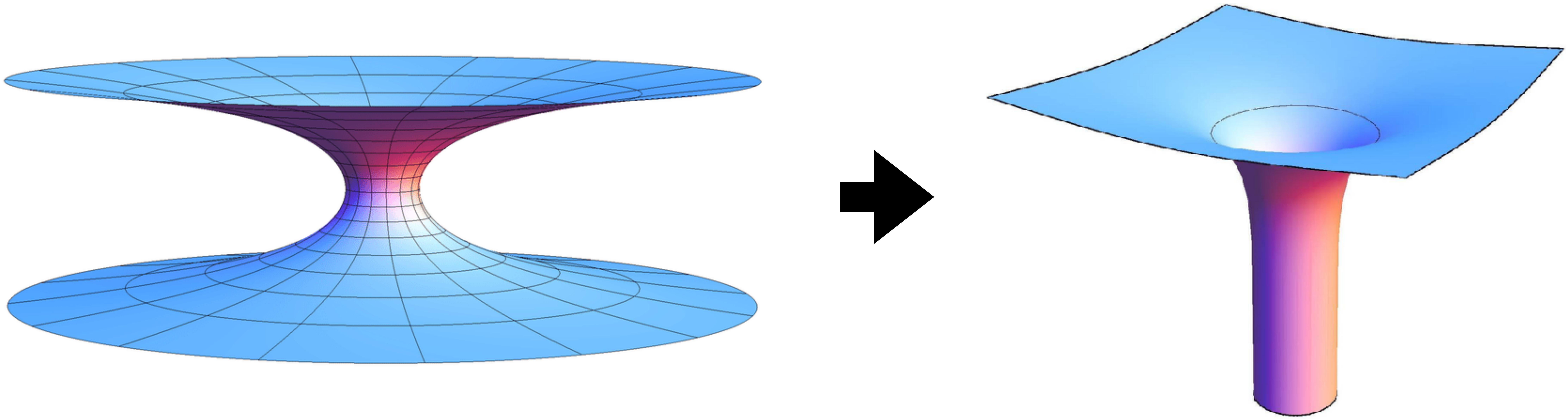
$$\partial_t \alpha \sim -2\alpha K$$

2. Shift aims (roughly) to minimise $\bar{\Gamma}^i = \bar{\gamma}^{jk}\bar{\Gamma}^i_{jk}$

$$\partial_t \beta^i \sim \bar{\Gamma}^i - \eta\beta^i$$

Black hole “punctures”

The initial “wormhole” metric evolves into a “trumpet” shape that terminates at a finite radius outside the singularity



What relates to the shift in engrenage?

```python
#uservariables.py

# hard code number of ghosts to 3 here
num_ghosts = 3

# This file provides the list of (rescaled) variables to be evolved and
# assigns each one an index and its parity
# For description of the data structure see https://github.com/GRChombo/engrenage/wiki/Useful-code-background

idx_u       = 0    # scalar field
idx_v       = 1    # scalar field conjugate momentum (roughly the time derivative of u)
idx_phi     = 2    # conformal factor of metric, \gamma_ij = e^{4 \phi} \bar \gamma_ij
idx_hrr     = 3    # rescaled \epsilon_rr -> h_rr - deviation of rr component of the metric from flat
idx_htt     = 4    # rescaled \epsilon_tt -> h_tt - deviation of tt component of the metric from flat
idx_hpp     = 5    # rescaled \epsilon_pp -> h_pp - deviation of pp component of the metric from flat
idx_K       = 6    # mean curvature K
idx_arr     = 7    # rescaled \tilde A_rr -> a_rr - (roughly) time derivative of hrr
idx_att     = 8    # rescaled \tilde A_tt -> a_tt - (roughly) time derivative of htt
idx_app     = 9    # rescaled \tilde A_pp -> a_pp - (roughly) time derivative of hpp
idx_lambdar = 10   # rescaled \bar\Lambda -> lambda^r
idx_shiftr  = 11   # rescaled \beta^r -> radial shift - gauge variable for relabelling spatial points
idx_br      = 12   # rescaled B^r -> b^r - time derivative of shift
idx_lapse   = 13   # lapse - gauge variable for time slicing
```

What (physically) is the extrinsic curvature K_{ij}?

The extrinsic curvature can be viewed in two equivalent ways:

1. It is related to the Lie Derivative of the spatial metric along the normal vector congruence

$$K_{ij} \equiv -\frac{1}{2}\mathcal{L}_n \gamma_{ij}$$

In this way it is related to the time derivative of the metric as

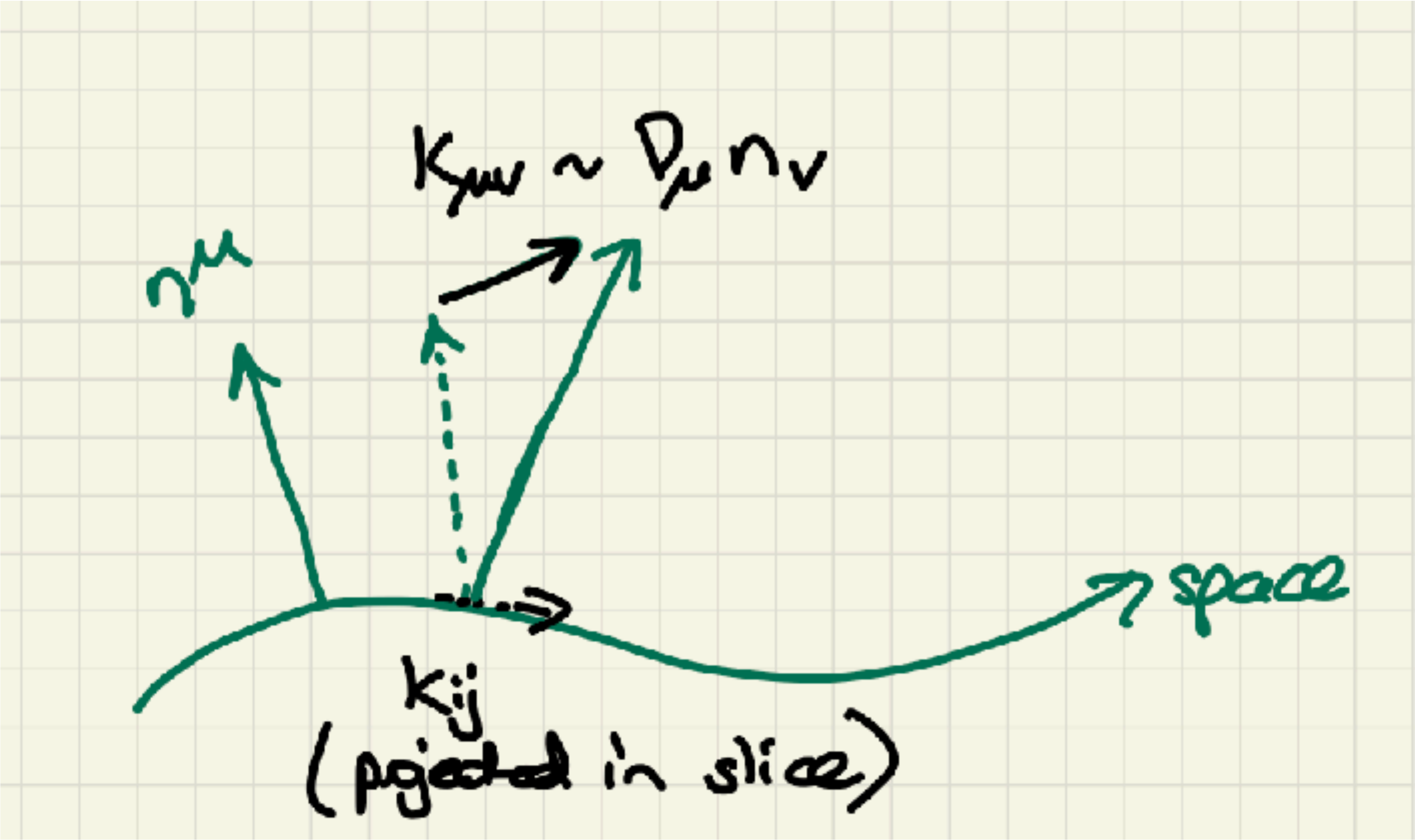
$$\partial_t \gamma_{ij} = -2\alpha K_{ij} + D_i\beta_j + D_j\beta_i$$

The extrinsic curvature can be viewed in two equivalent ways:

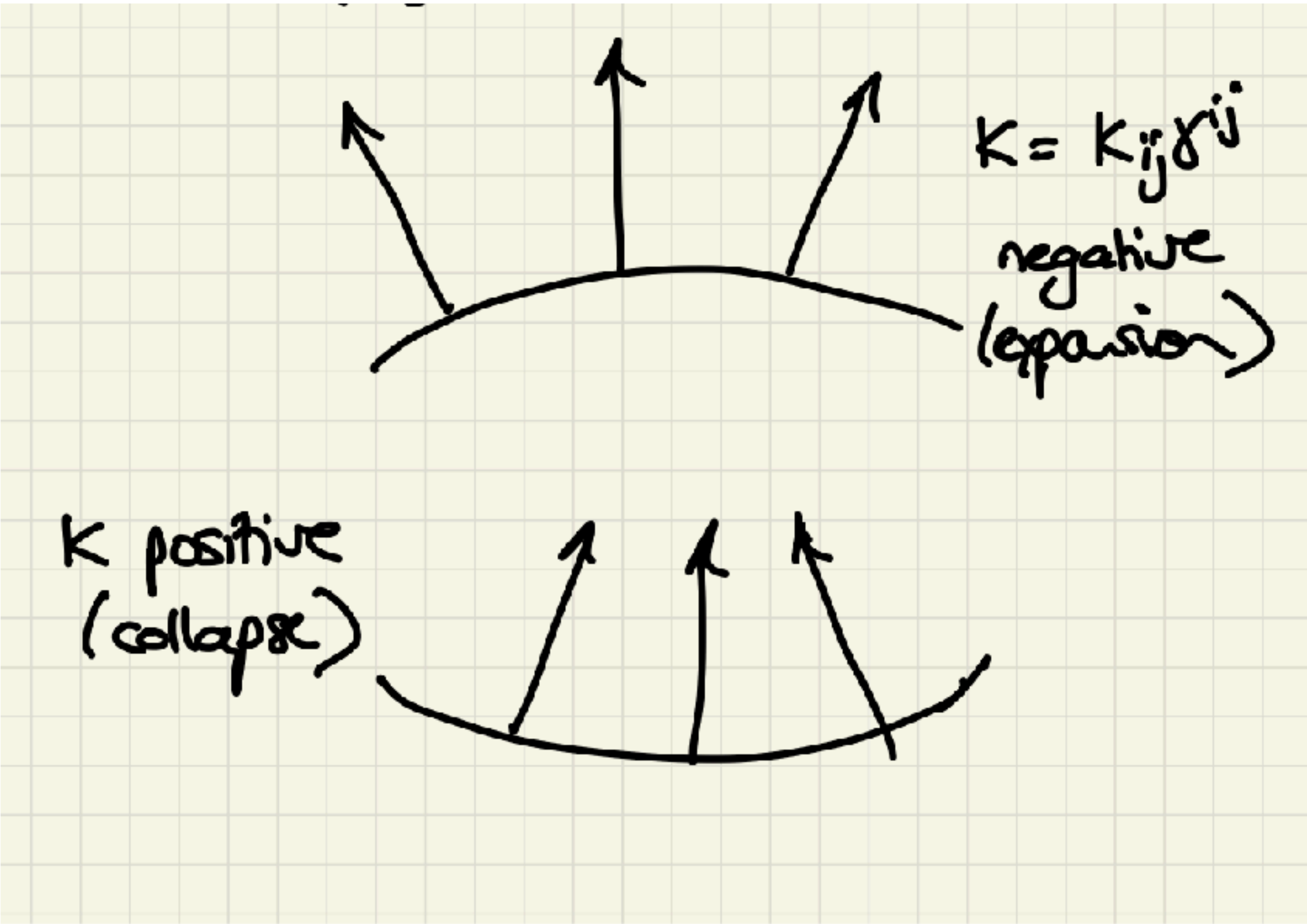
2. It is related to the (covariant) derivative of the normal vector projected into the spatial slice

What (physically) is the extrinsic curvature K_{ij}?

$$K_{ij} \equiv -\gamma^{\mu}_{i}\gamma^{\nu}_{j}\nabla_{\mu}n_{\nu}$$



What (physically) is the trace of the extrinsic curvature K?



BSSN decomposition of the extrinsic curvature

We perform a conformal decomposition plus a separation into trace and trace free parts as:

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

The rough motivation is to split out an overall expansion rate (the mean curvature K) and a traceless part relating to gravitational wave content

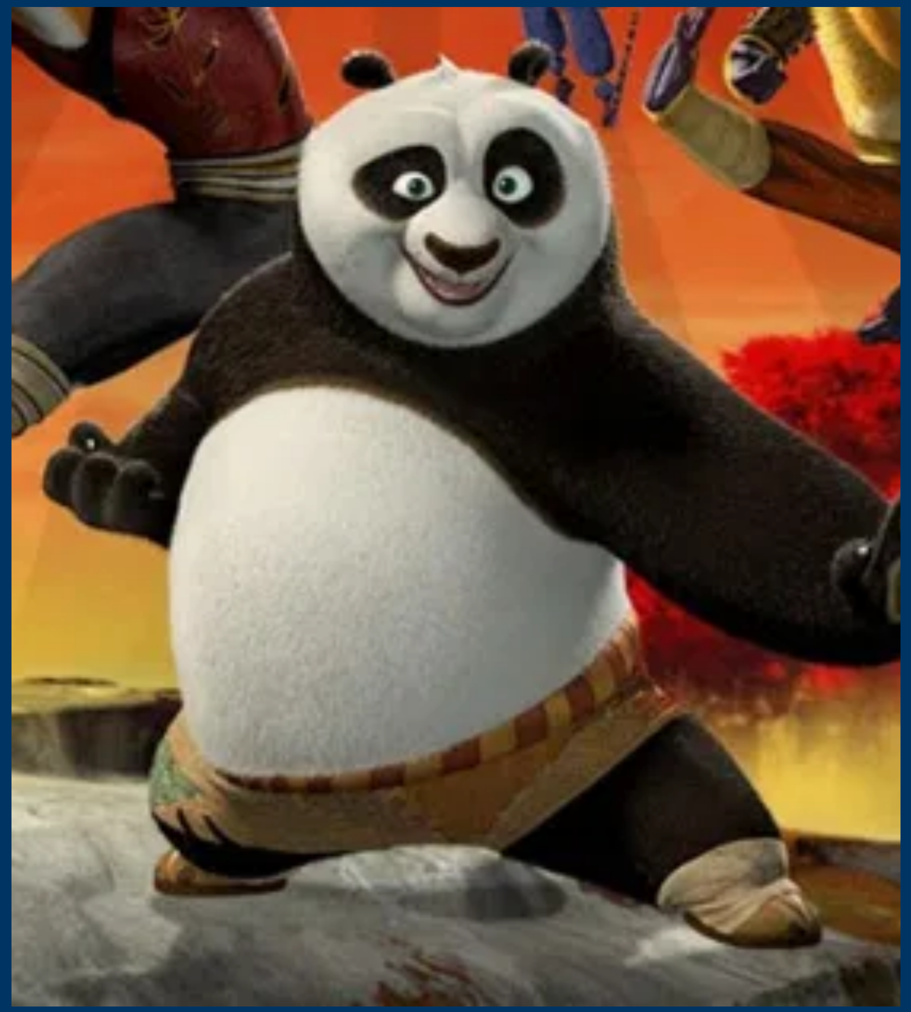
What relates to the K_{ij} in engrenage?

```python
#uservariables.py

# hard code number of ghosts to 3 here
num_ghosts = 3

# This file provides the list of (rescaled) variables to be evolved and
# assigns each one an index and its parity
# For description of the data structure see https://github.com/GRChombo/engrenage/wiki/Useful-code-background

idx_u       = 0    # scalar field
idx_v       = 1    # scalar field conjugate momentum (roughly the time derivative of u)
idx_phi     = 2    # conformal factor of metric, \gamma_ij = e^{4 \phi} \bar \gamma_ij
idx_hrr     = 3    # rescaled \epsilon_rr -> h_rr - deviation of rr component of the metric from flat
idx_htt     = 4    # rescaled \epsilon_tt -> h_tt - deviation of tt component of the metric from flat
idx_hpp     = 5    # rescaled \epsilon_pp -> h_pp - deviation of pp component of the metric from flat
idx_K       = 6    # mean curvature K
idx_arr     = 7    # rescaled \tilde A_rr -> a_rr - (roughly) time derivative of hrr
idx_att     = 8    # rescaled \tilde A_tt -> a_tt - (roughly) time derivative of htt
idx_app     = 9    # rescaled \tilde A_pp -> a_pp - (roughly) time derivative of hpp
idx_lambdar = 10   # rescaled \bar\Lambda -> lambda^r
idx_shiftr  = 11   # rescaled \beta^r -> radial shift - gauge variable for relabelling spatial points
idx_br      = 12   # rescaled B^r -> b^r - time derivative of shift
idx_lapse   = 13   # lapse - gauge variable for time slicing
```

End of lecture 1 -
you are now at level 1!

Lecture 2: Level one

Lecture 2: Four practical exercises



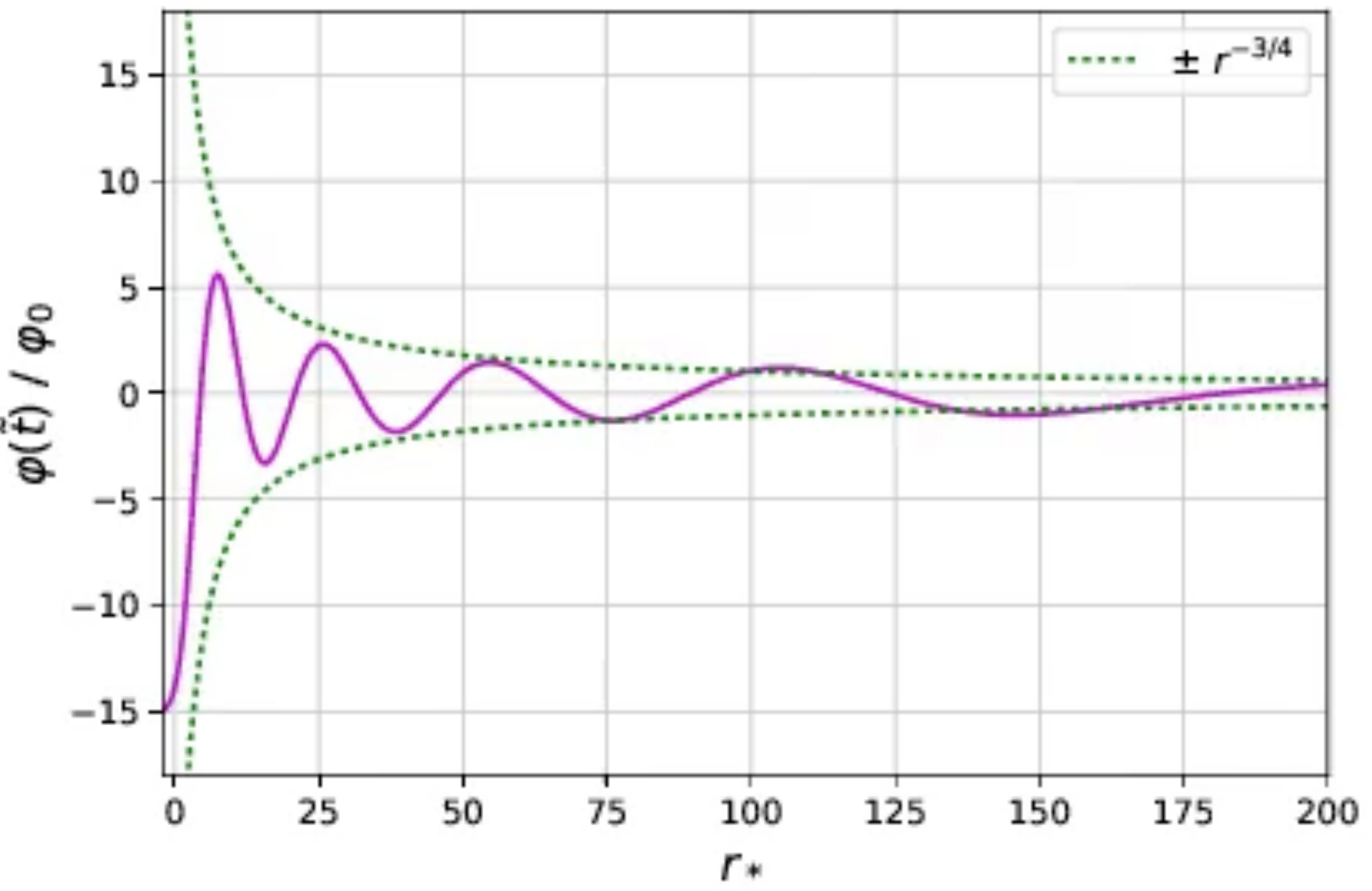
- Initial conditions - adding the scalar field to a BH
- Modifying equations of motion for the scalar
- Modifying the dynamical gauge for the metric
- Diagnostics - measuring scalar energy fluxes

Engrenage exercise 1: initial conditions

- Add a spatially constant scalar field $u_0 = 10^{-6}$ to the black hole initial conditions

- We need to make sure the Hamiltonian constraint is solved, so also set K to achieve this.

- ***Estimated lines of code required: 2-4***



https://inspirehep.net/literature/1731856

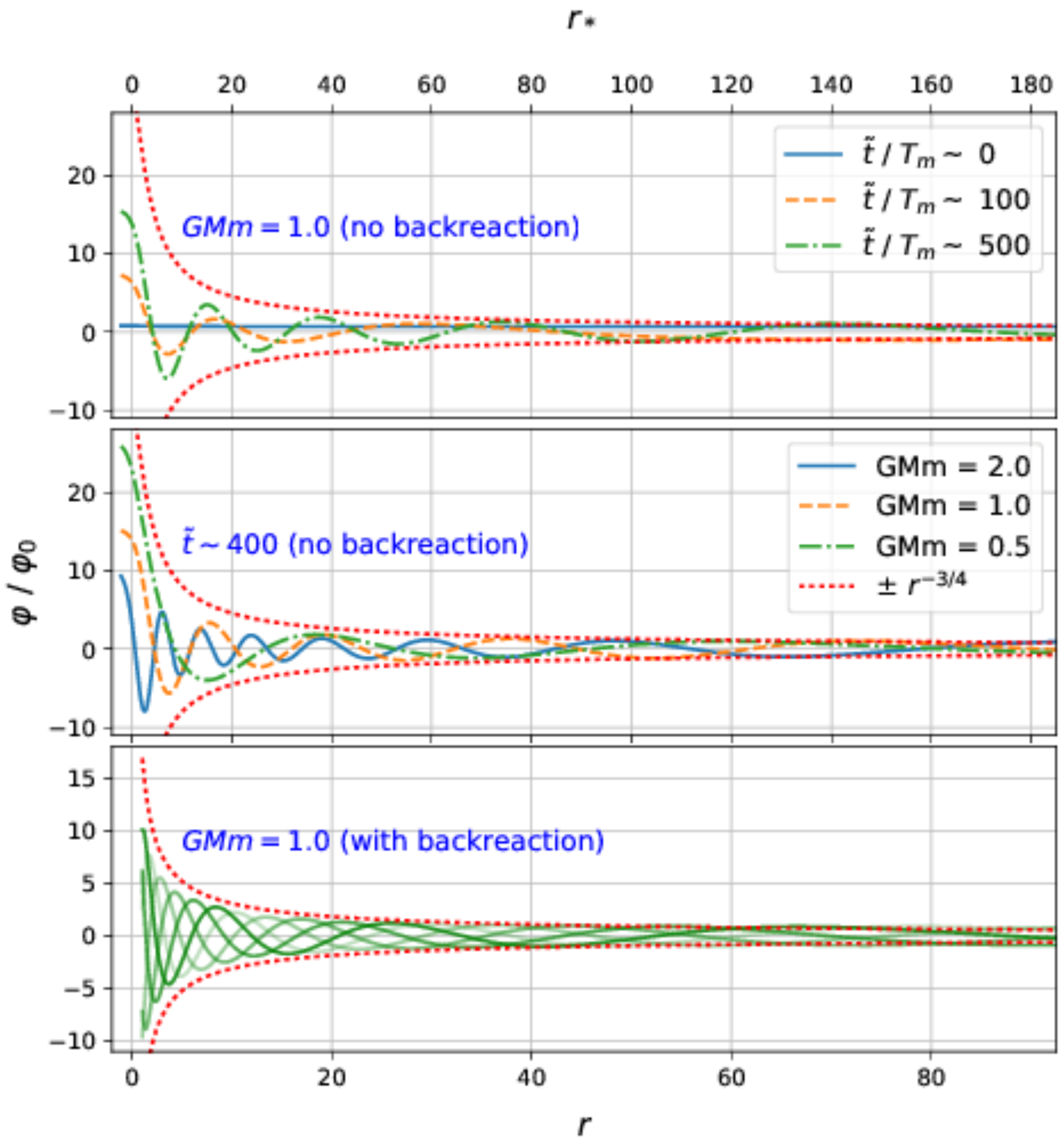
Engrenage exercise 2 - change the scalar eom

- Find and change the potential to:

$$V(u) = \frac{1}{2}\mu^2 u^2 + \frac{1}{4}\mu^2 \lambda u^4$$

Investigate the effect of changing the scalar mass μ and the self interaction λ.

- ***Estimated lines of code required: 2-3***



https://inspirehep.net/literature/1731856

Engrenage exercise 3 - change the gauge

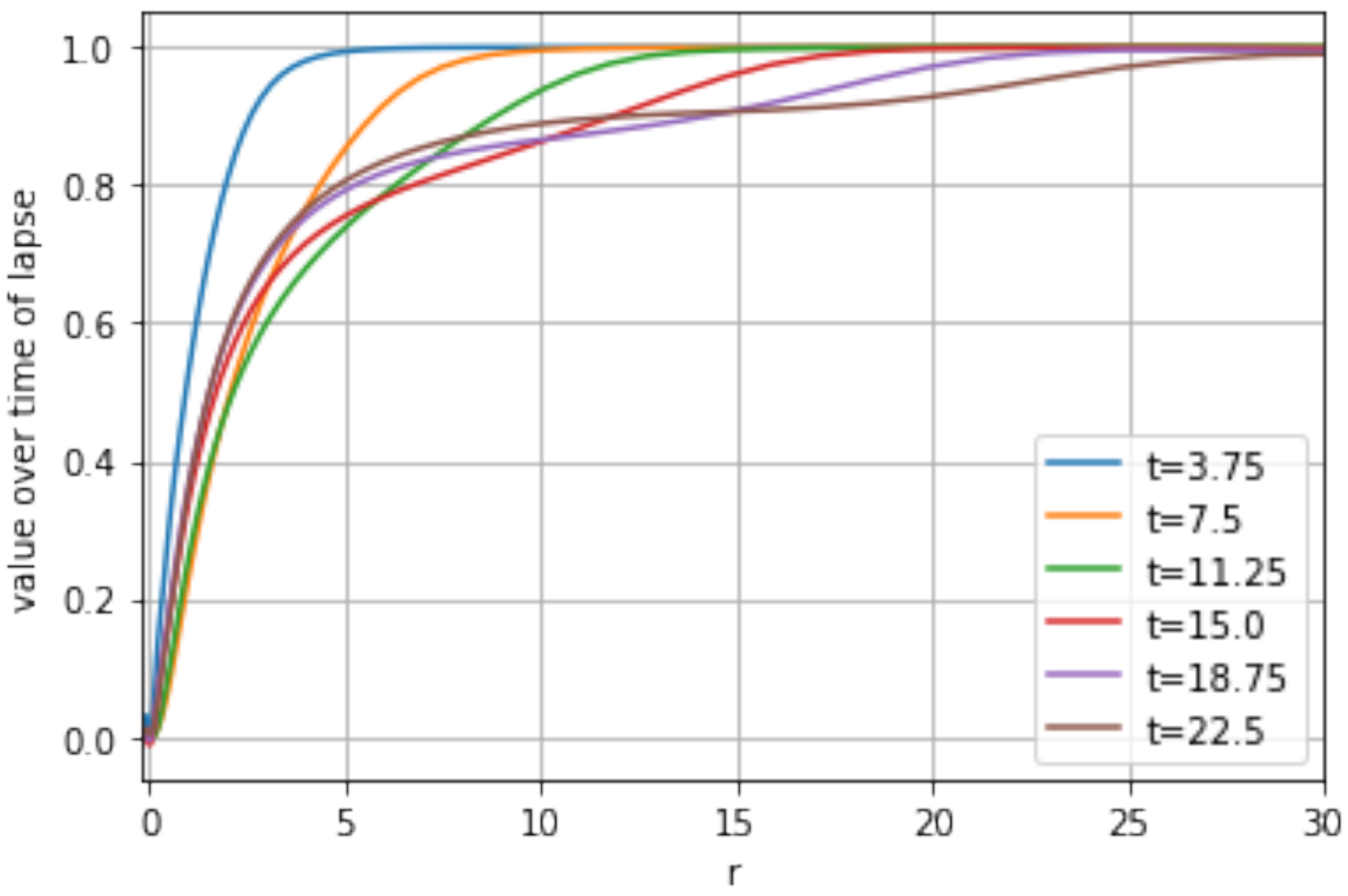
- Implement the shock avoiding gauge in https://inspirehep.net/literature/2111279

$$\partial_\tau \alpha = -\left(\alpha^2 + \kappa\right) K$$

with $\kappa = 0.05$

- What does it change about the evolution of the collapse of the lapse?

- How sensitive is stability to the choice of the parameter kappa?



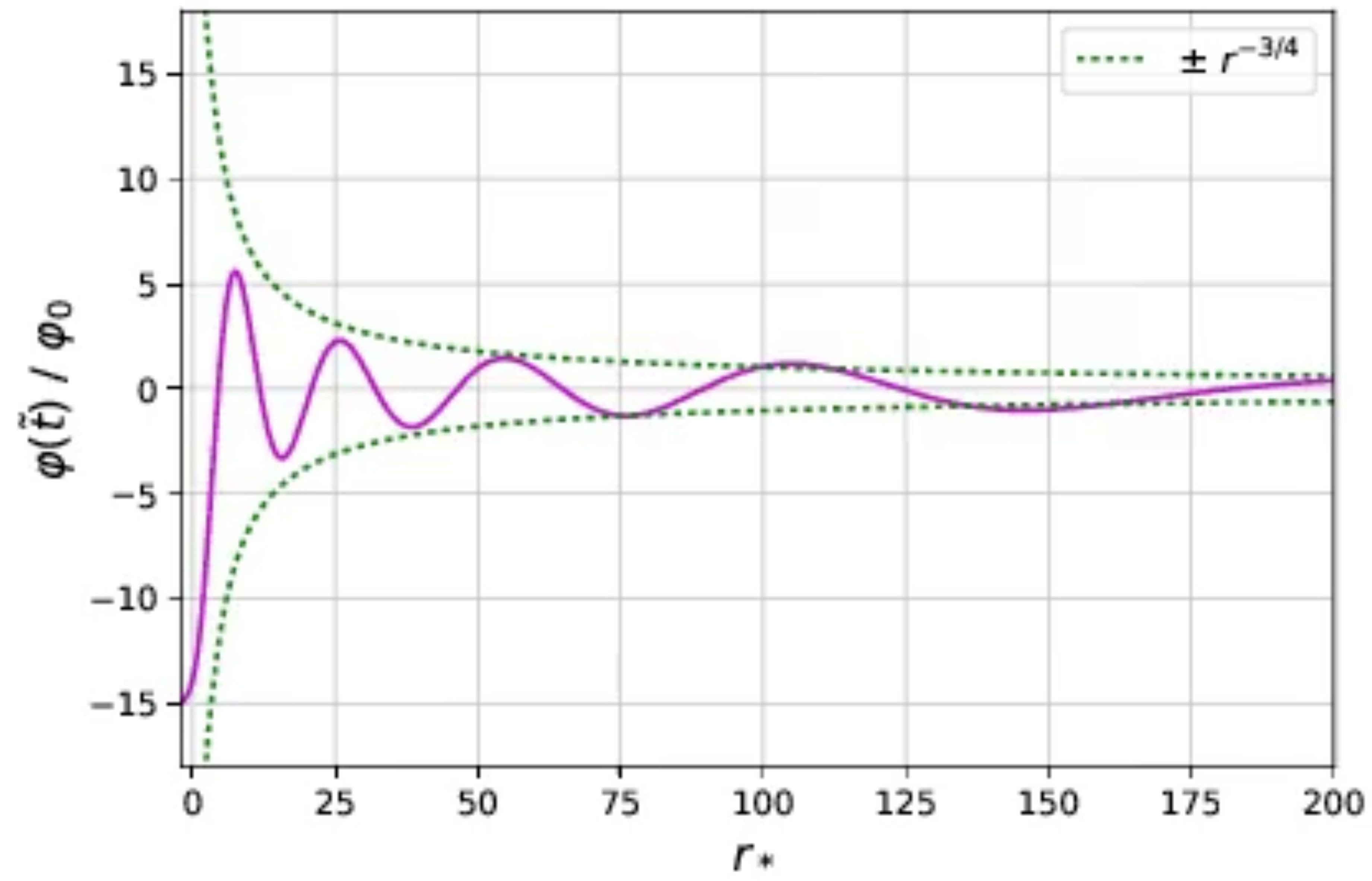
Engrenage exercise 4: diagnostics

- Write a diagnostic to calculate the radial flux across a spherical coordinate surface as a function of radius

$$F = 4\pi r^2 \sqrt{\gamma} S^r$$

- $S_i = -\, v \;\partial_j u$ is the momentum density of the scalar field

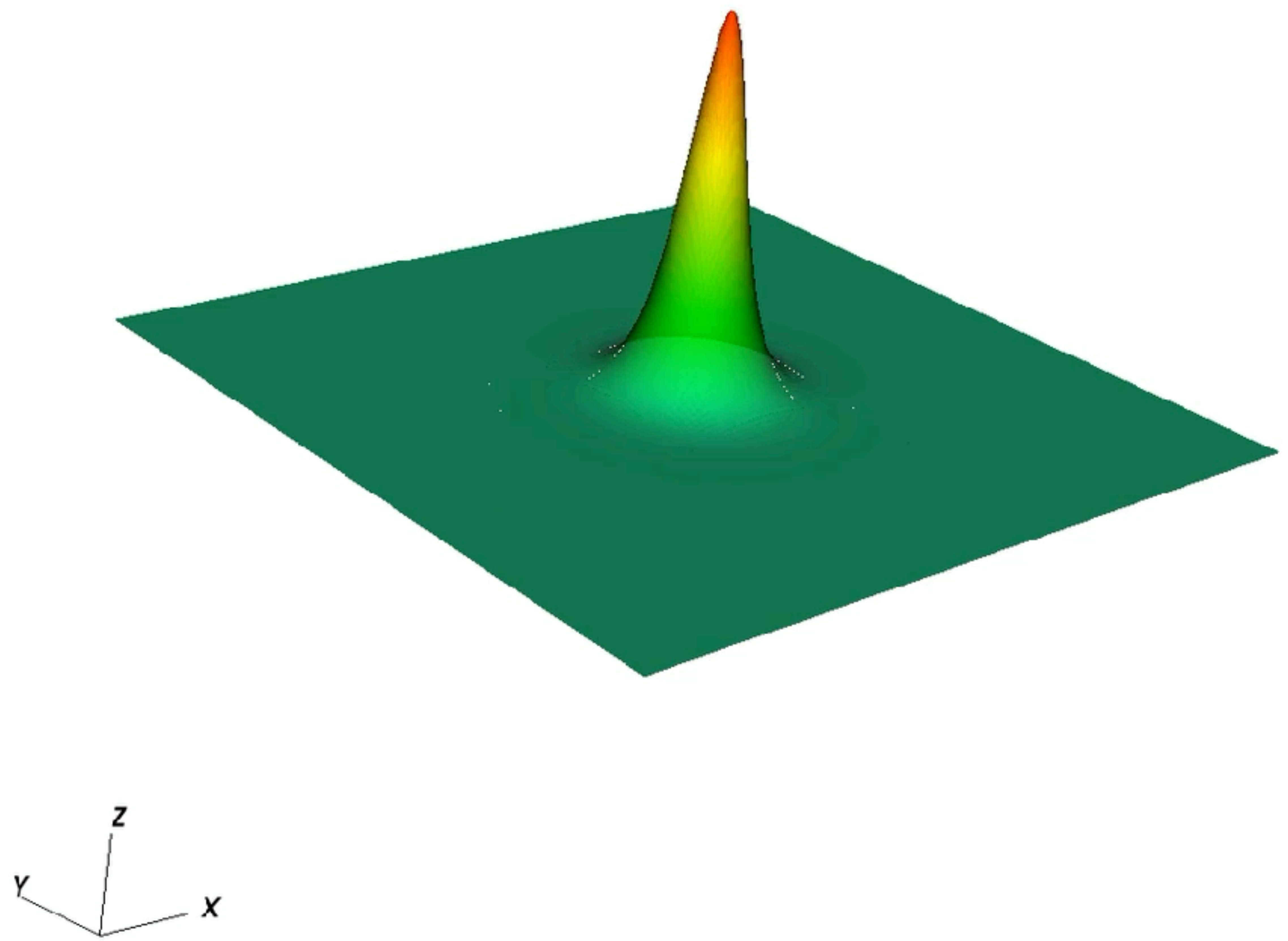
- What happens to the flux at small radii over time?



Extension - oscillaton

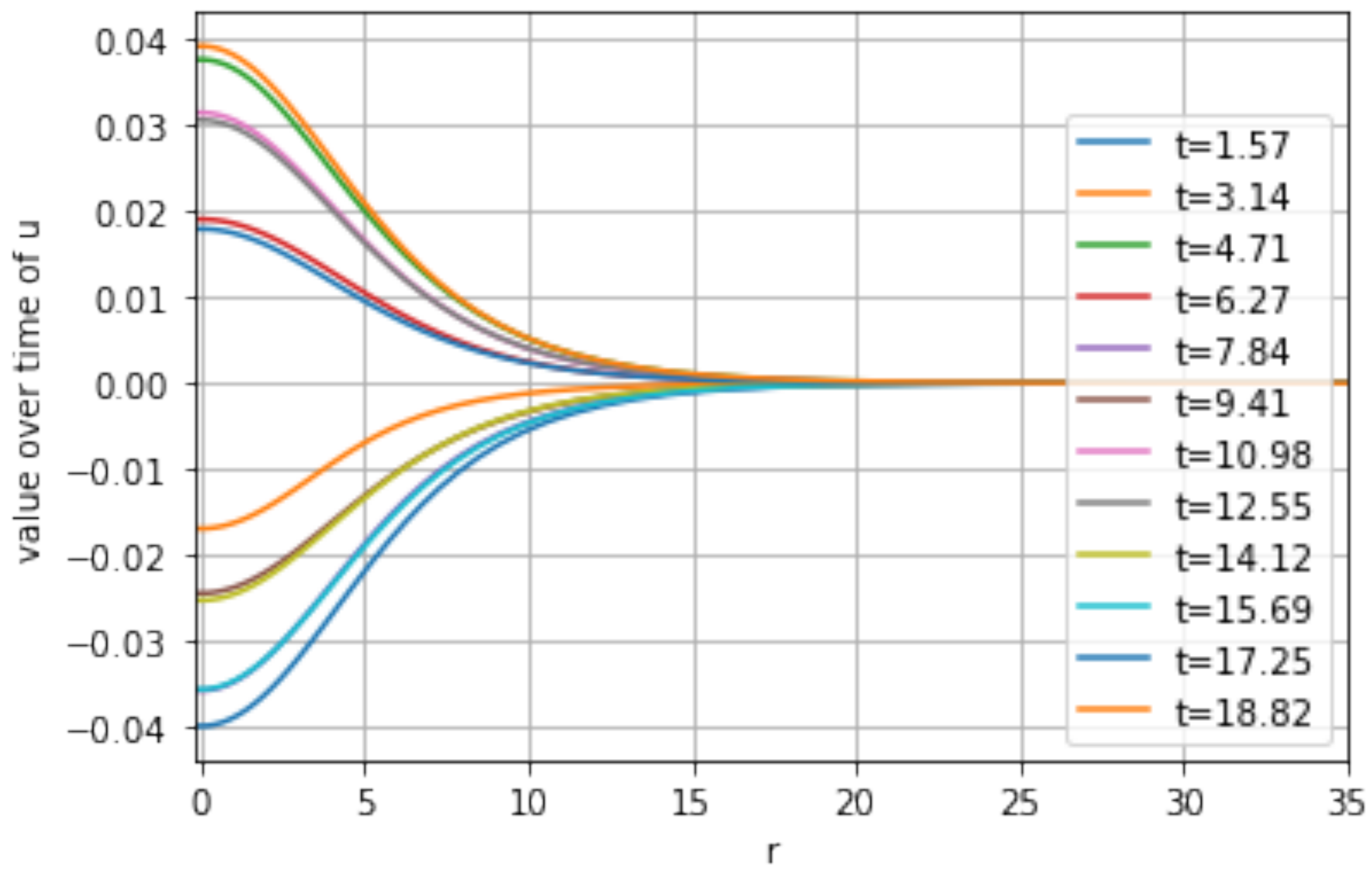
Engrenage oscillaton

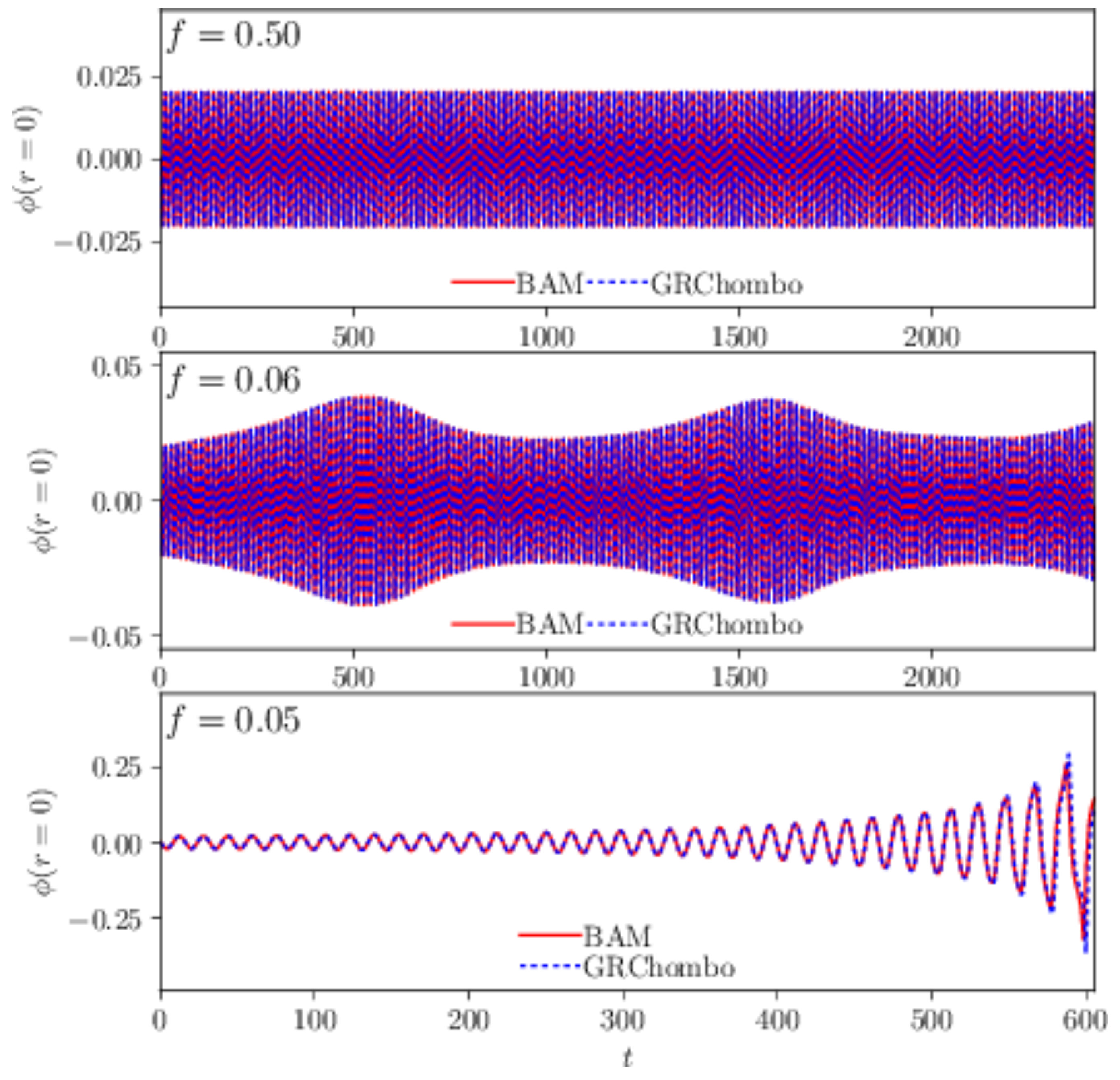
Field obeying massive Klein Gordon equation can have stable solitonic solutions with gravity



Engrenage oscillaton

- Repeat exercises 2-4 from the BH example for the oscillaton



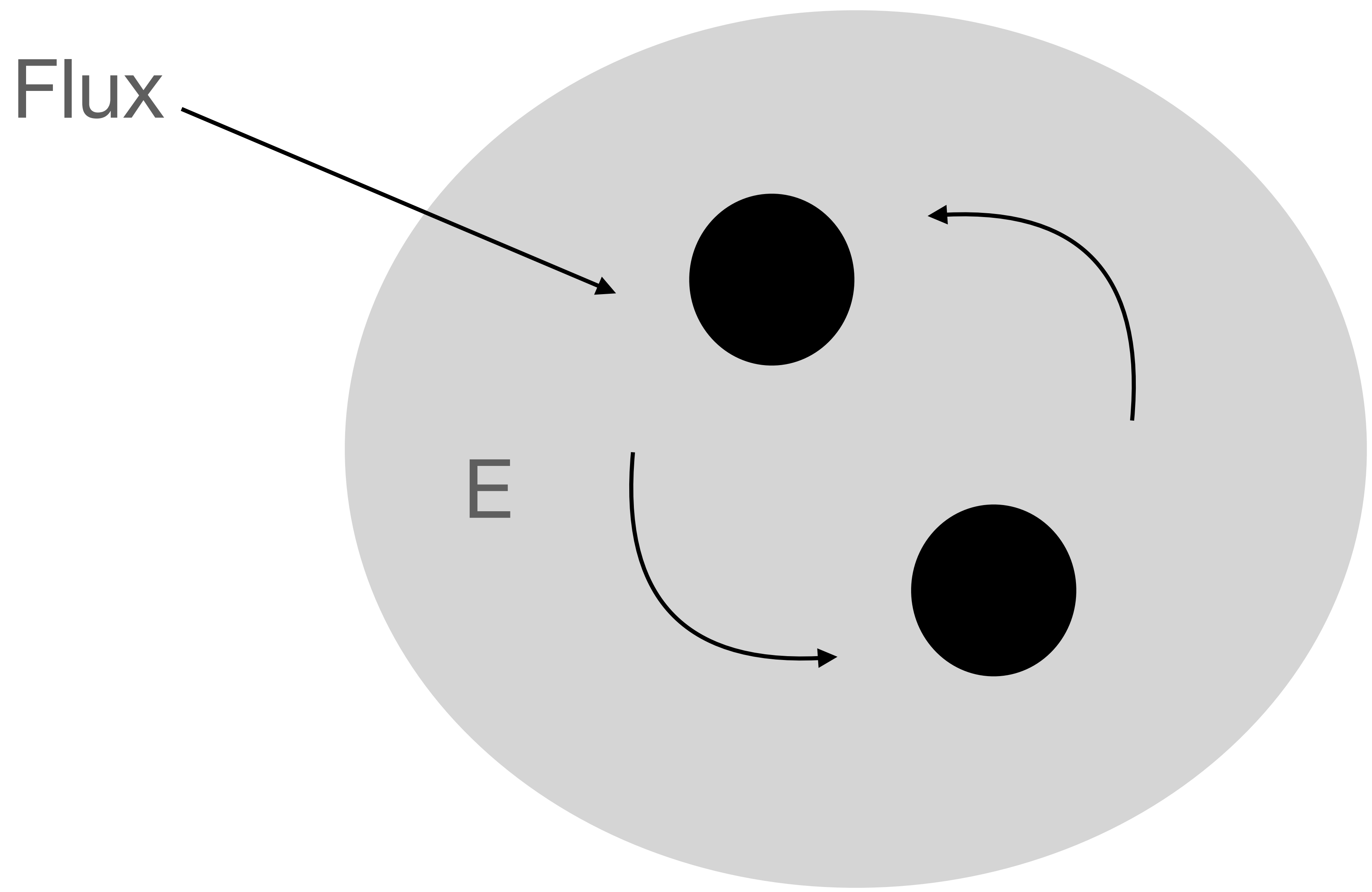


https://inspirehep.net/literature/1687181

Some useful technical points

Matter energy (non) conservation

Matter energy (non) conservation



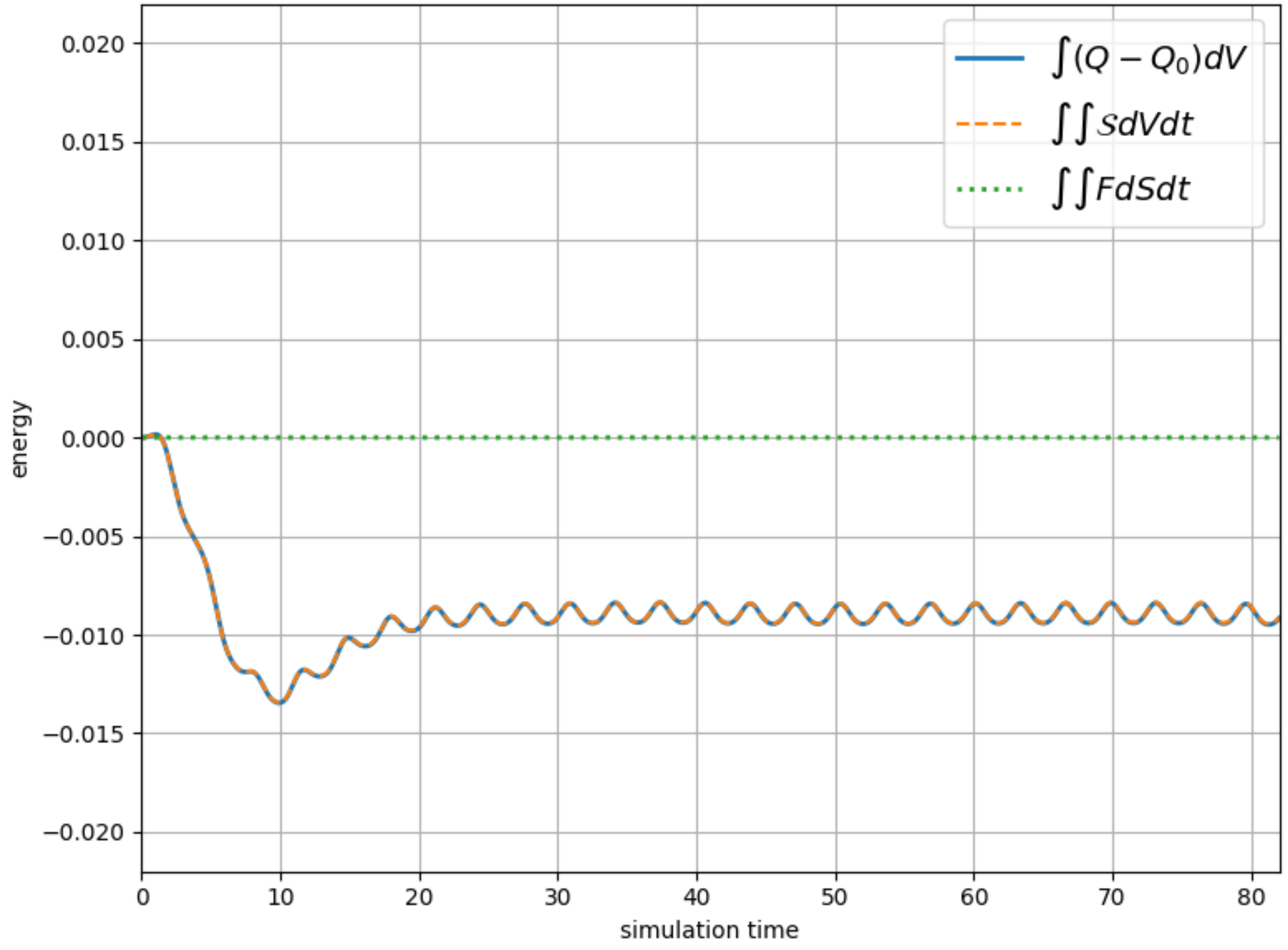
The global breakdown of energy conservation can be quantified as a source related to the curvature of the spacetime

$$\partial_t E = \text{Net Flux} + S$$

$$S \sim \int \Gamma^{\mu}_{\nu t} \, T^{\nu}_{\mu} \, dV$$

Matter energy (non) conservation

Oscillaton energy conservation



https://inspirehep.net/literature/1861156

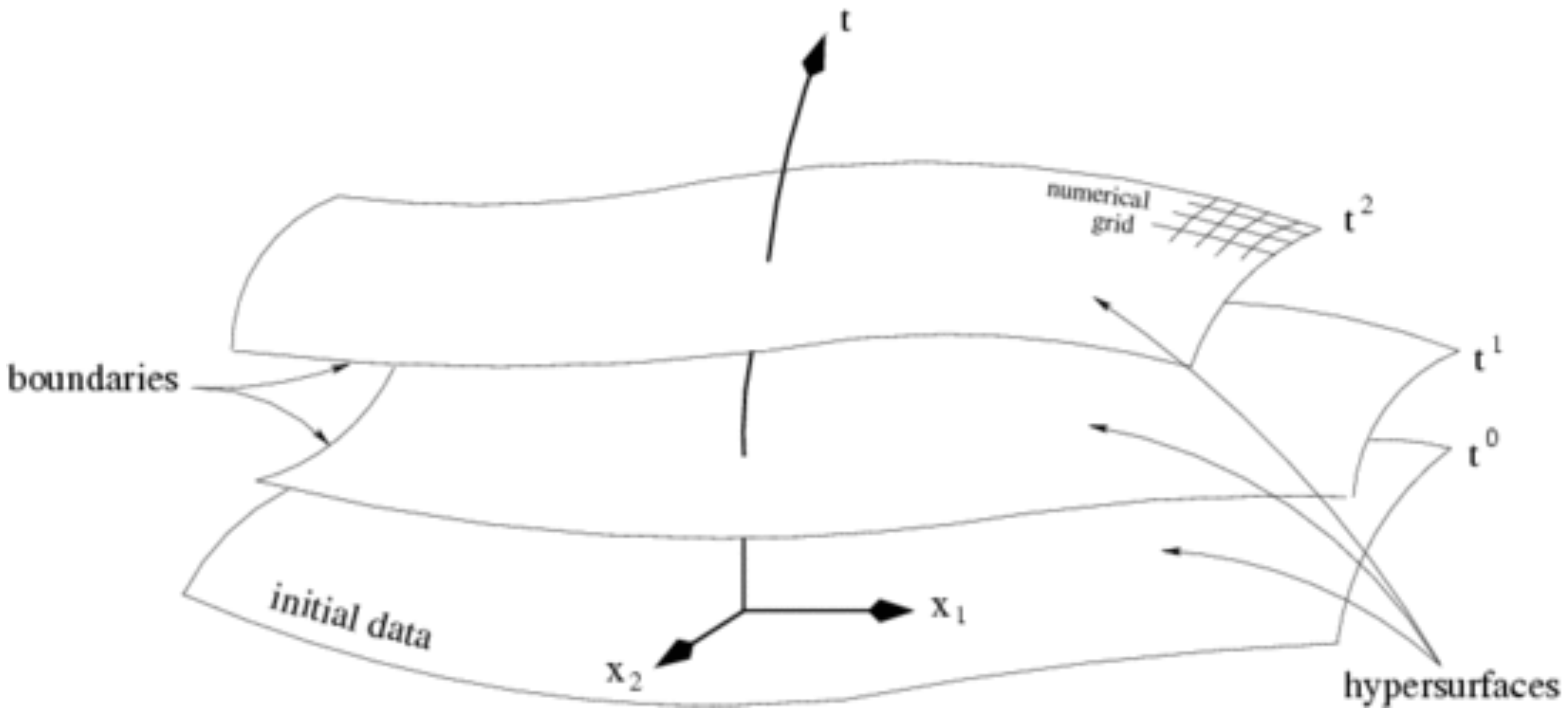
The global breakdown of energy conservation can be quantified as a source related to the curvature of the spacetime

$$\partial_t E = \text{Net Flux} + S$$

$$S \sim \int \Gamma^{\mu}_{\nu t}\, T^{\nu}_{\mu}\, dV$$

Gauge dependent quantities:
How can a “scalar” be gauge dependent?

Scalars



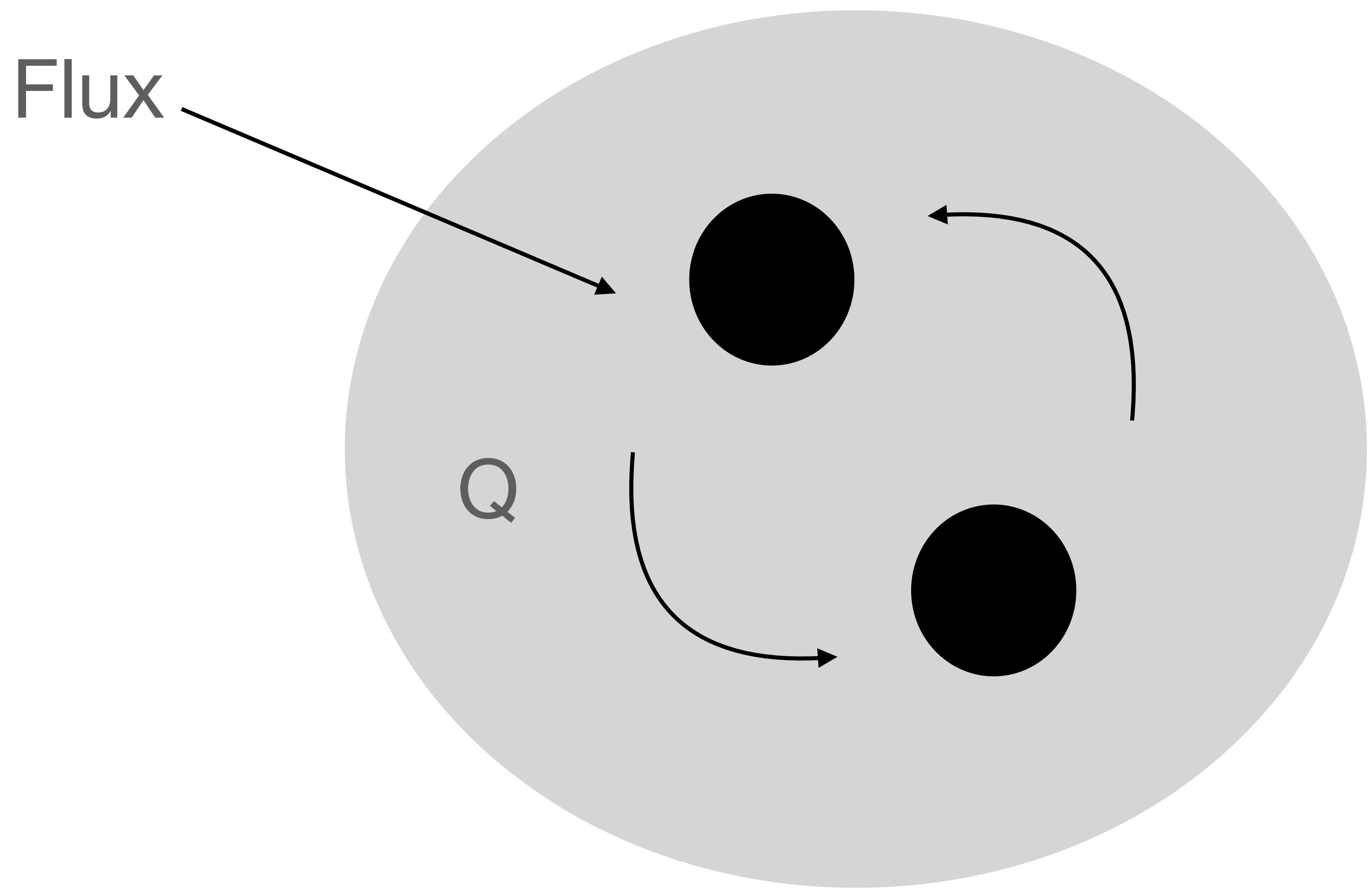
Consider the value of a scalar at some event E

Do all observers agree on the value of the scalar field u?

Do all observers agree on the value of the energy density?

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

Matter energy (non) conservation



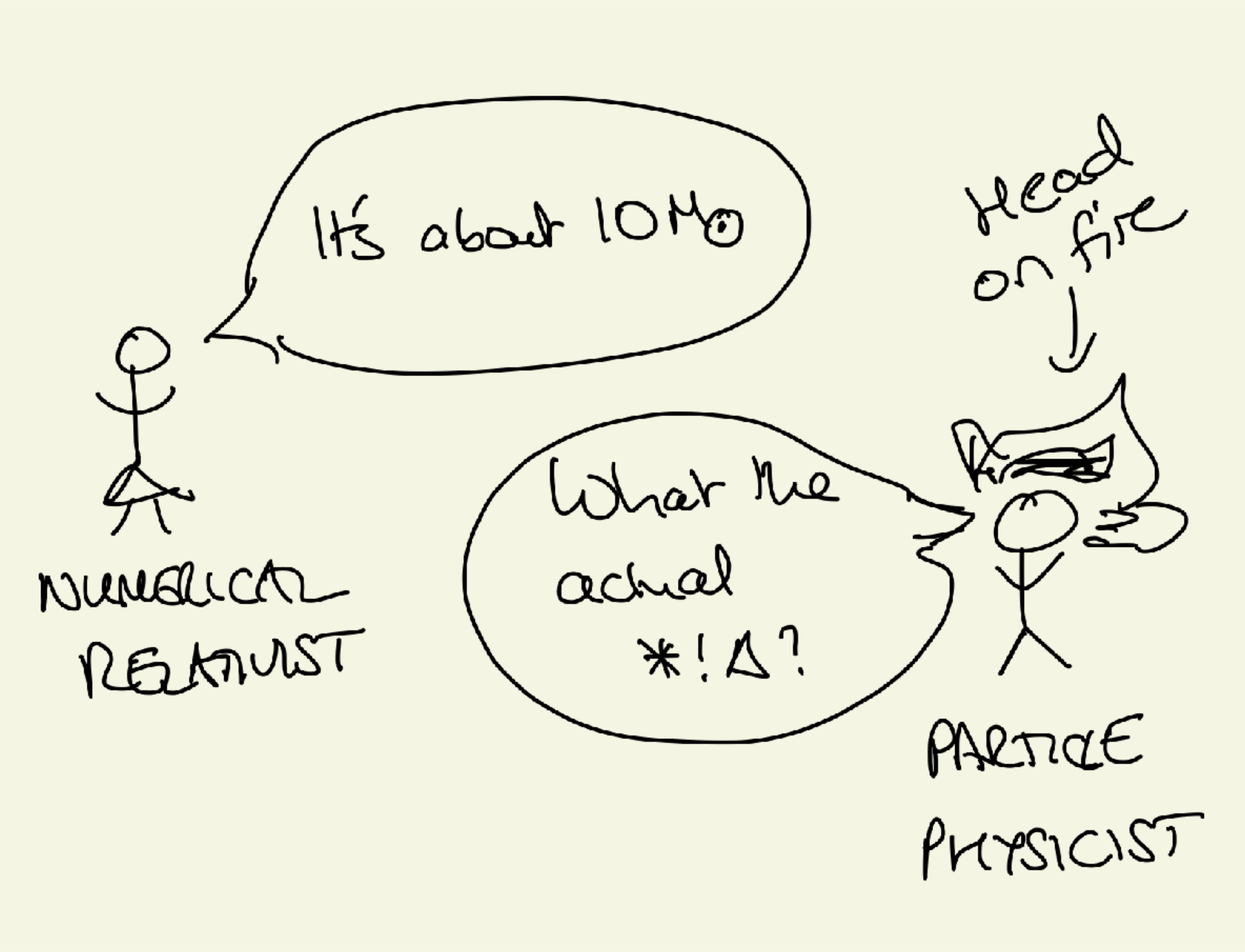
Is the integral over a spatial volume gauge dependent?

Who measures this?

Often more useful to say ***slicing dependent***

Units

What is the separation of the two black holes in your simulation?

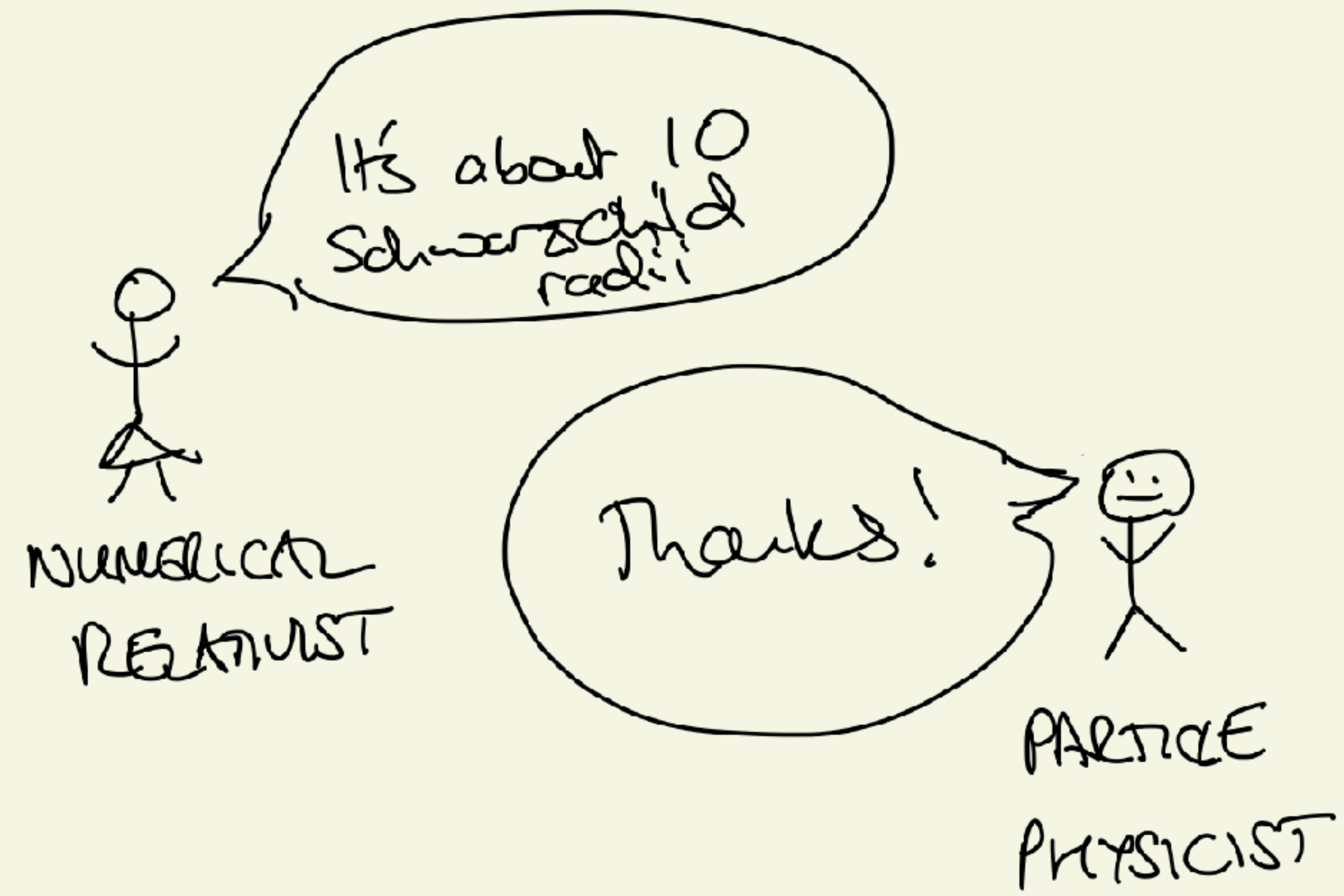


$G = c = 1$ for NR

THERE IS NO $\hbar$ in GR

If we set $M = M_{pl}$ then $\hbar$ is 1, but usually $\hbar \neq 1$, because $M = M_{\odot}$.

Usually we are describing a "curvature radius" not a mass.



You have now reached the end of the course, good luck in your research!

> Just remember to have fun, make mistakes, and persevere.

Advice from scipy.org